Ali Mohammadian

Matrix Theory, Math 5524

April 25, 2020

Principal Component Analysis

Introduction

We will utilize a linear transformation technique known as Principal Component Analysis (PCA) with the aim of performing a linear projection of high dimensional data onto a lower dimensional subspace where we retain the maximum amount of variance. We will discuss the theoretical nature of PCA and provide code examples of its utility.

Motivation for Principal Component Analysis

To predict the efficacy of distinct dimensions when used to describe a dataset, consider that many dimensions could involve correlated relationships with respect to each other for data being examined. It's prudent and useful to know what particular dimensions correlate and how strongly.

Dimensionality reduction is carried out when there is a need to view a smaller set of dimensions. Dimensionality reduction can either involve reducing the number of dimensions examined, or extracting information from those dimensions, thus creating a new set of axes (dimensions) to model the dataset on. In the case of simply reducing the number of dimensions, consider the resulting loss of variance (information). In the case of extracting new dimensions from the ones we had *a priori*, the new dimensions can maximize variance with respect to a dataset (retain maximum information given by the data matrix). [1] We can focus on a limited number of these new dimensions, which we will denote **principal components**, where the number of principal components will be less than our original number of dimensions.

Theoretical Setting

We will show the utility of using PCA with the Iris flower dataset. [2] Prior to doing so, we will lay the theoretical foundation relating to how and why PCA works.

Singular Value Decomposition

SVD is where any real matrix $A = USV^\top$ such that U, V are orthogonal matrices with orthonormal eigenvectors chosen from $AA^\top$ and $A^\top A$, respectively, and S is a diagonal matrix with r singular values, where r is the rank of A. With regards to dimensions: $A_{mxn} = U_{mxm}S_{mxn}V^\top_{nxn}$. Finding the SVD involves finding the eigenpairs of $AA^\top$ and $A^\top A$. The eigenvectors of $A^\top A$ provide columns for V, and the eigenvectors of $AA^\top$ provide columns of U. The singular values in S are square roots of eigenvalues of $AA^\top$ or $A^\top A$. Note that $AA^\top$ and $A^\top A$ have the same positive eigenvalues. Also, the singular values of S are arranged in descending order, are real numbers, and if A is real, U and V are real. [3] In comparison to an eigendecomposition, which works only on square matrices, SVD can work on rectangular matrices, providing it more adaptability in use for real-world datasets.

Covariance and Variance

Covariance is a measure of how much two dimensions correlate, denoted $cov(X, Y)$. This correlation can be described by how much a given point deviates from the mean(s) of our dataset with respect to those two dimensions. If a datapoint is greater than it's mean with respect to one dimension, but less than it's mean with respect to another dimension, we deduce a negative correlation between those dimensions for that point. Otherwise, we deduce a positive correlation between those dimensions.

A dataset with more negative trend has a corresponding negative covariance, and a dataset with more positive trends has a positive covariance. A dataset with a spread of

points biased towards a small number of dimensions will have a lower covariance (either positive or negative), and otherwise will have a higher covariance (either positive or negative). This will result in a covariance closer to zero. [4]

Measuring points in a dataset along just one dimension is a measure of variance, denoted $cov(x, x)$.

Formally, we denote the covariance of two variables X and Y as $cov_{X,Y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{N-1}$. [5] We can now form a covariance matrix between all dimensions provided, who's eigenpairs will describe a full multi-dimensional dataset. The eigenpairs with respect to their eigenvalues will denote how much importance they have in describing our dataset. We will examine the K most important eigenpairs and disregard the $N - K$ remaining eigenpairs. This is because the $N - K$ eigenpairs will not provide a meaningful contribution in adding variance. This refers to SVD decomposition where we retain the respective K largest singular values.

