# METHODOLOGY

The Image below shows the project workflow diagram at macro level-:

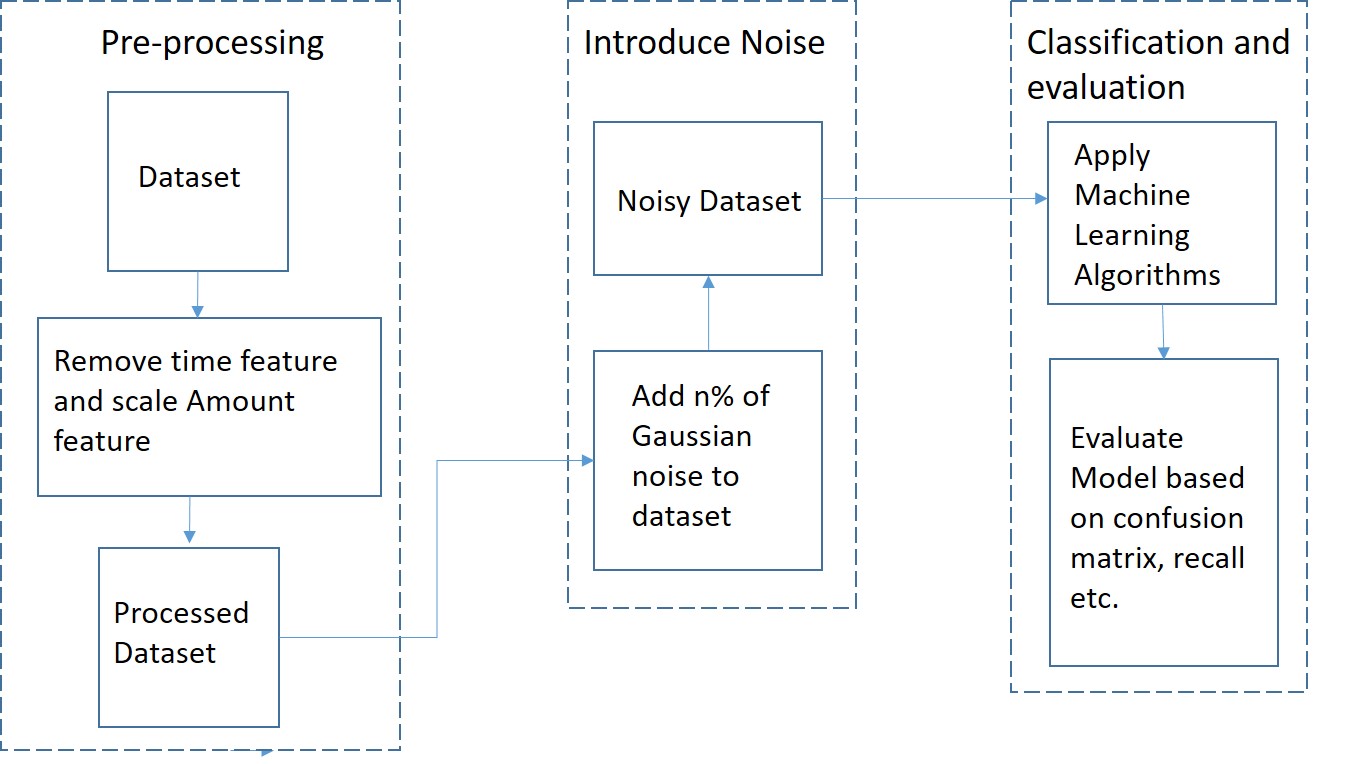


Figure 1: Project Workflow Diagram

1. **Data & Pre-Processing**

For our project, we used credit card fraud dataset from kaggle []. In the dataset there are 284807 records of transactions (made in September 2013 over two days) and 28 normalised numerical features which were obtained by using PCA on a set of original features, due to confidentiality issues the names of features are not revealed by author, other than these 28 features, two additional features ‘Time’ and ‘Amount’ are also provided for additional information. We drop the Time feature as it follows an increasing trend (from 0 to 172792) and does not give any information for labels in the format it is given, we normalise and scale the Amount feature as it can give important information like the amount of fraud and the value which can be saved by employing the fraud detection models. In the end, we get 29 scaled and normalised numerical features and one label column ‘Class’ containing 0's and 1's (1 denoting fraud transaction).

1. **Introduce Noise to dataset**

After pre-processing the dataset, we add noise to it and evaluate performance of three popular classification algorithms (Logistic Regression, Naïve Bayes Classifier and Random Bayes Classifier) at different noise levels.

