Chase Headley
Neal Sakash
Dr. Anderson
CSCI 334
5/1/17

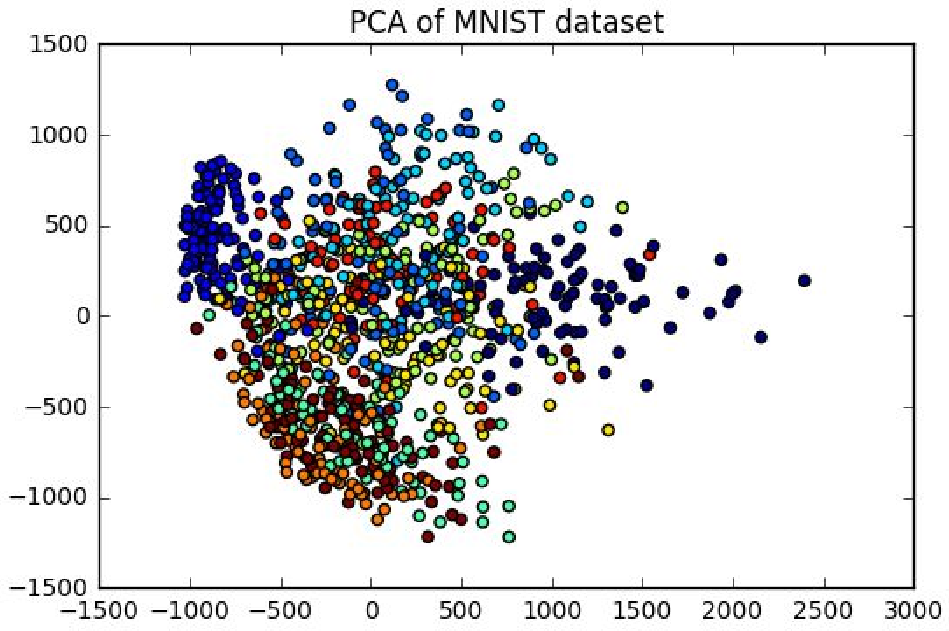
Programming Assignment Three

To demonstrate the use of principal component analysis we again used the MNIST dataset from assignment three. Our first step was to visualize the data using the PCA. 1000 samples were taken at random.

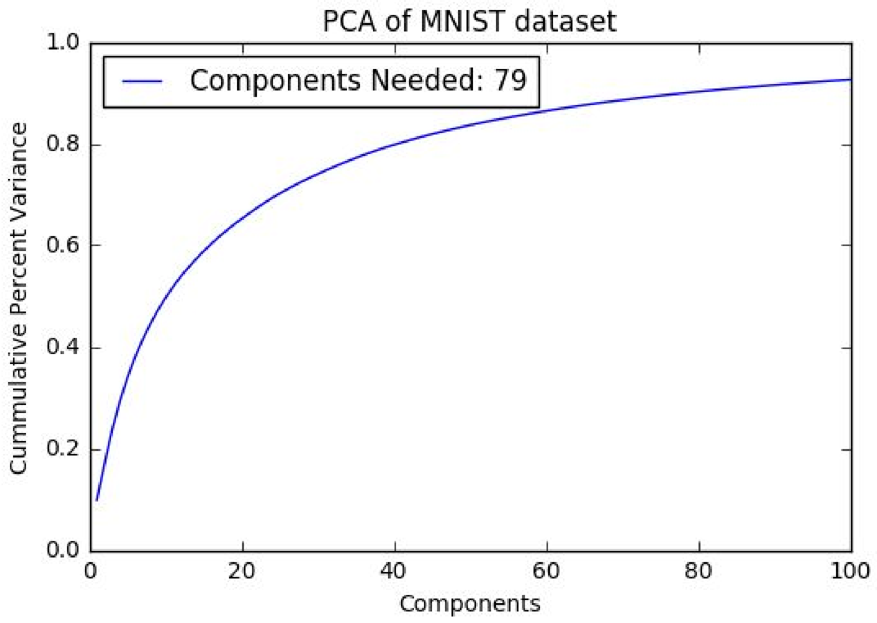
```
#Sample data
inxs = np.random.randint(y.shape[0], size=1000)
y = y[inxs]
x = x[inxs,:]
```

Next, using the PCA functions imported from scikit-learn, we visualized the first two principal components

```
Automatically created module for IPython interactive environment
FIRST TWO PRINCIPLE COMPONENTS:
(Explained Variance Ratio)
 [ 0.09904877  0.07394052]
```

