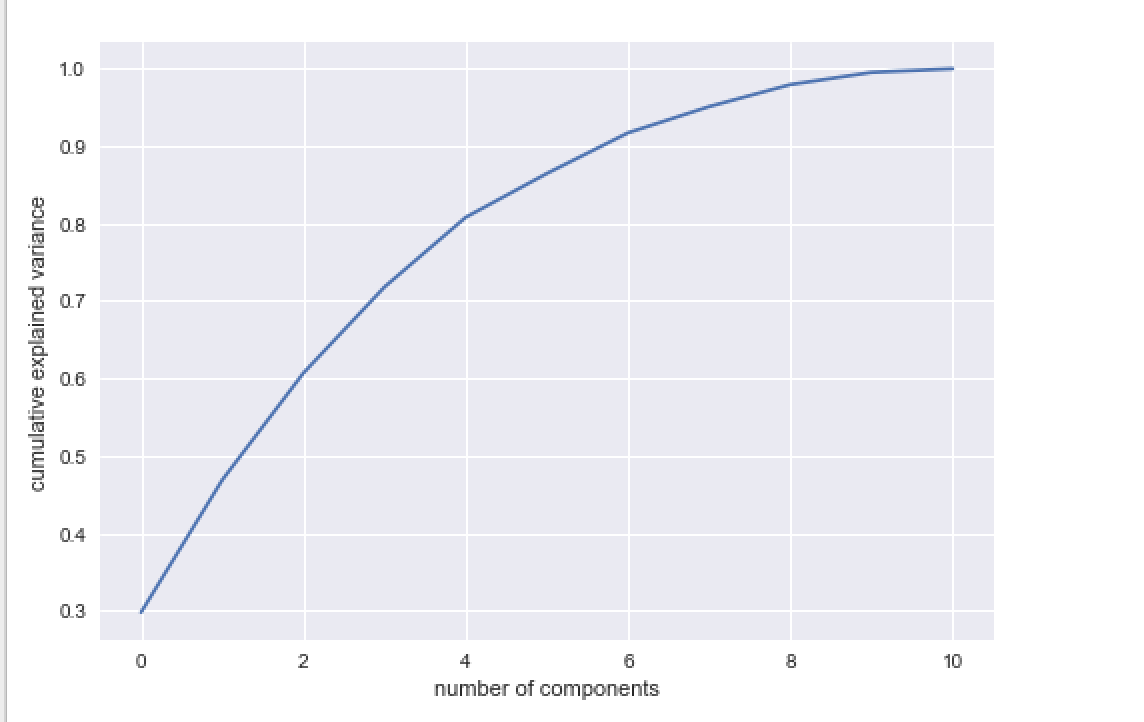
Documentation on PCA:

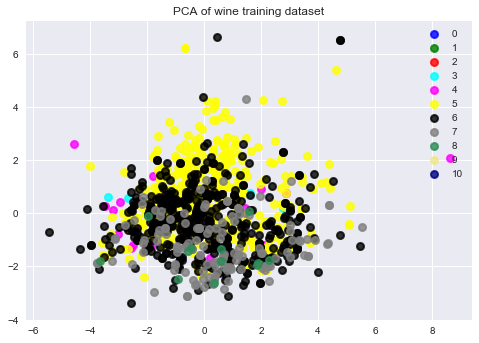
Choosing number of components on feature space in training data:



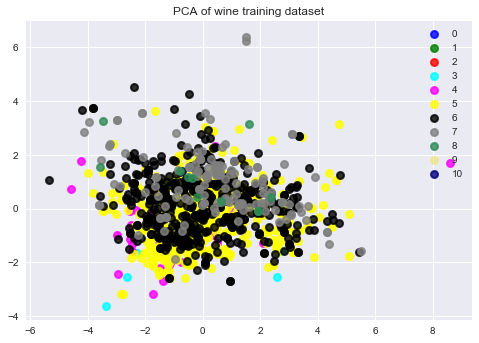
Steep diminishing return of explained variance happens vividly from 8 components with approximately 99 percent of retained variance, hence we decided to use 8 components.

