

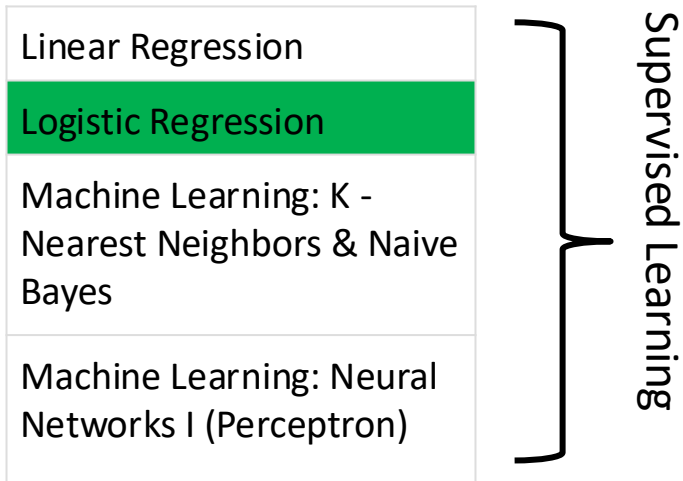


CS 540 Introduction to Artificial Intelligence
Logistic Regression, MLE, Regularization
University of Wisconsin-Madison
Fall 2026 Sections 1 & 2

Announcements

- HW4 due on **Friday Oct 9th at 11:59 PM**

- Class roadmap:



Outline

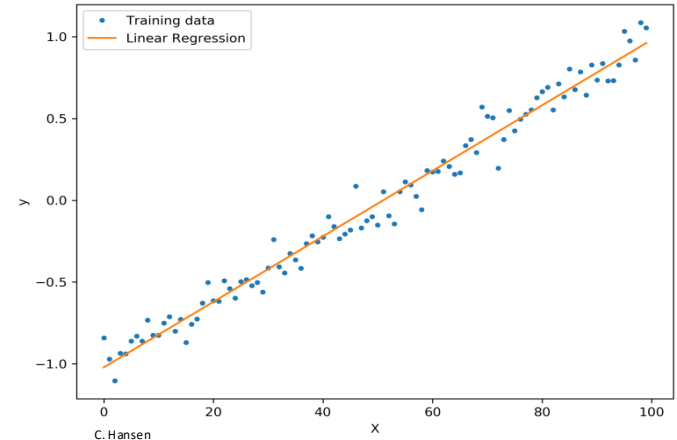
- Review of linear regression
- Maximum likelihood estimation
- Linear Classification and Logistic regression
- Regularization

Review: Linear Regression

Simplest form of regression: find a line that fits the data

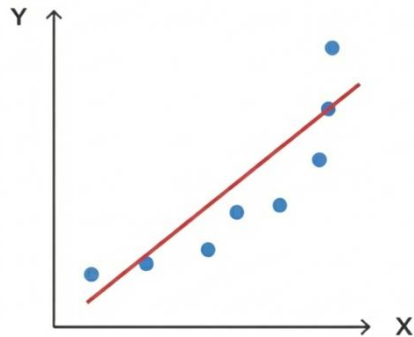
- When x is a scalar: $f_{(a,b)}(x) = ax + b$
- When x is a vector:

$$f(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_d x_d = \theta_0 + x^T \theta$$



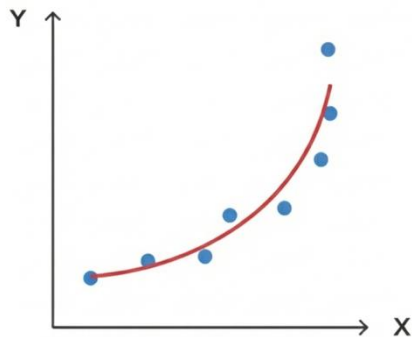
Polynomial Regression

- Linear regression tries to draw a straight line on your data.



