

REAL-TIME MONITORING AND FORECASTING OF TIME
SERIES IN HEALTHCARE APPLICATIONS

LAUREN ROBERTS

Thesis submitted for the degree of
Doctor of Philosophy



*School of Mathematics, Statistics & Physics
Newcastle University
Newcastle upon Tyne
United Kingdom*

January 2021

I dedicate this thesis to my parents, Sue and Chris. Thank you for always supporting and believing in me.

Acknowledgements

I would like to thank my supervisors, Dr Sarah Heaps and Prof Darren Wilkinson, for their invaluable support and guidance whilst researching and writing this thesis. Their enthusiasm for and understanding of statistics is inspiring and I have learned so much from them. Thank you to my third supervisor, Prof Paul Watson, along with the rest of the CDT team for giving me the opportunity to complete this thesis and for their practical and emotional support throughout. I would also like to thank the EPSRC for providing the sponsorship, Newcastle University Movement Laboratory research group for providing the data and Woodside Energy Ltd for letting me apply what I had learned in my research to another real-world project.

Of course, none of this would have been possible without the continued support from my family and friends, all of whom I am incredibly lucky to have around me. Thank you to my parents, Sue and Chris, my sister, Rebecca, and my long distance best friends, Charli, Lucy, Sophie and Natalie, who were always available at the other end of a call to spur me on. Thank you to Mac and Emily for providing many delicious dinners and plenty of laughs. Thank you to Newcastle University Pole Dancing Society for providing the perfect outlet and the countless friends I have made since joining, with special thanks to Emma for keeping me entertained both in and out of classes. Finally, thank you to my CDT friends, in particular Ashleigh, Hugo, Tom and Naomi, who reassured me along the way and were always free for a chat.

Abstract

Type II diabetes is an increasingly common disease, but one in which the effects suffered by patients, such as hyperglycaemia, can be improved through careful monitoring and control of the factors that influence blood glucose levels. Advances in the Internet of Things (IoT) have made monitoring a person's glucose levels more accessible, in that a continuous glucose monitoring (CGM) device in the form of a small sensor can be used to regularly report glucose levels to a bluetooth device, without the need for human intervention. Modelling the data from CGM devices online allows for short-term forecasts to be made that can assist in making real-time decisions regarding interventions to improve future glucose levels, such as behavioural changes. Additional data to monitor how active a person is can easily be collected by wrist-worn accelerometer devices. As activity levels directly impact glucose levels, bivariate models between glucose and activity data aim to provide improved forecasts.

State space models are fitted to glucose data and activity data using a Bayesian modelling framework. The posterior distributions of model parameters are learned via Markov chain Monte Carlo (MCMC) methods. High frequency (100 Hz), tri-axial accelerometer data are reported alongside glucose observations recorded at five minute intervals and are transformed into univariate activity summaries. Discrete-valued state space models, known as hidden Markov models (HMMs), are used to classify the observations from the different activity summaries into activity intensities. Normal and skew Normal within-state distributions are explored to better fit the observed activity summaries, as well as fitting models to transformations of the summaries where possible to reduce the skewness in the data.

Gaussian state space models, known as dynamic linear models (DLMs), are explored to describe glucose levels, incorporating seasonal and autoregressive (AR) components. The results from these models then provide the basis for bivariate models that incorporate known activity states. This additional information is included in the DLMs as a regression covariate, which is formed by a weighted sum of lagged activity zones. Models between glucose levels and lagged carbohydrate intake are also considered, to better understand the effects of activity and food on glucose levels.

A second application area is considered as an example of improved predictive performance where an influential variable is known alongside the quantity of interest. The production levels of liquid natural gas (LNG) at a gas plant are modelled by a DLM, with a regression on atmospheric temperature. The models are fitted in a frequentist framework for simplicity.

Contents

1	Introduction	1
1.1	Introduction and motivation for thesis	1
1.2	Glucose modelling	3
1.2.1	ARIMA models	3
1.2.2	Other models	5
1.2.3	Assessment of performance of models	6
1.3	Incorporating activity data	9
1.3.1	Background	9
1.3.2	Physiology	9
1.3.3	Pre-processing	11
1.4	Implications for modelling	13
1.5	Objectives of the thesis	17
1.6	Outline of the thesis	17
2	Exploratory data analysis	19
2.1	Background	19
2.2	Glucose data	19
2.2.1	Missing data	22
2.3	Accelerometer data	23
2.3.1	Pre-processing accelerometer data	23
2.4	Relationship between glucose and accelerometer data	34
2.5	Summary	37
3	Background	38
3.1	State space models	38
3.1.1	Hidden Markov models	39
3.1.2	Dynamic linear models	41
3.1.3	Model specifications for DLMS	42
3.1.4	Autoregressive models of order p	44

3.1.5	Dynamic linear regression	46
3.2	Bayesian inference	47
3.2.1	Markov chain Monte Carlo methods	49
3.2.2	Parameter inference	52
3.2.3	State inference	52
3.2.4	Inference for missing data	55
3.2.5	Data augmentation	56
3.3	Inference for HMMs	57
3.3.1	Parameter inference	57
3.3.2	State inference for HMMs	59
3.3.3	Numerical stability in a HMM	60
3.4	Inference for DLMs	62
3.4.1	Parameter inference	62
3.4.2	State inference for DLMs	64
3.4.3	Numerical stability in a DLM	69
3.5	Model checking	70
3.6	Summary	71
4	Models to classify and identify levels of activity	72
4.1	Background	72
4.2	Mixture models	73
4.2.1	Normal mixture model	73
4.3	Hidden Markov models	80
4.3.1	Normal HMM	81
4.3.2	Normal HMM with shared variance	84
4.3.3	Skew Normal HMM	87
4.4	Downsampling activity data	93
4.4.1	ENMO HMMs	94
4.4.2	BFEN HMMs	97
4.4.3	Step count HMMs	100
4.4.4	Comparison of results	106
4.5	Across-patient comparison	112
4.5.1	Comparison with annotated data	112
4.5.2	Activity zone comparison	114
4.6	Summary	116
5	Monitoring and forecasting glucose levels	118
5.1	Background	118
5.2	Autoregressive models	118

5.2.1	Results	120
5.3	Dynamic linear models	126
5.3.1	AR(p) DLM	126
5.3.2	Fourier seasonal model with local level component	136
5.3.3	A constrained variance model	139
5.3.4	AR(p) Fourier DLMS	141
5.4	Comparison of models	154
5.5	Across-patient comparison	155
5.6	Summary	157
6	Bivariate glucose models	158
6.1	Background	158
6.2	Further glucose and activity exploratory analysis	158
6.2.1	Glucose vs activity whilst fasting	160
6.3	DLMS for glucose data with known activity states	161
6.3.1	Local Fourier weighted regression model	162
6.3.2	A Gamma curve regression	167
6.3.3	Across-patient and model comparison	172
6.4	Modelling eating effects	177
6.4.1	The relationship between glucose and carbohydrates	178
6.4.2	Local Fourier eating effects model	178
6.5	Summary	181
7	Monitoring the production of liquid natural gas	184
7.1	Background	184
7.2	Exploratory data analysis	185
7.2.1	Production vs weather-related tags	186
7.3	DLMS for production data	188
7.3.1	Results	189
7.4	Modelling production with temperature	190
7.4.1	Results	192
7.4.2	Forecast sensitivity	193
7.5	Model improvements and further work	194
7.6	Summary	195
8	Conclusions and future work	197
8.1	Objectives and contributions of this thesis	197
8.2	Conclusions	198
8.3	Further work	202

List of Figures

- 2.1 Glucose profiles of eight type II diabetic patients. The euglycaemic range is indicated by the red, dashed lines on each profile. 20
- 2.2 Densities of the patient glucose profiles. 22
- 2.3 Tri-axial accelerometer data (thinned by 20 for plotting). 24
- 2.4 Accelerometer data summarised using the ENMO algorithm, Algorithm 2.1. 25
- 2.5 Accelerometer data summarised using the truncated ENMO algorithm, Algorithm 2.2. 26
- 2.6 Raw x -axis accelerometer data (top) filtered using a band-pass filter (second), high-pass filter (third) and low-pass filter (bottom) (thinned by 20 for plotting). 27
- 2.7 Accelerometer data summarised using the BFEN algorithm, Algorithm 2.3. 29
- 2.8 Accelerometer data summarised using the HFEN algorithm, Algorithm 2.4. 29
- 2.9 Accelerometer data summarised using a low-pass filter and EN. 30
- 2.10 Accelerometer data summarised using the HFEN₊ algorithm, Algorithm 2.5. 31
- 2.11 Accelerometer data summarised using the simple step count, Algorithm 2.6. 32
- 2.12 Accelerometer data summarised using the complete step count, Algorithm 2.7. 34
- 2.13 Correlation between glucose and lagged activity levels with 95% confidence interval error bars (plotted at five minute intervals), described by BFEN (blue), ENMO (gold) and the complete step count (red). 35
- 2.14 Correlation between change in glucose levels and lagged activity levels with 95% confidence interval error bars (plotted at five minute intervals), described by BFEN (blue), ENMO (gold) and the complete step count (red). 36

- 3.1 A DAG showing the dependency structure between the hidden states, $\{\theta_t\}$, and observations, $\{Y_t\}$ in a state space model. 38

- 4.1 Normal mixture model posterior empirical within-state densities for low (blue), medium (gold) and high (red) activity zones fitted to 10,000 BFEN observations (grey density). 78

4.2	Portion of BFEN data classified using a Normal mixture model into low (blue), medium (gold) and high (red) activity zones (thinned by 20 for plotting).	78
4.3	Normal HMM posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to 1 Hz BFEN data, compared to the BFEN density (grey).	82
4.4	MCMC output for μ_2 (blue) and μ_3 (magenta) in Model 4.2.	83
4.5	1 Hz BFEN data classified using a Normal HMM into low (blue), medium (gold) and high (red) levels of activity.	83
4.6	1 Hz ENMO data classified using a Normal HMM into low (blue), medium (gold) and high (red) levels of activity.	85
4.7	1 Hz BFEN data classified using a Normal HMM with shared within-state variance into low (blue), medium (gold) and high (red) levels of activity.	86
4.8	Normal HMM, with shared within-state variance, posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to 1 Hz BFEN data, compared to the actual BFEN density (grey).	86
4.9	Skew Normal HMM, with shared within-state scale parameter, posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to 1 Hz BFEN data, compared to the actual BFEN density (grey).	92
4.10	1 Hz BFEN data classified using a skew Normal HMM with shared within-state scale parameter into low (blue), medium (gold) and high (red) levels of activity.	93
4.11	Accelerometer data summarised using the ENMO algorithm aggregated over one second intervals (top – thinned by 20 for plotting) and aggregated over five minute intervals (bottom).	95
4.12	Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to ENMO data, compared to the actual ENMO density (grey).	96
4.13	ENMO data classified using a Normal HMM (left) and a skew Normal HMM (right) into low (blue), medium (gold) and high (red) levels of activity.	96
4.14	Accelerometer data summarised using the BFEN algorithm, aggregated over one second intervals (top – thinned by 20 for plotting) and aggregated over five minute intervals (bottom).	97
4.15	Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to BFEN data, compared to the actual BFEN density (grey).	99

4.16	Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to log BFEN data, compared to the actual log BFEN density (grey).	99
4.17	Raw BFEN data classified using a skew Normal HMM (left) and log-transformed BFEN data classified using a Normal HMM (right) into low (blue), medium (gold) and high (red) levels of activity.	101
4.18	Boxplots of the normalised one-step-ahead forecast MSEs for the skew Normal HMM fitted to the raw BFEN data and the Normal HMM fitted to the log-transformed BFEN data.	101
4.19	Accelerometer data summarised using the complete step count algorithm, aggregated over one second intervals (top – thinned by 20 for plotting) and aggregated over five minute intervals (bottom).	102
4.20	Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to step count data, compared to the actual step count density (grey).	103
4.21	Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to sqrt step count data, compared to the actual sqrt step count density (grey).	104
4.22	Step count data classified using a skew Normal HMM (left) and square-rooted step count data classified using a Normal HMM (right) into low (blue), medium (gold) and high (red) levels of activity.	105
4.23	Normal HMM posterior empirical within-state densities (solid) and skew Normal empirical within-state densities (dashed) for low (blue) to high (red) activity, through two intermediate activity zones (magenta and gold) fitted to the square-rooted step count data.	105
4.24	Step count data classified using a skew Normal HMM into four activity zones, from low (blue) to high (red), through two intermediate activity zones (magenta and gold) based on the square-rooted data.	107
4.25	Boxplots of the normalised one-step-ahead forecast MSE for the skew Normal HMM and Normal HMM fitted to the raw step count (SC) data and square-rooted step count data, with three and four hidden activity states.	107
4.26	Boxplots comparing the mean MSEs of the one-step-ahead forecasts for the models fitted to each of the activity summaries. The largest outlier for the ENMO data set (11.46) has been omitted from the plot to make the comparison easier.	109

4.27	Transition matrix element-wise posterior densities for the models fitted to each of the activity summaries. The position of each plot corresponds to the element of the matrix it relates to, i.e. the first row of plots corresponds to the first row of the transition matrix.	110
4.28	Ternary plot comparing the overall proportion of time spent in each activity zone for each activity summary.	111
4.29	The five minute BFEN (top) and ENMO (bottom) summaries classified using Model 4.3 and Model 4.4, respectively, using ND 1's accelerometer data. Activity zones are annotated as vertical lines, with width equal to the duration of the activity.	113
4.30	Log-transformed BFEN densities for each person: Patient 3, blue, solid; Patient 5, orange, solid; Patient 10, red, solid; ND 1, blue, dashed; ND 2, orange, dashed; and ND 3, red, dashed.	114
4.31	BFEN within-state posterior densities for each hidden state (one to three, left to right, respectively) for each person: Patient 3, blue, solid; Patient 5, orange, solid; Patient 10, red, solid; ND 1, blue, dashed; ND 2, orange, dashed; and ND 3, red, dashed.	115
5.1	AR coefficient posterior means and 95% posterior interval for orders $p = 1-12$.	122
5.2	ACF and PACF plots for Patient 3's glucose data.	123
5.3	Trace plots for σ^2 and μ for AR models of order $p = 1-12$. Plots are thinned by five.	123
5.4	Two-hour-ahead forecasts (solid) for AR models with $p = 1-12$ fitted to Patient 3's glucose levels, with 95% credible intervals (shaded, dashed). . .	124
5.5	Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for AR models with $p = 1-12$	125
5.6	Trace plots, ACF plots and histograms of the posterior draws of the AR(7) coefficients.	130
5.7	Model 5.2 AR coefficient posterior means and 95% posterior interval for orders $p = 1-10$	131
5.8	Trace plots for σ^2 and ψ_y^{-1} for AR DLMS of order $p = 1-10$. Plots are thinned by five.	131
5.9	Two-hour-ahead forecasts (solid) for AR DLMS with $p = 1-10$ fitted to Patient 3's glucose levels, with 95% credible intervals (dashed).	132
5.10	Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for AR DLMS with $p = 1-10$	133
5.11	Roots of the characteristic equation for the AR models of order 1-10 with coefficients specified by their posterior means.	135

5.12	The AR(4) DLM for Patient 3’s glucose levels decomposed into the smoothed local level component and the series corresponding to the complex pairs of roots for the first four days of data.	136
5.13	Trace plots, ACF plots and histograms of the posterior draws for the diagonal elements of W for the local Fourier DLM.	138
5.14	Two-hour-ahead forecasts (magenta) for the local Fourier DLM fitted to Patient 3’s glucose levels (blue), with 95% credible intervals (gold).	139
5.15	Trace plots, ACF plots and histograms of the posterior draws for the diagonal elements of W for the constrained variance local Fourier DLM.	142
5.16	Two-hour-ahead forecasts (magenta) for the constrained variance local Fourier DLM fitted to Patient 3’s glucose levels (blue), with 95% credible intervals (gold).	143
5.17	Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for Model 5.4.	143
5.18	The first four days of raw glucose data (top) and the corresponding smoothed states. The second row corresponds to the local level component, θ_1 , and the following rows are one of each of the Fourier harmonic components, θ_3 , θ_5 , θ_7 and θ_9 , generated by the constrained variance local Fourier DLM. . .	144
5.19	Posterior trace, ACF and density plot of ϕ for Model 5.5 (a).	148
5.20	Posterior density plots of V and the non-zero diagonal elements of W for Model 5.5 (a) (blue) and Model 5.4 (magenta).	148
5.21	Model 5.5 (b) AR coefficient posterior means and 95% posterior interval for orders $p = 1-9$	149
5.22	Model 5.5 (b) AR component roots for the mean posterior coefficients for orders $p = 1-9$	150
5.23	Trace plots for ψ_y^{-1} , σ^2 and $\psi_{1:4}^{-1}$ for AR models of order $p = 1-9$. Plots are thinned by five.	151
5.24	Two-hour-ahead forecasts (solid) for AR(p) Fourier DLMs with $p = 1-6$ fitted to Patient 3’s glucose levels, with 95% credible intervals (dashed). . .	152
5.25	Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for AR(p) Fourier DLMs with $p = 1-6$	152
5.26	The AR(6) Fourier DLM for the first four days of Patient 3’s glucose levels decomposed into the smoothed local level component, the smoothed harmonics and the series corresponding to the complex pairs of roots.	153
5.27	Two-hour-ahead forecasts for glucose models fitted to Patient 3’s glucose levels, with 95% credible intervals.	154

5.28	Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast from the glucose models presented.	155
5.29	Posterior densities for ψ_y^{-1} and $\psi_{1:5}^{-1}$ for Model 5.4 fitted to Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3's glucose levels.	156
5.30	Two-hour-ahead forecasts (magenta) from Model 5.4 fitted to Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3's glucose levels, with 95% credible intervals (gold).	157
6.1	The lagged correlations between glucose and each BFEN posterior modal activity state (zone one in blue, zone two in gold and zone three in red) for Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3.	159
6.2	The lagged correlations between glucose and each BFEN posterior modal activity state (zone one in blue, zone two in gold and zone three in red) for ND 2's data recorded whilst fasting, with 95% confidence intervals.	160
6.3	DAG showing the structure of a DLM for glucose levels, y_t^g , with hidden states θ_t , with a regression on the hidden activity states, Z_t , that describe the activity observations, y_t^a .	162
6.4	Step function of Model 6.1 for Patient 5 according to the posterior means of each β_j .	166
6.5	Densities of V and the diagonal elements of W for Model 6.1 (blue) compared to Model 5.4 (magenta).	166
6.6	Two-hour-ahead forecasts of Patient 5's glucose levels after fitting Model 6.1 (magenta) compared to Model 5.4 (orange), with 95% credible intervals (purple and green, respectively).	167
6.7	Lagged effect of activity in zone three on glucose levels for Patient 5, according to posterior means.	171
6.8	Two-hour-ahead forecasts of Patient 5's glucose levels after fitting Model 6.2 (magenta) compared to Model 5.4 (orange), with 95% credible intervals (purple and green, respectively).	171
6.9	Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for the step function DLM (magenta) and the scaled Gamma curve DLM (blue).	172
6.10	Posterior densities for m , c and γ in Model 6.2 for Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3.	173
6.11	The mean weighted curve in Model 6.2 for Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3, according to posterior draws of m , c and γ .	174

6.12	The mean cumulative effect of activity (orange) on glucose levels compared to the raw activity data (blue) when fitting Model 6.2 to Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3's data. The left-hand-side y-axis of each plot is the raw activity of each patient and the right-hand-side is the cumulative activity effect for each patient.	175
6.13	Two-hour-ahead forecasts of Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3's glucose levels after fitting Model 6.1 (magenta) compared to Model 5.4 (orange), with 95% credible intervals (purple and green, respectively).	176
6.14	The step function coefficients in Model 6.1 for Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3 according to the posterior means of each β	177
6.15	Lagged mean correlation between glucose levels and number of grams of carbohydrates eaten by ND 1 (blue) and ND 3 (red), with 95% confidence intervals.	178
6.16	Lagged effect of the number of grams of carbohydrates on glucose levels for ND 1 (blue) and ND 3 (red) according to the posterior means of Model 6.2 with observation Eq. (6.3).	180
6.17	Cumulative effect of eating carbohydrates over time (orange) according to Model 6.2 with observation Eq. (6.3) on ND 1 and ND 3's glucose levels, compared to the number of grams of carbohydrates consumed (blue).	181
6.18	Two-hour-ahead forecasts (magenta) on ND 1 and ND 3's glucose levels after fitting Model 6.2 with observation Eq. (6.3), with 95% credible intervals (orange).	182
7.1	Production levels of LNG over time (thinned by five for plotting).	185
7.2	An example of production levels of LNG over five days.	186
7.3	Pairs plots of production quantity against weather-related tags (thinned by 50 for plotting).	187
7.4	Radial plot of wind direction vs production quantities, coloured by wind speed.	188
7.5	The first two days of production data (top) with the corresponding smoothed states. The second row is the smoothed local level component S_1 , then S_2 and S_3 are the smoothed Fourier components θ_3 and θ_5	190
7.6	One-day-ahead forecasts (magenta) for production levels with 95% prediction intervals (orange) after fitting a local Fourier model.	191
7.7	Adjusted production levels of LNG over time after regressing on temperature (thinned by five for plotting).	192

7.8	Adjusted production levels of LNG over five days after regressing on temperature.	193
7.9	Mean proportion of observations within ± 100 Tonnes for forecasts with predictive horizons between one and 24 hours ahead from Model 7.1.	194
7.10	One-day-ahead forecasts (magenta) for production levels with 95% prediction intervals (orange) after fitting Model 7.1.	195
7.11	Proportion of forecasts from Model 7.1 within ± 100 Tonnes for a range of temperature errors.	196

List of Tables

5.1	Model information for AR(p) MCMC.	122
5.2	Model information for AR(p) DLM MCMC.	129
6.1	Model 6.2 MCMC information for Patient 3, Patient 10, ND 1, ND 2 and ND 3.	173

Chapter 1

Introduction

1.1 Introduction and motivation for thesis

Diabetes is a set of serious, lifelong metabolic conditions that cause a person's blood glucose levels to become too high (hyperglycaemia). The disease is usually broken down into two types: type I diabetes and type II diabetes, both of which are products of the body's misproduction of the hormone insulin. Insulin regulates the body's use and storage of glucose and fat; it controls blood glucose levels by signalling the body's cells to take in glucose from the blood, which can then be used for energy. If the body has sufficient energy, insulin signals the liver to store glucose instead. In type I diabetes the pancreas does not produce insulin due to the immune system attacking the cells that secrete it. As a result, type I diabetic patients must administer insulin into the body, though this can result in blood glucose levels becoming too low (hypoglycaemia). The symptoms of hypoglycaemia range from sweating, fatigue and dizziness, to sufferers becoming comatose in extreme cases. Type II diabetes stems from the pancreas not producing enough insulin or the body's cells not reacting to insulin and is often associated with obesity. Signs of hyperglycaemia include an increase in urination, thirst and hunger and regular hyperglycaemia over long periods of time can lead to longer term problems, such as organ damage that can lead to heart failure, loss of vision and amputation. Due to the potential life-threatening effects of diabetes, blood glucose levels must be closely monitored. Recently, Ahlqvist *et al.* (2018) suggested that there are actually five distinct types of diabetes, separated by the level of risk they present and by different patient characteristics. In future this could lead to much earlier targeting of precise treatments for patients who are at higher risk of diabetic complications.¹²

The Internet of Things (IoT) refers to the quickly expanding network of connected devices and sensors collecting large volumes of data. Amongst these sensors are continuous

¹<https://www.diabetes.co.uk/>

²<https://www.nhs.uk/conditions/diabetes/>

glucose monitoring (CGM) devices, microchips that are placed just beneath the skin to measure a patient's interstitial glucose levels, which are equivalent to blood glucose levels with a time lag of around 15 minutes. As a result of the lag in measurement accuracy, CGM devices must be calibrated with finger prick measurements, but this is much less often than a patient would usually have to take finger prick measurements if they did not have the CGM system. CGM allows patients to see a full blood glucose profile throughout the day and night, as regular blood glucose observations are sent from the monitor to a hand held device or pump. This gives patients the opportunity to check their blood glucose levels in real-time and learn particular trends in their glucose profile so they can be more proactive in managing their condition.³

Since the introduction of CGM devices, many studies have been conducted showing how their use has improved how well patients manage their condition. For example, Bode *et al.* (1999) showed a significant reduction in HbA1c (glycated haemoglobin) levels in patients who wore a CGM device for two one-week periods across five weeks and were able to adjust their treatment accordingly after observing their glucose profiles both times.

The artificial pancreas is a closed-loop modelling system developed for type I diabetes patients that aims to keep blood glucose levels in the euglycaemic range (range of normal glucose levels), reducing the risk of hypoglycaemia. A CGM device is used to monitor the patient's blood glucose levels, which are then transmitted to a device that calculates the amount of insulin required to normalise the blood glucose levels. Insulin is then pumped into the patient's system, without the need for human interaction. The advances in this system are particularly important for type I diabetics overnight. Patients experience an increased risk of a hypoglycaemic episode when sleeping as the counter-regulatory reaction to hypoglycaemia is blunted.⁴

So far, work using CGM devices has mainly focused on type I diabetic patients, but there is a large amount of scope for using these devices for type II diabetics, who make up 90% of those diagnosed with diabetes. Vigersky & Shrivastav (2017) provide an overview of several studies in which type II diabetic patients have been fitted with CGM devices. This includes Garg *et al.* (2006), who conclude that those fitted with a CGM device reduce their time spent in hypoglycaemia by 23%, nocturnal hypoglycaemia by 38% and hyperglycaemia by 21%, showing a vast improvement to overall glucose levels compared to those without a real-time CGM device. Time spent in the euglycaemic range was increased by 26% overall. Vigersky & Shrivastav (2017) highlight how forecasting blood glucose levels ahead of time would further increase these percentages of improvement.

³<https://www.diabetes.org.uk/guide-to-diabetes/managing-your-diabetes/testing/continuous-glucose-monitoring-cgm>

⁴<http://paediatrics.medschl.cam.ac.uk/research/the-artificial-pancreas-project/>

1.2 Glucose modelling

As demonstrated through work on the artificial pancreas, knowledge of a patient's current blood glucose levels can help in making real-time decisions. In the case of the artificial pancreas, a decision can automatically be made on how much insulin to issue a type I diabetic patient to correct blood glucose levels. Even more value can be added to situations like this if blood glucose levels could be predicted ahead of time, before the scenario of hypo- or hyperglycaemia has arisen. Fortunately, statistical inference provides the predictive framework to do this.

Bremer & Gough (1999) posed the question of whether blood glucose levels could be predicted from their previous values. They suggest that in order to effectively predict future blood glucose levels from previous values, the sampling rate of the blood glucose measurements must be frequent enough to capture detailed excursions: in particular this occurs when there are external factors that directly affect blood glucose levels, for example meals and exercise, and samples must be taken every two to five minutes. The data should be collected for as long as possible, preferably several days' worth, and there should be a clear distinction between data that are collected during periods with external influences and when there are none, for example when a patient is sleeping or fasting. Clearly, after allowing for any periodic (seasonal) behaviour and the effects of external influences, it is helpful if the differences or residuals of the remaining stochastic process can be treated as stationary.

1.2.1 ARIMA models

The most prominent CGM models appearing throughout current literature are autoregressive integrated moving average (ARIMA) models. The background on ARIMA models is discussed in Section 3.1.4. Sparacino *et al.* (2007) consider a first order autoregressive (AR(1)) model and a polynomial model and compare their predictive accuracy for blood glucose levels 30 minutes and 45 minutes ahead. Data are collected from 28 type I diabetic patients, each fitted with a CGM device, every three minutes. There is clear seasonality in blood glucose profiles due to the regular pattern of meal times and sleep. To allow for this form of non-stationarity, the parameter estimates for both models are updated at each time step that a new observation is introduced. These estimates are produced using weighted least squares: the data are weighted by a 'forgetting factor', μ^k (where $0 < \mu < 1$ and k is the lag), that determines how far back in time past blood glucose levels affect the current and future blood glucose levels and therefore the current parameters of each model. Higher values for μ achieve a smoother prediction profile, with the cost of a longer delay in detecting changes, whilst lower values of μ are more sensitive to noise. Since a noisier prediction profile leads to the bigger issue of increased false detection of hypo- and

hyperglycaemic episodes, $\mu = 0.8$ was chosen for calculating the model parameters. It was found that having a higher signal-to-noise ratio (SNR) resulted in reliable predictions for longer predictive horizons, however the 30 minute prediction horizon allowed enough time for detecting a hypoglycaemic episode and for sugar to have an effect.

Sparacino *et al.* (2007) found that for data sampled every three minutes, higher order AR models often yielded unstable predictions (predictions with high sensitivity to noise) and a first order model was sufficient in providing clinically significant predictive performance (predicted a hypo- or hyperglycaemic event far enough in advance for an intervention to be made). It is, however, suggested that higher orders of AR models are necessary for CGM data with higher sampling rates. Gani *et al.* (2009) chose an AR(30) model as the optimal model for predicting blood glucose levels 30 to 60 minutes ahead, using CGM data collected every minute. This order was chosen from a selection of optimal values, ranging from 25 to 35, for data from nine type I diabetic patients. Since the models with orders as high as this showed little difference in predictive performance, order 30 was chosen. The model was tested in three different scenarios, varying on how the model parameters were estimated and whether or not the data had been smoothed. Scenarios one and two used ordinary least squares estimates for the model parameters, with the first using the raw CGM data and the second using smoothed CGM data. The parameters for both models yielded coefficients that correspond to unphysiological behaviour, where their values did not gradually decrease over time, but the second model provided very accurate predictions for blood glucose levels 30 minutes ahead (root mean square error (RMSE) = 0.02 mmol/L), though this model was unstable when white noise was added to the raw signal. Scenario three used smoothed CGM data and regularised least squares estimates of the model parameters. This model was stable and had viable AR coefficients. Predictions were accurate (30-minutes-ahead RMSE = 0.1 mmol/L, 60-minutes-ahead RMSE = 0.8 mmol/L, 90-minutes-ahead RMSE = 1.6 mmol/L), with relatively short predictive lags – the time delays in correctly predicting glucose levels (30-minutes-ahead lag = zero minutes, 60-minutes-ahead lag = 12 minutes, 90-minutes-ahead lag = 38 minutes).

Reifman *et al.* (2007) propose an AR(10) model using CGM data sampled every minute from nine type I diabetic patients. The order of the AR model is determined through the Akaike information criterion (AIC) and, as for scenario three in Gani *et al.* (2009), uses regularised least squares to determine the parameters of the model. The model is tested, for each patient, for predictive horizons of 30-, 60- and 120 minutes ahead. As seen in Gani *et al.* (2009), the lag in predictions increased with the number of time steps ahead (figures not reported), as did the RMSE (1.23 mmol/L, 1.94 mmol/L and 2.99 mmol/L, respectively). The lower order of the model failed to capture some of the correlations in the blood glucose measurements, shown by a slight sinusoidal pattern in the autocorrelation function (ACF) plot. This may explain the increased RMSE in the predictions compared

to Gani *et al.* (2009). Another aspect explored by Reifman *et al.* (2007) is how well each AR model trained on individual patients does at predicting the other patients' blood glucose levels. In all cases, the model performance is only slightly reduced, showing that the autocorrelation variability across patients is very small.

Montaser *et al.* (2017) study whether using a seasonal ARIMA (SARIMA) model as opposed to a non-seasonal version significantly improves predictive accuracy. Data from 10 type I diabetic patients collected every 15 minutes over a period of eight hours, twice per patient, are concatenated to form one data set. The SARIMA model outperformed the ARIMA model for all predictive horizons (30 to 300 minutes ahead). A seasonal model with the exogenous variable insulin infusion was also considered and compared. This added information in the model again improved predictions, improving the mean absolute percentage error (MAPE) calculated from the SARIMA model by 61%. The data in this study, unlike that in Reifman *et al.* (2007), featured high inter-individual variation.

1.2.2 Other models

Aside from ARIMA models, a number of alternative approaches to modelling blood glucose data have been suggested.

Miller & Strange (2007) propose applying a discrete Fourier transform to the 24 hour CGM data of diabetic patients. The model is tested on a type II diabetic patient's glucose profile using five harmonics (11 parameters) and using 20 harmonics (41 parameters) to approximate the seasonality. Plotting the errors of both of these models shows that using five harmonics results in regular oscillations not being captured, whereas using 20 harmonics results in errors that resemble white noise. More complex models, however, can often lead to higher predictive variance and this was not assessed.

Kafali *et al.* (2014) build upon the existing personal health system COMMODITY⁵⁶ by proposing a hybrid diagnosis and monitoring architecture for diabetes (HYDRA). COMMODITY receives blood glucose level readings to a monitoring agent and sends an alert if the observed level requires action (a reading < 3 mmol/L signals critical hypoglycaemia: the condition being monitored). This means that a reading that only just falls within the acceptable region would not trigger an alert, which could mean that dangerous blood glucose levels could occur before the next observation. HYDRA improves this system by forecasting the next blood glucose level ahead if the current observation is within the acceptable hypoglycaemic range ($3 \leq \text{glucose} < 4$), but therefore could become critical. A general model is learned from a database of multiple patient data sets and is customised per patient as new data from that patient are received. The model is also fine-tuned based on 'hidden' patient-specific information, such as excessive alcohol use or shift work,

⁵<http://www.bodytel.com>

⁶<http://www.medtronicdiabetes.com/products/guardiancg>

incorporated as indicator variables that could improve predictive performance. The capability of the model is tested on synthetic data, generated from AR models, in a simple case study. When a value in the hypoglycaemic range is detected, a 30 minute forecast is made. The forecast accurately predicts the oncoming critical blood glucose level that occurs 15 minutes later.

1.2.3 Assessment of performance of models

It is important when fitting time series models to be able to assess how well they perform. Quantifying the error a model possesses is a crucial step in model development to decide whether or not the model is reliable enough to use. This section looks at the various features of models considered in the literature that influence how well they describe blood glucose profiles, such as numerical accuracy, time delay in predicting hypo- or hyperglycaemia and clinical point accuracy. The model-fits in the literature are frequentist, so the following are measures for point forecasts. The work in this thesis will take a Bayesian approach, allowing the predictive variance, i.e. the uncertainty in the point forecasts, to be easily additionally captured.

Numerical point accuracy

There are an extensive number of statistics that summarise the numerical forecasting accuracy of a model against the observed time series. These summaries are a straightforward way to compare the performance of different models, or to test whether a model meets a set of criteria.

The correlation coefficient between estimated blood glucose levels and observed blood glucose levels can be calculated, demonstrated by Leal *et al.* (2010). This method does not quantify the error in the model directly, but a strong positive value indicates that the model estimates are close to the observed glucose levels and the model performs well.

Petris *et al.* (2009) list the following as common measures of forecasting accuracy to compare time series models: the mean absolute deviation (MAD), the mean square error (MSE) and the MAPE. Frequently seen in the literature is the RMSE (Gani *et al.*, 2009; Reifman *et al.*, 2007), the square root of the MSE. The MSE is also known as the mean square predictive error (MSPE) when the error is the difference between a predicted signal and the observed signal, which is used by Sparacino *et al.* (2007). Montaser *et al.* (2017) use the MAPE to compare the performance of seasonal ARIMA models with non-seasonal ARIMA models.

Various governing bodies set criteria for the blood glucose levels reported by self-monitoring blood glucose (SMBG) systems in terms of the amount of error they can incur compared to reference blood glucose levels. The American Diabetes Association set the

goal that the total error (system plus user error) for SMBG systems should be less than 10% (Parkes *et al.*, 2000). The acceptable levels of error set out for SMBG and CGM devices should be considered when investigating the performance of blood glucose models to decide acceptable levels of error when forecasting ahead of time.

Additional to prediction point accuracy, Sparacino *et al.* (2007) assessed the performance of the polynomial and AR(1) models by quantifying the lag in predicting ‘trigger’ blood glucose levels (the time difference between the forecasted values that fall in the hypo- or hyperglycaemic ranges and their observations). Rather than quantifying the delay between the peaks of the glucose profile and the peaks of the forecast (and conversely between the troughs), where the delay is largest, the time delay was measured on the upward and downward trends of the original and predicted profiles. The point of measurement comparison was taken to be 75% of each peak-to-trough distance in each of the 28 time series, instead of where the original time series and where the predicted time series cross the threshold to trigger a critical glucose level. Quantifying how much lag a model forecast incurs is an important statistic to look at in blood glucose monitoring. Ideally, forecasts must capture a change in blood glucose levels with enough time for an intervention to take place and have a positive effect in order to prevent hypo- or hyperglycaemia.

Clinical point accuracy

Whilst having a model that accurately estimates or predicts blood glucose levels is useful, from a medical point of view what is more important is having a model that is clinically accurate. Setting a standard for all estimated blood glucose levels to be within a certain percentage of realised blood glucose levels is not always the most appropriate way to assess accuracy, since measurement error percentages have varied clinical significance for different blood glucose levels. Clinical accuracy consists of correctly estimating instances of hypoglycaemia, euglycaemia and hyperglycaemia so that the right corrective treatments can be applied. As an alternative to comparing how close estimated glucose levels are to actual blood glucose levels, the Clarke error-grid analysis (EGA) (Clarke *et al.*, 1987) was developed as a way of assessing how clinically accurate SMBG systems are. The analysis looks at the relative difference between the absolute value of the system-generated glucose level and the absolute value of the reference blood glucose level and classifies the clinical significance between the two observations with a letter from A to E. Observations classed as A correctly estimate the state (hypo-, hyper- or euglycaemia) of the blood glucose levels, whilst observations marked as E have resulted in the opposite intervention being taken (e.g. estimating a hypoglycaemic episode when the actual blood glucose levels indicate a hyperglycaemic episode). Leal *et al.* (2010) use Clarke EGA to assess the clinical accuracy of blood glucose levels from a CGM device. An AR model is proposed as a real-time

alternative to the CGM blood glucose estimates, with the Clarke EGA showing the much improved performance of this model at estimating and classifying current blood glucose levels.

The Clarke EGA has since been used to assess the clinical accuracy of models predicting blood glucose levels ahead of time by comparing them to observed blood glucose levels from CGM devices. Reifman *et al.* (2007) use Clarke EGA to compare the performance of an AR(10) model for three predictive horizons: 30-, 60- and 120-minute-ahead blood glucose levels. The portability of the AR(10) model is also assessed using Clarke EGA. This looks at how effective the model is at predicting one patient's blood glucose levels when trained using a different patient's blood glucose data.

A common extension of Clarke EGA is to use the continuous glucose error-grid analysis (CG-EGA). CG-EGA consists of Clarke EGA and rate error-grid analysis (R-EGA), which assesses how blood glucose levels change over time. Similar to the Clarke EGA, the R-EGA grid is divided into zones A to E, with an observation in zone A representing near perfect agreement between the rates of change of the reference blood glucose levels and the sensor. Zanderigo *et al.* (2007) use CG-EGA to compare the performances of the two models proposed by Sparacino *et al.* (2007) (a polynomial model and an AR(1) model). Using CG-EGA to assess the performance of the models indicates that the two perform very similarly, with the AR model only slightly outperforming the polynomial model in terms of the slightly higher number of points classed into zones A and B (clinically accurate in terms of intervention taken). To carry out R-EGA, data must be sampled at a high enough frequency to capture the change in blood glucose over time. This was possible for Zanderigo *et al.* (2007) as the blood glucose data used were sampled every three minutes, however Leal *et al.* (2010) state that the data sampled every 15 minutes used in their study were too infrequent to apply R-EGA. This coincides with the requirement Bremer & Gough (1999) stated for blood glucose levels to be sampled every two to five minutes in order to capture enough detail in blood glucose profiles: a higher sampling rate also allows for a more in depth analysis of a model's performance.

Consensus EGA was developed by Parkes *et al.* (2000) using the clinical judgement of 100 diabetes experts to update the Clarke EGA. The consensus EGA sets slightly altered definitions for the error zones A to E and has different boundaries based on whether a patient has type I or type II diabetes. The International Organization for Standardization stipulates that at least 99% of the blood glucose levels reported by SMBG systems must fall within zones A and B of the consensus error-grid (Freckmann *et al.*, 2015).

1.3 Incorporating activity data

1.3.1 Background

More recently it has become apparent that the effects of type II diabetes can be reduced, if not reversed, through strict control of patient diet and exercise (Nelson *et al.*, 2002). Since these factors have a highly influential effect on blood glucose levels, collecting and modelling these data alongside CGM measurements can provide a more informed inference for blood glucose forecasts. Part of the appeal of CGM devices is the reduced need for human interaction with the device to get blood glucose level readings. Recording meal times and detailed macronutrient information would increase this interaction again, though exercise is a factor that can be more automatically monitored, for example, through wearable accelerometers.

Conn *et al.* (2007) produced a meta-analysis of studies testing the effects of exercise interventions on metabolic control in type II diabetes patients. The analysis included studies that target exercise alone, as well as studies that target multiple self-management behaviours at the same time. Of the 103 reports analysed, all treatment groups that increased the amount of exercise performed by participants showed strongly significant improvement in metabolic control compared to control group members, who showed no significant improvement. Overall, studies that focussed on exercise intervention alone had twice as large an improvement in metabolic control after the intervention as studies that focussed on multiple self-management interventions. The difficulty in changing multiple behaviours at once is a notable factor in why the results of targeting exercise alone is more effective than targeting several, but the results also imply that exercise is a major component in type II diabetes management and an effective way of controlling the disease.

Tri-axial accelerometer data can be harvested from IoT activity watches and transformed to quantify how much activity has been performed. Various ways of representing these data will be discussed in Section 1.3.3. Fitting a joint statistical model to previous blood glucose levels and activity levels will allow for more accurate predictions of future blood glucose levels, increasing the chances of predicting abnormal blood glucose levels ahead of time. It also allows forecasts of glucose to be made under different exercise regimes to help patients learn the benefit of exercise interventions.

1.3.2 Physiology

Several studies have been conducted to test how different types of activity affect blood glucose levels. Postprandial (the period of time after a meal) blood glucose levels increase before the body produces enough insulin to bring them back down again. This spike is more prominent in people with type II diabetes, often resulting in hyperglycaemia.

Henson *et al.* (2016) proposed breaking up time spent in a sedentary position with short bursts of standing or walking to improve postprandial metabolic responses. The study was carried out on 22 postmenopausal women at high risk of developing type II diabetes. Subjects were assigned to two of three groups: prolonged, unbroken sitting (7.5 hours); prolonged sitting broken up with standing for five minutes every 30 minutes; and prolonged sitting broken up with walking for five minutes every 30 minutes. Breaking up sitting with standing was found to reduce the postprandial rise in glucose by 34%, whilst breaking up sitting with light-intensity walking reduced this by 28%. There were also positive effects of the two activities on insulin levels, reducing the postprandial amount by 20% and 37%, respectively. What is more, these positive effects carried on into the next day. This follows on from the evidence in Dunstan *et al.* (2012), where breaking up sedentary periods with two minutes of light- or moderate-intensity walking every 20 minutes also significantly reduced postprandial glucose and insulin levels.

As an alternative to breaking up sedentary time with walking, McCarthy *et al.* (2017) looked at whether performing seated upper body activity at the same level of energy expenditure as light-intensity walking would have the same effect on postprandial glucose levels. Blood glucose measurements of 13 inactive, obese adults at risk of type II diabetes were taken 30-, 60-, 120- and 180 minutes after breakfast and lunch. Results showed that the incremental area under the curve (iAUC) for glucose and insulin levels when breaking up sitting with arm ergometry every 30 minutes was lower than for breaking up sitting with light-intensity walking. Whilst this may be due to the perceived higher intensity of the arm ergometry, the results show that a postural change is not necessarily the only way to alleviate postprandial blood glucose levels.

Though it is clear that breaking up sedentary time with standing or walking has a significant, positive effect on postprandial blood glucose levels, McCarthy *et al.* (2017) assess how much cardiorespiratory fitness (CRF) levels contribute to this reduction. The study involved 34 healthy (non-obese) patients, evenly spread across three fitness levels: zero minutes of moderate to vigorous physical activity (MVPA) per week; 75 to 150 minutes of MVPA per week; and > 150 minutes of MVPA per week. The results show that those with a CRF level in the 25th percentile reduce their glucose levels by almost 50% when taking regular walking breaks, with initially high blood glucose levels prior to walking. Those with a CRF in the 75th percentile, however, only reduce their glucose levels by around 11% when adding in walking breaks, though they had much lower initial blood glucose levels on average. In summary, having a high CRF level or taking regular walking breaks reduces the postprandial blood glucose levels of those in prolonged sedentary positions.

Whilst the benefits of exercising to improve the effects of diabetes are widely known, Allen *et al.* (2008) state how 60% of those living with type II diabetes still do not engage in physical activity. This study therefore instead looks at the effects of CGM counselling

on 52 adults with type II diabetes. All patients received 90 minutes of diabetes education prior to using the CGM devices and the intervention group (27 patients) received additional counselling on self-efficacy. Self-efficacy was a main feature in the study, as it has a large impact on whether an individual will start/continue doing something. After eight weeks, the intervention group had significantly higher self-efficacy levels to continue exercising, significantly decreased minutes spent in light-intensity and sedentary activity with an increase in minutes spent in moderate activity and a significant decrease in HbA1c and body-mass index (BMI). These results show that undertaking counselling to improve self-efficacy plays a large part in increasing physical activity levels and, as a result, reducing blood glucose levels. It also supports the other evidence to show how exercise reduces blood glucose levels, particularly in those with type II diabetes.

1.3.3 Pre-processing

Extracting meaningful information from raw tri-axial accelerometer data is a challenging task. As well as the movement being performed, signals are made up of noise and gravity, which are picked up as vibrations by the sensor to measure acceleration. When the accelerometer is completely at rest the total magnitude of acceleration is equal to one, which corresponds to 1 g of purely gravitational acceleration. The data are often very big due to the high frequencies at which they are collected. Additionally, this makes plotting the data or summarising them computationally intensive. In the case of wanting to quantify how much activity a person has performed, various transformations can be applied to their accelerometer data.

Filtering raw tri-axial accelerometer data is a popular way of removing noise from the signals before transforming them into a quantitative activity level. The type of filter applied to data depends on the frequencies of noise that are to be removed; a high-pass filter will attenuate (gradually lose) signals with a frequency below a specified cut-off point (the stopband) and retain the frequencies above the cut-off (the passband), with the level of attenuation specified by the filter itself. A low-pass filter retains frequencies below a given cut-off point and a band-pass filter combines the low- and high-pass filters by only retaining frequencies of a signal that are within a given range (known as a bandwidth). An example of the use of band-pass filters is in wireless communication systems, where signals are sent via a given band of frequencies so as to not interfere with signals sent in other bandwidths (Shenoi, 2005).⁷⁸⁹

The ideal filter has a flat frequency response, meaning the frequencies within the passband are represented uniformly. The Butterworth filter (Butterworth, 1930) has a

⁷https://en.wikipedia.org/wiki/High-pass_filter

⁸https://en.wikipedia.org/wiki/Low-pass_filter

⁹https://en.wikipedia.org/wiki/Band-pass_filter

maximally flat frequency response in the passband and goes to zero in the stopband, which makes it a desirable and therefore commonly used filter.

Applying the Euclidean Norm (EN) to tri-axial data is a popular way of converting three signals into one. Van Hees *et al.* (2013) suggest five metrics for separating the gravitational component from the movement using tri-axial accelerometer data from a robot performing ‘arm movements’, each containing the EN combined with different levels of filtering. These metrics are also then applied to human accelerometer data to assess daily physical activity, where the daily energy expenditure of each participant is known. The five methods for separating the gravitational component out from the signal are: the EN; EN minus one (ENMO); EN of the band-pass filtered signals (BFEN); EN of the high-pass filtered signals (HFEN); and HFEN plus the EN of low-pass filtered signals minus one (HFEN₊). BFEN and EN were only applied to the human participants’ data sets: EN to assess the relevance of removing gravity from the signal and BFEN to assess the effects of high-frequency noise removal. ENMO consists of calculating the EN at each time point, then subtracting 1 g from the resulting signal. Typically, negative values are then truncated to zero. HFEN is the application of a high-pass fourth order Butterworth filter with a cut-off frequency of 0.2 Hz, followed by calculating the EN. BFEN is the band-pass version of HFEN: using a fourth order Butterworth filter with a cut-off frequency of 0.2 to 15 Hz. HFEN₊ is where HFEN is applied to the signals followed by adding on the EN of the low-pass fourth order Butterworth filtered raw signals (LFEN), also with a cut-off frequency of 0.2 Hz (it is assumed that all human activity is performed above this frequency). When there is no rotational movement, LFEN is equal to 1 g, hence the minus one in ENMO, but when rotational movement is present, the value can be lower or higher than 1 g. Adding this alternative value to the HFEN signal is therefore more precise. The study concluded that the metric used has a large impact on explaining the variance in daily physical activity, but that no one metric outperformed the others overall.

Van Hees *et al.* (2014) and da Silva *et al.* (2014) use ENMO to estimate population activity levels. Van Hees *et al.* (2014) propose two autocalibration methods to reduce the amount of error in accelerometer measurements. The impact of the new calibration method on estimating population activity was then observed through using ENMO and BFEN. Da Silva *et al.* (2014) apply ENMO to quantify the amount of activity and length of time spent in MVPA using accelerometer data from 9,000 members of the Pelotas birth cohorts in Brazil.

Doherty *et al.* (2017) trial the use of wrist-worn accelerometers in large scale studies as a measure of how much physical activity people do. This is an alternative to the common ways of measuring engagement in activity, which are often self-reported and therefore make it hard to gauge total levels of activity. This makes setting minimum levels of activity for a population to be healthy more challenging. Data were collected from over 100,000

subjects, each providing seven days of 100 Hz tri-axial data. The data are processed using ENMO, with the inclusion of a low-pass fourth order Butterworth filter with a cut-off frequency of 20 Hz. The study shows that accelerometers can be worn and used on a large scale to collect data of high quality.

Hurd *et al.* (2013) perform a study on different techniques used to analyse 100 Hz tri-axial accelerometer data in evaluating the functionality of the upper extremities. One of the methods involved measuring the amount of time spent in each activity intensity level (inactivity, low activity or high activity). The raw accelerometer signals were processed using a fourth order Butterworth high-pass filter at 0.1 Hz to remove the gravitational component. The EN was then taken at each time point and the data were summed into one minute epochs. Data were collected from 30 patients, 15 of which were control subjects. The activity bins were decided based on the percentage of the control group maximum activity value (CMA): inactivity $< 110 \text{ min/s}^2/\text{epoch}$; low activity $110 \text{ min/s}^2/\text{epoch} \leq \text{activity} \leq 33\% \text{ of CMA}$; and high activity $> 33\% \text{ of CMA}$.

As an alternative to converting accelerometer data into an ‘activity level’, data can instead be converted into a step count. Zhao (2010) describes an algorithm to do this: data are filtered to smooth the signals and the axis with the largest acceleration change is detected to calculate the steps; a dynamic threshold is calculated every 50 samples; when the signal crosses this threshold and the slope is negative, a step is detected. Whilst step-counts are a good way of quantifying activity, algorithms like this one are not as simple to implement as those listed above.

Knowledge of when a patient is asleep can give added insight to the patterns in their blood glucose levels. Van Hees *et al.* (2015) present a sleep detection method using accelerometer data. The tri-axial data were used to estimate the arm angle of each patient. The patient was assumed to be sleeping when there were periods of low frequency of changes in arm angle. If there was no change of more than 5° over at least five minutes, the patient was recorded as inactive. The Pearson’s correlation coefficient between total sleep duration detected and total sleep duration logged by patients was 0.98. Moderate performance of the algorithm in detecting sleep duration was reported.

1.4 Implications for modelling

The literature discussed throughout this chapter supports the idea that there is ample scope and demand for monitoring and forecasting the glucose levels of type II diabetes patients in real-time. The strong, positive impact that exercise has on glucose levels is something that should be taken advantage of, namely by monitoring patient activity levels alongside their glucose levels to provide more informed forecasts or forecasts under different future activity regimes. This section discusses the implications for modelling that

arise from the current research in this area.

The first thing that should be considered when monitoring patient glucose levels is how frequently measurements should be taken. Bremer & Gough (1999) suggest glucose levels should be sampled every two to five minutes to be able to capture excursions in the data. Gani *et al.* (2009) and Reifman *et al.* (2007) fit AR models to data that are sampled at the higher frequency of every one minute, but this results in both modelling approaches requiring a much higher order model to get enough information from past glucose levels for more accurate forecasting. Sparacino *et al.* (2007) use data that are sampled every three minutes and found that higher order models led to unstable predictions, so these should be avoided where possible. Blood glucose measurements for this thesis will be taken every five minutes to account for any important changes in the data, but to reduce the need for an overly complicated model. This is particularly important given the goal of this thesis to generate forecasts in real-time. Also following the advice of Bremer & Gough (1999), the data collection will be for as long as possible: at the time of collecting the initial glucose data sets, CGM technology allowed for accurate data collection for five to seven days.

Models need to account for the periodicity of glucose levels caused by regular meal times, sleeping and regularly performed activities and for the autocorrelation that remains after allowing for these effects. Both can be captured using high order AR models whose (latent) components can represent quasi-periodic behaviours of this nature, as well as high frequency residual noise (Huerta & West, 1999). Such high order AR models were used by Gani *et al.* (2009) and Reifman *et al.* (2007). Gani *et al.* (2009) suggest that knowledge of glucose levels for the previous 30 minutes is sufficient to infer what will happen to future glucose levels in the next 30 to 60 minutes. Montaser *et al.* (2017) found that fitting a seasonal model (SARIMA) improved predictions, whilst Miller & Strange (2007) experimented with fitting different numbers of harmonics in a discrete Fourier transform model to glucose levels. All of the above suggest that capturing the seasonality and autocorrelation in the data is key for being able to successfully forecast far enough ahead that predictions are useful to making decisions for any necessary glucose interventions. In this thesis, seasonal dynamic linear models (DLMs) and high order AR processes will be considered. Note that allowance for time-varying parameters in a DLM is similar in spirit to the forgetting factor estimation method used by Sparacino *et al.* (2007) when fitting their (static) AR and polynomial models; see Section 1.2.1.

The incorporation of additional information about a patient's lifestyle and/or influential factors is shown to improve predictive performance. Bremer & Gough (1999) highlight that the distinction between when there are external factors and when there are not is key for glucose modelling. In both Montaser *et al.* (2017) and Kafali *et al.* (2014), insulin infusion as an exogenous variable and additional lifestyle factors, respectively, are incorporated to infer glucose levels. Montaser *et al.* (2017) report how this greatly im-

proved the predictive performance of their seasonal model. Van Hees *et al.* (2015) provide a methodology into sleep detection in accelerometer data, which could be used to make the distinction between when external factors are present and when they are not. In this thesis, patient activity levels will be used to help infer glucose levels.

There is evidence to suggest that model parameters inferred from the glucose levels of one patient can be used to make inferences about other patients. Both Reifman *et al.* (2007) and Kafali *et al.* (2014) use this method, with model updates applied as new data are received and additional model tuning performed through the addition of lifestyle variables by Kafali *et al.* (2014). Reifman *et al.* (2007) found that cross-patient autocorrelation variability is small and so fitting the model to one patient and using it on another had very little difference in predictive performance. This is something to consider when fitting models to the glucose and activity levels of the patients in this thesis; are the parameters inferred for models fitted to activity data similar across patients, too? A universal model (or informative prior) for glucose and activity that can be used across patients would be beneficial in reducing computation time for real-time inference.

The predictive horizon needed to detect a change in glucose levels that requires intervention with enough notice for an intervention to have an effect must be considered when deciding the level of predictive accuracy models must possess. In type I diabetics, 30 minutes notice is enough time for a patient to take a food-based intervention and for it to have an impact prior to hypoglycaemia occurring. The amount of time it takes for a bout of activity to positively impact glucose levels must therefore be considered, too. This will be informed through exploratory data analysis and judgement from experts in metabolism.

The literature highlights two main ways of categorising model accuracy: numerically and from a clinical perspective. Both of these methods will be considered in this thesis. Quantitatively, models will be expressed as a Bayesian test quantity, their MSE, and later by how accurately they predict a clinically significant event. Additionally, the credible interval on forecasts will be analysed, since this needs to be within a reasonable range for forecasts to be worthwhile and reliable.

Kafali *et al.* (2014) consider what cut-off points should be for critical glucose levels for a behavioural prompt to be sent. The prompt should be sent prior to glucose levels becoming critical for it to be useful and with enough notice. The range of acceptable glucose levels in the literature is understood to be between 4 mmol/L and 7.8 mmol/L, so these values are a good place to start, however this range may need to be tailored to each patient. This is discussed further in Chapter 2. A further consideration on behavioural prompts is how frequently they should be sent such that they still have the intended effect. If prompts are issued too often, they could have the negative effect of being ignored by the patient, whereas if they are issued too infrequently, dangerous excursions in glucose

levels could occur when they could have been prevented. Allen *et al.* (2008) showed that self efficacy has a big impact on how likely people are to carry on exercising without being prompted; how behavioural prompts are issued so that they might incite consistent change should therefore also be considered.

Research surrounding the impact of exercise on glucose levels implies that the type of activity being performed is not important for an effect on glucose levels to be seen. Henson *et al.* (2016) show both standing and walking reduce postprandial glucose levels, with positive effects carrying on into the next day. Dunstan *et al.* (2012) show the same thing with subjects walking at different intensities and McCarthy *et al.* (2017) find that performing isolated upper body activity also reduces glucose levels, implying that a postural change is not necessarily needed. Overall, having a higher initial cardiovascular fitness level reduces glucose levels after eating, too, having the same effect as walking. For the purpose of this thesis, detection of the type of activity being performed is therefore not necessary and quantifying the activity intensity of each patient will be sufficient for aiding the prediction of glucose levels. For issuing behavioural prompts, patients should be encouraged to be active, but a specific activity prompt is not needed.

The method of collecting activity data should require minimal input from the patient. Doherty *et al.* (2017) use accelerometer data to avoid bias from patients, which is more likely to occur when patients are self-reporting how much activity they have done and of what type. This form of data collection involves very little interaction from the patient and data can be continuously recorded (as long as the patient is wearing the accelerometer), so will be used for monitoring activity levels for this thesis. Conn *et al.* (2007) infer that focussing on one glucose intervention at a time (exercise) is more effective on patients than encouraging change in several behaviours at once. Activity is the only intervention that will be monitored alongside glucose for the main part of this study for these reasons, however further data are collected on some additional patients and are discussed in Chapter 6.

The literature highlights several methods for pre-processing accelerometer data into activity data. All of the activity summaries discussed in Section 1.3.3 will be applied to a sample of patient accelerometer data in Chapter 2 and compared to see which methods summarise an activity profile best. This includes determining whether or not data should be filtered before pre-processing and then applying EN and step count algorithms.

Da Silva *et al.* (2014) quantify the amount of activity performed and the length of time spent in different activity zones. The length of time spent in each activity zone will be considered when fitting models to the activity data to classify the level of activity performed. Models should incorporate a dependency structure between observations that reflects realistic movement between activity levels. The amount of activity performed and the length of time spent in each activity zone on average are useful metrics to compare activity expenditure across patients.

The number of perceived activity zones reported in Hurd *et al.* (2013) is three: low-, medium- and high levels of activity. This is the number of zones that will initially be explored, though there will be some investigation into whether more (or fewer) lead to a better fitting model. Hurd *et al.* (2013) calculate the boundaries for these zones based on percentages of the maximum activity levels observed in a control group. The cut-off points in this thesis will be inferred from the data to align with the Bayesian inferential framework presented.

1.5 Objectives of the thesis

In summary of the literature presented and the implications for modelling that have been drawn from it, this thesis presents two main objectives.

1. The primary goal is to develop a joint time series model for glucose and activity levels of type II diabetes patients and computational methods for fitting the models in a Bayesian framework. Also to propose a strategy for using the models for real-time prediction.
2. The secondary goal is to investigate whether other information, such as food intake, could lead to improvements in the model.

1.6 Outline of the thesis

The remaining chapters of this thesis are outlined as follows. Chapter 2 is an exploratory analysis of the glucose and accelerometer data that were available at the start of this project. This initial investigation provides some insights into the types of models that will need to be developed, on top of the modelling implications learned from this chapter. Some initial transformations will be applied to the accelerometer data to create potential activity summaries that can be used in the modelling chapters of the thesis.

State space models and Bayesian inference are introduced in Chapter 3. The theory outlined in these topics provides the background needed to understand the models fitted in the modelling stages of the thesis and introduces the algorithms used to do so.

The remaining chapters of the thesis focus on developing and fitting models to aid in forecasting glucose levels. Chapter 4 explores models to classify activity data, transformed from raw accelerometer data, into activity intensity zones. Then, Chapter 5 provides some initial univariate glucose models to provide a baseline forecast performance that can be built upon in Chapter 6. Chapter 6 aims to combine the methods and results developed for the univariate activity and glucose models to provide a joint model that improves upon the glucose forecasts in Chapter 5.

A second application area is explored in Chapter 7. This application was investigated as part of a work placement at Woodside Energy Ltd. This chapter illustrates incorporating weather data in models for forecasting the production levels of liquid natural gas (LNG).

Finally, Chapter 8 summarises the contributions and conclusions of the thesis and suggests areas for further work.

Chapter 2

Exploratory data analysis

2.1 Background

In this chapter, the time series data sets that are modelled throughout the thesis are explored. There are two types of data set that are introduced: glucose data and accelerometer data. These data sets are available across 10 distinct patients. In addition to the raw accelerometer and glucose data, times when the CGM device collecting the glucose data was calibrated by each patient are also available.

The data were collected prior to the start of the project by the Movement Laboratory research group at Newcastle University. The patient data available are unlabelled, which presented difficulties in the interpretation of patient activity levels and the identification of sources of variation in the glucose data. Additional, annotated data were therefore collected from three healthy (without type II diabetes and relatively active) subjects later in the project to compare to the initial patient data sets analysed here. These data are partially labelled with meal times and exercise and are introduced later in Chapters 4, 5 and 6.

2.2 Glucose data

The blood glucose levels of the ten patients with type II diabetes were reported by a CGM device. These observations are recorded in millimoles per litre (mmol/L) every five minutes across a five day period for each patient. Patients calibrated the CGM device with finger prick measurements to maintain the accuracy of the observations.

The glucose profiles of eight out of the 10 patients are shown in Fig. 2.1. Two of the patient data sets (from Patient 2 and Patient 6) have been discarded as they are unsuitable for addressing the objective of jointly modelling glucose and accelerometer data; for both patients, the periods of data collection for the glucose and accelerometer data do not align.

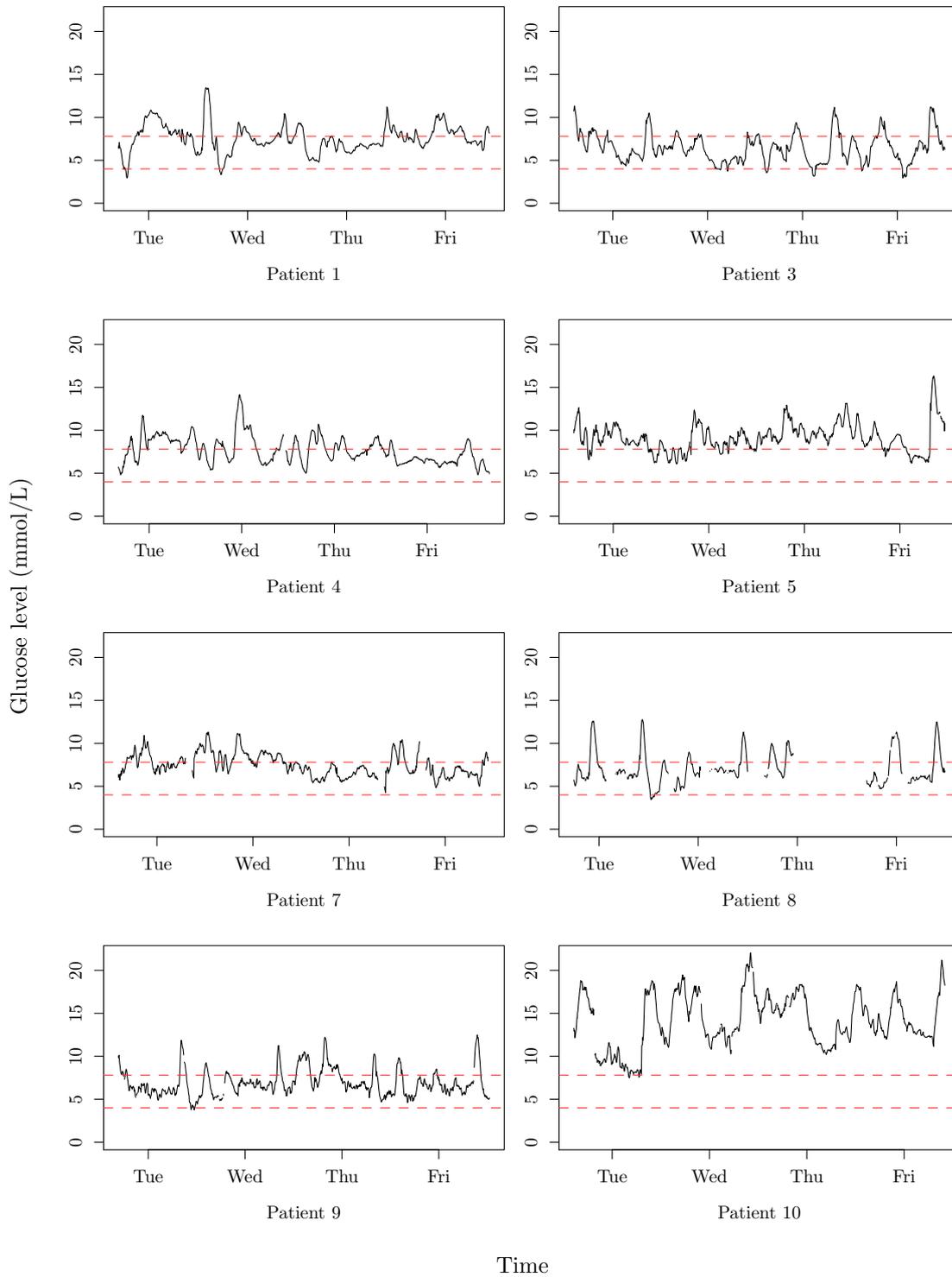


Figure 2.1: Glucose profiles of eight type II diabetic patients. The euglycaemic range is indicated by the red, dashed lines on each profile.

The euglycaemic range of glucose levels is between 4 mmol/L and 7.8 mmol/L. Type II diabetic patients have a tendency to record higher glucose levels than normal, as can be seen in Fig. 2.1 where all patients frequently record glucose levels above 7.8 mmol/L. In general the overall mean glucose level for each patient is also closer to this upper limit (or above). The higher glucose level recordings are one of the indicators that a person is diabetic.¹ Patient 10 consistently records glucose levels above 7.8 mmol/L, with some glucose observations above 20 mmol/L. Whilst higher glucose levels are dangerous and indicate hyperglycaemia, what constitutes a ‘trigger’ glucose level for a behavioural prompt must be tailored to each individual, since a reasonable threshold glucose level for Patient 2, for example, would be far too low of a threshold for Patient 10.

There is a diurnal pattern to the glucose profiles for each patient in Fig. 2.1, but the height of the peaks and number of peaks per day varies between patients and sometimes between days for the same patient. This is most likely due to the range of sizes of meals each patient eats throughout the day and throughout the week and also the range of activities they perform. There is also a substantial amount of noise in each profile; the glucose profile for Patient 4 is relatively smooth compared to that of Patient 5, but none of the profiles are completely smooth. Overall, the mean glucose level (trend) is constant across the five days for all patients, which is to be expected. These observations indicate that the models for glucose must account for seasonality and for noisy signals, as well as being able to adapt to the within-patient changes in seasonal pattern and levels of noise.

The density of each patient’s glucose profile is plotted in Fig. 2.2. The profiles are multimodal with relatively short tails, indicating that each profile could be described by a mixture of Normal densities. As in Fig. 2.1, the densities highlight the similarities between the glucose profiles of some patients. In particular, the distribution of glucose levels for Patients 1, 4 and 7 and Patients 8 and 9 can be grouped together: Patients 1, 4 and 7 have a similar mode with highest density, followed by a medium sized mode at a higher glucose level (though this is split into two modes for Patient 1); Patients 8 and 9 also share this similar mode with highest density, but the density describing their higher glucose levels is flatter, with multiple smaller modes. Patients 3 and 5 have similarly shaped densities, but on average at slightly lower glucose levels and at slightly higher glucose levels than the previously mentioned patients, respectively. Both profiles feature a main, central mode with a smaller mode to the left and another, flatter mode to the right. The density for the glucose levels of Patient 10 has a large variance, with a small mode at the top end of the range of glucose levels experienced by the other patients and then two larger modes of similar size at higher glucose levels. The modal density around 17 mmol/L is similar to the size of the density around 13 mmol/L, indicating this patient is experiencing very high glucose levels as often as they are mid-range levels of glucose.

¹<https://www.mayoclinic.org/diseases-conditions/type-2-diabetes/diagnosis-treatment/drc-20351199>

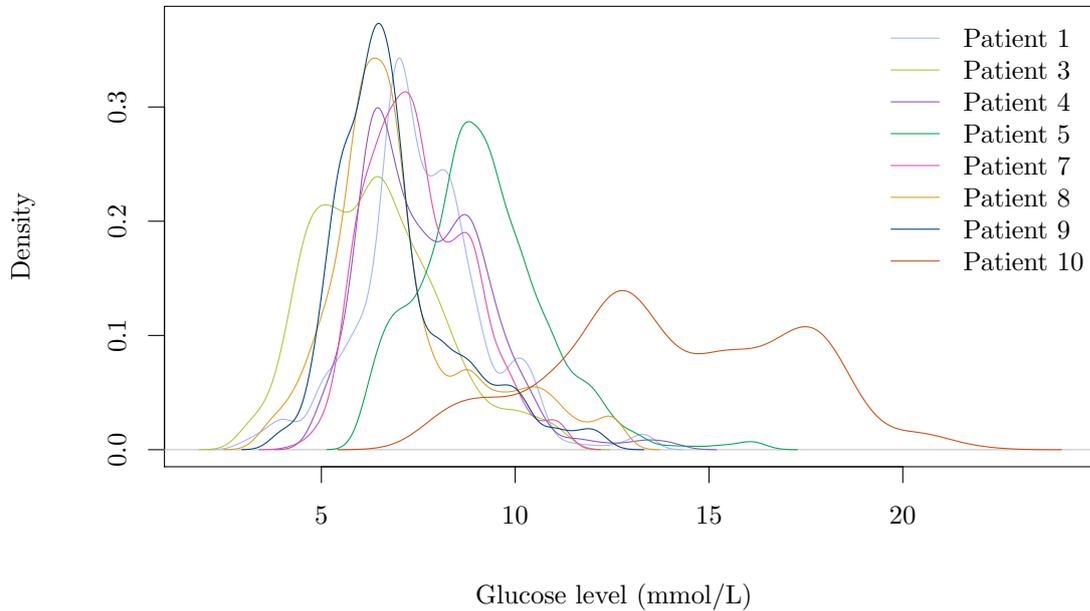


Figure 2.2: Densities of the patient glucose profiles.

Overall, although the location of the modes for each patient’s glucose profile varies, there is a clear similarity in the shapes of each profile. This supports the idea that a model could be trained on one patient’s data or by pooling information across patients, then tailored to each patient specifically as individual observations are received. This exploratory analysis also confirms the need for a dynamic, seasonal model, in particular one that could describe a mixture of Normal densities. Looking further ahead, the threshold for a behavioural prompt to be sent to a patient must be tailored to each individual, too. If glucose levels were monitored in the long term, this threshold could also be set dynamically for each patient if their glucose levels started to show an overall decrease (or increase) over time.

2.2.1 Missing data

To record glucose levels reported by the CGM device, each patient must be near a bluetooth device (typically a mobile phone) that receives the readings. If a patient moves out of range of the device when the CGM device attempts to send a reading, the observation is missed. These missing time stamps are not reported in the data automatically, so must be added in once the data have been imported, with the corresponding glucose level recorded as NA. Fig. 2.1 shows that Patient 8 has many missing values, with a large gap in their glucose profile for the majority of time points on Thursday and interspersed gaps for

the other days in their trial. The other patients have much shorter and fewer periods of missingness for all of the days. The modelling and inferential procedures used in this thesis must therefore be able to accommodate missing data, which are assumed to be missing at random. Inference is carried out in the Bayesian framework and so this essentially involves treating the missing observations as unknowns, constructing the joint posterior distribution of all missing data, parameters and other unknowns, and then integrating out the missing data. In this way it is possible to average over the uncertainty in their values. Further details of how this is performed will be discussed in Chapter 3.

2.3 Accelerometer data

The second data set available for each patient is 100 Hz tri-axial accelerometer data. The high frequency of the data introduces challenges in terms of computational impact and with interpretability. One week's worth of data has over 64 million rows and a file size of 3.5 GB, so reading this into software like R causes problems if the machine the data are being processed on has relatively low memory. This is because R reads the entire data set into the RAM of the machine, as well as all objects stored in the environment. One option to work around this problem is to read chunks of data into R at a time through a connection. This allows smaller sections of data to be analysed at a time whilst not overloading the RAM. The large file size, however, is not the only problem presented by the raw accelerometer data; the high frequency introduces a lot of noise into the signals, too. Fig. 2.3 shows a section of noisy signal along with some larger spikes, from which it is difficult to interpret the parts that are noise and the parts arising from periods of activity. Since the accelerometer data are to be used to gain insight into how active a patient has been, it is important that they can be summarised in a way that reflects this.

2.3.1 Pre-processing accelerometer data

To convert the accelerometer data into more interpretable and parsable formats, the techniques described in Section 1.3.3 are explored and evaluated. The pre-processing algorithms are separated into two categories: computing the EN (Euclidean Norm) after applying different filters to the raw signals and step count algorithms. A frequency of 1 Hz for the final activity data is considered here, with lower frequencies considered later in the thesis; see Chapter 4. The initial exploratory analysis for pre-processing the accelerometer data is carried out on the small segment (10,000 time points) of data from a single patient (Patient 3) shown in Fig. 2.3. The methods in this section take advantage of aggregating data over a one second epoch (window) to reduce the size of the transformed data set.

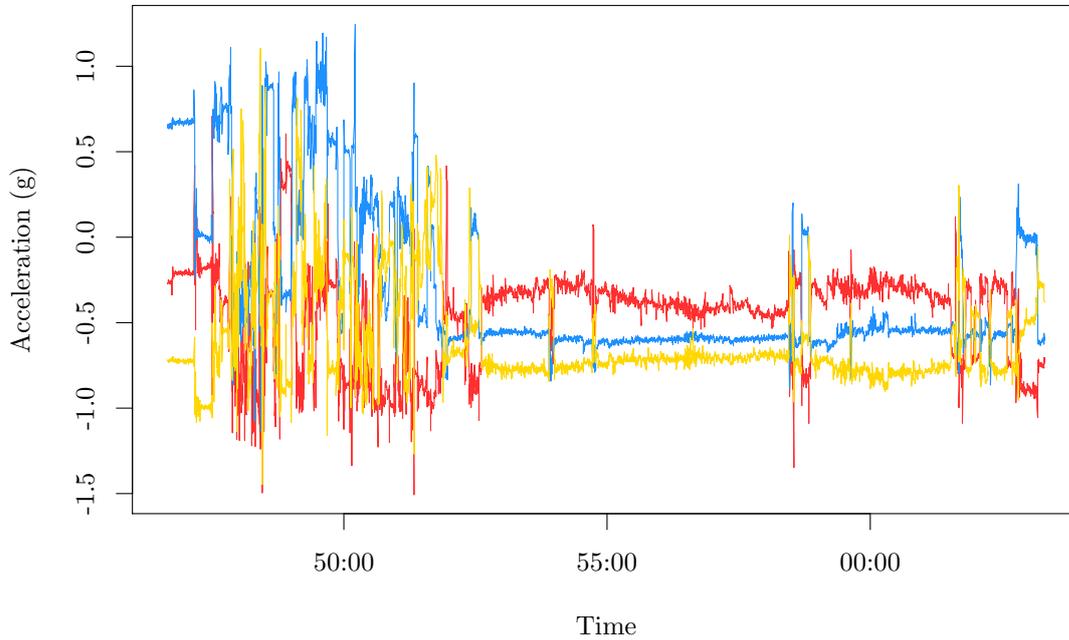


Figure 2.3: Tri-axial accelerometer data (thinned by 20 for plotting).

EN and signal filtering

As seen in the literature in Section 1.3.3, methods involving the EN are popular for pre-processing accelerometer data. The EN summarises a vector, $\underline{v} = (v_1, \dots, v_p)$, with a scalar ($e = \sqrt{\sum_{i=1}^p v_i^2}$) and is often referred to as the magnitude of the vector. Transforming the tri-axial accelerometer data using the EN can therefore be thought of as taking the magnitude of the total acceleration at each time point.

Algorithm 2.1: ENMO algorithm

1. Set $n =$ number of observations per window;
2. **for** $j = 1, \dots, m$, where m is the number of windows across the data set:
3. calculate

$$e_j = \sum_{i=1}^n \left\{ \sqrt{x_{(j-1)n+i}^2 + y_{(j-1)n+i}^2 + z_{(j-1)n+i}^2} - 1 \right\}.$$
4. **end**

The first algorithm considered here that utilises the EN is the ENMO (EN minus one) algorithm, Algorithm 2.1. ENMO subtracts one from the EN at each time point to account for the acceleration due to gravity, giving a value of zero when the patient is still. It is applied to a sample of raw data from Patient 3, with the resulting signal shown in

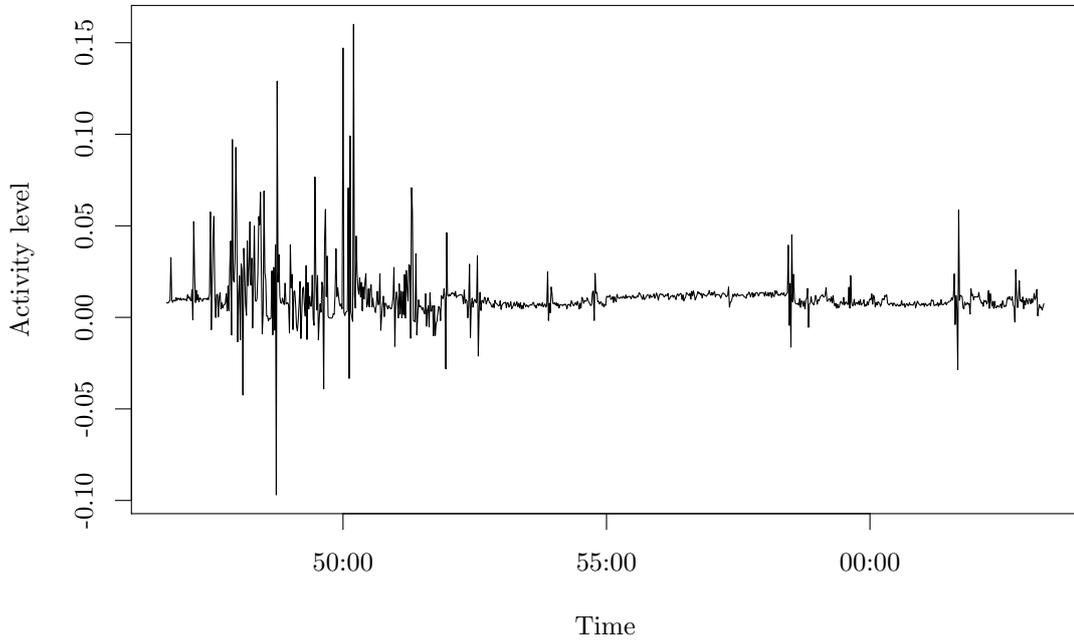


Figure 2.4: Accelerometer data summarised using the ENMO algorithm, Algorithm 2.1.

Fig. 2.4. The signal gives a clearer idea of the amount of activity being performed over time, with larger and smaller spikes reflecting the amount Patient 3 is moving. Algorithm 2.1 produces some negative activity observations. In terms of quantifying activity, a method that produces strictly positive measurements would be more interpretable, as intuitively a recording of zero would be equivalent to no activity. Negative observations are not necessarily a problem for monitoring activity, since the type of activity being performed is not important for reducing glucose levels, just the intensity.

