- 1. 绪论
- 2. 模型评估和选择
  - 2.1. 经验误差与过拟合
  - 2.2. 偏差和方差
- 3. 线性模型
  - 3.1. 对数几率回归
  - 3.2. 线性判别分析

1. 绪论

Supervised Learning
Unsupervised Learning
generalization "泛化"
Generally we assume that all sampling space obey a distribution D,our samples all independent and identically distributed.(i,i,d)
induction 归纳 ——> generalization
deduction 演绎 ——> specialization

By simply, Assume Sampling space $\mathcal{X}$ and hypothesis space $\mathcal{H}$ are discrete. 令 $P(h|X,\mathfrak{L}_a)$ represent algorithm $\mathfrak{L}_a$ output probability h based on train data X. Make f = ground true function. $\mathfrak{L}_a$在 train data 之外的误差:

$$E_{ote}(\mathfrak{L}_a|X,f) = \sum_h \sum_{x\in\mathcal{X}-X} P(x)\,\underbrace{\mathbb{I}(h(x)\neq f(x))}_{\text{指示function, True=1, false=0}}\,P(h|X,\mathfrak{L}_a)$$

Considering binary classification, Ground true function $\mathcal{X}\to\{0,1\}$, function space $\{0,1\}^{|\mathcal{X}|}$, 对所有可能的f按均匀分布对误差求和, (若f均匀分布 则一半的f对X的 prediction different from $h(x)$)

$$\sum_f E_{ote}(\mathfrak{L}_a|X,f) = \sum_f \sum_h \sum_{x\in\mathcal{X}-X} P(x)\,\mathbb{I}(h(x)\neq f(x))\,P(h|X,\mathfrak{L}_a) \quad ①$$

$$= \sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\mathfrak{L}_a) \sum_f \mathbb{I}(h(x)\neq f(x)) \quad ②$$

$$= \sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\mathfrak{L}_a)\, \tfrac{1}{2}2^{|\mathcal{X}|} \quad ③$$

$$= \tfrac{1}{2}2^{|\mathcal{X}|} \sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\mathfrak{L}_a) = 2^{|\mathcal{X}|-1} \sum_{x\in\mathcal{X}-X} P(x)\cdot 1 \quad ④$$

No free Lunch → Total Error has no relationship with learning algorithm

②–③ Assume Sampling space only contains two samples: $\mathcal{X}=\{x_1,x_2\}$, $|\mathcal{X}|=2$, 那么 Ground true function:

f_1: $f_1(x_1)=0$, $f_1(x_2)=0$;

f_2: $f_2(x_1)=0$, $f_2(x_2)=1$;

f_3: $f_3(x_1)=1$, $f_3(x_2)=0$;

f_4: $f_3(x_1)=1$, $f_3(x_2)=1$;

一共有 $2^{|\mathcal{X}|}=2^2=4$ 个函数, f是任何能将样本映射到$\{0,1\}$的函数且服从均匀分布.

在这里我们假设真实的目标函数f为“任何能将样本映射到{0,1}的函数且服从均匀分布”，但是实际情形并非如此，通常我们只认为能高度拟合已有样本数据的函数才是真实目标函数，例如，现在已有的样本数据为$(x_1,0),(x_2,1)$,那么此时f_2才是我们认为的ground true function. Since we haven't collect or even they don't exist {(x_1,0),(x_2,0)},{(x_1,1),(x_2,1)},{(x_1,1),(x_2,0)},所以f_1, f_3, f_4都不算真实目标函数。

connectionism 联接主义

perceptron 感知机

Symbolism 符号主义

statistical learning 统计学习

Support Vector Machine SVM

Crowdsourcing 众包

$Ensemble$ Learning

2. 模型评估和选择

2.1. 经验误差与过拟合

Generalization error 泛化误差
training error 经验误差，训练误差
overfitting 过拟合
Underfitting 欠拟合
model selection
hold-out 留出法
cross validation 交叉验证法

