Probability distributions

For your reference, we provide the following probability density functions.

- Normal distribution

$$\mathcal{N}(x \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

- Gamma distribution

$$\text{Gamma}(x \mid \alpha, \beta) = \begin{cases} \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x} & \text{if } x \in (0, \infty), \\ 0 & \text{else} \end{cases}$$

  where $\Gamma(\cdot)$ is the gamma function.

- Log-normal distribution

$$\text{Log-normal}(x \mid \mu, \tau) = \begin{cases} \frac{\sqrt{\tau}}{x\sqrt{2\pi}} \exp\left(-\frac{\tau}{2}(\log x - \mu)^2\right) & \text{if } x \in (0, \infty), \\ 0 & \text{else.} \end{cases}$$

Probability Theory

Problem 1 [2 points] Given is the joint PDF of 3 continuous random variables $p(a, b, c)$. Write down how the following expressions can be obtained using the rules of probability.

(i) $p(a)$ (ii) $p(c \mid a, b)$ (iii) $p(b \mid c)$

$$p(a) = \int\int p(a, b, c) db dc$$

$$p(c \mid a, b) = \frac{p(a, b, c)}{p(a, b)} = \frac{p(a, b, c)}{\int p(a, b, c) dc}$$

$$p(b \mid c) = \frac{p(b, c)}{p(c)} = \frac{\int p(a, b, c) da}{\int\int p(a, b, c) da db}$$

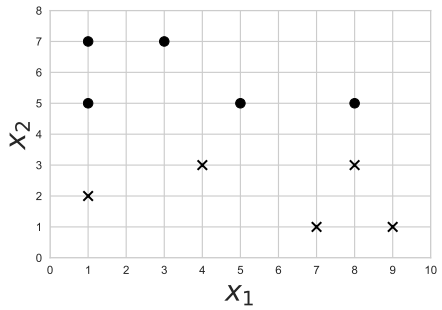
K Nearest Neighbors

Problem 2 [3 points] Given is the data in the figure below, with binary labels marked as • and ×:

a) How many points get misclassified during the leave-one-out cross-validation (LOOCV) procedure when using a 1-NN classifier?

Five.

b) The original data can be represented as a matrix $\boldsymbol{X}_{orig} \in \mathbb{R}^{10\times 2}$, where samples are stored as rows. We can apply a linear transformation to the data as $\boldsymbol{X}_{new} = \boldsymbol{X}_{orig} \cdot \boldsymbol{A}$, where $\boldsymbol{A} \in \mathbb{R}^{2\times 2}$ is the transformation matrix.



Find a linear transformation matrix $\boldsymbol{A} \in \mathbb{R}^{2\times 2}$, such that all points get classified correctly during the LOOCV procedure using a 1-NN classifier. Provide a short explanation.

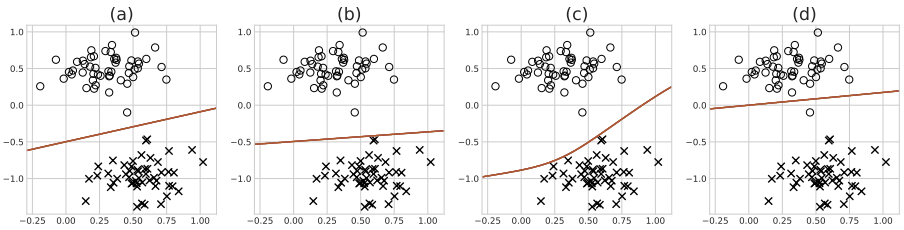
The transformation matrix

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

projects the data onto the x_2 axis, effectively disregarding the x_1 attribute. Given such a projection we can see that all points get correctly classified using the LOOCV procedure.

Classification

Problem 3 [3 points] We fitted 4 different classification algorithms on the same dataset and obtained 4 different decision boundaries.



Match each classifier to the respective decision boundary. Explain each of the answers with 1 sentence.

#	**Classification algorithm**	**Plot**
1.	Logistic regression	
2.	Hard margin linear SVM	
3.	Soft margin SVM with polynomial kernel	
4.	Perceptron	

3. Soft margin SVM with polynomial kernel - plot (c). The only method that has a nonlinear decision boundary.

1. Logistic regression - plot (d). Linear decision boundary + the only of the remaining methods that permit misclassifying some training samples.

