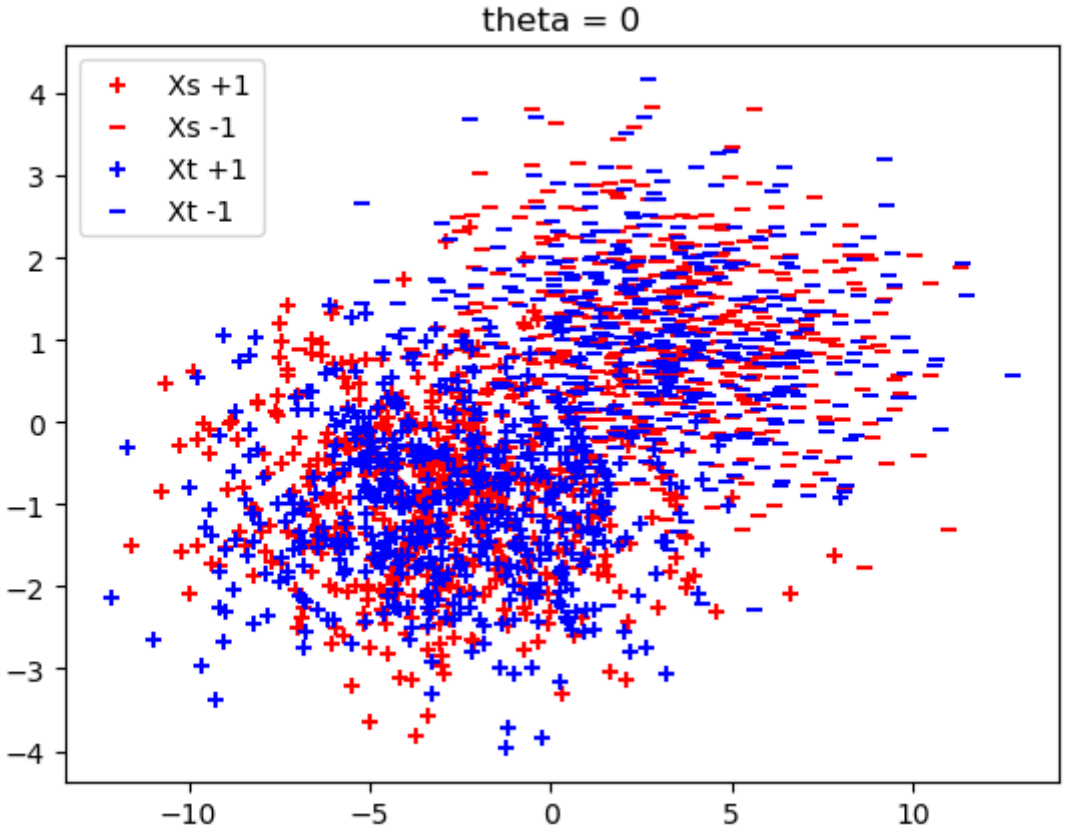
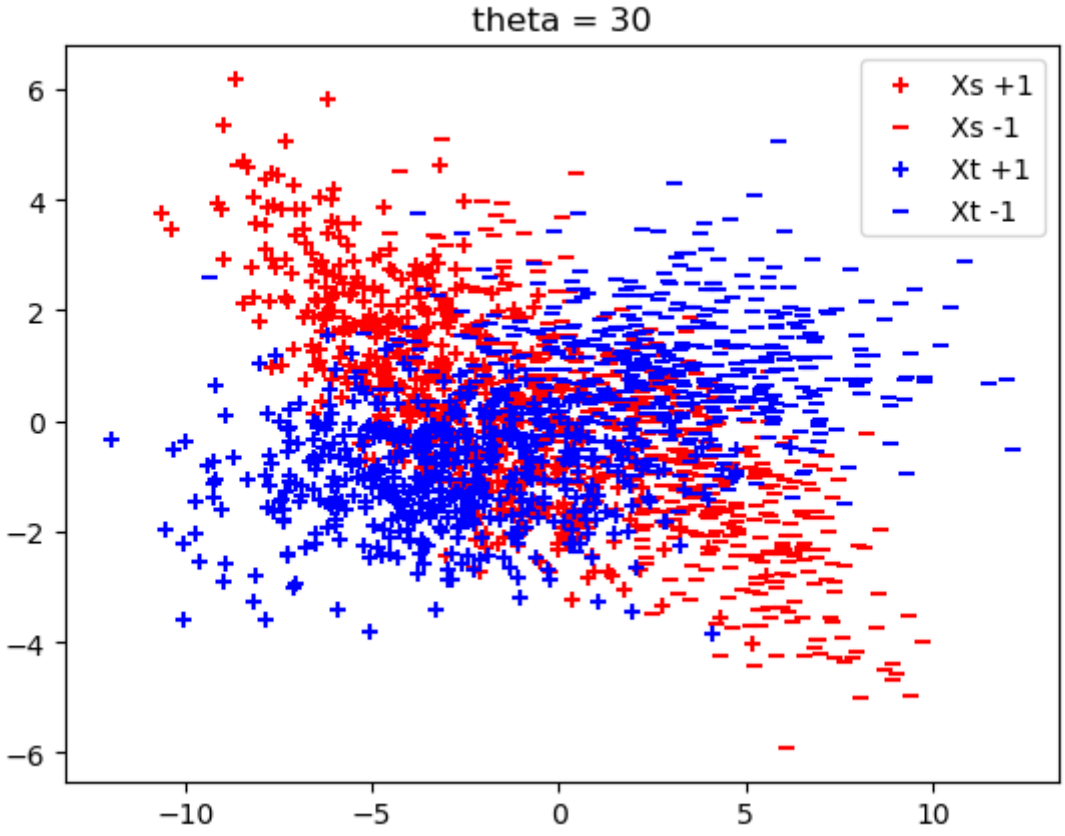
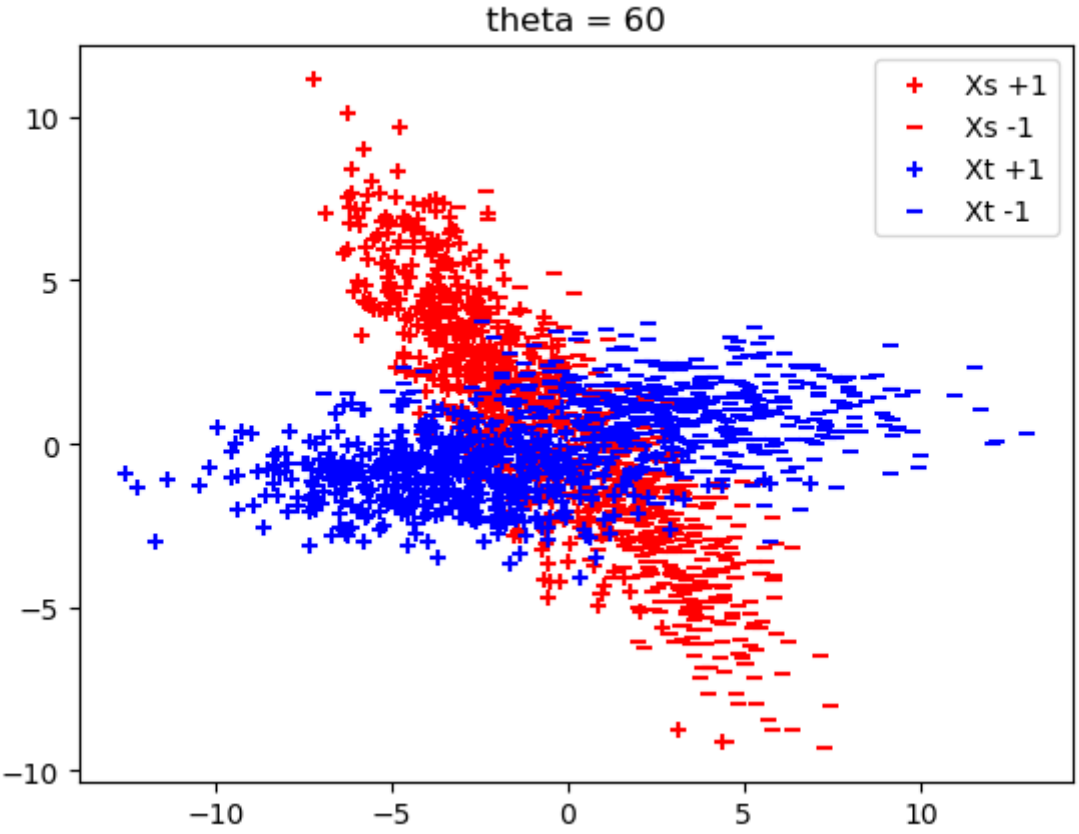
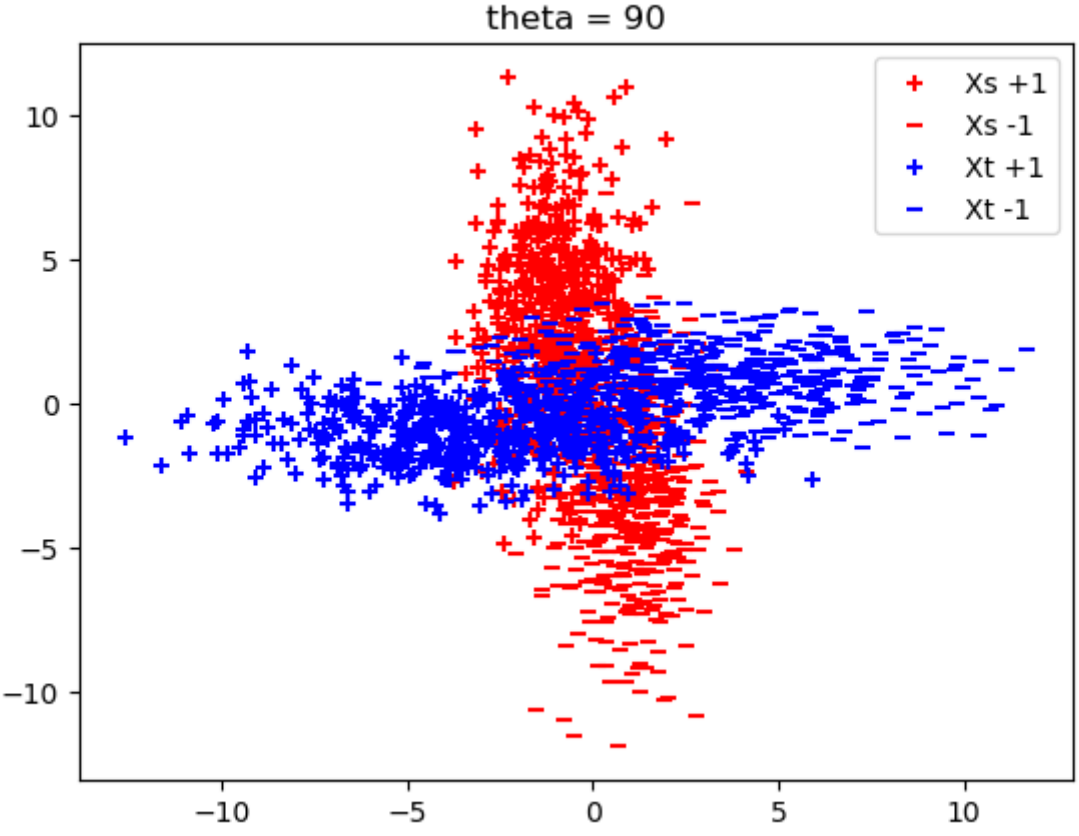
1

