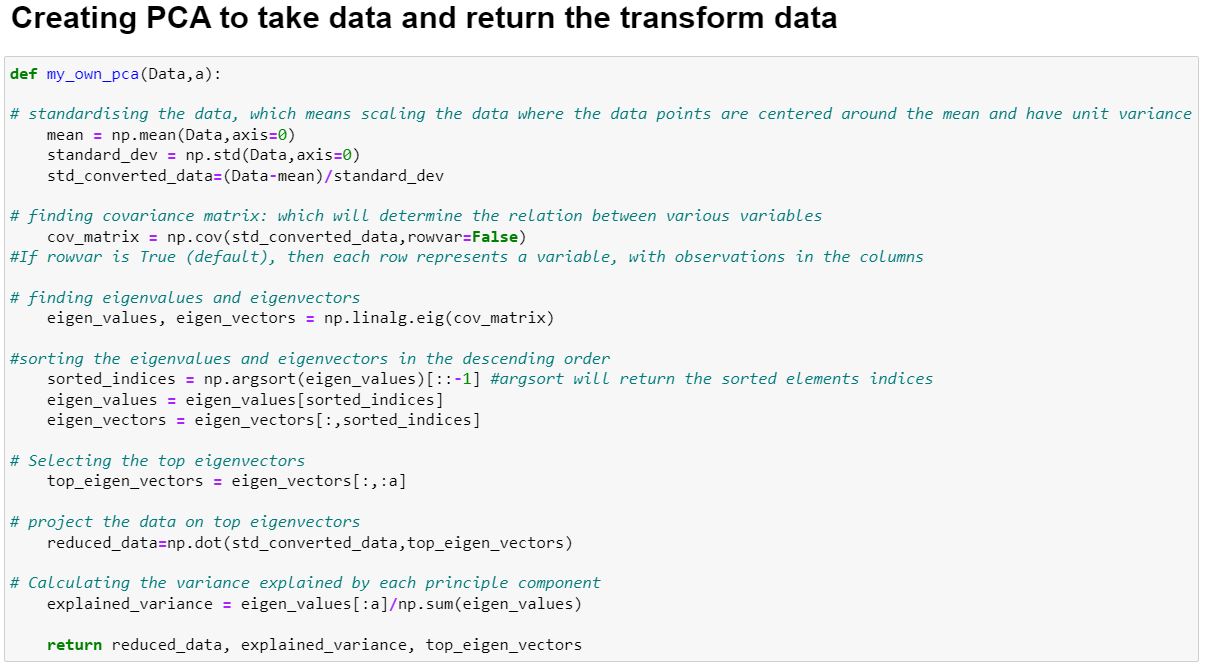
Report:

1. Principal Component Analysis (PCA)
   1. Implement PCA form scratch
2. Python code from scratch

***Importing supporting libraries*:**



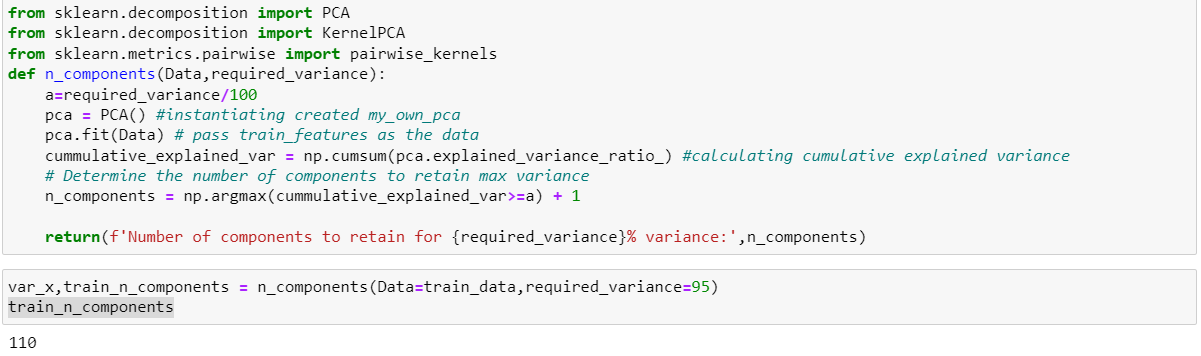
***This is the PCA function which will take data and number of principle components and return the transformed data***

