Juliette Woodrow
CS 109

Section #8
March 5, 2026

Section 8

Warmup!

(a) True or False: The log likelihood function that is used to estimate p of a Bernoulli, for a set of observations, must always be 0 or smaller. Briefly explain why.

Yes. All probabilities are between 0 and 1, so all logs of probabilities are negative! For example, $\log(0.5) \approx -0.69$.

(b) (Optional) When implementing logistic regression, a student decides to add a second intercept value. To do so they add an extra feature with value 0 to each datapoint. How will this impact training?

There will be no impact on training. When we compute $\theta^T x = \sum_{j=1}^{n} x_j \theta_j$, this new feature will have no contribution to the product, since no matter what theta value it is multiplied with, the result will still be zero.

(c) (Optional) When implementing logistic regression, a student decides to incorporate an additional term that represents a non-linear relationship or "interaction" between the features. Their dataset has two features, x_1 and x_2, and a corresponding label y for each datapoint. They add the interaction term $x_1 \cdot x_2$, so that the full model is $P(Y = y|X = x) = \sigma(\theta_0 + \theta_1 \cdot x_1 + \theta_2 \cdot x_2 + \theta_3 \cdot x_1 \cdot x_2)$. Explain briefly how this change impacts model training.

The training code would need to compute this feature (multiply x_1 by x_2) for each datapoint, so that the number of features and the length of the list of thetas increases by one. The gradient list is also one value longer, to include the gradient for the new theta. Inside the training for loop, the theta for this feature will be updated along with the others at each step.

Recalibrating an Uncalibrated Model

You have an uncalibrated binary classification model that outputs values $\hat{p} \in [0, 1]$. These outputs are meant to be the probability that $Y = 1$. However, the outputs from this model are not well-calibrated. For instance, among all examples where $\hat{p} = 0.9$, it was the case that Y was 1 only 70% of the time. To recalibrate the models outputs you decide to use Platt Recalibration, where the corrected probability that $Y = 1$ is:

$$P(Y = 1 \mid \hat{p}) = \sigma(a \cdot \hat{p} - 0.5)$$

$\sigma(z) = 1/(1 + e^{-z})$ is the sigmoid function and a is the parameter of the recalibration model. Here is the partial derivative of the Platt Recalibration model with respect to a:

$$\frac{\partial}{\partial a}\sigma(a \cdot \hat{p} - 0.5) = \sigma(a \cdot \hat{p} - 0.5) \cdot [1 - \sigma(a \cdot \hat{p} - 0.5)] \cdot \hat{p}$$

1. For a new datapoint the uncalibrated model outputs $\hat{p}$ of 0.9. If you use Platt Recalibration with $a = 2$ what is the recalibrated probability that $Y = 1$?

   Plug into the formula: $a \cdot \hat{p} - 0.5 = 2(0.9) - 0.5 = 1.3$, so the recalibrated probability is $\sigma(1.3)$.

2. You are given a training dataset with n outputs from the uncalibrated model $(\hat{p}^{(i)}, y^{(i)})$ where $\hat{p}^{(i)}$ is the uncalibrated output and $y^{(i)} \in \{0, 1\}$ is the true binary outcome. Explain how you could estimate the value of a that makes the $y^{(i)}$ values as likely as possible. Solve for any and all partial derivatives required by your answer.

   Solution v1 (more explanation):

   This problem is related to logistic regression. In both, we can use MLE to estimate parameters, and in both, the likelihood comes from the continuous PMF of the Bernoulli, since here we are still doing binary classification (Y is either 0 or 1):

   $$L(a) = \prod_{i=1}^{n} P(Y = 1 \mid \hat{p}^{(i)})^{y^{(i)}} (1 - P(Y = 1 \mid \hat{p}^{(i)}))^{1-y^{(i)}}$$

   $$LL(a) = \sum_{i=1}^{n} y^{(i)} \log P(Y = 1 \mid \hat{p}^{(i)}) + (1 - y^{(i)}) \log(1 - P(Y = 1 \mid \hat{p}^{(i)}))$$

   To find the value of a that maximizes $LL(a)$, we can take the derivative using the chain rule:

   $$\frac{\partial LL(a)}{\partial a} = \frac{\partial LL(a)}{\partial P(Y = 1 \mid \hat{p}^{(i)})} \cdot \frac{\partial P(Y = 1 \mid \hat{p}^{(i)})}{\partial a}$$

The second component is given to us in the problem (it is equivalent to $\sigma(a \cdot \hat{p}^{(i)} - 0.5) \cdot [1 - \sigma(a \cdot \hat{p}^{(i)} - 0.5)] \cdot \hat{p}^{(i)}$). The first term looks the same as in logistic regression:

$$\frac{\partial LL(a)}{\partial P(Y = 1 \mid \hat{p}^{(i)})} = \sum_{i=1}^{n} \left(\frac{y^{(i)}}{P(Y = 1 \mid \hat{p}^{(i)})} - \frac{1 - y^{(i)}}{1 - P(Y = 1 \mid \hat{p}^{(i)})} \right)$$

Using this derivative, you would find the best estimate for a using a gradient ascent.

