exploratory_and_tree.R

S. Varatharajan

Tue Oct 09 16:12:05 2018

```r
library(plyr)
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```r
library(ggplot2)
library(gridExtra)
library(ggthemes)
library(caret)
```

```
## Loading required package: lattice
```

```r
library(MASS)
library(randomForest)
```

```
## randomForest 4.6-14
```

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

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:gridExtra':
##
##     combine
```

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

```r
library(party)
```

```
## Loading required package: grid
```

```
## Loading required package: mvtnorm
```

```
## Loading required package: modeltools
```

```
## Loading required package: stats4
```

```
##
## Attaching package: 'modeltools'
```

```
## The following object is masked from 'package:plyr':
##
##     empty

## Loading required package: strucchange

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich
```

```r
library(readxl)
setwd("C:/Users/tsraj/Desktop/Acadgild project")
library(readr)
Churn <- read.csv("C:/Users/tsraj/Desktop/Acadgild project/Churn.csv")
data<-(Churn)
View(data)
library(caret)
library(randomForest)
library(rpart)
library(rpart.plot)
library(ggplot2)
library(lattice)
library(rattle)
```

```
## Rattle: A free graphical interface for data science with R.
## Version 5.2.0 Copyright (c) 2006-2018 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.

##
## Attaching package: 'rattle'

## The following object is masked from 'package:randomForest':
##
##     importance
```

```r
summary(data)
```

```
##  Account.Length  VMail.Message       Day.Mins         Eve.Mins
##  Min.   :  1.0   Min.   : 0.000   Min.   :  0.0    Min.   :  0.0
##  1st Qu.: 74.0   1st Qu.: 0.000   1st Qu.:143.7    1st Qu.:166.6
##  Median :101.0   Median : 0.000   Median :179.4    Median :201.4
##  Mean   :101.1   Mean   : 8.099   Mean   :179.8    Mean   :201.0
##  3rd Qu.:127.0   3rd Qu.:20.000   3rd Qu.:216.4    3rd Qu.:235.3
##  Max.   :243.0   Max.   :51.000   Max.   :350.8    Max.   :363.7
##
```

```
##     Night.Mins        Intl.Mins       CustServ.Calls      Churn
##   Min.   : 23.2    Min.   : 0.00    Min.   :0.000    Min.   :0.0000
##   1st Qu.:167.0    1st Qu.: 8.50    1st Qu.:1.000    1st Qu.:0.0000
##   Median :201.2    Median :10.30    Median :1.000    Median :0.0000
##   Mean   :200.9    Mean   :10.24    Mean   :1.563    Mean   :0.1449
##   3rd Qu.:235.3    3rd Qu.:12.10    3rd Qu.:2.000    3rd Qu.:0.0000
##   Max.   :395.0    Max.   :20.00    Max.   :9.000    Max.   :1.0000
##
##     Int.l.Plan        VMail.Plan       Day.Calls        Day.Charge
##   Min.   :0.00000   Min.   :0.0000   Min.   :  0.0   Min.   : 0.00
##   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.: 87.0   1st Qu.:24.43
##   Median :0.00000   Median :0.0000   Median :101.0   Median :30.50
##   Mean   :0.09691   Mean   :0.2766   Mean   :100.4   Mean   :30.56
##   3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:114.0   3rd Qu.:36.79
##   Max.   :1.00000   Max.   :1.0000   Max.   :165.0   Max.   :59.64
##
##     Eve.Calls         Eve.Charge       Night.Calls      Night.Charge
##   Min.   :  0.0    Min.   : 0.00    Min.   : 33.0    Min.   : 1.040
##   1st Qu.: 87.0    1st Qu.:14.16    1st Qu.: 87.0    1st Qu.: 7.520
##   Median :100.0    Median :17.12    Median :100.0    Median : 9.050
##   Mean   :100.1    Mean   :17.08    Mean   :100.1    Mean   : 9.039
##   3rd Qu.:114.0    3rd Qu.:20.00    3rd Qu.:113.0    3rd Qu.:10.590
##   Max.   :170.0    Max.   :30.91    Max.   :175.0    Max.   :17.770
##
##     Intl.Calls        Intl.Charge        State          Area.Code
##   Min.   : 0.000   Min.   :0.000    WV     : 106    Min.   :408.0
##   1st Qu.: 3.000   1st Qu.:2.300    MN     :  84    1st Qu.:408.0
##   Median : 4.000   Median :2.780    NY     :  83    Median :415.0
##   Mean   : 4.479   Mean   :2.765    AL     :  80    Mean   :437.2
##   3rd Qu.: 6.000   3rd Qu.:3.270    OH     :  78    3rd Qu.:510.0
##   Max.   :20.000   Max.   :5.400    OR     :  78    Max.   :510.0
##                                     (Other):2824
##       Phone
##   327-1058:   1
##   327-1319:   1
##   327-3053:   1
##   327-3587:   1
##   327-3850:   1
##   327-3954:   1
##   (Other) :3327
```

```r
library(C50)
#install.package('devtools') # Only needed if you dont have this installed.
library(devtools)
install_github('adam-m-mcelhinney/helpRFunctions')
```

```
## Installation failed: Could not resolve host: raw.githubusercontent.com
```

