

# Final Report

Will Gullion, Brian Nalley, Nate Nellis, McKade Thomas

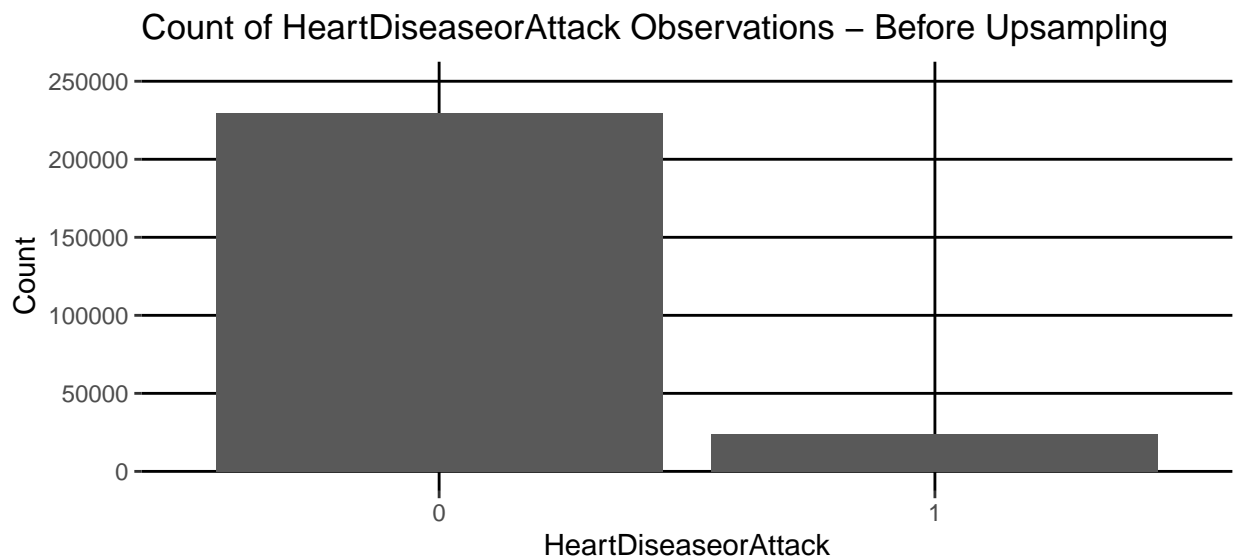
STAT 5650

## Introduction

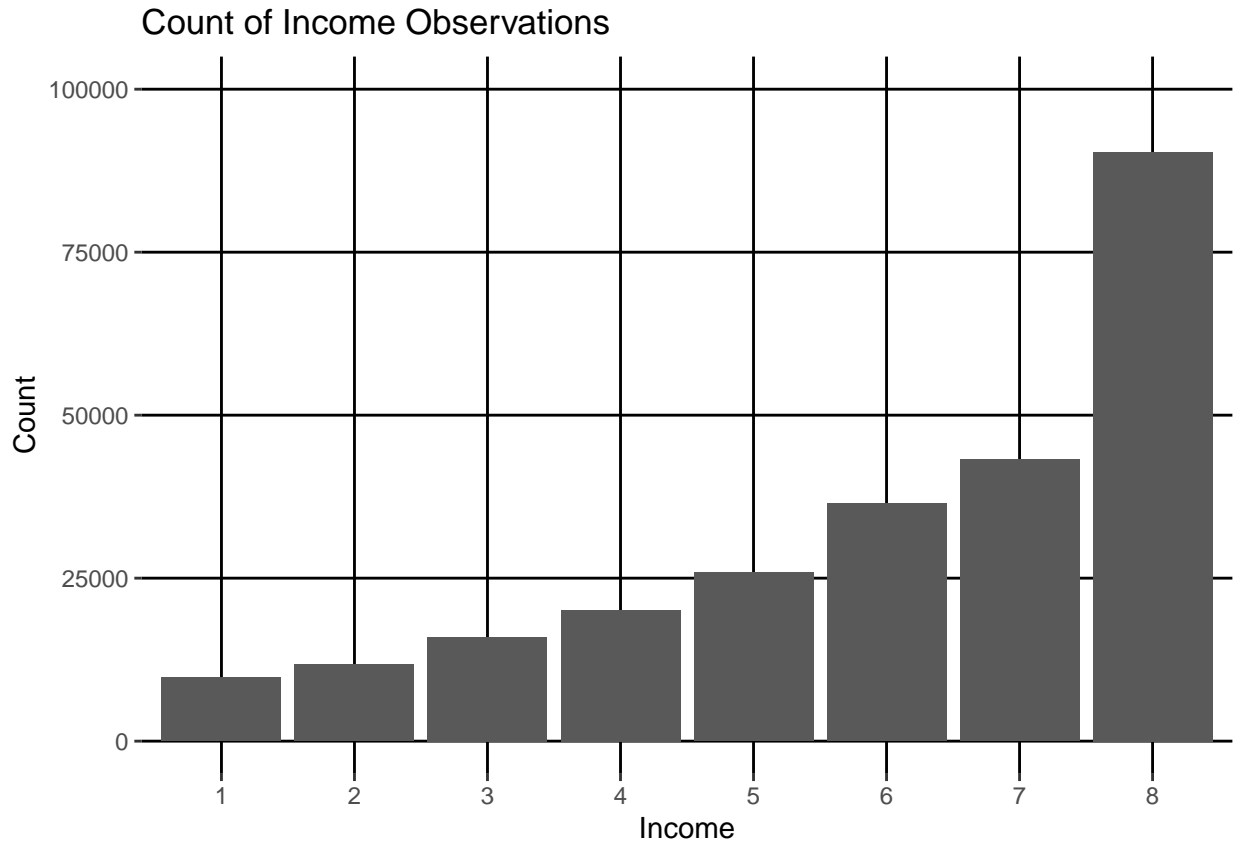
We chose to focus on data having to do with heart disease/heart attack indicators to understand which factors impact a person's probability to have a heart attack or suffer from strong levels of heart disease. Data was found on Kaggle and was modified from a CDC survey. See the *Data Dictionary* section for more details at the end of the document. The dataset contains 252,680 observations. The response variable we have is a binary one telling us if the person being interviewed suffered from a heart attack or not. There are a total of 20 available predictor variables.

