```
#@title Licensed under the Apache License, Version 2.0 (the "License")
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
```

Licensed under the Apache License, Version 2.0 (the "License");

```
import tensorflow as tf
print(tf.__version__)
```

```
2.3.0
```

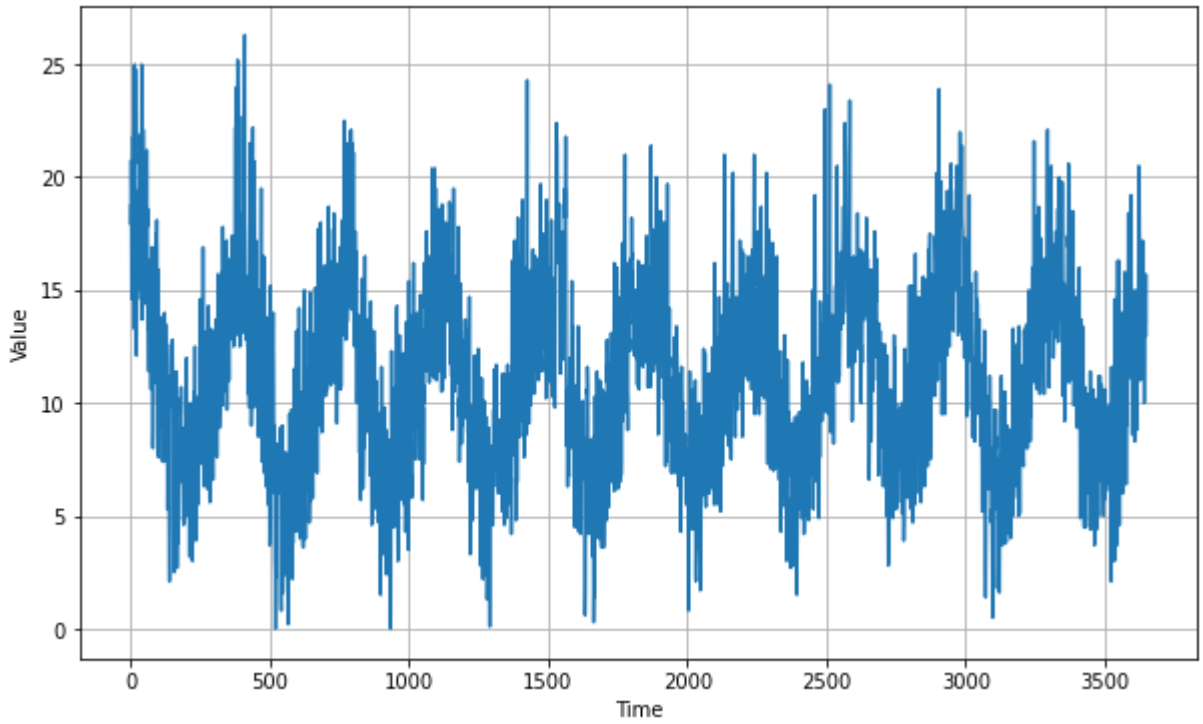
```
import numpy as np
import matplotlib.pyplot as plt
def plot_series(time, series, format="-", start=0, end=None):
    plt.plot(time[start:end], series[start:end], format)
    plt.xlabel("Time")
    plt.ylabel("Value")
    plt.grid(True)
```

```
!wget --no-check-certificate \
    https://raw.githubusercontent.com/jbrownlee/Datasets/master/daily-min-temperatures.csv \
    -O /tmp/daily-min-temperatures.csv
```

```
--2020-09-24 19:20:48--  https://raw.githubusercontent.com/jbrownlee/Datasets/master/dai
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.0.133, 151.10
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.0.133|:443.
