Checklist

Group project

Contributions of each member

Description of workflow

Statement of technology

References

Group project

Group 11

Fynn Rievers

Emma van Harten

Sam Smits

Giyanto Goossens

Evi Janssen

2024-05-31

Checklist

The submission includes the following.

- ☑ RMD document where it’s clear what is the code that corresponds to each question.
- ☑ Dataset
- ☑ html/PDF document with the following
  - ☑ Numbered questions and answers with text and all the necessary code.
  - ☑ Subtitle indicates the group number
  - ☑ Name of all group members
  - ☑ Details of specification of the work done by group members (e.g., who found the data, who did the pre-processing, who answered which questions, etc).
  - ☑ Statement of technology. Did you use any AI tools? How?

Group project

For the project, we use the following packages:

```
library(dplyr)
library(brms)
library(ggplot2)
library(tidyr)
library(future)
library(priorsense)
options(mc.cores = parallel::detectCores()) # paralellize if possible
options(brms.file_refit = "on_change") # save the files if the model has chang
ed
#nice plotting theme:
theme_set(theme_linedraw() +
            theme(panel.grid = element_blank()))
```

1. Dataset Selection (0.5pt)

Select a dataset with clusters such as schools, regions, or people with multiple observations per individual. (From for example, https://www.kaggle.com/ (https://www.kaggle.com/)) It would be a good idea to choose a smallish dataset (not too many rows, e.g., less than 1000) or subset it so that fitting the models doesn't take too long.

a. Describe the dataset with a couple of short sentences. What was its intended use? Are there papers that reference it? Provide information on how to obtain the dataset, including its source and any necessary preprocessing steps/feature engineering.

The dataset "Life Expectancy (WHO) Fixed" by Lasha (2023) contains a multitude of variables which may affect an individuals' life expectancies per country. It contains variables such as average alcohol consumption, BMI, or GDP per capita, vaccination status for different diseases or years of schooling, among others. A clustering variable can also be found, as the countries are grouped in regions. This dataset contains values ranging from the years 2000 to 2015; we will focus only on the year 2015. The dataset we used is a cleaned version of the original dataset "Life Expectancy (WHO)" by user KumarRajarshi (2018), which contained some outdated or inaccurate information. It also contained some missing values which have been imputed either through the means of using a three year average of a given variable; if said variable was missing for all years, the region average of that variable was computed. Notably, he models created on this dataset can't be used in practise as no new countries will be introduced in the data most likely. Hence the models created by this dataset are only intended for theoretical use and not for the prediction of the life expectancy for new countries that might arise. Furthermore, information about life expectancy, BMI and GDP per capita has been updated as per World Bank data. Information about vaccination status has been collected through other public WHO datasets. Lastly, some countries with an exceptionally high amount of missing variables have been removed entirely from the dataset.

Note should be taken that the dataset contains two variables, once whether a country is developing or not, and separately whether a country is developed or not. Whether countries were developed or developing appears to have been inferred based on Gross National Income per capita. We have converged this variable into a singular dummy variable.

```r
# Load the data and preprocessing steps go here
# Setting work directory to current folder
filepath = rstudioapi::getSourceEditorContext()$path
dirpath  = dirname(rstudioapi::getSourceEditorContext()$path)
setwd(dirpath)

life_expectancy <- read.csv("Life-Expectancy-Data-Updated.csv")

# quick look at the data
print(summary(life_expectancy))
```

```
   Country              Region               Year      Infant_deaths    Under_f
ive_deaths
 Length:2864        Length:2864        Min.   :2000   Min.   :  1.80   Min.   
:  2.300
 Class :character   Class :character   1st Qu.:2004   1st Qu.:  8.10   1st Q
u.:  9.675
 Mode  :character   Mode  :character   Median :2008   Median : 19.60   Median 
: 23.100
                                       Mean   :2008   Mean   : 30.36   Mean   
: 42.938
                                       3rd Qu.:2011   3rd Qu.: 47.35   3rd Q
u.: 66.000
 Adult_mortality  Alcohol_consumption  Hepatitis_B      Measles          BMI
 Min.   : 49.38   Min.   : 0.000      Min.   :12.00   Min.   :10.00   Min.   :
19.80
 1st Qu.:106.91   1st Qu.: 1.200      1st Qu.:78.00   1st Qu.:64.00   1st Qu.:
23.20
 Median :163.84   Median : 4.020      Median :89.00   Median :83.00   Median :
25.50
 Mean   :192.25   Mean   : 4.821      Mean   :84.29   Mean   :77.34   Mean   :
25.03
 3rd Qu.:246.79   3rd Qu.: 7.777      3rd Qu.:96.00   3rd Qu.:93.00   3rd Qu.:
26.40
     Polio         Diphtheria    Incidents_HIV     GDP_per_capita   Population_
mln
 Min.   : 8.0   Min.   :16.00   Min.   : 0.0100   Min.   :   148   Min.   :  
0.080
 1st Qu.:81.0   1st Qu.:81.00   1st Qu.: 0.0800   1st Qu.:  1416   1st Qu.:  
2.098
 Median :93.0   Median :93.00   Median : 0.1500   Median :  4217   Median :  
7.850
 Mean   :86.5   Mean   :86.27   Mean   : 0.8943   Mean   : 11541   Mean   :  3
6.676
 3rd Qu.:97.0   3rd Qu.:97.00   3rd Qu.: 0.4600   3rd Qu.: 12557   3rd Qu.:  2
3.688
 Thinness_ten_nineteen_years Thinness_five_nine_years   Schooling
 Min.   : 0.100              Min.   : 0.1            Min.   : 1.100
 1st Qu.: 1.600              1st Qu.: 1.6            1st Qu.: 5.100
 Median : 3.300              Median : 3.4            Median : 7.800
 Mean   : 4.866              Mean   : 4.9            Mean   : 7.632
 3rd Qu.: 7.200              3rd Qu.: 7.3            3rd Qu.:10.300
 Economy_status_Developed Economy_status_Developing Life_expectancy
 Min.   :0.0000           Min.   :0.0000            Min.   :39.40
 1st Qu.:0.0000           1st Qu.:1.0000            1st Qu.:62.70
 Median :0.0000           Median :1.0000            Median :71.40
 Mean   :0.2067           Mean   :0.7933            Mean   :68.86  
 3rd Qu.:0.0000           3rd Qu.:1.0000            3rd Qu.:75.40  
 [ reached getOption("max.print") -- omitted 1 row ]
```

```
# Check for NA's per variable in the year 2015 and create a summary table
na_summary<- life_expectancy %>%
  filter(Year == 2015) %>%
  summarise_all(~ mean(is.na(.)) * 100) %>%
  gather(key = "Variable", value = "NA_Percentage") %>%
  arrange(desc(NA_Percentage))
print(na_summary)
```

```
                      Variable NA_Percentage
1                      Country             0
2                       Region             0
3                         Year             0
4                Infant_deaths             0
5            Under_five_deaths             0
6              Adult_mortality             0
7          Alcohol_consumption             0
8                  Hepatitis_B             0
9                      Measles             0
10                         BMI             0
11                       Polio             0
12                  Diphtheria             0
13               Incidents_HIV             0
14              GDP_per_capita             0
15              Population_mln             0
16 Thinness_ten_nineteen_years             0
17    Thinness_five_nine_years             0
18                   Schooling             0
19    Economy_status_Developed             0
20   Economy_status_Developing             0
21             Life_expectancy             0
```

```
#no missing values



# Filter to Year == 2015 so we only deal with cross-sectional data. After that
we can drop Year variable

life_expectancy <- life_expectancy %>%
  filter(Year == 2015) %>%
  select(-Year)



# Economy_status_Developing is redundant, since Economy_status_Developed entai
ls all its information
life_expectancy <- life_expectancy %>%
  select(-Economy_status_Developing)



print(summary(life_expectancy))
```

```
   Country             Region          Infant_deaths   Under_five_deaths Adult
_mortality
 Length:179         Length:179         Min.   : 1.80   Min.   :  2.30    Min.   
: 49.38  
 Class :character   Class :character   1st Qu.: 6.65   1st Qu.:  7.85    1st Q
u.: 90.79  
 Mode  :character   Mode  :character   Median :15.20   Median : 17.50    Media
n :146.52  
                                       Mean   :23.56   Mean   : 31.68    Mean   
:163.67  
                                       3rd Qu.:36.55   3rd Qu.: 49.95    3rd Q
u.:215.65  
                                       Max.   :95.10   Max.   :140.20    Max.   
:513.48  
 Alcohol_consumption  Hepatitis_B      Measles           BMI           Polio     
 Min.   : 0.000      Min.   :22.0   Min.   :21.00   Min.   :20.5   Min.   :37.
00  
 1st Qu.: 1.360      1st Qu.:82.5   1st Qu.:64.00   1st Qu.:23.8   1st Qu.:85.
00  
 Median : 4.040      Median :92.0   Median :84.00   Median :26.2   Median :93.
00  
 Mean   : 4.729      Mean   :87.1   Mean   :80.23   Mean   :25.6   Mean   :88.
26  
 3rd Qu.: 7.760      3rd Qu.:97.0   3rd Qu.:94.00   3rd Qu.:27.0   3rd Qu.:97.
00  
 Max.   :16.720      Max.   :99.0   Max.   :99.00   Max.   :32.1   Max.   :99.
00  
   Diphtheria    Incidents_HIV     GDP_per_capita   Population_mln    
 Min.   :16.00   Min.   : 0.0100   Min.   :   306   Min.   :   0.090  
 1st Qu.:85.50   1st Qu.: 0.0800   1st Qu.:  1690   1st Qu.:   2.215  
 Median :93.00   Median : 0.1400   Median :  5391   Median :   9.110  
 Mean   :87.92   Mean   : 0.6098   Mean   : 12617   Mean   :  40.088  
 3rd Qu.:97.00   3rd Qu.: 0.3700   3rd Qu.: 14274   3rd Qu.:  27.445  
 Max.   :99.00   Max.   :14.3000   Max.   :105462   Max.   :1379.860  
 Thinness_ten_nineteen_years Thinness_five_nine_years   Schooling     
 Min.   : 0.10               Min.   : 0.100           Min.   : 1.400  
 1st Qu.: 1.50               1st Qu.: 1.500           1st Qu.: 5.950  
 Median : 3.50               Median : 3.400           Median : 8.700  
 Mean   : 4.55               Mean   : 4.594           Mean   : 8.361  
 3rd Qu.: 6.50               3rd Qu.: 6.450           3rd Qu.:11.050  
 Max.   :26.70               Max.   :27.300           Max.   :14.100  
 Economy_status_Developed Life_expectancy
 Min.   :0.0000           Min.   :50.90  
 1st Qu.:0.0000           1st Qu.:66.30  
 Median :0.0000           Median :73.00  
 Mean   :0.2067          Mean   :71.46  
 3rd Qu.:0.0000          3rd Qu.:76.85  
 Max.   :1.0000          Max.   :83.80  
```

```
# Don't see any strange data/outliers



##### Next step is to scale numerical values and convert categorical variables
to factors

# Ensure categorical variables are treated as a categorical variable
life_expectancy <- life_expectancy %>%
  mutate(
    Country = as.factor(Country),
    Region = as.factor(Region),
    Economy_status_Developed = as.factor(Economy_status_Developed)
  )



# Identify numeric and categorical variables
numeric_vars <- life_expectancy %>%
  select_if(is.numeric) %>%
  names()



# Remove 'Life_expectancy' from the numeric variables to avoid scaling it
numeric_vars <- setdiff(numeric_vars, "Life_expectancy")




# Identify the updated list of categorical variables
categorical_vars <- life_expectancy %>%
  select_if(is.factor) %>%
  names()






# Convert categorical variables to factors
life_expectancy<- life_expectancy %>%
  mutate(across(all_of(categorical_vars), as.factor))

# Check the structure of the preprocessed data
str(life_expectancy)
```

```
'data.frame':	179 obs. of  19 variables:
 $ Country                    : Factor w/ 179 levels "Afghanistan",..: 165 149 
134 30 61 3 123 98 122 176 ...
 $ Region                     : Factor w/ 9 levels "Africa","Asia",..: 5 4 8 1 
1 1 5 1 8 2 ...
 $ Infant_deaths              : num  11.1 2.7 6.6 57 39.7 21.6 9.6 41.3 2.2 1
7.4 ...
 $ Under_five_deaths          : num  13 3.3 8.2 88 59.8 25.2 11.2 59 2.7 21.8 
...
 $ Adult_mortality            : num  105.8 57.9 223 340.1 261.7 ...
 $ Alcohol_consumption        : num  1.32 10.35 8.06 4.55 2.69 ...
 $ Hepatitis_B                : int  97 97 97 84 97 95 99 69 88 97 ...
 $ Measles                    : int  65 94 97 64 64 99 98 64 91 65 ...
 $ BMI                        : num  27.8 26 26.2 24.3 23.9 25.5 26.3 21.3 26.
6 21.7 ...
 $ Polio                      : int  97 97 97 77 96 95 99 68 95 97 ...
 $ Diphtheria                 : int  97 97 97 84 97 95 99 69 95 97 ...
 $ Incidents_HIV              : num  0.08 0.09 0.08 1.12 0.96 0.05 0.05 0.24 
0.04 0.12 ...
 $ GDP_per_capita             : int  11006 25742 9313 1383 661 4178 18445 467 
74356 2582 ...
 $ Population_mln             : num  78.53 46.44 144.1 23.3 2.09 ...
 $ Thinness_ten_nineteen_years: num  4.9 0.6 2.3 5.6 7.3 6 7.1 7.1 0.8 14.2 
...
 $ Thinness_five_nine_years   : num  4.8 0.5 2.3 5.5 7.2 5.8 6.9 7.1 0.7 14.5 
...
 $ Schooling                  : num  7.8 9.7 12 6.1 3.4 7.9 9.5 6.1 12.5 8 ...
 $ Economy_status_Developed   : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 1 2 
1 ...
 $ Life_expectancy            : num  76.5 82.8 71.2 57.6 60.9 76.1 76.9 65.5 8
2.3 75.1 ...
```

```
# Structure seems fine

# Summary of the preprocessed data
summary(life_expectancy)
```

```
                Country                                   Region   Infant_deaths  
 Afghanistan        :  1   Africa                          :51   Min.   : 1.80  
 Albania            :  1   Asia                            :27   1st Qu.: 6.65  
 Algeria            :  1   European Union                  :27   Median :15.20  
 Angola             :  1   Central America and Caribbean:19   Mean   :23.56  
 Antigua and Barbuda:  1   Rest of Europe                  :15   3rd Qu.:36.55  
 Argentina          :  1   Middle East                     :14   Max.   :95.10  
 Under_five_deaths Adult_mortality  Alcohol_consumption  Hepatitis_B      Meas
les      
 Min.   :  2.30    Min.   : 49.38   Min.   : 0.000      Min.   :22.0   Min.   
:21.00  
 1st Qu.:  7.85    1st Qu.: 90.79   1st Qu.: 1.360      1st Qu.:82.5   1st Q
u.:64.00  
 Median : 17.50    Median :146.52   Median : 4.040      Median :92.0   Median 
:84.00  
 Mean   : 31.68    Mean   :163.67   Mean   : 4.729      Mean   :87.1   Mean   
:80.23  
 3rd Qu.: 49.95    3rd Qu.:215.65   3rd Qu.: 7.760      3rd Qu.:97.0   3rd Q
u.:94.00  
 Max.   :140.20    Max.   :513.48   Max.   :16.720      Max.   :99.0   Max.   
:99.00  
      BMI           Polio          Diphtheria     Incidents_HIV     GDP_per_capi
ta     
 Min.   :20.5   Min.   :37.00   Min.   :16.00   Min.   : 0.0100   Min.   :   3
06  
 1st Qu.:23.8   1st Qu.:85.00   1st Qu.:85.50   1st Qu.: 0.0800   1st Qu.:  16
90  
 Median :26.2   Median :93.00   Median :93.00   Median : 0.1400   Median :  53
91  
 Mean   :25.6   Mean   :88.26   Mean   :87.92   Mean   : 0.6098   Mean   : 126
17  
 3rd Qu.:27.0   3rd Qu.:97.00   3rd Qu.:97.00   3rd Qu.: 0.3700   3rd Qu.: 142
74  
 Max.   :32.1   Max.   :99.00   Max.   :99.00   Max.   :14.3000   Max.   :1054
62  
 Population_mln     Thinness_ten_nineteen_years Thinness_five_nine_years   Sch
ooling     
 Min.   :   0.090   Min.   : 0.10               Min.   : 0.100           Min.   
: 1.400  
 1st Qu.:   2.215   1st Qu.: 1.50               1st Qu.: 1.500           1st Q
u.: 5.950  
 Median :   9.110   Median : 3.50               Median : 3.400           Media
n : 8.700  
 Mean   :  40.088   Mean   : 4.55               Mean   : 4.594           Mean   
: 8.361  
```

```
 3rd Qu.:  27.445   3rd Qu.: 6.50              3rd Qu.: 6.450          3rd Q
u.:11.050
 Max.   :1379.860   Max.   :26.70              Max.   :27.300          Max.   
:14.100
 Economy_status_Developed Life_expectancy
 0:142                    Min.   :50.90  
 1: 37                    1st Qu.:66.30  
                          Median :73.00  
                          Mean   :71.46  
                          3rd Qu.:76.85  
                          Max.   :83.80  
 [ reached getOption("max.print") -- omitted 1 row ]
```

b. Report the number of observations, columns (with their meaning) and their data types. Indicate clearly what you will use as dependent variable/label.

