

# Homework 1

**Name:** Mukul Sati

**GTID:** 902917135

February 9, 2016

# Contents

<b>1 Homework 1</b>	<b>2</b>
1.1 PCA: . . . . .	2
1.2 LDA . . . . .	5

# Homework 1

## 1.1 PCA:

For my implementation of PCA, I show the 10th eigenvector instead of the 20th, as the 20th is zero as the data is 13-dimensional. The most important eigenvector, the 10th eigenvector and the plot for the ratio of the sorted eigenvalues to the sum of the eigenvalues for the wine dataset for two runs, one with 5 and the other time with 50 random samples of each class selected for training are:

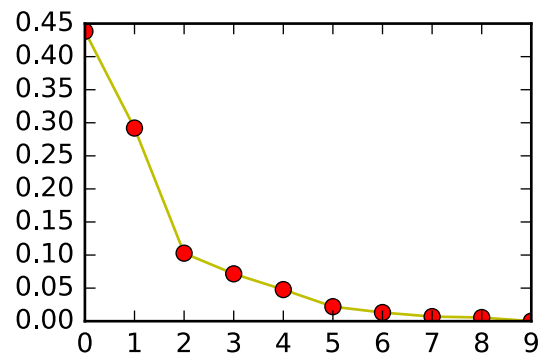
Using 5 samples for training:

1th eigenvector:

```
[-0.17329488 -0.20271621  0.3047377   0.31665541 -0.29724256 -0.39521372
 -0.34620417  0.33706582 -0.29203623 -0.23723838  0.29240437 -0.13024439
 -0.11700622]
```

10th eigenvector:

```
[ 0.60522263  0.05808817 -0.06650262  0.40954098  0.15297067 -0.50725881
  0.16167838 -0.16979499  0.23088354 -0.11639725 -0.10762356 -0.15024257
 -0.14660136]
```



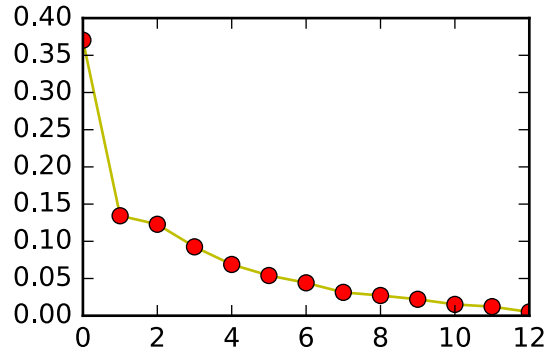
Using 50 samples for training:

1th eigenvector:

```
[-0.36074472  0.04586331 -0.14700784  0.21151058 -0.23564709 -0.3775952
 -0.38762705  0.23752171 -0.2339237   -0.38124082 -0.00746195 -0.22962842
 -0.37918735]
```

10th eigenvector:

```
[-0.32656667  0.11534686  0.57177716 -0.5509419  -0.15439106  0.09300774
  0.06585905 -0.31385883 -0.12714822 -0.2333455   0.0708041  -0.19313485
 -0.036336   ]
```



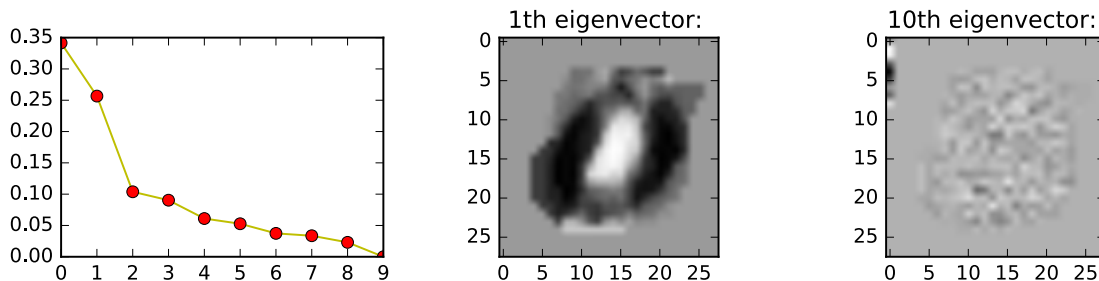
I also use a variable number of eigenvectors for the reconstruction, discarding those for which the ratio:  $(\text{eigenvalue})/(\text{sum of eigenvalues})$  falls below a threshold (I use a threshold of 0.03). Using this to reconstruct a test-example, I get the following reconstruction errors (I report both the standardized and non-standardized errors):

