Bayesain Project

Ammar Hasan

18 February 2019

Introduction

This report summaries the Bayesian analysis of the Reisby data-set. The Reisby data-set is based on a 4 week (first placebo) psychiatric study which investigates response of depressed patients to IMI. The Bayesian analysis will investigate how the drug affects depression via the Hamilton Depression Score.

Data Setup and Cleaning

Data

The data is loaded from a .Rdata file containing the Riesby data-set introduced in the introduction.

Data Exploration

Correlation (Pairs Plot)

	hd	week	lnimi	lndmi	female	reactive_depression
hd	1.0000000	-0.3322674	-0.1165766	-0.3521130	-0.0624613	-0.0561811
week	-0.3322674	1.0000000	0.0357592	0.1239101	-0.0118326	0.0325663
lnimi	-0.1165766	0.0357592	1.0000000	0.2102979	0.0859277	-0.0365988
lndmi	-0.3521130	0.1239101	0.2102979	1.0000000	0.0945810	-0.1001591
female	-0.0624613	-0.0118326	0.0859277	0.0945810	1.0000000	0.1158473
reactive_depression	-0.0561811	0.0325663	-0.0365988	-0.1001591	0.1158473	1.0000000

The strongest correlations are negative and weak-moderate, and occur between Hamilton index with week and DMI, and rather obvious as the increase in blood concentration of the antidepressant would alleviate depression over weeks of treatment. Most other correlations are weak to very weak.
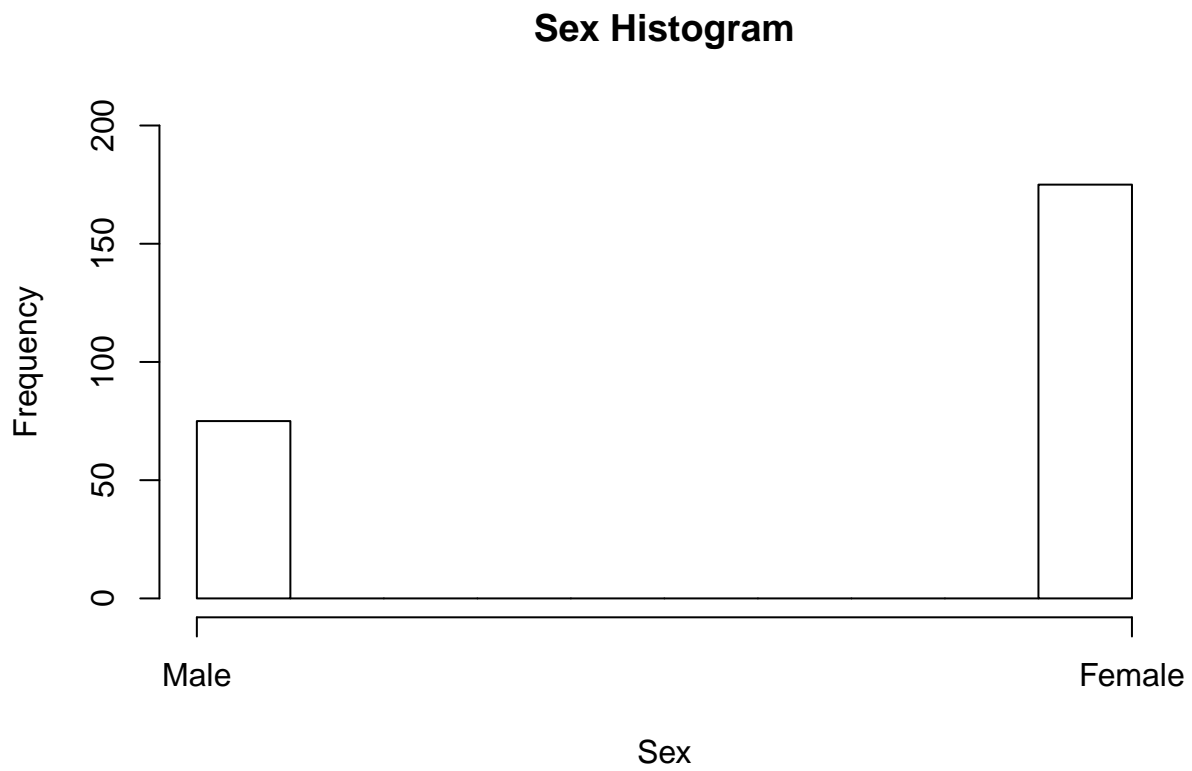
Graphical Summaries

