# **CAPSTONE PROJECT**

# Sayanha Mondal

Electronics-part-2

22095102

First I looked at the dataset carefully to understand what kind of preprocessing techniques to use.

* There were rows containing ‘0’ as age in the column so I dropped those rows to eliminate the missing value.
* I started with binning to group continuous numerical data into discrete intervals or bins.I chose the column ‘Age’ , and then defined bins to reduce the wide range of values.Equal width Binning.Replace the continuous values in the feature column with the bin labels.

Binning can smooth out noisy data and make the patterns more discernible of the model.

Then dropped the actual ‘Age’ column.

# Label encoding: Encoded all the categorical data To Numerical format for algorithms to work. Label encoded all the columns with categorical data.

* StandardScaler is used to scale numerical features to have a mean of 0 and a standard deviation of 1, ensuring that the features are on the same scale and preventing certain features from dominating others during model training.
* train\_test\_split is used to split the dataset into separate training and testing sets, allowing the model to be trained on one portion and evaluated on another to assess its generalization performance , keeping the test\_size =0.2 for maximum accuracy.
* Data distribution analysis : Plotted the fraud and non-fraud ratio.It shows that the dataset is very much imbalanced.
* SMOTE Technique: To balance the imbalanced dataset we used this technique. It generates synthetic samples of the minority class to address class imbalance, enhancing the model's ability to detect rare events like fraud or anomalies.

While applying boosting I noticed a very low f1 score due to an imbalanced dataset , the ratio was almost 1:16 .

* Then applied Gradient Boosting and we can see that the f1 score is way too low.
* So to prevent overfitting I chose k-fold cross validation to assess the models’ generalization performance.It divides the dataset into n subsets and the model is trained 5 times. Each time uses a different subset as the validation set and the rest as the training set. The final performance metric is the average of the k evaluation results, providing a more robust estimation. That’s why I feel it increased the f1 score really high making it a better model.
* After applying random forest and XGBoost it shows XGBoost is little better.I think as it uses a gradient boosting approach, which enables it to build more complex models by iteratively improving weak learners, leading to better accuracy and generalization.That’s why XGBoost performed little well giving a f1 score of 0.97518 .

I feel as the data was really imbalanced , balancing the data was the real task as imbalanced data was giving high accuracy but the model f1 score was really low. Also overfitting the algorithm also gave a low score .