

Temperature and Seoul Bike

Aim:

- To demonstrate how to do a machine learning in R for beginners (no prerequisite knowledge)
- To use a linear regression model to predict Seoul bike rent demand from temperature
- Only necessary codes are included and codes such as making plot neat are excluded

Code written by: Ggodhuri AI (Youtuber)

- URL for the youtube: https://www.youtube.com/channel/UCUdyx4YZ_bkBmBS3aq_7Xag

Data set source:

- Sathishkumar V E, Jangwoo Park, and Yongyun Cho. 'Using data mining techniques for bike sharing demand prediction in metropolitan city.' Computer Communications, Vol.153, pp.353-366, March, 2020
- Sathishkumar V E and Yongyun Cho. 'A rule-based model for Seoul Bike sharing demand prediction using weather data' European Journal of Remote Sensing, pp. 1-18, Feb, 2020
- Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science.
- URL: <https://archive.ics.uci.edu/ml/datasets/Seoul+Bike+Sharing+Demand>

Load Data

```
# Before loading data, the two column names need to be changed for the encoding issue.
# Temperature(?C) -> Temperature
# Dew point temperature(?C) -> Dew point temperature

data_frame = read.csv('SeoulBikeData.csv')
summary(data_frame)
```

```
##      Date      Rented.Bike.Count      Hour      Temperature
## Length:8760   Min.    :  0.0   Min.    : 0.00   Min.    : -17.80
## Class :character 1st Qu.: 191.0   1st Qu.: 5.75   1st Qu.:  3.50
## Mode  :character Median : 504.5   Median :11.50   Median : 13.70
##              Mean   : 704.6   Mean   :11.50   Mean   : 12.88
##              3rd Qu.:1065.2   3rd Qu.:17.25   3rd Qu.: 22.50
##              Max.   :3556.0   Max.   :23.00   Max.   : 39.40
## Humidity... Wind.speed..m.s. Visibility..10m. Dew.point.temperature
## Min.    : 0.00   Min.    :0.000   Min.    : 27   Min.    : -30.600
## 1st Qu.:42.00   1st Qu.:0.900   1st Qu.: 940   1st Qu.: -4.700
## Median :57.00   Median :1.500   Median :1698   Median :  5.100
## Mean   :58.23   Mean   :1.725   Mean   :1437   Mean   :  4.074
## 3rd Qu.:74.00   3rd Qu.:2.300   3rd Qu.:2000   3rd Qu.: 14.800
## Max.   :98.00   Max.   :7.400   Max.   :2000   Max.   : 27.200
## Solar.Radiation..MJ.m2. Rainfall.mm. Snowfall..cm. Seasons
## Min.    :0.0000   Min.    : 0.0000   Min.    :0.00000   Length:8760
```

