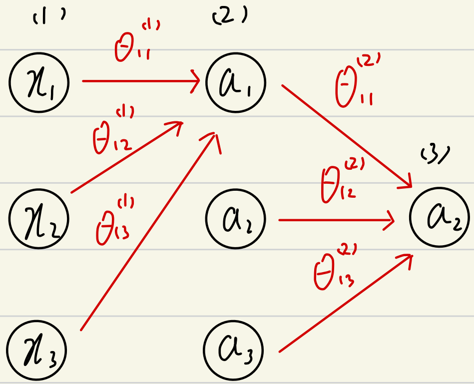
神经网络:



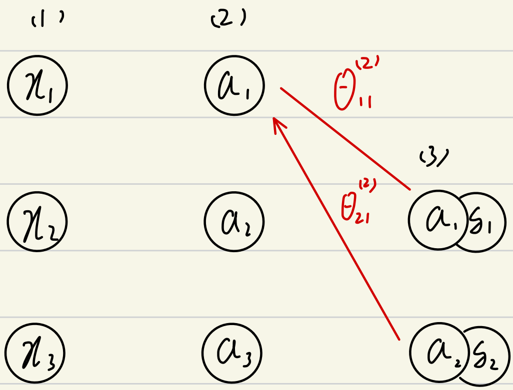
正向传播 (1)→(2)

Single: $z^{(2)} = x_1^{(1)}\theta_{11}^{(1)} + x_2^{(1)}\theta_{12}^{(1)} + x_3^{(1)}\theta_{13}^{(1)} + b_1$

$$= \begin{bmatrix}\theta_{11}^{(1)} & \theta_{12}^{(1)} & \theta_{13}^{(1)}\end{bmatrix}\begin{bmatrix}x_1^{(1)}\\ x_2^{(1)}\\ x_3^{(1)}\end{bmatrix} + b_1$$

all: $$\begin{bmatrix}\theta_{11}^{(1)} & \theta_{12}^{(1)} & \theta_{13}^{(1)}\\ \theta_{21}^{(1)} & \theta_{22}^{(1)} & \theta_{23}^{(1)}\\ \theta_{31}^{(1)} & \theta_{32}^{(1)} & \theta_{33}^{(1)}\end{bmatrix}\begin{bmatrix}x_1^{(1)}\\ x_2^{(1)}\\ x_3^{(1)}\end{bmatrix} + \begin{bmatrix}b_1^{(1)}\\ b_2^{(1)}\\ b_3^{(1)}\end{bmatrix} = \begin{bmatrix}z_1^{(2)}\\ z_2^{(2)}\\ z_3^{(2)}\end{bmatrix}$$

$$\theta^{(1)}x + b^{(1)} = z^{(2)},\quad g(z) = a$$



反向传播: (3)→(2)

$$\delta^{(3)} = (a^{(3)} - y)\, g'(z^{(3)})$$

Single: $\delta_1^{(2)} = \delta_1^{(3)}\theta_{11}^{(2)} g'(z_1^{(2)}) + \delta_2^{(2)}\theta_{21}^{(2)} g'(z_1^{(2)})$

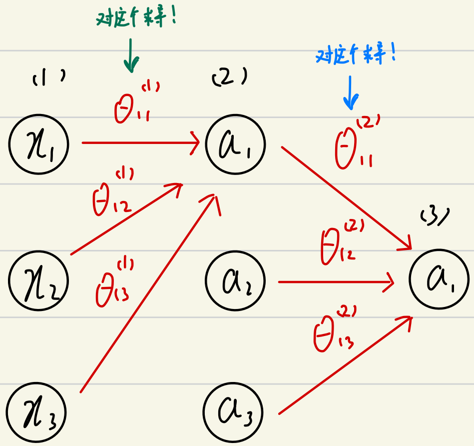
$$= \begin{bmatrix}\theta_{11}^{(2)} & \theta_{21}^{(2)}\end{bmatrix}\begin{bmatrix}\delta_1^{(3)}\\ \delta_2^{(3)}\end{bmatrix} \times g'(z_1^{(2)})$$

all: $$\begin{bmatrix}\theta_{11}^{(2)} & \theta_{21}^{(2)}\\ \theta_{12}^{(2)} & \theta_{22}^{(2)}\\ \theta_{13}^{(2)} & \theta_{23}^{(2)}\end{bmatrix}\begin{bmatrix}\delta_1^{(3)}\\ \delta_2^{(3)}\end{bmatrix} \times \begin{bmatrix}g'(z_1^{(2)})\\ g'(z_2^{(2)})\\ g'(z_3^{(2)})\end{bmatrix} = \begin{bmatrix}\delta_1^{(2)}\\ \delta_2^{(2)}\\ \delta_3^{(2)}\end{bmatrix}$$

