~~~~
---
title: "LinearReggresion"
output: html_notebook
---

# Loading ggplot
```{r}
library(ggplot2)
```



### Print the head of the dataset

```{r}
path <- "/Users/pulkitbatra/Downloads/archive-2/train.csv"
trainingSet = read.csv(path)

```

Check for NA and missing values
is.na return a vector with value TT for missing values.

```{r}
numberOfNA = length(which(is.na(trainingSet)==T))
if(numberOfNA > 0) {
  cat('Number of missing values found: ', numberOfNA)
  cat('\nRemoving missing values...')
  trainingSet = trainingSet[complete.cases(trainingSet), ]
}

```

Check for outliers
Divide the graph area in 2 columns

```{r}
par(mfrow = c(1, 2))
# Boxplot for X
boxplot(trainingSet$x, main='X', sub=paste('Outliers: ',
boxplot.stats(trainingSet$x)$out))
# Boxplot for Y
boxplot(trainingSet$y, main='Y', sub=paste('Outliers: ',
boxplot.stats(trainingSet$y)$out))
```

```{r}
cor(trainingSet$x, trainingSet$y)
```

 0.99 shows a very strong relation.
```{r}
regressor = lm(formula = y ~.,
               data = trainingSet)
```

```{r}
summary(regressor)
```


 plot

```{r}
ggplot() +
  geom_point(aes(x = trainingSet$x, y = trainingSet$y),
             colour = 'red') +
  geom_line(aes(x = trainingSet$x, y = predict(regressor, newdata = trainingSet)),
~~~~

~~~
          colour = 'blue') +
  ggtitle('X vs Y (Training set)') +
  xlab('X') +
  ylab('Y')
```


 ## Test

```{r}
testPath <- "/Users/pulkitbatra/Downloads/archive-2/test.csv"
testSet = read.csv(testPath)

y_pred = predict(regressor, newdata = testSet)
```


 Visualsing the result

```{r}
ggplot() +
  geom_point(aes(x = testSet$x, y = testSet$y),
             colour = 'red') +
  geom_line(aes(x = trainingSet$x, y = predict(regressor, newdata = trainingSet)),
            colour = 'blue') +
  ggtitle('X vs Y (Test set)') +
  xlab('X') +
  ylab('Y')
```


 # Plot shows model was a good fit.

```{r}
compare <- cbind (actual=testSet$x, y_pred)  # combine actual and predicted
mean (apply(compare, 1, min)/apply(compare, 1, max))
mean(0.9,0.9,0.9,0.9)
```


 ### Check for residual mean and distribution

```{r}
plot(trainingSet$y, resid(regressor),
     ylab="Residuals", xlab="Price",
     main="Residual plot")
mean(regressor$residuals)
```
~~~