Algorithm 2.2: Truncated ENMO algorithm

1. Set $n =$ number of observations per window;
2. **for** $j = 1, \dots, m$, where m is the number of windows across the data set:
3. calculate

$$e_j = \sum_{i=1}^n \max \left\{ 0, \sqrt{x_{(j-1)n+i}^2 + y_{(j-1)n+i}^2 + z_{(j-1)n+i}^2} - 1 \right\}.$$

4. **end**

A negative ENMO value occurs when $\sqrt{x^2 + y^2 + z^2} < 1$, meaning the magnitude of acceleration at that time point is negligible and the negative value is the result of a downward acceleration in the same direction of gravity. A negative activity observation

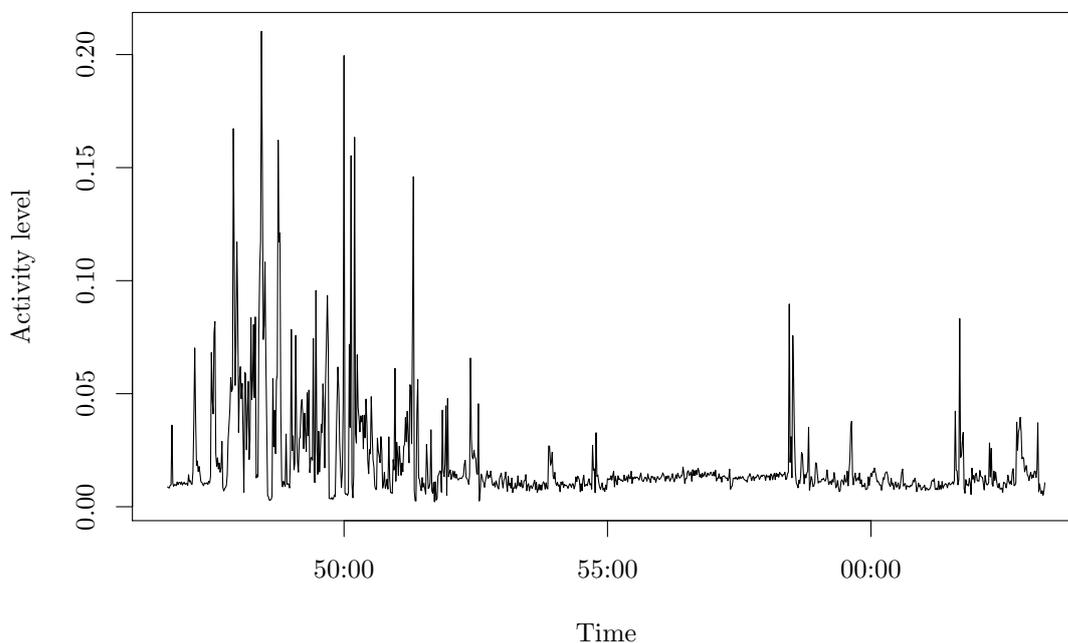


Figure 2.5: Accelerometer data summarised using the truncated ENMO algorithm, Algorithm 2.2.

($e_j < 0$) should therefore be interpreted as an epoch with a low level of activity, regardless of the magnitude of the observation. Summing over a window that contains a mix of positive and negative values reduces the overall level of activity observed for that epoch, which results in activity observations being under-reported.

An alternative approach to eliminate the impact small accelerometer readings (negative ENMO observations) have on the overall activity level is to truncate negative observations after calculating ENMO at each time point to zero, Algorithm 2.2. The resulting truncated ENMO signal is shown in Fig. 2.5. Some new, large spikes in activity have appeared, showing that the ENMO algorithm was dampening some of the activity levels being reported. This could have a much bigger impact on the activity observations per epoch if the data are downsampled to a lower frequency. Both of the activity profiles from Algorithms 2.1 and 2.2 show clear spikes in activity levels where it seems Patient 3 is at rest. In Fig. 2.5 in particular, zones of activity appear to occur in bands of clustered activity observations: ‘low’ activity appears to be observations less than 0.03, a ‘medium’ level of activity is observations less than 0.1 and higher levels of activity are above 0.1.

As mentioned in Section 1.3.3, signal filtering can be carried out on the raw accelerometer data to attenuate frequencies in a given range prior to applying the EN. Examples of the band-pass, high-pass and low-pass fourth order Butterworth filters applied to the x -axis accelerometer data are shown in Fig. 2.6. The band-pass and high-pass filters are

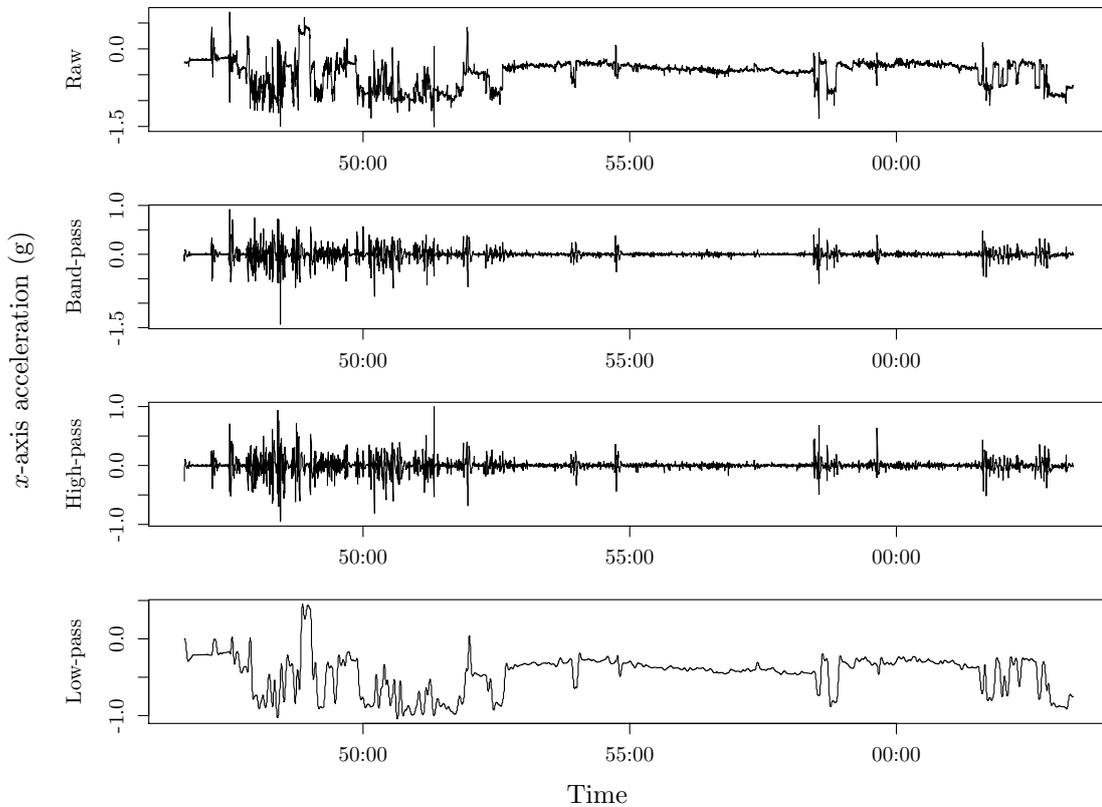


Figure 2.6: Raw x -axis accelerometer data (top) filtered using a band-pass filter (second), high-pass filter (third) and low-pass filter (bottom) (thinned by 20 for plotting).

fairly similar in how they transform the raw signal: the frequencies below 0.2 Hz are removed in both examples, resulting in a signal that is centred about zero, with frequencies above 15 Hz additionally removed in the band-pass filtered data. This is equivalent to looking only at the size of acceleration over time (in this case in the x -axis) and removing the effect of gravity. Attenuating higher frequencies in the band-pass filtered signal removes additional noise from the acceleration. In contrast, the low-pass filter smooths over the signal by picking out only the low frequencies, leaving only the trend of the signal remaining. This is equivalent to picking out the effect of gravity on each movement. Comparing the raw data to the filtered signals, the shape of the low-pass filtered signal has been removed from the raw data to produce the band-pass and high-pass filtered signals and it can be seen that superimposing the high-pass and low-pass filtered signals would reconstruct the raw data.

The BFEN (band-pass filtered EN) algorithm, Algorithm 2.3, applies a band-pass filter to the x -, y - and z -axis signals, computes the EN at each time point and then sums the transformed data over a given epoch (one second). Similarly, the HFEN (high-pass

Algorithm 2.3: BFEN algorithm

1. Apply band-pass filter, B , to raw accelerometer data in x , y and z directions:

$$B(x) = B_x; \quad B(y) = B_y; \quad B(z) = B_z;$$

2. Set n = number of observations per window;
3. **for** $j = 1, \dots, m$, where m is the number of windows across the data set:
4. calculate the EN:

$$e_j = \sum_{i=1}^n \left\{ \sqrt{B_{x,(j-1)n+i}^2 + B_{y,(j-1)n+i}^2 + B_{z,(j-1)n+i}^2} \right\}.$$

5. **end**

Algorithm 2.4: HFEN algorithm

1. Apply high-pass filter, H , to raw accelerometer data in x , y and z directions:

$$H(x) = H_x; \quad H(y) = H_y; \quad H(z) = H_z;$$

2. Set n = number of observations per window;
3. **for** $j = 1, \dots, m$, where m is the number of windows across the data set:
4. calculate the EN:

$$e_j = \sum_{i=1}^n \left\{ \sqrt{H_{x,(j-1)n+i}^2 + H_{y,(j-1)n+i}^2 + H_{z,(j-1)n+i}^2} \right\}.$$

5. **end**

filtered EN) algorithm, Algorithm 2.4, follows this process, too, but substitutes applying a band-pass filter with applying a high-pass filter. Both algorithms result in activity data sets that are positive-valued, Fig. 2.7 and Fig. 2.8. The resulting activity profiles have no obvious differences, which suggests that, for this sample of data, there are not many higher frequencies present that are skewing the overall activity profile. However, this might not be the case for all patients' accelerometer data, so use of the BFEN algorithm is preferred over HFEN to avoid picking up extra noise when interpreting activity levels. There is less distinction between the spikes in the activity profiles compared to the ENMO and truncated ENMO profiles, making it harder to infer by eye different zones of activity.

The HFEN₊ algorithm, Algorithm 2.5, incorporates adding together a low-pass filtered signal and high-pass filtered signal then subtracting one at each time step. The low-pass filtered x -, y - and z -axis signals (transformed using the EN) are shown in Fig. 2.9. This signal is centred about one, which is equivalent to the effect of gravity when at rest, with no rotational movement. After subtracting one from this signal, the observations are the

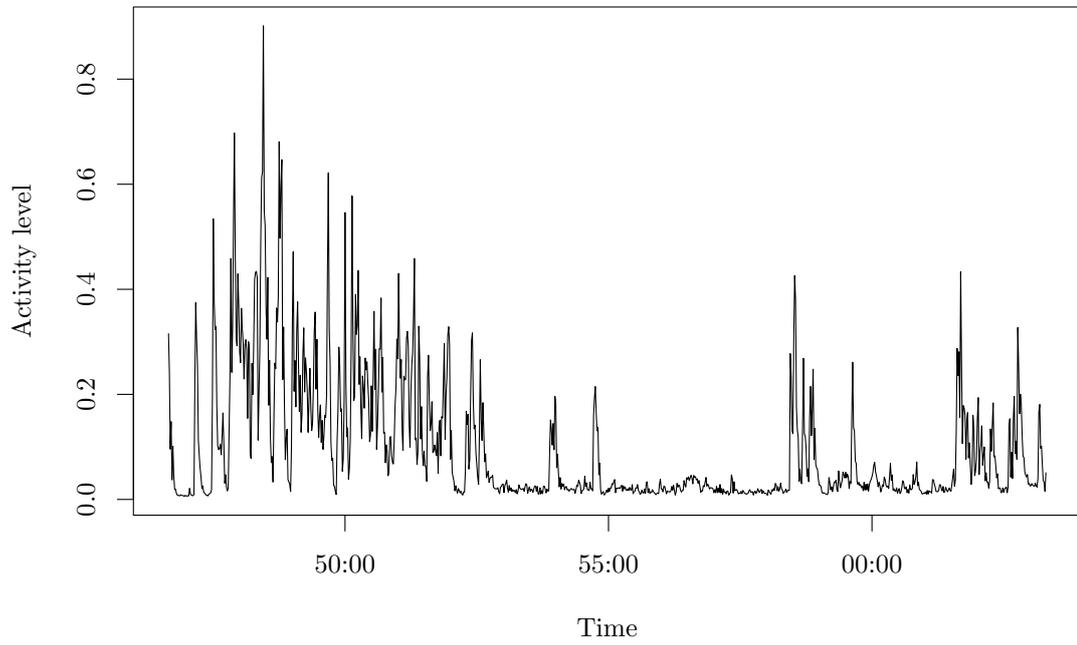


Figure 2.7: Accelerometer data summarised using the BFEN algorithm, Algorithm 2.3.

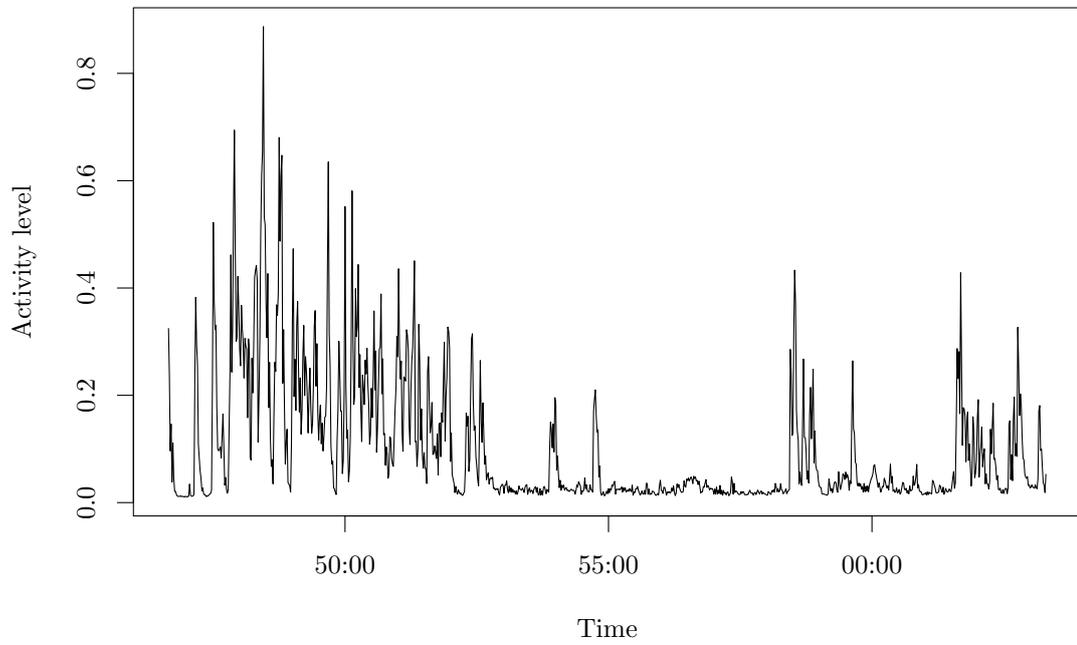


Figure 2.8: Accelerometer data summarised using the HFEN algorithm, Algorithm 2.4.

Algorithm 2.5: HFEN₊ algorithm

1. Apply low-pass filter, L , to raw accelerometer data in x , y and z directions:

$$L(x) = L_x; \quad L(y) = L_y; \quad L(z) = L_z;$$

2. Apply high-pass filter, H , to raw accelerometer data in x , y and z directions:

$$H(x) = H_x; \quad H(y) = H_y; \quad H(z) = H_z;$$

3. Set n = number of observations per window;
4. **for** $j = 1, \dots, m$, where m is the number of windows across the data set:
5. calculate the EN for the low-pass and high-pass filtered data and minus 1:

$$e_j = \sum_{i=1}^n \left\{ \sqrt{H_{x,(j-1)n+i}^2 + H_{y,(j-1)n+i}^2 + H_{z,(j-1)n+i}^2} + \sqrt{L_{x,(j-1)n+i}^2 + L_{y,(j-1)n+i}^2 + L_{z,(j-1)n+i}^2} - 1 \right\}.$$

6. **end**

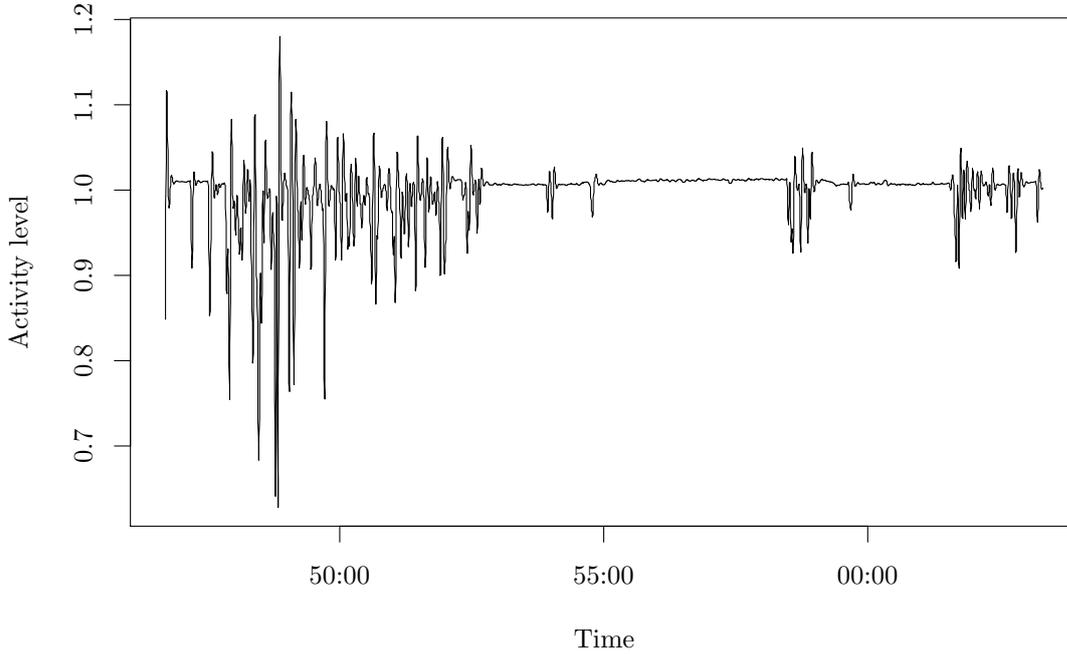


Figure 2.9: Accelerometer data summarised using a low-pass filter and EN.

parts of the acceleration that are caused by rotational movement, that might otherwise have been attenuated by a pure high-pass filter. The HFEN₊ algorithm can therefore be thought of as a corrected version of the HFEN algorithm (van Hees *et al.*, 2011).

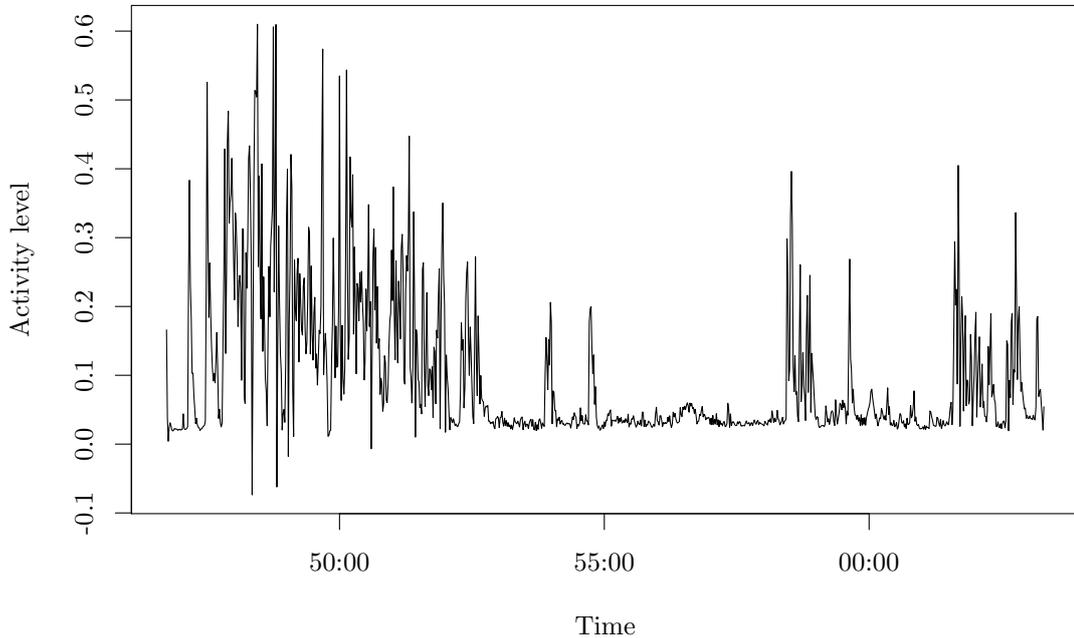


Figure 2.10: Accelerometer data summarised using the HFEN_+ algorithm, Algorithm 2.5.

The activity profile produced by Algorithm 2.5 is shown in Fig. 2.10. This corrected signal reintroduces negative values into the data set, which, when summarised over larger epochs, would introduce the same concerns that are raised from Algorithm 2.1. Overall, the activity profile is similar to the BFEN and HFEN algorithms, Fig. 2.7 and Fig. 2.8, respectively, but the large spike in activity around 48:00 has been attenuated, suggesting that part of this movement is rotational.

Step count algorithms

Here, two algorithms to infer a step count are considered. The second algorithm builds on the first by adding constraints to the number of steps that can be detected.

Algorithm 2.6 outlines a simple step count. The raw accelerometer data are processed using BFEN (Algorithm 2.3) without summing over an epoch ($n = 1$) and a step is counted each time a negative slope in the signal crosses a threshold that is calculated dynamically, every 0.5 seconds. The resulting step count profile is shown in Fig. 2.11. The profile is very noisy and, contrary to the profiles generated from the EN algorithms discussed prior to this, shows high levels of activity from 53:00 to 55:00. This is a side effect of using a dynamic threshold; as the BFEN signal fluctuates slightly even when the patient is at rest and the threshold for steps during these periods is low, the simple step count algorithm will class each time the signal crosses the threshold as a step.

Algorithm 2.6: Simple step count algorithm

1. Preprocess accelerometer data using BFEN (Algorithm 2.3) with $n = 1$ to get e_1, e_2, e_3, \dots ;
2. **for** $i = 1, \dots, m$, where $m =$ number of windows across data set:
3. set $count_i = 0$;
4. **for** $j = 1, \dots, n'$, where $n' =$ number of 0.5 second windows in i , let $c = (i - 1)n'$ and $l =$ number of observations in j :
5. calculate threshold

$$u_j = \frac{\max\{e_{(c+j-1)l+1:(c+j)l}\} + \min\{e_{(c+j-1)l+1:(c+j)l}\}}{2};$$
6. **for** $k = 1, \dots, l$:
7. **if** $e_{(c+j-1)l+k} < u_j$ & $e_{(c+j-1)l+k-1} > u_j$ **then** $count_i = count_i + 1$;
8. **end**
9. **end**
10. **end**

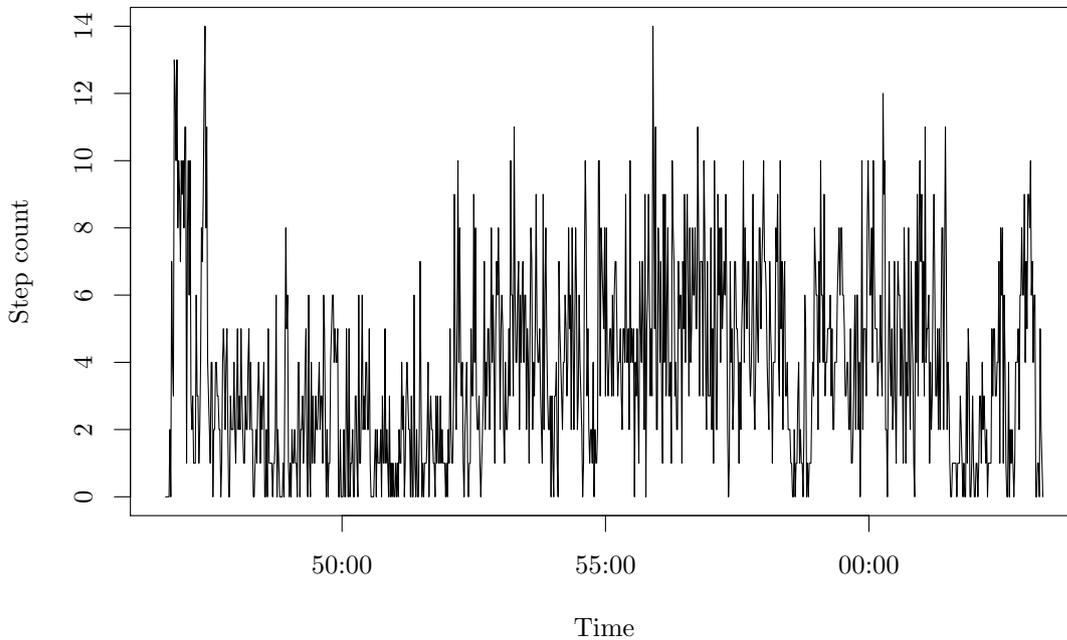


Figure 2.11: Accelerometer data summarised using the simple step count, Algorithm 2.6.

Algorithm 2.7: Complete step count algorithm

1. Preprocess accelerometer data using BFEN (Algorithm 2.3) with $n = 1$ to get e_1, e_2, e_3, \dots ;
2. Set $q = 0$;
3. **for** $i = 1, \dots, m$, where $m =$ number of windows across data set:
4. set $count_i = 0$;
5. **for** $j = 1, \dots, n'$, where $n' =$ number of 0.5 second windows in i , let $c = (i - 1)n'$ and $l =$ number of observations in j :
6. calculate threshold

$$u_j = \frac{\max\{e_{(c+j-1)l+1:(c+j)l}\} + \min\{e_{(c+j-1)l+1:(c+j)l}\}}{2};$$
7. **for** $k = 1, \dots, l$:
8. **if** $e_{(c+j-1)l+k} < u_j$ & $e_{(c+j-1)l+k-1} > u_j$ **then** $q = q + 1$;
 $p_q = (c + j - 1)l + k$;
9. **end**
10. **end**
11. **end**
12. Calculate time difference between elements of $\underline{p} = (p_1, \dots, p_q)$ to get d_1, d_2, \dots, d_{q-1} ;
13. Set $s = 0$;
14. **for** $r = 1, \dots, q - 1$:
15. **if** $d_r < 0.2$ seconds | $d_r > 2$ seconds **then** $s = s + 1$; $D_s = r$;
16. **end**
17. **for** $w = 2, \dots, s$:
18. **if** $D_w - D_{w-1} \geq 4$ **then** let $\underline{v} = \lfloor \frac{p_{D_{w-1}:D_w}}{n'l} \rfloor$ and $count_{\underline{v}} = count_{\underline{v}} + 1$;
19. **end**

The complete step count algorithm, Algorithm 2.7, imposes the constraints on detecting steps set out by Zhao (2010) to yield a more realistic step count profile: steps must occur within 0.2 to two seconds of each other, as it is unrealistic for a person to run more than five steps per second or walk less than one step every two seconds; and steps must be part of a rhythmic pattern, so the count only begins once four steps have been taken consecutively. The results of this more advanced algorithm are shown in Fig. 2.12. This profile more closely resembles the activity profiles in Figs 2.7, 2.8 and 2.10, with a prolonged burst of activity up until 53:00, followed by more sparse bursts of lower intensity activity. The complete step count algorithm picks up some additional steps in the parts of the profile that appear flat in the EN and filtering algorithms, which suggests that even with the constraints, sections of noise in the data that are periodic can still be mistaken for activity. The output from Algorithm 2.7 still appears more plausible than the output from Algorithm 2.6, however. In terms of number of steps, the output is also slightly more

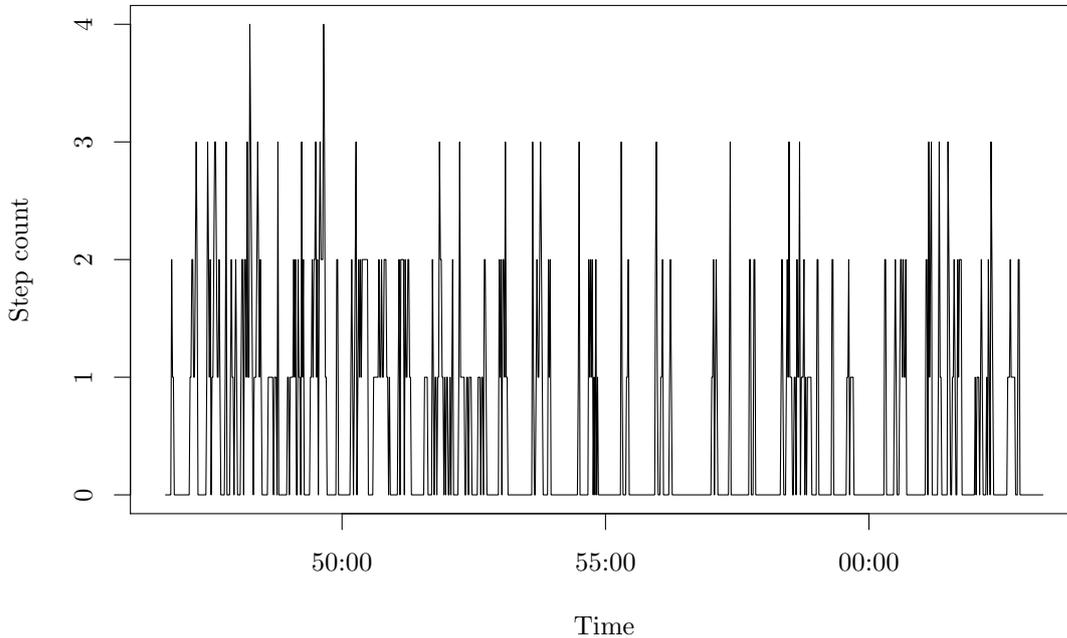


Figure 2.12: Accelerometer data summarised using the complete step count, Algorithm 2.7.

interpretable than the scale produced by the EN algorithms, as it is easier to visualise four steps than it is to think of 0.8 as an amount of activity, for example.

Across all of the algorithms discussed in this section, excluding Algorithm 2.6, the resulting activity profiles show clear periods of activity and inactivity and can be readily interpreted. The purpose of transforming the accelerometer data is to quantify the amount of activity patients are doing, so it can be used as a predictor of glucose levels. Each activity summary has positive aspects: the ENMO and truncated ENMO data sets have clear clusters of observations that suggest three clear activity zones; the BFEN, HFEN and HFEN₊ algorithms remove unwanted noise and trends from the data by including signal filtering; and the complete step count data set is most easily interpreted. A selection of activity summaries will therefore be assessed further in Chapter 4, to see which is best for achieving the aims of the thesis: (i) the ENMO data are chosen as an example of a data set that contains negative values; (ii) the BFEN data are chosen as a positive-valued, filtered data set; (iii) the complete step count data are chosen for comparison as they are discrete-valued.

2.4 Relationship between glucose and accelerometer data

The literature on the effect of activity on glucose levels unanimously reports that doing some activity, in particular after meals and in regular short bouts throughout the day, has

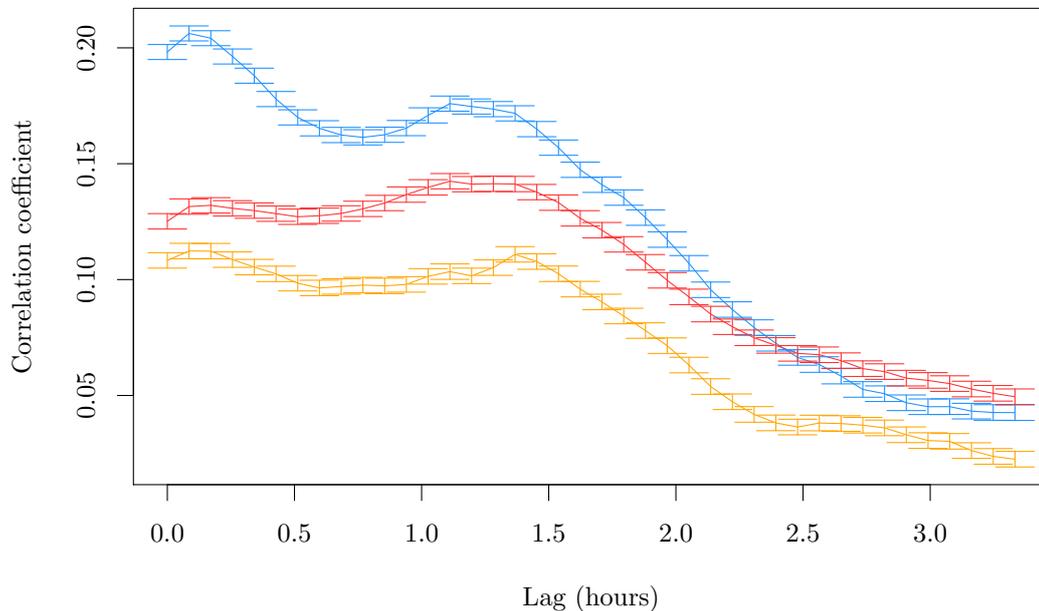


Figure 2.13: Correlation between glucose and lagged activity levels with 95% confidence interval error bars (plotted at five minute intervals), described by BFEN (blue), ENMO (gold) and the complete step count (red).

a positive impact on the time it takes for postprandial spikes in glucose levels to reduce back to an acceptable level (Dunstan *et al.*, 2012; Henson *et al.*, 2016; McCarthy *et al.*, 2017). Observing this impact directly in patient glucose profiles is difficult. In a clinical setting, the impact of activity on glucose levels can be measured more easily by varying this between subjects whilst other influential factors, such as diet and meal times of each subject, are controlled. In real life, data are unlabelled and the many other variables affecting a person's glucose levels are not controlled, which makes observing the impact of activity on glucose levels challenging.

When plotting glucose levels against activity levels, a negative linear relationship is expected: higher activity levels result in lower glucose levels. Intuitively, this relationship is not expected to be observed immediately, since there will be a time lag between the moment an activity is performed and the time it takes for the body to respond, resulting in reduced glucose levels. As the glucose data and activity data sets have mismatched frequencies, a spline is fitted to the glucose data so that a correlation can be directly calculated, without reducing the frequency of the activity data. This is done via the `spline` function in R, using the default "fmm" method.

The correlations between glucose levels and lagged activity levels for the three activity summaries chosen to be analysed in the previous section are shown in Fig. 2.13. The

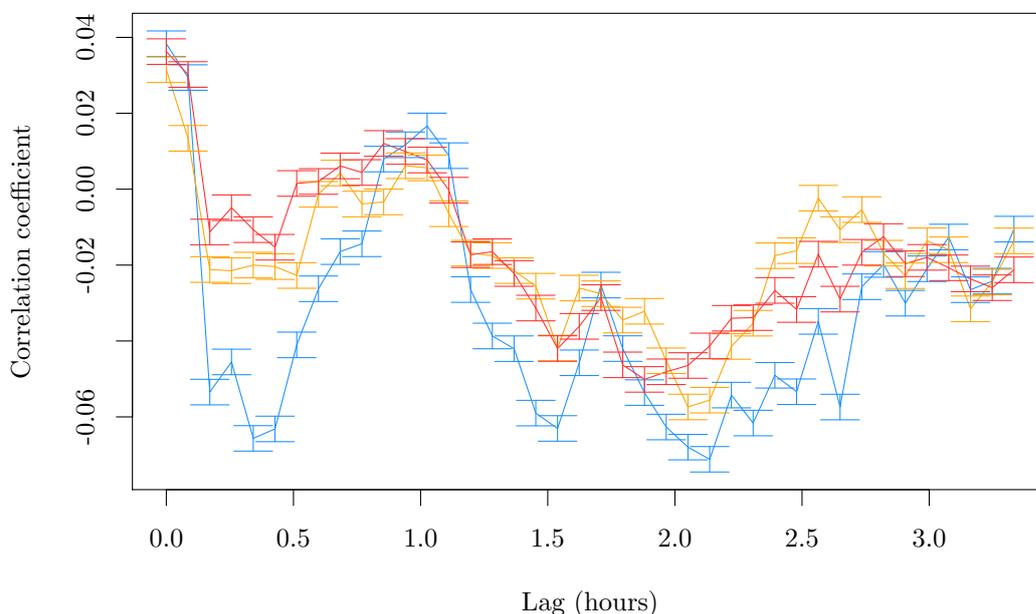


Figure 2.14: Correlation between change in glucose levels and lagged activity levels with 95% confidence interval error bars (plotted at five minute intervals), described by BFEN (blue), ENMO (gold) and the complete step count (red).

number of correlations plotted has been thinned by five for clarity. The plot shows a positive relationship between glucose levels and lagged activity levels for all three summaries, which after 1.5 hours steadily declines towards zero. There are two modes in the correlation profiles for each summary, one at the five minute lag and one at the 1.5 hour lag. The first mode is much more pronounced for the BFEN data and is where the strongest correlation is, whilst the second increase in the correlation coefficient is similar across all three data sets. The BFEN data consistently show a higher correlation with glucose levels for all time lags than the ENMO and step count data, but none of the correlation coefficients across the data sets are particularly strong. The positive relationship between glucose and activity levels was also not expected and does not follow the results in the literature.

An alternative option for exploring the relationship between glucose levels and activity levels is to instead look at the change in glucose against lagged activity, since higher activity levels ultimately result in glucose levels declining. A negative linear relationship is again expected. The results in Fig. 2.14 now show a negative relationship between lagged activity levels and the change in glucose levels, but the correlations describing this are very small. There are two main troughs in the correlation profiles across all of the activity summaries: one around the 30 minute lag and one around two hours. This suggests that the dynamic effect of activity on future changes in glucose levels may depend on the intensity of the

activity being performed. A model that prescribes a linear relationship between glucose and lagged activity, over a range of lags, is unlikely to capture this complex relationship. Therefore, a flexible approach will be adopted, in which activity observations are first classified into intensity zones. This is explored further in the joint modelling stages of this thesis (Chapter 6).

2.5 Summary

In this chapter, an exploratory analysis of the glucose datasets of eight type II diabetic patients and of a sample of accelerometer data was carried out.

The distribution of glucose observations for each patient was analysed and showed that each profile could be described by a mixture of Normal distributions. Whilst some of the glucose profiles suffered from missing data, it is assumed that these data are missing at random due to the nature of the way the data are recorded: it is assumed that there is no underlying cause for the patient being too far from the bluetooth device that would receive the glucose observation.

Different pre-processing methods are explored for converting the raw tri-axial accelerometer data into a univariate activity summary. These algorithms involved signal filtering, the EN and computing step counts and reduced the 100 Hz data down to 1 Hz by summing the processed signals over one second epochs. The activity summaries typically show positively skewed data with a long tail and the ENMO and HFEN₊ summaries contain negative observations. To assess how portable the models fitted in this thesis are, three summaries are chosen for further analysis and model fitting: the ENMO, BFEN and complete step count algorithms. ENMO provides an example of a dataset with negative observations and a very long tail, the BFEN algorithm is strictly positive, and the complete step count algorithm is an example of a discrete data summary.

An initial investigation into the relationship between glucose and the three activity summaries is also conducted. The correlation between glucose and each activity summary is computed at different lags, as well as the correlation between the change in glucose and lagged activity. These correlation profiles show that the relationship between these two variables over time is non-linear and requires further analysis. Both figures (2.13 and 2.14, respectively) show a multimodal correlation profile for all three activity summaries, suggesting there are different relationships between glucose and activity when activity is performed at different intensities. The activity observations will therefore be classified into intensity zones before further analysis is carried out.

Chapter 3

Background

3.1 State space models

State space models offer a flexible framework for modelling time series. They can be applied to data with irregular patterns and that lack stability in their underlying system, as well as to univariate and multivariate time series. State space models also offer a natural progression to including additional explanatory time series in the form of regression terms, which can provide necessary knowledge to explain away non-stationarity, such as jumps in the mean or non-constant variance. For background on state space models and their applications to time series, see Doucet *et al.* (2001) and West & Harrison (2006).

Traditional time series models, such as autoregressive moving average (ARMA) models, are fitted directly to observations, so there is no clean way to separate ‘signal’ from ‘noise’. The models applied typically require stationarity in order to be easily fitted, which often requires a transformation of the raw data set. This is straightforward when the data have a regular pattern or trend, but becomes more difficult to achieve with more complex data sets. State space models introduce a simple form of dependency structure; the models consider an observed time series $\{Y_t\}$ as the output of a system of hidden states $\{\theta_t\}$, which form a Markov chain. Here $\{Y_t\}$ are considered to be univariate, though the methods extend to multivariate time series. The Markovian dependency in the hidden states means that θ_t depends only on θ_{t-1} , i.e. there is no additional information about θ_t in $\theta_{1:t-2}$ and these states are conditionally independent given θ_{t-1} . Since the observations

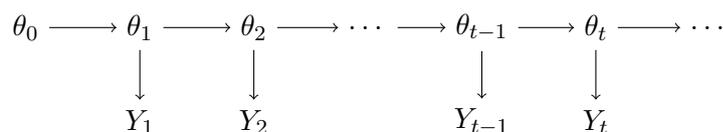


Figure 3.1: A DAG showing the dependency structure between the hidden states, $\{\theta_t\}$, and observations, $\{Y_t\}$ in a state space model.

are measurements of the states, Y_t depends only on θ_t and is conditionally independent of the previous observations $Y_{1:t-1}$, given θ_t . The observations are, however, autocorrelated marginally due to their dependence on the states.

The structure of a state space model is shown in Fig. 3.1 via a directed acyclic graph (DAG). A DAG is a graphical representation of the relationship between variables. Each variable is depicted as a node, and arrows between nodes indicate their conditional independence structure. The graph is said to be acyclic because there are no loops. As a result, the conditional independence relationships defining a state space model can be deduced from Fig. 3.1; for example, it is observed that Y_t is conditionally independent of $(\theta_{0:t-1}, Y_{1:t-1})$ given θ_t , since to connect Y_t with any Y_s or θ_s , for some $s < t$, the path must cross through θ_t . This results in $f_t(y_t|\theta_{0:t-1}, y_{1:t-1}) = f_t(y_t|\theta_t)$. This illustration relies on formal properties of graphical models that can be proved (Lauritzen, 1996).

A state space model can be defined more formally as consisting of a p -variate time series $\{\theta_t : t = 0, 1, \dots\}$ and an m -variate time series $\{Y_t : t = 1, 2, \dots\}$, satisfying the assumptions:

A1 $\{\theta_t\}$ form a Markov chain.

A2 $\{Y_t\}$ are conditionally independent given the states $\{\theta_t\}$.

These assumptions result in a model that is completely specified by its initial distribution $\pi(\theta_0)$ and conditional density or mass functions, $g_t(\theta_t|\theta_{t-1})$ and $f_t(y_t|\theta_t)$, $t \geq 1$. It can therefore be written that, for any $t > 0$,

$$\pi(\theta_{0:T}, y_{1:T}) = \pi(\theta_0) \prod_{t=1}^T g_t(\theta_t|\theta_{t-1}) f_t(y_t|\theta_t). \quad (3.1)$$

From Eq. (3.1) any other distribution of interest can be derived. The form of this joint distribution means computations can be implemented by recursive algorithms, including the problems of estimation and forecasting, presented in Section 3.2.3 (Petris *et al.*, 2009, Chapter 2). These algorithms are considered in the contexts of hidden Markov models (HMMs) and DLMS in Sections 3.3.2 and 3.4.2.

3.1.1 Hidden Markov models

A HMM is a state space model in which the hidden states $\{Z_t\}$ are discrete-valued random variables, i.e. $Z_t \in \{z_1^*, \dots, z_K^*\}$ for K hidden states. Here $\theta_t = Z_t$ to denote the discrete state space. The sequence of hidden states is governed by $K \times K$ transition matrices Π_t , analogous to the state transition kernel g_t in a state space model where θ_t is continuous-valued. In Π_t , each element $\pi_{t,i,j}$ represents the probability of transitioning to state j from state i at time $t \geq 1$. Only time homogeneous HMMs are considered in this thesis,

in which the transition probabilities $\Pi_t = \Pi$ and within-state (conditional) distributions $f_t = f$ do not vary over time. The observed process $\{Y_t\}$ in a HMM can be discrete or continuous-valued and can be univariate or multivariate.

HMMs facilitate conversations surrounding an observed sequence of events as well as the hidden sequence of events associated with it. Inferring the hidden states of a time series allows more information about the process to be gained. In essence, a HMM adds labels to an otherwise unlabelled set of time ordered data.

As in general state space models, an initial distribution, here $\underline{\nu} = (\nu_1, \dots, \nu_K)$ where $\nu_k = Pr(Z_0 = k)$, over the states must be specified to start the chain. In the case of the Markov chain being irreducible and aperiodic, a good selection for the initial distribution is the stationary distribution of the chain, defined as the solution to the matrix equation $\underline{\delta}\Pi = \underline{\delta}$, for row vector $\underline{\delta}$. The stationary distribution is also the distribution that the Markov chain converges to, regardless of the initial state. A Markov chain is said to be irreducible if each state can be reached from any other state in a finite number of steps, i.e. for all i, j and finite h , $Pr(Z_{t+h} = j | Z_t = i) > 0$. A Markov chain is periodic if, for any given state j , the chain returns to that state in a finite number of steps d_j (or a multiple of d_j) after starting from it, i.e.

$$d_j = \gcd\{h : Pr(Z_{t+h} = j, Z_{t+h-1} \neq j, \dots, Z_{t+1} \neq j | Z_t = j) > 1\},$$

where gcd is the greatest common divisor. A Markov chain is then aperiodic if, for all states, it is not periodic, i.e. for all j , $d_j = 1$. For the purpose of this thesis, the initial distribution is considered separately from the stationary distribution in the models. This is justified since choosing the initial distribution to be equal to the stationary distribution would prevent the selection of conjugate priors, thereby complicating computational inference and, given a sufficiently long time series, the choice of initial distribution becomes irrelevant.

Non-standard versions of HMMs allow for different orders of dependency in the hidden states, where Z_t depends on $Z_{t-1:t-d}$ and $d \neq 1$. A special case of this is when $d = 0$, so the $\{Z_t\}$ are independent. This is known as a finite mixture model and is briefly considered in Chapter 4. The discrete states of HMMs also allow for different families of within-state distributions to be specified, i.e. a model can use different distributions for each hidden state, as long as the within-state likelihood can be evaluated. Further information on HMMs, including inference for and examples, see Frühwirth-Schnatter (2006) and Rabiner (1989).

3.1.2 Dynamic linear models

A DLM is a type of state space model in which the conditional dependence of Y_t on θ_t and θ_t on θ_{t-1} is linear and the distributions, g_t and f_t , are Gaussian. Specifically, there is a Gaussian prior distribution on the initial p -dimensional state vector, θ_0 , an observation equation, Eq. (3.2), and a system (or state) equation, Eq. (3.3), for $t \geq 1$:

$$\theta_0 \sim N_p(m_0, C_0),$$

$$Y_t = F_t \theta_t + v_t, \quad v_t \sim N(0, V_t), \quad (3.2)$$

$$\theta_t = G_t \theta_{t-1} + w_t, \quad w_t \sim N_p(0, W_t). \quad (3.3)$$

The observation and system equations show a linear relationship between the observations and the hidden states and between the hidden states themselves, via known coefficient matrices F_t and G_t with dimensions $1 \times p$ and $p \times p$, respectively. The system of equations evolves at each time step with some added noise at both levels: $\{v_t\}$ and $\{w_t\}$. The observation noise and system noise are independent sequences of independent Gaussian random errors, both with mean zero and known variance matrices V_t and W_t , respectively. As the w_t are independent and identically distributed (i.i.d.), the θ_t form a Markov chain. Additionally, the initial state vector θ_0 is independent of $\{v_t\}$ and $\{w_t\}$. This specifies a time-inhomogeneous model, though special time-homogeneous cases of DLMs will also be considered, where $F_t = F$, $G_t = G$, $V_t = V$ and $W_t = W$. A good introduction to DLMs is provided by Petris *et al.* (2009).

Superposition of models

The Gaussian joint distribution for $\{(\theta_t, Y_t)\}$ generated by a DLM results in analytical solutions being available for the problems of estimation and forecasting via the Kalman filter (see Section 3.4.2). The prerequisite for these problems is a completely specified model, which can be difficult to achieve for complex data sets. The simplest approach to take to overcome this difficulty is to combine separate DLMs into one, known as superposition. As DLMs are, by definition, linear, models can be combined linearly, too. Each individual model represents a different aspect of the desired time series, for example the trend or seasonality, and are added together to construct a more complex overall model for the time series. Given a univariate time series $\{Y_t\}$ decomposed into $n > 1$ independent components, $\{Y_{1,t}, \dots, Y_{n,t}\}$, such that Y_t is the sum of the individual observations, $Y_t = \sum_{i=1}^n Y_{i,t}$, superposition of such components is as follows:

- the observation matrix, F_t and the state vector, θ_t , are the concatenation of the

individual model observation matrices and distinct state vectors, respectively

$$F_t = \begin{bmatrix} F_{1,t} & \dots & F_{n,t} \end{bmatrix}, \quad \theta_t = \begin{bmatrix} \theta_{1,t} & \dots & \theta_{n,t} \end{bmatrix}';$$

- the observation variance, V_t , is the sum of the individual variances

$$V_t = \sum_{i=1}^n V_{i,t};$$

- the state matrix, G_t , and the system covariance matrix, W_t , are block diagonal matrices, with the individual state matrices and system covariance matrices on the diagonals

$$G_t = \begin{bmatrix} G_{1,t} & 0 & \dots & 0 \\ 0 & G_{2,t} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & G_{n,t} \end{bmatrix}, \quad W_t = \begin{bmatrix} W_{1,t} & 0 & \dots & 0 \\ 0 & W_{2,t} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & W_{n,t} \end{bmatrix}.$$

3.1.3 Model specifications for DLMS

Here, the families of models used to describe different time series features are introduced. More specifically, the structure of DLM components for modelling trend and seasonality is outlined. The combination of such components results in the classical trend + seasonal component + noise time series decomposition.

Trend models

The simplest and most common way of modelling the trend of a time series is by a polynomial DLM. In practice, this model is very unstable for anything beyond locally-linear, but is demonstrated here as it will be used within more complex DLMS. The general form of a n th order polynomial model is represented in a n -dimensional state space with time invariant coefficient matrices $F_t = F$ and $G_t = G$ and constant observation variance $V_t = V$ and system covariance $W_t = W$, of the forms

$$F = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}, \quad G = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \\ 0 & 1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & 1 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \quad W = \begin{bmatrix} W_1 & 0 & \dots & 0 \\ 0 & W_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & W_n \end{bmatrix}.$$

Choosing a sufficiently large n for the order of a polynomial model can result in a

model sensitive enough to forecast any reasonable trajectory. Typically though, the trend of a model is a smoothed version of the overall time series, so smaller values for n are usually chosen. Two examples are $n = 1$, known as a random walk with noise model, and $n = 2$, known as a linear growth model.

The random walk with noise model, also known as a local level model, is the simplest form of DLM and is used when data have no clear trend. In this model the states are taken to be the mean at time t , $\theta_t = \mu_t$, and $F = G = 1$, with dimensions $m = p = 1$:

$$\begin{aligned} Y_t &= \mu_t + v_t, & v_t &\sim N(0, V), \\ \mu_t &= \mu_{t-1} + w_t, & w_t &\sim N(0, W). \end{aligned}$$

The observations $\{Y_t\}$ are modelled as noisy observations of the local mean level, whilst the random walk for μ_t is also subjected to noise, resulting in a non-stationary trend. Allowing $W = 0$ results in a stationary, constant mean model. Forecasts from this model at time t are constant, equal to the mean $m_t = E(\mu_t|y_{1:t})$ at time t for any k -step-ahead forecast, i.e. $f_t(k) = E(Y_{t+k}|y_{1:t}) = m_t$.

The linear growth model builds on the local level model by adding a time-varying slope β_t to the random walk on $\theta_t = (\mu_t, \beta_t)'$:

$$\begin{aligned} Y_t &= \mu_t + v_t, & v_t &\sim N(0, V), \\ \mu_t &= \mu_{t-1} + \beta_{t-1} + w_{1,t}, & w_{1,t} &\sim N(0, \sigma_\mu), \\ \beta_t &= \beta_{t-1} + w_{2,t}, & w_{2,t} &\sim N(0, \sigma_\beta). \end{aligned}$$

The model assumes that μ_t increases linearly in time and so offers a more flexible forecast than a random walk plus noise model alone. The forecasts at time t reflect the current mean level $m_{1,t} = E(\mu_t|y_{1:t})$ plus k times the current slope $m_{2,t} = E(\beta_t|y_{1:t})$ in a k -step-ahead forecast, i.e. $f_t(k) = m_{1,t} + km_{2,t}$.

Seasonal models

Seasonal models are important in capturing the repeated, cyclical behaviour in a time series. Two ways to do this are via a seasonal factor model or a Fourier form seasonal model.

Consider a completely seasonal time series $\{Y_t\}$ with period s . Each observation can be thought of as the mean for the corresponding time interval across each period, α_i , for $i = 1, \dots, s$, plus some noise. A seasonal factor model uses this idea, setting θ_t equal to some permutation of the α_i in a s -dimensional space. This configuration is fine for data with few observations per period, for example quarterly data with a yearly pattern where $s = 4$, however the number of parameters in the model increases quadratically with

s and so estimation and forecasting becomes computationally intensive when s is large, particularly when parameters are unknown.

A Fourier form seasonal model offers an alternative method of capturing seasonality, with a reduced parameter space. Instead of using s seasonal effects, the model approximates the period of a time series with a number of harmonics, $q \leq s/2$. The number of harmonics chosen determines the smoothness of the seasonal cycle, ranging from smoothest using just a few harmonics to roughest when $q = \lfloor s/2 \rfloor$, where $\lfloor \cdot \rfloor$ denotes the floor function; choosing $q = 0$ recovers a constant function. This introduces the notion that a time series with large s can be modelled with as little as 1 harmonic, offering a more parsimonious solution to capturing the periodicity in a time series by vastly reducing the size of the parameter space needed to do so.

In a Fourier form seasonal model with q harmonics, the evolution matrix for the j th harmonic is

$$H_j = \begin{bmatrix} \cos \omega_j & \sin \omega_j \\ -\sin \omega_j & \cos \omega_j \end{bmatrix},$$

given the set of Fourier frequencies $\omega_j = \frac{2\pi j}{s}$. The corresponding observation matrix is $F_j = \begin{bmatrix} 1 & 0 \end{bmatrix}$. The entire seasonal model has system and observation matrices

$$G = \text{blockdiag}(H_1, \dots, H_q) \quad \text{and} \quad F = \begin{bmatrix} 1 & 0 & 1 & \dots & 0 \end{bmatrix},$$

resulting in a $(2 \times q)$ -dimensional state space. In the special case where s is odd and a saturated seasonal model is required, $H_{q+1} = 1$ and $F_{q+1} = 1$ are added.

3.1.4 Autoregressive models of order p

ARMA models are some of the most commonly used time series models, popularised by Box & Jenkins (1976). A non-standard use of ARMA models is in describing the hidden states in a state space model. In this section, ARMA models are introduced in generality and then more specifically in relation to DLMS.

A univariate ARMA(p, q) model can be written:

$$Y_t = \mu + \sum_{j=1}^p \phi_j (Y_{t-j} - \mu) + \sum_{j=1}^q \psi_j \epsilon_{t-j} + \epsilon_t, \quad \epsilon_t \sim N(0, \sigma^2), \quad (3.4)$$

for some $p, q \in \mathbb{Z}^*$, where \mathbb{Z}^* is the set of positive integers, including zero. The AR order of the model determines the order of Markov dependence between the current observation at time t and the observations prior to it, denoted by p . The moving average (MA) order of the model determines the order of Markov dependence between the error at each time step and previous observation errors, denoted by q . When using ARMA models, the

process $\{Y_t\}$ is generally assumed to be stationary: having constant mean and variance.¹ Differences can be applied to series that appear to be non-stationary; this yields ARIMA models, where the order, d , of I determines the number of differences required to make the series stationary. An ARMA model is then fitted to the differenced data. In classical time series analysis, the order of p and q are determined by looking at the autocorrelation and partial autocorrelation plots of the data, or via selection criterion such as the BIC or AIC.

ARMA models are special cases of state space models and can be written in the form of DLMS. For the purpose of the thesis, only the DLM representation of an AR(p) is presented, which is equivalent to an ARMA($p, q = 0$) model. The extension to a full ARMA(p, q) model can be found in (Petris *et al.*, 2009, Chapter 3).

Consider a stationary AR(p) model with zero mean: Eq. (3.4), with $\psi_j = 0$ for all $j = 1, \dots, q$ and $\mu = 0$. This is a DLM with $V = 0$ and $W = \text{diag}(\sigma^2, 0, \dots, 0)$, with observation and system matrices

$$F = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}, \quad G = \begin{bmatrix} \phi_1 & 1 & 0 & \dots & 0 \\ \phi_2 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_p & 0 & 0 & \dots & 0 \end{bmatrix}.$$

Defining the state vector $\theta_t = (\theta_{1,t}, \dots, \theta_{p,t})'$, the observation equation results in $y_t = \theta_{1,t}$. Recursively substituting the equations for $\theta_{2,t}, \dots, \theta_{p,t}$ recovers the equation for an AR(p) model. A non-zero, constant mean can be incorporated into the model by adding a local level component to the model with $V = 0$. This can be extended to models with non-stationary means by allowing V to be non-zero. Choosing a non-zero V in a DLM ARMA representation results in a DLM with an ARMA hidden layer.

One benefit of representing an ARMA model as a DLM and using this as a component in a bigger model is that the presence of an ARMA component can be used to explain residual autocorrelation in the data that is not explained by the seasonal or trend component. The overall model for non-stationary data is also more interpretable, thinking in terms of a trend plus seasonal component driving the process, as opposed to differencing data to achieve stationarity before fitting an ARMA(p, q).

Stationary region of an AR(p) model

A time series $\{y_t\}$ is said to be (weakly) stationary if its mean remains constant over time and its autocovariance function $\Gamma_i = \text{Cov}(y_t, y_{t+i})$ depends only on the lag i , for $i = 0, 1, \dots$. Assuming stationarity of a process prevents the predictive variance from

¹The stationary variance will be a complicated function of the other parameters, e.g. $\frac{\sigma^2}{1-\phi^2}$ for AR(1)

growing without bound, which is often a reasonable assumption to make and is particularly useful when forecasting far into the future.

The characteristic polynomial of a process is defined as $\Phi(B) = 1 - \sum_{j=1}^p \phi_j B^j$, where B is the backward shift operator $By_t = y_{t-1}$. When the modulus of the reciprocal roots of the characteristic polynomial is less than one, $|\alpha_i| < 1$, $i = 1, \dots, p$, the process is stationary. For models of order $p = 1$ and $p = 2$, the stationary region is simple to define. For models of order $p \geq 3$, however, the order of the characteristic polynomial increases and the geometry of the stationary region becomes more difficult to define. This makes it increasingly difficult to specify suitable priors for such regions.

A simpler option for restricting the coefficients of an $\text{AR}(p)$ model to the stationary region is to reparametrise them in terms of the model partial autocorrelations. The lag p partial autocorrelation is the correlation at lag p after adjusting for the intervening values, i.e. it is the correlation between the residuals produced when fitting a linear model between the observation at time t and the intervening values from $t - 1, \dots, t - p + 1$, and the residuals when fitting the model between the observation at lag $t - p$ instead. The stationary region of an $\text{AR}(p)$ is defined when the modulus of the partial autocorrelations, ρ_i for $i = 1, \dots, p$, is less than one. The AR coefficients can be expressed in terms of the partial autocorrelations (Congdon, 2007):

$$\phi_i^{(p)} = \begin{cases} \rho_i & \text{if } i = p, \\ \phi_i^{(p-1)} - \rho_p \phi_{p-i}^{(p-1)} & \text{if } i = 1, \dots, p-1. \end{cases} \quad (3.5)$$

Here $\phi_i^{(p)}$ refers to the i th coefficient in an $\text{AR}(p)$ model. This representation allows for the coefficients to be easily restricted to the stationary region by imposing the simple constraint $|\rho_i| < 1$ for $i = 1, \dots, p$.

3.1.5 Dynamic linear regression

State space models offer a natural progression to include explanatory variables in a model. Specifically, incorporating regression components into a DLM is easily done and is looked at here.

Consider the response variable Y_t to a set of non-stochastic explanatory variables $x_{1,t}, \dots, x_{p,t}$ over time, where $t \geq 1$. A dynamic linear regression is defined by the model

$$Y_t = \beta_{1,t} + \beta_{2,t}x_{1,t} + \dots + \beta_{p+1,t}x_{p,t} + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma^2),$$

for i.i.d. errors ε_t . Here, the temporal evolution of the regression coefficients is modelled, introducing a flexible framework for the dependence of the response $\{Y_t\}$ on the explanatory variables to change over time. The DLM is described by the state vec-

tor, $\theta_t = (\beta_{1,t}, \dots, \beta_{p+1,t})'$, a time dependent observation matrix $F_t = (1, x_{1,t}, \dots, x_{p,t})$, that includes the value of the explanatory variables at time t and observation variance $V = \sigma^2$. The evolution matrix G_t can be taken as the identity matrix and $W_t = \text{diag}(w_{1,t}, \dots, w_{p+1,t})$, where $w_{1,t}, \dots, w_{p+1,t}$ are independent variances of the regression coefficients. A standard (static) regression model is recovered by setting $W_t = 0$ for all t .

Working examples on the superposition of the components mentioned in this chapter are found in Chapter 5 and Chapter 6, where such DLMS are applied to the glucose data sets.

3.2 Bayesian inference

When drawing inferences about an event with uncertain outcome or a quantity with uncertain value, the information available is never perfect. A data set is limited to the number of samples collected on a subset of variables available, all of which are subject to measurement error, meaning there is always a margin of uncertainty about a phenomenon that cannot be controlled. Classical (frequentist) statistics draws conclusions purely from the data available, for example via the theory of hypothesis testing, whereas Bayesian statistics incorporates quantifying the uncertainty surrounding the event or quantity into the conclusions drawn using probability. The models in this thesis are fitted in a Bayesian framework, with the exception of the models in Chapter 7 that use frequentist methods. Further references on Bayesian statistics are Berger *et al.* (1994) and Bernardo & Smith (2009).

In Bayesian statistics, the probability describing the initial uncertainty attached to an event or quantity is incorporated into the analysis through prior information. Before collecting data, the researcher formulates beliefs about the unknown event or quantity (these beliefs can differ between researchers and are therefore subjective) and assigns them prior probabilities or prior probability distributions, respectively. These priors are subsequently updated in light of the data using the likelihood, resulting in a posterior probability for the uncertain event or a posterior distribution for the unknown quantity. This process of learning is formulated by calculating conditional probabilities via Bayes' Theorem. Bayes' Theorem for uncertain events is presented in Theorem 3.1.

Theorem 3.1 (Bayes' Theorem). *Given two events, A and B , the law of conditional probability states that the joint probability is $Pr(A \cap B) = Pr(A|B)Pr(B) = Pr(B|A)Pr(A)$. It follows that*

$$Pr(A|B) = \frac{Pr(B|A)Pr(A)}{Pr(B)}.$$

The research presented in this thesis uses Bayesian inference to fit parametric models, primarily HMMs and DLMS as presented so far in this chapter. Generalising Theorem 3.1

in the context of Bayesian parametric modelling, unknown quantities (parameters) are in place of event A and additional experimental information (data) is in place of event B . Formally, let $\underline{\psi} = (\psi_1, \dots, \psi_p)$ be a set of unknown continuous parameters, with prior distribution summarised by the density $\pi(\underline{\psi})$. Information from the data $\underline{y} = (y_1, \dots, y_n)$ is contained in the likelihood $L(\underline{\psi}|\underline{y}) = p(\underline{y}|\underline{\psi})$, which is the joint density (or mass function for discrete \underline{y}) of the data given the unknown quantities $\underline{\psi}$. The data, \underline{y} , are univariate here but can also be multivariate ($y_t = \underline{y}_t$). The posterior density $\pi(\underline{\psi}|\underline{y})$ contains the information from the prior, updated by the likelihood, and is computed using a generalisation of Bayes' Theorem called Bayes' formula:

$$\pi(\underline{\psi}|\underline{y}) = \frac{p(\underline{y}|\underline{\psi})\pi(\underline{\psi})}{p(\underline{y})},$$

where $p(\underline{y}) = \int p(\underline{y}|\underline{\psi})\pi(\underline{\psi})d\underline{\psi}$ is the marginal distribution of \underline{y} . When $\underline{\psi}$ is discrete, the posterior mass function is computed from the prior mass function and the marginal distribution is described by the sum $p(\underline{y}) = \sum_{\underline{\psi}} \pi(\underline{\psi})p(\underline{y}|\underline{\psi})$. In both cases, the marginal distribution is a normalising constant to ensure the posterior density integrates (or sums) to one, so the posterior density can more simply be written

$$\pi(\underline{\psi}|\underline{y}) \propto \pi(\underline{\psi})p(\underline{y}|\underline{\psi}).$$

To compute the posterior density, the form of the prior distribution and of the likelihood must first be decided; the chosen forms are specific to the modelling scenario presented. The form of the likelihood describing the data-generating process is usually dictated by the problem that is being addressed. The choice of prior distribution is often less obvious and more than one distribution can be suitable. Where available, the prior can be chosen so that it has the same functional form as the likelihood, known as conjugacy, which results in a closed form posterior distribution. The use of conjugacy can make analytical and computational inference simpler. An example of a conjugate family, and the main group of conjugate distributions that will be used in this thesis, is the exponential family. All of the distributions within the exponential family have conjugate priors; these distributions include, but are not limited to, the Normal and multivariate Normal, Gamma, Beta and Binomial distributions.

The parameters that specify the prior distribution are known as hyperparameters. Like the form of the prior, the hyperparameter values should be chosen to reflect the researcher's beliefs about the parameter and the level of certainty about those beliefs. As the choice of hyperparameters is subjective, the posterior distribution can look different depending on their values, especially when there are few data available. The posterior is less sensitive to perturbations in the prior hyperparameters when it is paired with

a data-rich likelihood. It is often beneficial to reparametrise the prior distribution by specifying the hyperparameters in terms of more interpretable statistics, such as the mean and variance or coefficient of variation. Examples of this are carried out in Chapter 6 using the Gamma density.

3.2.1 Markov chain Monte Carlo methods

Often in Bayesian inference, the posterior distribution of the parameters of interest is analytically intractable and it is not possible to find closed form solutions for the mean and variance of the posterior distribution or the marginal distribution of the parameters. In such cases, simulation-based methods can be used to approximate the posterior distribution, one of which is the Markov chain Monte Carlo (MCMC) method (Brooks *et al.*, 2011; Gamerman & Lopes, 2006).

MCMC is a widely used technique that involves generating samples from a Markov chain, with analytically tractable transition probabilities, whose stationary distribution is the target posterior distribution (also known as the target distribution). The nature of MCMC means that as long as the number of iterations, N , is large enough, the draws will eventually constitute a sample from the posterior distribution from any supporting starting point. The chain is then said to have converged to its stationary distribution. The period of initial iterations before convergence is achieved is often referred to as burn in and must be discarded. As the samples are autocorrelated (by construction), the information content of N MCMC samples is typically less than the information content of N independent samples. Mixing refers to the rate of posterior exploration. For example, if samples are highly autocorrelated and exploration is slow, the chain is said to be mixing poorly. Diagnostics for convergence and mixing are considered later in this section, after considering two widely used MCMC algorithms.

Gibbs sampling algorithm

Where conjugacy is available, the full conditional distributions for a set of parameters can be easily derived. These are the conditional posteriors for each parameter or parameter block given all the other unknowns. In such cases, the Gibbs sampling algorithm samples directly from the full conditional distributions. This method is described in Algorithm 3.1; for a multidimensional parameter, $\underline{\psi} = (\psi_1, \dots, \psi_p)$, with target density $\pi(\underline{\psi}|\underline{y})$, the Gibbs sampler sets arbitrary initial values from the parameter space $\underline{\psi}^{(0)} = (\psi_1^{(0)}, \dots, \psi_p^{(0)})$ and then updates these values one at a time by drawing $\underline{\psi}^{(i)}|\underline{\psi}^{(i-1)}$, for $i = 1, \dots, N$, from the relevant conditional distributions. This extends to elements that are multidimensional by drawing several values at once from the multivariate conditional distribution, known as block updating.

Algorithm 3.1: Gibbs sampler

1. Initialise $\underline{\psi}^{(0)} = (\psi_1^{(0)}, \dots, \psi_p^{(0)})$;
2. **for** $j = 1, \dots, N$:
3. generate $\psi_1^{(j)}$ from $\pi(\psi_1 | \psi_2 = \psi_2^{(j-1)}, \dots, \psi_p = \psi_p^{(j-1)})$;
4. generate $\psi_2^{(j)}$ from $\pi(\psi_2 | \psi_1 = \psi_1^{(j)}, \psi_3 = \psi_3^{(j-1)}, \dots, \psi_p = \psi_p^{(j-1)})$;
5. \vdots
6. generate $\psi_p^{(j)}$ from $\pi(\psi_k | \psi_1 = \psi_1^{(j)}, \dots, \psi_{p-1} = \psi_{p-1}^{(j)})$
7. **end**

Metropolis-Hastings algorithm

The Metropolis-Hastings (MH) algorithm is a more flexible Markov chain generating method that generates a proposal for the next state of the chain from an arbitrary proposal distribution when the target distribution cannot be sampled from directly. There is then a correction step in which the proposal can be rejected if it moves the chain to a region of lower posterior density. Algorithm 3.2 describes the MH process for a parameter ψ ; a sample update $\tilde{\psi}$ is proposed from an arbitrary transition kernel $q(\psi, \tilde{\psi})$ (the proposal distribution), that specifies the probability of moving from ψ to $\tilde{\psi}$. The proposed value of the chain $\tilde{\psi}$ is accepted with probability

$$A(\psi, \tilde{\psi}) = \min \left\{ 1, \frac{\pi(\tilde{\psi} | \underline{y}) q(\tilde{\psi}, \psi)}{\pi(\psi | \underline{y}) q(\psi, \tilde{\psi})} \right\}. \quad (3.6)$$

If the value is rejected, the chain stays in the current state. This process of sampling results in a Markov chain with stationary distribution equal to the target distribution.

Algorithm 3.2: Metropolis-Hastings algorithm

1. Initialise $\psi^{(0)}$;
2. **for** $j = 1, \dots, N$:
3. generate $\tilde{\psi}$ from $q(\psi^{(j-1)}, \tilde{\psi})$;
4. compute $A = A(\psi^{(j-1)}, \tilde{\psi})$ according to the acceptance probability in (3.6);
5. generate an independent random variable $u \sim U(0, 1)$;
6. **if** $u \leq A$ **then** set $\psi^{(j)} = \tilde{\psi}$ **else** set $\psi^{(j)} = \psi^{(j-1)}$;
7. **end**

The proposal distribution features tuning parameter(s) λ , the values of which are chosen to influence the rate at which proposed parameters are accepted. A proposal with a high rejection rate will result in a ‘sticky’ Markov chain where the state will be constant for many iterations. A high acceptance rate means the chain will move very slowly through the state space due to the high positive autocorrelation, meaning a very large number of

iterations, N , is necessary to achieve convergence and a representative sample from the posterior once convergence is achieved. When the parameter space is large, a satisfactory acceptance rate becomes more difficult to achieve for all of the parameters simultaneously.

In the case where the proposal distribution is symmetric about the current value of the chain, the ratio of the proposal distribution evaluated at the previous and proposed values of the chain cancels in the acceptance probability, so Eq. (3.6) simplifies to

$$A(\psi, \tilde{\psi}) = \min \left\{ 1, \frac{\pi(\tilde{\psi}|\underline{y})}{\pi(\psi|\underline{y})} \right\}.$$

Whilst the normalising constant $p(\underline{y})$ is often unknown when the posterior distribution is intractable, the MH algorithm can still be applied since this constant also cancels in the ratio $\pi(\tilde{\psi}|\underline{y})/\pi(\psi|\underline{y})$.

For a multidimensional parameter space, the MH algorithm can be embedded within Gibbs, sampling parameters one at a time from their full conditional distributions, either directly, if a closed form is available, or indirectly using a MH step. This is known as Metropolis within Gibbs. The Gibbs sampling algorithm can be thought of as a special case of the Metropolis within Gibbs algorithm, where the proposal distribution is specified by the full conditional distribution and the acceptance probability, A , is always one. In general, the Metropolis within Gibbs algorithm is more flexible as it is not constrained to conjugate prior distributions, but is more difficult to tune in higher dimensions. This method, as well as the two algorithms separately, is used throughout the model fitting stages of this thesis.

Diagnosing convergence and assessing mixing

When an MCMC algorithm runs for a sufficient number of iterations, the samples generated approximately resemble samples from the target distribution. The resulting chain must be checked to ensure that it has converged and that it is mixing well. This can easily be done manually, by looking at diagnostic plots.

The first problem of convergence can be diagnosed by looking at the trace plot of the chain. As mentioned when introducing MCMC methods, the ‘burn in’ samples before the chain has converged must be discarded. The remaining samples should appear stationary, exploring the sample space about the mode. If a parameter is multimodal, the trace plot may switch between the different modes. This highlights the importance of running the algorithm for long enough that the whole parameter space is explored.

Mixing is assessed by calculating the ACF of the chain. A poorly mixing chain will have strong correlation between samples, which means the chain will take longer to explore the full parameter space. Sometimes, to reduce computational storage overheads, the MCMC

output is thinned, retaining only every n th iteration. This can also make it easier to judge whether the chain has converged. Plotting the ACF against lag, if the autocorrelations decay to zero quickly as the lag increases, the chain can be said to be mixing well. The output from a chain that has converged but has poor mixing can be valid so long as the algorithm has been run for long enough.

3.2.2 Parameter inference

A state space model has three components: an observed process $\{Y_t\}$, a latent process $\{\theta_t\}$ and the model parameters, $\underline{\psi}$. In addition to the latent process, in a Bayesian setting the model parameters are also treated as unknown random variables. Computational inference can easily be done using the MCMC methods described in Section 3.2.1. To implement such sampling methods efficiently, the full conditional distributions of the unknown parameters must be derived, which primarily involves understanding the dependency structures between the individual parameters and how they interact with the rest of the model they are describing.

The assumptions of the state space model processes (A1 and A2, Section 3.1) are assumed to hold conditionally on the model parameters. Using Eq. (3.1), the joint density of the observations, latent process and model parameters is given by

$$(Y_1, \dots, Y_T, \theta_0, \dots, \theta_T, \underline{\psi}) \sim \pi(\theta_0 | \underline{\psi}) \pi(\underline{\psi}) \prod_{t=1}^T f_t(y_t | \theta_t, \underline{\psi}) g_t(\theta_t | \theta_{t-1}, \underline{\psi}).$$

Here, $\pi(\underline{\psi})$ is the joint prior of the parameters. The model parameters can be further separated into two groups: the parameters associated with the latent process, $\underline{\psi}_{hid}$, and those related to the observation layer, $\underline{\psi}_{obs}$. Taking the parameter dependencies of the observations and the states into consideration, given the model assumptions, the joint distribution is written:

$$(Y_1, \dots, Y_T, \theta_0, \dots, \theta_T, \underline{\psi}) \sim \pi(\theta_0 | \underline{\psi}_{hid}) \pi(\underline{\psi}_{obs}) \pi(\underline{\psi}_{hid}) \prod_{t=1}^T f_t(y_t | \theta_t, \underline{\psi}_{obs}) g_t(\theta_t | \theta_{t-1}, \underline{\psi}_{hid}). \quad (3.7)$$

All of the marginal posterior distributions of the quantities of interest, i.e. the latent process and the unknown parameters, can be derived from Eq. (3.7).

The general results for the static parameter inference specific to HMMs and DLMS are provided in Sections 3.3.1 and 3.4.1.

3.2.3 State inference

One of the main problems of interest in time series analysis is predicting future observations. The latent process of a state space model drives the observations, so learning

about this process is key to forecasting the time series. Inferring the unobserved states and predicting future values of the observed data are addressed through state estimation and forecasting by calculating the relevant conditional distributions. The methods in this section are presented for univariate, continuous-valued states and univariate observations for simplified notation, but are easily extended for multivariate cases. There are also analogous algorithms for HMMs where the states are categorical.

Consider the conditional distribution $\pi(\theta_s|y_{1:t})$ of the underlying states given the observed process at some time s . Notationally, the dependence on the model parameters $\underline{\psi}$ has been dropped in this section. Computing this conditional distribution is known as filtering when $s = t$, smoothing when $s < t$ and forecasting when $s > t$. The recursive nature of state space models and the sequential nature of time series data can be exploited to make writing the filtering, smoothing and forecasting recursions simple. The following demonstrates how this is achieved by marginalising over the joint density of the observations and hidden states in a state space model, Eq. (3.1).

Filtering

Filtering is based on data arriving sequentially, as is usually the case in time series scenarios. As a result, this process is also known as forward filtering, in reference to moving forward in time. Using the data that are available up to time $t - 1$, the one-step-ahead predictions for θ_t and y_t can be calculated, then an update step is performed as a new observation arrives to get $\pi(\theta_t|y_{1:t})$. The online calculations take advantage of the conditional independence properties of state space models and the steps are shown below.

1. Prediction:

- (a) calculate the one-step-ahead predictive density of the states given the data up to time $t - 1$:

$$\begin{aligned}\pi(\theta_t|y_{1:t-1}) &= \int \pi(\theta_t, \theta_{t-1}|y_{1:t-1})d\theta_{t-1}, \\ &= \int g_t(\theta_t|\theta_{t-1}, y_{1:t-1})\pi(\theta_{t-1}|y_{1:t-1})d\theta_{t-1} \\ &= \int g_t(\theta_t|\theta_{t-1})\pi(\theta_{t-1}|y_{1:t-1})d\theta_{t-1};\end{aligned}$$

- (b) calculate the one-step-ahead predictive density of the observations using the predictive density calculated in step 1a:

$$\begin{aligned}\pi(y_t|y_{1:t-1}) &= \int \pi(y_t, \theta_t|y_{1:t-1})d\theta_t, \\ &= \int f_t(y_t|\theta_t, y_{1:t-1})\pi(\theta_t|y_{1:t-1})d\theta_t\end{aligned}$$

$$= \int f_t(y_t|\theta_t)\pi(\theta_t|y_{1:t-1})d\theta_t.$$

2. Updating: calculation of the filtering density is carried out by combining the predictive densities in step 1 via Bayes' Theorem:

$$\begin{aligned}\pi(\theta_t|y_{1:t}) &= \frac{\pi(\theta_t|y_{1:t-1})f_t(y_t|\theta_t, y_{1:t-1})}{\pi(y_t|y_{1:t-1})}, \\ &= \frac{\pi(\theta_t|y_{1:t-1})f_t(y_t|\theta_t)}{\pi(y_t|y_{1:t-1})}.\end{aligned}$$

Forecasting

It follows naturally from filtering, which involves computing the one-step-ahead predictive densities for both the states and the observations, that predictions further ahead in time might be of interest; time series applications are often concerned with the prediction of future values of a phenomenon. The k -step-ahead prediction of the observation or state layer, for $k > 0$, can easily be computed by recursively computing the one-step-ahead predictive densities.

The k -step-ahead forecast for the states builds on the one-step-ahead prediction in step 1a above, taking advantage of the conditional independence properties of the states and observations:

$$\begin{aligned}\pi(\theta_{t+k}|y_{1:t}) &= \int \pi(\theta_{t+k}, \theta_{t+k-1}|y_{1:t})d\theta_{t+k-1}, \\ &= \int g_{t+k}(\theta_{t+k}|\theta_{t+k-1}, y_{1:t})\pi(\theta_{t+k-1}|y_{1:t})d\theta_{t+k-1}, \\ &= \int g_{t+k}(\theta_{t+k}|\theta_{t+k-1})\pi(\theta_{t+k-1}|y_{1:t})d\theta_{t+k-1}.\end{aligned}$$

It follows, in the same pattern that the one-step-ahead observations follow, that the k -step-ahead predictions for the observations are also based on the conditional independence properties and are computed using the k -step-ahead prediction of the states:

$$\begin{aligned}\pi(y_{t+k}|y_{1:t}) &= \int \pi(y_{t+k}, \theta_{t+k}|y_{1:t})d\theta_{t+k}, \\ &= \int f_{t+k}(y_{t+k}|\theta_{t+k}, y_{1:t})\pi(\theta_{t+k}|y_{1:t})d\theta_{t+k}, \\ &= \int f_{t+k}(y_{t+k}|\theta_{t+k})\pi(\theta_{t+k}|y_{1:t})d\theta_{t+k}.\end{aligned}$$

The forecasting recursions show that all of the information about the system up to time t is contained in the conditional density $\pi(\theta_t|y_{1:t})$. This density can be thought of as the initial distribution of a state space model that specifies the joint distribution between the current and future values of the states and the future values of the observations, with

conditional distributions $g_{t+k}(\theta_{t+k}|\theta_{t+k-1})$ and $f_{t+k}(y_{t+k}|\theta_{t+k})$. The initial distribution determines the future evolution of the model.