## Exploratory Data Analysis

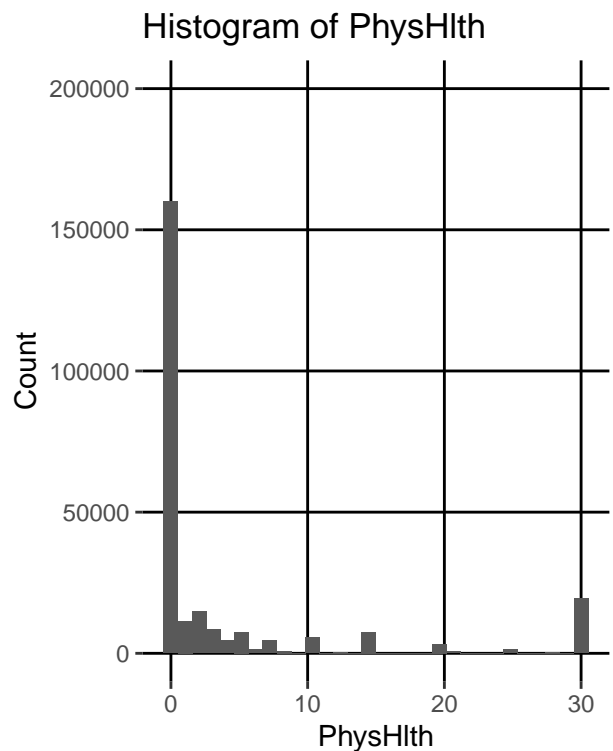
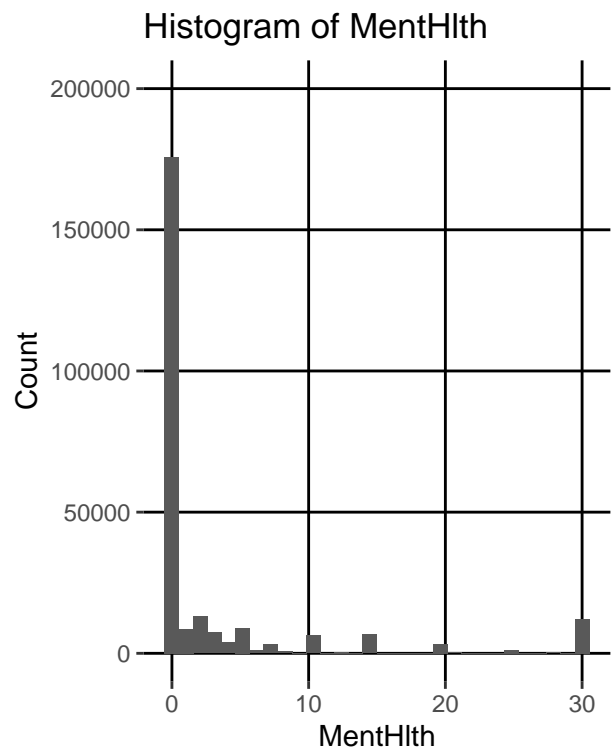
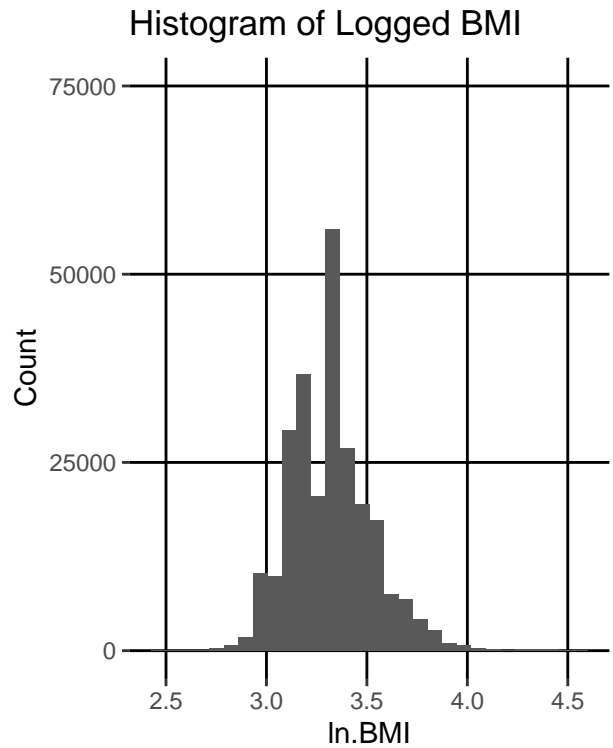
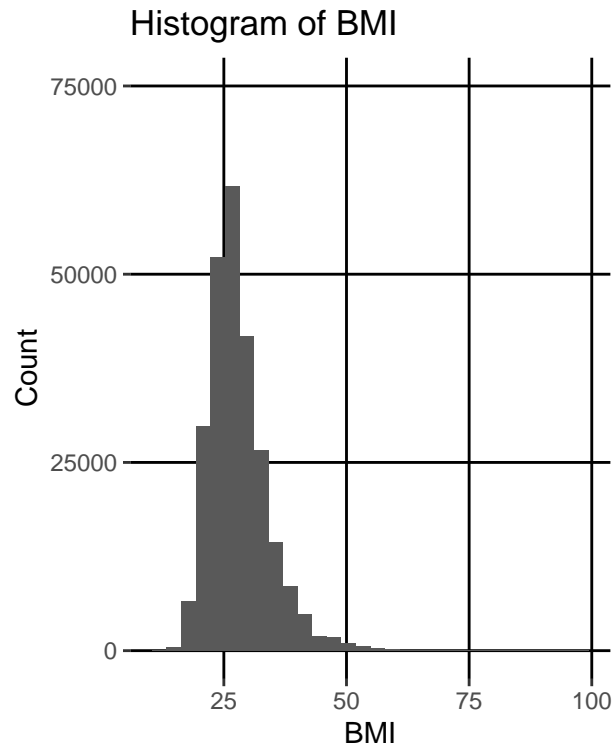
Taking a closer look at our predictor variable, we noticed that only 9.4% of participants in the survey had heart disease or a heart attack, 23,893 responded yes versus 229,787 who responded no. This means that we could just predict a 0 for all observations and we could get an accuracy of over 90%. Since we aren't concerned in this instance with a high accuracy, but actually a decent sensitivity (getting accurate results when someone is at risk of a heart attack or heart disease), we utilized Upsampling in some of our methods to skew the model to predict the results we wanted. This decreased overall accuracy, but resulted in a much better prediction accuracy of finding those people with heart disease or who had a heart attack. Even with a loss in accuracy, we felt that this was a wise move as predicting someone who has heart disease is much more important than identifying someone who doesn't have heart disease. See the below bar chart to compare the two categories visually.



