Hand - written Part

学号: t11902210　　姓名: 张一凡　　(交换生)

code question report 在后面

Problem 1

Consider a binary classification data set $\{(x_n, y_n)\}_{n=1}^{N}$ with $x_n \in R^d$ and $y_n \in \{-1, 1\}$. For any weight vector w within a linear model, define an error function

$$err(w^T x, y) = (\max(1 - y w^T x, 0))^2$$

That is, $$E_{in}(w) = \frac{1}{N} \sum_{n=1}^{N} (\max(1 - y_n w^T x_n, 0))^2$$

Running gradient descent to optimize $E_{in}(w)$ requires calculating its gradient direction $\nabla E_{in}(w)$ (and then move opposite to that direction). What is $\nabla E_{in}(w)$?

$$\nabla E_{in}(w) = \frac{\partial E_{in}(w)}{\partial w} = \frac{1}{N} \sum_{n=1}^{N} \cancel{\max(1 - y_n w^T x_n, 0)} \cdot \frac{\partial f(w)}{\partial w}$$

$$f(w) = [\max(1 - y_n w^T x_n, 0)]^2$$

$$= \begin{cases} (1 - y_n w^T x_n)^2 & (y_n w^T x_n \le 1) \\ 0 & (y_n w^T x_n > 1) \end{cases}$$

$$\frac{\partial f(w)}{\partial w} = \begin{cases} -2 y_n x_n (1 - y_n w^T x_n) & (y_n w^T x_n \le 1) \\ 0 & (y_n w^T x_n > 1) \end{cases}$$

$$\cancel{\nabla E_{in}(w)} = -2 y_n x_n (\max((1 - y_n w^T x_n), 0))$$

$$\nabla E_{in}(w) = \begin{cases} -\frac{2}{N} \sum_{n=1}^{N} y_n x_n (1 - y_n w^T x_n) & (y_n w^T x_n \le 1) \\ 0 & (y_n w^T x_n > 1) \end{cases}$$

$$= -\frac{2}{N} \sum_{n=1}^{N} y_n x_n (\max(1 - y_n w^T x_n, 0))$$

Problem 2.

Consider a process that generates d-dimensional vectors $x_1, x_2 \dots x_N$ independently from a multivariate Gaussian distribution $\mathcal{N}(u, I)$, where $u \in R^d$ is an unknown parameter vector and $I \in R^{d \times d}$ is an identity matrix. The maximum likelihood estimate of u is

$$u^* = \arg\max_{u \in R^d} \prod_{n=1}^{N} P_u(x_n).$$

where P_u is the probability density function of $\mathcal{N}(u, I)$. Prove that

$$u^* = \frac{1}{N} \sum_{n=1}^{N} x_n.$$

Gaussian distribution (d)

$\mathcal{N}(u.I)$ definition $\Rightarrow$ $p_u(x_n) = \frac{1}{\sqrt{(2\pi)^d |I|}} e^{-\frac{1}{2}(x_n-u)^T I^{-1}(x_n-u)}$

$\left(x \sim \mathcal{N}(\mu.\Sigma) \quad f(x) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)} \right)$

maximum likelihood:

$u^* = \arg\max \prod_{n=1}^{N} p_u(x_n) \qquad u_0 = \arg\max \prod_{n=1}^{N} p_u(x_n)$

turn "multiply" to "add" use "log"

$\log u_0 = \sum_{n=1}^{N} \log p_u(x_n) \rightarrow$ find max

$\log p_u(x_n) = \log \frac{1}{\sqrt{(2\pi)^d |I|}} + \left[-\frac{1}{2}(x_n-u)^T \right]^{-1} (x_n-u)$

$= -\frac{1}{2}\log(2\pi)^d |I| - \frac{1}{2}(x_n-u)^T I^{-1}(x_n-u)$

$= -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|I| - \frac{1}{2}(x_n-u)^T I^{-1}(x_n-u)$

$\rightarrow$ find max

上述问题等价于 $-\log p_u(x_n)$ 的 min.

