西安交通大学
XI'AN JIAOTONG UNIVERSITY

机器学习在信息安全中的应用

言浬
li.yan.88@xjtu.edu.cn

2024年11月

第二章：线性模型

2.1 线性回归

2.1 线性回归

线性判别模型

判别模型

- 性质
  - 建模预测变量和观测变量之间的关系
  - 也称作条件模型 (Conditional Models)
- 分类
  - 确定性判别模型：$y = f_\theta(x)$
  - 概率判别模型：$p_\theta(y|x)$

本节集中介绍线性判别模型 (linear regression)

线性判别模型

判别模型

- 性质
  - 建模预测变量和观测变量之间的关系
  - 也称作条件模型 (Conditional Models)
- 分类
  - 确定性判别模型：$y = f_\theta(x)$
  - 概率判别模型：$p_\theta(y|x)$

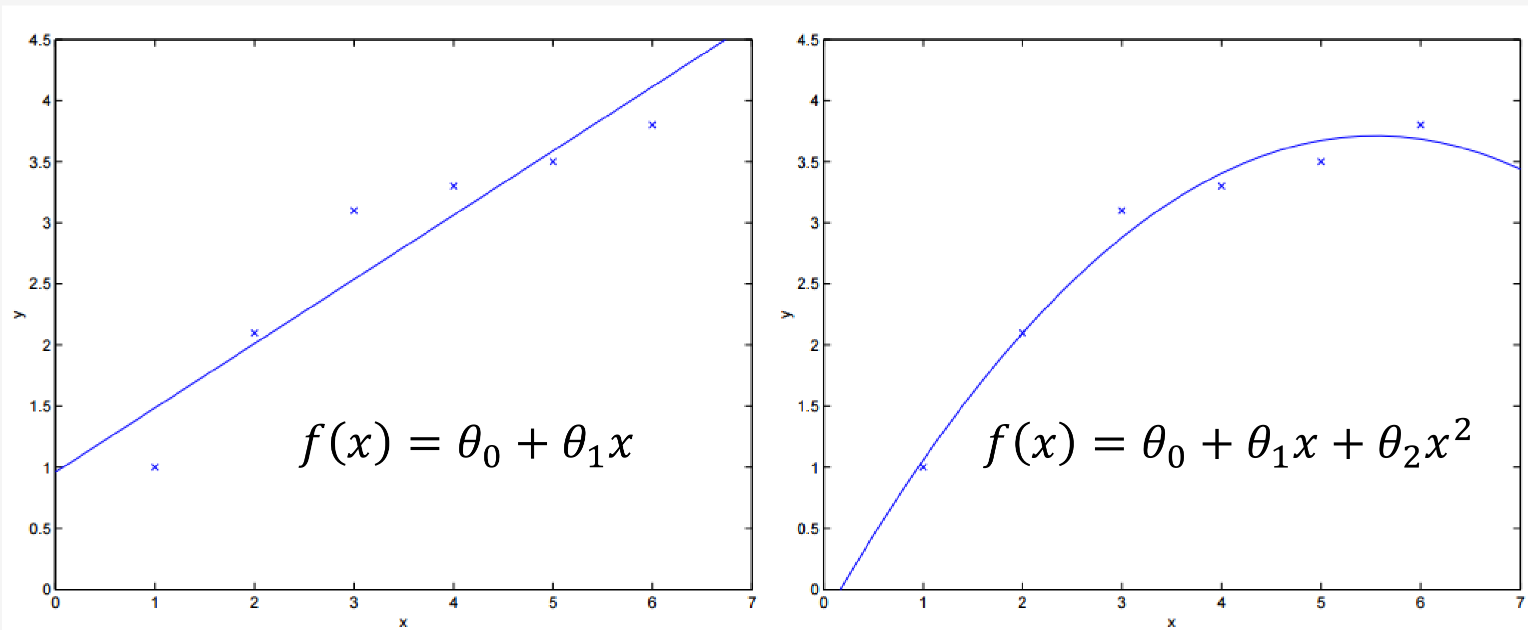
线性判别模型 (linear regression)

$$y = f_\theta(x) = \theta_0 + \sum_{j=1}^{d} \theta_j x_j = \theta^\top x$$

$$x = (1, x_1, x_2, \cdots, x_d)$$

2.1 线性回归

- 一维的线性回归和二次回归（都是线性模型）



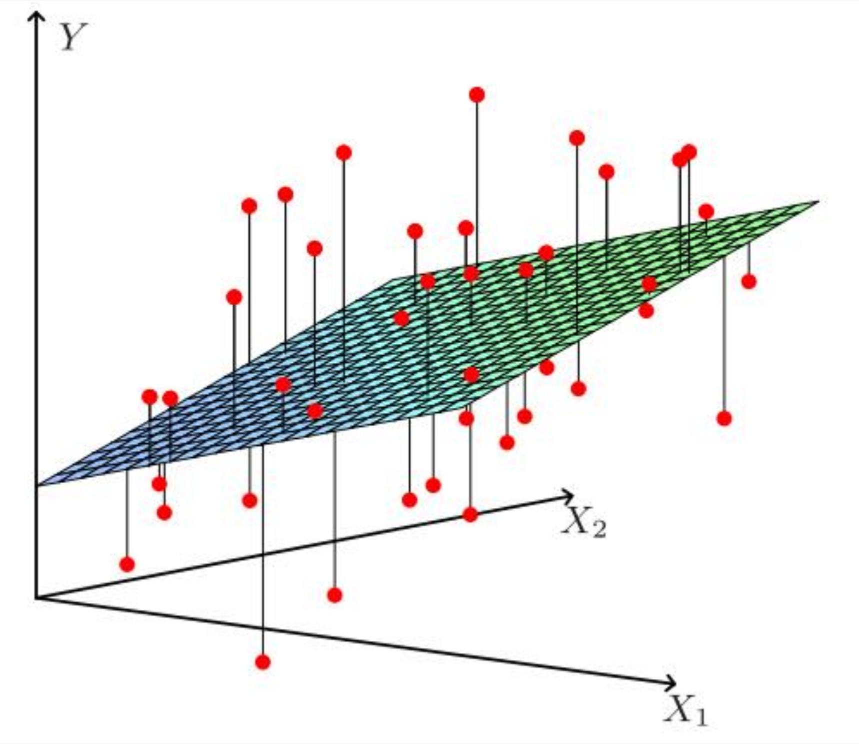
线性回归

二次回归
(一种广义线性模型)

2.1 线性回归

- 二维的线性回归模型

$$f(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$



2.1 线性回归

学习目标

- 使预测值和真实值的距离越近越好

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f_\theta(x_i))$$

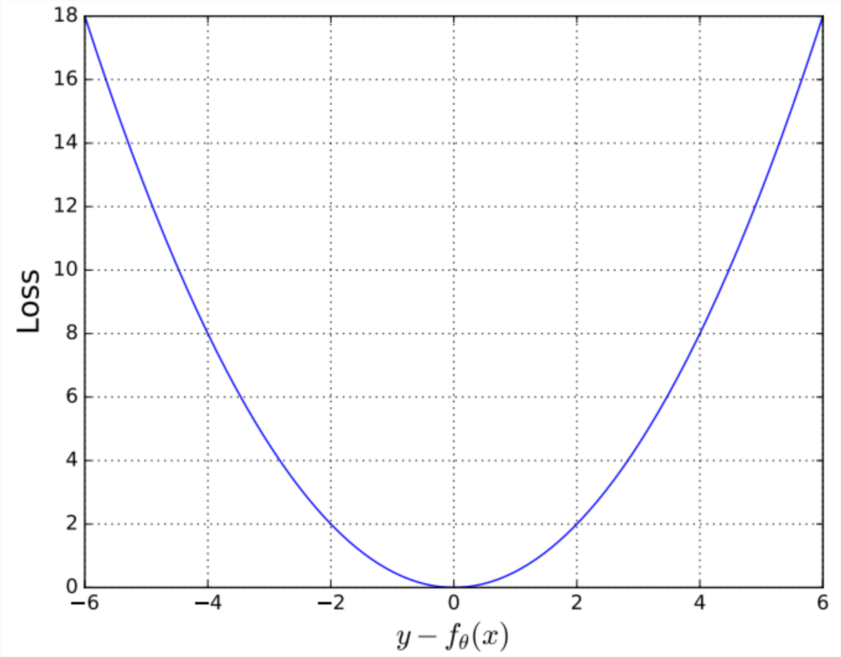
- 损失函数$\mathcal{L}(y_i, f_\theta(x_i))$ 测量预测值和真实值之间的误差，越小越好
- 具体损失函数的定义依赖于具体的数据和任务
- 最广泛使用的损失回归函数：平方误差(squared loss)

$$\mathcal{L}(y_i, f_\theta(x_i)) = \frac{1}{2}\left(y_i - f_\theta(x_i)\right)^2$$

2.1 线性回归

平方误差

$$\mathcal{L}\big(y_i, f_\theta(x_i)\big) = \frac{1}{2}\big(y_i - f_\theta(x_i)\big)^2$$



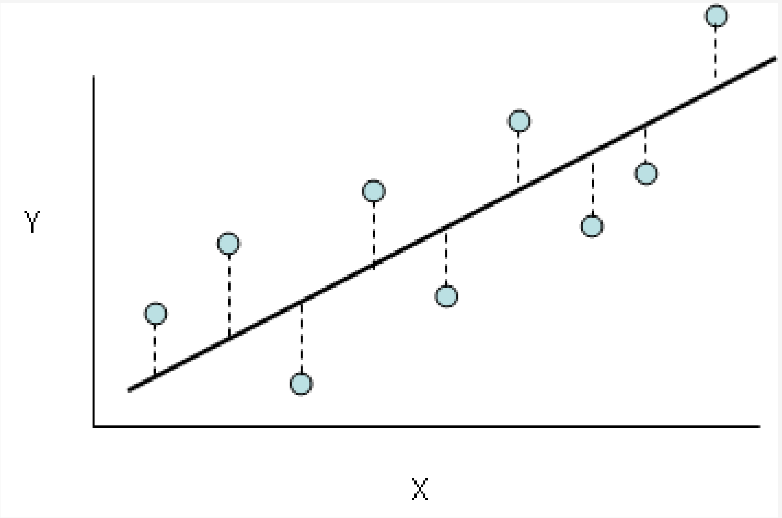
- 对预测误差大的有更大的惩罚
- 容忍很小的预测误差
  - 观测误差等
  - 提升模型的泛化能力

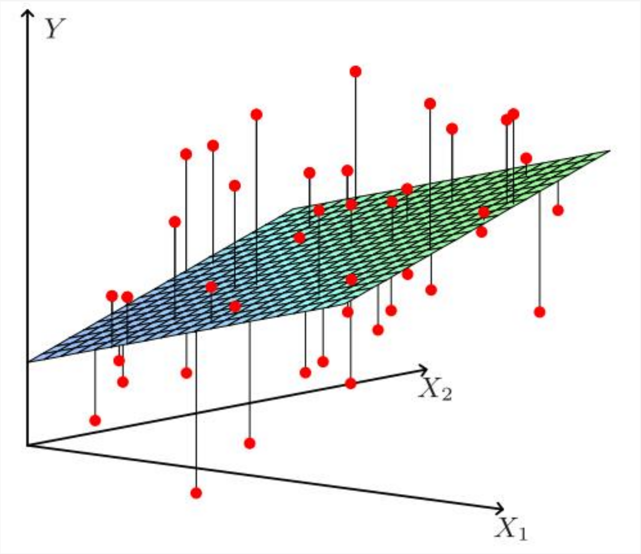
2.1 线性回归

最小均方误差回归

- 优化目标是最小化训练数据上的均方误差

$$J_\theta = \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - f_\theta(x_i)\right)^2 \qquad \min_\theta J_\theta$$

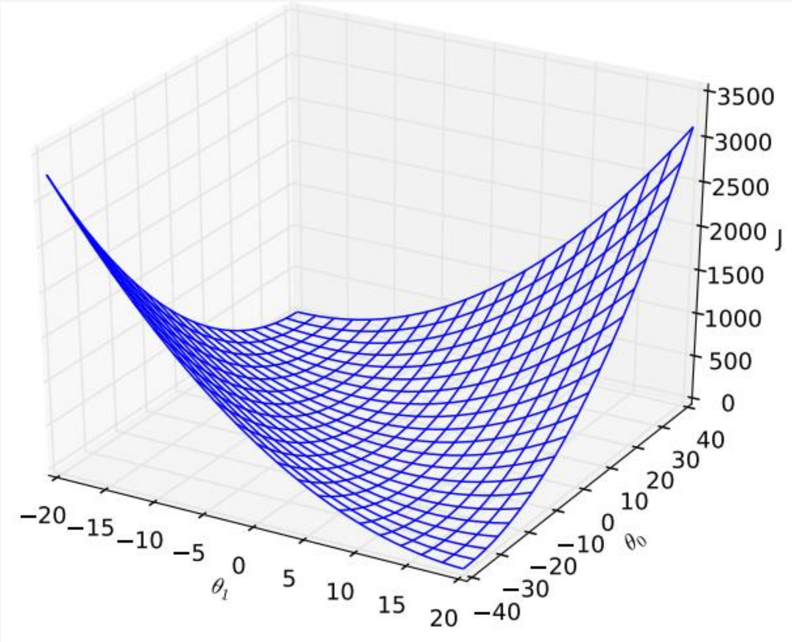


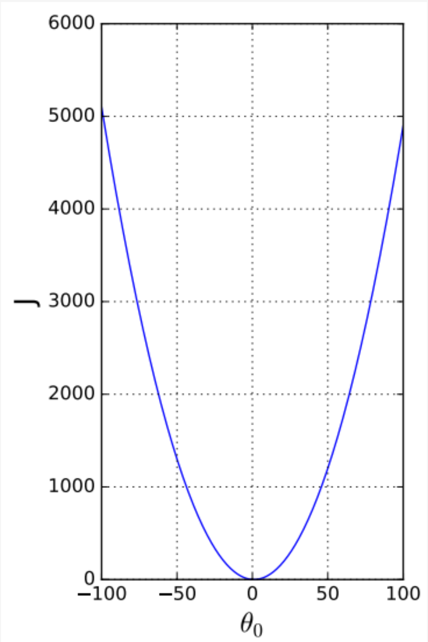


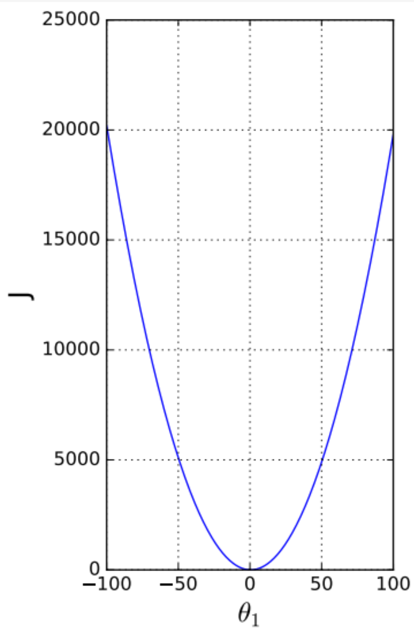
最小化目标函数

- 举一个$N=1$的简单示例，对于数据点$(x,y)=(2,1)$

$$J(\theta)=\frac{1}{2}(y-\theta_0-\theta_1 x)^2=\frac{1}{2}(1-\theta_0-2\theta_1)^2$$

