The Mathematics of Machine Learning
Homework Set 1

Due 18 February 2026 before 13:00
via Canvas

You are allowed to work on this homework in pairs. One person per pair submits the answers via Canvas. Make sure to put both names on the submission. Write your answers to theory exercises in LaTeX; for programming exercises submit a Jupyter notebook.

1 Theory Exercises

1. [4 pt]

   (a) [2 pt] What is the Bayes-optimal predictor f_{B} for binary classification with $Y \in \{-1, +1\}$ and the following cost-sensitive loss, which considers a false negative worse than a false positive?

$$L(Y, \hat{Y}) = \begin{cases} 0 & \text{if } \hat{Y} = Y, \\ 1 & \text{if } Y = -1 \text{ and } \hat{Y} = +1, \\ 10 & \text{if } Y = +1 \text{ and } \hat{Y} = -1. \end{cases}$$

   (b) [2 pt] For least-squares regression with the absolute error loss[1],

$$L(Y, \hat{Y}) = |Y - \hat{Y}|,$$

   the Bayes optimal predictor is such that $f_{\mathrm{B}}(X)$ is any median of Y under $P^*(Y|X)$. This follows from the following lemma:

   Lemma 1. *For any random variable Y with distribution P,*

$$\mathbb{E}\left[\,|Y - c|\,\right]$$

   is minimized in c by any median of P.

[1] NB This is a common alternative to the squared error loss $L(Y, \hat{Y}) = (Y - \hat{Y})^2$ that we considered in the lecture. The absolute error is less sensitive to large errors, which may be an advantage if there may be outliers (extreme points with small probability).

Prove this lemma. You may use without proof that at least one median m always exists.
Hint 1: The median of any distribution P is any point m such that

$$P(Y \leq m) \geq \frac{1}{2} \qquad \text{and} \qquad P(Y \geq m) \geq \frac{1}{2}.$$

Hint 2: By symmetry, it is sufficient to show that if $c < m$ and m is a median of P, then

$$\mathbb{E}[|Y - c|] \geq \mathbb{E}[|Y - m|].$$

(You do not have to prove this.)
Hint 3: Let $\mathbf{1}\{A\}$ be the indicator for any event A, which is 1 if A happens and 0 otherwise. Show that, if $c < m$, then

$$\begin{aligned}\mathbb{E}[|Y - c|] - \mathbb{E}[|Y - m|] = \mathbb{E}[(c - m)\mathbf{1}\{Y \leq c\}] \\ + \mathbb{E}[(2Y - m - c)\mathbf{1}\{c < Y < m\}] \\ + \mathbb{E}[(m - c)\mathbf{1}\{Y \geq m\}],\end{aligned}$$

and find a simpler lower bound on this expression using that $Y \geq c$ in the middle case.
Hint 4: Use the properties of the median to show that the lower bound from Hint 3 is non-negative.

2 Programming Exercise

The following programming exercise is to be implemented in Python, using a Jupyter notebook. As a starting point, you may use the notebook `Homework1-start.ipynb`, which is available from the course website.

Setup The notebook simulates a training set of size $N = 100$ and a test set of size 10 000 for binary classification with $X \in \mathbb{R}^2$ and $Y \in \{-1, +1\}$ by sampling from the distribution P^* defined via:

$$\begin{aligned}P^*(Y = +1) &= 0.8 \\ P^*(X|Y) &= \mathcal{N}\Big(\begin{pmatrix} Y \\ -Y \end{pmatrix}, 1.3I\Big),\end{aligned}$$

where $\mathcal{N}(\mu, \Sigma)$ denotes a multi-variate normal distribution with mean μ and co-variance matrix Σ.

The notebook further plots the data (both training and test sets together) and shows how to apply a 15-nearest neighbor classifier to it. It further demonstrates how to evaluate the density of a multi-variate normal distribution.

2. [4 pt]

   (a) [2 pt] Similar to Figure 2.4 in the book, plot the error = average 0/1-loss both on the training set and on the test set for the K-nearest neighbor classifier as a function of $K = 40, \ldots, 1$.
   Hint: if you are using pyplot for plotting, then `plt.gca().invert_xaxis()` *reverses the direction of the horizontal axis, so you can make it decreasing in K.*

   (b) [2 pt] Derive a way to compute the Bayes-optimal classifier f_{B} for the 0/1-loss, and add its average error on the test set as a horizontal line to the plot.
   Hint to calculate f_{B}: For any given x, we need to determine whether $P^*(Y = +1|X = x) \geq P^*(Y = -1|X = x)$ or not. Since X is a continuous variable, we need to interpret this in terms of densities to make sense. Then, by Bayes' rule,

$$P^*(Y = y|X = x) = \frac{P^*(Y = y)\phi(x; \mu_y, \Sigma)}{\sum_{y' \in \{-1,+1\}} P^*(Y = y')\phi(x; \mu_{y'}, \Sigma)},$$

   where $\phi(x; \mu, \Sigma)$ is the density of $\mathcal{N}(\mu, \Sigma)$ at point x, $\mu_y = \begin{pmatrix} y \\ -y \end{pmatrix}$ and $\Sigma = 1.3I$.