UNIVERSITY OF SOUTHAMPTON COMP6208W1

SEMESTER 2 EXAMINATION 2022/23

ADVANCED MACHINE LEARNING

Duration 120 mins (2 hours)

This paper is a WRITE-ON examination paper.

You ***must*** write your Student ID on this Page and must not write your name anywhere on the paper.

All answers should be written within the designated boxes in this examination paper and sufficient space is provided for each question.

If, for some reason, space is required to complete or correct an answer to a question, use the "Additional Space" provided on the facing or adjacent page to the question. Clearly indicate which question the answer corresponds to.

No credit will be given for answers presented elsewhere and without clear indication of to what question they correspond. Blue answer books may be used for scratch; they will be discarded without being looked at.

Answer all parts of the question in section A (40 marks)
and ALL three questions from section B (20 marks each)

Student ID:

Question	Mark	*Arithmetic checked*	*Double Marked*
A1	/40		
B2	/20		
B3	/20		
B4	/20		
Total:	/100		

University approved calculators MAY be used.

A foreign language translation dictionary (paper version) is permitted provided it contains no notes, additions or annotations.

9 page examination paper

 Version 1

Section A

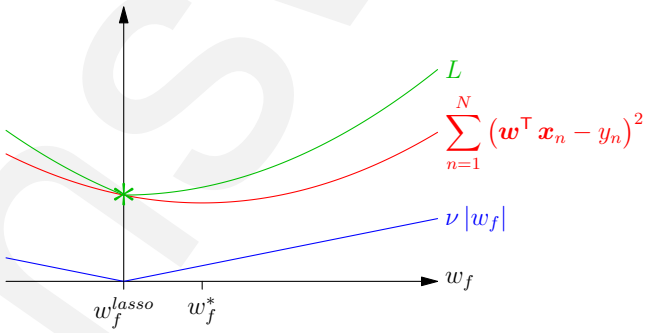
A 1

(a) A popular method for fitting a dataset $\{(\boldsymbol{x}_k, y_k)|k = 1, 2, \ldots, m\}$, where $\boldsymbol{x}_k \in \mathbb{R}^p$ are p-dimensional feature vectors and $y_k \in \mathbb{R}$ are targets, is to minimise a loss function, $\mathcal{L}(\boldsymbol{w})$, with an L_1 regulariser (the so called Lasso method)

$$\mathcal{L}(\boldsymbol{w}) = \sum_{k=1}^{m} \left(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_k - y_k\right)^2 + \nu \sum_{i=1}^{p} |w_i|,$$

where $\boldsymbol{w} = (w_1, w_2, \ldots, w_p)^{\mathsf{T}}$. Explain why this regulariser is used and either graphically or otherwise explain how it works. [5 marks]

The regulariser punishes large weights, but it does so in such a way that it finds a sparse solution. By adding a regulariser $|w_i|$ (for each component) to a quadratic function it will push the mimima to zero where otherwise the minimum value of w_i would be small provided the curvature is small.



(b) Explain why a convolutional operator acting on an image is approximately translationally equivariant and explain why it is not fully equivariant. [5 marks]

The convolutional operator is translationally equivariant in that if we translate the input (e.g. by shifting the camera) shift the output will be (approximately) translated. It is not fully equivariant both because of the edge effect and the effect of pixelation.

(c) Explain the kind of problems where the following learning machines excel

(i) Random Forest

(ii) Support Vector Machines

(iii) Convolutional Neural Networks

(iv) Hierarchical Bayesian Models

[5 marks]

(There are multiple correct answers. Below is an indicative answer only.)

(i) Random Forests work well with almost any tabular data where the features can be a mix of categorical and numerical data, with missing data and multiclass problems

(ii) Support Vector Machines work well on small dataset of clean data, e.g. all numerical data. Natively they work with binary classification although this can extended

(iii) Convolutional Neural Networks work well on data that has local spatial structure which is translationally invariant

(iv) Hierarchical Bayesian Models work well when the physical mechanism is well understood, although some parameters of the model are unknown.