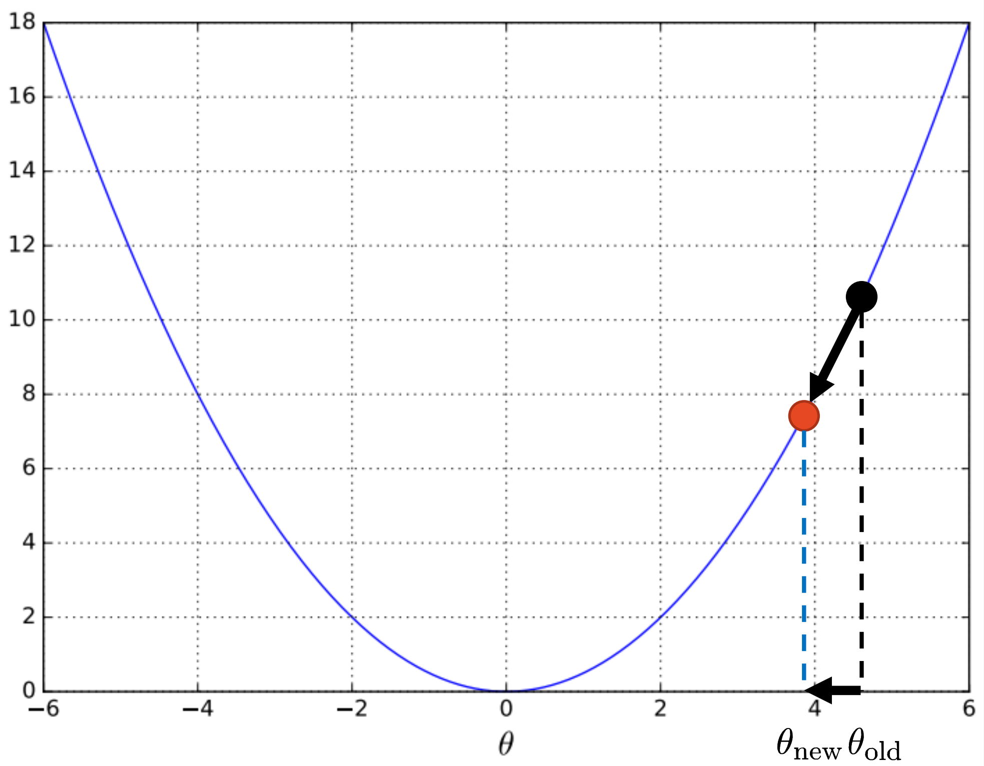






梯度学习方法

$\mathcal{L}(\theta)$



$$\theta_{\text{new}} \leftarrow \theta_{\text{old}} - \eta \frac{\partial \mathcal{L}(\theta)}{\partial \theta}$$

西安交通大学
XI'AN JIAOTONG UNIVERSITY

2.2 梯度更新方式

2.2：梯度更新方式

2.2.1 批量梯度下降

2.2.2 随机梯度下降

2.2.3 小批量梯度下降

2.2.4 基本搜索步骤

2.2 梯度更新方式

批量梯度下降

- 优化目标

$$J(\theta) = \frac{1}{2N}\sum_{i=1}^{N}\left(y_i - f_\theta(x_i)\right)^2 \qquad \min_{\theta} J(\theta)$$

- 根据整个批量数据的梯度更新参数 $\theta_{\text{new}} \leftarrow \theta_{\text{old}} - \eta \frac{\partial J(\theta)}{\partial \theta}$

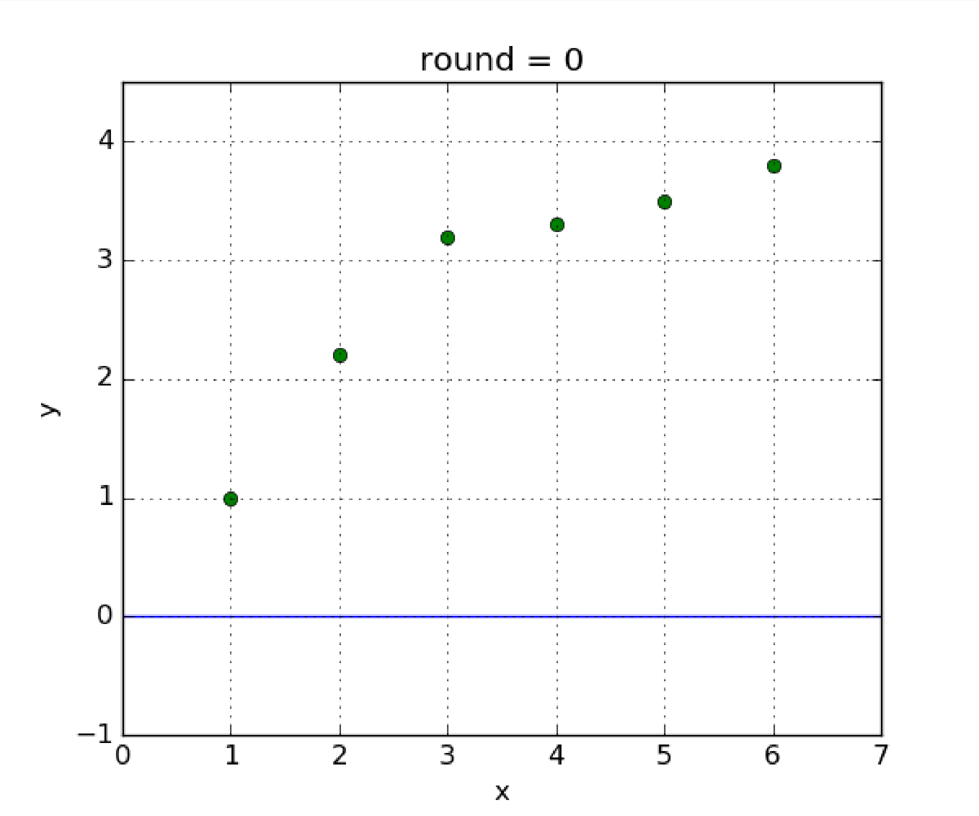
$$\frac{\partial J(\theta)}{\partial \theta} = -\frac{1}{N}\sum_{i=1}^{N}\left(\left(y_i - f_\theta(x_i)\right)\frac{\partial f_\theta(x_i)}{\partial \theta}\right)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_\theta(x_i)\right)x_i$$

$$\theta_{\text{new}} = \theta_{\text{old}} + \eta\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_\theta(x_i)\right)x_i$$

2.2 梯度更新方式

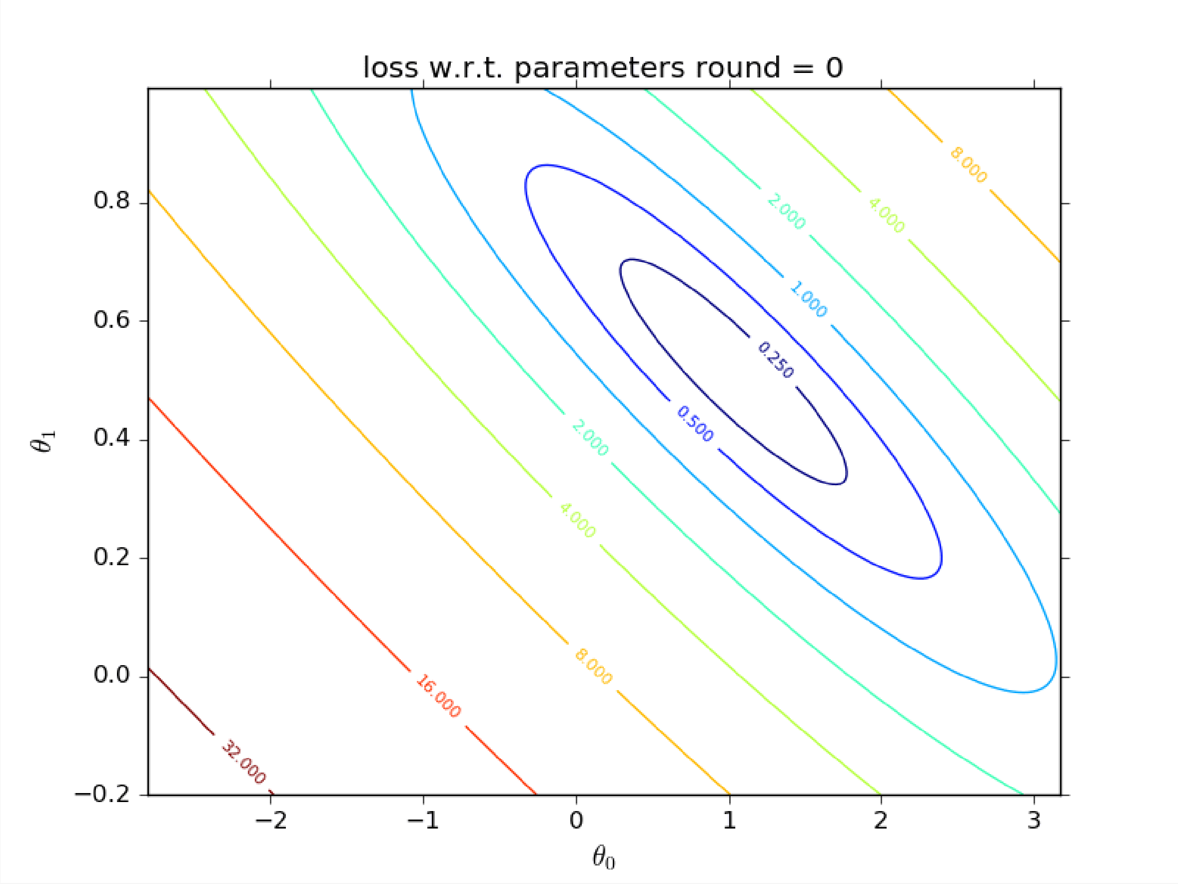
利用线性模型学习-曲线拟合



$$f(x) = \theta_0 + \theta_1 x$$

2.2 梯度更新方式

利用线性模型学习-参数改变



批量梯度更新

2.2 梯度更新方式

随机梯度下降

- 优化目标

$$J^{(i)}(\theta)=\frac{1}{2}\big(y_i-f_\theta(x_i)\big)^2 \qquad \min_\theta \frac{1}{N}\sum_i J^{(i)}(\theta)$$

- 根据整个批量数据的梯度更新参数 $\theta_{\text{new}} = \theta_{\text{old}}-\eta\frac{\partial J^{(i)}(\theta)}{\partial\theta}$

$$\frac{\partial J^{(i)}(\theta)}{\partial\theta}=-\big(y_i-f_\theta(x_i)\big)\frac{\partial f_\theta(x_i)}{\partial\theta}$$

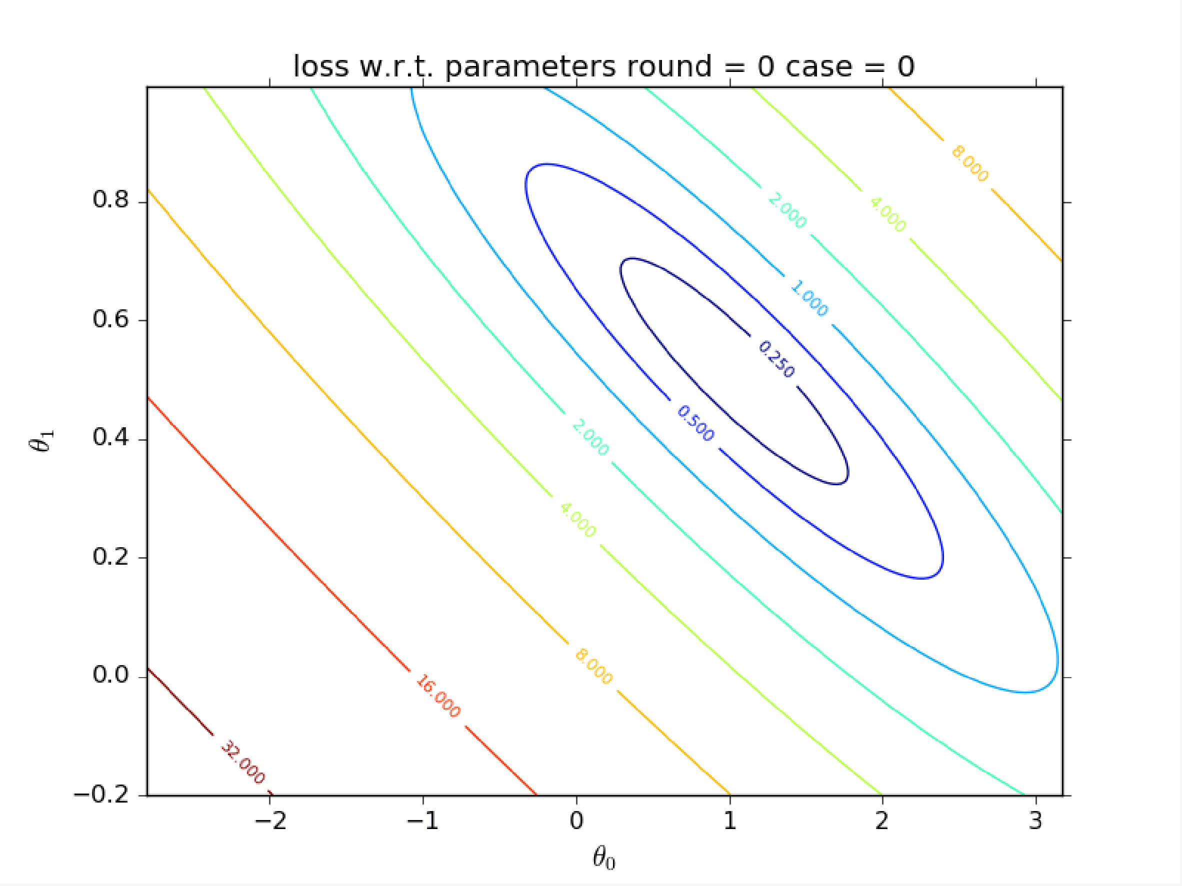
$$=-(\,y_i-f_\theta(x_i))\,x_i$$

$$\theta_{\text{new}}=\theta_{\text{old}}+\eta\big(y_i-f_\theta(x_i)\big)x_i$$

- 对比批量梯度下降
  - 更快地更新参数(优点)
  - 学习中不确定性或震荡(缺点)

