HW 2 SUNID = BENMA

1a. $Loss(x, y, w) = \max\{0, 1 - w \cdot \phi(x) y\}$

$$\nabla_w Loss = \begin{cases} 0 & , 0 > 1 - (w \cdot \phi(x) \cdot y) \\ -\phi(x) y & , \text{else} \end{cases}$$

① Vector $\phi(x) = \{pretty:1, bad:1\} \Rightarrow \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$, $Y = -1$

$w \leftarrow w - \gamma \nabla_w Loss = w - 0.5 \nabla_w Loss$

$margin = w \cdot \phi(x) y = 1 \Rightarrow \nabla_w = -\phi(x) y$

$$\nabla_w = -\phi(x) y = -1 \cdot \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \cdot -1 = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$w \leftarrow w - 0.5 \nabla_w = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} - 0.5 \cdot \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -0.5 \\ 0 \\ -0.5 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

② Vector $\phi(x) = \{good:1, plot:1\} \Rightarrow \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$, $Y = +1$

$$margin = w \cdot \phi(x) y = \begin{bmatrix} -0.5 \\ 0 \\ -0.5 \\ 0 \\ 0 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \cdot 1 = 0 < 1 \Rightarrow \nabla_w = -\phi(x) y$$

$$\nabla_w = -1 \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \cdot 1 = \begin{bmatrix} 0 \\ -1 \\ 0 \\ -1 \\ 0 \\ 0 \end{bmatrix}$$

$$w \leftarrow w - 0.5 \nabla_w = \begin{bmatrix} -0.5 \\ 0 \\ -0.5 \\ 0 \\ 0 \\ 0 \end{bmatrix} - 0.5 \begin{bmatrix} 0 \\ -1 \\ 0 \\ -1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -0.5 \\ 0.5 \\ -0.5 \\ 0.5 \\ 0 \\ 0 \end{bmatrix}$$

③ Vector $\phi(x) = \{not:1, good:1\} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$, $Y = -1$

$$margin = w \cdot \phi(x) y = \begin{bmatrix} -0.5 \\ 0.5 \\ -0.5 \\ 0.5 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \cdot (-1) < 1 \Rightarrow \nabla_w = -\phi(x) y$$

$$\nabla_w = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \\ -1 \\ 0 \end{bmatrix}$$

$$w \leftarrow w - 0.5\nabla_w = \begin{bmatrix} -0.5 \\ 0.5 \\ -0.5 \\ 0.5 \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 0.5 \\ 0 \\ 0 \\ 0.5 \\ 0 \end{bmatrix} = \begin{bmatrix} -0.5 \\ 0 \\ -0.5 \\ 0.5 \\ -0.5 \\ 0 \end{bmatrix}$$

$\phi(x) = \{\text{pretty}:1, \text{scenery}:1\} \rightarrow \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$, $Y = +1$

④ $\text{margin} = w \cdot \phi(x) Y = \begin{bmatrix} -0.5 \\ 0 \\ -0.5 \\ 0.5 \\ -0.5 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \cdot (+1) < 1$

$\nabla_w \text{Loss}(x, y, w) = -\phi(x) Y = \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \\ -1 \end{bmatrix}$

$w \leftarrow w - \eta \nabla_w \text{Loss}(x, y, w) = \begin{bmatrix} -0.5 \\ 0 \\ -0.5 \\ 0.5 \\ -0.5 \\ 0 \end{bmatrix} - \begin{bmatrix} -0.5 \\ 0 \\ 0 \\ 0 \\ -0.5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -0.5 \\ 0.5 \\ -0.5 \\ 0.5 \end{bmatrix}$

---

15. $\phi(x) = \{\text{good}:1\} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$, $Y_1 = +1$

$\phi_2(x) = \{\text{bad}:1\} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, $Y_2 = -1$

$\phi_3(x) = \{\text{not}:1, \text{good}:1\} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$, $Y_3 = -1$

