1. Team number (as registered in Brightspace).

11

2. Students’ name and Purdue e-mail (as registered in Brightspace). The project is to be done in groups of 3 students.

Joshua Hu @hu503

Nicholas Dullam @ndullam

Vishal Kolla @vkolla

3. [Up to 3 lines] Definition of the problem, possibly relevant to your interests.

A loan default is when a borrower fails to pay back their debt according to their initial arrangement with the debtor. This often bad for both parties involved. Using data of financial transactions for a given individual, can we predict when a loan default will occur, allowing us to predict and prevent them?

4. [Up to 5 lines] Which dataset will be used? The dataset should be already publicly available. For possible datasets, see the course website https://www.cs.purdue.edu/homes/jhonorio/21spring-cs37300.html Describe how many samples/features has the dataset. Give me some idea of how you plan to transform the dataset into a table of data that a machine learning algorithm can use. If the dataset is too large, then choose fewer samples (for instance 1000 or smaller). If the dataset has too many features, then arbitrarily choose few of those features (for instance 100 or smaller). Remember that cross validation takes time, and the larger the dataset, the more time this will take.

Loan Default Prediction Dataset. We will prune the dataset to a smaller sample of 1000 and choose a subset of the first 50 features. For example, we will remove the loss quantifying feature as in the scope of our problem we are only interested in the binary result of whether a default occurred or not, and not the magnitude of the default.

5. URL where the above dataset is available.

https://www.kaggle.com/c/loan-default-prediction

6. [Up to 5 lines] Which TWO machine learning algorithms are going to be used? (e.g., SVM, classification trees, etc.) You are allowed to either implement this from scratch or use third-party code, e.g., scikit-learn.

Given our feature-set, we are going to use Principal Component Analysis and K-nearest neighbors.

7. [Up to 5 lines] Which cross-validation technique(s) is(are) going to be used? (e.g., training/validation/testing, k-fold cross-validation, bootstrapping). You MUST implement this from scratch. You should specify how you will apply the cross-validation technique(s). For instance, for training/validation/testing, specify which percentage of the samples will be used for training, which percentage for validation, and which percentage for testing (e.g., 40%, 30%, 30%). For k-fold cross validation, specify the value of k (e.g., k=10). For bootstrapping, specify the number of bootstraps B (e.g., B=30).

We will use training/validation/testing, using the included test data in our dataset. As we have already done with our training dataset, we will be pruning the test dataset to 1000 samples (due to its large initial size). For the respective validation and testing sets, we will be using a 50/50 approach, with 50% of the testing set being used for validation, and 50% being used for testing (leading to an overall 50/25/25 considering the inclusion of the training set).

Since we are already pruning our training data, we won’t need the entire test data (which is very large), and will similarly prune it, and then partition it into a validation and testing set.

8. [Up to 10 lines] Which hyperparameter(s) is(are) going to be tuned. You MUST implement this from scratch.

Since we are using K-nearest neighbors, we will need to tune the hyperparameter k (number of k-nearest neighbors to consider), alongside the number of dimensions to consider in PCA by considering the coverage of the variance.

9. [Up to 10 lines] Which TWO experimental results will you show? (e.g., plots of number of samples versus accuracy using different subsets of the dataset, hyperparameter versus accuracy, ROC curves, etc.) You MUST implement this from scratch. Plots of the data itself DO NOT count as experimental results.

We will measure and plot the accuracy of our models as a function of our hyperparameters and sample size, demonstrating the effects of the hyperparameter tuning and size of sample on the accuracy of our models.

10. Which programming language are you going to use? (Only MATLAB, C++, Java and Python are allowed. Jupyter notebooks are NOT allowed.)

Python