Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

1. Explain in your own words what effect does the choice of Covariance matrix have on the Bivariate Gaussian (compare spherical, elliptical). What does it mean when the covariance matrix is not diagonal?

2. Do the linear transformation assignment

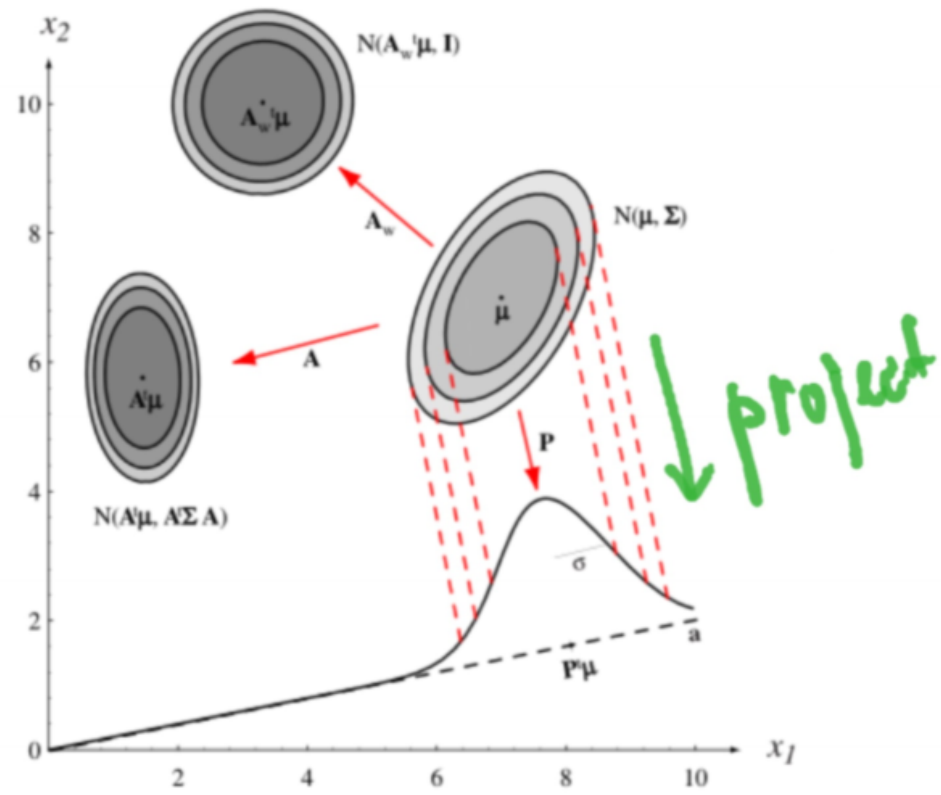


Figure 2.8: The action of a linear transformation on the feature ace will convert an arbitrary normal distribution into another normal distribution. One transformation, **A**, takes the source distribution into distribution $N(\mathbf{A}^t\mu, \mathbf{A}^t\Sigma\mathbf{A})$. Another linear transformation — a projection **P** onto line **a** — leads to $N(\mu, \sigma^2)$ measured along **a**. While the transforms yield distributions in a different ace, we show them ser-imposed on the original $x_1 - x_2$ ace. A whitening transform leads to a circularly symmetric Gaussian, here shown dilaced.

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

- Consider a random vector $\vec{x} = [x_1, x_2, \ldots, x_n]$ and a linear transformation

  $f(\vec{x}) = A\vec{x} + \vec{b} := \vec{y}$

  Assume that $\vec{x} \sim \mathcal{N}(\vec{\mu}_{\vec{x}}, \Sigma_{\vec{x}})$ and $\vec{b} \sim \mathcal{N}(\vec{0}, \Sigma_{\vec{b}})$.