2.2 梯度更新方式

利用线性模型学习-参数改变



随机梯度更新

2.2 梯度更新方式

小批量梯度下降

算法思想

批量梯度下降和随机梯度下降的结合

训练步骤

- 将整个训练集分成 K 个小批量（mini-batches）

$$\{1,2,3,\cdots,K\}$$

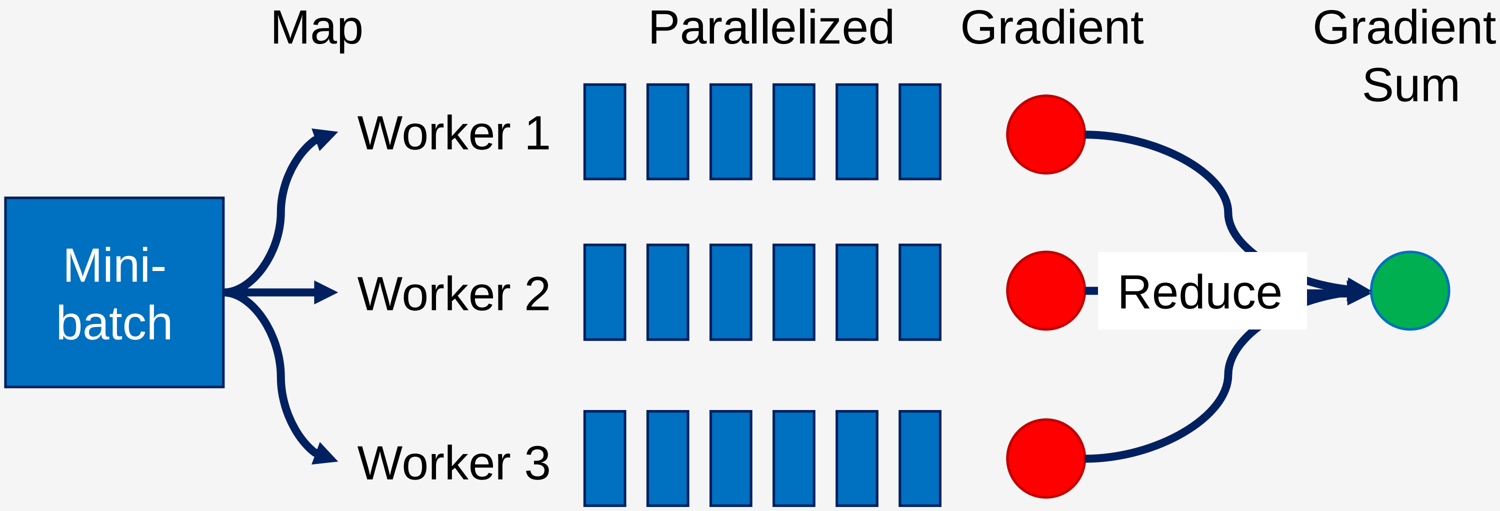
- 对于每一个小批量 k, 做一步批量下降来降低

$$J^{(k)}(\theta)=\frac{1}{2N_k}\sum_{i=1}^{N_k}\left(y_i-f_\theta(x_i)\right)^2$$

- 对于每一个小批量，更新参数 $\theta_{\text{new}} = \theta_{\text{old}}-\eta\frac{\partial J^{(k)}(\theta)}{\partial\theta}$

2.2 梯度更新方式

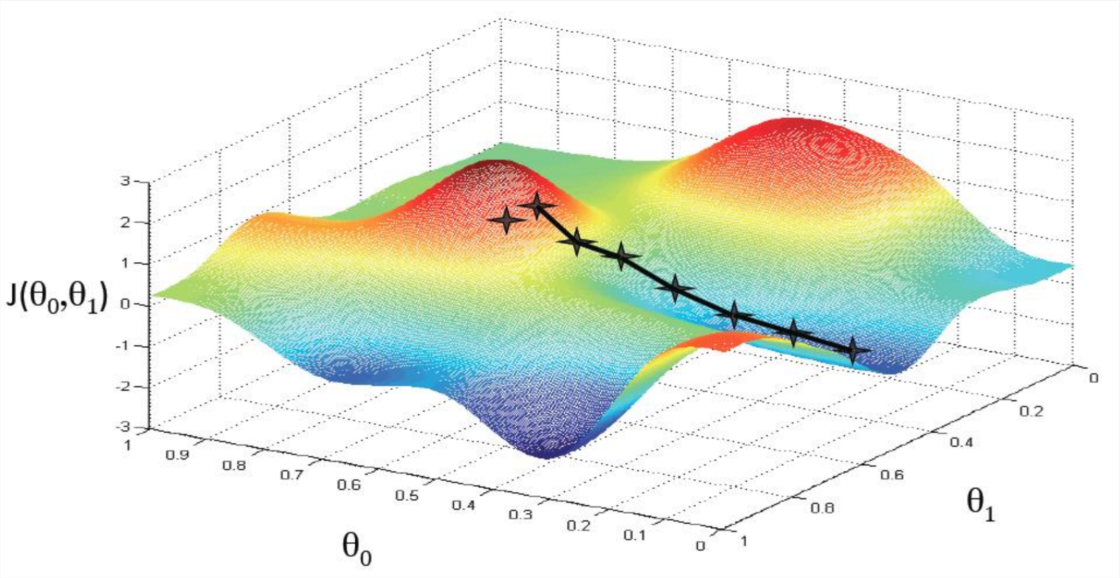
- 结合了批量梯度下降和随机梯度下降的优点
  - 批量梯度下降的优秀的稳定性
  - 随机梯度下降的快速更新
- 小批量梯度下降很适合使用在并行化计算中
  - 将每个小批量数据进一步切分，每个线程分别计算梯度，最后再加和这些梯度



2.2 梯度更新方式

基本搜索步骤

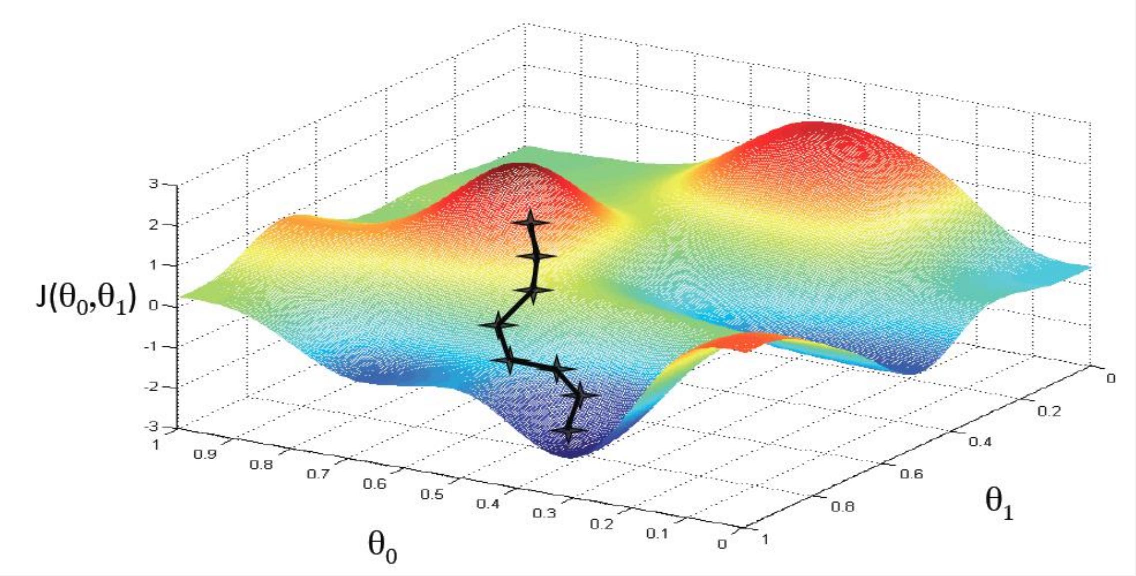
- 随机选择一个参数初始化 θ
- 根据数据和梯度算法来更新 θ
- 直到走到局部一个最小区域(local minimum)



2.2 梯度更新方式

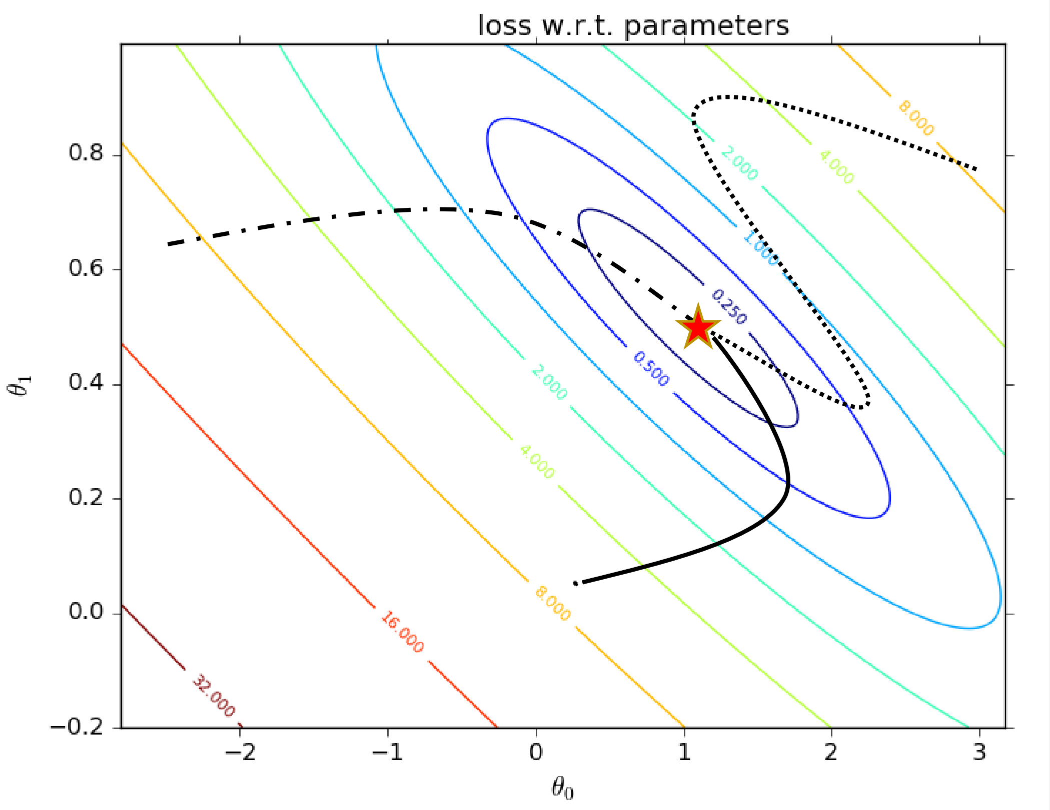
基本搜索步骤

- 随机选择一个新的参数初始化 θ
- 根据数据和梯度算法来更新 θ
- 直到走到局部一个最小区域(local minimum)



2.2 梯度更新方式

凸优化目标函数具有唯一最小点



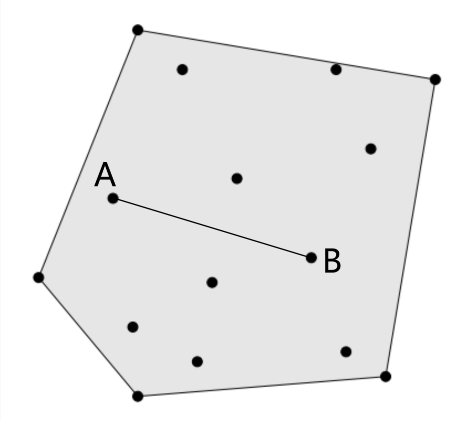
- 不同的初始化参数最终也会学习到相同的最优值

凸集 (Convex Set)

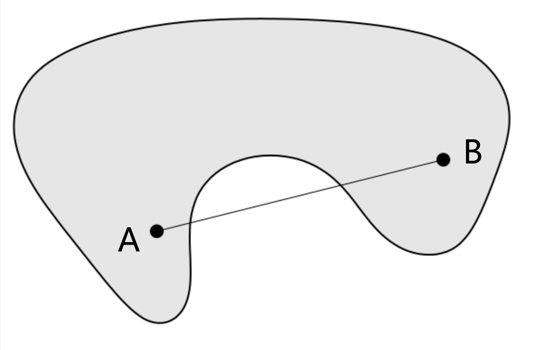
一个点集S被称为凸集，当且仅当该S里的任意两点A和B的连线上任意一点同样属于S

$$tx_1 + (1-t)x_2 \in S$$

$$\text{for all } x_1, x_2 \in S, 0 \le t \le 1$$



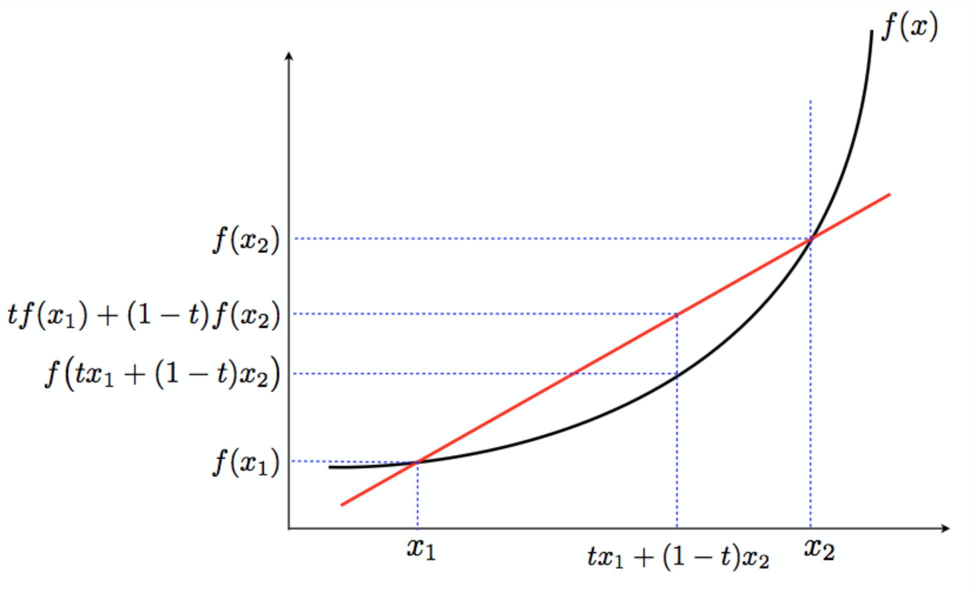
凸集



非凸集

2.2 梯度更新方式

凸函数 (Convex Function)