Smoothing

When a time series has been observed for some $t = 1, \dots, T$, it can be of interest to retrospectively learn about the state of the underlying system that drove such observations. This practice is known as smoothing. Like filtering and forecasting, smoothing is performed recursively, but moving backwards from time T . The smoothing recursion can be broken down into two steps: calculating the (backward) transition probabilities $\pi(\theta_t|\theta_{t+1}, y_{1:T})$ and then calculating the smoothing distribution $\pi(\theta_t|y_{1:T})$.

1. The backward transition probabilities are calculated using the conditional independence properties of state space models and Bayes' formula:

$$\begin{aligned}\pi(\theta_t|\theta_{t+1}, y_{1:T}) &= \pi(\theta_t|\theta_{t+1}, y_{1:t}), \\ &= \frac{\pi(\theta_t|y_{1:t})g_{t+1}(\theta_{t+1}|\theta_t, y_{1:t})}{\pi(\theta_{t+1}|y_{1:t})}, \\ &= \frac{\pi(\theta_t|y_{1:t})g_{t+1}(\theta_{t+1}|\theta_t)}{\pi(\theta_{t+1}|y_{1:t})}.\end{aligned}$$

2. The conditional distribution of θ_t given $y_{1:T}$ is computed as a marginal distribution of $\pi(\theta_t, \theta_{t+1}|y_{1:T})$, substituting in the transition probabilities in step 1:

$$\begin{aligned}\pi(\theta_t|y_{1:T}) &= \int \pi(\theta_t, \theta_{t+1}|y_{1:T})d\theta_{t+1}, \\ &= \int \pi(\theta_{t+1}|y_{1:T})\pi(\theta_t|\theta_{t+1}, y_{1:T})d\theta_{t+1}, \\ &= \int \pi(\theta_{t+1}|y_{1:T})\frac{\pi(\theta_t|y_{1:t})g_{t+1}(\theta_{t+1}|\theta_t)}{\pi(\theta_{t+1}|y_{1:t})}d\theta_{t+1}, \\ &= \pi(\theta_t|y_{1:t})\int \pi(\theta_{t+1}|y_{1:T})\frac{g_{t+1}(\theta_{t+1}|\theta_t)}{\pi(\theta_{t+1}|y_{1:t})}d\theta_{t+1}.\end{aligned}$$

Whilst writing the filtering, smoothing and forecasting recursions is simple, computation of the relevant conditional distributions can be very challenging. Sections 3.3.2 and 3.4.2 show how these problems simplify in the cases of HMMs and DLMS.

3.2.4 Inference for missing data

It is common for missing observations to occur in the data collection process of real-world applications. Such missed observations can occur due to the observation not existing, such as when a process does not occur at a particular time point, or because of a recording error, such as when a sensor is out of range or if the battery dies and the observation

cannot be noted. The missing observations are usually recorded in a time series as **NA**. Assuming these data are missing at random, missing observations can be easily handled by state space models.

Consider the case where the univariate observation $y_t = \text{NA}$. In this case, the filtering density $\pi(\theta_t|y_{1:t}) = \pi(\theta_t|y_{1:t-1})$, since no information is carried by the current observation. The filtering density is therefore simply equal to the one-step-ahead predictive density at time $t - 1$. This is equivalent to constructing the joint distribution between the missing data, parameters and other unknowns and integrating out the missing observations.

As missing observations are accounted for in the filtering recursion, the smoothing recursion remains the same. This is because the smoothing recursion only depends on the data through the filtering and one-step-ahead predictions.

3.2.5 Data augmentation

When fitting state space models, it is often of interest to reconstruct the latent process up to time T and to learn the model parameters. It can be difficult or impossible (for some state space models) to sample directly from the posterior parameter distribution $\pi(\underline{\psi}|y_{1:T})$. Data augmentation involves expanding the state space of the MCMC sampler to include the states $\theta_{0:T}$ so that the resulting target density is

$$\pi(\theta_{0:T}, \underline{\psi}|y_{1:T}) = \pi(\theta_{0:T}|\underline{\psi}, y_{1:T})\pi(\underline{\psi}|y_{1:T}),$$

which can be sampled easily using the framework of Gibbs sampling, due to the conditional dependency assumptions of state space models (Albert & Chib, 1993; Frühwirth-Schnatter, 1994b).

The Gibbs sampler alternates drawing $\theta_{0:T}$ from $\pi(\theta_{0:T}|\underline{\psi}, y_{1:T})$ and drawing $\underline{\psi}$ from $\pi(\underline{\psi}|\theta_{0:T}, y_{1:T})$. Simulating the parameter vector from the relevant full conditional distributions is specific to the problem, but general algorithms to simulate the state vector are available. The smoothing recursion in Section 3.2.3 provides an algorithm for sampling from the posterior of the state vectors given the data up to time T and the parameters, $\underline{\psi}$. The density of the states at time T , $\pi(\theta_T|\underline{\psi}, y_{1:T})$ is equivalent to the filtering density at time T . The Gibbs sampler therefore requires running the filtering recursion up to time T in a forward sweep so a value for θ_T can be generated, followed by implementing the smoothing recursion to sample the states at times $T - 1, \dots, 0$. This method of sampling is known as forward filtering backward sampling (Carter & Kohn, 1994; Frühwirth-Schnatter, 1994a) and its implementation specific to HMMs and DLMS is described in Sections 3.3.2 and 3.4.2, respectively. Forward filtering backward sampling allows the states to be updated in a single block, as opposed to one at a time which can lead to poor convergence.

3.3 Inference for HMMs

Inference for HMMs typically involves learning about the sequence of hidden states, $\{Z_t\}$, that drive an observed process, $\{Y_t\}$, and learning about the model parameters, such as the transition probabilities and the parameters of the within-state distributions. The HMMs fitted in Chapter 4 are an example of such inference, where the most likely sequence of underlying zones of activity that explain the activity data are learned, as well as the distributions of the model parameters. This section details the methods used to fit such HMMs, culminating in an algorithm to implement the methods.

3.3.1 Parameter inference

The complete data posterior for a HMM with discrete states, \underline{z} , is written

$$\pi(\underline{\psi}|\underline{z}, \underline{y}) \propto p(\underline{y}, \underline{z}|\underline{\psi})\pi(\underline{\psi}).$$

The joint prior over the parameter vector factorises into the product of the prior for $\underline{\psi}_{obs}$ and the prior for $\underline{\psi}_{hid}$. Because of the structure of the joint density in Eq. (3.7), when the prior can be factorised in this way, the posterior can too, so the complete data posterior is the product of two components:

$$\pi(\underline{\psi}|\underline{z}, \underline{y}) = \pi(\underline{\psi}_{obs}|\underline{z}, \underline{y})\pi(\underline{\psi}_{hid}|\underline{z}),$$

with

$$\begin{aligned} \pi(\underline{\psi}_{obs}|\underline{z}, \underline{y}) &\propto p(\underline{y}|\underline{z}, \underline{\psi}_{obs})\pi(\underline{\psi}_{obs}), \\ \pi(\underline{\psi}_{hid}|\underline{z}) &\propto p(\underline{z}|\underline{\psi}_{hid})\pi(\underline{\psi}_{hid}). \end{aligned}$$

The parameters associated with the observed and hidden processes of a HMM with K hidden states are $\underline{\psi}_{obs} = (\underline{\psi}_{obs,1}, \dots, \underline{\psi}_{obs,K})$ and $\underline{\psi}_{hid} = (\underline{\nu}, \underline{\Pi})$, where $\underline{\psi}_{obs,k}$ are the within-state parameters for state k , $\underline{\nu}$ is the initial distribution over the states and $\underline{\Pi}$ is the transition matrix. When learning about the parameters in one state $\underline{\psi}_{obs,k}$ would not change the beliefs about any of the others, it is fair to assume that $\underline{\psi}_{obs,1}, \dots, \underline{\psi}_{obs,K}$ are independent a priori. Likewise for the hidden process parameters, for example, the rows of the transition matrix, $\underline{\pi}_k$, are regarded as independent a priori. Altogether, the joint prior factorises into:

$$\begin{aligned} \pi(\underline{\psi}) &= \pi(\underline{\psi}_{hid})\pi(\underline{\psi}_{obs}), \\ &= \pi(\underline{\nu}) \prod_{k=1}^K \pi(\underline{\pi}_k)\pi(\underline{\psi}_{obs,k}). \end{aligned}$$

The density of the complete data likelihood $p(\underline{y}|\underline{z}, \underline{\psi}_{obs})$ depends on the chosen within-state distributions, but under assumption A2,

$$\begin{aligned} p(\underline{y}|\underline{z}, \underline{\psi}_{obs}) &= \prod_{t=1}^T p(y_t|z_t, \underline{\psi}_{obs}), \\ &= \prod_{k=1}^K \prod_{t:z_t=k} p(y_t|\underline{\psi}_{obs,k}). \end{aligned} \quad (3.8)$$

Under assumption A1, the probability mass function $p(\underline{z}|\underline{\psi}_{hid})$ is simply the product of the relevant probabilities in the initial distribution and transition matrix for each time point:

$$\begin{aligned} p(\underline{z}|\underline{\psi}_{hid}) &= Pr(Z_0 = z_0|\underline{\nu}) \prod_{t=1}^T Pr(Z_t = z_t|Z_{t-1} = z_{t-1}, \underline{\Pi}), \\ &= \nu_{z_0} \prod_{t=1}^T \pi_{z_t, z_{t-1}}. \end{aligned}$$

The overall form of $\pi(\underline{\psi}_{obs}|\underline{z}, \underline{y})$ is problem specific and is therefore discussed alongside the models in Chapter 4. The form of $\pi(\underline{\psi}_{hid}|\underline{z})$, however, can be determined given the choice of a prior for the hidden state parameters. A Dirichlet prior is chosen for the rows of the transition matrix and the initial distribution.

The Dirichlet distribution is parametrised by a vector $\underline{\alpha}$ of positive real values. Draws from the distribution are vectors of the same length as $\underline{\alpha}$ that sum to one. This makes the Dirichlet distribution a suitable prior for probabilities like those in the initial distribution and transition matrix. The Dirichlet is also a conjugate prior to the multinomial form of the likelihood $p(\underline{z}|\underline{\psi}_{hid})$, so the target distribution can be directly sampled from in the corresponding MCMC scheme step. The prior distributions for $\underline{\nu}$ and $\underline{\pi}_k$ are written

$$\begin{aligned} \underline{\nu} &= (Pr(Z_0 = 1|\underline{\nu}), \dots, Pr(Z_0 = K|\underline{\nu})) \sim D(\underline{\gamma} = (\gamma_1, \dots, \gamma_K)), \\ \underline{\pi}_k &= (Pr(Z_t = 1|Z_{t-1} = k), \dots, Pr(Z_t = K|Z_{t-1} = k)) \sim D(\underline{\alpha}_k = (\alpha_{k,1}, \dots, \alpha_{k,K})). \end{aligned}$$

From the above prior specification, the full conditional distributions of the hidden process parameters can be derived. Given the independence properties already discussed, $\underline{\nu}$ and $\underline{\Pi}$ can be considered separately. For the transition matrix:

$$\begin{aligned} \pi(\underline{\Pi}|\underline{z}) &\propto \prod_{k=1}^K \prod_{l=1}^K \pi_{k,l}^{\alpha_{k,l}-1} \times \prod_{k=1}^K \prod_{l=1}^K \pi_{k,l}^{n_{k,l}(\underline{z})}, \\ &\propto \prod_{k=1}^K \prod_{l=1}^K \pi_{k,l}^{\alpha_{k,l}+n_{k,l}(\underline{z})-1}, \end{aligned}$$

where $\underline{n}_k(\underline{z}) = (n_{k,1}(\underline{z}), \dots, n_{k,K}(\underline{z}))$ is a vector of the number transitions from state k to state l . So

$$\underline{\pi}_k | \underline{z} \sim D(\underline{\alpha}_k + \underline{n}_k(\underline{z})). \quad (3.9)$$

Similarly for $\underline{\nu} | \underline{z}$,

$$\underline{\nu} | \underline{z} \sim D(\underline{\gamma} + \underline{m}(\underline{z})), \quad (3.10)$$

where $\underline{m}(\underline{z}) = (m_1(\underline{z}), \dots, m_K(\underline{z}))$ is a vector of the number of occurrences of each state at time $t = 0$.

3.3.2 State inference for HMMs

As discussed in Section 3.2.5, the most efficient way to simulate values from the latent process of a state space model is by implementing a forward filtering backward sampling scheme. The computational inference for $\pi(\underline{z} | \underline{y}, \underline{\psi})$ in a HMM requires working with the probability mass functions of the conditional densities derived for filtering and smoothing in Section 3.2.3 to reflect the discrete state space. The computations are easier to calculate since the recursions no longer contain potentially computationally intractable integrals.

Sampling from the posterior $\pi(\underline{z} | \underline{y}, \underline{\psi})$ consists of two main components: recursively computing the filtered probabilities in a forward sweep over the hidden states and the data and a backward sweep that computes the smoothed probabilities and samples the states from $t = T, \dots, 0$.

The forward filter over a discrete state space is described in Algorithm 3.3. Taking advantage of the assumptions for state space models, and therefore HMMs, the conditional probabilities in Algorithm 3.3 can be written in terms of the initial probabilities and transition probabilities:

$$\begin{aligned} Pr(Z_0 = k | \underline{\psi}) &= \nu_k, \\ Pr(Z_t = \ell | Z_{t-1} = k, \underline{y}_{1:t-1}, \underline{\psi}) &= Pr(Z_t = \ell | Z_{t-1} = k, \Pi) = \pi_{k,\ell}. \end{aligned}$$

Furthermore, the conditional densities of the observations given the states are simply:

$$p(y_t | Z_t = \ell, \underline{\psi}) = p(y_t | \underline{\psi}_{obs,\ell}),$$

where $\underline{\psi}_{obs,\ell}$ are the parameters of the within-state distribution for state ℓ .

The filtered probabilities at time T in Algorithm 3.3 provide the first step of the backward recursion, from which $Z_T = \tilde{z}_T$ is simulated. The full forward backward sampling algorithm over a discrete state space is shown in Algorithm 3.4. As with the forward recursion, the conditional probabilities in the backward recursion also simplify based on

Algorithm 3.3: Forward filter over a discrete state space

1. Initialise the forward recursion at $t = 1$:

$$Pr(Z_1 = \ell | \underline{\psi}) = \sum_{k=1}^K Pr(Z_1 = \ell | Z_0 = k, \underline{\psi}) Pr(Z_0 = k | \underline{\psi})$$

and so

$$Pr(Z_1 = \ell | y_1, \underline{\psi}) = \frac{p(y_1 | Z_1 = \ell, \underline{\psi}) Pr(Z_1 = \ell | \underline{\psi})}{p(y_1 | \underline{\psi})},$$

where

$$p(y_1 | \underline{\psi}) = \sum_{k=1}^K p(y_1 | Z_1 = k, \underline{\psi}) Pr(Z_1 = k | \underline{\psi});$$

2. **for** $t = 2, \dots, T$:

3. compute the one-step-ahead predictive probabilities for $\ell = 1, \dots, K$:

$$Pr(Z_t = \ell | y_{1:t-1}, \underline{\psi}) = \sum_{k=1}^K Pr(Z_t = \ell | Z_{t-1} = k, y_{1:t-1}, \underline{\psi}) Pr(Z_{t-1} = k | y_{1:t-1}, \underline{\psi});$$

4. compute the filtered probabilities:

$$Pr(Z_t = \ell | y_{1:t}, \underline{\psi}) = \frac{p(y_t | Z_t = \ell, \underline{\psi}) Pr(Z_t = \ell | y_{1:t-1}, \underline{\psi})}{p(y_t | y_{1:t-1}, \underline{\psi})},$$

where

$$p(y_t | y_{1:t-1}, \underline{\psi}) = \sum_{k=1}^K p(y_t | Z_t = k, y_{1:t-1}, \underline{\psi}) Pr(Z_t = k | y_{1:t-1}, \underline{\psi}).$$

5. **end**

assumptions A1 and A2:

$$Pr(Z_{t+1} = \tilde{z}_{t+1} | Z_t = k, y_{1:t}, \underline{\psi}) = Pr(Z_{t+1} = \tilde{z}_{t+1} | Z_t = k, \Pi) = \pi_{k, \tilde{z}_{t+1}}.$$

The block Gibbs sampler for simulating values from the joint posterior distribution $\pi(\underline{z}, \underline{\psi} | \underline{y})$ using the method of data augmentation in Section 3.2.5 is outlined in Algorithm 3.5.

3.3.3 Numerical stability in a HMM

Computational underflow occurs when a computer evaluates a non-zero result as zero, where the precise value of the result is too small for the computer to present. This can

Algorithm 3.4: Forward backward algorithm

1. Run Algorithm 3.3 to compute the filtered probabilities $Pr(Z_t = k|y_{1:t}, \underline{\psi})$ for $k = 1, \dots, K$ and for times $t = 0, \dots, T$;
2. Simulate a value for Z_T , say \tilde{z}_T , from $Pr(Z_T = k|y_{1:T}, \underline{\psi})$, where $k = 1, \dots, K$;
3. **for** $t = T - 1, \dots, 0$:
4. compute the backward probabilities for $k = 1, \dots, K$:

$$Pr(Z_t = k|Z_{t+1} = \tilde{z}_{t+1}, y_{1:t}, \underline{\psi}) = \frac{Pr(Z_{t+1} = \tilde{z}_{t+1}|Z_t = k, y_{1:t}, \underline{\psi})Pr(Z_t = k|y_{1:t}, \underline{\psi})}{\sum_{\ell=1}^K Pr(Z_{t+1} = \tilde{z}_{t+1}|Z_t = \ell, y_{1:t}, \underline{\psi})Pr(Z_t = \ell|y_{1:t}, \underline{\psi})},$$
 and simulate a value for Z_t from the distribution defined by these probabilities.
5. **end**

Algorithm 3.5: Block Gibbs sampler for a HMM

1. Initialise $\underline{\psi}_{obs,k}^{(0)}$, $\underline{\nu}^{(0)}$ and $\underline{\pi}_k^{(0)}$, for all k ;
2. **for** $i = 1, \dots, N$:
3. run the forward backward algorithm (Algorithm 3.4) to sample the hidden states $\underline{z}^{(i)}$;
4. draw $\underline{\pi}_k^{(i)}$ from (3.9) for $k = 1, \dots, K$;
5. draw $\underline{\nu}^{(i)}$ from (3.10);
6. draw $\underline{\psi}_{obs}^{(i)}$ for $k = 1, \dots, K$ from the relevant full conditional distributions given $\underline{z}^{(i)}$;
7. **end**

cause further issues when such a value is used later on in calculations, for example as a denominator in a fraction, which would result in **Inf**. Underflow is even more likely when taking the product of small numbers, for example probabilities. It is therefore important to prevent underflow wherever possible.

One simple method for preventing underflow when working with probabilities is to take the natural logarithm of the calculation (assuming that all of the probabilities are greater than zero), then exponentiate the final result once all of the computations have been carried out. Taking logs of a product reduces the calculation to the sum of the logged values, which are much easier to store. A challenge arises, however, when computational underflow is the result of taking the sum over small numbers, for example, the filtered probabilities calculated in Algorithm 3.3. When the distributions of the hidden states of a time series are distinct, with little support overlap, the density of each observation given each state, $p(y_t|Z_t = \ell, \underline{\psi})$, can evaluate to very small probabilities that can lead to underflow. To overcome this, the log-sum-exp trick can be implemented.

Log-sum-exp trick

The log-sum-exp trick is used to prevent computational underflow and overflow (where the result of a calculation is too large for computer memory), when summing over the product of very small (or large) numbers. Let $\underline{p} = (p_1, \dots, p_K)$ be a vector of products of probabilities, where at least one of the elements is small enough to cause underflow, and suppose the quantity of interest is $\sum_{k=1}^K p_k$. Taking the log of the sum as it is does not solve the problem, since the terms within the logged sum cannot be separated. Instead, write the log of the sum as:

$$\log \sum_{k=1}^K p_k = \log \sum_{k=1}^K \exp(\log p_k).$$

Further, letting $m = \max_k \{\log p_k\}$, the log of the sum can be written:

$$\begin{aligned} \log \sum_{k=1}^K p_k &= \log \sum_{k=1}^K \exp(\log p_k - m + m), \\ &= \log \left\{ \exp(m) \sum_{k=1}^K \exp(\log p_k - m) \right\}, \\ &= m + \log \sum_{k=1}^K \exp(\log p_k - m), \end{aligned}$$

which is known as the log-sum-exp trick. Subtracting m within the exponential in the sum allows the log probability, $\log p_k$, to be used in the calculation, preventing underflow (or overflow) as desired. Additionally, subtracting m means the largest element being exponentiated is zero, which prevents the sum from blowing up when dealing with very large numbers that cause computational overflow.

3.4 Inference for DLMS

In order to successfully model a process using a DLM, data must be used to learn about the model parameters. Ultimately this still involves learning about the latent process that drives the observed sequence, but the Gaussian state space defined for DLMS simplifies such inference.

3.4.1 Parameter inference

The static parameters in a DLM are $\underline{\psi} = (m_0, C_0, V_t, W_t)$. For the modelling scenarios in this thesis, V_t and W_t are time-homogeneous, so the time-varying notation is therefore dropped from here on, setting $V_t = V$ and $W_t = W$. Typically, the parameters of the

initial distribution of the states, m_0 and C_0 , are assumed to be known; as the system of a DLM is dynamic, the new information received at each time point informs the distribution of the states θ_t and the initial distribution soon becomes unimportant. The system and observation matrices, G_t and F_t , are determined by the components being used in the model, for example a local level or seasonal component, so do not feature in the parameter vector. Inference for $\pi(\underline{\psi}|\underline{\theta}, \underline{y})$ therefore only concerns the observation variance V and the system covariance matrix W , which is assumed to be diagonal.

Specify the unknown variances in terms of the precisions, $V = \psi_y^{-1}$ and $W = \text{diag}(\psi_1^{-1}, \dots, \psi_p^{-1})$. A simple prior to use for the parameters of the variances is a d -inverse-Gamma prior, which assumes $\psi_y, \psi_1, \dots, \psi_p$ follow $d = p + 1$ independent Gamma distributions:

$$\begin{aligned}\psi_y &\sim Ga(a_y, b_y), \\ \psi_i &\sim Ga(a_i, b_i), \quad \text{for } i = 1, \dots, p.\end{aligned}$$

The result of this prior assumption is that the posteriors for the precisions also follow inverse-Gamma distributions.

The form of the complete data likelihood is Gaussian, since the underlying states and errors are Gaussian:

$$p(\underline{y}|\underline{\theta}, \psi_y) = \prod_{t=1}^T f_t(y_t|\theta_t, \psi_y),$$

and

$$y_t|\theta_t, \psi_y \sim N(F_t\theta_t, \psi_y^{-1}).$$

The posterior density for V can therefore easily be derived:

$$\begin{aligned}\pi(\psi_y|\underline{y}, \underline{\theta}) &\propto \pi(\psi_y) \prod_{t=1}^T f_t(y_t|\theta_t, \psi_y), \\ &\propto \psi_y^{a_y-1} \exp(-b_y\psi_y) \prod_{t=1}^T \frac{1}{\sqrt{2\pi\psi_y^{-1}}} \exp\left[-\frac{\psi_y}{2}(y_t - F_t\theta_t)^2\right], \\ &\propto \psi_y^{\frac{T}{2}+a_y-1} \exp\left[-\psi_y\left\{\frac{1}{2}\sum_{t=1}^T (y_t - F_t\theta_t)^2 + b_y\right\}\right],\end{aligned}$$

which is

$$\psi_y^{-1}|\underline{y}, \underline{\theta} \sim I-Ga\left(a_y + \frac{T}{2}, b_y + \frac{1}{2}\sum_{t=1}^T (y_t - F_t\theta_t)^2\right). \quad (3.11)$$

The conditional density of $\pi(\underline{\theta}|W)$ is also Gaussian:

$$\theta_t|\theta_{t-1}, W \sim N_p(G_t\theta_{t-1}, W).$$

The conjugate posterior density for the elements of W are derived in the same way. As W is diagonal and the ψ_i are independent a priori, the full conditional distributions for each can be considered separately:

$$\begin{aligned} \pi(\psi_i|\underline{\theta}) &\propto \pi(\psi_i) \prod_{t=1}^T g_t(\theta_t|\theta_{t-1}, \psi_i), \\ &\propto \psi_i^{a_i-1} \exp(-b_i\psi_i) \prod_{t=1}^T \frac{1}{\sqrt{2\pi\psi_i^{-1}}} \exp\left[-\frac{\psi_i}{2}(\theta_{t,i} - (G_t\theta_{t-1})_i)^2\right], \\ &\propto \psi_i^{\frac{T}{2}+a_i-1} \exp\left[-\psi_i\left\{\frac{1}{2}\sum_{t=1}^T(\theta_{t,i} - (G_t\theta_{t-1})_i)^2 + b_i\right\}\right], \end{aligned}$$

which is

$$\psi_i^{-1}|\underline{\theta} \sim I-Ga\left(a_i + \frac{T}{2}, b_i + \frac{1}{2}\sum_{t=1}^T(\theta_{t,i} - (G_t\theta_{t-1})_i)^2\right). \quad (3.12)$$

The independence of the parameters a posteriori means they can be updated in a single block in a Gibbs sampler.

3.4.2 State inference for DLMS

The initial state-vector and the errors in the system and observation equations in a DLM are specified by Gaussian distributions. As a result, the joint distribution of the states and observations is also Gaussian. This means the integrals in the filtering, smoothing and forecasting recursions described in Section 3.2.3 can be evaluated in closed form and the algorithms reduce to the simpler task of computing the mean and variance for a series of conditional Gaussian distributions.

The Kalman filter

The Kalman filter (Kalman, 1960) is a filtering recursion that lends itself specifically to DLMS. If $\theta_{t-1}|y_{1:t-1} \sim N(m_{t-1}, C_{t-1})$, the prediction and updating steps at time t are as follows.

1. Prediction:

- (a) the mean a_t and variance R_t of the one-step-ahead predictive density of the states given the data up to time $t - 1$ are:

$$\begin{aligned} a_t &= \mathbb{E}(\theta_t|y_{1:t-1}), \\ &= \mathbb{E}\{\mathbb{E}(\theta_t|\theta_{t-1}, y_{1:t-1})\}, \\ &= \mathbb{E}(G_t\theta_{t-1}|y_{1:t-1}), \end{aligned}$$

$$= G_t m_{t-1}$$

and

$$\begin{aligned} R_t &= \text{Var}(\theta_t | y_{1:t-1}), \\ &= \text{E}\{\text{Var}(\theta_t | \theta_{t-1}, y_{1:t-1})\} + \text{Var}\{\text{E}(\theta_t | \theta_{t-1}, y_{1:t-1})\}, \\ &= \text{E}(W | y_{1:t-1}) + \text{Var}(G_t \theta_{t-1} | y_{1:t-1}), \\ &= W + G_t C_{t-1} G_t'. \end{aligned}$$

(b) the mean f_t and variance Q_t of the one-step-ahead forecast of the observations are:

$$\begin{aligned} f_t &= \text{E}(Y_t | y_{1:t-1}), \\ &= \text{E}\{\text{E}(Y_t | \theta_t, y_{1:t-1})\}, \\ &= \text{E}(F_t \theta_t | y_{1:t-1}), \\ &= F_t a_t \end{aligned}$$

and

$$\begin{aligned} Q_t &= \text{Var}(Y_t | y_{1:t-1}), \\ &= \text{E}\{\text{Var}(Y_t | \theta_t, y_{1:t-1})\} + \text{Var}\{\text{E}(Y_t | \theta_t, y_{1:t-1})\}, \\ &= \text{E}(V | y_{1:t-1}) + \text{Var}(F_t \theta_t | y_{1:t-1}), \\ &= V + F_t R_t F_t'. \end{aligned}$$

2. Updating: after observing $Y_t = y_t$, the filtering density is calculated via Bayes' Theorem, where $\theta_t | y_{1:t-1} \sim N(a_t, R_t)$ is the 'prior' and $y_t | \theta_t \sim N(f_t, Q_t)$ is the 'likelihood';

$$\begin{aligned} \pi(\theta_t | y_{1:t}) &\propto \pi(\theta_t | y_{1:t-1}) f_t(y_t | \theta_t), \\ &\propto \exp \left[-\frac{1}{2} (y_t - F_t \theta_t)' V^{-1} (y_t - F_t \theta_t) \right] \times \\ &\quad \exp \left[-\frac{1}{2} (\theta_t - a_t)' R_t^{-1} (\theta_t - a_t) \right], \\ &\propto \exp \left[-\frac{1}{2} \left\{ \theta_t' (F_t' V^{-1} F_t + R_t^{-1}) \theta_t - 2 \theta_t' (F_t' V^{-1} y_t + R_t^{-1} a_t) \right\} \right], \end{aligned}$$

which is a Gaussian density with parameters $m_t = C_t (F_t' V^{-1} y_t + R_t^{-1} a_t)$ and $C_t = (F_t' V^{-1} F_t + R_t^{-1})^{-1}$.

The expressions for C_t and m_t in the updating step involve inverting potentially large

matrices, which is computationally expensive and can cause numerical instability. To avoid this, C_t is rearranged using the Woodbury matrix identity (Woodbury, 1950),

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}.$$

Rearranging C_t using this identity gives

$$\begin{aligned} C_t &= R_t - R_t F_t' (F_t R_t F_t' + V)^{-1} F_t R_t, \\ &= R_t - R_t F_t' Q_t^{-1} F_t R_t \end{aligned}$$

and substituting this into the equation for m_t and simplifying gives

$$\begin{aligned} m_t &= a_t + R_t F_t' (F_t R_t F_t' + V)^{-1} (y_t - F_t a_t), \\ &= a_t + R_t F_t' Q_t^{-1} e_t. \end{aligned}$$

These computations no longer involve matrix inversion as Q_t is a scalar. The innovations e_t are the forecasting errors, defined by $e_t = y_t - f_t = y_t - F_t a_t$.

As explained in Section 3.2.4, in the case of a missing observation, or missing observations, a new observation is not available to perform the updating step and so $\pi(\theta_t | y_{1:t}) = \pi(\theta_t | y_{1:t-1})$. This is equivalent to setting $m_t = a_t$ and $C_t = R_t$, which demonstrates how the uncertainty about the state increases until a new observation is available.

Forecasting

Forecasting in a state space model is an extension of the one-step-ahead predictions that are calculated in the filtering recursion. It follows that the forecasting recursion for DLMS is an extension of the Kalman filter, which again involves computing the means and variances of the respective Gaussian distributions.

Let $a_t(k) = E(\theta_{t+k} | y_{1:t})$, $R_t(k) = \text{Var}(\theta_{t+k} | y_{1:t})$, $f_t(k) = E(Y_{t+k} | y_{1:t})$ and $Q_t(k) = \text{Var}(Y_{t+k} | y_{1:t})$. At time t , $a_t(0) = m_t$ and $R_t(0) = C_t$, then for $k \geq 1$, the mean and variance of the k -step-ahead state forecast given the observations up to time t are:

$$\begin{aligned} a_t(k) &= E(\theta_{t+k} | y_{1:t}), \\ &= E\{E(\theta_{t+k} | y_{1:t}, \theta_{t+k-1})\}, \\ &= E(G_{t+k} \theta_{t+k-1} | y_{1:t}), \\ &= G_{t+k} a_t(k-1) \end{aligned}$$

and

$$\begin{aligned}
 R_t(k) &= \text{Var}(\theta_{t+k}|y_{1:t}), \\
 &= \text{Var}\{(\mathbf{E}(\theta_{t+k}|y_{1:t}, \theta_{t+k-1}))\} + \mathbf{E}\{\text{Var}(\theta_{t+k}|y_{1:t}, \theta_{t+k-1})\}, \\
 &= G_{t+k}R_t(k-1)G_{t+k}' + W.
 \end{aligned}$$

The mean and variance of the k -step-ahead observation forecast given the observations up to time t are:

$$\begin{aligned}
 f_t(k) &= \mathbf{E}(Y_{t+k}|y_{1:t}), \\
 &= \mathbf{E}\{\mathbf{E}(Y_{t+k}|y_{1:t}, \theta_{t+k})\}, \\
 &= \mathbf{E}(F_{t+k}\theta_{t+k}|y_{1:t}), \\
 &= F_{t+k}a_t(k)
 \end{aligned}$$

and

$$\begin{aligned}
 Q_t(k) &= \text{Var}(Y_{t+k}|y_{1:t}), \\
 &= \text{Var}\{(\mathbf{E}(Y_{t+k}|y_{1:t}, \theta_{t+k}))\} + \mathbf{E}\{\text{Var}(Y_{t+k}|y_{1:t}, \theta_{t+k})\}, \\
 &= F_{t+k}R_t(k)F_{t+k}' + V.
 \end{aligned}$$

As the above equations hold for $k = 1$ in the prediction step of the Kalman filter, they also hold for $k > 1$ by induction.

The Kalman smoother

Backward smoothing allows inference about the underlying states of a system once the data collection process is complete. As the observations are known, smoothing incurs less uncertainty in the state distribution than filtering. As with the Kalman filter, the Kalman smoother requires only the computation of the mean and variance of the corresponding Gaussian distribution of the smoothed states.

Letting $\theta_{t+1}|y_{1:T} \sim N(s_{t+1}, S_{t+1})$ the Kalman smoothing recursion is as follows:

1. The backward transition probabilities are calculated using Bayes' Theorem and the rules of conditional independence, where the 'prior' is $\theta_t|y_{1:t} \sim N(m_t, C_t)$ and the 'likelihood' is $\theta_{t+1}|\theta_t \sim N(G_{t+1}\theta_t, W_{t+1})$:

$$\begin{aligned}
 \pi(\theta_t|\theta_{t+1}, y_{1:T}) &\equiv \pi(\theta_t|\theta_{t+1}, y_{1:t}) \propto \pi(\theta_t|y_{1:t})g_{t+1}(\theta_{t+1}|\theta_t), \\
 &\propto \exp\left[-\frac{1}{2}(\theta_t - m_t)'C_t^{-1}(\theta_t - m_t)\right] \times
 \end{aligned}$$

$$\begin{aligned} & \exp \left[-\frac{1}{2}(\theta_{t+1} - G_{t+1}\theta_t)'W^{-1}(\theta_{t+1} - G_{t+1}\theta_t) \right], \\ \propto & \exp \left[-\frac{1}{2} \left\{ \theta_t'(C_t^{-1} - G_{t+1}'W^{-1}G_{t+1})\theta_t \right. \right. \\ & \left. \left. - 2\theta_t'(C_t^{-1}m_t + G_{t+1}'W^{-1}\theta_{t+1}) \right\} \right], \end{aligned}$$

which is a Gaussian distribution with mean $(C_t^{-1} - G_{t+1}'W^{-1}G_{t+1})^{-1}(C_t^{-1}m_t + G_{t+1}'W^{-1}\theta_{t+1})$ and variance $(C_t^{-1} - G_{t+1}'W^{-1}G_{t+1})^{-1}$. These can be further simplified to avoid large matrix inversion using the Woodbury matrix identity as before to give:

$$\begin{aligned} h_t &= \mathbf{E}(\theta_t|\theta_{t+1}, y_{1:T}) \\ &= m_t + C_t G_{t+1}' (G_{t+1} C_t G_{t+1}' + W)^{-1} (\theta_{t+1} - G_{t+1} m_t), \\ &= m_t + C_t G_{t+1}' R_{t+1}^{-1} (\theta_{t+1} - a_{t+1}), \\ H_t &= \text{Var}(\theta_t|\theta_{t+1}, y_{1:T}) \\ &= C_t - C_t G_{t+1}' R_{t+1}^{-1} G_{t+1} C_t. \end{aligned}$$

2. The mean s_t and variance S_t of the smoothed states given the data up to time T are then calculated as:

$$\begin{aligned} s_t &= \mathbf{E}\{\mathbf{E}(\theta_t|\theta_{t+1}, y_{1:T})\}, \\ &= m_t + C_t G_{t+1}' R_{t+1}^{-1} (s_{t+1} - a_{t+1}) \end{aligned}$$

and

$$\begin{aligned} S_t &= \text{Var}\{\mathbf{E}(\theta_t|\theta_{t+1}, y_{1:T})\} + \mathbf{E}\{\text{Var}(\theta_t|\theta_{t+1}, y_{1:T})\}, \\ &= C_t - C_t G_{t+1}' R_{t+1}^{-1} G_{t+1} C_t + C_t G_{t+1}' R_{t+1}^{-1} S_{t+1} R_{t+1}^{-1} G_{t+1} C_t, \\ &= C_t - C_t G_{t+1}' R_{t+1}^{-1} (R_{t+1} - S_{t+1}) R_{t+1}^{-1} G_{t+1} C_t. \end{aligned}$$

The forward filtering backward sampling algorithm for computing $\pi(\theta_{0:T}|y_{1:T}, V, W)$ is outlined in Algorithm 3.6.

Algorithm 3.6: Forward filtering backward sampling

1. Run the Kalman filter to compute the moments of the filtering distributions;
2. Draw $\theta_T \sim N(m_T, C_T)$;
3. **for** $t=T-1, \dots, 0$:
4. draw $\theta_t \sim N(h_t, H_t)$;
5. **end**

The block Gibbs sampler for simulating values from the joint distribution $\pi(\theta_{0:T}, V, W|y_{1:T})$ is shown in Algorithm 3.7.

Algorithm 3.7: Block Gibbs sampler for a DLM

1. Initialise $\psi_y^{(0)}$ and $\psi_j^{(0)}$ for $j = 1, \dots, p$;
2. **for** $i = 1, \dots, N$:
3. run the forward filtering backward sampling algorithm (Algorithm 3.6) to simulate $\underline{\theta}^{(i)}$;
4. draw $(\psi_y^{-1})^{(i)}$ from (3.11);
5. draw $(\psi_j^{-1})^{(i)}$ from (3.12) for $j = 1, \dots, p$;
6. **end**

3.4.3 Numerical stability in a DLM

The Kalman filter iteratively updates the posterior covariance C_t on each sweep of the algorithm. Directly updating this matrix is subject to issues of numerical instability; when the observation variance V is small, $C_t \approx R_t - R_t$, which can result in a non-symmetric or a negative definite covariance matrix due to rounding errors. A covariance matrix must be symmetric and positive semi-definite, so two possible methods for ensuring this and improving stability in the Kalman filter and Kalman smoother are presented here.

Joseph form

The Joseph form update (Bucy & Joseph, 2005) of C_t is given by

$$C_t = (I - K_t F_t) R_t (I - K_t F_t)' + K_t V K_t'$$

where $K_t = R_t F_t' Q_t^{-1}$ is known as the Kalman gain. This way of calculating C_t ensures the matrix is positive semi-definite by taking advantage of the properties of positive semi-definite matrices. A matrix A is positive semi-definite if for any non-zero vector \underline{b} , $\underline{b} A \underline{b}' \geq 0$. The Joseph form covariance update is the sum of two positive semi-definite matrices, of which the result is always positive semi-definite. Calculating C_t in this way ensures stability in the filter, but requires more calculations.

Singular value decomposition

An alternative method to using the Joseph form update is to update the singular value decomposition (SVD) of C_t instead of C_t directly (Oshman & Bar-Itzhack, 1986; Wang *et al.*, 1992). The SVD of any $m \times n$ real or complex matrix, A , breaks the matrix into the product of three matrices: $A = U D B'$, where U is a $m \times m$ orthogonal matrix, D

is a $m \times n$ matrix with $D_{i,j} = 0$ for $i \neq j$ and B is a $n \times n$ orthogonal matrix. The non-zero elements of D are the square roots of the positive eigenvalues of AA' (or $A'A$), known as singular values, and U and B are made up of the eigenvectors of AA' and $A'A$ respectively. As AA' and $A'A$ are positive semi-definite, their eigenvalues are all positive or zero, so D is positive semi-definite. When A is a symmetric matrix, for example when it is a covariance matrix, $A = UDU'$, since $AA' = A'A = A^2$ in this case.

The `d1m` package (Petris, 2010) in R provides an extensive range of functions that allow DLMS to be easily constructed and manipulated, including functions for filtering, smoothing and forecasting. The Kalman filter and Kalman smoother in the `d1m` package, `d1mFilter` and `d1mSmooth`, use the SVD method to improve stability. The `d1m` package is used throughout this thesis when constructing DLMS.

3.5 Model checking

Statistical model checking can be divided into two categories: checking model assumptions are met and checking model performance. Checking model assumptions is carried out less frequently in Bayesian statistics, as it is generally acknowledged that models do not represent the true data generating mechanisms and cannot be perfect. In a time series context, the inferential objective is very often forecasting. Therefore, measuring how useful a model is involves checking how well it fulfils this objective, in other words, by assessing the model predictive performance.

At each iteration of the block Gibbs sampling schemes described in Algorithm 3.5 and Algorithm 3.7, a realisation of the Markov chain whose stationary distribution is the joint distribution $\pi(\theta_{0:T}, \underline{\psi} | y_{1:T})$ is generated (specific to HMMs or DLMS). The set of parameters at iteration j of the chain, $\underline{\psi}^{(j)}$, describes the state space model with latent states $\underline{\theta}^{(j)}$. Conditional on $\theta_{1:t}^{(j)}$ and $\underline{\psi}^{(j)}$, a sample can be simulated from a n -step ahead forecast distribution at any time t . By repeating this exercise for an ensemble of samples from the posterior, we obtain a sample from the associated n -step ahead posterior predictive distribution. This accounts for both the epistemic uncertainty in the parameter values and the aleatory uncertainty in the process. Model performance is then measured using summaries from the ensemble of forecasts, which can include probability intervals based on the standard deviation of the forecasts to visualise the model uncertainty.

Examples of assessing model performance in practice are found in the remaining modelling chapters of this thesis.

3.6 Summary

In this chapter, the models that are fitted in the subsequent chapters of this thesis and the inferential procedures for fitting such models are presented. The family of models explored are state space models, which are explained in generality and then specifically for DLMS and HMMs. Bayesian inference is introduced, along with some of the general algorithms and procedures that it contains.

State space models provide a flexible framework for modelling time series data, building a dependency between the observations in a series via a series of latent states that represent an unobserved data generating process. These latent states form a Markov chain. Furthermore, in state space models advantage of the sequential nature of time series data can be taken to calculate the filtering, smoothing and forecasting distributions of the states.

HMMs are explained as state space models with a finite number of discrete hidden states. As a result of this definition, they are used to classify observations in a series into the sequence of latent states that generated them. As a result of the discrete state space, the often computationally intractable integral in the filtering, smoothing and forecasting recursions of state space models is reduced to a sum over the corresponding conditional probabilities. The forward backward algorithm can then be used to sample the series of hidden states in an MCMC scheme with data augmentation. HMMs will be used to classify activity observations into their underlying activity intensity zones.

DLMS are continuous, linear state space models, with a Gaussian state space. This chapter explained how complex DLMS are easily constructed by superimposing individual DLMS, which are then interpreted in terms of the individual components. This is how the models for glucose will be constructed later in the thesis. Because of their Gaussian nature, computational inference for the latent states consists of sequentially updating their mean and variance, given the prior initial distribution of the state vector and the available data, known as Kalman filtering and smoothing. These Kalman recursion algorithms are used to carry out forward filtering backward sampling in an MCMC scheme with data augmentation.

The parameter inference results presented in this chapter will also be used throughout the subsequent modelling chapters of this thesis.

Chapter 4

Models to classify and identify levels of activity

4.1 Background

The activity summaries presented in Section 2.3.1 represent an observed activity output from each patient over time. As implied by the results discussed in the literature review, the relationship between glucose and activity levels is not linear; walking and running have a similar overall effect on glucose, but would have very different corresponding activity observations. The multimodal lagged correlation plots, Fig. 2.13 and Fig. 2.14, also imply that the relationship between glucose and activity might not be the same for each of the activity intensities. To capture the intensity of activity, or activity ‘zone’, that each activity observation comes from, models to classify each observation into a set of hidden states can be fitted. Mixture models are first fitted as a baseline for comparison for the subsequently fitted, more complex, HMMs. The number of iterations to run each of the MCMC schemes for to fit the models is chosen by performing short pilot runs on the data and scaling up based on the effective sample size these runs produce. This number then dictates the thinning to use to reduce the MCMC output to approximately 10,000 iterations. The burn in is similarly chosen based on the trace plots of the pilot runs.

The literature in Section 1.3 suggests that there are three main activity states: low-, medium- and high activity. In the context of HMMs, several configurations are considered. Models are fitted using a known number of hidden states, namely $K = 3$ hidden activity states, trialling different within-state distributions to best explain the nature of the activity data. Initially, models are fitted to just 10,000 observations from one activity summary, the BFEN activity data, at a frequency of 1 Hz. This is later extended to the full data set of two other activity data summaries, ENMO data and step count data, to assess the flexibility of the models explored. Models fitted to lower frequencies of activity data

are explored and the number of hidden states used to capture the levels of activity being performed is adjusted where appropriate to compare model performance.

4.2 Mixture models

A mixture model, as mentioned in Section 3.1.1, can be considered a HMM with zero-order dependency between the hidden states. The unobserved states, where $Z_t \in 1, \dots, K$, are modelled as i.i.d. random variables on which the observed process, $\{Y_t\}$, is modelled conditionally. As a result, the observed parameters of a mixture model are the same as those in a HMM, $\underline{\psi}_{obs} = (\underline{\psi}_{obs,1}, \dots, \underline{\psi}_{obs,K})$, but the hidden parameters are reduced to a single vector providing the probability of each state, $\underline{\psi}_{hid} = \underline{\pi} = (\pi_1, \dots, \pi_K)$, since the transition probabilities between states are not considered.

Recall the 1 Hz BFEN activity data in Section 2.3.1. These data show a large amount of skew: the majority of observations are clustered around zero, but there is a range of shorter and longer spikes of higher ‘activity’. Modelling the distribution of the activity data by breaking them into a mixture of three components seems reasonable: the low-activity component representing the bulk of the data distributed about zero, a medium-activity component representing the shorter spikes of activity and the third, high-activity component representing the longer tail in the data.

4.2.1 Normal mixture model

To take advantage of the conjugacies of Gaussian distributions and keep the initial mixture model relatively simple, a Normal distribution is chosen initially as the within-state distribution. Whilst the observed activity levels in the BFEN data are always non-negative and could therefore be described by a within-state distribution on \mathbb{R}^+ , such as the Gamma distribution, the models should be applicable to any of the activity summaries explored. They should therefore have the flexibility to include those summaries containing negative values. It should be noted here that a Gaussian distribution, along with other candidates for a continuous within-state distribution, would not be suitable for the 1 Hz step count data summaries due to the discrete nature of these data sets. This will be considered later in the chapter.

The hierarchical Bayesian model describing the within-state distribution and prior distributions for the within-state parameters for the Normal mixture model is summarised in Model 4.1.

Model 4.1

$$y_t | Z_t = k, \mu_k, \tau_k \sim N(\mu_k, \tau_k^{-1}),$$

$$Pr(Z_t = k | \underline{\pi}) = \pi_k,$$

$$\begin{aligned}\mu_k &\sim N(\mu_0, \sigma^2) \text{ with } \mu_1 < \mu_2 < \mu_3, \\ \tau_k &\sim Ga(\alpha, \beta), \\ \underline{\pi} = (\pi_1, \pi_2, \pi_3) &\sim Dir(\underline{\alpha} = (\alpha_1, \alpha_2, \alpha_3)).\end{aligned}$$

The parameters of the observation layer of Model 4.1 are $\underline{\psi}_{obs,k} = (\mu_k, \tau_k)$, the within-state mean and precision, for each hidden state $k = 1, \dots, 3$. The data $\{y_t\}$ are the activity observations and the hidden states $\{Z_t\}$ are the corresponding activity zones. The model prior specification is semi-conjugate (conditional on the state labels) rather than fully conjugate, because the mean and precision of each state are independent a priori. A fully conjugate specification would be described by:

$$\begin{aligned}\mu_k | \tau &\sim N(\mu_0, n_0 \tau^{-1}) \text{ with } \mu_1 < \mu_2 < \mu_3, \\ \tau &\sim Ga(\alpha, \beta).\end{aligned}$$

A semi-conjugate model is chosen to align with models later in the chapter, where a fully conjugate specification is unavailable. A semi-conjugate prior specification also allows for a more flexible model to be fitted; straightforward, analytic computation of the joint prior for the μ_k that respects the identifiability constraint requires each unordered μ_k to be assigned the same prior (see below). This in turn for a fully conjugate specification imposes the constraint that τ must be equal across the hidden states, too. This constraint is explored later in the chapter.

The prior on the within-state means is subject to the constraint that $\mu_1 < \mu_2 < \mu_3$ to prevent label switching during the MCMC sampling. To understand label switching, recall the complete data likelihood from Eq. (3.8):

$$p(\underline{y} | \underline{z}, \underline{\psi}_{obs}) = \prod_{k=1}^K \prod_{t: z_t=k} p(y_t | \underline{\psi}_{obs,k}).$$

It is clear that relabelling the states and corresponding parameters accordingly does not change the likelihood; it is invariant to permutations in the state labels. Coupled with a prior for the $\underline{\psi}_{obs,k}$ that is independent across k and that assigns the same distribution to each $\underline{\psi}_{obs,k}$ would therefore yield parameters that were non-identifiable in the posterior. Consequently, the posterior samples for the corresponding mixture model, or more generally the corresponding HMM, are subject to label switching, where the hidden states permute during the MCMC sampling process. More information on label switching can be found in Frühwirth-Schnatter (2006). The addition of the identifiability constraint on the within-state means forces the sampler to maintain a unique set of labels given the order of the means. This is easy to impose given the geometry of the activity zones.

If a common Normal prior is assumed for each unordered μ_k independently, then

accounting for the identifiability constraint implies that μ_1 is drawn marginally from the minimum of three Normal distributions, μ_2 is drawn conditionally from the minimum of two Normals, with a lower-truncation at μ_1 , and μ_3 is drawn from a Normal distribution, lower-truncated at μ_2 . To derive the full conditional distributions for each μ_k , the joint prior $\pi(\mu_1, \mu_2, \mu_3)$ must be computed to account for the dependency between the means.

Consider i.i.d. continuous random variables, X_1, \dots, X_n . The probability of at least one of these being less than x is

$$Pr(X_{(1)} \leq x) = 1 - Pr(X_{(1)} \geq x) = 1 - Pr(X_1 \geq x, \dots, X_n \geq x) = 1 - \{1 - F(x)\}^n.$$

In what follows, let $\Phi(x)$ be the standard Normal cumulative density function (CDF) with associated probability density function (PDF) $\phi(x)$. The PDF of each μ_k is the derivative of the CDF and is calculated below.

As μ_1 is drawn from the minimum of three Normal distributions, the CDF of μ_1 is:

$$F(\mu_1) = 1 - \left\{ 1 - \Phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right) \right\}^3.$$

The PDF of μ_1 is then:

$$f(\mu_1) = \frac{dF(\mu_1)}{d\mu_1} = \frac{3}{\sigma} \phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right) \left\{ 1 - \Phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right) \right\}^2.$$

The CDF of μ_2 is a minimum of two Normal distributions with a lower truncation. The CDF of a truncated Normal random variable, lower-truncated at a and upper-truncated at b is:

$$F(x) = \frac{\Phi\left(\frac{x-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)}.$$

The CDF of μ_2 truncated at μ_1 is therefore written:

$$F(\mu_2|\mu_1) = 1 - \left\{ 1 - \frac{\Phi\left(\frac{\mu_2-\mu_0}{\sigma}\right) - \Phi\left(\frac{\mu_1-\mu_0}{\sigma}\right)}{1 - \Phi\left(\frac{\mu_1-\mu_0}{\sigma}\right)} \right\}^2,$$

with PDF:

$$f(\mu_2|\mu_1) = \frac{2\phi\left(\frac{\mu_2-\mu_0}{\sigma}\right)}{\sigma\left\{1 - \Phi\left(\frac{\mu_1-\mu_0}{\sigma}\right)\right\}} \left\{ 1 - \frac{\Phi\left(\frac{\mu_2-\mu_0}{\sigma}\right) - \Phi\left(\frac{\mu_1-\mu_0}{\sigma}\right)}{1 - \Phi\left(\frac{\mu_1-\mu_0}{\sigma}\right)} \right\}.$$

The CDF of μ_3 is simply a truncated Normal CDF:

$$F(\mu_3|\mu_1, \mu_2) = F(\mu_3|\mu_2) = \frac{\Phi\left(\frac{\mu_3-\mu_0}{\sigma}\right) - \Phi\left(\frac{\mu_2-\mu_0}{\sigma}\right)}{1 - \Phi\left(\frac{\mu_2-\mu_0}{\sigma}\right)},$$

with PDF:

$$f(\mu_3|\mu_2) = \frac{\phi\left(\frac{\mu_3 - \mu_0}{\sigma}\right)}{\sigma \left\{1 - \Phi\left(\frac{\mu_2 - \mu_0}{\sigma}\right)\right\}}.$$

The joint prior distribution of all μ_k is then:

$$\begin{aligned} \pi(\mu_1, \mu_2, \mu_3) &= f(\mu_1)f(\mu_2|\mu_1)f(\mu_3|\mu_2), \\ &= \frac{3}{\sigma} \phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right) \left\{1 - \Phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right)\right\}^2 \\ &\quad \times \frac{2\phi\left(\frac{\mu_2 - \mu_0}{\sigma}\right)}{\sigma \left\{1 - \Phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right)\right\}} \left\{1 - \frac{\Phi\left(\frac{\mu_2 - \mu_0}{\sigma}\right) - \Phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right)}{1 - \Phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right)}\right\} \\ &\quad \times \frac{\phi\left(\frac{\mu_3 - \mu_0}{\sigma}\right)}{\sigma \left\{1 - \Phi\left(\frac{\mu_2 - \mu_0}{\sigma}\right)\right\}}, \\ &= \frac{6}{\sigma^3} \phi\left(\frac{\mu_1 - \mu_0}{\sigma}\right) \phi\left(\frac{\mu_2 - \mu_0}{\sigma}\right) \phi\left(\frac{\mu_3 - \mu_0}{\sigma}\right), \\ &\propto \pi(\mu_1)\pi(\mu_2)\pi(\mu_3), \text{ with } \mu_1 < \mu_2 < \mu_3. \end{aligned}$$

The μ_k can therefore be considered independently when calculating the full conditional distributions, as long as the ordering constraint is respected in the sampler. The derivation of the truncated Normal CDFs and the joint prior for all μ_k is novel.

A Gamma prior is assigned to the within-state precisions τ_k . This is a sensible prior as it restricts the sampling space to the set of positive real numbers and is semi-conjugate to the Normal distribution, used for the likelihood.

The complete data likelihood, given the Normal within-state distributions, is written

$$p(\underline{y}|\underline{z}, \underline{\psi}_{obs}) \propto \prod_{k=1}^3 \prod_{t:z_t=k} \tau_k^{1/2} \exp\left[-\frac{\tau_k}{2}(y_t - \mu_k)^2\right]. \quad (4.1)$$

The full conditional distributions of the observation layer parameters can therefore be derived using Bayes' formula to combine the prior information with Eq. (4.1) to construct the posterior parameter distribution:

$$\begin{aligned} \pi(\mu_1, \mu_2, \mu_3, \tau_1, \tau_2, \tau_3|\underline{y}, \underline{z}) &\propto \prod_{k=1}^3 \pi(\mu_k)\pi(\tau_k) \prod_{t:z_t=k} \tau_k^{1/2} \exp\left[-\frac{\tau_k}{2}(y_t - \mu_k)^2\right], \\ &\propto \prod_{k=1}^3 \phi\left(\frac{\mu_k - \mu_0}{\sigma}\right) \tau_k^{\alpha-1} e^{-\tau_k\beta} \prod_{t:z_t=k} \tau_k^{1/2} \exp\left[-\frac{\tau_k}{2}(y_t - \mu_k)^2\right], \end{aligned}$$

with $\mu_1 < \mu_2 < \mu_3$. Then conditionally for each parameter:

$$\pi(\mu_k|\cdot) \propto \exp\left[-\frac{1}{2\sigma^2}(\mu_k - \mu_0)^2 - \frac{\tau_k}{2} \sum_{t:z_t=k} (y_t - \mu_k)^2\right],$$

$$\begin{aligned} &\propto \exp \left[-\frac{1}{2} \left\{ \frac{1}{\sigma^2} (\mu_k^2 - 2\mu_0\mu_k) + \tau_k \sum_{t:z_t=k} (\mu_k^2 - 2y_t\mu_k) \right\} \right], \\ &\propto \exp \left[-\frac{1}{2} \left\{ \mu_k^2 \left(\frac{1}{\sigma^2} + \tau_k N_k(\underline{z}) \right) - 2\mu_k \left(\frac{\mu_0}{\sigma^2} + \tau_k \sum_{t:z_t=k} y_t \right) \right\} \right], \end{aligned}$$

so

$$\mu_k | \cdot \sim N \left(\left(\frac{1}{\sigma^2} + \tau_k N_k(\underline{z}) \right)^{-1} \left(\frac{\mu_0}{\sigma^2} + \tau_k \sum_{t:z_t=k} y_t \right), \left(\frac{1}{\sigma^2} + \tau_k N_k(\underline{z}) \right)^{-1} \right), \quad (4.2)$$

subject to the constraint $\mu_1 < \mu_2 < \mu_3$. Each μ_k can therefore be sampled in the order of μ_1, μ_2, μ_3 from the Normal distribution in Eq (4.2) with a lower-truncation at μ_{k-1} where $\mu_0 = -\infty$. Next,

$$\pi(\tau_k | \cdot) \propto \tau_k^{\alpha + N_k(\underline{z})/2 - 1} \exp \left[-\tau_k \left(\beta + \frac{1}{2} \sum_{t:z_t=k} (y_t - \mu_k)^2 \right) \right],$$

so

$$\tau_k | \cdot \sim Ga \left(\alpha + \frac{N_k(\underline{z})}{2}, \beta + \frac{1}{2} \sum_{t:z_t=k} (y_t - \mu_k)^2 \right). \quad (4.3)$$

The full conditional distribution of the hidden parameter $\underline{\pi}$ is analogous to the full conditional distributions derived for $\underline{\nu}$ and $\underline{\pi}_k$ in Eq. (3.10) and Eq. (3.9), respectively. The Dirichlet prior on $\underline{\pi}$ results in a Dirichlet posterior, such that

$$\underline{\pi} | \underline{z} \sim D(\underline{\alpha} + \underline{m}'(\underline{z})), \quad (4.4)$$

where $\underline{m}'(\underline{z}) = (m'_1(\underline{z}), m'_2(\underline{z}), m'_3(\underline{z}))$ is a vector of the number of observations sampled from each of the hidden states.

As the hidden states are independent, they do not need to be sampled recursively using Algorithm 3.4. Instead, the states are sampled from the conditional probability mass function defined by

$$Pr(Z_t = \ell | y_t, \underline{\pi}, \underline{\psi}_{obs}) = \frac{p(y_t | \underline{\psi}_{obs, \ell}) \pi_\ell}{\sum_{k=1}^3 p(y_t | \underline{\psi}_{obs, k}) \pi_k}, \quad (4.5)$$

for all t .

Altogether, the block Gibbs sampler for Model 4.1 is provided in Algorithm 4.1.

Results

Algorithm 4.1 was run for $N = 200,000$ iterations, plus a burn in of 1,000 and was thinned to every 15th sample. A subset of 10,000 observations from the activity data of one patient

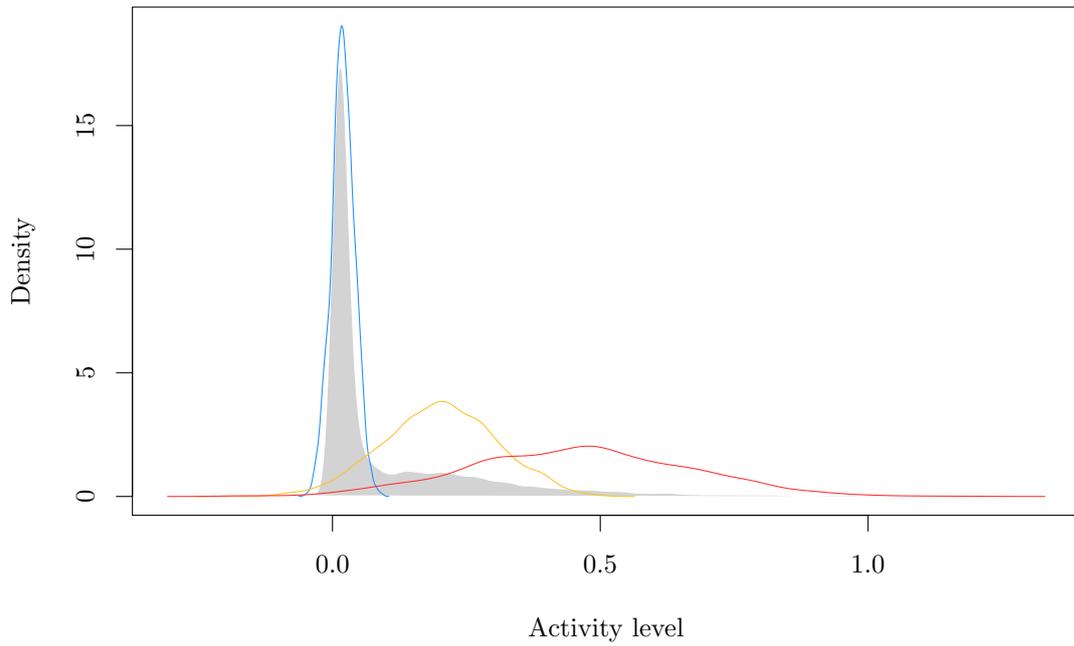


Figure 4.1: Normal mixture model posterior empirical within-state densities for low (blue), medium (gold) and high (red) activity zones fitted to 10,000 BFEN observations (grey density).

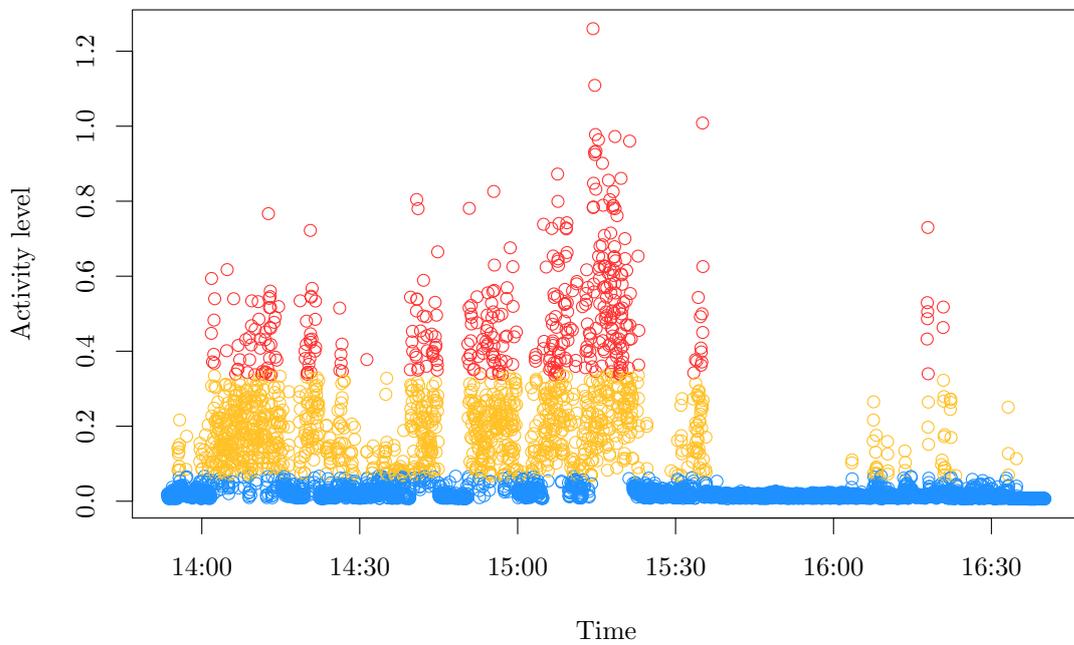


Figure 4.2: Portion of BFEN data classified using a Normal mixture model into low (blue), medium (gold) and high (red) activity zones (thinned by 20 for plotting).

Algorithm 4.1: Normal mixture model MCMC scheme

1. Initialise $\mu_k^{(0)}$, $\tau_k^{(0)}$ and $\underline{\pi}^{(0)}$, for all k , by sampling from the prior distributions in Model 4.1;
2. **for** $i = 1, \dots, N$:
3. sample the hidden states $\underline{z}^{(i)}$ according to the probabilities (4.5);
4. **for** $k = 1, \dots, 3$:
5. draw $\mu_k^{(i)}$ from (4.2), truncated on $(\mu_{k-1}^{(i)}, \mu_{k+1}^{(i-1)})$ where $\mu_0^{(i)} = -\infty$ and $\mu_4^{(i-1)} = \infty$ for all i ;
6. **end**
7. draw $\tau_k^{(i)}$ from (4.3) for $k = 1, \dots, 3$;
8. draw $\underline{\pi}^{(i)}$ from (4.4);
9. **end**

is used. This is sufficient to see the range of activity levels being performed by the patient and gauge how suitable the model is. The hyperparameters in the priors specified for μ_k , τ_k and $\underline{\pi}$ in Model 4.1 are chosen to be

$$\mu_0 = 0.1, \quad \sigma = 0.5, \quad \alpha = 1, \quad \beta = 1, \quad \text{and} \quad \underline{\alpha} = (10, 2, 1),$$

respectively. The priors for μ_k and τ_k are the same for all k , which reflects prior ignorance about the within-state distributions. The value of μ_0 is set at 0.1, approximately the mean of the BFEN data and σ is set at 0.5, approximately two standard deviations of the data. The $Ga(1, 1)$ prior on τ_k has a large enough variance to cover a range of plausible values for each of the within-state precisions. The hyperparameter $\underline{\alpha}$ is reflective of the number of minutes that would be expected for a person to spend in each activity zone relative to the other zones, i.e. 10 minutes in zone one, two minutes in zone two and one minute in zone three. Note that this prior asymmetry alone was not enough to prevent label switching, which is why an identifiability constraint on the μ_k was still required.

The results from running Algorithm 4.1, and indeed the results from the models fitted throughout this chapter, are summarised in two ways: via a plot of the posterior empirical within-state densities, Fig. 4.1; and via a plot of the modal posterior state classifications, Fig. 4.2. The posterior within-state density plot is produced by taking a sample of the MCMC draws for the observation parameters and drawing a random variable for each state on each draw. These posterior densities are then compared to the actual density of the data. On each iteration of Algorithm 4.1, and all subsequent HMM MCMC algorithms, the state that was sampled for each observation is recorded. The modal state that was sampled for each observation once the algorithm has run is then used to classify the observations in the modal state classification plot.

The MCMC process has correctly identified three distinct activity states, shown clearly

by the three densities in Fig. 4.1 and bands of coloured observations in Fig. 4.2. The posterior densities capture the components within the data by fitting a low activity component with a small posterior variance to the bulk of the data at zero, a second component with a larger variance to the density above zero and a third, flat component to capture the long tail of low density, higher observations. The curves do not allocate much density to negative activity levels, which shows the data provide enough information to encourage realistic posterior parameter densities. There seem to be sensible boundaries between the state allocations; there is very little overlap between the observations allocated to each state in Fig. 4.2.

The lowest state, blue observations in Fig. 4.2 and blue posterior within-state density in Fig.4.1, corresponds to periods of inactivity, i.e. the observed activity levels close to zero. The second activity state, gold observations and gold posterior within-state density, reflects small movements carried out by the patient. Recalling that the data are at a frequency of one observation per second, this is a logical interpretation for the medium activity component. Lastly, the third activity state, red observations and red within-state density, corresponds to larger movements performed by the patient and periods of activity, that could involve walking.

The number of transitions between states for the classified observations is 1,047 (calculated as the mean of the number of transitions at each iteration of the sampler), out of 10,000 data points. The aim of the HMMs fitted in the subsequent sections is to introduce dependency between the observations so that this figure and the most likely sequence of states driving it are realistic. A transition matrix is constructed by taking the mean over the number of transitions between each pair of states at each iteration of the MCMC sampler:

$$P = \begin{pmatrix} 6643 & 219 & 5 \\ 224 & 1787 & 297 \\ 0 & 302 & 523 \end{pmatrix}.$$

The mean counts in this matrix support the use of a more complex structure to model the underlying activity zones as there is a clear difference in probability of what the next state in the process will be, depending on the current value of the state.

4.3 Hidden Markov models

The results from fitting the mixture model in the previous section support the use of a HMM, in which the hidden states exhibit the behaviour of a first-order Markov chain. This assumption prevents the number of transitions between states from reflecting unrealistic behaviours, for example regularly switching between zones one and three without passing through state two, and has the effect of smoothing over the distribution of mixture

components over time.

4.3.1 Normal HMM

As in the mixture model, a HMM with a semi-conjugate Normal within-state distribution is fitted first. The Bayesian hierarchical model, described in Model 4.2, extends the specification in Model 4.1 to include the probability of transitioning to state k from state j , with hidden parameters $\underline{\psi}_{hid} = (\nu, \underline{\pi}_j)$, for $j = 1, \dots, 3$.

Model 4.2

$$\begin{aligned}
 y_t | Z_t = k, \mu_k, \tau_k &\sim N(\mu_k, \tau_k^{-1}), \\
 Pr(Z_0 = k | \underline{\nu}) &= \nu_k, \\
 Pr(Z_t = k | Z_{t-1} = j) &= \pi_{j,k}, \\
 \mu_k &\sim N(\mu_0, \sigma^2) \text{ with } \mu_1 < \mu_2 < \mu_3, \\
 \tau_k &\sim Ga(\alpha, \beta), \\
 \underline{\nu} = (\nu_1, \nu_2, \nu_3) &\sim Dir(\underline{\gamma} = (\gamma_1, \gamma_2, \gamma_3)), \\
 \underline{\pi}_j = (\pi_{j,1}, \pi_{j,2}, \pi_{j,3}) &\sim Dir(\underline{\alpha}_j = (\alpha_{j,1}, \alpha_{j,2}, \alpha_{j,3})).
 \end{aligned}$$

The assumption of independence, a priori, between the observed and hidden parameters, $\underline{\psi}_{obs}$ and $\underline{\psi}_{hid}$, of Model 4.2 means that the full conditional distributions for μ_k and τ_k are the same as for Model 4.1, Eq. (4.2) and (4.3), respectively. The formulation of the complete data likelihood is also the same, Eq. (4.1), and the full conditional distributions for $\underline{\pi}_k$ and $\underline{\nu}$ are as derived in Section 3.3.1, Eq. (3.9) and (3.10).

The block Gibbs sampler for Model 4.2 follows the structure of the block Gibbs sampler outlined for HMMs in Algorithm 3.5, where $\underline{\psi}_{obs,k} = (\mu_k, \tau_k)$ are drawn as in Algorithm 4.1.

Results

The block Gibbs sampler for Model 4.2 was run for $N = 100,000$ iterations, plus a burn in of 1,000 and was thinned by 10. The hyperparameters $\mu_0 = 0.1$, $\sigma = 0.5$, $\alpha = 1$ and $\beta = 1$ are the same as for the Normal mixture model. The Dirichlet prior on the transition matrix has large values on the diagonal to discourage switching between states. As for the prior on $\underline{\pi}$ in Model 4.1, these values were determined based on prior beliefs about the amount of time (seconds) expected to stay in each state and additionally were scaled down to reduce the prior variance:

$$\underline{\alpha} = \begin{pmatrix} 0.50 & 0.05 & 0.05 \\ 0.05 & 0.15 & 0.05 \\ 0.05 & 0.05 & 0.10 \end{pmatrix},$$

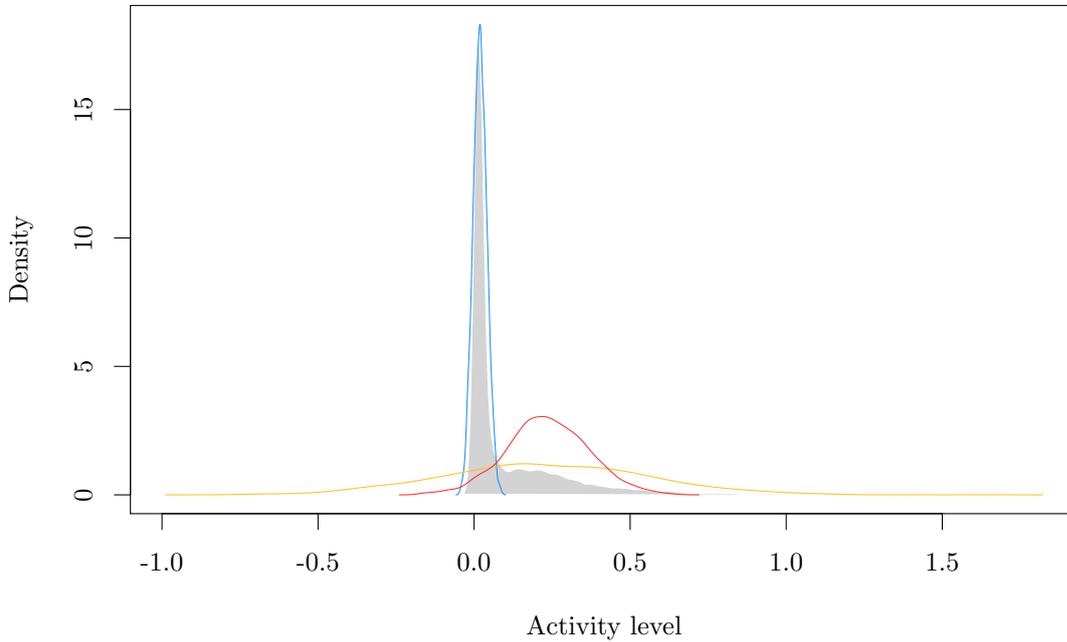


Figure 4.3: Normal HMM posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to 1 Hz BFEN data, compared to the BFEN density (grey).

so $\underline{\alpha}_1 = (0.5, 0.05, 0.05)$ and so on. A stronger prior with smaller variance was tested and was found to have very little overlap with the posterior transition probability densities, suggesting it could drive the posterior density to include values that are not reflective of the behaviour in the data.

It is expected that type II diabetic patients spend most of their time in the lowest activity zone, with short bursts of activity that fall into the second zone and some shorter bouts of higher activity (zone three). The initial state vector prior, $\underline{\gamma} = (0.95, 0.03, 0.02)$, reflects the high probability of being in activity zone one.

The posterior within-state densities in Fig. 4.3 show that the Gibbs sampling algorithm for Model 4.2 has not detected three distinct state components based on three different within-state means. The state corresponding to the lowest activity zone (blue density) sufficiently captures the large density of observations about zero, however the areas of highest density of the two other hidden states overlap. The posterior draws for μ_2 and μ_3 are summarised in Fig. 4.4 and show that the densities are very similar, whilst the ordering constraint is forcing the draws for μ_3 (magenta) to be marginally higher than those of μ_2 (blue). Instead, the sampling algorithm has differentiated the second and third states with different posterior variances. The posterior within-state density of state two (gold) is very flat, capturing the higher activity observations that stretch out into the tail. The third state has a smaller posterior within-state variance, concentrated on the observations just

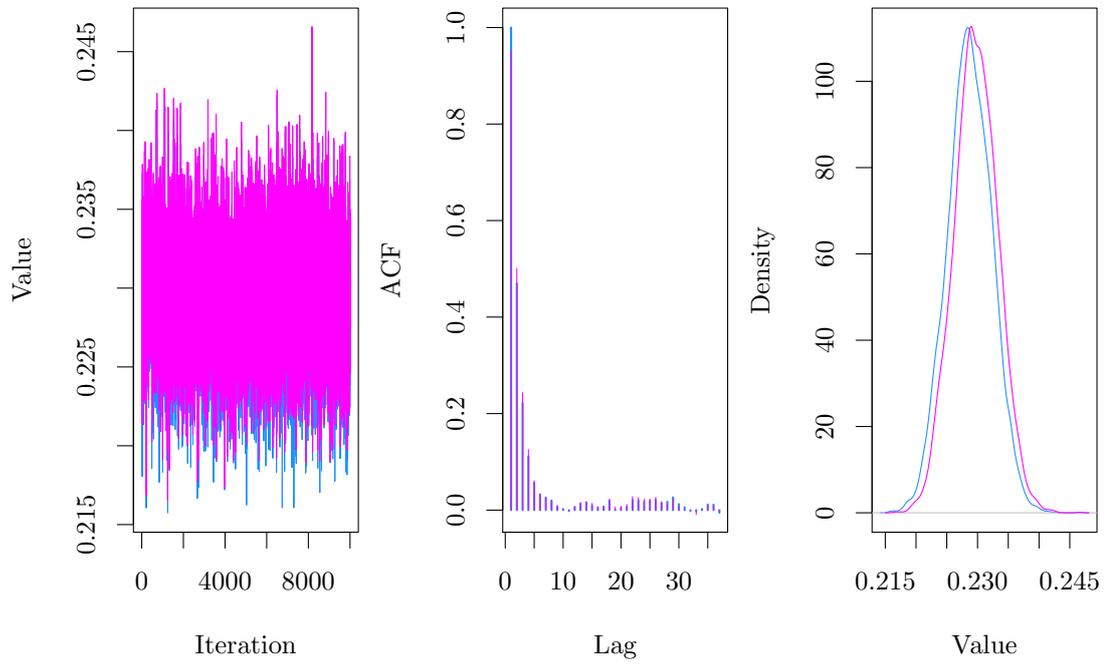


Figure 4.4: MCMC output for μ_2 (blue) and μ_3 (magenta) in Model 4.2.

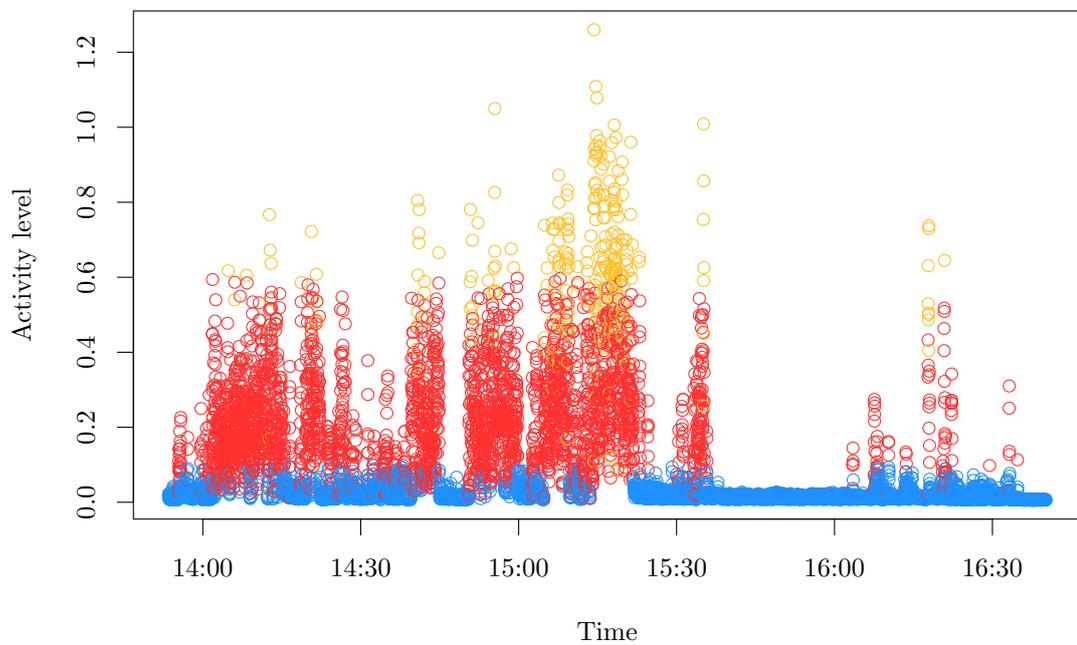


Figure 4.5: 1 Hz BFEN data classified using a Normal HMM into low (blue), medium (gold) and high (red) levels of activity.

above zero.

The effect the similar posterior within-state means and different posterior within-state variances of states two and three has on the modal classification of the observations is shown in Fig. 4.5. Despite the constraint on the means, the forward backward algorithm samples the second state (gold) for the higher observations and the third state (red) for the medium level observations, because of the wider variance on the second state. This misclassification and the overlapping posterior densities make Model 4.2 unsuitable for classifying the activity data.

