NTUEE, WAN - CYUAN FAN 范萬泉

"no collaborators, with reference"

Homework 1

Problem 1 :

ML tech. HW2 B04502105 范萬泉

Descent Methods for Probabilistic SVM

1. $\min_{A,B} F(A,B) = \frac{1}{N}\sum_{n=1}^{N} \ln\left(1+\exp\left(-y_n\left(A\cdot\left(w_{SVM}^T\phi(x_n)+b_{SVM}\right)+B\right)\right)\right)$

Let $z_n = w_{SVM}^T\phi(x_n)+b_{SVM}$, $p_n = \theta(-y_n(Az_n+B))$, $\theta(s) = \frac{\exp(s)}{1+\exp(s)}$

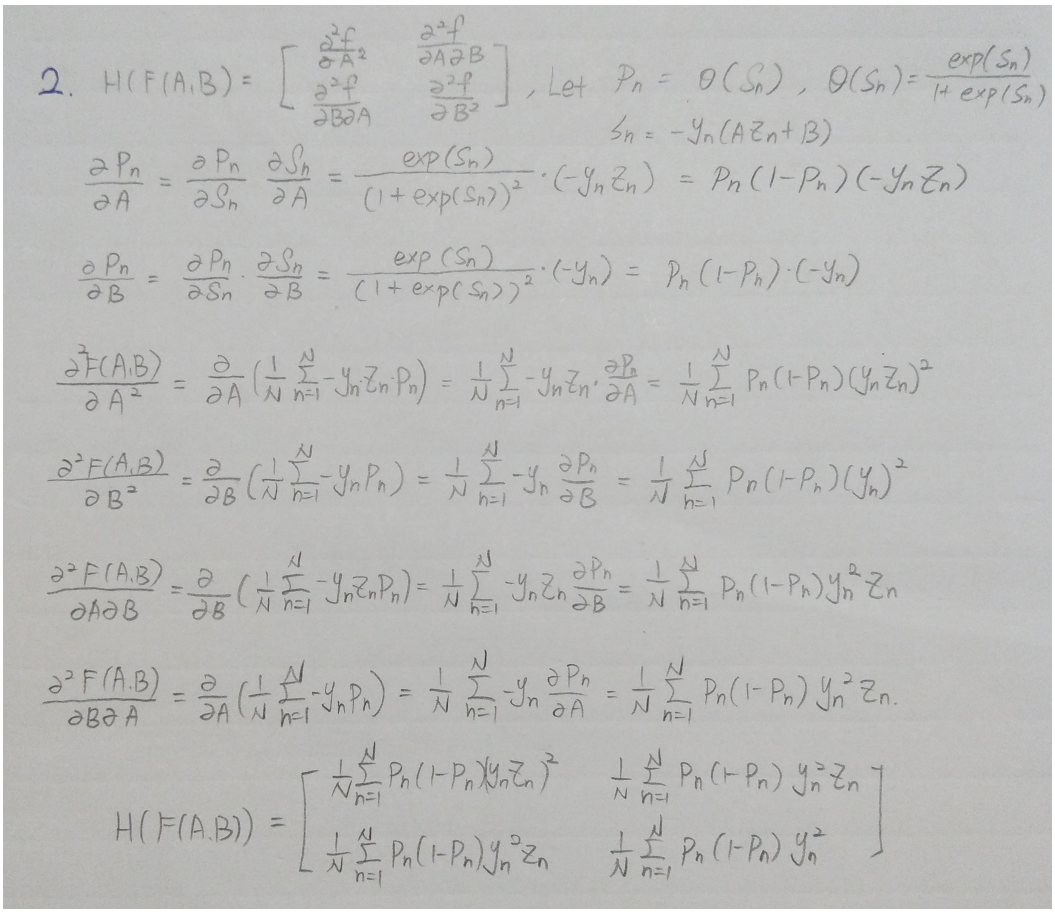
$$\Rightarrow \min_{A,B} F(A,B) = \frac{1}{N}\sum_{n=1}^{N} \ln\left(1+\exp(-y_n(Az_n+B))\right)$$

$$\frac{\partial F(A,B)}{\partial A} = \frac{1}{N}\sum_{n=1}^{N} \frac{-y_n z_n \exp(-y_n(Az_n+B))}{1+\exp(-y_n(Az_n+B))} = \frac{1}{N}\sum_{n=1}^{N} -y_n z_n p_n$$

