DIPARTIMENTO DI ELETTRONICA, INFORMAZIONE E BIOINGEGNERIA
Politecnico di Milano
Machine Learning (Code: 097683)
July 17, 2019

Surname:	
Name:	
Student ID:	
Row:	Column:

Time: 2 hours 30 minutes — Prof. Marcello Restelli — Maximum Marks: 34

- The following exam is composed of **10 exercises** (one per page). The first page needs to be filled with your **name, surname and student ID**. The following pages should be used **only in the large squares** present on each page. Any solution provided either outside these spaces or **without a motivation** will not be considered for the final mark.
- During this exam you are **not allowed to use electronic devices**, such as laptops, smartphones, tablets and/or similar. As well, you are not allowed to bring with you any kind of note, book, written scheme and/or similar. You are also not allowed to communicate with other students during the exam.
- The first reported violation of the above mentioned rules will be annotated on the exam and will be considered for the final mark decision. The second reported violation of the above mentioned rules will imply the immediate expulsion of the student from the exam room and the **annulment of the exam.**
- You are allowed to write the exam either with a pen (black or blue) or a pencil. It is your responsibility to provide a readable solution. We will not be held accountable for accidental partial or total cancellation of the exam.
- The exam can be written either in **English** or **Italian**.
- You are allowed to withdraw from the exam at any time without any penalty. You are allowed to leave the room not early than half the time of the duration of the exam. You are not allowed to keep the text of the exam with you while leaving the room.
- **Three of the points will be given on the basis on how quick you are in solving the exam.** If you finish earlier than 45 min before the end of the exam you will get 3 points, if you finish earlier than 30 min you will get 2 points and if you finish earlier than 15 min you will get 1 point (the points cannot be accumulated).
- The box on Page 12 can only be used to complete the Exercises 9 and/or 10.

Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5	Ex. 6	Ex. 7	Ex. 8	Ex. 9	Ex. 10	Time	Tot.
/ 5	/ 5	/ 5	/ 2	/ 2	/ 2	/ 2	/ 2	/ 3	/ 3	/ 3	/ 34

Exercise 1 (5 marks)

What is the **VC dimension** of a hypothesis space? What can it be used for?

Exercise 2 (5 marks)

Explain the cross-validation procedure and for what it can be used.

Exercise 3 (5 marks)

Explain what are **eligibility traces** and describe the TD(λ) algorithm.

Exercise 4 (2 marks)

```
1 load data x t;
2 lambda = [1e-5, 0.01, 10];
3 L = lasso(x, t, 'Lambda', lambda); %Evaluate only on the vector
     lambda
4 for ii = 1:3
5     W(:, ii) = ridge(t, x, lambda(ii));
6 end
7 [~, sel_mod_r] = min(sum((W .* x - t).^2) + lambda .* W.^2);
8 [~, sel_mod_l] = min(sum((L .* x - t).^2) + lambda .* abs(W));
```

1. Describe the operations performed by the MatLab code snippet above;
2. Suggest the corrections to apply to the code to make it a proper ML procedure.

1.
    - Line 1: loading data for regression
    - Line 3: fitting LASSO regression with different regularization parameters $\lambda \in \{10^{-5}\ 0.01\ 10\}$
    - Line $4-6$: fitting RIDGE regression with different regularization parameters $\lambda \in \{10^{-5}\ 0.01\ 10\}$
    - Line $7-8$: selecting the best model among the LASSO ones and the RIDGE ones
2. The data should be normalized, for instance with the function `zscore()`, and the procedure to select the model is incorrect, since the performance should be tested on independent data and the performance indes should be the RSS (where here we evalutate the regularized loss).

Exercise 5 (2 marks)

Consider a generic linear regression model $y(\mathbf{x}_n, \mathbf{w}) = \mathbf{w}^\top \mathbf{x}_n$. Tell if one should use the Online Gradient Descent method to learn the optimal value for the parameter $\mathbf{w}$ in each one of the following 4 different situations. Motivate your answer.

1. Small number of parameters;
2. The loss function is $L(\mathbf{w}|\mathbf{x}_n, t_n) = (y(\mathbf{x}_n, \mathbf{w}) - t_n)^2 + \lambda \sum_i w_i^2$ $(\lambda \in \mathbb{R}^+)$;
3. Huge number of samples;
4. The loss function is $L(\mathbf{w}|\mathbf{x}_n, t_n) = \begin{cases} (y(\mathbf{x}_n, \mathbf{w}) - t_n)^2 & \text{if } |y(\mathbf{x}_n, \mathbf{w}) - t_n| < \delta \\ |y(\mathbf{x}_n, \mathbf{w}) - t_n| & \text{if } |y(\mathbf{x}_n, \mathbf{w}) - t_n| > \delta \end{cases}$, with $\delta \in \mathbb{R}^+$.

