Title: Gender Recognition by Voice and Speech Analysis

Group Number: 48

First Name	Last Name	Online Students? (Y or N)	Monday or Tuesday	Shared with ITMD 525? (Y or N)
Meghna	Patel	N	Monday	N
Prathamesh	Raje	N	Monday	N

Table of Contents

1. Introduction and Motivations

Humans are naturally capable of distinguishing between male and female voices. However, to achieve this using a computer application, we need to convert the voice into its acoustic properties which can then be analyzed to predict whether a given voice sample is of male or female. This is a little tricky process and it requires accurate analysis and predictions.

The motivation to take on this project comes from our interest towards speech analysis, machine learning, artificial Intelligence and increasing trend in speech analytics global market. A research from grandviewresearch.com suggests that the speech analytics global market is increasing every year globally, where it recorded a revenue of 572.3 million in 2014, 626.4 million in 2015 and is predicted to be gradually rise in the consecutive years. Moreover, with a thought of the project's future scope, this can be used to create an application that will take voice .wav files as input to analyze and identify the gender. The models and algorithms defined in our project can then be implemented to make such applications in future.

2. Data Description

The data set belongs to language processing and artificial intelligence domain. We collected the data from https://www.kaggle.com/primaryobjects/voicegender. The dataset used in the project is obtained from the voice samples which are preprocessed by acoustic analysis in R using the seewave and tuneR packages. The frequency range is narrowed to 0Hz to 280Hz (human vocal range) for analytics purpose. The dataset consists of 3168 records which consists equal number of male and female voice samples. There are 21 columns that includes 20 acoustic properties of each voice sample and one label column for classification of male and female. Features are as given below:

- **meanfreq**: mean frequency (in kHz)
- **sd**: standard deviation of frequency
- **median**: median frequency (in kHz)
- **Q25**: first quantile (in kHz)
- **Q75**: third quantile (in kHz)
- **IQR**: interquartile range (in kHz)
- **skew**: skewness
- **kurt**: kurtosis
- **sp.ent**: spectral entropy
- **sfm**: spectral flatness
- **mode**: mode frequency
- **centroid**: frequency centroid
- **meanfun**: average of fundamental frequency measured across acoustic signal
- **minfun**: minimum fundamental frequency measured across acoustic signal
- **maxfun**: maximum fundamental frequency measured across acoustic signal
- **meandom**: average of dominant frequency measured across acoustic signal
- **mindom**: minimum of dominant frequency measured across acoustic signal
- **maxdom**: maximum of dominant frequency measured across acoustic signal
- **dfrange**: range of dominant frequency measured across acoustic signal

- **modindx**: modulation index. Calculated as the accumulated absolute difference between adjacent measurements of fundamental frequencies divided by the frequency range
- **label**: male or female

3. Research Problems and Solutions

Below are the research problems that we want to solve and their solutions.

- Given a voice sample, predict whether it is a male voice or female voice.
  Solution: Since our dependent variable "label" is qualitative and binary, we will perform logistic regression analysis on the data set provided. We will build different logistic regression models using model selection techniques and evaluate them based on model evaluation.

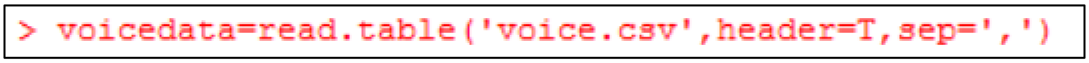
- Which factor is the most powerful indicator for gender recognition?
  Solution: We will use Classification and Regression Tree (CART) model to identify the most powerful indicator for gender recognition. The decision tree build from CART model will also be used to identify the gender of voice sample.

- Can we say that mean frequency of male voice is higher than mean frequency of female voice?
  Solution: We will create box plots of mean frequency for male and female labels and interpret them. We will analyze variance between them and perform hypothesis testing on two sample means.

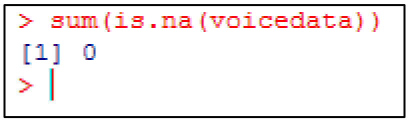
4. Model Learning

4.1. Data Processing

The dataset used is already preprocessed in R using the seewave and tuneR packages. The screenshots shown below shows the initial steps of loading the data and data examination steps:



```
> voicedata=read.table('voice.csv',header=T,sep=',')
```

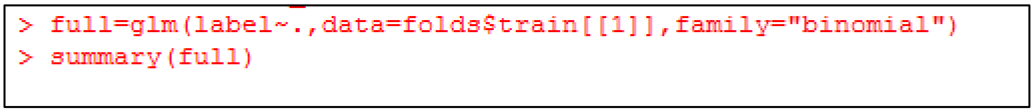


```
> sum(is.na(voicedata))
[1] 0
> 
```

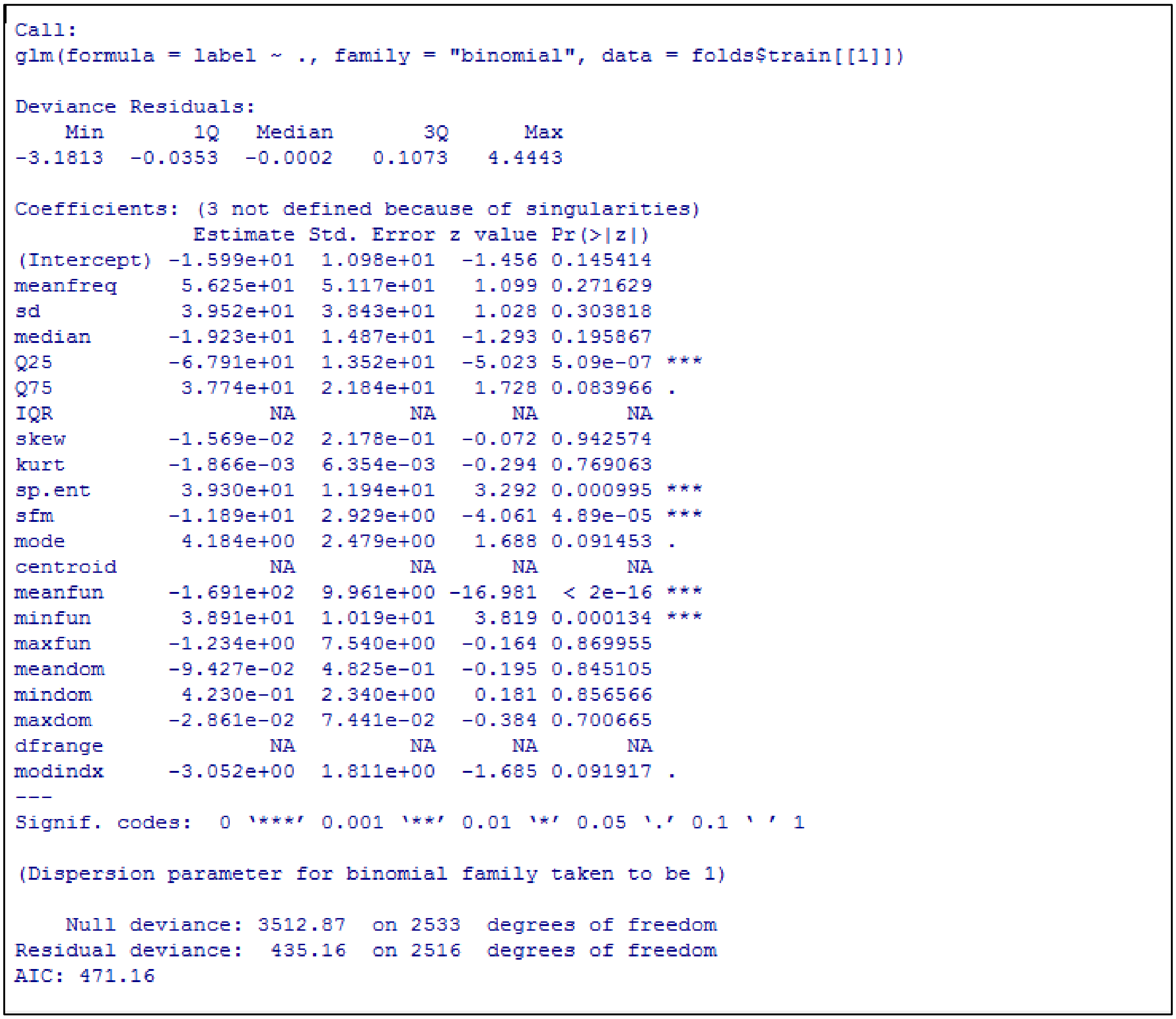
The above R command checks whether our data contains any missing (NA) values or not. The output 0 indicates that there are **no missing values** in our data set. Hence no need to process the data.

4.2. Data Analytics Tasks and Processes

1. Model learning for Solution 1 using logistic regression model
   a. Backward elimination (Manually):
      In this model, we start with the full model initially and then keep removing the x variable with the highest p value in each step as shown below:

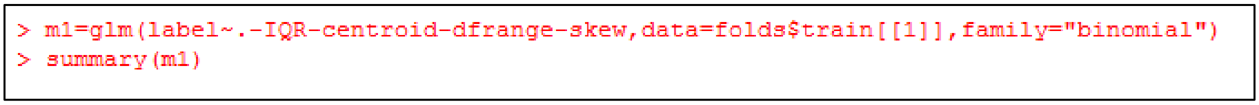


```
> full=glm(label~.,data=folds$train[[1]],family="binomial")
> summary(full)
```

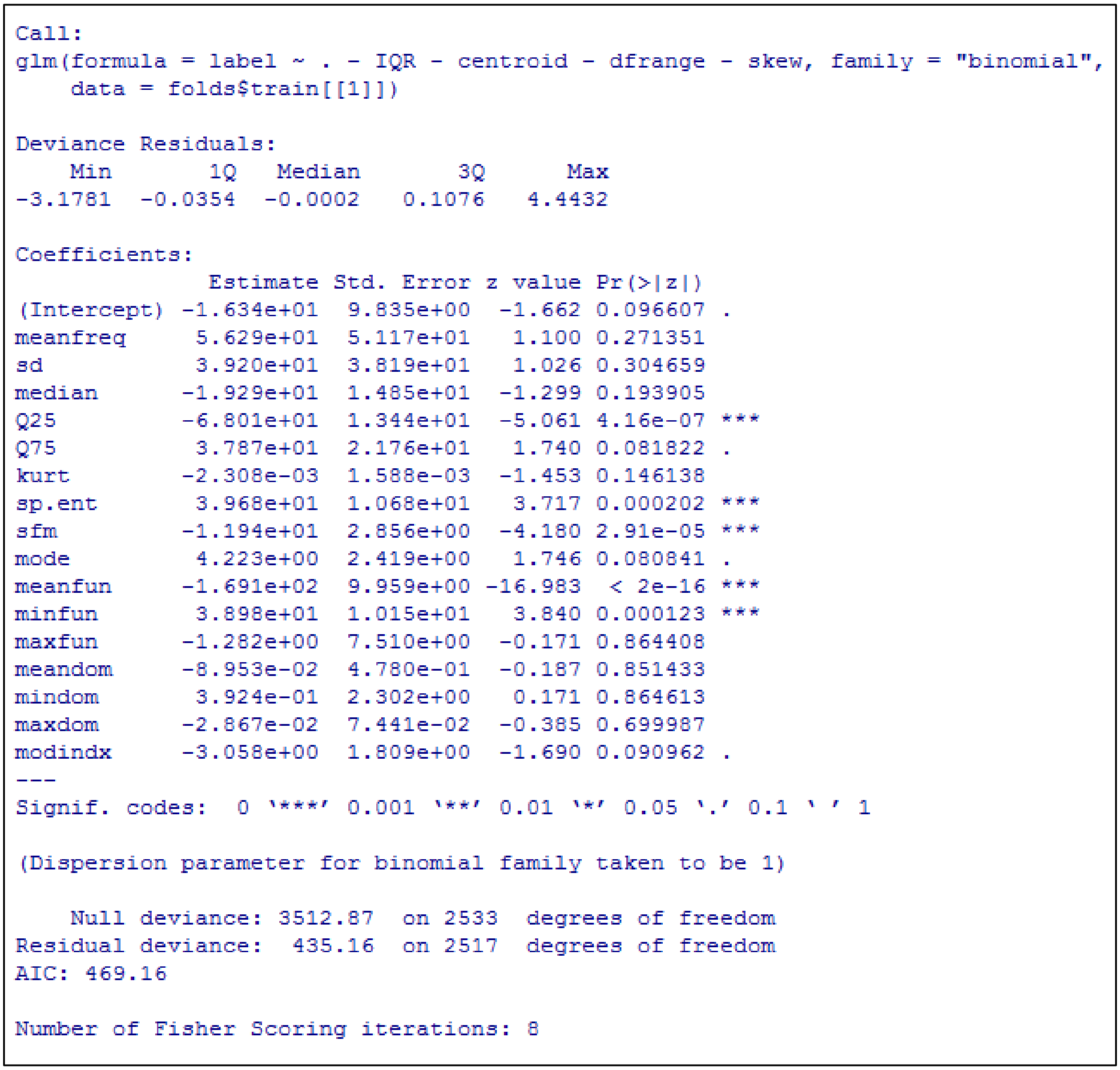


```
Call:
glm(formula = label ~ ., family = "binomial", data = folds$train[[1]])

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-3.1813  -0.0353  -0.0002   0.1073   4.4443  

Coefficients: (3 not defined because of singularities)
              Estimate Std. Error z value Pr(>|z|)    
(Intercept) -1.599e+01  1.098e+01  -1.456 0.145414    
meanfreq     5.625e+01  5.117e+01   1.099 0.271629    
sd           3.952e+01  3.843e+01   1.028 0.303818    
median      -1.923e+01  1.487e+01  -1.293 0.195867    
Q25         -6.791e+01  1.352e+01  -5.023 5.09e-07 ***
Q75          3.774e+01  2.184e+01   1.728 0.083966 .  
IQR                 NA         NA      NA       NA    
skew        -1.569e-02  2.178e-01  -0.072 0.942574    
kurt        -1.866e-03  6.354e-03  -0.294 0.769063    
sp.ent       3.930e+01  1.194e+01   3.292 0.000995 ***
sfm         -1.189e+01  2.929e+00  -4.061 4.89e-05 ***
mode         4.184e+00  2.479e+00   1.688 0.091453 .  
centroid            NA         NA      NA       NA    
meanfun     -1.691e+02  9.961e+00 -16.981  < 2e-16 ***
minfun       3.891e+01  1.019e+01   3.819 0.000134 ***
maxfun      -1.234e+00  7.540e+00  -0.164 0.869955    
meandom     -9.427e-02  4.825e-01  -0.195 0.845105    
mindom       4.230e-01  2.340e+00   0.181 0.856566    
maxdom      -2.861e-02  7.441e-02  -0.384 0.700665    
dfrange             NA         NA      NA       NA    
modindx     -3.052e+00  1.811e+00  -1.685 0.091917 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  435.16  on 2516  degrees of freedom
AIC: 471.16
```

In the next step, we remove x variable: skew as it has the highest p-value and since IQR, centroid and dfrange have p-values as NA we remove them too in this step as shown below:

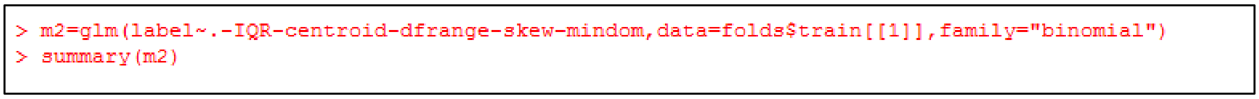


```
> m1=glm(label~.-IQR-centroid-dfrange-skew,data=folds$train[[1]],family="binomial")
> summary(m1)
```

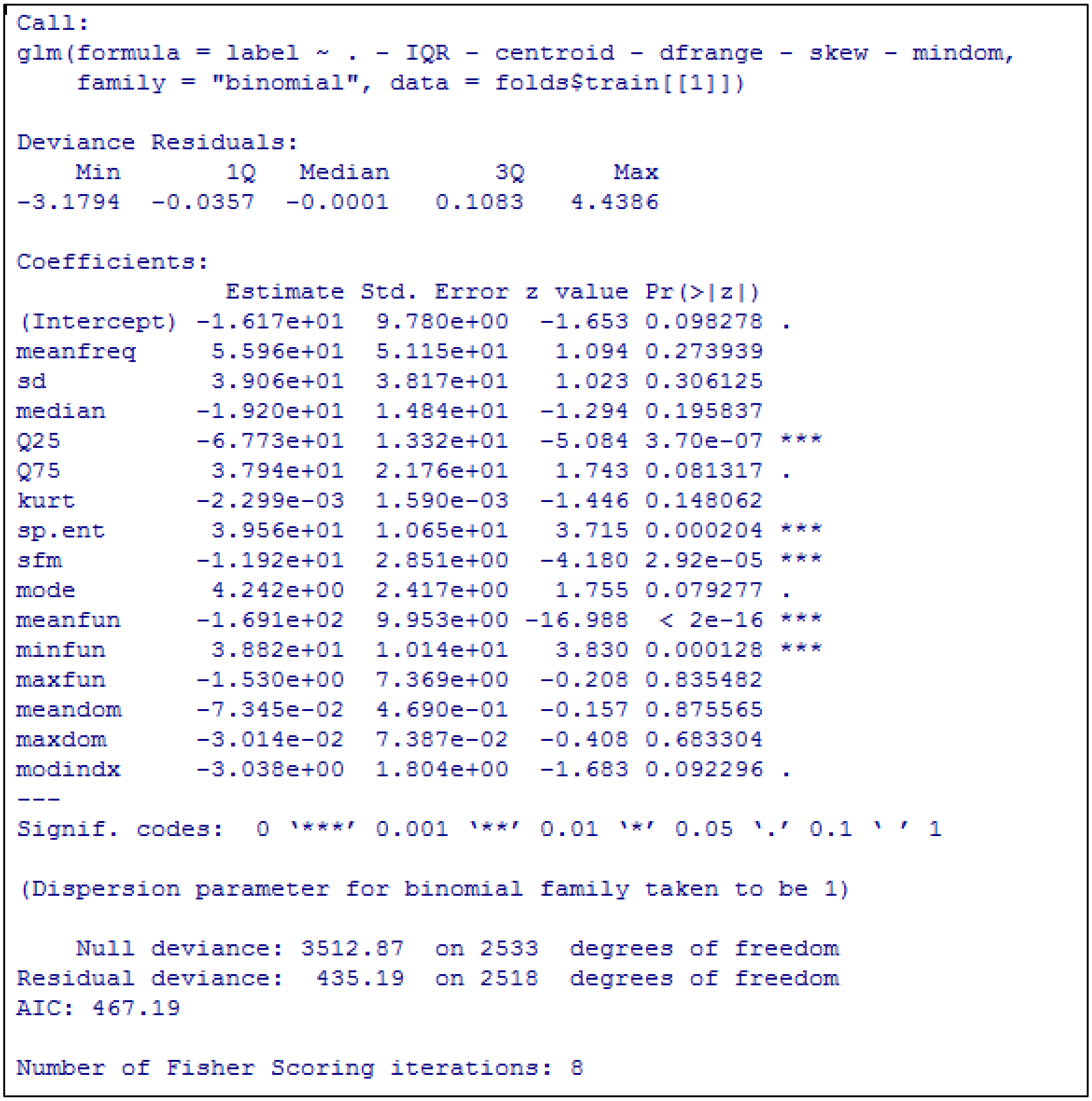


```
Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew, family = "binomial",
    data = folds$train[[1]])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1781  -0.0354  -0.0002   0.1076   4.4432

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.634e+01  9.835e+00  -1.662 0.096607 .
meanfreq     5.629e+01  5.117e+01   1.100 0.271351
sd           3.920e+01  3.819e+01   1.026 0.304659
median      -1.929e+01  1.485e+01  -1.299 0.193905
Q25         -6.801e+01  1.344e+01  -5.061 4.16e-07 ***
Q75          3.787e+01  2.176e+01   1.740 0.081822 .
kurt        -2.308e-03  1.588e-03  -1.453 0.146138
sp.ent       3.968e+01  1.068e+01   3.717 0.000202 ***
sfm         -1.194e+01  2.856e+00  -4.180 2.91e-05 ***
mode         4.223e+00  2.419e+00   1.746 0.080841 .
meanfun     -1.691e+02  9.959e+00 -16.983  < 2e-16 ***
minfun       3.898e+01  1.015e+01   3.840 0.000123 ***
maxfun      -1.282e+00  7.510e+00  -0.171 0.864408
meandom     -8.953e-02  4.780e-01  -0.187 0.851433
mindom       3.924e-01  2.302e+00   0.171 0.864613
maxdom      -2.867e-02  7.441e-02  -0.385 0.699987
modindx     -3.058e+00  1.809e+00  -1.690 0.090962 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  435.16  on 2517  degrees of freedom
AIC: 469.16

Number of Fisher Scoring iterations: 8
```

