Learning to Rest: Predicting Sleep from Fitness

STAT 447 Group Project

Michael Garbus (mgarbus2), Michelle Shen (mjshen3), Sarah Yang (sarahxy2)
STAGES cohort investigator group

Contents

Summary

As college students who sometimes have to prioritize other things over health, sleep, or both, we are interested in how fitness can affect the quality of sleep. We were able to get datasets from the Stanford Technology Analytics and Genomics in Sleep (STAGES) study which contains collected data on over 1,500 anonymized adult or adolescent patients evaluated for sleep disorders thanks to the National Sleep Research Resource The National Sleep Research Resource. The STAGES data contained sleep polysomnography recordings (where in electrodes are attached to the body to record sleep data — the gold standard for sleep studies), surveys done on the participants before the study, and surveys done on the participants after the study. The surveys contained fitness data and past health data which we used to determine people who were fit, and people who were good sleepers. We used this data to determine if we could predict a good sleeper from their fitness attributes using 3 machine learning techniques: XGBoost, KNN, and Elastic-Net.

Summary & Context of Data

The data used in this project is sourced from STAGES - Stanford Technology, Analytics, and Genomics in Sleep. The STAGES study is a cross-sectional, multi-center study created to study the vital infrastructure necessary for sleep and sleep disorder research. Approximately 30,000 patients provided data at various sleep clinic sites over a period of less than 4 years. The patients in the study were recruited from clinical centers across the U.S. and Canada, and each subject spent approximately 1-3 hours completing questionnaires, neurocognitive testing, photography, and blood sampling for purposes of the study. Patients were also given an Actigraph device to wear for at least 2 weeks at home to monitor fitness activity and health metrics.

The data was gathered through surveys and in controlled environments through which implementation of protocol and training of staff were developed with standard operating procedures and quality assurance and control of data.

Libraries

```
library(data.table)
library(dplyr)
library(ggplot2)
library(readxl)
library(vtable)
library(curl)
library(fastDummies)
library(glmnet)
```

```
#Prepare data
sleep_data_full <- fread('/cloud/project/fa21-prj-mgarbus2-mjshen3-sarahxy2/datasets/stages-dataset-0.1
sleep_data <- sleep_data_full[,-c(1,2)]
SRDBVars <- read_excel('/cloud/project/fa21-prj-mgarbus2-mjshen3-sarahxy2/datasets/STAGESPSGKeySRBDVari
SRDBVars<- SRDBVars[,c(2,15,18,19)]

#Convert sleep time
SRDBVars$sleep_time <- (SRDBVars$sleep_time)/60
```

Summary Statistics of Key Variables: Sleep time, Age, and BMI

```
#sumtable(SRDBVars)
```

```r
library(tidyr)

#Remove those who are pregnant (2 women)
sleep_data <- sleep_data[-which(sleep_data[,c('mdhx_1200')] == 1),]

#select fitness data

fitness_data <- sleep_data[,c('subject_code','dem_0500','dem_0800','fss_1000',
                              'gad_0800','phq_1000','nose_0300',
                              'nose_0500','diet_0340','diet_0400',
                              'diet_0700','soclhx_0501',
                              'soclhx_0700','soclhx_0900','soclhx_1500',
                              'famhx_0700','ess_0900','narc_1600')]


data_dictionary <- fread('/cloud/project/fa21-prj-mgarbus2-mjshen3-sarahxy2/datasets/stages-data-dictio

#select columns
row_dict_vals <- which(data_dictionary$id %in% colnames(fitness_data))

#Load data dictionary and make a table of our variables.


mean(fitness_data$age, na.rm = T)
```

```
## [1] NA
```

```r
kable(data_dictionary[row_dict_vals,c('id','display_name')])
```

id	display_name
subject_code	STAGES Subject Identifier
dem_0800	Body mass index (BMI)
dem_0500	Participant's sex
fss_1000	Fatigue Severity Scale: Total Score
gad_0800	Generalized Anxiety Disorder-7 Questionnaire: Total Score
phq_1000	Patient Health Questionnaire 9: Total Score
nose_0300	Trouble breathing through nose
nose_0500	Unable to get enough air through nose during exercise or exertion
diet_0340	Food intake - No regular meals
diet_0400	Eating impact on alertness/wakefulness
diet_0700	Self-perception of weight
soclhx_0501	Exercise, rarely or never
soclhx_0700	Alcohol consumption, number of times
soclhx_0900	Caffeine consumption, number of servings per day
soclhx_1500	Street or recreational drugs consumption, ever
famhx_0700	Family History of Fibromyalgia or Chronic Fatigue
ess_0900	Epworth Sleepiness Scale: Total score
narc_1600	Muscle weakness, number of episode

Summary of Questionnaires & Surveys

We included the results of some of the surveys used in the data. Summaries of the surveys are below.

The Epworth Sleepiness Scale (ESS) was developed by Dr. Murray Johns in 1990, and it is a scale to assess the daytime sleepiness of his Sleep Medicine private practice patients. It is a self-administered questionnaire with 8 questions where respondents rate, on a 4-point scale, their likelihood of dozing off or falling asleep while engaged in eight different, common activities. The survey takes no more than 3 minutes to complete, and it is an objective measure of daytime sleepiness. The survey does not ask about subjective feelings of alertness/drowsiness, but simply how likely dozing off is. The activities listed in the ESS questionnaire are: sitting and reading, watching TV, inactively sitting in a public place (e.g. a theatre), as a passenger in a car for an hour without a break, lying down to rest in the afternoon when circumstances permit, sitting and talking to someone, and being in a car while stopped for a few minutes in traffic. Through these relatable and easily imaginable scenarios, the survey respondent would provide the likelihood of falling asleep/dozing. This is one of the predictors used to predict the qualities of a good sleeper.

In addition to using the ESS results as a predictor, the dataset also includes the use of the Fatigue Severity Scale (FSS), which is a 9-item scale used to measure the severity of fatigue and its effect on a person's activities and lifestyle among patients with a variety of disorders. The items are scored on a 7-point scale with 7 being "strongly agree," and a higher score indicating greater fatigue and negative results from it. An example prompt would be "Fatigue is among my most disabling symptoms."

The `phq_0900` variable is the sum for Patient Health Questionnaire. Each item is scored from 0 to 3, while the total score range from 0 to 27. The score can be divided into the following categories of increasing severity: Not Clinically significant (0-4), Minimal Symptoms (5-9), Minor depression/Dysthymia/Mild Major Depression (10-14), Moderately Severe Major Depression (15-19), and Severe Major Depression (20-27) Kroenke K et al. 2001. We created a categorical variable divided based off of those who received a Mild/Major Depression score and above, and those who did not.