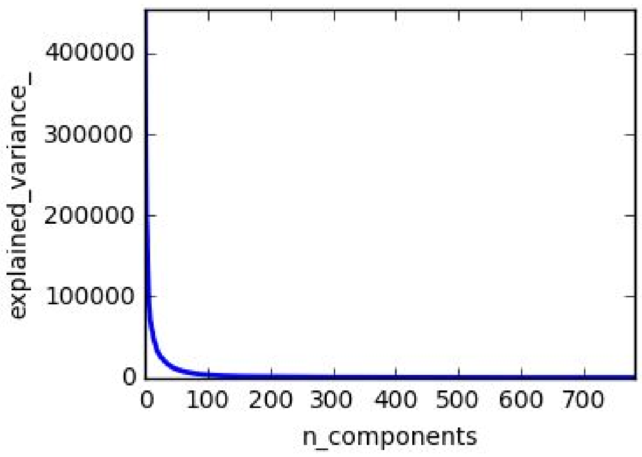


Our next task was to determine the amount of principal components needed to reach 90% of the explained variance. Running a simple loop of 100 iterations we plotted the cumulative percent variance per the number of components. We counted each occurrence of components above 90% and used this number to determine the number needed to reach 90%.

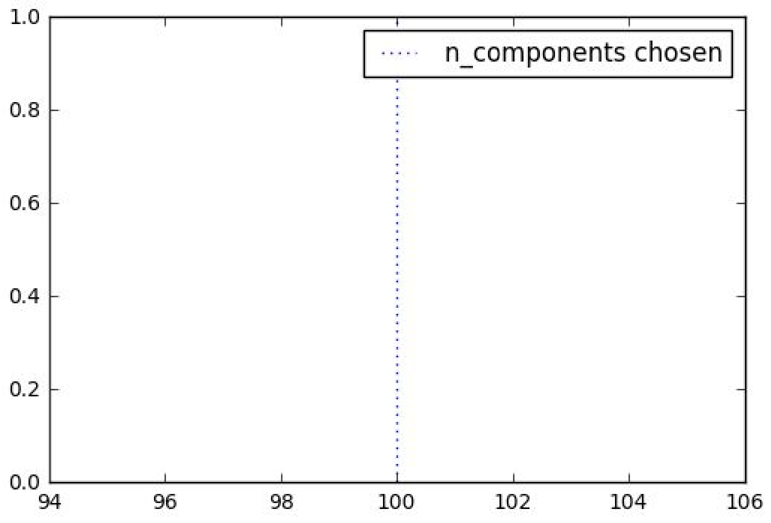


For our next task we practiced with another example of using PCA to chain it with a classifier. With this example we used logistic regression. Picking only the “6” and “8” digits we plotted the PCA Spectrum.

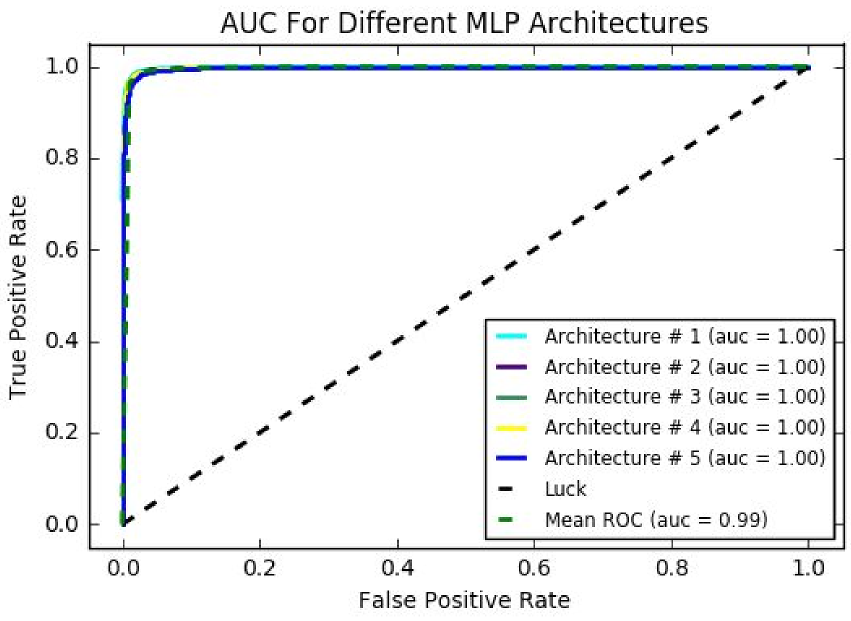
```
Shape of data and labels are : (13701, 784) (13701,)
```



For our prediction we chose the components 75, 100, and 125. 100 was determined to be the best estimator



Moving on we then chained the pca to the MLP Classifier from a previous assignment. We ran five different architectures using multiple iterations. All validation scores were above 90%. A very noticeable difference from our previous use of the classifier in assignment two.



Chaining these two showed the usefulness of chained classification in machine learning. Using the PCA and MLP classifier together gave a more accurate prediction model. The PCA reduced the unsupervised dimensionality of the MNIST dataset, while the MLP iteratively trained and mapped the dataset into the appropriate outputs for the model.