$$\theta^{(2)T}\delta^{(2)} \times g'(z^{(2)}) = \delta^{(2)}$$

偏导: $\frac{\partial}{\partial \theta_{ij}^{(l)}} J(\theta) = a_j^{(l)}\delta_i^{(l+1)}$ $\quad \frac{\partial}{\partial b_i^{(l)}} J(\theta) = \delta_i^{(l+1)}$

$$\frac{\partial}{\partial \theta^{(1)}} J(\theta) = \begin{bmatrix}x_1\delta_1^{(2)} & x_2\delta_1^{(2)} & x_3\delta_1^{(2)}\\ x_1\delta_2^{(2)} & x_2\delta_2^{(2)} & x_3\delta_2^{(2)}\\ x_1\delta_3^{(2)} & x_2\delta_3^{(2)} & x_3\delta_3^{(2)}\end{bmatrix} = \begin{bmatrix}\delta_1^{(2)}\\ \delta_2^{(2)}\\ \delta_3^{(2)}\end{bmatrix}\begin{bmatrix}x_1 & x_2 & x_3\end{bmatrix}$$



① $J(\theta) \approx \frac{1}{2}(a^{(3)} - y)^2$

$$\frac{dJ(\theta)}{da^{(3)}} = a^{(3)} - y$$

② $a^{(3)} = g(z^{(3)})$

$$\frac{da^{(3)}}{dz^{(3)}} = g'(z^{(3)}) = a^{(3)}(1 - a^{(3)})$$

③ $z^{(3)} = a_1^{(2)}\theta_{11}^{(2)} + a_2^{(2)}\theta_{12}^{(2)} + a_3^{(2)}\theta_{13}^{(2)} + b$

$$\frac{dz^{(3)}}{d\theta_{11}^{(2)}} = a_1^{(2)} + 0 + 0 + 0$$

④ $$\frac{dJ(\theta)}{d\theta_{11}^{(2)}} = \frac{dJ(\theta)}{da^{(3)}} \cdot \frac{da^{(3)}}{dz^{(3)}} \cdot \frac{dz^{(3)}}{d\theta_{11}^{(2)}}$$

$$= \underbrace{(a^{(3)} - y) \cdot g'(z^{(3)})}_{\delta^{(3)}} \cdot a_1^{(2)}$$

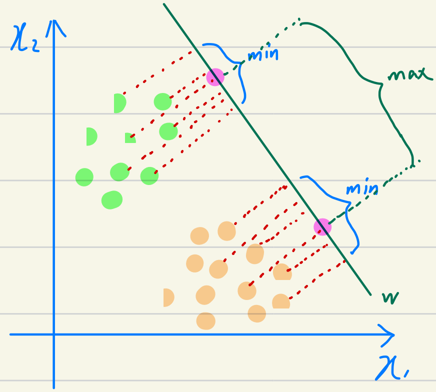
$$\frac{dJ(\theta)}{d\theta_{11}^{(1)}} = \frac{dJ(\theta)}{da^{(3)}} \cdot \frac{da^{(3)}}{dz^{(3)}} \cdot \frac{dz^{(3)}}{da_1^{(2)}} \cdot \frac{da_1^{(2)}}{dz_1^{(2)}} \cdot \frac{dz_1^{(2)}}{d\theta_{11}^{(1)}}$$

$$= \underline{(a^{(3)} - y) \cdot g'(z^{(3)})} \cdot \theta_{11}^{(2)} \cdot g'(z_1^{(2)}) \cdot x_1$$

$$\underline{\delta^{(3)}\,\theta_{11}^{(2)}\,g'(z_1^{(2)})}\,x_1$$

$$\delta_1^{(2)}\,x_1$$

线性判别分析:



思想：如图所示，要找出一条线，使样本点映射后易于区分(类内小，类间大)

$$X = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{bmatrix}_{n \times p} \quad (x_i\text{均为列向量}) \qquad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad n \times 1$$

设 w 为 $(p \times 1)$ 列向量

推导：映射后的值：$z_i = w^T x_i$ ，均值：$\bar{z} = \frac{1}{N}\sum_{i=1}^{N} z_i = \frac{1}{N}\sum_{i=1}^{N} w^T x_i$

映射后的方差：$S = \frac{1}{N}\sum_{i=1}^{N}(z_i - \bar{z})^2 = \frac{1}{N}\sum_{i=1}^{N}(z_i - \bar{z})(z_i - \bar{z})^T$

