STAT 578 Bayesian Analysis and Computation

Final Group Project

Due: May 10th, 2021

Students: Yu (Rain) Huang, Tai-Te Su, Mars Xie

Instructor: Dr. Trevor Park

Project Title

Understanding the Impact of Physical Abilities on Subjective Wellbeing Among Older Adults: A Bayesian Analysis

Introduction

In the past few decades, the population has been aging rapidly especially in developed countries. Take the United States as an example. According to the data recently released by the Census Bureau (2021), the number of older adults aged 65 and older is expected to reach 89 million in 2050. With advancing age, the likelihood of developing disability could increase substantially due to age-related decline in physiological reserves and chronic health conditions (Rantakokko, Manty, & Rantanen, 2013). This issue is of great importance because studies have shown that older adults who maintain functionally independent are more likely to remain active in the community and have better overall health outcomes (Davis et al., 2015; Shafrin, Sullivan, Goldman, & Gill, 2017). In addition, impaired physical function can not only prevent older people from doing things that are meaningful, but it can also cause a tremendous amount of psychological stress to them. All of these could have a detrimental impact on their wellbeing and quality of life (Freedman et al., 2012; George, 2010).

While previous studies have found a link between higher physical function and longevity and better health outcomes, more research is still needed to investigate whether physical functions would affect older adults' psychological wellbeing. Understanding this relationship has implications for the development of health promotion programs that aim to improve wellbeing among the older population in the future. To provide more evidence and contribute to the existing body of literature, therefore, the goal of the present study was to examine the extent to which physical functions, measured by grip strength and gait speed, are associated with subjective wellbeing among older adults in the United States.

Methods

Data description

We obtained the 2012 Health and Retirement Study (HRS) data from the Institute for Social Research (ISR) Survey Research Center (SRC) at the University of Michigan. HRS is a national longitudinal study of multi-faceted statuses including health status of older Americans. We selected variables from the respondent-level physical performance measures and biomarkers for our research purposes. The variables we chose and their definitions are as follows:

- HHID: household identification number, which uniquely identifies each household in the 2012 study.
- PN: person number, which, in combination with HHID, uniquely identifies a respondent or respondent's spouse or partner(s).
- Wellbeing: life satisfaction, which represents the respondent's satisfaction level (on a 5 Likert-type scale; from 1 being not at all satisfied, to 5 being completely satisfied) of their life as a whole.
- Age: respondent's current age (in years).
- Education: respondent's years of education.
- BMI: body mass index, expressed in units of kg/m^2, is a measure of body fat based on a person's height and weight. The higher the value, the more the body fat.
- Gait speed: average results from two walking tests, which record the time (in seconds) the respondent took to walk 12 feet at their usual walking speed. The higher the value, the slower the walking speed.
- Grip: average results of two hand strength tests, which record the measurements from a dynamometer (in kilograms) when the respondent squeezing a handle for a few seconds. The higher the value, the greater the hand strength.
- Balance time: balance test result, which records the time length (in seconds) the respondent was able to stand and place the heel of one foot touching the toes of the other foot for either 30 or 60 seconds. The higher the value, the better the balance performance.

We intended to build a regression model on respondents' subjective life satisfaction ratings (wellbeing) using predictors described above. We obtained n = 2794 respondents' data after removing all the missing values from each of the explanatory variable. After further scrutinizing each variable, we decided to eliminate the variable "Balance time" since its measurements have ambiguous interpretations (by grouping measurements from 2 different tests together) and including it in the model might jeopardize our model performance and interpretations of the model results.

Model formulation

Households are natural clusters in terms of sampling, so we chose households as a grouping variable for modeling. After filtering out households with only one person, we have n = 1148 respondents out of 571 households remaining, which is adequate regarding group size. The hierarchical regression model is specified as follows:

Let $y_{ij} = r,\ \ wellbeing\ rating\ of\ respondent\ i\ \ in\ household\ j,$

$$r = 1, \dots, 5, \qquad i = 1, \dots, 1148, \qquad j = 1, \dots, 571$$

There are 5 explanatory variables (Age(v1), Education(v2), BMI(v3), Gait speed(v4), and Grip(v5)). For respondent i in household j,

$$X_{ij} = \left(x_{ij,1,}\, x_{ij,2,}\, x_{ij,3,}\ x_{ij,4,}\ x_{ij,5}\right),$$
$$i = 1, \dots, 1148, \qquad j = 1, \dots, 571$$

Since our response variable, wellbeing, is on an ordinal scale, we chose to employ a proportional odds model (or ordered logistic regression model) and specify it as follows:

$$logit\left(P\left(y_{ij} \le r \middle| X_{ij}, \theta\right)\right) = \ln\left(\frac{P\left(y_{ij} \le r \middle| X_{ij}, \theta\right)}{P\left(y_{ij} > r \middle| X_{ij}, \theta\right)}\right) = c_r - \left(\alpha_j + X_{ij}\beta\right),$$
$$r = 1, \dots, 4, \qquad i = 1, \dots, 1148, \qquad j = 1, \dots, 571$$
$$\beta = (\beta_1, \dots, \beta_5)^T,$$
$$c_1 \le c_2 \le c_3 \le c_4$$

Alternatively,

$$P\left(y_{ij} \le r \middle| X_{ij}, \theta\right) = \frac{1}{1 + e^{-[c_r - (\alpha_j + X_{ij}\beta)]}}$$

Then,

$$P\left(y_{ij} = 1\right) = P\left(y_{ij} \le 1 \middle| X_{ij}, \theta\right),$$
$$P\left(y_{ij} = r\right) = P\left(y_{ij} \le r \middle| X_{ij}, \theta\right) - P\left(y_{ij} \le r - 1 \middle| X_{ij}, \theta\right), for\ r = 2, \dots, 4,$$
$$P\left(y_{ij} = 5\right) = 1 - P\left(y_{ij} \le 4 \middle| X_{ij}, \theta\right)$$

Note that we included a separate random intercept α for each household to incorporate a (random) household effect into our model. On the other hand, the regression coefficients, $\beta_1, \dots, \beta_5$, do not vary by group, thus are treated as fixed effects in frequentist terms. The ordered parameters $c_r's$ are cutoff points that are needed for estimation of the coefficients. They are ordered because it is assumed in this model that, underlying observed y, there is a latent continuous variable y* for which y = r when $c_{r-1} \le$ y*$\le c_r$. As in random effect models, it is often useful to treat the groups as randomly sampled, so that the intercepts

$$\alpha_1, \dots, \alpha_{571}$$

are independent and from the same distribution. We considered giving them a common normal distribution with hyperparameter σ_α^2 such that:

$$\alpha_j | \, \sigma_\alpha^2 \sim iid. N(0, \sigma_\alpha^2)$$

To avoid a biased estimation of σ_α^2, we would use a noninformative prior on σ_α^2. Since σ_α^2 describes the variance of unobserved (latent) quantities, instead of using the "reciprocal" prior for σ_α^2, we would use

$$p(\sigma_\alpha^2) \propto (\sigma_\alpha^2)^{-1/2}, \;\; \sigma_\alpha^2 > 0$$

, which puts finite mass near zero.