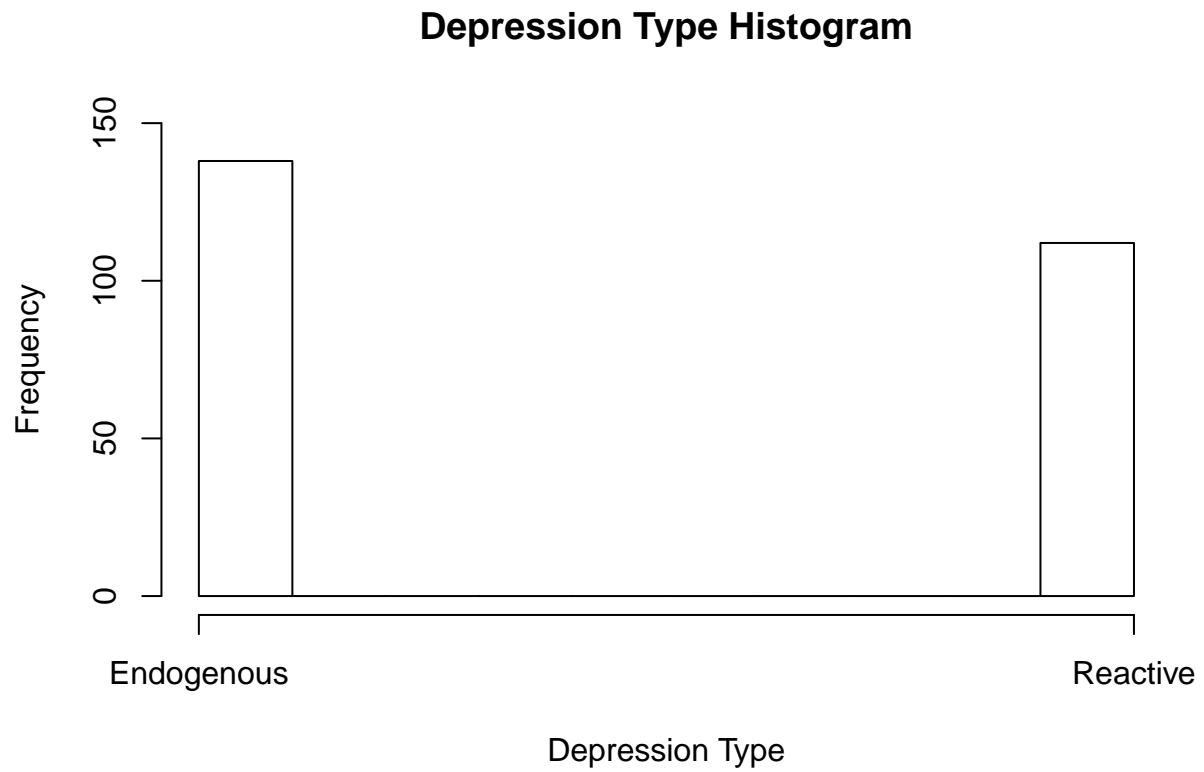**Hamilton Scores**

Hamilton Index Scores Histogram
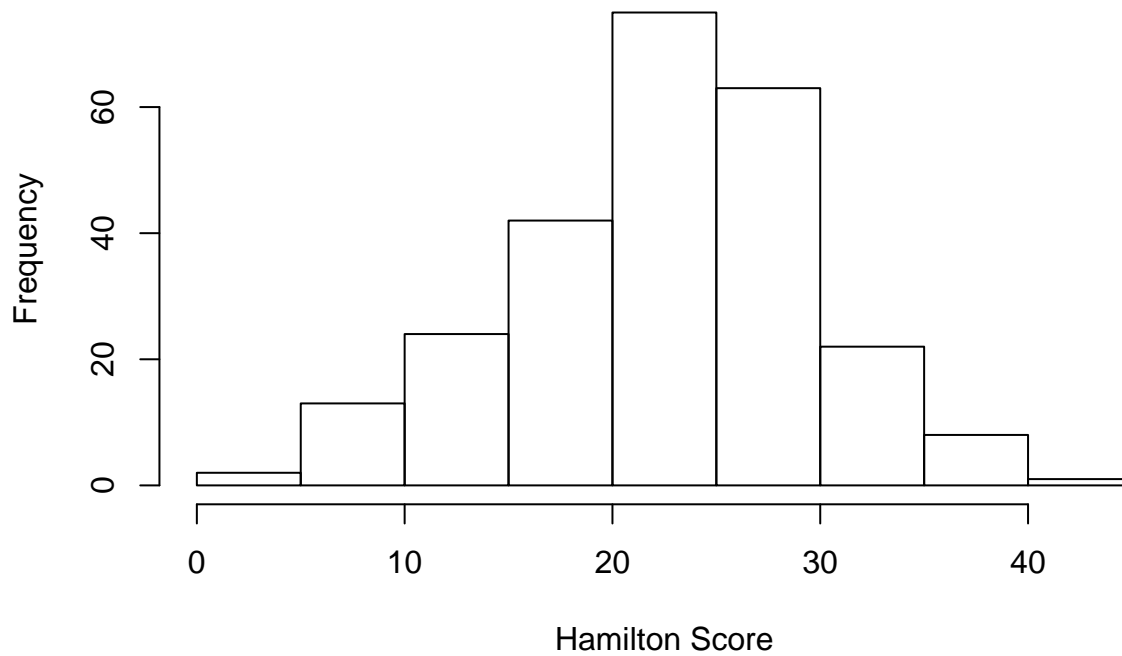
Most Hamilton scores are above 20 (i.e. moderate and severe depression dominates against mild/normal).