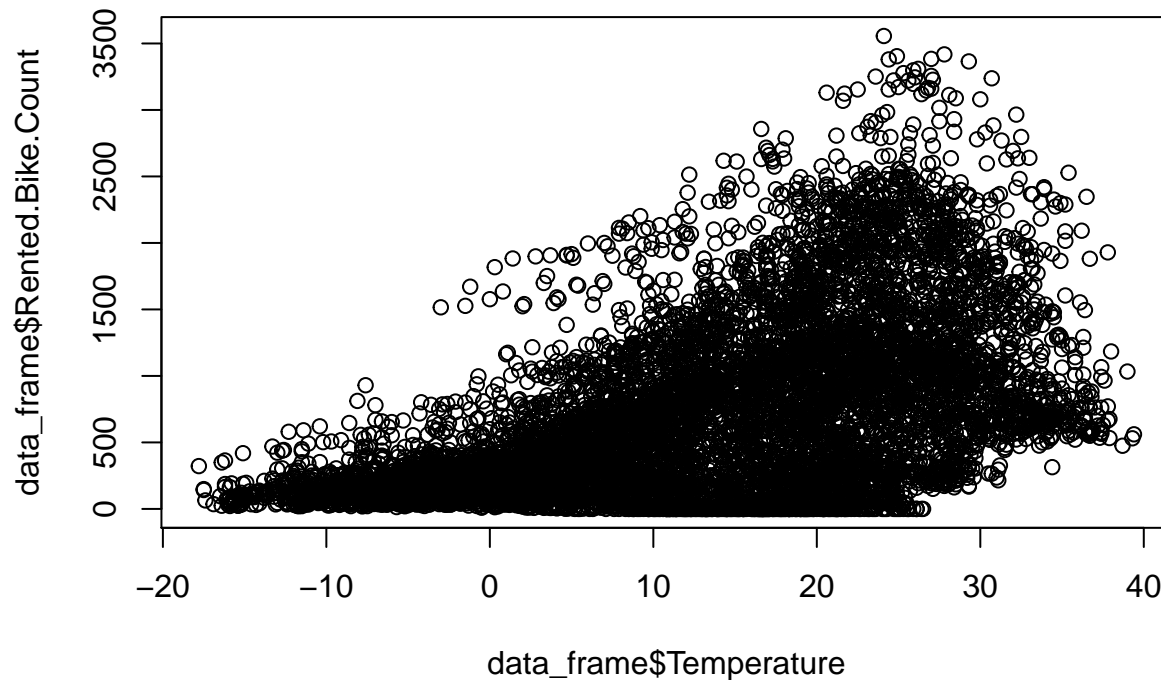
```
## 1st Qu.:0.0000      1st Qu.: 0.0000  1st Qu.:0.00000  Class :character
## Median :0.0100      Median : 0.0000  Median :0.00000  Mode  :character
## Mean   :0.5691      Mean   : 0.1487  Mean   :0.07507
## 3rd Qu.:0.9300      3rd Qu.: 0.0000  3rd Qu.:0.00000
## Max.   :3.5200      Max.   :35.0000  Max.   :8.80000
##      Holiday      Functioning.Day
## Length:8760      Length:8760
## Class :character  Class :character
## Mode  :character  Mode  :character
##
##
##
```

```
head(data_frame)
```

```
##      Date Rented.Bike.Count Hour Temperature Humidity... Wind.speed..m.s.
## 1 01/12/2017           254    0      -5.2      37           2.2
## 2 01/12/2017           204    1      -5.5      38           0.8
## 3 01/12/2017           173    2      -6.0      39           1.0
## 4 01/12/2017           107    3      -6.2      40           0.9
## 5 01/12/2017            78    4      -6.0      36           2.3
## 6 01/12/2017           100    5      -6.4      37           1.5
## Visibility..10m. Dew.point.temperature Solar.Radiation..MJ.m2. Rainfall.mm.
## 1           2000              -17.6              0           0
## 2           2000              -17.6              0           0
## 3           2000              -17.7              0           0
## 4           2000              -17.6              0           0
## 5           2000              -18.6              0           0
## 6           2000              -18.7              0           0
## Snowfall..cm. Seasons      Holiday Functioning.Day
## 1           0 Winter No Holiday      Yes
## 2           0 Winter No Holiday      Yes
## 3           0 Winter No Holiday      Yes
## 4           0 Winter No Holiday      Yes
## 5           0 Winter No Holiday      Yes
## 6           0 Winter No Holiday      Yes
```

Scatter plot between Temperature and Rented bike count

```
plot(x=data_frame$Temperature ,y=data_frame$Rented.Bike.Count)
```



Modeling by using linear regression (a basic form of machine learning)

```
# y(bike rent count) = a (parameter) * x (temperature) + b (parameter)
# by using regression, we can derive parameters a and b which represent the data best
# (minimizing the errors)

# formula argument format = Y ~ X
linear_regression_model = glm(formula = Rented.Bike.Count ~ Temperature, data=data_frame)
summary(linear_regression_model)

##
## Call:
## glm(formula = Rented.Bike.Count ~ Temperature, data = data_frame)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1100.60   -336.57    -49.69    233.81   2525.19
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  329.9525     8.5411   38.63  <2e-16 ***
## Temperature  29.0811     0.4862   59.82  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 295390.5)
##
##      Null deviance: 3643934363  on 8759  degrees of freedom
## Residual deviance: 2587029843  on 8758  degrees of freedom
## AIC: 135205
##
```

```
## Number of Fisher Scoring iterations: 2
```

