Customer Churn Analysis

```
> ## Set up the directory
> setwd("C:\\Users\\Varsha\\OneDrive\\pro\\churn")

> ## Load the data
> data1 <- read.csv("C:\\Users\\Varsha\\OneDrive\\pro\\churn\\churn.csv")

> ## Install the required packages
> library(caret)
> library(rpart)
> library(C50)
> library(rattle)
> library(party)
> library(partykit)
> library(randomForest)
> library(ROCR)
> library(ggplot2)
> library(reshape2)
> library(car)
> library(corrplot)
> library(e1071)


> ## Know the Data
> dim(data1)
[1] 3333   21

> head(data1)
  State Account.Length Area.Code    Phone Int.l.Plan VMail.Plan VMail.Message
Day.Mins
1    KS            128       415 382-4657         no        yes             25
265.1
2    OH            107       415 371-7191         no        yes             26
161.6
3    NJ            137       415 358-1921         no         no              0
243.4
4    OH             84       408 375-9999        yes         no              0
299.4
5    OK             75       415 330-6626        yes         no              0
166.7
6    AL            118       510 391-8027        yes         no              0
223.4
```

```
  Day.Calls Day.Charge Eve.Mins Eve.Calls Eve.Charge Night.Mins Night.Calls N
ight.Charge
1       110      45.07    197.4        99      16.78      244.7          91
11.01
2       123      27.47    195.5       103      16.62      254.4         103
11.45
3       114      41.38    121.2       110      10.30      162.6         104
7.32
4        71      50.90     61.9        88       5.26      196.9          89
8.86
5       113      28.34    148.3       122      12.61      186.9         121
8.41
6        98      37.98    220.6       101      18.75      203.9         118
9.18
  Intl.Mins Intl.Calls Intl.Charge CustServ.Calls Churn.
1      10.0          3        2.70              1 False.
2      13.7          3        3.70              1 False.
3      12.2          5        3.29              0 False.
4       6.6          7        1.78              2 False.
5      10.1          3        2.73              3 False.
6       6.3          6        1.70              0 False.

> str(data1)
'data.frame':  3333 obs. of  21 variables:
 $ State         : Factor w/ 51 levels "AK","AL","AR",..: 17 36 32 36 37 2 20
25 19 50 ...
 $ Account.Length: int  128 107 137 84 75 118 121 147 117 141 ...
 $ Area.Code     : int  415 415 415 408 415 510 510 415 408 415 ...
 $ Phone         : Factor w/ 3333 levels "327-1058","327-1319",..: 1927 1576
1118 1708 111 2254 1048 81 292 118 ...
 $ Int.l.Plan    : Factor w/ 2 levels "no","yes": 1 1 1 2 2 2 1 2 1 2 ...
 $ VMail.Plan    : Factor w/ 2 levels "no","yes": 2 2 1 1 1 1 2 1 1 2 ...
 $ VMail.Message : int  25 26 0 0 0 0 24 0 0 37 ...
 $ Day.Mins      : num  265 162 243 299 167 ...
 $ Day.Calls     : int  110 123 114 71 113 98 88 79 97 84 ...
 $ Day.Charge    : num  45.1 27.5 41.4 50.9 28.3 ...
 $ Eve.Mins      : num  197.4 195.5 121.2 61.9 148.3 ...
 $ Eve.Calls     : int  99 103 110 88 122 101 108 94 80 111 ...
 $ Eve.Charge    : num  16.78 16.62 10.3 5.26 12.61 ...
 $ Night.Mins    : num  245 254 163 197 187 ...
 $ Night.Calls   : int  91 103 104 89 121 118 118 96 90 97 ...
 $ Night.Charge  : num  11.01 11.45 7.32 8.86 8.41 ...
 $ Intl.Mins     : num  10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 8.7 11.2 ...
 $ Intl.Calls    : int  3 3 5 7 3 6 7 6 4 5 ...
 $ Intl.Charge   : num  2.7 3.7 3.29 1.78 2.73 1.7 2.03 1.92 2.35 3.02 ...
```

```
 $ CustServ.Calls: int  1 1 0 2 3 0 3 0 1 0 ...
 $ Churn.        : Factor w/ 2 levels "False.","True.": 1 1 1 1 1 1 1 1 1 1 .
..
> colnames(data1)
 [1] "State"          "Account.Length" "Area.Code"      "Phone"          "Int
.l.Plan"
 [6] "VMail.Plan"     "VMail.Message"  "Day.Mins"       "Day.Calls"      "Day
.Charge"
[11] "Eve.Mins"       "Eve.Calls"      "Eve.Charge"     "Night.Mins"     "Nig
ht.Calls"
[16] "Night.Charge"   "Intl.Mins"      "Intl.Calls"     "Intl.Charge"    "Cus
tServ.Calls"
[21] "Churn."
> sum(is.na(data1))
[1] 0
> class(data1)
[1] "data.frame"


> ####################
> ## Data Munging ###
> ####################

> data1$Churn. <- as.integer(data1$Churn.)
> data1$Int.l.Plan <- as.integer(data1$Int.l.Plan)
> data1$VMail.Plan <- as.integer(data1$VMail.Plan)
> data1$Churn.[data1$Churn.==1] <- 0
> data1$Churn.[data1$Churn.==2] <- 1
> data1$Int.l.Plan[data1$Int.l.Plan==1] <- 0
> data1$Int.l.Plan[data1$Int.l.Plan==2] <- 1
> data1$VMail.Plan[data1$VMail.Plan==1] <- 0
> data1$VMail.Plan[data1$VMail.Plan==2] <- 1


> ############################
> ## Drop unwanted variable ###
> ############################

> data1$State <- NULL
> data1$Area.Code <- NULL
> data1$Phone <- NULL

> ##############################
> ## Exploratory Data Analysis ##
> ##############################
```

```
> summary(data1)
 Account.Length    Int.l.Plan        VMail.Plan     VMail.Message       Day.Mins       Day.Calls
 Min.   :  1.0   Min.   :0.00000   Min.   :0.0000   Min.   : 0.000   Min.   :  0.0   Min.   :  0.0
 1st Qu.: 74.0   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.: 0.000   1st Qu.:143.7   1st Qu.: 87.0
 Median :101.0   Median :0.00000   Median :0.0000   Median : 0.000   Median :179.4   Median :101.0
 Mean   :101.1   Mean   :0.09691   Mean   :0.2766   Mean   : 8.099   Mean   :179.8   Mean   :100.4
 3rd Qu.:127.0   3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:20.000   3rd Qu.:216.4   3rd Qu.:114.0
 Max.   :243.0   Max.   :1.00000   Max.   :1.0000   Max.   :51.000   Max.   :350.8   Max.   :165.0
   Day.Charge       Eve.Mins       Eve.Calls       Eve.Charge      Night.Mins     Night.Calls
 Min.   : 0.00   Min.   :  0.0   Min.   :  0.0   Min.   : 0.00   Min.   : 23.2   Min.   : 33.0
 1st Qu.:24.43   1st Qu.:166.6   1st Qu.: 87.0   1st Qu.:14.16   1st Qu.:167.0   1st Qu.: 87.0
 Median :30.50   Median :201.4   Median :100.0   Median :17.12   Median :201.2   Median :100.0
 Mean   :30.56   Mean   :201.0   Mean   :100.1   Mean   :17.08   Mean   :200.9   Mean   :100.1
 3rd Qu.:36.79   3rd Qu.:235.3   3rd Qu.:114.0   3rd Qu.:20.00   3rd Qu.:235.3   3rd Qu.:113.0
 Max.   :59.64   Max.   :363.7   Max.   :170.0   Max.   :30.91   Max.   :395.0   Max.   :175.0
  Night.Charge      Intl.Mins       Intl.Calls      Intl.Charge    CustServ.Calls      Churn.
