## Lecture 15: Exploratory Data Analysis : PCA, Clustering

In this class, we are going to discuss methods that allow us to understand the large amounts of data we collected so far. The data could have been generated from any of the design methods we have discussed but we are going to consider the simplest of them all: a completely random design (CRD). Recall that in CRD, to collect data of $N$ samples with $c$ number of modifications of the design variable we aim to study, we select subsets of the $N$ samples using $\binom{N}{c}$. And for each one of the subsets, we measure our characteristics (denote them as $x_i, i \in 1, 2, 3, \ldots, N$) of interest. Now, our task is to understand "what happened in our experiment" using some summary statistics. The first thing we can do is to compute the mean and variance of each subset of the samples and perform significance tests like before. Alternatively, we can also use various visualization techniques to see if we can pick up any trends and outliers. There are obvious challenges with these methods especially when our measured characteristics are not scalar values but are arrays of different properties, microscopic images, or a function-like measurement (such as spectroscopy, or X-ray diffraction). Typically, scientists use a couple of methods to visualize statistical summaries of the so-called high-dimensional data: dimensionality reduction and clustering. Here, the usage of the word 'dimension' is highly misleading: it has nothing to do with the commonly accepted notion of dimension as the minimal number of basis sets required to represent any point in space but rather simply refers to data sets where the variation between different points is capture along different arrays. Here we have moved to a more geometric interpretation of the experiment where each sample is a point in a space whose Cartesian coordinate axes are denoted using our high-dimensional $x_i$. It is common to refer to our modifications as labels identified by the subset they belong to. Within this geometric notion, we represent our experiment as a collection of data point $\mathcal{D} = (x_i, y_i)$ where $x_i$ is our high-dimensional characteristic and $y_i$ is our modification using the label notation. The statistical approaches we use are also called unsupervised learning wherein the information on the modifications (or labels) is left out in the models but is only used to draw inferences. Assuming that we have collected a characteristic that has $d$ different values, we denote the data from the experiment as a matrix $X$ of size $N \times d$ such that each row is a sample.

## Principal Component Analysis (PCA)

The first model we discuss is a dimensionality reduction method called PCA short hand for principal component analysis. The model assumes a linearity of the samples $x_i$ around a single mean $\mu$ parameterized by a lower dimensional (say $q$) vector $\lambda$ and a matrix-valued parameter $V_{d\times q}$:

$$f(\lambda) = \mu + V\lambda$$

The linear model of PCA is a generative model where by varying the values of $\lambda$ you can emulate responses corresponding to your experimental data used to obtain the parameters $\mu, V$. A solution to the PCA problem is obtained by minimizing the reconstruction error on the X given as:

$$\begin{aligned}\text{error} &= \sum_{i=1}^{N} \|x_i - f(\lambda_i)\|^2 \\ &= \sum_{i=1}^{N} \|x_i - \mu - V\lambda_i\|^2\end{aligned}$$

A solution that minimizes the error above is given as follows:

$$\hat{\mu} = \bar{x}_i$$
$$\hat{\lambda}_i = V^\top (x_i - \hat{\mu})$$

Substituting them into the reconstruction error formula we obtain :

$$\text{error} = \sum_{i=1}^{N} \left\| (x_i - \hat{\mu}) - VV^\top (x_i - \hat{\mu}) \right\|^2$$

Which is equivalent to saying that we want to project mean-centered $x$ using a matrix $K = VV^\top$. Suppose further that we want the projection $V\lambda$ to maximize the variance along each dimension. Note that the above projection would place $d$ dimensional vectors into a much lower dimension $q << d$. One way to maximize the covariance after projection along each dimension is to have a diagonal covariance matrix in $q$ such that the variance between dimensions is low while it is maximum with self. Using the matrix notation for $X_{N\times d} = V_{d\times q}\Lambda_{q\times d}$, we can write the covariance of the projected points :

$$\begin{aligned} C_\lambda &= \frac{1}{n}\Lambda\Lambda^\top \\ &\propto V^\top \tilde{x}\tilde{x}^\top V \quad (\tilde{x} = x - \hat{\mu}) \\ &= V^\top C_{\tilde{x}} V \end{aligned}$$

