Queen Mary
University of London

Late-Summer Examination period 2020

MTH793P: Advanced machine learning

You should attempt ALL questions. Marks available are shown next to the questions.

In completing this assessment, you may use books, notes, and the internet. You may use calculators and computers, but you should show your working for any calculations you do. You must not seek or obtain help from anyone else.

At the start of your work, please **copy out and sign** the following declaration:

> I declare that my submission is entirely my own, and I have not sought or obtained help from anyone else.

All work should be **handwritten**, and should **include your student number**.

You have **24 hours** in which to complete and submit this assessment. When you have finished your work:

- scan your work, convert it to a **single PDF file** and upload this using the upload tool on the QMplus page for the module;
- e-mail a copy to maths@qmul.ac.uk with your student number and the module code in the subject line;
- with your e-mail, include a photograph of the first page of your work together with either yourself or your student ID card.

You are not expected to spend a long time working on this assessment. We expect you to spend about **3 hours** to complete the assessment, plus the time taken to scan and upload your work. Please try to upload your work well before the end of the assessment period, in case you experience computer problems. **Only one attempt is allowed – once you have submitted your work, it is final**.

Examiners: M. Benning

The notation log refers to the natural logarithm. The function $\text{rank}(L)$ returns the rank of a matrix L. All computations should be done by hand where possible, with marks being awarded for intermediate steps in order to discourage computational aids.

Question 1 [35 marks].

(a) Explicitly derive the expectation for an absolutely continuous random variable X that follows an exponential distribution, i.e. the probability density functions reads

$$\rho(x) := \begin{cases} \lambda \exp(-\lambda x) & x \geq 0 \\ 0 & \text{otherwise} \end{cases},$$

where λ is one added to the seventh digit of your student ID. **[7]**

(b) Explicitly derive the variance for an absolutely continuous random variable X that follows an exponential distribution. **[7]**

(c) For a uniform (and absolutely continuous) random variable X on $[0, 1]$ compute the expectation of $f(X)$ for

$$f(x) := \begin{cases} d_1 \,(1 - \log(x)) & x \in [0, 1/d_2] \\ 0 & \text{otherwise} \end{cases},$$

where d_1 is one added to the eighth digit and d_2 is one added to the seventh digit of your student ID. Make use of the convention $0 \log(0) = 0$. **[7]**

(d) Compute the gradient of the function $J : [-1, 1]^n \to [-1, 1]^n$ defined as $J(x) := -\sum_{j=1}^n \sqrt{1 - x_j^2}$. **[7]**

(e) Explicitly derive the Bregman distance $D_J(x, y)$ with respect to the function J as defined in Question 1 (d). **[7]**

Solution:

(a) We compute the expectation for an absolutely continuous random variable X that follows an exponential distribution as follows:

$$\begin{aligned}\mathbb{E}_x[x] &= \int_{-\infty}^{\infty} x\rho(x)\,dx = \int_0^{\infty} x\lambda\exp(-\lambda x)\,dx\,,\\ &= -\int_{-\infty}^{\infty} x\left(\frac{d}{dx}\exp(-\lambda x)\right)\,dx = -\left[x\exp(-\lambda x)\right]_0^{\infty} + \int_0^{\infty}\exp(-\lambda x)\,dx\\ &= \left[-\frac{\exp(-\lambda x)}{\lambda}\right]_0^{\infty} = \frac{1}{\lambda}\,.\end{aligned}$$

We can subsequently substitute the student ID digit for λ.

This is a new exercise, which can easily be computed with help of the lecture notes.