Relation of the Covariance Matrix and SVD

Given a real covariance matrix C, C is always symmetric as the covariance of two dimensions is defined as $cov(X, Y) = E[(x - E(X)) * (y - E(Y))]$, and the equation doesn't change upon switching the positions of x and y. [6] Since C is symmetric and real, it can be diagonalized. [7] Note $C = \frac{1}{N-1}A^\top A$ to be the NxN covariance matrix and $A = USV^\top$ to be the SVD of an MxN data matrix A, where N is the set of features or dimensions and M is the set of data points with respect to those dimensions. We claim that the eigenvectors of C will be the same as the right singular vectors of A.

$Proof:$

$A^\top A = VSU^\top USV^\top = VSSV^\top = VS^2V^\top,$

and $C = V\frac{S^2}{N-1}V^\top$.

Hence, the eigenvectors of C are the same as the right singular vectors and the eigenvalues of C can be denoted as $\lambda_i = \frac{s_i^2}{N-1}$. [8]

Principal Component Analysis, Graphical Representation:

PCA aims to use the principal components from the covariance matrix to describe our new coordinate system.

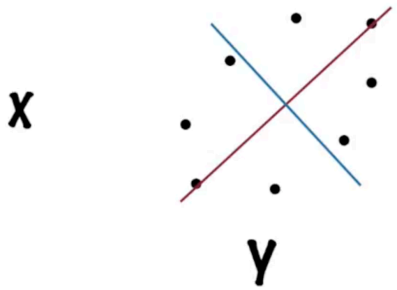

Figure 1: Depicts a scatterplot showing X and Y dimensions having a positive correlation, with the red line being our first principal direction and the blue line orthogonal to it our second principal direction. Image is from [9]

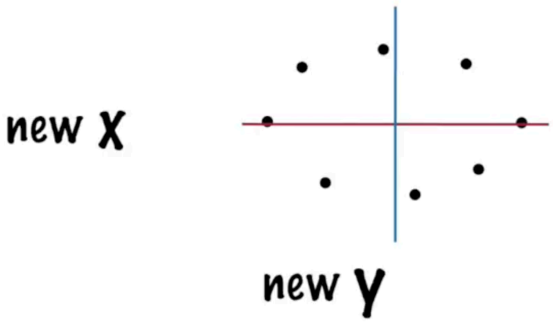

Figure 2: Depicts the same points with our new coordinate system utilizing our principal components. Image is from [9]

Essentially, PCA asks the question for whether a new basis exists in which a linear combination of the original basis will better express our dataset.

Implementing Principal Component Analysis:

Using the methodology described in obtaining a matrix with right singular values, or equivalently the covariance matrix, we have applied PCA to the Iris dataset. The Iris dataset contains 4 dimensions: sepal length, sepal width, petal length, petal width. There are 3 class labels: Iris Setosa, Iris Versicolour, and Iris Virginica. In general practice, there are three steps involved in applying PCA:

1. Organize the dataset as an MxN matrix.
2. Subtract off the mean for each row of data.
3. Calculate the SVD or the eigenvectors of the covariance matrix.

We can then examine the variances associated with the principal components, where we find the largest variances (most information) is given by the first k principal components. In our implementation, our K principal components refer to '1st PC' and '2nd PC' shown below.

We have attached code in python showing such an implementation.

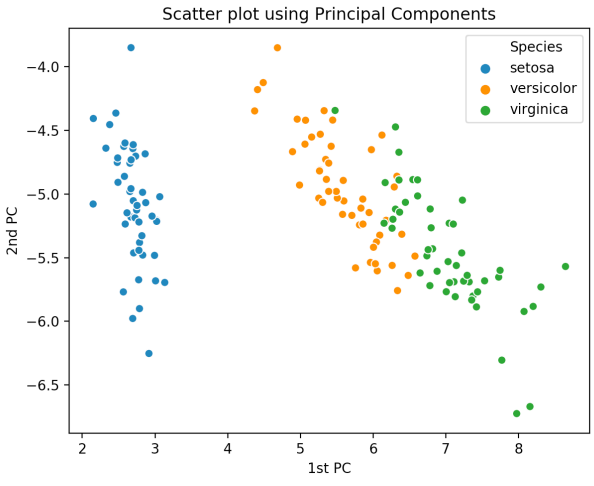

Figure 3: Implementing PCA on the Iris dataset

As shown, we have described the data with just two principal components.

