SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

01.112 Machine Learning, Fall 2019
Homework 3

Due 2 Nov 2019, 11:59 pm

This homework will be graded by Sun Xiaobing

Question 1 [20 points] Download and install the widely used SVM implementation LIBSVM (`https://github.com/cjlin1/libsvm`, or `https://www.csie.ntu.edu.tw/~cjlin/libsvm/`; clicking on either link takes you to the webpage). We expect you to install the package on your own – this is part of learning how to use off-the-shelf machine learning software. Read the documentation to understand how to use it.

Download `promoters` folder. In that folder are `training.txt` and `test.txt`, which respectively contain 74 training examples and 32 test examples in LIBSVM format. The goal is to predict whether a certain DNA sequence is a promoter[1] or not based on 57 attributes about the sequence (this is a binary classification task).

Run LIBSVM to classify promoters with different kernels (0-3), using default values for all other parameters. What is your test accuracy for each kernel choice?

$$\text{Kernel 0 (linear) accuracy} = 27/32 = 84\%$$
$$\text{Kernel 1 (polynomial) accuracy} = 26/32 = 81\%$$
$$\text{Kernel 2 (RBF) accuracy} = 29/32 = 91\%$$
$$\text{Kernel 3 (Sigmoid) accuracy} = 14/32 = 44\%$$

Question 2 [30 points] Suppose we are looking for a maximum margin linear classifier through the origin, i.e., the bias $b = 0$. This means that we have to minimize

$$\frac{1}{2}\|\mathbf{w}\|^2 \text{ subject to } y^{(t)}\mathbf{w}\cdot\mathbf{x}^{(t)} \geq 1, t = 1, ..., n.$$

(a) [15 points] Suppose there are two training examples $\mathbf{x}^{(1)} = (1, 1)^T$ and $\mathbf{x}^{(2)} = (1, 0)^T$ with labels $y^{(1)} = 1$ and $y^{(2)} = -1$. What is the $\mathbf{w}$ in this case, and what is the margin γ?

In this case, the SVM uses both data points as support vectors, such that $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ satisfy the equations of the margin, as given below.

$$\mathbf{w}\cdot\mathbf{x}^{(t)} = 1 \rightarrow w_1x_{1,1} + w_2x_{1,2} = 1 \rightarrow w_1 + w_2 = 1$$
$$\mathbf{w}\cdot\mathbf{x}^{(t)} = -1 \rightarrow w_1x_{2,1} + w_2x_{2,2} = -1 \rightarrow w_1 = -1$$

[1] A promoter is a region of DNA that facilitates the transcription of a particular gene. The ability to predict promoters is of practical importance in searching for new promoter sequences.

Solving the two equations, we get, $\mathbf{w}^* = (-1, 2)^T$

(b) [15 points] How will the parameters $\mathbf{w}$ and the margin γ change in the previous question if the bias/offset parameter b is allowed to be non-zero?

Similar to previous solution, the SVM uses both data points as support vectors, such that $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ satisfy the equations of the margin, as given below.

$$\mathbf{w} \cdot \mathbf{x}^{(t)} + w_0 = 1 \rightarrow w_1 x_{1,1} + w_2 x_{1,2} + w_0 = 1 \rightarrow w_1 + w_2 + w_0 = 1 \quad (1)$$

$$\mathbf{w} \cdot \mathbf{x}^{(t)} + w_0 = -1 \rightarrow w_1 x_{2,1} + w_2 x_{2,2} + w_0 = -1 \rightarrow w_1 + w_0 = -1 \quad (2)$$

Since, there are three unknowns, we would need one equation. We know that the decision boundary is mid-way between the margins. In this case, $\mathbf{x}^{(3)} = (1, 0.5)^T$, lies on the decision boundary. We substitute $\mathbf{x}^{(3)}$ in the equation of the decision boundary as given below.

$$w_1 x_{1,1} + w_2 x_{1,2} + w_0 = 0 \rightarrow w_1 + 0.5 w_2 + w_0 = 0 \quad (3)$$

Using Equations 1, 2, 3, we get $\mathbf{w}^* = (0, 2)^T$ and $w_0 = -1$.

