STA303 Final Project

Christopher Jung

2024-4-06-2:34pm

Introduction

There is a concerning increase of brain stroke over the years. Due to the unawareness of the significance of other symptoms to the disease, the relationship is often overlooked. Diabetes, BMI, and heart disease are significant factors that can cause a stroke. High blood pressure and high cholesterol are also some underlying factors. The goal of the analysis is to study the significance of these 5 variables to brain stroke. In past literature, it is stated that Diabetes, BMI, and heart disease have positive relationship with stroke and aligns with this study. Existing literature also hints heart disease and high cholesterol may be confounding variables which aligns with this analysis.

Methods

Study Population

The original data used for this study comes from The Behavioral Risk Factor Surveillance System (BRFSS) survey from 2015 that is conducted annually to study health-related risk factors among Americans. The cleaned version of the data for this study contains 253680 observations, each observation representing an individual.

Results

Inputting the dataset

```
#number of rows ie observations
nrow(health_data)
```

```
## [1] 253680
```

```
#indicates no missing values
sum(is.na(health_data))
```

```
## [1] 0
```

```
head(health_data)
```

```
##   Diabetes_012 HighBP HighChol CholCheck BMI Smoker Stroke HeartDiseaseorAttack
## 1            0      1        1         1  40      1      0                     0
## 2            0      0        0         0  25      1      0                     0
## 3            0      1        1         1  28      0      0                     0
## 4            0      1        0         1  27      0      0                     0
## 5            0      1        1         1  24      0      0                     0
## 6            0      1        1         1  25      1      0                     0
##   PhysActivity Fruits Veggies HvyAlcoholConsump AnyHealthcare NoDocbcCost
## 1            0      0       1                 0             1           0
## 2            1      0       0                 0             0           1
## 3            0      1       0                 0             1           1
## 4            1      1       1                 0             1           0
## 5            1      1       1                 0             1           0
## 6            1      1       1                 0             1           0
##   GenHlth MentHlth PhysHlth DiffWalk Sex Age Education Income
## 1       5       18       15        1   0   9         4      3
## 2       3        0        0        0   0   7         6      1
## 3       5       30       30        1   0   9         4      8
## 4       2        0        0        0   0  11         3      6
## 5       2        3        0        0   0  11         5      4
## 6       2        0        2        0   1  10         6      8
```

change the MentHlth and PhysHlth to binary factors variables

Changing the data from a 30 level categorical variable to binary categorical variable. Since the number increased from 0 to 30 in ordinal order, 0 to 15 was set to 0, and 16 to 30 was set to 1.

The type of value was set as numerical which it should be factor variables since they are categorical variables so the variable type was changed to factors.

```
glimpse(health_data)
```

```
## Rows: 253,680
## Columns: 22
## $ Diabetes_012         <fct> 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 2, 0, 0, 2, 0, 0, 0~
## $ HighBP               <fct> 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 1, 1~
## $ HighChol             <fct> 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1~
## $ CholCheck            <fct> 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ BMI                  <dbl> 40, 25, 28, 27, 24, 25, 30, 25, 30, 24, 25, 34, 2~
## $ Smoker               <fct> 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 0, 0~
## $ Stroke               <fct> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0~
## $ HeartDiseaseorAttack <fct> 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0~
## $ PhysActivity         <fct> 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1~
## $ Fruits               <fct> 0, 0, 1, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1~
## $ Veggies              <fct> 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1~
## $ HvyAlcoholConsump    <fct> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ AnyHealthcare        <fct> 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ NoDocbcCost          <fct> 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0~
## $ GenHlth              <fct> 5, 3, 5, 2, 2, 2, 3, 3, 5, 2, 3, 3, 3, 4, 4, 2, 3~
## $ MentHlth             <fct> 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0~
## $ PhysHlth             <fct> 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0~
## $ DiffWalk             <fct> 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0~
## $ Sex                  <fct> 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0~
```

```
## $ Age                 <fct> 9, 7, 9, 11, 11, 10, 9, 11, 9, 8, 13, 10, 7, 11, ~
## $ Education           <fct> 4, 6, 4, 3, 5, 6, 6, 4, 5, 4, 6, 5, 5, 4, 6, 6, 4~
## $ Income              <fct> 3, 1, 8, 6, 4, 8, 7, 4, 1, 3, 8, 1, 7, 6, 2, 8, 3~
```

EDA

The response variable: stroke

The risk factors: - diabetes - bmi - heart disease

The possible confounders: - high cholesterol - high blood pressure

```
df <- data.frame(Stroke = c("No Stroke", "Stroke"),
                 Probability = c(nrow(health_data[health_data$Stroke==0,])/nrow(health_data), nrow(heal

knitr::kable(df, caption = "Stroke Probability")
```

Table 1: Stroke Probability

Stroke	Probability
No Stroke	0.9594292
Stroke	0.0405708

Getting the probability table for stroke vs no stroke

```
#probability table for diabetes
no_dia=health_data[health_data$Diabetes_012==0,]
pre_dia = health_data[health_data$Diabetes_012==1,]
dia = health_data[health_data$Diabetes_012==2,]

df <- data.frame(Stroke = c("No Stroke with no diabetes", "Stroke with no diabetes", "No Stroke with pr
                 Probability = c(nrow(no_dia[no_dia$Stroke == 0,])/ nrow(no_dia), nrow(no_dia[no_dia$St

knitr::kable(df, caption = "Stroke Probability for Diabetes")
```

Table 2: Stroke Probability for Diabetes

Stroke	Probability
No Stroke with no diabetes	0.9683720
Stroke with no diabetes	0.0316280
No Stroke with pre-diabetes	0.9427769
Stroke with pre-diabetes	0.0572231
No Stroke with diabetes	0.9075426
Stroke with diabetes	0.0924574

Getting the probability table for stroke vs no stroke for diabetes

```r
#probability table for heart disease
no_heart_disease =health_data[health_data$HeartDiseaseorAttack==0,]
heart_disease = health_data[health_data$HeartDiseaseorAttack==1,]


df <- data.frame(Stroke = c("No Stroke with no heart disease/attack", "Stroke with no heart disease/atta
                 Probability = c(nrow(no_heart_disease[no_heart_disease$Stroke == 0,])/ nrow(no_heart_di

knitr::kable(df, caption = "Stroke Probability for Heart Disease")
```

Table 3: Stroke Probability for Heart Disease

Stroke	Probability
No Stroke with no heart disease/attack	0.9723440
Stroke with no heart disease/attack	0.0276560
No Stroke with heart disease/attack	0.8352237
Stroke with heart disease/attack	0.1647763

Getting the probability table for stroke vs no stroke for heart disease/attack

```r
#probability table for high blood pressure
no_highBP =health_data[health_data$HighBP==0,]
highBP = health_data[health_data$HighBP==1,]


df <- data.frame(Stroke = c("No Stroke with no high blood pressure", "Stroke with no high blood pressur
                 Probability = c(nrow(no_highBP[no_highBP$Stroke == 0,])/ nrow(no_highBP), nrow(no_highB

knitr::kable(df, caption = "Stroke Probability for Blood Pressure")
```

Table 4: Stroke Probability for Blood Pressure

Stroke	Probability
No Stroke with no high blood pressure	0.981588
Stroke with no high blood pressure	0.018412
No Stroke with high blood pressure	0.929936
Stroke with high blood pressure	0.070064

Getting the probability table for stroke vs no stroke for high blood pressure

```r
#probability table for high cholesterol
no_highChol = health_data[health_data$HighChol==0,]
highChol = health_data[health_data$HighChol==1,]


df <- data.frame(Stroke = c("No Stroke with no high cholesterol", "Stroke with no high cholesterol", "No
                 Probability = c(nrow(no_highChol[no_highChol$Stroke == 0,])/ nrow(no_highChol), nrow(no

knitr::kable(df, caption = "Stroke Probability for Cholesterol")
```

Table 5: Stroke Probability for Cholesterol

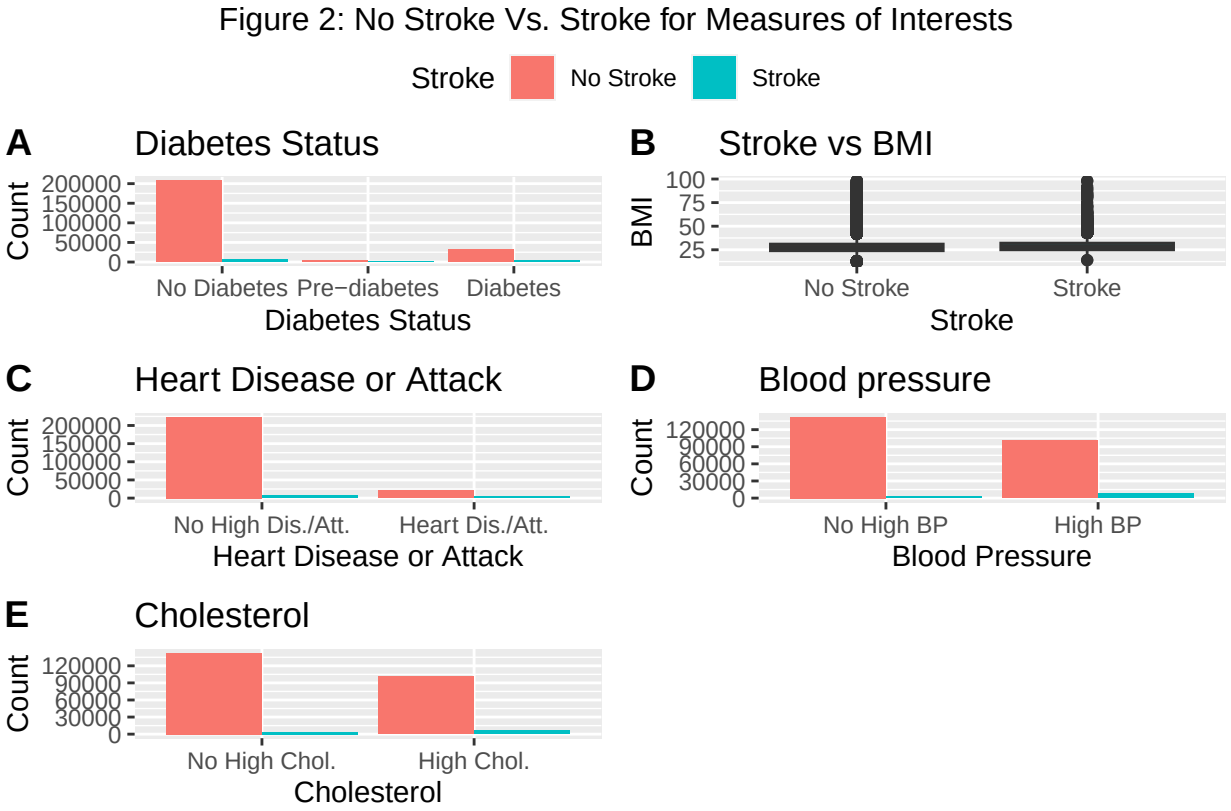
Stroke	Probability
No Stroke with no high cholesterol	0.9751111
Stroke with no high cholesterol	0.0248889
No Stroke with high cholesterol	0.9381361
Stroke with high cholesterol	0.0618639

