PCA Whitening

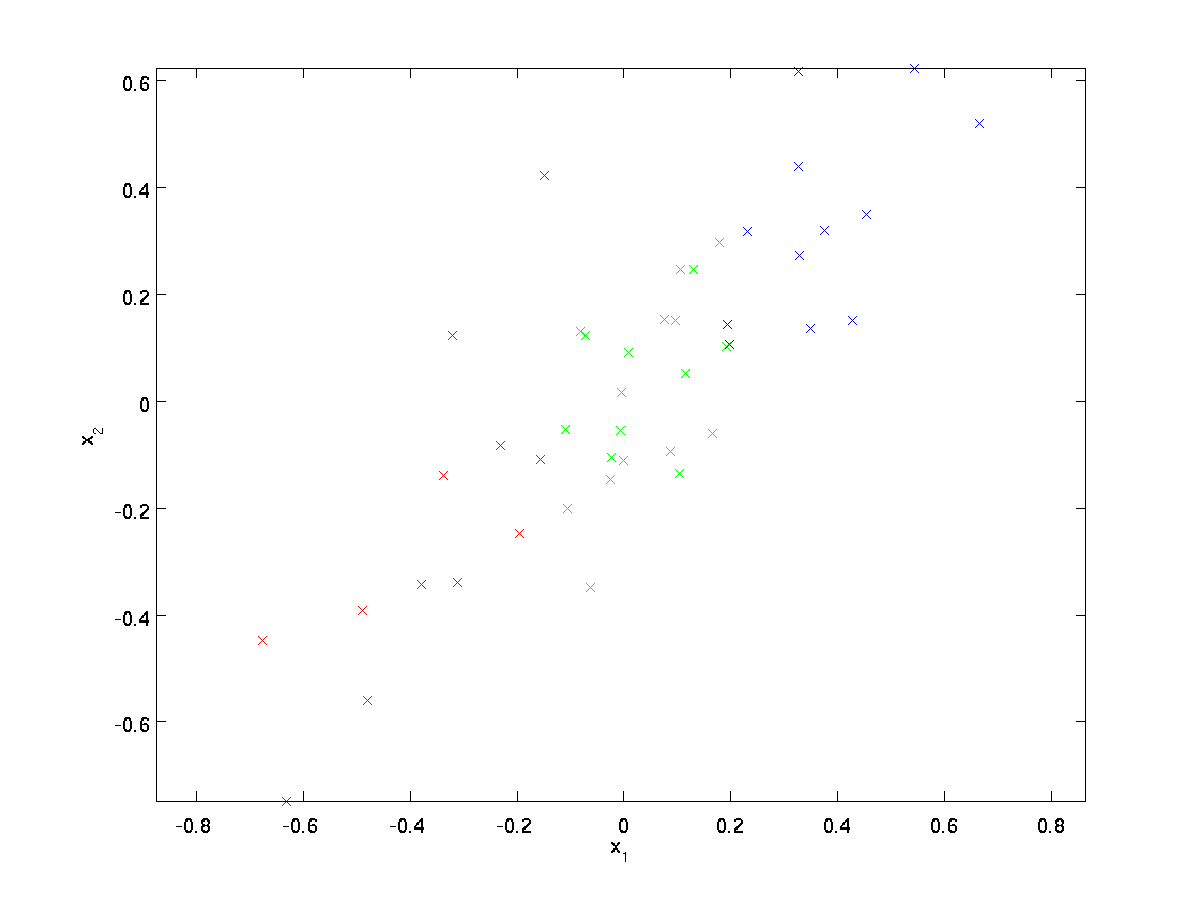
**1.Introduction**

Principal Components Analysis (PCA) is a dimensionality reduction algorithm that can be used to significantly speed up your unsupervised feature learning algorithm. More importantly, understanding PCA will enable us to later implement **whitening**, which is an important pre-processing step for many algorithms.

Suppose you are training your algorithm on images. Then the input will be somewhat redundant, because the values of adjacent pixels in an image are highly correlated. Concretely, suppose we are training on 16x16 grayscale image patches. Then are 256 dimensional vectors, with one feature corresponding to the intensity of each pixel. Because of the correlation between adjacent pixels, PCA will allow us to approximate the input with a much lower dimensional one, while incurring very little error.