Question 3 [20 points] In this problem, we consider constructing new kernels by combining existing kernels. Recall that for some function $K(\mathbf{x}, \mathbf{z})$ to be a kernel, we need to be able to write it as an inner product of vectors from some high-dimensional feature space:

$$K(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^T \phi(\mathbf{z})$$

Mercer's theorem gives a necessary and sufficient condition for a function K to be a kernel: its corresponding kernel matrix has to be symmetric and positive semidefinite, where the elements of a kernel matrix are inner products between all pairs of examples.
Suppose that $K_1(\mathbf{x}, \mathbf{z})$ and $K_2(\mathbf{x}, \mathbf{z})$ are kernels over $\mathcal{R}^n$x$\mathcal{R}^n$. For each of the cases below, state whether K is also a kernel. If it is, prove it. If it is not, give a counter example. (*Hints: You can use either Mercer's theorem or the definition of a kernel, as needed.*).

1. $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z}) K_2(\mathbf{x}, \mathbf{z})$
   $K(\mathbf{x}, \mathbf{z})$ is a kernel, if $K_1(\mathbf{x}, \mathbf{z})$ and $K_2(\mathbf{x}, \mathbf{z})$ are also kernels.
   Let, $K_1(\mathbf{x}, \mathbf{z}) = \phi^{(1)}(\mathbf{x})^T \phi^{(1)}(\mathbf{z})$ and $K_2(\mathbf{x}, \mathbf{z}) = \phi^{(2)}(\mathbf{x})^T \phi^{(2)}(\mathbf{z})$
   Let us construct $\phi(\mathbf{x}) = [\phi^{(1)}(\mathbf{x}) \phi^{(2)}(\mathbf{x})]$
   Given that
   $$\begin{aligned} K(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^T \phi(\mathbf{z}) &= [\phi^{(1)}(\mathbf{x}) \phi^{(2)}(\mathbf{x})]^T [\phi^{(1)}(\mathbf{z}) \phi^{(2)}(\mathbf{z})] \\ &= \textstyle\sum_{i,j} \phi_i^{(1)}(x) \phi_j^{(2)}(x) \sum_{i,j} \phi_i^{(1)}(z) \phi_j^{(2)}(z) \\ &= \textstyle\sum_{i,j} \phi_i^{(1)}(x) \phi_j^{(2)}(x) \phi_i^{(1)}(z) \phi_j^{(2)}(z) \\ &= \textstyle\sum_{i,j} \phi_i^{(1)}(x) \phi_i^{(1)}(z) \phi_j^{(2)}(x) \phi_j^{(2)}(z) \\ &= \textstyle\sum_i \phi_i^{(1)}(x) \phi_i^{(1)}(z) \sum_j \phi_j^{(2)}(x) \phi_j^{(2)}(z) \\ &= (\phi^{(1)}(\mathbf{x})^T \phi^{(1)}(\mathbf{z}))(\phi^{(2)}(\mathbf{x})^T \phi^{(2)}(\mathbf{z})) \\ &= K_1(\mathbf{x}, \mathbf{z}) K_2(\mathbf{x}, \mathbf{z}) \end{aligned}$$

   Hence, $K(\mathbf{x}, \mathbf{z})$ is a kernel.

2. $K(\mathbf{x}, \mathbf{z}) = aK_1(\mathbf{x}, \mathbf{z}) + bK_2(\mathbf{x}, \mathbf{z})$, where $a, b > 0$ are real numbers
   $K(\mathbf{x}, \mathbf{z})$ is a kernel, if $K_1(\mathbf{x}, \mathbf{z})$ and $K_2(\mathbf{x}, \mathbf{z})$ are also kernels.
   Let, $K_1(\mathbf{x}, \mathbf{z}) = \phi^{(1)}(\mathbf{x})^T\phi^{(1)}(\mathbf{z})$ and $K_2(\mathbf{x}, \mathbf{z}) = \phi^{(2)}(\mathbf{x})^T\phi^{(2)}(\mathbf{z})$
   Let us construct $\phi(\mathbf{x}) = [\sqrt{a}\phi^{(1)}(\mathbf{x})\sqrt{b}\phi^{(2)}(\mathbf{x})]$
   Given that
   $$\begin{aligned} K(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^T\phi(\mathbf{z}) &= [\sqrt{a}\phi^{(1)}(\mathbf{x})\sqrt{b}\phi^{(2)}(\mathbf{x})]^T[\sqrt{a}\phi^{(1)}(\mathbf{z})\sqrt{b}\phi^{(2)}(\mathbf{z})] \\ &= [a\phi^{(1)}(\mathbf{x})^T\phi^{(1)}(\mathbf{z})] + [b\phi^{(2)}(\mathbf{x})^T\phi^{(2)}(\mathbf{z})] \\ &= aK_1(\mathbf{x}, \mathbf{z}) + bK_2(\mathbf{x}, \mathbf{z}) \end{aligned}$$
   Hence, $K(\mathbf{x}, \mathbf{z})$ is a kernel.