When it comes to the predictor variables there are 12 binary, 3 numerical, and 5 factor variables. Having fewer numerical variables is less than desirable, especially when it comes to how some of the factor variables are coded/binning. An example of this terrible survey coding is the **Income** variable where the CDC survey sets the highest bucket capped at \$75,000+. Due to how low that is for the maximum response allowed, a full third of the observations fall into that category.



For the three numerical variables, **BMI**, **MentHlth**, and **PhysHlth**, there is a definite skewness that we can see in their distributions. This is especially true for **PhysHlth** and **MentHlth**, which are how many times in the last 30 days a survey respondent reported having poor physical or mental health respectively. A huge percentage of respondents reported zero days, 70% for **MentHlth** and 63% for **PhysHlth**. To correct for these skewed variables, we logged **BMI** to enforce normality constraints. The For **MentHlth** and **PhysHlth** we created binary variables where any value greater than 0 was coded as a 1 and was coded as a 0 otherwise. We originally attempted to log these variables but the skew wasn't reduced very much and we ended up finding that the binary variables worked out much better. These transformations were valuable in the various classification methods that we tried, even though they did reduce our numerical variables to really just the **ln.BMI** variable.



See more information on the variables in the **Data Dictionary** Section at the end of the document.

## Upsample and feature subsets from RF and LASSO

```
## UpSample to deal with class imbalance
new_train <- upSample(health_train[,2:22], as.factor(health_train[,1]), list=FALSE) %>%
  mutate(HeartDiseaseorAttack = as.numeric(Class)-1)

# new_train$HeartDiseaseorAttack <- new_train$Class
# new_train <- new_train[,-22]

## Test Set Upsample ##
new_test <- upSample(health_test[,2:22], as.factor(health_test[,1]), list = FALSE) %>%
  mutate(HeartDiseaseorAttack = as.numeric(Class)-1)
# new_test$HeartDiseaseorAttack <- new_test$Class
# new_test <- new_test[,-22]

rf_best_features <- new_train %>% dplyr::select(HeartDiseaseorAttack, Age, GenHlth, ln.BMI, Income, HighBP,
                                                HighChol, CholCheck, Smoker,
                                                Stroke, Diabetes, HvyAlcoholConsump,
                                                NoDocbcCost, GenHlth, DiffWalk,
                                                Sex, Age, Income, bin.PhysHlth)

lasso_best_features <- new_train %>% dplyr::select(HeartDiseaseorAttack, HighBP,
                                                HighChol, CholCheck, Smoker,
                                                Stroke, Diabetes, HvyAlcoholConsump,
                                                NoDocbcCost, GenHlth, DiffWalk,
                                                Sex, Age, Income, bin.PhysHlth)

lasso_test_features <- new_test %>% dplyr::select(HeartDiseaseorAttack, HighBP,
                                                HighChol, CholCheck, Smoker,
                                                Stroke, Diabetes, HvyAlcoholConsump,
                                                NoDocbcCost, GenHlth, DiffWalk,
                                                Sex, Age, Income, bin.PhysHlth)
```

To determine which variables were most important to our analysis, we performed variable selection both through random forest and LASSO logistic regression. The random forest determined that **Age**, **GenHlth**, **ln.BMI**, **Income**, **HighBP**, **Education** and **HighChol** were the most important variables. While the LASSO preferred **HighBP**, **HighChol**, **CholCheck**, **Smoker**, **Stroke**, **Diabetes**, **HvyAlcoholConsump**, **NoDocbcCost**, **GenHlth**, **DiffWalk**, **Sex**, **Age**, **Income** and **bin.PhysHlth**. So we see that while there is some overlap between the two, the LASSO selected quite a few more variables as being important to the analysis. As another method for variable selection, we conducted a LASSO linear regression. Comparing the minimum and 1-SE values, test set predictions were nearly identical for the two models, each with a percent correctly classified of just north of 77% and specificity of 79%. However, the 1-SE version selected fewer variables as important compared with the minimum model selecting all variables except for **AnyHealthcare** and **ln.BMI**. These recommended variables will give us another option to construct potentially smaller and more interpretable models and, at a minimum, will give us alternatives to compare the full models with.

## LDA

We conducted linear discriminant analysis (LDA) on our health data and obtained a cross-validated predictive accuracy of 76.99% and a specificity of 79.91%. Cross-validated predictive accuracy and specificity decreased slightly, to 75.23% and 79.07% respectively using the random forest selected subset of predictive variables.

Interestingly, using variables selected by LASSO logistic regression, both values increased slightly to 77.09% and 80.01% respectively.