令 $p_u(x_n)$: $\frac{d}{2}\log 2\pi + \frac{1}{2}\log|I| + \frac{1}{2}(x_n-u)^T I^{-1}(x_n-u)$

$\sum \log p_u(x_n) = \frac{dN}{2}\log 2\pi + \frac{N}{2}\log|I| + \frac{1}{2}\sum (x_n-u)^T I^{-1}(x_n-u)$

固定值

对 u 求导. 令 $\sum (x_n-u)^T I^{-1}(x_n-u)$ 取最小, (I 为单位阵)

令 $g(u) = \sum (x_n-u)^T I^{-1}(x_n-u) = \sum \|x_n-u\|^2$

~~$\frac{\partial \sum \|x_n-u\|^2}{\partial u} = -\sum 2(x_n-u) \Rightarrow 0 \quad x_n = u$~~

$\frac{\partial g(u)}{\partial u} = -2\sum (x_n-u) = 0 \qquad \sum_{i=1}^{N}(x_n-u) = \sum_{i=1}^{N} x_n - Nu = 0$

即 $u = \frac{1}{N}\sum_{i=1}^{N} x_n \Rightarrow u^* = \frac{1}{N}\sum_{i=1}^{N} x_n$

Problem 3:

A classic binary classification data set that cannot be seperated by any line is called the XOR data set. with

$x = [x_1, x_2]$	y
$x_1 = [+1, +1]$	$y_1 = -1$
$x_2 = [-1, +1]$	$y_2 = +1$
$x_3 = [-1, -1]$	$y_3 = -1$
$x_4 = [+1, -1]$	$y_4 = +1$

You can see why it is called XOR by Interpreting +1 as a boolean value value of True and -1 as FALSE.

Consider a second-order feature transform $\Phi_2(x) = (1, x_1, x_2, x_1^2, x_1x_2, x_1^2)$ that converts the data set to

$z=\tilde{\Phi}_2(x)$	y
$z_1=\tilde{\Phi}_2(x_1)$	$y_1=-1$
$z_2=\tilde{\Phi}_2(x_2)$	$y_2=+1$
$z_3=\tilde{\Phi}_2(x_3)$	$y_3=-1$
$z_4=\tilde{\Phi}_2(x_4)$	$y_4=+1$

Show a perception $\hat{w}$ in the $\mathcal{Z}$-space that seperates the data. That is $y_n=\text{sign}(\tilde{w}^T z_n)$ for $n=1,2,3,4$.
Then, plot the classification boundary $\tilde{w}^T\tilde{\Phi}_2(x)=0$ in the $\mathcal{X}$-space. Your boundary should look like a quadratic curve that classifies x_1,x_2,x_3,x_4 perfectly.

解：将 $x_1\cdots x_4$ 代入 $z_1=z_4$（特征转换）

$$\begin{cases} z_1=\tilde{\Phi}_2([+1,+1])=(1,1,1,1,1,1)^T \Rightarrow y_1=-1 \\ z_2=\tilde{\Phi}_2([-1,+1])=(1,-1,1,1,-1,1)^T \Rightarrow y_2=+1 \\ z_3=\tilde{\Phi}_2([-1,-1])=(1,-1,-1,1,1,1)^T \Rightarrow y_3=-1 \\ z_4=\tilde{\Phi}_2([+1,-1])=(1,1,-1,1,-1,1)^T \Rightarrow y_4=+1 \end{cases}$$

$$\text{sign}(x)=\begin{cases} 1 & x>0 \\ 0 & x=0 \\ -1 & x<0 \end{cases}$$

先设 $\tilde{w}=(0,0,0,0,0,0)^T$

① $\tilde{w}^T z_1=0\neq(-1)$ 不同号 $\Rightarrow \tilde{w}=\tilde{w}+y_1z_1=(-1,-1,-1,-1,-1,-1)^T$

