Lecture 07: Random Matrices I

Nikola Zlatanov

Innopolis University

Advanced Statistics

20-th of March to 27-th of March, 2023

Motivation

- The vast majority of applications in data science, ML, and computer vision are based on data from random vectors and/or random matrices.
- An image mathematically is represented as a matrix.
- In fact, all images, digitally stored or processed by computers, are simply the storage or processing of matrices.
- Computers do not know that these are images. For them, these are matrices.
- Hence, we now need to develop statistics for vectors and matrices and then see how these statistics are applied in practice.

Mean of Random Vectors

- Let $\boldsymbol{X} = [X_1, X_2, ..., X_d]^T$, i.e., $\boldsymbol{X} \in \mathbb{R}^d$, be a randomly generated d-dimensional vector according to some distribution $f_{\boldsymbol{X}}(\boldsymbol{x})$.
- How to compute the mean and variance of random vectors?
- The mean of $\boldsymbol{X}$ is given by the following vector

$$E[\boldsymbol{X}] = \Big[E[X_1], E[X_2], ..., E[X_d]\Big]^T = [\mu_1, \mu_2, ..., \mu_d]^T \qquad (1)$$

  where $\mu_i = E[X_i]$, for $i = 1, 2, ..., d$.
- How about variance? Is it also going to be a vector?

Variance of Random Vectors

- If $d = 1$, then $\boldsymbol{X} = X_1$ and then we know how to compute the variance as

$$\begin{aligned}\text{VAR}[\boldsymbol{X}]\Big|_{d=1} &= E\Big[(X_1 - E[X_1])^2\Big] = E\Big[(X_1 - E[X_1])(X_1 - E[X_1])\Big] \\ &= E\Big[X_1^2\Big] - E[X_1]^2 \end{aligned} \tag{2}$$

- But what do we do if $d > 1$? How can we have a square of a vector to obtain something like $E\Big[(\boldsymbol{X} - E[\boldsymbol{X}])^2\Big]$

Covariance Matrix

- Well, for $d > 1$, the variance becomes a matrix that is called *the covariance matrix*, defined as

$$\begin{aligned} \mathrm{COV}(\boldsymbol{X}) &= E\Big[(\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{X} - E[\boldsymbol{X}])^T\Big] \\ &\overset{(a)}{=} E\Big[\boldsymbol{X}\boldsymbol{X}^T\Big] - E\Big[\boldsymbol{X}\Big]E\Big[\boldsymbol{X}^T\Big] \\ &= E\Big[\boldsymbol{X}\boldsymbol{X}^T\Big] - \boldsymbol{\mu}\boldsymbol{\mu}^T, \end{aligned} \tag{3}$$

where (a) you need to prove at home.

Covariance Matrix

- The (i, j)-th element of the matrix $\mathrm{COV}(\boldsymbol{X})$ is given by

$$\begin{aligned}[\mathrm{COV}(\boldsymbol{X})]_{ij} &= E\Big[(X_i - E[X_i])(X_j - E[X_j])\Big] \\ &= E\Big[X_i X_j\Big] - E\Big[X_i\Big]E\Big[\boldsymbol{X}_j\Big], \end{aligned} \tag{4}$$

  which is the definition of covariance between RVs X_i and X_j
- The (i, i)-th element of $\mathrm{COV}(\boldsymbol{X})$, i.e., the main diagonal elements of the matrix $\mathrm{COV}(\boldsymbol{X})$, is given by

$$\begin{aligned}[\mathrm{COV}(\boldsymbol{X})]_{ii} &= E\Big[(X_i - E[X_i])^2\Big] \\ &= E\Big[X_i^2\Big] - E\Big[X_i\Big]^2 = \mathrm{VAR}(X_i) \end{aligned} \tag{5}$$

- $\mathrm{COV}(\boldsymbol{X})$ is a symmetric matrix since $[\mathrm{COV}(\boldsymbol{X})]_{ij} = [\mathrm{COV}(\boldsymbol{X})]_{ji}$ holds, which follows from (6) and (7).

