# Experiment No.11

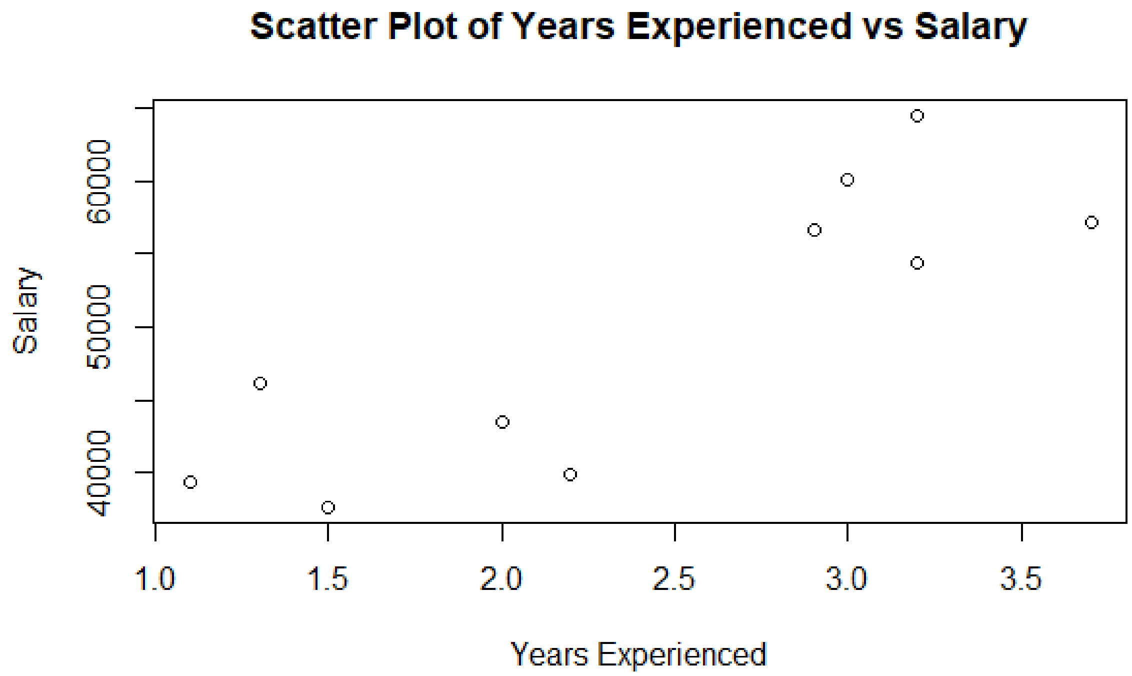
**1.Create the Data Frame:-**

```
data <- data.frame(
Years_Exp = c(1.1, 1.3, 1.5, 2.0, 2.2, 2.9, 3.0, 3.2, 3.2, 3.7),
Salary = c(39343.00, 46205.00, 37731.00, 43525.00,
                              39891.00, 56642.00, 60150.00, 54445.00, 64445.00, 57189.00)
)
```

2. **Scatter plot of the given dataset:-**

```
plot(data$Years_Exp, data$Salary,
         xlab = "Years Experienced",
         ylab = "Salary",
         main = "Scatter Plot of Years Experienced vs Salary")
```

**Output:**



**3.Implement Simple Linear Regression**

```
install.packages('caTools')

library(caTools)

split = sample.split(data$Salary, SplitRatio = 0.7)

trainingset = subset(data, split == TRUE)

testset = subset(data, split == FALSE)

lm.r= lm(formula = Salary ~ Years_Exp,

                    data = trainingset)

summary(lm.r)
```

**Output:**

```
Call:
lm(formula = Salary ~ Years_Exp, data = trainingset)
Residuals:
     1       2       3       5       6       8      10 
  463.1  5879.1 -4041.0 -6942.0  4748.0   381.9  -489.1 
Coefficients:
            Estimate Std. Error t value Pr(>|t|)   
(Intercept)    30927       4877   6.341  0.00144 **
Years_Exp       7230       1983   3.645  0.01482 * 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Residual standard error: 4944 on 5 degrees of freedom
Multiple R-squared:  0.7266,	Adjusted R-squared:  0.6719 
F-statistic: 13.29 on 1 and 5 DF,  p-value: 0.01482
```

**4. Predict values using predict function:-**

```
new_data <- data.frame(Years_Exp = c(4.0, 4.5, 5.0))

predicted_salaries <- predict(lm.r, newdata = new_data)

print(predicted_salaries)
```