凸函数的定义

$f:\mathbb{R}^n \to \mathbb{R}$ 是凸函数:　　$\textbf{dom}\, f$ 是一个凸集，并且满足

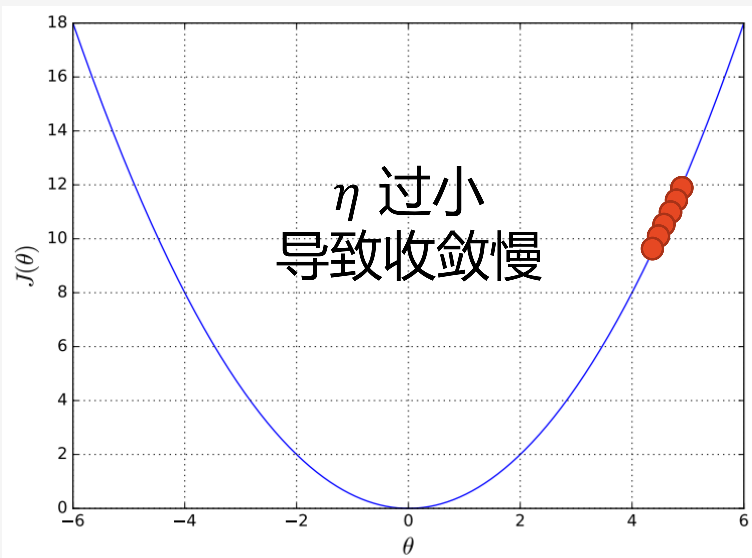
$$f(tx_1+(1-t)x_2) \le tf(x_1)+(1-t)f(x_2)$$

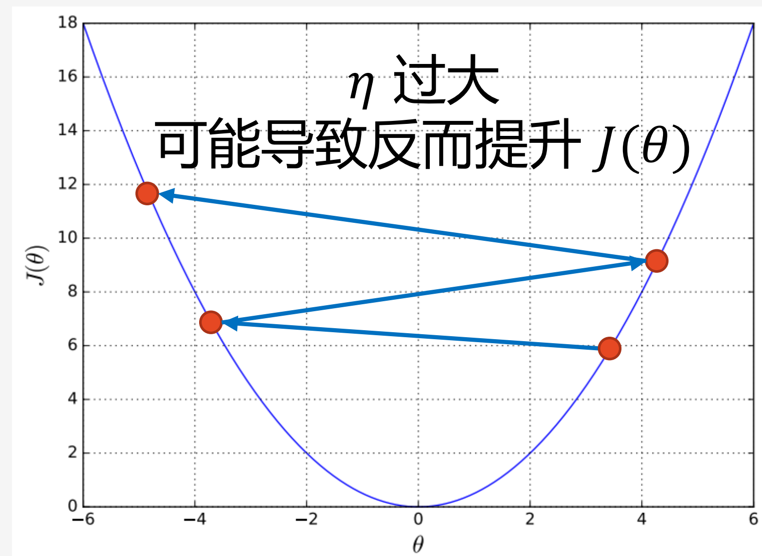
$$\forall x_1, x_2 \in \textbf{dom} f, 0 \le t \le 1$$

2.2 梯度更新方式

学习率的选择

$$\theta_{\text{new}} = \theta_{\text{old}} - \eta \frac{\partial J(\theta)}{\partial \theta}$$





- 初始点可能距离最优点太远，从而导致收敛速度慢

- 可能越过最优点
- 可能无法收敛
- 甚至可能发散

- 要检查梯度下降是否有效工作，可以打印出每几个迭代得到的损失 $J(\theta)$, 如果发现 $J(\theta)$ 并没有正常地下降，调整学习率 η

2.3 线性回归矩阵形式

2.3 线性回归矩阵形式

从代数视角来看线性回归

训练数据矩阵

$$\boldsymbol{X}=\begin{pmatrix}\boldsymbol{x}^{(1)}\\\boldsymbol{x}^{(2)}\\\vdots\\\boldsymbol{x}^{(n)}\end{pmatrix}=\begin{pmatrix}x_1^{(1)} & x_2^{(1)} & \cdots & x_d^{(1)}\\x_1^{(2)} & x_2^{(2)} & & x_d^{(2)}\\ \vdots & & \ddots & \vdots\\x_1^{(n)} & x_2^{(n)} & \cdots & x_d^{(n)}\end{pmatrix}\quad \text{参数 }\boldsymbol{\theta}=\begin{pmatrix}\theta_1\\\theta_2\\\vdots\\\theta_d\end{pmatrix}\quad \text{标签 }\boldsymbol{y}=\begin{pmatrix}y_1\\y_2\\\vdots\\y_n\end{pmatrix}$$

预测

$$\hat{\boldsymbol{y}}=\boldsymbol{X}\boldsymbol{\theta}=\begin{pmatrix}\boldsymbol{x}^{(1)}\boldsymbol{\theta}\\\boldsymbol{x}^{(2)}\boldsymbol{\theta}\\\vdots\\\boldsymbol{x}^{(n)}\boldsymbol{\theta}\end{pmatrix}$$

目标函数

$$J(\boldsymbol{\theta})=\frac{1}{2}(\boldsymbol{y}-\hat{\boldsymbol{y}})^\top(\boldsymbol{y}-\hat{\boldsymbol{y}})=\frac{1}{2}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\theta})^\top(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\theta})$$

2.3 线性回归矩阵形式

目标函数

$$J(\boldsymbol{\theta}) = \frac{1}{2}(\boldsymbol{y} - \boldsymbol{X\theta})^{\top}(\boldsymbol{y} - \boldsymbol{X\theta}) \qquad \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$

对参数向量的梯度

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -\boldsymbol{X}^{\top}(\boldsymbol{y} - \boldsymbol{X\theta})$$

最优参数求解

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 0 \rightarrow \boldsymbol{X}^{\top}(\boldsymbol{y} - \boldsymbol{X\theta}) = 0$$

$$\rightarrow \boldsymbol{X}^{\top}\boldsymbol{y} = \boldsymbol{X}^{\top}\boldsymbol{X\theta}$$

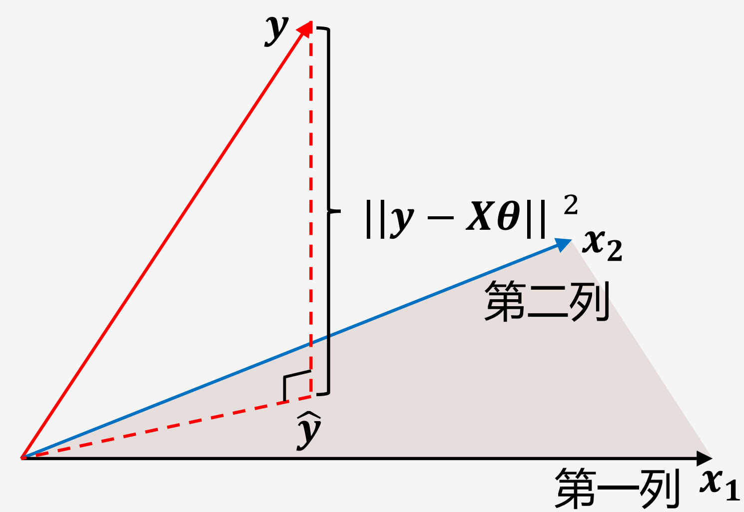
$$\rightarrow \widehat{\boldsymbol{\theta}} = (\boldsymbol{X}^{\top}\boldsymbol{X})^{-1}\boldsymbol{X}^{\top}\boldsymbol{y}$$

2.3 线性回归矩阵形式

预测值

$$\hat{\boldsymbol{y}} = \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top \boldsymbol{y} = \boldsymbol{H}\boldsymbol{y}$$

$\boldsymbol{H}$:帽子矩阵



几何解释

- 数据矩阵的列向量$[\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_d]$张成一个 $\mathbb{R}^n$上的子空间
- $\boldsymbol{H}$就是将标签向量$\boldsymbol{y}$投影到该子空间的映射

$$\boldsymbol{X} = \begin{pmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_d^{(1)} \\ x_1^{(2)} & x_2^{(2)} & & x_d^{(1)} \\ \vdots & & \ddots & \vdots \\ x_1^{(n)} & x_2^{(n)} & \cdots & x_d^{(n)} \end{pmatrix} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_d] \qquad \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

2.3 线性回归矩阵形式

$X^T X$为奇异矩阵的情况

- 当数据矩阵的一些列向量线性相关时
  - 例如$\boldsymbol{x}_2 = 3\boldsymbol{x}_1$
- $\boldsymbol{X}^\top \boldsymbol{X}$为奇异矩阵，所以$\hat{\boldsymbol{\theta}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$无法被直接计算。

解决方案

- 正则化（Regularization）
- $J(\boldsymbol{\mu}) = \frac{1}{2}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})^\top(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}) + \frac{\lambda}{2}||\boldsymbol{\theta}||_2^2$

带正则项的线性回归矩阵形式

优化目标

$$J(\boldsymbol{\theta}) = \frac{1}{2}(\boldsymbol{y} - \boldsymbol{X\theta})^{\top}(\boldsymbol{y} - \boldsymbol{X\theta}) + \frac{\lambda}{2}||\boldsymbol{\theta}||_2^2 \qquad \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$

对参数向量的梯度

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -\boldsymbol{X}^{\top}(\boldsymbol{y} - \boldsymbol{X\theta}) + \lambda\boldsymbol{\theta}$$

最优参数求解

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 0 \rightarrow -\boldsymbol{X}^{\top}(\boldsymbol{y} - \boldsymbol{X\theta}) + \lambda\boldsymbol{\theta} = 0$$

$$\rightarrow \boldsymbol{X}^{\top}\boldsymbol{y} = (\boldsymbol{X}^{\top}\boldsymbol{X} + \lambda\boldsymbol{I})\boldsymbol{\theta}$$

$$\rightarrow \widehat{\boldsymbol{\theta}} = (\boldsymbol{X}^{\top}\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^{\top}\boldsymbol{y}$$

2.4 最大似然估计

2.4 最大似然估计

判别模型

- 建模预测变量和观测变量之间的关系
- 又名条件模型（Conditional Models)
- 确定性判别模型：$y = f_\theta(x)$
- **概率判别模型**：$p_\theta(y|x)$

带高斯白噪声的线性拟合

$$y = f_\theta(x) + \epsilon = \theta_0 + \sum_{j=1}^{d} \theta_j x_j + \epsilon = \theta^\top x + \epsilon$$

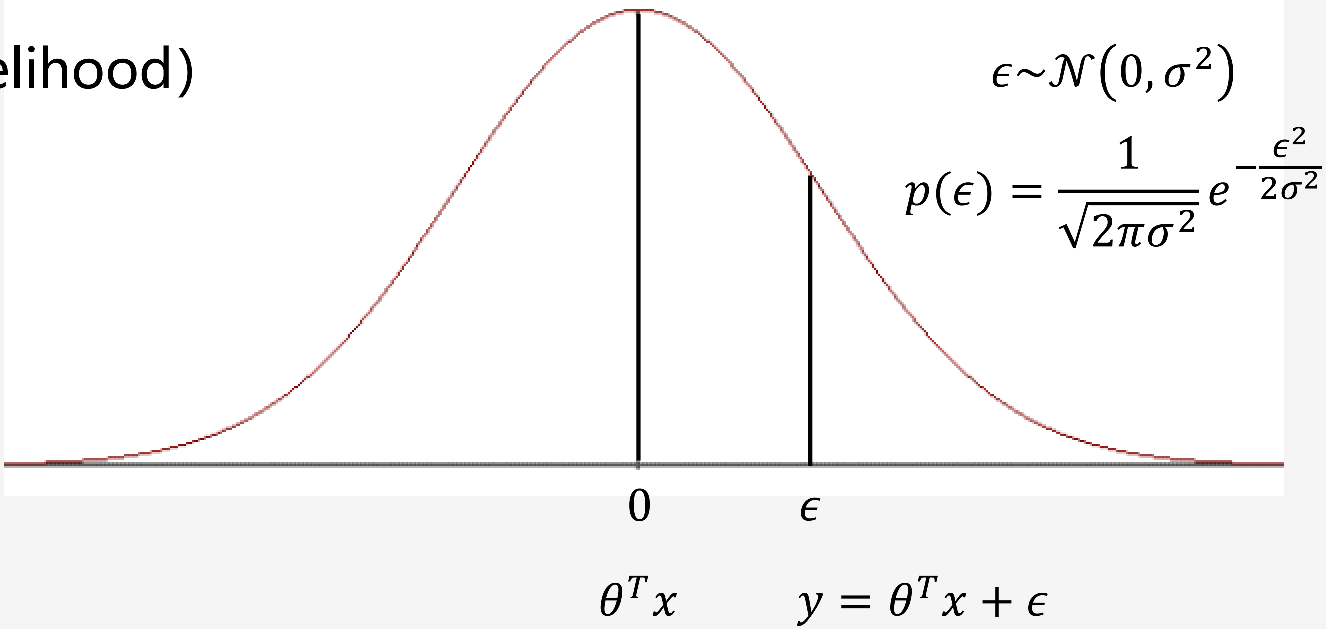
$$\epsilon \sim \mathcal{N}\left(0, \sigma^2\right)$$

$$x = (1, x_1, x_2, \dots, x_d)$$

2.4 最大似然估计

优化目标

最大似然（likelihood）



一个数据点的标签预测似然

$$p(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-\theta^\top x)^2}{2\sigma^2}}$$