TASKS: ① * What is the distribution of $f(\vec{x})$?
- Find the parameters of this distribution

② * What is the distribution of $\vec{z} = [\vec{x}, \vec{y}]^T$
- Find the parameters of this distribution

HOW TO DO
- Its your choice whether to do it in PYTHON or by hand, but the solution by har

3. What is the meaning of Mahalanobis distance? What is the relation of this to the eigenvalues of the Covariance matrix? Draw a sketch either in Python or by hand for the Bivariate case (K=2)

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

Solution:

1.

Covariance on ~~G~~ Bivariate Gaussian:

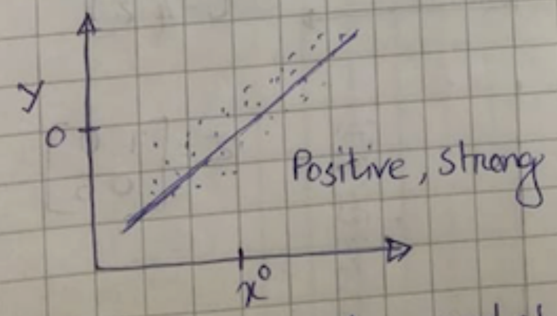


Fig: The diagonal spread of the data is captured by the covariance

It is called the covariance of data.

$$\sigma(x,y) = E[(x-E(x))(y-E(y))]$$

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

7

Covariance Matrix for Bivariate (x, Y)

$$\Sigma = \begin{bmatrix} \sigma(x,x) & \sigma(x,y) \\ \sigma(y,x) & \sigma(y,y) \end{bmatrix}$$

If, x is positively correlated with y, y also does same with x.

So,

$$\sigma(x,y) = \sigma(y,x)$$

Now, we can see the matrix is a symmetric matrix with variances in diagonal position while covariance are off-diagonal.

Thats why Bivariate Gaussian is expressed by its mean and its 2×2 covarince matrix.

Similiarly, 3×3 is for 3 dimensional

n×n is for n dimensional

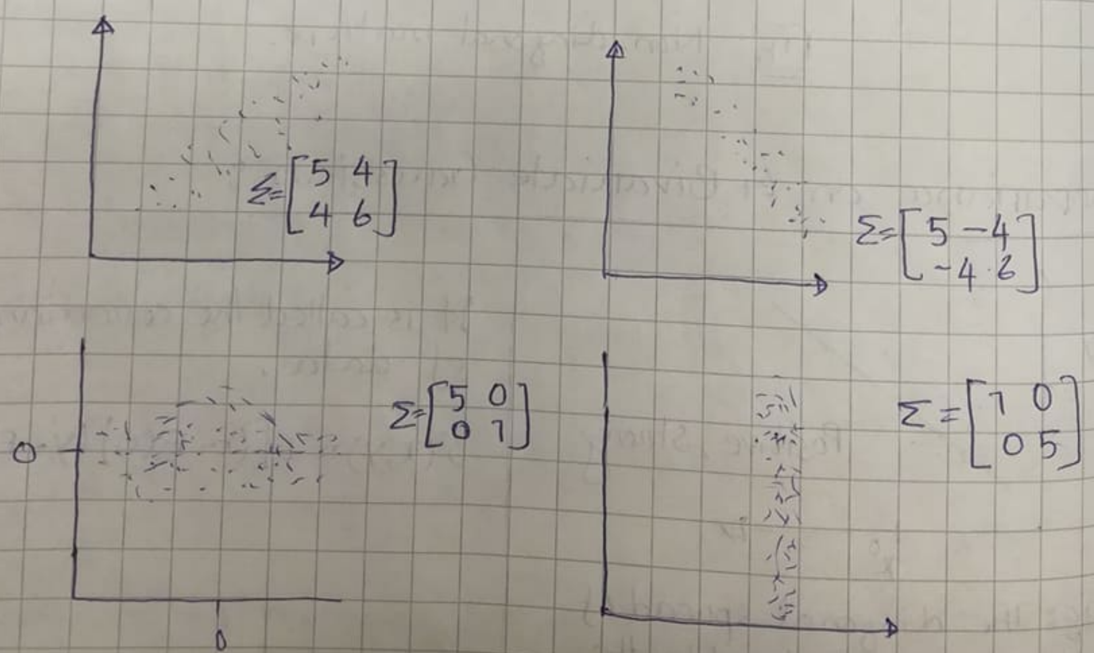


Fig 8: Visual illustration of the overall shape of the data defines the covariance matrix for Bivariate.

