Objective:

In this session, you will learn to build your first simple linear regression model, to validate your model and interpret the results, and the evaluation metrics. You will also observe the diagnostic plots and validate linear regression assumptions.

Key takeaways:

- Data pre-processing
- Covariance and coefficient
- Building univariate linear predictive model using lm() on training data
- Validate the results using test data
- Error metrics

Problem Definition:

A large child education toy company (company name is confidential and data is masked) which sells edutainment tablets and gaming systems both online and in retail stores in the US wanted to analyze the customer data. They are operating from last 15 years and maintaining all transactional information data. The given data 'CustomerData.csv' is a sample of customer level data extracted and processed for the analysis from various set of transactional files. Using this data, they want us to understand the life time value of each customer (LTV). This will enable them to design marketing strategies and customize the product offerings. The objective of activity is building a regression model to predict the customer revenue based on other factors and understand the influence of other attributes on revenue.
Dependent (or Target) Attribute: TotalRevenueGenerated
Independent Attribute: NoOfUnitsPurcahsed

Steps to follow:

1. Read the data 'CustomerData.csv' into R.

2. Understand the structure of the data and pre-process
   a) Drop the attribute 'CustomerID'
   b) Check for missing values
   c) Convert 'City' as a factor variable

3. Let us study the correlations in the data. Note that correlation work only on the numeric attributes. Did you notice our data set has both numeric and categorical (factors in R language) variables? Here are the R code, to select only numeric attributes. Kindly note this is a reference code and you might need to make the necessary changes.
   *Please explore *corrplot* from online resources*

   ```
   install.packages("corrplot")
   library(corrplot)
   # looking at the names of each variables given the in R default data set
   names(mtcars)
   ```

```
# Taking few variables from the main data set and storing in a separate data set
sub <- subset(mtcars,select=c(mpg,disp,hp,wt)) # to include certain variables
sub <- subset(mtcars,select=-c(mpg,disp,hp,wt)) # to exclude certain variables
# correlation plot showing the numbers
corrplot(cor(sub), method = 'number')
# correlation plot showing as circles or ellipse. The default value is circle.
corrplot(cor(sub),method = 'ellipse')
```

4. Split the data into train and test data sets:

```
rows=seq(1,nrow(data),1)
set.seed(123)
trainRows=sample(rows,(70*nrow(data))/100)
train = data[trainRows,]
test = data[-trainRows,]
```

5. Build linear regression and interpret the results taking “NoOfUnitsPurchased”.

```
#Example code
LinReg<-lm(TotalRevenueGenerated~NoOfUnitsPurchased, data=train)
summary(LinReg)
plot(LinReg)
# study the results of LinReg models
```

6. Error metrics evaluation on train data and test data

```
library(DMwR)
#Error verification on train data
regr.eval(train$TotalRevenueGenerated, LinReg$fitted.values)
#Error verification on test data
Pred<-predict(LinReg,test)
regr.eval(test$TotalRevenueGenerated, Pred)
```