Data Science for Mathematicians
Exercises: Logistic Regression and Generalized Linear Models

Instructions

Show all working and justify your answers. State any assumptions you make. When computing sigmoid values, you may use a calculator or leave answers in terms of $\sigma(\cdot)$ where exact decimal values are not essential. For proofs, state clearly which definitions, theorems, or propositions from the lecture you invoke.

Exercises

Exercise 1. Logit and sigmoid computations

(a) Compute $\operatorname{logit}(p)$ for $p = 0.2,\ 0.5,\ 0.8,\ 0.95$.

(b) Compute $\sigma(z)$ for $z = -3,\ -1,\ 0,\ 1,\ 3$.

(c) Verify that $\sigma(\operatorname{logit}(0.8)) = 0.8$ and $\operatorname{logit}(\sigma(1)) = 1$.

Exercise 2. Binary cross-entropy loss evaluation
A logistic regression model produces the following predicted probabilities for four observations with true labels $\mathbf{y} = (1, 0, 1, 0)^T$:

$$\mathbf{p} = (0.9,\ 0.3,\ 0.6,\ 0.1)^T.$$

(a) Compute the per-sample loss J_i for each observation.

(b) Compute the total loss $J = \sum_{i=1}^{4} J_i$.

(c) Suppose the model instead predicted $\mathbf{p}' = (0.5,\ 0.5,\ 0.5,\ 0.5)^T$. Compute the total loss and compare it with part (b). Which model is better, and why?

Exercise 3. One iteration of gradient descent
Consider a dataset with $n = 4$ observations and a single predictor:

i	1	2	3	4
x_i	-1	0	1	2
y_i	0	0	1	1

We fit the model $p(x) = \sigma(\beta_0 + \beta_1 x)$ with $\boldsymbol{\beta}^{(0)} = (0, 0)^T$ and learning rate $\eta = 0.1$.

(a) Write down the augmented design matrix $\mathbf{X}$ and the label vector $\mathbf{y}$.

(b) Compute the predicted probabilities $\mathbf{p}$ at iteration 0.

(c) Compute the gradient $\nabla J = \mathbf{X}^T(\mathbf{p} - \mathbf{y})$.

(d) Perform the parameter update to obtain $\boldsymbol{\beta}^{(1)}$.

(e) Compute the BCE loss $J(\boldsymbol{\beta}^{(0)})$ and $J(\boldsymbol{\beta}^{(1)})$, and verify that the loss decreased.

Exercise 4. Odds ratio interpretation

A logistic regression model for predicting diabetes ($y = 1$) from three predictors yields the fitted coefficients:

$$\hat{\beta}_0 = -4.0, \quad \hat{\beta}_1 = 0.035 \text{ (age)}, \quad \hat{\beta}_2 = 0.50 \text{ (BMI)}, \quad \hat{\beta}_3 = -0.80 \text{ (exercise hours/week)}.$$

(a) Compute the odds ratio $e^{\hat{\beta}_j}$ for each predictor and interpret each in one sentence.

(b) For a 50-year-old patient with BMI = 30 and exercise = 3 hours/week, compute the linear predictor η, the predicted probability $\hat{p}$, and the predicted class at threshold 0.5.

(c) By how many hours per week must this patient increase exercise (holding age and BMI constant) to bring $\hat{p}$ below 0.5?

Exercise 5. Symmetry of the sigmoid

(a) Prove that $\sigma(-z) = 1 - \sigma(z)$ for all $z \in \mathbb{R}$ directly from the definition $\sigma(z) = 1/(1 + e^{-z})$.

(b) Using part (a) and the derivative formula $\sigma'(z) = \sigma(z)(1 - \sigma(z))$, show that the derivative is symmetric about $z = 0$, i.e., $\sigma'(-z) = \sigma'(z)$.

(c) Prove that $z = 0$ is the unique global maximum of $\sigma'(z)$ and compute its value.