This dataset contains information about 179 different countries over 16 years, which each row containing information about one country for one year. As stated previously, we will only look at data from 2015, therefore having 179 rather than 2864 rows. As for columns, the dataset contains the following: Region (distributed in nine different regions, namely: Africa, Asia, Central America and Carribean, EU, Middle East, Oceania, North America, Rest of Europe, South America. Then, the dataset contains a column indicating year, which we remove after filtering to 2015-data only. Next are Infant deaths (Infant_deaths), deaths under five years old (Under_five_deaths), and adult deaths (Adult_mortality) per 1000 people respectively. The variable Alcohol_consumption contains the consumption of alcohol per capita for individuals >=15 years old in liters. The variables Hepatitis_B, Measles, Polio, and Diphteria contain the percentage of individuals vaccinated against the respective disease. Meanwhile, the variable Incidents_HIV measures the amount of occurences of HIV per 1000 people for individuals aged 15-49. The BMI is covered through the column of the same name, while the variables “Thinness_ten_nineteen_years” and “Thinness_five_nine_years” contain the prevalence of people in the specified age intervals who have a BMI of 2 standard deviations below the median. The variables GDP_per_capita and Population_mln (=million) are explained through their names. The variable Schooling contains the average amount of years spent in school by people aged 25+, while the columns beginning with “Economy_stats_xyz” contain the dummy variable whether a country is developing or developed, which we formed into a singular dummy variable with developed = 1 and developing = 0. Lastly, the variable Life_expectancy contains the average life expectancy in the country (not split by sex). This will be out dependent variable.

2. Split the data and tranform columns as necessary. (0.5pt)

Split the data into training (80%) and test set (80%). Transform the columns if necessary.

```
#setting seed for reproducability
set.seed(123)

# assigning 80% to training set
life_expectancy_train <- slice_sample(life_expectancy, prop = 0.8)

# assigning the other 20% to the test set
life_expectancy_test <- anti_join(life_expectancy, life_expectancy_train)




# Scale numeric variables for training set
life_expectancy_train <- life_expectancy_train %>%
  mutate(across(all_of(numeric_vars), ~ scale(.) %>% as.vector()))  # (scale() scales each numeric variable by substracting the mean and dividing by the standard deviation)


# Scale numeric variables for test set
life_expectancy_test <- life_expectancy_test %>%
  mutate(across(all_of(numeric_vars), ~ scale(.) %>% as.vector()))  # (scale() scales each numeric variable by substracting the mean and dividing by the standard deviation)
```

3. Model Exploration (3pt)

a. Fit multiple appropriate models to the dataset (as many models as there are members in the group, with a minimum of two models). Models might vary in the multilevel structure, informativeness of their priors (but not just trivial changes), model of the data/likelihood, etc. (I recommend not to use no pooling models since they tend to take a long time and it’s very hard to assign good priors).

```
#######################################################
#-- Model 1:  Random intercept for Region (Giyanto) --#
#######################################################

# Define the formula for the model
formula.m1 <- bf(
  Life_expectancy ~ Alcohol_consumption + Adult_mortality + Hepatitis_B + Meas
les + BMI + Polio + Diphtheria + Incidents_HIV + GDP_per_capita + Population_m
ln + Thinness_ten_nineteen_years + Thinness_five_nine_years + Schooling + Econ
omy_status_Developed + (1 | Region)
)


#setting up the priors using Gelman et al's default priors
get_prior(formula.m1, data = life_expectancy)

#alpha
print(c(mean(life_expectancy_train$Life_expectancy), 2.5*sd(life_expectancy_tr
ain$Life_expectancy)))
#sigma
print(1/sd(life_expectancy_train$Life_expectancy))


GelmanEtAl <- c(
  prior(normal(71.37, 19.92), class = "Intercept"),
  prior(normal(0, 19.92), class = "b"),
  prior(exponential(0.1255), class = "sigma")
)




m.1 <- brm(formula.m1,
              data = life_expectancy_train,
              family = gaussian(),
              prior = GelmanEtAl,
              seed = 123,
              file = paste0(dirpath,"/fits/Bayesian_Group_Assignment_m1"))

m.1

##############################################################################
##################
#-- Model 2: Random intercept for Region | Priors based on EDA & Student distr
ibution (Fynn) --#
##############################################################################
```

```
##################

# Setting up priors based on previously conducted EDA on this dataset
InformedPriors <- c(
  prior(normal(71.37, 19.92), class = "Intercept"),
  prior(normal(0, 19.92), class = "b"),
  prior(normal(-0.8, 5), class = "b", coef="Adult_mortality"),
  prior(normal(0, 5), class = "b", coef="Schooling"),
  prior(normal(0, 5), class = "b", coef="BMI"),
  prior(normal(0, 30), class = "b", coef="Population_mln"),
  prior(exponential(0.1255), class = "sigma")
)



m.2 <- brm(formula.m1,
               data = life_expectancy_train,
               family = student(),
               prior = InformedPriors,
               seed = 123,
               file = paste0(dirpath,"/fits/Bayesian_Group_Assignment_m2"))

m.2

#######################################################################
#-- Model 3:  Random intercept for Region | Horseshoe priors (Evi) --#
######################################################################

formula.m1 <- bf(
  Life_expectancy ~ Alcohol_consumption + Adult_mortality + Hepatitis_B + Measles + BMI + Polio + Diphtheria + Incidents_HIV + GDP_per_capita + Population_mln + Thinness_ten_nineteen_years + Thinness_five_nine_years + Schooling + Economy_status_Developed + (1 | Region)
)

HorseshoePriors <- c(
  prior(horseshoe(df = 1, scale_global = 0.5, scale_slab = 10), class = "b"))

m.3 <- brm(
  formula = formula.m1,
  data = life_expectancy_train,
  family = gaussian(),
  prior = HorseshoePriors,
  file = paste0(dirpath,"/fits/Bayesian_Group_Assignment_m3"),
  seed = 123)
```

```
m.3

################################################################################
#####
#-- Model 4:  Varying intercept and varying slopes model | Gelman priors (Sam)
--#
################################################################################
#####

# Define the formula for the model
formula.m4 <- bf(
  Life_expectancy ~ Alcohol_consumption + Adult_mortality + Hepatitis_B + Meas
les + BMI + Polio + Diphtheria + Incidents_HIV + GDP_per_capita + Population_m
ln + Thinness_ten_nineteen_years + Thinness_five_nine_years + Schooling + Econ
omy_status_Developed + (1 + GDP_per_capita || Region)
)

GelmanEtAl <- c(
  prior(normal(71.37, 19.92), class = "Intercept"),
  prior(normal(0, 19.92), class = "b"),
  prior(exponential(0.1255), class = "sigma")
)

m.4 <- brm(formula.m4,
               data = life_expectancy_train,
               family = gaussian(),
               prior = GelmanEtAl,
               file = paste0(dirpath,"/fits/Bayesian_Group_Assignment_m4"),
               seed = 123)

m.4

##############################################################################
#-- Model 5:  Correlated varying intercept & varing slopes model (Emma) --#
#############################################################################
# Define the formula for the model
formula.m5 <- bf(
  Life_expectancy ~ Alcohol_consumption + Adult_mortality + Hepatitis_B + Meas
les + BMI + Polio + Diphtheria + Incidents_HIV + GDP_per_capita + Population_m
ln + Thinness_ten_nineteen_years + Thinness_five_nine_years + Schooling + Econ
omy_status_Developed + (1 + GDP_per_capita | Region)
)

prior_info_m5 <- get_prior(
  Life_expectancy ~ Alcohol_consumption + Hepatitis_B + Measles + BMI + Polio
+ Diphtheria + Incidents_HIV + GDP_per_capita + Population_mln + Thinness_ten_
nineteen_years + Thinness_five_nine_years + Schooling + Economy_status_Develop
ed + (1 + GDP_per_capita | Region),
  data = life_expectancy_train)

gelman_priors_m5 <- c(GelmanEtAl, prior(lkj(2), class = cor))

m.5 <- brm(formula.m5,
  data = life_expectancy_train,
  prior = gelman_priors_m5,
  control = list(adapt_delta = .95),
  seed = 123,
  file = paste0(dirpath,"/fits/Bayesian_Group_Assignment_m5")
)

m.5
```

b. Explain each model and describe its structure (what they assume about potential population- level or group-level effects), and the type of priors used.

3.1 Model 1: Random Intercept for Region (Giyanto)

This model predicts life expectancy using multiple predictors, including alcohol consumption, hepatitis B vaccination coverage, measles vaccination coverage, body mass index (BMI), polio vaccination coverage, diphtheria vaccination coverage, HIV incidence, GDP per capita, population size, thinness among ten to nineteen-year-olds, thinness among five to nine-year-olds, schooling, and whether the economy is developed. The model incorporates a random intercept for regions to account for unobserved heterogeneity across different regions.

The fixed effects (population-level effects) in this model represent the average impact of each predictor on life expectancy across all regions. By including a random intercept for regions, the model allows the baseline life expectancy to vary between different regions, capturing regional differences that are not explained by the fixed effects.

The priors used in this model are based on recommendations by Gelman et al. The prior for the intercept (α) is a normal distribution with a mean of 71.37 and a standard deviation of 19.92, reflecting the average life expectancy across all regions according to the data distribution. The prior for the fixed effects coefficients (β) is also a normal distribution with a mean of 0 and a standard deviation of 19.92, which serves as a weakly informative prior. This allows the data to dominate the estimates while providing some regularization. The prior for the residual standard deviation (σ) follows an exponential distribution with a rate of 0.1255, suggesting a relatively small standard deviation and encouraging the model to fit the data closely.

3.2 Model 2: Random intercept for Region | Priors based on EDA & Student distribution (Fynn)

The second model builds on the first one by keeping most of the Gelman et al. priors, but using information of previous studies on this dataset in order to learn which features are more/less strongly correlated with our outcome variable, Life expectancy. I did this in the following ways: i opted for smaller standard deviations for those variables which have been confirmed to be strongly correlated (>= +/-0.65) with life expectancy in at least two out of three chosen highly upvoted EDA/ML studies on Kaggle working with this dataset or the original WHO dataset that this data is derived from. If any variables also had exceptionally low correlation in 2+ of these studies, I set up larger standard deviations for them. To gather this information, I read through the EDA reports of the studies with a main focus on the heatmaps provided.

Following this, I attempted to use reasonable intuition to decide which variables may have a larger effect size on life expectancy by considering which variables have a direct vs. indirect effect on an individuals' life (i.e., Adult mortality has a very direct effect: If many adults die, odds are that many young adults die, lowering the life expectancy; on the other hand, Schooling is highly correlated but likely in more indirect ways: less schooling leads to worse job prospects leads to lower wages leads to a less healthy lifestyle leading to lower life expectancy.) For variables which i expect to have a very direct and therefore probably larger effect size, I changed their mean to be the of effect sizes according those previous studies who attested it as being high correlation (see the example for adult mortality below). This leads to the following changes in conclusion:

Adult_mortality (which was originally not part of model 1 but has been added subsequently after being found highly correlated) was found in two studies to be highly negatively correlated with life expectancy by 2/3 studies, and given that it is a very direct causal connection, it is probably not only a significant but also a strong predictor. For being significant, it receives a relatively low SD of 5, roughly a quarter of the SD set by Gelman et al. One of the studies attested a correlation of -0.65 with life expectancy, the other attested a correlation of -0.95. This averages to -0.8, which will be used as the mean.

Schooling is a more indirect predictor which nonetheless appears to have a strong connection with life expectancy. Therefore the mean will remain the same (0), but the standard deviation will be lowered to 5.

BMI is a more difficult choice to make, as a very low BMI may indicate starvation and higher mortality due to that, while high BMI could lead to one of two conclusions: lower expectancy due to health complications related to obesity, or higher life expectancy due to abundance in food, as is observed in many highly developed western countries. Given that it is difficult to really tell the direction of BMI, I decided to not change the mean, but still change the SD as BMI is undoubtably a relevant factor in life expectancy, which is also backed up by previous research.

Lastly, all three studies agreed that the population count of the country has little to no effect on the life expectancy, as such it received a much higher standard deviation of 30. One could also consider to remove the variable as a predictor overall, but i considered it to be a better option to leave it in while ensuring to model a high level of uncertainty.

For the other variables in the dataset , there are definitely still varying degrees of correlation with life expectancy, meaning more fine-tuning would be possible, but for many of them i couldn’t find sufficient agreement between the studies to really justify making specific changes over the generally recommended priors of Gelman et al.

Furthermore, one more experiment conducted within my model was a switch between different family choices. Considering that life expectancy can’t be negative and, especially for newly developing nations has a tendency to increase drastically (consider the effect of industrialization on life expectancy), looking at a lognormal() distribution was a choice; however my result were consistently worse when using anything except Gaussian(). I assume that a lognormal distribution might be interesting when looking at more longitudinal as life expectancy likely requires some time to increase. A change to a student distribution showed a mildly positive effect though and has thus been kept.

3.3 Model 3: Random intercept for Region | Horseshoe priors (Evi)

This model predicts life-expectancy on the basis of the same predictors as previously established models. The model varies in its approach to the priors used for training the model.

Horseshoe priors are Bayesian prior distributions that allow for sparsity in regression coefficients by allowing some coefficients to be close to zero while allowing others to have large magnitudes. The regularization terms included in horseshoe priors encompass a global shrinkage component, which applies a level of shrinkage to all coefficients, and a slab component, which acts as a heavy-tailed distribution allowing some coefficients to remain large despite the global shrinkage, thereby promoting sparsity in the model.

The scale_global parameter sets the scale of the global shrinkage component of the prior. The scale_slab parameter sets the scale of the slab component of the prior. A larger scale (like 10) allows for a wider range of plausible coefficient values.

3.4 Model 4: Varying intercept and varying slopes model | Gelman priors (Sam)

As with the first, second and third model, this model predicts the life expectancy using the different predictors in the dataset.The model furthermore incorporates a random intercept for regions to account for unobserved heterogeneity across different regions. Additionally, a varying slope for the different regions is added, as the effect of GDP per capita on life expectancy could be different accross the different regions. The varying slope and the varying intercept for the region, is the group level effect in this case. The other paramters in the model imply the population level effect.

The model accounts for regional differences in the baseline life expectancy and the different effect of GDP per capita per region. The significant (when analysing the confidence intervals) standard deviations (sd(Intercept) and sd(GDP_per_capita)) indicate that there is notable variation in life expectancy and the impact of GDP per capita across regions. Additionally, Incidents_HIV has a significant negative effect and GDP_per_capita and Schooling have a significant positive effect on life expectancy when analysing the confidence intervals.

The priors used in this model are the same as explored in the first model.

3.5 Model 5: Correlated varying intercept & varing slopes model (Emma)

This model also predicts life expectancy based on several predicters used in the dataset, with varying intercepts and slopes for GDP_per_capita by Region. The model allows the intercept and slope for GDP_per_capita to vary by region. This means that each region can have its own baseline life expectancy and its own effect of GDP_per_capital on life expectancy.

There is a negative correlation which suggests that regions with higher baseline life expectancy tend to have a weaker relationship between GDP_per_capita and life expectancy, and vice versa. However, the confidence interval (-0.87 to 0.57) is quite wide. This indicates that there is some uncertainty about the direction and strenght of this correlation between regions and GDP_per_capita.

The priors used in this model are the Gelman et al, but the correlation parameter is added.