$\phi_4(x) = \{\text{not}:1, \text{bad}:1\} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$, $Y_4 = +1$

$$\text{margin} = w \cdot \phi(x) Y = \begin{bmatrix} \phi_1(x) Y_1 \\ \phi_2(x) Y_2 \\ \phi_3(x) Y_3 \\ \phi_4(x) Y_4 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \\ -1 & -1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \begin{bmatrix} b \\ b \\ b \\ b \end{bmatrix}$$

$$= \begin{bmatrix} w_2 \cdot Y_1 \\ w_3 \cdot Y_2 \\ w_1 + w_2 \cdot Y_3 \\ w_1 + w_3 \cdot Y_4 \end{bmatrix} = \begin{bmatrix} w_2 \\ -w_3 \\ -w_1 - w_2 \\ w_1 + w_3 \end{bmatrix}$$

If margin >0 then $\begin{cases} w_2>0 \\ -w_3>0 \\ -w_1-w_2>0 \\ w_1+w_3>0 \end{cases} \Rightarrow \begin{cases} w_2>0 \Rightarrow -w_2<0 \\ w_3<0 \Rightarrow -w_3>0 \\ w_1+w_2<0 \Rightarrow w_1<-w_2<0 \\ w_1+w_3>0 \Rightarrow w_1>-w_3>0 \end{cases}$

so w_1 has to meet both >0 and <0, w_1 does not exist

Therefore ~~no~~ no w in linear classifier can get 0 error

2a. $Loss(x,y,w) = (\sigma(w\cdot\phi(x))-y)^2 = [(1+e^{-w\phi(x)})^{-1}-y]^2$

2b. $\nabla_w Loss(x,y,w) = 2[(1+e^{-w\phi(x)})^{-1}-y] \cdot \underline{2\sigma(w\phi(x))}$

$= 2[(1+e^{-w\phi(x)})^{-1}-y]\cdot(-1)\cdot(-\phi(x))\cdot(1+e^{-w\phi(x)})^{-2}\cdot e^{-w\phi(x)}$

$= 2[(1+e^{-w\phi(x)})^{-1}-y]\cdot\phi(x)\cdot e^{-w\phi(x)}\cdot(1+e^{-w\phi(x)})^{-2}$

or $2(p-y)\cdot\phi(x)\cdot e^{-w\phi(x)}\cdot p^2$, $p=\sigma(w\cdot\phi(x))$

2c. $\nabla_w = 2(p-1)p^2\cdot\phi(x)\cdot e^{-w\phi(x)}$, $y=1$

$= 2\left(-\frac{e^{-w\phi(x)}}{1+e^{-w\phi(x)}}\right)\cdot\phi(x)\cdot e^{-w\phi(x)}\cdot(1+e^{-w\phi(x)})^{-2}$

$= -2\phi(x)\cdot e^{-2w\phi(x)}\cdot(1+e^{-w\phi(x)})^{-3}$ ≤ 0

when $w\to+\infty$, $e^{-2w}\to 0$, $e^{-2w\phi(x)}\to e^{-\infty}=0$

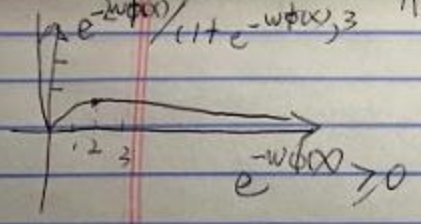
$\nabla_w Loss = -2\cdot\phi(x)\cdot 0\cdot(1+0)^{-3}=0$

So when $w=+\infty$, $\nabla_w Loss(x,y,w)=0$

2d. $\nabla_w Loss_{max} = ||-2\phi(x)e^{-2w\phi(x)} \cdot (1+e^{-w\phi(x)})^{-3}||_{max}$

$= 2||\phi(x)|| \cdot ||\frac{e^{-2w\cdot\phi(x)}}{(1+e^{-w\phi(x)})^3}||_{max}$

when $\nabla_w Loss$ max, $||\frac{e^{-2w\phi(x)}}{(1+e^{-w\phi(x)})^3}||_{max}$



Therefore when $e^{-w\phi(x)} = 2$, $||\frac{e^{-2w\phi(x)}}{(1+e^{-w\cdot\phi(x)})^3}||_{max} = \frac{4}{27}$

$\nabla_w Loss(x,y,w)_{max} = 2 \cdot ||\phi(x)|| \cdot \frac{4}{27} = \frac{8}{27}||\phi(x)||$