Exercise 6. Convexity of the per-sample loss

Consider the per-sample BCE loss for a single observation (x, y) with $y \in \{0, 1\}$ and a scalar parameter β (no intercept):

$$J(\beta) = -\big[y \log \sigma(\beta x) + (1 - y) \log(1 - \sigma(\beta x))\big].$$

(a) Compute $\frac{\mathrm{d}J}{\mathrm{d}\beta}$ using the chain rule. Verify that you obtain $(\sigma(\beta x) - y)x$.

(b) Compute $\frac{\mathrm{d}^2 J}{\mathrm{d}\beta^2}$ and show that it equals $\sigma(\beta x)(1 - \sigma(\beta x))x^2$.

(c) Conclude that $J(\beta)$ is convex in β. Under what condition on the data is it *strictly* convex?

Exercise 7. Equivalence of MLE and BCE minimization

Let $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with $y_i \in \{0, 1\}$ be a binary classification dataset modeled by $\mathbb{P}(Y_i = 1 \mid \mathbf{x}_i) = \sigma(\boldsymbol{\beta}^T \mathbf{x}_i)$.

(a) Write down the log-likelihood $\ell(\boldsymbol{\beta})$ for this model.

(b) Show that $\arg\max_{\boldsymbol{\beta}} \ell(\boldsymbol{\beta}) = \arg\min_{\boldsymbol{\beta}} J(\boldsymbol{\beta})$, where J is the BCE loss.

(c) Prove that the gradient of the log-likelihood is $\nabla \ell(\boldsymbol{\beta}) = \mathbf{X}^T(\mathbf{y} - \mathbf{p})$, where $\mathbf{p}$ is the vector of predicted probabilities. Relate this to the gradient of J.

Exercise 8. Exponential family verification

The Poisson distribution has probability mass function $\mathbb{P}(Y = k) = \frac{\lambda^k e^{-\lambda}}{k!}$ for $k = 0, 1, 2, \ldots$

(a) Rewrite the Poisson PMF in exponential family form $\exp\left\{\frac{y\theta - b(\theta)}{\phi} + c(y, \phi)\right\}$. Identify θ, $b(\theta)$, ϕ, and $c(y, \phi)$.

(b) Verify that $\mathbb{E}[Y] = b'(\theta)$ and $\mathrm{Var}(Y) = \phi\, b''(\theta)$.

(c) What is the canonical link function for the Poisson distribution? Justify your answer.

Exercise 9. Failure modes of OLS for classification

An analyst fits a linear regression model $\hat{y} = \beta_0 + \beta_1 x$ to a binary classification dataset and obtains $\hat{\beta}_0 = 0.4$ and $\hat{\beta}_1 = 0.15$.

(a) For what range of x does this model predict $\hat{y} \notin [0, 1]$?

(b) Compute $\mathrm{Var}(Y_i \mid x_i)$ at $x = 0, 2, 4$ under the Bernoulli assumption $p_i = \hat{y}_i$. Does the constant-variance assumption of OLS hold?

(c) Explain, in precise mathematical terms, why the residuals $\epsilon_i = y_i - \hat{y}_i$ cannot follow a Gaussian distribution when $y_i \in \{0, 1\}$.

Exercise 10. Decision boundary analysis

A logistic regression model for classifying emails as spam ($y = 1$) or not spam ($y = 0$) uses two features: x_1 (number of exclamation marks) and x_2 (email length in words). The fitted parameters are $\hat{\beta}_0 = -2.0$, $\hat{\beta}_1 = 1.5$, and $\hat{\beta}_2 = -0.01$.

(a) Write down the equation of the decision boundary (the set of points where $\hat{p} = 0.5$) and sketch it in the (x_1, x_2)-plane.

(b) An email has $x_1 = 3$ exclamation marks and $x_2 = 100$ words. Compute $\hat{p}$ and the predicted class.

(c) A colleague suggests that since $|\hat{\beta}_2|$ is small, the feature x_2 is unimportant. Critique this claim, considering the scale of x_2 relative to x_1.