Covariance Matrix

- Hence, the entire matrix $\mathrm{COV}(\boldsymbol{X})$ looks like this:

Covariance Matrix

- A positive semi-definite matrix $\boldsymbol{A}$ satisfies

$$\boldsymbol{v}^T \boldsymbol{A} \boldsymbol{v} \geq 0, \quad \forall \boldsymbol{v} \neq \boldsymbol{0} \tag{6}$$

- Fact: The matrix $\mathrm{COV}(\boldsymbol{X})$ is a positive semi-definite matrix.
  Proof:

$$\begin{aligned} \boldsymbol{v}^T \mathrm{COV}(\boldsymbol{X}) \boldsymbol{v} &= \boldsymbol{v}^T E\Big[(\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{X} - E[\boldsymbol{X}])^T\Big] \boldsymbol{v} \\ &= E\Big[\boldsymbol{v}^T (\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{X} - E[\boldsymbol{X}])^T \boldsymbol{v}\Big] \\ &\overset{(a)}{=} E\Big[YY\Big] = E\Big[Y^2\Big] \overset{(b)}{\geq} 0 \end{aligned} \tag{7}$$

  where (a) comes from $Y = v^T(\boldsymbol{X} - E[\boldsymbol{X}]) = (\boldsymbol{X} - E[\boldsymbol{X}])^T \boldsymbol{v}$ and (b) holds since expectation of a squared RV is always non-negative

Normal Random Vector

- Let $\boldsymbol{Z} = [Z_1, Z_2, ..., Z_d]$, were $Z_i \sim N(0,1)$ are i.i.d.
- Then, $E[\boldsymbol{Z}] = \boldsymbol{0} = [0, 0, ..., 0]$ and $\text{COV}(\boldsymbol{Z}) = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix.
- In such case, we use the following notation $\boldsymbol{Z} \sim N(0, \boldsymbol{I})$.
- The PDF of $\boldsymbol{Z}$ is given by

$$f_{\boldsymbol{Z}}(\boldsymbol{z}) = \prod_{i=1}^{d} f_Z(z_i) = \prod_{i=1}^{d} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z_i^2}{2}\right)$$

$$= \frac{1}{(2\pi)^{d/2}} \exp\left(-\frac{\sum_{i=1}^{d} z_i^2}{2}\right) = \frac{1}{(2\pi)^{d/2}} \exp\left(-\frac{||\boldsymbol{z}||_2^2}{2}\right) \quad (8)$$

- Note that $f_{\boldsymbol{Z}}(\boldsymbol{z})$ depends only on the norm of $\boldsymbol{z}$. As a result, if I rotate $\boldsymbol{z}$ by an orthogonal matrix $\boldsymbol{U}$, as $\hat{\boldsymbol{z}} = \boldsymbol{U}\boldsymbol{z}$, then $f_{\hat{\boldsymbol{Z}}}(\hat{\boldsymbol{z}}) = f_{\boldsymbol{U}\boldsymbol{Z}}(\boldsymbol{U}\boldsymbol{z}) = f_{\boldsymbol{Z}}(\boldsymbol{z})$ since $||\boldsymbol{U}\boldsymbol{z}||_2^2 = ||\boldsymbol{z}||_2^2$

Normal Random Vector

- Let $\boldsymbol{X} = [X_1, X_2, ..., X_d]$ be defined as

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{Z} + \boldsymbol{\mu}, \tag{9}$$

where $\boldsymbol{A}$ is a fixed matrix, $\boldsymbol{\mu}$ is a fixed vector, and $\boldsymbol{Z} \sim N(0, \boldsymbol{I})$. Then, $\boldsymbol{X}$ is general Gaussian random vector.

- The mean and covariance of $\boldsymbol{X}$ are given by

$$E[\boldsymbol{X}] = \boldsymbol{\mu} \tag{10}$$

$$\begin{aligned} \mathrm{COV}[\boldsymbol{X}] &= E\Big[(\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{X} - E[\boldsymbol{X}])^T\Big] = E[\boldsymbol{A}\boldsymbol{Z}(\boldsymbol{A}\boldsymbol{Z})^T] \\ &= E[\boldsymbol{A}\boldsymbol{Z}\boldsymbol{Z}^T\boldsymbol{A}^T] = \boldsymbol{A}E[\boldsymbol{Z}\boldsymbol{Z}^T]\boldsymbol{A}^T = \boldsymbol{A}\boldsymbol{I}\boldsymbol{A}^T \\ &= \boldsymbol{A}\boldsymbol{A}^T \end{aligned} \tag{11}$$

Normal Random Vector

- Let us denote $\mathrm{COV}[\boldsymbol{X}]$ as

$$\boldsymbol{\Sigma} = \mathrm{COV}[\boldsymbol{X}] = \boldsymbol{A}\boldsymbol{A}^T \tag{12}$$