Getting the probability table for stroke vs no stroke for high cholesterol

Numerical Summaries of the important variables in the research question

```
plot_diabetes <- ggplot(health_data, aes(x = Diabetes_012)) +
  geom_bar(aes(fill = Stroke), position = "dodge") +
  labs(title = "Diabetes Status", x="Diabetes Status", y="Count") +
  scale_x_discrete(labels=c("No Diabetes", "Pre-diabetes", "Diabetes")) +
  scale_fill_discrete(labels=c("No Stroke", "Stroke"))

plot_bmi <- ggplot(health_data, aes(x=Stroke, y=BMI, fill = Stroke)) +
  geom_boxplot() +
  labs(title = "Stroke vs BMI", x="Stroke", y="BMI") +
  scale_x_discrete(labels=c("No Stroke", "Stroke")) +
  scale_fill_discrete(labels=c("No Stroke", "Stroke"))

plot_heart <- ggplot(health_data, aes(x = HeartDiseaseorAttack)) +
  geom_bar(aes(fill = Stroke), position = "dodge") +
  labs(title = "Heart Disease or Attack", x="Heart Disease or Attack", y="Count") +
  scale_x_discrete(labels=c("No High Dis./Att.", "Heart Dis./Att.")) +
  scale_fill_discrete(labels=c("No Stroke", "Stroke"))

plot_bp <- ggplot(health_data, aes(x = HighBP)) +
  geom_bar(aes(fill = Stroke), position = "dodge") +
  labs(title = "Blood pressure", x="Blood Pressure", y="Count") +
  scale_x_discrete(labels=c("No High BP", "High BP")) +
  scale_fill_discrete(labels=c("No Stroke", "Stroke")) +
  scale_y_continuous(breaks=seq(0,200000, 30000))


plot_chol <- ggplot(health_data, aes(x = HighChol)) +
  geom_bar(aes(fill = Stroke), position = "dodge") +
  labs(title = "Cholesterol", x="Cholesterol", y="Count") +
  scale_x_discrete(labels=c("No High Chol.", "High Chol.")) +
  scale_fill_discrete(labels=c("No Stroke", "Stroke")) +
  scale_y_continuous(breaks=seq(0,200000, 30000))


figure_2 <- ggarrange(plot_diabetes, plot_bmi, plot_heart, plot_bp, plot_chol,
                    labels = c("A", "B", "C", "D", "E"),
```

```
                    ncol = 2, nrow = 3, common.legend = TRUE)

annotate_figure(figure_2, top = text_grob("Figure 2: No Stroke Vs. Stroke for Measures of Interests", si
```



This figure displays plots that show No Stroke vs Stroke for each category within different measures of inte

So about 4 percent of people have stroke in this dataset which is much lower than what we expect.

It shows that people with diabetes have the highest proportion of Stroke.

The BMI for the Stroke and Non-Stroke groups are similar, where Stroke group is slightly higher. This aligns with the second literature.

It shows that those with heart disease/attack has a higher proportion of stroke in its group than those without a heart disease/attack.

It shows that people with heart disease/attack have much higher proportion of stroke in its group than those without

It shows that people with high blood pressure (hypertension) have much higher proportion of stroke in its group than those without.

It shows that people with high cholesterol have higher proportion of stroke by in its group than those without.

Construct Full model

```
logit.mod1 <- glm(Stroke ~ Diabetes_012 + HighBP + HighChol + CholCheck + BMI + Smoker +
                    HeartDiseaseorAttack + PhysActivity + Fruits + Veggies + HvyAlcoholConsump +
                    AnyHealthcare + NoDocbcCost + GenHlth + MentHlth + PhysHlth + DiffWalk + Sex +
```

```
                    Age + Education + Income, family = binomial(link = logit), data = health_data)

summary(logit.mod1)
```

```
##
## Call:
## glm(formula = Stroke ~ Diabetes_012 + HighBP + HighChol + CholCheck +
##     BMI + Smoker + HeartDiseaseorAttack + PhysActivity + Fruits +
##     Veggies + HvyAlcoholConsump + AnyHealthcare + NoDocbcCost +
##     GenHlth + MentHlth + PhysHlth + DiffWalk + Sex + Age + Education +
##     Income, family = binomial(link = logit), data = health_data)
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -5.834758   0.389719 -14.972  < 2e-16 ***
## Diabetes_0121          -0.049304   0.067508  -0.730 0.465180
## Diabetes_0122           0.178948   0.025258   7.085 1.39e-12 ***
## HighBP1                 0.496064   0.025932  19.130  < 2e-16 ***
## HighChol1               0.202319   0.023185   8.726  < 2e-16 ***
## CholCheck1              0.285195   0.084839   3.362 0.000775 ***
## BMI                    -0.018930   0.001745 -10.850  < 2e-16 ***
## Smoker1                 0.159064   0.022373   7.110 1.16e-12 ***
## HeartDiseaseorAttack1   0.962350   0.024279  39.637  < 2e-16 ***
## PhysActivity1           0.014084   0.023551   0.598 0.549829
## Fruits1                 0.028448   0.022961   1.239 0.215360
## Veggies1               -0.154386   0.025459  -6.064 1.33e-09 ***
## HvyAlcoholConsump1     -0.107518   0.055214  -1.947 0.051500 .
## AnyHealthcare1          0.075467   0.056167   1.344 0.179069
## NoDocbcCost1            0.189732   0.034594   5.485 4.14e-08 ***
## GenHlth2                0.359950   0.054427   6.613 3.75e-11 ***
## GenHlth3                0.770223   0.053011  14.529  < 2e-16 ***
## GenHlth4                1.080686   0.056316  19.190  < 2e-16 ***
## GenHlth5                1.292615   0.063352  20.404  < 2e-16 ***
## MentHlth1               0.157159   0.033820   4.647 3.37e-06 ***
## PhysHlth1               0.115938   0.030622   3.786 0.000153 ***
## DiffWalk1               0.538483   0.026236  20.524  < 2e-16 ***
## Sex1                    0.062148   0.022493   2.763 0.005728 **
## Age2                    0.012426   0.287589   0.043 0.965536
## Age3                    0.639066   0.245498   2.603 0.009237 **
## Age4                    0.829180   0.235754   3.517 0.000436 ***
## Age5                    1.076532   0.229505   4.691 2.72e-06 ***
## Age6                    1.149271   0.225985   5.086 3.66e-07 ***
## Age7                    1.354017   0.223028   6.071 1.27e-09 ***
## Age8                    1.485110   0.222056   6.688 2.26e-11 ***
## Age9                    1.577917   0.221644   7.119 1.09e-12 ***
## Age10                   1.706374   0.221600   7.700 1.36e-14 ***
## Age11                   1.896958   0.221813   8.552  < 2e-16 ***
## Age12                   2.020597   0.222227   9.092  < 2e-16 ***
## Age13                   2.122161   0.221860   9.565  < 2e-16 ***
## Education2             -0.104886   0.307163  -0.341 0.732753
## Education3              0.141643   0.303956   0.466 0.641218
## Education4              0.049691   0.302263   0.164 0.869418
## Education5              0.122109   0.302413   0.404 0.686373
```

```
## Education6             0.121315   0.302711   0.401 0.688596
## Income2               -0.060618   0.049648  -1.221 0.222097
## Income3               -0.120563   0.048811  -2.470 0.013510 *
## Income4               -0.231604   0.048902  -4.736 2.18e-06 ***
## Income5               -0.283646   0.048806  -5.812 6.18e-09 ***
## Income6               -0.420202   0.049152  -8.549  < 2e-16 ***
## Income7               -0.511670   0.050940 -10.045  < 2e-16 ***
## Income8               -0.712266   0.051091 -13.941  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86126  on 253679  degrees of freedom
## Residual deviance: 71104  on 253633  degrees of freedom
## AIC: 71198
##
## Number of Fisher Scoring iterations: 8
```

Full model using logistic glm was fit. The canonical link was logit.

```
df <- data.frame(Variables = c("Diabetes", "BMI", "Heart Disease or Attack", "High Blood Pressure", "Hig

knitr::kable(df, caption = "Table 1: Partial Data Summary of the Original Model")
```

Table 6: Table 1: Partial Data Summary of the Original Model

Variables	OddsRatio	PValue
Diabetes	1.195959	< 2e-16
BMI	0.981248	< 2e-16
Heart Disease or Attack	2.617841	< 2e-16
High Blood Pressure	1.642245	< 2e-16
High Cholesterol	1.224239	< 2e-16
Smoking	1.172413	1.16e-12
Age 70 to 74	6.665587	< 2e-16
Age 75 to 79	7.542827	< 2e-16
Age 80 or older	8.349160	< 2e-16

