Homework #1

RELEASE DATE: 03/05/2019

DUE DATE: 03/26/2019, BEFORE 14:00 ON GRADESCOPE

QUESTIONS ABOUT HOMEWORK MATERIALS ARE WELCOMED ON THE FACEBOOK FORUM.

Please upload your solutions (without the source code) to Gradescope as instructed.

For problems marked with (*), please follow the guidelines on the course website and upload your source code to CEIBA. You are encouraged to (but not required to) include a README to help the TAs check your source code. Any programming language/platform is allowed.

Any form of cheating, lying, or plagiarism will not be tolerated. Students can get zero scores and/or fail the class and/or be kicked out of school and/or receive other punishments for those kinds of misconducts.

Discussions on course materials and homework solutions are encouraged. But you should write the final solutions alone and understand them fully. Books, notes, and Internet resources can be consulted, but not copied from.

Since everyone needs to write the final solutions alone, there is absolutely no need to lend your homework solutions and/or source codes to your classmates at any time. In order to maximize the level of fairness in this class, lending and borrowing homework solutions are both regarded as dishonest behaviors and will be punished according to the honesty policy.

You should write your solutions in English or Chinese with the common math notations introduced in class or in the problems. We do not accept solutions written in any other languages.

This homework set comes with 160 points and 20 bonus points. In general, every homework set would come with a full credit of 160 points, with some possible bonus points.

Transforms: Explicit versus Implicit

Consider the following training data set:

$$\mathbf{x}_1 = (1,0), y_1 = -1 \qquad \mathbf{x}_2 = (0,1), y_2 = -1 \qquad \mathbf{x}_3 = (0,-1), y_3 = -1$$

$$\mathbf{x}_4 = (-1,0), y_4 = +1 \qquad \mathbf{x}_5 = (0,2), y_5 = +1 \qquad \mathbf{x}_6 = (0,-2), y_6 = +1$$

$$\mathbf{x}_7 = (-2,0), y_7 = +1$$

1. Use following nonlinear transformation of the input vector $\mathbf{x} = (x_1, x_2)$ to the transformed vector $\mathbf{z} = (\phi_1(\mathbf{x}), \phi_2(\mathbf{x}))$:

$$\phi_1(\mathbf{x}) = 2x_2^2 - 4x_1 + 2 \qquad \phi_2(\mathbf{x}) = x_1^2 - 2x_2 - 3$$

What is the equation of the optimal separating "hyperplane" in the $\mathcal{Z}$ space? Explain your answer, mathematically or pictorially.

2. Consider the same training data set as Question 1, but instead of explicitly transforming the input space $\mathcal{X}$ to $\mathcal{Z}$, apply the hard-margin support vector machine algorithm with the kernel function

$$K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x}^T\mathbf{x}')^2,$$

which corresponds to a second-order polynomial transformation. Set up the optimization problem using $(\alpha_1, \cdots, \alpha_7)$ and numerically solve for them (you can use any package you want). What is the optimal $\boldsymbol{\alpha}$? Based on those $\boldsymbol{\alpha}$, which are the support vectors?

3. Following Question 2, what is the corresponding nonlinear curve in the $\mathcal{X}$ space? Please provide calculation steps of your choice.

4. Compare the two nonlinear curves found in Questions 1 and 3, should they be the same? Why or why not?

Kernels and Transforms

5. Recall that in the derivation of (one-dimensional) Gaussian kernel, we derived its associated transform

$$\boldsymbol{\phi}(x) = \exp(-x^2) \cdot \left(1, \sqrt{\tfrac{2}{1!}}x, \sqrt{\tfrac{2^2}{2!}}x^2, \ldots\right).$$

Let

$$\tilde{\boldsymbol{\phi}}(x) = \left(1, \sqrt{\tfrac{2}{1!}}x, \sqrt{\tfrac{2^2}{2!}}x^2, \ldots\right).$$

Prove that $\exp(-x^2) = \frac{1}{\|\tilde{\boldsymbol{\phi}}(x)\|}$. In other words, $\boldsymbol{\phi}(x)$ can be viewed as a normalized version of $\tilde{\boldsymbol{\phi}}(x)$.