```r
library(helpRFunctions)
names(data)
```

```
##  [1] "Account.Length" "VMail.Message"  "Day.Mins"       "Eve.Mins"
##  [5] "Night.Mins"     "Intl.Mins"      "CustServ.Calls" "Churn"
##  [9] "Int.l.Plan"     "VMail.Plan"     "Day.Calls"      "Day.Charge"
## [13] "Eve.Calls"      "Eve.Charge"     "Night.Calls"    "Night.Charge"
## [17] "Intl.Calls"     "Intl.Charge"    "State"          "Area.Code"
## [21] "Phone"
```

```r
dim(data)
```

```
## [1] 3333   21
```

```r
library(caret)
library(zoo)

#data[is.na(data)] <- mean(data, na.rm = TRUE)
str(data)
```

```
## 'data.frame':    3333 obs. of  21 variables:
##  $ Account.Length: int  128 107 137 84 75 118 121 147 117 141 ...
##  $ VMail.Message : int  25 26 0 0 0 0 24 0 0 37 ...
##  $ Day.Mins      : num  265 162 243 299 167 ...
##  $ Eve.Mins      : num  197.4 195.5 121.2 61.9 148.3 ...
##  $ Night.Mins    : num  245 254 163 197 187 ...
##  $ Intl.Mins     : num  10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 8.7 11.2 ...
##  $ CustServ.Calls: int  1 1 0 2 3 0 3 0 1 0 ...
##  $ Churn         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Int.l.Plan    : int  0 0 0 1 1 1 0 1 0 1 ...
##  $ VMail.Plan    : int  1 1 0 0 0 0 1 0 0 1 ...
##  $ 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.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.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.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 ...
##  $ State         : Factor w/ 51 levels "AK","AL","AR",..: 17 36 32 36 37 2
## 20 25 19 50 ...
##  $ 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 ...
```

```r
summary(data)
```

```
##  Account.Length  VMail.Message       Day.Mins        Eve.Mins
##  Min.   :  1.0   Min.   : 0.000   Min.   :  0.0   Min.   :  0.0
##  1st Qu.: 74.0   1st Qu.: 0.000   1st Qu.:143.7   1st Qu.:166.6
##  Median :101.0   Median : 0.000   Median :179.4   Median :201.4
##  Mean   :101.1   Mean   : 8.099   Mean   :179.8   Mean   :201.0
##  3rd Qu.:127.0   3rd Qu.:20.000   3rd Qu.:216.4   3rd Qu.:235.3
##  Max.   :243.0   Max.   :51.000   Max.   :350.8   Max.   :363.7
```

```
##
##     Night.Mins        Intl.Mins      CustServ.Calls       Churn
##   Min.   : 23.2    Min.   : 0.00    Min.   :0.000    Min.   :0.0000
##   1st Qu.:167.0    1st Qu.: 8.50    1st Qu.:1.000    1st Qu.:0.0000
##   Median :201.2    Median :10.30    Median :1.000    Median :0.0000
##   Mean   :200.9    Mean   :10.24    Mean   :1.563    Mean   :0.1449
##   3rd Qu.:235.3    3rd Qu.:12.10    3rd Qu.:2.000    3rd Qu.:0.0000
##   Max.   :395.0    Max.   :20.00    Max.   :9.000    Max.   :1.0000
##
##     Int.l.Plan        VMail.Plan        Day.Calls        Day.Charge
##   Min.   :0.00000   Min.   :0.0000   Min.   :  0.0    Min.   : 0.00
##   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.: 87.0    1st Qu.:24.43
##   Median :0.00000   Median :0.0000   Median :101.0    Median :30.50
##   Mean   :0.09691   Mean   :0.2766   Mean   :100.4    Mean   :30.56
##   3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:114.0    3rd Qu.:36.79
##   Max.   :1.00000   Max.   :1.0000   Max.   :165.0    Max.   :59.64
##
##     Eve.Calls         Eve.Charge        Night.Calls       Night.Charge
##   Min.   :  0.0    Min.   : 0.00    Min.   : 33.0    Min.   : 1.040
##   1st Qu.: 87.0    1st Qu.:14.16    1st Qu.: 87.0    1st Qu.: 7.520
##   Median :100.0    Median :17.12    Median :100.0    Median : 9.050
##   Mean   :100.1    Mean   :17.08    Mean   :100.1    Mean   : 9.039
##   3rd Qu.:114.0    3rd Qu.:20.00    3rd Qu.:113.0    3rd Qu.:10.590
##   Max.   :170.0    Max.   :30.91    Max.   :175.0    Max.   :17.770
##
##     Intl.Calls        Intl.Charge        State         Area.Code
##   Min.   : 0.000   Min.   :0.000    WV     : 106    Min.   :408.0
##   1st Qu.: 3.000   1st Qu.:2.300    MN     :  84    1st Qu.:408.0
##   Median : 4.000   Median :2.780    NY     :  83    Median :415.0
##   Mean   : 4.479   Mean   :2.765    AL     :  80    Mean   :437.2
##   3rd Qu.: 6.000   3rd Qu.:3.270    OH     :  78    3rd Qu.:510.0
##   Max.   :20.000   Max.   :5.400    OR     :  78    Max.   :510.0
##                                     (Other):2824
##        Phone
##   327-1058:    1
##   327-1319:    1
##   327-3053:    1
##   327-3587:    1
##   327-3850:    1
##   327-3954:    1
##   (Other) :3327