(b) We compute the variance via

$$\begin{aligned}\mathrm{Var}_x[x] = \mathbb{E}_x[x^2] - (\mathbb{E}_x[x])^2 &= \int_0^{\infty} x^2\lambda\exp(-\lambda x)\,dx - \frac{1}{\lambda^2}\\ &= \left[-x^2\exp(-\lambda x)\right]_0^{\infty} + 2\int_0^{\infty} x\exp(-\lambda x)\,dx - \frac{1}{\lambda^2}\\ &= -2\int_0^{\infty} x\left(\frac{d}{dx}\frac{1}{\lambda}\exp(-\lambda x)\right)\,dx - \frac{1}{\lambda^2}\\ &= \left[-\frac{2x}{\lambda}\exp -\lambda x\right]_0^{\infty} + \frac{2}{\lambda}\int_0^{\infty} x\exp(-\lambda x)\,dx - \frac{1}{\lambda^2}\\ &= \frac{2}{\lambda^2} - \frac{1}{\lambda^2} = \frac{1}{\lambda^2}\,.\end{aligned}$$

This is a new exercise, which can easily be computed with help of the lecture notes.

(c) We compute

$$\begin{aligned}\mathbb{E}_x[f(x)] &= \int_0^1 f(x)\,dx = d_1\int_0^{\frac{1}{d_2}} 1 - \log(x)\,dx = d_1\left[2x - x\log x\right]_0^{\frac{1}{d_2}}\\ &= \frac{d_1}{d_2}\left(2 - \log\left(\frac{1}{d_2}\right)\right) = \frac{d_1}{d_2}\left(2 + \log\left(d_2\right)\right)\,.\end{aligned}$$

This exercise is similar to Exercise 1, Coursework 3

(d) It is straight-forward to compute

$$\nabla J(x) = \begin{pmatrix}\frac{x_1}{\sqrt{1-x_1^2}}\\ \frac{x_2}{\sqrt{1-x_2^2}}\\ \vdots\\ \frac{x_n}{\sqrt{1-x_n^2}}\end{pmatrix}\,.$$

This question can be answered based on basic calculus.

(e) Based on the previous exercise and the definition of the Bregman distance one computes

$$D_J(x, y) = \sum_{j=1}^{n} \left[\frac{1 - x_j y_j}{\sqrt{1 - y_j^2}} - \sqrt{1 - x_j^2} \right] .$$

This exercise is similar to Exercise 2, Coursework 3

Question 2 [35 marks].

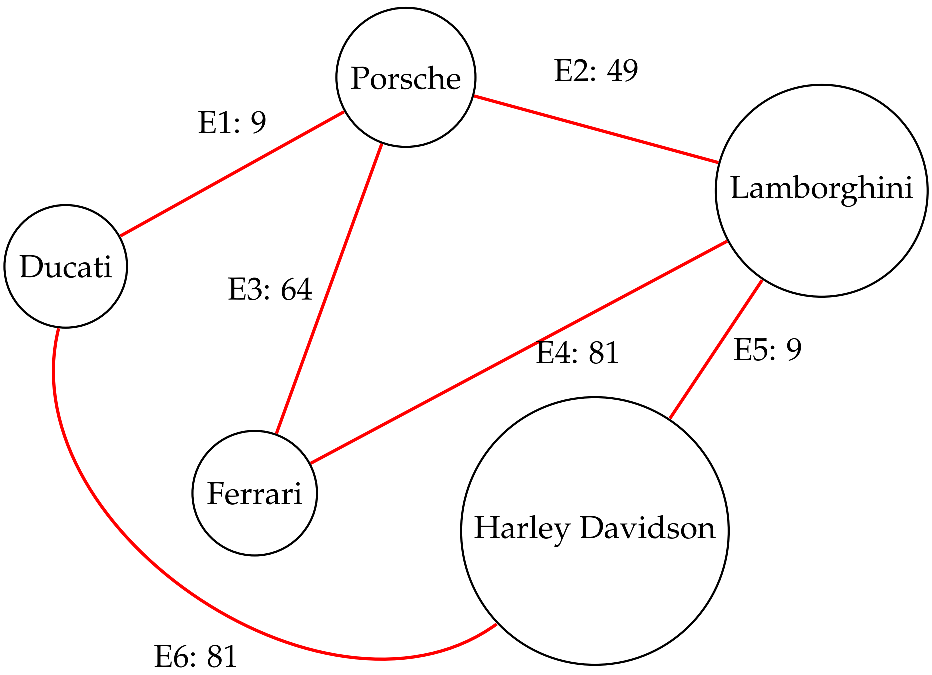
(a) You want to decide whether or not to ease lock-down restrictions. You base your decision on three factors:

- Are people required to wear face-masks when going outside?
- Have covid-19 case numbers dropped in previous weeks?
- Is comprehensive testing and tracing in place?