则类1：$\begin{cases} \bar{z}_1 = \frac{1}{N}\sum_{i=1}^{N} w^T x_i \\ S_1 = \frac{1}{N}\sum_{i=1}^{N}(z_i - \bar{z})(z_i - \bar{z})^T \end{cases}$ ，类2：$\begin{cases} \bar{z}_2 = \frac{1}{N}\sum_{i=1}^{N} w^T x_i \\ S_2 = \frac{1}{N}\sum_{i=1}^{N}(z_i - \bar{z})(z_i - \bar{z})^T \end{cases}$

类间：$(\bar{z}_1 - \bar{z}_2)^2$ ，类内：$S_1 + S_2$ ，目标：类内小，类间大 $\longrightarrow$ loss 函数

loss 函数：$J(w) = \frac{(\bar{z}_1 - \bar{z}_2)^2}{S_1 + S_2}$ ，目标：$\arg\max(J(w))$

$J(w)$：分子 $= \left(\frac{1}{N_1}\sum_{i=1}^{N_1} w^T x_i - \frac{1}{N_2}\sum_{i=1}^{N_2} w^T x_i\right)^2 = \left[w^T\left(\frac{1}{N_1}\sum_{i=1}^{N_1} x_i - \frac{1}{N_2}\sum_{i=1}^{N_2} x_i\right)\right]^2 = \left[w^T(\bar{x}_{c_1} - \bar{x}_{c_2})\right]^2$

分母以 S_1 为例：$S_1 = \frac{1}{N_1}\sum_{i=1}^{N_1}\left(w^T x_i - \frac{1}{N_1}\sum_{j=1}^{N_1} w^T x_j\right)\left(w^T x_i - \frac{1}{N_1}\sum_{j=1}^{N_1} w^T x_j\right)^T$ 　把 w^T 提出来

$$= \frac{1}{N_1}\sum_{i=1}^{N_1} w^T (x_i - \bar{x}_{c_1})(x_i - \bar{x}_{c_1})^T w = w^T\left[\frac{1}{N_1}\sum_{i=1}^{N_1}(x_i - \bar{x}_{c_1})(x_i - \bar{x}_{c_1})^T\right] w = w^T S_{c_1} w$$

$\therefore$ 分母 $= w^T S_{c_1} w + w^T S_{c_2} w = w^T (S_{c_1} + S_{c_2}) w$

$$\therefore J(w) = \frac{w^T(\bar{x}_{c_1} - \bar{x}_{c_2})(\bar{x}_{c_1} - \bar{x}_{c_2})^T w}{w^T(S_{c_1} + S_{c_2})w}$$

$$\because J(w) = \frac{w^T(\bar{x}_{c_1}-\bar{x}_{c_2})(\bar{x}_{c_1}-\bar{x}_{c_2})^T w}{w^T(S_{c_1}+S_{c_2})w}$$

，令 $S_b = (\bar{x}_{c_1}-\bar{x}_{c_2})(\bar{x}_{c_1}-\bar{x}_{c_2})^T$ 类间方差

$S_w = (S_{c_1}+S_{c_2})$ 类内方差

则 $J(w) = w^T S_b w \cdot (w^T S_w w)^{-1}$

$$\frac{\partial J(w)}{\partial w} = 2 S_b w (w^T S_w w)^{-1} + w^T S_b w \cdot (-1)\cdot(w^T S_w w)^{-2}\cdot 2 S_w \cdot w = 0$$，可得

$$S_b w \cdot \underbrace{(w^T S_w w)}_{\in R} = \underbrace{w^T S_b w}_{\in R} S_w \cdot w$$

$S_w : (p \times p)$ $S_b : (p \times p)$ $w : (1 \times p)$ $w^T : (p \times 1)$

求的是方向向量

$$S_w w = \frac{w^T S_w w}{w^T S_b w} S_b w \longrightarrow w = \frac{w^T S_w w}{w^T S_b w} S_w^{-1} S_b w \propto S_w^{-1} S_b w$$

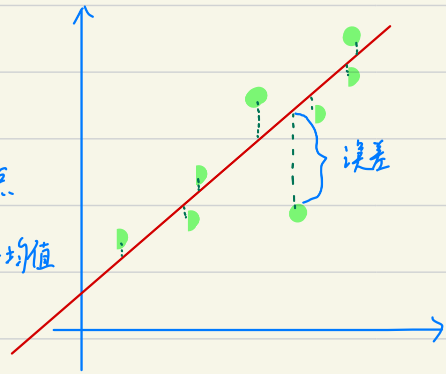
$$S_b w = (\bar{x}_{c_1}-\bar{x}_{c_2})\underbrace{(\bar{x}_{c_1}-\bar{x}_{c_2})^T}_{1 \times p} \cdot \underset{p \times 1}{w} \propto (\bar{x}_{c_1}-\bar{x}_{c_2})$$