Exercise 11. Comparing GLM components

A transportation agency models the number of traffic accidents Y_i at intersection i as a function of daily vehicle count x_{i1} (in thousands) and number of lanes x_{i2}.

(a) Argue why a Poisson GLM is more appropriate than linear regression for this problem. Address both the distribution of the response and the range constraint on the mean.

(b) Write down the three GLM components (random, systematic, link) for this model using the canonical link.

(c) The fitted model yields $\hat{\beta}_0 = -0.50$, $\hat{\beta}_1 = 0.12$, $\hat{\beta}_2 = 0.30$. Interpret $e^{\hat{\beta}_1}$ and $e^{\hat{\beta}_2}$ in the context of the problem.

(d) Predict the expected number of accidents for an intersection with $x_1 = 10$ (thousand vehicles) and $x_2 = 4$ lanes.

Exercise 12. Gradient descent convergence analysis

You are implementing logistic regression from scratch in Python. After running gradient descent for 100 iterations with learning rate $\eta = 0.5$ on a dataset with $n = 200$ and $p = 5$, you observe that the loss oscillates wildly and does not converge.

(a) Explain why, despite the convexity of the BCE loss, gradient descent can still fail to converge. What role does the learning rate play?

(b) Propose two concrete modifications to the algorithm that could resolve the issue, and explain why each would help.

(c) Write a NumPy function that performs a single gradient descent step for logistic regression. The function should take $\mathbf{X}$, $\mathbf{y}$, $\boldsymbol{\beta}$, and η as inputs and return the updated $\boldsymbol{\beta}$ and the current loss.

Solutions

Solution 1. Logit and sigmoid computations

(a) Using $\operatorname{logit}(p) = \log\bigl(p/(1-p)\bigr)$:

$$\begin{aligned}
\operatorname{logit}(0.2) &= \log\frac{0.2}{0.8} = \log 0.25 \approx -1.386,\\
\operatorname{logit}(0.5) &= \log\frac{0.5}{0.5} = \log 1 = 0,\\
\operatorname{logit}(0.8) &= \log\frac{0.8}{0.2} = \log 4 \approx 1.386,\\
\operatorname{logit}(0.95) &= \log\frac{0.95}{0.05} = \log 19 \approx 2.944.
\end{aligned}$$

(b) Using $\sigma(z) = 1/(1+e^{-z})$:

$$\begin{aligned}
\sigma(-3) &= \frac{1}{1+e^{3}} \approx \frac{1}{1+20.086} \approx 0.047,\\
\sigma(-1) &= \frac{1}{1+e^{1}} \approx \frac{1}{1+2.718} \approx 0.269,\\
\sigma(0) &= \frac{1}{1+1} = 0.5,\\
\sigma(1) &= \frac{1}{1+e^{-1}} \approx \frac{1}{1+0.368} \approx 0.731,\\
\sigma(3) &= \frac{1}{1+e^{-3}} \approx \frac{1}{1+0.050} \approx 0.953.
\end{aligned}$$

(c) Since the sigmoid is the inverse of the logit, we have $\sigma(\operatorname{logit}(p)) = p$ for all $p \in (0,1)$, and $\operatorname{logit}(\sigma(z)) = z$ for all $z \in \mathbb{R}$.
For the first identity: $\operatorname{logit}(0.8) = \log 4$, so $\sigma(\log 4) = 1/(1+e^{-\log 4}) = 1/(1+1/4) = 4/5 = 0.8$.
For the second identity: $\sigma(1) = 1/(1+e^{-1})$, and

$$\operatorname{logit}\left(\frac{1}{1+e^{-1}}\right) = \log\frac{1/(1+e^{-1})}{e^{-1}/(1+e^{-1})} = \log\frac{1}{e^{-1}} = \log e = 1.$$

