Random Forest with out month of the year

Biswajeet

4 October 2018

```
library(DataExplorer)
```

```
## Warning: package 'DataExplorer' was built under R version 3.5.1
```

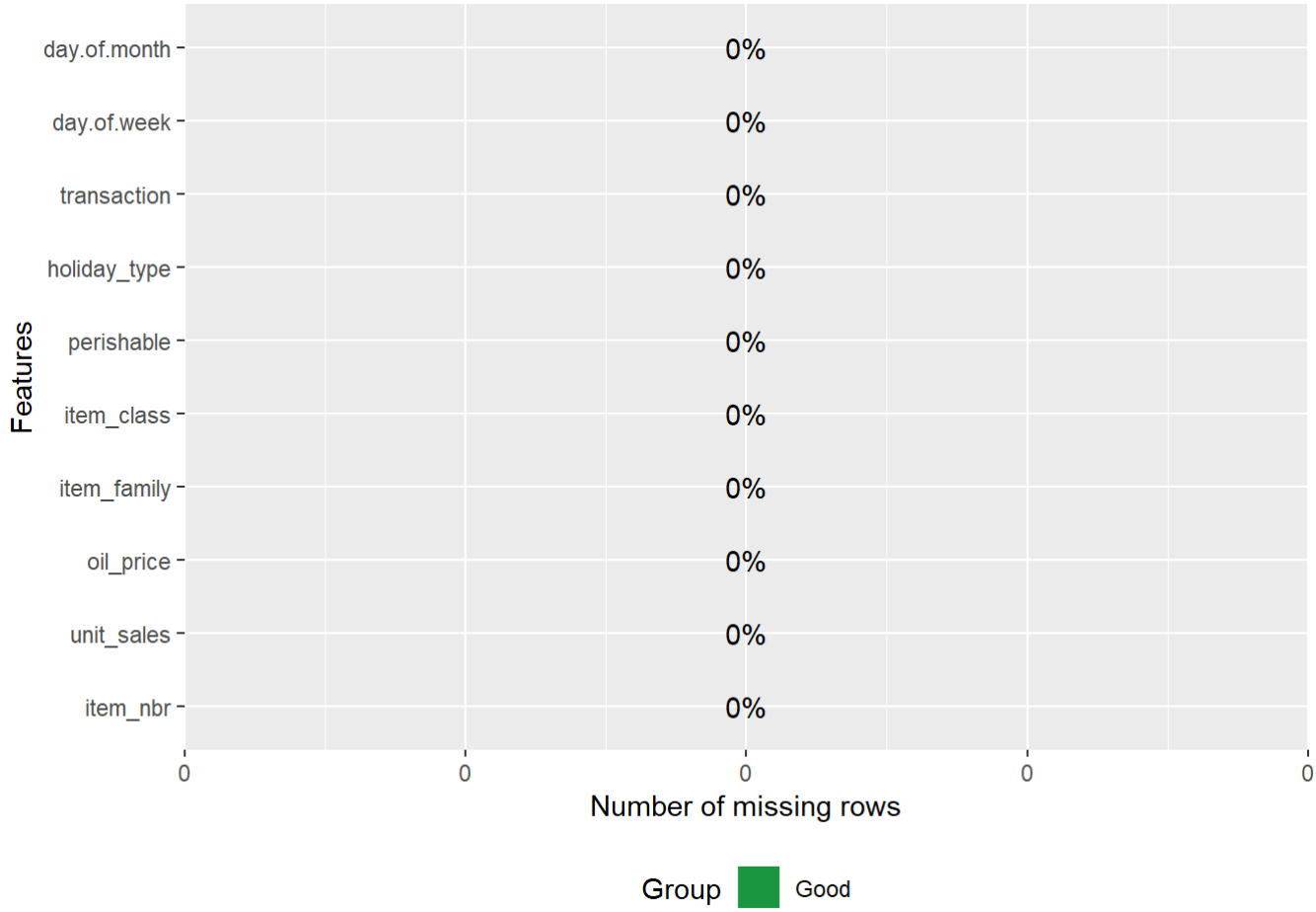
```
data<-read.csv('newtrain2.csv')
```

Here date,id,month of year column are removed

```
data<-data[,-c(1,2,11)]
str(data)
```

```
## 'data.frame':    3582 obs. of  10 variables:
##  $ item_nbr    : int  108831 208384 257847 305229 314384 315176 364606 364738 464374 50233
1 ...
##  $ unit_sales  : num  135 121 108 105 143 ...
##  $ oil_price   : num  93.1 93.1 93.1 93.1 93.1 ...
##  $ item_family : Factor w/ 6 levels "BEVERAGES","BREAD/BAKERY",..: 6 3 1 4 4 1 4 1 1 2 ...
##  $ item_class  : int  2416 2502 1120 1014 1004 1124 1014 1124 1124 2702 ...
##  $ perishable  : int  1 1 0 0 0 0 0 0 0 1 ...
##  $ holiday_type: Factor w/ 5 levels "Additional","Event",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ transaction : int  3487 3487 3487 3487 3487 3487 3487 3487 3487 3487 ...
##  $ day.of.week : Factor w/ 7 levels "Fri","Mon","Sat",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ day.of.month: int  2 2 2 2 2 2 2 2 2 2 ...
```

```
plot_missing(data)
```




```
names(data)
```

```
##  [1] "item_nbr"     "unit_sales"   "oil_price"    "item_family"
##  [5] "item_class"   "perishable"   "holiday_type" "transaction"
##  [9] "day.of.week"  "day.of.month"
```

converting all categorical to factor variables

```
table(data$item_family)
```

```
##
##    BEVERAGES BREAD/BAKERY         EGGS    GROCERY I        MEATS
##         1968          180          360          540          180
##      POULTRY
##          354
```

```
data$item_nbr<-as.factor(data$item_nbr)
data$item_family<-as.factor(data$item_family)
data$holiday_type<-as.factor(data$holiday_type)