pairs(data[1:8])
```
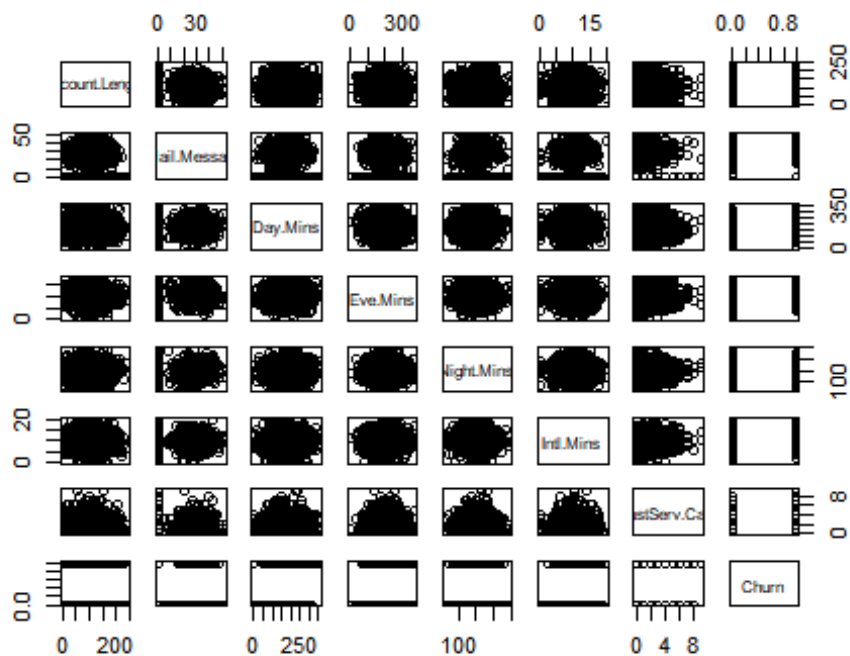
```r
# enable multi-core processing
library(doParallel)
```

```
## Loading required package: foreach
```

```
## Loading required package: iterators
```

```
## Loading required package: parallel
```

```r
#cl <- makeCluster(detectCores())
dim(data)
```

```
## [1] 3333    21
```

```r
registerDoParallel()
set.seed(12345)
dataTrain<-data[1:2500,]
dataTest<-data[2501:3033,]
head(dataTrain)
```

```
##   Account.Length VMail.Message Day.Mins Eve.Mins Night.Mins Intl.Mins
## 1            128            25    265.1    197.4      244.7      10.0
## 2            107            26    161.6    195.5      254.4      13.7
## 3            137             0    243.4    121.2      162.6      12.2
## 4             84             0    299.4     61.9      196.9       6.6
## 5             75             0    166.7    148.3      186.9      10.1
## 6            118             0    223.4    220.6      203.9       6.3
##   CustServ.Calls Churn Int.l.Plan VMail.Plan Day.Calls Day.Charge
```

```
## 1               1       0        0          1      110      45.07
## 2               1       0        0          1      123      27.47
## 3               0       0        0          0      114      41.38
## 4               2       0        1          0       71      50.90
## 5               3       0        1          0      113      28.34
## 6               0       0        1          0       98      37.98
##    Eve.Calls Eve.Charge Night.Calls Night.Charge Intl.Calls Intl.Charge
## 1         99      16.78          91        11.01          3        2.70
## 2        103      16.62         103        11.45          3        3.70
## 3        110      10.30         104         7.32          5        3.29
## 4         88       5.26          89         8.86          7        1.78
## 5        122      12.61         121         8.41          3        2.73
## 6        101      18.75         118         9.18          6        1.70
##    State Area.Code      Phone
## 1    KS       415 382-4657
## 2    OH       415 371-7191
## 3    NJ       415 358-1921
## 4    OH       408 375-9999
## 5    OK       415 330-6626
## 6    AL       510 391-8027
```

```r
head(dataTest)
```

```
##      Account.Length VMail.Message Day.Mins Eve.Mins Night.Mins Intl.Mins
## 2501            121            41    215.5    241.8      147.0       9.6
## 2502            101             0    124.8    257.2      193.2      13.4
## 2503            115             0    178.7    271.0      245.9      16.4
## 2504            168             0    183.2    179.2      292.8       9.9
## 2505             90             0    167.5    139.1      138.4      13.0
## 2506             70             0    147.1    200.0      234.9      12.5
##      CustServ.Calls Churn Int.l.Plan VMail.Plan Day.Calls Day.Charge
## 2501              1     0          0          1        95      36.64
## 2502              0     0          0          0        66      21.22
## 2503              2     0          0          0       114      30.38
## 2504              2     0          0          0       131      31.14
## 2505              1     0          0          0        96      28.48
## 2506              3     0          0          0       105      25.01
##      Eve.Calls Eve.Charge Night.Calls Night.Charge Intl.Calls Intl.Charge
## 2501        92      20.55         108         6.61          3        2.59
## 2502        85      21.86         115         8.69          4        3.62
## 2503        96      23.04          94        11.07          5        4.43
## 2504        73      15.23         100        13.18          5        2.67
## 2505       104      11.82          87         6.23          1        3.51
## 2506       135      17.00          65        10.57          9        3.38
##      State Area.Code      Phone
## 2501    UT       415 337-2348
## 2502    IL       415 342-8702
## 2503    DC       408 393-5802
## 2504    IN       415 384-2219
```