2.4 最大似然估计

概率判别模型的学习

最大化训练数据的似然

$$\max_{\theta} \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i-\theta^\top x_i)^2}{2\sigma^2}}$$

最大化训练数据的对数似然

$$\log \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i-\theta^\top x_i)^2}{2\sigma^2}} = \sum_{i=1}^{N} \log \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i-\theta^\top x_i)^2}{2\sigma^2}} = -\sum_{i=1}^{N} \frac{\left(y_i - \theta^\top x_i\right)^2}{2\sigma^2} + \text{const}$$

$$\min_{\theta} \sum_{i=1}^{N} \left(y_i - \theta^\top x_i\right)^2$$

等价于最小均方误差学习

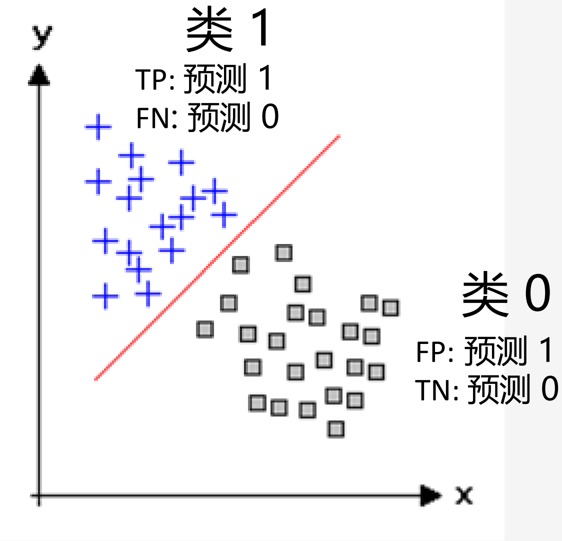
2.5 分类指标

2.5 分类指标

评估指标

	预测	
标签	1	0
1	True Positive	False Negative
0	False Positive	True Negative

- **True / False**
  - True：预测 = 标签
  - False：预测 ≠ 标签
- **Positive / Negative**
  - Positive：预测$y = 1$
  - Negative：预测$y = 0$



2.5 分类指标

评估指标

预测

标签		1	0
	1	True Positive	False Negative
	0	False Positive	True Negative

精度(Accuracy)

- 分类正确的样本占样本总数的比例

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

2.5 分类指标

评估指标

预测

标签		1	0
	1	True Positive	False Negative
	0	False Positive	True Negative

精确率(Precision)

- 预测为1的样本中标签为1的比例

$$\text{Prec} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

预测

标签		1	0
	1	True Positive	False Negative
	0	False Positive	True Negative

召回率(Recall)

- 标签为1的样本中预测为1的比例

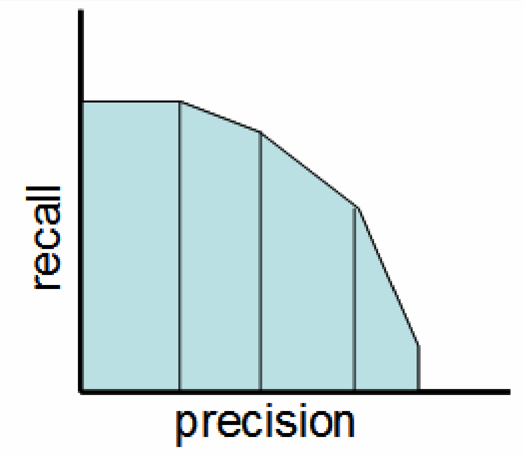
$$\text{Rec} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

2.5 分类指标

评估指标

- 精确率和召回率的权衡

$$\hat{y} = \begin{cases} 1, & p_\theta(y = 1|x) > h \\ 0, & \text{otherwise} \end{cases}$$



- 阈值越高，精确率越高，召回率越低
  - 极端情况：阈值=0.99
- 阈值越低，精确率越低，召回率越高
  - 极端情况：阈值=0

- F1分数

$$\text{F1} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$$

2.5 分类指标

评估指标

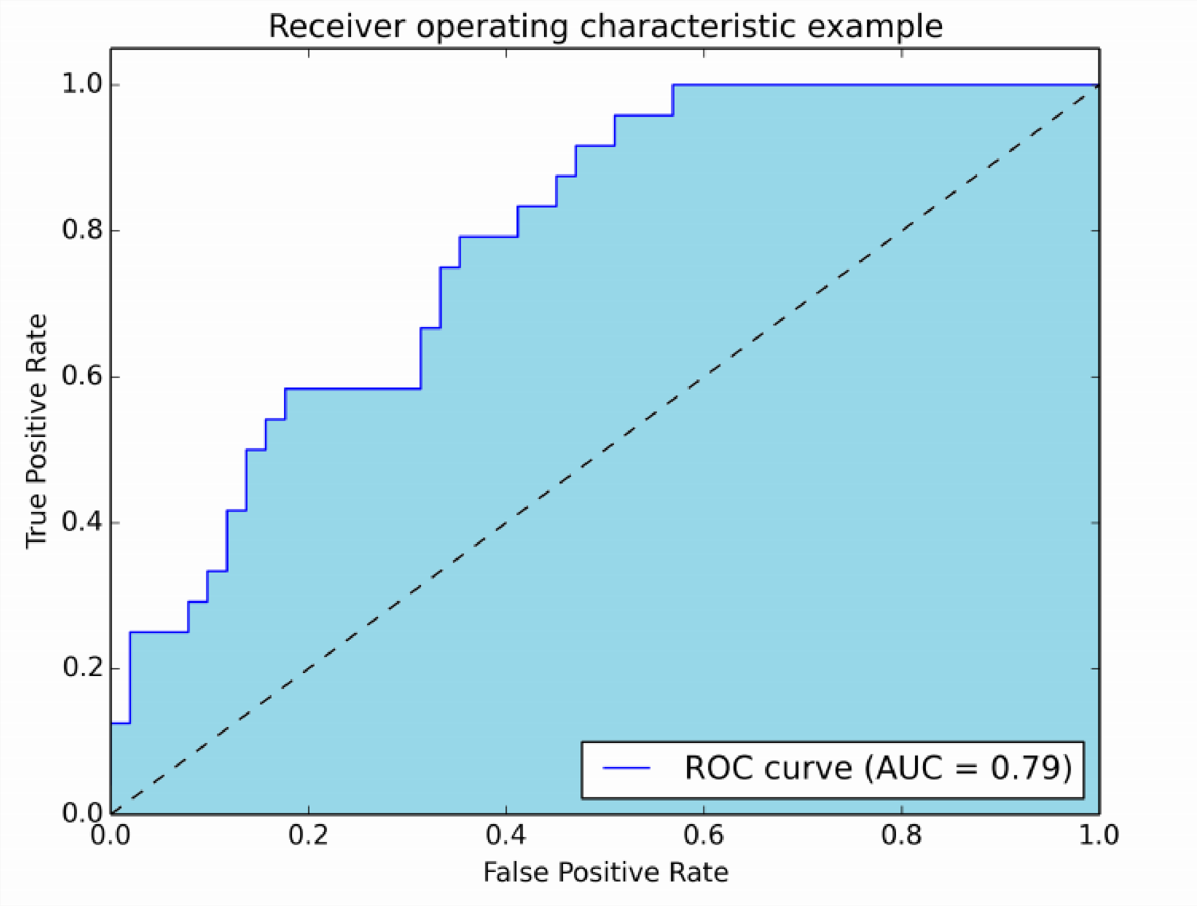
- 基于排序的度量：ROC曲线下面积（AUC）

- True Positive Rate

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

- False Positive Rate

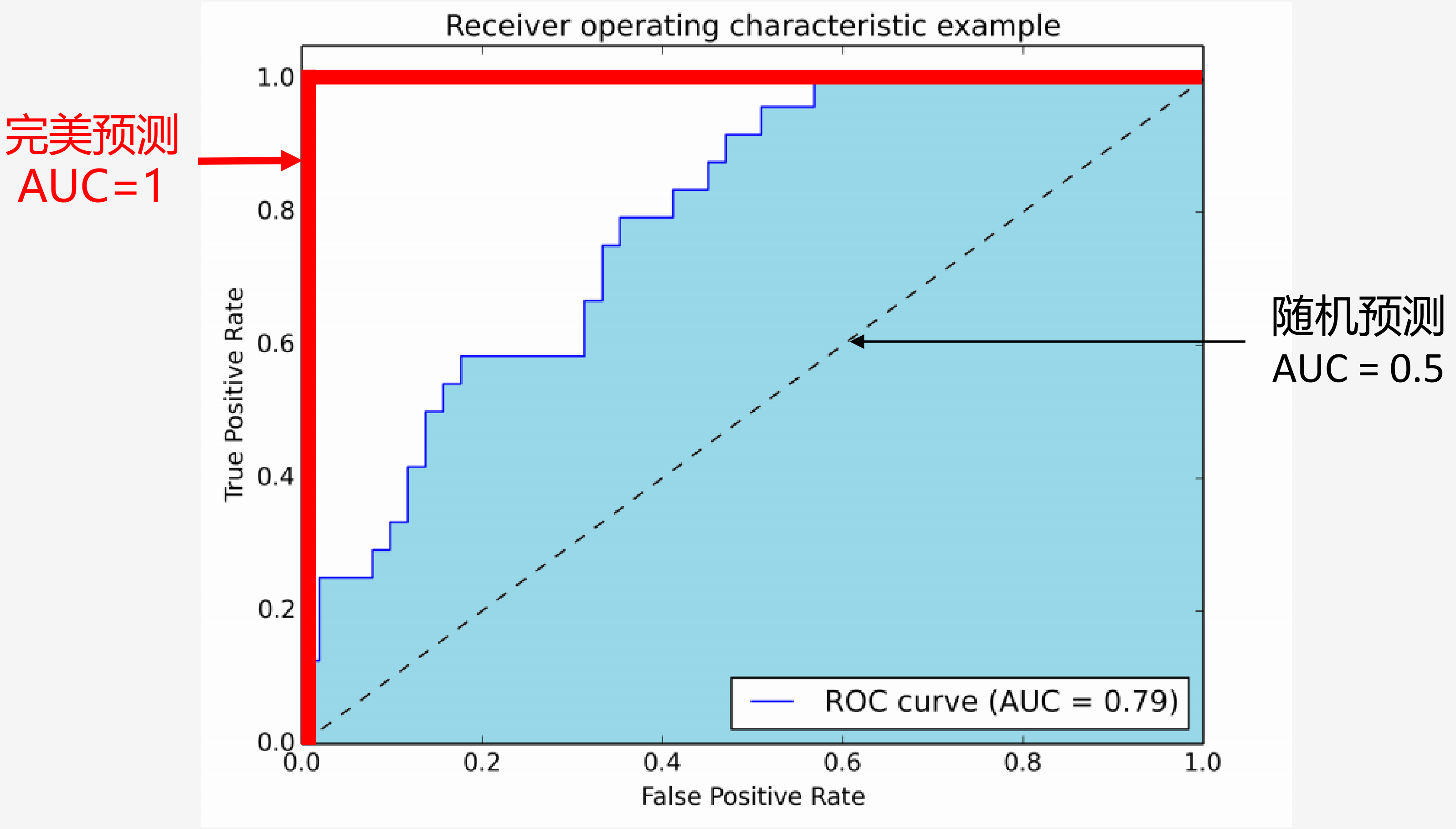
$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}$$



2.5 分类指标

评估指标

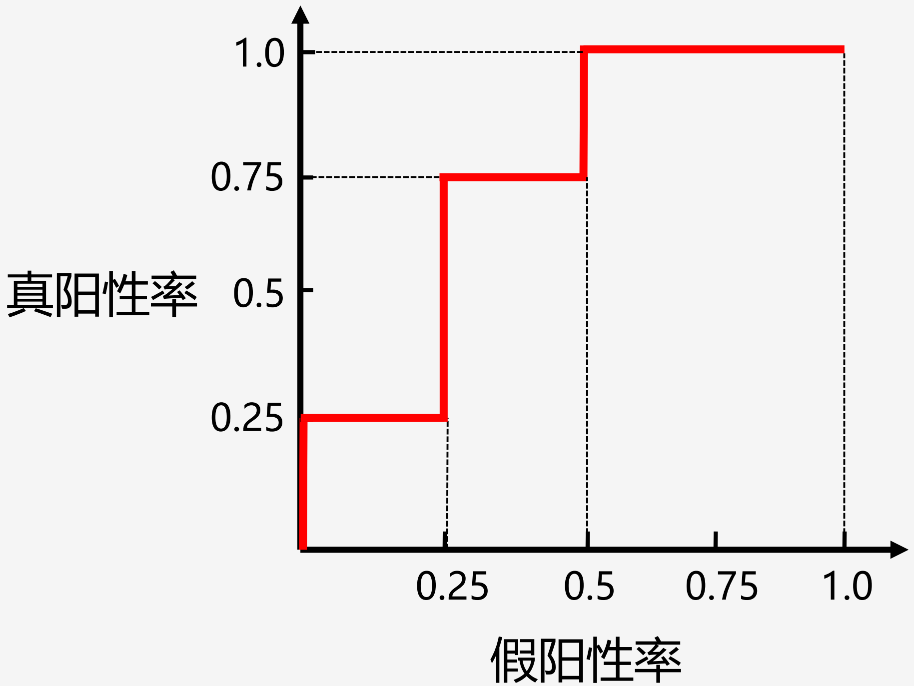
- 基于排序的度量：ROC曲线下面积（AUC）



2.5 分类指标

评估指标

- AUC计算例子



AUC = 0.75

Prediction	Label
0.91	1
0.85	0
0.77	1
0.72	1
0.61	0
0.48	1
0.42	0
0.33	0

2.6 逻辑斯谛回归

2.6 逻辑斯谛回归

分类问题

给定

- 样本空间$\mathbb{X}$中一个样本x ($x \in \mathbb{X}$)的描述
- 一个固定的类别集：$C = \{c_1, c_2, \cdots, c_m\}$

