HW 3

Kevin Lin

2/23/2026

1

Given $X_1, X_2, \dots, X_n$ are n i.i.d. Bernoulli random variables following:

$$P(X = x; p) = p^x (1-p)^{1-x}, \quad x = 0, 1$$

where p is the probability of X being 1. We want to find the MLE of p using the given samples. The likelihood function is given by:

$$L(p) = \prod_{i=1}^{n} P(X_i = x_i; p) = \prod_{i=1}^{n} p^{x_i} (1-p)^{1-x_i}$$

To find the MLE, we take the natural logarithm of the likelihood function:

$$\ell(p) = \ln L(p) = \sum_{i=1}^{n} \left(x_i \ln p + (1 - x_i) \ln(1-p) \right)$$

Next, we differentiate $\ell(p)$ with respect to p and set it to zero to find the critical points:

$$\frac{d\ell(p)}{dp} = \sum_{i=1}^{n} \left(\frac{x_i}{p} - \frac{1-x_i}{1-p} \right) = 0$$

This simplifies to:

$$\sum_{i=1}^{n} \frac{x_i}{p} = \sum_{i=1}^{n} \frac{1-x_i}{1-p}$$

Multiplying both sides by $p(1-p)$ gives:

$$(1-p) \sum_{i=1}^{n} x_i = p \sum_{i=1}^{n} (1 - x_i)$$

Let $S = \sum_{i=1}^{n} x_i$, then we can rewrite the equation as:

$$\begin{aligned}(1-p)S &= p(n-S) \\ S - pS &= pn - pS \\ S &= pn \\ p &= \frac{S}{n} \therefore \\ \hat{p} &= \frac{\sum_{i=1}^{n} x_i}{n}\end{aligned}$$

Thus, the MLE of p is the sample mean of the observed data.

2

We are given a binary classification problem with outputs $y \in \{0, 1\}$, and assume that $p(y|x)$ is Bernoulli $\hat{p}(x; \theta^*)$ for some parameter vector $\theta^* \in \mathbb{R}^d$, where for each θ, $\hat{p}(x; \theta)$ returns a number between $[0, 1]$.

(a) We know the cross-entropy loss for each parameter θ for a set of n examples with predictions $\hat{p}(x_i; \theta)$ and true labels y_i is:

$$CE(\theta) = -\frac{1}{n} \sum_{i=1}^{n} (y_i \ln \hat{p}_i + (1 - y_i) \ln(1 - \hat{p}_i))$$

We can derive this loss function from the principle of the MLE:

$$\begin{aligned}\hat{\theta} &= \arg\max_{\theta} P(y_1, y_2, \ldots, y_n | x_1, x_2, \ldots, x_n; \theta) \\ &= \arg\max_{\theta} \prod_{i=1}^{n} P(y_i | x_i; \theta) \\ &= \arg\max_{\theta} \prod_{i=1}^{n} \hat{p}(x_i; \theta)^{y_i} (1 - \hat{p}(x_i; \theta))^{1-y_i} \\ &= \arg\max_{\theta} \sum_{i=1}^{n} (y_i \ln \hat{p}(x_i; \theta) + (1 - y_i) \ln(1 - \hat{p}(x_i; \theta))) \\ &= \arg\min_{\theta} -\frac{1}{n} \sum_{i=1}^{n} (y_i \ln \hat{p}(x_i; \theta) + (1 - y_i) \ln(1 - \hat{p}(x_i; \theta))) \\ &= \arg\min_{\theta} CE(\theta)\end{aligned}$$

Thus, minimizing the cross-entropy loss is equivalent to maximizing the likelihood of the observed data.

(b) Let's instantiate the prediction as the sigmoid function applied to a linear combination of input features $\hat{p}(x; \theta) = \sigma(\theta^T x)$, where $\sigma(z) = \frac{1}{1+e^{-z}}$.

