CSE512 Fall 2018 - Machine Learning - Homework 5

Name: Manideep Attanti
Solar Id: 112028167
Netid email: manideep.attanti@stonybrook.edu

(1.1) $H(x) = \text{sgn}\{f(x)\}$

$$f(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$$

To show:

$$\epsilon_{Training} = \frac{1}{N}\sum_{j=1}^{N} \delta(H(x^j \neq y^j)) \leq \frac{1}{N}\sum_{j=1}^{N} \exp(-f(x^j) y^j$$

$$\delta(H(x^j) \neq y^j) = \begin{cases} 1 & \text{if } H(x^j) \neq y^j \\ 0 & \text{otherwise} \end{cases}$$

$$H(x^j) = \text{sgn}\{f(x^j)\}$$

$$\therefore \delta(H(x^j) \neq y^j) = \begin{cases} 1 & \text{if } y^j \cdot f(x^j) < 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\therefore \epsilon_{Training} = \frac{1}{N}\sum_{j=1}^{N} \begin{cases} 1 & \text{if } y^j \cdot f(x^j) < 0 \\ 0 & \text{otherwise} \end{cases}$$

Now $\exp(-f(x^j) \cdot y^j) \geq 1$ if $y^j \cdot f(x^j) < 0$

$\because \exp(+ve) \geq 1$

and $\exp(-f(x^j) \cdot y^j) \geq 0$ if $y^j \cdot f(x^j) > 0$

Hence,

$$\epsilon_{Training} = \frac{1}{N}\sum_{j=1}^{N} \delta(H(x^j) \neq y^j) \leq \frac{1}{N}\sum_{j=1}^{N} \exp(-f(x^j) y^j)$$

(1.2) $$w_j^{(t+1)} = \frac{w_j^{(t)} \exp(-\alpha_t y^j h_t(x^j))}{Z_t}$$

$$Z_t = \sum_{j=1}^{N} w_j^{(t)} \exp(-\alpha_t y^j h_t(x^j))$$

To show: $\frac{1}{N} \sum_{j=1}^{N} \exp(-f(x^j) y^j) = \prod_{t=1}^{T} Z_t$

$$f(x^j) = \sum_{t=1}^{T} \alpha_t h_t(x^j)$$

$$w_j^{(t+1)} = w_j^{(t)} \cdot \frac{\exp(-\alpha_t y^j h_t(x^j))}{Z_t}$$

$$= w_j^{(1)} \cdot \prod_{t=1}^{T} \frac{\exp(-\alpha_t y^j h_t(x^j))}{Z_t}$$

All w_j's are initialized to $\frac{1}{N}$

$$\therefore w_j^{(t+1)} = \frac{1}{N} \cdot \frac{\exp\left(\sum_{t=1}^{T} -\alpha_t y^j h_t(x^j)\right)}{\prod_{t=1}^{T} Z_t}$$

$$= \frac{1}{N} \cdot \frac{\exp\left(-y^j \sum_{t=1}^{T} \alpha_t h_t(x^j)\right)}{\prod_{t=1}^{T} Z_t}$$

$$= \frac{1}{N} \cdot \frac{\exp(-y^j f(x^j))}{\prod_{t=1}^{T} Z_t}$$

$$\sum_{j=1}^{N} w_j^{(t+1)} = 1$$

$$\Rightarrow \frac{1}{N} \frac{\sum_{j=1}^{N} \exp(-y^j f(x^j))}{\prod_{t=1}^{T} z_t} = 1$$

$$\Rightarrow \frac{1}{N} \sum_{j=1}^{N} \exp(-f(x^j) y^j) = \prod_{t=1}^{T} z_t$$