Solution 2. Binary cross-entropy loss evaluation

(a) The per-sample loss is $J_i = -[y_i \log p_i + (1-y_i)\log(1-p_i)]$.

$$\begin{aligned}
J_1 &= -[1\cdot\log(0.9) + 0\cdot\log(0.1)] = -\log(0.9) \approx 0.105,\\
J_2 &= -[0\cdot\log(0.3) + 1\cdot\log(0.7)] = -\log(0.7) \approx 0.357,\\
J_3 &= -[1\cdot\log(0.6) + 0\cdot\log(0.4)] = -\log(0.6) \approx 0.511,\\
J_4 &= -[0\cdot\log(0.1) + 1\cdot\log(0.9)] = -\log(0.9) \approx 0.105.
\end{aligned}$$

(b) The total loss is

$$J = 0.105 + 0.357 + 0.511 + 0.105 = 1.078.$$

(c) With $\mathbf{p}' = (0.5, 0.5, 0.5, 0.5)^T$, each per-sample loss is $-\log(0.5) = \log 2 \approx 0.693$, so the total loss is $J' = 4\log 2 \approx 2.773$. Since $J = 1.078 < 2.773 = J'$, the first model is better. The uniform prediction $p_i = 0.5$ corresponds to the uninformative initialization; the first model's lower loss reflects that it has learned useful structure from the data.

Solution 3. One iteration of gradient descent

(a) The augmented design matrix and label vector are

$$\mathbf{X} = \begin{pmatrix} 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{pmatrix}, \qquad \mathbf{y} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}.$$

(b) At $\boldsymbol{\beta}^{(0)} = (0,0)^T$, every linear predictor is $z_i = 0$, so $p_i = \sigma(0) = 0.5$ for all i. Thus $\mathbf{p} = (0.5, 0.5, 0.5, 0.5)^T$.

(c) The residual vector is $\mathbf{p} - \mathbf{y} = (0.5, 0.5, -0.5, -0.5)^T$. The gradient is

$$\nabla J = \mathbf{X}^T(\mathbf{p} - \mathbf{y}) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ -1 & 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} 0.5 \\ 0.5 \\ -0.5 \\ -0.5 \end{pmatrix} = \begin{pmatrix} 0 \\ -1.5 \end{pmatrix}.$$

(d) The parameter update is

$$\boldsymbol{\beta}^{(1)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} - 0.1 \begin{pmatrix} 0 \\ -1.5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0.15 \end{pmatrix}.$$

(e) At iteration 0, $J(\boldsymbol{\beta}^{(0)}) = -4\log(0.5) = 4\log 2 \approx 2.773$.
At iteration 1, the linear predictors are $z_i = 0.15x_i$, giving $\mathbf{z} = (-0.15, 0, 0.15, 0.30)^T$.
The predicted probabilities are

$$\mathbf{p} \approx (0.463, 0.500, 0.537, 0.574)^T.$$

The loss is

$$\begin{aligned} J(\boldsymbol{\beta}^{(1)}) &= -[0 \cdot \log(0.463) + 1 \cdot \log(0.537) + 0 \cdot \log(0.500) + 1 \cdot \log(0.500) \\ &\qquad + 1 \cdot \log(0.537) + 0 \cdot \log(0.463) + 1 \cdot \log(0.574) + 0 \cdot \log(0.426)] \\ &= -[\log(0.537) + \log(0.500) + \log(0.537) + \log(0.426)] \\ &\approx -[(-0.623) + (-0.693) + (-0.623) + (-0.853)] \\ &\approx 2.661. \qquad \text{(Note: computed correctly below.)} \end{aligned}$$

More carefully:

$$\begin{aligned} J(\boldsymbol{\beta}^{(1)}) &= -\big[\log(1 - 0.463) + \log(1 - 0.500) + \log(0.537) + \log(0.574)\big] \\ &= -\big[\log(0.537) + \log(0.500) + \log(0.537) + \log(0.574)\big] \\ &\approx 0.623 + 0.693 + 0.623 + 0.555 = 2.494. \end{aligned}$$

Since $2.494 < 2.773$, the loss decreased after one iteration.