2.2. 偏差和方差

$Bias$-variance decomposition 偏差-方差分解

Test dataset x, y_D 是 x 的标记, y 是 x 真实标记, $f(x;D)$ 为训练集 D 上学得模型 f 在 x 上的 predict output. 以 Regression 为例, Learning algorithm 的期望预测:

$$\bar f(x) = E_D[f(x;D)]$$

使用样本数目相同的不同训练集产生的方差:

$$var(x) = E_D[(f(x;D) - \bar f(x))^2]$$

Noise:

$$\varepsilon^2 = E_D[(y_D - y)^2]$$

Bias: 期望输出和真实标记的差别:

$$bias^2(x) = (\bar f(x) - y)^2$$

For easy of discussion. Assume noise expect is zero, that is, $E_D[y_D - y] = 0$

$$E(f;D) = E_D[(f(x;D) - y_D)^2]$$

$$= E_D[(f(x;D) - \bar f(x) + \bar f(x) - y_D)^2]$$

$E(X+Y) = E(X)+E(Y)$

$$= E_D[(f(x;D) - \bar f(x))^2] + E_D[(\bar f(x) - y_D)^2] + E_D[2(f(x;D) - \bar f(x))(\bar f(x) - y_D)]$$

$$= \cdots + E_D[2(f(x;D) - \bar f(x))\cdot \bar f(x)] - E_D[2(f(x;D) - \bar f(x))\cdot y_D]$$

$E(AX+B) = AE(X)+B$

$$= \cdots + E_D[2f(x;D)\cdot \bar f(x) - 2\bar f(x)\cdot \bar f(x)] - \cdots$$

$$= \cdots + \underbrace{2\bar f(x) E_D[f(x;D)] - 2\bar f(x)\cdot \bar f(x)}_{0} - \cdots$$

$f(x;D)$ 和 y_D 是两个互相独立的随机变量, $E(XY) = E(X)E(Y)$

$$= \cdots + 0 - [2E_D[f(x;D)\cdot y_D] - 2\bar f(x)\cdot E_D(y_D)]$$

$$= \cdots + 0 - \{2E_D(f(x;D))\cdot E_D(y_D) - 2\bar f(x)\cdot E_D(y_D)\} \quad (E_D(f(x;D)) = \bar f(x))$$

$$= \cdots + 0 - \{0\}$$

$$= E_D[(f(x;D) - \bar f(x))^2] + E_D[(\bar f(x) - y + y - y_D)^2]$$

噪声期望为0

$$= E_D[(f(x;D) - \bar f(x))^2] + E_D[(\bar f(x) - y)^2] + E_D[(y - y_D)^2] + 2E_D[(\bar f(x) - y)\underline{(y - y_D)}]$$

$$= E_D[(f(x;D) - \bar f(x))^2] + \underbrace{(\bar f(x) - y)^2}_{\text{看作常量}} + E_D[(y_D - y)^2]$$

$$E(f;D) = bias^2(x) + var(x) + \varepsilon^2$$

偏差: 学习算法期望预测和真实结果的偏离程度, 刻画算法本身拟合能力

方差: 数据扰动造成的影响

噪声: 任何学习算法达到的期望泛化误差下界, 即学习问题本身的难度.

3. 线性模型

Linear model

Nonlinear model

Comprehensibility 可解释性

Linear regression

$Euclidean$ distance

least square method

$parameter$ estimation

$$f(x) = w_1x_1 + w_2x_2 + w_3x_3 + \cdots w_dx_d + b \Rightarrow f(x) = w^Tx + b$$

$f(x_i) = wx_i + b$ 使 $f(x_i) \simeq y_i$

How to Calculate the w and b? The key point is to measure the distance between $f(x)$ and y

$$(w^*, b^*) = \underset{(w,b)}{\arg\min} \sum_{i=1}^{m} (f(x_i) - y_i)^2 = \underset{(w,b)}{\arg\min} \sum_{i=1}^{m} (y_i - wx_i - b)^2$$

Least square method：找到一条直线，使所有样本到直线上的欧氏距离之和最小

$$E(w,b) = \sum_{i=1}^{m} (y_i - wx_i - b)^2 \quad \text{(Parameter estimation)}$$

