ECG Heartbeat Diagnosis

Nayan Kaushal

3/9/2022

```r
#loading relevant libraries
library(MASS)
library(ggplot2)
library(dplyr)
```

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

## The following object is masked from 'package:MASS':
##
##     select

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

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

```r
library(class)
library(dummies)
```

```
## dummies-1.5.6 provided by Decision Patterns
```

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.6     v purrr   0.3.4
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::select() masks MASS::select()
```

```
library(glmnet)
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

## Loaded glmnet 4.1-3
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.1.2

## randomForest 4.7-1

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':
##
##     combine

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
library(tree)
```

```
## Registered S3 method overwritten by 'tree':
##   method     from
##   print.tree cli
```

#Overview:

The dataset I will be analyzing consists of Electrocardiogram (ECG) signals of single heartbeats, derived from The PTB Diagnostic ECG Database, and associated labels describing whether the heartbeat is normal or abnormal.

The dataset consists of 187 vectors in the dataset describing the th heartbeat ECG signal. This is a set of 187 measurements at consecutive time points (in the preprocessing step the signals have been cropped, downsampled to sampling frequency 125Hz, and padded with zeroes if necessary). As common in imaging settings, we can think of these measurements as 187 'variables'. The response variable is a categorical variable indicating whether the heartbeat is normal or abnormal (0: normal, 1 abnormal).

Aim:

To train a classification model that given the ECG signal of a heartbeat is able to predict whether it is normal or abnormal.

#Dataset:

The dataset has been stored in the file ptb.Rdata. It consists of the following components:

X__train: a 12552x187 matrix where every row contains the th heartbeat signal; y__train: a vector of length 12552 with associated diagnostic labels (0: normal, 1 abnormal); X__test: a 2000x187 matrix where every row contains a heartbeat signal. For these 2000 test observations, we are not given the label.

#Preliminary Data Analysis:

```
#loading the .RData file
load('/Users/nayankaushal/Desktop/BIOST 546/Final Project/ptb.RData')

#creating data frame for training & validation
df = cbind(X_train, label = y_train)

#assigning X_test to test
test = X_test

#printing the dataframe
head(df)
```

```
##          X1        X2          X3          X4         X5          X6         X7
## 1 1.0000000 0.9003242 0.358589947 0.051458672 0.04659643 0.126823336 0.13330632
## 2 1.0000000 0.7946815 0.375386506 0.116883114 0.00000000 0.171923310 0.28385898
## 3 0.9090289 0.7914821 0.423168659 0.186712101 0.00000000 0.007836456 0.06303237
## 5 1.0000000 0.8672383 0.201360136 0.099349499 0.14133649 0.120934360 0.10851567
## 6 0.9489833 0.5052651 0.004175744 0.022512708 0.05954975 0.107298478 0.11038490
## 8 1.0000000 0.4603807 0.122177958 0.009296149 0.12571934 0.220008850 0.26737493
##           X8         X9        X10        X11        X12        X13        X14
## 1 0.11912480 0.11061589 0.11304700 0.10656402 0.10696921 0.11588331 0.12236629
## 2 0.29375386 0.32591218 0.34508348 0.36178109 0.36239952 0.36611009 0.36796537
## 3 0.07700171 0.07495741 0.07734242 0.07734242 0.08722317 0.09199318 0.09505963
## 5 0.09639267 0.09343584 0.10082791 0.08693081 0.09402721 0.09580130 0.09639267
## 6 0.11129267 0.11655773 0.11819172 0.11310820 0.13235295 0.12999274 0.13289760
## 8 0.26294822 0.26029217 0.27401504 0.28198317 0.28198317 0.29659140 0.29570606
##          X15        X16        X17        X18        X19       X20        X21
## 1 0.12236629 0.11952998 0.11588331 0.12236629 0.12601297 0.1337115 0.13492706
```

```
## 2 0.37414965 0.37786025 0.38218924 0.38466296 0.39888683 0.4013605 0.41805813
## 3 0.09642249 0.10494038 0.10800681 0.11379898 0.11652470 0.1199319 0.12436116
## 5 0.08959196 0.09491425 0.08959196 0.09491425 0.09816676 0.1023063 0.09964518
## 6 0.13725491 0.15123457 0.15795207 0.15559186 0.18010166 0.1870007 0.20025417
## 8 0.30146083 0.30987161 0.31651172 0.31916776 0.31562638 0.3227092 0.34351483
##          X22       X23       X24       X25       X26       X27       X28
## 1 0.14262562 0.1511345 0.1584279 0.1636953 0.1738250 0.1888169 0.2078606
## 2 0.44341373 0.4576376 0.4879406 0.5207174 0.5596784 0.6042053 0.6345084
## 3 0.13219762 0.1458262 0.1526406 0.1635434 0.1754685 0.1894378 0.2047700
## 5 0.09668835 0.1085157 0.1277351 0.1283264 0.1490242 0.1723832 0.2040213
## 6 0.22512709 0.2398330 0.2745098 0.2872186 0.3137255 0.3367829 0.3473130
## 8 0.36609119 0.3820274 0.3970784 0.4130146 0.4395750 0.4727756 0.4988933
##         X29       X30       X31       X32       X33       X34       X35
## 1 0.2309562 0.2585089 0.2945705 0.3257698 0.3626418 0.3982982 0.4294976
## 2 0.6536797 0.6728510 0.6784168 0.6604823 0.6215214 0.5559679 0.4823748
## 3 0.2293015 0.2528109 0.2759795 0.3025554 0.3212947 0.3339012 0.3458262
## 5 0.2217623 0.2542874 0.2847427 0.3092844 0.3296866 0.3441750 0.3598462
## 6 0.3542120 0.3400508 0.3168119 0.2677923 0.2236747 0.1801017 0.1561365
## 8 0.5258964 0.5542275 0.5807880 0.5900841 0.5922975 0.5874281 0.5661798
##         X36       X37       X38       X39       X40        X41       X42
## 1 0.4493517 0.4509724 0.4189627 0.3727715 0.3103728 0.25000000 0.2042139
## 2 0.4384663 0.3784787 0.3512678 0.3197279 0.3067409 0.29560915 0.2931354
## 3 0.3485520 0.3471891 0.3342419 0.3066440 0.2746167 0.23781942 0.2064736
## 5 0.3335305 0.2921348 0.2534004 0.2167357 0.1714962 0.13867535 0.1159077
## 6 0.1348947 0.1140160 0.1067538 0.1047567 0.0882353 0.09368192 0.1011256
## 8 0.5254537 0.4763170 0.4311642 0.3864542 0.3466136 0.31695440 0.2992474
##          X43        X44        X45        X46        X47        X48        X49
## 1 0.16896272 0.14748785 0.13047002 0.12439222 0.11750405 0.11669368 0.11588331
## 2 0.29189858 0.29251701 0.27891156 0.27891156 0.28076684 0.28076684 0.28571430
## 3 0.18364565 0.16252129 0.14514481 0.13526405 0.12572402 0.11993185 0.11584327
## 5 0.10644589 0.10348906 0.08515671 0.08338261 0.08900059 0.08190420 0.08072147
## 6 0.09694989 0.09658679 0.09495280 0.09658679 0.09531590 0.09259259 0.09931009
## 8 0.29260734 0.28021249 0.27091634 0.26294822 0.26250553 0.26029217 0.26073483
##          X50        X51        X52        X53        X54        X55        X56
## 1 0.11871961 0.11547812 0.11385737 0.11952998 0.11669368 0.12277147 0.12074554
## 2 0.27458256 0.27520099 0.27396414 0.28447741 0.27643785 0.27520099 0.27767470
## 3 0.11277683 0.11005111 0.11311755 0.11073254 0.10902896 0.11175469 0.11175469
## 5 0.08604376 0.08131284 0.07687759 0.08870491 0.09639267 0.10289770 0.08811354
## 6 0.10312273 0.10366739 0.10620915 0.10584604 0.09912854 0.09876543 0.10384895
## 8 0.26117751 0.26029217 0.25896415 0.25940681 0.25763613 0.25807878 0.26383355
##          X57       X58        X59        X60       X61        X62       X63
## 1 0.11669368 0.1227715 0.12641816 0.13168557 0.1418152 0.13938412 0.1450567
## 2 0.27952999 0.2826221 0.27952999 0.27334571 0.2683983 0.26901668 0.2677798
## 3 0.11618399 0.1199319 0.12129472 0.12061328 0.1178876 0.12299830 0.1243612
## 5 0.09461857 0.1058545 0.10141928 0.08811354 0.1005322 0.09550562 0.1046718
## 6 0.10439361 0.1005810 0.09277415 0.09204793 0.1000363 0.09077705 0.0905955
## 8 0.25763613 0.2576361 0.24701196 0.24037185 0.2390438 0.23771581 0.2363878
##          X64        X65        X66        X67        X68        X69        X70
## 1 0.14343598 0.14100486 0.14059968 0.13816856 0.13695300 0.13209076 0.12844409
## 2 0.25726655 0.25231910 0.25293753 0.25726655 0.24984539 0.25108224 0.25108224
## 3 0.12572402 0.12402044 0.12606473 0.12708688 0.12197615 0.12708688 0.12640545
## 5 0.09166174 0.08988764 0.09520993 0.09846245 0.08515671 0.08515671 0.09166174
## 6 0.09803922 0.09622367 0.08750908 0.09822077 0.09023239 0.08533043 0.09005083
## 8 0.23505977 0.24169987 0.24169987 0.24302790 0.24214254 0.24037185 0.23727313
```