2. Hard margin linear SVM - plot (a). Linear decision boundary + no points are misclassified + has the highest margin.

4. Perceptron - plot (b). All points are correctly classified, but the margin is smaller than in SVM case (plot (a)).

Problem 4 [5 points] We have a binary classification problem with 1-dimensional data. The training samples from class 1 are $\{-1, 1, 3, 6, 1\}$ and the samples from class 2 are $\{5.5, 7, 9, 9, 9.5\}$.

We model class priors as a categorical distribution with parameters $\{\pi_1, \pi_2\}$. We model the class-conditionals as Gaussian distributions $\mathcal{N}(\mu_1, \sigma_1^2)$ and $\mathcal{N}(\mu_2, \sigma_2^2)$. Assume that the variances of class-conditionals are known to be $\sigma_1^2 = 2$ and $\sigma_2^2 = 1$.

a) Find the maximum likelihood estimates (MLE) of the prior class probabilities $\{\pi_1, \pi_2\}$, as well as of the means $\{\mu_1, \mu_2\}$ of the class conditional densities.

The MLE solution for the prior class probabilities is equal to the fraction of instances in each class:

$$\pi_1 = \pi_2 = \frac{5}{10} = 0.5$$

The MLE solution for the means is the the mean of the instances belonging to each class:

$$\mu_1 = \frac{-1+1+3+6+1}{5} = 2 \qquad \mu_2 = \frac{5.5+7+9+9+9.5}{5} = 8$$

b) What class will the point $x = 6$ be assigned to? Justify your answer. *Hint:* $\ln 1/\sqrt{2} \approx -0.35$

Ignoring constants we have:

$$p(y=1|x=6) = p(x=6|\mu=2, \sigma^2=2) \times \pi_1 = 0.5 \frac{1}{\sqrt{2\pi 2}} e^{-\frac{(6-2)^2}{4}}$$

$$p(y=2|x=6) = p(x=6|\mu=8, \sigma^2=1) \times \pi_2 = 0.5 \frac{1}{\sqrt{2\pi}} e^{-\frac{(6-8)^2}{2}}$$

Since this is a two class problem, it is convenient to calculate the log posterior probability ratio:

$$\ln \frac{p(y=1 \mid x=6)}{p(y=2 \mid x=6)} = \left(\ln(\frac{0.5}{\sqrt{2\pi}}) - \ln \frac{1}{\sqrt{2}} - \ln(\frac{0.5}{\sqrt{2\pi}}) + \ln \exp(-4+2) \right) \approx -2 \times 0.35 = -0.75$$

Thus, the point will be assigned to class 2.

c) How many points $x \in \mathbb{R}$ lie on the decision boundary? That is, for how many points $x \in \mathbb{R}$ does it hold $p(y = 1 \mid x) = p(y = 2 \mid x)$? Provide a mathematical justification for your answer.

Two. The decision boundary is the set of points that solve the equation

$$\ln \frac{p(y=1 \mid x)}{p(y=2 \mid x)} = 0$$

Since $\sigma_1^2 \neq \sigma_2^2$, this is a quadratic equation with two roots.

Dimensionality Reduction

Problem 5 [5 points] Outline the main steps of applying PCA to reduce the dimensionality of a dataset $\mathbf{X} \in \mathbb{R}^{N \times D}$ from D to K.

1. Center the data
$$\tilde{\boldsymbol{X}} = \boldsymbol{X} - \mathbf{1}_N \underbrace{(\frac{1}{N}\mathbf{1}_N^T \boldsymbol{X})}_{\tilde{\boldsymbol{x}}}.$$
2. Compute the covariance matrix
$$\boldsymbol{\Sigma} = \frac{1}{N}\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}}.$$
3. Compute the eigendecompoisiton of $\boldsymbol{\Sigma}$
$$\boldsymbol{\Sigma} = \boldsymbol{\Gamma}^T\boldsymbol{\Lambda}\boldsymbol{\Gamma}^T.$$
4. Choose the top-K eigenvectors $\boldsymbol{\Gamma_K}$ (the ones that correspond to the K largest eigenvalues) of the covariance matrix $\boldsymbol{\Sigma}$.
5. Project the data
$$\boldsymbol{Y} = \tilde{\boldsymbol{X}}\boldsymbol{\Gamma_K}.$$

Deep Learning

Problem 6 [4 points] Consider the following classification problem. There are two real-valued features x_1 and x_2, and a binary class label. The ground truth class labels are generated according to the following rule:

$$y = \begin{cases} 1 & \text{if } x_2 \geq |x_1|, \\ 0 & \text{else} \end{cases}$$

a) Can this function be perfectly represented by a feed-forward neural network with no hidden layers and 1 softmax output layer? Why or why not?

