CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 4 (Probability) Problem 1

Complete

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 4 (Probability) Problem 2

Complete

CS 6301.503 Spring 2019 — Scott C. Waggener (scw180000)

Homework 4 (Probability) Problem 3

(a) You're person B and want to win, which version of the game do you play?

Solution: The standard version ■

Proof: Think of this in terms of maximizing information gain with each question. If we do not receive an answer immedaitely after each question our best strategy would be to choose the top 20 questions that provide the most mutual information about the object without considering how information gain at a given question is conditioned on the answer of previous questions.

If we do receive an answer after each question, we can choose information maximizing questions based on known realizations of the random variables given as answers. □

(b) **Solution:** The modified game ■

Proof: The input tensor passes through multiple layers before being converted to a feature vector. The dense layer then uses this feature vector to eventually provide a class pmf vector. In this construction the feature vector can be thought of as answers to the 20 questions which are used to compute each class probability. There is no dependency on which features a network extracts at a given layer conditioned on the features extracted from a previous layer. □

(c) **Solution:** Essentially, provide redundancy assuming the probability of an incorrect answer is not 0.5. ■

Proof: Of course, if we are given an incorrect answer with probability 0.5 we have an entropy maximizing uniform distribution across both possible answers. We will receive no information from each answer and will be unable to play the game.

Otherwise, we can attempt to provide some redundancy in our questions to help minimize the uncertainty about answers. We can tailor the amount of overlap based on the probability of receiving a correct or incorrect answer, with greater overlap required as the probabilities approach 0.5/0.5. Furthermore, we can trivially handle the case where we are more likely to receive an incorrect answer by negating the given answer.

The chain rule can be written as

$$p\Big(x_{n-1}|\hat{x}_{n-2},\hat{x}_{n-3},\ldots,\hat{x}_0\Big) = \frac{p\Big(x_{n-1},\hat{x}_{n-2},\hat{x}_{n-3},\ldots,\hat{x}_0\Big)}{p\Big(\hat{x}_{n-2},\hat{x}_{n-3},\ldots,\hat{x}_0\Big)} \tag{1}$$

Tweaking the overlap of questions would exert control over the joint probability seen in the denominator (and partially in numerator).

□

(d) **Solution:** Strategy change is similar to the one used for the original game, but providing redundancy blindly during the questioning process. ■

Proof: Now we are playing the modified game, but with some probability the given answer will be wrong. At the end of the round we will have a probability for each class conditioned on 20 answers which are only correct with some probability. Again, we wouldn't play with a 0.5/0.5 probability split since each answer provides no information. Otherwise we need to provide some redundancy in our questions, but we cannot determine the best redundancy providing questions conditioned on answers to other questions.

Since our answers are only correct with some probability we cannot exclude any classes from our output pmf. Additionally, a network would provide this redundancy by extracting multiple features that may point to a given class. Since each extracted feature may be incorrect, having multiple features that indicate membership to a given class provides redundancy.

□

CS 6301.503 Spring 2019 **Homework 4 (Probability) Problem 4**	Scott C. Waggener (scw180000)

Solution: The plot was generated using 100 trials for each of the possible number of known questions. Sampling was done using `np.random.binomial` with $n = 1,\ p = 0.25,\ \text{size} = (100 - \text{known})$

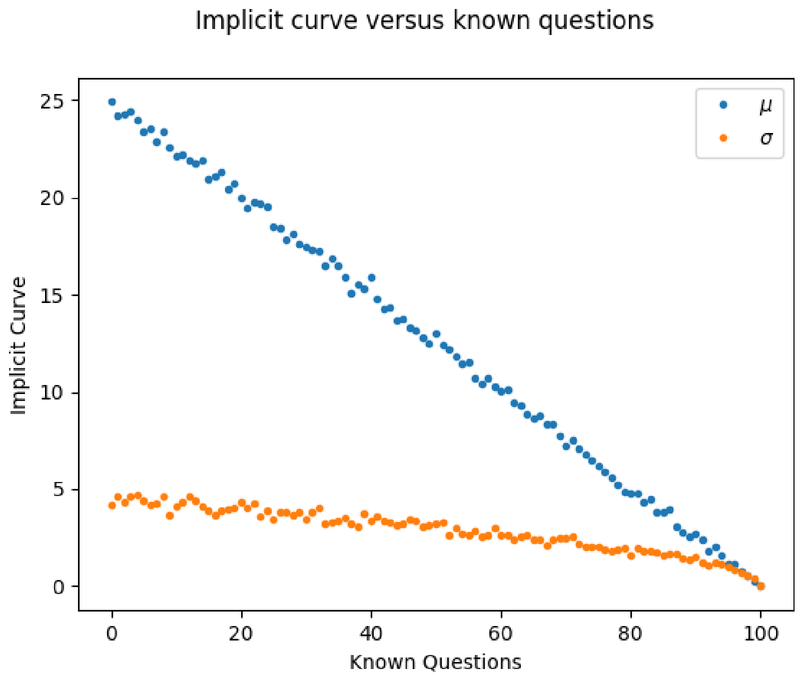

Figure 1. Implicit curve μ, σ versus number of known questions.

■

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 4 (Probability) Problem 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?

Solution: 10 bits ■

Proof: Since entropy can informally be defined as the information in the realization of a random variable, we can compute the entropy for the given problem. We have 1280 images in each of the 1000 classes, meaning

$$\begin{aligned} H(x(s)) &= -\sum_{i=1}^{1000} \frac{1}{1000} \log_2\left(\frac{1}{1000}\right) && (2) \\ &= -\frac{1000}{1000} \log_2\left(\frac{1}{1000}\right) && (3) \\ &= \log_2 1000 && (4) \\ &\approx 9.97 && (5) \end{aligned}$$

□