

# **LINMA2472 Algorithms in Data Science**

January 2026

## **Instructions**

1. Write your name, surname, NOMA on each sheet.
2. Write in the frames, not outside (your exam will be scanned and only the text within the frame will appear on-screen for the grader)
3. The exam lasts 3 hours.
4. The back face (verso) of each sheet is not read, you can use it as draft paper.
5. You can use your own blank paper as draft paper.
6. Calculators and other electronic devices are forbidden.
7. You must hand in everything at the end of the exam : questions, answers, drafts.
8. Please hand in the questions in the correct order.

**Question 1 : Kernels (applications)**

Suppose that you have  $N$  datapoints  $x_i \in \mathcal{X}$  with labels  $y_i \in \{-1, 1\}$  that you want to classify, where  $1 \leq i \leq N$ . To do so, you will try two different classical methods, enhanced with the kernel trick. For the whole exercise, we introduce the following kernel-related definitions :

- $\mathcal{H}$  is an RKHS,
- $\langle \cdot, \cdot \rangle_{\mathcal{H}} : \mathcal{H} \times \mathcal{H} \rightarrow \mathbb{R}$  is the inner product associated with  $\mathcal{H}$ ,
- $\| \cdot \|_{\mathcal{H}} : \mathcal{H} \rightarrow \mathbb{R}^+$  is the induced norm in  $\mathcal{H}$  such that  $\|f\|_{\mathcal{H}} = \sqrt{\langle f, f \rangle_{\mathcal{H}}}$  for any  $f \in \mathcal{H}$ ,
- $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  is the (symmetric positive-definite) kernel,
- $\phi : \mathcal{X} \rightarrow \mathcal{H}$  is a feature map such that  $k(x, y) = \langle \phi(x), \phi(y) \rangle_{\mathcal{H}}$  for any  $x, y \in \mathcal{X}$ ,
- $K \in \mathbb{R}^{N \times N}$  is the kernel matrix such that  $K_{ij} = k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle_{\mathcal{H}}$ .

**A) Method I : hard-margin kernel support-vector machine ( $k$ -SVM).**  $k$ -SVM tries to maximize the *margin* of the hyperplane which would separate the data embedded in the feature space. The decision function for any new input  $x \in \mathcal{X}$  thus reads  $d_I(x) = \langle v^*, \phi(x) \rangle_{\mathcal{H}} + b^*$ , where the direction vector of the hyperplane  $v^* \in \mathcal{H}$  and the bias  $b^* \in \mathbb{R}$  are found through the  $k$ -SVM optimization problem. Let  $\hat{y}_I(x)$  be the class predicted by the model  $x$ . This decision rule satisfies  $\hat{y}_I(x) = \text{sign}(d_I(x))$ , where the sign function is defined as  $\text{sign}(t) = -1$  if  $t < 0$ ,  $\text{sign}(0) = 0$  and  $\text{sign}(t) = 1$  if  $t > 0$ .

1. A famous theorem allows to decompose  $v^* = \sum_{i=1}^N \phi(x_i) \gamma_i^*$ , with  $\gamma^* \in \mathbb{R}^N$ . What is the name of this theorem ?

**Solution:** The **Representer Theorem**.

2. Write down the (primal)  $k$ -SVM optimization problem in terms of  $\gamma \in \mathbb{R}^N$  and  $b \in \mathbb{R}$ , using only  $K$ ,  $y_i$  and  $N$ .

**Solution:**

$$\begin{aligned} \min_{\gamma \in \mathbb{R}^N, b \in \mathbb{R}} \quad & \frac{1}{2} \gamma^\top K \gamma \\ \text{s.t.} \quad & y_i \left( \sum_{j=1}^N K_{ij} \gamma_j + b \right) \geq 1, \quad \forall i \in \{1, \dots, N\} \end{aligned}$$

or equivalently,  $y_i((K\gamma)_i + b) \geq 1$  for all  $i$ .

