**AY: 2024-2025**
EXAM | **AI-ECUE221**
**June 2025**

**M1-S2: Dept. of Electrical Engineering**
**Teacher: A. Mhamdi**
**Time Limit: $1^1/_2$ h**

This document contains **11** pages numbered from **1/11** to **11/11**. As soon as it is handed over to you, make sure it is complete. The **3** tasks are independent and can be treated in the order that suits you.

The following rules apply:

❶ **A handwritten double-sided A4** sheet is permitted.

❷ **Any electronic material**, except basic calculator, is prohibited.

❸ **Mysterious or unsupported answers** will not receive full credit.

❹ **Round results** to the nearest thousandth *(i.e., third digit after the decimal point)*.

❺ **Task N$^{\underline{o}}$3:** Each correct answer will grant a mark with no negative scoring.

## Task N$^{\underline{o}}$1

⌛ 35mn | ($7^1/_2$ points)



Consider a binary classification problem in $\mathbb{R}^2$ with the following dataset consisting of five points:

$$D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), (\mathbf{x}_3, y_3), (\mathbf{x}_4, y_4), (\mathbf{x}_5, y_5)\}$$

where:

$$\begin{aligned}
\mathbf{x}_1 &= (1, 2), & y_1 &= +1\\
\mathbf{x}_2 &= (2, 3), & y_2 &= +1\\
\mathbf{x}_3 &= (3, 3), & y_3 &= +1\\
\mathbf{x}_4 &= (-1, -1), & y_4 &= -1\\
\mathbf{x}_5 &= (-2, -2), & y_5 &= -1
\end{aligned}$$

(a) (4 points) Find the optimal hard margin SVM classifier for this dataset[1].

By examining the data, we can identify that the support vectors are:

$$\mathbf{x}_1 = (1, 2) \quad \text{from the positive class}$$
$$\mathbf{x}_4 = (-1, -1) \quad \text{from the negative class}$$

[1] $\mathbf{x}_1$ and $\mathbf{x}_4$ are the support vectors.

$$\mathcal{L}_D(\boldsymbol{\alpha}) = \alpha_1 + \alpha_4 - \frac{1}{2}\left(\alpha_1^2 x_1^\top x_1 - 2\alpha_1\alpha_4 x_1^\top x_4 + \alpha_4^2 x_4^\top x_4\right) \quad \text{s.t. } \alpha_1 - \alpha_4 = 0$$

$\mathcal{L}_D$ can be re-written as:

$$\begin{aligned}
\mathcal{L}_D(\boldsymbol{\alpha}) &= 2\alpha_1 - \frac{1}{2}\alpha_1^2\left(\underbrace{x_1^\top x_1}_{5} - 2\,\underbrace{x_1^\top x_4}_{-3} + \underbrace{x_4^\top x_4}_{2}\right) \\
&= 2\alpha_1 - \frac{13}{2}\alpha_1^2
\end{aligned}$$

Compute $\dfrac{\partial \mathcal{L}_D(\boldsymbol{\alpha})}{\partial \alpha_1}$:

$$\frac{\partial \mathcal{L}_D(\boldsymbol{\alpha})}{\partial \alpha_1} = 2 - 13\alpha_1$$

Setting $\dfrac{\partial \mathcal{L}_D(\boldsymbol{\alpha})}{\partial \alpha_1} = 0$ yields:

$$\alpha_1 = \alpha_4 = \frac{2}{13}$$

The optimal value for $\mathbf{w}^\star$ is:

$$\begin{aligned}
\mathbf{w}^\star &= \alpha_1 \mathbf{x}_1 - \alpha_4 \mathbf{x}_4 \\
&= \left[\frac{4}{13}, \frac{6}{13}\right]^\top
\end{aligned}$$

Using the support vector $\mathbf{x}_1$, we get:

$$\begin{aligned}
b^\star &= 1 - \mathbf{x}_1^\top \mathbf{w}^\star \\
&= -\frac{3}{13}
\end{aligned}$$

