# Models Cross Validation

*Sanatan Das*

*May 24, 2018*

## Contents

# Initial Data Preparation

## Load the Data

```
# load the data set from excel file
default_rates <- read_excel("C:/view/opt/apps/git/compscix-415-1-assignments/data/peps3xx.xls")

# add factor to the 'char' columns
default_rates$Name <- as.factor(default_rates$Name)
default_rates$State <- as.factor(default_rates$State)
default_rates$ZipCode <- as.factor(default_rates$ZipCode)
default_rates$ProgLength <- as.factor(default_rates$ProgLength)
default_rates$SchoolType <- as.factor(default_rates$SchoolType)
default_rates$EthnicCode <- as.factor(default_rates$EthnicCode)
default_rates$Prate <- as.factor(default_rates$Prate)
default_rates$CongDis <- as.factor(default_rates$CongDis)
# convert the columns to 'double' data type
default_rates$Drate <- as.double(default_rates$Drate)
default_rates$Num <- as.double(default_rates$Num)
default_rates$Denom <- as.double(default_rates$Denom)
```

## Split the Data (training set and test set)

```
# split the data (training data - 80% and test data - 20%)
set.seed(29283)
# Let's create our training set using sample_frac.
train_set <- default_rates %>% sample_frac(0.8)
# Print train set
train_set
```

```
## # A tibble: 18,372 x 20
##    RecordId OPEID  Name    Address    City  State StateDesc ZipCode ZipExt
##       <dbl> <chr>  <fct>   <chr>      <chr> <fct> <chr>     <fct>   <chr>
##  1      335 001170 CLAREM~ 500 EAST ~ CLAR~ CA    CALIFORN~ 91711   6400
##  2    10023 022704 SOUTHE~ 2545 VALL~ BIRM~ AL    ALABAMA   35244   2083
##  3     1899 001969 KENTUC~ 3000 FRED~ OWEN~ KY    KENTUCKY  42301   6057
##  4     9062 020788 COLLEC~ 7353 SOUT~ MIDV~ UT    UTAH      84047   3022
##  5    22438 037063 AMERIC~ 5000C COC~ MARG~ FL    FLORIDA   33063   3901
##  6    18447 001785 ANDERS~ 1100 EAST~ ANDE~ IN    INDIANA   46012   3495
##  7    19902 003509 UNIVER~ SOUTHERN ~ MEMP~ TN    TENNESSEE 38152   4611
##  8    20830 009192 SIERRA~ 999 TAHOE~ INCL~ NV    NEVADA    89451   0000
##  9     1822 001936 NEOSHO~ 800 WEST ~ CHAN~ KS    KANSAS    66720   2699
## 10    17798 041559 AVEDA ~ 6020 EAST~ INDI~ IN    INDIANA   46250   4746
## # ... with 18,362 more rows, and 11 more variables: ProgLength <fct>,
## #   SchoolType <fct>, Year <chr>, Num <dbl>, Denom <dbl>, Drate <dbl>,
## #   Prate <fct>, EthnicCode <fct>, CongDis <fct>, Region <chr>, Avg <chr>
```

```
# let's create our testing set using the RecordId column. Fill in the blanks.
test_set <- default_rates %>% filter(!(default_rates$RecordId %in% train_set$RecordId))
# Print test set
test_set
```

```
## # A tibble: 4,593 x 20
```

```
##    RecordId OPEID  Name     Address   City  State StateDesc ZipCode ZipExt
##       <dbl> <chr>  <fct>    <chr>     <chr> <fct> <chr>     <fct>   <chr>
##  1     4.00 001003 FAULKNE~ 5345 ATL~ MONT~ AL    ALABAMA   36109   3398
##  2     9.00 001004 UNIVERS~ PALMER C~ MONT~ AL    ALABAMA   35115   6000
##  3    11.0  001005 ALABAMA~ 915 SOUT~ MONT~ AL    ALABAMA   36104   5714
##  4    14.0  001007 CENTRAL~ 1675 CHE~ ALEX~ AL    ALABAMA   35010   0000
##  5    15.0  001007 CENTRAL~ 1675 CHE~ ALEX~ AL    ALABAMA   35010   0000
##  6    22.0  001012 BIRMING~ 900 ARKA~ BIRM~ AL    ALABAMA   35254   0002
##  7    23.0  001012 BIRMING~ 900 ARKA~ BIRM~ AL    ALABAMA   35254   0002
##  8    36.0  001019 HUNTING~ 1500 EAS~ MONT~ AL    ALABAMA   36106   2148
##  9    41.0  001022 JEFFERS~ 2601 CAR~ BIRM~ AL    ALABAMA   35215   3098
## 10    44.0  001023 JUDSON ~ 302 BIBB~ MARI~ AL    ALABAMA   36756   2504
## # ... with 4,583 more rows, and 11 more variables: ProgLength <fct>,
## #   SchoolType <fct>, Year <chr>, Num <dbl>, Denom <dbl>, Drate <dbl>,
## #   Prate <fct>, EthnicCode <fct>, CongDis <fct>, Region <chr>, Avg <chr>
```