(One mark each for every basic correct answer and an additional mark where at least part of the answer is more detailed.)

(d) Explain how Gradient Boosting works. [5 marks]

Gradient boosting is used on regression problems where a strong learner is generated by adding together weak learners (usually decision trees). The strong learner is built iteratively by training a weak learner on residual errors between the current strong learner and the targets. The weak learner is then added to the strong learner to create the next iteration of the strong learner.

(e) Consider the loss function

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^{\mathsf{T}}\mathbf{X}\mathbf{X}^{\mathsf{T}}\boldsymbol{w} + \nu\|\boldsymbol{w}\|$$

where $\mathbf{X}$ is the design matrix, $\boldsymbol{w}$ is a set of weights and $\nu > 0$ is a scalar. Explain why $\mathcal{L}(\boldsymbol{w})$ is convex? [5 marks]

The loss function $\mathcal{L}(\boldsymbol{w}) = \sum_i A_i(\boldsymbol{w})$ will be convex if each term $A_i(\boldsymbol{w})$ is convex. The first term is convex as the Hessian $\mathbf{X}\mathbf{X}^{\mathsf{T}}$ will be positive semi-definite. The second term is convex as any norm is convex with $\nu > 0$.

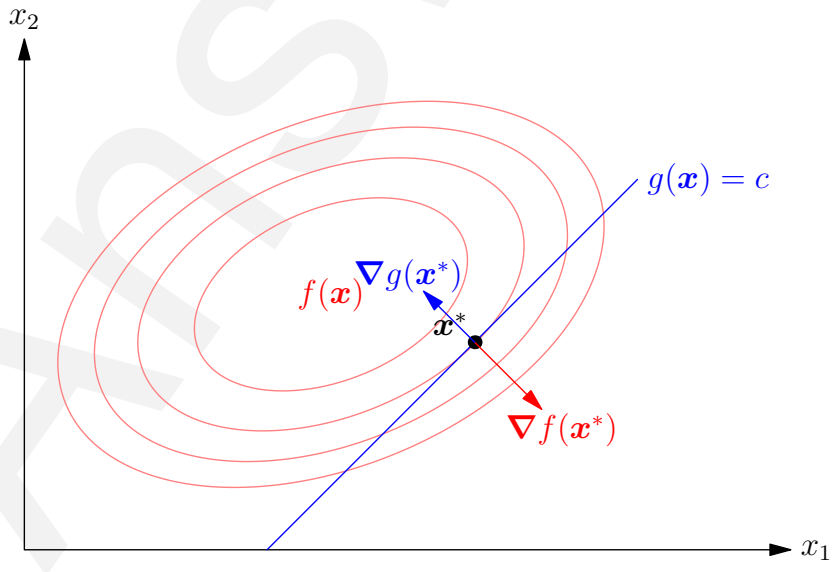
(f) Explain how to do model selection in Bayesian inference and explain why this is especial useful when using Gaussian Processes. [5 marks]

Bayesian inference uses a model, $\mathcal{M}$, through Bayes formula

$$p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M}) = \frac{p(\mathcal{D}|\boldsymbol{\theta},\mathcal{M})\,p(\theta|\mathcal{M})}{p(\mathcal{D}|\mathcal{M})}.$$

where the likelihood $p(\mathcal{D}|\boldsymbol{\theta},\mathcal{M})$ and prior $p(\theta|\mathcal{M})$ are conditioned the model. The marginal likelihood or evidence $p(\mathcal{D}|\mathcal{M})$ is a measure of how well the model, $\mathcal{M}$, explains the evidence. To select between models we can choose the model with the highest marginal likelihood (or we can put a prior on the models and compute the posterior probability of the model given the data). In GP we can computer the marginal likelihood in closed form making it very easy to select hyper-parameters.