## QDA

We also conducted quadratic discriminant analysis (QDA) which provided an interesting mix of results. the cross-validated percent correctly classified is relatively low at 73.15%; however, the cross-validated specificity is quite high at 88.25%. Using the variables selected as most important by random forest, QDA cross-validated predictive accuracy decreased slightly to 71.4% while specificity increased to 89.99%. We observed a similar pattern when the predictor variables selected by LASSO logistic regression, with a cross-validated predictive accuracy decreasing to 72.16% and specificity increasing to 89.76%.

## LASSO Logistic Regression

In addition to variable selection, we also used LASSO logistic regression to make predictions on our test. We first compared the minimum and 1-SE values for lambda, for which the test set predictions were nearly identical for the two models, each with a percent correctly classified of just north of 77% and specificity of 79%. However, the 1-SE version selected fewer variables as important compared with the minimum model which selected all variables except for **AnyHealthcare** and **ln.BMI**. Because the 1-SE version gave us a more parsimonious model for approximately equivalent prediction accuracy, we chose to proceed with this version both for variable selection as well as to make predictions on our test set.

## Random Forests

We built the random forest using the predictors that the variable importance plot indicated had the largest effects on the random forest’s classification ability. Those variables ended up being Age, GenHlth, ln.BMI, Income, HighBP, Education, and HighChol. After creating the random forest, we found that it classified approximately 75% of the training set correctly. The next step was to validate the random forest using the testing set. We saw similar performance, with just under 75% correctly classified. Perhaps more important than the PCC was the sensitivity and specificity which came in at 79% and 70%, respectively. We see a huge increase in the sensitivity, thanks to our upsampling technique.

## Summary Table

Table 1: Prediction Results for Individual Methods

	PCC	Specificity
LDA	76.99	79.91
LDA w/RF-Selected Variables	75.23	79.07
LDA w/LASSO-Selected Variables	77.09	80.01
QDA	73.15	88.25
QDA w/RF-Selected Variables	71.40	89.99
QDA w/LASSO-Selected Variables	72.16	89.76
LASSO Logistic Regression	77.10	79.53
RF w/ RF-Selected Variables	74.73	69.69
RF w/ LASSO-Selected Variables	79.11	74.45

# Data Dictionary

Data found by us at <https://www.kaggle.com/alexteboul/heart-disease-health-indicators-dataset>

Description of variables found in [https://www.cdc.gov/brfss/annual\\_data/2015/pdf/codebook15\\_llcp.pdf](https://www.cdc.gov/brfss/annual_data/2015/pdf/codebook15_llcp.pdf)

## Response Variable

**HeartDiseaseorAttack**, binary

```
summary(health$HeartDiseaseorAttack)
```

```
##          0          1
## 229787  23893
```

## Predictor Variables

**HighBP**, binary

- Adults who have been told they have high blood pressure by a doctor, nurse, or other health professional

```
summary(health$HighBP)
```

```
##          0          1
## 144851 108829
```

**HighChol**, binary

- Have you EVER been told by a doctor, nurse or other health professional that your blood cholesterol is high?

```
summary(health$HighChol)
```

```
##          0          1
## 146089 107591
```

**CholCheck**, binary

- Cholesterol check within past five years

```
summary(health$CholCheck)
```

```
##          0          1
##   9470 244210
```

**BMI**, numerical

```
summary(health$BMI)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   12.00   24.00   27.00   28.38   31.00   98.00
```

```
hist(as.numeric(health$BMI))
```



**ln.BMI**, numerical

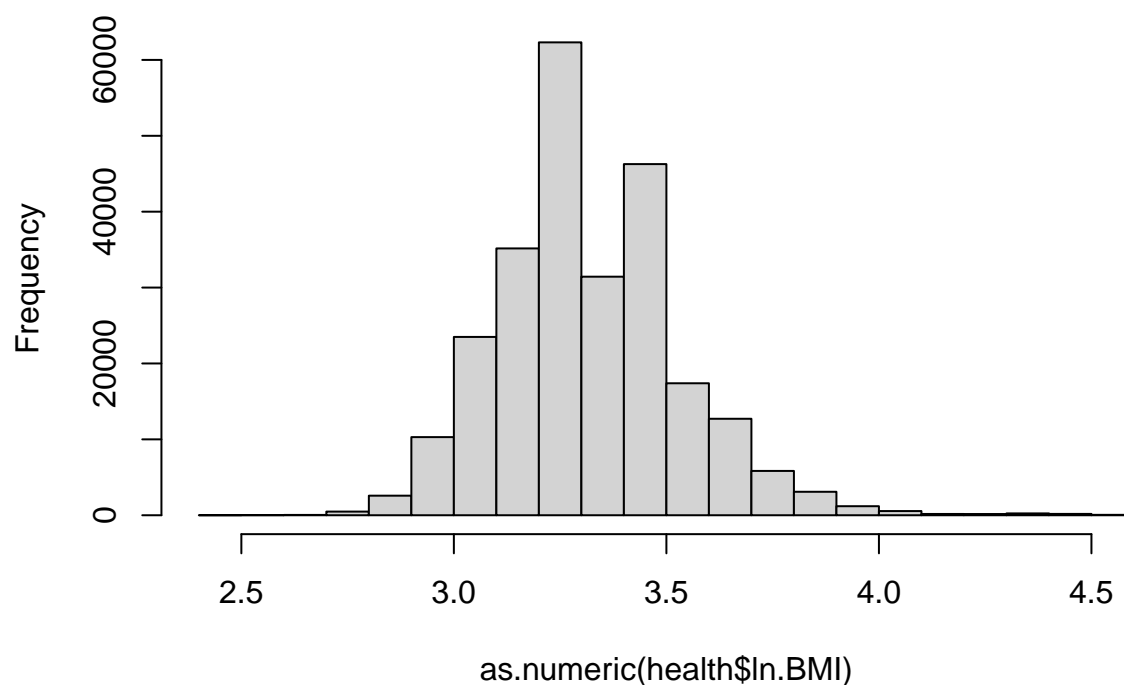
- Logged version of the BMI variable to reduce the impact of outliers and enforce better normality constraints.

```
summary(health$ln.BMI)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	2.485	3.178	3.296	3.323	3.434	4.585

```
hist(as.numeric(health$ln.BMI))
```

## Histogram of as.numeric(health\$ln.BMI)



**Smoker**, binary

- Have you smoked at least 100 cigarettes in your entire life?