When using 5 samples for training:  
 Number of eigenvectors being used 6  
 Number of eigenvectors being used 6  
 Reconstruction error (standardized): 2.71043010221  
 Reconstruction error: 382.636408354

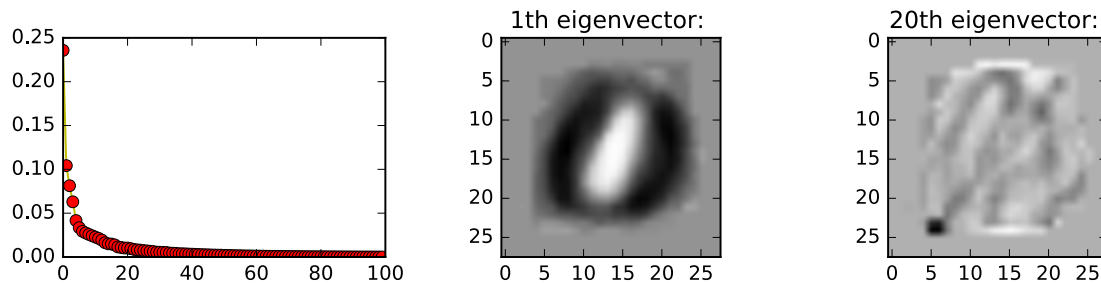
When using 50 samples for training:  
 Number of eigenvectors being used 9  
 Number of eigenvectors being used 9  
 Reconstruction error (standardized): 1.52713874005  
 Reconstruction error: 271.290247093

For mnist, I show the 10th eigenvector when using just 5 samples from each class and the 20th otherwise. The most important eigenvector, the 10th eigenvector and the plot for the ratio of the sorted eigenvalues to the sum of the eigenvalues for the wine dataset for two runs, one with 5 and the other time with 50 random samples of each class selected for training are:

Using 5 samples for training:



Using 50 samples for training:

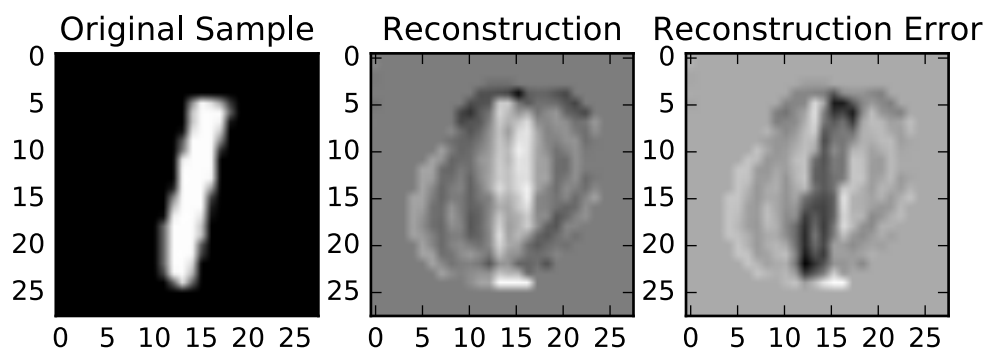


Similar to the wine data-set, I again use a variable number of eigenvectors for the reconstruction, discarding those for which the ratio:  $(\text{eigenvalue})/(\text{sum of eigenvalues})$  falls below a threshold (I use a threshold of 0.0005 here). Using this to reconstruct a test-example, I get the following reconstruction and errors:

When using 5 samples for training:

Number of eigenvectors being used 10

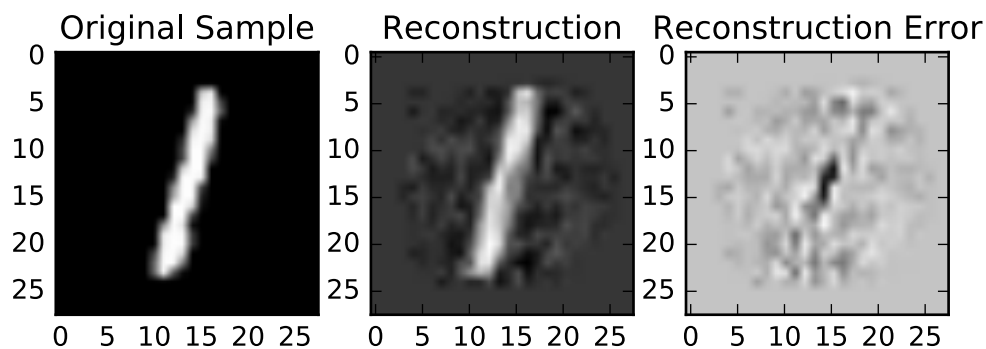
Quantitative reconstruction error: 6.19755694843



When using 50 samples for training:

Number of eigenvectors being used 76

Quantitative reconstruction error: 2.40242376975



## 1.2 LDA

I perform LDA for class discrimination maximizing dimensionality reduction after performing PCA for class agnostic dimensionality reduction. The most important eigenvector, the 10th eigenvector and the plot for the ratio of the sorted eigenvalues to the sum of the eigenvalues for the wine dataset for two runs, one with 5 and the other time with 50 random samples of each class selected for training are:

Using 5 samples for training:

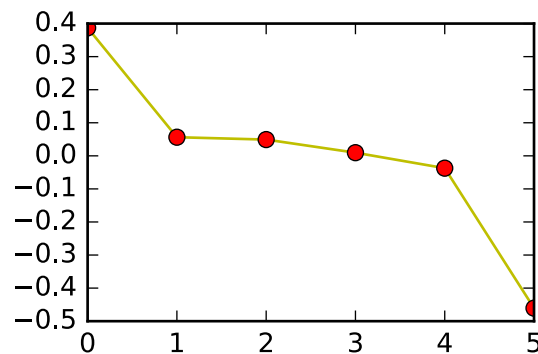
First reducing dimensions by PCA to 6

1th eigenvector:

```
[ 0.29843372  0.66332795 -0.25177639 -0.06559815 -0.27536181 -0.16917021
  0.08292033  0.28106747 -0.16475257  0.20097222  0.10967338  0.36485704
 -0.01547049]
```

6th eigenvector:

```
[ 0.04976015  0.18888549  0.30857903  0.35719219  0.53742549 -0.3754083
 -0.22672149  0.4567791  -0.06042899  0.01389432 -0.10547593 -0.17664826
  0.05529106]
```



Using 50 samples for training:

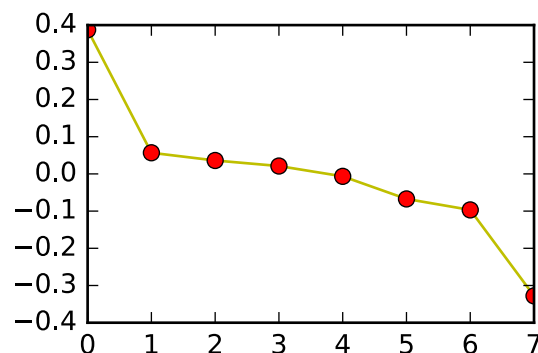
First reducing dimensions by PCA to 8

1th eigenvector:

```
[-0.31813651  0.28219942 -0.10883571  0.06746691 -0.08983933  0.13389594
  0.01737428  0.24901605 -0.12166199 -0.45685287  0.32797142  0.36946639
 -0.49684877]
```