$\therefore w \propto S_w^{-1}(\bar{x}_{c_1}-\bar{x}_{c_2})$

经验：$b = -\frac{1}{2} w^T(\bar{x}_{c_1}-\bar{x}_{c_2})$

模型 $y = w^T x + b$

线性回归



使用场景：目标值与特征线性

思想：使用一个多元一次方程去拟合所有数据点

度量尺度：均方误差(MSE)，即数据误差平方和的平均值

MSE越小，拟合效果越佳

MSE: $\frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2$

令：

$$\underset{(n\times m)}{x} = \begin{bmatrix} 1 & x_1^1 & x_2^1 & \cdots & x_m^1 \\ 1 & x_1^2 & x_2^2 & \cdots & x_m^2 \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_1^n & x_2^n & \cdots & x_m^n \end{bmatrix} \quad \underset{(m\times 1)}{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_m \end{bmatrix} \quad \underset{(n\times 1)}{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad \underset{(n\times 1)}{\hat{y}} = \begin{bmatrix} 1 & x_1^1 & x_2^1 & \cdots & x_m^1 \\ 1 & x_1^2 & x_2^2 & \cdots & x_m^2 \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_1^n & x_2^n & \cdots & x_m^n \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_m \end{bmatrix} = x\theta$$

$\hat{y} = x\theta$ ， 损失函数 $J(\theta) = \frac{1}{m}\sum_{i=1}^{m}(x\theta - y_i)^2$ ，目标：$\min J(\theta)$

1. 最小二乘法:

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}(x\theta - y_i)^2 = \frac{1}{m}(x\theta - y)^T(x\theta - y) = \frac{1}{m}[(x\theta)^T - y^T](x\theta - y)$$

$$= \frac{1}{m}(\theta^T x^T - y^T)(x\theta - y) = \frac{1}{m}(\theta^T x^T x\theta - \theta^T x^T y - y^T x\theta + y^T y)$$

$\because y^T x\theta$ 与 $\theta^T x^T y$ 互为转置，但均为标量

$\therefore J(\theta) = \frac{1}{m}(\theta^T x^T x\theta - 2\theta^T x^T y + y^T y)$

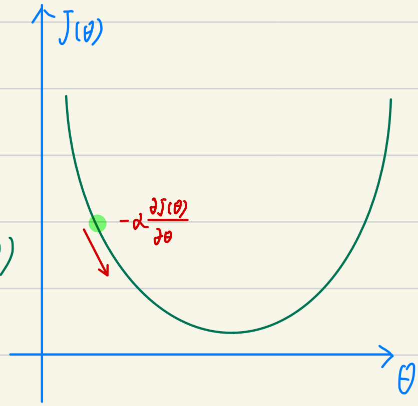
$$\frac{\partial}{\partial\theta}J(\theta) = \frac{1}{m}\cdot\frac{\partial}{\partial\theta}(\theta^T x^T x\theta - 2\theta^T x^T y + y^T y) = \frac{1}{m}\cdot(-2x^T y + 2x^T x\theta)$$

令 $\frac{\partial J(\theta)}{\partial\theta} = 0$，$\frac{1}{m}(-2x^T y + 2x^T x\theta) = 0$ 可得

$$x^T y = x^T x\theta \Rightarrow \theta = (x^T x)^{-1}x^T y$$

2、梯度下降法

右图为$J(\theta)$图象，要使$J(\theta)$最小，就要令其向箭头方向移动 $\Rightarrow$ 减去学习率×偏导



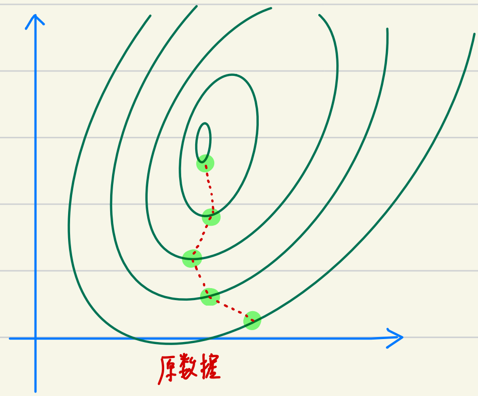
推导：

由(1)可得 $\frac{\partial J(\theta)}{\partial\theta} = \frac{1}{m}(2x^Tx\theta - 2x^Ty) = \frac{2}{m}x^T(x\theta-y)$