Solution v2 (less explanation):

To estimate a, set up the log-likelihood for Bernoulli outcomes and differentiate with respect to a. Using the chain rule:

$$\frac{\partial LL(a)}{\partial a} = \sum_{i=1}^{n} \left(\frac{y^{(i)}}{P_i} - \frac{1 - y^{(i)}}{1 - P_i} \right) \cdot \sigma(a\hat{p}^{(i)} - 0.5) \left[1 - \sigma(a\hat{p}^{(i)} - 0.5) \right] \hat{p}^{(i)}$$

where $P_i = P(Y = 1 \mid \hat{p}^{(i)})$.

Then update a using gradient ascent:

$$a \leftarrow a + \eta \frac{\partial LL(a)}{\partial a}$$

This gives the MLE estimate of the recalibration slope a.

Vision Test Logistic Regression

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. This formula uses the sigmoid function:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \qquad \frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$$

Explain how you could estimate a user's vision score (θ) based on their 20 responses, $(f^{(1)}, y^{(1)})$ to $(f^{(20)}, y^{(20)})$, 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 all partial derivatives necessary.

We are going to solve this problem by finding the MLE estimate of θ. To find the MLE estimate, we are going to find the argmax of the log likelihood function. To calculate argmax we are going to use gradient ascent, which requires that we know the partial derivative of the log likelihood function with respect to theta.

First we write the log likelihood:

$$L(\theta) = \prod_{i=1}^{20} p^{y_i}(1-p)^{[1-y_i]}$$

$$LL(\theta) = \sum_{i=1}^{20} (y_i \log(p) + (1 - y_i) \log(1 - p))$$

Then we find the derivative of log likelihood with respect to θ. We first do this for one data point:

$$\frac{\partial LL}{\partial \theta} = \frac{\partial LL}{\partial p} \cdot \frac{\partial p}{\partial \theta}$$

We can calculate both the smaller partial derivatives independently:

$$\frac{\partial LL}{\partial p} = \frac{y_i}{p} - \frac{1 - y_i}{1 - p}$$

$$\frac{\partial p}{\partial \theta} = p[1 - p]$$

Putting it all together for one letter:

$$\begin{aligned}\frac{\partial LL}{\partial \theta} &= \frac{\partial LL}{\partial p} \cdot \frac{\partial p}{\partial \theta} \\ &= \Big[\frac{y_i}{p} - \frac{1 - y_i}{1 - p}\Big] p[1 - p] \\ &= y_i(1 - p) - p(1 - y_i) \\ &= y_i - p \\ &= y_i - \sigma(\theta + f)\end{aligned}$$

For all twenty examples:

$$\frac{\partial LL}{\partial \theta} = \sum_{i=1}^{20} y_i - \sigma(\theta + f^{(i)})$$

Note: In this problem you've practiced all the mechanics and math behind logistic regression. The problem has all the same structural features (including a similar assumption), but it is not itself, logistic regression. Wahoo!

Decoding Movement for a Brain-Controlled Prosthetic Leg (Optional)

Engineers are designing a brain-controlled prosthetic ankle that infers a user's intended movement from electrical activity in their leg muscles. To train the system, the user performs known movements while electrodes measure muscle activity (EMG), producing labeled data that link muscle signals to intended actions.

At each time step, two muscle sensors are recorded:

- S_{TA}: tibialis anterior ("lift up" muscle),
- S_{GA}: gastrocnemius ("press down" muscle).

Each sensor reading is labeled as either `Active` (A) or `Quiet` (Q). The user can intend one of three movements:

$$U = \text{lift foot up}, \qquad D = \text{press foot down}, \qquad N = \text{neutral/relax}.$$

During calibration, the engineers measured how often each sensor fired while the user intended each movement. They found:

- When the user intends Up, sensor TA is Active 90% of the time and sensor GA is Active 20% of the time.
- When the user intends Down, sensor TA is Active 10% of the time and sensor GA is Active 85% of the time.
- When the user intends Neutral, sensor TA is Active 10% of the time and sensor GA is Active 10% of the time.