The `gad_0800` variable is the sum of the Generalised Anxiety Disorder Assessment (GAD-7). A score of above 10 means that the patient has severe anxiety, or at least must be recommended for further evaluation Spitzer R et al. 2006. We created a categorical based off of the score being above 10.

```
post_psg <- read_excel('datasets/STAGES post sleep questionnaire 2020-09-06 deidentified.xlsx', na = "NA
# remove unrecorded data
post_psg <- post_psg[-which(is.na(post_psg$modified.date_of_evaluation)),]

post_psg <- post_psg[,c(1,5:7, 9:10)]

first_inner <- merge(SRDBVars, fitness_data,  by.y = 'subject_code', by.x = 's_code')
second_inner <- merge(post_psg, first_inner,  by.y = 's_code', by.x = 'subject_id')

fitness_data <- second_inner

fitness_data$age[is.na(fitness_data$age)] <- mean(fitness_data$age, na.rm = T)

fitness_data$dem_0800 <- replace_na(fitness_data$dem_0800, mean(fitness_data$dem_0800, na.rm = T))

fitness_data$diet_0400 <- replace_na(fitness_data$diet_0400, mean(fitness_data$diet_0400, na.rm = T))

fitness_data$soclhx_0501 <- replace_na(fitness_data$soclhx_0501, 0)


fitness_data$fss_1000 <- replace_na(fitness_data$fss_1000, median(fitness_data$fss_1000, na.rm = T))
fitness_data$fss_1000[fitness_data$fss_1000 <= 36] <- 0
fitness_data$fss_1000[fitness_data$fss_1000 > 36] <- 1
#1 means fatigued
```

```r
names(which(colSums(is.na(fitness_data)) > 0))
```

```
##  [1] "did_you_awaken_during_night"        "awaken_how_many_times_during_night"
##  [3] "awakenings_compared_to_usual"       "compared_usual_sleep_duration"
##  [5] "compared_usual_feel_upon_awakening" "bmi"
##  [7] "gad_0800"                           "phq_1000"
##  [9] "nose_0300"                          "nose_0500"
## [11] "diet_0340"                          "diet_0700"
## [13] "soclhx_0700"                        "soclhx_0900"
## [15] "soclhx_1500"                        "famhx_0700"
## [17] "ess_0900"                           "narc_1600"
```

```r
fitness_data$gad_0800 <- replace_na(fitness_data$gad_0800, median(fitness_data$gad_0800, na.rm = T))
fitness_data$gad_0800[fitness_data$gad_0800 < 10] <- 0
fitness_data$gad_0800[fitness_data$gad_0800 >= 10] <- 1
# 1 means anxious

fitness_data$diet_0700 <- replace_na(fitness_data$diet_0700, median(fitness_data$diet_0700, na.rm = T))
fitness_data$diet_0700[fitness_data$diet_0700 != 0] <- 4
fitness_data$diet_0700[fitness_data$diet_0700 == 0] <- 1
fitness_data$diet_0700[fitness_data$diet_0700 == 4] <- 0
# 1 means unhealthy

fitness_data$famhx_0700 <- replace_na(fitness_data$famhx_0700, 0)
fitness_data$famhx_0700[fitness_data$famhx_0700 == -55] <- 0

fitness_data$narc_1600 <- replace_na(fitness_data$narc_1600, 0)
fitness_data$narc_1600[fitness_data$narc_1600 <= 3] <- 0
fitness_data$narc_1600[fitness_data$narc_1600 >= 3] <- 1
#If muscle weak occurs


fitness_data$soclhx_0900 <- replace_na(fitness_data$soclhx_0900, median(fitness_data$soclhx_0900, na.rm
fitness_data$soclhx_0900[fitness_data$soclhx_0900 <= 2] <- 0
fitness_data$soclhx_0900[fitness_data$soclhx_0900 >= 1] <- 1
#caffeine


fitness_data$soclhx_0700 <- replace_na(fitness_data$soclhx_0700, 0)
#Number of alcoholic drink frequency


#fitness_data$soclhx_1320 <- replace_na(fitness_data$soclhx_1320, 0)
#cigarettes

fitness_data$soclhx_1500 <- replace_na(fitness_data$soclhx_1500, 0)
fitness_data$soclhx_1500[fitness_data$soclhx_1500 >= 1] <- 1
# Drug usage, 1 = drug user if ever used drugs


fitness_data$phq_1000 <- replace_na(fitness_data$phq_1000, median(fitness_data$phq_1000, na.rm = T))
fitness_data$phq_1000[fitness_data$phq_1000 < 10] <- 0
fitness_data$phq_1000[fitness_data$phq_1000 >= 10] <- 1
# 1 means depressed
```

```
fitness_data$diet_0340 <- replace_na(fitness_data$diet_0340, 0)
fitness_data$diet_0340[fitness_data$diet_0340 < 1] <- 0
fitness_data$diet_0340[fitness_data$diet_0340 > 1] <- 1
#1 means no regular meal intake


fitness_data$bmi <- replace_na(fitness_data$bmi, mean(fitness_data$bmi, na.rm = T))


fitness_data$awakenings_compared_to_usual[is.na(fitness_data$awakenings_compared_to_usual)] <- 'same'
fitness_data$ess_0900 <- replace_na(fitness_data$ess_0900, 0)
fitness_data$compared_usual_feel_upon_awakening[is.na(fitness_data$compared_usual_feel_upon_awakening)]

length(fitness_data$dem_0500[fitness_data$dem_0500 == ""])
```

```
## [1] 18
```

```
# 18 unrecorded, assigning to M
fitness_data$dem_0500[fitness_data$dem_0500 == ""] <- "M"

fitness_data$nose_0500 <- replace_na(fitness_data$nose_0500,0)
fitness_data$nose_0500[fitness_data$nose_0500 < 2 ] <- 0
fitness_data$nose_0500[fitness_data$nose_0500 >= 2 ] <- 1
#1 means cant breathe

fitness_data$nose_0300 <- replace_na(fitness_data$nose_0300,0)
fitness_data$nose_0300[fitness_data$nose_0300 < 2 ] <- 0
fitness_data$nose_0300[fitness_data$nose_0300 >= 2 ] <- 1

#1 means cant breathe/difficulty
```

```
#sumtable(fitness_data)
```

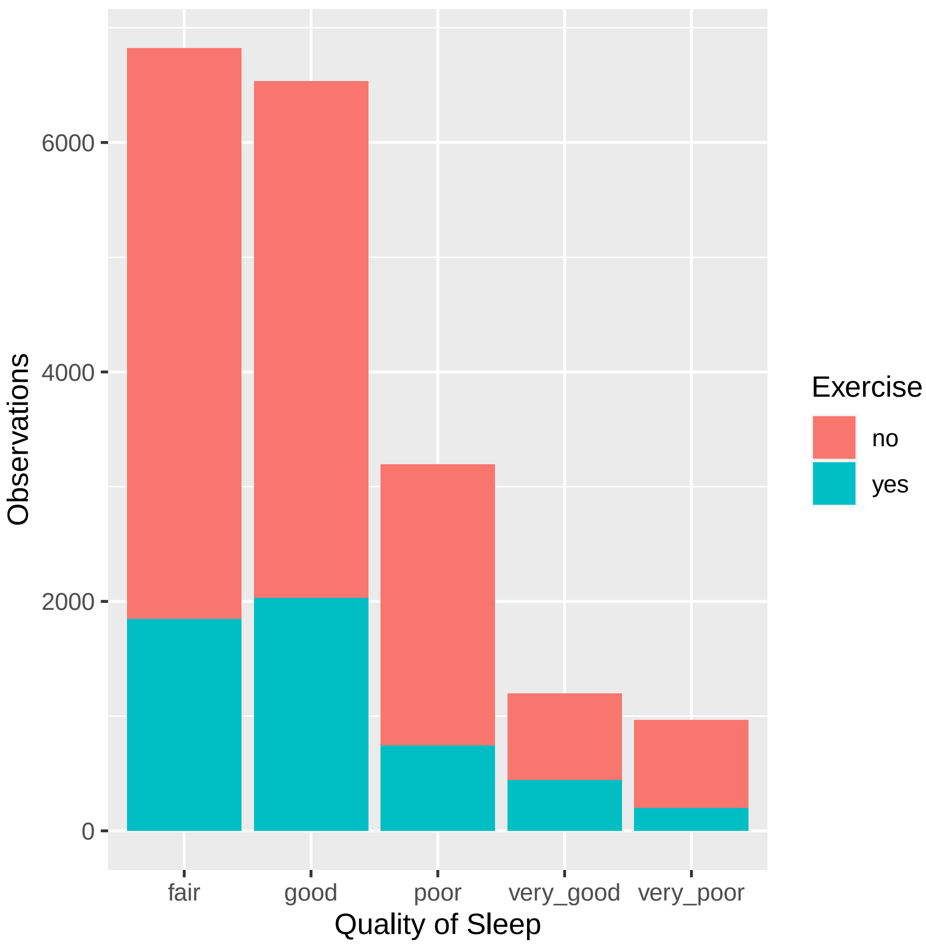
Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
sleep_time	1687	335.513	98.928	60.5	283.5	402.5	655
age	1675	45.789	15.169	0	34	57	84
bmi	1656	31.145	8.635	11.9	25.4	35.225	75