 Min.   : 1.040   Min.   : 0.00   Min.   : 0.000   Min.   :0.000   Min.   :0.000   Min.   :0.0000
 1st Qu.: 7.520   1st Qu.: 8.50   1st Qu.: 3.000   1st Qu.:2.300   1st Qu.:1.000   1st Qu.:0.0000
 Median : 9.050   Median :10.30   Median : 4.000   Median :2.780   Median :1.000   Median :0.0000
 Mean   : 9.039   Mean   :10.24   Mean   : 4.479   Mean   :2.765   Mean   :1.563   Mean   :0.1449
 3rd Qu.:10.590   3rd Qu.:12.10   3rd Qu.: 6.000   3rd Qu.:3.270   3rd Qu.:2.000   3rd Qu.:0.0000
 Max.   :17.770   Max.   :20.00   Max.   :20.000   Max.   :5.400   Max.   :9.000   Max.   :1.0000
```

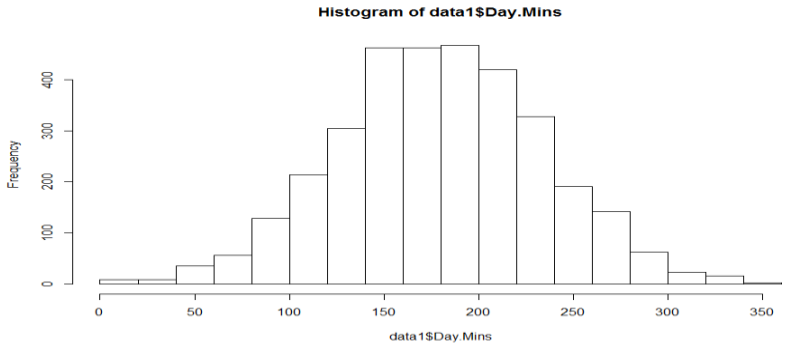
```
> sapply(data1, sd)
Account.Length     Int.l.Plan     VMail.Plan  VMail.Message       Day.Mins      Day.Calls     Day.Charge
    39.8221059      0.2958791      0.4473979     13.6883654     54.4673892     20.0690842      9.2594346
     Eve.Mins      Eve.Calls     Eve.Charge     Night.Mins   Night.Calls  Night.Charge      Intl.Mins
    50.7138444     19.9226253      4.3106676     50.5738470     19.5686093      2.2758728      2.7918395
   Intl.Calls    Intl.Charge CustServ.Calls         Churn.
    2.4612143      0.7537726      1.3154910      0.3520674
```

```
> cormat <- round(cor(data1), digits = 2)
> cormat
```

	Account.Length	Int.l.Plan	VMail.Plan	VMail.Message	Day.Mins	Day.Calls	Day.Charge	Eve.Mins	Eve.Calls	Eve.Charge	Night.Mins	Night.Calls
Account.Length	1.00	0.02	0.00	0.00	0.01	0.04	0.01	-0.01	0.02	-0.01	-0.01	-0.01
Int.l.Plan	0.02	1.00	0.01	0.01	0.05	0.00	0.05	0.02	0.01	0.02	-0.03	0.01
VMail.Plan	0.00	0.01	1.00	0.96	0.00	-0.01	0.00	0.02	-0.01	0.02	0.01	0.02
VMail.Message	0.00	0.01	0.96	1.00	0.00	-0.01	0.00	0.02	-0.01	0.02	0.01	0.01
Day.Mins	0.01	0.05	0.00	0.00	1.00	0.01	1.00	0.01	0.02	0.01	0.00	0.02
Day.Calls	0.04	0.00	-0.01	-0.01	0.01	1.00	0.01	-0.02	0.01	-0.02	0.02	-0.02
Day.Charge	0.01	0.05	0.00	0.00	1.00	0.01	1.00	0.01	0.02	0.01	0.00	0.02
Eve.Mins	-0.01	0.02	0.02	0.02	0.01	-0.02	0.01	1.00	-0.01	1.00	-0.01	0.01
Eve.Calls	0.02	0.01	-0.01	-0.01	0.02	0.01	0.02	-0.01	1.00	-0.01	0.00	0.01
Eve.Charge	-0.01	0.02	0.02	0.02	0.01	-0.02	0.01	1.00	-0.01	1.00	-0.01	0.01
Night.Mins	-0.01	-0.03	0.01	0.01	0.00	0.02	0.00	-0.01	0.00	-0.01	1.00	0.01
Night.Calls	-0.01	0.01	0.02	0.01	0.02	-0.02	0.02	0.01	0.01	0.01	0.01	1.00
Night.Charge	-0.01	-0.03	0.01	0.01	0.00	0.02	0.00	-0.01	0.00	-0.01	1.00	0.01
Intl.Mins	0.01	0.05	0.00	0.00	-0.01	0.02	-0.01	-0.01	0.01	-0.01	-0.02	-0.01
Intl.Calls	0.02	0.02	0.01	0.01	0.01	0.00	0.01	0.00	0.02	0.00	-0.01	0.00
Intl.Charge	0.01	0.05	0.00	0.00	-0.01	0.02	-0.01	-0.01	0.01	-0.01	-0.02	-0.01
CustServ.Calls	0.00	-0.02	-0.02	-0.01	-0.01	-0.02	-0.01	-0.01	0.00	-0.01	-0.01	-0.01
Churn.	0.02	0.26	-0.10	-0.09	0.21	0.02	0.21	0.09	0.01	0.09	0.04	0.01

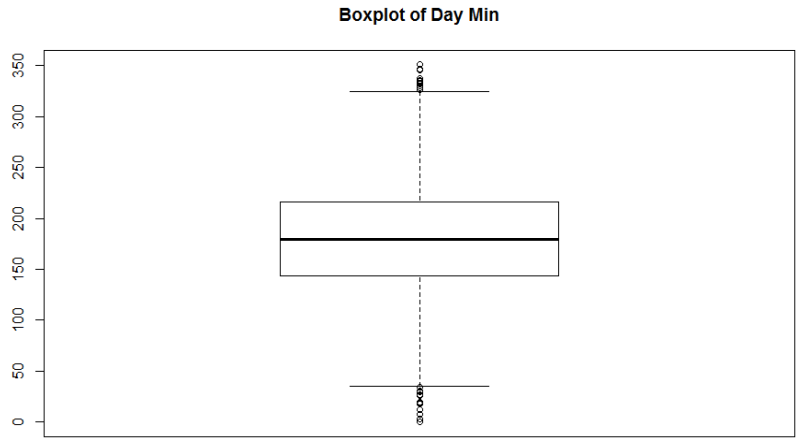
	Night.Charge	Intl.Mins	Intl.Calls	Intl.Charge	CustServ.Calls	Churn.