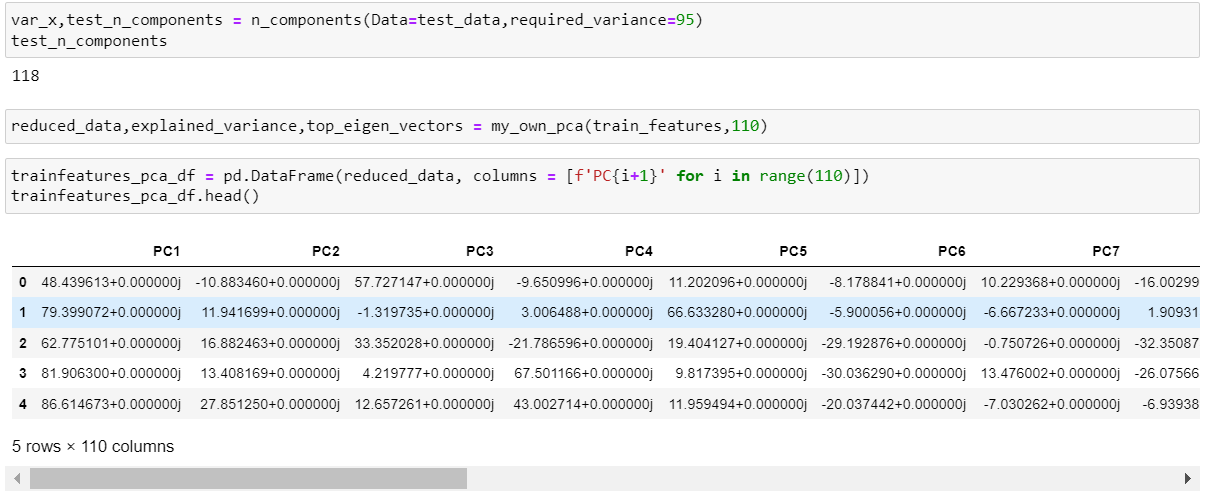


***Reading the test and train data excluding the last column:***

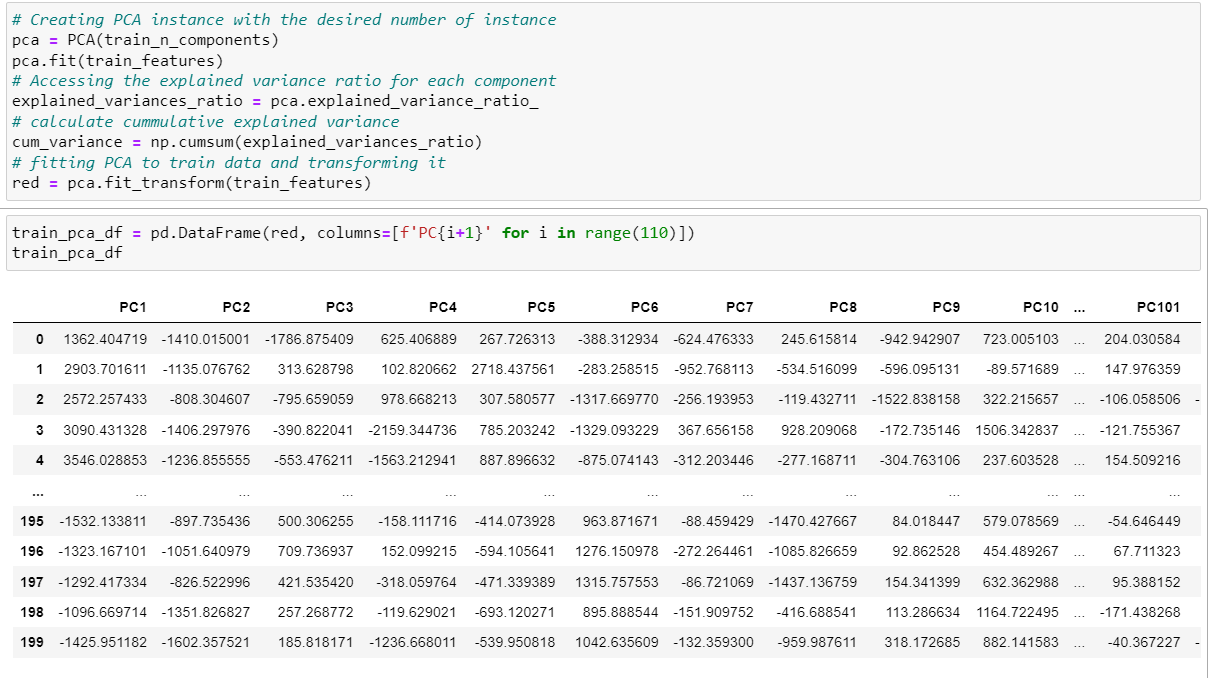
****

***Apply your PCA implementation to the Train dataset to reduce the dimensionality of the images and choosing significant amount of variance as 95%***

