Simple Linear Regression

Project 1: Predicting the price of Toyota corolla based on the age.
Problem Statement
A large Toyota car dealership offers purchasers of new cars, the option to buy their used car as part of a trade-in. A new promotion promises to pay high prices for the used Toyota Corollas for purchasers of a new car. The dealers then sell the used car for a small profit.
To ensure a reasonable profit, the dealer needs to be able to predict the price that the dealership will get for the used cars. For that reason, data was collected on all previous sales of used Toyota Corollas at the dealership. The goal is to predict the price of a used Toyota Corolla based on its age. Age is given in months.

Data Set:

Approach: Create a simple linear regression model

Steps

1. Read the data 'Toyota_SimpleReg.csv' into R.
2. Understand the structure of the data and perform the required pre-processing steps
    a. Drop the drop the "Id", "model" attributes
    b. Check for missing values etc.
3. Check correlation between independent and dependent variable.
4. Split the data into train and test data sets.
5. Build the linear regression model and interpret the results
   lm(DV~IV, data = dataframe_name)
6. Review residual plots and analyze the model summary
7. Evaluate error metrics evaluation on train data and test data
   library(DMwR)
   regr.eval (train$DV,model$fitted.values)
   regr.eval (test$DV, predict(model, test)

Project 2: Toy Company wants to predict the revenue generated by each customer

Regression Problem
A large child education toy company which sells edutainment tablets and gaming systems both online and in retail stores wanted to analyze the customer data. They have been operating for the last few years and have maintained all transactional information data. The given data 'CustomerData.csv' is a sample

of customer level data extracted and processed for the analysis from various sets of transactional files.

The objectives of the project are to build a regression model to predict the customer revenue based on other factors and understand the influence of other attributes on revenue.

Data Set:
Customer data:

Steps:

1. Read the data 'CustomerData.csv' into R.
2. Understand the structure of the data and perform the required pre-processing steps
   a) Drop the attribute 'CustomerID'
   b) Convert 'City' into factor
3. Split the data into train and test data sets
4. Build the linear regression model and interpret the results
5. Review residual plots and analyze the model summary
6. Evaluate error metrics evaluation on train data and test data library(DMwR)
7. Spend the time on understanding the summary of the results and perform experiments with multiple combinations of attributes (by dropping the attributes which are not significant).
8. Standardize the data (except the target variable) and split into train and test.
9. Build the model again using the standardized dataset and compare the summaries with the un-standardized data.
10. Compute the error metrics on both Train and Test data

Problem Statement:
A large child education toy company which sells edutainment tablets and gaming systems both online and in retail stores wanted to analyze the customer data. They are operating from last few 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.
The objectives are

- Building a classification model to predict whether the given customer will churn or not churn based on other known factors

Steps: Logistic Regression Model:

1. Read the data sets 'CustomerData_Classification.csv' into R.

2. Understand the structure of the data and pre-process the data
a) Drop the attribute ‘CustomerID’
b) Convert ‘City’ as factor variable
3. Target attribute is: Churned
4. Convert the attributes to appropriate data type.
5. Split the data into train and test data sets
6. Build logistic regression and interpret the results
7. Generate the error metrics on train and test data
   a. Precision
   b. Recall
   c. Accuracy
8. Evaluation on train & test data
9. Identify the important features and build the logistic regression on these features
10. Study the ROC curve and identify the best threshold values.
11. Generate the evaluation of error metrics on train and test data based on the threshold
    a. Precision
    b. Recall
    c. Accuracy

Classification Problem

Project 3,4,5: Toy company wants to analyze the customer records.

Note it has 3 flavors of algorithms so can be divided into 3 different groups

Problem Statement:

A large child education toy company which sells its products online as well as in retail stores wants to improve its business. They would like to categorize the customers based of revenue so that they can make business decisions accordingly.

a. Classify which customers are Regular and Premium based on Revenues.
b. Classify using C5.0 & rpart
c. Predict the Revenues the customer is likely to contribute.

Data Set:

CustomerData.csv

Approach:

Apply Decision Tree Algorithm for classification

Steps:

Pre-processing the data:

1. Understand the problem statement
2. Load the “CustomerData.csv” data into R.
3. Understand the data and identify the pre-processing steps to be applied on the data.
4. Apply the below pre-processing steps.
   a. Remove the attribute “CustomerID”
   b. Check for the missing values and impute it using knnImputation
   c. Convert the categorical attributes in to factor using as.factor()
   d. Bin the numeric attributes if you think it is good for analysis.
5. Custom bin the target as follow
   a. Revenue with less than $150 as “Regular” customers, and
   b. Revenue with greater than $150 as “Premium” customers.
   c. Convert “Revenue” into factor
6. Split the data into train and test data (70:30 ratio).