```
summary(health$Smoker)
```

```
##      0      1  
## 141257 112423
```

**Stroke**, binary

```
summary(health$Stroke)
```

```
##      0      1  
## 243388 10292
```

**Diabetes**, factor

- 0 is no diabetes,

- 1 is pre-diabetes

- 2 is diabetes

```
summary(health$Diabetes)
```

```
##      0      1      2  
## 213703  4631 35346
```



**PhysActivity**, binary

- During the past month, other than your regular job, did you participate in any physical activities or exercises such as running, calisthenics, golf, gardening, or walking for exercise?

```
summary(health$PhysActivity)
```

```
##          0          1  
## 61760 191920
```

**Fruits**, binary

- Consume Fruit 1 or more times per day

```
summary(health$Fruits)
```

```
##          0          1  
## 92782 160898
```

**Veggies**, binary

- Consume Vegetables 1 or more times per day

```
summary(health$Veggies)
```

```
##          0          1  
## 47839 205841
```

**HvyAlcoholConsum**, binary

- Heavy drinkers (adult men having more than 14 drinks per week and adult women having more than 7 drinks per week)

```
summary(health$HvyAlcoholConsum)
```

```
##          0          1  
## 239424 14256
```

**AnyHealthcare**, binary

- Do you have any kind of health care coverage, including health insurance, prepaid plans such as HMOs, or government plans such as Medicare, or Indian Health Service?

```
summary(health$AnyHealthcare)
```

```
##          0          1  
## 12417 241263
```

**NoDocbcCost**, binary

- Was there a time in the past 12 months when you needed to see a doctor but could not because of cost?

```
summary(health$NoDocbcCost)
```

```
##          0          1  
## 232326 21354
```

**GenHlth**, factor

- Would you say that in general your health is:
- 1 = Excellent
- 2 = Very Good
- 3 = Good
- 4 = Fair
- 5 = Poor

```
summary(health$GenHlth)
```

```
##      1      2      3      4      5
## 45299 89084 75646 31570 12081
```

**MentHlth**, numerical

- Now thinking about your mental health, which includes stress, depression, and problems with emotions, for how many days during the past 30 days was your mental health not good?

```
summary(health$MentHlth)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   3.185   2.000   30.000
```

**bin.MentHlth**, binary

- Binary version of **MentHlth** where 0 days is coded as 0 and any number of days is coded as 1.

```
summary(health$bin.MentHlth)
```

```
##      0      1
## 175680  78000
```

**PhysHlth**, numerical

- Now thinking about your physical health, which includes physical illness and injury, for how many days during the past 30 days was your physical health not good?

```
summary(health$PhysHlth)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   4.242   3.000   30.000
```

**bin.PhysHlth**, binary

- Binary version of **PhysHlth** where 0 days is coded as 0 and any number of days is coded as 1.

```
summary(health$bin.PhysHlth)
```

```
##      0      1
## 160052  93628
```

**DiffWalk**, binary

- Do you have serious difficulty walking or climbing stairs?

```
summary(health$DiffWalk)
```

```
##          0          1  
## 211005  42675
```

Sex, binary  
- 0 = Female  
- 1 = Male

```
summary(health$Sex)
```

```
##          F          M  
## 141974 111706
```

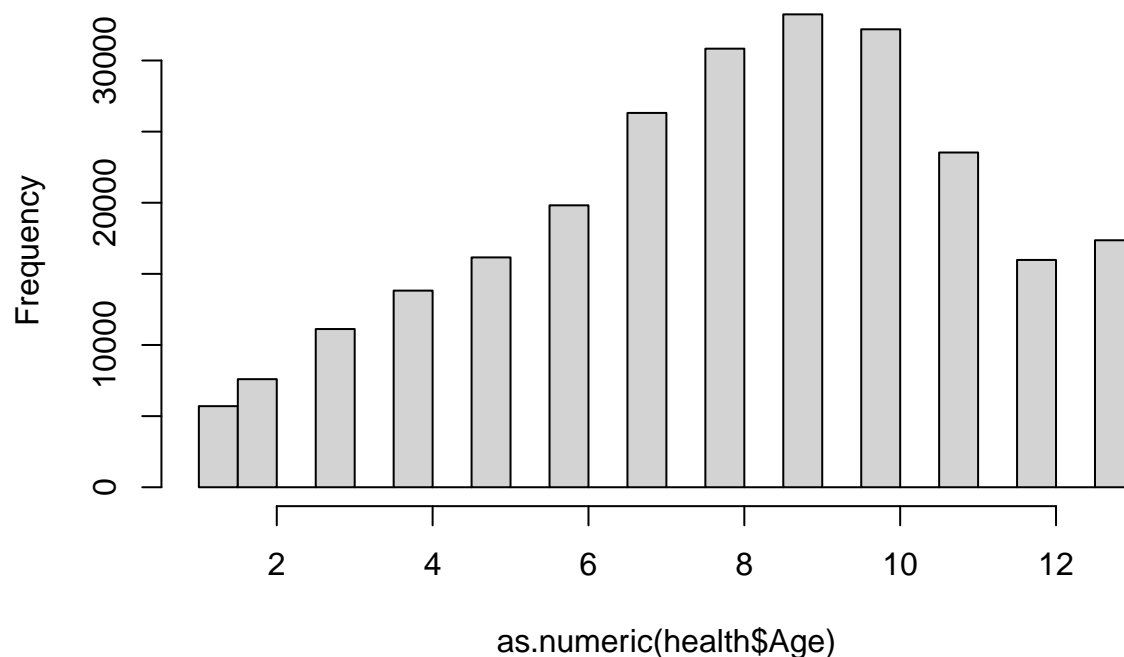
Age, factor  
- 1 Age 18 to 24  
- 2 Age 25 to 29  
- 3 Age 30 to 34  
- 4 Age 35 to 39  
- 5 Age 40 to 44  
- 6 Age 45 to 49  
- 7 Age 50 to 54  
- 8 Age 55 to 59  
- 9 Age 60 to 64  
- 10 Age 65 to 69  
- 11 Age 70 to 74  
- 12 Age 75 to 79  
- 13 Age 80 or older  
- 14 Don't Know / Refused to answer (I removed these as well)