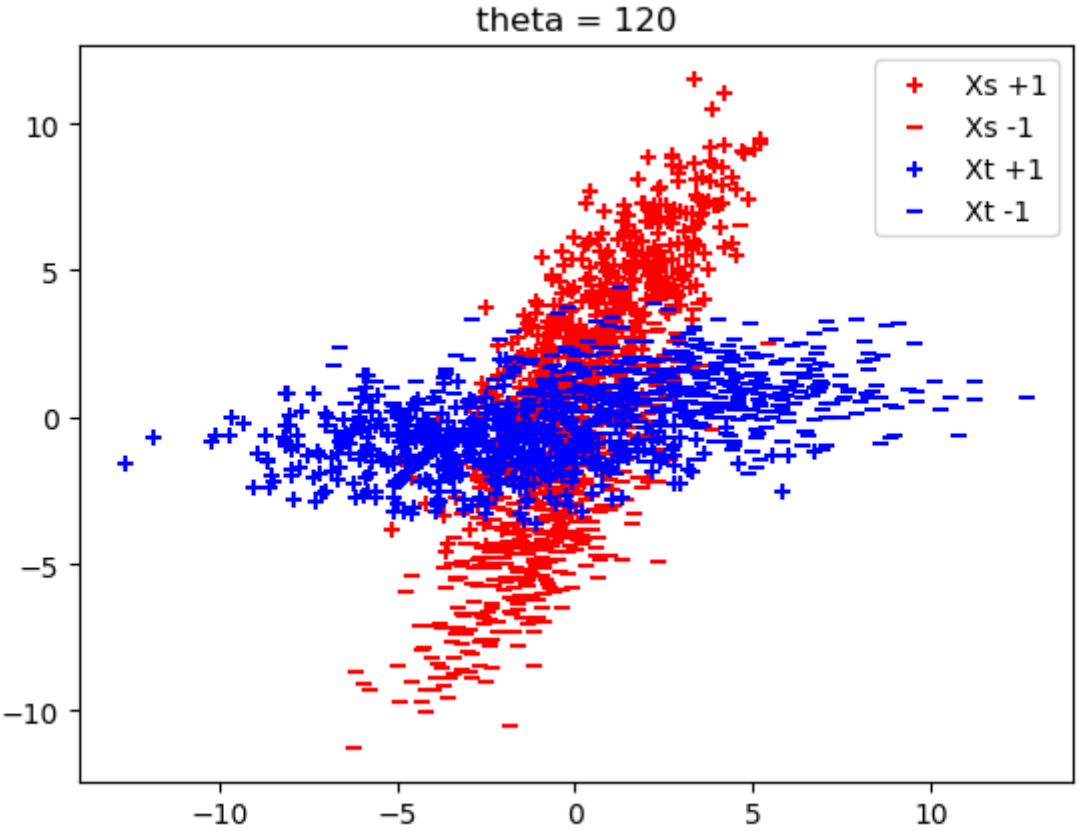
(a)

