20180673 하재현

Problem1

a)

(+1) pretty good

(−1) bad plot

(−1) not good

(+1) pretty scenery

Weight Vecter: (pretty, good, bad, plot, not, scenery)

$$\nabla_{\mathrm{w}}\mathrm{Loss}_{\mathrm{hinge}}(x,y,w)=\begin{cases}0 \ \ if \ \ w\cdot\phi(x)y\geq 1\\ -\phi(x)y \ \ \text{otherwise}\end{cases}$$

(1)

$$\phi(x)=(1,1,0,0,0,0)$$
$$\phi(x)y=(1,1,0,0,0,0)$$
$$w\cdot\phi(x)y=0$$
$$\mathrm{Loss}_{\mathrm{hinge}}(x,y,w)=1$$
$$\nabla_{\mathrm{w}}\mathrm{Loss}_{\mathrm{hinge}}(x,y,w)=(-1,-1,0,0,0,0)$$
$$w=(1,1,0,0,0,0)$$

(2)

$$\phi(x)=(0,0,1,1,0,0)$$
$$\phi(x)y=(0,0,-1,-1,0,0)$$
$$w\cdot\phi(x)y=0$$
$$\mathrm{Loss}_{\mathrm{hinge}}(x,y,w)=1$$
$$\nabla_{\mathrm{w}}\mathrm{Loss}_{\mathrm{hinge}}(x,y,w)=(0,0,1,1,0,0)$$
$$w=(1,1,-1,-1,0,0)$$

(3)

$$\phi(x)=(0,1,0,0,1,0)$$

$$\phi(x)y = (0,-1,0,0,-1,0)$$

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

$$\mathrm{Loss}_{\mathrm{hinge}}(x,y,w) = 2$$

$$\nabla_{\mathrm{w}}\mathrm{Loss}_{\mathrm{hinge}}(x,y,w) = (0,1,0,0,1,0)$$

$$w = (1,0,-1,-1,-1,0)$$

(4)

$$\phi(x) = (1,0,0,0,0,1)$$

$$\phi(x)y = (1,0,0,0,0,1)$$

$$w \cdot \phi(x)y = 1$$

$$\mathrm{Loss}_{\mathrm{hinge}}(x,y,w) = 0$$

$$\nabla_{\mathrm{w}}\mathrm{Loss}_{\mathrm{hinge}}(x,y,w) = (0,0,0,0,0,0)$$

$$w = (1,0,-1,-1,-1,0)$$

$$\therefore w = (1,0,-1,-1,-1,0)$$

b)

Weight vector

$$w = (good, bad, not)$$

Dataset

+1	Good	(1,0,0)
−1	Bad	(0,1,0)
−1	Not good	(1,0,1)
+1	Not Bad	(0,1,1)

Proof

$w \cdot \phi(x)y \geq 1$ then zero error

$$w_1 \geq 1$$

$$w_2 \leq -1$$

$$w_1 + w_3 \leq -1$$

$$w_2 + w_3 \geq 1$$

From this, we can get

$$2 \leq w_3 \leq -2$$

Which is not feasible.

We can fix this by adding interaction feature like "not good"

Problem2

a)

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

b)

$$\begin{aligned}\nabla_{\text{w}}\text{Loss}(x,y,w) &= \nabla_{\text{w}}(y-p)^2 \\ &= 2(y-p) \cdot \ \nabla_{\text{w}}(y-p) \\ &= -2(y-p)\phi(x)e^{-w\phi(x)}\text{p}^2 \\ &= -2p(1-p)(y-p)\phi(x)\end{aligned}$$

c)

$$\nabla_{\text{w}}\text{Loss}(x,0,w) = 2p^2(1-p)\phi(x)$$

The gradient is nearly 0 when $\text{p} = 1$ or 0 which means

$$\text{w} \cdot \phi(x) \to +\infty \ \ or - \infty$$

$$\therefore \text{when } \ \text{w} \to +\infty \ \ or - \infty, ||\nabla_{\text{w}}\text{Loss}(x,0,w)|| \ \ is \ \ smallest$$

d)