Account.Length	-0.01	0.01	0.02	0.01	0.00	0.02
Int.l.Plan	-0.03	0.05	0.02	0.05	-0.02	0.26
VMail.Plan	0.01	0.00	0.01	0.00	-0.02	-0.10
VMail.Message	0.01	0.00	0.01	0.00	-0.01	-0.09
Day.Mins	0.00	-0.01	0.01	-0.01	-0.01	0.21
Day.Calls	0.02	0.02	0.00	0.02	-0.02	0.02
Day.Charge	0.00	-0.01	0.01	-0.01	-0.01	0.21
Eve.Mins	-0.01	-0.01	0.00	-0.01	-0.01	0.09
Eve.Calls	0.00	0.01	0.02	0.01	0.00	0.01
Eve.Charge	-0.01	-0.01	0.00	-0.01	-0.01	0.09
Night.Mins	1.00	-0.02	-0.01	-0.02	-0.01	0.04
Night.Calls	0.01	-0.01	0.00	-0.01	-0.01	0.01
Night.Charge	1.00	-0.02	-0.01	-0.02	-0.01	0.04
Intl.Mins	-0.02	1.00	0.03	1.00	-0.01	0.07
Intl.Calls	-0.01	0.03	1.00	0.03	-0.02	-0.05
Intl.Charge	-0.02	1.00	0.03	1.00	-0.01	0.07
CustServ.Calls	-0.01	-0.01	-0.02	-0.01	1.00	0.21
Churn.	0.04	0.07	-0.05	0.07	0.21	1.00

```
> ## Histogram of day minutes
> plot.new()
> hist(data1$Day.Mins)
```




```
> plot.new()
> boxplot(data1$Day.Mins)
```

```
> title("Boxplot of Day Min")
```




```
#######################################
> ## split dataset into train and test ##
> #######################################
>
> set.seed(1234)
> ## 70% training and 30% testing data
> ind <- sample(2, nrow(data1), replace = TRUE, prob=c(0.7,0.3))
> train <- data1[ind==1,]
> test <- data1[ind==2,]
>
> ##########################################################################
#################################
>
> ## Model 1 ##
> ## Logistic Regression ##
>
> ## select the variables to use based on forward selection procedure
> ## Lower AIC indicates better model
>
> # forward Elimination
>
> mod1 <- glm(Churn.~1, data = train)
> biggest <- formula(glm(Churn.~., data = train))
> biggest
Churn. ~ Account.Length + Int.l.Plan + VMail.Plan + VMail.Message +
    Day.Mins + Day.Calls + Day.Charge + Eve.Mins + Eve.Calls +
    Eve.Charge + Night.Mins + Night.Calls + Night.Charge + Intl.Mins +
```

```
    Intl.Calls + Intl.Charge + CustServ.Calls
> forwardTest <- step(mod1, direction = "forward", scope = biggest)
Start:  AIC=1690.76
Churn. ~ 1

                 Df Deviance    AIC
+ Int.l.Plan      1   266.63 1556.6
+ CustServ.Calls  1   270.41 1589.8
+ Day.Charge      1   273.77 1619.0
+ Day.Mins        1   273.77 1619.0
+ VMail.Plan      1   279.50 1667.9
+ VMail.Message   1   280.30 1674.7
+ Eve.Mins        1   280.40 1675.5
+ Eve.Charge      1   280.40 1675.5
+ Intl.Charge     1   280.98 1680.4
+ Intl.Mins       1   280.98 1680.4
+ Intl.Calls      1   281.61 1685.7
+ Night.Charge    1   281.97 1688.7
+ Night.Mins      1   281.97 1688.7
+ Account.Length  1   282.20 1690.6
<none>                282.45 1690.8
+ Eve.Calls       1   282.40 1692.3
+ Night.Calls     1   282.43 1692.6
+ Day.Calls       1   282.44 1692.7

Step:  AIC=1556.56
Churn. ~ Int.l.Plan

                 Df Deviance    AIC
+ CustServ.Calls  1   253.02 1434.8
+ Day.Charge      1   258.62 1486.5
+ Day.Mins        1   258.62 1486.5
+ VMail.Plan      1   263.78 1533.2
+ VMail.Message   1   264.54 1540.0
+ Eve.Mins        1   264.85 1542.8
+ Eve.Charge      1   264.85 1542.8
+ Intl.Charge     1   265.35 1547.2
+ Intl.Mins       1   265.35 1547.2
+ Intl.Calls      1   265.73 1550.6
+ Night.Charge    1   266.04 1553.3
+ Night.Mins      1   266.04 1553.3
<none>                266.63 1556.6
+ Account.Length  1   266.43 1556.8
+ Eve.Calls       1   266.60 1558.3
+ Night.Calls     1   266.61 1558.4
```

```
+ Day.Calls       1   266.63 1558.6

Step:  AIC=1434.85
Churn. ~ Int.l.Plan + CustServ.Calls

                 Df Deviance    AIC
+ Day.Charge      1   244.48 1355.8
+ Day.Mins        1   244.48 1355.8
+ VMail.Plan      1   250.26 1410.9
+ VMail.Message   1   250.94 1417.3
+ Eve.Mins        1   251.04 1418.2
+ Eve.Charge      1   251.04 1418.2
+ Intl.Charge     1   251.59 1423.5
+ Intl.Mins       1   251.60 1423.5
+ Intl.Calls      1   252.31 1430.2
+ Night.Charge    1   252.49 1431.9
+ Night.Mins      1   252.49 1431.9
<none>                253.02 1434.8
+ Account.Length  1   252.84 1435.2
+ Eve.Calls       1   253.00 1436.7
+ Night.Calls     1   253.01 1436.8
+ Day.Calls       1   253.01 1436.8

Step:  AIC=1355.78
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge

                 Df Deviance    AIC
+ VMail.Plan      1   241.74 1331.1
+ Eve.Mins        1   242.36 1337.1
+ Eve.Charge      1   242.36 1337.1
+ VMail.Message   1   242.38 1337.4
+ Intl.Charge     1   242.93 1342.8
+ Intl.Mins       1   242.94 1342.8
+ Intl.Calls      1   243.65 1349.7
+ Night.Charge    1   243.98 1352.9
+ Night.Mins      1   243.98 1352.9
<none>                244.48 1355.8
+ Account.Length  1   244.33 1356.3
+ Night.Calls     1   244.46 1357.5
+ Eve.Calls       1   244.47 1357.6
+ Day.Calls       1   244.47 1357.6
+ Day.Mins        1   244.48 1357.8

Step:  AIC=1331.14
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.Plan
```

```
                 Df Deviance    AIC
+ Eve.Mins        1   239.48 1310.9
+ Eve.Charge      1   239.48 1310.9
+ Intl.Charge     1   240.18 1317.8
+ Intl.Mins       1   240.18 1317.8
+ Intl.Calls      1   240.93 1325.1
+ Night.Charge    1   241.19 1327.7
+ Night.Mins      1   241.19 1327.7
+ VMail.Message   1   241.52 1330.9
<none>                241.74 1331.1
+ Account.Length  1   241.58 1331.6
+ Night.Calls     1   241.71 1332.9
+ Day.Calls       1   241.73 1333.0
+ Eve.Calls       1   241.73 1333.0
+ Day.Mins        1   241.74 1333.1

Step:  AIC=1310.91
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.Plan +
