Introduction to Machine Learning Part 4: Linear Classification

CS 540

Yingyu Liang

Review: machine learning basics

Math formulation

- Given training data $\{(x_i, y_i): 1 \le i \le n\}$ i.i.d. from distribution D
- Find $y = f(x) \in \boldsymbol{\mathcal{H}}$ that minimizes $\hat{L}(f) = \frac{1}{n}\sum_{i=1}^{n} l(f, x_i, y_i)$
- s.t. the expected loss is small

$$L(f) = \mathbb{E}_{(x,y)\sim D}[l(f, x, y)]$$

Machine learning 1-2-3

- Collect data and extract features
- Build model: choose hypothesis class $\mathcal{H}$ and loss function l
- Optimization: minimize the empirical loss

Machine learning 1-2-3

Experience

- Collect data and extract features
- Build model: choose hypothesis class $\mathcal{H}$ and loss function l
- Optimization: minimize the empirical loss

Prior knowledge

Example: Linear regression

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

l_2 loss

Linear model $\mathcal{H}$

Why l_2 loss

- Why not choose another loss
  - l_1 loss, hinge loss, exponential loss, …
- Empirical: easy to optimize
  - For linear case: $\mathrm{w} = (X^T X)^{-1} X^T y$
- Theoretical: a way to encode prior knowledge

Questions:

- What kind of prior knowledge?
- Principal way to derive loss?

Maximum Likelihood Estimation

Maximum Likelihood Estimation (MLE)

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Let $\{P_\theta(x, y): \theta \in \Theta\}$ be a family of distributions indexed by θ

- Would like to pick θ so that $P_\theta(x, y)$ fits the data well

Maximum Likelihood Estimation (MLE)

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Let $\{P_\theta(x, y): \theta \in \Theta\}$ be a family of distributions indexed by θ

- "fitness" of θ to one data point (x_i, y_i)

$$\text{likelihood}(\theta; x_i, y_i) \coloneqq P_\theta(x_i, y_i)$$

Maximum Likelihood Estimation (MLE)

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Let $\{P_\theta(x, y): \theta \in \Theta\}$ be a family of distributions indexed by θ

- "fitness" of θ to i.i.d. data points $\{(x_i, y_i)\}$

$$\text{likelihood}(\theta; \{x_i, y_i\}) \coloneqq P_\theta(\{x_i, y_i\}) = \prod_i P_\theta(x_i, y_i)$$

Maximum Likelihood Estimation (MLE)

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Let $\{P_\theta(x, y): \theta \in \Theta\}$ be a family of distributions indexed by θ

- MLE: maximize "fitness" of θ to i.i.d. data points $\{(x_i, y_i)\}$

$$\theta_{ML} = \mathrm{argmax}_{\theta \in \Theta} \prod_i P_\theta(x_i, y_i)$$

Maximum Likelihood Estimation (MLE)

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Let $\{P_\theta(x, y): \theta \in \Theta\}$ be a family of distributions indexed by θ

- MLE: maximize "fitness" of θ to i.i.d. data points $\{(x_i, y_i)\}$

$$\theta_{ML} = \operatorname{argmax}_{\theta \in \Theta} \log[\prod_i P_\theta(x_i, y_i)]$$

$$\theta_{ML} = \operatorname{argmax}_{\theta \in \Theta} \sum_i \log[P_\theta(x_i, y_i)]$$

Maximum Likelihood Estimation (MLE)

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Let $\{P_\theta(x, y): \theta \in \Theta\}$ be a family of distributions indexed by θ

- MLE: negative log-likelihood loss

$$\theta_{ML} = \mathrm{argmax}_{\theta \in \Theta} \sum_i \log(P_\theta(x_i, y_i))$$

$$l(P_\theta, x_i, y_i) = -\log(P_\theta(x_i, y_i))$$
$$\hat{L}(P_\theta) = -\sum_i \log(P_\theta(x_i, y_i))$$

MLE: conditional log-likelihood

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Let $\{P_\theta(y|x): \theta \in \Theta\}$ be a family of distributions indexed by θ

Only care about predicting y from x; do not care about p(x)