4. Model checking (3pt)

a. Perform a prior sensitivity analysis for each model and modify the model if appropriate. Justify.

```
powerscale_sensitivity(m.1)
```

```
Sensitivity based on cjs_dist:
# A tibble: 27 × 4
   variable                 prior likelihood diagnosis
   <chr>                    <dbl>      <dbl> <chr>
 1 b_Intercept           0.000953     0.0210 -
 2 b_Alcohol_consumption 0.000701     0.113  -
 3 b_Adult_mortality     0.000608     0.103  -
 4 b_Hepatitis_B         0.000345     0.103  -
 5 b_Measles             0.000213     0.0849 -
 6 b_BMI                 0.000249     0.0898 -
 7 b_Polio               0.000306     0.120  -
 8 b_Diphtheria          0.000296     0.110  -
 9 b_Incidents_HIV       0.000585     0.103  -
10 b_GDP_per_capita      0.000955     0.0876 -
# ℹ 17 more rows
```

```
powerscale_sensitivity(m.2)
```

```
Sensitivity based on cjs_dist:
# A tibble: 28 × 4
   variable                prior likelihood diagnosis
   <chr>                   <dbl>      <dbl> <chr>
 1 b_Intercept           0.00441     0.0418 -
 2 b_Alcohol_consumption 0.00440     0.139  -
 3 b_Adult_mortality     0.00813     0.119  -
 4 b_Hepatitis_B         0.00338     0.117  -
 5 b_Measles             0.00281     0.103  -
 6 b_BMI                 0.00433     0.0939 -
 7 b_Polio               0.00433     0.111  -
 8 b_Diphtheria          0.00453     0.146  -
 9 b_Incidents_HIV       0.00530     0.114  -
10 b_GDP_per_capita      0.00530     0.121  -
# ℹ 18 more rows
```

```
powerscale_sensitivity(m.3)
```

```
Sensitivity based on cjs_dist:
# A tibble: 43 × 4
   variable                prior likelihood diagnosis
   <chr>                   <dbl>      <dbl> <chr>
 1 b_Intercept           0.00783     0.0545 -
 2 b_Alcohol_consumption 0.0119      0.160  -
 3 b_Adult_mortality     0.00523     0.126  -
 4 b_Hepatitis_B         0.00647     0.0234 -
 5 b_Measles             0.00724     0.0735 -
 6 b_BMI                 0.0169      0.0553 -
 7 b_Polio               0.00476     0.0413 -
 8 b_Diphtheria          0.00738     0.0337 -
 9 b_Incidents_HIV       0.00757     0.185  -
10 b_GDP_per_capita      0.00691     0.172  -
# ℹ 33 more rows
```

```
powerscale_sensitivity(m.4)
```

```
Sensitivity based on cjs_dist:
# A tibble: 37 × 4
   variable                 prior likelihood diagnosis
   <chr>                    <dbl>      <dbl> <chr>
 1 b_Intercept           0.00131      0.0600 -
 2 b_Alcohol_consumption 0.000601     0.188  -
 3 b_Adult_mortality     0.000341     0.147  -
 4 b_Hepatitis_B         0.000472     0.101  -
 5 b_Measles             0.000343     0.139  -
 6 b_BMI                 0.000212     0.101  -
 7 b_Polio               0.000191     0.110  -
 8 b_Diphtheria          0.000436     0.111  -
 9 b_Incidents_HIV       0.000499     0.131  -
10 b_GDP_per_capita      0.000834     0.108  -
# ℹ 27 more rows
```

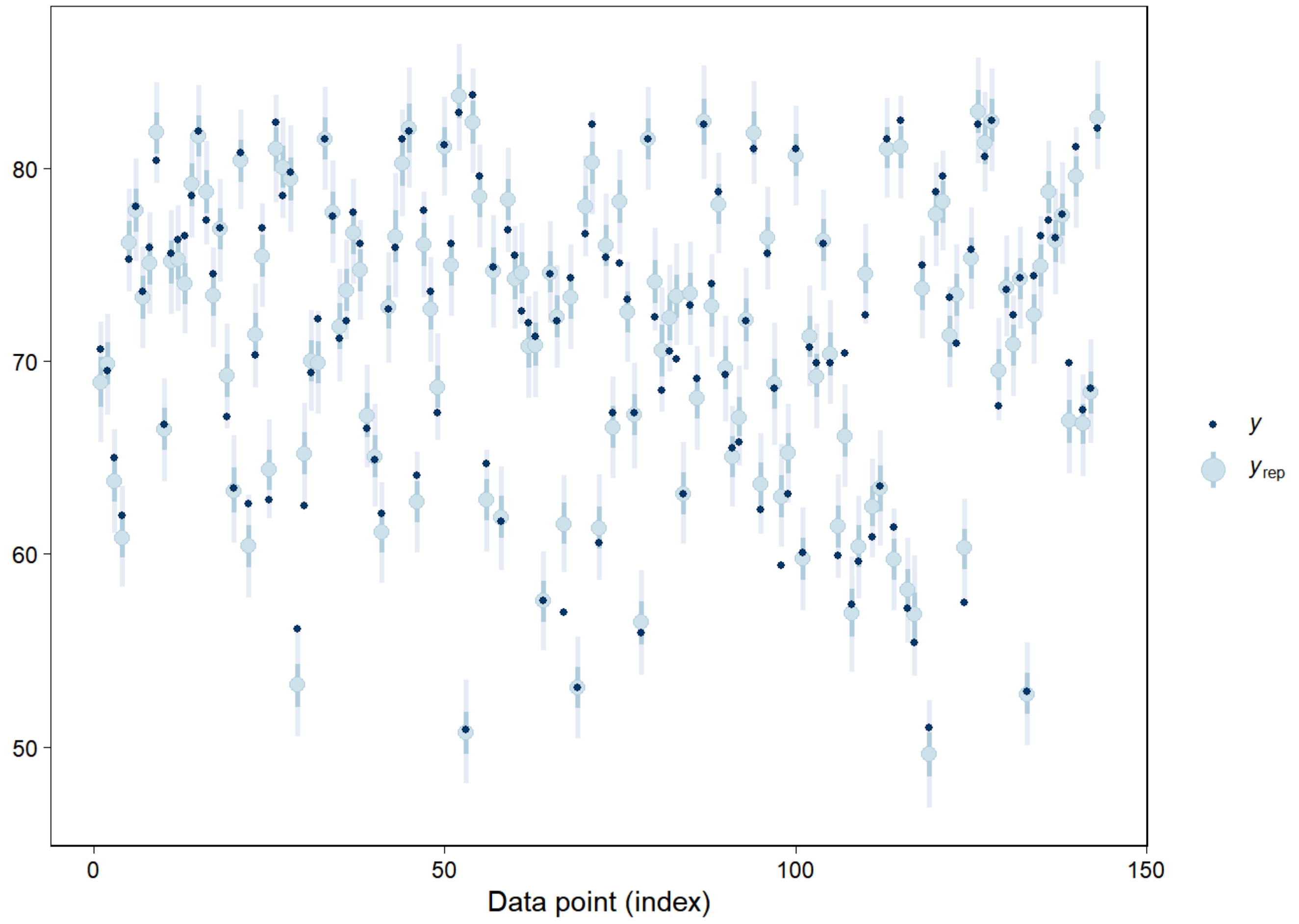
```
powerscale_sensitivity(m.5)
```

```
Sensitivity based on cjs_dist:
# A tibble: 38 × 4
   variable                prior likelihood diagnosis
   <chr>                   <dbl>      <dbl> <chr>    
 1 b_Intercept           0.00160     0.0499 -        
 2 b_Alcohol_consumption 0.00559     0.166  -        
 3 b_Adult_mortality     0.00196     0.140  -        
 4 b_Hepatitis_B         0.00319     0.0969 -        
 5 b_Measles             0.00202     0.137  -        
 6 b_BMI                 0.00156     0.119  -        
 7 b_Polio               0.00225     0.133  -        
 8 b_Diphtheria          0.00201     0.111  -        
 9 b_Incidents_HIV       0.00353     0.115  -        
10 b_GDP_per_capita      0.00961     0.0971 -        
# ℹ 28 more rows
```

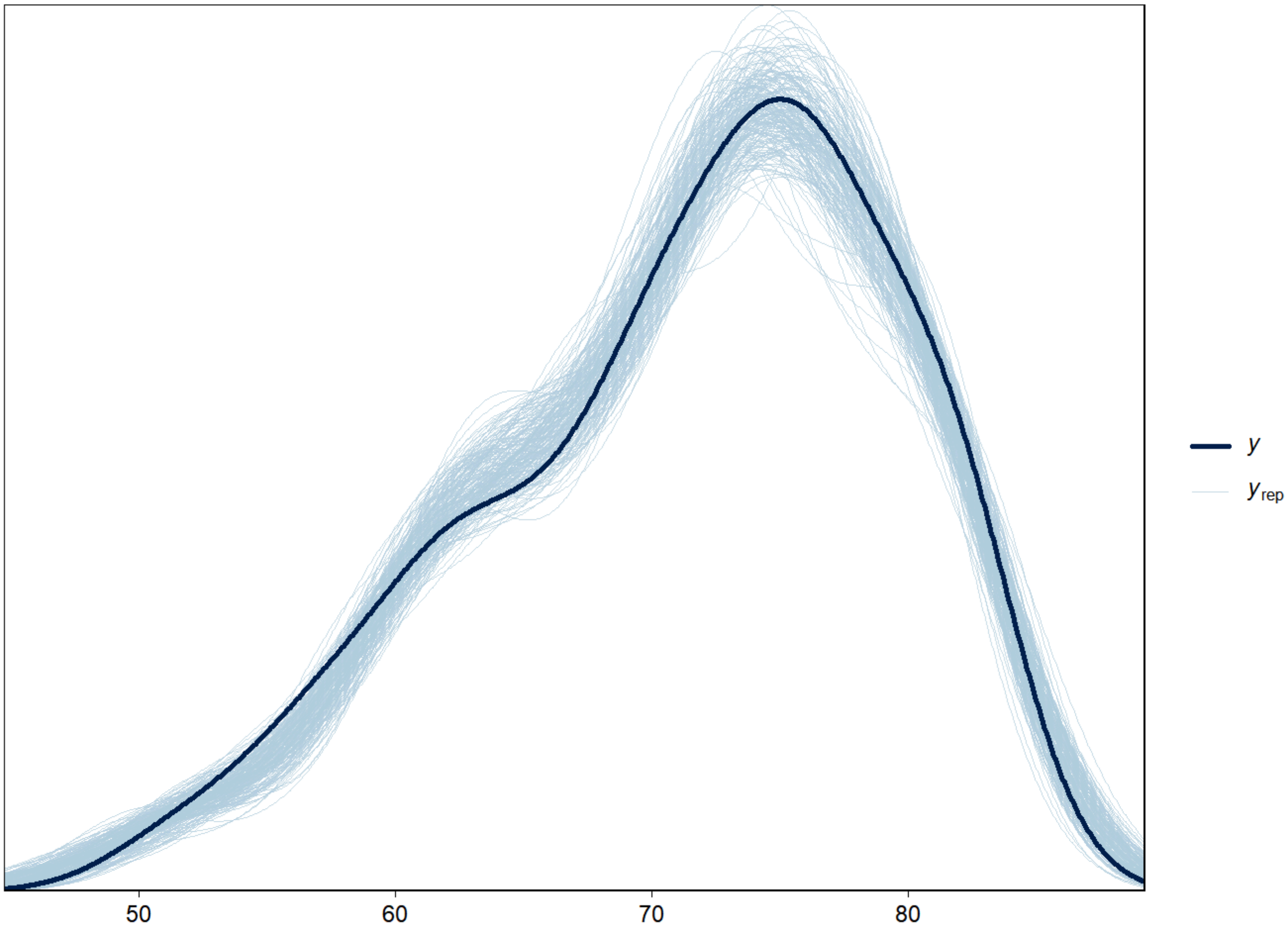
The powerscale sensitivities indicate no errors for any of the 5 models.

b. Conduct posterior predictive checks for each model to assess how well they fit the data. Explain what you conclude.

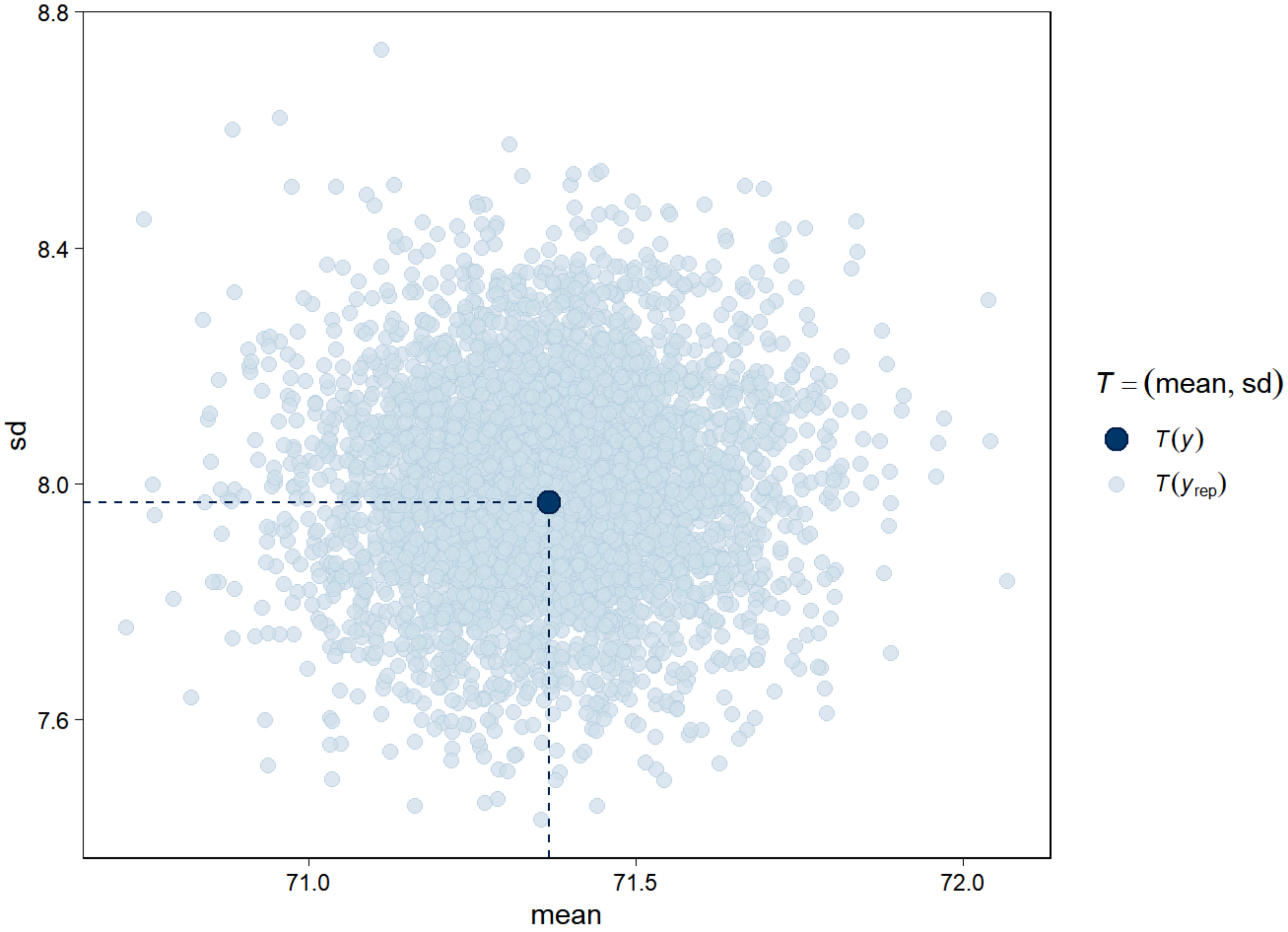
```
# Model 1: Random Intercept for Region (Giyanto)
print(pp_check(m.1, type = "intervals"))
```