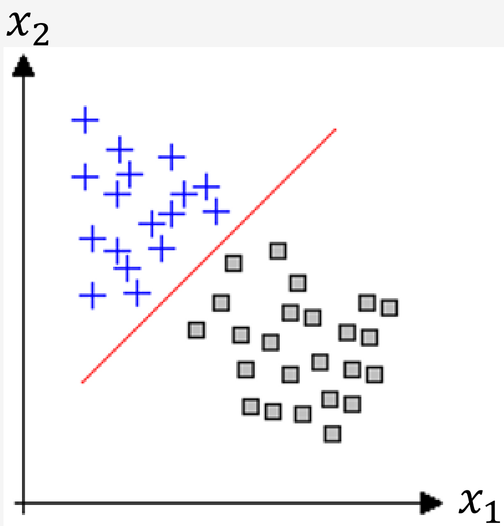
求解

- x的类别:$f(x) \in C$，其中$f(x)$是一个定义域为$\mathbb{X}$，值域为C的类别函数

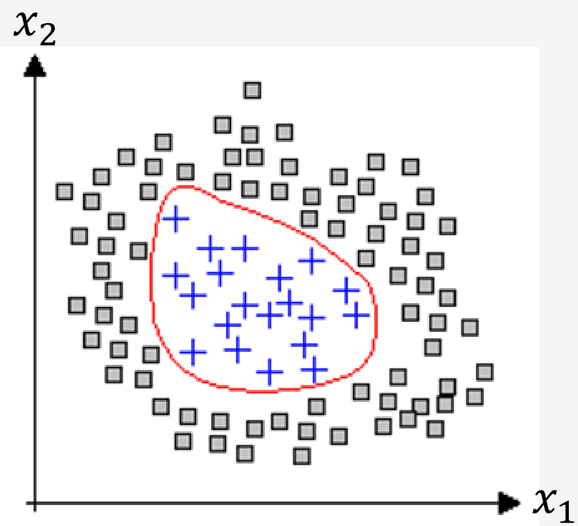
二分类

- 假如类别集是二元的，即$C = \{0,1\}$({错误，正确}，{负，正})，那么这就是二分类问题

二分类



线性可分



线性不可分

线性可分性：是否存在 $ax_1 + bx_2 + c = 0$

使得对于所有的正例：$ax_1 + bx_2 + c > 0$

对于所有的负例：$ax_1 + bx_2 + c < 0$

线性判别模型

判别模型

- 性质
  - 建模预测变量和观测变量之间的关系
  - 也称作条件模型(Conditional Models)
- 分类
  - 确定性判别模型：$y = f_\theta(x)$
    - 对于分类任务不可微分
  - 概率判别模型：$p_\theta(y|x)$
    - 对于分类任务可微分

二分类

$$p_\theta(y = 1|x)$$

$$p_\theta(y = 0|x) = 1 - p_\theta(y = 1|x)$$

熵（Entropy）

- 在信息论中，熵用来衡量一个随机事件的不确定性

- 自信息（Self Information）

$$I(x) = -\log\big(p(x)\big)$$

  - x表示一个事件
  - $p(x)$表示x发生的概率
  - 信息量， x越不可能发生时，它一旦发生后的信息量就越大

熵（Entropy）

- 在信息论中，熵用来衡量一个随机事件的不确定性
- 自信息（Self Information）

$$I(x) = -\log\big(p(x)\big)$$

- 熵的计算

$$\begin{aligned} H(X) &= \mathbb{E}_X[\mathrm{I}(x)] \\ &= \mathbb{E}_X[-\log(p(x))] \\ &= -\sum_{x \in \mathcal{X}} p(x) \log p(x) \end{aligned}$$

2.6 逻辑斯谛回归

熵（Entropy）

- 假设对于这门课程，我们有三种可能的情况发生

事件编号	事件	概率 p	信息量 I
x_1	优秀	$p=0.7$	$I=-\ln(0.7)=0.36$
x_2	及格	$p=0.2$	$I=-\ln(0.2)=1.61$
x_3	不及格	$p=0.1$	$I=-\ln(0.1)=2.30$

- 某某同学不及格！好大的信息量！相比较来说， “优秀”事件的信息量反而小了很多。上面的问题的熵是：

$$
\begin{aligned}
H(p) &= -[p(x_1)\ln p(x_1)+p(x_2)\ln p(x_2)+p(x_3)\ln p(x_3)] \\
&= 0.7\times 0.36+0.2\times 1.61+0.1\times 2.30 \\
&= 0.804
\end{aligned}
$$

2.6 逻辑斯谛回归

损失函数

交叉熵损失

- 离散的情况 $H(p,q) = -\sum_x p(x) \log q(x)$
- 连续的情况 $H(p,q) = -\int_x p(x) \log q(x)\, dx$

分类问题计算交叉熵损失

Ground Truth	0	1	0	0	0
Prediction	0.1	0.6	0.05	0.05	0.2

$$\mathcal{L}(y, x, p_\theta) = -\sum_k \delta(y = c_k) \log p_\theta(y = c_k | x)$$

$$\delta(z) = \begin{cases} 1, & z \text{ is true} \\ 0, & \text{otherwise} \end{cases}$$

2.6 逻辑斯谛回归

二分类的交叉熵

	Class 1	Class 2
真实值	0	1
预测值	0.3	0.7

- 损失函数

$$
\begin{aligned}
\mathcal{L}(y, x, p_\theta) &= -\delta(y=1)\log p_\theta(y=1|x) - \delta(y=0)\log p_\theta(y=0|x) \\
&= -y\log p_\theta(y=1|x) - (1-y)\log\big(1-p_\theta(y=1|x)\big)
\end{aligned}
$$

2.6 逻辑斯谛回归

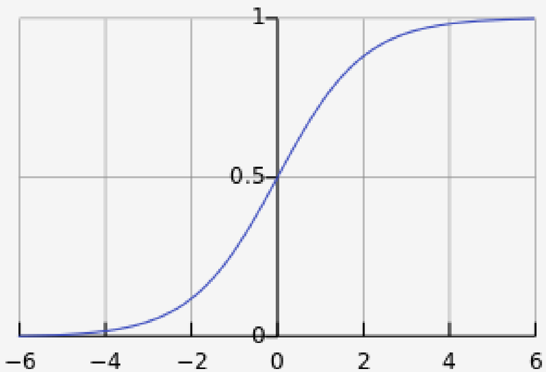
逻辑斯谛(Logistic)回归

- 逻辑斯谛回归是一个二分类模型

$$p_\theta(y=1|x)=\sigma(\theta^\top x)=\frac{1}{1+e^{-\theta^\top x}}$$

$$p_\theta(y=0|x)=\frac{e^{-\theta^\top x}}{1+e^{-\theta^\top x}}$$

Sigmoid函数



- 交叉熵损失函数

$$\mathcal{L}(y,x,p_\theta)=-y\log\sigma(\theta^\top x)-(1-y)\log(1-\sigma(\theta^\top x))$$

- 梯度

$$\frac{\partial\mathcal{L}(y,x,p_\theta)}{\partial\theta}=-y\frac{1}{\sigma(\theta^\top x)}\sigma(z)\big(1-\sigma(z)\big)x-(1-y)\frac{-1}{1-\sigma(\theta^\top x)}\sigma(z)\big(1-\sigma(z)\big)x$$

$$=\big(\sigma(\theta^\top x)-y\big)x$$

$$\theta\leftarrow\theta+\eta\Big(y-\sigma(\theta^\top x)\Big)x$$

$$\boxed{\frac{\partial\sigma(z)}{\partial z}=\sigma(z)(1-\sigma(z))}$$

线性回归：$\theta_{\text{new}}=\theta_{\text{old}}+\eta\big(y_i-f_\theta(x_i)\big)x_i$

标签的决定

- 逻辑斯谛回归求出的概率

$$p_\theta(y=1|x)=\delta(\theta^\top x)=\frac{1}{1+e^{-\theta^\top x}}$$

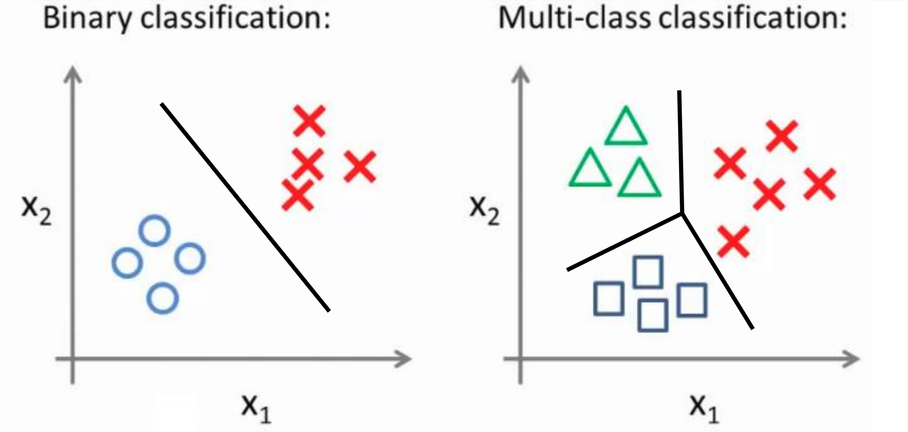
$$p_\theta(y=0|x)=\frac{e^{-\theta^\top x}}{1+e^{-\theta^\top x}}$$

- 设置阈值(threshold)h决定示例最终标签

$$\hat{y}=\begin{cases}1, & p_\theta(y=1|x)>h\\ 0, & \text{otherwise}\end{cases}$$

2.6 逻辑斯谛回归

多分类



多分类交叉熵

$$\mathcal{L}(y, x, p_\theta) = -\sum_k \delta(y = c_k) \log p_\theta(y = c_k | x)$$

$$\delta(z) = \begin{cases} 1, & z \text{ is true} \\ 0, & \text{otherwise} \end{cases}$$

真实值	0	1	0
预测值	0.1	0.7	0.2

2.6 逻辑斯谛回归

多类别逻辑斯谛回归

- 类别集

$$C = \{c_1, c_2, \cdots, c_m\}$$

- 预测$p_\theta(y = c_j|x)$的概率

$$p_\theta(y = c_j|x) = \frac{e^{\theta_j^\top x}}{\sum_{k=1}^{m} e^{\theta_k^\top x}} \quad \text{for } j = 1, \cdots, m$$

- Softmax
  - 参数$\theta = \{\theta_1, \theta_2, \cdots, \theta_m\}$
  - 可以标准化成$m - 1$组参数

多类别逻辑斯谛回归

- 对一个示例的学习 $(x, y = c_j)$
  - 最大对数化似然(log-likelihood)

$$\max_{\theta} \log p_\theta(y = c_j|x)$$

  - 梯度

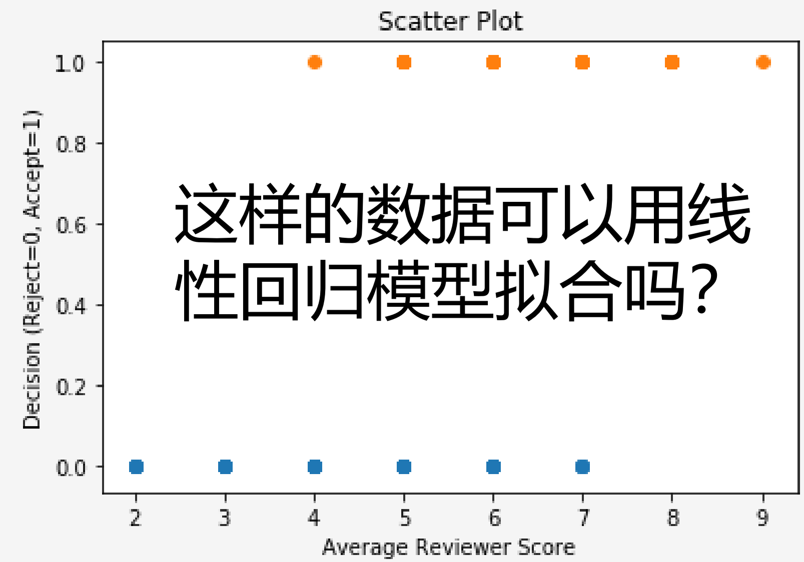
$$\begin{aligned}
\frac{\partial \log p_\theta\left(y = c_j \middle| x\right)}{\partial \theta_j} &= \frac{\partial}{\partial \theta_j} \log \frac{e^{\theta_j^\top x}}{\sum_{k=1}^m e^{\theta_k^\top x}} \\
&= x - \frac{\partial}{\partial \theta_j} \log \sum_{k=1}^m e^{\theta_k^\top x} \\
&= x - \frac{e^{\theta_j^\top x}}{\sum_{k=1}^m e^{\theta_k^\top x}} x = \left(1 - p_\theta\left(y = c_j \middle| x\right)\right) x
\end{aligned}$$

线性模型应用举例

论文审稿结果分类

二分类任务：

- 论文平均得分 $x \in \Re$， $y \in \{0,1\}$
- ICLR'18审稿意见数据集



这样的数据可以用线性回归模型拟合吗?

