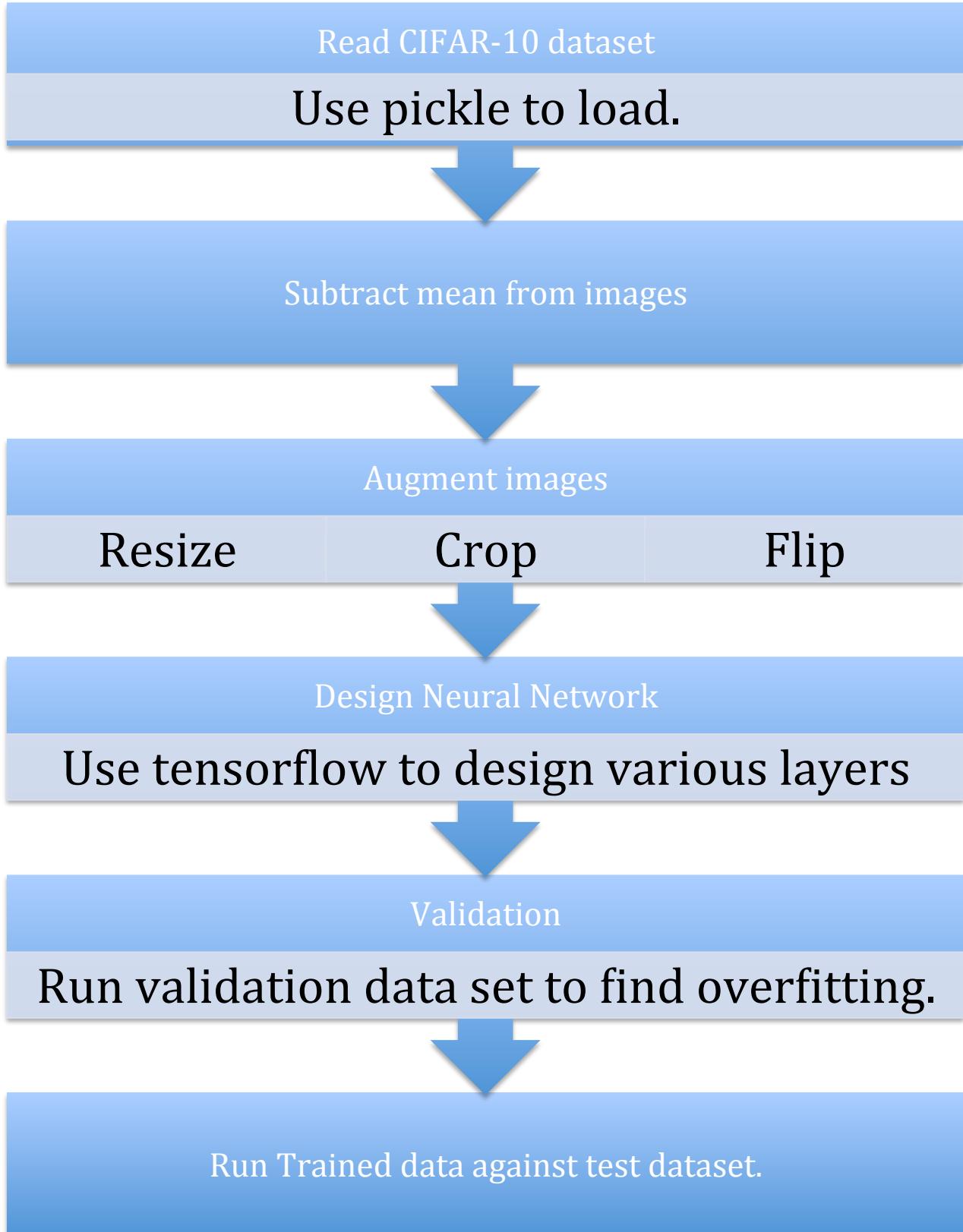


Table of Contents

STEPS FOLLOWED IN ALGORITHM.....	2
VARIOUS USE CASES.....	3
VARYING NUMBER OF EPOCHS.....	3
1.) EPOCHS = 10	3
2.) EPOCHS = 100.....	4
3.) EPOCHS = 1000.....	5
4.) EPOCHS = 2000.....	6
5.) EPOCHS = 3000.....	7
OVERALL PERFORMANCE BY VARYING EPOCHS.....	8
VARYING FILTER SIZE.....	9
1.) FILTER SIZE IS 2X2 FOR BOTH LAYERS.....	9
2.) FILTER SIZE IS 5X5 FOR BOTH LAYERS.....	10
3.) FILTER SIZE 10X10 FOR BOTH LAYERS.....	11
4.) FILTER SIZE 20X20 FOR BOTH LAYERS.....	12
OVERALL PERFORMANCE.....	13
VARYING NUMBER OF FILTERS	14
1.) FIRST LAYER OF SIZE 4 AND SECOND LAYER OF SIZE 16	14
2.) FIRST LAYER IS 6 AND SECOND LAYER IS 16	15
3.) FIRST LAYER IS 8 AND SECOND LAYER IS 16	16
4.) FIRST LAYER IS 6 AND SECOND LAYER IS 10	17
5.) FIRST LAYER IS 6 AND SECOND LAYER IS 18	18
6.) FIRST LAYER IS 8 AND SECOND LAYER IS 20	19
OVERALL PERFORMANCE.....	20
VARYING SIZE OF FULLY CONNECTED LAYERS	21
1.) BOTH FULLY CONNECTED LAYERS OF SIZE 50	21
3.) FULLY CONNECTED LAYER 1 = 100 AND FULLY CONNECTED LAYER 2 = 50.....	23
4.) FULLY CONNECTED LAYER 1 = 150 AND FULLY CONNECTED LAYER 2 = 150.....	24
5.) FULLY CONNECTED LAYER 1 = 120 AND FULLY CONNECTED LAYER 2 = 84	25
6.) FULLY CONNECTED LAYER 1 = 200 AND FULLY CONNECTED LAYER 2 = 200	26
OVERALL PERFORMANCE.....	27
VARYING THE REGULARIZER	28
1.) WEIGHT DECAY = 1E-8.....	28
2.) WEIGHT DECAY = 1E-10.....	29
3.) WEIGHT DECAY = 1E-15.....	30
4.) WEIGHT DECAY = 0.00001.....	31
MY COMMENTS.....	32
SOURCE CODE	32
REFERENCES USED.....	39

Steps followed in Algorithm



Various use Cases.

Here I have listed accuracy result with various combinations.

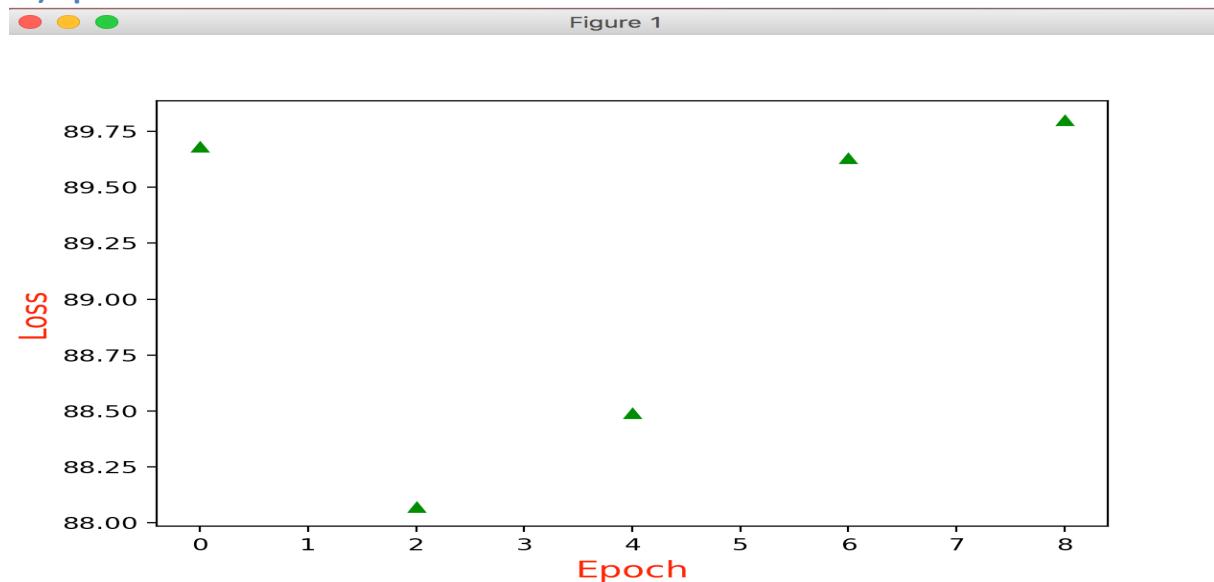
Varying number of epochs.

Here are the results when I varied epoch steps.

Constant Variables:

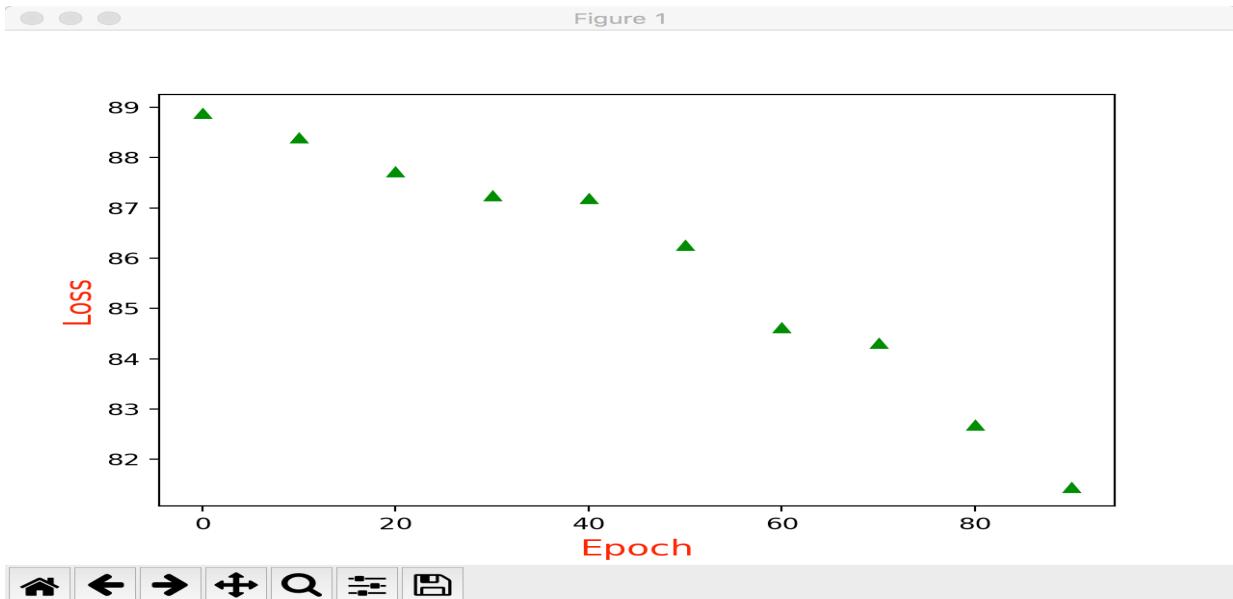
Layers as described in HW itself. I have just varied number of epochs keeping all configuration same as mentioned in the HW itself. Learning rate = 0.001

1.) Epochs = 10



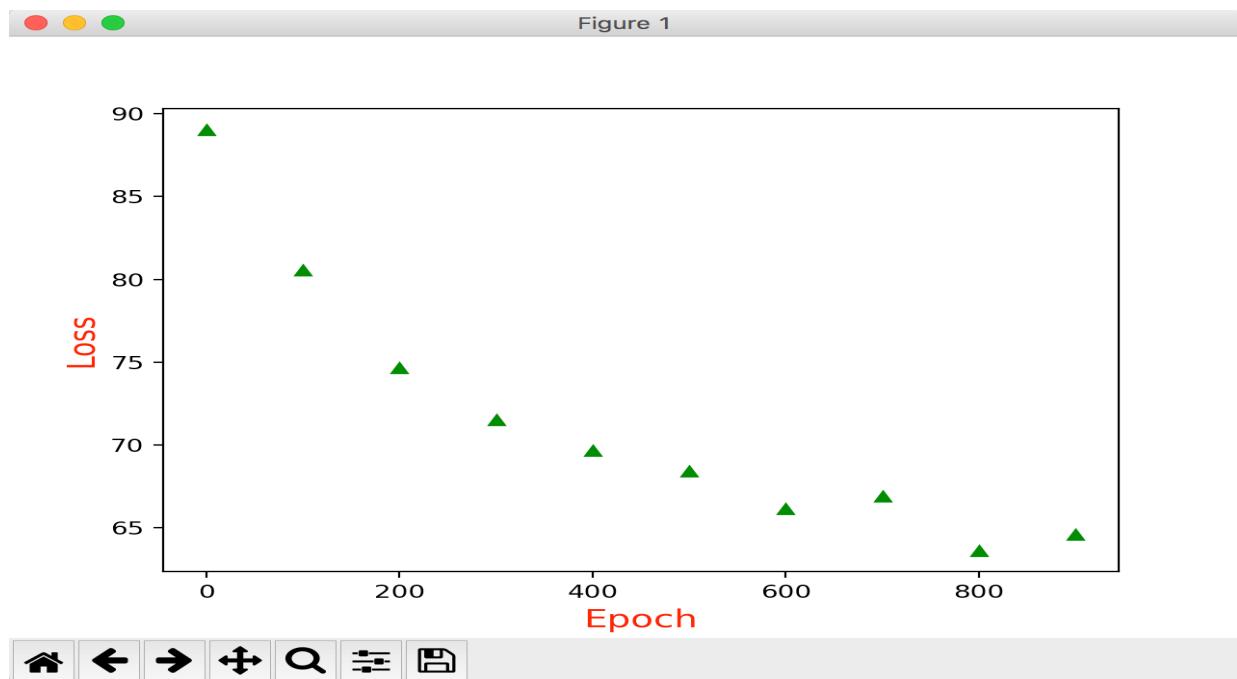
```
Initialized with learning_rate 0.001
epoch 0000 : loss is 037.06, accuracy on training set 12.50 %, accuracy on validation set 10.32 %
epoch 0002 : loss is 013.79, accuracy on training set 12.50 %, accuracy on validation set 11.93 %
epoch 0004 : loss is 009.45, accuracy on training set 9.38 %, accuracy on validation set 11.51 %
epoch 0006 : loss is 007.71, accuracy on training set 14.06 %, accuracy on validation set 10.37 %
epoch 0008 : loss is 006.58, accuracy on training set 18.75 %, accuracy on validation set 10.20 %
Accuracy on test model: 10.4
Total loss in all epochs: 74.5898218155
>Hit Enter To Close....
Confusion Matrix followed by accuracy of each class
[ 0  92 243  15  54 112  91 305  61  27](0) airplane    0.0%
[ 0  89 176  11  47 103  71 418  64  21](1) automobile  8.9%
[ 0 115 112  33  69 118 119 359  48  27](2) bird      11.2%
[ 0  99  96  31  60 120 116 409  44  25](3) cat       3.1%
[ 0  98 111  44  50 124 111 399  39  24](4) deer      5.0%
[ 0  98 108  26  59 130 127 397  37  18](5) dog       13.0%
[ 0  75 155  16  48  91  84 440  75  16](6) frog      8.4%
[ 0 126 105  31  42 112 110 416  25  33](7) horse     41.6%
[ 0  84 329   9  50  70  50 279 105  24](8) ship      10.5%
[ 0  82 186  15  81 115  93 360  45  23](9) truck     2.3%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)
```

2.) Epochs = 100



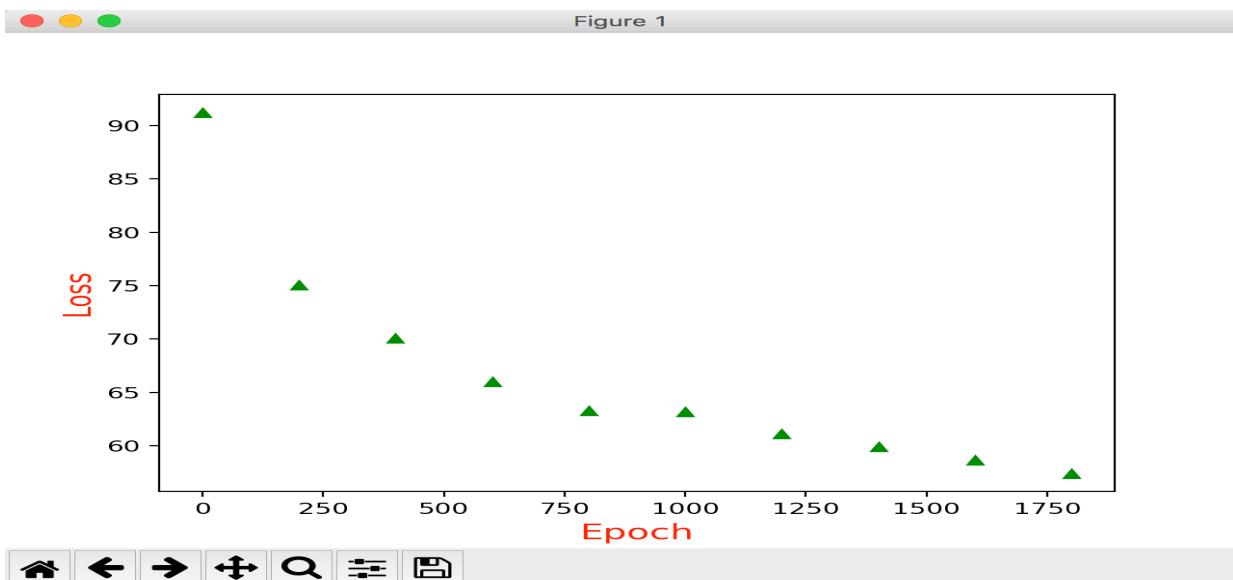
```
Initialized with learning_rate 0.001
epoch 0000 : loss is 022.03, accuracy on training set 17.19 %, accuracy on validation set 11.12 %
epoch 0010 : loss is 004.08, accuracy on training set 15.62 %, accuracy on validation set 11.61 %
epoch 0020 : loss is 003.00, accuracy on training set 14.06 %, accuracy on validation set 12.28 %
epoch 0030 : loss is 002.52, accuracy on training set 10.94 %, accuracy on validation set 12.76 %
epoch 0040 : loss is 002.27, accuracy on training set 17.19 %, accuracy on validation set 12.82 %
epoch 0050 : loss is 002.42, accuracy on training set 14.06 %, accuracy on validation set 13.75 %
epoch 0060 : loss is 002.32, accuracy on training set 17.19 %, accuracy on validation set 15.39 %
epoch 0070 : loss is 002.30, accuracy on training set 17.19 %, accuracy on validation set 15.70 %
epoch 0080 : loss is 002.30, accuracy on training set 25.00 %, accuracy on validation set 17.32 %
epoch 0090 : loss is 002.29, accuracy on training set 15.62 %, accuracy on validation set 18.56 %
Accuracy on test model: 18.88
Total loss in all epochs: 45.52368927
>Hit Enter To Close.....
Confusion Matrix followed by accuracy of each class
[[303 190 46 20 75 78 77 72 64 75](0) airplane 30.3%
 [ 78 280 45 27 92 94 69 43 78 194](1) automobile 28.0%
 [133 144 115 35 115 144 102 73 45 94](2) bird 11.5%
 [ 56 174 76 44 144 141 122 73 41 129](3) cat 4.4%
 [ 89 170 62 16 161 150 133 81 39 99](4) deer 16.1%
 [ 63 166 76 52 143 212 85 74 37 92](5) dog 21.2%
 [ 58 176 47 29 121 96 214 69 42 148](6) frog 21.4%
 [ 80 147 69 41 163 117 67 123 52 141](7) horse 12.3%
 [161 261 38 19 48 79 62 66 133 133](8) ship 13.3%
 [ 56 234 36 27 83 73 59 62 67 303](9) truck 30.3%
 (0) (1) (2) (3) (4) (5) (6) (7) (8) (9)
```

