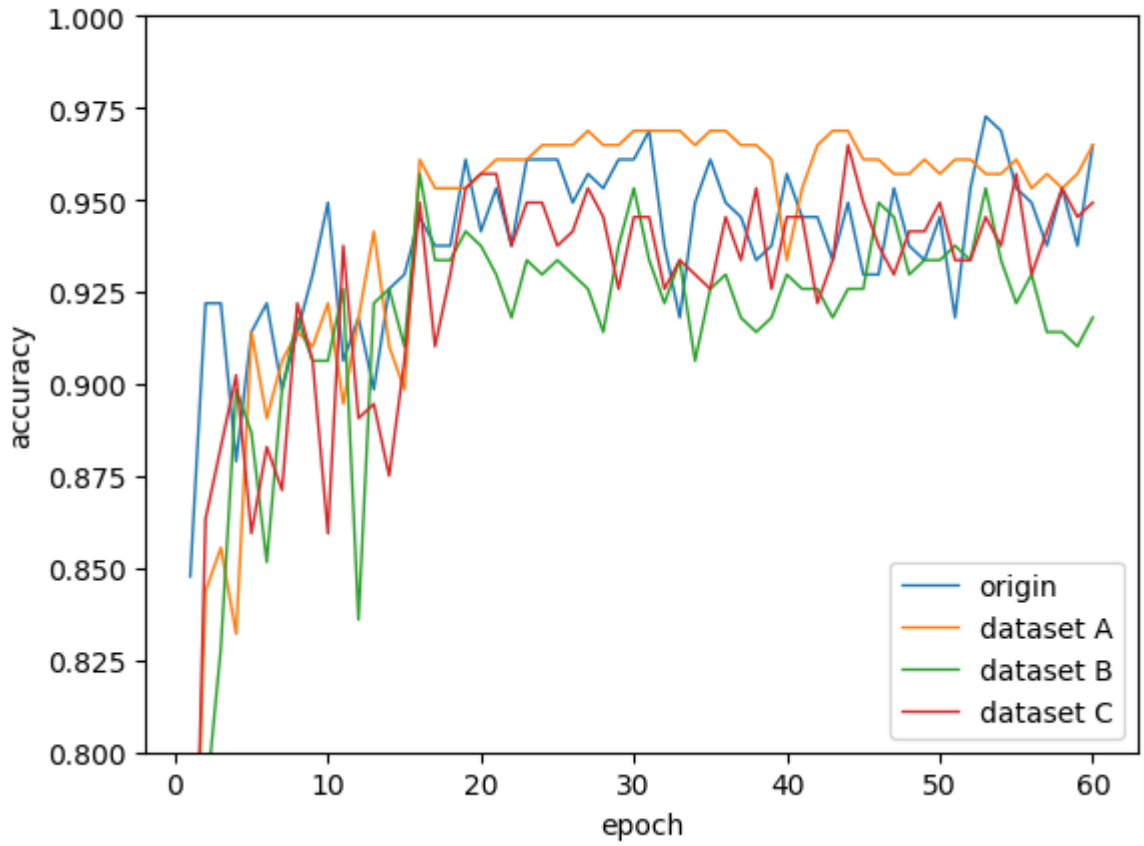
01-hw

liangjiadong 1500010692

question 1

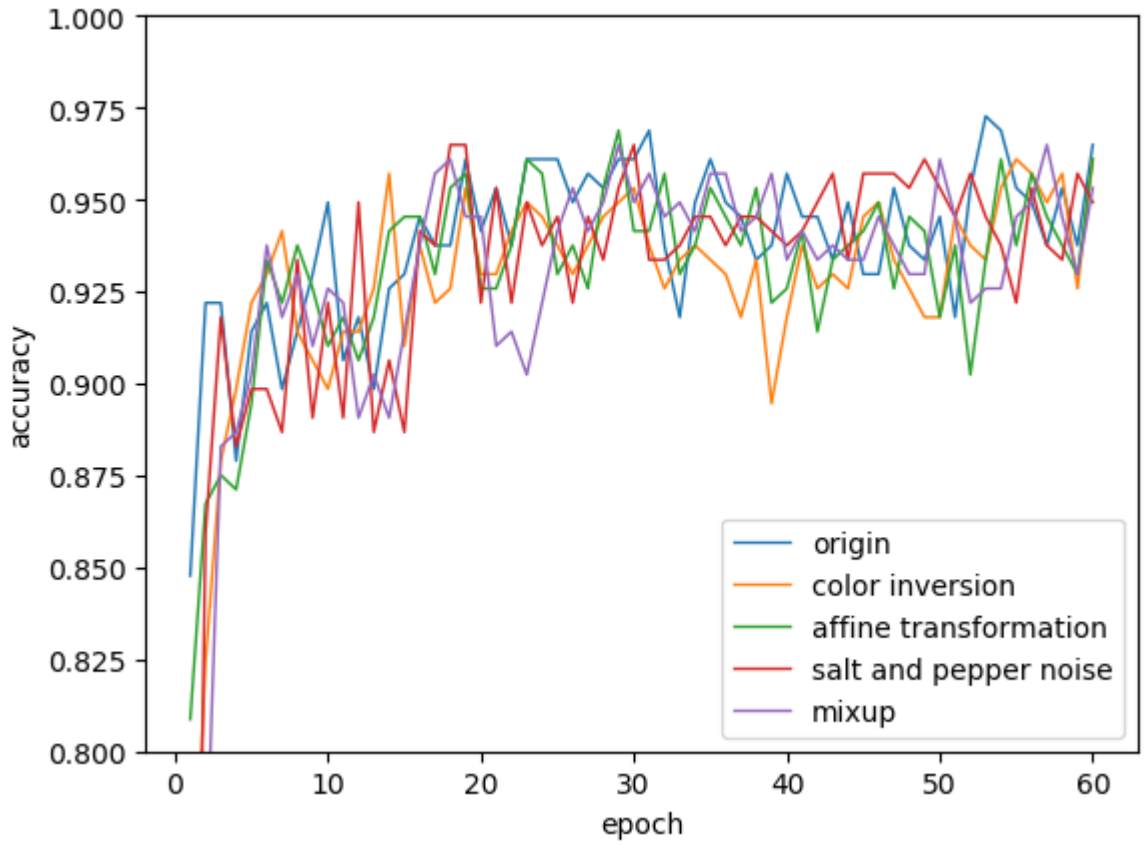
We can see, when we give up some data, the points have