Principal Component Analysis

Justin Kim

November 28, 2016

1 Introduction

Many times, a large number of measurements or observations can be made for each sample, thereby leading to a large number of dimensions, sometimes more than the number of samples. This is especially applicable in the life sciences as thousands of mRNA and protein level measurements can be made for every sample. Visualizing such high-dimensional data can be difficult, so a way to reduce the number of dimensions without losing much information would be useful.

1.1 Principal Component Analysis

Principal Component Analysis (PCA) is a statistical method that does just that by identifying specific linear combinations of variables that would retain as much variation in the data as possible. These linear combinations, or principal components, are the new dimensions for the data. It is important to note that as a result, they are, by definition, orthogonal to one another. It should be noted this method relies on the assumption that some dimensions account for more of the data's variation than others do and that the data actually can be characterized by linear combinations. It is interesting to note that the number of dimensions can be reduced to at most the number of samples without losing any information.

The motivation behind maximizing the variation in the data can be demonstrated by measuring the signal-to-noise ratio (SNR), which is the ratio of the variance in signal to the variance in noise. The higher the SNR, the more precise the data, and the lower the SNR, the more contaminated the data. The assumption is that by maximizing variance, as much variance from the signal will be retained, so the SNR will remain high.

PCA is essentially a change of bases. The original set of bases were the dimensions, and the new set are the principal components. As Singular Value Decomposition (SVD) also represents a change of bases, it is no coincidence the two are closely related. In fact, the two terms are sometimes even used interchangeably. Thus, SVD will be used in both the derivation and implementation of PCA here.

2 The derivation of PCA

First, the equation behind SVD must be derived since it is so similar to that of PCA.
Let matrix X be defined as an m x n matrix in which each row is a vector of measurements of a particular type with zero mean. Thus, each column of X represents a set of measurements from one particular trial. The covariance matrix Sx can be defined as:

$$Sx = \frac{1}{n-1} X X^T$$

A couple interesting properties are that Sx is a symmetric m x m matrix (the product of any matrix and its transpose is a square, symmetric matrix), the diagonal terms of Sx are the variances of each measurement type, and the off-diagonal terms are the covariances between measurement types. It should also be noted that an unbiased estimator for the sample variances is calculated as all terms are divided by (n - 1) rather than n. In order to reduce redundancies as much as possible, it would be useful to be able to transform matrix X into a matrix Y with zero covariances.

2.1 Describing the Principal Component Matrix

More precisely, the goal is to have an orthonormal matrix P such that $Y = PX$, where $Sy = \frac{1}{n-1}YY^T$ is diagonalized. The rows of P will be the principal components of X.
This definition of Sy can be rewritten as

$$Sy = \frac{1}{n-1}YY^T = \frac{1}{n-1}(PX)(PX)^T = \frac{1}{n-1}PXX^TP^T$$

Since XX^T is a symmetric matrix, it can be rewritten as $XX^T = EDE^T$, where E is a matrix of XX^T's eigenvectors as columns, and D is a diagonal matrix. Let matrix P be defined such that each row p_i is an eigenvector of XX^T, in which case, $P = E^T$, making $XX^T = P^TDP$. This can be substituted into the above equation to be

$$Sy = \frac{1}{n-1}PXX^TP^T = \frac{1}{n-1}PP^TDPP^T = \frac{1}{n-1}PP^{-1}DPP^{-1} = \frac{1}{n-1}D$$

since the transpose of an orthogonal matrix is equal to its inverse (meaning $P^T = P^{-1}$). This definition of P diagonalizes Sy, so it satisfies our needs. This value of P also means that the principal components of X are the rows of P or the eigenvectors of XX^T.

