MATH108B: Principal Component Analysis

Bella, Sadie, Trevor, Lia, Mitchell

June 2020

1 What is Principal Component Analysis?

In order to derive meaning from sets of data, it is crucial that they can be properly interpreted. In one or two dimensions, this can be a simple matter of identifying a pattern or correlation. When trying to analyze data in a larger dimension, however, noticing patterns becomes much more challenging. **Principal Component Analysis (PCA)** is a method that allows the dimension of the data to be reduced without losing its meaning in context. PCA transforms a large set of (possibly) correlated variables into a much smaller, uncorrelated set of variables that maintains most of the information that the large set contained. It can be thought of as a transformation of data from the initial variable coordinate system to some new, orthogonal axes that coincide with the maximum amount of variation within the original data. This allows for extremely large sets of data to be visualized and understood in a clear and concise way. PCA is a method that provides simplicity in analysis without sacrificing accuracy. In the following paper, we will define and contextualize some fundamental concepts that are a part of PCA, deconstruct and explain the math behind the method, and give an/some example(s) that will demonstrate the usefulness of PCA when applied in disciplined analysis.

2 Fundamental Concepts

In this section we hope to provide a basis of knowledge to contextualize the operations utilized when performing principal component analysis. Listed below are the explanations of concepts which are fundamental to the understanding of PCA.

- We begin by defining the simple concept of **matrices**. Matrices are arrays of quantities which form a square or rectangular table of values. When matrices are used to store data, the vertical columns represent one type of variable, such as characteristics or outcomes, and horizontal rows represent another, typically observations.

- From matrices we may acquire eigenvalues and eigenvectors. First remember that a vector is just a quantity which has a direction and magnitude. For a set of data, a vector orients a part of the information in space relative to the rest of the information. We consider the multiplication of a vector by a matrix to be a transformation of that vector. **Eigenvectors** are vectors that denote a specific direction which, when transformed by a specific matrix, simply become a scalar multiple of the vector in that direction. The **eigenvalue** that correspond to an eigenvector is the scalar value that describes the scaling of a vector in that specific direction. In other words, for a given matrix, the eigenvectors are those which maintain their direction when undergoing the transformation performed by that matrix, and eigenvalues are the measurement of how much the transformation stretches or compresses vectors in that specific direction. For some matrices, one may acquire an **eigenvalue decomposition**, or a factorization of the matrix in terms of only its eigenvectors and eigenvalues. An eigenvalue decomposition of a matrix is a valuable resource for summarizing the manner in which the data within the matrix behaves.

- The patterns and behavior of data cannot be interpreted without an understanding of the **variance** of the variables. The term variance refers to the measurement of the variation of a single variable. Consequently, **covariance** is the measurement of variation between two variables with respect to each other. When the covariance is positive it means the two variables are correlated, meaning an increase in one points to an increase in the other. When covariance is negative, the variables are inversely correlated, so an increase in one gives a decrease in the other. The **covariance matrix** for a matrix X is given by $X^T X$, where X^T represents the *transpose* of matrix X. The covariance matrix has the variances of each variable on its diagonal, and the covariance of that variable with each corresponding variable as its other entries.

- Now we define the **principal components** of PCA, which are the transformations of the eigenvectors of the covariance matrix. The corresponding eigenvalues for each eigenvector thus give the amount of variance in each principal component. Principal components can be viewed as uncorrelated variables that are linear combinations of the initial variables and give a new intuition to interpreting the data. They are uncorrelated because they are all perpendicular, or orthogonal, to one another. This is sufficient to form an *orthonormal basis* for the covariance matrix. Alone, they don't have much meaning. However, coheisvely they represent the directions within the data which explain the greatest variance among the variables, thus help us develop an understanding between the relationship of the data. Thus, they may be viewed as new axes that give the best angle from which to view the data. When performing PCA, principal components are ordered from those with the highest variance to those with the lowest. In other words, the principal components are ordered so that their corresponding eigenvalues are ordered from largest to smallest. The first principal component accounts for the greatest amount of variance in the data and so on. This is why dimension reduction is possible without losing information, as the components that best describe the data are prioritized, and the data retains its relation within itself throughout the process.

- The **loadings** in principal component analysis are the elements of the eigenvectors of the covariance matrix, or the principal components. Loadings describe the covariances between the variable and the unit-scaled components. In simpler terms, they explain how much each eigenvector contributes to a principal component. Loadings form the entries to our singular matrix of eigenvalues that we use in singular value decomposition.

