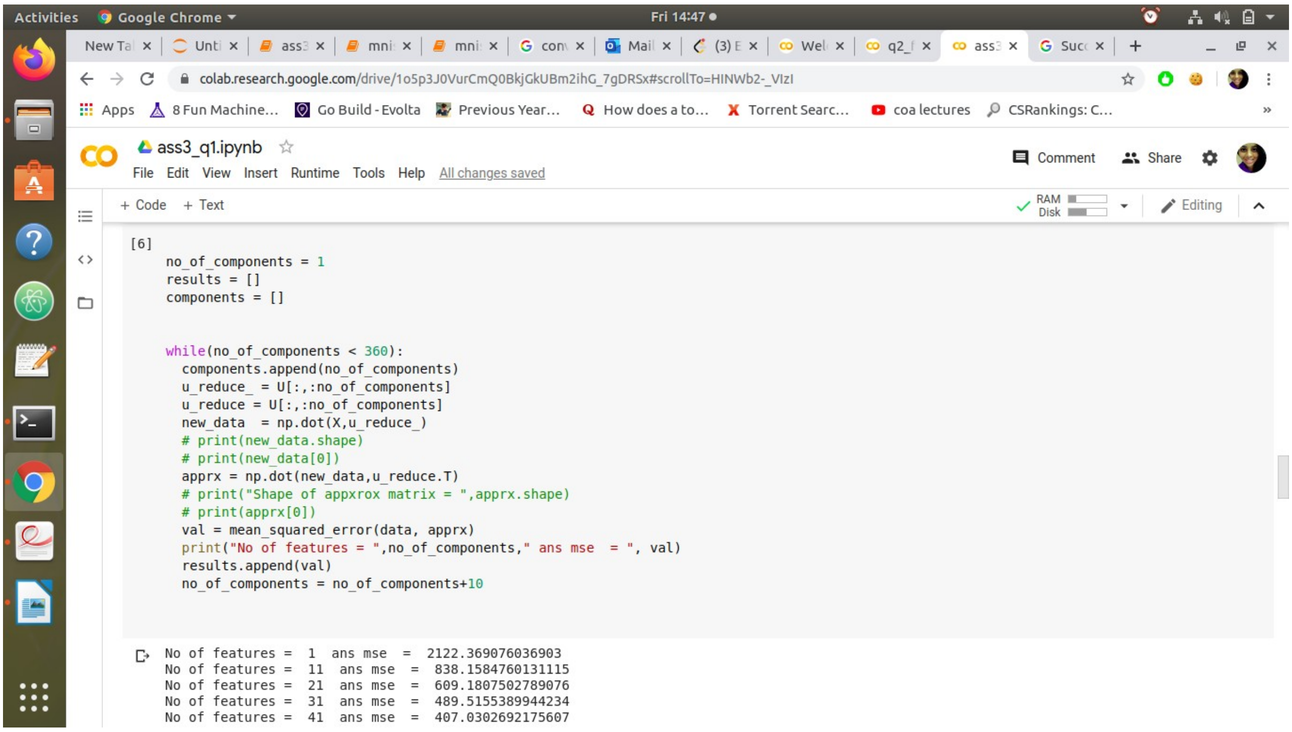
Question 1:

Plotted a graph using different number of components:

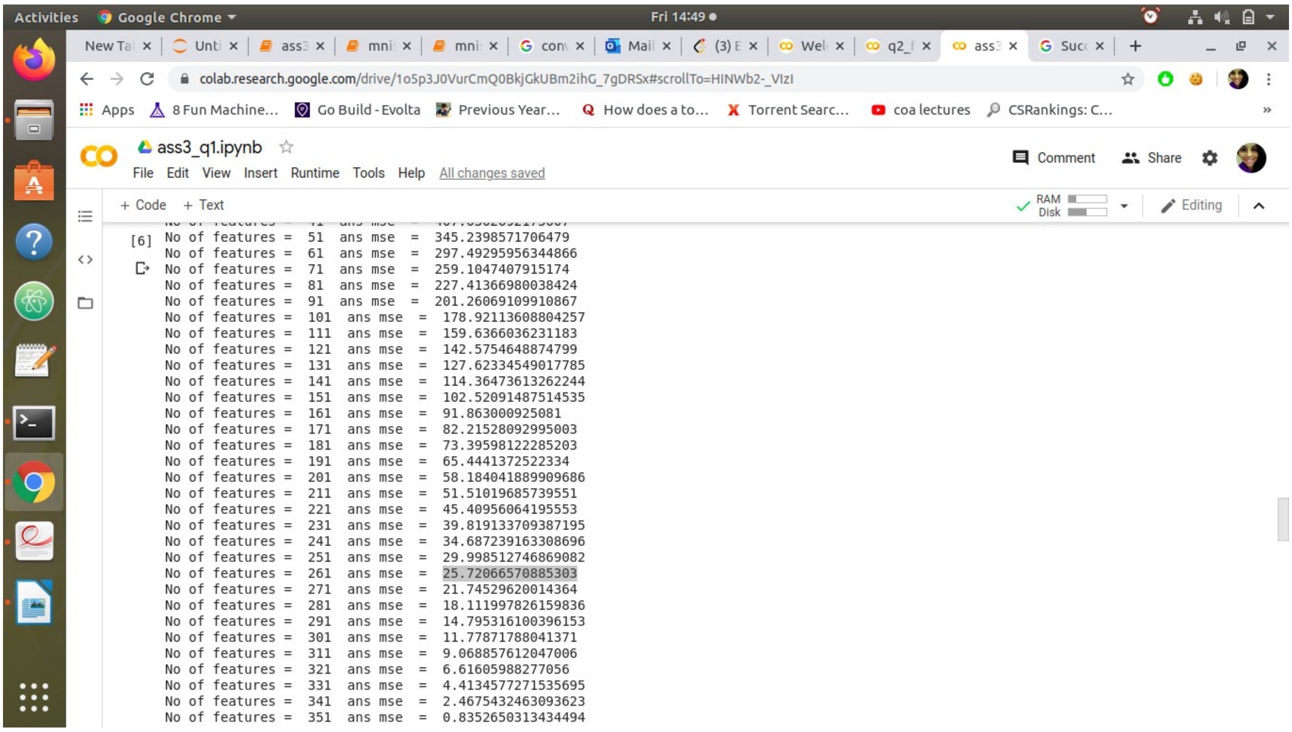
Code snippet showing the number of components being increasing 10 components at a time


Activities
Google Chrome
Fri 14:47
colab.research.google.com/drive/1o5p3J0VurCmQ0BkjGkUBm2ihG_7gDRSx#scrollTo=HINWb2-_VIzI
ass3_q1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing

```
[6]
    no_of_components = 1
    results = []
    components = []


    while(no_of_components < 360):
      components.append(no_of_components)
      u_reduce_ = U[:,:no_of_components]
      u_reduce = U[:,:no_of_components]
      new_data  = np.dot(X,u_reduce_)
      # print(new_data.shape)
      # print(new_data[0])
      apprx = np.dot(new_data,u_reduce.T)
      # print("Shape of appxrox matrix = ",apprx.shape)
      # print(apprx[0])
      val = mean_squared_error(data, apprx)
      print("No of features = ",no_of_components," ans mse  = ", val)
      results.append(val)
      no_of_components = no_of_components+10
```

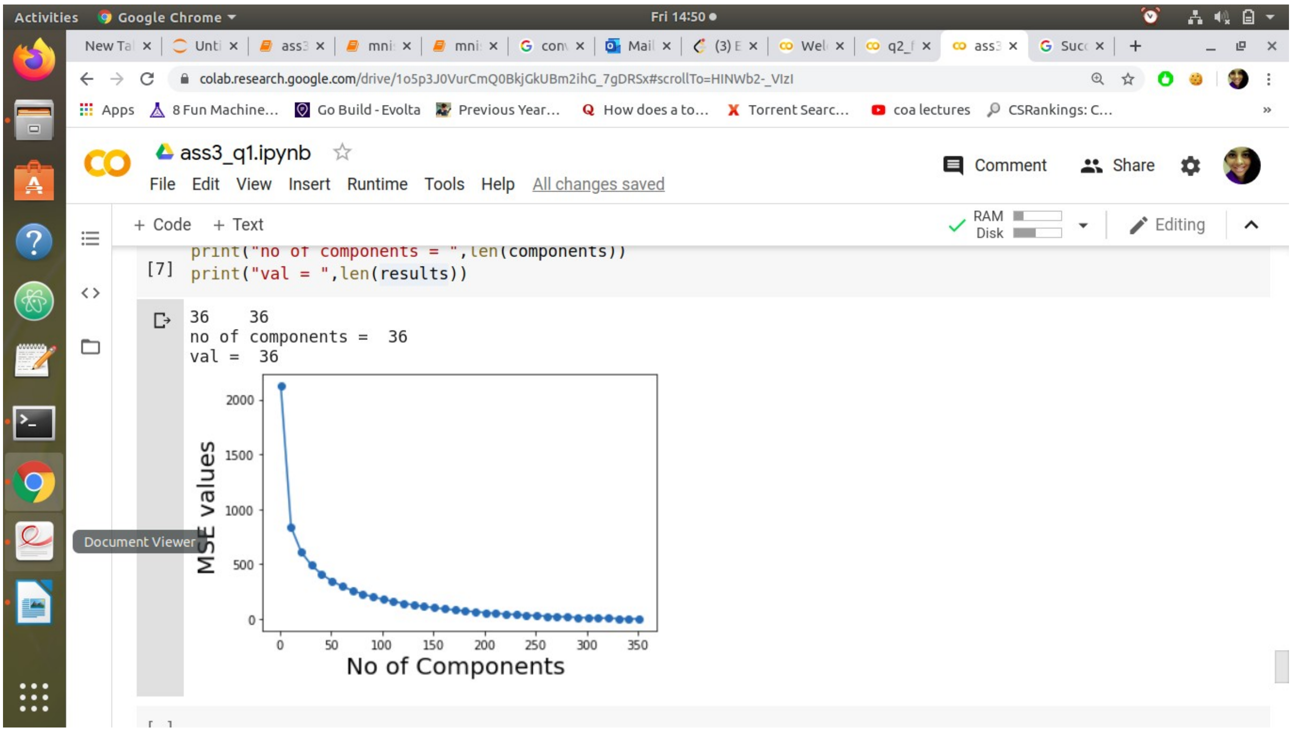
```
No of features =  1  ans mse  =  2122.369076036903
No of features =  11  ans mse  =  838.1584760131115
No of features =  21  ans mse  =  609.1807502789076
No of features =  31  ans mse  =  489.5155389944234
No of features =  41  ans mse  =  407.0302692175607
```