In the next step, we remove x variable: mindom as it has the highest p-value as shown below:

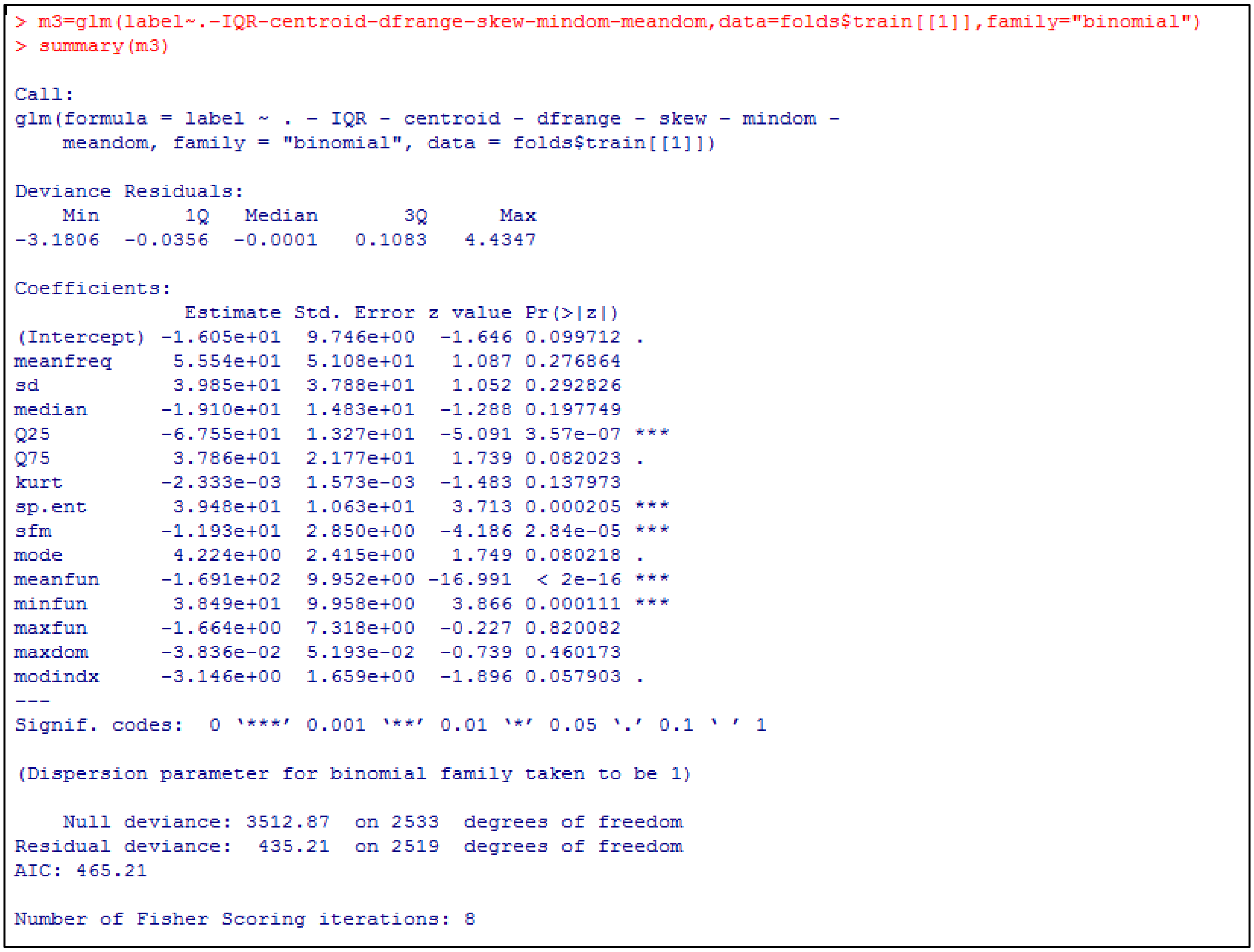


```
> m2=glm(label~.-IQR-centroid-dfrange-skew-mindom,data=folds$train[[1]],family="binomial")
> summary(m2)
```



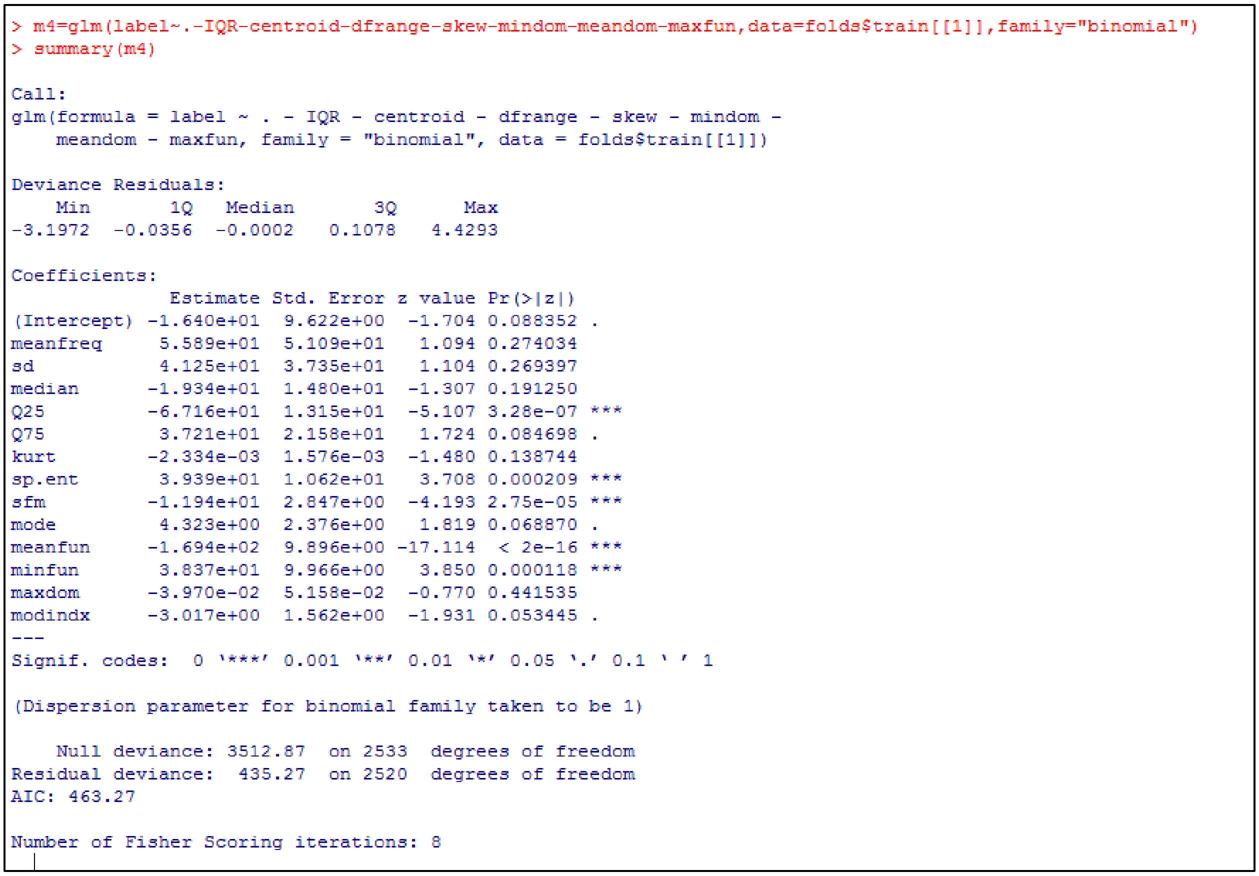
```
Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom,
    family = "binomial", data = folds$train[[1]])

Deviance Residuals:
    Min        1Q    Median        3Q       Max
-3.1794   -0.0357   -0.0001    0.1083    4.4386

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.617e+01  9.780e+00  -1.653 0.098278 .
meanfreq     5.596e+01  5.115e+01   1.094 0.273939
sd           3.906e+01  3.817e+01   1.023 0.306125
median      -1.920e+01  1.484e+01  -1.294 0.195837
Q25         -6.773e+01  1.332e+01  -5.084 3.70e-07 ***
Q75          3.794e+01  2.176e+01   1.743 0.081317 .
kurt        -2.299e-03  1.590e-03  -1.446 0.148062
sp.ent       3.956e+01  1.065e+01   3.715 0.000204 ***
sfm         -1.192e+01  2.851e+00  -4.180 2.92e-05 ***
mode         4.242e+00  2.417e+00   1.755 0.079277 .
meanfun     -1.691e+02  9.953e+00 -16.988  < 2e-16 ***
minfun       3.882e+01  1.014e+01   3.830 0.000128 ***
maxfun      -1.530e+00  7.369e+00  -0.208 0.835482
meandom     -7.345e-02  4.690e-01  -0.157 0.875565
maxdom      -3.014e-02  7.387e-02  -0.408 0.683304
modindx     -3.038e+00  1.804e+00  -1.683 0.092296 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  435.19  on 2518  degrees of freedom
AIC: 467.19

Number of Fisher Scoring iterations: 8
```

In the next step, we remove x variable: meandom as it has the highest p-value as shown below:



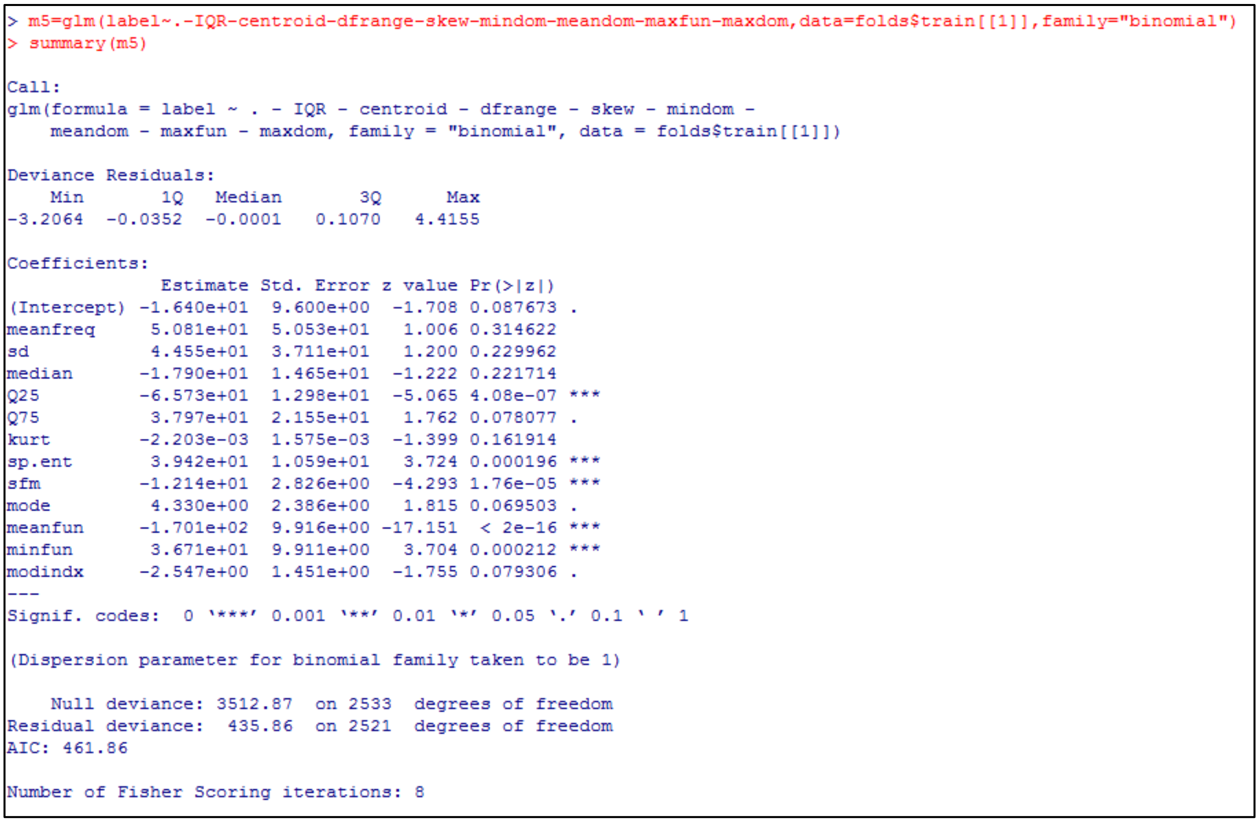
```
> m3=glm(label~.-IQR-centroid-dfrange-skew-mindom-meandom,data=folds$train[[1]],family="binomial")
> summary(m3)

Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom -
    meandom, family = "binomial", data = folds$train[[1]])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1806  -0.0356  -0.0001   0.1083   4.4347

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.605e+01  9.746e+00  -1.646 0.099712 .
meanfreq     5.554e+01  5.108e+01   1.087 0.276864
sd           3.985e+01  3.788e+01   1.052 0.292826
median      -1.910e+01  1.483e+01  -1.288 0.197749
Q25         -6.755e+01  1.327e+01  -5.091 3.57e-07 ***
Q75          3.786e+01  2.177e+01   1.739 0.082023 .
kurt        -2.333e-03  1.573e-03  -1.483 0.137973
sp.ent       3.948e+01  1.063e+01   3.713 0.000205 ***
sfm         -1.193e+01  2.850e+00  -4.186 2.84e-05 ***
mode         4.224e+00  2.415e+00   1.749 0.080218 .
meanfun     -1.691e+02  9.952e+00 -16.991  < 2e-16 ***
minfun       3.849e+01  9.958e+00   3.866 0.000111 ***
maxfun      -1.664e+00  7.318e+00  -0.227 0.820082
maxdom      -3.836e-02  5.193e-02  -0.739 0.460173
modindx     -3.146e+00  1.659e+00  -1.896 0.057903 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  435.21  on 2519  degrees of freedom
AIC: 465.21

Number of Fisher Scoring iterations: 8
```

In the next step, we remove x variable: maxfun as it has the highest p-value as shown below:



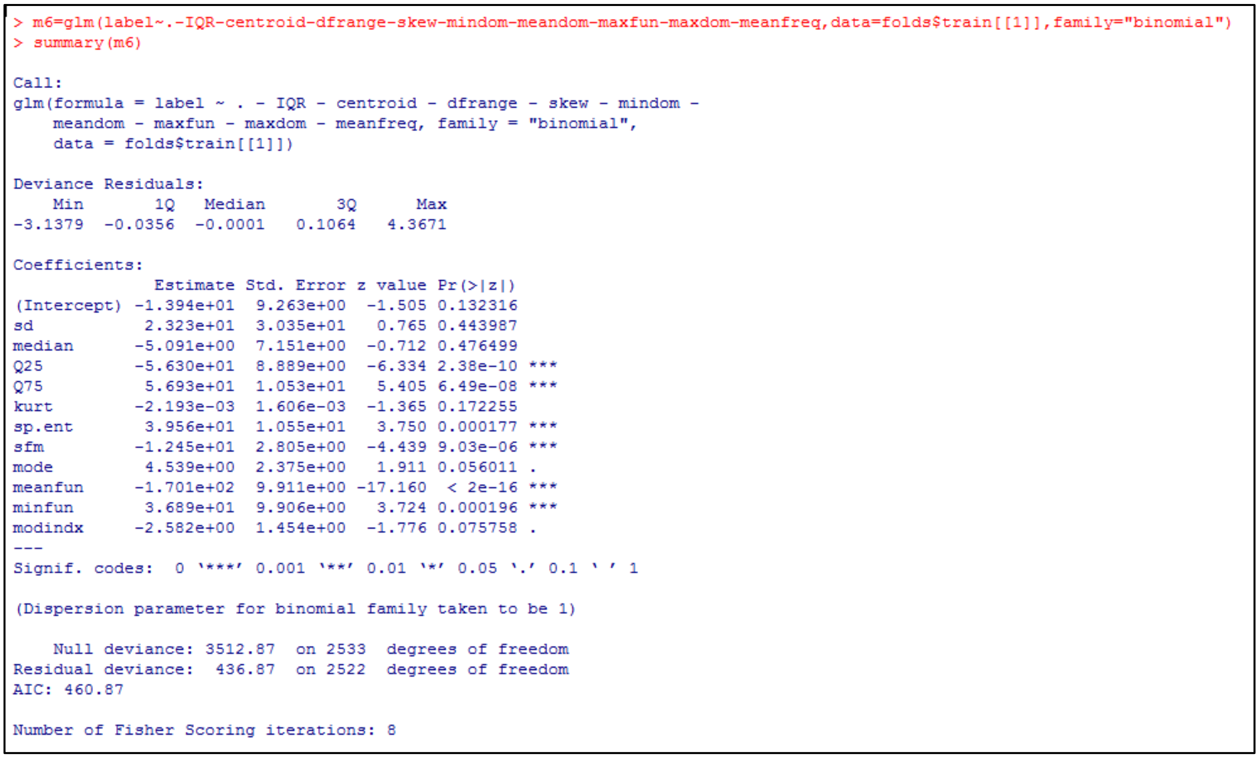
```
> m4=glm(label~.-IQR-centroid-dfrange-skew-mindom-meandom-maxfun,data=folds$train[[1]],family="binomial")
> summary(m4)

Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom - 
    meandom - maxfun, family = "binomial", data = folds$train[[1]])

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-3.1972  -0.0356  -0.0002   0.1078   4.4293  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept) -1.640e+01  9.622e+00  -1.704 0.088352 .  
meanfreq     5.589e+01  5.109e+01   1.094 0.274034    
sd           4.125e+01  3.735e+01   1.104 0.269397    
median      -1.934e+01  1.480e+01  -1.307 0.191250    
Q25         -6.716e+01  1.315e+01  -5.107 3.28e-07 ***
Q75          3.721e+01  2.158e+01   1.724 0.084698 .  
kurt        -2.334e-03  1.576e-03  -1.480 0.138744    
sp.ent       3.939e+01  1.062e+01   3.708 0.000209 ***
sfm         -1.194e+01  2.847e+00  -4.193 2.75e-05 ***
mode         4.323e+00  2.376e+00   1.819 0.068870 .  
meanfun     -1.694e+02  9.896e+00 -17.114  < 2e-16 ***
minfun       3.837e+01  9.966e+00   3.850 0.000118 ***
maxdom      -3.970e-02  5.158e-02  -0.770 0.441535    
modindx     -3.017e+00  1.562e+00  -1.931 0.053445 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  435.27  on 2520  degrees of freedom
AIC: 463.27

Number of Fisher Scoring iterations: 8
```

In the next step, we remove x variable: maxdom as it has the highest p-value as shown below:



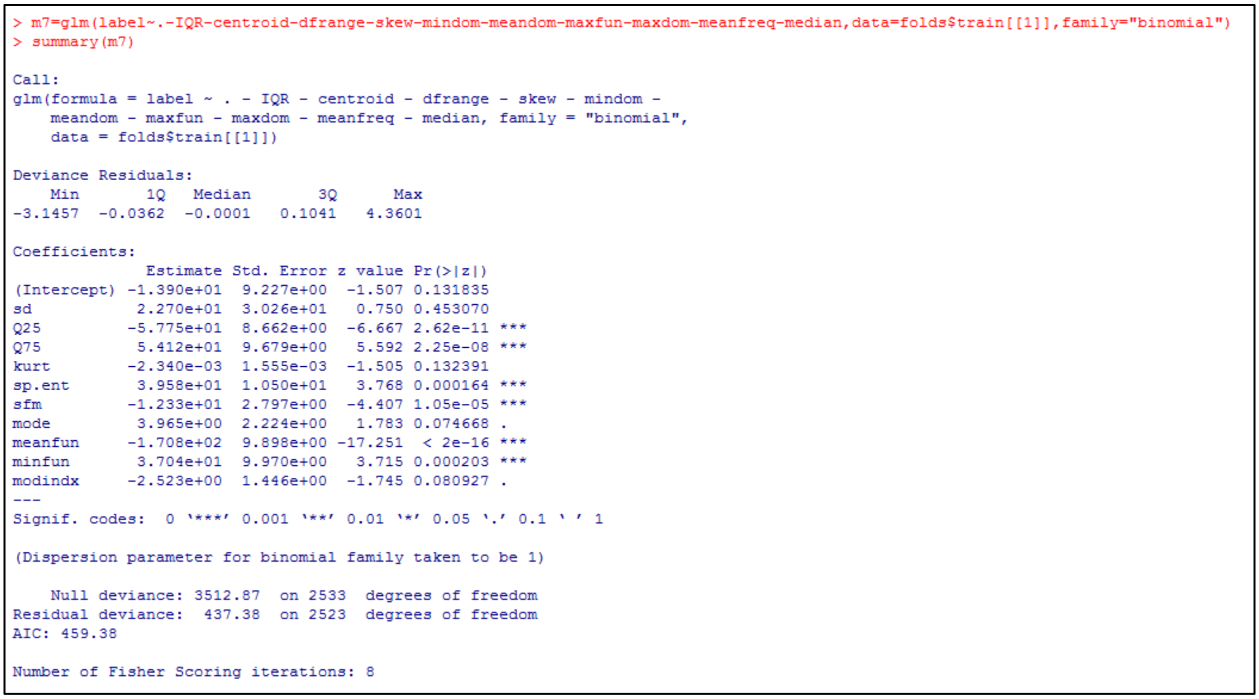
```
> m5=glm(label~.-IQR-centroid-dfrange-skew-mindom-meandom-maxfun-maxdom,data=folds$train[[1]],family="binomial")
> summary(m5)

Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom -
    meandom - maxfun - maxdom, family = "binomial", data = folds$train[[1]])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2064  -0.0352  -0.0001   0.1070   4.4155

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.640e+01  9.600e+00  -1.708 0.087673 .
meanfreq     5.081e+01  5.053e+01   1.006 0.314622
sd           4.455e+01  3.711e+01   1.200 0.229962
median      -1.790e+01  1.465e+01  -1.222 0.221714
Q25         -6.573e+01  1.298e+01  -5.065 4.08e-07 ***
Q75          3.797e+01  2.155e+01   1.762 0.078077 .
kurt        -2.203e-03  1.575e-03  -1.399 0.161914
sp.ent       3.942e+01  1.059e+01   3.724 0.000196 ***
sfm         -1.214e+01  2.826e+00  -4.293 1.76e-05 ***
mode         4.330e+00  2.386e+00   1.815 0.069503 .
meanfun     -1.701e+02  9.916e+00 -17.151  < 2e-16 ***
minfun       3.671e+01  9.911e+00   3.704 0.000212 ***
modindx     -2.547e+00  1.451e+00  -1.755 0.079306 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  435.86  on 2521  degrees of freedom
AIC: 461.86

Number of Fisher Scoring iterations: 8
```

In the next step, we remove x variable: meanfreq as it has the highest p-value as shown below:



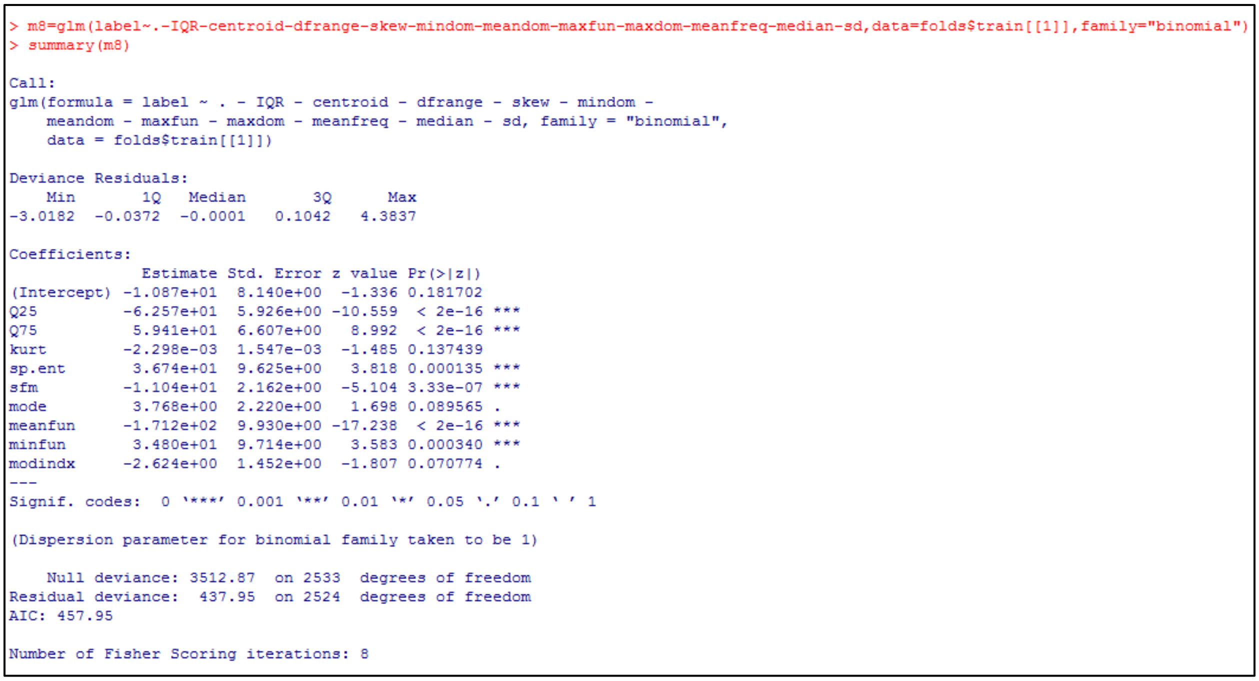
```
> m6=glm(label~.-IQR-centroid-dfrange-skew-mindom-meandom-maxfun-maxdom-meanfreq,data=folds$train[[1]],family="binomial")
> summary(m6)

Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom -
    meandom - maxfun - maxdom - meanfreq, family = "binomial",
    data = folds$train[[1]])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1379  -0.0356  -0.0001   0.1064   4.3671

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.394e+01  9.263e+00  -1.505 0.132316
sd           2.323e+01  3.035e+01   0.765 0.443987
median      -5.091e+00  7.151e+00  -0.712 0.476499
Q25         -5.630e+01  8.889e+00  -6.334 2.38e-10 ***
Q75          5.693e+01  1.053e+01   5.405 6.49e-08 ***
kurt        -2.193e-03  1.606e-03  -1.365 0.172255
sp.ent       3.956e+01  1.055e+01   3.750 0.000177 ***
sfm         -1.245e+01  2.805e+00  -4.439 9.03e-06 ***
mode         4.539e+00  2.375e+00   1.911 0.056011 .
meanfun     -1.701e+02  9.911e+00 -17.160  < 2e-16 ***
minfun       3.689e+01  9.906e+00   3.724 0.000196 ***
modindx     -2.582e+00  1.454e+00  -1.776 0.075758 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  436.87  on 2522  degrees of freedom
AIC: 460.87

Number of Fisher Scoring iterations: 8
```

In the next step, we remove x variable: median as it has the highest p-value as shown below:



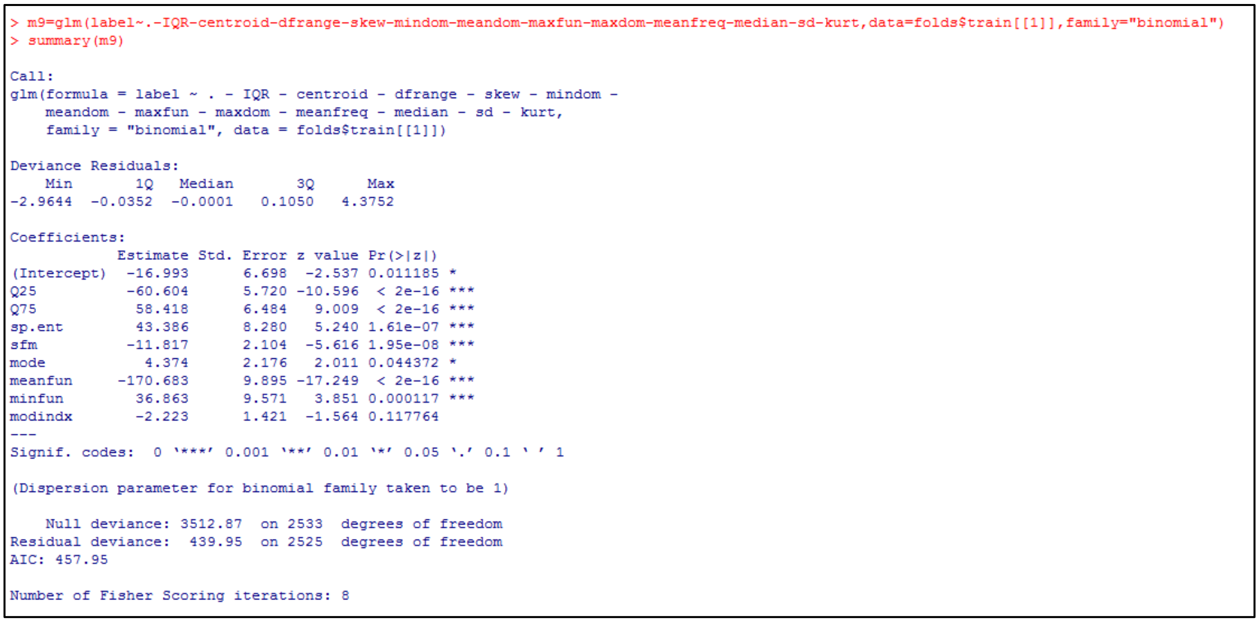
```
> m7=glm(label~.-IQR-centroid-dfrange-skew-mindom-meandom-maxfun-maxdom-meanfreq-median,data=folds$train[[1]],family="binomial")
> summary(m7)

Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom - 
    meandom - maxfun - maxdom - meanfreq - median, family = "binomial", 
    data = folds$train[[1]])

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-3.1457  -0.0362  -0.0001   0.1041   4.3601  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept) -1.390e+01  9.227e+00  -1.507 0.131835    
sd           2.270e+01  3.026e+01   0.750 0.453070    
Q25         -5.775e+01  8.662e+00  -6.667 2.62e-11 ***
Q75          5.412e+01  9.679e+00   5.592 2.25e-08 ***
kurt        -2.340e-03  1.555e-03  -1.505 0.132391    
sp.ent       3.958e+01  1.050e+01   3.768 0.000164 ***
sfm         -1.233e+01  2.797e+00  -4.407 1.05e-05 ***
mode         3.965e+00  2.224e+00   1.783 0.074668 .  
meanfun     -1.708e+02  9.898e+00 -17.251  < 2e-16 ***
minfun       3.704e+01  9.970e+00   3.715 0.000203 ***
modindx     -2.523e+00  1.446e+00  -1.745 0.080927 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  437.38  on 2523  degrees of freedom
AIC: 459.38

Number of Fisher Scoring iterations: 8
```

In the next step, we remove x variable: sd as it has the highest p-value as shown below:



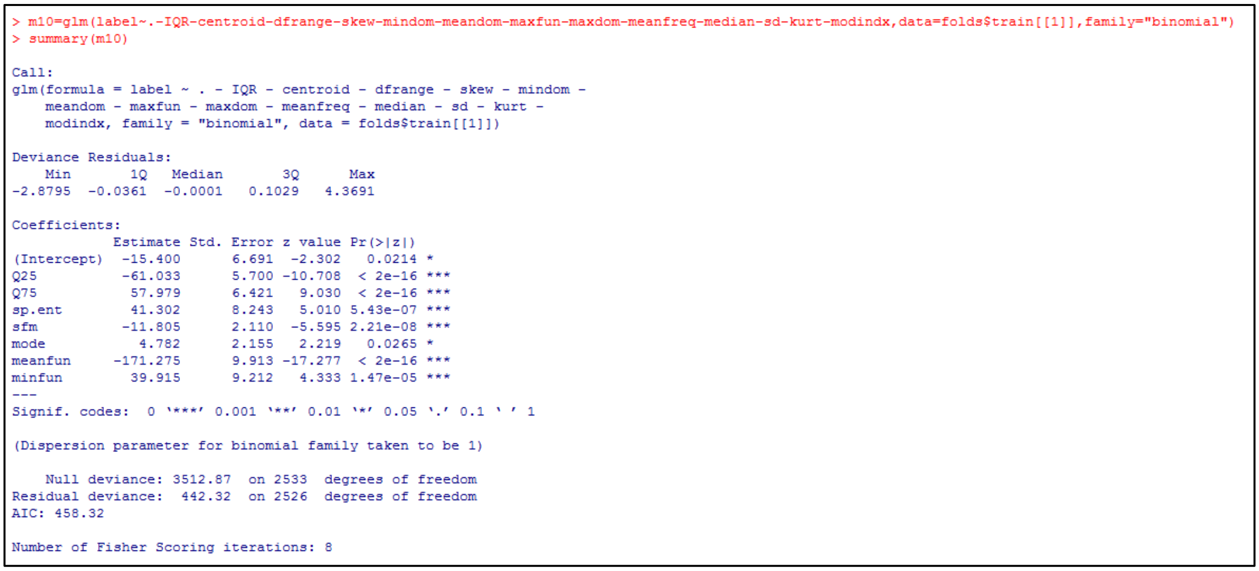
```
> m8=glm(label~.-IQR-centroid-dfrange-skew-mindom-meandom-maxfun-maxdom-meanfreq-median-sd,data=folds$train[[1]],family="binomial")
> summary(m8)

Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom - 
    meandom - maxfun - maxdom - meanfreq - median - sd, family = "binomial", 
    data = folds$train[[1]])

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-3.0182  -0.0372  -0.0001   0.1042   4.3837  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept) -1.087e+01  8.140e+00  -1.336 0.181702    
Q25         -6.257e+01  5.926e+00 -10.559  < 2e-16 ***
Q75          5.941e+01  6.607e+00   8.992  < 2e-16 ***
kurt        -2.298e-03  1.547e-03  -1.485 0.137439    
sp.ent       3.674e+01  9.625e+00   3.818 0.000135 ***
sfm         -1.104e+01  2.162e+00  -5.104 3.33e-07 ***
mode         3.768e+00  2.220e+00   1.698 0.089565 .  
meanfun     -1.712e+02  9.930e+00 -17.238  < 2e-16 ***
minfun       3.480e+01  9.714e+00   3.583 0.000340 ***
modindx     -2.624e+00  1.452e+00  -1.807 0.070774 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  437.95  on 2524  degrees of freedom
AIC: 457.95

Number of Fisher Scoring iterations: 8
```

In the next step, we remove x variable: kurt as it has the highest p-value as shown below:



```
> m9=glm(label~.-IQR-centroid-dfrange-skew-mindom-meandom-maxfun-maxdom-meanfreq-median-sd-kurt,data=folds$train[[1]],family="binomial")
> summary(m9)

Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom -
    meandom - maxfun - maxdom - meanfreq - median - sd - kurt,
    family = "binomial", data = folds$train[[1]])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9644  -0.0352  -0.0001   0.1050   4.3752

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -16.993      6.698  -2.537 0.011185 *
Q25          -60.604      5.720 -10.596  < 2e-16 ***
Q75           58.418      6.484   9.009  < 2e-16 ***
sp.ent        43.386      8.280   5.240 1.61e-07 ***
sfm          -11.817      2.104  -5.616 1.95e-08 ***
mode           4.374      2.176   2.011 0.044372 *
meanfun     -170.683      9.895 -17.249  < 2e-16 ***
minfun        36.863      9.571   3.851 0.000117 ***
modindx       -2.223      1.421  -1.564 0.117764
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  439.95  on 2525  degrees of freedom
AIC: 457.95

Number of Fisher Scoring iterations: 8
```

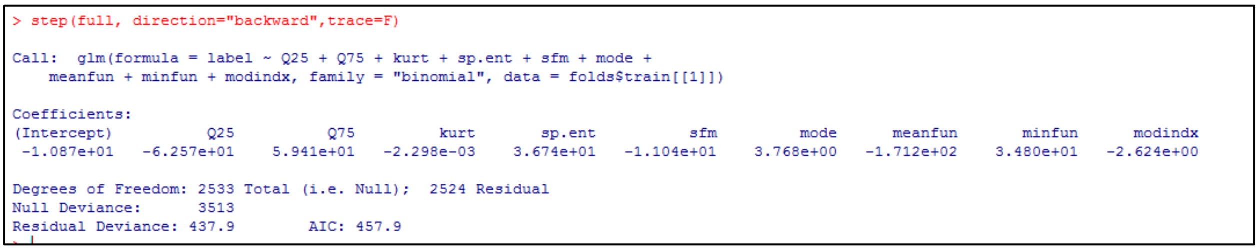
In the next step, we remove x variable: modindx as it has the highest p-value as shown below:



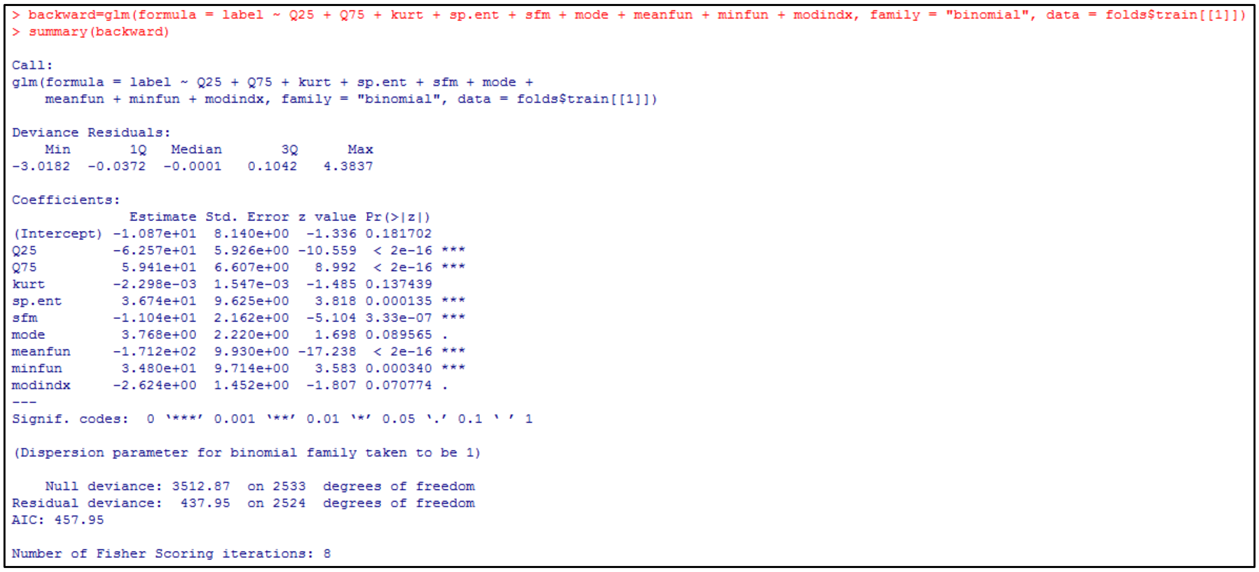
```
> m10=glm(label~.-IQR-centroid-dfrange-skew-mindom-meandom-maxfun-maxdom-meanfreq-median-sd-kurt-modindx,data=folds$train[[1]],family="binomial")
> summary(m10)

Call:
glm(formula = label ~ . - IQR - centroid - dfrange - skew - mindom -
    meandom - maxfun - maxdom - meanfreq - median - sd - kurt -
    modindx, family = "binomial", data = folds$train[[1]])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8795  -0.0361  -0.0001   0.1029   4.3691

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -15.400      6.691  -2.302   0.0214 *
Q25          -61.033      5.700 -10.708  < 2e-16 ***
Q75           57.979      6.421   9.030  < 2e-16 ***
sp.ent        41.302      8.243   5.010 5.43e-07 ***
sfm          -11.805      2.110  -5.595 2.21e-08 ***
mode           4.782      2.155   2.219   0.0265 *
meanfun     -171.275      9.913 -17.277  < 2e-16 ***
minfun        39.915      9.212   4.333 1.47e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  442.32  on 2526  degrees of freedom
AIC: 458.32

Number of Fisher Scoring iterations: 8
```

Thus, in the model m10 we see that all the x variables has p-values greater than 0.05 and hence, we do not remove any x variable further.