Variable Selection

Forward Stepwise selectin using AIC

```
## Forward Stepwise elimination based on AIC ##
sel.var.forward.aic <- step(glm(Stroke~1, family = binomial(link = logit), data = health_data), scope =
select_var_forward_aic<-attr(terms(sel.var.forward.aic), "term.labels")   #gives us what variables are
select_var_forward_aic
```

```
##  [1] "GenHlth"              "HeartDiseaseorAttack" "Age"
##  [4] "DiffWalk"             "HighBP"               "Income"
##  [7] "BMI"                  "HighChol"             "Smoker"
```

```
## [10] "Diabetes_012"         "Veggies"              "MentHlth"
## [13] "NoDocbcCost"          "PhysHlth"             "Education"
## [16] "CholCheck"            "Sex"                  "HvyAlcoholConsump"
```

```
logit.forward.aic <- glm(Stroke ~., data = health_data[,which(colnames(health_data) %in% c(select_var_f
summary(logit.forward.aic)
```

```
##
## Call:
## glm(formula = Stroke ~ ., family = binomial, data = health_data[,
##     which(colnames(health_data) %in% c(select_var_forward_aic,
##         "Stroke"))])
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -5.759175   0.387111 -14.877  < 2e-16 ***
## Diabetes_0121          -0.049196   0.067503  -0.729 0.466123
## Diabetes_0122           0.179025   0.025254   7.089 1.35e-12 ***
## HighBP1                 0.495849   0.025930  19.123  < 2e-16 ***
## HighChol1               0.201917   0.023179   8.711  < 2e-16 ***
## CholCheck1              0.296805   0.084562   3.510 0.000448 ***
## BMI                    -0.019033   0.001739 -10.942  < 2e-16 ***
## Smoker1                 0.157382   0.022342   7.044 1.86e-12 ***
## HeartDiseaseorAttack1   0.962702   0.024277  39.655  < 2e-16 ***
## Veggies1               -0.145179   0.024634  -5.893 3.78e-09 ***
## HvyAlcoholConsump1     -0.111279   0.055183  -2.017 0.043743 *
## NoDocbcCost1            0.180929   0.034033   5.316 1.06e-07 ***
## GenHlth2                0.358778   0.054418   6.593 4.31e-11 ***
## GenHlth3                0.767606   0.052971  14.491  < 2e-16 ***
## GenHlth4                1.077006   0.056218  19.158  < 2e-16 ***
## GenHlth5                1.288475   0.063205  20.386  < 2e-16 ***
## MentHlth1               0.156292   0.033807   4.623 3.78e-06 ***
## PhysHlth1               0.115905   0.030556   3.793 0.000149 ***
## DiffWalk1               0.537294   0.026029  20.642  < 2e-16 ***
## Sex1                    0.060060   0.022395   2.682 0.007321 **
## Age2                    0.010890   0.287586   0.038 0.969795
## Age3                    0.639159   0.245494   2.604 0.009226 **
## Age4                    0.829389   0.235748   3.518 0.000435 ***
## Age5                    1.076420   0.229500   4.690 2.73e-06 ***
## Age6                    1.150129   0.225977   5.090 3.59e-07 ***
## Age7                    1.355193   0.223020   6.077 1.23e-09 ***
## Age8                    1.488125   0.222042   6.702 2.06e-11 ***
## Age9                    1.581229   0.221627   7.135 9.70e-13 ***
## Age10                   1.713227   0.221549   7.733 1.05e-14 ***
## Age11                   1.904543   0.221748   8.589  < 2e-16 ***
## Age12                   2.029172   0.222148   9.134  < 2e-16 ***
## Age13                   2.132251   0.221745   9.616  < 2e-16 ***
## Education2             -0.103923   0.306923  -0.339 0.734913
## Education3              0.143687   0.303701   0.473 0.636129
## Education4              0.053569   0.302001   0.177 0.859209
## Education5              0.127974   0.302146   0.424 0.671893
## Education6              0.129309   0.302437   0.428 0.668975
## Income2                -0.058588   0.049619  -1.181 0.237696
## Income3                -0.118665   0.048785  -2.432 0.015000 *
```

```
## Income4               -0.229352   0.048866  -4.694 2.69e-06 ***
## Income5               -0.281060   0.048766  -5.763 8.24e-09 ***
## Income6               -0.416668   0.049095  -8.487  < 2e-16 ***
## Income7               -0.506925   0.050853  -9.968  < 2e-16 ***
## Income8               -0.706092   0.050953 -13.858  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86126  on 253679  degrees of freedom
## Residual deviance: 71108  on 253636  degrees of freedom
## AIC: 71196
##
## Number of Fisher Scoring iterations: 8
```

The forward stepwise for AIC is done.

Diagnostics and Model Validation for forward selection based on AIC

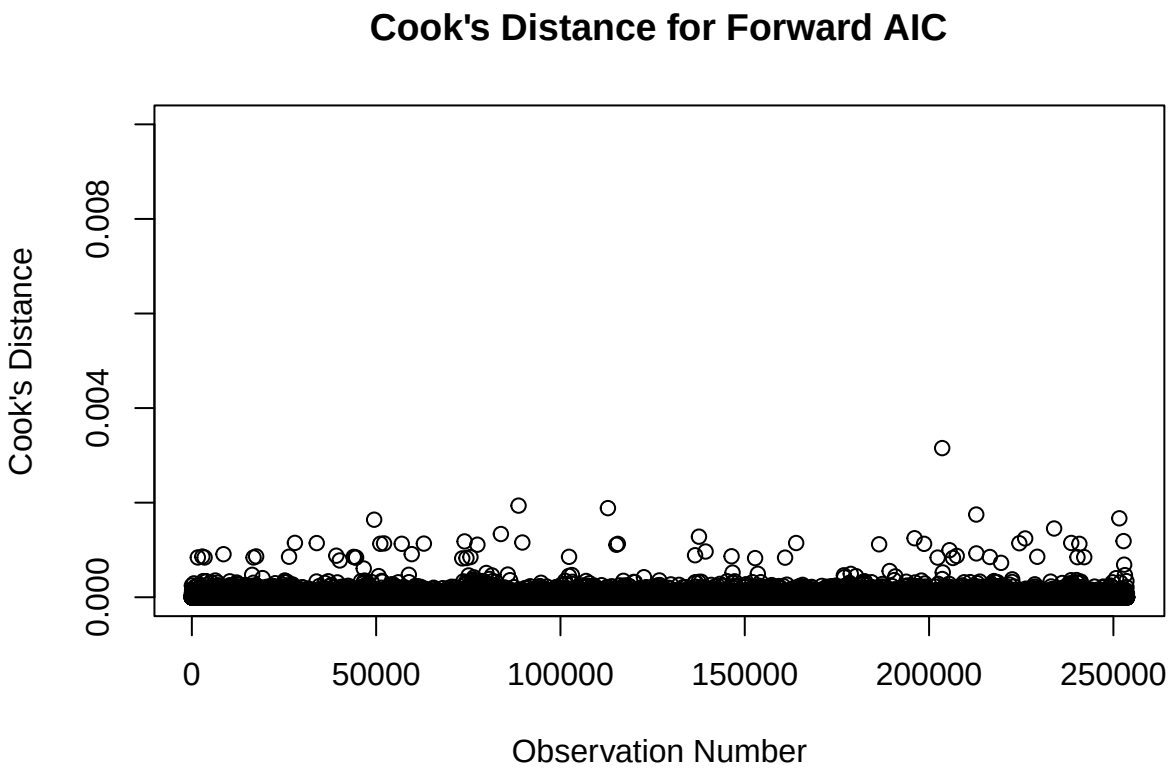
```
#influential points that affect the estimation of all fitted values (Cook's Distance) for AIC

di.aic <-cooks.distance(logit.forward.aic)
length(which(di.aic > qf(0.5, 44, 253680-43-1)))  #qf(0.5, p+1, n-p-1), where n =253680, p = 43
```

```
## [1] 0
```

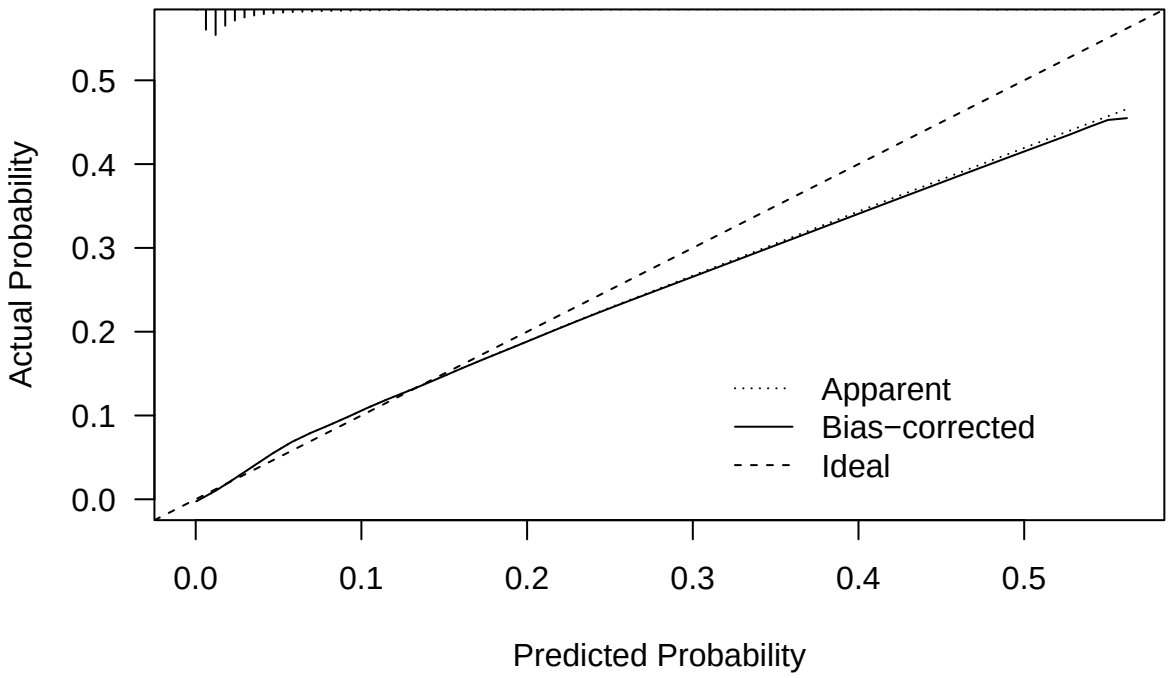
```
qf(0.5, 44, 253680-43-1)
```

```
## [1] 0.9848926
```

```
plot(di.aic, ylim = c(0,0.01), main = "Cook's Distance for Forward AIC", xlab = "Observation Number", yl
```




```
## Calibration plot for BIC with lrm from rms package ##
## Fit the model with lrm from rms package ##
lrm.forward.aic <- lrm(Stroke ~ ., data = health_data[,which(colnames(health_data) %in% c(select_var_fo
cross.calib.forward.aic <- calibrate(lrm.forward.aic, method="crossvalidation", B=10) # model calibrati
plot(cross.calib.forward.aic, las=1, xlab = "Predicted Probability")
```