The hyperplane is therefore given by:

$$4x_1 + 6x_2 - 3 = 0$$

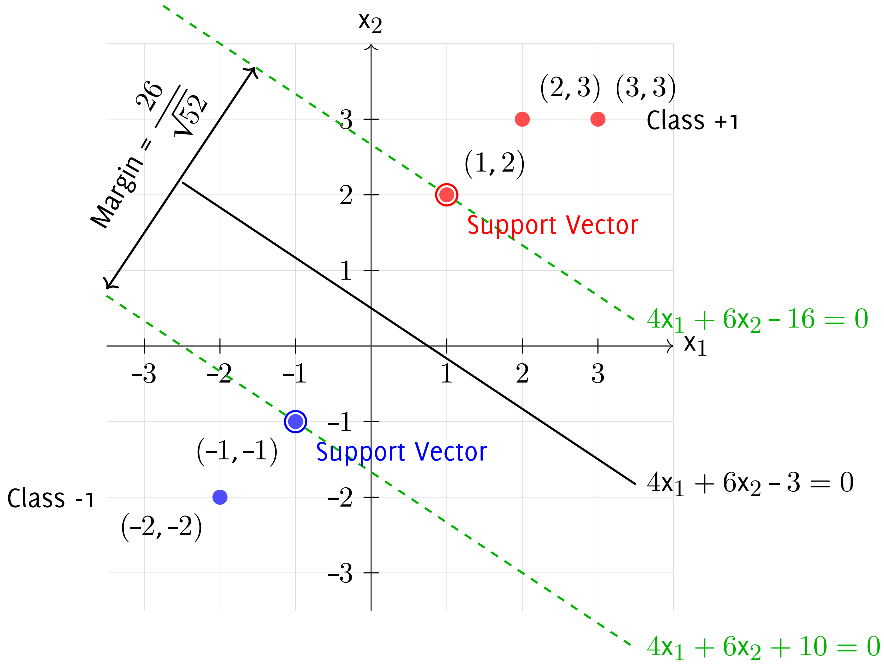


(b) (1 point) Calculate the margin of the classifier.

The margin width is calculated as:

$$\text{Margin Width} = \frac{2}{\|\mathbf{w}\|} = \frac{2}{\sqrt{(4/13)^2 + (6/13)^2}} = \frac{26}{\sqrt{52}}$$

(c) ($2\frac{1}{2}$ points) Visualize the dataset, decision boundary, and margin.




MEMENTO!



- The **dual problem** is given by:

$$\mathcal{L}_D(\boldsymbol{\alpha}) = \sum_{k=1}^{n} \alpha_k - \frac{1}{2} \sum_{k,l=1}^{n} \alpha_k \alpha_l y_k y_l \mathbf{x}_k^\top \mathbf{x}_l \quad \text{s.t.} \sum_{k=1}^{n} \alpha_k y_k = 0$$

- The optimal $\mathbf{w}^\star$ is a linear combination of support vectors:

$$\mathbf{w}^\star = \sum_{k \in SV} \alpha_k y_k \mathbf{x}_k$$

- The bias $b^\star$ is computed using any support vector $\mathbf{x}_k$:

$$b^\star = y_k - \mathbf{w}^{\star\top} \mathbf{x}_k$$

## Task N°2

⌛ 40mn | (9 points)

Our dataset consists of four points in a 2D (x, y) plane:

$$\mathsf{A} = (1,\ 3) \qquad \mathsf{B} = (2,\ 5)$$
$$\mathsf{C} = (3,\ 4) \qquad \mathsf{D} = (5,\ 6)$$

(a) Perform PCA on the dataset:

i. (1 point) Center the data.

Mean of each feature:

$$\text{Mean of x-coordinates} = \frac{1+2+3+5}{4} = \frac{11}{4} = 2.75$$
$$\text{Mean of y-coordinates} = \frac{3+5+4+6}{4} = \frac{18}{4} = 4.5$$

In order to center the data, we need to subtract the mean from each data point:

$$\text{Centered Point 1} = (1 - 2.75, 3 - 4.5) = (-1.75, -1.5)$$
$$\text{Centered Point 2} = (2 - 2.75, 5 - 4.5) = (-0.75, 0.5)$$
$$\text{Centered Point 3} = (3 - 2.75, 4 - 4.5) = (0.25, -0.5)$$
$$\text{Centered Point 4} = (5 - 2.75, 6 - 4.5) = (2.25, 1.5)$$



ii. (2 points) Compute the covariance matrix.