table(data$day.of.week)
```

```
##
## Fri Mon Sat Sun Thu Tue Wed
## 518 493 520 519 517 497 518
```

```
table(data$day.of.month)
```

```
##
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
##  99 120 118 120 120 120 120 119 120 120 119 120 119 120 118 120 119 119
##  19  20  21  22  23  24  25  26  27  28  29  30  31
## 119 119 119 120 119 119 119 120 120 119  99 100  60
```

```
data$day.of.week<-as.factor(data$day.of.week)
data$day.of.month<-as.factor(data$day.of.month)
data$item_class<-as.factor(data$item_class)
data$perishable<-as.factor(data$perishable)
str(data)
```

```
## 'data.frame':    3582 obs. of  10 variables:
##  $ item_nbr    : Factor w/ 20 levels "108831","208384",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ unit_sales  : num  135 121 108 105 143 ...
##  $ oil_price   : num  93.1 93.1 93.1 93.1 93.1 ...
##  $ item_family : Factor w/ 6 levels "BEVERAGES","BREAD/BAKERY",..: 6 3 1 4 4 1 4 1 1 2 ...
##  $ item_class  : Factor w/ 10 levels "1004","1014",..: 8 9 3 2 1 5 2 5 5 10 ...
##  $ perishable  : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 1 1 1 2 ...
##  $ holiday_type: Factor w/ 5 levels "Additional","Event",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ transaction : int  3487 3487 3487 3487 3487 3487 3487 3487 3487 3487 ...
##  $ day.of.week : Factor w/ 7 levels "Fri","Mon","Sat",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ day.of.month: Factor w/ 31 levels "1","2","3","4",..: 2 2 2 2 2 2 2 2 2 2 ...
```

creating dummy variables for categorical and factor variables

```
library(caret)
```

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

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.5.1
```

```
dmy <- dummyVars(" ~ .",,data = data)
trsf <- data.frame(predict(dmy, newdata = data))
head(trsf)
```