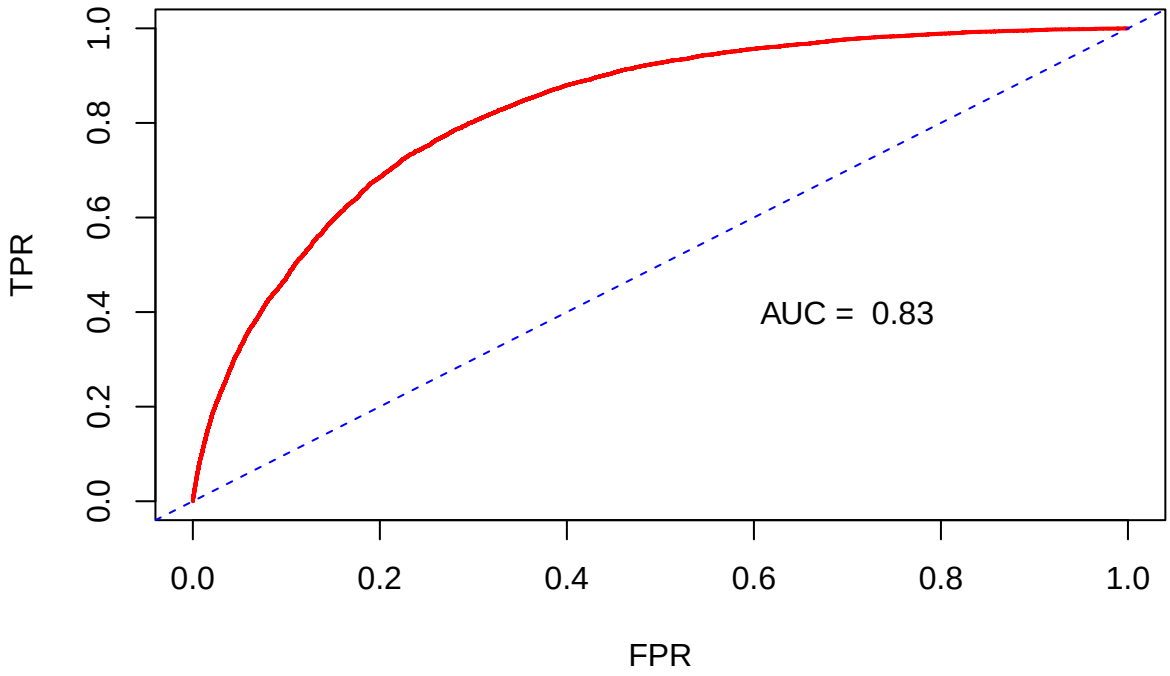

```
##
## n=253680   Mean absolute error=0.004   Mean squared error=6e-05
## 0.9 Quantile of absolute error=0.009
```

```
### Discrimination with ROC curve for forward selection for BIC
p <- predict(lrm.forward.aic, type = "fitted")

roc_logit <- roc(health_data$Stroke ~ p)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
## The True Positive Rate ##
TPR <- roc_logit$sensitivities
## The False Positive Rate ##
FPR <- 1 - roc_logit$specificities

plot(FPR, TPR, xlim = c(0,1), ylim = c(0,1), type = 'l', lty = 1, lwd = 2,col = 'red')
abline(a = 0, b = 1, lty = 2, col = 'blue')
text(0.7,0.4,label = paste("AUC = ", round(auc(roc_logit),2)))
```



Forward Stepwise selectin using BIC

```
## Forward Stepwise elimination based on BIC ##
sel.var.forward.bic <- step(glm(Stroke~1, family = binomial(link = logit), data = health_data), scope =
select_var_forward_bic<-attr(terms(sel.var.forward.bic), "term.labels")   #gives us what variables are
select_var_forward_bic
```

```
##  [1] "GenHlth"              "HeartDiseaseorAttack" "Age"
##  [4] "DiffWalk"             "HighBP"               "Income"
##  [7] "BMI"                  "HighChol"             "Smoker"
## [10] "Diabetes_012"         "Veggies"              "MentHlth"
## [13] "NoDocbcCost"          "PhysHlth"             "CholCheck"
```

```
logit.forward.bic <- glm(Stroke ~., data = health_data[,which(colnames(health_data) %in% c(select_var_f
summary(logit.forward.bic)
```

```
##
## Call:
## glm(formula = Stroke ~ ., family = binomial, data = health_data[,
##     which(colnames(health_data) %in% c(select_var_forward_bic,
##         "Stroke"))])
##
## Coefficients:
```

```
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -5.661898   0.244105 -23.195  < 2e-16 ***
## Diabetes_0121          -0.049899   0.067497  -0.739 0.459734
## Diabetes_0122           0.182623   0.025198   7.248 4.24e-13 ***
## HighBP1                 0.495542   0.025900  19.133  < 2e-16 ***
## HighChol1               0.200867   0.023178   8.666  < 2e-16 ***
## CholCheck1              0.294127   0.084523   3.480 0.000502 ***
## BMI                    -0.018848   0.001735 -10.862  < 2e-16 ***
## Smoker1                 0.162184   0.021893   7.408 1.28e-13 ***
## HeartDiseaseorAttack1  0.973720   0.024051  40.486  < 2e-16 ***
## Veggies1               -0.143844   0.024460  -5.881 4.09e-09 ***
## NoDocbcCost1            0.179893   0.034009   5.290 1.23e-07 ***
## GenHlth2                0.357764   0.054398   6.577 4.81e-11 ***
## GenHlth3                0.764779   0.052903  14.456  < 2e-16 ***
## GenHlth4                1.070057   0.056085  19.079  < 2e-16 ***
## GenHlth5                1.280121   0.063013  20.315  < 2e-16 ***
## MentHlth1               0.152828   0.033752   4.528 5.95e-06 ***
## PhysHlth1               0.120379   0.030510   3.946 7.96e-05 ***
## DiffWalk1               0.533911   0.025920  20.598  < 2e-16 ***
## Age2                    0.013937   0.287523   0.048 0.961340
## Age3                    0.635960   0.245429   2.591 0.009564 **
## Age4                    0.825654   0.235667   3.503 0.000459 ***
## Age5                    1.071122   0.229426   4.669 3.03e-06 ***
## Age6                    1.142665   0.225912   5.058 4.24e-07 ***
## Age7                    1.345892   0.222955   6.037 1.57e-09 ***
## Age8                    1.479249   0.221976   6.664 2.66e-11 ***
## Age9                    1.574133   0.221555   7.105 1.20e-12 ***
## Age10                   1.705104   0.221454   7.700 1.37e-14 ***
## Age11                   1.893352   0.221636   8.543  < 2e-16 ***
## Age12                   2.014463   0.222026   9.073  < 2e-16 ***
## Age13                   2.117967   0.221611   9.557  < 2e-16 ***
## Income2                -0.050304   0.049511  -1.016 0.309625
## Income3                -0.105653   0.048536  -2.177 0.029497 *
## Income4                -0.211289   0.048381  -4.367 1.26e-05 ***
## Income5                -0.253937   0.047991  -5.291 1.21e-07 ***
## Income6                -0.381989   0.047917  -7.972 1.56e-15 ***
## Income7                -0.463712   0.049210  -9.423  < 2e-16 ***
## Income8                -0.653365   0.048311 -13.524  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86126  on 253679  degrees of freedom
## Residual deviance: 71141  on 253643  degrees of freedom
## AIC: 71215
##
## Number of Fisher Scoring iterations: 8
```