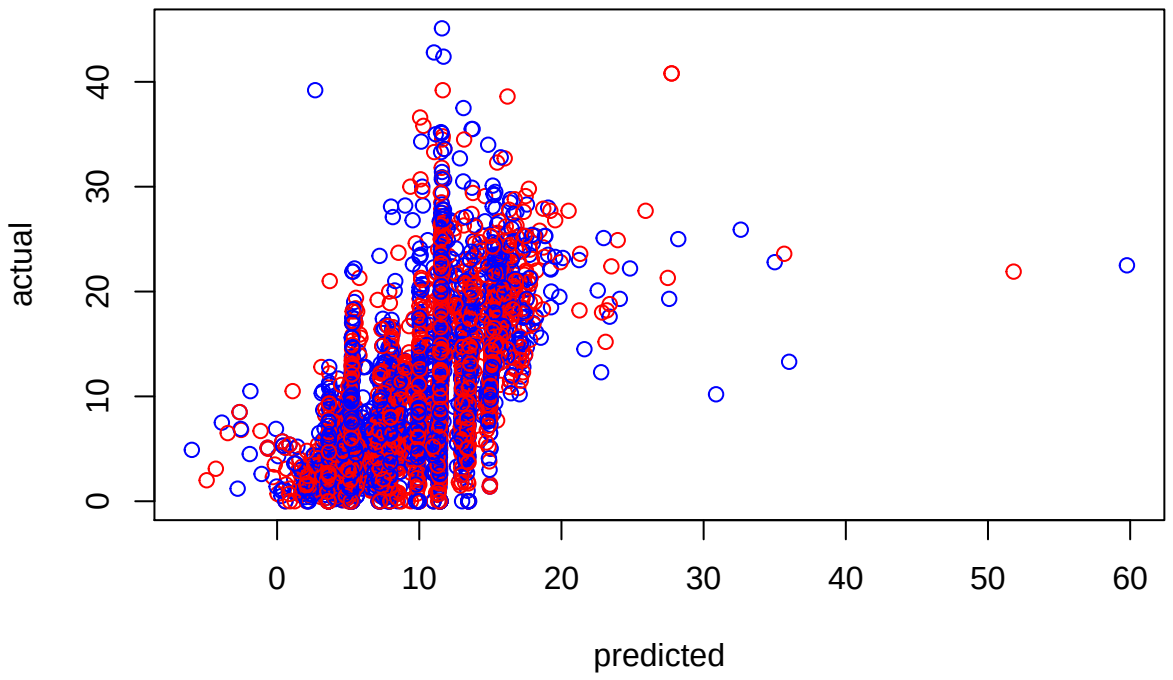
# Evaluation of Linear Regression model

## Predict the test data (lm)

```
lm_predict <- predict(lm_0, test_set)
```

## Predicted vs Actual

```
plot(lm_predict,test_set$Drate, col=c('red', 'blue'), xlab="predicted", ylab="actual")
```




### Mean Absolute Error

```
lm_diffs <- lm_predict - test_set$Drate
lm_mae <- mae(lm_diffs)
lm_mae
```

```
## [1] 3.88506
```