For a 2D dataset, the covariance matrix $\boldsymbol{C}$ is:

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

Computing each element:

$$\begin{aligned} \text{var(x)} &= \frac{(-1.75)^2 + (-0.75)^2 + (0.25)^2 + (2.25)^2}{3} \\ &= \frac{8.75}{3} \\ &\approx 2.917 \end{aligned}$$

$$\text{var(y)} = \frac{(-1.5)^2 + (0.5)^2 + (-0.5)^2 + (1.5)^2}{4}$$
$$= \frac{5}{3}$$
$$\approx 1.667$$

$$\text{cov(x, y)} = \frac{(-1.75)(-1.5) + (-0.75)(0.5) + (0.25)(-0.5) + (2.25)(1.5)}{4}$$
$$= \frac{5.5}{3}$$
$$\approx 1.833$$

$$\therefore \qquad \boldsymbol{C} = \begin{bmatrix} 2.917 & 1.833 \\ \\ 1.833 & 1.667 \end{bmatrix}$$



iii. (2 points) Find the eigenvalues and the eigenvectors of the covariance matrix.

To find the eigenvalues, we solve:

$$\det(\boldsymbol{C} - \lambda \mathbf{I}_2) = 0$$

For our covariance matrix:

$$\det\left(\begin{bmatrix} 2.917 - \lambda & 1.833 \\ 1.833 & 1.667 - \lambda \end{bmatrix}\right) = 0$$
$$(2.917 - \lambda)(1.667 - \lambda) - 1.833^2 = 0$$
$$\lambda^2 - 4.584\lambda + 1.503 = 0$$

Using the quadratic formula:

$$\lambda = \frac{4.584 \pm \sqrt{(4.584)^2 - 4 \cdot 1 \cdot 1.503}}{2 \cdot 1}$$
$$= \frac{4.584 \pm \sqrt{15.001}}{2}$$
$$= \frac{4.584 \pm 3.873}{2}$$

Therefore:

$$\lambda_1 = 4.228 \text{ (larger eigenvalue)} \quad \text{and} \quad \lambda_2 = 0.355 \text{ (smaller eigenvalue)}$$

For the eigenvector $\mathbf{v}_1 = [v_{11},\ v_{12}]^T$ corresponding to $\lambda_1 = 4.228$:

$$\begin{cases} (2.917 - 4.228)v_{11} + 1.833v_{12} = 0 \\ 1.833v_{11} + (1.667 - 4.228)v_{12} = 0 \end{cases} \implies \begin{cases} -1.311v_{11} + 1.833v_{12} = 0 \\ 1.833v_{11} - 2.561v_{12} = 0 \end{cases}$$

If we set $v_{11} = 1$, then $v_{12} = 0.715$, giving us $\mathbf{v}_1 = [1,\ 0.715]^T$.

Normalizing:

$$\begin{aligned} |\mathbf{v}_1| &= \sqrt{1^2 + (0.715)^2} \approx 1.229 \\ \mathbf{v}_1 &= \frac{[1,\ 0.715]^T}{1.229} \approx [0.814,\ 0.582]^T \end{aligned}$$

For the eigenvector $\mathbf{v}_2$ corresponding to $\lambda_2 = 0.355$:

$$\begin{cases} (2.917 - 0.355)v_{21} + 1.833v_{22} = 0 \\ 1.833v_{21} + (1.667 - 0.355)v_{22} = 0 \end{cases} \implies \begin{cases} 2.562v_{21} + 1.833v_{22} = 0 \\ 1.833v_{21} + 1.312v_{22} = 0 \end{cases}$$

If we set $v_{21} = 1$, then $v_{22} = -1.398$, giving us $\mathbf{v}_2 = [1,\ -1.398]^T$.

Normalizing:

$$\begin{aligned} |\mathbf{v}_2| &= \sqrt{1^2 + (1.398)^2} \approx 1.719 \\ \mathbf{v}_2 &= \frac{[1,\ -1.398]^T}{1.719} \approx [0.582,\ -0.813]^T \end{aligned}$$

(b) (2 points) Identify the principal components and explain which one captures the most variance.