You do not ease lock-down measures unless case numbers have dropped. However, a drop in case numbers is only sufficient for easing lock-down measures if either people are required to wear face masks when going outside or if comprehensive testing and tracing is also in place.

Model this binary decision process with a perceptron and choose appropriate weights to mimic the decision process accurately. Justify your choice of weights. **[7]**

(b) Write down the incidence matrix for the following weighted, undirected graph:



Order the columns of the incidence matrix alphabetically according to the vertex name and the rows according to the edge numbering (E1, E2, E3, ...). **[6]**

(c) Compute the corresponding graph Laplacian for the incidence matrix in Question 2(b). **[7]**

(d) We want to use the graph from Question 2(b) to determine whether a node in the graph belongs to the class "cars" or the class "motorbikes". Suppose we are in a semi-supervised setting, where the node "Lamborghini" is already labelled $v_{\text{Lamborghini}} = 1$ (class "cars") and the node "Ducati" is labelled as $v_{\text{Ducati}} = 0$ (class "motorbikes"). Determine the labels for all remaining nodes following the procedure explained in the lecture notes, and classify each node. **[8]**

(e) Manually determine parameters $w_1 \in \mathbb{R}$, $w_2 \in \mathbb{R}^2$, $b_1 \in \mathbb{R}$ and $b_2 \in \mathbb{R}$ of a neural network of the form

$$f(x_1, x_2) = w_1 \max\left(0, w_2^\top \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + b_2\right) + b_1$$

so that it mimics the logical NAND function, i.e.

x_1	x_2	$f(x_1, x_2)$
0	0	1
1	0	1
0	1	1
1	1	0

.

[7]

Solution:

(a) This binary decision process can for example be modelled with the following perceptron:

$$f(x_1, x_2, x_3) = \begin{cases} 0 & 2x_1 + 4x_2 + 2x_3 \leq 5 \\ 1 & 2x_1 + 4x_2 + 2x_3 > 5 \end{cases},$$

where $x_1, x_2, x_3 \in \{0, 1\}$ represent variables associated to the three factors mentioned in the problem description. Different states are represented as follows:

$f(x_1, x_2, x_3)$	x_1	x_2	x_3
1	1	1	1
1	0	1	1
0	1	0	1
1	1	1	0
0	1	0	0
0	0	1	0
0	0	0	1
0	0	0	0

Hence, lock-down measures are not eased unless case numbers have dropped. However, a drop in case numbers is only sufficient for easing lock-down measures if either people are required to wear face masks when going outside or if comprehensive testing and tracing is also in place.

This exercise is similar to Exercise 3 on Coursework 3

(b) The incidence matrix for the displayed graph is

$$M_w = \left(\begin{array}{c|ccccc} \text{E1} & -3 & 0 & 0 & 0 & 3 \\ \text{E2} & 0 & 0 & 0 & -7 & 7 \\ \text{E3} & 0 & -8 & 0 & 0 & 8 \\ \text{E4} & 0 & -9 & 0 & 9 & 0 \\ \text{E5} & 0 & 0 & -3 & 3 & 0 \\ \text{E6} & -9 & 0 & 9 & 0 & 0 \\ \hline & \text{Ducati} & \text{Ferrari} & \text{Harley Davidson} & \text{Lamborghini} & \text{Porsche} \end{array}\right)$$

This question is similar to Exercise 2 on Coursework 4.