- If data naturally curves, we may need to use polynomial regression.

$$f(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \dots + \theta_n x^n$$



Likelihood Function

- Captures the probability of seeing some data as a function of model parameters: $\mathcal{L}(\theta; X) = P_{\theta}(X)$
 - If data is iid, we have

$$\mathcal{L}(\theta; X) = \prod_j p_{\theta}(x_j)$$

- Often more convenient to work with the log likelihood
 - Log is a monotonic + strictly increasing function

Maximum Likelihood

- For some set of data, find the parameters that maximize the likelihood / log-likelihood

$$\hat{\theta} = \arg \max_{\theta} \mathcal{L}(\theta; X)$$

- Example:** suppose we have n samples from a Bernoulli distribution

$$p_{\theta}(x) = \begin{cases} \theta & \text{if } x = 1 \\ 1 - \theta & \text{if } x = 0 \end{cases}$$

Then, if there are h 1's: $\mathcal{L}(\theta; X) = \prod_{i=1}^n p_{\theta}(x) = \theta^h (1 - \theta)^{n-h}$

Maximum Likelihood: Example (cont'd)

- Want to maximize likelihood w.r.t. Θ

$$\mathcal{L}(\theta; X) = \prod_{i=1}^n p_{\theta}(x) = \theta^h (1 - \theta)^{n-h}$$

- Differentiate (use product rule) and set to 0. Get

$$\theta^{h-1} (1 - \theta)^{n-h-1} (h - n\theta) = 0$$

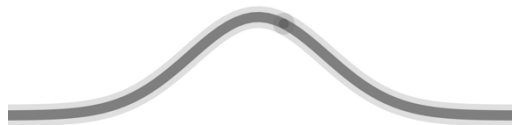
- So: ML estimate is $\hat{\theta} = \frac{h}{n}$

Maximum Likelihood Estimation: Gaussian Model

Observed some real-valued data

$$\{x_1, x_2, \dots, x_n\}$$

Model class: Gaussian model



$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

What is the MLE for the given data?

Maximum Likelihood Estimation: Gaussian Model

Observe some data: $x_1, x_2, \dots, x_n \in \mathbb{R}$

Likelihood:

$$L(\mu, \sigma^2) = \prod_{i=1}^n p(x_i \mid \mu, \sigma^2) = \prod_{i=1}^n \left[\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right) \right]$$

Fitting parameters is maximizing likelihood w.r.t to μ, σ^2
(maximize likelihood that data was generated by the model)

Maximum Likelihood Estimation: Gaussian Model

$$\text{MLE} \quad \arg \max_{\mu, \sigma^2} \prod_{i=1}^n p(x_i \mid \mu, \sigma^2)$$

Equivalently we can maximize the log likelihood:

$$\begin{aligned} \arg \max_{\mu, \sigma^2} \sum_{i=1}^n \log \left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp \left(-\frac{(x_i - \mu)^2}{2\sigma^2} \right) \right) \\ = n(-\log \sqrt{2\pi} - \log \sigma) - \sum_{i=1}^n \left(\frac{(x_i - \mu)^2}{2\sigma^2} \right) \end{aligned}$$

Maximum Likelihood Estimation: Gaussian Model

Setting the partial derivatives to zero:

$$\frac{\partial L}{\partial \mu} = -\frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu) = 0 \quad \Rightarrow \quad \mu = \frac{1}{n} \sum_{i=1}^n x_i$$

$$\frac{\partial L}{\partial \sigma} = -\frac{n}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^n (x_i - \mu)^2 = 0 \quad \Rightarrow \quad \sigma = \sqrt{\sum_{i=1}^n \left(\frac{(x_i - \mu)^2}{n} \right)}$$

MLE using Conditional Likelihoods

- Similar idea, but now using conditional probabilities:

$$\mathcal{L}(\theta; Y, X) = p_{\theta}(Y|X)$$

- If data is iid, we have

$$\mathcal{L}(\theta; Y, X) = \prod_j p_{\theta}(y_j|x_j)$$

- Shortly, we will apply this to linear classification: yields **logistic regression**.