By transformation of variables, it is equivalent to

$$p(\sigma_\alpha) \propto 1, \qquad \sigma_\alpha > 0$$

Considering the parameters $\beta \; and \; c_r's$, we would like them to be noninformative as well:

$$p(\beta_1, \beta_2, \beta_3, \beta_4, \beta_5) \propto 1$$

$$p(c_1, c_2, c_3, c_4) \propto 1$$

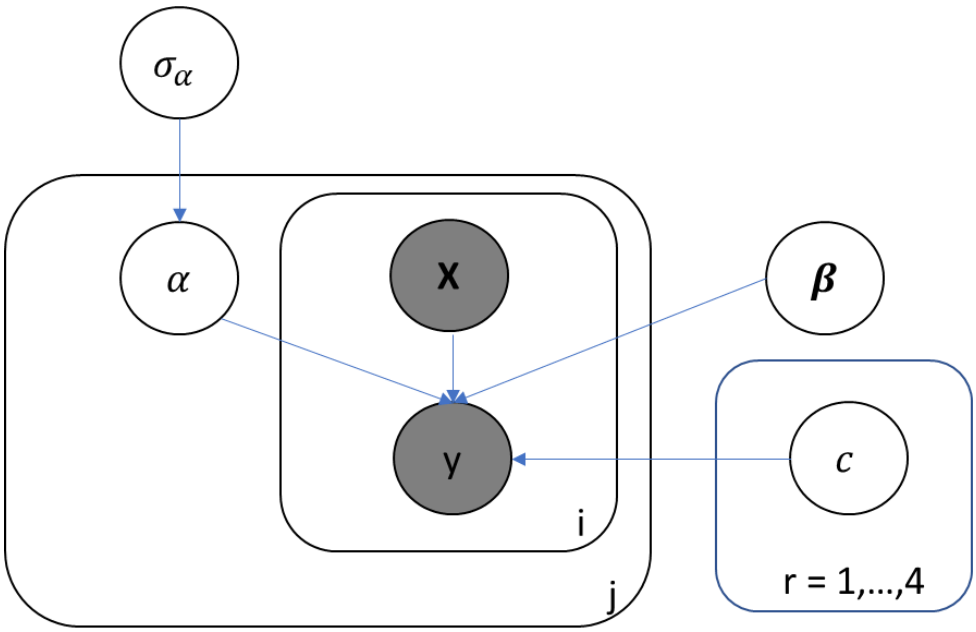
Because we chose to run the model using JAGS, we replaced the improper priors with proper but diffuse priors:

$$\beta_1, \beta_2, \beta_3, \beta_4, \beta_5 \sim iid. N(0, 100^2)$$

$$c_1, c_2, c_3, c_4 \sim iid. N(0, 10^2)$$

$$\sigma_\alpha \sim Exp(0.001)$$

Below is a Directed Acyclic Graph for our constructed model:



Model implementation

We constructed our model using JAGS (see the Appendix for the complete JAGS model code) and used R to interact with JAGS. To aid interpretation, we standardized each explanatory variable by centering and scaling it to 1 sample standard deviation. We then created a data list in R ("d1" in the R code provided in the Appendix) to pass to JAGS. We used 4 Markov Chains and initialized the c parameters to be an ordered vector; otherwise, JAGS will initialize them to the same value. Other nodes were auto-initialized.

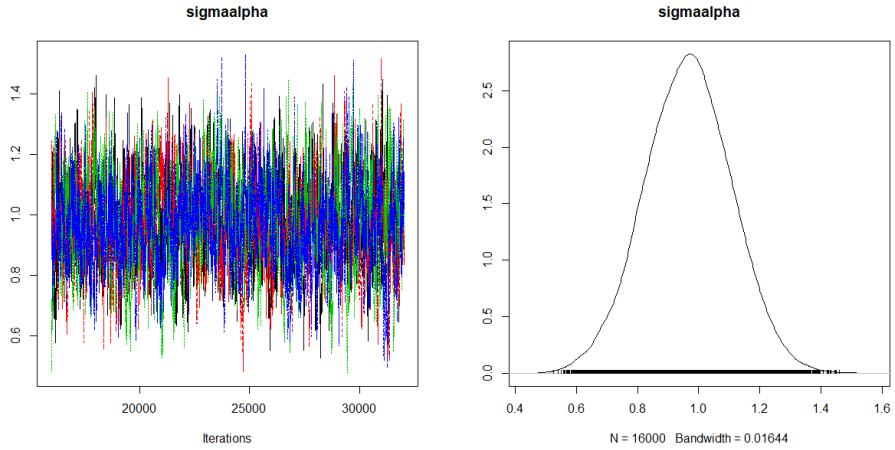
Results

Convergence diagnostics

The result below comes from our last 16,000 iterations with 4 chains. Based on our model construction above, here are our Gelman-Rubin statistics (R^2) results:

	Point estimate	Upper C.I.		Point estimate	Upper C.I.
beta[1]	1.00	1.00	c[1]	1.00	1.00
beta[2]	1.00	1.00	c[1]	1.00	1.01
beta[3]	1.00	1.00	c[1]	1.00	1.00
beta[4]	1.00	1.00	c[1]	1.00	1.00
beta[5]	1.00	1.00	Sigmaalpha	1.01	1.03

As we can see from above, each top-level parameter has R-squared smaller than or equal to 1.1, the rule-of-thumb value, so it shows good convergence overall based on our samples. Additionally, after plotting the posterior distribution of each top-level parameter (provided in the Appendix), we could find clear evidence for good convergence. For instance, based on the graph below for sigmaalpha, there is a relatively stable variation in different chains and the posterior density plot is unimodal and smooth. From the trace plot (see next page), we can see four chains with different colors and traces converge pretty well. We consider these chains to be coming from the same sampling region.