(c) The corresponding graph Laplacian then reads

$$L_w = M_w^\top M_w = \left(\begin{array}{c|ccccc} \text{Ducati} & 90 & 0 & -81 & 0 & -9 \\ \text{Ferrari} & 0 & 145 & 0 & -81 & -64 \\ \text{H. D.} & -81 & 0 & 90 & -9 & 0 \\ \text{Lamborghini} & 0 & -81 & -9 & 139 & -49 \\ \text{Porsche} & -9 & -64 & 0 & -49 & 122 \\ \hline & \text{Ducati} & \text{Ferrari} & \text{H. D.} & \text{Lamborghini} & \text{Porsche} \end{array}\right)$$

This question is similar to Exercise 2 on Coursework 4.

(d) From the lecture notes we know that the label vector $v \in \mathbb{R}^5$ can be decomposed as

$$v = P_{R^\perp}^\top \tilde{v} + P_R^\top y \,,$$

where P_R denotes the projection of v onto the known indices, and $P_{R^\perp}$ onto the unknown indices. The known indices are denoted by y, the unknown by $\tilde{v}$. For

$$v = \begin{pmatrix} v_{\text{Ducati}} \\ v_{\text{Ferrari}} \\ v_{\text{Harley Davidson}} \\ v_{\text{Lamborghini}} \\ v_{\text{Porsche}} \end{pmatrix}$$

we know the first and the forth entry; the first belongs to the class "motorbikes" and therefore takes on the value $v_{\text{Ducati}} = 0$, whereas the forth entry belongs to the class "cars", hence $v_{\text{Lamborghini}} = 0$. Thus, for $y = \begin{pmatrix} 1 & 0 \end{pmatrix}^\top$ we have

$$v = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tilde{v} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} .$$

From the lecture notes we also know that we can estimate $\tilde{v}$ via

$$\begin{aligned} \tilde{v} &= \arg\min_{\overline{v}} \left\| M_w \left(P_{R^\perp}^\top \overline{v} + P_R^\top y \right) \right\|^2 , \\ &= - \left(P_{R^\perp} L_w P_{R^\perp}^\top \right)^{-1} \left(P_{R^\perp} L_w P_R^\top y \right) , \end{aligned}$$

which for our matrices reads

$$\begin{pmatrix} 145 & 0 & -64 \\ 0 & 90 & 0 \\ -64 & 0 & 122 \end{pmatrix} \tilde{v} = \begin{pmatrix} 81 \\ 9 \\ 49 \end{pmatrix} ,$$

Solving this linear system leads to the (approximate) solution

$$\tilde{v} \approx \begin{pmatrix} 0.9576 \\ 0.1 \\ 0.9040 \end{pmatrix} .$$

Rounding all values above $1/2$ to one and below $1/2$ to zero then yields the classification

$$v = \begin{pmatrix} v_{\text{Ducati}} \\ v_{\text{Ferrari}} \\ v_{\text{Harley Davidson}} \\ v_{\text{Lamborghini}} \\ v_{\text{Porsche}} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} .$$

This question is similar to Question 3 of the mock exam.

(e) A possible choice of weights w_1, w_2 and bias b_1 and b_2 are

$$w_1 = -1, \quad w_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad b_1 = 1 \quad \text{and} \quad b_2 = -1\,.$$

This way we obtain $f(0,0) = 1 - \max(0,-1) = 1$, $f(1,0) = 1 - \max(0,0) = 1$, $f(0,1) = 1 - \max(0,0) = 1$ and $f(1,1) = 1 - \max(0,1) = 0$. I will accept any weights and biases as correct answers that yield $f(0,0) = 1$, $f(1,0) = 1$, $f(0,1) = 1$ and $f(1,1) = 0$.

This question is similar to Exercise 1 of Coursework 4.

Question 3 [30 marks].