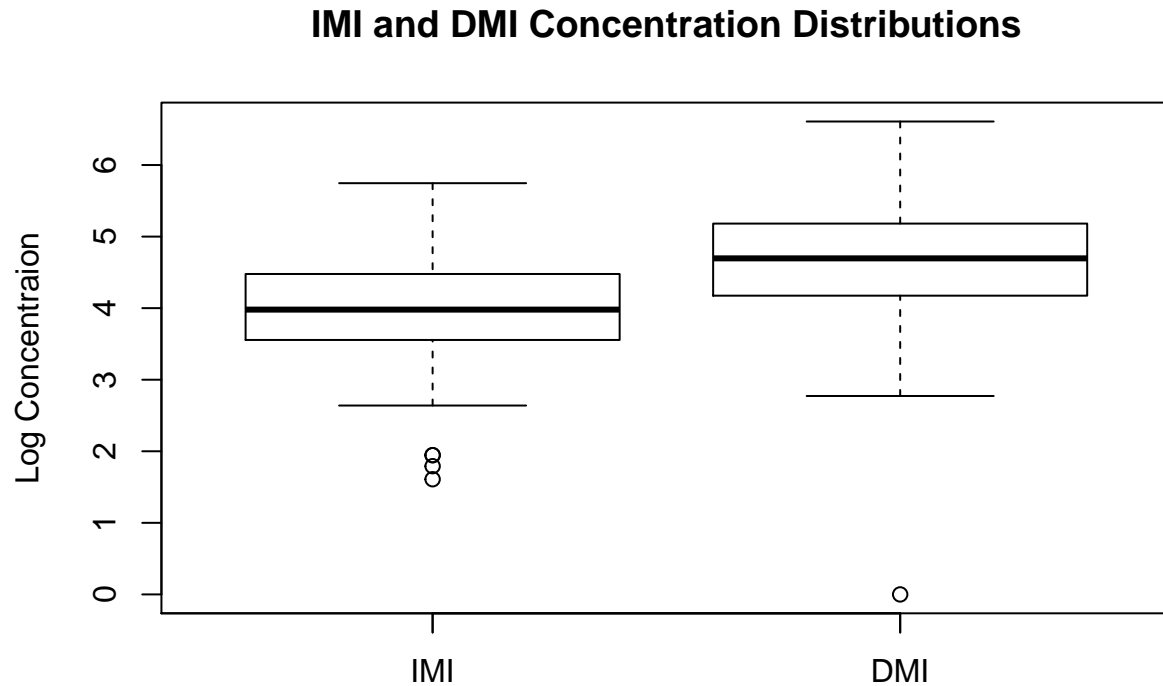## **Sex Histogram**

The female test subjects are overwhelmingly higher than the males (nearly double!).

Depression Type Histogram

Most Depression cases in the test population are Endogenous (i.e. not a reaction to an environmental event).

IMI and DMI Concentration Distributions

Generally tight distribution *especially around 25% to 75%) for both with little outliers. Both distributions look nearly identical sans the shift up with DMI. It seems that after being processed as DMI, the concentration of the antidepressant in the blood increases. This similar shape is no surprise considering that DMI is the processed form of IMI and hence they interact with one another.

Models

General Diagnostic notes

For diagnostics, all models start with 1000 samples with no thinning and no burn in discard. The models are then diagnosed using the following techniques/measurements:

- Trace plots: Used to observe general mixing via parameter values. Aim to have random caterpillar like shape or the traces.
- Effective size: Used to observe correlation effect on data by checking how effective the samples are compared to independent data. Aim to have them consistently close and somewhere near half the samples.
- Gelman Rubin: Used to check convergence of model by comparing MCMC chains, aim to have values of 1 or very close to 1 (less than 1.2 or ideally less than 1.05).
- Auto-correlation plots: Useful to check need for thinning and the effect of thinning by checking correlations between sample, lags presented here should remain consistently low.

Linear Regression

In this method a relationship between a predictor x and a response y is established through a Linear Function with a coefficients for the predictors ($b < 0$s) and an intercept ($b0$).

Data Cleaning and Preparation

Weeks

Need to have weeks represented as a predictor that indicates on whether it is a placebo week or not for linear regression since this is not a time series (no dependence between results of weeks possible here!).

Standardisation and Predictor Separation

The response (Hamilton index) and predictors (everything else) are separated for the modelling stage. All non indicator predictors (IMI and DMI) are standardised (subtracted by mean divide by standard deviation) to ensure that predictor impact is not decided by their measurement scale.

Basic Multiple

Multiple Linear Regression is a simple extension of Linear regression where multiple predictors (each with their own coefficients) are introduced.

Modelling and Diagnostics

The model was constructed in JAGS using a multiple linear regression based on a normal distribution using uninformative priors gamma(0.001, 0.001) for precision and $N(0, 10^6)$ for coefficients. Truncation to (0,60) was considered for The final model is (range for the Hamilton scores), but was dropped due to performance issues and minor DIC improvement.

For the diagnosis, initially the trace plot showed good caterpillar like shape and the Gelman-Rubin scores were close to 1 (highest 1.04 for upper b0), but the sample sizes of $b1$ and $b0$ were poor (around 600 compared to 1400-3800 for other parameters), and their auto correlation plots shows significant lag till around 7-8. A burn in discard of a 1000 and a thin of 8 seemed to have fixed the problem (no significant initial lag and sample sizes around 3600-4500).