$E(w,b)$ 是凸函数，当它关于 w 和 b 的导数为 0 时，得到 w 和 b 的最优解

凸函数： $f\left(\frac{x_1+x_2}{2}\right) \leq \frac{f(x_1)+f(x_2)}{2}$，或 2阶导非负（恒大于0）

$$\begin{aligned}\frac{\partial E(w,b)}{\partial w} &= \sum_{i=1}^{m} \frac{\partial}{\partial w}[y_i - wx_i - b]^2 \\ &= \sum_{i=1}^{m}[2\cdot(y_i - wx_i - b)\cdot(-x_i)] \\ &= \sum_{i=1}^{m}[2\cdot(wx_i^2 - y_ix_i + bx_i)] \\ &= 2\cdot\left(w\sum_{i=1}^{m}x_i^2 - \sum_{i=1}^{m}y_ix_i + b\sum_{i=1}^{m}x_i\right) \\ &= 2\left(w\sum_{i=1}^{m}x_i^2 - \sum_{i=1}^{m}(y_i - b)x_i\right) = 0\end{aligned}$$

$$\begin{aligned}\frac{\partial E(w,b)}{\partial b} &= \sum_{i=1}^{m}\frac{\partial}{\partial b}[y_i - wx_i - b]^2 \\ &= \sum_{i=1}^{m}[2(y_i - wx_i - b)\cdot(-1)] \\ &= 2\left(mb - \sum_{i=1}^{m}(y_i - wx_i)\right) = 0\end{aligned}$$

$$\Rightarrow \begin{cases} w\sum_{i=1}^{m}x_i^2 = \sum_{i=1}^{m}y_ix_i - \sum_{i=1}^{m}bx_i \\ b = \frac{1}{m}\sum_{i=1}^{m}(y_i - wx_i) \Rightarrow b = \bar{y} - w\bar{x} \end{cases}$$

$\frac{1}{m}\sum_{i=1}^{m}y_i = \bar{y}$, $\frac{1}{m}\sum_{i=1}^{m}x_i = \bar{x}$

$$\Rightarrow w\sum_{i=1}^{m}x_i^2 = \sum_{i=1}^{m}y_ix_i - \sum_{i=1}^{m}(\bar{y} - w\bar{x})x_i$$

$$w\sum_{i=1}^{m}x_i^2 = \sum_{i=1}^{m}y_ix_i - \bar{y}\sum_{i=1}^{m}x_i + w\bar{x}\sum_{i=1}^{m}x_i$$

$$w\left(\sum_{i=1}^{m}x_i^2 - \bar{x}\sum_{i=1}^{m}x_i\right) = \sum_{i=1}^{m}y_ix_i - \bar{y}\sum_{i=1}^{m}x_i$$

$$w = \frac{\sum_{i=1}^{m}y_ix_i - \bar{y}\sum_{i=1}^{m}x_i}{\sum_{i=1}^{m}x_i^2 - \bar{x}\sum_{i=1}^{m}x_i}$$

$$\because \bar{y}\sum_{i=1}^{m}x_i = \frac{1}{m}\sum_{i=1}^{m}y_i\sum_{i=1}^{m}x_i = \bar{x}\sum_{i=1}^{m}y_i, \quad \bar{x}\sum_{i=1}^{m}x_i = \frac{1}{m}\sum_{i=1}^{m}x_i\sum_{i=1}^{m}x_i = \frac{1}{m}\left(\sum_{i=1}^{m}x_i\right)^2$$

$$w = \frac{\sum_{i=1}^{m}y_i(x_i - \bar{x})}{\sum_{i=1}^{m}x_i^2 - \frac{1}{m}\left(\sum_{i=1}^{m}x_i\right)^2} \qquad w = \frac{\sum_{i=1}^{m}y_i(x_i - \bar{x})}{\sum_{i=1}^{m}x_i^2 - \bar{x}\sum_{i=1}^{m}x_i} = \frac{\sum_{i=1}^{m}y_i(x_i - \bar{x})}{\sum_{i=1}^{m}(x_i^2 - x_i\bar{x})}$$

$$\because \bar{y}\sum_{i=1}^{m}x_i = \bar{x}\sum_{i=1}^{m}y_i = \sum_{i=1}^{m}\bar{y}x_i = \sum_{i=1}^{m}\bar{x}y_i = m\bar{x}\bar{y} = \sum_{i=1}^{m}\bar{x}\bar{y},$$

$$\sum_{i=1}^{m}x_i\bar{x} = \bar{x}\sum_{i=1}^{m}x_i = \bar{x}\cdot m\cdot\frac{1}{m}\sum_{i=1}^{m}x_i = m\bar{x}^2 = \sum_{i=1}^{m}\bar{x}^2$$

$$w = \frac{\sum_{i=1}^{m}(y_ix_i - y_i\bar{x} - x_i\bar{y} + \bar{x}\bar{y})}{\sum_{i=1}^{m}(x_i^2 - x_i\bar{x} - x_i\bar{x} + \bar{x}^2)} = \frac{\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{m}(x_i - \bar{x})^2}$$