2e. $Loss(x,y,w) = (\sigma(w\cdot\phi(x)) - y)^2$

$Loss_{squared}(x,y',w) = (w\cdot\phi(x) - y')^2$

$\nabla Loss(x,y,w) = 2(\sigma(w\cdot\phi(x)) - y)\frac{\partial\sigma(w\cdot\phi(x))}{\partial w} = 0 \Rightarrow y = \sigma(w\cdot\phi(x))$

$\nabla Loss_{squared}(x,y',w) = 2(w\cdot\phi(x) - y')\cdot\phi(x) = 0 \Rightarrow y' = w\cdot\phi(x)$

$y = \sigma(w\cdot\phi(x)) = \sigma(y') = (1+e^{-y'})^{-1}$

$y(1+e^{-y'}) = 1$

$-y' = -\ln(y^{-1}-1)$

$y' = \ln(y^{-1}-1)^{-1}$

$= \ln(\frac{y}{1-y})$

3d.

=== walter hill's undisputed is like a 1940s warner bros . b picture , and i mean that as a compliment .

Truth: 1, Prediction: -1 [WRONG]

b 1 * 0.34 = 0.34

warner 1 * 0.21 = 0.21

bros 1 * 0.14 = 0.14

and 1 * 0.1 = 0.1

1940s 1 * 0.06 = 0.06

is 1 * 0.06 = 0.06

i 1 * 0.03 = 0.03

mean 1 * 0.03 = 0.03

, 1 * 0.03 = 0.03

a 2 * -0.03 = -0.06

. 2 * -0.03 = -0.06

like 1 * -0.07 = -0.07

hill's 1 * -0.09 = -0.09

undisputed 1 * -0.09 = -0.09

that 1 * -0.1 = -0.1

walter 1 * -0.11 = -0.11

picture 1 * -0.15 = -0.15

compliment 1 * -0.18 = -0.18

as 1 * -0.22 = -0.22

Explanation: The predictor have the wrong feature vector on the positive words such as “compliment” or like.

Fix: update the vector phi to have correct value.

=== home alone goes hollywood , a funny premise until the kids start pulling off stunts not even steven spielberg would know how to do . besides , real movie producers aren't this nice .

Truth: -1, Prediction: 1 [WRONG]

funny 1 * 0.39 = 0.39

real 1 * 0.35 = 0.35

start 1 * 0.28 = 0.28

spielberg 1 * 0.25 = 0.25

kids 1 * 0.24 = 0.24

home 1 * 0.24 = 0.24

steven 1 * 0.23 = 0.23

know 1 * 0.22 = 0.22

even 1 * 0.18 = 0.18

stunts 1 * 0.12 = 0.12

, 2 * 0.03 = 0.06

pulling 1 * 0.05 = 0.05

nice 1 * 0.05 = 0.05

do 1 * 0.05 = 0.05

until 1 * 0.01 = 0.01

producers 1 * -0.02 = -0.02

a 1 * -0.03 = -0.03

. 2 * -0.03 = -0.06

alone 1 * -0.06 = -0.06

this 1 * -0.07 = -0.07

the 1 * -0.08 = -0.08

would 1 * -0.08 = -0.08

aren't 1 * -0.1 = -0.1

to 1 * -0.11 = -0.11

off 1 * -0.11 = -0.11

how 1 * -0.13 = -0.13

not 1 * -0.15 = -0.15

hollywood 1 * -0.17 = -0.17

premise 1 * -0.19 = -0.19

movie 1 * -0.19 = -0.19

goes 1 * -0.27 = -0.27

besides 1 * -0.27 = -0.27

Explanation: the review has too many positive words such as “nice”, “funny”, etc. to make predictor to think this is a positive comment. In fact, it can’t read the tone.

Fix: need to add word in phrase or sentence and add function for predictor to understand the sarcasm.

=== a heady , biting , be-bop ride through nighttime manhattan , a loquacious videologue of the modern male and the lengths to which he'll go to weave a protective cocoon around his own ego .

