Ethan Davidson

Luke Dosen

Austin Eddington

Harmeet Singh

Machine Learning Final Project

Introduction:

The purpose of this project was to take the principles and foundations we have studied in this Machine Learning course thus far and use them to build a predictive model based on our chosen dataset. We chose to analyze a dataset that recorded anonymized credit card transactions with labels that denoted whether the transaction was genuine or fraudulent.

Dataset description:

We downloaded the dataset from Kaggle, and it is formatted as a .csv file with a total of 31 columns, each with a header and a total of 284,807 rows with data to be analyzed. The data collected in this set represents transactions made by credit cards in September of 2013 by european credit card holders. It is highly unbalanced, as there were 492 frauds out of the 284,807 counted transactions which make up a total of 0.172%. The first column, labeled Time, represents the number of seconds elapsed between the transaction at any given row and the first row. The following 28 columns are anonymized features that were transformed into numerical input variables using a PCA transformation. The second to last column represents the transaction amount as a float in dollars and cents. The final column we used as a label, as it

represents whether the charge was labeled fraudulent (represented by 1) or non-fraudulent (represented by 0).

Baseline approach description:

The data had approximately 30,000 training examples and we used 8500 examples for training, and 1500 for test and validation. The data was majority valid charges, with less than 1% being fraudulent charges. The approach we used was to read in the dataset into a 2d array and we used the holdout method and split the data into a train set, test set, and validation set. A support vector machine, logistic regression, and K Nearest Neighbors model was trained on the data and had an accuracy of 100%.

Method description:

To preprocess the data, we used the pandas library to read in the .csv file dataset. To lessen our compile times we decided not to use the full dataset and instead set the X features as the first 10,000 rows (excluding the header row), and used the first five anonymized numerical columns as features to analyze. For our label y we chose the same rows and used the final column that classified whether a transaction was fraudulent. We then used the train_test_split method from the Sklearn library and split the features and label into 70% training set and 30% testing set. We further split the training set in half, to add a validation set.

Evaluation:

Out of the three models tested in this analysis, Logsitic Regression was the most accurate in determining which cases were fraud. The SVM classifier was the second most accurate at making this classification with the KNN classifier scoring the lowest. Despite some classifiers performing better than others, they all achieved a relatively similar score.

Discussion:

During the training process we discovered a number of outliers but due to time constraints were not able to remove. It is reasonable to believe that these outliers somewhat confused the model during the training phase. It would be interesting to see how much more accurate the model could be if we were able to remove these outliers.

Conclusion:

All in all, we were surprised to see that all three of these classifiers performed relatively similarly in this classification task. If we were able to do this project again, it would be interesting to see how finer parameter tuning could further affect the scores.

Appendix

Ethan: Set up Github repo, completed data preprocessing, implemented svm training, assisted in report writing and code bug fixing

Harmeet: Found dataset, helped with data preprocessing data and training svm and prediction.

Luke: report writing and model creation.