MSE values for different number of components:


Activities
Google Chrome
Fri 14:49
colab.research.google.com/drive/1o5p3J0VurCmQ0BkjGkUBm2ihG_7gDRSx#scrollTo=HINWb2-_VIzI
ass3_q1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
[6]

```
No of features =  51  ans mse  =  345.2398571706479
No of features =  61  ans mse  =  297.49295956344866
No of features =  71  ans mse  =  259.1047407915174
No of features =  81  ans mse  =  227.41366980038424
No of features =  91  ans mse  =  201.26069109910867
No of features =  101  ans mse  =  178.92113608804257
No of features =  111  ans mse  =  159.6366036231183
No of features =  121  ans mse  =  142.5754648874799
No of features =  131  ans mse  =  127.62334549017785
No of features =  141  ans mse  =  114.36473613262244
No of features =  151  ans mse  =  102.52091487514535
No of features =  161  ans mse  =  91.863000925081
No of features =  171  ans mse  =  82.21528092995003
No of features =  181  ans mse  =  73.39598122285203
No of features =  191  ans mse  =  65.4441372522334
No of features =  201  ans mse  =  58.184041889909686
No of features =  211  ans mse  =  51.51019685739551
No of features =  221  ans mse  =  45.40956064195553
No of features =  231  ans mse  =  39.819133709387195
No of features =  241  ans mse  =  34.687239163308696
No of features =  251  ans mse  =  29.998512746869082
No of features =  261  ans mse  =  25.72066570885303
No of features =  271  ans mse  =  21.74529620014364
No of features =  281  ans mse  =  18.111997826159836
No of features =  291  ans mse  =  14.795316100396153
No of features =  301  ans mse  =  11.77871788041371
No of features =  311  ans mse  =  9.068857612047006
No of features =  321  ans mse  =  6.61605988277056
No of features =  331  ans mse  =  4.4134577271535695
No of features =  341  ans mse  =  2.4675432463093623
No of features =  351  ans mse  =  0.8352650313434494
```

Graph :



```
print("no of components = ",len(components))
print("val = ",len(results))
```

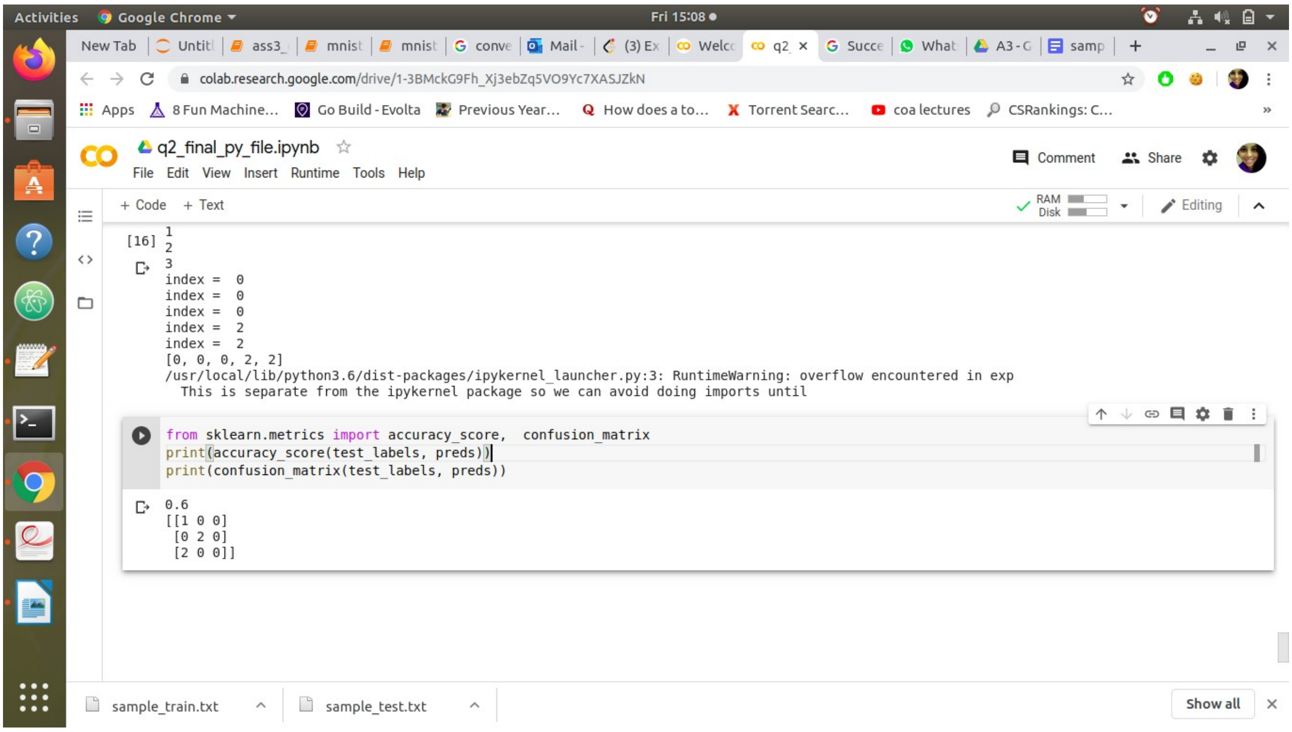
```
36    36
no of components =  36
val =  36
```

N = 261

Question 2:

Accuracy and Confusion Matrix for classification:
(Using Logistic Regression)

(using sample train and test files)



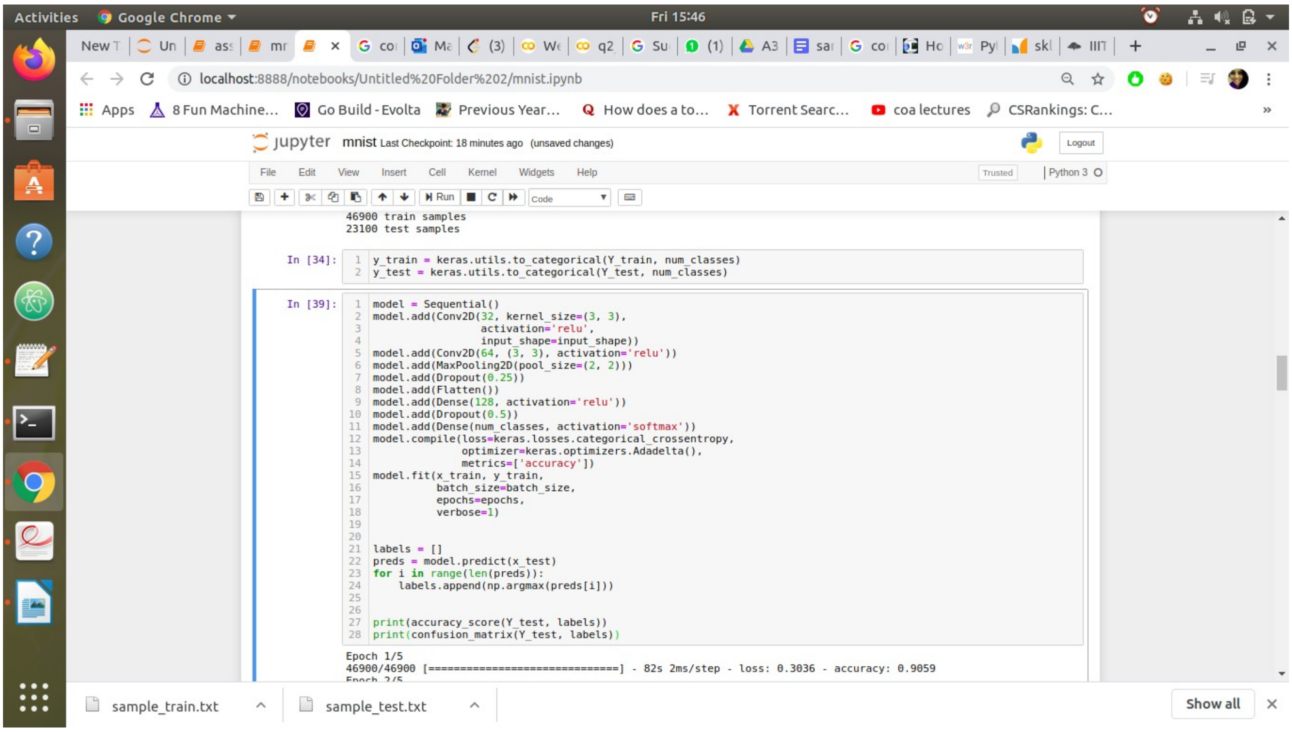
```
[16] 1
     2
     3
     index =  0
     index =  0
     index =  0
     index =  2
     index =  2
     [0, 0, 0, 2, 2]
     /usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:3: RuntimeWarning: overflow encountered in exp
       This is separate from the ipykernel package so we can avoid doing imports until
```

```
from sklearn.metrics import accuracy_score,  confusion_matrix
print(accuracy_score(test_labels, preds))
print(confusion_matrix(test_labels, preds))
```

```
0.6
[[1 0 0]
 [0 2 0]
 [2 0 0]]
```