**Output:**

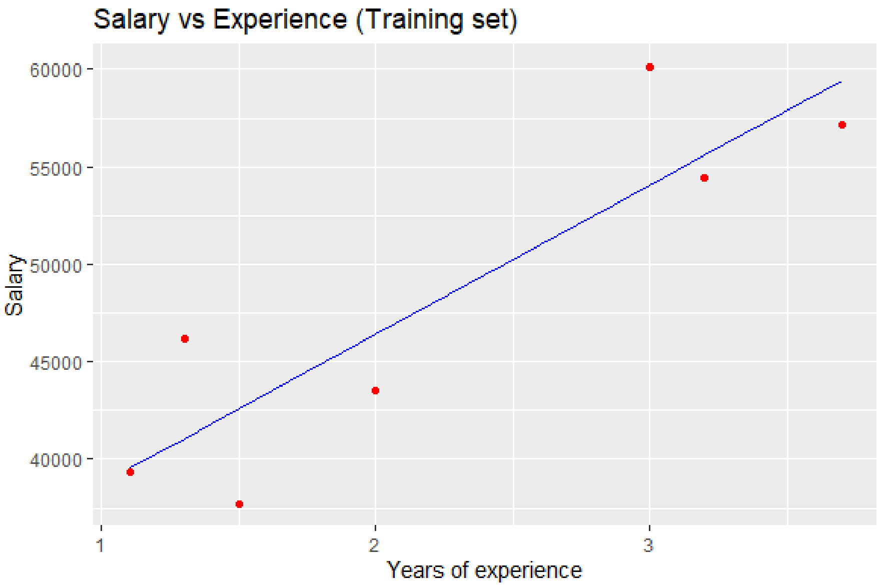
```
       1        2        3 
65673.14 70227.40 74781.66 
```

### 5. Visualizing the Training set results:-

```
# Visualising the Training set results
ggplot() + geom_point(aes(x = trainingset$Years_Ex,
                                            y = trainingset$Salary), colour = 'red') +
geom_line(aes(x = trainingset$Years_Ex,
                              y = predict(lm.r, newdata = trainingset)), colour = 'blue') +

ggtitle('Salary vs Experience (Training set)') +
xlab('Years of experience') +
ylab('Salary')
```

