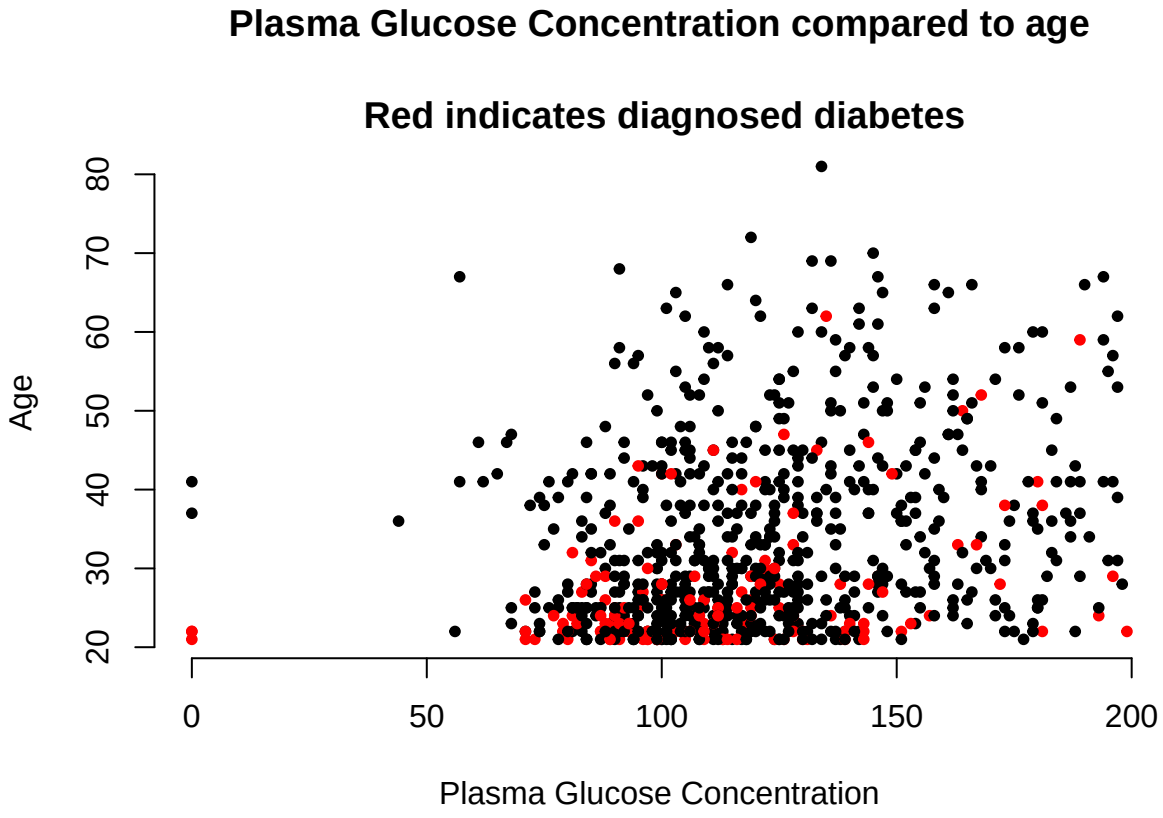
Assignment 3:

Step 1

Using the data from the csv file, a scatter plot is created representing the age and glucose level are plotted against each other. The red color of the plot indicates that the patient has been diagnosed with diabetes, while black means the patient is not diagnosed with diabetes.

The outcomes are binary and the variables doesn't seem to be correlated, which means that the data is suitable to use with logistic regression. However, the presence of diabetes seems to scattered across the entire plot, which would make it hard to predict the outcome based on the 2 selected variables.