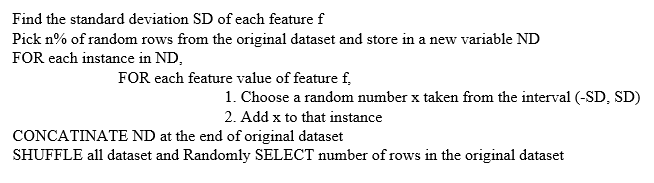
For adding n% of noise to a given dataset, we use the following algorithm []-:

Figure 2: Algorithm for adding noise to a dataset

Below is an example of how we have added noise to the dataset

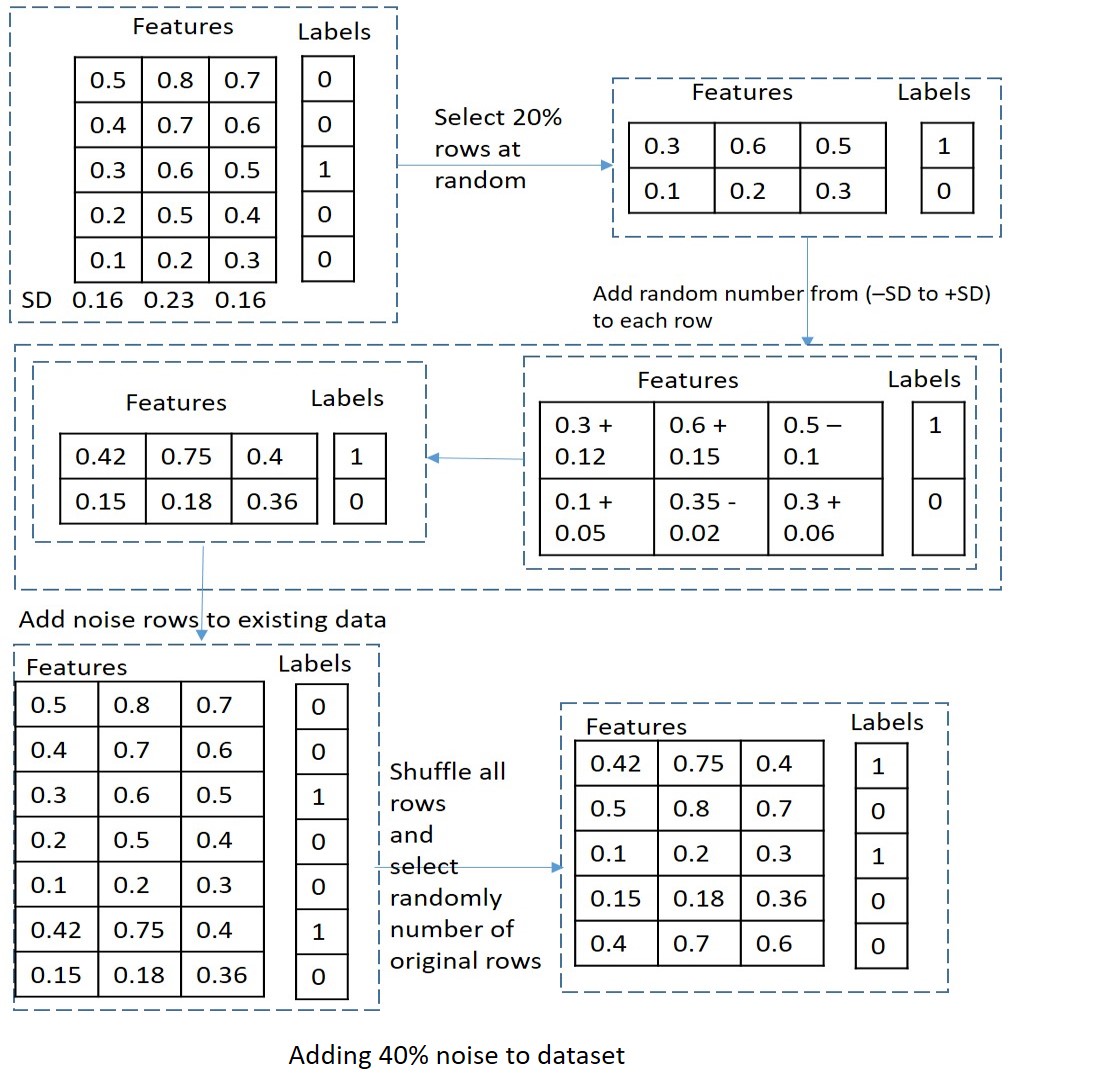


Figure 3: Example of addition of 40% noise to a dataset

1. **Classification and Evaluation**

After adding noise to the dataset the dataset was split into train set and test set with a 70:30 split. We use Logistic Regression, Naïve Bayes Classifier and Random Forest Classifier with default hyper parameters for classification of frauds as we are monitoring the effect of noise on different ML algorithms. We will discussed hyper-parameter tuning more in future work section.

For evaluation of our models we have used a probability threshold of 0.5 for the classification of a transaction as a fraud. If the probability of a transaction being fraud is given to be more than 50% by an algorithm then the transaction is classified as a fraud else regular. Also, we are monitoring the effect of noise on different ML algorithms so as far as we keep the probability threshold same for every algorithm it should not affect the impact of noise on the performance of an algorithm.

Since our dataset is highly unbalanced, the negative class (frauds) account for only 0.172% of all transactions evaluation metrics like accuracy and AUC under ROC should not be considered for evaluating the models. Since we are interested in correctly predicting the negative class (frauds) correctly, we have used confusion matrix and recall for negative class (TN/TN+ FN) as a metric to evaluate our models performance. We are also showing specificity (TN/TN+FP) at different noise levels, why we do that is discussed in more detail in results section.