Truth: 1, Prediction: -1 [WRONG]

ride 1 * 0.9 = 0.9

modern 1 * 0.22 = 0.22

ego 1 * 0.2 = 0.2

of 1 * 0.11 = 0.11

and 1 * 0.1 = 0.1

, 3 * 0.03 = 0.09

he'll 1 * 0.06 = 0.06

cocoon 1 * 0 = 0

videologue 1 * 0 = 0

loquacious 1 * 0 = 0

lengths 1 * 0 = 0

protective 1 * 0 = 0

weave 1 * 0 = 0

nighttime 1 * 0 = 0

be-bop 1 * 0 = 0

through 1 * -0.01 = -0.01

heady 1 * -0.02 = -0.02

. 1 * -0.03 = -0.03

biting 1 * -0.04 = -0.04

a 3 * -0.03 = -0.09

own 1 * -0.11 = -0.11

manhattan 1 * -0.14 = -0.14

his 1 * -0.15 = -0.15

the 2 * -0.08 = -0.16

which 1 * -0.16 = -0.16

to 2 * -0.11 = -0.22

go 1 * -0.23 = -0.23

male 1 * -0.31 = -0.31

around 1 * -0.35 = -0.35

Explanation: Too many words that do not assign values such as “cocoon”, “protective”, “loquacious”, etc. from the predictor, the rest of word reviews lead to the wrong prediction.

Fix: Add more new words to the dictionary of the vector

=== graças às interações entre seus personagens , o filme torna-se não apenas uma história divertida sobre uma curiosa perseguição , mas também um belo estudo de personagens .

Truth: 1, Prediction: -1 [WRONG]

de 1 * 0.17 = 0.17

uma 2 * 0.07 = 0.14

o 1 * 0.13 = 0.13

também 1 * 0.07 = 0.07

, 2 * 0.03 = 0.06

sobre 1 * 0.03 = 0.03

graças 1 * 0 = 0

perseguição 1 * 0 = 0

história 1 * 0 = 0

divertida 1 * 0 = 0

torna-se 1 * 0 = 0

seus 1 * 0 = 0

curiosa 1 * 0 = 0

às 1 * 0 = 0

interações 1 * 0 = 0

entre 1 * 0 = 0

estudo 1 * 0 = 0

personagens 2 * 0 = 0

. 1 * -0.03 = -0.03

belo 1 * -0.06 = -0.06

filme 1 * -0.06 = -0.06

apenas 1 * -0.07 = -0.07

não 1 * -0.14 = -0.14

mas 1 * -0.18 = -0.18

um 1 * -0.31 = -0.31

Explanation: The predictor does not understand non-English language.

Fix: need to add translator for the predictor or add non-English words as vector in the classifier.

=== if you are into splatter movies , then you will probably have a reasonably good time with the salton sea .

Truth: -1, Prediction: 1 [WRONG]

you 2 * 0.23 = 0.46

movies 1 * 0.35 = 0.35

time 1 * 0.31 = 0.31

good 1 * 0.29 = 0.29

if 1 * 0.27 = 0.27

into 1 * 0.24 = 0.24

are 1 * 0.22 = 0.22

sea 1 * 0.22 = 0.22

with 1 * 0.15 = 0.15

will 1 * 0.13 = 0.13

salton 1 * 0.07 = 0.07

, 1 * 0.03 = 0.03

probably 1 * 0.02 = 0.02

splatter 1 * 0 = 0

a 1 * -0.03 = -0.03

. 1 * -0.03 = -0.03

the 1 * -0.08 = -0.08

reasonably 1 * -0.1 = -0.1

have 1 * -0.19 = -0.19

then 1 * -0.23 = -0.23

Explanation: The predictor cannot judge if then condition, failed to give a good prediction.

Fix: Add if condition to the predictor in the context and train with dataset to correct the predictions.

=== while the ensemble player who gained notice in guy ritchie's lock , stock and two smoking barrels and snatch has the bod , he's unlikely to become a household name on the basis of his first starring vehicle .

Truth: -1, Prediction: 1 [WRONG]