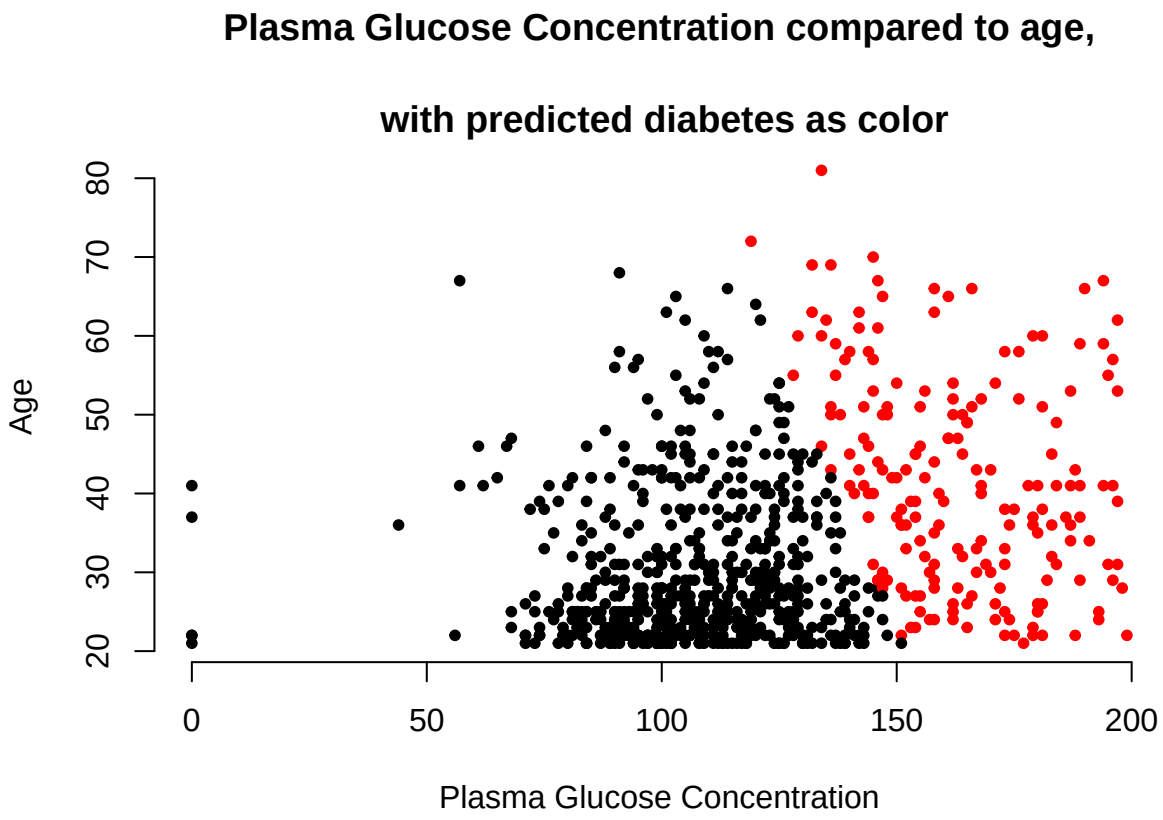
Step 2

Based on the outcome of the training of the logistic model seen below, the probabilistic model can be determined to be:

$$P(Y = 1) = \frac{1}{1 + e^{5.9124 - 0.0356 * X1 - 0.0248 * X2}}$$

The misclassification rate of 26,4% means that more than a quarter of the time, the predicted outcome is not correct. A high rate of misclassifications is understandable based on the fact that the presence of diabetes is spread out in the plot in step 1. Without any clear correlation between diagnosed diabetes and the 2 variables, It's hard to get a classification rate that is good. Therefore, the predicted values lead to a lot scatter plot that looks like the one below, containing a lot of misclassifications.

```
##
## Attaching package: 'dplyr'

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

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

##    pred
##       0   1
##   0 436  64
##   1 138 130

##
## Call:
## glm(formula = as.factor(V9) ~ ., family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3367  -0.7775  -0.5087   0.8367   3.1630
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.912449   0.462620  -12.78  < 2e-16 ***
## V2           0.035644   0.003290   10.83  < 2e-16 ***
## V8           0.024778   0.007374    3.36 0.000778 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 993.48  on 767  degrees of freedom
## Residual deviance: 797.36  on 765  degrees of freedom
## AIC: 803.36
##
## Number of Fisher Scoring iterations: 4

## Misclassification rate:  0.2630208
```



Step 3

In step 3, the decision boundary is calculated and inserted into the plot from step 2.

