

# MSBD 6000B Project I Report

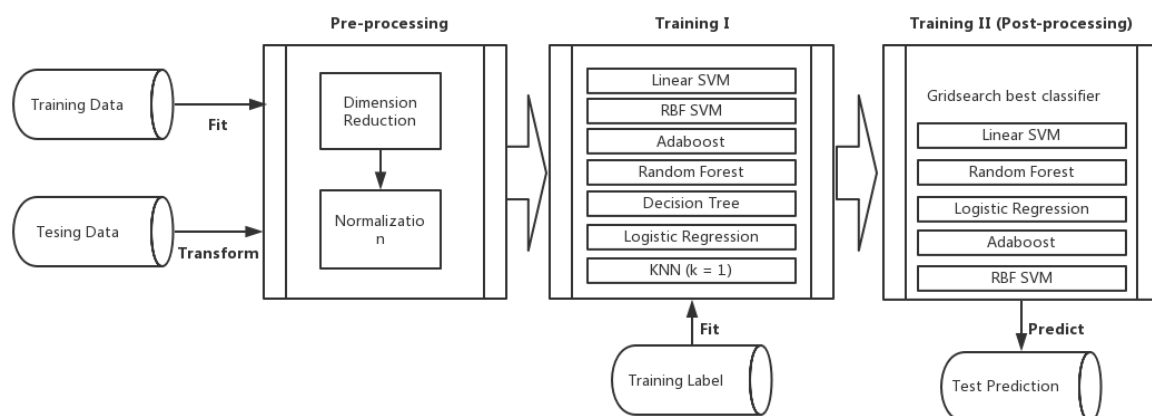
Student Name: LIU Yan Yun      Student Id: 20384933

## I. Introduction

In this project, I use a group of traditional machine learning classifiers which are well-packaged in sklearn to solve the given classification problem. The main workflow of my project:

1. Load the training data and training label
2. Dimension reduction.
3. Data normalization and standardisation.
4. Train machine learning model with seven classifiers.
5. Given seven output prediction as input features, select the best classifier from grisearch to give last-layer prediction.
6. Re-train the model with all the data and give prediction for test data.

The workflow is shown below:



## II. Pre-processing and dimension reduction

### Before we start - check correlation among features

To have a better understanding about the dataset, I check the correlation coefficient for features with `np.corrcoef`. The result indicate that some features are highly linear dependent, so dimension reduction is nessessary.

### Dimension Reduction

To aviod collinearity, I use several feature selection tools to check which features need to be preserved. In function `Select_best_features`, I set five dimation reduction methods:

```
VarianceThreshold(threshold=(.8 * (1 - .8))),
PCA(32),
PCA(48),
SelectFromModel(LinearSVC(C=1, penalty="l1", dual=False)),
SelectFromModel(ExtraTreesClassifier())
```

using grid search to select one with the best score.

## Data standardization and normalization

Pre-processing stage is essential in machine learning methodology. A standard, normalized dataset will provide some benefits for training. In function `Normalize_train_test_features`, I use `StandardScaler` in sklearn to normalize training data and testing data.

## III. Training model

In training stage, my idea is to train seven classifiers and output seven set of predictions. Then use the seven output as input features, again train another model to get final prediction. The workflow is shown below:



### Stage I: Using seven classifier to get predictions

To get more robust prediction, I use seven different classifiers and train these models one by one, predicting seven results for each test sample as new input features, and give prediction for each training sample as new training features as well.

### Stage II: Give final prediction using seven features

In this part, I used the output predictions as 7 features to predict final result. The model is selected by grid-search in function `Get_final_prediction_gridsearch`.