HOMEWORK-1 REPORT
Big Data for Health Informatics

Table 2: Descriptive statistics for alive and dead patients

Metric	Deceased patients	Alive patients
Event Count		
1. Average Event Count	982.014	498.118
2. Max Event Count	8635	12627
3. Min Event Count	1	1
Encounter Count		
1. Average Encounter Count	23.038	15.452
2. Max Encounter Count	203	391
3. Min Encounter Count	1	1
Record Length		
1. Average Record Length	127.532	159.2
2. Max Record Length	1972	2914
3. Min Record Length	0	0

Table 3: model_partb.py results (Model Performances on Training Data)

Model	**Accuracy**	**AUC**	**Precision**	**Recall**	**F-score**
Logistic Regression	0.9545	0.9454	0.9869	0.8988	0.9408
SVM	0.9952	0.9955	0.9911	0.9970	0.9941
Decision Tree	0.7763	0.7476	0.7922	0.6012	0.6836

Table 4: Model performance on test data

Model	**Accuracy**	**AUC**	**Precision**	**Recall**	**F-score**
Logistic Regression	0.7381	0.7375	0.6804	0.7333	0.7059
SVM	0.7381	0.7389	0.6768	0.7444	0.7090
Decision Tree	0.6714	0.6569	0.6329	0.5556	0.6836

We can see significant change in the auc/acc scores from train to test data as expected. Since decision tree with max_depth of 5 is a weak learner for a dataset with 3190 features. We can notice a low score compared to SVM or Logistic. Prediction can be further improved with feature engineering, and using finely tuned ensemble methods. Please review the improvements mentioned in the last section.

Table 5: Cross Validation

CV Strategy	Accuracy	AUC
K-Fold	0.7285	0.7115
Randomized	0.7357	0.7143

My_Model Peformance:

my_model	Accuracy	AUC
K-Fold	0.7453	0.7338
Randomized	0.7095	0.6910

Model Description:

Step1: Generate X_test

- I have used ETL functions from etl.py for generating test features as well.

Step2: Feature Selection

- Using LinearSVC and SelectFromModel methods

Step3: Identifying an Ensemble Learner

- I have implemented Random Forest Classifier and Adaboost Classifier

Step4: Grid Search for optimal parameters.

Feature Selection:
Since we are dealing with no. of samples much lesser than features and data is very sparse, I have decided to select features with non-zero coefficients using the SelectFromModel function in sklearn.

I have considered LogisticRegression and LinearSVC as both are great algorithm for feature selection in classification problem with sparse data. Based on cross validation LinearSVC performed well for this dataset.

Predictive Model:
Since Ensemble Learners are good learners for high dimension data I implemented Adaboost and RandomForest classifiers. Random Forest performed slightly better than Adaboost. RandomForest avoids overfitting inherently so I chose this over the other.

Tuning Parameters:
Once after selecting the model tuning is an important part of machine learning process.
Tuned Parameters:
C (from LinearSVC) : 1.5

Threshold(SelectFromModel) : 0.15
No. estimators (RandomForest): 50

RandomForest with 500 estimators has significant improvement compared to 50 estimators, as suspected it is a case of overfitting as the performance didn't reflect on the test data.

Results:

My model gave an AUC 0.74% compared base models when compared against auc scores from cross-validation. Slightly better than the Adaboost classifier, although Adaboost classifier can be tuned in terms of base estimator and learning rate.

Improvement:

This model can be further improved by feature engineering.

- Incorporating domain knowledge
  - Ex: event_id's related to major illness
  - Using un-supervised learning methods such as PCA, to extract statistically significant features.