Let differentiate by p

Then we get $\text{p}(2-3\text{p})$ which implies that when $\text{p} = \frac{2}{3}$, magnitude is largest

$$\therefore \ \text{largest magintude is } \frac{8}{27}||\phi(x)||$$

e)

$$\sigma\left(w \cdot \phi(x)\right) - y = 0$$

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

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

$$\ln\left(\frac{1}{\text{y}} - 1\right) = -w \cdot \phi(x)$$

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

Problem3

d)

(1)

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 * 0.3800000000000001 = 0.3800000000000001
real	1.0 * 0.3300000000000001 = 0.3300000000000001
start	1.0 * 0.2900000000000001 = 0.2900000000000001
kids	1.0 * 0.25000000000000006 = 0.25000000000000006
spielberg	1.0 * 0.25000000000000006 = 0.25000000000000006
know	1.0 * 0.22000000000000033 = 0.22000000000000033
home	1.0 * 0.22000000000000006 = 0.22000000000000006
steven	1.0 * 0.22000000000000006 = 0.22000000000000006
even	1.0 * 0.16999999999999998 = 0.16999999999999998
stunts	1.0 * 0.12999999999999998 = 0.12999999999999998
do	1.0 * 0.06000000000000001 = 0.06000000000000001
nice	1.0 * 0.060000000000000005 = 0.060000000000000005
pulling	1.0 * 0.05 = 0.05
until	1.0 * 0.020000000000000004 = 0.020000000000000004
,	2.0 * 0.00999999999999745 = 0.0199999999999949
a	1.0 * -0.00999999999999822 = -0.00999999999999822
.	2.0 * -0.009999999999997402 = -0.019999999999994803
producers	1.0 * -0.02 = -0.02
alone	1.0 * -0.05 = -0.05
would	1.0 * -0.07999999999999997 = -0.07999999999999997
this	1.0 * -0.08000000000000002 = -0.08000000000000002
the	1.0 * -0.08999999999999733 = -0.08999999999999733
aren't	1.0 * -0.09999999999999999 = -0.09999999999999999
how	1.0 * -0.1000000000000001 = -0.1000000000000001
not	1.0 * -0.1099999999999996 = -0.1099999999999996
to	1.0 * -0.12999999999999787 = -0.12999999999999787
movie	1.0 * -0.12999999999999984 = -0.12999999999999984

off 1.0 * -0.12999999999999998 = -0.12999999999999998

hollywood 1.0 * -0.18999999999999995 = -0.18999999999999995

premise 1.0 * -0.20000000000000004 = -0.20000000000000004

goes 1.0 * -0.24 = -0.24

besides 1.0 * -0.2700000000000001 = -0.2700000000000001

분석: Until을 기점으로 부정문으로 바뀌고 있는데, 이 분석 시스템은 그걸 캐치하지 못하고 각 단어만을 체크해서 긍정으로 판단한 것 같음.

해결: 특정 단어(Until 등)을 기점으로 감정을 끊어서 혹은 반대로 인식할 수 있어야 함

(2)

a perfectly competent and often imaginative film that lacks what little lilo & stitch had in spades -- charisma .

Truth: 1, Prediction: -1 [WRONG]

