Data Science for Mathematicians

Lesson 6: Logistic Regression and Generalized Linear Models

Department of Mathematics and Computer Science

Outline

Recap: The Linear Regression Model

So far we have modeled a continuous response $y \in \mathbb{R}$ via

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \qquad \epsilon_i \sim \mathcal{N}(0, \sigma^2).$$

Recap: The Linear Regression Model

So far we have modeled a continuous response $y \in \mathbb{R}$ via

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \qquad \epsilon_i \sim \mathcal{N}(0, \sigma^2).$$

Key assumptions:

- Response is continuous and unbounded: $Y_i \mid \mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\beta}^T \mathbf{x}_i, \sigma^2)$
- OLS solution: $\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$
- Loss function: mean squared error, minimized by gradient descent (Week 5)

Recap: The Linear Regression Model

So far we have modeled a continuous response $y \in \mathbb{R}$ via

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \qquad \epsilon_i \sim \mathcal{N}(0, \sigma^2).$$

Key assumptions:

- Response is continuous and unbounded: $Y_i \mid \mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\beta}^T \mathbf{x}_i, \sigma^2)$
- OLS solution: $\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$
- Loss function: mean squared error, minimized by gradient descent (Week 5)

Question: What happens when the response is not continuous?

The Classification Problem

In **binary classification**, the target is discrete: $y_i \in \{0, 1\}$.

Real-world examples:

- **Medical diagnosis:** malignant ($y = 1$) vs. benign ($y = 0$)
- **Spam detection:** spam ($y = 1$) vs. not spam ($y = 0$)
- **Fraud detection:** fraudulent ($y = 1$) vs. legitimate ($y = 0$)

The Classification Problem

In **binary classification**, the target is discrete: $y_i \in \{0, 1\}$.

Real-world examples:

- **Medical diagnosis:** malignant ($y = 1$) vs. benign ($y = 0$)
- **Spam detection:** spam ($y = 1$) vs. not spam ($y = 0$)
- **Fraud detection:** fraudulent ($y = 1$) vs. legitimate ($y = 0$)

Goal: Learn a function $\mathbf{x} \in \mathbb{R}^p \mapsto \hat{y} \in \{0, 1\}$.

Why Linear Regression Fails for Classification

Naive approach: fit $\hat{y} = \boldsymbol{\beta}^T \mathbf{x}$ with OLS, then threshold at 0.5.

Why Linear Regression Fails for Classification

Naive approach: fit $\hat{y} = \boldsymbol{\beta}^T \mathbf{x}$ with OLS, then threshold at 0.5.

Problem 1: Unbounded predictions.

- $\hat{y} = \boldsymbol{\beta}^T \mathbf{x} \in (-\infty, \infty)$, but we need $\mathbb{P}(Y = 1 \mid \mathbf{x}) \in [0, 1]$.
- Predictions like 1.5 or -0.3 have no probabilistic meaning.

Why Linear Regression Fails for Classification

Naive approach: fit $\hat{y} = \boldsymbol{\beta}^T \mathbf{x}$ with OLS, then threshold at 0.5.

Problem 1: Unbounded predictions.

- $\hat{y} = \boldsymbol{\beta}^T \mathbf{x} \in (-\infty, \infty)$, but we need $\mathbb{P}(Y = 1 \mid \mathbf{x}) \in [0, 1]$.
- Predictions like 1.5 or -0.3 have no probabilistic meaning.

Problem 2: Heteroscedasticity.

- $Y_i \sim \text{Bernoulli}(p_i) \implies \text{Var}(Y_i \mid \mathbf{x}_i) = p_i(1 - p_i)$.
- Variance depends on the mean—violates OLS homoscedasticity assumption.

Why Linear Regression Fails for Classification

Naive approach: fit $\hat{y} = \boldsymbol{\beta}^T \mathbf{x}$ with OLS, then threshold at 0.5.

Problem 1: Unbounded predictions.

- $\hat{y} = \boldsymbol{\beta}^T \mathbf{x} \in (-\infty, \infty)$, but we need $\mathbb{P}(Y = 1 \mid \mathbf{x}) \in [0, 1]$.
- Predictions like 1.5 or -0.3 have no probabilistic meaning.

Problem 2: Heteroscedasticity.

- $Y_i \sim \text{Bernoulli}(p_i) \implies \text{Var}(Y_i \mid \mathbf{x}_i) = p_i(1 - p_i)$.
- Variance depends on the mean—violates OLS homoscedasticity assumption.

Problem 3: Non-Gaussian errors.

- Error $\epsilon_i = y_i - \boldsymbol{\beta}^T \mathbf{x}_i$ takes only two values—cannot be Gaussian.

The Key Insight

These are not minor issues—they are **fundamental violations** of OLS assumptions.

Solution: Do not model y directly. Instead, model the **conditional probability**:

$$p(\mathbf{x}) \equiv \mathbb{P}(Y = 1 \mid X = \mathbf{x}).$$

The Key Insight

These are not minor issues—they are **fundamental violations** of OLS assumptions.

Solution: Do not model y directly. Instead, model the **conditional probability**:

$$p(\mathbf{x}) \equiv \mathbb{P}(Y = 1 \mid X = \mathbf{x}).$$

We need a function that:

- Takes the linear predictor $\eta = \boldsymbol{\beta}^T \mathbf{x} \in (-\infty, \infty)$
- Maps it to a valid probability $p(\mathbf{x}) \in [0, 1]$

This is precisely the role of **logistic regression**.

From Probability to Odds

We build the transformation in two steps.

Step 1: The **odds ratio** removes the upper bound.

$$\text{odds} = \frac{p}{1-p}, \qquad p \in (0,1) \implies \text{odds} \in (0, \infty).$$

From Probability to Odds

We build the transformation in two steps.

Step 1: The **odds ratio** removes the upper bound.

$$\text{odds} = \frac{p}{1-p}, \qquad p \in (0,1) \implies \text{odds} \in (0,\infty).$$

Example: Horse race

If $p = 0.8$, then odds $= \frac{0.8}{0.2} = 4$ ("4 to 1 in favor").

If $p = 0.2$, then odds $= \frac{0.2}{0.8} = 0.25$ ("4 to 1 against").

The Logit Function

Step 2: Take the logarithm to remove the lower bound.

Definition: Logit function

The logit function $\operatorname{logit}\colon (0,1) \to \mathbb{R}$ is defined by

$$\operatorname{logit}(p) = \log\left(\frac{p}{1-p}\right).$$

The Logit Function

Step 2: Take the logarithm to remove the lower bound.

