Principle Component Analysis on Student Performance Data

Shubham Barudwale
School of Computing Sciences and Engineering, VIT Chennai, Tamilnadu, India 600127
Email: barudwaleshubham.dinesh2015@vit.ac.in

Abstract-

Similar to feature selection, we can use feature extraction to reduce the number of features in a dataset. However, while we maintained the original features when we used feature selection algorithms, such as sequential backward selection, we use feature extraction to transform or project the data onto a new feature space. This reduces the dimentionality of data and canproduce the efficient results in many cases

Using Priciple component analysis the data dimentinality can be reduced and we get better results. But in many cases the results can be worse.

1. Introduction, Algorithm and Methodology

Principal component analysis (PCA) is an unsupervised linear transformation technique that is widely used across different fields, most prominently for dimensionality reduction. Other popular applications of PCA include exploratory data analyses and de-noising of signals in stock market trading, and the analysis genome data and gene expression levels in the field of bioinformatics. PCA helps us to identify patterns in data based on the correlation between features. In a nutshell, PCA aims to find the directions of maximum variance in high-dimensional data and projects it onto a new subspace with equal or fewer dimensions that the original one. The orthogonal axes (principal components) of the new subspace can be interpreted

the approach in a few simple steps:

1. Standardize the d -dimensional dataset.

2. Construct the covariance matrix.

3. Decompose the covariance matrix into its eigenvectors and eigenvalues.

4. Select k eigenvectors that correspond to the k largest eigenvalues, where k is the dimensionality of the new feature subspace (k d ≤).

5. Construct a projection matrix W from the "top" k eigenvectors.

6. Transform the d -dimensional input dataset X using the projection matrix W to obtain the new k -dimensional feature subspace.

If we use PCA for dimensionality reduction, we construct a d k × -dimensional transformation matrix W that allows us to map a sample vector x onto a new k -dimensional feature subspace that has fewer dimensions than the original d -dimensional feature space.

As a result of transforming the original d -dimensional data onto this new k -dimensional subspace (typically k d <<), the first principal component will have the largest possible variance, and all consequent principal components will have the largest possible variance given that they are uncorrelated (orthogonal) to the other principal components. Note that the PCA directions are highly sensitive to data scaling, and we need to standardize the features prior to PCA if the features were measured on different scales and we want to assign equal importance to all features.

3. Data

Student performance dataset is the dataset from which we are trying to gain some information from 650 student’s academic performance data and various attributes which are affecting the performance. Attributes include sex, age, study time, free time, family information etc.

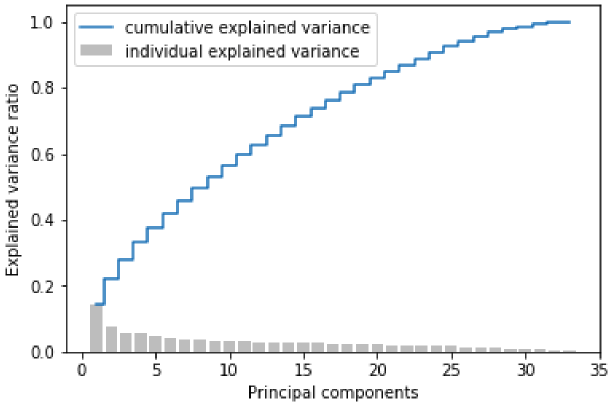
We will be using the Logistic regerssion method to learn and predict the performance. Out of 650 I have taken 455 data entries for lerning and rest for prediction i.e testing the performance of the algorithm.(70:30). The labels or target functions are mentioned in the ‘new’ field which is added manually. bad(0-7), medium(8-14), good(15-20) .

5. Experiments

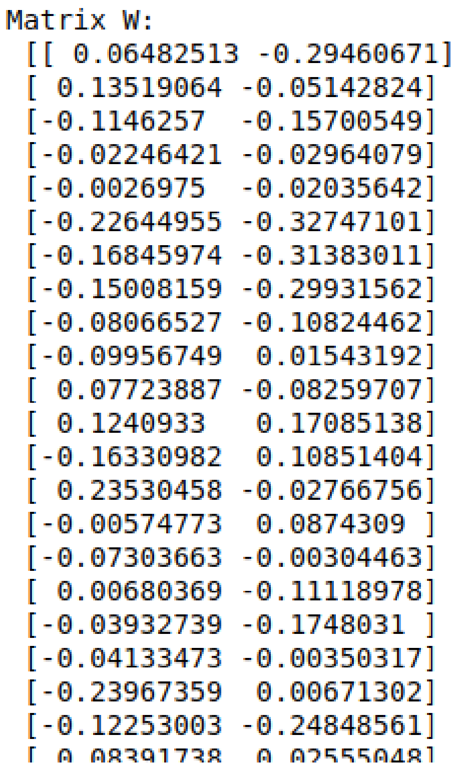
After completing the mandatory preprocessing steps by executing the preceding code, let's advance to the second step: constructing the covariance matrix. The symmetric d d × -dimensional covariance matrix, where d is the number of dimensions in the dataset, stores the pairwise covariances between the different features.