```
## 2505    NM        415 347-6164
## 2506    MS        510 376-9940

indexNA <- as.vector(sapply(dataTrain[,1:21],function(x)
{length(which(is.na(x)))!=0}))
dataTrain <- dataTrain[,!indexNA]
dataTrain<-na.exclude(dataTrain)
library(tree)

str(Churn)

## 'data.frame':    3333 obs. of  21 variables:
##  $ Account.Length: int  128 107 137 84 75 118 121 147 117 141 ...
##  $ VMail.Message : int  25 26 0 0 0 0 24 0 0 37 ...
##  $ Day.Mins      : num  265 162 243 299 167 ...
##  $ Eve.Mins      : num  197.4 195.5 121.2 61.9 148.3 ...
##  $ Night.Mins    : num  245 254 163 197 187 ...
##  $ Intl.Mins     : num  10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 8.7 11.2 ...
##  $ CustServ.Calls: int  1 1 0 2 3 0 3 0 1 0 ...
##  $ Churn         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Int.l.Plan    : int  0 0 0 1 1 1 0 1 0 1 ...
##  $ VMail.Plan    : int  1 1 0 0 0 0 1 0 0 1 ...
##  $ 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.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.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.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 ...
##  $ State         : Factor w/ 51 levels "AK","AL","AR",..: 17 36 32 36 37 2
20 25 19 50 ...
##  $ 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 ...

sapply(Churn ,function(x) sum(is.na(x)))

## Account.Length  VMail.Message        Day.Mins        Eve.Mins      Night.Mins
##              0              0               0               0               0
##      Intl.Mins CustServ.Calls           Churn      Int.l.Plan      VMail.Plan
##              0              0               0               0               0
##      Day.Calls     Day.Charge       Eve.Calls      Eve.Charge     Night.Calls
##              0              0               0               0               0
##   Night.Charge     Intl.Calls     Intl.Charge           State       Area.Code
##              0              0               0               0               0
##          Phone
##              0

Churn<- Churn[complete.cases(Churn), ]
numeric.var <- sapply(Churn, is.numeric)
```

```r
corr.matrix <- cor(Churn[,numeric.var])
corrplot(corr.matrix, main="\n\nCorrelation Plot for Numerical Variables",
method="number")
```


Correlation Plot for Numerical Variables

```r
dim(Churn)
```

```
## [1] 3333    21
```
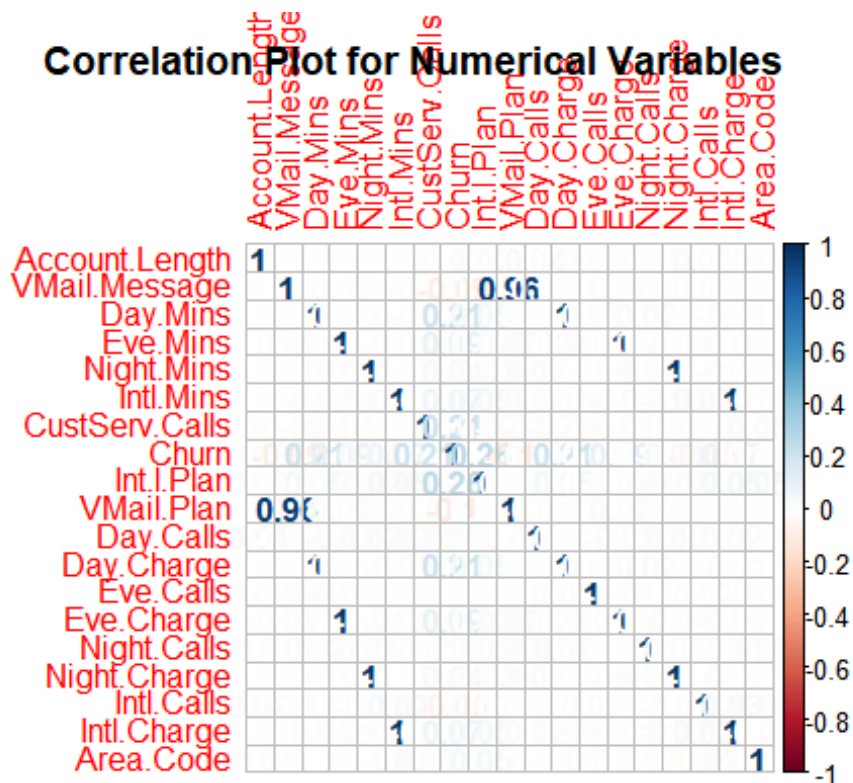
```r
summary(Churn)
```

```
##   Account.Length  VMail.Message       Day.Mins         Eve.Mins
##   Min.   :  1.0   Min.   : 0.000   Min.   :  0.0   Min.   :  0.0
##   1st Qu.: 74.0   1st Qu.: 0.000   1st Qu.:143.7   1st Qu.:166.6
##   Median :101.0   Median : 0.000   Median :179.4   Median :201.4
##   Mean   :101.1   Mean   : 8.099   Mean   :179.8   Mean   :201.0
##   3rd Qu.:127.0   3rd Qu.:20.000   3rd Qu.:216.4   3rd Qu.:235.3
##   Max.   :243.0   Max.   :51.000   Max.   :350.8   Max.   :363.7
##
##    Night.Mins       Intl.Mins     CustServ.Calls      Churn
##   Min.   : 23.2   Min.   : 0.00   Min.   :0.000   Min.   :0.0000
##   1st Qu.:167.0   1st Qu.: 8.50   1st Qu.:1.000   1st Qu.:0.0000
##   Median :201.2   Median :10.30   Median :1.000   Median :0.0000
##   Mean   :200.9   Mean   :10.24   Mean   :1.563   Mean   :0.1449
##   3rd Qu.:235.3   3rd Qu.:12.10   3rd Qu.:2.000   3rd Qu.:0.0000
##   Max.   :395.0   Max.   :20.00   Max.   :9.000   Max.   :1.0000
##
##    Int.l.Plan        VMail.Plan        Day.Calls        Day.Charge
```

