

# CS5542 Big Data Analytics and App

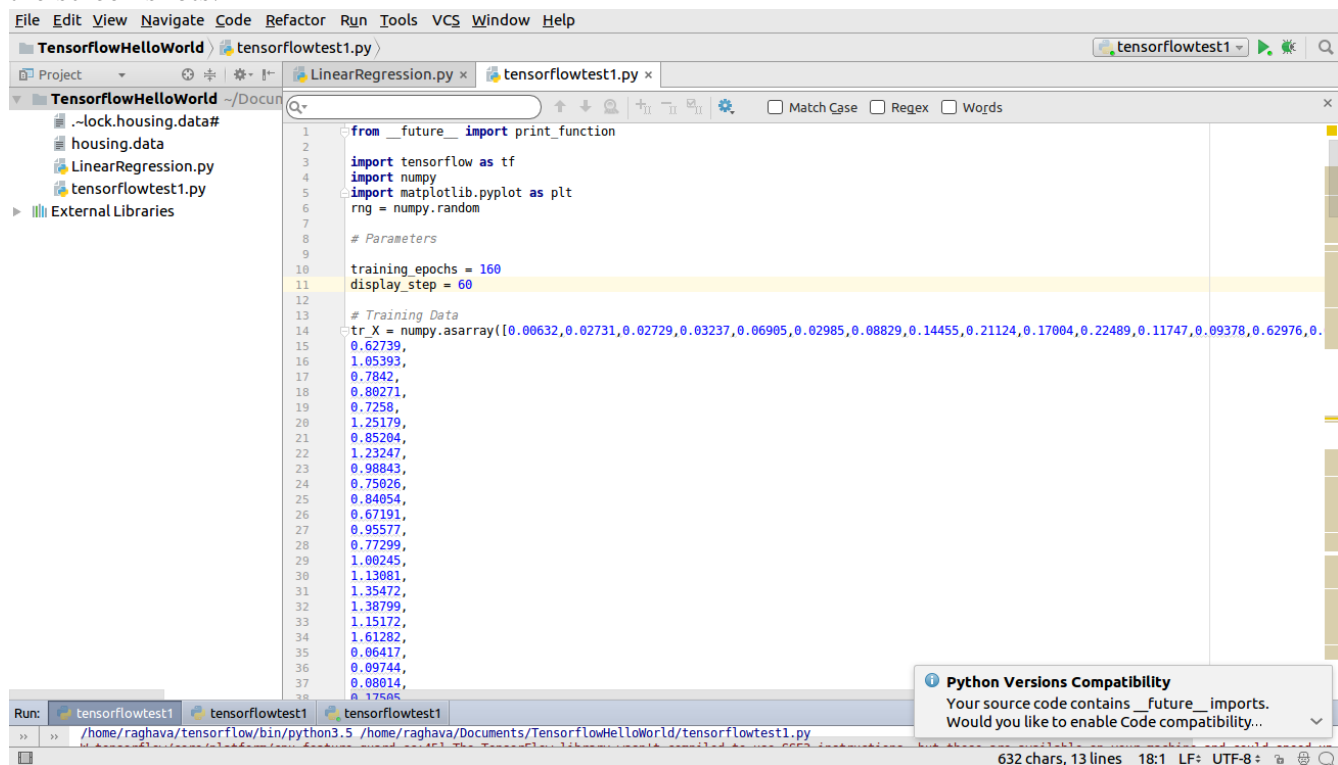
## LAB ASSIGNMENT #7

Linear Regression using Tensor flow:

[1]In statistics, **linear regression** is an approach for modeling the relationship between a scalar dependent variable  $y$  and one or more explanatory variables (or independent variables) denoted  $X$ . The case of one explanatory variable is called simple **linear regression**.