3.) Epochs = 1000



```
Initialized with learning_rate 0.001
epoch 0000 : loss is 014.36, accuracy on training set 9.38 %, accuracy on validation set 10.97 %
epoch 0100 : loss is 002.40, accuracy on training set 17.19 %, accuracy on validation set 19.44 %
epoch 0200 : loss is 002.07, accuracy on training set 23.44 %, accuracy on validation set 25.34 %
epoch 0300 : loss is 001.85, accuracy on training set 32.81 %, accuracy on validation set 28.48 %
epoch 0400 : loss is 001.90, accuracy on training set 28.12 %, accuracy on validation set 30.33 %
epoch 0500 : loss is 001.85, accuracy on training set 32.81 %, accuracy on validation set 31.58 %
epoch 0600 : loss is 001.74, accuracy on training set 32.81 %, accuracy on validation set 33.84 %
epoch 0700 : loss is 001.76, accuracy on training set 37.50 %, accuracy on validation set 33.08 %
epoch 0800 : loss is 002.11, accuracy on training set 34.38 %, accuracy on validation set 36.38 %
epoch 0900 : loss is 001.72, accuracy on training set 31.25 %, accuracy on validation set 35.41 %
Accuracy on test model: 36.95 %
Total loss in all epochs: 31.7550922632
>Hit Enter To Close....
Confusion Matrix followed by accuracy of each class
[[362 41 159  9 13 21 61 28 256 50](0) airplane 36.2%
 [ 36 322 35 11 12  6 91 40 210 237](1) automobile 32.2%
 [104 19 402 49 43 115 150 43 38 37](2) bird 40.2%
 [ 36 27 192 197 61 206 223 63 22 63](3) cat 10.7%
 [ 59 25 211 50 151 129 236 74 32 33](4) deer 15.1%
 [ 21  7 233 88 58 372 100 78 14 29](5) dog 37.2%
 [ 20 32 103 29 24 35 667 16 19 55](6) frog 66.7%
 [ 32 24 185 50 74 120 117 317 21 60](7) horse 31.7%
 [122 86 48  9  7 14 60 17 539 98](8) ship 53.9%
 [ 24 133 53 11 16 12 111 23 161 456](9) truck 45.6%
 (0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]
```

4.) Epochs = 2000

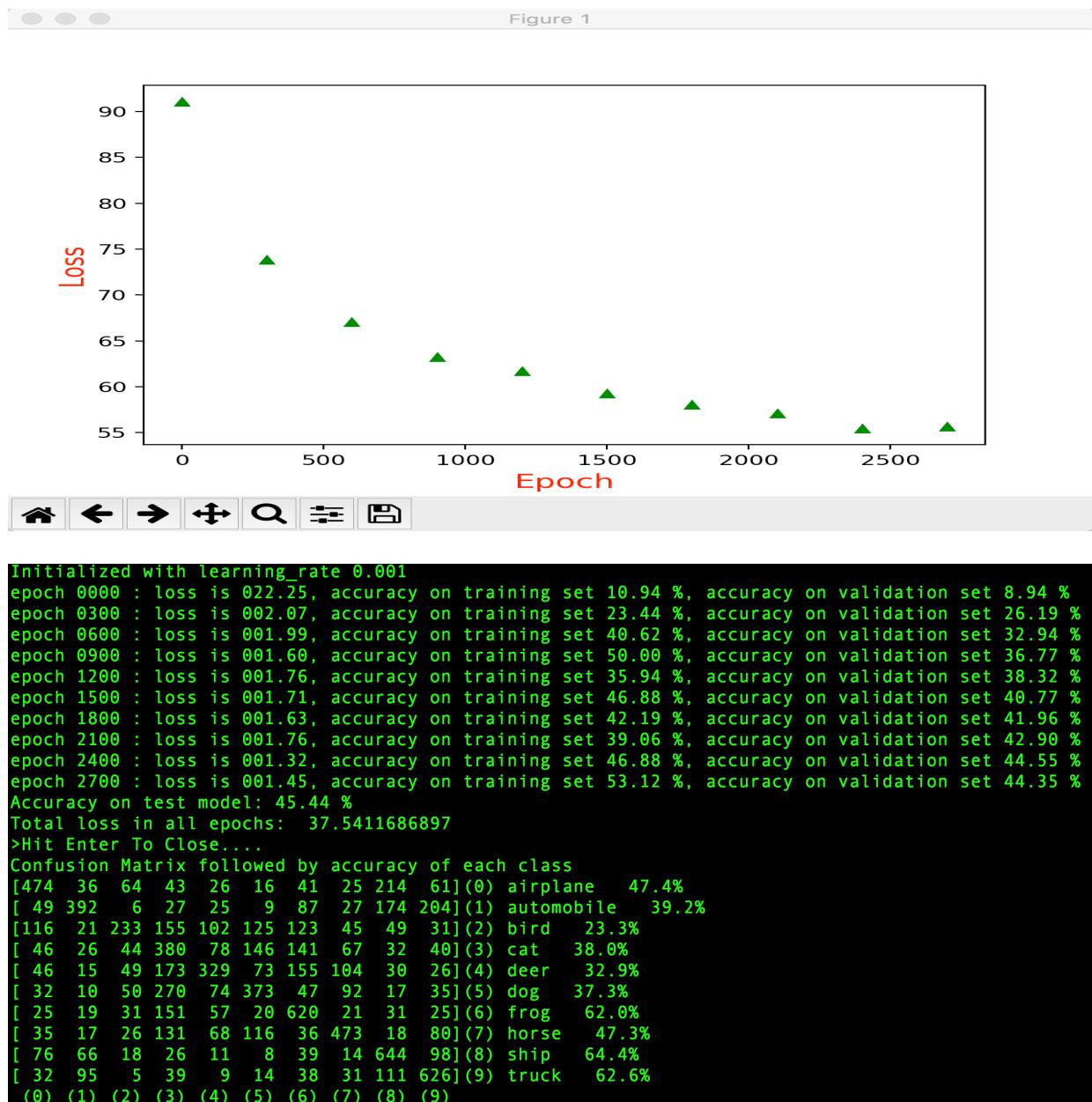


```
Initialized with learning_rate 0.001
epoch 0000 : loss is 024.20, accuracy on training set 4.69 %, accuracy on validation set 8.80 %
epoch 0200 : loss is 001.98, accuracy on training set 31.25 %, accuracy on validation set 24.93 %
epoch 0400 : loss is 001.98, accuracy on training set 23.44 %, accuracy on validation set 29.91 %
epoch 0600 : loss is 002.00, accuracy on training set 26.56 %, accuracy on validation set 34.00 %
epoch 0800 : loss is 001.86, accuracy on training set 32.81 %, accuracy on validation set 36.71 %
epoch 1000 : loss is 001.93, accuracy on training set 42.19 %, accuracy on validation set 36.84 %
epoch 1200 : loss is 001.72, accuracy on training set 42.19 %, accuracy on validation set 38.89 %
epoch 1400 : loss is 001.60, accuracy on training set 40.62 %, accuracy on validation set 40.07 %
epoch 1600 : loss is 001.68, accuracy on training set 46.88 %, accuracy on validation set 41.32 %
epoch 1800 : loss is 001.46, accuracy on training set 50.00 %, accuracy on validation set 42.59 %
Accuracy on test model: 43.82 %
Total loss in all epochs: 40.3960852623
>Hit Enter To Close....
```

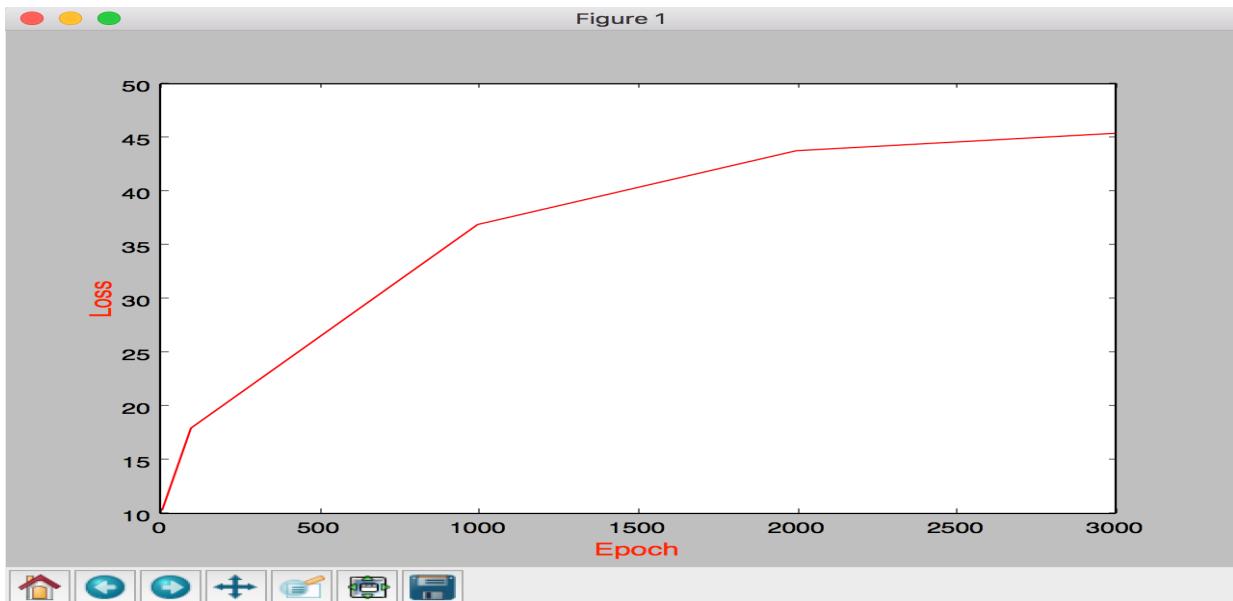
Confusion Matrix followed by accuracy of each class

[513 76 71 14 44 6 42 21 133 80](0) airplane	51.3%
[41 453 6 10 32 12 50 17 64 315](1) automobile	45.3%
[148 27 287 52 164 87 88 68 31 48](2) bird	28.7%
[73 42 82 158 164 153 146 95 15 72](3) cat	15.8%
[59 44 68 42 425 55 131 106 23 47](4) deer	42.5%
[54 14 73 117 160 331 70 145 4 32](5) dog	33.1%
[39 52 45 53 84 26 606 31 10 54](6) frog	60.6%
[50 32 29 52 137 81 31 469 7 112](7) horse	46.9%
[130 162 18 10 39 4 29 22 477 109](8) ship	47.7%
[34 132 7 13 27 4 29 39 52 663](9) truck	66.3%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)	

5.) Epochs = 3000



Overall Performance by varying epochs

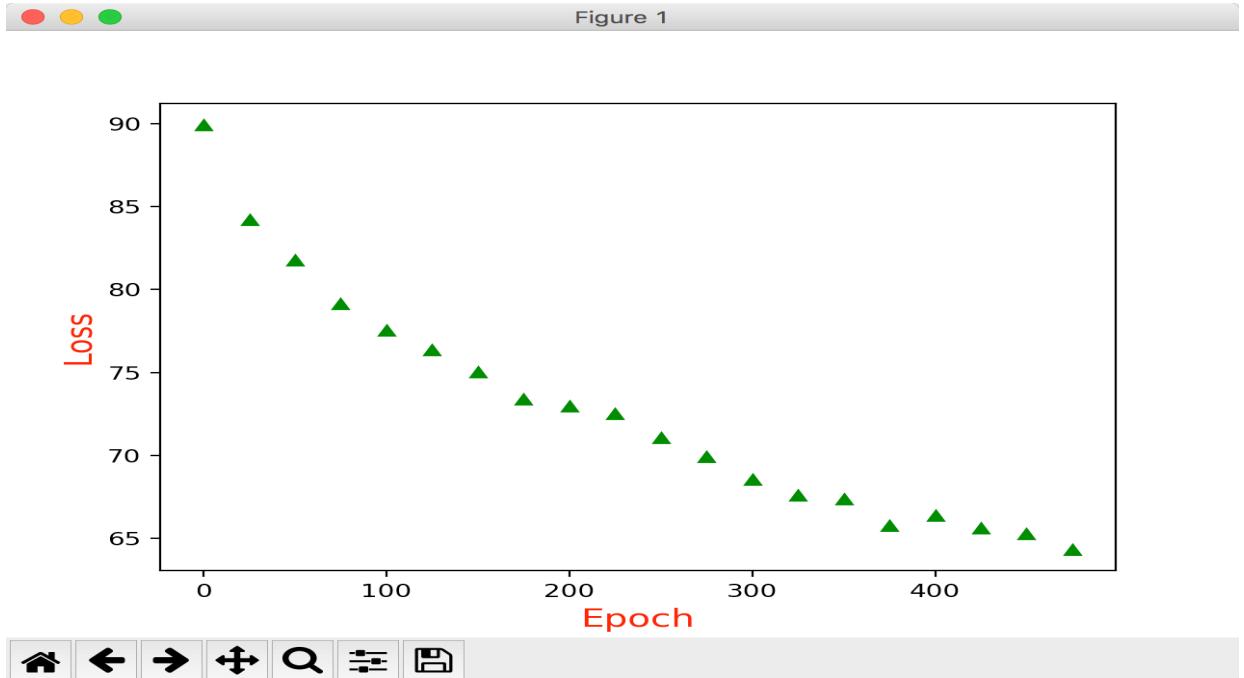


Varying filter size

Constants: Learning rate = 0.001.

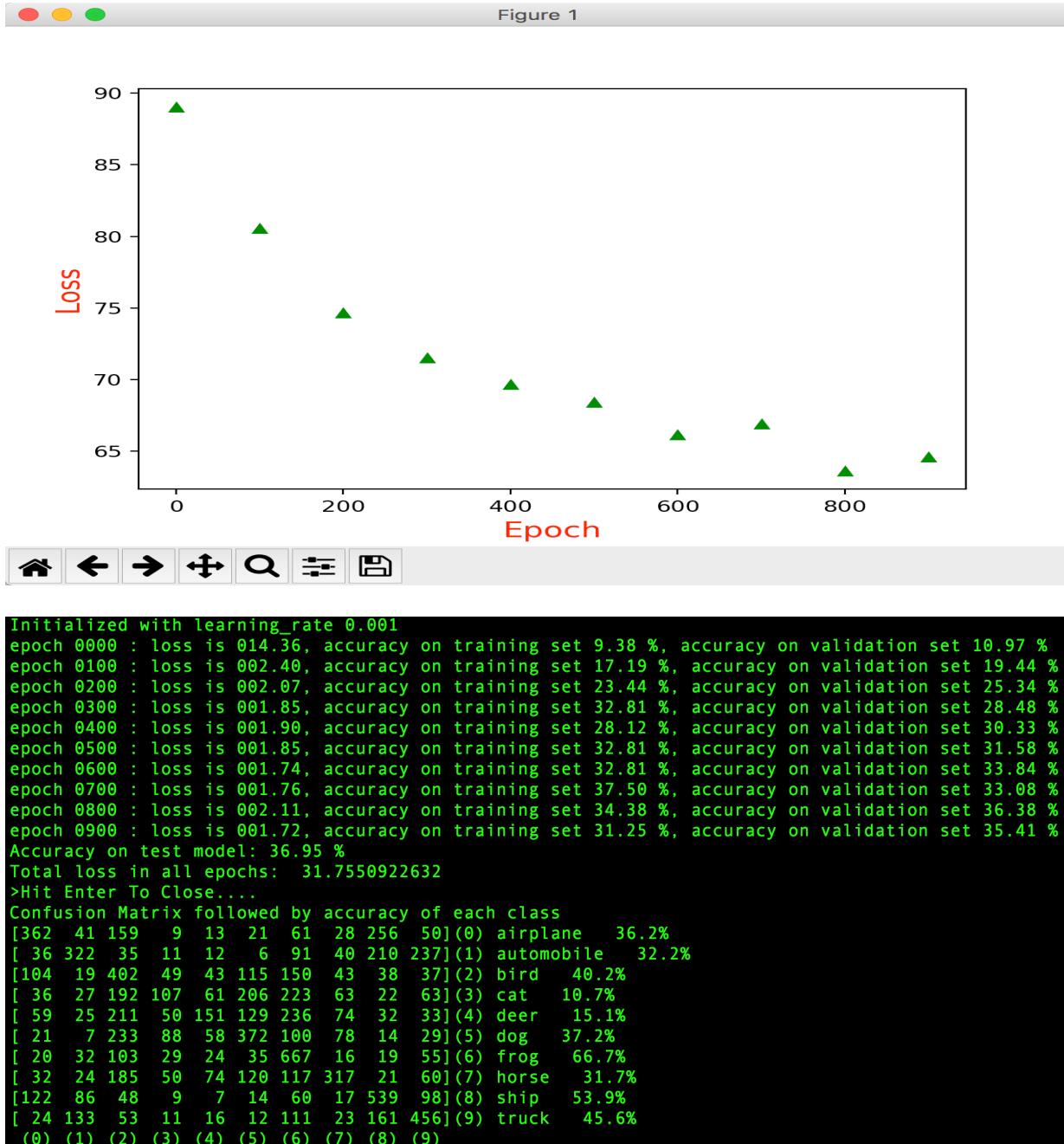
Number of epochs = 1000

1.) Filter size is 2x2 for both layers.



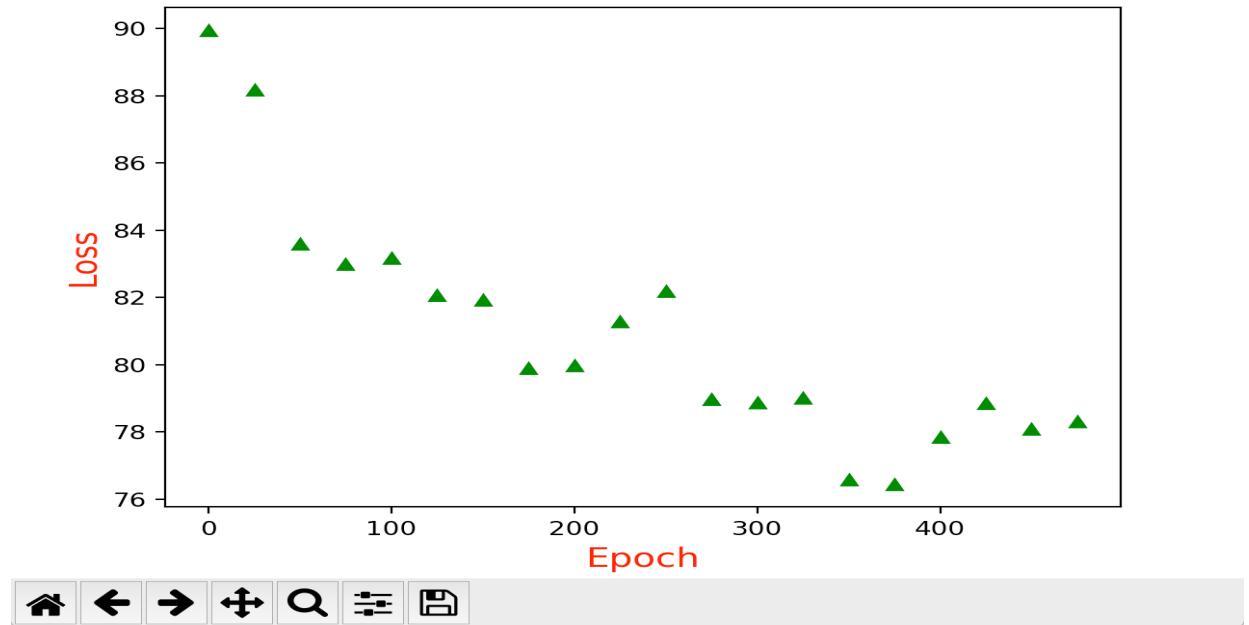
```
Initialized with learning_rate 0.001
epoch 0000 : loss is 020.89, accuracy on training set 17.19 %, accuracy on validation set 10.06 %
epoch 0025 : loss is 003.50, accuracy on training set 17.19 %, accuracy on validation set 15.78 %
epoch 0050 : loss is 002.56, accuracy on training set 17.19 %, accuracy on validation set 18.24 %
epoch 0075 : loss is 002.30, accuracy on training set 28.12 %, accuracy on validation set 20.88 %
epoch 0100 : loss is 002.31, accuracy on training set 17.19 %, accuracy on validation set 22.46 %
epoch 0125 : loss is 002.14, accuracy on training set 21.88 %, accuracy on validation set 23.65 %
epoch 0150 : loss is 002.10, accuracy on training set 28.12 %, accuracy on validation set 24.98 %
epoch 0175 : loss is 002.12, accuracy on training set 17.19 %, accuracy on validation set 26.62 %
epoch 0200 : loss is 002.13, accuracy on training set 25.00 %, accuracy on validation set 27.04 %
epoch 0225 : loss is 001.97, accuracy on training set 18.75 %, accuracy on validation set 27.50 %
epoch 0250 : loss is 001.85, accuracy on training set 28.12 %, accuracy on validation set 28.95 %
epoch 0275 : loss is 001.91, accuracy on training set 26.56 %, accuracy on validation set 30.09 %
epoch 0300 : loss is 001.77, accuracy on training set 43.75 %, accuracy on validation set 31.47 %
epoch 0325 : loss is 001.77, accuracy on training set 42.19 %, accuracy on validation set 32.43 %
epoch 0350 : loss is 001.97, accuracy on training set 31.25 %, accuracy on validation set 32.65 %
epoch 0375 : loss is 001.92, accuracy on training set 35.94 %, accuracy on validation set 34.25 %
epoch 0400 : loss is 001.87, accuracy on training set 34.38 %, accuracy on validation set 33.65 %
epoch 0425 : loss is 001.84, accuracy on training set 25.00 %, accuracy on validation set 34.40 %
epoch 0450 : loss is 001.79, accuracy on training set 39.06 %, accuracy on validation set 34.75 %
epoch 0475 : loss is 001.76, accuracy on training set 35.94 %, accuracy on validation set 35.69 %
Accuracy on test model: 38.01 %
Total loss in all epochs: 60.4734382629
>Hit Enter To Close.....
Confusion Matrix followed by accuracy of each class
[[437 84 141 12 30 34 75 45 103 39](0) airplane 43.7%
[ 26 562 7 12 10 15 130 28 75 135](1) automobile 56.2%
[ 77 28 385 59 32 107 164 91 24 33](2) bird 38.5%
[ 31 70 122 106 36 252 174 150 14 45](3) cat 10.6%
[ 29 46 229 66 70 138 198 177 24 23](4) deer 7.0%
[ 12 20 138 80 40 387 95 191 15 22](5) dog 38.7%
[ 22 103 78 44 20 54 561 56 9 53](6) frog 56.1%
[ 13 60 75 51 24 129 78 479 13 78](7) horse 47.9%
[183 169 34 9 7 32 62 23 421 60](8) ship 42.1%
[ 40 303 16 12 9 18 92 39 78 393](9) truck 39.3%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)
```

2.) Filter size is 5x5 for both layers.