```
##  Min.   :0.00000    Min.   :0.0000    Min.   :  0.0    Min.    : 0.00
##  1st Qu.:0.00000    1st Qu.:0.0000    1st Qu.: 87.0    1st Qu.:24.43
##  Median :0.00000    Median :0.0000    Median :101.0    Median :30.50
##  Mean   :0.09691    Mean   :0.2766    Mean   :100.4    Mean    :30.56
##  3rd Qu.:0.00000    3rd Qu.:1.0000    3rd Qu.:114.0    3rd Qu.:36.79
##  Max.   :1.00000    Max.   :1.0000    Max.   :165.0    Max.    :59.64
##
##    Eve.Calls        Eve.Charge       Night.Calls       Night.Charge
##  Min.   :  0.0    Min.   : 0.00    Min.   : 33.0    Min.    : 1.040
##  1st Qu.: 87.0    1st Qu.:14.16    1st Qu.: 87.0    1st Qu.: 7.520
##  Median :100.0    Median :17.12    Median :100.0    Median : 9.050
##  Mean   :100.1    Mean   :17.08    Mean   :100.1    Mean    : 9.039
##  3rd Qu.:114.0    3rd Qu.:20.00    3rd Qu.:113.0    3rd Qu.:10.590
##  Max.   :170.0    Max.   :30.91    Max.   :175.0    Max.    :17.770
##
##    Intl.Calls       Intl.Charge        State        Area.Code
##  Min.   : 0.000   Min.   :0.000    WV     : 106   Min.   :408.0
##  1st Qu.: 3.000   1st Qu.:2.300    MN     :  84   1st Qu.:408.0
##  Median : 4.000   Median :2.780    NY     :  83   Median :415.0
##  Mean   : 4.479   Mean   :2.765    AL     :  80   Mean   :437.2
##  3rd Qu.: 6.000   3rd Qu.:3.270    OH     :  78   3rd Qu.:510.0
##  Max.   :20.000   Max.   :5.400    OR     :  78   Max.   :510.0
##                                    (Other):2824
##        Phone
##  327-1058:   1
##  327-1319:   1
##  327-3053:   1
##  327-3587:   1
##  327-3850:   1
##  327-3954:   1
##  (Other) :3327
```

head(Churn)

```
##   Account.Length VMail.Message Day.Mins Eve.Mins Night.Mins Intl.Mins
## 1            128            25    265.1    197.4      244.7      10.0
## 2            107            26    161.6    195.5      254.4      13.7
## 3            137             0    243.4    121.2      162.6      12.2
## 4             84             0    299.4     61.9      196.9       6.6
## 5             75             0    166.7    148.3      186.9      10.1
## 6            118             0    223.4    220.6      203.9       6.3
##   CustServ.Calls Churn Int.l.Plan VMail.Plan Day.Calls Day.Charge
## 1              1     0          0          1       110      45.07
## 2              1     0          0          1       123      27.47
## 3              0     0          0          0       114      41.38
## 4              2     0          1          0        71      50.90
## 5              3     0          1          0       113      28.34
## 6              0     0          1          0        98      37.98
##   Eve.Calls Eve.Charge Night.Calls Night.Charge Intl.Calls Intl.Charge
## 1        99      16.78          91        11.01          3        2.70
```

```
## 2          103       16.62          103         11.45           3         3.70
## 3          110       10.30          104          7.32           5         3.29
## 4           88        5.26           89          8.86           7         1.78
## 5          122       12.61          121          8.41           3         2.73
## 6          101       18.75          118          9.18           6         1.70
##     State Area.Code      Phone
## 1     KS        415 382-4657
## 2     OH        415 371-7191
## 3     NJ        415 358-1921
## 4     OH        408 375-9999
## 5     OK        415 330-6626
## 6     AL        510 391-8027
```

```r
Churn$Account.Length<-as.factor(Churn$Account.Length)
Churn$VMail.Message<-as.factor(Churn$VMail.Message)
Churn$CustServ.Calls<-as.factor(Churn$CustServ.Calls)
library(GGally)
library(ISLR)
require(GGally)
data(tips, package="reshape")

pm = ggpairs(data=Churn,
             columns=7:9,
             upper = list(continuous = "density"),
             lower = list(combo = "facetdensity"),
             title="Churn data",
             colour = "Red")
```

```
## Warning in warn_if_args_exist(list(...)): Extra arguments: "colour" are
## being ignored. If these are meant to be aesthetics, submit them using the
## 'mapping' variable within ggpairs with ggplot2::aes or
## ggplot2::aes_string.
```