No. The function is non-linear.

b) Design a two layer feed-forward network (that is, one hidden layer followed by an output layer, two weight matrices in total) that represents this function. You are allowed to use the hard thresholding activation function $\sigma(x)$ as the elementwise nonlinearity.

$$\sigma(x) = \begin{cases} 1 & x > 0 \\ 0 & \text{else} \end{cases}$$

Specify the number of neurons and values of weights in each layer.

Let the first hidden neuron of the first layer compute the following output:

$$h_1 = \sigma(1.0 \times x_2 - 1.0 \times x_1 + 0.0) = \begin{cases} 1 & x_2 > x_1 \\ 0 & \text{else} \end{cases}$$

and the second hidden neuron compute the following output:

$$h_2 = \sigma(1.0 \times x_2 + 1.0 \times x_1 + 0.0) = \begin{cases} 1 & x_2 > -x_1 \\ 0 & \text{else} \end{cases}$$

In the second layer we combine them to produce the final output:

$$y = 1.0 \times h_1 + 1.0 \times h_2 - 1 = \begin{cases} 1 & x_2 > x_1 \quad \text{and} \quad x_2 > -x_1 \\ 0 & \text{else} \end{cases} = \begin{cases} 1 & x_2 > |x_1| \\ 0 & \text{else} \end{cases}$$

Here we've used the bias term -1 to construct what is effectively an AND gate.

Optimization

You are given the following objective function

$$f(x_1, x_2) = 0.5x_1^2 + x_2^2 + 2x_1 + x_2 + \cos(\sin(\sqrt{\pi})).$$

Problem 7 [2 points] Compute the minimum (x_1^*, x_2^*) of $f(x_1, x_2)$ analytically.

As f is a sum of convex functions, it is convex (in x_1 and x_2).

To find the minimum, simply compute the gradient and set it to zero

$$\nabla f(x_1, x_2) = \begin{pmatrix} x_1 + 2 \\ 2x_2 + 1 \end{pmatrix} \stackrel{!}{=} \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$
$$\implies \begin{pmatrix} x_1^* \\ x_2^* \end{pmatrix} = \begin{pmatrix} -2 \\ -\frac{1}{2} \end{pmatrix}.$$

Problem 8 [2 points] Perform 2 steps of gradient descent on $f(x_1, x_2)$ starting from the point $(x_1^{(0)}, x_2^{(0)}) = (0, 0)$ with learning rate $\tau = 1$.

We already know the gradient from Problem 9. The first update

$$\begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix} = \begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \end{pmatrix} - \tau \begin{pmatrix} x_1^{(0)} + 2 \\ 2x_2^{(0)} + 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} - 1 \begin{pmatrix} 0 + 2 \\ 0 + 1 \end{pmatrix} = \begin{pmatrix} -2 \\ -1 \end{pmatrix}.$$

The second update

$$\begin{pmatrix} x_1^{(2)} \\ x_2^{(2)} \end{pmatrix} = \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix} - \tau \begin{pmatrix} x_1^{(1)} + 2 \\ 2x_2^{(1)} + 1 \end{pmatrix} = \begin{pmatrix} -2 \\ -1 \end{pmatrix} - 1 \begin{pmatrix} -2+2 \\ -2+1 \end{pmatrix} = \begin{pmatrix} -2 \\ 0 \end{pmatrix}.$$

Problem 9 [2 points] Will the gradient descent procedure from Problem 8 ever converge to the true minimum (x_1^*, x_2^*)? Why or why not? If the answer is no, how can we fix it?

By performing one more iteration of GD we observe that

$$\begin{pmatrix} x_1^{(3)} \\ x_2^{(3)} \end{pmatrix} = \begin{pmatrix} x_1^{(2)} \\ x_2^{(2)} \end{pmatrix} - \tau \begin{pmatrix} x_1^{(2)} + 2 \\ 2x_2^{(2)} + 1 \end{pmatrix} = \begin{pmatrix} -2 \\ 0 \end{pmatrix} - 1 \begin{pmatrix} -2+2 \\ 0+1 \end{pmatrix} = \begin{pmatrix} -2 \\ -1 \end{pmatrix} = \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix}$$

That is, we are stuck iterating between $\boldsymbol{x^{(1)}}$ and $\boldsymbol{x^{(2)}}$ forever.