1. NO: the analytic solution provided by the Least Square method is a viable option when we have only few parameters, since its most computationally demanding part is the inversion of a matrix having dimension equal to the number of parameters;
2. NO: this is RIDGE regression, for which it exists a closed form solution for the optimal value of the parameters;
3. YES/NO: again the LS methods scales cubically with the number of the parameters and only linearly with the number of samples. If the dataset is so huge it cannot be process at the same time one might consider the Gradient Descent as a viable solution, considering only batches of data.
4. YES: since there does not exist a closed form solution (the loss function has some non differentiable points) for this loss, one of the possible solution is to apply Gradient Descent.

Exercise 6 (2 marks)

Tell if the following statements about solving MDPs are true or false. Motivate your answers.

1. In a finite state MDP there only exist optimal policies which are Markovian, stationary and deterministic;
2. If we observe a finite episode on an MDP, we should consider the undiscounted ($\gamma = 1$) cumulative reward as performance metric;
3. The Bellman Expectation Operator can be used to compute the optimal policy of an MDP;
4. There is a unique optimal policy in an MDP (if true provide a theoretical result, if false a counterexample).

1. FALSE: we are assured that Markovian, stationary and deterministic optimal policy, but there might be others which are still optimal but do not have the aforementioned properties;
2. FALSE: it might be an indefinite horizon problem, in which some of the episodes terminates and some other not. In this case, using the undiscounted cumulative reward as a performance metric on the former kind of episodes might lead to infinite values;
3. YES: for instance it is used in Policy iteration coupled with a greedy update to find the optimal policy. Another possible solution, when the policy space is limited, is to use them to solve the MDP by a brute force approach;
4. FALSE: for instance if in a state two different actions a_1 and a_2 provide the same reward, has the same transition probabilities, and one of them a_1 is included in the optimal policy, then substituting a_1 with a_2 in the optimal policy would provide an optimal policy too.

Exercise 7 (2 marks)

Tell which technique or approach would you use for the following purposes. Motivate your choice.

1. Reduce the variance of a model;
2. Select a model without retraining it on a different set of data;
3. Reduce the bias of the model, without increasing its variance;
4. Select a model by exploiting the huge computational power available to you.

1. We can increase the number of points used in the training set. This would, in principle, reduce the model variance. Otherwise, we could use Bagging;
2. We could use an adjustment technique to correct the value of the loss function provided by the already trained models penalizing their complexity;
3. In this case Boosting is a viable option, since it uses a set of simple models, whose variance is limited, to generate a more complex overall model;
4. in this case Leave One Out is a viable option, which scales with the number of samples, and, therefore requires a lot of computational power to be applied. None the less, the estimates provided by this method of the method error is the most accurate one among the methods we covered.

Exercise 8 (2 marks)

Consider, one at a time, the following characteristics for an ML problem:

1. Non-linear relationship between input and output;
2. Prior information on the input distribution;
3. Availability of labeled samples;
4. Real-time prediction capabilities.

Provide motivations for the use of either a **parametric** or **non-parametric** method.

1. PARAMETRIC/NON-PARAMETRIC: to model a nonlinear relationship one might rely on non-linear kernels or K-NN, which provides non-linear separating boundaries, but also some parametric models are able to provide this characteristic, e.g., Naive Bayes or Linear Regression with the use of basis functions;
2. PARAMETRIC: it is easier to include a priori information on parametric methods, in the specific if they are Bayesian ones we could use prior distribution to do that;
3. PARAMETRIC/NON-PARAMETRIC: the fact that we have a supervised problem does not influence the choice between these two classes of methods;
4. PARAMETRIC: since usually they take long times for training and are usually fast when it is required to provide a prediction on newly seen data.

