Lecture 4

Last time we introduced co-variance matrix from a d-dimensional data set. To recap, let $\mathbf{X}$ be a vector random variable $(X_1, X_2, \ldots, X_d)^T$. The co-variance matrix is defined by

$$Cov[\mathbf{X}] = \begin{bmatrix} Var[X_1] & Cov[X_1, X_2] & Cov[X_1, X_3] & ... & Cov[X_1, X_d] \\ Cov[X_2, X_1] & Var[X_2] & Cov[X_2, X_3] & ... & Cov[X_2, X_d] \\ Cov[X_3, X_1] & Cov[X_3, X_2] & Var[X_3] & ... & Cov[X_3, X_d] \\ . & . & . & ... & . \\ . & . & . & ... & . \\ Cov[X_d, X_1] & Cov[X_d, X_2] & Cov[X_d, X_3] & ... & Var[X_d] \end{bmatrix}$$

Our objective is to rotate our axes so that the data is"more aligned" with the new axes than with the original ones. By this we mean that in the new axes system the co-variances vanish - there is no (linear) interplay between the co-ordinates. All we need to consider are the variances (average amount of squared fluctuations) along the new axes. It will then be easy to identify the "good" directions where most of data variation happens and the "redundant ones" (those where the fluctuations are small/negligible).

Let $\widetilde{\mathbf{X}} = (\tilde{X}_1, \tilde{X}_2, \ldots, \tilde{X}_d)^T$ be the original vector random variable $\mathbf{X}$ expressed in the new axis. The co-variance matrix of $\widetilde{\mathbf{X}}$ is

$$Cov[\widetilde{\mathbf{X}}] = \begin{bmatrix} Var[\tilde{X}_1] & 0 & 0 & ... & 0 \\ 0 & Var[\tilde{X}_2] & 0 & ... & 0 \\ 0 & 0 & Var[\tilde{X}_3] & ... & 0 \\ . & . & . & ... & . \\ . & . & . & ... & . \\ 0 & 0 & 0 & ... & Var[\tilde{X}_d] \end{bmatrix}$$

How to compute just from the data what our new axes should in order to achieve the desired result (diagonal covariance matrix)? We will first look at how we project a given data set onto the new axes. This (as will be clear soon) can be achieved through a (canonical) dot product of vectors: Let $\mathbf{a} = [a_1, a_2, \ldots, a_d]^T$ and $\mathbf{b} = [b_1, b_2, \ldots, b_d]^T$ be vectors in $\mathbb{R}^d$. Let α

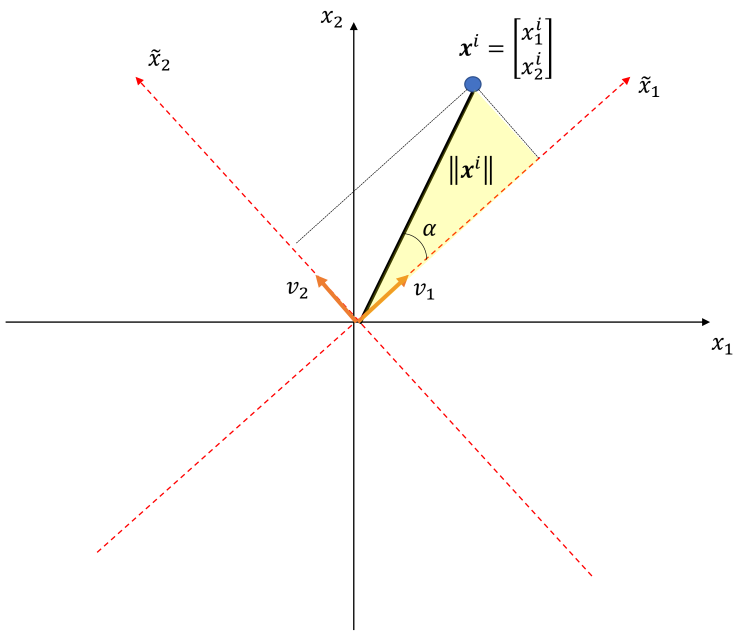
be the angle between $\mathbf{a}$ and $\mathbf{b}$. The *dot product* between $\mathbf{a}$ and $\mathbf{b}$ is

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^T \mathbf{b} = \sum_{i=1}^{d} a_i \cdot b_i = ||\mathbf{a}|| \cdot ||\mathbf{b}|| \cos(\alpha),$$

where $||\mathbf{a}||$ is the (L_2) *norm* (or length) of $\mathbf{a}$, defined as $||\mathbf{a}|| = \sqrt{a_1^2 + a_2^2 + \ldots + a_n^2}$. Note that $||\mathbf{a}||^2 = \mathbf{a}^T \mathbf{a}$.