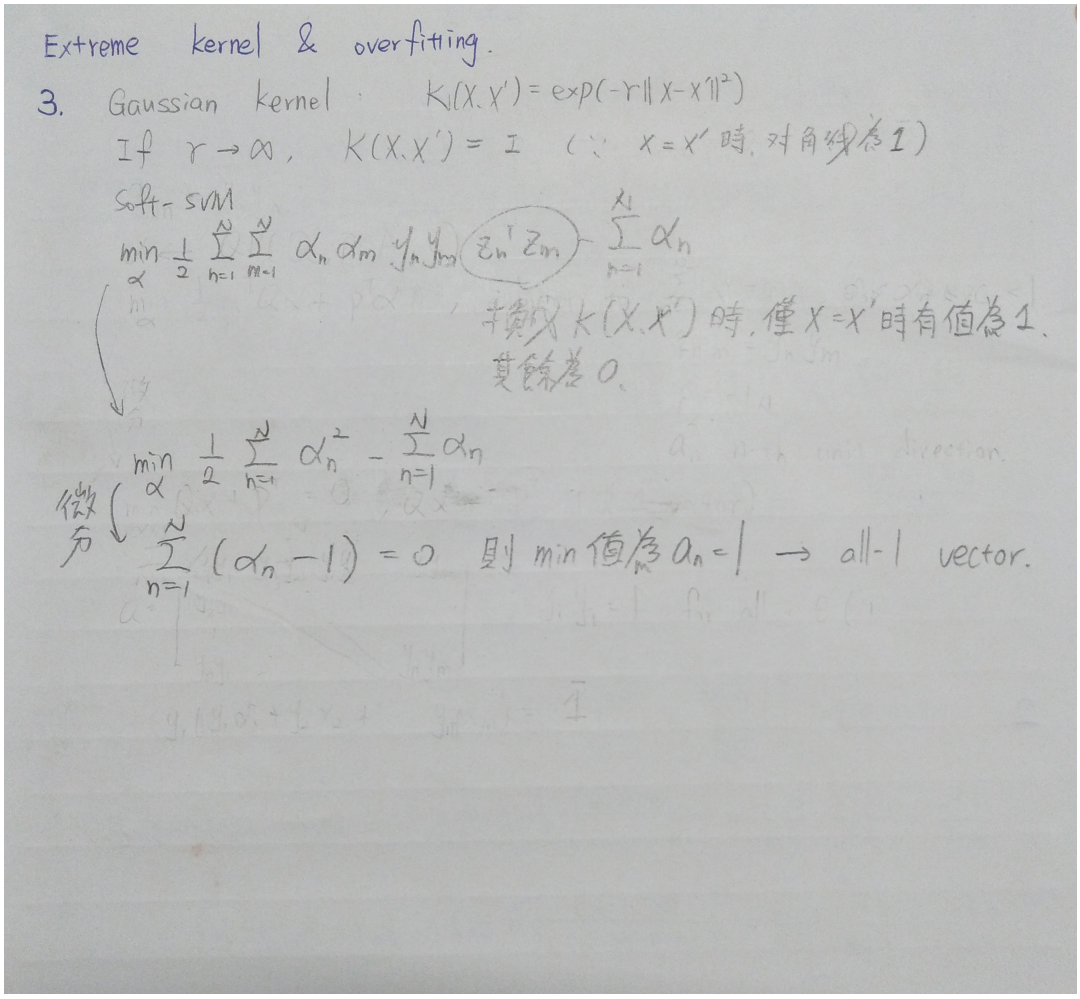
$$\frac{\partial F(A,B)}{\partial B} = \frac{1}{N}\sum_{n=1}^{N} \frac{-y_n \cdot \exp(-y_n(Az_n+B))}{1+\exp(-y_n(Az_n+B))} = \frac{1}{N}\sum_{n=1}^{N} -y_n p_n$$

$$\nabla F(A,B) = \frac{1}{N}\sum_{n=1}^{N} p_n y_n \cdot (-z_n \hat{e}_A + -\hat{e}_B)$$

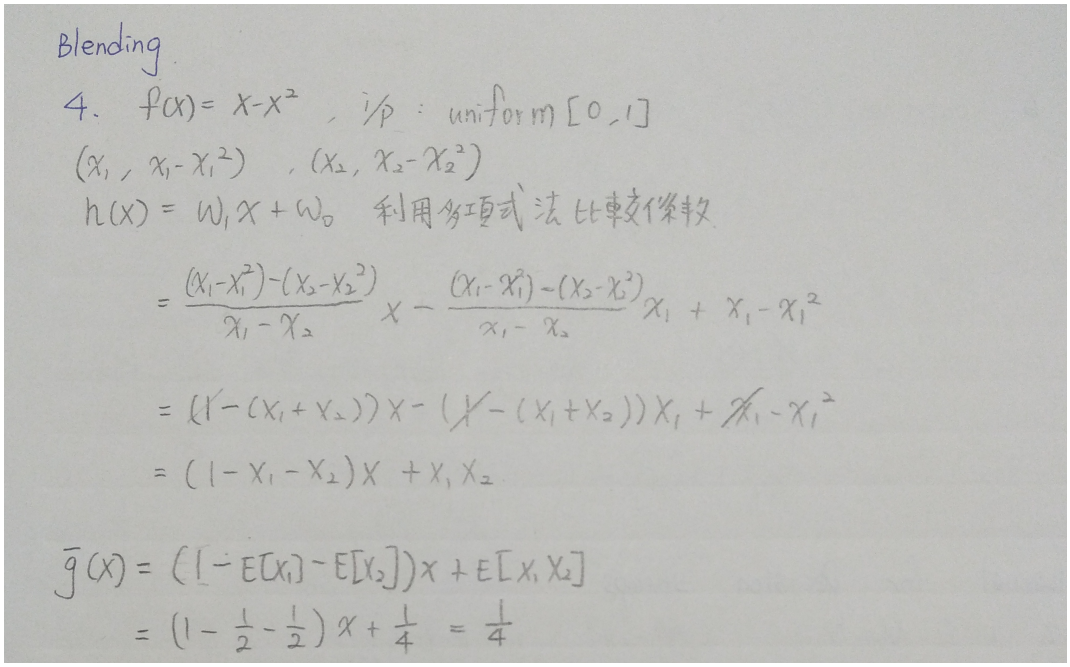
Problem 2 :


2. H(F(A,B)) = [∂²f/∂A² ∂²f/∂A∂B ; ∂²f/∂B∂A ∂²f/∂B²], Let Pn = θ(Sn), θ(Sn) = exp(Sn)/(1+exp(Sn))
Sn = -yn(Azn+B)
∂Pn/∂A = ∂Pn/∂Sn · ∂Sn/∂A = exp(Sn)/(1+exp(Sn))² · (-yn zn) = Pn(1-Pn)(-yn zn)
∂Pn/∂B = ∂Pn/∂Sn · ∂Sn/∂B = exp(Sn)/(1+exp(Sn))² · (-yn) = Pn(1-Pn)·(-yn)
∂²F(A,B)/∂A² = ∂/∂A (1/N Σ_{n=1}^{N} -yn zn Pn) = 1/N Σ -yn zn · ∂Pn/∂A = 1/N Σ Pn(1-Pn)(yn zn)²
∂²F(A,B)/∂B² = ∂/∂B (1/N Σ_{n=1}^{N} -yn Pn) = 1/N Σ -yn ∂Pn/∂B = 1/N Σ Pn(1-Pn)(yn)²
∂²F(A,B)/∂A∂B = ∂/∂B (1/N Σ_{n=1}^{N} -yn zn Pn) = 1/N Σ -yn zn ∂Pn/∂B = 1/N Σ Pn(1-Pn) yn² zn
∂²F(A,B)/∂B∂A = ∂/∂A (1/N Σ_{n=1}^{N} -yn Pn) = 1/N Σ -yn ∂Pn/∂A = 1/N Σ Pn(1-Pn) yn² zn.
H(F(A,B)) = [1/N Σ_{n=1}^{N} Pn(1-Pn)(yn zn)² 1/N Σ Pn(1-Pn) yn² zn ; 1/N Σ Pn(1-Pn) yn² zn 1/N Σ Pn(1-Pn) yn²]

