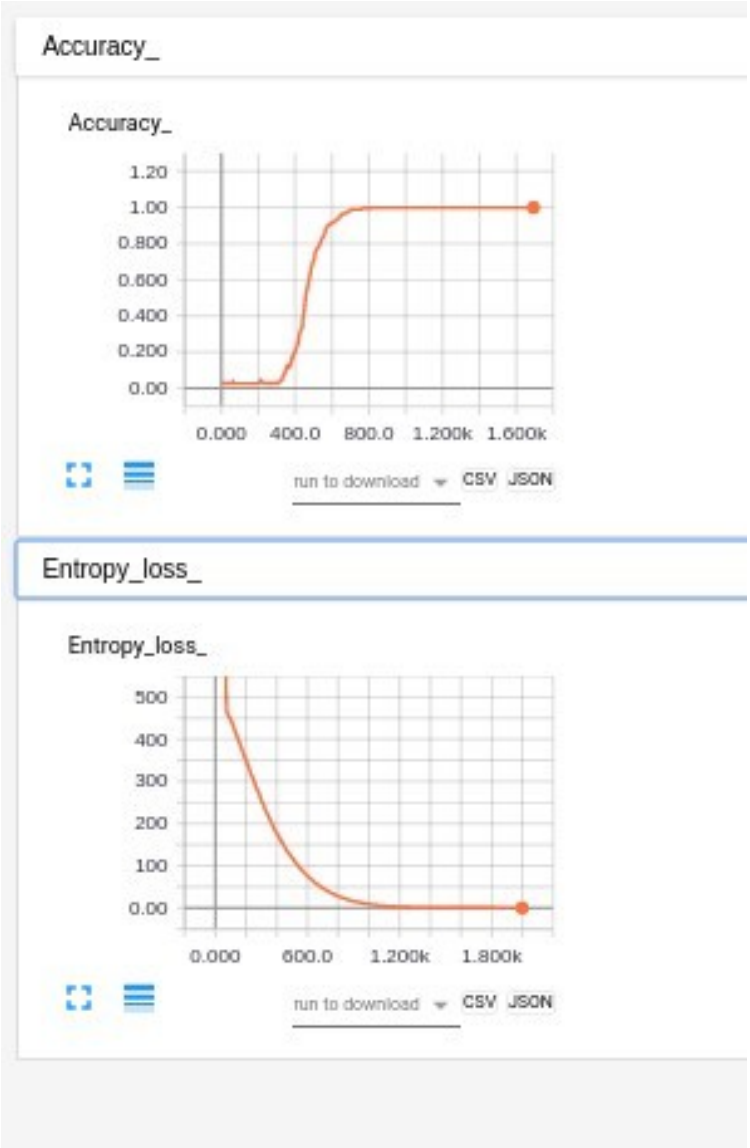
Report : 10
Zafar Mahmood

Q 1 : Neural Network

work flow : [Input → Hidden → Output]

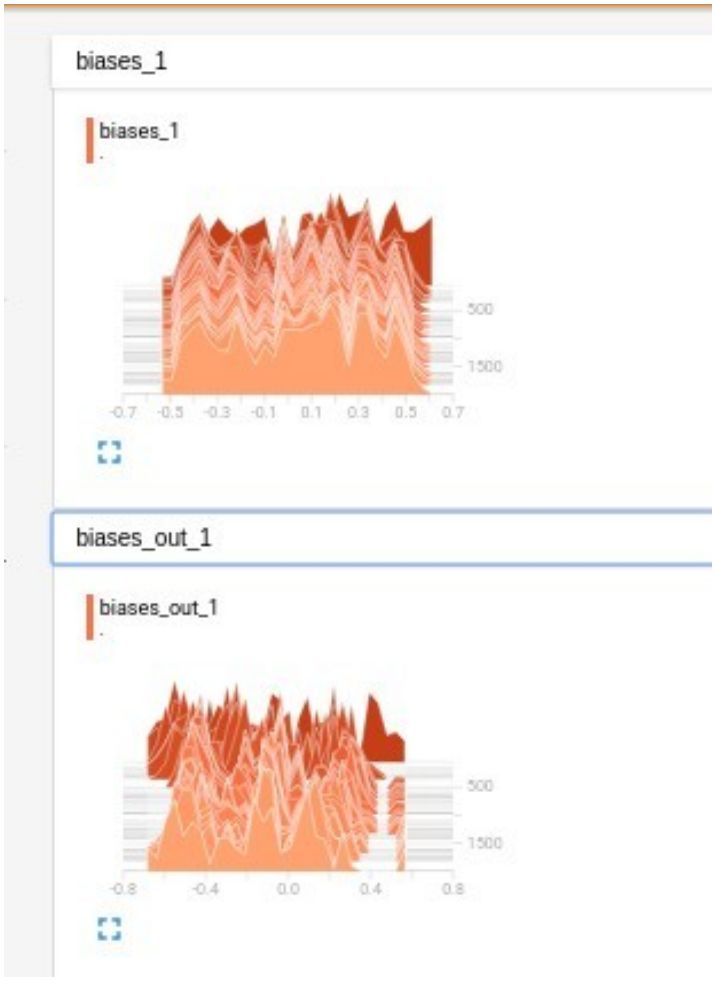
used the Regularized Model and Received Pretty Good output
Nodes = 256 , learning rate = 0.01 , epochs = 2000



