Multivariate Stats

PCA

NULL SPACE

0.1 Introduction: setting the background

PCA is a technique that helps us reduce complex, high dimensional datasets into lower dimensions to reveal the hidden patterns in the data. We will begin to explain how PCA works with the help of a toy example. As an experimenter, we are often trying to measure various quantities in our system. Also what often happens is that when we take many measurements, the data appears unclear and clouded. Below is the experiment laid out using which we will learn the PCA technique.

- Consider that we are studying the motion of a **spring**. The system comprises of a ball of mass m attached to a frictionless spring.

- We will then stretch the spring and let it oscillate - the frictionless spring will oscillate indefinitely along the x axis about its equilibrium position with a certain frequency. So, the dynamics of this system revolves around measuring the distance (position) x at various points in time.

- Now here is the tricky bit in observation - we can look at this spring-ball system from various angles and from each angle our perspective of the position at different time points will be different. Ultimately we must decide to measure the position of the ball in a $3D$ space since we live and see in a $3D$ world (our perspective is in $3D$).

- With that said, let's say that we place 3 movie cameras around our system to observe it from 3 different angles. **NOTE** here that each camera will record a $2D$ image (since camera's only see in $2D$). So basically, **each camera would record a $2D$ image of the position of the ball**.

- Let us say that the three cameras view the ball image in three distinct directions. Let us denote these directions as: $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$. You can think of these vectors as denoting the angle or direction at which the camera is placed in the system. The picture of the system is shown below. After the system diagram we have also shown three **scatterplots**. The points on the scatterplot denotes the position taken by the ball at various points in time. The three scatterplots at differently spread out since they denote the measurements in ball position by three different viewing angles of the three cameras.

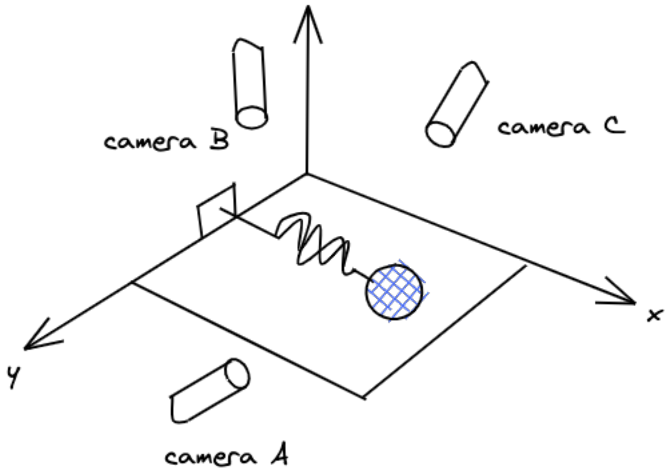

Figure 1: The system

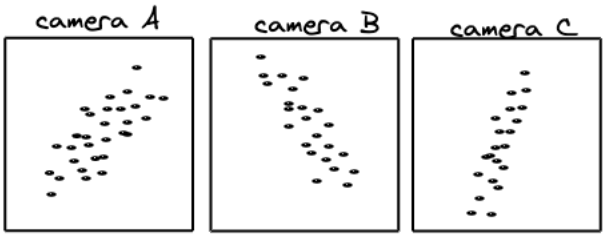

Figure 2: The different positions of ball at different angles

- Why could we not simply have just measured the movements along the x axis ? It is because in real life when we are measuring some variables, we do not know which variables might reflect the system dynamics in the best way possible. So we end up measuring more dimensions (variables) than we need.

- Remember that after all this is done, our ultimate goal is to extract the x from our data - that is one single best measure of positions of the ball at different points in time.

- Also remember that after we do our measurements, there is noise in our data. Noise means that we have many angles measuring essentially the same movement. So our goal now is to find the hidden structure in the data - the best measure (or best axis) that significantly explains the positions of the ball.

0.2 Change of basis

We want to ultimately measure the **dynamics of the system along the x axis**. But in reality, we do not know what the **real x axis is**. What we do have, is some axes along which we measured the positions of the ball. From our measurements, we need to find the x axis positions of the ball - that is, the axis along which the significant amount of movement happens in the ball. More importantly, we want to find from our data, the **unit basis vector (direction) that best represents a significant portion of the movements of the ball (the true x axis)**.

