CS 6140 Machine Learning
Northeastern University, Fall 2018

Homework 3

Name: Ruchitha Midigarahalli Shanmugha Sundar NUID: 001838207

1 Problem 1

Entropy is given by the formula:

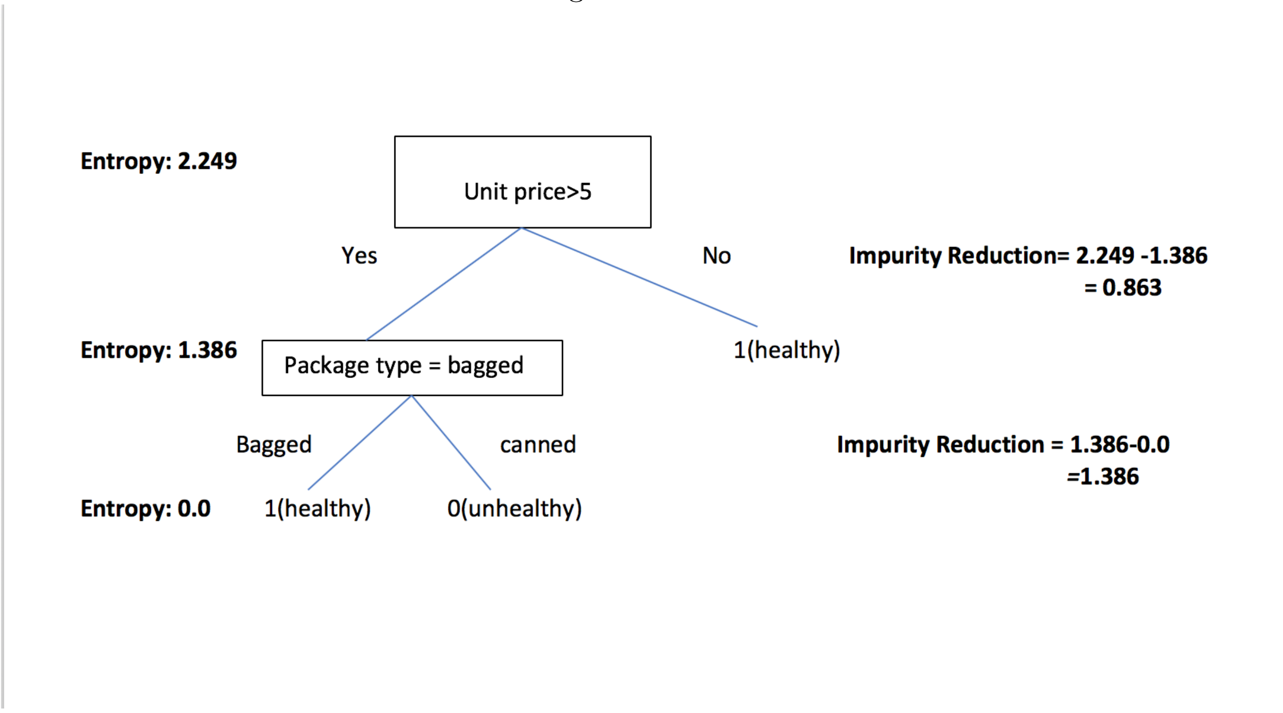
$$E = -|S|(p_s log P_s + (1 - P_s) log(1 - P_s))$$

For this dataset, we have $|S| = 4$ and $P_s = \frac{3}{4} = 0.75$

Using this entropy equation at the root node, we have: Entropy = 2.249

For the first level package-type, unit-price>5 and contains>5, all has same entropy value = 1.386. So we can split on any condition.