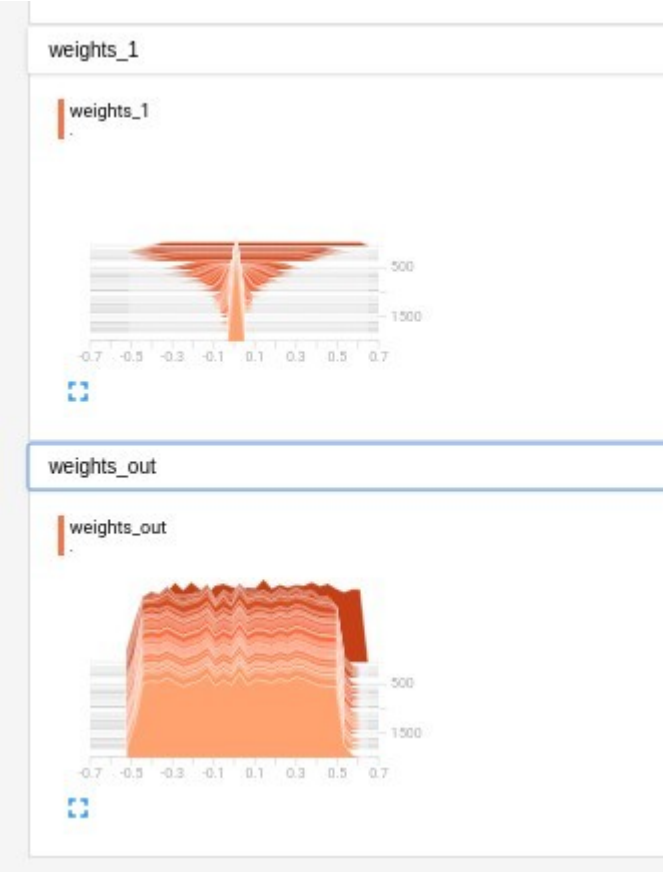
Biases


SCALARS
IMAGE
biases_1
0.600
0.200
-0.200
-0.600
0.000
600.0
1.200k
1.800k
biases_out_1
biases_out_1
0.600
0.200
-0.200
-0.600
0.000
600.0
1.200k
1.800k
weights_1
weights_1
0.600
0.200
-0.200
-0.600
0.000
600.0
1.200k
1.800k

Histogram :

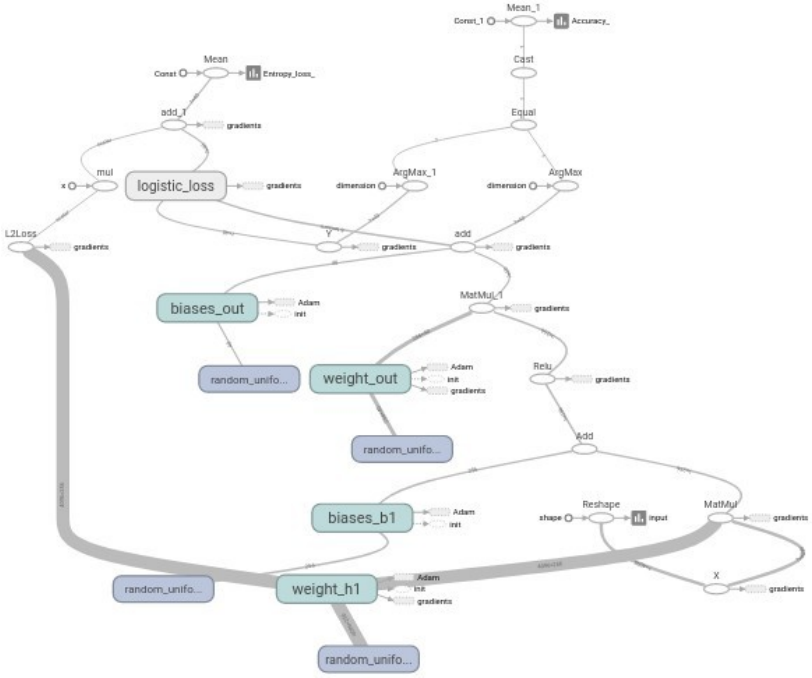

biases_1
biases_1
500
1500
-0.7
-0.5
-0.3
-0.1
0.1
0.3
0.5
0.7
biases_out_1
biases_out_1
500
1500
-0.8
-0.4
0.0
0.4
0.8

Weights :

