+ linear algebra:

- defining an objective expression with data (N x M) and models / designing algorithm to find an optimal solution. (model parameters)
- a tool to understand the geometric shape of high-dimensional data embeddings within the ML models. (approximation by computing the two statistics: mean and covariance matrix and its spectral decomposition)

CS 461: Machine Learning Principles

Class 3: Sept. 12
Linear Algebra
(Symmetric Matrix, Spectral/ SVD Decomposition)

Instructor: Diana Kim

Spectral Decomposition of Symmetric Matrix
(covariance matrix for ML)

Spectral Decomposition of Symmetric Matrix

$+ \Sigma = E \Lambda E^t$

$$\Sigma = \begin{bmatrix} | & & \\ e_1, e_2, \ldots e_n \\ | & & \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & & & \\ 0 & \ldots & \ldots & \lambda_n \end{bmatrix} \cdot \begin{bmatrix} e_1^t \\ e_2^t \\ \ldots \\ e_n^t \end{bmatrix}$$

$$\Sigma = \lambda_1 \cdot e_1 e_1^t + \lambda_2 \cdot e_2 e_2^t + \ldots + \lambda_n \cdot e_n e_n^t$$

- Σ: symmetric matrix
- e_i:eigenvectors, ${e_i}^t \cdot e_j = 0 \; , \; i \neq j$
- λ_i are eigenvalues

Note)

$$A \cdot \Lambda = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ & & \vdots & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & & \\ 0 & \cdots & \cdots & \lambda_n \end{bmatrix}$$

$+ [\lambda_1\, a(1), \lambda_2\, a(2), \ldots, \lambda_n\, a(n)], a(i): (i)\ th\ column\ \ vector\ of\ A$

$$\Lambda \cdot A = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & & \\ 0 & \cdots & \cdots & \lambda_n \end{bmatrix} \cdot \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ & & \vdots & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

Spectral Decomposition of Symmetric Matrix

Proof)

- $\Sigma\, e_i = \lambda_i e_i$

$+\,\Sigma\, E = E\,\Lambda$

+ and E *is an orthonormal matrix*

- $$e_j^t \Sigma e_i = \lambda_i e_j^t e_i$$
  $$e_i^t \Sigma e_j = \lambda_i e_i^t e_j$$

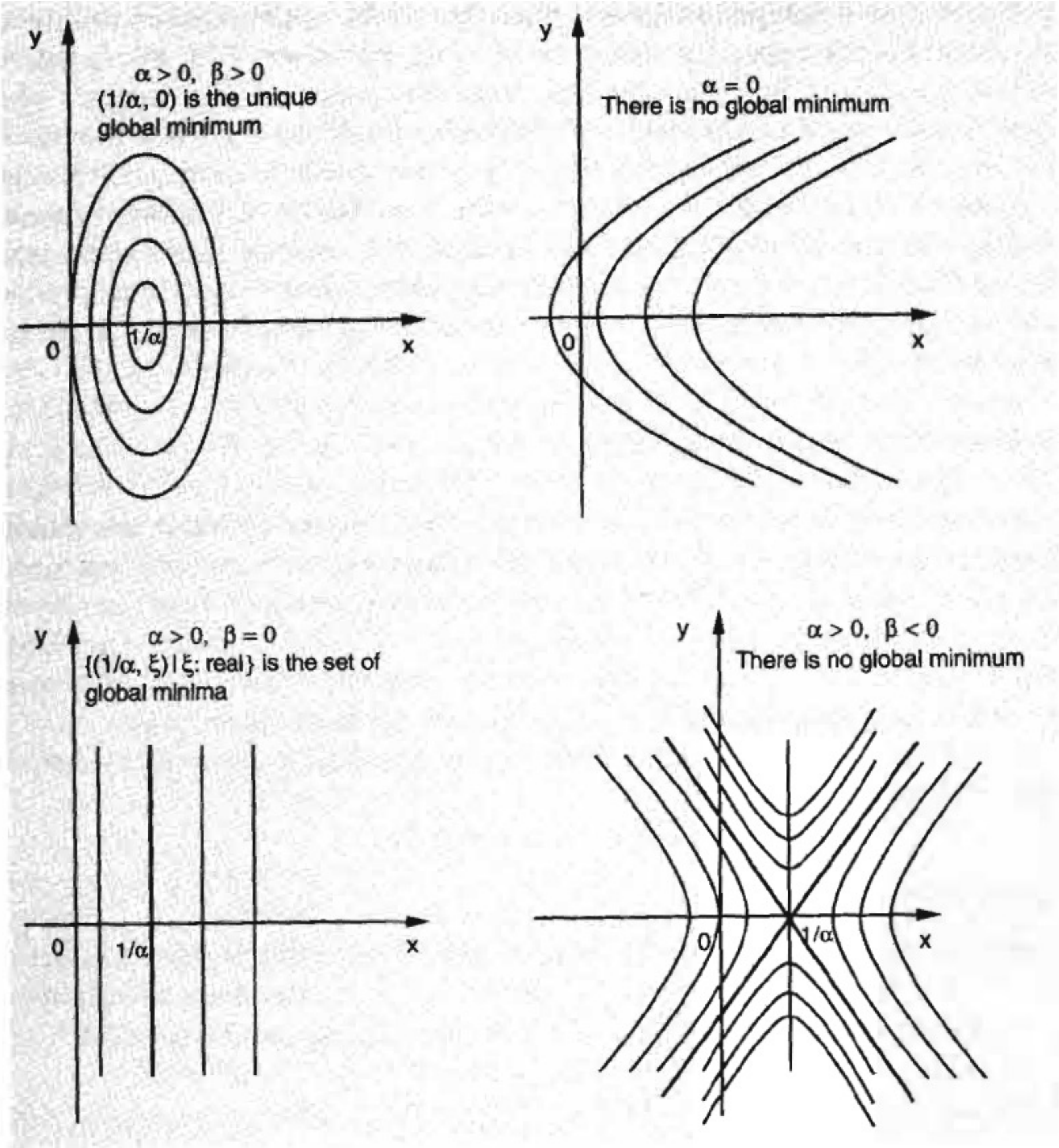
The Geometric Shape of Jointly Gaussian Density