$$\theta_{new} = \theta_{old} - \alpha\cdot\frac{2}{m}x^T(x\theta-y)$$

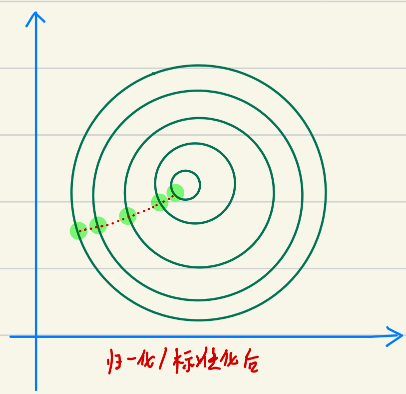
归一化/标准化

使用梯度下降法时，归一化/标准化可加快梯度下降的求解速度，如右图



归一化：

$$x' = \frac{x-\min(x)}{\max(x)-\min(x)}$$



标准化：

$$x' = \frac{(x-\mu)}{\sigma}$$

μ为期望 σ为标准差

正则化

常见的向量范数：

L_0 范数：$||x||_0$ 表示向量 x 中非零元素的个数

L_1 范数：$||x||_1 = \sum_{i=1}^{n} |x_i|$ 表示非零元素的绝对值之和

L_2 范数：$||x||_2 = \sqrt{\sum_{i=1}^{n} x^2}$

L_p 范数：$||x||_p = \sqrt[p]{\sum_{i=1}^{n} x^p}$

过拟合：

红：恰好，$y = w_0 + w_1 x_1$

绿：过拟合，$y = w_0 + w_1 x_1 + w_2 x_2 + \cdots$

如何令 $y = w_0 + w_1 x_1 \rightarrow y = w_0 + w_1 x_1 + w_2 x_2 + \cdots$ 呢？

减少向量 w 的个数 or 令某些参数减小 → 正则化

推理：损失函数 $J(w, x, y)$

减小某些 w 的值，但又不知道该减少哪一个 w，因此可以设 限制条件

① $\begin{cases} |w_1| + |w_2| + |w_3| + \cdots + |w_n| = ||w||_1 \leq C \\ \text{argmin}\, J(w, x, y) \end{cases}$

② $\begin{cases} \sqrt{w_1^2 + w_2^2 + w_3^2 + \cdots + w_n^2} = ||w||_2 \leq C \\ \text{argmin}\, J(w, x, y) \end{cases}$

先研究①：引入拉格朗日函数，可得

$$L(w) = \underset{\uparrow \text{min}}{J(w, x, y)} + \lambda(||w||_1 - C)$$

要令 $L(w)$ 最小，则令 $\frac{\partial L(w)}{\partial \lambda} = 0$，设其解为 λ^*

则 $\min L(w,\lambda) = \min J(w,x,y) + \lambda^*(\|w\|_1 + C)$

$$= \min J(w,x,y) + \lambda^*\|w\|_1 + \lambda^* C$$

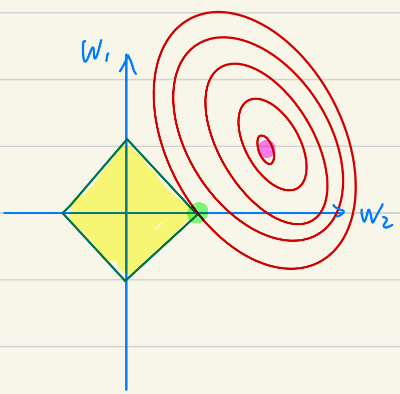
$$\propto \underbrace{\min J(w,x,y) + \lambda^*\|w\|_1}_{L_1\text{正则化}}$$