### Implementing Linear Regression through Tensor Flow:

**Housing data** set from the UCI machine Learning Repository is used. For this lab assignment only **50** rows from the data set are used for **training** and another **50** rows are used for the **test data**. Here are the screen shots:



```
File Edit View Navigate Code Refactor Run Tools VCS Window Help
TensorflowHelloWorld tensorflowtest1.py tensorflowtest1
Project LinearRegression.py tensorflowtest1.py
TensorflowHelloWorld
  .lock.housing.data#
  housing.data
  LinearRegression.py
  tensorflowtest1.py
  External Libraries
1 from __future__ import print_function
2
3 import tensorflow as tf
4 import numpy
5 import matplotlib.pyplot as plt
6 rng = numpy.random
7
8 # Parameters
9
10 training_epochs = 160
11 display_step = 60
12
13 # Training Data
14 tr_X = numpy.asarray([0.00632, 0.02731, 0.02729, 0.03237, 0.06985, 0.02985, 0.08829, 0.14455, 0.21124, 0.17004, 0.22489, 0.11747, 0.09378, 0.62976, 0.
15 0.62739,
16 1.05393,
17 0.7842,
18 0.80271,
19 0.7258,
20 1.25179,
21 0.85204,
22 1.23247,
23 0.98843,
24 0.75026,
25 0.84054,
26 0.67191,
27 0.95577,
28 0.77299,
29 1.00245,
30 1.13081,
31 1.35472,
32 1.38799,
33 1.15172,
34 1.61282,
35 0.86417,
36 0.89744,
37 0.80014,
38 0.17505,
```

Python Versions Compatibility  
Your source code contains `__future__` imports.  
Would you like to enable Code compatibility...

632 chars, 13 lines 18:1 LF: UTF-8

File Edit View Navigate Code Refactor Run Tools VCS Window Help

TensorflowHelloWorld tensorflowtest1.py

Project LinearRegression.py tensorflowtest1.py

TensorflowHelloWorld

- lock.housing.data#
- housing.data
- LinearRegression.py
- tensorflowtest1.py
- External Libraries

```
49 0.21977,  
50 1 )  
51 tr_Y = numpy.asarray([18,  
52 0,  
53 0,  
54 0,  
55 0,  
56 0,  
57 12.5,  
58 12.5,  
59 12.5,  
60 12.5,  
61 12.5,  
62 12.5,  
63 12.5,  
64 0,  
65 0,  
66 0,  
67 0,  
68 0,  
69 0,  
70 0,  
71 0,  
72 0,  
73 0,  
74 0,  
75 0,  
76 0,  
77 0,  
78 0,  
79 0,  
80 0,  
81 0,  
82 0,  
83 0,  
84 0,  
85 0,  
86 0
```

Run: tensorflowtest1 tensorflowtest1 tensorflowtest1

/home/raghava/tensorflow/bin/python3.5 /home/raghava/Documents/TensorflowHelloWorld/tensorflowtest1.py

632 chars, 13 lines 18:1 LF: UTF-8

**Python Versions Compatibility**  
Your source code contains `_future_` imports.  
Would you like to enable Code compatibility...

File Edit View Navigate Code Refactor Run Tools VCS Window Help

TensorflowHelloWorld tensorflowtest1.py

Project LinearRegression.py tensorflowtest1.py

TensorflowHelloWorld

- lock.housing.data#
- housing.data
- LinearRegression.py
- tensorflowtest1.py
- External Libraries

```
101 )  
102 n_samples = tr_X.shape[0]  
103  
104 # tf Graph Input  
105 X = tf.placeholder("float")  
106 Y = tf.placeholder("float")  
107  
108 # Set model weights  
109 W = tf.Variable(rng.randn(), name="weight")  
110 b = tf.Variable(rng.randn(), name="bias")  
111  
112 # Construct a linear model  
113 predict = tf.add(tf.multiply(X, W), b)  
114  
115 # Mean squared error  
116 cost = tf.reduce_sum(tf.pow(predict-Y, 2))/(2*n_samples)  
117 # Gradient descent  
118 train_optimizer = tf.train.GradientDescentOptimizer(0.5).minimize(cost)  
119  
120 # Initializing the variables  
121 init = tf.global_variables_initializer()  
122  
123 # Launch the graph  
124 with tf.Session() as sess:  
125     sess.run(init)  
126  
127     # Fit all training data  
128     for epoch in range(training_epochs):  
129         for (x, y) in zip(tr_X, tr_Y):  
130             sess.run(train_optimizer, feed_dict={X: x, Y: y})  
131  
132     # Display logs per epoch step  
133     if (epoch+1) % display_step == 0:  
134         c = sess.run(cost, feed_dict={X: tr_X, Y: tr_Y})  
135         print("Epoch:", '%04d' % (epoch+1), "cost=", "{:.9f}".format(c),  
136               "W=", sess.run(W), "b=", sess.run(b))  
137  
138     print("Optimization Completed!")
```

Run: tensorflowtest1 tensorflowtest1 tensorflowtest1

/home/raghava/tensorflow/bin/python3.5 /home/raghava/Documents/TensorflowHelloWorld/tensorflowtest1.py

121:41 LF: UTF-8

**Python Versions Compatibility**  
Your source code contains `_future_` imports.  
Would you like to enable Code compatibility...