$$f_X(\vec{x}) = \frac{1}{\sqrt{(2\pi)^N |\Sigma|}} \exp^{-\frac{1}{2}(\vec{x}-\vec{\mu})^t \Sigma^{-1} (\vec{x}-\vec{\mu})}$$

Quadratic Function of $\vec{X}$

$$f(x) = x^t A x + bx + c$$

The symmetric matrix A tells us the behavior of the quadratic function.



$$f(x, y) = \frac{1}{2}(\alpha x^2 + \beta y^2) - x$$

Figure, 1.1.4.
From Non-Linear Programming by Dimitri P. Bertsekas.

We will focus on specifically Covariance Matrix.

- Symmetric

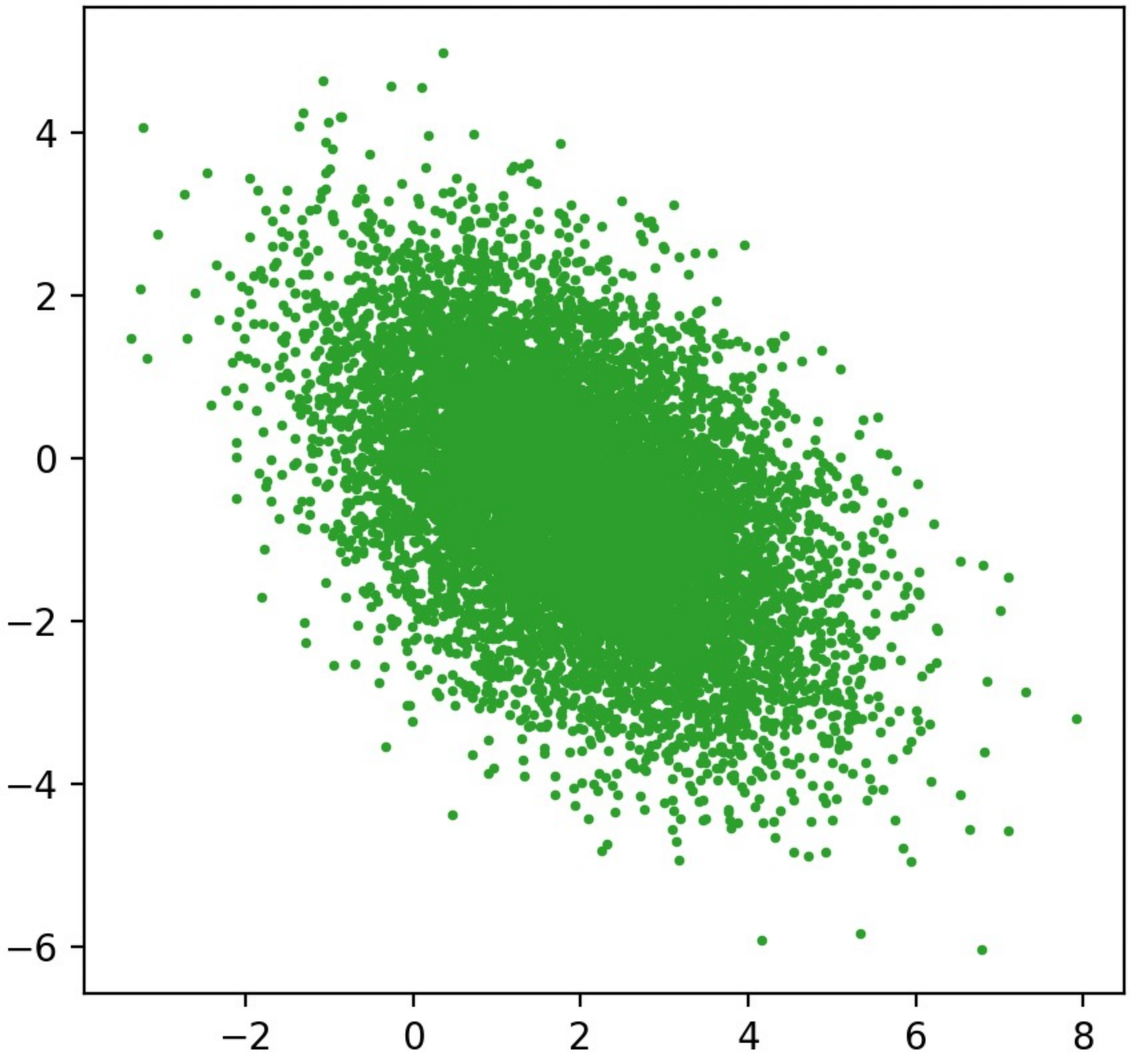
$$\Sigma = E[(\vec{X} - \vec{\mu})(\vec{X} - \vec{\mu})^t]$$
$$= E[\vec{X}\vec{X}^t] - \mu\mu^t$$

- Positive semi–definite
  (non-negative definite)

$$x^t \Sigma x \geq 0 \quad \forall x$$

+ Covariance matrix's eigenvalues are ≥ 0

Gaussian Samples & its Covariance Matrix



$$\Sigma = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}$$

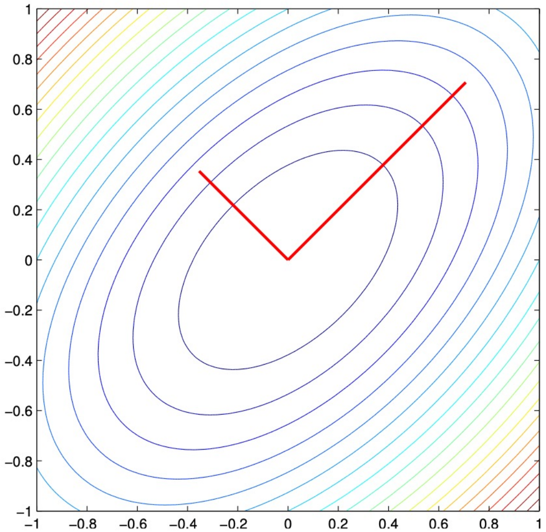
+ the covariance matrix shows
how the data is dispersed (which direction/ variation)