Posterior Summaries (Highest Interval Densities)

Table 2: Multiple Linear Model Posterior Summary

	lower	upper
b[1]	-2.1200027	1.6339644
b[2]	-2.9141154	0.6353164
b[3]	1.5602314	5.1699172
b[4]	-0.9898316	0.6401728
b[5]	-3.0522702	-1.3907497
b0	21.1708578	24.5728039
sd	6.0194002	7.1947908

The highest intervals for most coefficient seems rather wide (makes it difficult to make comments!), which could be due to factors like uncertainty about their effect in the model and/or due to limited data samples. Nonetheless, the results show that $b5$ shows the most negative effect on the score (more IMI in blood makes you less depressed) and $b3$ shows the most positive effect on the score (being in a placebo week makes you more depressed). The Standard Deviations are relatively wide around 6-7, and a $b0$ of around 21-25 suggests that if all the impacts of other parameters adds up to zero the expected score is that.

DIC

Table 3: DIC for Multiple Linear Model

Deviation	Penalty	Penalised Dev.
1654	6.97	1661

The DIC score on its own is hard to interpret, so it is better to first take a look at the DIC scores of the other models first.

Term Interactions

It is possible to extend Linear Models with term interactions. Term interactions are non-linear terms that extend linear model via relationships between variables (usually decided by a coefficient c). A potential interaction to add is between IMI and DMI since these two predictors seem to interact, which is not surprising considering that DMI is the processed form of IMI.

Modelling and Diagnostics (IMI and DMI)

The model was constructed in JAGS similarly to the previous multiple linear model, but a new prior for the coefficient c controlling the effect of the interaction is introduced with a uninformative prior of $N(0, 10^6)$.

When it comes to the diagnostics, the trace plots here showed burn in issues with initial values changing rapidly suggesting that perhaps the effect of coefficient c is poorly understood initially and some burn in is needed. However, Other diagnostics showed similar results to the previous model, as Gelam showed convergence, small $b0$ and $b1$ effective sizes were observed (around 500) and lastly lag of up to 9 was observed in the auto-correlation plots with $b0$ and $b1$ being the worst. Nonetheless, the addition of a burn in discard of 2000 and thinning value of 9, the issues with the trace plots were gone and the effective sizes were now similar to the previous model being around 3800-4500.

Posterior Summaries (IMI and DMI Highest Interval Densities)

Table 4: Linear Model with IMI and DMI Interaction Terms Posterior Summary

	lower	upper
b[1]	-2.0230770	1.6565737
b[2]	-2.7637925	0.5377701
b[3]	1.5951818	5.3595293
b[4]	-1.1327336	0.7015442
b[5]	-3.2074611	-1.4837894
b0	21.2413041	24.7043294
c	-0.9916844	1.0223646
sd	6.0492315	7.2426883

The posterior summaries seem similar in fact near identical, with wide coefficients, strong negative $b5$ and strong positive $b3$, a similar sd and $b0$. The effect of C seems to range from around -0.90 to 1.0 suggesting a varied effect based on individual.

Comparison with Multiple Linear Regression (IMI and DMI DIC)

Table 5: DIC for Multiple Linear Model With IMI and DMI Interaction Terms

Deviation	Penalty	Penalised Dev.
1655	8.21	1663

Unsurprisingly, the similarities in the posterior translated to similar DIC results which showed penalized deviance that is within room for error. It seems the interaction did not cause any significant differences to the model! This is perhaps because a strong interaction effect between IMI and DMI is not supported by

the available data. The reason for this is perhaps because of the weak correlation between the values (seen in exploration), and perhaps also the variable and difficult to predict effect (seen in c posterior summary).

Modelling and Diagnostics (Female & Reactive Depression)

Since the other interaction did not yield any strong results another one is tried. The only remaining obvious interaction is between sex and depression type, perhaps sex plays a role on most likely type of depression?

The model is practical identical to the previous interaction model bar the variables being used for the interaction, hence the same tuning is carried over and verified. Some additional thinning (24) was needed according to small c and $b0$ sample sizes (around 1800 for c and 2100 for $b0$) and lag up around 4 for both which remained for a while even with high thinning,

Posterior Summaries (Female & Reactive Depression Highest Interval Densities)

Table 6: Linear Model with Interaction Terms (Female & Reactive)
Posterior Summary

	lower	upper
b[1]	-5.5549358	-0.6791937
b[2]	-8.9850207	-2.7975939
b[3]	1.5094955	5.3137504
b[4]	-0.6820216	0.9968610
b[5]	-2.9732179	-1.2822302
b0	22.8246086	26.8181643
c	3.1027652	10.6013633
sd	5.9196330	7.0807085

The most obvious change compared to the other interaction model is the different c parameter value, which is significantly higher with a positive value suggesting a positive relationship. Some of the other parameters changed as well, as for instance the coefficients for sex and reactive depression have a strong negative impact now (females and reactive depression type patients more depressed).

Comparison with Multiple Linear Regression (Female & Reactive Depression DIC)

Table 7: DIC for Linear Model with interaction terms (Female & Reactive)

Deviation	Penalty	Penalised Dev.
1642	7.93	1650

Unfortunately, even though this model seems to perform better than the previous two, it seems that again the improvement is not significant, and hence the evidence that this model or its assumptions are correct is not too strong. Again, this could be either because this interaction is not supported by the data.

Hierarchical

Another possible extension to linear models is via hierarchies. This is done by considering that the data consists of groups which have different parameters for their distributions. In this example we can consider

that every individual is a different group (each person is different), which according to the wide range of scores observed in the exploration stage and the uncertainty in the coefficients seen in the previous models might be a reasonable assumption.