A further example to emphasise the poor performance of Model 4.2 is in fitting the model to the 1 Hz ENMO data, classified in Fig. 4.6. Here, the sampling algorithm has drawn similar within-state means for all three hidden states, again with three different posterior variances. This causes the phenomenon of the resulting activity states being split; as in the BFEN data, the bulk of observations about zero are allocated to state one, however the long tails of the posterior within-state densities for the other two states result in observations either side of zero being allocated to a mixture of the other two classes. Negative ENMO activity observations being classified as coming from higher zones of activity is an incorrect interpretation of the data. The second state also corresponds to the state with the largest variance, so the observations above zero are misclassified (the red observations correspond to state three, but should be classed as the second, medium level activity zone).

To overcome this ‘states within states’ problem caused by the unrestricted within-state variance parameters, the states are given a shared precision, τ . This model constraint should force the MCMC algorithm to find distinct means for each state.

4.3.2 Normal HMM with shared variance

The Normal HMM with shared within-state variance, parametrised by a shared precision τ , is shown in Model 4.3.

Model 4.3

$$\begin{aligned}
 y_t | Z_t = k, \mu_k, \tau_k = \tau &\sim N(\mu_k, \tau^{-1}), \\
 Pr(Z_0 = k | \underline{\nu}) &= \nu_k, \\
 Pr(Z_t = k | Z_{t-1} = j) &= \pi_{j,k}, \\
 \mu_k &\sim N(\mu_0, \sigma^2) \text{ with } \mu_1 < \mu_2 < \mu_3, \\
 \tau &\sim Ga(\alpha, \beta), \\
 \underline{\nu} = (\nu_1, \nu_2, \nu_3) &\sim Dir(\underline{\gamma} = (\gamma_1, \gamma_2, \gamma_3)), \\
 \underline{\pi}_j = (\pi_{j,1}, \pi_{j,2}, \pi_{j,3}) &\sim Dir(\underline{\alpha}_j = (\alpha_{j,1}, \alpha_{j,2}, \alpha_{j,3})).
 \end{aligned}$$

The within-state mean is subject to the same identifiability constraint as before. The

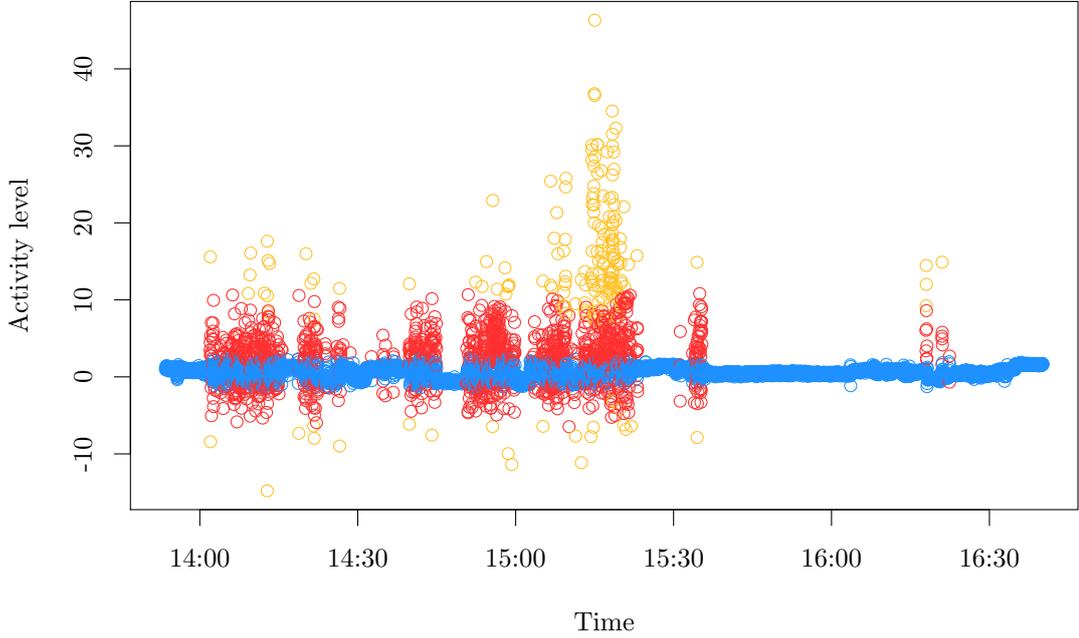


Figure 4.6: 1 Hz ENMO data classified using a Normal HMM into low (blue), medium (gold) and high (red) levels of activity.

subscript on τ_k in the likelihood, Eq. (4.1), is dropped to reflect the model updates, and, when combined with the prior for τ , produces the following full conditional distribution:

$$\tau|\underline{y}, \underline{z} \sim Ga\left(\alpha + \frac{T}{2}, \beta + \frac{1}{2} \sum_{k=1}^3 \sum_{t:z_t=k} (y_t - \mu_k)^2\right). \quad (4.6)$$

The full conditional distributions for the μ_k are as in Eq. (4.2) and the hidden parameter full conditional distributions also remain the same as in Eq. (3.9) and Eq. (3.10). The MCMC scheme is implemented in a block Gibbs sampler.

Results

Model 4.3 was run under the same conditions as Model 4.2: using the same number of iterations, burn in period and thin, and using the same prior hyperparameters.

Implementing the shared variance parameter in Model 4.3 has the desired effect of improving the identifiability of the three distinct activity zones. Fig. 4.7 shows three clear bands of activity, with reasonable boundaries between them. The lowest activity zone now includes some non-zero activity observations, though these observations still reflect very low levels of activity. The boundaries between the observations classified within the subsequent activity zones have also shifted slightly higher than in Model 4.1. The

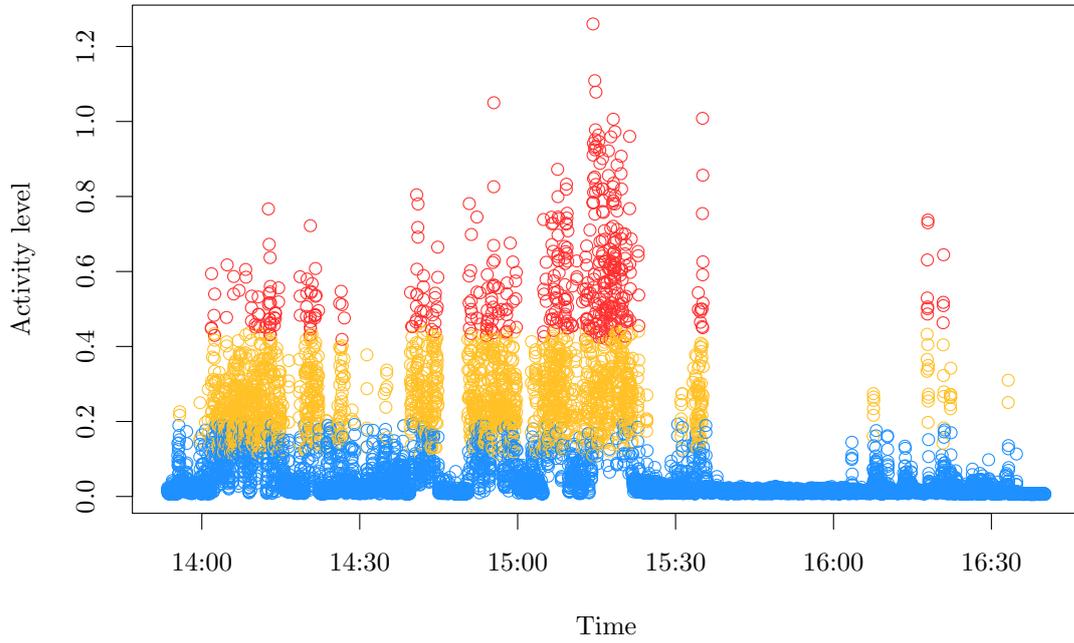


Figure 4.7: 1 Hz BFEN data classified using a Normal HMM with shared within-state variance into low (blue), medium (gold) and high (red) levels of activity.

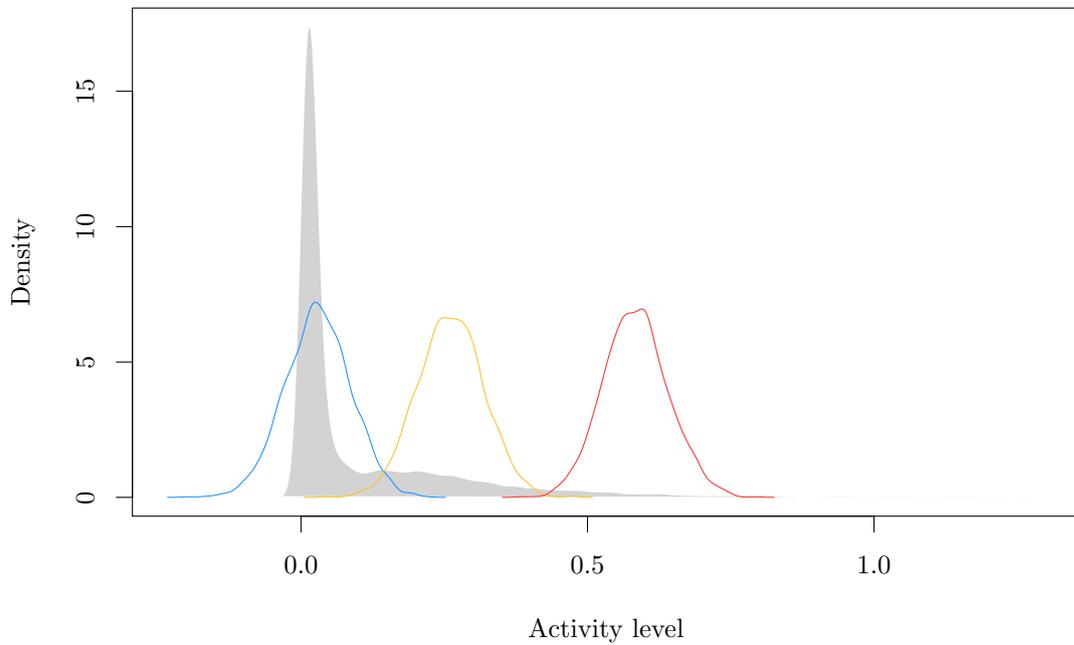


Figure 4.8: Normal HMM, with shared within-state variance, posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to 1 Hz BFEN data, compared to the actual BFEN density (grey).

interpretation of each activity zone given the marginally different boundaries is still the same, with zone one representing the patient being ‘at rest’, zone two representing small movements and zone three being larger movements/activity.

Fig. 4.8 shows the three within-state densities are spread reasonably across the density of the data, with what look like sensible within-state means. There is an area of overlapping density between the posterior within-state densities of the two lower activity zones (gold and blue), which has caused more of a blended boundary between these two states in the classification plot. As a result of setting a shared variance between the states, the variance of state one is quite large in comparison to the data it represents, but this is a trade-off for better overall fit of the model in overcoming identifiability problems.

The mean number of transitions between states on each iteration of the sampler is 745, which is a positive change that the first-order dependency between the states has brought about: a reduction from the 1,047 state transitions that were seen in the mixture model.

Overall, the shared variance parameter has improved the fit of the within-state densities to the data. It is clear, however, that whilst Model 4.2 does a good job at classifying the activity observations into sensible activity zones, the fit of the posterior within-state densities could be improved upon to fit closer to the actual density of the data. To overcome this, an alternative within-state distribution that accommodates skewed data is proposed.

4.3.3 Skew Normal HMM

To better fit the long tails in the activity data, a skew Normal within-state distribution is explored. The skew Normal distribution has three parameters: location, $\xi \in \mathbb{R}$; shape, $\alpha \in \mathbb{R}$; and scale, $\omega \in \mathbb{R} > 0$. The location and scale parameters are analogous to the mean and standard deviation in a Normal distribution, with the location representing the most dense part of the distribution and the scale representing the spread of the density. The shape parameter controls how skewed the distribution is: $\alpha < 0$ represents a negatively skewed density, $\alpha > 0$ represents a positively skewed density and $\alpha = 0$ recovers a Normal distribution. As with the Normal distribution, the skew Normal distribution is specified on the set of real numbers, so can accommodate all of the activity summaries. The hierarchical HMM is described in Model 4.4.

Model 4.4

$$\begin{aligned}
 y_t | Z_t = k, \xi_k, \omega_k = \omega, \alpha_k &\sim SN(\xi_k, \omega, \alpha_k), \\
 Pr(Z_0 = k | \mathcal{Y}) &= \nu_k, \\
 Pr(Z_t = k | Z_{t-1} = j) &= \pi_{j,k}, \\
 \xi_k &\sim N(m_\xi, s_\xi^2) \text{ with } \xi_1 < \xi_2 < \xi_3, \\
 \alpha_k &\sim N(m_\alpha, s_\alpha^2),
 \end{aligned}$$

$$\begin{aligned}\omega &\sim Ga(a, b), \\ \underline{\nu} &= (\nu_1, \nu_2, \nu_3) \sim Dir(\underline{\gamma} = (\gamma_1, \gamma_2, \gamma_3)), \\ \underline{\pi}_j &= (\pi_{j,1}, \pi_{j,2}, \pi_{j,3}) \sim Dir(\underline{\alpha}_j = (\alpha_{j,1}, \alpha_{j,2}, \alpha_{j,3})).\end{aligned}$$

As with the models trialled in this chapter so far, an ordering constraint is imposed on the location parameter ξ_k , for $k = 1, \dots, 3$, such that $\xi_1 < \xi_2 < \xi_3$ to prevent label switching in the MCMC sampler. The Normal prior distribution on each ξ_k therefore implies that ξ_1 is drawn from the minimum of three Normals, ξ_2 is drawn from the minimum of two Normals, lower-truncated at ξ_1 , and ξ_3 is drawn from a Normal distribution, lower truncated at ξ_2 . This results in a joint prior that is proportional to the product of the individual Normal prior distributions, as seen in Section 4.2.1:

$$\pi(\xi_1, \xi_2, \xi_3) \propto \pi(\xi_1)\pi(\xi_2)\pi(\xi_3) \text{ with } \xi_1 < \xi_2 < \xi_3.$$

Results for the skew Normal model with separate within-state scale parameters are not presented here, but showed the same identifiability problems as Model 4.2, where the states appear to be split when classifying the observations, as in Fig. 4.5 and Fig. 4.6, due to the location parameter of each state being similar and a large variation in the scale parameter of each state. To prevent identifiability issues between the states, the scale parameter is made common: $\omega_k = \omega$ for all k . The shape parameter α_k has the same prior hyperparameters for all k to reflect prior ignorance with respect to variation in skewness across activity zones.

The complete data likelihood reflective of the new within-state distribution is:

$$\begin{aligned}p(\underline{y}|\underline{z}, \underline{\psi}_{obs}) &= \prod_{k=1}^3 \prod_{t:z_t=k} p(y_t|\underline{\psi}_{obs,k}), \\ &= \prod_{k=1}^3 \prod_{t:z_t=k} \frac{2}{\omega\sqrt{2\pi}} \exp\left\{-\frac{(y_t - \xi_k)^2}{2\omega^2}\right\} \Phi\left(\alpha_k \left(\frac{y_t - \xi_k}{\omega}\right)\right).\end{aligned}$$

The novel derivations of the full conditional distributions for α_k , ξ_k and shared ω are then:

$$\begin{aligned}\pi(\underline{\psi}_{obs}|\underline{y}, \underline{z}) &\propto \pi(\omega) \prod_{k=1}^3 \pi(\xi_k)\pi(\alpha_k) \prod_{t:z_t=k} p(y_t|\underline{\psi}_{obs,k}), \\ &\propto \omega^{a-1} e^{-\omega b} \prod_{k=1}^3 \exp\left[-\frac{1}{2s_\xi^2}(\xi_k - m_\xi)^2\right] \exp\left[-\frac{1}{2s_\alpha^2}(\alpha_k - m_\alpha)^2\right] \\ &\quad \prod_{t:z_t=k} \frac{1}{\omega} \exp\left[-\frac{1}{2\omega^2}(y_t - \xi_k)^2\right] \Phi\left(\alpha_k \left(\frac{y_t - \xi_k}{\omega}\right)\right), \\ &\propto \omega^{a-N-1} e^{-\omega b}\end{aligned}$$

$$\begin{aligned} & \times \prod_{k=1}^3 \exp \left[-\frac{1}{2s_\xi^2} (\xi_k - m_\xi)^2 - \frac{1}{2s_\alpha^2} (\alpha_k - m_\alpha)^2 - \frac{1}{2\omega^2} \sum_{t:z_t=k} (y_t - \xi_k)^2 \right] \\ & \times \prod_{t:z_t=k} \Phi \left(\alpha_k \left(\frac{y_t - \xi_k}{\omega} \right) \right), \end{aligned}$$

so

$$\pi(\omega | \underline{y}, \underline{z}) \propto \omega^{a-N-1} \exp \left[-\omega b - \frac{1}{2\omega^2} \sum_{k=1}^3 \sum_{t:z_t=k} (y_t - \xi_k)^2 \right] \prod_{t:z_t=k} \Phi \left(\alpha_k \left(\frac{y_t - \xi_k}{\omega} \right) \right), \quad (4.7)$$

$$\pi(\alpha_k | \underline{y}, \underline{z}) \propto \exp \left[-\frac{\alpha_k (\alpha_k - 2m_\alpha)}{2s_\alpha^2} \right] \prod_{t:z_t=k} \Phi \left(\alpha_k \left(\frac{y_t - \xi_k}{\omega} \right) \right), \quad (4.8)$$

$$\pi(\xi_k | \underline{y}, \underline{z}) \propto \exp \left[-\frac{1}{2\omega^2} \sum_{t:z_t=k} (y_t - \xi_k)^2 - \frac{\xi_k (\xi_k - 2m_\xi)}{2s_\xi^2} \right] \prod_{t:z_t=k} \Phi \left(\alpha_k \left(\frac{y_t - \xi_k}{\omega} \right) \right). \quad (4.9)$$

The full conditional distributions of the within-state parameters are analytically intractable and cannot be sampled from directly. MH steps are therefore necessary for sampling from the posterior probability densities of each one. To do this, a proposal distribution must be suggested for each parameter.

To respect the ordering constraint in ξ , ξ_k^* is proposed from a truncated Normal distribution (truncated between the current values of ξ_{k-1} and ξ_{k+1} as appropriate), with mean ξ_k and standard deviation equal to tuning parameter $\lambda_{\xi,k}$, i.e. $\xi_k^* \sim TN(\xi_k, \lambda_{\xi,k}; \xi_{k-1}, \xi_{k+1})$ where $\xi_0 = -\infty$ and $\xi_4 = \infty$. The proposal ratio for ξ_k is then:

$$\begin{aligned} \frac{q(\xi_k^*, \xi_k)}{q(\xi_k, \xi_k^*)} &= \frac{\phi \left(\frac{\xi_k - \xi_k^*}{\lambda_{\xi,k}} \right)}{\lambda_{\xi,k} \left(\Phi \left(\frac{\xi_{k+1} - \xi_k^*}{\lambda_{\xi,k}} \right) - \Phi \left(\frac{\xi_{k-1} - \xi_k^*}{\lambda_{\xi,k}} \right) \right)} \frac{\lambda_{\xi,k} \left(\Phi \left(\frac{b - \xi_k}{\lambda_{\xi,k}} \right) - \Phi \left(\frac{\xi_{k-1} - \xi_k}{\lambda_{\xi,k}} \right) \right)}{\phi \left(\frac{\xi_k^* - \xi_k}{\lambda_{\xi,k}} \right)} \\ &= \frac{\Phi \left(\frac{\xi_{k+1} - \xi_k}{\lambda_{\xi,k}} \right) - \Phi \left(\frac{\xi_{k-1} - \xi_k}{\lambda_{\xi,k}} \right)}{\Phi \left(\frac{\xi_{k+1} - \xi_k^*}{\lambda_{\xi,k}} \right) - \Phi \left(\frac{\xi_{k-1} - \xi_k^*}{\lambda_{\xi,k}} \right)}. \end{aligned}$$

For ξ_1 between $(-\infty, \xi_2)$ this simplifies to:

$$\frac{q(\xi_1^*, \xi_1)}{q(\xi_1, \xi_1^*)} = \frac{\Phi \left(\frac{\xi_2 - \xi_1}{\lambda_{\xi,1}} \right)}{\Phi \left(\frac{\xi_2 - \xi_1^*}{\lambda_{\xi,1}} \right)}.$$

For ξ_2 between (ξ_1, ξ_3) the ratio is:

$$\frac{q(\xi_2^*, \xi_2)}{q(\xi_2, \xi_2^*)} = \frac{\Phi \left(\frac{\xi_3 - \xi_2}{\lambda_{\xi,2}} \right) - \Phi \left(\frac{\xi_1 - \xi_2}{\lambda_{\xi,2}} \right)}{\Phi \left(\frac{\xi_3 - \xi_2^*}{\lambda_{\xi,2}} \right) - \Phi \left(\frac{\xi_1 - \xi_2^*}{\lambda_{\xi,2}} \right)}.$$

For ξ_3 between (ξ_2, ∞) the ratio simplifies to:

$$\frac{q(\xi_3^*, \xi_3)}{q(\xi_3, \xi_3^*)} = \frac{1 - \Phi\left(\frac{\xi_2 - \xi_3}{\lambda_{\xi,3}}\right)}{1 - \Phi\left(\frac{\xi_2 - \xi_3^*}{\lambda_{\xi,3}}\right)} = \frac{\Phi\left(\frac{\xi_3 - \xi_2}{\lambda_{\xi,3}}\right)}{\Phi\left(\frac{\xi_3^* - \xi_2}{\lambda_{\xi,3}}\right)}.$$

A Normal proposal is used for $\underline{\alpha}$, centred at the previous value of α_k for each k and with standard deviation equal to a tuning parameter, i.e. $\alpha_k^* \sim N(\alpha_k, \lambda_{\alpha,k}^2)$. The proposal ratio drops out of the acceptance probability as this is symmetric. For ω , a log-Normal proposal distribution is used, with median equal to the previous value of ω and standard deviation equal to a tuning parameter, i.e. $\omega^* \sim LN(\omega, \lambda_{\omega}^2)$.

The MCMC scheme for Model 4.4 follows a Metropolis within Gibbs algorithm, to allow the intractable within-state parameters to be sampled in the same block. This process is detailed in Algorithm 4.2. A Gibbs step is used for $\underline{\nu}$ and $\underline{\pi}$ as they have standard full conditional distributions.

Results

Algorithm 4.2 was run for $N = 150,000$ iterations, plus a burn in of 1,000 and was thinned by 15. The hyperparameters were chosen to reflect similar prior assumptions about the data as the models fitted so far: a $Ga(1, 1)$ prior is assigned to ω ; the location parameters ξ_k are given a prior mean of $m_{\xi} = 0.1$, approximately the mean expected for BFEN data, and standard deviation $s_{\xi} = 0.5$, approximately two standard deviations of the BFEN data; the scale parameter was given a $N(1, 3^2)$ prior, to reflect prior ignorance about the skewness of the within-state distributions.

The tuning parameters for each of the proposal distributions were: $\underline{\lambda}_{\xi} = (0.01, 0.1, 0.05)$, resulting in acceptance rates of $(0.017, 0.019, 0.068)$, respectively; $\underline{\lambda}_{\alpha} = (2, 0.65, 0.4)$, resulting in acceptance rates of $(0.62, 0.043, 0.075)$, respectively; and $\lambda_{\omega} = 0.1$, resulting in an acceptance rate of 0.090. These rates are all lower than the 10% lower bound for an ideal MH acceptance rate, with the exception of the rate for α_1 which is higher and the rate for ω which is borderline, however because Algorithm 4.2 was run for long enough, convergence of each of the parameters was reached and the trace plots of each of the parameters show signs of mixing well.

The posterior within-state densities using the skew Normal distribution, Fig. 4.9, show that the lowest activity level (blue) has a much closer fit to the density of the data with the additional scale parameter. The posterior density for α_1 has a mean of 35.58, emphasising a strong positive skew to activity zone one, which is seen in the positively skewed data overall. The posterior within-state densities for the other two activity zones are similar to the Normal within-state densities in Model 4.3, with the posterior density for the shape parameter for these states centring about zero.

Algorithm 4.2: Skew Normal HMM MCMC scheme

1. Initialise $\xi_k^{(0)}$, $\alpha_k^{(0)}$, $\omega^{(0)}$, $\underline{\nu}^{(0)}$ and $\underline{\pi}_k^{(0)}$, for all k , by sampling from prior distributions in Model 4.4;
2. **for** $i = 1, \dots, N$:
3. run Algorithm 3.4 to sample the hidden states $\underline{z}^{(i)}$;
4. draw $\underline{\pi}_k^{(i)}$ from (3.9) for $k = 1, \dots, 3$;
5. draw $\underline{\nu}^{(i)}$ from (3.10);
6. **for** $\xi_k^{(i)}$, $k = 1, \dots, 3$:
7. generate a proposal ξ_k^* from $q(\xi_k^{(i-1)}, \xi_k^*)$, truncated on $(\xi_{k-1}^{(i)}, \xi_{k+1}^{(i-1)})$ where $\xi_0^{(i)} = -\infty$ and $\xi_4^{(i-1)} = \infty$ for all i ;
8. evaluate the acceptance probability

$$A(\xi_k^* | \xi_k^{(i-1)}) = \min \left(1, \frac{\pi(\xi_k^* | \cdot) q(\xi_k^*, \xi_k^{(i-1)})}{\pi(\xi_k | \cdot) q(\xi_k^{(i-1)}, \xi_k^*)} \right);$$
9. generate an independent random variable $u \sim U(0, 1)$;
10. **if** $u \leq A(\xi_k^* | \xi_k^{(i-1)})$ **then** set $\xi_k^{(i)} = \xi_k^*$ **else** set $\xi_k^{(i)} = \xi_k^{(i-1)}$;
11. **end**
12. **for** $\alpha_k^{(i)}$, $k = 1, \dots, 3$:
13. generate a proposal for α_k^* from $q(\alpha_k^{(i-1)}, \alpha_k^*)$;
14. evaluate the acceptance probability

$$A(\alpha_k^* | \alpha_k^{(i-1)}) = \min \left(1, \frac{\pi(\alpha_k^* | \cdot)}{\pi(\alpha_k | \cdot)} \right);$$
15. generate an independent random variable $u \sim U(0, 1)$;
16. **if** $u \leq A(\alpha_k^* | \alpha_k^{(i-1)})$ **then** set $\alpha_k^{(i)} = \alpha_k^*$ **else** set $\alpha_k^{(i)} = \alpha_k^{(i-1)}$;
17. **end**
18. generate a proposal for ω^* from $q(\omega^{(i-1)}, \omega^*)$;
19. evaluate the acceptance probability

$$A(\omega^* | \omega^{(i-1)}) = \min \left(1, \frac{\pi(\omega^* | \cdot) q(\omega^*, \omega^{(i-1)})}{\pi(\omega | \cdot) q(\omega^{(i-1)}, \omega^*)} \right);$$
20. generate an independent random variable $u \sim U(0, 1)$;
21. **if** $u \leq A(\omega^* | \omega^{(i-1)})$ **then** set $\omega^{(i)} = \omega^*$ **else** set $\omega^{(i)} = \omega^{(i-1)}$;
22. **end**

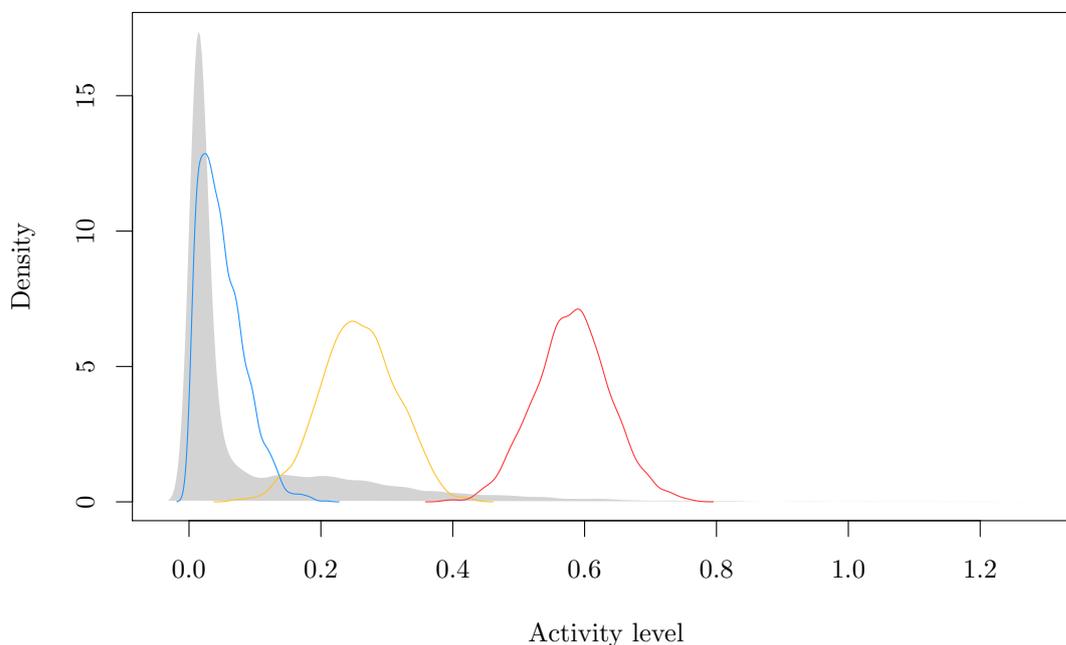


Figure 4.9: Skew Normal HMM, with shared within-state scale parameter, posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to 1 Hz BFEN data, compared to the actual BFEN density (grey).

Whilst the fit of the posterior within-state densities has improved with the additional degree of freedom from the shape parameter, the resulting modal state classifications in Fig. 4.10 are, on the whole, the same as those from Model 4.3. The mean number of transitions between states is also very similar at 742. These results show that two candidate models should be considered for classifying the activity observations into activity zones, Model 4.3 and Model 4.4. How necessary the shape parameter in Model 4.4 is for correctly identifying the activity zones will be assessed in the following sections.

The main drawback of the current way the models are being fit is how computationally expensive the parameter estimation is. The models so far have used only 1/6th of the data (10,000 observations) and have taken approximately one week to complete. Running the processes with larger segments or all of the data is not realistically going to yield any results in a time frame that would be suitable for real-time glucose prediction, which is the main aim of the thesis. Parameters at least need to be estimated within a day to perform model forecasts. Since the volume of 1 Hz activity data is large, using a smaller portion of it should still be effective in estimating model parameters so long as the portion of data used is rich enough, but faster methods are needed. One of the potential routes to achieving this speed-up is to use a faster programming language. Alternatively, data sets can be downsampled to lower frequencies before models are fitted.

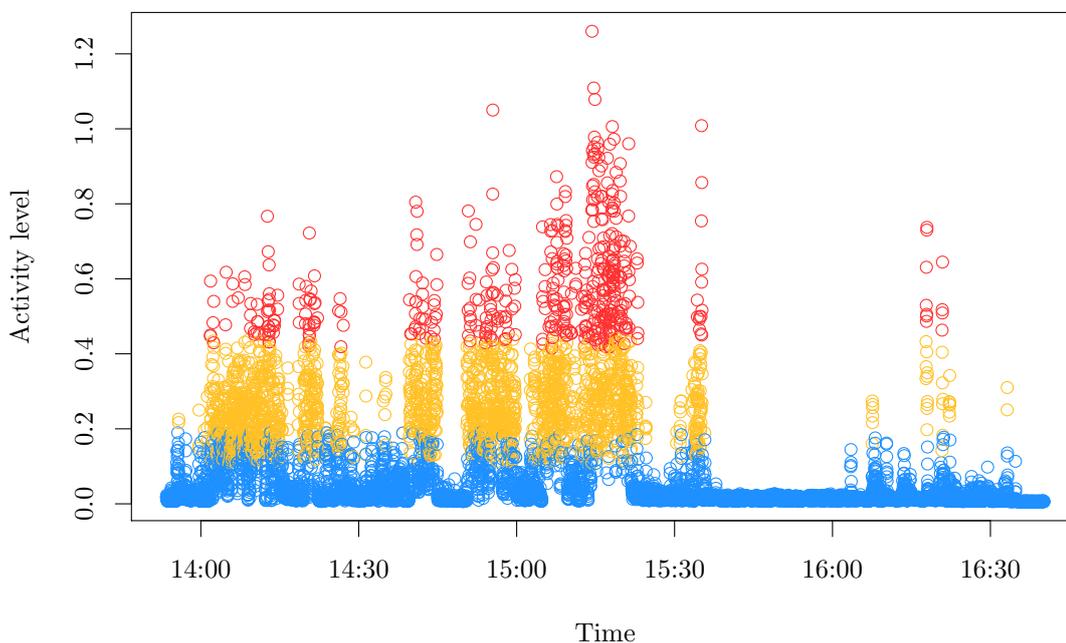


Figure 4.10: 1 Hz BFEN data classified using a skew Normal HMM with shared within-state scale parameter into low (blue), medium (gold) and high (red) levels of activity.

4.4 Downsampling activity data

The HMMs fitted so far show a good initial classification of activity data summarised using the BFEN algorithm into three distinct levels. The computationally expensive MCMC process for a subset of data points is, however, impractical. The simplest way to overcome this drawback is to reduce the size of the activity data sets; this can easily be done by implementing a larger window to aggregate over the transformed data in each of the algorithms discussed in Chapter 2. The activity summary transformations considered are the ENMO algorithm, BFEN algorithm and step count algorithm. Further HMMs are fitted to these activity summaries aggregated over five minute intervals: additional to reducing the overall size of the activity data sets, this choice of window aligns the frequency of the activity data with that of the glucose data for future joint modelling. Normal and skew Normal within-state distributions are again considered, along with raw and transformed activity summaries. In this section, a Normal HMM refers to Model 4.3 and a skew Normal HMM refers to Model 4.4. These labels are used interchangeably. The prior hyperparameters of the hidden parameters for both models are

$$\underline{\gamma} = (0.95, 0.03, 0.02), \quad \underline{\alpha} = \begin{pmatrix} 10 & 1 & 1 \\ 1 & 3 & 1 \\ 1 & 1 & 2 \end{pmatrix};$$

the same prior on the initial probability vector as for the models fitted to the 1 Hz activity summaries, but less concentrated priors on the probabilities in the transition matrix. The prior hyperparameters for the observation parameters are specified in each section. They are chosen to represent prior ignorance about the within-state distributions.

4.4.1 ENMO HMMs

The ENMO activity data feature negative observations and are positively skewed with long tails. The 1 Hz ENMO data alongside the same ENMO-transformed accelerometer data aggregated over five minute intervals are shown in Fig. 4.11. Aggregating the data over a larger window has smoothed over the activity profile and has resulted in the periods of rest and activity becoming more prominent in comparison to the 1 Hz data. The downsampled data still contain some negative observations, so a log or square root transform cannot be applied to reduce the long tails. The skew Normal within-state HMM is therefore a good candidate for capturing the long tails and classifying the activity observations into activity zones. Both Models 4.3 and 4.4 are fitted to these data for comparison. A HMM with three hidden components still seems suitable for these data, as there appear to be three bands of observations in the data: an at-rest component, a medium-level component, and a high-level component to explain the large spikes.

The hyperparameters used for the Normal HMM are: $\mu_0 = 500$, $\sigma = 30$, $\alpha = 1$ and $\beta = 1$ and for the skew Normal HMM are: $m_\xi = 500$, $s_\xi = 30$, $m_\alpha = 1$, $s_\alpha = 4$, $\alpha = 1$ and $\beta = 1$. The block Gibbs samplers were run for $N = 200,000$ iterations, with additional burn in of 1,000 and a thin of 15 and $N = 150,000$ iterations, with additional burn in of 1,000 and a thin of 12, respectively.

The resulting posterior within-state densities from fitting Model 4.3 (solid lines) and Model 4.4 (dashed lines) are in Fig. 4.12. The means of the two lower activity zones for the Normal HMM are quite close together, with the second activity level centred around the small mode at 500. The third activity zone is centred around 1,000, covering the next highest modes in the data and most of the long tail for the higher activity observations, but has a considerable amount of density overlapping the two lower states. As a result of this, the second zone of activity does not dominate any clear range of activity observations in the classification plot, Fig. 4.13 (left), with relatively few gold coloured observations being shown.

The skew Normal model performs much better for the ENMO data set. The posterior locations of the hidden states are more spread out, at sensible points along the range of activity levels. The lowest activity within-state density comes down before the second mode in the data, better differentiating between the observations relating to low and no activity. As the zone one within-state density better captures the shape of the dominant low activity zone with a single component, the other two components are not also trying to

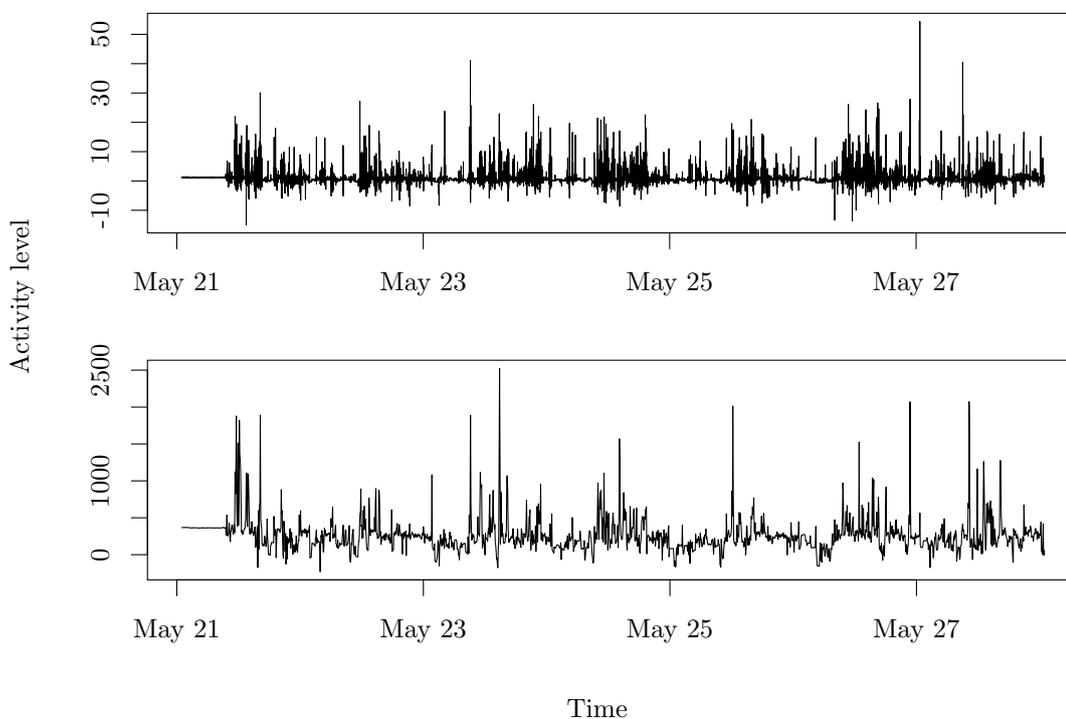


Figure 4.11: Accelerometer data summarised using the ENMO algorithm aggregated over one second intervals (top – thinned by 20 for plotting) and aggregated over five minute intervals (bottom).

improve the fit of the first component, like in the Normal HMM. The position of the highest activity state covers more of the low density range of high activity level observations. Despite the better fit to the data in terms of interpretable activity zones, the posterior skewness parameter, α_k , is only different from zero for the lowest activity state, with a mean of 0.78, which represents only a small level of positive skew. This is confirmed by the bell-shaped curves in Fig. 4.12. This small improvement to the fit of the lowest activity component has had a big, positive impact on the fit of the other components.

The classified observations in Fig. 4.13 (right) show three clear bands of activity, with only the highest observations being classified as activity zone three. The classification of each observation and the boundaries between each band of activity seem sensible. With the downsampled frequency of the data, the three activity levels that have emerged from fitting Model 4.4 are at rest (zone one), light activity (zone two) and higher intensity or prolonged activity (zone three). Overall, Model 4.4 fits well to the ENMO data.

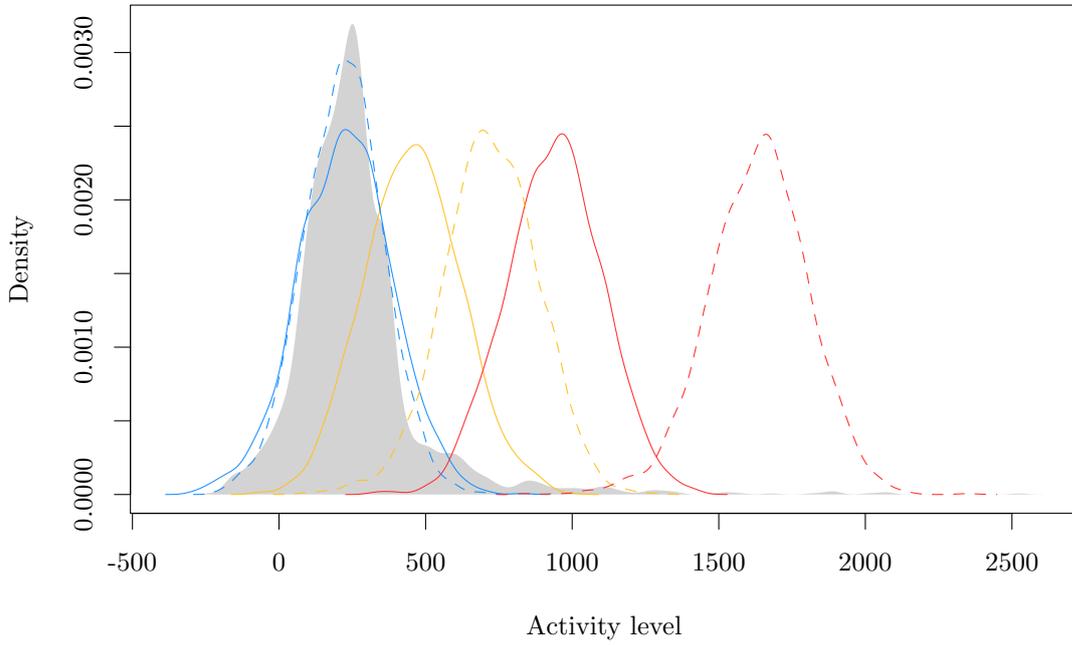


Figure 4.12: Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to ENMO data, compared to the actual ENMO density (grey).

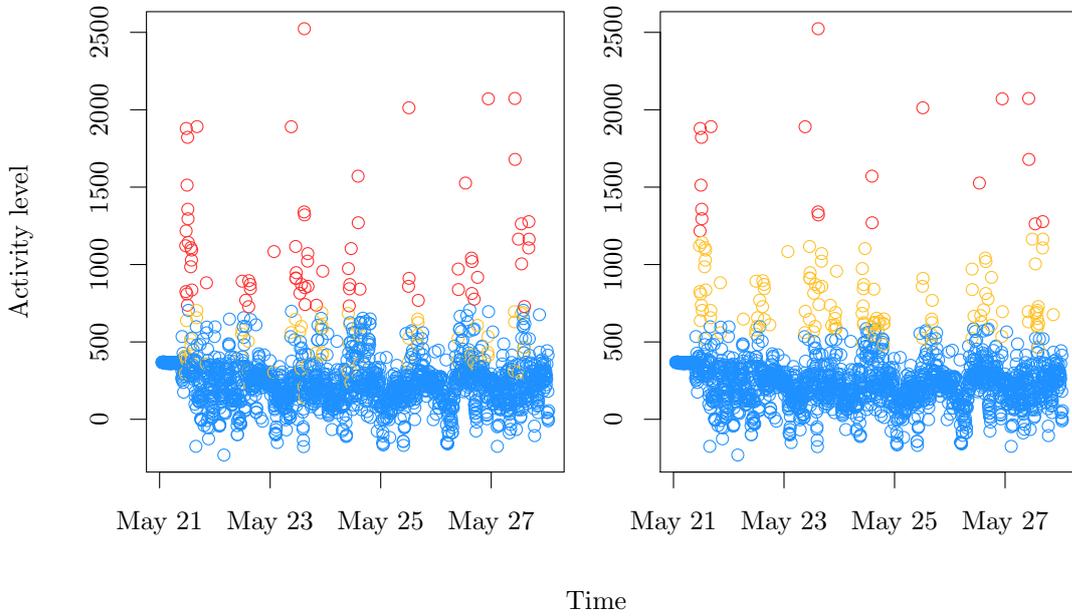


Figure 4.13: ENMO data classified using a Normal HMM (left) and a skew Normal HMM (right) into low (blue), medium (gold) and high (red) levels of activity.

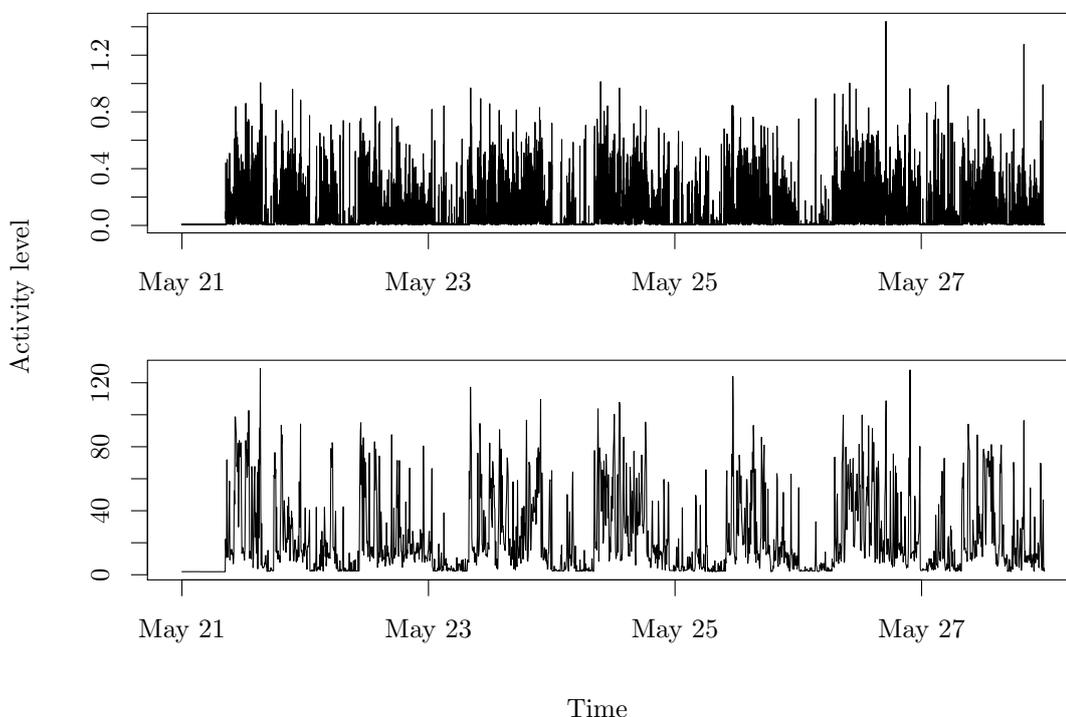


Figure 4.14: Accelerometer data summarised using the BFEN algorithm, aggregated over one second intervals (top – thinned by 20 for plotting) and aggregated over five minute intervals (bottom).

4.4.2 BFEN HMMs

Transforming the accelerometer data with the BFEN algorithm and a window size of five minutes has the same impact of smoothing over the 1 Hz activity profile as is seen in the ENMO summary, whilst accentuating the points in the data when the patient is active or at rest. The one second and five minute aggregations are seen in Fig. 4.14.

To reduce the skewness in the BFEN density, depicted in grey in Fig. 4.15, a transformation can be applied. Since the observations in the BFEN summary are strictly positive (> 0), a log-transform is used and this transformed density is shown in grey in Fig. 4.16. The transformed data show three clear activity components, with what looks like a mixture of three Normal densities. This is evidence to suggest the Normal HMM may be sufficient in modelling the transformed data.

The hyperparameters used for the Normal HMM are: $\mu_0 = 10$, $\sigma = 7$, $\alpha = 1$ and $\beta = 1$ and for the skew Normal HMM are: $m_\xi = 10$, $s_\xi = 7$, $m_\alpha = 2$, $s_\alpha = 4$, $\alpha = 1$ and $\beta = 1$. The block Gibbs samplers were run for $N = 100,000$ iterations, plus a burn in of 1,000 and with a thin of 10 and $N = 500,000$ iterations, plus a burn in of 1,000 and with a thin of 30, respectively. For the log-transformed data, the hyperparameters used for the

Normal HMM are: $\mu_0 = 2$, $\sigma = 1.5$, $\alpha = 1$ and $\beta = 1$ and for the skew Normal HMM are: $m_\xi = 2$, $s_\xi = 1.5$, $m_\alpha = 0$, $s_\alpha = 3$, $\alpha = 1$ and $\beta = 1$. The block Gibbs samplers were both run for $N = 100,000$ iterations, plus a burn in of 1,000 and with a thin of 10. The MCMC schemes took considerably less time to run on the downsampled data, completing within one to two days. Typically the skew Normal model took longer than the Normal model due to the increased complexity of the model and larger number of parameters, but this was also dependent on the number of iterations. This is the case for subsequent activity models, too.

For the raw five minute BFEN data, Fig. 4.15 shows a slightly better fit to the density by the skew Normal model (Model 4.4), where the within-state density of zone one shows a positive skew and the curve comes down sharply at zero. The variance of the Normal HMM densities is slightly smaller than the scale of the skew Normal densities, with the posterior skew Normal curves appearing slightly wider and flatter than the Normal curves, but overall the two models fit quite similarly. The posterior densities of the means and locations of each state overlap. Despite the weight of the posterior within-state density for zone one with a skew Normal distribution appearing higher than the weight of the Normal within-state density, the stationary distributions of the two models are similar: the mean posterior stationary distribution for Model 4.3 is $(0.72, 0.15, 0.13)$ and for Model 4.4 is $(0.70, 0.16, 0.14)$. Looking in more detail at the posterior within-state densities in Fig. 4.16 for the log-transformed data, the two models again fit very similarly. The skew Normal model shows a small negative skew for zone one (blue) and zone three (red) of -1.11 and -1.67 , respectively, which allows the posterior densities to better fit the tails of the data.

As Models 4.3 and 4.4 have identified three similar zones in the raw and transformed BFEN data, only the classification results for one model from each data set are compared. The skew Normal model for the raw data fits slightly better than the Normal data, whilst the Normal model is sufficient in describing the log-transformed data. The classifications of the raw observations according to each of these model results are shown in Fig. 4.17.

The raw BFEN classification plot, Fig. 4.17 (left), shows three clear, equal-width bands of activity, which look similar to the results achieved by the models already fitted in this chapter. The classification plot for the log-transformed data, however, has achieved something different: fitting the model to the transformed data and plotting the results on the raw data has a similar effect as if the variance between the states was allowed to differ, without the identifiability problems. The resulting posterior zones of activity are more reflective of the interpretable zones of activity that were detected when fitting the mixture model, Model 4.1: complete inactivity in zone one, low activity in zone two, i.e. the patient is awake but is not moving around much, and ‘active’ in zone three, i.e. the patient is moving around and this may involve walking. The mean number of transitions between the states here is 486, which, over a period of one week, seems reasonable. For the

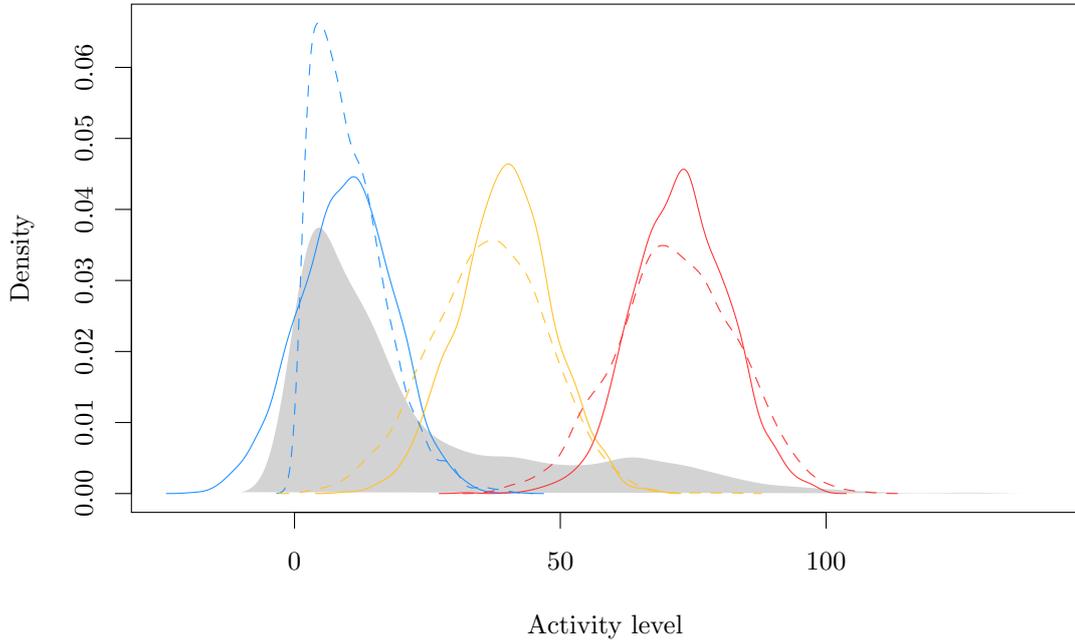


Figure 4.15: Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to BFEN data, compared to the actual BFEN density (grey).

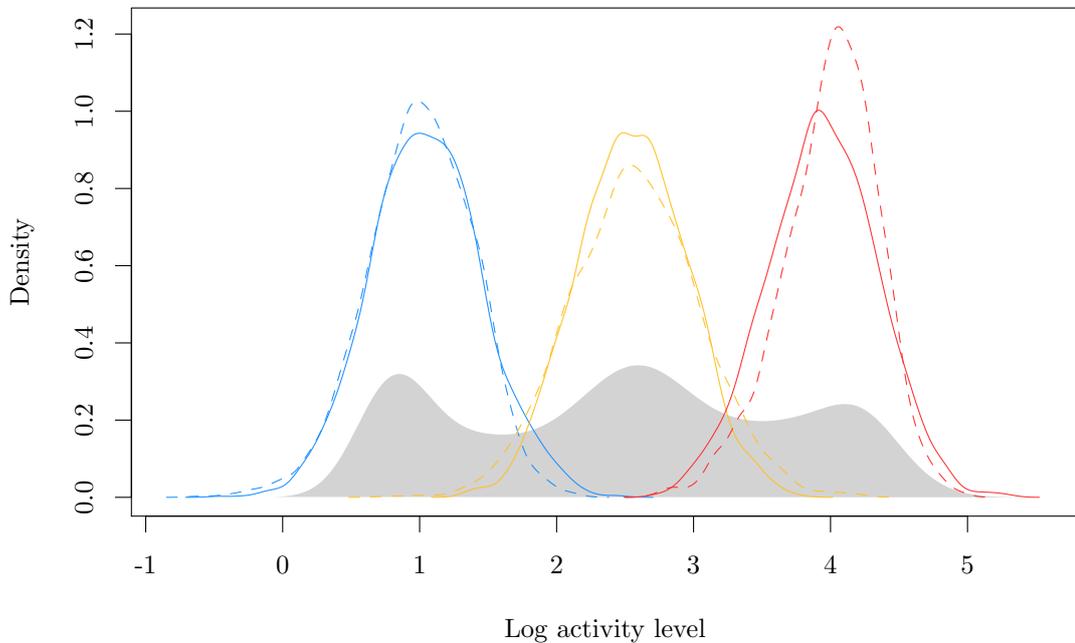


Figure 4.16: Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to log BFEN data, compared to the actual log BFEN density (grey).

skew Normal model on raw data, the mean number of transitions is similar but slightly lower at 463.

To determine which combination of fitted model and activity data summary displays the greatest agreement, the MSEs of the one-step-ahead forecasts are compared. The MSE here is defined for each time point as the squared deviation between the observation at time t and a draw from its one-step-ahead forecast distribution, averaged across MCMC iterations. As the scale of the MSEs cannot directly be compared for the raw and log-transformed results, the MSEs are normalised by dividing by the square of the mean of the corresponding activity summary (the raw or log-transformed data) and are shown in Fig. 4.18. Normalising the MSEs in this way is equivalent to computing the squared coefficient of variation of the errors, which is a standardised measure of the dispersion of each of the model results.¹ Other methods for normalising the MSEs include dividing by the range of the MSEs or the variance. Most notable for the two summaries is the long tails that are present in the distribution of errors for the raw data compared to the log-transformed data. The distribution of MSEs for the log-transformed data, for both models, is much tighter and on average the errors are lower. The Normal HMM and the skew Normal HMM have similar MSE distributions for the log-transformed and raw data, confirming the similarity in performance of the models for each activity summary. Overall, Model 4.3 being fitted to the log-transformed data is the preferred model and summary; the model produces the lowest average MSE and explains the hidden states of the underlying activity density well, offering a classification that can be easily interpreted, too. The simpler within-state distribution also leads to a more straightforward MCMC scheme, which is why this model is chosen over the skew Normal model applied to the log-transformed data, that provides similar results in all aspects of the analysis.

4.4.3 Step count HMMs

The step count data aggregated over one second intervals are discrete in their nature and it is therefore unsuitable to describe the activity zones for these data with continuous within-state distributions. Fig. 4.19 shows the step count data aggregated over five minute intervals (bottom) compared to the one second data. The aggregation over a larger window brings about a large enough range of observations for the data to be considered continuous, which can be seen in the comparative figure. This means an attempt at describing the step count data with the current models, Model 4.3 and Model 4.4, is possible. The shape of the step count summary is similar to that of the BFEN five minute summary, with a clear distinction between rest and active periods: there are flat periods in the profile that coincide with night time when the patient is asleep.

As with the BFEN data, a transformation can be applied to the step count data to

¹<https://stats.idre.ucla.edu/stata/ado/analysis/>

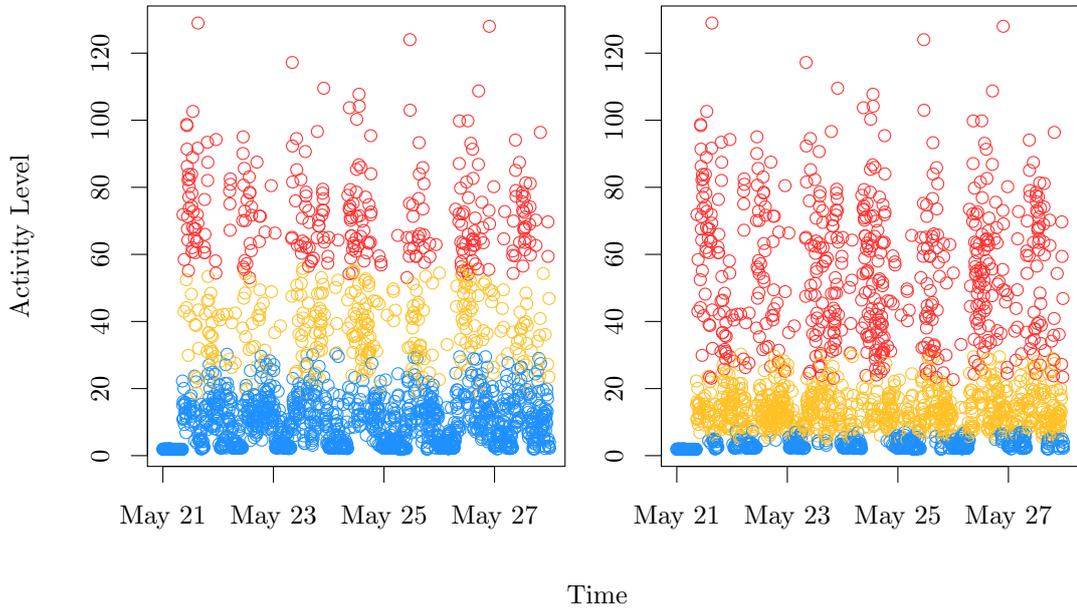


Figure 4.17: Raw BFEN data classified using a skew Normal HMM (left) and log-transformed BFEN data classified using a Normal HMM (right) into low (blue), medium (gold) and high (red) levels of activity.

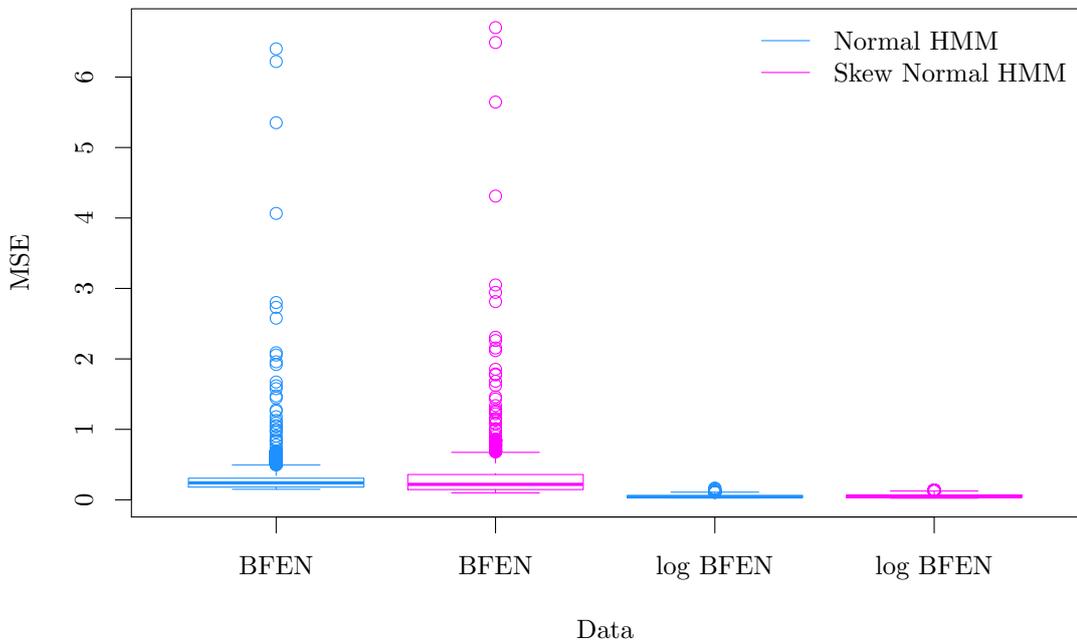


Figure 4.18: Boxplots of the normalised one-step-ahead forecast MSEs for the skew Normal HMM fitted to the raw BFEN data and the Normal HMM fitted to the log-transformed BFEN data.

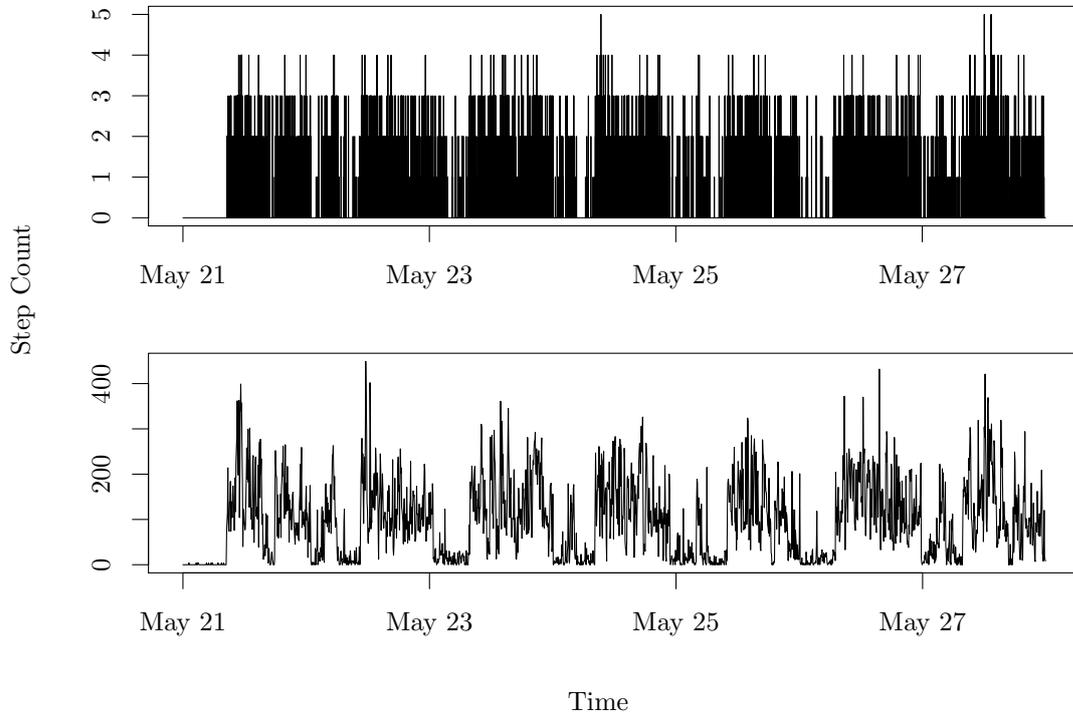


Figure 4.19: Accelerometer data summarised using the complete step count algorithm, aggregated over one second intervals (top – thinned by 20 for plotting) and aggregated over five minute intervals (bottom).

reduce their positive skew. As there are several zero-observations in these data, a square root transformation is applied. The two models are applied to both sets of data.

The hyperparameters used for the Normal HMM are: $\mu_0 = 100$, $\sigma = 10$, $\alpha = 1$ and $\beta = 1$ and for the skew Normal HMM are: $m_\xi = 100$, $s_\xi = 10$, $m_\alpha = 2$, $s_\alpha = 4$, $\alpha = 1$ and $\beta = 1$. The block Gibbs samplers were run for $N = 100,000$ iterations, with additional burn in of 1,000 and a thin of 10 and $N = 150,000$ iterations, with additional burn in of 1,000 and a thin of 12, respectively. For the square-rooted data, the hyperparameters used for the Normal HMM are: $\mu_0 = 10$, $\sigma = 5$, $\alpha = 1$ and $\beta = 1$ and for the skew Normal HMM are: $m_\xi = 10$, $s_\xi = 5$, $m_\alpha = 1$, $s_\alpha = 3$, $\alpha = 1$ and $\beta = 1$. The block Gibbs samplers were run for $N = 100,000$ iterations, plus a burn in of 1,000 and a thin of 10 and $N = 500,000$ iterations, plus a burn in of 1,000 and a thin of 50, respectively.

Fitting Model 4.3 and Model 4.4 to the raw step count data produces the posterior within-state densities in Fig. 4.20. Once again, the models perform similarly, detecting hidden states in the same sections of the raw data. The skew Normal model offers a tighter fit to the lowest activity zone by adding a strong positive skew to this state. As with the BFEN data, the posterior stationary distributions for the raw step count summary

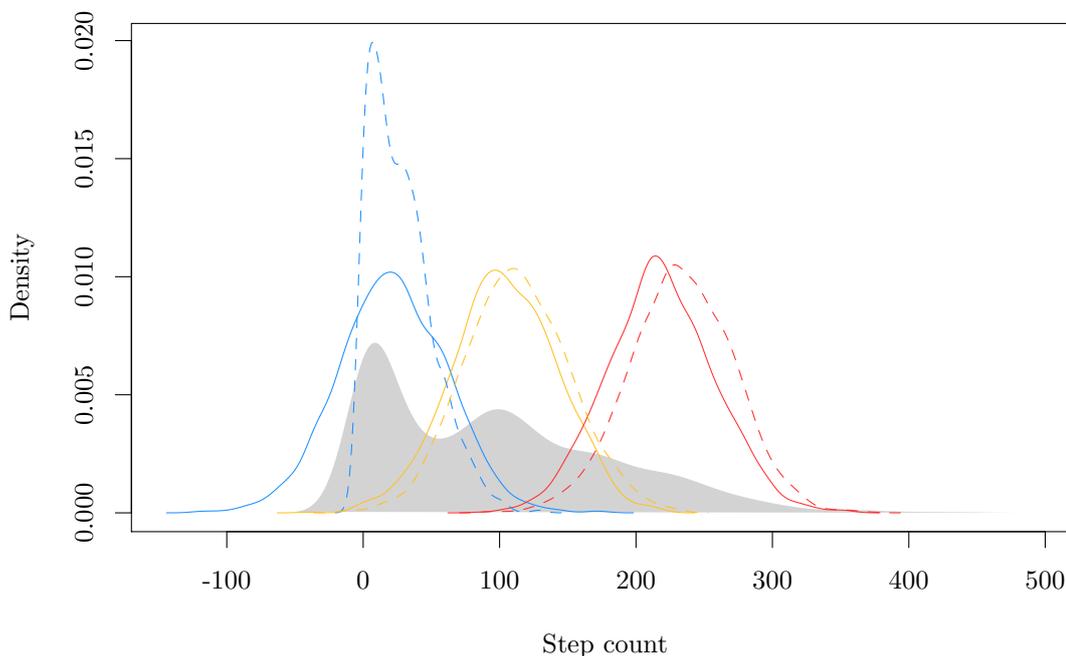


Figure 4.20: Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to step count data, compared to the actual step count density (grey).

are similar for both models, despite the weights of the posterior within-state densities appearing to be slightly different. The posterior means for the stationary distributions are $(0.40, 0.40, 0.20)$ for the Normal within-state HMM and $(0.40, 0.43, 0.17)$ for the skew Normal HMM.

The posterior within-state densities from fitting the models to the square-rooted data are shown in Fig. 4.21. After applying the transformation to the raw step count data, the long tail has been reduced, but there appear to be up to four underlying states, represented by the four peaks that can be seen in the grey density. As a result of the additional mode, there are some identifiability issues that arise when fitting a three-state model to the data: the posterior density for the location parameter of zone two in the skew Normal HMM results is bimodal, sampling from densities that coincide with the two central modes, and zone one and zone three also cover the two lower and two upper modes, respectively. This leads to a bimodal posterior stationary distribution too. The Normal HMM performs better on this data set, but models with $K = 4$ hidden states should be explored to see if this improves the fit and performance. The fourth peak in the transformed data is a possible side effect of working with data that come from a discrete state space.

The models fitted to the raw step count data and the Normal HMM fitted to the square-rooted step count data all produce similar classification plots, two of which are seen in

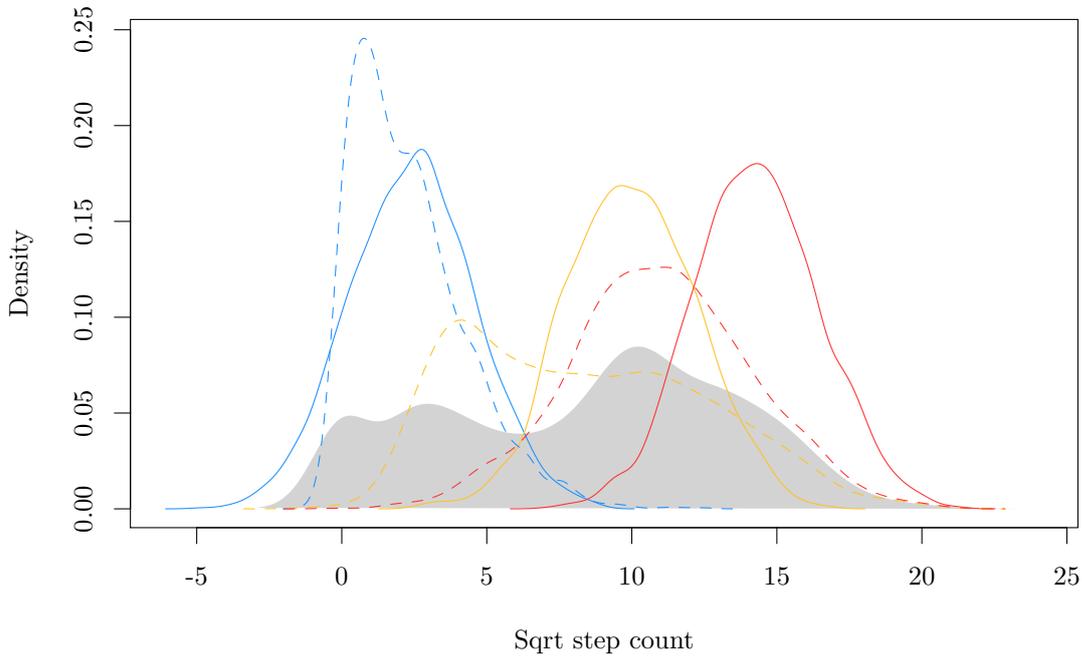


Figure 4.21: Skew Normal HMM (dashed) and Normal HMM (solid) posterior empirical within-state densities for low (blue), medium (gold) and high (red) levels of activity fitted to sqrt step count data, compared to the actual sqrt step count density (grey).

Fig. 4.22 for comparison. The main difference between the results for these three models is the boundaries between the activity zones. For the skew Normal raw data classification (Fig. 4.22, left) and the Normal classification (not shown), the boundary between zones one and two is more gradual, with a clearer line between the upper two activity zones. For the Normal HMM fitted to the square-rooted data, Fig. 4.22 (right), the opposite is true. This is based on how sharply the overlapping curves of the posterior within-state distributions decrease to (near) zero. The skew Normal HMM fitted to the square-rooted data is not shown, but the classification is similar to that of the Normal HMM on the ENMO data, Fig. 4.13, left: there are very few zone two activity observations due to the dominance of the other two states across the data.

Detecting $K = 4$ states

The transformed step count data, with density shown in Fig. 4.23 (grey), have up to four modes. Consequently, the posterior within-state densities for these data could be improved. Model 4.3 and Model 4.4 are fitted to these data with $K = 4$ hidden states to see if this improves the fit and to test the flexibility of the models being tested.

The model hyperparameters, number of iterations, burn in and thin are the same as for the three-state square-rooted data, however the priors on the initial distribution and

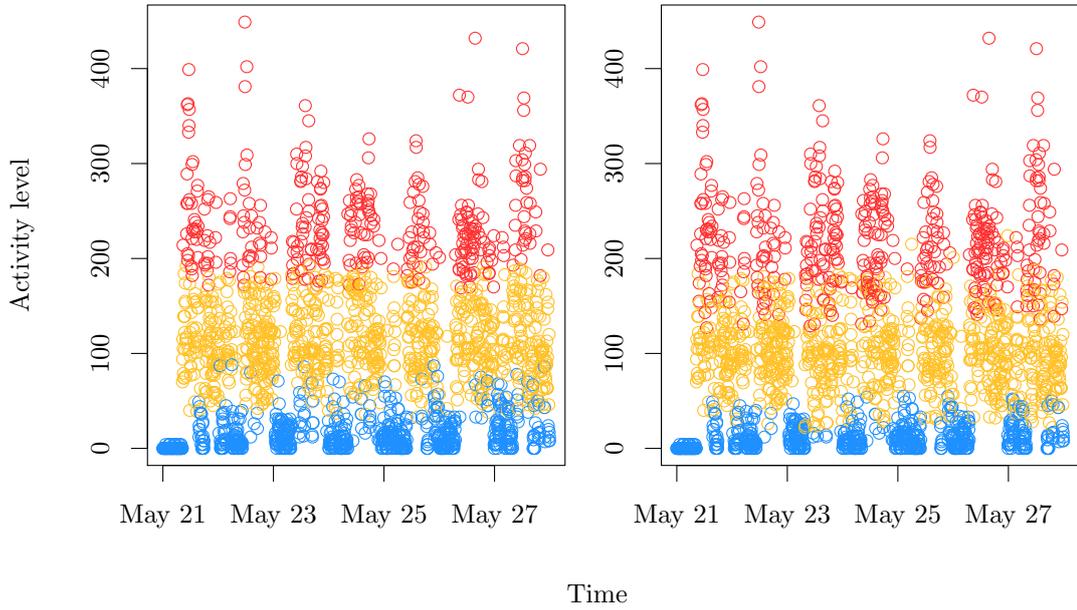


Figure 4.22: Step count data classified using a skew Normal HMM (left) and square-rooted step count data classified using a Normal HMM (right) into low (blue), medium (gold) and high (red) levels of activity.

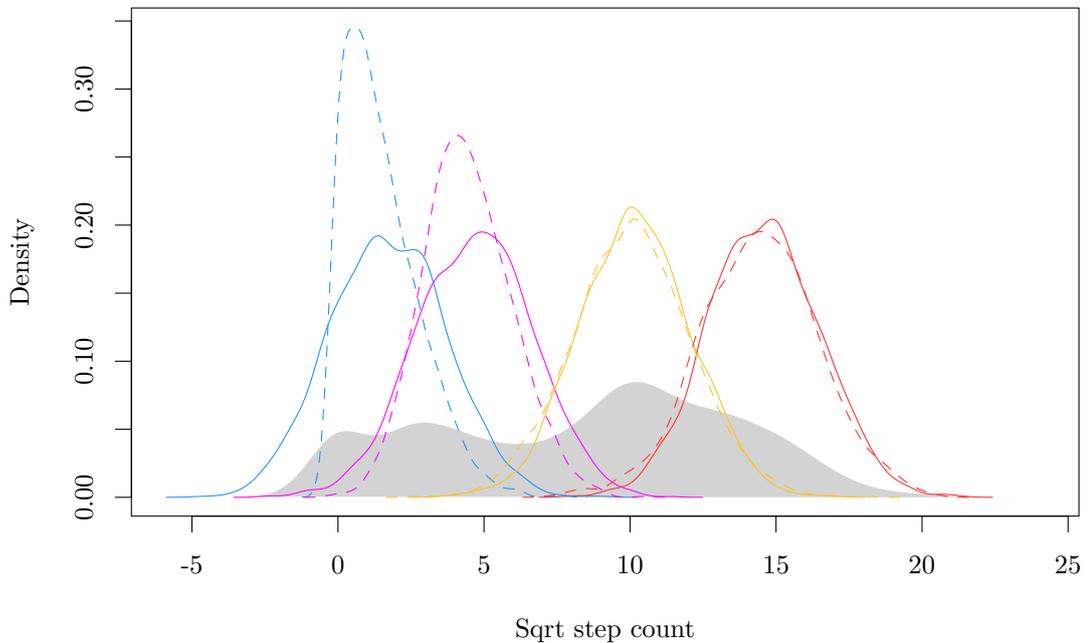


Figure 4.23: Normal HMM posterior empirical within-state densities (solid) and skew Normal empirical within-state densities (dashed) for low (blue) to high (red) activity, through two intermediate activity zones (magenta and gold) fitted to the square-rooted step count data.

transition matrix are adapted to account for the additional state. Their hyperparameters are:

$$\underline{\gamma} = (0.8, 0.15, 0.03, 0.02), \quad \underline{\alpha} = \begin{pmatrix} 8 & 1 & 1 & 1 \\ 1 & 4 & 1 & 1 \\ 1 & 1 & 3 & 1 \\ 1 & 1 & 1 & 2 \end{pmatrix}.$$

The posterior within-state densities in Fig. 4.23 after fitting the Normal and skew Normal HMMs show a much improved fit to the actual density of the data, compared to the three-state models in Fig. 4.21. The four distinct modes have been identified as four distinct hidden states by both models. The skew Normal model shows an improved fit to the lowest hidden state (blue, dashed) by assigning a positive skew to the distribution of this component so that the curve comes down sharply at zero where the step count data begin. The posterior mean stationary distributions for both models are similar, at $(0.24, 0.14, 0.40, 0.22)$ and $(0.20, 0.16, 0.41, 0.23)$ for the Normal and skew Normal models, respectively.

In terms of classification performance, the two models perform very similarly, so only the skew Normal HMM classification results are shown in Fig. 4.24. There are clear boundaries between each activity component and the location of these boundaries seems sensible for each of the states.

Comparing the overall performance of the three-state and four-state models, the normalised one-step-ahead MSE distributions are presented in Fig. 4.25. The figure shows that although the models fitted to the raw data (first two boxplots) visually fit well to the data, they have a higher MSE and long tails of higher errors. The models applied to the square-rooted data perform similarly, with the exception of the three-state skew Normal model that classified most points into two states. Although the four-state models provide within-state posterior densities that fit well to the data, they do not offer much of an improvement in terms of model performance for classification or for lower errors. Comparing Fig. 4.24 to Fig. 4.22 (right), the two lower activity zones (blue and magenta) are equivalent to the lowest activity zone (blue). As the three-state Normal model applied to the square-rooted data provides a good solution to classifying the step count data, this is the model that is chosen. If, however, a three-state model is not sufficient for modelling other patient data sets, the four-state model can be utilised.