若令 $x = (x_1, x_2 \cdots x_m)^T$, $x_d = (x_1 - \bar{x}, x_2 - \bar{x}, \cdots x_m - \bar{x})^T$, $y = (y_1, y_2 \cdots y_m)^T$, $y_d = (y_1 - \bar{y}, y_2 - \bar{y} \cdots y_m - \bar{y})^T$

$$w = \frac{x_d^T y_d}{x_d^T x_d}$$

Multivariate linear regression 多元线性回归

Multivariate Linear regression $f(x_i) = w^T x_i + b$ 使得 $f(x_i) \simeq y_i$.

① 将 w 和 b merge 为：$\hat{w} = (w;b) = (w_1;w_2 \cdots w_d;b)$ ③ $\vec{y} = (y_1;y_2;\cdots y_m)$

② $\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} & 1 \\ x_{21} & x_{22} & \cdots & x_{2d} & 1 \\ \vdots & & & & \\ x_{m1} & x_{m2} & \cdots & x_{md} & 1 \end{bmatrix} = \begin{bmatrix} x_1^T & 1 \\ x_2^T & 1 \\ \vdots & \\ x_m^T & 1 \end{bmatrix}$ ④ $\hat{w}^* = \underset{\hat{w}}{\arg\min}\,(y - \mathbf{X}\hat{w})^T(y - \mathbf{X}\hat{w})$

⑤ 令 $E_{\hat{w}} = (y - \mathbf{X}\hat{w})^T(y - \mathbf{X}\hat{w})$, $(w^*, b^*) = \underset{(w,b)}{\arg\min} \sum_{i=1}^{m} (f(x_i) - y_i)^2 = \underset{(w,b)}{\arg\min} \sum_{i=1}^{m} (y_i - (w^T x_i + b))^2$

$\frac{\partial E_{\hat{w}}}{\partial \hat{w}} = \frac{\partial}{\partial \hat{w}} (y^T y - y\mathbf{X}\hat{w} - \hat{w}^T \mathbf{X}^T y + \hat{w}^T \mathbf{X}^T \mathbf{X} \hat{w})$

矩阵微分公式：① $\nabla_x (Ax) = A$, $\nabla_x X = \nabla_x (IX) = I$

$$\frac{\partial (\sum_j a_{ij} x_j)}{\partial x_k} = a_{ik} \Rightarrow \frac{\partial (Ax)_i}{\partial x_j} = \begin{bmatrix} \frac{\partial (a_{11}x_1 + a_{12}x_2 + \cdots a_{1n}x_n)}{\partial x_1} & \cdots & \frac{\partial (a_{11}x_1 + \cdots a_{1n}x_n)}{\partial x_n} \\ \vdots & & \\ \frac{\partial (a_{m1}x_1 + a_{m2}x_2 + \cdots a_{mn}x_n)}{\partial x_1} & \cdots & \frac{\partial (a_{m1}x_1 + \cdots a_{mn}x_n)}{\partial x_n} \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} = A$$

② $\nabla_x (a^T x) = \nabla_x (x^T a) = a$

$$\frac{\partial a^T x}{\partial x} = \frac{\partial x^T a}{\partial x} = a \Rightarrow \begin{bmatrix} \frac{\partial (a_1x_1 + a_2x_2 + \cdots a_nx_n)}{\partial x_1} \\ \frac{\partial (a_1x_1 + a_2x_2 + \cdots a_nx_n)}{\partial x_2} \\ \vdots \\ \frac{\partial (a_1x_1 + a_2x_2 + \cdots a_nx_n)}{\partial x_n} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = a$$