- Principal component analysis **scores** are the projections of the initial data into the space defined by the principal components (the space of eigenvectors of the covariance matrix, or the orthonormal basis). A projection refers to the transformation of variables from one space to another. PCA scores are the coefficients for each initial variable in each principal component; they reflect how much of each initial variable is a part of each principal component.

- One method to perform principal component analysis is by using singular value decomposition. **Singular value decomposition** is a generalization of eigenvalue decomposition that can be applied to any matrix. It allows for data sets that are not as user-friendly to be deconstructed so that the behaviour of the data can be more easily identified. **Singular values** for a given matrix are values that serve as eigenvalues for both the matrix as well as its transpose. Mathematically, they are the square root of the eigenvalues of the covariance matrix. Geometrically they may be viewed as the norms of the results of applying the matrix to an eigenvector of the covariance matrix. Singular values describe the same idea of stretching that eigenvalues describe for the more user-friendly matrices.

3 Classic Method for PCA

In this section, we discuss the classic method for PCA, namely by using the covariance matrix X^TX of a centered data matrix X to generate two matrices: a diagonal matrix composed of the eigenvalues of X^TX, and another matrix composed of the loadings. We will show that the projection of X onto the matrix of loadings produces a matrix of scores, which, using PCA, can reduce the dimension of our original data to present our data more effectively.

Lemma. For any $n \times m$ matrix A, A^TA is symmetric and $m \times m$.

Proof. Since $(A^T)^T = A$, and $(AB)^T = B^TA^T$, thus $(A^TA)^T = A^T(A^T)^T$ $= A^TA$, hence A^TA is symmetric. Furthermore, since A^T is $m \times n$, while A is $n \times m$, then A^TA is $m \times m$.

■

This tells us that for our original $n \times m$ data matrix X, the covariance matrix $X^T X$ is symmetric and is $m \times m$. The Spectral Theorem tells us that given a symmetric matrix, we can decompose it into an orthonormal matrix Q that is an eigenbasis- that is, whose columns are the eigenvectors of $X^T X$, and a diagonal matrix Λ whose entries are composed of the eigenvalues of $X^T X$ such that

$$X^T X = Q \Lambda Q^{-1}.$$

Definition. Let the **eigendecomposition** of our matrix $X^T X$ be the decomposition of our covariance matrix in the following sequence: Q, which is our eigenbasis, and the diagonal matrix of eigenvalues such that the entries of Λ start from the top left, and each column $\vec{v}_i$ and each eigenvalue λ_i occupy the ith spot in their respective sequences.

Since the largest eigenvalues correspond to the eigenvectors which representing the greatest degree of variance, when we reorder our eigenvalues in descending order, then our corresponding eigenvectors—which we now call loadings—will also be reordered such that the loadings that correspond to the measurements with the greatest variance will be listed first. Now, we take the reordered principal components in our sequence and use them as the columns of a new matrix W.

As we established above, Q is orthonormal, and since W is a reordering of the same columns of Q, then W must also be orthonormal. Thus, we can project our original $n \times m$ data set X onto this $m \times m$ orthonormal basis W. Let T be the data set such that

$$T = XW.$$

where T is our **principal component matrix**.

Since each row x_i of X represents an input type, when x_i is multiplied to a loading $\vec{v}_j$ in W, its score t_{ij} represents the projection of one of the input's recorded measurements into the new basis. Thus the rows of T form the original inputs, but with new projected measurements.

Since X is $n \times m$, and W, is $m \times m$, then T is also $n \times m$, which means we haven't reduced principal components at all, and in fact the data is arguably just as robust as before. However, since we ordered our loadings from most variable to least variable, then we can reduce the dimension of W to truncate the principal components that describe the dataset the least from the back of W. In fact, we can choose some dimension $r < m$ such that W is $n \times r$, which when multiplied with X, makes T $n \times r$. So our data remains untouched, but the less significant measurements are removed.

For example, if we have a list of 400 inputs, with 20 different measurements, our X would be 400×20. With this much data, it becomes difficult to represent patterns accurately and clearly, especially as a graph. However, we could follow this method of PCA, and perhaps we could remove 18 principal components that were far less variable, depending on our desired variance. Then our W would be 400×2, hence our T would be 400×2, which could then be represented visually as a two dimensional scatter plot.