Effect Sample Size

Coefficient	Effective sample sizes	Coefficient	Effective sample sizes
Beta[1]	46780.363	c[1]	4731.994
Beta[2]	45906.801	c[2]	2883.200
Beta[3]	52452.777	c[3]	5946.052
Beta[4]	40992.025	c[4]	23312.202
Beta[5]	47699.410	Sigmaalpha	1004.588

As we can see from the table above (based on our 16,000 iterations and 4 chains, which sums up to 64,000 iterations in total), we have relatively adequate effective simulation sample sizes for all top-level nodes (all larger than 400, which follows the recommendation from our textbook BDA3).

Inference from simulation sampling

```
summary(x1[,c('beta[1]', 'beta[2]', 'beta[3]', 'beta[4]', 'beta[5]')])
##
## Iterations = 16001:32000
## Thinning interval = 1
## Number of chains = 4
## Sample size per chain = 16000
```

Empirical mean and standard deviation for each top-level parameter plus standard error of the mean

	Mean	SD	Naive SE	Time-series SE
beta[1]	0.17928	0.07143	0.0002823	0.0003305
beta[2]	0.03265	0.07094	0.0002804	0.0003322
beta[3]	-0.06834	0.06243	0.0002468	0.0002727
beta[4]	-0.29170	0.07146	0.0002825	0.0003536
beta[5]	0.03331	0.06443	0.0002547	0.0002950

According to our model, we have five independent variables, and here are the quantiles for the regression coefficient of each variable from the posterior simulation samples.

	2.5%	25%	50%	75%	97.5%
beta[1]	0.04063	0.13132	0.17870	0.22693	0.32034
beta[2]	-0.10595	-0.01519	0.03215	0.07995	0.17228
beta[3]	-0.18925	-0.11065	-0.06858	-0.02673	0.05483
beta[4]	-0.43281	-0.33955	-0.29121	-0.24326	-0.15256
beta[5]	-0.09353	-0.01016	0.03346	0.07676	0.15857

From the table, we can see that only Beta [1] and Beta [4] have a 95% posterior credible interval that does not contain 0, which shows a significant relationship with wellbeing. Beta [1] represents the coefficient for age, and Beta [4] is the coefficient for gait speed. Furthermore, we can see the time-series SE as our Monte Carlo standard error for the mean of our simulation samples. For Beta [1], the empirical mean is about 0.17928 with an approximately 0.0003 Monte Carlo standard error, and Beta [4] has -0.2917 for its empirical mean and 0.0003 Monte Carlo standard error. The inferences are made based on the iterations between 16001 and 32000. Based on these values, we can see that our estimation for the means of Beta [1] and Beta [4] are relatively significant with small Monte Carlo standard errors.

From the ordered logistic regression formula, we can derive, for a given predictor x, that

$$\frac{odds(P(y \quad \leq r|x_1, \theta))}{odds(P(y \quad \leq r|x_2, \theta))} = e^{(x_2 - x_1)\beta}$$

If $x_1 = x_2 + 1$,

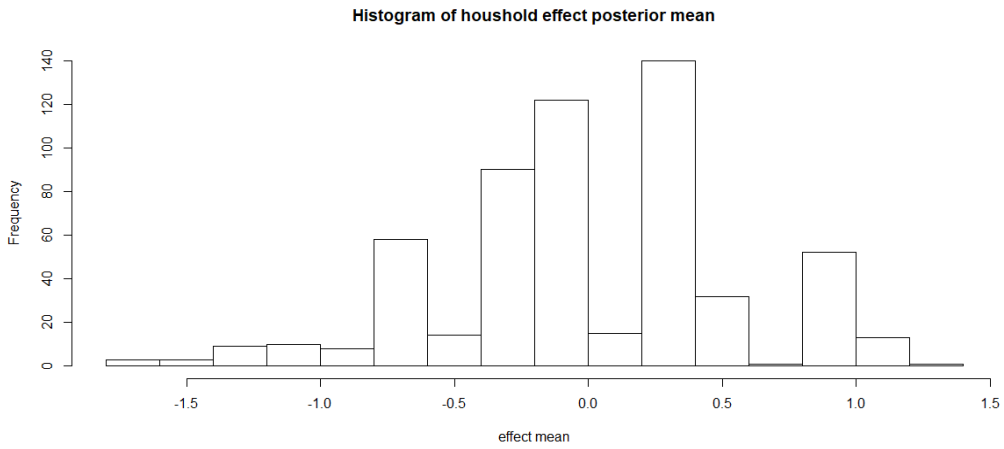
$then\ the\ odds\ ratio\ for\ the\ response\ variable\ being\ less\ than\ a\ specific\ rating\ is\ e^{-\beta}.$

Putting into context of our data, this implies that, for one unit of increase in x, it is $e^{-\beta}\ more\ likely\ to$ rate lower on wellbeing.

Here are the interpretations for the estimated means of the coefficients with a statistically significant effect:

- Beta[1] for Age: An older adult with one standard deviation increase in age is $e^{-0.179}$=0.836 times more likely to rate lower in wellbeing (in other words, he/she is 1.196 times more likely to rate higher in wellbeing);
- Beta[4] for Gait: An older adult with one standard deviation increase in walking time is $e^{0.29} = 1.34$ more likely to rate lower in wellbeing.

To investigate household effects, we could take a look at the posterior distribution of the means for the random intercept α.