```
##   item_nbr.108831 item_nbr.208384 item_nbr.257847 item_nbr.305229
## 1               1               0               0               0
## 2               0               1               0               0
## 3               0               0               1               0
## 4               0               0               0               1
## 5               0               0               0               0
## 6               0               0               0               0
##   item_nbr.314384 item_nbr.315176 item_nbr.364606 item_nbr.364738
## 1               0               0               0               0
## 2               0               0               0               0
## 3               0               0               0               0
## 4               0               0               0               0
## 5               1               0               0               0
## 6               0               1               0               0
##   item_nbr.464374 item_nbr.502331 item_nbr.557256 item_nbr.582863
## 1               0               0               0               0
## 2               0               0               0               0
## 3               0               0               0               0
## 4               0               0               0               0
## 5               0               0               0               0
## 6               0               0               0               0
##   item_nbr.807493 item_nbr.819932 item_nbr.903286 item_nbr.1047679
## 1               0               0               0                0
## 2               0               0               0                0
## 3               0               0               0                0
## 4               0               0               0                0
## 5               0               0               0                0
## 6               0               0               0                0
##   item_nbr.1047690 item_nbr.1066900 item_nbr.1066901 item_nbr.1074327
## 1                0                0               0                0
## 2                0                0               0                0
## 3                0                0               0                0
## 4                0                0               0                0
## 5                0                0               0                0
## 6                0                0               0                0
##   unit_sales oil_price item_family.BEVERAGES item_family.BREAD.BAKERY
## 1    135.003     93.14                     0                        0
## 2    121.000     93.14                     0                        0
## 3    108.000     93.14                     1                        0
## 4    105.000     93.14                     0                        0
## 5    143.000     93.14                     0                        0
## 6    130.000     93.14                     1                        0
##   item_family.EGGS item_family.GROCERY.I item_family.MEATS
## 1                0                     0                0
## 2                1                     0                0
## 3                0                     0                0
## 4                0                     1                0
## 5                0                     1                0
## 6                0                     0                0
##   item_family.POULTRY item_class.1004 item_class.1014 item_class.1120
## 1                   1               0               0               0
## 2                   0               0               0               0
## 3                   0               0               0               1
## 4                   0               0               1               0
## 5                   0               1               0               0
## 6                   0               0               0               0
##   item_class.1122 item_class.1124 item_class.1136 item_class.2302
```

```
## 1               0               0               0               0
## 2               0               0               0               0
## 3               0               0               0               0
## 4               0               0               0               0
## 5               0               0               0               0
## 6               0               1               0               0
##   item_class.2416 item_class.2502 item_class.2702 perishable.0
## 1               1               0               0            0
## 2               0               1               0            0
## 3               0               0               0            1
## 4               0               0               0            1
## 5               0               0               0            1
## 6               0               0               0            1
##   perishable.1 holiday_type.Additional holiday_type.Event
## 1            1                       0                 0
## 2            1                       0                 0
## 3            0                       0                 0
## 4            0                       0                 0
## 5            0                       0                 0
## 6            0                       0                 0
##   holiday_type.Holiday holiday_type.Normal.day holiday_type.Work.Day
## 1                    0                       1                     0
## 2                    0                       1                     0
## 3                    0                       1                     0
## 4                    0                       1                     0
## 5                    0                       1                     0
## 6                    0                       1                     0
##   transaction day.of.week.Fri day.of.week.Mon day.of.week.Sat
## 1        3487               0               0               0
## 2        3487               0               0               0
## 3        3487               0               0               0
## 4        3487               0               0               0
## 5        3487               0               0               0
## 6        3487               0               0               0
##   day.of.week.Sun day.of.week.Thu day.of.week.Tue day.of.week.Wed
## 1               0               0               0               1
## 2               0               0               0               1
## 3               0               0               0               1
## 4               0               0               0               1
## 5               0               0               0               1
## 6               0               0               0               1
##   day.of.month.1 day.of.month.2 day.of.month.3 day.of.month.4
## 1              0              1              0              0
## 2              0              1              0              0
## 3              0              1              0              0
## 4              0              1              0              0
## 5              0              1              0              0
## 6              0              1              0              0
##   day.of.month.5 day.of.month.6 day.of.month.7 day.of.month.8
## 1              0              0              0              0
## 2              0              0              0              0
## 3              0              0              0              0
## 4              0              0              0              0
## 5              0              0              0              0
## 6              0              0              0              0
##   day.of.month.9 day.of.month.10 day.of.month.11 day.of.month.12
## 1              0               0               0              0
## 2              0               0               0              0
```