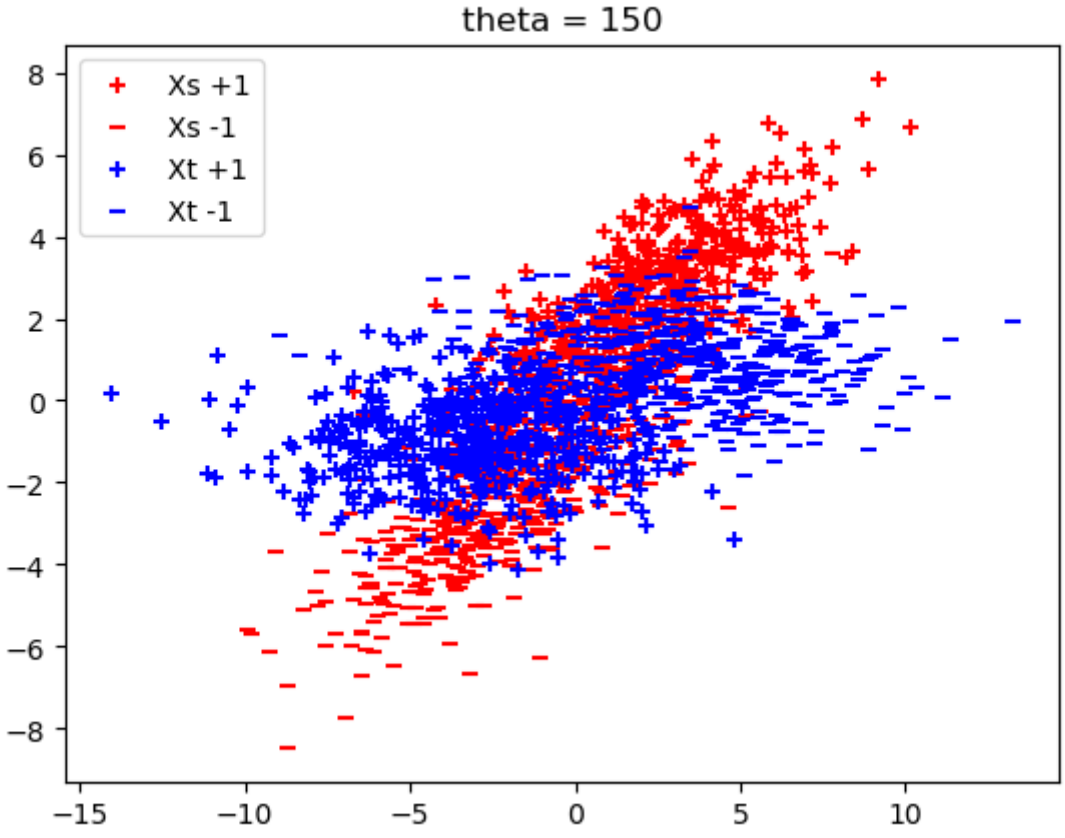

theta = 0
Xs +1
Xs -1
Xt +1
Xt -1
4
3
2
1
0
−1
−2
−3
−4
−10
−5
0
5
10

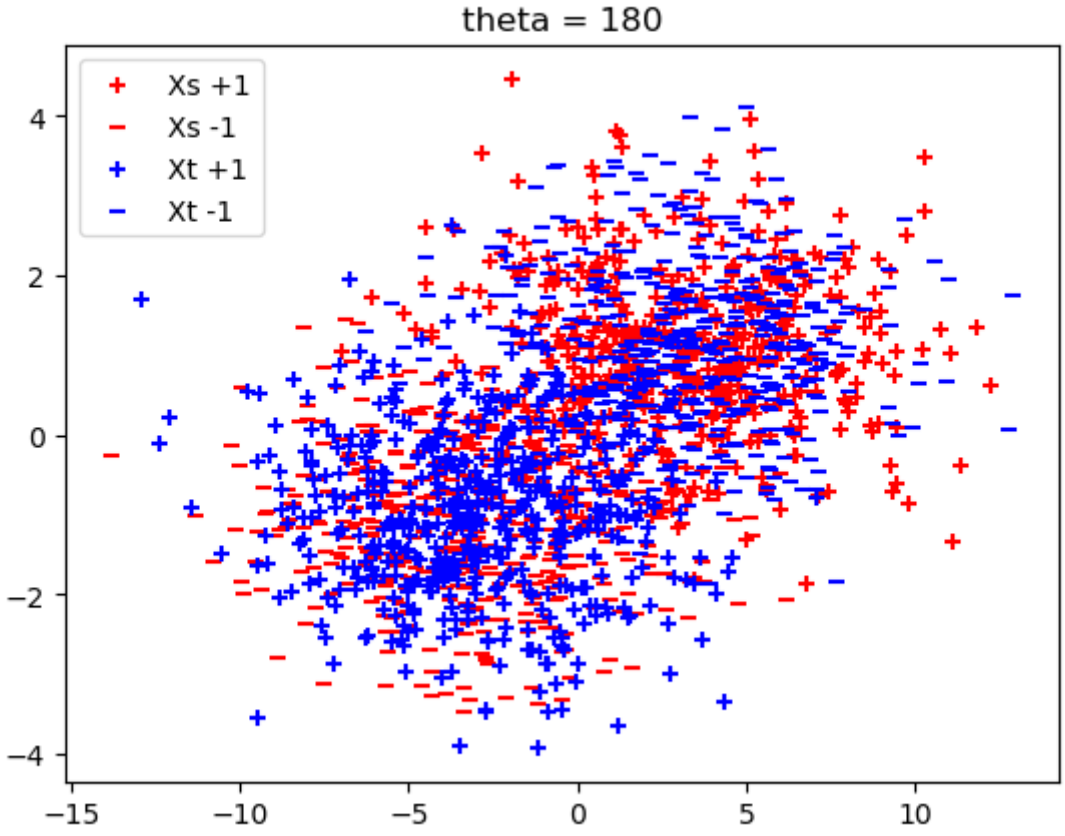

theta = 30
Xs +1
Xs -1
Xt +1
Xt -1
6
4
2
0
−2
−4
−6
−10
−5
0
5
10


theta = 60
Xs +1
Xs -1
Xt +1
Xt -1
10
5
0
−5
−10
−10
−5
0
5
10

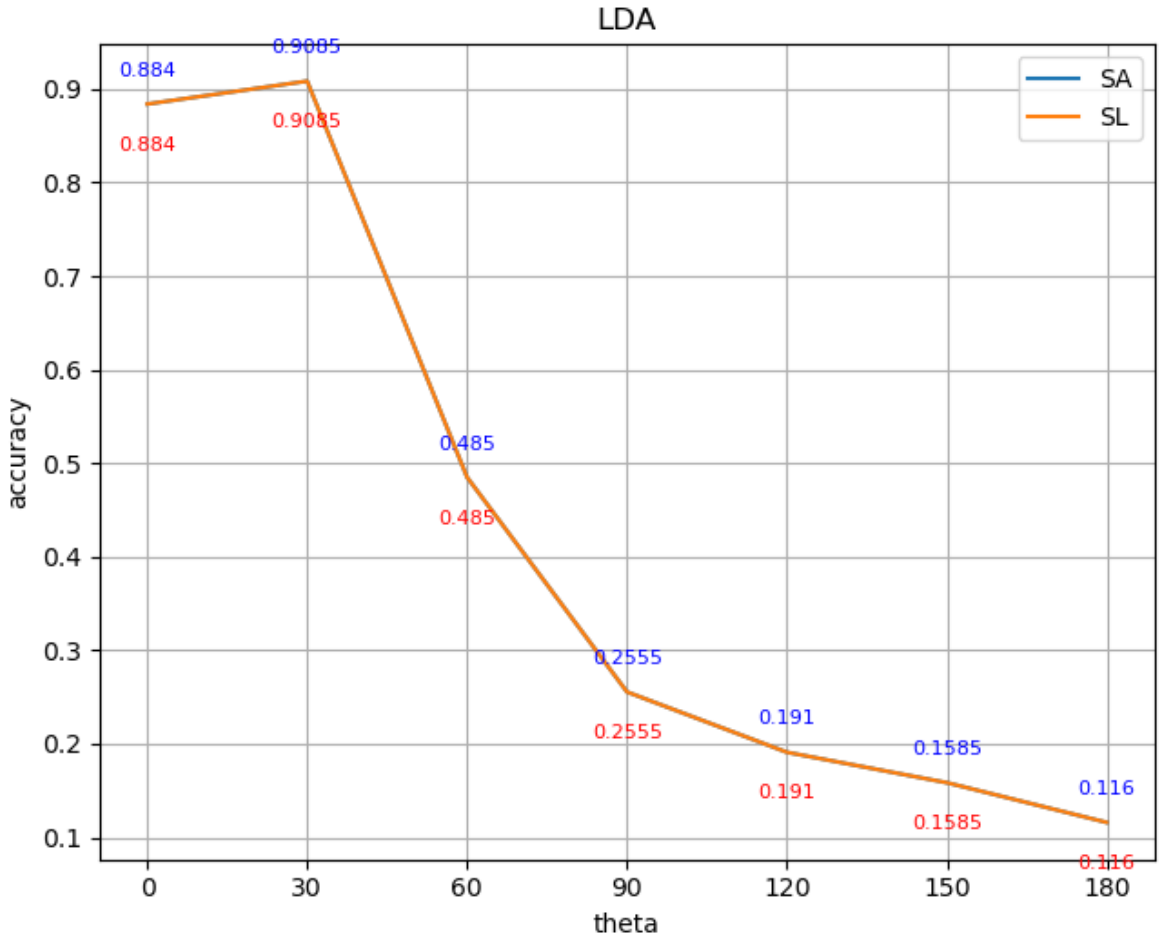