ID Preparation

To make accessing subjects to find their groups easier the IDs are converted to $(1, 2, 3 \ldots)$ counting form using factors and levels.

Modelling and Diagnostics

Again, the model was constructed in JAGS similarly to the simple multiple linear model, but with new priors for the c parameters controlling the effect of the groups is introduced with a uninformative prior of $N(0, gamma(0.001, 0.001))$.

For the diagnosis, the trace plots showed similar behavior to the previous interaction models, with most parameters changing rapidly initially suggesting a need for burn in and a misunderstood effect for the c parameters. When it comes to the other diagnosis methods, they seem to show significant issues with $b1$, $b2$ and $b0$ with tiny effective size (around 40-90!), poor convergence (uppers of 1.19 for $b1$, 1.09 for $b2$ and 1.21 for $b0$) and lastly significant lag (around 20-26 for most b parameters including $b1$, $b2$ and $b0$).

Hence, to resolve these issues, a burn in discard of 3000, a thinning value of 40 and an increase of samples sizes to 2000 were introduced. This resolved most of the issues as the initial lag in the trace plots was one or two at worst and effective sizes were now more balanced around 5500-8000.

Posterior Summaries (Highest Interval Densities)

	lower	upper
b[1]	-3.457485	2.5799358
b[2]	-4.007742	1.6636275
b[3]	2.209231	4.6280738
b[4]	-1.017813	1.3409082
b[5]	-3.093320	-0.7602051
b0	20.080787	25.5019191

While again similar to previous models, these results show some noticeable changes with some parameters, as most parameters are now wider except $b5$ (IMI drug) and $b3$ (placebo) which are narrower (especially at the lower and upper level respectively). This perhaps suggests that a lower effect of the drug a more varied effect due to other factors compared to other models.

Comparison with Multiple Linear Regression with Terms (DIC)

Table 9: DIC for Hierarchical Multiple Linear Model

Deviation	Penalty	Penalised Dev.
1440	60.08	1500

This model seems to do noticeably better than the previous two with a lower penalized deviance of around 1500 compared to the previous two of around 1660. This proves that each person acts as a different group with different distribution behavior which explains the uncertainty.

AR(1)

AR models are time series models (they allow dependence on responses of earlier times) that are based on a linear like model that predicts using a given number of previous responses with the help of a<0 coefficients to control these previous responses effect. AR models also have another a0 coefficient usually referred to as a mean *mu* which controls the convergence point. In this example, AR models would suggest that scores from previous weeks effect other weeks, which would not be a difficult assumption to make since the depression severity of a previous week is likely to impact the next as seen in the correlation matrix in the exploration stage.

Since we are only dealing with 4 weeks, a AR model which uses one previous response only is a reasonable choice since there are not many previous responses to choose from!

Data Cleaning and Perparation (Subject Matricies)

However, a big problem with implementing AR models here arises because we need to use every subject previous response not any previous response. And, as shown with the performance of the hierarchical model the individuals of the study have a strong level of independence and using any other previous response would seriously negatively impact the model.

Hence, to ensure that we only use previous responses of the same subjects the Hamilton scores are stored in a matrix of rows of subject IDs and columns of weeks which will be iterated through using a nested loop. To also ensure that other predictors will be available during the nested loop, the other predictors are also stored in similar matrices.

However, since not all test subjects have results for all weeks we can only use a portion of the subject (52 instead of 66), this may effect results.

Modelling and Diagnostics

The AR(1) model is constructed in JAGS using an a1 coefficient based on a normal prior of $N(0, 0.35^2)$ to ensure it avoids values 1.0 and -1.0 which caused walking issues during diagnosis. The *mu* parameter is based on weakly informative a prior of $N(0, 10^6)$ that is truncated to (0,60), the use of the informative prior is based on the Hamilton score ranges and was done because mu had a very wide range with a full uninformative prior. The precision of the AR model is using an uninformative prior of gamma(0.001, 0.001). Finally, hierarchical behavior is also introduced here since it proved to improve the model behavior, the hierarchical c coefficient will be using an uninformative prior of $N(0, dgamma(0.001, 0.001)^-2)$ for every group as before.

During diagnosis, the model displayed similar behavior to the previous model, with its parameters (especially a1 and hierarchical standard deviation) showing initial rapid changes which shows a burn in discard requirement. But, it seems most of the convergence and correlation issues moved to the hierarchical standard deviation with very high Gelman-Rubin upper interval of 1.75, very poor effective size around 50 and significant lag till around 90. Hence, a significant thinning of 100 was introduced with increased sample sizes of 3000 which made it converge and fixed most of the correlation issues, but it still had initial lag of around 5 at worst and poor effective sizes of around 5000 vs 12000 for other parameters.

Posterior Summaries (Highest Interval Densities)

Table 10: AR(1) Posterior Summaries

	lower	upper
a1	0.6951395	0.9137709
b[1]	-1.1262641	0.4386760

	lower	upper
b[2]	-1.5803522	0.1324573
b[3]	-1.1682670	2.0928328
b[4]	-1.7931349	1.3140888
mu	1.1895676	20.6455152
sd	4.1106244	5.1922584
sd_hier	0.0148008	1.2538716

An a1 parameter of around 0.7-0.91 suggests a decaying random walk converging to a given value. For the other parameters it seems that the mu parameter is quite wide, suggesting that it is difficult to come to a consensus about the average score.