Figure 1: sumtable result

```
#self-reported data
sleep_diary <- read_excel('/cloud/project/fa21-prj-mgarbus2-mjshen3-sarahxy2/datasets/STAGES Sleep Diar
sleep_diary <- sleep_diary[,c(1:11)]
sleep_quality_exercise <- sleep_diary |>
 na.omit() |>
 group_by(quality_of_sleep) |>
  count(modified.exercise_yesyeserday_yes_no)
kable(sleep_quality_exercise)
```

quality_of_sleep	modified.exercise_yesyeserday_yes_no	n
fair	no	4972
fair	yes	1849
good	no	4507
good	yes	2028
poor	no	2451
poor	yes	742
very_good	no	752
very_good	yes	444
very_poor	no	770
very_poor	yes	197

```
# Making bar plot for self-reported data
ggplot(sleep_quality_exercise, aes(x = quality_of_sleep, y = n,
                                   fill = modified.exercise_yesyeserday_yes_no,
                                   label = n)) +
  geom_bar(stat = "identity") +
  xlab("Quality of Sleep") +
  ylab("Observations") +
  labs(fill = "Exercise")
```




```
  geom_text(size = 3, position = position_stack(vjust = 0.5))
```

```
## geom_text: parse = FALSE, check_overlap = FALSE, na.rm = FALSE
## stat_identity: na.rm = FALSE
```

```
## position_stack
```

```r
post_psg <- read_excel('datasets/STAGES post sleep questionnaire 2020-09-06 deidentified.xlsx',
                       na = "NA")
# remove unrecorded data
post_psg <- post_psg[-which(is.na(post_psg$modified.date_of_evaluation)),]
#lapply(post_psg, function(x) sum(is.na(x))) # values
library(dplyr)
#Can be graphed

post_psg |>
  group_by(awaken_how_many_times_during_night) |>
  summarize(count = n())
```

```
## # A tibble: 7 x 2
##   awaken_how_many_times_during_night count
##   <chr>                              <int>
## 1 1                                    181
## 2 1_to_2                                 1
## 3 2                                    274
## 4 3                                    347
## 5 4                                    247
## 6 more_4                               455
## 7 <NA>                                 224
```

```r
median(post_psg$awaken_how_many_times_during_night, na.rm = T) #Assign 3 to medium value
```

```
## [1] "3"
```

```r
#replace NA values with median, "3"
post_psg$awaken_how_many_times_during_night[is.na(post_psg$awaken_how_many_times_during_night)] <- media
post_psg$awaken_how_many_times_during_night[post_psg$awaken_how_many_times_during_night == '1_to_2'] <-

kable(table(post_psg$awaken_how_many_times_during_night))
```

Var1	Freq
1	181
2	275
3	571
4	247
more_4	455

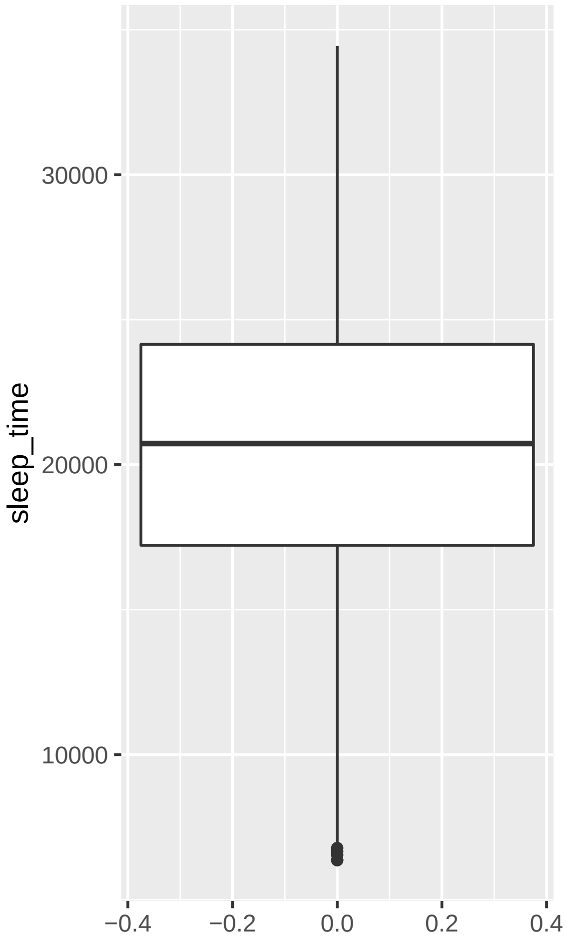
```r
#Feature generation

#mean(fitness_data$age,na.rm = T)
#median(fitness_data$age,na.rm = T)
#Average age is 45.7886, median is 46.
#7 hours needed: https://www.cdc.gov/sleep/about_sleep/how_much_sleep.html
fitness_data$better_than_avg_sleep <- as.numeric(fitness_data$sleep_time > mean(fitness_data$sleep_time
#Drop people who have NA in awakenings instead of assign to "same" in case

# A good sleeper has an Above average sleep time, Epsworth Sleep Scale below 16,
#meaning not severe excessive daytime sleepiness,
#compared_usual_feel_upon_awakening same or more rested, less or same # of awakenings,


fitness_data$good_sleeper <- as.numeric(fitness_data$compared_usual_feel_upon_awakening %in% c('same','r
```

```
sum(fitness_data$good_sleeper)
```