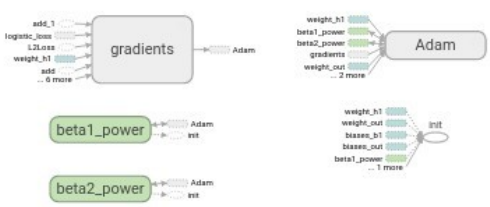

weights_1
weights_1
500
1500
-0.7 -0.5 -0.3 -0.1 0.1 0.3 0.5 0.7
weights_out
weights_out
500
1500
-0.7 -0.5 -0.3 -0.1 0.1 0.3 0.5 0.7

Graph :

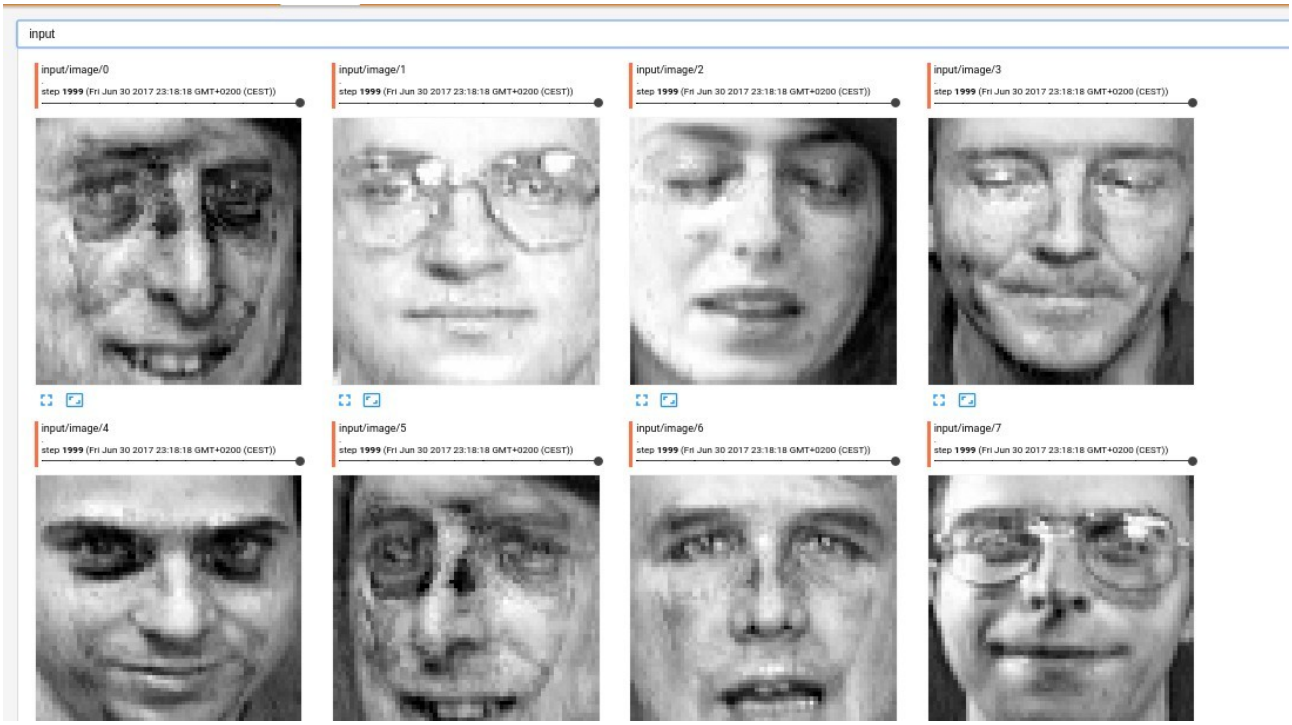
Main Graph



Auxiliary Nodes



Images :


input
input/image/0
step 1999 (Fri Jun 30 2017 23:18:18 GMT+0200 (CEST))
input/image/1
step 1999 (Fri Jun 30 2017 23:18:18 GMT+0200 (CEST))
input/image/2
step 1999 (Fri Jun 30 2017 23:18:18 GMT+0200 (CEST))
input/image/3
step 1999 (Fri Jun 30 2017 23:18:18 GMT+0200 (CEST))
input/image/4
step 1999 (Fri Jun 30 2017 23:18:18 GMT+0200 (CEST))
input/image/5
step 1999 (Fri Jun 30 2017 23:18:18 GMT+0200 (CEST))
input/image/6
step 1999 (Fri Jun 30 2017 23:18:18 GMT+0200 (CEST))
input/image/7
step 1999 (Fri Jun 30 2017 23:18:18 GMT+0200 (CEST))

Q : 2) Deep Neural Network

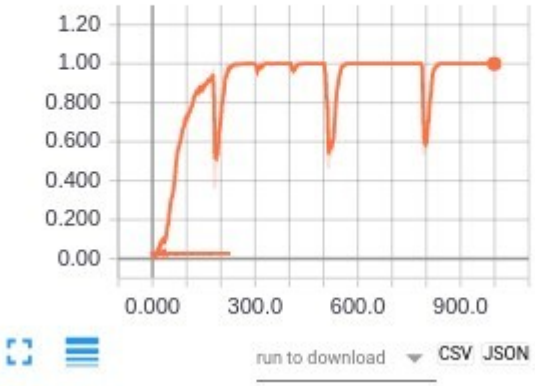
Used three layers with 256 to 512 nodes

Used the regularized model over the number of weights hidden layers , to make it fit over the test set.