  Then, the PDF of $\boldsymbol{X}$, also known as the general multivariate Gaussian distribution is given by

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}\det(\boldsymbol{\Sigma})}\exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})^T}{2}\right), \tag{13}$$

  where $\boldsymbol{\Sigma}^{-1}$ is the inverse of $\boldsymbol{\Sigma}$.
- Note that the general multivariate Gaussian distribution of $\boldsymbol{X}$, $f_{\boldsymbol{X}}(\boldsymbol{x})$, depends on two parameters the mean $\boldsymbol{\mu}$ vector and the covariance matrix $\boldsymbol{\Sigma}$.
- We denote the general Gaussian vector as $\boldsymbol{X} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

Geometrical Interpretation

Geometrically, the distributions of $\boldsymbol{Z}$ and $\boldsymbol{X}$ can be drawn as follows as follows

Principal Component Analysis (PCA)

- Let $\boldsymbol{X} \in \mathbb{R}^d$, where d is very large, has a covariance matrix $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu} = \mathbf{0}$.
- We would like to reduce the dimension of $\boldsymbol{X}$, d, such that we have very little loss of information.
- Ideally, we would like a mapping $F : \boldsymbol{X} \to \boldsymbol{Y}$, where $\boldsymbol{Y} \in \mathbb{R}^k$ and $k \ll d$.
- Note that we did that in the previous lecture, using the Johnson–Lindenstrauss Lemma.
- Note that the Johnson–Lindenstrauss algorithm is blind (does not care) about the statistics of the data.
- Maybe, if we know the statistics of the data, say the covariance matrix, $\boldsymbol{\Sigma}$, we will do a more accurate reduction of dimensions.
- Maybe, if we can reduce the dimension d to $1 \leq k \leq 3$, then we can even visualise the data. Note that we humans cannot visualise above 3 dimensions. This is where PCA is very useful, i.e., as a tool to visualise data.

PCA

- Let us start with the extreme case: Let us reduce the dimension d to one.
- Hence, we need a one dimensional RV Y that best "explains" the d-dimensional random vector $\boldsymbol{X}$.
- Geometrically, this equates to having an optimal line through data points, drawn as:

Principal Component Analysis (PCA)

- Mathematical, we need to find a unit vector $\boldsymbol{v}$ which maximizes the variance of the projection of data points on $\boldsymbol{v}$
- Mathematically, we need to maximize

$$\mathrm{VAR}[\langle \boldsymbol{X}, \boldsymbol{v} \rangle],$$

where

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle = \sum_{i=1}^{d} a_i b_i$$

denotes the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$.

- On the other hand,

$$\begin{aligned} \mathrm{VAR}[\langle \boldsymbol{X}, \boldsymbol{v} \rangle] &= E[\langle \boldsymbol{X}, \boldsymbol{v} \rangle^2] - \langle E[\boldsymbol{X}^T], E[\boldsymbol{v}] \rangle^2 = E[\langle \boldsymbol{X}, \boldsymbol{v} \rangle^2] \\ &= E[\boldsymbol{v}^T \boldsymbol{X} (\boldsymbol{v}^T \boldsymbol{X})^T] = E[\boldsymbol{v}^T \boldsymbol{X} \boldsymbol{X}^T \boldsymbol{v}] = \boldsymbol{v}^T E[\boldsymbol{X} \boldsymbol{X}^T] \boldsymbol{v} \\ &= \boldsymbol{v}^T \boldsymbol{\Sigma} \boldsymbol{v} \end{aligned} \tag{14}$$

Principal Component Analysis (PCA)

- Hence,

$$\max_{\boldsymbol{v}} \text{VAR}[\langle \boldsymbol{X}, \boldsymbol{v} \rangle] = \max_{\boldsymbol{v}} \boldsymbol{v}^T \boldsymbol{\Sigma} \boldsymbol{v} = \lambda_1 \tag{15}$$

  where λ_1 is the largest eigenvalue and the optimal $\boldsymbol{v}$, denoted by $\boldsymbol{v}_1$, that achieves the maximization is the eigenvector corresponding to λ_1.