```
##          X71        X72        X73        X74        X75        X76        X77
## 1 0.12844409 0.12803890 0.12520260 0.12236629 0.11709887 0.11264182 0.1130470
## 2 0.24984539 0.24180581 0.24118738 0.24366111 0.24489796 0.23933209 0.2418058
## 3 0.12879045 0.12640545 0.12436116 0.12504259 0.13219762 0.12947190 0.1270869
## 5 0.07628622 0.08308693 0.07746895 0.08397398 0.07539918 0.07421644 0.0836783
## 6 0.09331881 0.08315178 0.09222949 0.08297022 0.08242556 0.09749455 0.1223675
## 8 0.24347056 0.24214254 0.22487827 0.22133687 0.21868083 0.21425410 0.2222222
##          X78        X79        X80        X81        X82        X83        X84
## 1 0.12763371 0.16531605 0.17949757 0.16126418 0.17666127 0.18273906 0.17463534
## 2 0.23871367 0.24242425 0.24118738 0.23067409 0.23252937 0.22820038 0.23747681
## 3 0.12367973 0.12640545 0.12299830 0.12061328 0.12197615 0.12061328 0.12640545
## 5 0.07392076 0.07392076 0.06978119 0.07303371 0.07273802 0.06771141 0.07007688
## 6 0.11619463 0.11946260 0.13997822 0.09694989 0.12055192 0.11328976 0.09277415
## 8 0.22133687 0.21336874 0.20894201 0.21071270 0.21425410 0.21071270 0.20982736
##          X85        X86        X87        X88        X89        X90        X91
## 1 0.15153971 0.14789303 0.13492706 0.12277147 0.10696921 0.09805510 0.09440843
## 2 0.24304268 0.24366111 0.24304268 0.26901668 0.26345083 0.29066172 0.27643785
## 3 0.12265758 0.12402044 0.12163544 0.12470187 0.12844974 0.13151619 0.13185690
## 5 0.08338261 0.06268480 0.07244235 0.07096393 0.07865169 0.07539918 0.06534595
## 6 0.08732753 0.08333334 0.08496732 0.07244009 0.07952069 0.08641975 0.07933915
## 8 0.21027003 0.21735281 0.22045153 0.21823816 0.21691014 0.21691014 0.21159805
##          X92        X93        X94        X95        X96        X97        X98
## 1 0.08914100 0.08914100 0.08873582 0.09076175 0.08589952 0.08589952 0.08914100
## 2 0.27829313 0.25170067 0.25664812 0.25231910 0.24613482 0.24675325 0.23809524
## 3 0.13492334 0.13662691 0.13560477 0.13662691 0.13458262 0.13492334 0.13492334
## 5 0.08072147 0.07332939 0.06475458 0.08013010 0.07185098 0.07007688 0.06564163
## 6 0.07425563 0.08333334 0.02487291 0.01724764 0.20769790 0.53013796 1.00000000
## 8 0.20628597 0.20584330 0.20628597 0.20097388 0.20407259 0.20451528 0.20495795
##          X99       X100       X101       X102       X103       X104       X105
## 1 0.08427877 0.05794165 0.00000000 0.11628849 0.30956239 0.83427876 0.96434361
## 2 0.22077923 0.23067409 0.23562153 0.24613482 0.24427953 0.25046381 0.25850341
## 3 0.13594548 0.14071551 0.13901192 0.14071551 0.14105622 0.14446338 0.14480409
## 5 0.07539918 0.06386753 0.06386753 0.07007688 0.06623300 0.06889415 0.06712005
## 6 0.92483658 0.44135803 0.00000000 0.02251271 0.07461873 0.11002178 0.12273058
## 8 0.21425410 0.21469676 0.20982736 0.22664896 0.26604691 0.24391323 0.24612661
##         X106       X107       X108       X109       X110       X111       X112
## 1 0.56158835 0.08144246 0.03241491 0.11264182 0.15235008 0.13573743 0.12479740
## 2 0.25602970 0.34693879 0.41682127 0.51700681 0.86951143 0.98453927 0.55534941
## 3 0.14446338 0.14412266 0.14514481 0.14889267 0.14821124 0.15127768 0.14821124
## 5 0.06238912 0.06918983 0.06534595 0.07066824 0.06386753 0.06682436 0.08456535
## 6 0.10130719 0.11365287 0.10856935 0.11855483 0.11456064 0.13162673 0.13380538
## 8 0.21735281 0.21425410 0.19964586 0.20672864 0.19477645 0.19034971 0.18857902
##        X113       X114      X115      X116      X117      X118      X119
## 1 0.1235818 0.12358185 0.1158833 0.1154781 0.1166937 0.1247974 0.1280389
## 2 0.2418058 0.03092146 0.0445269 0.2275819 0.2683983 0.2813853 0.3166358
## 3 0.1468484 0.14787053 0.1492334 0.1516184 0.1632027 0.1884157 0.2091993
## 5 0.1055588 0.08900059 0.1017150 0.1250739 0.1374926 0.1428149 0.1593731
## 6 0.1425200 0.15395787 0.1521423 0.1630356 0.1659404 0.1926289 0.2002542
## 8 0.1903497 0.19477645 0.1912351 0.1903497 0.1974325 0.1921204 0.1841523
##        X120      X121       X122       X123       X124       X125       X126
## 1 0.1300648 0.1268233 0.12965964 0.13614263 0.00000000 0.00000000 0.00000000
## 2 0.3259122 0.3314781 0.33951762 0.34879407 0.34693879 0.34446505 0.35003093
## 3 0.2160136 0.2538331 0.25826234 0.26030666 0.26132879 0.23270869 0.21976151
## 5 0.1250739 0.1096984 0.08219988 0.06327616 0.04997043 0.04376109 0.03548196
```

```
## 6 0.2182280 0.2403776 0.26724765 0.30700800 0.32244009 0.33315179 0.35039940
## 8 0.2151394 0.3678619 0.57724655 0.89597166 0.93625498 0.37760070 0.07658256
##         X127       X128       X129       X130       X131       X132      X133
## 1 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.0000000
## 2 0.35930735 0.36672851 0.37662336 0.38589981 0.39764997 0.41682127 0.0000000
## 3 0.21260647 0.16967632 0.14037478 0.12981261 0.12674616 0.12538330 0.1260647
## 5 0.03962152 0.03962152 0.03962152 0.04435245 0.04050858 0.03873448 0.0000000
## 6 0.35893247 0.34713143 0.00000000 0.00000000 0.00000000 0.00000000 0.0000000
## 8 0.00000000 0.12350598 0.21027003 0.25320938 0.26560426 0.27180168 0.2775564
##        X134      X135      X136      X137      X138       X139      X140
## 1 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000000 0.0000000
## 2 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000000 0.0000000
## 3 0.1287905 0.1127768 0.1352641 0.2289608 0.5908007 0.99045998 1.0000000
## 5 0.2031342 0.5901833 0.9305145 0.9858072 0.2953874 0.06534595 0.1052632
## 6 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000000 0.0000000
## 8 0.2793271 0.2740150 0.2766711 0.2713590 0.2726870 0.28065515 0.2810978
##        X141       X142      X143       X144      X145      X146      X147
## 1 0.0000000 0.00000000 0.0000000 0.00000000 0.0000000 0.0000000 0.0000000
## 2 0.0000000 0.00000000 0.0000000 0.00000000 0.0000000 0.0000000 0.0000000
## 3 0.6749574 0.33253834 0.1315162 0.07768314 0.1342419 0.1672913 0.1662692
## 5 0.1017150 0.08338261 0.0000000 0.00000000 0.0000000 0.0000000 0.0000000
## 6 0.0000000 0.00000000 0.0000000 0.00000000 0.0000000 0.0000000 0.0000000
## 8 0.2833112 0.29260734 0.3014608 0.32359451 0.3328907 0.3505976 0.0000000
##        X148      X149     X150      X151      X152      X153      X154
## 1 0.0000000 0.0000000 0.000000 0.0000000 0.0000000 0.0000000 0.0000000
## 2 0.0000000 0.0000000 0.000000 0.0000000 0.0000000 0.0000000 0.0000000
## 3 0.1683135 0.1683135 0.170017 0.1741056 0.1781942 0.1836457 0.1863714
## 5 0.0000000 0.0000000 0.000000 0.0000000 0.0000000 0.0000000 0.0000000
## 6 0.0000000 0.0000000 0.000000 0.0000000 0.0000000 0.0000000 0.0000000
## 8 0.0000000 0.0000000 0.000000 0.0000000 0.0000000 0.0000000 0.0000000
##        X155      X156      X157      X158      X159      X160     X161
## 1 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.000000
## 2 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.000000
## 3 0.1911414 0.1945485 0.2017036 0.2081772 0.2122658 0.2197615 0.227598
## 5 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.000000
## 6 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.000000
## 8 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.000000
##        X162     X163      X164      X165 X166 X167 X168 X169 X170 X171 X172
## 1 0.0000000 0.000000 0.0000000 0.0000000    0    0    0    0    0    0    0
## 2 0.0000000 0.000000 0.0000000 0.0000000    0    0    0    0    0    0    0
## 3 0.2388416 0.249063 0.2545145 0.2701874    0    0    0    0    0    0    0
## 5 0.0000000 0.000000 0.0000000 0.0000000    0    0    0    0    0    0    0
## 6 0.0000000 0.000000 0.0000000 0.0000000    0    0    0    0    0    0    0
## 8 0.0000000 0.000000 0.0000000 0.0000000    0    0    0    0    0    0    0
##   X173 X174 X175 X176 X177 X178 X179 X180 X181 X182 X183 X184 X185 X186 X187
## 1    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
## 2    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
## 3    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
## 5    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
## 6    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
## 8    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
##   label
## 1     0
## 2     0
```

