Lisa Yan
CS 109

Section #8
June 3-5, 2020

Section 8

---

1. **MLE and MAP**

   To start, let us look briefly at how we calculate the parameters for Maximum Likelihood and Maximum a Posteriori Estimation. As usual, we let f be some probability distribution function, and we let g some prior probability distribution function.

$$\theta_{MLE} = \arg\max_{\theta} \prod_{i=1}^{n} f(X_i|\theta)$$

$$\theta_{MAP} = \arg\max_{\theta} \prod_{i=1}^{n} f(X_i|\theta)g(\theta)$$

   These look suspiciously similar, which begs the question: what is the difference between MLE and MAP?

   (a) The only difference between the MLE and MAP updates is the introduction of an additional equation $g(\theta)$. In words, what are we doing when we add $g(\theta)$?

   (b) Write the log likelihood function $LL(\theta)$ for both MLE and MAP, being sure to use the properties of log to simplify your work.

   (c) Let us now look at $\theta = \arg\max LL(\theta)$ for θ_{MAP} and θ_{MLE}. Notably, there is one additional term for θ_{MAP} — this is not surprising, given our earlier observation. What does that additional term need to equal so that θ_{MAP} and θ_{MLE} are equal?

   (d) If we do the above so that $\theta_{MAP} = \theta_{MLE}$, what kind of probability distribution function should $g(\theta)$ be equal to? And what does that say about the relationship between θ_{MAP} and θ_{MLE}?

2. **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$(X_1 = 1, Y = \text{Haha})$	returns the number of users where $X_1 = 1$ and $Y = \text{Haha}$.
count$(Y = \text{Love})$	returns the number of users where $Y = \text{Love}$.
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.

3. **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 the approach you describe in your answer.

Formula reference for Logistic Regression:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

$$LL(\theta) = \sum_{i=0}^{n} y^{(i)} \log \sigma(\theta^T \mathbf{x}^{(i)}) + (1 - y^{(i)}) \log \left[1 - \sigma(\theta^T \mathbf{x}^{(i)})\right]$$

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=0}^{n} \left[y^{(i)} - \sigma(\theta^T \mathbf{x}^{(i)}) \right] x_j^{(i)}$$

4. **The Most Important Features**

Let's explore saliency, a measure of how important a feature is for classification. We define the saliency of the ith input feature for a given example $(\mathbf{x}, y)$ to be the absolute value of the partial derivative of the log likelihood of the sample prediction, with respect to that input feature $\left|\frac{\partial LL}{\partial x_i}\right|$. In the images below, we show both input images and the corresponding saliency of the input features (in this case, input features are pixels):

First consider a trained logistic regression classifier with weights θ. Like the logistic regression classifier that you wrote in your homework it predicts binary class labels. In this question we allow the values of $\mathbf{x}$ to be real numbers, which doesn't change the algorithm (neither training nor testing).

a. What is the Log Likelihood of a single training example $(\mathbf{x}, y)$ for a logistic regression classifier?

b. Calculate is the saliency of a single feature (x_i) in a training example $(\mathbf{x}, y)$.

c. Show that the ratio of saliency for features i and j is the ratio of the absolute value of their weights $\frac{|\theta_i|}{|\theta_j|}$.