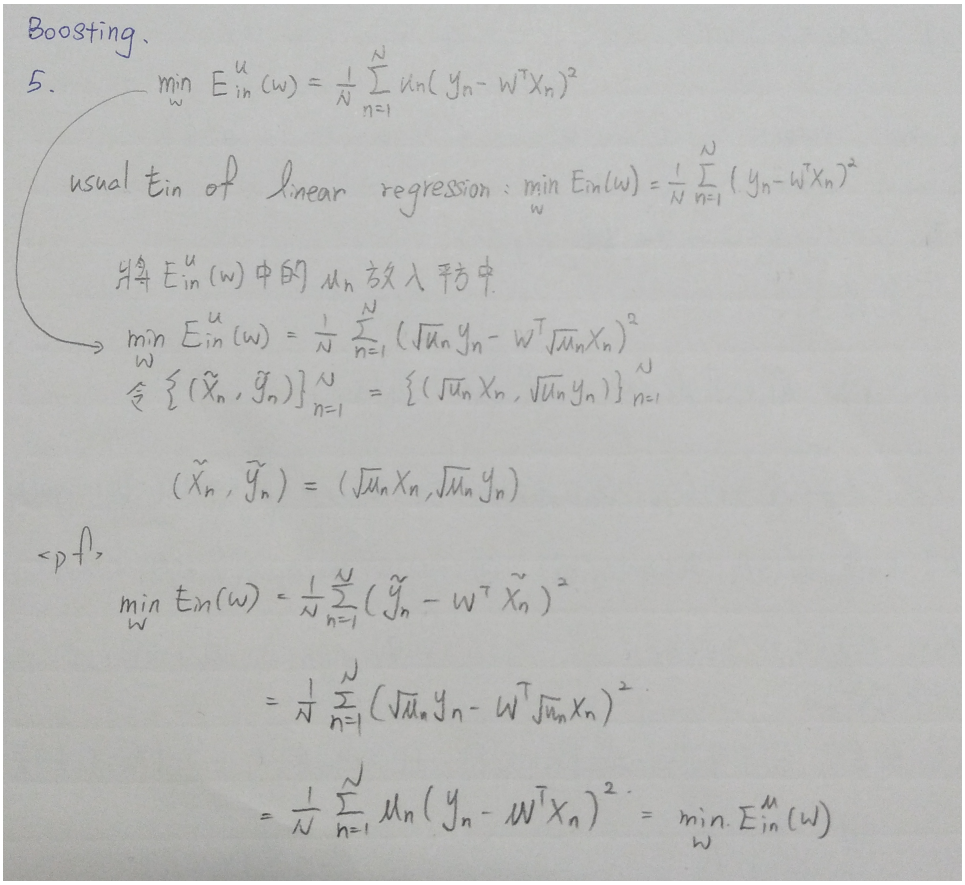
Problem 3 :


Extreme kernel & overfitting.
3. Gaussian kernel : $K(X,X') = \exp(-\gamma\|X-X'\|^2)$
If $\gamma \to \infty$, $K(X,X') = I$ (∵ $X=X'$ 時, 对角线为1)
Soft-SVM
$\min_{\alpha} \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m z_n^T z_m - \sum_{n=1}^{N} \alpha_n$
换成 $K(X,X')$ 時, 僅 $X=X'$ 時有值为1.
其餘为0.
$\min_{\alpha} \frac{1}{2} \sum_{n=1}^{N} \alpha_n^2 - \sum_{n=1}^{N} \alpha_n$
微分 $\sum_{n=1}^{N} (\alpha_n - 1) = 0$ 則 min 值为 $\alpha_n = 1$ → all-1 vector.

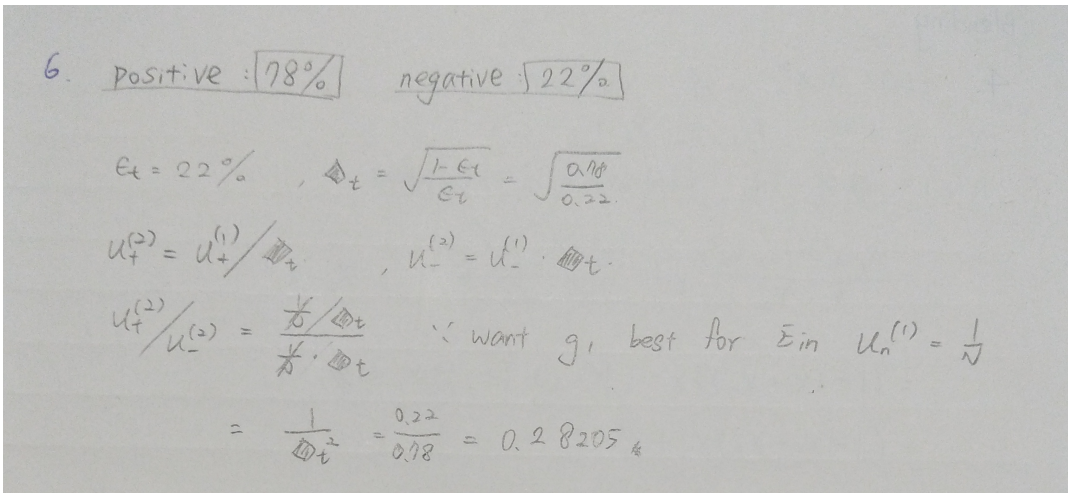
Problem 4 :