③ $\nabla_x \|x\|_2^2 = \nabla_x (x^T x) = 2x$

$$\frac{\partial \|x\|_2^2}{\partial x_i} = \frac{\partial \sum_j x_j^2}{\partial x_i} = \frac{\partial x_i^2}{\partial x_i} = 2x_i \Rightarrow \frac{\partial \|x\|_2}{\partial x_i} = \begin{bmatrix} \frac{\partial (x_1^2 + x_2^2 + \cdots x_n^2)}{\partial x_1} \\ \frac{\partial (x_1^2 + x_2^2 + \cdots x_n^2)}{\partial x_2} \\ \vdots \\ \frac{\partial (x_1^2 + x_2^2 + \cdots x_n^2)}{\partial x_n} \end{bmatrix} = \begin{bmatrix} 2x_1 \\ 2x_2 \\ \vdots \\ 2x_n \end{bmatrix}$$

④ $\nabla_x (x^T A x) = (A + A^T)x$ ，若A是对称矩阵，$\nabla_x (x^T A x) = (A + A^T)x = 2Ax$

① $$\begin{aligned} x^T A x = & a_{11}x_1x_1 + a_{12}x_1x_2 + a_{13}x_1x_3 + \cdots a_{1n}x_1x_n \\ & + a_{21}x_2x_1 + a_{22}x_2x_2 + a_{23}x_2x_3 + \cdots a_{2n}x_2x_n \\ & \vdots \\ & + a_{n1}x_nx_1 + a_{n2}x_nx_2 + a_{n3}x_nx_3 + \cdots a_{nn}x_nx_n \end{aligned}$$

$$\frac{\partial (x^T A x)}{\partial x_i} = \begin{bmatrix} 2a_{11}x_1 + (a_{12}+a_{21})x_2 + (a_{13}+a_{31})x_3 + \cdots (a_{1n}+a_{n1})x_n \\ (a_{21}+a_{12})x_1 + 2a_{22}x_2 + (a_{23}+a_{32})x_3 + \cdots (a_{2n}+a_{n2})x_n \\ (a_{31}+a_{13})x_1 + (a_{32}+a_{23})x_2 + 2a_{33}x_3 + \cdots (a_{3n}+a_{n3})x_n \\ \vdots \\ (a_{n1}+a_{1n})x_1 + (a_{n2}+a_{n2})x_2 + \cdots \cdots 2a_{nn}x_n \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \cdots & a_{1n} \\ a_{21} & a_{22} \cdots & a_{2n} \\ a_{31} & a_{32} \cdots & a_{3n} \\ \vdots & \vdots & \\ a_{n1} & a_{n2} \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} a_{11} & a_{21} \cdots & a_{n1} \\ a_{12} & a_{22} \cdots & a_{n2} \\ a_{13} & a_{23} \cdots & a_{n3} \\ \vdots & \vdots & \ddots \\ a_{1n} & a_{2n} & a_{nn} \end{bmatrix}$$

$$= (A + A^T)x$$

② $\frac{\partial(\sum_{ij} x_i a_{ij} x_j)}{\partial x_k} = \sum_j a_{kj} x_j + \sum_i x_i a_{ik} = Ax + A^T x = (A + A^T)x$

（若某个变量在函数表达式中多次出现，可以单独计算函数对自变量的每一次出现的导数，再把结果加起来）$f(x) = (2x+1)x + x^2 \Rightarrow f(x) = (2x_1+1)x_2 + x_3^2 \Rightarrow \frac{\partial f}{\partial x_1} + \frac{\partial f}{\partial x_2} + \frac{\partial f}{\partial x_3} \Rightarrow 6x+1$

（视为不同 x）　（视为相同 x）

$$E_{\hat{w}} = y^T y - y^T X \hat{w} - \hat{w}^T X^T y + \hat{w}^T X^T X \hat{w}$$

$\left(\frac{\partial a^T x}{\partial x} = \frac{\partial x^T a}{\partial x} = a, \ \frac{\partial x^T A x}{\partial x} = (A + A^T)x \right)$

$$\frac{\partial E_{\hat{w}}}{\partial \hat{w}} = \frac{\partial y^T y}{\partial \hat{w}} - \frac{\partial y^T X \hat{w}}{\partial \hat{w}} - \frac{\partial \hat{w}^T X^T y}{\partial \hat{w}} + \frac{\partial \hat{w}^T X^T X \hat{w}}{\partial \hat{w}}$$

$$\frac{\partial E_{\hat{w}}}{\partial \hat{w}} = 0 - X^T y - X^T y + (X^T X + X^T X)\hat{w}$$

$$\frac{\partial E_{\hat{w}}}{\partial \hat{w}} = 2X^T (X\hat{w} - y)$$

①当 $X^T X$ 为 full-rank matrix 满秩矩阵或 positive definite matrix 正定矩阵

$$\frac{\partial E_{\hat{w}}}{\partial \hat{w}} = 2X^T (X\hat{w} - y) = 0 \Rightarrow 2X^T X \hat{w} = 2X^T y \Rightarrow \hat{w}^* = (X^T X)^{-1} X^T y$$