0.2.1 What is a basis ?

Recall from linear algebra concepts that **basis vectors** are linearly independent vectors with which we can represent any point in a given vector space. For example if we are given a simple $2D$ vector space where each point (or each vector) has a x and y coordinate, then one possible set of basis vectors for this space would be:

$$\boldsymbol{e}_1 = [1, 0],\ \boldsymbol{e}_2 = [0, 1] \tag{1}$$

If we package these basis vectors in a nice little matrix, we will obtain an **identity matrix** of the following form:

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{2}$$

What this means is that if we are given a point in the $2D$ space say, $(x = 3, y = 4)$, then this point can be expressed as a **linear combination of the basis vectors of the xy vector space**:

$$\begin{bmatrix} 3 \\ 4 \end{bmatrix} = 3 \times \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 4 \times \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{3}$$

NOTE - The above mentioned basis is called the **standard basis** however it is possible to have many possible basis vectors that can express a set of coordinates in a vector space. Also an important to remember is that we usualyl consider **unit basis vectors**. Remember these points as we move along. But **why are we talking about basis vectors ?** Because it is possible to express the same set of coordinates in a vector space in some other basis and uncover patterns in the data that were not visible in the standard basis representation of the same coordinates. **In the language of the geeks** - The basis vectors **span** the vector space, are unit vectors and are orthogonal to each other.

0.2.2 The naive (standard) basis

Now in our experiment, the measurement is essentially a $2D$ coordinate of position of the ball at different points in time, from different camera angles. Our notation is as follows - At a point in time (a measurement at one point in time is one **observation**), camera A records a ball position as (x_A, y_A). So when we combine one observation - that is one position measurement, with all three cameras, we get one observation as:

$$\boldsymbol{X}_i = \begin{bmatrix} x_A \\ y_A \\ x_B \\ y_B \\ x_C \\ y_C \end{bmatrix} \tag{4}$$

We hence view each trial or each observation as a 6 dimensional vector where each camera contributes a 2 dimensional projection of the ball's position to the entire vector $\boldsymbol{X}_i$. Suppose now that we have 72000 of such vectors (or such observations). Now let us denote a more general mathematical description of this scenario:

- Each sample $\boldsymbol{X}$ is an m dimensional vector where m is the number of measurement types or the number of variables.
- We say that each sample is a vector that in the m dimensional space that is spanned by some **orthonormal basis** (orthonormal just means that the basis vectors are unit vectors and are orthogonal to each other).
- Now we know from basic concepts that each observation is a point in the vector space that can be represented as linear combinations of our basis vectors.
- Now when we obtain our data measurements, they are typically in the standard basis. We saw earlier how a $2D$ standard basis looks like. In a similar manner as the $2D$ case, we can represent an m dimensional basis vector as:

$$\boldsymbol{b}_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \boldsymbol{b}_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \cdots, \boldsymbol{b}_m = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \tag{5}$$

  Again, if we package this stuff in a matrix, we will get an $m \times m$ identity matrix of the form:

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \\ \vdots \\ \boldsymbol{b}_m \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \vdots & 1 \end{bmatrix} \tag{6}$$

  Where each row is an orthonormal basis vector $\boldsymbol{b}_i$ with m components.

0.3 Change of basis

This is the centrality of PCA. We ask the question - **is there another basis, which is a linear combination of the original basis, that best re-expresses our dataset ?**

We say that PCA is nothing but a technique that re-expresses our dataset in terms of new (and more efficient) basis vectors. Now we note some important notation specific points:

- Let $\boldsymbol{X}$ denote our entire dataset, in which each **column** is a single sample (measurement) and there are 72000 such measurements or vectors. So $\boldsymbol{X}$ is effectively a $(m \times n)$ matrix where $m = 6$ and $n = 72000$.
- Let $\boldsymbol{Y}$ be another $(m \times n)$ matrix related to $\boldsymbol{X}$ via a linear transformation $\boldsymbol{P}$. So we say that $\boldsymbol{X}$ is the original data and $\boldsymbol{Y}$ is the new representation of this data and is obtained by premultiplying $\boldsymbol{P}$ with the original dataset:
$$\boldsymbol{PX} = \boldsymbol{Y} \tag{7}$$
- Let $\boldsymbol{p_i}$ be the rows of matrix $\boldsymbol{P}$.
- Let $\boldsymbol{x_i}$ be the columns of matrix $\boldsymbol{X}$.
- Let $\boldsymbol{y_i}$ be the columns of matrix $\boldsymbol{Y}$.
- **A word of caution**: In this section section we are taking $\boldsymbol{x_i}$ and $\boldsymbol{y_i}$ as columns but in later sections we will take them as row vectors. It will be specified explicitly so you need not worry.
- $\boldsymbol{P}$ is a matrix that transforms $\boldsymbol{X}$ into $\boldsymbol{Y}$.
- Geometrically, the tranformation $\boldsymbol{P}$ is such that it **rotates** and **stretches** the original matrix $\boldsymbol{X}$ such that we get $\boldsymbol{Y}$.
- The rows of $\boldsymbol{P}$ that is: $\{\boldsymbol{p_1}, \boldsymbol{p_2}, \cdots, \boldsymbol{p_m}\}$ are the **set of new basis vectors** meant for re-expressing the columns of $\boldsymbol{X}$. We write this as:
$$\boldsymbol{PX} = \begin{bmatrix} \boldsymbol{p_1} \\ \vdots \\ \boldsymbol{p_m} \end{bmatrix} \begin{bmatrix} \boldsymbol{x_1} & \cdots & \boldsymbol{x_n} \end{bmatrix} \tag{8}$$
$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{p_1} \cdot \boldsymbol{x_1} & \cdots & \boldsymbol{p_1} \cdot \boldsymbol{x_n} \\ \vdots & \dots & \dots \\ \boldsymbol{p_m} \cdot \boldsymbol{x_1} & \cdots & \boldsymbol{p_m} \cdot \boldsymbol{x_n} \end{bmatrix} \tag{9}$$
- Now we note that each column of the matrix $\boldsymbol{Y}$ is of the general form:
$$\boldsymbol{y_i} = \begin{bmatrix} \boldsymbol{p_1} \cdot \boldsymbol{x_i} \\ \vdots \\ \boldsymbol{p_m} \cdot \boldsymbol{x_i} \end{bmatrix} \tag{10}$$

  We then state that $\boldsymbol{y_i}$ is a projection of $\boldsymbol{x_i}$ onto the new basis $\{\boldsymbol{p_1}, \cdots, \boldsymbol{p_m}\}$. The **rows of P are the set of new basis vectors that re-express X**.
- We say that PCA ultimately reduces to finding the best **change of basis** and the terminology says that the set of new basis vectors $\{\boldsymbol{p_1}, \cdots, \boldsymbol{p_m}\}$ are called the **principal components** of $\boldsymbol{X}$.

0.3.1 Questions to answer

Now that the theory is laid out, we want to now answer the following questions:

- What is the best way to re-express $\boldsymbol{X}$?
- What is a good choice for $\boldsymbol{P}$?
- What features would be like $\boldsymbol{Y}$ to exhibit ?
- What do we mean by re-expressing the data in the **best way** ?

0.4 Noise and rotation

Every data has two components - a **noise** component and a **signal** component. Our goal is to find the signal (or the significant pattern in data) amidst the enormous noise (randomness in the data). We measure noise and signal with respect to **variance** and try to compute a key measure known as the **signal to noise ratio** given as:

$$SNR = \frac{\sigma^2_{signal}}{\sigma^2_{noise}} \tag{11}$$

Moreove, a high SNR ratio indicates precise measurements and a low SNR ratio indicates noisy data. Look at the figure below which shows the recording of ball movements from camera A:

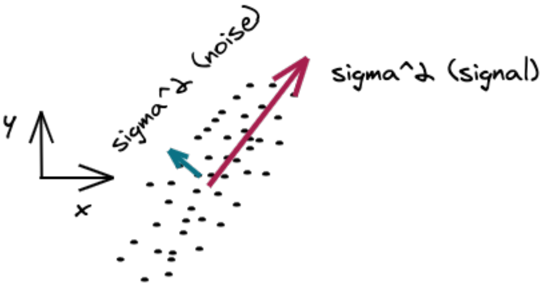

Figure 3: direction along which variance is max is along the best fit line and not along the direction of standard basis vectors

- Note that since our spring travels in a straight line, the camera should ideally record motion in a straight line as well. So, any **spread deviating from the strict straight line motion is noise**.
- The above diagram clearly shows the direction in which the main motion is (signal) and the randomness in positions (noise).
- We note that the **directions that capture the largest variances in our measurement space are our dynamics of interest**.