Solution 4. Odds ratio interpretation

(a) The odds ratios are:

$$\begin{aligned} e^{\hat\beta_1} &= e^{0.035} \approx 1.036: && \text{each additional year of age multiplies the odds of diabetes by 1.036.} \\ e^{\hat\beta_2} &= e^{0.50} \approx 1.649: && \text{each unit increase in BMI multiplies the odds by 1.649.} \\ e^{\hat\beta_3} &= e^{-0.80} \approx 0.449: && \text{each additional hour of exercise per week multiplies the odds by 0.4} \end{aligned}$$

In words: age and BMI increase diabetes risk, while exercise decreases it.
BMI has the strongest effect per unit.

(b) The linear predictor is

$$\eta = -4.0+0.035(50)+0.50(30)+(-0.80)(3) = -4.0+1.75+15.0-2.4 = 10.35.$$

The predicted probability is $\hat{p} = \sigma(10.35) = 1/(1+e^{-10.35}) \approx 0.99997$, so the predicted class is $\hat{y} = 1$ (diabetes).

(c) We need $\eta < 0$ for $\hat{p} < 0.5$. Let the additional exercise hours be Δ. Then

$$10.35 + (-0.80)\Delta < 0 \implies \Delta > \frac{10.35}{0.80} = 12.94.$$

The patient would need to increase exercise by at least 12.94 hours per week (for a total of about 16 hours/week). This unrealistic result reflects the strong influence of the other predictors (particularly BMI) on the prediction.

Solution 5. Symmetry of the sigmoid

(a) We compute directly:

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

Meanwhile,

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

Multiplying numerator and denominator by e^{z}:

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

(b) Using $\sigma'(z) = \sigma(z)(1-\sigma(z))$:

$$\begin{aligned}\sigma'(-z) &= \sigma(-z)(1-\sigma(-z)) \\ &= (1-\sigma(z))\cdot\sigma(z) \quad \text{(by part (a))} \\ &= \sigma(z)(1-\sigma(z)) = \sigma'(z).\end{aligned}$$

(c) Since σ' is symmetric about $z=0$ and continuous, any extremum must occur at $z=0$ or come in pairs. We compute $\sigma'(0) = \sigma(0)(1-\sigma(0)) = 0.5\times 0.5 = 0.25$.

To show this is a global maximum, consider the function $f(s) = s(1-s)$ for $s \in (0,1)$. We have $f'(s) = 1-2s$, which vanishes at $s = 0.5$ and satisfies $f''(s) = -2 < 0$, confirming a strict maximum. Since σ is a strictly increasing bijection from $\mathbb{R}$ to $(0,1)$ and $\sigma(0) = 0.5$, the composition $\sigma'(z) = f(\sigma(z))$ attains its unique maximum at $z=0$ with value $\sigma'(0) = 0.25$.

Solution 6. Convexity of the per-sample loss

(a) Let $u = \beta x$ so that $p = \sigma(u)$. By the chain rule,

$$\frac{\mathrm{d}J}{\mathrm{d}\beta} = \frac{\mathrm{d}J}{\mathrm{d}p}\cdot\frac{\mathrm{d}p}{\mathrm{d}u}\cdot\frac{\mathrm{d}u}{\mathrm{d}\beta}.$$

From the lecture derivation:

$$\frac{\mathrm{d}J}{\mathrm{d}p} = \frac{p-y}{p(1-p)}, \qquad \frac{\mathrm{d}p}{\mathrm{d}u} = p(1-p), \qquad \frac{\mathrm{d}u}{\mathrm{d}\beta} = x.$$

Multiplying these three factors:

$$\frac{\mathrm{d}J}{\mathrm{d}\beta} = \frac{p-y}{p(1-p)} \cdot p(1-p) \cdot x = (p-y)x = (\sigma(\beta x) - y)x.$$