According to the histogram of the posterior means of household effect, we can tell that there are great differences in ratings of wellbeing among households. Another way to examine household effects is to look at the hyperparameter. Sigmaalpha has a posterior mean of almost 1 and a 95% posterior credible interval between 0.6711 and 1.2395, which is pretty large and consistent with the above histogram. We suggest that more studies are needed to further explain what factors are contributing to the observed household differences.

```
summary(x1[,'sigmaalpha'])
##
## Iterations = 16001:32000
## Thinning interval = 1
## Number of chains = 4
## Sample size per chain = 16000
```

Empirical mean and standard deviation for sigmaalpha and standard error of the mean

	Mean	SD	Naive SE	Time-series SE
Sigmaalpha	0.9691541	0.1428267	0.0005646	0.0045213

Quantiles for sigmaalpha from the posterior simulation samples

	2.5%	25%	50%	75%	97.5%
Sigmaalpha	0.6711	0.8787	0.9736	1.0647	1.2395

Conclusion

Our findings highlighted that it is older adults' walking speed, rather than grip strength, that is significantly associated with their psychological wellbeing. One possible explanation is that compared to grip strength, walking speed can better reflect an older individual's overall functional ability and health status. For instance, whether or not a person can walk safely and successfully depends on his or her global muscle strength (e.g., core muscle strength and lower extremity muscle strength). High level of balance function, proprioception, and coordination are also required for an older individual to walk fast and safe. On the contrary, grip strength only reveals an individual's local muscle strength. Also, it is plausible that older adults who can walk faster are more likely to engage in leisure activities that are meaningful and interesting to them. For example, older adults who have the ability to walk faster are more likely to travel and visit their relatives/friends compared to their frail counterpart. Engaging in these activities could thus lead to a sense of life satisfaction and thus higher wellbeing. Future studies can further examine the pathway through which walking speed could affect wellbeing.

To briefly sum up, we found that instead of increasing local muscle strength, more emphasis should be placed on improving older adults' overall function and mobility (i.e., ambulation). Our findings underscored this nuance and provided insights into the factors that may contribute to subjective wellbeing among older adults. Moving forward, we suggest that more physical performance measures could be included in future studies. Moreover, studies employing an experimental design (e.g., provision of walking or exercise program) should be considered so that the relationship observed in the present study can be further verified. In conclusion, older adults' walking speed is associated with their wellbeing, and we recommend that this component should be taken into account when developing interventions to promote health and wellbeing in the older-adult population.

References

Bureau, U. S. C. (2021, April 26). *Population and Housing Unit Estimates*. The United States Census Bureau. https://www.census.gov/programs-surveys/popest.html.

Davis, J. C., Bryan, S., Li, L. C., Best, J. R., Hsu, C. L., Gomez, C., Vertes, K. A., & Liu-Ambrose, T. (2015). Mobility and cognition are associated with wellbeing and health related quality of life among older adults: A cross-sectional analysis of the Vancouver Falls Prevention cohort. *BMC Geriatrics*, *15*, 75. https://doi.org/10.1186/s12877-015-0076-2

Freedman, V. A., Stafford, F., Schwarz, N., Conrad, F., & Cornman, J. C. (2012). Disability, participation, and subjective wellbeing among older couples. *Social Science & Medicine*, *74*(4), 588–596. https://doi.org/10.1016/j.socscimed.2011.10.018

George, L. K. (2010). Still happy after all these years: Research frontiers on subjective wellbeing in later life. *The Journals of Gerontology: Series B*, *65B*(3), 331–339. https://doi.org/10.1093/geronb/gbq006

Rantakokko, M., Mänty, M., & Rantanen, T. (2013). Mobility decline in old age. *Exercise and Sport Sciences Reviews*, *41*(1), 19–25. https://doi.org/10.1097/JES.0b013e3182556f1e

Shafrin, J., Sullivan, J., Goldman, D. P., & Gill, T. M. (2017). The association between observed mobility and quality of life in the near elderly. *PLoS ONE*, *12*(8). https://doi.org/10.1371/journal.pone.0182920

Appendix

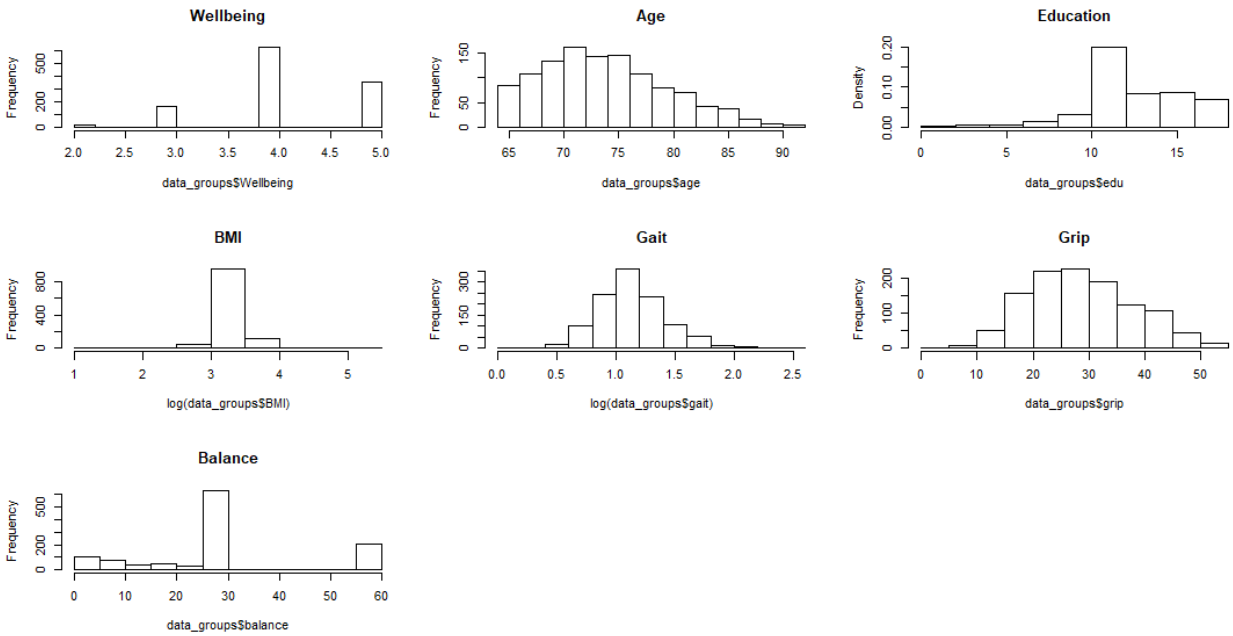
##data Preprocessing

```
library(dplyr)

## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##     filter, lag

## The following objects are masked from 'package:base':
##     intersect, setdiff, setequal, union

library(na.tools)
data <- read.csv("D:\\study\\spring 2021\\stat578\\Final Project\\STAT578 Final Project.csv", header= T)
data <- data[-c(1)]
data.new <- na.rm(data)
data.new$HHID <- as.factor(data.new$HHID)
names(data.new)[names(data.new) == "Wellbeing12"] <- "Wellbeing"
names(data.new)[names(data.new) == "education"] <- "edu"
names(data.new)[names(data.new) == "gaitspeed12"] <- "gait"
names(data.new)[names(data.new) == "grip12"] <- "grip"
names(data.new)[names(data.new) == "balancetime12"] <- "balance"
count <- data.new %>% count(HHID)
data.new = inner_join(data.new, count, by = 'HHID')
colnames(data.new)[10] = 'n_person'
data_groups = data.new[which(data.new$n_person > 1),]
data_groups$HHID = droplevels(data_groups$HHID)
```