```
summary(health$Age)
```

```
##          1          2          3          4          5          6          7          8          9         10         11         12         13  
##   5700   7598  11123  13823  16157  19819  26314  30832  33244  32194  23533  15980  17363
```

```
hist(as.numeric(health$Age))
```

## Histogram of as.numeric(health\$Age)



### Education, factor

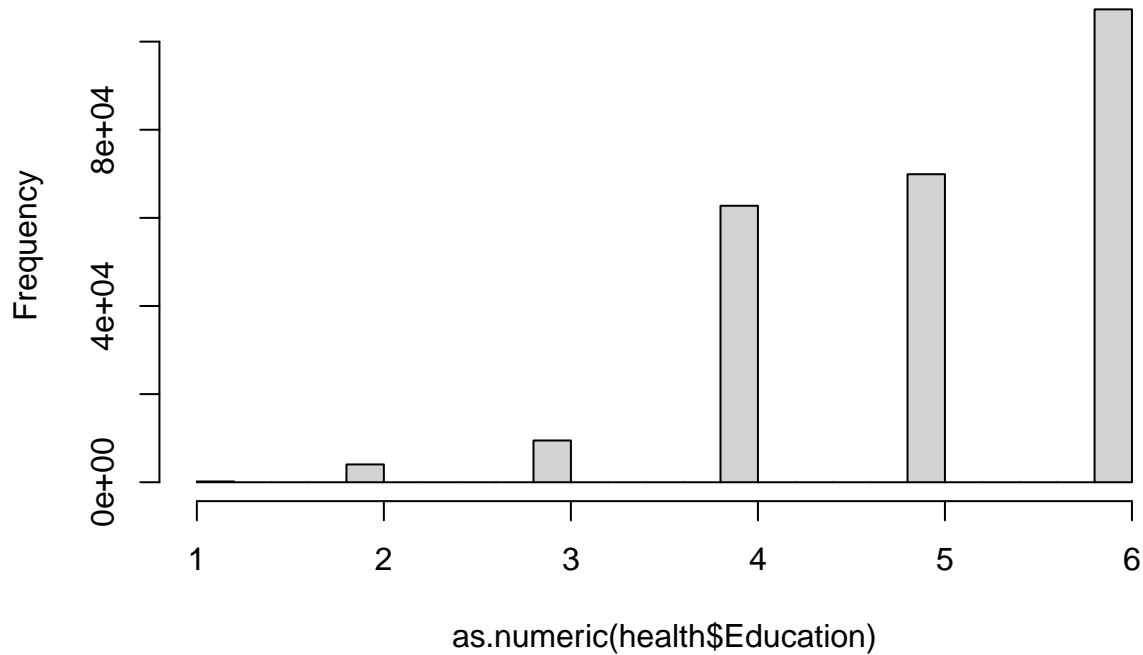
- What is the highest grade or year of school you completed?
- 1 Never attended school or only kindergarten
- 2 Grades 1 through 8 (Elementary)
- 3 Grades 9 through 11 (Some high school)
- 4 Grade 12 or GED (High school graduate)
- 5 College 1 year to 3 years (Some college or technical school)
- 6 College 4 years or more (College graduate)

```
summary(health$Education)
```

```
##      1      2      3      4      5      6
##  174  4043  9478 62750 69910 107325
```