3. Let  $\gamma^* \in \mathbb{R}^N$  and  $b^* \in \mathbb{R}$  be the solutions of the  $k$ -SVM problem (we assume that they exist and that they are bounded). Give the expression for  $d_I(x)$  in terms of  $x$ ,  $\gamma^*$ ,  $b^*$ ,  $k$  and  $x_i$ .

Moreover, give the expression for  $\hat{y}_I(x)$ .

**Solution:**

$$d_I(x) = \sum_{i=1}^N \gamma_i^* k(x_i, x) + b^*,$$

$$\hat{y}_I(x) = \text{sign}(d_I(x)) = \text{sign} \left( \sum_{i=1}^N \gamma_i^* k(x_i, x) + b^* \right).$$

**B) Method II : (squared) distance to the (lifted) mean.** We define the centers of each class as

$$\mu_+ = \frac{1}{n_+} \sum_{\substack{i=1 \\ y_i=1}}^N \phi(x_i), \quad \mu_- = \frac{1}{n_-} \sum_{\substack{i=1 \\ y_i=-1}}^N \phi(x_i),$$

where  $n_+$  (resp.  $n_-$ ) is the number of points labeled +1 (resp. -1). We call  $\hat{y}_{II}(x)$  the class predicted for a new point  $x$  by the method of closest mean. The decision rule reads :

$$\hat{y}_{II}(x) = \begin{cases} 1 & \text{if } \|\phi(x) - \mu_+\|_{\mathcal{H}}^2 < \|\phi(x) - \mu_-\|_{\mathcal{H}}^2, \\ 0 & \text{if } \|\phi(x) - \mu_+\|_{\mathcal{H}}^2 = \|\phi(x) - \mu_-\|_{\mathcal{H}}^2, \\ -1 & \text{if } \|\phi(x) - \mu_+\|_{\mathcal{H}}^2 > \|\phi(x) - \mu_-\|_{\mathcal{H}}^2. \end{cases}$$

1. Propose and motivate an expression for the decision function  $d_{II}(x)$  which would satisfy the equation  $\hat{y}_{II}(x) = \text{sign}(d_{II}(x))$ , using only  $\phi(x)$ ,  $\mu_-$ ,  $\mu_+$ ,  $\langle \cdot, \cdot \rangle_{\mathcal{H}}$  and  $\|\cdot\|_{\mathcal{H}}^2$ .

Simplify the expression as much as possible.

**Solution:** We need to satisfy the condition on  $\text{sign}(d_{II}(x))$ . A solution is

$$\begin{aligned} d_{II}(x) &= \|\phi(x) - \mu_-\|_{\mathcal{H}}^2 - \|\phi(x) - \mu_+\|_{\mathcal{H}}^2 \\ &= \|\phi(x)\|_{\mathcal{H}}^2 - 2\langle \phi(x), \mu_- \rangle_{\mathcal{H}} + \|\mu_-\|_{\mathcal{H}}^2 - \|\phi(x)\|_{\mathcal{H}}^2 + 2\langle \phi(x), \mu_+ \rangle_{\mathcal{H}} - \|\mu_+\|_{\mathcal{H}}^2 \\ &= 2\langle \phi(x), \mu_+ - \mu_- \rangle_{\mathcal{H}} + \|\mu_-\|_{\mathcal{H}}^2 - \|\mu_+\|_{\mathcal{H}}^2. \end{aligned}$$

This choice of  $d_{II}(x)$  is great because it simplifies to an affine function, because it is easy to differentiate (ex : for optimization purposes), because it is easy to compute, because it naturally comes from the conditions, because it is natural to look at squared norms (ex : least squares methods),...

2. Express the squared distances  $\|\phi(x) - \mu_+\|_{\mathcal{H}}^2$  and  $\|\phi(x) - \mu_-\|_{\mathcal{H}}^2$  using only  $x$ ,  $k$ ,  $x_i$ ,  $y_i$ ,  $N$ ,  $n_+$  and  $n_-$ .

