ML-Assignment-1

Prateek yadav
109981
UG

Q1 X, Y discrete R.V. and

$P(X=x_1, Y=y_1) = 0.1$, $P(X=x_1, Y=y_2) = 0.1$

$P(X=x_2, Y=y_1) = 0.2$, $P(X=x_2, Y=y_2) = 0.2$

$P(X=x_3, Y=y_1) = 0.1$, $P(X=x_3, Y=y_2) = 0.3$

Compute $P(X), P(Y), P(X|Y), P(Y|X)$.

$P(X) \rightarrow$

i	1	2	3
$P(X=x_i)$	0.2	0.4	0.4

$$P(X=x_i) = \sum_{\forall j} P(X=x_i, Y=y_j)$$

$P(X=x_1) = 0.1+0.1 = 0.2$

$P(X=x_2) = 0.2+0.2 = 0.4$

$P(X=x_3) = 0.1+0.3 = 0.4$

$\sum_{\forall i} P(X=x_i) = 1.$

$P(Y) \rightarrow$

i	1	2
$P(Y=y_i)$	0.4	0.6

Similar computation

$P(Y=y_1) = 0.1+0.2+0.1 = 0.4$

$P(Y=y_2) = 0.1+0.2+0.3 = 0.6$

$\sum_{\forall i} P(Y=y_i) = 1$

$$P(X|Y) = \frac{P(X,Y)}{P(Y)} \Rightarrow P(X|Y=y_1) = \frac{P(X, Y=y_1)}{P(Y=y_1)}$$

$$P(X|Y=y_2) = \frac{P(X, Y=y_2)}{P(Y=y_2)}$$

$$P(X=x_i|Y=y_j) = \frac{P(X=x_i, Y=y_j)}{P(y_j)}$$

$$P(X=x_1|Y=y_1) = \frac{P(X=x_1, Y=y_1)}{P(Y=y_1)}$$

$$= \frac{0.1}{0.4} = \frac{1}{4}$$

$P(X|Y)$

	x_1	x_2	x_3
$P(X\|Y=y_1)$	0.25	0.5	0.25
$P(X\|Y=y_2)$	1/6	2/6	3/6

Similarly rest can be found answer in table

Similarly $P(Y|X) = \frac{P(Y,X)}{P(X)}$ i.e $P(Y=y_i|X=x_i) = \frac{P(Y=y_i, X=x_i)}{P(X=x_i)}$

$P(Y=y_1|X=x_1) = \frac{P(Y=y_1, X=x_1)}{P(X=x_1)}$

$= \frac{0.1}{0.2} = 1/2$

Similarly rest

$P(Y|X) \longrightarrow$

	y_1	y_2
$P(Y\|X=x_1)$	$1/2$	$1/2$
$P(Y\|X=x_2)$	$1/2$	$1/2$
$P(Y\|X=x_3)$	$1/4$	$3/4$

Q2

$p(x|\omega_1) \sim N(0,1)$

$p(x|\omega_2) \sim N([1 \cdot 1]^T, I)$.

$P(\omega_1) = P(\omega_2) = 1/2$.

Predict decision Boundary.

$\cancel{P(\omega|x)}$ $P(\omega=\omega_i|x) = \frac{P(x|\omega=\omega_i) \cdot P(\omega_i)}{\sum_{j=1}^{2} P(x|\omega=\omega_j) \cdot P(\omega_i)}$

$P(\omega_i) = 1/2 \quad \forall i \in \{1,2\}$.

$\Rightarrow P(\omega=\omega_i|x) = \frac{P(x|\omega=\omega_i)}{\sum_{j=1}^{2} P(x|\omega=\omega_j)}$

If $\frac{P(\omega=\omega_1|x)}{P(\omega=\omega_2|x)} > 1 \Rightarrow$ Predict ω_1

else Predict ω_2

let $\cancel{P(\omega=\omega_1|x) = \eta(x)}$, ~~then~~ $\cancel{P(\omega=\omega_2|x) = 1-\eta(x)}$.