Accuracy and Cost

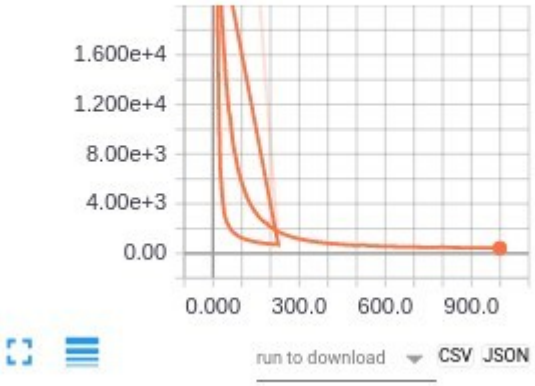
Accuracy_

Accuracy_

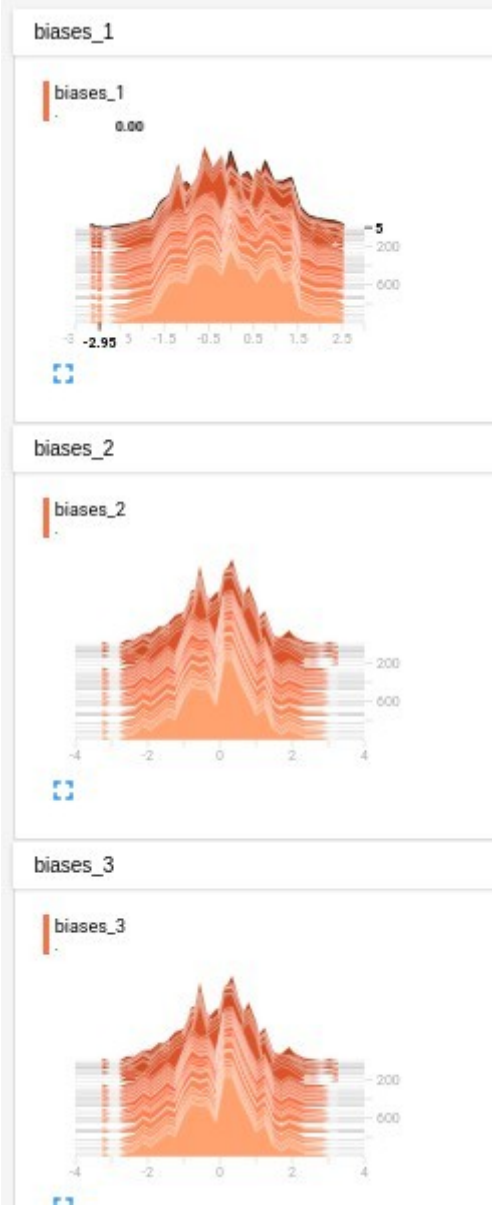


Entropy_loss_

Entropy_loss_



Biases

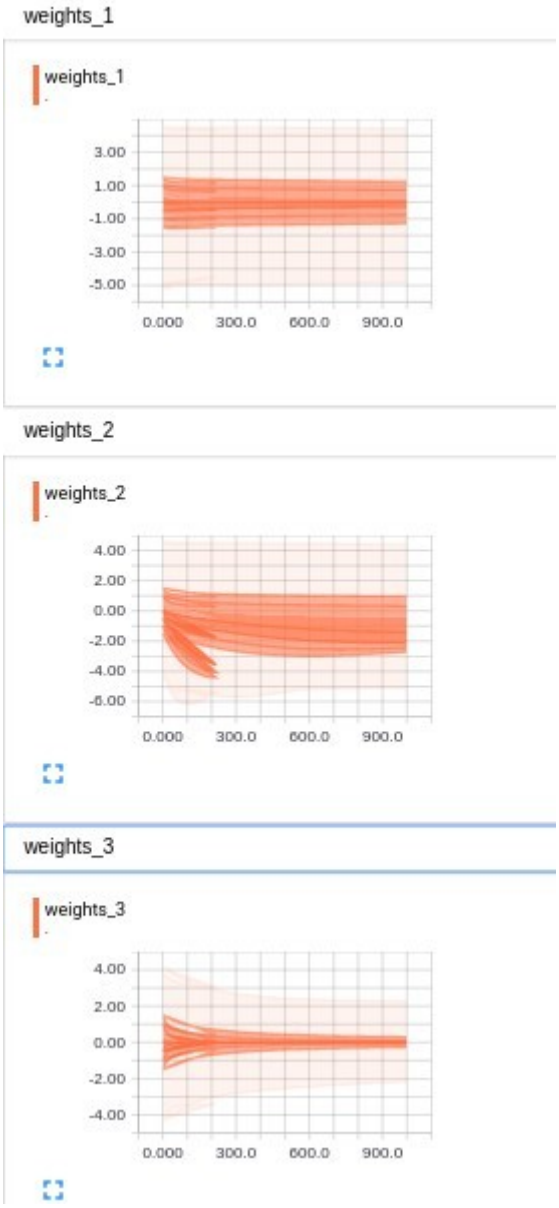