film 1.0 * 0.3400000000000001 = 0.3400000000000001

& 1.0 * 0.3200000000000001 = 0.3200000000000001

stitch 1.0 * 0.20000000000000004 = 0.20000000000000004

perfectly 1.0 * 0.18000000000000002 = 0.18000000000000002

lilo 1.0 * 0.15 = 0.15

charisma 1.0 * 0.12999999999999998 = 0.12999999999999998

often 1.0 * 0.10999999999999999 = 0.10999999999999999

-- 1.0 * 0.0999999999999997 = 0.0999999999999997

and 1.0 * 0.06000000000000038 = 0.06000000000000038

what 1.0 * 0.010000000000000018 = 0.010000000000000018

spades 1.0 * 0 = 0.0

. 1.0 * -0.009999999999997402 = -0.009999999999997402

a 1.0 * -0.00999999999999822 = -0.00999999999999822

in 1.0 * -0.07999999999999964 = -0.07999999999999964

imaginative 1.0 * -0.08 = -0.08

that 1.0 * −0.09000000000000029 = −0.09000000000000029

competent 1.0 * −0.16 = −0.16

little 1.0 * −0.32 = −0.32

had 1.0 * −0.3700000000000003 = −0.3700000000000003

lacks 1.0 * −0.5600000000000003 = −0.5600000000000003

분석: had 같은 중립 단어가 너무 큰 weight을 갖고 있음

해결: 중립 단어는 sentiment에 별 영향을 주지 않는다는 것을 알아야 함.

(3)

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 * 0.8900000000000006 = 0.8900000000000006

modern 1.0 * 0.20000000000000004 = 0.20000000000000004

ego 1.0 * 0.20000000000000004 = 0.20000000000000004

of 1.0 * 0.09999999999999928 = 0.09999999999999928

and 1.0 * 0.06000000000000038 = 0.06000000000000038

he'll 1.0 * 0.060000000000000005 = 0.060000000000000005

, 3.0 * 0.009999999999999745 = 0.029999999999992352

be−bop 1.0 * 0 = 0.0

nighttime 1.0 * 0 = 0.0

loquacious 1.0 * 0 = 0.0

videologue 1.0 * 0 = 0.0

lengths 1.0 * 0 = 0.0

weave 1.0 * 0 = 0.0

protective 1.0 * 0 = 0.0

cocoon 1.0 * 0 = 0.0

. 1.0 * −0.009999999999997402 = −0.009999999999997402

heady	1.0 * -0.02 = -0.02
a	3.0 * -0.00999999999999822 = -0.0299999999999994663
through	1.0 * -0.03999999999999997 = -0.03999999999999997
biting	1.0 * -0.04 = -0.04
own	1.0 * -0.09000000000000014 = -0.09000000000000014
manhattan	1.0 * -0.12999999999999998 = -0.12999999999999998
which	1.0 * -0.1399999999999997 = -0.1399999999999997
his	1.0 * -0.1699999999999995 = -0.1699999999999995
the	2.0 * -0.08999999999999733 = -0.17999999999999466
go	1.0 * -0.18000000000000002 = -0.18000000000000002
to	2.0 * -0.12999999999999787 = -0.25999999999999573
male	1.0 * -0.2900000000000001 = -0.2900000000000001
around	1.0 * -0.3700000000000002 = -0.3700000000000002

분석: to, the, male, around 같은 sentiment에 별 영향을 주지 않는, 중립에 가까운 단어들이 편향되었다.

해결: Classifier는 특정 단어가 한쪽으로 편향되었음을 알고, 이들의 편향성을 없애야 한다

(4)

'it's painful to watch witherspoon's talents wasting away inside unnecessary films like legally blonde and sweet home abomination , i mean , alabama . '

Truth: -1, Prediction: 1 [WRONG]

sweet	1.0 * 0.5600000000000003 = 0.5600000000000003
painful	1.0 * 0.2700000000000001 = 0.2700000000000001
home	1.0 * 0.22000000000000006 = 0.22000000000000006
films	1.0 * 0.17 = 0.17
inside	1.0 * 0.08 = 0.08
and	1.0 * 0.06000000000000038 = 0.06000000000000038
legally	1.0 * 0.060000000000000005 = 0.060000000000000005
blonde	1.0 * 0.060000000000000005 = 0.060000000000000005

mean	1.0 * 0.04000000000000001 = 0.04000000000000001
i	1.0 * 0.019999999999999983 = 0.019999999999999983
,	2.0 * 0.009999999999999745 = 0.01999999999999949
away	1.0 * 3.469446951953614e-18 = 3.469446951953614e-18
'it's	1.0 * 0 = 0.0
witherspoon's	1.0 * 0 = 0.0
wasting	1.0 * 0 = 0.0
abomination	1.0 * 0 = 0.0
alabama	1.0 * 0 = 0.0
.	1.0 * -0.009999999999997402 = -0.009999999999997402
talents	1.0 * -0.009999999999999997 = -0.009999999999999997
like	1.0 * -0.09999999999999985 = -0.09999999999999985
to	1.0 * -0.12999999999999787 = -0.12999999999999787
'	1.0 * -0.1500000000000007 = -0.1500000000000007
unnecessary	1.0 * -0.16 = -0.16
watch	1.0 * -0.16999999999999996 = -0.16999999999999996