```
## 3              0               0               0               0
## 4              0               0               0               0
## 5              0               0               0               0
## 6              0               0               0               0
##   day.of.month.13 day.of.month.14 day.of.month.15 day.of.month.16
## 1               0               0               0               0
## 2               0               0               0               0
## 3               0               0               0               0
## 4               0               0               0               0
## 5               0               0               0               0
## 6               0               0               0               0
##   day.of.month.17 day.of.month.18 day.of.month.19 day.of.month.20
## 1               0               0               0               0
## 2               0               0               0               0
## 3               0               0               0               0
## 4               0               0               0               0
## 5               0               0               0               0
## 6               0               0               0               0
##   day.of.month.21 day.of.month.22 day.of.month.23 day.of.month.24
## 1               0               0               0               0
## 2               0               0               0               0
## 3               0               0               0               0
## 4               0               0               0               0
## 5               0               0               0               0
## 6               0               0               0               0
##   day.of.month.25 day.of.month.26 day.of.month.27 day.of.month.28
## 1               0               0               0               0
## 2               0               0               0               0
## 3               0               0               0               0
## 4               0               0               0               0
## 5               0               0               0               0
## 6               0               0               0               0
##   day.of.month.29 day.of.month.30 day.of.month.31
## 1               0               0               0
## 2               0               0               0
## 3               0               0               0
## 4               0               0               0
## 5               0               0               0
## 6               0               0               0
```

splitting in to test and train

```
library(randomForest,quietly = TRUE)
```

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

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

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

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

```
library(caTools)
set.seed(123)
sample<-sample.split(trsf$unit_sales,SplitRatio = 0.7)
trainDF<-trsf[sample,]
testDF<-trsf[!sample,]
```

Random forest model