8th eigenvector:

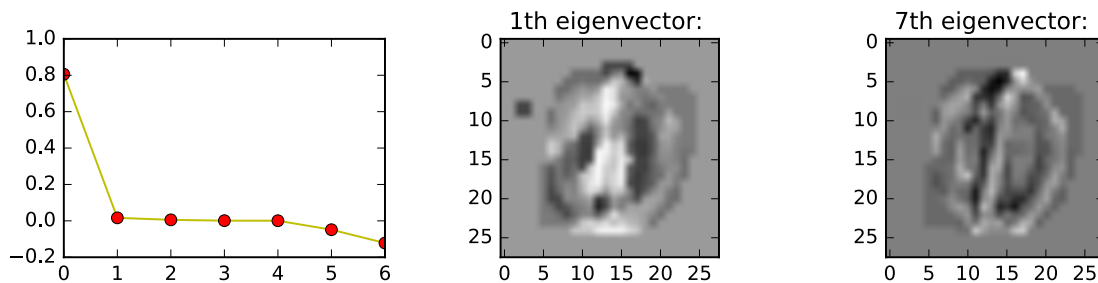
```
[-0.28705488  0.20787809 -0.01802137  0.43791094 -0.12295442 -0.25010104
 -0.08499514 -0.14146212  0.66773616 -0.10349893 -0.27291773 -0.17162908
 -0.12987367]
```



I carry out LDA post PCA similarly for mnist data.

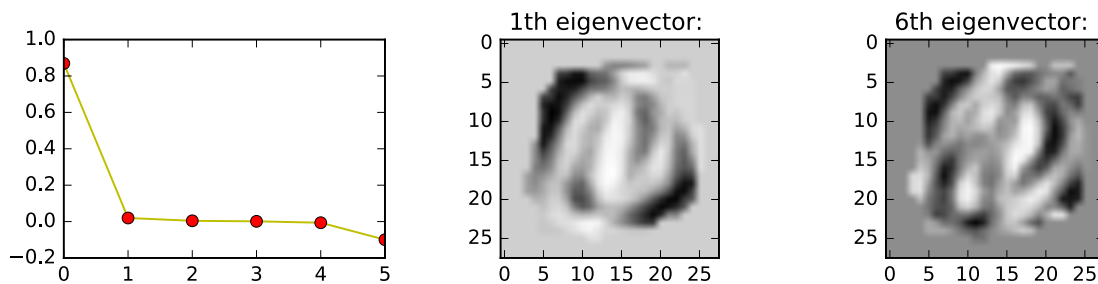
Using 5 samples for training:

First reducing dimensions by PCA to 7



Using 50 samples for training:

First reducing dimensions by PCA to 6

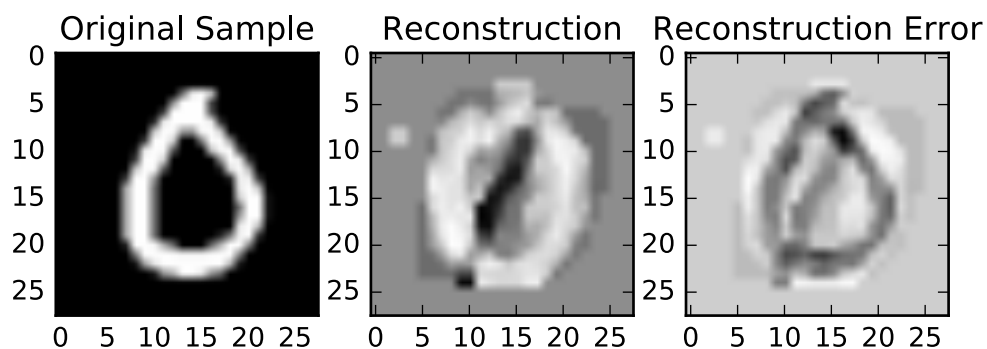


Lets view the reconstructions from this:

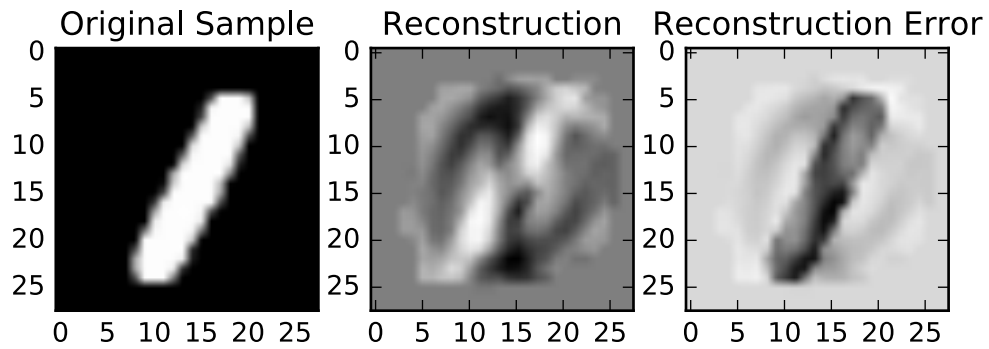
When using 5 samples for training:

Number of eigenvectors being used 6

Quantitative reconstruction error: 7.86467257675

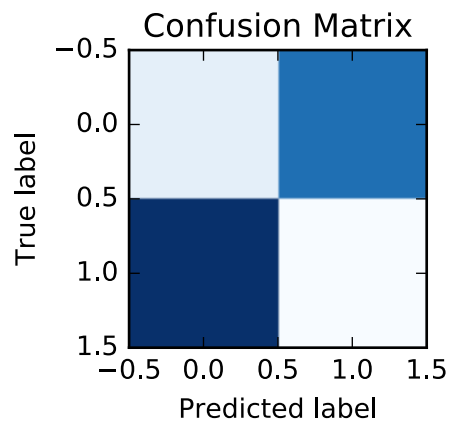


When using 50 samples for training:  
 Number of eigenvectors being used 5  
 Quantitative reconstruction error: 8.86273990486



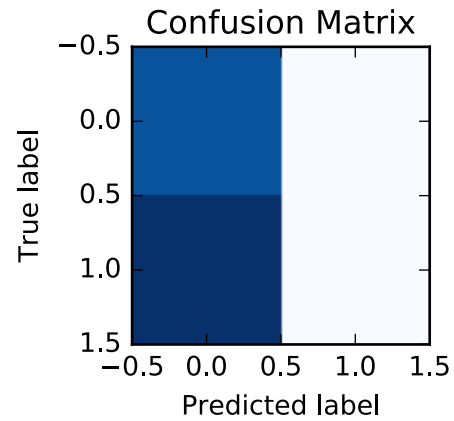
Now, I use the discriminant to classify:

When using 5 samples for training:  
 Accuracy = 9.598109%



When using 50 samples for training:  
 Accuracy = 46.335697%





-----

NameError

Traceback (most recent call last)

```
<ipython-input-130-d480a42849eb> in <module>()
    1 #Train
----> 2 trainingMeans = [trainingDataUnlabeled[0].mean(), trainingDataUnlabeled[1].mean()]
    3 trainingVariance = totalTrainingData.var()
    4 weights = (trainingMeans[1]*trainingMeans[1] - trainingMeans[0]*trainingMeans[0])/trainingV
    5 bias = 0.5 + ((trainingMeans[1]*trainingMeans[1] - trainingMeans[0]*trainingMeans[0])/2*tra
```

NameError: name 'trainingDataUnlabeled' is not defined