	TL;DR	_bibtex	abstract	authorids	authors	conf_1	conf_2	conf_3	decision	review	review_1	review_2	review_3	title
	None	@article{\nsharma2018hyperedge2vec:,\ntitle={H...	Data structured in form of overlapping or non-...	[sharm170@umn.edu, srjoty@ntu.edu.sg, himanshu...	[Ankit Sharma, Shafiq Joty, Himanshu Kharkwal,...	3.0	3.0	[illegible]	Reject	5.000000	5.0	5.0	5.0	Hyperedge2vec: Distributed Representations for...
	Query-based black-box attacks on deep neural n...	@article{\nnitin2018exploring,\ntitle={Explori...	Existing black-box attacks on deep neural netw...	[abhagoji@princeton.edu, _w@eecs.berkeley.edu,...	[Arjun Nitin Bhagoji, Warren He, Bo Li, Dawn S...	4.0	3.0	4.0	Reject	6.000000	5.0	6.0	7.0	Exploring the Space of Black-box Attacks on De...
	A theory and algorithmic framework for predict...	@article{\nd.2018learning,\ntitle={Learning We...	Predictive models that generalize well under d...	[fredrikj@mit.edu, kallus@cornell.edu, urish22...	[Fredrik D. Johansson, Nathan Kallus, Uri Shal...	3.0	3.0	4.0	Reject	6.666667	5.0	8.0	7.0	Learning Weighted Representations for Generali...
	We prove that DNN is a recursively approximate...	@article{\nzheng2018understanding,\ntitle={Und...	Deep learning achieves remarkable generalizati...	[zhenggh@mail.ustc.edu.cn, jtsang@bjtu.edu.cn,...	[Guanhua Zheng, Jitao Sang, Changsheng Xu]	3.0	3.0	[illegible]	Reject	3.666667	2.0	3.0	6.0	Understanding Deep Learning Generalization by

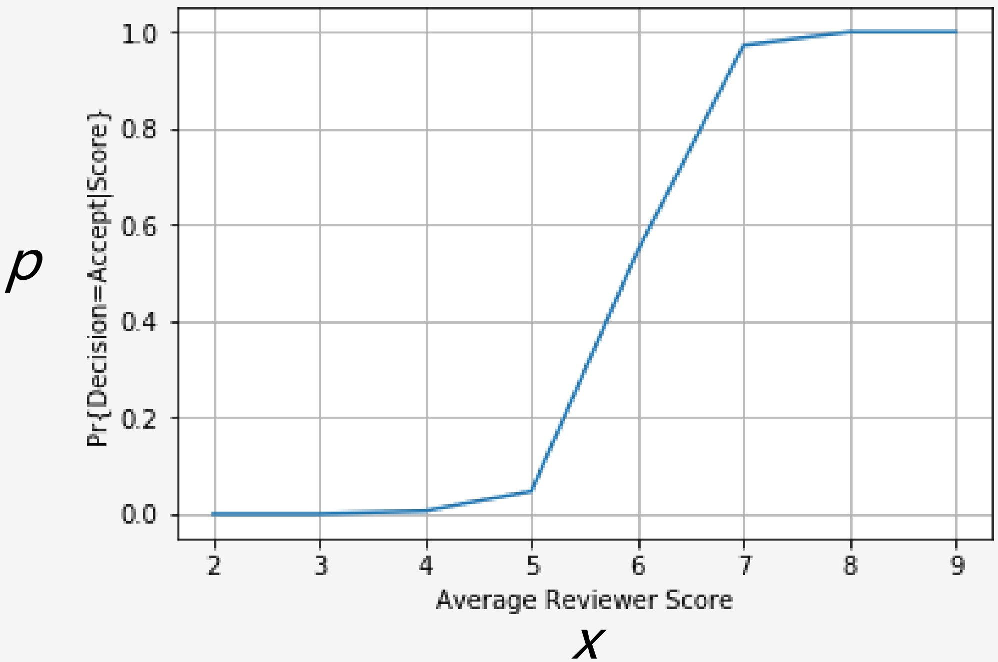
线性模型应用举例

论文审稿结果分类

二分类任务：

Pr{审稿结果 = 接收 | 分数}

- 试试计算并绘制 $p = \Pr\{y = 1 | x\}$



- 用线性回归将 p 拟合成 x 的一个函数?

$$p = \theta_0 + \theta_1 x$$

- 拟合效果会如何?
  - 概率 p 总是保持在 [0, 1] 之间

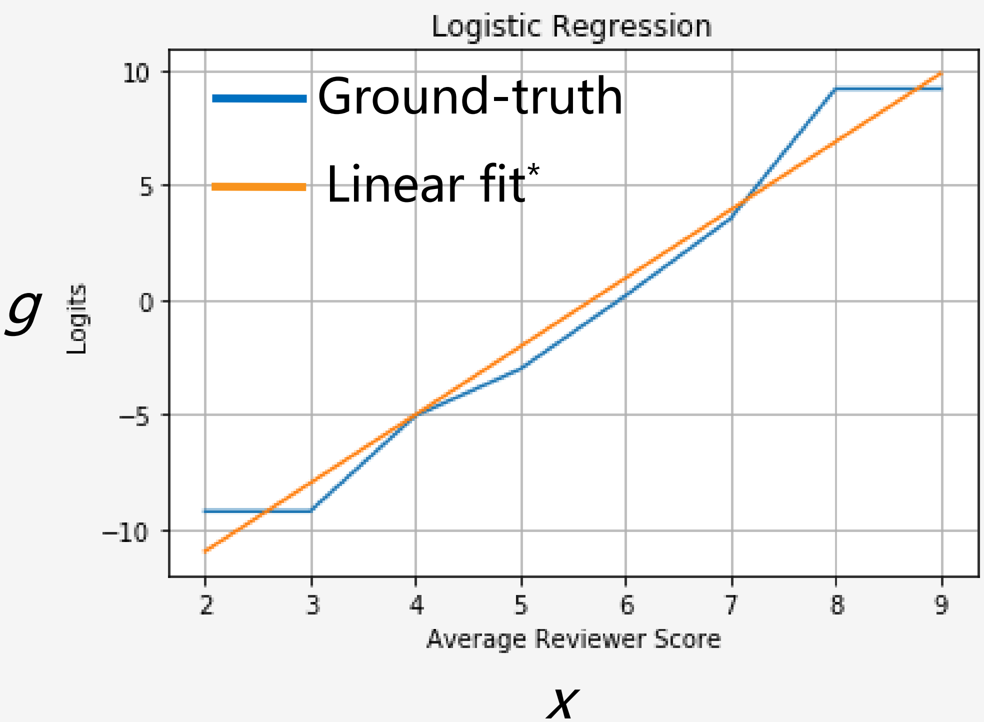
线性模型应用举例

论文审稿结果分类

二分类任务：

Logit function

- 试试建立一个以概率 p 为自变量的逻辑函数 $g = \log(\frac{p}{1-p})$



- g 的范围是多少?

$$g \in [-\infty, \infty]$$

- 逻辑斯蒂回归的实质：用线性回归去拟合logit function

$$g = \log(\frac{p}{1-p}) = \theta_0 + \theta_1 x$$

线性模型应用举例

论文审稿结果分类

二分类任务：

Pr{审稿结果 = 接收 | 分数}

$$g = \log\left(\frac{p}{1-p}\right) = \theta_0 + \theta_1 x \quad \Rightarrow \quad p = \frac{1}{1+e^{-(\theta_0+\theta_1 x)}}$$

- Pr{审稿结果 = 拒稿 | 分数} 如何计算?

$$1-p = \frac{e^{-(\theta_0+\theta_1 x)}}{1+e^{-(\theta_0+\theta_1 x)}}$$

- 如何确定最优的模型参数 θ_0 和 θ_1 ?

线性模型应用举例

论文审稿结果分类

二分类任务：

- 用最大似然估计（Maximum Likelihood Estimation）来确定参数
  - 假设已知模型参数（即 θ_0 和 θ_1），请考虑下面的训练数据集。该数据集来自我们模型的可能性有多大?

#	X	Y
1	$x_1 = 3$	$y_1 = 0$
2	$x_2 = 8$	$y_2 = 1$
..		
..		
N	$x_N = 6$	$y_N = 1$

$$Likelihood = \frac{e^{-(\theta_0 + 3\theta_1)}}{1 + e^{-(\theta_0 + 3\theta_1)}} * \frac{1}{1 + e^{-(\theta_0 + 8\theta_1)}} * \cdots \frac{1}{1 + e^{-(\theta_0 + 6\theta_1)}}$$

线性模型应用举例

论文审稿结果分类

二分类任务:

- 用最大似然估计（Maximum Likelihood Estimation）来确定参数
  - 假设已知模型参数（即 θ_0 和 θ_1），请考虑下面的训练数据集。该数据集来自我们模型的可能性有多大?

#	X	Y
1	$x_1 = 3$	$y_1 = 0$
2	$x_2 = 8$	$y_2 = 1$
..		
..		
N	$x_N = 6$	$y_N = 1$

$g(\theta_0, \theta_1)$ 仅和模型参数有关的函数

$$\downarrow$$

$$Log - Likelihood$$

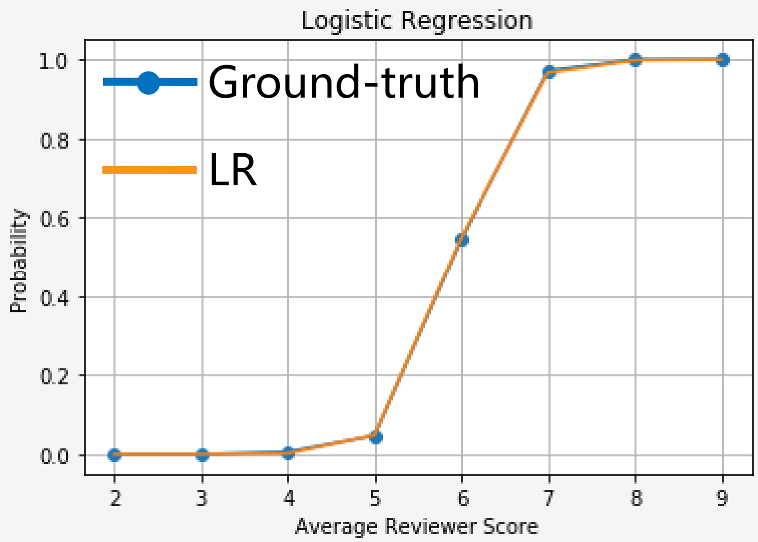
$$= \log\left(\frac{e^{-(\theta_0+3\theta_1)}}{1+e^{-(\theta_0+3\theta_1)}}\right) + \log\left(\frac{1}{1+e^{-(\theta_0+8\theta_1)}}\right) + \ldots \log\left(\frac{1}{1+e^{-(\theta_0+6\theta_1)}}\right)$$

- 找到能够最大化 g 的模型参数 θ_0 和 θ_1，即最大似然估计

线性模型应用举例

论文审稿结果分类

二分类任务：




```python
from sklearn import linear_model

#Instantiate an LR object
logreg = sklearn.linear_model.LogisticRegression(C=1e5);

#Recall: your training data must have a column of ones for the constant term
xd = np.ones((numPapers,2));
xd[:,0] = np.append(rscores,ascores)

yd = np.append(rlabels,alabels);

logreg.fit(xd,yd);

#Plot Pr{Accept|Score}
rv = np.ones((len(revRange),2));
rv[:,0] = revRange;
prpredict=logreg.predict_proba(rv)
```

- 从回归到分类：如果接收的概率 > 0.5，则输出审稿结果为“接收”。

线性模型应用举例

垃圾邮件分类

二分类任务：

- 邮件x ∈ 所有邮件，y ∈ {垃圾邮件，非垃圾邮件}

$$f: x \rightarrow y$$

2024/10/11 (周五) 15:34

alarm@raysyun.com 代表 人力资源 <hr@hoau.net>

关于2024申请通知!

收件人

已禁用此邮件中的链接和其他功能。若要启用该功能，请将此邮件移动到收件箱中。
已将此邮件转换为纯文本格式。
Outlook 禁止访问下列具有潜在不安全因素的附件: 2024.docx.

请及时查阅，逾期被视为放弃领取资格

此为加密文件（查阅码 2508）

垃圾邮件?

经验：

- 所有已被标识的“垃圾邮件”和“非垃圾邮件”，即“有标签的训练数据集”

性能：

- “垃圾邮件”的识别准确率

“监督学习（分类问题）”

线性模型应用举例

垃圾邮件分类

二分类任务：

- UCI 垃圾邮件数据集 https://archive.ics.uci.edu/ml/datasets/Spambase

Attribute Information:

The last column of 'spambase.data' denotes whether the e-mail was considered spam (1) or not (0), i.e. unsolicited commercial e-mail. Most of the attributes indicate whether a particular word or character was frequently occuring in the e-mail. The run-length attributes (55-57) measure the length of sequences of consecutive capital letters. For the statistical measures of each attribute, see the end of this file. Here are the definitions of the attributes:

48 continuous real [0,100] attributes of type word_freq_WORD
= percentage of words in the e-mail that match WORD, i.e. 100 * (number of times the WORD appears in the e-mail) / total number of words in e-mail. A "word" in this case is any string of alphanumeric characters bounded by non-alphanumeric characters or end-of-string.

6 continuous real [0,100] attributes of type char_freq_CHAR]
= percentage of characters in the e-mail that match CHAR, i.e. 100 * (number of CHAR occurences) / total characters in e-mail

1 continuous real [1,...] attribute of type capital_run_length_average
= average length of uninterrupted sequences of capital letters

1 continuous integer [1,...] attribute of type capital_run_length_longest
= length of longest uninterrupted sequence of capital letters

1 continuous integer [1,...] attribute of type capital_run_length_total
= sum of length of uninterrupted sequences of capital letters
= total number of capital letters in the e-mail

1 nominal {0,1} class attribute of type spam
= denotes whether the e-mail was considered spam (1) or not (0), i.e. unsolicited commercial e-mail.

- 57 个实数值或整数值特征
- 二元输出类别

- 用多参数的逻辑斯蒂回归拟合

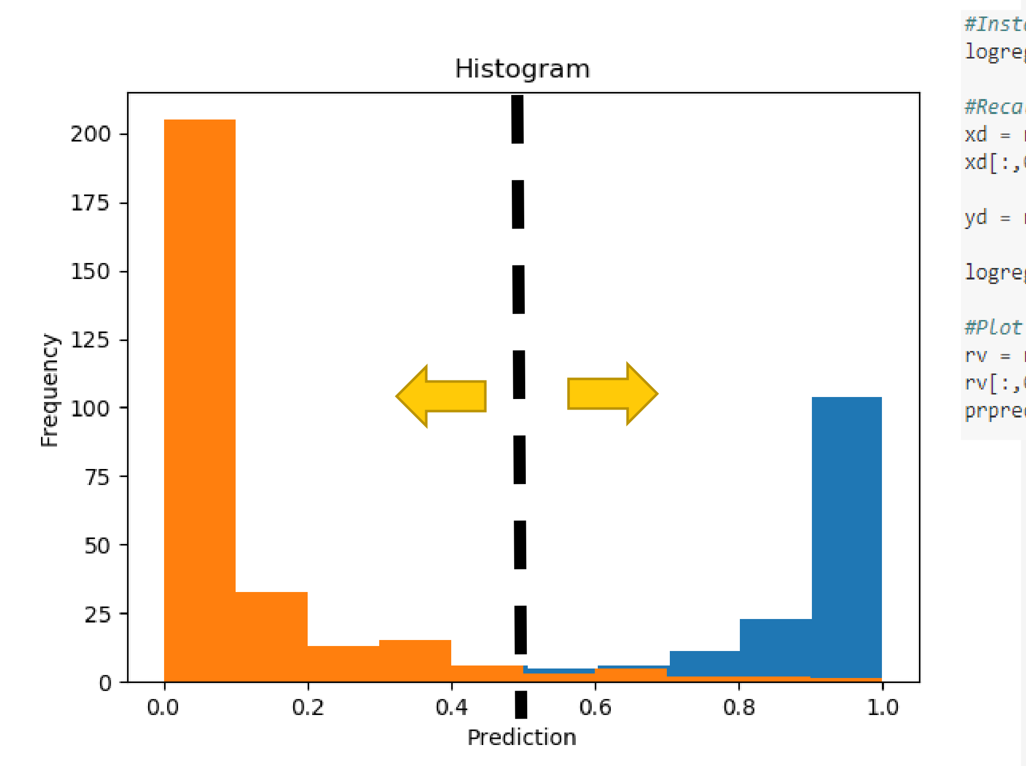
$$p_{spam} = \frac{1}{1+e^{-(\theta_0+\sum_{i=1}^{M}\theta_i x_i)}}$$

线性模型应用举例

垃圾邮件分类

二分类任务：

- UCI 垃圾邮件数据集 https://archive.ics.uci.edu/ml/datasets/Spambase
- 90%用作训练数据集，10%用作测试数据集




```
#Instantiate an LR object
logreg = sklearn.linear_model.LogisticRegression(C=1e5);

#Recall: your training data must have a column of ones for the constant term
xd = np.ones((numPapers,2));
xd[:,0] = np.append(rscores,ascores)

yd = np.append(rlabels,alabels);

logreg.fit(xd,yd);

#Plot Pr{Accept|Score}
rv = np.ones((len(revRange),2));
rv[:,0] = revRange;
prpredict=logreg.predict_proba(rv)
```

哪些邮件被误分类了?