    Eve.Mins

                 Df Deviance    AIC
+ Intl.Charge     1   237.82 1296.5
+ Intl.Mins       1   237.82 1296.5
+ Intl.Calls      1   238.63 1304.6
+ Night.Charge    1   238.87 1306.9
+ Night.Mins      1   238.87 1306.9
+ VMail.Message   1   239.25 1310.6
<none>                239.48 1310.9
+ Account.Length  1   239.30 1311.2
+ Eve.Charge      1   239.44 1312.5
+ Day.Calls       1   239.44 1312.6
+ Night.Calls     1   239.45 1312.6
+ Eve.Calls       1   239.46 1312.7
+ Day.Mins        1   239.47 1312.8

Step:  AIC=1296.54
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.Plan +
    Eve.Mins + Intl.Charge

                 Df Deviance    AIC
+ Intl.Calls      1   236.93 1289.6
+ Night.Charge    1   237.22 1292.5
+ Night.Mins      1   237.22 1292.5
+ VMail.Message   1   237.60 1296.3
```

```
<none>                237.82 1296.5
+ Account.Length  1   237.65 1296.8
+ Eve.Charge      1   237.78 1298.2
+ Night.Calls     1   237.79 1298.2
+ Day.Calls       1   237.80 1298.3
+ Eve.Calls       1   237.80 1298.3
+ Intl.Mins       1   237.81 1298.4
+ Day.Mins        1   237.82 1298.5

Step:  AIC=1289.63
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.Plan +
    Eve.Mins + Intl.Charge + Intl.Calls

                 Df Deviance    AIC
+ Night.Charge    1   236.32 1285.6
+ Night.Mins      1   236.32 1285.6
+ VMail.Message   1   236.68 1289.1
<none>                236.93 1289.6
+ Account.Length  1   236.73 1289.7
+ Eve.Charge      1   236.88 1291.2
+ Night.Calls     1   236.90 1291.3
+ Eve.Calls       1   236.90 1291.3
+ Day.Calls       1   236.91 1291.4
+ Intl.Mins       1   236.91 1291.4
+ Day.Mins        1   236.92 1291.6

Step:  AIC=1285.56
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.Plan +
    Eve.Mins + Intl.Charge + Intl.Calls + Night.Charge

                 Df Deviance    AIC
+ VMail.Message   1   236.08 1285.2
+ Night.Mins      1   236.10 1285.4
<none>                236.32 1285.6
+ Account.Length  1   236.12 1285.6
+ Eve.Charge      1   236.28 1287.1
+ Night.Calls     1   236.28 1287.2
+ Eve.Calls       1   236.29 1287.3
+ Intl.Mins       1   236.30 1287.4
+ Day.Calls       1   236.31 1287.4
+ Day.Mins        1   236.31 1287.5

Step:  AIC=1285.19
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.Plan +
    Eve.Mins + Intl.Charge + Intl.Calls + Night.Charge + VMail.Message
```

```
                  Df Deviance    AIC
+ Night.Mins       1   235.87 1285.0
+ Account.Length   1   235.87 1285.0
<none>                 236.08 1285.2
+ Eve.Charge       1   236.03 1286.7
+ Night.Calls      1   236.05 1286.9
+ Eve.Calls        1   236.06 1286.9
+ Intl.Mins        1   236.06 1287.0
+ Day.Calls        1   236.07 1287.1
+ Day.Mins         1   236.08 1287.1

Step:  AIC=1285
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.Plan +
    Eve.Mins + Intl.Charge + Intl.Calls + Night.Charge + VMail.Message +
    Night.Mins

                  Df Deviance    AIC
+ Account.Length   1   235.64 1284.8
<none>                 235.87 1285.0
+ Eve.Charge       1   235.81 1286.5
+ Night.Calls      1   235.84 1286.7
+ Eve.Calls        1   235.84 1286.8
+ Intl.Mins        1   235.84 1286.8
+ Day.Calls        1   235.85 1286.9
+ Day.Mins         1   235.86 1287.0

Step:  AIC=1284.78
Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.Plan +
    Eve.Mins + Intl.Charge + Intl.Calls + Night.Charge + VMail.Message +
    Night.Mins + Account.Length

               Df Deviance    AIC
<none>              235.64 1284.8
+ Eve.Charge    1   235.58 1286.2
+ Night.Calls   1   235.62 1286.5
+ Intl.Mins     1   235.62 1286.5
+ Eve.Calls     1   235.62 1286.6
+ Day.Calls     1   235.63 1286.7
+ Day.Mins      1   235.64 1286.8
> modlogit <- glm(Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge + VMail.P
lan +
+                   Eve.Mins + Intl.Charge + Intl.Calls + Night.Charge + VMai
l.Message +
```

```
+                   Night.Mins + Account.Length, family = "binomial", data =
train)
> summary(modlogit)

Call:
glm(formula = Churn. ~ Int.l.Plan + CustServ.Calls + Day.Charge +
    VMail.Plan + Eve.Mins + Intl.Charge + Intl.Calls + Night.Charge +
    VMail.Message + Night.Mins + Account.Length, family = "binomial",
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1062  -0.5139  -0.3405  -0.1956   3.2758

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)    -8.342328   0.647462 -12.885  < 2e-16 ***
Int.l.Plan      1.939548   0.172960  11.214  < 2e-16 ***
CustServ.Calls  0.532979   0.047360  11.254  < 2e-16 ***
Day.Charge      0.070880   0.007650   9.266  < 2e-16 ***
VMail.Plan     -2.386813   0.705752  -3.382  0.00072 ***
Eve.Mins        0.007373   0.001366   5.398 6.73e-08 ***
Intl.Charge     0.393236   0.092140   4.268 1.97e-05 ***
Intl.Calls     -0.098026   0.030421  -3.222  0.00127 **
Night.Charge   29.759389  23.383054   1.273  0.20313
VMail.Message   0.047105   0.021977   2.143  0.03208 *
Night.Mins     -1.335564   1.052259  -1.269  0.20436
Account.Length  0.002179   0.001662   1.312  0.18965
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1903.3  on 2361  degrees of freedom
Residual deviance: 1510.0  on 2350  degrees of freedom
AIC: 1534

Number of Fisher Scoring iterations: 6

>
> #influence Plot (clearly shows outliers)
> influenceIndexPlot(modlogit, vars = c("cook", "hat"), id.n = 3)
>
> ##confidence interval
> confint(modlogit)
```

```
Waiting for profiling to be done...