$\frac{\eta(x)}{1-\eta(x)}$

$$\frac{P(\omega=\omega_1|x)}{P(\omega=\omega_2|x)} = \frac{P(x|\omega=\omega_1)\cdot P(\omega_1)}{P(x|\omega=\omega_2)\cdot P(\omega_2)} = \frac{P(x|\omega=\omega_1)\cdot 1/2}{P(x|\omega=\omega_2)\cdot 1/2}$$

$$= \frac{P(x|\omega=\omega_1)}{P(x|\omega=\omega_2)}$$

Now $P(x|\omega=\omega_1) = \frac{1}{(\sqrt{2\pi})^2 |I|^{1/2}} e^{-1/2 (x-0)^T I^{-1} (x-0)}$

$$= \frac{1}{(\sqrt{2\pi})^2 \cdot 1} \cdot e^{-1/2\, x^T x}.$$

$P(x|\omega=\omega_2) = \frac{1}{(\sqrt{2\pi})^2 |I|^{1/2}} e^{-1/2 (x-\mu)^T I^{-1} (x-\mu)}$, $\mu = \begin{bmatrix}1\\1\end{bmatrix}$.

$$= \frac{1}{(\sqrt{2\pi})^2 \cdot 1} \cdot e^{-1/2 (x-\mu)^T (x-\mu)}.$$

So $\frac{P(\omega=\omega_1|x)}{P(\omega=\omega_2|x)} = \frac{P(x|\omega=\omega_1)}{P(x|\omega=\omega_2)} = \frac{e^{-1/2\, x^T x}}{e^{-1/2 (x-\mu)^T (x-\mu)}}$

$L(x) = \ln\left(\frac{P(\omega=\omega_1|x)}{P(\omega=\omega_2|x)}\right) = \frac{-1}{2}\left(x^T x - (x-\mu)^T (x-\mu)\right).$

$= -\frac{1}{2}\left(x^T x - (x^T x - \mu^T x - x^T\mu + \mu^T\mu)\right) = -\frac{1}{2}\left(-\mu^T x - x^T \mu + \mu^T \mu\right)$

So if $L(x) > 0$ then ω_1 else ω_2

$L(x) = -\frac{1}{2}\left(-[1\ 1]\begin{bmatrix}x_1\\x_2\end{bmatrix} - [x_1\ x_2]\begin{bmatrix}1\\1\end{bmatrix} + [1\ 1]\begin{bmatrix}1\\1\end{bmatrix}\right)$

$= -\frac{1}{2}\left(-(x_1+x_2) - (x_1+x_2) + 2\right) \Rightarrow \frac{1}{\cancel{2}}\left(\cancel{2}(x_1+x_2-1)\right)$

$= (x_1+x_2-1)$

So predict ω_1 if $x_1+x_2 > 1$ else ω_2

Q3. $er_D^c[h] = E_{(x,y)\sim D}[\ell_c(y, h(x))]$.

$$\ell_c(y,\hat{y}) = \begin{cases} c & (y=-1)\wedge(\hat{y}=1) \\ 1-c & (y=+1)\wedge(\hat{y}=-1) \\ 0 & y=\hat{y} \end{cases}$$

prediction of ~~h(x)~~). hypothesis

$E[\text{loss}(\hat{y}=1)] = (1-\eta(x))(1-c)$.

$E[\text{loss}(\hat{y}=-1)] = \eta(x)c$

Decision if $E(\text{loss}(\hat{y}=1)) < E[\text{loss}(\hat{y}=-1)]$ predict $\hat{y}=1$

else predict $\hat{y}=-1$

$(1-\eta(x))(1-c) < \eta(x)c$.

$1-\eta(x)-c+\cancel{c\eta(x)} < \cancel{\eta(x)c}$

$1-c < \eta(x)$

so if $\eta(x) > 1-c$ predict/set $\hat{y}=1$

else set/predict $\hat{y}=-1$.

If $a,b>0$ are used in loss function

$$\ell_c(y,\hat{y}) = \begin{cases} b & (y=-1)\wedge(\hat{y}=1) \\ a & (y=1)\wedge(\hat{y}=-1) \\ 0 & y=\hat{y} \end{cases}$$

$E[\text{loss}(\hat{y}=1)] = (1-\eta(x))\cdot a$, $E[\text{loss}(\hat{y}=-1)] = \eta(x)b$.