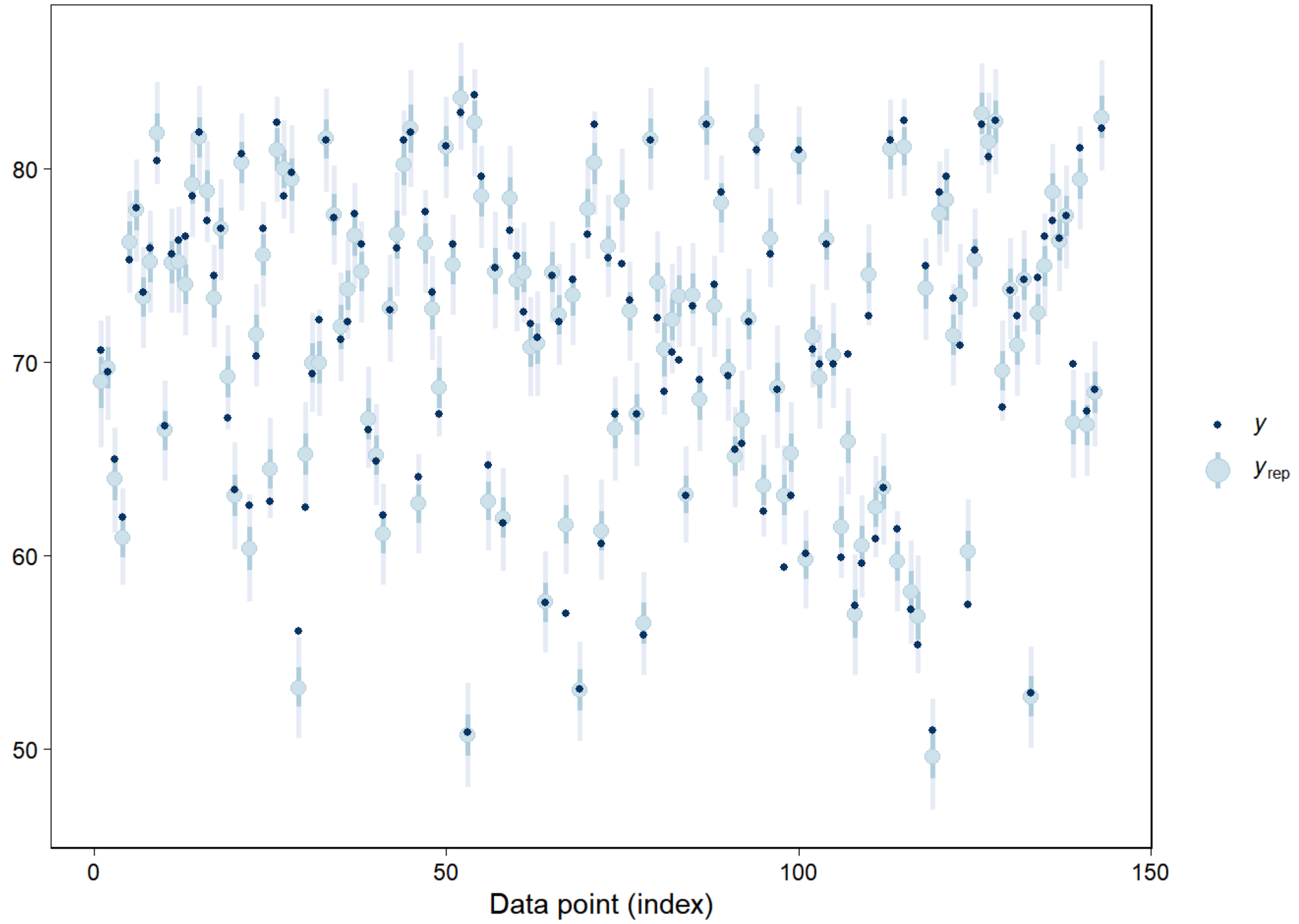
```
print(pp_check(m.1, ndraws = 200))
```



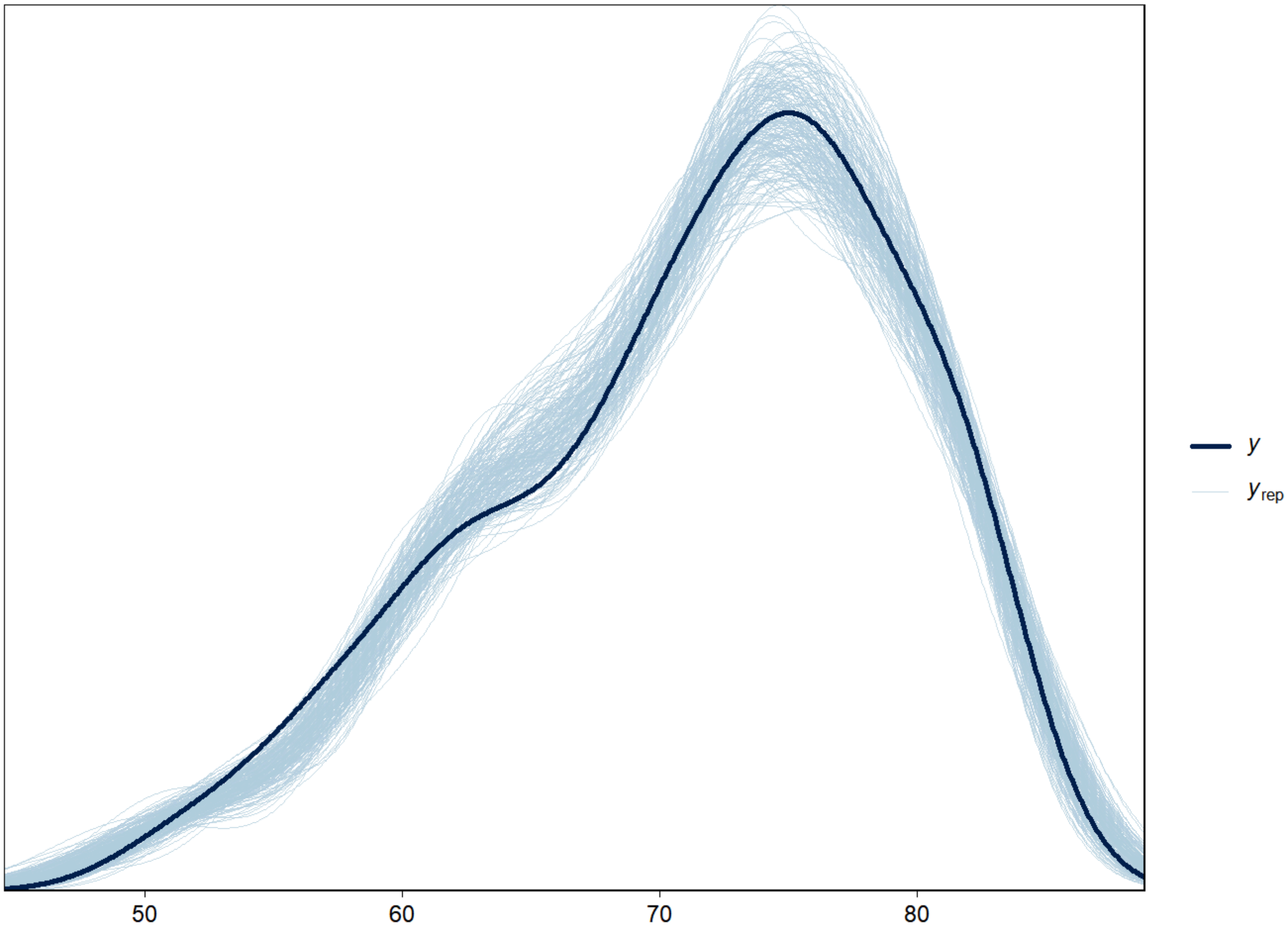
```
print(pp_check(m.1, type = "stat_2d"))
```



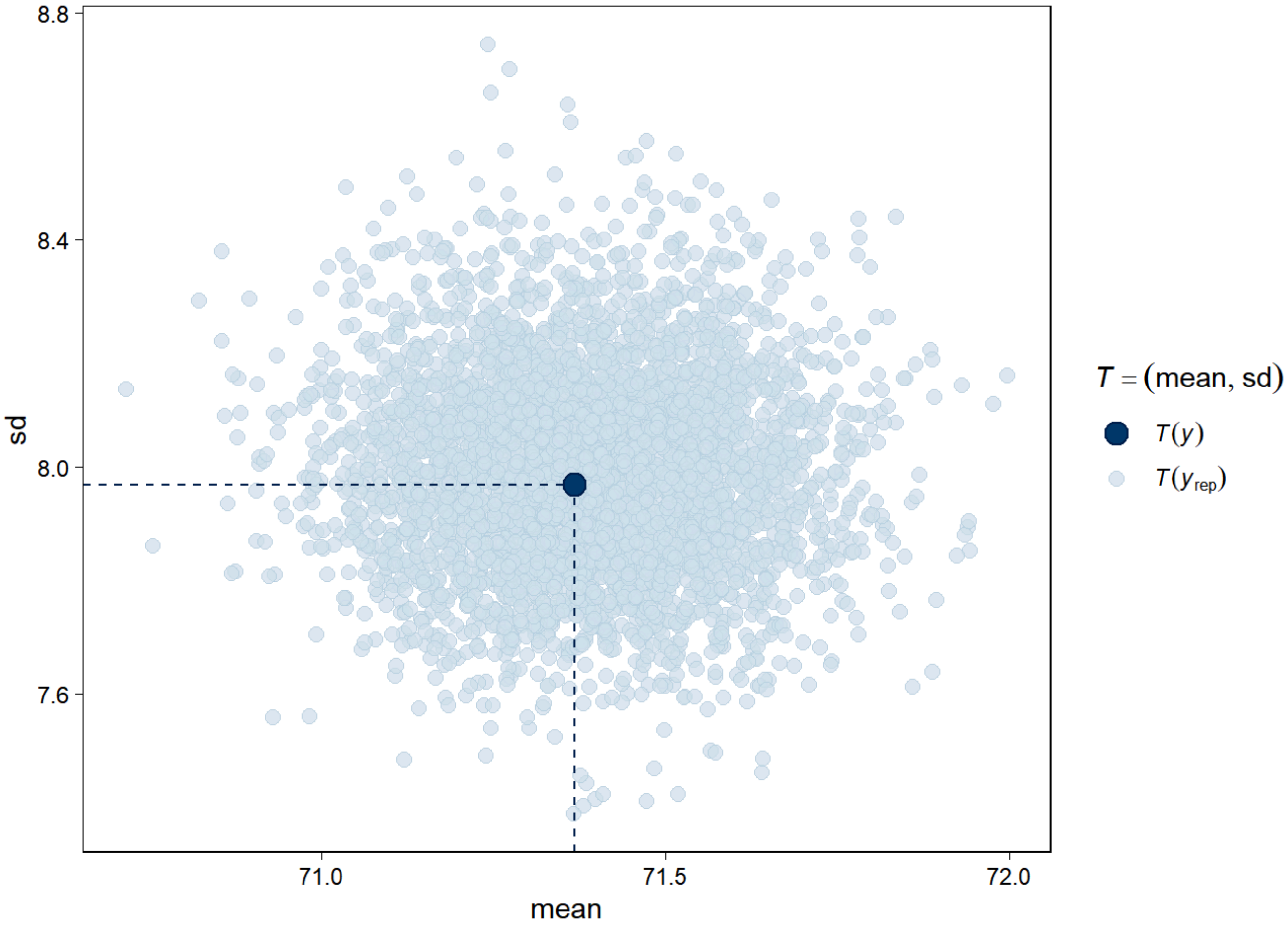
```
# Model 2: Random Intercept for Region with informed priors & Student's T dist
ribution (Fynn)
print(pp_check(m.2, type = "intervals"))
```



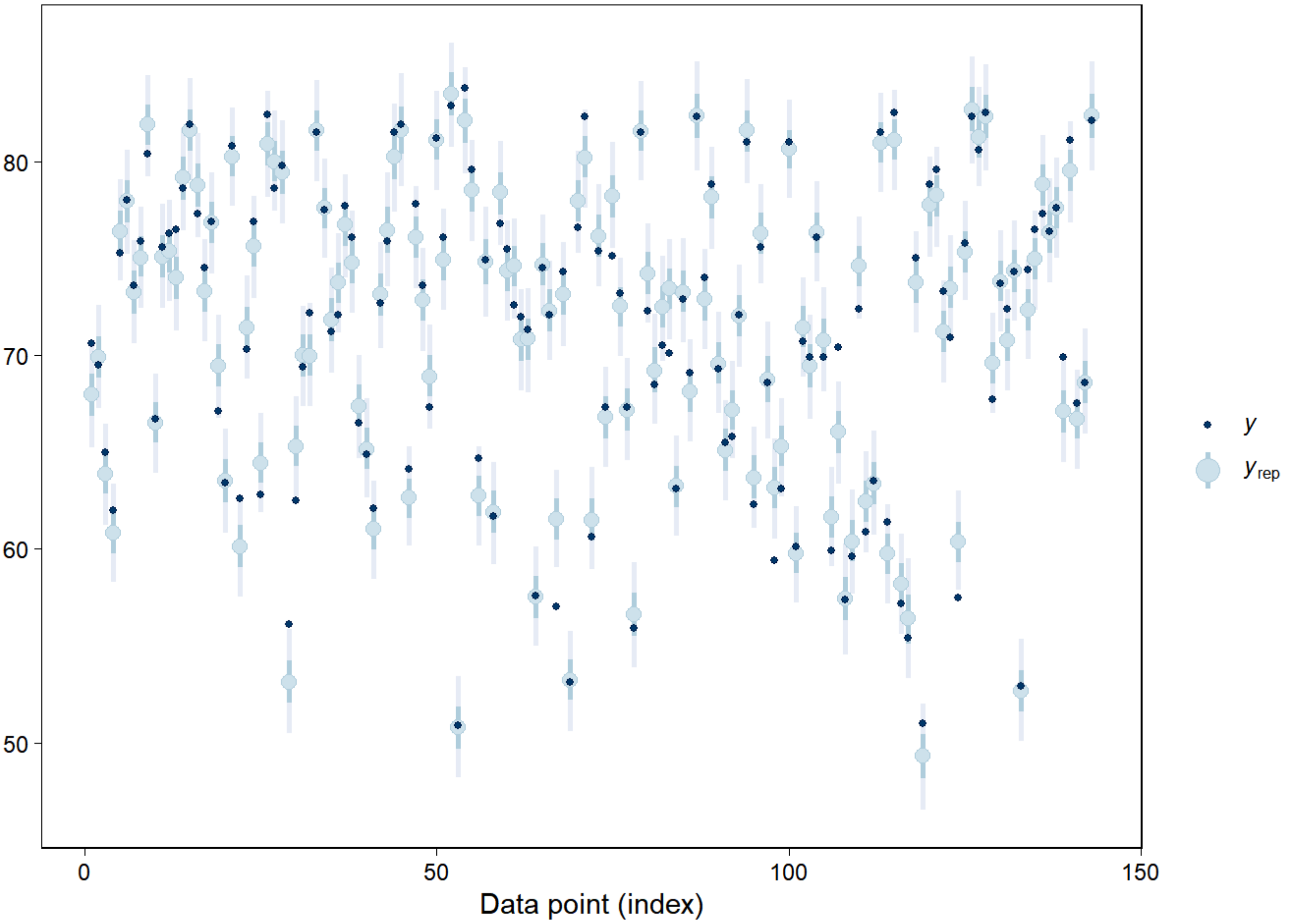
```
print(pp_check(m.2, ndraws = 200))
```