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help
TensorflowHelloWorld tensorflowtest1.py
Project LinearRegression.py tensorflowtest1.py x
TensorflowHelloWorld ~/Documents
  .lock.housing.data#
  housing.data
  LinearRegression.py
  tensorflowtest1.py
  External Libraries
121 init = tf.global_variables_initializer()
122 for epoch in range(training_epochs):
123     for (x, y) in zip(tr_X, tr_Y):
124         sess.run(train_optimizer, feed_dict={X: x, Y: y})
125
126     # Display logs per epoch step
127     if (epoch+1) % display_step == 0:
128         c = sess.run(cost, feed_dict={X: tr_X, Y: tr_Y})
129         print("Epoch:", '%04d' % (epoch+1), "cost=", "{:.9f}".format(c),
130               "\n", sess.run(W), "b=", sess.run(b))
131
132 print("Optimization Completed!")
133 training_cost = sess.run(cost, feed_dict={X: tr_X, Y: tr_Y})
134 print("Train cost=", training_cost, "\n", sess.run(W), "b=", sess.run(b), '\n')
135
136 # Graphic display
137 plt.plot(tr_X, tr_Y, 'ro', label='Actual data')
138 plt.plot(tr_X, sess.run(W) * tr_X + sess.run(b), label='Fitted line')
139 plt.legend()
140 plt.show()
141
142 # Testing example, as requested (Issue #2)
143 test_X = numpy.asarray([0.08873,
144                        0.04337,
145                        0.0536,
146                        0.04981,
147                        0.0136,
148                        0.01311,
149                        0.02055,
150                        0.01432,
151                        0.15445,
152                        0.10328,
153                        0.14932,
154                        0.17171,
155                        0.11027,
156                        0.1265,
157                        0.01951,
158                        0.03584,
159                        0.04370])
160
161 Run: tensorflowtest1 tensorflowtest1 tensorflowtest1
162 /home/raghava/tensorflow/bin/python3.5 /home/raghava/Documents/TensorflowHelloWorld/tensorflowtest1.py
163 121:41 LF: UTF-8
```

**Python Versions Compatibility**  
Your source code contains `__future__` imports.  
Would you like to enable Code compatibility...

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help
TensorflowHelloWorld tensorflowtest1.py
Project LinearRegression.py tensorflowtest1.py x
TensorflowHelloWorld ~/Documents
  .lock.housing.data#
  housing.data
  LinearRegression.py
  tensorflowtest1.py
  External Libraries
121 init = tf.global_variables_initializer()
191 0.04203,
192 0.02875,
193 0.04294,
194 0.12204,
195 0.11504,
196 0.12083,
197 0.08187,
198 0.06861)
199
200 test_Y = numpy.asarray([21,
201                        21,
202                        21,
203                        75,
204                        90,
205                        85,
206                        100,
207                        25,
208                        25,
209                        25,
210                        25,
211                        25,
212                        25,
213                        17.5,
214                        80,
215                        80,
216                        12.5,
217                        12.5,
218                        12.5,
219                        0,
220                        0,
221                        0,
222                        0,
223                        0,
224                        0,
225                        0,
226                        0,
227                        0])
228
229 Run: tensorflowtest1 tensorflowtest1 tensorflowtest1
230 /home/raghava/tensorflow/bin/python3.5 /home/raghava/Documents/TensorflowHelloWorld/tensorflowtest1.py
231 121:41 LF: UTF-8
```

**Python Versions Compatibility**  
Your source code contains `__future__` imports.  
Would you like to enable Code compatibility...