Weights

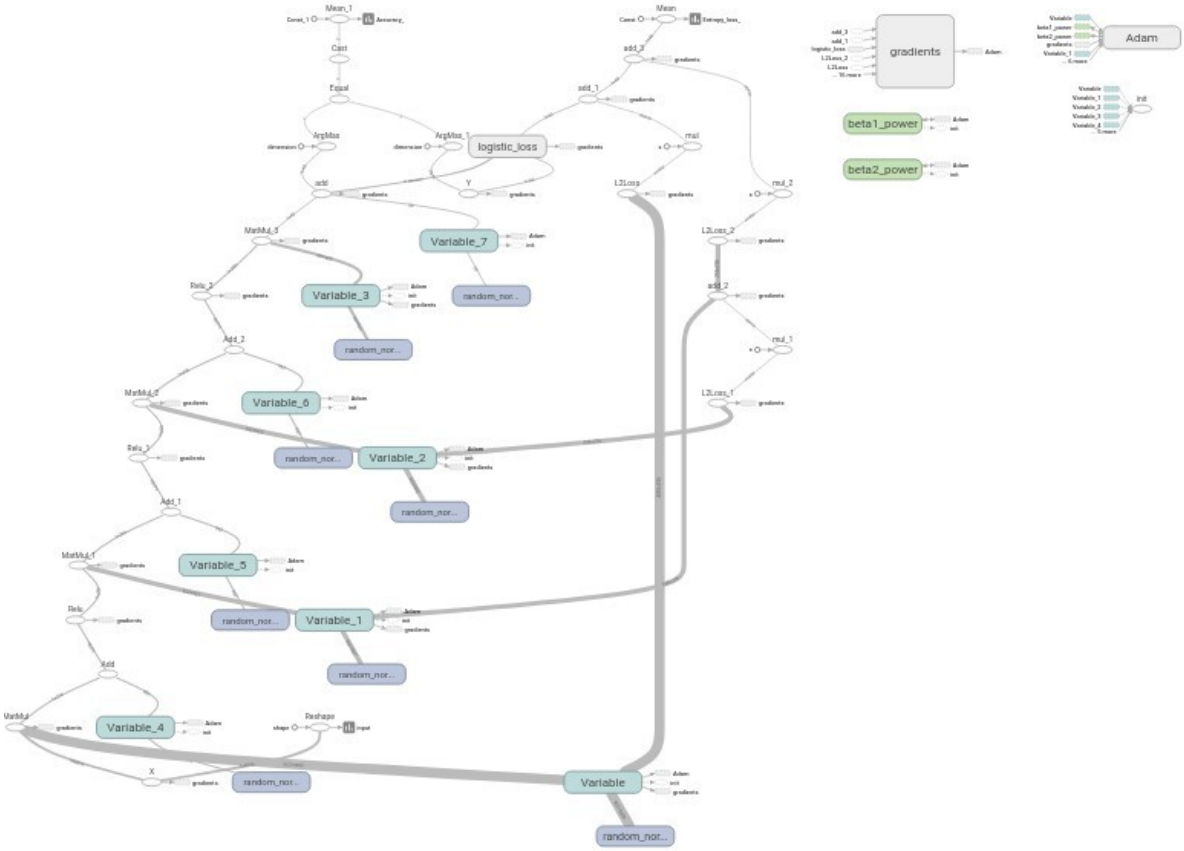

weights_1
weights_1
200
600
-5
-3
-1
1
3
5
weights_2
weights_2
200
600
-7
-5
-3
-1
1
3
5
weights_3
weights_3
200
600
-5
-3
-1
1
3
5

