Homework 04 – Probability

Arthur J. Redfern
arthur.redfern@utdallas.edu
Feb 11, 2019

---

0 Outline

1 Logistics

Assigned:	Mon Feb 11, 2019
Due:	Mon Feb 18, 2019
Format:	PDF uploaded to eLearning

2 Reading

1. Read: Probability
   https://github.com/arthurredfern/UT-Dallas-CS-6301-CNNs/blob/master/Lectures/xNNs_04_Probability.pdf

2. Read: Visual information theory
   http://colah.github.io/posts/2015-09-Visual-Information/

3 Theory

3. Consider the standard game of 20 questions as described in the following pseudo code:

   // standard game of 20 questions (**interleaved** questions and answers)

```
Winner = Person A
Person A chooses an object X but doesn't tell Person B what object X they chose
For q = 1 to 20
    Person B asks a yes or no question to help determine what the object is
    Person A responds truthfully with yes or no
    If the question asked by Person B is "Is the object X" then
        Winner = Person B
        Break
```

See https://en.wikipedia.org/wiki/Twenty_Questions or other web sites if you're unfamiliar with the game.

Now consider a modified version of 20 questions where

```
// modified game of 20 questions (all questions then all answers then 1 guess)
Winner = Person A
Person A chooses an object X but doesn't tell Person B what object X they chose
For q = 1 to 19
    Person B asks a yes or no question to help determine what the object is
For q = 1 to 19
    Person A responds truthfully with yes or no to each question q from Person B
Person B asks "Is the object Y?"
If X == Y
    Winner = Person B
```

3.A. You're Person B and want to win. Which version of the game do you play, the standard or the modified? Why?

3.B. Consider a classification network where multiple CNN or RNN layers transform a data tensor to a feature vector and a dense layer transforms the feature vector to a class pmf vector:

Data tensor → [multiple layers] → feature vector → [dense layer] → class pmf vector

Is this more similar to the standard or modified version of 20 questions? Why?

3.C. Change the standard game such that with some probability Person A will answer each question incorrectly. How does this change the strategy of Person B? After answering this question, look at the chain rule of probability with the correct given values replaced by estimates that may or may not be correct (i.e., instead of the typical $p(x_{n-1} \mid x_{n-2}, \ldots, x_0)$ consider $p(x_{n-1} \mid x_{n-2}^{hat}, \ldots, x_0^{hat})$ where hat indicates estimate).

3.D. Now change the modified game such that with some probability Person A will answer each question incorrectly. How does this change the strategy of Person B? After answering this question, think about the number of features vs the number of classes that are being estimated.

4. A multiple choice test can be viewed as a test with an implicit curve that is noisy and grade dependent. Consider a 100 question ABCD multiple choice test and assume that for each question on the test a person either knows the answer or has no idea and makes a totally random guess. Define the implicit curve as the number of questions answered correctly via knowledge or guess minus the number of questions answered correctly via knowledge. For example, a person knows the answer to 73 questions, guesses on 27 questions and gets a total of 79 questions correct; the implicit curve for this case would be 79 – 73 = 6. Plot the mean and standard deviation of the implicit curve vs the number of questions that a person answered correctly via knowledge. Use Python to simulate many trials for each point and Matplotlib to plot.

5. Assume ImageNet has 1.28 million images of size 3 x 256 x 256 with 1280 images each in 1000 different classes. How many bits of information are in the ImageNet labels?

4 Practice

6. [**Optional**] Create a count based N-gram model for 26 character + 1 space prediction with N = 4: $p(c_3 \mid c_2, c_1, c_0)$. In addition to the N = 4 model, also construct N = 3, 2 and 1 models (for use when there's no information in the N = 4 case). Create these models by:

- Downloading a large text
- Processing it to remove all symbols other than the 26 characters and 1 space
- Counting how many times each possible sequence occurs and normalizing

Now choose plausible characters for 3 letters c_0, c_1, c_2 and start sequentially generating text:

$$c_0, c_1, c_2, c_3 = \arg\max p(c_3 \mid c_2, c_1, c_0),\ c_4 = \arg\max p(c_4 \mid c_3, c_2, c_1), \ldots$$

How well does it work? How long a word / phrase can you coherently generate? For additional information see: https://en.wikipedia.org/wiki/N-gram

7. [**Optional**] Take a relatively simple network and save the intermediate feature maps $\mathbf{x}$ after ReLU for a batch of images. Quantize these feature maps to unsigned 8 bits ([0, 255]) via $\mathbf{x}_q = \text{round}(255*\mathbf{x}/\max(\mathbf{x}))$. How many bytes (elements) are there in the quantized feature maps? Now create a Huffman code for this set of quantized feature maps and use it to compress the quantized feature maps. How many bytes are in the Huffman encoded quantized feature maps? Which value in [0, 255] is assigned the shortest code word?