**Solution:**

$$\begin{aligned} \|\phi(x) - \mu_+\|_{\mathcal{H}}^2 &= \|\phi(x)\|_{\mathcal{H}}^2 - 2\langle \phi(x), \mu_+ \rangle_{\mathcal{H}} + \|\mu_+\|_{\mathcal{H}}^2 \\ &= k(x, x) - \frac{2}{n_+} \sum_{i:y_i=1} k(x, x_i) + \frac{1}{n_+^2} \sum_{i:y_i=1} \sum_{j:y_j=1} k(x_i, x_j), \\ \|\phi(x) - \mu_-\|_{\mathcal{H}}^2 &= k(x, x) - \frac{2}{n_-} \sum_{i:y_i=-1} k(x, x_i) + \frac{1}{n_-^2} \sum_{i:y_i=-1} \sum_{j:y_j=-1} k(x_i, x_j). \end{aligned}$$

3. Just for this subquestion, assume that  $n_+ = n_-$  (classes are balanced) and that  $\|\mu_+\|_{\mathcal{H}} = \|\mu_-\|_{\mathcal{H}}$ . Simplify the rule  $\hat{y}_{II}(x)$  to express it only in terms of  $x$ ,  $N$ ,  $k$ ,  $x_i$  and  $y_i$ .

*Hint* : What is the relation between  $N$ ,  $n_+$  and  $n_-$  ?

**Solution:** Since  $n_+ = n_- = N/2$  and  $\|\mu_+\|_{\mathcal{H}} = \|\mu_-\|_{\mathcal{H}}$ , we have :

$$\begin{aligned} d_{II}(x) &= 2\langle \phi(x), \mu_+ - \mu_- \rangle_{\mathcal{H}} \\ &= \frac{2}{n_+} \sum_{i:y_i=1} k(x, x_i) - \frac{2}{n_-} \sum_{i:y_i=-1} k(x, x_i) \\ &= \frac{4}{N} \sum_{i=1}^N y_i k(x, x_i). \end{aligned}$$

Therefore,  $\hat{y}_{II}(x) = \text{sign} \left( \frac{4}{N} \sum_{i=1}^N y_i k(x, x_i) \right) = \text{sign} \left( \sum_{i=1}^N y_i k(x, x_i) \right)$ .

**C) Methods comparison.**

1. Are the decision functions  $d_I(x)$  and  $d_{II}(x)$  affine functions of  $\phi(x)$  in  $\mathcal{H}$ ? Justify briefly.

**Solution:** Yes, both are affine functions of  $\phi(x)$  :

—  $d_I(x) = \langle v^*, \phi(x) \rangle_{\mathcal{H}} + b^*$  is affine (linear plus constant).

—  $d_{II}(x) = \langle 2(\mu_+ - \mu_-), \phi(x) \rangle_{\mathcal{H}} + \|\mu_-\|_{\mathcal{H}}^2 - \|\mu_+\|_{\mathcal{H}}^2$  is also affine (linear plus constant).

Both have the form  $\langle w, \phi(x) \rangle_{\mathcal{H}} + c$  for some  $w \in \mathcal{H}$  and  $c \in \mathbb{R}$ .

2. If possible, give simple conditions under which both methods would be equivalent, meaning that for any new datapoint  $x \in \mathcal{X}$ , they would always predict the same class.

**Solution:** The methods would be equivalent iff  $d_I(x)$  and  $d_{II}(x)$  have the same sign for all  $x$ , i.e.  $d_I(x) = cd_{II}(x)$  for  $c > 0$ . A simple choice is  $c = 1$  so  $d_I(x) = d_{II}(x)$ .

This gives the conditions  $v^* = 2(\mu_+ - \mu_-)$  and  $b^* = \|\mu_-\|_{\mathcal{H}}^2 - \|\mu_+\|_{\mathcal{H}}^2$ .

3. What method would you prefer to use for your classification task? Choose a method and give at least two arguments in favor of it.

