[AIML2015] Homework 1: PCA and Naive Bayes Classification

Francesco Cucari - 1535743

December 19, 2015

1 Dataset Description

The dataset is composed of 216 images chosen among the about 7000 images contained in the COIL-100 dataset. I chose 3 objects among the 100 available: obj4, obj10 and obj28. For each object (*class*) I read 72 images.

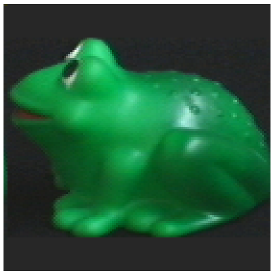


Figure 1: Obj28

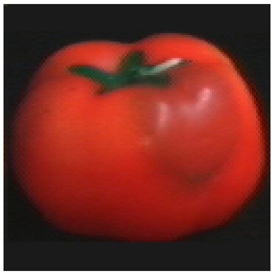


Figure 2: Obj4

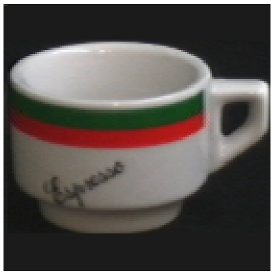
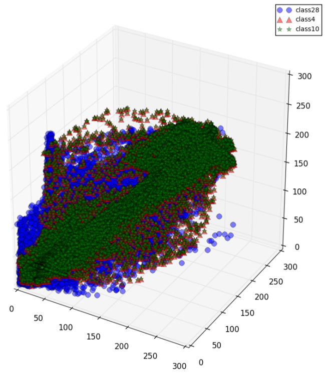


Figure 3: Obj10



My dataset is stored in form of a 216x49152 matrix where the columns are the different *features*, and every row represents a separate object *example*. Plotting the dataset, I can state that the problem of multi-dimensional data is its visualization. As the dimensionality of the data increases, many types of data analysis and classification problems become significantly harder. Sometimes the data also becomes increasingly sparse in the space it occupies. This phenomenon is called "*the curse of dimensionality*". So, the main purpose of a Principal Component Analysis (PCA) is the analysis of data to identify and finding patterns to reduce the dimensions of the dataset with minimal loss of information.

The desired outcome of the PCA is to project a feature space (my dataset) onto a smaller subspace that represents data in a good way.

2 Principal Component Visualization

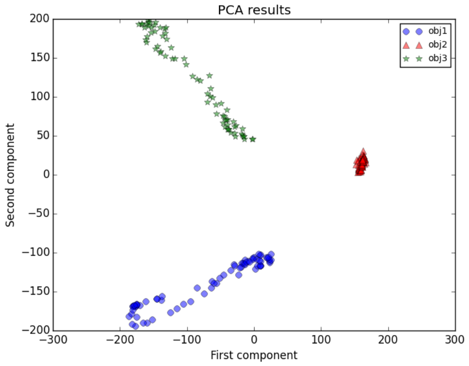


Figure 4: First-Second components

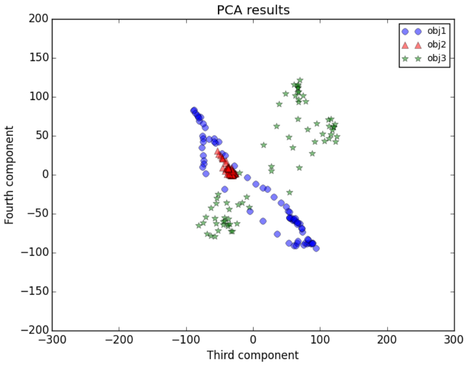


Figure 5: Third-Fourth components

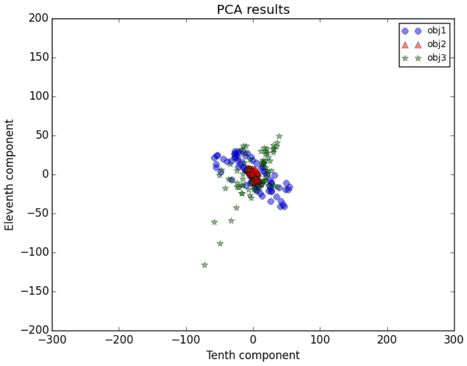


Figure 6: Tenth-Eleventh components

In order to reduce the dimensionality of our feature space, I project the feature space via PCA onto a smaller subspace, where the eigenvectors will form the axes of this new feature subspace.
I can notice that it's easier visualize data on a lower dimensional space and the data of each classes don't overlap if I choose the proper components.