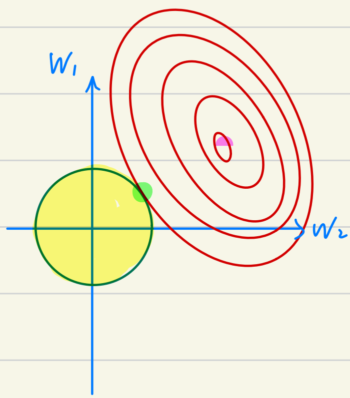
∴损失函数：$J(w,x,y) + \lambda\|w\|_1$

$$= J(w,x,y) + \lambda\sum_{i=1}^{n}|w_i|$$

L_2 正则化同理

几何解释：





：w_1 与 w_2 的取值范围　　：损失函数最小的点

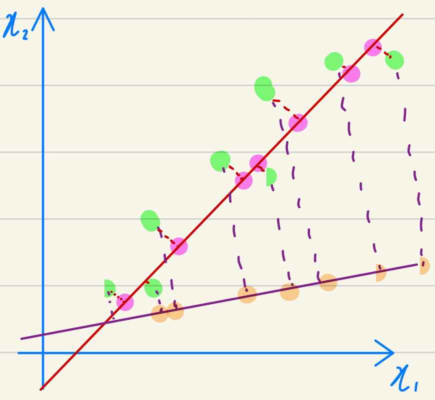
：w_1 与 w_2 取值范围内与　最近的点，也是 w_1 与 w_2 取值点.

主成分分析(PCA)

作用：降维降噪

目的：找到一条线，使映射后数据之间分隔大

如：● 的分隔比 ● 大 ⟶



设：

$$x = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \vdots & \vdots & & \vdots \\ x_1^m & x_2^m & \cdots & x_n^m \end{bmatrix} \quad (m \times n)$$

$$w: [w_1 \; w_2 \; \cdots \; w_k] \quad (n \times k) \qquad w_i (n \times 1)$$

推导：分隔大 ⟶ 方差大

$$S_j = \frac{1}{m}\sum_{i=1}^{m}(x_i - \bar{x})^2 \quad \text{[单个特征的方差]}$$

① 均值归零：所有数据减去其对应特征的均值

则 $\bar{x} = [0 \; 0 \; 0 \cdots 0]$，

$$S(w) = \left.\begin{bmatrix} \frac{1}{m}\sum_{i=1}^{m}(x_1^{(i)})^2 \\ \frac{1}{m}\sum_{i=1}^{m}(x_2^{(i)})^2 \\ \vdots \\ \frac{1}{m}\sum_{i=1}^{m}(x_n^{(i)})^2 \end{bmatrix}\right\} \text{求和}$$

目标：$\arg\max S(w)$，求偏导，可得

$$\frac{\partial S(w)}{\partial w} = \begin{bmatrix} \frac{\partial S(w_1)}{\partial w_1} \\ \frac{\partial S(w_2)}{\partial w_2} \\ \vdots \\ \frac{\partial S(w_n)}{\partial w_n} \end{bmatrix} = \frac{2}{m}\begin{bmatrix} \sum_{i=1}^{m}(x_1^{(i)}w_1 + x_2^{(i)}w_2 + \cdots + x_n^{(i)}w_n)x_1^{(i)} \\ \sum_{i=1}^{m}(x_1^{(i)}w_1 + x_2^{(i)}w_2 + \cdots + x_n^{(i)}w_n)x_2^{(i)} \\ \vdots \\ \sum_{i=1}^{m}(x_1^{(i)}w_1 + x_2^{(i)}w_2 + \cdots + x_n^{(i)}w_n)x_n^{(i)} \end{bmatrix} = \frac{2}{m}\begin{bmatrix} \sum_{i=1}^{m}(x^{(i)}w)x_1^{(i)} \\ \sum_{i=1}^{m}(x^{(i)}w)x_2^{(i)} \\ \vdots \\ \sum_{i=1}^{m}(x^{(i)}w)x_n^{(i)} \end{bmatrix}$$

$$= \frac{2}{m}\left[x^{(1)} \; x^{(2)} \cdots x^{(n)}\right]\begin{bmatrix} x^{(1)}w \\ x^{(2)}w \\ \vdots \\ x^{(m)}w \end{bmatrix} = \frac{2}{m}\cdot x^T(xw)$$

使用梯度上升法，$W_{new} = W_{old} + \alpha\frac{\partial S(w)}{\partial w}$，取得最大值。

基础

条件概率：$P(B|A)=\frac{P(AB)}{P(A)}$　　$P(AB)$为A，B事件一同发生的概率

$P(B|A)$为在A发生的前提下B发生的概率

贝叶斯公式：$P(A|B)=\frac{P(B|A)P(A)}{P(B)}=\frac{P(B|A)P(A)}{P(B|A)P(A)+P(B|\bar{A})P(\bar{A})}$　（$\bar{A}$表非A）

似然函数：　样本集 $D=\{x_1, x_2 \cdots x_n\}$

$$l(\theta)=P(D|\theta)=P(x_1,x_2\cdots x_n|\theta)=P(x_1|\theta)\cdot P(x_2|\theta)\cdots P(x_n|\theta)=\prod_{i=1}^{n}P(x_i|\theta)$$