```
## 3     0
## 5     0
## 6     0
## 8     0
```

```r
#printing sample size
nrow(df)
```

```
## [1] 12552
```

```r
#printing number of predictors
length(df)-1
```

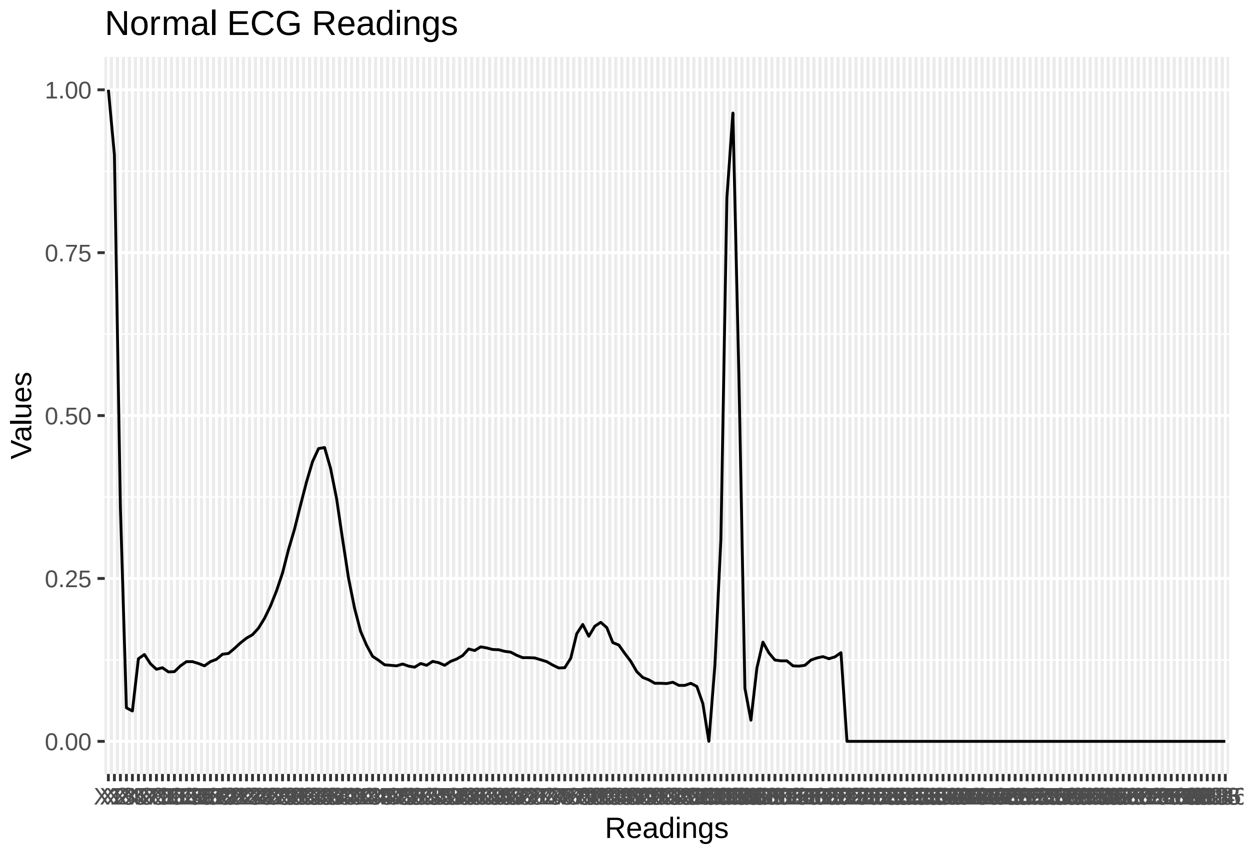
```
## [1] 187
```

```r
#printing number of observations in each class
table(df$M)
```

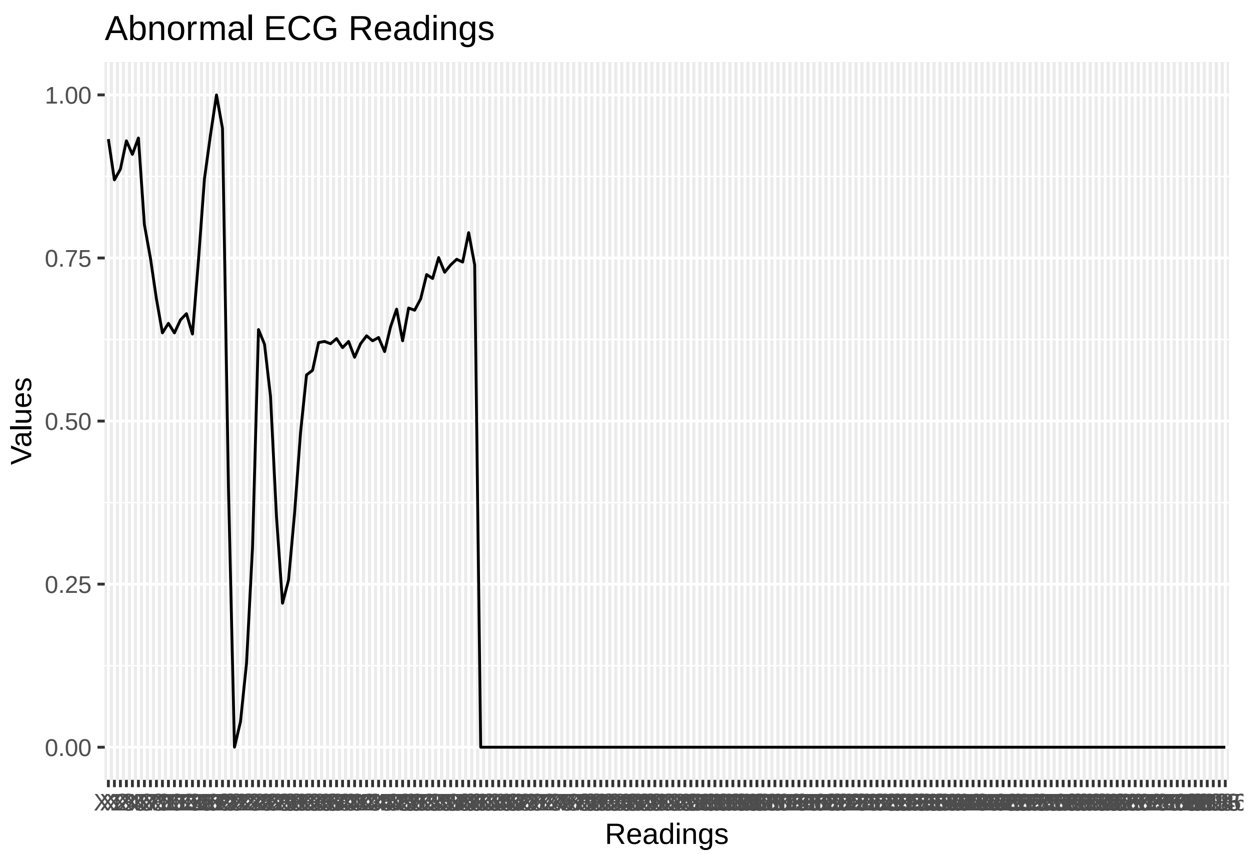
```
## < table of extent 0 >
```

#Exploratory Data Analysis:

```r
#reshaping the dataset
df_reshaped = df %>% mutate(id = rownames(df))
df_reshaped = melt(df_reshaped, id = c("id", "label"))

#plotting the normal ECG readings
df_reshaped %>% filter(id == 1) %>% ggplot(aes(x = factor(variable), y = value, group = 1)) + geom_line
```




```
#plotting the abnormal ECG readings
df_reshaped %>% filter(id == 116) %>% ggplot(aes(x = factor(variable), y = value, group = 1)) + geom_lin
```



#Splitting the Training data into train and test:

```
#setting seed
set.seed(0)

#dividing data into a training set of 400 observations and test set
train_id = sample(nrow(df), as.integer(nrow(df)*0.7))
train_df = df[train_id,]
test_df  = df[-train_id,]
```

#Generalized Linear Model

Creating a generalized linear model:

```
#creating a generalized linear model using glm() function
glm.model = glm(formula = label ~ ., family = binomial(link = "logit"), data = train_df)

#printing the model summary
summary(glm.model)
```

```
##
## Call:
## glm(formula = label ~ ., family = binomial(link = "logit"), data = train_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
```

```
## -3.6823   0.0003   0.1864   0.5862   2.5642
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    4.57610    1.08376   4.222 2.42e-05 ***
## X1            -2.80692    1.07489  -2.611 0.009018 **
## X2            -3.92879    0.28502 -13.784  < 2e-16 ***
## X3             4.10523    0.39455  10.405  < 2e-16 ***
## X4             0.31416    0.56495   0.556 0.578154
## X5             5.96180    0.79732   7.477 7.59e-14 ***
## X6             9.05158    1.20477   7.513 5.77e-14 ***
## X7           -15.49211    1.82264  -8.500  < 2e-16 ***
## X8            -1.92831    2.28820  -0.843 0.399384
## X9            -0.43531    2.66131  -0.164 0.870070
## X10            3.48332    2.82808   1.232 0.218065
## X11            2.03060    3.01932   0.673 0.501242
## X12           -3.29446    2.89979  -1.136 0.255913
## X13           -6.88349    2.98594  -2.305 0.021150 *
## X14            1.52157    2.63842   0.577 0.564144
## X15            5.14130    2.66449   1.930 0.053661 .
## X16           -3.31552    2.59618  -1.277 0.201575
## X17           -0.76380    2.64369  -0.289 0.772648
## X18            4.84553    2.55853   1.894 0.058242 .
## X19            3.77758    2.48789   1.518 0.128916
## X20           -0.93539    2.35317  -0.398 0.690998
## X21           -2.76608    2.68728  -1.029 0.303328
## X22            4.00258    2.65516   1.507 0.131690
## X23            2.07188    2.51723   0.823 0.410462
## X24            3.72614    2.47616   1.505 0.132374
## X25           -6.65726    2.82123  -2.360 0.018290 *
## X26           -4.33384    2.74169  -1.581 0.113942
## X27            3.09048    2.79213   1.107 0.268357
## X28           -0.75020    2.63209  -0.285 0.775629
## X29            1.86042    2.63953   0.705 0.480915
## X30           -1.29249    2.30629  -0.560 0.575194
## X31           -1.18420    2.51532  -0.471 0.637787
## X32           -3.47956    2.39958  -1.450 0.147038
## X33           -7.24959    2.77914  -2.609 0.009092 **
## X34            1.41225    2.81465   0.502 0.615845
## X35            1.78872    2.69418   0.664 0.506741
## X36            2.79944    2.56717   1.090 0.275504
## X37            4.89982    2.62836   1.864 0.062292 .
## X38            0.58071    2.82015   0.206 0.836858
## X39            1.07137    2.87118   0.373 0.709040
## X40           -7.96304    2.95566  -2.694 0.007056 **
## X41           -3.02820    3.09636  -0.978 0.328080
## X42           -1.19969    3.02134  -0.397 0.691315
## X43            1.85438    3.09030   0.600 0.548463
## X44           -7.95184    3.00039  -2.650 0.008043 **
## X45            1.46995    3.04960   0.482 0.629795
## X46            5.54999    3.03829   1.827 0.067747 .
## X47           10.10076    2.88943   3.496 0.000473 ***
## X48            4.89125    2.90458   1.684 0.092186 .
## X49            1.80006    2.93113   0.614 0.539138
```

```
## X50           5.99269    2.84329   2.108 0.035060 *
## X51          -0.54863    2.62498  -0.209 0.834445
## X52           1.61178    2.72072   0.592 0.553575
## X53           1.36806    2.71067   0.505 0.613775
## X54          -4.95861    2.04038  -2.430 0.015089 *
## X55          -2.14554    2.30614  -0.930 0.352186
## X56          -1.65650    2.30472  -0.719 0.472300
## X57           3.01212    2.16819   1.389 0.164762
## X58           1.62163    1.95727   0.829 0.407379
## X59          -1.65780    1.86447  -0.889 0.373920
## X60           0.10488    1.93064   0.054 0.956677
## X61           4.70062    1.87912   2.501 0.012367 *
## X62          -2.69806    1.80629  -1.494 0.135253
## X63          -0.09088    1.79699  -0.051 0.959666
## X64          -1.79678    1.64144  -1.095 0.273676
## X65          -0.03989    1.91350  -0.021 0.983370
## X66          -2.07961    2.14608  -0.969 0.332530
## X67           2.11350    1.98623   1.064 0.287295
## X68          -1.42985    1.84021  -0.777 0.437157
## X69           3.12432    2.10860   1.482 0.138419
## X70          -1.83031    1.94737  -0.940 0.347274
## X71           1.98863    1.68139   1.183 0.236917
## X72          -1.81709    1.25815  -1.444 0.148666
## X73           0.11109    1.08091   0.103 0.918139
## X74           0.07992    0.87133   0.092 0.926922
## X75          -0.52334    0.83893  -0.624 0.532748
## X76          -1.05138    1.02460  -1.026 0.304829
## X77           0.16510    1.12047   0.147 0.882858
## X78          -0.14080    1.12953  -0.125 0.900796
## X79          -0.08281    1.04190  -0.079 0.936652
## X80          -0.70534    0.91939  -0.767 0.442974
## X81           0.82340    0.90445   0.910 0.362622
## X82           0.69583    0.88979   0.782 0.434207
## X83          -0.52385    0.82931  -0.632 0.527600
## X84           0.59302    0.78088   0.759 0.447598
## X85          -1.63664    0.74800  -2.188 0.028669 *
## X86           1.31847    0.75469   1.747 0.080632 .
## X87           0.44906    0.73204   0.613 0.539588
## X88          -1.12808    0.66366  -1.700 0.089172 .
## X89           0.25276    0.65095   0.388 0.697793
## X90           0.49304    0.67573   0.730 0.465608
## X91           1.50113    0.71900   2.088 0.036816 *
## X92          -0.83789    0.71725  -1.168 0.242729
## X93           0.49299    0.72920   0.676 0.498996
## X94          -0.23398    0.74426  -0.314 0.753231
## X95           0.17542    0.73415   0.239 0.811152
## X96          -0.83756    0.70460  -1.189 0.234559
## X97           0.67366    0.64967   1.037 0.299773
## X98           0.41159    0.61992   0.664 0.506728
## X99           0.34956    0.62612   0.558 0.576641
## X100         -0.46014    0.63582  -0.724 0.469251
## X101          0.49093    0.65127   0.754 0.450967
## X102         -0.27066    0.63101  -0.429 0.667972
## X103          1.05757    0.63534   1.665 0.095994 .
```

