Celsius2Fahrenheit

August 31, 2021

1 Predicting the conversion algorithm from ^oF to ^oC using a simple neural network

1.1 Importing required modules

[1]:
```
import matplotlib.pyplot as plt
import numpy as np
import tensorflow as tf
```

[2]:
```
from jupyterthemes import jtplot
jtplot.style('onedork', figsize=(16, 9))
```

1.2 Generating the labelled input data

The Celsius value is the feature. The Fahrenheit value is the label.

[3]:
```
celsius_feature = np.array([-40, -10, 0, 8, 15, 22, 38], dtype=float)
fahrenheit_label = np.array([-40, 14, 32, 46, 59, 72, 100], dtype=float)
```

1.3 Creating layers for the neural network

[4]:
```
l0 = tf.keras.layers.Dense(units=1, input_shape=[1])
```

1.4 Generating the model

[5]:
```
model = tf.keras.Sequential([l0])

model.compile(loss='mean_squared_error', optimizer=tf.keras.optimizers.Adam(0.
 ↪1))
```

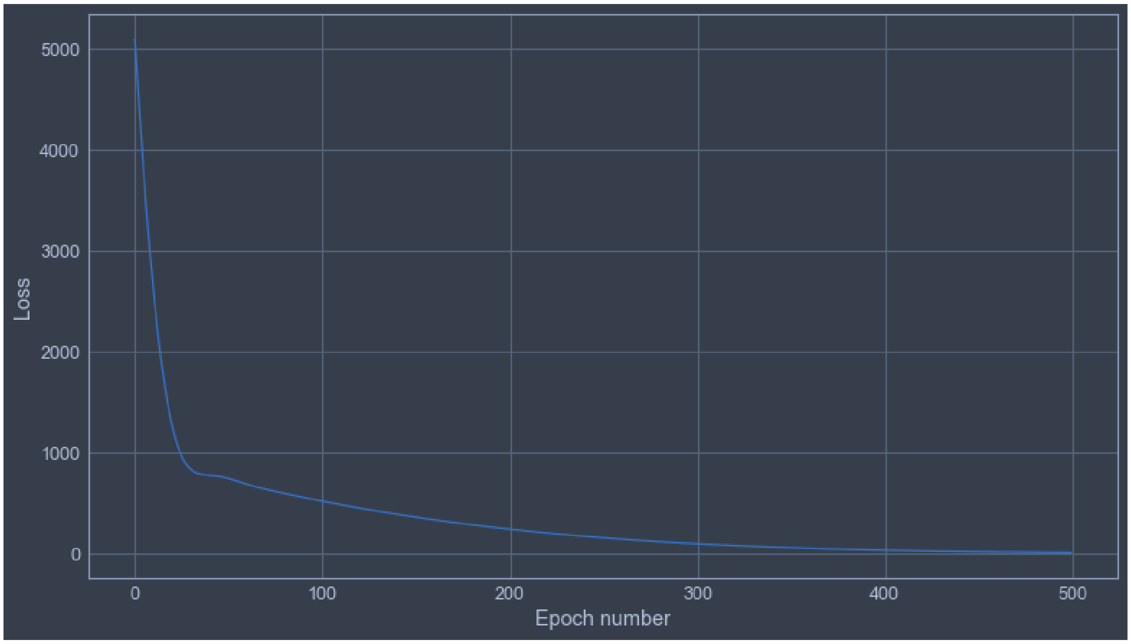
1.5 Training the model

[6]:
```
history = model.fit(celsius_feature, fahrenheit_label, epochs=500,␣
 ↪verbose=False)
print("Finished training the model")
```

```
Finished training the model
```

[7]:

```
plt.xlabel('Epoch number')
plt.ylabel('Loss')
plt.plot(history.history['loss'])
plt.show()
```



[8]:

```
print("These are the learnt weights: {}".format(l0.get_weights()))
```

```
These are the learnt weights: [array([[1.8254217]], dtype=float32),
array([28.759565], dtype=float32)]
```