분석: 이상하게도 painful이 긍정적 단어로 평가되고 있고, like 이하의 구문들은 예시를 소개하는 것이므로 영향을 미치면 안되지만 sweet가 큰 영향을 미치고 있다.

해결: Painful은 부정적 단어임을 알아야 하고, 특정 구문은 아예 영향이 없는 (ex: 예시 소개) 구문임을 알아야 한다.

(5)

dull , if not devoid of wit , this shaggy dog longs to frisk through the back alleys of history , but scarcely manages more than a modest , snoozy charm .

Truth: -1, Prediction: 1 [WRONG]

history	1.0 * 0.8000000000000005 = 0.8000000000000005
back	1.0 * 0.5300000000000004 = 0.5300000000000004
charm	1.0 * 0.4200000000000002 = 0.4200000000000002
manages	1.0 * 0.3200000000000001 = 0.3200000000000001

if	1.0 * 0.27999999999999947 = 0.27999999999999947
wit	1.0 * 0.2 = 0.2
of	2.0 * 0.09999999999999928 = 0.19999999999999857
shaggy	1.0 * 0.04000000000000001 = 0.04000000000000001
,	4.0 * 0.00999999999999745 = 0.0399999999999898
dog	1.0 * 0.01000000000000001 = 0.01000000000000001
longs	1.0 * 0 = 0.0
frisk	1.0 * 0 = 0.0
alleys	1.0 * 0 = 0.0
snoozy	1.0 * 0 = 0.0
.	1.0 * -0.009999999999997402 = -0.009999999999997402
a	1.0 * -0.00999999999999822 = -0.00999999999999822
than	1.0 * -0.009999999999999997 = -0.009999999999999997
through	1.0 * -0.03999999999999997 = -0.03999999999999997
but	1.0 * -0.07999999999999992 = -0.07999999999999992
this	1.0 * -0.08000000000000002 = -0.08000000000000002
the	1.0 * -0.08999999999999733 = -0.08999999999999733
not	1.0 * -0.1099999999999996 = -0.1099999999999996
modest	1.0 * -0.11999999999999998 = -0.11999999999999998
to	1.0 * -0.12999999999999787 = -0.12999999999999787
scarcely	1.0 * -0.16 = -0.16
more	1.0 * -0.19000000000000059 = -0.19000000000000059
devoid	1.0 * -0.6400000000000003 = -0.6400000000000003
dull	1.0 * -1.1000000000000008 = -1.1000000000000008

분석: history가 딱히 편향적인 단어가 아니지만, 가끔 등장하다보니 너무 한쪽으로 편향되어 있고, scarcely에 의해 뒤에 구문이 부정문 쪽으로 해석되어야 하지만 그렇지 않았다.

해결: history가 편향되지 않은 단어임을 알아야 하고, 특정 단어가 등장하면 그 이후의 구문이 의미가 바뀔 수 있다는 것을 알아야 한다.

(6)

wickedly funny , visually engrossing , never boring , this movie challenges us to think about the ways we consume pop culture .

Truth: 1, Prediction: -1 [WRONG]

culture	1.0 * 0.47000000000000025 = 0.47000000000000025
funny	1.0 * 0.3800000000000001 = 0.3800000000000001
ways	1.0 * 0.3300000000000001 = 0.3300000000000001
engrossing	1.0 * 0.22000000000000006 = 0.22000000000000006
us	1.0 * 0.21999999999999922 = 0.21999999999999922
wickedly	1.0 * 0.18000000000000002 = 0.18000000000000002
think	1.0 * 0.07 = 0.07
visually	1.0 * 0.04 = 0.04
we	1.0 * 0.02999999999999981 = 0.02999999999999981
,	3.0 * 0.00999999999999745 = 0.029999999999992352
challenges	1.0 * 0.019999999999999997 = 0.019999999999999997
consume	1.0 * 0 = 0.0
.	1.0 * -0.009999999999997402 = -0.009999999999997402
this	1.0 * -0.08000000000000002 = -0.08000000000000002
the	1.0 * -0.08999999999999733 = -0.08999999999999733
about	1.0 * -0.10000000000000069 = -0.10000000000000069
to	1.0 * -0.12999999999999787 = -0.12999999999999787
movie	1.0 * -0.12999999999999984 = -0.12999999999999984
pop	1.0 * -0.19000000000000003 = -0.19000000000000003
never	1.0 * -0.3900000000000001 = -0.3900000000000001
boring	1.0 * -1.1600000000000008 = -1.1600000000000008

분석: never boring 이라서 상당한 긍정이지만, 각각을 따로 해석해서 상당한 부정으로 해석해버렸다

해결: 단어를 조합함으로써 단어 각각의 sentiment와는 다른 sentiment로 바꿀 수 있음을 알아야 한다.

(7)

rain is a small treasure , enveloping the viewer in a literal and spiritual torpor that is anything but cathartic .

Truth: 1, Prediction: -1 [WRONG]

treasure	1.0 * 0.3100000000000001 = 0.3100000000000001
small	1.0 * 0.22000000000000006 = 0.22000000000000006
is	2.0 * 0.06000000000000011 = 0.12000000000000022
and	1.0 * 0.06000000000000038 = 0.06000000000000038
spiritual	1.0 * 0.05 = 0.05
enveloping	1.0 * 0.04 = 0.04
,	1.0 * 0.009999999999999745 = 0.009999999999999745
cathartic	1.0 * 0 = 0.0
.	1.0 * -0.009999999999997402 = -0.009999999999997402
a	2.0 * -0.00999999999999822 = -0.01999999999999644
rain	1.0 * -0.03 = -0.03
in	1.0 * -0.07999999999999964 = -0.07999999999999964
but	1.0 * -0.07999999999999992 = -0.07999999999999992
literal	1.0 * -0.08 = -0.08
the	1.0 * -0.08999999999999733 = -0.08999999999999733
that	1.0 * -0.09000000000000029 = -0.09000000000000029
viewer	1.0 * -0.12999999999999998 = -0.12999999999999998
anything	1.0 * -0.18 = -0.18
torpor	1.0 * -0.20000000000000004 = -0.20000000000000004

분석: that, the 같은 중립 단어가 편향되어 있음. 다만 그렇게 편향 정도가 크지 않고, is와 and도 똑같이 편향되어 있고 또 positive value와 negative value가 큰 차이가 없는 것으로 미루어 보아 조금씩 weight가 잘못 매겨진 것이 합쳐져서 생긴 문제로 추측된다.

해결: 중립 단어가 중립임을 알고, 편향 단어가 좀 더 편향되어있음을 알아야 한다.

(8)

patchy combination of soap opera , low-tech magic realism and , at times , ploddingly sociological commentary .

Truth: -1, Prediction: 1 [WRONG]

magic	1.0 * 0.5100000000000002 = 0.5100000000000002
realism	1.0 * 0.4200000000000002 = 0.4200000000000002
times	1.0 * 0.18 = 0.18
of	1.0 * 0.09999999999999928 = 0.09999999999999928
and	1.0 * 0.06000000000000038 = 0.06000000000000038
at	1.0 * 0.039999999999999945 = 0.039999999999999945
,	3.0 * 0.009999999999999745 = 0.029999999999992352
patchy	1.0 * 0 = 0.0
low-tech	1.0 * 0 = 0.0
ploddingly	1.0 * 0 = 0.0
sociological	1.0 * 0 = 0.0
.	1.0 * -0.009999999999997402 = -0.009999999999997402
combination	1.0 * -0.11999999999999998 = -0.11999999999999998
opera	1.0 * -0.2700000000000001 = -0.2700000000000001
commentary	1.0 * -0.2700000000000001 = -0.2700000000000001
soap	1.0 * -0.4100000000000002 = -0.4100000000000002