2.2 Using SVD

As mentioned earlier, SVD will be used. Let X^TX be a symmetric m x m matrix with rank r. Let $\{v_1, v_2, ..., v_r\}$ be the set of orthonormal n x 1 eigenvectors with associated eigenvalues $\{\lambda_1, \lambda_2, ..., \lambda_r\}$. Thus, $Sxv_i = \lambda_i v_i$ is true. Let σ_i be positive and real and be the singular values such that $\sigma_i = \sqrt{\lambda_i}$. Finally, let $\{u_1, u_2, ..., u_r\}$ be the set of orthonormal m x 1 vectors defined by $u_i = \frac{1}{\sigma_i}Xv_i$. This can be reorganized as $u_i\sigma_i = Xv_i$ to result in the value version of SVD.
At this point, a new diagonal matrix Σ can be constructed, in which $\Sigma_{ii} = \sigma_i$, where the elements of σ have been rank-ordered such that $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r$. Likewise, orthogonal matrices V and U can be constructed as $V = \left[\begin{array}{cccc} v_1 & v_2 & ... & v_n \end{array}\right]$ and $U = \left[\begin{array}{cccc} u_1 & u_2 & ... & u_m \end{array}\right]$, where (n - r) and (m - r) orthonormal vectors were appended to V and U respectively. It should be noted that these additional vectors should have no effect on the final solution since variations associated with them should be zero.
Thus, $u_i\sigma_i = Xv_i$ can be rewritten as $XV = U\Sigma$ when considering all dimensions at once, and since V is orthogonal, $V^{-1} = V^T$, this can be rewritten to be $X = U\Sigma V^T$. This means that any matrix can be expressed as the product of an orthogonal matrix, a diagonal matrix, and another orthogonal matrix. When calculating the SVD of a matrix, these are the three outputted matrices. One interesting observation to note is that U and V span all possible

inputs and outputs.
So, let us take the m x n matrix X and a new n x m matrix Y, such that $Y = \frac{1}{\sqrt{n-1}}X^T$, where each column of Y has a mean of zero. When considering Y^TY,

$$Y^TY = (\frac{1}{\sqrt{n-1}}X^T)^T(\frac{1}{\sqrt{n-1}}X^T) = \frac{1}{n-1}X^{TT}X^T = \frac{1}{n-1}XX^T = Sx$$

It should be noted that the columns of matrix V from the SVD of Y are the eigenvectors of Sx, and as shown above, the principal components of X are the eigenvectors of Sx. Thus, the principal components of X are the columns of V.

3 Implementation

```
function [signals, pc, vars] = pca(data)
    % get dimensions of dataset
    [m, n] = size(data);
    % get mean of each row of dataset
    rowMeans = mean(data, 2);
    % subtract mean from each element of dataset to normalize it
    normData = data - repmat(rowMeans, 1, n);

    % construct new matrix of which svd will be calculated
    newMatrix = normData’ / sqrt(n - 1);

    % calculate svd of new matrix
    [u, s, pc] = svd(newMatrix);

    % calculate the variances
    s = diag(s);
    vars = s .* s;

    % project the original data
    signals = pc’ * normData;
end
```

4 Comparison to Least-Squares Regression

The main difference between PCA and Ordinary Least Squares (OLS) regression is that PCA essentially minimizes the orthogonal distance between the data point and the principal component while OLS minimizes the vertical distance. However, it should be noted that these two are different in purpose. The principal component is not necessarily meant to be a model for the data. It is mainly meant to transform the bases of the data to a new one that retains the variation in the data as much as possible while reducing the data’s dimensionality. In fact, OLS can be run on the principal components once PCA is performed.

5 Analysis of Dataset 1

This dataset can be found here[1].

There are 27,648 dimensions with 105 samples. More specifically, expression levels of 27,648 genes are measured in 105 breast tumor samples. A paper describing PCA uses this dataset as an example [2].

There were a lot of NaN values throughout the dataset, and since SVD does not work with those invalid values, the data had to be cleaned. While these NaN values would have ideally been evaluated on a row by row basis (i.e. for each gene), all values of NaN just set to 0. One area for improvement for this analysis would be to consider other default values for each gene.