The decision boundary is decided with:

$$\beta_0 + \beta 1 * x_1 + \beta_2 * x_2 = 0$$

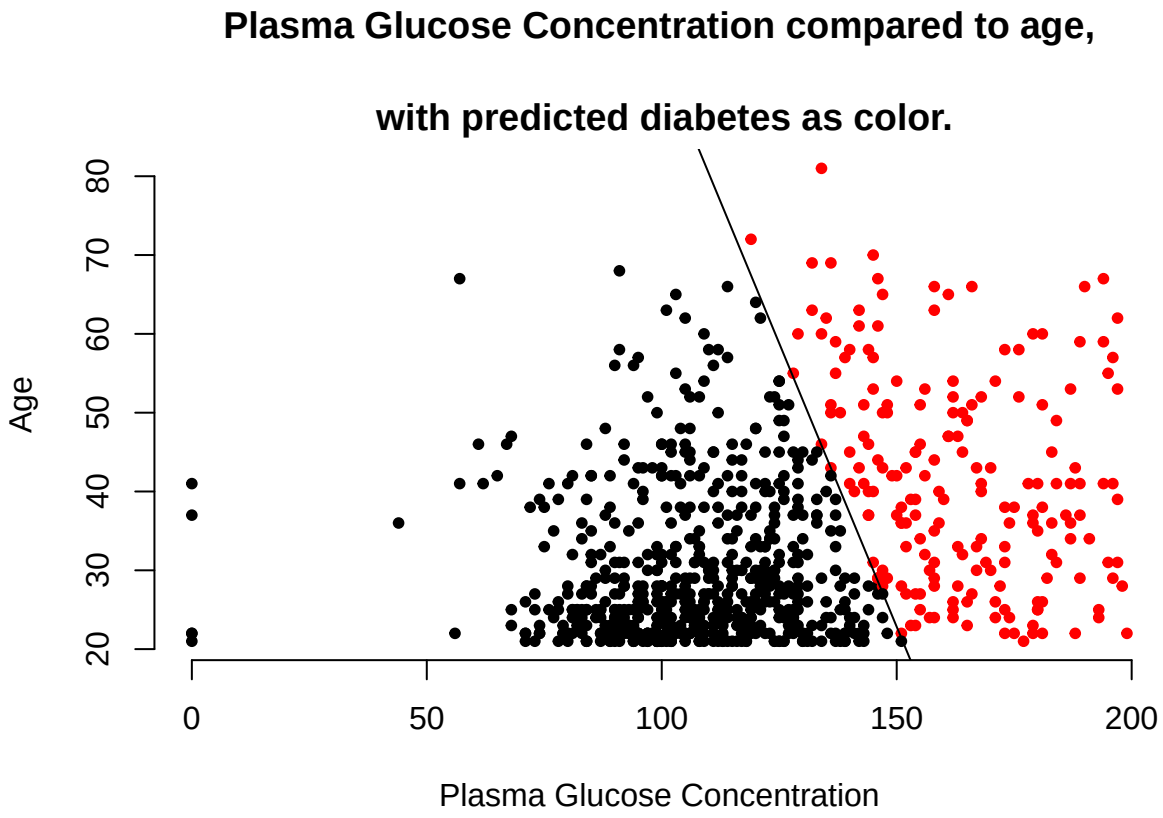
This means that the the line on the $y = kx + m$ format can be calculated with:

$$x_2 = -\frac{\beta_0}{\beta_2} - \frac{\beta_1}{\beta_2} x_1$$

Using the coefficient values from the glm function, we get the function:

