Positive Class = whether loan will not be returned

Pos Class = NOT FULLY PAID

Neg. Class = FULLY PAID

# CSE 494: Artificial Intelligence for Cyber Security: Spring 2019

## Lab 3: Supervised Learning demo

### Objectives of the lab:

- Use the Decision Tree classifier for a binary classification problem using scikit-learn DecisionTreeClassifier method
- Extend it to Random Forests classifier and understand how an ensemble of weak classifiers works.
- Use a parametric model namely logistic regression on a binary classification problem.

The first part of the demo would be a demo on the Decision Tree classifier (20 min.) - to refresh your knowledge on Decision Trees, you may take a brief look into http://www.cs.cmu.edu/~aarti/Class/10601/slides/DTreesAndOverfitting-9-13-2011.pdf (taken from CMU ML course)

For this demo, we would be using the data from Lending Club publicly available dataset. The file is available for download from the blackboard – loan_data.csv

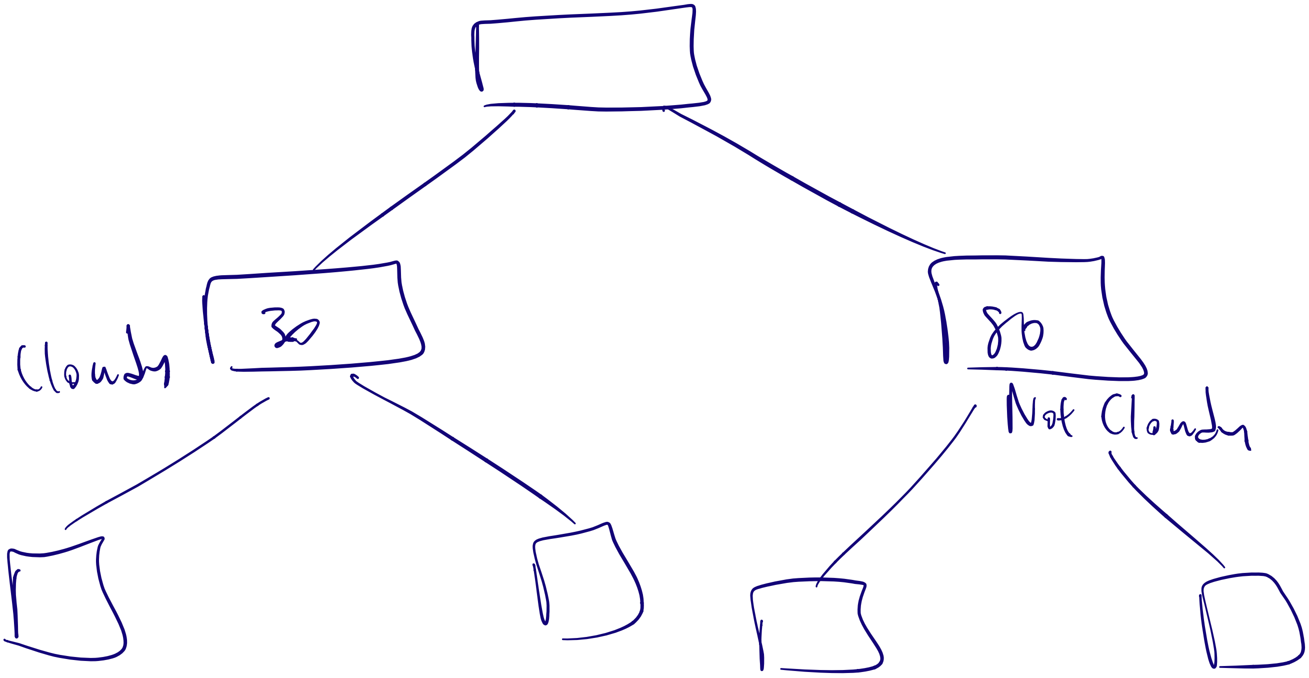
1. Decision Tree Classifier
    a. Exploratory data analysis and preprocessing for input to the DecisionTreeClassifer method in scikit-learn, handle missing values (scikit-learn does not implicitly)
    b. Understanding train test split and cross validation for evaluating classifiers in supervised learning scenarios. Also balancing the dataset in terms of class balance.
    *c.* Understanding the effect of different hyper-parameters: *criterion, max_depth, min_samples_leaf, random_state*
    *d.* Evaluate the case of overfitting when the split is over-achieved.
    *e.* Evaluate the decision_path of the tree for a given sample. Tune Decision Tree classifier like pruning the tree (scikit learn does not implicitly facilitate this)
    f. Evaluate the performance of the classifier with different hyper parameter settings by plotting the precision, recall, F1 over a 5-fold cross validation.