                      2.5 %       97.5 %
(Intercept)     -9.634422663 -7.094929687
Int.l.Plan       1.600748272  2.279482698
CustServ.Calls   0.440944240  0.626777995
Day.Charge       0.056051046  0.086055834
VMail.Plan      -3.819824854 -1.048889316
Eve.Mins         0.004713108  0.010070673
Intl.Charge      0.214070724  0.575458129
Intl.Calls      -0.158834259 -0.039534497
Night.Charge   -16.039683432 75.685210773
VMail.Message    0.004539979  0.090832435
Night.Mins      -3.402259805  0.725444769
Account.Length  -0.001076547  0.005440962
>

> # put the coefficients and confidence interval in a format onto a useful sc
ale
> exp(modlogit$coefficients)
   (Intercept)     Int.l.Plan CustServ.Calls     Day.Charge     VMail.Plan
Eve.Mins
  2.382171e-04   6.955610e+00   1.704001e+00   1.073453e+00   9.192219e-02
1.007401e+00
   Intl.Charge     Intl.Calls   Night.Charge  VMail.Message     Night.Mins Ac
count.Length
  1.481768e+00   9.066253e-01   8.401143e+12   1.048232e+00   2.630098e-01
1.002182e+00

> exp(confint(modlogit))
Waiting for profiling to be done...
                     2.5 %       97.5 %
(Intercept)    6.543700e-05 8.292991e-04
Int.l.Plan     4.956740e+00 9.771624e+00
CustServ.Calls 1.554174e+00 1.871571e+00
Day.Charge     1.057652e+00 1.089867e+00
VMail.Plan     2.193164e-02 3.503266e-01
Eve.Mins       1.004724e+00 1.010122e+00
Intl.Charge    1.238710e+00 1.777945e+00
Intl.Calls     8.531377e-01 9.612368e-01
Night.Charge   1.081568e-07 7.407461e+32
VMail.Message  1.004550e+00 1.095085e+00
Night.Mins     3.329794e-02 2.065650e+00
Account.Length 9.989240e-01 1.005456e+00
>
> ## odds ratio only
```

```
> exp(coef(modlogit))
   (Intercept)     Int.l.Plan CustServ.Calls     Day.Charge     VMail.Plan
Eve.Mins
  2.382171e-04   6.955610e+00   1.704001e+00   1.073453e+00   9.192219e-02
1.007401e+00
   Intl.Charge     Intl.Calls   Night.Charge  VMail.Message     Night.Mins Ac
count.Length
  1.481768e+00   9.066253e-01   8.401143e+12   1.048232e+00   2.630098e-01
1.002182e+00
>
> ## odds ratio and 95% CI
> exp(cbind(OR=coef(modlogit), confint(modlogit)))
Waiting for profiling to be done...
                          OR        2.5 %       97.5 %
(Intercept)    2.382171e-04 6.543700e-05 8.292991e-04
Int.l.Plan     6.955610e+00 4.956740e+00 9.771624e+00
CustServ.Calls 1.704001e+00 1.554174e+00 1.871571e+00
Day.Charge     1.073453e+00 1.057652e+00 1.089867e+00
VMail.Plan     9.192219e-02 2.193164e-02 3.503266e-01
Eve.Mins       1.007401e+00 1.004724e+00 1.010122e+00
Intl.Charge    1.481768e+00 1.238710e+00 1.777945e+00
Intl.Calls     9.066253e-01 8.531377e-01 9.612368e-01
Night.Charge   8.401143e+12 1.081568e-07 7.407461e+32
VMail.Message  1.048232e+00 1.004550e+00 1.095085e+00
Night.Mins     2.630098e-01 3.329794e-02 2.065650e+00
Account.Length 1.002182e+00 9.989240e-01 1.005456e+00
>
>
> ##########################################################################
##################################
>
> ## Model 2 ##
> ## Support Vector Machine ##
>
> svmModel <- svm(Churn.~., data=train, gamma=0.1, cost=1)
> print(svmModel)

Call:
svm(formula = Churn. ~ ., data = train, gamma = 0.1, cost = 1)


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  radial
       cost:  1
```

```
      gamma:  0.1
    epsilon:  0.1


Number of Support Vectors:  1450

> summary(svmModel)

Call:
svm(formula = Churn. ~ ., data = train, gamma = 0.1, cost = 1)


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.1
    epsilon:  0.1


Number of Support Vectors:  1450




>
>
> ################################################################################
################################
>
> ## Model 3 ##
> ## Random Forest ##
>
> randomForestModel <-randomForest(Churn.~., data = train, ntree=500, mtry=5,
importance=TRUE)
> print(randomForestModel)

Call:
 randomForest(formula = Churn. ~ ., data = train, ntree = 500,      mtry = 5,
importance = TRUE)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 5
```

```
          Mean of squared residuals: 0.04275197
                    % Var explained: 64.25
> importance(randomForestModel)
                  %IncMSE IncNodePurity
Account.Length -0.2677925      7.915987
Int.l.Plan     69.9034825     19.152294
VMail.Plan     21.0979868      5.651569
VMail.Message  22.2770010      8.597558
Day.Mins       37.1338488     37.043714
Day.Calls       1.3440773      7.052715
Day.Charge     33.6039234     34.933099
Eve.Mins       22.3704674     18.534550
Eve.Calls      -0.2421886      6.737962
Eve.Charge     23.2872859     19.136085
Night.Mins     14.2618489      9.182289
Night.Calls     1.1920180      7.717779
Night.Charge   14.1502290      9.361246
Intl.Mins      22.4014940     11.211129
Intl.Calls     46.6956250     16.201543
Intl.Charge    23.7686652     11.822977
CustServ.Calls 99.7362521     37.937501
>
> plot.new()
> varImpPlot(randomForestModel, type = 1, pch = 19, col=1, cex=1.0, main = ""
)
> abline(v=45, col="blue")
>
> plot.new()
> varImpPlot(randomForestModel, type = 2, pch=19, col=1, cex=1.0, main = "")
> abline(v=45, col="blue")
>
> ###########################################################################
#################################
>
> ## Model 4 ##
> ## Knowledge Discovery: Build a decision tree using C5.0 for churn ##
>
> # the decision variable class must be converted into a factor
> # the variable in order for C50 to process correctly
>
> data1$Churn. <- as.factor(data1$Churn.)
