Supervised Learning & Logistic Regression

Slides by: Minh Haoi (SBU)

Image Categorization is Important

What scene is it?

Forest City Campus Sea

What animal is it?

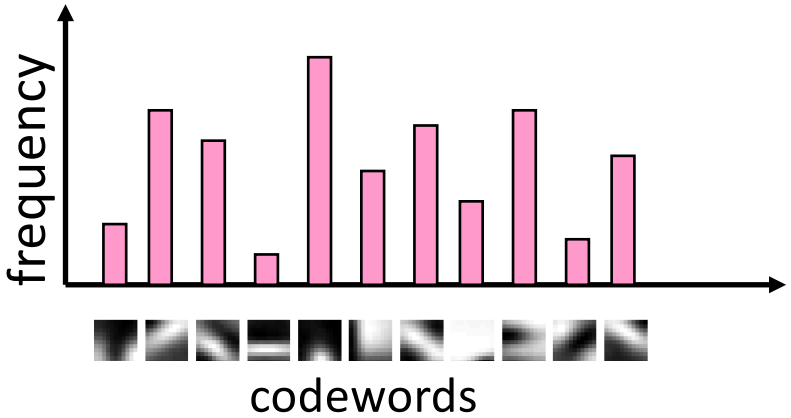
Dog Cat Lion Penguin

What is he doing?

Reading Phoning Taking photo

Supervised Learning

Input (features) → Output (targets, labels)



Forest

City

Campus

Sea

Problem formulation

Given the training set: $(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)$

Task: find a predictor function: $y = f(\mathbf{x})$

General Machine Learning Approach

- Using domain/prior knowledge, assume a model for the predictor
  - E.g., linear model, quadratic model

- The functional form of the model is fixed,
  but it has unknown parameters $y = f(\mathbf{x}; \Theta)$

- Use training data to <u>learn</u> the model's parameters

$$(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n) \longrightarrow \Theta$$

- Use the <u>learned</u> parameters for prediction: $\mathbf{x}, \Theta \longrightarrow y$

Logistic Regression

Logistic Regression is a discriminative classifier:

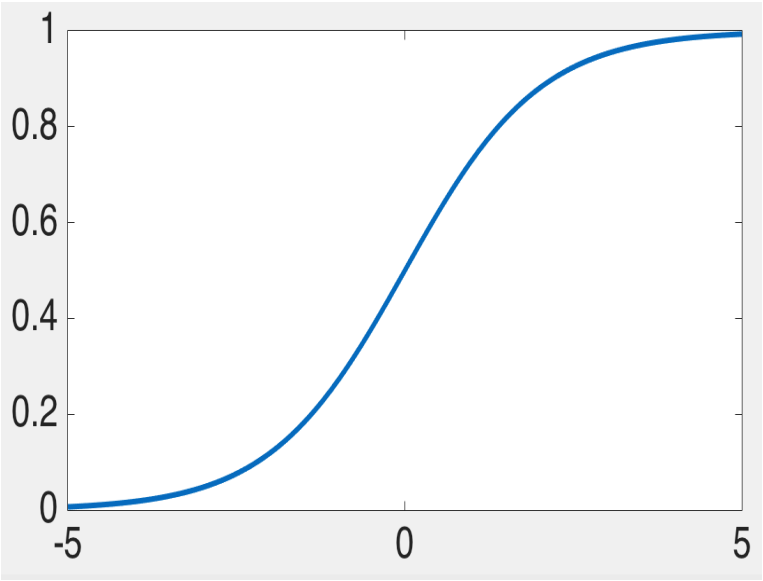
+ Learn P(Y|**X**) directly
+ Assume a functional form of P(Y|**X**)