- Finally, the PCA Algorithm in dimension one: Each data vector $\boldsymbol{X}_i$ is reduced to a point Y_i simply by performing

$$Y_i = \langle \boldsymbol{X}_i, \boldsymbol{v}_1 \rangle$$

Principal Component Analysis (PCA)

- Next, let use reduce the dimension d to two.
- Hence, we need a two dimensional vector $\boldsymbol{Y} = [Y_1, Y_2]$ that best "explains" the d-dimensional random vector $\boldsymbol{X}$.
- Having maximized the variance in one direction, let's find another direction, orthogonal to the first one, which maximizes the variance in that direction.
- This again equates to line through data points, drawn as:

Principal Component Analysis (PCA)

- Mathematically, we need

$$\max_{\boldsymbol{v},\boldsymbol{v}\perp\boldsymbol{v}_1} \text{VAR}[\langle \boldsymbol{X}, \boldsymbol{v}\rangle] = \max_{\boldsymbol{v},\boldsymbol{v}\perp\boldsymbol{v}_1} \boldsymbol{v}^T\boldsymbol{\Sigma}\boldsymbol{v} = \lambda_2 \tag{16}$$

  where λ_1 is the second largest eigenvalue and the optimal $\boldsymbol{v}$, denoted by $\boldsymbol{v}_2$, that achieves the maximization is the eigenvector corresponding to λ_2.

- Finally, the PCA Algorithm in dimension two: Each data vector $\boldsymbol{X}_i$ is reduced to two points $\boldsymbol{Y}_i = [Y_{1i}, Y_{2i}]$ simply by performing

$$\boldsymbol{Y}_i = [Y_{1i}, Y_{2i}] = [\langle \boldsymbol{X}_i, \boldsymbol{v}_1\rangle, \langle \boldsymbol{X}_i, \boldsymbol{v}_2\rangle]$$

Principal Component Analysis (PCA)

- If we continue to reducing d to three dimensions, we have mathematically

$$\max_{\boldsymbol{v},\boldsymbol{v}\perp\boldsymbol{v}_1,\boldsymbol{v}\perp\boldsymbol{v}_2} \text{VAR}[\langle \boldsymbol{X}, \boldsymbol{v}\rangle] = \max_{\boldsymbol{v},\boldsymbol{v}\perp\boldsymbol{v}_1,\boldsymbol{v}\perp\boldsymbol{v}_2} \boldsymbol{v}^T\boldsymbol{\Sigma}\boldsymbol{v} = \lambda_3 \qquad (17)$$

  where λ_3 is the third largest eigenvalue and the optimal $\boldsymbol{v}$, denoted by $\boldsymbol{v}_3$, that achieves the maximization is the eigenvector corresponding to λ_3.

- Finally, the PCA Algorithm in dimension three: Each data vector $\boldsymbol{X}_i$ is reduced to three points $\boldsymbol{Y}_i = [Y_{1i}, Y_{2i}, Y_{3i}]$ simply by performing

$$\boldsymbol{Y}_i = [Y_{1i}, Y_{2i}, Y_{3i}] = [\langle \boldsymbol{X}_i, \boldsymbol{v}_1\rangle, \langle \boldsymbol{X}_i, \boldsymbol{v}_2\rangle, \langle \boldsymbol{X}_i, \boldsymbol{v}_3\rangle]$$