6. Let $\cos(\mathbf{x}, \mathbf{x}')$ measure the cosine of the angle between two non-zero vectors $\mathbf{x}$ and $\mathbf{x}'$ in $\mathbb{R}^d$. The function cos is typically called the cosine similarity between two vectors. Prove or disprove that $\cos(\mathbf{x}, \mathbf{x}')$ is a valid kernel. (*Hint: to prove, you'd better construct its associated transform; to disprove, you may use Mercer's condition to construct a counter-example on positive semi-definiteness.*)

Radius of Transformed Vectors via the Kernel

Recall that for support vector machines, d_{VC} is upper bounded by $\frac{R^2}{\rho^2}$, where ρ is the margin and R is the radius of the minimum hypersphere that $\mathcal{Z}$ resides in. In general, R should come from our knowledge on the learning problem, but we can *estimate* it by looking at the minimum hypersphere that the transformed training examples resides in. In particular, we want to seek for the optimal R that solves

$$(P) \quad \min_{R \in \mathbb{R}, \mathbf{c} \in \mathbb{R}^d} \quad R^2 \quad \text{subject to } \|\mathbf{z}_n - \mathbf{c}\|^2 \le R^2 \text{ for } n = 1, 2, \cdots, N.$$

7. Let λ_n be the Lagrange multipliers for the n-th constraint above. Following the derivation of the dual support vector machine in class, write down (P) as an equivalent optimization problem

$$\min_{R \in \mathbb{R}, \mathbf{c} \in \mathbb{R}^d} \quad \max_{\lambda_n \ge 0} \quad L(R, \mathbf{c}, \boldsymbol{\lambda}).$$

What is $L(R, \mathbf{c}, \boldsymbol{\lambda})$?

8. Using (assuming) strong duality, the solution to (P) in Question 7 would be the same as the Lagrange dual problem

$$(D) \quad \max_{\lambda_n \ge 0} \quad \min_{R \in \mathbb{R}, \mathbf{c} \in \mathbb{R}^d} \quad L(R, \mathbf{c}, \boldsymbol{\lambda}).$$

List all KKT conditions of (P) and (D). Then, prove or disprove the following for the optimal solution of (P) and (D):

$$\text{if } \textstyle\sum_{n=1}^N \lambda_n \ne 0, \text{ then } \mathbf{c} = \left(\sum_{n=1}^N \lambda_n \mathbf{z}_n\right) \Big/ \left(\sum_{n=1}^N \lambda_n\right)$$

9. Continue from Question 8 and assume that all the $\mathbf{z}_n$ are different, which implies that the optimal $R > 0$. Using the KKT conditions to simplify the Lagrange dual problem, and obtain a dual problem that involves only λ_n. One form of the dual problem should look like

$$(D') \quad \max_{\lambda_n \ge 0} \quad \text{Objective}(\boldsymbol{\lambda}) \quad \text{subject to } \sum_{n=1}^N \lambda_n = \text{constant}$$

Derive the dual problem step by step.

10. Continue from Question 9 and solve the (D') that involves the kernel $K(\mathbf{x}, \mathbf{x}') = \mathbf{z}^T \mathbf{z}'$. How can the optimal R be calculated using the kernel trick based on some i with $\lambda_i > 0$? Please provide the derivation steps.

Hard-Margin versus Soft-Margin

11. Assume that the data set is separable in the $\mathcal{Z}$ space. That is, hard-margin SVM has an optimal solution of some vector $\boldsymbol{\alpha}^*$. Prove that if $C \ge \max_{1 \le n \le N} \alpha_n^*$, the vector $\boldsymbol{\alpha}^*$ is also an optimal solution to the soft-margin SVM.

Operation of Kernels

12. For a given valid kernel K, consider a new kernel $\tilde{K}(\mathbf{x}, \mathbf{x}') = pK(\mathbf{x}, \mathbf{x}')$ for some $p > 0$. Prove or disprove that for the dual of soft-margin support vector machine, using $\tilde{K}$ along with a new $\tilde{C} = \frac{C}{p}$ instead of K with the original C leads to an equivalent g_{SVM} classifier.