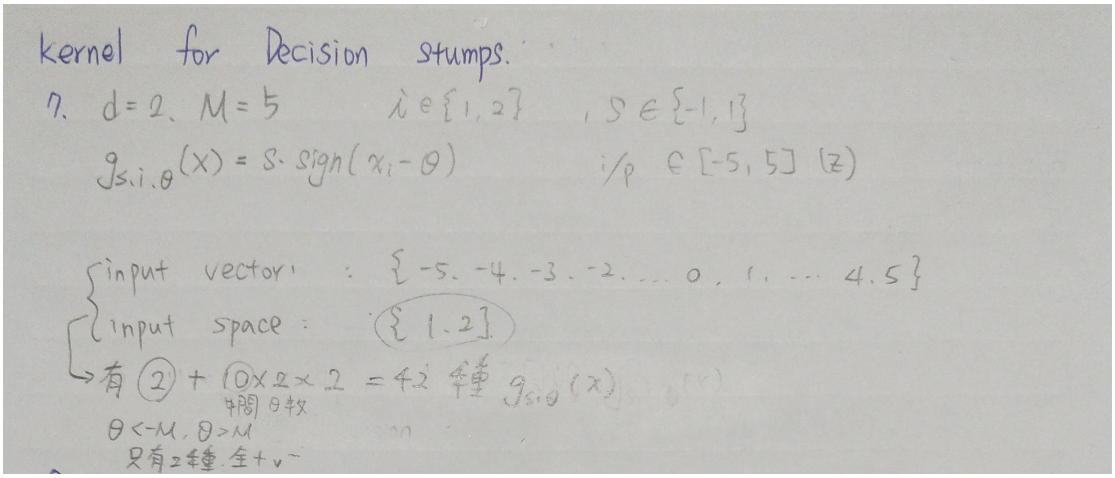
Problem 5 :



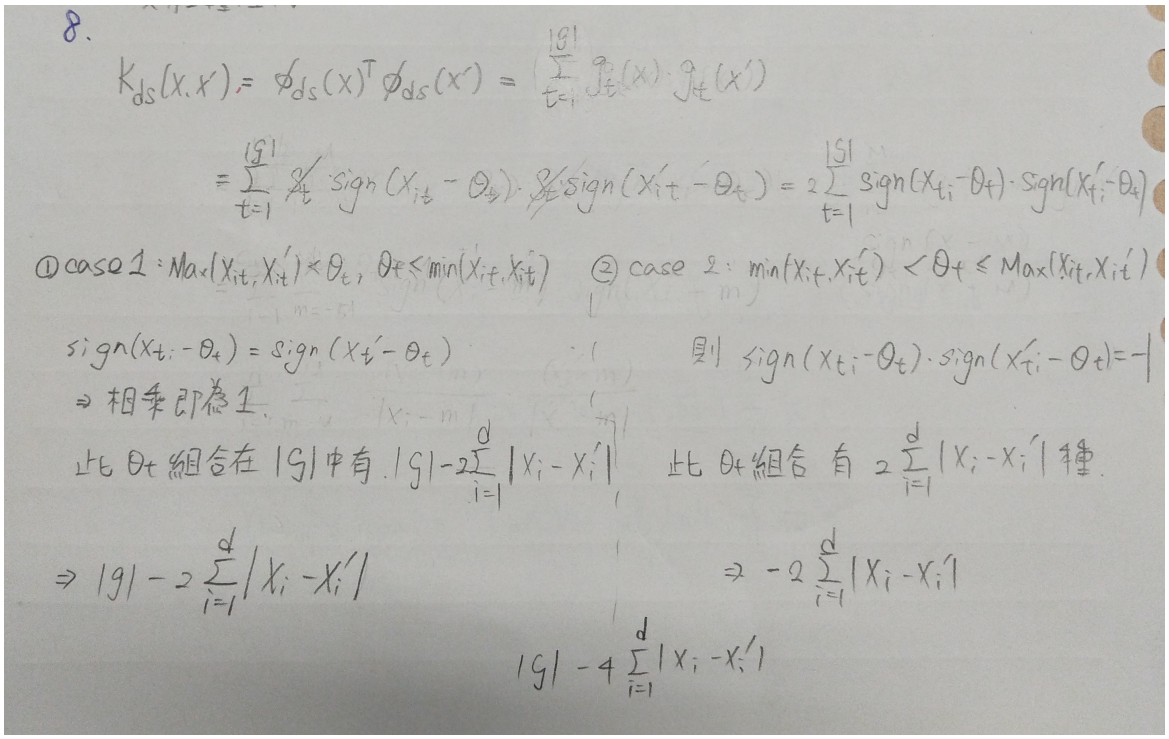
Problem 6 :



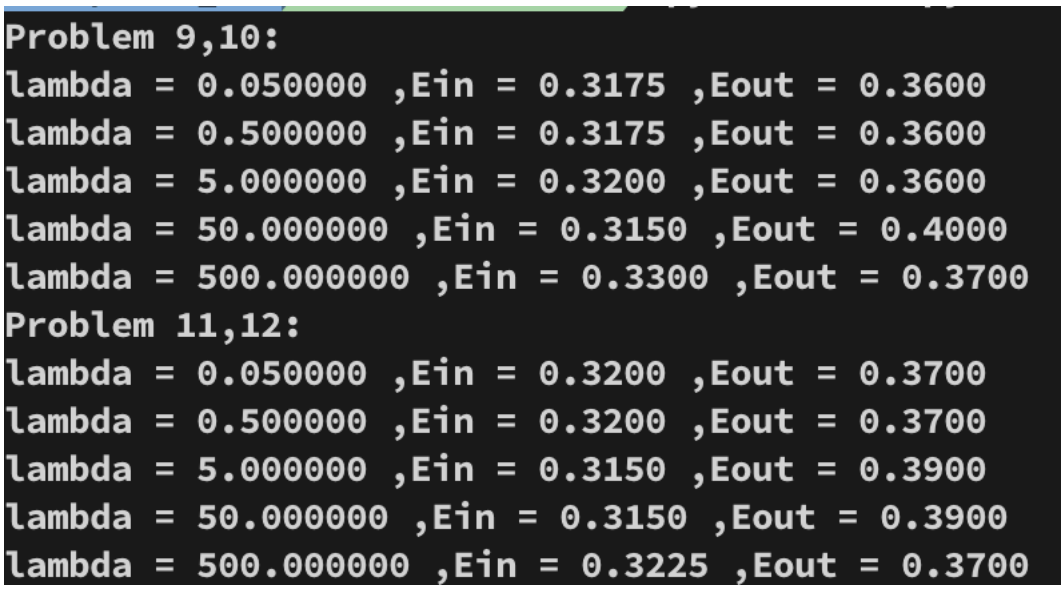
Problem 7 :



Problem 8 :


