

Homework 8

Procedure

First, We took first 100 vectors from each class as **train data**, and save the rest 100 from each class as **test data**.

We did **PCA** for dimension reduction. Since the dimension of patterns ($N = 1000$) is much larger than the dimension of space(240), we choose our k to be 240, as identity operator. (We also tried set precision of dimension reduction to be 99.5%, then we get k to be 170) Through **SVD**, U as a $240 * 240$ is our matrix for further operation in Linear regression.

Then, we add padding for reduced train data, and we created a $1000 * 10$ matrix Z . Every 100 rows stand for the same class, which has 1 at corresponding position, and 0 at rest.

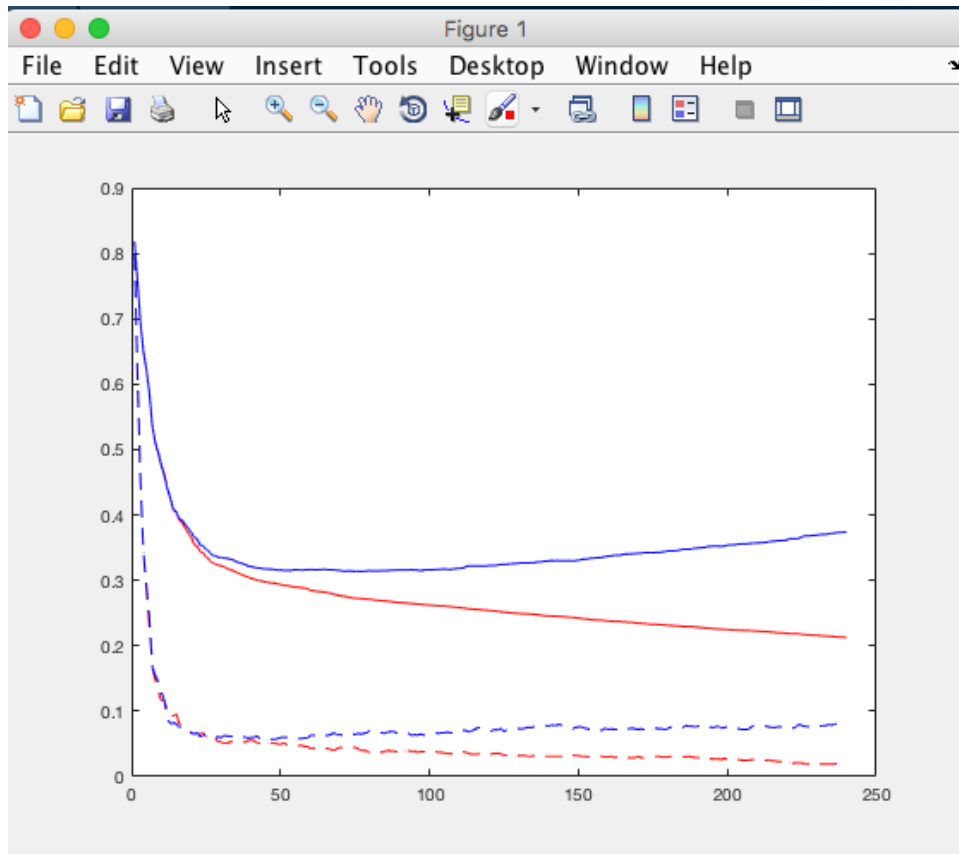
Next, we compute W_{opt} through the projection of all train data onto U matrix, using its transpose as matrix Φ , and project Z onto the subspace Φ formed, which is the smallest distance from Z to the space. That is our optimal decision function W_{opt} .

Next, We compute the **mean square error**, and **misclassification rate** through a loop on all train data, using MAX Ind to find the index of max value of current vector, while using find command to locate the supposed current class, and compare them, if not equal, number of misclassification add 1. We went through the whole loop, then compute how much misclassification in total data.