Visualization of training data with quality labels, based on 8 components:

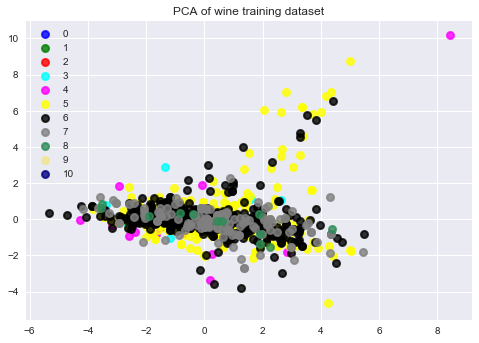
1. PCA-1 vs. PCA-2 feature space:



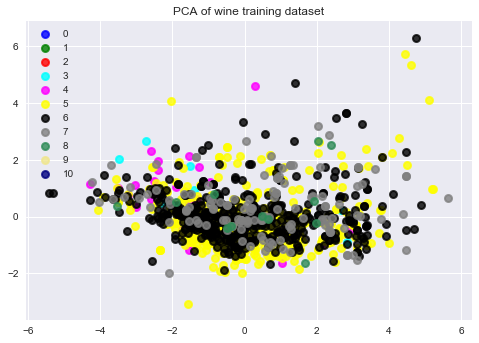
1. PCA-1 vs. PCA-3:



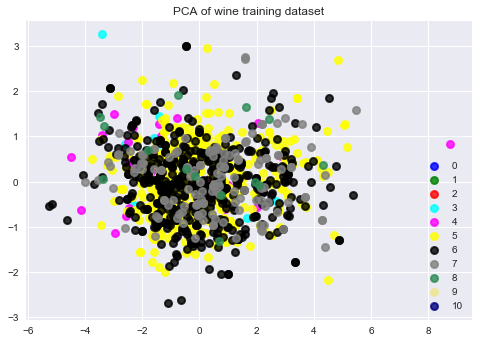
1. PCA-1 vs. PCA-4



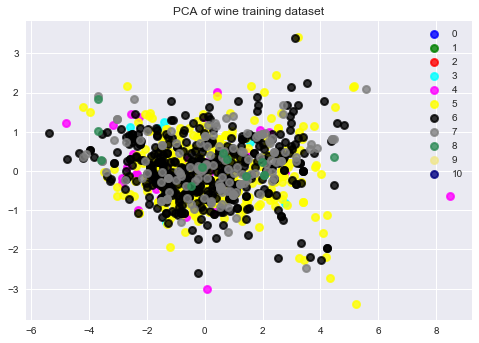
1. PCA-1 vs. PCA-5



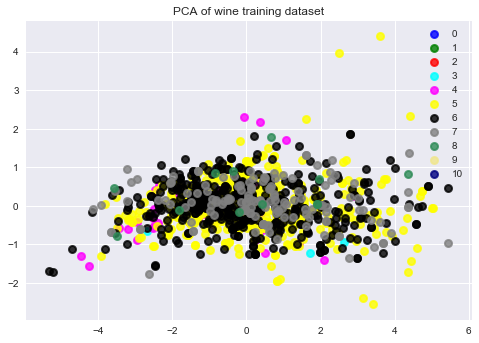
1. PCA-1 vs. PCA-6



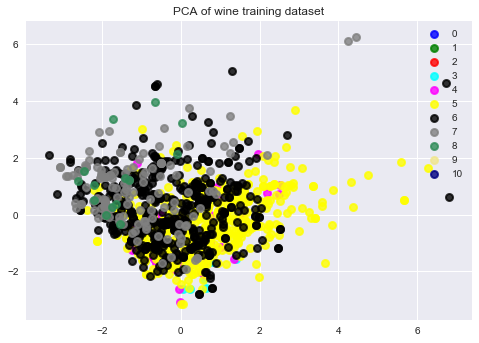
1. PCA-1 vs. PCA-7



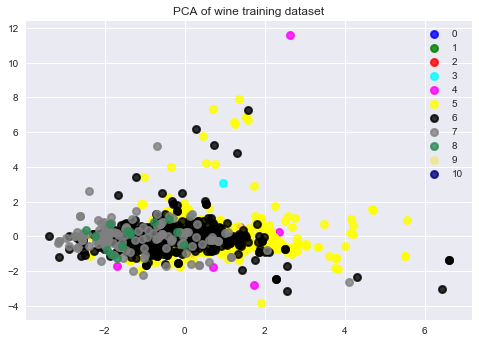
1. PCA-1 vs. PCA-8



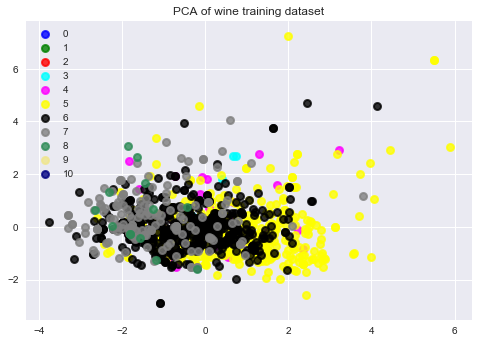
1. PCA-2 vs. PCA-3



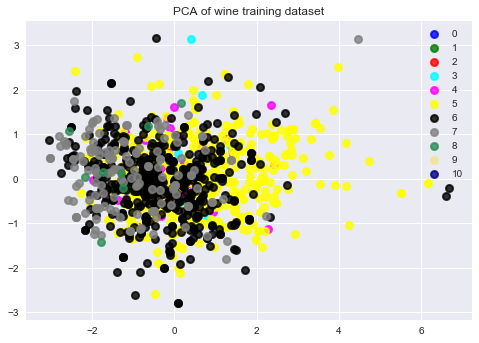
1. PCA-2 vs. PCA-4



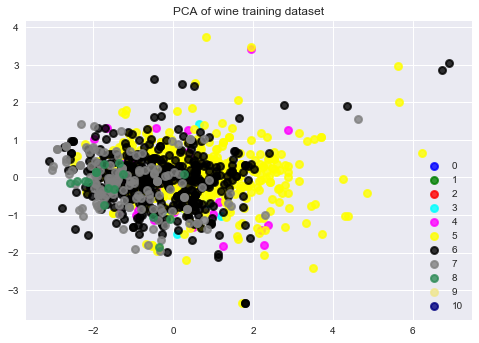
1. PCA-2vs. PCA-5



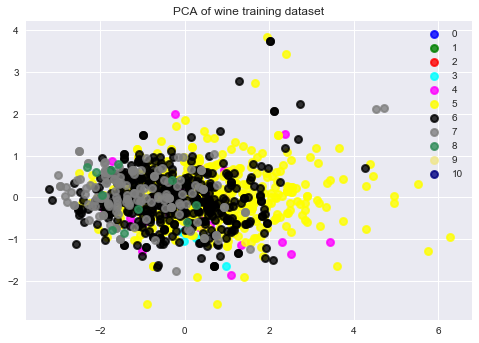
1. PCA-2 vs. PCA-6



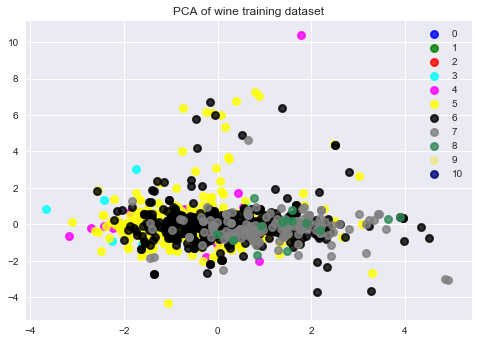
1. PCA-2 vs. PCA-7



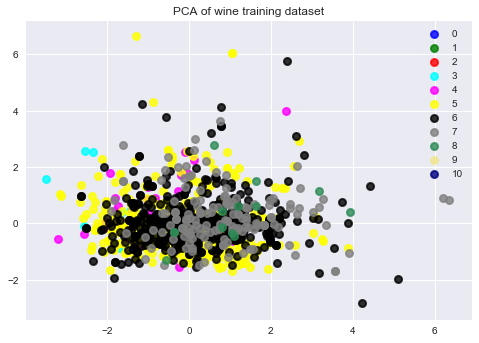
1. PCA-2 vs. PCA-8



1. PCA-3 vs. PCA-4