Diagnostics and Model Validation for forward selection based on BIC

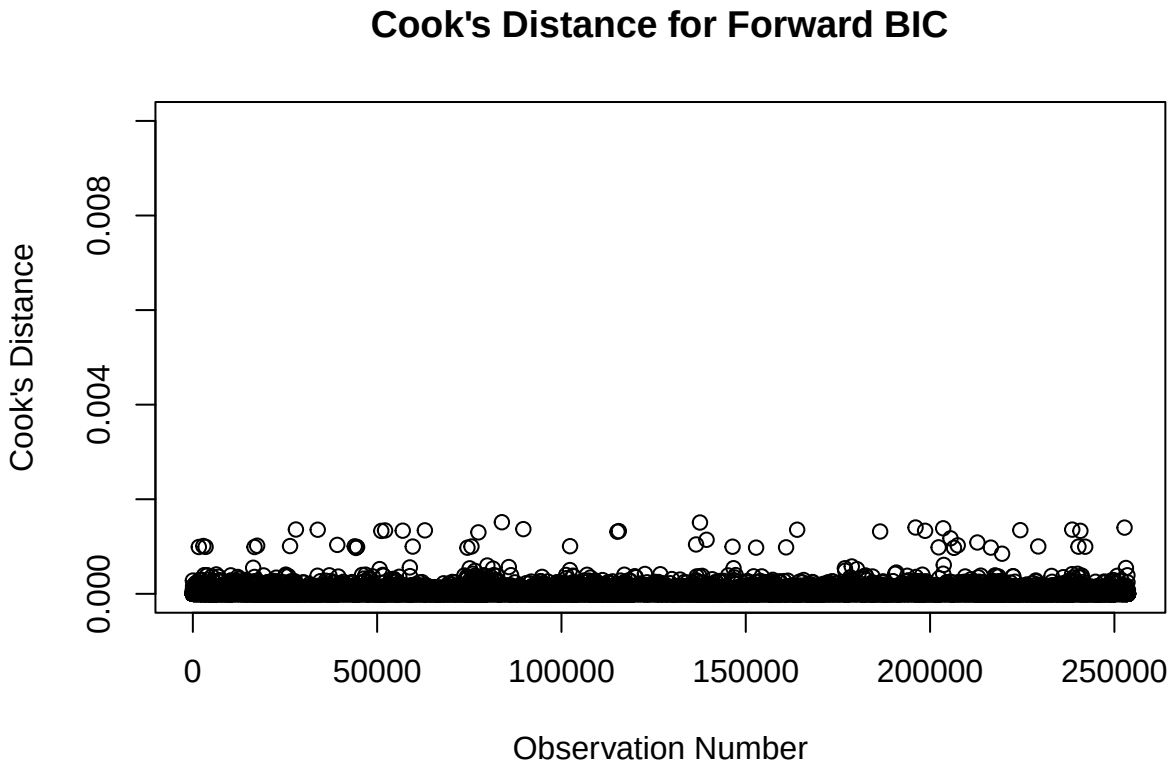
```r
#influential points that affect the estimation of all fitted values (Cook's Distance) for BIC

di.bic <-cooks.distance(logit.forward.bic)
length(which(di.bic > qf(0.5, 37, 253680-36-1)))  #qf(0.5, p+1, n-p-1), where n =253680, p = 36
```

```
## [1] 0
```

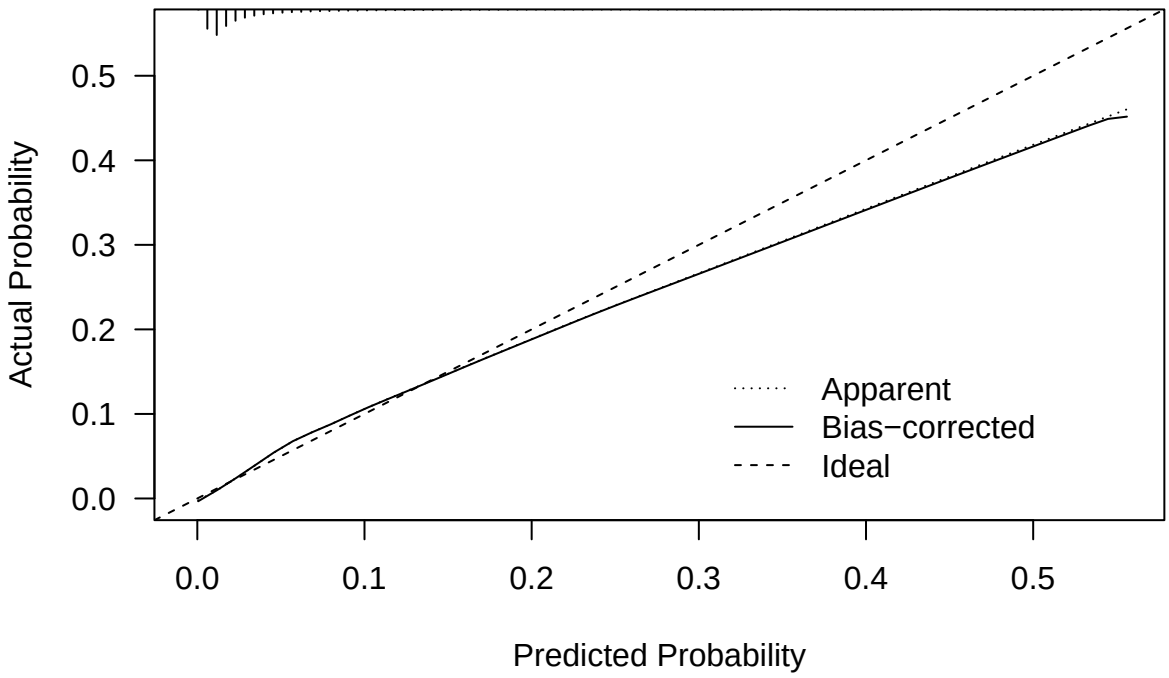
```r
qf(0.5, 47, 253680-46-1)
```

```
## [1] 0.9858545
```

```r
plot(di.bic, ylim = c(0,0.01), main = "Cook's Distance for Forward BIC", xlab = "Observation Number", yl
```




```r
## Calibration plot for BIC with lrm from rms package ##
## Fit the model with lrm from rms package ##
lrm.forward.bic <- lrm(Stroke ~ ., data = health_data[,which(colnames(health_data) %in% c(select_var_fo
cross.calib.forward.bic <- calibrate(lrm.forward.bic, method="crossvalidation", B=10) # model calibrati
plot(cross.calib.forward.bic, las=1, xlab = "Predicted Probability")
```



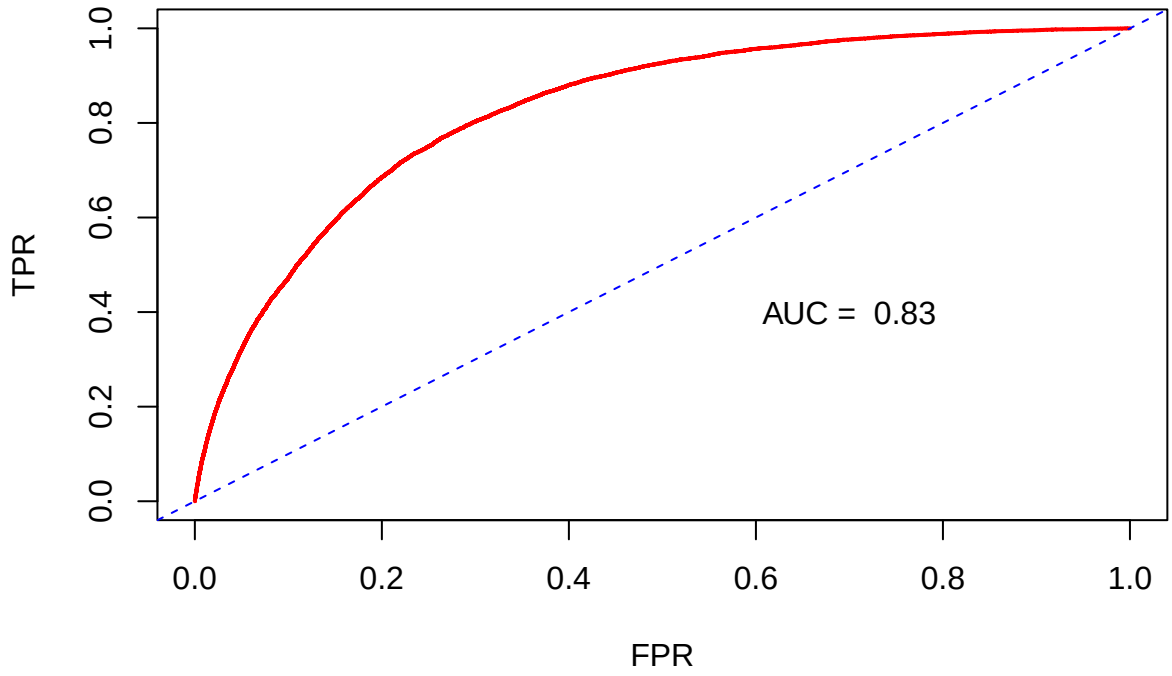

```
##
## n=253680   Mean absolute error=0.005   Mean squared error=6e-05
## 0.9 Quantile of absolute error=0.009
```

```
### Discrimination with ROC curve for forward selection for BIC
p <- predict(lrm.forward.bic, type = "fitted")

roc_logit <- roc(health_data$Stroke ~ p)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

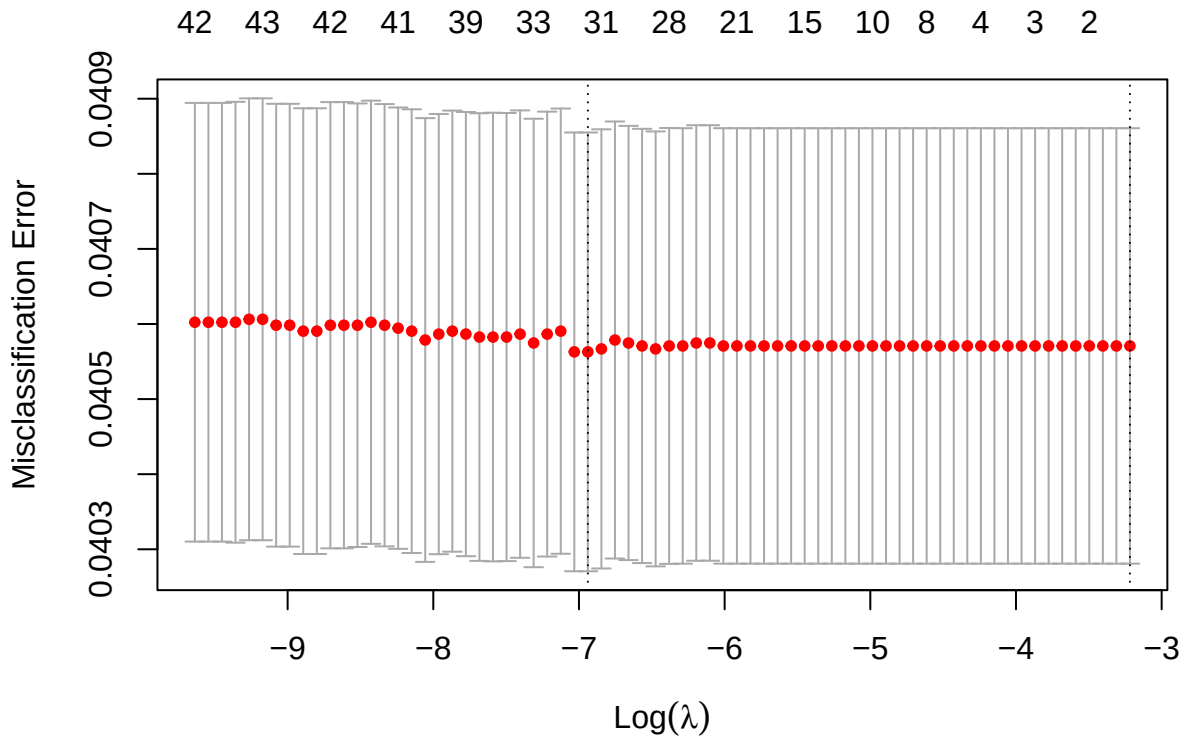
```
## The True Positive Rate ##
TPR <- roc_logit$sensitivities
## The False Positive Rate ##
FPR <- 1 - roc_logit$specificities

plot(FPR, TPR, xlim = c(0,1), ylim = c(0,1), type = 'l', lty = 1, lwd = 2,col = 'red')
abline(a = 0, b = 1, lty = 2, col = 'blue')
text(0.7,0.4,label = paste("AUC = ", round(auc(roc_logit),2)))
```



ROC curve for forward Stepwise on BIC

Variable selection based on LASSO