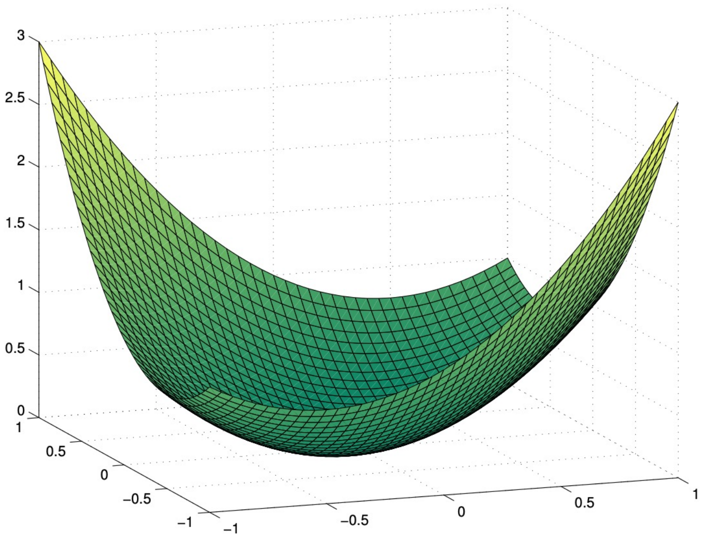
Multivariate Gaussian Density $\vec{X}$ is expressed in the quadratic form but with $\mathbf{\Sigma^{-1}}$

$$f_X(\vec{x}) = \frac{1}{\sqrt{(2\pi)^N |\Sigma|}} \exp^{-\frac{1}{2}(\vec{x}-\vec{\mu})^t \Sigma^{-1} (\vec{x}-\vec{\mu})}$$

Note (quadratic vs. Gaussian's inverse quadratic)

$$x^t \Sigma x \quad \text{vs.} \quad x^t \Sigma^{-1} x$$



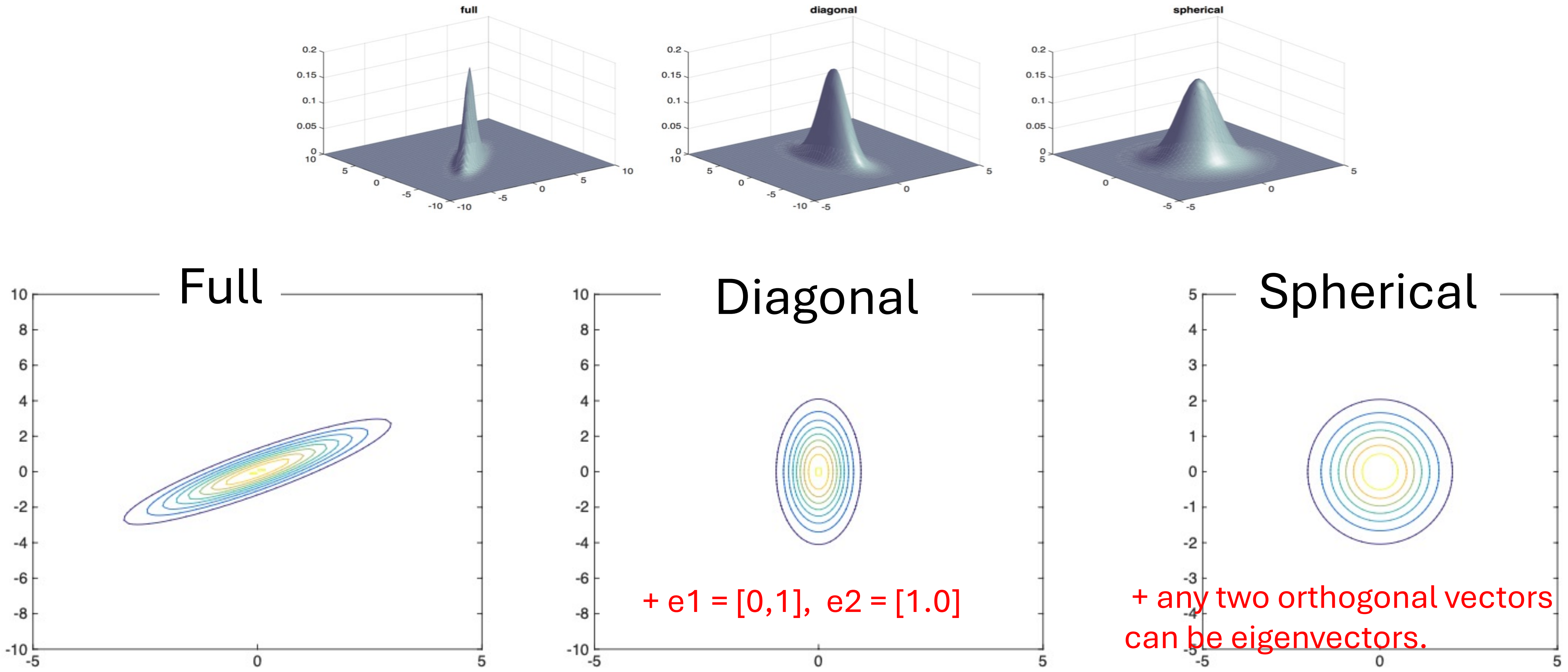


$$\Sigma = x^t \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} x$$

From https://users.oden.utexas.edu/~omar/inverse_probs/numopt.pdf

Quadratic function dispersion is proportional inversely to eigenvalues.
The eigenvalue represents the curvature in the direction of the eigenvector.