1 Finding Our Axes in the 2-D Case

Let $\mathbf{x}^i = [x_1^i, x_2^i]^T \in \mathbb{R}^2$ be a data point in 2-D space. To provide information about our new axes, let $\mathbf{v}_1$ and $\mathbf{v}_2$ be *unit* vectors ($||\mathbf{v}_1|| = ||\mathbf{v}_2|| = 1$) that are *orthogonal to each other* ($\mathbf{v}_1 \cdot \mathbf{v}_2 = 0$). When changing our axes (x_1, x_2) to $(\tilde{x}_1, \tilde{x}_2)$, we want to project $\mathbf{x}^i$ onto $\mathbf{v}_1$ and $\mathbf{v}_2$ to obtain $\widetilde{\mathbf{x}}^i = [\tilde{x}_1^i, \tilde{x}_2^i]^T$. Let α be the angle between $\mathbf{x}^i$ and $\mathbf{v}_1$.



To obtain our first co-ordinate $\tilde{x}_1^i$, we notice that (recall $||\mathbf{v}_1|| = 1$):

$$\tilde{x}_1^i = ||\mathbf{x}^i|| \cos(\alpha) = ||\mathbf{v}_1|| \cdot ||\mathbf{x}^i|| \cos(\alpha) = \mathbf{v}_1^T \mathbf{x}^i$$

Similarly, we also find that $\tilde{x}_2^i = \mathbf{v}_2^T \mathbf{x}^i$. Using these facts, let $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2]$ be the matrix containing the direction vectors for each axes as columns. We see that

$$\mathbf{V}^T \cdot \mathbf{x}^i = \tilde{\mathbf{x}}^i$$

Suppose now that we are in our new axes system $(\tilde{x}_1, \tilde{x}_2)$; how do we express our point $\tilde{\mathbf{x}}^i$ back in the original system (x_1, x_2)? We would of course multiply $\tilde{\mathbf{x}}^i$ by the matrix $(\mathbf{V}^T)^{-1}$, the inverse matrix to $\mathbf{V}^T$. Because of the properties of the direction vectors (they are orthogonal to each other and have length 1 - we also say that they form an orthonormal basis), finding this matrix is very simple.

Claim: $(\mathbf{V}^T)^{-1} = \mathbf{V}$

Proof.

$$\begin{aligned}
\mathbf{V}^T \cdot \mathbf{V} &= \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \end{bmatrix} \cdot [\mathbf{v}_1 \mathbf{v}_2] \\
&= \begin{bmatrix} \mathbf{v}_1^T \mathbf{v}_1 & \mathbf{v}_1^T \mathbf{v}_2 \\ \mathbf{v}_2^T \mathbf{v}_1 & \mathbf{v}_2^T \mathbf{v}_2 \end{bmatrix} \\
&= \begin{bmatrix} ||\mathbf{v}_1||^2 & 0 \\ 0 & ||\mathbf{v}_2||^2 \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
&= I_2
\end{aligned}$$

Similarly, $\mathbf{V} \cdot \mathbf{V}^T = I_2$ □

2 Projecting in d-dimensional Space

There was nothing in the arguments above that was special to 2 dimensions. Exactly the same trick can be applied to change co-ordinate systems in $d > 2$-dimensional case. If we let $(\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_d)$ be our new axes and $\mathbf{x}^i = [x_1^i, x_2^i, \ldots, x_d^i]^T$ a data point expressed in the original axes, we can project our point onto our new axis using:

$$\mathbf{V}^T \cdot \mathbf{x}^i = \tilde{\mathbf{x}}^i.$$

As before, $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d]$, each $\mathbf{v}_i$ being the unit direction vector of axis $\tilde{x}_i$ orthogonal to the other direction vectors.

If we wanted to express $\tilde{\mathbf{x}}^i$ in the original axes, we would simply use:

$$\mathbf{V} \cdot \tilde{\mathbf{x}}^i = \mathbf{x}^i$$