(b) Differentiating again, we use $\frac{\mathrm{d}p}{\mathrm{d}\beta} = p(1-p) \cdot x$:

$$\begin{aligned}\frac{\mathrm{d}^2 J}{\mathrm{d}\beta^2} &= \frac{\mathrm{d}}{\mathrm{d}\beta}\big[(\sigma(\beta x) - y)x\big] = x \cdot \frac{\mathrm{d}}{\mathrm{d}\beta}\sigma(\beta x) \\ &= x \cdot \sigma(\beta x)(1-\sigma(\beta x)) \cdot x = \sigma(\beta x)(1-\sigma(\beta x))\, x^2.\end{aligned}$$

(c) Since $\sigma(\beta x) \in (0,1)$, we have $\sigma(\beta x)(1-\sigma(\beta x)) > 0$, and $x^2 \geq 0$. Therefore $\frac{\mathrm{d}^2 J}{\mathrm{d}\beta^2} \geq 0$ for all β, proving convexity.

Strict convexity requires $\frac{\mathrm{d}^2 J}{\mathrm{d}\beta^2} > 0$, which holds if and only if $x \neq 0$. In a dataset with n observations, the total loss is strictly convex if at least one observation has $x_i \neq 0$.

Solution 7. Equivalence of MLE and BCE minimization

(a) Under the Bernoulli model $Y_i \mid \mathbf{x}_i \sim \text{Bernoulli}(\sigma(\boldsymbol{\beta}^T\mathbf{x}_i))$, the log-likelihood is

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n}\big[y_i \log \sigma(\boldsymbol{\beta}^T\mathbf{x}_i) + (1-y_i)\log(1-\sigma(\boldsymbol{\beta}^T\mathbf{x}_i))\big].$$

(b) The BCE loss is defined as $J(\boldsymbol{\beta}) = -\ell(\boldsymbol{\beta})$. Since negation reverses the ordering,

$$\arg\max_{\boldsymbol{\beta}} \ell(\boldsymbol{\beta}) = \arg\min_{\boldsymbol{\beta}}\big[-\ell(\boldsymbol{\beta})\big] = \arg\min_{\boldsymbol{\beta}} J(\boldsymbol{\beta}).$$

(c) We compute the gradient of ℓ using the chain rule. From part (a), and applying the same three-factor chain rule as in the lecture derivation for each summand:

$$\frac{\partial \ell}{\partial \beta_j} = \sum_{i=1}^{n}(y_i - p_i)x_{ij},$$

where $p_i = \sigma(\boldsymbol{\beta}^T\mathbf{x}_i)$. Note the sign: the numerator in $\frac{\mathrm{d}\ell_i}{\mathrm{d}p_i}$ is $y_i/p_i - (1-y_i)/(1-p_i) = (y_i - p_i)/[p_i(1-p_i)]$, which after cancellation with $p_i(1-p_i)$ from σ' gives $(y_i - p_i)x_{ij}$.

In matrix form, $\nabla\ell(\boldsymbol{\beta}) = \mathbf{X}^T(\mathbf{y} - \mathbf{p})$. Since $J = -\ell$, we have $\nabla J(\boldsymbol{\beta}) = -\nabla\ell(\boldsymbol{\beta}) = \mathbf{X}^T(\mathbf{p} - \mathbf{y})$, consistent with the lecture.