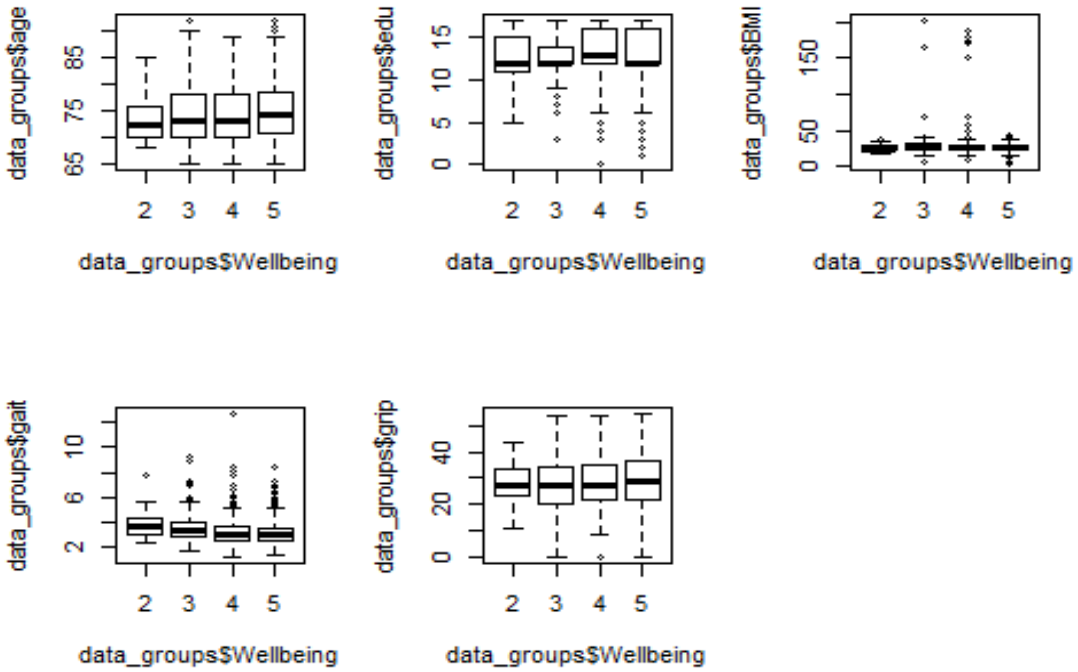
##EDA

```
par(mfrow=c(3,3))
hist(data_groups$Wellbeing)
hist(data_groups$age)
hist(data_groups$edu, freq = FALSE)
hist(log(data_groups$BMI))
hist(log(data_groups$gait))
```

```
hist(data_groups$grip)
hist(data_groups$balance)
par(mfrow=c(2,3))
```




```
boxplot(data_groups$age~data_groups$Wellbeing)
boxplot(data_groups$edu~data_groups$Wellbeing)
boxplot(data_groups$BMI~data_groups$Wellbeing)
boxplot(data_groups$gait~data_groups$Wellbeing)
boxplot(data_groups$grip~data_groups$Wellbeing)
```



Model formulation

JAGS Model code

```
"model{
 for (i in 1:length(Wellbeing)){
  mu[i] <- alpha[HHID[i]] + beta[1]*age[i] + beta[2]*edu[i] + beta[3]*BMI[i] + beta[4]*gait[i] +
beta[5]*grip[i]
  Wellbeing[i] ~ dordered.logit(mu[i], c[1:4])
 }

 for (j in 1:4){
  c0[j] ~ dnorm(0, 0.01)
 }
  c[1:4] <- sort(c0)

 for (h in 1:ngroups){
  alpha[h] ~ dnorm(0, 1/sigmaalpha^2)
 }
  sigmaalpha ~ dexp(0.001)
```

```
  beta ~ dmnorm(beta0, sigmabetainv)

  sigmasqalpha <- sigmaalpha^2
}
"
```

```
## [1] "model{\n  for (i in 1:length(Wellbeing)){\n    mu[i] <- alpha[HHID[i]] + beta[1]*age[i] + beta[2]*edu[i] + beta[3]*BMI[i] + beta[4]*gait[i] + beta[5]*grip[i] \n    Wellbeing[i] ~ dordered.logit(mu[i], c[1:4])\n  }\n    \n  for (j in 1:4){\n    c0[j] ~ dnorm(0, 0.01)\n  }\n    c[1:4] <- sort(c0)\n\n  for (h in 1:ngroups){\n    alpha[h] ~ dnorm(0, 1/sigmaalpha^2)\n  }\n    sigmaalpha ~ dexp(0.001)\n  \n    beta ~ dmnorm(beta0, sigmabetainv)\n  \n    sigmasqalpha <- sigmaalpha^2\n}\n"
```

##Data List and Initializations for JAGS

```
d1 <- list(Wellbeing = factor(data_groups$Wellbeing),
           age = as.vector(scale(data_groups$age)),
           edu = as.vector(scale(data_groups$edu)),
           gait = as.vector(scale(data_groups$gait)),
           grip = as.vector(scale(data_groups$grip)),
           BMI = as.vector(scale(data_groups$BMI)),
           HHID = data_groups$HHID,
           ngroups = length(unique(data_groups$HHID)),
           beta0 = rep(0,5),
           sigmabetainv = 100^(-2)*diag(5)
           )
inits1 <- list(list(c0 = c(-2,-1,0,1)),
               list(c0 = c(-1,-0.5,0,0.5)),
               list(c0 = c(-2.5,-1.5,0,1.5)),
               list(c0 = c(-1.5,-1,0,1)))
```