4.4.4 Comparison of results

The models developed in Section 4.3 have provided effective methods for classifying and identifying the underlying hidden states of the activity summaries presented. In this section, the ‘best’ models chosen for each of the activity summaries are compared: the ENMO data classified by fitting a skew Normal HMM, Model 4.4; the BFEN data classified

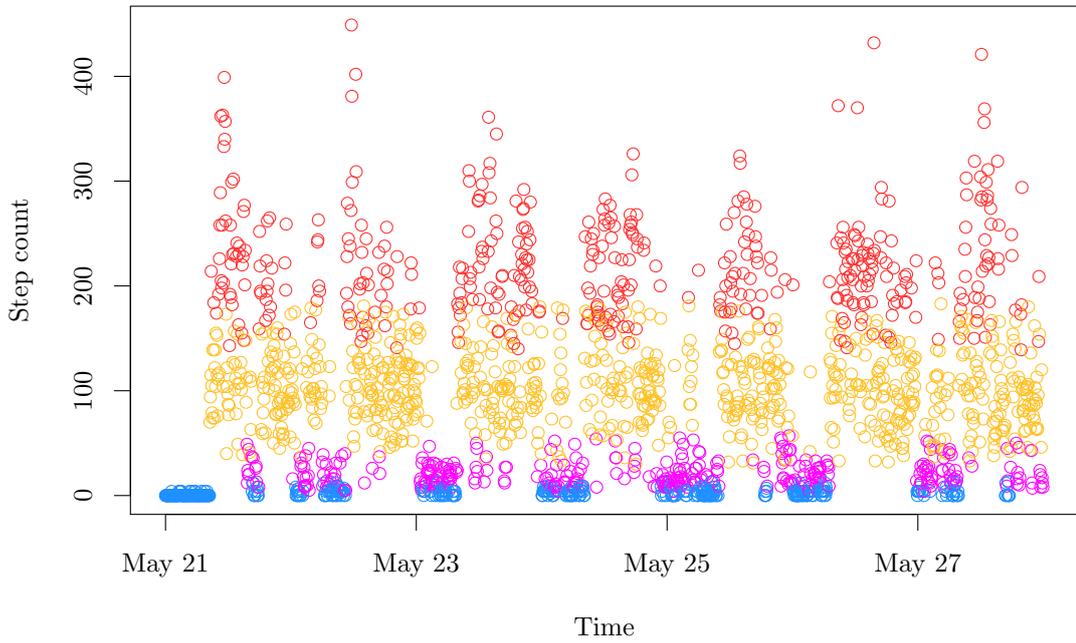


Figure 4.24: Step count data classified using a skew Normal HMM into four activity zones, from low (blue) to high (red), through two intermediate activity zones (magenta and gold) based on the square-rooted data.

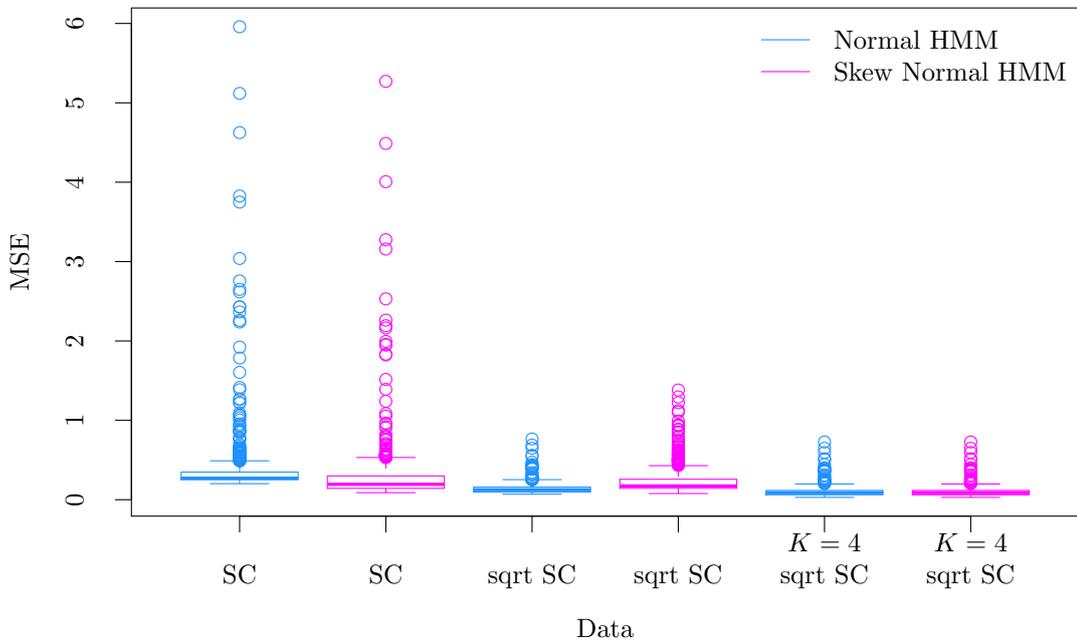


Figure 4.25: Boxplots of the normalised one-step-ahead forecast MSE for the skew Normal HMM and Normal HMM fitted to the raw step count (SC) data and square-rooted step count data, with three and four hidden activity states.

by fitting a Normal HMM, Model 4.3, to the log-transformed data; and the step count data classified by fitting a Normal HMM to the square-rooted data. How each of the classified activity summaries describes each of the hidden activity components must be assessed, as the activity zones will be used to model glucose levels in the joint models fitted in Chapter 6.

The mean MSEs produced by the one-step-ahead forecasts on each iteration of the MCMC scheme for each of the models can be compared to measure the predictive performance of each model for each activity summary. As the scale of each summary is different, the MSEs must be normalised for a direct comparison to be made, which is done by dividing by the square of the mean of the corresponding activity summary, as with the raw and transformed data sets earlier in the section. Boxplots of the mean MSEs of each data set are shown in Fig. 4.26. The distribution of MSEs for the BFEN data set covers the lowest values and has the fewest of outliers. The step count data set is second best, with a slightly higher mean and longer tail of outliers. The ENMO data set has the widest range of MSEs and the most outliers, of which some are large. There is one outlier that has been omitted from the plot for this summary, with a normalised mean MSE of 11.46. As there is such a long tail in the ENMO data set, and the within-state variance is bigger, there is more room for a more erroneous value for the one-step-ahead forecast to be sampled. As the log BFEN within-state distributions have a smaller variance, the errors produced by the one-step-ahead forecasts are relatively small.

The transition matrix that describes each HMM informs the probability of moving between each of the hidden states. The posterior densities of each of the transition probabilities for the models fitted to the step count, ENMO and BFEN data sets are shown in Fig.4.27. The first row of Fig. 4.27 corresponds to the first row of the transition matrix. For all of the activity summaries, the probability of staying in state one is very high, with tight posterior densities around high probabilities for all three summaries. The probability of moving from state one to state two is much smaller, and moving to state three is smaller again. For the ENMO data, the probability of moving to one of the other states from state one is very close to zero, whereas the probability of moving to the second activity state for the other activity summaries is slightly higher, with a close to zero probability of moving to the third state. In the second row of densities, the posterior densities for the ENMO data are wider, with approximately equal probability of moving back to the first activity zone or staying in the second zone and a higher chance of moving to zone three when the patient is in zone two than there was if the patient was in zone one. The flatter densities reflect less certainty about the posterior probabilities, which comes from the lack of data points in this state to inform the prior. For the step count and BFEN data sets, the posterior densities in this state are still tight and well informed, since there are a lot of observations to inform the prior. The densities for both of these summaries show a

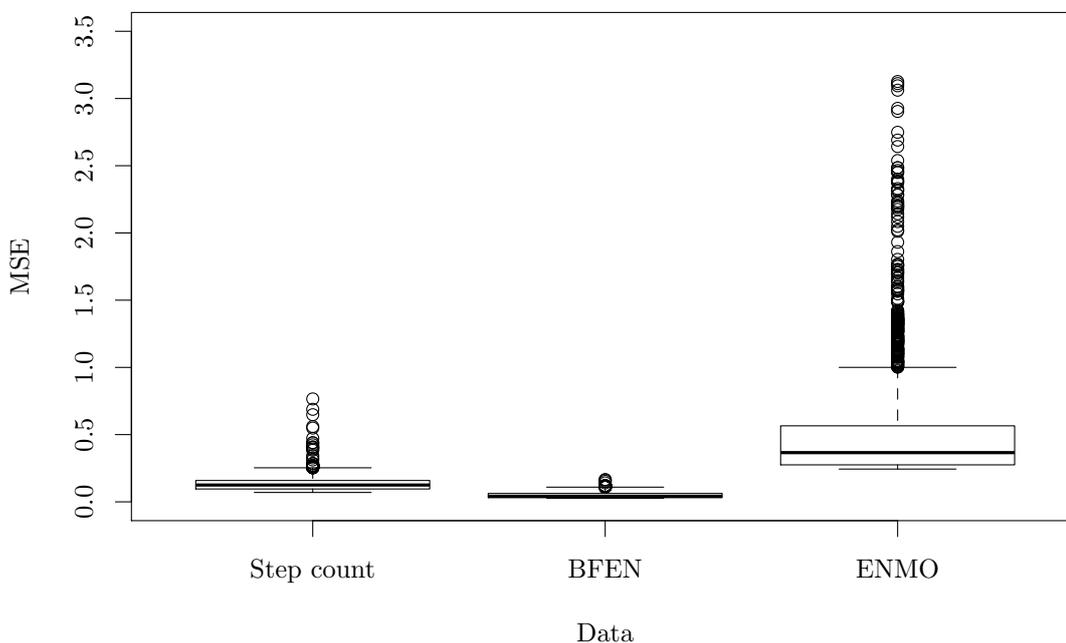


Figure 4.26: Boxplots comparing the mean MSEs of the one-step-ahead forecasts for the models fitted to each of the activity summaries. The largest outlier for the ENMO data set (11.46) has been omitted from the plot to make the comparison easier.

high probability of staying in state two, followed by the small probability of moving to state three and a small probability of moving back to state one. Finally, the third row of posterior densities shows an even flatter posterior for the ENMO data, since there are very few observations in this state. The probability of staying in state three or moving to one of the other states is approximately equal. For the step count and BFEN summaries, there is a high probability of staying in zone three and increasingly small probabilities of moving to states two or one, with the probability of moving to zone one being negligible. The BFEN and step count transition densities are similar in structure to the prior beliefs about the hidden process; there are strong probabilities on the diagonal and the probability of moving between non-adjacent states is low. The classification plots for the BFEN and step count summaries, Fig. 4.17 (right) and Fig. 4.22 (right), respectively, show that zone one observations are mainly observed at night time, so the low probability of moving to this zone from one of the other states should be small, and the probability of staying in the other two states and moving between the two is more likely. For the ENMO data, the transition probabilities generally encourage the state to regress back to the first activity zone, which is supported by the majority of observations in Fig. 4.13 (right) falling into this activity zone.

To further explore the posterior transition densities, Fig. 4.28 shows the proportion of

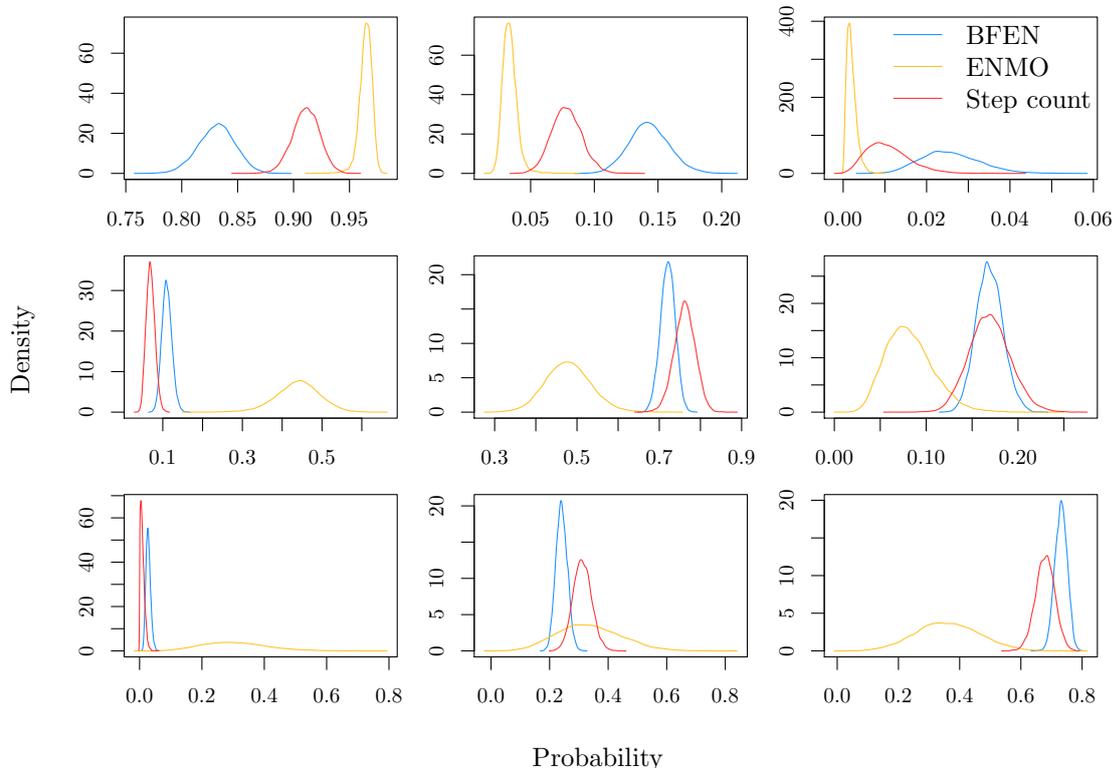


Figure 4.27: Transition matrix element-wise posterior densities for the models fitted to each of the activity summaries. The position of each plot corresponds to the element of the matrix it relates to, i.e. the first row of plots corresponds to the first row of the transition matrix.

time that Patient 3 spends in each of the activity zones according to the classification of each of the summaries. The plot shows how the BFEN and step count summaries perform similarly, falling in the centre of the ternary plot. The ENMO activity summary, however, is very different. The majority of observations in the ENMO data come from the lowest activity level, with small proportions coming from zones two and three. As is expected, the empirical proportions shown are equal to the equilibrium weights of the posterior mean stationary distributions for each of the summaries: for the BFEN summary this is $(0.31, 0.41, 0.29)$, for the step count summary the stationary distribution is $(0.35, 0.42, 0.23)$ and for the ENMO summary this is $(0.92, 0.07, 0.01)$. There is a more uniform distribution of the proportion of observations in each of the activity zones using the BFEN and step count summaries. This disparity in distribution of observations per state raises the question over which interpretation of ‘higher level’ activity is most useful.

It is likely that the observations that fall in the high level activity state for the ENMO data only involve high intensity activity, i.e. walking or running. As people that suffer from type II diabetes are often quite inactive, it could be realistic for such a small proportion of their observations to be in this zone. It could however also be an identifiability problem

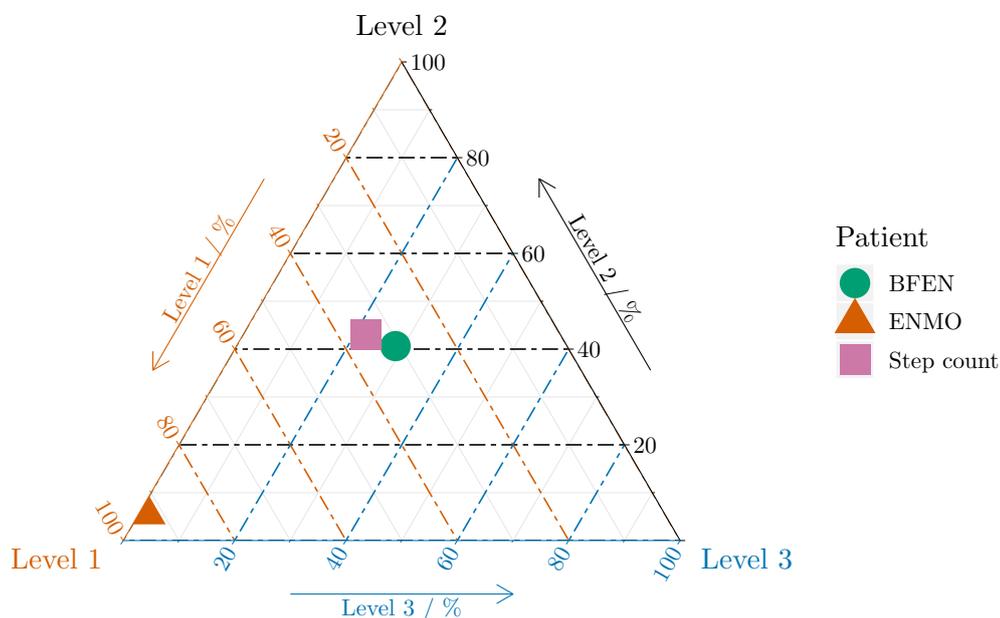


Figure 4.28: Ternary plot comparing the overall proportion of time spent in each activity zone for each activity summary.

with this activity summary, since the majority of the observations fall between zero and 500. A benefit of the BFEN and step count summaries is that they have both identified the lowest zones of activity as when the patient is sleeping, which is bound to have a different effect on glucose levels as to when the patient is awake, then the other two zones represent low activity and higher activity.

As the data for the current set of patient activity levels are unlabelled, it is difficult to know which of the activity summaries is most useful in classifying observations. To exploit the relationship between glucose levels and activity levels, it is important to represent the three zones of activity accurately, since the way that activity levels affect glucose varies between intensities. Further exploration into what activities the different zones represent would therefore be helpful. Provisionally, based on the results discussed in this section, the Normal HMM fitted to the log-transformed BFEN data is the best model to take forward to use when modelling the glucose data. The underlying activity zones are interpretable and this combination of model and activity summary has the lowest MSE across those discussed.

4.5 Across-patient comparison

The activity models so far have only been fitted to one patient. It is important to look at how well the HMMs perform on different patients' activity summaries to ensure the models developed work consistently. Additionally, one of the issues raised with the results from fitting the current models is that there is no way of checking how representative the resulting classifications are of different types of activity, not least because there are large discrepancies between the ENMO classifications and the BFEN and step count classifications. To rectify this, some extra glucose and accelerometer data have been collected on three non-diabetic people, referred to as ND 1, ND 2 and ND 3. One of these data sets, from ND 1, is partially labelled with the types of activity that are being performed throughout the day, such as when they are walking or sitting. Classifying these data and annotating the activities on top of them will help determine which activity summary is best modelled by a HMM. The data sets of ND 2 and ND 3 are not labelled, but are known to contain a range of activities, so modelling the additional data will aid in confirming the intensities of activity that are observed in the current patient data and determining the maximum activity levels expected to be observed.

In this section, the annotated accelerometer data of ND 1 are transformed using the BFEN and ENMO algorithms and classified, then the activity annotations are compared to the classified activity zones for each summary. The results from fitting HMMs to other diabetic patients and the additional non-diabetic patients are then compared to those collected so far. The hyperparameters chosen for each patient's summary are the same as for the corresponding model and summary for Patient 3 in the chapter so far. The sampling algorithms were run for a sufficient number of iterations for convergence to be achieved.

4.5.1 Comparison with annotated data

Additional data have been collected on three non-diabetic people over a ten day period. As the data for ND 1 are labelled, they are a good source of comparison for the classification results to see how observations are grouped together in the underlying activity zones of each activity summary. The accelerometer data of ND 1 are processed using the BFEN and ENMO algorithms with a five minute window. As the step count summary produces similar classifications to the BFEN summary, only the BFEN summary is looked at in this section. Model 4.3 is fitted to the log-transformed BFEN data and Model 4.4 is fitted to the ENMO data. One day of the classification results for each summary is seen in Fig. 4.29 with known activity intensity zones annotated, along with the activity labels. The intensity of the annotated activities is coloured to reflect the predominant state at the time.

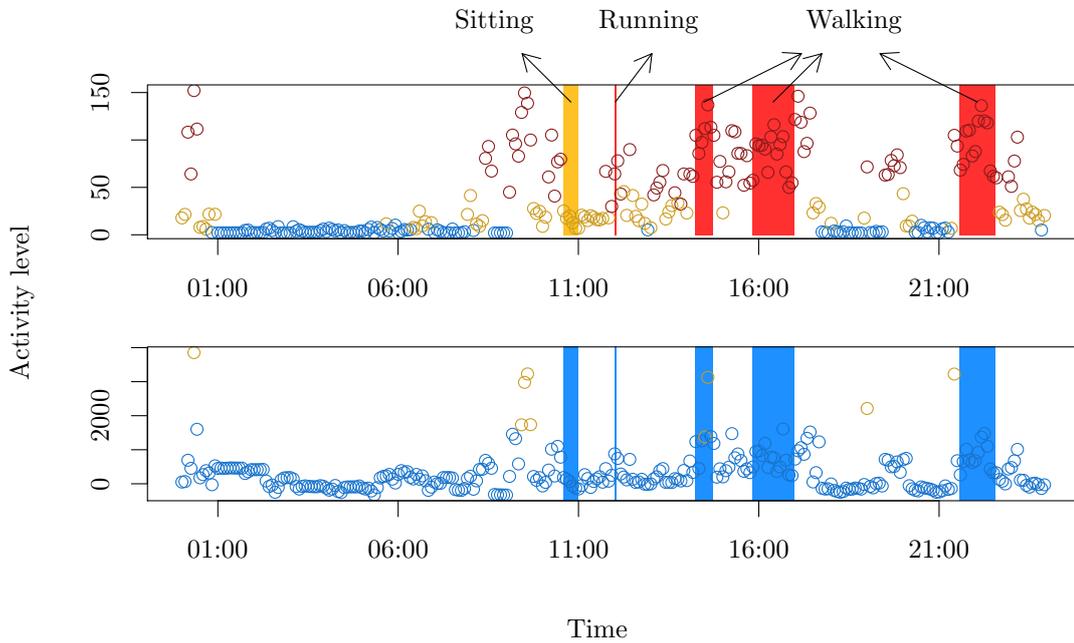


Figure 4.29: The five minute BFEN (top) and ENMO (bottom) summaries classified using Model 4.3 and Model 4.4, respectively, using ND 1's accelerometer data. Activity zones are annotated as vertical lines, with width equal to the duration of the activity.

The annotated BFEN classification, Fig. 4.29 top, confirms that the underlying activity zones should be interpreted as inactivity/sleep for zone one, sedentary/low activity for zone two and 'active' for zone three. The level one, low intensity activities, such as sitting or mindfulness sessions, therefore correspond to zone two for this classification (gold) and the other activities, such as walking and running, correspond to zone three (red). The plot shows how the empirical classification of the BFEN observations correctly 'identifies' the annotated activities. Over the full data set, the proportion of observations that match the annotations, given the interpretation of the model summary, is 84%. For the annotated ENMO classification, Fig. 4.29 bottom, the underlying states do not consistently identify the annotated activities. The summary makes no distinction between sitting and walking, and fails to detect the running event at all. As a result, the ENMO classifications and annotations only match 28% of the time. The ENMO activity classifications are therefore not as reliable or useful in identifying underlying activity zones as the BFEN activity classifications are.

In the BFEN summary, all activities that involve an increased heart rate, such as walking and running, are classed in the same activity state (zone three). In Section 1.4, evidence suggested that a distinction between the two intensities is not important, with studies finding that standing and walking have a similar impact on glucose levels, as do

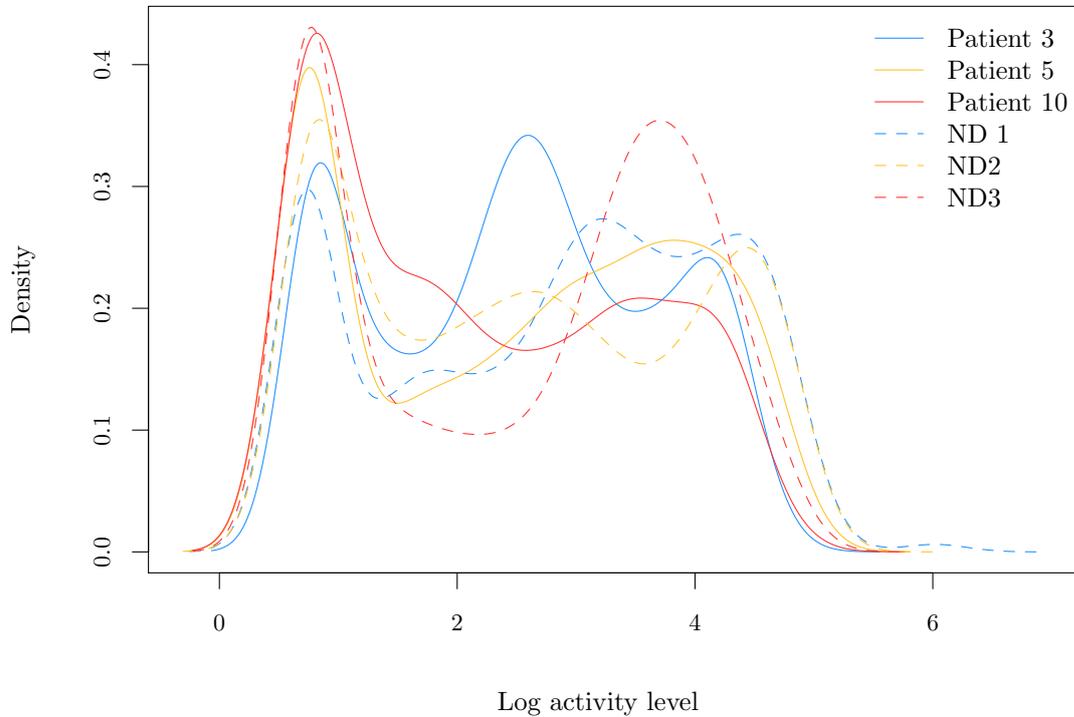


Figure 4.30: Log-transformed BFEN densities for each person: Patient 3, blue, solid; Patient 5, orange, solid; Patient 10, red, solid; ND 1, blue, dashed; ND 2, orange, dashed; and ND 3, red, dashed.

seated and non-seated activities (Dunstan *et al.*, 2012; Henson *et al.*, 2016; McCarthy *et al.*, 2017). The use of the classifications of activity levels according to the BFEN summary can therefore be explored further for bivariate glucose prediction. This is carried out in Chapter 6 where joint models are explored.

4.5.2 Activity zone comparison

It is important to ensure the model chosen to classify the underlying activity zones of a given activity summary performs well across different patients. The comparison of the posterior within-state densities produced for a range of patients is useful for this analysis, and is also useful for determining the extent to which a universal model can be developed for all patients, or whether models need to be tailored to individuals.

Fig. 4.30 shows the densities of the log-transformed BFEN summaries for Patient 3, Patient 5 and Patient 10 and ND 1, ND 2 and ND 3. Taking the log of each summary has reduced the variation in the range between each of the patients, where the maximum activity levels observed are vastly different. ND 1 has a longer upper tail than the other

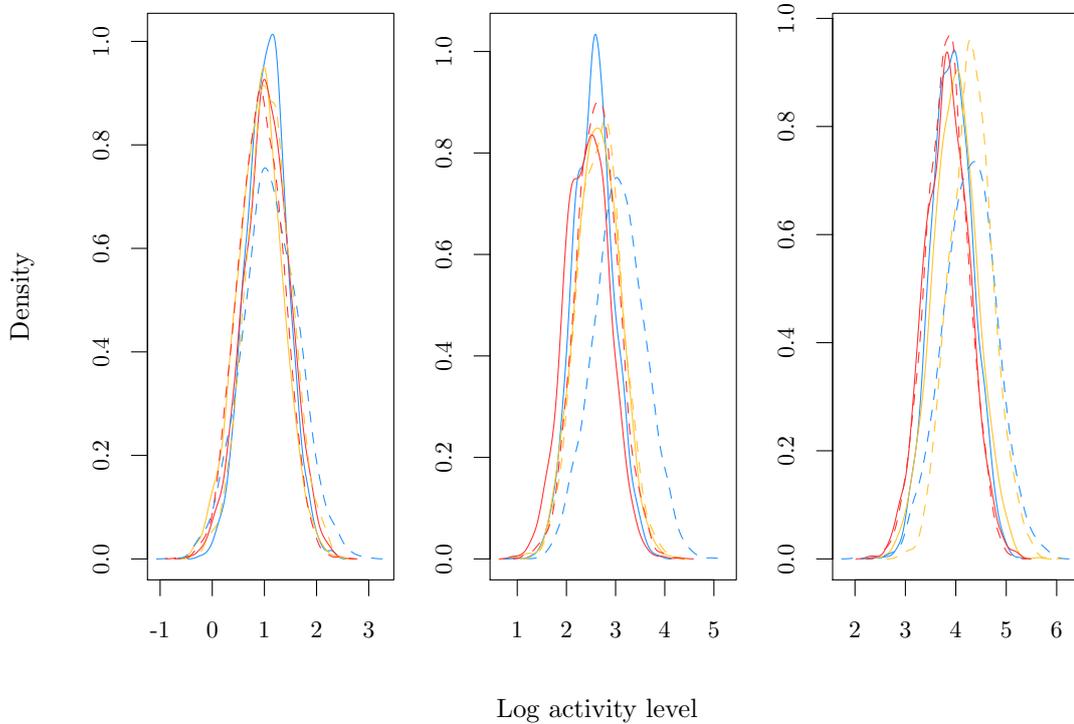


Figure 4.31: BFEN within-state posterior densities for each hidden state (one to three, left to right, respectively) for each person: Patient 3, blue, solid; Patient 5, orange, solid; Patient 10, red, solid; ND 1, blue, dashed; ND 2, orange, dashed; and ND 3, red, dashed.

patients, because of the significantly higher activity observations that were recorded when the patient was running, but there is little density here. The log-transformation has had a positive effect on reducing the impact of differing maximum activity across patients. The shapes of the densities in the mid-range of activity levels vary, but in general each density can be described by a mixture of three Normals. The similarities between the patient summaries imply that a universal model for activity could be possible.

Fig. 4.31 shows the posterior within-state densities for each patients' BFEN activity summaries, where each panel represents one of the hidden states (one to three, left to right). On the whole, the densities across the patients are the same, however the posterior within-state density for the second and third activity zones for ND 1 and the third zone for ND 2 are centred about slightly higher activity levels than for the other patients. For the third activity zone, the lower tails of the densities fall around a similar value to the other patients and the upper tails are just above the tails of the other patients, which account for the higher values observed for these patients. If used to cross-classify the other patients' data, these densities are unlikely to change the resulting classification. For the second activity zone, the higher mean of the posterior within-state density for ND 1 results in a

greater overlap with the third activity zone, which could result in the observations on the boundaries of these states being classified differently than with the posterior densities of the other patients. Overall, Fig. 4.31 confirms that the BFEN summary with a log-Normal within-state distribution is a versatile way of classifying data from different patients, and is one that produces consistent results across patients, too.

The results in this section show that the posterior within-state densities across patients are very similar, especially for the patients who have been diagnosed with diabetes. This opens up the possibility of developing a model that can be used across all patients, or at least that can be used to develop an informative prior for applying the model to new patients, perhaps allowing the model to be fitted quickly to short time series. Whilst the maximum activity observation across the diabetic and non-diabetic patients varies dramatically, the log-transform of the data ensures that the posterior density of the third activity zone is similar across both groups, showing that a lack of higher level observations does not significantly affect the fitted HMM. These results, along with the high proportion of matching annotations, prove that the BFEN summary works well for consistently monitoring the activity levels of patients.

4.6 Summary

The aim of this chapter was to find an appropriate model and activity summary that classifies the activity observations from the chosen summary into realistic and interpretable activity intensity zones. This was mainly carried out by fitting HMMs to the data, after briefly considering a HMM with independent hidden states (a mixture model). Use of a HMM with first order Markovian dependency between the hidden states more accurately captured the way the sequence of states evolved over time, with fewer transitions between the states that better described how a person moves between activity states.

Three activity summaries were considered, the BFEN activity summary, Algorithm 2.3, ENMO, Algorithm 2.1, and the complete step count, Algorithm 2.7. Because of the long positive tails in these summaries, two within-state distributions were also considered: the Normal distribution, for simplicity and conjugacy; and the skew Normal distribution, to better fit the long tails. The skew Normal HMM successfully improved the fit of the posterior within-state distributions for the ENMO activity summary in particular, which could not easily be transformed to reduce the skew of the data because of the negative observations this summary contained. A Normal HMM was sufficient after transforming the other activity summaries, however, reducing the model complexity.

For the step count data, a higher number of hidden states was used to classify the data, after plots of the square-root transformed data showed four modes. Fitting a four-state HMM improved the fit of the posterior within-state distributions to the data and

showed that the HMM can be adapted easily. A three-state model was sufficient for the other summaries and when comparing the three- and four-state models for the square-root transformed step count data, the three-state HMM still provided interpretable results.

An important section of this chapter was in comparing the results across the different activity summaries. Additional partially labelled data collected from healthy patients allowed for the interpretations of the classified observations from the ENMO and BFEN summaries to be validated. In doing so, it was deduced that the ENMO algorithm was not an appropriate summary for classifying observations into activity intensity zones, as the observations from this algorithm were frequently miss-classified. This would lead to incorrect inferences when using this information to model glucose levels. The BFEN algorithm was therefore chosen for transforming the accelerometer observations hereafter, with the log-Normal HMM used for classifying the observations it produces.

Additional investigation into how the Normal HMM performs on the log-transformed BFEN data from other patients was also carried out. The posterior within-state distributions across diabetic and non-diabetic patients were very similar, confirming that a universal model for activity or an informative prior could be used across patients.

Chapter 5

Monitoring and forecasting glucose levels

5.1 Background

The exploratory data analysis carried out on the glucose profiles of eight diabetic patients in Chapter 2 showed the need for a dynamic, seasonal model to capture the diurnal, but often varied, nature of blood glucose levels over time. To model the autocorrelation between observations, $AR(p)$ models are first explored, to measure how well a basic dependency structure explains the changes in future glucose levels for short-term (up to two hours) forecasts. These models are fitted for different orders, p . Following this, DLMS are fitted to glucose levels to see if a more complex model improves predictive performance. Models are developed using Patient 3's glucose data (the same patient whose activity data informed the modelling decisions in Chapter 4) holding back one day, which will be used to compare the predictive performance of the different models.

5.2 Autoregressive models

AR models have been widely implemented in modelling glucose levels in previous research (Gani *et al.*, 2009; Sparacino *et al.*, 2007). The success of such models at forecasting glucose levels varies depending on the order of the model and the length of the predictive horizon. The glucose profiles for each patient, Fig. 2.1, show a seasonal pattern, with rises in glucose levels corresponding to meal times, for example. Complex roots in the characteristic polynomial of an $AR(p)$ model give rise to quasi-periodic behaviour (Huerta & West, 1999) and so the chosen order must be high enough for the model to allow for this kind of behaviour. Stationarity is an important property to consider when fitting AR models to time series. If the parameters of the model lie outside of the stationary

region, the forecasts it produces become unstable, with the predictive variance increasing without bound. Despite their seasonality, the glucose profiles are assumed to be stationary for the purpose of this section, which is a reasonable assumption to make as the overall mean glucose level for each patient is relatively constant and so is the variance in glucose levels. If a high enough order is selected, an $AR(p)$ model could therefore be sufficient in predicting glucose levels over a short time horizon. The simple AR models fitted in this section provide a baseline of predictive performance for the more complex models fitted later in the chapter.

A Bayesian AR model of order p is described in Model 5.1.

Model 5.1

$$y_t | \mu, \sigma_y, \underline{\phi} \sim N(\mu + (X_t - \mu \mathbf{1}'_p) \underline{\phi}, \sigma_y^2),$$

where $X_t = (y_{t-1}, \dots, y_{t-p})$ and $\phi_i(\rho)$ is described by Eq. (3.5), for $i = 1, \dots, p$,

$$\rho_k = 2\rho_k^* - 1 \text{ with } \rho_k^* \sim Be(a_1, b_1) \text{ for } k = 1, \dots, p,$$

$$\mu \sim N(f, g^2),$$

$$\sigma_y^{-2} \sim Ga(a_2, b_2).$$

As described in Section 3.1.4, the stationary region of an $AR(p)$ model becomes difficult to define in terms of the model parameters when $p > 2$, however the model coefficients, $\underline{\phi}$, can be expressed in terms of the model partial autocorrelations, ρ_k for $k = 1, \dots, p$, as seen in Eq. (3.5). The stationary region of the model is then defined when the modulus of the model partial autocorrelations are less than one. In Model 5.1, this is implemented by assigning ρ_k^* a Beta prior, which restricts it to the $(0, 1)$ space, then applying the transformation $\rho_k = 2\rho_k^* - 1$ to produce a prior defined on $(-1, 1)$. The ρ_k^* have shared prior hyperparameters a_1 and b_1 .

Updates to the coefficient vector $\underline{\phi}$ in the MCMC sampler are made via updates to each ρ_k^* , to avoid introducing the Jacobian of the transformation. A closed form posterior distribution is not available for ρ_k^* , so a MH update is necessary for each one. A proposal value, $\tilde{\rho}_k^*$, is generated from $Be(\lambda_k \rho_k^* + \delta, \lambda_k (1 - \rho_k^*) + \delta)$. The parameter δ is a ‘nudge’ parameter: a small increment that is added into the proposal distribution to avoid the chain sticking in one of the tails of the distribution, and λ_k is a tuning parameter chosen to manage the acceptance rate of the chain. The acceptance probability is formed by combining the prior for ρ_k^* , the proposal distribution and the likelihood, which is computed after transforming ρ_k^* to ρ_k to $\underline{\phi}$.

A Normal prior distribution is assigned to the mean, μ , and a Gamma distribution is assigned to the precision σ_y^{-2} as these choices are semi-conjugate. Combining these with the likelihood, where given the length of the series it is reasonable to assume $p(y_{1:p}) \propto \mathbf{1}$, the full conditional distributions for these parameters are derived as follows:

$$\begin{aligned}
 \pi(\mu, \sigma_y^{-2} | \underline{y}, \underline{\phi}) &\propto \pi(\mu) \pi(\sigma_y^{-2}) \prod_{t=p+1}^T p(y_t | \underline{\phi}, \mu, \sigma_y^{-2}), \\
 &\propto \exp \left[-\frac{1}{2g^2} (\mu - f)^2 \right] \exp[-b_2 \sigma_y^{-2}] (\sigma_y^{-2})^{a_2-1} \\
 &\quad \prod_{t=p+1}^T \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp \left[-\frac{1}{2\sigma_y^2} (y_t - \mu - (X_t - \mu \underline{1}'_p) \underline{\phi}) \right],
 \end{aligned}$$

so

$$\begin{aligned}
 \mu | \underline{y}, \underline{\phi}, \sigma_y^{-2} &\sim N \left(\left(\frac{1}{g^2} + \frac{1}{\sigma_y^2} \sum_{t=p+1}^T (1 - \underline{\phi}' \underline{1}_p)^2 \right)^{-1} \left(\frac{f}{g^2} + \frac{1}{\sigma_y^2} \sum_{t=p+1}^T (1 - \underline{1}'_p \underline{\phi}) (y_t - X_t \underline{\phi}) \right), \right. \\
 &\quad \left. \left(\frac{1}{g^2} + \frac{1}{\sigma_y^2} \sum_{t=p+1}^T (1 - \underline{\phi}' \underline{1}_p)^2 \right)^{-1} \right) \quad (5.1)
 \end{aligned}$$

and

$$\sigma_y^{-2} | \underline{y}, \underline{\phi}, \mu \sim Ga \left(a_2 + \frac{T-p}{2}, b_2 + \frac{1}{2} \sum_{t=p+1}^T (y_t - \mu - (X_t - \mu \underline{1}'_p) \underline{\phi})^2 \right). \quad (5.2)$$

As $\mu | \underline{y}, \underline{\phi}, \sigma_y^{-2}$ and $\sigma_y^{-2} | \underline{y}, \underline{\phi}, \mu$ exist in a closed form, Gibbs steps are used to update μ and σ_y^{-2} . The full Metropolis within Gibbs scheme is shown in Algorithm 5.1.

5.2.1 Results

AR models of order 1–12 have been fitted to the glucose data using Algorithm 5.1. The highest order of 12 corresponds to glucose levels up to an hour behind. The models are fitted to the first four days of glucose data and tested using out-of-sample validation on the remaining day of data.

The prior hyperparameters for each of the models are the same; a $Be(1,1)$ prior is assigned to the ρ_k^* and a $Ga(1,1)$ is assigned to σ_y^{-2} to reflect prior ignorance, and μ has a prior mean of seven and standard deviation of one, which covers a range of plausible values for the mean glucose level. A summary of the number of iterations each model was run for after the burn in period, the tuning parameters and the acceptance rate for each ρ_k^* for each model is shown in Table 5.1. Each model had a burn in of 1,000 iterations and was thinned to 10,000 samples. Some of the models required longer runs to get the same effective sample size, as the draws were more autocorrelated. A nudge of $\delta = 0.001$ was used in each proposal to avoid the chain sticking. The MCMC runs for these models completed within a few hours for most orders, due to the low number of

Algorithm 5.1: AR(p) MCMC scheme	
1.	Initialise $\mu^{(0)}$, $\sigma_y^{-2(0)}$ and $\rho_k^{*(0)}$, for all k , by sampling from prior distributions in Model 5.1;
2.	for $i = 1, \dots, N$:
3.	draw $\mu^{(i)}$ from (5.1);
4.	draw $\sigma_y^{-2(i)}$ from (5.2);
5.	for $k = 1, \dots, p$:
6.	generate a proposal $\tilde{\rho}_k^*$ from $q(\rho_k^{*(i-1)}, \tilde{\rho}_k^*)$;
7.	calculate $\tilde{\phi}$ using Eq. (3.5), with $\rho_1 = \rho_1^{(i)}, \dots, \rho_{k-1} = \rho_{k-1}^{(i)}, \rho_{k+1} = \rho_{k+1}^{(i-1)}, \dots, \rho_p = \rho_p^{(i-1)}$;
8.	evaluate the acceptance probability $A(\rho_k^{*(i-1)}, \tilde{\rho}_k^*) = \min \left(1, \frac{p(\underline{y} \cdot)\pi(\tilde{\rho}_k^*)q(\tilde{\rho}_k^*, \rho_k^{*(i-1)})}{p(\underline{y} \cdot)\pi(\rho_k^{*(i-1)})q(\rho_k^{*(i-1)}, \tilde{\rho}_k^*)} \right);$
9.	accept $\tilde{\rho}_k^*$ with probability $A(\rho_k^{*(i-1)}, \tilde{\rho}_k^*)$: set $\rho_k^{*(i)} = \tilde{\rho}_k^*$, otherwise set $\rho_k^{*(i)} = \rho_k^{*(i-1)}$;
10.	end
11.	end

iterations required for convergence. For the longer runs of $p = 5, 11$, the MCMC scheme still completed running within a day.

Fig. 5.1 shows the posterior means and 95% credible intervals for the AR coefficients for each model. In all models, ϕ_1 is the largest coefficient and for the AR(1) model this is estimated close to one, which represents a model close to a random walk. The higher order coefficients of orders $p \geq 2$ are much smaller, so there is expected to be only a small improvement in forecasts between these models, in particular for $p \geq 5$. A look at the ACF plot in Fig. 5.2 (left) confirms the high correlation between lagged observations up to two hours apart, but the partial autocorrelation function (PACF) plot (right) shows that much of this is caused by propagation of the autocorrelation in the first four lags. This supports the small posterior coefficient results obtained for the higher order AR models with $p > 4$.

The trace plots for μ and σ_y^2 are shown in Fig. 5.3, for all models. The posterior draws for μ (right) have converged to the same value, as expected. The posterior draws for σ_y^2 (left) overlap for orders 2–12, but are higher for the AR(1) model. This shows that including lag- k terms for $k = 2, 3, \dots$ allows more of the variation in the data to be explained, but this effect quickly plateaus.

Ensemble forecasts have been calculated for a predictive horizon of two hours for each of the models, where each forecast in the ensemble is calculated using a posterior draw

Table 5.1: Model information for AR(p) MCMC.

Order, p	Iterations, N	Tuning parameters, $\lambda_{1:p}$	Acceptance rates
1	20,000	200	0.38
2	20,000	200, 30	0.29, 0.23
3	20,000	800, 700, 700	0.46, 0.73, 0.69
4	30,000	800, 700, 800, 1,000	0.43, 0.74, 0.69, 0.73
5	500,000	500, 600, 300, 600, 1,200	0.35, 0.73, 0.56, 0.66, 0.75
6	40,000	200, 600, 500, 300, 100, 1,200	0.24, 0.73, 0.63, 0.56, 0.38, 0.75
7	40,000	200, 600, 400, 500, 300, 300, 1,300	0.23, 0.73, 0.60, 0.64, 0.57, 0.57, 0.76
8	20,000	200, 600, 400, 600, 500, 300, 900, 1,400	0.25, 0.71, 0.57, 0.68, 0.57, 0.55, 0.74, 0.77
9	30,000	200, 600, 400, 500, 300, 300, 1,100, 100, 1,400	0.21, 0.73, 0.61, 0.65, 0.56, 0.56, 0.75, 0.39, 0.77
10	20,000	200, 600, 400, 500, 500, 300, 700, 100, 600, 1,200	0.22, 0.73, 0.60, 0.65, 0.63, 0.57, 0.69, 0.39, 0.67, 0.75
11	500,000	200, 600, 400, 500, 500, 300, 700, 100, 600, 100, 1,200	0.21, 0.73, 0.60, 0.64, 0.64, 0.57, 0.68, 0.39, 0.67, 0.39, 0.75
12	20,000	200, 600, 400, 500, 500, 300, 700, 100, 600, 100, 100, 1,400	0.23, 0.73, 0.60, 0.65, 0.63, 0.57, 0.69, 0.39, 0.66, 0.39, 0.39, 0.77

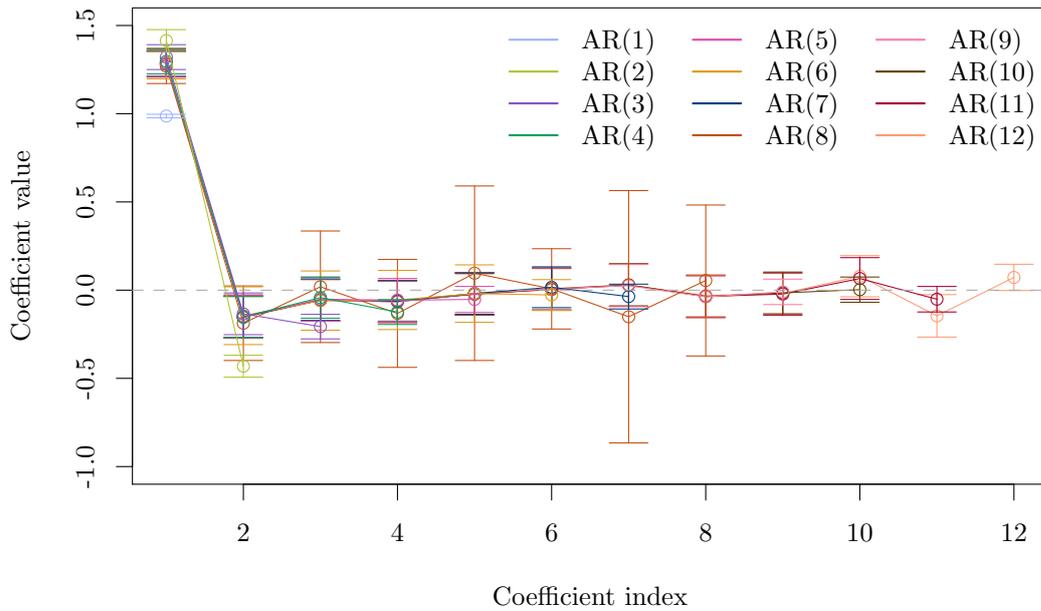


Figure 5.1: AR coefficient posterior means and 95% posterior interval for orders $p = 1-12$.

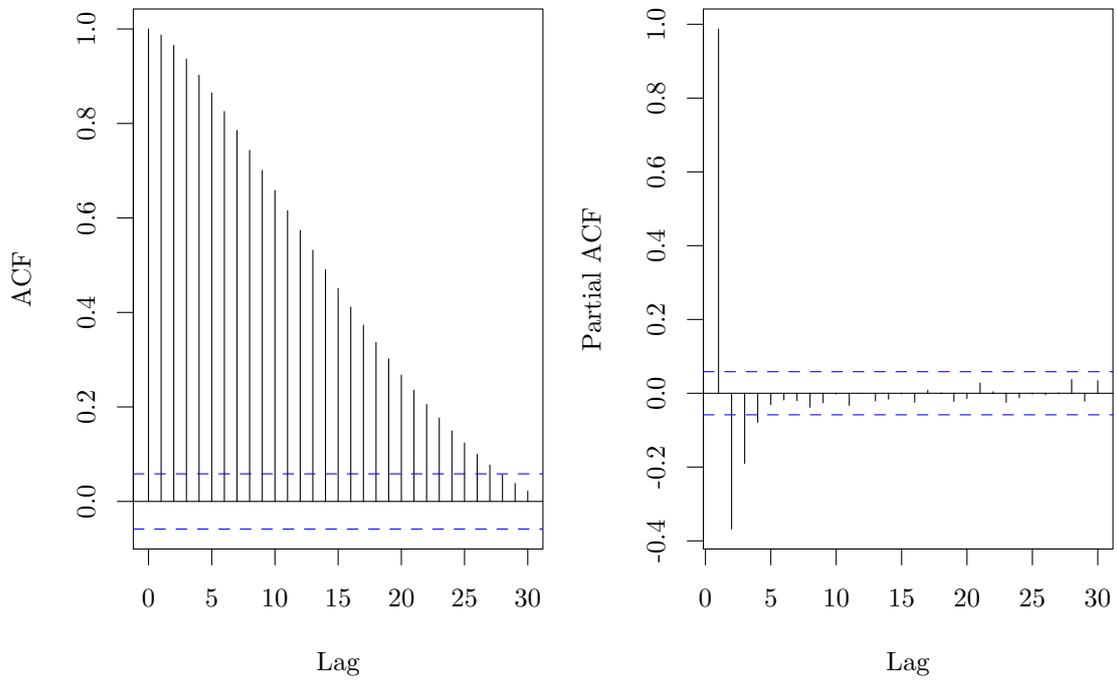


Figure 5.2: ACF and PACF plots for Patient 3's glucose data.

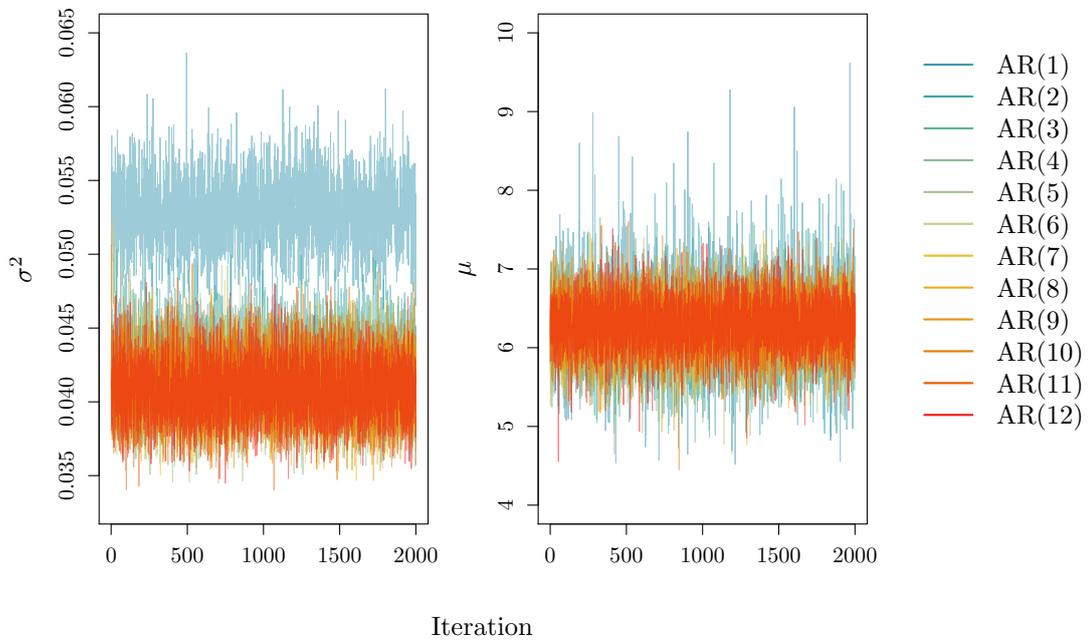


Figure 5.3: Trace plots for σ^2 and μ for AR models of order $p = 1-12$. Plots are thinned by five.

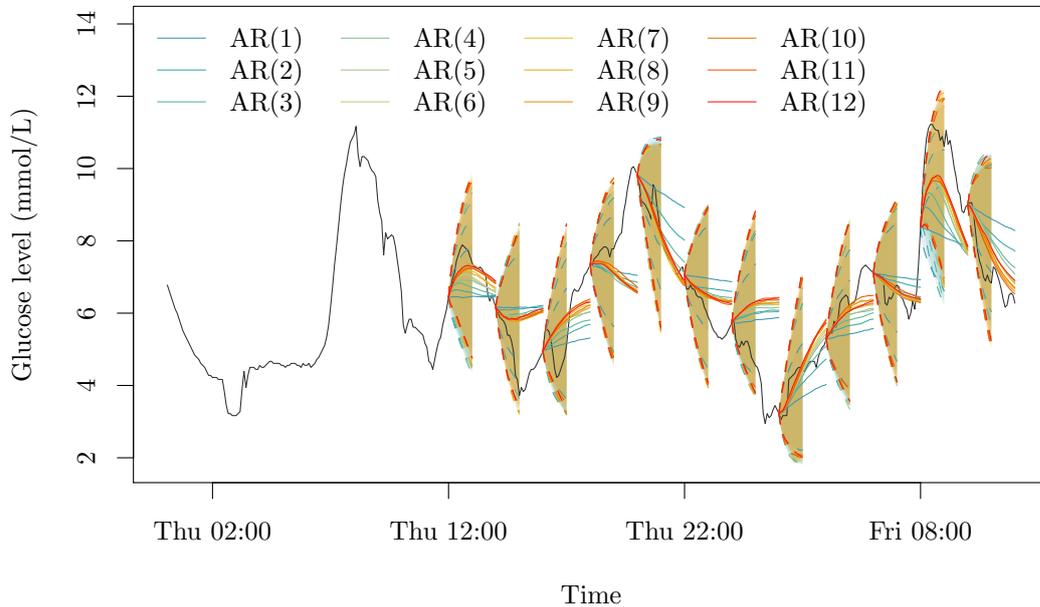


Figure 5.4: Two-hour-ahead forecasts (solid) for AR models with $p = 1-12$ fitted to Patient 3's glucose levels, with 95% credible intervals (shaded, dashed).

from Algorithm 5.1. To get a range of forecasts starting from different points in the glucose profile, an ensemble forecast is performed every two hours over the last day of data and then compared to the observed series. The forecasts produced at each of the selected time stamps provide a credible interval of the forecasts that can be expected using the posterior densities of each of the model parameters. The mean of each set of forecasts along with a 95% credible interval for the first hour of each forecast for each model is shown in Fig. 5.4. As the posterior parameter estimates imply, as p increases the improvement in forecasts becomes negligible, as the forecasts converge to a similar mean forecast. The credible intervals for each model also converge to a similar size. The forecasts for the AR(1) model are relatively flat, as expected, and the curvature in the data is predicted better as p increases.

The performance of each model for each forecasting index is quantified in Fig. 5.5. The plot shows a point estimate of the mean number of standard deviations the mean forecast of each model is from the observed series. The figure shows a similar story to the forecasts in Fig. 5.4, where the improvement in models plateaus for orders of $p \geq 4$. The mean number of standard deviations for these higher orders is very similar across the forecast indices. The plot shows the particular forecasts that all of the models performs worse on, for example forecasts 7 and 11. At forecast 7, the glucose profile starts to flatten, which causes the forecast to be drawn on a similar trajectory, when in reality the curve dips

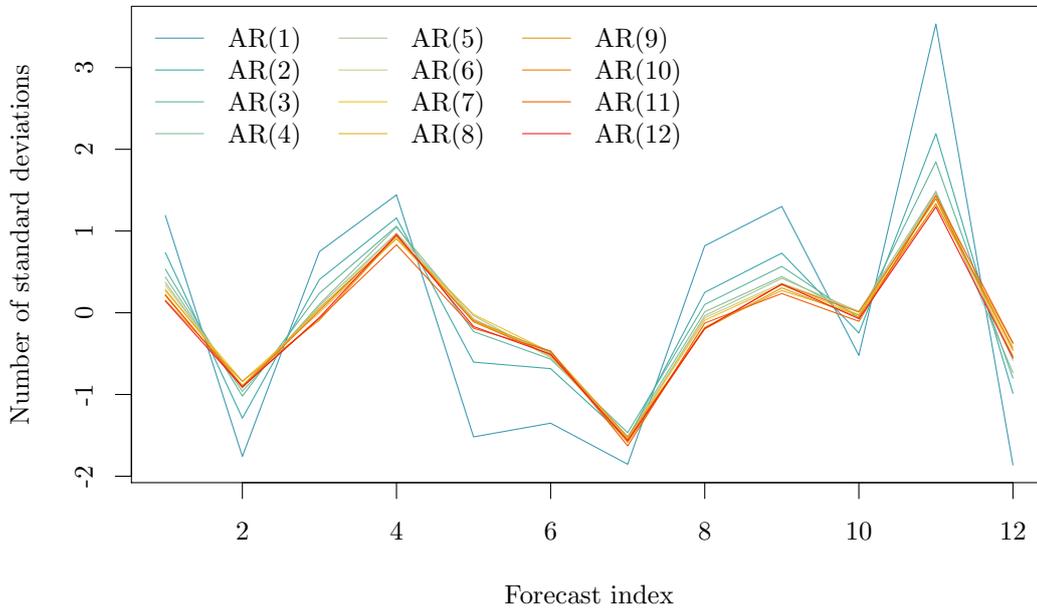


Figure 5.5: Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for AR models with $p = 1-12$.

again. At forecast 11, the height of the peak in glucose levels is underestimated, causing a larger error. The lower order forecasts, orders one to three, perform particularly badly at this point as they predict a fairly flat trajectory regardless of the starting point, due to not including much of the local glucose history.

As there is little predictive improvement to justify the added model complexity for orders higher than 4, the ‘best’ AR model is chosen as the AR(4) model. The forecasts for this model perform reasonably well, but there is clear room for improvement, in particular for taking forecasts at points in the glucose profile where glucose is increasing or decreasing, which miss the subsequent peaks and troughs in the data. For almost all of the forecasts, the credible intervals enclose the observed glucose levels. These intervals, however, are very large, and given that each interval encloses a range of future glucose levels, where some forecasts would result in a hyperglycaemic behavioural prompt and others would not, forecasts from this model would not be reliable. More complex models should therefore be explored to try to narrow the predictive intervals and to improve upon the fit of the AR models so that forecasts can better predict when a change in behaviour of the glucose profile is due. In particular, the AR models do not (explicitly) account for the diurnal pattern in glucose levels.

5.3 Dynamic linear models

DLMs allow data to be thought of as the combination of their features, such as a trend and seasonal component, and allow the underlying system of a time series to evolve over time, potentially accounting for a non-constant mean or changing variance. One of the key features of glucose levels is their periodicity, which creates some degree of structure but is subject to variation from underlying factors. Accurately modelling this behaviour is important for predicting future glucose levels. In this section, models are built using superposition of the DLM components introduced in Chapter 3. Relatively high order AR components in the form of a DLM and Fourier seasonal components are combined with an overall level, which is mean-reverting, to develop effective predictive models.

5.3.1 AR(p) DLM

In Section 5.2, glucose levels were described by AR models of orders one to 12. In this section, glucose is measured as a noisy realisation of an AR(p) model, rather than a realisation without noise. Choosing the order of the AR model to be greater than one in this case increases the order of Markovian dependency between the states. A non-zero mean is included in the DLM via a local level component, with the corresponding element in W equal to zero to make the mean static. A DLM representation of an AR(p) model was introduced in Section 3.1.4, but the full model specification for an AR(p) DLM with non-zero mean μ is shown in Model 5.2.

Model 5.2

$$y_t | \theta_t, V \sim N(F\theta_t, V),$$

$$\theta_t | \theta_{t-1}, W \sim N_{p+1}(G\theta_{t-1}, W),$$

$$F = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \end{bmatrix}, \quad G = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & \phi_1 & 1 & 0 & \dots & 0 \\ 0 & \phi_2 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \phi_p & 0 & 0 & \dots & 0 \end{bmatrix},$$

$\phi_i(\rho)$ is described by Eq. (3.5), for $i = 1, \dots, p$,

where $\rho_k = 2\rho_k^* - 1$ with $\rho_k^* \sim Be(a_1, b_1)$ for $k = 1, \dots, p$,

$$V = \psi_y^{-1}, \quad W = \text{diag}(0, \sigma^2, 0, \dots, 0),$$

$$\psi_y \sim Ga(a_2, b_2),$$

$$\sigma^{-2} \sim Ga(a_3, b_3).$$

As in Model 5.1, the coefficients of the AR process are restricted to the stationary

region by exploiting the relationship between the coefficients, $\underline{\phi}$, and the model partial autocorrelations, $\underline{\rho}$. MH steps are used to update $\underline{\phi}$ as before, by sequentially drawing ρ_k^* from a Beta proposal and transforming it to ρ_k and then updating $\underline{\phi}$ according to Eq. (3.5). The likelihood in the acceptance ratio is the product:

$$\prod_{t=p+1}^T p(\theta_{t,2} | \theta_{t-1:t-p,2}, \underline{\phi}, \sigma^2),$$

where $\theta_{t,2} | \cdot \sim N(X_t \underline{\phi}, \sigma^2)$ and $X_t = (\theta_{t-1,2}, \dots, \theta_{t-p,2})$. This is derived by writing out the state equations element-wise:

$$\begin{aligned} \theta_{t,1} &= \theta_{t-1,1} = \mu, \\ \theta_{t,2} &= \phi_1 \theta_{t-1,2} + \theta_{t-1,3} + w_{t,2}, \end{aligned} \tag{5.3}$$

$$\theta_{t,3} = \phi_2 \theta_{t-1,2} + \theta_{t-1,4}, \tag{5.4}$$

⋮

$$\theta_{t,p} = \phi_{p-1} \theta_{t-1,2} + \theta_{t-1,p+1}, \tag{5.5}$$

$$\theta_{t,p+1} = \phi_p \theta_{t-1,2}. \tag{5.6}$$

Using Eq (5.4) to replace $\theta_{t-1,3}$ in Eq. (5.3) gives

$$\theta_{t,2} = \phi_1 \theta_{t-1,2} + \phi_2 \theta_{t-2,2} + \theta_{t-2,4} + w_{t,2}.$$

Continuing in this fashion, after $p - 1$ replacements the equation for $\theta_{t,2}$ becomes

$$\theta_{t,2} = \phi_1 \theta_{t-1,2} + \phi_2 \theta_{t-2,2} + \dots + \phi_p \theta_{t-p,2} + w_{t,2},$$

where $w_{t,2} \sim N(0, \sigma^2)$. In other words, an AR(p) model for $\theta_{t,2}$. The states are sampled using forward filtering backward sampling, Algorithm 3.6. The full conditional distributions for ψ_y and $\psi_2 = \sigma^{-2}$ are as in Eq. (3.11) and Eq. (3.12), then the full Metropolis within Gibbs sampler is in Algorithm 5.2.

As p increases, the zero-elements on the diagonal of the system covariance matrix cause numerical instability in the MCMC process. To prevent computational errors, a small increment of $\varepsilon = 1 \times 10^{-6}$ is added to the diagonal elements of W . This addition makes the matrix non-singular and is small enough not to affect the overall model.

Results

Model 5.2 has been fitted to the first four days of glucose data for orders of $p = 1-10$. This corresponds to incorporating the observations a maximum of 50 minutes beforehand

Algorithm 5.2: AR(p) DLM MCMC scheme

1. Initialise $\psi_y^{(0)}$, $\mu^{(0)}$, $\sigma_y^{-2(0)}$ and $\rho_k^{*(0)}$, for all k , by sampling from prior distributions in Model 5.2;
2. **for** $i = 1, \dots, N$:
3. run the forward filtering backward sampling algorithm (Algorithm 3.6) to simulate $\underline{\theta}^{(i)}$;
4. draw $\psi_y^{(i)}$ from (3.11);
5. draw $\sigma^{-2(i)}$ from (3.12);
6. **for** $k = 1, \dots, p$:
7. generate a proposal $\tilde{\rho}_k^*$ from $q(\rho_k^{*(i-1)}, \tilde{\rho}_k^*)$;
8. calculate $\tilde{\phi}$ using Eq. (3.5), with
 $\rho_1 = \rho_1^{(i)}, \dots, \rho_{k-1} = \rho_{k-1}^{(i)}, \rho_{k+1} = \rho_{k+1}^{(i-1)}, \dots, \rho_p = \rho_p^{(i-1)}$;
9. evaluate the acceptance probability

$$A(\rho_k^{*(i-1)}, \tilde{\rho}_k^*) = \min \left(1, \frac{p(\theta_{p+1:T,2} | \cdot) \pi(\tilde{\rho}_k^*) q(\tilde{\rho}_k^*, \rho_k^{*(i-1)})}{p(\theta_{p+1:T,2} | \cdot) \pi(\rho_k^{*(i-1)}) q(\rho_k^{*(i-1)}, \tilde{\rho}_k^*)} \right);$$
10. accept $\tilde{\rho}_k^*$ with probability $A(\rho_k^{*(i-1)}, \tilde{\rho}_k^*)$: set $\rho_k^{*(i)} = \tilde{\rho}_k^*$, otherwise set $\rho_k^{*(i)} = \rho_k^{*(i-1)}$;
11. **end**
12. **end**

into modelling the current glucose level. The model hyperparameters were chosen as: $a_1 = b_1 = 1$, $a_2 = b_2 = 1$ and $a_3 = 1.1$, $b_3 = 0.01$. The $Ga(1,1)$ prior on ψ_y and $Ga(1.1, 0.01)$ prior on σ^{-2} encourages the posterior to support the allocation of more of the variance to the observation layer, rather than the hidden states. As in the proposal distribution for the AR coefficients in Algorithm 5.1, a nudge of 0.001 was added for each model, to avoid the chain sticking. Details of the number of iterations after burn in for each order, the tuning parameters used and the acceptance rates achieved are in Table 5.2. Each run had a burn in of 1,000 and was thinned to 10,000 draws. Due to the added model complexity and higher number of iterations, the MCMC schemes for these models took slightly longer than the traditional AR(p) models, but still completed within a day.

Whilst the marginal posteriors for the model parameters of Model 5.2 were unimodal for most of the model orders explored, for the AR(7) DLM the posterior densities for the AR coefficients were multimodal. From the trace plots, Fig. 5.6, an increase in one coefficient corresponds to a decrease in another, suggesting there are two modes in the joint posterior with comparable support. However, from Fig. 5.6, the sampler seems to jump between the two modes fairly readily and so a posterior sample of size 200,000 was deemed adequate to estimate the mass of each mode.

Table 5.2: Model information for AR(p) DLM MCMC.

Order, p	Iterations, N	Tuning parameters, $\lambda_{1:p}$	Acceptance rates
1	200,000	2,500	0.42
2	50,000	2,500, 600	0.16, 0.28
3	100,000	4,000, 1,600, 1,300	0.20, 0.32, 0.36
4	500,000	4,000, 3,000, 4,000, 4,000	0.18, 0.35, 0.41, 0.43
5	200,000	7,000, 6,000, 7,000, 6,000, 6,000	0.20, 0.40, 0.43, 0.45, 0.43
6	200,000	7,000, 6,000, 7,000, 6,000, 6,000, 5,000	0.77, 0.20, 0.38, 0.36, 0.41, 0.42
7	200,000	7,000, 6,000, 7,000, 6,000, 6,000, 5,000, 5,000	0.43, 0.23, 0.38, 0.37, 0.34, 0.37, 0.43
8	500,000	7,000, 6,000, 7,000, 6,000, 6,000, 5,000, 5,000, 5,000	0.20, 0.40, 0.44, 0.45, 0.46, 0.44, 0.44, 0.42
9	400,000	7,000, 6,000, 7,000, 6,000, 6,000, 5,000, 5,000, 5,000, 5,000	0.76, 0.20, 0.39, 0.38, 0.39, 0.39, 0.42, 0.43, 0.41
10	400,000	9,000, 15,000, 10,000, 10,000, 9,000, 10,000, 5,000, 7,000, 10,000, 15,000	0.22, 0.46, 0.45, 0.47, 0.46, 0.48, 0.44, 0.45, 0.47, 0.46

A summary of the coefficients from all of the AR(p) DLMs is seen in Fig. 5.7. This figure shows the mean and 95% upper and lower bounds of the posterior densities for each parameter. The main coefficients in these models are once again the first two, ϕ_1 and ϕ_2 , with the later coefficients falling closer to zero. The posterior variances in the coefficients for the AR(p) DLMs are notably larger than for the traditional AR(p) model coefficients, and the coefficients show a greater difference from zero. For the majority of the model orders, the coefficients decrease as their index increases. For the AR(9) model however, the coefficients are quite far from zero, and in some cases the coefficients of the higher order lags are larger in magnitude, for example at index 7. Because the parameters for this model also have larger posterior variances, the forecasts produced from this model may also see an increased predictive variance.

The trace plots for ψ_y^{-1} and σ^2 for all model orders are shown in Fig. 5.8. The observation variance for each model is similar, whilst the innovation variance decreases as p increases, up to $p = 4$, after which the posterior distributions are similar.

The predictive performance of the AR(p) DLMs for orders up to $p = 10$ has a similar ceiling to the traditional AR(p) models of order up to $p = 12$. The improvement in forecasts for the DLM representation, however, Fig. 5.9, happens at a much lower order than for the traditional AR representation. For the models fitted in Section 5.2, the forecasts converged to a similar trajectory for orders $p \geq 4$. For the models here, there is a large improvement between orders $p = 1$ and $p = 2$, followed by only small incremental

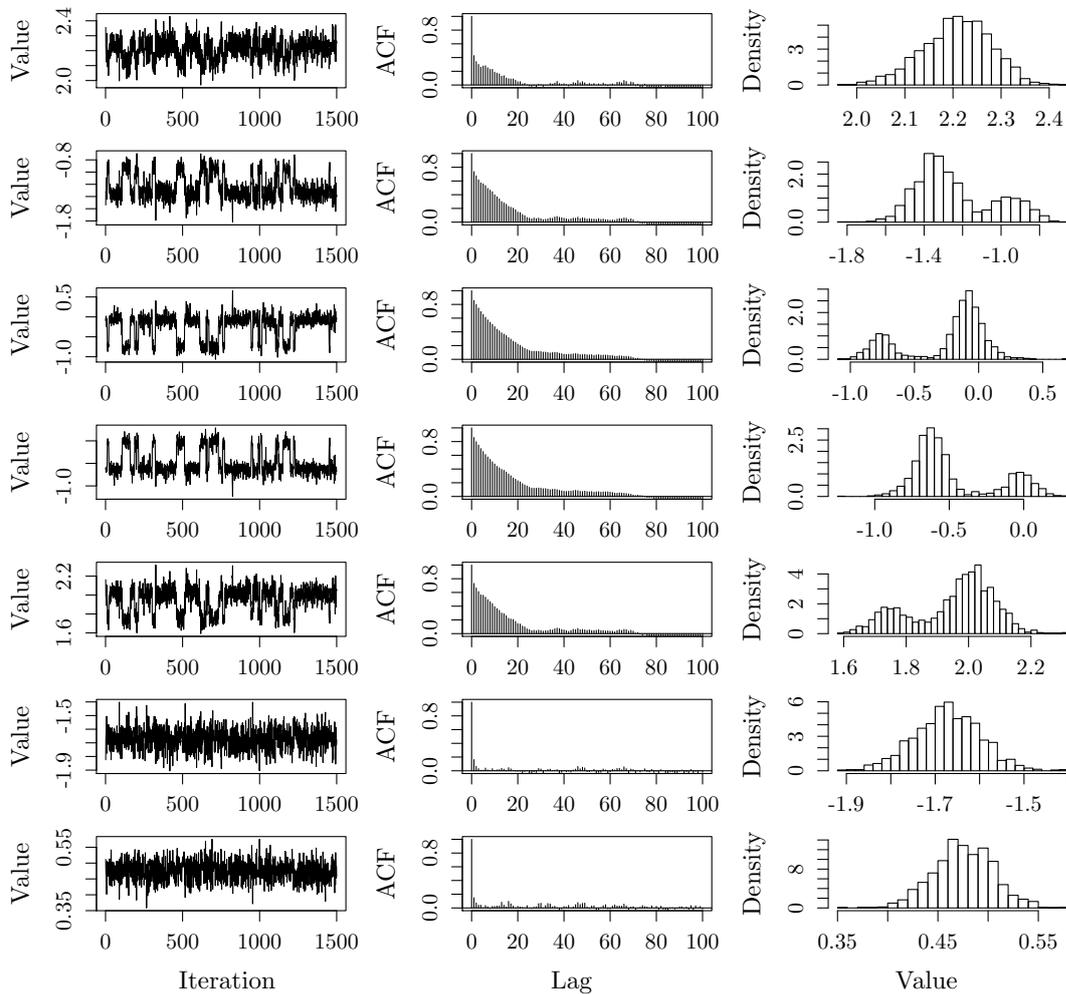


Figure 5.6: Trace plots, ACF plots and histograms of the posterior draws of the AR(7) coefficients.

improvements for orders thereafter. A lower order $AR(p)$ DLM therefore has a similar predictive performance as a higher order $AR(p)$ traditional model. The credible interval on the AR(7) DLM forecast does not appear to be wider despite the bimodality of the posterior, confirming that the two modes fit the data comparably. The credible interval on the AR(9) model forecasts, however, is much larger than for any of the other models. In particular, the interval on the forecasts at the final forecast index ranges between ± 300 (limits not shown on plot). The reasons for this were discussed previously.

Fig. 5.10 shows the mean number of standard deviations each forecast is away from the observed time series for each model, at each forecasting index. The plot shows how the forecasts converge to a similar trajectory for models of order $p \geq 2$, with similar sized credible intervals resulting in forecasts being a similar number of standard deviations from

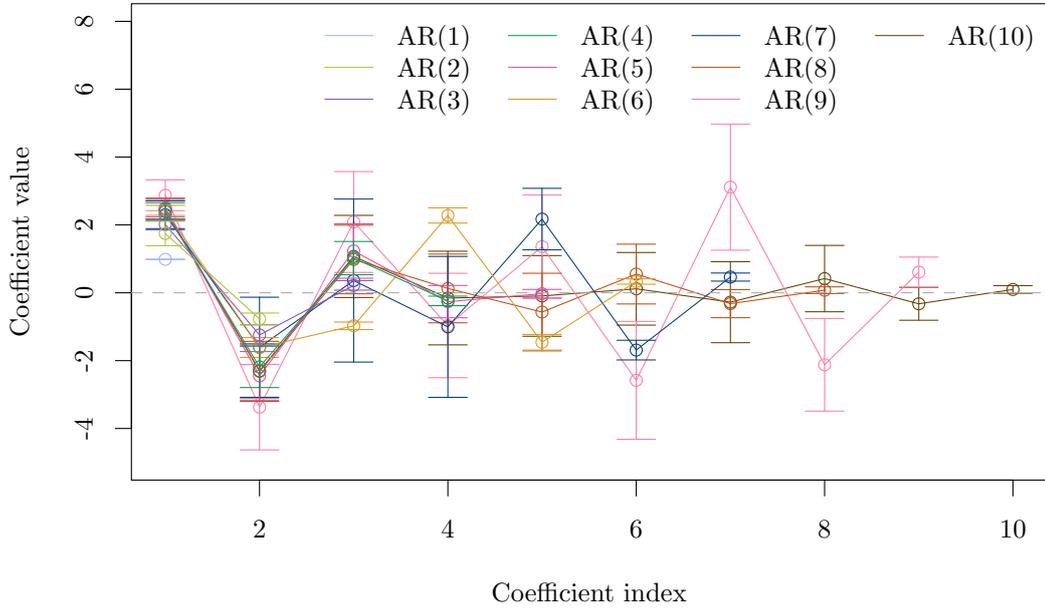


Figure 5.7: Model 5.2 AR coefficient posterior means and 95% posterior interval for orders $p = 1$ – 10 .

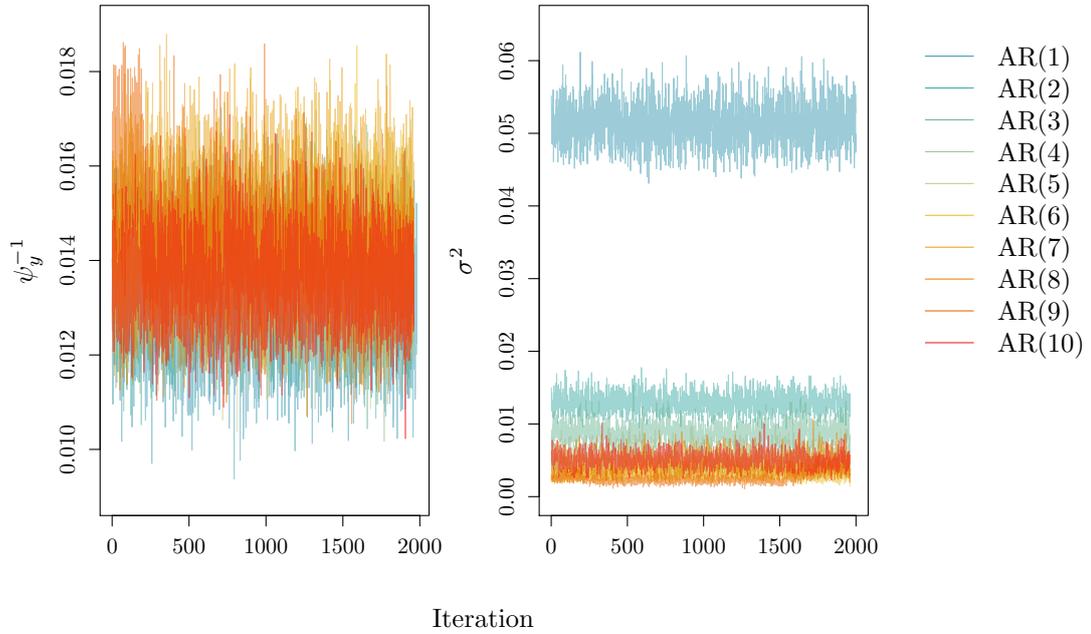


Figure 5.8: Trace plots for σ^2 and ψ_y^{-1} for AR DLMs of order $p = 1$ – 10 . Plots are thinned by five.

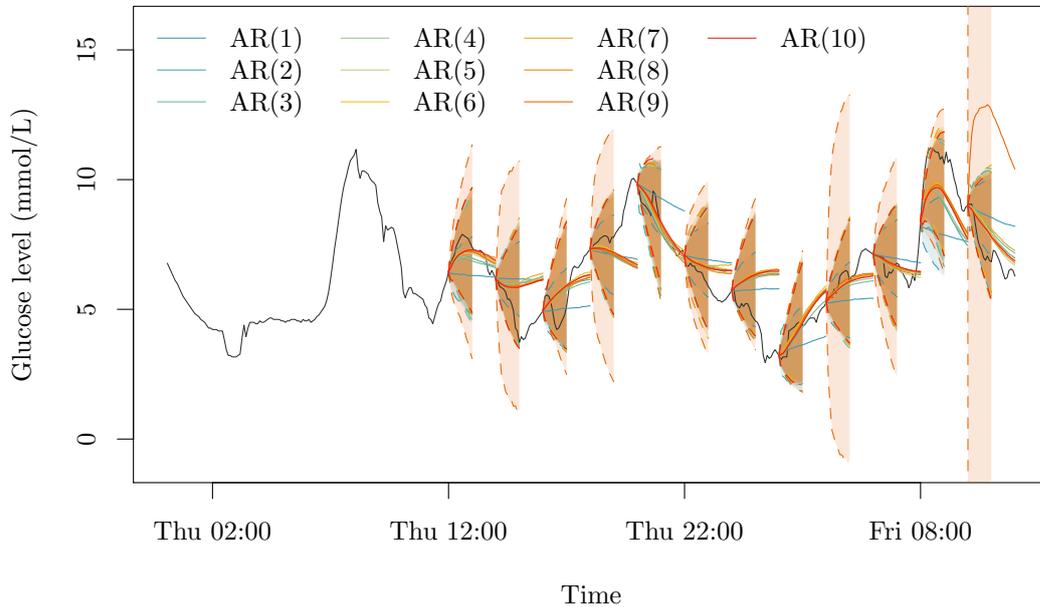


Figure 5.9: Two-hour-ahead forecasts (solid) for AR DLMs with $p = 1-10$ fitted to Patient 3's glucose levels, with 95% credible intervals (dashed).

the observed series for each of these model. From the figure, it is clear that the $AR(p)$ DLM lacks in performance for forecasts taken at points in the series where glucose levels change trajectory, for example at forecast index 7, and underestimates the peaks in the data, such as at forecast index 11, but overall the forecasts are reasonable. Model improvements to better capture the series in these cases are explored in the coming sections. The mean numbers of standard deviations for the $AR(9)$ model appear smaller than for the other model orders, but this is a side effect of the much larger credible intervals and therefore is a sign of a poorer fitting model. It is expected that forecasts should on average be around one standard deviation from the observed series.