HTTP request sent, awaiting response... 200 OK
Length: 67921 (66K) [text/plain]
Saving to: ‘/tmp/daily-min-temperatures.csv’

/tmp/daily-min-temp 100%[===================>]  66.33K  --.-KB/s    in 0.02s

2020-09-24 19:20:49 (4.05 MB/s) - ‘/tmp/daily-min-temperatures.csv’ saved [67921/67921]
```

```
import csv
time_step = []
temps = []

with open('/tmp/daily-min-temperatures.csv') as csvfile:
# YOUR CODE HERE. READ TEMPERATURES INTO TEMPS
  reader = csv.reader(csvfile, delimiter=',')
  next(reader)
```

```
  steps = 0
  for row in reader:
    temps.append(float(row[1]))
    time_step.append(steps)
    steps+=1
# HAVE TIME STEPS BE A SIMPLE ARRAY OF 1, 2, 3, 4 etc

#everything machine learning requires it to be in numpy arrays
series = np.array(temps)
time = np.array(time_step)
plt.figure(figsize=(10, 6))
plot_series(time, series)
```




```
split_time = 2500
time_train = time[:split_time]
x_train = series[:split_time]
time_valid = time[split_time:]
x_valid = series[split_time:]

window_size = 30
batch_size = 32
shuffle_buffer_size = 1000



def windowed_dataset(series, window_size, batch_size, shuffle_buffer):
    series = tf.expand_dims(series, axis=-1)
    dataset = tf.data.Dataset.from_tensor_slices(series)
    dataset = dataset.window(window_size + 1, shift=1, drop_remainder=True)
    dataset = dataset.flat_map(lambda window: window.batch(window_size + 1))
    dataset = dataset.shuffle(shuffle_buffer).map(lambda window: (window[:-1], window[1:]))
```

```
    dataset = dataset.shuffle(shuffle_buffer).map(lambda window: (window[:-1], window[1:]))
    dataset = dataset.batch(batch_size).prefetch(1)
    return dataset


def model_forecast(model, series, window_size):
    ds = tf.data.Dataset.from_tensor_slices(series)
    ds = ds.window(window_size, shift=1, drop_remainder=True)
    ds = ds.flat_map(lambda w: w.batch(window_size))
    ds = ds.batch(32).prefetch(1)
    forecast = model.predict(ds)
    return forecast


tf.keras.backend.clear_session()
tf.random.set_seed(51)
np.random.seed(51)
window_size = 64
batch_size = 256
train_set = windowed_dataset(x_train, window_size, batch_size, shuffle_buffer_size)
print(train_set)
print(x_train.shape)

model = tf.keras.models.Sequential([
  tf.keras.layers.Conv1D(filters=32, kernel_size=5, strides=1, padding="causal", activation="
  tf.keras.layers.LSTM(64, return_sequences=True),
  tf.keras.layers.LSTM(64, return_sequences=True),
  tf.keras.layers.Dense(30, activation="relu"),
  tf.keras.layers.Dense(10, activation="relu"),
  tf.keras.layers.Dense(1),
  tf.keras.layers.Lambda(lambda x: x * 400)
])

lr_schedule = tf.keras.callbacks.LearningRateScheduler(
    lambda epoch: 1e-8 * 10**(epoch / 20))
optimizer = tf.keras.optimizers.SGD(lr=1e-8, momentum=0.9)
model.compile(loss=tf.keras.losses.Huber(),
              optimizer=optimizer,
              metrics=["mae"])
history = model.fit(train_set, epochs=100, callbacks=[lr_schedule])
```

```
<PrefetchDataset shapes: ((None, None, 1), (None, None, 1)), types: (tf.float64, tf.floa
(2500,)
Epoch 1/100
10/10 [==============================] - 0s 24ms/step - loss: 31.1571 - mae: 31.6550
Epoch 2/100
10/10 [==============================] - 0s 22ms/step - loss: 30.5778 - mae: 31.0756
Epoch 3/100
10/10 [==============================] - 0s 24ms/step - loss: 29.6825 - mae: 30.1801
Epoch 4/100
10/10 [==============================] - 0s 21ms/step - loss: 28.5613 - mae: 29.0586
Epoch 5/100
10/10 [==============================] - 0s 24ms/step - loss: 27.1974 - mae: 27.6945
Epoch 6/100
10/10 [==============================] - 0s 29ms/step - loss: 25.5017 - mae: 25.9986
Epoch 7/100
10/10 [==============================] - 0s 30ms/step - loss: 23.3464 - mae: 23.8429
Epoch 8/100