##Model Running and Convergence Diagnostics

```
library(rjags)
```

```
## Loading required package: coda
```

```
## Linked to JAGS 4.3.0
```

```
## Loaded modules: basemod,bugs

load.module("glm")

## module glm loaded

list.modules()

## [1] "basemod" "bugs"    "glm"

m1 <- jags.model('D:\\study\\spring 2021\\stat578\\Final Project\\HRS.5.bug', d1, inits1, n.chains = 4)

## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 1148
##    Unobserved stochastic nodes: 577
##    Total graph size: 11472
##
## Initializing model

update(m1, 1000)
x1 = coda.samples(m1, c('beta','sigmaalpha','c'), n.iter = 2000)
gelman.diag(x1, autoburnin = FALSE) #run more

## Potential scale reduction factors:
##           Point est. Upper C.I.
## beta[1]         1.00       1.00
## beta[2]         1.00       1.00
## beta[3]         1.00       1.00
## beta[4]         1.00       1.00
## beta[5]         1.00       1.00
## c[1]            1.00       1.00
## c[2]            1.00       1.01
## c[3]            1.00       1.00
## c[4]            1.00       1.00
## sigmaalpha      1.01       1.03
```

```
##
## Multivariate psrf
##
## 1.01

x1 = coda.samples(m1, c('beta','sigmaalpha','c'), n.iter = 4000)
gelman.diag(x1, autoburnin = FALSE) #run more

## Potential scale reduction factors:
##            Point est. Upper C.I.
## beta[1]         1.00       1.00
## beta[2]         1.00       1.00
## beta[3]         1.00       1.00
## beta[4]         1.00       1.00
## beta[5]         1.00       1.00
## c[1]            1.00       1.01
## c[2]            1.01       1.01
## c[3]            1.00       1.01
## c[4]            1.00       1.00
## sigmaalpha      1.02       1.06
##
## Multivariate psrf
##
## 1.02

x1 = coda.samples(m1, c('beta','sigmaalpha','c'), n.iter = 8000)
gelman.diag(x1, autoburnin = FALSE) #run more

## Potential scale reduction factors:
##            Point est. Upper C.I.
## beta[1]         1.00       1.00
## beta[2]         1.00       1.00
## beta[3]         1.00       1.00
## beta[4]         1.00       1.00
## beta[5]         1.00       1.00
## c[1]            1.00       1.00
```

```
## c[2]            1.00       1.01
## c[3]            1.00       1.01
## c[4]            1.00       1.00
## sigmaalpha      1.01       1.04
##
## Multivariate psrf
##
## 1.01
```

```
x1 = coda.samples(m1, c('beta','sigmaalpha','c'), n.iter = 16000)
gelman.diag(x1, autoburnin = FALSE)
```

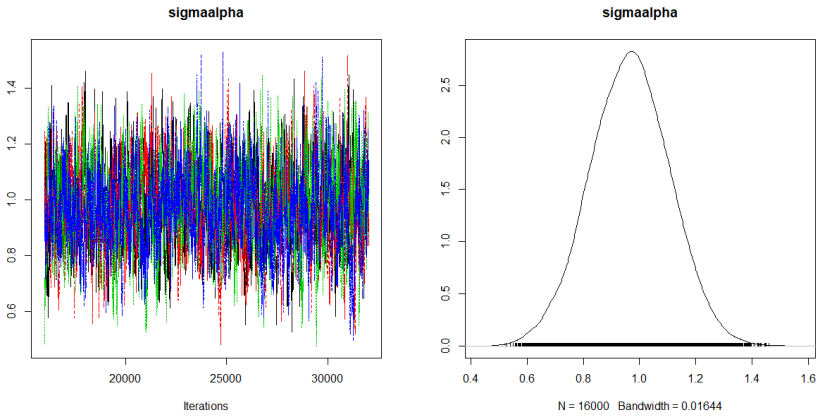
```
## Potential scale reduction factors:
##           Point est. Upper C.I.
## beta[1]            1       1.00
## beta[2]            1       1.00
## beta[3]            1       1.00
## beta[4]            1       1.00
## beta[5]            1       1.00
## c[1]               1       1.00
## c[2]               1       1.00
## c[3]               1       1.00
## c[4]               1       1.00
## sigmaalpha         1       1.01
##
## Multivariate psrf
##
## 1
```

```
effectiveSize(x1[,])
```

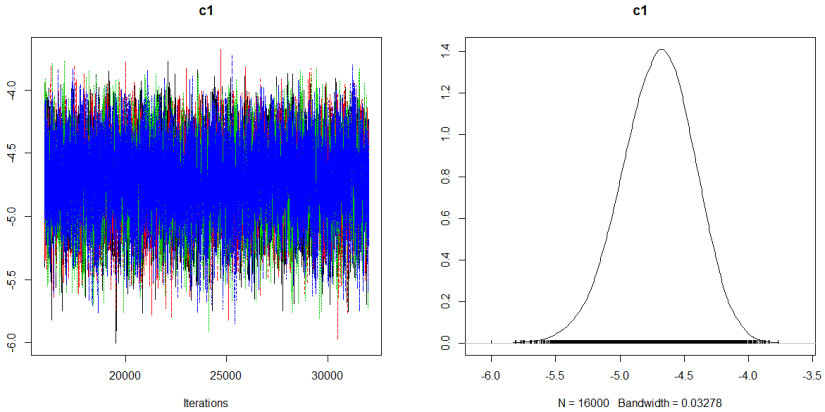
```
##    beta[1]    beta[2]    beta[3]    beta[4]    beta[5]      c[1]      c[2]
##  46780.363  45906.801  52452.777  40992.025  47699.410   4731.994   2883.200
##       c[3]       c[4] sigmaalpha
##   5946.052  23312.202   1004.588
```