So if $(1-\eta(x))a < \eta(x)b$ predict $\hat{y}=1$

else $\hat{y}=-1$.

$\Rightarrow$ if $\eta(x) > \frac{a}{a+b}$ predict/set $\hat{y}=1$

else $\hat{y}=-1$.

E0-270 Machine Learning Assignment-1

Name: Prateek Yadav
Date: 2 Feb 2017

Instructions

This Latex File contains Question number 4 and 5. Question 1,2,3 are submitted handwritten in room 309 CSA.

Question-4

1. a

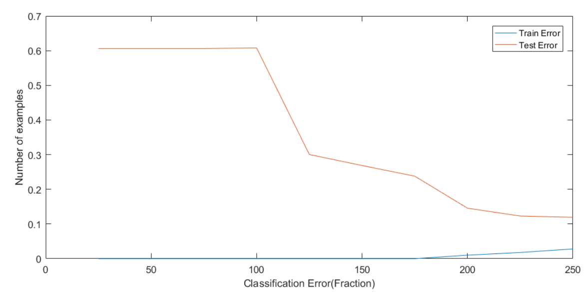

Figure 1: Learning Curve: Classification error(fractional) on y-axis and Number of examples used while training. Training is done on train.txt provided in the data folder.

2. b

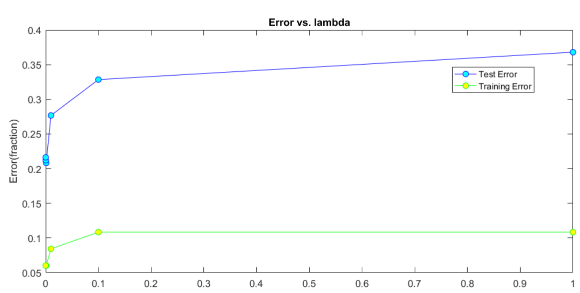

Figure 2:

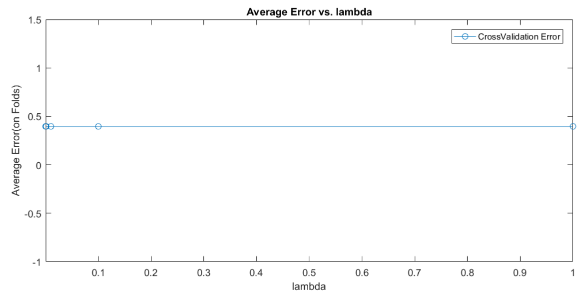

Figure 3: Learning Curve: Classification error(fractional) on y-axis and Number of examples used while training. Training is done on train.txt provided in the data folder.

Question-5

1. Naive Bayes Training Accuracy: 99.75%
   Test Accuracy: 76%
   Confusion Matrix:

118	33
39	110

2. Perceptron

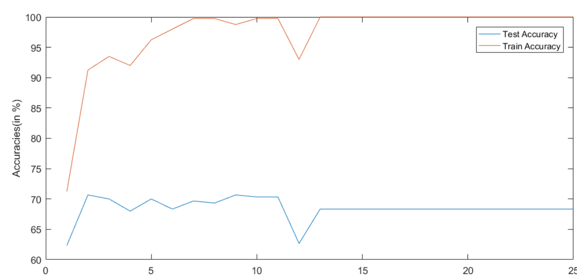

Figure 4: Training and Test Accuracies on y-axis and Number of iterations on x-axis

3. Winnow

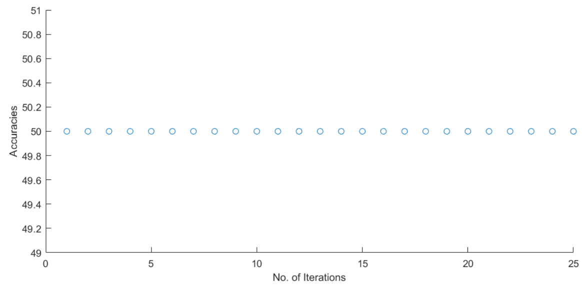

Figure 5: Training and Test Accuracies on y-axis and Number of iterations on x-axis