vehicle 1 * 0.54 = 0.54

notice 1 * 0.47 = 0.47

first 1 * 0.35 = 0.35

has 1 * 0.28 = 0.28

and 2 * 0.1 = 0.2

he's 1 * 0.16 = 0.16

ensemble 1 * 0.13 = 0.13

of 1 * 0.11 = 0.11

who 1 * 0.1 = 0.1

, 2 * 0.03 = 0.06

while 1 * 0.04 = 0.04

unlikely 1 * 0.01 = 0.01

barrels 1 * 0 = 0

lock 1 * 0 = 0

household 1 * 0 = 0

ritchie's 1 * 0 = 0

snatch 1 * 0 = 0

smoking 1 * 0 = 0

gained 1 * 0 = 0

bod 1 * 0 = 0

two 1 * -0.02 = -0.02

a 1 * -0.03 = -0.03

. 1 * -0.03 = -0.03

become 1 * -0.03 = -0.03

on 1 * -0.05 = -0.05

in	1 * -0.05 = -0.05
to	1 * -0.11 = -0.11
basis	1 * -0.11 = -0.11
starring	1 * -0.12 = -0.12
his	1 * -0.15 = -0.15
stock	1 * -0.17 = -0.17
the	3 * -0.08 = -0.24
player	1 * -0.33 = -0.33
name	1 * -0.35 = -0.35
guy	1 * -0.49 = -0.49

Explanation: predictor did not learn enough new words to make a correct prediction

Fix: add new words to learn

3f. n gram extracts more words than the word feature extractor, therefore sometimes it's redundant.

But when n is small, it can help extract more words and uncommon words from the dataset to train and learn

Review:

=== a heady , biting , be-bop ride through nighttime manhattan , a loquacious videologue of the modern male and the lengths to which he'll go to weave a protective cocoon around his own ego .

Truth: 1, Prediction: -1 [WRONG]

Analyze: There are many words that words extractor can't predict, for example:

cocoon 1 * 0 = 0

videologue 1 * 0 = 0

loquacious 1 * 0 = 0

lengths 1 * 0 = 0

protective 1 * 0 = 0

weave 1 * 0 = 0

nighttime 1 * 0 = 0

be-bop 1 * 0 = 0

n gram extractor is better than the word extract since it can use smaller words like, logua, protect, night, coon to the learning process from the dataset to return positive feed back such as "protective" is positive, coon is positive as well as loquacious.

=== a heady , biting , be-bop ride through nighttime manhattan , a loquacious videologue of the modern male and the lengths to which he'll go to weave a protective cocoon around his own ego .

Truth: 1, Prediction: 1 [CORRECT]

tow 2 * 0.33 = 0.66

vea 1 * 0.44 = 0.44

eng 1 * 0.44 = 0.44

ide 2 * 0.22 = 0.44

tim 1 * 0.42 = 0.42

got 1 * 0.41 = 0.41

ive 1 * 0.41 = 0.41

4a

	$\mu_1=(2,3)$	$\mu_2=(2,-1)$
$\phi(x_1)=(1,0)$	10	2 ✓
$\phi(x_2)=(1,2)$	2 ✓	10
$\phi(x_3)=(3,0)$	10	2 ✓
$\phi(x_4)=(2,2)$	1 ✓	9

$z_1=2$, $z_2=1$, $z_3=2$, $z_4=1$

$\mu_2=(\frac{1+3}{2}, \frac{0}{2})=(2,0)$

$\mu_1=(\frac{1+2}{2}, 2)=(1.5,2)$

	$\mu_1=(1.5,2)$	$\mu_2=(2,0)$
$\phi(x_1)=(1,0)$	4.25	1
$\phi(x_2)=(1,2)$	0.25	5
$\phi(x_3)=(3,0)$	6.25	1
$\phi(x_4)=(2,2)$	0.25	4

$z_1=2$, $z_2=1$, $z_3=2$, $z_4=1$

$\mu_2=(2,0)$

$\mu_1=(1.5,2)$

4c. Algorithm to pre-assign the points in the same cluster

Step1. compute the centroids for each assigned each point globally to the particular cluster group.

Step2. Assigned points are omitted from calculation of the minimization among rest of the points.

Step3. Those assigned points are omitted when calculating the average for centroid.

Step4. Minimize the reconstruction loss of the assigned centroids.

4d. It can lead to a different cluster (could be better) if choosing different initializations.

Reason is that The K-means algorithms does not guarantee unique clustering, sometimes running the same K on the same dataset, the bad initialization can be trapped in a local minima instead of the global minimum (K-means is guaranteed to decrease the loss function each iteration and will converge to a local minimum)

How we initialize K-means will impact the final cluster. Such as K-means++ can place centroids on training points so that these centroids tend to be distant from one another. Or make initial center close enough to the final centers to make convergence faster and also help building good clusters.

4e. No, since we have random linear in number of data objects, the K-means won't be deterministic.