When it comes to the b coefficients, their values seemed to have narrowed down alongside the standard deviation. Interestingly, this model thinks the strongest negative impact on score can be the type of depression $b4$ (being reactive), but that it can also be positively impacting as well (i.e. people who have reactive depression can be more depressed but also less depressed than non-reactive depression patients depending on the person). Also, surprisingly, the strongest positively impacting coefficient is now the sex, with female being more depressed (but sometimes this can be the other way around!).

When it comes to the impact of the drug itself, this model suggests it is mostly negative according to $b1$ and $b2$ (IMI and DMI), but not as strong as other models suggested it is!

Comparison with Hierarchical Linear Regression (DIC)

Table 11: DIC for AR(1) Model

Deviation	Penalty	Penalised Dev.
922	8.77	931

This model seems to perform significantly better than all the other models introduced here! With DIC penalized scores around 930 compared to the other models (around 1500-1660). This not only shows that there is indeed evidence of dependence between results of different weeks, but also that the assumptions made on the last section based on the summaries might also be true!

However, this is assuming that the sub-setting of the data-set to only include the patients with full records every week did not impact the model!

Conclusion

In conclusion, when it comes to modelling the problem itself it seems that the AR(1) model with hierarchies performed best according to its DIC scores, and hence it is the selected model to represent the problem. The choice model suggested that there is dependence between the depression score and weeks, and that each person acts as a unique group when it comes to their depression behavior.

The summaries of the chosen model and the others suggested that the IMI drug reduces the patients depression scores, however other factors like the type of depression and sex can have a strong impact as well depending on the individuals.

However, some issues there were some issues with the analysis and the data. The exploration stage showed that the sexes are not well represented in the data, moreover as seen during the AR(1) model construction not all patients had records for all weeks, hence the AR(1) model only considered a subset of the data which may have effected results.

In the future, it might be wise to try this analysis with a different data-set, perhaps ones with a different sex distribution or even more weeks to try a different AR model with more dependent previous results. Moreover, it might also be wise to try a multi-class classification method to ensure models only predict at exact classes.

Appendix

Abbrevations

IMI - antidepressant drug imipramine

DMI - desmethylimipramlne (Processed IMI)

AR - Auto Regressive Models

AR(1) - Auto Regressive Models (1st Degree)

JAGS - Just Another Gibbs Sampler MCMC based Bayesian sampler

MCMC - Markov chain Monte Carlo

DIC - Deviance Information Criterion

Functions Code

DIC Table

```r
# Creates a table for DIC with a given caption
dic_table <- function(dic, caption) {

  # Find values
  mean_dev = round(c(sum(dic$deviance)))
  pen = round(c(sum(dic$penalty)), 2)
  mean_pen_dev = round(c(mean_dev+pen))

  # Create DF
  dic.data = data.frame(mean_dev, pen, mean_pen_dev)

  # Print DF using kable
  kable(dic.data,
        col.names = c("Deviation", "Penalty", "Penalised Dev."),
        caption = caption)
}
```

Data and Exploration Code

Data Loading

```r
load("Reisby.RData")
Reisby = as.data.frame(Reisby)
```

Correlation (Pairs Plot)

```r
kable(cor(Reisby[,-1])) # no id
```

Hamilton Scores Histogram

```r
hist(Reisby$hd,
     main = "Hamilton Index Scores Histogram",
     xlab = "Hamilton Score")
```

Sex Histogram

```r
# Replace x axis here with male/female axis

hist(Reisby$female, main = "Sex Histogram",
     xlab = "Sex", xaxt = "n", ylim = c(0, 200))
axis(1, at=0:1, labels=c("Male","Female"))
```

Depression Type Histogram

```r
hist(Reisby$reactive_depression,
     main = "Depression Type Histogram",
     xlab = "Depression Type", xaxt = "n", ylim = c(0, 150))
axis(1, at=0:1, labels=c("Endogenous","Reactive"))
```

DMI and IMI Boxplot

```r
boxplot(Reisby$lnimi, Reisby$lndmi,
        main = "IMI and DMI Concentration Distributions",
        ylab = "Log Concentraion")
# uses 3 labels due to strange bug
axis(1, at=0:2, labels=c("IMI","IMI", "DMI"))
```

Multiple Linear Model Code

Weeks Cleaining

```
placebo = c(ifelse(Reisby$week == 0, 1, 0))
```

Predictor Separation

```
# Separate predictors (x) from response (y)
x = Reisby[,-2:-3]
x$placebo = placebo
y = Reisby[,2]
```

Standarisation

```
# scale IMI and DMI

scaled_dmi = (Reisby$lnimi -
                mean(Reisby$lnimi))/sd(Reisby$lnimi)
scaled_imi = (Reisby$lndmi -
                mean(Reisby$lndmi))/sd(Reisby$lndmi)

# Introduce scaled predictors to other predictors

lin_x_scaled = x[, -2:-3]

lin_x_scaled$scaled_dmi = scaled_dmi
lin_x_scaled$scaled_imi = scaled_imi
```