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help
TensorflowHelloWorld tensorflowtest1.py
Project LinearRegression.py tensorflowtest1.py x
TensorflowHelloWorld ~/Documents
  .lock.housing.data#
  housing.data
  LinearRegression.py
  tensorflowtest1.py
  External Libraries
121 init = tf.global_variables_initializer()
233 25,
234 0,
235 0,
236 0,
237 0,
238 0,
239 0,
240 0,
241 0,
242 28,
243 28,
244 28,
245 0,
246 0,
247 0,
248 0,
249 0
250 0
251
252
253 print("Testing... (Mean square loss Comparison)")
254 testing_cost = sess.run(
255     tf.reduce_sum(tf.pow(predict - Y, 2)) / (2 * test_X.shape[0]),
256     feed_dict={X: test_X, Y: test_Y}) # same function as cost above
257 print("Test cost=", testing_cost)
258 print("Absolute mean square loss difference:", abs(
259     training_cost - testing_cost))
260
261 plt.plot(test_X, test_Y, 'bo', label='Test data')
262 plt.plot(tr_X, sess.run(W) * tr_X + sess.run(b), label='Fitted Line')
263 plt.legend()
264 plt.show()

Run: tensorflowtest1 tensorflowtest1 tensorflowtest1
/home/raghava/tensorflow/bin/python3.5 /home/raghava/Documents/TensorflowHelloWorld/tensorflowtest1.py
121:41 LF: UTF-8
```

**Python Versions Compatibility**  
Your source code contains `_future_` imports.  
Would you like to enable Code compatibility...

Here are the results obtained from running the logistic regression program on the data set:

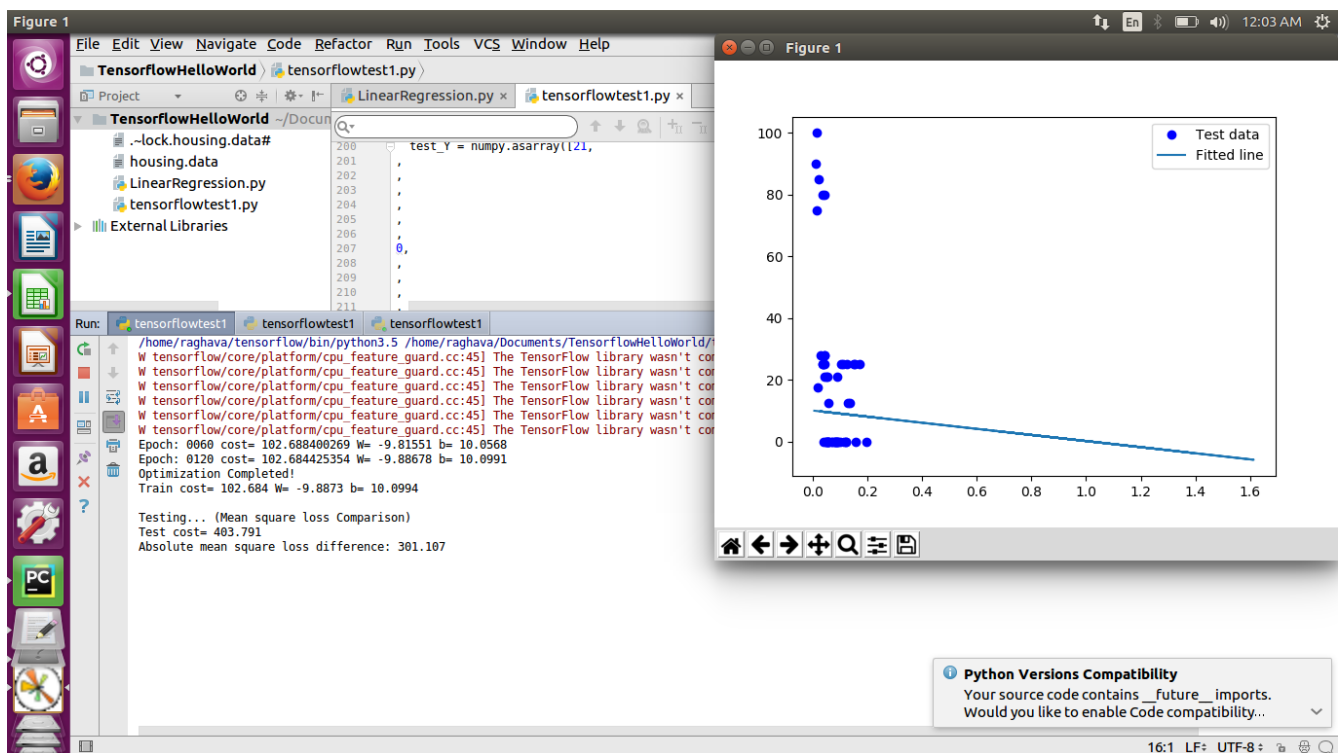
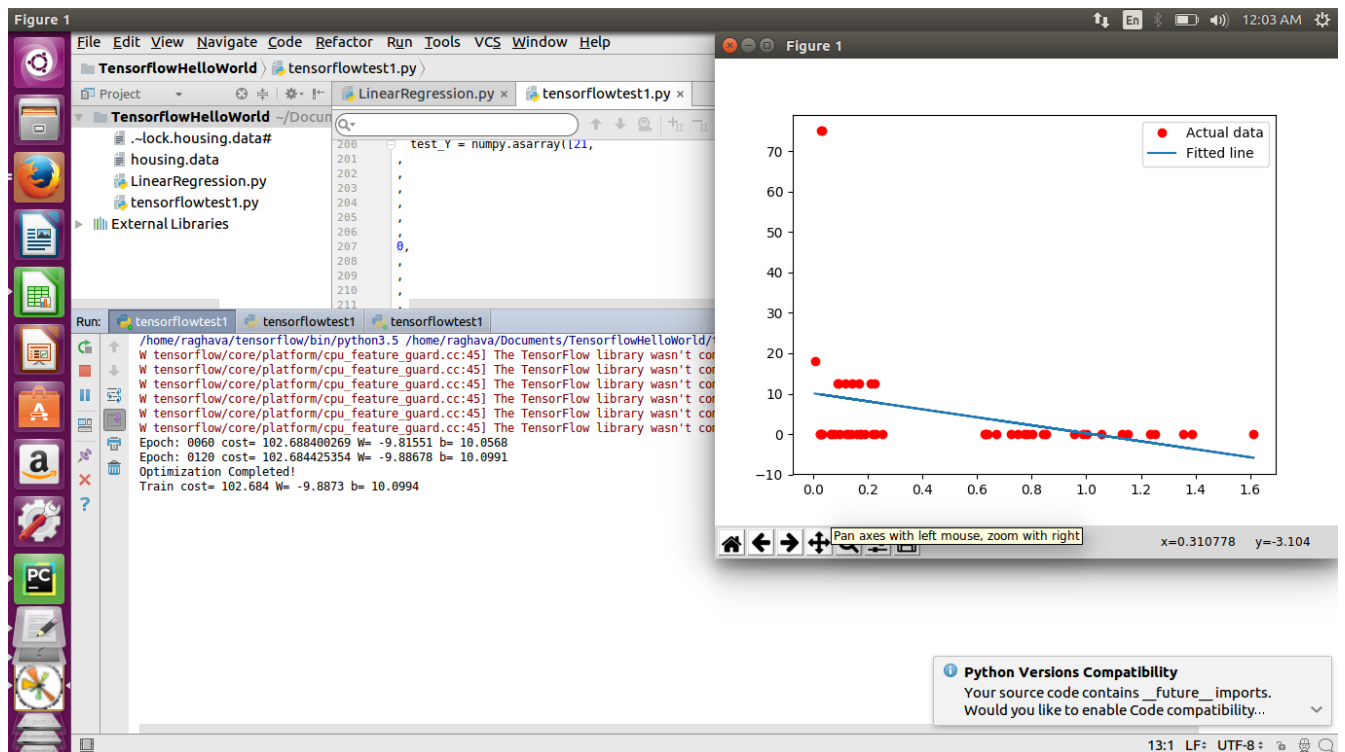
```