Question 3:

CNN model:


jupyter mnist Last Checkpoint: 18 minutes ago (unsaved changes)
localhost:8888/notebooks/Untitled%20Folder%202/mnist.ipynb
Python 3
sample_train.txt
sample_test.txt
Show all

```
46900 train samples
23100 test samples
```

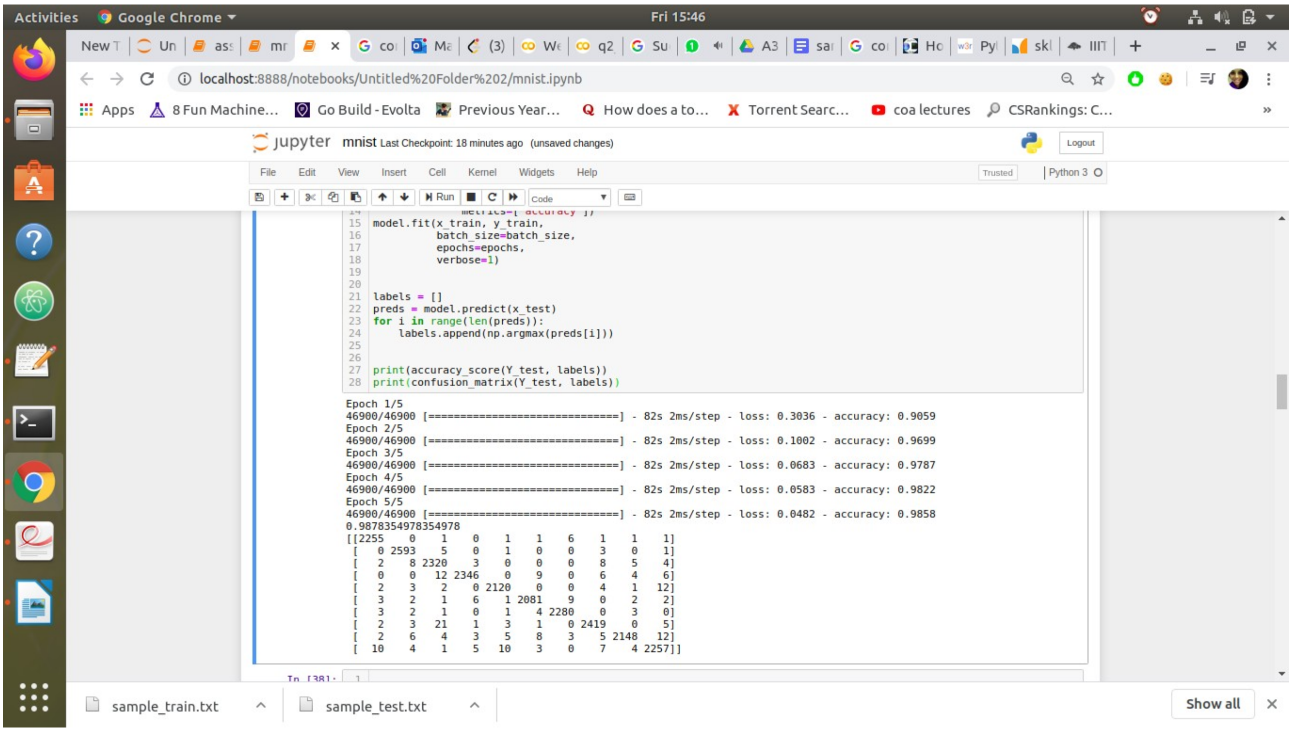
In [34]:

```python
y_train = keras.utils.to_categorical(Y_train, num_classes)
y_test = keras.utils.to_categorical(Y_test, num_classes)
```

In [39]:

```python
model = Sequential()
model.add(Conv2D(32, kernel_size=(3, 3),
                 activation='relu',
                 input_shape=input_shape))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(num_classes, activation='softmax'))
model.compile(loss=keras.losses.categorical_crossentropy,
              optimizer=keras.optimizers.Adadelta(),
              metrics=['accuracy'])
model.fit(x_train, y_train,
          batch_size=batch_size,
          epochs=epochs,
          verbose=1)


labels = []
preds = model.predict(x_test)
for i in range(len(preds)):
    labels.append(np.argmax(preds[i]))


print(accuracy_score(Y_test, labels))
print(confusion_matrix(Y_test, labels))
```

```
Epoch 1/5
46900/46900 [==============================] - 82s 2ms/step - loss: 0.3036 - accuracy: 0.9059
```


jupyter mnist Last Checkpoint: 18 minutes ago (unsaved changes)
localhost:8888/notebooks/Untitled%20Folder%202/mnist.ipynb
Python 3
sample_train.txt
sample_test.txt
Show all

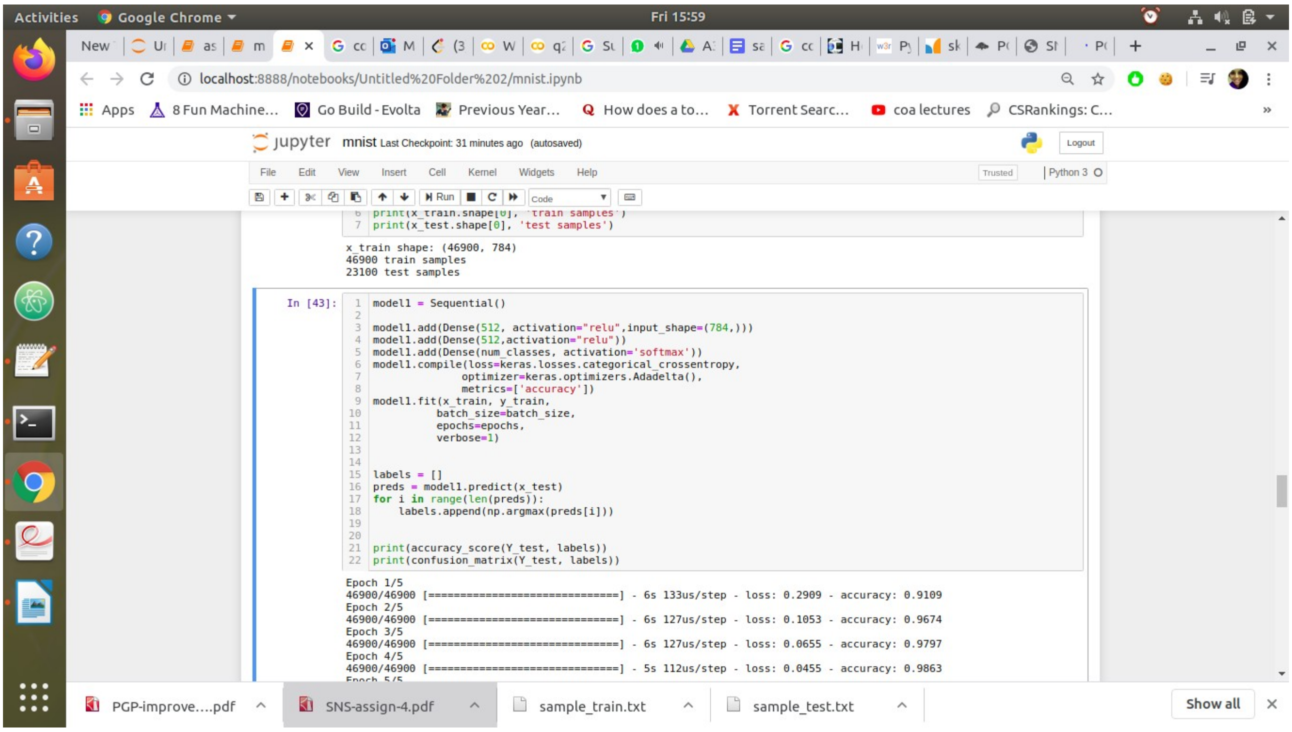
```python
model.fit(x_train, y_train,
          batch_size=batch_size,
          epochs=epochs,
          verbose=1)


labels = []
preds = model.predict(x_test)
for i in range(len(preds)):
    labels.append(np.argmax(preds[i]))


print(accuracy_score(Y_test, labels))
print(confusion_matrix(Y_test, labels))
```

```
Epoch 1/5
46900/46900 [==============================] - 82s 2ms/step - loss: 0.3036 - accuracy: 0.9059
Epoch 2/5
46900/46900 [==============================] - 82s 2ms/step - loss: 0.1002 - accuracy: 0.9699
Epoch 3/5
46900/46900 [==============================] - 82s 2ms/step - loss: 0.0683 - accuracy: 0.9787
Epoch 4/5
46900/46900 [==============================] - 82s 2ms/step - loss: 0.0583 - accuracy: 0.9822
Epoch 5/5
46900/46900 [==============================] - 82s 2ms/step - loss: 0.0482 - accuracy: 0.9858
0.9878354978354978
[[2255    0    1    0    1    1    6    1    1    1]
 [   0 2593    5    0    1    0    0    3    0    1]
 [   2    8 2320    3    0    0    0    8    5    4]
 [   0    0   12 2346    0    9    0    6    4    6]
 [   2    3    2    0 2120    0    0    4    1   12]
 [   3    2    1    6    1 2081    9    0    2    2]
 [   3    2    1    0    1    4 2280    0    3    0]
 [   2    3   21    1    3    1    0 2419    0    5]
 [   2    6    4    3    5    8    3    5 2148   12]
 [  10    4    1    5   10    3    0    7    4 2257]]
```

