

Simple Linear Model

The example linear fit model uses an equation of

$$y = 2x - 1$$

So the data looks like:

x	y
-1	-3
0	-1
1	1
2	3
3	5
4	7

The code: <https://github.com/Univ-Wyo-Education/F21-1010/blob/main/lab/lab-11/linear1.py>

```
1: import tensorflow as tf
2: import numpy as np
3: from tensorflow import keras
4: model = tf.keras.Sequential([keras.layers.Dense(units=1, input_shape=[1])])
5: model.compile(optimizer='sgd', loss='mean_squared_error')
6:
7: # y = 2x - 1
8: xs = np.array([-1.0, 0.0, 1.0, 2.0, 3.0, 4.0], dtype=float)
9: ys = np.array([-3.0, -1.0, 1.0, 3.0, 5.0, 7.0], dtype=float)
10:
11: model.fit(xs, ys, epochs=500)
12:
13: print("expect output of approx. 19.0")
14: print(model.predict([10.0]))
```

Create a new linear fit set of data with the equation

$$y = 4x - 6$$

Update the data and re-train the model to use the new set of data. Then do a prediction with it.