EE488B Special Topics in EE <Deep Learning and AlphaGo> September 20, 2017
EE, KAIST, Fall 2017 Due: 1:00pm, September 27

Problem Set #1

Warning: Homeworks will not be graded if submitted after the deadline. For all problems, show detailed reasoning.

0. (Reading assignment) DL Book Chapters 1 $\sim$ 6

About choices: For some problems, you can choose to solve only one of two, where one problem requires analysis and the other requires programming. If you submit answers for both, then we will only grade one of them. We will normalize your scores so that the means and variances of the two problems are the same. There may be issues due to imbalance in group sizes. For example, if there is only one student solving problem 1 and all the other students solve problem 2, then the normalized score of the student solving problem 1 will be the same as the average score of the students solving problem 2, which may not be fair for the student who chose to solve a potentially more challenging problem. Furthermore, if there are only two students solving problem 1 and all the other students solve problem 2, then the variance of scores of the smaller group is less reliable, which may be problematic when normalizing scores. Therefore, we will adjust scores so that the minority group is not penalized due to small group size.

Between problems 1 and 2, you only need to solve one of them.

1. (Constrained optimization – 10 points) Consider a constrained minimization problem given as

$$\min_{(x,y):\, y+2x\geq 30} f(x,y)$$

where $f(x,y) = 5(x+y)^2 + (x-y)^2$.

(a) Find a local minimum using the KKT condition.

(b) Assume the constraint is changed to $y + 2x \leq 30$ and find a local minimum using the KKT condition.

2. (Barrier and penalty methods – 10 points) Consider a constrained minimization problem given as

$$\min_{(x,y):\, y+2x\geq 30} f(x,y)$$

where $f(x,y) = 5(x+y)^2 + (x-y)^2$. Implement gradient descent with barrier and penalty methods and compare the results. You may use any programming language. Explain how your codes work. Submit your codes and plots.

Between problems 3 and 4, you only need to solve one of them.

3. (Worst-case error – 10 points) Consider a problem of estimating θ from a set of samples $\{x^{(1)}, \ldots, x^{(m)}\}$ generated *i.i.d.* according to Bernoulli(θ), where $0 \leq \theta \leq 1$. Assume $m = 1$. Can you think of an estimator that minimizes the worst-case absolute error over all θ? Namely, find an estimator $\bar{\theta}_1$ based on $x^{(1)}$ that minimizes the following.

$$\max_{0\leq\theta\leq 1} \mathbb{E}[|\bar{\theta}_1 - \theta|]$$

What is the minimum worse-case absolute error $\mathbb{E}[|\bar{\theta}_1 - \theta|]$ that the estimator achieves?[1]

4. (Underfitting and overfitting – 10 points) Write a program that can produce figures similar to those in pages 12 and 14 of lecture notes #5. Your figures do not need to look exacly like those in the lecture notes, but try to make them look similar. Training and test data should be generated as follows.

- Generate 10 *i.i.d.* training examples (x_1, y_1), (x_2, y_2), ..., (x_{10}, y_{10}), where x_i's are uniform between -1 and 3 and $y_i = ax_i^2 + bx_i + c + z_i$, $i = 1, 2, \ldots, 10$, where z_i's are *i.i.d.* uniform noise between -0.2 and 0.2 and are independent of x_i's. Pick a, b and c appropriately so that your figure looks similar to that in page 12 of lecture notes #5.

- Generate 100 test examples such that x values are uniformly spaced between -1 and 3 and $y = ax^2 + bx + c$, where a, b and c are the same as the ones you chose for generating the train data. Note that we are not including noise in the test examples, i.e., no z_i's.

You may use any programming language. Explain how your codes work. Submit your codes and plots.

5. (Size imbalance – 1 point) Think about how to adjust scores appropriately so that the minority group is not penalized due to small group size. Read "About choices" in the beginning of this problem set. You will get one point for this problem if you submit any solution.

6. (Your own questions – 1 point) Submit one or more questions. Think of good questions. Some will be answered during class. You will get one point if you submit any question.

7. (Suggestions – not to be graded) Submit any suggestions you might have that could improve this course.

8. (Clicker questions – not to be graded) Submit a short question with two to four answers. Some will be chosen and used for attendance checking using clickers.

[1] See the solution for problem #1 of the last year's problem set #2 for a related problem with an example of bias-variance tradeoff. Last year's problem sets and their solutions are available at KLMS.