******

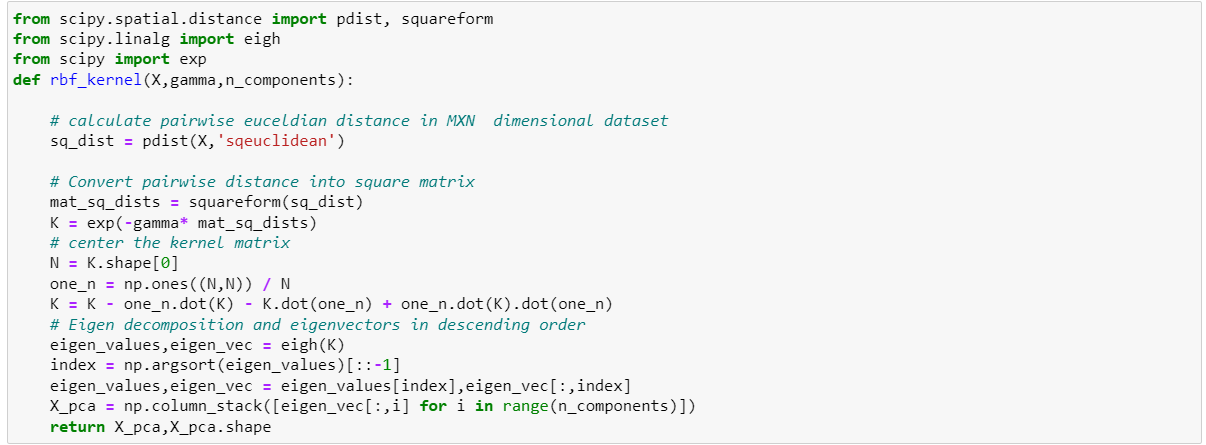
******

***Applying PCA with the help of Sklearn:***

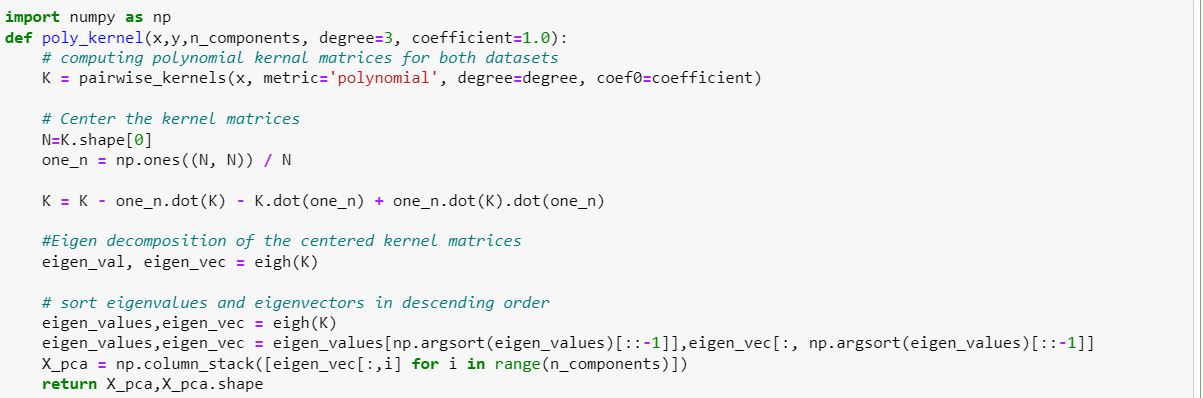
******

**Part 2:**

***KCPA with RBF Kernel:***

******

***KCPA