We can fix this by decreasing the learning rate (adaptive stepsize, etc.).

Problem 10 [2 points] Given two convex functions $g_1 : \mathbb{R} \to \mathbb{R}$ and $g_2 : \mathbb{R} \to \mathbb{R}$, prove or disprove that the function $h(x) = g_1(g_2(x))$ is also convex.

Disprove by counterexample:

Consider $g_1(x) = -x$ and $g_2(x) = x^2$. Both g_1 and g_2 are convex, but $h(x) = -x^2$ is concave.

Regression

John Doe is a data scientist, and he wants to fit a polynomial regression model to his data. For this, he needs to choose the degree of the polynomial that works best for his problem.

Unfortunately, John hasn't attended IN2064, so he writes the following code for choosing the optimal degree of the polynomial:

```
X, y = load_data()
best_error = -1
best_degree = None

for degree in range(1, 50):
    w = fit_polynomial_regression(X, y, degree)
    y_predicted = predict_polynomial_regression(X, w, degree)
    error = compute_mean_squared_error(y, y_predicted)
    if (error <= best_error) or (best_error == -1):
        best_error = error
        best_degree = degree
```

```
print("Best degree is " + str(best_degree))
```

Assume that the functions are implemented correctly and do what their name suggests.
(e.g., `fit_polynomial_regression` returns the optimal coefficients `w` for a polynomial regression model with the given degree.)

Problem 11 [3 points] What is the output of this code? Explain in 1-2 sentences why this code doesn't do what it's supposed to do.

Output: `Best degree is 49`
(`range(1, 50)` in Python goes to 49, but 50 is also accepted as a correct answer)

Error on the training set *always* goes down when we use a higher degree polynomial (unless it's already 0, then it stays at 0).

Problem 12 [2 points] Describe in 1-2 sentences a possible way to fix the problem with this code. (You don't need to write any code, just describe the approach.)

Split data into train and validation sets. Choose the degree that achieves the lowest mean squared error on the validation set (not on the training set!).

Remark: Regularization does not help at all in this case. No matter which λ you choose, higher degree polynomial is still able to fit the training data better.

SVM & Constrained optimization

Problem 13 [5 points] The goal is to minimize the function $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = 4x^2$ subject to $f_1(x) = -2x + 1 \leq 0$ using constrained optimization methods.

a) Write down the Lagrangian $L(x, \alpha)$ for this constrained optimization problem. Denote the necessary Lagrange multiplier as α.

$$L(x, \alpha) = f(x) + \alpha f_1(x) = 4x^2 + \alpha(1 - 2x) = 4x^2 - 2\alpha x + \alpha\,.$$

b) Obtain the Lagrange dual function $g(\alpha)$ from the Lagrangian $L(x, \alpha)$.

The Lagrange dual function is obtained by minimizing $L(x, \alpha)$ w.r.t. x. This is done by calculating the corresponding partial derivative,

$$\frac{\partial L}{\partial x} = 8x - 2\alpha\,,$$

setting it to zero and solving for x,

$$x = \frac{\alpha}{4}\,.$$

By substituting this back into $L(x, \alpha)$ we obtain

$$g(\alpha) = L(\alpha/4, \alpha) = -\frac{1}{4}\alpha^2 + \alpha\,.$$

c) State the dual problem explicitly and solve it to obtain the value for the Lagrange multiplier α.

The Lagrange dual problem is

$$\begin{aligned} &\text{maximize } g(\alpha) = -\frac{1}{4}\alpha^2 + \alpha \\ &\text{s.t. } \alpha \geq 0 \end{aligned}$$

The solution is given by calculating the derivative,

$$g'(\alpha) = -\frac{\alpha}{2} + 1\,,$$

and setting it to zero. Solving for α gives

$$\alpha^* = 2\,.$$

d) What is the duality gap in this problem? Justify your answer.

Since f is convex and f_1 is affine Slater's theorem applies and the duality gap is zero.

e) Obtain the solution to the original problem from the solution of the dual problem.