Roots of the characteristic equation

The roots of the characteristic equation of an AR model can be investigated to determine whether the model describes pseudo-periodic behaviour (Huerta & West, 1999). Recall the characteristic polynomial defined in Section 3.1.4:

$$\Phi(B) = 1 - \sum_{j=1}^p \phi_j B^j,$$

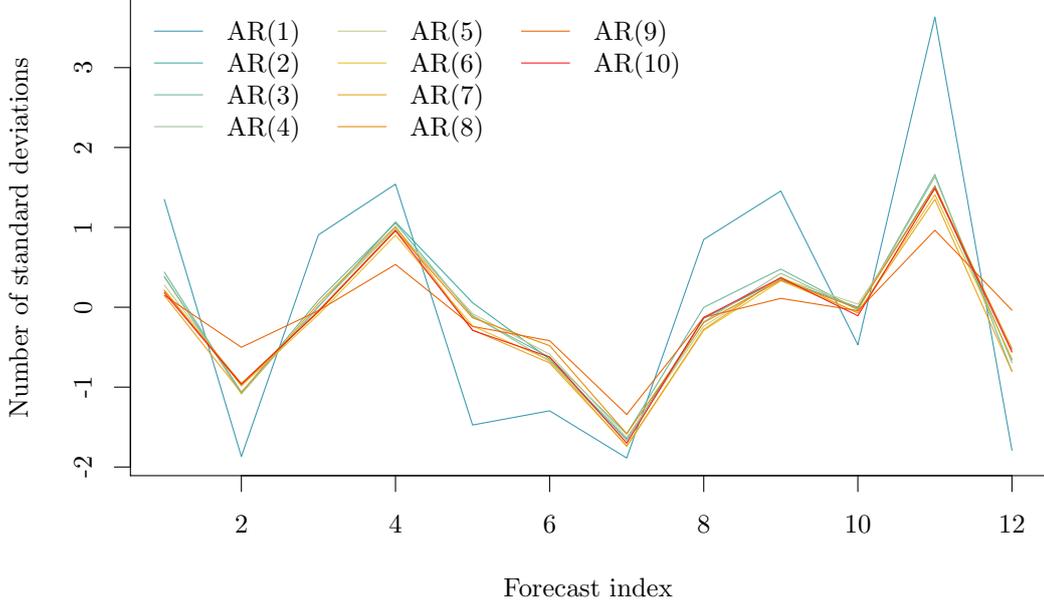


Figure 5.10: Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for AR DLMs with $p = 1-10$.

which, when combined with a zero-mean stationary stochastic process, X_1, X_2, \dots , defines an $\text{AR}(p)$ model: $\Phi(B)X_t = \varepsilon_t$. The reciprocal roots, $|\alpha_j| < 1$, of the characteristic polynomial determine the stationary region of an $\text{AR}(p)$ model. The roots, x , solve the characteristic equation:

$$\prod_{j=1}^p (1 - \alpha_j x) = 0. \quad (5.7)$$

Inverting $\Phi(B)X_t = \varepsilon_t$ and substituting in the product in Eq. (5.7) obtains

$$X_t = \frac{1}{\prod_{j=1}^p (1 - \alpha_j B)} \varepsilon_t,$$

which can be expressed in terms of partial fractions as

$$X_t = \sum_{j=1}^C Z_{tj} + \sum_{j=2C+1}^p A_{tj},$$

where the $\{Z_{tj}\}$ and $\{A_{tj}\}$ are latent processes corresponding to the complex pairs of roots and real roots, respectively. Suppose the modulus of the j -th complex pair of roots is r_j and their argument is ω_j . It can be shown that the corresponding latent process $\{Z_{tj}\}$

follows an ARMA(2,1) model, with AR operator

$$\Phi_{Z,j}(B) = 1 - 2r_j \cos(\omega_j)B + r_j^2 B^2$$

and moving average operator

$$\Theta_{Z,j}(B) = d_j + e_j B.$$

However, only the AR operator is important for determining the long run behaviour of the process. To understand why this is, consider the induced ACF for Z_{tj} , derived by solving a difference equation of the same form. For $k \geq 2$ the ACF at lag k , ρ_k , solves the linear difference equation

$$\{1 - 2r_j \cos(\omega_j)B + r_j^2 B^2\}\rho_k = 0.$$

The behaviour of these equations is governed by the auxiliary equation

$$x^2 - 2r_j \cos(\omega_j)x + r_j^2 = 0,$$

whose roots are $x = r_j \cos(\omega_j) \pm i\{r_j \sin(\omega_j)\}$, which are complex. The general solution to each difference equation therefore takes the form

$$r_j^k \{f_j \cos(\omega_j k) + g_j \sin(\omega_j k)\},$$

where f_j and g_j are constants. The values of the constants depend on the initial values ρ_0 and ρ_1 , which in turn are influenced by r_j and ω_j , as well as e_j and d_j in the moving average operator. Irrespective of these values, the general solution is a periodic function, with period $2\pi/\omega_j$. The amplitude of the oscillations depends on r_j^k , which changes with k . When $|r_j| < 1$, the amplitude of the oscillations in the ACF therefore decays to zero as k increases.

Given this property of AR models, if the roots of the characteristic equation are complex, the AR process is said to have pseudo-seasonal behaviour, i.e. be describing seasonality. The roots of the characteristic equations for the AR(p) DLM models fitted above are therefore investigated in this section to determine from which order, if any, the models begin to describe seasonal behaviour. The roots of the models are plotted in Fig. 5.11. Due to the ambiguity in the labelling of roots from different MCMC draws, the roots are computed based on the posterior mean values of the AR coefficients for this exploratory exercise.

Fig. 5.11 shows an increasingly large imaginary component to the roots as p increases. For $p \geq 4$, the models have between one and three pairs of complex conjugate roots with sizeable imaginary parts, suggesting these models may capture pseudo-seasonal behaviour. This confirms what is seen in Fig. 5.9, where the models of order greater than four capture

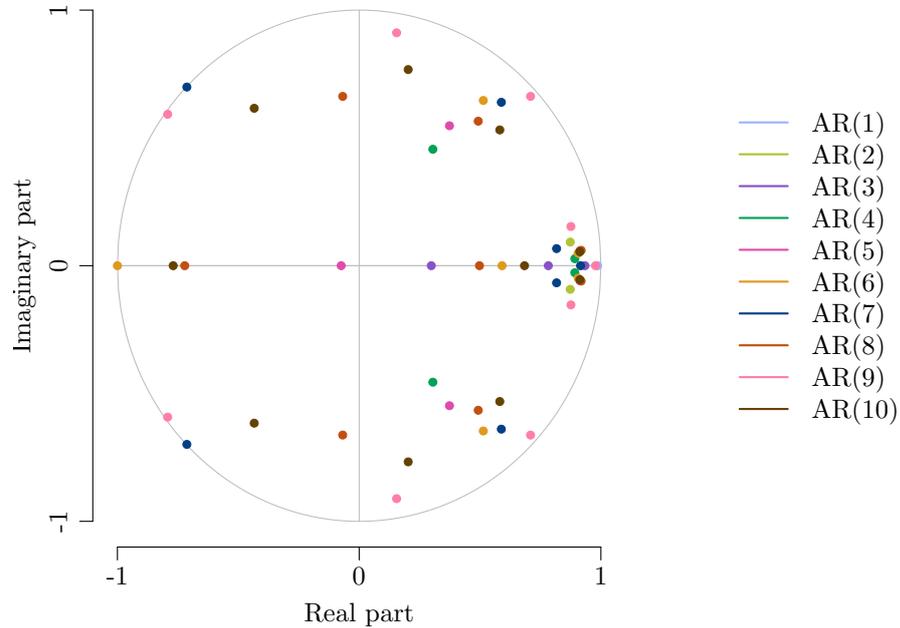


Figure 5.11: Roots of the characteristic equation for the AR models of order 1–10 with coefficients specified by their posterior means.

the shape of the glucose profile much better. Note that by construction, all of the roots in the figure fall within the unit circle.

Further exploration into the complex roots is carried out by decomposing the AR component of the DLM, $\theta_{t,2}$. The AR(4) DLM is used as an example. Setting each of the model parameters at their posterior mean and smoothing over the hidden states, as before, time series corresponding to each of the complex pairs of roots, Z_{tj} , and each real root, A_{tj} , are obtained. As there are two pairs of complex roots for the AR(4) DLM and no real roots, the two series corresponding to the complex pairs of roots are shown in Fig. 5.12. When adding together the three series displayed, the glucose data are recovered.

The series corresponding to the first complex root, C_1 , in Fig. 5.12 describes some periodicity in the noise of the glucose signal. The series corresponding to the second complex root, C_2 , is the periodic component with largest amplitude, found in each of the decomposed AR(p) DLMs. The local level component here, however, describes the majority of the seasonality in the glucose data. The purpose of the local level component in the models is to describe the non-zero mean of the glucose data, so these results show that a high order AR component is not sufficient in picking up this behaviour, though it does describe some higher frequency periodicity. Models to better describe the seasonal behaviour should therefore be explored.

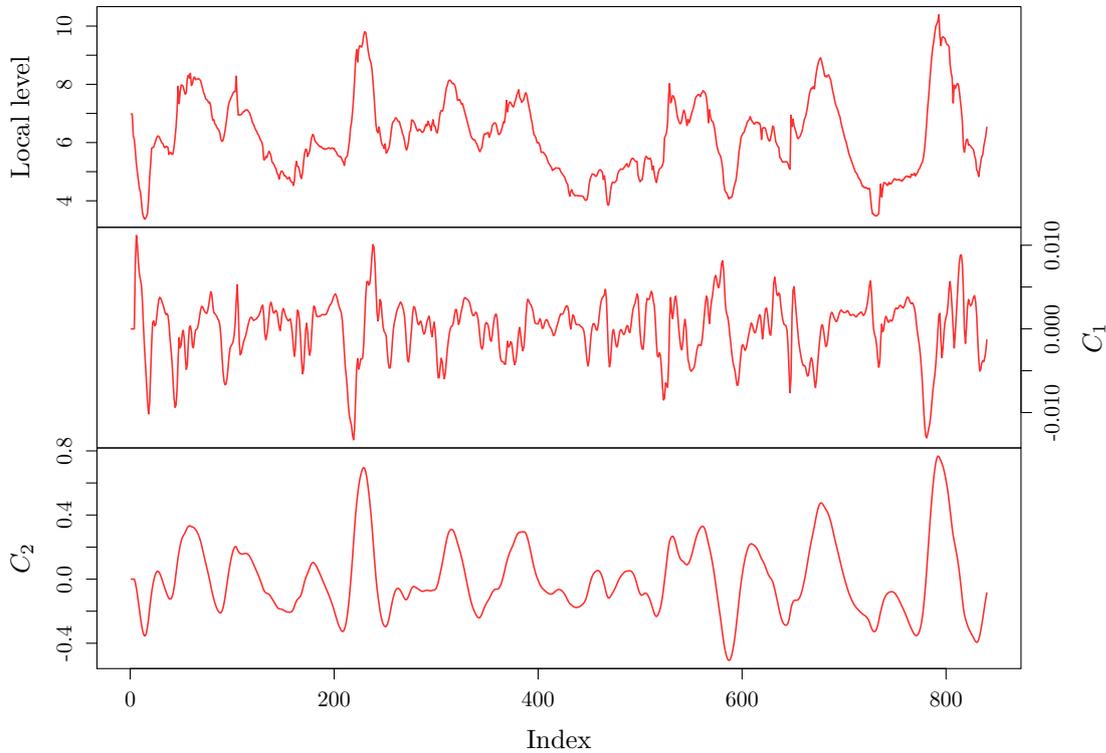


Figure 5.12: The AR(4) DLM for Patient 3’s glucose levels decomposed into the smoothed local level component and the series corresponding to the complex pairs of roots for the first four days of data.

5.3.2 Fourier seasonal model with local level component

As an alternative to a high order AR DLM, a more direct and (perhaps) more parsimonious way to capture the seasonality in the glucose profiles is to incorporate a Fourier seasonal component in a DLM. Describing the periodicity in the glucose data using a Fourier component requires a number of harmonics, q , to be specified. This component then corresponds to $2q$ diagonal elements in the system covariance matrix, W , that need to be learned. For series with a large number of observations, s , per period, q is typically chosen to be much smaller than $s/2$ which can lead to a parsimonious representation of seasonal behaviour. A local level component can also be paired with a Fourier component to model the variation in the mean over time.

A DLM with a local level component plus a Fourier component, referred to as a local Fourier DLM, is specified in Model 5.3.

Model 5.3

$$y_t | \theta_t, V \sim N(F\theta_t, V),$$

$$\theta_t | \theta_{t-1}, W \sim N_{2q+1}(G\theta_{t-1}, W),$$

$$\begin{aligned}
 F &= \begin{bmatrix} 1 & 1 & 0 & 1 & \dots & 0 \end{bmatrix}, & G &= \text{blockdiag}(1, H_1, \dots, H_q), \\
 \text{where } H_j &= \begin{bmatrix} \cos \omega_j & \sin \omega_j \\ -\sin \omega_j & \cos \omega_j \end{bmatrix} & \text{and } \omega_j &= \frac{2\pi j}{s} \text{ for } j = 1, \dots, q, \\
 V &= \psi_y^{-1}, & W &= \text{diag}(\psi_1^{-1}, \dots, \psi_{2q+1}^{-1}), \\
 & & \psi_y &\sim \text{Ga}(a_1, b_1), \\
 & & \psi_i &\sim \text{Ga}(a_2, b_2), \text{ for } i = 1, \dots, 2q + 1.
 \end{aligned}$$

The prior distributions assigned to the precisions ψ_y and ψ_i , for $i = 1, \dots, 2q + 1$, are the standard priors that were discussed in Section 3.4.1. The full conditional distributions for these parameters are in Eq. (3.11) and Eq. (3.12), respectively. The block Gibbs sampler is as described in Algorithm 3.7.

The number of observations per period in the glucose data is $s = 288$. A number of $q = 4$ harmonics is chosen to model each period, which is discussed in more detail in Section 5.3.3.

Results

Algorithm 3.7 was run for $N = 300,000$ iterations, plus a burn in of 1,000 and thinned by 30. The prior hyperparameters were: $a_1 = 1$, $b_1 = 1$, $a_2 = 1.1$, $b_2 = 0.01$, $C_0 = 3$, $m_0 = 7$. The $\text{Ga}(1.1, 0.01)$ prior on the inverse diagonal elements of W is to encourage posterior support for small values of the variances of the states. This is because stochastic variation in the seasonal component is thought to likely be slow.

The posterior densities for the diagonal elements of W , Fig.5.13, show that the elements corresponding to the even Fourier nodes (rows three, five, seven and nine of the figure) have a much larger value than those corresponding to the odd Fourier nodes (rows two, four, six and eight) and than the state of the local level component (row one). In particular, the final element of W is centred around 0.8, 400 times the size of the value for the even nodes. This is because only the odd Fourier nodes feature in the observation equation. As a result of this, the credible interval on the forecasts increases rapidly, shown in Fig. 5.14, making the model unsuitable. Despite the large errors, Fig. 5.14 shows that the local Fourier DLM uses the seasonality observed in the data up to the time of the forecast to predict future values; it is clear in the forecasts of the AR models that the predicted values trend towards the mean, whereas the forecasts from the local Fourier DLM, in particular the first and penultimate forecasts that previously missed the upcoming peaks in the glucose profile, better follow the shape of the profile, to an extent. The increase in glucose levels for these two forecasts has been overestimated by the current model. Another problem with the current credible intervals is that they include physiologically implausible values, such as negative glucose levels.

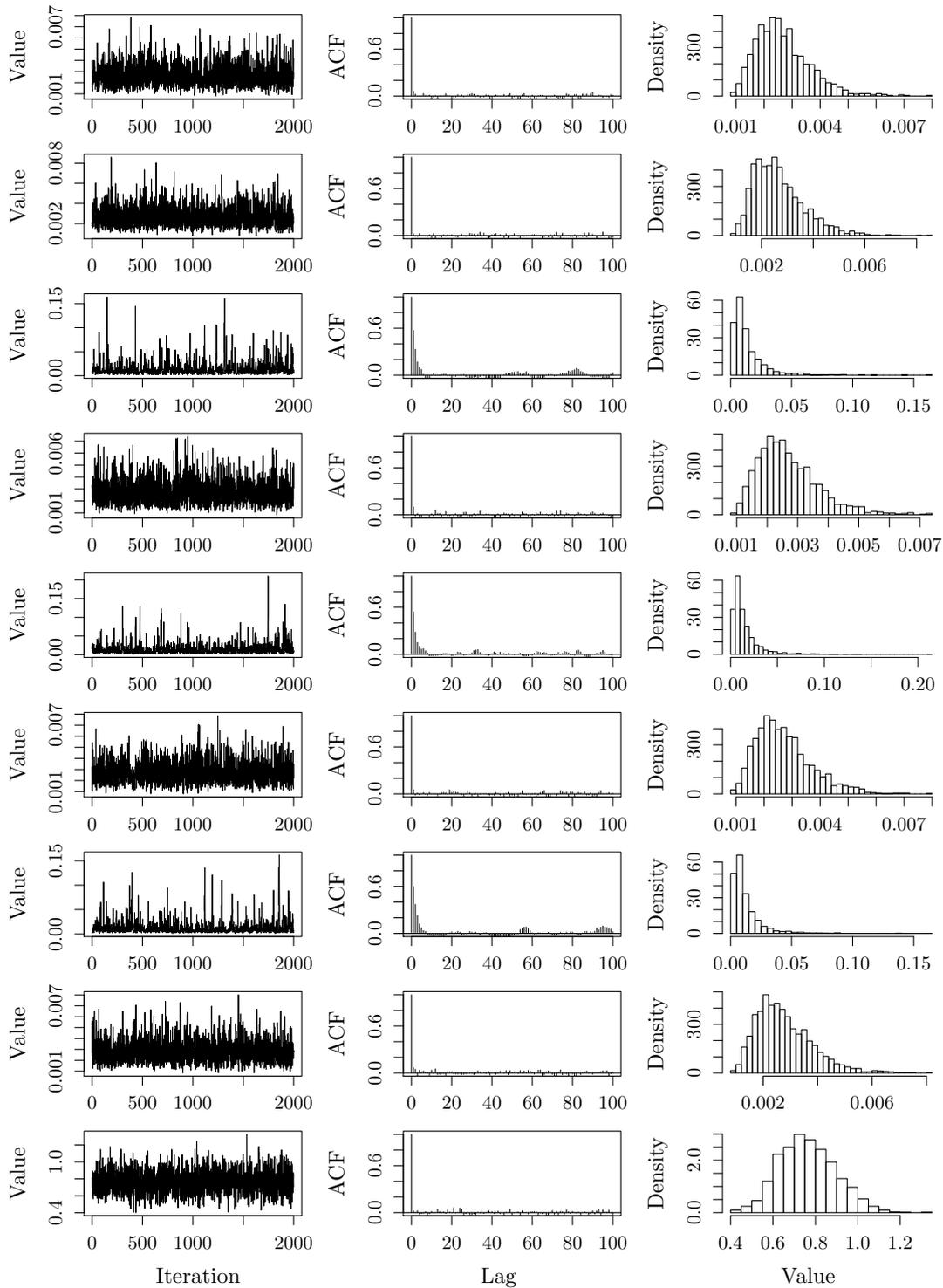


Figure 5.13: Trace plots, ACF plots and histograms of the posterior draws for the diagonal elements of W for the local Fourier DLM.

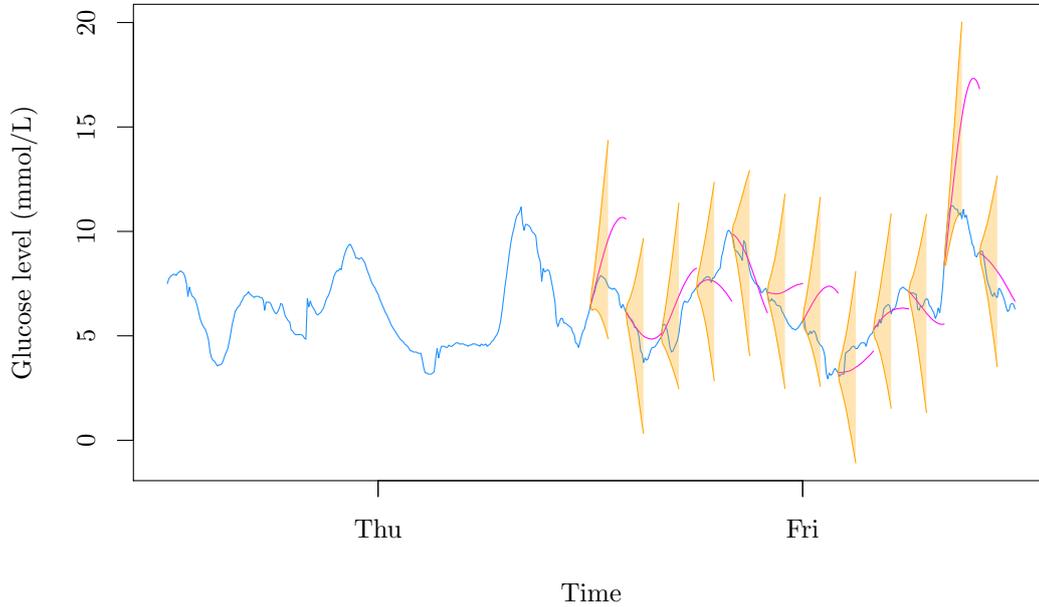


Figure 5.14: Two-hour-ahead forecasts (magenta) for the local Fourier DLM fitted to Patient 3's glucose levels (blue), with 95% credible intervals (gold).

5.3.3 A constrained variance model

The main problem with the local Fourier DLM is that the posterior parameter distributions of the even Fourier nodes in W are centred about large values, corresponding to a large variance in the evolution of the corresponding states. In turn this leads to a large credible interval around the forecasts. One way to prevent this from happening is to constrain the pairs of variances in the Fourier terms of W to be equal. This is reasonable because the $\{2(j-1)+1\}$ -st and $2j$ -th components in the seasonal part of the state vector correspond to the j -th harmonic and its conjugate. Each is composed of terms which oscillate at the same frequency and so it seems justifiable to assume they have a common innovation variance. In doing so, W becomes:

$$W = \text{diag}(\psi_1^{-1}, \psi_2^{-1}, \psi_2^{-1}, \dots, \psi_{q+1}^{-1}, \psi_{q+1}^{-1}).$$

The adapted local Fourier DLM is shown in Model 5.4.

Model 5.4

$$\begin{aligned} y_t | \theta_t, V &\sim N(F\theta_t, V), \\ \theta_t | \theta_{t-1}, W &\sim N_{2q+1}(G\theta_{t-1}, W), \\ F &= \begin{bmatrix} 1 & 1 & 0 & 1 & \dots & 0 \end{bmatrix}, \quad G = \text{blockdiag}(1, H_1, \dots, H_q), \end{aligned}$$

$$\begin{aligned}
 \text{where } H_j &= \begin{bmatrix} \cos \omega_j & \sin \omega_j \\ -\sin \omega_j & \cos \omega_j \end{bmatrix} \text{ and } \omega_j = \frac{2\pi j}{s} \text{ for } j = 1, \dots, q, \\
 V &= \psi_y^{-1}, \quad W = \text{diag}(\psi_1^{-1}, \psi_2^{-1}, \psi_2^{-1}, \dots, \psi_{q+1}^{-1}), \\
 \psi_y &\sim Ga(a_1, b_1), \\
 \psi_i &\sim Ga(a_2, b_2), \text{ for } i = 1, \dots, q + 1.
 \end{aligned}$$

The novel calculation of the full conditional distributions of the constrained pairs of variances is carried out by combining the prior for ψ_i^{-1} , for $i = 2, \dots, q + 1$ with the likelihood for the corresponding elements of the state vector: $\theta_{t,2i:2i+1} | \theta_{t-1}, \psi_i^{-1} \sim N_2([G\theta_{t-1}]_{2i:2i+1}, \psi_i^{-1} I_2)$:

$$\begin{aligned}
 \pi(\psi_i | \theta_{t,2i:2i+1}) &\propto \pi(\psi_i) \prod_{t=1}^T g(\theta_{t,2i:2i+1} | \theta_{t-1}, \psi_i), \\
 &\propto \psi_i^{a_2-1} \exp[-b_2 \psi_i] \prod_{t=1}^T |\psi_i I_2|^{1/2} \\
 &\quad \exp \left[-\frac{1}{2} (\theta_{t,2i:2i+1} - [G\theta_{t-1}]_{2i:2i+1})' \psi_i I_2 (\theta_{t,2i:2i+1} - [G\theta_{t-1}]_{2i:2i+1}) \right], \\
 |\text{diag}(a_1, \dots, a_n)| &= \prod_{i=1}^n a_i, \text{ so} \\
 &\propto \psi_i^{a_2+T-1} \exp \left[-\psi_i \left\{ b_2 + \right. \right. \\
 &\quad \left. \left. \frac{1}{2} \sum_{t=1}^T (\theta_{t,2i:2i+1} - [G\theta_{t-1}]_{2i:2i+1})' (\theta_{t,2i:2i+1} - [G\theta_{t-1}]_{2i:2i+1}) \right\} \right],
 \end{aligned}$$

then

$$\begin{aligned}
 \psi_i | \theta_{t,2i:2i+1} &\sim Ga \left(a_2 + T, \right. \\
 &\quad \left. b_2 + \frac{1}{2} \sum_{t=1}^T (\theta_{t,2i:2i+1} - [G\theta_{t-1}]_{2i:2i+1})' (\theta_{t,2i:2i+1} - [G\theta_{t-1}]_{2i:2i+1}) \right). \quad (5.8)
 \end{aligned}$$

The block Gibbs sampler in Algorithm 3.7 is updated to reflect the variance constraint in Algorithm 5.3.

Results

Algorithm 5.3 was run for $N = 200,000$ iterations, plus a burn in of 1,000 and thinned by 20 to produce a posterior sample of 10,000 per parameter. The prior hyperparameters are the same as for the local Fourier DLM. Four harmonics were once again chosen to model the seasonal component of the glucose data. The MCMC scheme ran in approximately

Algorithm 5.3: Block Gibbs sampler for a local Fourier DLM

1. Initialise $\psi_y^{(0)}$ and $\psi_j^{(0)}$ for $j = 1, \dots, q + 1$;
2. **for** $i = 1, \dots, N$:
3. run the forward filtering backward sampling algorithm (Algorithm 3.6) to simulate $\underline{\theta}^{(i)}$;
4. draw $(\psi_y^{-1})^{(i)}$ from (3.11);
5. draw $(\psi_1^{-1})^{(i)}$ from (3.12);
6. draw $(\psi_j^{-1})^{(i)}$ from (5.8) for $j = 2, \dots, q + 1$;
7. **end**

one day.

The posterior draws for the Fourier component variances are now much smaller, Fig. 5.15, following a similar trace to the odd nodes in Model 5.3. As a result, the credible intervals on the first hour of the forecasts, Fig. 5.16, do not increase as rapidly as in the unconstrained DLM. The credible intervals also no longer include zero and negative glucose levels in the forecasts. The forecasts produced by Model 5.4 for a predictive horizon of one hour, the forecasting region within the shaded credible intervals, are very close to the observed series. Looking at a shorter predictive horizon is therefore one way that the reliability of forecasts could be improved further.

In comparison to the $\text{AR}(p)$ DLM forecasts, Model 5.4 is better at predicting the peaks and troughs in the glucose profile, whereas the $\text{AR}(p)$ models tend to underestimate or miss these.

The smoothed states (posterior means) that make up the DLM are shown in Fig. 5.18, along with the raw glucose data. The number of harmonics chosen to model the seasonal component must strike a balance between explaining enough of the variation in the data and smoothing over noise. Fig. 5.18 shows the smoothed local level component, S_1 , picks up the noise in the glucose data, and the smoothed harmonics ($S_{2:5}$) describe the periodicity. Adding additional harmonics did not noticeably change the predictive mean, but predictive variances became larger. The smoothed state for the fourth harmonic, S_5 , is similar to that of the series corresponding to the second complex root, C_2 in Fig. 5.12. The amplitude of S_5 , however, is larger, since the local component is no longer describing the seasonality in the data.

5.3.4 $\text{AR}(p)$ Fourier DLMs

The models developed so far have produced reasonable results, but have room for improvement. The seasonal constrained variance local Fourier DLM captures fluctuations in glucose levels well, but has a large predictive variance, whilst the $\text{AR}(p)$ DLM and traditional $\text{AR}(p)$ models do not adequately capture the seasonality in the data. A possible

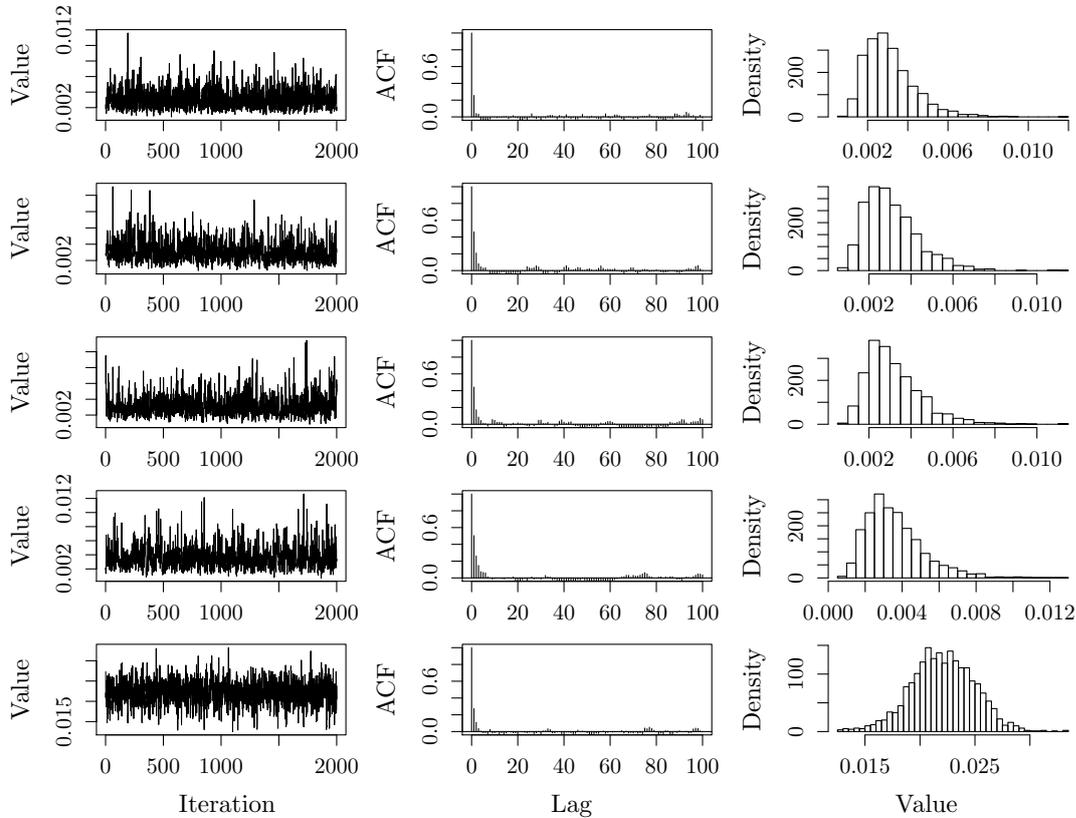


Figure 5.15: Trace plots, ACF plots and histograms of the posterior draws for the diagonal elements of W for the constrained variance local Fourier DLM.

improvement to the models is to replace the local level component, which evolves according to a random walk, with a stationary AR process which is therefore mean-reverting. This is more in keeping with the belief that the overall mean glucose level is constant in time and may reduce the predictive variance of long-term forecasts. A lower order AR component is more likely to produce a useful model here, as higher order AR models can capture pseudo-seasonal behaviour and will therefore be attempting to describe the same aspects of the data as the Fourier harmonics. A higher order AR component could however reduce the need for as many harmonics.

Model 5.5

$$y_t | \theta_t, V \sim N(F\theta_t, V),$$

$$\theta_t | \theta_{t-1}, W \sim N_{2q+p+1}(G\theta_{t-1}, W),$$

$$F = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 & 1 & \dots & 0 \end{bmatrix}, \quad G = \text{blockdiag}(1, \Phi, H_1, \dots, H_q),$$

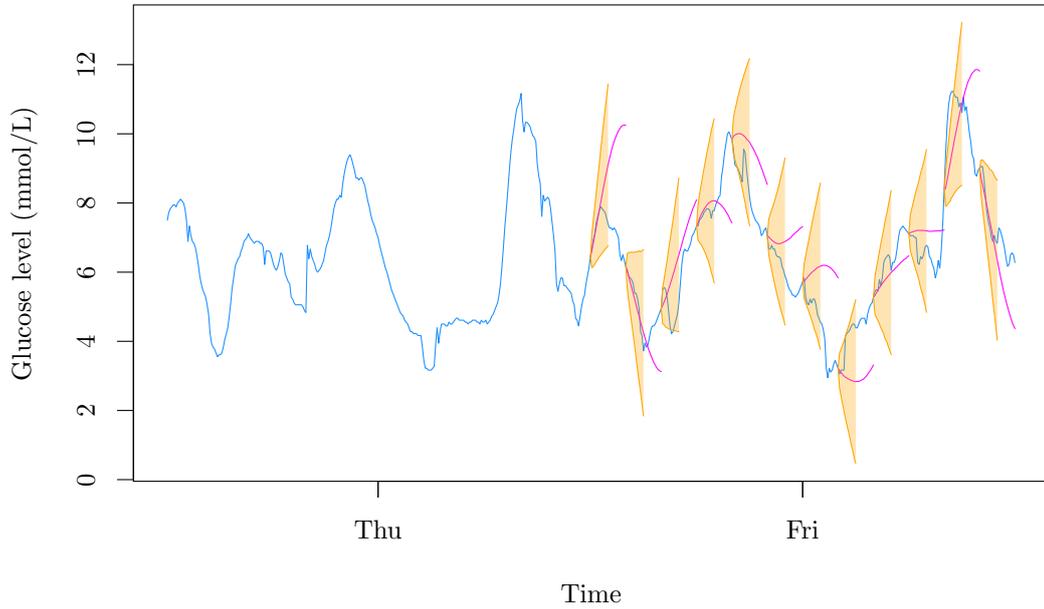


Figure 5.16: Two-hour-ahead forecasts (magenta) for the constrained variance local Fourier DLM fitted to Patient 3's glucose levels (blue), with 95% credible intervals (gold).

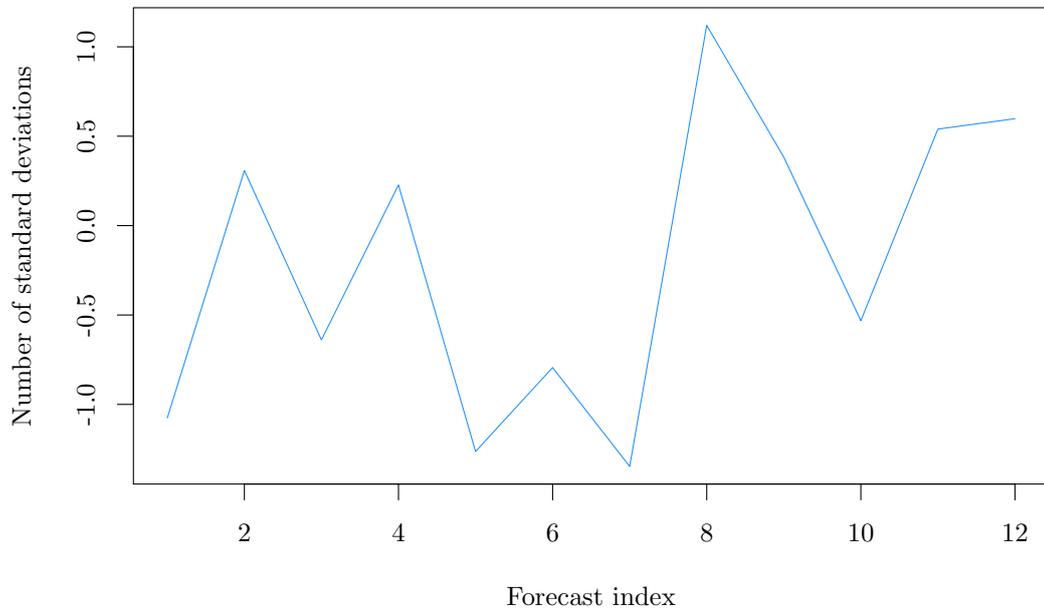


Figure 5.17: Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for Model 5.4.

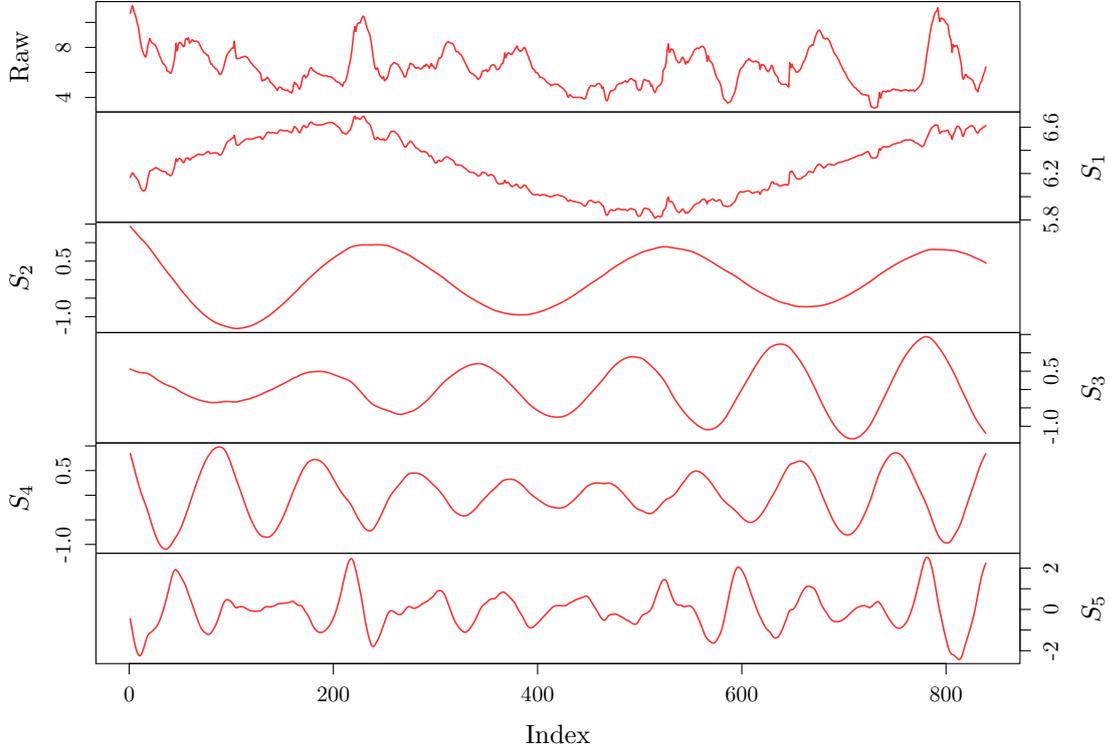


Figure 5.18: The first four days of raw glucose data (top) and the corresponding smoothed states. The second row corresponds to the local level component, θ_1 , and the following rows are one of each of the Fourier harmonic components, θ_3 , θ_5 , θ_7 and θ_9 , generated by the constrained variance local Fourier DLM.

$$\text{where } H_j = \begin{bmatrix} \cos \omega_j & \sin \omega_j \\ -\sin \omega_j & \cos \omega_j \end{bmatrix} \text{ for } j = 1, \dots, q, \text{ and } \Phi = \begin{bmatrix} \phi_1 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \phi_p & 0 & \dots & 0 \end{bmatrix},$$

$$V = \psi_y^{-1}, \quad W = \text{diag}(0, \sigma^2, 0, \dots, 0, \psi_1^{-1}, \dots, \psi_{q+1}^{-1}),$$

$$\psi_y \sim Ga(a_1, b_1),$$

$$\sigma^2 \sim Ga(a_2, b_2),$$

$$\psi_i \sim Ga(a_3, b_3), \text{ for } i = 1, \dots, q + 1.$$

(a) *Imposed stationarity, $p = 1$:*

$$\phi \sim Be(a_4, b_4);$$

(b) *Normal prior, $p \geq 1$:*

$$\underline{\phi} \sim N_p(\underline{m}, \Lambda^{-1}).$$

Model 5.5 describes an $AR(p)$ Fourier DLM with two prior specifications for the AR coefficients. The first, Model 5.5 (a), is for a stationary $AR(1)$ coefficient, $\phi \sim Be(a_4, b_4)$.

In the AR models fitted thus far, stationarity has been imposed by transforming AR coefficients to partial autocorrelations and, in the DLM case, by a zero innovation variance on the mean. In an AR(1) model, the Beta prior can be assigned directly to ϕ , as this is equal to the lag one partial autocorrelation. It is assumed that the partial autocorrelation is positive, so a transformation to the $(-1, 1)$ range is not deemed necessary. For this model, a MH step is needed for ϕ . This is a simplified version of the MH step in Algorithms 5.1 and 5.2; $\tilde{\phi}$ is drawn from a Beta proposal distribution $Be(\lambda\phi + \delta, \lambda(1 - \phi) + \delta)$, where λ is a tuning parameter and δ is a nudge parameter. The full conditional distribution for ϕ is then combined with the proposal to evaluate the acceptance probability. The novel full conditional is derived as:

$$\begin{aligned} \pi(\phi|\cdot) &\propto \pi(\phi) \prod_{t=1}^T p(\theta_{t,2}|\cdot), \\ &\propto \phi^{a_4-1} (1-\phi)^{b_4-1} \prod_{t=1}^T \exp\left\{-\frac{1}{\sigma^2}(\theta_{t,2} - \phi\theta_{t-1,2})^2\right\}, \\ &\propto \phi^{a_4-1} (1-\phi)^{b_4-1} \exp\left\{-\frac{1}{\sigma^2} \sum_{t=1}^T (\theta_{t,2} - \phi\theta_{t-1,2})^2\right\}. \end{aligned} \quad (5.9)$$

The full Metropolis within Gibbs scheme is described in Algorithm 5.4.

Algorithm 5.4: Metropolis within Gibbs scheme for an AR(1) Fourier DLM	
1.	Initialise $\psi_y^{(0)}$, $\sigma^{-2(0)}$, $\phi^{(0)}$ and $\psi_j^{(0)}$ for $j = 1, \dots, q$;
2.	for $i = 1, \dots, N$:
3.	run the forward filtering backward sampling algorithm (Algorithm 3.6) to simulate $\underline{\theta}^{(i)}$;
4.	draw $(\psi_y^{-1})^{(i)}$ from (3.11);
5.	draw $(\sigma^{-2})^{(i)}$ from (3.12);
6.	draw $(\psi_j^{-1})^{(i)}$ from (5.8) for $j = 1, \dots, q$;
7.	generate a proposal $\tilde{\phi}$ from $q(\phi^{(i-1)}, \tilde{\phi})$;
8.	evaluate the acceptance probability
	$A(\phi^{(i-1)}, \tilde{\phi}) = \min\left(1, \frac{\pi(\tilde{\phi} \cdot)q(\tilde{\phi}, \phi^{(i-1)})}{\pi(\phi^{(i-1)} \cdot)q(\phi^{(i-1)}, \tilde{\phi})}\right);$
9.	accept $\tilde{\phi}$ with probability $A(\phi^{(i-1)}, \tilde{\phi})$: set $\phi^{(i)} = \tilde{\phi}$, otherwise set $\phi^{(i)} = \phi^{(i-1)}$;
10.	end

As the complexity of the DLM increases, replacing the AR(1) component with an AR(p) component for $p \geq 1$, Model 5.5 (b) assigns a Normal prior to the AR coefficients, $\underline{\phi}$, to take advantage of the semi-conjugate relationships this creates, thus reducing the

complexity of the MCMC algorithm. The cost is that the guarantee that the AR component is mean-reverting is lost. Model 5.5 is fitted for $p = 1$ in scenarios (a) and (b) to compare the effect of removing the stationarity condition from the coefficient sample space. Higher orders are then fitted using the Normal prior. The full conditional distribution for the AR coefficients under Model 5.5 (b) is derived as:

$$\begin{aligned} \pi(\underline{\phi}|\cdot) &\propto \pi(\underline{\phi}) \prod_{t=p+1}^T p(\theta_{t,2}|\cdot), \\ &\propto \exp \left[-\frac{1}{2}(\underline{\phi} - \underline{m})' \Lambda (\underline{\phi} - \underline{m}) \right] \prod_{t=p+1}^T \exp \left[-\frac{1}{2\sigma^2}(\theta_{t,2} - X_t \underline{\phi})^2 \right], \\ &\propto \exp \left[-\frac{1}{2} \left\{ \underline{\phi}' \Lambda \underline{\phi} - 2\underline{m}' \Lambda \underline{\phi} + \frac{1}{\sigma^2} \sum_{t=p+1}^T (\underline{\phi}' X_t' X_t \underline{\phi} - 2\theta_{t,2} X_t \underline{\phi}) \right\} \right], \\ &\propto \exp \left[-\frac{1}{2} \left\{ \underline{\phi}' \left(\Lambda + \frac{1}{\sigma^2} \sum_{t=p+1}^T X_t' X_t \right) \underline{\phi} - 2 \left(\underline{m}' \Lambda + \frac{1}{\sigma^2} \sum_{t=p+1}^T \theta_{t,2} X_t \right) \underline{\phi} \right\} \right], \end{aligned}$$

so

$$\underline{\phi}|\cdot \sim N_p \left(\left(\Lambda + \frac{1}{\sigma^2} \sum_{t=p+1}^T X_t' X_t \right)^{-1} \left(\underline{m}' \Lambda + \frac{1}{\sigma^2} \sum_{t=p+1}^T \theta_{t,2} X_t \right), \left(\underline{m}' \Lambda + \frac{1}{\sigma^2} \sum_{t=p+1}^T \theta_{t,2} X_t \right) \right). \quad (5.10)$$

The multivariate Normal target distribution in Eq. 5.10 means that $\underline{\phi}$ can be sampled in a single draw, rather than the one-at-a-time MH sampling employed when using the partial autocorrelation parametrisation. The block Gibbs sampler for the Normal model is described in Algorithm 5.5.

Algorithm 5.5: Block Gibbs sampler for a AR(p) Fourier DLM

1. Initialise $\psi_y^{(0)}$, $\sigma^{-2(0)}$, $\underline{\phi}^{(0)}$ and $\psi_j^{(0)}$ for $j = 1, \dots, q$;
2. **for** $i = 1, \dots, N$:
3. run the forward filtering backward sampling algorithm (Algorithm 3.6) to simulate $\underline{\theta}^{(i)}$;
4. draw $(\psi_y^{-1})^{(i)}$ from (3.11);
5. draw $(\sigma^{-2})^{(i)}$ from (3.12);
6. draw $(\psi_j^{-1})^{(i)}$ from (5.8) for $j = 1, \dots, q$;
7. draw $\underline{\phi}^{(i)}$ from (5.10);
8. **end**

Stationary AR(1) results

Algorithm 5.4 was run for $N = 300,000$ iterations, with additional burn in of 1,000 and a thin of 30. The model hyperparameters were chosen similarly to the rest of the models; $a_1 = b_1 = 1$, a $Ga(1.1, 0.01)$ prior was assigned to σ^{-2} and the ψ_i , for $i = 1, \dots, q + 1$, and a $Be(2, 2)$ prior was assigned to ϕ . This was chosen in favour of the uniform prior ($a_4 = b_4 = 1$) used earlier for the partial autocorrelations to make values near the boundaries at zero and one less credible. The tuning parameter $\lambda = 500$ and a nudge of $\delta = 0.001$ was used. The resulting acceptance rate was 0.62.

The results from fitting Model 5.5 (a) are not drastically different from those of Model 5.4. The posterior density for ϕ is bimodal, shown in Fig. 5.19. The majority of draws are from the top end of the possible values for ϕ , centred about 0.9, but there is a cluster of draws from the bottom end of the sample space, too. An AR(1) coefficient of one is equivalent to a random walk process, which recovers the local Fourier DLM, so it is not surprising that the two models have similar marginal posteriors for the elements of V and W , Fig. 5.20. A random walk is a non-stationary process, so values of ϕ close to one allow the mean-reverting process to wander as much as possible whilst still being considered stationary. The small mode at zero for ϕ corresponds to a white noise process, which is made up of i.i.d., zero-mean Gaussian variables. Here, the algorithm is only identifying the Fourier harmonics, plus some noise. As this is a smaller mode, the model can be assumed to be mostly similar to the local Fourier DLM. The benefit of having the AR(1) component in the model is marginal, due to the large value for ϕ , however it can be seen in longer term forecasts; the stationarity that the component imposes on the process prevents forecasts for longer predictive horizons from increasing without bound. This also benefits the credible interval on forecasts, which otherwise also increase quickly in width. Plots of forecasts from an AR(1) Fourier DLM with a Normal prior on ϕ are discussed in the next section.

Normal AR(p) results

The Normal AR(p) model is fitted for $p = 1-9$. The hyperparameters used for running Algorithm 5.5 are the same as for Algorithm 5.4, but with the prior hyperparameters for $\underline{\phi}$ consistent with Model 5.5 (b): $\underline{m} = 0.51_p$ and $\Lambda^{-1} = 2I_p$. The number of iterations were: 300,000, 20,000, 20,000, 100,000, 200,000, 200,000, 200,000, 200,000 and 200,000 for orders 1 to 9. Each algorithm had a burn in of 1,000 and was thinned to 10,000 observations. Because of the model complexity, the MCMC schemes for these models took approximately one to two days to run.

The posterior coefficient means and 95% posterior intervals are shown in Fig. 5.21. The first thing to note here is the posterior density of ϕ in the AR(1) Fourier DLM; this

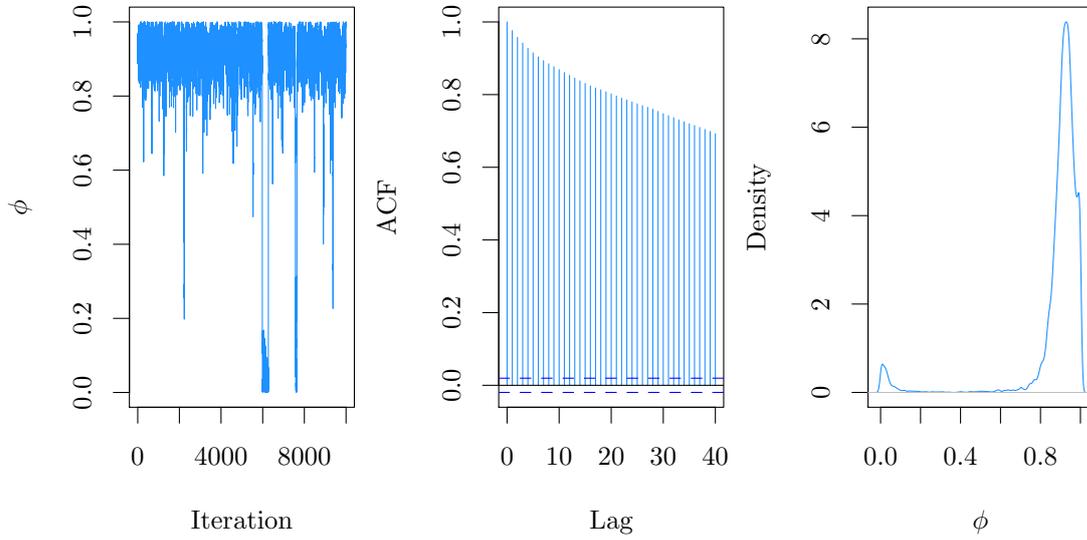


Figure 5.19: Posterior trace, ACF and density plot of ϕ for Model 5.5 (a).

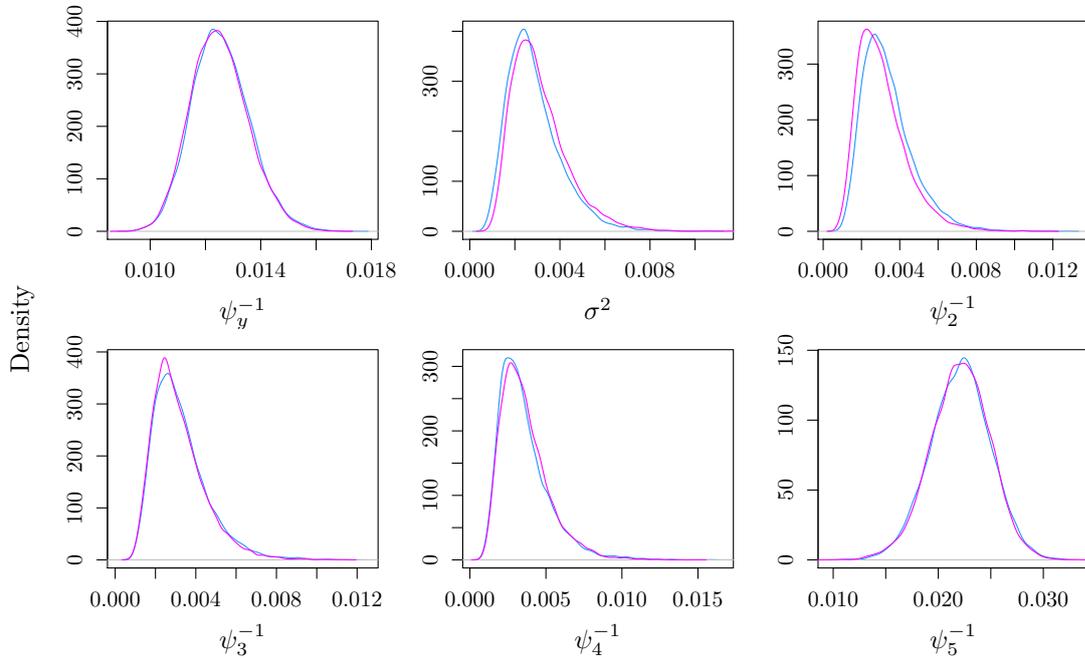


Figure 5.20: Posterior density plots of V and the non-zero diagonal elements of W for Model 5.5 (a) (blue) and Model 5.4 (magenta).

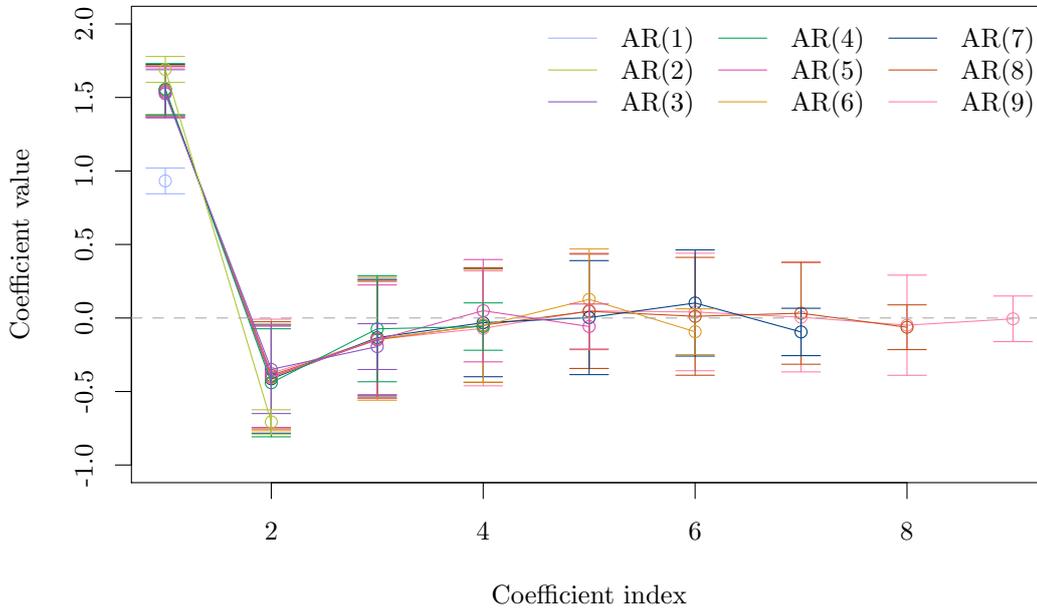


Figure 5.21: Model 5.5 (b) AR coefficient posterior means and 95% posterior interval for orders $p = 1-9$.

is centred in the same range of posterior values that Model 5.5 (a) drew for ϕ , and so the data are suggestive of a stationarity process. The higher order model coefficients follow a similar shape to the coefficients of the traditional $AR(p)$ models fitted in Section 5.2, with a large value for ϕ_1 , a non-zero value for ϕ_2 and densities centred about zero or small values for the coefficients thereafter.

The corresponding roots to the characteristic equations described by the posterior means of the AR coefficients in each of the AR Fourier DLMs are shown plotted in Fig. 5.22. The $AR(p)$ DLMs for orders $p \geq 4$ showed pseudo-seasonal behaviour, i.e. had complex roots. The same is true for the $AR(p)$ Fourier DLMs of order $p \geq 4$ here, though the magnitude of the imaginary components is smaller than in the $AR(p)$ DLMs, which is likely because the Fourier harmonics are describing the seasonality, too. The forecasts from these models therefore may be less useful. All of the roots fall within the unit circle, which suggests that the coefficients from each model fall largely within the stationary region.

The trace plots for the observation and non-zero system variance parameters are plotted in Fig. 5.23 for each model order. The main difference seen between the results is in moving from a first order AR process to higher orders. The observation variance for all model orders is the same, but the variance explained by different components of the underlying system changes. For the first order model, the variance of the AR process,

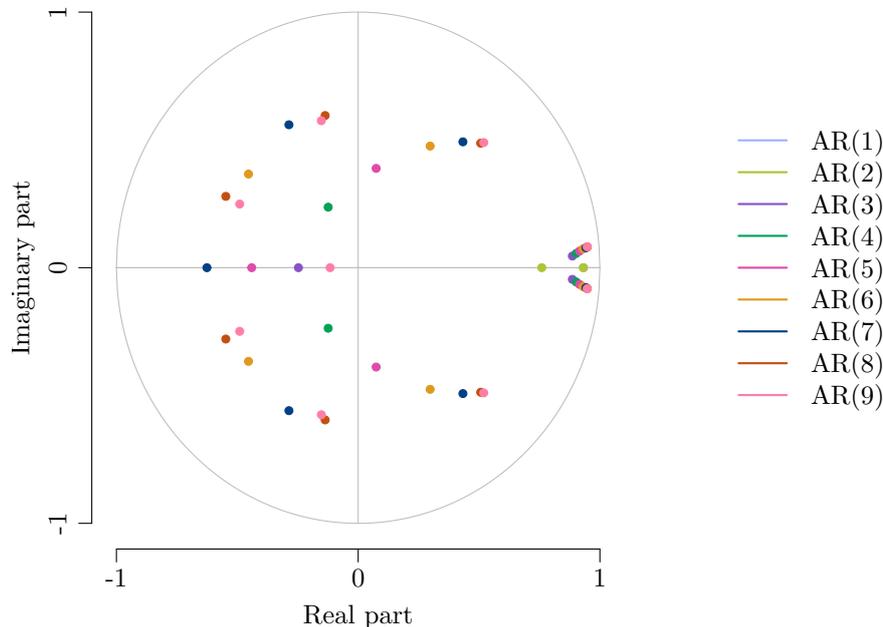


Figure 5.22: Model 5.5 (b) AR component roots for the mean posterior coefficients for orders $p = 1-9$.

σ^2 , is small and the variances of the Fourier harmonics have much longer tails, whereas for orders $p \geq 2$ the variance for the AR process is higher and the posterior draws for the Fourier harmonic variances are more concentrated at small values. In particular, the fourth harmonic has a larger variance for the AR(1) Fourier DLM, which is most similar to the local Fourier DLM. For these DLMs, the fourth harmonic soaks up a lot of the residual periodic variation that is not captured by the rest of the model, which is seen in the plot of the smoothed states for the local Fourier DLM, Fig. 5.18. The higher order AR component instead describes something akin to this periodic variation in the remaining AR(p) Fourier DLMs, resulting in a smaller variance for the last pair of states.

The mean forecast and 95% credible interval for two-hour-ahead out of sample predictive horizons for AR(p) Fourier DLMs of order one to six are shown in Fig. 5.24. For higher orders of $p > 6$, the credible intervals on forecasts rapidly increased as the forecasting index increased and the forecasts oscillated between large positive and negative glucose values, making the results unreliable. This shows that such models were overfitting the data and the problem was likely caused by the pseudo-seasonal behaviour being confounded with the Fourier harmonics. For the models shown, however, the forecasts look good. The first order Fourier DLM produces forecasts similar to the local Fourier DLM, as expected from the results discussed so far, whilst the higher order Fourier models produce forecast more similar to the AR(p) DLMs. The AR component appears to have had the

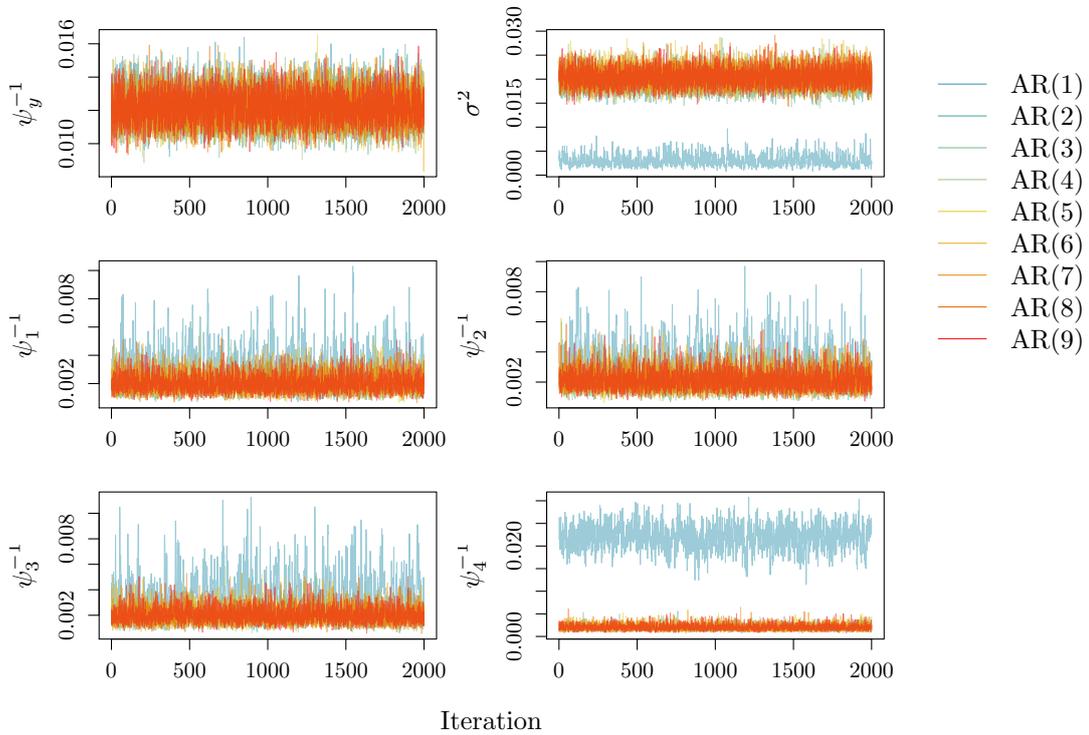


Figure 5.23: Trace plots for ψ_y^{-1} , σ^2 and $\psi_{1:4}^{-1}$ for AR models of order $p = 1-9$. Plots are thinned by five.

effect of dampening the seasonality in the forecasts, which at several forecast indices has improved the predictive performance. This is seen quantitatively in Fig. 5.25, where the mean number of standard deviations of the mean forecast from the observed time series is plotted. Each model has a similarly sized credible interval about their mean forecasts, whilst the higher order AR Fourier DLMs consistently produce mean forecasts that are fewer standard deviations from the observed series.

A closer look at the smoothed states and the series corresponding to the roots of the AR component of the AR(6) Fourier DLM, using the posterior means of the model parameters, is shown in Fig. 5.26. The Fourier harmonics in this model are clearly describing periodic behaviour in the series, with the local level and AR component describing the residual perturbations in glucose levels. The three complex pairs of roots, C_1 , C_2 and C_3 , do not appear to be picking up any specific seasonality in the data, but instead represent irregular patterns in the data. The third complex root shows a similar pattern to C_2 in the AR(4) DLM. There is still some seasonality left over in the AR component, shown by the seasonal behaviour of the local level component (top).

The models fitted in this section have shown how an AR process can be used alongside a Fourier seasonal component to control the predictive variance in longer term forecasts.

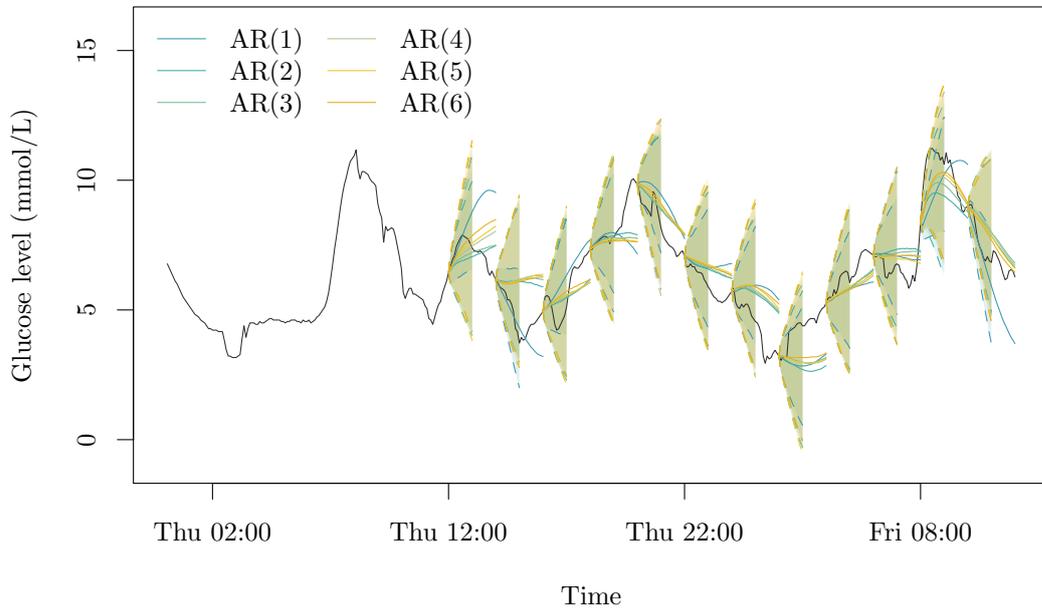


Figure 5.24: Two-hour-ahead forecasts (solid) for $AR(p)$ Fourier DLMs with $p = 1-6$ fitted to Patient 3's glucose levels, with 95% credible intervals (dashed).

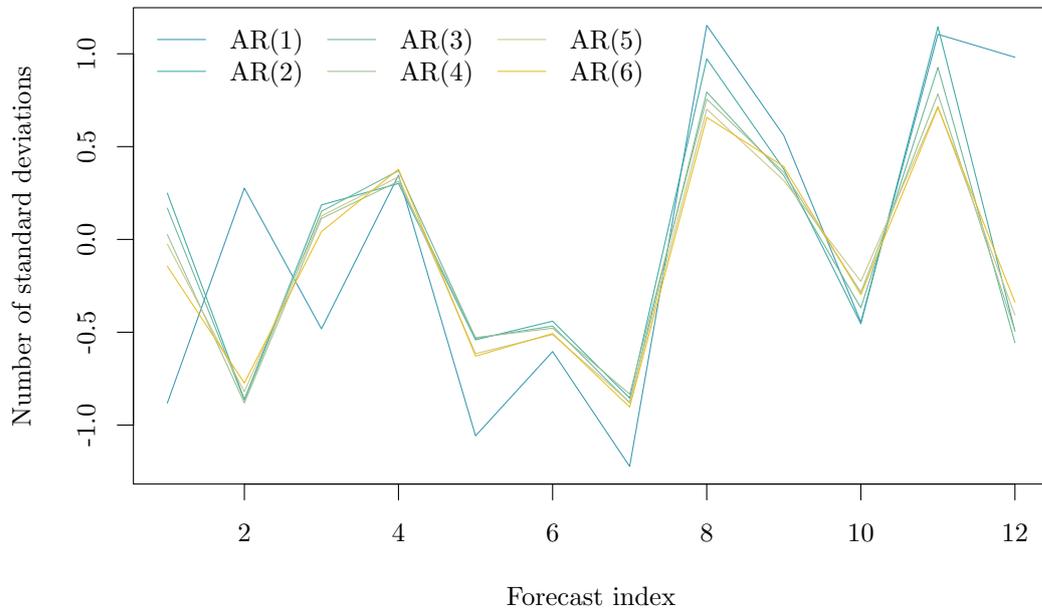


Figure 5.25: Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for $AR(p)$ Fourier DLMs with $p = 1-6$.

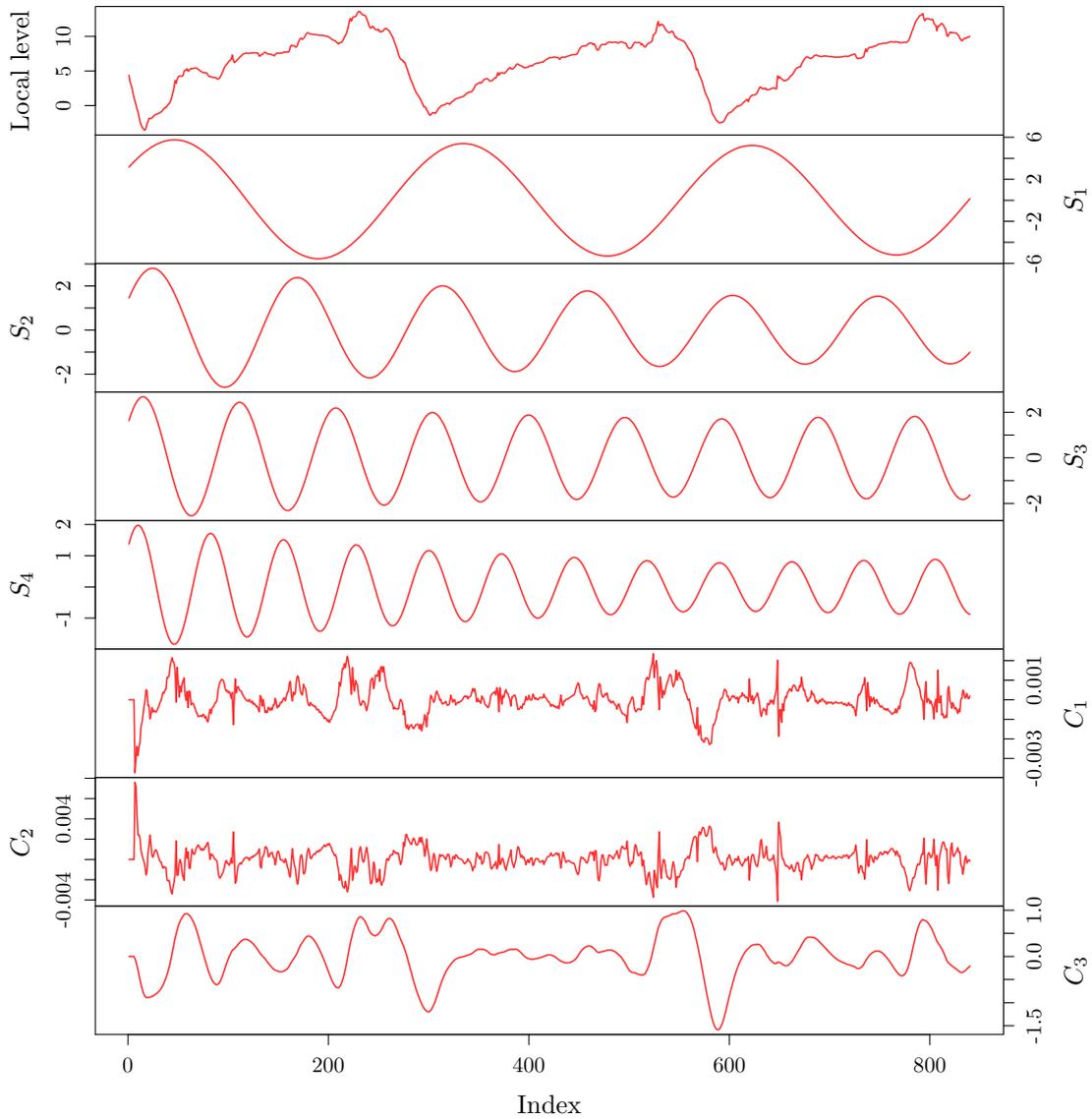


Figure 5.26: The AR(6) Fourier DLM for the first four days of Patient 3's glucose levels decomposed into the smoothed local level component, the smoothed harmonics and the series corresponding to the complex pairs of roots.

Whilst there have been improvements in how close the mean forecasts are to the observed series as the order of the AR component increases, this reaches a limit for orders greater than six, which corresponds to glucose levels greater than a 30 minute lag. The forecast improvements are marginal in relation to the added model complexity and still produce large credible intervals.

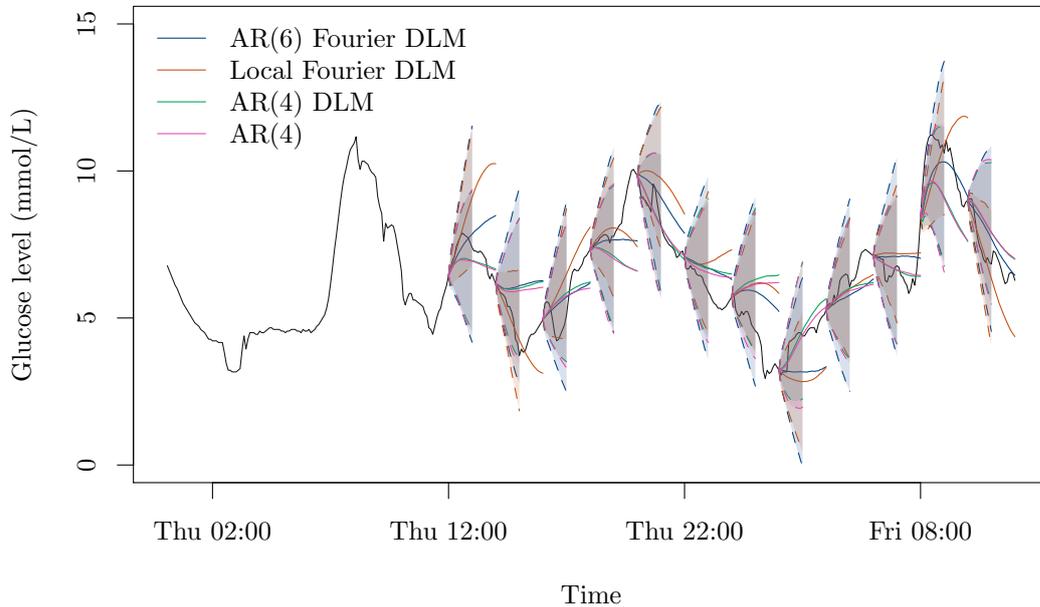


Figure 5.27: Two-hour-ahead forecasts for glucose models fitted to Patient 3's glucose levels, with 95% credible intervals.

5.4 Comparison of models

An overall comparison between the traditional AR(4) model, the AR(4) DLM, the constrained variance local Fourier DLM and the AR(6) Fourier DLM results is presented here.

Fig. 5.27 shows the mean two-hour-ahead forecasts for each of the models, with 95% credible intervals. The figure highlights the benefit of using a Fourier component to describe the periodicity in glucose over a higher order AR component alone, as the local Fourier DLM and the AR(6) Fourier DLM track the upcoming peaks in glucose levels more closely, at forecast indices 1 and 11. The local Fourier DLM also accurately predicts glucose levels at index 2, which all of the other models miss. Forecasts from the traditional AR(4) model and AR(4) DLM tend towards the mean, so only capture glucose levels well when they are trending in this direction. The number of standard deviations each mean forecast is from observed glucose levels for each model is shown in Fig. 5.28. This confirms what is seen in Fig. 5.27, that the Fourier model forecasts are, in general, closer to the observed glucose levels than the AR models. At a glance, it seems as though the AR(6) Fourier DLM performs better than the local Fourier DLM as the number of standard deviations is smaller, however the credible intervals on these forecasts is larger. The credible intervals on the local Fourier DLM are the narrowest across all of the models.

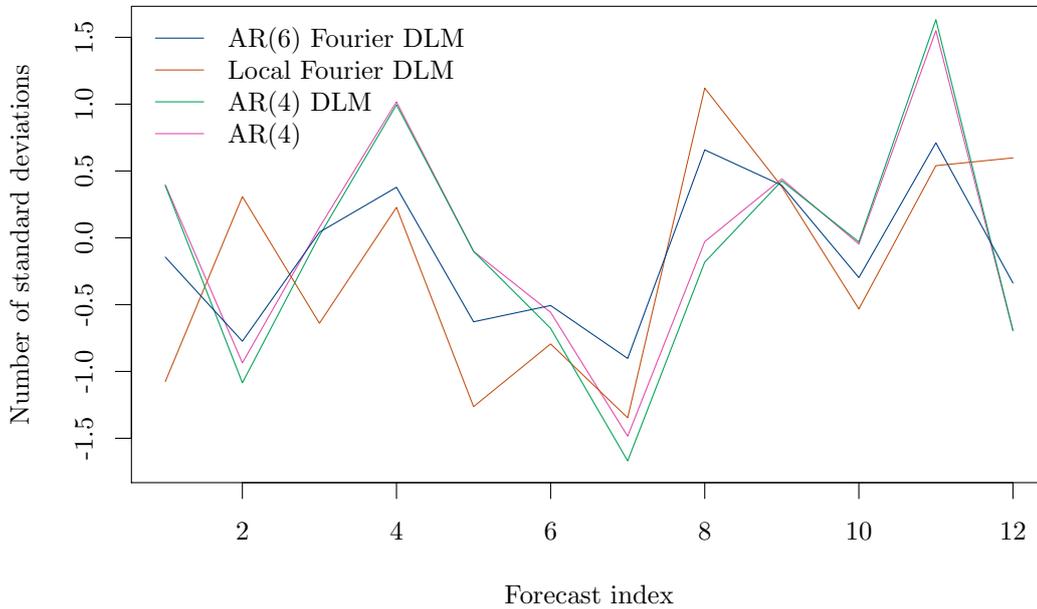


Figure 5.28: Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast from the glucose models presented.

The constrained variance local Fourier DLM is more likely to correctly predict an extreme glucose level that would require human behavioural intervention across all of the models discussed here and, overall, predicts glucose levels well. The credibility intervals of forecasts from this model are smaller than for the other models, whilst still including the actual glucose levels in these intervals. This model is chosen as the univariate model for glucose levels going forward.

5.5 Across-patient comparison

As with the models to classify activity levels, it is important to assess how well the chosen local Fourier DLM performs across patients. The models are fitted to the three non-diabetic patients, Patient 5 and Patient 10, as before.

The posterior densities of the model variance parameters are plotted in Fig. 5.29. It is clear from the plots that the variance parameters are different between patients, for both diabetic and non-diabetic individuals. Unlike the within-state distributions of the HMM describing the hidden activity zones, the parameters for the DLMs describing patient glucose levels must therefore be learned separately. It is not surprising that these parameters differ so much between patients, since the way glucose levels change over time is largely dependent on factors that are likely to vary a great deal, such as diet and exercise.