3. $K(\mathbf{x}, \mathbf{z}) = aK_1(\mathbf{x}, \mathbf{z}) - bK_2(\mathbf{x}, \mathbf{z})$, where $a, b > 0$ are real numbers
   The above expression does not satisfy positive semi-definite property, required by Mercer's theorem. A counter example is given below. For example, consider, $a = 1$ and $b = 1$,
   $$K_1(\mathbf{x}, \mathbf{z}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, K_2(\mathbf{x}, \mathbf{z}) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$
   Substituting the above values in the required kernel equation, $K(\mathbf{x}, \mathbf{z}) = aK_1(\mathbf{x}, \mathbf{z}) - bK_2(\mathbf{x}, \mathbf{z})$
   $$K(\mathbf{x}, \mathbf{z}) = a\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - b\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = -\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} < 0$$
   Hence, $K(\mathbf{x}, \mathbf{z})$ is not a kernel.

4. $K(\mathbf{x}, \mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$, where $f : \mathcal{R}^n \to \mathcal{R}$ be any real valued function of x.
   From kernel property 2, we know that, $K(\mathbf{x}, \mathbf{z}) = f(\mathbf{x})K(\mathbf{x}, \mathbf{z})f(\mathbf{z})$

   From kernel property 1, we define, $K(\mathbf{x}, \mathbf{z}) = \mathbb{I}$, as the identity matrix.

   Therefore, $K(\mathbf{x}, \mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$

   Hence, $K(\mathbf{x}, \mathbf{z})$ is a kernel.

Question 4 [30 points]

(a) [10 points] In logistic regression, we find parameters of a logistic (sigmoid) function that maximize the likelihood of a set of training examples $((x^{(1)}, y^{(1)}), ..., (x^{(n)}, y^{(n)}))$. The likelihood is given by
$$\prod_{i=1}^{n} P(y^{(i)} \mid x^{(i)}) \tag{4}$$
However, we re-express the problem of maximizing the likelihood as minimizing the following expression:
$$\frac{1}{n}\sum_{i=1}^{n} \log\big(1 + \exp\big(-y^{(i)}(\theta \cdot x^{(i)} + \theta_0)\big)\big). \tag{5}$$

(Note that both maximization and minimization problems have the same optimal θ and θ_0.) What *computational* advantage does Equation 2 have over Equation 1? *(Hint: try randomly generating, say, 1,000 probabilities in Python and multiplying them together as in Eq. 1.)*

Progressively multiplying many probabilities together as in Equation 1 quickly gives a result that is too small to be representable in computer memory (this is known as an *underflow* problem). In contrast, Equation 2 uses a sum over terms that makes this problem less likely to occur.

(b) [20 points] You are given a training set `diabetes_train.csv`. Each row in the file contains whether a patient has diabetes (+1: yes, -1: no), followed by values of 20 unknown features. **Write code to train a logistic regression model with stochastic gradient descent (SGD)**. Run SGD for 10,000 iterations, and save the model weights after every 100 iterations. Plot the log-likelihood of the training data given by your model at every 100 iterations. (Log-likelihood is $\log \prod_{i=1}^{n} P(y^{(i)}|x^{(i)}) = \sum_{i=1}^{n} \log P(y^{(i)}|x^{(i)})$ where $(x^{(i)}, y^{(i)})$ is an example.) Provide crystal clear instructions along with the source code on how to execute it. (*Hints: If your stochastic gradient descent code in the previous homework is written modularly enough, you could save time by reusing it here. Try a learning rate of 0.1*).