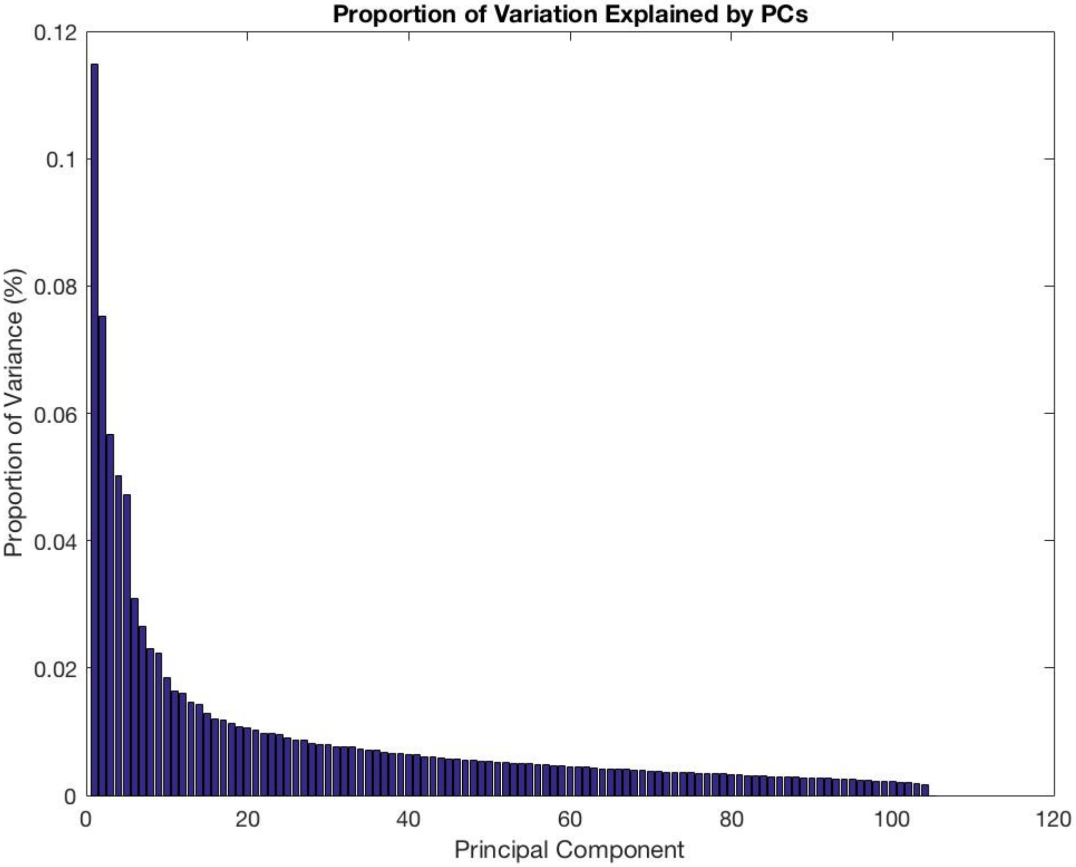

Figure 1: Proportion of Variation Explained by Principal Components

This distribution of the proportion of variation explained by each principal component seems reasonable since the first few principal components typically disproportionately represent the most variation. This is motivated by the idea each subsequent principal component has the extra constraint of being orthogonal to every preceding principal component, so it would not be able to capture as much variation. It should be noted that this distribution is similar to that shown in the paper. Unfortunately, the principal components for the specific genes examined in the paper (i.e. GATA3 and XBP1) were not examined specifically because it was not clear which rows corresponded to those genes. The dataset will be examined more closely to identify which rows correspond to which genes.

```
cancerMatrix = geoseriesread('GSE5325_series_matrix.txt');
cancerData = double(cancerMatrix.Data);
% clean data
cancerData(isnan(cancerData)) = 0;
[signals, pc, vars] = pca(cancerData);

% calculate total vars
totalVar = sum(vars);
[vm, vn] = size(vars);
varProportions = arrayfun(@(x) x / totalVar, vars);
bar(1:vm, varProportions);
title('Proportion of Variation Explained by PCs');
xlabel('Principal Component');
ylabel('Proportion of Variance (%)');
```

6 Analysis of Dataset 2

This dataset can be found here[3].

There are 54,675 dimensions with 54 samples. More specifically, gene expression levels were measured in blood samples of male subjects who ingested alcohol. These blood samples were taken at five different time points, when their blood alcohol content was 0%, 0.04%, 0.08%, 0.04%, and 0.02%. This dataset will be examined more closely, and PCA will be run on it (unfortunately, understanding the derivation took longer than planned).

```
etohData = geoseriesread('GSE20489_series_matrix.txt');
% clean data
pcaResults = pca(etohData.Data);
```

7 Conclusion

It is clear that PCA is important, and understanding it would certainly be useful when analyzing and visualizing high-dimensional data. However, it has some limitations, which have been addressed in various modified forms. For example, it is assumed that the data can be represented with linear models, which may not always be the case. Thus, when nonlinear transformations are performed prior to PCA, the process is often called kernel PCA. Furthermore, PCA assumes that the data can be represented with normal distributions, but this may also not be the case, so by removing this assumption, nonlinear optimizations can be made with Independent Component Analysis (ICA). It should be noted, though, that with enough samples, the Central Limit Theorem makes it reasonable to assume normal distributions. Still, these modified forms could be interesting to examine further.

8 References

1. https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE5325 2. Ringnr, M. (2008). What is principal component analysis? Nature Biotechnology, 26(3), 303-304. doi:10.1038/nbt0308-303 3. https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE5325