Madelon Report

Proving Classification Model viability on Synthetic Data

Purpose

In recent years, the accumulation of large quantities of data has opened the floodgates on the practicality of machine learning techniques for decision making across a variety of business types. Previously, such methods were the realm of academics or theorists alone. Today, however, machine learning techniques are becoming more usable, as is the accessibility of large quantities of data to train such techniques on.

However, prior to implementation of such techniques in a live decision making setting, there remains the challenge of proving to decision makers (who must decide what avenues to follow on the basis of real dollars spent/earned) the viability of such models.

Our purpose here is to demonstrate the power and accessibility of said models, and further their implementation in the business setting of choice.

Problem Statement

The Madelon dataset is a classic dataset used to demonstrate certain machine learning capacities. The canonical dataset is made up of 4,400 rows of synthetically generated data. An additional dataset, hosted on a sql database server, follows the same generation methodology and consists of up to 200,000 rows. Where the canonical dataset contains 500 columns, plus a target label of 1 or -1, the database set features 1000 rows plus a target. Both datasets contain a certain number of informative features, with some being generative and others redundant.

In order to prove classification model viability, we split the data into train and test sets. Our goal is to predict the target labels on the test set without prior model exposure. We build our model using only data from the train portion.

Methodology

We follow a typical data science methodology to approach and solve the problem of predicting classification labels. Workflow is as follows:

1) Import, Clean, and Examine Data
2) Select features from model
3) Train various models on training data, tuning hyperparameters

4) Score final model

Outcomes

Unsupervised learning methods proved extraordinarily reliable for deciphering informative features from random noise in this dataset. Particularly, a Decision Tree Classifier was run internally against the training dataset on both UCI and Database data, with each feature recursively set as the target and the rest of the features as the training data. 20 features were produced on both sets of data, with different columns proving important for each.

From the 20 imformative features, Principal component analysis revealed that 5 Principal Components clearly discuss the vast majority of the variance in the data, as seen in the chart below.

img/PCA5.png

These 5 Principal components were then fed to a K Neighbors Classifier and a Support Vector Classifier, both of which had comparable results. On the UCI dataset, R2 scores were .89 and .90 for KNC and SVC test sets. Results on the database set scored noticably lower, with a final test score of .789 using a voting classifier trained with KNC and SVC models. It is likely the lower score is due to a higher degree of randomness inherent in the data.

In addition to unsupervised learning, multiple other feature selection techniques were utilized, including SelectKBest and SelectFromModel. Ultimately, however, understanding of how the dataset was generated in combination with superior results led to only retaining the features extracted through unsupervised learning.

In the future, model performance could potentially be increased through further work in selecting better features, rather than using PCA to narrow the feature set from 20 to 5. Furthermore, neural networks have been shown to provide perfect solutions to this problem, and would be worthwhile to investigate for use in future problems.