a high probability drop.



question 2

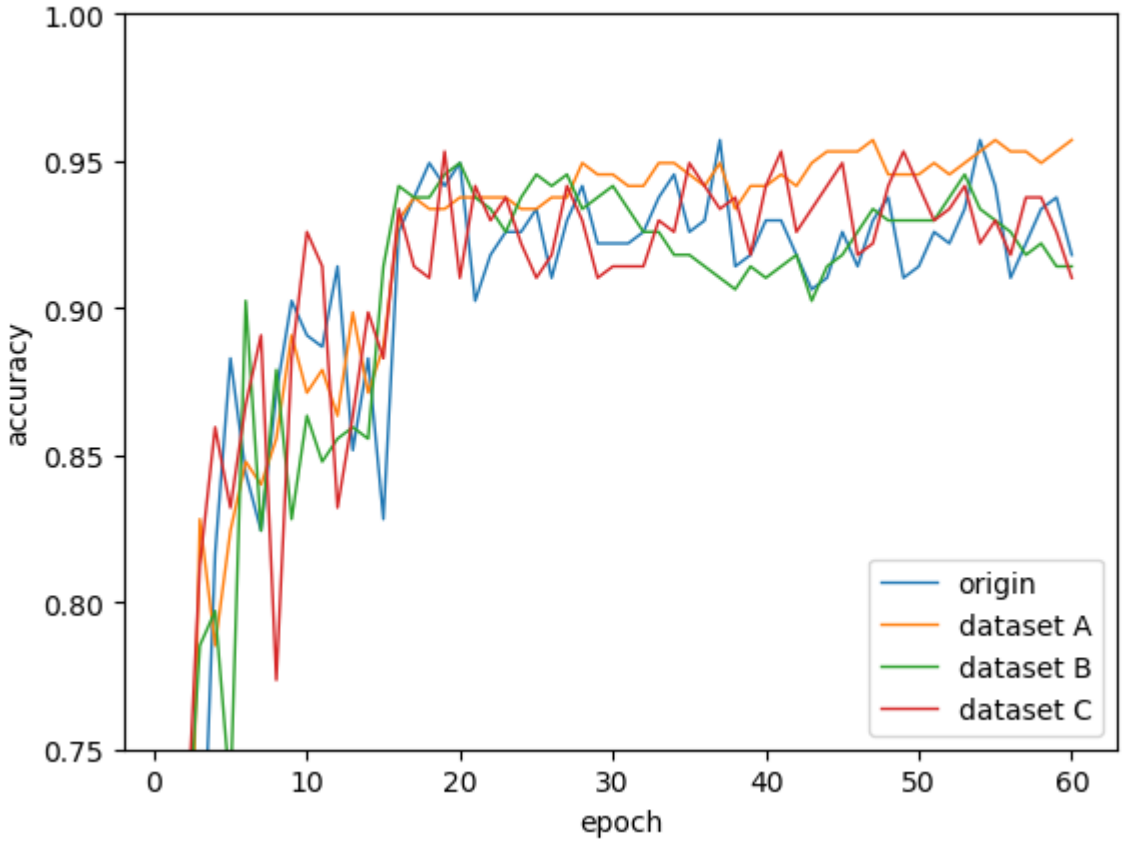
When we do data augmentation, the accuracy does not obviously grow. But it doesn't drop, either.

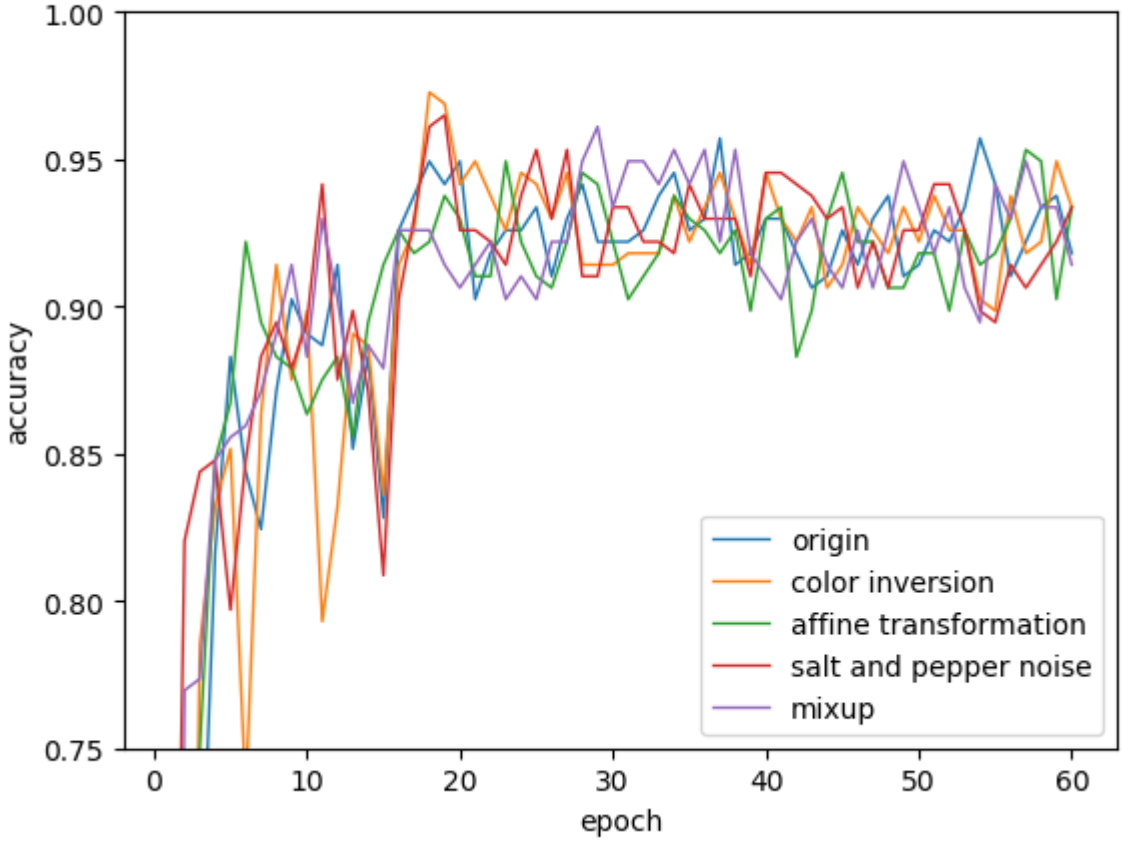
But mixup method can make points slightly grow. I think if the augmentation was combined with original dataset, the result will be much better.