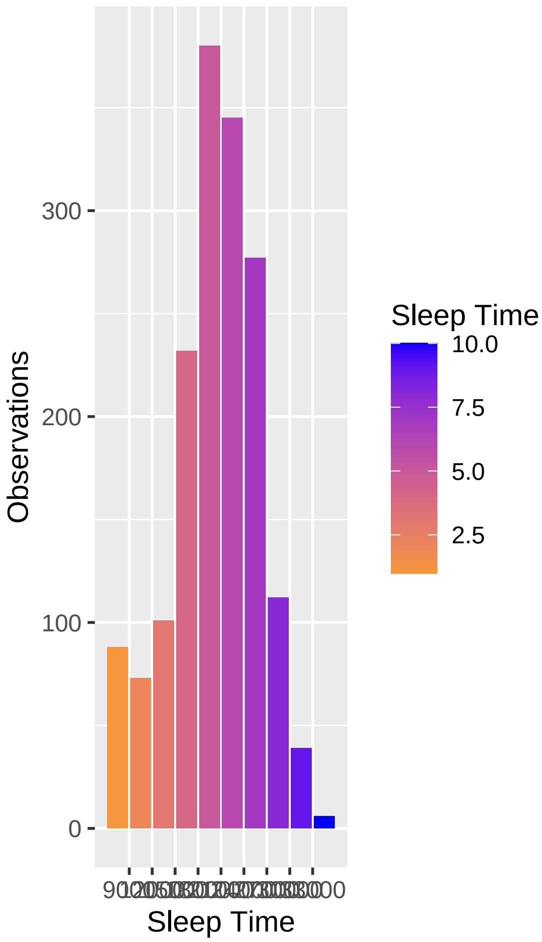
```
## [1] 529
```

```
#529 "good sleepers" in this dataset!
```

```
par(mfrow = c(3,3))
# Change csv file into a data table to manipulate
data <- data.table::fread("/cloud/project/fa21-prj-mgarbus2-mjshen3-sarahxy2/datasets/PSGKeyVariables.cs

# Manipulating the data table
# Removing the 7 most sleeps out of 1687 observations and then filtering out all outliers
data1 <- data[!(sleep_time>35000), ]
data2 <- data[!sleep_time %in% boxplot.stats(sleep_time)$out]

# Graphing boxplot to get an idea of the range and distribution of data
ggplot(data2, aes(y = sleep_time)) + geom_boxplot()
```




```
# Corresponding bar plot that will be used to separate sleep_time into categories
ggplot(data2, aes(x = sleep_time)) +
  geom_bar(aes(fill = ..x..)) + scale_x_binned(n.breaks = 10) +
  xlab("Sleep Time") +
  ylab("Observations") +
  scale_fill_gradient2(low='white', mid='orange', high='blue', name = "Sleep Time")
```




```
summary(data2$sleep_time)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    6360   17220   20730   20308   24150   34440
```

```
# Ranking sleep_time as a categorical variable from the 10 bins,
#1 being the worst sleep quality and 10 being the best sleep quality
brk <- c(0, 9000, 12000, 15000, 18000, 21000, 24000, 27000, 30000, 33000, Inf)
data2[, category := cut(sleep_time, breaks = brk, include.lowest = TRUE,
                        labels = c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10"))]
```

Data Wrangling

```
fitness_data_table <- as.data.table(fitness_data)
```

```
fitness_data_table[, .(avg_sleep_time = mean(sleep_time, na.rm = TRUE),
                       count = length(sleep_time)), by = good_sleeper]
```

```
##    good_sleeper avg_sleep_time count
## 1:            1       414.1342   529
## 2:            0       297.4194  1049
```

From our classifications of a good sleeper, a good sleeper will have, on average, 117 minutes more sleep than that of a poorer sleeper.

```r
fitness_data_table[, .(avg_sleep_time = mean(sleep_time, na.rm = TRUE), count = length(sleep_time)),
                   by = compared_usual_feel_upon_awakening][order(avg_sleep_time)]
```

```
##    compared_usual_feel_upon_awakening avg_sleep_time count
## 1:                        more_rested       291.6565   131
## 2:                        less_rested       307.4433   467
## 3:                               same       356.4153   980
```

From this result of average sleep time among surveyed participants, we can oddly see that those who felt more rested than usual slept for less time (in minutes) than other categories. We speculate that those participants felt more rested because they did not enter deep sleep for as long as others, or simply woke up during an optimal point during the REM cycle and did not feel groggy. Or perhaps those that feel more_rested during this study just did not get adequate sleep prior to participating, and some factors during the survey period allowed them to sleep more soundly. Because the feeling upon awakening is subjective relative to each participant, we do not have a concrete, quantitative reason behind the correlation.

```r
fitness_data_table[, .(avg_sleep_time = mean(sleep_time, na.rm = TRUE),
                       median_sleep_time = median(sleep_time, na.rm = TRUE),
                       count = length(sleep_time)),
                   by = list(nose_0300, nose_0500)][order(-avg_sleep_time)]
```

```
##    nose_0300 nose_0500 avg_sleep_time median_sleep_time count
## 1:         0         0       338.8081            344.75  1144
## 2:         1         0       332.0595            346.00   126
## 3:         1         1       331.2667            337.50   195
## 4:         0         1       327.7611            338.50   113
```

From this output, we can see that a person without the breathing/sinus issues in question has a higher average sleep time than those who do have one or both breathing issues. Interestingly, a person with trouble breathing (nose_0300) has a higher median sleep time than one without breathing issues by about 1 minute. Although this difference is noticeable, it is most likely not significant.

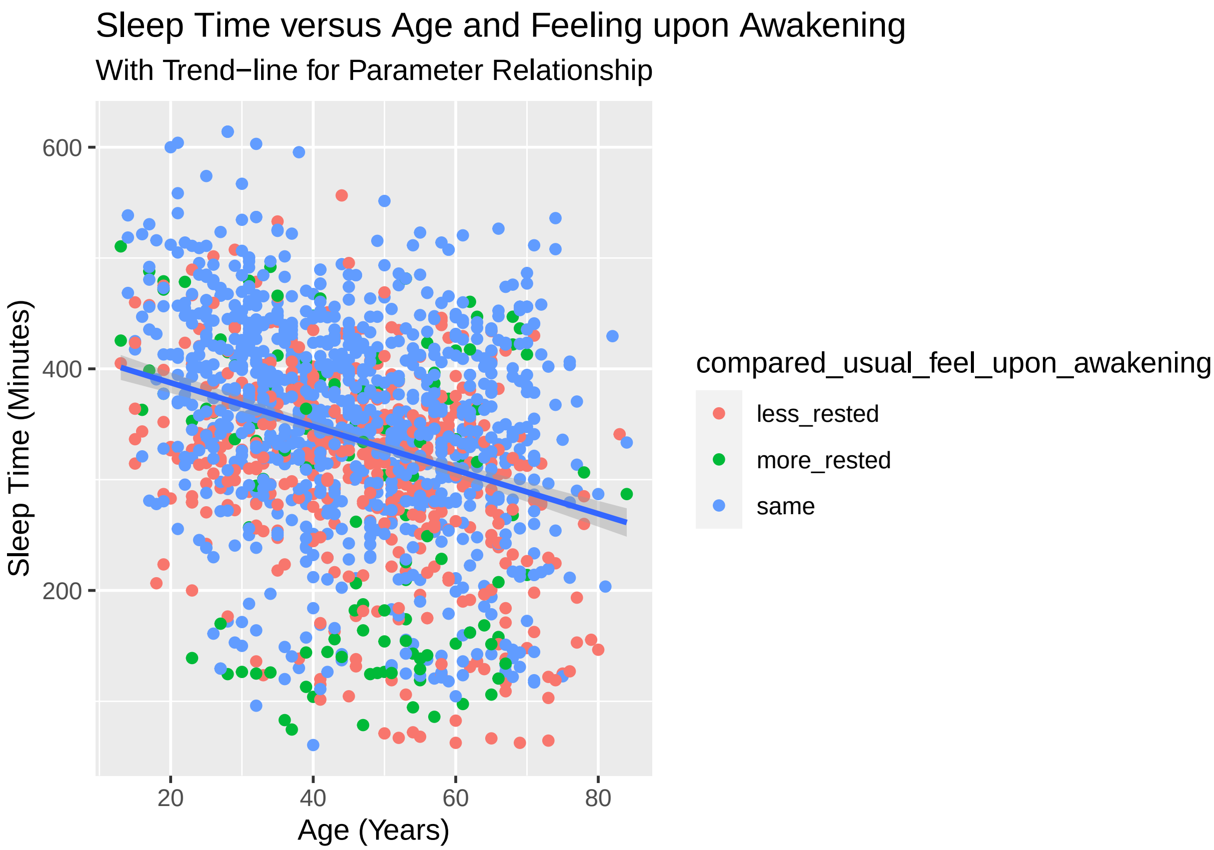
```r
fitness_data_table[, .(avg_sleep_time = mean(sleep_time, na.rm = TRUE), count = length(sleep_time)),
                   by = list(gad_0800, good_sleeper, dem_0500)][order(-avg_sleep_time)]
```

```
##    gad_0800 good_sleeper dem_0500 avg_sleep_time count
## 1:        1            1        M       444.9359    39
## 2:        0            1        M       415.3083   240
## 3:        1            1        F       412.1500    60
## 4:        0            1        F       406.9553   190
## 5:        1            0        F       348.6549   113
## 6:        0            0        F       308.5871   442
## 7:        0            0        M       277.4691   437
## 8:        1            0        M       262.2018    57
```