TensorflowHelloWorld - [~/Documents/TensorflowHelloWorld] - .../tensorflowtest1.py - PyCharm Community Edition 2016.3.2
File Edit View Navigate Code Refactor Run Tools VCS Window Help
TensorflowHelloWorld tensorflowtest1.py
Project LinearRegression.py tensorflowtest1.py x
TensorflowHelloWorld ~/Documents
  .lock.housing.data#
243
Run: tensorflowtest1 tensorflowtest1 tensorflowtest1
/home/raghava/tensorflow/bin/python3.5 /home/raghava/Documents/TensorflowHelloWorld/tensorflowtest1.py
W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use SSE3 instructions, but these are available on your machine and could speed up
W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use SSE4.1 instructions, but these are available on your machine and could speed u
W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use SSE4.2 instructions, but these are available on your machine and could speed u
W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use AVX instructions, but these are available on your machine and could speed up C
W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use AVX2 instructions, but these are available on your machine and could speed up C
W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use FMA instructions, but these are available on your machine and could speed up C
Epoch: 0060 cost= 102.689392090 W= -9.80828 b= 10.0477
Epoch: 0120 cost= 102.684432983 W= -9.88668 b= 10.099
Optimization Completed!
Train cost= 102.684 W= -9.88729 b= 10.0994