```
## X104          -1.35770    0.62241  -2.181 0.029156 *
## X105           1.75048    0.62763   2.789 0.005287 **
## X106          -1.95975    0.64194  -3.053 0.002267 **
## X107           1.37234    0.70359   1.950 0.051118 .
## X108          -0.45799    0.72388  -0.633 0.526937
## X109          -0.82522    0.69008  -1.196 0.231764
## X110           0.49729    0.71900   0.692 0.489160
## X111          -0.33495    0.74010  -0.453 0.650855
## X112          -0.27719    0.74592  -0.372 0.710187
## X113          -0.03733    0.76887  -0.049 0.961278
## X114          -0.34017    0.78411  -0.434 0.664412
## X115          -0.02377    0.71245  -0.033 0.973382
## X116          -0.39753    0.66129  -0.601 0.547748
## X117           0.74506    0.63226   1.178 0.238637
## X118          -1.55588    0.63924  -2.434 0.014935 *
## X119           0.76192    0.70339   1.083 0.278713
## X120           0.34284    0.72403   0.474 0.635845
## X121          -1.77776    0.74134  -2.398 0.016483 *
## X122           1.35982    0.74904   1.815 0.069461 .
## X123          -0.78594    0.73103  -1.075 0.282325
## X124           0.94968    0.74632   1.272 0.203200
## X125          -0.26520    0.81049  -0.327 0.743512
## X126          -0.79547    0.82523  -0.964 0.335075
## X127           0.16551    0.79118   0.209 0.834297
## X128           0.26274    0.82069   0.320 0.748857
## X129          -0.08557    0.83314  -0.103 0.918196
## X130          -0.15773    0.83275  -0.189 0.849771
## X131           0.62419    0.90212   0.692 0.488987
## X132          -1.53156    0.94101  -1.628 0.103618
## X133           0.56424    0.88074   0.641 0.521756
## X134           0.25670    0.96208   0.267 0.789611
## X135           0.20751    0.97356   0.213 0.831210
## X136          -1.59347    1.05767  -1.507 0.131916
## X137           1.40984    1.22365   1.152 0.249256
## X138          -0.27123    1.22108  -0.222 0.824220
## X139          -0.25100    0.92441  -0.272 0.785985
## X140          -0.70311    0.96028  -0.732 0.464053
## X141          -1.51051    1.03342  -1.462 0.143835
## X142           1.74375    1.13176   1.541 0.123380
## X143           1.66663    1.53215   1.088 0.276698
## X144          -4.20895    1.54325  -2.727 0.006385 **
## X145           1.93449    1.22357   1.581 0.113872
## X146          -0.71614    1.18995  -0.602 0.547291
## X147          -1.43254    1.13494  -1.262 0.206870
## X148           3.22291    1.19454   2.698 0.006975 **
## X149          -4.97416    1.73975  -2.859 0.004248 **
## X150           2.57294    1.71528   1.500 0.133610
## X151           0.01423    1.33339   0.011 0.991486
## X152          -1.26620    1.66029  -0.763 0.445679
## X153           0.54696    1.89246   0.289 0.772565
## X154           0.05533    1.76754   0.031 0.975027
## X155           1.82736    2.50555   0.729 0.465803
## X156          -2.72854    2.32098  -1.176 0.239756
## X157           0.43415    1.45615   0.298 0.765588
```

```
## X158           1.50060    1.71059   0.877 0.380356
## X159          -3.81169    1.78044  -2.141 0.032285 *
## X160           4.06945    1.85849   2.190 0.028550 *
## X161          -5.49000    3.35818  -1.635 0.102089
## X162           2.96838    3.46555   0.857 0.391700
## X163           2.88313    2.21684   1.301 0.193410
## X164          -0.88178    2.25077  -0.392 0.695232
## X165          -6.37712    2.78321  -2.291 0.021947 *
## X166           9.20540    2.78181   3.309 0.000936 ***
## X167           8.19836    6.60545   1.241 0.214550
## X168          -7.00727    6.06641  -1.155 0.248052
## X169          -4.77211    3.18619  -1.498 0.134199
## X170           8.95009    5.82924   1.535 0.124690
## X171          -8.24530    5.57785  -1.478 0.139348
## X172           7.59731    3.37314   2.252 0.024303 *
## X173           4.92943    5.42386   0.909 0.363434
## X174         -12.10783    6.36718  -1.902 0.057223 .
## X175           6.76774    7.20810   0.939 0.347778
## X176          -8.11832   12.02465  -0.675 0.499587
## X177           6.52320   11.22539   0.581 0.561166
## X178          -1.17803    4.98744  -0.236 0.813278
## X179          -6.69303    7.36557  -0.909 0.363513
## X180           5.90399    8.52927   0.692 0.488809
## X181           2.98540    6.97150   0.428 0.668485
## X182          -6.82397    7.46660  -0.914 0.360753
## X183           2.41594    7.00450   0.345 0.730160
## X184          -4.39601    6.16794  -0.713 0.476020
## X185        -217.45536  216.46007  -1.005 0.315090
## X186         205.66975  200.13909   1.028 0.304122
## X187               NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9780.1  on 8785  degrees of freedom
## Residual deviance: 6090.5  on 8599  degrees of freedom
## AIC: 6464.5
##
## Number of Fisher Scoring iterations: 7
```

Training and Testing Model Accuracy:

```r
#=========================================
#Training
#predicting the training set probability
glm.prob.train = predict(glm.model, newdata = train_df,  type = "response")
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```r
#predicting the outcome
glm.label.train = rep(0, nrow(train_df))
```

```r
glm.label.train[glm.prob.train > .5] = 1

print('Training Data Accuracy')
```

```
## [1] "Training Data Accuracy"
```

```r
# Prediction Accuracy
mean(glm.label.train == train_df$label)
```

```
## [1] 0.8480537
```

```r
print('Training Data Confusion Matrix')
```

```
## [1] "Training Data Confusion Matrix"
```

```r
# Confusion matrix
tt.glm.train = table(True = glm.label.train, Predicted = train_df$label)
tt.glm.train
```

```
##     Predicted
## True    0    1
##    0 1263  447
##    1  888 6188
```

```r
#=============================================
#Testing
#predicting the test set probability
glm.prob.test = predict(glm.model, newdata = test_df,  type = "response")
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```r
#predicting the outcome
glm.label.test = rep(0, nrow(test_df))
glm.label.test[glm.prob.test > .5] = 1

print('Testing Data Accuracy')
```

```
## [1] "Testing Data Accuracy"
```

```r
# Prediction Accuracy
mean(glm.label.test == test_df$label)
```

```
## [1] 0.834838
```

```r
print('Testing Data Confusion Matrix')
```

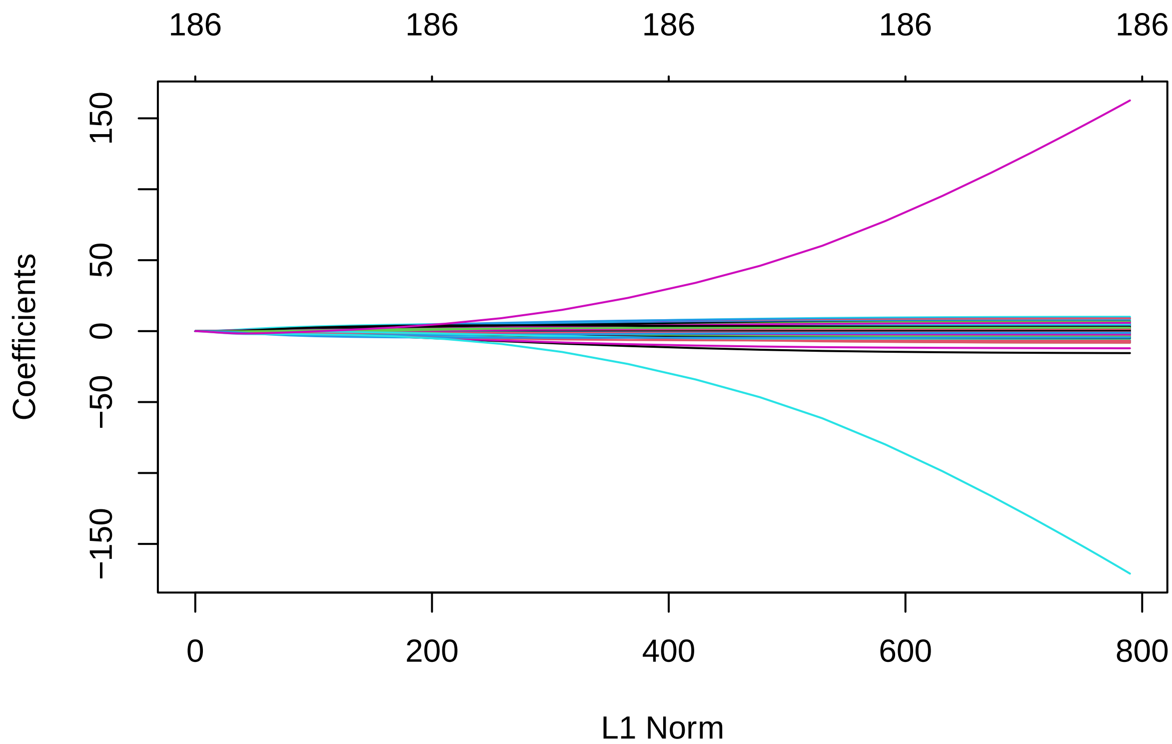
```
## [1] "Testing Data Confusion Matrix"
```

```r
# Confusion matrix
tt.glm.test = table(True = glm.label.test, Predicted = test_df$label)
tt.glm.test
```

```
##      Predicted
## True    0    1
##    0  488  215
##    1  407 2656
```

#Ridge Regularization:

```r
#constructing data matrix X train
X_train = model.matrix(label ~ -1 + ., data = train_df)

#constructing outcome vector y train
y_train = train_df$label

#constructing data matrix X test
X_test = model.matrix(label ~ -1 + ., data = test_df)

