The zip file contains below python files:

1. **Config.py** : set the Hyper parameters and directories in this file. Hyper parameters include the paper parameters as well as my own added parameters.
2. **Decision\_tree.py** : Contain the DecisionTree class in this file. I used the pre-order traversal tree search to create the tree. I used the maximum depth and loss function as the stopping criteria for tree. The loss function can be based on Gini factor or entropy.
3. **RandomForest.py** : Contain the RandomForest class.
4. **BiasForest.py** : Contains the Biased random forest class which I named it BRAF.
5. **Utils.py**: Contain the methods for reading the data, calculating the binary metrics, running the K fold cross validation, saving the figures and trained models and the results of the k-fold.
6. **Main.py**: Runs the main API.

Also below folders contain the results:

1. **Curves** : contain the ROC and PRC curves of k-folds. And Also in each curve I summarized accuracy, recall and other metrics.
2. **Results** : Contains the csv results of k fold.
3. **Saved\_models**: Contains the trained models in k-fold cross validation
4. **Tree\_graphs** : Contain the graphs of the trees in the biased forests. These trees have the information of each nodes and edges on them.

**My suggestions:**

1. **Using Entropy instead of Gini metric increase the overall accuracy and precision.**
2. **Using the weighted classes increase the precision, decrease the false positive rate and increase AUC ROC.**

**BS**: I used weighted Gini and Entropy rather than simply equally considering positive and negative labels and then I added methods name+w1+w2 for example for the entropy loss function with weights= [.5,1.5] the folders name would be “**Curves\_entropy\_.5\_1.5**”