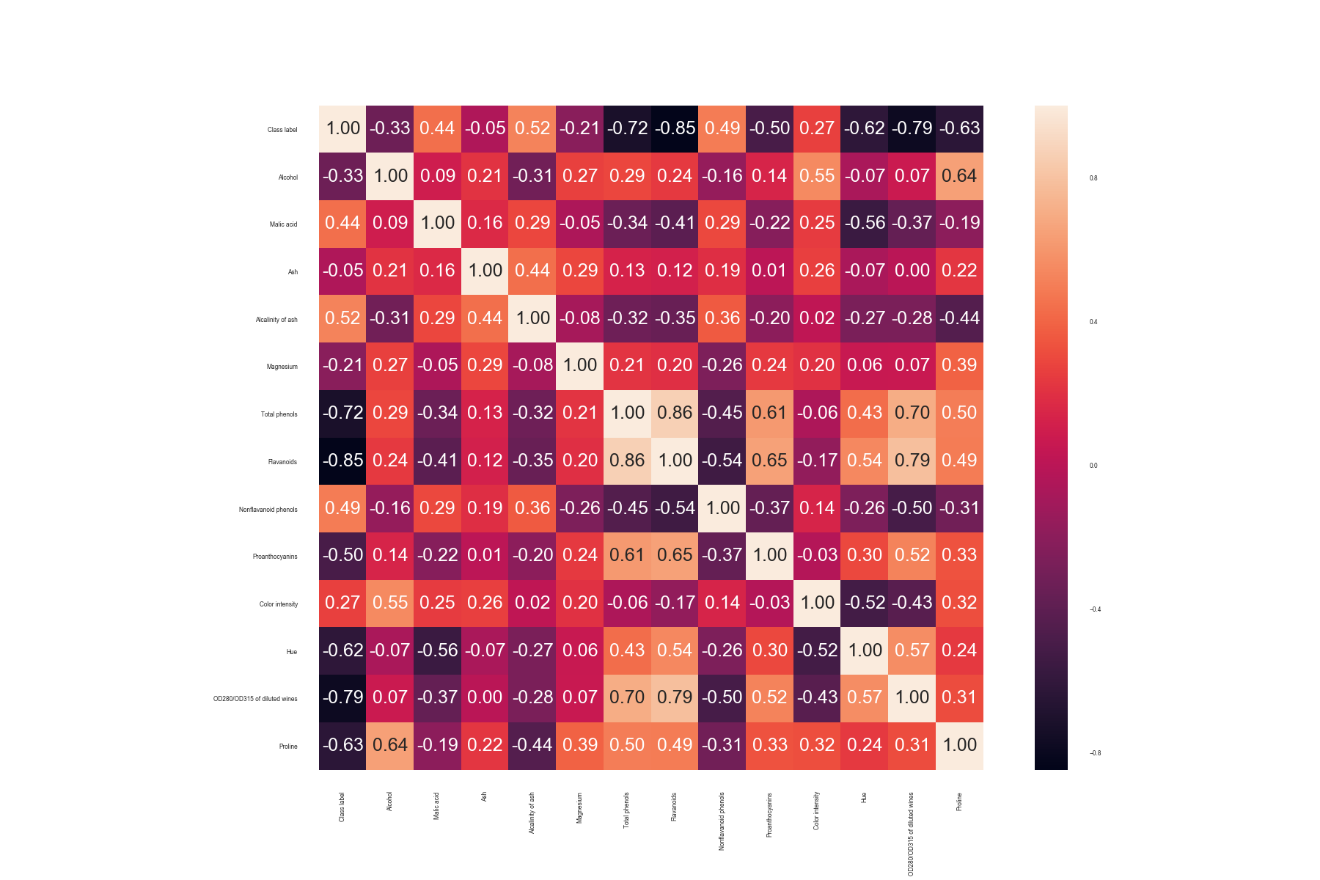
Ruozhong Yang (ry8)