(a) Perform two steps of k-means clustering by hand for the five data points $x_1 = -13$, $x_2 = -17$, $x_3 = 0$, $x_4 = 3$, and $x_5 = -2$. Assume $k = 2$ clusters and initialise your variables as

$$z_0 := \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}^\top ,$$

and

$$\mu_0 := \begin{pmatrix} d \\ 0 \end{pmatrix} ,$$

where d is one added to the eighth digit of your student ID number. For each iteration, update the variable z^l first, and then μ^l. Here $l \in \{1, 2\}$ denotes the iteration index. Does the iteration converge? **[8]**

(b) Find vectors $u \in \mathbb{R}^2$ and $v \in \mathbb{R}^3$ such that the following identity is satisfied for all known values:

$$uv^\top = \begin{pmatrix} -6 & 12 & ? \\ 2 & -4 & d \end{pmatrix} .$$

Here d is one added to the seventh digit of your student ID number. What value do you obtain at the missing entry denoted by a question mark? **[7]**

(c) Compute an approximation $\hat{L} \in \mathbb{R}^{2\times 3}$ with $\operatorname{rank}(\hat{L}) = 1$ of the matrix

$$X := d \begin{pmatrix} -2 & 3 & 2 \\ 2 & 2 & 3 \end{pmatrix}$$

by hand, where d is one added to the last digit of your student ID, that satisfies $\|\hat{L} - X\|_{\mathrm{Fro}} \leq \|L - X\|_{\mathrm{Fro}}$, for all $L \in \mathbb{R}^{2\times 3}$ with $\operatorname{rank}(L) = 1$. **[8]**

(d) Formulate a proximal gradient descent algorithm for a matrix completion problem of the form

$$(\hat{L}, \hat{S}) = \arg \min_{L,S \in \mathbb{R}^{s\times n}} \left\{ \frac{1}{2} \|P_\Omega(L + S) - y\|^2 + \alpha_1 \|L\|_* + \alpha_2 \|S\|_1 \right\} .$$

Here $P_\Omega \in \mathbb{R}^{s\times n} \to \mathbb{R}^r$ is a (known) projection operator that projects the known entries, specified by the index set Ω, of its argument to a vector of length r. The vector of known entries is denoted by $y \in \mathbb{R}^r$, α_1 and α_2 are positive constants, and $\|\cdot\|_*$ and $\|\cdot\|_1$ are the nuclear norm and the matrix one norm known from the lecture respectively. How does the algorithm compare to the robust PCA method introduced in the lecture? **[7]**

Solution:

(a) The update formulae for k-means clustering are

$$z_{ik}^{l+1} = \begin{cases} 1 & k = \arg\min_{j\in\{1,2\}} |x_i - \mu_j^l|^2 \\ 0 & \text{otherwise} \end{cases} ,$$

and

$$\mu^{l+1} = \frac{\sum_{i=1}^5 z_{ik}^{l+1} x_i}{\sum_{i=1}^5 z_{ik}^{l+1}} .$$

For $d = 2$ we compute

$$\begin{pmatrix} |x_1 - \mu_1^0|^2 & |x_2 - \mu_1^0|^2 & |x_3 - \mu_1^0|^2 & |x_4 - \mu_1^0|^2 & |x_5 - \mu_1^0|^2 \\ |x_1 - \mu_2^0|^2 & |x_2 - \mu_2^0|^2 & |x_3 - \mu_2^0|^2 & |x_4 - \mu_2^0|^2 & |x_5 - \mu_2^0|^2 \end{pmatrix}$$
$$= \begin{pmatrix} 225 & 361 & 4 & 1 & 16 \\ 169 & 289 & 0 & 9 & 4 \end{pmatrix}$$

for the squared differences. Hence, we compute

$$z^1 = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \end{pmatrix} ,$$

and, consequently, also

$$\mu^1 = \begin{pmatrix} \frac{\sum_{i=1}^5 z_{i1}^1 x_i}{\sum_{i=1}^5 z_{i1}^1} \\ \frac{\sum_{i=1}^5 z_{i2}^1 x_i}{\sum_{i=1}^5 z_{i2}^1} \end{pmatrix} = \begin{pmatrix} 3 \\ \frac{-13-17-2}{4} \end{pmatrix} = \begin{pmatrix} 3 \\ -8 \end{pmatrix} .$$