4. d
   The word with highest positive weights are: (in decending order)
   'titanic', 'story', 'it', 'bit', 'williams', 'also', '10', 'has', 'her', 'episode'
   The word with highest absolute weights are: (in decending order)
   'titanic', 'bad', 'seagal', 'no','they' 'story', 'would', 'out', 'worst', 'it'

5. e
   The value of γ is bounded by $(R * ||W||)/\sqrt{(}M)$ so $\gamma <= 2.0753e + 06$

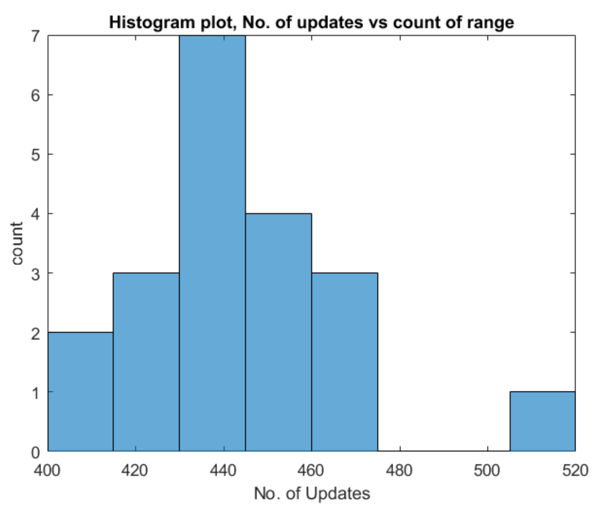

Figure 6: Number of updates on X-axis and Count for each class on y-axis. (Note counts should sum to 1 and they do.)

6. f
   Trained the perceptron using different values of eta ranging from 0.05 to

0.6 with increments of 0.05.

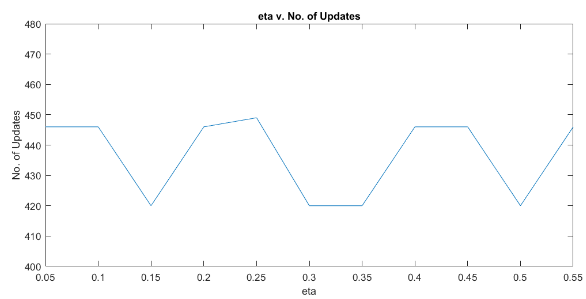

Figure 7: Number of updates on y-axis and values of eta on x-axis.

Table 1: This table contains For each value the number of eta the number updates made, train and test accuracies.

eta	Number of Updates	Train Accuracy(in %)	Testing Accuracy(in %)
0.05	446	100	69
0.1	446	100	69
0.15	420	100	75
0.2	446	100	69
0.25	449	100	68.33333333
0.3	420	100	75
0.35	420	100	75
0.4	446	100	69
0.45	446	100	69
0.5	420	100	75
0.55	446	100	69
0.6	420	100	75

7. RunTime Analysis
   The runtime Complexity of Perceptron algorithm is may be $O(n^2)$ where n being the number of training examples used.

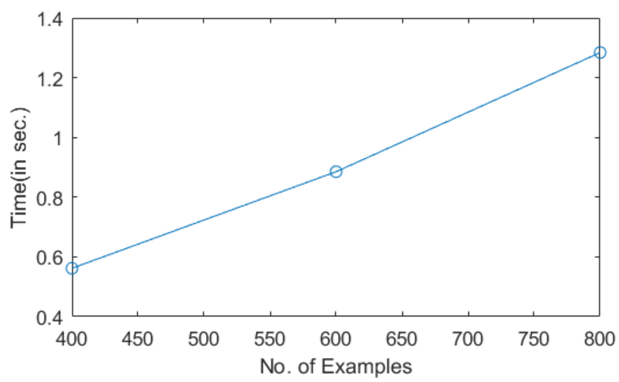

Figure 8: Runtime of function on y-axis and number of examples on x-axis.