with polynomial Kernel:***

******

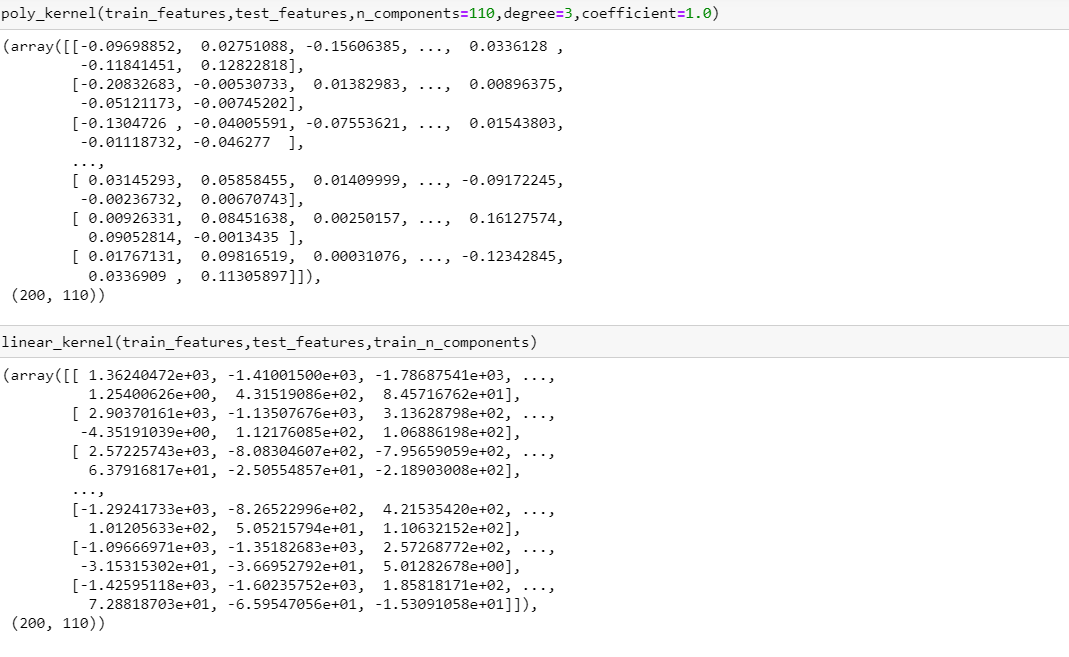
***KCPA with linear kernel:***

******

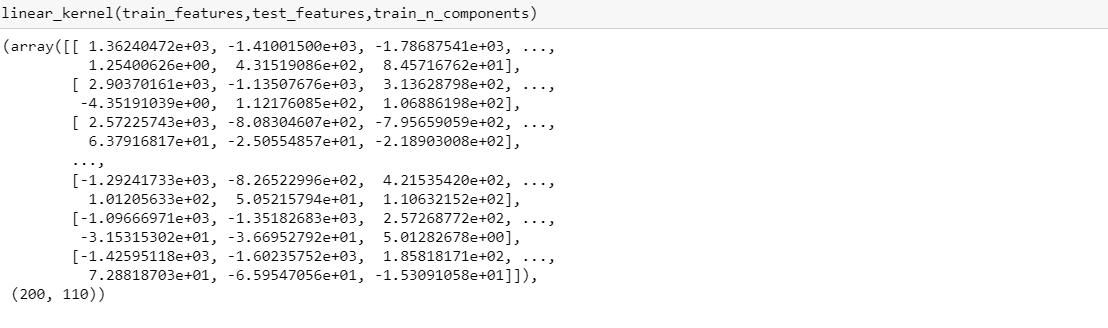
***Applying Radial Base Function on train dataset:***