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

2.

2. $x \sim N(\mu_x, \Sigma_x)$ and $Y = Ax + b$ where $b \sim N(0, \Sigma_b)$

Since x and b is from normal distribution, Y and $(x^T, Y^T)^T$ are also from normal distribution.

To find the parameters we only need to know the its mean and variance.

Now, $Y = Ax + b$

mean, $E(Y) = A\,E(x) + E(b) = A\,E(x) = A\mu_x$

variance, $\Sigma_Y = E[Y - E(Y)]^T (Y - E(Y)] + var(b)$

$= E[(Ax + b - AE(x) - b)^T (Ax + b - AE(x) - b)] + \Sigma_b$

$= E[(Ax - AE(x)^T (Ax - AE(x)] + \Sigma_b$

$= A^T E[(x - E(x)^T (x - E(x))] A + \Sigma_b$

$= A^T \Sigma_x A + \Sigma_b$

We can say that $Y = Ax + b$ is a normal ~~distribution~~ random vector with mean $A\mu_x$ and variance $A^T\Sigma_x A + \Sigma_b$

The joint distribution of $(x, Y)^T$ is also normal distribution.

Its mean, $\mu_{xy} = \begin{pmatrix} \mu_x \\ A\mu_x \end{pmatrix}$

$cov(x, Y) = E[xY^T] - E[x]E[Y^T]$

$= E[x(Ax+b)^T - \mu_x A^T \mu_x^T$

$= E[xAx + bx]^T - \mu_x A^T \mu_x^T$

$= E[xx^T]A^T - \mu_x A^T \mu_x^T = \Sigma_x A^T$

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

Now, $\text{cov}(Y, x) = E[x^T Y] - E[x^T]\, E[Y]$

$$= E[x^T(Ax+b) - \mu_x^T A \mu_x]$$

$$= E[x^T A x - \mu_x^T A \mu_x]$$

$$= E[x^T x - \mu_x^T \mu_x]A = \Sigma_x A$$

In this deviation we use that the covariance between X and b are zero. Now the variance matrix is,

$$\Sigma_{xy} = \begin{bmatrix} \text{var}(x) & \text{cov}(x,y) \\ \text{cov}(y,x) & \text{var}(y) \end{bmatrix}$$

$$= \begin{bmatrix} \Sigma_x & \Sigma_x A^T \\ A\Sigma_x & A^T \Sigma_x A + \Sigma_b \end{bmatrix}$$

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

3.

