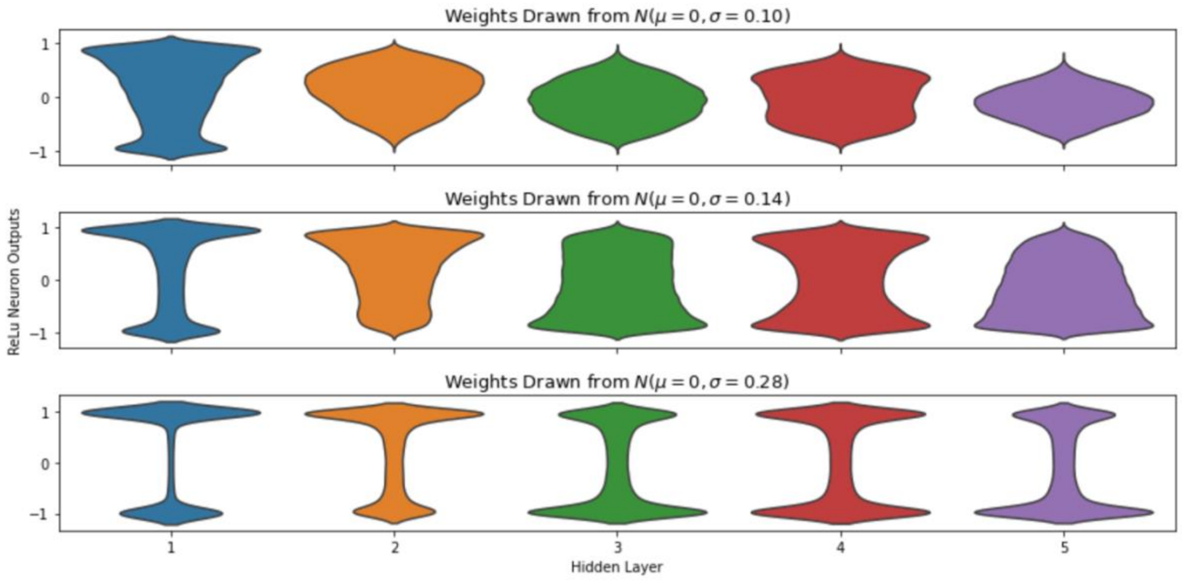
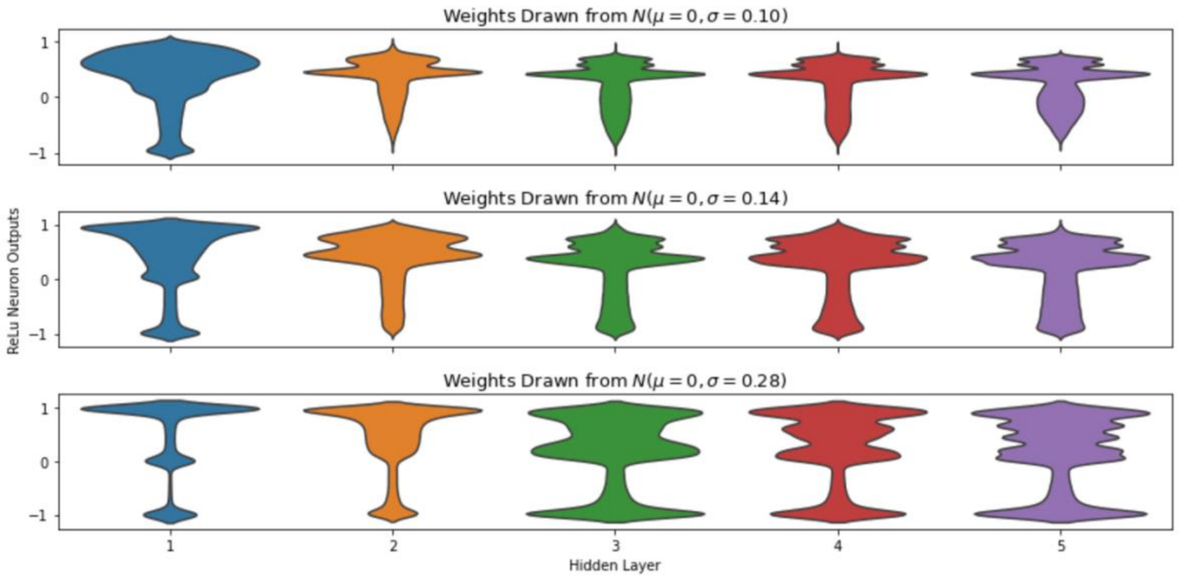
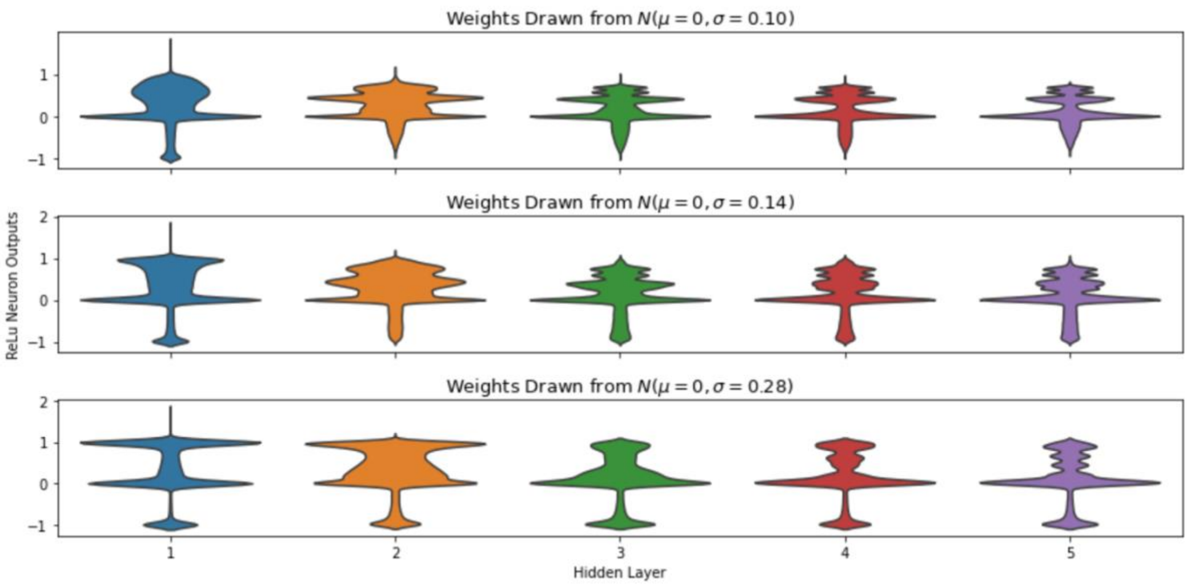
How I ran my jupyter notebook

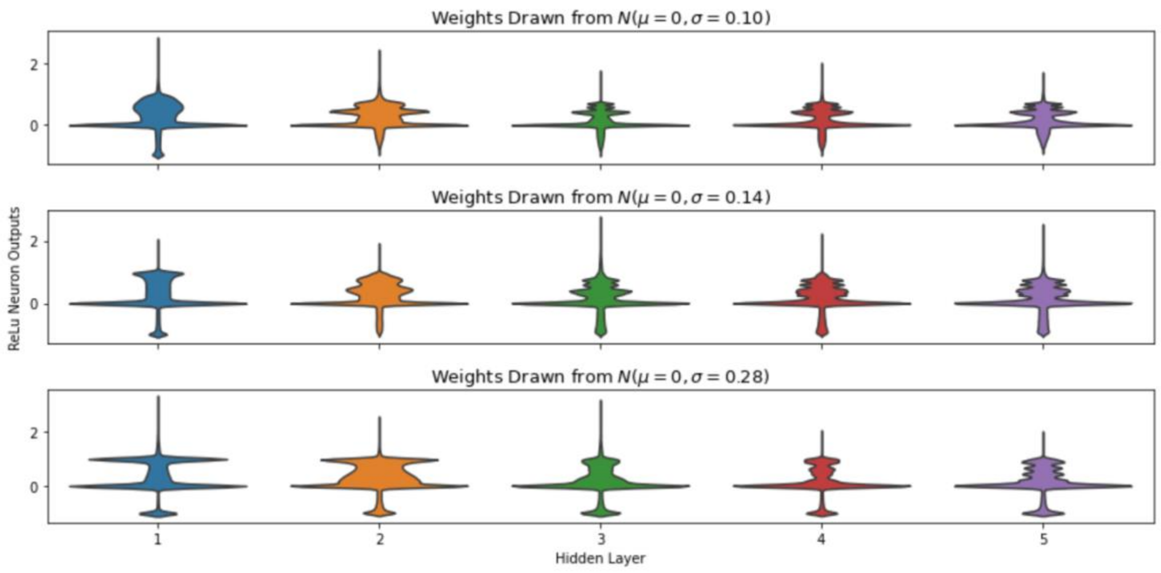
- I used NYU HPC Jupiter notebook from interactive apps to run my notebook with 14 cores and GPU because Google Codelab free version does have enough memory space. Google Codelab Link: https://colab.research.google.com/drive/1KZoqn3Xmqheo4ztORpWITsB79c5Zujjk?usp=sharing
- Since I couldn't submit my Jupyter notebook on Brightspace, I uploaded all the files to Google drive.
    - https://drive.google.com/drive/folders/1lYLcq5-b3EIr26pW5FNmU74AhTqFaqDy?usp=sharing

Problem 1





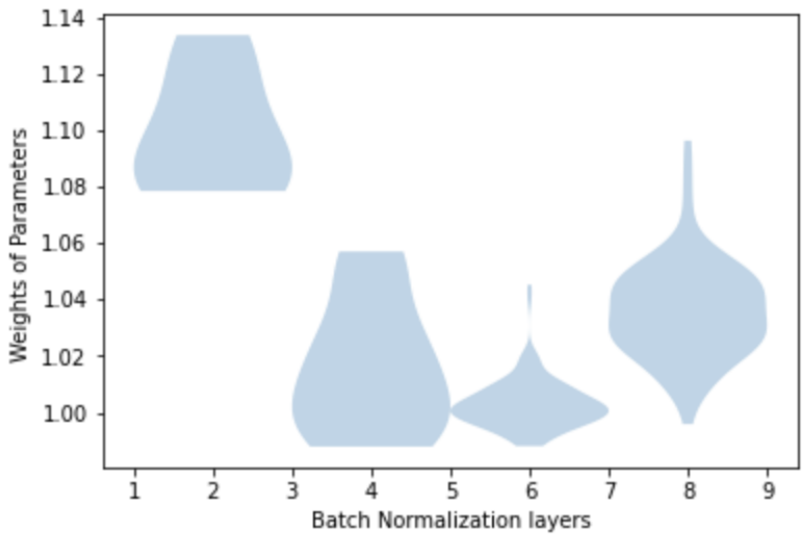

Weights Drawn from $N(\mu = 0, \sigma = 0.10)$
Weights Drawn from $N(\mu = 0, \sigma = 0.14)$
Weights Drawn from $N(\mu = 0, \sigma = 0.28)$
ReLu Neuron Outputs
Hidden Layer
1
2
3
4
5


Weights Drawn from $N(\mu = 0, \sigma = 0.10)$
Weights Drawn from $N(\mu = 0, \sigma = 0.14)$
Weights Drawn from $N(\mu = 0, \sigma = 0.28)$
ReLu Neuron Outputs
Hidden Layer
1
2
3
4
5

Problem 2

Part 2

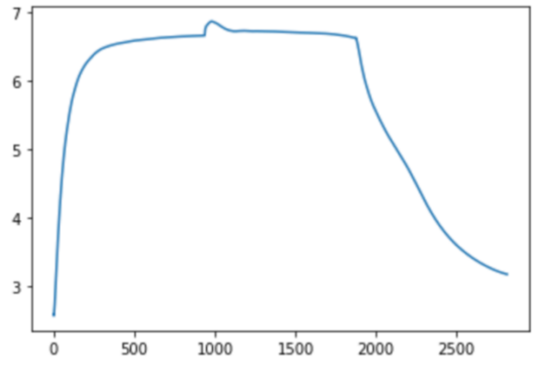
- Plot the distribution of learned batch norm parameters for each layer



Problem 3

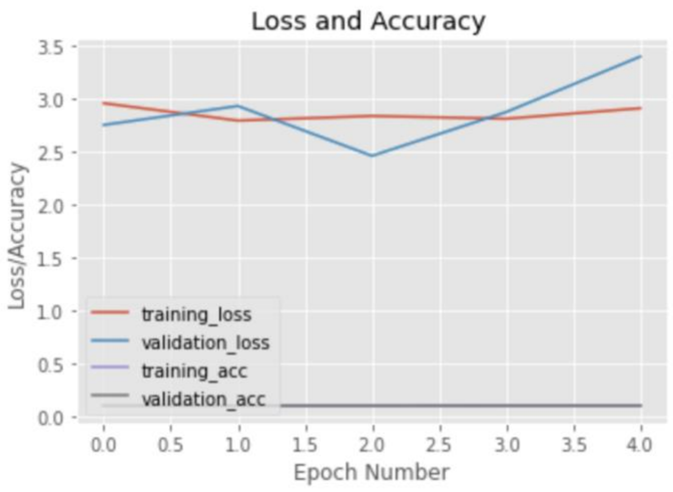
Part 1

- Plot the training loss as a function of the learning rate.

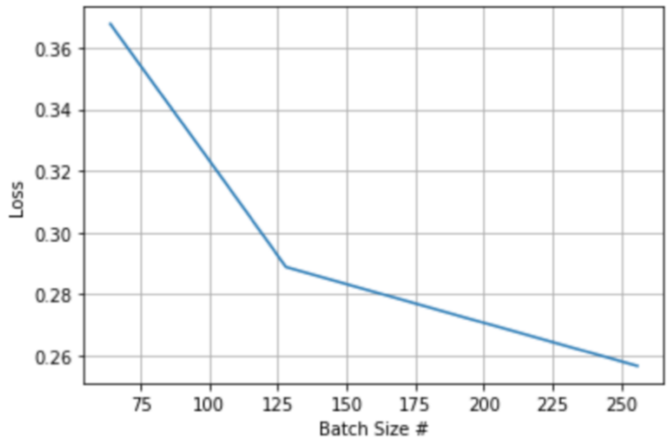


Part 2

- Plot train/validation loss and accuracy curve



Part 3

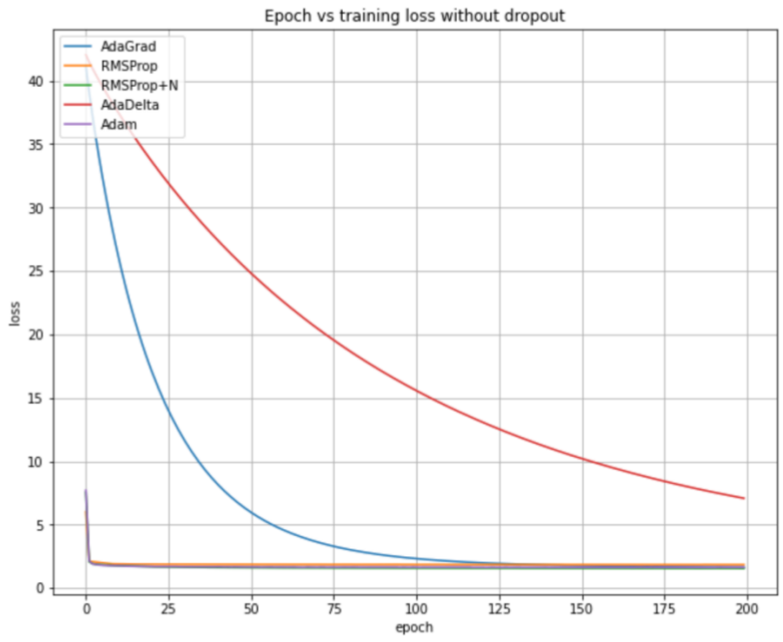


Problem 4

Part 2 without dropout

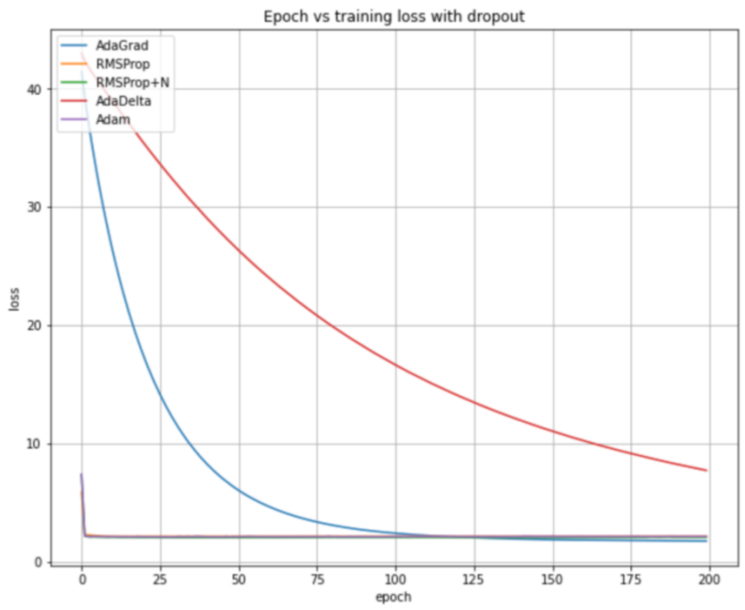
- plot the training loss vs the number of epochs.

```
lowest training loss
AdaGrad: 1.6244221925735474
RMSProp: 1.8377635478973389
RMSProp+N: 1.5503463745117188
AdaDelta: 7.079044342041016
Adam: 1.6213159561157227
```

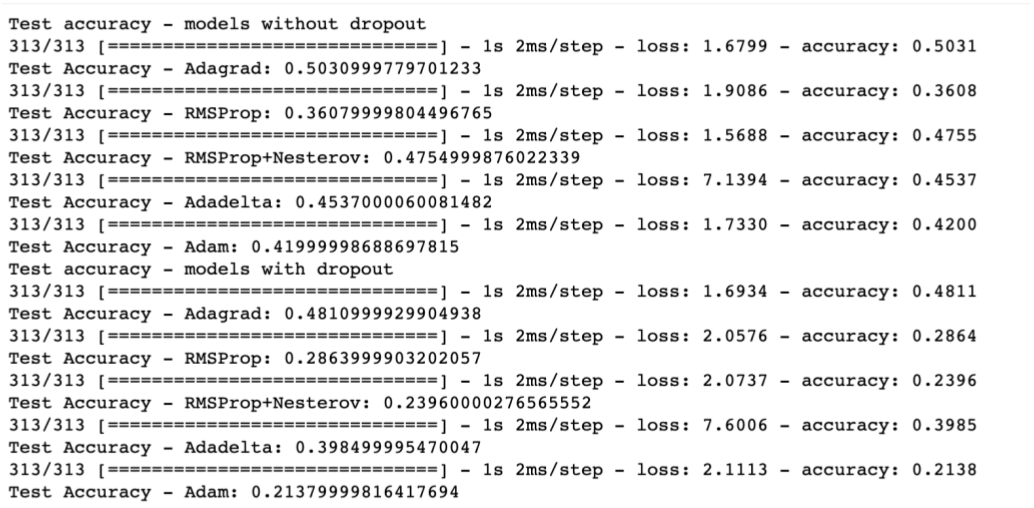


Part 3 with dropout

```
lowest training loss
AdaGrad: 1.7369143962860107
RMSProp: 2.115575075149536
RMSProp+N: 2.028042793273926
AdaDelta: 7.716529846191406
Adam: 2.0822479724884033
```



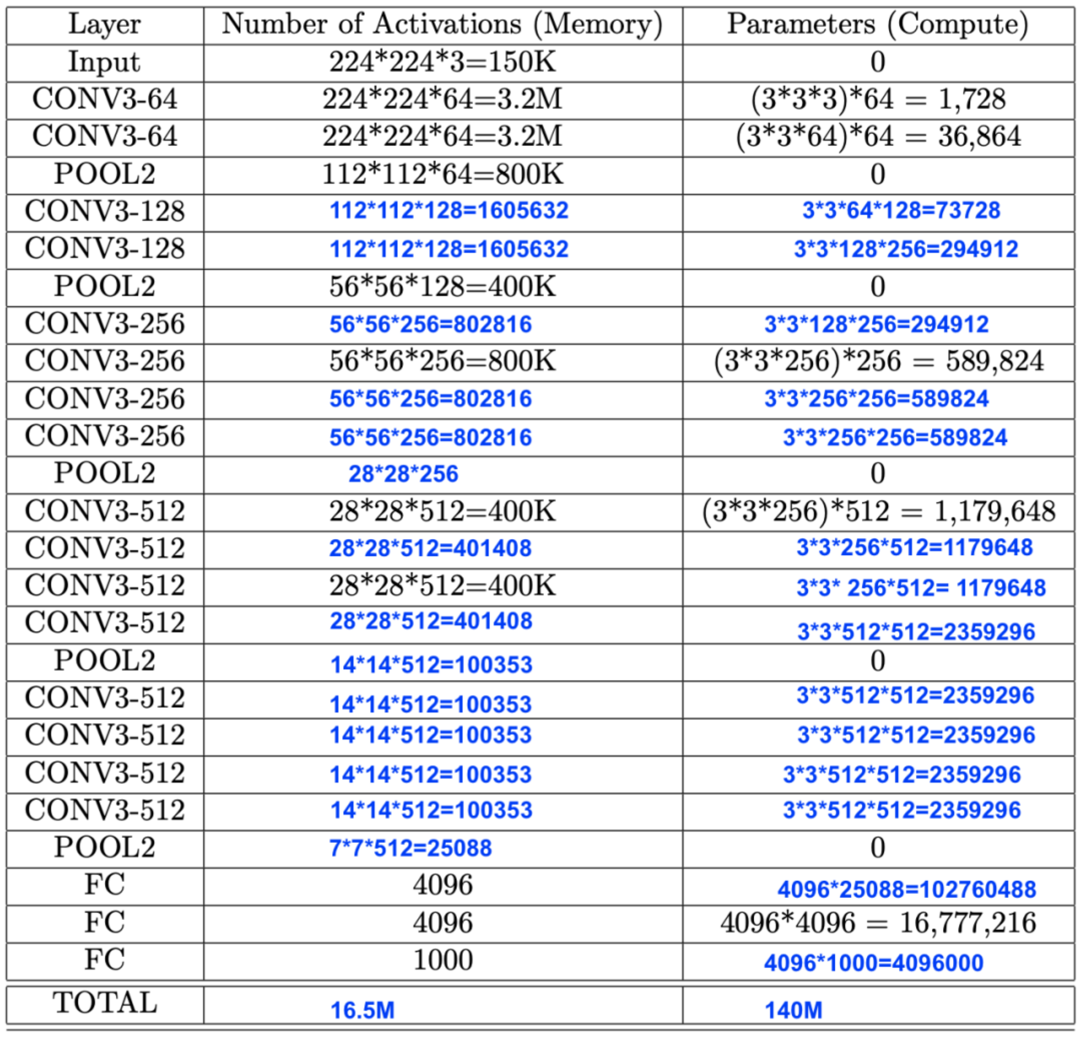
Part 4



```
Test accuracy - models without dropout
313/313 [==============================] - 1s 2ms/step - loss: 1.6799 - accuracy: 0.5031
Test Accuracy - Adagrad: 0.5030999779701233
313/313 [==============================] - 1s 2ms/step - loss: 1.9086 - accuracy: 0.3608
Test Accuracy - RMSProp: 0.36079999804496765
313/313 [==============================] - 1s 2ms/step - loss: 1.5688 - accuracy: 0.4755
Test Accuracy - RMSProp+Nesterov: 0.4754999876022339
313/313 [==============================] - 1s 2ms/step - loss: 7.1394 - accuracy: 0.4537
Test Accuracy - Adadelta: 0.4537000060081482
313/313 [==============================] - 1s 2ms/step - loss: 1.7330 - accuracy: 0.4200
Test Accuracy - Adam: 0.41999998688697815
Test accuracy - models with dropout
313/313 [==============================] - 1s 2ms/step - loss: 1.6934 - accuracy: 0.4811
Test Accuracy - Adagrad: 0.4810999929904938
313/313 [==============================] - 1s 2ms/step - loss: 2.0576 - accuracy: 0.2864
Test Accuracy - RMSProp: 0.2863999903202057
313/313 [==============================] - 1s 2ms/step - loss: 2.0737 - accuracy: 0.2396
Test Accuracy - RMSProp+Nesterov: 0.23960000276565552
313/313 [==============================] - 1s 2ms/step - loss: 7.6006 - accuracy: 0.3985
Test Accuracy - Adadelta: 0.398499995470047
313/313 [==============================] - 1s 2ms/step - loss: 2.1113 - accuracy: 0.2138
Test Accuracy - Adam: 0.21379999816417694
```

Problem 5

Part 1



Layer	Number of Activations (Memory)	Parameters (Compute)
Input	224*224*3=150K	0
CONV3-64	224*224*64=3.2M	(3*3*3)*64 = 1,728
CONV3-64	224*224*64=3.2M	(3*3*64)*64 = 36,864
POOL2	112*112*64=800K	0
CONV3-128	112*112*128=1605632	3*3*64*128=73728
CONV3-128	112*112*128=1605632	3*3*128*256=294912
POOL2	56*56*128=400K	0
CONV3-256	56*56*256=802816	3*3*128*256=294912
CONV3-256	56*56*256=800K	(3*3*256)*256 = 589,824
CONV3-256	56*56*256=802816	3*3*256*256=589824
CONV3-256	56*56*256=802816	3*3*256*256=589824
POOL2	28*28*256	0
CONV3-512	28*28*512=400K	(3*3*256)*512 = 1,179,648
CONV3-512	28*28*512=401408	3*3*256*512=1179648
CONV3-512	28*28*512=400K	3*3* 256*512= 1179648
CONV3-512	28*28*512=401408	3*3*512*512=2359296
POOL2	14*14*512=100353	0
CONV3-512	14*14*512=100353	3*3*512*512=2359296
CONV3-512	14*14*512=100353	3*3*512*512=2359296
CONV3-512	14*14*512=100353	3*3*512*512=2359296
CONV3-512	14*14*512=100353	3*3*512*512=2359296
POOL2	7*7*512=25088	0
FC	4096	4096*25088=102760488
FC	4096	4096*4096 = 16,777,216
FC	1000	4096*1000=4096000
TOTAL	16.5M	140M

Table 1: VGG19 memory and weights