(g) Below is shown contour lines for a function $f(\boldsymbol{x}) = \boldsymbol{x}^\mathsf{T}\mathbf{Q}\boldsymbol{x}$ that we wish to minimise subject to a linear constraint $g(\boldsymbol{x}) = c$. At the point $\boldsymbol{x}^*$ indicated, sketch the gradient $\boldsymbol{\nabla} f(\boldsymbol{x}^*)$ and $\boldsymbol{\nabla} g(\boldsymbol{x}^*)$. Use the diagram to explain why $\boldsymbol{\nabla}\mathcal{L} = 0$ finds the solution to the constrained optimisation problem where $\mathcal{L} = f(\boldsymbol{x}) - \lambda g(\boldsymbol{x})$. [5 marks]



(Note the directions are unique up to a change of sign.)

The constraint $\boldsymbol{\nabla}\mathcal{L} = 0$ picks points where $\boldsymbol{\nabla} f(\boldsymbol{x}) \propto \boldsymbol{\nabla} g(\boldsymbol{x})$. As the gradient is orthogonal to the contour lines this finds points where the constraint (a contour line) is parallel to the contour of $f(\boldsymbol{x})$, which is the condition for a local extremum.

(h) Describe in words what the Wasserstein distance measures. [5 marks]

The Wasserstein distance measures the minimum shifting of probability mass (that is probability mass time distance) required to transform one probability distribution to another.

End of question A1

Section B

B 2

(a) We can write the loss function for ridge regression as

$$\mathcal{L}(\boldsymbol{w}) = \|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 + \nu\|\boldsymbol{w}\|_2^2,$$

where **X** is the design matrix and $\boldsymbol{y}$ is a vector of target values.

(i) Calculate the weights, $\boldsymbol{w}^*$ that minimises the loss function.

(ii) Using the singular value decomposition $\mathbf{X} = \mathbf{USV}^\mathsf{T}$ write the optimal weight vector $\boldsymbol{w}^*$ in terms of **V**, **U** and **S**.

(iii) Hence show that $\boldsymbol{w}^* = \mathbf{V}\hat{\mathbf{S}}^+\mathbf{U}^\mathsf{T}\boldsymbol{y}$, where $\hat{\mathbf{S}}^+$ is a diagonal matrix with non-zero elements $\hat{S}^+_{ii} = s_i/(s_i^2 + \nu)$, and s_i are the singular values of **X** (i.e. $s_i = S_{ii}$).

(5 marks for each sub-part) [15 marks]

The gradient of the loss is

$$\nabla\mathcal{L}(\boldsymbol{w}) = 2\mathbf{X}^\mathsf{T}\mathbf{X}\boldsymbol{w} - 2\,\mathbf{X}^\mathsf{T}\boldsymbol{y} + 2\,\nu\,\boldsymbol{w}.$$

Setting this to zero and solving for w we find

$$\boldsymbol{w}^* = \left(\mathbf{X}^\mathsf{T}\mathbf{X} + \nu\,\boldsymbol{I}\right)^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{y}.$$

(5 marks for deriving this.)

Substituting in $\mathbf{X} = \mathbf{USV}^\mathsf{T}$ then

$$\mathbf{X}^\mathsf{T}\mathbf{X} + \nu\,\boldsymbol{I} = \mathbf{V}\left(\mathbf{S}^\mathsf{T}\mathbf{S} + \nu\,\mathbf{I}\right)\mathbf{V}^\mathsf{T}$$

so that

$$\boldsymbol{w}^* = \mathbf{V}\left(\mathbf{S}^\mathsf{T}\mathbf{S} + \nu\,\mathbf{I}\right)^{-1}\mathbf{S}^\mathsf{T}\mathbf{U}^\mathsf{T}\boldsymbol{y}$$

where we have used the tact that V and U are orthogonal matrices so that $\mathbf{VV}^\mathsf{T} = \mathbf{I}$ and $\mathbf{U}^\mathsf{T}\mathbf{U} = \mathbf{I}$. *(5 marks for this part)*