极大似然估计：　$\arg\max l(\theta)=\arg\max \ln l(\theta)$

$$\ln l(\theta)=\ln\prod_{i=1}^{n}P(x_i|\theta)=\sum_{i=1}^{n}\ln P(x_i|\theta)$$

一维高斯分布：$f(x)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$　$(-\infty<x<\infty)$

n维高斯分布：$f(x)=\frac{1}{(2\pi)^{n/2}}\cdot\frac{1}{|\Sigma|^{1/2}}\cdot e^{-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)}$

logistic 回归:

Sigmoid 概率函数推导: $P(C_1|x) = \dfrac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1)+P(x|C_0)P(C_0)} = \dfrac{1}{1+\dfrac{P(x|C_0)P(C_0)}{P(x|C_1)P(C_1)}}$

$$= \frac{1}{1+\exp\left(-\ln\frac{P(x|C_1)P(C_1)}{P(x|C_0)P(C_0)}\right)}$$

令 $a = \ln\dfrac{P(x|C_1)P(C_1)}{P(x|C_0)P(C_0)}$ 得 $\dfrac{1}{1+\exp(-a)}$

a与线性函数 $\theta x+b$ 有何关系?　　设服从同协方差高斯分布

$P(x|C_k) = \dfrac{1}{(2\pi)^{n/2}}\cdot\dfrac{1}{|\Sigma|^{1/2}}\exp\left[-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\right]$　其中μ为数学期望, Σ为协方差矩阵

$$a = \ln\frac{P(x|C_1)P(C_1)}{P(x|C_0)P(C_0)} = \ln\frac{P(x|C_1)P(C_1)}{P(x|C_2)P(C_2)} + \ln\frac{P(C_1)}{P(C_2)}$$

$$= \ln\frac{\cancel{\frac{1}{(2\pi)^{n/2}}}\cdot\cancel{\frac{1}{|\Sigma|^{1/2}}}\cdot\exp\left[-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right]}{\cancel{\frac{1}{(2\pi)^{n/2}}}\cdot\cancel{\frac{1}{|\Sigma|^{1/2}}}\cdot\exp\left[-\frac{1}{2}(x-\mu_2)^T\Sigma^{-1}(x-\mu_2)\right]} + \ln\frac{P(C_1)}{P(C_2)}$$

$$= \cancel{\ln}\,\cancel{\exp}\left[-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)+\frac{1}{2}(x-\mu_2)^T\Sigma^{-1}(x-\mu_2)\right] + \ln\frac{P(C_1)}{P(C_2)}$$

$$= \left[\cancel{-\frac{1}{2}x^T\Sigma^{-1}x} + \underline{(\Sigma^{-1}\mu_1)^T}x - \frac{1}{2}\mu_1\Sigma^{-1}\mu_1 + \cancel{\frac{1}{2}x^T\Sigma^{-1}x} - \underline{(\Sigma^{-1}\mu_2)^T}x + \frac{1}{2}\Sigma^{-1}\mu_2\right] + \ln\frac{P(C_1)}{P(C_2)}$$

$$= \underbrace{\left[\Sigma^{-1}(\mu_1-\mu_2)\right]^T}_{\theta}x + \underbrace{\left[-\frac{1}{2}\mu_1\Sigma^{-1}\mu_1+\frac{1}{2}\mu_2\Sigma^{-1}\mu_2+\ln\frac{P(C_1)}{P(C_2)}\right]}_{b}$$

$$= \theta^T x + b$$

构造损失函数：利用极大似然估计

$$l(w)=\prod_{i=1}^{m}P(c_i|x^{(i)};w)=\prod_{i=1}^{m}(h_w(x))^{y^{(i)}}\cdot[1-h_w(x)]^{1-y^{(i)}}$$

$$L(w)=\ln l(w)=\sum_{i=1}^{m}y^{(i)}\ln h_w(x^{(i)})+(1-y^{(i)})\ln[1-h_w(x^{(i)})]$$

$$J(w)=-\frac{1}{m}\sum_{i=1}^{m}y^{(i)}\ln h_w(x^{(i)})+(1-y^{(i)})\ln[1-h_w(x^{(i)})]$$ 又称二元交叉熵