8.
$K_{ds}(x,x') = \phi_{ds}(x)^T \phi_{ds}(x') = \sum_{t=1}^{|G|} g_t(x) \cdot g_t(x')$
$= \sum_{t=1}^{|G|} s_t \cdot sign(x_{it} - \theta_t) \cdot s_t \cdot sign(x'_{it} - \theta_t) = 2 \sum_{t=1}^{|G|} sign(x_{ti} - \theta_t) \cdot sign(x'_{ti} - \theta_t)$
① case 1 : $Max(x_{it}, x'_{it}) < \theta_t$, $\theta_t \leq min(x_{it}, x'_{it})$
② case 2 : $min(x_{it}, x'_{it}) < \theta_t \leq Max(x_{it}, x'_{it})$
$sign(x_{ti} - \theta_t) = sign(x'_{ti} - \theta_t)$
則 $sign(x_{ti} - \theta_t) \cdot sign(x'_{ti} - \theta_t) = -1$
⇒ 相乘即為 1.
此 θ_t 組合在 $|G|$ 中有 $|G| - 2\sum_{i=1}^{d} |x_i - x'_i|$
此 θ_t 組合 有 $2\sum_{i=1}^{d} |x_i - x'_i|$ 種.
$\Rightarrow |G| - 2\sum_{i=1}^{d} |x_i - x'_i|$
$\Rightarrow -2\sum_{i=1}^{d} |x_i - x'_i|$
$|G| - 4\sum_{i=1}^{d} |x_i - x'_i|$

Problem 9 ,10 ,11 ,12 :



9. Lambda = 50 的時候有最小值Ein(g) = 0.3150。

10. Lambda = 0.05 , 0.5 , 5 三個數字的時候有最小值Eout(g) = 0.3600。

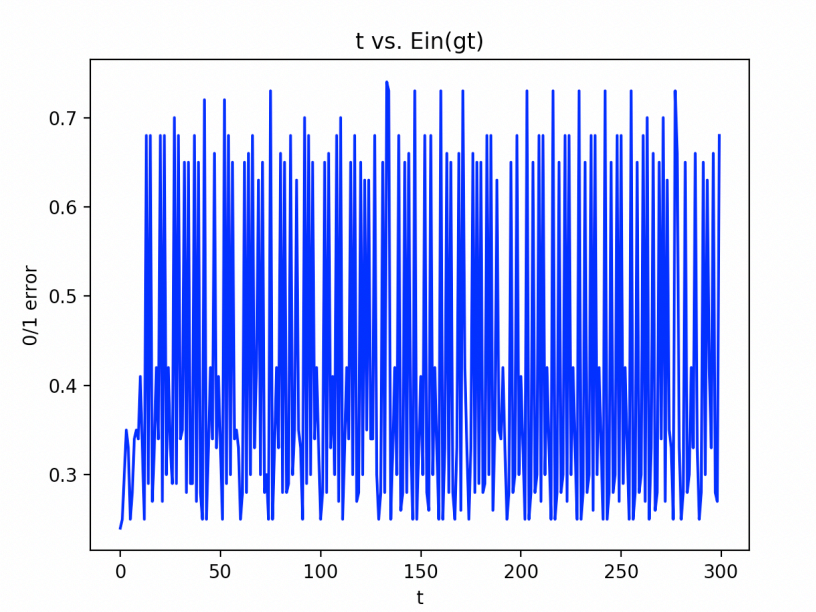
11.Lambda = 5 , 50 的時候有最小值Ein(G) = 0.3150。

經過bagging後，對於λ 在5,50者，確實有使得Ein下降的情況，但基本上都不是下降很多，表示多個g結合後計算的分隔線，實際上可能和原本在第9題一個g形成的linear 差不多。

12. Lambda = 0.05 , 0.5 , 500 三個數字的時候有最小值Eout(g) = 0.3700。

經過bagging，我們發現Eout(G)反而變大，相對於前者0.36上升0.01，表示多個結合的G形成之model，反而有點overfitting在我們的training data上。

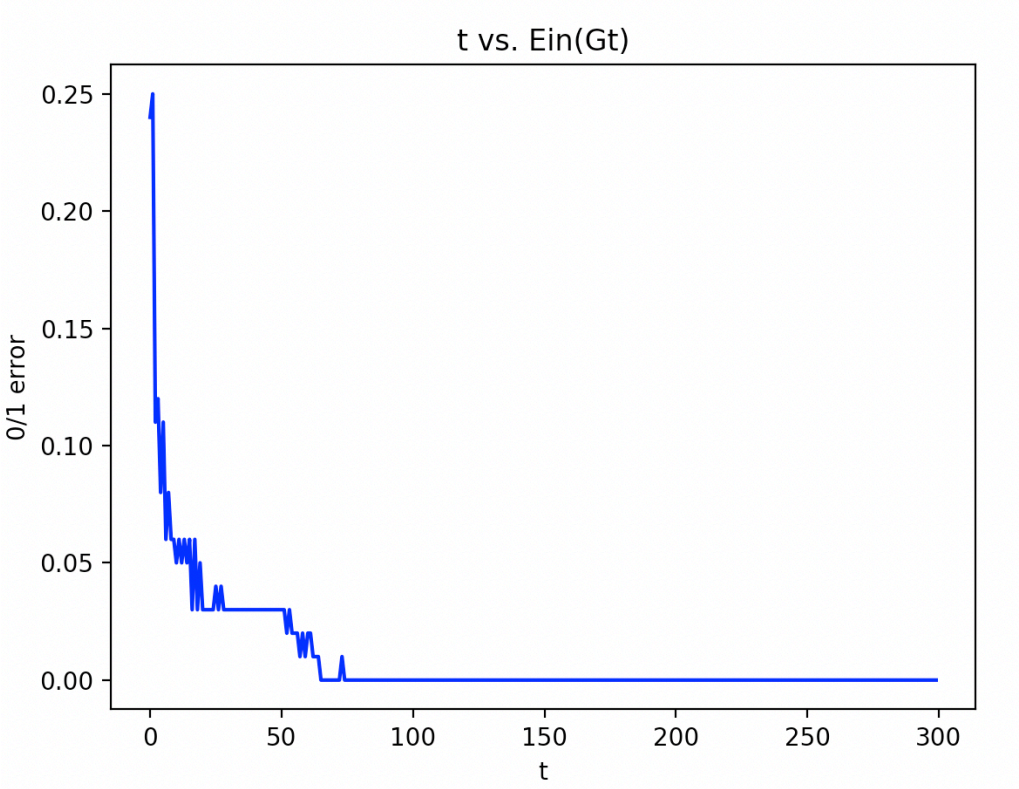
Problem 13 :