theta = 90
Xs +1
Xs -1
Xt +1
Xt -1
10
5
0
−5
−10
−10
−5
0
5
10

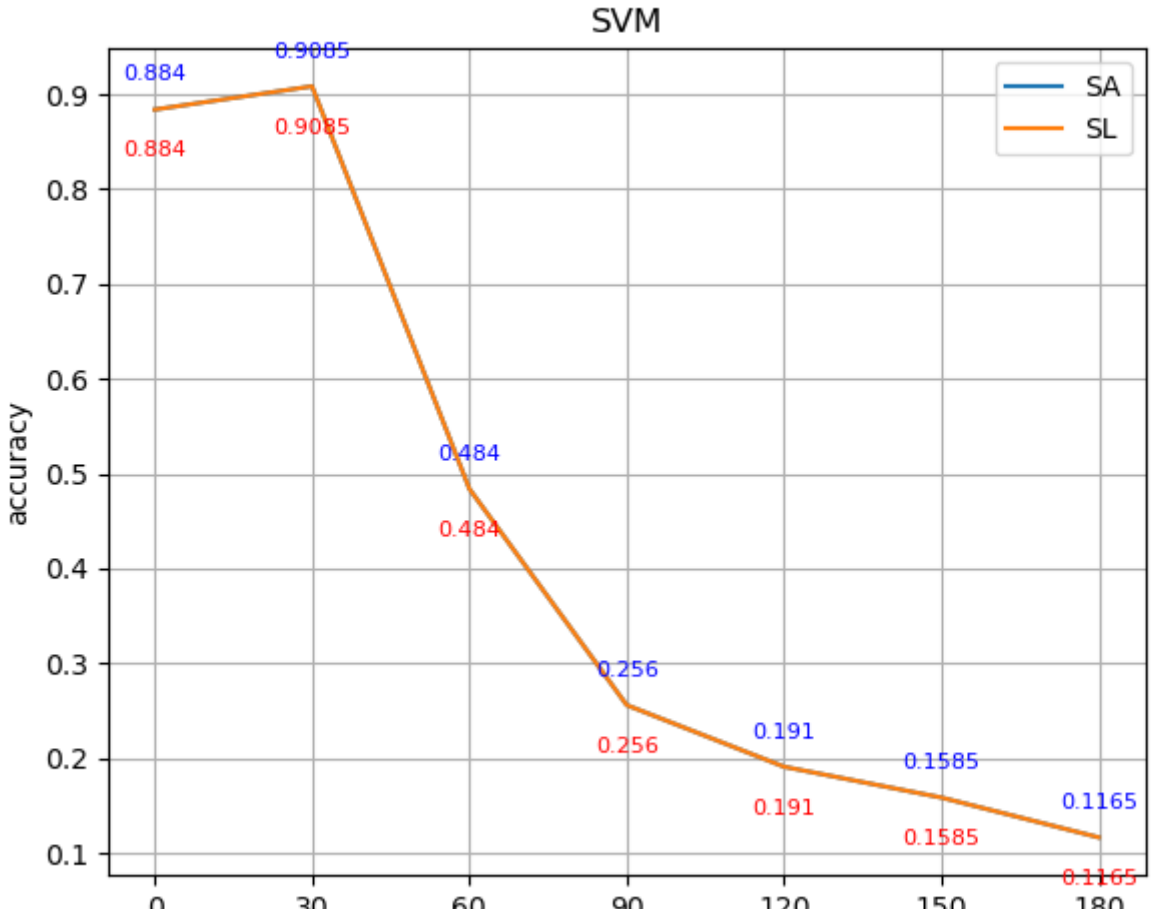

theta = 120
Xs +1
Xs -1
Xt +1
Xt -1
10
5
0
−5
−10
−10
−5
0
5
10

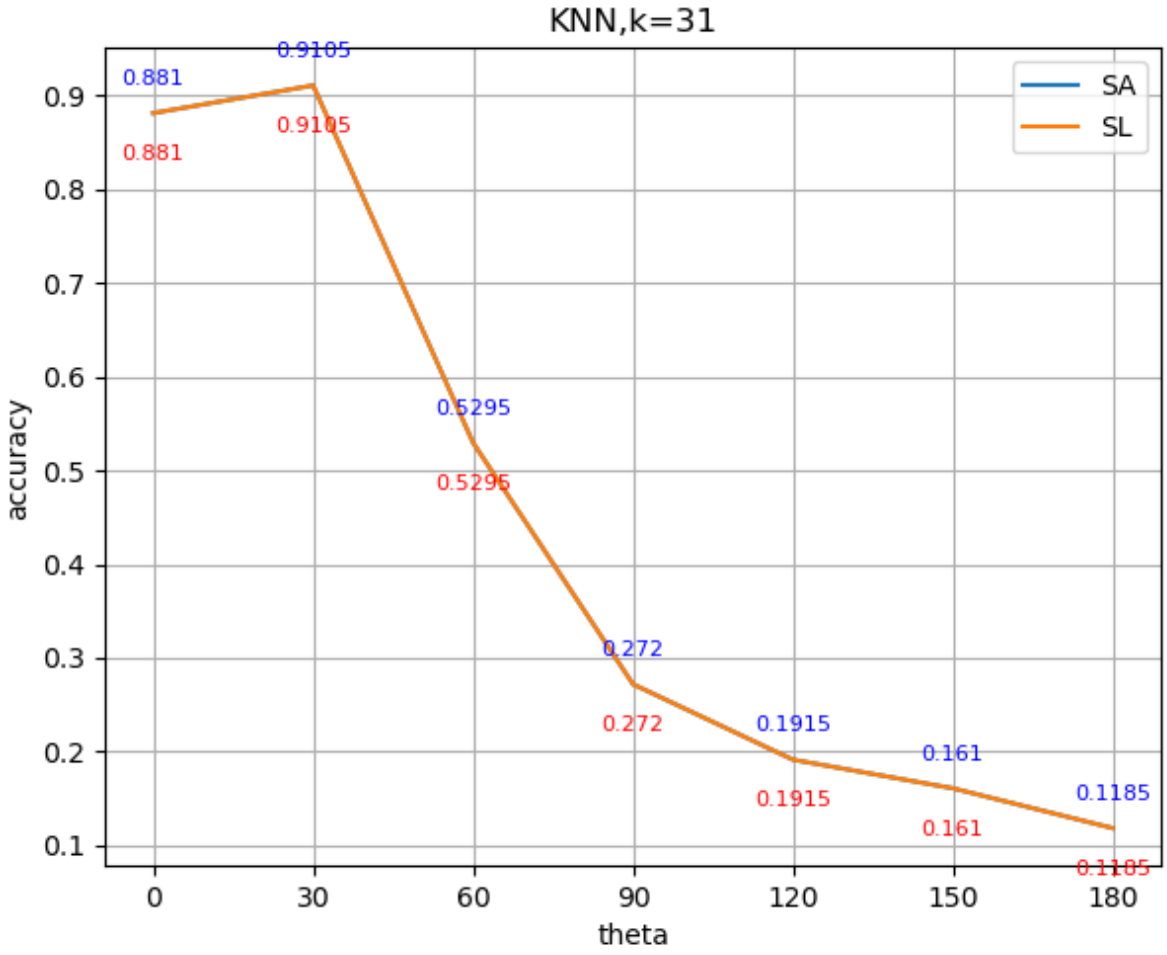

theta = 150
Xs +1
Xs -1
Xt +1
Xt -1
8
6
4
2
0
−2
−4
−6
−8
−15
−10
−5
0
5
10



(b)