Here is an output of average sleep times grouped by whether we classified them as a good sleeper, their gender, and whether they are diagnosed with a general anxiety disorder from the questionnaire. A general trend here is that a participant with a general anxiety disorder will tend to sleep longer, and men will tend to sleep longer than women as well. One thing to notice is that, of the top six average sleep times, five of them were categorized as good sleepers except for 89 women (third row of output). Despite not being categorized as a good sleeper, they had the third highest average sleep time.

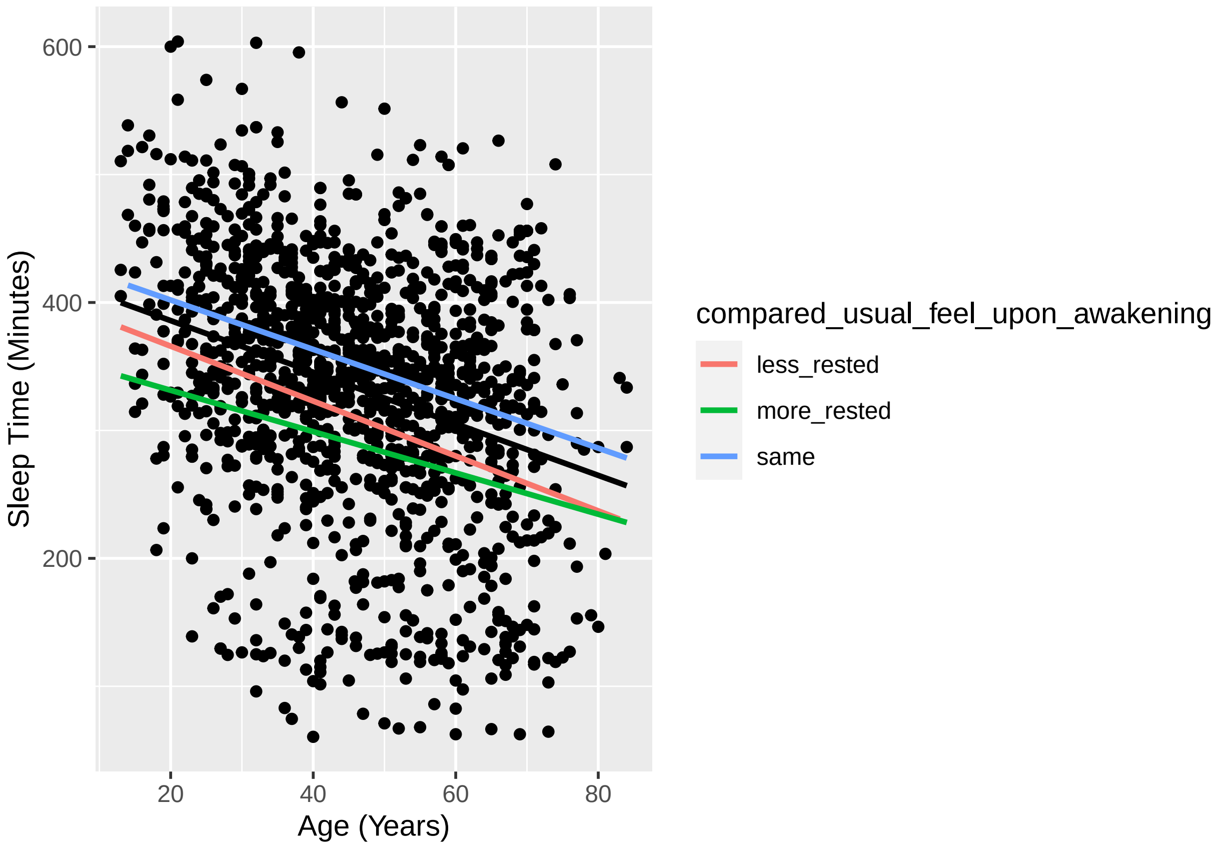
Visualizations

```r
ggplot(fitness_data, aes(x = age, y = sleep_time)) +
  geom_point(aes(color=compared_usual_feel_upon_awakening)) +
  #geom_vline(xintercept = 4500) +
  #geom_hline(yintercept = 45) +
  geom_smooth(method = "lm", formula=y~x) +
  ggtitle("Sleep Time versus Age and Feeling upon Awakening",
          subtitle = "With Trend-line for Parameter Relationship") +
  xlab("Age (Years)") +
  ylab("Sleep Time (Minutes)")
```