b. Stepwise backward elimination using step() function

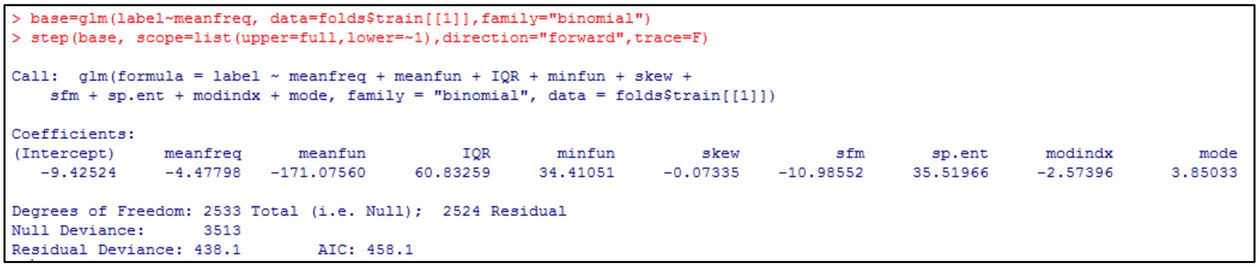


```
> step(full, direction="backward",trace=F)

Call:  glm(formula = label ~ Q25 + Q75 + kurt + sp.ent + sfm + mode +
    meanfun + minfun + modindx, family = "binomial", data = folds$train[[1]])

Coefficients:
(Intercept)          Q25          Q75         kurt       sp.ent          sfm         mode      meanfun       minfun      modindx
 -1.087e+01   -6.257e+01    5.941e+01   -2.298e-03    3.674e+01   -1.104e+01    3.768e+00   -1.712e+02    3.480e+01   -2.624e+00

Degrees of Freedom: 2533 Total (i.e. Null);  2524 Residual
Null Deviance:      3513
Residual Deviance: 437.9        AIC: 457.9
```

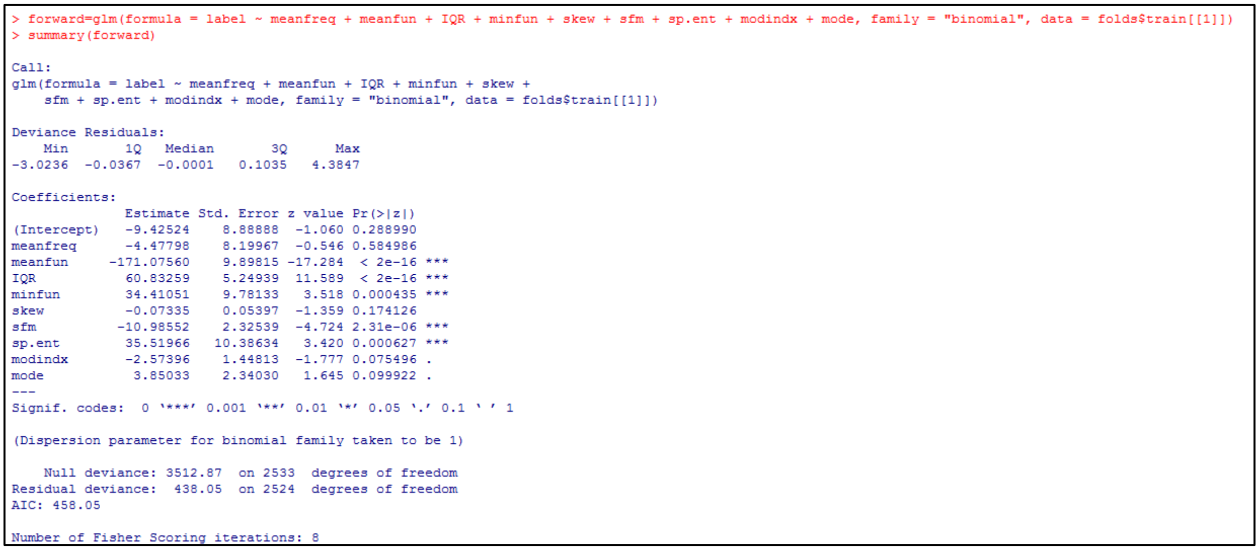


```
> backward=glm(formula = label ~ Q25 + Q75 + kurt + sp.ent + sfm + mode + meanfun + minfun + modindx, family = "binomial", data = folds$train[[1]])
> summary(backward)

Call:
glm(formula = label ~ Q25 + Q75 + kurt + sp.ent + sfm + mode +
    meanfun + minfun + modindx, family = "binomial", data = folds$train[[1]])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.0182  -0.0372  -0.0001   0.1042   4.3837

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.087e+01  8.140e+00  -1.336 0.181702
Q25         -6.257e+01  5.926e+00 -10.559  < 2e-16 ***
Q75          5.941e+01  6.607e+00   8.992  < 2e-16 ***
kurt        -2.298e-03  1.547e-03  -1.485 0.137439
sp.ent       3.674e+01  9.625e+00   3.818 0.000135 ***
sfm         -1.104e+01  2.162e+00  -5.104 3.33e-07 ***
mode         3.768e+00  2.220e+00   1.698 0.089565 .
meanfun     -1.712e+02  9.930e+00 -17.238  < 2e-16 ***
minfun       3.480e+01  9.714e+00   3.583 0.000340 ***
modindx     -2.624e+00  1.452e+00  -1.807 0.070774 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  437.95  on 2524  degrees of freedom
AIC: 457.95

Number of Fisher Scoring iterations: 8
```

c. Stepwise forward elimination using step() function

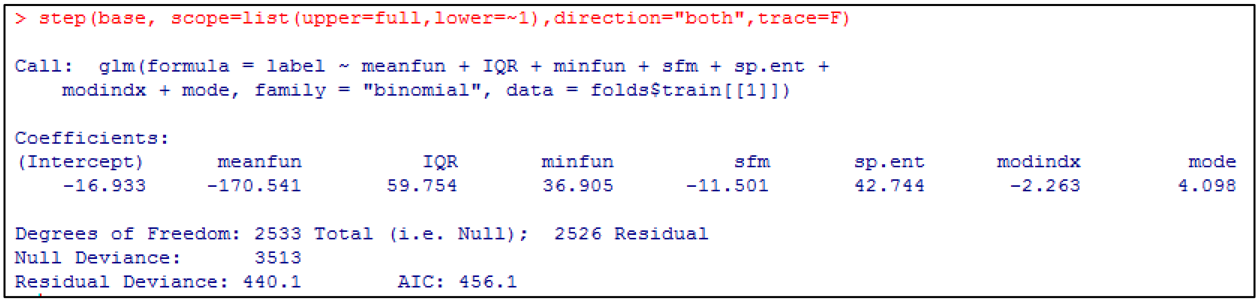


```
> base=glm(label~meanfreq, data=folds$train[[1]],family="binomial")
> step(base, scope=list(upper=full,lower=~1),direction="forward",trace=F)

Call:  glm(formula = label ~ meanfreq + meanfun + IQR + minfun + skew +
    sfm + sp.ent + modindx + mode, family = "binomial", data = folds$train[[1]])

Coefficients:
(Intercept)     meanfreq      meanfun          IQR       minfun         skew          sfm       sp.ent      modindx         mode
   -9.42524     -4.47798   -171.07560     60.83259     34.41051     -0.07335    -10.98552     35.51966     -2.57396      3.85033

Degrees of Freedom: 2533 Total (i.e. Null);  2524 Residual
Null Deviance:      3513
Residual Deviance: 438.1        AIC: 458.1
```

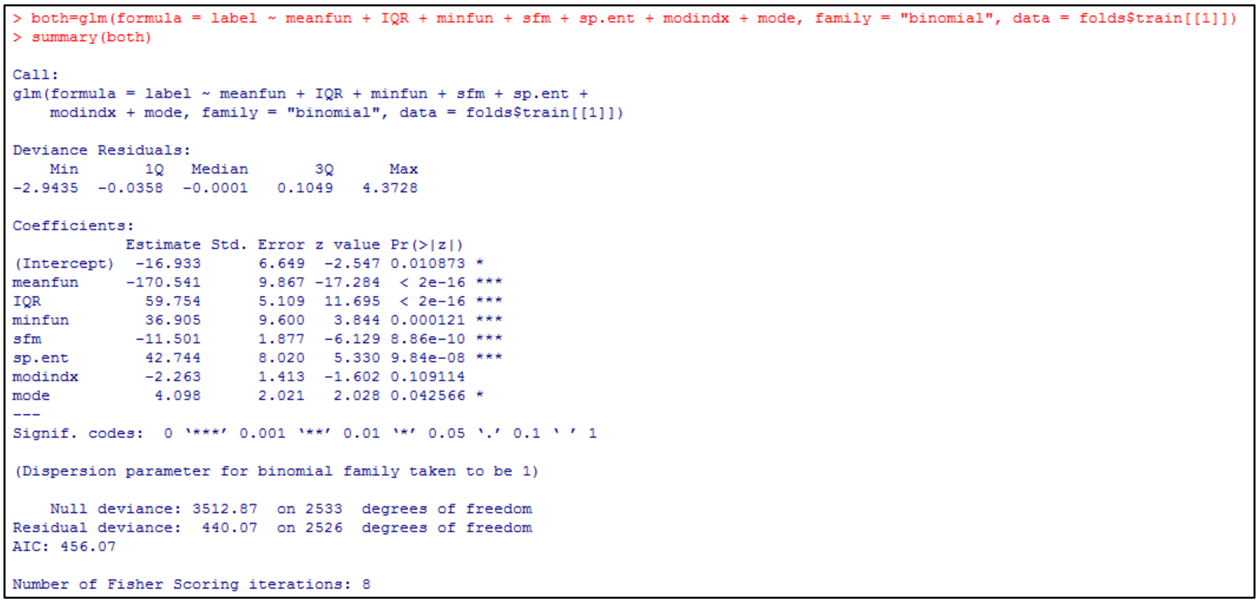


```
> forward=glm(formula = label ~ meanfreq + meanfun + IQR + minfun + skew + sfm + sp.ent + modindx + mode, family = "binomial", data = folds$train[[1]])
> summary(forward)

Call:
glm(formula = label ~ meanfreq + meanfun + IQR + minfun + skew +
    sfm + sp.ent + modindx + mode, family = "binomial", data = folds$train[[1]])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.0236  -0.0367  -0.0001   0.1035   4.3847

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -9.42524    8.88888  -1.060 0.288990
meanfreq      -4.47798    8.19967  -0.546 0.584986
meanfun     -171.07560    9.89815 -17.284  < 2e-16 ***
IQR           60.83259    5.24939  11.589  < 2e-16 ***
minfun        34.41051    9.78133   3.518 0.000435 ***
skew          -0.07335    0.05397  -1.359 0.174126
sfm          -10.98552    2.32539  -4.724 2.31e-06 ***
sp.ent        35.51966   10.38634   3.420 0.000627 ***
modindx       -2.57396    1.44813  -1.777 0.075496 .
mode           3.85033    2.34030   1.645 0.099922 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  438.05  on 2524  degrees of freedom
AIC: 458.05

Number of Fisher Scoring iterations: 8
```

d. Stepwise elimination with direction=both

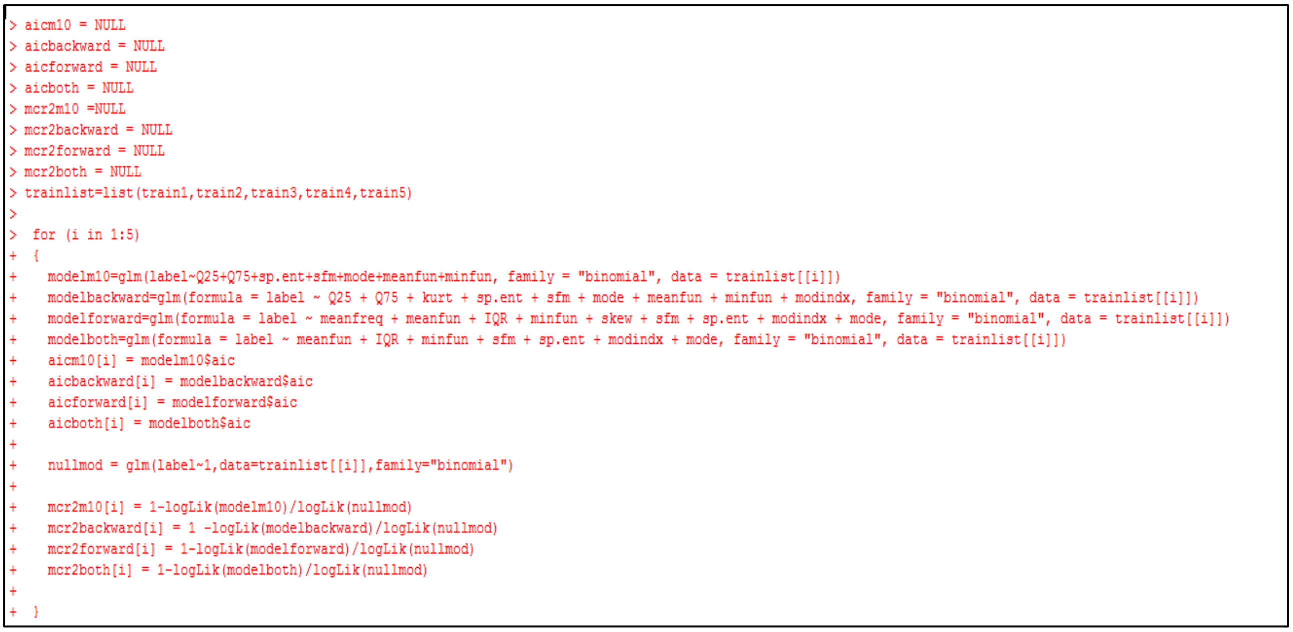


```
> step(base, scope=list(upper=full,lower=~1),direction="both",trace=F)

Call:  glm(formula = label ~ meanfun + IQR + minfun + sfm + sp.ent + 
    modindx + mode, family = "binomial", data = folds$train[[1]])

Coefficients:
(Intercept)      meanfun          IQR       minfun          sfm       sp.ent      modindx         mode  
    -16.933     -170.541       59.754       36.905      -11.501       42.744       -2.263        4.098  

Degrees of Freedom: 2533 Total (i.e. Null);  2526 Residual
Null Deviance:      3513 
Residual Deviance: 440.1        AIC: 456.1
```



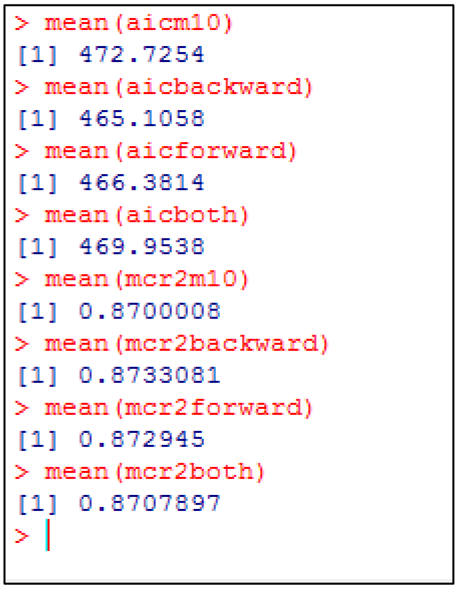
```
> both=glm(formula = label ~ meanfun + IQR + minfun + sfm + sp.ent + modindx + mode, family = "binomial", data = folds$train[[1]])
> summary(both)

Call:
glm(formula = label ~ meanfun + IQR + minfun + sfm + sp.ent + 
    modindx + mode, family = "binomial", data = folds$train[[1]])

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-2.9435  -0.0358  -0.0001   0.1049   4.3728  

Coefficients:
            Estimate Std. Error z value Pr(>|z|)    
(Intercept)  -16.933      6.649  -2.547 0.010873 *  
meanfun     -170.541      9.867 -17.284  < 2e-16 ***
IQR           59.754      5.109  11.695  < 2e-16 ***
minfun        36.905      9.600   3.844 0.000121 ***
sfm          -11.501      1.877  -6.129 8.86e-10 ***
sp.ent        42.744      8.020   5.330 9.84e-08 ***
modindx       -2.263      1.413  -1.602 0.109114    
mode           4.098      2.021   2.028 0.042566 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  440.07  on 2526  degrees of freedom
AIC: 456.07

Number of Fisher Scoring iterations: 8
```

e. Computing AIC and Mc Fadden R^2 for models built above on the remaining Training datasets train2, train3, train4 and train5 in the for loop shown below:



```
> aicm10 = NULL
> aicbackward = NULL
> aicforward = NULL
> aicboth = NULL
> mcr2m10 =NULL
> mcr2backward = NULL
> mcr2forward = NULL
> mcr2both = NULL
> trainlist=list(train1,train2,train3,train4,train5)
>
>  for (i in 1:5)
+  {
+    modelm10=glm(label~Q25+Q75+sp.ent+sfm+mode+meanfun+minfun, family = "binomial", data = trainlist[[i]])
+    modelbackward=glm(formula = label ~ Q25 + Q75 + kurt + sp.ent + sfm + mode + meanfun + minfun + modindx, family = "binomial", data = trainlist[[i]])
+    modelforward=glm(formula = label ~ meanfreq + meanfun + IQR + minfun + skew + sfm + sp.ent + modindx + mode, family = "binomial", data = trainlist[[i]])
+    modelboth=glm(formula = label ~ meanfun + IQR + minfun + sfm + sp.ent + modindx + mode, family = "binomial", data = trainlist[[i]])
+    aicm10[i] = modelm10$aic
+    aicbackward[i] = modelbackward$aic
+    aicforward[i] = modelforward$aic
+    aicboth[i] = modelboth$aic
+
+    nullmod = glm(label~1,data=trainlist[[i]],family="binomial")
+
+    mcr2m10[i] = 1-logLik(modelm10)/logLik(nullmod)
+    mcr2backward[i] = 1 -logLik(modelbackward)/logLik(nullmod)
+    mcr2forward[i] = 1-logLik(modelforward)/logLik(nullmod)
+    mcr2both[i] = 1-logLik(modelboth)/logLik(nullmod)
+
+  }
```

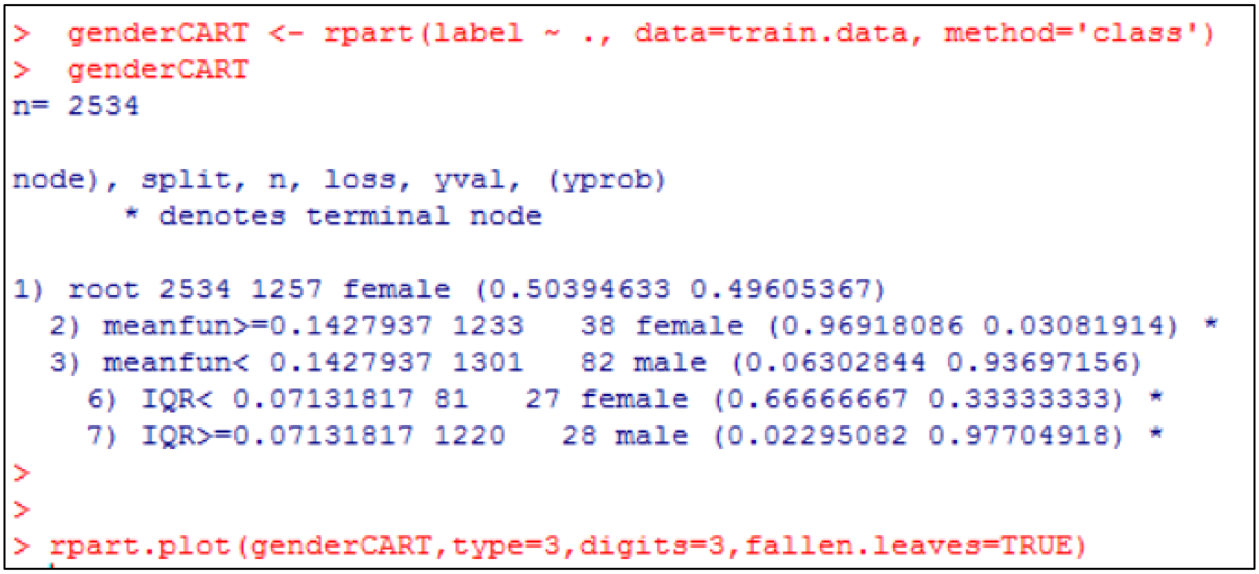
Output of the above loop gives the AIC and Mc Fadden R^2 of the four models shown in step a, b, c and d:



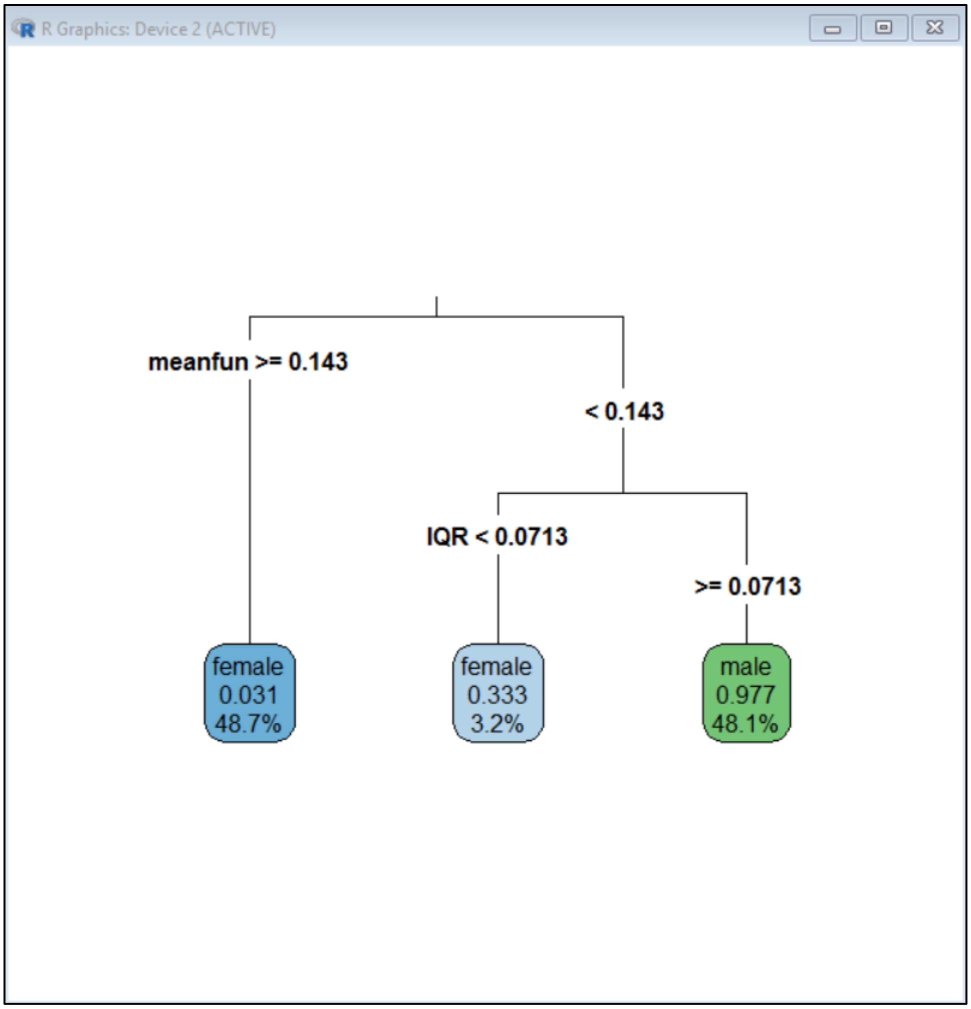
```
> mean(aicm10)
[1] 472.7254
> mean(aicbackward)
[1] 465.1058
> mean(aicforward)
[1] 466.3814
> mean(aicboth)
[1] 469.9538
> mean(mcr2m10)
[1] 0.8700008
> mean(mcr2backward)
[1] 0.8733081
> mean(mcr2forward)
[1] 0.872945
> mean(mcr2both)
[1] 0.8707897
> |
```

2. Model learning for Solution 2 using CART Model

The rpart() function below create a CART model on the training dataset and the rplot.plot() function creates a CART plot:

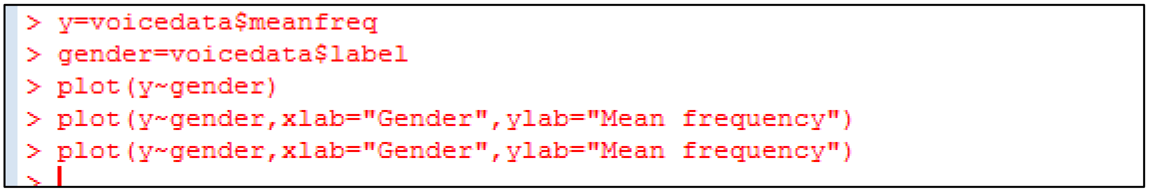


```
>  genderCART <- rpart(label ~ ., data=train.data, method='class')
>  genderCART
n= 2534 

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 2534 1257 female (0.50394633 0.49605367)  
  2) meanfun>=0.1427937 1233   38 female (0.96918086 0.03081914) *
  3) meanfun< 0.1427937 1301   82 male (0.06302844 0.93697156)  
    6) IQR< 0.07131817 81   27 female (0.66666667 0.33333333) *
    7) IQR>=0.07131817 1220   28 male (0.02295082 0.97704918) *
> 
> 
> rpart.plot(genderCART,type=3,digits=3,fallen.leaves=TRUE)
```

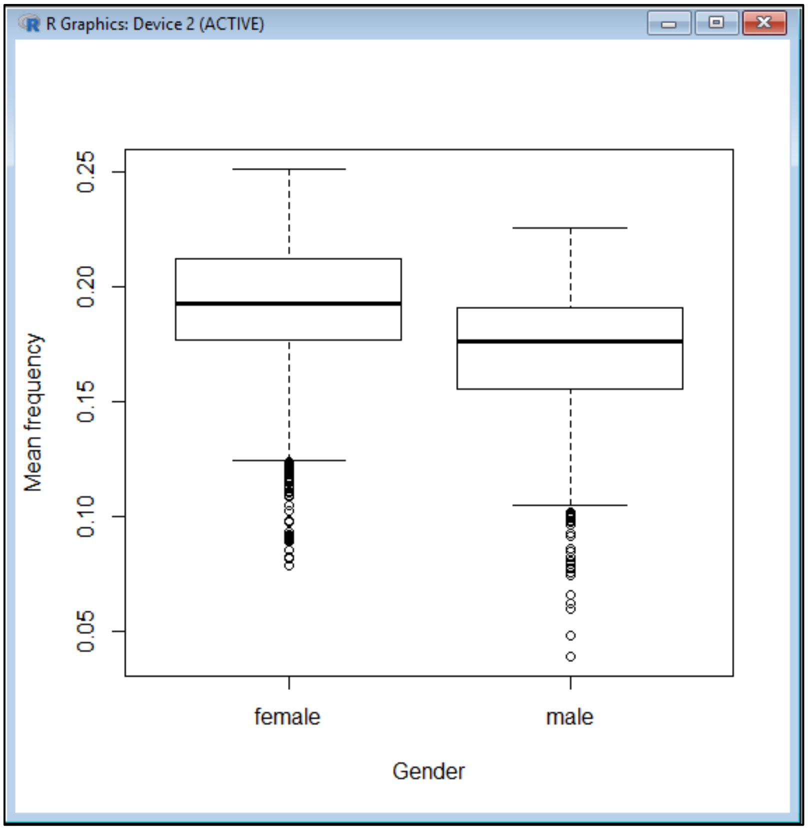


3. Model learning for Solution 3

The below code is used to plot the box plot of the mean frequency of female and male.



```
> y=voicedata$meanfreq
> gender=voicedata$label
> plot(y~gender)
> plot(y~gender,xlab="Gender",ylab="Mean frequency")
> plot(y~gender,xlab="Gender",ylab="Mean frequency")
> 
```



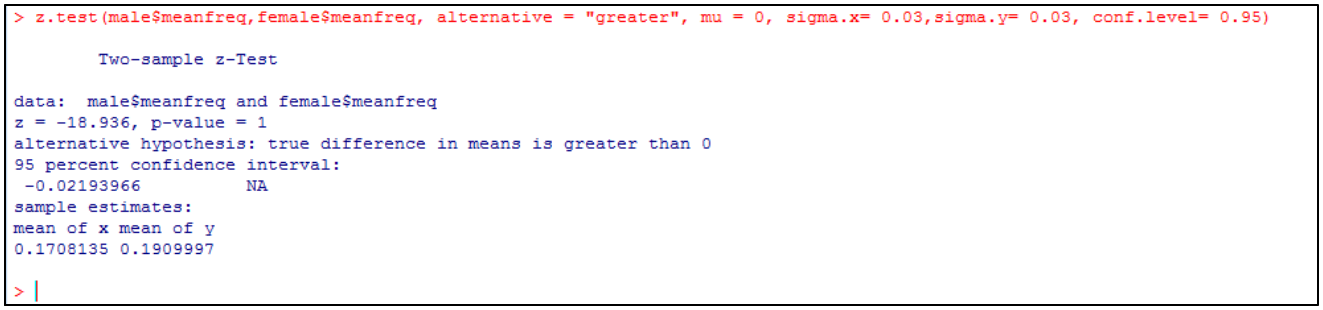
We can clearly see that the mean frequency of female voices is greater than the mean frequency of the male voices.

Here, we also performed hypothesis test to analyze whether the male mean frequency is higher than the female mean frequency:

$\boldsymbol{H_0}$: µ1 ≤ µ2
$\boldsymbol{H_a}$: µ1 > µ2

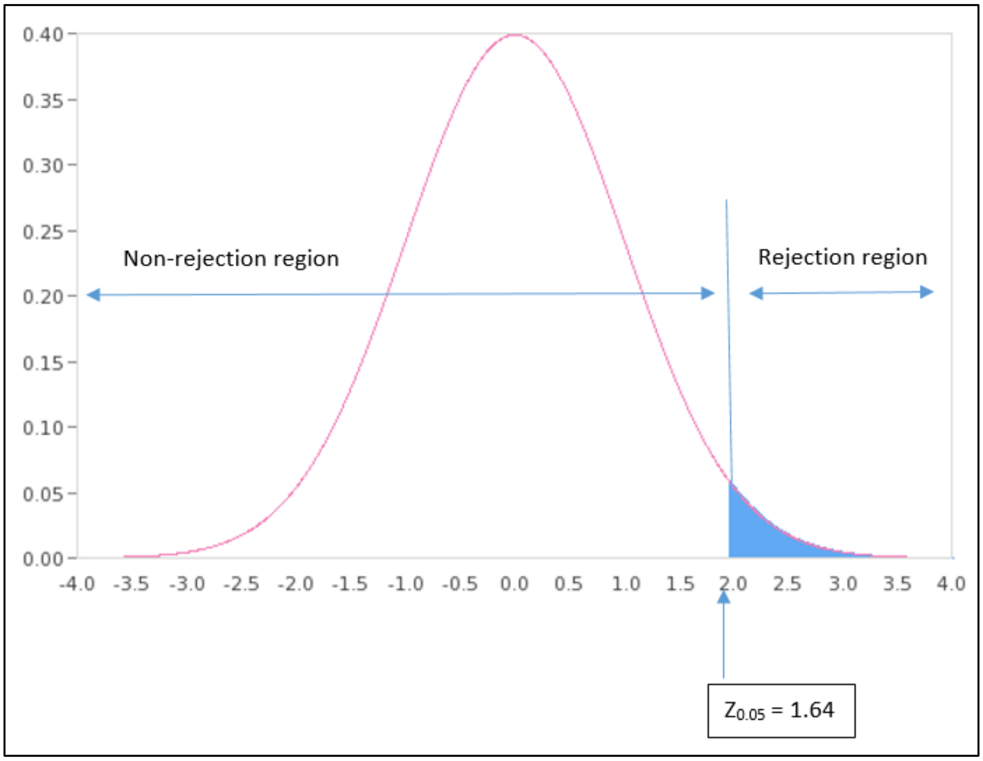
Where, µ1 is mean frequency of male voices and µ2 is mean frequency of female voices
α = 0.05 (95% confidence level)
Critical value *zc* = 1.64 and test statistic z = -18.93

We use the z.test function to compute the p-value for hypothesis test as shown below:



```
> z.test(male$meanfreq,female$meanfreq, alternative = "greater", mu = 0, sigma.x= 0.03,sigma.y= 0.03, conf.level= 0.95)

        Two-sample z-Test

data:  male$meanfreq and female$meanfreq
z = -18.936, p-value = 1
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 -0.02193966          NA
sample estimates:
mean of x mean of y
0.1708135 0.1909997

> |
```

Since, the test statistic falls in non-rejection region and p-value > α (0.05), we fail to reject the $\boldsymbol{H_0}$ at 95% confidence interval. And thus, we can say that there is not enough evidence to support the fact that mean frequency of the male voices is higher than the mean frequency of female voices. The bell curve below represents our hypothesis test:

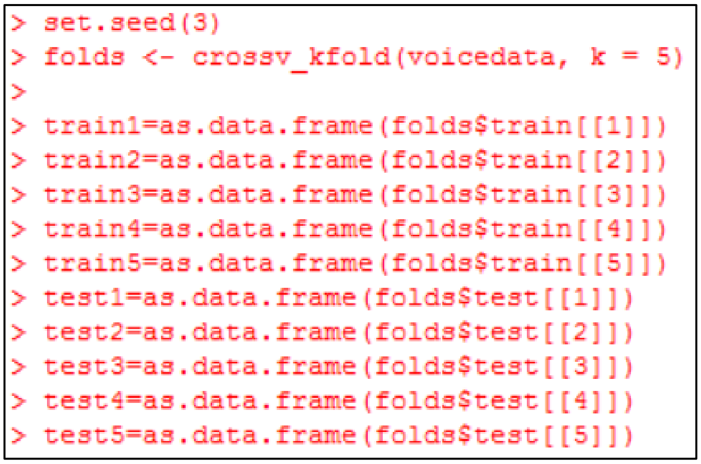


5. Evaluations and Results

5.1. Evaluation Methods

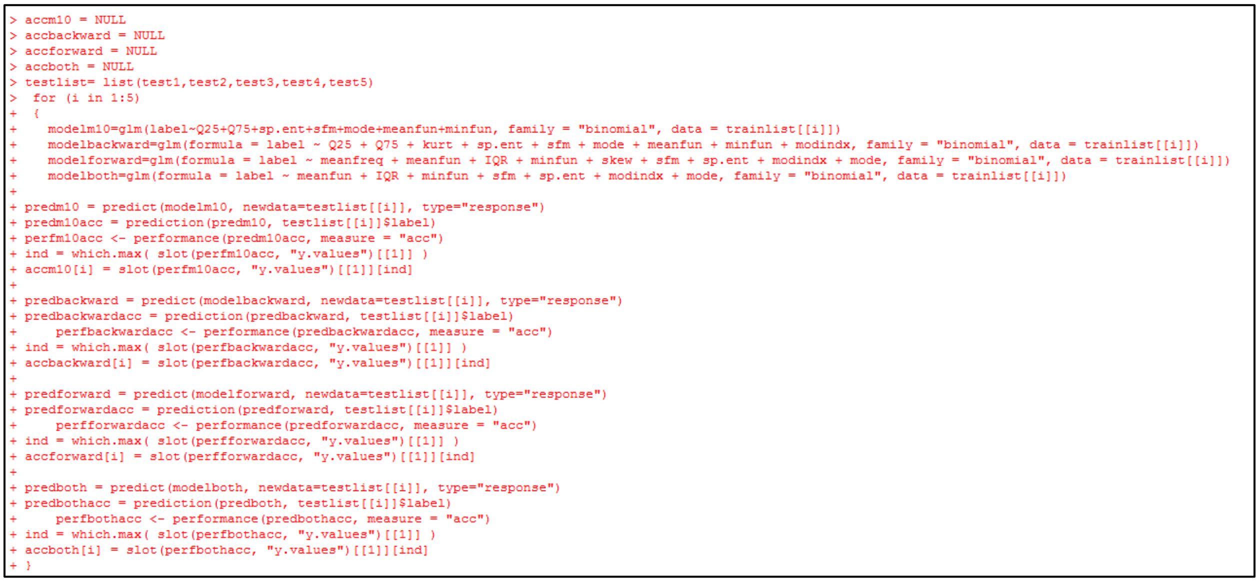
1. Evaluating the models in solutions 1

   a. Splitting the data using n-fold cross validation
   We split the dataset using the n-fold cross validation technique with n=5 to obtain the training and testing datasets:



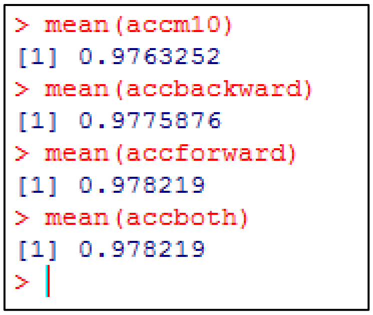
```
> set.seed(3)
> folds <- crossv_kfold(voicedata, k = 5)
>
> train1=as.data.frame(folds$train[[1]])
> train2=as.data.frame(folds$train[[2]])
> train3=as.data.frame(folds$train[[3]])
> train4=as.data.frame(folds$train[[4]])
> train5=as.data.frame(folds$train[[5]])
> test1=as.data.frame(folds$test[[1]])
> test2=as.data.frame(folds$test[[2]])
> test3=as.data.frame(folds$test[[3]])
> test4=as.data.frame(folds$test[[4]])
> test5=as.data.frame(folds$test[[5]])
```

   b. Evaluation based on Accuracy
   Here we have used for loop logic to compute the accuracy of all the four models on all the test data sets obtained from the n-fold cross validation, as shown below:



```
> accm10 = NULL
> accbackward = NULL
> accforward = NULL
> accboth = NULL
> testlist= list(test1,test2,test3,test4,test5)
>  for (i in 1:5)
+  {
+    modelm10=glm(label~Q25+Q75+sp.ent+sfm+mode+meanfun+minfun, family = "binomial", data = trainlist[[i]])
+    modelbackward=glm(formula = label ~ Q25 + Q75 + kurt + sp.ent + sfm + mode + meanfun + minfun + modindx, family = "binomial", data = trainlist[[i]])
+    modelforward=glm(formula = label ~ meanfreq + meanfun + IQR + minfun + skew + sfm + sp.ent + modindx + mode, family = "binomial", data = trainlist[[i]])
+    modelboth=glm(formula = label ~ meanfun + IQR + minfun + sfm + sp.ent + modindx + mode, family = "binomial", data = trainlist[[i]])
+
+ predm10 = predict(modelm10, newdata=testlist[[i]], type="response")
+ predm10acc = prediction(predm10, testlist[[i]]$label)
+ perfm10acc <- performance(predm10acc, measure = "acc")
+ ind = which.max( slot(perfm10acc, "y.values")[[1]] )
+ accm10[i] = slot(perfm10acc, "y.values")[[1]][ind]
+
+ predbackward = predict(modelbackward, newdata=testlist[[i]], type="response")
+ predbackwardacc = prediction(predbackward, testlist[[i]]$label)
+     perfbackwardacc <- performance(predbackwardacc, measure = "acc")
+ ind = which.max( slot(perfbackwardacc, "y.values")[[1]] )
+ accbackward[i] = slot(perfbackwardacc, "y.values")[[1]][ind]
+
+ predforward = predict(modelforward, newdata=testlist[[i]], type="response")
+ predforwardacc = prediction(predforward, testlist[[i]]$label)
+     perfforwardacc <- performance(predforwardacc, measure = "acc")
+ ind = which.max( slot(perfforwardacc, "y.values")[[1]] )
+ accforward[i] = slot(perfforwardacc, "y.values")[[1]][ind]
+
+ predboth = predict(modelboth, newdata=testlist[[i]], type="response")
+ predbothacc = prediction(predboth, testlist[[i]]$label)
+     perfbothacc <- performance(predbothacc, measure = "acc")
+ ind = which.max( slot(perfbothacc, "y.values")[[1]] )
+ accboth[i] = slot(perfbothacc, "y.values")[[1]][ind]
+ }
```

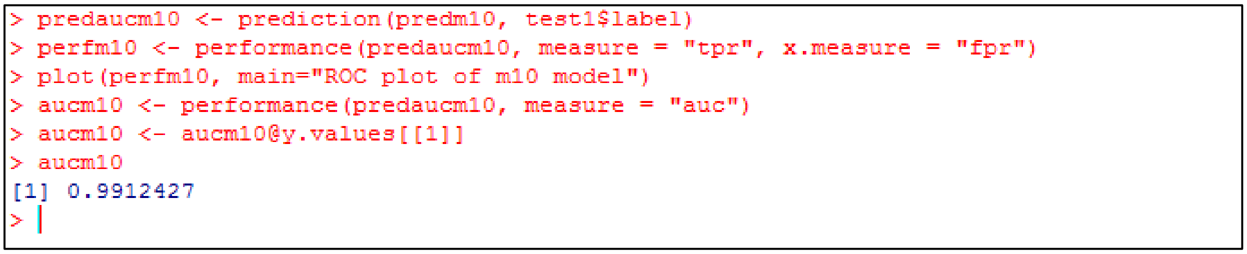
Output of the above for loop is:



```
> mean(accm10)
[1] 0.9763252
> mean(accbackward)
[1] 0.9775876
> mean(accforward)
[1] 0.978219
> mean(accboth)
[1] 0.978219
> 
```