Model

```
# Parameters to pass to JAGS

data = list(x = lin_x_scaled[,-1], y = y,
            n = nrow(lin_x_scaled),
            p = ncol(lin_x_scaled[,-1]))

# JAGS model (as a string)

model_string = "
model {

  # Coefficients

  b0 ~ dnorm(0, 1E-6)

  for (j in 1:p) {
    b[j] ~ dnorm(0, 1E-6)
  }

  # Model
```

```
  tau ~ dgamma(0.001, 0.001)
  sd = pow(tau, -0.5)

  for (i in 1:n) {
    y[i] ~ dnorm(mu[i], tau)
    mu[i] = b0 + inprod(b, x[i,])
  }

}
"

# model compilation and sampling

thin = 8
iter = 1000

model_tc = textConnection(model_string)
model = jags.model(model_tc, data = data, n.chains = 4)

update(model, n.iter = iter)

lin_samples = coda.samples(model,
                           variable.names = c("b0","sd","b"),
                           n.iter = iter * thin,
                           thin = thin)

lin_dic = dic.samples(model,
                      n.iter = iter * thin,
                      thin = thin)
```

Summary

```
hpd = HPDinterval(lin_samples)
kable(hpd[[1]],
      caption = "Multiple Linear Model Posterior Summary")
```

DIC

```
dic_table(lin_dic, "DIC for Multiple Linear Model")
```

Linear Term Interactions Code

Model (IMI & DMI)

```r
# Parameters to pass to JAGS

data = list(x = lin_x_scaled[,-1], y = y,
            n = nrow(lin_x_scaled),
            p = ncol(lin_x_scaled[,-1]))

# JAGS model (as a string)

model_string = "
model {

  # Coefficients and interaction

  b0 ~ dnorm(0, 1E-6)
  c ~ dnorm(0, 1E-6)

  for (j in 1:p) {
    b[j] ~ dnorm(0, 1E-6)
  }

  # Model

  tau ~ dgamma(0.001, 0.001)
  sd = pow(tau, -0.5)

  for (i in 1:n) {
    y[i] ~ dnorm(mu[i], tau)
    mu[i] = b0 + inprod(b, x[i,]) + c * x[i, 4] * x[i, 5]
  }

}
"

# model compilation and sampling

thin = 9
iter = 1000

model_tc = textConnection(model_string)
model = jags.model(model_tc, data = data, n.chains = 4)

update(model, n.iter = iter*2)

terms_samples = coda.samples(model,
                             variable.names = c("b0","sd","b", "c"),
                             n.iter = iter * thin, thin = thin)

terms_dic = dic.samples(model, n.iter = iter * thin, thin = thin)
```

Summary (IMI & DMI)

```
hpd = HPDinterval(terms_samples)
kable(hpd[[1]],
      caption =
      "Linear Model with IMI and DMI
      Interaction Terms Posterior Summary")
```

DIC (IMI & DMI)

```
dic_table(terms_dic,
          "DIC for Multiple Linear Model
          With IMI and DMI Interaction Terms")
```

Model (Reactive & Female)

```
# Parameters to pass to JAGS

data = list(x = lin_x_scaled[,-1], y = y,
            n = nrow(lin_x_scaled),
            p = ncol(lin_x_scaled[,-1]))

# JAGS model (as a string)

model_string = "
model {

  # Coefficients and interaction

  b0 ~ dnorm(0, 1E-6)
  c ~ dnorm(0, 1E-6)

  for (j in 1:p) {
    b[j] ~ dnorm(0, 1E-6)
  }

  # Model

  tau ~ dgamma(0.001, 0.001)
  sd = pow(tau, -0.5)

  for (i in 1:n) {
    y[i] ~ dnorm(mu[i], tau)
    mu[i] = b0 + inprod(b, x[i,]) + c * x[i, 1] * x[i, 2]
  }

}
"

# model compilation and sampling

thin = 24
```

```
iter = 1000

model_tc = textConnection(model_string)
model = jags.model(model_tc, data = data, n.chains = 4)

update(model, n.iter = iter*2)

terms2_samples = coda.samples(model,
                              variable.names = c("b0","sd","b", "c"),
                              n.iter = iter * thin, thin = thin)

terms2_dic = dic.samples(model, n.iter = iter * thin, thin = thin)
```

Summary (Reactive & Female)

```
hpd = HPDinterval(terms2_samples)
kable(hpd[[1]],
      caption =
      "Linear Model with Interaction Terms
      (Female & Reactive) Posterior Summary")
```

DIC (Reactive & Female)

```
dic_table(terms2_dic, "DIC for Linear Model with interaction terms
          (Female & Reactive)")
```

Linear Hierarchical Code

ID Perparation

```r
# Convert ids to (1,2,3 ..) form

h_lin_x_scaled = lin_x_scaled
h_lin_x_scaled$id = as.factor(h_lin_x_scaled$id)
levels(h_lin_x_scaled$id) =
  1:length(levels(h_lin_x_scaled$id))
```