Model Analysis

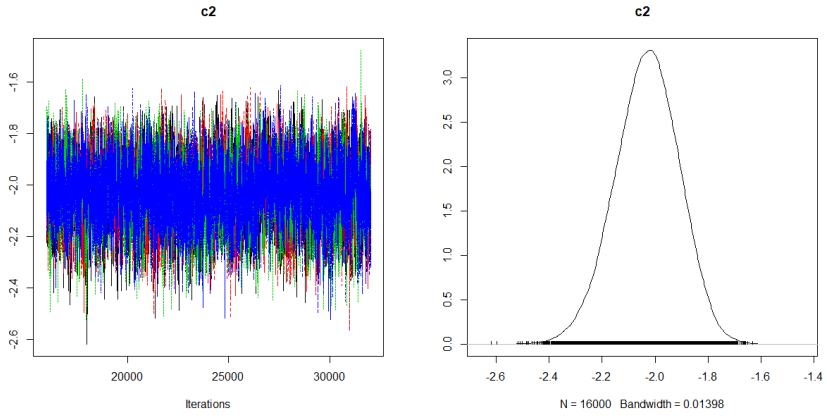
```
plot(x1[,'sigmaalpha'], smooth = FALSE, main = "sigmaalpha") #convergence is good, approx. density is smooth
```



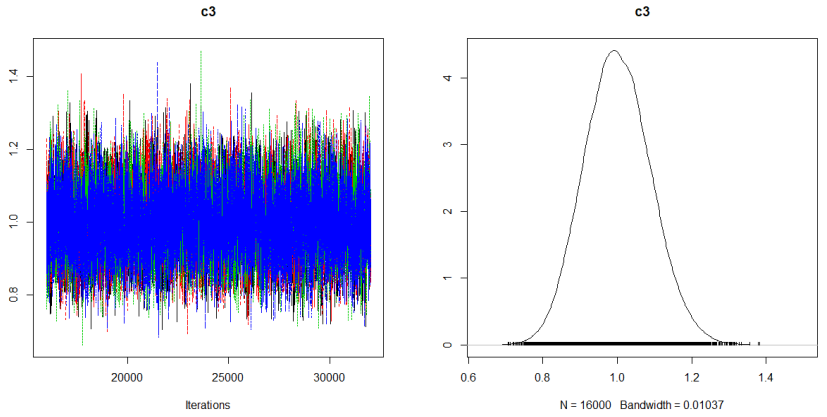

```
plot(x1[,'c[1]'], smooth = FALSE, main = "c1") #good convergence, smooth approx. density
```



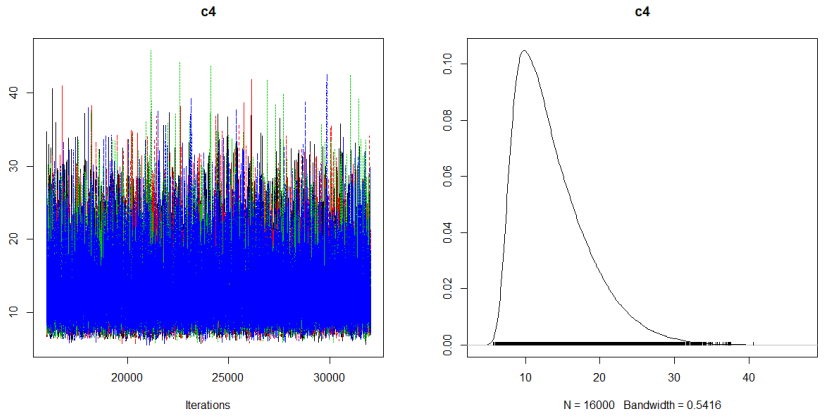

```
plot(x1[,'c[2]'], smooth = FALSE,main = "c2") #good convergence, smooth approx. density
```



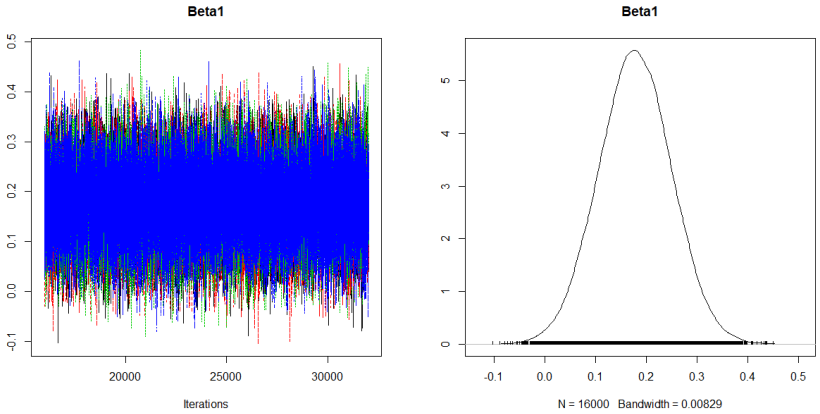

```
plot(x1[,'c[3]'], smooth = FALSE,main = "c3") #good convergence, smooth approx. density
```



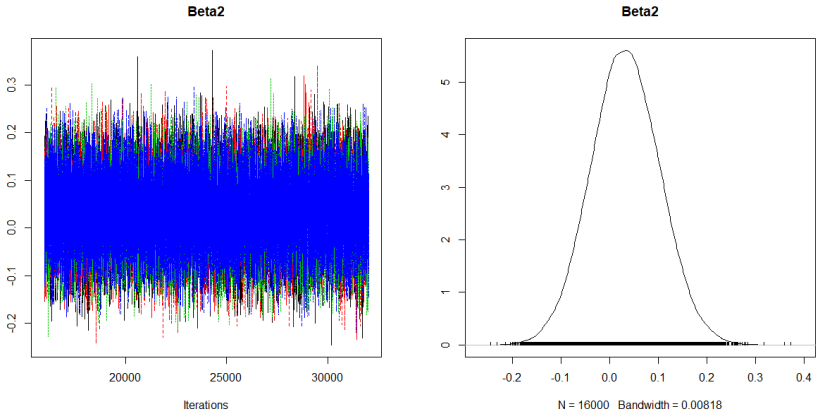

```
plot(x1[,'c[4]'], smooth = FALSE,main = "c4")  #good convergence, smooth approx. density
```