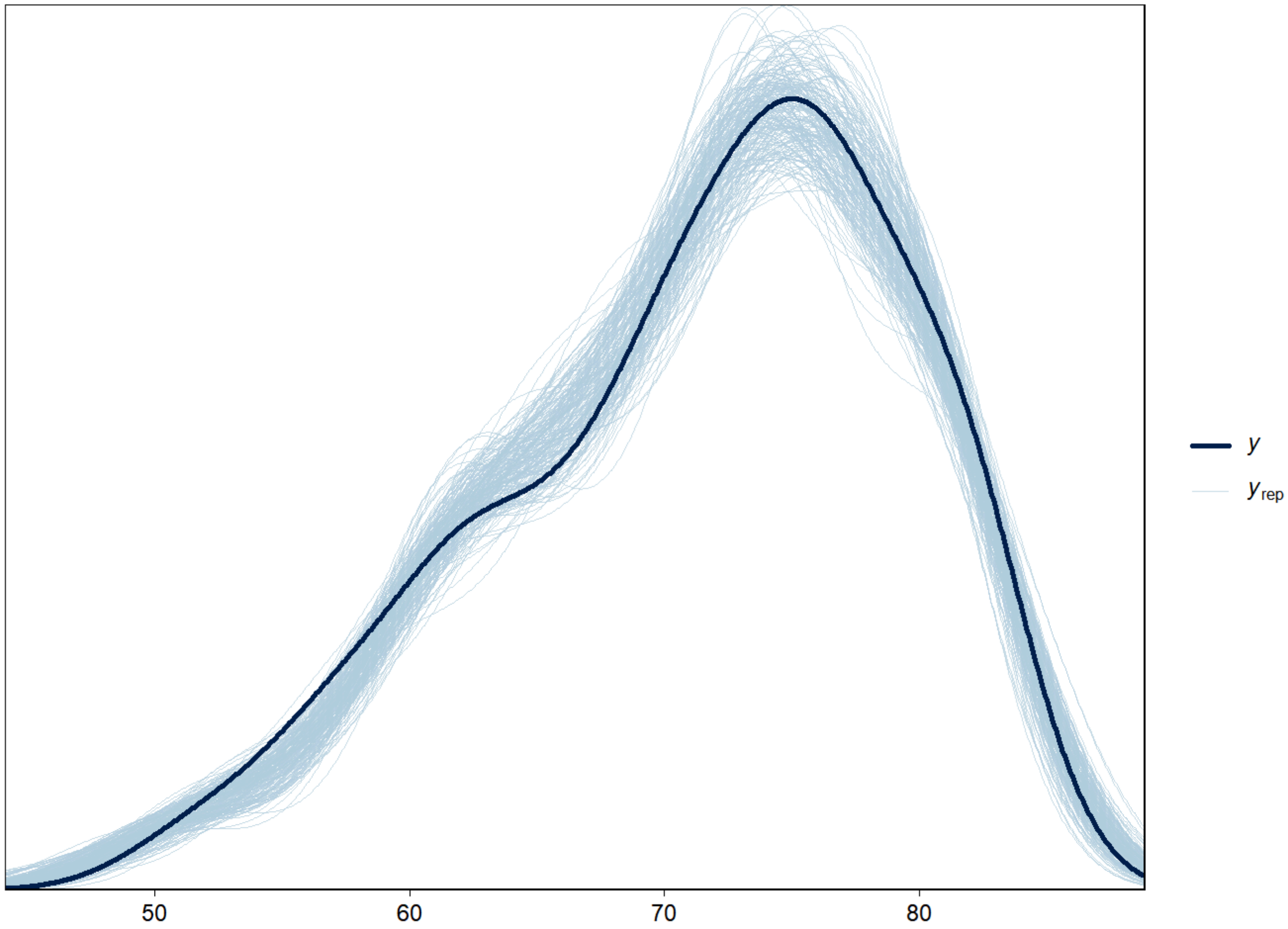
```
print(pp_check(m.2, type = "stat_2d"))
```



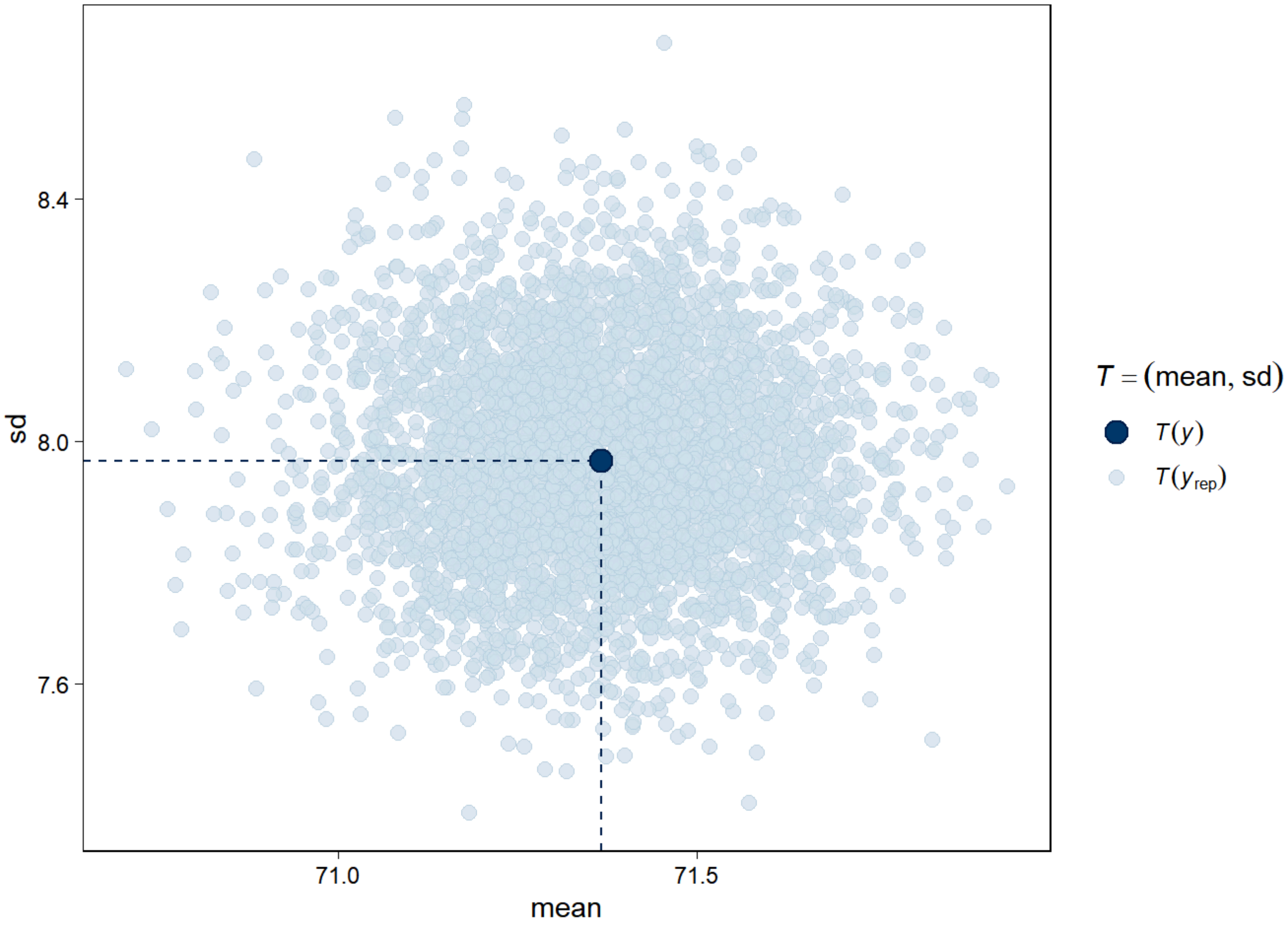
```
# Model 3: Random Intercept for Region with Horseshoe priors(Evi)
print(pp_check(m.3, type = "intervals"))
```



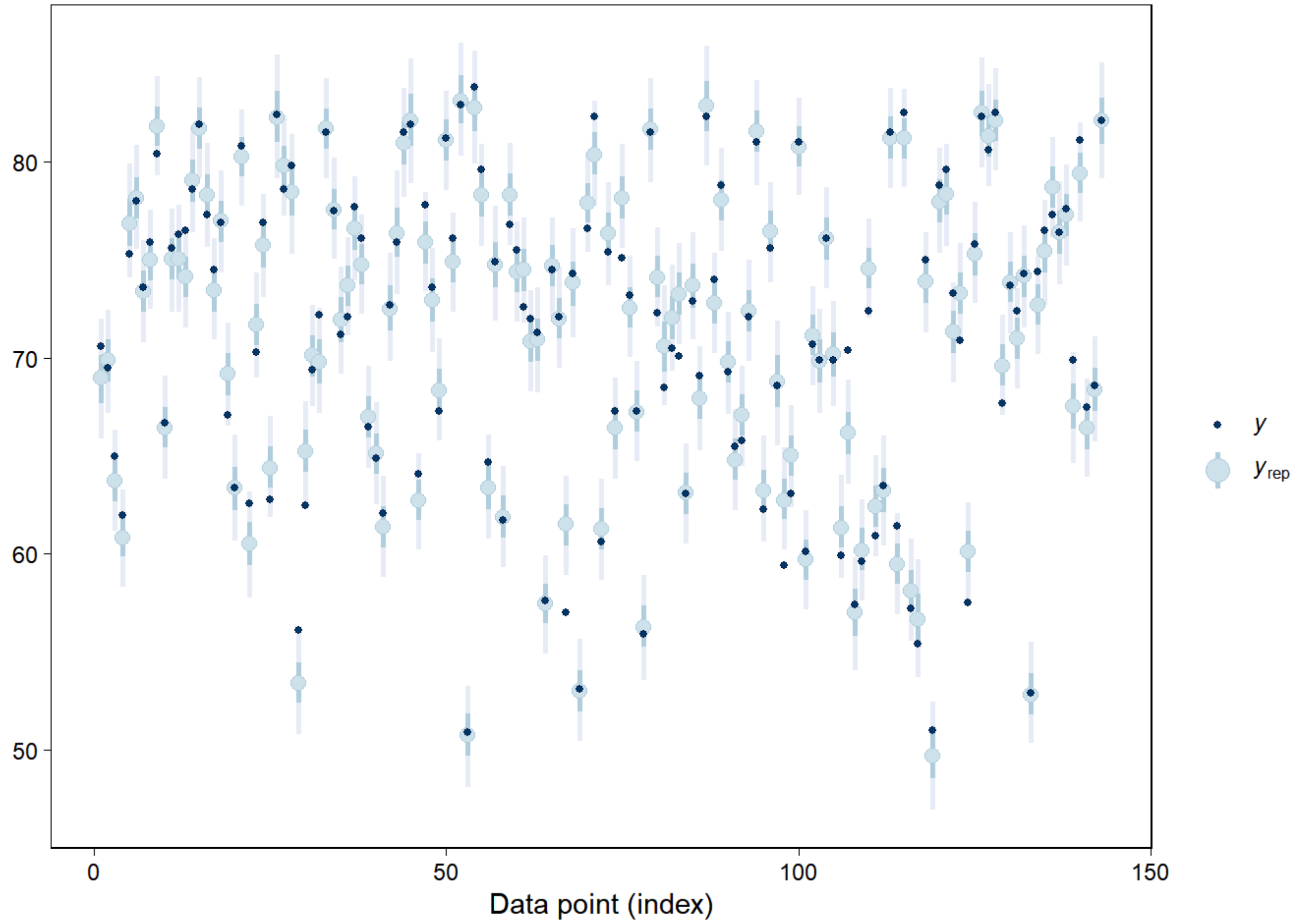
```
print(pp_check(m.3, ndraws = 200))
```



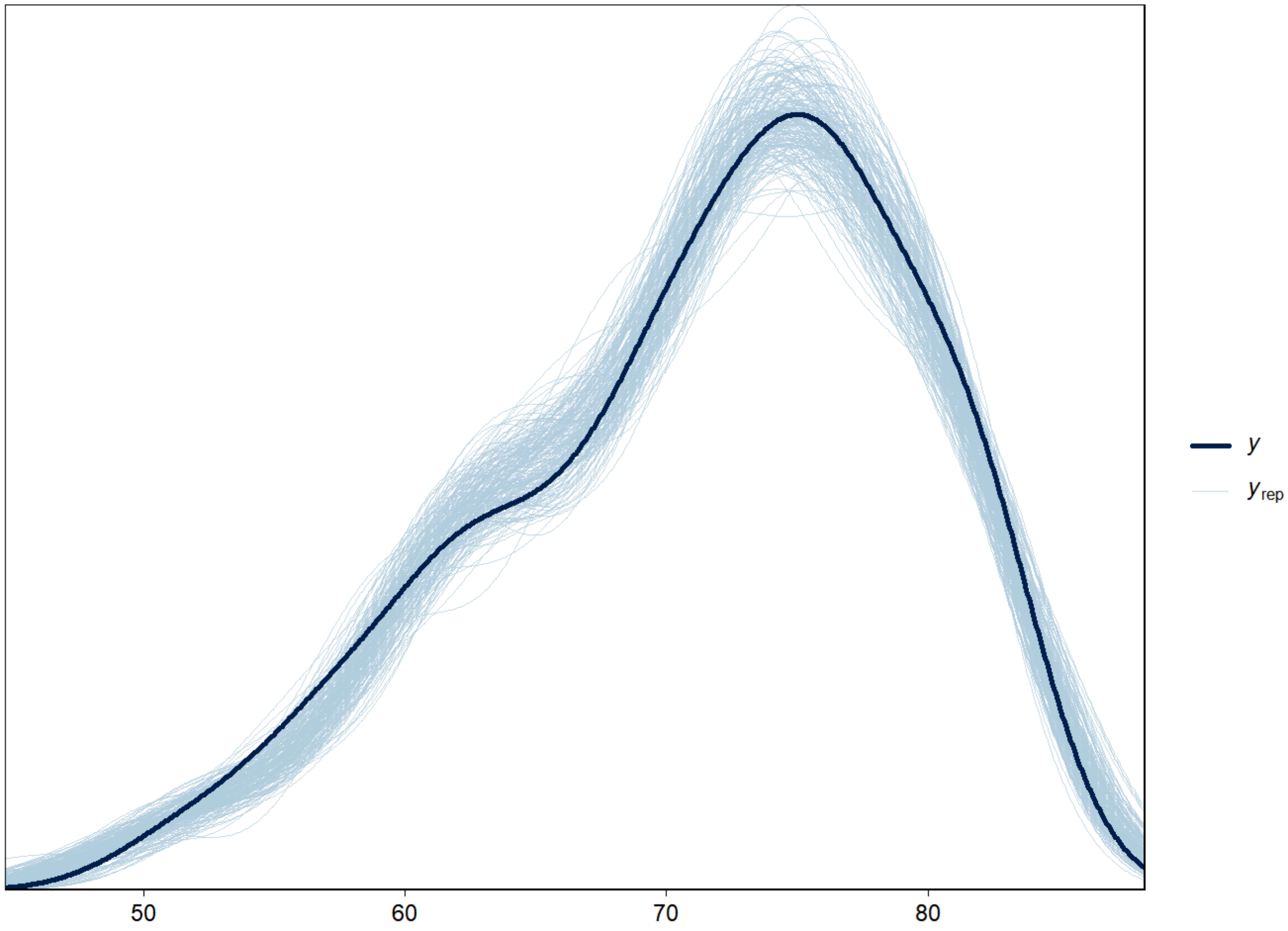
```
print(pp_check(m.3, type = "stat_2d"))
```



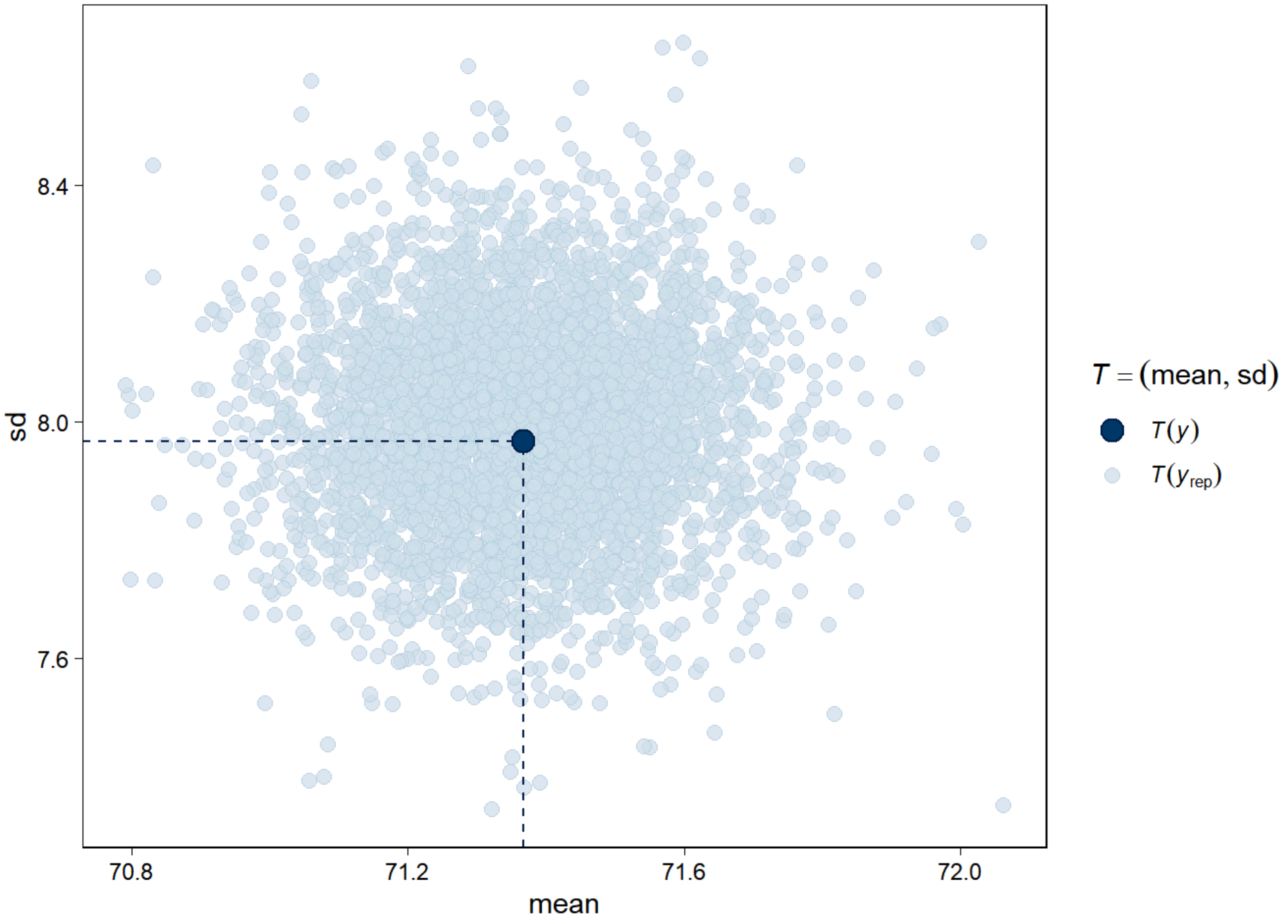
```
# Model 4: Varying intercept and varying slopes model with Gelman priors (Sam)
print(pp_check(m.4, type = "intervals"))
```