Solution 8. Exponential family verification

(a) We rewrite the Poisson PMF:

$$\frac{\lambda^k e^{-\lambda}}{k!} = \exp\big\{k\log\lambda - \lambda - \log(k!)\big\}.$$

Comparing with the exponential family form $\exp\{\frac{y\theta - b(\theta)}{\phi} + c(y,\phi)\}$:

$$\begin{aligned}
\theta &= \log\lambda, \\
b(\theta) &= e^{\theta} \quad (\text{since } \lambda = e^{\theta}), \\
\phi &= 1, \\
c(y,\phi) &= -\log(y!).
\end{aligned}$$

(b) We verify:

$$\begin{aligned}
b'(\theta) &= \frac{\mathrm{d}}{\mathrm{d}\theta}e^{\theta} = e^{\theta} = \lambda = \mathbb{E}[Y]. \\
b''(\theta) &= \frac{\mathrm{d}^2}{\mathrm{d}\theta^2}e^{\theta} = e^{\theta} = \lambda.
\end{aligned}$$

Since $\phi = 1$, we get $\mathrm{Var}(Y) = \phi \cdot b''(\theta) = \lambda$, which matches the well-known variance of the Poisson distribution.

(c) The canonical link is $g(\mu) = \theta$, where $\mu = \mathbb{E}[Y] = \lambda$ and $\theta = \log\lambda$. Therefore the canonical link is

$$g(\lambda) = \log\lambda,$$

the natural logarithm. This is the log link, which maps the positive mean $\lambda \in (0,\infty)$ to the entire real line $\mathbb{R}$, ensuring compatibility with the linear predictor $\eta = \boldsymbol{\beta}^T\mathbf{x} \in \mathbb{R}$.

Solution 9. Failure modes of OLS for classification

(a) The model predicts $\hat{y} = 0.4 + 0.15x$. We need $\hat{y} > 1$ or $\hat{y} < 0$:

$$\begin{aligned}
0.4 + 0.15x > 1 &\implies x > 4, \\
0.4 + 0.15x < 0 &\implies x < -2.667.
\end{aligned}$$

For $x > 4$ or $x < -8/3 \approx -2.667$, the model predicts values outside $[0,1]$, which cannot be interpreted as probabilities.

(b) Under the Bernoulli assumption, $\mathrm{Var}(Y_i \mid x_i) = p_i(1-p_i)$ where $p_i = 0.4 + 0.15x_i$:

$$\begin{aligned}
x = 0: &\quad p = 0.4, \quad \mathrm{Var} = 0.4 \times 0.6 = 0.24, \\
x = 2: &\quad p = 0.7, \quad \mathrm{Var} = 0.7 \times 0.3 = 0.21, \\
x = 4: &\quad p = 1.0, \quad \mathrm{Var} = 1.0 \times 0.0 = 0.00.
\end{aligned}$$

The variance changes from 0.24 to 0.21 to 0 across these values. The OLS assumption of constant variance (homoscedasticity) is violated; the variance is a quadratic function of the predicted mean.

(c) For a given x_i, the prediction $\hat{y}_i = \beta_0 + \beta_1 x_i$ is a fixed constant. The residual $\epsilon_i = y_i - \hat{y}_i$ can only take two values: $1 - \hat{y}_i$ (when $y_i = 1$) and $-\hat{y}_i$ (when $y_i = 0$). A random variable supported on exactly two points cannot follow a Gaussian distribution, which is a continuous distribution supported on all of $\mathbb{R}$. More formally, the Gaussian has a density with respect to Lebesgue measure, whereas a two-point distribution is discrete (supported on a set of measure zero). This violates the normality assumption required for the statistical inference machinery of OLS.

Solution 10. Decision boundary analysis

(a) The decision boundary is defined by $\hat{p} = 0.5$, which occurs when the linear predictor is zero:

$$-2.0 + 1.5\,x_1 - 0.01\,x_2 = 0 \implies x_2 = 150\,x_1 - 200.$$

This is a straight line in the (x_1, x_2)-plane with slope 150 and x_2-intercept -200. Points above the line (larger x_2) correspond to $\hat{p} < 0.5$ (not spam), while points below correspond to $\hat{p} > 0.5$ (spam), since $\hat{\beta}_2 < 0$.

(b) The linear predictor is $\eta = -2.0 + 1.5(3) - 0.01(100) = -2.0 + 4.5 - 1.0 = 1.5$. The predicted probability is $\hat{p} = \sigma(1.5) = 1/(1 + e^{-1.5}) \approx 0.818$. Since $0.818 > 0.5$, the predicted class is spam ($\hat{y} = 1$).