However, we can't always choose $p = 2$ for the convenience of graph generation. Instead, PCA allows for the reduction of big data by presenting a method to which data can be filtered out for convenience. In addition to another form of PCA, we will explain further the method for selecting the appropriate quantity of loadings.

4 Using Singular Value Decomposition to Perform PCA

In this section, we further discuss a method of performing PCA that uses singular value decomposition (SVD). This method of computation allows us to determine the number of principal components needed to adequately describe the data.

When working with large amounts of data, computing the eigendecomposition of $X^T X$ can be computationally challenging. By using singular value decomposition, we can compute the principal component matrix T without finding our eigendecomposition, or W. SVD can be performed on any matrix, so taking the singular value decomposition of the data matrix X, we have,

$$X = U\Sigma V^*,$$

$$\Sigma = \begin{pmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_k \end{pmatrix} \text{ for singular values } \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq 0$$

and U and V are unitary matrices ($U^*U = I, V^*V = I$). This represents the transformation X as follows: V^* is a rotation of the standard basis, Σ then scales this by its singular values, then U is another rotation. Additionally, $V = W$ from above. We can then say,

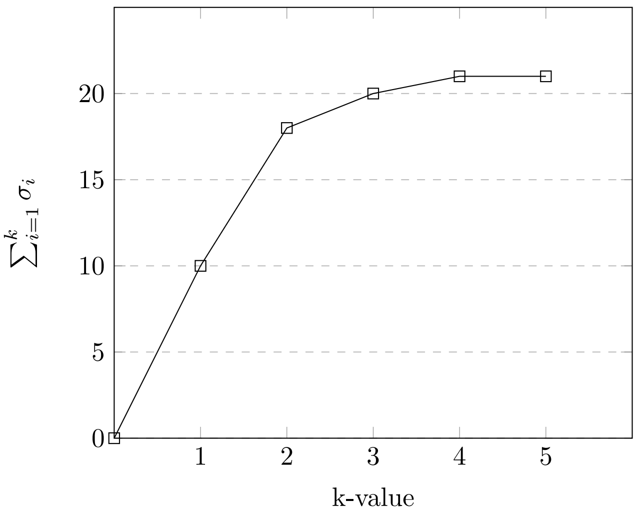
$$T = XV = U\Sigma V^*V = U\Sigma.$$

Then,

$$T_r = U_r\Sigma_r.$$

which represents the first r columns of the corresponding matrices and help us reduce the dimension of the data.

SVD can also help determine where to truncate data by finding which value of r is a sufficient representation of the data. Since the singular values are ordered, graphing each k-value against $\sum_{i=1}^{k} \sigma_i$ gives a curve that approaches a maximum value. Since the singular values get smaller, this curve plateaus. Thus, through this graph, we can see how many values it takes to describe a significant portion of the data. With steeper curves, this cut off is easier to determine. With these graphs, there is an easily identifiable k-value for which after the graph seems to flatten. This k-value is a good choice for r. For example, in the graph below, the curve begins to flatten at k=2, so r=2 is a good choice.



For graphs that increase more gradually and lack an obvious cut-off point, you can choose a percentage of the data to represent. For example, if you want to represent at least 95% of the data, take 95% of the maximum value of $\sum_{i=1}^{k} \sigma_i$, and determine how many values of σ it takes to reach that threshold. The number of singular values it takes to reach this will be your value for r. In the instance of a graph like this, this choice of r is fairly arbitrary—there is no particular reason to not use one more or one fewer singular value.

Moreover, regardless of how much of the data we wish to decribe, it can always be helpful to take r=2. Since two dimensions are easy to work with and visualize, taking r=2 allows us to understand the data on a geometric level.

5 How Do We Use PCA?

In this section, we will discuss the practical applications of using SVD to perform Principal Component Analysis on a matrix to tie in everything we have discussed. We will first show how we can apply PCA on a matrix of a smaller dimension. Next, we will use a large dataset and use software to break down the true applications of PCA and reducing big data into smaller dimensions. We will reiterate the utility of PCA. We will also discuss possible visualizations and how to use **scree plots** to show which principal components explain the greatest variance. Finally, we will wrap up this entire paper by summarizing the process as we solve for principal components.