Definition: Logit function

The logit function $\operatorname{logit}\colon (0,1) \to \mathbb{R}$ is defined by

$$\operatorname{logit}(p) = \log\left(\frac{p}{1-p}\right).$$

Key properties:

- Strictly increasing bijection $(0,1) \to \mathbb{R}$
- $\operatorname{logit}(1/2) = 0$ (maximum uncertainty maps to zero)
- $\operatorname{logit}(p) \to -\infty$ as $p \to 0^+$; $\operatorname{logit}(p) \to +\infty$ as $p \to 1^-$

Logit: Numerical Examples

Example

- $p = 0.9$: odds $= 9$, log-odds $= \log 9 \approx 2.197$
- $p = 0.1$: odds ≈ 0.111, log-odds ≈ -2.197
- $p = 0.5$: odds $= 1$, log-odds $= 0$

Logit: Numerical Examples

Example

- $p = 0.9$: odds $= 9$, log-odds $= \log 9 \approx 2.197$
- $p = 0.1$: odds ≈ 0.111, log-odds ≈ -2.197
- $p = 0.5$: odds $= 1$, log-odds $= 0$

The **symmetry** $\mathrm{logit}(p) = -\mathrm{logit}(1-p)$ reflects that $p = 0.5$ is the point of maximum uncertainty.

Summary of the two-step transformation:

$$\underbrace{p \in (0,1)}_{\text{probability}} \xrightarrow{\text{odds}} \underbrace{\frac{p}{1-p} \in (0,\infty)}_{\text{half-line}} \xrightarrow{\log} \underbrace{\log \frac{p}{1-p} \in \mathbb{R}}_{\text{real line}}.$$

The Core Modeling Assumption

We assume that the **log-odds is a linear function of the predictors**:

$$\boxed{\log\left(\frac{p(\mathbf{x})}{1-p(\mathbf{x})}\right) = \boldsymbol{\beta}^T \mathbf{x}.}$$

The Core Modeling Assumption

We assume that the **log-odds is a linear function of the predictors**:

$$\boxed{\log\left(\frac{p(\mathbf{x})}{1-p(\mathbf{x})}\right) = \boldsymbol{\beta}^T\mathbf{x}.}$$

This is a **linear model**—but not for the probability directly.

It is a linear model for the *logit-transformed* probability.

The Core Modeling Assumption

We assume that the **log-odds is a linear function of the predictors**:

$$\boxed{\log\left(\frac{p(\mathbf{x})}{1-p(\mathbf{x})}\right) = \boldsymbol{\beta}^T\mathbf{x}.}$$

This is a **linear model**—but not for the probability directly.

It is a linear model for the *logit-transformed* probability.

Next step: Invert the logit to express $p(\mathbf{x})$ explicitly.

Inverting the Logit: Deriving the Sigmoid

Let $\eta = \boldsymbol{\beta}^T \mathbf{x}$. Starting from $\log\left(\frac{p}{1-p}\right) = \eta$:

$$\begin{aligned}
\frac{p}{1-p} &= e^{\eta} \\
p &= e^{\eta}(1-p) = e^{\eta} - e^{\eta} p \\
p(1+e^{\eta}) &= e^{\eta} \\
p &= \frac{e^{\eta}}{1+e^{\eta}} = \frac{1}{1+e^{-\eta}}.
\end{aligned}$$

Substituting $\eta = \boldsymbol{\beta}^T \mathbf{x}$:

$$\boxed{p(\mathbf{x}) = \frac{1}{1+e^{-\boldsymbol{\beta}^T \mathbf{x}}}.}$$

Inverting the Logit: Deriving the Sigmoid

Let $\eta = \boldsymbol{\beta}^T \mathbf{x}$. Starting from $\log\left(\frac{p}{1-p}\right) = \eta$:

$$\frac{p}{1-p} = e^{\eta}$$

$$p(1 + e^{\eta}) = e^{\eta}$$

$$p = \frac{e^{\eta}}{1 + e^{\eta}} = \frac{1}{1 + e^{-\eta}}.$$

Substituting $\eta = \boldsymbol{\beta}^T \mathbf{x}$:

$$\boxed{p(\mathbf{x}) = \frac{1}{1 + e^{-\boldsymbol{\beta}^T \mathbf{x}}}.}$$

Inverting the Logit: Deriving the Sigmoid

Let $\eta = \boldsymbol{\beta}^T \mathbf{x}$. Starting from $\log\left(\frac{p}{1-p}\right) = \eta$:

$$\frac{p}{1-p} = e^{\eta}$$

$$p = \frac{e^{\eta}}{1+e^{\eta}} = \frac{1}{1+e^{-\eta}}.$$

Substituting $\eta = \boldsymbol{\beta}^T \mathbf{x}$:

$$\boxed{p(\mathbf{x}) = \frac{1}{1+e^{-\boldsymbol{\beta}^T \mathbf{x}}}.}$$

Inverting the Logit: Deriving the Sigmoid

Let $\eta = \boldsymbol{\beta}^T \mathbf{x}$. Starting from $\log\left(\frac{p}{1-p}\right) = \eta$:

$$
\begin{aligned}
\frac{p}{1-p} &= e^{\eta} \\
p &= e^{\eta}(1-p) = e^{\eta} - e^{\eta} p \\
p(1+e^{\eta}) &= e^{\eta} \\
p &= \frac{e^{\eta}}{1+e^{\eta}} = \frac{1}{1+e^{-\eta}}.
\end{aligned}
$$

Inverting the Logit: Deriving the Sigmoid

Let $\eta = \boldsymbol{\beta}^T \mathbf{x}$. Starting from $\log\left(\frac{p}{1-p}\right) = \eta$:

$$
\begin{aligned}
\frac{p}{1-p} &= e^{\eta} \\
p &= e^{\eta}(1-p) = e^{\eta} - e^{\eta} p \\
p(1+e^{\eta}) &= e^{\eta} \\
p &= \frac{e^{\eta}}{1+e^{\eta}} = \frac{1}{1+e^{-\eta}}.
\end{aligned}
$$

Substituting $\eta = \boldsymbol{\beta}^T \mathbf{x}$:

$$
\boxed{p(\mathbf{x}) = \frac{1}{1+e^{-\boldsymbol{\beta}^T \mathbf{x}}}.}
$$

The Sigmoid Function

Definition: Sigmoid function

The sigmoid function $\sigma\colon \mathbb{R} \to (0,1)$ is defined as

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

The logistic regression model: $p(\mathbf{x}) = \sigma(\boldsymbol{\beta}^T \mathbf{x})$.

The Sigmoid Function

Definition: Sigmoid function

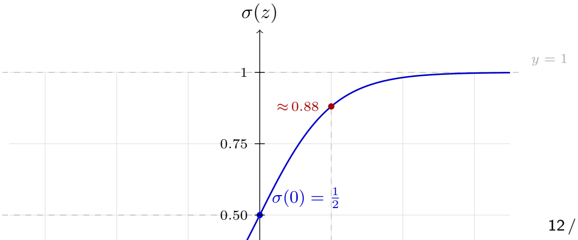
The sigmoid function $\sigma : \mathbb{R} \to (0,1)$ is defined as

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

The logistic regression model: $p(\mathbf{x}) = \sigma(\beta^T \mathbf{x})$.