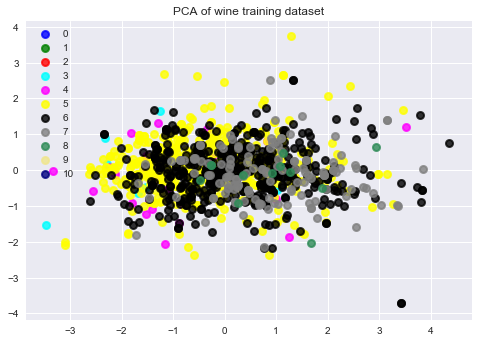
1. PCA-3 vs. PCA-5



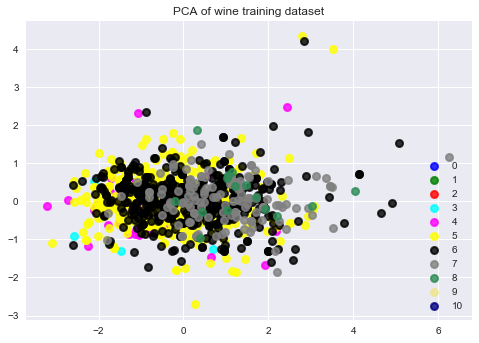
1. PCA-3 vs. PCA-6



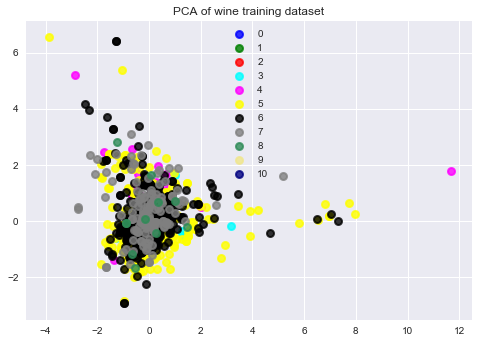
1. PCA-3 vs. PCA-7



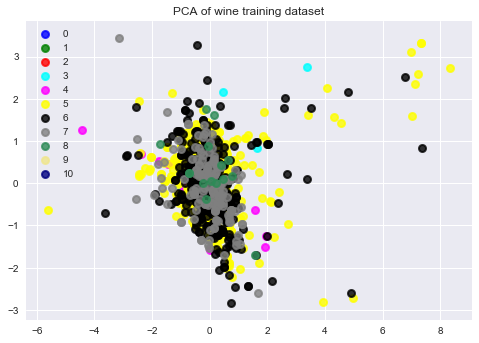
1. PCA-3 vs. PCA-8



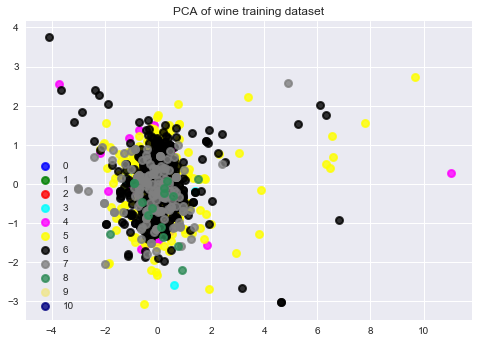
1. PCA-4 vs. PCA-5



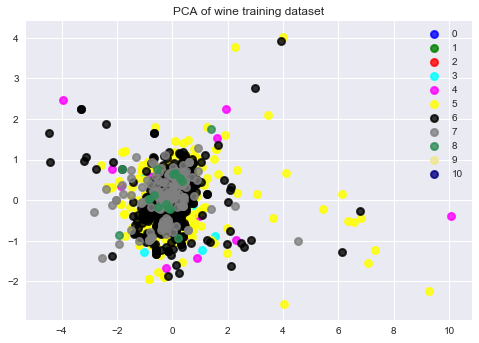
1. PCA-4 vs. PCA-6



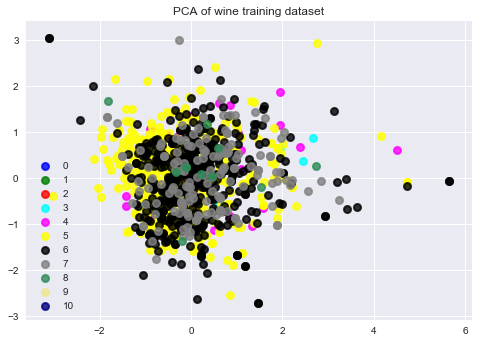
1. PCA-4 vs. PCA-7



1. PCA-4 vs. PCA-8



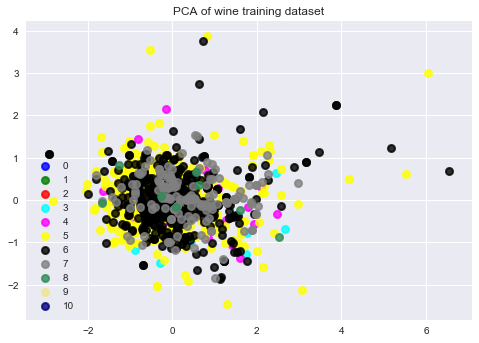
1. PCA-5 vs. PCA-6



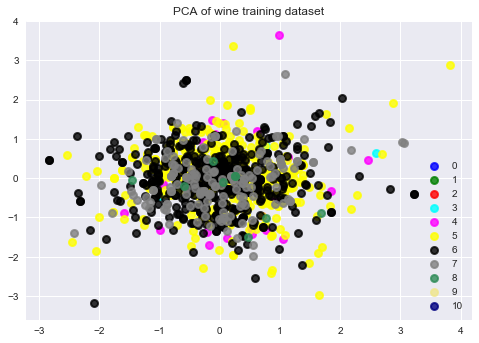
1. PCA-5 vs. PCA-7



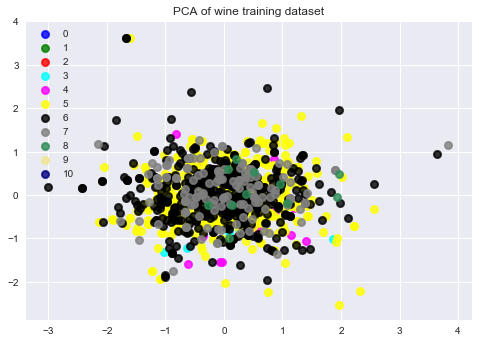
1. PCA-5 vs. PCA-8



1. PCA-6 vs. PCA-7



1. PCA-6 vs. PCA-8



1. PCA-7 vs. PCA-8



There are 28 combinations of PCA features since we have 8 new feature space from PCA-dimensionality reduction.

Application in previous classifier/regression method (with 10-fold cross validation):

1. Multinomial logistic regression(red wine):

* Previous result:
  + Train accuracy : 0.567470956211
  + Test accuracy: 0.5375
* PCA result:
  + Train accuracy: 0.5871313672922251
  + Test accuracy: 0.5729166666666666
* CV prediction error rate(accuracy):
  + Previous: [ 0.50617284 0.61490683 0.52795031 0.50625 0.5375 0.6375 0.5 0.525 0.53797468 0.51592357]
  + Accuracy: 0.59 (+/- 0.14)
  + PCA: [0.41358025 0.60869565 0.59006211 0.4625 0.55625 0.63125 0.5875 0.5625 0.60759494 0.59872611]
* Accuracy: 0.58 (+/- 0.16)