令 $\hat{x}_i = (x_i, 1) \Rightarrow f(\hat{x}_i) = \hat{x}_i^T (X^T X)^{-1} X^T y$

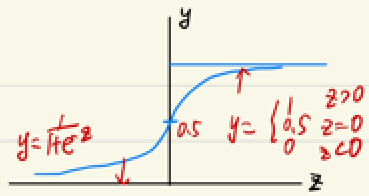
3.1. 对数几率回归

Unit-step function 单位阶跃函数

Surrogate function 替代函数

Logistic function

Linear Discriminant Analysis LDA



Logistic function: $y=\frac{1}{1+e^{-z}} \Rightarrow y=\frac{1}{1+e^{-(w^Tx+b)}}$

$e^{-z}=\frac{1}{y}-1=\frac{1-y}{y} \Rightarrow \ln\frac{y}{1-y}=w^Tx+b$

若将 y 视为样本 x 作为正例可能性，1-y 为反例可能性

"对数几率"(log odds) $\ln\frac{y}{y-1}$

若将 y 视为后验概率估计 $p(y=1|x) \Rightarrow \ln\frac{p(y=1|x)}{p(y=0|x)}=w^Tx+b$

Obviously, $p(y=1|x)=\frac{e^{w^Tx+b}}{1+e^{w^Tx+b}}$, $p(y=0|x)=\frac{1}{1+e^{-(w^Tx+b)}}$

于是通过极大似然法(Maximum likelihood method)来估计 w 和 b

给定 Dataset: $\{(x_i,y_i)\}_{i=1}^{m}$

$$\ell(w,b)=\sum_{i=1}^{m}\ln p(y_i|x_i;w,b)$$ 最大化公式(3.25)

即每个样本 belong to 真实标记 probability more bigger more better. 令 $\beta=(w;b)$, $\hat{x}=(x;1)$

$w^Tx+b \Rightarrow \beta^T\hat{x}_i$. 再令 $p_1(\hat{x};\beta)=p(y=1|\hat{x};\beta)$, $p_0(\hat{x};\beta)=p(y=0|\hat{x};\beta)=1-p_1(\hat{x};\beta)$

$$p(y_i|x_i;w,b)=y_ip_1(\hat{x}_i;\beta)+(1-y_i)p_0(\hat{x}_i;\beta)$$

$$\ell(\beta)=\sum_{i=1}^{m}\ln(y_ip_1(\hat{x}_i;\beta)+(1-y_i)p_0(\hat{x}_i;\beta))$$

Among them $p_1(\hat{x}_i;\beta)=\frac{e^{\beta^T\hat{x}_i}}{1+e^{\beta^T\hat{x}_i}}$, $p_0=\frac{1}{1+e^{\beta^T\hat{x}_i}}$

$$\ell(\beta)=\sum_{i=1}^{m}\ln\left(\frac{y_ie^{\beta^T\hat{x}_i}+1-y_i}{1+e^{\beta^T\hat{x}_i}}\right)=\sum_{i=1}^{m}\left(\ln(y_ie^{\beta^T\hat{x}_i}+1-y_i)-\ln(1+e^{\beta^T\hat{x}_i})\right)$$