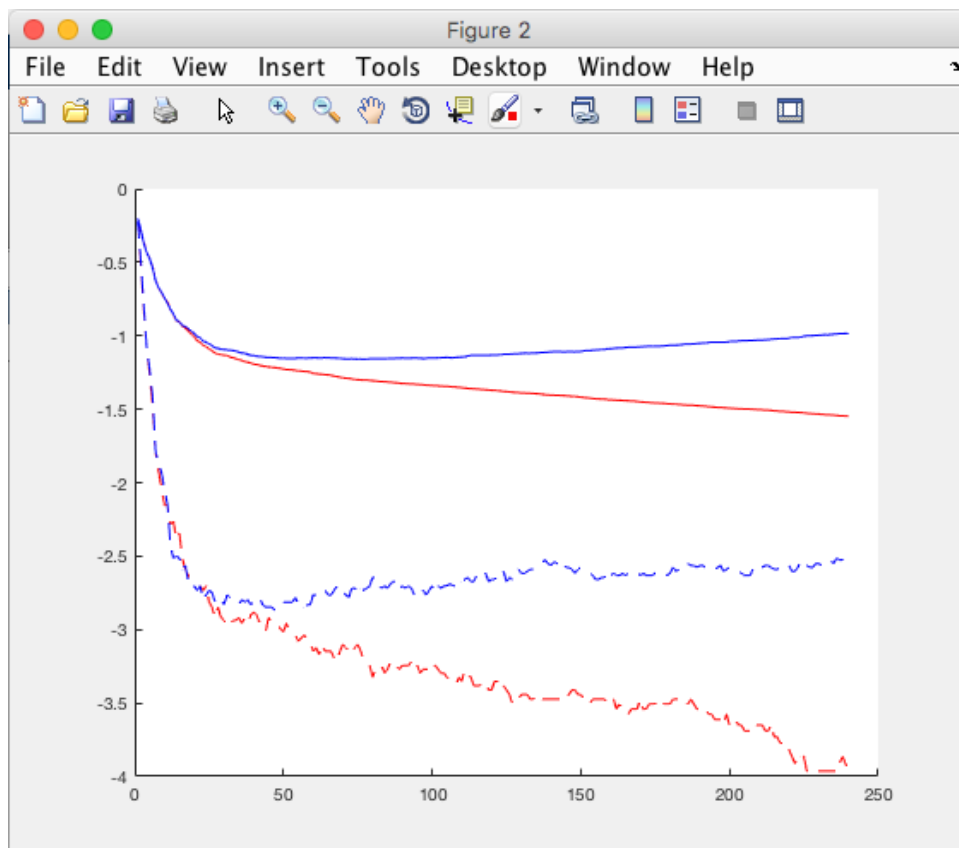
Next, start testing. In a large loop for the feature K to be 1,2...240. Then U matrix will vary from containing only U_1 , to $U_1 U_2$, till full U . (also padding one as bias in the last) Mapping new data onto varying U , Using the same W_{opt} we have from previous to compute test error. Then the rests are similar as before.

Observation

Linear Scaling:



Logarithmic Scaling:



MSE train: solid red
MSE test: solid blue
MISS train: dashed red
MISS test: dashed blue

Discussion

We could tell from the graph, the train error is drastic decreasing from beginning, and gradually converge to 0. However, the test data has reach its minimum around $K = 40$, then it starts to increasing once again. Then 40 will be our optimal value to get the max correct classification. if we took $K < 40$, at very left, this would be a underfitting problem, which may mean that this kind of classification can not meet the user's requirement, and couldn't really tell the characteristics of data. This Model is too inflexible. If K is chosen to be much larger than 40, it is a overfitting problem, which means this model is too flexible, later on using on test data, it will lead to terrible misclassification. We actually use the case when $K = 170$, since we tried data reduction precision to be 99%, and get $m = 170$ in Matrix U . It is a bit better than the case $K = 240$, but still a overfitting problem.