5.1 PCA on a Small Dataset

The aim of this subsection is to show how the method of PCA works in lower dimensions. This helps us understand how the process works, as commonly in large datasets this work is done through software with the aid of a computer's processing power. The matrix we will be performing PCA on is:

$$X' = \begin{bmatrix} 5 & 2 \\ 6 & 3 \\ 4 & 4 \end{bmatrix} \quad \text{where x, y will denote the features (columns) of the data}$$

First, we center the data. Centering the data is an important because it helps us capture the variance of the dataset with each components being relative to *each other*, versus relative to a normal numerical scale. Geometrically, you can think of this as a change of axes. For simplicity of calculations, we will not normalize our data. Since $\bar{x} = 5, \bar{y} = 3$, we center our data by subtracting the columnwise mean from each column. Now:

$$X = \begin{bmatrix} 0 & -1 \\ 1 & 0 \\ -1 & 1 \end{bmatrix}$$

Now, we compute the covariance matrix to explore how the columns are related to each other. The covariance matrix is computed by taking the transpose of the matrix representation times itself, or $X^T X$.

$$C = X^T X = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

Notice that the matrix is symmetric, since the off diagonal entries are $cov(x, y) = cov(y, x)$. Additionally, since our off diagonal entries are negative, we know that both observations x and y are negatively correlated- that is: as one increases the other decreases. Now that we have our covariance matrix, we compute its eigendecomposition in order to perform PCA. We calculate the eigenvalues of our covariance matrix (C_λ) to form our singular matrix, which is a diagonal matrix with the square root of our eigenvalues as entries ($U_{ii} = \sigma_i = \sqrt{C_\lambda}$).

Now, we find our eigenvectors and normalize them to create an orthonormal eigenbasis. So, now we have our **singular matrix** (Σ), and an orthonormal eigenbasis or **loading matrix** (V) as follows:

$$\Sigma = \begin{bmatrix} \sqrt{3} & 0 \\ 0 & 1 \end{bmatrix}, V = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}$$

Note that V is unitary since it is an orthonormal basis of eigenvectors. That is, $V^* = V^T$. Now, we find our U by finding the projection of our original data on our orthonormal covariance eigenvectors- that is, multiplying our original (centered) matrix with the respective orthonormal eigenvector, normalized with respect to the eigenvalue. That is, $U_i = X(\frac{1}{\sigma_i} v_i)$. So, we have:

$$U_1 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{-1}{\sqrt{6}} \\ \frac{1}{\sqrt{6}} \end{bmatrix} = \begin{bmatrix} \frac{-1}{\sqrt{6}} \\ \frac{-1}{\sqrt{6}} \\ \sqrt{\frac{2}{3}} \end{bmatrix}$$

$$U_2 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix}$$

We concatenate the columns of U to get

$$U = \begin{bmatrix} \frac{-1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{6}} & \frac{1}{\sqrt{2}} \\ \sqrt{\frac{2}{3}} & 0 \end{bmatrix}$$

Now, we have calculated our matrix of left-singular vectors and have finished performing SVD. To find our principal component matrix, or P, we take our orthogonal projection matrix and multiply it with its singular values to find the weight of each principal component which is the projection of the orthogonal basis scaled to the eigenvalues:

$$P = XV = (U\Sigma V*)V = U\Sigma \text{ using SVD, and since V is unitary}$$

$$= \begin{bmatrix} \frac{-1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{6}} & \frac{1}{\sqrt{2}} \\ \sqrt{\frac{2}{3}} & 0 \end{bmatrix} \begin{bmatrix} \sqrt{3} & 0 \\ 0 & 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 & -1 \\ -1 & 1 \\ 2 & 0 \end{bmatrix}$$

5.2 What PCA Tells Us

Now that we have our principal component matrix, how do we use this to find out information about our dataset?

Variance is one way to explain our data, specifically the range of values our dataset takes. One way to compute the **total variance** of the original data is to compute the sum of the columnwise variance. When we perform PCA, we can also compute total variance when we sum all the variance of the principal components. Remember, PCA only 'twists' and 'flattens' our data if you will, but does not change the relation of the variables to each other.

When we perform PCA, we can identify which components of our principal matrix describe the data best by calculating the **explained variance**. This means we can condense our original matrix by selecting only a few columns that will describe the data to our desired accuracy.

The **fraction of variance explained by each principal component** is the ratio of the variance of a principal component, divided by the total variance. Mathematically speaking, we can compute total variance with N observations after performing PCA by doing the following:

$$Var(X) = \frac{\sum_i \sigma_i^2}{N}$$

To find the variance explained by the ith PC or $(Var)_i(X)$:

$$(Var)_i = \frac{\sigma_i^2}{\sum_i \sigma_i^2}$$

Note that N, or the number of data points we have cancels out. Thus, to find how many principal components we need to use to explain a certain variance, we sum over the total explained variances to get the desired value.

