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?

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?

(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?

[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.

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 $\mathcal{R}^n \mathrm{x} \mathcal{R}^n$. 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. $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z}) K_2(\mathbf{x}, \mathbf{z})$
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
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
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.

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

$$\prod_{i=1}^{n} P(y^{(i)} \mid x^{(i)}) \tag{1}$$

However, we re-express the problem of maximizing the likelihood as minimizing the following expression:

$$\frac{1}{n} \sum_{i=1}^{n} \log\big(1 + \exp\big(-y^{(i)}(\theta \cdot x^{(i)} + \theta_0)\big)\big). \tag{2}$$

(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.)*

(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*).