**BIKE SHARING USING R PROGAMMING**

str(train)

**output**

'data.frame': 10886 obs. of 12 variables:

$ datetime : chr "2011-01-01 00:00:00" "2011-01-01 01:00:00" "2011-01-01 02:00:00" "2011-01-01 03:00:00" ...

$ season : int 1 1 1 1 1 1 1 1 1 1 ...

$ holiday : int 0 0 0 0 0 0 0 0 0 0 ...

$ workingday: int 0 0 0 0 0 0 0 0 0 0 ...

$ weather : int 1 1 1 1 1 2 1 1 1 1 ...

$ temp : num 9.84 9.02 9.02 9.84 9.84 ...

$ atemp : num 14.4 13.6 13.6 14.4 14.4 ...

$ humidity : int 81 80 80 75 75 75 80 86 75 76 ...

$ windspeed : num 0 0 0 0 0 ...

$ casual : int 3 8 5 3 0 0 2 1 1 8 ...

$ registered: int 13 32 27 10 1 1 0 2 7 6 ...

$ count : int 16 40 32 13 1 1 2 3 8 14 ...

str(test)

**output**

'data.frame': 6493 obs. of 9 variables:

$ datetime : chr "2011-01-20 00:00:00" "2011-01-20 01:00:00" "2011-01-20 02:00:00" "2011-01-20 03:00:00" ...

$ season : int 1 1 1 1 1 1 1 1 1 1 ...

$ holiday : int 0 0 0 0 0 0 0 0 0 0 ...

$ workingday: int 1 1 1 1 1 1 1 1 1 1 ...

$ weather : int 1 1 1 1 1 1 1 1 1 2 ...

$ temp : num 10.7 10.7 10.7 10.7 10.7 ...

$ atemp : num 11.4 13.6 13.6 12.9 12.9 ...

$ humidity : int 56 56 56 56 56 60 60 55 55 52 ...

$ windspeed : num 26 0 0 11 11 ...

# getting some information about the combined data

str(data)

**output**

'data.frame': 17379 obs. of 12 variables:

$ datetime : chr "2011-01-01 00:00:00" "2011-01-01 01:00:00" "2011-01-01 02:00:00" "2011-01-01 03:00:00" ...

$ season : int 1 1 1 1 1 1 1 1 1 1 ...

$ holiday : int 0 0 0 0 0 0 0 0 0 0 ...

$ workingday: int 0 0 0 0 0 0 0 0 0 0 ...

$ weather : int 1 1 1 1 1 2 1 1 1 1 ...

$ temp : num 9.84 9.02 9.02 9.84 9.84 ...

$ atemp : num 14.4 13.6 13.6 14.4 14.4 ...

$ humidity : int 81 80 80 75 75 75 80 86 75 76 ...

$ windspeed : num 0 0 0 0 0 ...

$ casual : num 3 8 5 3 0 0 2 1 1 8 ...

$ registered: num 13 32 27 10 1 1 0 2 7 6 ...

$ count : num 16 40 32 13 1 1 2 3 8 14 ...

summary(data)

datetime season holiday workingday

Length:17379 Min. :1.000 Min. :0.00000 Min. :0.0000

Class :character 1st Qu.:2.000 1st Qu.:0.00000 1st Qu.:0.0000

Mode :character Median :3.000 Median :0.00000 Median :1.0000

Mean :2.502 Mean :0.02877 Mean :0.6827

3rd Qu.:3.000 3rd Qu.:0.00000 3rd Qu.:1.0000

Max. :4.000 Max. :1.00000 Max. :1.0000

weather temp atemp humidity

Min. :1.000 Min. : 0.82 Min. : 0.00 Min. : 0.00

1st Qu.:1.000 1st Qu.:13.94 1st Qu.:16.66 1st Qu.: 48.00

Median :1.000 Median :20.50 Median :24.24 Median : 63.00

Mean :1.425 Mean :20.38 Mean :23.79 Mean : 62.72

3rd Qu.:2.000 3rd Qu.:27.06 3rd Qu.:31.06 3rd Qu.: 78.00

Max. :4.000 Max. :41.00 Max. :50.00 Max. :100.00

windspeed casual registered count

Min. : 0.000 Min. : 0.00 Min. : 0.00 Min. : 0

1st Qu.: 7.002 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.: 0

Median :12.998 Median : 3.00 Median : 23.00 Median : 28

Mean :12.737 Mean : 22.56 Mean : 97.44 Mean :120

3rd Qu.:16.998 3rd Qu.: 26.00 3rd Qu.:155.00 3rd Qu.:192

Max. :56.997 Max. :367.00 Max. :886.00 Max. :977

# creating some boxplots on the count of rentals

boxplot(train$count~train$hour,xlab="hour", ylab="count of users")



boxplot(train$casual~train$hour,xlab="hour", ylab="casual users")

boxplot(train$registered~train$hour,xlab="hour", ylab="registered users")



# creating boxplots for rentals with different variables to see the variation

boxplot(train$registered~train$day,xlab="day", ylab="registered users")



boxplot(train$casual~train$day,xlab="day", ylab="casual users")



boxplot(train$registered~train$weather,xlab="weather", ylab="registered users")

boxplot(train$casual~train$weather,xlab="weather", ylab="casual users")



boxplot(train$registered~train$temp,xlab="temp", ylab="registered users")



boxplot(train$casual~train$temp,xlab="temp", ylab="casual users")