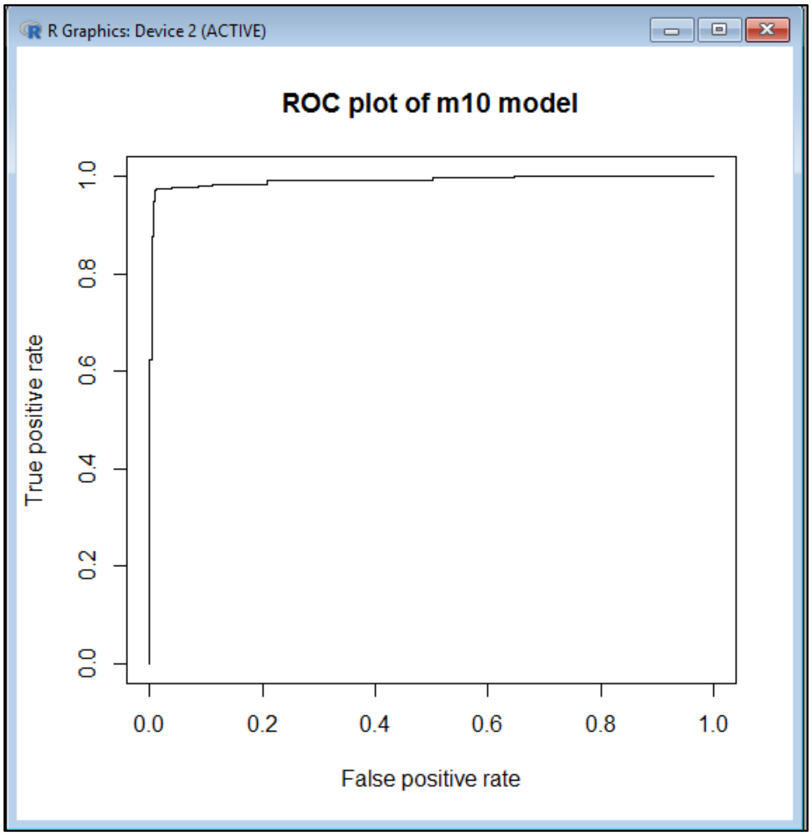
c. Evaluation based on the Area Under Curve (AUC)

i. AUC for the m10 model

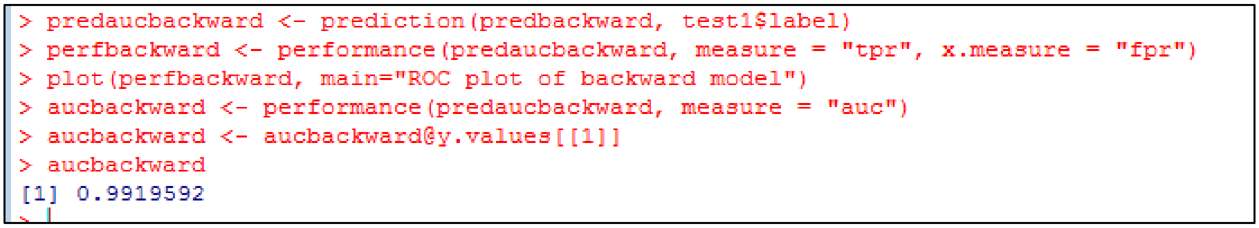


```
> predaucm10 <- prediction(predm10, test1$label)
> perfm10 <- performance(predaucm10, measure = "tpr", x.measure = "fpr")
> plot(perfm10, main="ROC plot of m10 model")
> aucm10 <- performance(predaucm10, measure = "auc")
> aucm10 <- aucm10@y.values[[1]]
> aucm10
[1] 0.9912427
> 
```

The ROC plot of the model m10 is shown below:

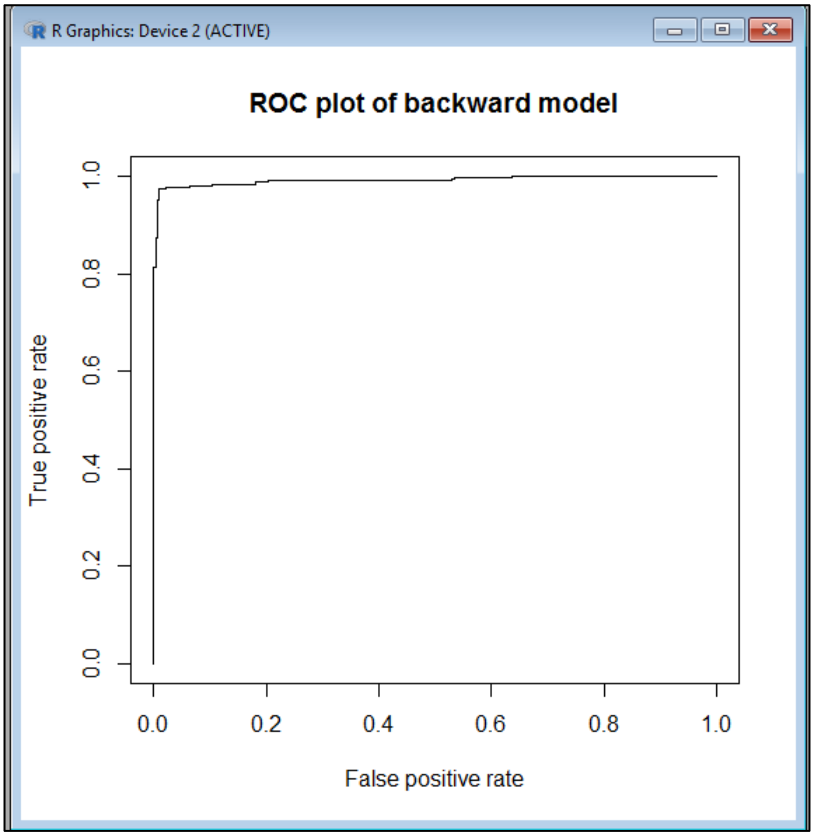


ii. AUC for the backward model

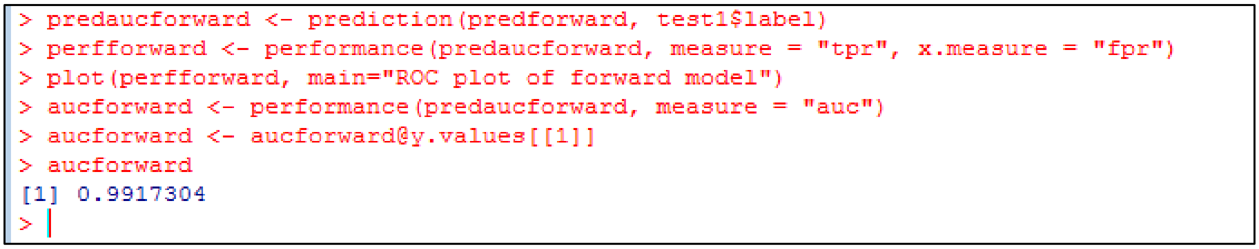


```
> predaucbackward <- prediction(predbackward, test1$label)
> perfbackward <- performance(predaucbackward, measure = "tpr", x.measure = "fpr")
> plot(perfbackward, main="ROC plot of backward model")
> aucbackward <- performance(predaucbackward, measure = "auc")
> aucbackward <- aucbackward@y.values[[1]]
> aucbackward
[1] 0.9919592
>
```

The ROC plot of the model backward is shown below:

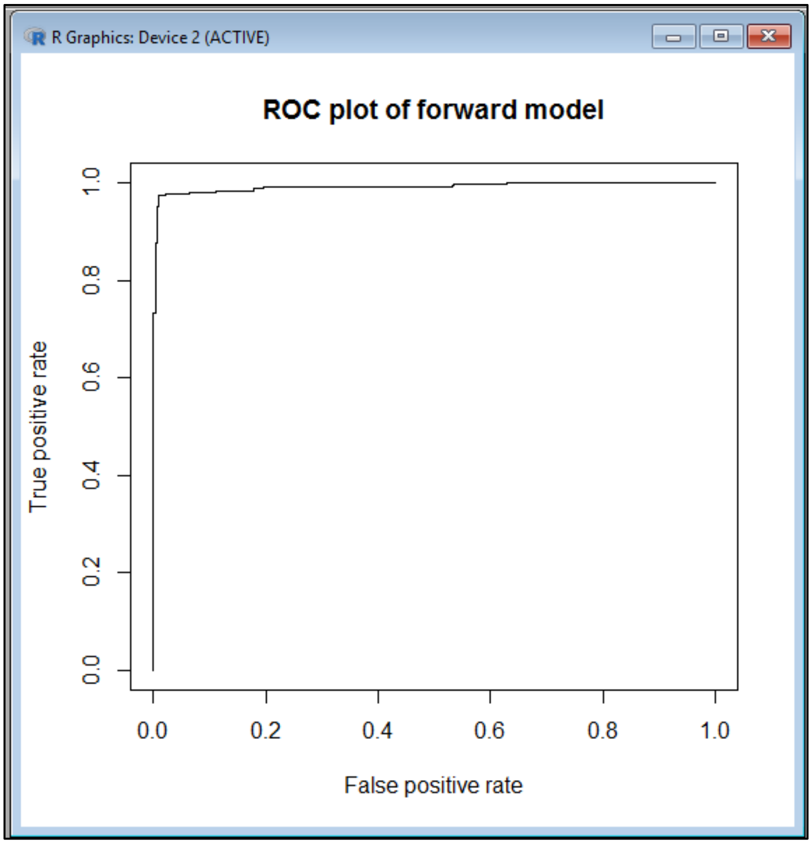


iii. AUC for the forward model

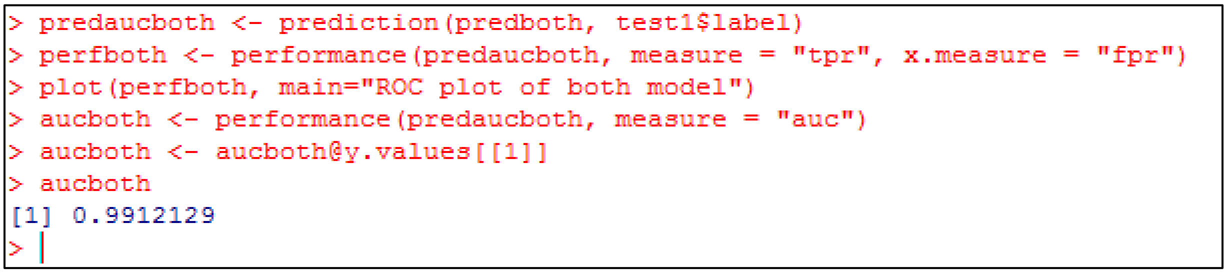


```
> predaucforward <- prediction(predforward, test1$label)
> perfforward <- performance(predaucforward, measure = "tpr", x.measure = "fpr")
> plot(perfforward, main="ROC plot of forward model")
> aucforward <- performance(predaucforward, measure = "auc")
> aucforward <- aucforward@y.values[[1]]
> aucforward
[1] 0.9917304
> 
```

The ROC plot of the model forward is shown below:

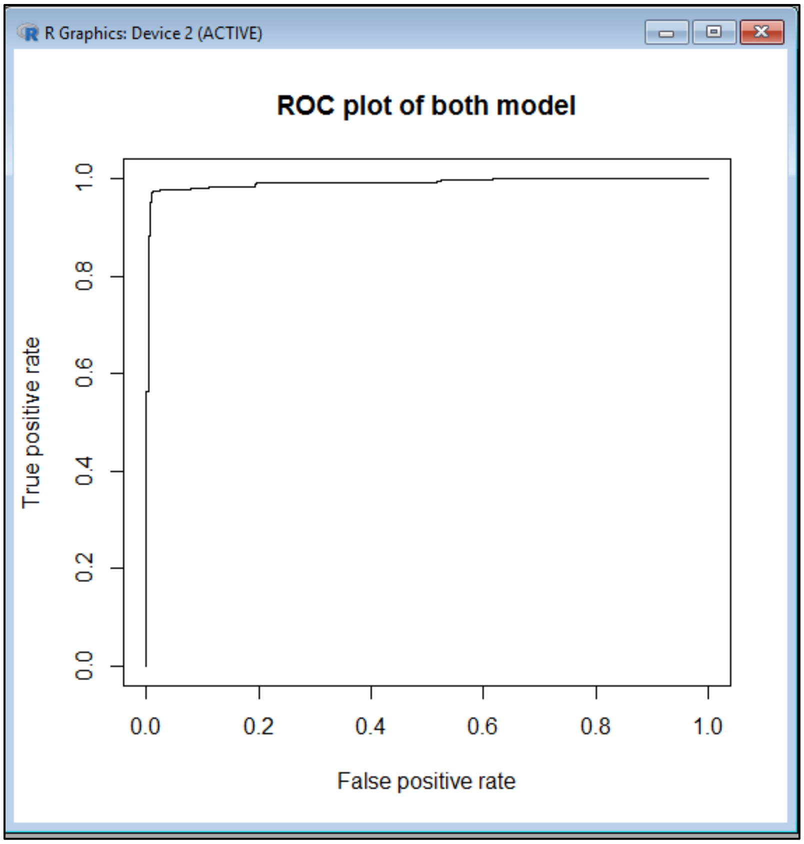


iv. AUC for the both model



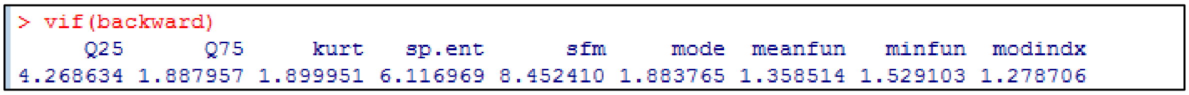
```
> predaucboth <- prediction(predboth, test1$label)
> perfboth <- performance(predaucboth, measure = "tpr", x.measure = "fpr")
> plot(perfboth, main="ROC plot of both model")
> aucboth <- performance(predaucboth, measure = "auc")
> aucboth <- aucboth@y.values[[1]]
> aucboth
[1] 0.9912129
> 
```

The ROC plot of the model both is shown below:

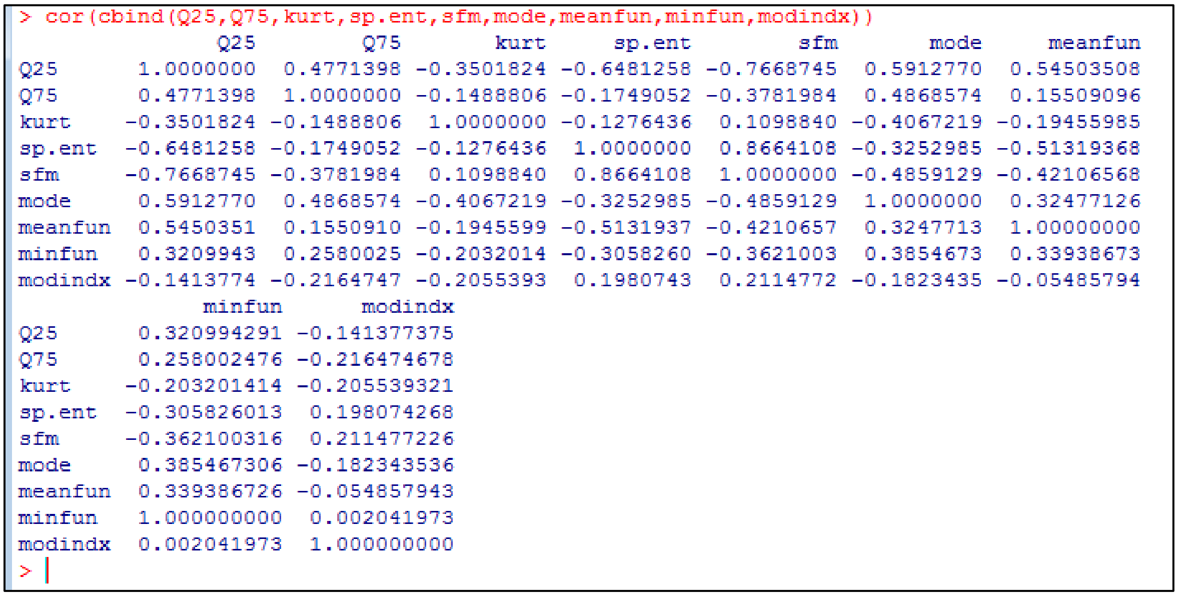


d. Optimizing the backward model

i. Remove multi-collinearity problem



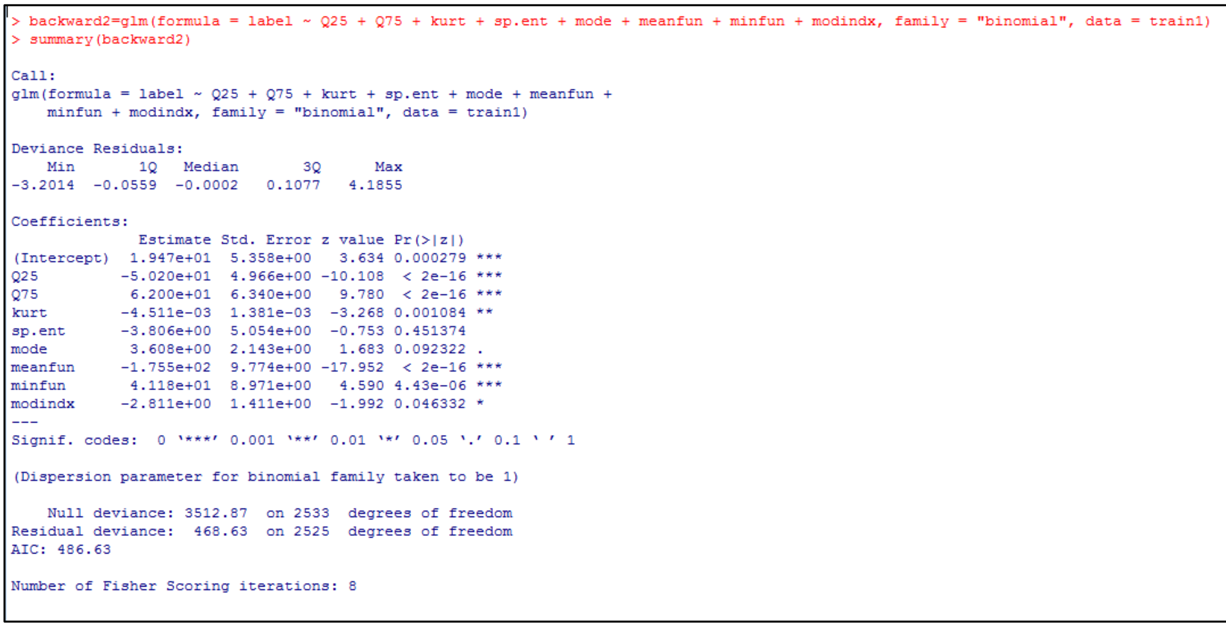
```
> vif(backward)
     Q25      Q75     kurt   sp.ent      sfm     mode  meanfun   minfun  modindx 
4.268634 1.887957 1.899951 6.116969 8.452410 1.883765 1.358514 1.529103 1.278706 
```



```
> cor(cbind(Q25,Q75,kurt,sp.ent,sfm,mode,meanfun,minfun,modindx))
               Q25        Q75       kurt     sp.ent        sfm       mode     meanfun
Q25      1.0000000  0.4771398 -0.3501824 -0.6481258 -0.7668745  0.5912770  0.54503508
Q75      0.4771398  1.0000000 -0.1488806 -0.1749052 -0.3781984  0.4868574  0.15509096
kurt    -0.3501824 -0.1488806  1.0000000 -0.1276436  0.1098840 -0.4067219 -0.19455985
sp.ent  -0.6481258 -0.1749052 -0.1276436  1.0000000  0.8664108 -0.3252985 -0.51319368
sfm     -0.7668745 -0.3781984  0.1098840  0.8664108  1.0000000 -0.4859129 -0.42106568
mode     0.5912770  0.4868574 -0.4067219 -0.3252985 -0.4859129  1.0000000  0.32477126
meanfun  0.5450351  0.1550910 -0.1945599 -0.5131937 -0.4210657  0.3247713  1.00000000
minfun   0.3209943  0.2580025 -0.2032014 -0.3058260 -0.3621003  0.3854673  0.33938673
modindx -0.1413774 -0.2164747 -0.2055393  0.1980743  0.2114772 -0.1823435 -0.05485794
              minfun      modindx
Q25      0.320994291 -0.141377375
Q75      0.258002476 -0.216474678
kurt    -0.203201414 -0.205539321
sp.ent  -0.305826013  0.198074268
sfm     -0.362100316  0.211477226
mode     0.385467306 -0.182343536
meanfun  0.339386726 -0.054857943
minfun   1.000000000  0.002041973
modindx  0.002041973  1.000000000
> |
```