#constructing outcome vector y test
y_test = test_df$label

# Validation set approach -- Train model only on training set
grid = 10^seq(5,-18,length=100)
ridge.mod = glmnet(X_train, y_train, alpha=0, lambda = grid, thresh =1e-8, family = "binomial")

#plot ridge model
plot(ridge.mod)
```

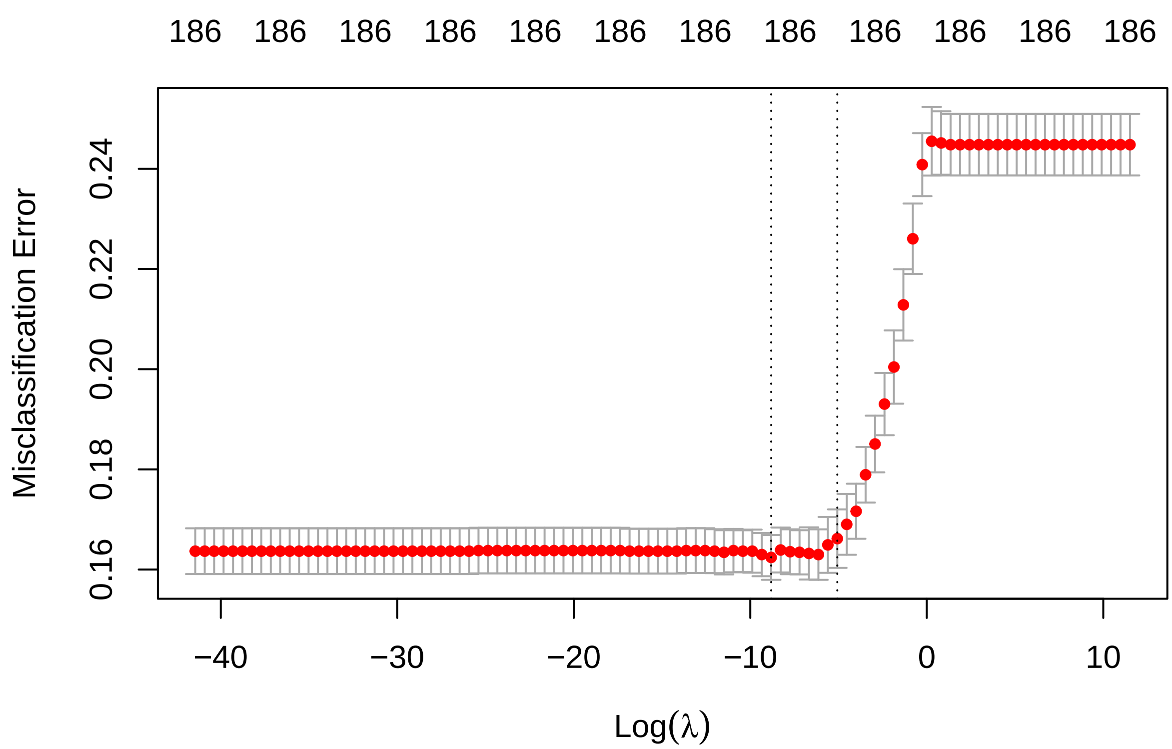



```r
#Performing 10 fold cross validation
cv.out = cv.glmnet(X_train, y_train, alpha=0, lambda = grid, family = 'binomial', type.measure = "class
```

```
## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values
```

```
## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values

## Warning in regularize.values(x, y, ties, missing(ties), na.rm = na.rm):
## collapsing to unique 'x' values
```

```
#plotting 10 fold cross validation
plot(cv.out)
```




```
#finding the optimal lambda value
bestlam = cv.out$lambda.min
print(paste0('Optimal Value of lambda = ',bestlam))
```

```
## [1] "Optimal Value of lambda = 0.000148496826225446"
```

```
#finding the best ridge model with the optimal lambda
ridge.best = glmnet(X_train, y_train, alpha=0, lambda = bestlam, family = 'binomial', type.measure = "cl
coef(ridge.best)
```

```
## 188 x 1 sparse Matrix of class "dgCMatrix"
##                      s0
## (Intercept)   4.43673958
## X1           -2.67914131
```

```
## X2            -3.84166674
## X3             3.95152257
## X4             0.43981156
## X5             5.89916258
## X6             7.84742174
## X7           -13.13293537
## X8            -2.94347639
## X9            -0.28380816
## X10            2.35371354
## X11            1.25103921
## X12           -2.06057548
## X13           -4.76556363
## X14            1.14207032
## X15            3.05766488
## X16           -2.35165371
## X17            0.04071529
## X18            3.33994479
## X19            2.94675768
## X20           -0.51266689
## X21           -1.52931859
## X22            3.24446509
## X23            2.08249712
## X24            2.62406824
## X25           -4.34684083
## X26           -3.24434863
## X27            1.58890804
## X28           -0.61204795
## X29            0.94417219
## X30           -1.42836727
## X31           -1.95183298
## X32           -3.05458232
## X33           -5.15376983
## X34            0.69237670
## X35            1.83633818
## X36            2.68903921
## X37            3.71158510
## X38            0.99510619
## X39            0.24339283
## X40           -5.63630080
## X41           -3.23167905
## X42           -1.82169888
## X43           -0.04675489
## X44           -5.37883455
## X45            1.38353565
## X46            5.16630837
## X47            9.08524261
## X48            4.99298501
## X49            2.58263446
## X50            4.46150983
## X51            0.36356167
## X52            1.52730957
## X53            0.65664413
## X54           -3.82044404
## X55           -2.47727125
```

```
## X56          -1.37605288
## X57           2.64111067
## X58           1.36875555
## X59          -1.05541339
## X60           0.24609761
## X61           3.79628789
## X62          -2.11307628
## X63          -0.32787924
## X64          -1.64884494
## X65          -0.33261115
## X66          -1.51367376
## X67           1.46170270
## X68          -0.60212869
## X69           2.11194714
## X70          -1.00555089
## X71           1.34609386
## X72          -1.43929954
## X73           0.03129412
## X74           0.09966301
## X75          -0.59788990
## X76          -0.98102553
## X77           0.16173297
## X78          -0.13527477
## X79          -0.10372942
## X80          -0.63518844
## X81           0.77918247
## X82           0.67358089
## X83          -0.44909410
## X84           0.46949624
## X85          -1.45377283
## X86           1.17552798
## X87           0.48745699
## X88          -1.07290825
## X89           0.22006952
## X90           0.52006736
## X91           1.43031961
## X92          -0.79978340
## X93           0.46058358
## X94          -0.17665059
## X95           0.10954244
## X96          -0.78816897
## X97           0.63826975
## X98           0.40031934
## X99           0.35981303
## X100         -0.46723855
## X101          0.44642191
## X102         -0.20356145
## X103          0.93896219
## X104         -1.21777683
## X105          1.58103519
## X106         -1.79985066
## X107          1.27204807
## X108         -0.42629051
## X109         -0.78691464
```

```
## X110          0.43026544
## X111         -0.29796765
## X112         -0.27195630
## X113         -0.06589222
## X114         -0.32325229
## X115         -0.03644095
## X116         -0.37622288
## X117          0.71032534
## X118         -1.52185742
## X119          0.78452632
## X120          0.27586578
## X121         -1.67695959
## X122          1.27582191
## X123         -0.71984786
## X124          0.91205288
## X125         -0.27201426
## X126         -0.76234663
## X127          0.17739627
## X128          0.19735723
## X129         -0.02431887
## X130         -0.17218634
## X131          0.58960985
## X132         -1.45870612
## X133          0.49826977
## X134          0.27129681
## X135          0.17413367
## X136         -1.46644435
## X137          1.27352540
## X138         -0.21621121
## X139         -0.25395469
## X140         -0.69679924
## X141         -1.47137591
## X142          1.74344797
## X143          1.43361747
## X144         -3.92587944
## X145          1.87891705
## X146         -0.71765962
## X147         -1.42022783
## X148          3.01569514
## X149         -4.72087384
## X150          2.42406606
## X151          0.06829189
## X152         -1.23974362
## X153          0.41216901
## X154          0.11974871
## X155          1.41457307
## X156         -2.18424653
## X157          0.54208163
## X158          1.30077656
## X159         -3.54882502
## X160          3.71532306
## X161         -4.59020696
## X162          2.21985999
## X163          2.75879001
```

```
## X164          -0.92705165
## X165          -5.52177192
## X166           8.45765684
## X167           6.21027139
## X168          -5.20129917
## X169          -4.37244837
## X170           7.08173201
## X171          -6.35613766
## X172           6.74759629
## X173           4.49313134
## X174         -10.90519110
## X175           5.88419230
## X176          -6.51064552
## X177           5.41412430
## X178          -1.53796160
## X179          -5.52505503
## X180           4.84934977
## X181           3.46176039
## X182          -6.62994013
## X183           2.30261631
## X184          -4.57819714
## X185         -53.99910583
## X186          53.02233266
## X187           .
```

Training and Testing Model Accuracy:

```r
#Training
#predicting the training set probability
ridge.prob.train = predict(ridge.best, newx = X_train, type = "response")

#predicting the outcome
ridge.label.train = rep(0, nrow(X_train))
ridge.label.train[ridge.prob.train > .5] = 1

print('Training Data Accuracy')
```

```
## [1] "Training Data Accuracy"
```

```r
# Prediction Accuracy
mean(ridge.label.train == y_train)
```

```
## [1] 0.8486228
```

```r
print('Training Data Confusion Matrix')
```

```
## [1] "Training Data Confusion Matrix"
```

```r
# Confusion matrix
tt.ridge.train = table(True = ridge.label.train, Predicted = y_train)
tt.ridge.train
```

```
##     Predicted
## True    0    1
##    0 1258  437
##    1  893 6198
```

```
#Testing
#predicting the test set probability
ridge.prob.test = predict(ridge.best, newx = X_test,  type = "response")

#predicting the outcome
ridge.label.test = rep(0, nrow(X_test))
ridge.label.test[ridge.prob.test > .5] = 1

print('Testing Data Accuracy')
```

```
## [1] "Testing Data Accuracy"
```

```
# Prediction Accuracy
mean(ridge.label.test == y_test)
```

```
## [1] 0.8366968
```

```
print('Testing Data Confusion Matrix')
```

```
## [1] "Testing Data Confusion Matrix"
```

```
# Confusion matrix
tt.ridge.test = table(True = ridge.label.test, Predicted = y_test)
tt.ridge.test
```

```
##     Predicted
## True    0    1
##    0  478  198
##    1  417 2673
```

#Decision Trees Model:

```
# Fit (overgrown) tree
tree.med<-tree(label~.,train_df)
summary(tree.med)
```

```
##
## Classification tree:
## tree(formula = label ~ ., data = train_df)
## Variables actually used in tree construction:
## [1] "X5"   "X34"  "X50"  "X110" "X30"  "X39"  "X83"  "X4"   "X23"
## Number of terminal nodes:  13
## Residual mean deviance:  0.6365 = 5584 / 8773
## Misclassification error rate: 0.1338 = 1176 / 8786
```

```r
#plotting the overgrown tree
dev.new()
plot(tree.med)
text(tree.med)

#set seed
set.seed(2)

# performing cross validation on the overgrown tree to check for the best estimate of depth
cv.med=cv.tree(tree.med, FUN=prune.misclass)
cv.med
```

```
## $size
## [1] 13 12 10  9  8  6  1
##
## $dev
## [1] 1419 1419 1500 1500 1604 1615 1960
##
## $k
## [1]  -Inf   0.0  52.0  55.0  86.0  91.0 109.6
##
## $method
## [1] "misclass"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

```r
plot(cv.med$size,cv.med$dev,type="b")

# Pruning the overgrown tree
prune.med<-prune.tree(tree.med,best=12)
plot(prune.med)
text(prune.med,pretty=0)
```

Training and Testing Accuracy:

```r
#Training
#predicting the training set label
tree.label.train = predict(prune.med, newdata = train_df,  type = "class")

print('Training Data Accuracy')
```

```
## [1] "Training Data Accuracy"
```

```r
# Prediction Accuracy
mean(tree.label.train == train_df$label)
```

```
## [1] 0.8661507
```

```r
#===============================================
#Testing
#predicting the test set label
tree.label.test = predict(prune.med, newdata = test_df,  type = "class")

print('Test Data Accuracy')
```

```
## [1] "Test Data Accuracy"
```

```r
# Prediction Accuracy
mean(tree.label.test == test_df$label)
```

```
## [1] 0.8571429
```

#Random Forest Model:

```r
#setting seed
set.seed(2)

#random forest model with m = p/3 (which is 60)
rf.med<-randomForest(label~., data=train_df, mtry = 60, importance=TRUE)

#Predicting Training Data response and accuracy
print('Training Data Accuracy')
```

```
## [1] "Training Data Accuracy"
```

```r
yhat.rf<-predict(rf.med, newdata=train_df)
mean(yhat.rf == train_df$label)
```