In [38]:

MLP:

model:


localhost:8888/notebooks/Untitled%20Folder%202/mnist.ipynb
jupyter mnist Last Checkpoint: 31 minutes ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
PGP-improve....pdf
SNS-assign-4.pdf
sample_train.txt
sample_test.txt
Show all

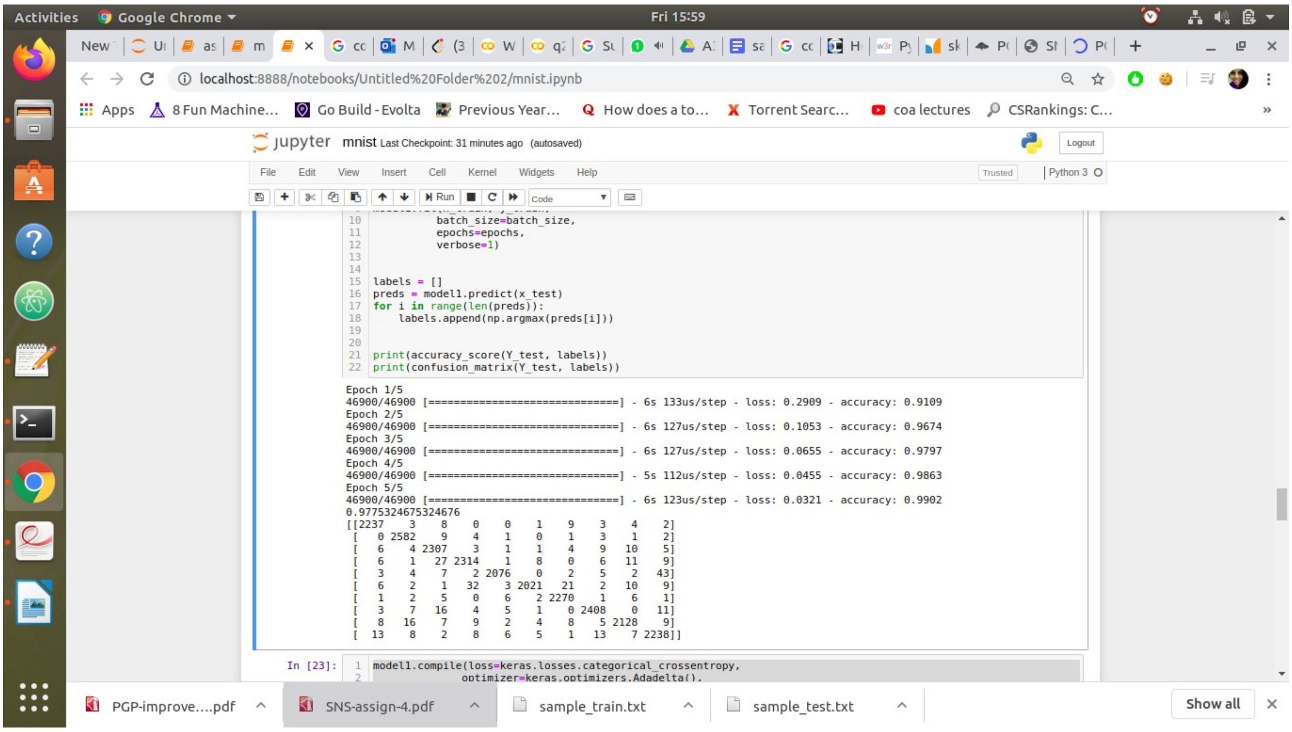
```
print(x_train.shape[0], 'train samples')
print(x_test.shape[0], 'test samples')
```

```
x_train shape: (46900, 784)
46900 train samples
23100 test samples
```

In [43]:

```
model1 = Sequential()

model1.add(Dense(512, activation="relu",input_shape=(784,)))
model1.add(Dense(512,activation="relu"))
model1.add(Dense(num_classes, activation='softmax'))
model1.compile(loss=keras.losses.categorical_crossentropy,
              optimizer=keras.optimizers.Adadelta(),
              metrics=['accuracy'])
model1.fit(x_train, y_train,
          batch_size=batch_size,
          epochs=epochs,
          verbose=1)


labels = []
preds = model1.predict(x_test)
for i in range(len(preds)):
    labels.append(np.argmax(preds[i]))


print(accuracy_score(Y_test, labels))
print(confusion_matrix(Y_test, labels))
```

```
Epoch 1/5
46900/46900 [==============================] - 6s 133us/step - loss: 0.2909 - accuracy: 0.9109
Epoch 2/5
46900/46900 [==============================] - 6s 127us/step - loss: 0.1053 - accuracy: 0.9674
Epoch 3/5
46900/46900 [==============================] - 6s 127us/step - loss: 0.0655 - accuracy: 0.9797
Epoch 4/5
46900/46900 [==============================] - 5s 112us/step - loss: 0.0455 - accuracy: 0.9863
Epoch 5/5
```


localhost:8888/notebooks/Untitled%20Folder%202/mnist.ipynb
jupyter mnist Last Checkpoint: 31 minutes ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
PGP-improve....pdf
SNS-assign-4.pdf
sample_train.txt
sample_test.txt
Show all

```
          batch_size=batch_size,
          epochs=epochs,
          verbose=1)


labels = []
preds = model1.predict(x_test)
for i in range(len(preds)):
    labels.append(np.argmax(preds[i]))


print(accuracy_score(Y_test, labels))
print(confusion_matrix(Y_test, labels))
```

```
Epoch 1/5
46900/46900 [==============================] - 6s 133us/step - loss: 0.2909 - accuracy: 0.9109
Epoch 2/5
46900/46900 [==============================] - 6s 127us/step - loss: 0.1053 - accuracy: 0.9674
Epoch 3/5
46900/46900 [==============================] - 6s 127us/step - loss: 0.0655 - accuracy: 0.9797
Epoch 4/5
46900/46900 [==============================] - 5s 112us/step - loss: 0.0455 - accuracy: 0.9863
Epoch 5/5
46900/46900 [==============================] - 6s 123us/step - loss: 0.0321 - accuracy: 0.9902
0.9775324675324676
[[2237    3    8    0    0    1    9    3    4    2]
 [   0 2582    9    4    1    0    1    3    1    2]
 [   6    4 2307    3    1    1    4    9   10    5]
 [   6    1   27 2314    1    8    0    6   11    9]
 [   3    4    7    2 2076    0    2    5    2   43]
 [   6    2    1   32    3 2021   21    2   10    9]
 [   1    2    5    0    6    2 2270    1    6    1]
 [   3    7   16    4    5    1    0 2408    0   11]
 [   8   16    7    9    2    4    8    5 2128    9]
 [  13    8    2    8    6    5    1   13    7 2238]]
```