In order to decide the number of principal components needed to preserve data without much distortion, I must analyze the corresponding eigenvalues of the eigenvectors. In fact the eigenvalues represent the "magnitude" of the eigenvectors. I'm interested in those eigenvectors with the much larger eigenvalues, since they contain more information about data.

An useful measure is the so-called "*explained variance*" which can be calculated from the eigenvalues. The explained variance tells us how much information (variance) can be attributed to each of the principal components. So, the parameter "*explained_variance_ratio_*" returns the percentage of variance explained by each of the selected components. In my analysis I have these values:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11
32.4	26.2	7.2	5.8	4	2.3	1.8	1.5	1.4	1.0	0.9

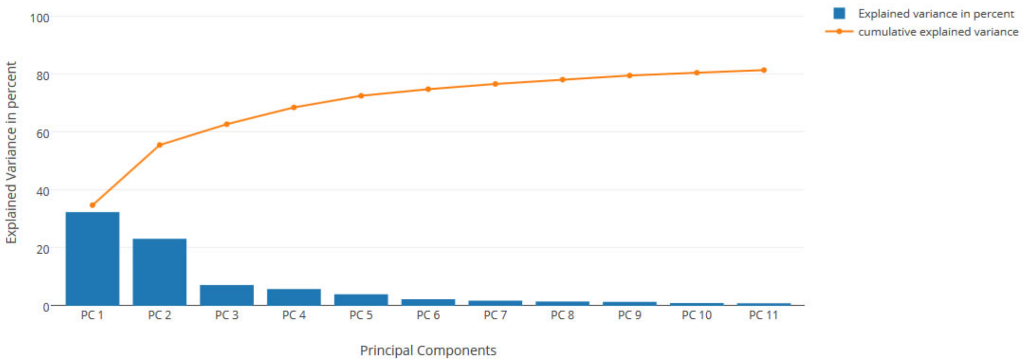


Figure 7: Cumulative Explained Variance

In conclusion, with regard to the table and the Fig.7, the first two principal components contain 58.6% of the information, while the other principal components can be dropped without losing too much information.

3 Classification

The Naive Bayes classifier is:

$$
\begin{aligned}
\hat{y} &= \arg\max_{y\in(1,2,3)} P(y|x_1, ..., x_{49152}) \\
&= \arg\max_{y\in(1,2,3)} \frac{P(x_1, ..., x_{49152}|y)P(y)}{P(x_1, ..., x_49152)} \\
&= \arg\max_{y\in(1,2,3)} P(x_1, ..., x_{49152}|y)P(y) \\
&= \arg\max_{y\in(1,2,3)} P(y) \prod_{i=1}^{49152} P(x_i|y) \\
&= \arg\max_{y\in(1,2,3)} \frac{1}{3} \prod_{i=1}^{49152} \frac{1}{\sqrt{2\pi\sigma_y^2}} e^{\frac{-(x_i-\mu_y)^2}{2\sigma_y^2}}
\end{aligned}
$$

Where I consider $P(x|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} e^{\frac{-(x-\mu_y)^2}{2\sigma_y^2}}$ as a Gaussian and the distribution of labels as uniform ($P(y = i) = \frac{1}{3}, i = 1, 2, 3$).

After the splitting of all data, the training and the testing of Naive Bayes classifier with Gaussian class-conditional distribution, I compute the *score()* function which returns the mean accuracy on the given test data and labels. The obtained score is: 1.0. Then, I repeat the splitting, the training and the testing for the data projected onto first two principal components and onto the third and fourth. The former returns 1.0 and the latter 0.78.