### Root Mean Squared Error

```
lm_rmse <- rmse(lm_diffs)
lm_rmse
```

```
## [1] 5.346097
```

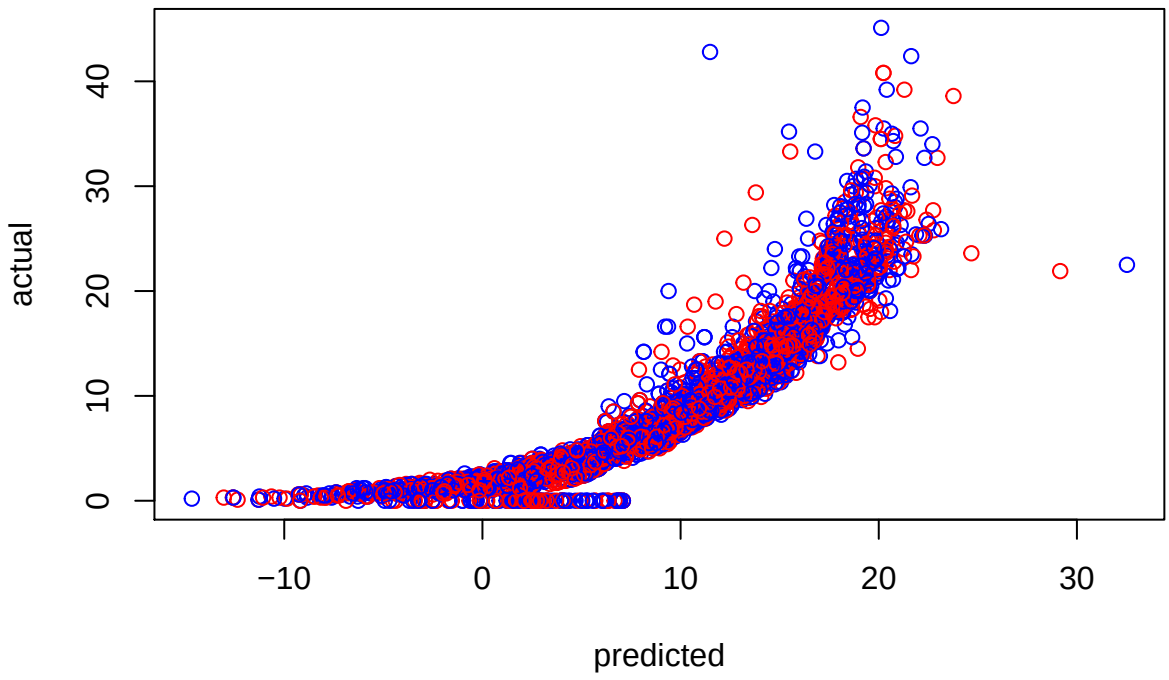
## Evaluation of MARS model

### Predict the test data (mars)

```
mars_predict <- predict(mars_0, test_set)[, 1]
```

### Predicted vs Actual

```
plot(mars_predict,test_set$Drate, col=c('red', 'blue'), xlab="predicted", ylab="actual")
```




### Mean Absolute Error

```
mars_diffs <- mars_predict - test_set$Drate
mars_mae <- mae(mars_diffs)
mars_mae
```

```
## [1] 2.119824
```

### Root Mean Squared Error

```
mars_rmse <- rmse(mars_diffs)
mars_rmse
```

```
## [1] 3.068444
```

# Final Note

## Model Selection

From the above validation, we see that the Linear regression model has MAE = 3.88506 and RMSE = 5.346097 where the MARS model has MAE = 2.119824 and RMSE = 3.068444. So, the MARS model performs better on the test data. We will use the MARS model on our final application.