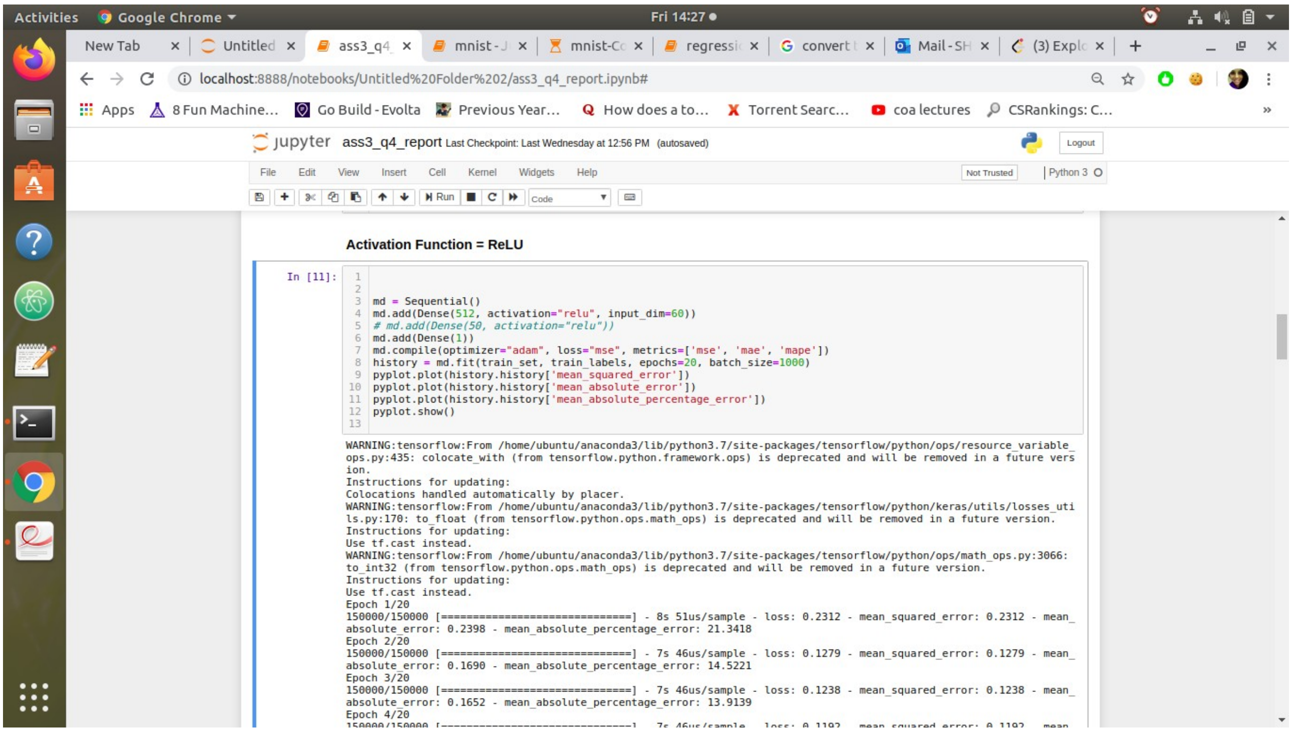
In [23]:

```
model1.compile(loss=keras.losses.categorical_crossentropy,
              optimizer=keras.optimizers.Adadelta(),
```

Question 4:

Different Activation Functions:

1. ReLU

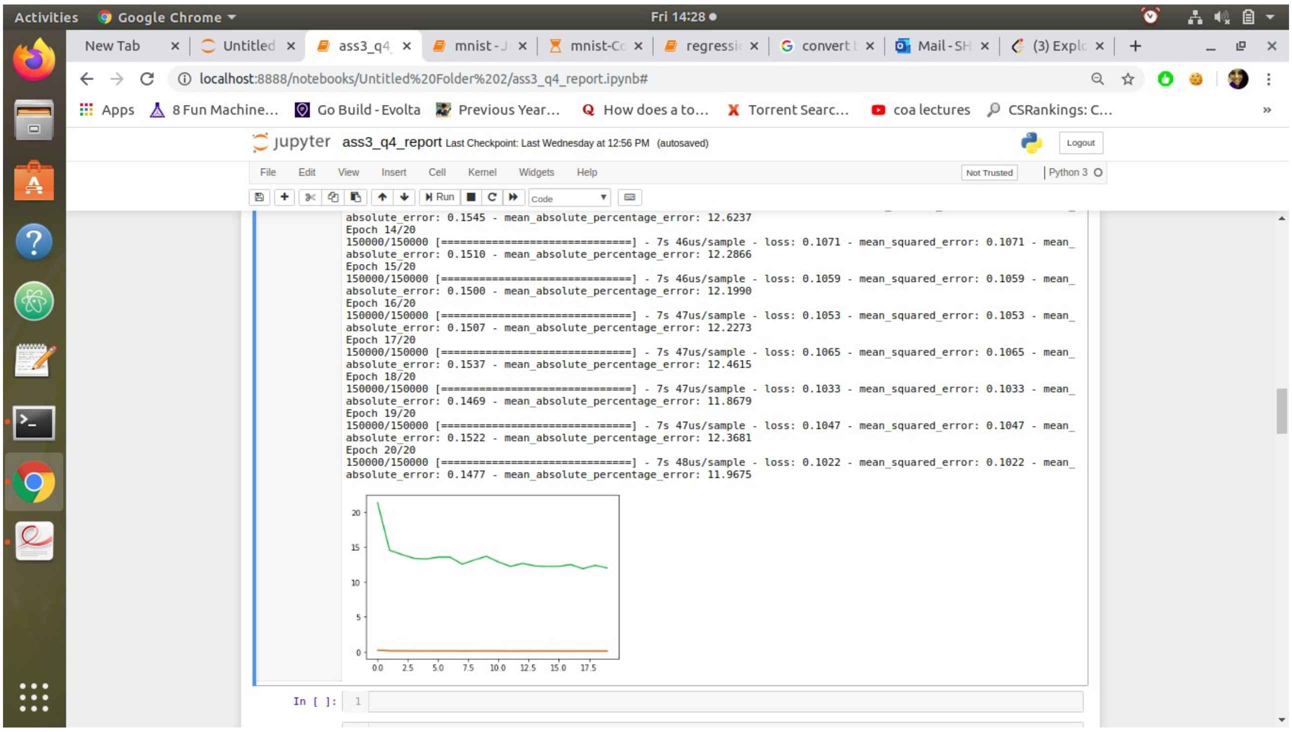


Activation Function = ReLU

In [11]:

```


md = Sequential()
md.add(Dense(512, activation="relu", input_dim=60))
# md.add(Dense(50, activation="relu"))
md.add(Dense(1))
md.compile(optimizer="adam", loss="mse", metrics=['mse', 'mae', 'mape'])
history = md.fit(train_set, train_labels, epochs=20, batch_size=1000)
pyplot.plot(history.history['mean_squared_error'])
pyplot.plot(history.history['mean_absolute_error'])
pyplot.plot(history.history['mean_absolute_percentage_error'])
pyplot.show()

```

```
WARNING:tensorflow:From /home/ubuntu/anaconda3/lib/python3.7/site-packages/tensorflow/python/ops/resource_variable_
ops.py:435: colocate_with (from tensorflow.python.framework.ops) is deprecated and will be removed in a future vers
ion.
Instructions for updating:
Colocations handled automatically by placer.
WARNING:tensorflow:From /home/ubuntu/anaconda3/lib/python3.7/site-packages/tensorflow/python/keras/utils/losses_uti
ls.py:170: to_float (from tensorflow.python.ops.math_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.cast instead.
WARNING:tensorflow:From /home/ubuntu/anaconda3/lib/python3.7/site-packages/tensorflow/python/ops/math_ops.py:3066:
to_int32 (from tensorflow.python.ops.math_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.cast instead.
Epoch 1/20
150000/150000 [==============================] - 8s 51us/sample - loss: 0.2312 - mean_squared_error: 0.2312 - mean_
absolute_error: 0.2398 - mean_absolute_percentage_error: 21.3418
Epoch 2/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1279 - mean_squared_error: 0.1279 - mean_
absolute_error: 0.1690 - mean_absolute_percentage_error: 14.5221
Epoch 3/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1238 - mean_squared_error: 0.1238 - mean_
absolute_error: 0.1652 - mean_absolute_percentage_error: 13.9139
Epoch 4/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1192 - mean_squared_error: 0.1192 - mean_
```

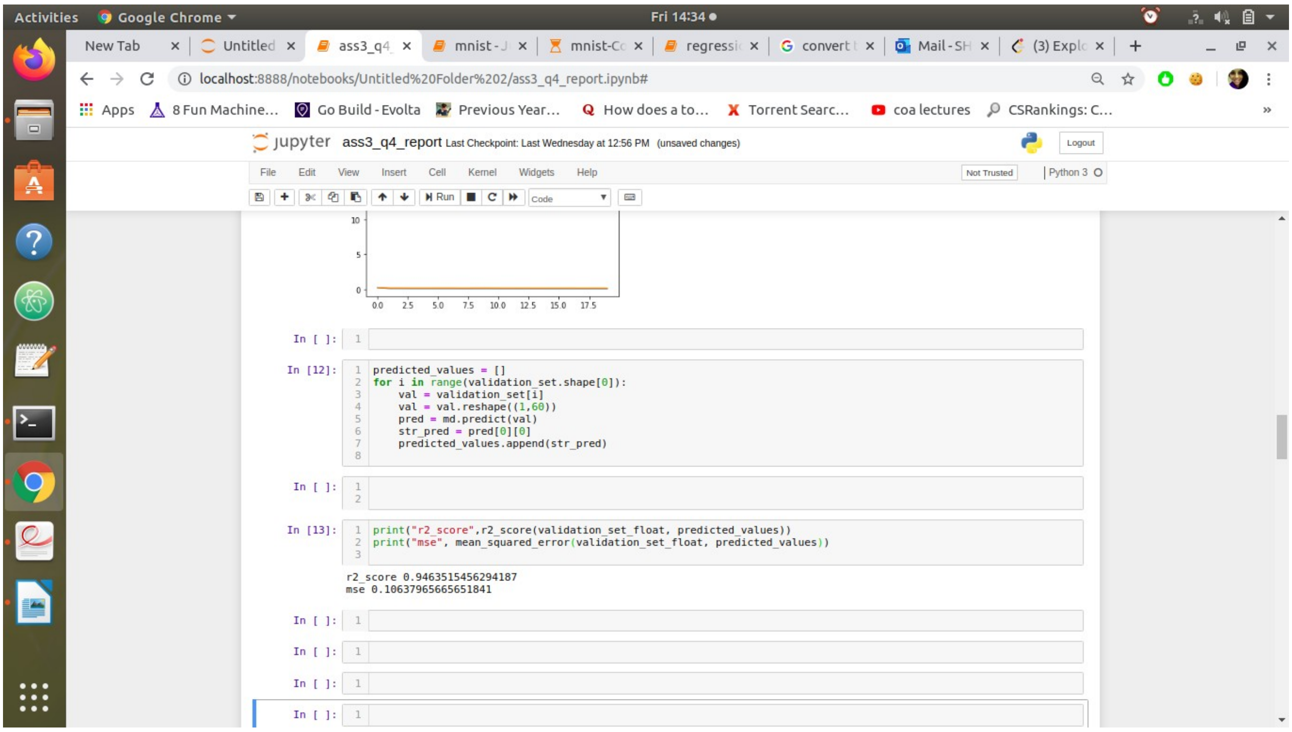


```
absolute_error: 0.1545 - mean_absolute_percentage_error: 12.6237
Epoch 14/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1071 - mean_squared_error: 0.1071 - mean_
absolute_error: 0.1510 - mean_absolute_percentage_error: 12.2866
Epoch 15/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1059 - mean_squared_error: 0.1059 - mean_
absolute_error: 0.1500 - mean_absolute_percentage_error: 12.1990
Epoch 16/20
150000/150000 [==============================] - 7s 47us/sample - loss: 0.1053 - mean_squared_error: 0.1053 - mean_
absolute_error: 0.1507 - mean_absolute_percentage_error: 12.2273
Epoch 17/20
150000/150000 [==============================] - 7s 47us/sample - loss: 0.1065 - mean_squared_error: 0.1065 - mean_
absolute_error: 0.1537 - mean_absolute_percentage_error: 12.4615
Epoch 18/20
150000/150000 [==============================] - 7s 47us/sample - loss: 0.1033 - mean_squared_error: 0.1033 - mean_
absolute_error: 0.1469 - mean_absolute_percentage_error: 11.8679
Epoch 19/20
150000/150000 [==============================] - 7s 47us/sample - loss: 0.1047 - mean_squared_error: 0.1047 - mean_
absolute_error: 0.1522 - mean_absolute_percentage_error: 12.3681
Epoch 20/20
150000/150000 [==============================] - 7s 48us/sample - loss: 0.1022 - mean_squared_error: 0.1022 - mean_
absolute_error: 0.1477 - mean_absolute_percentage_error: 11.9675
```

In [ ]:

Activities
Google Chrome
Fri 14:28
localhost:8888/notebooks/Untitled%20Folder%202/ass3_q4_report.ipynb#
jupyter ass3_q4_report Last Checkpoint: Last Wednesday at 12:56 PM (autosaved)
Logout
Not Trusted
Python 3
Run
Code
20
15
10
5
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

Accuracy:

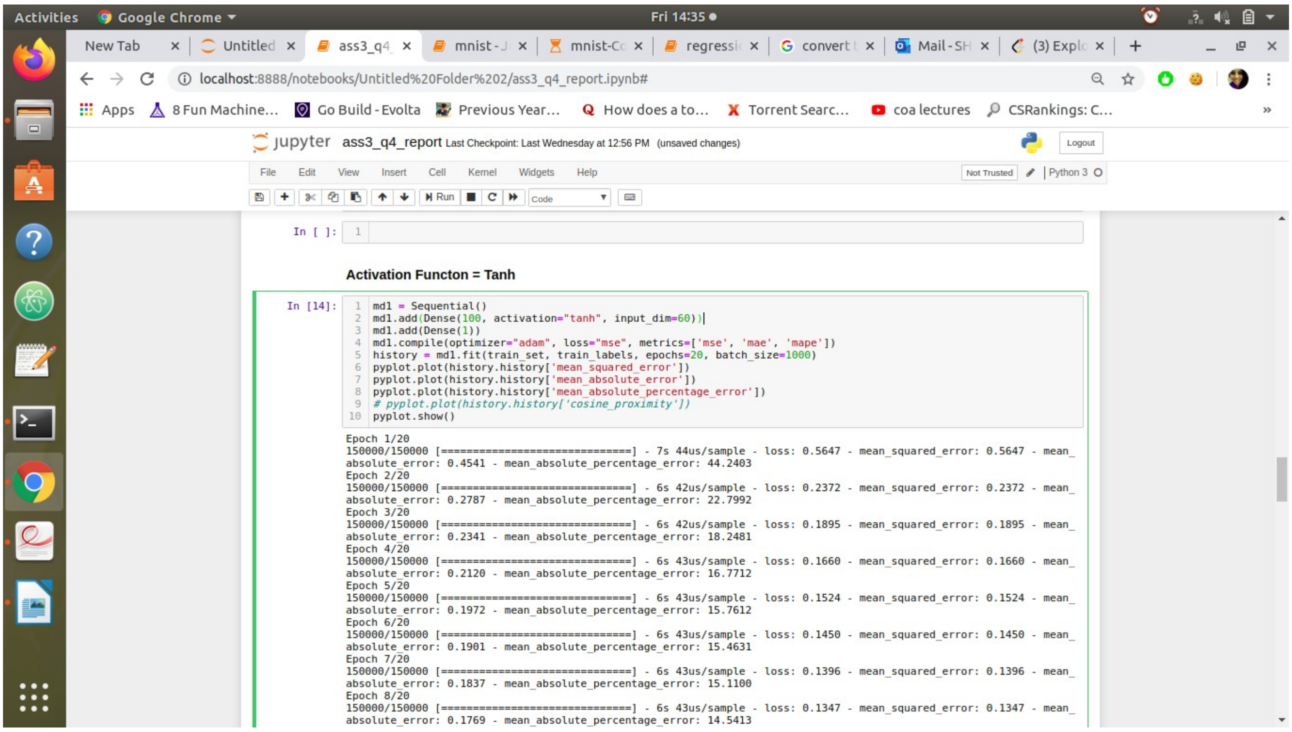

Activities
Google Chrome
Fri 14:34
localhost:8888/notebooks/Untitled%20Folder%202/ass3_q4_report.ipynb#
jupyter ass3_q4_report Last Checkpoint: Last Wednesday at 12:56 PM (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code
10
5
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

```
In [12]:
predicted_values = []
for i in range(validation_set.shape[0]):
    val = validation_set[i]
    val = val.reshape((1,60))
    pred = md.predict(val)
    str_pred = pred[0][0]
    predicted_values.append(str_pred)
```

```
In [13]:
print("r2_score",r2_score(validation_set_float, predicted_values))
print("mse", mean_squared_error(validation_set_float, predicted_values))
```

```
r2_score 0.9463515456294187
mse 0.10637965665651841
```

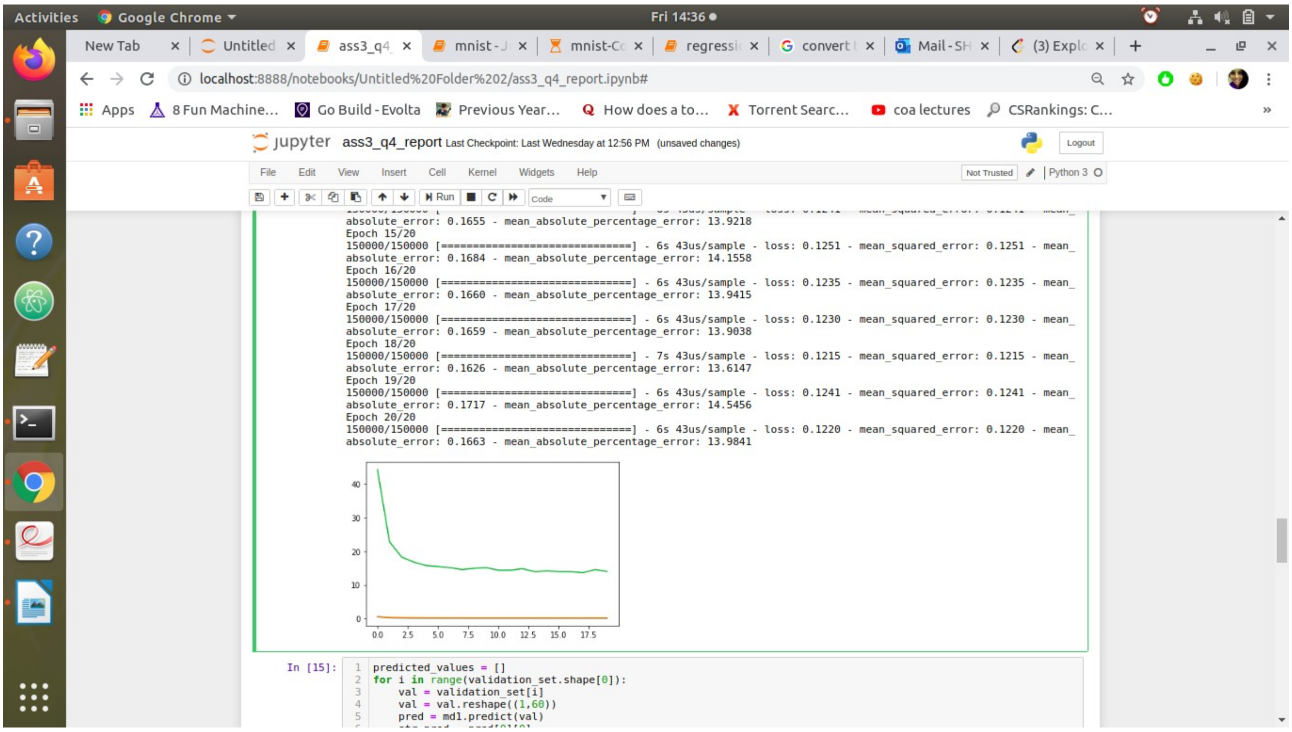
Activation Function: Tanh


Activities
Google Chrome
Fri 14:35
localhost:8888/notebooks/Untitled%20Folder%202/ass3_q4_report.ipynb#
jupyter ass3_q4_report Last Checkpoint: Last Wednesday at 12:56 PM (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code