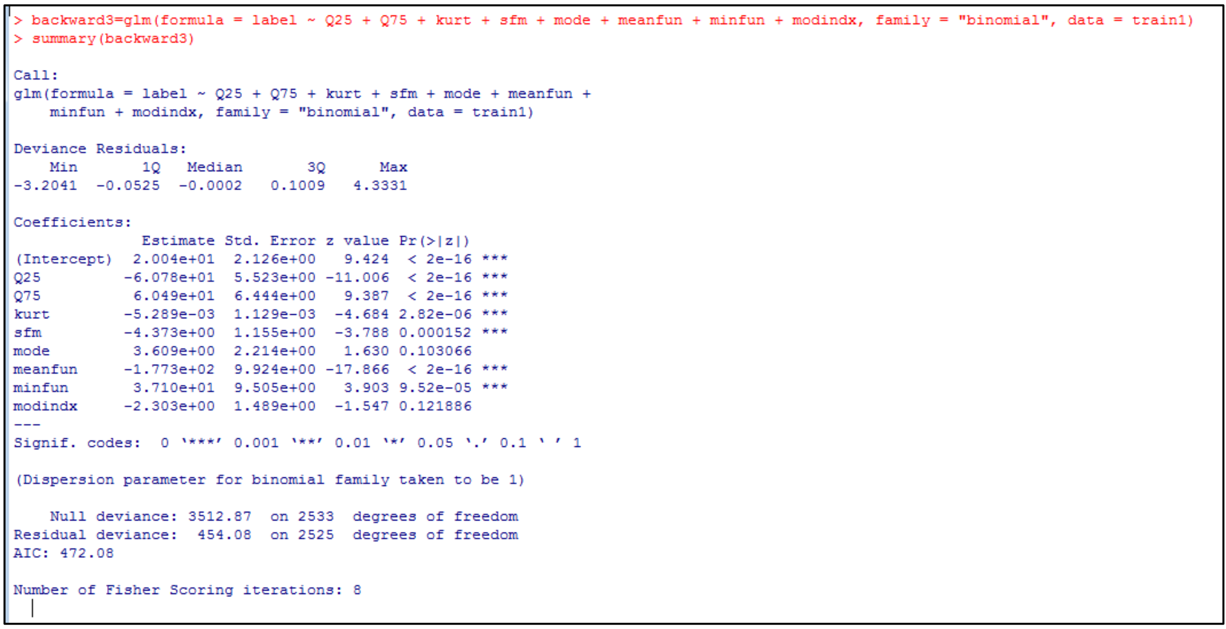
Thus, we can see that there exists multi-collinearity problem because of strong correlation between sfm and sp.ent variables. Hence, we remove any one of them and rebuild the model. Shown below are the two new versions of the model backward:

Backward2: backward model without sfm variable



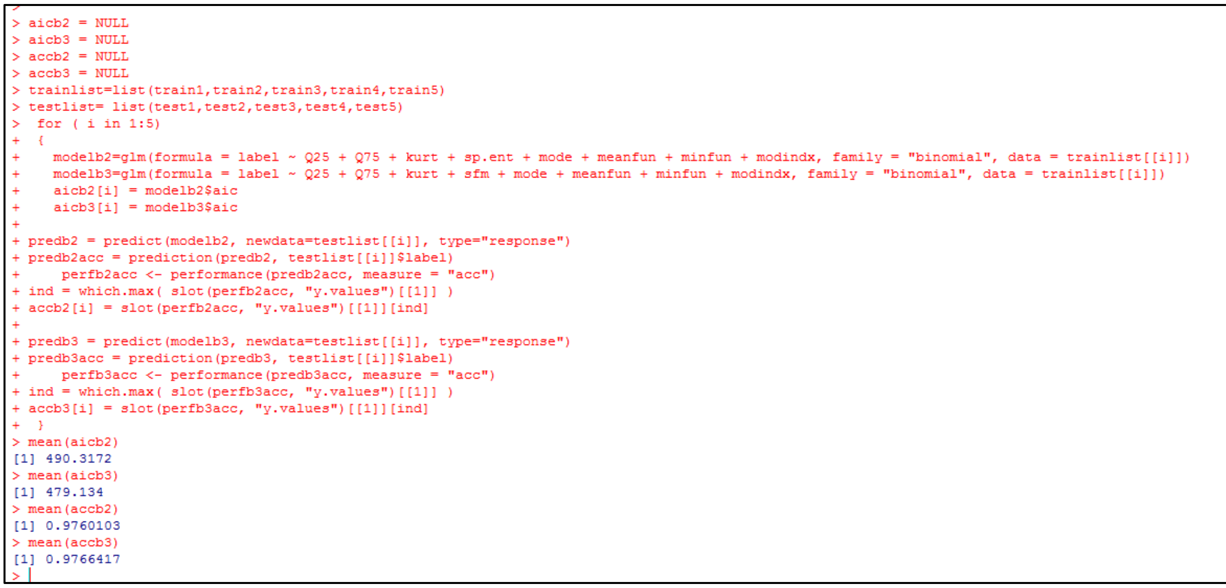
```
> backward2=glm(formula = label ~ Q25 + Q75 + kurt + sp.ent + mode + meanfun + minfun + modindx, family = "binomial", data = train1)
> summary(backward2)

Call:
glm(formula = label ~ Q25 + Q75 + kurt + sp.ent + mode + meanfun +
    minfun + modindx, family = "binomial", data = train1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2014  -0.0559  -0.0002   0.1077   4.1855

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.947e+01  5.358e+00   3.634 0.000279 ***
Q25         -5.020e+01  4.966e+00 -10.108  < 2e-16 ***
Q75          6.200e+01  6.340e+00   9.780  < 2e-16 ***
kurt        -4.511e-03  1.381e-03  -3.268 0.001084 **
sp.ent      -3.806e+00  5.054e+00  -0.753 0.451374
mode         3.608e+00  2.143e+00   1.683 0.092322 .
meanfun     -1.755e+02  9.774e+00 -17.952  < 2e-16 ***
minfun       4.118e+01  8.971e+00   4.590 4.43e-06 ***
modindx     -2.811e+00  1.411e+00  -1.992 0.046332 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  468.63  on 2525  degrees of freedom
AIC: 486.63

Number of Fisher Scoring iterations: 8
```

Backward3: backward model without sp.ent variable



```
> backward3=glm(formula = label ~ Q25 + Q75 + kurt + sfm + mode + meanfun + minfun + modindx, family = "binomial", data = train1)
> summary(backward3)

Call:
glm(formula = label ~ Q25 + Q75 + kurt + sfm + mode + meanfun +
    minfun + modindx, family = "binomial", data = train1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2041  -0.0525  -0.0002   0.1009   4.3331

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.004e+01  2.126e+00   9.424  < 2e-16 ***
Q25         -6.078e+01  5.523e+00 -11.006  < 2e-16 ***
Q75          6.049e+01  6.444e+00   9.387  < 2e-16 ***
kurt        -5.289e-03  1.129e-03  -4.684 2.82e-06 ***
sfm         -4.373e+00  1.155e+00  -3.788 0.000152 ***
mode         3.609e+00  2.214e+00   1.630 0.103066
meanfun     -1.773e+02  9.924e+00 -17.866  < 2e-16 ***
minfun       3.710e+01  9.505e+00   3.903 9.52e-05 ***
modindx     -2.303e+00  1.489e+00  -1.547 0.121886
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3512.87  on 2533  degrees of freedom
Residual deviance:  454.08  on 2525  degrees of freedom
AIC: 472.08

Number of Fisher Scoring iterations: 8
```

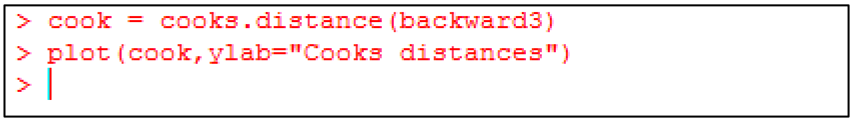
Computing AIC and Accuracy of the above two models:



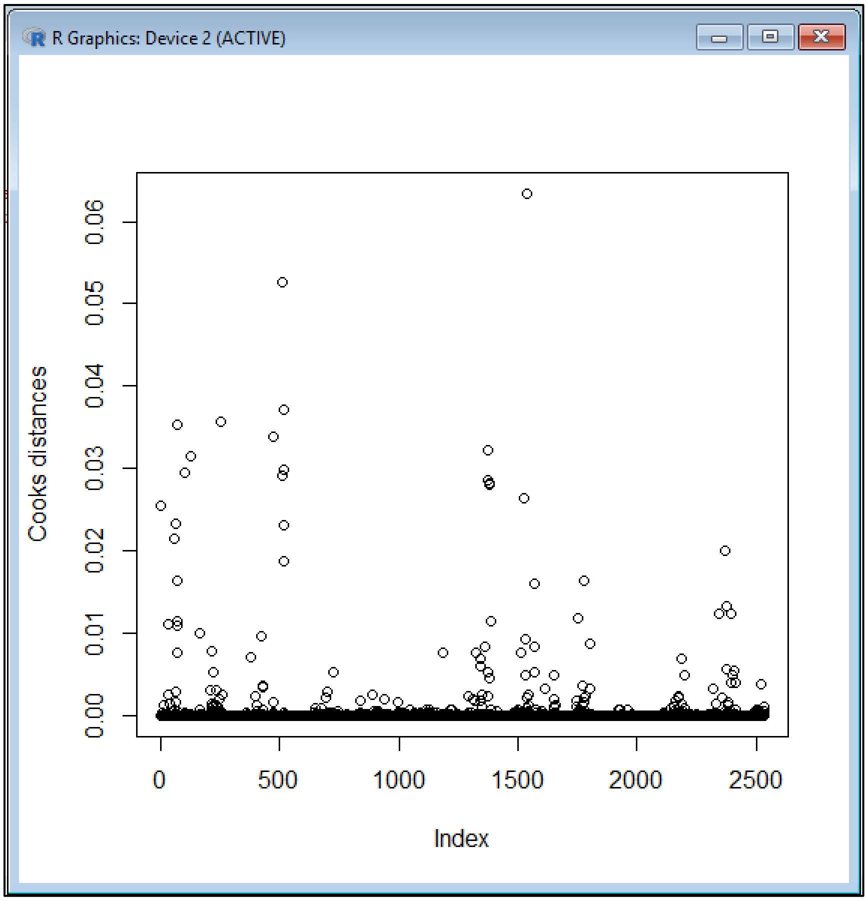
```
> aicb2 = NULL
> aicb3 = NULL
> accb2 = NULL
> accb3 = NULL
> trainlist=list(train1,train2,train3,train4,train5)
> testlist= list(test1,test2,test3,test4,test5)
>  for ( i in 1:5)
+  {
+    modelb2=glm(formula = label ~ Q25 + Q75 + kurt + sp.ent + mode + meanfun + minfun + modindx, family = "binomial", data = trainlist[[i]])
+    modelb3=glm(formula = label ~ Q25 + Q75 + kurt + sfm + mode + meanfun + minfun + modindx, family = "binomial", data = trainlist[[i]])
+    aicb2[i] = modelb2$aic
+    aicb3[i] = modelb3$aic
+
+ predb2 = predict(modelb2, newdata=testlist[[i]], type="response")
+ predb2acc = prediction(predb2, testlist[[i]]$label)
+     perfb2acc <- performance(predb2acc, measure = "acc")
+ ind = which.max( slot(perfb2acc, "y.values")[[1]] )
+ accb2[i] = slot(perfb2acc, "y.values")[[1]][ind]
+
+ predb3 = predict(modelb3, newdata=testlist[[i]], type="response")
+ predb3acc = prediction(predb3, testlist[[i]]$label)
+     perfb3acc <- performance(predb3acc, measure = "acc")
+ ind = which.max( slot(perfb3acc, "y.values")[[1]] )
+ accb3[i] = slot(perfb3acc, "y.values")[[1]][ind]
+  }
> mean(aicb2)
[1] 490.3172
> mean(aicb3)
[1] 479.134
> mean(accb2)
[1] 0.9760103
> mean(accb3)
[1] 0.9766417
> 
```

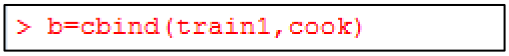
From the above screen shot, we can see that the AIC of Backward3 is low and its accuracy is higher. Therefore, we select Backward3 as our model for further analysis.

ii. Remove the influential points
Here we measured the influential points in the dataset using the Cooks distance and plotted its graph for visualization:



```
> cook = cooks.distance(backward3)
> plot(cook,ylab="Cooks distances")
> 
```

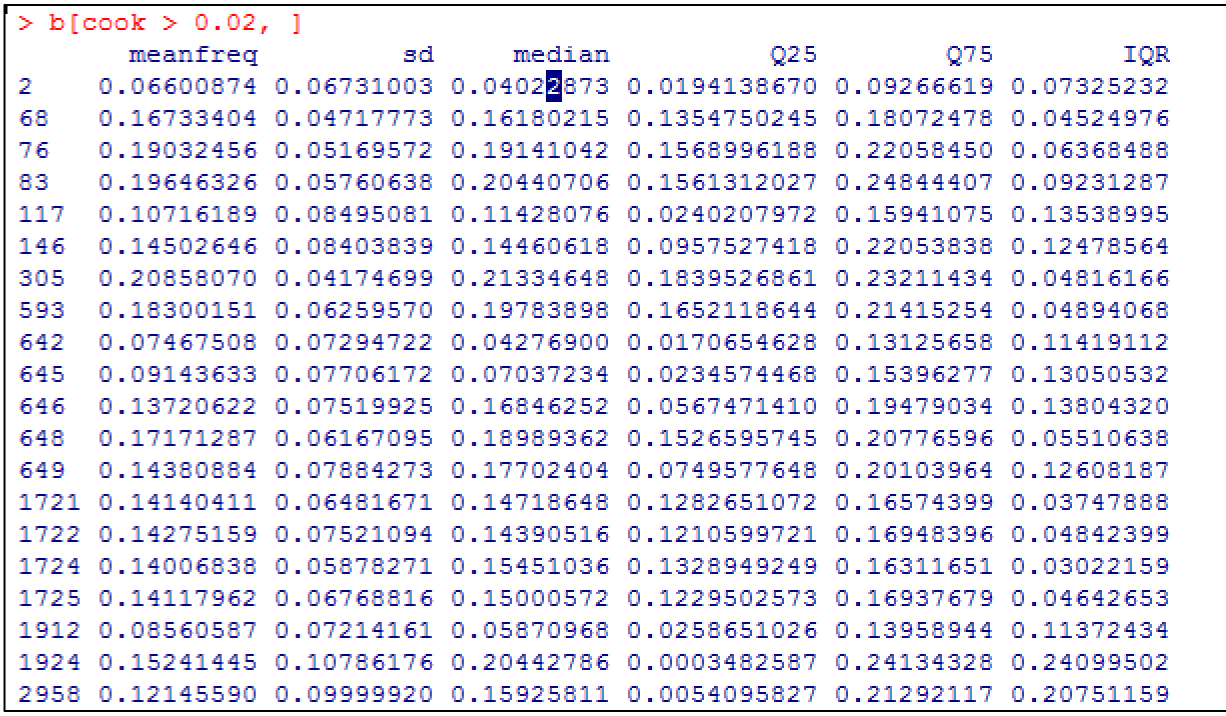




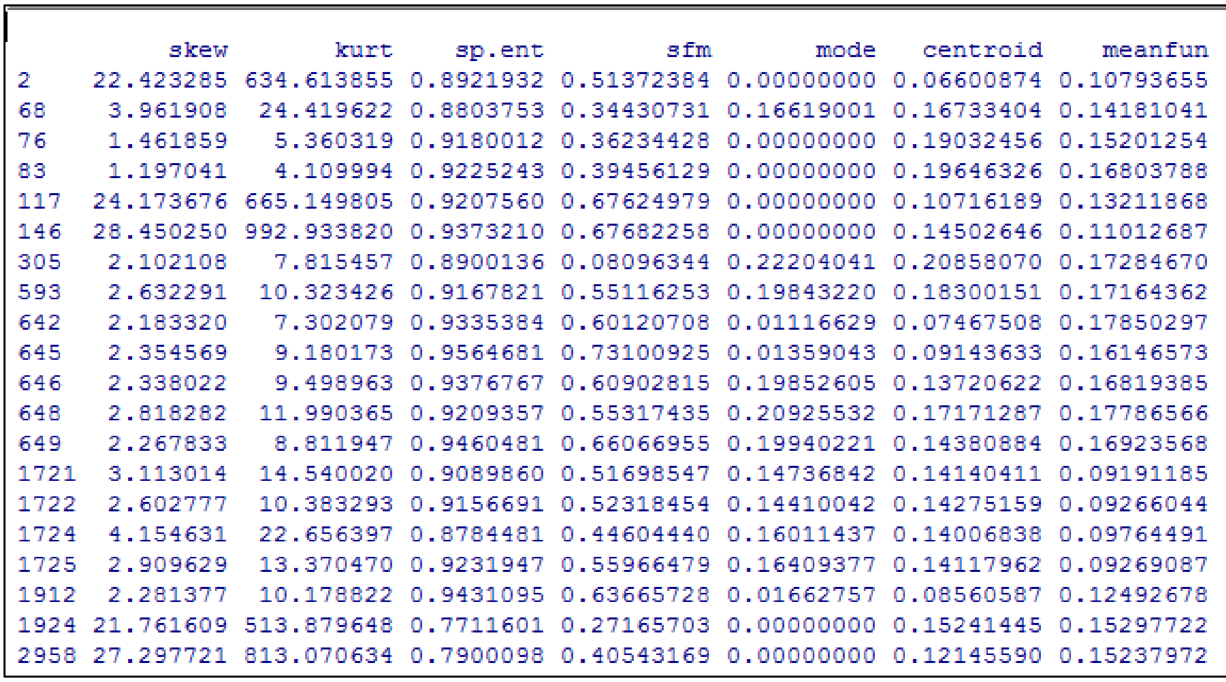
```
> b=cbind(train1,cook)
```

We took threshold values as 0.02.