(1.3)
(a) $z_t = (1-\epsilon_t)\exp(-\alpha_t) + \epsilon_t \exp(\alpha_t)$

To minimize z_t w.r.t α_t

$$\frac{\partial z_t}{\partial \alpha_t} = 0$$

$$\Rightarrow -(1-\epsilon_t)\exp(-\alpha_t) + \epsilon_t \exp(\alpha_t) = 0$$

$$\Rightarrow \epsilon_t \exp(\alpha_t) = (1-\epsilon_t)\exp(-\alpha_t)$$

$$\Rightarrow (\exp(\alpha_t))^2 = \frac{1-\epsilon_t}{\epsilon_t}$$

$$\Rightarrow \exp(\alpha_t) = \sqrt{\frac{1-\epsilon_t}{\epsilon_t}}$$

$$\Rightarrow \boxed{\alpha_t = \ln\left(\sqrt{\frac{1-\epsilon_t}{\epsilon_t}}\right)}$$

$$z_t^{opt} = (1-\epsilon_t)\exp\left(-\ln\sqrt{\frac{1-\epsilon_t}{\epsilon_t}}\right) + \epsilon_t \exp\left(\ln\sqrt{\frac{1-\epsilon_t}{\epsilon_t}}\right)$$

$$= (1-\epsilon_t)\cdot\sqrt{\frac{\epsilon_t}{1-\epsilon_t}} + \epsilon_t \cdot \sqrt{\frac{1-\epsilon_t}{\epsilon_t}}$$

$$= \sqrt{\epsilon_t(1-\epsilon_t)} + \sqrt{\epsilon_t(1-\epsilon_t)}$$

$$= 2\sqrt{\epsilon_t(1-\epsilon_t)}$$

$$\boxed{\therefore\ Z_t^{opt} = 2\sqrt{\epsilon_t(1-\epsilon_t)}}$$

(b) $\epsilon_t = \frac{1}{2} - \gamma_t$

$$Z_t = Z_t^{opt} = 2\sqrt{\epsilon_t(1-\epsilon_t)}$$

$$= 2\sqrt{\left(\frac{1}{2}-\gamma_t\right)\left(\frac{1}{2}+\gamma_t\right)}$$

$$= 2\sqrt{\frac{1}{4}-\gamma_t^2}$$

$$= \sqrt{1-4\gamma_t^2}$$

$$1+x \leq e^x \quad \forall x \in \mathbb{R}$$

$\because$ $1+x$ is a tangent to e^x at $x=0$ and since e^x is convex, it will always be above its tangent

$$\Rightarrow 1+(-x) \leq e^{-x}$$

$$\Rightarrow 1-x \leq e^{-x}$$

$$\Rightarrow \sqrt{1-x} \leq e^{-x/2}$$

$$\Rightarrow \sqrt{1-4\gamma_t^2} \leq e^{-2\gamma_t^2}$$

$$\Rightarrow \boxed{Z_t \leq \exp(-2\gamma_t^2)}$$

(c) from 1.1 and 1.2 we have

$$\epsilon_{Training} \leq \prod_{t=1}^{T} Z_t$$

$$= \exp\left(-2\sum_{t=1}^{T} \gamma_t^2\right)$$

each classifier is better than random

$\Rightarrow \gamma_t \geq \gamma$ for $\gamma > 0 \ \forall t$

$$\Rightarrow \epsilon_{Training} \leq \exp\left(-2\sum_{t=1}^{T} \gamma^2\right)$$

$$\Rightarrow \boxed{\epsilon_{Training} \leq \exp(-2T\gamma^2)}$$

This is because

$$-2T\gamma^2 \leq -\sum_{t=1}^{T} \gamma_t^2 \times 2$$

$$\because \gamma_t \geq \gamma$$

Q2.5.1)