② $\tilde{w}^T z_2=-1+1-1-1+1-1=-2$ 与 $+1$ 不同号
$\Rightarrow \tilde{w}=\tilde{w}+y_2z_2=(0,-2,0,0,-2,0)^T$

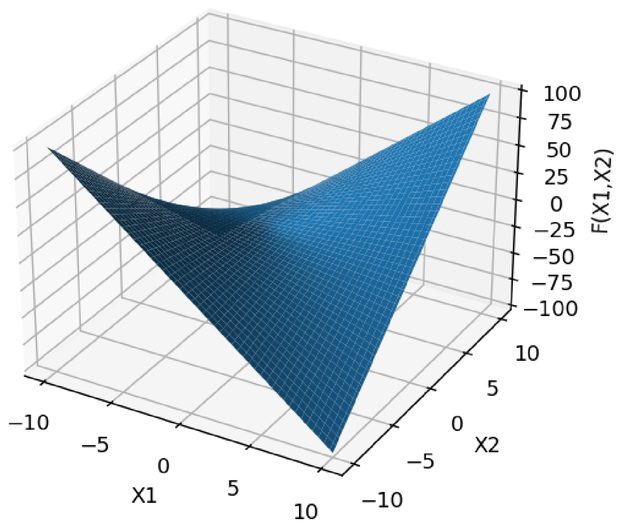
③ $\tilde{w}^T z_3=0+2+0+0\neq-2+0=0$ 与 (-1) 不同号
$\tilde{w}=\tilde{w}+y_3z_3=(-1,-\overset{-1}{0},1,-1,-\overset{3}{2},-1)^T$

④ $\tilde{w}^T z_4=-1-3-1-1+4-1=-3$ 与 y_4 不同号
$\tilde{w}=\tilde{w}+y_4z_4=(0,\overset{0}{-2},0,0,\overset{-4}{-6},0)^T$

权重向量 $\tilde{w}=(0,\overset{0}{-2},0,0,\overset{-4}{-6},0)^T$ 作用

$\tilde{w}^T\cdot\tilde{\Phi}_2(x)=-2x_1-6x_2=0 \Rightarrow x_1x_2=0$ 边界

$x_1=0$ 或 $x_2=0$



The left side shows the final image I drew using Python and related drawing libraries.

The mathematical expression for the image is below.

f(x1,x2)=x1*x2

Problem.4

Consider building binary classification with AdaBoost algorithm. a weak classifier $g_t(x)$ is trained on a data set $\{x_n,y_n\}_{n=1}^N$ with weights $\{w_n^t\}_{n=1}^N$ at the time step t. The error rate is defined as $\varepsilon_t=\dfrac{\sum_{n=1}^N w_n^t\cdot\delta(g_t(x_n),y_n)}{\sum_{n=1}^N w_n^t}$, where $\delta(g_t(x_n),y_n)=1$ if $g_t(x_n)\neq y_n$ and $\delta(g_t(x_n),y_n)=0$ otherwise. For the next time step, the data set is reweighted to emphasize on misclassified samples through the following rules

$$w_n^{t+1} = \begin{cases} w_n^t \cdot d_t & \text{if } g_t(x_n) \neq y_n \\ w_n^t / d_t & \text{if } g_t(x_n) = y_n \end{cases}$$

Show that $d_t = \sqrt{(1-\varepsilon_t)/\varepsilon_t}$ can degrade the previous classifier $g_t(x)$ and make its error rate become 0.5 with new weights $\{w_n^{t+1}\}_{n=1}^N$:

$$\frac{\sum_{n=1}^N w_n^{t+1} \delta(g_t(x_n), y_n)}{\sum_{n=1}^N w_n^{t+1}} = 0.5$$