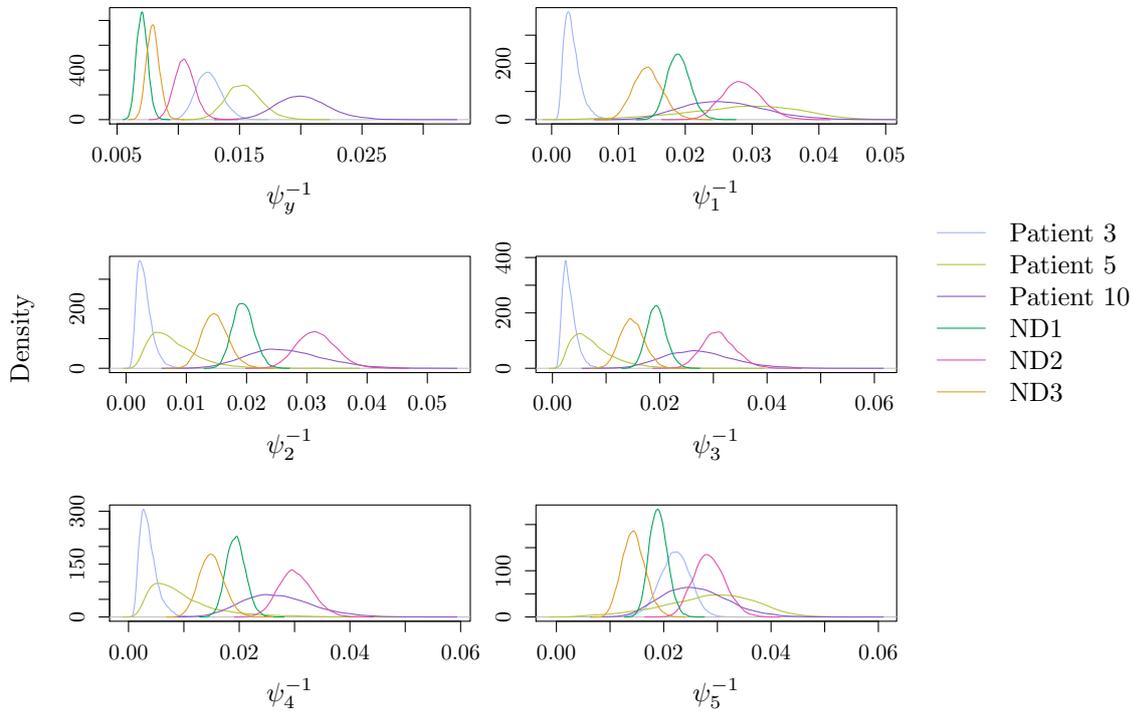


Figure 5.29: Posterior densities for ψ_y^{-1} and $\psi_{1:5}^{-1}$ for Model 5.4 fitted to Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3's glucose levels.

The out of sample forecasts produced for each patient are plotted in Fig. 5.30. As seen with Patient 3 already, the model performs well, with the poorer forecasts appearing when the forecast is taken from an apex in glucose levels, for example forecast 2 for Patient 5. The forecasts for each patient are also more accurate for the first hour, which is to be expected, as forecasts in general become less reliable for longer predictive horizons. The large credible intervals that are seen in Patient 3 are also seen in the other patients.

It is worth noting here that the glucose profile for ND 2 is relatively flat, so the forecasts for this patient perform well. This patient was (voluntarily) fasting for this part of the glucose collection period, so has few large spikes in the corresponding part of their glucose profile. This has essentially removed part of the seasonality in the data (arising from regular meal times), making it easier to model. The univariate models developed so far, and namely the local Fourier DLM presented here, present a reasonable method for forecasting glucose levels for up to one hour ahead. For forecasts further into the future, like the two-hour-ahead forecasts explored here, a better informed model is necessary in order to be able to predict changes in glucose levels that otherwise do not follow the exact pattern of more recent glucose levels. Incorporating information on outside factors like activity levels or food intake is one way to provide this model improvement.

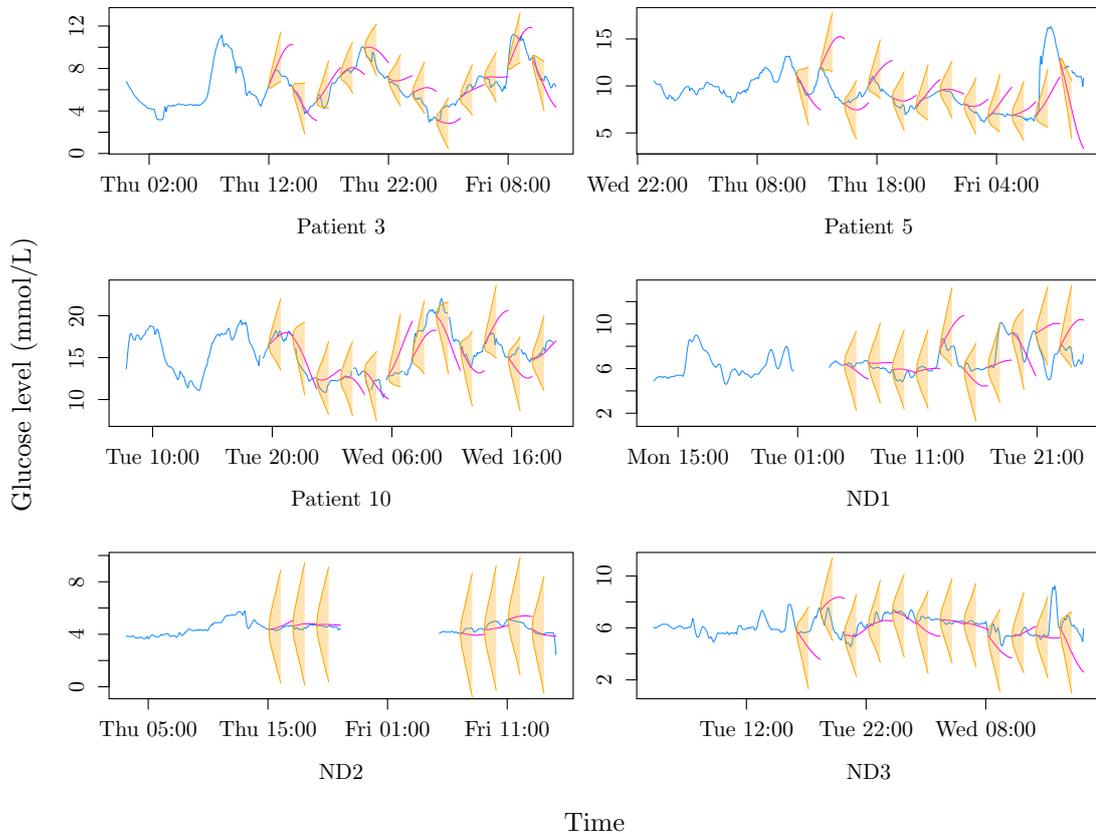


Figure 5.30: Two-hour-ahead forecasts (magenta) from Model 5.4 fitted to Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3's glucose levels, with 95% credible intervals (gold).

5.6 Summary

The aim of this chapter was to fit a model to glucose levels that could capture the periodicity of glucose over time and produce accurate forecasts. DLMs were explored for this aim, using two main approaches to model the seasonal glucose behaviour: a Fourier seasonal component and a high order AR component. Whilst the forecasts from models using an AR component were reasonable, they failed to predict most of the extreme glucose levels that would result in a behavioural prompt. The Fourier models were better at capturing these events, so the local Fourier model was chosen as the best baseline model to build upon in the subsequent joint modelling chapter.

Models combining the Fourier and higher order AR components in a DLM were also considered, but the added model complexity from the higher order AR component did not provide a significant enough improvement to forecasts. The pseudo-seasonal behaviour of the AR component was also something that needed to be used carefully, since unstable forecasts were seen for orders greater than seven.

Chapter 6

Bivariate glucose models

6.1 Background

The models for predicting glucose levels explored so far in this thesis have performed with varying success. Forecasts of glucose levels that perform better are where the behaviour of the glucose profile does not change unexpectedly, i.e. where glucose levels behave in a similar fashion as they have before. One way to improve forecasts to capture unexpected changes in glucose levels is to include some additional information on a variable that could influence such changes. This is done by jointly modelling the additional information alongside glucose levels.

It is well known that activity levels influence glucose levels, as discussed in the literature in Section 1.3.2. The relationship between these two variables is explored further in this chapter, then bivariate models are developed based on the results of the HMMs fitted to activity levels in Chapter 4 and the glucose models developed in Chapter 5. Additional to this, the relationship between glucose levels and food intake is modelled later in the chapter to verify the results obtained from the bivariate glucose and activity models. The MCMC schemes in this chapter typically take approximately one day to run. The models developed in this chapter are novel.

6.2 Further glucose and activity exploratory analysis

The BFEN activity summary with a log-Normal HMM emerged as the most consistent model-summary pairing to classify activity observations into activity zones in Chapter 4. The premise of the models developed in this chapter is to exploit the relationship between glucose and activity to fit a bivariate model that can provide better forecasts for glucose levels than when modelling glucose levels alone. As the relationship between glucose levels and activity levels is not linear, the posterior modal classification of each observation according to the HMM is used to learn more about the relationship between glucose and

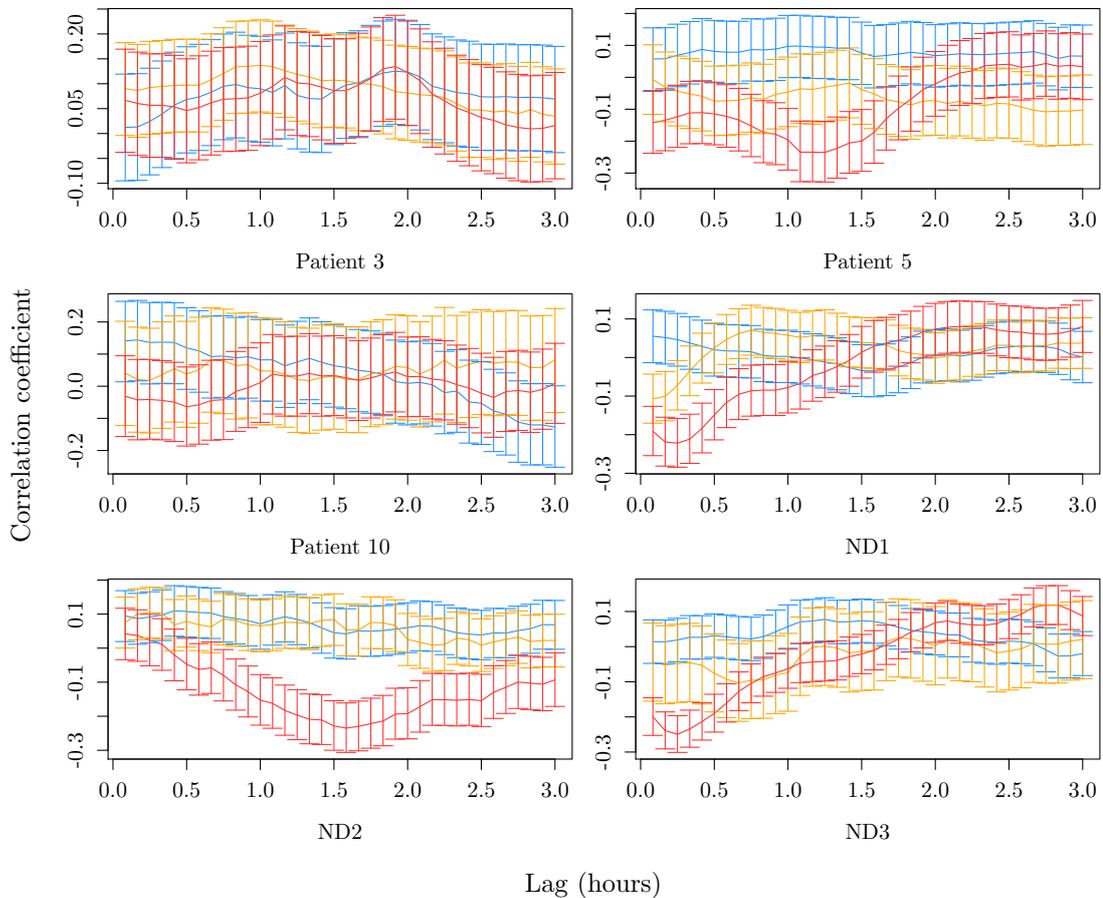


Figure 6.1: The lagged correlations between glucose and each BFEN posterior modal activity state (zone one in blue, zone two in gold and zone three in red) for Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3.

activity at different intensities of activity, which will help inform some of the bivariate modelling decisions.

Fig. 6.1 shows the correlation between glucose and lagged activity levels, given the posterior modal classification of each activity observation. The relationship between the glucose and activity data sets varies from patient to patient. One of the possible reasons for this is the other confounding factors that are influencing glucose alongside activity, such as food intake, which in turn will influence the correlation profiles. A difference between diabetic and non-diabetic patients' correlation profiles is expected, as diabetes is a disease that affects the body's response to controlling glucose levels. There seem to be three different relationships observed in the patient profiles shown, where the main difference between the profiles is in how the higher activity zone interacts with glucose levels. The two lower activity zones have a relatively flat profile centred at a positive, but very near zero, correlation for all patients, which is not surprising because they both represent

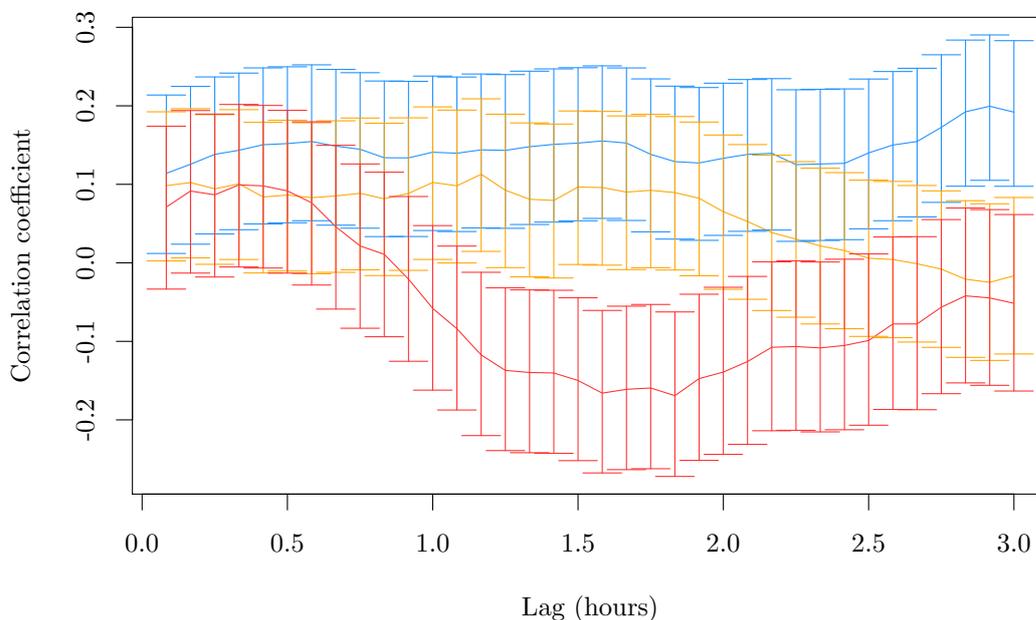


Figure 6.2: The lagged correlations between glucose and each BFEN posterior modal activity state (zone one in blue, zone two in gold and zone three in red) for ND 2's data recorded whilst fasting, with 95% confidence intervals.

low/inactivity. The profiles of Patient 3 and Patient 10 show a flat correlation profile for all three activity zones, Patient 5 and ND 2 show an increasingly negative correlation for the third zone, which troughs at a lag of around one hour and then increases towards zero again, and ND 1 and ND 3 share a sharp dip in correlation at a lag of around 15 minutes, which then slowly increases towards zero and to the same level of correlation as the two lower zones. The differences in each set of profiles means that a model that learns the relationship between the two variables in order to predict future glucose levels could look very different from patient to patient, and in particular between diabetic and non-diabetic people. An adaptable model is therefore necessary.

6.2.1 Glucose vs activity whilst fasting

One of the key features of ND 2's glucose profile is that for the last six days of data collection, the person was (voluntarily) fasting. As a result, in theory this should reveal the (almost) exact relationship between glucose and activity, at least for this patient, since food intake is the main factor that causes a change in glucose levels.

The correlations between ND 2's glucose and lagged activity levels according to the posterior modal activity zones for the data recorded whilst they were fasting are shown in Fig. 6.2. The negative curve for zone three that was seen for this patient in Fig. 6.1 is

shifted slightly, with a small positive correlation first observed at the lower lags before the curve dips. There is some evidence of this shape for the third zone in the correlation profile of Patient 5, but not in the other patients. An explanation for the lack of shape in the profiles of Patient 3 and Patient 10 is that the upper limit on the activity levels observed from them was the lowest out of the group of patients whose data have been modelled. Both patients record a similar, lower level of activity, which is seen in the density plot in Fig. 4.30. These patients also have a lower density of observations that fall into the third activity zone than the other zones, so the lack of shape could also be attributed to the lack of higher level observations from these patients altogether.

An expert in the application area advised that a low intensity activity, such as walking, has the desired effect of gradually reducing glucose levels up to around 30 minutes after the event, then the effect tapers off. When high intensity activity is performed, an initial increase in glucose levels is observed before they decrease one to two hours later. The latter is the effect observed in ND 2's data, whereas the former is similar to what is seen in ND 1 and ND 3. In the BFEN activity summary configuration, there is no distinction between low impact activity, such as walking, and higher impact activity, such as running, as both of these activities are likely to be classified into activity zone three, which explains why the lagged correlation profiles differ between patients if the higher levels of activity are not being performed as much. Observations in the two lower activity states are likely to represent sedentary activities.

6.3 DLMS for glucose data with known activity states

The univariate DLMS developed in Chapter 5, namely the local Fourier model, provide a good baseline in predictive performance to build upon for the bivariate models in this chapter. Activity data are incorporated into Model 5.4 via a regression component. The states of the activity observations are treated as known, taking the activity state at each time point to be the posterior modal classification given by the Normal HMM on the log-transformed BFEN data. This approach of treating the states as known is known as 'Markov melding', a method outlined by Goudie *et al.* (2019). Joining the activity and glucose models in this way reduces the computational challenge of learning both sets of parameters in the same sweep and maintains the interpretability of the joint model. However, the cost is that the posterior uncertainty about the state allocation is not propagated.

The activity regression component can be incorporated into the joint model via a DLM component, where the regression coefficients can vary over time, or added onto the observation equation with a static coefficient. McCarthy *et al.* (2017) show how those with higher CRF levels reduce their postprandial glucose levels by a smaller percentage overall. Over a timespan of around one week (the amount of data available for each patient),

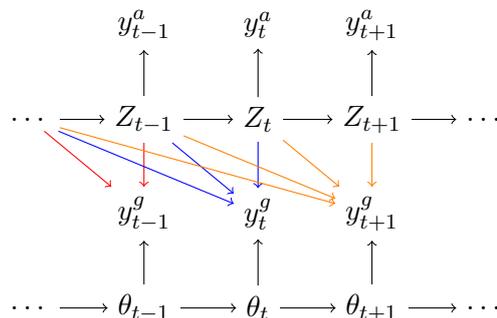


Figure 6.3: DAG showing the structure of a DLM for glucose levels, y_t^g , with hidden states θ_t , with a regression on the hidden activity states, Z_t , that describe the activity observations, y_t^a .

the fitness levels of the diabetic patients whose glucose levels are being monitored are considered constant, so the regression coefficients that quantify the impact of activity on glucose levels are time-invariant. If patients were being monitored over a longer period, a dynamic regression component might become necessary if the behavioural prompts issued from the model improve the overall fitness levels of the patients. For the models fitted here however, a fixed regression component is used.

Intuitively it makes sense that activity does not have an immediate effect on glucose levels. Indeed, this is evident from the plots in Fig. 6.1. It is therefore sensible to regress on the previous j activity levels in the model to predict glucose. Each lagged activity level is assigned a weight according to the impact it has on the current glucose level, as there is evidence in the lagged correlation plots, Fig. 6.1, to suggest the effect is not constant across lags. The model dependency structure is shown in the DAG in Fig. 6.3.

6.3.1 Local Fourier weighted regression model

In light of the exploratory analysis in Fig. 6.1 and the belief that activity states one and two represent largely sedentary behaviours, glucose is regressed only on lagged activity that is classified as belonging to state three. The initial regression of glucose, $\{y_t^g\}$, on activity data, $\{y_t^a\}$, features a step function to assign weights to the lagged activity levels that are classified into zone three according to the posterior mode, multiplied by a transformation of the corresponding raw activity observations. The remaining variation is then described by the local Fourier DLM. An indicator variable specifies whether or not the hidden state of the activity observation at lag k is equal to zone three, $\mathbb{I}(Z_{t-k} = 3)$, then this is multiplied by the raw, log-transformed activity observation minus the posterior mean of the third activity zone, μ_3 , according to the Normal HMM, to aid in parameter interpretation. The regression coefficient of this term is equal to the weight for that lag, determined by the step function. The activity levels up to a given lag are regressed on to sum over the effects of all recent activity that influences glucose. The full weighted regression model is described

in Model 6.1. Here, n_s is the total number of steps in the model, with step j going from lag b_{j-1} to lag b_j , for $j = 1, \dots, n_s$ and where $b_0 = 0$. Using a common coefficient, β_j , for lags in $b_{j-1} + 1$ to b_j is a parsimonious approximation to a model that has a different coefficient for each lag.

Model 6.1

$$\begin{aligned}
 y_t^g &= F\theta_t + \sum_{j=1}^{n_s} \beta_j \sum_{i=b_{j-1}+1}^{b_j} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3) + v_t, \quad v_t \sim N(0, V), \\
 \theta_t | \theta_{t-1}, W &\sim N_{2q+1}(G\theta_{t-1}, W), \\
 F &= \begin{bmatrix} 1 & 1 & 0 & 1 & \dots & 0 \end{bmatrix}, \quad G = \text{blockdiag}(1, H_1, \dots, H_q), \\
 \text{where } H_j &= \begin{bmatrix} \cos \omega_j & \sin \omega_j \\ -\sin \omega_j & \cos \omega_j \end{bmatrix} \text{ and } \omega_j = \frac{2\pi j}{s} \text{ for } j = 1, \dots, q, \\
 \beta_j &\sim N(m, s^2), \\
 V = \psi_y^{-1}, \quad W &= \text{diag}(\psi_1^{-1}, \psi_2^{-1}, \psi_2^{-1}, \dots, \psi_{q+1}^{-1}), \\
 \psi_y &\sim \text{Ga}(a_1, b_1), \\
 \psi_i &\sim \text{Ga}(a_2, b_2), \text{ for } i = 1, \dots, q + 1.
 \end{aligned}$$

The full conditional distribution of each step coefficient, β_j , is derived as follows, where $\hat{y}_t = y_t^g - F\theta_t$:

$$\begin{aligned}
 \pi(\beta_j | \cdot) &\propto \pi(\beta_j) \prod_{t=1}^T p(\hat{y}_t | \cdot), \\
 &\propto \exp \left[-\frac{1}{2s^2} (\beta_j - m)^2 \right] \prod_{t=1}^T \exp \left[-\frac{\psi_y}{2} \left\{ \hat{y}_t - \sum_{j=1}^{n_s} \beta_j \sum_{i=b_{j-1}+1}^{b_j} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3) \right\}^2 \right], \\
 &\propto \exp \left[-\frac{1}{2} \left\{ \beta_j^2 \left(\frac{1}{s^2} + \psi_y \sum_{t=1}^T \left\{ \sum_{i=b_{j-1}+1}^{b_j} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3) \right\}^2 \right) \right. \right. \\
 &\quad \left. \left. - 2\beta_j \left(\frac{m}{s^2} + \psi_y \sum_{t=1}^T \left\{ \sum_{i=b_{j-1}+1}^{b_j} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3) \right. \right. \right. \right. \\
 &\quad \left. \left. \left. \left. \left(\hat{y}_t - \sum_{l \neq j} \beta_l \sum_{i=b_{l-1}+1}^{b_l} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3) \right) \right) \right) \right\} \right], \\
 &\equiv \exp \left[-\frac{1}{2} \{ \beta_j^2 v' - 2\beta_j m' \} \right],
 \end{aligned}$$

so

$$\beta_j | \cdot \sim N(v'^{-1}m', v'^{-1}), \quad (6.1)$$

where

$$v' = \frac{1}{s^2} + \psi_y \sum_{t=1}^T \left\{ \sum_{i=b_{j-1}+1}^{b_j} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3) \right\}^2$$

and

$$m' = \frac{m}{s^2} + \psi_y \sum_{t=1}^T \left\{ \sum_{i=b_{j-1}+1}^{b_j} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3) \right. \\ \left. \left(\hat{y}_t - \sum_{l \neq j} \beta_l \sum_{i=b_{l-1}+1}^{b_l} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3) \right) \right\}.$$

The full conditional distributions of the unknown parameters of the local Fourier DLM, $(\psi_y, \psi_1, \dots, \psi_{q+1})$ are as derived previously in Eq. 3.11 and 3.12, respectively. However, a substitution of $y_t^g - \sum_{j=1}^{n_s} \beta_j \sum_{i=b_{j-1}+1}^{b_j} (\log y_{t-i}^a - \mu_3) \mathbb{I}(Z_{t-i} = 3)$ for y_t in each equation is necessary. As the hidden activity zones are assumed to be known, the MCMC scheme only needs to run the forward filtering backward sampling algorithm to sample the hidden states of the DLM, $\underline{\theta}$ and draw samples from the relevant full conditional distributions of the model parameters. The full scheme is described in Algorithm 6.1.

Algorithm 6.1: Block Gibbs sampler for Model 6.1

1. Initialise $\psi_y^{(0)}, \psi_j^{(0)}$ for $j = 1, \dots, q+1$ and $\beta_k^{(0)}$ for $k = 1, \dots, n_s$;
2. **for** $i = 1, \dots, N$:
3. run the forward filtering backward sampling algorithm (Algorithm 3.6) to simulate $\underline{\theta}^{(i)}$;
4. draw $(\psi_y^{-1})^{(i)}$ from (3.11);
5. draw $(\psi_1^{-1})^{(i)}$ from (3.12);
6. draw $(\psi_j^{-1})^{(i)}$ from (5.8) for $j = 2, \dots, q+1$;
7. draw $\beta_k^{(i)}$ from (6.1) for $k = 1, \dots, n_s$;
8. **end**

Results

Model 6.1 is fitted to Patient 5's data, since there appears to be much less information about the impact of activity on glucose in the lagged correlation plots for the other example diabetic patients, Fig. 6.1. Algorithm 6.1 was run for $N = 200,000$ iterations, after a burn in of 1,000 iterations and was thinned to 10,000 draws. The hyperparameters used were $a_1 = b_1 = 1$, $a_2 = 1.1$, $b_2 = 0.01$, $m = -2$ and $s = 2$. The DLM components have the

same prior hyperparameters as in the univariate models and the regression coefficients have a common prior centred around negative values, as expected, but with a fairly large variance. The Fourier component has $q = 4$ harmonics and the data have a period of $s = 288$. The initial distribution on the states has mean $m_0 = (8, \dots, 8)$ and variance $C_0 = \text{diag}(1, \dots, 1)$.

Informed by the exploratory analysis in Fig. 6.1, the activity data up to a lag of two hours are used to model each glucose level. The step sizes are also chosen based on Patient 5's zone three lagged correlation profile, with boundaries at lags 30 minutes, 45 minutes, one hour, one hour and 20 minutes, one hour and 35 minutes and two hours. The posterior means of the β_j result in the step function in Fig. 6.4. The figure shows the activity performed between one and 1.5 hours prior has the strongest effect on glucose levels, with a large negative weight assigned to this step. This dip coincides with the dip seen in the correlation profile for Patient 5. The step function features a positive weight for the first step before the weight becomes negative, which is similar to what is seen in ND 2's fasting correlation profile and what is expected from the relationship between glucose and activity of higher intensity. The posterior standard deviations for each of the β_j are 0.036, 0.042, 0.038, 0.047, 0.046 and 0.036, respectively. These values show there is a relatively large posterior variance, but for the main effects on glucose, lags with coefficients β_4 and β_5 (i.e. for the period one hour to one hour and 35 minutes), this does not affect the sign of the coefficient.

The posterior densities for the DLM parameters are shown in Fig. 6.5, alongside the corresponding densities for the local Fourier DLM fitted to the glucose levels of Patient 5 without the regression. Most notable is the slight increase in variance of the local level, ψ_1 , and the decrease in variance of the fourth harmonic, ψ_5 . The fourth harmonic picks up the residual seasonality not described by the first three harmonics, so a decrease in the variance of the states corresponding to this harmonic supports the activity regression component describing some of the seasonal variation in glucose levels, likely because high activity is performed at a similar time each day.

Fig. 6.6 shows the mean and 95% credible intervals of an ensemble of out-of-sample forecasts cast using the MCMC output, in comparison to the same forecasts cast using the univariate local Fourier DLM. With the exception of the first and third forecasting indices, there is a clear improvement in how closely the forecasts follow the observed glucose levels, particularly for forecast indices 5, 6, 8 and 10. This confirms that a bivariate model between glucose and activity can lead to improved forecasts for future glucose levels. It is noted, however, that the credible intervals are slightly wider for the bivariate model forecasts.

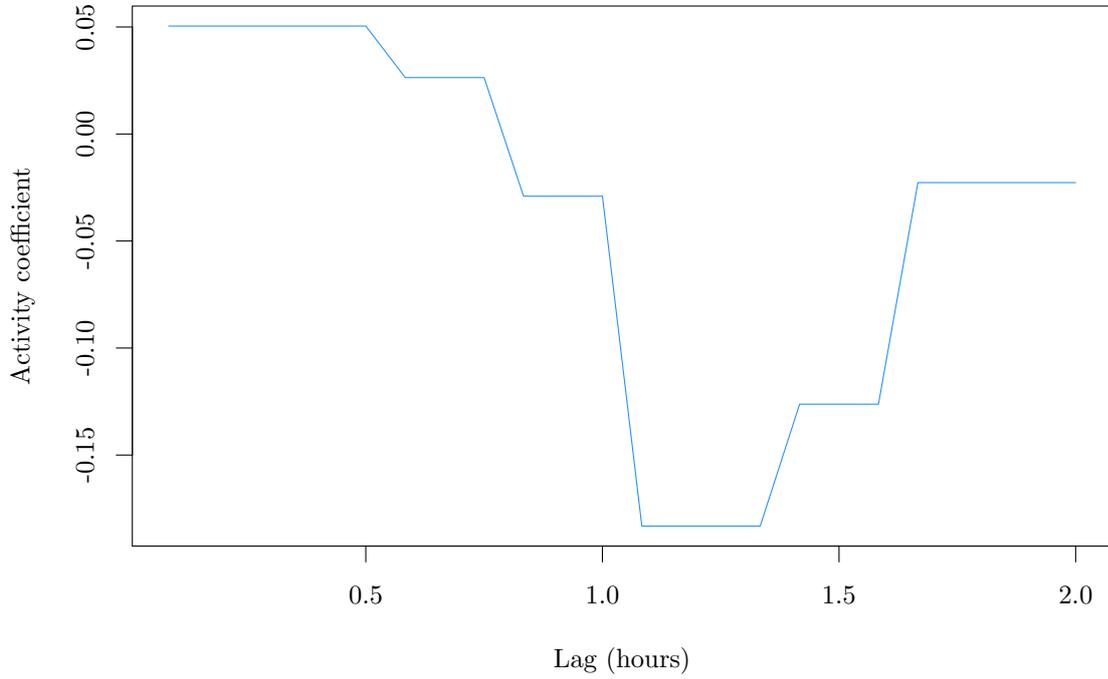


Figure 6.4: Step function of Model 6.1 for Patient 5 according to the posterior means of each β_j .

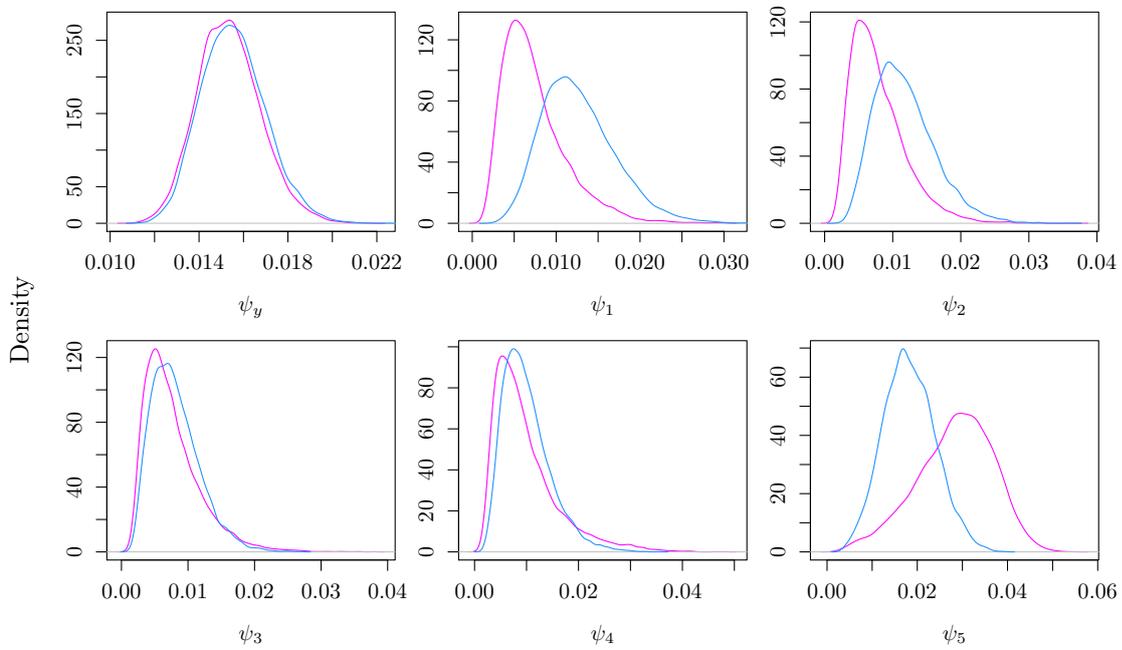


Figure 6.5: Densities of V and the diagonal elements of W for Model 6.1 (blue) compared to Model 5.4 (magenta).

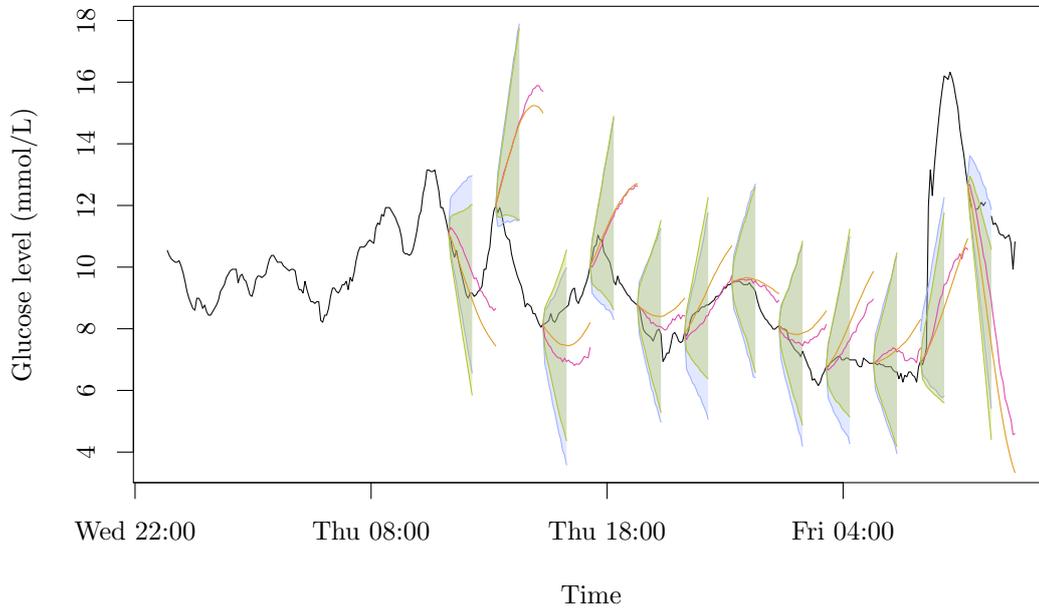


Figure 6.6: Two-hour-ahead forecasts of Patient 5's glucose levels after fitting Model 6.1 (magenta) compared to Model 5.4 (orange), with 95% credible intervals (purple and green, respectively).

6.3.2 A Gamma curve regression

The step function in Model 6.1 that was learned for Patient 5 looks as though it could be modelled parametrically by a scaled Gamma density function, i.e. $\gamma f(k; \alpha, \beta)$ where f is the Gamma density function, evaluated at lag k , given shape and rate parameters α and β and γ is a scale factor. If successful, this would offer a more parsimonious solution to modelling the relationship between glucose and activity levels. The model can be simplified further by removing dependence on the actual mean-centred activity values in state three; the model fitted here does just this. Note that the additional scale factor, adds a degree of freedom in describing the stretch in the y -direction of the curve. This allows for curves where the area under the curve does not integrate to one, as well as allowing the sign of the weights the curve describes to be negative.

The many possible combinations of shape and rate parameter for the Gamma curve coupled with the scale factor parameter open up the possibility of identifiability issues. To overcome this, the Gamma curve is reparametrised in terms of its mean and coefficient of variation: for a random variable $X \sim Ga(\alpha, \beta)$, the expected value of X is written $m = E(X) = \alpha/\beta$ and the variance of X is $Var(X) = \alpha/\beta^2$. The coefficient of variation is defined as the ratio of the standard deviation to the mean: $c = \sqrt{Var(X)}/E(X) = 1/\sqrt{\alpha}$. Rearranging c for α and substituting this into the equation for m to get β , the resulting

reparametrisation is: $\alpha = 1/c^2$ and $\beta = 1/(c^2m)$. Some prior knowledge of where the mean is expected to be, based on the information in the correlation profiles for other patients, coupled with the results from Model 6.1, can therefore be incorporated to prevent problems with identifying a suitable curve. The model is described in Model 6.2.

Model 6.2

$$\begin{aligned}
 y_t^g &= F\theta_t + \gamma \sum_{i=0}^k e_i \mathbb{I}(Z_{t-i} = 3) + v_t, \quad v_t \sim N(0, V), \\
 \text{where } e_i &= \frac{\beta^\alpha}{\Gamma(\alpha)} (i + 0.5)^{\alpha-1} e^{-\beta(i+0.5)}, \\
 \theta_t | \theta_{t-1}, W &\sim N_{2q+1}(G\theta_{t-1}, W), \\
 F &= \begin{bmatrix} 1 & 1 & 0 & 1 & \dots & 0 \end{bmatrix}, \quad G = \text{blockdiag}(1, H_1, \dots, H_q), \\
 \text{where } H_j &= \begin{bmatrix} \cos \omega_j & \sin \omega_j \\ -\sin \omega_j & \cos \omega_j \end{bmatrix} \quad \text{and } \omega_j = \frac{2\pi j}{s} \text{ for } j = 1, \dots, q, \\
 V &= \psi_y^{-1}, \quad W = \text{diag}(\psi_1^{-1}, \psi_2^{-1}, \psi_2^{-1}, \dots, \psi_{q+1}^{-1}), \\
 \psi_y &\sim \text{Ga}(a_1, b_1), \\
 \psi_i &\sim \text{Ga}(a_2, b_2), \text{ for } i = 1, \dots, q + 1, \\
 \gamma &\sim N(a_3, b_3^2), \\
 c &\sim \text{Ga}(a_4, b_4) \text{ and } m \sim \text{Ga}(a_5, b_5).
 \end{aligned}$$

The full conditional distribution for γ is the result of combining the Normal prior and likelihood, which results in the Normal target distribution:

$$\gamma | \cdot \sim N \left(\left(\frac{1}{b_3^2} + \psi_y \sum_{t=1}^T (\underline{e}' \mathbb{I})^2 \right)^{-1} \left(\frac{a_3}{b_3^2} + \psi_y \sum_{t=1}^T \hat{y}_t \underline{e}' \mathbb{I} \right), \left(\frac{1}{b_3^2} + \psi_y \sum_{t=1}^T (\underline{e}' \mathbb{I})^2 \right)^{-1} \right), \quad (6.2)$$

where $\underline{e}' \mathbb{I} = e_0 \mathbb{I}(Z_t = 3) + \dots + e_k \mathbb{I}(Z_{t-k} = 3)$ and $\hat{y}_t = y_t^g - F\theta_t$.

As α and β are functions of the mean and coefficient of variation, m and c do not have semi-conjugate full conditional distributions and so a MH step is necessary for each of these parameters. The full conditional distribution for m is:

$$\begin{aligned}
 \pi(m | \cdot) &\propto \pi(m) \prod_{t=1}^T p(\hat{y}_t | \cdot), \\
 &\propto m^{a_5-1} e^{-b_5 m} \prod_{t=1}^T \exp \left[-\frac{\psi_y}{2} \left(\hat{y}_t - \gamma \underline{e}' \mathbb{I} \right)^2 \right], \\
 &\propto m^{a_5-1} \exp \left[-b_5 m - \frac{\psi_y}{2} \sum_{t=1}^T \left(\hat{y}_t - \gamma \underline{e}' \mathbb{I} \right)^2 \right],
 \end{aligned}$$

and similarly for c :

$$\pi(c|\cdot) \propto c^{a_4-1} \exp \left[-b_4 c - \frac{\psi_y}{2} \sum_{t=1}^T \left(\hat{y}_t - \gamma \underline{e}' \mathbb{I} \right)^2 \right].$$

A log-Normal proposal distribution is used for both parameters, with median equal to the previous value of the parameter and standard deviation equal to a tuning parameter, λ . The Metropolis within Gibbs scheme is written in Algorithm 6.2. When drawing $(\psi_y, \psi_1, \dots, \psi_{q+1})$, the substitution of $y_t^g - \gamma \underline{e}' \mathbb{I}$ for y_t in Eq. (3.11) and Eq. (3.12) is made.

Algorithm 6.2: Block Metropolis-within-Gibbs sampler for Model 6.2

1. Initialise $\psi_y^{(0)}, \psi_j^{(0)}$ for $j = 1, \dots, q + 1$, $\gamma^{(0)}, m^{(0)}$ and $c^{(0)}$;
2. **for** $i = 1, \dots, N$:
3. run the forward filtering backward sampling algorithm (Algorithm 3.6) to simulate $\underline{\theta}^{(i)}$;
4. draw $(\psi_y^{-1})^{(i)}$ from (3.11);
5. draw $(\psi_1^{-1})^{(i)}$ from (3.12);
6. draw $(\psi_j^{-1})^{(i)}$ from (5.8) for $j = 2, \dots, q + 1$;
7. draw $\gamma^{(i)}$ from (6.2);
8. generate a proposal \tilde{m} from $q(m^{(i-1)}, \tilde{m})$;
9. evaluate the acceptance probability

$$A(m^{(i-1)}, \tilde{m}) = \min \left(1, \frac{\pi(\tilde{m}|\cdot)q(\tilde{m}, m^{(i-1)})}{\pi(m^{(i-1)}|\cdot)q(m^{(i-1)}, \tilde{m})} \right);$$
10. accept \tilde{m} with probability $A(m^{(i-1)}, \tilde{m})$: set $m^{(i)} = \tilde{m}$, otherwise set $m^{(i)} = m^{(i-1)}$;
11. generate a proposal \tilde{c} from $q(c^{(i-1)}, \tilde{c})$;
12. evaluate the acceptance probability

$$A(c^{(i-1)}, \tilde{c}) = \min \left(1, \frac{\pi(\tilde{c}|\cdot)q(\tilde{c}, c^{(i-1)})}{\pi(c^{(i-1)}|\cdot)q(c^{(i-1)}, \tilde{c})} \right);$$
13. accept \tilde{c} with probability $A(c^{(i-1)}, \tilde{c})$: set $c^{(i)} = \tilde{c}$, otherwise set $c^{(i)} = c^{(i-1)}$;
14. **end**

Results

Algorithm 6.2 was run for $N = 200,000$ iterations, plus a burn in of 1,000 iterations and thinned by 20. The DLM hyperparameters were as before and the weighted curve hyperparameters were: $a_3 = -5$, $b_3 = 2$, $a_4 = 1$, $b_4 = 1$, $a_5 = 10$, $b_5 = 1$. This is

consistent with negative weights, with the peak of the effect between glucose and activity occurring around lag 10. A lag of up to two hours was considered for the activity data. The prior on c has a large variance, as the data should provide enough information about this parameter given the stronger prior on m .

The MCMC output for the regression parameters produce a Gamma curve with a mean m whose posterior mean is centred about 16, which corresponds to a peak effect of activity on glucose levels around one hour and 20 minutes after the higher level of activity is performed. The coefficient of variation is around 0.15, which corresponds to a fairly small dispersion about the mean. The scale factor, centred about -1.25, confirms the negative relationship between activity and glucose. The resulting posterior mean curve, $\gamma f(k; \alpha, \beta)$ for each $k = 0.5, \dots, 23.5$, showing the relationship between glucose levels and the lagged third activity zone according to the posterior draws from m , c and γ is shown in Fig. 6.7, in comparison to the step function learned in the previous section. The curve produces a similar shape to the step function, which shows that a more parsimonious solution to separate regression coefficients is available and suggests that including an indicator variable without the raw activity observations contains enough information to learn the relationship between glucose and activity for this patient. The posterior standard deviations for γ , m and c are 0.19, 0.30 and 0.02; because these parameters are used to produce the scaled Gamma curve, the posterior uncertainty is greatest at the peak of the curve, as opposed to a more uniform uncertainty for the step function at all lags. A 95% posterior credibility interval estimates the peak between -0.15 and -0.25, which is more concentrated than the step function, which estimates the peak between -0.08 and -0.26. The posterior densities for the DLM parameters are similar to those of Model 6.1, shown in Fig. 6.5.

The forecasts from Model 6.2 are shown in Fig. 6.8, once again alongside the forecasts from Model 5.4. The forecasts are calculated using the same method of taking an ensemble of forecasts using draws from the MCMC chain and computing the mean and 95% credible intervals from these. The mean forecasts show a further improvement in fitting the scaled Gamma curve DLM over the step function DLM and the local Fourier DLM alone. The forecasts track much more closely to the observed glucose levels, for example at forecast indices 5 and 10, where there are only small deviations. Overall, the credible intervals on the forecasts are also smaller from the scaled Gamma curve model.

The number of standard deviations that the mean forecasts for each of the bivariate models fall from the observed glucose levels are shown in Fig. 6.9. The plot shows how both models perform very similarly.

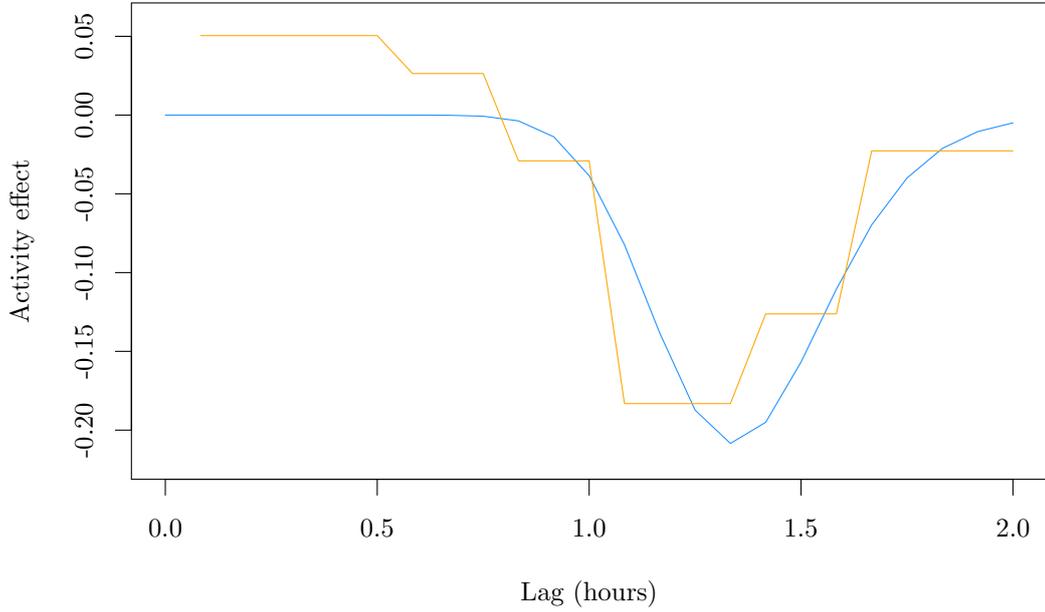


Figure 6.7: Lagged effect of activity in zone three on glucose levels for Patient 5, according to posterior means.

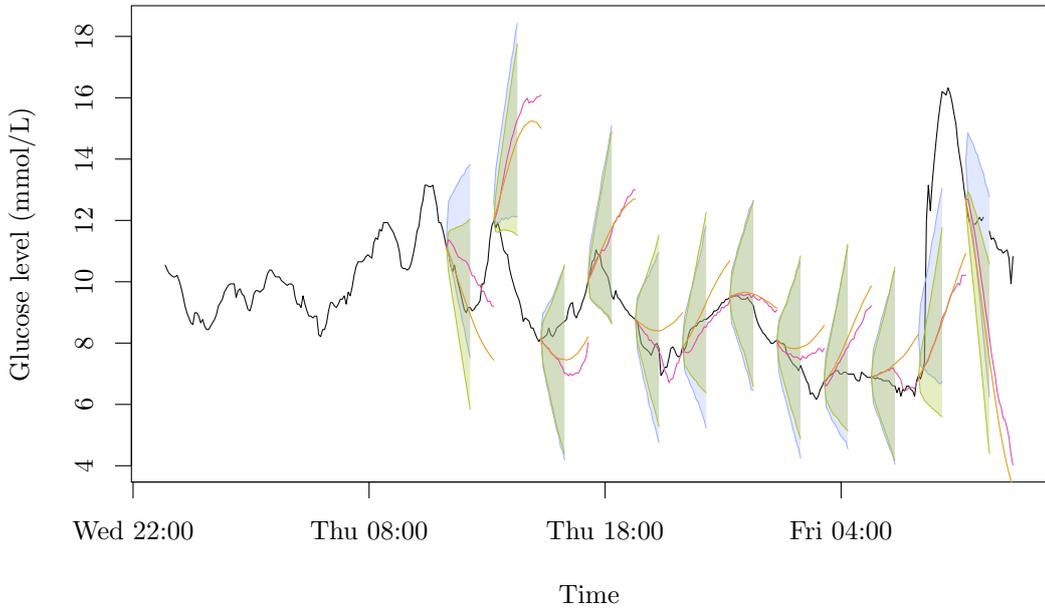


Figure 6.8: Two-hour-ahead forecasts of Patient 5's glucose levels after fitting Model 6.2 (magenta) compared to Model 5.4 (orange), with 95% credible intervals (purple and green, respectively).

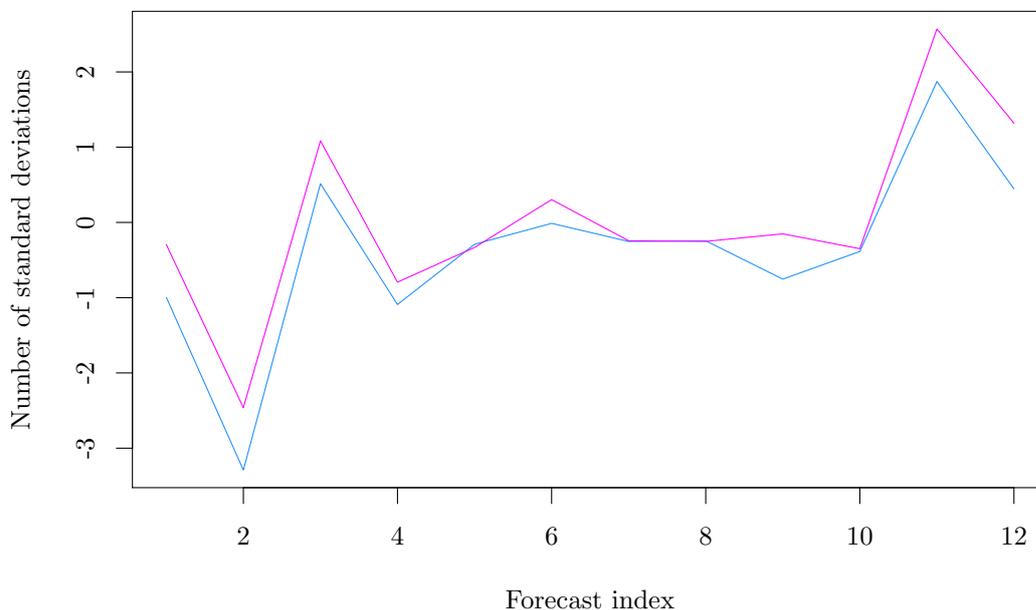


Figure 6.9: Mean number of standard deviations of each mean forecast from the observed series for each two hour forecast, for the step function DLM (magenta) and the scaled Gamma curve DLM (blue).

6.3.3 Across-patient and model comparison

Model 6.2 is fitted to the other sample patients (Patient 3, Patient 10, ND 1, ND 2 and ND 3) to assess how well it fits to and performs on glucose and activity data with potentially different lagged relationships.

Different priors were explored for this model with each patient, with those used summarised in Table 6.1. Whilst a $Ga(1, 1)$ prior was sufficient on c for Patient 5, the MCMC chain stuck at values near zero for most other patients and so a prior bounded away from zero was necessary to prevent this from happening. When the coefficient of variation is zero, the Gamma density is not properly defined and so the other parameters, the mean, m , and scale factor, γ , cannot be learned. A smaller prior variance was also used for this parameter, favouring values of c between zero and one. This was reasonable as the variance of the posterior weighted curve is expected to be quite small. The prior on γ reflects the negative relationship expected between glucose and activity and the prior on m for each patient is based on the lagged correlation profiles. The hyperparameters for the DLM parameters are the same as for Patient 5. Algorithm 6.2 was run for a burn in of 1,000 iterations per patient, followed by the number of iterations, N , shown in the table. The output was thinned to 10,000 iterations. The different numbers of iterations were chosen to achieve a similar posterior effective sample size per patient, which took longer

Table 6.1: Model 6.2 MCMC information for Patient 3, Patient 10, ND 1, ND 2 and ND 3.

Patient	Iterations	Hyperparameters	Tuning parameters	Acceptance rates
3	200,000	$a_3 = -5, b_3 = 2,$ $a_4 = 5, b_4 = 10,$ $a_5 = 10, b_5 = 1$	$\lambda_m = 0.02, \lambda_c = 0.04$	0.63, 0.58
10	200,000	$a_3 = -2, b_3 = 2,$ $a_4 = 2, b_4 = 2,$ $a_5 = 5, b_5 = 1$	$\lambda_m = 0.2, \lambda_c = 0.5$	0.71, 0.61
ND 1	100,000	$a_3 = -5, b_3 = 2,$ $a_4 = 5, b_4 = 5,$ $a_5 = 4, b_5 = 1$	$\lambda_m = 0.3, \lambda_c = 0.6$	0.34, 0.40
ND 2	200,000	$a_3 = -5, b_3 = 2,$ $a_4 = 2, b_4 = 2,$ $a_5 = 12, b_5 = 2$	$\lambda_m = 0.01, \lambda_c = 0.04$	0.96, 0.84
ND 3	100,000	$a_3 = -2, b_3 = 2,$ $a_4 = 5, b_4 = 5,$ $a_5 = 4, b_5 = 1$	$\lambda_m = 0.3, \lambda_c = 0.6$	0.78, 0.54

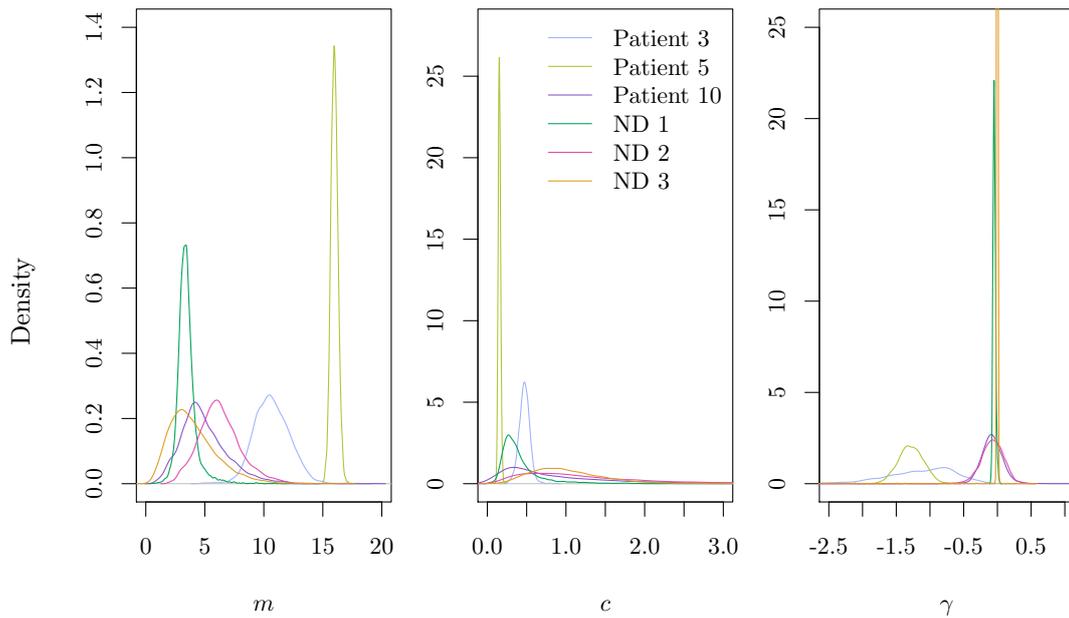


Figure 6.10: Posterior densities for m , c and γ in Model 6.2 for Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3.

for some patients. The tuning parameters were chosen for acceptance rates between 0.1 and 0.8, which was successful for all but ND 2 where the rates are higher.

The posterior densities for the weighted curve parameters for each patient are plotted in Fig. 6.10. Only Patient 3 and Patient 5 have scale parameters that are not consistent

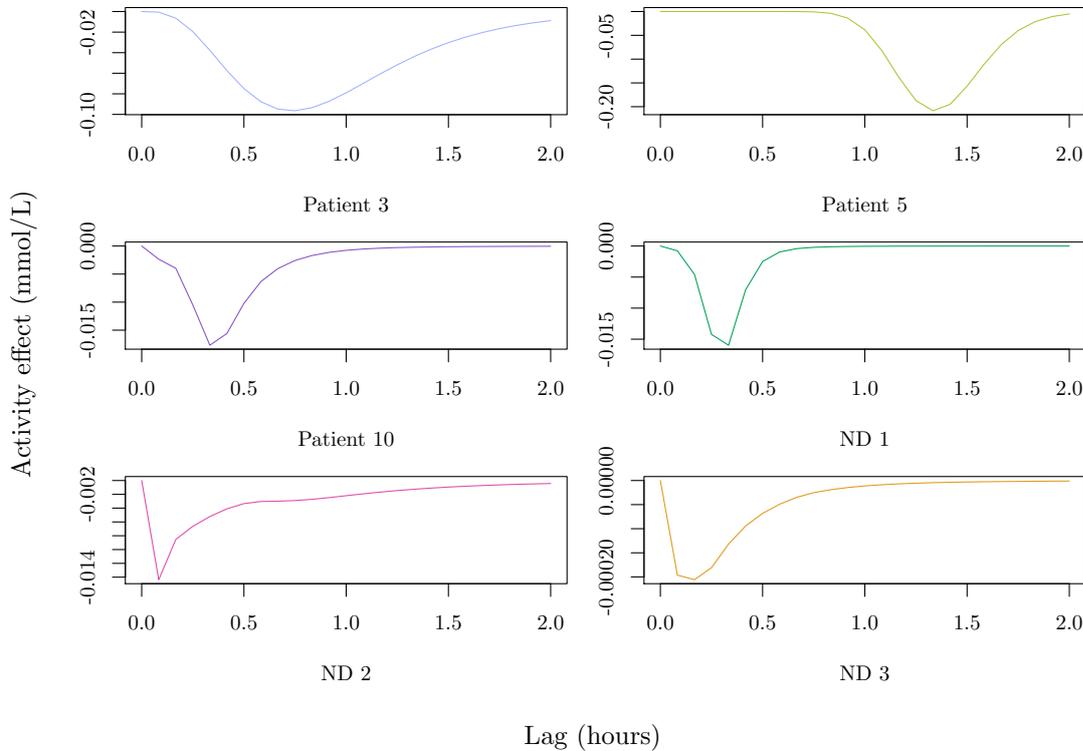


Figure 6.11: The mean weighted curve in Model 6.2 for Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3, according to posterior draws of m , c and γ .

with zero, implying that only a small effect from activity is detected in the other patients' data. The posterior densities for m and c for Patient 10, ND 2 and ND 3 mainly follow the shape of their priors, which also implies there is a lack of information in these data sets. The weighted curves produced by taking the mean of posterior samples from m , c and γ are shown in Fig. 6.11. The shapes of the curves mimic what is seen in the correlation plots for the most parts, but the very small values on the y -axes for all but Patient 3 and Patient 5 confirm that these two patients are the only ones where a notable activity effect is observed. The y -axis is particularly small for ND 3. The posterior variance in the curves for each Patient falls about the peak, as with Patient 5; the peaks of Patient 3, 10 and ND 2 have a 95% credible interval of ± 0.06 , whilst the peak of ND 1 has a 95% credible interval of ± 0.015 and ND 3 of ± 0.04 , but there is less information in these patients' data.

Despite the small weight associated with some of the activity lags, cumulatively these can still have an impact on glucose levels. Fig. 6.12 shows the cumulative effect of activity on glucose compared to the raw activity observations for the last day and a half of activity data, shown as the mean cumulative effect from 2,000 posterior draws. For higher activity observations performed by Patient 3 and Patient 5, the cumulative effect of activity reduces

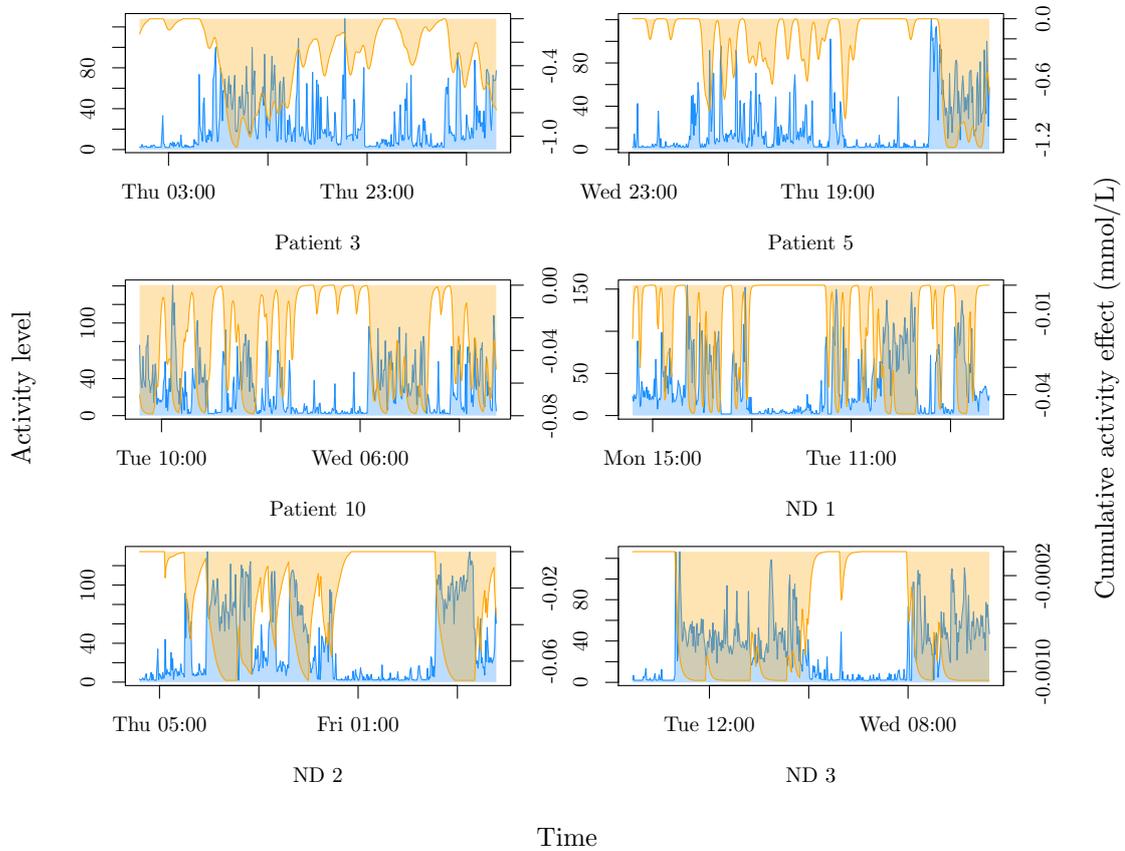


Figure 6.12: The mean cumulative effect of activity (orange) on glucose levels compared to the raw activity data (blue) when fitting Model 6.2 to Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3's data. The left-hand-side y-axis of each plot is the raw activity of each patient and the right-hand-side is the cumulative activity effect for each patient.

glucose levels by over 1 mmol/L, whereas the other patients see only a small cumulative effect. The raw activity data plotted for each patient show that even though some of the activity levels performed by the latter set of patients are much higher than the former two, this does not mean that the relationship between glucose and activity becomes more prominent. One reason that the model picks up less of or struggles to detect a clear relationship between glucose and activity for some patients and not others could be that some subjects perform activity at similar times each day, so this seasonal behaviour may be described by the Fourier harmonics instead. This would explain why the seemingly clear lagged correlation profile for ND 2 is not detected.

The improved model forecasts for each patient are shown in Fig. 6.13. The largest improvements are in the first two patients, as expected, though as there is only a small amount of variation in the glucose profile of ND 2, the small correction in forecasts from including the activity regression has made these forecasts track even closer to the observed

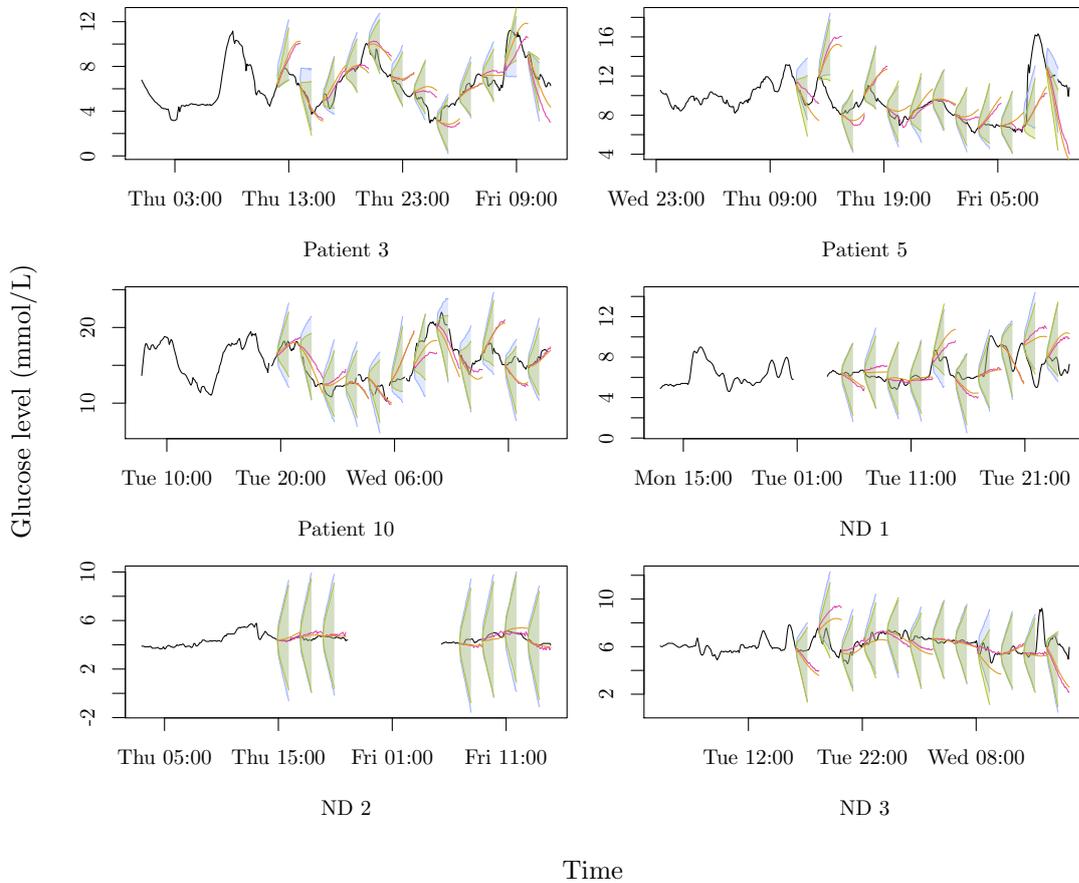


Figure 6.13: Two-hour-ahead forecasts of Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3's glucose levels after fitting Model 6.1 (magenta) compared to Model 5.4 (orange), with 95% credible intervals (purple and green, respectively).

values than before.

For comparison, the step function DLM was also fitted to each patient. The resulting step functions according to the posterior means of the step coefficients are shown in Fig. 6.14. The plots show a range of both positive and negative coefficients for all of the patients, which could explain why the weighted curves for the scaled Gamma curve DLM are so close to zero for most patients. The scales on the y -axes of the functions are still small, though these values are additionally scaled by the transformed activity observations. Despite the lagged activity effects looking quite different for the two models for each patient, particularly for Patient 10 and ND 2, the forecasts are similar. This shows that including the raw activity observations in Model 6.2 would be unlikely to further improve these models. For each patients' set of step function coefficients, the standard deviations for Patient 3 and ND 2's coefficients are similar to those of Patient 5, however the posterior variance for Patient 10 is larger, with standard deviations between 0.06 and

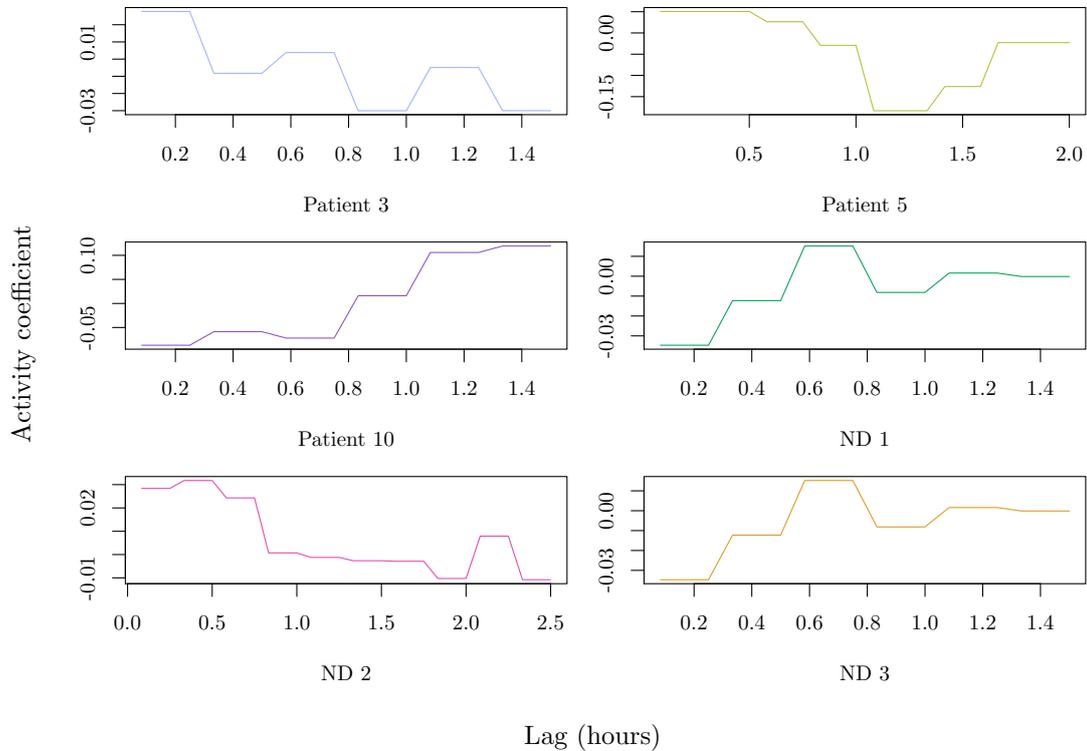


Figure 6.14: The step function coefficients in Model 6.1 for Patient 3, Patient 5, Patient 10, ND 1, ND 2 and ND 3 according to the posterior means of each β .

0.09, and for ND 1 and ND 3 are again smaller, with standard deviations around 0.015. This is similar to what was seen in the scaled Gamma curve, where the posterior variance for patients with less information in their data was smaller.

6.4 Modelling eating effects

Whilst the bivariate models between glucose levels and activity data have improved forecasts of glucose levels for some patients, there is still room for improvement. The variable with the largest effect on all patient glucose levels is food intake. Knowledge of patient eating times and details on what they have eaten could therefore also aid prediction of future glucose levels, and provide improvements to forecasts where the activity and glucose models have not.

Data are available for ND 1 and ND 3 on the number of grams of carbohydrates consumed throughout the data collection period. A breakdown of the number of grams of other macronutrients consumed by ND 1 is also available, but this was not recorded by ND 3. As carbohydrates are broken down into glucose, their consumption directly affects blood glucose levels, so monitoring this variable against glucose levels presents the

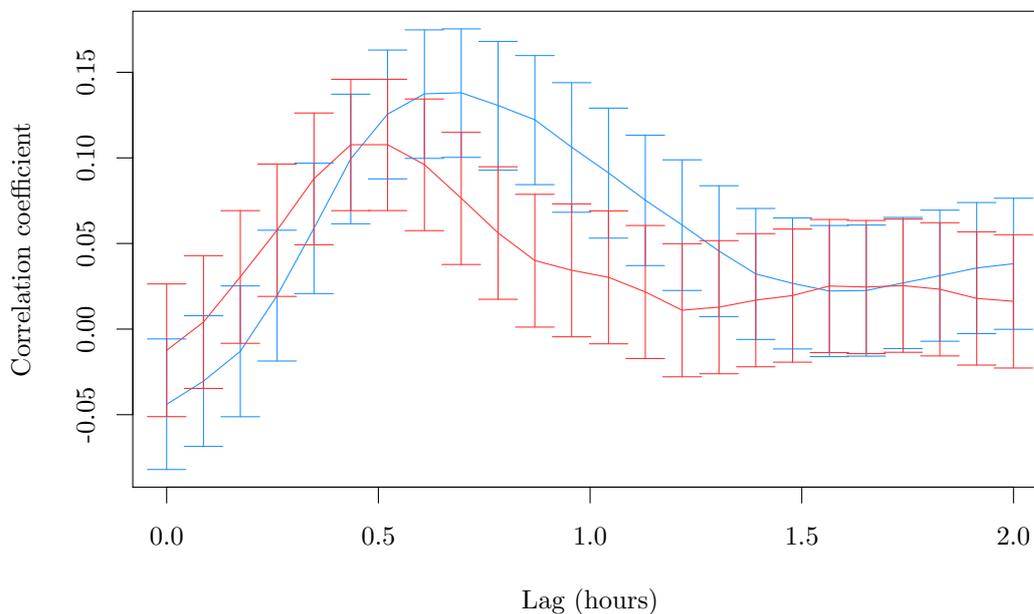


Figure 6.15: Lagged mean correlation between glucose levels and number of grams of carbohydrates eaten by ND 1 (blue) and ND 3 (red), with 95% confidence intervals.

opportunity to carry out some initial modelling of eating effects on patient glucose levels.

6.4.1 The relationship between glucose and carbohydrates

Some exploratory analysis into the relationship between food intake, specifically the number of grams of carbohydrates, and glucose levels is necessary to determine the kind of model that might be suitable between them. The lagged correlation between the two variables for both patients is plotted in Fig. 6.15.

The correlation profiles for both patients have a clear, similar shape, with the strongest correlation between number of carbohydrates and higher glucose levels occurring around 30 minutes after consumption. A weighted regression model therefore seems like a sensible starting point for these variables. What is more, the correlation profiles approximately resemble Gamma densities and a similar bivariate glucose model incorporating this shape has already been developed (Model 6.2).

6.4.2 Local Fourier eating effects model

To model the effect of food intake on glucose levels, a curve described by a Gamma density with an additional scale parameter, γ , is suggested. This model is the same as Model 6.2, however now the effect at lag k , e_k , is the impact of the number of grams of carbohydrates

consumed instead of the impact of activity from activity zone three. The indicator variable, $\mathbb{I}(Z_{t-k} = 3)$, is replaced by the number of grams of carbohydrates consumed at lag k , C_k , which is zero when no food is consumed. The observation equation becomes:

$$y_t^g = F\theta_t + \gamma \sum_{i=0}^k e_i C_i + v_t, \quad v_t \sim N(0, V). \quad (6.3)$$

The full conditional distributions for all of the model parameters are the same as for Model 6.2, with the relevant substitution of C_k for $\mathbb{I}(Z_{t-k} = 3)$. The Metropolis within Gibbs scheme is as described in Algorithm 6.2.

Results

Results are presented for both ND 1 and ND 3, since there are only two patients' data available. The model information for both patients is the same: Algorithm 6.2 was run for $N = 200,000$ iterations, plus a burn in of 1,000 and a thin of 20. The model hyperparameters were $a_1 = b_1 = 1$, $a_2 = 1.1$, $b_2 = 0.01$, $a_3 = 2$, $b_3 = 2$, $a_4 = 10$, $b_4 = 1$, $a_5 = .5$ and $b_5 = 1$. These choices are based on the prior information provided in the lagged correlation plots, Fig. 6.15, which imply the mean of the curve should be between six and 12 (30 minutes to one hour). The local Fourier DLM priors are the same as in previous models. The initial state distribution parameters were $m_0 = (8, \dots, 8)$ and $C_0 = \text{diag}(1, \dots, 1)$. The maximum lag considered is chosen based on how long the effect on glucose levels of a single eating event persists. Based on the correlation plots, this was chosen to be two hours.

The resulting posterior mean curves describing the relationship between glucose levels and the number of grams of carbohydrates consumed are shown in Fig. 6.16 for both patients. As is seen in Fig. 6.15, the peak effect from eating for ND 3 occurs slightly sooner (around 30 minutes) than for ND 1, which occurs around 45 minutes after the event. Both models have a similar coefficient of variation, with mean 0.25 and 0.26 for ND 1 and ND 3, respectively. Most notable about the results for the patients is how much larger the peak eating effect is for ND 3 than it is for ND 1, though the scale of both curves is small (up to around 0.005 for ND 1 and 0.015 for ND 3). The posterior mean of γ for ND 3 is approximately 1.75 times the size of the posterior mean of γ for ND 1 (0.061 and 0.036, respectively). The posterior densities for ψ_y and ψ_1, \dots, ψ_5 for both patients are similar to those in the univariate DLM for these patients.

The small scale of the posterior lagged eating effects does not necessarily mean that the effect of carbohydrates on glucose levels is small, as the weights are multiplied by the number of grams of carbohydrates consumed, which can be large. The data recorded by ND 1 contain values over 200 g and by ND 3 just over 100 g. The wider range of

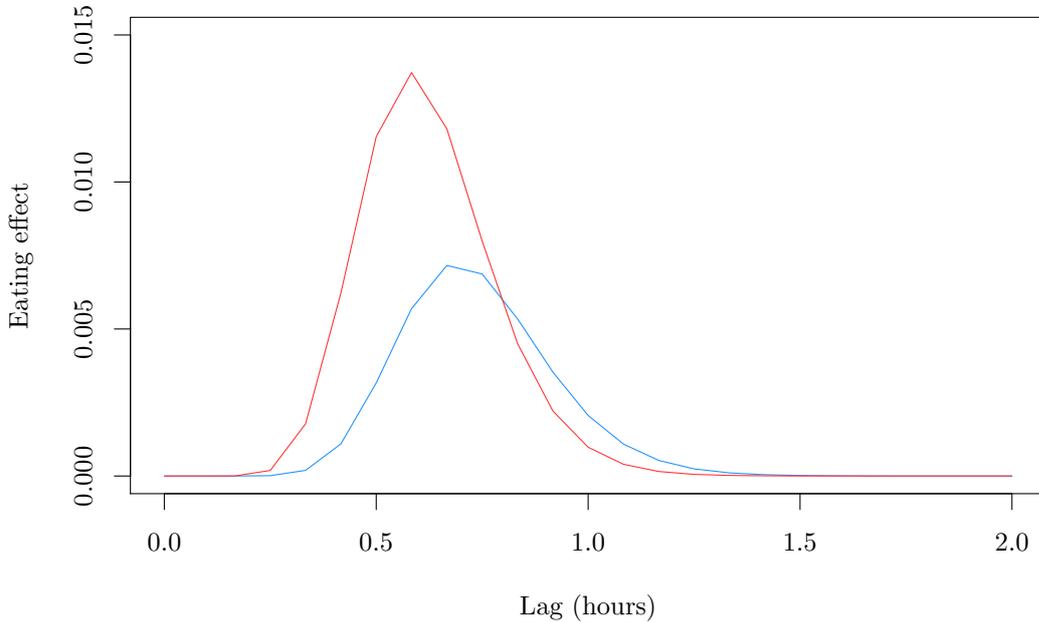


Figure 6.16: Lagged effect of the number of grams of carbohydrates on glucose levels for ND 1 (blue) and ND 3 (red) according to the posterior means of Model 6.2 with observation Eq. (6.3).

values recorded by ND 1 also explains why the curve for this patient has a smaller peak than ND 3. The cumulative effect of eating carbohydrates over time according to the corresponding weights is shown in Fig. 6.17 for both patients. The effect is computed for a sample of 2,000 posterior draws from the chain of each model, then the mean of the cumulative effect is plotted. The figure shows that even though the scale of the weights is small, relative to the scale of the glucose observations there is a notable cumulative effect from eating. This effect is, however, smaller than expected, especially in comparison to the size of the major peaks in the glucose profiles. Given the posterior standard deviations for the Gamma curve parameters for both patients, the peaks of these curves vary by ± 0.002 and ± 0.003 for ND 1 and ND 3, respectively.

