Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

QUESTION
1

Assignment Number: 1
Student Name: Nikhil Mittal
Roll Number: 17111056
Date: September 10, 2017

Lerning with prototypes:
A two class problem where the prototypes are the points (1, 0) (green) and (0, 1) (red). Let the green point be $\mu_+(1,0)$ and red point be $\mu_-(0,1)$. The decision boundary is the hyperplane where any point lying on the hyperplane is at equal distance from both the prototypes.
1. $d\left(z^1, z^2\right) = \left\langle z^1 - z^2, U\left(z^1 - z^2\right)\right\rangle, U = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$

Let's take a point z $\in \mathbb{R}^2$, finding it's distance from both the prototypes :
So, z = $\begin{bmatrix} x \\ y \end{bmatrix}$, $\mu_+ = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\mu_- = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$d\left(\mu_+, z\right) = \left\langle \mu_+ - z, U\left(\mu_+ - z\right)\right\rangle$

$d\left(\mu_+, z\right) = \left\langle \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix}\right)\right\rangle$
$d\left(\mu_+, z\right) = \left\langle \begin{bmatrix} 1-x \\ -y \end{bmatrix}, \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}\left(\begin{bmatrix} 1-x \\ -y \end{bmatrix}\right)\right\rangle$
$d\left(\mu_+, z\right) = \left\langle \begin{bmatrix} 1-x \\ -y \end{bmatrix}, \begin{bmatrix} 3*(1-x) \\ -y \end{bmatrix}\right\rangle$
$d\left(\mu_+, z\right) = \begin{bmatrix} 1-x \\ -y \end{bmatrix}^T . \begin{bmatrix} 3*(1-x) \\ -y \end{bmatrix}$
$d\left(\mu_+, z\right) = 3(1-x)^2 + y^2$

$d\left(\mu_-, z\right) = \left\langle \mu_- - z, U\left(\mu_- - z\right)\right\rangle$

$d\left(\mu_-, z\right) = \left\langle \begin{bmatrix} 0 \\ 1 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}\left(\begin{bmatrix} 0 \\ 1 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix}\right)\right\rangle$
$d\left(\mu_-, z\right) = \left\langle \begin{bmatrix} -x \\ 1-y \end{bmatrix}, \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}\left(\begin{bmatrix} -x \\ 1-y \end{bmatrix}\right)\right\rangle$
$d\left(\mu_-, z\right) = \left\langle \begin{bmatrix} -x \\ 1-y \end{bmatrix}, \begin{bmatrix} -3x \\ 1-y \end{bmatrix}\right\rangle$
$d\left(\mu_-, z\right) = \begin{bmatrix} -x \\ 1-y \end{bmatrix}^T . \begin{bmatrix} -3x \\ 1-y \end{bmatrix}$
$d\left(\mu_-, z\right) = 3x^2 + (1-y)^2$
Equate the difference of distances to decision boundary,
$f(z) = d\left(\mu_+, z\right) - d\left(\mu_-, z\right)$
For equation of f(z), $f(z) = 0$
$Then, d\left(\mu_+, z\right) - d\left(\mu_-, z\right) = 0$
$3(1-x)^2 + y^2 - 3x^2 - (1-y)^2 = 0$
$3(1 + x^2 - 2x) + y^2 - 3x^2 - (1 + y^2 - 2y) = 0$ $3 + 3x^2 - 6x + y^2 - 3x^2 - 1 - y^2 - 2y = 0$
$-6x + 2y + 2 = 0$

$y = 3x - 1$

The mathematical expression for the decision boundary is :

$$\boxed{y = 3x - 1}$$

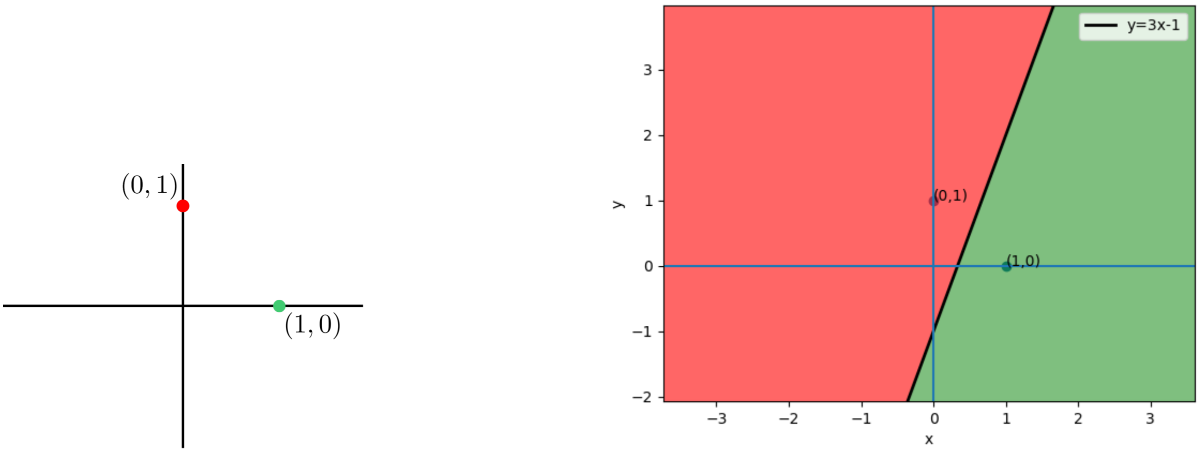

Figure 1: Learning with Prototypes: the figure on the left shows the two prototypes. The figure on the right shows what the decision boundary if the distance measure used is $d(\mathbf{z}^1, \mathbf{z}^2) = \langle \mathbf{z}^1 - \mathbf{z}^2, V(\mathbf{z}^1 - \mathbf{z}^2) \rangle$, for any two points $\mathbf{z}^1, \mathbf{z}^2 \in \mathbb{R}^2$. The decision boundary in this case is the line $y = 3x - 1$.

2. $d\left(z^1, z^2\right) = \left\langle z^1 - z^2, V\left(z^1 - z^2\right)\right\rangle, V = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$

Let's take a point z $\in \mathbb{R}^2$, finding it's distance from both the prototypes :

So, z = $\begin{bmatrix} x \\ y \end{bmatrix}$, $\mu_+ = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\mu_- = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$d\left(\mu_+, z\right) = \left\langle \mu_+ - z, V\left(\mu_+ - z\right)\right\rangle$

$$d\left(\mu_+, z\right) = \left\langle \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} \right) \right\rangle$$

$$d\left(\mu_+, z\right) = \left\langle \begin{bmatrix} 1-x \\ -y \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1-x \\ -y \end{bmatrix} \right) \right\rangle$$

$$d\left(\mu_+, z\right) = \left\langle \begin{bmatrix} 1-x \\ -y \end{bmatrix}, \begin{bmatrix} 1-x \\ 0 \end{bmatrix} \right\rangle$$

$$d\left(\mu_+, z\right) = \begin{bmatrix} 1-x \\ -y \end{bmatrix}^T . \begin{bmatrix} 1-x \\ -y \end{bmatrix}$$

$d\left(\mu_+, z\right) = (1-x)^2$

$d\left(\mu_-, z\right) = \left\langle \mu_- - z, V\left(\mu_- - z\right)\right\rangle$

$$d\left(\mu_-, z\right) = \left\langle \begin{bmatrix} 0 \\ 1 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} \right) \right\rangle$$

$$d\left(\mu_-, z\right) = \left\langle \begin{bmatrix} -x \\ 1-y \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \left(\begin{bmatrix} -x \\ 1-y \end{bmatrix} \right) \right\rangle$$

$$d\left(\mu_-, z\right) = \left\langle \begin{bmatrix} -x \\ 1-y \end{bmatrix}, \begin{bmatrix} -x \\ 0 \end{bmatrix} \right\rangle$$

$$d\left(\mu_-, z\right) = \begin{bmatrix} -x \\ 1-y \end{bmatrix}^T . \begin{bmatrix} -x \\ 0 \end{bmatrix}$$

$d\left(\mu_-, z\right) = x^2$

Equate the difference of distances to decision boundary,

$f(z) = d\left(\mu_+, z\right) - d\left(\mu_-, z\right)$

For equation of f(z), $f(z) = 0$

$Then, d\left(\mu_+, z\right) - d\left(\mu_-, z\right) = 0$

$(1-x)^2 - x^2 = 0$

$1 + x^2 - 2x - x^2 = 0$

$1 - 2x = 0$

$x = \frac{1}{2}$

The mathematical expression for the decision boundary is :

$$\boxed{x = \frac{1}{2}}$$

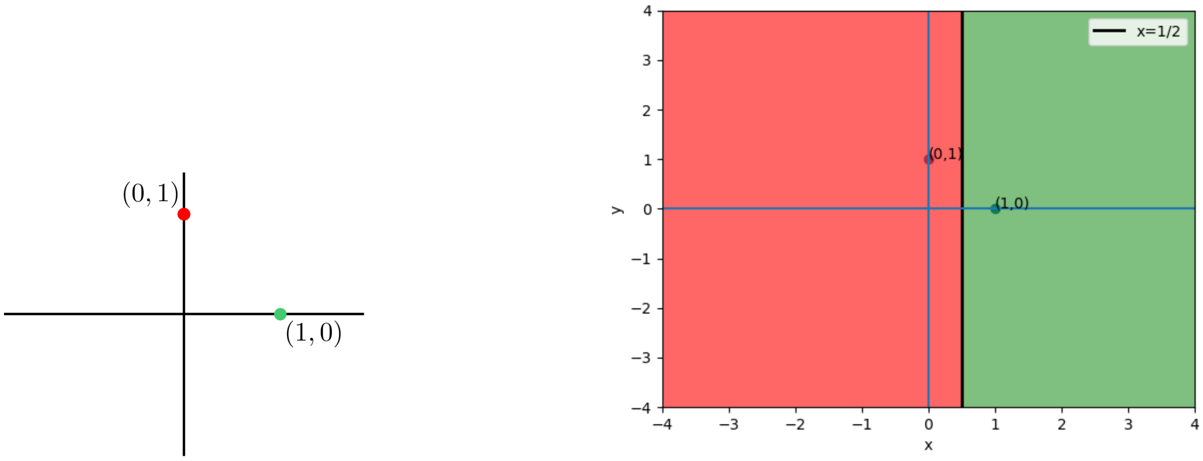

Figure 2: Learning with Protypes: the figure on the left shows the two prototypes. The figure on the right shows what the decision boundary if the distance measure used is $d(\mathbf{z}^1, \mathbf{z}^2) = \big\langle \mathbf{z}^1 - \mathbf{z}^2, U(\mathbf{z}^1 - \mathbf{z}^2)\big\rangle$, for any two points $\mathbf{z}^1, \mathbf{z}^2 \in \mathbb{R}^2$. The decision boundary in this case is the line $x = 1/2$.

Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

QUESTION 2

Assignment Number: 1
Student Name: Nikhil Mittal
Roll Number: 17111056
Date: September 10, 2017

Likelihood distribution :

$$P[y^i|x^i,\mathbf{w}] = \mathcal{N}\left(\left\langle \mathbf{w}, x^i\right\rangle, \sigma^2\right) \ = \frac{1}{\sigma\sqrt{2\pi}} e^{-\left(y^i-\left\langle w,x^i\right\rangle\right)^2 \big/ 2\sigma^2}$$

Prior distribution :

Using Beta-prior, the beta distribution is defined over the interval [0, 1] this distribution can be scaled to an interval [p, q].

We have a constraint,

$$\|\mathbf{w}\| \leq r$$

So the $\|\mathbf{w}\|$ can take values from 0 to r. For scaling the beta distribution from $[0,1]$ to $[0,r]$.
$w = x * (r-0) + 0$

$w = rx$

$$\begin{aligned} \beta'(\alpha,\beta) &= \frac{(w-0)^{\alpha-1}\,.\,(r-w)^{\beta-1}}{(r-0)^{\alpha+\beta-1}\,.\,\mathrm{B}(\alpha,\ \beta)} \\ &= \frac{(w)^{\alpha-1}\,.\,(r-w)^{\beta-1}}{(r)^{\alpha+\beta-1}\,.\,\mathrm{B}(\alpha,\ \beta)} \end{aligned}$$

So Taking a prior of the following form :

$$\mathbb{P}[\mathbf{w}] = \begin{cases} \frac{(\mathbf{w})^{\alpha-1}.(r-\mathbf{w})^{\beta-1}}{(r)^{\alpha+\beta-1}.\,\mathrm{B}(\alpha,\,\beta)} & ,\ 0 \leq \mathbf{w} \leq r \\ 0 & ,\ otherwise \end{cases}$$

Map Estimate is proportional to the product of likelihood and prior. So,

$$\log P[\mathbf{w} \mid \mathbf{X},\mathbf{y}] = \log P[\mathbf{y}|\mathbf{X},\mathbf{w}] + \log P[\mathbf{w}]$$

There's another distribution called Truncated normal distribution, it is a probability density function **f**, for , is given by

$$f(x;\mu,\sigma,a,b) = \frac{\phi\left(\frac{x-\mu}{\sigma}\right)}{\sigma\left(\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)\right)}$$

and by $\mathbf{f = 0}$ **otherwise**.

Here,

$$\phi\left(\xi\right)=\frac{1}{2\pi}\exp\left(-\frac{1}{2}\xi^{2}\right)$$

is the probability density function of the standard normal distribution and $\Phi\left(.\right)$ is its cumulative distribution function

$$\Phi\left(x\right)=\frac{1}{2}\left(1+erf\left(\frac{x}{\sqrt{2}}\right)\right)$$

This can also be used as a prior. So

$$Put\ a=0,b=r.$$

$$f\left(w;\mu,\sigma,0,r\right)=\frac{\phi\left(\frac{w-\mu}{\sigma}\right)}{\sigma\left(\Phi\left(\frac{r-\mu}{\sigma}\right)-\Phi\left(\frac{-\mu}{\sigma}\right)\right)}$$

So this is P[**w**] .

Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

Assignment Number: 1
Student Name: Nikhil Mittal
Roll Number: 17111056
Date: September 10, 2017

QUESTION 3

We need to design a likelihood and prior distribution such that $\hat{\mathbf{w}}_{\text{fr}}$ this is the map estimate. The Posterior distribution on $\mathbf{w}$ can be seen as proportional to the product of the likelihood and the prior distributions.

Let's take the gaussian likelihood and gaussian prior,
Here $\mathbf{X}$ is $\mathbf{n}$*$\mathbf{d}$ matrix where $\mathbf{n}$ is the no. of data points and $\mathbf{d}$ is the no. of features.

$$P[y^i|x^i, w] = \mathcal{N}\left(\left\langle w, x^i \right\rangle, \sigma^2\right) \; = \frac{1}{\sigma\sqrt{2\pi}} e^{-\left(y^i - \left\langle w, x^i \right\rangle\right)^2 \big/ 2\sigma^2}$$

$$\mathrm{P}[\mathbf{y}|\mathbf{X}, \mathbf{w}] = \frac{1}{\sigma\sqrt{2\pi}} \prod_{i=1}^{n} e^{-\left(y^i - \left\langle w, x^i \right\rangle\right)^2 \big/ 2\sigma^2}$$

So Log likelihood will be :

$\log P[\mathbf{y}|\mathbf{X}, \mathbf{w}] = C - \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y^i - \left\langle \mathbf{w}, \mathbf{x^i} \right\rangle\right)^2$

Taking Gaussian prior, with a minor modification :

Consider a β Such that

$$\beta = \begin{bmatrix} \frac{1}{\beta_1} & \frac{1}{\beta_2} & \cdot \cdot\cdot & \frac{1}{\beta_d} \end{bmatrix}, \beta_j > 0$$

and d is the dimension of β. Then,

$$\mathrm{I}' = \beta \cdot \mathbf{I}$$

Where $\mathbf{I}$ is the identity matrix.

$$\mathrm{I}' = \begin{bmatrix} \frac{1}{\beta_1} & 0 & . & . & 0 \\ 0 & \frac{1}{\beta_2} & . & . & 0 \\ 0 & . & . & . & 0 \\ 0 & . & . & . & 0 \\ 0 & . & . & . & \frac{1}{\beta_d} \end{bmatrix}$$

Using Gaussian prior,

$$P[\mathbf{w}] = \mathcal{N}\left(0, \rho^2 . \mathrm{I}'\right) = \frac{1}{\sqrt{(2\pi)^d \rho^2 \left|\mathrm{I}'\right|}} \exp\left(-\frac{\mathbf{w}^T \mathrm{I}'^{-1} \mathbf{w}}{2\rho^2}\right)$$

Actually I' is diagonal matrix. So, It's inverse is :

$$\mathrm{I}'^{-1} = \begin{bmatrix} \beta_1 & 0 & . & . & 0 \\ 0 & \beta_2 & . & . & 0 \\ 0 & . & . & . & 0 \\ 0 & . & . & . & 0 \\ 0 & . & . & . & \beta_d \end{bmatrix}$$

So,

$$\mathbf{w}^T \mathrm{I}'^{-1} \mathbf{w} = \sum_{i=1}^{d} \beta_j \left(\mathbf{w}_j\right)^2$$

So,

$$P[\mathbf{w}] = \frac{1}{\sqrt{(2\pi)^d \rho^2 \left|\mathrm{I}'\right|}} \exp\left(-\frac{\sum_{j=1}^{d} \beta_j \left(\mathbf{w}_j\right)^2}{2\rho^2}\right)$$

Taking log we get :

$\log \mathrm{P}[\mathbf{w}] = C' - \frac{1}{2\rho^2} \sum_{j=1}^{d} \beta_j \left(\mathbf{w}_j\right)^2$

So the posterior distribution on $\mathbf{w}$:

$\mathrm{P}\ [\mathbf{w} \mid \mathbf{X}, \mathbf{y}] \propto \mathrm{P}[\mathbf{y}|\mathbf{X}, \mathbf{w}].\ \mathrm{P}[\mathbf{w}]$

$\log P[\mathbf{w} \mid \mathbf{X}, \mathbf{y}] = \log P[\mathbf{y}|\mathbf{X}, \mathbf{w}] + \log P[\mathbf{w}]$

$\widehat{\mathbf{w}}_{\mathbf{MAP}} = \arg\max_{\mathbf{w}} \log P[\mathbf{y}|\mathbf{X}, \mathbf{w}] + \log P[\mathbf{w}]$

Using the expressions of log-likelihood and log-prior, we can say that :

$$\widehat{\mathbf{w}}_{\mathbf{MAP}} = \arg\max_{\mathbf{w}} C - \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y^i - \left\langle \mathbf{w}, \mathbf{x}^i \right\rangle\right)^2 + C' - \frac{1}{2\rho^2} \sum_{j=1}^{d} \beta_j \left(\mathbf{w}_j\right)^2$$

Ignoring constants

$$\widehat{\mathbf{w}}_{\mathbf{MAP}} = \arg\min_{\mathbf{w}} \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y^i - \left\langle \mathbf{w}, \mathbf{x}^i \right\rangle\right)^2 + \frac{1}{2\rho^2} \sum_{j=1}^{d} \beta_j \left(\mathbf{w}_j\right)^2$$

Multiplying the equation by $2\sigma^2$, we get Equation 1:

$$\widehat{\mathbf{w}}_{\mathbf{MAP}} = \arg\min_{\mathbf{w}} \sum_{i=1}^{n} \left(y^i - \left\langle \mathbf{w}, \mathbf{x}^i \right\rangle\right)^2 + \frac{\sigma^2}{\rho^2} \sum_{j=1}^{d} \beta_j \left(\mathbf{w}_j\right)^2$$

Let's take α_j, such that,

$$\alpha_j = \frac{\sigma^2}{\rho^2}.\beta_j$$

So,

$$\alpha = [\alpha_1 \; \alpha_2 \; ... \; \alpha_d] \;\; , \alpha_j > 0$$

and d is the dimension. So Equation 1 changes to :

$$\widehat{\mathbf{w}}_{\mathbf{MAP}} = \arg\min_{\mathbf{w}} \; \sum_{i=1}^{n} \left(y^i - \left\langle \mathbf{w}, \mathbf{x}^i \right\rangle\right)^2 + \sum_{j=1}^{d} \alpha_j \left(\mathbf{w}_j\right)^2$$

This is similar to the $\widehat{\mathbf{w}}_{fr}$ as given in the question.

So the likelihood distribution is

$$\mathrm{P}[\mathbf{y}|\mathbf{X}, \mathbf{w}] = \frac{1}{\sigma\sqrt{2\pi}} \prod_{i=1}^{n} e^{-\left(y^i - \left\langle w, x^i \right\rangle\right)^2 \big/ 2\sigma^2}$$

And the prior distribution is Gaussian prior

$$P[\mathbf{w}] = \mathcal{N}\left(0, \rho^2 . \mathrm{I}'\right) = \frac{1}{\sqrt{(2\pi)^d \, \rho^2 \, |\mathrm{I}'|}} \exp\left(-\frac{\mathbf{w}^T \mathrm{I}'^{-1} \mathbf{w}}{2\rho^2}\right)$$

Which is equivalent to this

$$P[\mathbf{w}] = \frac{1}{\sqrt{(2\pi)^d \, \rho^2 \, |\mathrm{I}'|}} \exp\left(-\frac{\sum_{j=1}^{d} \beta_j \left(\mathbf{w}_j\right)^2}{2\rho^2}\right)$$

Hence these are explicit forms for the distributions.

$\widehat{\mathbf{w}}_{fr}$ has a closed form solution which can be derived as follows :

Take derivative of $\widehat{\mathbf{w}}_{fr}$ w.r.t $\mathbf{w}$

$$\frac{d\widehat{\mathbf{w}}_{\mathbf{fr}}}{dx} = (2)\sum_{i=1}^{n}\left(y^i - \left\langle \mathbf{w}, \mathbf{x}^i \right\rangle\right)\left(-\mathbf{x}^i\right) + \frac{d}{dw}(\sum_{j=1}^{d}\alpha_j\left(\mathbf{w}_j\right)^2)$$

$$\frac{d\widehat{\mathbf{w}}_{\mathbf{fr}}}{dx} = (-2)\sum_{i=1}^{n}\left(\mathbf{x}^i y^i - \mathbf{x}^i\left\langle \mathbf{w}, \mathbf{x}^i \right\rangle\right) + \sum_{j=1}^{d}\frac{d}{d\mathbf{w}_j}(\alpha_j\left(\mathbf{w}_j\right)^2)$$

$$\frac{d\widehat{\mathbf{w}}_{\mathbf{fr}}}{dx} = (-2)\sum_{i=1}^{n}\left(\mathbf{x}^i y^i - \mathbf{w}.(\mathbf{x}^i)^2\right) + \sum_{j=1}^{d}(2\alpha_j\mathbf{w}_j)$$

$$\frac{d\widehat{\mathbf{w}}_{\mathbf{fr}}}{dx} = (-2)\sum_{i=1}^{n}\left(\mathbf{x}^i y^i\right) + 2\ \mathbf{w}.\sum_{i=1}^{n}(\mathbf{x}^i)^2 + 2\alpha^T\cdot\mathbf{w}$$

Equating this equation to zero to get the value of $\mathbf{w}$, We get :

$$\mathbf{w}.\sum_{i=1}^{n}(\mathbf{x}^i)^2 + \alpha^T\mathbf{w} = \sum_{i=1}^{n}\left(\mathbf{x}^i y^i\right)$$

$$\mathbf{w}\mathbf{X}^T\mathbf{X} + \alpha^T\mathbf{w} = \mathbf{X}^T\mathbf{Y}$$

$$\mathbf{w}\left(\mathbf{X}^T\mathbf{X} + \alpha^T I\right) = \mathbf{X}^T\mathbf{Y}$$

$$\mathbf{w} = \left(\mathbf{X}^T\mathbf{X} + \alpha^T I\right)^{-1}\mathbf{X}^T\mathbf{Y}$$

So this is the closed form expression for $\widehat{\mathbf{w}}_{fr}$:

$$\widehat{\mathbf{w}}_{fr} = \left(\mathbf{X}^T\mathbf{X} + \alpha^T I\right)^{-1}\mathbf{X}^T\mathbf{Y}$$

Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

QUESTION 4

Assignment Number: 1
Student Name: Nikhil Mittal
Roll Number: 17111056
Date: September 10, 2017

$$\left\{\widehat{\mathbf{W}}, \left\{\widehat{\xi_i}\right\}\right\} = \underset{\mathbf{W},\ \left\{\widehat{\xi_i}\right\}}{arg\ min} \sum_{k=1}^{K} \left\|\mathbf{w}^k\right\|_2^2 + \sum_{i=1}^{n} \xi_i$$

$$s.t. \left\langle \mathbf{w}^{y^i}, x^i \right\rangle \geq \left\langle \mathbf{w}^k, x^i \right\rangle + 1 - \xi_i \ , \forall i \ \forall k \neq y^i$$

$\xi_i \geq 0, \forall i$ So $\widehat{\xi_i}$ is for every data point.

The given constraint: $\left\langle \mathbf{w}^{y^i}, x^i \right\rangle \geq \left\langle \mathbf{w}^k, x^i \right\rangle + 1 - \xi_i \ , \forall i \ \forall k \neq y^i$

$$\xi_i \geq \left\langle \mathbf{w}^k, x^i \right\rangle + 1 - \left\langle \mathbf{w}^{y^i}, x^i \right\rangle \ , \forall i \ \forall k \neq y^i$$

$$\xi_i \geq 1 + \left\langle \mathbf{w}^k, x^i \right\rangle - \left\langle \mathbf{w}^{y^i}, x^i \right\rangle \ , \forall i \ \forall k \neq y^i$$

So we can safely say that : $\xi_i = \max\left(1 + \left\langle \mathbf{w}^k, x^i \right\rangle - \left\langle \mathbf{w}^{y^i}, x^i \right\rangle\right), \forall i, \ \forall k \neq y^i$

As,$\eta^i = \left\langle \mathbf{W}, \mathbf{x}^i \right\rangle \Rightarrow \eta_k^i = \left\langle \mathbf{w}^k, \mathbf{x}^i \right\rangle and \ \eta_y^i = \left\langle \mathbf{w}^{y^i}, \mathbf{x}^i \right\rangle$

$$\xi_i \geq 1 + \eta_k^i - \eta_y^i \quad \forall k \neq y^i$$

ξ_i is evaluated for all $\mathbf{w} \in \mathbf{W}$, except for k $\neq$ y. We will take the largest possible value for ξ_i. So ξ_i will be maximum only when η_k is maximum, satisfying the condition that $k \neq y$.

So ξ_i can be s.t.,

$$\xi_i \geq 1 + \max \eta_k^i - \eta_y^i \quad \forall k \neq y^i$$

But because of the objective we will not want to set it any larger than necessary. As setting larger than that value will be unnecessary in the objective.

$\xi_i = 1 + \max\limits_{k \neq y} \eta_k^i - \eta_y^i$ and $\xi_i \geq 0$

Therefore ξ_i is always a positive function and can be written as :

$$\xi_i = \left[1 + \max_{k \neq y} \eta_k^i - \eta_y^i\right]_+$$

The above is same as $l_{cs}\left(y^i, \eta^i\right)$. It is called as the crammer-singer loss function.

Hence, $l_{cs}\left(y^i, \eta^i\right) = \xi_i$

So the constrained formulation can now be written in an unconstrained formulation which is same as (P2). As ξ_i can be replaced by l_{cs}. Hence by this expression (P2) can be derived from (P1) and vice-versa.

Suppose $\left\{\mathbf{W}^0, \{\xi_i^0\}\right\}$ is the optimum for P1 equation.
So ξ_i satisfies the cosntraints.

For $\left\{\mathbf{W}^0, \{\xi_i^0\}\right\}$ the sum is minimum. Such that
$\xi_i^0 = \max\left(1 + \langle \mathbf{w}^k, x^i\rangle - \left\langle \mathbf{w}^{y^i}, x^i\right\rangle\right)$ *and* $\xi_i \geq 0$

So, $\xi_i^0 = 1 + \max\limits_{k\neq y} \eta_k^i - \eta_y^i$ *and* $\xi_i \geq 0$
Since $\xi_i^0 \geq 0$ so $1 + \max\limits_{k\neq y} \eta_k^i - \eta_y^i \geq 0$
As stated $l_{cs}\left(y^i, \eta^i\right)$.

Hence $l_{cs} = \xi_i^0$, So the problem (P2) after plugging the value of l_{cs} becomes similar to (P1).

(P2) is :

$\left\{\widehat{\mathbf{W}}\right\} = arg\ \underset{\mathbf{W}}{min} \sum_{k=1}^{K} \left\|\mathbf{w}^k\right\|_2^2 + \sum_{i=1}^{n} l_{cs}\left(y^i, \eta^i\right)$

Then, Put value of l_{cs}. The problem becomes,

$\left\{\widehat{\mathbf{W}}\right\} = arg\ \underset{\mathbf{W}}{min} \sum_{k=1}^{K} \left\|\mathbf{w}^k\right\|_2^2 + \sum_{i=1}^{n} \xi_i^0$

As we know that above has minimum at $\left\{\mathbf{W}^0, \{\xi_i^0\}\right\}$
This problem is already solved as we know, $\mathbf{W}^0$ minimizes $\sum_{k=1}^{K} \left\|\mathbf{w}^k\right\|_2^2$ and the $\sum_{i=1}^{n} \xi_i^0$ is just a constant and is already the optimal value.

So $\mathbf{W}^0$ is also a solution for (P2).

Now, $\mathbf{W}^1$ is an optimum for (P2). This means :

$\left\{\widehat{\mathbf{W}}\right\} = arg\ \underset{\mathbf{W}}{min} \sum_{k=1}^{K} \left\|\mathbf{w}^k\right\|_2^2 + \sum_{i=1}^{n} l_{cs}\left(y^i, \eta^i\right)$ has minima at $\mathbf{W}^1$.

$\sum_{i=1}^{n} l_{cs}(y^i, \eta^i)$ is the sum of losses.

$l_{cs}(y^i, \eta^i) = \left[1 + \max\limits_{k\neq y} \eta_k^i - \eta_y^i\right]_+$
$\eta_k^i = \langle \mathbf{w}^k, \mathbf{x}^i\rangle$ *and* $\eta_y^i = \left\langle \mathbf{w}^{y^i}, \mathbf{x}^i\right\rangle$

$l_{cs}(y^i, \eta^i) \geq \left[1 + \langle \mathbf{w}^k, \mathbf{x}^i\rangle - \left\langle \mathbf{w}^{y^i}, \mathbf{x}^i\right\rangle\right]_+ \forall i;\ \forall k \neq y^i$. As $l_{cs}(y^i, \eta^i)$ is the maximum value so this inequality should be satisfied.

So we can say that there is some ξ_i^1 which is equivalent to $l_{cs}(y^i, \eta^i)$ and $\xi_i^1 \geq 0$ as $l_{cs}(y^i, \eta^i) \geq 0$.

So let,

$$\xi_i^1 \geq 1 + \left\langle \mathbf{w}^k, \mathbf{x}^i \right\rangle - \left\langle \mathbf{w}^{y^i}, \mathbf{x}^i \right\rangle \; \forall i; \; \forall k \neq y^i$$

Now, coming to (P1) as per given constraint:

$$\left\langle \mathbf{w}^{y^i}, x^i \right\rangle \geq \left\langle \mathbf{w}^k, x^i \right\rangle + 1 - \xi_i \;, \forall i \; \forall k \neq y^i$$

$$\xi_i \geq \left\langle \mathbf{w}^k, x^i \right\rangle + 1 - \left\langle \mathbf{w}^{y^i}, x^i \right\rangle \;, \forall i \; \forall k \neq y^i$$

$$\xi_i \geq \xi_i^1$$

Also,

$$\xi_i^1 \geq 0 \quad \Rightarrow \quad \xi_i \geq 0$$

So the $\sum_{i=1}^n \xi_i^1$ is minimum as it is same as $\sum_{i=1}^n l_{cs}$.

So If I plug in ξ_i in (P2) instead of l_{cs}. So it becomes constrained problem instead of unconstrained, but the meaning remains same.

So, $\sum_{k=1}^K \left\|\mathbf{w}^k\right\|_2^2 + \sum_{i=1}^n \xi_i$ has min at $\mathbf{W}^1$ and $\{\xi_i\}$ value as $\left\{\xi_i^1\right\}$, as it is optimum.

$\left\{\widehat{\mathbf{W}}, \left\{\widehat{\xi_i}\right\}\right\} = \underset{\mathbf{W}, \; \{\widehat{\xi_i}\}}{arg\ min} \sum_{k=1}^K \left\|\mathbf{w}^k\right\|_2^2 + \sum_{i=1}^n \xi_i$ has optimum at $\left\{\mathbf{W}^1, \{\xi_i^1\}\right\}$.

Therefore (P1) and (P2) are equivalent.

Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

Assignment Number: 1
Student Name: Nikhil Mittal
Roll Number: 17111056
Date: September 10, 2017

QUESTION 5

$f\left(\mathbf{w}\right) = \sum_{i=1}^{n}\left[1 - y^i < \mathbf{w}, x^i >\right]_+$

$= \sum_{i=1}^{n}\mathbf{v}\left(x^i, y^i\right)$, such that $\mathbf{v}\left(x^i, y^i\right) = \begin{cases} 1 - y^i < \mathbf{w}, x^i > & , y^i < \mathbf{w}, x^i > < 1 \\ 0 & , y^i < \mathbf{w}, x^i > \geq 1 \end{cases}$

$f\left(\mathbf{w}\right)$ is just the sum of hinge losses, where hinge function is not differentiable at $y^i < \mathbf{w}, x^i >= 1$. To find the sub-differential of f at w,

$$\frac{df\left(\mathbf{w}\right)}{d\mathbf{w}} = \sum \frac{dv\left(x^i, y^i\right)}{d\mathbf{w}}$$

$$\frac{dv\left(x^i, y^i\right)}{d\mathbf{w}} = \begin{cases} -y^i x^i & , y^i < \mathbf{w}, x^i > < 1 \\ 0 & , y^i < \mathbf{w}, x^i > \geq 1 \end{cases}$$

$$\bigtriangledown f = \frac{df\left(\mathbf{w}\right)}{d\mathbf{w}}$$

Since $\bigtriangledown f$ is the subgradient of f. Therefore it should satisfy

$$f\left(\mathbf{w}'\right) \geq f\left(\mathbf{w}\right) + \left\langle \bigtriangledown f, \mathbf{w} - \mathbf{w}' \right\rangle$$

As given in the question, g= $\sum_{i=1}^{n} h_i$ where

$$h_i = \begin{cases} -y^i x^i & , y^i \left\langle \mathbf{w}, x^i \right\rangle < 1 \\ 0 & , y^i \left\langle \mathbf{w}, x^i \right\rangle \geq 1 \end{cases}$$

So by the equations **g** appears to be same as $f\left(\mathbf{w}\right)$. This can be proved by showing that for every $\mathbf{w}' \in \mathbb{R}^d, f\left(\mathbf{w}'\right) \geq f\left(\mathbf{w}\right) + \langle g, \mathbf{w}' - \mathbf{w}\rangle$:

$$f\left(\mathbf{w}'\right) = \sum_{i=1}^{n}\left[1 - y^i < \mathbf{w}', x^i >\right]_+$$

$$f\left(\mathbf{w}\right)=\sum_{i=1}^{n}\left[1-y^{i}<\mathbf{w},x^{i}>\right]_{+}$$

$$f\left(\mathbf{w}'\right)-f\left(\mathbf{w}\right)=\sum_{i=1}^{n}\left[1-y^{i}<\mathbf{w}',x^{i}>\right]_{+}-\sum_{i=1}^{n}\left[1-y^{i}<\mathbf{w},x^{i}>\right]_{+}$$

$f\left(\mathbf{w}'\right)-f\left(\mathbf{w}\right)=\sum_{i=1}^{n}\left[1-y^{i}<\mathbf{w}',x^{i}>\right]_{+}-\left[1-y^{i}<\mathbf{w},x^{i}>\right]_{+}$

Let's say that $f\left(\mathbf{w}'\right)-f\left(\mathbf{w}\right)=\sum_{i=1}^{n}s^{i}$

So s^{i} will be :

$$s^{i}=\begin{cases}\left(1-y^{i}<\mathbf{w}',x^{i}>\right)-\left(1-y^{i}<\mathbf{w},x^{i}>\right) & ,y^{i}.<\mathbf{w}',x^{i}><1 \quad ,y^{i}.<\mathbf{w},x^{i}><1\\ -\left(1-y^{i}<\mathbf{w},x^{i}>\right) & ,y^{i}.<\mathbf{w}',x^{i}>\geq 1 \quad ,y^{i}.<\mathbf{w},x^{i}><1\\ \left(1-y^{i}<\mathbf{w}',x^{i}>\right) & ,y^{i}.<\mathbf{w}',x^{i}><1 \quad ,y^{i}.<\mathbf{w},x^{i}>\geq 1\\ 0 & ,y^{i}.<\mathbf{w}',x^{i}>\geq 1 \quad ,y^{i}.<\mathbf{w},x^{i}>\geq 1\end{cases}$$

The 1st part of s^{i} can be simplified as following :
$=\left(1-y^{i}<\mathbf{w}',x^{i}>\right)-\left(1-y^{i}<\mathbf{w},x^{i}>\right)$

$=y^{i}\left(<\mathbf{w}',x^{i}>-<\mathbf{w},x^{i}>\right)$

$=y^{i}\left(<\mathbf{w}-\mathbf{w}',x^{i}>\right)$

So s^{i} will be :

$$s^{i}=\begin{cases}y^{i}\left(<\mathbf{w}-\mathbf{w}',x^{i}>\right) & ,y^{i}.<\mathbf{w}',x^{i}><1 \quad ,y^{i}.<\mathbf{w},x^{i}><1\\ -\left(1-y^{i}<\mathbf{w},x^{i}>\right) & ,y^{i}.<\mathbf{w}',x^{i}>\geq 1 \quad ,y^{i}.<\mathbf{w},x^{i}><1\\ \left(1-y^{i}<\mathbf{w}',x^{i}>\right) & ,y^{i}.<\mathbf{w}',x^{i}><1 \quad ,y^{i}.<\mathbf{w},x^{i}>\geq 1\\ 0 & ,y^{i}.<\mathbf{w}',x^{i}>\geq 1 \quad ,y^{i}.<\mathbf{w},x^{i}>\geq 1\end{cases}$$

$f\left(\mathbf{w}'\right)\geq f\left(\mathbf{w}\right)+\langle g,\mathbf{w}'-\mathbf{w}\rangle$
We can also prove this if we satisy this :

$f\left(\mathbf{w}'\right)-f\left(\mathbf{w}\right)\geq\langle g,\mathbf{w}'-\mathbf{w}\rangle$
So the i-th element of the expression on the LHS is s^{i}.