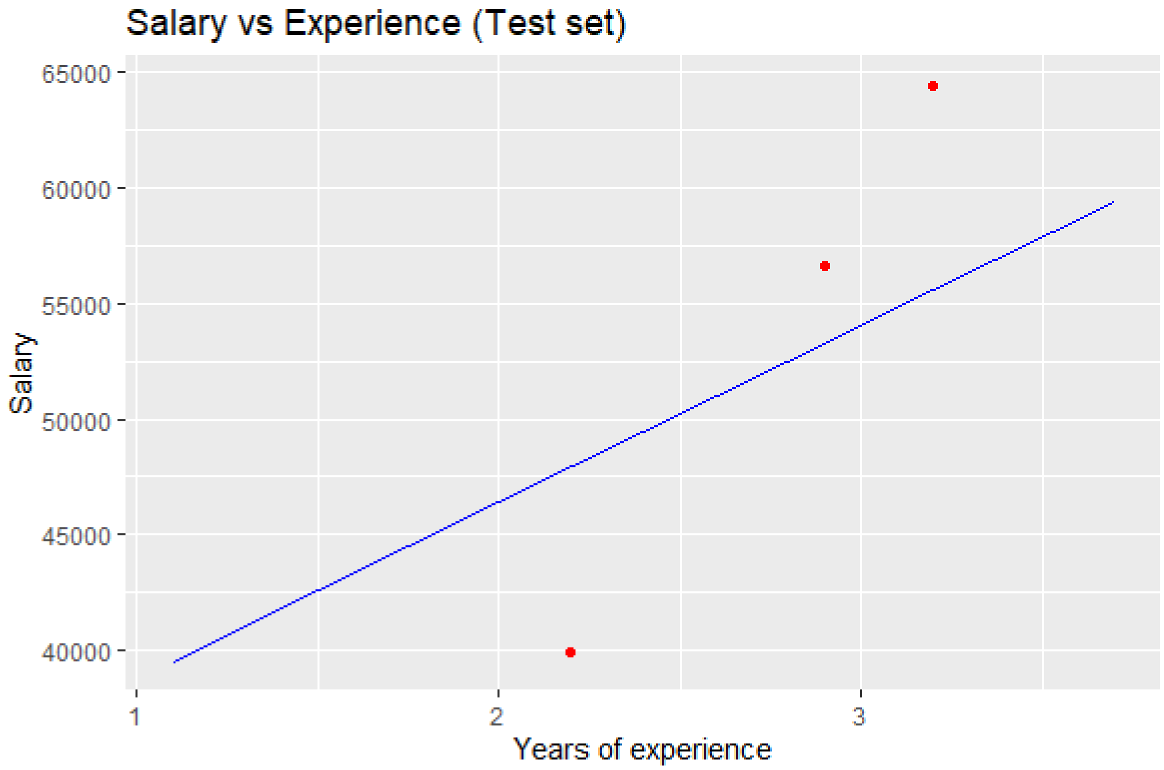
**Output:**



### 6.Visualizing the Testing set results:-

```
ggplot() +
geom_point(aes(x = testset$Years_Exp, y = testset$Salary),
                    colour = 'red') +
geom_line(aes(x = trainingset$Years_Exp,
                              y = predict(lm.r, newdata = trainingset)),
                              colour = 'blue') +
ggtitle('Salary vs Experience (Test set)') +
xlab('Years of experience') +
ylab('Salary')
```

**Output:**

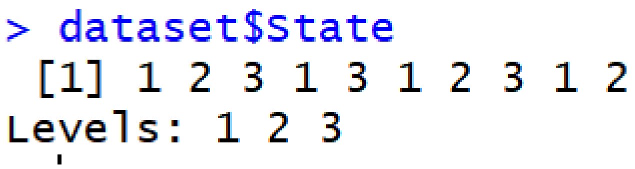


- **Implementation of Multi Linear Regression**

**1.Encoding Categorical Data**

```
dataset = read.csv('data2.csv')
dataset$State = factor(dataset$State,
                                                  levels = c('New York', 'California', 'Florida'),
                                                  labels = c(1, 2, 3))
dataset$State
```

**Output:**



```
> dataset$State
 [1] 1 2 3 1 3 1 2 3 1 2
Levels: 1 2 3
```

**2. Predicting Results**

library(caTools)

set.seed(123)

split = sample.split(dataset$Profit, SplitRatio = 0.8)

training_set = subset(dataset, split == TRUE)

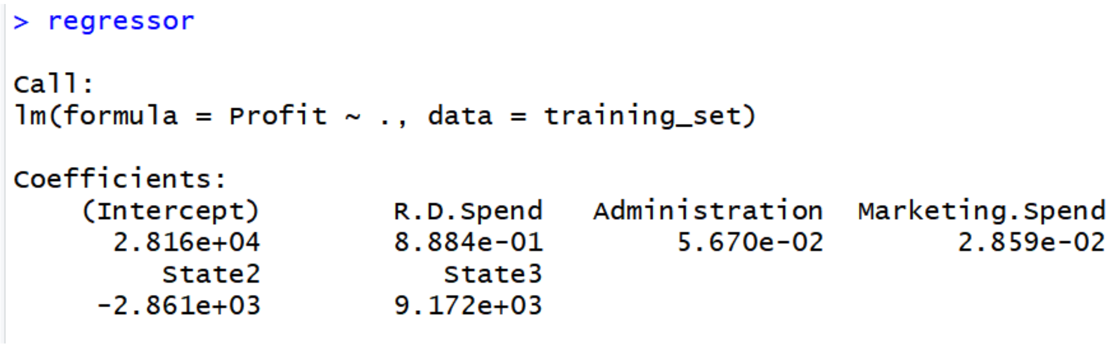
test_set = subset(dataset, split == FALSE)

regressor = lm(formula = Profit ~ .,

data = training_set)

y_pred = predict(regressor, newdata = test_set)

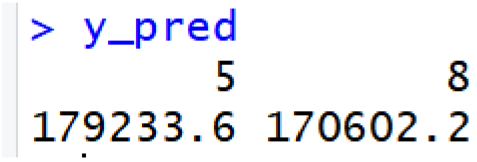
**Output:**



```
> regressor

Call:
lm(formula = Profit ~ ., data = training_set)

Coefficients:
    (Intercept)        R.D.Spend   Administration  Marketing.Spend
      2.816e+04        8.884e-01        5.670e-02        2.859e-02
         State2           State3
     -2.861e+03        9.172e+03
```



```
> y_pred
       5        8
179233.6 170602.2
```