- MLE: negative conditional log-likelihood loss

$$\theta_{ML} = \text{argmax}_{\theta \in \Theta} \sum_i \log(P_\theta(y_i|x_i))$$

$$l(P_\theta, x_i, y_i) = -\log(P_\theta(y_i|x_i))$$

$$\hat{L}(P_\theta) = -\sum_i \log(P_\theta(y_i|x_i))$$

MLE: conditional log-likelihood

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Let $\{P_\theta(y|x): \theta \in \Theta\}$ be a family of distributions indexed by θ

P(y|x): discriminative;
P(x,y): generative

- MLE: negative conditional log-likelihood loss

$$\theta_{ML} = \operatorname{argmax}_{\theta \in \Theta} \sum_i \log(P_\theta(y_i|x_i))$$

$$l(P_\theta, x_i, y_i) = -\log(P_\theta(y_i|x_i))$$

$$\hat{L}(P_\theta) = -\sum_i \log(P_\theta(y_i|x_i))$$

Example: l_2 loss

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Find $f_\theta(x)$ that minimizes $\hat{L}(f_\theta) = \frac{1}{n}\sum_{i=1}^{n}(f_\theta(x_i) - y_i)^2$

Example: l_2 loss

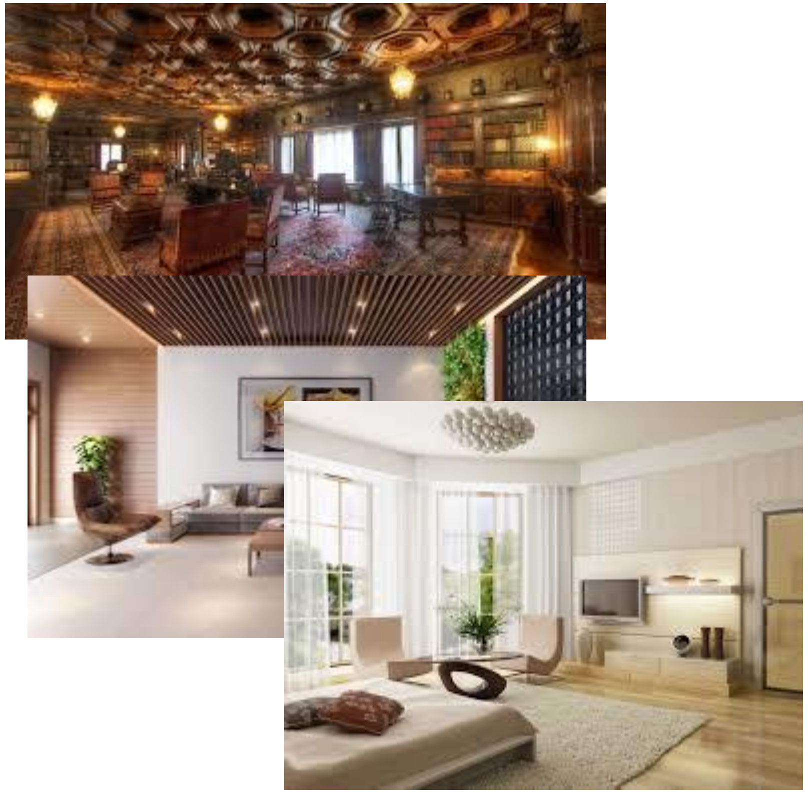
- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Find $f_\theta(x)$ that minimizes $\hat{L}(f_\theta) = \frac{1}{n}\sum_{i=1}^{n}(f_\theta(x_i) - y_i)^2$

l_2 loss: Normal + MLE

- Define $P_\theta(y|x) = \text{Normal}(y; f_\theta(x), \sigma^2)$
- $\log(P_\theta(y_i|x_i)) = \frac{-1}{2\sigma^2}(f_\theta(x_i) - y_i)^2 - \log(\sigma) - \frac{1}{2}\log(2\pi)$
- $\theta_{ML} = \text{argmin}_{\theta\in\Theta}\frac{1}{n}\sum_{i=1}^{n}(f_\theta(x_i) - y_i)^2$