******

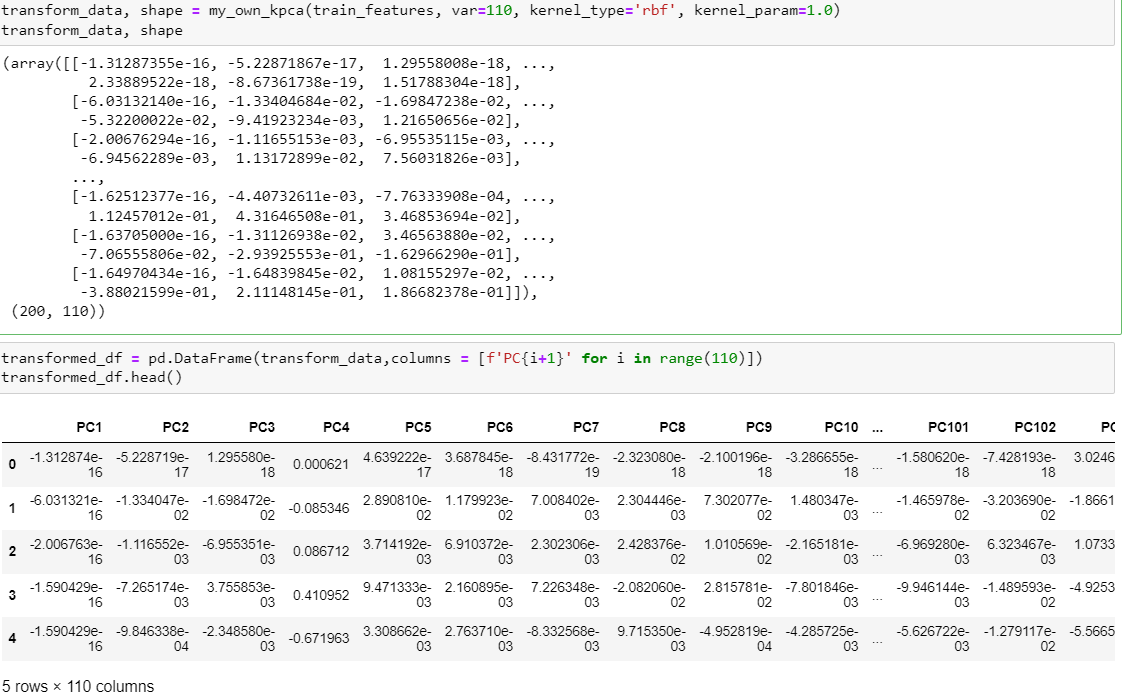
***Applying polynomial kernel on train dataset:***