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

```r
#Predicting Test Data response and accuracy
print('Test Data Accuracy')
```

```
## [1] "Test Data Accuracy"
```

```r
yhat.rf<-predict(rf.med,newdata=test_df)
mean(yhat.rf == test_df$label)
```

```
## [1] 0.9699947
```

```r
#Plotting the variable importances
importance(rf.med)
```

```
##                0         1 MeanDecreaseAccuracy MeanDecreaseGini
## X1    7.94687612  4.575943             8.997702     4.715056e+00
## X2   21.57525368 24.049314            25.881427     4.734988e+01
## X3   21.85872016 20.173139            28.300037     5.222197e+01
```

```
## X4   26.80314231 22.848969            32.932687     8.418150e+01
## X5   70.52672012 36.141262            65.755101     2.972841e+02
## X6   19.91057161 10.519399            21.912747     3.870230e+01
## X7   20.09481972 11.614835            23.707974     2.771532e+01
## X8   22.02486962 10.503026            23.525410     2.770984e+01
## X9   14.28933247  7.657444            13.946496     1.497525e+01
## X10  10.82288309  7.234829            12.052657     1.202738e+01
## X11  13.24486070  7.382801            11.434082     1.072783e+01
## X12  10.91762083  6.835421            12.700097     1.036078e+01
## X13  12.33931431  6.356801            12.562619     1.185362e+01
## X14  11.67035757  5.056296            12.803136     9.921502e+00
## X15  10.72999241  5.080387            13.008888     8.547339e+00
## X16  10.68164312  5.095508            12.861941     9.574073e+00
## X17  12.51206375  3.646654            13.820625     9.606227e+00
## X18   9.72996571  4.040682            11.773796     7.211879e+00
## X19  12.20746089  6.039714            14.581748     1.081163e+01
## X20  11.89553815  5.438136            14.491042     1.145752e+01
## X21  10.56730598  7.652224            13.900053     1.111973e+01
## X22  14.69840594  6.642734            16.620762     1.715057e+01
## X23  13.16441448  9.201110            16.195678     2.021881e+01
## X24  13.85278460  7.959137            14.865995     1.843655e+01
## X25  16.13889026 10.260706            18.700391     2.242474e+01
## X26  15.23164829  9.913467            15.809636     2.270232e+01
## X27  15.45825971 13.289171            17.486883     2.763089e+01
## X28  17.02656471 19.551576            23.918902     4.048555e+01
## X29  26.74874627 20.031907            28.524938     6.077755e+01
## X30  27.33326457 21.646208            28.887728     7.166656e+01
## X31  21.03173673 16.898966            22.507060     5.741830e+01
## X32  25.18841578 17.677514            25.660150     7.846888e+01
## X33  27.76809586 19.948579            28.398287     1.176919e+02
## X34  23.47338826 20.239453            27.510775     1.122254e+02
## X35  18.22168617 20.153239            24.651457     7.381205e+01
## X36  13.76048113 16.744835            20.471068     3.998093e+01
## X37  16.67260482 16.616709            21.380330     2.993911e+01
## X38  14.76977894 14.122611            19.401825     2.625257e+01
## X39  18.17498282 12.850936            20.486945     3.629935e+01
## X40  17.24607073 13.209315            20.443112     3.145161e+01
## X41  17.71304502 12.971683            19.770366     3.728927e+01
## X42  17.09896465 10.950061            16.913336     2.580318e+01
## X43  17.46834141 12.438360            16.891458     4.253092e+01
## X44  13.62770966  9.167762            14.046263     1.992614e+01
## X45  12.23540195  9.873732            14.266276     1.817623e+01
## X46  11.33253800 13.208231            16.336706     1.828625e+01
## X47  10.18942497 11.689693            14.247289     2.097839e+01
## X48   9.73634353 12.047236            13.917883     2.256750e+01
## X49   9.54261120  7.310337            11.044618     1.164441e+01
## X50   8.84210766  9.607217            11.290537     1.141593e+01
## X51   9.31383776  8.719233            11.305478     1.126761e+01
## X52   8.04561212  8.584564            11.065273     1.096746e+01
## X53   8.42964251  8.612792            10.997698     9.112668e+00
## X54   8.52646073  6.893172            11.063971     1.048043e+01
## X55   8.80709300  5.981938            11.582240     8.795459e+00
## X56   8.54767671  5.239608            10.619010     9.166238e+00
## X57   8.35564457  6.526694            10.400166     9.532551e+00
```

```
## X58   8.28088704  7.778289            10.752109     9.001181e+00
## X59   9.93691026  6.580947            12.045502     1.220094e+01
## X60  10.39925497  7.280033            12.421356     1.158730e+01
## X61   9.42709518  7.345589            11.759824     9.376141e+00
## X62  10.53589063  8.111131            14.631562     1.169333e+01
## X63   9.25319202  7.953562            12.357239     1.011997e+01
## X64   8.72653359  5.523777            10.811770     8.540112e+00
## X65   7.76963071  6.672395             9.681742     7.642509e+00
## X66   9.36809736  7.394631            11.985620     8.701814e+00
## X67   7.90041582  5.520909             9.266963     8.154367e+00
## X68   8.20164609  5.888541             9.560579     8.782856e+00
## X69  10.80053804  6.149928            13.528622     1.023342e+01
## X70   9.67438410  7.404447            12.639719     9.816864e+00
## X71  10.57236980  8.967616            14.243778     1.204394e+01
## X72   9.94677451  9.447832            14.114508     1.306837e+01
## X73   9.38036083  8.786779            12.561776     1.011718e+01
## X74   9.16523597 12.055368            14.896436     9.129626e+00
## X75   8.93729162 10.515961            13.727319     9.197008e+00
## X76   7.51182633  9.386277            11.462126     7.726812e+00
## X77   8.67432622  8.074653            12.541840     7.775969e+00
## X78   9.43940202 10.947420            13.650766     1.190399e+01
## X79   9.95565213  9.426925            12.018623     1.276833e+01
## X80  10.13294286  6.570787            11.059238     1.591173e+01
## X81  11.79860138  9.447092            13.635208     2.375851e+01
## X82  13.50403276 12.505087            15.395618     3.789160e+01
## X83  19.45871597 14.781695            20.238334     6.604646e+01
## X84  15.92928691 10.364968            16.746109     4.087979e+01
## X85  13.91500450 11.398440            15.650707     2.960153e+01
## X86  13.64038278  9.186388            13.727335     2.407553e+01
## X87  12.60239651  9.396482            15.266181     1.306351e+01
## X88  15.56357742  7.877197            17.792523     1.420189e+01
## X89  15.67247401  7.267858            17.672027     1.427131e+01
## X90  13.70399388  9.443294            16.543262     1.253966e+01
## X91  13.17718516  9.537396            17.149426     1.130075e+01
## X92  13.03354570  6.739538            15.751829     1.102062e+01
## X93  12.95443743  9.143613            15.940662     1.140526e+01
## X94  12.08772162 11.817738            16.812348     1.188480e+01
## X95  10.66370997 13.009605            15.605157     1.233645e+01
## X96  10.24494276 11.584166            13.568254     1.137412e+01
## X97  12.49162249 12.168089            15.206709     1.337770e+01
## X98  11.51559738 11.888487            15.445613     1.474037e+01
## X99  13.29738576 13.131746            16.537721     1.719018e+01
## X100 13.76822897 12.887711            18.934401     1.324025e+01
## X101 12.89243202 11.723577            17.578861     1.264988e+01
## X102 12.80675115 11.750314            17.955894     1.199755e+01
## X103 11.32345462 13.038741            14.742272     1.226778e+01
## X104 12.99247500 13.827644            16.261297     1.649383e+01
## X105 14.97941021 13.645559            17.759550     2.119123e+01
## X106 14.71259055 13.632471            19.069788     1.787317e+01
## X107 10.64747890 10.304189            14.706709     1.152458e+01
## X108 10.23071104  8.389302            12.418927     1.296238e+01
## X109  8.17914372  9.052959             9.727603     1.838464e+01
## X110  9.55975447 12.442624            12.379684     3.521980e+01
## X111  8.82207708  9.291280             9.822888     3.047404e+01
```

```
## X112  9.10684416 11.073477            11.220664     3.870677e+01
## X113 12.41878546 12.632434            13.160500     7.500417e+01
## X114 12.89563211 13.523018            13.935192     7.563228e+01
## X115  7.36740801  7.978864             8.212151     2.808928e+01
## X116  5.22506217  6.837809             6.817884     1.736056e+01
## X117  8.29775988  7.203931             7.857840     2.166540e+01
## X118  6.45755114  6.064769             6.767510     1.211050e+01
## X119  6.94209447  5.693481             7.701801     6.166110e+00
## X120  6.41443206  5.095299             6.365242     6.619279e+00
## X121  4.06851510  5.927395             6.335053     9.153694e+00
## X122  4.61226348  4.823827             5.604438     6.421466e+00
## X123  6.57322914  6.646667             8.178640     6.325242e+00
## X124  6.36794720  5.292470             6.946846     5.656848e+00
## X125  5.49211962  4.700594             5.600084     9.100047e+00
## X126  6.20124383  6.132440             7.293836     5.980633e+00
## X127  8.21828849  6.533565             8.559518     7.507608e+00
## X128  5.91318705  6.058410             7.291107     5.945315e+00
## X129  7.13769805  6.435500             7.974822     5.770923e+00
## X130  9.56226545  6.080044             8.309713     8.269286e+00
## X131  9.89327374  7.072781            10.142448     1.193846e+01
## X132  7.13209533  7.482807             8.833842     1.051258e+01
## X133  6.07130762  7.692746             8.666733     7.493008e+00
## X134  6.34032845  5.269913             7.315000     6.370011e+00
## X135  5.13703762  5.631222             7.179960     4.269813e+00
## X136  6.23710381  4.699112             7.033808     4.849429e+00
## X137  5.95351406  5.360530             7.611140     4.913550e+00
## X138  5.75946537  4.160053             6.146660     4.199676e+00
## X139  5.24293741  5.502341             6.912986     4.336440e+00
## X140  5.56539271  5.134291             6.907682     3.670787e+00
## X141  4.91756286  4.933932             6.397422     3.232710e+00
## X142  3.65474809  4.832596             5.686680     1.923879e+00
## X143  3.52742718  3.374124             4.282539     1.981414e+00
## X144  3.71814874  2.813937             4.465279     2.596752e+00
## X145  4.55598930  3.748095             4.382112     1.915164e+00
## X146  4.74164816  4.198683             5.422915     2.851849e+00
## X147  5.01128709  3.727436             5.604550     2.563072e+00
## X148  4.86917151  3.305078             5.275452     2.648449e+00
## X149  7.44237263  3.747825             6.360030     5.929011e+00
## X150  8.02711939  4.117635             7.433853     6.005158e+00
## X151  7.70516003  4.701086             8.145566     5.367078e+00
## X152  5.02606863  4.542506             5.812024     2.858043e+00
## X153  4.49793720  3.303172             4.748166     1.807194e+00
## X154  3.99389030  7.958917             8.798371     4.559223e+00
## X155  5.82415860  8.505222             9.177298     4.560082e+00
## X156  6.39804877  7.550645             8.887142     5.306306e+00
## X157  4.38957107  4.182979             5.447821     1.548599e+00
## X158  4.51451688  3.913840             5.551857     1.744940e+00
## X159  5.57479639  1.922860             4.262506     1.181749e+00
## X160  3.42955940  2.858145             4.059581     6.235430e-01
## X161  2.39981287  2.812992             3.005942     4.022426e-01
## X162  2.18700397  2.525497             3.328803     6.095328e-01
## X163  6.37379951  6.150588             7.240562     2.154510e+00
## X164  4.56907143  4.508917             5.611774     1.384716e+00
## X165  3.77553016  3.434157             4.187749     9.069034e-01
```

```
## X166  4.71235248  5.365157             6.153478     1.124835e+00
## X167  2.93666741  3.756990             4.004932     5.908359e-01
## X168  2.31609795  1.764601             2.404822     2.724194e-01
## X169  2.60052422  1.754551             2.583438     3.156350e-01
## X170  2.06078039  1.497438             2.062861     2.145090e-01
## X171  2.36250144  1.936827             2.274188     1.634160e-01
## X172  2.05706134  0.926364             1.568440     1.575163e-01
## X173  1.89168341  1.829826             2.006807     1.932645e-01
## X174 -0.56223477  1.387506             1.135309     5.267963e-02
## X175  2.71143427  1.825329             2.486822     8.209707e-02
## X176  1.73824765  2.050748             2.324753     1.230490e-01
## X177  1.15056587  1.866318             2.000121     1.626666e-01
## X178  2.36044936  2.041276             2.710428     1.479294e-01
## X179  1.00100150  0.000000             1.001002     7.323789e-02
## X180  0.01340003 -1.417050            -1.000071     5.252392e-02
## X181  0.00000000  1.417019             1.417050     2.922185e-02
## X182  0.00000000  1.001002             1.001002     2.666667e-03
## X183  0.00000000  0.000000             0.000000     1.754938e-02
## X184  0.00000000  1.001002             1.001002     7.762821e-03
## X185  0.00000000  0.000000             0.000000     0.000000e+00
## X186  0.00000000  0.000000             0.000000     0.000000e+00
## X187  0.00000000  0.000000             0.000000     0.000000e+00
```

```
varImpPlot(rf.med)
```