Properties:

- $\sigma(z) \to 0$ as $z \to -\infty$
- $\sigma(0) = 1/2$
- $\sigma(z) \to 1$ as $z \to +\infty$
- Anti-symmetry: $\sigma(-z) = 1 - \sigma(z)$



Sigmoid: Numerical Examples

Example

- $\sigma(0) = \dfrac{1}{1 + e^0} = 0.5$ (maximum uncertainty)
- $\sigma(2) = \dfrac{1}{1 + e^{-2}} \approx 0.880$
- $\sigma(-2) \approx 0.119$

Sigmoid: Numerical Examples

Example

- $\sigma(0) = \dfrac{1}{1 + e^0} = 0.5$ (maximum uncertainty)
- $\sigma(2) = \dfrac{1}{1 + e^{-2}} \approx 0.880$
- $\sigma(-2) \approx 0.119$

In a logistic regression model, if $\boldsymbol{\beta}^T \mathbf{x} = 2$, the model assigns probability $\approx 88\%$ to class 1.

The bounded range $(0, 1)$ ensures outputs are always valid probabilities—resolving the central flaw of linear regression.

The Derivative of the Sigmoid

Theorem: Sigmoid derivative

$$\frac{\mathrm{d}}{\mathrm{d}z}\sigma(z) = \sigma(z)\big(1 - \sigma(z)\big).$$

The Derivative of the Sigmoid

Theorem: Sigmoid derivative

$$\frac{\mathrm{d}}{\mathrm{d}z}\sigma(z) = \sigma(z)\big(1 - \sigma(z)\big).$$

Proof sketch.

By the quotient rule with $u = 1$, $v = 1 + e^{-z}$:

$$\sigma'(z) = \frac{e^{-z}}{(1 + e^{-z})^2} = \frac{1}{1 + e^{-z}} - \left(\frac{1}{1 + e^{-z}}\right)^2 = \sigma(z) - [\sigma(z)]^2. \qquad \square$$

The Derivative of the Sigmoid

Theorem: Sigmoid derivative

$$\frac{\mathrm{d}}{\mathrm{d}z}\sigma(z) = \sigma(z)\big(1 - \sigma(z)\big).$$

Proof sketch.

By the quotient rule with $u = 1$, $v = 1 + e^{-z}$:

$$\sigma'(z) = \frac{e^{-z}}{(1+e^{-z})^2} = \frac{1}{1+e^{-z}} - \left(\frac{1}{1+e^{-z}}\right)^2 = \sigma(z) - [\sigma(z)]^2. \qquad \square$$

Why this matters: The derivative is expressed entirely in terms of $\sigma(z)$ itself. This will produce an elegant cancellation in the gradient of the loss function.

The Bernoulli Likelihood

Since $Y_i \sim \text{Bernoulli}(p_i)$ with $p_i = \sigma(\boldsymbol{\beta}^T \mathbf{x}_i)$, the probability of a single observation is

$$\mathbb{P}(Y_i = y_i \mid \mathbf{x}_i; \boldsymbol{\beta}) = p_i^{y_i}(1 - p_i)^{1-y_i}.$$

The Bernoulli Likelihood

Since $Y_i \sim \text{Bernoulli}(p_i)$ with $p_i = \sigma(\boldsymbol{\beta}^T \mathbf{x}_i)$, the probability of a single observation is

$$\mathbb{P}(Y_i = y_i \mid \mathbf{x}_i; \boldsymbol{\beta}) = p_i^{y_i}(1 - p_i)^{1-y_i}.$$

Under the i.i.d. assumption, the **likelihood** of the full dataset is

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} p_i^{y_i}(1 - p_i)^{1-y_i}.$$

The Bernoulli Likelihood

Since $Y_i \sim \text{Bernoulli}(p_i)$ with $p_i = \sigma(\boldsymbol{\beta}^T \mathbf{x}_i)$, the probability of a single observation is

$$\mathbb{P}(Y_i = y_i \mid \mathbf{x}_i; \boldsymbol{\beta}) = p_i^{y_i}(1 - p_i)^{1-y_i}.$$

Under the i.i.d. assumption, the **likelihood** of the full dataset is

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} p_i^{y_i}(1 - p_i)^{1-y_i}.$$

The **log-likelihood**:

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left[y_i \log p_i + (1 - y_i) \log(1 - p_i) \right].$$

Binary Cross-Entropy Loss

Negating the log-likelihood converts maximization to minimization.

Definition: Binary cross-entropy loss

The **binary cross-entropy (BCE)** loss is

$$J(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \big[y_i \log p_i + (1 - y_i) \log(1 - p_i) \big],$$

where $p_i = \sigma(\boldsymbol{\beta}^T \mathbf{x}_i)$.

Binary Cross-Entropy Loss

Negating the log-likelihood converts maximization to minimization.

Definition: Binary cross-entropy loss

The **binary cross-entropy (BCE)** loss is

$$J(\beta) = -\sum_{i=1}^{n} \left[y_i \log p_i + (1 - y_i) \log(1 - p_i) \right],$$

where $p_i = \sigma(\beta^T \mathbf{x}_i)$.

Example

True label $y_i = 1$:

- $p_i = 0.9 \implies J_i = -\log(0.9) \approx 0.105$ (small loss)
- $p_i = 0.1 \implies J_i = -\log(0.1) \approx 2.303$ (severe penalty)

Information-Theoretic Interpretation

The name *cross-entropy* comes from information theory.

For observation i with true label y_i:

- Empirical distribution P: deterministic $(P(Y = y_i) = 1)$
- Model distribution $Q = (1 - p_i,\ p_i)$

Information-Theoretic Interpretation

The name *cross-entropy* comes from information theory.

For observation i with true label y_i:

- Empirical distribution P: deterministic ($P(Y = y_i) = 1$)
- Model distribution $Q = (1 - p_i,\ p_i)$

The cross-entropy measures the coding cost when using Q to encode draws from P:

$$H(P, Q) = -\sum_{k \in \{0,1\}} P(Y = k) \log Q(Y = k) = J_i(\boldsymbol{\beta}).$$

Information-Theoretic Interpretation

The name *cross-entropy* comes from information theory.

For observation i with true label y_i:

- Empirical distribution P: deterministic ($P(Y = y_i) = 1$)
- Model distribution $Q = (1 - p_i,\ p_i)$

The cross-entropy measures the coding cost when using Q to encode draws from P:

$$H(P, Q) = -\sum_{k \in \{0,1\}} P(Y = k) \log Q(Y = k) = J_i(\beta).$$

Minimizing BCE $\iff$ minimizing the information-theoretic dissimilarity between the model's predictions and the observed labels.

Convexity of the Loss Function

Theorem

The binary cross-entropy loss $J(\beta)$ is **convex** with respect to β.

Convexity of the Loss Function

Theorem

The binary cross-entropy loss $J(\beta)$ is **convex** with respect to β.

Proof sketch.

