Chris Piech
CS 109

Section #8
Nov 28, 2017

Section 8: Machine Learning

1. **Vision Test**

   You decide that the vision tests given by eye doctors would be more precise if we used an approach inspired by logistic regression. In a vision test a user looks at a letter with a particular font size and either correctly guesses the letter or incorrectly guesses the letter.

   You assume that the probability that a particular patient is able to guess a letter correctly is:

   $$p = \sigma(\theta + f)$$

   Where θ is the user's vision score and f is the font size of the letter.

   Explain how you could estimate a user's vision score (θ) based on their 20 responses $(f^{(1)}, y^{(1)}) \ldots (f^{(20)}, y^{(20)})$, where $y^{(i)}$ is an indicator variable for whether the user correctly identified the ith letter and $f^{(i)}$ is the font size of the ith letter. Solve for any and all partial derivatives required by your answer.

2. **Run a Tensor Flow Algorithm**

   In section train a deep learning algorithm which can detect whether or not a hand written digit is a 1 or a 0. Design a simple deep learning model.

   

   Some questions that to think about:

   a. What is the difference between a test and a train set?
   b. How can you use log-likelihood to understand training?

3. **Deep Dream**

   Deep dream is an algorithm that can imagine what an image would look like if it had certain objects in it (eg take Van Gough's Starry Night and imagine it with fish). It works by using a trained deep learning model to modify image pixels.

Assume that you have a fully trained network similar to the one from the previous question, except that it is trained to detect if 8x8 images ($\mathbf{x}$) are pictures of cats ($Y = 1$) or not-cats ($Y = 0$). Your deep learning network has one hidden layer with 20 hidden neurons.

$$h_j = \sigma(\sum_{i=0}^{64} \theta_{i,j}^{(h)} \mathbf{x}_i) \qquad \hat{y} = \sigma(\sum_{i=0}^{20} \theta_{i}^{(y)} \mathbf{x}_i)$$

a. How can you chose the image from your training dataset which maximally activates the final neuron ($\hat{y}$, the one that indicates the model believes it is looking at a cat)?

b. How can you find the pixel values for a picture (likely not in your dataset) which maximally activates the second hidden neuron (h_2)? This is an involved question. Hint: You should not be changing the weights of your model, assume that they are fixed. Solve for any and all partial derivatives required by your answer.

c. Given an image, how can you alter it slightly so that it is more catlike?

4. **Multiclass Bayes**

In this problem we are going to explore how to write Naive Bayes for multiple output classes. We want to predict a single output variable Y which represents how a user feels about a book. Unlike in your homework, the output variable Y can take on one of the *four* values in the set {Like, Love, Haha, Sad}. We will base our predictions off of three binary feature variables $X_1, X_2,$ and X_3 which are indicators of the user's taste. All values $X_i \in \{0, 1\}$.

We have access to a dataset with 10,000 users. Each user in the dataset has a value for X_1, X_2, X_3 and Y. You can use a special query method **count** that returns the number of users in the dataset with the given *equality* constraints (and only equality constraints). Here are some example usages of **count**:

count($X_1 = 1, Y =$ Haha)	returns the number of users where $X_1 = 1$ and $Y =$ Haha.
count($Y =$ Love)	returns the number of users where $Y =$ Love.
count($X_1 = 0, X_3 = 0$)	returns the number of users where $X_1 = 0$, and $X_3 = 0$.

You are given a new user with $X_1 = 1, X_2 = 1, X_3 = 0$. What is the best prediction for how the user will feel about the book (Y)? You may leave your answer in terms of an argmax function. You should explain how you would calculate all probabilities used in your expression. Use **Laplace estimation** when calculating probabilities.