분석: low-tech와 magic realism 간의 관계를 분석하지 못해 생긴 에러

해결: 단어 간의 관계가 전체 문장의 sentiment에 영향을 미침을 알아야 함

(9)

the best thing i can say about this film is that i can't wait to see what the director does next .

Truth: 1, Prediction: -1 [WRONG]

does	1.0 * 0.6200000000000003 = 0.6200000000000003
best	1.0 * 0.44000000000000017 = 0.44000000000000017

film 1.0 * 0.3400000000000001 = 0.3400000000000001

wait 1.0 * 0.11999999999999998 = 0.11999999999999998

see 1.0 * 0.08 = 0.08

is 1.0 * 0.06000000000000011 = 0.06000000000000011

i 2.0 * 0.019999999999999983 = 0.039999999999999966

what 1.0 * 0.010000000000000018 = 0.010000000000000018

. 1.0 * -0.009999999999997402 = -0.009999999999997402

say 1.0 * -0.07000000000000009 = -0.07000000000000009

this 1.0 * -0.08000000000000002 = -0.08000000000000002

director 1.0 * -0.09 = -0.09

that 1.0 * -0.09000000000000029 = -0.09000000000000029

about 1.0 * -0.10000000000000069 = -0.10000000000000069

can 1.0 * -0.10999999999999999 = -0.10999999999999999

to 1.0 * -0.12999999999999787 = -0.12999999999999787

the 2.0 * -0.08999999999999733 = -0.17999999999999466

thing 1.0 * -0.22999999999999998 = -0.22999999999999998

next 1.0 * -0.29000000000000015 = -0.29000000000000015

can't 1.0 * -0.46000000000000024 = -0.46000000000000024

분석: 문장에서 can't 는 긍정의 의미로 쓰였으나 부정의 의미로 인식해버려 문제가 생겼다

해결: 각 단어가 문장 전체의 맥락에서 어떤 의미로 쓰이는지 알아야 함.

(10)

only in its final surprising shots does rabbit-proof fence find the authority it's looking for .

Truth: -1, Prediction: 1 [WRONG]

does 1.0 * 0.6200000000000003 = 0.6200000000000003

authority 1.0 * 0.21000000000000005 = 0.21000000000000005

it's 1.0 * 0.1900000000000009 = 0.1900000000000009

surprising 1.0 * 0.19000000000000003 = 0.19000000000000003 fence 1.0 * 0.19000000000000003 = 0.19000000000000003

rabbit-proof 1.0 * 0.18000000000000002 = 0.18000000000000002

looking 1.0 * 0.13999999999999999 = 0.13999999999999999

for 1.0 * 1.5612511283791264e-16 = 1.5612511283791264e-16

. 1.0 * -0.009999999999997402 = -0.009999999999997402

in 1.0 * -0.07999999999999964 = -0.07999999999999964

the 1.0 * -0.08999999999999733 = -0.08999999999999733

its 1.0 * -0.08999999999999979 = -0.08999999999999979

shots 1.0 * -0.12999999999999998 = -0.12999999999999998

final 1.0 * -0.15999999999999992 = -0.15999999999999992

find 1.0 * -0.5200000000000002 = -0.5200000000000002

only 1.0 * -0.5900000000000003 = -0.5900000000000003

분석: does 같은 중립적 의미의 단어가 너무 높은 weight를 가지고 있음

해결: 중립 단어는 sentiment에 영향을 별로 미치지 않음을 알아야 함.

f)

n을 1부터 10까지 해서 돌려본 결과

0.507

0.474

0.421

0.319

0.283

0.272

0.273

0.273

0.294

0.310

로 6~8이 가장 작게 나왔다.