The Hessian is

$$\nabla^2 J(\beta) = \sum_{i=1}^{n} \underbrace{\sigma_i(1-\sigma_i)}_{\geq 0} \mathbf{x}_i \mathbf{x}_i^T.$$

For any $\mathbf{v} \in \mathbb{R}^{p+1}$:

$$\mathbf{v}^T \nabla^2 J(\beta)\, \mathbf{v} = \sum_{i=1}^{n} \sigma_i(1-\sigma_i)(\mathbf{x}_i^T \mathbf{v})^2 \geq 0.$$

□

Convexity of the Loss Function

Theorem

The binary cross-entropy loss $J(\beta)$ is **convex** with respect to β.

Proof sketch.

The Hessian is

$$\nabla^2 J(\beta) = \sum_{i=1}^{n} \underbrace{\sigma_i(1-\sigma_i)}_{\geq 0} \mathbf{x}_i \mathbf{x}_i^T.$$

For any $\mathbf{v} \in \mathbb{R}^{p+1}$:

$$\mathbf{v}^T \nabla^2 J(\beta)\, \mathbf{v} = \sum_{i=1}^{n} \sigma_i(1-\sigma_i)(\mathbf{x}_i^T \mathbf{v})^2 \geq 0. \qquad \square$$

Implication: No local minima. Gradient descent converges to the **unique global minimum**.

The Optimization Problem

We seek the parameter vector that minimizes the convex BCE loss:

$$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} \left\{ -\sum_{i=1}^{n} \left[y_i \log p_i + (1 - y_i) \log(1 - p_i) \right] \right\}.$$

The Optimization Problem

We seek the parameter vector that minimizes the convex BCE loss:

$$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} \left\{ -\sum_{i=1}^{n} \left[y_i \log p_i + (1 - y_i) \log(1 - p_i) \right] \right\}.$$

Key difference from OLS: No closed-form solution exists.

We must use iterative optimization—gradient descent:

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - \eta \nabla J(\boldsymbol{\beta}^{(t)}).$$

The Optimization Problem

We seek the parameter vector that minimizes the convex BCE loss:

$$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} \left\{ -\sum_{i=1}^{n} \left[y_i \log p_i + (1 - y_i) \log(1 - p_i) \right] \right\}.$$

Key difference from OLS: No closed-form solution exists.

We must use iterative optimization—gradient descent:

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - \eta \, \nabla J(\boldsymbol{\beta}^{(t)}).$$

We need the gradient $\nabla J(\boldsymbol{\beta})$.

Deriving the Gradient: Chain Rule

For a single sample, apply the chain rule:

$$\frac{\partial J_i}{\partial \beta_j} = \frac{\partial J_i}{\partial p_i} \cdot \frac{\partial p_i}{\partial z_i} \cdot \frac{\partial z_i}{\partial \beta_j},$$

where $z_i = \boldsymbol{\beta}^T \mathbf{x}_i$ and $p_i = \sigma(z_i)$.

Deriving the Gradient: Chain Rule

For a single sample, apply the chain rule:

$$\frac{\partial J_i}{\partial \beta_j} = \frac{\partial J_i}{\partial p_i} \cdot \frac{\partial p_i}{\partial z_i} \cdot \frac{\partial z_i}{\partial \beta_j},$$

where $z_i = \boldsymbol{\beta}^T \mathbf{x}_i$ and $p_i = \sigma(z_i)$.

The three components:

1. $\dfrac{\partial J_i}{\partial p_i} = \dfrac{p_i - y_i}{p_i(1 - p_i)}$

Deriving the Gradient: Chain Rule

For a single sample, apply the chain rule:

$$\frac{\partial J_i}{\partial \beta_j} = \frac{\partial J_i}{\partial p_i} \cdot \frac{\partial p_i}{\partial z_i} \cdot \frac{\partial z_i}{\partial \beta_j},$$

where $z_i = \boldsymbol{\beta}^T \mathbf{x}_i$ and $p_i = \sigma(z_i)$.

The three components:

1. $\dfrac{\partial J_i}{\partial p_i} = \dfrac{p_i - y_i}{p_i(1 - p_i)}$
2. $\dfrac{\partial p_i}{\partial z_i} = p_i(1 - p_i)$ (sigmoid derivative)

Deriving the Gradient: Chain Rule

For a single sample, apply the chain rule:

$$\frac{\partial J_i}{\partial \beta_j} = \frac{\partial J_i}{\partial p_i} \cdot \frac{\partial p_i}{\partial z_i} \cdot \frac{\partial z_i}{\partial \beta_j},$$

where $z_i = \boldsymbol{\beta}^T \mathbf{x}_i$ and $p_i = \sigma(z_i)$.

The three components:

1. $\dfrac{\partial J_i}{\partial p_i} = \dfrac{p_i - y_i}{p_i(1 - p_i)}$
2. $\dfrac{\partial p_i}{\partial z_i} = p_i(1 - p_i) \quad$ (sigmoid derivative)
3. $\dfrac{\partial z_i}{\partial \beta_j} = x_{ij}$

The Elegant Cancellation

Multiplying the three terms:

$$\frac{\partial J_i}{\partial \beta_j} = \frac{p_i - y_i}{\cancel{p_i(1 - p_i)}} \cdot \cancel{p_i(1 - p_i)} \cdot x_{ij}.$$

The Elegant Cancellation

Multiplying the three terms:

$$\frac{\partial J_i}{\partial \beta_j} = \frac{p_i - y_i}{\cancel{p_i(1 - p_i)}} \cdot \cancel{p_i(1 - p_i)} \cdot x_{ij}.$$

The $p_i(1 - p_i)$ terms cancel exactly, giving the remarkably simple result:

$$\boxed{\frac{\partial J_i}{\partial \beta_j} = (p_i - y_i)\, x_{ij}.}$$

The Elegant Cancellation

Multiplying the three terms:

$$\frac{\partial J_i}{\partial \beta_j} = \frac{p_i - y_i}{\cancel{p_i(1 - p_i)}} \cdot \cancel{p_i(1 - p_i)} \cdot x_{ij}.$$

The $p_i(1 - p_i)$ terms cancel exactly, giving the remarkably simple result:

$$\boxed{\frac{\partial J_i}{\partial \beta_j} = (p_i - y_i)\, x_{ij}.}$$

Summing over all observations:

$$\frac{\partial J}{\partial \beta_j} = \sum_{i=1}^{n} \left(\sigma(\boldsymbol{\beta}^T \mathbf{x}_i) - y_i \right) x_{ij}.$$

The Vectorized Gradient

The full gradient vector in compact form:

$$\boxed{\nabla J(\beta) = \mathbf{X}^T(\mathbf{p} - \mathbf{y}),}$$

where $\mathbf{p} = (\sigma(\beta^T\mathbf{x}_1), \ldots, \sigma(\beta^T\mathbf{x}_n))^T$.

The Vectorized Gradient

The full gradient vector in compact form:

$$\boxed{\nabla J(\beta) = \mathbf{X}^T(\mathbf{p} - \mathbf{y}),}$$

where $\mathbf{p} = (\sigma(\beta^T\mathbf{x}_1), \ldots, \sigma(\beta^T\mathbf{x}_n))^T$.

Comparison with OLS gradient:

$$\nabla J_{\text{OLS}}(\beta) = \frac{2}{n}\mathbf{X}^T(\mathbf{X}\beta - \mathbf{y}).$$

The Vectorized Gradient

The full gradient vector in compact form:

$$\boxed{\nabla J(\boldsymbol{\beta}) = \mathbf{X}^T(\mathbf{p} - \mathbf{y}),}$$

where $\mathbf{p} = (\sigma(\boldsymbol{\beta}^T\mathbf{x}_1), \dots, \sigma(\boldsymbol{\beta}^T\mathbf{x}_n))^T$.

Comparison with OLS gradient:

$$\nabla J_{\text{OLS}}(\boldsymbol{\beta}) = \frac{2}{n}\mathbf{X}^T(\mathbf{X}\boldsymbol{\beta} - \mathbf{y}).$$

Both share the structure $\mathbf{X}^T(\text{prediction} - \text{truth})$.

This is not a coincidence—it arises from using a canonical link function with an exponential family distribution.

Gradient Descent Update Rule

The complete gradient descent algorithm for logistic regression:

1. Initialize $\beta^{(0)}$ (e.g., $\mathbf{0}$). Choose learning rate η.

Gradient Descent Update Rule

The complete gradient descent algorithm for logistic regression:

1. Initialize $\beta^{(0)}$ (e.g., $\mathbf{0}$). Choose learning rate η.
2. **Repeat** until convergence:
   1. Compute predictions: $\mathbf{p} = \sigma(\mathbf{X}\beta^{(t)})$
   2. Compute gradient: $\nabla J = \mathbf{X}^T(\mathbf{p} - \mathbf{y})$
   3. Update parameters: $\beta^{(t+1)} = \beta^{(t)} - \eta \nabla J$

Gradient Descent Update Rule

The complete gradient descent algorithm for logistic regression:

1. Initialize $\boldsymbol{\beta}^{(0)}$ (e.g., $\mathbf{0}$). Choose learning rate η.
2. **Repeat** until convergence:
   1. Compute predictions: $\mathbf{p} = \sigma(\mathbf{X}\boldsymbol{\beta}^{(t)})$
   2. Compute gradient: $\nabla J = \mathbf{X}^T(\mathbf{p} - \mathbf{y})$
   3. Update parameters: $\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - \eta \nabla J$

Convergence is guaranteed by the convexity of $J(\boldsymbol{\beta})$, provided η is sufficiently small.

Example 1: Medical Screening (Setup)

Example

Single predictor x (standardized biomarker), binary outcome y (disease).

i	1	2	3	4	5	6
x_i	-2	-1	0	1	2	3
y_i	0	0	0	1	1	1

Model: $p(x) = \sigma(\beta_0 + \beta_1 x)$. Initialize $\beta^{(0)} = (0, 0)^T$, $\eta = 0.1$.

Example 1: Iteration 0

With $\beta^{(0)} = (0, 0)^T$: every $z_i = 0$, so $p_i = 0.5$ for all i.

Residual vector: $\mathbf{p} - \mathbf{y} = (0.5,\ 0.5,\ 0.5,\ -0.5,\ -0.5,\ -0.5)^T$.

Example 1: Iteration 0

With $\beta^{(0)} = (0, 0)^T$: every $z_i = 0$, so $p_i = 0.5$ for all i.

Residual vector: $\mathbf{p} - \mathbf{y} = (0.5,\ 0.5,\ 0.5,\ -0.5,\ -0.5,\ -0.5)^T$.

Gradient:

$$\nabla J = \mathbf{X}^T(\mathbf{p} - \mathbf{y}) = \begin{pmatrix} 0 \\ -4.5 \end{pmatrix}.$$

Example 1: Iteration 0

With $\beta^{(0)} = (0, 0)^T$: every $z_i = 0$, so $p_i = 0.5$ for all i.

Residual vector: $\mathbf{p} - \mathbf{y} = (0.5,\ 0.5,\ 0.5,\ -0.5,\ -0.5,\ -0.5)^T$.

Gradient:

$$\nabla J = \mathbf{X}^T(\mathbf{p} - \mathbf{y}) = \begin{pmatrix} 0 \\ -4.5 \end{pmatrix}.$$

Update:

$$\beta^{(1)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} - 0.1 \begin{pmatrix} 0 \\ -4.5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0.45 \end{pmatrix}.$$

The zero first component reflects the balanced dataset; β_1 increases as expected.

Example 1: Iteration 1

With $\boldsymbol{\beta}^{(1)} = (0, 0.45)^T$, the linear predictors are $z_i = 0.45x_i$:

$$\mathbf{p} \approx (0.289,\ 0.389,\ 0.500,\ 0.611,\ 0.711,\ 0.794)^T.$$

Example 1: Iteration 1

With $\beta^{(1)} = (0, 0.45)^T$, the linear predictors are $z_i = 0.45x_i$:

$$\mathbf{p} \approx (0.289,\ 0.389,\ 0.500,\ 0.611,\ 0.711,\ 0.794)^T.$$

Gradient:

$$\nabla J \approx \begin{pmatrix} 0.294 \\ -2.334 \end{pmatrix}.$$

Example 1: Iteration 1

With $\boldsymbol{\beta}^{(1)} = (0, 0.45)^T$, the linear predictors are $z_i = 0.45x_i$:

$$\mathbf{p} \approx (0.289,\ 0.389,\ 0.500,\ 0.611,\ 0.711,\ 0.794)^T.$$

Gradient:

$$\nabla J \approx \begin{pmatrix} 0.294 \\ -2.334 \end{pmatrix}.$$

Update:

$$\boldsymbol{\beta}^{(2)} = \begin{pmatrix} 0 \\ 0.45 \end{pmatrix} - 0.1 \begin{pmatrix} 0.294 \\ -2.334 \end{pmatrix} = \begin{pmatrix} -0.029 \\ 0.683 \end{pmatrix}.$$

β_1 continues to grow, sharpening discrimination between classes.

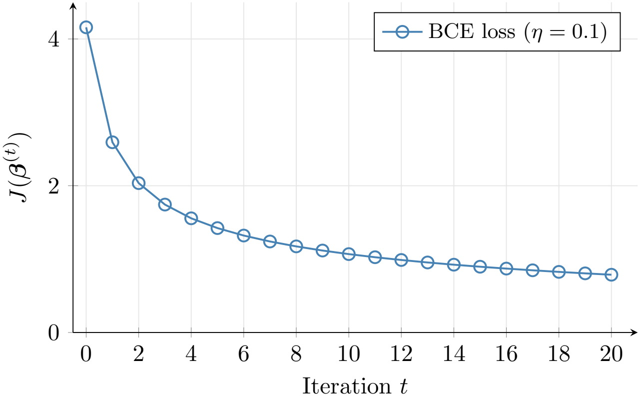
Example 1: Convergence

t	0	1	2	3	4	5	10
$J(\beta^{(t)})$	4.159	2.592	2.035	1.743	1.556	1.423	1.068

Example 1: Convergence

t	0	1	2	3	4	5	10
$J(\beta^{(t)})$	4.159	2.592	2.035	1.743	1.556	1.423	1.068

Monotone decrease, consistent with convexity. Parameters stabilize near $\hat{\beta} \approx (-0.41,\ 1.55)^T$.