>
> #Run the C50 agorithm for decision tree
>
> c50_tree <- C5.0(Churn.~., data = data1)
```

```
>
> #display the summary
> summary(c50_tree)

Call:
C5.0.formula(formula = Churn. ~ ., data = data1)


C5.0 [Release 2.07 GPL Edition]       Wed Nov 09 12:11:41 2016
-------------------------------

Class specified by attribute `outcome'

Read 3333 cases (18 attributes) from undefined.data

Decision tree:

Day.Mins > 264.4:
:...VMail.Plan > 0:
:   :...Int.l.Plan <= 0: 0 (45/1)
:   :   Int.l.Plan > 0: 1 (8/3)
:   VMail.Plan <= 0:
:   :...Eve.Mins > 187.7:
:       :...Night.Mins > 126.9: 1 (94/1)
:       :   Night.Mins <= 126.9:
:       :   :...Day.Mins <= 277: 0 (4)
:       :       Day.Mins > 277: 1 (3)
:       Eve.Mins <= 187.7:
:       :...Eve.Charge <= 12.26: 0 (15/1)
:           Eve.Charge > 12.26:
:           :...Day.Mins <= 277:
:               :...Night.Mins <= 224.8: 0 (13)
:               :   Night.Mins > 224.8: 1 (5/1)
:               Day.Mins > 277:
:               :...Night.Mins > 151.9: 1 (18)
:                   Night.Mins <= 151.9:
:                   :...Account.Length <= 123: 0 (4)
:                       Account.Length > 123: 1 (2)
Day.Mins <= 264.4:
:...CustServ.Calls > 3:
    :...Day.Mins <= 160.2:
    :   :...Eve.Charge <= 19.83: 1 (79/3)
    :   :   Eve.Charge > 19.83:
    :   :   :...Day.Mins <= 120.5: 1 (10)
    :   :       Day.Mins > 120.5: 0 (13/3)
```

```
:   Day.Mins > 160.2:
:   :...Eve.Charge <= 12.05:
:       :...Eve.Calls <= 125: 1 (16/2)
:       :   Eve.Calls > 125: 0 (3)
:       Eve.Charge > 12.05:
:       :...Day.Mins <= 175.7:
:           :...Eve.Mins <= 212.1: 1 (16/2)
:           :   Eve.Mins > 212.1: 0 (18)
:           Day.Mins > 175.7:
:           :...Int.l.Plan <= 0: 0 (83/5)
:               Int.l.Plan > 0:
:               :...Intl.Calls <= 3: 1 (4)
:                   Intl.Calls > 3: 0 (9/1)
:  CustServ.Calls <= 3:
:  :...Int.l.Plan > 0:
:      :...Intl.Calls <= 2: 1 (51)
:      :   Intl.Calls > 2:
:      :   :...Intl.Mins <= 13.1: 0 (173/7)
:      :       Intl.Mins > 13.1: 1 (43)
:      Int.l.Plan <= 0:
:      :...Day.Mins <= 223.2: 0 (2221/60)
:          Day.Mins > 223.2:
:          :...Eve.Charge <= 20.5: 0 (295/22)
:              Eve.Charge > 20.5:
:              :...VMail.Plan > 0: 0 (20)
:                  VMail.Plan <= 0:
:                  :...Night.Mins > 174.2: 1 (50/8)
:                      Night.Mins <= 174.2:
:                      :...Day.Mins <= 246.6: 0 (12)
:                          Day.Mins > 246.6:
:                          :...Day.Charge <= 43.33: 1 (4)
:                              Day.Charge > 43.33: 0 (2)


Evaluation on training data (3333 cases):

           Decision Tree
         ----------------
         Size      Errors

           31  120( 3.6%)   <<


          (a)   (b)    <-classified as
         ----  ----
```

```
          2830    20    (a): class 0
           100   383    (b): class 1


        Attribute usage:

        100.00% Day.Mins
         93.67% CustServ.Calls
         90.61% Int.l.Plan
         20.73% Eve.Charge
          8.97% VMail.Plan
          8.40% Intl.Calls
          6.48% Intl.Mins
          6.33% Night.Mins
          5.76% Eve.Mins
          0.57% Eve.Calls
          0.18% Account.Length
          0.18% Day.Charge


Time: 0.1 secs

> C5imp(c50_tree, metric = "usage")
               Overall
Day.Mins        100.00
CustServ.Calls   93.67
Int.l.Plan       90.61
Eve.Charge       20.73
VMail.Plan        8.97
Intl.Calls        8.40
Intl.Mins         6.48
Night.Mins        6.33
Eve.Mins          5.76
Eve.Calls         0.57
Account.Length    0.18
Day.Charge        0.18
VMail.Message     0.00
Day.Calls         0.00
Night.Calls       0.00
Night.Charge      0.00
Intl.Charge       0.00
> C5imp(c50_tree, metric = "splits")
                Overall
Day.Mins       26.666667
Eve.Charge     13.333333
```

```
Night.Mins     13.333333
Int.l.Plan     10.000000
Eve.Mins        6.666667
Intl.Calls      6.666667
VMail.Plan      6.666667
Account.Length  3.333333
CustServ.Calls  3.333333
Day.Charge      3.333333
Eve.Calls       3.333333
Intl.Mins       3.333333
VMail.Message   0.000000
Day.Calls       0.000000
Night.Calls     0.000000
Night.Charge    0.000000
Intl.Charge     0.000000
>
> ## run the C50 algorithm and show the decision rules
> C50_rule_result <- C5.0(Churn.~., data = data1, rules=TRUE)
> summary(C50_rule_result)

Call:
C5.0.formula(formula = Churn. ~ ., data = data1, rules = TRUE)


C5.0 [Release 2.07 GPL Edition]       Wed Nov 09 12:11:42 2016
-------------------------------

Class specified by attribute `outcome'

Read 3333 cases (18 attributes) from undefined.data

Rules:

Rule 1: (2221/60, lift 1.1)
        Int.l.Plan <= 0
        Day.Mins <= 223.2
        CustServ.Calls <= 3
        ->  class 0  [0.973]

Rule 2: (45/1, lift 1.1)
        Int.l.Plan <= 0
        VMail.Plan > 0
        Day.Mins > 264.4
        ->  class 0  [0.957]
```

```
Rule 3: (1972/87, lift 1.1)
        Day.Mins <= 264.4
        Intl.Mins <= 13.1
        Intl.Calls > 2
        CustServ.Calls <= 3
        ->  class 0  [0.955]

Rule 4: (1223/95, lift 1.1)
        Int.l.Plan <= 0
        Day.Mins > 175.7
        Day.Mins <= 264.4
        Eve.Charge > 12.05
        ->  class 0  [0.922]

Rule 5: (162/14, lift 1.1)
        VMail.Plan <= 0
        Night.Mins <= 126.9