Linear regression via MLE

- But first, let us revisit linear regression. Recall, we minimized the squared loss:

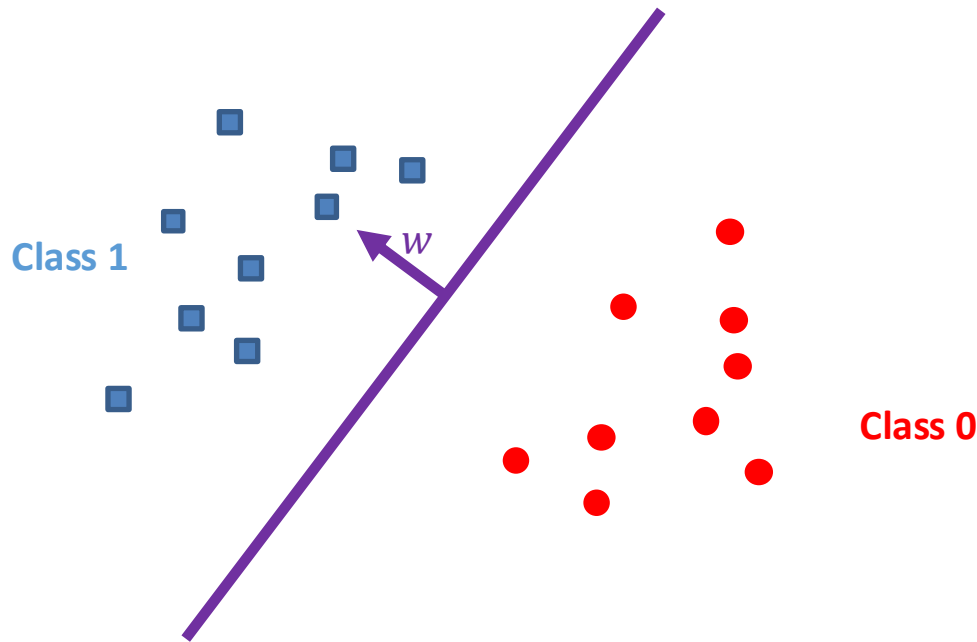
$$\min_{\theta} \frac{1}{n} \sum_{i=1}^n (\theta^{\top} x_i - y_i)^2$$

- We can show that this is also the MLE under a Gaussian conditional distribution (we will skip the details in class):

$$p_{\theta}(y|x) = \mathcal{N}(y; \theta^{\top} x, \sigma^2) \propto \exp \left(-\frac{1}{2\sigma^2} (y - \theta^{\top} x)^2 \right)$$

(Binary) Linear Classification

- In the last lecture, we discussed linear regression. What about classification with linear models?



Linear Classification: Attempt 1

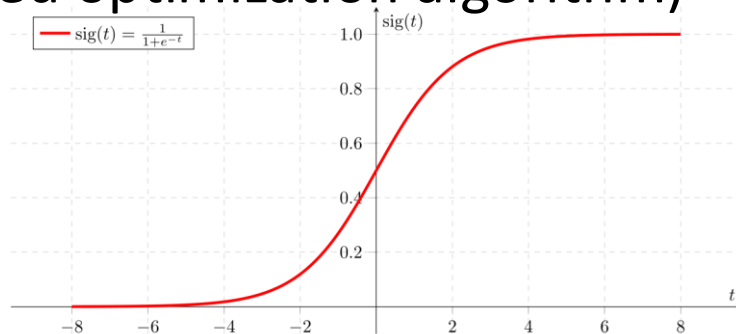
- Hyperplane: solutions to $\theta^T x = c$
 - note: d-1 dimensional
- So... try to use such hyperplanes as separators?
 - Model: $f_{\theta}(x) = \theta^T x$
 - Predict: $y=1$ if $\theta^T x > 0$, $y=0$ otherwise?
 - That is, $y = \text{step}(f_{\theta}(x))$
 - Train: 0/1 loss, or,

Difficult to optimize!!

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^n \mathbb{1} \{ \text{step}(f_{\theta}(x_i)) \neq y_i \}$$

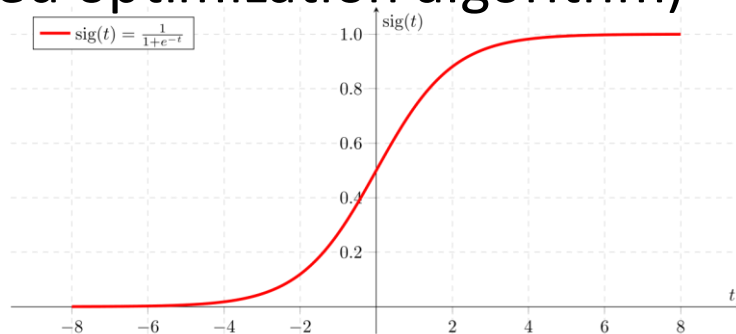
Linear Classification: Attempt 2

- Let us think probabilistically. Learn $P_{\theta}(y|x)$ instead
- How?
 - Specify the conditional distribution $P_{\theta}(y|x)$
 - Use maximum likelihood estimation (MLE) to derive a loss
 - Run gradient descent (or related optimization algorithm)