Example 1: Classification Results

Fitted model: $\hat{p}(x) = \sigma(-0.41 + 1.55\,x)$.

Decision boundary at $x^* = -\beta_0/\beta_1 \approx 0.26$.

Example 1: Classification Results

Fitted model: $\hat{p}(x) = \sigma(-0.41 + 1.55\,x)$.

Decision boundary at $x^* = -\beta_0/\beta_1 \approx 0.26$.

i	1	2	3	4	5	6
x_i	-2	-1	0	1	2	3
y_i	0	0	0	1	1	1
$\hat{p}(x_i)$	0.03	0.12	0.40	0.76	0.94	0.99
$\hat{y}_i$	0	0	0	1	1	1

Example 1: Classification Results

Fitted model: $\hat{p}(x) = \sigma(-0.41 + 1.55\,x)$.

Decision boundary at $x^* = -\beta_0/\beta_1 \approx 0.26$.

i	1	2	3	4	5	6
x_i	-2	-1	0	1	2	3
y_i	0	0	0	1	1	1
$\hat{p}(x_i)$	0.03	0.12	0.40	0.76	0.94	0.99
$\hat{y}_i$	0	0	0	1	1	1

Perfect classification: Accuracy $= 1.0$, Precision $= 1.0$, Recall $= 1.0$, $F_1 = 1.0$.

Example 2: Loan Approval (Setup)

Example

Two predictors: income (x_1) and credit history (x_2). Outcome: approval (y).

i	1	2	3	4	5	6
x_{1i}	-2	-1	0	1	0	2
x_{2i}	-1	-2	-1	1	2	1
y_i	0	0	0	1	1	1

Model: $p(\mathbf{x}) = \sigma(\beta_0 + \beta_1 x_1 + \beta_2 x_2)$. Initialize $\boldsymbol{\beta}^{(0)} = \mathbf{0}$, $\eta = 0.1$.

Example 2: First Two Iterations

Iteration 0 ($\beta^{(0)} = \mathbf{0}$, all $p_i = 0.5$):

$$\nabla J = \begin{pmatrix} 0 \\ -3.0 \\ -4.0 \end{pmatrix}, \qquad \beta^{(1)} = \begin{pmatrix} 0 \\ 0.30 \\ 0.40 \end{pmatrix}.$$

Example 2: First Two Iterations

Iteration 0 ($\beta^{(0)} = \mathbf{0}$, all $p_i = 0.5$):

$$\nabla J = \begin{pmatrix} 0 \\ -3.0 \\ -4.0 \end{pmatrix}, \qquad \beta^{(1)} = \begin{pmatrix} 0 \\ 0.30 \\ 0.40 \end{pmatrix}.$$

Iteration 1 ($\beta^{(1)} = (0, 0.30, 0.40)^T$):

$$\nabla J \approx \begin{pmatrix} 0.009 \\ -1.657 \\ -2.391 \end{pmatrix}, \qquad \beta^{(2)} = \begin{pmatrix} -0.001 \\ 0.466 \\ 0.639 \end{pmatrix}.$$

Example 2: First Two Iterations

Iteration 0 ($\boldsymbol{\beta}^{(0)} = \mathbf{0}$, all $p_i = 0.5$):

$$\nabla J = \begin{pmatrix} 0 \\ -3.0 \\ -4.0 \end{pmatrix}, \qquad \boldsymbol{\beta}^{(1)} = \begin{pmatrix} 0 \\ 0.30 \\ 0.40 \end{pmatrix}.$$

Iteration 1 ($\boldsymbol{\beta}^{(1)} = (0, 0.30, 0.40)^T$):

$$\nabla J \approx \begin{pmatrix} 0.009 \\ -1.657 \\ -2.391 \end{pmatrix}, \qquad \boldsymbol{\beta}^{(2)} = \begin{pmatrix} -0.001 \\ 0.466 \\ 0.639 \end{pmatrix}.$$

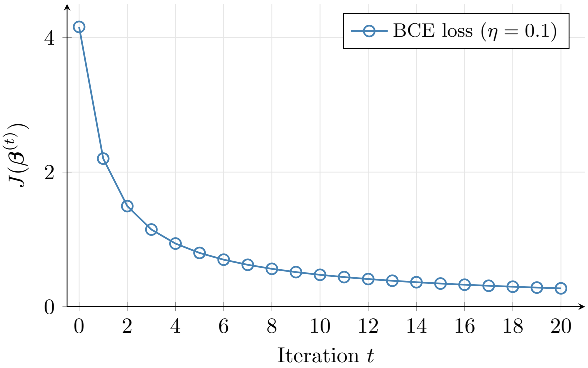
Both β_1 and β_2 grow, sharpening discrimination. The larger gradient component for x_2 suggests credit history has stronger influence.

Example 2: Convergence

t	0	1	2	3	4	5	10
$J(\beta^{(t)})$	4.159	2.201	1.495	1.146	0.938	0.799	0.473

Example 2: Convergence

t	0	1	2	3	4	5	10
$J(\beta^{(t)})$	4.159	2.201	1.495	1.146	0.938	0.799	0.473



Parameters stabilize near $\hat{\beta} \approx (-0.10,\ 1.10,\ 1.80)^T$.

Example 2: Coefficient Interpretation

Fitted model: $\hat{p}(\mathbf{x}) = \sigma(-0.10 + 1.10\,x_1 + 1.80\,x_2)$.

Odds ratio interpretation:

- Income: $e^{1.10} \approx 3.00$ — a one-unit increase in income multiplies the odds of approval by ≈ 3

Example 2: Coefficient Interpretation

Fitted model: $\hat{p}(\mathbf{x}) = \sigma(-0.10 + 1.10\,x_1 + 1.80\,x_2)$.

Odds ratio interpretation:

- Income: $e^{1.10} \approx 3.00$ — a one-unit increase in income multiplies the odds of approval by ≈ 3
- Credit history: $e^{1.80} \approx 6.05$ — a one-unit increase multiplies the odds by ≈ 6

Example 2: Coefficient Interpretation

Fitted model: $\hat{p}(\mathbf{x}) = \sigma(-0.10 + 1.10\, x_1 + 1.80\, x_2)$.