On choosing Unit-price >5 for the first level split, we will be left with package-type and contains>5. They both have the same entropy value = 0.0
On choosing package type, we can notice that all the leaf nodes are pure data set.
Decision tree for this is as shown in the figure below:



2 Problem 2

]

Let

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

Initialize: $D_1(i) = 1/N$

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$$

Recurrence update in the algorithm is done as given below:

$$D_{T+1}(i) = D_1(i) * \frac{e^{-y_i \alpha_1 h_1(x_i)}}{Z_1} * ... * \frac{e^{-y_i \alpha_T h_T(x_i)}}{Z_T}$$

$$= \frac{D_1(i) exp\Bigg(-y_i \sum_{t=1}^{T} \alpha_t h_t(x_i)\Bigg)}{\Pi_{t=1}^{T} Z_t}$$

W.K.T.

$$H(X) = sign(F(x))$$

If $H(x)! = y$ then $yF(x) <= 0$
$\Longrightarrow$

$$e^{-yF(x)} >= 1$$

$\Longrightarrow$

$$1(H(x)! = y) <= e^{-yF(x)}$$

Now, Training error:

$$Z_t = \sum_{i=1}^{m} D_1(i) 1(H(x_i)! = y_i)$$

$$<= \sum_{i=1}^{m} D_1(i) exp(-y_i F(x_i))$$

$$= \sum_{i=1}^{m} D_{T+1}(i) \Pi_{t=1}^{T} Z_t$$

$$= \Pi_{t=1}^{T} Z_t - - - Equation1$$

The above three equation uses the fact that D_{T+1} is a distribution that sums to 1.
Now using our choice of α_t, we have:

$$Z_t = \sum_{t=1}^{m} D_t(i) e^{-\alpha_t y_i h_t(x_i)}$$

$$= \sum_{i:y_i = h_t(x_i)} D_t(i) e^{-\alpha_t} + \sum_{i:y_i != h_t(x_i)} D_t(i) e^{\alpha_t}$$

By definition of ϵ_t :

$$= e^{-\alpha_t}(1 - \epsilon_t) + e^{\alpha_t}\epsilon_t$$

By choosing α_t to minimize the expression, we have:

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

substituting $\epsilon_t = \frac{1}{2} - \gamma_t$ we have:

$$= e^{-\alpha_t}(\frac{1}{2} + \gamma_t) + e^{\alpha_t}(\frac{1}{2} - \gamma_t)$$

W.K.T $(1 + x) <= e^x$ for all real value of x:

$$= \sqrt{1 - 4\gamma_t 2} <= e^{-2\gamma_t^2} - - - - - Equation2$$

This proof shows that the training error is upper bounded by $\Pi_t Z_t$. we can minimize this expression by minimizing Z_t . By the equation of Z_t, we know that it can be minimized over the choice of α_t which is greedily chosen at each round t. From Equation 2 we can notice that the training error at most is an exponentially decreasing function for any $\gamma > 0$.This proves that Ada-Boost minimizes exponential loss.
]
Initially,

$$D_t(1) = 1/8 = 0.125$$

Using the below equations we obtain the values in the table shown below:

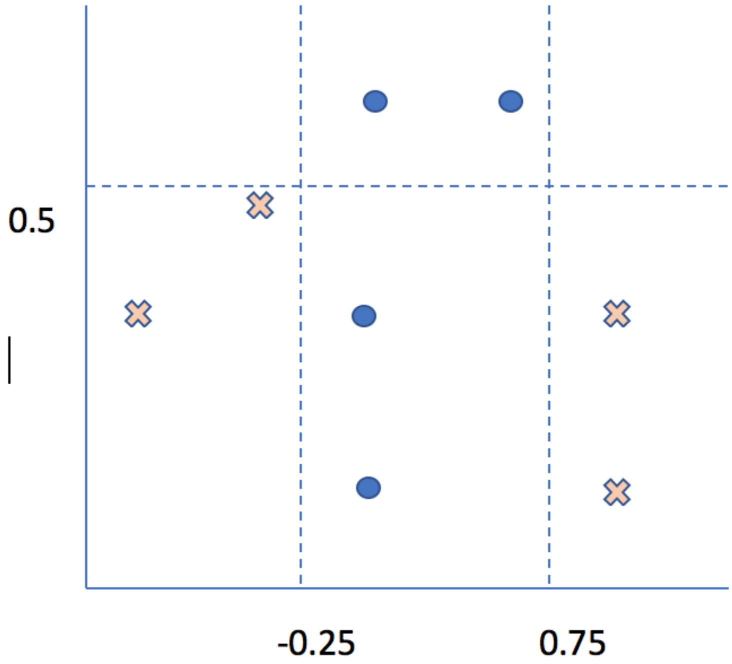
$$\epsilon_t = \sum_{i=1}^{n} w_i \text{ for all misclassified data points}$$

$$\alpha_t = \frac{1}{2} log\left(\frac{1 + \epsilon_t}{1 + \epsilon_t}\right)$$

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

$$D_{t+1}(i) = D_t(i)\frac{exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

t	ϵ_t	α_t	Z_t	$D_T(1)$	$D_T(2)$	$D_T(3)$	$D_T(4)$	$D_T(5)$	$D_T(6)$	$D_T(7)$	$D_T(8)$
1	0.250	0.549	0.866	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125
2	0.167	0.805	0.745	0.083	0.083	0.083	0.083	0.250	0.250	0.083	0.083
3	0.100	1.099	0.600	0.250	0.250	0.050	0.050	0.150	0.150	0.050	0.050



Ada-boost has zero training error. AdaBoost is a non-linear ensemble classifier, which involves several weak learners. As a result it out performs the single decision stump.

3 Problem 3

]
We know that sigmoid function:

$$\sigma(z) = \frac{1}{1+e^{-z}}$$

On taking a derivative w.r.t to z we get:

$$\sigma^{'}(z) = \sigma(z)(1-\sigma(z)) - - - Equation1$$

Squared Loss is given by :

$$J = \frac{1}{n}\sum_{i=1}^{n}(o^{(i)} - y^{(i)}) - - - - Equation2$$

Taking derivative w.r.t $o^{(i)}$ we have

$$\frac{\delta J}{\delta o^{(i)}} = (o^{(i)} - y^{(i)}) - - - - Equation3$$

On taking derivative w.r.t w_{12} and using chain rule, we have:

$$\frac{\delta J}{\delta w_{12}} = \sum_{i=1} n \frac{\delta E}{\delta o^{(i)}} \frac{\delta o^{(i)}}{\delta z_i} \frac{\delta z_i}{\delta h_2} \frac{\delta h_2}{\delta z_{h_2}} \frac{\delta z_{h_2}}{\delta w_{12}} - - Equation4$$

By using Equation 1 we have:

$$\frac{\delta o_i}{\delta z_i} = o^{(i)}(1 - o^{(i)}) - - - Equation5$$

and

$$\frac{\delta h_2}{\delta z_{h_2}} = h_2(1 - h_2) - -Equation6$$

Also we have,

$$Z_{h_2} = \sum_{i=1}^{m} w_{ij} o^{(i)}$$

Differentiation we have:

$$\frac{\delta Z_{h_2}}{\delta w_{12}} = \frac{\delta \sum_{i=1}^{m} w_{ij} x_i}{\delta w_{12}} = x_1^{(i)} - -Equation7$$

Also,

$$Z_i = \sum_{i=1}^{m} w_{ij} h_i$$

Differentiation we have:

$$\frac{\delta Z_i}{\delta h_2} = \frac{\delta \sum_{i=1}^{m} w_{ij} h_i}{\delta h_2} = w_2^{(2)} - -Equation7 - - - Equation8$$

Using Equation 3,5,6,7 and 8 in 4 we have:

$$\frac{\delta J}{\delta w_{12}} = \frac{1}{n} \sum_{i=1}^{n} (o^{(i)} - y^{(i)}) o^{(i)} (1 - o^{(i)}) w_2^{[2]} h_2 (1 - h_2) x_1^{(i)}$$

]
The below set of weights forms a triangle:

$$\begin{bmatrix} 1 & 0 & 4 \\ 1 & 4 & 0 \\ 5 & 1 & 1 \end{bmatrix} \begin{bmatrix} x1 \\ x2 \\ 1 \end{bmatrix}$$

We are assuming all the points are correctly classified by the lines of this triangle.
For all the points to be within this triangle, output of the hidden neurons should be [1,1,1]
]
Even though the output from the hidden layer neurons are linear, when given to the output neuron it becomes non-linear. Step-function is not capable of handling non-linear problems. Hence this can not be solved using step-function.