Experiments with Soft-Margin Support Vector Machine

For Questions 13 to 16, we are going to experiment with a real-world data set. Download the processed US Postal Service Zip Code data set with extracted features of intensity and symmetry for training and testing:

`http://www.amlbook.com/data/zip/features.train`

`http://www.amlbook.com/data/zip/features.test`

The format of each row is

`digit intensity symmetry`

We will consider binary classification problems of the form "one of the digits" (as the positive class) versus "other digits" (as the negative class).

The training set contains thousands of examples, and some quadratic programming packages cannot handle this size. We recommend that you consider the LIBSVM package

`http://www.csie.ntu.edu.tw/~cjlin/libsvm/`

Regardless of the package that you choose to use, please read the manual of the package carefully to make sure that you are indeed solving the soft-margin support vector machine taught in class like the dual formulation below:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n\alpha_m y_n y_m K(\mathbf{x}_n, \mathbf{x}_m) - \sum_{n=1}^{N}\alpha_n \\ \text{s.t.} \quad & \sum_{n=1}^{N} y_n \alpha_n = 0 \\ & 0 \le \alpha_n \le C \quad n = 1, \cdots, N \end{aligned}$$

In the following questions, please use the 0/1 error for evaluating E_{in}, E_{val} and E_{out} (through the test set). Some practical remarks include

(i) Please tell your chosen package to **not** automatically scale the data for you, lest you should change the effective kernel and get different results.

(ii) It is your responsibility to check whether your chosen package solves the designated formulation with enough numerical precision. Please read the manual of your chosen package for software parameters whose values affect the outcome—any ML practitioner needs to deal with this kind of added uncertainty.

13. (*) Consider the linear soft-margin SVM. That is, either solve the primal formulation of soft-margin SVM with the given $\mathbf{x}_n$, or take the linear kernel $K(\mathbf{x}_n, \mathbf{x}_m) = \mathbf{x}_n^T \mathbf{x}_m$ in the dual formulation. For the binary classification problem of "2" versus "not 2", plot $\|\mathbf{w}\|$ versus $\log_{10} C \in \{-5, -3, -1, 1, 3\}$. Describe your findings.

14. (*) Consider the polynomial kernel $K(\mathbf{x}_n, \mathbf{x}_m) = (1 + \mathbf{x}_n^{\mathrm{T}}\mathbf{x}_m)^Q$, where Q is the degree of the polynomial. With $Q = 2$, and the binary classification problem of "4" versus "not 4", plot E_{in} versus $\log_{10} C \in \{-5, -3, -1, 1, 3\}$. Describe your findings.

15. (*) Consider the Gaussian kernel $K(\mathbf{x}_n, \mathbf{x}_m) = \exp\left(-\gamma||\mathbf{x}_n - \mathbf{x}_m||^2\right)$. With $\gamma = 80$, and the binary classification problem of "0" versus "not 0". Consider values of $\log_{10} C$ within $\{-2, -1, 0, 1, 2\}$. Plot the the distance of any free support vector to the hyperplane in the (infinite-dimensional) $\mathcal{Z}$ space versus $\log_{10} C$. Describe your findings.

16. (*) Following Question 15 and consider a validation procedure that randomly samples 1000 examples from the training set for validation and leaves the other examples for training g_{SVM}^-. Fix $C = 0.1$ and use the validation procedure to choose the best $\log_{10}\gamma \in \{-2, -1, 0, 1, 2\}$ according to E_{val}. If there is a tie of E_{val}, choose the smallest γ. Repeat the procedure 100 times. Plot a histogram for the number of times each $\log_{10}\gamma$ is selected.

Bonus: Constant Feature for Support Vector Machine

17. (Bonus, 10 points) In the derivation of the Gaussian kernel, we see that the first feature component in $\Phi(\mathbf{x})$ is actually a constant 1; in the derivation of the polynomial kernel, we also see that the first feature component is a constant. Prove or disprove that after solving the soft-margin SVM, if we calculate the optimal weight value w_i that corresponds to the constant feature component z_i, we would get $w_i = 0$.

18. (Bonus, 10 points) For a given valid kernel K, consider a new *function* $\tilde{K}(\mathbf{x}, \mathbf{x}') = K(\mathbf{x}, \mathbf{x}') + q$ for any real value q (which can be negative). Prove or disprove that for the dual of soft-margin support vector machine, using $\tilde{K}$ instead of K leads to an equivalent g_{SVM} classifier.