Linear classification

Example 1: image classification



Indoor

outdoor

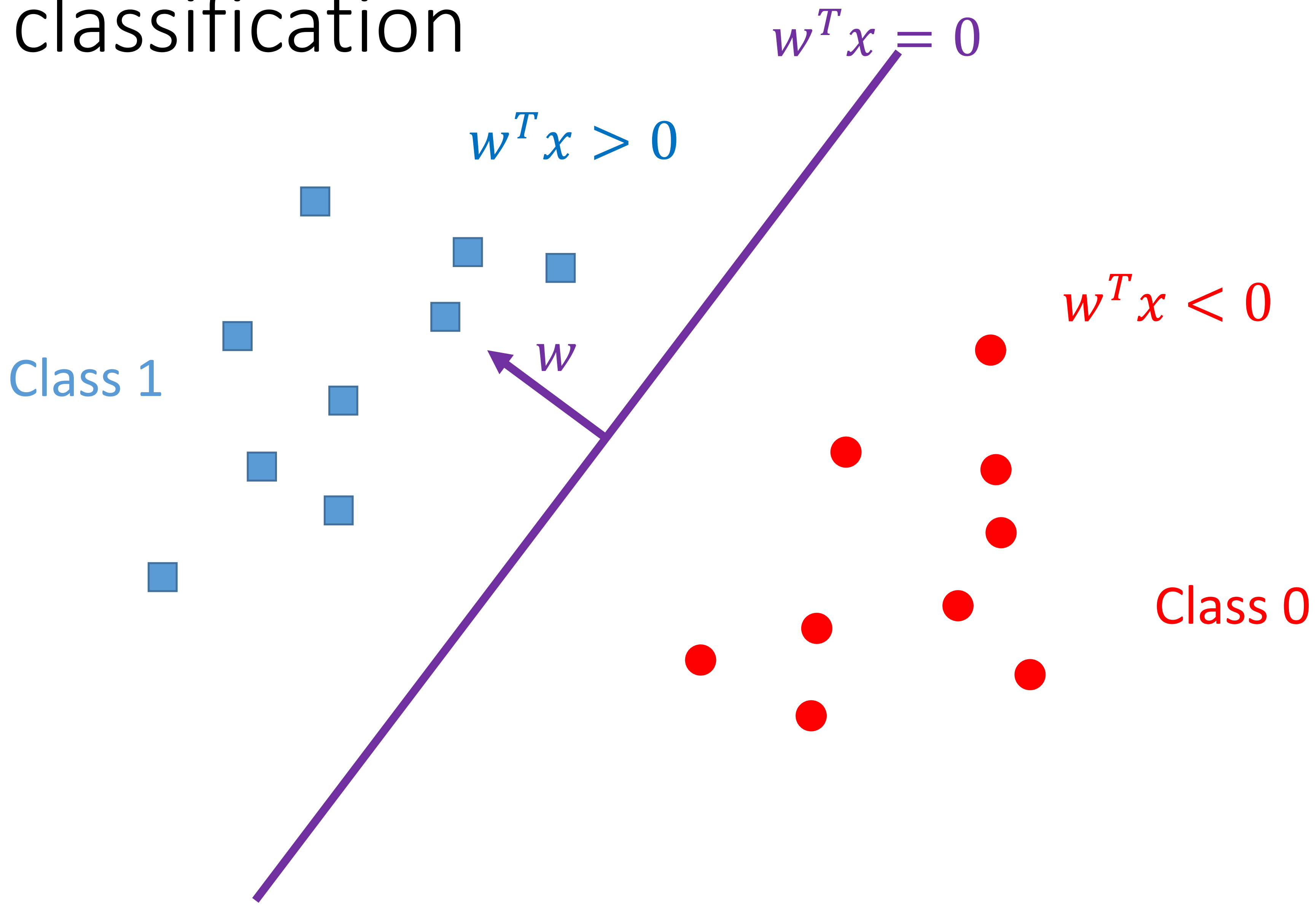
Example 2: Spam detection

	#"$"	#"Mr."	#"sale"	…	Spam?
Email 1	2	1	1		Yes
Email 2	0	1	0		No
Email 3	1	1	1		Yes
…					
Email n	0	0	0		No
New email	0	0	1		??

