Linear and Logistic Regression

CS 760@UW-Madison

Goals for the lecture

- understand the concepts
  - linear regression
  - closed form solution for linear regression
  - lasso
  - RMSE, MAE, and R-square
  - logistic regression for linear classification
  - gradient descent for logistic regression
  - multiclass logistic regression
  - cross entropy

Linear Regression

Linear regression

- Given training data $\{(x^{(i)}, y^{(i)}): 1 \leq i \leq m\}$ i.i.d. from distribution D
- Find $f_w(x) = w^T x$ that minimizes $\hat{L}(f_w) = \frac{1}{m}\sum_{i=1}^{m}\left(w^T x^{(i)} - y^{(i)}\right)^2$

l_2 loss; also called mean squared error

Hypothesis class $\mathcal{H}$

Linear regression: optimization

- Given training data $\{(x^{(i)}, y^{(i)}): 1 \leq i \leq m\}$ i.i.d. from distribution D
- Find $f_w(x) = w^T x$ that minimizes $\hat{L}(f_w) = \frac{1}{m}\sum_{i=1}^{m}\left(w^T x^{(i)} - y^{(i)}\right)^2$

- Let X be a matrix whose i-th row is $\left(x^{(i)}\right)^T$, y be the vector $\left(y^{(1)}, \dots, y^{(m)}\right)^T$

$$\hat{L}(f_w) = \frac{1}{m}\sum_{i=1}^{m}\left(w^T x^{(i)} - y^{(i)}\right)^2 = \frac{1}{m}\|Xw - y\|_2^2$$

Linear regression: optimization

- Set the gradient to 0 to get the minimizer

$$\nabla_w \hat{L}(f_w) = \nabla_w \frac{1}{m} \|Xw - y\|_2^2 = 0$$

$$\nabla_w [(Xw - y)^T (Xw - y)] = 0$$

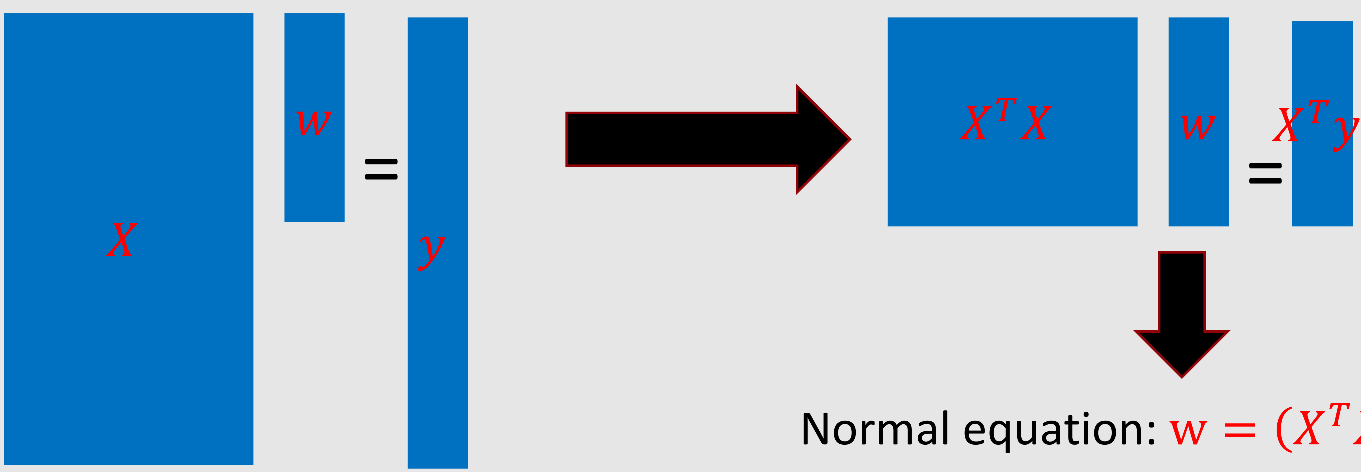
$$\nabla_w [\, w^T X^T X w - 2 w^T X^T y + y^T y] = 0$$

$$2X^T X w - 2X^T y = 0$$

$$\mathrm{w} = (X^T X)^{-1} X^T y$$

Linear regression: optimization

- Algebraic view of the minimizer
  - If X is invertible, just solve $Xw = y$ and get $w = X^{-1}y$
  - But typically X is a tall matrix



Normal equation: $w = (X^T X)^{-1} X^T y$

Linear regression with bias

- Given training data $\left\{\left(x^{(i)}, y^{(i)}\right): 1 \leq i \leq m\right\}$ i.i.d. from distribution D
- Find $f_{w,b}(x) = w^T x + b$ to minimize the loss

Bias term

- Reduce to the case without bias:
  - Let $w' = [w;\ b], x' = [x; 1]$
  - Then $f_{w,b}(x) = w^T x + b = (w')^T (x')$

Linear regression with lasso penalty

- Given training data $\{(x^{(i)}, y^{(i)}): 1 \le i \le m\}$ i.i.d. from distribution D
- Find $f_w(x) = w^T x$ that minimizes

$$\hat{L}(f_w) = \frac{1}{m}\sum_{i=1}^{m}\left(w^T x^{(i)} - y^{(i)}\right)^2 + \lambda|w|_1$$

lasso penalty: l_1 norm of the parameter, encourages sparsity

Evaluation Metrics

- Root mean squared error (RMSE)
- Mean absolute error (MAE) – average l_1 error
- R-square (R-squared)
- Historically all were computed on training data, and possibly adjusted after, but really should cross-validate

R-square

- Formulation 1:

$$R^2 = 1 - \frac{\sum_i (y_i - h(\vec{x_i}))^2}{\sum_i (y_i - \overline{y})^2}$$

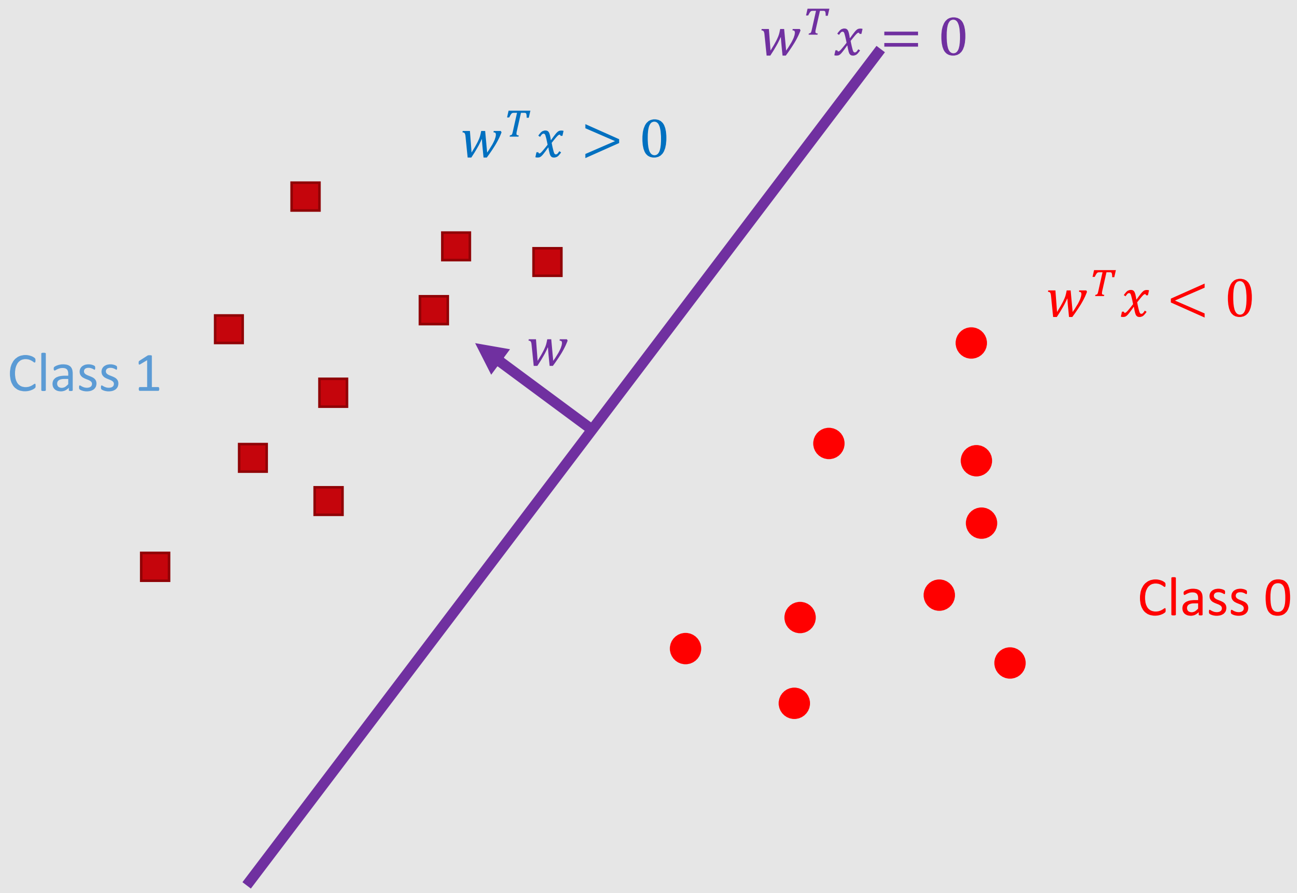
- Formulation 2: square of Pearson correlation coefficient r between the label and the prediction.

  Recall for x, y:

$$r = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_i (x_i - \overline{x})^2}\sqrt{\sum_i (y_i - \overline{y})^2}}$$

Linear Classification

Linear classification



Linear classification: natural attempt

- Given training data $\{(x^{(i)}, y^{(i)}): 1 \leq i \leq m\}$ i.i.d. from distribution D
- Hypothesis $f_w(x) = w^T x$
  - $y = 1$ if $w^T x > 0$
  - $y = 0$ if $w^T x < 0$
- Prediction: $y = \text{step}(f_w(x)) = \text{step}(w^T x)$

Linear model $\mathcal{H}$

Linear classification: natural attempt

- Given training data $\{(x^{(i)}, y^{(i)}): 1 \leq i \leq m\}$ i.i.d. from distribution D
- Find $f_w(x) = w^T x$ to minimize

$$\hat{L}(f_w) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{I}[\text{step}(w^T x^{(i)}) \neq y^{(i)}]$$

- Drawback: difficult to optimize
  - NP-hard in the worst case

0-1 loss

Linear classification: simple approach

- Given training data $\{(x^{(i)}, y^{(i)}): 1 \leq i \leq m\}$ i.i.d. from distribution D
- Find $f_w(x) = w^T x$ that minimizes $\hat{L}(f_w) = \frac{1}{m}\sum_{i=1}^{m}\left(w^T x^{(i)} - y^{(i)}\right)^2$

Reduce to linear regression; ignore the fact $y \in \{0,1\}$

Linear classification: simple approach

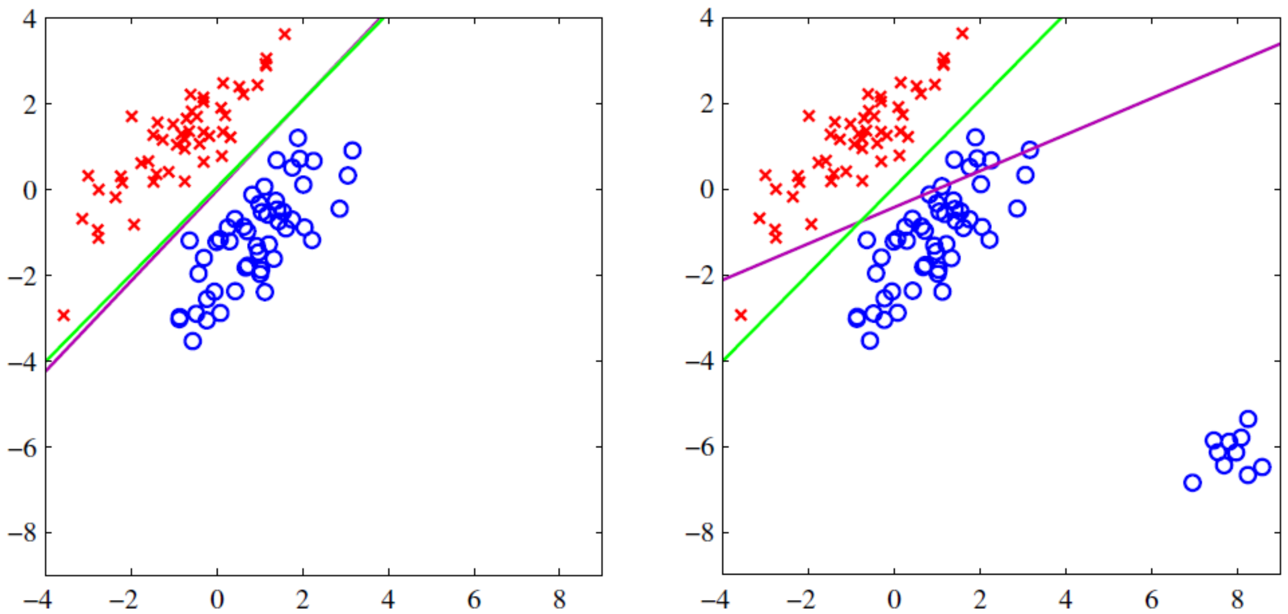
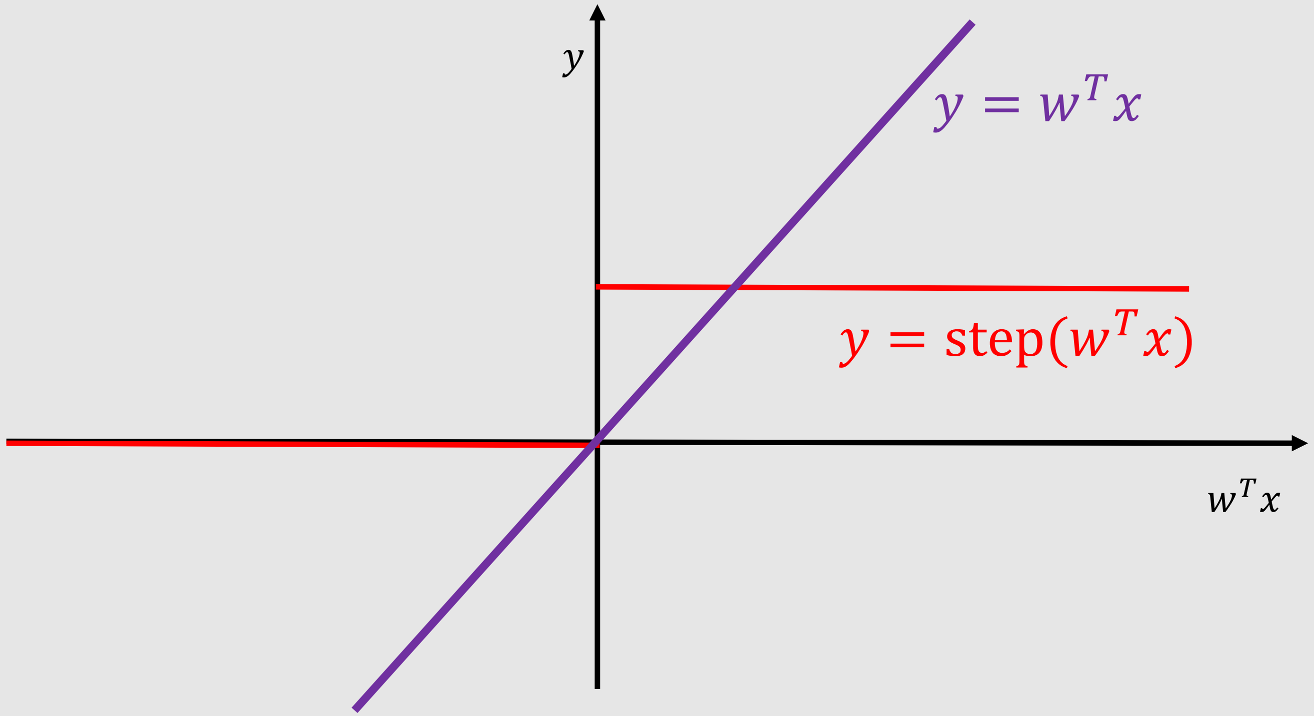


Figure 4.4 The left plot shows data from two classes, denoted by red crosses and blue circles, together with the decision boundary found by least squares (magenta curve) and also by the logistic regression model (green curve), which is discussed later in Section 4.3.2. The right-hand plot shows the corresponding results obtained when extra data points are added at the bottom left of the diagram, showing that least squares is highly sensitive to outliers, unlike logistic regression.

Drawback: not robust to "outliers"

Figure borrowed from *Pattern Recognition and Machine Learning*, Bishop

Compare the two



Between the two

- Prediction bounded in $[0,1]$
- Smooth

- Sigmoid: $\sigma(a) = \frac{1}{1+\exp(-a)}$

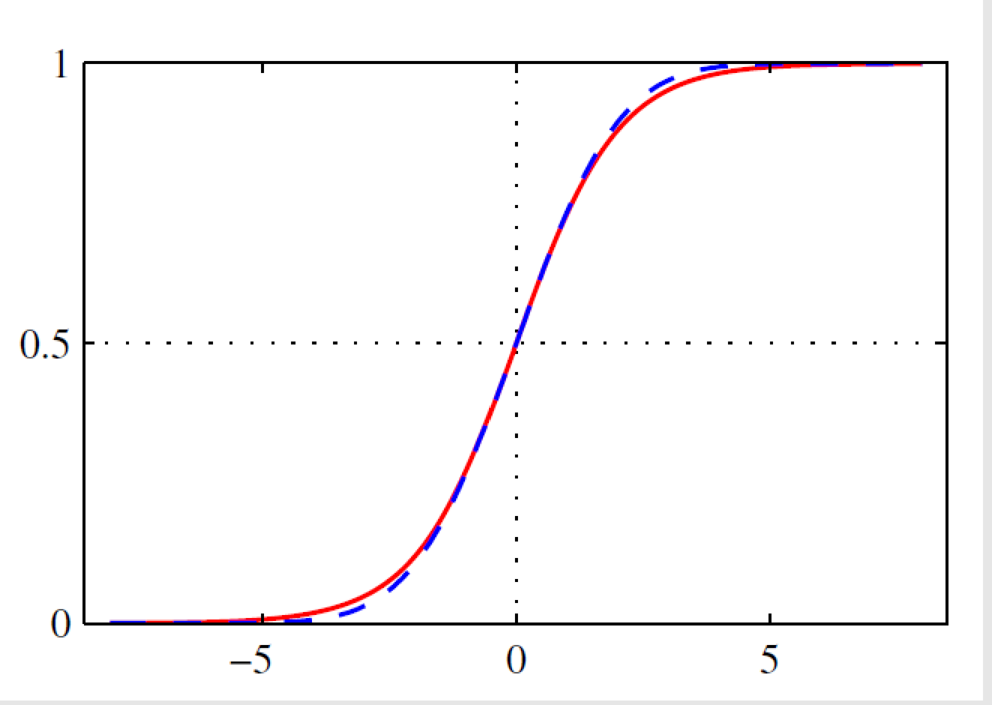


Figure borrowed from *Pattern Recognition and Machine Learning*, Bishop

Linear classification: sigmoid prediction

- Squash the output of the linear function

$$\text{Sigmoid}(w^T x) = \sigma(w^T x) = \frac{1}{1 + \exp(-w^T x)}$$

- Find w that minimizes $\hat{L}(f_w) = \frac{1}{m}\sum_{i=1}^{m}\left(\sigma(w^T x^{(i)}) - y^{(i)}\right)^2$

Linear Classification by Logistic Regression

Linear classification: logistic regression

- Squash the output of the linear function

$$\text{Sigmoid}(w^T x) = \sigma(w^T x) = \frac{1}{1 + \exp(-w^T x)}$$

- A better approach: Interpret as a probability

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

$$P_w(y = 0|x) = 1 - P_w(y = 1|x) = 1 - \sigma(w^T x)$$

Linear classification: logistic regression

- Find $f_w(x) = w^T x$ that minimizes $\hat{L}(f_w) = \frac{1}{m}\sum_{i=1}^{m}\left(w^T x^{(i)} - y^{(i)}\right)^2$
- Find w that minimizes

$$\hat{L}(w) = -\frac{1}{m}\sum_{i=1}^{m} \log P_w\left(y^{(i)} \middle| x^{(i)}\right)$$

$$\hat{L}(w) = -\frac{1}{m}\sum_{y^{(i)}=1} \log\sigma(w^T x^{(i)}) - \frac{1}{m}\sum_{y^{(i)}=0} \log[1 - \sigma\left(w^T x^{(i)}\right)]$$

Logistic regression:
MLE with sigmoid

Linear classification: logistic regression

- Given training data $\{(x^{(i)}, y^{(i)}) : 1 \leq i \leq m\}$ i.i.d. from distribution D
- Find w that minimizes

$$\hat{L}(w) = -\frac{1}{m} \sum_{y^{(i)}=1} \log \sigma(w^T x^{(i)}) - \frac{1}{m} \sum_{y^{(i)}=0} \log[1 - \sigma(w^T x^{(i)})]$$

No close form solution;
Need to use gradient descent

Properties of sigmoid function

- Bounded

$$\sigma(a) = \frac{1}{1 + \exp(-a)} \in (0,1)$$

- Symmetric

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

- Gradient

$$\sigma'(a) = \frac{\exp(-a)}{(1 + \exp(-a))^2} = \sigma(a)(1 - \sigma(a))$$

Multiple-Class Logistic Regression

Review: binary logistic regression

- Sigmoid

$$\sigma(w^T x + b) = \frac{1}{1 + \exp(-(w^T x + b))}$$

- Interpret as conditional probability

$$p_w(y = 1|x) = \sigma(w^T x + b)$$

$$p_w(y = 0|x) = 1 - p_w(y = 1|x) = 1 - \sigma(w^T x + b)$$

- How to extend to multiclass?

Review: binary logistic regression

- Suppose we model the class-conditional densities $p(x|y=i)$ and class probabilities $p(y=i)$
- Conditional probability by Bayesian rule:

$$p(y=1|x) = \frac{p(x|y=1)p(y=1)}{p(x|y=1)p(y=1) + p(x|y=2)p(y=2)} = \frac{1}{1+\exp(-a)} = \sigma(a)$$

where we define

$$a \coloneqq \ln \frac{p(x|y=1)p(y=1)}{p(x|y=2)p(y=2)} = \ln \frac{p(y=1|x)}{p(y=2|x)}$$

Review: binary logistic regression

- Suppose we model the class-conditional densities $p(x|y=i)$ and class probabilities $p(y=i)$
- $p(y=1|x) = \sigma(a) = \sigma(w^T x + b)$ is equivalent to setting log odds to be linear:

$$a = \ln \frac{p(y=1|x)}{p(y=2|x)} = w^T x + b$$

- Why linear log odds?

Review: binary logistic regression

- Suppose the class-conditional densities $p(x|y=i)$ is normal

$$p(x|y=i) = N(x|\mu_i, I) = \frac{1}{(2\pi)^{d/2}} \exp\{-\frac{1}{2}||x-\mu_i||^2\}$$

- log odd is

$$a = \ln \frac{p(x|y=1)p(y=1)}{p(x|y=2)p(y=2)} = w^T x + b$$

where

$$w = \mu_1 - \mu_2, \qquad b = -\frac{1}{2}\mu_1^T \mu_1 + \frac{1}{2}\mu_2^T \mu_2 + \ln \frac{p(y=1)}{p(y=2)}$$

Multiclass logistic regression

- Suppose we model the class-conditional densities $p(x|y=i)$ and class probabilities $p(y=i)$
- Conditional probability by Bayesian rule:

$$p(y=i|x) = \frac{p(x|y=i)p(y=i)}{\sum_j p(x|y=j)p(y=j)} = \frac{\exp(a_i)}{\sum_j \exp(a_j)}$$

where we define

$$a_i \coloneqq \ln\left[p(x|y=i)p(y=i)\right]$$

Multiclass logistic regression

- Suppose the class-conditional densities $p(x|y=i)$ is normal

$$p(x|y=i) = N(x|\mu_i, I) = \frac{1}{(2\pi)^{d/2}} \exp\{-\frac{1}{2}||x-\mu_i||^2\}$$

- Then

$$a_i \coloneqq \ln\left[p(x|y=i)p(y=i)\right] = -\frac{1}{2}x^T x + \left(w^i\right)^T x + b^i$$

where

$$w^i = \mu_i, \qquad b^i = -\frac{1}{2}\mu_i^T \mu_i + \ln p(y=i) + \ln \frac{1}{(2\pi)^{d/2}}$$

Multiclass logistic regression

- Suppose the class-conditional densities $p(x|y=i)$ is normal

$$p(x|y=i) = N(x|\mu_i, I) = \frac{1}{(2\pi)^{d/2}} \exp\{-\frac{1}{2}||x-\mu_i||^2\}$$

- Cancel out $-\frac{1}{2}x^T x$, we have

$$p(y=i|x) = \frac{\exp(a_i)}{\sum_j \exp(a_j)}, \qquad a_i \coloneqq \left(w^i\right)^T x + b^i$$

where

$$w^i = \mu_i, \qquad b^i = -\frac{1}{2}\mu_i^T \mu_i + \ln p(y=i) + \ln \frac{1}{(2\pi)^{d/2}}$$

Multiclass logistic regression: conclusion

- Suppose the class-conditional densities $p(x|y=i)$ is normal

$$p(x|y=i) = N(x|\mu_i, I) = \frac{1}{(2\pi)^{d/2}} \exp\{-\frac{1}{2}\left||x-\mu_i|\right|^2\}$$

- Then

$$p(y=i|x) = \frac{\exp(\left(w^i\right)^T x + b^i)}{\sum_j \exp((w^j)^T x + b^j)}$$

which is the hypothesis class for multiclass logistic regression

- It is softmax on linear transformation; it can be used to derive the negative log-likelihood loss (cross entropy)

Softmax

- A way to squash $a = (a_1, a_2, \ldots, a_i, \ldots)$ into probability vector p

$$\mathrm{softmax}(a) = \left(\frac{\exp(a_1)}{\sum_j \exp(a_j)}, \frac{\exp(a_2)}{\sum_j \exp(a_j)}, \ldots, \frac{\exp(a_i)}{\sum_j \exp(a_j)}, \ldots \right)$$

- Behave like max: when $a_i \gg a_j (\forall j \neq i)$, $p_i \cong 1, p_j \cong 0$

Cross entropy for conditional distribution

- Let $p_{\text{data}}(y|x)$ denote the empirical distribution of the data
- Negative log-likelihood

$$-\frac{1}{m}\sum_{i=1}^{m}\log p\left(y = y^{(i)} \middle| x^{(i)}\right) = -\mathrm{E}_{p_{\text{data}}(y|x)} \log p(y|x)$$

is the cross entropy between p_{data} and the model output p

- Information theory viewpoint: KL divergence

$$D(p_{\text{data}}||p) = \mathrm{E}_{p_{\text{data}}}[\log \frac{p_{\text{data}}}{p}] = \underbrace{\mathrm{E}_{p_{\text{data}}}[\log p_{\text{data}}]}_{\text{Entropy; constant}} - \underbrace{\mathrm{E}_{p_{\text{data}}}[\log p]}_{\text{Cross entropy}}$$

Cross entropy for full distribution

- Let $p_{\text{data}}(x, y)$ denote the empirical distribution of the data
- Negative log-likelihood

$$-\frac{1}{m}\sum_{i=1}^{m} \log p(x^{(i)}, y^{(i)}) = -\mathrm{E}_{p_{\text{data}}(x,y)} \log p(x, y)$$