The second part of the demo would be on understanding Random Forests (20 min.) - to refresh your knowledge on Random Forests, you may take a brief look into http://www.cs.cmu.edu/~aarti/Class/10601/slides/DTreesAndOverfitting-9-13-2011.pdf (taken from CMU ML course)

2. Random Forests classifier
   - *a.* Understanding the effect of different hyper-parameters: *criterion, max_depth, min_samples_leaf, random_state, number_estimators.*
   - b. Running a loop with increasing number of trees in the random forest and checking accuracy with confusion matrix
   - c. Evaluate the performance of the classifier with different hyper parameter settings by plotting the precision, recall, F1 over a 5-fold cross validation.
   - d. Training Random Forests without a validation set? How many trees are enough? For Random Forests you don't need to keep aside a validation set to get an unbiased estimate of the test set error (this is only for Random Forests, which using bagging, not for Decision Trees). Instead, one can use the out of bag error estimate (see http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm).

The last part of the demo would be on understanding Logistic Regression (20 min.) - to refresh your knowledge on Random Forests, you may take a brief look into https://www.cs.cmu.edu/~mgormley/courses/10701-f16/slides/lecture5.pdf (taken from CMU ML course)

3. Logistic Regression
   - a. Understanding the sigmoid function
   - b. Understand the effect of hyper-parameters: penalty, C, max_iter and the returned parameters of the model
   - c. Evaluate the performance of the classifier with different hyper parameter settings by plotting the precision, recall, F1 over a 5-fold cross validation.



DT is good for binary class

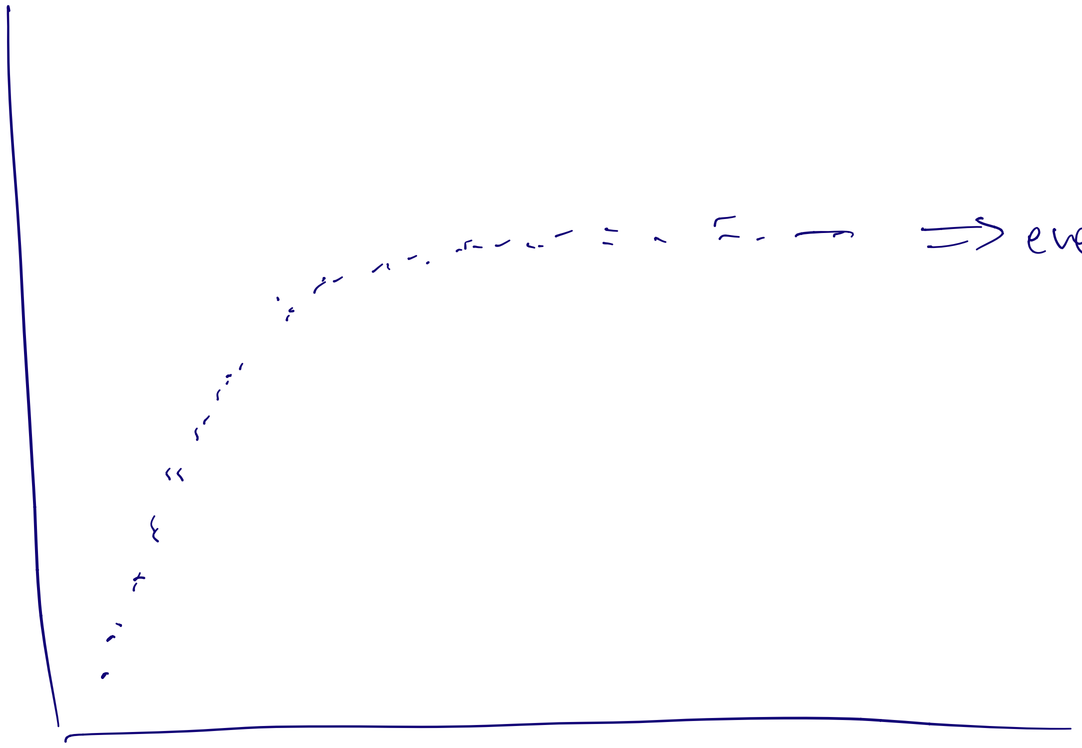
k-fold = 5 subsets

train on 4 subsets

test on i-th subset that is held back

| 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|

$$\text{Precision} = \frac{tp}{tp + fp}$$



⟹ eventually levels off