```
#LASSO variable selection
fmla =formula(Stroke ~.-Stroke)
x <- model.matrix(fmla, data=health_data)
y <- health_data$Stroke

cv.out = cv.glmnet(x, y, family = "binomial", type.measure = "class", nfolds = 25, alpha = 1)

plot(cv.out)
```




```
best.lambda <- cv.out$lambda.min
best.lambda
```

```
## [1] 0.0009692857
```

```
co<-coef(cv.out, s = "lambda.min")

#Selection of the significant features(predictors)

## threshold for variable selection ##

thresh <- 0.00 #what should the threshold be?
# select variables #
inds<-which(abs(co) > thresh)
variables<-row.names(co)[inds]
sel_var_lasso<-variables[!(variables %in% '(Intercept)')]
sel_var_lasso
```

```
##  [1] "Diabetes_0122"         "HighBP1"               "HighChol1"
##  [4] "CholCheck1"            "BMI"                   "Smoker1"
##  [7] "HeartDiseaseorAttack1" "Veggies1"              "NoDocbcCost1"
## [10] "GenHlth3"              "GenHlth4"              "GenHlth5"
## [13] "MentHlth1"             "PhysHlth1"             "DiffWalk1"
## [16] "Age2"                  "Age3"                  "Age4"
## [19] "Age5"                  "Age6"                  "Age9"
```

```
## [22] "Age10"                "Age11"                "Age12"
## [25] "Age13"                "Education3"           "Income2"
## [28] "Income3"              "Income6"              "Income7"
## [31] "Income8"
```

Lasso selection is ran. It finds the best lambda and compares to the threshold to select the appropriate variables based on coefficients.

```
logit.lasso <- glm(Stroke ~ Diabetes_012 + HighBP + HighChol + BMI + Smoker + HeartDiseaseorAttack + Ve

summary(logit.lasso)
```

```
##
## Call:
## glm(formula = Stroke ~ Diabetes_012 + HighBP + HighChol + BMI +
##     Smoker + HeartDiseaseorAttack + Veggies + NoDocbcCost + GenHlth +
##     MentHlth + PhysHlth + DiffWalk + Age + Income, family = "binomial",
##     data = health_data)
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -5.386757   0.230717 -23.348  < 2e-16 ***
## Diabetes_0121          -0.047660   0.067504  -0.706 0.480173
## Diabetes_0122           0.185253   0.025196   7.352 1.95e-13 ***
## HighBP1                 0.499905   0.025887  19.311  < 2e-16 ***
## HighChol1               0.203833   0.023172   8.797  < 2e-16 ***
## BMI                    -0.018739   0.001734 -10.804  < 2e-16 ***
## Smoker1                 0.160779   0.021890   7.345 2.06e-13 ***
## HeartDiseaseorAttack1   0.974971   0.024053  40.535  < 2e-16 ***
## Veggies1               -0.143310   0.024459  -5.859 4.65e-09 ***
## NoDocbcCost1            0.173129   0.033960   5.098 3.43e-07 ***
## GenHlth2                0.359634   0.054394   6.612 3.80e-11 ***
## GenHlth3                0.767168   0.052898  14.503  < 2e-16 ***
## GenHlth4                1.073146   0.056080  19.136  < 2e-16 ***
## GenHlth5                1.283301   0.063010  20.367  < 2e-16 ***
## MentHlth1               0.152501   0.033748   4.519 6.22e-06 ***
## PhysHlth1               0.121276   0.030512   3.975 7.05e-05 ***
## DiffWalk1               0.534772   0.025920  20.631  < 2e-16 ***
## Age2                    0.010351   0.287516   0.036 0.971282
## Age3                    0.628740   0.245420   2.562 0.010410 *
## Age4                    0.819786   0.235655   3.479 0.000504 ***
## Age5                    1.066733   0.229418   4.650 3.32e-06 ***
## Age6                    1.139498   0.225903   5.044 4.55e-07 ***
## Age7                    1.344096   0.222948   6.029 1.65e-09 ***
## Age8                    1.477863   0.221968   6.658 2.78e-11 ***
## Age9                    1.573799   0.221549   7.104 1.22e-12 ***
## Age10                   1.706850   0.221449   7.708 1.28e-14 ***
## Age11                   1.895745   0.221631   8.554  < 2e-16 ***
## Age12                   2.017139   0.222022   9.085  < 2e-16 ***
## Age13                   2.120733   0.221608   9.570  < 2e-16 ***
## Income2                -0.049766   0.049507  -1.005 0.314786
## Income3                -0.105042   0.048531  -2.164 0.030430 *
## Income4                -0.210578   0.048376  -4.353 1.34e-05 ***
```

```
## Income5               -0.252671   0.047986  -5.266 1.40e-07 ***
## Income6               -0.380204   0.047910  -7.936 2.09e-15 ***
## Income7               -0.461391   0.049201  -9.378  < 2e-16 ***
## Income8               -0.649099   0.048291 -13.442  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86126  on 253679  degrees of freedom
## Residual deviance: 71154  on 253644  degrees of freedom
## AIC: 71226
##
## Number of Fisher Scoring iterations: 8
```

Method:

For Lasso selection method, the best lambda that minimizes the missclassification error and maximizing AUC is simulated. Using the best lambda, the coefficients that are satisfied are selected. If the coefficient meets the threshold of 0, the variable associated with the coefficient is selected as a predictor.

Result:

Using 25 folds and using minimum lambda as the best lambda in place of lambda with 1 standard. The best lambda that minimizes the missclassification error and maximizing auc was found. It was compared to the threshold of 0. The predictors Diabetes, High Blood Pressure, High Cholesterol, BMI, Smoking Status, Heart Disease/Attack, Veggies,.. were selected as signficant predictors.

Diagnostics and Model Validation

```
#influential points that affect the estimation of all fitted values (Cook's Distance) for LASSO

di.lasso <-cooks.distance(logit.lasso)
length(which(di.lasso > qf(0.5, 36, 253680-35-1)))  #qf(0.5, p+1, n-p-1), where n =253680, p = 35
```

```
## [1] 0
```

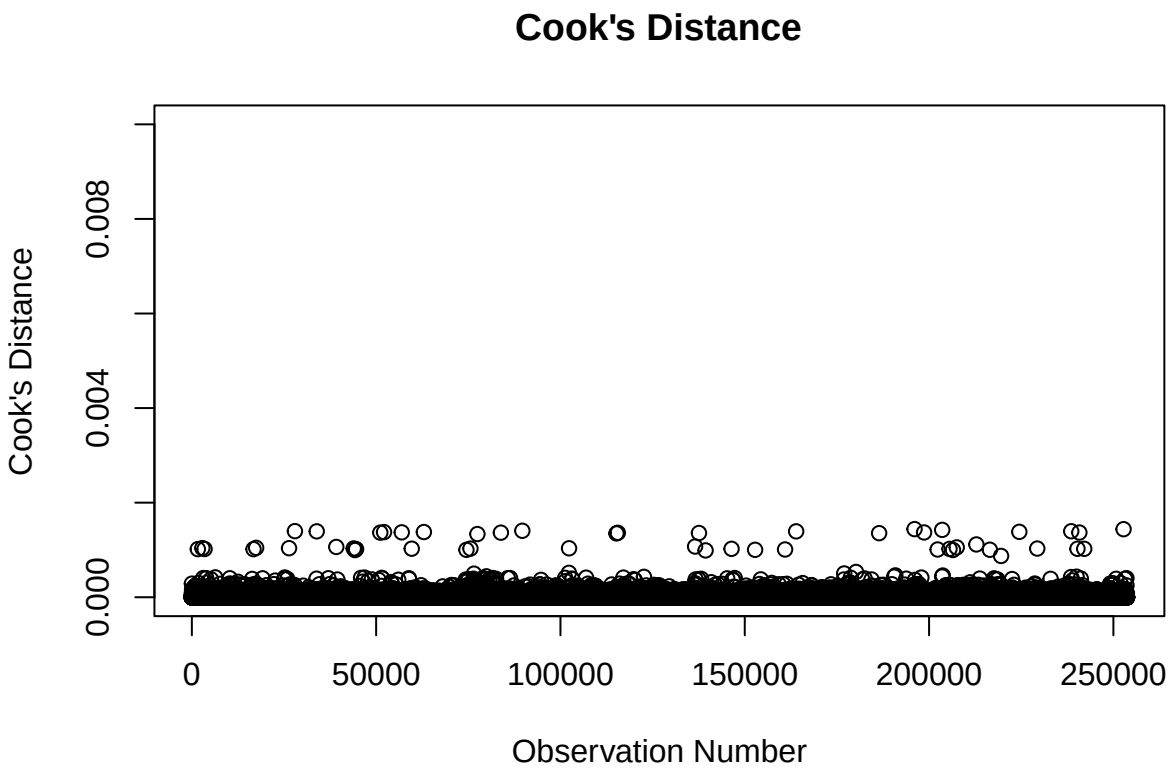
Running cook's distance