Exercise 9 (3 marks)

Consider a hard margin SVM for classification. After the training phase, you get the following support vectors $\mathbf{x}_1 = [4\ 1]^\top$, $\mathbf{x}_2 = [6\ 3]^\top$, and $\mathbf{x}_3 = [3\ 4]^\top$, with the corresponding labels $t_1 = t_2 = -1$, and $t_3 = 1$, and weights $\alpha_1 = 1/8$, $\alpha_2 = 1/8$, and $\alpha_3 = 1/4$, and a bias $b = -1/2$. Answer the following questions and motivate:

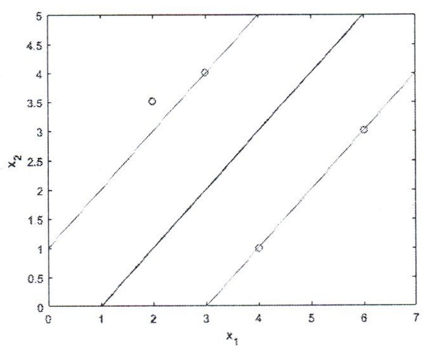
1. Plot and provide the analytic formula of the boundary and the margins determined by the SVM in the input space (hint: start by plotting the points in the training set);

2. Do you need to introduce soft margins in the current situation, or a hard margin SVM is sufficient?

3. If we had as additional point $x_4 = [2\ 7/2]^\top$, $t_4 = 1$ in the training set, would you consider retraining the SVM?

4. Do you think that this problem would benefit from the use of a non-linear kernel in the SVM formulation?

The plot is the following:



1. The parameter vector is $\mathbf{w} = \sum_{i=1}^{3} t_i \alpha_i \mathbf{x}_i = [-\frac{1}{2}\ \frac{1}{2}]^\top$, the boundary is, therefore $x_2 = x_1 - 1$ and the margins $x_2 = x_1 - 1 \pm 2$;

2. Since the training set is linearly separable, we do not need to use soft margins to find a feasible solution for the SVM;

3. Since the new point is outside the margins it would not be a support vector, and, therefore, even with a retraining i would have the same solution for the SVM;

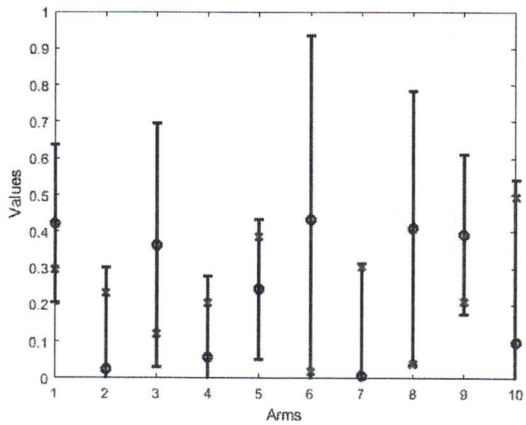
4. Again, since the dataset is already linearly separable, it does not seem a good idea to introduce non-linear kernels in the SVM.

Exercise 10 (3 marks)

We used the UCBV algorithm on a stochastic MAB problem, where the rule to pull an arm a_{i_t} at round t is according to the following upper bound:

$$i_t := \arg\max_i \left(\frac{R_t(a_i)}{N_t(a_i)} + \sqrt{\frac{V_t(a_i)\log(t)}{N_t(a_i)}} + \frac{3\log(t)}{N_t(a_i)} \right),$$

where $N_t(a_i)$ denotes the number of times the arm a_i has been pulled up to round t, $R_t(a_i)$ the cumulated reward of arm a_i up to round t, and $V_t(a_i)$ denotes the estimates of the variance of the arm a_i at round t. Assume that after several rounds the situation is the following:



where the crosses are the real expected reward of the arms, the circles are the estimated ones and the lines denotes the bounds provided by UCBV.

1. What arm are we going to pull in the next round? Is it the optimal one? Which arm we are going to pull most of the time as $T \to +\infty$?
2. Do you think that this algorithm is using more information about the reward distributions than UCB1 or not?
3. How could you tell if the UCBV algorithm is a better choice than UCB1 to solve a MAB stochastic problem?

1. We are going to pull the arm with the largest upper bound, i.e., a_6, which is not the optimal one, since it has a real expected reward (red cross) which is the lowest one. The optimal arm is a_{10}, the one with the largest expected reward and it is the one we are going to pull the most as $T \to +\infty$;
2. Since it is using also the estimates of the variance of the reward $V_t(a_i)$ it seems that this algorithm is exploiting more information about the reward process;
3. We can compare them theoretically or empirically. Theoretically we can comparte this two algorithm by checking their upper bounds, where the tightest is the best one. Empirically we can run the two on different provblems and evaluate which one is providing the lowest cumulate regret on a given time horizon T.

Student's name: