

Q-1(a) Dataset A converges at 180,372

Dataset A does not converge.

Q-1(b) Gradients of dataset B is very large and hence need smaller value of learning rate in order to make sure the change in weight remain within the dataset.

Graph of distribution of dataset A & B are plotted herewith in next page

Q-1(c) i) using a different learning rate i.e. smaller value of learning rate for B will be useful for dataset B to converge

(ii) Yes, decreasing the learning rate over time will be helpful for dataset B to converge

(iii) Linear scaling of dataset B would not help in it to converge.

(iv) Yes, adding regularized  $\|w\|_2^2$  to loss function will help penalize the gradient and thus will help dataset B to converge

(v) Adding zero mean gaussian noise to the training set would not help data labels

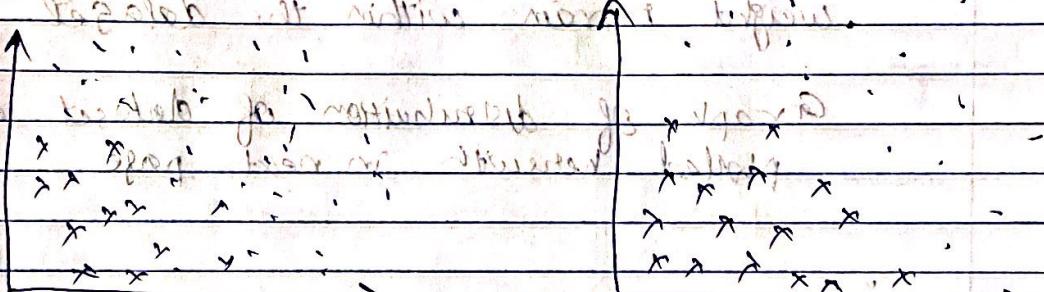
Is not a good  
Espresso  
category

Page 83  
SVM

Q-1(d) Yes F.E., SVMs are vulnerable to dataset like B.

Because if the classes are located reasonably far away then the SVM would not be effective in optimizing the max dist b/w ~~and~~ <sup>best possible</sup> decision boundary and classes.

Q-1(e)



Also note that even though the margin is small, the classes overlap significantly.

Max width =  $2 \times \text{margin}$  with maximum overlap.

Margin =  $\frac{\text{width}}{2}$  with maximum overlap.

Margin =  $\frac{\text{width}}{2}$  with maximum overlap.

Width =  $2 \times \text{margin}$  with maximum overlap.

Q-2(a)

Size of dictionary : 1758.

(b) Naïve Bayes has an accuracy of 97.84%.

(c) Words Top 5 indicative words for Naïve Bayes are:

'claim', 'won', 'prize', 'tire', 'urgent'.

d). Best kernel radius is 0.1.

V.P = positive related token

N.V.P = positive baseline neg token

unlabel - neutral tokens

discrepancy in baseline tag

Q-3 (c) Using Bayes rule to compute posterior distribution of  $\theta$ :

$$p(\theta|y, x) = \frac{p(\theta, x, y)}{p(y, x)}$$

$$= \frac{p(y|x; \theta) \cdot p(\theta|x)}{p(y|x) \cdot p(x)}$$

$$= \frac{p(y|x, \theta) \cdot p(\theta|x) \cdot p(x)}{p(y|x) \cdot p(x)}$$

$$= \frac{p(y|x, \theta) \cdot p(\theta|x)}{p(y|x)}$$

The mode of the posterior distribution is given by

$$\theta_{MAP} = \operatorname{argmax}_{\theta} p(\theta|x) = \operatorname{argmax}_{\theta} \frac{p(y|x, \theta) \cdot p(\theta|x)}{p(y|x)}$$

Noting that  $p(\theta|x) = p(\theta)$  and that the denominator is independent of  $\theta$  gives:

$$\theta_{MAP} = \operatorname{argmax}_{\theta} p(y|x, \theta) \cdot p(\theta)$$

Q. 3(b) Using result from 3(a)

$$\theta_{\text{MAP}} = \underset{\theta}{\operatorname{argmax}} p(y|x, \theta) p(\theta)$$

$$= \underset{\theta}{\operatorname{argmax}} \log(p(y|x, \theta) p(\theta))$$

$$= \underset{\theta}{\operatorname{argmin}} -\log(p(y|x, \theta) p(\theta))$$

$$= \underset{\theta}{\operatorname{argmin}} (-\log p(y|x, \theta) - \log p(\theta))$$

Since  $\theta \sim N(0, \eta^2 I)$ , the prior distribution is given by

$$p(\theta) = (2\pi)^{-n/2} \exp\left(-\frac{1}{2}\theta^T \Sigma^{-1} \theta\right)$$

where  $\Sigma = \eta^2 I \Rightarrow |\Sigma| = \eta^n$  and  $\Sigma^{-1} = \frac{1}{\eta^2} I$ .

$$-\log p(\theta) = \log(2\pi\eta) + \frac{1}{2\eta^2} \theta^T \theta = \log(2\pi\eta) + \frac{1}{2\eta^2} \|\theta\|_2^2$$

Comparing the above result with MLE estimation with L2 regularization gives

$$\lambda = \frac{1}{2\eta^2}$$

Q-3(c) Given that the random noise is independent of every training example  $(x^i, y^i)$  and is distributed according to a Gaussian distribution i.e.  $\epsilon^i \sim \mathcal{N}(0, \sigma^2)$

$$p(y^i | x^i, \theta) = p(\epsilon^i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\epsilon^i}{\sigma^2}\right) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^i - \theta^T x^i)^2}{2\sigma^2}\right)$$

The loss function for one training example for MAP estimation will then be given as:

$$J_{MAP}^i = -\log p(y^i | x^i, \theta) + \frac{1}{2\sigma^2} \|\theta\|_2^2 = \log(\sqrt{2\pi}\sigma) + \frac{(y^i - \theta^T x^i)^2}{2\sigma^2} + \frac{1}{2\sigma^2} \|\theta\|_2^2$$

Given the design input matrix  $x$  and output column vector  $\vec{y}$

$$J_{MAP} = \log(\sqrt{2\pi}\sigma) + \frac{1}{2\sigma^2} (\vec{y} - x\theta)^T (\vec{y} - x\theta) + \frac{1}{2\sigma^2} \theta^T \theta$$

To get  $\theta_{MAP} = \underset{\theta}{\operatorname{argmin}} J_{MAP}$ ,  $\nabla_{\theta} J_{MAP} = 0$

$$\Rightarrow \nabla_{\theta} \left( \frac{1}{2\sigma^2} (\vec{y} - x\theta)^T (\vec{y} - x\theta) + \frac{1}{2\sigma^2} \theta^T \theta \right) = 0$$

$$\Rightarrow -\frac{1}{\sigma^2} (x^T x \theta - x^T \vec{y}) + \frac{1}{\sigma^2} \theta = 0$$

$$\Rightarrow \left( \frac{1}{\sigma^2} x^T x - \frac{1}{\sigma^2} I_k \right) \theta = \frac{1}{\sigma^2} x^T \vec{y}$$

$$\Rightarrow \theta = \left( \frac{1}{\sigma^2} x^T x - \frac{1}{\sigma^2} I_k \right)^{-1} \left( \frac{1}{\sigma^2} x^T \vec{y} \right)$$

where  $I_k$  is an identity matrix of dimension  $(k \times k)$  and  $k$  is the number of features.

Q-3(d)

Using the result from previous parts, we have

$$J_{MAP} = (-\log p(y|x, \theta) + \log p(\theta))$$

Since  $\theta_i$  is independently distributed  $\text{as } L(0, b)$ :

$$p(\theta_i) = \frac{1}{2b} \exp\left(-\frac{|\theta_i|}{b}\right)$$

$$\Rightarrow -\log p(\theta_i) = \log(2b) + \frac{|\theta_i|}{b}$$

$$\Rightarrow J_{MAP} = \log(\sqrt{2\pi}) + \frac{1}{2} \frac{(\vec{y} - \vec{x}\theta)^T (\vec{y} - \vec{x}\theta)}{\theta^T \theta}$$

Comparing this with the L1 regularization of MLE:

$$J(\theta) = \|\vec{x}\theta - \vec{y}\|_2^2 + \gamma \|\theta\|_1 \text{ gives } \gamma = \frac{1}{b}$$

$$\therefore J(\theta) = \|\vec{x}\theta - \vec{y}\|_2^2 + \frac{1}{b} \|\theta\|_1$$

$$\text{Let } \lambda = \frac{1}{b} \text{ then } J(\theta) = \|\vec{x}\theta - \vec{y}\|_2^2 + \lambda \|\theta\|_1$$

$$\text{Let } \lambda = \frac{1}{b} \text{ then } J(\theta) = \|\vec{x}\theta - \vec{y}\|_2^2 + \lambda \|\theta\|_1$$

$$(x^T x + \lambda^2 I) \theta = x^T y - \lambda \theta$$

Q-4(c) For this section first iteration, taking the first example

$$\theta^0 = 0 \text{ result in:}$$

$$\theta' = \theta^0 + \alpha(y' - g(\theta^0 \phi')) \phi' = \theta' = \alpha(y' - g(\theta^{0T} \phi')) \phi'$$

where  $\phi'$  is the high dimensional feature mapping of  $x^i$ .

Next,

$$\theta^2 = \theta^1 + \alpha(y^2 - g(\theta^1 \phi^2)) \phi^2 = \alpha y^1 - g(\theta^{0T} \phi^1) \phi^1 + \alpha(y^2 - g(\theta^{1T} \phi^2)) \phi^2 + \alpha(y^3 - g(\theta^{2T} \phi^3)) \phi^3$$

$$\theta^2 = \theta^1 + \alpha(y^2 - g(\theta^1 \phi^2)) \phi^2 = \alpha y^1 - g(\theta^{0T} \phi^1) \phi^1 + \alpha(y^2 - g(\theta^{1T} \phi^2)) \phi^2$$

$$\theta^3 = \theta^2 + \alpha(y^3 - g(\theta^2 \phi^3)) \phi^3$$

$$= \alpha(y^1 - g(\theta^{0T} \phi^1)) \phi^1 + \alpha(y^2 - g(\theta^{1T} \phi^2)) \phi^2 + \alpha(y^3 - g(\theta^{2T} \phi^3)) \phi^3$$

Let  $B_j = \alpha(y^{j-1} \phi^j)$  be a scalar value that represent the coefficient of  $\phi^j$ , then  $\theta$  can be written as:

$$\theta^i = \sum_{j=1}^i B_j \phi^j$$

Q. 4(a)  
Contd

when  $\beta_i$  and  $\phi$  have the same number of rows which is the total no of training sample used.

The prediction on the new training sample  $x^{i+1}$  is given as:

$$h_{\phi,i}(x^{i+1}) = g(\phi^{i+1} \cdot \phi^{i+1}) = g\left(\sum_{j=1}^l \beta_j \phi_j \cdot \phi^{i+1}\right) = g\left(\sum_{j=1}^l \beta_j \langle \phi_j, \phi^{i+1} \rangle\right)$$

$$h_{\phi,i}(x^{i+1}) = g\left(\sum_{j=1}^l \beta_j k(\phi_j, \phi^{i+1})\right)$$

where  $k(x, z)$  is the kernel function.

The model is updated by adding a new element in  $\beta$  given as:

$$\beta_{i+1} = \alpha y^{i+1} - h_{\phi,i}(x^{i+1}).$$

Translates into value of  $\beta$  in  $(\beta_1, \beta_2, \dots, \beta_l)$  in next iteration is not affected up to machine error.

$$\text{Q-5(a)} \quad \text{CE}(y^i, \hat{y}^i) = -\sum_{k=1}^K y_k^i \log\left(\frac{e^{z_k^i}}{\sum_{k=1}^K e^{z_k^i}}\right) = -\sum_{k=1}^K y_k^i (z_k^i - \log \sum_{k=1}^K e^{z_k^i})$$

$$\frac{\partial \text{CE}(y^i, \hat{y}^i)}{\partial z_j^i} = -\sum_{k=1}^K y_k^i \left( \frac{\partial z_k^i}{\partial z_j^i} - \frac{1}{\sum_{k=1}^K e^{z_k^i}} \frac{\partial \log \sum_{k=1}^K e^{z_k^i}}{\partial z_j^i} \right)$$

$$= -\sum_{k=1}^K y_k^i \left( \frac{\partial z_k^i}{\partial z_j^i} - \frac{1}{\sum_{k=1}^K e^{z_k^i}} \cdot \frac{\partial \sum_{k=1}^K e^{z_k^i}}{\partial z_j^i} \right)$$

$$= -y_j^i + \sum_{k=1}^K y_k^i \frac{\Delta e^{z_j^i}}{\sum_{k=1}^K e^{z_k^i}} = -y_j^i + \frac{e^{z_j^i}}{\sum_{k=1}^K e^{z_k^i}} \sum_{k=1}^K y_k^i$$

$$= -y_j^i + \frac{e^{z_j^i}}{\sum_{k=1}^K e^{z_k^i}} \quad (II) \quad - y_j^i - y_j^i$$

$$\Rightarrow \nabla_z \text{CE}(y^i, \hat{y}^i) = \hat{y}^i - y^i$$

(b) Model baseline accuracy = 9.8%.  
Model regularized accuracy = 9.8%.

Plot attached - baseline

(c) Plot attached - regularized.