1. Divide your training data in **train and validation set** and then test with the testing set provided in assignment 1.

ANSWER:

Given the dataset in assignment 1, I successfully split the data first into training and testing set (X_train, X_test, y_train, y_test) using train_test_split(), after obtaining the training set I have again divided the data into training and validation set (X_train_set, X_validation_set, y_train_set, y_validation_set). Then, performed feature scaling using standardscaler to make independent variables on same scale. Used Random Forest and Xgboost algorithm to train the data and then used validation set to evaluate the model.

Used n_estimators=300 in Random forest, default was 100, it increased accuracy from 0.9835519004223161 to 0.9842742831740386, again used different parameters like max_depth, max_features, min_sample_leaf, but did not see significant improvement in the performance of the model. Hence, the default values of the model works really well, applying feature scaling only increases load on the model for the given dataset.

Used learning rate=1 in Xgboost, default was 0.3, it significantly increased the performance of the model. Accuracy moved up from 0.9477106023560792 to 0.972716159146477 and confusion matrix also got improved as below:

Before-

```
Confusion Matrix: [[11760   353]
 [  588  5295]]
```

After-

```
Confusion Matrix: [[11858   255]
 [  236  5647]]
```

After using validation set, used test set to test the model.

In Random Forest,

Validation set:

```
Accuracy: 0.9841631473660813
Confusion Matrix: [[12041    72]
 [  213  5670]]
AUROC: 0.9789249777917808
```

Test set:

```
Accuracy: 0.9844409868859747
Confusion Matrix: [[15105    85]
 [  265  7040]]
AUROC: 0.9790638451843611
```

Hence, high accuracy and AUROC values tells that it has performed well on both the validation and test set.

In XGBoost algorithm,

Validation set:

```
Accuracy: 0.972716159146477
Confusion Matrix: [[11858   255]
 [  236  5647]]
AUROC: 0.9694163250727305
```

Test set:

```
Accuracy: 0.9707490553456324
Confusion Matrix: [[14870   320]
 [  338  6967]]
AUROC: 0.9663319152924467
```

Hence, high accuracy and AUROC values tells that it has performed well on both the validation and test set. But the false positives and false negatives in the confusion matrix indicates some improvement is needed.

2. Compare the two classifiers (which ever you used in your assignment 1) **using a t-test** and report which one is significantly better than the other.

ANSWER:

To perform the t-test, performance metrics were calculated as below:

Random Forest Metrics-

- Accuracy: 0.9842
- Precision: 0.9875
- Recall: 0.9638
- F1 Score: 0.9755
- AUROC: 0.9789

XGBoost Metrics-

- Accuracy: 0.9727
- Precision: 0.9568
- Recall: 0.9599
- F1 Score: 0.9583
- AUROC: 0.9694

The t-test compares these metrics and identifies if two algorithms are different or not.

After performing t-test, we observed there is considerable difference between the two classifiers.
After examining the above metrics, we can tell that Random Forest algorithm outperforms XGBoost algorithm.

3. Use a k-fold (k=10) cross validation and report the performance of the testing set. Explain which model you used finally for generating the predictions of each sample test data.

ANSWER:

As the given dataset is imbalanced, I used stratified k-fold cross validation.
Calculated mean of Accuracy, Confusion matrix and AUROC whose values were as follows:

Random Forest Cross Validation Accuracy: Mean = 0.9818443864151666
XGBoost Cross Validation Accuracy: Mean = 0.9737712346810738
Random Forest Confusion Matrix: Mean = 2811.8
XGBoost Confusion Matrix: Mean = 2811.8
Random Forest AUROC: Mean = 0.9769965090741207
XGBoost AUROC: Mean = 0.969201396186748

Looking at the above values, we can say that, Random Forest has higher mean accuracy and mean AUROC than XGBoost, while both perform similar on confusion matrix.

Both Random Forest and XGBoost obtained high mean accuracy and mean AUROC when using k-fold cross-validation, however, Random Forest's mean accuracy and AUROC were somewhat higher.

XGBoost had an accuracy of 0.9707 and an AUROC of 0.9663 in the normal model fit on the testing set, but Random Forest had a better accuracy (0.9844) and AUROC (0.9791).

The normal model fit performance on the testing set shows that Random Forest outperforms XGBoost in terms of accuracy and AUROC. The k-fold cross-validation data, on the other hand, show an average or mean performance over numerous folds, and the difference between the two models are not major, but Random Forest still wins.

Based on the above observations, Random Forest algorithm was chosen for generating predictions as it consistently performed better than XGBoost algorithm.