$$x_2 = -\frac{-5.912}{0.025} - \frac{0.036}{0.025} x_1 \approx 239 - 1.44 x_1$$

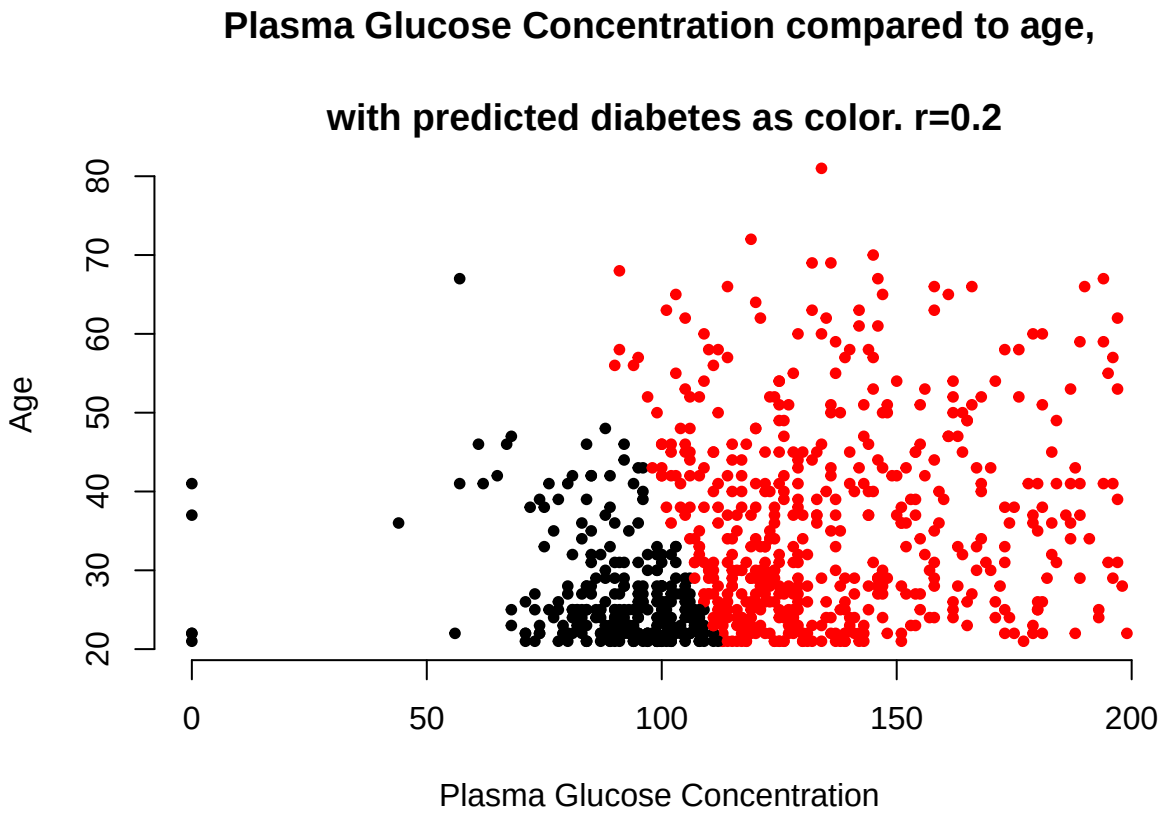
The decision boundary clearly separates the predicted cases of diabetes from the ones where diabetes is predicted to be false. However, when comparing to the actual outcomes, a lot of the observations to the left of the decision boundary are really cases of diabetes but classified as non-diabetes. This outcome is because of the linear decision boundary, which splits the data into 2 with a straight line. Since the actual cases of diabetes can't be split between true and false with a straight line, there will be misclassifications.