# again some boxplots with different variables

# these boxplots give important information about the dependent variable with respect to the independent variables

boxplot(train$registered~train$year,xlab="year", ylab="registered users"boxplot(train$casual~train$year,xlab="year", ylab="casual users")



boxplot(train$registered~train$windspeed,xlab="year", ylab="registered users")



boxplot(train$casual~train$windspeed,xlab="year", ylab="casual users")



boxplot(train$registered~train$humidity,xlab="humidity", ylab="registered users")



boxplot(train$casual~train$humidity,xlab="humidity", ylab="casual users")



#using decision trees for binning some variables, this was a really important step in feature engineering

d=rpart(registered~hour,data=train)

fancyRpartPlot(d)

d=rpart(casual~hour,data=train)

fancyRpartPlot(d)



f=rpart(registered~temp,data=train)

fancyRpartPlot(f)

f=rpart(casual~temp,data=train)

fancyRpartPlot(f)



table(data$year\_part)

1 5

8645 8734

plot(train$temp,train$count)



# dividing total data depending on windspeed to impute/predict the missing values

table(data$windspeed==0)

FALSE TRUE

15199 2180

set.seed(415)

fit <- randomForest(windspeed ~ season+weather +humidity +month+temp+ year+atemp, data=wind\_1,importance=TRUE, ntree=250)

fit

Call:

randomForest(formula = windspeed ~ season + weather + humidity + month + temp + year + atemp, data = wind\_1, importance = TRUE, ntree = 250)

Type of random forest: regression

Number of trees: 250

No. of variables tried at each split: 2

Mean of squared residuals: 28.63599

% Var explained: 42.97

str(data)

'data.frame': 17379 obs. of 24 variables:

$ datetime : chr "2011-01-01 00:00:00" "2011-01-01 01:00:00" "2011-01-01 02:00:00" "2011-01-01 03:00:00" ...

$ season : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...

$ holiday : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...

$ workingday: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 2 ...

$ weather : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...

$ temp : num 9.84 9.02 9.02 9.84 9.84 ...

$ atemp : num 14.4 13.6 13.6 14.4 14.4 ...

$ humidity : int 81 80 80 75 75 80 86 75 76 47 ...

$ windspeed : num 9.03 9.05 9.05 9.15 9.15 ...

$ casual : num 3 8 5 3 0 2 1 1 8 8 ...

$ registered: num 13 32 27 10 1 0 2 7 6 102 ...

$ count : num 16 40 32 13 1 2 3 8 14 110 ...

$ hour : int 1 2 3 4 5 7 8 9 10 20 ...

$ day : chr "Saturday" "Saturday" "Saturday" "Saturday" ...

$ year : Factor w/ 2 levels "2011","2012": 1 1 1 1 1 1 1 1 1 1 ...

$ day\_part : num 0 0 0 0 0 0 0 0 0 0 ...

$ dp\_reg : num 1 1 1 1 1 1 4 5 3 6 ...

$ dp\_cas : num 1 1 1 1 1 1 1 2 3 4 ...

$ temp\_reg : num 1 1 1 1 1 1 1 1 2 1 ...

$ temp\_cas : num 1 1 1 1 1 1 1 1 1 1 ...

$ year\_part : num 1 1 1 1 1 1 1 1 1 1 ...

$ day\_type : chr "weekend" "weekend" "weekend" "weekend" ...

$ month : int 1 1 1 1 1 1 1 1 1 1 ...

$ weekend : num 1 1 1 1 1 1 1 1 1 0 ...

boxplot(train$logreg~train$weather,xlab="weather", ylab="registered users")



boxplot(train$logreg~train$season,xlab="season", ylab="registered users")



# final model building using random forest

# note that we build different models for predicting for registered and casual users

# this was seen as giving best result after a lot of experimentation

set.seed(415)

fit1 <- randomForest(logreg ~ hour +workingday+day+holiday+ day\_type +temp\_reg+humidity+atemp+windspeed+season+weather+dp\_reg+weekend+year+year\_part, data=train,importance=TRUE, ntree=250)

fit1

Call:

randomForest(formula = logreg ~ hour + workingday + day + holiday + day\_type + temp\_reg + humidity + atemp + windspeed + season + weather + dp\_reg + weekend + year + year\_part, data = train, importance = TRUE, ntree = 250)

Type of random forest: regression

Number of trees: 250

No. of variables tried at each split: 5

Mean of squared residuals: 0.09932523

% Var explained: 94.93

set.seed(415)

fit2 <- randomForest(logcas ~hour + day\_type+day+humidity+atemp+temp\_cas+windspeed+season+weather+holiday+workingday+dp\_cas+weekend+year+year\_part, data=train,importance=TRUE, ntree=250)

fit2

Call:

randomForest(formula = logcas ~ hour + day\_type + day + humidity + atemp + temp\_cas + windspeed + season + weather + holiday + workingday + dp\_cas + weekend + year + year\_part, data = train, importance = TRUE, ntree = 250)

Type of random forest: regression

Number of trees: 250

No. of variables tried at each split: 5

Mean of squared residuals: 0.2364734

% Var explained: 89.36