10907 Pattern Recognition

Lecturers
Prof. Dr. Thomas Vetter ⟨thomas.vetter@unibas.ch⟩
Prof. Dr. Ivan Dokmanić ⟨ivan.dokmanic@unibas.ch⟩
Dr. Dennis Madsen ⟨dennis.madsen@unibas.ch⟩
Dana Rahbani ⟨dana.rahbani@unibas.ch⟩

Tutors
Martin Ramm ⟨martin.ramm@unibas.ch⟩
Viktor Gsteiger ⟨v.gsteiger@unibas.ch⟩

Exercise 1 — Normal Distributions

Deadline **10.10** Upload .ZIP on Adam.

The exercises are done in **groups of 2 students**. **Only upload 1 version** of the exercise on Adam and **specify your group partner.**

Upload in .zip format containing **1 .pdf file** with your answers to the questions and the **code folder** files (do NOT include the data folder). Create the .zip file with the `createSubmission.py` script.

DO NOT send us pictures of your hand-written exercise - if you do the computations by hand, then properly scan the pages such that they have a white background!

-0.5 Point for uploading in wrong format!

1 Compute by hand [7p]

1.1 Multivariate Normal Distribution [3p]

Consider a bivariate (2D multivariate) normal population with $\mu_1 = -2, \mu_2 = 1, \sigma_1^2 = 6, \sigma_2^2 = 6$, and with cross correlation coefficient, $\rho_{12} = -\frac{1}{2}$.

1. Expand the full probability density (simplify as much as possible) [1p]
2. Determine the main axes and sketch the constant-density contour at one standard deviation [2p]

1.2 Independence [2p]

Consider $\boldsymbol{X} = [X_1, X_2, X_3]^\intercal$ distributed according to $\mathcal{N}(\boldsymbol{X} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$ with

$$\boldsymbol{\mu} = \begin{bmatrix} -3 \\ 1 \\ 4 \end{bmatrix}, \qquad \boldsymbol{\Sigma} = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 6 & -3 \\ 0 & -3 & 6 \end{bmatrix}.$$

Which of the following pairs of random variables are independent? Explain.

1. X_3 and X_1 [0.5p]
2. X_3 and X_2 [0.5p]
3. $2X_1 - X_2 - X_3$ and $X_3 - X_2$ [1p]

1.3 Conditional Distribution [1p]

Calculate the conditional distribution of X_1, given that $X_2 = x_2$ in the joint distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Compare the conditional distribution $P(X_1 \mid X_2 = 5)$ to the marginal distribution $P(X_1)$ in a plot.

$$\boldsymbol{\mu} = \begin{bmatrix} -2 \\ 10 \end{bmatrix}, \qquad \boldsymbol{\Sigma} = \begin{bmatrix} 6 & -3 \\ -3 & 6 \end{bmatrix}$$

1.4 Bayes Classification [1p]

Classify a point $\mathbf{X} = [-2.6, -2]$ into one of two classes, where each class follows a normal distribution with parameters $\boldsymbol{\mu_1} = [-4, -2]$ and $\boldsymbol{\mu_2} = [-1, -2]$ and

(a) isotropic and identical covariance matrices.

(b) covariance matrices:

$$\mathbf{\Sigma_1} = \begin{bmatrix} 1.5 & 1.8 \\ 1.8 & 6 \end{bmatrix}, \qquad \mathbf{\Sigma_2} = \begin{bmatrix} 1.5 & 0.6 \\ 0.6 & 0.6 \end{bmatrix}.$$

Hint: As nothing is stated about the prior probability, equal prior probabilities should be assumed.

2 Coding in Python - Maximum Likelihood implementation [3p]

Estimation of dataset distributions for a 2D toy dataset example. The main function for this exercise is found in the file `ex1-ML_1_Toy.py`. You need to fill in the `TODO` sections in the file. Do NOT use build in *cov*, *mean*, *inv*, *det* functions from numpy - you need to implement this functionality!

1. Estimate the Multivariate normal distribution (MVND) for each dataset (compute sample mean and sample covariance matrix as we assume the data is gaussian distributed). What is the sample mean and covariance matrix of each of the datasets? [1p]
2. Implement the probability density function (pdf) of the 2-dimensional MVND. Also implement and make use of the helper functions: matrix-inverse and matrix-determinant. How many points from the test set are classified correctly? [1p]
3. Send the final plot that the code outputs where 10 random points are sampled from each distribution (details below on how this can be done) [1p].

Sample from the 2 distributions (see function `sample_from_2d_gaussian`). The only random generator you are allowed to use is the uniform distribution $\mathcal{U}(0, 1)$ (`np.random.random()`). These samples need to be transformed into samples from a standard normal distribution $\mathcal{N}(0, 1)$. This is can be achieved with the *Box-Muller transform*:

$$z_0 = \sqrt{-2\log(u_0)}\cos(2\pi u_1)$$
$$z_1 = \sqrt{-2\log(u_0)}\sin(2\pi u_1)$$

with u_0 and u_1 being samples from a uniform distribution. Finally the 2d vector sampled from a MVND needs to be computed. To do so, we can compute:

$$\boldsymbol{x}_{sample} = L\boldsymbol{z} + \boldsymbol{\mu}, z_i \sim \mathcal{N}(0, 1)$$

where L is the matrix factorization of the covariance matrix:

$$\Sigma = LL^T$$

which can be obtained with the Cholesky decomposition of the covariance matrix Σ.