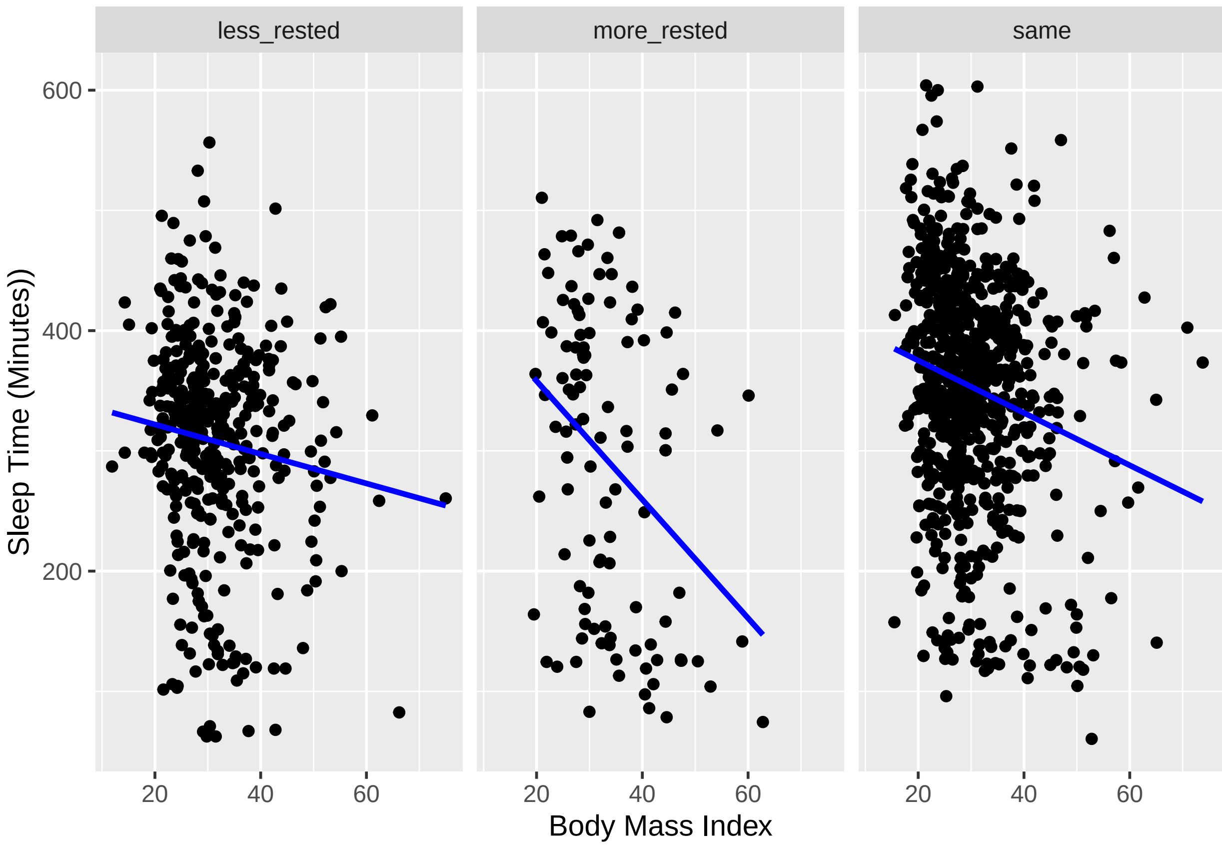
From this colored scatterplot with a line of best fit, we find a general trend that average sleep time among patients will decrease as they grow older in age. We can also see that the scatterplot has greater density of points near the higher sleep times, with a thinner distribution of points among the lower sleep times (250 minutes or less). It is difficult to see a clear difference between less_rested, more_rested, and same feelings of well-restedness from this plot, so we will create another visualization to focus on that.

```r
ggplot(data=na.omit(fitness_data),
    mapping=aes(x=age, y=sleep_time)) +
geom_point() +
geom_smooth(method = "lm", formula=y~x, colour="black", se=FALSE) +
geom_smooth(aes(color=compared_usual_feel_upon_awakening),
            method = "lm", formula=y~x, se = FALSE) +
xlab("Age (Years)") +
ylab("Sleep Time (Minutes)")
```



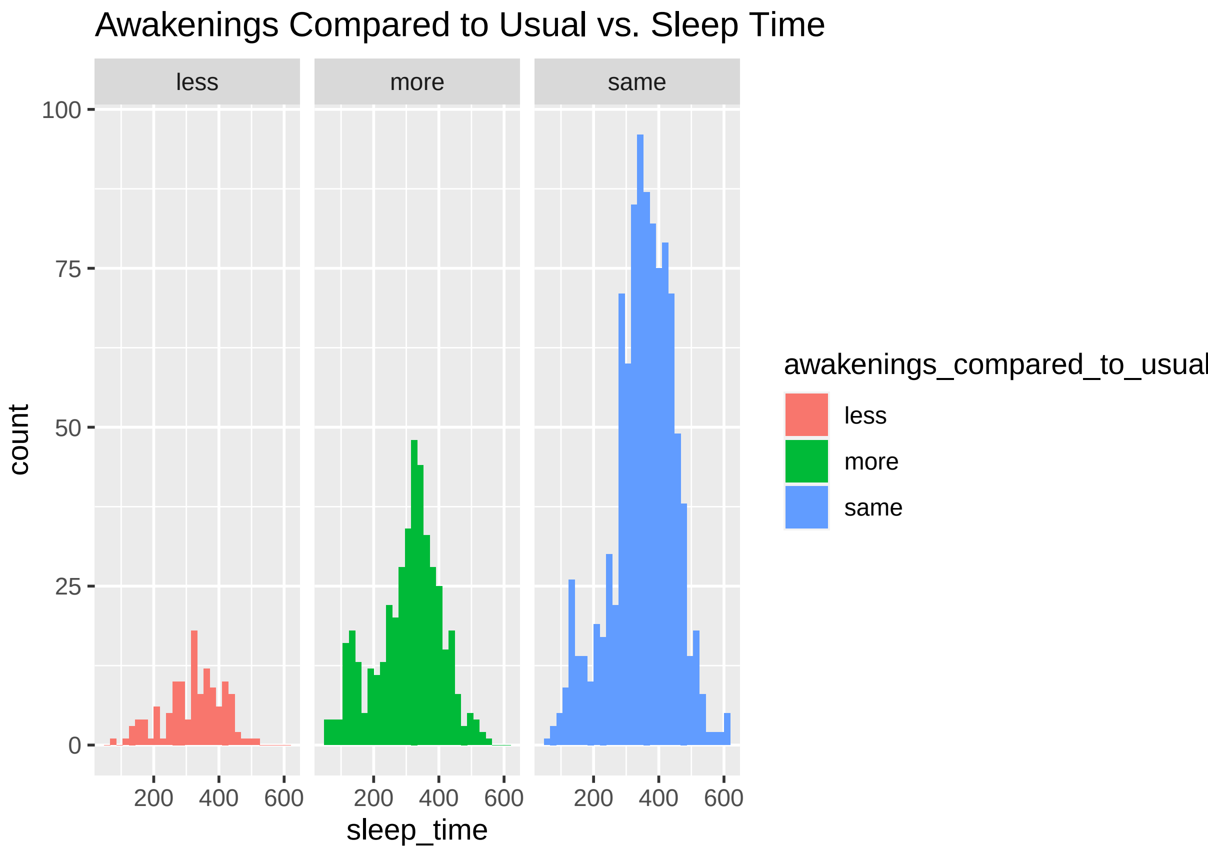
From this second scatterplot, this time with three lines of best fit based on the groupings of feeling upon awakening, we see that the scatterplot creates a visualization of the conclustion seen from the data wrangling table. Those who felt the "same" upon awakening tended to sleep for a longer average time than those that felt less rested or more rested. Something new we can see from the visualization is that, between less_rested and more_rested groups, the difference among sleep_time is more drastic among younger ages. However, near ages ~80, there is not quite a difference in prediction of average sleep times between the less_rested and more_rested groups. We believe this occurred due to lack of enough observations from the elderly age group, leading to those points at age 80 to be higher leverage points.

```
ggplot(data=na.omit(fitness_data),
       mapping=aes(x=bmi,y=sleep_time)) +
      facet_wrap(~ compared_usual_feel_upon_awakening, nrow = 1) +
  geom_point() +
  geom_smooth(method = "lm", formula=y~x, colour="blue", se=FALSE) +
  xlab("Body Mass Index") +
  ylab("Sleep Time (Minutes))")
```