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

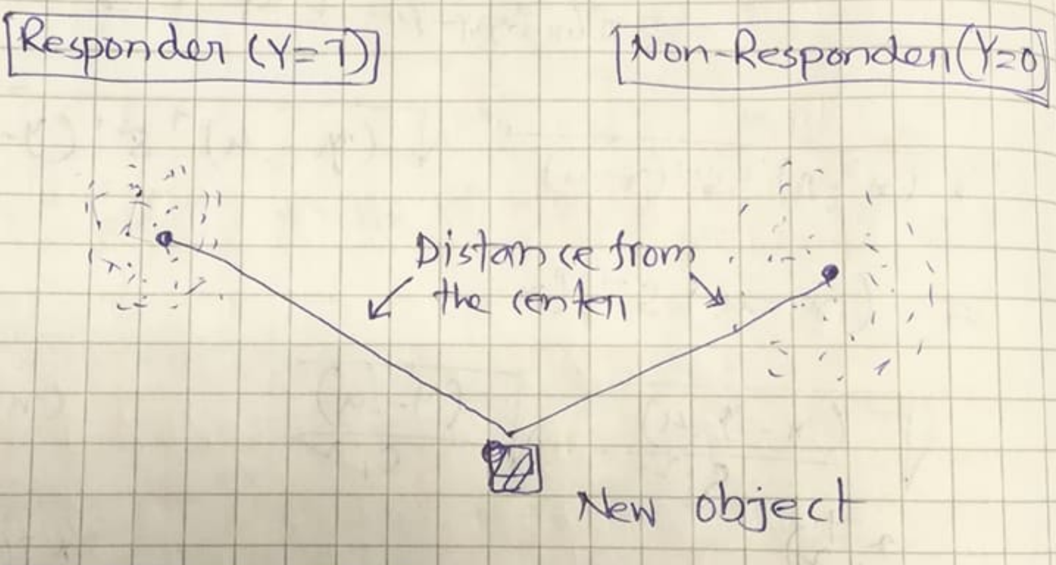
Mahalanobis Distance

Content

1. What it is?
2. Where it is used?
3. Issues with Euclidian distance?
4. Mahalanobis Distance formula
5. How does it get rid of some of the issues of Euclidian distance?

Where it is used?

- In classification problem.



Euclidean Distance

The distance between Responders' center or Non-responders' center (i) and New Object (j).

Euclidean distance =

$$d(i,j) = \sqrt{(|x_{i_1} - x_{j_1}|)^2 + (|x_{i_2} - x_{j_2}|)^2 + \dots + (|x_{i_n} - x_{j_n}|)^2}$$

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

(u_1, v_1) (u_2, v_2) 2

D = Distance

$$D = \sqrt{(u_1 - u_2)^2 + (v_1 - v_2)^2}$$

Properties:

(I) $d(i,j) \geq 0$ Always positive

(II) $d(i,i) = 0$ Distance from same object is 0

(III) $d(i,j) = d(j,i)$ Both ways should be same

(IV) $d(i,j) \leq d(i,k) + d(k,j)$

↳ Distance between two points will be shorter always in straight line. Any object lie on the ~~stao~~ middle of straight line is ~~equal~~: the distances are equal.

Euclidean Distance Issues:

A

Income (K)	Lot size
75.0	19.6
25.0	20.6
35.0	27.6

B

Income	Lot size
75,000	19.6
25,000	20.6
35,000	27.6

C

Income (K)	Lot size
75.0	19600
25.0	25,000
35.0	35,000

→ Due to scale of ~~data~~ these data the Eclidean Distances are different.

→ Highly dependent on Scale

→ So solving the scaling problem we use ~~stand~~ stand-ardization $(x-\mu)/\sigma$

x = Data Point

μ = Mean

σ = std

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

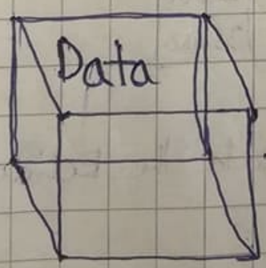
But

- Even after the standardization one more issue is remaining
- Some of the variables ~~are~~ have collinearity.
- If there is the collinearity / ~~co~~ relation, ~~we might add the~~ then don't you think, you are counting same impact multiple times.
- So, you need a better method - which gets rid of scaling impact as well as collinearity impact of variables.
- Mahalanobis distance is the technique.

Mahalanobis Distance:

Mahalanobis distance measure does the following:

- it transforms the covariable into uncorrelated variables
- and makes their varriances equal to 1
- and then it calculates simple Euclidean distance.



Ma. Dis Does →

(I) Removes Collinearity
(II) Transforms them to uncorrelated
(III) Makes the variance 1
(IV) Calculates the Euclidean Distance Finally.

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

Formula:

$$D^2 = (x-m)^T C^{-1} (x-m) \quad \text{—(1)}$$

Where,

D^2 = Mahalanobis Distance

x = Vector of data

m = Vector of mean of independent variables

C^{-1} = Inverse Covariance matrix of independent variables

T = Indicates vector should be transposed.

	x_1	x_2	x_3	x_4	x_5
x_1	Var (x_1,x_1)	Cov	Cov	Cov	Cov
x_2		Var (x_2,x_2)	"	"	"
x_3	Cov	Cov	Var (.)	Cov	Cov
x_4	"	"	Cov	Var (x_4,x_4)	
x_5	"	"	Cov	Cov	Var.

Fig: Covariance Matrix

→ The diagonal is Variance

→ Rest of ~~di~~ → Covariance

From the Equation (1)

x = vector of data

for Multivariate,

$$X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} \text{ Vector} \qquad \cancel{M} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \vdots \\ \mu_n \end{bmatrix} \text{ ~~vector~~}$$

Data ~~Variab~~   ~~Mean~~

M = Mean

$[\mu]$ vector

From (1) eqn

$(x-m)$ = Distance from ~~mean~~ mean

$$C^{-1} = \frac{1}{\text{Covariance Matrix}}$$

Isn't it standardization?

→ Standardization $= (x-\mu)/\sigma$

→ Multivariate standardization $= (X\text{metrics} - \mu\text{metrics}) / \text{Var-cov metrics}$.

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

5

Covariance Value in ~~D~~ Mahalanobis Distance:

→ Covariance Value = High ↪ Correlation high.

→ Covariance Value = Low ↪ Correlation low.

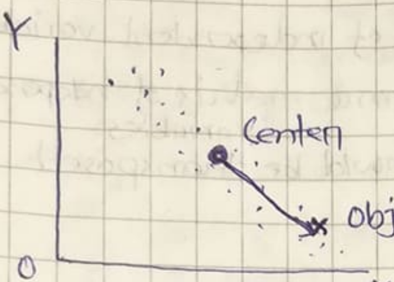


Fig: High Covariance

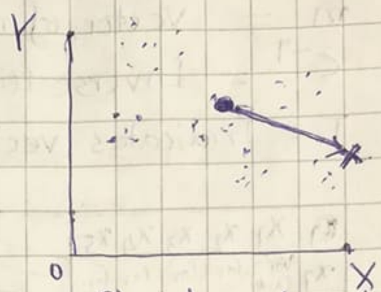


Fig: Low Covariance

$\frac{\cancel{x-a}\;\; x-m}{C \text{ (good/high number)}}$ = low value

$\frac{x-m}{C \text{ (bad/low)}}$ = high value

▨ Eigen Values of Covariance Matrix & Mahalanobis Distance:

$\begin{bmatrix} 5 & 0 \\ 0 & 1 \end{bmatrix}$ ← Diagonal Matrix

→ Covariance = 0 (zeros)

So, this means the variances must be equal to the Eigen-values λ (lamda).

$\begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}$ ← Non diagonal Matrix

→ Covariance = Non-zero

The eigen values represents the variance magnitude in the direction of largest spread of data

Eigen values $\propto$ Variance $\propto$ Mahalanobis Distance.

Bivariate Gaussian + Linear Transformation of Gaussians

19.09.2019

Diagonal Covariance Matrix :

When the covariances of a matrix are ~~is~~ zero (0) then that is diagonal covariance matrix.

$$\begin{bmatrix} 5 & 0 \\ 0 & 1 \end{bmatrix}$$

Fig: Diagonal Covariance Matrix

Non-diagonal Covariance Matrix :

When the covariances of a matrix are not zero (0) then that is called non-diagonal matrix.

$$\begin{bmatrix} 5 & 1 \\ 1 & 1 \end{bmatrix}$$

Fig: Non-diagonal matrix.