We can guess the eigenvector & eigenvalues from the gaussian contours.



From Murphy Figure 3.5 and 3.6. "An Introduction"

Q: Can we design a new Gaussian random vector $Y \sim N(0, \Sigma^*)$ from $X \sim N(0, I)$?

- Y = AX

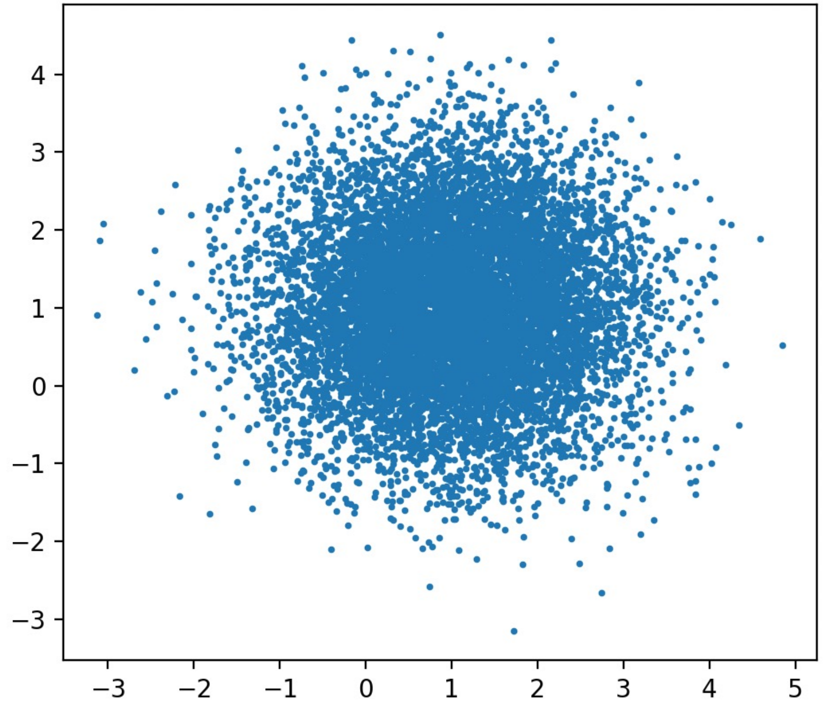
- COV (Y) =

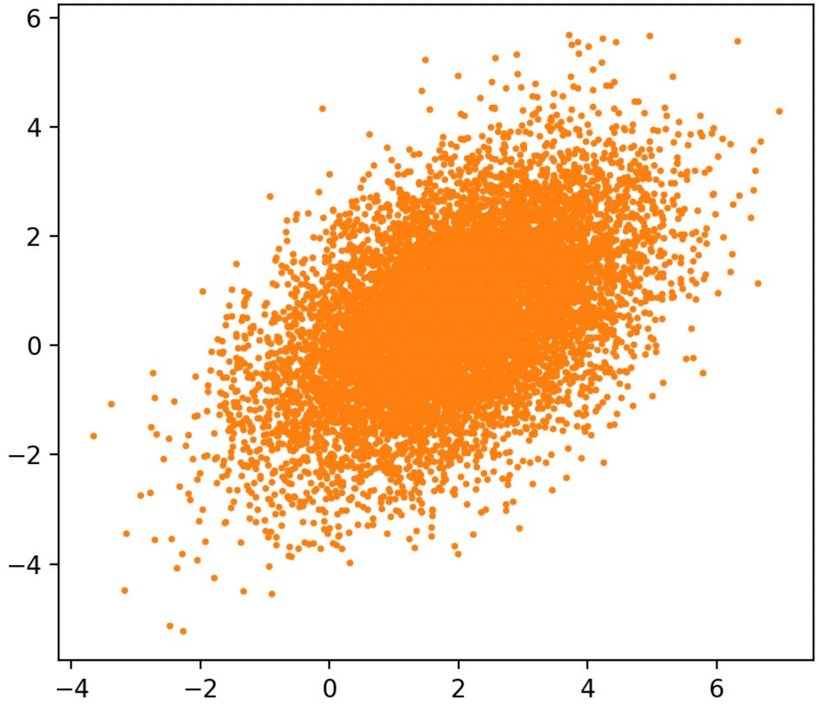
- What A will be? + $A = E\,\Lambda^{1/2}$ $where\ \Sigma^* = E\,\Lambda^{1/2}\Lambda^{1/2}E^t$