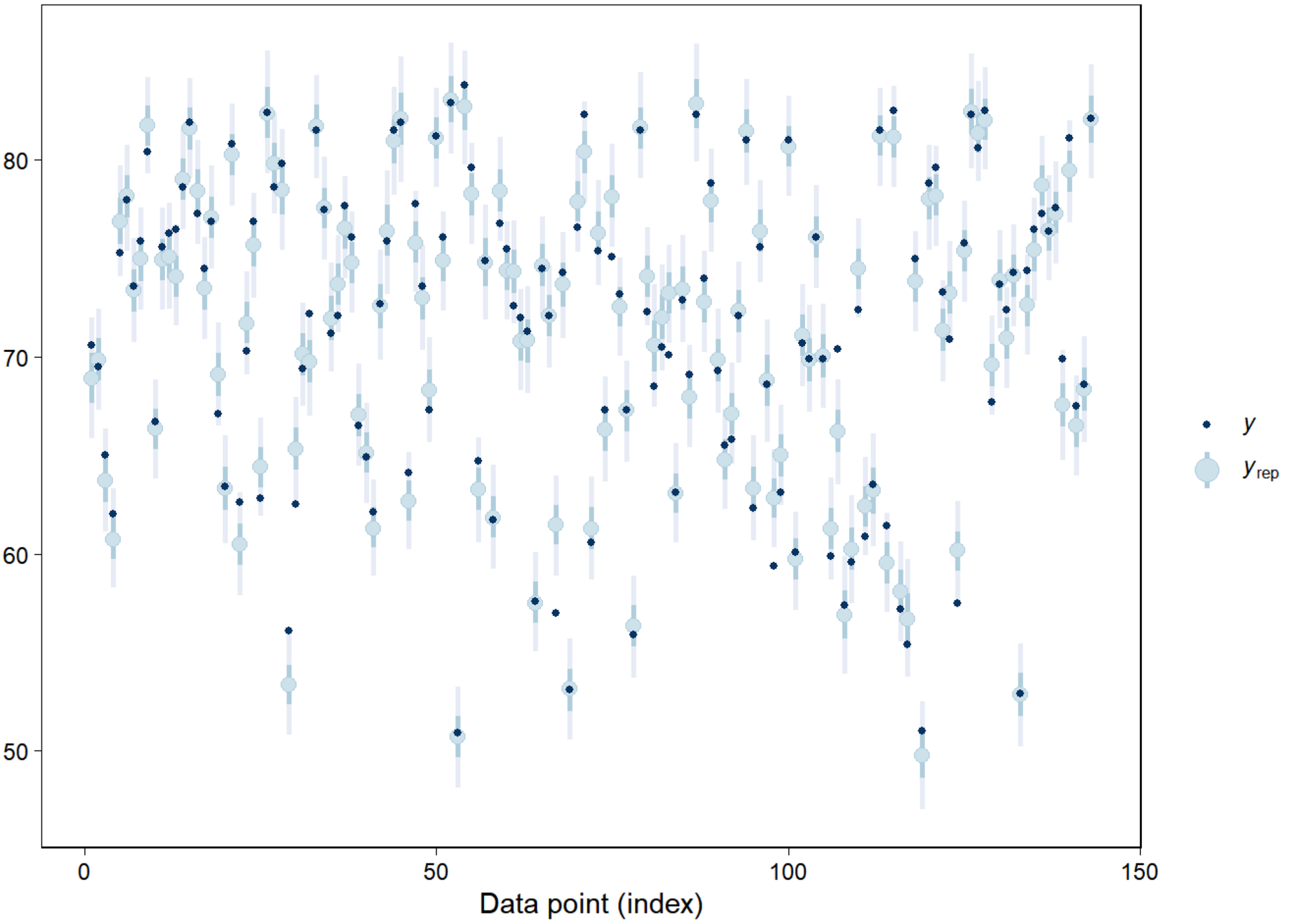
```
print(pp_check(m.4, ndraws = 200))
```



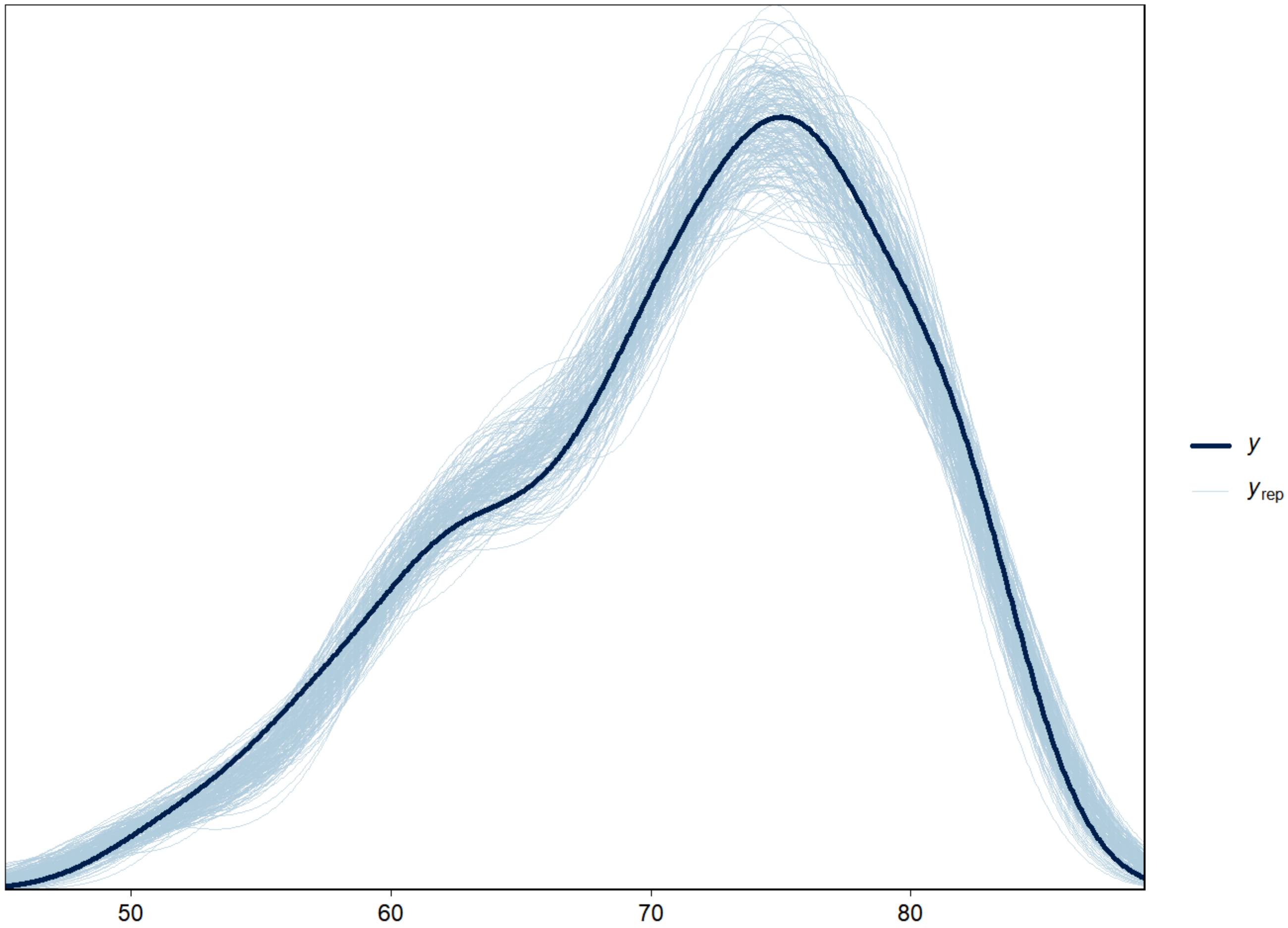
```
print(pp_check(m.4, type = "stat_2d"))
```



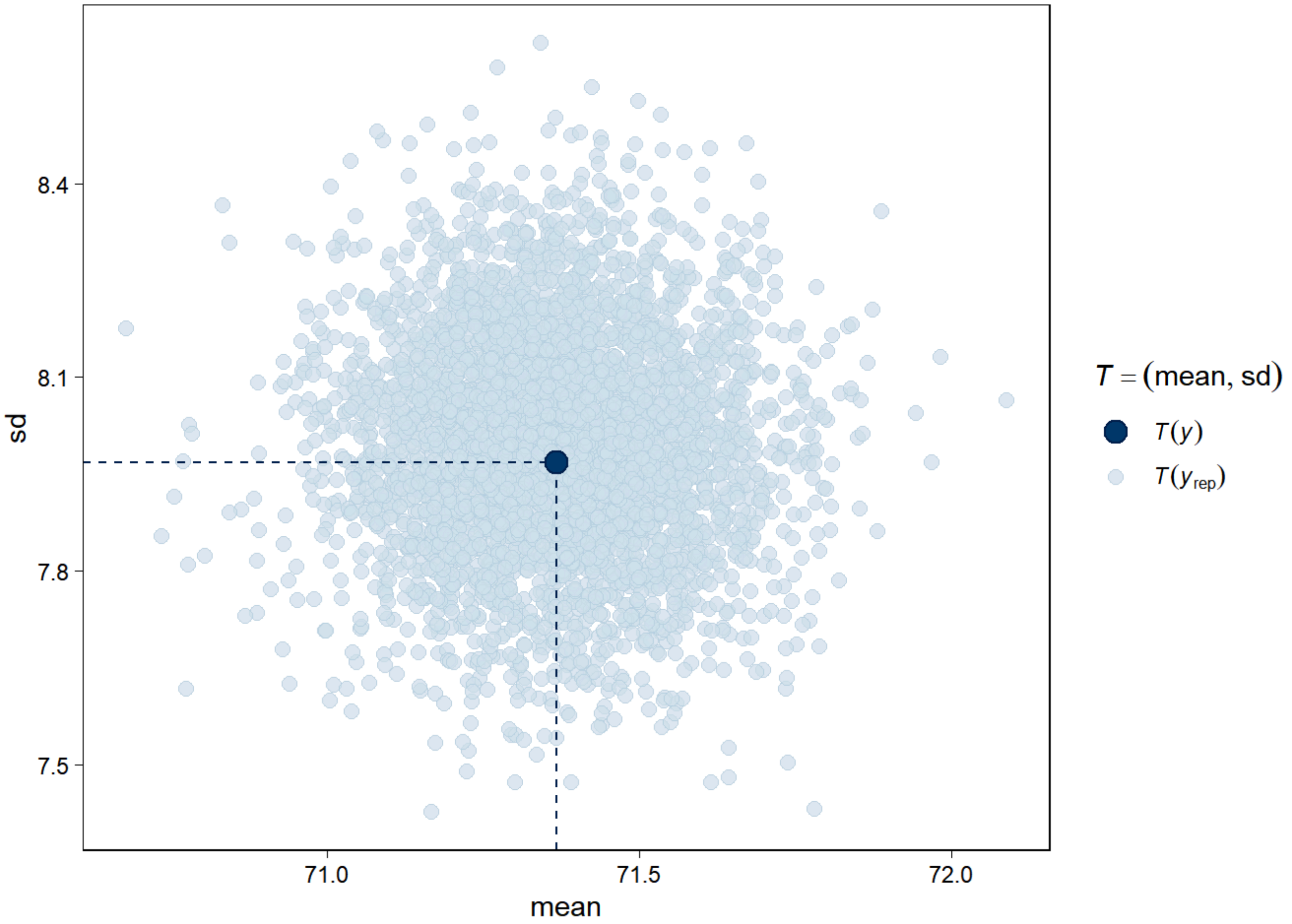
```
# Model 5: Correlated varying intercept & varing slopes model (Emma)
print(pp_check(m.5, type = "intervals"))
```



```
print(pp_check(m.5, ndraws = 200))
```



```
print(pp_check(m.5, type = "stat_2d"))
```



4.1 Model 1: Random Intercept for Region (Giyanto)

The posterior predictive checks for Model 1 (Random Intercept for Region) indicate that the model fits the data well. The intervals plot shows that most observed data points fall within the 95% credible intervals, capturing the variability in the data. The density overlay plot demonstrates a good match between the observed and simulated densities, suggesting the model accurately represents the overall distribution of life expectancy. The scatter plot of summary statistics confirms that the model's predictions align with the observed data in terms of both the mean and standard deviation. Therefore, this model is suitable for further analysis and inference.

4.2 Model 2: Random intercept for Region | Priors based on EDA & Student distribution (Fynn)

When comparing to the previous model, the posterior predictive checks seem to indicate a slight positive effect. A large majority of the data remains in the 95% CI, the density plot line seems to follow the simulated densities even closer And the scatter plot also indicates a very central point. As such, these model changes show positive progress, although a higher risk of overfitting needs to be considered when making major standard deviation changes to multiple variables.

4.3 Model 3: Random intercept for Region | Horseshoe priors (Evi)

Conducting predictive checks for the third model indicates a generally good fit to the data. Upon closer examination of the interval plot, it becomes evident that most data points fall within the 95% credible intervals. However, as some outliers appear on the plot, it is important to be mindful of their potential impact. The density overlay plot demonstrates consistency between the model's predictions and the observed data distributions, suggesting an accurate model fit. Furthermore, the scatter plot reveals that the observed points are well within the predictive cloud, indicating that the model adequately captures both the central tendency and variability of the data.

4.4 Model 4: Varying intercept and varying slopes model | Gelman priors (Sam)

By analysing the posterior plots, it is evident that the model is pretty accurate at predicting the life expectancy in each country. The predictive distrubutions follow the shape of the posterior distribution of life expectancy closely. Still there is some significant standard deviation presents in the predictions. This is further explained by the predictions per country in the dataset. Some of the countries are predicted well within the interval, while other prediction are completely different from the actual value. By the analysis of the scatter plot we see that in general, the predicted average is about the same as the actual mean life expectancy. The same can be said for the standard deviation. Still, there are some outliers in the data, which implies that there might still be some room for the model to improve.