        ->  class 0  [0.909]

Rule 6: (640/60, lift 1.1)
        Day.Mins <= 175.7
        Eve.Mins > 212.1
        ->  class 0  [0.905]

Rule 7: (1324/158, lift 1.0)
        Eve.Mins <= 187.7
        ->  class 0  [0.880]

Rule 8: (66, lift 6.8)
        VMail.Plan <= 0
        Day.Mins > 277
        Eve.Charge > 12.26
        Night.Mins > 151.9
        ->  class 1  [0.985]

Rule 9: (60, lift 6.8)
        Int.l.Plan > 0
        Intl.Calls <= 2
        ->  class 1  [0.984]

Rule 10: (57, lift 6.8)
        Int.l.Plan > 0
        Intl.Mins > 13.1
        ->  class 1  [0.983]
```

```
Rule 11: (32, lift 6.7)
        Day.Mins <= 120.5
        CustServ.Calls > 3
        ->  class 1  [0.971]

Rule 12: (48/1, lift 6.6)
        VMail.Plan <= 0
        Day.Mins > 264.4
        Eve.Charge > 12.26
        Night.Mins > 224.8
        ->  class 1  [0.960]

Rule 13: (23, lift 6.6)
        Account.Length > 123
        VMail.Plan <= 0
        Day.Mins > 277
        ->  class 1  [0.960]

Rule 14: (79/3, lift 6.6)
        Day.Mins <= 160.2
        Eve.Charge <= 19.83
        CustServ.Calls > 3
        ->  class 1  [0.951]

Rule 15: (43/2, lift 6.4)
        Int.l.Plan <= 0
        VMail.Plan <= 0
        Day.Mins > 246.6
        Eve.Charge > 20.5
        ->  class 1  [0.933]

Rule 16: (84/5, lift 6.4)
        Day.Mins <= 175.7
        Eve.Mins <= 212.1
        CustServ.Calls > 3
        ->  class 1  [0.930]

Rule 17: (28/2, lift 6.2)
        Day.Mins <= 264.4
        Eve.Calls <= 125
        Eve.Charge <= 12.05
        CustServ.Calls > 3
        ->  class 1  [0.900]

Rule 18: (78/8, lift 6.1)
```

```
        VMail.Plan <= 0
        Day.Mins > 223.2
        Eve.Charge > 20.5
        Night.Mins > 174.2
        ->  class 1  [0.888]

Rule 19: (143/23, lift 5.8)
        VMail.Plan <= 0
        Day.Mins > 264.4
        Eve.Charge > 12.26
        ->  class 1  [0.834]

Rule 20: (114/24, lift 5.4)
        VMail.Plan <= 0
        Day.Mins > 223.2
        Eve.Charge > 20.5
        ->  class 1  [0.784]

Rule 21: (152/58, lift 4.3)
        Day.Mins > 223.2
        Eve.Charge > 20.5
        ->  class 1  [0.617]

Default class: 0


Evaluation on training data (3333 cases):

             Rules
           ----------------
             No      Errors

             21  116( 3.5%)   <<


            (a)   (b)    <-classified as
           ----  ----
           2827    23    (a): class 0
             93   390    (b): class 1


        Attribute usage:

         96.25% Day.Mins
         84.70% Int.l.Plan
```

```
         82.81% CustServ.Calls
         60.97% Intl.Calls
         60.88% Intl.Mins
         59.83% Eve.Mins
         44.64% Eve.Charge
         12.51% VMail.Plan
          9.30% Night.Mins
          0.84% Eve.Calls
          0.69% Account.Length


Time: 0.1 secs

>
>
> ###############################################################################
#############
>
> ### Prediction
>
> modlogitPred <- predict(modlogit, test, type = "response")
> svmModelPred <- predict(svmModel, test, type = "response")
> randomForestModelPred <- predict(randomForestModel, test, type = "response"
)
>
> # this will create results as new column in a dataset
> test$YHatLogit <- predict(modlogit, test, type = "response")
> test$YHatSVM <- predict(svmModel, test, type = "response")
> test$YHatRF <- predict(randomForestModel, test, type = "response")
>
>
> ## These are theshold parameter setting controls> ## Set up the directory
> ## Load the data
> ## Install the required packages
> ## Know the Data
> ####################
> ## Data Munging ###
> ####################
> ##############################
> ## Drop unwanted variable ###
> ##############################

> ################################
> ## Exploratory Data Analysis ##
> ################################
```

```
> ## Histogram of day minutes
#####################################
> ## split dataset into train and test ##
> #####################################
> ## Model 1 ##
> ## Logistic Regression ##
>
> ## select the variables to use based on forward selection procedure
> ## Lower AIC indicates better model
>
> # forward Elimination
> #influence Plot (clearly shows outliers)
> ##confidence interval
> # put the coefficients and confidence interval in a format onto a useful sc
ale
> ## odds ratio only
> ## odds ratio and 95% CI
> ## Model 2 ##
> ## Support Vector Machine ##
>
> svmModel <- svm(Churn.~., data=train, gamma=0.1, cost=1)
> print(svmModel)
> ##############################################################################
################################
>
> ## Model 3 ##
> ## Random Forest ##
> ## Model 4 ##
> ## Knowledge Discovery: Build a decision tree using C5.0 for churn ##
>
> # the decision variable class must be converted into a factor
> # the variable in order for C50 to process correctly
> ## run the C50 algorithm and show the decision rules
> ### Prediction
tting controls


> predict1 <- function(t) ifelse(modlogitPred > t, 1, 0)
> predict2 <- function(t) ifelse(svmModelPred > t, 1, 0)
> predict3 <- function(t) ifelse(randomForestModelPred > t, 1, 0)
>
> confusionMatrix(predict1(0.5), test$Churn.) ## Logistic Regression
> confusionMatrix(predict2(0.5), test$Churn.) ## SVM Model
> confusionMatrix(predict3(0.5), test$Churn.) ## RandomForest
> table(predict1(0.5), test$Churn.)
```

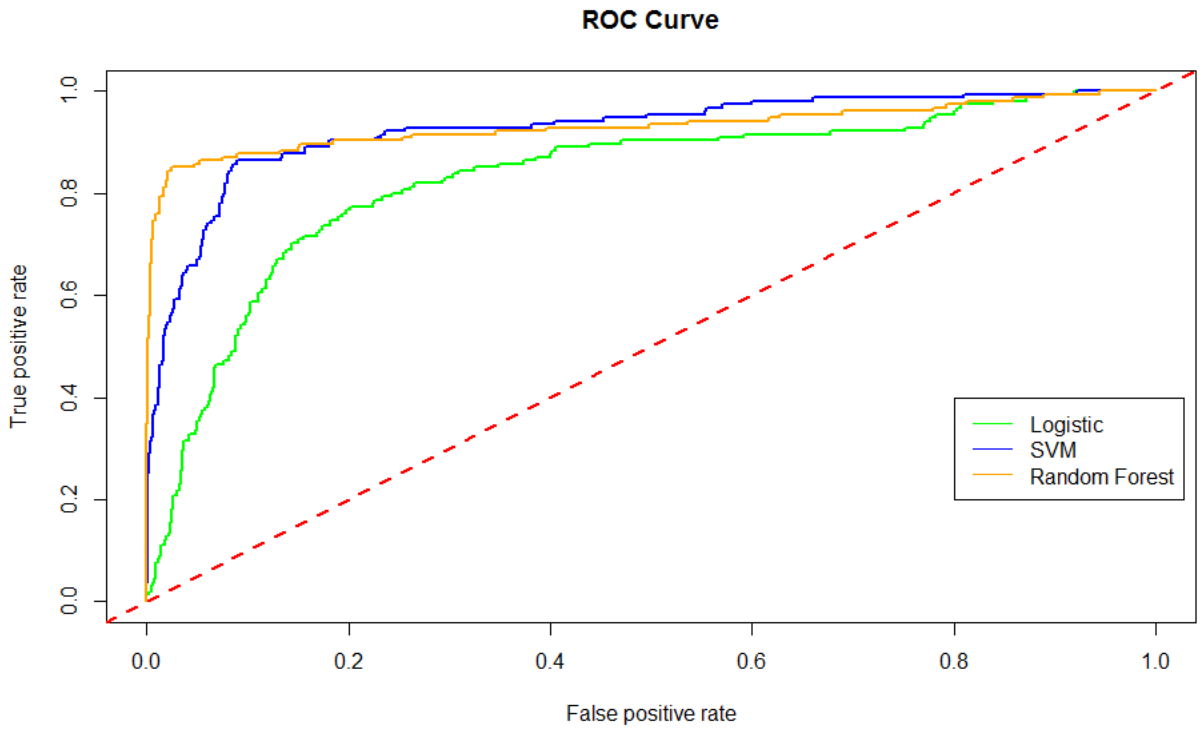
```
    0   1
  0 792 123
  1  24  32
> table(predict2(0.5), test$Churn.)

    0   1
  0 813 107
  1   3  48
> table(predict3(0.5), test$Churn.)

    0   1
  0 809  38
  1   7 117

> ##Accuracy
> mean(predict1(0.5) == test$Churn.)## Accuracy of logit model 85%
[1] 0.8486097
> mean(predict2(0.5) == test$Churn.)## Accuracy of SVM model 89%
[1] 0.8867147
> mean(predict3(0.5) == test$Churn.)## Accuracy of RF model 95%
[1] 0.953656

>
> ############################
> ## ROC For Unpruned Model
> ############################
> LogitPrediction <- prediction(test$YHatLogit, test$Churn.)
> SVMPrediction <- prediction(test$YHatSVM, test$Churn.)
> RFPrediction <- prediction(test$YHatRF, test$Churn.)
>
> perfLogit <- performance(LogitPrediction, "tpr", "fpr")
> perfSVM <- performance(SVMPrediction, "tpr", "fpr")
> perfRF <- performance(RFPrediction, "tpr", "fpr")
>
> plot.new()
> plot(perfLogit, col="green", lwd=2.5)
> plot(perfSVM, add = TRUE, col ="blue", lwd=2.5)
> plot(perfRF, add = TRUE, col = "orange", lwd=2.5)
> abline(0,1,col="Red", lwd=2.5, lty=2)
> title("ROC Curve")
> legend(0.8,0.4,c("Logistic", "SVM", "Random Forest"), lty=c(1,1,1),
+        lwd = c(1.4,1.4,1.4), col=c("green", "blue", "orange"))
> ## We can see random forest is the appropriate model for this
>
```




```
> ### AUC(area under curve) calculation metrics
>
> logit.auc <- performance(LogitPrediction, "auc")
> svm.auc <- performance(SVMPrediction, "auc")
> rf.auc <- performance(RFPrediction, "auc")
>
> logit.auc #AUC=82.96%
An object of class "performance"
Slot "x.name":
[1] "None"

Slot "y.name":
[1] "Area under the ROC curve"

Slot "alpha.name":
[1] "none"

Slot "x.values":
list()

Slot "y.values":
[[1]]
[1] 0.829562
```

```
Slot "alpha.values":
list()

> svm.auc # AUC=92.58%
An object of class "performance"
Slot "x.name":
[1] "None"

Slot "y.name":
[1] "Area under the ROC curve"

Slot "alpha.name":
[1] "none"

Slot "x.values":
list()

Slot "y.values":
[[1]]
[1] 0.9257827


Slot "alpha.values":
list()

> rf.auc # AUC=93.1%
An object of class "performance"
Slot "x.name":
[1] "None"

Slot "y.name":
[1] "Area under the ROC curve"

Slot "alpha.name":
[1] "none"

Slot "x.values":
list()

Slot "y.values":
[[1]]
[1] 0.9315307
```

```
Slot "alpha.values":
list()
```