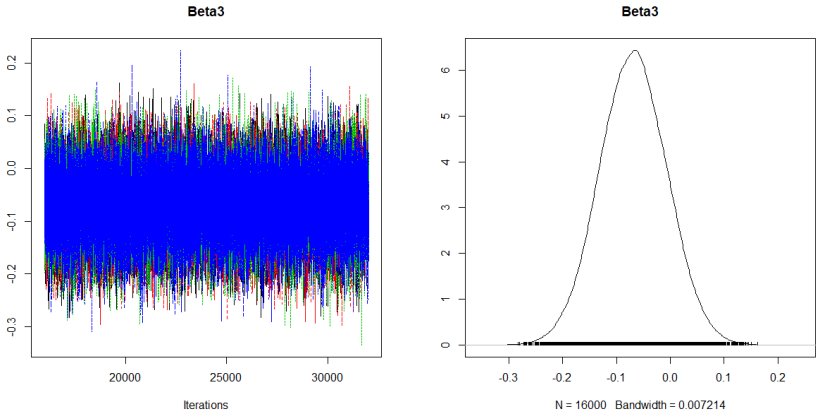

```
plot(x1[,'beta[1]'], smooth = FALSE, main = "Beta1")  #good convergence, smooth approx. density
```



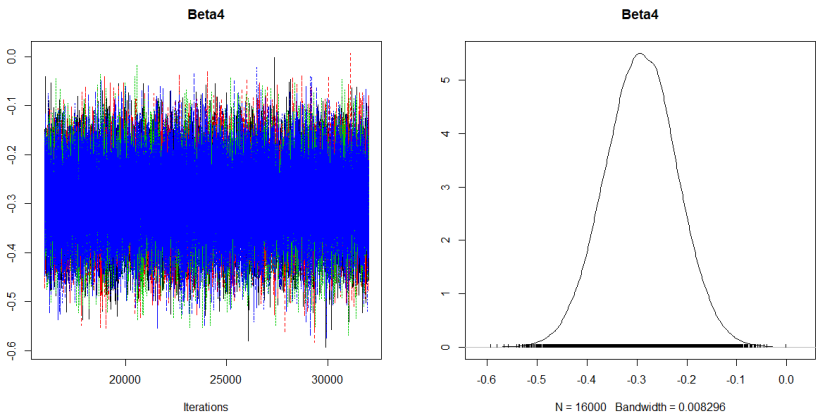

```
plot(x1[,'beta[2]'], smooth = FALSE, main = "Beta2")  #good convergence, smooth approx. density
```



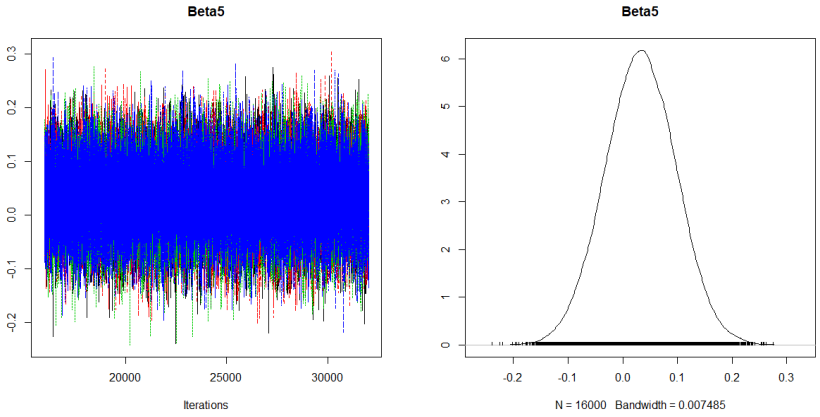

```
plot(x1[,'beta[3]'], smooth = FALSE, main = "Beta3")  #good convergence, smooth approx. density
```




```
plot(x1[,'beta[4]'], smooth = FALSE, main = "Beta4")  #good convergence, smooth approx. density
```




```
plot(x1[,'beta[5]'], smooth = FALSE, main = "Beta5")  #good convergence, smooth approx. density
```




```
summary(x1[,c('beta[1]', 'beta[2]', 'beta[3]', 'beta[4]', 'beta[5]')])

## Iterations = 16001:32000
## Thinning interval = 1
## Number of chains = 4
## Sample size per chain = 16000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##             Mean      SD  Naive SE Time-series SE
## beta[1]  0.17928 0.07143 0.0002823      0.0003305
## beta[2]  0.03265 0.07094 0.0002804      0.0003322
## beta[3] -0.06834 0.06243 0.0002468      0.0002727
```

```
## beta[4] -0.29170 0.07146 0.0002825      0.0003536
## beta[5]  0.03331 0.06443 0.0002547      0.0002950
##
## 2. Quantiles for each variable:
##
##             2.5%      25%      50%      75%    97.5%
## beta[1]  0.04063  0.13132  0.17870  0.22693  0.32034
## beta[2] -0.10595 -0.01519  0.03215  0.07995  0.17228
## beta[3] -0.18925 -0.11065 -0.06858 -0.02673  0.05483
## beta[4] -0.43281 -0.33955 -0.29121 -0.24326 -0.15256
## beta[5] -0.09353 -0.01016  0.03346  0.07676  0.15857
```

```
summary(x1[,'sigmaalpha'])
```

```
##
## Iterations = 16001:32000
## Thinning interval = 1
## Number of chains = 4
## Sample size per chain = 16000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##          Mean             SD       Naive SE Time-series SE
##     0.9691541      0.1428267      0.0005646      0.0045213
##
## 2. Quantiles for each variable:
##
##   2.5%    25%    50%    75%  97.5%
## 0.6711 0.8787 0.9736 1.0647 1.2395
```

```
x1 = coda.samples(m1, c('beta','sigmaalpha','c','alpha'), n.iter = 20000)
post.samp = as.matrix(x1)
dim(post.samp)
```

```
## [1] 80000   581
```

```
effectiveSize(x1[,])

  beta[1]    beta[2]    beta[3]
##  68474.112  67486.342  71552.990  55007.533  58345.959  59221.661  66883.589
##    beta[4]    beta[5]       c[1]       c[2]       c[3]       c[4] sigmaalpha
##  51333.239  58099.786   5667.124   3504.400   6395.413  28463.187   1202.660

hist(apply(post.samp[,paste0("alpha[",1:571,"]")], 2, mean))
```