目标：$\arg\min J(w)$，使用梯度下降法　　设 $g(x)=\frac{1}{1+\exp(-x)}$，$g'(x)=g(x)(1-g(x))$

$$\frac{\partial J(w)}{\partial w_j}=-\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\frac{1}{h_w(x^{(i)})}\cdot\underline{\frac{\partial}{\partial w_j}h_w(x^{(i)})}-(1-y^{(i)})\frac{1}{1-h_w(x^{(i)})}\cdot\underline{\frac{\partial}{\partial w_j}h_\theta(x^{(i)})}\right]$$

$$=-\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\frac{1}{h_w(x^{(i)})}-(1-y^{(i)})\frac{1}{1-h_w(x^{(i)})}\right]\underline{\frac{\partial}{\partial w_j}g(x^{(i)}w)}\rightarrow g(x^{(i)}w)(1-g(x^{(i)}w))\cdot x_j^{(i)}$$

$$=-\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}(1-g(x^{(i)}w))-(1-y^{(i)})(g(x^{(i)}w))\right]x_j^{(i)}$$

$$=-\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}-\cancel{y^{(i)}g(x^{(i)}w)}-g(x^{(i)}w)+\cancel{y^{(i)}g(x^{(i)}w)}\right]x_j^{(i)}$$

$$=-\frac{1}{m}\sum_{i=1}^{m}(y^{(i)}-h_w(x^{(i)}))x_j^{(i)}=\frac{1}{m}\sum_{i=1}^{m}(h_w(x^{(i)})-y^{(i)})x_j^{(i)}$$

偏导矩阵：$\frac{\partial J(w)}{\partial w}=\frac{1}{m}x^T(h_w(x^{(i)})-y^{(i)})$

梯度下降：$W_{new}=W_{old}-\alpha\frac{\partial J(w)}{\partial w}$

多分类：softmax 回归 (one vs one)

$$P(C_k|x)=\frac{P(x|C_k)P(C_k)}{P(x|C_1)P(C_1)+P(x|C_2)P(C_2)+\cdots P(x|C_K)P(C_K)}$$

$$=\frac{P(x|C_k)P(C_k)}{\sum_{i=1}^{K}P(x|C_i)P(C_i)}=\frac{e^{\ln(P(x|C_k)P(C_k))}}{\sum_{i=1}^{K}e^{\ln[P(x|C_i)P(C_i)]}}=\frac{e^{a_k}}{\sum_{i=1}^{K}e^{a_i}}$$

其中 $a_k=\ln[P(x|C_k)P(C_k)]$，设服从高斯分布

$$P(x|C_k)=\frac{1}{(2\pi)^{n/2}}\cdot\frac{1}{|\Sigma|^{1/2}}\exp\left[-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\right]$$ 其中 μ 为数学期望，Σ 为协方差矩阵

$$a=\ln P(x|C_k)P(C_k)=\ln P(x|C_k)+\ln P(C_k)$$

$$=\ln\frac{1}{(2\pi)^{n/2}}\cdot\frac{1}{|\Sigma|^{1/2}}\cdot\exp\left[-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\right]+\ln P(C_k)$$

$$=\left[-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\right]+\ln P(C_k)+\underbrace{\ln\left(\frac{1}{(2\pi)^{n/2}}\cdot\frac{1}{|\Sigma|^{1/2}}\right)}_{\text{常数}}$$

$$=\underbrace{(\Sigma^{-1}\mu_k)^T}_{\theta}x\underbrace{-\frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k+\ln P(C_k)}_{b}$$

$$a=\theta^Tx+b,$$

同样，利用极大似然估计

似然函数：$L(w)=\prod_{i=1}^{m}\prod_{k=1}^{K}P(C_k|x;w)=\prod_{i=1}^{m}\prod_{k=1}^{K}\left[\frac{\exp(xw_k)}{\sum_{l=1}^{K}\exp(xw_l)}\right]^{y_k^{(i)}}$，$k\in[1,2\cdots K)$

$$J(w)=\ln L(w)=\ln\prod_{i=1}^{m}\prod_{k=1}^{K}\left[\frac{\exp(xw_k)}{\sum_{l=1}^{K}\exp(xw_l)}\right]^{y_k^{(i)}}=\sum_{i=1}^{m}\sum_{k=1}^{K}\left(\cancel{\ln e}^{xw_k}-\ln\sum_{l=1}^{K}e^{xw_l}\right)\cdot y_k^{(i)}$$

对于每个数据来说，$J_m(w)=\sum_{k=1}^{K}\left(xw_k-\ln\sum_{i=1}^{K}e^{xw_i}\right)\cdot y_m$

$$\frac{\partial J_m(w)}{\partial w}=\underset{(n\times 1)}{x_m^T}\cdot\underset{(1\times k)}{\left(y_m-\frac{e^{xw}}{\sum_{i=1}^{K}e^{xw_i}}\right)}$$ 再用梯度上升法即可