Activity Sheet

Learning outcomes:

After completing this exercise, you should be able to understand and perform below tasks.

1. Building Regression models
   Regression model using logistic regression technique
2. Validating the model results and optimizing the model
3. Handling multicollinearity and dimensionality reduction
4. Evaluation of error metrics
5. Applying the models on un-seen data
   a. Splitting data into train and test data sets
   b. Comparing the error metrics
6. Interpretation of the results

Logistic Regression- Understanding the output

1. Read the German Credit Data into R
2. Separate the numeric and categorical attributes into two data frames
3. Check if the data types are appropriate and convert if necessary
4. Discretize the numeric attributes with 5 equal frequency bins.
5. Merge the two data sets. Eliminate the missing records
6. Make a train and test split in 80:20 ratio
7. Build a logistic regression model and interpret the results
   #glm(Target~variable,data,family="binomial")

Logistic Regression- with multiple attributes

8. Build logistic regression model using all the attributes in the data
9. Evaluation on train and test data. Observe that the output of the logistic regression is probabilities. Set a threshold value and classify the probabilities
   prob<-predict(model,type="response")
   data$pred<-as.factor(ifelse(prob>0.5,1,0))
10. Compute the confusion matrix and identify the appropriate metric for this problem
11. Check for multicollinearity and use stepAIC to obtain attributes for model building
12. Update the model and check if the metric improved
13. Setting threshold using ROC curve

```
library(ROCR)
library(ggplot2)
predicted <- predict(Model,type="response")
prob <- prediction(predicted, data$RESPONSE)
tprfpr <- performance(prob, "tpr", "fpr")
```

```
#A<-performance(prob,"auc")
plot(tprfpr)
str(tprfpr)
cutoffs <- data.frame(cut=tprfpr@alpha.values[[1]], fpr=tprfpr@x.values[[1]],
                      tpr=tprfpr@y.values[[1]])
tpr <- unlist(slot(tprfpr, "y.values"))
fpr <- unlist(slot(tprfpr, "x.values"))
auc <- performance(prob,"auc")
auc <- unlist(slot(auc, "y.values"))
roc <- data.frame(tpr, fpr)
ggplot(roc) + geom_line(aes(x = fpr, y = tpr)) +
 geom_abline(intercept=0,slope=1,colour="gray") +
 ylab("Sensitivity") +    xlab("1 - Specificity")
```