Predict the bike rent count by using the derived a and b for any temperature

```
temp_temperature = 23
a = 29.0811
b = 329.9525

y = a * temp_temperature + b
y
```

```
## [1] 998.8178
```

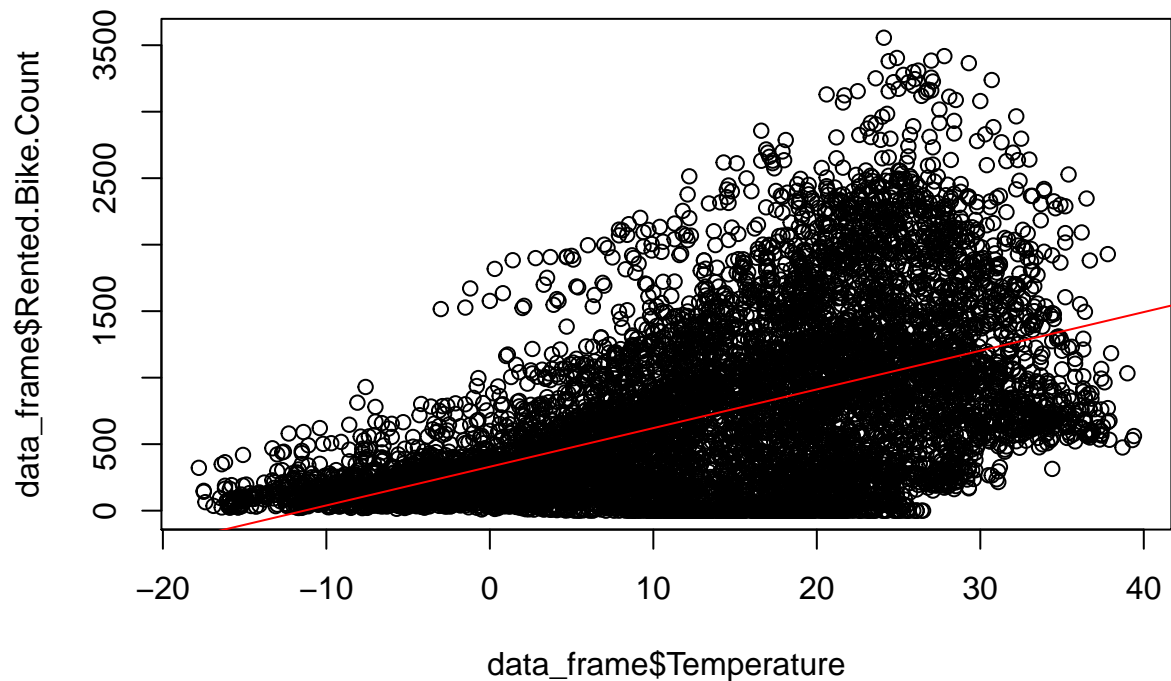
```
predict(linear_regression_model, newdata=data.frame(Temperature=c(23)))
```

```
##          1
```

```
## 998.8178
```

Plot the best fit line in red using the derived parameters

```
plot(x=data_frame$Temperature, y=data_frame$Rented.Bike.Count)
abline(a=b, b=a, col='red')
```



To

show several predicted bike count according to given temperatures

```
temperature_list = c(-10,0,10,20,30,40)
pred_list = predict(linear_regression_model, newdata=data.frame(Temperature=temperature_list))

data.frame(Temperature=temperature_list, Pred_bike_count=pred_list)
```

```
##   Temperature Pred_bike_count
```

```
## 1          -10          39.14152
```

## 2	0	329.95251
## 3	10	620.76350
## 4	20	911.57449
## 5	30	1202.38548
## 6	40	1493.19647