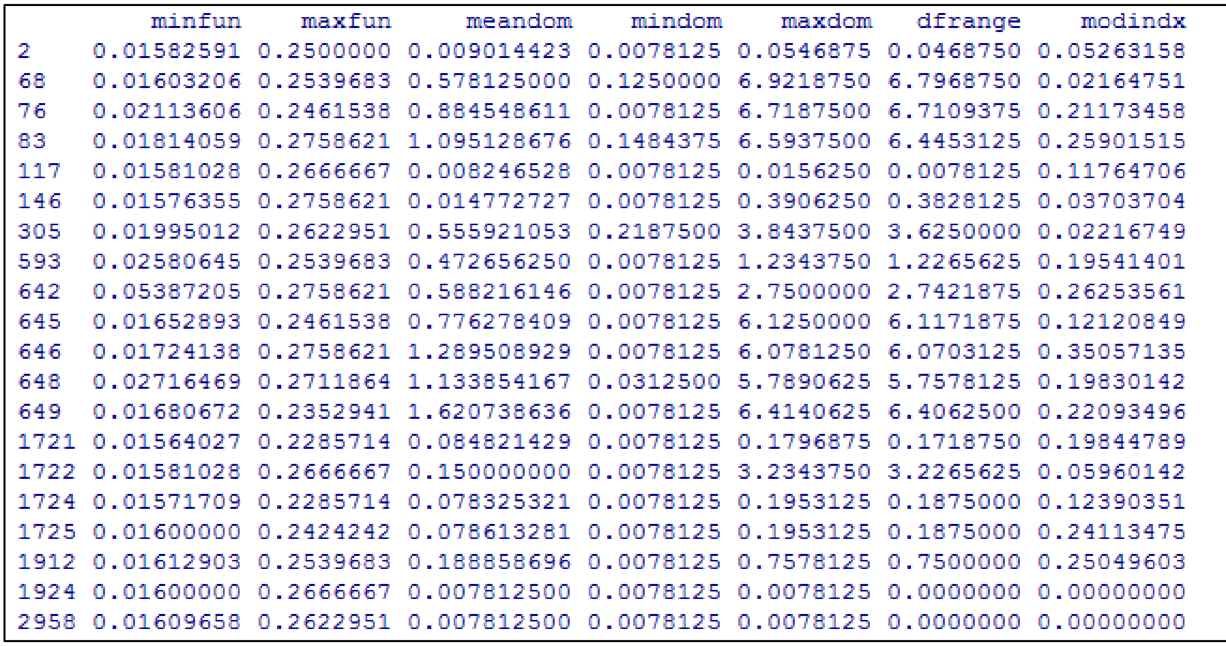
Printing the influential points from the dataset:



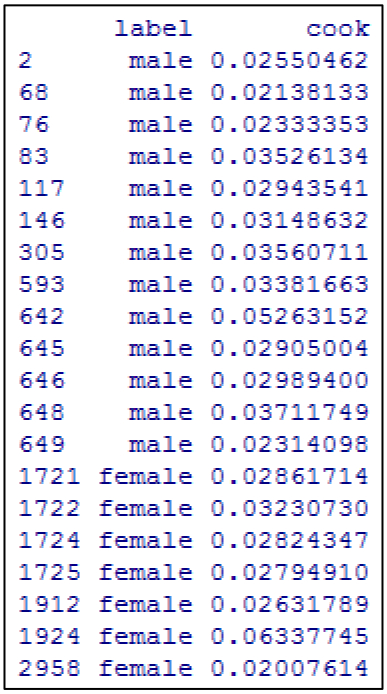
```
> b[cook > 0.02, ]
       meanfreq         sd     median          Q25        Q75        IQR
2    0.06600874 0.06731003 0.04022873 0.0194138670 0.09266619 0.07325232
68   0.16733404 0.04717773 0.16180215 0.1354750245 0.18072478 0.04524976
76   0.19032456 0.05169572 0.19141042 0.1568996188 0.22058450 0.06368488
83   0.19646326 0.05760638 0.20440706 0.1561312027 0.24844407 0.09231287
117  0.10716189 0.08495081 0.11428076 0.0240207972 0.15941075 0.13538995
146  0.14502646 0.08403839 0.14460618 0.0957527418 0.22053838 0.12478564
305  0.20858070 0.04174699 0.21334648 0.1839526861 0.23211434 0.04816166
593  0.18300151 0.06259570 0.19783898 0.1652118644 0.21415254 0.04894068
642  0.07467508 0.07294722 0.04276900 0.0170654628 0.13125658 0.11419112
645  0.09143633 0.07706172 0.07037234 0.0234574468 0.15396277 0.13050532
646  0.13720622 0.07519925 0.16846252 0.0567471410 0.19479034 0.13804320
648  0.17171287 0.06167095 0.18989362 0.1526595745 0.20776596 0.05510638
649  0.14380884 0.07884273 0.17702404 0.0749577648 0.20103964 0.12608187
1721 0.14140411 0.06481671 0.14718648 0.1282651072 0.16574399 0.03747888
1722 0.14275159 0.07521094 0.14390516 0.1210599721 0.16948396 0.04842399
1724 0.14006838 0.05878271 0.15451036 0.1328949249 0.16311651 0.03022159
1725 0.14117962 0.06768816 0.15000572 0.1229502573 0.16937679 0.04642653
1912 0.08560587 0.07214161 0.05870968 0.0258651026 0.13958944 0.11372434
1924 0.15241445 0.10786176 0.20442786 0.0003482587 0.24134328 0.24099502
2958 0.12145590 0.09999920 0.15925811 0.0054095827 0.21292117 0.20751159
```



```
          skew       kurt    sp.ent        sfm       mode   centroid    meanfun
2    22.423285 634.613855 0.8921932 0.51372384 0.00000000 0.06600874 0.10793655
68    3.961908  24.419622 0.8803753 0.34430731 0.16619001 0.16733404 0.14181041
76    1.461859   5.360319 0.9180012 0.36234428 0.00000000 0.19032456 0.15201254
83    1.197041   4.109994 0.9225243 0.39456129 0.00000000 0.19646326 0.16803788
117  24.173676 665.149805 0.9207560 0.67624979 0.00000000 0.10716189 0.13211868
146  28.450250 992.933820 0.9373210 0.67682258 0.00000000 0.14502646 0.11012687
305   2.102108   7.815457 0.8900136 0.08096344 0.22204041 0.20858070 0.17284670
593   2.632291  10.323426 0.9167821 0.55116253 0.19843220 0.18300151 0.17164362
642   2.183320   7.302079 0.9335384 0.60120708 0.01116629 0.07467508 0.17850297
645   2.354569   9.180173 0.9564681 0.73100925 0.01359043 0.09143633 0.16146573
646   2.338022   9.498963 0.9376767 0.60902815 0.19852605 0.13720622 0.16819385
648   2.818282  11.990365 0.9209357 0.55317435 0.20925532 0.17171287 0.17786566
649   2.267833   8.811947 0.9460481 0.66066955 0.19940221 0.14380884 0.16923568
1721  3.113014  14.540020 0.9089860 0.51698547 0.14736842 0.14140411 0.09191185
1722  2.602777  10.383293 0.9156691 0.52318454 0.14410042 0.14275159 0.09266044
1724  4.154631  22.656397 0.8784481 0.44604440 0.16011437 0.14006838 0.09764491
1725  2.909629  13.370470 0.9231947 0.55966479 0.16409377 0.14117962 0.09269087
1912  2.281377  10.178822 0.9431095 0.63665728 0.01662757 0.08560587 0.12492678
1924 21.761609 513.879648 0.7711601 0.27165703 0.00000000 0.15241445 0.15297722
2958 27.297721 813.070634 0.7900098 0.40543169 0.00000000 0.12145590 0.15237972
```

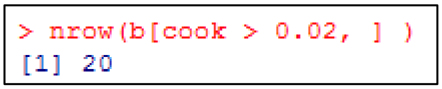


```
        minfun    maxfun     meandom    mindom    maxdom   dfrange    modindx
2    0.01582591 0.2500000 0.009014423 0.0078125 0.0546875 0.0468750 0.05263158
68   0.01603206 0.2539683 0.578125000 0.1250000 6.9218750 6.7968750 0.02164751
76   0.02113606 0.2461538 0.884548611 0.0078125 6.7187500 6.7109375 0.21173458
83   0.01814059 0.2758621 1.095128676 0.1484375 6.5937500 6.4453125 0.25901515
117  0.01581028 0.2666667 0.008246528 0.0078125 0.0156250 0.0078125 0.11764706
146  0.01576355 0.2758621 0.014772727 0.0078125 0.3906250 0.3828125 0.03703704
305  0.01995012 0.2622951 0.555921053 0.2187500 3.8437500 3.6250000 0.02216749
593  0.02580645 0.2539683 0.472656250 0.0078125 1.2343750 1.2265625 0.19541401
642  0.05387205 0.2758621 0.588216146 0.0078125 2.7500000 2.7421875 0.26253561
645  0.01652893 0.2461538 0.776278409 0.0078125 6.1250000 6.1171875 0.12120849
646  0.01724138 0.2758621 1.289508929 0.0078125 6.0781250 6.0703125 0.35057135
648  0.02716469 0.2711864 1.133854167 0.0312500 5.7890625 5.7578125 0.19830142
649  0.01680672 0.2352941 1.620738636 0.0078125 6.4140625 6.4062500 0.22093496
1721 0.01564027 0.2285714 0.084821429 0.0078125 0.1796875 0.1718750 0.19844789
1722 0.01581028 0.2666667 0.150000000 0.0078125 3.2343750 3.2265625 0.05960142
1724 0.01571709 0.2285714 0.078325321 0.0078125 0.1953125 0.1875000 0.12390351
1725 0.01600000 0.2424242 0.078613281 0.0078125 0.1953125 0.1875000 0.24113475
1912 0.01612903 0.2539683 0.188858696 0.0078125 0.7578125 0.7500000 0.25049603
1924 0.01600000 0.2666667 0.007812500 0.0078125 0.0078125 0.0000000 0.00000000
2958 0.01609658 0.2622951 0.007812500 0.0078125 0.0078125 0.0000000 0.00000000
```



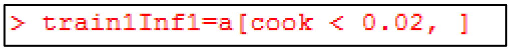
```
      label       cook
2      male 0.02550462
68     male 0.02138133
76     male 0.02333353
83     male 0.03526134
117    male 0.02943541
146    male 0.03148632
305    male 0.03560711
593    male 0.03381663
642    male 0.05263152
645    male 0.02905004
646    male 0.02989400
648    male 0.03711749
649    male 0.02314098
1721 female 0.02861714
1722 female 0.03230730
1724 female 0.02824347
1725 female 0.02794910
1912 female 0.02631789
1924 female 0.06337745
2958 female 0.02007614
```

Printing the count of influential points in the dataset:



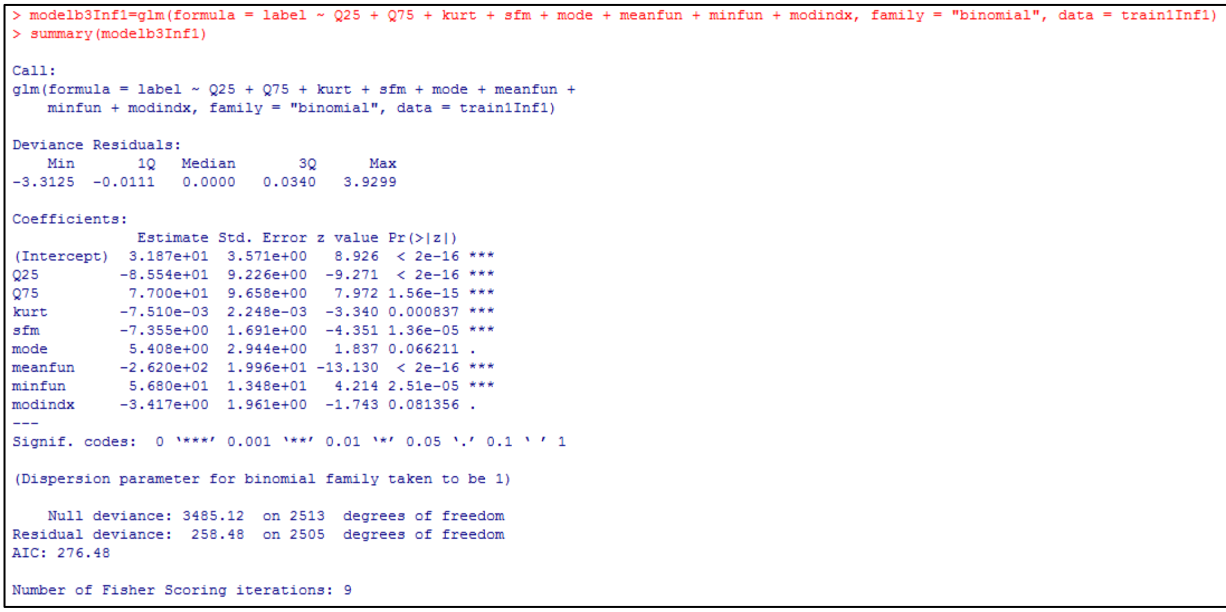
```
> nrow(b[cook > 0.02, ] )
[1] 20
```

Creating dataset by removing influential points from the original dataset:



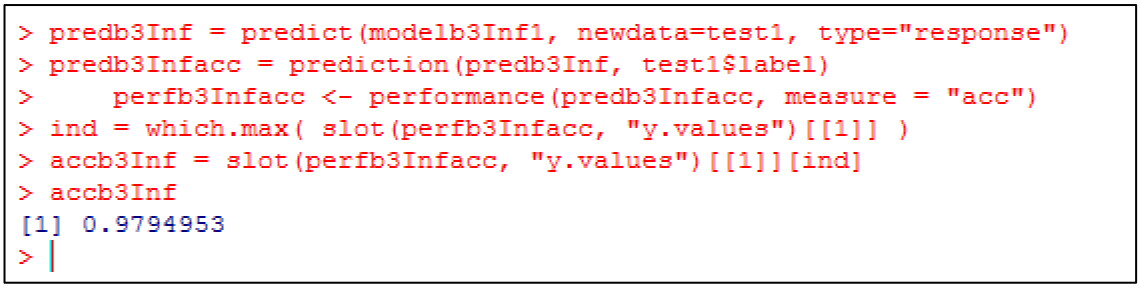
```
> train1Infl=a[cook < 0.02, ]
```

Re-Building the model on the new dataset created above (without influential points):



```
> modelb3Inf1=glm(formula = label ~ Q25 + Q75 + kurt + sfm + mode + meanfun + minfun + modindx, family = "binomial", data = train1Inf1)
> summary(modelb3Inf1)

Call:
glm(formula = label ~ Q25 + Q75 + kurt + sfm + mode + meanfun + 
    minfun + modindx, family = "binomial", data = train1Inf1)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-3.3125  -0.0111   0.0000   0.0340   3.9299  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)  3.187e+01  3.571e+00   8.926  < 2e-16 ***
Q25         -8.554e+01  9.226e+00  -9.271  < 2e-16 ***
Q75          7.700e+01  9.658e+00   7.972 1.56e-15 ***
kurt        -7.510e-03  2.248e-03  -3.340 0.000837 ***
sfm         -7.355e+00  1.691e+00  -4.351 1.36e-05 ***
mode         5.408e+00  2.944e+00   1.837 0.066211 .  
meanfun     -2.620e+02  1.996e+01 -13.130  < 2e-16 ***
minfun       5.680e+01  1.348e+01   4.214 2.51e-05 ***
modindx     -3.417e+00  1.961e+00  -1.743 0.081356 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3485.12  on 2513  degrees of freedom
Residual deviance:  258.48  on 2505  degrees of freedom
AIC: 276.48

Number of Fisher Scoring iterations: 9
```

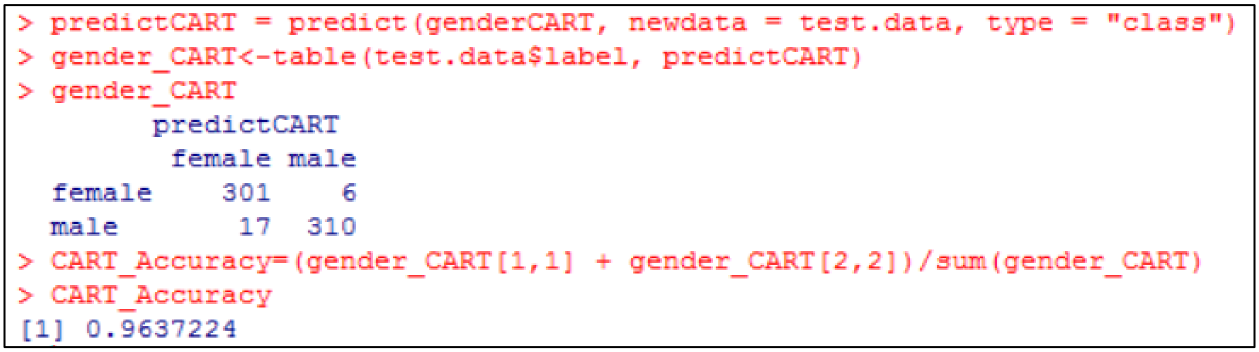
Calculating the Accuracy of the above model modelb3Infl



```
> predb3Inf = predict(modelb3Inf1, newdata=test1, type="response")
> predb3Infacc = prediction(predb3Inf, test1$label)
>     perfb3Infacc <- performance(predb3Infacc, measure = "acc")
> ind = which.max( slot(perfb3Infacc, "y.values")[[1]] )
> accb3Inf = slot(perfb3Infacc, "y.values")[[1]][ind]
> accb3Inf
[1] 0.9794953
> 
```

2. Evaluating the solutions 2: CART model

Predict function and confusion matrix is used to calculate the accuracy of the CART model:



```
> predictCART = predict(genderCART, newdata = test.data, type = "class")
> gender_CART<-table(test.data$label, predictCART)
> gender_CART
        predictCART
         female male
  female    301    6
  male       17  310
> CART_Accuracy=(gender_CART[1,1] + gender_CART[2,2])/sum(gender_CART)
> CART_Accuracy
[1] 0.9637224
```

5.2. Results and Findings

a. Solution 1: Logistic Regression

The initial findings from the model selection and evaluation we get:

	Model m10	Backward	Forward	Both
AIC	472.7254	465.1058	466.3814	469.9538
Mc Fadden R^2	0.87000082	0.87330808	0.87294498	0.87078972
Accuracy	0.9763252	0.9775876	0.978219	0.978219
AUC	99.12427	99.19592	99.17304	99.12129

From the above table, we can see that model backward is having low AIC value, high McFadden R2, high accuracy and high AUC. Hence, we selected Backward model as the best model from our initial analysis. Further, we examined this model for multi-collinearity and built two models. The findings for these two new models backward2 and backward3 is shown below:

	Backward2	Backward3
AIC	490.31	479.13
Accuracy	97.60	97.66

As shown from the above table, we can say that the Backward3 model is the best model after removing the multi-collinearity problem. Next we examined the model for influential points and optimized the model by removing the them from our data set. The findings for this new model is shown below:

AIC of Backward3 w/o influential points: 276.48
Accuracy of Backward3 w/o influential points: 97.94%

<u>Equation of our final logistic regression model after optimization is as below:</u>

Log(Odds) = 31.87 - 85.54(Q25) + 77(Q75) - 0.00751(kurt) - 7.355(sfm) + 5.408(mode) - 262(meanfun) + 56.8(minfun) - 3.417(modindx) + e

Odds= $$\frac{p}{1-p} = \frac{P(Y=1)}{P(Y=0)}$$

Measure the odds that event Y = male

<u>Interpreting the coefficients of equation</u>

Coefficient of Q25 is -85.54

Interpretation: Assuming all other x variables constant, for every single unit of increase in Q25, log (p/(1-p)) would decrease by 85.54 where p = Pr(Y=male).

Coefficient of Q75 is 77
Interpretation: Assuming all other x variables constant, for every single unit of increase in Q75, log (p/(1-p)) would increase by 77 where p = Pr(Y=male).

Coefficient of kurt is -0.00751
Interpretation: Assuming all other x variables constant, for every single unit of increase in kurt, log (p/(1-p)) would decrease by 0.00751 where p = Pr(Y=male).

Coefficient of sfm is -7.355
Interpretation: Assuming all other x variables constant, for every single unit of increase in sfm, log (p/(1-p)) would decrease by 7.355 where p = Pr(Y=male).

Coefficient of mode is 5.408
Interpretation: Assuming all other x variables constant, for every single unit of increase in mode, log (p/(1-p)) would increase by 5.408 where p = Pr(Y=male).

Coefficient of meanfun is -262
Interpretation: Assuming all other x variables constant, for every single unit of increase in meanfun, log (p/(1-p)) would decrease by 262 where p = Pr(Y=male).

Coefficient of minfun is 56.8
Interpretation: Assuming all other x variables constant, for every single unit of increase in minfun, log (p/(1-p)) would increase by 56.8 where p = Pr(Y=male).

Coefficient of modindx is -3.417
Interpretation: Assuming all other x variables constant, for every single unit of increase in modindx, log (p/(1-p)) would decrease by 3.417 where p = Pr(Y=male).

b. CART model: Results and findings
The accuracy of the CART model is 96.37% using the predict function.

c. Hypothesis testing: Results and findings
From the hypothesis test performed earlier, we found that test statistic falls in non-rejection region and thus, we failed to reject the H_0 at 95% confidence interval. In other words, our hypothesis that mean frequency of male voice is higher than mean frequency of female voices was proved to be wrong.

6. Conclusions and Future Work

6.1. Conclusions

Thus, we build two predictive models: 1) Logistic regression model and 2) CART model, using which we can predict the gender of a person based on acoustic properties of his/her voice. Moreover, based on CART model analysis, we can conclude that mean fundamental frequency is the most powerful deciding factor for gender recognition. Finally, based on hypothesis testing we concluded that mean frequency of male voices is not higher than mean frequency of female voices.

6.2. Limitations

One limitation in our project is that data set is too small. It contains only 3168 records which are only a small range of real-world voices. Hence, we cannot simply rely on our models build and conclusions made. In fact, in real-world there may be wide range of voices wherein male voices are having higher frequency than female voices. A larger data set and variety of voice samples may also increase the accuracy of the models.

6.3. Potential Improvements or Future Work

For potential improvement, we can collect broad range of real-world voice samples and build predictive models on large amount of data set. Moreover, apart from building Logistic regression model and CART model, we can apply several other machine learning techniques such as Random Forest, Support Vector Machine (SVM) and XGBoost for classification. We can build and evaluate models using afore-mentioned methods and try to achieve even more accuracy in gender identification with lowest possible misclassification error rate.