**INFO 7390 SVM Assignment Bharat Parwani**

**Code Explanation**

**# Here we are loading all the required packages**

library(caret)

library(e1071)

data(GermanCredit)

dataset = GermanCredit

**# In the below code the str() function gives the compact structure of the data with its variables**

**# The second line of code is selecting all the rows and the first seven columns from the data as a list using the function lapply(). Also the line of code is checking if the loaded dataset is in dataframe format or not, if not it will convert it into a dataframe.**

**# Thus, again checking the structure of the dataset in dataframe**

str(dataset)

dataset[,1:7] = as.data.frame(lapply(dataset[,1:7], scale))

str(dataset)

**# The below line of code sample() function is used, which divides the data into parts, thus here the code is taking 200 values out of the total 1000 values.**

**# Storing those 200 values in test\_dateset variable and the remaining values are stored in train\_dateset**

sample\_index = sample(1000, 200)

test\_dateset = dataset[sample\_index,]

train\_dateset = dataset[-sample\_index,]

**# Here we created 2 models using kernels and used the function tune() to tune the model over various values of cost and gamma**

model\_lin <- tune(svm,Class~.,kernel="linear",data=train\_dateset,ranges=list(gamma=2^(-10:5),cost=2^(2:10)))

summary(model\_lin)

model\_rad <- tune(svm,Class~.,kernel="radial",data=train\_dateset,ranges=list(gamma=2^(-10:5),cost=2^(2:10)))

summary(model\_rad)

**# In this line of code we used the predict function which is used to display predictions from the results of various fitting of the model functions**

predictions <- predict(model, test\_dateset[-10])

**# In this line of code, table() function is used which displays the results in a tabular form. We can also see the sorted results**

table(test\_dateset[,10], predictions)