```r
print(pm)
```

```
## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive
```

Churn data


```
pm = ggpairs(data=Churn,
             columns=4:9,
             upper = list(continuous = "density"),
             lower = list(combo = "facetdensity"),
             title="Churn data",
             colour = "Green")

## Warning in warn_if_args_exist(list(...)): Extra arguments: "colour" are
## being ignored. If these are meant to be aesthetics, submit them using the
## 'mapping' variable within ggpairs with ggplot2::aes or
## ggplot2::aes_string.

print(pm)

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
```
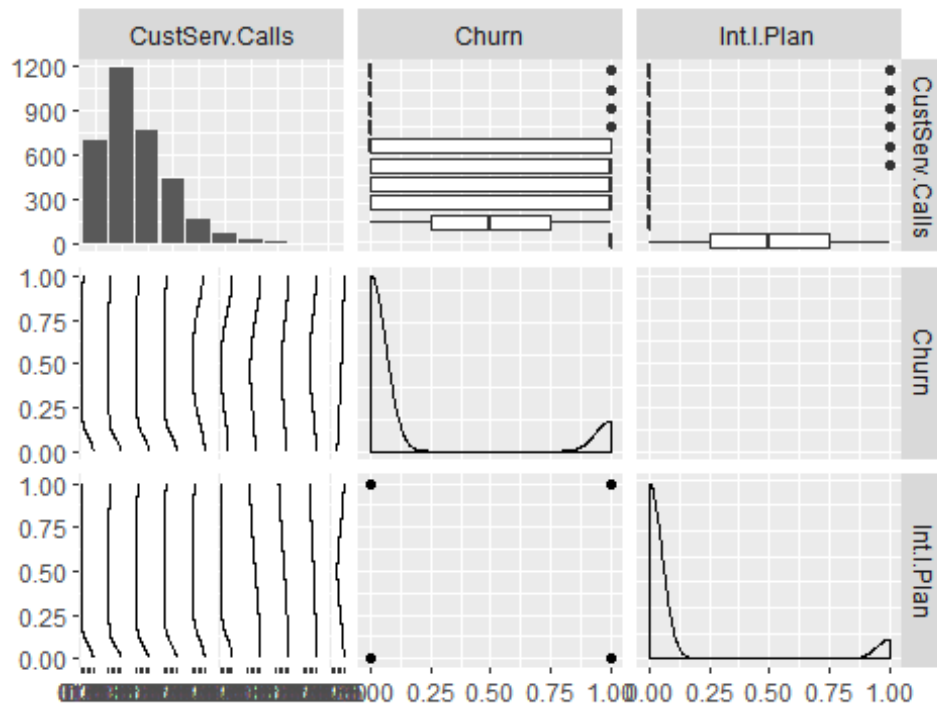
```
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive
```


Churn data

```
pm = ggpairs(data=Churn,
             columns=8:15,
             upper = list(continuous = "density"),
             lower = list(combo = "facetdensity"),
             title="Churn data",
             colour = "Red")

## Warning in warn_if_args_exist(list(...)): Extra arguments: "colour" are
## being ignored. If these are meant to be aesthetics, submit them using the
## 'mapping' variable within ggpairs with ggplot2::aes or
## ggplot2::aes_string.

print(pm)