Then, minimizing the cross-entropy loss is equivalent to minimizing the familiar logistic loss given by:

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ln(1 + e^{-\tilde{y}_i \theta^T x_i}) \text{ where } \tilde{y}_i = \begin{cases} 1 & \text{if } y_i = 1 \\ -1 & \text{if } y_i = 0 \end{cases}$$

We can show this by substituting $\hat{p}(x_i;\theta) = \sigma(\theta^T x_i)$ into the cross-entropy loss:

$$CE(\theta) = -\frac{1}{n}\sum_{i=1}^{n} \big(y_i \ln \sigma(\theta^T x_i) + (1 - y_i)\ln(1 - \sigma(\theta^T x_i))\big)$$

$$= -\frac{1}{n}\sum_{i=1}^{n} \left(y_i \ln \frac{1}{1 + e^{-\theta^T x_i}} + (1 - y_i)\ln\left(1 - \frac{1}{1 + e^{-\theta^T x_i}}\right)\right)$$

$$= -\frac{1}{n}\sum_{i=1}^{n} \left(y_i \ln \frac{1}{1 + e^{-\theta^T x_i}} + (1 - y_i)\ln \frac{1}{1 + e^{\theta^T x_i}}\right)$$

$$= -\frac{1}{n}\sum_{i=1}^{n} \left(-y_i \ln(1 + e^{-\theta^T x_i}) - (1 - y_i)\ln(1 + e^{\theta^T x_i})\right)$$

$$= \frac{1}{n}\sum_{i=1}^{n} \left(y_i \ln(1 + e^{-\theta^T x_i}) + (1 - y_i)\ln(1 + e^{\theta^T x_i})\right)$$

We know that:

$$\tilde{y}_i = \begin{cases} 1 & \text{if } y_i = 1 \\ -1 & \text{if } y_i = 0 \end{cases}$$

Thus, we can define $\tilde{y}_i$ such that:

$$\tilde{y}_i = 2y_i - 1 \therefore$$
$$y_i = \frac{1 + \tilde{y}_i}{2}, 1 - y_i = \frac{1 - \tilde{y}_i}{2}$$

Substituting this back into the expression for $CE(\theta)$ gives:

$$CE(\theta) = \frac{1}{n}\sum_{i=1}^{n} \left(\frac{1 + \tilde{y}_i}{2}\ln(1 + e^{-\theta^T x_i}) + \frac{1 - \tilde{y}_i}{2}\ln(1 + e^{\theta^T x_i})\right)$$

Thus, if we plug in $\tilde{y}_i = 1$, we get:

$$CE(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ln(1 + e^{-\theta^T x_i})$$

And if we plug in $\tilde{y}_i = -1$, we get:

$$CE(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ln(1 + e^{\theta^T x_i})$$

which is equivalent to the logistic loss function:

$$L(\theta) = \begin{cases} \frac{1}{n}\sum_{i=1}^{n} \ln(1 + e^{-\theta^T x_i}) & \text{if } \tilde{y}_i = 1 \\ \frac{1}{n}\sum_{i=1}^{n} \ln(1 + e^{\theta^T x_i}) & \text{if } \tilde{y}_i = -1 \end{cases}$$

3

4

(a) See attached Jupyter notebook for code and plots.

(b)

Dataset	Gaussian Naive Bayes	Linear Logistic Regression
1	Each feature has mixed overlap between 1 and 2. Because GNB assumes that each feature is independent, they are modeled separately and thus the boundary becomes a vertical line.	Dataset is linearly separable so logistic regression performs really well, giving us a clean fit.
2	The decision boundary becomes quadratic, so GNB can create a smooth circled boundary. However because the features are correlated in a slanted ellipse, naive bayes fails to recognize this and thus still somewhat poorly fits, but is still better than logistic regression.	Fails as data is not linearly separable but rather overlaps. Thus, it draws a line that minimizes classification error, but it cannot match the ellipse structure.
3	Each class is radially symmetric, and even though features are not independent (related to radius), GNB can still model it well in circles.	Fails as data is not linearly separable. Thus again, it draws a line that minimizes error but cannot match circular shape as it is constrained to 1 dimension