```
qf(0.5, 36, 253680-35-1)
```

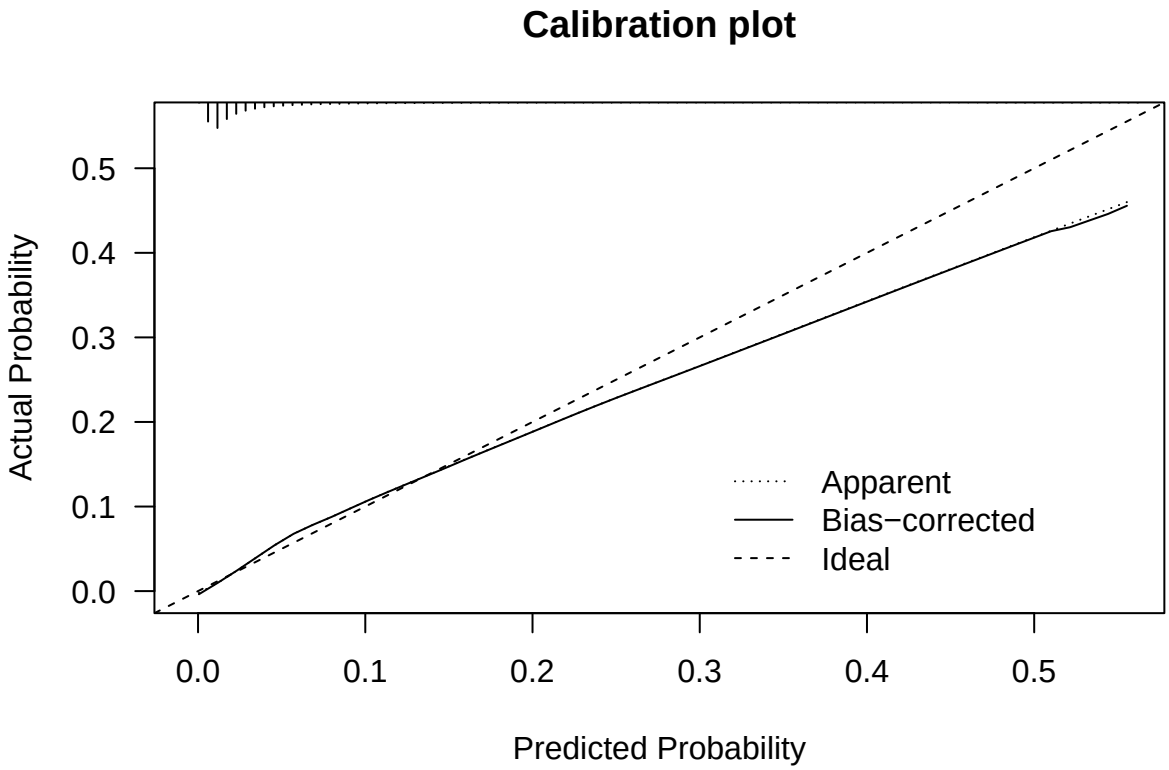
```
## [1] 0.9815463
```

The cooks distance cutoff

```
plot(di.lasso, ylim = c(0,0.01), main = "Cook's Distance", xlab = "Observation Number", ylab = "Cook's
```




```
## Calibration plot by LASSO with lrm from rms package ##
lrm.lasso <- lrm(Stroke ~ Diabetes_012 + HighBP + HighChol + BMI + Smoker + HeartDiseaseorAttack + Vegg
cross.calib.lasso <- calibrate(lrm.lasso, method="crossvalidation", B=10) # model calibration B means k
plot(cross.calib.lasso, las=1, xlab = "Predicted Probability", main = "Calibration plot")
```



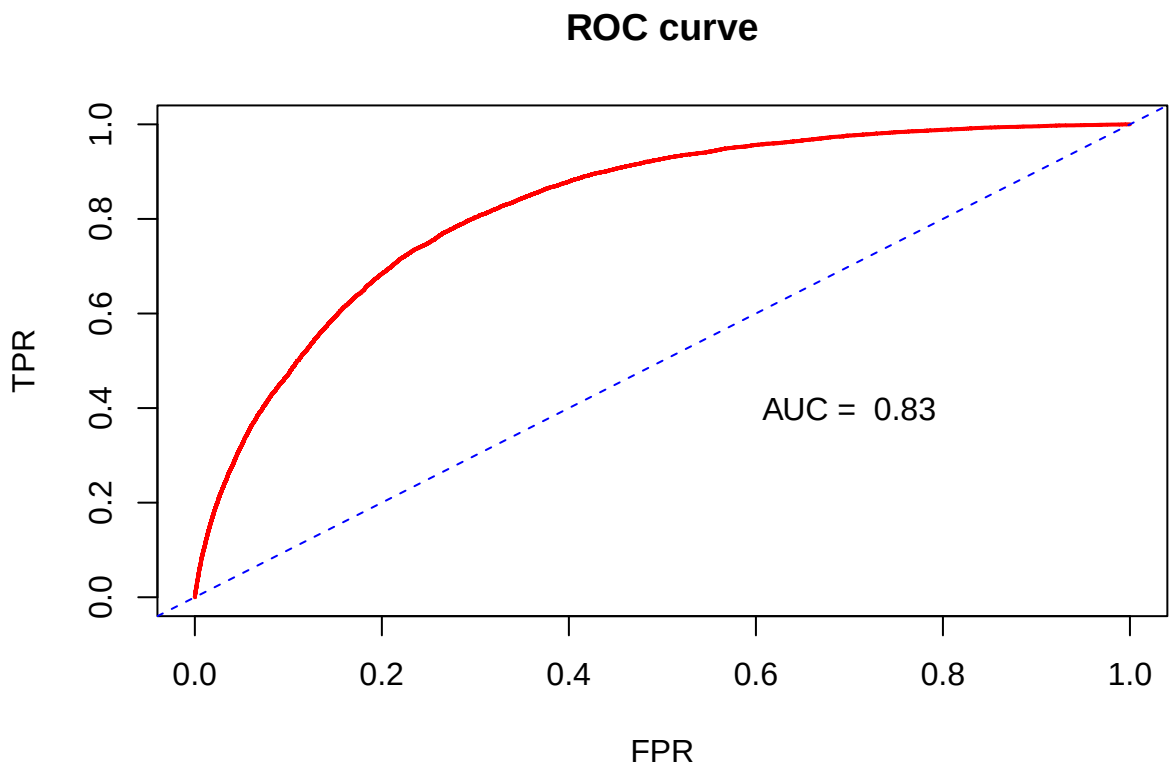

```
##
## n=253680   Mean absolute error=0.005   Mean squared error=6e-05
## 0.9 Quantile of absolute error=0.009
```

```
### Discrimination with ROC curve for LASSO
p <- predict(lrm.lasso, type = "fitted")

roc_logit <- roc(health_data$Stroke ~ p)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
## The True Positive Rate ##
TPR <- roc_logit$sensitivities
## The False Positive Rate ##
FPR <- 1 - roc_logit$specificities

plot(FPR, TPR, xlim = c(0,1), ylim = c(0,1), type = 'l', lty = 1, lwd = 2,col = 'red', main = "ROC curve
abline(a = 0, b = 1, lty = 2, col = 'blue')
text(0.7,0.4,label = paste("AUC = ", round(auc(roc_logit),2)))
```



Since all observations are much below 0.9815463, there are no influential observations that affect the estimation of all fitted values.

These are the model diagnostics and model validation plots for LASSO. It includes cooks distance plot, calibration plot, and ROC curve.

Final Model:

We choose Lasso model as our final model because AUC, mean absolute error and mean squared error are same between the lasso model and the forward selection bic model, and aic model, but LASSO model is a simpler model so it is better.

numerical summaries of final model

```
df <- data.frame(Variables = c("Diabetes", "BMI", "Heart Disease or Attack", "High Blood Pressure", "Hi