Linear Classification: Attempt 2

- Let us think probabilistically. Learn $P_{\theta}(y|x)$ instead
- How?
 - Specify the **conditional distribution** $P_{\theta}(y|x)$
 - Use **maximum likelihood estimation (MLE)** to derive a loss
 - Run gradient descent (or related optimization algorithm)

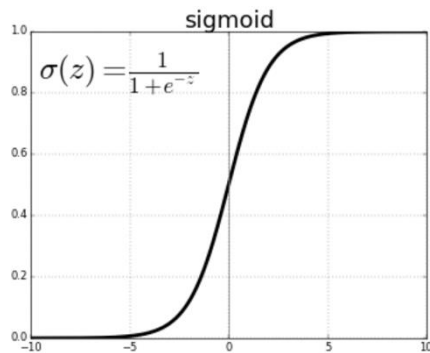


Logistic Regression: Conditional Distribution

- Notation:

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

↑
Sigmoid



- Conditional Distribution:

$$P_{\theta}(y = 1|x) = \sigma(\theta^T x) = \frac{1}{1 + \exp(-\theta^T x)}$$

Logistic Regression: Loss

- Conditional log likelihood:

$$\ell(\theta; X, Y) = \log P_{\theta}(X, Y) = \log \prod_{i=1}^n p_{\theta}(y_i | x_i)$$

- So MLE is:

$$\max_{\theta} \ell(\theta; X, Y) = \min_{\theta} - \sum_{i=1}^n \log p_{\theta}(y_i | x_i)$$

Or,
$$\min_{\theta} - \sum_{i:y_i=1} \log(\sigma(\theta^{\top} x_i)) - \sum_{i:y_i=0} \log(1 - \sigma(\theta^{\top} x_i))$$

Logistic Regression

$$\min_{\theta} - \sum_{i:y_i=1} \log(\sigma(\theta^\top x_i)) - \sum_{i:y_i=0} \log(1 - \sigma(\theta^\top x_i))$$

- No closed form solution, but can minimize using gradient descent

Logistic regression: Summary

- **Logistic regression = sigmoid conditional distribution + MLE**

- More precisely:

- Give training data iid from some distribution D ,

- **Train:**

$$\max_{\theta} \ell(\theta; X, Y) = \min_{\theta} - \sum_{i=1}^n \log p_{\theta}(y_i | x_i)$$

- **Test:** output label probabilities

$$P_{\theta}(y = 1 | x) = \sigma(\theta^T x) = \frac{1}{1 + \exp(-\theta^T x)}$$

Break and Quiz

Q1.1 The decision boundary of logistic regression is:

- A. Linear, because we classify based on $w^T x > 0$ or $w^T x < 0$.
- B. Nonlinear, because the sigmoid is a nonlinear function
- D. It depends on the data

Break and Quiz

Q1.1 The decision boundary of logistic regression is:

- A. Linear, because we classify based on $w^T x > 0$ or $w^T x < 0$.**
- B. Nonlinear, because the sigmoid is a nonlinear function
- D. It depends on the data

Break and Quiz

Q1.2 Assume the classifier predicts class 1 when $\theta^T x + \theta_0 > 0$ and predicts class 0 otherwise. Consider

$$\theta^T x + \theta_0 = x_1 - 2x_2 + 1.$$

Which class does the classifier assign to the point (1, 0)?

- A. Class 0
- B. Class 1
- C. It lies exactly on the decision boundary
- D. There is not enough information

Break and Quiz

Q1.2 Assume the classifier predicts class 1 when $\theta^T x + \theta_0 > 0$ and predicts class 0 otherwise. Consider

$$\theta^T x + \theta_0 = x_1 - 2x_2 + 1.$$

Which class does the classifier assign to the point (1, 0)?