The eigenvalues tell us how much variance is explained by each principal component:

$$\text{Total variance} = \lambda_1 + \lambda_2 = 4.228 + 0.355 = 4.583$$

$$\text{Proportion explained by PC}_1 = \frac{\lambda_1}{\text{Total variance}} \approx \frac{4.228}{4.583} \approx 0.922 \text{ or } 92.2\%$$

$$\text{Proportion explained by PC}_2 = \frac{\lambda_2}{\text{Total variance}} \approx \frac{0.355}{4.583} \approx 0.078 \text{ or } 7.8\%$$

To project the centered data points onto the principal components, we multiply each centered data point by the eigenvectors.

Projection onto $\text{PC}_1$ ($\mathbf{v}_1 = [0.814,\ 0.582]^\mathsf{T}$):

$$\text{Projection of A} = (-1.75 \times 0.814) + (-1.5 \times 0.582) = -2.298$$
$$\text{Projection of B} = (-0.75 \times 0.814) + (0.5 \times 0.582) = -0.32$$
$$\text{Projection of C} = (0.25 \times 0.814) + (-0.5 \times 0.582) = -0.088$$
$$\text{Projection of D} = (2.25 \times 0.814) + (1.5 \times 0.582) = 2.704$$

Projection onto $\text{PC}_2$ ($\mathbf{v}_2 = [0.582,\ -0.813]^\mathsf{T}$):

$$\text{Projection of A} = (-1.75 \times 0.582) + (-1.5 \times -0.813) = 0.2$$
$$\text{Projection of B} = (-0.75 \times 0.582) + (0.5 \times -0.813) = -0.843$$
$$\text{Projection of C} = (0.25 \times 0.582) + (-0.5 \times -0.813) = 0.552$$
$$\text{Projection of D} = (2.25 \times 0.582) + (1.5 \times -0.813) = 0.09$$

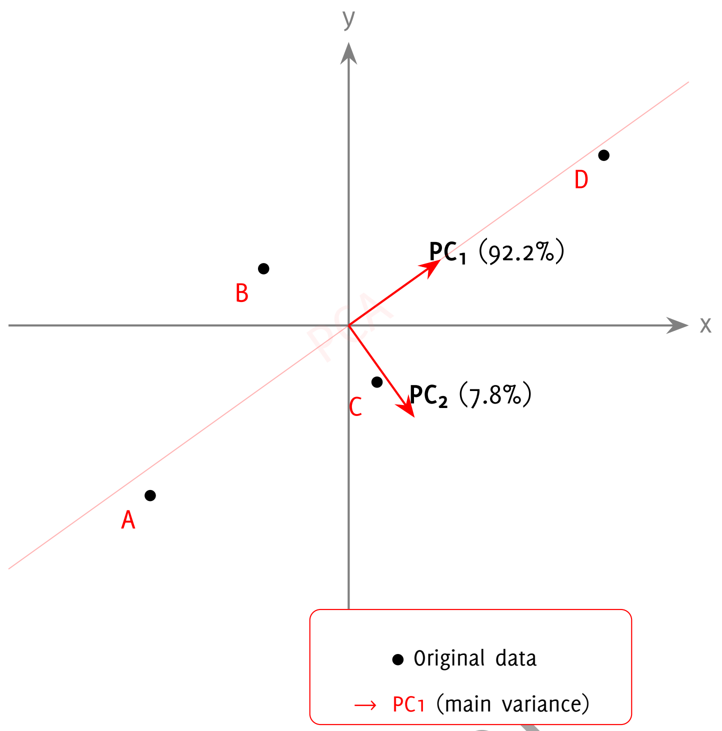
The transformed data points in the new PC coordinate system are:

$$\text{P}_\text{A} = (-2.298,\ 0.2) \qquad \text{P}_\text{B} = (-0.32,\ -0.843)$$
$$\text{P}_\text{C} = (-0.088,\ 0.552) \qquad \text{P}_\text{D} = (2.704,\ 0.09)$$

Since $\text{PC}_1$ captures approximately 92% of the variance, we could reduce our dimensionality by keeping only the first principal component:

$$\text{Point A (reduced)} \equiv \text{P}_\text{A} = -2.298$$
$$\text{Point B (reduced)} \equiv \text{P}_\text{B} = -0.32$$
$$\text{Point C (reduced)} \equiv \text{P}_\text{C} = -0.088$$
$$\text{Point D (reduced)} \equiv \text{P}_\text{D} = 2.704$$

(c) (2 points) Draw the principal components on a scatter plot and describe their relationship to the spread of data.