One way of visualizing how PCA helps us measure variance is through a **scree plot**. A scree plot shows the fraction of variance explained by each principal component in descending order. We plot our ith principal component on the x-axis, and the fraction of variance explained on the y-axis.

5.3 A Scree Plot in Context

Using the 2-dimensional matrix from section 5.1 as an example, we can use a scree plot to visualize how Principal Components explain the same amount of variance as the original features of the data. In the original matrix, each feature represents an equal amount of the total variance. In our PC matrix, each feature (principal component) represents more variance in the data than its following principal component. We can see that in our specific example the two features of the original matrix explained 50% of the total variance each whereas the first Principal Component represents 75% of the total variance and the second 25%. These variance explained percentages can be calculated manually. We know that our matrices are centered so this means the variance of each matrix column is

$$\frac{\sum_i^n (x_i^2)}{n}$$

Now we show that the Principal Component matrix explains the same amount of variance as the original matrix. Note that the variance of a centered dataset is the same as the original data- this is why we use PCA because it maintains the original relationship. Below we compute the total variance, and as explained earlier the explained variance of each principal component is the ratio of variance of the individual column with the total variance.

$$\text{Total Variance} = Var(\begin{bmatrix} 0 & -1 \\ 1 & 0 \\ -1 & 1 \end{bmatrix}) = Var(x) + Var(y) = \frac{2}{3} + \frac{2}{3} = \frac{4}{3}$$

$=$

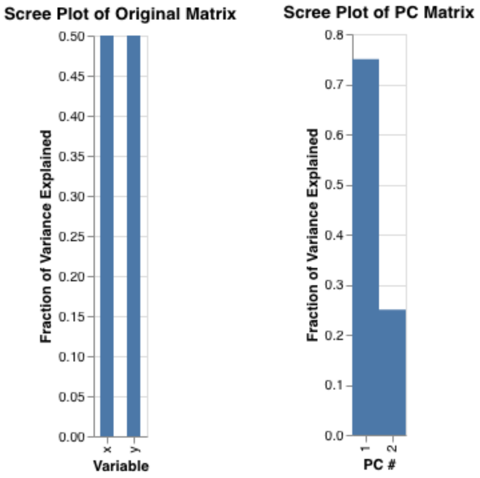
$$Var(\frac{1}{\sqrt{2}}\begin{bmatrix} -1 & -1 \\ -1 & 1 \\ 2 & 0 \end{bmatrix}) = Var(PC1) + Var(PC2) = 1 + \frac{1}{3} = \frac{4}{3}$$

Thus, now that we have total variance we can compute that the variance explained of the principal components is:

$$(Var)_1 = \frac{1}{\frac{4}{3}} = \frac{3}{4}$$

$$(Var)_2 = \frac{\frac{1}{3}}{\frac{4}{3}} = \frac{1}{4}$$

This computation is reflected in the scree plot below, which gives us a visualization of the total variance that each principal component describes.



This 2 dimensional matrix isn't the best example for PCA because both components would most likely be used anyways, as just using a single component only makes up 75% of the original data. However, in large datasets with many features, PCA can be a very important tool in reducing the amount of features needed while maintaining a desired percentage of the total variance.

6 Bibliography

1. Bagheri, Reza. “Understanding Singular Value Decomposition and Its Application in Data Science.” Medium, Towards Data Science, 9 Jan. 2020, towardsdatascience.com/understanding-singular-value-decomposition-and-its-application-in-data-science-388a54be95d.

2. Jaadi, Zakaria. “A Step by Step Explanation of Principal Component Analysis.” Built In, builtin.com/data-science/step-step-explanation-principal-component-analysis.

3. Janakiev, Nikolai. “Understanding the Covariance Matrix.” DataScience+, 3 Aug. 2018, datascienceplus.com/understanding-the-covariance-matrix/.

4. “Singular Value Decomposition.” Wikipedia, Wikimedia Foundation, 2 June 2020, en.wikipedia.org/wiki/Singular_value_decomposition.

5. Cheplyaka, Roman. ”Explained Variance in PCA”. 11 Dec. 2017, https://roche.info/articles/2017-12-11-pca-explained-variance