垃圾邮件分类准确率：~92%

线性模型应用举例

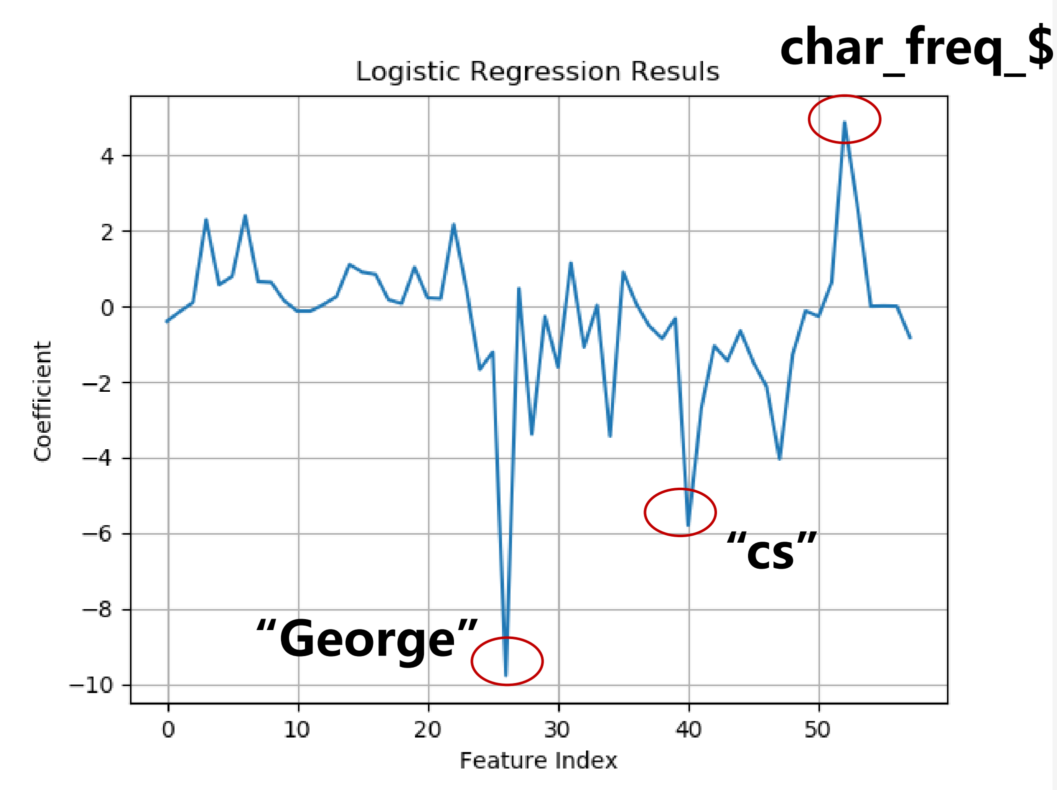
垃圾邮件分类

二分类任务：

- 哪些特征更加重要?
- θ_i 接近于0，代表着什么?

$$p_{spam} = \frac{1}{1+e^{-(\theta_0+\sum_{i=1}^{M}\theta_i x_i)}}$$

- 合理的假设是：具有较大绝对值的 θ_i 系数对应的特征对模型预测性能更重要

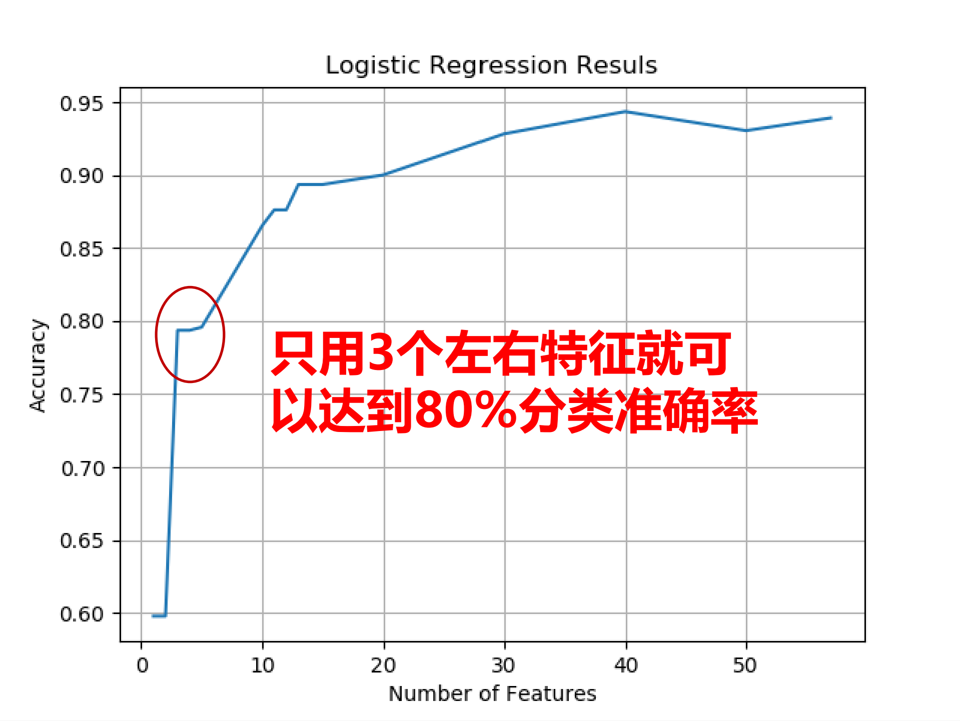


线性模型应用举例

垃圾邮件分类

二分类任务：

- 只用绝对值排名靠前的k个特征，重新训练和评估模型性能



- 是否可以显式地训练参数，以优先构建一个“稀疏”的模型？为什么？
  - 低复杂度的模型不容易过拟合

- 模型训练的目的是为了最小化损失函数：

$$\hat{\theta} = \min_{\theta} Loss(\theta)$$

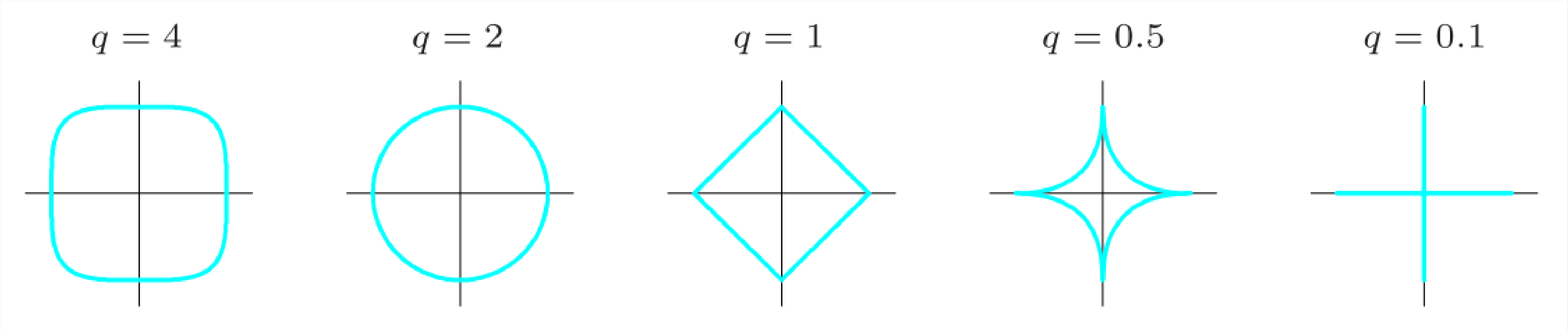
如何改变这个损失函数以降低训练出来的模型的复杂度？

线性模型应用举例

垃圾邮件分类

二分类任务：

- 在损失函数中引入正则化项
- 经典正则化方法$\|\theta\|_q$的数值分布图



Ridge　　LASSO

“正则化”后的损失函数

$$\hat{\theta} = \min_{\theta}\{Loss(\theta) + c\|\theta\|_0\}$$

c控制正则化项的相对重要性

线性模型应用举例

垃圾邮件分类

二分类任务：

- 在损失函数中引入正则化项

Logistic Regression (aka logit, MaxEnt) classifier.

In the multiclass case, the training algorithm uses the one-vs-rest (OvR) scheme if the 'multi_class' option is set to 'ovr', and uses the cross- entropy loss if the 'multi_class' option is set to 'multinomial'. (Currently the 'multinomial' option is supported only by the 'lbfgs', 'sag' and 'newton-cg' solvers.)

This class implements regularized logistic regression using the 'liblinear' library, 'newton-cg', 'sag' and 'lbfgs' solvers. It can handle both dense and sparse input. Use C-ordered arrays or CSR matrices containing 64-bit floats for optimal performance; any other input format will be converted (and copied).

The 'newton-cg', 'sag', and 'lbfgs' solvers support only L2 regularization with primal formulation. The 'liblinear' solver supports both L1 and L2 regularization, with a dual formulation only for the L2 penalty.

Read more in the User Guide.

Parameters: **penalty** : str, 'l1' or 'l2', default: 'l2'

Used to specify the norm used in the penalization. The 'newton-cg', 'sag' and 'lbfgs' solvers support only l2 penalties.

New in version 0.19: l1 penalty with SAGA solver (allowing 'multinomial' + L1)

C : float, default: 1.0

Inverse of regularization strength; must be a positive float. Like in support vector machines, smaller values specify stronger regularization.

应该使用哪一种正则化方法?

c 应该如何选择?

线性模型应用举例

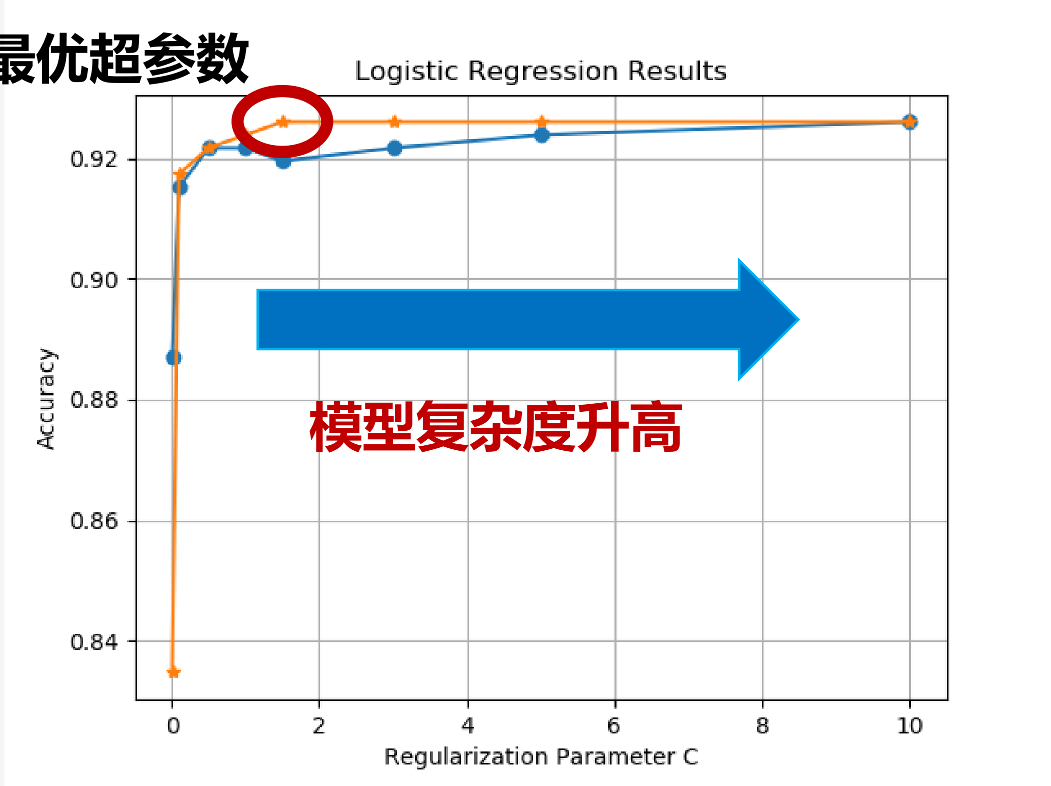
垃圾邮件分类

二分类任务：

- 应该使用哪一种正则化方法?
- c 应该如何选择?

Ridge (L2)

Lasso (L1)



总结

线性模型总结

- 线性回归是机器学习中最基础的参数化学习模型，线性回归任务是机器学习中最基础的有监督学习任务。

- 逻辑斯谛回归，虽然其名字包含“回归”二字，但它是最具有代表性的机器学习分类模型，至今还在学术研究和工业落地场景中被广泛使用。

	激活函数	损失函数	优化方法
线性回归	-	$(y-\mathbf{w}^{\mathrm{T}}\mathbf{x})^2$	最小二乘、梯度下降
Logistic 回归	$\sigma(\mathbf{w}^{\mathrm{T}}\mathbf{x})$	$\mathbf{y}\log\sigma(\mathbf{w}^{\mathrm{T}}\mathbf{x})$	梯度下降
Softmax 回归	$\mathrm{softmax}(W^{\mathrm{T}}\mathbf{x})$	$\mathbf{y}\log\mathrm{softmax}(W^{\mathrm{T}}\mathbf{x})$	梯度下降

西安交通大學
XI'AN JIAOTONG UNIVERSITY

谢谢大家！