CS3244 Exam 1: Part 3

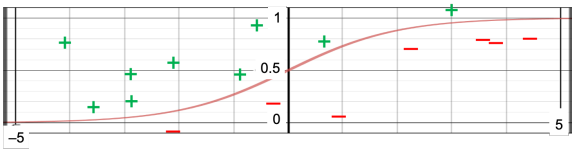
28 Sep 2020

Please do not turn to the next page until you are told to do so by your proctor.

- This exam part is worth **25** marks out of a **75** mark total for all three parts.
- This exam part is estimated to take you about **25** minutes to complete.
- This exam part has a total of **8** questions.
- This exam part has only *text response* (essay) questions and *worked response* (show your intermediate workings / calculations) questions. Be mindful of the estimated time for a response; longer answers may incur penalties and cost you extra time that you could use to answer other questions.
- You can visit `http://www.comp.nus.edu.sg/~cs3244/2010/e1.part3.html` to reach the entry form for this survey if you lose your browser window.
- Do remember that you will need to key in and submit your answers to the according assessment system as designated by your proctor or by exam central.

ANSWERS VERSION 1

[Questions 1–2] The below image purportely shows how a Logistic Regression classifer boundary looks like (A version of this image did actually appear on a fairly famous data science website).



1. (Text Response; 3 marks) Describe why this is an incorrect representation of how Logistic Regression functions.

   Explanation: There are several problems with the diagram as shown. For your understanding, we have made this model answer more comprehensive than is expected for full credit. In Logistic Regression, the red sigmoid-like curve is not a classification boundary, but rather a mapping function. Logistic Regression maps the real valued signal $\theta^T x$ to the bounded interval (0,1) as a probability of positive classification. As such, both positive and negative instances should not be depicted as being on the plane in the diagram itself.

2. (Text Response; 1 mark) Describe one aspect of this image which is representative of Logistic Regression.

   Explanation: Instances lie directly on the red curve, such that their signal (as signified by the x value) maps directly to the probability of classification (the corresponding y value on the curve). The output of logistic regression is a classification, not a confidence value, so a more accurate diagram would have all of the the instances left of x=0 mapped to negative classifications (a row of instances along y=0) and those right of x=0 mapped to positive classifications (a row of instances along y=1).

[Questions 3–5] You have been given n datasets — $D_1, D_2, ..., D_n$ — and a supervised learner L to learn h. L is to learn a single variable function $h : x \rightarrow y$, where $x, y \in \mathbb{R}$. We want to use the learner and different datasets to learn and predict using the average hypothesis: $\bar{h}(x)$.

The relevant classes are defined below using pseudocode:

```
1  class Hypothesis {
2  /* This class represents a hypothesis - h(x) - returned when a learner learns from a dataset.
3  We only show the relevant function singatures and variables.
4  */
5      int predict(int x) {
6          // Computes the prediction for input x (integer) and returns h(x) an integer
7      }
8  }
9
10 class Learner {
11 /* This class represents a learner based on some learning algorithm (eg. SGD) and hypothesis class.
12 */
13     Hypothesis learn(Dataset D) {
14         /* Takes a dataset D and returns an object of class Hypothesis.
15         You can assume the hypothesis returned is the one with least training error.
16         */
17     }
18 }
19
20 class AverageHypothesis extends Hypothesis {
21 /* This class represents the average hypothesis and will be the one you are completing.
22 */
23     hypotheses = [] //List/Array to be populated
24     void initialize(Dataset[] datasets) {
25         //This function will populate the hypotheses list
26     }
27
28     int predict(int x) {
29         //This function returns the prediction of the average hypotheses on input x
30     }
31 }
```

3. (Worked Response; 3 marks) Code the INITIALIZE function (Line 26) in the AVERAGEHYPOTHESIS class. Assume that the other classes' functions are complete. Clear pseudocode is sufficient.

```
void initialize(Dataset D1, ..., Dataset Dn) {
    Learner L = new Learner();
    for dataset in D1, ..., Dn {
        hypotheses.add(L.learn(dataset);
    }
}
```

Explanation:

- Marks are deducted for not writing code.
- Any pseudocode that closely resembles the above fetches you full marks.
- You must use the learn function.
- Common mistake - Populating list with datasets.

4. (Worked Response; 3 marks) Code the PREDICT function (Line 30) in the AVERAGEHYPOTHESIS class. Assume that the other classes' functions are complete. Clear pseudocode is sufficient.

```
int predict(int x) {
    int result = 0;
    int n = hypotheses.size();
    for hypothesis in hypotheses {
        result += hypothesis.predict(x);
    }
    return result / n;
}
```

Explanation:

- Marks are deducted for not writing code.
- Any pseudocode that closely resembles the above fetches you full marks.

- You must use the predict function and x itself.
- You must clearly state what is averaged.

5. Now assume that the learner L learns a function of the hypothesis class $y = mx + b$, and you are given 3 2-point datasets in the form of the (x, y) values below:

$$D1 : (1,3), (2,4)$$
$$D2 : (1,5), (2,1)$$
$$D3 : (1,2), (2,9)$$

(Worked Response; 4 marks) Describe the output of the call to your INITIALIZE function, and subsequently, what your PREDICT function returns for $x = 3$. Assume the LEARN function runs until the best hypothesis is obtained given the complexity of the hypothesis class.
Show relevant intermediate return values for full credit.

Explanation: H1: Line learned from 2 points in D1 –> y = x + 2
H2: y = -4x + 9
H3: y = 7x - 5
initialize populates the list with the above hypothesis.
Substituting x and averaging, predict returns (5 -3 + 16)/3 = 6

- You must show all the lines learned.
- Try to avoid calculation errors.
- Full marks are awarded for correct intermediary steps and final output.

6. (Text Response; 4 marks) The expressive power of linear models is low compared against other learners that capture both linear and non-linear relationships natively.

Why then are linear models still relevant in the real-world? Justify your response.

Explanation: Linear models are still relevant due to the following reasons. First, complex models (e.g., non-linear) may suffer from overfitting problems when the samples are insufficient. Second, it is easy to interpret and uses less computational resources. Lastly, linear models can represent non-linearity by using data transformation and kernel methods.

7. (Text Response; 3 marks) We learned about **Laplace Smoothing** and showed how it solves zero values in Naïve Bayes. Answer whether you would or would not apply Laplacian smoothing for skewed data. Justify your response.

Explanation: The answer depends. Generally, we apply Laplacian smoothing when we have insufficient data. We trust smoothing less when we have sufficient data to believe the bias is actually there. We should use smoothing but with a small k with we have skewed data because a large k would impose a strong uniform bias on the data.

8. (Text Response; 4 marks) Say that you know stochastic noise is very high in a dataset you have been assigned to conduct supervised learning on. State one condition for why you would use a high variance model, and one condition why you would not.

Justify your answer for full credit.

Explanation: Possible reasons to use a high variance model-

- When we have a large dataset, the errors seem to average out and we can learn an accurate representation
- If the training dataset is representative of the underlying target function, using a high variance model will be able to approximate the target quite accurately
- Useful for approximating high complexity target functions
- Bias-Variance tradeoff: Model with high variance will be helpful for finding a function with low bias (useful in cases stated in point 2).

Possible reasons for not using a high variance model-

- A high variance model will most likely lead to overfitting on the training dataset and not be able to generalise well.
- The model will closely capture the high stochastic noise

This marks the end of this part of the exam.
These is no additional material beyond this point.