**2. Example and Mathematical Background**

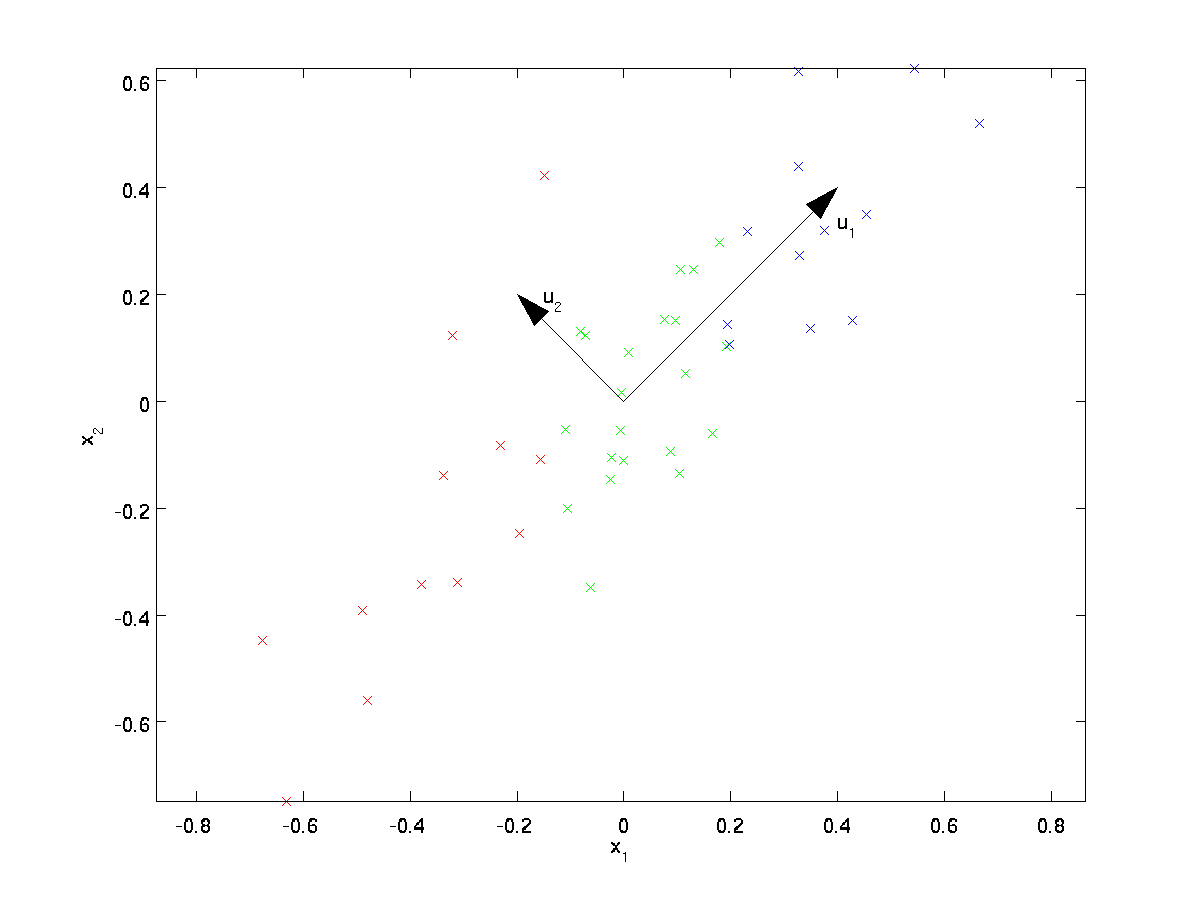
For our running example, we will use a dataset  with n=2 dimensional inputs, so that . Suppose we want to reduce the data from 2 dimensions to 1. (In practice, we might want to reduce data from 256 to 50 dimensions, say; but using lower dimensional data in our example allows us to visualize the algorithms better.) Here is our dataset:



This data has already been pre-processed so that each of the features x1 and x2 have about the same mean (zero) and variance.

For the purpose of illustration, we have also colored each of the points one of three colors, depending on their x1 value; these colors are not used by the algorithm, and are for illustration only.

PCA will find a lower-dimensional subspace onto which to project our data.  
From visually examining the data, it appears that u1 is the principal direction of variation of the data, and u2 the secondary direction of variation:



I.e., the data varies much more in the direction u1 than u2. To more formally find the directions u1 and u2, we first compute the matrix Σ as follows:

