Machine Learning 1 — Mock Exam

1 Preliminaries

- Please write your immatriculation number **but not your name** on *every* page you hand in.
- The exam is closed book. You may, however, take one A4 sheet of handwritten notes.
- The exam is limited to 2×60 minutes.
- If a question says "Describe in 2–3 sentences" or "Show your work" or something similar, these mean the same: give a succinct description or explanation.
- This exam consists of 4 pages, 10 problems. You can earn up to 23 points.

Problem 0 [$\sqrt{-4}$ points] We suffer from gender bias. It's not clear which gender we favour in scoring, but it certainly—so science tells us—influences our grading skills. Help yourself by *not* writing your name on your sheets. Just fill in your immatriculation number on every sheet you hand in. Make sure it is easily readable.

2 Linear Algebra and Probability Theory

Problem 1 [2 points] Let A be a real, invertible matrix. Show that

$$\det(A^{-1}) = \det(A)^{-1}$$

Hint: make use of the identity $\det(BC) = \det(B)\det(C)$ *for square matrices* B *and* C*.*

We set $B = A$ and $C = A^{-1}$, and observe that $BC = I$ with I being the identity matrix. Hence:

$$\det(A)\det(A^{-1}) = \det(AA^{-1}) = \det(I) = 1$$
$$\Rightarrow \det(A^{-1}) = \frac{1}{\det(A)} = \det(A)^{-1} \qquad \square$$

3 Decision Trees

Problem 2 [5 points] Real world datasets are not always perfect—some contain systematic errors such as duplicated attributes or attributes with only one value. Furthermore, not all machine learning algorithms are well-suited to handling such errors. For example, Naive Bayes performs poorly when attributes are duplicated.

When working with decision trees, do duplicated attributes or one-value attributes affect the tree? Answer YES or NO for both types (duplicated and one-value) and explain why for each. Be sure to use Information Gain (i.e., Mutual Information) in your explanation.

A decision tree would not be affected by a duplicate nor single-valued variable.

One valued attribute: Suppose a split at a one-valued attribute is being considered at any stage in the tree formation. In this case, all the data, will be associated with a single child node which would be identical to the root node for the split. Therefore, the information gain for that split would be zero, making the one-valued attribute the least likely candidate for a split.

Duplicate attribute: Suppose A1 and A2 are the attributes which are duplicates of each other. If neither of them are in the decision tree and are being considered for a split, we can see that their duplicate nature means that they will induce the same partition on the data after the split. As a result, their information gain will be the same. Any one of them can be chosen for the split.

If A1 is already in the tree and A2 is being considered for a split, the root node of the split already contains a value (or a range of values) of A1. If A1 takes a single value at the root, then A2 must also take a single value and from the previous case, we know that the information gain due to A2 would be zero. If A1 takes a range of values at the root node of the split, then A2 will have non-negative information gain and could allow further improvement in the training error of the decision tree. (Note: This could have been accomplished by further splitting on A1 as well.)

4 Linear regression

Here we explore a regression model where the noise variance is a function of the input (variance increases as a function of input). Specifically

$$y = wx + \epsilon$$

where $\sigma > 0$, the noise ϵ is normally distributed with mean 0 and standard deviation σx and both $w \in \mathbb{R}$ and $x \in \mathbb{R}$. The value of σ is assumed to be known. We can write the model more compactly as $y \sim \mathcal{N}(wx, \sigma^2 x^2)$.

Problem 3 [1 point] True or false? We can get multiple solutions if we find an optimal w by optimising the sum of squared errors using gradient descent.

$$L(w) = \prod_n \mathcal{N}(y_n | wx_n, \sigma^2 x_n^2) \rightarrow \tag{1}$$

$$NLL = \sum_n \frac{(wx_n - y_n)^2}{2\sigma^2 x_n^2} + const(w) \rightarrow \tag{2}$$

$$0 = \frac{\partial}{\partial w} NLL = \sum_n \frac{2(wx_n - y_n)x_n}{2\sigma^2 x_n^2} \tag{3}$$

$$= \sum_n \frac{w}{\sigma^2} - \frac{y_n}{\sigma^2 x_n} \rightarrow \tag{4}$$

$$w = \frac{1}{N} \sum_n \frac{y_n}{x_n} \tag{5}$$

So we have a unique global ML solution, no local optima, so we could as well find it via gradient descent. So the answer is "FALSE".

Problem 4 [1 point] For $w = 1$ what would a plot for sampled pairs (x, y) look like? *Sketch* the plot for $0 \leq x \leq 3$.

It should be an expanding cone looking plot where one can easily fit a line since y is a function of x. The dots should be distributed in a Gaussian fashion on both sides of the functional line in accordance with the mean of the distributed points being on said line and the variance of the point increasing as x increases.

Problem 5 [2 points] How is the ratio y/x distributed for a fixed (constant) x? Show your work!

Dividing or multiplying a Gaussian by a constant will result in another Gaussian. So if $y \sim N(wx, \sigma^2 x^2)$ then $y/x \sim N(wx/x, \sigma^2 x^2/x^2)$ or $y/x \sim N(w, \sigma^2)$

Suppose we now have n training points and targets $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, where each x_i is chosen at random and the corresponding y_i is subsequently sampled from $y \sim \mathcal{N}(w^* x_i, \sigma^2 x_i^2)$, with some true underlying parameter value w^*—the value of σ^2 is the same as in our model.

Problem 6 [3 points] What is the maximum likelihood estimate of w as a function of the training data? Show your work.

From the previous problem we see that $y/x \sim N(w, \sigma^2)$. Since we have both y_i and x_i and the maximum likelihood estimate of a Gaussian is the mean, we can calculate $w^* = \frac{1}{n} y_i/x_i$

Problem 7 [2 points] What is the variance of this estimator due to noise in the target outputs as a function of n and σ^2 for *fixed* inputs $x_1, \ldots, x_n$?

The variance of the sample mean of a Gaussian is σ^2/n

5 Neural networks

Problem 8 [2 points] Suppose you have a deep neural network with sigmoid activation functions $\phi(x) = 1/(1 + \exp(-x))$ on the hidden units. Can your network learn if you only have biases (i.e., inputs with a constant value) to the first hidden layer, but not to the other hidden layers or the output layer?

Sure. The bias can be easily learned to be propagated to the higher layers.

Problem 9 [3 points] Suppose you have a deep neural network with sigmoid activation functions $\phi(x) = 1/(1+\exp(-x))$ on the hidden units. Can your network learn if you have no biases (i.e., inputs with a constant value) at all?

Sure. Suppose you have a hidden unit with all ingoing weights being 0. The output of this unit will be 1/2. That unit can therefore act as a bias. So you don't need them.

6 Kernels

Problem 10 [2 points] It is known that the function $k_1(x, x') = (\sigma_0^2 + xx')^p$ is a valid kernel (covariance function), where p is a positive integer and $\sigma_0 \in \mathbb{R}$ is a constant. Is the function $k_2(x, x') = -x^2x'^2 - 2xx' - 1$ also a valid kernel? Write down the deduction.

No. Since k_1 is a valid kernel, k_1 is positive semidefinite. Therefore, $k_2 = -k_1$ with $p = 2, \sigma_0 = 1$ is negative semidefinite and thus cannot be a kernel.

Alternative solutions are fine.