The following is an outline (flowchart) of what the code does, **line numbers are references to pca.alim2.py**:

1. Import the '*Iris.csv*' file which contains our dataset. **[Line 8]**
2. Generate a vector which contains all of the rows of the original matrix but only the first column, which will be our ID labels (0, 1, 2, 3, 4). **[Line 15]**
3. Normalize the data by removing the mean (mean becomes 0, with standard deviation of 1) and scaling to unit variance. **[Line 25]** Also see [11] for details on this normalization process.
4. Create a matrix from the normalized dataset in step 3. **[Line 30]**
5. From the matrix in step 4, calculate and create our corresponding covariance matrix. **[Line 34]**
6. From the covariance matrix created, calculate the eigenpairs (which will equal the right singular vectors for the respective SVD decomposition). **[Line 42]**
7. Create a new matrix from step 6 that has all rows but only the first two columns. These columns are our principal components **[Line 49]**
8. With the ID labels in step 2, create a new coordinate system with the columns from step 7. **[Line 55]**
9. Generate new description labels called '1st PC', '2nd PC', and 'Species'. **[Line 63]**
10. Plot the labels with the new coordinate system and display results. **[Line 72-74]**

Limits and Assumptions [10]

1. We frame our problem as a change of basis, which requires linearity.
2. Sufficiency of mean and variance in terms of describing a probability distribution. The only zero-mean probability distribution that can be fully described by the variance is the Normal or Gaussian distribution, so the probability distribution for each data point is assumed to be Gaussian. Deviating from this could invalidate results.

A Gaussian distribution will guarantee the signal to noise ratio, denoted $SNR = \frac{s^2_{signal}}{s^2_{noise}}$, and the covariance matrix can characterize the noise and redundancies of the data. We say a high SNR will represent high precision data, and a low SNR will indicate noisy data.

3. We assume large variances will be more important.
4. The principal components are orthogonal, making the PCA soluble with decomposition techniques.

Conclusions

PCA's strengths and weaknesses revolve around its non-parametric model. If one is cognizant of some features *a priori* as being a part of describing a system, they can parametrically incorporate those assumptions. With an increasing amount of data constantly being assessed throughout many fields, comes an increasing amount of noise and uncertainty. Tools like Principal Component Analysis allow for scientists to hone in on dimensions that maximize variance, disregard the dimensions with low variance, and discover new dimensions to consider post-PCA analysis when evaluating exactly what dimensions ended up correlating the strongest.

References:

[1]. Prof. Alexander Ihler, 'PCA, SVD', University of California, Irvine. https://www.youtube.com/watch?v=F-nfsSq42ow

[2]. https://archive.ics.uci.edu/ml/datasets/iris - Dataset of Iris flowers used in experiment

[3]. Singular Value Decomposition - MIT. http://web.mit.edu/course/other/be.400/OldFiles/www/SVD/Singular_Value_Decomposition.htm

[4]. https://www.youtube.com/watch?v=5HNr_j6LmPc - Visual Explanation of Principal Component Analysis, Covariance, SVD

[5]. https://stattrek.com/matrix-algebra/covariance-matrix.aspx - Discussion on covariance matrices.

[6]. https://stats.stackexchange.com/questions/52976/is-a-sample-covariance-matrix-always-symmetric-and-positive-definite - Properties of covariance matrix

[7]. https://people.math.carleton.ca/~kcheung/math/notes/MATH1107/wk10/10_symmetric_matrices.html - Properties of symmetric matrices

[8]. http://www.ifis.uni-luebeck.de/~moeller/Lectures/WS-16-17/Web-Mining-Agents/PCA-SVD.pdf - Relation between PCA and SVD

[9]. https://www.youtube.com/watch?v=5HNr_j6LmPc - Visual explanation of PCA, SVD

[10]. https://www.cs.princeton.edu/picasso/mats/PCA-Tutorial-Intuition_jp.pdf - Mathematical derivation of PCA, SVD

[11]. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html