Model

```r
# Parameters to pass to JAGS

data = list(x = h_lin_x_scaled[,-1], y = y,
            n = nrow(h_lin_x_scaled),
            p = ncol(h_lin_x_scaled[,-1]),
            n_subjects = length(unique(h_lin_x_scaled[,1])),
            subject = h_lin_x_scaled$id)

# JAGS model (as a string)

model_string = "
model {

  # Coefficients

  b0 ~ dnorm(0, 1E-6)

  for (j in 1:p) {
    b[j] ~ dnorm(0, 1E-6)
  }

  # Hierarchies

  for (k in 1:n_subjects) {
    c[k] ~ dnorm(0, tau_hier)
  }

  tau_hier ~ dgamma(0.001, 0.001)
  sd_hier = pow(tau_hier, -0.5)

  # Model

  tau ~ dgamma(0.001, 0.001)
  sd = pow(tau, -0.5)

  for (i in 1:n) {
    y[i] ~ dnorm(mu[i], tau)
    mu[i] = b0 + inprod(b, x[i,]) + c[subject[i]]
  }
```

```r
}
"

# model construction and sampling

iter = 2000
burn = 3000
thin = 40

model_tc = textConnection(model_string)
model = jags.model(model_tc, data = data, n.chains = 4)

update(model, n.iter = burn)
h_samples = coda.samples(model,
                         variable.names = c("b0","sd","b", "sd_hier"),
                         n.iter = iter * thin,
                         thin = thin)

h_dic = dic.samples(model, n.iter = iter * thin, thin = thin)
```

Summary

```r
hpd = HPDinterval(h_samples,
                  caption =
                  "Hierarchical Linear Model Posterior Summaries")
kable(hpd[[1]][-7:-72,])
```

DIC

```r
dic_table(h_dic, "DIC for Hierarchical Multiple Linear Model")
```

AR Code

Subject Matricies Construction

```r
# Scale IMI and DMI as usal

scaled_imi = (Reisby$lnimi - mean(Reisby$lnimi))/sd(Reisby$lnimi)
scaled_dmi = (Reisby$lndmi - mean(Reisby$lndmi))/sd(Reisby$lndmi)

SReisby = Reisby[, -4:-5]
SReisby$scaled_dmi = scaled_dmi
SReisby$scaled_imi = scaled_imi

# Matrix construction

# Only use full weeks

weekCount = count(SReisby, "id")
full_ids = subset(weekCount, freq == max(SReisby$week) + 1)
sub_SReisby = subset(SReisby, id %in% full_ids$id)
ids = unique(sub_SReisby$id)

# Construct them to be ids * weeks

score_mat = matrix(nrow = length(ids),
                   ncol = max(sub_SReisby$week) + 1)
imi_mat = matrix(nrow = length(ids),
                 ncol = max(sub_SReisby$week) + 1)
dmi_mat = matrix(nrow = length(ids),
                 ncol = max(sub_SReisby$week) + 1)
female_mat = matrix(nrow = length(ids),
                    ncol = max(sub_SReisby$week) + 1)
dep_mat = matrix(nrow = length(ids),
                 ncol = max(sub_SReisby$week) + 1)

# Use nested loop to fill

for(i in 1:length(ids)){
  for(j in 0:max(sub_SReisby$week)){
    score_mat[i, j+1] = subset(sub_SReisby, id == ids[i] & week == j)$hd
    imi_mat[i, j+1] = subset(sub_SReisby, id == ids[i] & week == j)$scaled_imi
    dmi_mat[i, j+1] = subset(sub_SReisby, id == ids[i] & week == j)$scaled_dmi
    female_mat[i, j+1] = subset(sub_SReisby, id == ids[i] & week == j)$female
    dep_mat[i, j+1] = subset(sub_SReisby, id == ids[i] & week == j)$reactive_depression

  }
}
```

Model

```r
### Modelling (AR(1))

# JAGS parameters
```

```r
# Reuse Hierarchical model ids
data = list(y = score_mat, x1 = imi_mat,
            x2 = dmi_mat, x3 = female_mat,
            x4 = dep_mat, nrow = nrow(score_mat),
            ncol = ncol(score_mat), p = 4)

# JAGS model (as a string)

model_string = "
model {

  # Predictor coefficents

  for (j in 1:p) {
    b[j] ~ dnorm(0, 1E-6)
  }

  # Hierachital Groups c parameters

  for (l in 1:nrow) {
    c[l] ~ dnorm(0, tau_hier)
  }

  tau_hier ~ dgamma(0.001, 0.001)
  sd_hier = pow(tau_hier, -0.5)

  # Model (use nested loop to iterate matricies)

  for (i in 1:nrow) {
    for(k in 2:ncol){
      y[i,k] ~ dnorm (mu + a1 * (y[i, (k-1)] - mu) +
               x1[i, k] * b[1] + x2[i, k] * b[2] +
               x3[i, k] * b[3] + x4[i, k] * b[4] + c[i], tau)
    }
  }

  mu ~ dnorm(0, 1E-6) T(0,60)
  a1 ~ dnorm (0, 0.35^-2)
  tau ~ dgamma (0.001, 0.001)
  sd = pow (tau , -0.5)
}
"

thin = 100
iter = 3000

model_tc = textConnection(model_string)
model = jags.model(model_tc, data = data, n.chains = 4)

## Sampling

update(model, n.iter = iter)
```

```
ar1_samples = coda.samples(model,
                           variable.names = c("mu", "a1", "sd",
                                              "b","sd_hier"),
                           n.iter = iter * thin, thin = thin)

ar1_dic = dic.samples(model, n.iter = iter * thin, thin = thin)
```

Summary

```
hpd = HPDinterval(ar1_samples)
kable(hpd[[1]], caption = "AR(1) Posterior Summaries")
```

DIC

```
dic_table(ar1_dic, "DIC for AR(1) Model")
```