Activation Functon = Tanh

```
In [14]:
md1 = Sequential()
md1.add(Dense(100, activation="tanh", input_dim=60))
md1.add(Dense(1))
md1.compile(optimizer="adam", loss="mse", metrics=['mse', 'mae', 'mape'])
history = md1.fit(train_set, train_labels, epochs=20, batch_size=1000)
pyplot.plot(history.history['mean_squared_error'])
pyplot.plot(history.history['mean_absolute_error'])
pyplot.plot(history.history['mean_absolute_percentage_error'])
# pyplot.plot(history.history['cosine_proximity'])
pyplot.show()
```

```
Epoch 1/20
150000/150000 [==============================] - 7s 44us/sample - loss: 0.5647 - mean_squared_error: 0.5647 - mean_
absolute_error: 0.4541 - mean_absolute_percentage_error: 44.2403
Epoch 2/20
150000/150000 [==============================] - 6s 42us/sample - loss: 0.2372 - mean_squared_error: 0.2372 - mean_
absolute_error: 0.2787 - mean_absolute_percentage_error: 22.7992
Epoch 3/20
150000/150000 [==============================] - 6s 42us/sample - loss: 0.1895 - mean_squared_error: 0.1895 - mean_
absolute_error: 0.2341 - mean_absolute_percentage_error: 18.2481
Epoch 4/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1660 - mean_squared_error: 0.1660 - mean_
absolute_error: 0.2120 - mean_absolute_percentage_error: 16.7712
Epoch 5/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1524 - mean_squared_error: 0.1524 - mean_
absolute_error: 0.1972 - mean_absolute_percentage_error: 15.7612
Epoch 6/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1450 - mean_squared_error: 0.1450 - mean_
absolute_error: 0.1901 - mean_absolute_percentage_error: 15.4631
Epoch 7/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1396 - mean_squared_error: 0.1396 - mean_
absolute_error: 0.1837 - mean_absolute_percentage_error: 15.1100
Epoch 8/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1347 - mean_squared_error: 0.1347 - mean_
absolute_error: 0.1769 - mean_absolute_percentage_error: 14.5413
```

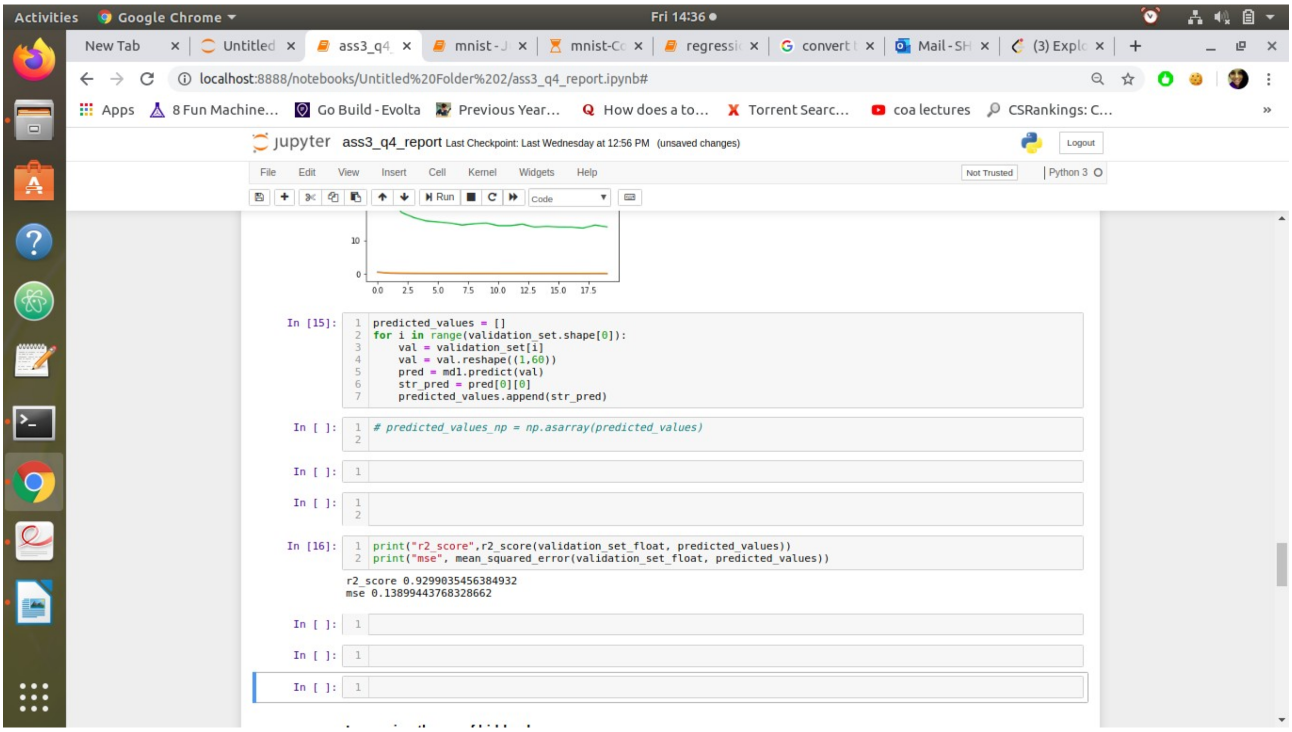


jupyter ass3_q4_report Last Checkpoint: Last Wednesday at 12:56 PM (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
40
30
20
10
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

```
absolute_error: 0.1655 - mean_absolute_percentage_error: 13.9218
Epoch 15/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1251 - mean_squared_error: 0.1251 - mean_
absolute_error: 0.1684 - mean_absolute_percentage_error: 14.1558
Epoch 16/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1235 - mean_squared_error: 0.1235 - mean_
absolute_error: 0.1660 - mean_absolute_percentage_error: 13.9415
Epoch 17/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1230 - mean_squared_error: 0.1230 - mean_
absolute_error: 0.1659 - mean_absolute_percentage_error: 13.9038
Epoch 18/20
150000/150000 [==============================] - 7s 43us/sample - loss: 0.1215 - mean_squared_error: 0.1215 - mean_
absolute_error: 0.1626 - mean_absolute_percentage_error: 13.6147
Epoch 19/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1241 - mean_squared_error: 0.1241 - mean_
absolute_error: 0.1717 - mean_absolute_percentage_error: 14.5456
Epoch 20/20
150000/150000 [==============================] - 6s 43us/sample - loss: 0.1220 - mean_squared_error: 0.1220 - mean_
absolute_error: 0.1663 - mean_absolute_percentage_error: 13.9841
```

```
In [15]:
predicted_values = []
for i in range(validation_set.shape[0]):
    val = validation_set[i]
    val = val.reshape((1,60))
    pred = mdl.predict(val)
```