******

***Finally applying liner kernel:***

******

***Applying My\_own\_kcpa on train\_data:***

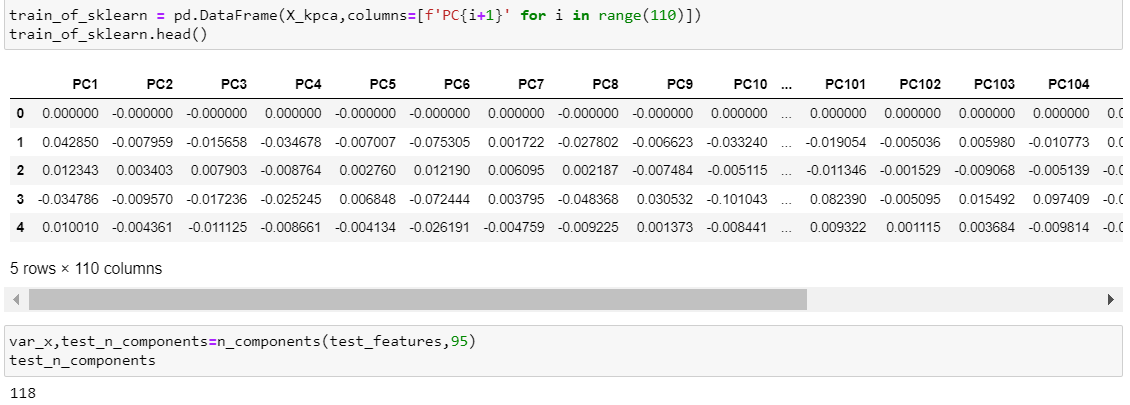
******

**Part 3: Testing and evaluation.**

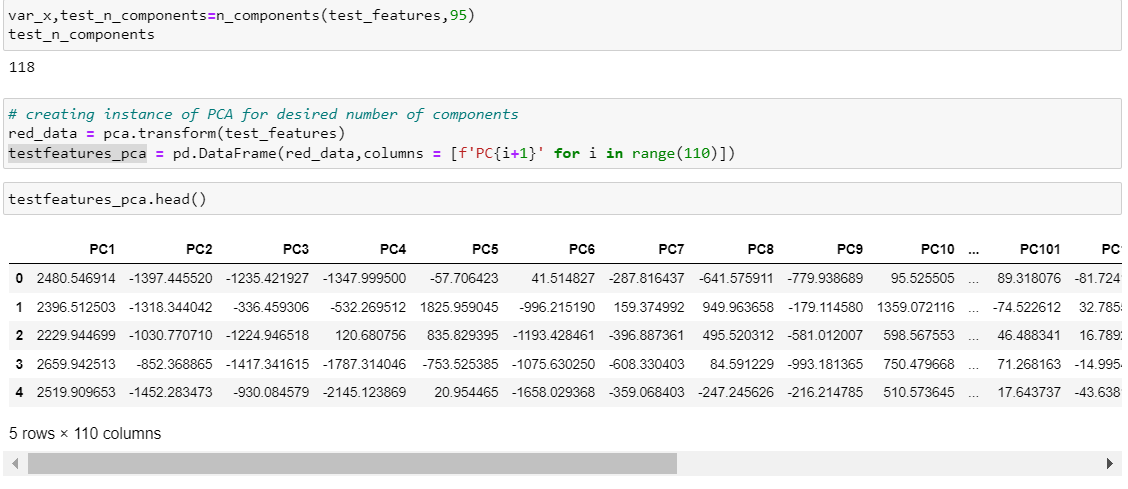
***Using sklearn on train and test data\_set:***