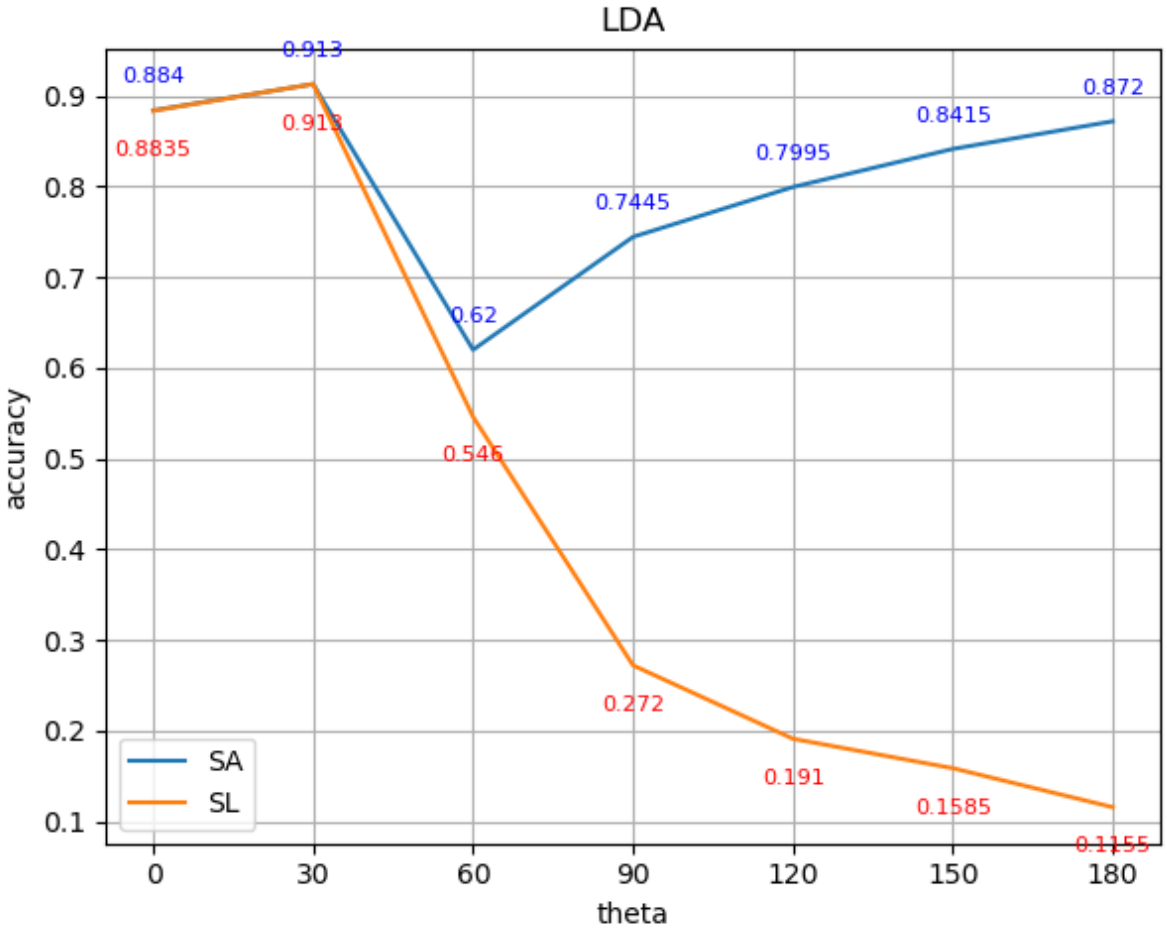

LDA
SA
SL
accuracy
theta
0.884
0.9085
0.485
0.2555
0.191
0.1585
0.116

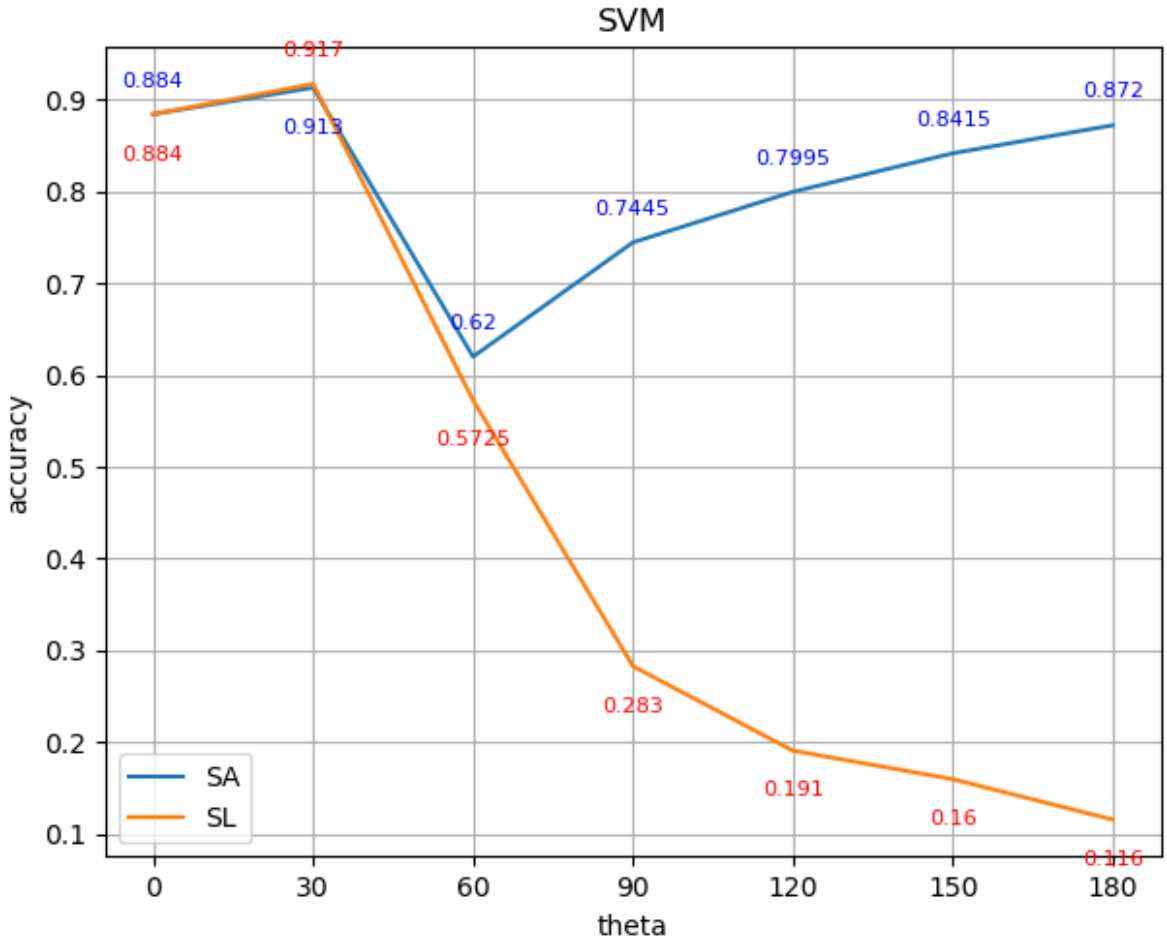

SVM
SA
SL
accuracy
0.884
0.9085
0.484
0.256
0.191
0.1585
0.1165

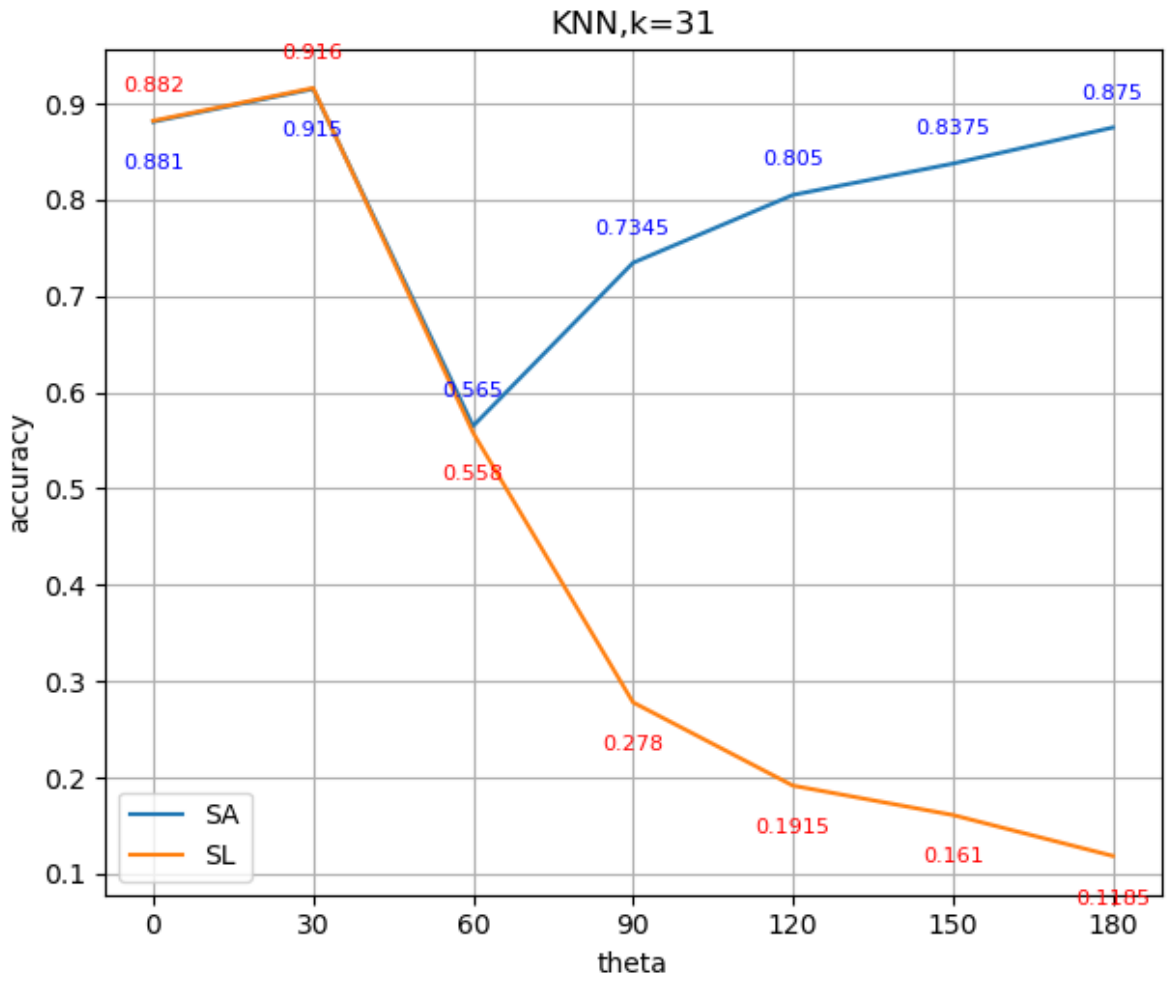


They perform the same.

(c)


LDA
0.884
0.913
0.872
0.8415
0.8835
0.913
0.7995
0.7445
0.62
0.545
0.272
0.191
0.1585
0.1155
SA
SL
accuracy
theta
0
30
60
90
120
150
180
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

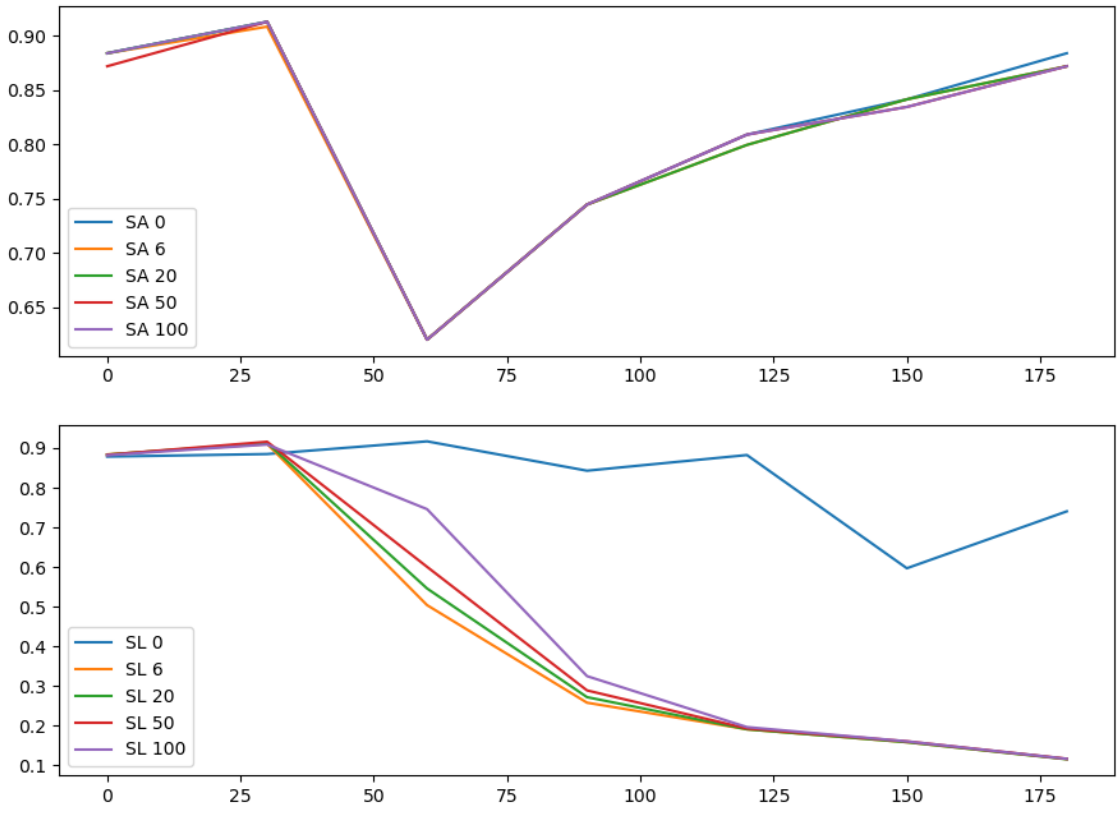

SVM
0.917
0.884
0.872
0.913
0.8415
0.884
0.7995
0.7445
0.62
0.5725
0.283
0.191
0.16
0.116
SA
SL
accuracy
theta
0
30
60
90
120
150
180
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9



SA performs better.

Sign flipping helps improving performance of SA.

(d)



N_TL has positive effects on SA with theta is large. Maybe because target domain data can remove the ambiguity of sign of eigenvectors.

N_TL has negative effects on SL. Maybe because target domain data has no relations to source domain data and mislead classifier.

(d)

sign flipping	standardization	SA	SL
True	False	0.7685	0.6105
True	True	0.913	0.913
False	False	0.561	0.56
False	True	0.9085	0.9085

Maybe standardization makes classifier easier to find optimal points and less affected by those relative large numbers.

2

(a)

N_T	N_S	alpha	epsilon
1	100	0.1	5.86
1	100	0.5	21.62
1	100	0.9	38.64

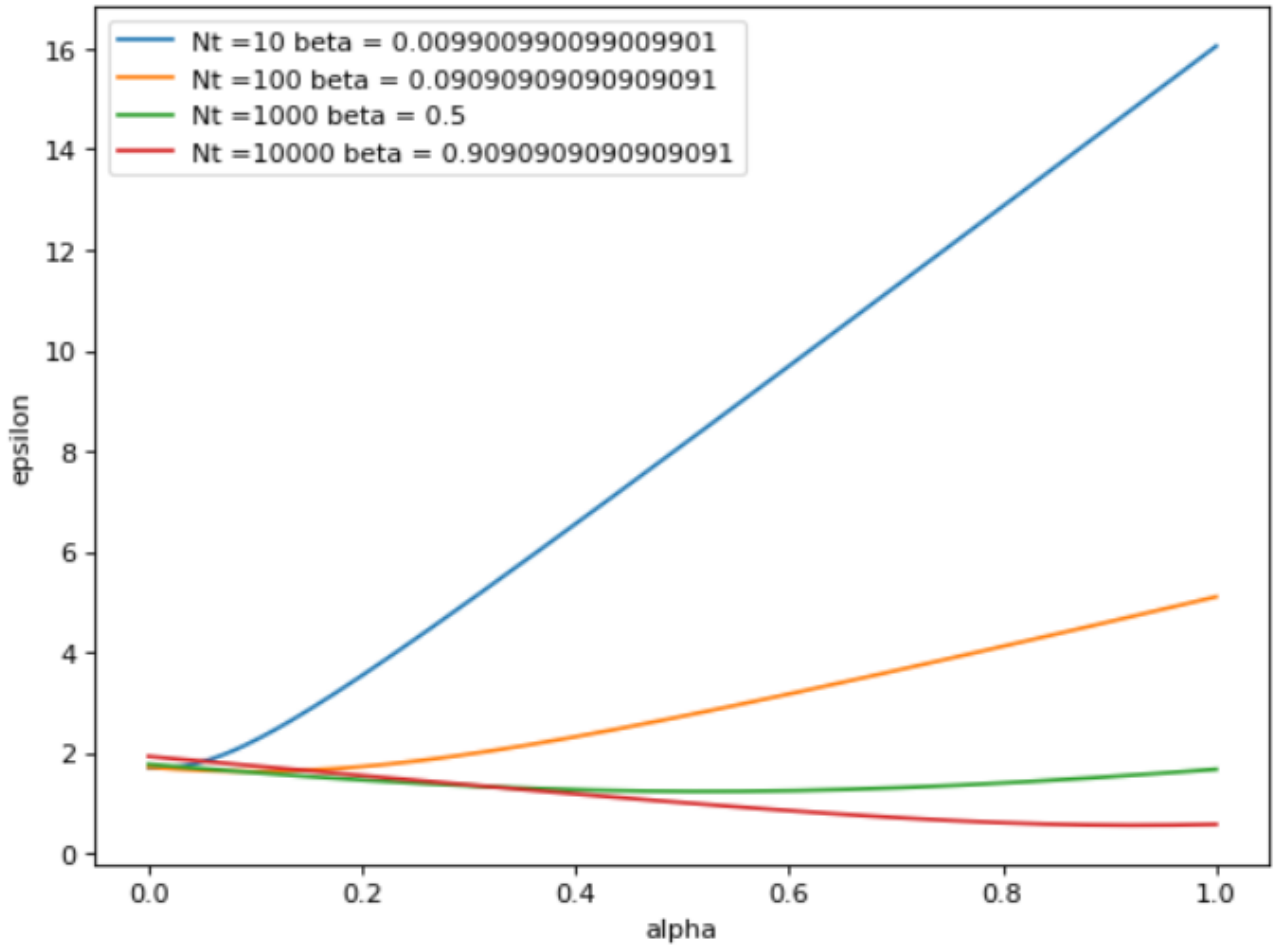
10	1000	0.1	2.25
10	1000	0.5	8.12
10	1000	0.9	14.46

100	10000	0.1	0.86
100	10000	0.5	2.94
100	10000	0.9	5.19

1000	100000	0.1	0.36
1000	100000	0.5	1.06
1000	100000	0.9	1.82

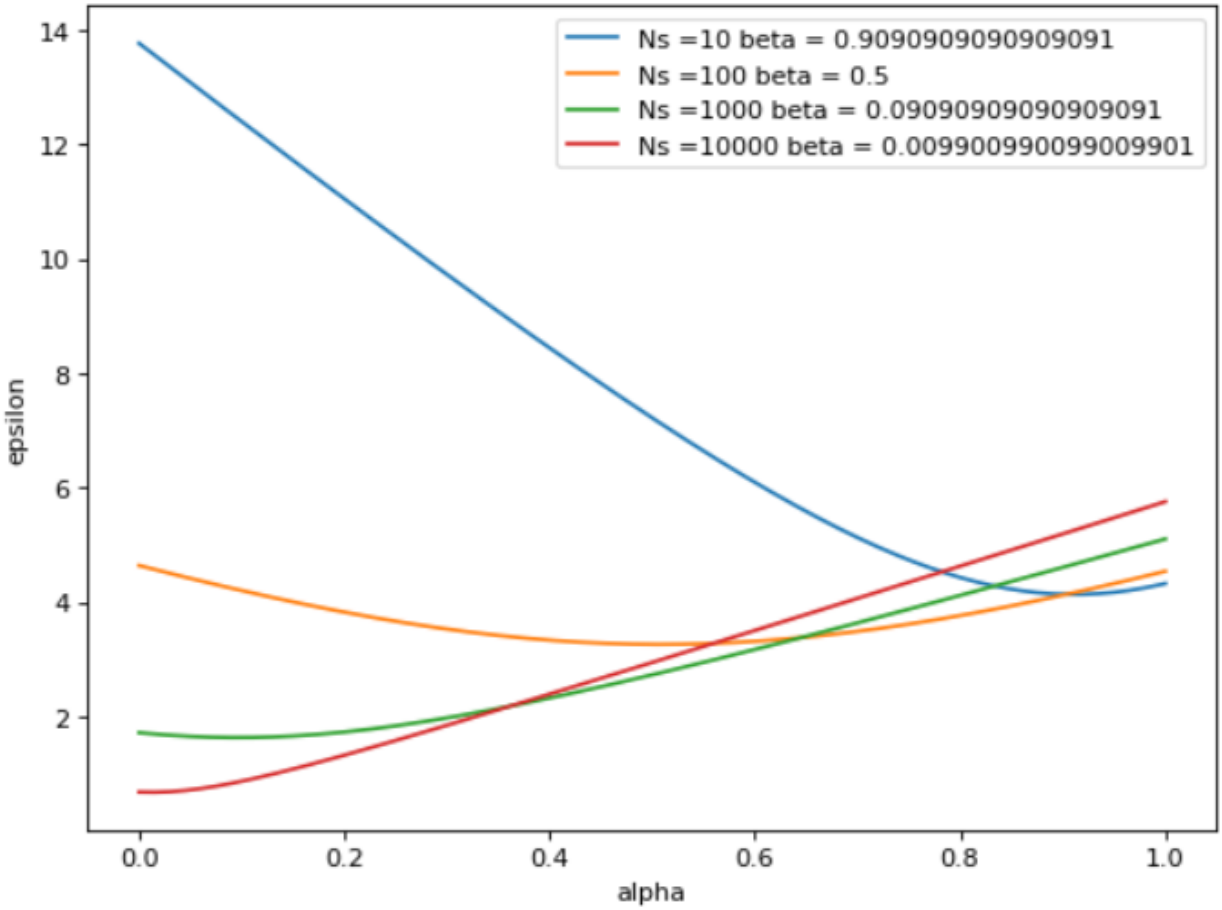
No, it seems that epsilon does not converge.

(b)



As alpha increases, the epsilon goes down first and then goes up. As N_T goes larger, the optimal value of alpha becomes larger.