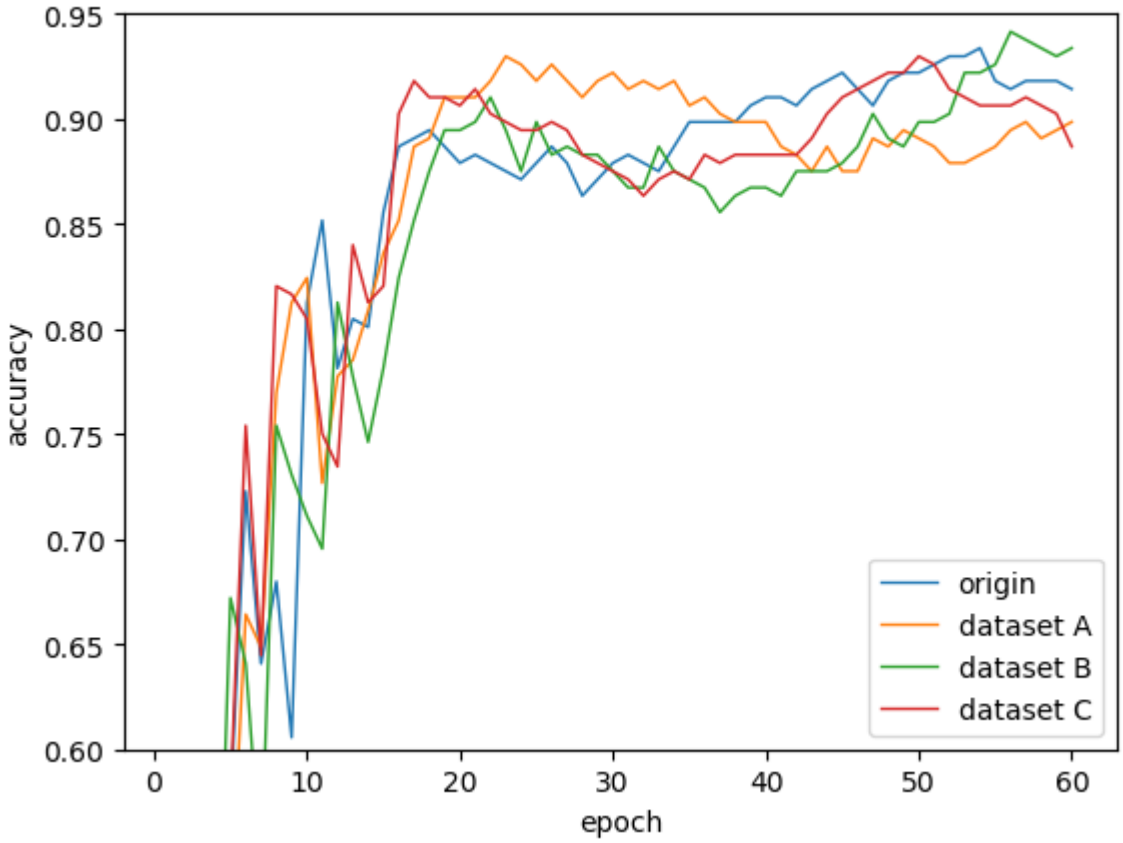


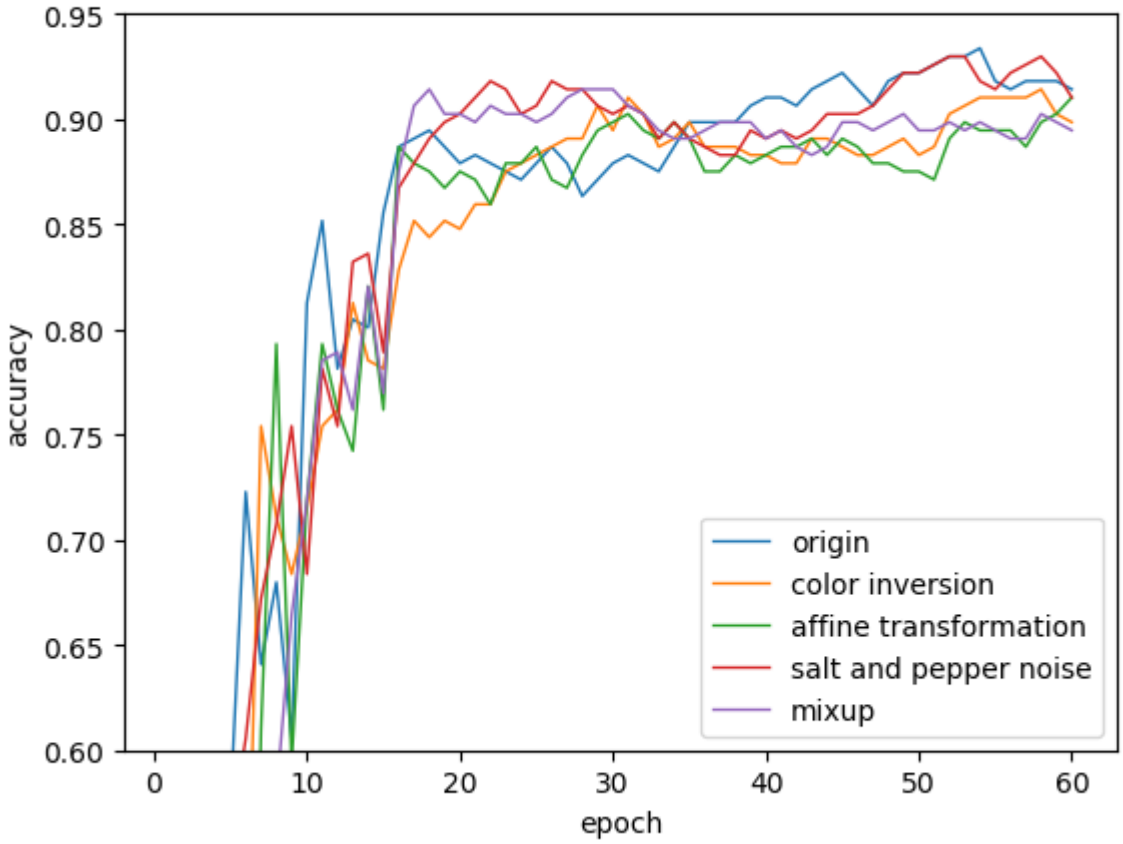
question 3

The figure represent the results give up some labels which is appointed before training. And their behaviors are not easy to describe. I think the figure is quite anti-intuitive. I have never thought the behavior on dataset A is so good, it's points is higher than origin.


1.00
0.95
0.90
0.85
0.80
0.75
accuracy
0
10
20
30
40
50
60
epoch
origin
dataset A
dataset B
dataset C


1.00
0.95
0.90
0.85
0.80
0.75
accuracy
0
10
20
30
40
50
60
epoch
origin
color inversion
affine transformation
salt and pepper noise
mixup


0.95
0.90
0.85
0.80
0.75
0.70
0.65
0.60
accuracy
0
10
20
30
40
50
60
epoch
origin
dataset A
dataset B
dataset C


0.95
0.90
0.85
0.80
0.75
0.70
0.65
0.60
accuracy
0
10
20
30
40
50
60
epoch
origin
color inversion
affine transformation
salt and pepper noise
mixup

And last, we show the accuracy of the last epoch of every model.

	all	**10k**	**30k**
origin	0.96484375	0.9140625	0.91796875
tag '8' '9' and '0'	0.96484375	0.8984375	0.95703125
tag '6', '7', '8', '9' and '0'	0.91796875	0.93359375	0.9140625
tag '1', '2', '3', '4' and '5'	0.94921875	0.88671875	0.91015625
color converge	0.9609375	0.8984375	0.93359375
affine trasform	0.9609375	0.91015625	0.93359375
salt pepper noise	0.94921875	0.91015625	0.93359375
mixup	0.953125	0.89453125	0.9140625