A. Class 0

B. Class 1

C. It lies exactly on the decision boundary

D. There is not enough information

Break and Quiz

Q1.3 Assume the classifier predicts class 1 when $\theta^T \mathbf{x} + \theta_0 > 0$ and predicts class 0 otherwise. Suppose we have a classifier

$$\theta^T \mathbf{x} + \theta_0 = x_1 + x_2 - 2.$$

We change the coefficient of x_1 from 1 to 3:

$$\theta^T \mathbf{x} + \theta_0 = 3x_1 + x_2 - 2.$$

What is the most likely effect on the decision boundary?

- A. It shifts parallel to itself
- B. Its orientation changes
- C. It disappears
- D. Nothing changes

Break and Quiz

Q1.3 Assume the classifier predicts class 1 when $\theta^T \mathbf{x} + \theta_0 > 0$ and predicts class 0 otherwise. Suppose we have a classifier

$$\theta^T \mathbf{x} + \theta_0 = x_1 + x_2 - 2.$$

We change the coefficient of x_1 from 1 to 3:

$$\theta^T \mathbf{x} + \theta_0 = 3x_1 + x_2 - 2.$$

What is the most likely effect on the decision boundary?

- A. It shifts parallel to itself
- B. Its orientation changes**
- C. It disappears
- D. Nothing changes

Break and Quiz

Q1.4 Assume the classifier predicts class 1 when $\theta^T \mathbf{x} + \theta_0 > 0$ and predicts class 0 otherwise. Consider the two classifiers

$$\theta^T \mathbf{x} + \theta_0 = x_1 + x_2 - 1 \quad \text{and}$$

$$\theta^T \mathbf{x} + \theta_0 = 2x_1 + 2x_2 - 2.$$

How do their predictions compare?

- A. They always make the same predictions
- B. They always make opposite predictions
- C. They have different decision boundaries
- D. They are the same only when $x_1 = x_2$

Break and Quiz

Q1.4 Assume the classifier predicts class 1 when $\theta^T \mathbf{x} + \theta_0 > 0$ and predicts class 0 otherwise. Consider the two classifiers

$$\theta^T \mathbf{x} + \theta_0 = x_1 + x_2 - 1 \quad \text{and}$$

$$\theta^T \mathbf{x} + \theta_0 = 2x_1 + 2x_2 - 2.$$

How do their predictions compare?

- A. They always make the same predictions**
- B. They always make opposite predictions
- C. They have different decision boundaries
- D. They are the same only when $x_1 = x_2$

Regularization

- **Regularization:** a penalty on model complexity added to the training objective. Trades a little ***training error*** for better performance on unseen data (***test error***).
- Regularization in linear regression:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^n (y_i - \theta^{\top} x_i)^2 + \lambda \|\theta\|^2$$

This is called *Ridge regression*.

Regularization in linear regression (cont'd)

- Ridge regression $\min_{\theta} \frac{1}{n} \sum_{i=1}^n (y_i - \theta^\top x_i)^2 + \lambda \|\theta\|^2$
 - Closed form solution $\theta = (X^T X + \lambda n I)^{-1} X^T y$
 - Can also minimize using gradient descent
- Lasso regression $\min_{\theta} \frac{1}{n} \sum_{i=1}^n (y_i - \theta^\top x_i)^2 + \lambda \sum_{i=1}^d |\theta_i|$
 - No closed form solution. But can minimize via (sub-)gradient descent.

Regularization in logistic regression

- Recall objective in logistic regression:

$$\min_{\theta} - \sum_{i:y_i=1} \log(\sigma(\theta^\top x_i)) - \sum_{i:y_i=0} \log(1 - \sigma(\theta^\top x_i))$$

- We can add regularization terms $\|\theta\|^2$ or $\lambda \sum_{i=1}^d |\theta_i|$ to this objective.

Q2.1: Suppose you find that your linear regression model has much lower training error than testing error. In such situation which of the following options would you

- A. Introduce polynomial degree variables
- B. Use Ridge or Lasso regularization

Q2.1: Suppose you find that your linear regression model has much lower training error than testing error. In such situation which of the following options would you

A. Introduce polynomial degree variables

B. Use Ridge or Lasso regularization

Suggested Reading

- Linear regression, logistic regression, stochastic gradient descent by Prof. Zhu
<https://pages.cs.wisc.edu/~jerryzhu/cs540/handouts/regression.pdf>