Exam of
Fundamental Principles of Machine Learning
december 17, 2021 – 3h

DON'T RETURN THIS SHEET BEFORE YOU ARE ALLOWED TO DO SO!
(you can write your name on the copies in the meantime)

General advice:

- **Authorized documents: 6 pages of personal notes.**
- Do not hesitate to do the exercises in any order you like: start with the ones you feel are quick to deal with.
- When you are allowed to start, before you start the first exercise, go through the subject quickly. In each exercise, the most difficult question is not necessarily the last, please feel free to skip some questions. Don't hesitate to go and scrap off points where they are easy to take. (vous pouvez "aller grapiller les points")
- **French:** Vous êtes autorisés à composer en Français. (Y compris en insérant des mots techniques comme overfitting ou regularization en anglais quand vous ne savez pas la traduction).
- **French:** si certains bouts de l'énoncé ne sont pas clairs, je peux les traduire ! N'hésitez pas à demander si vous n'êtes pas sûrs.
- Calculators not allowed (and useless). No electronic device allowed (cell phone, etc).
- At the end, we will collect your papers. You can leave after you have returned your copy.
- It is forbidden to lick the paper to stick it. Anonymization is canceled this year. (covid..)

DON'T RETURN THIS SHEET BEFORE YOU ARE ALLOWED TO DO SO!
(you can write your name on the copies in the meantime)

1 Lecture related questions (8.5 points)

1. (1 pt) When you have overfitting, what are the things you can do? (assuming we keep the same family of models). Cite as many possible solutions as you know, and each time, quickly explain your choice (1-2 lines per "solution" to overfitting).

2. (0.5 pt) When you have underfitting, what are the things you can do? (assuming we keep the same family of models). Cite as many possible solutions as you know, and each time, quickly explain your choice (1-2 lines per "solution" to overfitting).

3. (0.5 pt) What are the purpose of the validation and test sets?

4. (0.5 pt) Define what is a feature map.

5. (0.5 pt) Define what is a Kernel.

6. (0.5 pt) What is the logical link between large weights (after learning) and overfitting? (in the general case).

7. (1 pt) Explain the purpose of Ridge regression, the idea behind it, and how/why it works (Ridge is the one with the term $+\lambda||w||_2^2$).

8. (1.5 pt) Recall the definition of the SVM: what is the definition of the margin, what do we want to maximize, and where does the name "SVM" comes from?

9. (0.5 pt) Explain, with 1 or 2 simple schematic plots, why the Lasso regularization can lead to some of the weights being exactly 0. Hint: for simpicity, you may want to use the idea of a 1-Dimensional data set.

10. We recall the PCA compression and decompression formula, assuming the matrix P is of size $p \times d$, where $p < d$ is the dimension after PCA, and d is the original dimension of each data point.
The PCA compression formula is : $\vec{x}' = \vec{x}_{compressed} = P(\vec{x} - \langle \vec{x} \rangle)$
The decompression formula is : $\vec{x}_{decompressed} = P^T \vec{x}' + \langle \vec{x} \rangle$.
Where $\langle \vec{x} \rangle$ is the empirical average of the data, feature by feature. (There were typos in the slides!)

    - (0.5 pt) How come there are d dimensions in the reconstructed output, although we went through an intermediate object of dimension $p < d$? Are all these d components independent?
    - (0.5 pt) Does the matrix P have an inverse (in general)? Is this a deep fact, or is it happening “by chance”?

11. (1 pt) Compute the covariance matrix of the following (very small) data set:
$$X = \left\{ \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 4 \\ -1 \end{pmatrix}, \begin{pmatrix} -3 \\ -6 \\ 1 \end{pmatrix} \right\}$$
(Notes: giving the detailed formula and sketching the computation, without doing all the additions and multiplications, already gives you half the point. You can keep the full computation for the end of the exam, since it takes time and does not "pay well" in terms of points).

2 MAP (3 points)

We want to compute the MAP estimation of the parameter λ for a random variable X that is expected to follow an exponential law, $\rho(x) = \lambda e^{-\lambda x}$, where we have an exponential prior for the parameter λ: $\rho(\lambda) = \tau e^{-\lambda \tau}$. We recall that the expetation value for an exponential random variable X is $E[X] = 1/\lambda$.

1. (2.5 pt) Do compute the MAP from the start, i.e. from its definition, recalling the fundamental steps (that are general to all distributions) as well as the computations that apply to this precise case.

2. (0.25 pt) Interpret (comment on) the limit $N \to \infty$.

3. (0.25 pt) Interpret (comment on) the limit $\tau \to 0$.

3 Gradient Descent (1.5 pt)

In figure 1, we plot the "Loss" function $L(w) = w^4 + w^3$ (here there is no data appearing, the numbers have already been plugged in... don't overthink it, $L(w)$ is just a function).

