# Chapter 4 [Dimensionality Reduction]

Usually, the databases used in data mining consists of millions of records and thousands of variables. It is unlikely that all of the variables are independent, with no correlation structure among them. In this scenario, the problem of *multicollinearity* arises. It is a condition where some of the predictor variables are strongly correlated with each other. It may lead to instability in the solution space and consequently erroneous inferences. In model building exercises practitioners always follow the principle of parsimony i.e. to explain the behaviour of the output variable with as lesser number of explanatory variables as possible. This makes the process of interpretation easier. So, the use of too many predictor variables to model a relationship with a response variable can unnecessarily complicate the interpretation of the analysis. Also, retaining too many variables may lead to *overfitting*, and under such circumstances prediction of future observations becomes faulty. So, it becomes clear that by decreasing the number of predictors to an optimum count we can achieve a significant level of accuracy in the setup.

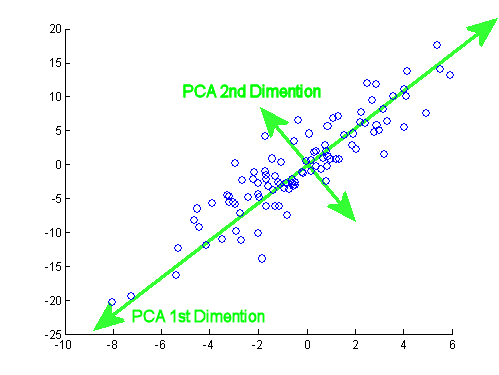
There are several methods by which this reduction in dimension can be carried out. Some of the well-appreciated methods are; Principal components analysis (PCA), Factor analysis, Backward Feature Elimination, Decision Trees and Random Forest, User-defined composites etc.

Let’s set up a specific example to illustrate how PCA works. Assume that you have a database of emails and you want to classify (using some machine learning numerical algorithm) each email as spam/not spam. To achieve this goal, you construct a mathematical representation of each email as a bag-of-words vector. This is a binary vector, where each position corresponds to a specific word from an alphabet. For an email, each entry in the bag-of-words vector is the number of times a corresponding word appears in an email (0 if it does not appear at all).

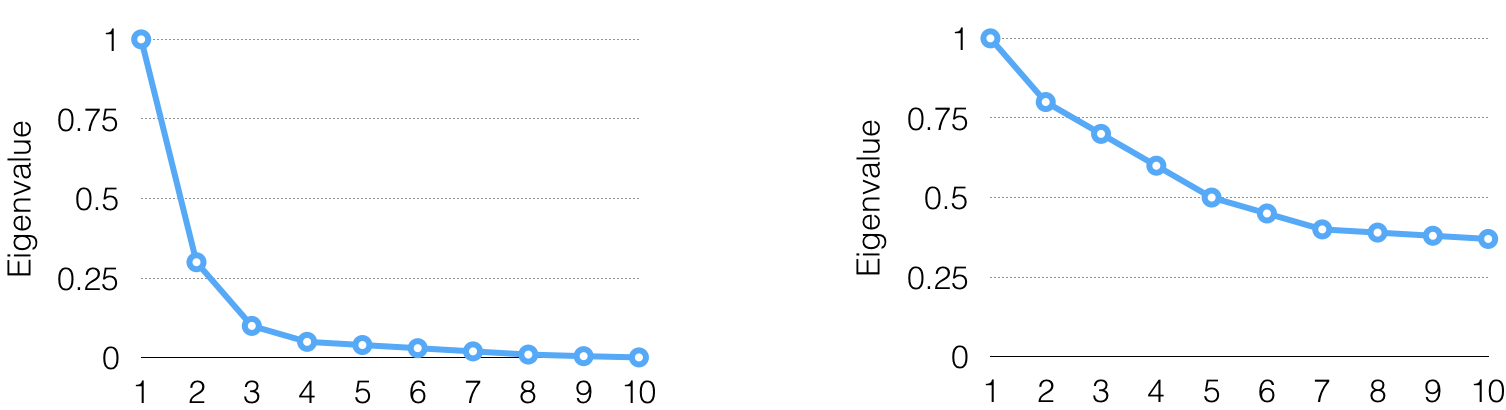
Assume you have constructed a bag-of-words from each email, and as a result you have a sample of bag-of-words vectors x1…. xm. However, not all dimensions (words) of your vectors are informative for the spam/not spam classification. For instance, words “lottery”, “credit”, “pay” would be better features for spam classification than “dog”, “cat”, “tree”. For a mathematical way to reduce dimension we will use PCA.

For PCA you should construct an m-by-m covariance matrix from your sample x1…. xm and compute its eigenvectors and eigenvalues. Next sort the resulting numbers in a decreasing order and choose p top eigenvalues. Applying PCA to your sample of vectors is projecting them onto eigenvectors corresponding to top p eigenvalues. Now, your output data is the projection of original data onto p eigenvectors, the dimension of projected data has been reduced to p.

A reader might wonder, what is special about projecting bag-of-word vectors onto the top eigenvectors of covariance matrix? How does it help to extract the most informative part of original data? This is illustrated on a 2-dimensional picture below, where the blue points are 2-dimensional (for simplicity) observations.



The eigenvectors of covariance matrix have a special property that they point towards the directions of the most variance within the data. As you can see on the picture, the 1st dimension vector points towards the direction of the highest variance and the 2nd dimension vector points towards the highest variance in the subspace, orthogonal to the 1st vector. Thus, projecting onto top eigenvectors preserves maximum variance, and roughly speaking, capturing more variance means capturing more information to analyze.



Graph#1: Exponential Decay 3                                                        Graph#2: Exponential Decay 7

Another question is how to choose the number of top eigenvectors to project on? According to my experience, a good way to choose it to plot the eigenvalues and find the point on the plot, where the eigenvalues start to decay exponentially. The eigenvalue plots for two different datasets (left and right) are illustrated on the charts above. On the left chart the point of exponential decay is 3, and on the right chart it is 7, which means that one should select 3 top eigenvalues for the left dataset and 7 for the right one. Also, you should think out of the box and see if PCA in itself is an appropriate method for your problem. With the example above, the left plot of eigenvalues shows a fast exponential decay, thus PCA is great for that problem. However, the eigenvalues on the right decay is almost linear, so PCA is not recommended.

Finally, after you have computed the low-dimensional PCA projection of your bag-of-words vectors, you can use this projection instead of original emails in classification algorithms, such as Logistic Regression or Support Vector Machine to classify the emails as spam/not spam. When projections are used instead of original emails, algorithm training will be much faster and overfitting will be reduced.