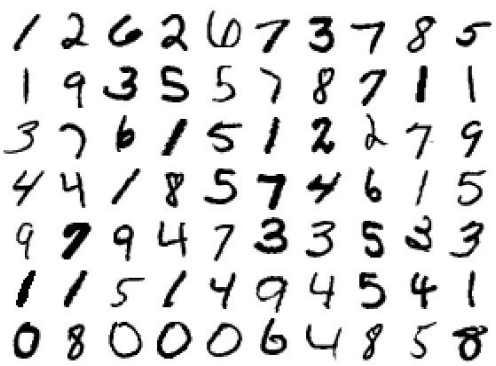
Christopher Rico
MSDS 422
Assignment 4
10/20/19

Introduction

Within the realm of data science, there is a constant quest to optimize the tradeoff between model accuracy and speed. In a situation with boundless memory and compute speeds, accuracy would always win. However, that is almost never the case, so at times it becomes necessary to take a closer look at what a compromise between accuracy and training speed really means.

Computer vision, one of the most intensive topics in data science, tends to be notoriously resource-heavy, requiring vast amounts of input data, memory, and processor time, so image classification model optimization is of great interest. In furtherance of optimizing this trade-off, two computer vision modeling approaches will be attempted on the MNIST dataset. MNIST, considered to be the "hello world" of image classification, is a well-recognized series of several thousand handwritten digits between 0-9. A modeling technique will be developed that will be performant in classification results and training time.

Research Design and Statistical Methods



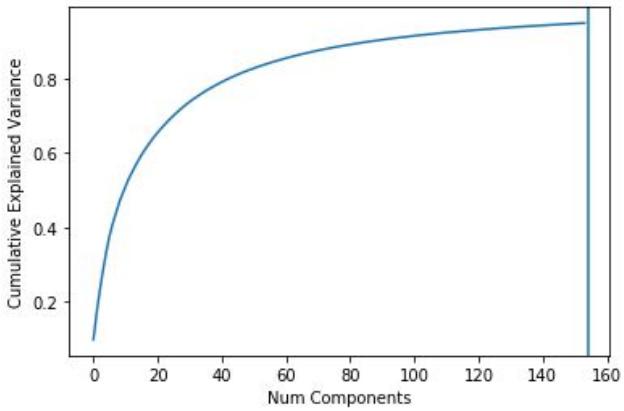
Sample of MNIST images

The MNIST dataset is a well-known dataset that consists of 70,000 28x28px images of hand-written digits between 0-9. Each data point contains 785 variables: 784 pixel values and 1 label. The machine learning competition website Kaggle has broken up these 70,000 images into a 42,000 image training set and a 28,000 image test set, which will be the training and test datasets used to asses the modeling approaches.

The first approach taken is to establish baseline results using a random forest classifier (RFC) trained on all 784 explanatory variables. Training speed and accuracy will be assessed, then principal components analysis (PCA) will be performed on the dataset. PCA is a type of unsupervised learning that attempts to reduce the dimensionality of the dataset by retaining all components of the data that are sources of variance, while removing

those which do not contribute to the data variance. This has the effect of making a dataset less computationally intensive to run through a model.

Programming Work



Model construction and evaluation was completed entirely in the cloud, using Google Colaboratory.

Model development was performed in the cloud using Google Colab as the development environment and Scikit-Learn as the data science package.

An RFC was created, then timed while fitting and predicting on the MNIST dataset. Then, PCA was used to collapse the dataset down to components that represented 95% of the variance. For the MNIST dataset, this means using 154 of the original 784 explanatory pixels, a large decrease. Finally, a second RFC was timed while fitting and predicting on the reduced MNIST dataset. Predictions were submitted to Kaggle for scoring.

Results and Recommendation

The RFC trained on all 784 components scored .938, with a training and prediction time of 4.4 seconds. However, RFC trained on principal components only scored 0.099 with a combined PCA, training and prediction time of 24 seconds.

Only one recommendation can accompany these results: performing PCA on these data is not a worthwhile trade-off. A RFC trained on all 784 components offers fast computational speed and good accuracy.

A shared version of the interactive Jupyter notebook used to run this analysis can be found at:
https://colab.research.google.com/drive/1LYeGjNhGk3nmKEDOZKJaCp6VtDsOvk8A

My Kaggle profile with predictions can be found at:
https://www.kaggle.com/christophrico