Testing... (Mean square loss Comparison)
Test cost= 403.791
Absolute mean square loss difference: 301.107

Process finished with exit code 0

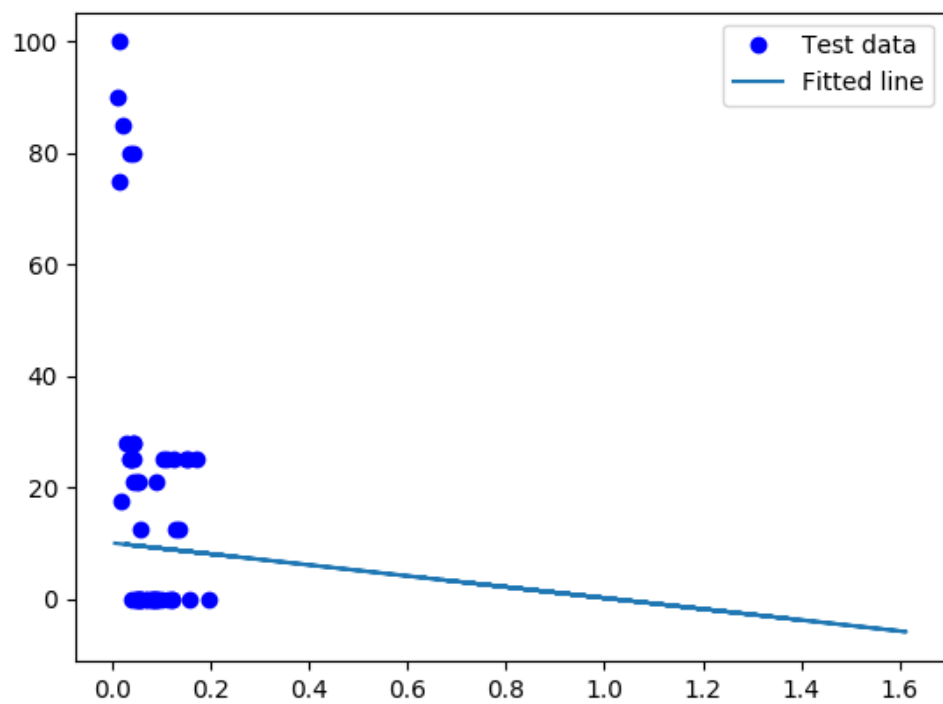
18:1 LF: UTF-8
```

**Python Versions Compatibility**  
Your source code contains `_future_` imports.  
Would you like to enable Code compatibility...



The training cost came out as **102.684**  
The test cost came out as **403.791**  
The Absolute mean square loss difference: **301.107**

Tes  
t:



References:

[1] [https://en.wikipedia.org/wiki/Logistic\\_regression](https://en.wikipedia.org/wiki/Logistic_regression)

for an idea of how to plot the data, little reference is taken from the internet:

[https://github.com/aymericdamien/TensorFlow-](https://github.com/aymericdamien/TensorFlow-Examples/blob/master/examples/2_BasicModels/linear_regression.py)

[Examples/blob/master/examples/2\\_BasicModels/linear\\_regression.py](https://github.com/aymericdamien/TensorFlow-Examples/blob/master/examples/2_BasicModels/linear_regression.py)

<https://blog.altoros.com/visualizing-tensorflow-graphs-with-tensorboard.html>