- Note that the best direction is not along the x axis or the y axis (basis vector directions), but rather along a sort of best fit line.
- With this it is obvious to us that the current basis vector are not so efficient since they do not capture the direction of maximum variation of the movements.
- So, we aim to now maximize the variance (and SNR) by finding an appropriate rotation of this naive basis such that our new basis vector corresponds to the direction of the maximum variance (the best fit line).

0.4.1 Redundancy

From figure 3 we see that the only measurement that really matters is the measurement along the best fit line. So we say that the the other direction is just causing redundancy in our data - it is perhaps not required to analyze our data. What redundancy in data captures is - **correlation**. We see that higher values of points along x are associated with higher values along the y direction. But the point is that - **what is the point of having two measurements when only one measurement is really all we need to capture the variation in data ?** So we answer this by re-expressing our data such that we discard this redundancy by throwing away the directions that have almost no effect in explaining significant portions of our data. This is the centrality behind **dimensional reduction** - we are reducing redundancy in data and only presenting the directions along which maximum variation in the data is explained.

0.5 The covariance matrix

Now we generalize the previous notions to higher dimensions. Let us consider two sets of measurements with 0 means.

$$A = \{a_1, a_2, \cdots, a_n\},\ B = \{b_1, b_2, \cdots, b_n\} \tag{12}$$

The variances of A and B are defined as:

$$\sigma_A^2 = \frac{1}{n}\sum_{i=1}^{n} a_i^2,\ \sigma_B^2 = \frac{1}{n}\sum_{i=1}^{n} b_i^2 \tag{13}$$

For sake of simplicity we have taken $(1/n)$ instead of $(1/(n-1))$. Similarly, the covariance between A and B is defined as:

$$\sigma_{AB}^2 = \frac{1}{n}\sum_{i=1}^{n} a_i b_i \tag{14}$$

Note that the covariance is a measure of the degree of linear relationship between two variables. Now we write A and B (the sets of measurements) in terms of row vectors:

$$\boldsymbol{a} = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix},\ \boldsymbol{b} = \begin{bmatrix} b_1 & b_2 & \cdots & b_n \end{bmatrix} \tag{15}$$

With this, the covariane can be easily expressed as a dot product matrix of these two vectors:

$$\sigma^2_{AB} = \frac{1}{n}\boldsymbol{a}\boldsymbol{b}^T \tag{16}$$

With that explained, let us take a more general case. Instead of just two sets of measurements of the form $\boldsymbol{a}$ and $\boldsymbol{b}$ let us say that we have m sets of measurements denoted as: $\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_m}$ and we suppose that each measurement variable has n observations. So if we package all these sets of measurements into a neat matrix, we get an $(m \times n)$ matrix $\boldsymbol{X}$ of the form:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x_1} \\ \vdots \\ \boldsymbol{x_m} \end{bmatrix} \tag{17}$$

NOTE that each row corresponds to a set of observations or measurements of a certain type - the row vector $\boldsymbol{x_1}$ indicates all the observation measurements along the $\boldsymbol{x_1}$ variable. On the other hand, **each column corresponds to all the different variable measurements for one particular observation**. Now we can form a general version of our **covariance matrix** as follows:

$$\boldsymbol{C_X} = \frac{1}{n}\boldsymbol{X}\boldsymbol{X^T} \tag{18}$$

We note that the $(i,j)^{th}$ element of this matrix is the dot product of the i^{th} type measurement vector and the j^{th} type measurement vector. Here are some imprtant properties of $\boldsymbol{C_X}$:

- It is a square symmetric $m \times m$ matrix.
- The diagonal terms are the variances of the different measurement types.
- The off diagonal terms are the covariances between different measurement types. The covariance values reflect redundancy in our data.
- Large values of diagonal terms are what we are interested in since they capture the interesting features of the data.
- We will now manipulate $\boldsymbol{C_X}$ to obtain $\boldsymbol{C_Y}$ by changing the basis such that the new representation is more efficient (recall the section on change of basis).