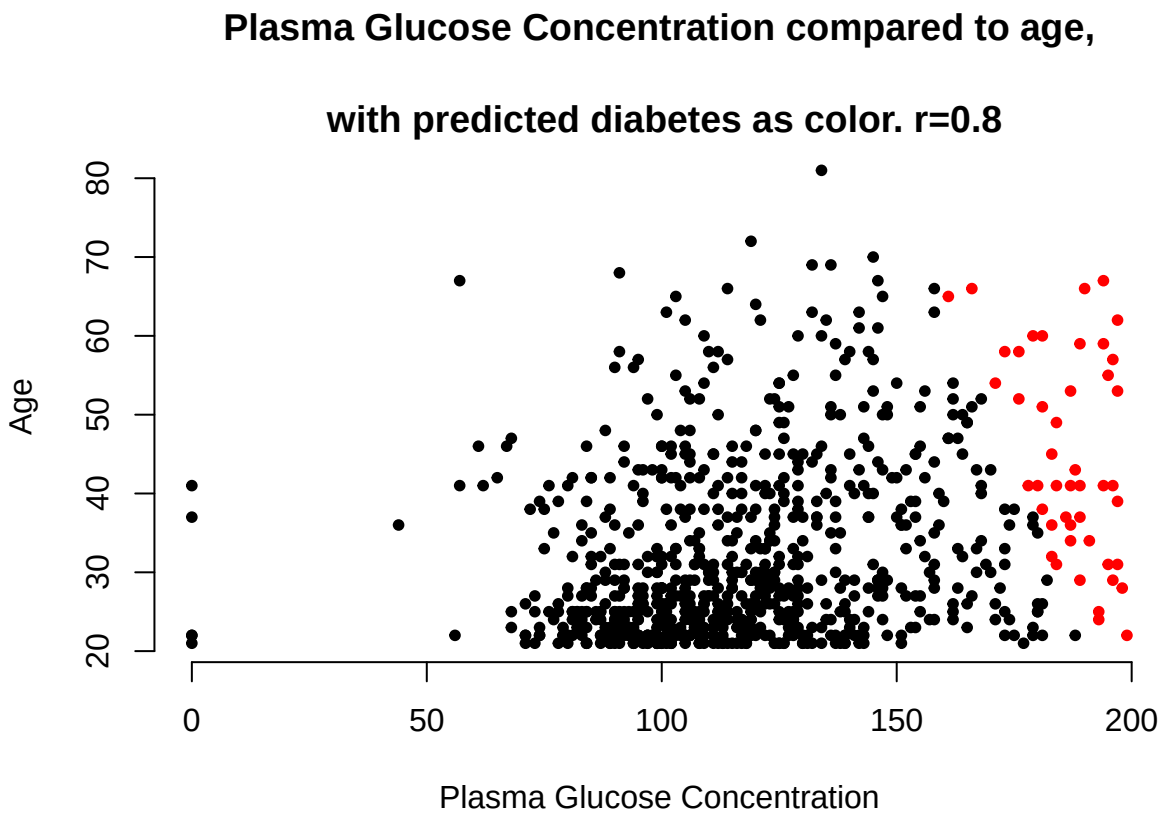


Step 4

Changing the threshold to 0.8 means that the probability of diabetes has to be higher in order for the model to classify it as true. This will lead to a smaller number of observations being classified as diabetes compared to r=0.5.

The opposite happens if changing the threshold to 0.2, where the probability now doesn't need to be as high in order for the model to classify it as diabetes. This leads to a higher number of observations being classified as diabetes compared to r=0.5.


Plasma Glucose Concentration compared to age,
with predicted diabetes as color. r=0.2
Age
20
30
40
50
60
70
80
0
50
100
150
200
Plasma Glucose Concentration