**Solution:** I would prefer **Method I ( $k$ -SVM)** for the following reasons :

- (a) **Maximum margin** : Intelligent choice of separating hyperplane.
- (b) **Sparse and scalable** : The solution typically uses only a subset of training points (support vectors), making predictions (inference) more efficient. This makes the method scalable to larger or more complex datasets at inference.
- (c) **Theoretical guarantees** : SVMs have strong theoretical foundations with bounds on generalization error based on margin maximization.
- (d) **Improvable** : Extension to soft-margin  $k$ -SVM for more robustness to outliers and better generalization.

I would prefer **Method II (nearest mean)** for the following reasons :

- (a) **Simple and interpretable** : Simple and intuitive motivation, easy to understand and to interpret.
- (b) **No pre-compute** : No need to solve an optimization problem to get the coefficients.
- (c) **Non-separable data** : Works despite non-linearly separable data in  $\mathcal{H}$
- (d) **Adaptable** : Easily adaptable to new datapoints.

**Question 2 : Diffusion**

You would like to find a method for deblurring images. You have a program  $b(x, \sigma)$  that can be used to blur an image  $x$  with blurring intensity  $\sigma$ . For blurring intensity  $\sigma = 0$ ,  $b(x, \sigma) = x$  and the larger  $\sigma$  is, the more the image is blurred.

- Given a datasets of clear images, explain how you can use the Evidence Lower Bound to train a model  $f_\theta(y, \sigma)$ , parameterized by  $\theta$ , that attempts to predicts the clear image  $x$  that produced the blurry image  $y = b(x, \sigma)$ .

**Solution:** We want to train a Denoising Auto-Encoder. We define  $Z_\sigma$  to be the latent variable in the space of images of blur intensity  $\sigma$  that produces the clear image  $X$  and the variable  $Y_\sigma = b(X, \sigma)$ . Sample clear images  $x$  from the dataset and various different blur intensities  $\sigma$ . Create pairs  $(y, x)$  with  $y = b(x, \sigma)$ . The ELBO bounds  $\log p(y|\sigma)$  :

$$\log f_X(x) \geq -D_{\text{KL}}((Y|X=x)||Z) + \mathbb{E}[\log f_{X|Z}(x|Y)].$$

The first term serves as regularizer and the second term is the error.

- You now use a trained model  $f_{\theta^*}$  to deblur an image  $y$  of supposed blurred intensity  $\sigma$  using  $\hat{x} = f_{\theta^*}(y, \sigma)$ . You observe that, while this works well for small  $\sigma$ , the estimate is less precise for deblurring image with high blurred intensity  $\sigma$ . How would you suggest to improve this using an iterative approach?

**Solution:**  $f_{\theta^*}(y, \sigma)$  is based on an estimate of the gradient of the log of the PDF of the distribution of images blurred with intensity  $\sigma$ . For higher  $\sigma$ , this distribution is more spread out, so the gradient is less precise. For large  $\sigma$ , it's only good to get *close* to the distribution of less blurry images.

Therefore, we use an **iterative refinement** with decreasing blur levels. Define  $\sigma_0 = \sigma > \sigma_1 > \dots > \sigma_T \approx 0$ . Initialize  $\hat{x}_0 = f_{\theta^*}(y, \sigma_0)$ . For  $k = 1, \dots, T$  : simulate a “less blurry” observation by  $\tilde{y}_k = b(\hat{x}_{k-1}, \sigma_k)$ , then set  $\hat{x}_k = f_{\theta^*}(\tilde{y}_k, \sigma_k)$ . Final deblurred image :  $\hat{x}_T$ .

