CS5402 Assignment 5 (Binary Classification by Support Vector Machine)

Due Nov 30 2016 5:00PM

<u>This is an optional assignment. If one of your previous assignments is with a low score and you do a better job on this assignment, you can replace the lowest one by this one.</u>

One training dataset $(\boldsymbol{X}, Y)$ is provided in 'LinearSeparable.mat', where matrix $\boldsymbol{X}$ (size: D*N) contains N training samples of dimension D (In this assignment, D=2 so we can easily visualize the results). If you want to access the n^{th} sample, you can use $\boldsymbol{X}(:, n)$ in Matlab. $\boldsymbol{Y}$ (size: N*1) is the binary label vector, i.e., $\boldsymbol{Y} = [y_1, y_2, \dots, y_N]^T$. If you want to access the label of the n^{th} sample, you can use $\boldsymbol{Y}(n)$ in Matlab. We are going to train Support Vector Machine (SVM) to classify the data into two classes.

1. Primal Linear Hard-Margin Support Vector Machine

(1) Apply the primal linear hard-margin SVM algorithm onto the training dataset to obtain the $(\boldsymbol{w}_{primal}, b_{primal})$.
Note: use the quadprog() function in Matlab to solve the Quadratic Programming optimization problem.

(2) Identify which training samples in the dataset are support vectors and obtain $\boldsymbol{svIdx}_{primal}$. For example, if sample 1, 7 and 11 in the N training samples are support vectors, then $\boldsymbol{svIdx}_{primal} = [1; 7; 11]$.

(3) Compute the largest margin you achieved from your primal SVM.

A Matlab function visualizeSVM(X,Y,w,b,svIdx) is provided by the instructor to visualize your results.

2. Dual Linear Hard-Margin Support Vector Machine

(1) Apply the dual linear hard-margin SVM algorithm onto the training dataset to obtain the $\boldsymbol{\alpha}$ (size: N*1).
Note: use the quadprog() function in Matlab to solve the Quadratic Programming optimization problem.

(2) Identify which training samples in the dataset are support vectors and obtain $\boldsymbol{svIdx}_{dual}$.

(3) Use the support vectors to compute $(\boldsymbol{w}_{dual}, b_{dual})$.

(4) Compute the largest margin you achieved from your dual SVM.

A Matlab function visualizeSVM(X,Y,w,b,svIdx) is provided by the instructor to visualize your results.

3. Bonus to early birds (up to 10%): if you are the n^{th} submission $(1 \leq n \leq 10)$, you earn (11-n)% bonus points for this homework.

Submit your codes and a written report to Canvas in which you briefly discuss at least the following things:

- Summarize what you think the homework was about (what was the task; what were you trying to achieve).
- Describe the algorithms you used to solve the homework problems. (The mathematical optimization problem; the input arguments to quadprog(),etc.)
- Run your program and show quantitative (w,b,margin, support vector sample index) and qualitative (pictures by visualizeSVM()) results. Explain any practical design decisions you may have during the experiments.
- Summarize your observations from the experiments.