******

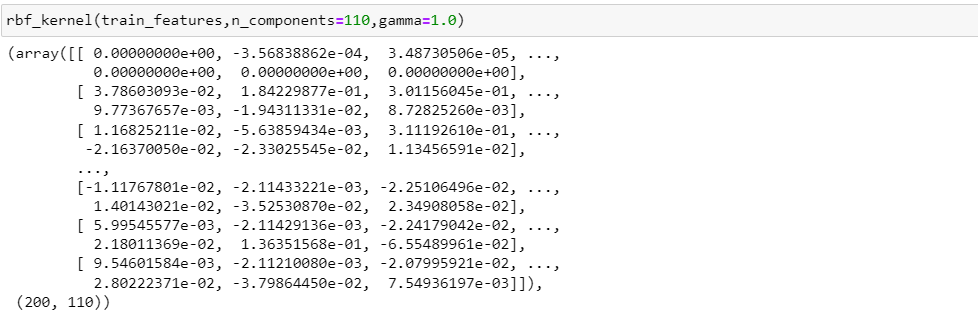
****

****

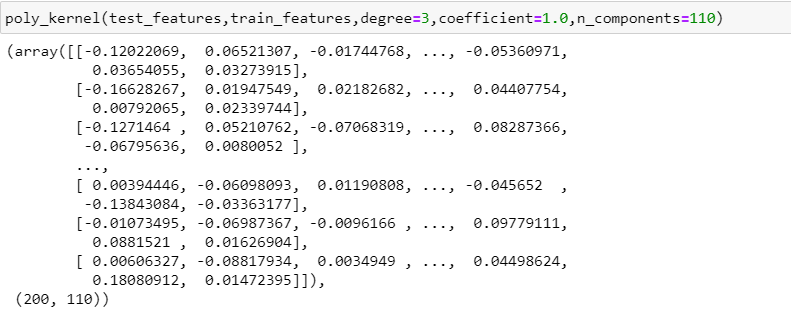
***Applying on test data set:***

******

***Applying RBF and polynomial kernel on test\_data:***

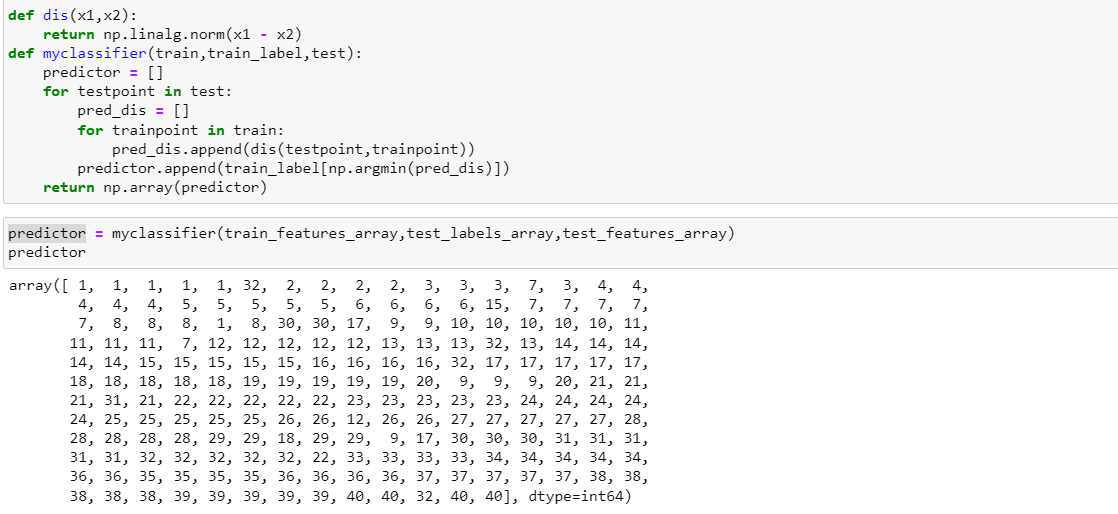
******

***Polynomial:***

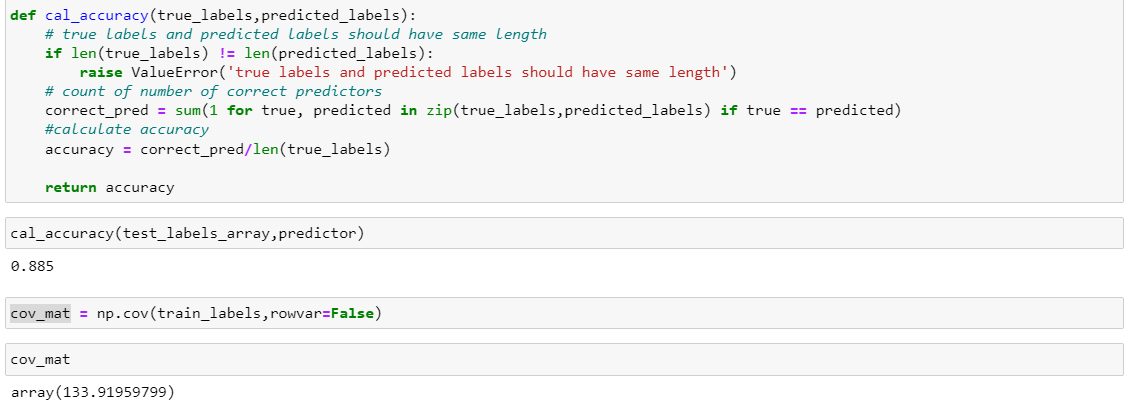
******

***Calculation of accuracy:***

***Creating classifier:***

******

***Function for accuracy:***

******