Assume a probability as Sigmoid function

$$P(Y=1|\mathbf{X}) = sigmoid(\boldsymbol{\theta}^T\mathbf{X})$$

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

$$P(Y=0|\mathbf{X}) = 1 - P(Y=1|\mathbf{X})$$

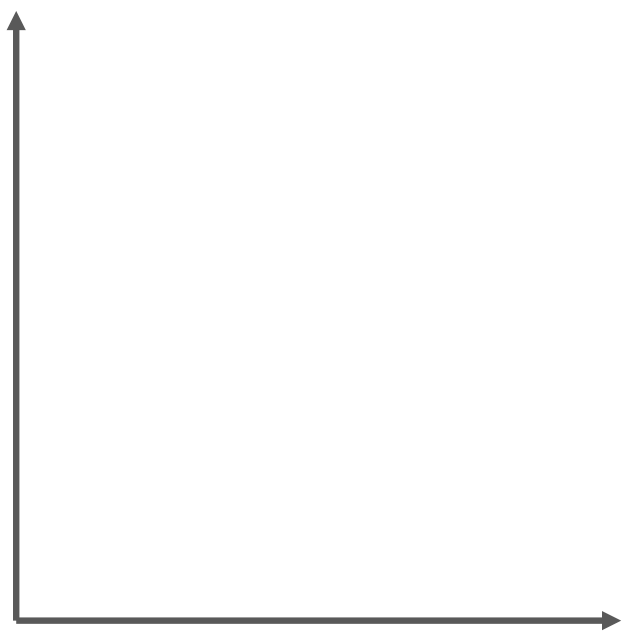


Understand the Sigmoid

Logistic Regression – Linear Classifier

Classification decision: $P(Y = 1|\mathbf{X}) \geq 0.5$

This happens when $\boldsymbol{\theta}^T \mathbf{X} \geq 0$



Learning the Parameters

Maximize the Conditional likelihood $P(\{Y^j\}|\{\mathbf{X}^j\}, \boldsymbol{\theta})$

Independently Identically Distributed (iid) assumption

$$= \prod_j P(Y^j|\mathbf{X}^j, \boldsymbol{\theta})$$

Maximize the Conditional Log-likelihood $L(\boldsymbol{\theta}) = \sum_j \log(P(Y^j|\mathbf{X}^j, \boldsymbol{\theta}))$

$$L(\boldsymbol{\theta}) = \sum_j Y^j \log(P(Y=1|\mathbf{X}^j, \boldsymbol{\theta})) + (1-Y^j)\log(P(Y=0|\mathbf{X}^j, \boldsymbol{\theta}))$$

Learning the Parameters

Maximize the Conditional Log-likelihood

$$L(\boldsymbol{\theta}) = \sum_j \log(P(Y^j|\mathbf{X}^j, \boldsymbol{\theta}))$$

$$L(\boldsymbol{\theta}) = \sum_j Y^j \log(P(Y=1|\mathbf{X}^j, \boldsymbol{\theta})) + (1-Y^j)\log(P(Y=0|\mathbf{X}^j, \boldsymbol{\theta}))$$

$$= \sum_j [Y^j(\boldsymbol{\theta}^T\mathbf{X}^j) - \log(1+\exp(\boldsymbol{\theta}^T\mathbf{X}^j))]$$

Bad News: No closed-form solution

Good News: The function is concave => Easy to optimize

Optimization with Gradient Ascent

Iterative optimization:

$$\boldsymbol{\theta}^{(t+1)} := \boldsymbol{\theta}^t + \eta \left. \frac{\partial L}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta} = \boldsymbol{\theta}^t}$$

$$\frac{\partial L}{\partial \boldsymbol{\theta}} =$$

Logistic Regression Review

Objective: maximize the log-likelihood

$$L(\boldsymbol{\theta}) = \sum_j [Y^j(\boldsymbol{\theta}^T \mathbf{X}^j) - \log(1 + \exp(\boldsymbol{\theta}^T \mathbf{X}^j))]$$

Optimization with gradient ascent

$$\boldsymbol{\theta}^{(t+1)} := \boldsymbol{\theta}^t + \eta \left. \frac{\partial L}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta} = \boldsymbol{\theta}^t}$$