IE598 MLF F18

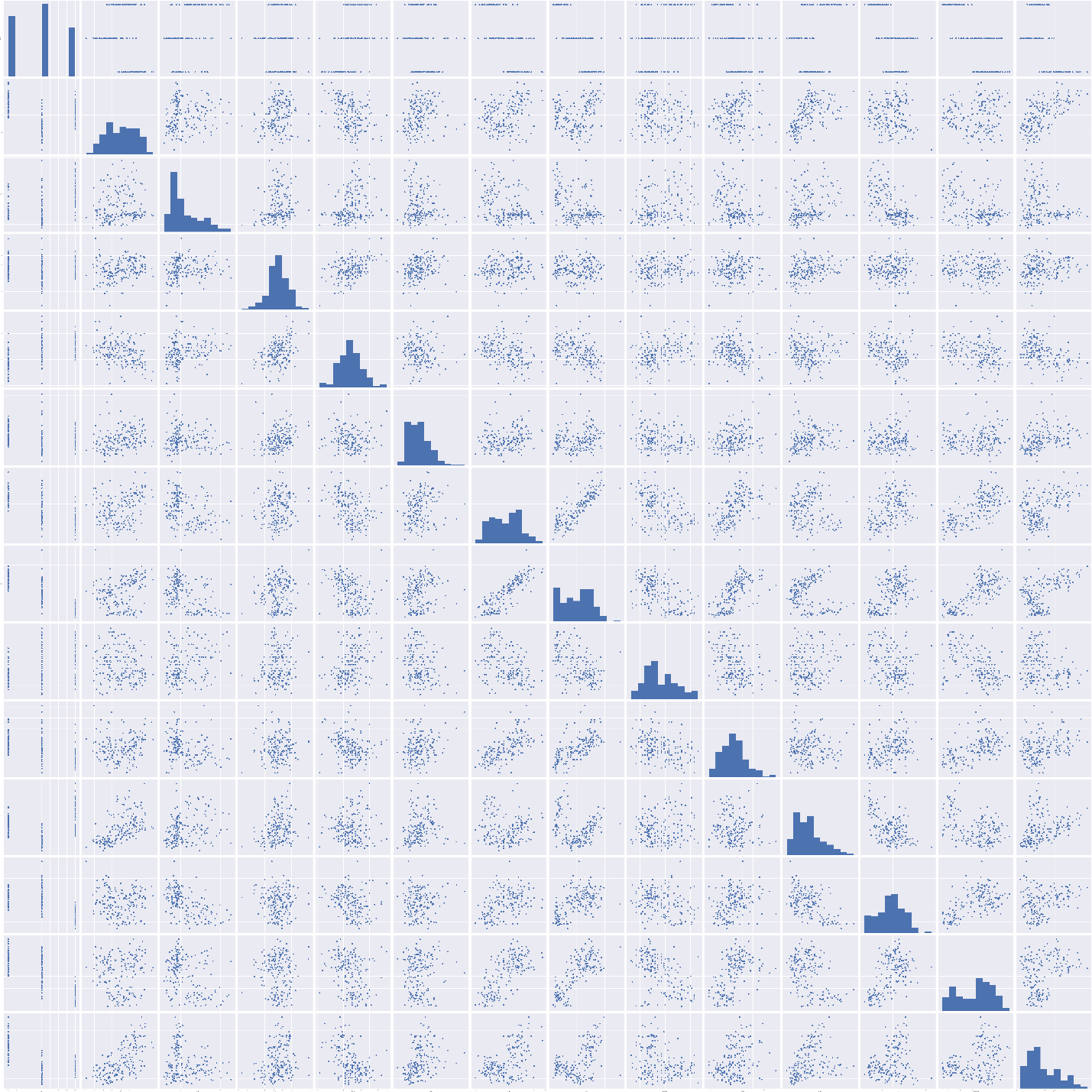
Module 5 Homework (Dimensionality Reduction)

**Part 1: Exploratory Data Analysis**

The first step is to understand the data base with 178 rows and 14 columns. There are several features and a label. In other to understand to relation between features, we print the correlation matrix and the scatterplot matrix. Those two matrixes are such huge matrixes that I have to submit it in other file to get a clear view.



* 1. Correlation matrix



* 1. Scatterplot matrix(all)

Later, we split data into training and test sets. Use random\_state = 42. Use 80% of the data for the training set. Use the same split for all experiments. In this

**Part 2: Logistic regression classifier v. SVM classifier - baseline**

Fit a logistic classifier model to both datasets using SKlearn. Calculate its accuracy score for both in sample and out of sample (train and test sets). We can get:

(LR\_part2\_train)score: 1.0000000000

(LR\_part2\_test)score: 1.0000000000

Fit a SVM classifier model to both datasets using SKlearn. Calculate its accuracy score for both in sample and out of sample (train and test sets). We can get:

(SVM\_part2\_train)score: 0.9929577465

(SVM\_part2\_test)score: 0.9722222222

**Part 3: Perform a PCA on both datasets**

Refit both a logistic and SVM classifier on the PCA transformed datasets. In all those cases, we choose 13 components. Calculate accuracy scores for both in sample and out of sample (train and test sets) on both datasets. We can get:

(LR\_part3\_train)score: 1.0000000000

(LR\_part3\_test)score: 1.0000000000

(SVM\_part3\_train)score: 0.9929577465

(SVM\_part3\_test)score: 0.9722222222

**Part 4: Perform and LDA on both datasets**

Refit both a logistic and SVM classifier on the LDA transformed datasets. In all those cases, we choose 13 components. Calculate accuracy scores for both in sample and out of sample (train and test sets) on both datasets. We can get:

(LR\_part4\_train)score: 1.0000000000

(LR\_part4\_test)score: 0.9722222222

(SVM\_part4\_train)score: 1.0000000000

(SVM\_part4\_test)score: 0.9722222222

**Part 5: Perform a kPCA on both datasets**

Refit both a logistic and SVM classifier on the kPCA transformed datasets. Use the rbf kernel. Test several different values for Gamma. Calculate accuracy scores for both in sample and out of sample (train and test sets) on both datasets. In this case, we choose gamma form10^(-3)to 1. We can get:

((gamma=0.001 )LR\_part5\_train)score: 0.8450704225

((gamma=0.001 )LR\_part5\_test)score: 0.8611111111

((gamma=0.01 )LR\_part5\_train)score: 0.9788732394

((gamma=0.01 )LR\_part5\_test)score: 0.9444444444

((gamma=0.1 )LR\_part5\_train)score: 0.9929577465(best result)

((gamma=0.1 )LR\_part5\_test)score: 1.0000000000(best result)

((gamma=1.0 )LR\_part5\_train)score: 0.5563380282

((gamma=1.0 )LR\_part5\_test)score: 0.3888888889

((gamma=0.001 )SVM\_part5\_train)score: 0.8732394366

((gamma=0.001 )SVM\_part5\_test)score: 0.8611111111

((gamma=0.01 )SVM\_part5\_train)score: 0.9788732394

((gamma=0.01 )SVM\_part5\_test)score: 0.9444444444

((gamma=0.1 )SVM\_part5\_train)score: 0.9718309859(best result)

((gamma=0.1 )SVM\_part5\_test)score: 0.9722222222(best result)

((gamma=1.0 )SVM\_part5\_train)score: 0.5000000000

((gamma=1.0 )SVM\_part5\_test)score: 0.3888888889

Through all those results the best one of logistic regression is the result when gamma=0.1, the best one of SVM is also when gamma=0.1

**Part 6: Conclusions**

In this case, the best model of each part all do a good job. Their scores are so high that I am concerning about overfitting.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Experiment 1 (Wine) | | | |
|  | Logistic | | SVM | |
|  |  |  |  |  |
| Baseline | Train Acc: | 1 | Train Acc: | 0.992958 |
| Test Acc: | 1 | Test Acc: | 0.972222 |
|  |  |  |  |  |
| PCA transform | Train Acc: | 1 | Train Acc: | 0.992958 |
| Test Acc: | 1 | Test Acc: | 0.972222 |
|  |  |  |  |  |
| LDA transform | Train Acc: | 1 | Train Acc: | 1 |
| Test Acc: | 0.972222 | Test Acc: | 0.972222 |
|  |  |  |  |  |
| kPCA transform | Train Acc: | 0.992958 | Train Acc: | 0.971831 |
| Test Acc: | 1 | Test Acc: | 0.972222 |
|  | (gamma=0.1) | | (gamma=0.1) | |

* 1. Results worksheet

**Part 7: Appendix**

<https://github.com/yrz437396236/IE598_F18_HW1/tree/master/IE598_F18-HW5>

Raw predict scores:

(LR\_part2\_train)score: 1.0000000000

(LR\_part2\_test)score: 1.0000000000

(SVM\_part2\_train)score: 0.9929577465

(SVM\_part2\_test)score: 0.9722222222

(LR\_part3\_train)score: 1.0000000000

(LR\_part3\_test)score: 1.0000000000

(SVM\_part3\_train)score: 0.9929577465

(SVM\_part3\_test)score: 0.9722222222

(LR\_part4\_train)score: 1.0000000000

(LR\_part4\_test)score: 0.9722222222

(SVM\_part4\_train)score: 1.0000000000

(SVM\_part4\_test)score: 0.9722222222

((gamma=0.001 )LR\_part5\_train)score: 0.8450704225

((gamma=0.001 )LR\_part5\_test)score: 0.8611111111

((gamma=0.01 )LR\_part5\_train)score: 0.9788732394

((gamma=0.01 )LR\_part5\_test)score: 0.9444444444

((gamma=0.1 )LR\_part5\_train)score: 0.9929577465

((gamma=0.1 )LR\_part5\_test)score: 1.0000000000

((gamma=1.0 )LR\_part5\_train)score: 0.5563380282

((gamma=1.0 )LR\_part5\_test)score: 0.3888888889

((gamma=0.001 )SVM\_part5\_train)score: 0.8732394366

((gamma=0.001 )SVM\_part5\_test)score: 0.8611111111

((gamma=0.01 )SVM\_part5\_train)score: 0.9788732394

((gamma=0.01 )SVM\_part5\_test)score: 0.9444444444

((gamma=0.1 )SVM\_part5\_train)score: 0.9718309859

((gamma=0.1 )SVM\_part5\_test)score: 0.9722222222

((gamma=1.0 )SVM\_part5\_train)score: 0.5000000000

((gamma=1.0 )SVM\_part5\_test)score: 0.3888888889