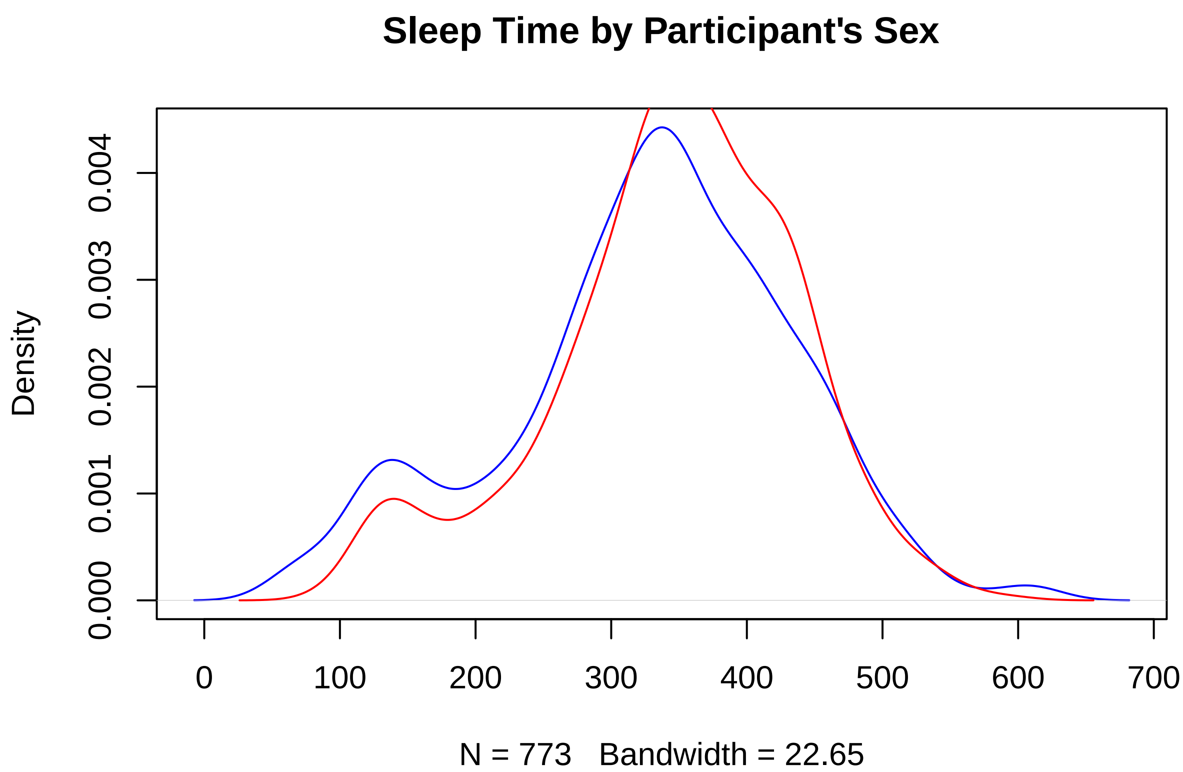
Next is a facet_wrap visualization broken up by feelings of restedness upon awakening, with Body Mass Index among the x-axis. From our understanding, people with a Body between 18.4 and 25.9 are generally considered of healthy BMI. Values under the range are typically classified as underweight, and value over the range are classified as overweight or obese, depending on extremity. Our theory was that people within the healthy BMI range would be considered healthy, and there existed a correlation between health and average sleep time. We can see from these plots that those in the more_rested category have a far more extreme negative relationship between BMI and Sleep Time, compared to those that felt the same or less_rested upon awakening. There could be a correlation between BMI, feelings of well-restedness, and sleep time.

```
ggplot(fitness_data, aes(x=sleep_time,
                         fill = awakenings_compared_to_usual)) +
    geom_histogram(bins = 30, freq = T) +
    facet_wrap( ~ awakenings_compared_to_usual) +
    labs(title = "Awakenings Compared to Usual vs. Sleep Time")
```



From this set of histograms, we compare the distribution of sleep times against a subjective number of awakenings compared to the usual. We can see that the fewest number of participants woke up less during the study than usual, and most participants woke up the same number of times during the night. From two of the three distributions, we can see a slightly bimodal distribution of sleep times, with a peak near 125 minutes and then around the group average (a value between 300-400 minutes).

```
plot(density(fitness_data[fitness_data$dem_0500 == "M",]$sleep_time),
     col = 'blue', main = "Sleep Time by Participant's Sex")
lines(density(fitness_data[fitness_data$dem_0500 == "F",]$sleep_time),
      col = "red")
```



For our last visualization, we created a density plot to analyze the proportion of sleepers receiving minutes of sleep, and it was categorized by the patient's sex, whether they were Male or Female. We can see that a greater proportion of women than men slept between 300-500 minutes, and the men's distribution would have a slightly greater standard deviation in sleep minutes than the women's, with a greater proportion of men sleeping far too little hours.

Modeling

We selected our models based off of prediction accuracy. Because of this, there may be some low values in the sensitivity and specificity.

Extreme Gradient Boosted Tree (XGBoost)