Let's see RHS of the condition:

$\langle g,\mathbf{w}'-\mathbf{w}\rangle=\sum_{i=1}^{n}h^{i}\left\langle\mathbf{w}'-\mathbf{w}\right\langle$

ith element of $\langle g,\mathbf{w}'-\mathbf{w}\rangle, b^{i}=\begin{cases}\left\langle -y^{i}x^{i},\mathbf{w}'-\mathbf{w}\right\rangle & ,y^{i}\left\langle\mathbf{w},x^{i}\right\rangle<1\\ 0 & ,y^{i}\left\langle\mathbf{w},x^{i}\right\rangle\geq 1\end{cases}$

So, simplifying and using the property of inner product

$$b^i = \begin{cases} -y^i \left\langle \mathbf{w}' - \mathbf{w}, x^i \right\rangle & , y^i \left\langle \mathbf{w}, x^i \right\rangle < 1 \\ 0 & , y^i \left\langle \mathbf{w}, x^i \right\rangle \geq 1 \end{cases}$$

$$b^i = \begin{cases} y^i \left\langle \mathbf{w} - \mathbf{w}', x^i \right\rangle & , y^i \left\langle \mathbf{w}, x^i \right\rangle < 1 \\ 0 & , y^i \left\langle \mathbf{w}, x^i \right\rangle \geq 1 \end{cases}$$

Now comparing LHS and RHS both. There are two cases, First when $y^i . \left\langle \mathbf{w}, x^i \right\rangle < 1$

Then, b^i, s^i

$$s^i = \begin{cases} y^i \left\langle \mathbf{w} - \mathbf{w}', x^i \right\rangle & , y^i . \left\langle \mathbf{w}', x^i \right\rangle \ < 1 \\ -\left(1 - y^i \left\langle \mathbf{w}, x^i \right\rangle\right) & , y^i . \left\langle \mathbf{w}', x^i \right\rangle \ \geq 1 \end{cases}$$

which is equal to

$$s^i = \begin{cases} y^i \left\langle \mathbf{w} - \mathbf{w}', x^i \right\rangle & , y^i . \left\langle \mathbf{w}', x^i \right\rangle \ < 1 \\ y^i \left\langle \mathbf{w}, x^i \right\rangle - 1 & , y^i . \left\langle \mathbf{w}', x^i \right\rangle \ \geq 1 \end{cases}$$

So when $y^i . \left\langle \mathbf{w}', x^i \right\rangle \ < 1$ then $\quad s^i = b^i = y^i \left\langle \mathbf{w} - \mathbf{w}', x^i \right\rangle$

And when $y^i . \left\langle \mathbf{w}', x^i \right\rangle \ \geq 1$ then $\quad s^i \geq b^i$. Which can be observed as follows :
Since $y^i . \left\langle \mathbf{w}', x^i \right\rangle \ \geq 1$
Therefore, $-y^i . \left\langle \mathbf{w}', x^i \right\rangle \ \leq -1$

Adding $y^i . \left\langle \mathbf{w}, x^i \right\rangle$ on both the sides of inequality,
Therefore, $y^i . \left\langle \mathbf{w} - \mathbf{w}', x^i \right\rangle \ \leq y^i . \left\langle \mathbf{w}, x^i \right\rangle - 1$
Which is same as $b^i \leq s^i$
For the second case $y^i . \left\langle \mathbf{w}, x^i \right\rangle \geq 1$

Then, b^i, s^i

$b^i = 0$, and

$$s^i = \begin{cases} 1 - y^i \left\langle \mathbf{w}', x^i \right\rangle & , y^i . \left\langle \mathbf{w}', x^i \right\rangle \ < 1 \\ 0 & , y^i . \left\langle \mathbf{w}', x^i \right\rangle \ \geq 1 \end{cases}$$

So when $y^i . \left\langle \mathbf{w}', x^i \right\rangle \ \geq 1$ then $\quad s^i = b^i = 0$

and when $y^i . \left\langle \mathbf{w}', x^i \right\rangle \ < 1$ then $\quad s^i > b^i$. Which can be observed as follows :
So, $y^i . \left\langle \mathbf{w}', x^i \right\rangle \ < 1$

So, $1 - y^i . \left\langle \mathbf{w}', x^i \right\rangle \ > 0$

Since $s^i = 1 - y^i . \left\langle \mathbf{w}', x^i \right\rangle$ and $b^i = 0$

so the inequality becomes, $s^i > b^i$

Hence $s^i \geq b^i$

So concluding from both the cases, we can say that $s^i \geq b^i$ for any **i**

Therefore, $f\left(\mathbf{w}'\right) - f\left(\mathbf{w}\right) \geq \langle g, \mathbf{w}' - \mathbf{w}\rangle$
$f\left(\mathbf{w}'\right) \geq f\left(\mathbf{w}\right) + \langle g, \mathbf{w}' - \mathbf{w}\rangle$
Hence, **g** is a member of the subdifferential of f at **w**.

Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

Assignment Number: 1
Student Name: Nikhil Mittal
Roll Number: 17111056
Date: September 10, 2017

QUESTION 6

Part 1 solution:

Used k-NN algorithm with the Euclidean metric to perform classification for different values of k = 1, 2, 3, 5, 10.
The test errors (no. of the 20K points that were incorrectly classified) :

For k = 1, No. of points incorrectly classified = 4815

$Error = \frac{4815}{20000} = 0.24075$

For k = 2, No. of points incorrectly classified = 4815

$Error = \frac{4815}{20000} = 0.24075$

For k = 3, No. of points incorrectly classified = 3872

$Error = \frac{3872}{20000} = 0.1936$

For k = 5, No. of points incorrectly classified = 3581

$Error = \frac{3581}{20000} = 0.17905$

For k = 10, No. of points incorrectly classified = 3348

$Error = \frac{3348}{20000} = 0.16915$

Figure **4**, Graph showing test accuracies (fraction of the 20K points that were correctly classified) vs k.

Observation: As the value of k increases the accuracy of our experiment increases. So we can infer that on increasing the no. of nearest neighbours the accuracy of k-NN may increase, but this we have observed only for small values of k.

k	Accuracy
1	0.75925
2	0.75925
3	0.8064
5	0.82095
10	0.8326

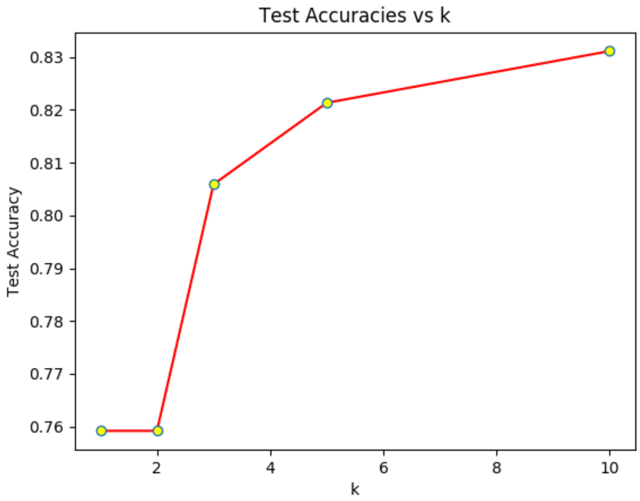

Figure 3: Plot of Test accuracies vs k

I also ran K-NN for values of k other than those provided in question i.e., for these set of k values= [1, 2, 3, 5, 10, 15, 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000, 20000] . The plot for accuracy vs K is :

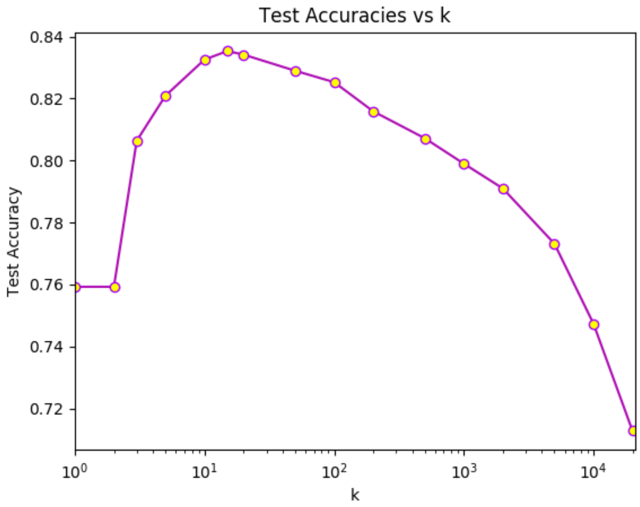

Figure 4: Plot of Accuracy vs k

Observation : The Accuracy of KNN increases only till a small value of k, and then it starts to decrease. Accuracy increases only till k = 15 and then starts to decrease slowly and steadily.

These are the vlues observed :

k	Accuracy
1	0.75925
2	0.75925
3	0.8064
5	0.82095
10	0.8326
15	0.8353
20	0.8342
50	0.829
100	0.8253
200	0.8159
500	0.8072
1000	0.799
2000	0.791
5000	0.773
10000	0.7473
20000	0.7127

The above table shows different values of Accuracy for values of k of different orders.

Part 2 solution:

The validation technique used : k-fold validation, the given training data is partitioned into k equal sized subsets. Of the k subsets, a single subset is the validation data for testing the model (or our new test data), and remaining $k - 1$ subsets make up our training data. Repeat this process k times (the folds), with each of the k subsets used exactly once as the validation data. Take the average of the accuracies of the k results obtained from the folds to give an accuracy.

Used 5-fold validation, with the following possible values of k in k-NN = [1, 2, 3, 5, 10, 20]
Results are as follows :
k = 1, 5-fold accuracies = (0.7628, 0.7647, 0.7588, 0.7652, 0.7578)
k = 2, 5-fold accuracies = (0.7628, 0.76475, 0.7588, 0.76525, 0.7578)
k = 3, 5-fold accuracies = (0.8083, 0.8116, 0.8042, 0.8069, 0.8014)
k = 5, 5-fold accuracies = (0.8187, 0.8262, 0.815, 0.8185, 0.8165)
k = 10, 5-fold accuracies = (0.8294, 0.8357, 0.8285, 0.8313, 0.8303)
k = 15, 5-fold accuracies = (0.828, 0.8369, 0.829, 0.8313, 0.8306)
k = 20, 5-fold accuracies = (0.8282, 0.835, 0.8279, 0.833, 0.83)
k = 100, 5-fold accuracies = (0.8197, 0.8225, 0.8205, 0.8251, 0.8216)

k	avg(Accuracy)
1	0.7619
2	0.7619
3	0.8064
5	0.819
10	0.8310
15	0.8311
20	0.8308
100	0.8219

So a good value of k, based on our validation technique is the one with highest accuracy which is obtained in the case when k = 15, so k can be chosen as 10 or 15 as not much difference.

Part 3 solution :

I tried this for different k values as I faced an issue that my training process for LMNN metric got killed when training from whole data. So instead of training for whole data, learned the LMNN metric using a fraction of data. So the results obtained is:

For k = 10, and 1/6th of the training data LMNN metric is learnt and then accuracy is calculated using this metric instead of euclidean distance.

1) When maximum No. of iterations = 2000
Accuracy obtained is 0.83625 for k = 10 and 1/6th of training data.

2) When maximum No. of iterations = 5000
Accuracy obtained is 0.83605 for k = 10 and 1/6th of training data.

3) When maximum No. of iterations = 8000
Accuracy obtained is 0.8361 for k = 10 and 1/6th of training data.

Therefore fixing no. of iterations to 2000 for model.

Extra Credit part :

I have used two techniques : ITML and LSML. Following are the results :

1) Using the ITML technique with 1/4th of the training data and k = 10 and fixing the no. of constraints to 50, the accuracy obtained is = 0.8273.

2)Using the ITML technique with 1/4th of the training data and k = 10 and fixing the no. of constraints to 100, the accuracy obtained is = 0.8298.

3)When k = 15 for 1/4th of the training data and fixing the no. of constraints to 500, the accuracy obtained is 0.83375.

Hence, storing and using the last observation.

Following are the results using LSML technique:

1) Using 1/4th of the training data for fitting the model and k = 10 and fixing the no. of constraints to 100, the accuracy obtained is = 0.8141.

2) Using 1/4th of the training data and k = 10, fixing the no. of constraints = 500, the accuracy obtained is 0.8153.

3) Using 1/3rd of the training data for fitting the model and k = 10, fixing the no. of constraints = 100, accuracy obtained is 0.8154.

4) Using half of the training data, fixing the no. of constraints = 1000 and k = 10, accuracy obtained is 0.8155.

Hence storing and using the model in the last observation.