The covariance in the $d$ dimensional space $C_{\tilde{x}}$ is a symmetric positive definite matrix and thus can be decomposed (using a result from spectral decomposition of matrices) as follows:

$$C_{\tilde{x}} = E^\top DE$$

where $E$ is the Eigen vector matrix and $D$ is the diagonal matrix of Eigen values. Substituting this back in the covariance for $q$ dimensional space, we get that :

$$C_\lambda = (V^\top E^\top) D (EV)$$

Therefore, to make the $C_\lambda$ a diagonal matrix, we need to set $V = E^{-1}$ which completes the parametrization of our PCA model. The Eigenvector matrices are orthogonal thus their inverse is their transpose i.e. $V = E^\top$. We can use the Eigenvalues in the diagonal matrix $D$ to quantify the projected variance along each dimension in the $q$ space thus assigning importance. When we truncate the Eigenvalues to a threshold, we get much lower dimensional projections of our original measurement which can be used to summarize our experiments. Note that the lower dimensional parameterization of any sample is given by: $\lambda_i = E(x_i - \hat{\mu})$.

**Example : Seeds dataset**

We will use the implementation of PCA in a python package called `scikit-learn` to illustrate the application of PCA to data collected from a random sampling or design method. The data consists of wheat seeds of three kinds each randomly sampled and measured under an X-ray technique. Seven parameters of interest are measured for each seed, and we have a total of 210 samples. We can use PCA to obtain a primary understanding of whether the three kinds studied are inherently

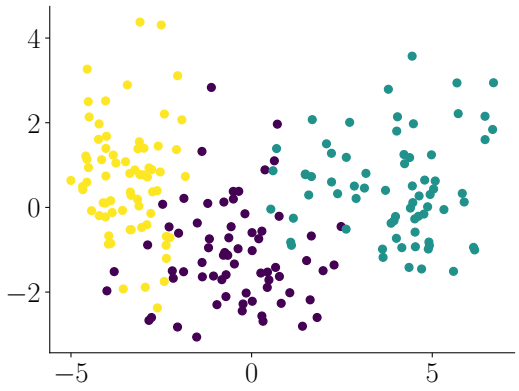


Figure 1: PCA of wheat seeds revealing that the seeds studied have a near linear separation between the measured properties

different given the measured properties. In Figure 1, we plot each sample (colored based on the type) in the two-dimensional projection using the respective $\lambda_i$ values.

Observe that the three types of seeds are nearly linearly separable in the two-dimensional projection with diagonal covariance given by $\begin{pmatrix} 10.79 & 0.0 \\ 0.0 & 2.13 \end{pmatrix}$. The accompanying code can be found in the jupyter notebook linked on the website.

## Clustering ($k$-means)

Another popular approach to obtain a similar summary as before for a completed experiment is clustering wherein we try to find similarities in the data such that they can be grouped together. The notion of similarity plays a crucial role in the success of these algorithms and will have a significant impact on the inferences we can obtain. For the purposes of this lecture, we assume that we can obtain a similarity based on a measure of array distances such as the mean squared error between an array of properties. A classical clustering method is called $k$-means where we aim to find $k$ centers in the data that can serve as the prototypical examples of different types we expect. The algorithm proceeds in an iterative fashion between a) assigning and b) updating the means. In the assigning step, we assign a label to each sample in the data based on the computed similarity to the current set of centers. The centers are then updated based on the assignment using a simple arithmetic mean of samples that are assigned the same label. The accompanying notebook has the application of the k-means algorithm to the seeds dataset discussed above.

There are plenty of other methods that are suitable for the task of exploratory data analysis. If you are interested, I encourage you to check out Chapter 14 of the book “The Elements of Statistical Learning”.