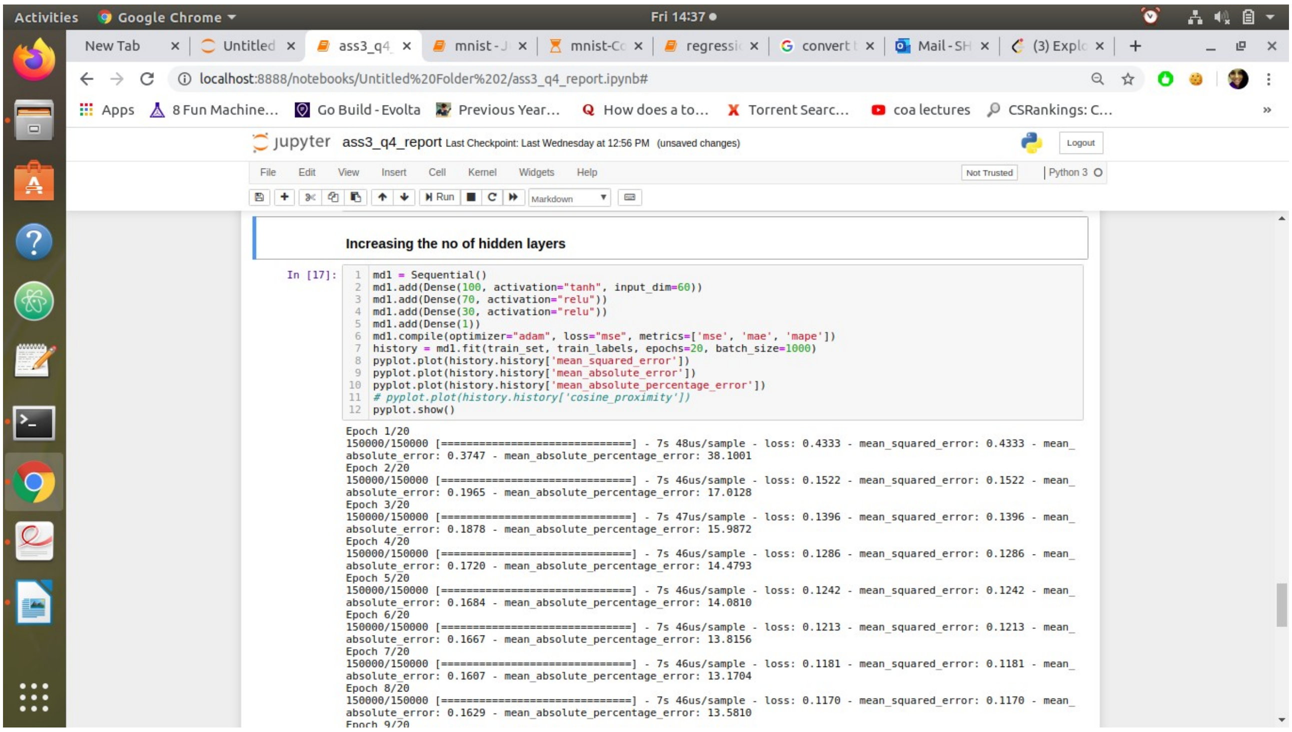
Accuracy:



jupyter ass3_q4_report Last Checkpoint: Last Wednesday at 12:56 PM (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
10
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

```
In [15]:
predicted_values = []
for i in range(validation_set.shape[0]):
    val = validation_set[i]
    val = val.reshape((1,60))
    pred = mdl.predict(val)
    str_pred = pred[0][0]
    predicted_values.append(str_pred)

In [ ]:
# predicted_values_np = np.asarray(predicted_values)

In [ ]:

In [ ]:

In [16]:
print("r2_score",r2_score(validation_set_float, predicted_values))
print("mse", mean_squared_error(validation_set_float, predicted_values))

r2_score 0.9299035456384932
mse 0.13899443768328662

In [ ]:

In [ ]:

In [ ]:
```

Increasing number of hidden layers and using 2 different activation functions (relu and tanh) and showing the accuracy:

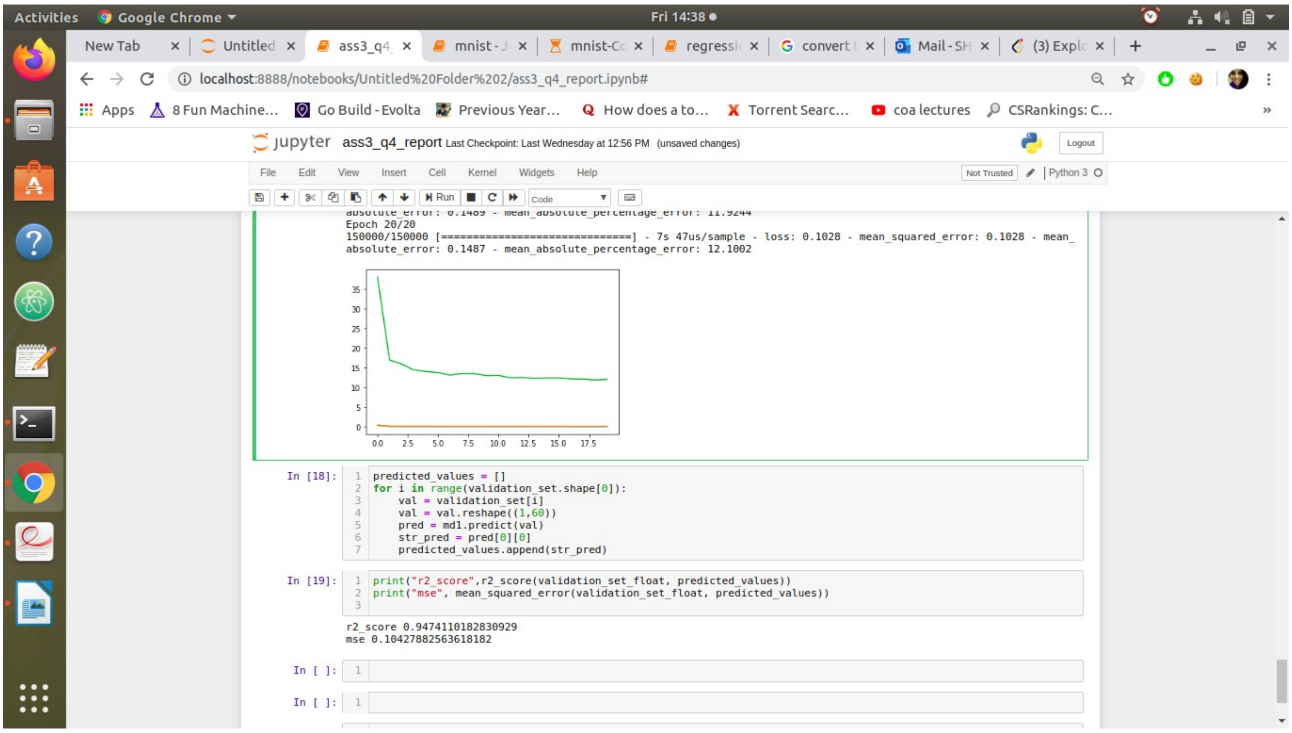

Activities
Google Chrome
Fri 14:37
localhost:8888/notebooks/Untitled%20Folder%202/ass3_q4_report.ipynb#
jupyter ass3_q4_report Last Checkpoint: Last Wednesday at 12:56 PM (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Markdown

Increasing the no of hidden layers

In [17]:

```
mdl = Sequential()
mdl.add(Dense(100, activation="tanh", input_dim=60))
mdl.add(Dense(70, activation="relu"))
mdl.add(Dense(30, activation="relu"))
mdl.add(Dense(1))
mdl.compile(optimizer="adam", loss="mse", metrics=['mse', 'mae', 'mape'])
history = mdl.fit(train_set, train_labels, epochs=20, batch_size=1000)
pyplot.plot(history.history['mean_squared_error'])
pyplot.plot(history.history['mean_absolute_error'])
pyplot.plot(history.history['mean_absolute_percentage_error'])
# pyplot.plot(history.history['cosine_proximity'])
pyplot.show()
```

```
Epoch 1/20
150000/150000 [==============================] - 7s 48us/sample - loss: 0.4333 - mean_squared_error: 0.4333 - mean_
absolute_error: 0.3747 - mean_absolute_percentage_error: 38.1001
Epoch 2/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1522 - mean_squared_error: 0.1522 - mean_
absolute_error: 0.1965 - mean_absolute_percentage_error: 17.0128
Epoch 3/20
150000/150000 [==============================] - 7s 47us/sample - loss: 0.1396 - mean_squared_error: 0.1396 - mean_
absolute_error: 0.1878 - mean_absolute_percentage_error: 15.9872
Epoch 4/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1286 - mean_squared_error: 0.1286 - mean_
absolute_error: 0.1720 - mean_absolute_percentage_error: 14.4793
Epoch 5/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1242 - mean_squared_error: 0.1242 - mean_
absolute_error: 0.1684 - mean_absolute_percentage_error: 14.0810
Epoch 6/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1213 - mean_squared_error: 0.1213 - mean_
absolute_error: 0.1667 - mean_absolute_percentage_error: 13.8156
Epoch 7/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1181 - mean_squared_error: 0.1181 - mean_
absolute_error: 0.1607 - mean_absolute_percentage_error: 13.1704
Epoch 8/20
150000/150000 [==============================] - 7s 46us/sample - loss: 0.1170 - mean_squared_error: 0.1170 - mean_
absolute_error: 0.1629 - mean_absolute_percentage_error: 13.5810
Epoch 9/20
```


Activities
Google Chrome
Fri 14:38
localhost:8888/notebooks/Untitled%20Folder%202/ass3_q4_report.ipynb#
jupyter ass3_q4_report Last Checkpoint: Last Wednesday at 12:56 PM (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code
35
30
25
20
15
10
5
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

```
absolute_error: 0.1489 - mean_absolute_percentage_error: 11.9244
Epoch 20/20
150000/150000 [==============================] - 7s 47us/sample - loss: 0.1028 - mean_squared_error: 0.1028 - mean_
absolute_error: 0.1487 - mean_absolute_percentage_error: 12.1002
```

In [18]:

```
predicted_values = []
for i in range(validation_set.shape[0]):
    val = validation_set[i]
    val = val.reshape((1,60))
    pred = mdl.predict(val)
    str_pred = pred[0][0]
    predicted_values.append(str_pred)
```

In [19]:

```
print("r2_score",r2_score(validation_set_float, predicted_values))
print("mse", mean_squared_error(validation_set_float, predicted_values))

```

```
r2_score 0.9474110182830929
mse 0.10427882563618182
```

In [ ]:

In [ ]: