

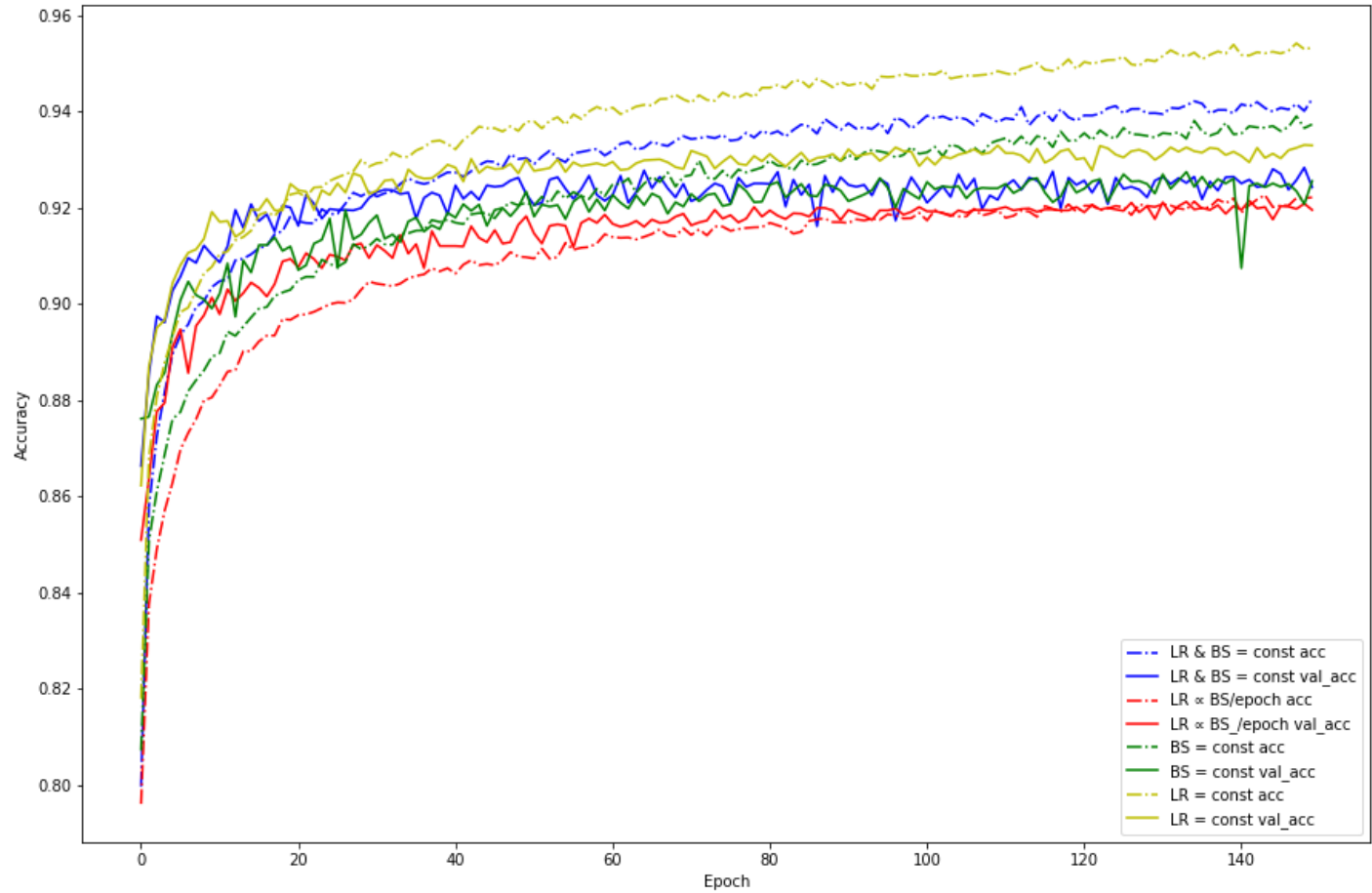
# Fashion Mnist

## Architecture:

Model: "sequential\_4"

Layer (type)	Output Shape	Param #
=====		
conv2d_30 (Conv2D)	(None, 28, 28, 64)	320
=====		
max_pooling2d_8 (MaxPooling2)	(None, 14, 14, 64)	0
=====		
dropout_12 (Dropout)	(None, 14, 14, 64)	0
=====		
conv2d_31 (Conv2D)	(None, 14, 14, 32)	8224
=====		
max_pooling2d_9 (MaxPooling2)	(None, 7, 7, 32)	0
=====		
dropout_13 (Dropout)	(None, 7, 7, 32)	0
=====		
flatten_4 (Flatten)	(None, 1568)	0
=====		
dense_8 (Dense)	(None, 256)	401664
=====		
dropout_14 (Dropout)	(None, 256)	0
=====		
dense_9 (Dense)	(None, 10)	2570
=====		

Total params: 412,778  
Trainable params: 412,778  
Non-trainable params: 0

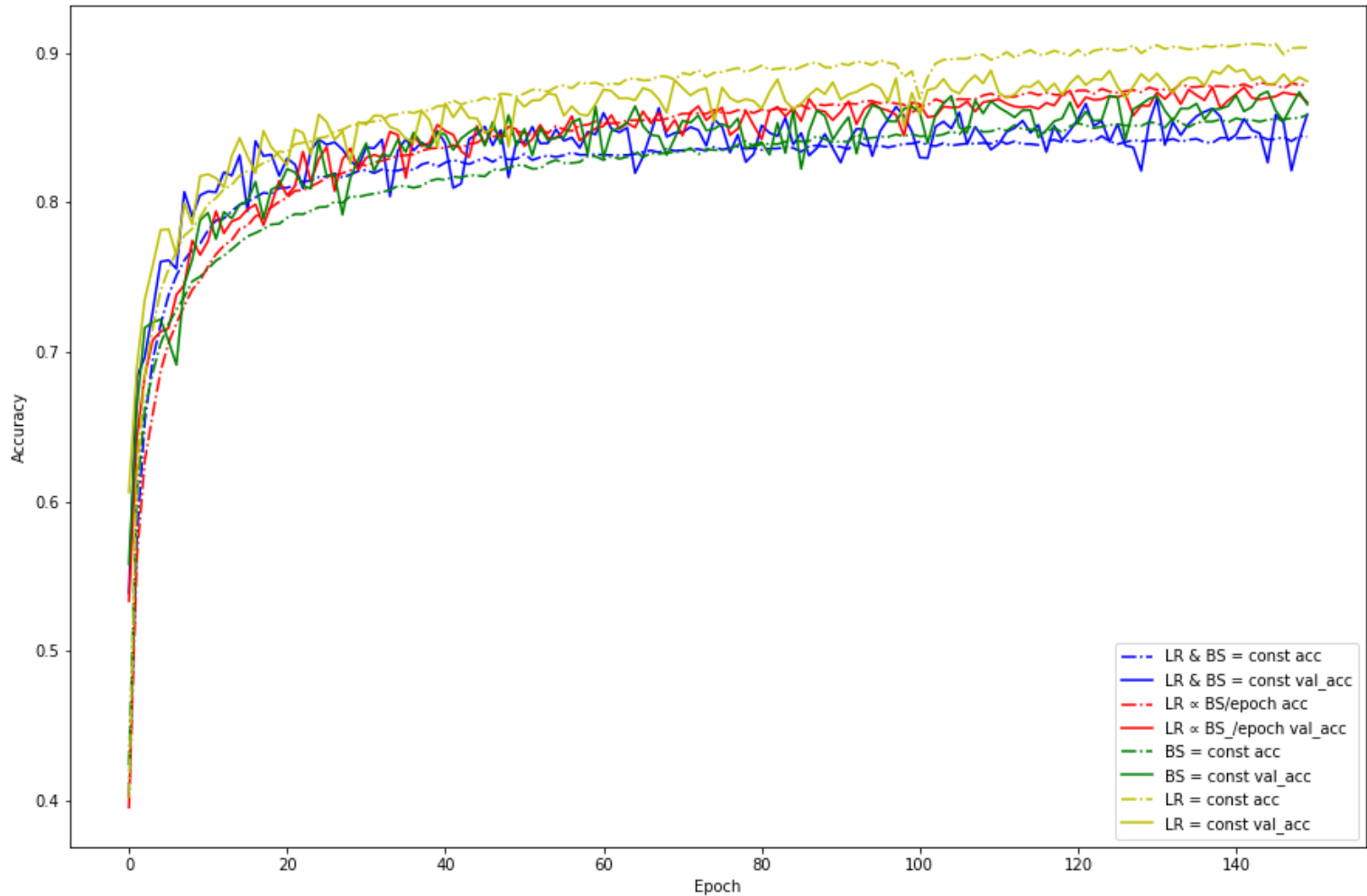


# Cifar-10

## Architecture:

Model: "sequential"

Layer (type)	Output Shape	Param #
=====		
conv2d (Conv2D)	(None, 32, 32, 32)	896
activation (Activation)	(None, 32, 32, 32)	0
batch_normalization (BatchNo	(None, 32, 32, 32)	128
conv2d_1 (Conv2D)	(None, 32, 32, 32)	9248
activation_1 (Activation)	(None, 32, 32, 32)	0
batch_normalization_1 (Batch	(None, 32, 32, 32)	128
max_pooling2d (MaxPooling2D)	(None, 16, 16, 32)	0
dropout (Dropout)	(None, 16, 16, 32)	0
conv2d_2 (Conv2D)	(None, 16, 16, 64)	18496
activation_2 (Activation)	(None, 16, 16, 64)	0
batch_normalization_2 (Batch	(None, 16, 16, 64)	256
conv2d_3 (Conv2D)	(None, 16, 16, 64)	36928
activation_3 (Activation)	(None, 16, 16, 64)	0
batch_normalization_3 (Batch	(None, 16, 16, 64)	256
max_pooling2d_1 (MaxPooling2	(None, 8, 8, 64)	0
dropout_1 (Dropout)	(None, 8, 8, 64)	0
conv2d_4 (Conv2D)	(None, 8, 8, 128)	73856
activation_4 (Activation)	(None, 8, 8, 128)	0
batch_normalization_4 (Batch	(None, 8, 8, 128)	512
conv2d_5 (Conv2D)	(None, 8, 8, 128)	147584
activation_5 (Activation)	(None, 8, 8, 128)	0
batch_normalization_5 (Batch	(None, 8, 8, 128)	512
max_pooling2d_2 (MaxPooling2	(None, 4, 4, 128)	0
dropout_2 (Dropout)	(None, 4, 4, 128)	0
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 10)	20490
=====		
Total params: 309,290		
Trainable params: 308,394		
Non-trainable params: 896		

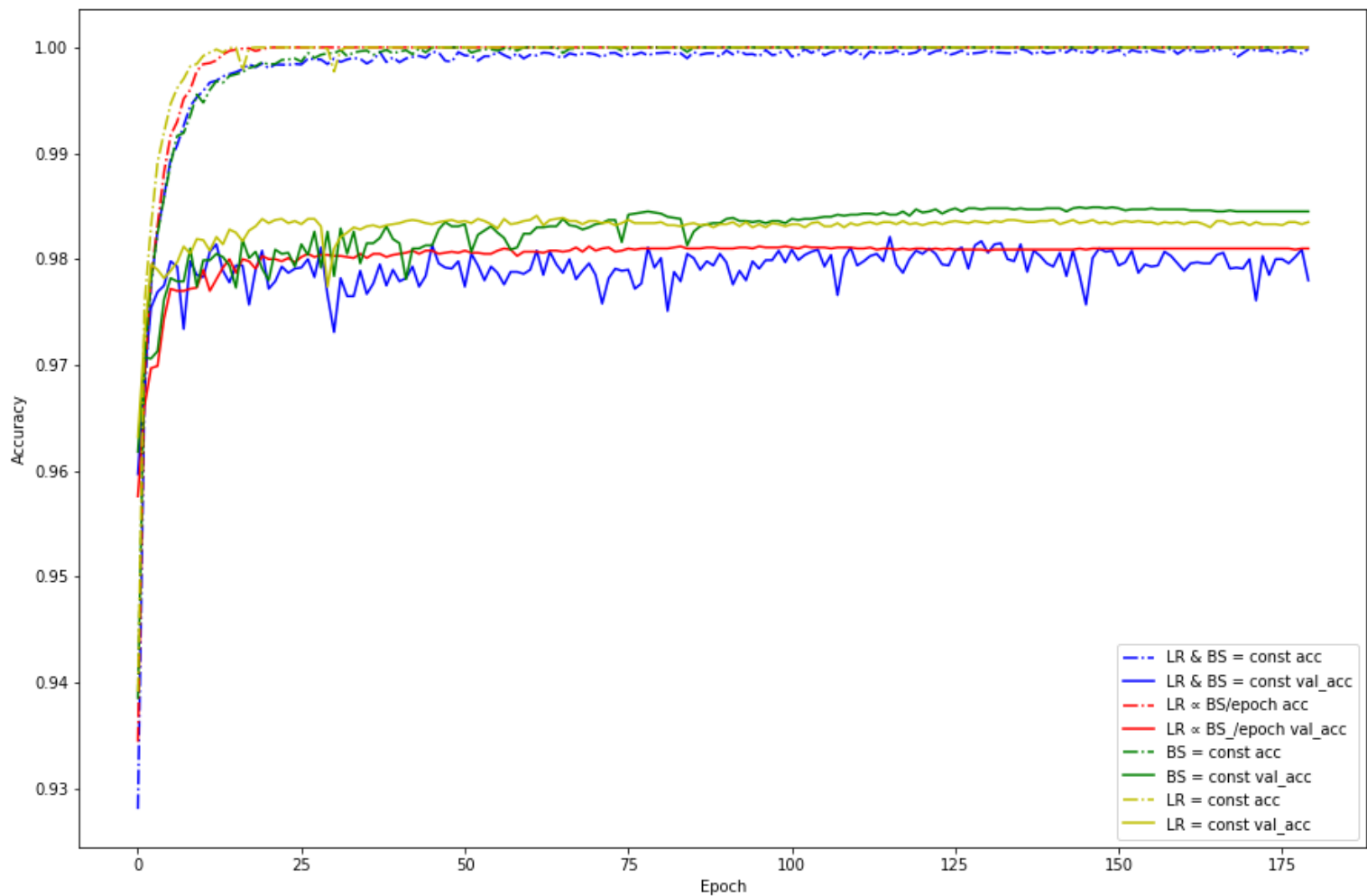


## Mnist

### Architecture:

Model: "sequential\_4"

Layer (type)	Output Shape	Param #
=====		
flatten_4 (Flatten)	(None, 784)	0
=====		
dense_8 (Dense)	(None, 128)	100480
=====		
dense_9 (Dense)	(None, 10)	1290
=====		
Total params: 101,770		
Trainable params: 101,770		
Non-trainable params: 0		
=====		



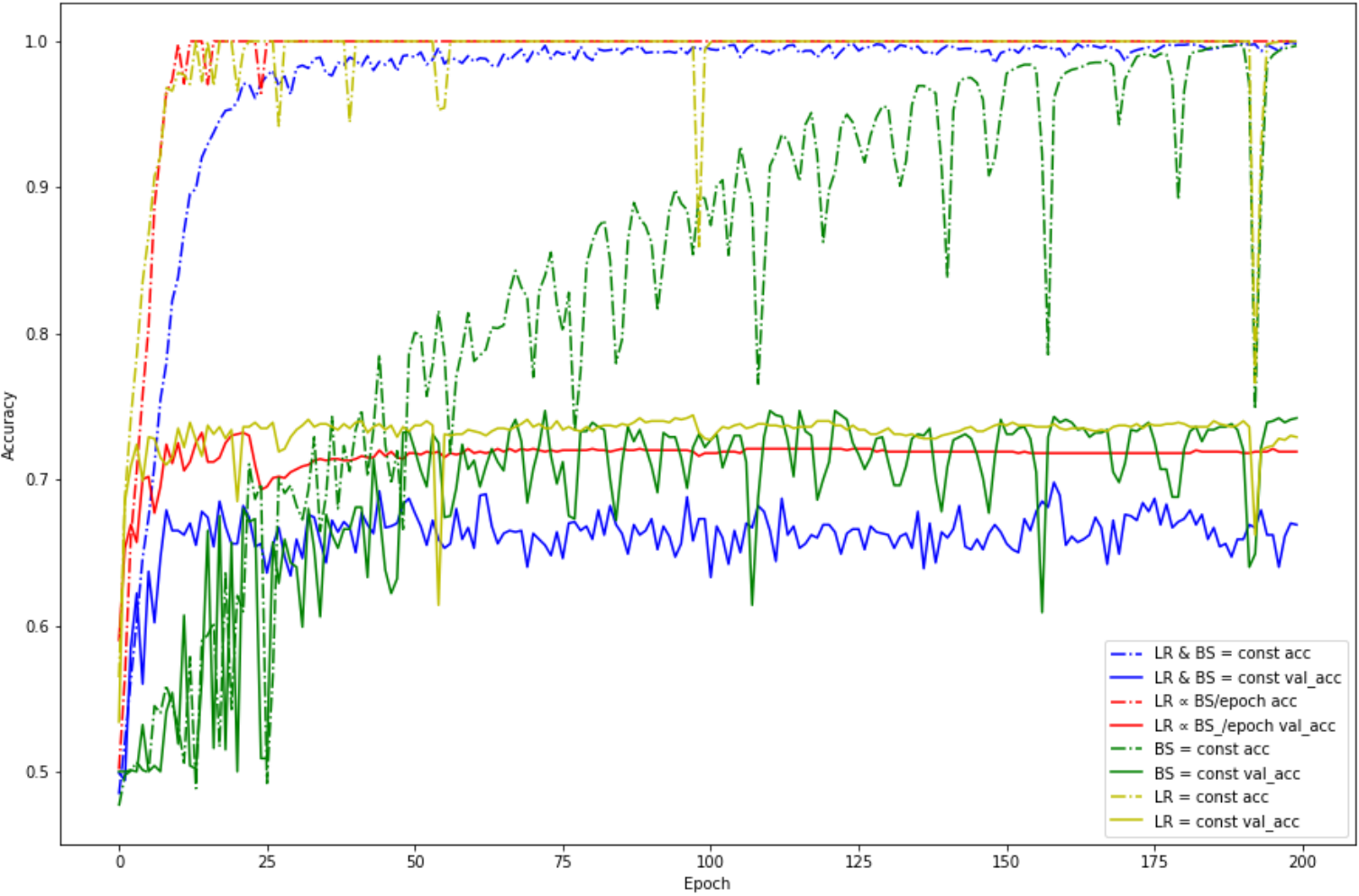
## Cats vs Dogs

### Architecture:

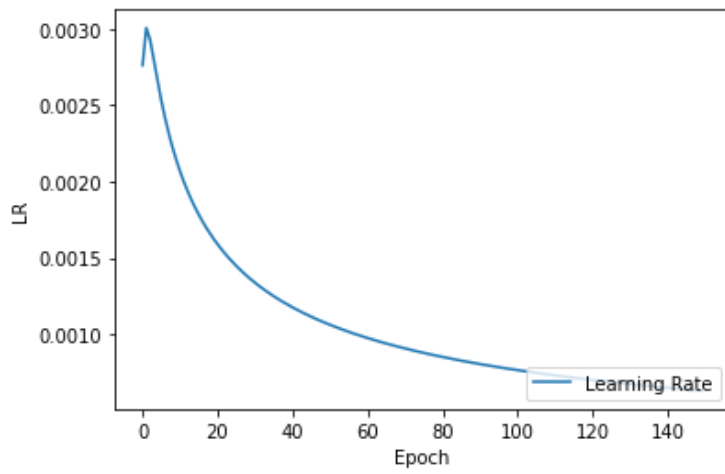
Model: "model\_1"

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 150, 150, 3)]	0
conv2d_25 (Conv2D)	(None, 148, 148, 16)	448
max_pooling2d_3 (MaxPooling2)	(None, 74, 74, 16)	0
conv2d_26 (Conv2D)	(None, 72, 72, 32)	4640
max_pooling2d_4 (MaxPooling2)	(None, 36, 36, 32)	0

conv2d_27 (Conv2D)	(None, 34, 34, 64)	18496
max_pooling2d_5 (MaxPooling2	(None, 17, 17, 64)	0
flatten_1 (Flatten)	(None, 18496)	0
dense_2 (Dense)	(None, 512)	9470464
dense_3 (Dense)	(None, 1)	513
=====		
Total params: 9,494,561		
Trainable params: 9,494,561		
Non-trainable params: 0		



## Learning Rate



$$LR = ((batch\_size)/((epoch+1)**(3/2)*80))/8$$

Where:

epoch = 1,2,3,4, - - - - - 150

batch\_size = 5,10,15,20