Another useful measure for evaluating the quality of the output of a classifier on the dataset is the *confusion matrix*. The diagonal elements represent the number of points for which the predicted label is equal to the true label, while off-diagonal elements are those that are mislabeled by the classifier. The higher the diagonal values of the confusion matrix the better, indicating many correct predictions.

So, I can state that the accuracy for first two principal components is the same of the all samples. This is a relevant result, because as seen before the most of variance is explained by first two components and thus increasing the number of examples and features, the accuracy of the first two-components is approximately the same as all the examples. On the contrary, the accuracy of the other components is lower than the first two components. A high accuracy indicates many correct predictions: as you can see, in the Fig.8 and in the Fig.9 all predicted labels are equal to the true labels. This is not true considering other components, in fact there are some mislabeled predictions. The Tab.1 shows the normalized values of confusion matrix and the Fig.10 shows it in a graphic way.

	obj28	obj4	obj10
obj28	0.74	0.26	0.00
obj4	0.32	0.64	0.04
obj10	0.04	0.00	0.96

Table 1: Confusion matrix for third and fourh components

You can see these mislabeled predictions in the Fig.12, where the predicted labels are plotted and the mislabeled prections are marked.

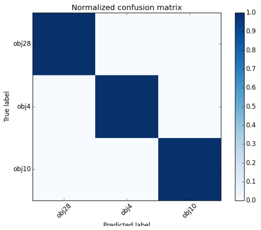


Figure 8: Confusion matrix for all samples

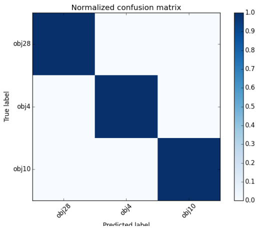


Figure 9: Confusion matrix for first and second components

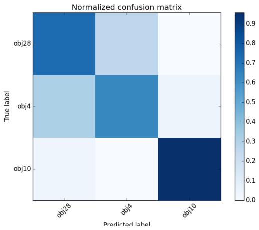


Figure 10: Confusion matrix for third and fourth components

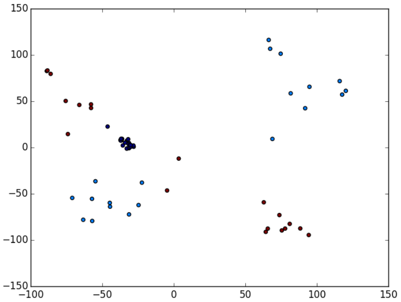


Figure 11: Plot of x-test

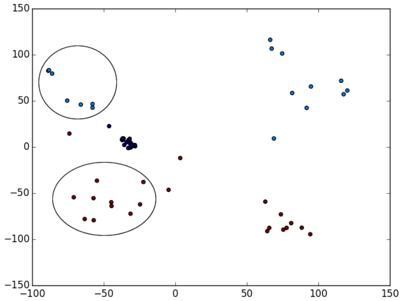


Figure 12: Plot of predicted x compared to x-test

3.1 Decision Boundaries

The high value of accuracy is related to the fact the three chosen classes are easily separable if I consider the first two principal components, as you can see in the Fig.13. While, if I consider the third and fourth components, the decision boundaries are not well defined (Fig.14) because points of the three classes occupy the same region.

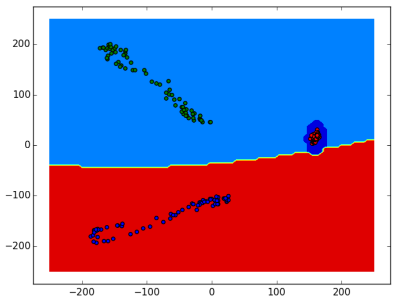


Figure 13: Decision boundaries for first and second principal components

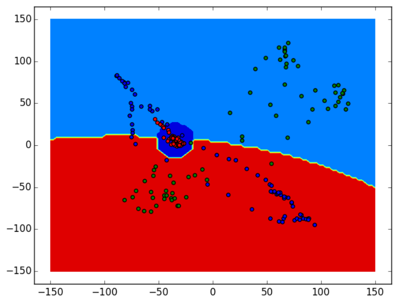


Figure 14: Decision boundaries for third and fourth principal components