4.5 Model 5: Correlated varying intercept & varing slopes model (Emma)

Model 5 shows no clear differences in terms of performance in comparison to other models, indicating a generally similar performance. The interval plot shows most values fall into the 95% credible intervals, the density overlay plot indicates some variability, but is for the most part consistent. The scatter plot once again indicates most observations to be centered, albeit with a few outliers

5. Model Comparison (1.5pt)

a. Use k-fold cross-validation to compare the models.

```
plan(multisession)

# Model 1
k <- loo::kfold_split_random(K = 10, N = nrow(life_expectancy_train))
kf.m.1 <- kfold(m.1, chains = 1, folds = k, save_fits = T)
kf.m.1
```

```


Based on 10-fold cross-validation.

           Estimate   SE
elpd_kfold   -281.2 10.4
p_kfold        27.2  4.8
kfoldic       562.4 20.7
```

```
# Model 2
k <- loo::kfold_split_random(K = 10, N = nrow(life_expectancy_train))
kf.m.2 <- kfold(m.2, chains = 1, folds = k, save_fits = T)
kf.m.2
```

```


Based on 10-fold cross-validation.

           Estimate   SE
elpd_kfold   -275.6  9.8
p_kfold        23.0  4.0
kfoldic       551.2 19.5
```

```
# Model 3
k <- loo::kfold_split_random(K = 10, N = nrow(life_expectancy_train))
kf.m.3 <- kfold(m.3, chains = 1, folds = k, save_fits = T)
kf.m.3
```

```


Based on 10-fold cross-validation.

           Estimate   SE
elpd_kfold   -269.9  9.2
p_kfold        15.3  2.2
kfoldic       539.7 18.4
```

```
# Model 4
kf.m.4 <- kfold(m.4, chains = 1, folds = k, save_fits = T)
kf.m.4
```

```


Based on 10-fold cross-validation.

           Estimate   SE
elpd_kfold   -272.9  9.0
p_kfold        24.1  3.2
kfoldic       545.7 17.9
```

```
# Model 5
k <- loo::kfold_split_random(K = 10, N = nrow(life_expectancy_train))
kf.m.5 <- kfold(m.5, chains = 1, folds = k, save_fits = T)
kf.m.5
```

```


Based on 10-fold cross-validation.

           Estimate   SE
elpd_kfold   -277.2  9.5
p_kfold        28.1  4.0
kfoldic       554.5 19.1
```

```
plan(sequential)
```

b. Determine the best model based on predictive accuracy and justify your decision.

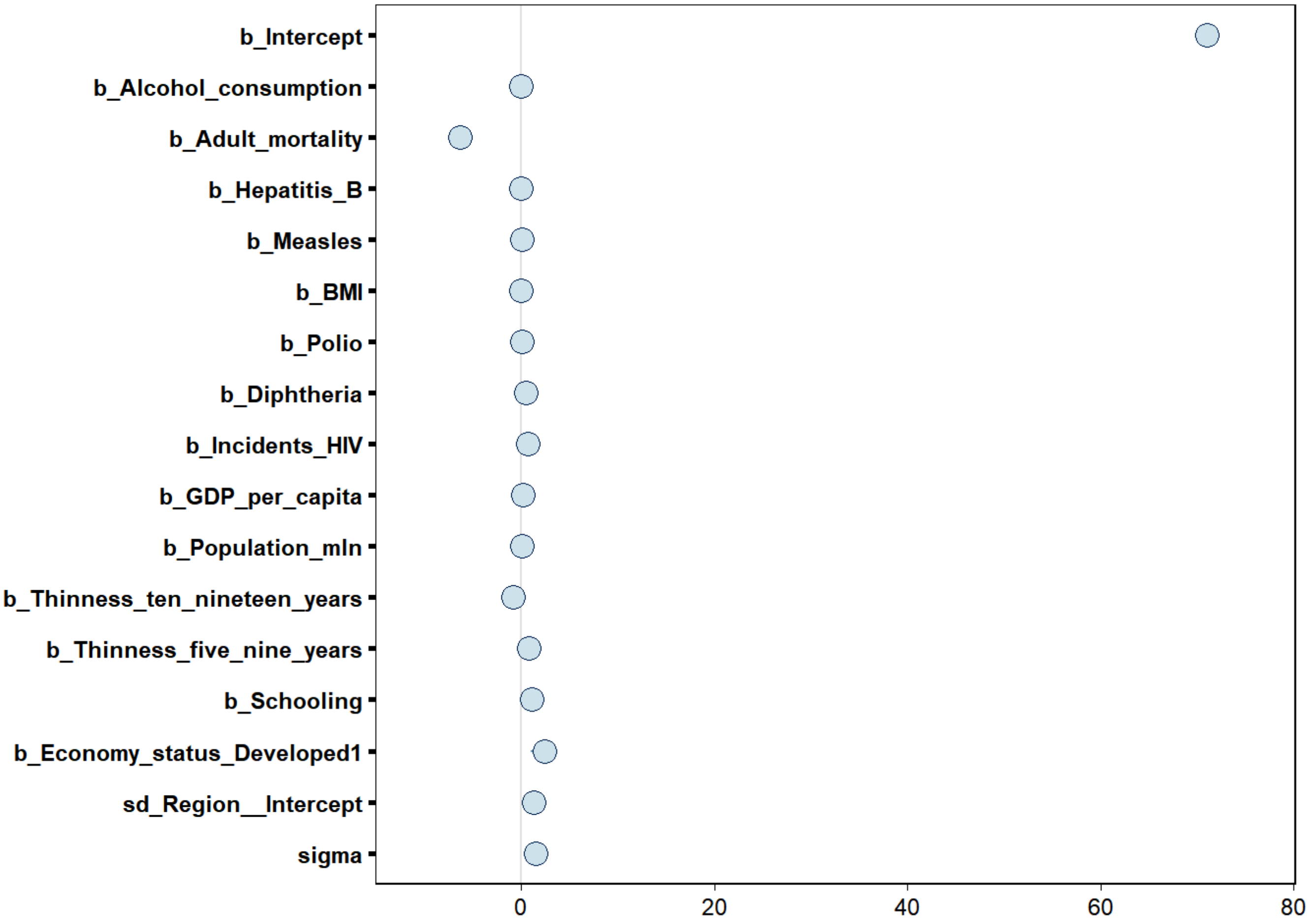
```
loo_compare(kf.m.1,kf.m.2, kf.m.3, kf.m.4, kf.m.5)
```

```
    elpd_diff se_diff
m.3   0.0       0.0  
m.4  -3.0       3.7  
m.2  -5.7       4.3  
m.5  -7.4       4.3  
m.1 -11.3       4.7  
```

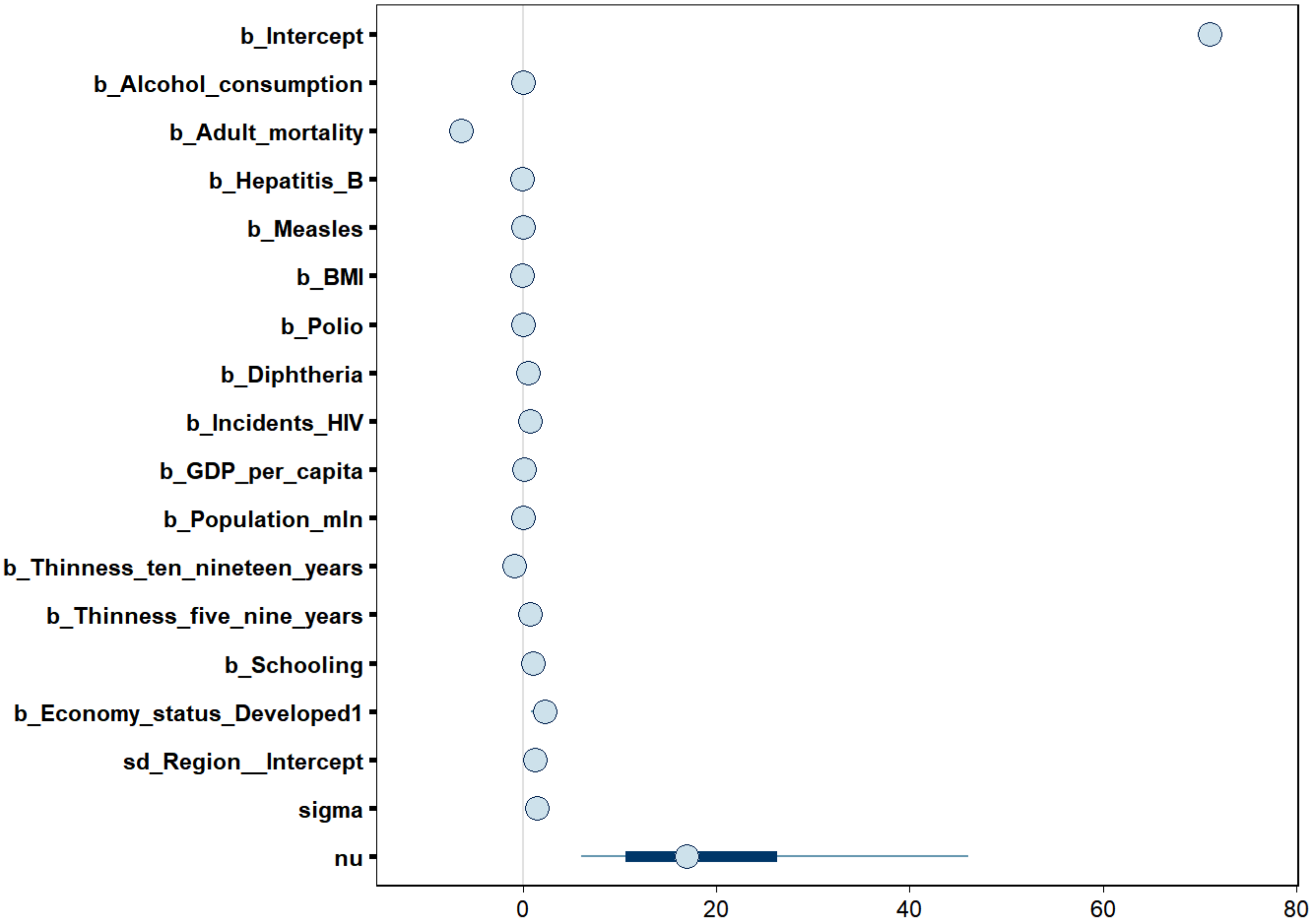
Based on the loo_compare results and the cross-validated log-score rule, Model 3 is the best model with the highest ELPD. The difference between Model 3 and Model 1 is significant and reliable, with an elpd_diff of 11.3 compared to 2×se_diff of 9.4. Differences between Model 3 and Models 2, 4, and 5 are not statistically significant. Therefore, we conclude that Model 3 is the best model due to its significant improvement over Model 1 and the lack of reliable differences from Models 2, 4, and 5.

6. Interpretation of Important Parameters (1.5pt)

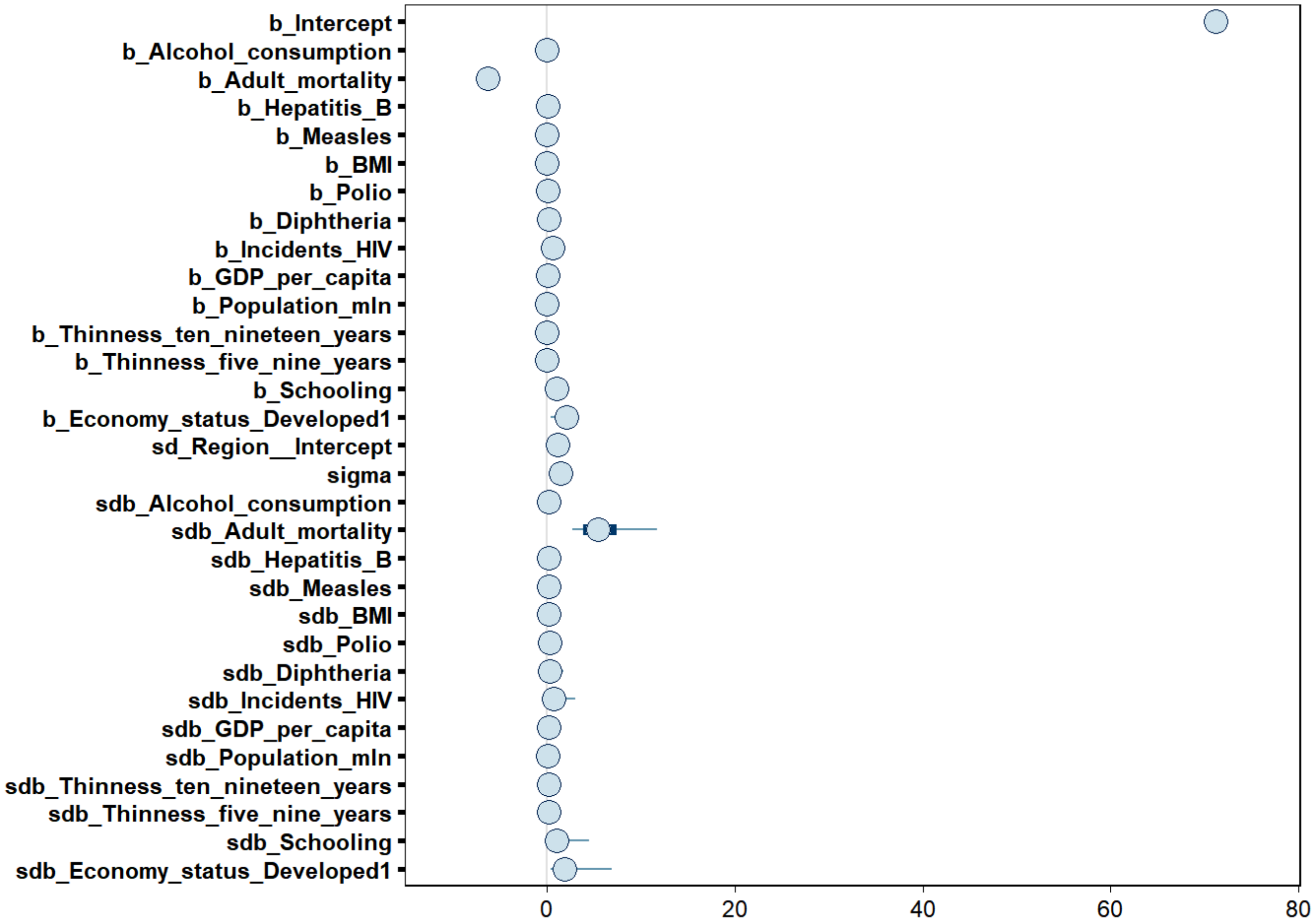
```
mcmc_plot(m.1)
```



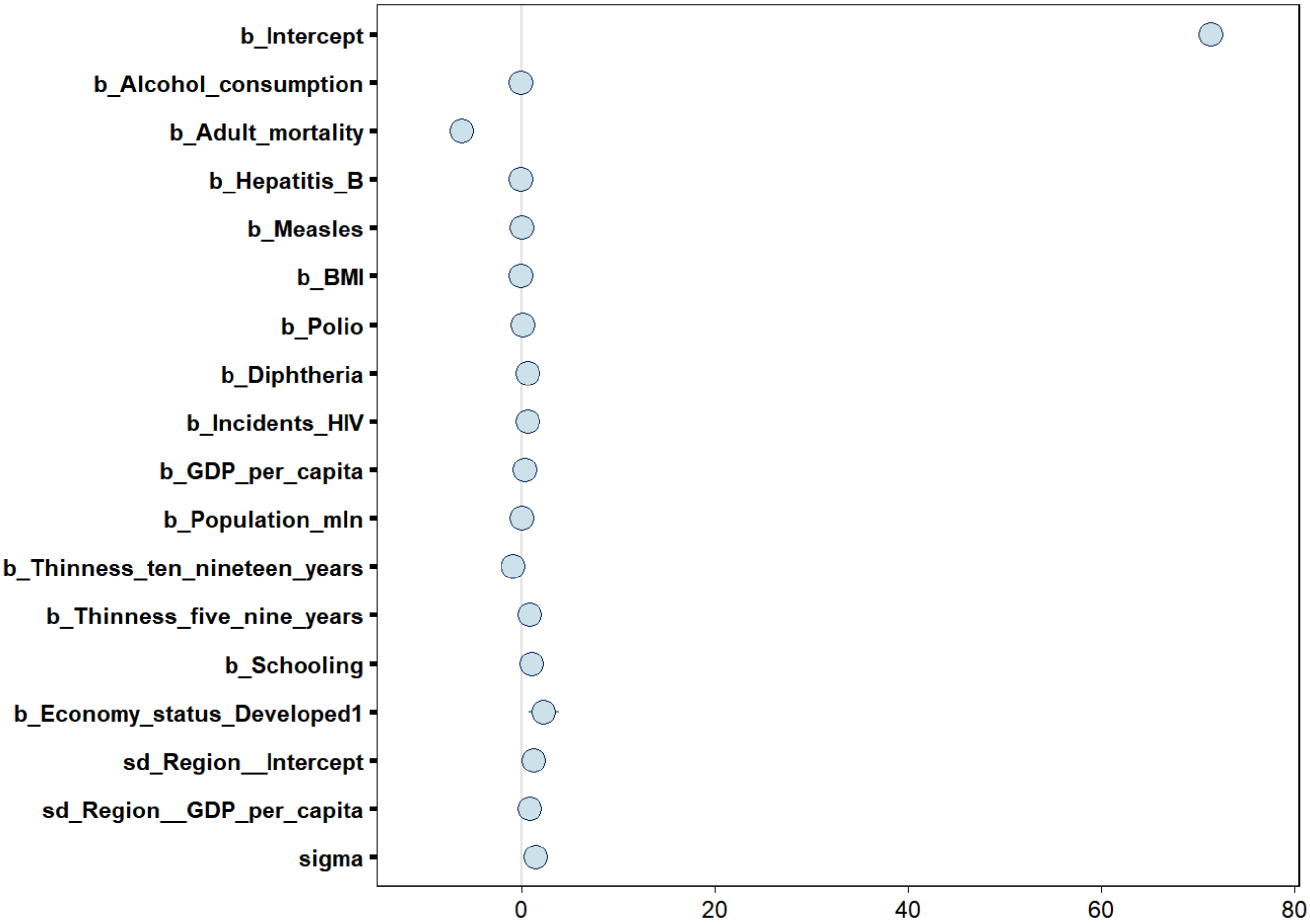
```
mcmc_plot(m.2)
```


b_Intercept
b_Alcohol_consumption
b_Adult_mortality
b_Hepatitis_B
b_Measles
b_BMI
b_Polio
b_Diphtheria
b_Incidents_HIV
b_GDP_per_capita
b_Population_mln
b_Thinness_ten_nineteen_years
b_Thinness_five_nine_years
b_Schooling
b_Economy_status_Developed1
sd_Region__Intercept
sigma
nu
0
20
40
60
80