**Question 3 : Implicit**

Consider the function

$$f(x, \lambda) : x^3 - \lambda x^2 + x - 1$$

If  $|\lambda| < \sqrt{3}$ , the derivative  $\partial_1 f(x, \lambda) = 3x^2 - 2\lambda x + 1$  is positive for all  $x$ . Therefore, for any  $\lambda \in (-\sqrt{3}, \sqrt{3})$ , there exists a unique  $x^*$  such that  $f(x^*, \lambda) = 0$ . Let  $x^*(\lambda) : (-\sqrt{3}, \sqrt{3}) \rightarrow \mathbb{R}$  be this solution.

— We have  $x^*(1) = 1$ , find the value of derivative  $\partial x^*(1)$ .

**Solution:** We have

$$\partial x^*(\lambda) = \frac{-\partial_2 f(1, 1)}{\partial_1 f(1, 1)} = \frac{(x^*)^2}{3(x^*)^2 - 2\lambda x^* + 1} = \frac{1}{2}$$

— What is the value of the second derivative  $\partial^2 x^*(1)$ ?

**Solution:** Let  $n, d$  be the numerator and denominator :

$$n = (x^*)^2 \quad d = 3(x^*)^2 - 2\lambda x^* + 1.$$

We have

$$\partial^2 x^*(\lambda) = \frac{n'd - nd'}{d^2}$$

where  $n' = 2x^*(x^*)' = 2 \cdot 1 \cdot (1/2) = 1$ ,  $d = 2$ ,  $n = 1$  and

$$d' = 6x^*(x^*)' - 2\lambda' x^* - 2\lambda(x^*)' = 3 - 2 - 2/2 = 0$$

so

$$\partial^2 x^*(\lambda) = \frac{1 \cdot 2 - 1 \cdot 0}{2^2} = \frac{2}{4} = \frac{1}{2}.$$

**Question 4 : Hessian coloring**

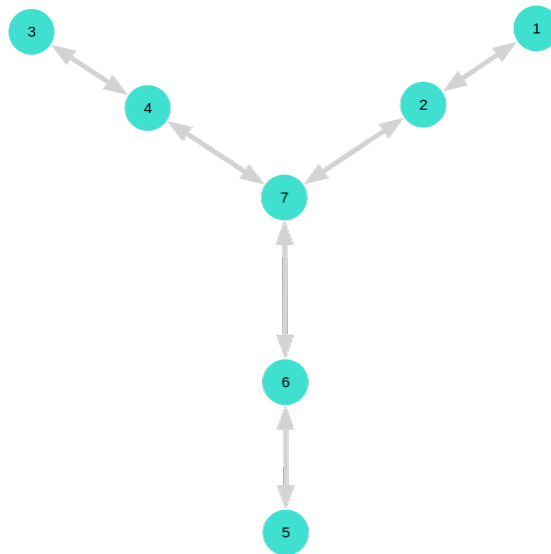
Consider the function

$$f : \mathbb{R}^7 \rightarrow \mathbb{R} : f(x) = \sum_{i=1}^3 x_{2i}(x_{2i-1} - x_7) + \sum_{i=1}^7 x_i^2$$

Our aim is to compute the Hessian  $\nabla^2 f(x)$  by exploiting its sparsity.

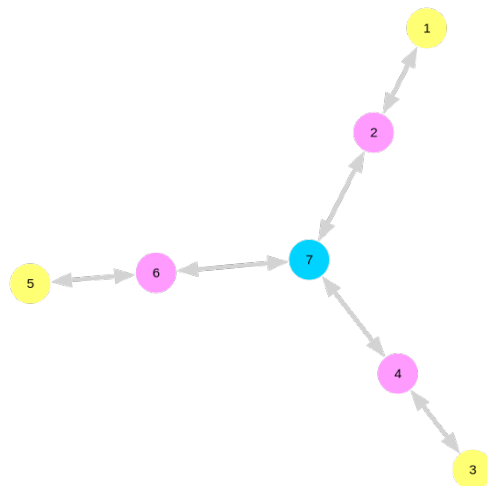
— Find the adjacency graph of the Hessian matrix :

**Solution:**



— How many colors would be used with **star** coloring ?