(c)

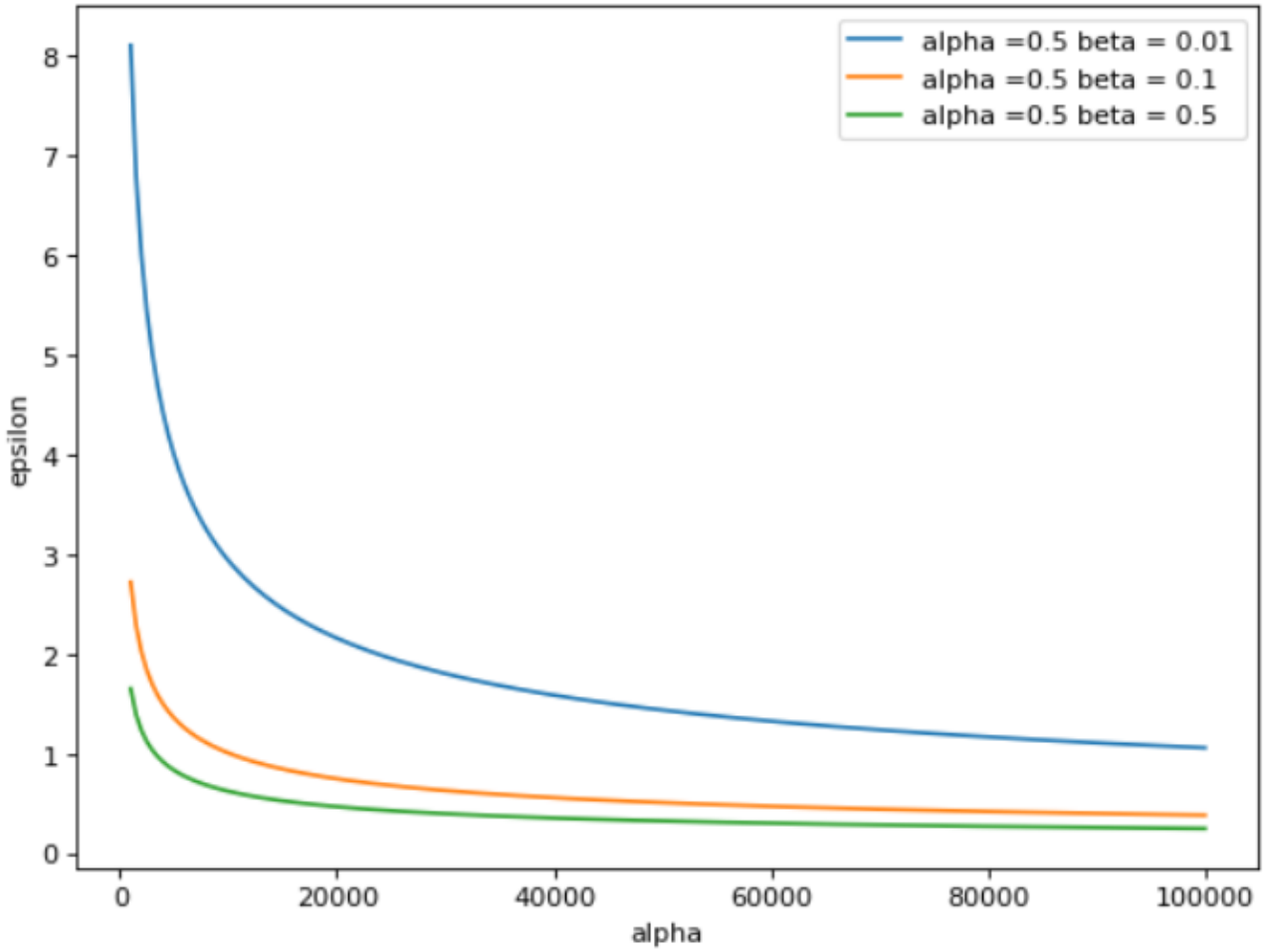


As alpha increases, the epsilon goes down first and then goes up. As N_S goes larger, the optimal value of alpha becomes smaller.

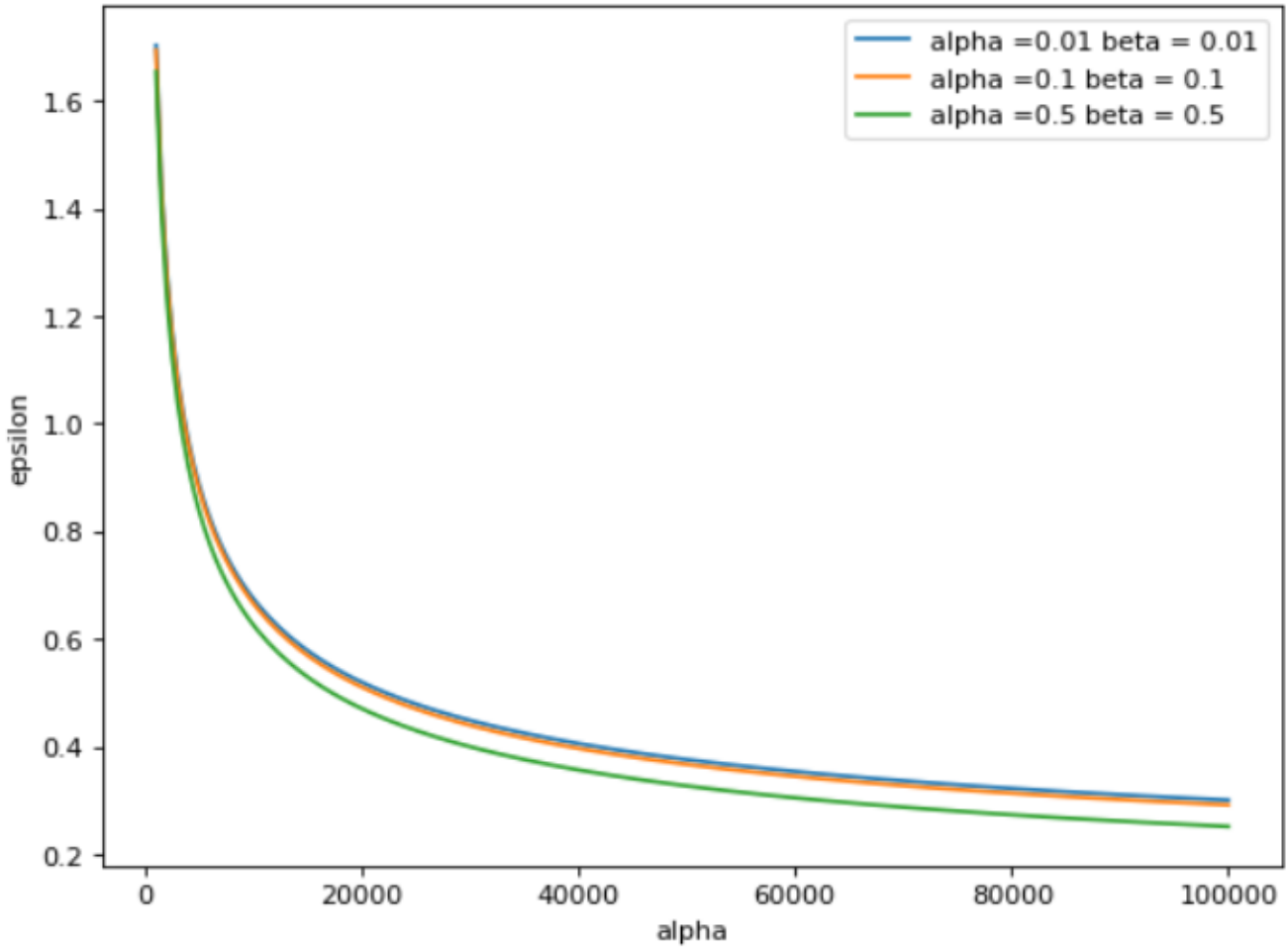
(d)

(iii)

alpha = 0.5


alpha =0.5 beta = 0.01
alpha =0.5 beta = 0.1
alpha =0.5 beta = 0.5
epsilon
alpha
0
1
2
3
4
5
6
7
8
0
20000
40000
60000
80000
100000

alpha = beta



alpha = 0.5 can give a better result with less data points. With enough data points, alpha = beta performs better.

(e)

No, it just depends on alpha beta and N.