```
RF<-randomForest(unit_sales~.,data = trainDF,ntree=501, mtry = 3, nodesize = 10,
                 importance=TRUE)
RF
```

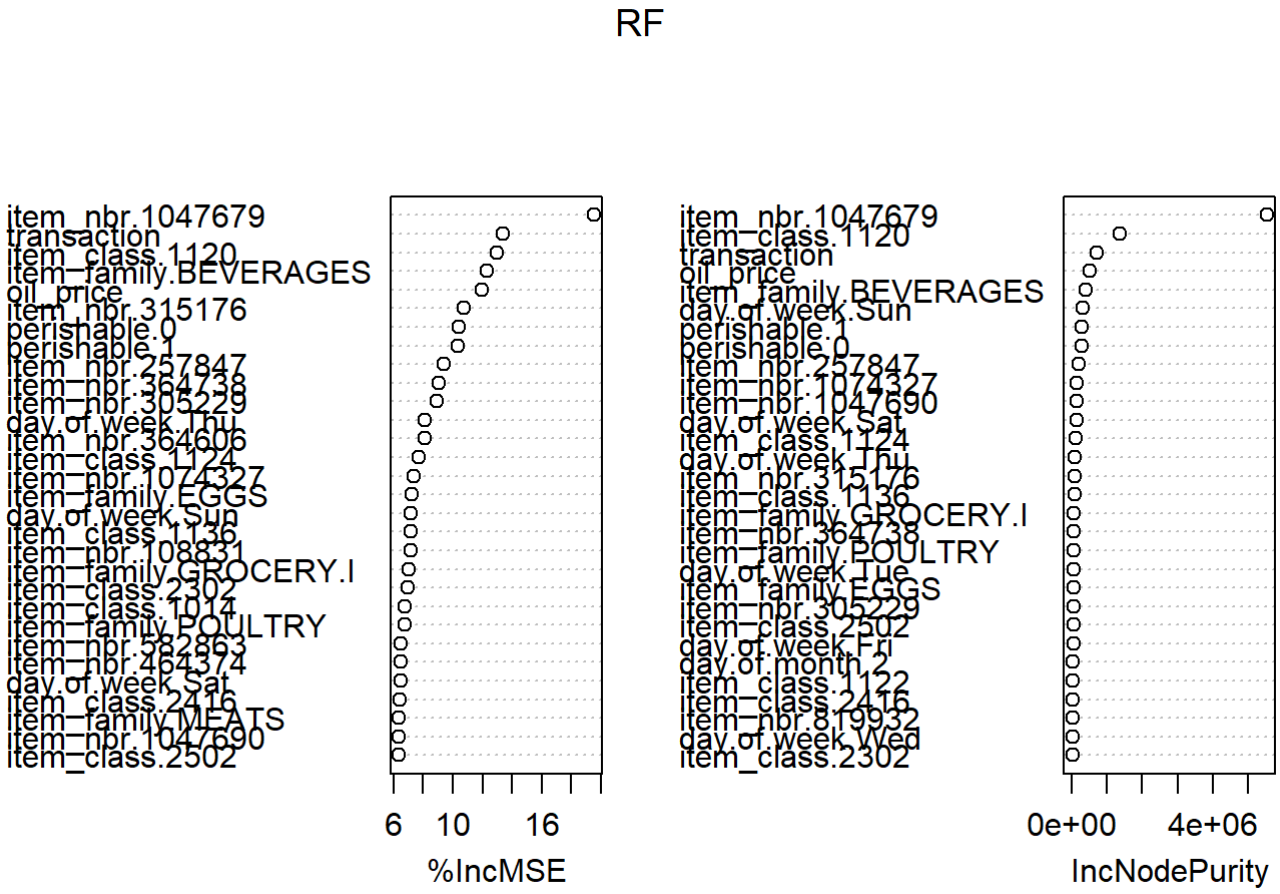
```
##
## Call:
##  randomForest(formula = unit_sales ~ ., data = trainDF, ntree = 501,      mtry = 3, nodesi
ze = 10, importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 501
## No. of variables tried at each split: 3
##
##           Mean of squared residuals: 4183.222
##                     % Var explained: 58.55
```