(c) The colleague's claim is misleading. The magnitude of a coefficient depends on the scale of the corresponding feature. Here x_2 (email length) takes values in the hundreds, while x_1 (exclamation marks) takes small integer values. The contribution of x_2 to the linear predictor is $\hat{\beta}_2 x_2 = -0.01 \times 100 = -1.0$, which is comparable to $\hat{\beta}_1 x_1 = 1.5 \times 3 = 4.5$ in terms of influence on η. Coefficient magnitudes are only directly comparable when the features are on the same scale (e.g., after standardization).

Solution 11. Comparing GLM components

(a) Traffic accident counts are non-negative integers ($Y_i \in \{0, 1, 2, \dots\}$), which matches the support of the Poisson distribution. Linear regression assumes $Y_i \in \mathbb{R}$ and can predict negative values, which are meaningless for counts. Furthermore, the mean number of accidents λ_i must be positive. The identity link used in OLS does not enforce this constraint, whereas the log link guarantees $\lambda_i = e^{\eta_i} > 0$ for any linear predictor $\eta_i \in \mathbb{R}$.

(b) The three GLM components are:

- *Random component:* $Y_i \sim \text{Poisson}(\lambda_i)$.
- *Systematic component:* $\eta_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}$.
- *Link function:* $g(\lambda_i) = \log \lambda_i$ (the canonical log link), so $\lambda_i = e^{\eta_i}$.

(c) The exponentiated coefficients give multiplicative effects on the mean count:

$$e^{\hat{\beta}_1} = e^{0.12} \approx 1.127: \text{ each additional thousand vehicles per day multiplies the expected accident count by 1.127 (a 12.7\% increase).}$$

$$e^{\hat{\beta}_2} = e^{0.30} \approx 1.350: \text{ each additional lane multiplies the expected accident count by 1.350 (a 35.0\% increase).}$$

(d) The linear predictor is $\eta = -0.50 + 0.12(10) + 0.30(4) = -0.50 + 1.20 + 1.20 = 1.90$. The predicted mean count is $\hat{\lambda} = e^{1.90} \approx 6.69$ accidents.

Solution 12. Gradient descent convergence analysis

(a) Convexity guarantees that any local minimum is also the global minimum, but it does not guarantee convergence for an arbitrary step size. If the learning rate η is too large, gradient descent can overshoot the minimum, jumping to the opposite side of the loss surface. On a convex function, the iterates then oscillate with increasing amplitude around the minimum rather than converging to it. Convergence of gradient descent on a convex function with Lipschitz-continuous gradient requires $\eta < 2/L$, where L is the Lipschitz constant of the gradient (the largest eigenvalue of the Hessian).

(b) Two modifications:

- *Reduce the learning rate.* A smaller η (e.g., 0.01 or 0.001) reduces the step size, preventing overshooting. This directly addresses the oscillation by ensuring the update stays within the neighborhood of convergence.
- *Use a learning rate schedule.* Start with a moderate η and decay it over iterations (e.g., $\eta_t = \eta_0/(1 + \alpha t)$). Early iterations benefit from larger steps for fast initial progress, while later iterations use smaller steps for fine convergence.

(c) A NumPy implementation:

```
import numpy as np

def logistic_gd_step(X, y, beta, eta):
    z = X @ beta
    p = 1.0 / (1.0 + np.exp(-z))
    loss = -np.sum(y * np.log(p) + (1 - y) * np.log(1 - p))
    grad = X.T @ (p - y)
    beta_new = beta - eta * grad
    return beta_new, loss
```

Here `X` is the $n \times (p + 1)$ design matrix (with intercept column), `y` is the n-vector of labels, `beta` is the $(p + 1)$-vector of parameters, and `eta` is the learning rate. The function returns the updated parameter vector and the loss at the current iterate.