0.6 Diagonalize

Before we begin, a restatement of goals is necessary - **We want to minimize redundancy, captured by the covariance terms and maximize the signal, measure by the variance**. So in the optimized covariance matrix $\boldsymbol{C_Y}$ our off diagonal covariance terms should be 0. Additionally, each successive dimension in matrix $\boldsymbol{Y}$ should be rank ordered as per variance. Now this is essentially what the PCA algorithm does - it selects a set of **orthonormal** basis vectors, or analogously, an orthonormal matrix $\boldsymbol{P}$ with basis vectors $\{\boldsymbol{p_1}, \cdots, \boldsymbol{p_m}\}$. Here are the basic steps in the algorithm:

- Select a normalized direction in the m dimensional space along which the variance of $\boldsymbol{X}$ is maximized. Save this direction vector as $\boldsymbol{p_1}$.
- Find another direction along which the variance is maximized, but this time ensure that the new direction we are looking for is orthogonal to the first one we found. Keep doing this and saving the direction vectors as $\boldsymbol{p_i}$.
- The resulting ordered set of $\boldsymbol{p}$ vectors are called the **principal components**.

0.7 PCA using Eigendecomposition

PCA is typically solved using Eigenvector decomposition. Recall that our dataset is the matrix $\boldsymbol{X}$ of dimension $(m \times n)$ where m is the number of measurement types or variables and n is the number of observations. We want to find an orthonormal matrix $\boldsymbol{P}$ in $\boldsymbol{Y} = \boldsymbol{PX}$ such that $\boldsymbol{C_Y} = \frac{1}{n}\boldsymbol{YY^T}$ is a **diagonal matrix**. The rows of $\boldsymbol{P}$ are the principal components of $\boldsymbol{X}$. Here is how it is done:

$$\boldsymbol{C_Y} = \frac{1}{n}\boldsymbol{YY^T} \tag{19}$$

$$= \frac{1}{n}(\boldsymbol{PX})(\boldsymbol{PX})^T \tag{20}$$

$$= \frac{1}{n}\boldsymbol{PXX}^T\boldsymbol{P}^T \tag{21}$$

$$= \boldsymbol{P}\left(\frac{1}{n}\boldsymbol{XX^T}\right)\boldsymbol{P^T} \tag{22}$$

$$\boldsymbol{C_Y} = \boldsymbol{PC_XP^T} \tag{23}$$

Now as a general rule we note that a square symmetric matrix A can be diagonalized by an orthogonal matrix of its Eigenvectors. The general result states that $\boldsymbol{A} = \boldsymbol{EDE^T}$ where $\boldsymbol{D}$ is a diagonal matrix and $\boldsymbol{E}$ is a matrix of eigenvectors of $\boldsymbol{A}$ where the eigenvectors are just the columns of this matrix. Now the trick is to select $\boldsymbol{P}$ such that its rows $\boldsymbol{p_i}$ are the eigenvectors of the original covariance matrix $\frac{1}{n}\boldsymbol{XX^T}$. This would further imply that the eigenvalue matrix $\boldsymbol{E^T}$ would be equal to $\boldsymbol{P}$. WE equate the tranpose since our eigenvectors are arranged as columns whereas we want them as rows in $\boldsymbol{P}$. Note that since $\boldsymbol{P}$ is an orthonormal matrix, we have $\boldsymbol{P^{-1}} = \boldsymbol{P^T}$. Now we prove that the optimized covariance matrix is indeed a diagonal matrix.

$$\boldsymbol{C_Y} = \boldsymbol{PC_XP^T} \tag{24}$$

$$= \boldsymbol{P}(\boldsymbol{EDE^T})\boldsymbol{P^T} \tag{25}$$

$$= \boldsymbol{P}(\boldsymbol{P^TDP})\boldsymbol{P^T} \tag{26}$$

$$= (\boldsymbol{PP^T})\boldsymbol{D}(\boldsymbol{PP^T}) \tag{27}$$

$$= (\boldsymbol{PP^{-1}})\boldsymbol{D}(\boldsymbol{PP^{-1}}) \tag{28}$$

$$\boldsymbol{C_Y} = \boldsymbol{D} \tag{29}$$

With the above set of formulations it is proved that our selection of $\boldsymbol{P}$ as the matrix of eigenvectors, indeed diagonalizes $\boldsymbol{C_Y}$. Now we note the following points:

- The principal components of $\boldsymbol{X}$ are the eigenvectors of $\boldsymbol{C_X}$.
- The i^{th} diagonal value of $\boldsymbol{C_Y}$ is the variance of $\boldsymbol{X}$ along $\boldsymbol{p_i}$.
- In practise, we formulate our data matrix as a deviation matrix first, by subtracting out the means of each measurement type before we compute our covariance matrix.

References

[1] Jonathon Schlens (Google research) - A tutorial on PCA

[2] Lecture notes - Linear algebra