For k = 2
Sum of squares = 5.3648 * 10^8
p1 = 79.8157
p2 = 54.8055
p3 = 67.3106

For k = 4
Sum of squares = 4.6111 * 10^8
p1 = 67.8812
p2 = 86.8329
p3 = 77.3571

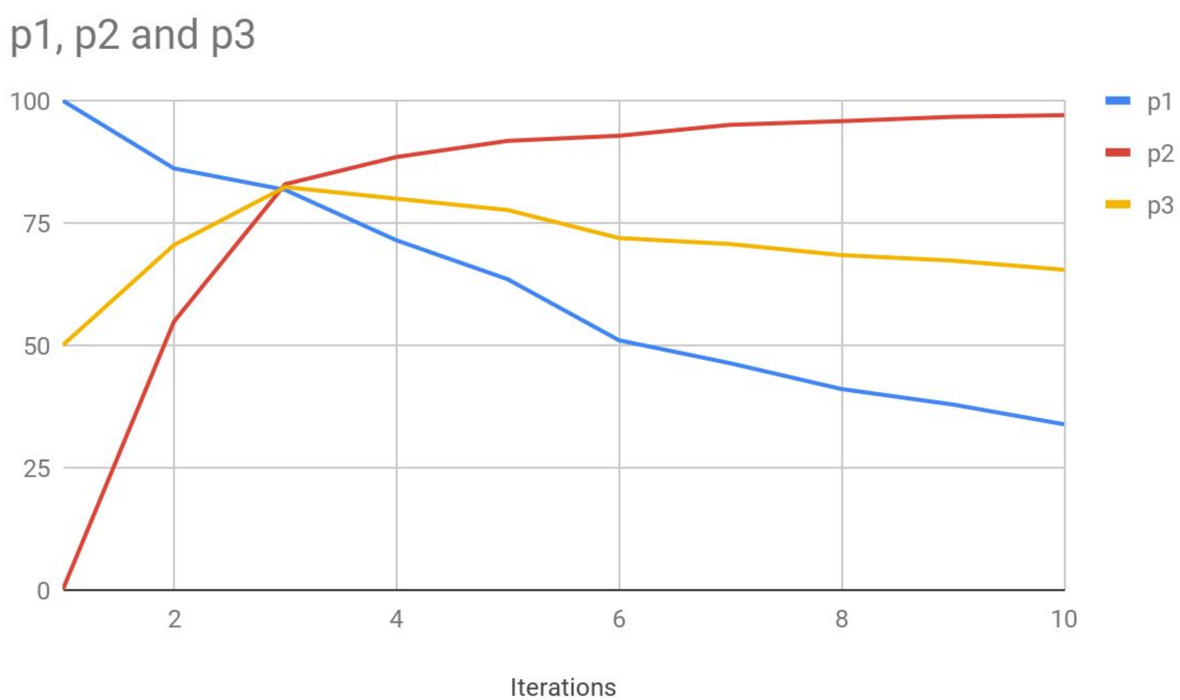
For k = 6
Sum of squares = 4.3135 * 10^8
p1 = 55.1765
p2 = 94.435
p3 = 74.8058

Q2.5.2) 8 iterations for K = 6

Q2.5.3 and 2.5.4)

K	1	2	3	4	5	6	7	8	9	10
SS (10^8)	5.961	5.382	4.832	4.640	4.490	4.288	4.138	4.031	3.945	3.853
p1	100	86.14	81.75	71.45	63.49	51.02	46.33	41.06	37.90	33.86
p2	0	54.82	82.91	88.46	91.76	92.80	95.04	95.79	96.67	97.01
p3	50	70.48	82.33	79.95	77.63	71.91	70.68	68.42	67.29	65.43


SS (*10^8) vs. Iterations
SS (*10^8)
6
5.5
5
4.5
4
3.5
2
4
6
8
10
Iterations


p1, p2 and p3
p1
p2
p3
100
75
50
25
0
2
4
6
8
10
Iterations

Q3.4.2) 5-fold cross-validation accuracy = 15.6443%

Q3.4.3) Tuned the values of C and gamma for 12 different values each i.e. 144 iterations and found out the best value of C to be 2^11 and gamma to be 2.
For C = 2048 and gamma = 2, 5-fold cross-validation accuracy = 87.90%

Value of C is varied from 2^-5 to 2^17 with new C = 4*old_C
Value of gamma is varied from 2^-15 to 2^7 with new gamma = 4*gamma

Attached models.mat which contains the accuracies at other C and gamma.
Each row corresponds to same C value and varying gamma.

Q3.4.5) Tuned the values of C and gamma for 12 different values each i.e. 144 iterations and found out the best value of C to be 32 and gamma to be 2.
For C = 32 and gamma = 2, 5-fold cross-validation accuracy = 93.75%

Value of C is varied from 2^-5 to 2^17 with new C = 4*old_C
Value of gamma is varied from 2^-15 to 2^7 with new gamma = 4*gamma

Attached models_2.mat which contains the accuracies at other C and gamma.
Each row corresponds to same C value and varying gamma.

Q3.4.6) Kaggle accuracy 82.5%