- This approach can be continued straightforwardly to any dimension $k \leq d$

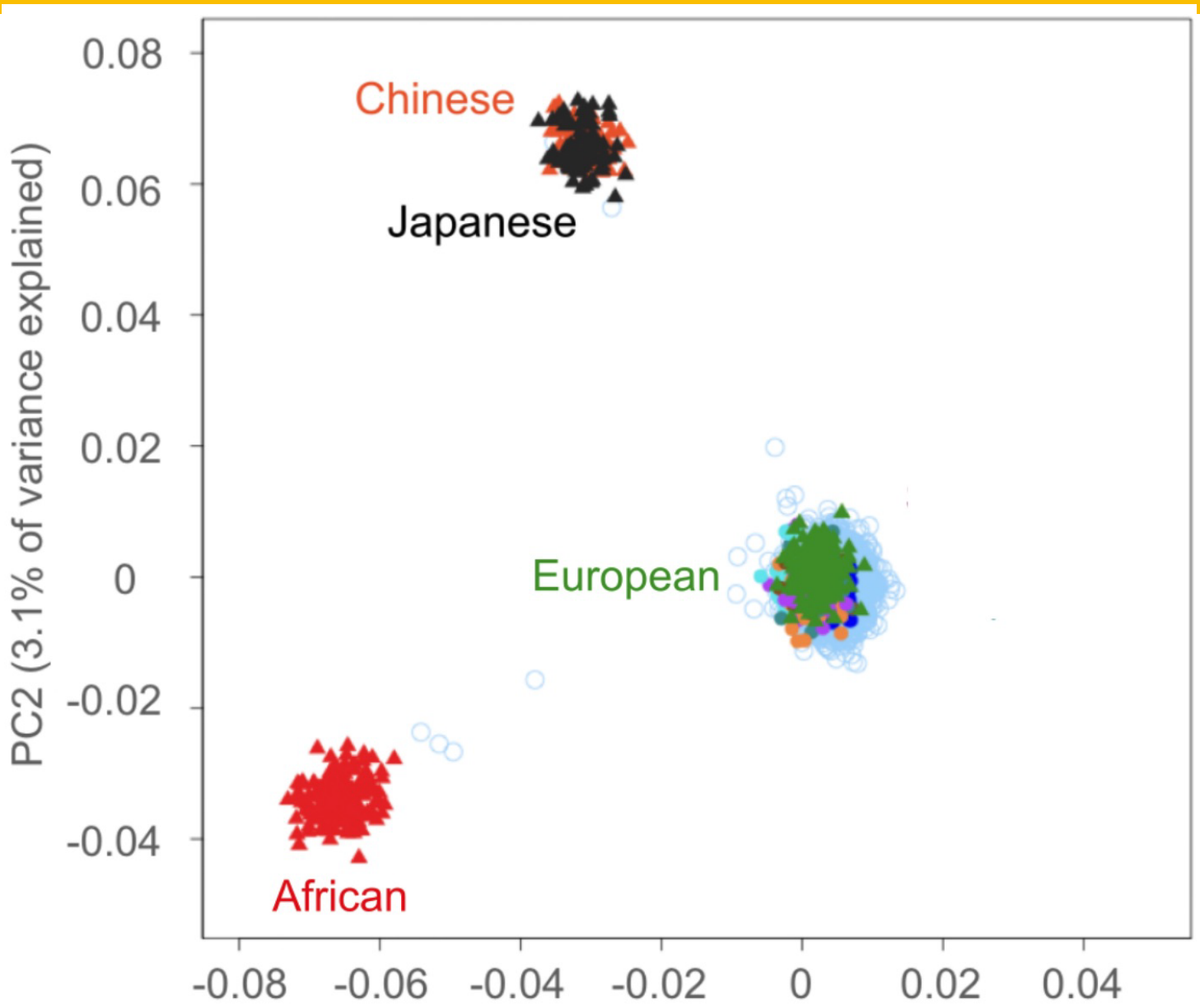
Principal Component Analysis (PCA)

- The General PCA Algorithm: To reduce the dimension of $\boldsymbol{X} \in \mathbb{R}^d$ from d to k, and thereby obtain $\boldsymbol{Y} \in \mathbb{R}^k$, we project $\boldsymbol{X}$ into the subspace defined by the k eigenvectors of the covariance matrix $\boldsymbol{\Sigma}$ that correspond to the k-th largest eigenvalues of $\boldsymbol{\Sigma}$.
- PCA is usefull since it maximizes the variability of $\boldsymbol{X} \in \mathbb{R}^d$ in $k \leq d$ dimensional space.

Principal Component Analysis (PCA)

- What PCA tells us about the covariance matrix $\boldsymbol{\Sigma}$ is that the spectrum of the matrix, i.e., the distribution of its eigenvalues, is a hidden information that resolves the information in $\boldsymbol{X}$ into sub-components expressed by the eigenvalues of the covariance matrix $\boldsymbol{\Sigma}$.
- One very important fact is that to do PCA, using he shown algorithm, the covariance matrix, $\boldsymbol{\Sigma}$, has to be known.

Example: PCA of Human Genomes



Principal Component Analysis (PCA)

- But, how accurate is the PCA, when the covariance matrix $\mathbf{\Sigma}$ is unknown but needs to be estimated from the data itself?
- How close should the estimated covariance matrix $\hat{\mathbf{\Sigma}}$ and the actual covariance matrix $\mathbf{\Sigma}$ need to be in order for PCA based on $\hat{\mathbf{\Sigma}}$ to be an approximate PCA on $\mathbf{\Sigma}$?
- Maybe, we can never have enough samples to estimate $\mathbf{\Sigma}$ accurately enough such that PCA based on $\hat{\mathbf{\Sigma}}$ to be a approximate PCA on $\mathbf{\Sigma}$?
- Maybe the genome image we showed does not have any connection to reality due to the low number of samples from which the covariance matrix was estimated?