Q: Can we design a new Gaussian random vector $Y \sim N(0, \Sigma^*)$ from $X \sim N(0, I)$?

$$\Sigma^* = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix}$$

Generating Correlated Samples from Isotropic Gaussian





Singular Vector Decomposition of the Matrix

Singular Vector Decomposition (a Rectangular Matrix)

$$A(n \times m) = \begin{bmatrix} & | & \\ u_1, u_2, & & ...u_n \\ & | & \end{bmatrix} \cdot \begin{bmatrix} \sqrt{\lambda_1} & 0 & ... & 0... \\ 0 & \sqrt{\lambda_2} & ... & 0...0 \\ \vdots & & & \\ 0 & ... & ... & \sqrt{\lambda_m} \\ \vdots & & & \\ 0 & ... & ... & 0 \\ 0 & ... & ... & 0 \end{bmatrix} \cdot \begin{bmatrix} v_1^t \\ v_2^t \\ ... \\ v_m^t \end{bmatrix}$$

$$A^t(m \times n) = \begin{bmatrix} & | & \\ v_1, v_2, & & ...v_m \\ & | & \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 & 0 & ... & 0... & 0 & 0 \\ 0 & \lambda_2 & ... & 0... & 0 & 0 \\ \vdots & & & & & \\ 0 & ... & ... & \sqrt{\lambda_m} & ... & 0 \end{bmatrix} \cdot \begin{bmatrix} u_1^t \\ u_2^t \\ ... \\ u_n^t \end{bmatrix}$$

$$AA^t = \begin{bmatrix} & | & \\ u_1, u_2, & ...u_n \\ & | & \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 & 0 & ... & 0 & ... & 0 & 0 \\ 0 & \lambda_2 & ... & 0 & ... & 0 & 0 \\ \vdots & 0 & ... & \lambda_m & & & \\ 0 & ... & 0 & ... & ... & 0 & 0 \\ 0 & ... & 0 & ... & ... & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} u_1^t \\ u_2^t \\ ... \\ u_n^t \end{bmatrix}$$

$$A^T A = \begin{bmatrix} & | & \\ v_1, v_2, & ...v_m \\ & | & \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 & 0 & ... & 0 \\ 0 & \lambda_2 & ... & 0 \\ \vdots & & & \\ 0 & ... & ... & \lambda_m \end{bmatrix} \cdot \begin{bmatrix} v_1^t \\ v_2^t \\ ... \\ v_m^t \end{bmatrix}$$

Singular Vector Decomposition (a Rectangular Matrix)

proof)

+ by using spectral decomposition

$$AA^t = U\Lambda U^t = U^{'}\Lambda^{'}U^{'t}$$

$$U'^t AA^t U' = \Lambda^{'}$$

$$(A^t U')^t (A^t U') = \Lambda^{'}$$

SVD application: Solving over-determined system

$$D \cdot w = y$$

$$where\ D\ (n \times m)\ \text{is data matrix}$$

- no solution

$$D^t D \cdot w = D^t y$$

$$where\ D\ (n \times m)\ \ \text{is data matrix}$$

- may exist a solution but found on a projected space

- $w = V\ \Lambda\ (rectangular\ form) E^t\ y$ $(by\ using\ SVD)$
- $w = \ (D^t D)^{-1} D^t y$ (by using inverse)
- both solutions are same, please try to prove it.

$$D^t D \cdot w = D^t b$$

$$where\ D\ (n \times m)\ \text{ is data matrix}$$

- May exist a solution but the solution on a projected space (approximation)
- what if $\mathrm{D^tD}$ does not have inverse?
- $\mathrm{i.e\ it's\ spectral\ decomposition\ contians\ zero\ eigenvalues.}$