This completes the first iteration. Computing the squared differences for the second iteration then yields

$$\begin{pmatrix} |x_1 - \mu_1^1|^2 & |x_2 - \mu_1^1|^2 & |x_3 - \mu_1^1|^2 & |x_4 - \mu_1^1|^2 & |x_5 - \mu_1^1|^2 \\ |x_1 - \mu_2^1|^2 & |x_2 - \mu_2^1|^2 & |x_3 - \mu_2^1|^2 & |x_4 - \mu_2^1|^2 & |x_5 - \mu_2^1|^2 \end{pmatrix}$$
$$= \begin{pmatrix} 256 & 400 & 9 & 0 & 25 \\ 25 & 81 & 64 & 121 & 36 \end{pmatrix} ,$$

for which we obtain

$$z^2 = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix} ,$$

and

$$\mu = \begin{pmatrix} \frac{1}{3} \\ -15 \end{pmatrix} .$$

Since $z^2 \neq z^1$, we cannot say at this stage that we have converged. Computing another iterate, however, reveals that the algorithm converges after three iterations.

This question is similar to Exercise 1 on Coursework 5.

(b) In order to satisfy the equality, the 2×3-matrix has to have rank one. Hence, if we choose

$$\begin{pmatrix} -6 & 12 & -3d \\ 2 & -4 & d \end{pmatrix},$$

we ensure that the entries of the first row are the entries of the second row multiplied by -3. This way both rows are linearly dependent, leading to a matrix of rank one. Two possible vectors u and v that satisfy

$$uv^\top = \begin{pmatrix} -6 & 12 & -3d \\ 2 & -4 & d \end{pmatrix}$$

are $u = \begin{pmatrix} 3 & -1 \end{pmatrix}^\top$ and $v = \begin{pmatrix} -2 & 4 & -d \end{pmatrix}^\top$.

This question can be answered based on the lecture notes content.

(c) From the lecture notes we know that the best possible rank-one approximation in terms of the Frobenius norm can be computed by computing the (incomplete) singular value decomposition of X. Like in a similar coursework exercise, we compute the eigenvalues of $XX^\top$ by solving the characteristic polynomial $\det(XX^\top - \lambda I) = 0$, i.e.

$$\det(XX^\top - \lambda I) = \det\left(\begin{pmatrix} 17d^2 - \lambda & 8d^2 \\ 8d^2 & 17d^2 - \lambda \end{pmatrix}\right) = \lambda^2 - 34d^2\lambda + 225d^2,$$

whose solutions are $\lambda_1 = 25d^2$ and $\lambda_2 = 9d^2$. Since the singular values are $\sigma_i = \sqrt{\lambda_i}$ for $i = 1, 2$, we obtain $\sigma_1 = 5d$ and $\sigma_2 = 3d$. The best rank one approximation can be computed by computing $\tilde{X} = u_1 u_1^\top X$, where u_1 is the singular vector that corresponds to σ_1. We determine u_1 by computing the kernel of $XX^\top - \lambda_1 I$, i.e.

$$\ker(XX^\top - \lambda_1 I) = \ker\left(d^2 \begin{pmatrix} -8 & 8 \\ 8 & -8 \end{pmatrix}\right) = \left\{ t \begin{pmatrix} 1 \\ 1 \end{pmatrix} \,\middle|\, t \in \mathbb{R} \right\}.$$

Since $u_1 \in \ker(XX^\top - \lambda_1 I)$ has to have norm one, we easily compute

$$u_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

As a consequence, the best rank-one approximation of X in terms of the Frobenius norm is computed via

$$\begin{aligned} \tilde{X} = u_1 u_1^\top X &= \frac{d}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \end{pmatrix} \begin{pmatrix} -2 & 3 & 2 \\ 2 & 2 & 3 \end{pmatrix} \\ &= \frac{d}{2} \begin{pmatrix} 0 & 5 & 5 \\ 0 & 5 & 5 \end{pmatrix}. \end{aligned}$$

This exercise is similar to Exercise 2(1) on Coursework 5.