Distribution Biases

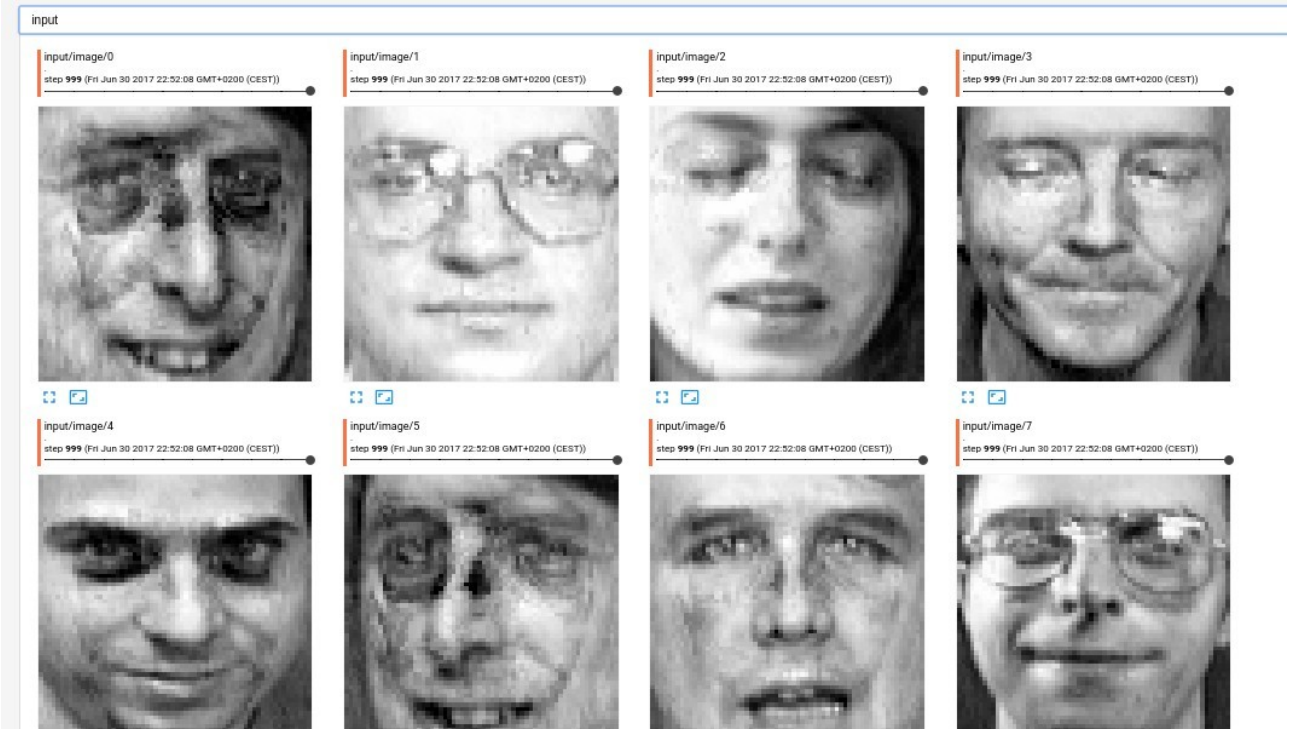
Distribution Weights



Graph :


gradients
Adam
beta1_power
beta2_power
logistic_loss
Variable_7
Variable_3
Variable_6
Variable_2
Variable_5
Variable_1
Variable_4
Variable
random_nor...

Images :


input
input/image/0
step 999 (Fri Jun 30 2017 22:52:08 GMT+0200 (CEST))
input/image/1
step 999 (Fri Jun 30 2017 22:52:08 GMT+0200 (CEST))
input/image/2
step 999 (Fri Jun 30 2017 22:52:08 GMT+0200 (CEST))
input/image/3
step 999 (Fri Jun 30 2017 22:52:08 GMT+0200 (CEST))
input/image/4
step 999 (Fri Jun 30 2017 22:52:08 GMT+0200 (CEST))
input/image/5
step 999 (Fri Jun 30 2017 22:52:08 GMT+0200 (CEST))
input/image/6
step 999 (Fri Jun 30 2017 22:52:08 GMT+0200 (CEST))
input/image/7
step 999 (Fri Jun 30 2017 22:52:08 GMT+0200 (CEST))

Q – 3) CNN layer

output is based on

[Input Layer - Convolutional Layer - Max Pooling Layer - Fully Connected Layer - Output Layer]

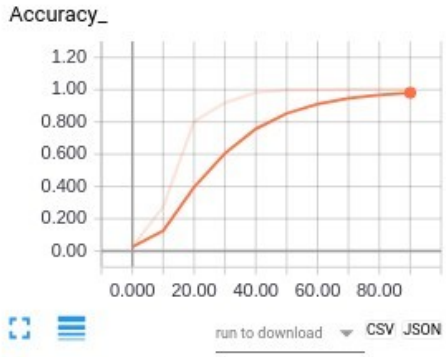
Convolution layer = 5 # Convolution filters are 5 x 5 pixels.
Filter = 16 # There are 16 of these filters.

Convolutional Layer 2.
Convolution Filter = 5 # Convolution filters are 5 x 5 pixels.
Filter for second layer = 36 # There are 36 of these filters.

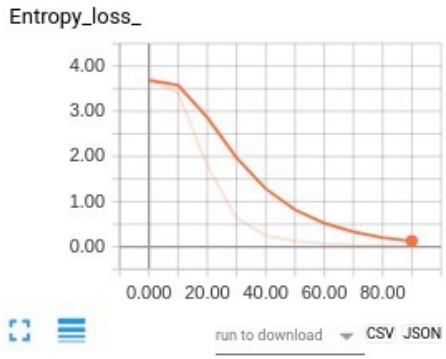
# Fully-connected layer.
FC = 128 (pretty heavy)

And received Pretty good output

Accuracy_

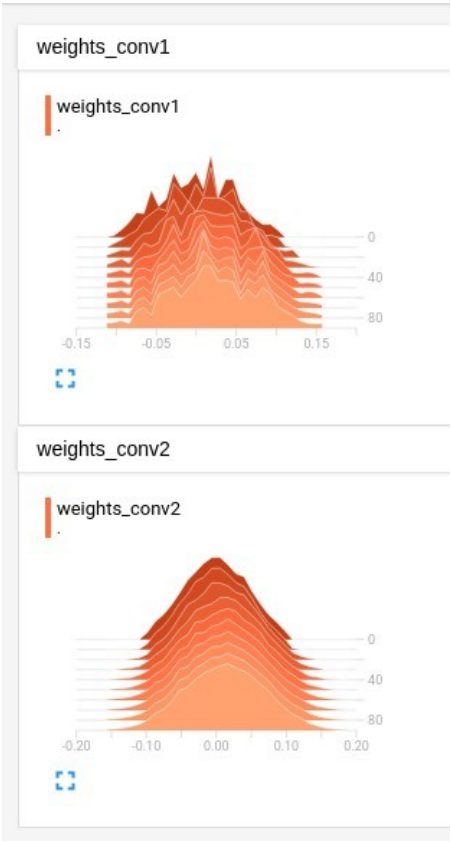


Entropy_loss_

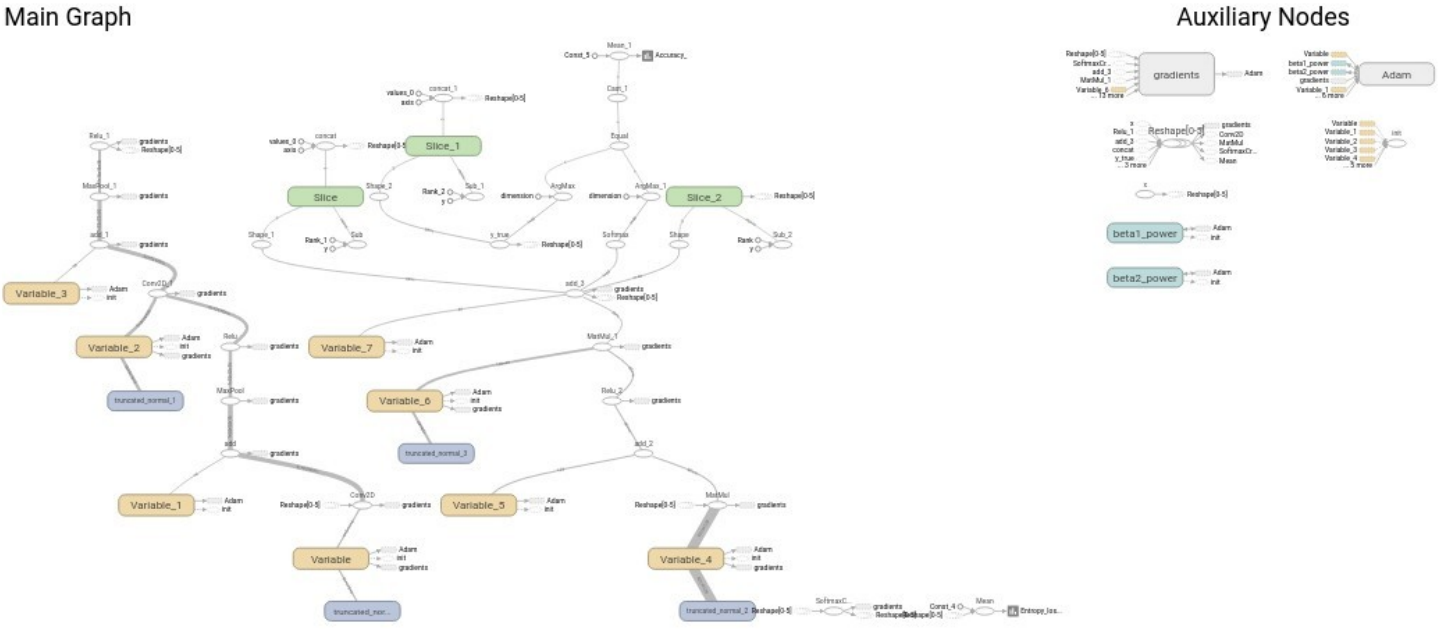


Now weights of layers

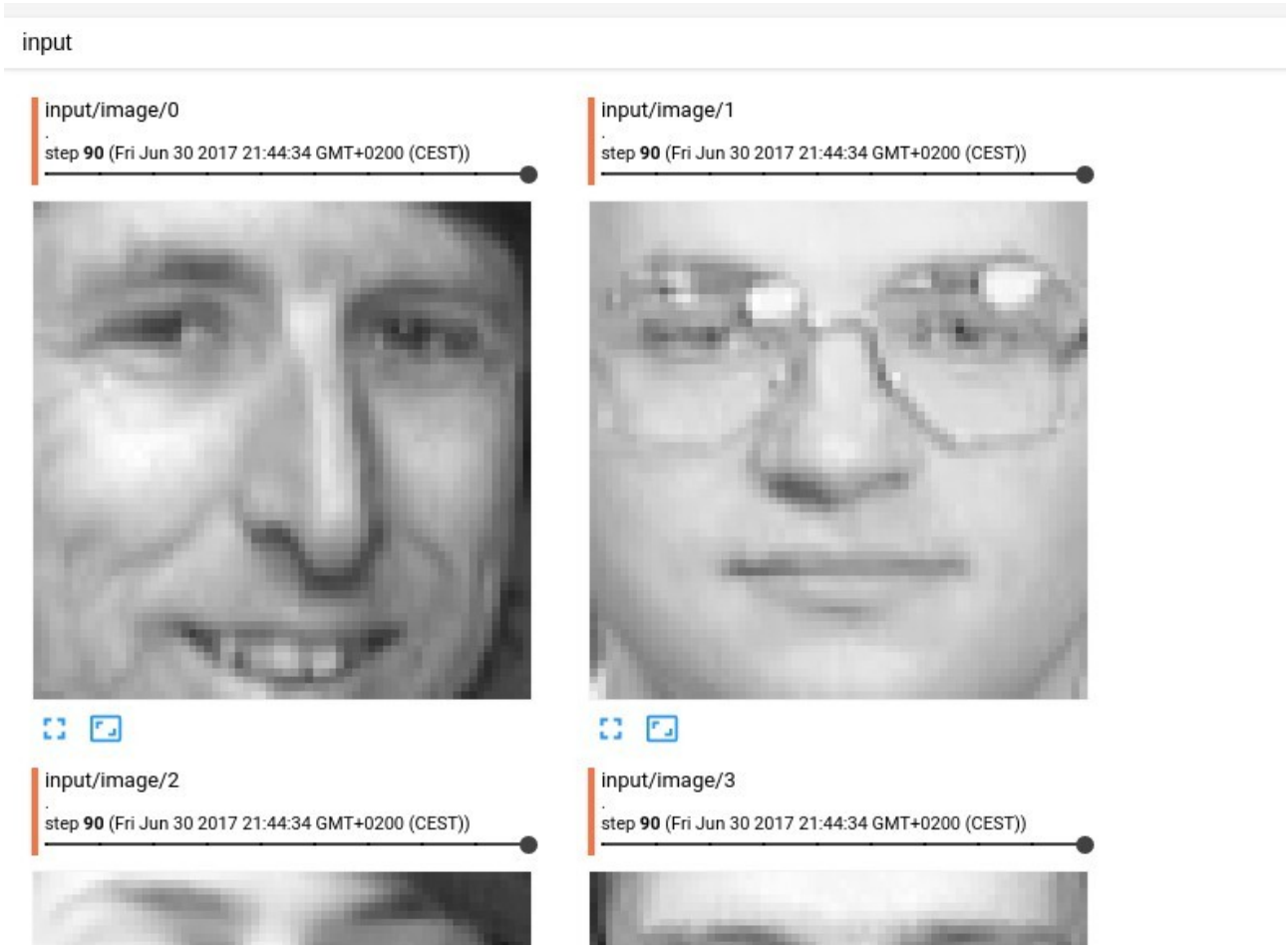
Histograms


weights_conv1
weights_conv1
0
40
80
-0.15
-0.05
0.05
0.15
weights_conv2
weights_conv2
0
40
80
-0.20
-0.10
0.00
0.10
0.20

Graph :



Images :



Q : Bonus Question :

Model Logistic (Reference from previous)
Output [1.0 , 1.0 , 1.0 , 1.0 , 1.0] ~ 1.0
Epochs : 2000
learning Rate : 0.01

Model Neural Network
Output [.97 , 1.0 , .97 , 0.97 , 1.0] ~ 1.0
Epochs : 2000
Number of Nodes : 256
Learning Rage : 0.01
Regularization Rate : 0.01

Model Deep Neural Network
Output [. 75 , .65 , .75 , .65 , .72] ~ 0.70
Epochs : 1000
learning Rage : 0.01
Regularization rate : 0.0001

Model CNN
Output [.97 , 0.97 , 0.97 , 0.97 , 0.97] ~ 0.97
Epochs : 100
learning rate : 0.0001