knitr::kable(df, caption = "Table 2: Partial Data Summary of the Final Model")
```

Table 7: Table 2: Partial Data Summary of the Final Model

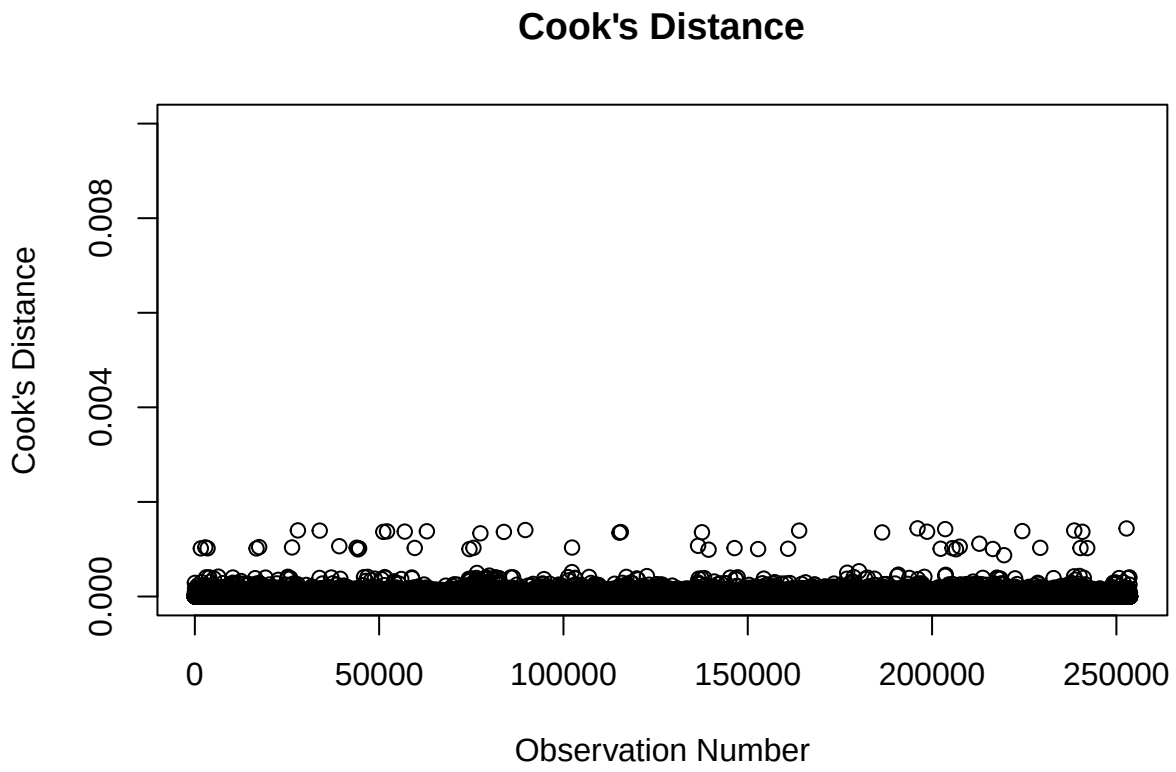
Variables	OddsRatio	PValue
Diabetes	1.3066079	< 2e-16
BMI	0.9831451	< 2e-16
Heart Disease or Attack	2.9247836	< 2e-16
High Blood Pressure	1.7635831	< 2e-16
High Cholesterol	1.2599925	< 2e-16
Smoking	1.2114213	< 2e-16

Variables	OddsRatio	PValue
Age 70 to 74	6.8699141	< 2e-16
Age 75 to 79	7.8076928	< 2e-16
Age 80 or older	8.7100367	< 2e-16

We find the odds ratio table for imporant measures of interests. We see that indeed our 5 variables of interests are significant.

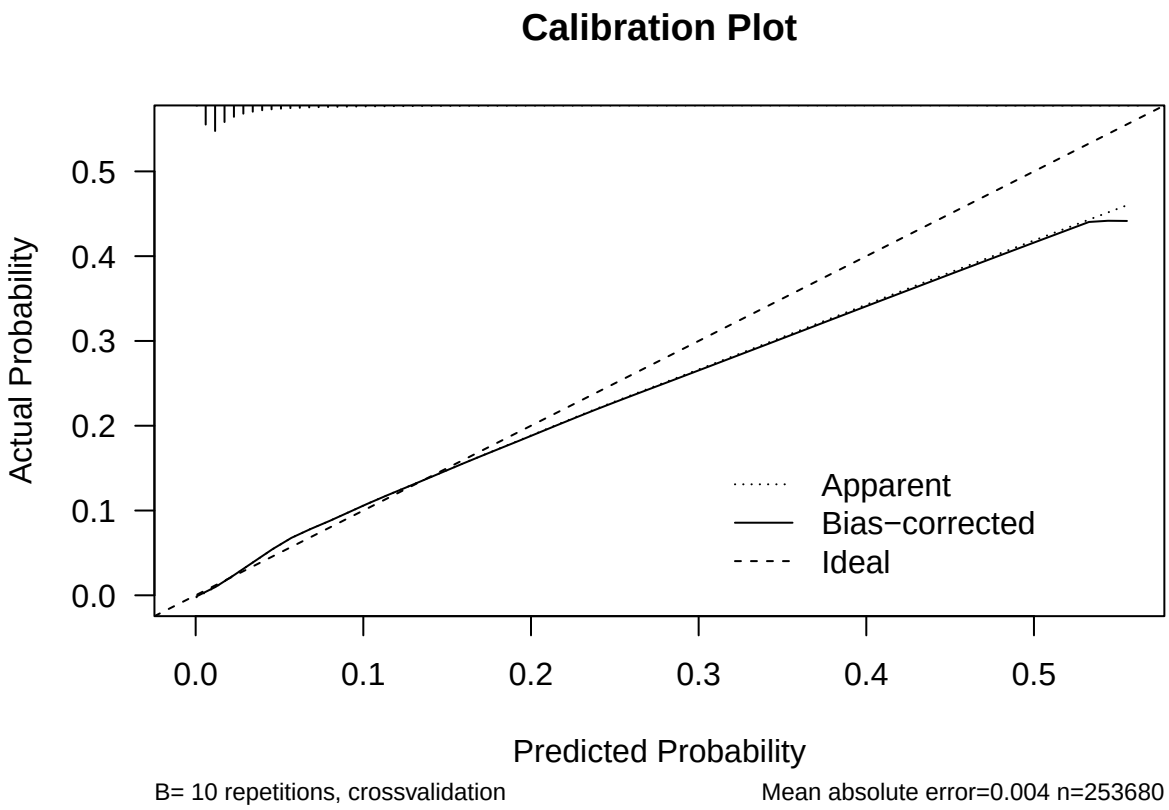
```
#cooks distance plot
plot(di.lasso, ylim = c(0,0.01), main = "Cook's Distance", xlab = "Observation Number", ylab = "Cook's D
mtext("Figure 3: Model Diagnostics and Validation Plots for the Final Model", side = 3, line = -1, outer
```

Figure 3: Model Diagnostics and Validation Plots for the Final Model



Cook's distance for the final model show that there are no influential observations

```
## Calibration plot by LASSO with lrm from rms package ##

lrm.lasso <- lrm(Stroke ~ Diabetes_012 + HighBP + HighChol + BMI + Smoker + HeartDiseaseorAttack + Veggi
cross.calib.lasso <- calibrate(lrm.lasso, method="crossvalidation", B=10) # model calibration B means k
plot(cross.calib.lasso, las=1, xlab = "Predicted Probability", main = "Calibration Plot")
```




```
##
## n=253680   Mean absolute error=0.004   Mean squared error=6e-05
## 0.9 Quantile of absolute error=0.009
```

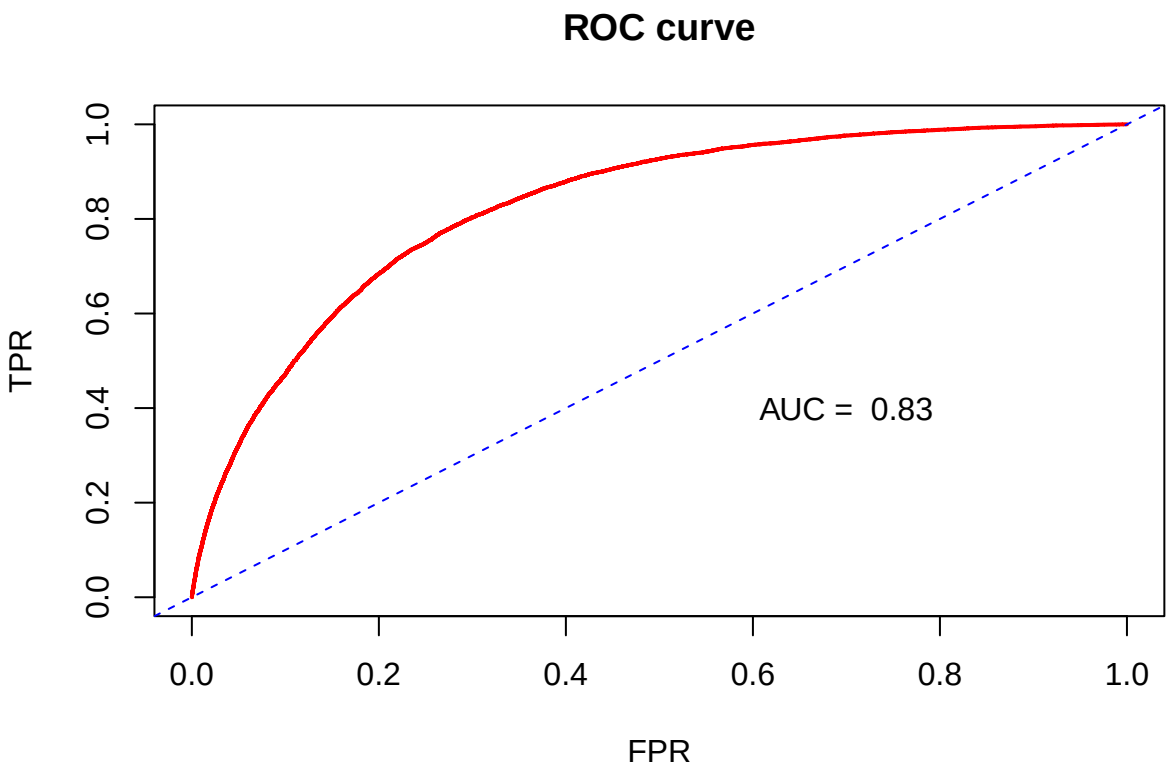
calibration plot looks very similar to the other forward selection bic/aic plots.

```
### Discrimination with ROC curve for LASSO
p <- predict(lrm.lasso, type = "fitted")

roc_logit <- roc(health_data$Stroke ~ p)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
## The True Positive Rate ##
TPR <- roc_logit$sensitivities
## The False Positive Rate ##
FPR <- 1 - roc_logit$specificities

plot(FPR, TPR, xlim = c(0,1), ylim = c(0,1), type = 'l', lty = 1, lwd = 2,col = 'red', main = "ROC curve
abline(a = 0, b = 1, lty = 2, col = 'blue')
text(0.7,0.4,label = paste("AUC = ", round(auc(roc_logit),2)))
```



The ROC curve looks very good as the AUC is 0.83

Discussion:

Limitations:

BIC/AIC -

When running different stepwise elimination, backward stepwise and stepwise elimination based on AIC or BIC iterates too many times due to an extremely large number of observations and does not finish. So, only forward stepwise for AIC and BIC were chosen for stewise regression.

Lasso -

Due to a large number of observations and unbalanced proportion of response variable, the missclassification error plot shows almost a straight line. The algorithm has a difficult time finding the best lambda with 1 standard error. This penalizes the dataset too heavily. To compensate, minimum lambda has been chosen to be the best lambda and number of folds was increased to 25.

The study has been done fully on the paper where we study the odds ratios of the final model and how the model is validated through model validation and model diagnostics.