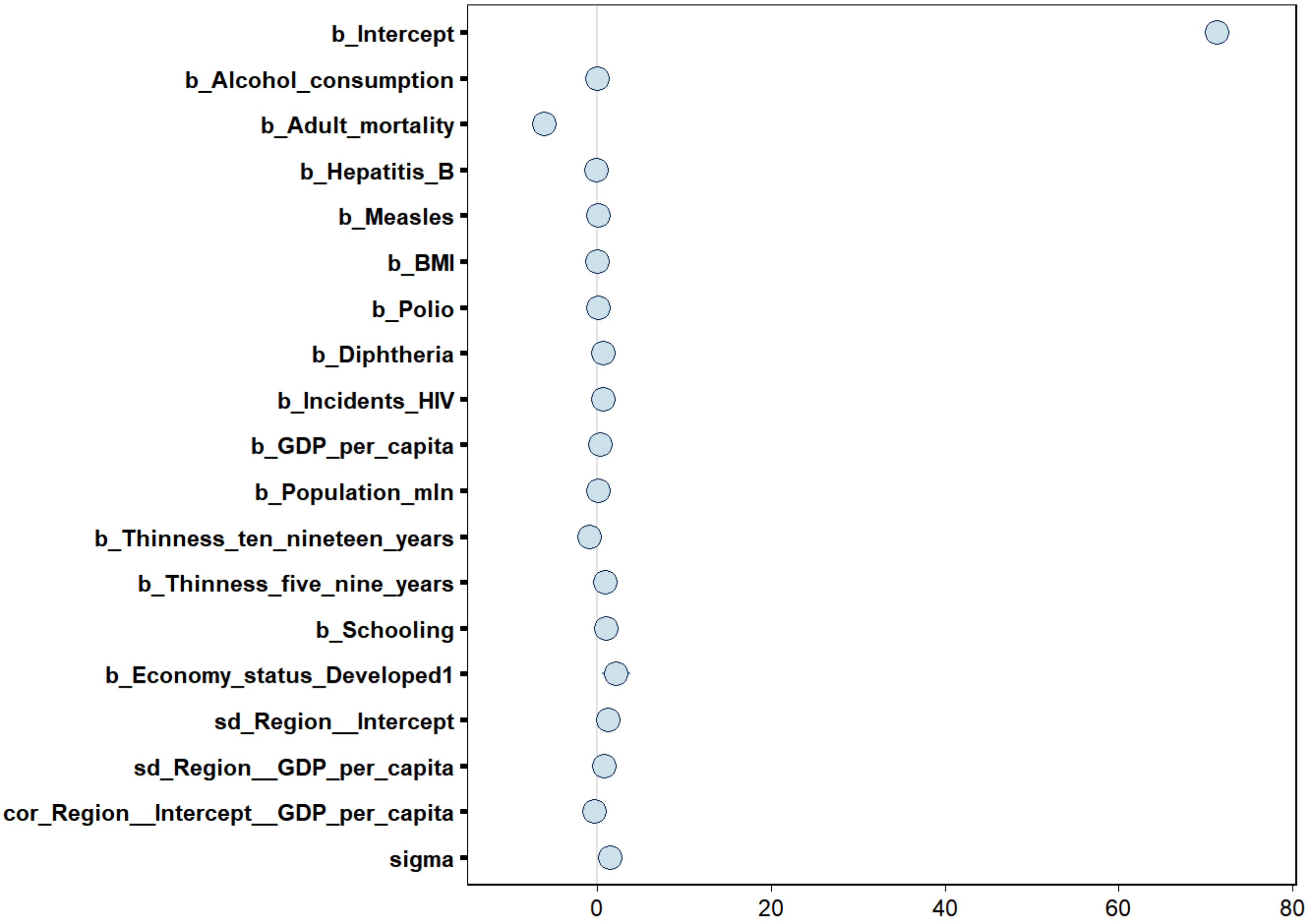
```
mcmc_plot(m.3)
```


b_Intercept
b_Alcohol_consumption
b_Adult_mortality
b_Hepatitis_B
b_Measles
b_BMI
b_Polio
b_Diphtheria
b_Incidents_HIV
b_GDP_per_capita
b_Population_mln
b_Thinness_ten_nineteen_years
b_Thinness_five_nine_years
b_Schooling
b_Economy_status_Developed1
sd_Region__Intercept
sigma
sdb_Alcohol_consumption
sdb_Adult_mortality
sdb_Hepatitis_B
sdb_Measles
sdb_BMI
sdb_Polio
sdb_Diphtheria
sdb_Incidents_HIV
sdb_GDP_per_capita
sdb_Population_mln
sdb_Thinness_ten_nineteen_years
sdb_Thinness_five_nine_years
sdb_Schooling
sdb_Economy_status_Developed1
0
20
40
60
80

```
mcmc_plot(m.4)
```


b_Intercept
b_Alcohol_consumption
b_Adult_mortality
b_Hepatitis_B
b_Measles
b_BMI
b_Polio
b_Diphtheria
b_Incidents_HIV
b_GDP_per_capita
b_Population_mln
b_Thinness_ten_nineteen_years
b_Thinness_five_nine_years
b_Schooling
b_Economy_status_Developed1
sd_Region__Intercept
sd_Region__GDP_per_capita
sigma
0
20
40
60
80

```
mcmc_plot(m.5)
```



When looking at the MCMC plots, we can find a few patterns across all models:

Adult_mortality is (unsurprisingly) consistently the strongest negative predictor of life expectancy. On the other hand, the level of economic development of a country is generally the strongest positive indicator. Beyond that, schooling and thinness in early childhood (5-9) are the next strongest positive factors. Interestingly, while thinness from 5-9 is a positive factor, thinness from 10-19 is the second strongest negative factor.

Comparing the cross-validation scores of all five models, we find rather small differences overall, but there are some tendencies:

A special note to be added: When beginning model exploration, the initial model 1 omitted the variable Adult_mortality, which was added to all models after being found to be strongly correlated through the research for model 2, which was the strongest model by far at that point in time. As such, this oversight confirmed that Adult_mortality is indeed an important variable to include in the model.

7. Report a loss function on the test set (Optional for bonus 0.5 to 1pt, depending on if you use RMSE or another function).

Report RMSE or other loss (or utility) function on the test set. (Transform it back if necessary).

```
# Loss function RMSE
rmse <- function(y, yrep){
  # We summarize our distribution of predictions with
  # a point prediction. In this case the average.
  # The distribution of predictions for each
  # out of sample observation is in the columns of yrep.
  # Or more accuraretly, samples (iter/2* nchains) from
  # the distribution of predictions
  yrep_mean <- colMeans(yrep)
  # Formula of RMSE:
  sqrt(mean((yrep_mean - y)^2))
}


# Predictions with RMSE
# Model 1
pred_m.1 <- posterior_predict(m.1, newdata = life_expectancy_test)
print(paste("RMSE Model 1: ", rmse(y = life_expectancy_test$Life_expectancy, y
rep = pred_m.1)))
```

```
[1] "RMSE Model 1:  1.91146182386899"
```

```
# Model 2
pred_m.2 <- posterior_predict(m.2, newdata = life_expectancy_test)
print(paste("RMSE Model 2: ", rmse(y = life_expectancy_test$Life_expectancy, y
rep = pred_m.2)))
```

```
[1] "RMSE Model 2:  1.92274110591797"
```

```
# Model 3
pred_m.3 <- posterior_predict(m.3, newdata = life_expectancy_test)
print(paste("RMSE Model 3: ", rmse(y = life_expectancy_test$Life_expectancy, y
rep = pred_m.3)))
```

```
[1] "RMSE Model 3:  1.84344445030423"
```

```
# Model 4
pred_m.4 <- posterior_predict(m.4, newdata = life_expectancy_test)
print(paste("RMSE Model 4: ", rmse(y = life_expectancy_test$Life_expectancy, y
rep = pred_m.4)))
```

```
[1] "RMSE Model 4:  1.84009843741068"
```

```
# Model 5
pred_m.5 <- posterior_predict(m.5, newdata = life_expectancy_test)
print(paste("RMSE Model 5: ", rmse(y = life_expectancy_test$Life_expectancy, y
rep = pred_m.5)))
```

```
[1] "RMSE Model 5:  1.83861330989505"
```

```
# Predictions with MAE
# Model 1
print(paste("MAE Model 1: ", mean(abs(pred_m.1 - life_expectancy_test$Life_exp
ectancy))))
```

```
[1] "MAE Model 1:  8.36236590050837"
```

```
# Model 2
print(paste("MAE Model 2: ", mean(abs(pred_m.2 - life_expectancy_test$Life_exp
ectancy))))
```

```
[1] "MAE Model 2:  8.37353914956068"
```

```
# Model 3
print(paste("MAE Model 3: ", mean(abs(pred_m.3 - life_expectancy_test$Life_exp
ectancy))))
```

```
[1] "MAE Model 3:  8.34484622885433"
```

```
# Model 4
print(paste("MAE Model 4: ", mean(abs(pred_m.4 - life_expectancy_test$Life_exp
ectancy))))
```

```
[1] "MAE Model 4:  8.40778784091711"
```

```
# Model 5
print(paste("MAE Model 5: ", mean(abs(pred_m.5 - life_expectancy_test$Life_exp
ectancy))))
```

```
[1] "MAE Model 5:  8.40419873466937"
```

A possible utility function could be made for a pension fund. Here, the objective would be to ensure the financial stability of the fund by accurately predicting life expectancy to determine pension payouts. In this case, the utility function could reflect the financial impact of overestimating and underestimating life expectancy.

Underestimation: If life expectancy is underestimated, the pension fund might run out of money sooner than expected, which results in insufficient funds for future payouts.
Overestimation: If the pension fund overestimates life expectancy, it will allocate more funds than necessary for payouts, assuming a longer payout period. This results in a larger portion of the fund being set aside for current payouts, reducing the amount available for investment and growth.

For this we made the following utility function that punishes underestimating more than overestimating, since the pension fund could run out of money when underestimating too much. (But the parameters can be changed).

```
util_life_expectancy <- function(actual, predicted, penalty_under = 1.5, penal
ty_over = 1) {
  ifelse(predicted - actual > 0,
         -penalty_over * (predicted - actual),
         -penalty_under * (actual - predicted))
}

eutil_life_expectancy <- function(y, yrep, penalty_under = 1.5, penalty_over =
1) {
  yrep_mean <- colMeans(yrep)
  mean(util_life_expectancy(y, yrep_mean, penalty_under, penalty_over))
}





# Calculate Expected Utility for each model --(**COPIED FROM AI**)--
expected_utility_m1 <- eutil_life_expectancy(y = life_expectancy_test$Life_exp
ectancy, yrep = pred_m.1)
print(paste("Expected Utility Model 1: ", expected_utility_m1))
```

```
[1] "Expected Utility Model 1:  -1.84186302514276"
```

```
expected_utility_m2 <- eutil_life_expectancy(y = life_expectancy_test$Life_exp
ectancy, yrep = pred_m.2)
print(paste("Expected Utility Model 2: ", expected_utility_m2))
```

```
[1] "Expected Utility Model 2:  -1.85818287155041"
```

```
expected_utility_m3 <- eutil_life_expectancy(y = life_expectancy_test$Life_exp
ectancy, yrep = pred_m.3)
print(paste("Expected Utility Model 3: ", expected_utility_m3))
```

```
[1] "Expected Utility Model 3:  -1.77765440389155"
```

```
expected_utility_m4 <- eutil_life_expectancy(y = life_expectancy_test$Life_exp
ectancy, yrep = pred_m.4)
print(paste("Expected Utility Model 4: ", expected_utility_m4))
```

```
[1] "Expected Utility Model 4:  -1.7497334443813"
```

```
expected_utility_m5 <- eutil_life_expectancy(y = life_expectancy_test$Life_exp
ectancy, yrep = pred_m.5)
print(paste("Expected Utility Model 5: ", expected_utility_m5))
```

```
[1] "Expected Utility Model 5:  -1.74204124199367"
```

Contributions of each member

- Fynn –> descriptives of dataset, second model and all code relating to it, describing best model, describing workflow and Statement of technology, leading group meetings
- Emma –> Creation of the fifth model and the explanations and conclusions in relation to this model
- Sam –> Creation of the fourth model and all explanations and conclusion relation to this, checking interpretations of the models by other group members.
- Giyanto –> pre-processing of dataset, first model and all code relating to it, Setting up CV, PPD, RMSE, and Utility function.
- Evi –> Building model 3 and providing explanations on its chosen structure. As well as interpretation of its outputs.

Description of workflow

On the day of the assignment release, group members met on zoom to go through the assignment and look for a dataset together until we collectively decided that we want to work with the life expectancy dataset. Fynn was tasked dealing with the description of that dataset, while Giyanto was tasked with the pre-processing steps. Following this, each member was to come up with one model for the model exploration. Giyanto voluntarily prepared further analysis (i.e., CV, RMSE, etc.). All other group members then applied his work on their own models and filled out the remaining steps. Discussion of Posterior Predictive Checks for model 5 was done by Fynn. Lastly, Fynn took care of the description of the best model performance, the statement of technology, as well as taking a last look over our documents before sending it in via Canvas, with other Group members also checking the final document for errors.

Statement of technology

Our group has made use of AI knowledge in order to clarify concepts and ensure that we understand our work, however no AI-generated code has been used except where explicitly specified

References

- Lasha. (2023). Life expectancy (WHO) fixed [Dataset]. In Kaggle. https://www.kaggle.com/datasets/lashagoch/life-expectancy-who-updated/code (https://www.kaggle.com/datasets/lashagoch/life-expectancy-who-updated/code)
- Rajarshi, K. [KumarRajarshi]. (2018). Life expectancy (WHO) [Dataset]. In Kaggle. https://www.kaggle.com/datasets/kumarajarshi/life-expectancy-who/code?datasetId=12603&sortBy=relevance (https://www.kaggle.com/datasets/kumarajarshi/life-expectancy-who/code?datasetId=12603&sortBy=relevance)
- Gadige, H. (2019, May 18). Life Expectancy Cleaning,EDA,Feature Engineering. Kaggle. https://www.kaggle.com/code/harshini564/life-expectancy-cleaning-eda-feature-engineering (https://www.kaggle.com/code/harshini564/life-expectancy-cleaning-eda-feature-engineering)
- Sai Kanuri, V. (2022, October 3). Life expectancy visualization. Kaggle. https://www.kaggle.com/code/varunsaikanuri/life-expectancy-visualization (https://www.kaggle.com/code/varunsaikanuri/life-expectancy-visualization)
- Ngala540. (2023, May 5). Life expectancy prediction. Kaggle. https://www.kaggle.com/code/ngala540/life-expectancy-prediction (https://www.kaggle.com/code/ngala540/life-expectancy-prediction)