The minimizer x^* of the original optimization problem is given by

$$x^* = \frac{\alpha^*}{4} = \frac{2}{4} = \frac{1}{2}$$

and the minimum is

$$f(x^*) = 1\,.$$

Problem 14 [3 points] Prove or disprove that the function $k : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ is a valid kernel.

$$k(\boldsymbol{x}, \boldsymbol{y}) = x_1 y_1 - x_2 y_2$$

A valid kernel $k(\boldsymbol{x}, \boldsymbol{y})$ by definition must correspond to an inner product $\langle \varphi(\boldsymbol{x}), \varphi(\boldsymbol{y}) \rangle$ for some feature map $\varphi : \mathbb{R}^2 \to \mathcal{H}$.

By the positive definiteness property of inner product it must hold that

$$\langle \boldsymbol{z}, \boldsymbol{z} \rangle \geq 0 \quad \text{for all } \boldsymbol{z} \in \mathcal{H}.$$

Consider $\boldsymbol{x} = (0, 1)^T$.

$$k(\boldsymbol{x}, \boldsymbol{x}) = 0 - 1 = -1 < 0,$$

which violates the positive definiteness property, hence k does not correspond to an inner product, hence k is not a valid kernel.

Alternative solution: By Mercer's theorem a valid kernel must produce a PSD Gram matrix $\mathbf{K}$ *(a.k.a. kernel matrix) when applied to any dataset* $\mathbf{X} = \{\boldsymbol{x}_1, ...\boldsymbol{x}_N\}$.

Consider the case with a single sample $\boldsymbol{x} = (0, 1)^T$.

In this case $k(\boldsymbol{x}, \boldsymbol{x}) = -1$ is the 1×1 Gram matrix. A negative scalar is an example of negative-definite matrix in $\mathbb{R}^{1\times1}$, which according to Mercer's theorem means that k is not a valid kernel.

Variational inference

Problem 15 [3 points] You are given a dataset consisting of N positive samples $\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$, $x_i \in \mathbb{R}^+$. You model the data-generating distribution (i.e., the likelihood) using a log-normal distribution, that is

$$p(x_i \mid \mu, \tau) = \text{Log-normal}(x_i \mid \mu, \tau),$$

where μ is the *known* and *fixed* mean parameter, and τ is the *unknown* precision parameter. You choose a Gamma distribution as the prior for τ, that is $p(\tau \mid a, b) = \text{Gamma}(\tau \mid a, b)$.

You would like to approximate the posterior $p(\tau \mid \boldsymbol{x}, \mu, a, b)$ using variational inference. Which of the following families of variational distributions $q(\tau)$ will yield the best approximation (in terms of KL-divergence)?

a) $q(\tau) = \text{Log-normal}(\tau \mid \nu, \beta) = \frac{\sqrt{\beta}}{\tau\sqrt{2\pi}} \exp\left(-\frac{\beta}{2}(\ln\tau - \nu)^2\right)$

b) $q(\tau) = \text{Normal}(\tau \mid \nu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2}(\tau - \nu)^2\right)$

c) $q(\tau) = \text{Gamma}(\tau \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)}\tau^{\alpha-1}\exp(-\beta\tau)$

d) $q(\tau) = \text{Inverse-gamma}(\tau \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)}\tau^{-\alpha-1}\exp\left(-\frac{\beta}{\tau}\right)$

Show your work! Just stating a), b), c) or d) is not enough!

Short answer:

For a Log-normal distribution with a known mean, Gamma is a conjugate prior for the precision. Therefore, the posterior of the precision is also a Gamma distribution. Thus, c) will yield the best approximation.

Long answer:

$$\begin{aligned} p(\tau \mid \boldsymbol{x}, \mu, a, b) &\propto \prod_i p(x_i \mid \mu, \tau) p(\tau \mid a, b) \\ &= \prod_i \left[\tau^{\frac{1}{2}} \exp\big(-\frac{\tau}{2}(\ln x_i - \mu)^2\big)\right] \tau^{a-1}\exp(-b\tau) \\ &= \tau^{\frac{N}{2}+a-1} \exp\left(-\big[\sum_i \frac{(\ln x_i - \mu)^2}{2} + b\big]\tau\right) \end{aligned}$$

We can see that the posterior matches a Gamma distribution up to the normalization constant:

$$p(\tau \mid \boldsymbol{x}, \mu, a, b) = \text{Gamma}\big(\tau \mid \frac{N}{2} + a, \sum_i \frac{(\ln x_i - \mu)^2}{2} + b\big)$$

Thus, c) will yield the best approximation (since the posterior is contained in the variational family (c), the KL divergence will be zero).