Engineers also found that when walking, a user spends about 30% of the time intending "Up," about 30% intending "Down," and the remaining 40% in "Neutral." Engineers model the two muscle sensors as independent once the users intended movement is specified.

We observe $S_{\text{TA}} = \text{Active}$ and $S_{\text{GA}} = \text{Active}$, which intended movement $M \in \{U, D, N\}$ is most likely?

First, we can formalize the information that has been provided to us in the problem statement.

Given Information:

- Prior probabilities: $P(U) = 0.30$, $P(D) = 0.30$, $P(N) = 0.40$
- Likelihoods for sensor TA:

$$P(S_{TA} = A \mid U) = 0.90, \quad P(S_{TA} = A \mid D) = 0.10, \quad P(S_{TA} = A \mid N) = 0.10$$

- Likelihoods for sensor GA:

$$P(S_{GA} = A \mid U) = 0.20, \quad P(S_{GA} = A \mid D) = 0.85, \quad P(S_{GA} = A \mid N) = 0.10$$

Objective: Find the most likely movement given both sensors are active, we want to find the movement that results in the highest probability given what we have observed about the sensors:

$$M^* = \arg \max_{M \in \{U,D,N\}} P(M \mid S_{TA} = A, S_{GA} = A)$$

Approach: Apply Bayes' theorem:

$$P(M \mid S_{TA} = A, S_{GA} = A) = \frac{P(S_{TA} = A, S_{GA} = A \mid M) \cdot P(M)}{P(S_{TA} = A, S_{GA} = A)}$$

Since the denominator is constant across all movements, we can focus on only maximizing the numerator:

$$M^* = \arg \max_{M \in \{U,D,N\}} P(S_{TA} = A, S_{GA} = A \mid M) \cdot P(M)$$

Calculations:
We can use the definition of conditional independence applied here to simplify our calculations: $P(S_{TA}, S_{GA} \mid M) = P(S_{TA} \mid M) \cdot P(S_{GA} \mid M)$ Using conditional independence:

$$\begin{aligned} P(S_{TA} = A, S_{GA} = A \mid U) &= 0.90 \times 0.20 = 0.18 \\ P(S_{TA} = A, S_{GA} = A \mid D) &= 0.10 \times 0.85 = 0.085 \\ P(S_{TA} = A, S_{GA} = A \mid N) &= 0.10 \times 0.10 = 0.01 \end{aligned}$$

Computing the unnormalized posteriors:

$$\begin{aligned} P(U) \cdot P(S_{TA} = A, S_{GA} = A \mid U) &= 0.30 \times 0.18 = 0.054 \\ P(D) \cdot P(S_{TA} = A, S_{GA} = A \mid D) &= 0.30 \times 0.085 = 0.0255 \\ P(N) \cdot P(S_{TA} = A, S_{GA} = A \mid N) &= 0.40 \times 0.01 = 0.004 \end{aligned}$$

Answer: Since $0.054 > 0.0255 > 0.004$, the most likely intended movement is **Up (U)**.

Normalization: We can calculate the exact probabilities using normalization.

To obtain the exact posterior probabilities, we normalize by dividing each unnormalized posterior by their sum:

First, compute the normalizing constant (evidence):

$$\begin{aligned} P(S_{TA} = A, S_{GA} = A) &= \sum_{M \in \{U,D,N\}} P(S_{TA} = A, S_{GA} = A \mid M) \cdot P(M) \\ &= 0.054 + 0.0255 + 0.004 \\ &= 0.0835 \end{aligned}$$

Then, compute the normalized posteriors:

$$P(U \mid S_{TA} = A, S_{GA} = A) = \frac{0.054}{0.0835} \approx 0.647$$

$$P(D \mid S_{TA} = A, S_{GA} = A) = \frac{0.0255}{0.0835} \approx 0.305$$

$$P(N \mid S_{TA} = A, S_{GA} = A) = \frac{0.004}{0.0835} \approx 0.048$$

Verification: $0.647 + 0.305 + 0.048 = 1.000$ ✓

$$P(U \mid \text{both active}) \approx 64.7\%, \quad P(D \mid \text{both active}) \approx 30.5\%, \quad P(N \mid \text{both active}) \approx 4.8\%$$