```
hist(as.numeric(health$Education))
```

## Histogram of as.numeric(health\$Education)



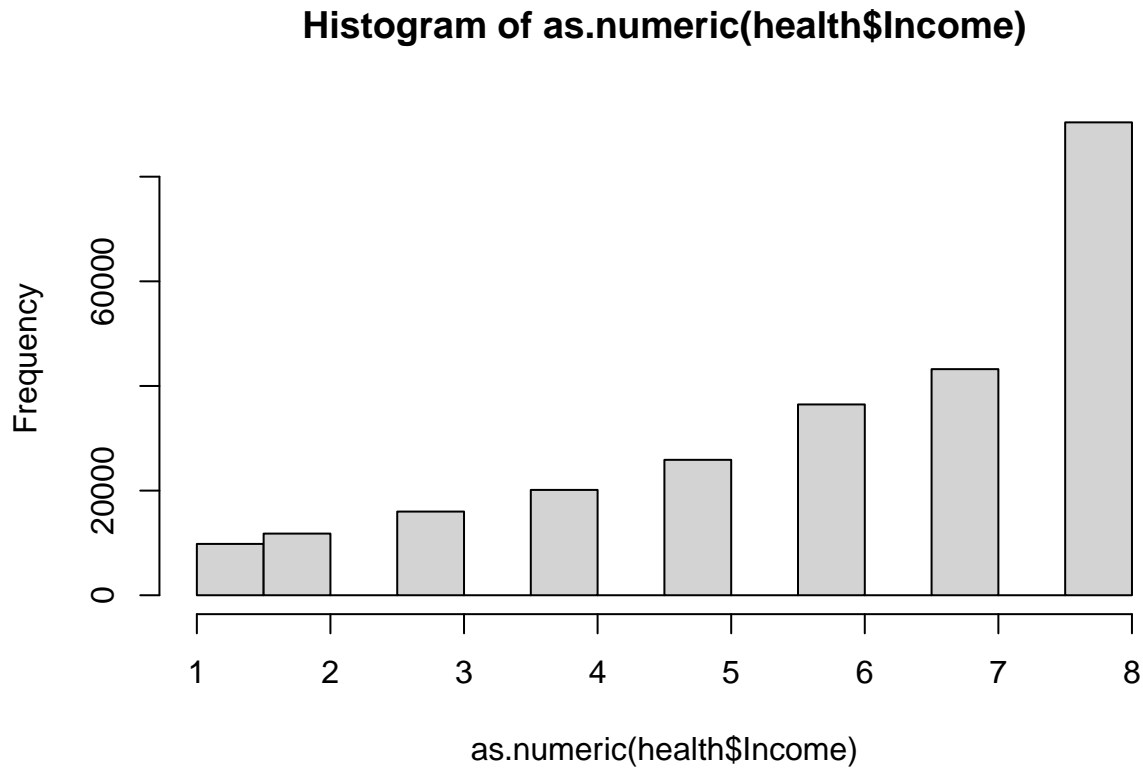
### Income, factor

- Is your annual household income from all sources
- 1 Less than \$10,000
- 2 Less than \$15,000 (\$10,000 to less than \$15,000)
- 3 Less than \$20,000 (\$15,000 to less than \$20,000)
- 4 Less than \$25,000 (\$20,000 to less than \$25,000)
- 5 Less than \$35,000 (\$25,000 to less than \$35,000)
- 6 Less than \$50,000 (\$35,000 to less than \$50,000)
- 7 Less than \$75,000 (\$50,000 to less than \$75,000)
- 8 \$75,000 or more

```
summary(health$Income)
```

```
##      1      2      3      4      5      6      7      8
## 9811 11783 15994 20135 25883 36470 43219 90385
```

```
hist(as.numeric(health$Income))
```



## Appendix: All code for this report

```
knitr::opts_chunk$set(echo = TRUE)
library(dplyr)
library(MASS)
library(verification)
library(caTools)
library(klaR)
library(randomForest)
library(glmnet)
library(ggplot2)
library(knitr)
library(gridExtra)
library(kableExtra)
library(caret)
health <- read.csv("heart_disease_health_indicators.csv")

health <- health %>% mutate(
  HeartDiseaseorAttack = as.factor(HeartDiseaseorAttack),
  HighBP = as.factor(HighBP), # LASSO approved
  HighChol = as.factor(HighChol), # LASSO approved
  CholCheck = as.factor(CholCheck),
  ln.BMI = log(BMI), # RF approved
```

```

Smoker = as.factor(Smoker), # LASSO approved
Stroke = as.factor(Stroke), # LASSO 1-SE only, RF approved
Diabetes = as.factor(Diabetes), # LASSO approved, RF approved
PhysActivity = as.factor(PhysActivity),
Fruits = as.factor(Fruits),
Veggies = as.factor(Veggies),
HvyAlcoholConsump = as.factor(HvyAlcoholConsump),
AnyHealthcare = as.factor(AnyHealthcare),
NoDocbcCost = as.factor(NoDocbcCost), # LASSO approved
GenHlth = as.factor(GenHlth), # LASSO approved, RF approved
ln.MentHlth = log(MentHlth + 1), # Probably shouldn't use
bin.MentHlth = as.factor(ifelse(MentHlth == 0, 0, 1)), #RF approved
fac.MentHlth = as.factor(MentHlth), # Probably shouldn't use
ln.PhysHlth = log(PhysHlth + 1), # Probably shouldn't use
bin.PhysHlth = as.factor(ifelse(PhysHlth == 0, 0, 1)), # LASSO approved, RF approved
fac.PhysHlth = as.factor(PhysHlth), # Probably shouldn't use
DiffWalk = as.factor(DiffWalk), # LASSO , RF approved
Sex = as.factor(ifelse(Sex == 0, "F", "M")), # LASSO approved, RF approved
Age = as.factor(Age), # LASSO approved, RF approved
Education = as.factor(Education), # RF approved
Income = as.factor(Income) # LASSO approved, RF approved
)

best_features <- c(1:4, 6:15, 18:23, 25, 28)
health_subset <- health[,best_features]

# Train and Test Sets
health2_split <- sample.split(health_subset, SplitRatio = 0.3)
health_test <- subset(health_subset, health2_split == TRUE)
health_train <- subset(health_subset, health2_split == FALSE)

ggplot(health, aes(HeartDiseaseorAttack)) +
  geom_bar() +
  scale_y_continuous(limits = c(0, 250000), breaks = seq(0, 250000, by = 50000)) +
  ylab("Count") +
  ggtitle("Count of HeartDiseaseorAttack Observations - Before Upsampling") +
  theme(panel.grid.major = element_line(colour = "black"),
        panel.background = element_rect(fill = NA))
ggplot(health, aes(Income)) +
  geom_bar() +
  scale_y_continuous(limits = c(0, 100000), breaks = seq(0, 100000, by = 25000)) +
  ylab("Count") +
  ggtitle("Count of Income Observations") +
  theme(panel.grid.major = element_line(colour = "black"),
        panel.background = element_rect(fill = NA))
grid.arrange(
ggplot(health, aes(BMI)) +
  geom_histogram(bins=30) +
  scale_y_continuous(limits = c(0, 75000), breaks = seq(0, 75000, by = 25000)) +
  ylab("Count") +
  ggtitle("Histogram of BMI") +

```