1. Decision tree:

* Previous result:
  + Train accuracy : 1.0
  + Test accuracy: 0.5354166666666667
* PCA result:
  + Train accuracy: 1.0
  + Test accuracy: 0.5333
* CV prediction error rate(accuracy):
  + Previous: [ 0.40740741 0.42857143 0.43478261 0.45625 0.38125 0.45625 0.4375 0.46875 0.49367089 0.49044586]
  + Accuracy: 0.48 (+/- 0.05)
  + PCA: [ 0.48148148 0.54658385 0.47204969 0.3875 0.39375 0.59375 0.49375 0.525 0.49367089 0.45859873]
  + Accuracy: 0.49 (+/- 0.10)

1. Decision tree with bagging:

* Previous result:
  + Train accuracy: 0.90437890974084
  + Test accuracy: 0.6125
* PCA result:
  + Train accuracy: 0.9070598748882931
  + Test accuracy: 0.5583333333333333
* CV prediction error rate(accuracy):
  + Previous: [ 0.5 0.47204969 0.45341615 0.4625 0.5 0.51875 0.49375 0.51875 0.46835443 0.50318471]
  + Accuracy: 0.51 (+/- 0.07)
  + PCA: [ 0.47530864 0.46583851 0.47826087 0.46875 0.45 0.60625 0.475 0.50625 0.44936709 0.42038217]
  + Accuracy: 0.50 (+/- 0.09)

1. Random Forest 5:

* Previous result:
  + Train accuracy: 0.9588918677390528
  + Test accuracy: 0.6041666666666666
* PCA result:
  + Train accuracy: 0.9597855227882037
  + Test accuracy: 0.5625
* CV prediction error rate(accuracy):
  + Previous: [ 0.48765432 0.47826087 0.53416149 0.49375 0.55625 0.6375 0.54375 0.5375 0.53164557 0.50318471]
  + Accuracy: 0.52 (+/- 0.07)
  + PCA: [ 0.51851852 0.49689441 0.48447205 0.48125 0.4875 0.5875 0.53125 0.5125 0.48101266 0.52866242]
  + Accuracy: 0.53 (+/- 0.07)

1. Adaboost 5:

* Previous result:
  + Train accuracy: 0.5576407506702413
  + Test accuracy: 0.5416666666666666
* PCA result:
  + Train accuracy: 0.5111706881143878
  + Test accuracy: 0.5291666666666667
* CV prediction error rate(accuracy):
  + Previous: [ 0.45679012 0.49689441 0.54658385 0.475 0.5375 0.65625 0.56875 0.525 0.53164557 0.50955414]
  + Accuracy: 0.53 (+/- 0.11)
  + PCA: [ 0.42592593 0.52795031 0.49068323 0.475 0.46875 0.55625 0.525 0.525 0.51265823 0.46496815]
  + Accuracy: 0.50 (+/- 0.07)

1. Gaussian Process (RBF kernel with 1-vs-1 classification):

* Previous result:
  + Train accuracy: 0.965
  + Test accuracy: 0.546
* PCA result:
  + Train accuracy: 0.793
  + Test accuracy: 0.606
* CV prediction error rate(accuracy):
  + Accuracy: 0.47 (+/- 0.06)
  + PCA Accuracy: 0.56 (+/- 0.12)

1. Kernel SVM (RBF kernel):

* Previous result:
  + Train accuracy: 0.7721179624664879
  + Test accuracy: 0.5854166666666667
* PCA result:
  + Train accuracy: 0.6613047363717605
  + Test accuracy: 0.6104166666666667
* CV prediction error rate(accuracy):
  + Accuracy: 0.50 (+/- 0.06)
  + PCA Accuracy: Accuracy: 0.59 (+/- 0.12)

For multinomial logistic regression, since general PCA is applicable in linear classifier/regression model, the results for training and test accuracy and accuracy from 10 fold cross validation evidently show the increase from the original version. However, in other models other than linear classifiers show increase in numbers in accuracies but we cannot definitively say that PCA was truly applicable here. For example, Gaussian process and Kernel SVM take kernel that makes linear feature values to nonlinear feature space show decrease in training accuracy and somewhat increment in test accuracy.

For nonlinear feature space usage such as kernels, Probabilistic PCA would be applicable.