The derivative

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = \sum_j [Y^j \mathbf{X}^j - \frac{\exp(\boldsymbol{\theta}^T \mathbf{X}^j)}{1 + \exp(\boldsymbol{\theta}^T \mathbf{X}^j)} \mathbf{X}^j]$$

$$= \sum_j [Y^j - P(Y = 1 | \mathbf{X}^j)] \mathbf{X}^j$$

Understand the gradient update

Optimization with gradient ascent $\boldsymbol{\theta}^{(t+1)} := \boldsymbol{\theta}^t + \eta \left.\frac{\partial L}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}^t}$

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = \sum_j [Y^j - P(Y=1|\mathbf{X}^j)]\mathbf{X}^j$$

Case 1: $Y^j = 1$

+ If $P(Y=1|\mathbf{X}^j)$ is big, $\mathbf{X}^j$ induces a weak pull

+ If $P(Y=1|\mathbf{X}^j)$ is small, $\mathbf{X}^j$ induces a strong pull

Case 2: $Y^j = 0$

+ If $P(Y=1|\mathbf{X}^j)$ is big, $\mathbf{X}^j$ induces a strong push

+ If $P(Y=1|\mathbf{X}^j)$ is small, $\mathbf{X}^j$ induces a weak push

Stochastic Gradient Descent with Mini-batches

Optimization with gradient ascent $$\boldsymbol{\theta}^{(t+1)} := \boldsymbol{\theta}^t + \eta \left.\frac{\partial L}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}^t}$$

$$L(\boldsymbol{\theta}) = \sum_j \underbrace{[Y^j(\boldsymbol{\theta}^T \mathbf{X}^j) - \log(1 + \exp(\boldsymbol{\theta}^T \mathbf{X}^j))]}_{L_j(\boldsymbol{\theta})}$$

Exact gradient computation $$\frac{\partial L}{\partial \boldsymbol{\theta}} = \sum_{j=1}^{n} \frac{\partial L_j}{\partial \boldsymbol{\theta}}$$

Approximate gradient computation $$\frac{\partial L}{\partial \boldsymbol{\theta}} \approx \frac{n}{|\mathcal{B}|} \sum_{j \in \mathcal{B}} \frac{\partial L_j}{\partial \boldsymbol{\theta}}$$

Logistic Regression for *k* classes

$$P(Y=1|\mathbf{X}) = \frac{\exp(\boldsymbol{\theta}_1^T\mathbf{X})}{1+\sum_{i=1}^{k-1}\exp(\boldsymbol{\theta}_i^T\mathbf{X})}$$

$$\vdots$$

$$P(Y=k-1|\mathbf{X}) = \frac{\exp(\boldsymbol{\theta}_{k-1}^T\mathbf{X})}{1+\sum_{i=1}^{k-1}\exp(\boldsymbol{\theta}_i^T\mathbf{X})}$$

$$P(Y=k|\mathbf{X}) = \frac{1}{1+\sum_{i=1}^{k-1}\exp(\boldsymbol{\theta}_i^T\mathbf{X})}$$

Equivalent Formulation: Soft-max

- Parameterization $$P(Y = j|\mathbf{X}) = \frac{\exp(\boldsymbol{\theta}_j^T \mathbf{X})}{\sum_{i=1}^{k} \exp(\boldsymbol{\theta}_i^T \mathbf{X})}$$

- Loss function

$$L(\boldsymbol{\theta}) = \sum_{j=1}^{n} \log(P(Y^j|\mathbf{X}^j, \boldsymbol{\theta}))$$

$$L(\boldsymbol{\theta}) = \sum_{j=1}^{n} \sum_{i=1}^{k} \delta(Y^j = i) \log \left(\frac{\exp(\boldsymbol{\theta}_j^T \mathbf{X})}{\sum_{i=1}^{k} \exp(\boldsymbol{\theta}_i^T \mathbf{X})} \right)$$

What you need to know

- Supervised learning
- Sigmoid function
- Logistic regression
  - Binary case
  - Multiple classes
- Stochastic Gradient Descent