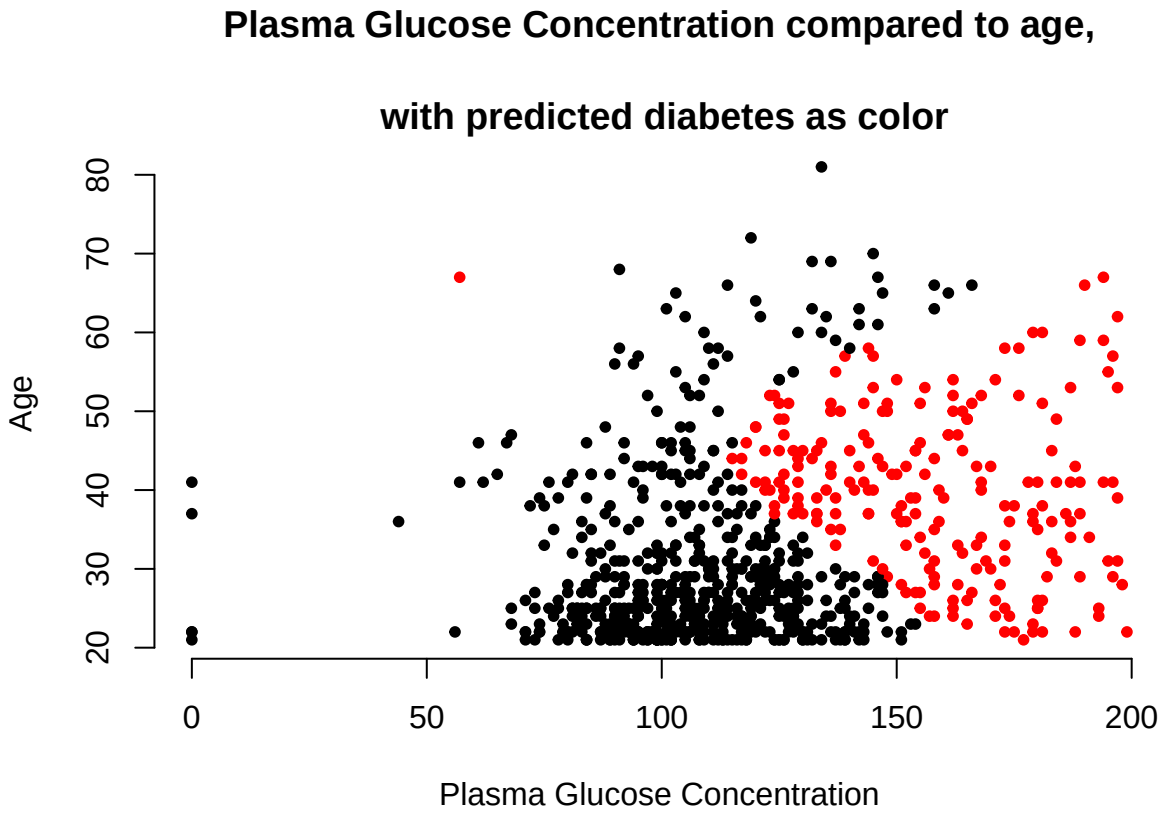
Step 5

The new variables introduced means that the decision boundary is no longer linear, since the polynomial degree is now 4. The misclassification rate has decreased a little bit which would indicate that the model is slightly better than before. This is also indicated by the AIC being lower than before.

```
##    pred
##       0   1
##   0 436  64
##   1 138 130
```

```
##
## Call:
## glm(formula = as.factor(V9) ~ ., family = "binomial", data = train2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2887  -0.7258  -0.4257   0.7451   2.5280
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.310e+00  1.129e+00  -8.243  < 2e-16 ***
## V2           3.793e-02  9.473e-03   4.004 6.23e-05 ***
## V8           1.457e-01  2.072e-02   7.031 2.05e-12 ***
## z1           1.278e-08  5.610e-09   2.278  0.02271 *
```

```
## z2          -1.780e-07  7.635e-08  -2.331  0.01976 *
## z3           8.515e-07  3.437e-07   2.478  0.01322 *
## z4          -1.698e-06  6.313e-07  -2.690  0.00715 **
## z5           8.127e-07  4.054e-07   2.004  0.04503 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 993.48  on 767  degrees of freedom
## Residual deviance: 741.61  on 760  degrees of freedom
## AIC: 757.61
##
## Number of Fisher Scoring iterations: 5

## Misclassification rate:  0.2447917
```




```
library(knitr)
<style>
body {
text-align: justify}
</style>
pima_indians_diabetes=read.csv("pima-indians-diabetes.csv", header=FALSE)

plot(pima_indians_diabetes$V2, pima_indians_diabetes$V8,
```

```
     main = "Plasma Glucose Concentration compared to age
     \n Red indicates diagnosed diabetes", xlab="Plasma Glucose Concentration",
     ylab="Age", pch=20, frame=FALSE,
     col=ifelse(pima_indians_diabetes == 1, "red", "black"))
library(dplyr)
library(tidyr)

train=tibble(pima_indians_diabetes)
train=train%>%select(V2,V8,V9)

r=0.5
m1=glm(as.factor(V9)~., train, family = "binomial")
prob=predict(m1, type = "response")
pred=ifelse(prob>r, 1, 0)

table(train$V9, pred)

summary(m1)
misclass=function(X,X1){
  n=length(X)
return(1-sum(diag(table(X,X1)))/n) }

mc=misclass(train$V9, pred)
cat("Misclassification rate: ", mc)

plot(pima_indians_diabetes$V2, pima_indians_diabetes$V8,
     main = "Plasma Glucose Concentration compared to age,