10/10 [==============================] - 0s 24ms/step - loss: 20.6148 - mae: 21.1108
Epoch 9/100
10/10 [==============================] - 0s 23ms/step - loss: 17.3142 - mae: 17.8091
Epoch 10/100
10/10 [==============================] - 0s 25ms/step - loss: 13.6449 - mae: 14.1371
Epoch 11/100
10/10 [==============================] - 0s 24ms/step - loss: 10.1273 - mae: 10.6152
Epoch 12/100
10/10 [==============================] - 0s 24ms/step - loss: 7.6175 - mae: 8.1025
Epoch 13/100
10/10 [==============================] - 0s 33ms/step - loss: 6.2869 - mae: 6.7711
Epoch 14/100
10/10 [==============================] - 0s 28ms/step - loss: 5.7015 - mae: 6.1856
Epoch 15/100
10/10 [==============================] - 0s 23ms/step - loss: 5.3344 - mae: 5.8166
Epoch 16/100
10/10 [==============================] - 0s 23ms/step - loss: 4.9409 - mae: 5.4206
Epoch 17/100
10/10 [==============================] - 0s 24ms/step - loss: 4.5578 - mae: 5.0338
Epoch 18/100
10/10 [==============================] - 0s 33ms/step - loss: 4.2340 - mae: 4.7085
Epoch 19/100
10/10 [==============================] - 0s 26ms/step - loss: 3.9611 - mae: 4.4360
Epoch 20/100
10/10 [==============================] - 0s 24ms/step - loss: 3.7448 - mae: 4.2177
Epoch 21/100
10/10 [==============================] - 0s 26ms/step - loss: 3.5855 - mae: 4.0566
Epoch 22/100
10/10 [==============================] - 0s 23ms/step - loss: 3.4641 - mae: 3.9344
Epoch 23/100
10/10 [==============================] - 0s 23ms/step - loss: 3.3711 - mae: 3.8414
Epoch 24/100
10/10 [==============================] - 0s 23ms/step - loss: 3.2948 - mae: 3.7645
Epoch 25/100
10/10 [==============================] - 0s 24ms/step - loss: 3.2288 - mae: 3.6978
Epoch 26/100
10/10 [==============================] - 0s 23ms/step - loss: 3.1662 - mae: 3.6346
Epoch 27/100
10/10 [==============================] - 0s 24ms/step - loss: 3.1014 - mae: 3.5693
Epoch 28/100
```

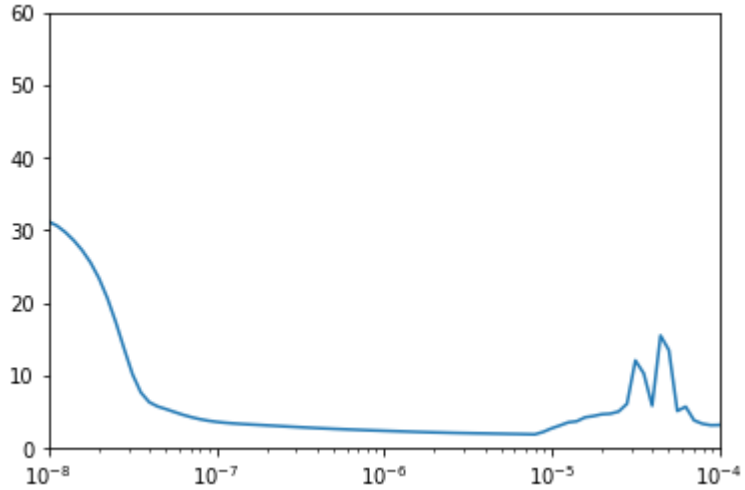
```
10/10 [==============================] - 0s 22ms/step - loss: 3.0380 - mae: 3.5053
Epoch 29/100
10/10 [==============================] - 0s 22ms/step - loss: 2.9716 - mae: 3.4379
Epoch 30/100
10/10 [==============================] - 0s 23ms/step - loss: 2.9063 - mae: 3.3712
Epoch 31/100
10/10 [==============================] - 0s 23ms/step - loss: 2.8452 - mae: 3.3098
Epoch 32/100
10/10 [==============================] - 0s 23ms/step - loss: 2.7842 - mae: 3.2479
Epoch 33/100
10/10 [==============================] - 0s 26ms/step - loss: 2.7256 - mae: 3.1891
Epoch 34/100
10/10 [==============================] - 0s 29ms/step - loss: 2.6740 - mae: 3.1379
Epoch 35/100
10/10 [==============================] - 0s 24ms/step - loss: 2.6212 - mae: 3.0848
Epoch 36/100
10/10 [==============================] - 0s 24ms/step - loss: 2.5728 - mae: 3.0357
Epoch 37/100
10/10 [==============================] - 0s 24ms/step - loss: 2.5254 - mae: 2.9877
Epoch 38/100
10/10 [==============================] - 0s 25ms/step - loss: 2.4800 - mae: 2.9415
Epoch 39/100
10/10 [==============================] - 0s 23ms/step - loss: 2.4355 - mae: 2.8968
Epoch 40/100
10/10 [==============================] - 0s 24ms/step - loss: 2.3933 - mae: 2.8541
Epoch 41/100
10/10 [==============================] - 0s 24ms/step - loss: 2.3523 - mae: 2.8124
Epoch 42/100
10/10 [==============================] - 0s 23ms/step - loss: 2.3124 - mae: 2.7718
Epoch 43/100
10/10 [==============================] - 0s 26ms/step - loss: 2.2730 - mae: 2.7315
Epoch 44/100
10/10 [==============================] - 0s 29ms/step - loss: 2.2338 - mae: 2.6913
Epoch 45/100
10/10 [==============================] - 0s 23ms/step - loss: 2.1954 - mae: 2.6523
Epoch 46/100