The first principal component captures most of the variance (92%), indicating that our 2D dataset can be effectively represented in a single dimension with minimal loss of information.



```julia
using DataFrames, MLJ
X = DataFrame(x=[1., 2, 3, 5], y=[3., 5, 4, 6])
#=
4×2 DataFrame
 Row │ x        y
     │ Float64  Float64
─────┼──────────────────
   1 │     1.0      3.0
   2 │     2.0      5.0
   3 │     3.0      4.0
   4 │     5.0      6.0
=#

```

```julia
# Load and fit ‘PCA‘
PCA = @load PCA pkg="MultivariateStats"
mach = machine(PCA(), X)
fit!(mach)

# Compute explained variance for each dimension
explained_variance = report(mach).principalvars
#=
2-element Vector{Float64}:
 4.228606549861576
 0.3547267834717567
=#

explained_variance ./= sum(explained_variance)
explained_variance .*= 100
#=
2-element Vector{Float64}:
 92.2605065424344
  7.739493457565602
=#

# Transform data to get components
components = MLJ.transform(mach, X)
#=
4×2 DataFrame
 Row │ x1          x2
     │ Float64     Float64
─────┼────────────────────────
   1 │  2.29607     0.201431
   2 │  0.318946   -0.843074
   3 │  0.0876672   0.5521
   4 │ -2.70268     0.0895427
=#
```

INSTITUTE OF TECHNOLOGICAL STUDIES OF BIZERTE

| | |
|---|---|
| **AY: 2024-2025** | **Full Name:** .................... |
| **M1-S2: Dept. of Electrical Engineering** | **ID:** .................... |
| EXAM \| **AI-ECUE221** | **Class:** **RAIA1** ................ |
| **June 2025** | **Room:** .................... |
| **Teacher: A. Mhamdi** | **Time Limit:** $1^1\!/_2$ h |

✂- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

## ANSWER SHEET

**Task N°3** ⌛ 15mn | ($3^1\!/_2$ points)

(a) ($^1\!/_2$ point) What is being calculated here?

```
using MLJ
LR = @load LinearRegressor pkg=GLM
mach = machine(LR(), X, y)
evaluate!(mach, resampling=Holdout(fraction_train=0.8),
↪   measure=[rms, rsquared])
```

√ Train-test split with 80% training data, reporting RMSE and R²

◯ 5-fold cross-validation using RMSE

◯ Bootstrapped confidence intervals for coefficients

◯ Feature importance scores using R²



(b) ($^1\!/_2$ point) What happens when executing this code?

```
struct Point
    x::Int
    y::Int
end
p = Point(1, 2)
p.x = 3
```

◯ Creates a new Point (3, 2)

◯ Creates a copy with updated x-value

√ Throws an error: `setfield!: immutable struct`

◯ Modifies p.x to 3 successfully

(c) ($^1\!/_2$ point) What does the following code output?

Do not write anything here

```julia
mutable struct Box
    content::String
end
b1 = Box("Apple")
b2 = b1
b2.content = "Orange"
println(b1.content)
```

○ "Apple"  √ "Orange"  ○ Throws an error  ○ Undefined beavior

(d) ($^1/_2$ point) Which of the following impurity measures is NOT differentiable with respect to class probabilities?

- ○ Gini impurity
- ○ Entropy
- ○ All of the above
- √ None of the above



(e) ($^1/_2$ point) In PCA, what is the relationship between the eigenvalues of the covariance matrix and the variance explained?

- ○ Eigenvalues are unrelated to variance in PCA.
- ○ Eigenvalues represent the correlation between original features and principal components.
- ○ Eigenvalues are inversely proportional to the variance explained.
- √ Eigenvalues correspond to the variance along each principal component.

(f) ($^1/_2$ point) In DBSCAN, which of the following statements about core points is FALSE?

- ○ A core point has at least MinPts points within its $\epsilon$-neighborhood (including itself).
- ○ A core point can be part of multiple clusters.
- √ A core point can be a noise point if it lies in a low-density region.
- ○ A core point always belongs to a cluster.

(g) ($^1/_2$ point) Which is not an assumption of linear regression?

√ Standardization  ○ Linearity  ○ Homoscedasticity  ○ Multivariate Normality