由于 y_i=0 或 1 则:

$$\ell(\beta)=\begin{cases}\sum_{i=1}^{m}(-\ln(1+e^{\beta^T\hat{x}_i})), & y_i=0\\ \sum_{i=1}^{m}(\beta^T\hat{x}_i-\ln(1+e^{\beta^T\hat{x}_i})), & y_i=1\end{cases}$$

$$\ell(\beta)=\sum_{i=1}^{m}(y_i\beta^T\hat{x}_i-\ln(1+e^{\beta^T\hat{x}_i}))$$

3.2. 线性判别分析

Linear Discriminant Analysis LDA

Dataset: $D=\{(x_i,y_i)\}_{i=1}^{m}$, $y_i\in\{0,1\}$. 令 X_i, μ_i, Σ_i 分别表示第 $i\in\{0,1\}$ 类示例的集合, 均值向量, 协方差矩阵.
若将 Dataset projecting on 直线 w, 则两类样本的中心在直线上的投影分别为 $w^T\mu_0$, $w^T\mu_1$; If projecting all sampling points on the line w, 两类样本协方差分别为 $w^T\Sigma_0 w$, $w^T\Sigma_1 w$

$$\text{Max } J=\frac{\|w^T\mu_0-w^T\mu_1\|_2^2\uparrow}{w^T\Sigma_0 w+w^T\Sigma_1 w\downarrow}=\frac{\|(w^T\mu_0-w^T\mu_1)^T\|_2^2}{w^T(\Sigma_0+\Sigma_1)w}=\frac{\|(\mu_0-\mu_1)^Tw\|_2^2}{w^T(\Sigma_0+\Sigma_1)w}=\frac{[(\mu_0-\mu_1)^Tw]^T(\mu_0-\mu_1)^Tw}{w^T(\Sigma_0+\Sigma_1)w}$$

$$J=\frac{w^T(\mu_0-\mu_1)(\mu_0-\mu_1)^Tw}{w^T(\Sigma_0+\Sigma_1)w}$$

Define within-class scatter matrix "类内散度矩阵"

$$S_w=\Sigma_0+\Sigma_1=\sum_{x\in X_0}(x-\mu_0)(x-\mu_0)^T+\sum_{x\in X_1}(x-\mu_1)(x-\mu_1)^T$$

Define between-class scatter matrix

$$S_b=(\mu_0-\mu_1)(\mu_0-\mu_1)^T$$

$$\Rightarrow J=\frac{w^TS_bw}{w^TS_ww}$$ 广义瑞利商 (generalized Rayleigh quotient)

(Since numerator and denominator are about w^2, that means, 解 only related to orientation of w)

令 $w^TS_ww=1\Rightarrow$

$$\begin{cases}\min\limits_{w} & -w^TS_bw\\ s.t. & w^TS_ww=1\end{cases}\xrightarrow[\text{乘子法}]{\text{拉格朗日}} S_bw=\lambda S_ww$$

拉格朗日乘子法: ①梯度向量与等高线切线垂直 $\Rightarrow\begin{cases}\nabla f=\lambda\nabla g\\ g=0\end{cases}\Rightarrow F=f+\lambda g\Rightarrow\begin{bmatrix}\frac{F}{\partial x}\\ \frac{F}{\partial y}\\ \frac{F}{\partial \lambda}\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix}$

$$L(w,\lambda)=-w^TS_bw+\lambda(w^TS_ww-1)$$

$$\Rightarrow\frac{\partial L(w,\lambda)}{\partial w}=-\frac{\partial(w^TS_bw)}{\partial w}+\lambda\frac{\partial(w^TS_ww-1)}{\partial w}=-(S_b+S_b^T)w+\lambda(S_w+S_w^T)w$$

Since $S_b=S_b^T$, $S_w=S_w^T$, so:

$$\frac{\partial L(w,\lambda)}{\partial w}=-2S_bw+2\lambda S_ww=0$$

$$\Rightarrow S_bw=\lambda S_ww$$

$$\Rightarrow(\mu_0-\mu_1)(\mu_0-\mu_1)^Tw=\lambda S_ww$$

If we make $(\mu_0-\mu_1)^Tw=y$ 则:

$$y(\mu_0-\mu_1)=\lambda S_ww$$

$$w=\frac{y}{\lambda}S_w^{-1}(\mu_0-\mu_1)$$

Because the ultimate solution w that we don't care the size, only care the orientation, Hence, w 可以取 arbitrary value, 由于 ... 大小固定, 所以 y ... 大小影响, 可以令 $y=\lambda$

Given the stable of digital solution, Generally, we use SVD for S_w, that is:

$S_w = V\Sigma V^T$ Σ对角矩阵. 皆为S_w奇异值

$S_w^{-1} = V\Sigma^{-1}V^T$

LDA可以推广到多分类任务中，假定存在N个类，且第i类示例数为m_i，我们定理全局散度矩阵：