证:

$$\varepsilon_t = \frac{\sum_{n=1}^N w_n^t \cdot \delta(g_t(x_n), y_n)}{\sum_{n=1}^N w_n^t}$$

$$w_n^{t+1} = \begin{cases} w_n^t \cdot d_t & (g_t(x_n) \neq y_n) \\ w_n^t / d_t & (g_t(x_n) = y_n) \end{cases} \qquad d_t = \sqrt{(1-\varepsilon_t)/\varepsilon_t}$$

$$d_t = \sqrt{(1-\varepsilon_t)/\varepsilon_t} = \sqrt{\frac{1}{\varepsilon_t} - 1}$$

$$w_n^{t+1} = w^t \cdot e$$

$$= \sqrt{\frac{\sum_{n=1}^N w_n^t}{\sum_{n=1}^N w_n^t \delta(g_t(x_n), y_n)} - 1}$$

$$= \sqrt{\frac{\sum w_n^t (1 - \delta(g_t(x_n), y_n))}{\sum_{n=1}^N w_n^t \delta(g_t(x_n), y_n)}} =$$

$$\frac{\sum_{n=1}^N w_n^{t+1} \delta(g_t(x_n), y_n)}{\sum_{n=1}^N w_n^{t+1}} = \frac{\sum_{n=1}^N w_n^t \cdot d_t \delta(g_t(x_n), y_n)}{\sum_{n=1}^N w_n^{t+1}} = \frac{\sum_{n=1}^N w_n^t \cdot e \delta((g_t(x_n), y_n)}{\sum_{n=1}^N w_n^{t+1}} \cdot d_t$$

$$= \frac{\sum_{n=1}^N w_n^t \delta(g_t(x_n), y_n)}{\sum_{n=1}^N w_n^t} \cdot \frac{\sum w_n^t}{\sum_n^{t+1}} \cdot d_t \qquad d_t \cdot \varepsilon_t \cdot \frac{\sum w_n^t}{\sum_n^{t+1}}$$

$$= \varepsilon_t \cdot \frac{\sum w_n^t}{\sum_n^{t+1}} \cdot \sqrt{(1-\varepsilon_t)/\varepsilon_t} = \sqrt{(1-\varepsilon_t)\varepsilon_t} \cdot \frac{\sum w_n^t}{\sum w_n^{t+1}}$$

$$\sum w_n^{t+1} d_t = \sum \left(w_n^t \cdot d_t \cdot \delta(g_t(x_n), y_n) + w_n^t \frac{d_t}{d_t} (1 - \delta(g_t(x_n), y_n)) \right)$$

$$= \sum w_n^t (d_t - 1) \delta(g_t(x_n), y_n) + \sum w_n^t$$

$$= (d_t - 1) \cdot \varepsilon_t \cdot \frac{\sum w_n^t}{d_t \sum w_n^t} + \frac{\sum w_n^t}{d_t}$$

$$\frac{\sum w_n^{t+1}}{\sum w_n} = \frac{(d_t^2 - 1) \cdot \varepsilon_t + 1}{d_t} = \frac{1 + (\frac{1-\varepsilon_t}{\varepsilon_t}) \cdot \varepsilon_t - \varepsilon_t}{d_t} = \frac{2 - \varepsilon_t}{\sqrt{(1-\varepsilon_t)/\varepsilon_t}}$$

$$\alpha_t = \sqrt{(1-\varepsilon_t)\varepsilon_t} \cdot d_t \cdot \varepsilon_t \cdot \frac{d_t}{(d_t^2 - 1)\varepsilon_t + 1} = \frac{d_t^2 \cdot \varepsilon_t}{(d_t^2 - 1)\varepsilon_t + 1}$$

$$= \frac{d_t^2}{d_t^2 - 1 + \frac{1}{\varepsilon_t}} = \frac{\frac{1}{\varepsilon_t} - 1}{\frac{1}{\varepsilon_t} - 2} = \frac{1}{2} = 0.5$$

$$= \frac{1 - \varepsilon_t}{1 - \varepsilon_t - \varepsilon_t + 1} = \frac{1 - \varepsilon_t}{2 - 2\varepsilon_t} = \frac{1}{2}$$

格式 Report

见后