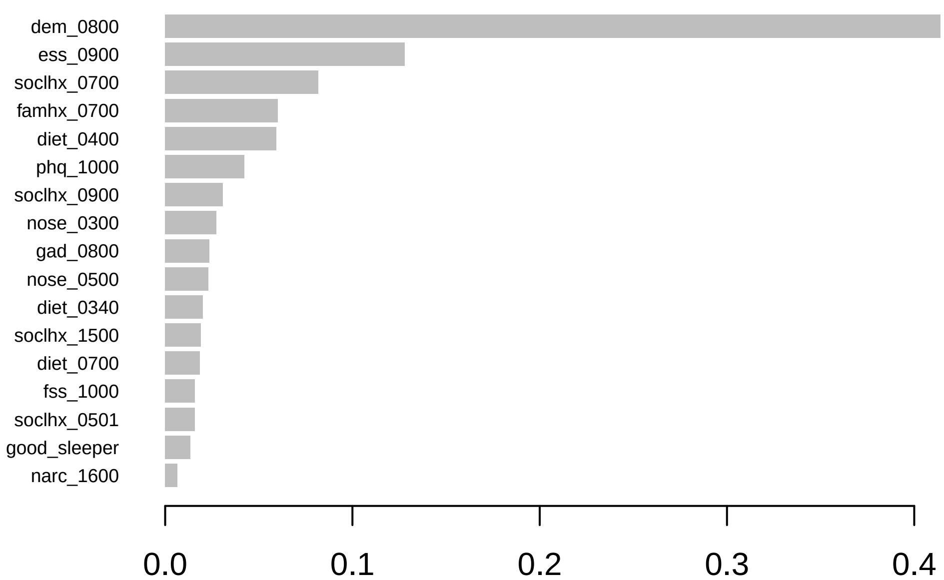
The first model we attempted was an extreme gradient boosted tree model. A boosted tree is a decision tree that achieves a high modelling accuracy by training new models to account for the training data that was previously incorrectly modeled.

```
library(xgboost) #Allows parallel computing!
library(fastDummies)
#Test/Train split of data
set.seed(447)

model_data <- fitness_data[,c('dem_0500','dem_0800','fss_1000','gad_0800',
                              'phq_1000','nose_0300','nose_0500','diet_0340',
                              'diet_0400','diet_0700','soclhx_0501','soclhx_0700',
                              'soclhx_0900','soclhx_1500','famhx_0700','ess_0900',
                              'narc_1600','good_sleeper')]
```

```r
#fastDummies::dummy_cols(model_data)


dummy_fitness <- fastDummies::dummy_cols(model_data, remove_first_dummy = TRUE)

dummy_fitness <- dummy_fitness[,c(-1,-20)]
sample_size <- floor(0.7 * nrow(dummy_fitness))
train_ind <- sample(seq_len(nrow(dummy_fitness)), size = sample_size)
train <- dummy_fitness[train_ind,-c(18,19)]
#fitness_data
train_y <- dummy_fitness[train_ind, 18]
test <- dummy_fitness[-train_ind,-c(18,19)]
test_y <- dummy_fitness[-train_ind, 18]


xgb_model <- xgboost(data = as.matrix(train),label = train_y, max_depth = 6,
                     eta = 0.1, nthread = 16, nrounds = 100 ,
                     objective = "binary:hinge", eval_metric = 'rmse', verbose = 0)

xgb_prediction <- predict(xgb_model, as.matrix(test))
table(xgb_prediction, test_y)
```

```
##               test_y
## xgb_prediction   0   1
##              0 118  87
##              1 107 162
```

```r
xgb_imp <- xgb.importance(model = xgb_model)
xgb.plot.importance(xgb_imp)
```



According to our XGB model, `dem_0800`, `ess_0900`, and `bmi` are the predictors that provided the most gain to the model. It has an accuracy of 60.3448276%, a specificity (true negative classification) of 37.8205128%, and a sensitivity (true positive classification) of 45.505618%.

KNN

The second model we used was KNN, which searches for the closest k neighbors to an observation for classification.

```
library(kknn)
library(caret)
library(class)
control <- trainControl(method = "repeatedcv", number = 5, repeats = 3)
knn.cvfit <- train(y ~ ., method = "knn",
                   data = data.frame("x" = train, "y" = as.factor(train_y)),
                   tuneGrid = data.frame(k =c(5,10,33,50)),
                   trControl = control)
knn.cvfit
```

```
## k-Nearest Neighbors
##
## 1104 samples
##   17 predictor
##    2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold, repeated 3 times)
## Summary of sample sizes: 883, 883, 883, 883, 884, 883, ...
## Resampling results across tuning parameters:
##
##   k   Accuracy   Kappa
##    5  0.5800247  0.1572730
##   10  0.5848444  0.1664574
##   33  0.6195695  0.2349712
##   50  0.6120115  0.2195504
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 33.
```

```
testpred = knn(train = train, test = test, cl = as.factor(train_y), k = 50)
table(testpred, as.factor(test_y))
```

```
##
## testpred   0   1
##        0 151 125
##        1  74 124
```

After considering a tuning grid of multiple K's and controlling using repeated cross validation, we discovered that `k = 33` was the correct amount of neighbors to consider. Unfortunately, we are unable to determine feature importance from this model. It had an accuracy of 58.0168776%, a sensitivity of 38.3900929%, and a specificity of 43.1428571%.

Elastic-Net Model

We next used an elastic-net model, which combines the strengths of the Ridge model, which is able to shrink down the coefficients of parameters to make them insignificant, and the LASSO model, which is able to completely remove parameters from the model.

```
#Logistic Regression model
library(glmnet)
```

```r
library(caret)
train <- train
train[,c('nose_0300', 'nose_0500', 'phq_1000', 'fss_1000', 'gad_0800',
         'diet_0700', 'famhx_0700', 'narc_1600',
         'soclhx_1500', 'diet_0340')] <- lapply((train[,c('nose_0300', 'nose_0500',
                                                            'phq_1000', 'fss_1000', 'gad_0800',
                                                            'diet_0700', 'famhx_0700',
                                                            'narc_1600', 'soclhx_1500',
                                                            'diet_0340')]),factor)

test[,c('nose_0300', 'nose_0500', 'phq_1000', 'fss_1000',
        'gad_0800', 'diet_0700', 'famhx_0700', 'narc_1600',
        'soclhx_1500', 'diet_0340')] <- lapply((test[,c('nose_0300', 'nose_0500',
                                                          'phq_1000', 'fss_1000', 'gad_0800',
                                                          'diet_0700', 'famhx_0700', 'narc_1600',
                                                          'soclhx_1500', 'diet_0340')]), factor)
train_y <- as.factor(train_y)
test_y <- as.factor(test_y)
train <- data.matrix(train)
test <- data.matrix(test)


library(glmnet)
glm_model <- glmnet(x = train, y = train_y, family = "binomial", alpha = 0.5)
lowest_lambda <- min(glm_model$lambda)
glm_predict <- predict(glm_model, test, s = lowest_lambda, type = "class")
table(glm_predict, test_y)
```

```
##              test_y
## glm_predict   0   1
##             0 144 108
##             1  81 141
```

```r
coef(glm_model, lowest_lambda)
```

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                          s1
## (Intercept)   0.677980208
## dem_0800     -0.003126287
## fss_1000     -0.118238993
## gad_0800     -0.524163453
## phq_1000     -0.311597270
## nose_0300     0.171017798
## nose_0500     0.219527452
## diet_0340    -0.279262938
## diet_0400    -0.259375123
## diet_0700     0.944408765
## soclhx_0501   .
## soclhx_0700   0.145251425
## soclhx_0900   0.548324620
## soclhx_1500   0.379854684
## famhx_0700   -0.793379285
## ess_0900     -0.044385447
## narc_1600     0.235347514
```

```
## good_sleeper  0.320245036
```

```
(141 + 144)/((141 + 144) + (81 + 108))
```

```
## [1] 0.6012658
```

It appears that the most significant health predictors of a good sleeper is the score on the `narc_1600`, the `fss_1000`, `soclhx_0900`, and `nose_0300`. Interestingly, BMI use does not appear to have a correlation with sleep, at least according to our data and our elastic-net model. It has an accuracy of 60.1265823%, a specificity (true negative classification) of 43.2432432%, and a sensitivity (true positive classification), at around 42.7272727%

Acknowledgements

This research has been conducted using the STAGES - Stanford Technology, Analytics and Genomics in Sleep Resource funded by the Klarman Family Foundation. The investigators of the STAGES study contributed to the design and implementation of the STAGES cohort and/or provided data and/or collected biospecimens, but did not necessarily participate in the analysis or writing of this report. The full list of STAGES investigators can be found at the project website.

The National Sleep Research Resource was supported by the U.S. National Institutes of Health, National Heart Lung and Blood Institute (R24 HL114473, 75N92019R002).