(d) A proximal gradient descent method for this problem can be formulated by splitting the objective into two parts

$$F(L,S) := \frac{1}{2}\|P_\Omega(L+S) - y\|^2 \qquad \text{and} \qquad R(L,S) := \alpha_1\|L\|_* + \alpha_2\|S\|_1\,,$$

and formulating

$$\begin{aligned}
\begin{pmatrix} L^{j+1} \\ S^{j+1} \end{pmatrix} &= (I + \tau\partial R)^{-1}\left(\begin{pmatrix} L^j \\ S^j \end{pmatrix} - \tau\,\nabla F(L^j, S^j)\right) \\
&= \begin{pmatrix} (I + \tau\alpha_1\partial\|\cdot\|_*)^{-1}\left(L^j - \tau P_\Omega^\top\left(P_\Omega(L^j + S^j) - y\right)\right) \\ (I + \tau\alpha_2\partial\|\cdot\|_1)^{-1}\left(S^j - \tau P_\Omega^\top\left(P_\Omega(L^j + S^j) - y\right)\right) \end{pmatrix} \\
&= \begin{pmatrix} \arg\min_{L\in\mathbb{R}^{s\times n}}\left\{\frac{1}{2}\left\|L - \left(L^j - \tau P_\Omega^\top\left(P_\Omega(L^j+S^j) - y\right)\right)\right\|_{\text{Fro}}^2 + \tau\alpha_1\|L\|_*\right\} \\ \arg\min_{S\in\mathbb{R}^{s\times n}}\left\{\frac{1}{2}\left\|S - \left(S^j - \tau P_\Omega^\top\left(P_\Omega(L^j+S^j) - y\right)\right)\right\|_{\text{Fro}}^2 + \tau\alpha_2\|S\|_1\right\} \end{pmatrix}\,,
\end{aligned}$$

for a step-size parameter $0 < \tau \leq 1$, and some initial values L^0 and S^0. Here j denotes the iteration index. If we define

$$X^j := \frac{1}{\tau}\begin{pmatrix} L^j - \tau P_\Omega^\top\left(P_\Omega(L^j + S^j) - y\right) \\ S^j - \tau P_\Omega^\top\left(P_\Omega(L^j + S^j) - y\right) \end{pmatrix}\,,$$

we can rewrite the algorithm to

$$\begin{aligned}
\begin{pmatrix} L^{j+1} \\ S^{j+1} \end{pmatrix} &= \begin{pmatrix} (I + \tau\alpha_1\partial\|\cdot\|_*)^{-1}\left(\tau X_1^j\right) \\ (I + \tau\alpha_2\partial\|\cdot\|_1)^{-1}\left(\tau X_2^j\right) \end{pmatrix}\,, \\
X^{j+1} &= \begin{pmatrix} \frac{1}{\tau}L^{j+1} - P_\Omega^\top\left(P_\Omega(L^{j+1} + S^{j+1}) - y\right) \\ \frac{1}{\tau}S^{j+1} - P_\Omega^\top\left(P_\Omega(L^{j+1} + S^{j+1}) - y\right) \end{pmatrix}\,,
\end{aligned}$$

as opposed to

$$\begin{aligned}
\begin{pmatrix} L^{j+1} \\ S^{j+1} \end{pmatrix} &= \begin{pmatrix} (I + \tau\alpha_1\partial\|\cdot\|_*)^{-1}\left(\tau X^j\right) \\ (I + \tau\alpha_2\partial\|\cdot\|_1)^{-1}\left(\tau X^j\right) \end{pmatrix}\,, \\
X^{j+1} &= X^j - (L^{j+1} + S^{j+1} - X)\,,
\end{aligned}$$

in case of robust PCA as introduced in the lecture.

This is a new exercise.

End of Paper.