Summary of PCA

Alexander

July 17, 2017

A note on the syntax used in the equations in here: vectors and matrices are denoted with bold font $\mathbf{x}$ and the scalar equivalent uses normal font x. Subscripts on scalars x_n generally indicate its index within a vector or matrix, and subscripts on vectors $\mathbf{x}_n$ indicates the length of the vector.

1 Principal component analysis (PCA)

Used for:

- Dimensionality reduction
- Data compression (lossy)
- Feature extraction
- Data visualization

PCA can be defined as the orthogonal projection of data onto a low dimensional subspace, called the *principal subspace* such that the variance of the projected data is maximized [1]. It can also be defined as the linear projection that minimizes the projection cost, but we will follow the first definition, although both arrive at the same mathematical result.

Consider a dataset $\mathbf{x}_1, ..., \mathbf{x}_n$ where $\mathbf{x}_n$ has dimensionality D, in our case, $D = 49$, for the 49 frequencies recorded in the FFT. The goal is to project dataset $\mathbf{x}_n$ onto a subspace of dimensionality M where $M < D$. For our purposes, we set $M = 3$ so that we could visualize the data in a three dimensional plot.

First we consider the covariance matrix of our data $\mathbf{x}_n$

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T \tag{1}$$

where $\bar{\mathbf{x}}$ is the mean of the data and N is the number of samples. We can project the covariance onto a D dimensional unit vector $\mathbf{u}_1$ with the intention of orienting it to maximize the variance of the projection onto the vector. The maximization problem must have the constraint $\mathbf{u}_1^T\mathbf{u}_1 = 1$ which is enforced by a Lagrange multiplier λ_1. The maximization of

$$\mathbf{u}_1^T\mathbf{S}\mathbf{u}_1 + \lambda_1(1 - \mathbf{u}_1^T\mathbf{u}_1) \tag{2}$$

with respect to $\mathbf{u}_1$ is performed by setting the derivative to zero. This maximization shows

$$\mathbf{S}\mathbf{u}_1 = \lambda_1 \mathbf{u}_1 \tag{3}$$

which says $\mathbf{u}_1$ is the first eigenvector of the covariance matrix $\mathbf{S}$, or the first *principal component*. λ_1 is the first eigenvalue, or the variance of the data projected onto $\mathbf{u}_1$. This method can be extended to find the first M principal components.

In short, PCA involves calculating the covariance matrix of the data $\mathbf{x}_n$ and finding its first M eigenvectors, then projecting the data onto the space defined by those vectors:

$$\mathbf{X}_{PCA} = \mathbf{X}\mathbf{U}^T \tag{4}$$

where $\mathbf{U}$ is a $M \times D$ matrix of principal component vectors.

2 Notes

- The code we used performs PCA slightly differently using a computationally efficient method of finding the eigenvectors. The method follows [2]
- The T-SNE algorithm is a cool way to do dimensionality reduction as well, its a non-linear method that does a better job of clustering similar data.

References

[1] C. M. Bishop, *Pattern recognition and machine learning.* springer, 2006.

[2] N. Halko, P.-G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM review*, vol. 53, no. 2, pp. 217–288, 2011.