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive
```

```
## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive

## Warning: Computation failed in `stat_density2d()`:
## bandwidths must be strictly positive
```
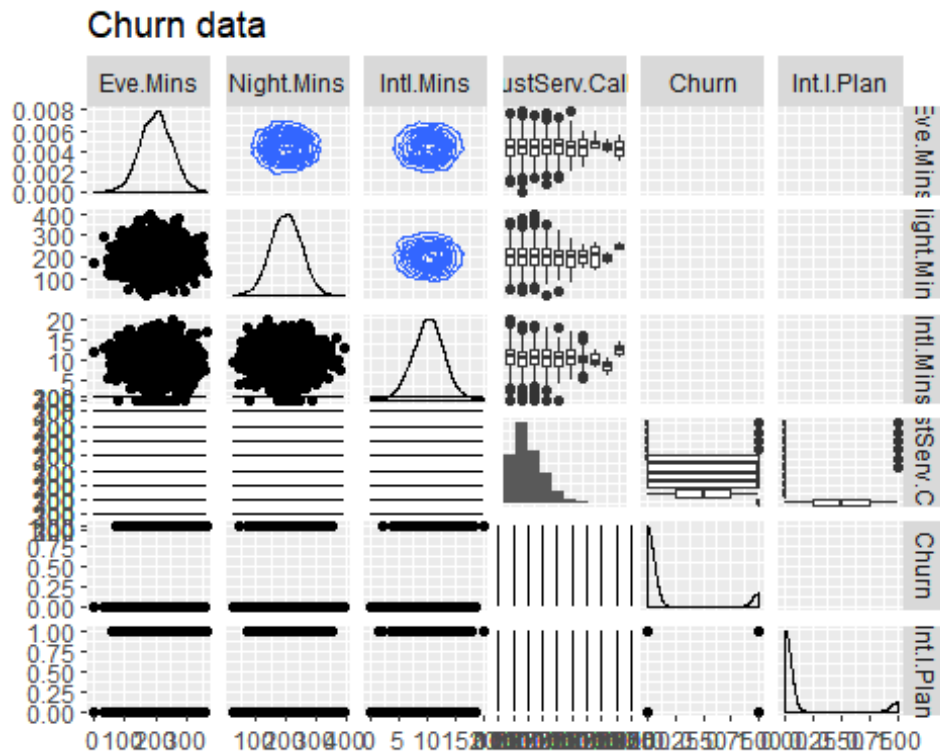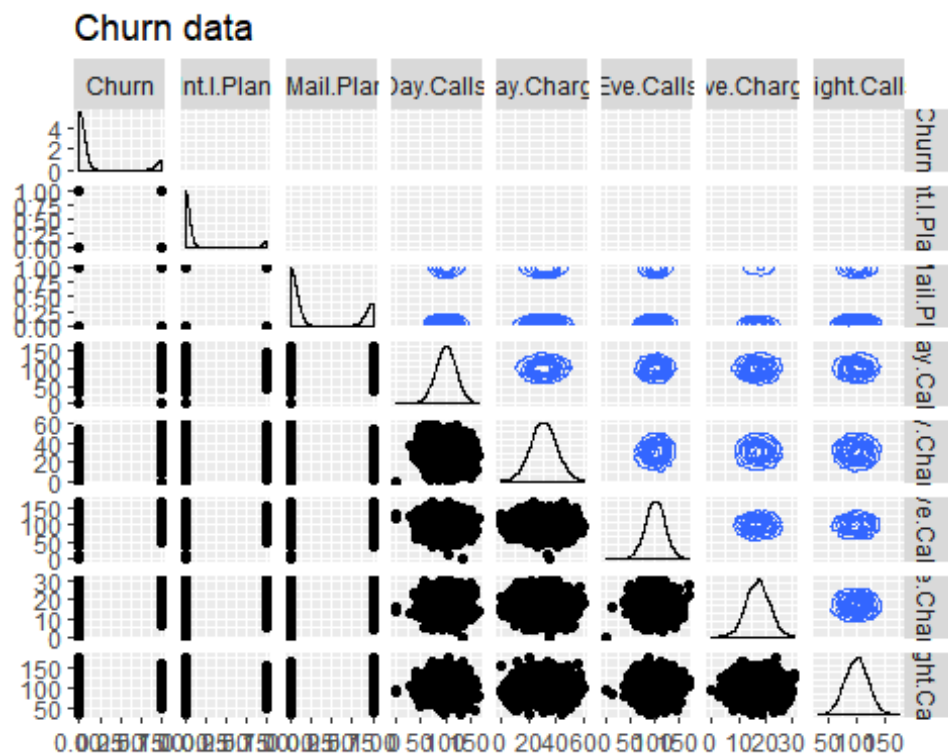
Churn data

```
pm = ggpairs(data=Churn,
             columns=16:18,
             upper = list(continuous = "density"),
             lower = list(combo = "facetdensity"),
             title="Churn data",
             colour = "Red")

## Warning in warn_if_args_exist(list(...)): Extra arguments: "colour" are
## being ignored. If these are meant to be aesthetics, submit them using the
## 'mapping' variable within ggpairs with ggplot2::aes or
## ggplot2::aes_string.

print(pm)
```

Churn data
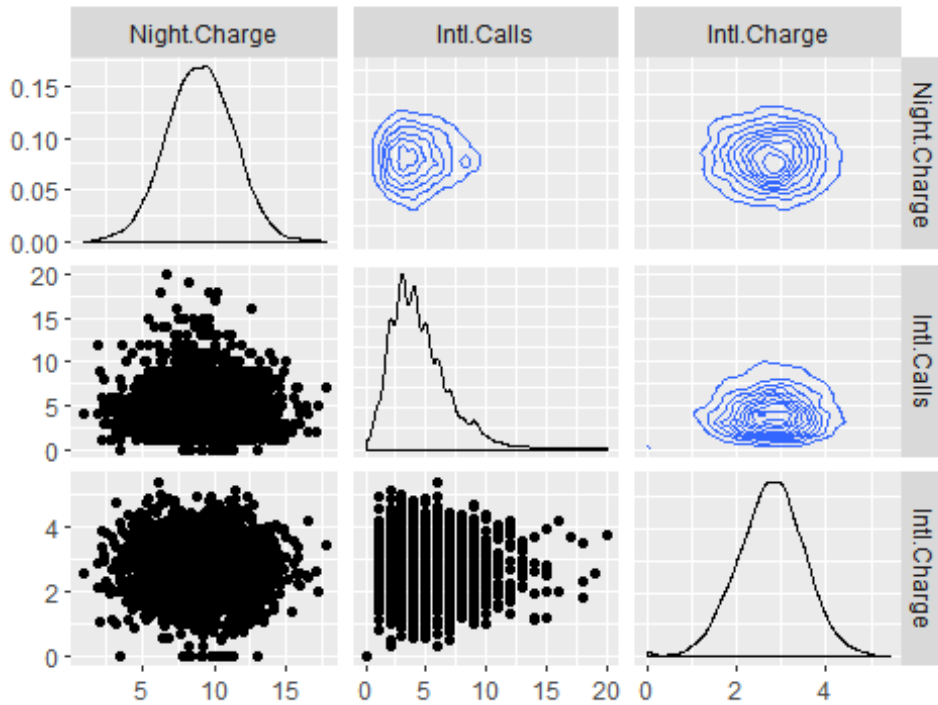

```
GGally::ggpairs(Churn[,8:16],ggplot2::aes(color='Red'),diag =
list(continuous=wrap("barDiag",
color="blue")),upper=list(continuous=wrap("cor", size=2)))

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```r
GGally::ggpairs(Churn[,3:8],ggplot2::aes(color='Red'),diag =
list(continuous=wrap("barDiag",
color="blue")),upper=list(continuous=wrap("cor", size=2)))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```r
GGally::ggpairs(Churn[,3:16],ggplot2::aes(color='Red'),diag =
list(continuous=wrap("barDiag",
color="blue")),upper=list(continuous=wrap("cor", size=2)))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```
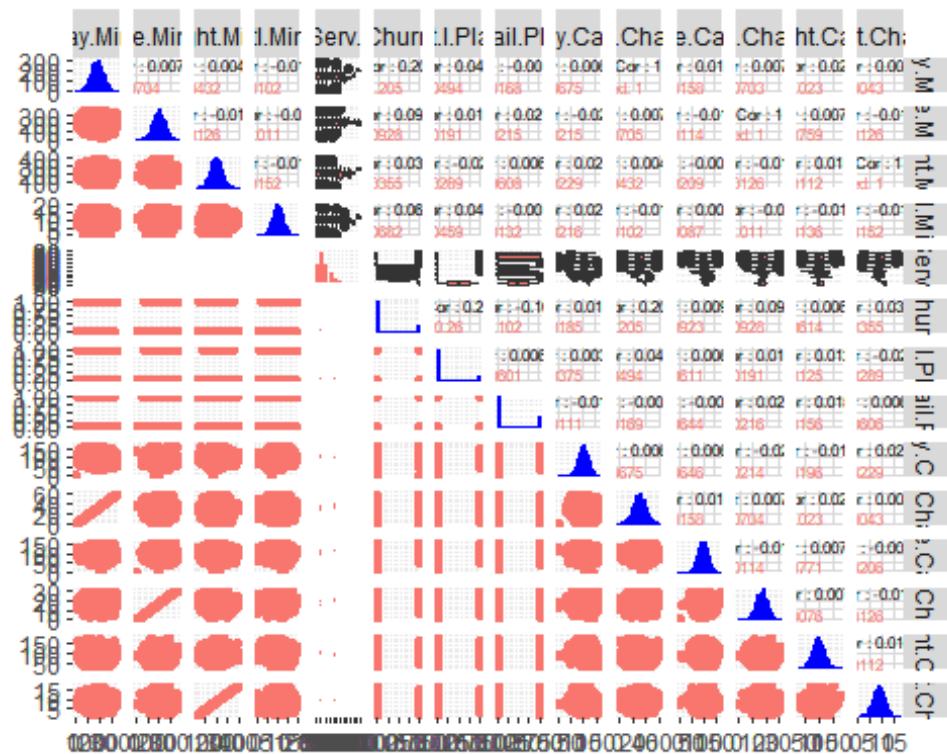
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```
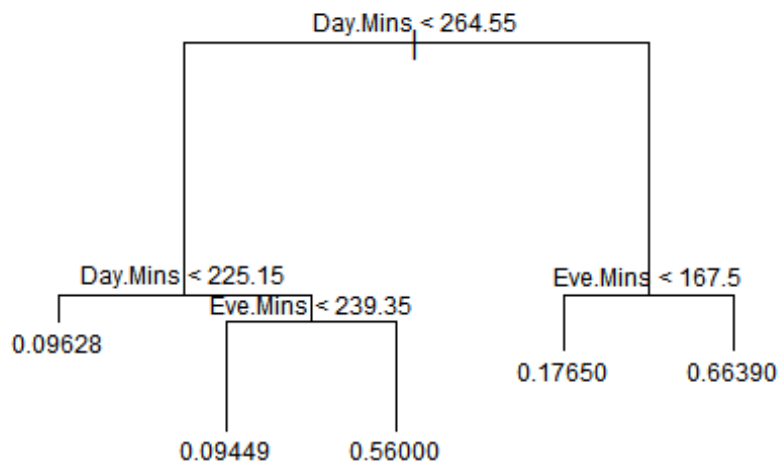


```
table(Churn$Churn)

##
##    0    1
## 2850  483

library(doParallel)
registerDoParallel()
set.seed(12345)
training<-Churn[1:2500,]
testing<-Churn[2501:3033,]

# creating decision tree model
library(tree)
modelRegtree<-
tree(Churn~Day.Mins+Eve.Mins+Intl.Mins+Day.Charge+Eve.Calls+Eve.Charge+Intl.C
alls+Intl.Charge,data=training)
plot(modelRegtree)
text(modelRegtree,pretty = 0, cex=0.75)
```

```
pred<-predict(modelRegtree, newdata = testing)
head(pred,nrow = 5)

##      2501       2502       2503       2504       2505       2506
## 0.09627792 0.09627792 0.09627792 0.09627792 0.09627792 0.09627792

head(pred,n = 5)

##      2501       2502       2503       2504       2505
## 0.09627792 0.09627792 0.09627792 0.09627792 0.09627792

ME<-sum(testing$Churn-pred)/nrow(testing)
ME

## [1] 0.02800086

RSS<-sum(testing$Churn-pred)^2
RSS

## [1] 222.7394

RMSE<-sqrt(RSS/nrow(testing))
RMSE

## [1] 0.64645

MAPE<-sum(abs(testing$Churn -pred)/testing$Churn)*100
MAPE
```

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