The mean two-hour-ahead out-of-sample forecasts made from an ensemble of forecasts using sample draws from the MCMC chain of Model 6.2 with observation Eq. (6.3) for ND 1 and ND 3 are shown in Fig. 6.18. There is an improvement in the fit of the forecasts shortly after carbohydrates are consumed, for example at forecast index 7 for ND 1 and index 11 for ND 3, but the model forecasts have not captured the full rise in glucose levels that appears to be associated with those meals. This suggests that there are aspects to the glucose profile that are still not sufficiently being captured by Model 6.2 with observation Eq. 6.3 or by Model 6.2 when activity information is being used.

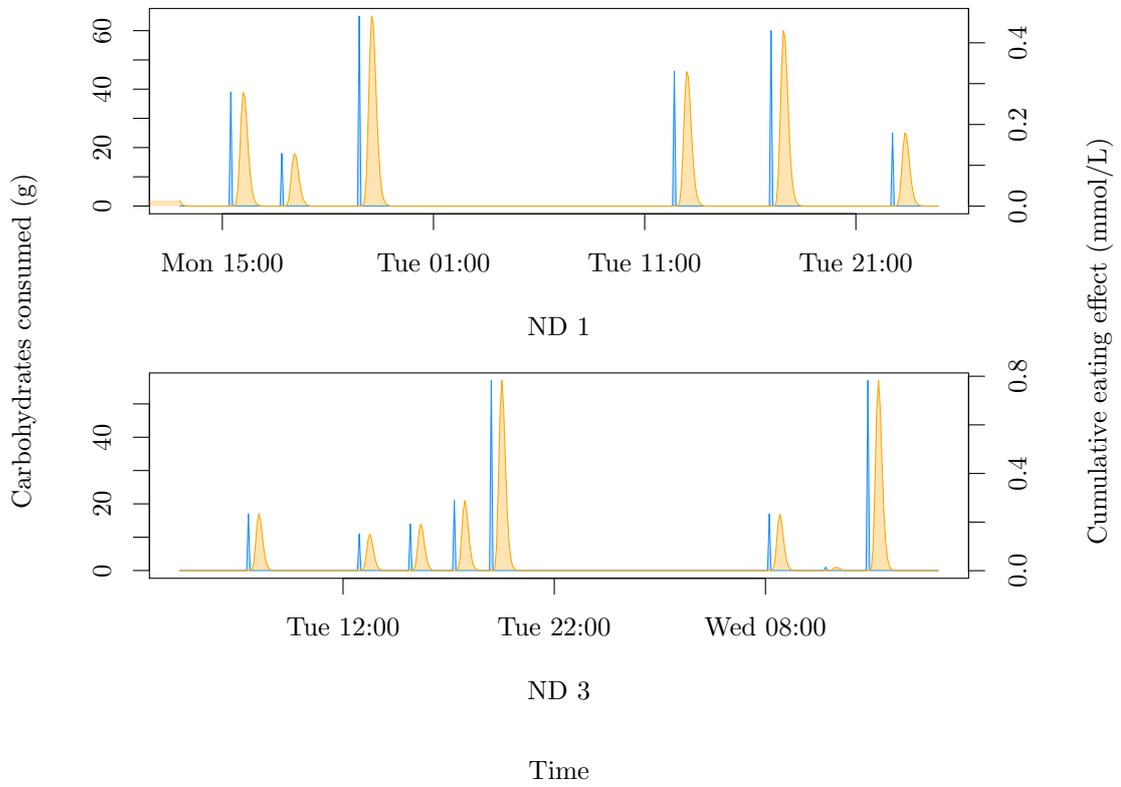


Figure 6.17: Cumulative effect of eating carbohydrates over time (orange) according to Model 6.2 with observation Eq. (6.3) on ND 1 and ND 3's glucose levels, compared to the number of grams of carbohydrates consumed (blue).

6.5 Summary

This chapter demonstrates the benefits of fitting bivariate models between glucose and activity using the technique of Markov melding.

Incorporating the posterior modal activity observation classifications from fitting a Normal HMM to the log-transformed BFEN data was first explored. The lagged correlations between the observations in each activity intensity zone and glucose levels were plotted, Fig. 6.1. Overall, there was little impact from the low- and medium- intensity activity zones on glucose, but the figure showed a varied relationship between the glucose and the high activity component from patient to patient. Patient 5's data were then used to develop the bivariate models, since they showed a clear relationship between the two variables and also followed the expected pattern given information from experts in metabolism.

A step function model was first explored, to capture the positive and negative lagged correlations between glucose and activity zone three. This curve also allowed for capturing the different relationships between glucose and activity between patients. The resulting

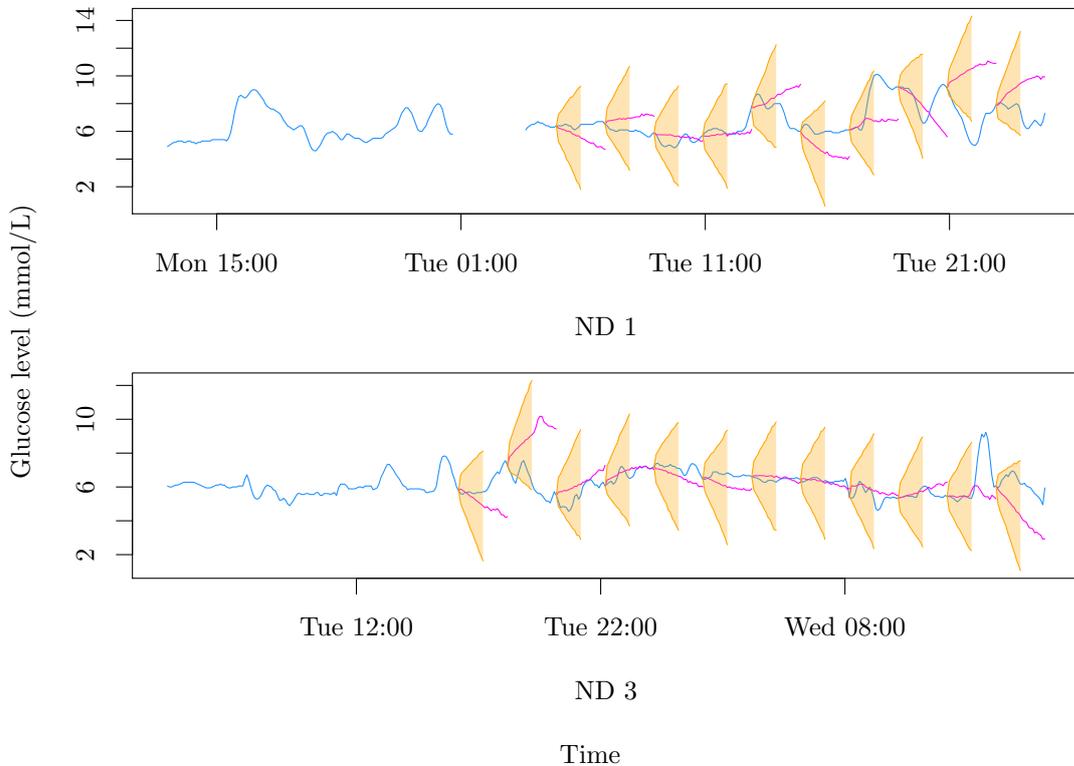


Figure 6.18: Two-hour-ahead forecasts (magenta) on ND 1 and ND 3's glucose levels after fitting Model 6.2 with observation Eq. (6.3), with 95% credible intervals (orange).

forecasts from this model showed clear improvements when compared to forecasts from the univariate local Fourier model fitted to glucose levels in Chapter 5. The step function was then also approximated by a scaled Gamma density, parametrised by the mean and coefficient of variation of the Gamma density to overcome identifiability problems. This model showed further improvements to glucose forecasts.

Some exploration into the relationship between glucose and food intake was then also performed on the additional non-diabetic patient data, where the number of grams of carbohydrates had been collected for two of the additional patients. A scaled Gamma curve seemed suitable for describing the relationship between these two variables, and fitting this joint model in a similar manner to the activity and glucose model showed small improvements in forecasting the glucose levels, compared to the univariate model. These results did not provide as large an improvement as expected, which suggests that the effect of carbohydrate intake on glucose was not fully being detected by the model.

Overall, the models fitted in this chapter showed how additional information can improve forecasts of the primary variable glucose. These improvements were seen across patients, where a relationship was clearly detected in the lagged correlation plots first.

For the patients whose forecasts did not improve significantly, it is suspected that their data were not rich enough to detect the impact of activity on their glucose levels.

Chapter 7

Monitoring the production of liquid natural gas

7.1 Background

This chapter demonstrates a second application area involving a similar model where incorporating exogenous information leads to improved forecasts. This work was carried out as part of a work placement at Woodside Energy Ltd, Australia's largest oil and gas company, at the end of the second year of research, which provided the unique opportunity to apply similar methods to those used in this thesis to an industry-based problem. Pluto gas plant (PGP) in Karratha, Western Australia produces tens of thousands of tonnes of LNG every day. Conditions at the plant are carefully controlled to maximise the amount of LNG produced. As LNG is produced, it fills a tank on site. Ships are scheduled to come and drain the tank when it is due to hit 'tank tops' (a full tank) so production does not need to slow down or stop, but there needs to be the right number of ships arriving according to the rate of production to optimise emptying the tank. It is therefore suggested that a forecast of short-term production levels (a few days in advance) would be beneficial to those scheduling the ships, in order to better predict when tank tops would be met.

One of the main factors impacting production levels at PGP is the weather. The weather in Karratha is hot all year round, but when the temperature reaches over 37°C the plant needs to run in a different way to compensate for the additional cooling needed (ideal temperatures for high production are lower). This is known to those controlling the conditions in the plant, so the approximate range of production levels expected given the current weather forecast is known. As well as high temperatures, large gusts of wind are often unpredictable and have a negative impact on production levels. The idea is therefore to fit a model to past production levels at PGP whilst taking into account past and future

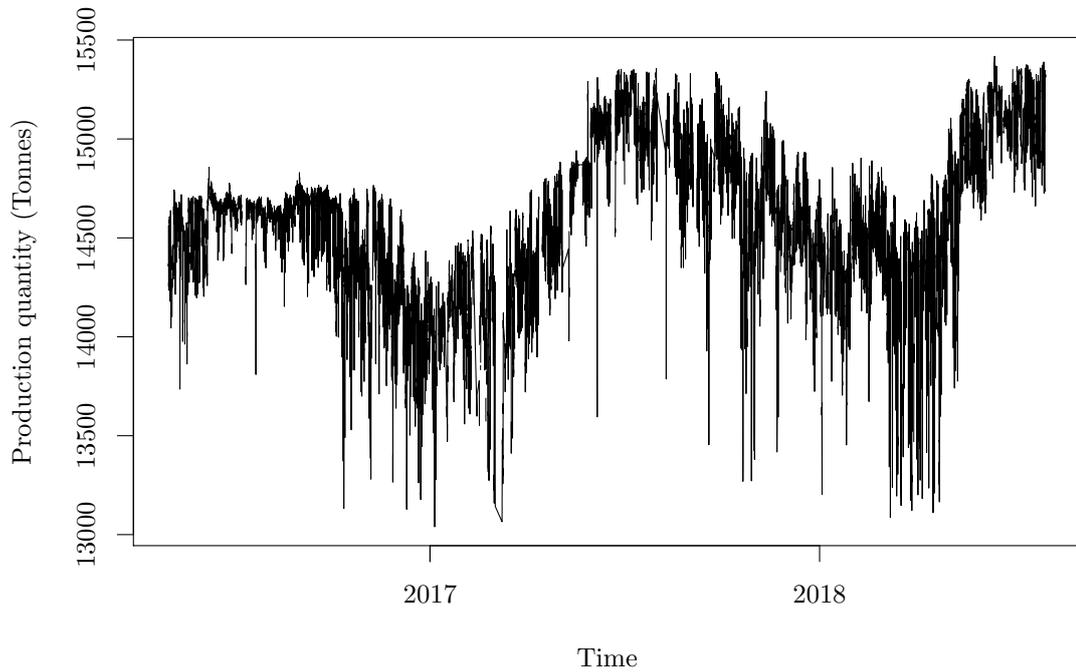


Figure 7.1: Production levels of LNG over time (thinned by five for plotting).

weather conditions to give more informed forecasts.

7.2 Exploratory data analysis

The conditions at PGP are monitored by hundreds of variables known as tags. Some of these tags are manually set in response to other conditions in the plant (known as control tags), and some are observed, for example temperature tags. All historical data recorded from each tag are made available in real-time and in different summaries, such as the mean value recorded over 10 minute intervals. There is also additional information available, such as when the plant is running in a steady state or not.

Data from 2016 to 2018 are explored for modelling. Data prior to this are not as useful as the way the plant is run was changed in 2016. Observations for each tag every 10 minutes are used, where each observation represents the median value observed for that period. Data from when the plant is not operating in a steady state are excluded (recorded as NA), because the behaviour of the plant is not predictable in this state and therefore is not accounted for in the models. Additionally, data from when the tank at the plant is being drained are excluded; when a ship is present the plant does not operate normally, since, for example, there are extra gases coming from the ships that affect conditions in the plant.

There is a clear annual seasonal pattern to the production data, Fig. 7.1, with high

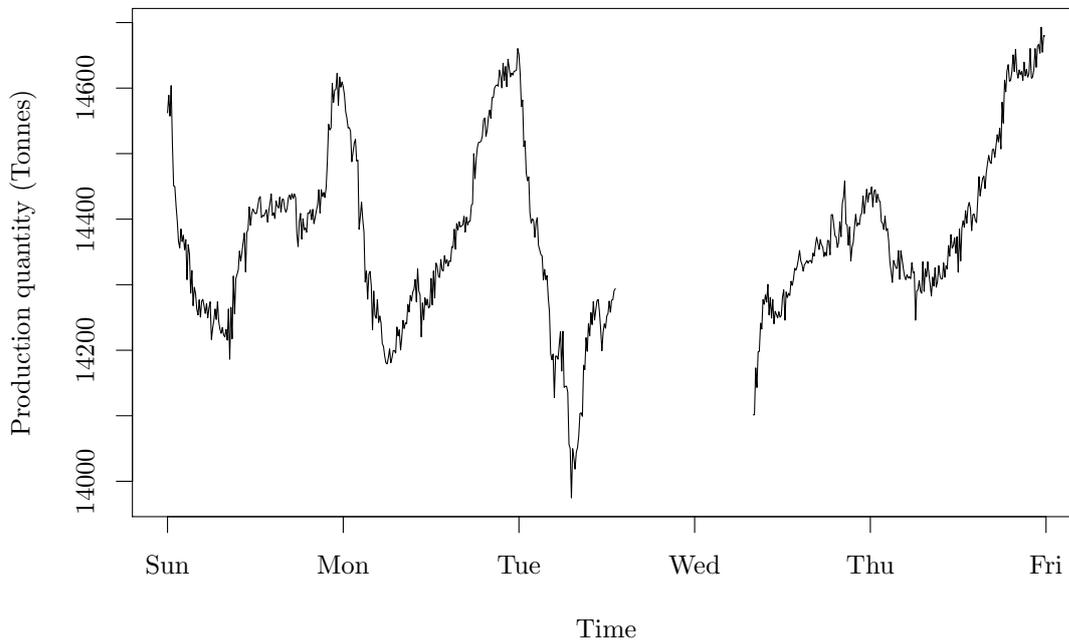


Figure 7.2: An example of production levels of LNG over five days.

levels of production in the winter months (May to September) and lower production levels in the summer when weather conditions are very hot. The summer months show an increased variability in production levels and there is also an increasing trend in average production levels from year to year, both of which contribute to non-stationarity. Fig. 7.2 provides an illustrative example of a closer look at production levels from day-to-day; there is a cyclic diurnal pattern, with peaks in production levels over night and lower levels during the day. Like the annual seasonality, this coincides with a rise and fall in temperatures. The shape of the time series on a daily basis is the same throughout the year, however in winter months, the variance in production levels is smaller whilst atmospheric temperatures are more stable. The daily production quantity profiles are not dissimilar to the glucose profiles in Fig. 2.1, but with fewer daily peaks and troughs. Similar models to those developed for forecasting glucose levels should therefore be explored for forecasting future production levels.

7.2.1 Production vs weather-related tags

The aim of this chapter is to forecast production levels of LNG, with additional information about the weather included in models to improve forecasts. The main tags of interest are therefore temperature tags, atmospheric pressure tags, wind speed and wind direction tags. Temperature data are readily available from PGP from sensors placed around the

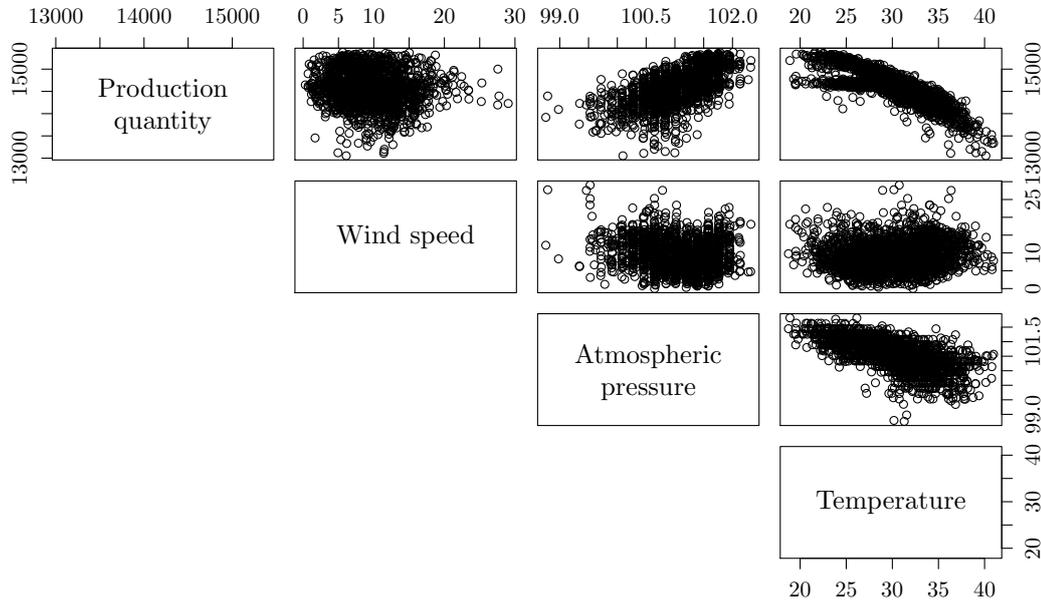


Figure 7.3: Pairs plots of production quantity against weather-related tags (thinned by 50 for plotting).

plant. These tags record temperatures that are strongly correlated with the atmospheric temperature and so are used to develop models initially. One temperature tag recorded at a plant inlet, where air is drawn in from outside, is selected for use.

Fig. 7.3 is a pairs plot of production levels and the weather tags, without wind direction. The plots show a strong negative correlation between temperature and production, as expected, which is approximately linear. There is a moderate positive correlation between atmospheric pressure and production, but this variable is also negatively correlated with temperature, suggesting collinearity. Temperature, which has a stronger causal relationship with production levels, is therefore favoured over atmospheric pressure. There is little evidence to suggest that wind speed has an effect on production.

To visualise how wind direction might affect production levels, a windrose plot is used, Fig. 7.4. This radial plot shows the production level according to wind direction and is coloured by the wind speed. As wind gusts are common, this plot determines whether there is a particular direction of the wind that impacts production levels more. The windrose plot shows some non-uniformity in the density of wind direction, suggesting southerly winds are not as common, however where the observations in this direction do occur, they tend to be associated with higher levels of production. Likewise, northerly winds also have fewer lower levels of production associated with them. The lower levels of production tend to occur when winds are coming from the north west or north east. The plot confirms

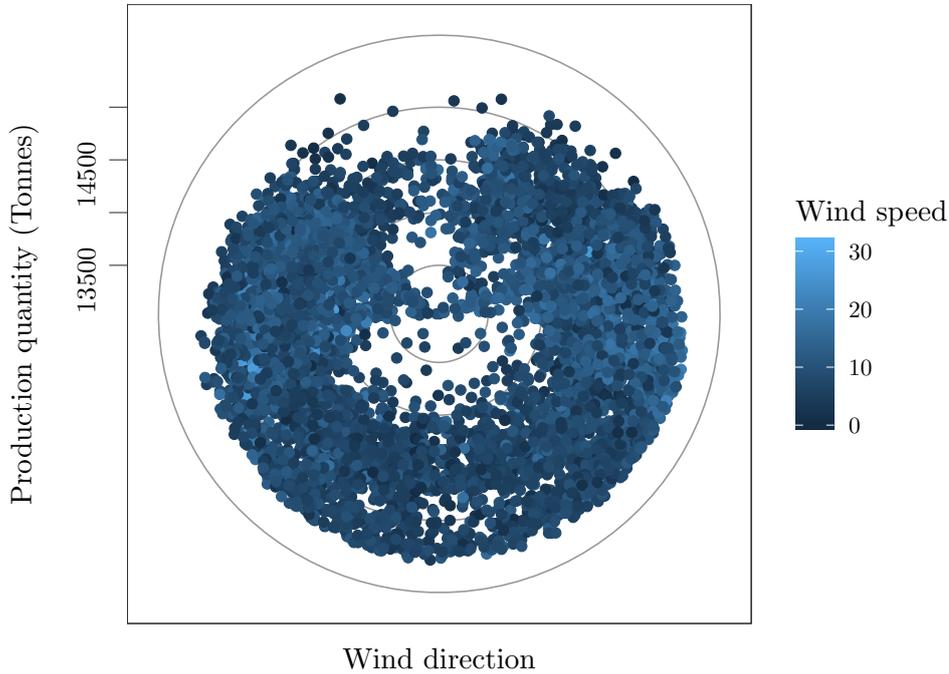


Figure 7.4: Radial plot of wind direction vs production quantities, coloured by wind speed.

that wind speed does not seem to affect production levels as much, since lighter coloured observations are not clustered in one particular area of the plot.

7.3 DLMS for production data

The time-varying mean of the production data along with their seasonal pattern make DLMS good candidate models to apply. The seasonality of production levels can be considered at two levels: annual and diurnal. The annual periodicity includes the variation caused by the change in season and is important for long-term forecasts that focus on the overall trend of the data, that smooths over daily fluctuations. Including the detail of daily production levels at this level would require a large number of harmonics, for example, to be able to forecast the production levels for the next few days. As the annual change is gradual, for short-term forecasts of up to a few days ahead it is more important to capture the diurnal pattern. A period of one day is easier to model since there are fewer observations per period and a time-varying local level can be incorporated to adjust for the annual periodicity.

A local Fourier model, with the pairs of variances of the q Fourier harmonics constrained to be equal as in Model 5.4, is therefore developed for these data. The period of the data is $s = 144$, one observation every 10 minutes over one day, approximated by $q = 2$ harmonics.

Due to the large volume of data and time constraints when work was carried out, the parameters of the DLM, $(\psi_y, \psi_1, \dots, \psi_{q+1})$ are learned through maximum likelihood estimation using the `d1mMLE` function from the `d1m` package. Finding the model parameters in this way took less than 30 minutes per model. The maximum likelihood estimates of each parameter are the values that maximise the complete data likelihood. These values are point estimates that do not incorporate parameter uncertainty. Each model's parameters are learned using the training data set (first 60% of the data), followed by out-of-sample validation on the remaining 40% of the data. Model forecasts are produced for predictive horizons of up to 24 hours, as weather forecasts after this become less reliable. It takes around 20 minutes for settings in the plant to change in reaction to conditions, so forecasts must be accurate for a minimum of 30 minutes ahead of time.

7.3.1 Results

The maximum likelihood estimates of the model parameters are $(\psi_y, \psi_1, \psi_2, \psi_3) = (1, 012.03, 149.88, 39.20, 135.29)$. The initial state hyperparameters $m_0 = (0, \dots, 0)$ and $C_0 = \text{diag}(1 \times 10^7, \dots, 1 \times 10^7)$ are the default `d1m` function parameters; changing these initial values has little effect on the overall model. A plot of the smoothed states in Fig. 7.5 shows that two harmonics fit well to the data, as there is no evidence of seasonality in the local level component and the main fluctuations in production quantity are represented in the two smoothed harmonic components.

To assess the model performance, an array of 1,000 one-day-ahead forecasts is taken at a random sample of 2,000 out-of-sample time points in the testing data. The forecasts ignore the epistemic uncertainty in the parameter values because they are simply conditioned on the plant estimates for parameters when issuing forecasts. Production quantity forecasts are considered accurate by industry standards if they fall within ± 100 Tonnes of the observed production levels. The mean forecast is calculated for each array of forecasts and the proportion of forecasts within the accuracy interval is calculated. For this model, the proportion of forecasts within ± 100 Tonnes for one-day-ahead forecasts is 65%.

Examples of out-of-sample one-day-ahead forecasts are shown in Fig. 7.6. The mean forecasts closely follow the pattern of production levels observed in the previous 24 hours, which is effective for the parts of the series that continue in the same trajectory as before, but for days where the behaviour in production levels changes, for example the fourth forecast in the figure, the predicted production levels are inaccurate. As there are fewer factors influencing production levels than there are on glucose levels, the model performs reasonably well.

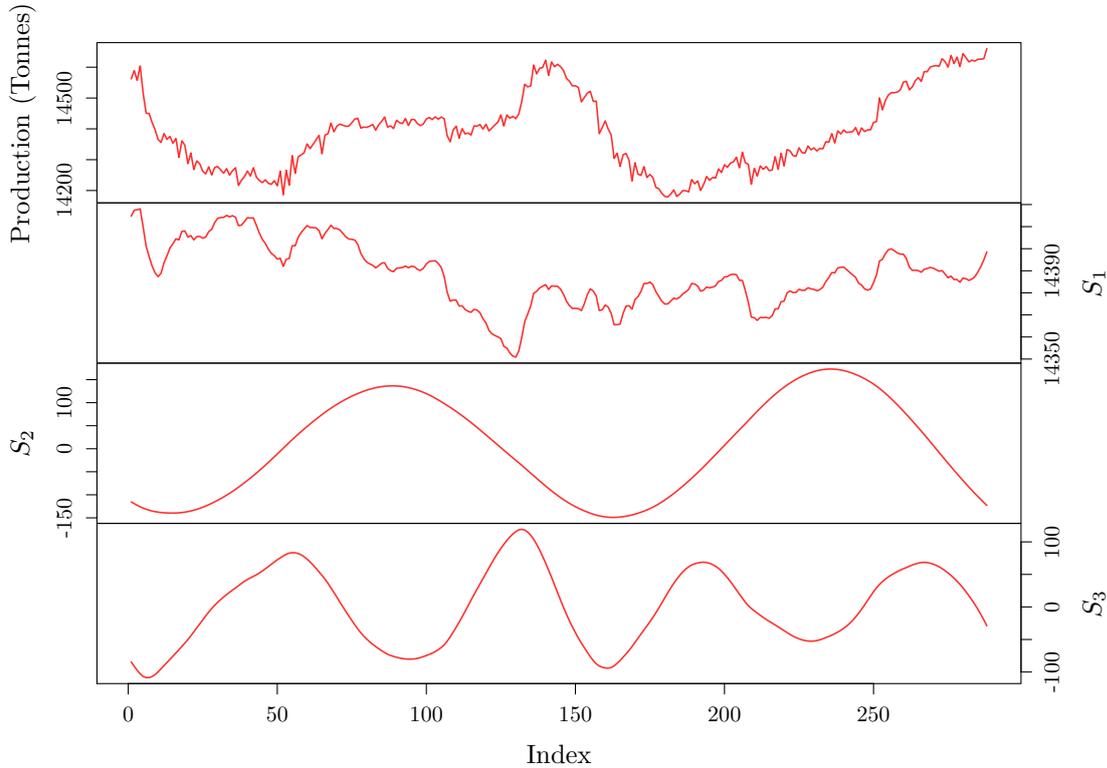


Figure 7.5: The first two days of production data (top) with the corresponding smoothed states. The second row is the smoothed local level component S_1 , then S_2 and S_3 are the smoothed Fourier components θ_3 and θ_5 .

7.4 Modelling production with temperature

To improve the forecasts for production levels, a model including the exogenous variable temperature is fitted. Temperature is the main factor that influences the plant settings at PGP and Fig. 7.3 shows the strong relationship between temperature and production quantity. For low temperatures ($< 27^\circ\text{C}$) and very high temperatures ($> 37^\circ\text{C}$), operators at the plant control LNG production slightly differently than for the mid-range temperatures, which explains the change in relationship between temperature and production in these areas of the pairs plot. For the purpose of the models in this section, the relationship is assumed to be linear, as for the majority of temperatures it is.

A linear model that regresses production quantity, y_t , on the current temperature, x_t , is fitted to the training data using least squares (Dekking *et al.*, 2005). The fitted linear model is:

$$y_t = 16782.99 - 74.79x_t + \varepsilon_t, \quad (7.1)$$

where ε_t are the residual errors between the fitted values and the observations. The model residuals are plotted in Fig. 7.7. Comparing this plot with the production quantities in

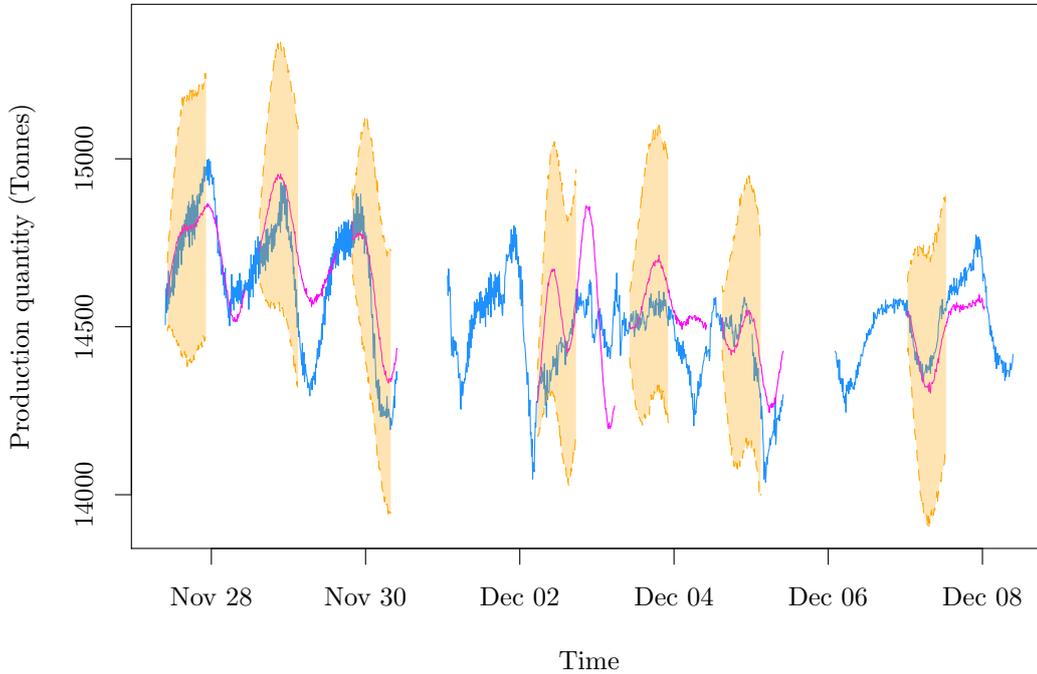


Figure 7.6: One-day-ahead forecasts (magenta) for production levels with 95% prediction intervals (orange) after fitting a local Fourier model.

Fig. 7.1, regressing on temperature alone has reduced the annual seasonal trend, though there is a prominent increasing mean. The model residuals should be a white noise process, which is clearly not the case. Therefore a more complex model that allows for the clear autocorrelation between the residuals is needed. The figure shows that a linear model alone is not enough to predict production quantities based on temperature. A closer look at the daily residuals, Fig. 7.8, shows a diurnal pattern remains.

The results from the linear model demonstrate how regressing production on temperature can partially detrend the data, making production levels easier to predict. A DLM that incorporates this information is described in Model 7.1.

Model 7.1

$$y_t = F\theta_t + \beta x_t + v_t, \quad v_t \sim N(0, V),$$

$$\theta_t = G\theta_{t-1} + w_t, \quad w_t \sim N(0, W),$$

where

$$F = \begin{bmatrix} 1 & 1 & 0 & 1 & \dots & 0 & 1 \end{bmatrix}, \quad G = \text{blockdiag}(1, H_1, \dots, H_q),$$

$$H_j = \begin{bmatrix} \cos \omega_j & \sin \omega_j \\ -\sin \omega_j & \cos \omega_j \end{bmatrix} \text{ and } \omega_j = \frac{2\pi j}{s} \text{ for } j = 1, \dots, q,$$

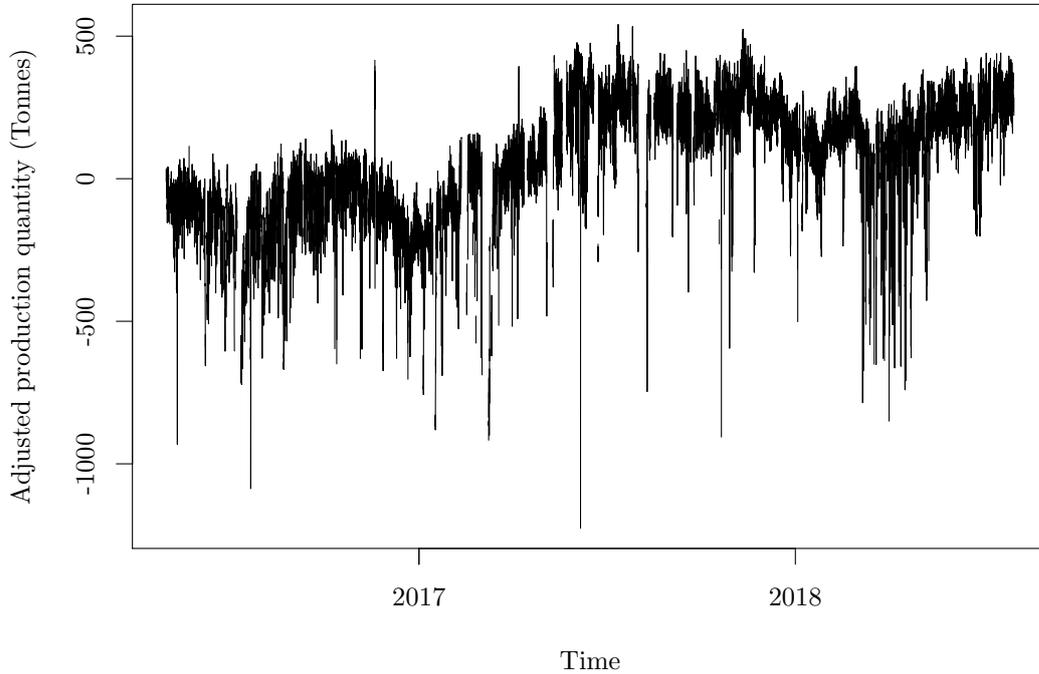


Figure 7.7: Adjusted production levels of LNG over time after regressing on temperature (thinned by five for plotting).

$$V = \psi_y^{-1}, \quad W = \text{diag}(\psi_1^{-1}, \dots, \psi_{q+1}^{-1}).$$

Model 7.1 is a local Fourier model with constrained harmonic variance pairs, plus the term βx_t , a regression on the temperature at time t . The DLM still contains a Fourier term to describe the remaining diurnal pattern in the data and the local level is described by a random walk. This is suitable since the mean is non-reverting. For computational convenience the regression coefficient is taken as the model coefficient in Eq. 7.1, $\beta = -74.79$, whilst the intercept term in the linear model is not needed, as this is incorporated into the random walk component. The model was tested with $q = 1$ and $q = 2$ harmonics and the second harmonic did not significantly improve forecasts, so results for the $q = 1$ model are presented.

7.4.1 Results

The maximum likelihood estimates of the parameters for Model 7.1, with $\beta = -74.79$, are $(\psi_y, \psi_1, \psi_2) = (910.95, 57.86, 279.41)$. The initial state hyperparameters are the same as for the univariate DLM.

An array of 1,000 one-day-ahead forecasts is produced for the same 2,000 random

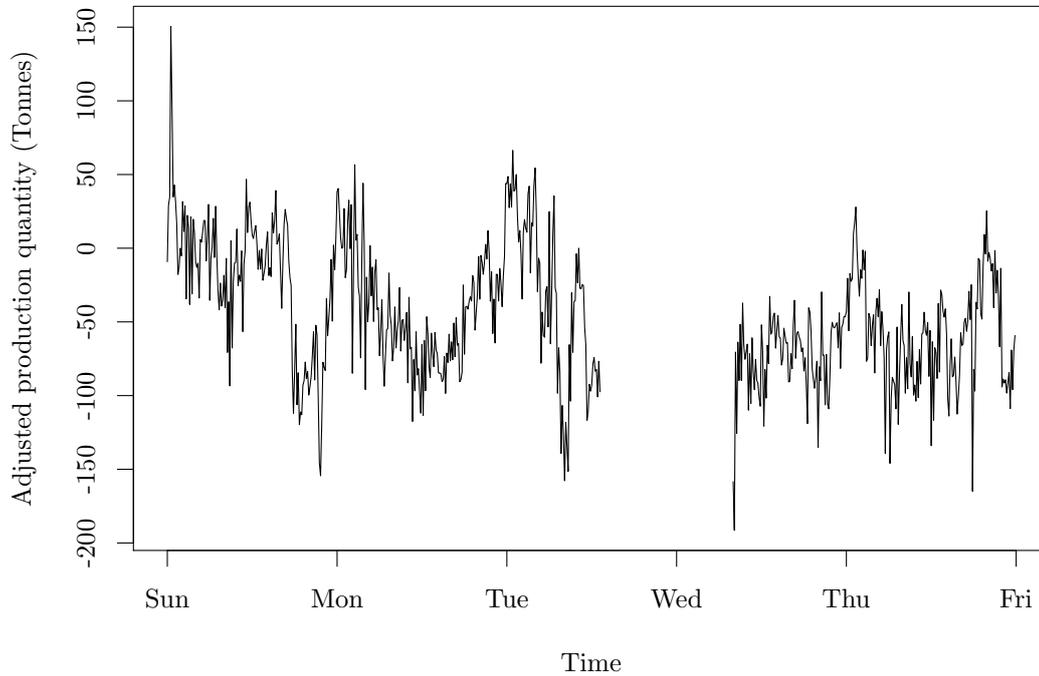


Figure 7.8: Adjusted production levels of LNG over five days after regressing on temperature.

out-of-sample time points in the test data. The temperatures used for the forecasts are the observed temperatures for the forecast period (sensitivity to errors in temperature forecasts is explored in the next section). The mean of each array is then used to calculate the proportion of forecasts within ± 100 Tonnes of the observed series, which is calculated as 76%. This is a considerable improvement to the univariate model. For shorter term forecasts, the proportion of observations that are considered accurate is even higher; a plot of the mean proportion of observations within ± 100 Tonnes for predictive horizons between one and 24 hours ahead is in Fig. 7.9.

Example forecasts are shown in Fig. 7.10. The forecasts show how powerful including an explanatory variable in a model can be, with the forecasts following the shape and fitting very closely to the observed series.

7.4.2 Forecast sensitivity

The accurate forecasts of production from Model 7.1 are based on the known, precise future temperatures at PGP. Here the model is tested on how sensitive the forecasts are to less accurate temperature forecasts. How far out a temperature forecast can be before the production level forecast becomes unreliable and whether forecasts are better in winter, when weather is steadier, than in the summer are investigated.

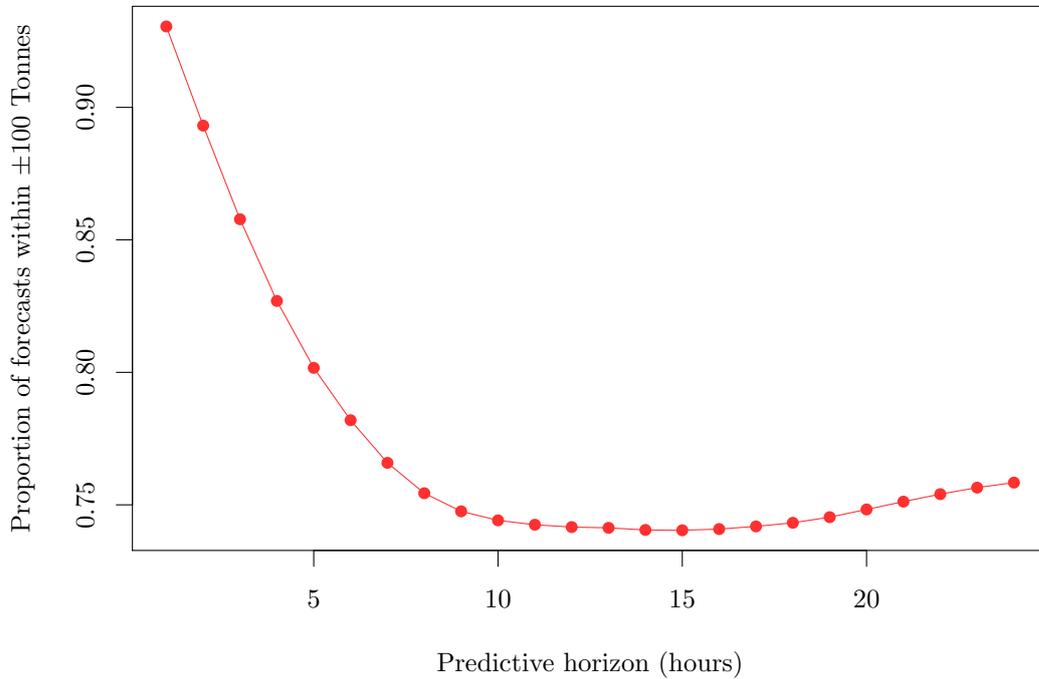


Figure 7.9: Mean proportion of observations within ± 100 Tonnes for forecasts with predictive horizons between one and 24 hours ahead from Model 7.1.

The temperatures used in forecasts are perturbed between $\pm 4^\circ\text{C}$ by increments of 0.5°C , allowing for a systematic additive bias for demonstrative purposes. The proportion of forecasts within ± 100 Tonnes for a range of predictive horizons are shown in Fig. 7.11. The level of accuracy for forecasts with temperatures within $\pm 1^\circ\text{C}$ of the true values is reasonable for each predictive horizon, however the proportion of accurate forecasts decreases quickly thereafter. As forecasts are only being made a day in advance, temperature forecasts for the following day can be assumed to be close to the actual temperatures, so forecasts should still fall within an acceptable range of future values.

A closer look at the forecasts produced by the model on summer data compared to winter data showed that there is no difference in forecast accuracy between the seasons. As the model is dynamic, it adapts well to changes in the mean production levels in the data.

7.5 Model improvements and further work

Model 7.1 is developed based on the relationship between production quantity and the temperature observed at an inlet of the plant. In real life, if weather forecasts of the

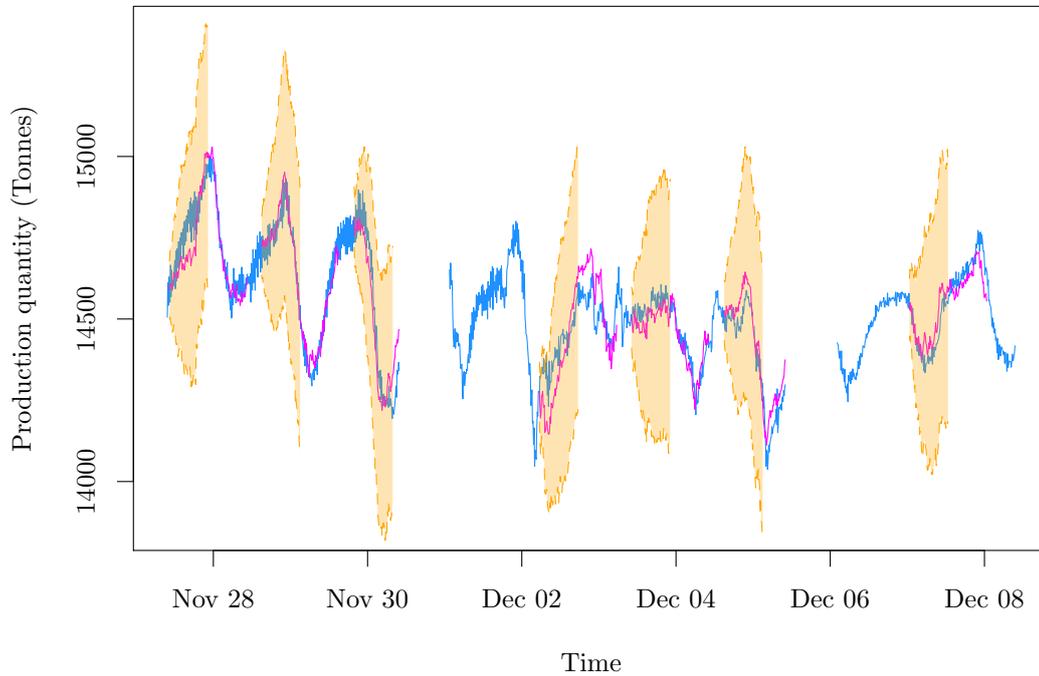


Figure 7.10: One-day-ahead forecasts (magenta) for production levels with 95% prediction intervals (orange) after fitting Model 7.1.

ambient temperature local to PGP are used for predicting LNG production, these must be adjusted in relation to the inlet temperature, or the model should be fitted based on historical ambient temperature data instead. Plant temperatures should be strongly correlated with the ambient temperature, but any differences could have a big impact on the parameters, in particular the regression coefficient, that are used in the model. The suitability of the model should then be reassessed based on the incorporation of local weather data. If possible, Bayesian methods of parameter inference should also be explored to allow proper incorporation of parameter uncertainty in forecast distributions, as well as jointly fitting the model in real-time. For further model improvements, including wind direction in the model could also be explored.

7.6 Summary

This chapter demonstrates the vast improvement that can be made by incorporating additional information in a model via an explanatory variable. By explaining a large proportion of the variation in the production quantity of LNG caused by the ambient temperature at the plant, better forecasts were achieved. The sensitivity of production forecasts to

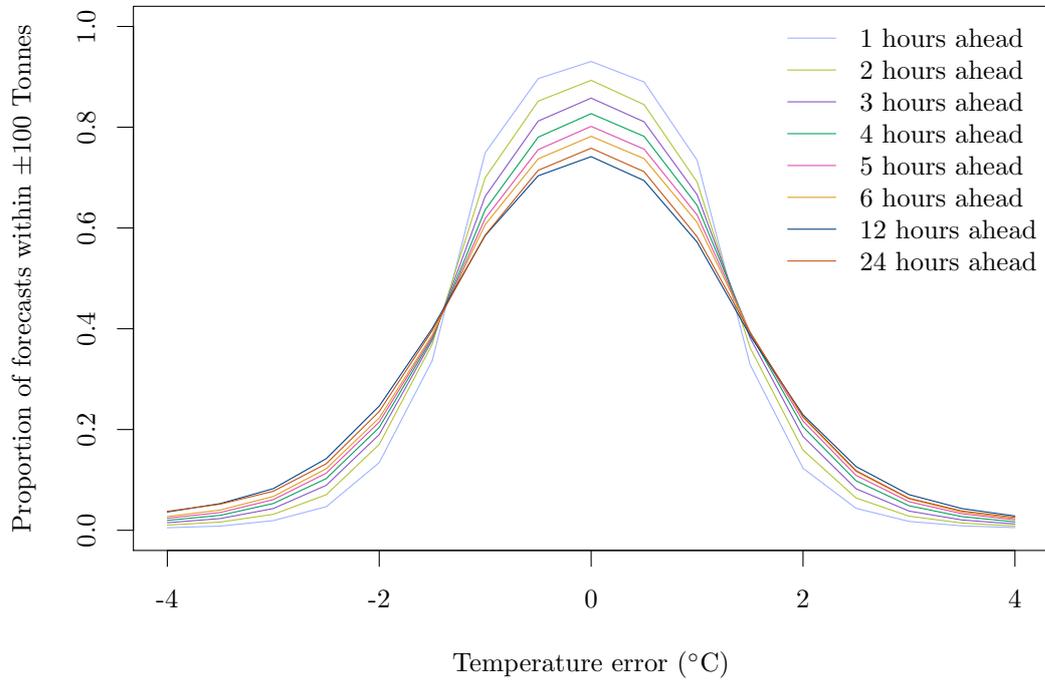


Figure 7.11: Proportion of forecasts from Model 7.1 within ± 100 Tonnes for a range of temperature errors.

perturbations in the temperature variable was also tested, demonstrating how accurate the weather forecast would need to be for production forecasts to be reliable if the model was to be used in practise.

Chapter 8

Conclusions and future work

8.1 Objectives and contributions of this thesis

The primary objective of this thesis was to develop a bivariate model for glucose levels and activity levels of type II diabetes patients and to fit the models in a Bayesian framework. This was achieved via DLMS for glucose levels combined with a regression component to incorporate information about activity data, which were classified into activity intensity zones by a HMM. The secondary goal was to investigate whether other information, such as food intake, could lead to improvements in the model. A joint model involving the number of grams of carbohydrates and glucose levels was developed for this purpose. A model for the secondary application of forecasting the amount of LNG produced at a gas plant was also investigated.

The modelling chapters of this thesis (4, 5 and 6) each contribute to an improved understanding of the complexity of jointly modelling noisy glucose and activity data. Additional to this, the models fitted in Chapter 7 in a very different problem context illustrate the flexibility of the types of models developed in this thesis. Both of the applications explored are novel areas for such time series models to be applied.

Previous research has involved transforming accelerometer data using different activity summaries and in some cases comparing methods, such as the work by van Hees *et al.* (2013), or using HMMs to model these data (Huang *et al.*, 2018), but a comprehensive comparison of different summaries with the transformed data classified using a HMM was not available. The HMMs fitted in Chapter 4 contribute a thorough exploration into the most suitable activity summary and within-state distribution for representing the perceived activity intensity zones of patient activity from accelerometer data. This analysis gave rise to further questions regarding which activity observations could be clustered together for monitoring their impact on glucose levels.

The univariate glucose models in Chapter 5 explored the use of dynamic models to

better describe the irregular seasonal patterns in the data, to improve upon classic AR models that had mainly been used in the literature on glucose prediction to date. These models were fitted using Bayesian inference to incorporate the model parameter uncertainty into the forecasts, then various models combining Fourier and AR components were developed and compared in an effort to improve the forecasts' accuracy and precision.

In Chapter 6, the posterior modal classification of the log-transformed BFEN activity data according to the Normal HMM from Chapter 4 was combined with the local Fourier DLM from Chapter 5 to build a bivariate model to improve glucose forecasts. This contributed to understanding the relationship between glucose and activity and how this differs between individuals. The model between food intake and glucose showed the impact this variable also has on glucose levels, though these data were only available for two patients.

Finally, the models developed in Chapter 7 incorporating temperature and the amount of LNG produced at a gas plant provided a novel solution to forecasting these quantities. The forecasts provide more accurate estimates of future LNG production levels compared to regression models that have been used in the past based on other variables in the plant. The model also serves as a good example of how, where a clear relationship between variables can be learned, model forecasts can be greatly improved.

8.2 Conclusions

In Chapter 1, the problems of glucose prediction and activity monitoring were introduced, with previous research grouped into three main areas: current glucose models and methods to assess their performance; the physiological impact of activity on glucose levels; and methods for pre-processing accelerometer data. The review of literature surrounding these topics provided an initial understanding of what had been done already for predicting glucose levels of type II diabetes patients and provided a wealth of information to be considered for the models that were subsequently developed. Whilst there had been several glucose-only models developed previously, work had not been done to model activity alongside glucose levels, despite there being vast amounts of research showing that activity influences glucose levels. The models previously developed and the literature surrounding the physiological impact of exercise on glucose levels provided a good basis for models to be explored and developed in this project.

Chapter 2 gave an overview of the glucose and accelerometer data available. A range of pre-processing methods for accelerometer data discussed in Chapter 1 were applied to a sample of patient accelerometer data to compare how their activity summaries differed and to assess their suitability for monitoring activity levels. Three methods were then chosen to evaluate further: an unfiltered summary (ENMO), a filtered summary (BFEN)

and a step count algorithm. Choosing to model three summaries in Chapter 4 provided insight into how flexible the activity models are, as well as highlighting any key differences in how they are classified. An exploration into the glucose data for eight diabetic patients showed that the density of the data for each patient could be described by a mixture of Gaussian densities, however the mean and variance of the mixtures were quite different between patients. The seasonal pattern within each profile consists of a number of peaks and troughs per day, but this number varies between patients and sometimes within the profile of each patient, too. This highlighted the need for a flexible model to be developed for glucose levels. It was also found that some of the profiles suffered from missing data, in varying levels of missingness, and so the models fitted needed to account for this. An initial investigation into the relationship between glucose and activity levels gave some insight into the non-homogeneous effect of activity on glucose over time.

Previous research surrounding the impact of activity on glucose levels showed that the type of activity being performed was not important in producing an effect and that the subject simply needed to be active. This provided the basis for the models developed in Chapter 4, where activity levels were classified into low-, medium- and high-intensity activity zones. The theory behind HMMs was explained in Chapter 3 along with the computational Bayesian inferential tools needed to fit them. Initially, a HMM with zero-order dependency, known as a mixture model, was fitted to the 1 Hz BFEN data. Whilst the posterior within-state distributions classified the data into three distinct states, this model saw a high number of transitions between states. This model provided a baseline for the traditional HMMs with a first-order Markov dependency between states that were fitted next, which saw a reduction in the number of transitions between states to a more physiologically realistic amount. The HMMs were fitted using Normal and skew Normal within-state distributions to the three activity summaries chosen in Chapter 2. In developing these two candidate models, an equal variance constraint was added to encourage the a posteriori identification of distinct means for the within-state densities, which improved the fit of the models. Adding in the skewness parameter improved the classification results particularly for the ENMO data, which had a long right tail and could not easily be transformed due to the negative observations. The BFEN and step count activity summaries were transformed using a log-transform and square-root transform, respectively, which reduced the skew in the data and reduced the need for a skewed within-state distribution in the HMM. The activity summaries were downsampled to a frequency of one observation every five minutes. This reduced the amount of time needed to run the MCMC algorithms to fit the HMMs by reducing the overall size of the data sets and also aligned the frequency of the activity data with that of the glucose data. This transformation also made the distinction between periods of activity and inactivity more prominent in the data.

The results from the activity model HMMs showed that the activity summary used to transform the tri-axial accelerometer data has a large impact on how the activity at each timestamp is interpreted. More specifically, there was a big difference between the step count and BFEN classifications compared to the ENMO classified data. After collecting labelled activity data, the ENMO activity summary was ruled out as a useful summary for use in future glucose models, as classification of this summary failed to detect some labelled higher intensity activities. The step count and BFEN summaries showed that the HMMs were versatile enough to produce similar classification results for discrete (step count) and continuous (BFEN) summaries. Comparing the Normal HMM on the log-transformed BFEN data of a subset of diabetic and non-diabetic patients, overall the posterior within-state distribution parameters were similar, which indicated that a HMM could be developed on one patient and used across other patients, without losing much accuracy in the classification of the realised activity observations.

The glucose models fitted in Chapter 5 proposed two main methods for capturing the seasonality of the glucose profiles: using a high order AR model, or AR component in a DLM, and using a Fourier seasonal component in a DLM. The challenges of capturing this seasonality arose from the irregular periodicity from day-to-day, due to different external factors, such as food and exercise. High order AR models were not sufficient in capturing the extreme peaks and troughs in the data, often under- and overestimating these values, respectively. For glucose prediction, accurately predicting a hyper- or hypoglycaemic glucose level is more important than correctly predicting the euglycaemic values, as these extremes are the glucose levels that require intervention from the subject. Improving the glucose models was therefore necessary and this was done by adding in a Fourier seasonal component. The DLMS with a Fourier component improved the prediction of the extreme glucose values. The forecasts from these models most closely followed the pattern of the glucose levels observed the previous day, so where glucose levels in the following period were quite different, the model did not perform as well. DLMS that included both a higher order AR component and a Fourier seasonal component were also fitted, but the combination of both elements did not provide a much better fit than when fitting a Fourier component with a local level component. Whilst a DLM adapts to changing variance and allows for non-stationarity, there was still room for improvement that came in the following chapter when additional information about some of the external factors influencing glucose was included. This chapter also showed how model parameters differed between patients, meaning that a universal glucose model was not a viable option. The models performed better for patients whose glucose profiles have less variability, namely the non-diabetic patients. Having some non-diabetic patient data was a useful comparative reference for the models fitted to diabetic patients throughout the thesis.

The joint models fitted between glucose and activity data in Chapter 6 combined the

results from the Normal HMM fitted to the log-transformed BFEN data with the local Fourier model fitted to the glucose data via a regression on the posterior modal activity level from the HMM. This method of Markov melding showed some promising results in which the forecasts of some of the patient glucose levels improved. The results showed how the amount of information available in each patient's data varied, with only a small effect from activity being detected in some cases, leading to only a small improvement in glucose forecasts for these patients. One of the reasons that the cumulative effect of activity on glucose levels is more difficult to detect for some patients could be because of the different CRF levels of patients. The effect of activity on glucose levels for those with higher CRF levels to begin with is smaller, as shown by McCarthy *et al.* (2017). This is consistent with what is seen in the joint model results for Patient 3 and Patient 5; diabetic patients tend to have lower CRF levels and therefore poorer metabolisms, so glucose levels rise more postprandially than for non-diabetic patients and then improve by a larger percentage when activity is performed. For Patient 3 and Patient 5, the cumulative effect of activity reduced glucose levels by over 1 mmol/L. For the non-diabetic patients and Patient 10, the effect from activity was much smaller, which could be because there is less of an impact from activity on glucose levels because of higher CRF levels for these patients.

In Chapter 6, a joint model involving the number of grams of carbohydrates consumed was fitted to two of the additional patients, where these data were available. One of the challenges with including data on the food that patients have consumed in models is in data quality; food data were collected for all three of the additional non-diabetic patients, however the information failed to log for one of the patients (ND 2) and tracking food intake for the other two patients required a lot of effort. One of the reasons for using accelerometer data instead was that they are much easier to collect, since it involves minimal intervention from the patient. They are also therefore less subject to human error, since the data are collected by a wrist-worn activity watch, then transformed using the appropriate algorithm. Nevertheless, it was useful to fit a model between glucose and number of grams of carbohydrates consumed because food intake is known to be the main influencing factor over glucose levels. It was surprising that the model did not therefore produce a bigger improvement in forecasts, with only a better fit immediately after the event at the start of where glucose levels began to increase. This implies that the model was still not capturing some crucial information that drives glucose levels, or possibly is not fully capturing the relationship between glucose and carbohydrate consumption.

In summary, a HMM with sensible within-state distributions that reflected realistic underlying activity zones for the BFEN and step count data sets was developed, which was able to provide useful activity summaries, namely the posterior modal state for each time point, to be used in joint models with glucose data. The dynamic glucose models developed provided a good baseline in predictive performance, which captured the periodicity in the

glucose profiles to a good degree of accuracy. The bivariate models between glucose and activity and glucose and the number of grams of carbohydrates consumed both showed improvements to forecasted glucose levels when compared to the univariate glucose models. This showed how incorporating exogenous variables in the model can lead to improved fit. The scaled Gamma curve activity DLM showed the biggest improvements to forecasts, when the relationship between glucose and activity was detectable, but this was very much patient dependent.

The models for monitoring the production quantity of LNG produced in Chapter 7 showed the vast improvement that can be seen in forecasts when an explanatory variable is included and when the relationship between the explanatory and dependent variable can be easily learned. A local Fourier DLM was fitted to production levels alone as a baseline, then a regression on the temperature at the plant was added to the model. As with the univariate local Fourier glucose DLM, the forecasts of production levels for the following day most closely followed the production levels of the previous day, which therefore had varying success depending on how similar successive days were. Fitting a linear model between production quantities and temperature detrended the production data a good amount, since temperature is one of the main factors in how well the LNG plant performs on a particular day and this relationship is approximately linear. The temperature coefficient learned from this model was then included in the production DLM, along with the temperature covariate. The forecasts from the model that included additional information on temperature were much more accurate, closely following the shape of the production quantity profile.

8.3 Further work

This section highlights some possible future directions for continuing the work on monitoring and forecasting glucose levels of type II diabetes patients.

One of the obvious branches of the work in this thesis to follow is to fit the bivariate glucose and activity model in real-time. A method for doing so is outlined here. The joint model consists of two parts: a HMM to learn the hidden activity zones and a DLM that incorporates past and current glucose and activity data. At time $t = 0$, there are no data to condition on, so uncertainty about the parameters of the models are summarised through their prior distributions. As data are collected, the posterior distributions of the parameters can be updated using MCMC methods. The process of performing these parameter updates will happen in two stages: firstly, the accelerometer data are pre-processed using the BFEN algorithm (2.3) over five minute windows and the HMM MCMC algorithm (3.5) is run, learning the posterior within-state distributions and sampling the hidden states upon running the forward-backward algorithm (3.4) on each

iteration; secondly, the scaled Gamma curve DLM MCMC algorithm (6.2) is run, incorporating the posterior modal activity states from the first stage. Ensembles of filtered values and forecasts can then be produced from the bivariate DLM using posterior draws for the model parameters.

At the end of the first day of data collection, the posterior densities of the model parameters are computed overnight based on the available data, then the MCMC scheme for updating these densities will not be run again until the following night. This is because the block MCMC schemes described in this thesis are not amenable to sequential use in real-time. Over the day, with parameters fixed at a value from their joint posterior, the filtering recursions can be used to classify activity levels and to summarise uncertainty about the DLM state using data up to time t . As each new data point becomes available, this can be advanced one step forward. On the basis of these filtered distributions, forecasts can be issued at any time.

The results in this thesis for activity models showed that the posterior within-state distributions for the Normal HMM fitted to the log-transformed BFEN activity summary were similar across all patients. Therefore because this was the chosen activity summary and model combination for use in the bivariate glucose model, it would be interesting to learn the parameters of the joint model on one patient and assess its performance on other patient data. This could reduce the amount of computational inference needed across patients. Alternatively the posterior for one patient could be used to develop an informative prior for other patients. Improvements to fitting the model in real-time could be continued by using sequential methods for state space models, such as sequential integrated nested Laplace approximation (INLA) (Martino & Riebler, 2014), for forecasting online. Sequential INLA is a fast, deterministic approximation of posterior quantities and so does not take long to converge or suffer from poor mixing. Another option is to use a particle filter method, such as iterated batch importance sampling (IBIS) (Chopin, 2002). This algorithm iteratively samples, reweights and resamples values for the parameters, which sequentially produces a discrete approximation for the posterior distribution. This method is more efficient than MCMC and is parallelisable (Lai *et al.*, 2020), making it more suitable for online inference. A faster programming language for fitting models could also be used.

Continuing on from forecasting glucose levels in real-time, an investigation into the glucose levels that should trigger a behavioural prompt for each patient could be conducted. The glucose profiles of each patient in this project showed that the boundaries for a hyper- or hypoglycaemic glucose recording would not be practical levels to trigger a behavioural prompt for everyone, as this would result in almost all observations and their forecasts producing a prompt. Adapting this per person would therefore be necessary and an interesting area to research. The frequency of behavioural prompts could then also be

investigated.

Methods to perform error-grid analysis, used to assess model performance based on whether or not a glucose prediction falls in the same range as the observed glucose levels (hyper-, hypo- or euglycaemic ranges), were introduced in Chapter 1. It would be interesting for clinicians to have this information, however, as with the behavioural prompts, it is clear from the patient data available that the actual ranges of euglycaemic glucose levels are subjective for type II diabetes patients and would therefore need to be looked into more carefully. This analysis is therefore left as future work.

In terms of model improvements, further improvement to the models developed in this thesis could be made by collecting, analysing and incorporating data on other variables that influence glucose and improving the quality of the data on food intake. One of the factors mentioned in the conclusion is the CRF levels of patients, which is a known influencer of glucose levels. Knowledge of patients' CRF levels prior to data collection, and possibly during, could be incorporated into models to see whether this coincides with a smaller realised activity effect in some patients. The way a glucose profile develops over time is also largely determined by the body's insulin response, which does not respond in the same way for diabetic people and this causes higher glucose levels. More knowledge of insulin at an individual level could therefore improve models further. The amount of patient intervention for the collection of data on additional variables should however be considered, as increased interaction leads to more human error and is also more likely to discourage subjects from collecting data for longer periods. Food intake data were incorporated in the bivariate model as the number of grams of carbohydrates consumed; an alternative to collecting these data directly would be to learn when a patient has eaten. This 'random eating process' would then underpin the glucose model. Fitting a model that involved glucose, carbohydrates and activity, based on the models already developed, is also a natural next step to take.

To make the models fitted more flexible, the number of hidden states in the activity HMM could be learned. This would not necessarily be a vital step for the log-transformed BFEN activity summary, since there seem to be three clear hidden states for most patients, however a HMM of unknown order might better suit the step count summary, whose square-root transformed data were trialled with four hidden states. Likewise, the number of harmonics in the Fourier component of the DLMS could be learned, but the added computational complexity may mean this approach is not worthwhile.

Finally, models to 'correct' the glucose values reported by CGM sensors, such as those by Facchinetti *et al.* (2013), are available. As the values reported are based on interstitial glucose levels, there is some discrepancy between these observations and actual blood glucose levels, which is why the CGM device records more accurate observations when it is regularly calibrated by a finger-prick glucose measurement. Adding in a glucose level

'correction' to models could therefore improve the inferred relationship between glucose and activity, for example.

References

- AHLQVIST, E., STORM, P., KÄRÄJÄMÄKI, A., MARTINELL, M., DORKHAN, M., CARLSSON, A., VIKMAN, P., PRASAD, R. B., ALY, D. M., ALMGREN, P. *et al.* 2018 Novel subgroups of adult-onset diabetes and their association with outcomes: a data-driven cluster analysis of six variables. *The Lancet Diabetes and Endocrinology* .
- ALBERT, J. H. & CHIB, S. 1993 Bayes inference via Gibbs sampling of autoregressive time series subject to Markov mean and variance shifts. *Journal of Business & Economic Statistics* **11** (1), 1–15.
- ALLEN, N. A., FAIN, J. A., BRAUN, B. & CHIPKIN, S. R. 2008 Continuous glucose monitoring counseling improves physical activity behaviors of individuals with type 2 diabetes: a randomized clinical trial. *Diabetes research and clinical practice* **80** (3), 371–379.
- BERGER, J. O., MORENO, E., PERICCHI, L. R., BAYARRI, M. J., BERNARDO, J. M., CANO, J. A., DE LA HORRA, J., MARTÍN, J., RÍOS-INSÚA, D., BETRÒ, B. *et al.* 1994 An overview of robust Bayesian analysis. *Test* **3** (1), 5–124.
- BERNARDO, J. M. & SMITH, A. F. 2009 *Bayesian theory*, , vol. 405. John Wiley & Sons.
- BODE, B. W., GROSS, T. M., THORNTON, K. R. & MASTROTOTARO, J. J. 1999 Continuous glucose monitoring used to adjust diabetes therapy improves glycosylated hemoglobin: a pilot study. *Diabetes research and clinical practice* **46** (3), 183–190.
- BOX, G. & JENKINS, G. M. 1976 *Time Series Analysis: Forecasting and Control*. Holden-Day.
- BREMER, T. & GOUGH, D. A. 1999 Is blood glucose predictable from previous values? a solicitation for data. *Diabetes* **48** (3), 445–451.
- BROOKS, S., GELMAN, A., JONES, G. & MENG, X.-L. 2011 *Handbook of Markov chain Monte Carlo*. CRC press.

- BUCY, R. S. & JOSEPH, P. D. 2005 *Filtering for stochastic processes with applications to guidance*, , vol. 326. American Mathematical Soc.
- BUTTERWORTH, S. 1930 On the theory of filter amplifiers. *Wireless Engineer* **7** (6), 536–541.
- CARTER, C. K. & KOHN, R. 1994 On Gibbs sampling for state space models. *Biometrika* **81** (3), 541–553.
- CHOPIN, N. 2002 A sequential particle filter method for static models. *Biometrika* **89** (3), 539–552.
- CLARKE, W. L., COX, D., GONDER-FREDERICK, L. A., CARTER, W. & POHL, S. L. 1987 Evaluating clinical accuracy of systems for self-monitoring of blood glucose. *Diabetes care* **10** (5), 622–628.
- CONGDON, P. 2007 *Bayesian statistical modelling*, , vol. 704. John Wiley & Sons.
- CONN, V., HAFDAHL, A., MEHR, D., LEMASTER, J., BROWN, S. & NIELSEN, P. 2007 Metabolic effects of interventions to increase exercise in adults with type 2 diabetes. *Diabetologia* **50** (5), 913–921.
- DEKKING, F. M., KRAAIKAMP, C., LOPUHAÄ, H. P. & MEESTER, L. E. 2005 *A Modern Introduction to Probability and Statistics: Understanding why and how*. Springer Science & Business Media.
- DOHERTY, A., JACKSON, D., HAMMERLA, N., PLÖTZ, T., OLIVIER, P., GRANAT, M. H., WHITE, T., VAN HEES, V. T., TRENELL, M. I., OWEN, C. G. *et al.* 2017 Large scale population assessment of physical activity using wrist worn accelerometers: The UK Biobank Study. *PloS one* **12** (2), e0169649.
- DOUCET, A., DE FREITAS, N. & GORDON, N. 2001 An introduction to sequential Monte Carlo methods. In *Sequential Monte Carlo methods in practice*, pp. 3–14. Springer.
- DUNSTAN, D. W., KINGWELL, B. A., LARSEN, R., HEALY, G. N., CERIN, E., HAMILTON, M. T., SHAW, J. E., BERTOVIC, D. A., ZIMMET, P. Z., SALMON, J. *et al.* 2012 Breaking up prolonged sitting reduces postprandial glucose and insulin responses. *Diabetes care* **35** (5), 976–983.
- FACCHINETTI, A., DEL FAVERO, S., SPARACINO, G., CASTLE, J. R., WARD, W. K. & COBELLI, C. 2013 Modeling the glucose sensor error. *IEEE Transactions on Biomedical Engineering* **61** (3), 620–629.

-
- FRECKMANN, G., SCHMID, C., BAUMSTARK, A., RUTSCHMANN, M., HAUG, C. & HEINEMANN, L. 2015 Analytical performance requirements for systems for self-monitoring of blood glucose with focus on system accuracy: relevant differences among ISO 15197: 2003, ISO 15197: 2013, and current FDA recommendations. *Journal of diabetes science and technology* **9** (4), 885–894.
- FRÜHWIRTH-SCHNATTER, S. 1994a Applied state space modelling of non-Gaussian time series using integration-based Kalman filtering. *Statistics and Computing* **4** (4), 259–269.
- FRÜHWIRTH-SCHNATTER, S. 1994b Data augmentation and dynamic linear models. *Journal of time series analysis* **15** (2), 183–202.
- FRÜHWIRTH-SCHNATTER, S. 2006 *Finite mixture and Markov switching models*. Springer Science & Business Media.
- GAMERMAN, D. & LOPES, H. F. 2006 *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*. CRC Press.
- GANI, A., GRIBOK, A. V., RAJARAMAN, S., WARD, W. K. & REIFMAN, J. 2009 Predicting subcutaneous glucose concentration in humans: data-driven glucose modeling. *IEEE Transactions on Biomedical Engineering* **56** (2), 246–254.
- GARG, S., ZISSER, H., SCHWARTZ, S., BAILEY, T., KAPLAN, R., ELLIS, S. & JOVANOVIĆ, L. 2006 Improvement in glycemic excursions with a transcutaneous, real-time continuous glucose sensor. *Diabetes care* **29** (1), 44–50.
- GOUDIE, R. J., PRESANIS, A. M., LUNN, D., DE ANGELIS, D. & WERNISCH, L. 2019 Joining and splitting models with Markov melding. *Bayesian analysis* **14** (1), 81.
- VAN HEES, V. T., FANG, Z., LANGFORD, J., ASSAH, F., MOHAMMAD, A., DA SILVA, I. C., TRENELL, M. I., WHITE, T., WAREHAM, N. J. & BRAGE, S. 2014 Autocalibration of accelerometer data for free-living physical activity assessment using local gravity and temperature: an evaluation on four continents. *Journal of Applied Physiology* **117** (7), 738–744.
- VAN HEES, V. T., GORZELNIAK, L., LEON, E. C. D., EDER, M., PIAS, M., TAHERIAN, S., EKELUND, U., RENSTRÖM, F., FRANKS, P. W., HORSCH, A. *et al.* 2013 Separating movement and gravity components in an acceleration signal and implications for the assessment of human daily physical activity. *PloS one* **8** (4), e61691.
- VAN HEES, V. T., RENSTRÖM, F., WRIGHT, A., GRADMARK, A., CATT, M., CHEN, K. Y., LÖF, M., BLUCK, L., POMEROY, J., WAREHAM, N. J. *et al.* 2011 Estimation of daily energy expenditure in pregnant and non-pregnant women using a wrist-worn tri-axial accelerometer. *PloS one* **6** (7).

- VAN HEES, V. T., SABIA, S., ANDERSON, K. N., DENTON, S. J., OLIVER, J., CATT, M., ABELL, J. G., KIVIMÄKI, M., TRENELL, M. I. & SINGH-MANOUX, A. 2015 A novel, open access method to assess sleep duration using a wrist-worn accelerometer. *PloS one* **10** (11), e0142533.
- HENSON, J., DAVIES, M. J., BODICOAT, D. H., EDWARDSON, C. L., GILL, J. M., STENSEL, D. J., TOLFREY, K., DUNSTAN, D. W., KHUNTI, K. & YATES, T. 2016 Breaking up prolonged sitting with standing or walking attenuates the postprandial metabolic response in postmenopausal women: a randomized acute study. *Diabetes Care* **39** (1), 130–138.
- HUANG, Q., COHEN, D., KOMARZYNSKI, S., LI, X.-M., INNOMINATO, P., LÉVI, F. & FINKENSTÄDT, B. 2018 Hidden Markov models for monitoring circadian rhythmicity in telemetric activity data. *Journal of The Royal Society Interface* **15** (139), 20170885.
- HUERTA, G. & WEST, M. 1999 Priors and component structures in autoregressive time series. *Journal of the Royal Statistical Society (Ser. B)* **61**, 881–899.
- HURD, W. J., MORROW, M. M. & KAUFMAN, K. R. 2013 Tri-axial accelerometer analysis techniques for evaluating functional use of the extremities. *Journal of Electromyography and Kinesiology* **23** (4), 924–929.
- KAFALI, Ö., SCHAECHTLE, U. & STATHIS, K. 2014 HYDRA: a HYbrid Diagnosis and monitoRing Architecture for diabetes. In *e-Health Networking, Applications and Services (Healthcom), 2014 IEEE 16th International Conference on*, pp. 531–536. IEEE.
- KALMAN, R. E. 1960 A new approach to linear filtering and prediction problems .
- LAI, Y., GOLIGHTLY, A. & BOYS, R. J. 2020 Sequential Bayesian inference for spatio-temporal models of temperature and humidity data. *Journal of Computational Science* **43**, 101125.
- LAURITZEN, S. L. 1996 *Graphical models*, , vol. 17. Clarendon Press.
- LEAL, Y., GARCIA-GABIN, W., BONDIA, J., ESTEVE, E., RICART, W., FERNÁNDEZ-REAL, J.-M. & VEHI, J. 2010 Real-time glucose estimation algorithm for continuous glucose monitoring using autoregressive models. *Journal of diabetes science and technology* **4** (2), 391–403.
- MARTINO, S. & RIEBLER, A. 2014 Integrated nested Laplace approximations (INLA). *Wiley StatsRef: Statistics Reference Online* pp. 1–19.
- MCCARTHY, M., EDWARDSON, C. L., DAVIES, M. J., HENSON, J., BODICOAT, D. H., KHUNTI, K., DUNSTAN, D. W., KING, J. A. & YATES, T. 2017 Fitness moderates

- glycemic responses to sitting and light activity breaks. *Medicine and science in sports and exercise* **49** (11), 2216–2222.
- MCCARTHY, M., EDWARDSON, C. L., DAVIES, M. J., HENSON, J., ROWLANDS, A., KING, J., BODICOAT, D. H., KHUNTI, K. & YATES, T. 2017 Breaking up sedentary time with seated upper body activity can regulate metabolic health in obese high risk adults: a randomised crossover trial. *Diabetes, Obesity and Metabolism* .
- MILLER, M. & STRANGE, P. 2007 Use of Fourier models for analysis and interpretation of continuous glucose monitoring glucose profiles. *Journal of diabetes science and technology* **1** (5), 630–638.
- MONTASER, E., DÍEZ, J.-L. & BONDIA, J. 2017 Stochastic seasonal models for glucose prediction in the artificial pancreas. *Journal of diabetes science and technology* **11** (6), 1124–1131.
- NELSON, K. M., REIBER, G. & BOYKO, E. J. 2002 Diet and exercise among adults with type 2 diabetes: findings from the third national health and nutrition examination survey (NHANES III). *Diabetes care* **25** (10), 1722–1728.
- OSHMANN, Y. & BAR-ITZHACK, I. Y. 1986 Square root filtering via covariance and information eigenfactors. *Automatica* **22** (5), 599–604.
- PARKES, J. L., SLATIN, S. L., PARDO, S. & GINSBERG, B. H. 2000 A new consensus error grid to evaluate the clinical significance of inaccuracies in the measurement of blood glucose. *Diabetes care* **23** (8), 1143–1148.
- PETRIS, G. 2010 An R package for dynamic linear models. *Journal of Statistical Software* **36** (12), 1–16.
- PETRIS, G., PETRONE, S. & CAMPAGNOLI, P. 2009 *Dynamic Linear Models with R*. Springer.
- RABINER, L. R. 1989 A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE* **77** (2), 257–286.
- REIFMAN, J., RAJARAMAN, S., GRIBOK, A. & WARD, W. K. 2007 Predictive monitoring for improved management of glucose levels. *Journal of diabetes science and technology* **1** (4), 478–486.
- SHENOI, B. A. 2005 *Introduction to digital signal processing and filter design*. John Wiley & Sons.

-
- DA SILVA, I. C., VAN HEES, V. T., RAMIRES, V. V., KNUTH, A. G., BIELEMANN, R. M., EKELUND, U., BRAGE, S. & HALLAL, P. C. 2014 Physical activity levels in three Brazilian birth cohorts as assessed with raw triaxial wrist accelerometry. *International journal of epidemiology* **43** (6), 1959–1968.
- SPARACINO, G., ZANDERIGO, F., CORAZZA, S., MARAN, A., FACCHINETTI, A. & COBELLI, C. 2007 Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series. *IEEE Transactions on biomedical engineering* **54** (5), 931–937.
- VIGERSKY, R. & SHRIVASTAV, M. 2017 Role of continuous glucose monitoring for type 2 in diabetes management and research. *Journal of Diabetes and its Complications* **31** (1), 280–287.
- WANG, L., LIBERT, G. & MANNEBACK, P. 1992 Kalman filter algorithm based on singular value decomposition. In *[1992] Proceedings of the 31st IEEE Conference on Decision and Control*, pp. 1224–1229. IEEE.
- WEST, M. & HARRISON, J. 2006 *Bayesian forecasting and dynamic models*. Springer Science & Business Media.
- WOODBURY, M. A. 1950 Inverting modified matrices. *Memorandum report* **42** (106), 336.
- ZANDERIGO, F., SPARACINO, G., KOVATCHEV, B. & COBELLI, C. 2007 Glucose prediction algorithms from continuous monitoring data: assessment of accuracy via continuous glucose error-grid analysis. *Journal of diabetes science and technology* **1** (5), 645–651.
- ZHAO, N. 2010 Full-featured pedometer design realized with 3-axis digital accelerometer. *Analog Dialogue* **44** (06), 1–5.