Defining

$$\hat{\mathbf{S}}^+ = \left(\mathbf{S}^\mathsf{T}\mathbf{S} + \nu\,\mathbf{I}\right)^{-1}\mathbf{S}^\mathsf{T}$$

then $w^* = \mathbf{V}\hat{\mathbf{S}}^+, \mathbf{U}^\mathsf{T}y$. Since S is a matrix that is zero everywhere except down the diagonal then $\mathbf{S}^\mathsf{T}\mathbf{S}$ will be a diagonal matrix and $\hat{\mathbf{S}}^+$ will only have non-zero elements along the diagonal equal to $\hat{S}^+_{ii} = s_i/(s_i^2 + \nu)$.
(5 marks for this part)

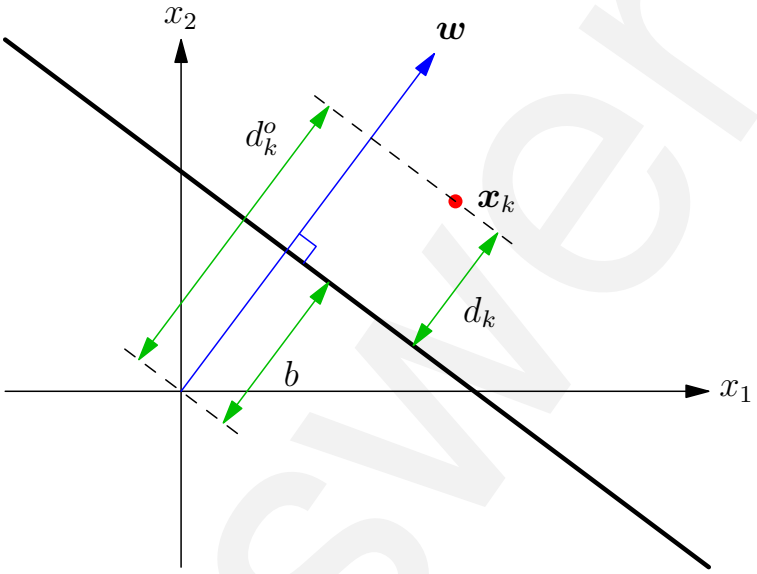
(b) Use the result that you derive to explain how adding the L_2 regulariser $\nu \|w\|_2^2$ improves the conditioning of the solution and is likely to improve generalisation. [5 marks]

The effect of adding a regulariser is to change the usual pseudo-inverse ($\mathbf{S}^+ = (\mathbf{S}^\mathsf{T}\mathbf{S})^{-1}\mathbf{S}^\mathsf{T}$ with diagonal components $S^+_{ii} = s_i^{-1}$) to $\hat{S}^+_{ii} = s_i/(s_i^2 + \nu)$. If for any singular value $s_i = 0$ the problem would be ill-posed and there would be an infinity of solutions—the pseudo-inverse is not defined. The problem now has a unique solution. That is, if $s_i = 0$ then $\hat{S}^+_{ii} = 0$ (rather than infinity without the regularisation term). Similarly if s_i is small $S^+_{ii} = s_i^{-1}$ would be very large and the pseudo-inverse poorly conditions (very sensitive to the training data—resulting in the machine having a large variance). However, if $s_i < \nu$ then $\hat{S}^+_{ii} < 1$ so that the regularised solution is much better conditioned.

End of question B2

B 3

(a) Write down a formula for the minimum distance, d_k^0, between $\boldsymbol{x}_k$ and a hyperplane through the origin perpendicular to $\boldsymbol{w}$, and the minimum distance d_k from $\boldsymbol{x}_k$ to the hyperplane perpendicular to $\boldsymbol{w}$ displaced by b. [5 marks]



$d_k^o = \frac{\boldsymbol{x}_k^T \boldsymbol{w}}{\|\boldsymbol{w}\|}$

$d_k = \frac{\boldsymbol{x}_k^T \boldsymbol{w}}{\|\boldsymbol{w}\|} - b$

(b) Depending on the category $y_k \in \{-1, 1\}$, write down the condition for a data point to be at least a distance $\gamma > 0$ above (or below if $y_k = -1$) the hyperplane shown in part (a). [5 marks]

$$y_k \left(\frac{\boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w}}{\|\boldsymbol{w}\|} - b \right) \geq \gamma$$