Odds ratio interpretation:

- Income: $e^{1.10} \approx 3.00$ — a one-unit increase in income multiplies the odds of approval by ≈ 3
- Credit history: $e^{1.80} \approx 6.05$ — a one-unit increase multiplies the odds by ≈ 6

Both positive $\implies$ higher income and longer credit history increase approval odds.

$|\hat{\beta}_2| > |\hat{\beta}_1| \implies$ credit history has a stronger influence.

Example 2: Test Set Evaluation

Applying the model to 8 new applicants:

i	1	2	3	4	5	6	7	8
$\hat{p}_i$	0.07	0.21	0.56	0.87	0.90	0.89	0.97	0.48
$\hat{y}_i$	0	0	1	1	1	1	1	0
y_i	0	0	0	1	1	1	1	1

Example 2: Test Set Evaluation

Applying the model to 8 new applicants:

i	1	2	3	4	5	6	7	8
$\hat{p}_i$	0.07	0.21	0.56	0.87	0.90	0.89	0.97	0.48
$\hat{y}_i$	0	0	1	1	1	1	1	0
y_i	0	0	0	1	1	1	1	1

Confusion matrix: $\text{TP} = 4$, $\text{TN} = 2$, $\text{FP} = 1$, $\text{FN} = 1$.

$$\text{Accuracy} = 0.75, \quad \text{Precision} = 0.80, \quad \text{Recall} = 0.80, \quad F_1 = 0.80.$$

Example 2: Test Set Evaluation

Applying the model to 8 new applicants:

i	1	2	3	4	5	6	7	8
$\hat{p}_i$	0.07	0.21	0.56	0.87	0.90	0.89	0.97	0.48
$\hat{y}_i$	0	0	1	1	1	1	1	0
y_i	0	0	0	1	1	1	1	1

Confusion matrix: $\text{TP} = 4$, $\text{TN} = 2$, $\text{FP} = 1$, $\text{FN} = 1$.

$$\text{Accuracy} = 0.75, \quad \text{Precision} = 0.80, \quad \text{Recall} = 0.80, \quad F_1 = 0.80.$$

Borderline cases ($\hat{p}_3 = 0.56$, $\hat{p}_8 = 0.48$) motivate threshold tuning.

Interpreting Logistic Regression Coefficients

In OLS: β_j is the additive change in y per unit increase in x_j.

In logistic regression: β_j is the additive change in the **log-odds**.

$$\log\left(\frac{p'}{1-p'}\right) = \log\left(\frac{p}{1-p}\right) + \beta_j.$$

Interpreting Logistic Regression Coefficients

In OLS: β_j is the additive change in y per unit increase in x_j.

In logistic regression: β_j is the additive change in the **log-odds**.

$$\log\left(\frac{p'}{1-p'}\right) = \log\left(\frac{p}{1-p}\right) + \beta_j.$$

Exponentiating gives a **multiplicative** interpretation:

$$\frac{\text{odds}'}{\text{odds}} = e^{\beta_j} \quad \text{(the \textbf{odds ratio})}.$$

Interpreting Logistic Regression Coefficients

In OLS: β_j is the additive change in y per unit increase in x_j.

In logistic regression: β_j is the additive change in the **log-odds**.

$$\log\left(\frac{p'}{1-p'}\right) = \log\left(\frac{p}{1-p}\right) + \beta_j.$$

Exponentiating gives a **multiplicative** interpretation:

$$\frac{\text{odds}'}{\text{odds}} = e^{\beta_j} \quad \text{(the \textbf{odds ratio})}.$$

- $\beta_j > 0 \implies e^{\beta_j} > 1$: increased odds
- $\beta_j < 0 \implies e^{\beta_j} < 1$: decreased odds
- $\beta_j = 0 \implies e^{\beta_j} = 1$: no effect

Odds Ratio: Clinical Example

Example: Heart disease risk

Logistic regression predicts 10-year heart disease risk.

Fitted coefficient for smoking: $\hat{\beta}_{\text{smoke}} = 0.693$.

Odds Ratio: Clinical Example

Example: Heart disease risk

Logistic regression predicts 10-year heart disease risk.

Fitted coefficient for smoking: $\hat{\beta}_{\text{smoke}} = 0.693$.

Odds ratio: $\text{OR} = e^{0.693} \approx 2.0$.

Interpretation: The odds of heart disease for a smoker are approximately **twice** those for a non-smoker, controlling for other variables.

Odds Ratio: Clinical Example

Example: Heart disease risk

Logistic regression predicts 10-year heart disease risk.

Fitted coefficient for smoking: $\hat{\beta}_{\text{smoke}} = 0.693$.

Odds ratio: $\text{OR} = e^{0.693} \approx 2.0$.

Interpretation: The odds of heart disease for a smoker are approximately **twice** those for a non-smoker, controlling for other variables.

The link function determines coefficient interpretation:

- Logit link $\rightarrow$ multiplicative effects on odds
- Identity link $\rightarrow$ additive effects on the mean

The Exponential Family

Definition: One-parameter exponential family

A distribution belongs to the **exponential family** if its density can be written as

$$f(y;\theta,\phi) = \exp\left\{\frac{y\theta - b(\theta)}{\phi} + c(y,\phi)\right\},$$

where θ is the natural parameter, $b(\theta)$ is the log-partition function, and $\phi > 0$ is the dispersion parameter.

The Exponential Family

Definition: One-parameter exponential family

A distribution belongs to the **exponential family** if its density can be written as

$$f(y;\theta,\phi) = \exp\left\{ \frac{y\theta - b(\theta)}{\phi} + c(y,\phi) \right\},$$

where θ is the natural parameter, $b(\theta)$ is the log-partition function, and $\phi > 0$ is the dispersion parameter.

Key properties from the log-partition function:

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

Members of the Exponential Family

Example: Bernoulli

$p^y(1-p)^{1-y} = \exp\big\{y \log \frac{p}{1-p} + \log(1-p)\big\}.$

Natural parameter: $\theta = \log\big(p/(1-p)\big)$ (the log-odds).
Log-partition: $b(\theta) = \log(1+e^\theta)$.

Members of the Exponential Family

Example: Bernoulli

$p^y(1-p)^{1-y} = \exp\big\{y \log \frac{p}{1-p} + \log(1-p)\big\}.$

Natural parameter: $\theta = \log\big(p/(1-p)\big)$ (the log-odds).
Log-partition: $b(\theta) = \log(1+e^\theta)$.

Example: Gaussian

Expanding the density: $\theta = \mu$, $b(\theta) = \theta^2/2$, $\phi = \sigma^2$.

Members of the Exponential Family

Example: Bernoulli

$p^y(1-p)^{1-y} = \exp\{y \log \frac{p}{1-p} + \log(1-p)\}.$

Natural parameter: $\theta = \log\big(p/(1-p)\big)$ (the log-odds).
Log-partition: $b(\theta) = \log(1+e^\theta)$.

Example: Gaussian

Expanding the density: $\theta = \mu$, $b(\theta) = \theta^2/2$, $\phi = \sigma^2$.