$E_{in}(g_T) = 0.68$

我們發現Ein值基本上時好時壞，並沒有一個穩定的遞減/增方向，因為每一次g(t)決定都是找所有可能切法中最好的，不會有累積前面所有g(t)的效果。

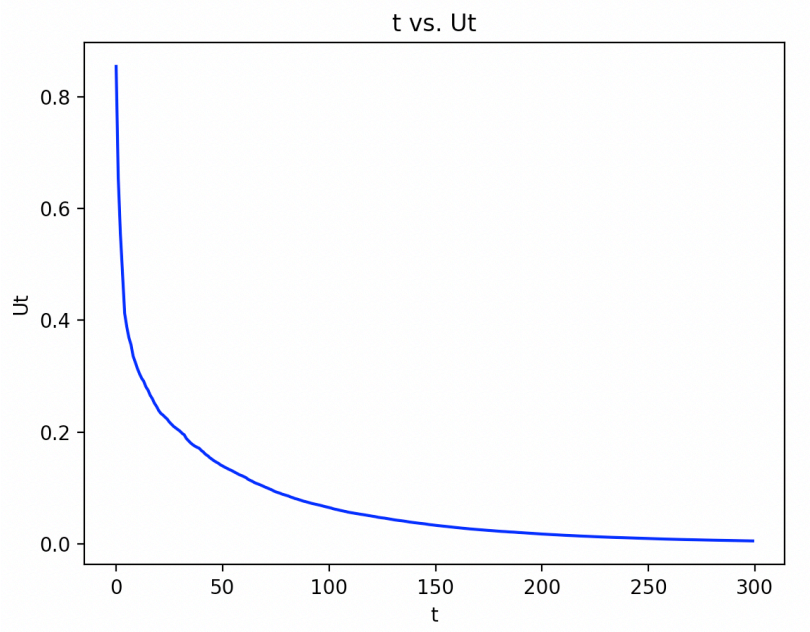
Problem 14 :



$E_{in}(G_T) = 0.0$

這題我們將前面的g(t)累積計算求出對應的G(t)，從實驗結果看出來，Ein數值不斷下降，直到完全沒有錯誤。這是因為我們累計了前面所有的g(t)因此就像講義上寫，會有更細的非線性切割來符合模型。

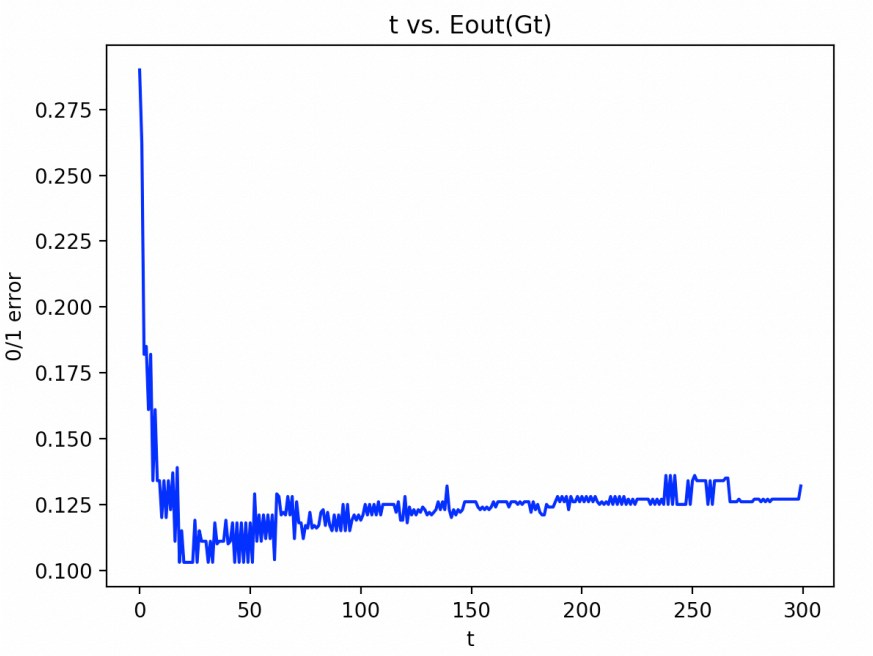
Problem 15 :



U_T : 0.005401486582613612

這裡的大U是將前面的u總和在每一個iteration記錄一次，我們知道此演算法會每次修正u值，如果正確的分類，我們把該data對應到的u值變小，因此合理我們每次g(t)越來越多結合成Gt時，分類效果會越來越好，也就導致對的分類越來越多，u值也隨之不斷下降。