**Solution:**



3

- How would the nonzero entries of the Hessian be obtained from Hessian-Vector Products using this star coloring?

**Solution:** The three Hessian-Vector Products (HVPs) corresponding to the two colors are :

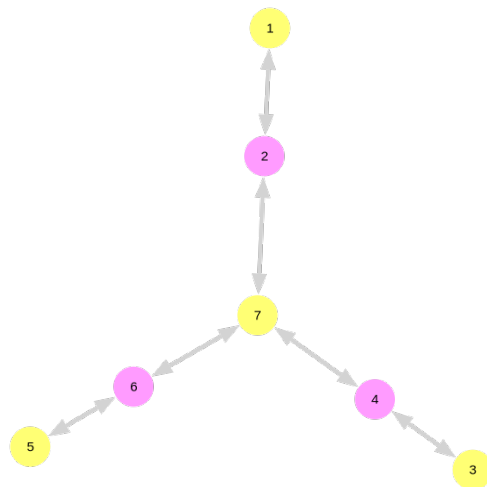
$$H \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{12} \\ a_{33} \\ a_{34} \\ a_{55} \\ a_{56} \\ 0 \end{bmatrix} \quad H \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} a_{12} \\ a_{22} \\ a_{34} \\ a_{44} \\ a_{56} \\ a_{66} \\ a_{27} + a_{47} + a_{67} \end{bmatrix} \quad H \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ a_{27} \\ 0 \\ a_{47} \\ 0 \\ a_{67} \\ a_{77} \end{bmatrix}$$

The diagonal entry  $a_{ii}$  is obtained from the HVP of the  $\phi(i)$ -th color. The off-diagonal entry  $a_{ij}$  of a star with center  $j$  in the subgraph of colors  $\phi(i)$  and  $\phi(j)$  is obtained from the  $i$ -th entry of the HVP of the  $\phi(j)$ -th color.

In this case, the off-diagonal entries  $a_{27}, a_{47}, a_{67}$  are obtained from the rows 2, 4, 6 respectively of the HVP  $He_7$ . For the off-diagonal entries  $a_{12}, a_{34}, a_{56}$ , the 2-color subgraph is a simple edge so both nodes can be treated as center of the star. So they can either be obtained from the rows 1, 3, 5 respectively of the HVP  $H(e_2 + e_4 + e_6)$  or from the rows 2, 4, 6 respectively of the HVP  $H(e_1 + e_3 + e_5)$ .

- How many colors would be used with **acyclic** coloring?

**Solution:**



2

- How would the nonzero entries of the Hessian be obtained from the Hessian-Vector Products using this acyclic coloring?

**Solution:**

$$H \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} a_{12} \\ a_{22} \\ a_{34} \\ a_{44} \\ a_{56} \\ a_{66} \\ a_{27} + a_{47} + a_{67} \end{bmatrix} \quad H \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{12} + a_{27} \\ a_{33} \\ a_{34} + a_{47} \\ a_{55} \\ a_{56} + a_{67} \\ a_{77} \end{bmatrix}$$

The diagonal entry  $a_{ii}$  is obtained from the HVP of the  $i$ th color. The off-diagonal entry  $a_{ij}$  of a leaf  $i$  in the subgraph of colors  $\phi(i)$  and  $\phi(j)$  is obtained from the  $i$ -th entry of the HVP of the  $\phi(j)$ -th color. We then subtract their values from the row  $j$  of the HVP of color  $\phi(i)$  and remove this edge from the subgraph to allow new leaves to emerge and continue the process.

In this case, the off-diagonal entries  $a_{12}, a_{34}, a_{56}$  are obtained from the rows 1, 3, 5 resp. of the HVP  $H(e_2 + e_4 + e_6)$ . Their values are then subtracted from the rows 2, 4, 6 resp. of the HVP  $H(e_1 + e_3 + e_5 + e_7)$ . The off-diagonal entries  $a_{27}, a_{47}, a_{67}$  are then obtained from these respective rows.