Both distributions we have used—Bernoulli for classification, Gaussian for regression—are members of the exponential family.

Generalized Linear Models: The Three Components

A **generalized linear model (GLM)** is defined by three components:

1. **Random component:** Y_i follows an exponential family distribution.

Generalized Linear Models: The Three Components

A **generalized linear model (GLM)** is defined by three components:

1. **Random component:** Y_i follows an exponential family distribution.
2. **Systematic component:** The linear predictor $\eta_i = \boldsymbol{\beta}^T \mathbf{x}_i$ (identical across all GLMs).

Generalized Linear Models: The Three Components

A **generalized linear model (GLM)** is defined by three components:

1. **Random component:** Y_i follows an exponential family distribution.
2. **Systematic component:** The linear predictor $\eta_i = \boldsymbol{\beta}^T \mathbf{x}_i$ (identical across all GLMs).
3. **Link function:** A monotonic, differentiable function g such that

$$g(\mu_i) = \eta_i, \qquad \mu_i = \mathbb{E}[Y_i].$$

Generalized Linear Models: The Three Components

A **generalized linear model (GLM)** is defined by three components:

1. **Random component:** Y_i follows an exponential family distribution.
2. **Systematic component:** The linear predictor $\eta_i = \boldsymbol{\beta}^T \mathbf{x}_i$ (identical across all GLMs).
3. **Link function:** A monotonic, differentiable function g such that

$$g(\mu_i) = \eta_i, \qquad \mu_i = \mathbb{E}[Y_i].$$

The link maps the range of μ_i to $\mathbb{R}$:

- Identity: $\mathbb{R} \to \mathbb{R}$ (Gaussian)
- Logit: $(0, 1) \to \mathbb{R}$ (Bernoulli)
- Log: $(0, \infty) \to \mathbb{R}$ (Poisson)

The Canonical Link

Each exponential family distribution has a distinguished **canonical link**:

$$g(\mu_i) = \theta_i, \qquad \text{where } \theta_i \text{ is the natural parameter.}$$

The Canonical Link

Each exponential family distribution has a distinguished **canonical link**:

$$g(\mu_i) = \theta_i, \qquad \text{where } \theta_i \text{ is the natural parameter.}$$

Since $\mu = b'(\theta)$, the canonical link is $g = (b')^{-1}$.

Why canonical links are special:

- The sufficient statistic $\mathbf{X}^T\mathbf{y}$ appears directly in the likelihood
- Score equations take a clean form
- This explains the elegant gradient $\nabla J = \mathbf{X}^T(\mathbf{p} - \mathbf{y})$

Three Instances of the GLM Framework

Component	Linear Regression	Logistic Regression	Poisson Regression
Distribution	$Y_i \sim \mathcal{N}(\mu_i, \sigma^2)$	$Y_i \sim \text{Bernoulli}(p_i)$	$Y_i \sim \text{Poisson}(\lambda_i)$
Mean range	$\mathbb{R}$	$(0, 1)$	$(0, \infty)$
Canonical link	$g(\mu) = \mu$	$g(p) = \log \frac{p}{1-p}$	$g(\lambda) = \log \lambda$
Loss	MSE	Cross-entropy	Poisson deviance

Three Instances of the GLM Framework

Component	**Linear Regression**	**Logistic Regression**	**Poisson Regression**
Distribution	$Y_i \sim \mathcal{N}(\mu_i, \sigma^2)$	$Y_i \sim \text{Bernoulli}(p_i)$	$Y_i \sim \text{Poisson}(\lambda_i)$
Mean range	$\mathbb{R}$	$(0, 1)$	$(0, \infty)$
Canonical link	$g(\mu) = \mu$	$g(p) = \log \frac{p}{1-p}$	$g(\lambda) = \log \lambda$
Loss	MSE	Cross-entropy	Poisson deviance

All three share the same systematic component $\eta_i = \boldsymbol{\beta}^T \mathbf{x}_i$.

Changing the distribution $\implies$ changing the link $\implies$ changing the loss.

Example: Poisson Regression

Example: Circuit board defects

Model: $Y_i \sim \text{Poisson}(\lambda_i)$ with log link:

$$\log \lambda_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}.$$

Example: Poisson Regression

Example: Circuit board defects

Model: $Y_i \sim \text{Poisson}(\lambda_i)$ with log link:

$$\log \lambda_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}.$$

Fitted: $\hat{\beta}_0 = -1.386$, $\hat{\beta}_1 = 0.030$, $\hat{\beta}_2 = 0.008$.

Interpretation: $e^{0.030} \approx 1.03$, so each additional cm^2 of board area multiplies the expected defect count by 1.03 (3% increase).

Example: Poisson Regression

Example: Circuit board defects

Model: $Y_i \sim \text{Poisson}(\lambda_i)$ with log link:

$$\log \lambda_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}.$$

Fitted: $\hat{\beta}_0 = -1.386$, $\hat{\beta}_1 = 0.030$, $\hat{\beta}_2 = 0.008$.

Interpretation: $e^{0.030} \approx 1.03$, so each additional cm^2 of board area multiplies the expected defect count by 1.03 (3% increase).

For $x_1 = 50$ cm^2, $x_2 = 250\,^{\circ}$C:

$$\hat{\lambda} = \exp(-1.386 + 0.030 \times 50 + 0.008 \times 250) \approx 8.28.$$

Key Formulas

Logistic regression model:

$$p(\mathbf{x}) = \sigma(\boldsymbol{\beta}^T \mathbf{x}) = \frac{1}{1 + e^{-\boldsymbol{\beta}^T \mathbf{x}}}.$$

Binary cross-entropy loss:

$$J(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \left[y_i \log p_i + (1 - y_i) \log(1 - p_i) \right].$$

Gradient:

$$\nabla J(\boldsymbol{\beta}) = \mathbf{X}^T (\mathbf{p} - \mathbf{y}).$$

Classification rule: Predict $\hat{y} = 1$ if $\sigma(\boldsymbol{\beta}^T \mathbf{x}) \geq \tau$.

Key Takeaways

- Linear regression fails for binary classification due to unbounded predictions, heteroscedasticity, and non-Gaussian errors.
- The sigmoid function $\sigma(z) = 1/(1 + e^{-z})$ maps the linear predictor to a valid probability, with the elegant derivative $\sigma'(z) = \sigma(z)(1 - \sigma(z))$.
- Maximum likelihood estimation on Bernoulli data yields the convex binary cross-entropy loss, guaranteeing gradient descent converges to the global optimum.
- The gradient $\nabla J = \mathbf{X}^T(\mathbf{p} - \mathbf{y})$ shares the same structure as the OLS gradient—a consequence of using the canonical link.
- Coefficients have an odds ratio interpretation: e^{β_j} is the multiplicative change in odds per unit increase in x_j.
- The GLM framework unifies linear, logistic, and Poisson regression through the choice of distribution and link function.