is the cross entropy between p_{data} and the model output p

Summary of the principles

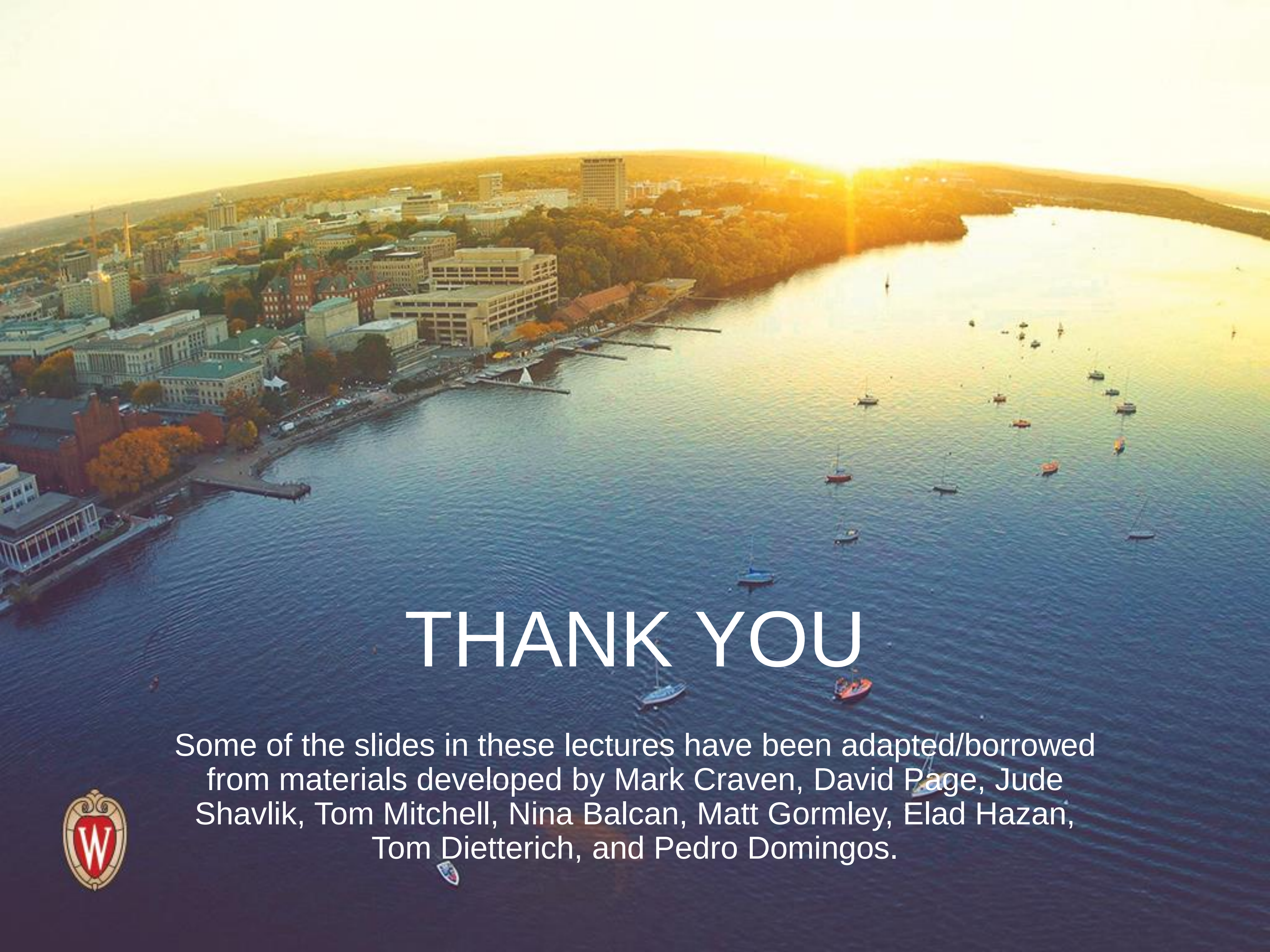
- Discriminative approach with negative log-likelihood loss
- Step 1: specify $p(y|x)$
- Step 2: use MLE to derive the negative log-likelihood loss

- Example: if $p(y|x)$ is sigmoid over a linear function of x, then we get logistic regression

Summary of the principles

- From generative to discriminative
- Step 0: specify $p(x|y)$ and $p(y)$
- Step 1: compute $p(y|x)$
- Step 2: use MLE to derive the negative log-likelihood loss

- Example: if $p(x|y)$ are Gaussians, then we get logistic regression



THANK YOU

Some of the slides in these lectures have been adapted/borrowed from materials developed by Mark Craven, David Page, Jude Shavlik, Tom Mitchell, Nina Balcan, Matt Gormley, Elad Hazan, Tom Dietterich, and Pedro Domingos.