3.) Filter size 10x10 for both layers.

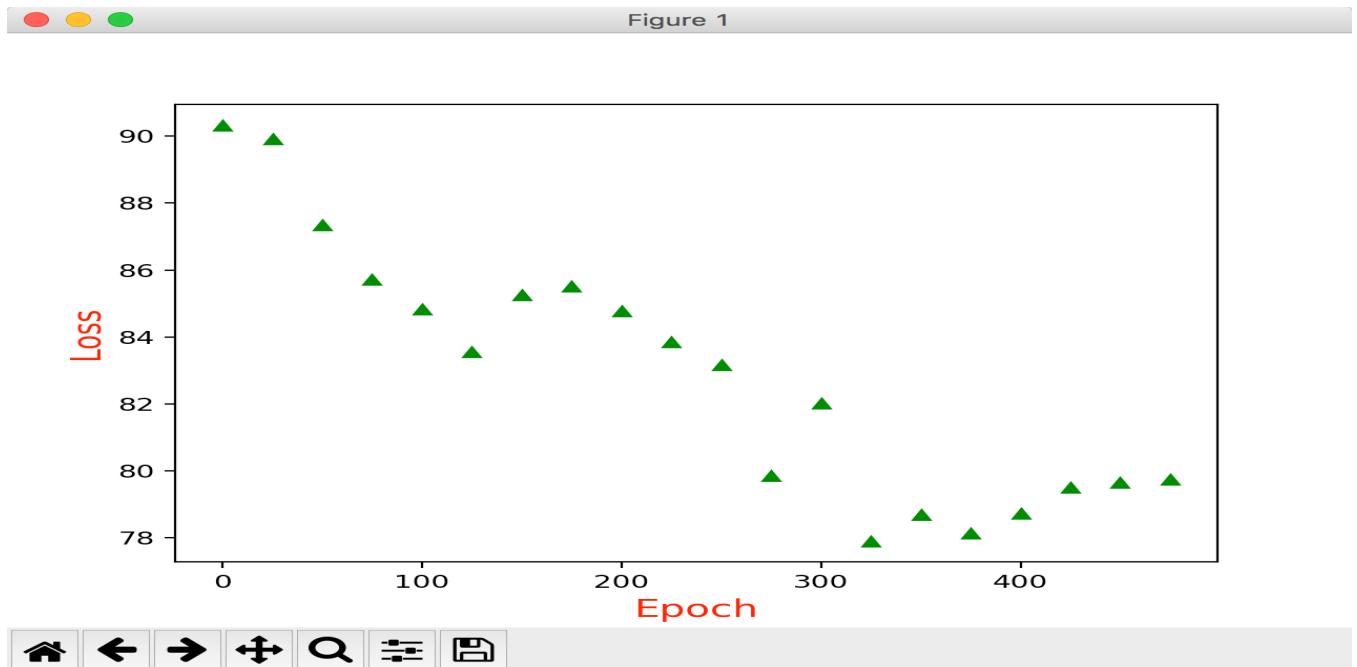
Figure 1



```

Initialized with learning_rate 0.001
epoch 0000 : loss is 012.58, accuracy on training set 9.38 %, accuracy on validation set 10.05 %
epoch 0025 : loss is 002.30, accuracy on training set 10.94 %, accuracy on validation set 11.82 %
epoch 0050 : loss is 002.28, accuracy on training set 17.19 %, accuracy on validation set 16.41 %
epoch 0075 : loss is 002.26, accuracy on training set 18.75 %, accuracy on validation set 17.02 %
epoch 0100 : loss is 002.20, accuracy on training set 10.94 %, accuracy on validation set 16.82 %
epoch 0125 : loss is 002.20, accuracy on training set 7.81 %, accuracy on validation set 17.94 %
epoch 0150 : loss is 002.18, accuracy on training set 25.00 %, accuracy on validation set 18.07 %
epoch 0175 : loss is 002.03, accuracy on training set 26.56 %, accuracy on validation set 20.11 %
epoch 0200 : loss is 001.97, accuracy on training set 26.56 %, accuracy on validation set 20.02 %
epoch 0225 : loss is 002.16, accuracy on training set 14.06 %, accuracy on validation set 18.72 %
epoch 0250 : loss is 002.18, accuracy on training set 18.75 %, accuracy on validation set 17.82 %
epoch 0275 : loss is 002.12, accuracy on training set 23.44 %, accuracy on validation set 21.04 %
epoch 0300 : loss is 002.16, accuracy on training set 12.50 %, accuracy on validation set 21.14 %
epoch 0325 : loss is 002.15, accuracy on training set 26.56 %, accuracy on validation set 21.00 %
epoch 0350 : loss is 001.97, accuracy on training set 20.31 %, accuracy on validation set 23.42 %
epoch 0375 : loss is 001.90, accuracy on training set 26.56 %, accuracy on validation set 23.56 %
epoch 0400 : loss is 001.94, accuracy on training set 18.75 %, accuracy on validation set 22.15 %
epoch 0425 : loss is 002.23, accuracy on training set 15.62 %, accuracy on validation set 21.15 %
epoch 0450 : loss is 002.09, accuracy on training set 23.44 %, accuracy on validation set 21.91 %
epoch 0475 : loss is 002.12, accuracy on training set 18.75 %, accuracy on validation set 21.70 %
Accuracy on test model: 23.22 %
Total loss in all epochs: 53.0105937719
>Hit Enter To Close...
Confusion Matrix followed by accuracy of each class
[[231  0 123  0  77 125  41  0  87 316](0) airplane  23.1%
 [ 74  0  27  0  47 26  66  0  64 696](1) automobile  0.0%
 [ 39  0 169  0 124 422 121  1  3 121](2) bird    16.9%
 [ 21  0  98  0 105 454 181  0  3 138](3) cat    0.0%
 [ 50  0 179  0 179 369 124  0  6 93](4) deer    17.9%
 [ 21  0 117  0  72 605 123  0  2 60](5) dog    60.5%
 [ 22  0  79  0 142 246 301  0  5 205](6) frog   30.1%
 [ 86  0 231  0 115 366  69  1  3 129](7) horse   0.1%
 [[141  0  33  0  38 24  29  0 111 624](8) ship    11.1%
 [ 50  0  31  0  53 23  67  0  51 725](9) truck   72.5%
 (0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]
```

4.) Filter size 20x20 for both layers

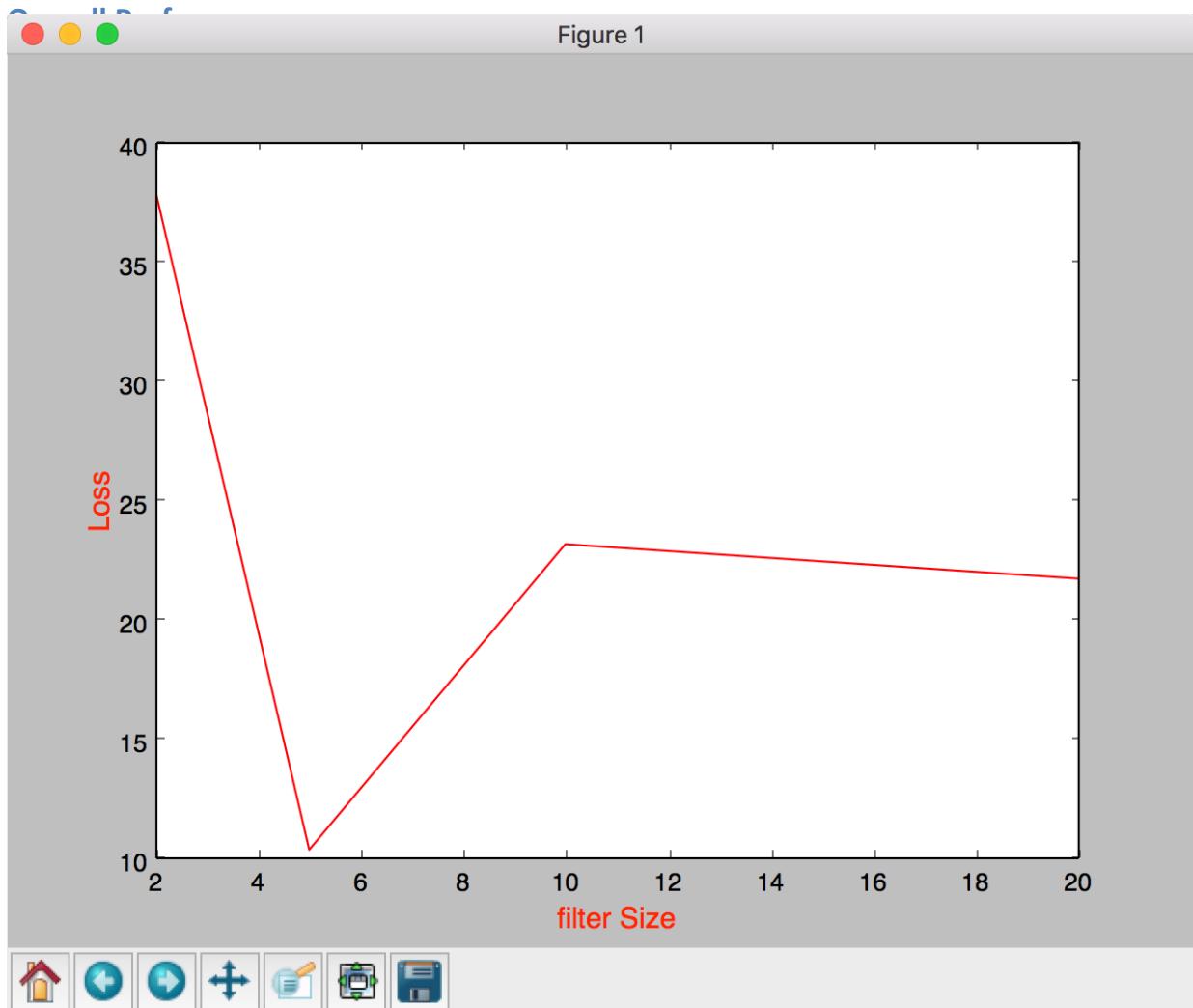


```

Initialized with learning_rate 0.001
epoch 0000 : loss is 007.48, accuracy on training set 12.50 %, accuracy on validation set 9.68 %
epoch 0025 : loss is 002.36, accuracy on training set 9.38 %, accuracy on validation set 10.09 %
epoch 0050 : loss is 002.28, accuracy on training set 15.62 %, accuracy on validation set 12.66 %
epoch 0075 : loss is 002.27, accuracy on training set 17.19 %, accuracy on validation set 14.28 %
epoch 0100 : loss is 002.35, accuracy on training set 12.50 %, accuracy on validation set 15.17 %
epoch 0125 : loss is 002.19, accuracy on training set 23.44 %, accuracy on validation set 16.44 %
epoch 0150 : loss is 002.26, accuracy on training set 9.38 %, accuracy on validation set 14.75 %
epoch 0175 : loss is 002.30, accuracy on training set 10.94 %, accuracy on validation set 14.48 %
epoch 0200 : loss is 002.24, accuracy on training set 14.06 %, accuracy on validation set 15.23 %
epoch 0225 : loss is 002.15, accuracy on training set 10.94 %, accuracy on validation set 16.15 %
epoch 0250 : loss is 002.12, accuracy on training set 10.94 %, accuracy on validation set 16.83 %
epoch 0275 : loss is 002.16, accuracy on training set 17.19 %, accuracy on validation set 20.14 %
epoch 0300 : loss is 002.07, accuracy on training set 25.00 %, accuracy on validation set 17.99 %
epoch 0325 : loss is 002.18, accuracy on training set 12.50 %, accuracy on validation set 22.10 %
epoch 0350 : loss is 002.25, accuracy on training set 18.75 %, accuracy on validation set 21.30 %
epoch 0375 : loss is 002.06, accuracy on training set 18.75 %, accuracy on validation set 21.87 %
epoch 0400 : loss is 002.08, accuracy on training set 26.56 %, accuracy on validation set 21.27 %
epoch 0425 : loss is 002.00, accuracy on training set 17.19 %, accuracy on validation set 20.50 %
epoch 0450 : loss is 002.19, accuracy on training set 15.62 %, accuracy on validation set 20.34 %
epoch 0475 : loss is 002.12, accuracy on training set 21.88 %, accuracy on validation set 20.26 %
Accuracy on test model: 21.77 %
Total loss in all epochs: 49.125664711
>Hit Enter To Close....
Confusion Matrix followed by accuracy of each class
[[106  0  61  0 182  0 105  17 391 138](0) airplane  10.6%
 [ 21  0 10  0  67  0 179  0 472 251](1) automobile  0.0%
 [ 21  0 114  0 448  0 247  32 61 77](2) bird  11.4%
 [ 10  0 69  0 432  0 348  20 36 85](3) cat  0.0%
 [ 15  0 87  0 446  0 301  21 33 97](4) deer  44.6%
 [ 3  0 175  1 487  0 219  54 13 48](5) dog  0.0%
 [ 3  0 15  0 269  0 538  6 29 140](6) frog  53.8%
 [ 24  0 132  0 457  0 216  49 35 87](7) horse  4.9%
 [ 24  0 5  0 66  0 78  0 645 182](8) ship  64.5%
 [ 9  0 9  0 76  0 174  0 453 279](9) truck  27.9%
 (0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]

```

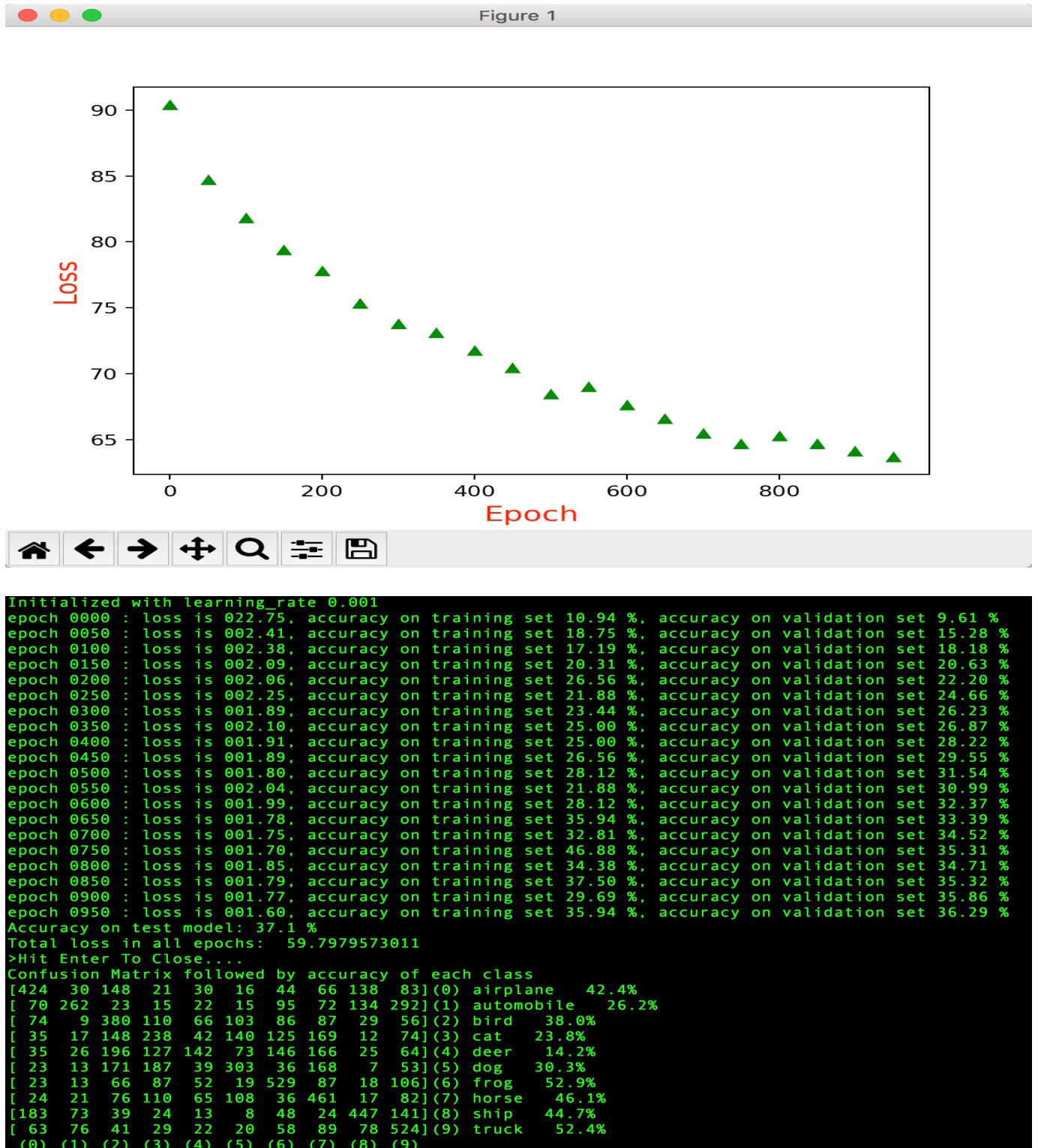
Figure 1



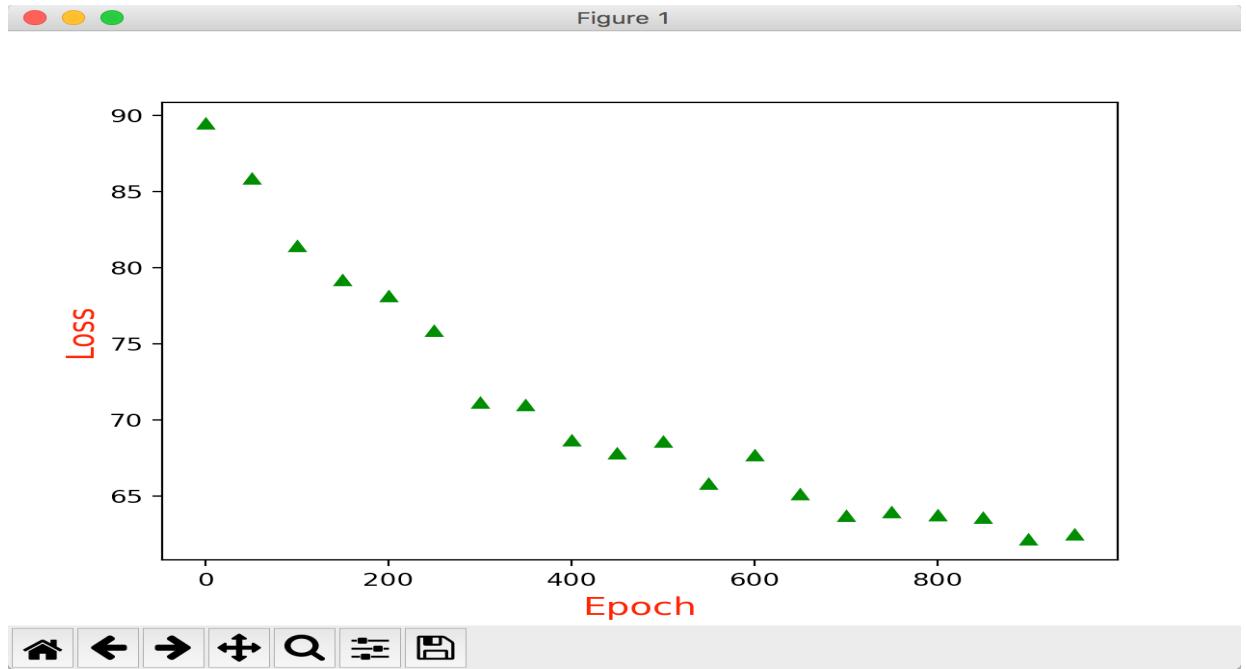
Varying number of filters

Constant: learning rate 0.001, epochs 500, Filter Size 5x5. Epoch = 1000

1.) First layer of size 4 and second layer of size 16



2.) First layer is 6 and second layer is 16



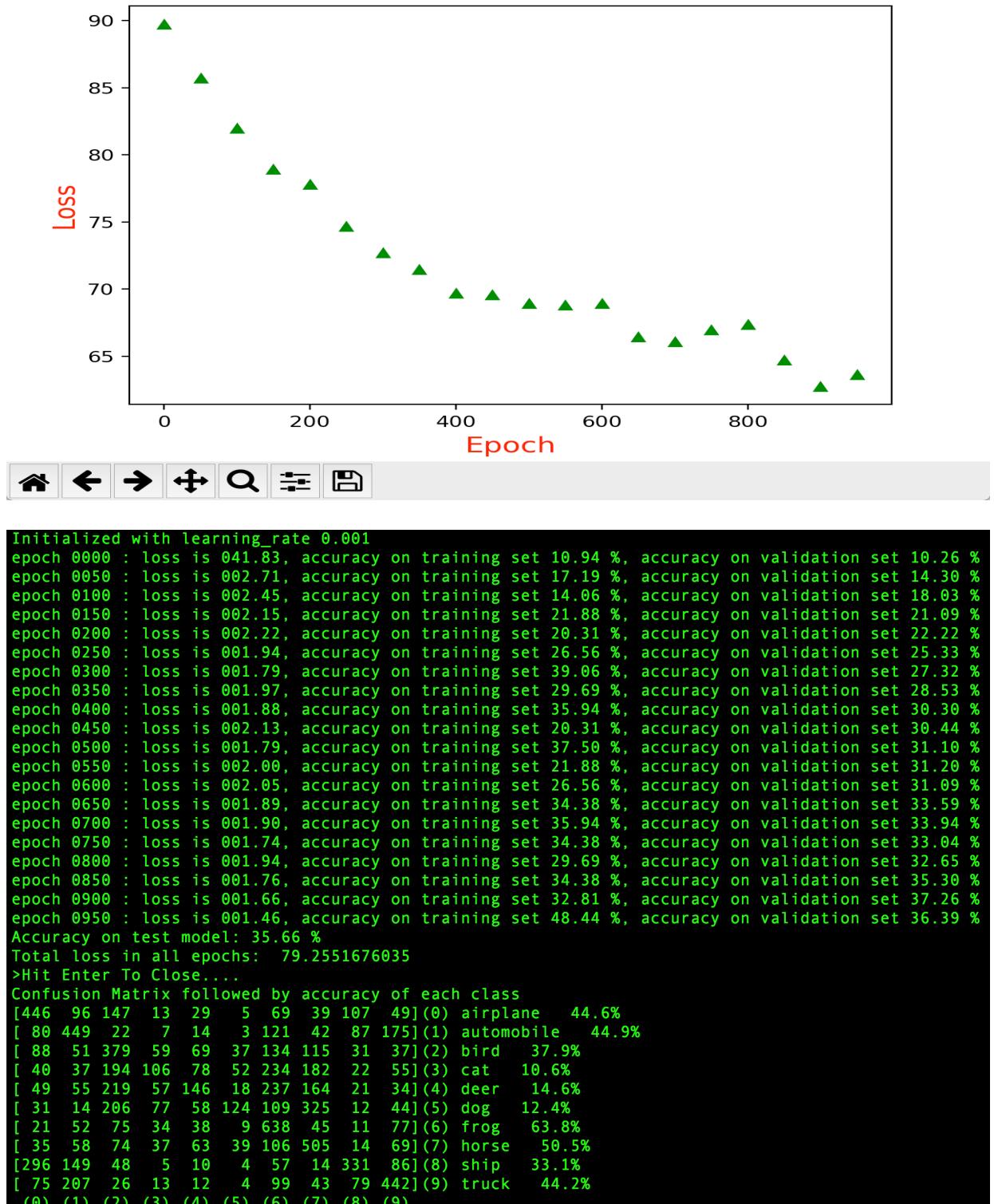
```

Initialized with learning_rate 0.001
epoch 0000 : loss is 010.94, accuracy on training set 12.50 %, accuracy on validation set 10.52 %
epoch 0050 : loss is 002.52, accuracy on training set 10.94 %, accuracy on validation set 14.13 %
epoch 0100 : loss is 002.39, accuracy on training set 12.50 %, accuracy on validation set 18.56 %
epoch 0150 : loss is 002.16, accuracy on training set 25.00 %, accuracy on validation set 20.82 %
epoch 0200 : loss is 002.05, accuracy on training set 18.75 %, accuracy on validation set 21.85 %
epoch 0250 : loss is 001.93, accuracy on training set 26.56 %, accuracy on validation set 24.14 %
epoch 0300 : loss is 001.80, accuracy on training set 39.06 %, accuracy on validation set 28.85 %
epoch 0350 : loss is 001.96, accuracy on training set 34.38 %, accuracy on validation set 29.03 %
epoch 0400 : loss is 001.74, accuracy on training set 29.69 %, accuracy on validation set 31.35 %
epoch 0450 : loss is 001.96, accuracy on training set 34.38 %, accuracy on validation set 32.19 %
epoch 0500 : loss is 001.92, accuracy on training set 32.81 %, accuracy on validation set 31.42 %
epoch 0550 : loss is 001.79, accuracy on training set 34.38 %, accuracy on validation set 34.19 %
epoch 0600 : loss is 001.82, accuracy on training set 28.12 %, accuracy on validation set 32.32 %
epoch 0650 : loss is 001.85, accuracy on training set 23.44 %, accuracy on validation set 34.88 %
epoch 0700 : loss is 001.78, accuracy on training set 43.75 %, accuracy on validation set 36.29 %
epoch 0750 : loss is 001.70, accuracy on training set 39.06 %, accuracy on validation set 36.03 %
epoch 0800 : loss is 001.90, accuracy on training set 29.69 %, accuracy on validation set 36.26 %
epoch 0850 : loss is 001.70, accuracy on training set 34.38 %, accuracy on validation set 36.42 %
epoch 0900 : loss is 001.73, accuracy on training set 40.62 %, accuracy on validation set 37.83 %
epoch 0950 : loss is 001.72, accuracy on training set 37.50 %, accuracy on validation set 37.51 %
Accuracy on test model: 36.36 %
Total loss in all epochs: 47.3529236317
>Hit Enter To Close...
Confusion Matrix followed by accuracy of each class
[[554  3 106 25 24 23 50 41 89 85](0) airplane  55.4%
[138  76 14 35 36 26 119 50 109 397](1) automobile  7.6%
[157  1 317 126 65 105 119 46 25 39](2) bird  31.7%
[ 51  2 130 240 74 169 181 77 17 59](3) cat  24.0%
[ 66  3 162 149 202 99 182 97 10 30](4) deer  20.2%
[ 35  2 151 215 69 341 79 81 4 23](5) dog  34.1%
[ 40  1 59 130 36 32 587 46 4 65](6) frog  58.7%
[ 33  3 66 111 74 162 87 388 11 65](7) horse  38.8%
[301  15 52 21 17 11 46 19 382 136](8) ship  38.2%
[ 87  7 25 46 30 13 106 56 81 549](9) truck  54.9%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)

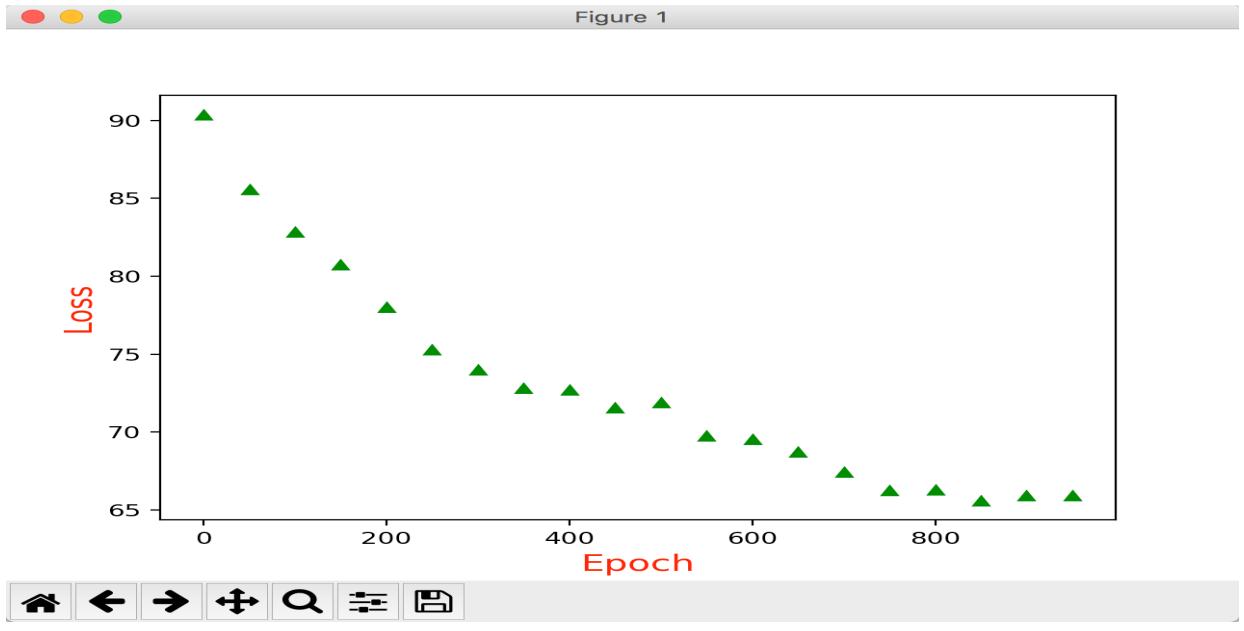
```

3.) First layer is 8 and second layer is 16

Figure 1



4.) First layer is 6 and second layer is 10

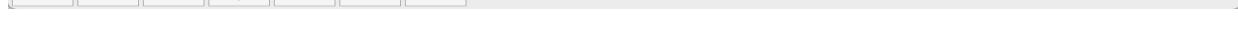
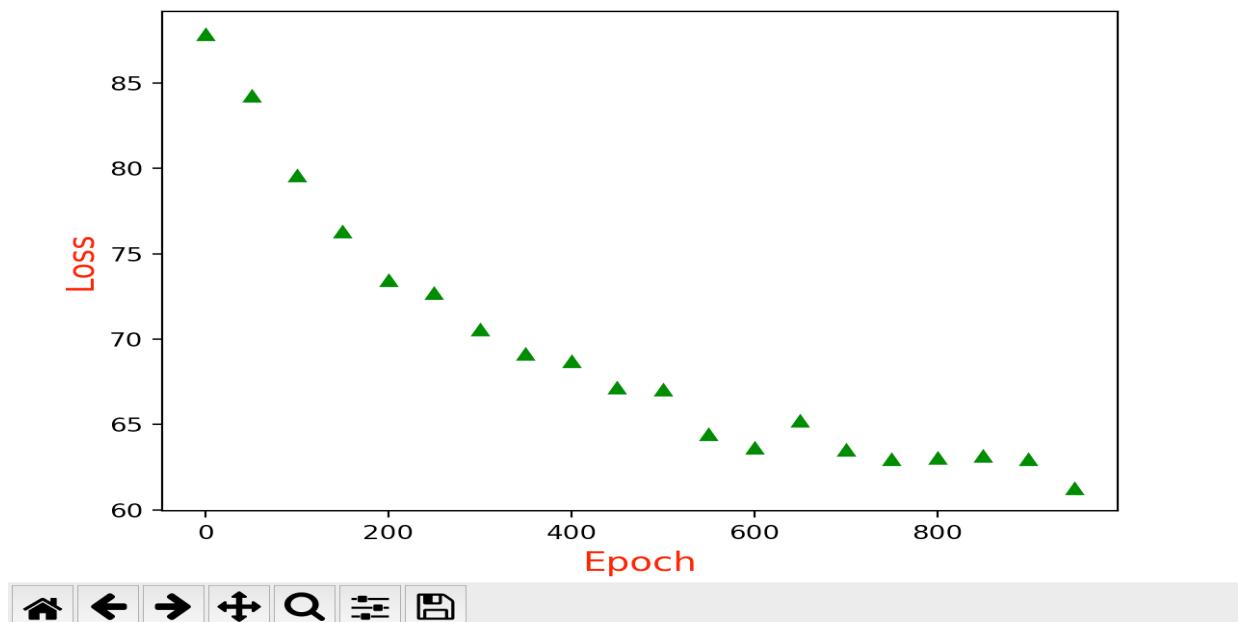


```

Initialized with learning_rate 0.001
epoch 0000 : loss is 018.99, accuracy on training set 14.06 %, accuracy on validation set 9.63 %
epoch 0050 : loss is 002.75, accuracy on training set 15.62 %, accuracy on validation set 14.44 %
epoch 0100 : loss is 002.28, accuracy on training set 17.19 %, accuracy on validation set 17.16 %
epoch 0150 : loss is 002.22, accuracy on training set 21.88 %, accuracy on validation set 19.26 %
epoch 0200 : loss is 002.31, accuracy on training set 23.44 %, accuracy on validation set 21.98 %
epoch 0250 : loss is 002.09, accuracy on training set 21.88 %, accuracy on validation set 24.72 %
epoch 0300 : loss is 001.99, accuracy on training set 29.69 %, accuracy on validation set 26.00 %
epoch 0350 : loss is 002.11, accuracy on training set 28.12 %, accuracy on validation set 27.19 %
epoch 0400 : loss is 001.83, accuracy on training set 34.38 %, accuracy on validation set 27.29 %
epoch 0450 : loss is 002.14, accuracy on training set 17.19 %, accuracy on validation set 28.42 %
epoch 0500 : loss is 001.93, accuracy on training set 28.12 %, accuracy on validation set 28.10 %
epoch 0550 : loss is 002.01, accuracy on training set 29.69 %, accuracy on validation set 30.23 %
epoch 0600 : loss is 001.90, accuracy on training set 31.25 %, accuracy on validation set 30.46 %
epoch 0650 : loss is 002.06, accuracy on training set 21.88 %, accuracy on validation set 31.28 %
epoch 0700 : loss is 001.84, accuracy on training set 37.50 %, accuracy on validation set 32.56 %
epoch 0750 : loss is 001.80, accuracy on training set 25.00 %, accuracy on validation set 33.73 %
epoch 0800 : loss is 001.68, accuracy on training set 39.06 %, accuracy on validation set 33.72 %
epoch 0850 : loss is 001.80, accuracy on training set 35.94 %, accuracy on validation set 34.40 %
epoch 0900 : loss is 001.90, accuracy on training set 31.25 %, accuracy on validation set 34.09 %
epoch 0950 : loss is 001.94, accuracy on training set 31.25 %, accuracy on validation set 34.09 %
Accuracy on test model: 34.07 %
Total loss in all epochs: 57.5635074377
>Hit Enter To Close....
Confusion Matrix followed by accuracy of each class
[[464  75  74  13   8  32  39  66 132  97](0) airplane  46.4%
 [ 74 456   1   4   6  22  54  60  66 257](1) automobile 45.6%
 [173  36 237  24  14 162 122 147  20  65](2) bird    23.7%
 [ 79  56  81  53  11 262 141 215  10  92](3) cat     5.3%
 [100  73 119  37  49 147 167 226  11  71](4) deer    4.9%
 [ 62  18 105  44   9 404  80 213   9  56](5) dog     40.4%
 [ 58  71  44  34  16  74 433 141  14 115](6) frog    43.3%
 [ 56  58  38  21  12 143  65 478   6 123](7) horse    47.8%
 [[233 183  16   7   8  20  32  23 339 139](8) ship    33.9%
 [ 62 244  10   8   1  14  36  57  74 494](9) truck   49.4%
 (0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]
```

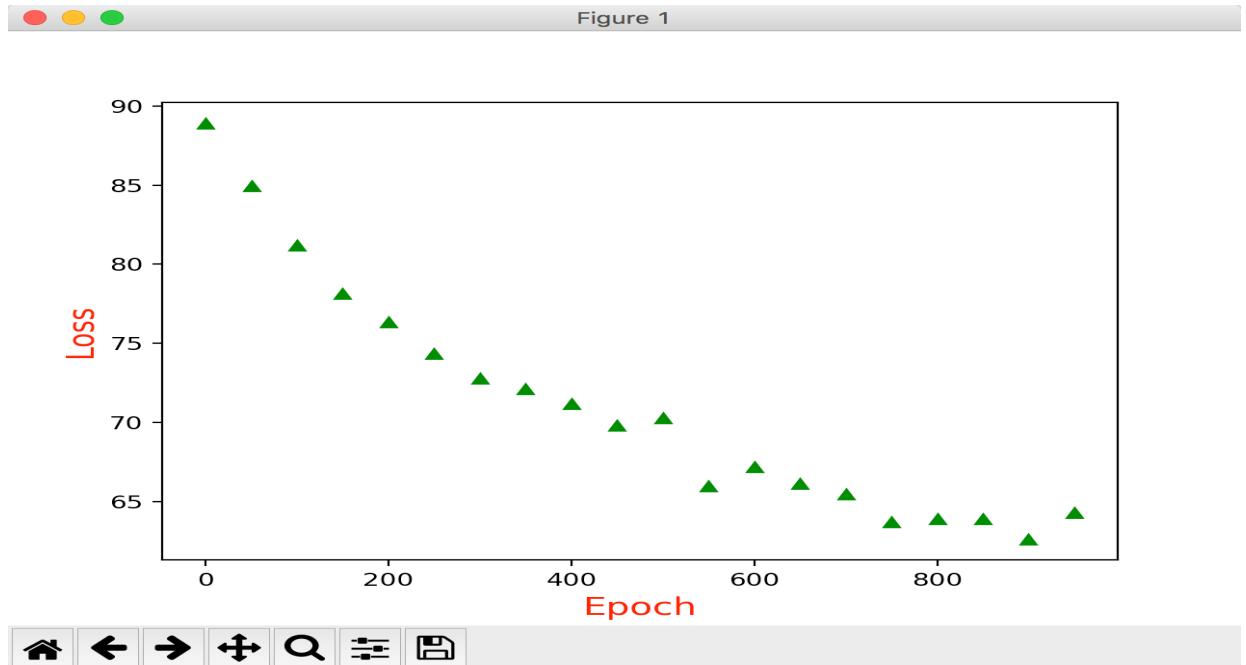
5.) First layer is 6 and second layer is 18

Figure 1



```
Initialized with learning_rate 0.001
epoch 0000 : loss is 014.35, accuracy on training set 7.81 %, accuracy on validation set 12.15 %
epoch 0050 : loss is 002.18, accuracy on training set 23.44 %, accuracy on validation set 15.77 %
epoch 0100 : loss is 002.20, accuracy on training set 20.31 %, accuracy on validation set 20.41 %
epoch 0150 : loss is 001.98, accuracy on training set 32.81 %, accuracy on validation set 23.72 %
epoch 0200 : loss is 002.06, accuracy on training set 25.00 %, accuracy on validation set 26.58 %
epoch 0250 : loss is 001.89, accuracy on training set 29.69 %, accuracy on validation set 27.33 %
epoch 0300 : loss is 001.74, accuracy on training set 37.50 %, accuracy on validation set 29.44 %
epoch 0350 : loss is 001.94, accuracy on training set 34.38 %, accuracy on validation set 30.87 %
epoch 0400 : loss is 001.72, accuracy on training set 35.94 %, accuracy on validation set 31.30 %
epoch 0450 : loss is 001.82, accuracy on training set 35.94 %, accuracy on validation set 32.84 %
epoch 0500 : loss is 001.94, accuracy on training set 34.38 %, accuracy on validation set 32.96 %
epoch 0550 : loss is 001.79, accuracy on training set 39.06 %, accuracy on validation set 35.58 %
epoch 0600 : loss is 001.80, accuracy on training set 40.62 %, accuracy on validation set 36.37 %
epoch 0650 : loss is 001.87, accuracy on training set 29.69 %, accuracy on validation set 34.81 %
epoch 0700 : loss is 002.03, accuracy on training set 17.19 %, accuracy on validation set 36.50 %
epoch 0750 : loss is 001.76, accuracy on training set 34.38 %, accuracy on validation set 37.05 %
epoch 0800 : loss is 001.96, accuracy on training set 35.94 %, accuracy on validation set 36.95 %
epoch 0850 : loss is 001.63, accuracy on training set 37.50 %, accuracy on validation set 36.85 %
epoch 0900 : loss is 001.79, accuracy on training set 40.62 %, accuracy on validation set 37.05 %
epoch 0950 : loss is 001.81, accuracy on training set 43.75 %, accuracy on validation set 38.74 %
Accuracy on test model: 38.49 %
Total loss in all epochs: 50.2623382807
>Hit Enter To Close.....
Confusion Matrix followed by accuracy of each class
[[293 56 197 6 36 22 20 33 305 32][0] (0) airplane 29.3%
[14 469 15 12 27 12 28 31 277 115][1] (1) automobile 46.9%
[44 37 352 48 94 178 58 63 92 34][2] (2) bird 35.2%
[10 63 144 118 111 280 75 78 67 54][3] (3) cat 11.8%
[22 50 139 48 247 201 75 118 70 30][4] (4) deer 24.7%
[8 27 140 67 110 450 41 92 37 28][5] (5) dog 45.0%
[6 69 69 60 92 99 433 33 58 81][6] (6) frog 43.3%
[17 59 72 26 103 152 26 433 45 67][7] (7) horse 43.3%
[71 106 39 8 12 17 18 7 681 41][8] (8) ship 68.1%
[26 231 25 12 22 21 38 34 218 373][9] (9) truck 37.3%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)
```

6.) First layer is 8 and second layer is 20

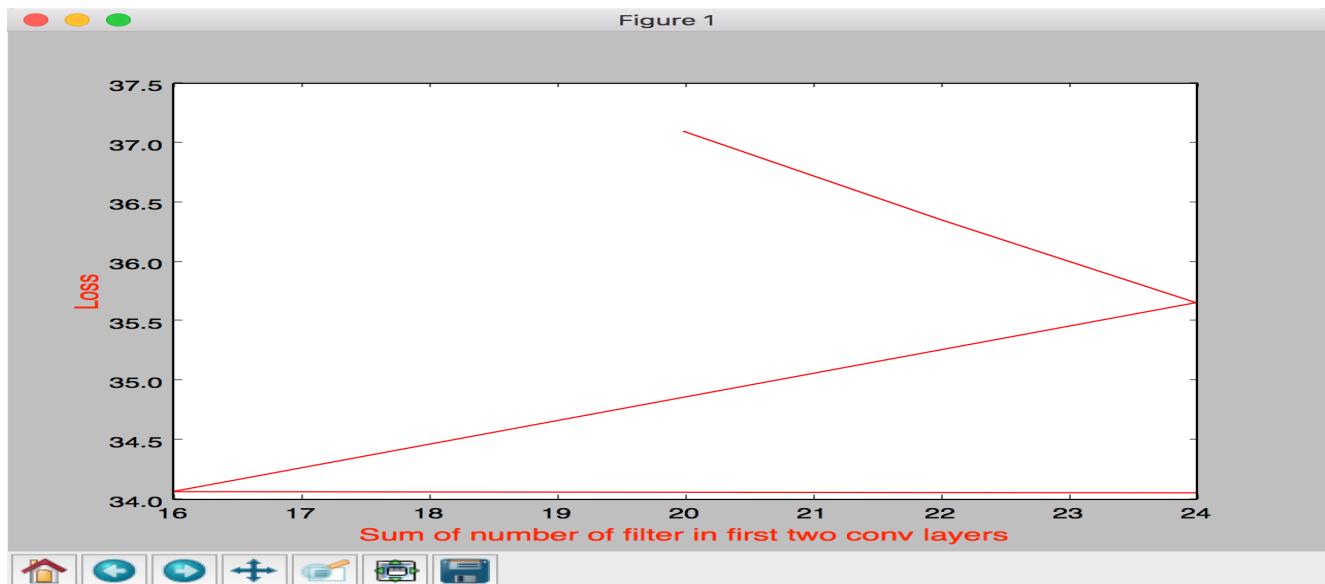


```

Initialized with learning_rate 0.001
epoch 0000 : loss is 016.91, accuracy on training set 6.25 %, accuracy on validation set 11.09 %
epoch 0050 : loss is 002.27, accuracy on training set 15.62 %, accuracy on validation set 15.06 %
epoch 0100 : loss is 002.19, accuracy on training set 21.88 %, accuracy on validation set 18.80 %
epoch 0150 : loss is 002.02, accuracy on training set 29.69 %, accuracy on validation set 21.85 %
epoch 0200 : loss is 001.79, accuracy on training set 37.50 %, accuracy on validation set 23.66 %
epoch 0250 : loss is 002.20, accuracy on training set 17.19 %, accuracy on validation set 25.65 %
epoch 0300 : loss is 001.96, accuracy on training set 14.06 %, accuracy on validation set 27.22 %
epoch 0350 : loss is 002.07, accuracy on training set 20.31 %, accuracy on validation set 27.87 %
epoch 0400 : loss is 001.99, accuracy on training set 26.56 %, accuracy on validation set 28.82 %
epoch 0450 : loss is 002.12, accuracy on training set 26.56 %, accuracy on validation set 30.21 %
epoch 0500 : loss is 002.07, accuracy on training set 32.81 %, accuracy on validation set 29.74 %
epoch 0550 : loss is 001.72, accuracy on training set 45.31 %, accuracy on validation set 34.02 %
epoch 0600 : loss is 001.90, accuracy on training set 35.94 %, accuracy on validation set 32.82 %
epoch 0650 : loss is 002.01, accuracy on training set 29.69 %, accuracy on validation set 33.87 %
epoch 0700 : loss is 001.71, accuracy on training set 34.38 %, accuracy on validation set 34.54 %
epoch 0750 : loss is 001.89, accuracy on training set 32.81 %, accuracy on validation set 36.30 %
epoch 0800 : loss is 001.71, accuracy on training set 34.38 %, accuracy on validation set 36.11 %
epoch 0850 : loss is 001.60, accuracy on training set 42.19 %, accuracy on validation set 36.09 %
epoch 0900 : loss is 001.74, accuracy on training set 42.19 %, accuracy on validation set 37.39 %
epoch 0950 : loss is 001.76, accuracy on training set 34.38 %, accuracy on validation set 35.70 %
Accuracy on test model: 34.6 %
Total loss in all epochs: 53.6267957687
>Hit Enter To Close....
Confusion Matrix followed by accuracy of each class
[[295 232 96 5 40 11 95 46 132 48](0) airplane 29.5%
[10 642 4 2 27 1 130 15 47 122](1) automobile 64.2%
[58 70 206 44 174 53 213 103 32 47](2) bird 20.6%
[15 81 56 106 182 68 263 135 10 84](3) cat 10.6%
[18 101 32 31 314 28 263 146 10 57](4) deer 31.4%
[4 41 70 85 199 185 140 226 3 47](5) dog 18.5%
[11 83 24 20 88 8 697 16 6 47](6) frog 69.7%
[10 98 26 27 166 38 95 416 9 115](7) horse 41.6%
[56 459 20 4 23 1 121 10 231 75](8) ship 23.1%
[8 427 6 3 44 2 70 28 44 368](9) truck 36.8%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]

```

Overall Performance

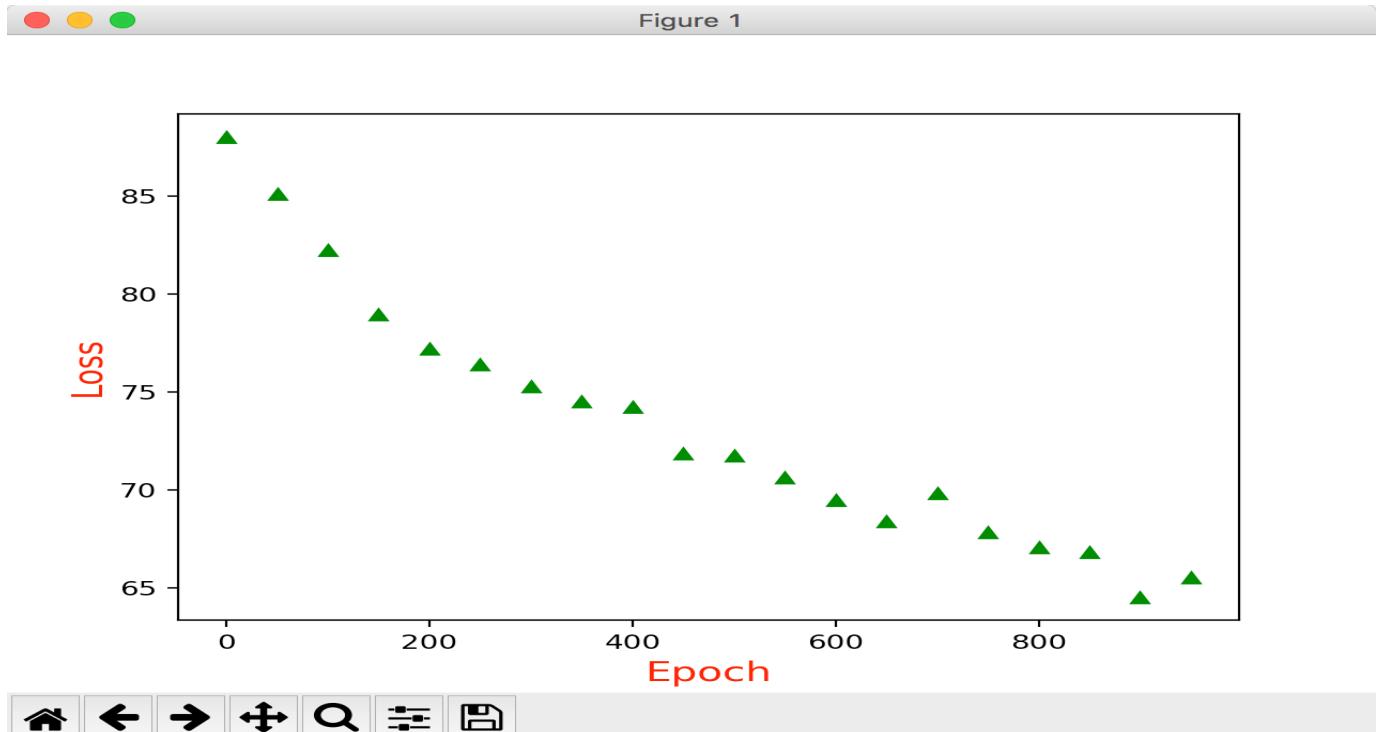


Varying size of fully connected Layers

Constant

Learning constant: 0.001. First layer has 6 5x5 filters and second layer has 16 5x5 filters as mentioned in the homework assignment. Epoch = 1000

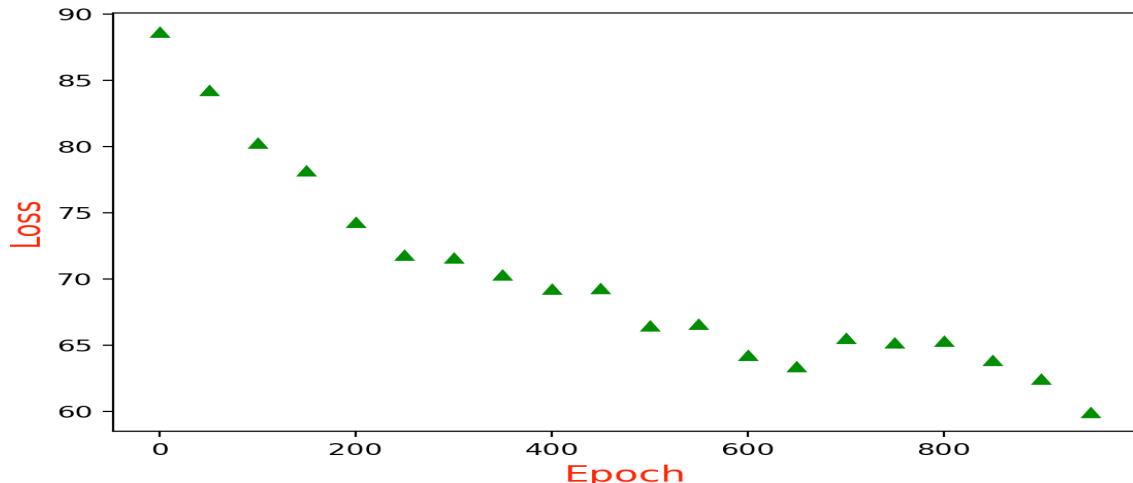
1.) Both fully connected layers of size 50



```
Initialized with learning_rate 0.001
epoch 0000 : loss is 017.55, accuracy on training set 9.38 %, accuracy on validation set 12.00 %
epoch 0050 : loss is 002.27, accuracy on training set 17.19 %, accuracy on validation set 14.90 %
epoch 0100 : loss is 002.35, accuracy on training set 14.06 %, accuracy on validation set 17.79 %
epoch 0150 : loss is 002.09, accuracy on training set 25.00 %, accuracy on validation set 21.05 %
epoch 0200 : loss is 002.05, accuracy on training set 25.00 %, accuracy on validation set 22.81 %
epoch 0250 : loss is 002.14, accuracy on training set 28.12 %, accuracy on validation set 23.61 %
epoch 0300 : loss is 001.99, accuracy on training set 23.44 %, accuracy on validation set 24.73 %
epoch 0350 : loss is 001.99, accuracy on training set 28.12 %, accuracy on validation set 25.49 %
epoch 0400 : loss is 002.12, accuracy on training set 23.44 %, accuracy on validation set 25.78 %
epoch 0450 : loss is 001.98, accuracy on training set 29.69 %, accuracy on validation set 28.15 %
epoch 0500 : loss is 001.97, accuracy on training set 29.69 %, accuracy on validation set 28.25 %
epoch 0550 : loss is 001.92, accuracy on training set 29.69 %, accuracy on validation set 29.37 %
epoch 0600 : loss is 001.80, accuracy on training set 31.25 %, accuracy on validation set 30.52 %
epoch 0650 : loss is 001.89, accuracy on training set 29.69 %, accuracy on validation set 31.62 %
epoch 0700 : loss is 001.84, accuracy on training set 31.25 %, accuracy on validation set 30.19 %
epoch 0750 : loss is 001.88, accuracy on training set 26.56 %, accuracy on validation set 32.18 %
epoch 0800 : loss is 001.78, accuracy on training set 37.50 %, accuracy on validation set 32.96 %
epoch 0850 : loss is 001.89, accuracy on training set 29.69 %, accuracy on validation set 33.19 %
epoch 0900 : loss is 001.78, accuracy on training set 28.12 %, accuracy on validation set 35.49 %
epoch 0950 : loss is 001.79, accuracy on training set 39.06 %, accuracy on validation set 34.50 %
Accuracy on test model: 34.7 %
Total loss in all epochs:  55.0560988188
>Hit Enter To Close.....
Confusion Matrix followed by accuracy of each class
[[405 73 131 11 36 80 45 18 147 54](0) airplane 40.5%
[ 39 373 9 17 22 41 94 19 92 294](1) automobile 37.3%
[ 85 26 263 49 66 289 122 23 23 54](2) bird 26.3%
[ 26 43 74 99 64 376 190 40 12 76](3) cat 9.9%
[ 34 51 103 65 137 288 219 38 21 44](4) deer 13.7%
[ 14 14 103 70 46 566 103 39 5 40](5) dog 56.6%
[ 21 30 47 43 33 144 559 13 8 102](6) frog 55.9%
[ 28 36 39 59 72 358 86 208 5 109](7) horse 20.8%
[[186 186 39 10 19 37 56 4 387 76](8) ship 38.7%
[ 61 213 10 9 10 45 91 13 75 473](9) truck 47.3%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]
```

2.) Fully connected Layer 1 = 50 and Fully connected Layer 2 = 100

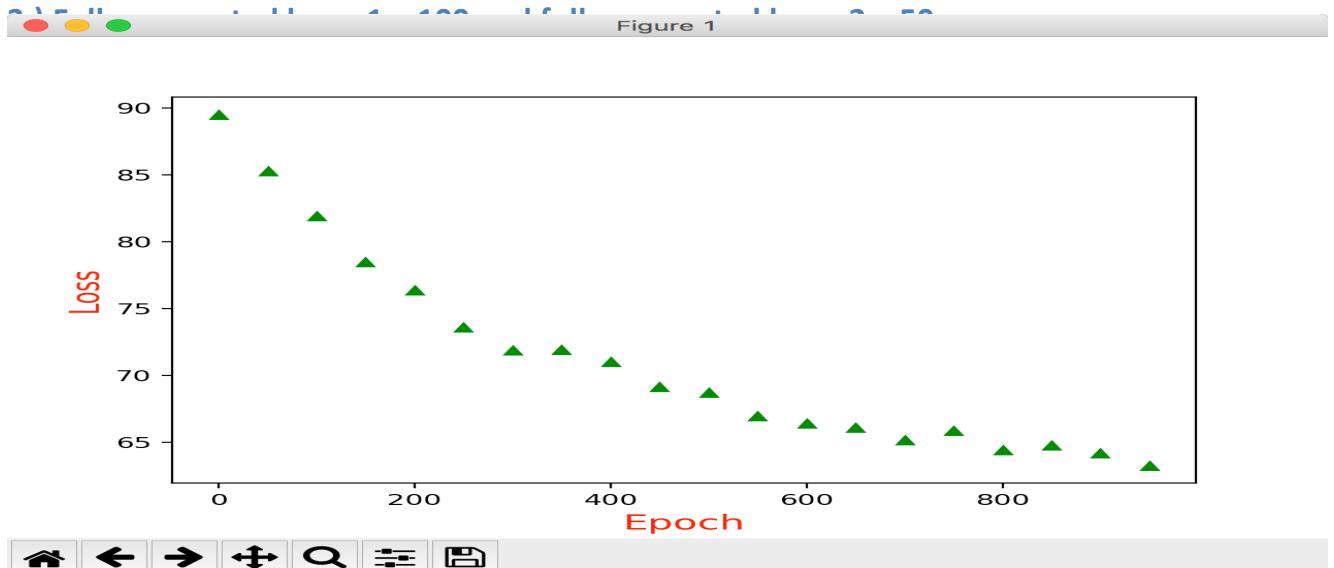
Figure 1



```

Initialized with learning_rate 0.001
epoch 0000 : loss is 015.39, accuracy on training set 12.50 %, accuracy on validation set 11.36 %
epoch 0050 : loss is 002.46, accuracy on training set 15.62 %, accuracy on validation set 15.77 %
epoch 0100 : loss is 002.32, accuracy on training set 20.31 %, accuracy on validation set 19.74 %
epoch 0150 : loss is 002.14, accuracy on training set 20.31 %, accuracy on validation set 21.82 %
epoch 0200 : loss is 002.01, accuracy on training set 28.12 %, accuracy on validation set 25.72 %
epoch 0250 : loss is 001.86, accuracy on training set 31.25 %, accuracy on validation set 28.21 %
epoch 0300 : loss is 001.88, accuracy on training set 29.69 %, accuracy on validation set 28.43 %
epoch 0350 : loss is 001.94, accuracy on training set 31.25 %, accuracy on validation set 29.71 %
epoch 0400 : loss is 001.89, accuracy on training set 29.69 %, accuracy on validation set 30.75 %
epoch 0450 : loss is 001.80, accuracy on training set 35.94 %, accuracy on validation set 30.70 %
epoch 0500 : loss is 001.63, accuracy on training set 31.25 %, accuracy on validation set 33.55 %
epoch 0550 : loss is 001.88, accuracy on training set 29.69 %, accuracy on validation set 33.40 %
epoch 0600 : loss is 001.79, accuracy on training set 39.06 %, accuracy on validation set 35.77 %
epoch 0650 : loss is 001.83, accuracy on training set 35.94 %, accuracy on validation set 36.61 %
epoch 0700 : loss is 001.80, accuracy on training set 35.94 %, accuracy on validation set 34.47 %
epoch 0750 : loss is 001.57, accuracy on training set 48.44 %, accuracy on validation set 34.81 %
epoch 0800 : loss is 001.96, accuracy on training set 21.88 %, accuracy on validation set 34.69 %
epoch 0850 : loss is 001.65, accuracy on training set 40.62 %, accuracy on validation set 36.15 %
epoch 0900 : loss is 001.66, accuracy on training set 35.94 %, accuracy on validation set 37.56 %
epoch 0950 : loss is 001.59, accuracy on training set 45.31 %, accuracy on validation set 40.06 %
Accuracy on test model: 39.17 %
Total loss in all epochs: 51.0525028706
>Hit Enter To Close...
Confusion Matrix followed by accuracy of each class
[[378 38 99 1 133 15 56 18 231 31](0) airplane 37.8%
[ 50 405 6 6 83 2 115 14 165 154](1) automobile 40.5%
[ 75 20 206 20 381 106 100 19 57 16](2) bird 20.6%
[ 23 35 66 42 454 138 139 41 35 27](3) cat 4.2%
[ 21 32 30 16 613 42 137 50 35 24](4) deer 61.3%
[ 9 12 66 44 421 267 58 86 18 19](5) dog 26.7%
[ 16 50 21 18 202 33 611 9 22 18](6) frog 61.1%
[ 13 25 17 27 377 87 57 322 20 55](7) horse 32.2%
[100 80 17 6 64 11 32 6 619 65](8) ship 61.9%
[ 35 149 6 10 98 9 75 29 135 454](9) truck 45.4%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)

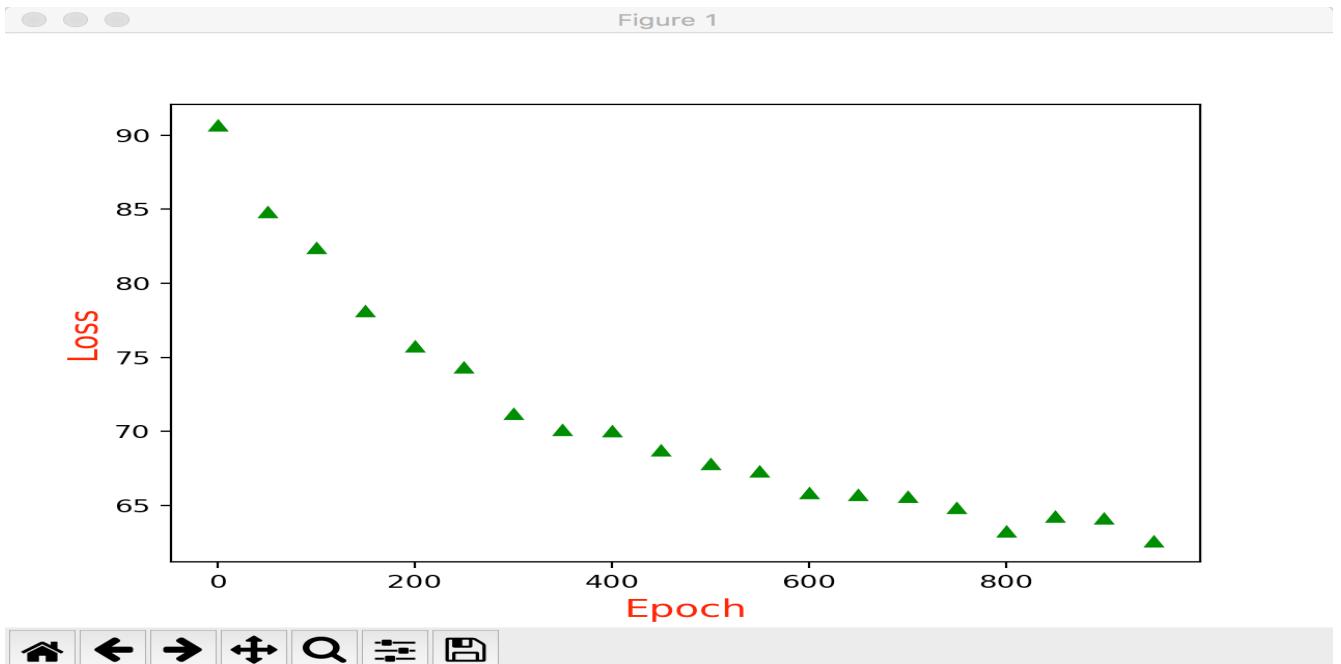
```



```

Initialized with learning_rate 0.001
epoch 0000 : loss is 025.07, accuracy on training set 10.94 %, accuracy on validation set 10.49 %
epoch 0050 : loss is 002.50, accuracy on training set 10.94 %, accuracy on validation set 14.71 %
epoch 0100 : loss is 002.29, accuracy on training set 17.19 %, accuracy on validation set 18.10 %
epoch 0150 : loss is 002.18, accuracy on training set 21.88 %, accuracy on validation set 21.52 %
epoch 0200 : loss is 002.15, accuracy on training set 17.19 %, accuracy on validation set 23.63 %
epoch 0250 : loss is 002.00, accuracy on training set 21.88 %, accuracy on validation set 26.41 %
epoch 0300 : loss is 002.12, accuracy on training set 25.00 %, accuracy on validation set 28.12 %
epoch 0350 : loss is 002.13, accuracy on training set 23.44 %, accuracy on validation set 28.11 %
epoch 0400 : loss is 001.98, accuracy on training set 31.25 %, accuracy on validation set 29.00 %
epoch 0450 : loss is 001.81, accuracy on training set 31.25 %, accuracy on validation set 30.88 %
epoch 0500 : loss is 001.90, accuracy on training set 23.44 %, accuracy on validation set 31.30 %
epoch 0550 : loss is 001.74, accuracy on training set 32.81 %, accuracy on validation set 33.05 %
epoch 0600 : loss is 002.04, accuracy on training set 29.69 %, accuracy on validation set 33.61 %
epoch 0650 : loss is 001.94, accuracy on training set 29.69 %, accuracy on validation set 33.92 %
epoch 0700 : loss is 001.79, accuracy on training set 35.94 %, accuracy on validation set 34.87 %
epoch 0750 : loss is 001.81, accuracy on training set 35.94 %, accuracy on validation set 34.14 %
epoch 0800 : loss is 001.88, accuracy on training set 25.00 %, accuracy on validation set 35.59 %
epoch 0850 : loss is 001.64, accuracy on training set 35.94 %, accuracy on validation set 35.27 %
epoch 0900 : loss is 001.66, accuracy on training set 42.19 %, accuracy on validation set 35.84 %
epoch 0950 : loss is 001.77, accuracy on training set 35.94 %, accuracy on validation set 36.77 %
Accuracy on test model: 36.9 %
Total loss in all epochs: 62.4083821774
>Hit Enter To Close...
Confusion Matrix followed by accuracy of each class
[[441 36 117 33 10 6 41 71 183 62](0) airplane 44.1%
 [35 302 8 17 21 3 106 64 156 288](1) automobile 30.2%
 [99 13 308 126 54 77 80 158 40 45](2) bird 30.8%
 [25 20 121 247 41 110 156 197 24 59](3) cat 24.7%
 [40 22 93 172 117 45 145 281 28 57](4) deer 11.7%
 [18 11 154 196 39 228 62 251 6 35](5) dog 22.8%
 [24 23 48 128 43 13 497 105 40 79](6) frog 49.7%
 [19 30 51 96 42 80 45 563 15 59](7) horse 56.3%
 [[134 104 42 22 15 4 37 26 499 117](8) ship 49.9%
 [48 149 16 38 16 11 74 66 94 488](9) truck 48.8%
 (0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]
```

4.) Fully connected layer 1 = 150 and fully connected layer 2 = 150



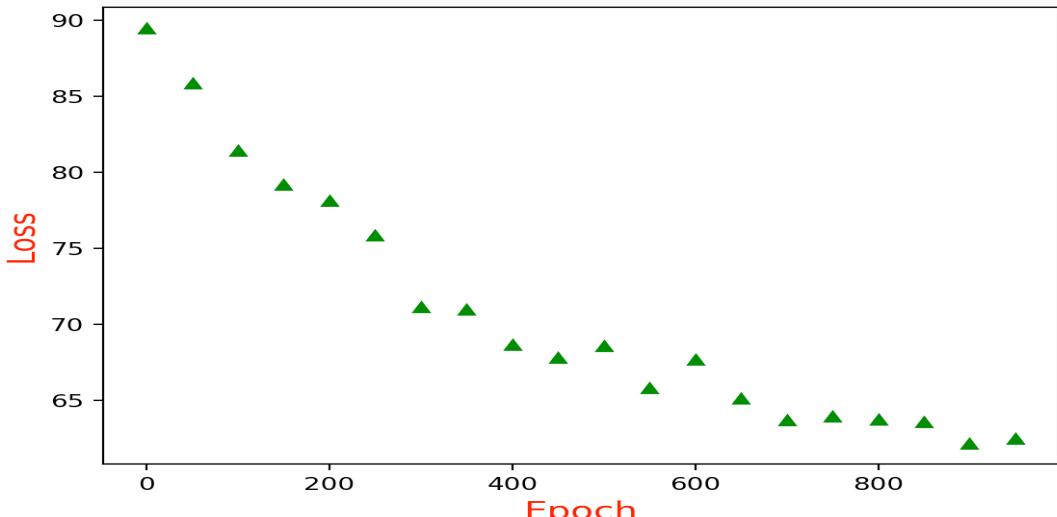
```

Initialized with learning_rate 0.001
epoch 000 : loss is 018.77, accuracy on training set 10.94 %, accuracy on validation set 9.34 %
epoch 0050 : loss is 002.56, accuracy on training set 14.06 %, accuracy on validation set 15.20 %
epoch 0100 : loss is 002.56, accuracy on training set 10.94 %, accuracy on validation set 17.61 %
epoch 0150 : loss is 002.06, accuracy on training set 29.69 %, accuracy on validation set 21.90 %
epoch 0200 : loss is 002.15, accuracy on training set 18.75 %, accuracy on validation set 24.26 %
epoch 0250 : loss is 001.90, accuracy on training set 29.69 %, accuracy on validation set 25.69 %
epoch 0300 : loss is 001.90, accuracy on training set 39.06 %, accuracy on validation set 28.83 %
epoch 0350 : loss is 001.94, accuracy on training set 28.12 %, accuracy on validation set 29.91 %
epoch 0400 : loss is 002.07, accuracy on training set 34.38 %, accuracy on validation set 29.97 %
epoch 0450 : loss is 001.95, accuracy on training set 34.38 %, accuracy on validation set 31.27 %
epoch 0500 : loss is 001.77, accuracy on training set 35.94 %, accuracy on validation set 32.19 %
epoch 0550 : loss is 001.72, accuracy on training set 42.19 %, accuracy on validation set 32.70 %
epoch 0600 : loss is 001.92, accuracy on training set 28.12 %, accuracy on validation set 34.14 %
epoch 0650 : loss is 001.54, accuracy on training set 43.75 %, accuracy on validation set 34.30 %
epoch 0700 : loss is 001.69, accuracy on training set 31.25 %, accuracy on validation set 34.41 %
epoch 0750 : loss is 001.91, accuracy on training set 26.56 %, accuracy on validation set 35.16 %
epoch 0800 : loss is 001.66, accuracy on training set 37.50 %, accuracy on validation set 36.74 %
epoch 0850 : loss is 001.98, accuracy on training set 32.81 %, accuracy on validation set 35.73 %
epoch 0900 : loss is 001.81, accuracy on training set 34.38 %, accuracy on validation set 35.88 %
epoch 0950 : loss is 001.83, accuracy on training set 29.69 %, accuracy on validation set 37.41 %
Accuracy on test model: 37.84 %
Total loss in all epochs: 55.6962538958
>Hit Enter To Close...
Confusion Matrix followed by accuracy of each class
[[412  59  87  18  97  26  32  14 195  60](0) airplane  41.2%
 [ 47 386   9  33  91  17  65  12 128 212](1) automobile  38.6%
 [ 95  13 243 127 230 116  54  29  54  39](2) bird    24.3%
 [ 37  22  67 280 215 161  74  34  44  66](3) cat     28.0%
 [ 50  31  82 133 434  65  84  47  35  39](4) deer    43.4%
 [ 19  10  88 232 213 318  29  47  17  27](5) dog     31.8%
 [ 19  35  47 142 205  31 381  10  45  85](6) frog    38.1%
 [ 23  29  52 126 232 137  35 270  23  73](7) horse   27.0%
 [131  87  22  25  48   9  27   7 562  82](8) ship    56.2%
 [ 42 135  18  49  66  24  38   16 114 498](9) truck   49.8%
 (0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]

```

5.) Fully connected layer 1 = 120 and fully connected layer 2 = 84

Figure 1



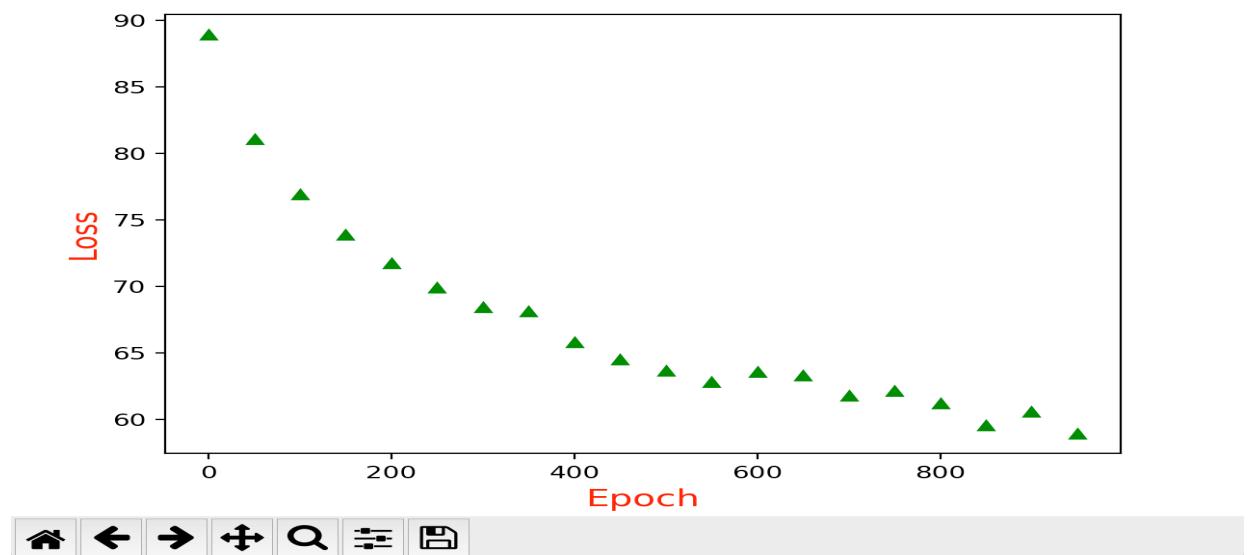
```

Initialized with learning_rate 0.001
epoch 0000 : loss is 010.94, accuracy on training set 12.50 %, accuracy on validation set 10.52 %
epoch 0050 : loss is 002.52, accuracy on training set 10.94 %, accuracy on validation set 14.13 %
epoch 0100 : loss is 002.39, accuracy on training set 12.50 %, accuracy on validation set 18.56 %
epoch 0150 : loss is 002.16, accuracy on training set 25.00 %, accuracy on validation set 20.82 %
epoch 0200 : loss is 002.05, accuracy on training set 18.75 %, accuracy on validation set 21.85 %
epoch 0250 : loss is 001.93, accuracy on training set 26.56 %, accuracy on validation set 24.14 %
epoch 0300 : loss is 001.80, accuracy on training set 39.06 %, accuracy on validation set 28.85 %
epoch 0350 : loss is 001.96, accuracy on training set 34.38 %, accuracy on validation set 29.03 %
epoch 0400 : loss is 001.74, accuracy on training set 29.69 %, accuracy on validation set 31.35 %
epoch 0450 : loss is 001.96, accuracy on training set 34.38 %, accuracy on validation set 32.19 %
epoch 0500 : loss is 001.92, accuracy on training set 32.81 %, accuracy on validation set 31.42 %
epoch 0550 : loss is 001.79, accuracy on training set 34.38 %, accuracy on validation set 34.19 %
epoch 0600 : loss is 001.82, accuracy on training set 28.12 %, accuracy on validation set 32.32 %
epoch 0650 : loss is 001.85, accuracy on training set 23.44 %, accuracy on validation set 34.88 %
epoch 0700 : loss is 001.78, accuracy on training set 43.75 %, accuracy on validation set 36.29 %
epoch 0750 : loss is 001.70, accuracy on training set 39.06 %, accuracy on validation set 36.03 %
epoch 0800 : loss is 001.90, accuracy on training set 29.69 %, accuracy on validation set 36.26 %
epoch 0850 : loss is 001.70, accuracy on training set 34.38 %, accuracy on validation set 36.42 %
epoch 0900 : loss is 001.73, accuracy on training set 40.62 %, accuracy on validation set 37.83 %
epoch 0950 : loss is 001.72, accuracy on training set 37.50 %, accuracy on validation set 37.51 %
Accuracy on test model: 36.36 %
Total loss in all epochs: 47.3529236317
>Hit Enter To Close....
Confusion Matrix followed by accuracy of each class
[[554  3 106 25 24 23 50 41 89 85](0) airplane 55.4%
[138  76 14 35 36 26 119 50 109 397](1) automobile 7.6%
[157  1 317 126 65 105 119 46 25 39](2) bird 31.7%
[ 51  2 130 240 74 169 181 77 17 59](3) cat 24.0%
[ 66  3 162 149 202 99 182 97 10 30](4) deer 20.2%
[ 35  2 151 215 69 341 79 81 4 23](5) dog 34.1%
[ 40  1 59 130 36 32 587 46 4 65](6) frog 58.7%
[ 33  3 66 111 74 162 87 388 11 65](7) horse 38.8%
[301 15 52 21 17 11 46 19 382 136](8) ship 38.2%
[ 87  7 25 46 30 13 106 56 81 549](9) truck 54.9%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)

```

6.) Fully connected layer 1 = 200 and fully connected layer 2 = 200

Figure 1

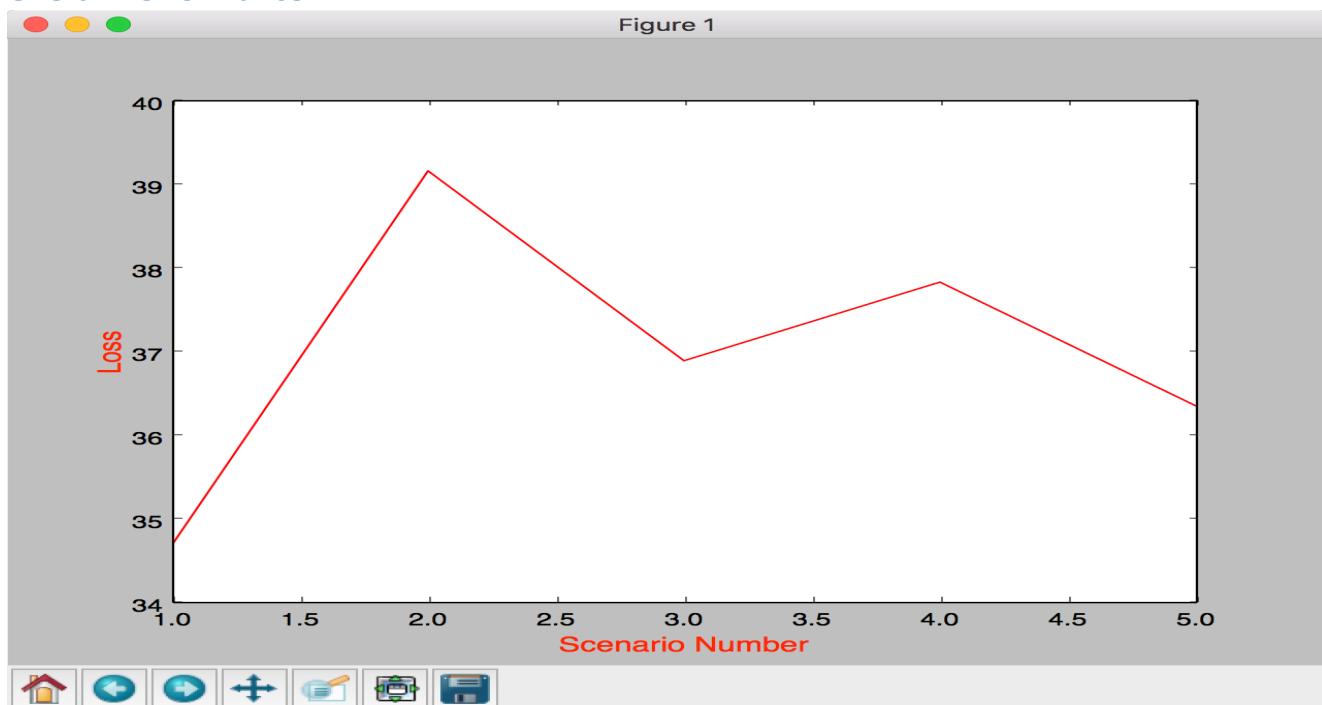


```

Initialized with learning_rate 0.001
epoch 0000 : loss is 013.90, accuracy on training set 12.50 %, accuracy on validation set 11.07 %
epoch 0050 : loss is 002.19, accuracy on training set 28.12 %, accuracy on validation set 18.95 %
epoch 0100 : loss is 002.08, accuracy on training set 23.44 %, accuracy on validation set 23.07 %
epoch 0150 : loss is 002.12, accuracy on training set 18.75 %, accuracy on validation set 26.13 %
epoch 0200 : loss is 001.89, accuracy on training set 21.88 %, accuracy on validation set 28.24 %
epoch 0250 : loss is 002.05, accuracy on training set 25.00 %, accuracy on validation set 30.08 %
epoch 0300 : loss is 001.84, accuracy on training set 34.38 %, accuracy on validation set 31.54 %
epoch 0350 : loss is 001.96, accuracy on training set 29.69 %, accuracy on validation set 31.88 %
epoch 0400 : loss is 001.79, accuracy on training set 32.81 %, accuracy on validation set 34.18 %
epoch 0450 : loss is 001.92, accuracy on training set 25.00 %, accuracy on validation set 35.48 %
epoch 0500 : loss is 001.86, accuracy on training set 29.69 %, accuracy on validation set 36.34 %
epoch 0550 : loss is 001.75, accuracy on training set 37.50 %, accuracy on validation set 37.19 %
epoch 0600 : loss is 001.74, accuracy on training set 29.69 %, accuracy on validation set 36.43 %
epoch 0650 : loss is 001.69, accuracy on training set 37.50 %, accuracy on validation set 36.68 %
epoch 0700 : loss is 001.57, accuracy on training set 34.38 %, accuracy on validation set 38.19 %
epoch 0750 : loss is 001.97, accuracy on training set 28.12 %, accuracy on validation set 37.85 %
epoch 0800 : loss is 001.84, accuracy on training set 34.38 %, accuracy on validation set 38.79 %
epoch 0850 : loss is 001.45, accuracy on training set 46.88 %, accuracy on validation set 40.41 %
epoch 0900 : loss is 001.54, accuracy on training set 43.75 %, accuracy on validation set 39.40 %
epoch 0950 : loss is 001.57, accuracy on training set 46.88 %, accuracy on validation set 41.04 %
Accuracy on test model: 40.28 %
Total loss in all epochs: 48.7058296204
>Hit Enter To Close....
Confusion Matrix followed by accuracy of each class
[[502 31 103 42 41 22 45 20 85 109](0) airplane 50.2%
 [ 57 292 10 24 40 24 58 42 48 405](1) automobile 29.2%
 [ 95 12 278 160 82 143 96 53 27 54](2) bird 27.8%
 [ 36 20 77 297 106 212 95 68 13 76](3) cat 29.7%
 [ 32 22 69 114 322 152 108 101 13 67](4) deer 32.2%
 [ 23 3 80 232 93 402 42 87 5 33](5) dog 40.2%
 [ 22 25 68 129 80 58 517 37 8 56](6) frog 51.7%
 [ 24 10 38 111 116 133 20 426 9 113](7) horse 42.6%
 [234 85 17 32 22 13 41 16 304 236](8) ship 30.4%
 [ 46 69 17 29 33 12 30 38 38 688](9) truck 68.8%
 (0) (1) (2) (3) (4) (5) (6) (7) (8) (9)]
```

Overall Performance

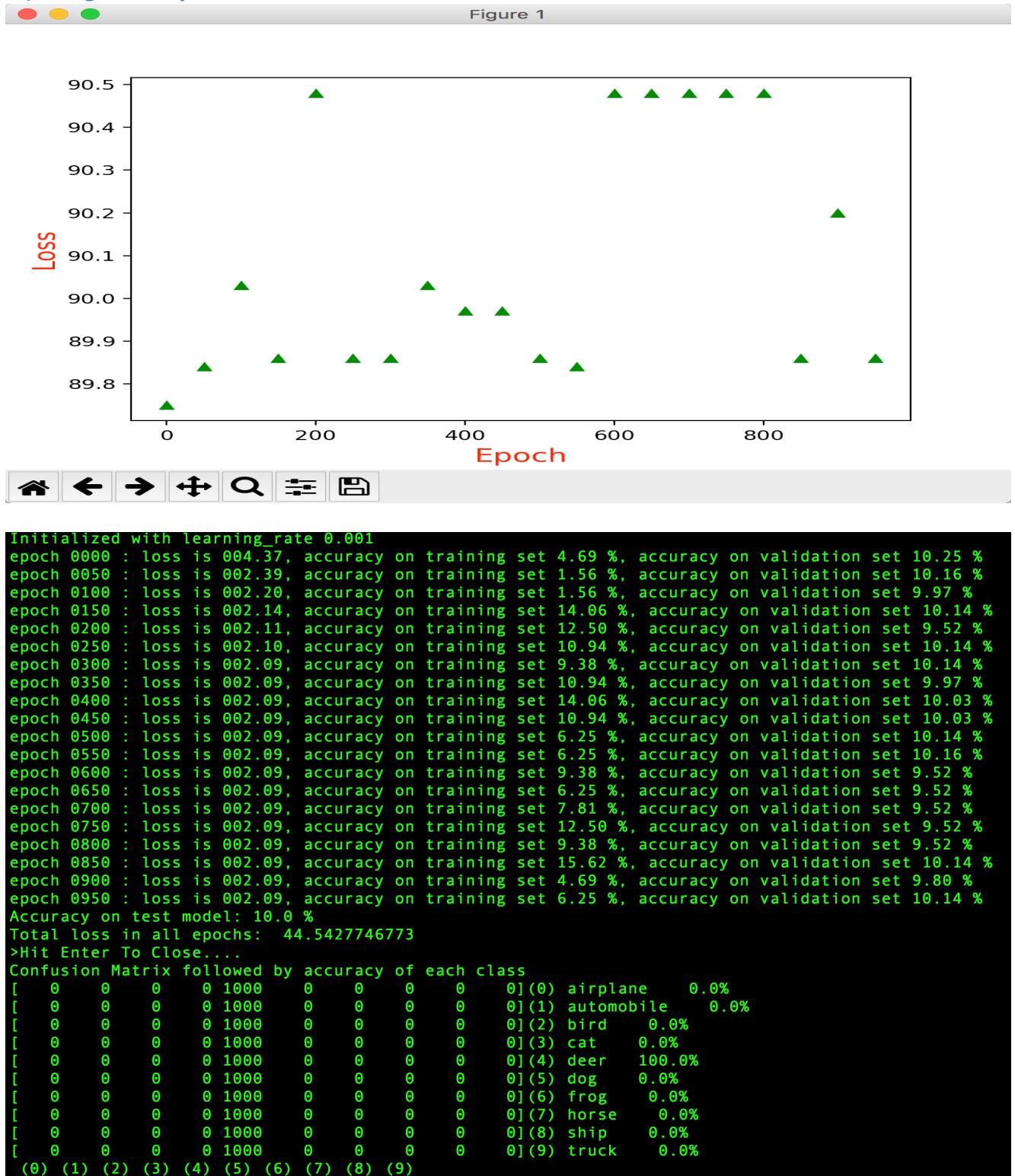
Figure 1



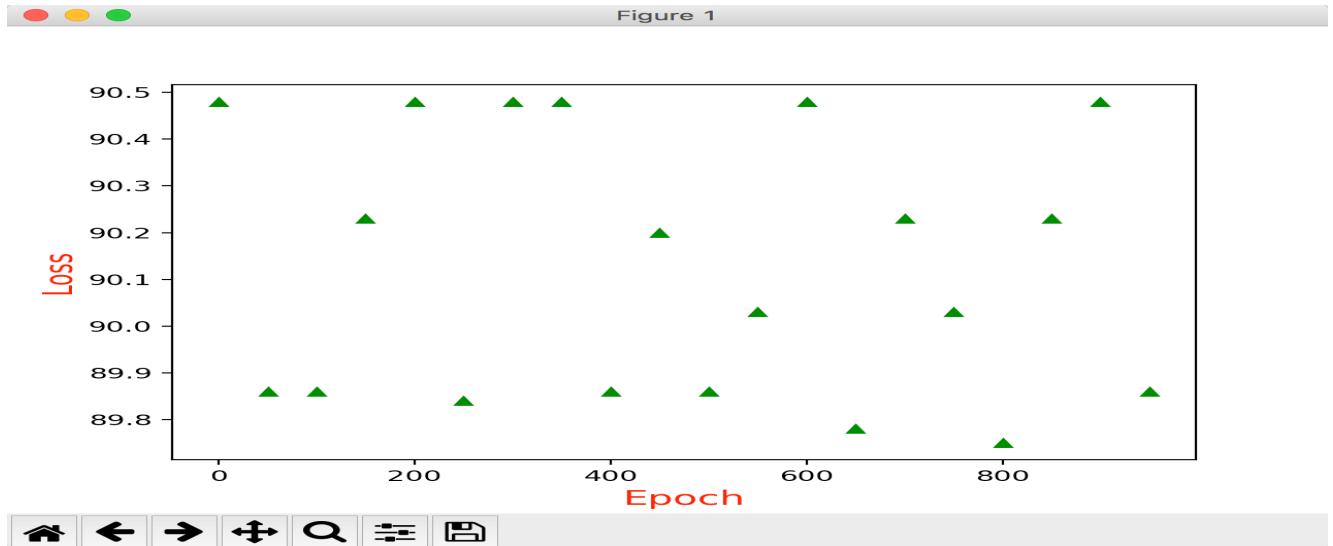
Varying the regularizer

Constants: All layers as mentioned in homework assignment. Epoch = 1000 and learning rate = 0.001

1.) Weight decay = 1e-8



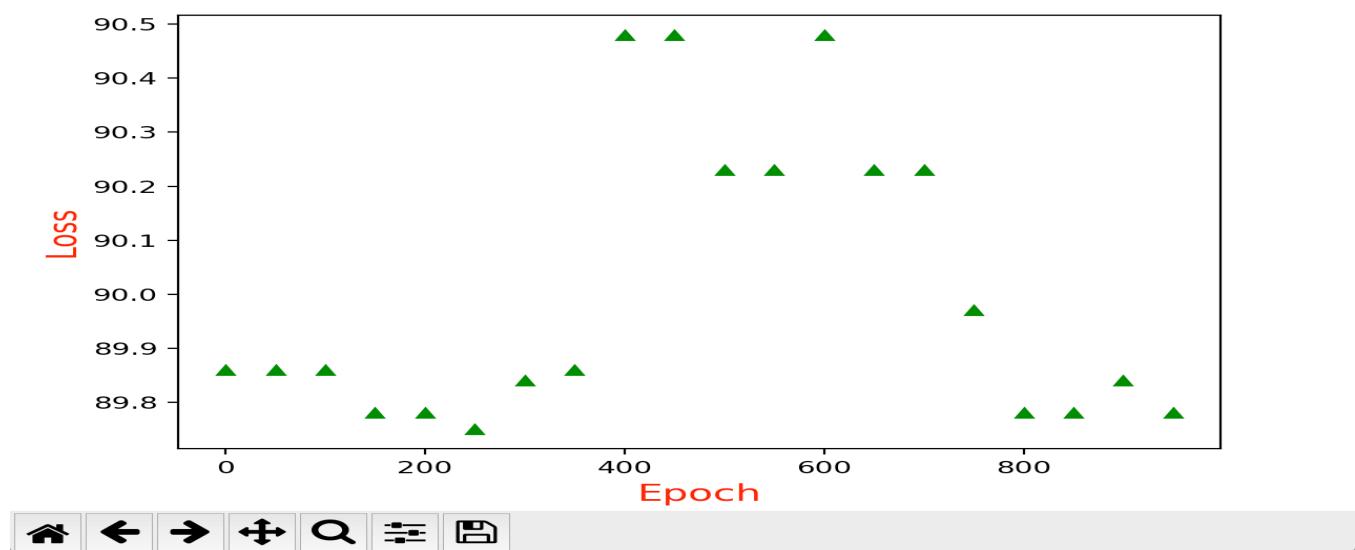
2.) Weight decay = 1e-10



```
Initialized with learning_rate 0.001
epoch 0000 : loss is 003.54, accuracy on training set 10.94 %, accuracy on validation set 9.52 %
epoch 0050 : loss is 47351215574482944.00, accuracy on training set 10.94 %, accuracy on validation set 10.14 %
epoch 0100 : loss is 47351215574482944.00, accuracy on training set 6.25 %, accuracy on validation set 10.14 %
epoch 0150 : loss is 438290520519737344.00, accuracy on training set 10.94 %, accuracy on validation set 9.77 %
epoch 0200 : loss is 438290520519737344.00, accuracy on training set 14.06 %, accuracy on validation set 9.52 %
epoch 0250 : loss is 438290520519737344.00, accuracy on training set 9.38 %, accuracy on validation set 10.16 %
epoch 0300 : loss is 438290520519737344.00, accuracy on training set 9.38 %, accuracy on validation set 9.52 %
epoch 0350 : loss is 438290520519737344.00, accuracy on training set 9.38 %, accuracy on validation set 9.52 %
epoch 0400 : loss is 438290520519737344.00, accuracy on training set 4.69 %, accuracy on validation set 10.14 %
epoch 0450 : loss is 438290520519737344.00, accuracy on training set 7.81 %, accuracy on validation set 9.80 %
epoch 0500 : loss is 438290520519737344.00, accuracy on training set 9.38 %, accuracy on validation set 10.14 %
epoch 0550 : loss is 438290520519737344.00, accuracy on training set 10.94 %, accuracy on validation set 9.97 %
epoch 0600 : loss is 438290520519737344.00, accuracy on training set 6.25 %, accuracy on validation set 9.52 %
epoch 0650 : loss is 438290520519737344.00, accuracy on training set 12.50 %, accuracy on validation set 10.22 %
epoch 0700 : loss is 438290520519737344.00, accuracy on training set 6.25 %, accuracy on validation set 9.77 %
epoch 0750 : loss is 438290520519737344.00, accuracy on training set 3.12 %, accuracy on validation set 9.97 %
epoch 0800 : loss is 438290520519737344.00, accuracy on training set 12.50 %, accuracy on validation set 10.25 %
epoch 0850 : loss is 438290520519737344.00, accuracy on training set 14.06 %, accuracy on validation set 9.77 %
epoch 0900 : loss is 438290520519737344.00, accuracy on training set 9.38 %, accuracy on validation set 9.52 %
epoch 0950 : loss is 438290520519737344.00, accuracy on training set 4.69 %, accuracy on validation set 10.14 %
Accuracy on test model: 10.0 %
Total loss in all epochs: 7.54564127998e+18
>Hit Enter To Close.....
Confusion Matrix followed by accuracy of each class
[[ 0  0 1000  0  0  0  0  0  0] (0) airplane  0.0%
 [ 0  0 1000  0  0  0  0  0  0] (1) automobile  0.0%
 [ 0  0 1000  0  0  0  0  0  0] (2) bird   100.0%
 [ 0  0 1000  0  0  0  0  0  0] (3) cat    0.0%
 [ 0  0 1000  0  0  0  0  0  0] (4) deer   0.0%
 [ 0  0 1000  0  0  0  0  0  0] (5) dog    0.0%
 [ 0  0 1000  0  0  0  0  0  0] (6) frog   0.0%
 [ 0  0 1000  0  0  0  0  0  0] (7) horse  0.0%
 [ 0  0 1000  0  0  0  0  0  0] (8) ship   0.0%
 [ 0  0 1000  0  0  0  0  0  0] (9) truck  0.0%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)
```

3.) Weight decay = 1e-15

Figure 1



```
Initialized with learning_rate 0.001
epoch 0000 : loss is 004.45, accuracy on training set 18.75 %, accuracy on validation set 10.14 %
epoch 0050 : loss is 000.63, accuracy on training set 7.81 %, accuracy on validation set 10.14 %
epoch 0100 : loss is 000.44, accuracy on training set 14.06 %, accuracy on validation set 10.14 %
epoch 0150 : loss is 000.38, accuracy on training set 10.94 %, accuracy on validation set 10.22 %
epoch 0200 : loss is 000.35, accuracy on training set 10.94 %, accuracy on validation set 10.22 %
epoch 0250 : loss is 000.34, accuracy on training set 9.38 %, accuracy on validation set 10.25 %
epoch 0300 : loss is 000.33, accuracy on training set 4.69 %, accuracy on validation set 10.16 %
epoch 0350 : loss is 000.33, accuracy on training set 7.81 %, accuracy on validation set 10.14 %
epoch 0400 : loss is 000.33, accuracy on training set 10.94 %, accuracy on validation set 9.52 %
epoch 0450 : loss is 000.33, accuracy on training set 7.81 %, accuracy on validation set 9.52 %
epoch 0500 : loss is 000.33, accuracy on training set 4.69 %, accuracy on validation set 9.77 %
epoch 0550 : loss is 000.33, accuracy on training set 17.19 %, accuracy on validation set 9.77 %
epoch 0600 : loss is 000.33, accuracy on training set 7.81 %, accuracy on validation set 9.52 %
epoch 0650 : loss is 000.33, accuracy on training set 6.25 %, accuracy on validation set 9.77 %
epoch 0700 : loss is 000.33, accuracy on training set 9.38 %, accuracy on validation set 9.77 %
epoch 0750 : loss is 000.33, accuracy on training set 9.38 %, accuracy on validation set 10.03 %
epoch 0800 : loss is 000.33, accuracy on training set 3.12 %, accuracy on validation set 10.22 %
epoch 0850 : loss is 000.33, accuracy on training set 12.50 %, accuracy on validation set 10.22 %
epoch 0900 : loss is 000.33, accuracy on training set 14.06 %, accuracy on validation set 10.16 %
epoch 0950 : loss is 000.33, accuracy on training set 6.25 %, accuracy on validation set 10.22 %
Accuracy on test model: 10.0 %
Total loss in all epochs: 11.1580699086
>Hit Enter To Close....
```

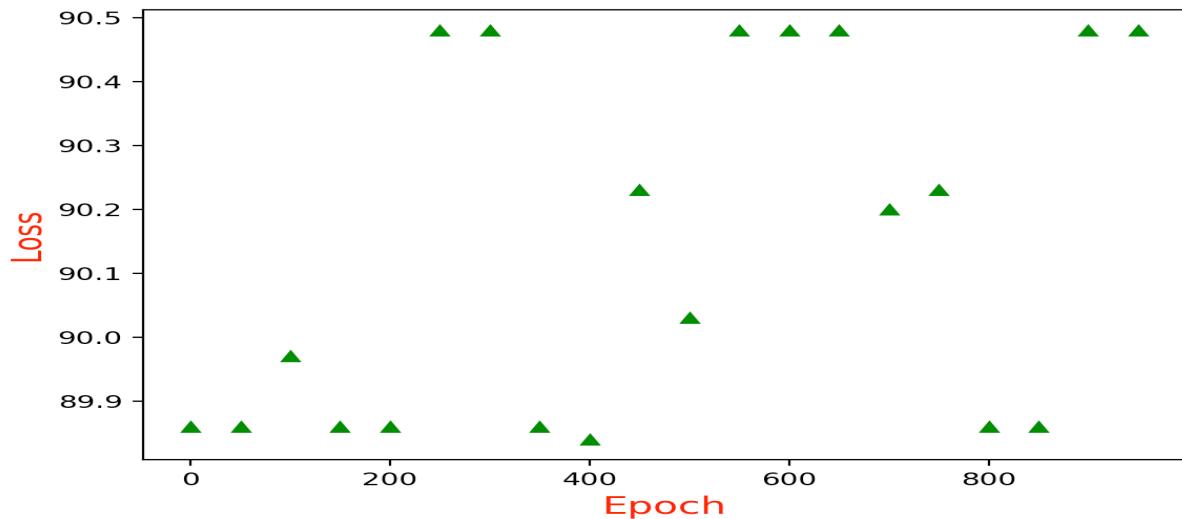
Confusion Matrix followed by accuracy of each class

[0 0 1000 0 0 0 0 0 0](0) airplane 0.0%
[0 0 1000 0 0 0 0 0 0](1) automobile 0.0%
[0 0 1000 0 0 0 0 0 0](2) bird 100.0%
[0 0 1000 0 0 0 0 0 0](3) cat 0.0%
[0 0 1000 0 0 0 0 0 0](4) deer 0.0%
[0 0 1000 0 0 0 0 0 0](5) dog 0.0%
[0 0 1000 0 0 0 0 0 0](6) frog 0.0%
[0 0 1000 0 0 0 0 0 0](7) horse 0.0%
[0 0 1000 0 0 0 0 0 0](8) ship 0.0%
[0 0 1000 0 0 0 0 0 0](9) truck 0.0%

(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)

4.) Weight decay = 0.00001

Figure 1



```
Initialized with learning_rate 0.001
epoch 0000 : loss is 004.25, accuracy on training set 6.25 %, accuracy on validation set 10.14 %
epoch 0050 : loss is 16504442.00, accuracy on training set 12.50 %, accuracy on validation set 10.14 %
epoch 0100 : loss is 16496192.00, accuracy on training set 7.81 %, accuracy on validation set 10.03 %
epoch 0150 : loss is 16487948.00, accuracy on training set 12.50 %, accuracy on validation set 10.14 %
epoch 0200 : loss is 16479717.00, accuracy on training set 4.69 %, accuracy on validation set 10.14 %
epoch 0250 : loss is 16471485.00, accuracy on training set 7.81 %, accuracy on validation set 9.52 %
epoch 0300 : loss is 16463260.00, accuracy on training set 7.81 %, accuracy on validation set 9.52 %
epoch 0350 : loss is 16455036.00, accuracy on training set 10.94 %, accuracy on validation set 10.14 %
epoch 0400 : loss is 16446818.00, accuracy on training set 15.62 %, accuracy on validation set 10.16 %
epoch 0450 : loss is 16438598.00, accuracy on training set 9.38 %, accuracy on validation set 9.77 %
epoch 0500 : loss is 16430376.00, accuracy on training set 17.19 %, accuracy on validation set 9.97 %
epoch 0550 : loss is 16422172.00, accuracy on training set 15.62 %, accuracy on validation set 9.52 %
epoch 0600 : loss is 16413957.00, accuracy on training set 7.81 %, accuracy on validation set 9.52 %
epoch 0650 : loss is 16405743.00, accuracy on training set 4.69 %, accuracy on validation set 9.52 %
epoch 0700 : loss is 16397542.00, accuracy on training set 12.50 %, accuracy on validation set 9.80 %
epoch 0750 : loss is 16389341.00, accuracy on training set 7.81 %, accuracy on validation set 9.77 %
epoch 0800 : loss is 16381145.00, accuracy on training set 7.81 %, accuracy on validation set 10.14 %
epoch 0850 : loss is 16372956.00, accuracy on training set 9.38 %, accuracy on validation set 10.14 %
epoch 0900 : loss is 16364766.00, accuracy on training set 6.25 %, accuracy on validation set 9.52 %
epoch 0950 : loss is 16356579.00, accuracy on training set 10.94 %, accuracy on validation set 9.52 %
Accuracy on test model: 10.0 %
Total loss in all epochs: 312178077.254
>Hit Enter To Close...
Confusion Matrix followed by accuracy of each class
[[ 0 1000 0 0 0 0 0 0 0] (0) airplane 0.0%
[ 0 1000 0 0 0 0 0 0 0] (1) automobile 100.0%
[ 0 1000 0 0 0 0 0 0 0] (2) bird 0.0%
[ 0 1000 0 0 0 0 0 0 0] (3) cat 0.0%
[ 0 1000 0 0 0 0 0 0 0] (4) deer 0.0%
[ 0 1000 0 0 0 0 0 0 0] (5) dog 0.0%
[ 0 1000 0 0 0 0 0 0 0] (6) frog 0.0%
[ 0 1000 0 0 0 0 0 0 0] (7) horse 0.0%
[ 0 1000 0 0 0 0 0 0 0] (8) ship 0.0%
[ 0 1000 0 0 0 0 0 0 0] (9) truck 0.0%
(0) (1) (2) (3) (4) (5) (6) (7) (8) (9)
```

My Comments

Varying various parameters results in interesting values. It is just that increasing a particular value can not keep increasing the accuracy or lower the loss beyond a particular value. That is investing beyond that point isn't valuable as drop in loss will be less than 1% or so.

The same case is observed in almost every variation. If I keep increasing number of filters in a particular layer then accuracy increases to a point and then it starts decreasing. In the assignment, I varied various parameters such as filter size, number of filters, regularizer etc. and in each case I observed that accuracy marginally increased beyond a point.

Source Code

```
__author__ = 'deepika'

import numpy as np
import pickle
import tensorflow as tf
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import cv2

cifar10_folder = './cifar10/'
train_datasets = ['data_batch_1', 'data_batch_2', 'data_batch_3',
'data_batch_4' ]
test_dataset = ['test_batch']
validation_dataset = ['data_batch_5']
c10_image_height = 31
c10_image_width = 31
c10_image_depth = 1
c10_num_labels = 10
c10_image_size = 32

def handleAugmentation(np_dataset_, image_width, image_height, image_depth):
    new_dataset = []
    for image in np_dataset_:
        height, width = image.shape[0], image.shape[1]
        # Resize by +10%
        res1 = cv2.resize(image,( int(height + 0.1 * height), int(width + 0.1 * width)),interpolation=cv2.INTER_CUBIC)
```

```

#Crop 90% from all 4 corners
crop_perc = 0.9

roi_size =
    int(res1.shape[0] * crop_perc), int(res1.shape[1] * crop_perc)
]
roi_tl_x = [0, res1.shape[0] - roi_size[0]]
roi_tl_y = [0, res1.shape[1] - roi_size[1]]

sel_x = np.random.randint(roi_tl_x[0], roi_tl_x[1])
sel_y = np.random.randint(roi_tl_y[0], roi_tl_y[1])
cur_roi = [sel_x, sel_y, sel_x + roi_size[0], sel_y + roi_size[1]]
res2 = res1[cur_roi[0]:cur_roi[2], cur_roi[1]:cur_roi[3]]


#Flip
# horizontal flip
res3 = cv2.flip(res2, 1)

# vertical flip
res3 = cv2.flip(res3, 0)

# all flip
res3 = cv2.flip(res3, -1)
#print "res3.shape", res3.shape

new_dataset.append(res3)

new_dataset = np.array([np.array(image_data).reshape(31, 31, image_depth)
for image_data in new_dataset])
return new_dataset

def randomize(dataset, labels):
    permutation = np.random.permutation(labels.shape[0])
    shuffled_dataset = dataset[permutation, :, :]
    shuffled_labels = labels[permutation]
    return shuffled_dataset, shuffled_labels

def one_hot_encode(np_array):
    return (np.arange(10) == np_array[:,None]).astype(np.float32)

def reformat_data(dataset, labels, image_width, image_height, image_depth):
    grayscale = 0.21*dataset[:,0:1024] + 0.72*dataset[:,1024:2048] +
    0.07*dataset[:,2048:3072]

```

```

np_dataset_ = np.array([np.array(image_data).reshape(image_width,
image_height, image_depth) for image_data in grayscale])
np_dataset_ = np_dataset_ - np_dataset_.mean(axis=1, keepdims=True)
np_dataset_ = handleAugmentation(np_dataset_, image_width, image_height,
image_depth)
np_labels_ = one_hot_encode(np.array(labels, dtype=np.float32))
np_dataset, np_labels = randomize(np_dataset_, np_labels_)
return np_dataset, np_labels

def flatten_tf_array(array):
    shape = array.get_shape().as_list()
    return tf.reshape(array, [shape[0], shape[1] * shape[2] * shape[3]])

def accuracy(predictions, labels):
    return (100.0 * np.sum(np.argmax(predictions, 1) == np.argmax(labels, 1)) /
predictions.shape[0])

with open(cifar10_folder + validation_dataset[0], 'rb') as f0:
    c10_validation_dict = pickle.load(f0)

c10_validation_dataset, c10_validation_labels = c10_validation_dict[b'data'],
c10_validation_dict[b'labels']
validation_dataset_cifar10, validation_labels_cifar10 =
reformat_data(c10_validation_dataset, c10_validation_labels, c10_image_size,
c10_image_size, c10_image_depth)

with open(cifar10_folder + test_dataset[0], 'rb') as f0:
    c10_test_dict = pickle.load(f0)

c10_test_dataset, c10_test_labels = c10_test_dict[b'data'],
c10_test_dict[b'labels']
test_dataset_cifar10, test_labels_cifar10 = reformat_data(c10_test_dataset,
c10_test_labels, c10_image_size, c10_image_size, c10_image_depth)

c10_train_dataset, c10_train_labels = [], []
for train_dataset in train_datasets:
    with open(cifar10_folder + train_dataset, 'rb') as f0:
        c10_train_dict = pickle.load(f0)
        c10_train_dataset_, c10_train_labels_ = c10_train_dict[b'data'],
c10_train_dict[b'labels']

```

```

c10_train_dataset.append(c10_train_dataset_)
c10_train_labels += c10_train_labels_

c10_train_dataset = np.concatenate(c10_train_dataset, axis=0)
train_dataset_cifar10, train_labels_cifar10 = reformat_data(c10_train_dataset,
c10_train_labels, c10_image_size, c10_image_size, c10_image_depth)
del c10_train_dataset
del c10_train_labels

#####
# Neural Net
#####

LENET5_BATCH_SIZE = 32
LENET5_PATCH_SIZE = 5
LENET5_PATCH_DEPTH_1 = 8
LENET5_PATCH_DEPTH_2 = 16
LENET5_NUM_HIDDEN_1 = 120 #120
LENET5_NUM_HIDDEN_2 = 84 #84

def variables_lenet5(patch_size = LENET5_PATCH_SIZE, patch_depth1 =
LENET5_PATCH_DEPTH_1,
                     patch_depth2 = LENET5_PATCH_DEPTH_2,
                     num_hidden1 = LENET5_NUM_HIDDEN_1, num_hidden2 =
LENET5_NUM_HIDDEN_2,
                     image_depth = 1, num_labels = 10):
    xavier = tf.contrib.layers.xavier_initializer()
    w1 = tf.get_variable("w1", shape=[patch_size, patch_size, image_depth,
patch_depth1], initializer=xavier)
    b1 = tf.Variable(tf.zeros([patch_depth1]))

    w2 = tf.get_variable("w2", shape=[patch_size, patch_size, patch_depth1,
patch_depth2], initializer=xavier)
    b2 = tf.Variable(tf.constant(1.0, shape=[patch_depth2]))

    w3 = tf.get_variable("w3", shape=[4*4*patch_depth2, num_hidden1],
initializer=xavier)
    b3 = tf.Variable(tf.constant(1.0, shape = [num_hidden1]))

    w4 = tf.get_variable("w4", shape=[num_hidden1, num_hidden2],
initializer=xavier)
    b4 = tf.Variable(tf.constant(1.0, shape = [num_hidden2]))

```

```

w5 = tf.get_variable("w5", shape=[num_hidden2,
num_labels], initializer=xavier)
b5 = tf.Variable(tf.constant(1.0, shape = [num_labels]))
variables = {
    'w1': w1, 'w2': w2, 'w3': w3, 'w4': w4, 'w5': w5,
    'b1': b1, 'b2': b2, 'b3': b3, 'b4': b4, 'b5': b5
}
return variables

def model_lenet5(data, variables):
    layer1_conv = tf.nn.conv2d(data, variables['w1'], [1, 1, 1, 1],
padding='VALID') + variables['b1']
    layer1_conv = tf.nn.relu(layer1_conv)

    layer1_pool = tf.nn.max_pool(layer1_conv, [1, 2, 2, 1], [1, 2, 2, 1],
padding='VALID')

    layer2_conv = tf.nn.conv2d(layer1_pool, variables['w2'], [1, 1, 1, 1],
padding='VALID') + variables['b2']
    layer2_conv = tf.nn.relu(layer2_conv)

    layer2_pool = tf.nn.max_pool(layer2_conv, [1, 2, 2, 1], [1, 2, 2, 1],
padding='VALID')

    flat_layer = flatten_tf_array(layer2_pool)

    layer3_fccd = tf.matmul(flat_layer, variables['w3']) + variables['b3']
    layer3_actv = tf.nn.relu(layer3_fccd)

    layer4_fccd = tf.matmul(layer3_actv, variables['w4']) + variables['b4']
    layer4_actv = tf.nn.relu(layer4_fccd)

    logits = tf.matmul(layer4_actv, variables['w5']) + variables['b5']
return logits

# Run the model
# parameters determining the model size
image_size = 32
num_labels = 10

#the datasets
train_dataset = train_dataset_cifar10

```

```

train_labels = train_labels_cifar10
test_dataset = test_dataset_cifar10
test_labels = test_labels_cifar10
validation_dataset = validation_dataset_cifar10
validation_labels = validation_labels_cifar10

#number of iterations and learning rate
display_step = 50
learning_rate = 0.001
batch_size = 64
num_steps = 1000

graph = tf.Graph()
with graph.as_default():
    #1) First we put the input data in a tensorflow friendly form.
    tf_train_dataset = tf.placeholder(tf.float32, shape=(batch_size,
c10_image_width, c10_image_height, c10_image_depth))
    tf_train_labels = tf.placeholder(tf.float32, shape = (batch_size, num_labels))
    tf_test_dataset = tf.constant(test_dataset, tf.float32)
    tf_validation_dataset = tf.constant(validation_dataset, tf.float32)

    #2) Then, the weight matrices and bias vectors are initialized
    variables = variables_lenet5(image_depth = c10_image_depth, num_labels =
num_labels)

    #3. The model used to calculate the logits (predicted labels)
    model = model_lenet5
    logits = model(tf_train_dataset, variables)

    #4. then we compute the softmax cross entropy between the logits and the
(actual) labels
    #regularizer = tf.contrib.layers.l2_regularizer(scale=0.00001)
    #reg_term = tf.contrib.layers.apply_regularization(regularizer,
tf.trainable_variables())
    #loss =
    (tf.reduce_mean(tf.nn.sigmoid_cross_entropy_with_logits(labels=tf_train_labels,
logits=logits)) + reg_term)

    cost = tf.reduce_mean(tf.nn.softmax_cross_entropy_with_logits(logits=logits,
labels=tf_train_labels))

    #5. The optimizer is used to calculate the Adam Optimizer of the cost function
#optimizer = tf.train.GradientDescentOptimizer(0.5).minimize(loss)

```

```

optimizer = tf.train.AdamOptimizer().minimize(cost)

# Predictions for the training, validation, and test data.
train_prediction = tf.nn.softmax(logits)
validation_prediction = tf.nn.softmax(model(tf_validation_dataset, variables))
test_prediction = tf.nn.softmax(model(tf_test_dataset, variables)) #
tf_test_dataset

with tf.Session(graph=graph) as session:
    tf.global_variables_initializer().run()
    print 'Initialized with learning_rate', learning_rate
    x_steps = []
    y_value = []

    epoch_loss = 0
    for step in range(num_steps):
        offset = (step * batch_size) % (train_labels.shape[0] - batch_size)
        batch_data = train_dataset[offset:(offset + batch_size), :, :, :]
        batch_labels = train_labels[offset:(offset + batch_size), :]

        feed_dict = {tf_train_dataset : batch_data, tf_train_labels : batch_labels}
        _, l, predictions = session.run([optimizer, cost, train_prediction],
                                         feed_dict=feed_dict)
        train_accuracy = accuracy(predictions, batch_labels)

        if step % display_step == 0:
            validation_accuracy = accuracy(validation_prediction.eval(),
                                             validation_labels)
            message = "epoch {:04d} : loss is {:06.2f}, accuracy on training set  

{:02.2f} %, accuracy on validation set {:02.2f} %".format(step, l,
train_accuracy, validation_accuracy)
            epoch_loss += 1
            print message
            x_steps.append(step)
            y_value.append(100 - validation_accuracy)

            print "Accuracy on test model:", accuracy(test_prediction.eval(),
test_labels), "%"
            print "Total loss in all epochs: ", epoch_loss
            result_test_pred = test_prediction.eval()
            plt.ylabel('Loss', fontsize=14, color='red')
            plt.xlabel('Epoch', fontsize=14, color='red')
            plt.plot(x_steps, y_value, 'g^')

```

```

plt.show()
raw_input(">Hit Enter To Close....")
plt.close()

print "Confusion Matrix followed by accuracy of each class"
f = open(cifar10_folder + 'batches.meta', 'rb')
datadict = pickle.load(f)
f.close()
test_l = datadict['label_names']

predictions = [np.argmax(pred) for pred in result_test_pred]
true_labels = [np.argmax(lbl) for lbl in test_labels]

cm = confusion_matrix(true_labels, predictions)
for i in range(c10_num_labels):
    actual_label = len(filter(lambda x: x == i, true_labels))
    numerator = cm[i][i]
    class_name = "({}) {} {:^10}{}".format(i, test_l[i], str(numerator * 100.0 / actual_label) + "%")
    print(str(cm[i, :]) + class_name)
class_numbers = ["({0})".format(i) for i in range(len(test_l))]
print("".join(class_numbers))

```

References Used

- 1.) <https://github.com/learningsociety/deepmodels/>
- 2.) <https://github.com/exelban/tensorflow-cifar-10>
- 3.) <http://ataspinar.com/2017/08/15/building-convolutional-neural-networks-with-tensorflow/>