```
importance(RF)
```

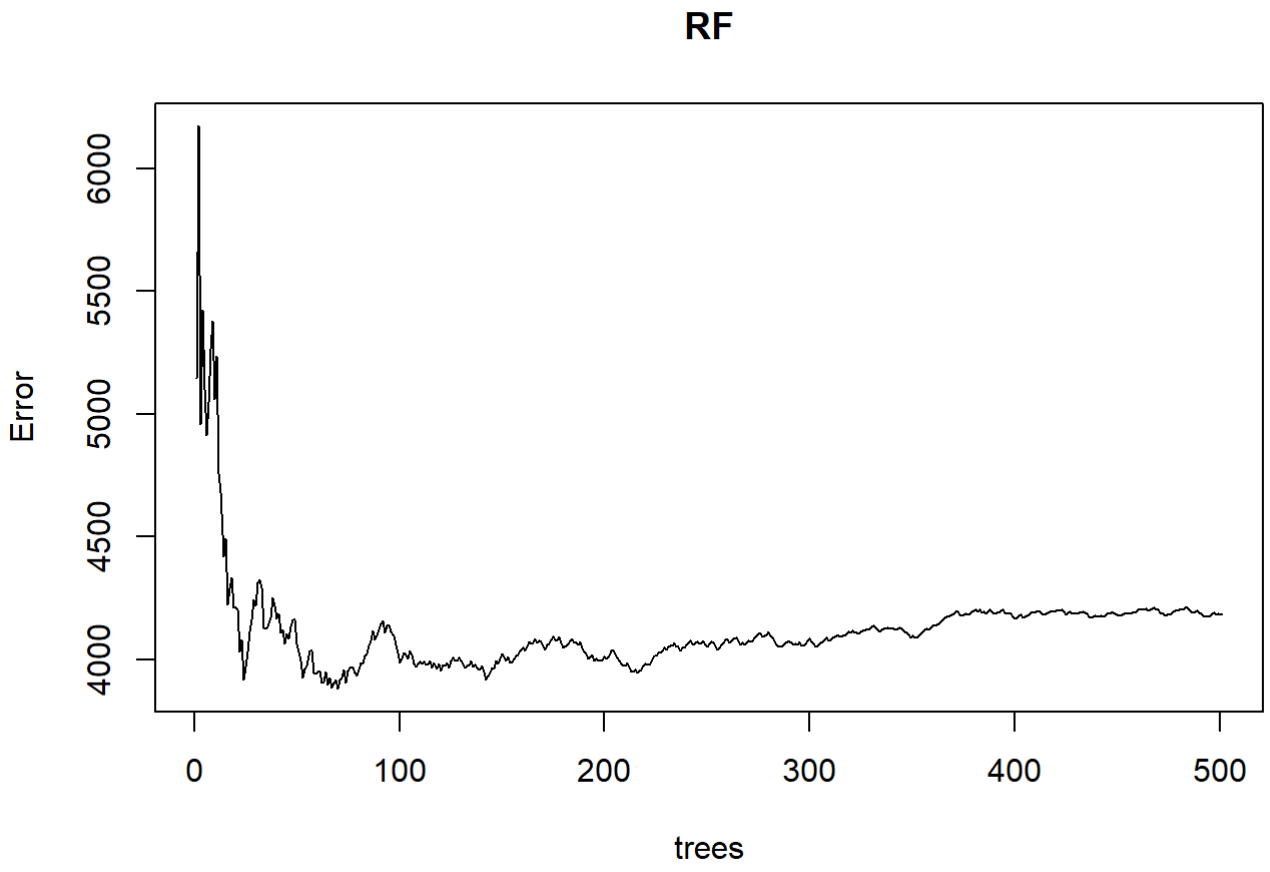
```
##                             %IncMSE IncNodePurity
## item_nbr.108831           7.1446346     33361.561
## item_nbr.208384           4.6858257     18690.949
## item_nbr.257847           9.3628041    203720.091
## item_nbr.305229           8.9442937     55636.075
## item_nbr.314384           6.1078915     24618.356
## item_nbr.315176          10.7368160     88116.543
## item_nbr.364606           8.1007608     21731.923
## item_nbr.364738           9.0758758     61213.981
## item_nbr.464374           6.4854453     28304.604
## item_nbr.502331           6.1441571     12923.144
## item_nbr.557256           4.4352935     22294.625
## item_nbr.582863           6.5112490     35452.318
## item_nbr.807493           5.3426405     23061.777
## item_nbr.819932           6.1698801     41068.943
## item_nbr.903286           5.8846974     19087.979
## item_nbr.1047679         19.5518169   5545734.707
## item_nbr.1047690          6.3552477    153672.580
## item_nbr.1066900          5.6840330     30685.775
## item_nbr.1066901          6.2741205     24212.869
## item_nbr.1074327          7.3492065    155623.886
## oil_price                11.9855696    523369.339
## item_family.BEVERAGES    12.3255489    389331.261
## item_family.BREAD.BAKERY  4.2528696     10256.543
## item_family.EGGS          7.2543344     55784.871
## item_family.GROCERY.I     6.9938113     64367.234
## item_family.MEATS         6.3627668     39372.688
## item_family.POULTRY       6.7259976     59669.175
## item_class.1004           6.1772829     27233.372
## item_class.1014           6.7279401     36678.067
## item_class.1120          12.9704185   1367189.544
## item_class.1122           5.4708977     42795.362
## item_class.1124           7.7248262    122854.179
## item_class.1136           7.1598339     77273.780
## item_class.2302           6.9302947     39689.316
## item_class.2416           6.4228715     41863.092
## item_class.2502           6.3453398     50008.585
## item_class.2702           4.3121741     11590.905
## perishable.0             10.4094488    277456.411
## perishable.1             10.3211679    293515.067
## holiday_type.Additional  -1.5401687      5402.042
## holiday_type.Event       -0.3380983      4703.896
## holiday_type.Holiday     -0.8543251      8293.883
## holiday_type.Normal.day  -0.7873432     15638.431
## holiday_type.Work.Day    -3.4411858     22920.829
## transaction              13.3965850    704249.135
## day.of.week.Fri           4.8118057     46311.276
## day.of.week.Mon          -0.4448523     34712.950
## day.of.week.Sat           6.4751400    144467.187
## day.of.week.Sun           7.1827460    307014.015
## day.of.week.Thu           8.1210746     92680.663
## day.of.week.Tue           4.6844488     57124.182
## day.of.week.Wed           4.1740135     40627.580
## day.of.month.1           -1.9043373     19168.284
## day.of.month.2            0.8416778     43822.693
## day.of.month.3           -3.7651876     29371.804
## day.of.month.4           -0.3175052     10592.609
```

```
## day.of.month.5          -4.4636413     16656.536
## day.of.month.6          -1.0312227     10011.815
## day.of.month.7          -1.8551531     28491.120
## day.of.month.8          -2.6019449     14629.334
## day.of.month.9          -4.1001359     15527.156
## day.of.month.10         -1.9841791     21403.138
## day.of.month.11         -2.1465758     12397.020
## day.of.month.12         -4.2042864     25071.796
## day.of.month.13         -3.9134826     12090.003
## day.of.month.14         -4.8565971     17440.605
## day.of.month.15         -4.0400680     13643.606
## day.of.month.16         -1.8828978     15175.488
## day.of.month.17         -3.0450402     15251.016
## day.of.month.18         -0.2000557     15770.341
## day.of.month.19         -4.2740796     28343.644
## day.of.month.20         -3.2541072     15650.250
## day.of.month.21         -5.6402202     13016.489
## day.of.month.22          2.1566420     13867.853
## day.of.month.23         -4.3621241     16581.770
## day.of.month.24         -1.6858180     18348.468
## day.of.month.25         -3.3017027     11286.629
## day.of.month.26         -1.0663309     12631.327
## day.of.month.27         -2.5725010     11538.154
## day.of.month.28         -2.7249006     16523.523
## day.of.month.29         -3.8105301     17964.922
## day.of.month.30         -3.8023473     14449.936
## day.of.month.31         -2.7960773      6819.755
```

```
varImpPlot(RF)
```



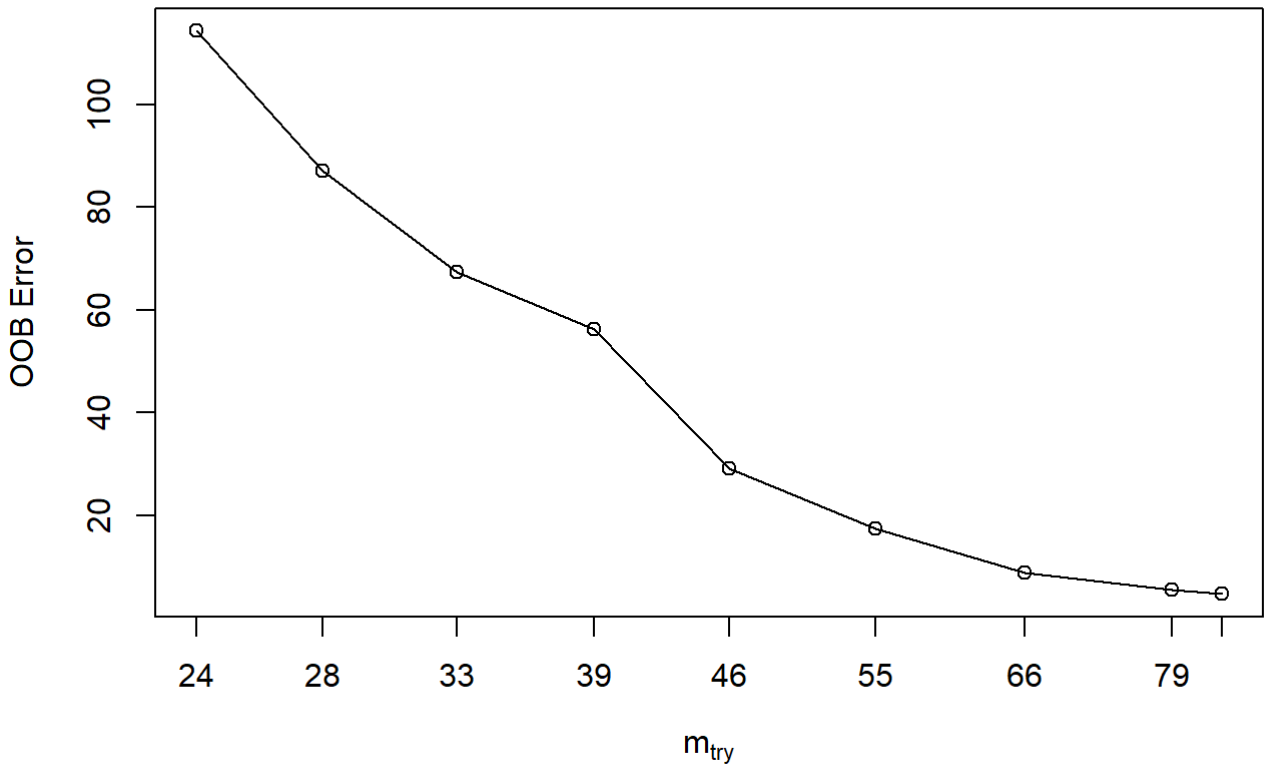
```
plot(RF)
```



tunning for optimal mtry

```
bestmtry<-tuneRF(trainDF,trainDF$unit_sales,ntreeTry = 200,stepFactor = 1.2,
                 improve = 0.01,trace = T,plot = T,importance = TRUE,doBest = TRUE)
```

```
## mtry = 28  OOB error = 87.05425
## Searching left ...
## mtry = 24    OOB error = 114.267
## -0.3125956 0.01
## Searching right ...
## mtry = 33    OOB error = 67.3121
## 0.2267798 0.01
## mtry = 39    OOB error = 56.21306
## 0.1648892 0.01
## mtry = 46    OOB error = 29.14721
## 0.4814869 0.01
## mtry = 55    OOB error = 17.45923
## 0.4009982 0.01
## mtry = 66    OOB error = 8.776604
## 0.4973087 0.01
## mtry = 79    OOB error = 5.539438
## 0.3688403 0.01
## mtry = 84    OOB error = 4.681556
## 0.1548681 0.01
```



after tunning the best mtry is 39

```
RF<-randomForest(unit_sales~.,data = trainDF,ntree=501, mtry = 39, nodesize = 10,
                 importance=TRUE)
RF
```

```
##
## Call:
##  randomForest(formula = unit_sales ~ ., data = trainDF, ntree = 501,      mtry = 39, nodes
ize = 10, importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 501
## No. of variables tried at each split: 39
##
##           Mean of squared residuals: 1703.181
##                     % Var explained: 83.13
```

```
PredictionsWithresponse<- predict(RF, testDF,type = "response")

predictions=PredictionsWithresponse
actual=testDF$unit_sales
```

Accuracy measurement

```
caret::RMSE(predictions,actual)
```

```
## [1] 37.44615
```

```
caret::R2(predictions,actual)
```

```
## [1] 0.7407455
```

```
MLmetrics::MAPE(predictions,actual)
```

```
## [1] 0.5753763
```