     \n with predicted diabetes as color", xlab="Plasma Glucose Concentration",
     ylab="Age", pch=20, frame=FALSE, col=ifelse(pred==1, "red", "black"))
slope=coef(m1)[2]/(-coef(m1)[3])
intercept=coef(m1)[1]/(-coef(m1)[3])

plot(pima_indians_diabetes$V2, pima_indians_diabetes$V8,
     main = "Plasma Glucose Concentration compared to age,
     \n with predicted diabetes as color.", xlab="Plasma Glucose Concentration",
     ylab="Age", pch=20, frame=FALSE, col=ifelse(pred==1, "red", "black"))
abline(intercept, slope)
r=0.2
pred2=ifelse(prob>r, 1, 0)

plot(pima_indians_diabetes$V2, pima_indians_diabetes$V8,
     main = "Plasma Glucose Concentration compared to age,
     \n with predicted diabetes as color. r=0.2",
     xlab="Plasma Glucose Concentration", ylab="Age", pch=20,
     frame=FALSE, col=ifelse(pred2==1, "red", "black"))

r=0.8
pred3=ifelse(prob>r, 1, 0)
plot(pima_indians_diabetes$V2, pima_indians_diabetes$V8,
     main = "Plasma Glucose Concentration compared to age,
     \n with predicted diabetes as color. r=0.8",
     xlab="Plasma Glucose Concentration", ylab="Age", pch=20,
     frame=FALSE, col=ifelse(pred3==1, "red", "black"))
```

```r
pima_indians_diabetes$z1=sapply(pima_indians_diabetes$V2, function(x) x^4)
pima_indians_diabetes$z2=sapply(pima_indians_diabetes$V2,
                                function(x) x^3) * pima_indians_diabetes$V8
pima_indians_diabetes$z3=sapply(pima_indians_diabetes$V2,
                                function(x) x^2) * sapply(pima_indians_diabetes$V8,
                                                          function(x) x^2)
pima_indians_diabetes$z4=pima_indians_diabetes$V2 * sapply(
  pima_indians_diabetes$V8, function(x) x^3)
pima_indians_diabetes$z5=sapply(pima_indians_diabetes$V8, function(x) x^4)

train2=tibble(pima_indians_diabetes)
train2=train2%>%select(V2,V8,V9,(z1:z5))

r=0.5
m2=glm(as.factor(V9)~., train2, family = "binomial")
prob4=predict(m2, type = "response")
pred4=ifelse(prob4>r, 1, 0)

table(train$V9, pred)

summary(m2)

mc2=misclass(train$V9, pred4)
cat("Misclassification rate: ", mc2)

plot(pima_indians_diabetes$V2, pima_indians_diabetes$V8,
     main = "Plasma Glucose Concentration compared to age,
     \n with predicted diabetes as color",
     xlab="Plasma Glucose Concentration", ylab="Age", pch=20,
     frame=FALSE, col=ifelse(pred4==1, "red", "black"))
```