Classifying using C50:

7. Build model C50 using “Revenue” as target attribute
   ```
   Library(C50)
   DT_C50 <- C5.0(Revenue~.,data=train)
   ```
8. Predict on the train and test data sets.
9. Predict "Revenue" for train and test datasets
   ```
   pred_Train = predict(DT_C50,newdata=train, type="class")
   pred_Test = predict(DT_C50, newdata=test, type="class")
   ```
10. Generate confusion matrix.
11. Calculate accuracy on the train and test data.

Classifying using rpart:

12. Build model rpart using “Revenue” as target attribute
    ```
    Library(rpart)
    DT_rpart_class<-rpart(Revenue~., data=train, method="class")
    ```
13. Predict on the train and test data sets.
    ```
    #b. Predict "Revenue" for train and test datasets
    pred_Train = predict(DT_rpart_class,newdata= train, type="class")
    pred_Test = predict(DT_rpart_class, newdata=test, type="class")
    ```
14. Generate confusion matrix.
15. Calculate accuracy on the train and test data.

Regression Problem using rpart:

1. Apply the pre-processing steps from 1-4 listed above.
2. Use the original revenue attribute present in the data.
3. Split the dataset into train and test (70:30 ratio)
4. Build model rpart using “Revenue” as target attribute

library(rpart)
DT_rpart <- rpart(Revenue~.,data= train, method="anova")

5. Plot the tree
   librar y(rpart.plot)
   rpart.plot(DT_rpart,type=3,extra=101,fallen.leaves = FALSE)
6. Predict the Revenue for train and test datasets
   pred_Train=predict(DT_rpart,newdata= train, type="vector")
   pred_Test=predict(DT_rpart, newdata= test, type="vector")
7. Check the evaluation metrics on train data
   regr.eval(train$Revenue,pred_Train)
8. Check the evaluation metrics on test data
   regr.eval(test$Revenue,pred_Test)

Clustering Activity:

Project 6: On the inbuild 'mtcars' data set, we will be clustering the similar cars based on different

Data Set: mtcars inbuilt data set

Approach:
features using K-means and Hierarchial clustering.

Steps:

1. Load inbuild 'mtcars' data available in R
2. Understand the data and apply the necessary pre-processing steps.
3. Normalize/Scale the data.

Note: Identify the cluster performance with and without normalizing/scaling the data and identify the importance of the scaling the data.

#Hierarchical Clustering Activity:

1. Calculate the distance between different cars using "dist" function using different distance methods.
   d <- dist(mydata, method = "euclidean") # distance matrix
   d
   Note: Experiment with different distance methods.
2. Build the hierarchical clustering using "hclust" function using agglomerative method
   ward.D2
   fit <- hclust(d, method="ward.D2")
   Note: You can explore different methods single, complete, average
3. Visualize the clusters. Tree like structure is called as dendrogram.

plot(fit)
# dendrogram displays all possible clusters from the data in bottom up approach

4. Creating 5 clusters using cutree function, “K” specifies number of cluster to create.
   groups <- cutree(fit, k=5) # cut tree into 5 clusters
   groups
   # draw dendogram with red borders around the 5 clusters
   rect.hclust(fit, k=5, border="red")
5. Append cluster labels to the actual data frame
   Mydata_cluster <- data.frame(mydata, groups)

Project 8:
K-means clustering:

1. Build the cluster using kmeans function by mentioning the number of clusters.
   # K-means clustering
   fit<-kmeans(mydata,centers=2)
   fit

2. Check sum of Inter cluster distance(betweenness) and Intra cluster distances(With-in sum of squares).
   fit$withinss
   sum(fit$withinss)
   #Cluster Centers
   fit$centers
   #To check cluster number of each row in data
   fit$cluster

3. Identifying the ideal number of cluster:
   - Write a for loop which should start with 2 clusters and build k-means model up to 15 clusters.
   - Capture the within-sum of squares for different number of cluster, save
     sum(fit$withinss) for each model.
   - Plot sum(fit$withinss) generated in all models
   - Find the best cluster based on the curve.

Project 9: Clustering
Data set: ‘Cereals.csv’

Cereals data: Identify similar cereals using K-means clustering

Cereals data: Data consists of the information of proteins, calories, vitamins, carbohydrates, minerals etc. for different cereals. Using K-means technique identify/cluster the similar cereals.

- Load the cereals data into R.
- Analyze the data and apply the required pre-processing steps and prepare data for clustering.
- Use a distance metric to compute distance matrix.
- Apply k-means clustering technique, identify the ideal number of cluster.
- Identify the similar cereals based on the clusters.