```

    theme(panel.grid.major = element_line(colour = "black"),
          panel.background = element_rect(fill = NA))
  ,
  ggplot(health, aes(ln.BMI)) +
    geom_histogram(bins=30) +
    scale_y_continuous(limits = c(0, 75000), breaks = seq(0, 75000, by = 25000)) +
    ylab("Count") +
    ggtitle("Histogram of Logged BMI") +
    theme(panel.grid.major = element_line(colour = "black"),
          panel.background = element_rect(fill = NA))
  ,
  nrow = 1)
grid.arrange(
  ggplot(health, aes(MentHlth)) +
    geom_histogram(bins=30) +
    scale_y_continuous(limits = c(0, 200000), breaks = seq(0, 200000, by = 50000)) +
    ylab("Count") +
    ggtitle("Histogram of MentHlth") +
    theme(panel.grid.major = element_line(colour = "black"),
          panel.background = element_rect(fill = NA))
  ,
  ggplot(health, aes(PhysHlth)) +
    geom_histogram(bins=30) +
    scale_y_continuous(limits = c(0, 200000), breaks = seq(0, 200000, by = 50000)) +
    ylab("Count") +
    ggtitle("Histogram of PhysHlth") +
    theme(panel.grid.major = element_line(colour = "black"),
          panel.background = element_rect(fill = NA))
  ,
  nrow = 1)
## UpSample to deal with class imbalance
new_train <- upSample(health_train[,2:22], as.factor(health_train[,1]), list=FALSE) %>%
  mutate(HeartDiseaseorAttack = as.numeric(Class)-1)

# new_train$HeartDiseaseorAttack <- new_train$Class
# new_train <- new_train[, -22]

## Test Set Upsample ##
new_test <- upSample(health_test[,2:22], as.factor(health_test[,1]), list = FALSE) %>%
  mutate(HeartDiseaseorAttack = as.numeric(Class)-1)
# new_test$HeartDiseaseorAttack <- new_test$Class
# new_test <- new_test[, -22]

rf_best_features <- new_train %>% dplyr::select(HeartDiseaseorAttack, Age, GenHlth, ln.BMI, Income, HighBP,
rf_test_features <- new_test %>% dplyr::select(HeartDiseaseorAttack, Age, GenHlth, ln.BMI, Income, HighBP,
lasso_best_features <- new_train %>% dplyr::select(HeartDiseaseorAttack, HighBP,
                                                    HighChol, CholCheck, Smoker,
                                                    Stroke, Diabetes, HvyAlcoholConsump,
                                                    NoDocbcCost, GenHlth, DiffWalk,
                                                    Sex, Age, Income, bin.PhysHlth)

```



```

lasso_test_features <- new_test %>% dplyr::select(HeartDiseaseorAttack, HighBP,
                                                HighChol, CholCheck, Smoker,
                                                Stroke, Diabetes, HvyAlcoholConsump,
                                                NoDocbcCost, GenHlth, DiffWalk,
                                                Sex, Age, Income, bin.PhysHlth)

set.seed(89729)
train_sample <- sample_n(rf_best_features, 10000)
test_sample <- sample_n(rf_test_features, 10000)

rf <- randomForest(as.factor(HeartDiseaseorAttack) ~ ., data = train_sample, n.trees = 100, importance = TRUE)

class.sum(train_sample$HeartDiseaseorAttack, predict(rf, type="prob")[,2], cap = "Training Data RF Performance")
class.sum(test_sample$HeartDiseaseorAttack, predict(rf, test_sample, type="prob")[,2], cap = "Test Data RF Performance")

# Random Forest with LASSO selected variables
train_sample.lasso <- sample_n(lasso_test_features, 10000)
test_sample.lasso <- sample_n(lasso_test_features, 10000)

rf.lasso <- randomForest(as.factor(HeartDiseaseorAttack) ~ ., data = train_sample.lasso, n.trees = 100)

class.sum(train_sample.lasso$HeartDiseaseorAttack, predict(rf.lasso, type="prob")[,2], cap = "Training Data RF Performance")
class.sum(test_sample.lasso$HeartDiseaseorAttack, predict(rf.lasso, test_sample.lasso, type="prob")[,2], cap = "Test Data RF Performance")

results <- data.frame(PCC = c(76.99, 75.23, 77.09, 73.15, 71.40, 72.16, 77.10, 74.73, 79.11),
                      Specificity = c(79.91, 79.07, 80.01, 88.25, 89.99, 89.76,
                                       79.53, 69.69, 74.45))

rownames(results) <- c("LDA", "LDA w/RF-Selected Variables",
                      "LDA w/LASSO-Selected Variables", "QDA", "QDA w/RF-Selected Variables",
                      "QDA w/LASSO-Selected Variables", "LASSO Logistic Regression", "RF w/ RF-Selected Variables", "RF w/ LASSO-Selected Variables")

results.table <- results %>%
  kbl(caption = "Prediction Results for Individual Methods") %>%
  kable_classic(full_width = FALSE, html_font = "Cambria",
                latex_options = "HOLD_position")

results.table
summary(health$HeartDiseaseorAttack)
summary(health$HighBP)
summary(health$HighChol)
summary(health$CholCheck)
summary(health$BMI)
hist(as.numeric(health$BMI))
summary(health$ln.BMI)
hist(as.numeric(health$ln.BMI))
summary(health$Smoker)
summary(health$Stroke)
summary(health$Diabetes)
summary(health$PhysActivity)
summary(health$Fruits)
summary(health$Veggies)
summary(health$HvyAlcoholConsump)
summary(health$AnyHealthcare)

```

```
summary(health$NoDocbcCost)
summary(health$GenHlth)
summary(health$MentHlth)
summary(health$bin.MentHlth)
summary(health$PhysHlth)
summary(health$bin.PhysHlth)
summary(health$DiffWalk)
summary(health$Sex)
summary(health$Age)
hist(as.numeric(health$Age))
summary(health$Education)
hist(as.numeric(health$Education))
summary(health$Income)
hist(as.numeric(health$Income))
```