# CS60050 MACHINE LEARNING

## Assignment 1

Group Members:

Rahul Mandal (20CS30039)
Sailada Vishnu Vardhan (20CS10051)

# Q2. Files:

- ## main.py

This file contains the code for naïve bayes classifier. It trains classifier using 10-fold cross validation along with Laplace correction.

Functions defined inside it are:

1. splitting_data(): this function divides the dataset into 80:20 training, testing datasets.
2. outliers(): basically detects the outliers and remove them if found.
3. If_null_value(): this function detects if the any dataset row has missing value and remove that row.
4. k_fold_train(): this function trains the splitted training dataset using k fold validation method. Here the value of k is 10.
5. normal(): returns the likelihood values
6. predict(): this function has naïve bayes implementation and predicts the output for the input data.
7. accuracy(): this function calculates the accuracy of the prediction by predict() function.

- ## Dataset_C.csv

This file contains the data to be used to train and test the classifier.
It has following features:

- Id
- Gender
- Age
- Driving_License
- Region_Code
- Previously_Insured
- Vehicle_Age
- Vehicle_Damage
- Annual_Premium
- Policy_Sales_Channel
- Vintage
- Response

## Procedure for naïve bayes classifier:

First the categorical data is encoded using LabelEncoder() for performing calculations as we cannot perform calculations on string and integer variables together.

Then program checks for the outliers and if found it removes it.

Now, the program splits the whole data set in 80:20 to training and test data. After that program trains the naïve bayes using 10-k fold validation method.

Program then extracts the training data which has the highest accuracy in 10-fold training.

It then tests the 20% test data on this highest accuracy train data and predict the outcome and calculate the final accuracy.

## Accuracy table for 10 fold validation on 80% train data:

| Accuracy for 1st run: | Accuracy with laplace |
|---|---|
| 63.40769 | 64.29046 |
| 64.00228 | 64.1162 |
| 64.49592 | 63.99279 |
| 63.6985 | 63.31878 |
| 63.73647 | 63.35675 |
| 64.43896 | 64.2491 |
| 63.85988 | 63.1574 |
| 64.1067 | 63.92633 |
| 63.10993 | 63.58458 |
| 64.11011 | 63.8728 |

Observation: There was no major difference in after applying laplace smoothing in the accuracies of 80% train data while 10 fold validation.