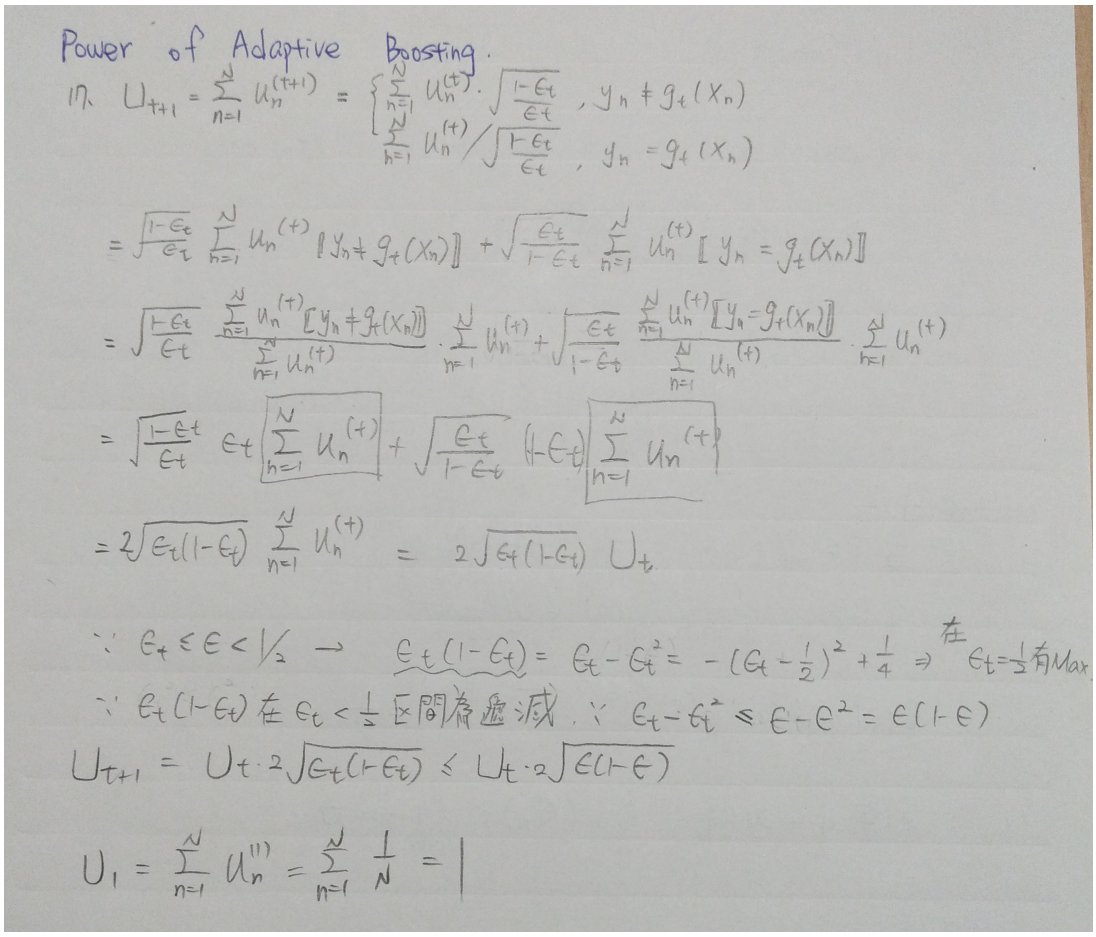
Problem 16:



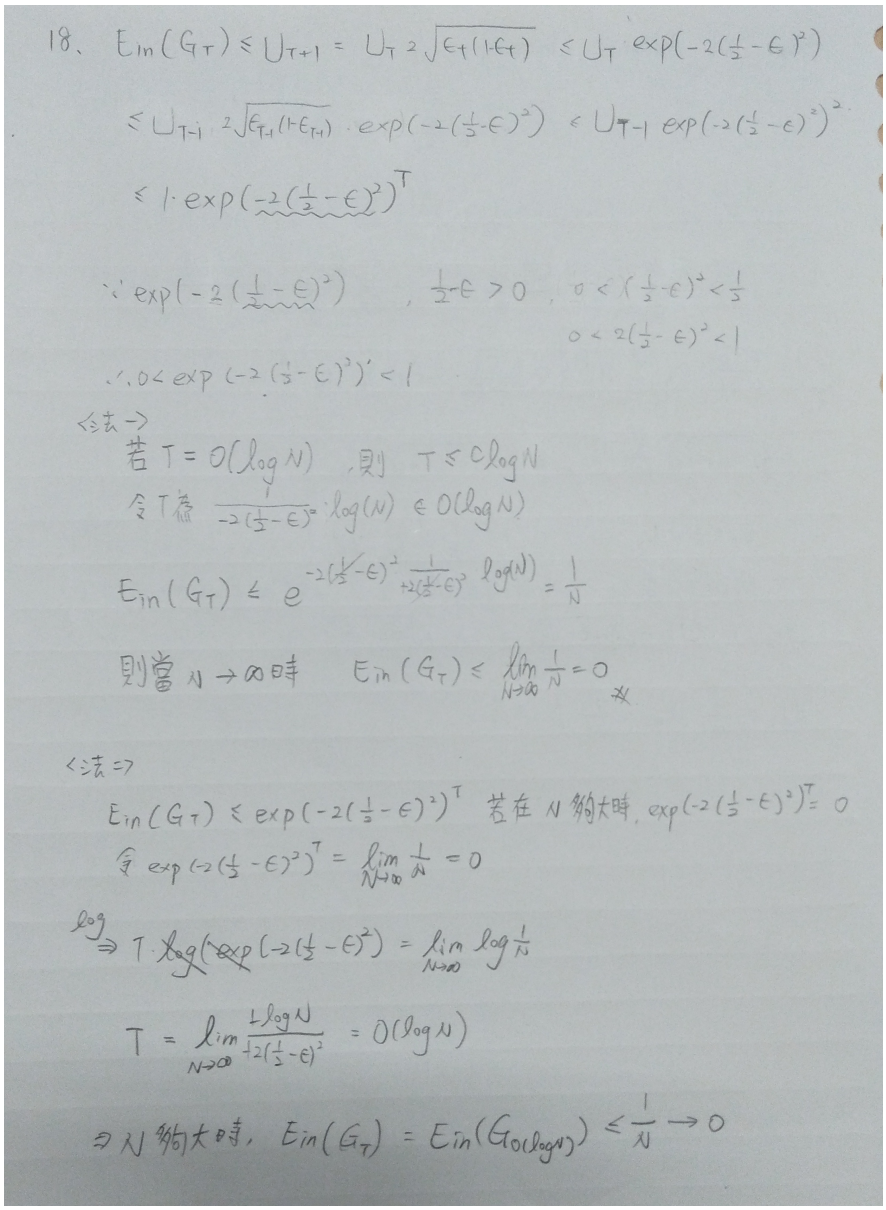
$E_{out}(G_T) = 0.132$

這階段我們做testing，也看出來整體error數值不斷下降，大概在t=40左右有最小值出現，在那之後又持續緩慢的上升，推測應該是model overfitting 在training data上頭，使得分類效果對於test data沒有那麼佳。

Problem 17 :