이 범위가 가장 작게 나온 이유는 6~80이 한 단어를 포함하기 적당한 크기이며, 동시에 연관성이 있는 두 단어를 적절히 묶을 수 있는 길이이며, at the 같은 작은 단어들을 묶어서 처리하기 적절한 정도의 길이기 때문이다.

Problem4

a)

(1)

$\mu_1 \rightarrow\ \mathrm{x}_1, x_2$

$\mu_2 \rightarrow\ \mathrm{x}_3, x_4$

$\mu_1 = (0{,}0.5)$

$\mu_2 = (2{,}1)$

$\mu_1 \rightarrow\ \mathrm{x}_1, x_2$

$\mu_2 \rightarrow\ \mathrm{x}_3, x_4$

$\mu_1 = (0{,}0.5)$

$\mu_2 = (2{,}1)$

Converge.

(2)

$\mu_1 \rightarrow\ \mathrm{x}_1, x_3$

$\mu_2 \rightarrow\ \mathrm{x}_2, x_4$

$\mu_1 = (1{,}0)$

$\mu_2 = (1{,}1.5)$

$\mu_1 \rightarrow\ \mathrm{x}_1, x_3$

$\mu_2 \rightarrow\ \mathrm{x}_2, x_4$

$\mu_1 = (1{,}0)$

$\mu_2 = (1{,}1.5)$

Converge

c)

원래의 알고리즘을 수정하여 제약을 추가하면 된다.

Algorithm:

우선 S를 순회하며 같은 cluster에 들어가야 하는 point들의 집합을 구해준다.

예를 들어, $S = \{(1,2),(1,4),(3,5)\}$ 라면 $\{(1,2,4),(3,5)\}$를 만든다.

기존의 kmeans 알고리즘은 두 단계로 나눌 수 있다. (PPT 참조)

Phase 1. Update Assignments

: For each point, re-assign to closest means

$$a_i = \underset{K}{argmin}\, dist(x_i, c_k)$$

Phase 2. Update Means

$$c_k = \frac{1}{|\{i: a_i = k\}|} \sum_{i:a_i=k} x\_i:$$

이를 약간 수정하여, 제약을 만족시키는 cluster에 할당하도록 바꾼다.

Phase 1. Update Assignments

: For each point, re-assign to closest means without violating constraints

If x_i is in S

S_{x_i} = Set x_i belongs to

If S_{x_i} not null and NOT_ASSIGNED

Let $\{x_{j1}, x_{j2}, \cdots, x_{jn}\} = S_{x_i}$

$a_{group} = \underset{k}{argmin}\left(dist(x_{j1}, c_k) + dist(x_{j2}, c_k) + \cdots + dist(x_{jn}, c_k)\right)$

For x_{ji} in S_{x_i}

$a_{x_{ji}} = a_{group}$

Mark S_{x_i} as ALREADY_ASSIGNED

Else

$a_i = \underset{k}{argmin}\, dist(x_i, c_k)$

Phase 2. Update Means

$$c_k = \frac{1}{|\{i: a_i = k\}|} \sum_{i:a_i=k} x\_i:$$

For all $S_i \in S$, Mark S_{x_i} as NOT_ASSIGNED

Correctness:

위 알고리즘의 핵심은 set의 member들을 모두 고려해서 가장 작은 거리가 될 수 있는 point에 할당하는 것이다.

NOT_ASSIGNED / ALREADY_ASSIGNED는 각 set 당 한번씩 계산하기 위한 장치이다.

k-means algorithm의 정당성 증명의 핵심은 각 phase가 total distance를 줄이는지 증명하는 것이다.

Phase 1은 argmin에 의해서 오로지 total distance를 줄이는 방향으로만 할당 가능하다.

Phase 2는 주어진 집합의 total distance를 줄이는 점이 점들의 평균이므로, total distance가 절대 늘어나진 않는다.

따라서 위의 contrained k-means algorithm은 converge한다.