Why classification

- Classification: a kind of summary
- Easy to interpret
- Easy for making decisions

Linear classification



Linear classification: natural attempt

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ 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 $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Find $f_w(x) = w^T x$ to minimize $\hat{L}(f_w) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}[\text{step}(w^T x_i) \neq y_i]$

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

0-1 loss

Linear classification: simple approach

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

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

Linear classification: logistic regression

- Probabilistic view: try to output the probability distribution $P(y|x)$

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

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

Sigmoid

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

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

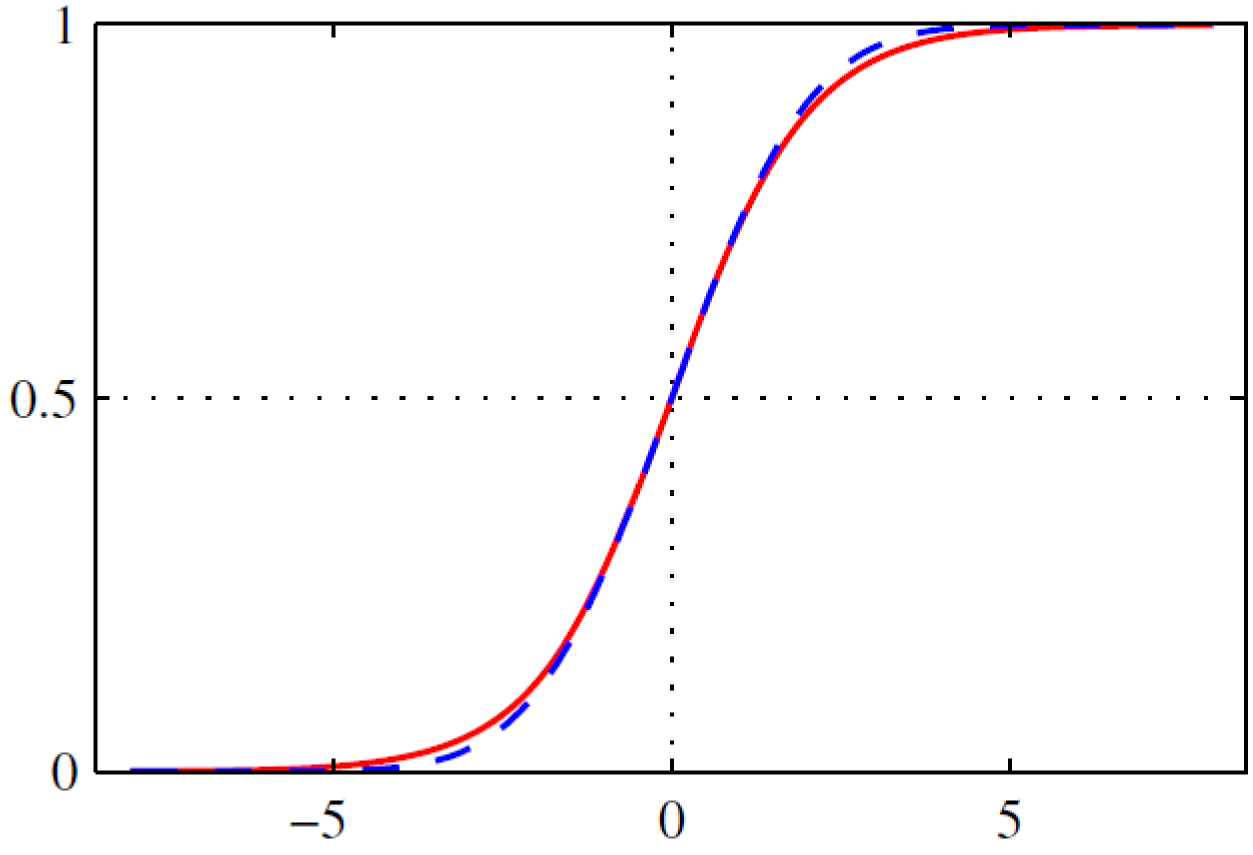


Figure borrowed from *Pattern Recognition and Machine Learning*, Bishop

Linear classification: logistic regression

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Find w that minimizes

$$\hat{L}(w) = -\frac{1}{n}\sum_{i=1}^{n} \log P_w(y_i|x_i)$$

$$\hat{L}(w) = -\frac{1}{n}\sum_{y_i=1} \log\sigma(w^T x_i) - \frac{1}{n}\sum_{y_i=0} \log[1 - \sigma(w^T x_i)]$$

Logistic regression:
MLE with sigmoid

Linear classification: logistic regression

- Given training data $\{(x_i, y_i): 1 \leq i \leq n\}$ i.i.d. from distribution D
- Find w that minimizes

$$\hat{L}(w) = -\frac{1}{n} \sum_{y_i=1} \log \sigma(w^T x_i) - \frac{1}{n} \sum_{y_i=0} \log[1 - \sigma(w^T x_i)]$$

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

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