(c) Define $\boldsymbol{w'} = \boldsymbol{w}/(\gamma \|\boldsymbol{w}\|)$ and $b' = b/\gamma$ to rewrite the condition from part (b) and explain why minimising $\|\boldsymbol{w}'\|^2$ is equivalent to maximising the margin γ. [5 marks]

$$y_k \left(\boldsymbol{x}_k^\mathsf{T} \boldsymbol{w}' - b'\right) \geq 1$$

$\|\boldsymbol{w}'\|^2 = 1/\gamma^2$**, thus minimising** $\|\boldsymbol{w}'\|$ **is equivalent to maximising** γ**.**

(d) Write down a Lagrangian for finding the maximal margin hyperplane for an SVM given data $(\boldsymbol{x}_k, y_k)$ for $k = 1, 2, \ldots, m$. [5 marks]

$$\mathcal{L}(\boldsymbol{w}', b', \boldsymbol{\alpha}) = \frac{\|\boldsymbol{w}'\|^2}{2} - \sum_{k=1}^{m} \alpha_k \left(y_k \left(\boldsymbol{x}_k^\mathsf{T} \boldsymbol{w}' - b'\right) - 1\right)$$

where maximising the first term minimises the margin and the second term imposes the constraint that the data points lie outside the margin (α_k **are Lagrange multipliers).**

End of question B3

B 4

(a) In a variational auto-encoder an input image, $\boldsymbol{x}$, is drawn from a dataset $\mathcal{D}$. The encoder generates a probability distribution $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ defined in a latent space. A vector in the latent space, $\boldsymbol{z}$, is sampled from $q_{\phi}(\boldsymbol{z}|\boldsymbol{x})$ and sent to the decoder that then generates a probability distribution in the space of images $p_{\boldsymbol{\theta}}(\boldsymbol{x}'|\boldsymbol{z})$. The loss function is defined as

$$\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}\big[\mathrm{KL}\big(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})\big\|\mathcal{N}(\mathbf{0}, \mathbf{I})\big) - \log(p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}))\big]$$

Provide an interpretation of the two terms in the loss function. [10 marks]

The first term is a KL-divergence or relative entropy between the encoding probability, $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$, and a unit normal distribution. It measure the length in nats of the encoding $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ where the underlying probability is assumed to be normally distributed. This term punishes distributions that are different from $\mathcal{N}(0, \mathbf{I})$ and particularly distributions with very small variance. The second term can be seen as the description length (in nats) of describing the input image x given the distribution $p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{z})$. It is small when the probability of sampling $\boldsymbol{x}$ from $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ is large. Minimising the loss finds a compromise between finding an expensive specific code and having to pay for generating images with large errors.

(b) By considering the second derivative show that $f(x) = -\log(x)$ is convex-up. [5 marks]

The second derivative of $-\log(x)$ is $1/x^2$ which is strictly positive. Thus $-\log(x)$ is convex-up.

(c) Jensen's inequality for convex-up function states that

$$\mathbb{E}\left[f(X)\right] \geq f(\mathbb{E}\left[X\right]).$$

Use this to show that for any two categorical distributions $\mathbb{P}(F = i) = f_i$ and $\mathbb{P}(G = i) = g_i$, the KL-divergence

$$\mathrm{KL}(F\|G) = -\sum_i f_i \log\left(\frac{g_i}{f_i}\right)$$

is non-negative. [5 marks]

We note that

$$\mathrm{KL}(F\|G) = \mathbb{E}_F\left[-\log\left(\frac{g_i}{f_i}\right)\right] \geq -\log\left(\mathbb{E}_F\left[\log\left(\frac{g_i}{f_i}\right)\right]\right)$$

by Jensen's inequality. Writing out the expectation

$$\mathrm{KL}(F\|G) \geq -\log\left(\sum_i f_i \frac{g_i}{f_i}\right) = -\log\left(\sum_i g_i\right) = \log(1) = 0.$$

End of question B4

END OF PAPER