Power of Adaptive Boosting.
17. $U_{t+1} = \sum_{n=1}^{N} u_n^{(t+1)} = \begin{cases} \sum_{n=1}^{N} u_n^{(t)} \cdot \sqrt{\frac{1-\epsilon_t}{\epsilon_t}}, & y_n \neq g_t(x_n) \\ \sum_{n=1}^{N} u_n^{(t)} / \sqrt{\frac{1-\epsilon_t}{\epsilon_t}}, & y_n = g_t(x_n) \end{cases}$
$= \sqrt{\frac{1-\epsilon_t}{\epsilon_t}} \sum_{n=1}^{N} u_n^{(t)} [\![y_n \neq g_t(x_n)]\!] + \sqrt{\frac{\epsilon_t}{1-\epsilon_t}} \sum_{n=1}^{N} u_n^{(t)} [\![y_n = g_t(x_n)]\!]$
$= \sqrt{\frac{1-\epsilon_t}{\epsilon_t}} \frac{\sum_{n=1}^{N} u_n^{(t)} [\![y_n \neq g_t(x_n)]\!]}{\sum_{n=1}^{N} u_n^{(t)}} \cdot \sum_{n=1}^{N} u_n^{(t)} + \sqrt{\frac{\epsilon_t}{1-\epsilon_t}} \frac{\sum_{n=1}^{N} u_n^{(t)} [\![y_n = g_t(x_n)]\!]}{\sum_{n=1}^{N} u_n^{(t)}} \cdot \sum_{n=1}^{N} u_n^{(t)}$
$= \sqrt{\frac{1-\epsilon_t}{\epsilon_t}} \epsilon_t \left[\sum_{n=1}^{N} u_n^{(t)} \right] + \sqrt{\frac{\epsilon_t}{1-\epsilon_t}} (1-\epsilon_t) \left[\sum_{n=1}^{N} u_n^{(t)} \right]$
$= 2\sqrt{\epsilon_t(1-\epsilon_t)} \sum_{n=1}^{N} u_n^{(t)} = 2\sqrt{\epsilon_t(1-\epsilon_t)}\, U_t$
$\because \epsilon_t \leq \epsilon < 1/2 \rightarrow \epsilon_t(1-\epsilon_t) = \epsilon_t - \epsilon_t^2 = -(\epsilon_t - \frac{1}{2})^2 + \frac{1}{4} \Rightarrow$ 在 $\epsilon_t = \frac{1}{2}$ 有 Max
$\because \epsilon_t(1-\epsilon_t)$ 在 $\epsilon_t < \frac{1}{2}$ 區間為遞減 $\because \epsilon_t - \epsilon_t^2 \leq \epsilon - \epsilon^2 = \epsilon(1-\epsilon)$
$U_{t+1} = U_t \cdot 2\sqrt{\epsilon_t(1-\epsilon_t)} \leq U_t \cdot 2\sqrt{\epsilon(1-\epsilon)}$
$U_1 = \sum_{n=1}^{N} u_n^{(1)} = \sum_{n=1}^{N} \frac{1}{N} = 1$

Problem 18 :


18. $E_{in}(G_T) \le U_{T+1} = U_T \cdot 2\sqrt{\epsilon_t(1-\epsilon_t)} \le U_T \exp(-2(\frac{1}{2}-\epsilon)^2)$
$\le U_{T-1} \cdot 2\sqrt{\epsilon_{t+1}(1-\epsilon_{t+1})} \cdot \exp(-2(\frac{1}{2}-\epsilon)^2) \le U_{T-1} \exp(-2(\frac{1}{2}-\epsilon)^2)^2$
$\le 1 \cdot \exp(-2(\frac{1}{2}-\epsilon)^2)^T$
$\because \exp(-2(\frac{1}{2}-\epsilon)^2)$, $\frac{1}{2}-\epsilon > 0$, $0 < (\frac{1}{2}-\epsilon)^2 < \frac{1}{2}$
$0 < 2(\frac{1}{2}-\epsilon)^2 < 1$
$\therefore 0 < \exp(-2(\frac{1}{2}-\epsilon)^2)^T < 1$
<法一>
若 $T = O(\log N)$,則 $T \le C\log N$
令 T 為 $\frac{}{-2(\frac{1}{2}-\epsilon)^2} \cdot \log(N) \in O(\log N)$
$E_{in}(G_T) \le e^{-2(\frac{1}{2}-\epsilon)^2 \frac{1}{+2(\frac{1}{2}-\epsilon)^2} \log(N)} = \frac{1}{N}$
則當 $N \to \infty$ 時 $E_{in}(G_T) \le \lim_{N\to\infty} \frac{1}{N} = 0$ 次
<法二>
$E_{in}(G_T) \le \exp(-2(\frac{1}{2}-\epsilon)^2)^T$ 若在 N 夠大時, $\exp(-2(\frac{1}{2}-\epsilon)^2)^T = 0$
令 $\exp(-2(\frac{1}{2}-\epsilon)^2)^T = \lim_{N\to\infty} \frac{1}{N} = 0$
log $\Rightarrow T \cdot \log(\exp(-2(\frac{1}{2}-\epsilon)^2)) = \lim_{N\to\infty} \log\frac{1}{N}$
$T = \lim_{N\to\infty} \frac{1\log N}{+2(\frac{1}{2}-\epsilon)^2} = O(\log N)$
$\Rightarrow$ N 夠大時, $E_{in}(G_T) = E_{in}(G_{O(\log N)}) \le \frac{1}{N} \to 0$

My Code is available on my Github :

"https://github.com/davidhalladay/Computer-vision-project/tree/master/mini_02"

reference:

1. *SVM , "https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html"*
2. *Matplotlib.contour , "https://matplotlib.org/api/_as_gen/matplotlib.pyplot.contour.html"*
3. *SVM.math , "https://scikit-learn.org/stable/modules/svm.html#svm-kernels"*
4. *SVM parameters details , "http://www.stardustsky.net/index.php/post/53.html"*
5. *Train_test_split , "https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html"*
6. *Mever's theorem , "https://en.wikipedia.org/wiki/Mercer%27s_theorem"*
7. *Construct kernel func. , "http://mlweb.loria.fr/book/en/constructingkernels.html"*
8. *Standford SVM , "http://cs229.stanford.edu/notes/cs229-notes3.pdf"*
9. *Kernel RR , "https://scikit-learn.org/stable/modules/generated/sklearn.kernel_ridge.KernelRidge.html#sklearn.kernel_ridge.KernelRidge"*