1. (0.25 pt) What are the local extrema of this Loss function (compute them, do not simply read the graph).
2. (0.25 pt) Where would Gradient Descent end, ideally?
3. (0.5 pt) What happens, if I start at $w = 1$, with a learning rate of $\eta < 1/7$? (compute the first GD update from that point).
4. (0.25 pt) What happens if I start at $w = -0.2$, with a small enough η? (answer qualitatively, with a drawing)
5. (0.25 pt) Draw also the case where η is way too large. (answer qualitatively, with a drawing)

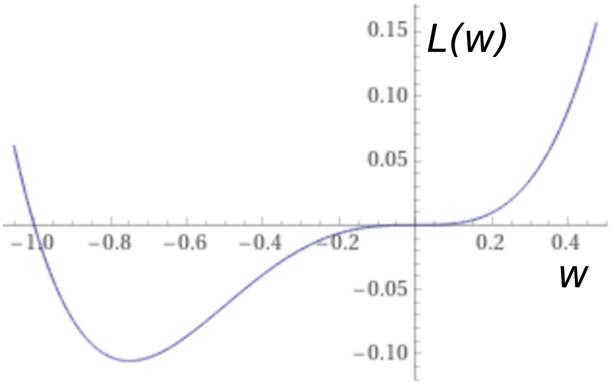

Figure 1: Loss $L(w)$, cf. exercise 3

4 A simple Loss (4.5 points)

We assume that we have real inputs $x \in \mathbb{R}^d$ and that the ground truth is also real-valued: $t_n \in \mathbb{R}$. We use the mini-batch strategy, i.e. examples are grouped in batches of m examples (for instance $N = 10^4, m = 1000$) and each epoch is split in N/m sub-epochs. Each sub-epoch e (which take values $e = 0, 1, 2, \ldots, \frac{N}{m} - 1$) then consists in optimizing a cost function J_e over a batch of m examples with indices $\mathcal{D}_e = \{em, em + 1, \ldots, em + m - 1\}$.

$$J_e(\Theta, X, T) = \frac{1}{2m} \sum_{n \in \mathcal{D}_e} (\vec{w}\vec{x}_n - t_n)^2 + \frac{1}{4}\lambda ||\vec{w}||_4^4 \tag{1}$$

$$\tag{2}$$

The general definition of the p-norm of a D-dimensional vector $\vec{x}$ is : $||\vec{x}||_p = \left(\sum_d^D x_d^p\right)^{1/p}$.

1. (1 pt) Compute $\vec{\nabla}_\Theta J_e$.

2. (0.25 pt) Explicit the update rule of the parameters, assuming we perform a (mini-batch) Gradient Descent with learning rate η.

3. (1.25 pt) Make the explicit computation of one update, with numbers, using (just for this question) $\eta = 1$, $\lambda = 1$, starting from $\vec{w} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$, with a mini-batch of data of size $m = 2$:
$X_e = \left\{ \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ -1 \end{pmatrix} \right\}, T_e = (1, 2)$.

4. (0.5 pt) Re-write the generic update rule of parameters (question 2) using matrix operations (using no sum symbol, $\sum$, at all). You can use this expression in the question (3) if you like.

5. (0.5 pt) Re-write the generic update rule of parameters (question 2), assuming we used some feature map $\phi()$. What is changing?

6. (0.5 pt) Write down the pseudo code of a full epoch.

7. (0.5 pt) Compared to Ridge regularization, what kind of behavior do you expect from this regularization term we introduced?

5 Logistic Regression (2.5 pts)

The logistic function is an activation function: $\sigma(z) = \frac{1}{1+e^{-z}}$. Concretely, this means the output of the network after applying this activation function is $y = \sigma(f_\Theta(x))$. The network is a single layer (no hidden layer) network: $f_\Theta(x) = \vec{\theta}.\vec{x}$

1. (0.5 pt) (this question can be skipped) First, prove that the derivative of $\sigma(u(\theta))$ with respect to θ, for a function u that is supposed to be sufficiently regular, is: $\partial_\theta \sigma\big(u(\theta)\big) = \sigma\big(u(\theta)\big)\Big(1 - \sigma\big(u(\theta)\big)\Big)\partial_\theta u$.
   More explictly, calling $y = \sigma\big(u(\theta)\big)$, we have the elegant formula: $\frac{\partial}{\partial\theta} y = y(1-y)\frac{\partial}{\partial\theta} u$
2. (0.25 pt) What should be the encoding of the labels?
3. (0.25 pt) How many classes can we handle?
4. (0.5 pt) Write the prediction function, $\hat{y} =$?
5. (1 pt) Write the pseudo code of the fit function.