$$\sigma_{jk} = \frac{1}{n}\sum_{i=1}^{n}\left(x_j^{(i)} - \mu_j\right)\left(x_k^{(i)} - \mu_k\right)$$

Here, µ j and µk are the sample means of feature and k , respectively. Note that the sample means are zero if we standardize the dataset. A positive covariance between two features indicates that the features increase or decrease together, whereas a negative covariance indicates that the features vary in opposite directions



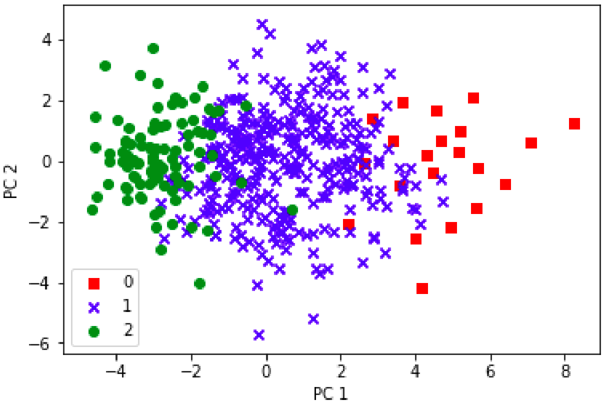
Since we want to reduce the dimensionality of our dataset by compressing it onto a new feature subspace, we only select the subset of the eigenvectors (principal components) that contains most of the information (variance). Since the eigenvalues define the magnitude of the eigenvectors, we have to sort the eigenvalues by decreasing magnitude; we are interested in the top k eigenvectors based on the values of their corresponding eigenvalues.



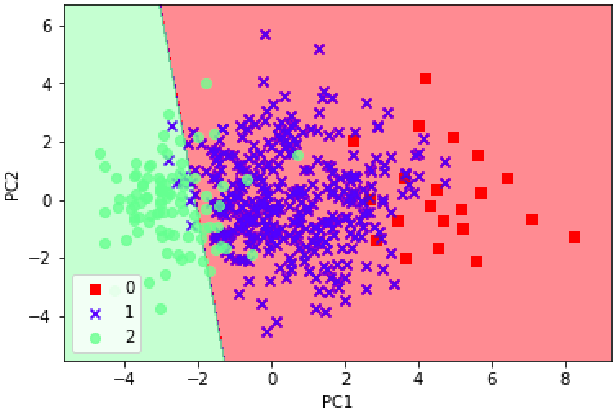
```
Matrix W:
 [[ 0.06482513 -0.29460671]
 [ 0.13519064 -0.05142824]
 [-0.1146257  -0.15700549]
 [-0.02246421 -0.02964079]
 [-0.0026975  -0.02035642]
 [-0.22644955 -0.32747101]
 [-0.16845974 -0.31383011]
 [-0.15008159 -0.29931562]
 [-0.08066527 -0.10824462]
 [-0.09956749  0.01543192]
 [ 0.07723887 -0.08259707]
 [ 0.1240933   0.17085138]
 [-0.16330982  0.10851404]
 [ 0.23530458 -0.02766756]
 [-0.00574773  0.0874309 ]
 [-0.07303663 -0.00304463]
 [ 0.00680369 -0.11118978]
 [-0.03932739 -0.1748031 ]
 [-0.04133473 -0.00350317]
 [-0.23967359  0.00671302]
 [-0.12253003 -0.24848561]
 [ 0.08391738  0.02555048]
```

Using these eigen values we calculate the principle components and we find the decision boundary based on the new data.

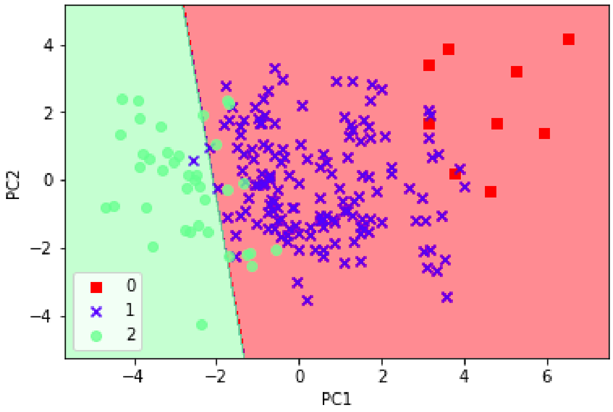
Using Scikit learn we can do PCA but in my case as PCA is unsuperwised learning and the data was overlapping the PCA could not divide the data correctly.



on train data:



for test data:



6. Conclusion

The data was not very distinguishabe and as the data was descrete the PCA was not working fine on the Student dataset and could not give the proper division on the data.

Scikit learn algorithm was also giving the same results.

7. Referances

1) Sklearn PCA.

2) Python_Machine_Learning Sebastian Raschka