10/10 [==============================] - 0s 25ms/step - loss: 2.1622 - mae: 2.6190
Epoch 47/100
10/10 [==============================] - 0s 25ms/step - loss: 2.1313 - mae: 2.5877
Epoch 48/100
10/10 [==============================] - 0s 25ms/step - loss: 2.1047 - mae: 2.5607
Epoch 49/100
10/10 [==============================] - 0s 30ms/step - loss: 2.0740 - mae: 2.5300
Epoch 50/100
10/10 [==============================] - 0s 23ms/step - loss: 2.0481 - mae: 2.5040
Epoch 51/100
10/10 [==============================] - 0s 25ms/step - loss: 2.0263 - mae: 2.4818
Epoch 52/100
10/10 [==============================] - 0s 23ms/step - loss: 2.0048 - mae: 2.4598
Epoch 53/100
10/10 [==============================] - 0s 26ms/step - loss: 1.9897 - mae: 2.4444
Epoch 54/100
10/10 [==============================] - 0s 23ms/step - loss: 1.9650 - mae: 2.4193
Epoch 55/100
10/10 [==============================] - 0s 22ms/step - loss: 1.9469 - mae: 2.4008
Epoch 56/100
10/10 [==============================] - 0s 24ms/step - loss: 1.9247 - mae: 2.3783
Epoch 57/100
```

```
10/10 [==============================] - 0s 23ms/step - loss: 1.9089 - mae: 2.3620
Epoch 58/100
10/10 [==============================] - 0s 23ms/step - loss: 1.8862 - mae: 2.3393
Epoch 59/100
10/10 [==============================] - 0s 24ms/step - loss: 1.8593 - mae: 2.3119
Epoch 60/100
10/10 [==============================] - 0s 23ms/step - loss: 2.1952 - mae: 2.6551
Epoch 61/100
10/10 [==============================] - 0s 26ms/step - loss: 2.6854 - mae: 3.1563
Epoch 62/100
10/10 [==============================] - 0s 25ms/step - loss: 3.0855 - mae: 3.5613
Epoch 63/100
10/10 [==============================] - 0s 33ms/step - loss: 3.5162 - mae: 3.9954
Epoch 64/100
10/10 [==============================] - 0s 25ms/step - loss: 3.6573 - mae: 4.1383
Epoch 65/100
10/10 [==============================] - 0s 24ms/step - loss: 4.2071 - mae: 4.6937
Epoch 66/100
10/10 [==============================] - 0s 24ms/step - loss: 4.3837 - mae: 4.8701
Epoch 67/100
10/10 [==============================] - 0s 28ms/step - loss: 4.6317 - mae: 5.1194
Epoch 68/100
10/10 [==============================] - 0s 31ms/step - loss: 4.7024 - mae: 5.1886
Epoch 69/100
10/10 [==============================] - 0s 25ms/step - loss: 5.0037 - mae: 5.4914
Epoch 70/100
10/10 [==============================] - 0s 24ms/step - loss: 6.0697 - mae: 6.5529
Epoch 71/100
10/10 [==============================] - 0s 23ms/step - loss: 12.0408 - mae: 12.5377
Epoch 72/100
10/10 [==============================] - 0s 24ms/step - loss: 10.2502 - mae: 10.7457
Epoch 73/100
10/10 [==============================] - 0s 26ms/step - loss: 5.8055 - mae: 6.2862
Epoch 74/100
10/10 [==============================] - 0s 27ms/step - loss: 15.4747 - mae: 15.9666
Epoch 75/100
10/10 [==============================] - 0s 34ms/step - loss: 13.4579 - mae: 13.9447
Epoch 76/100
10/10 [==============================] - 0s 25ms/step - loss: 5.0764 - mae: 5.5564
Epoch 77/100
10/10 [==============================] - 0s 31ms/step - loss: 5.6755 - mae: 6.1618
Epoch 78/100
10/10 [==============================] - 0s 27ms/step - loss: 3.7869 - mae: 4.2641
Epoch 79/100
10/10 [==============================] - 0s 33ms/step - loss: 3.3208 - mae: 3.7941
Epoch 80/100
10/10 [==============================] - 0s 25ms/step - loss: 3.1221 - mae: 3.5942
Epoch 81/100
10/10 [==============================] - 0s 23ms/step - loss: 3.1429 - mae: 3.6126
Epoch 82/100
10/10 [==============================] - 0s 24ms/step - loss: 3.7473 - mae: 4.2232
Epoch 83/100
10/10 [==============================] - 0s 27ms/step - loss: 4.0417 - mae: 4.5208
Epoch 84/100
10/10 [==============================] - 0s 23ms/step - loss: 5.3185 - mae: 5.8036
Epoch 85/100
10/10 [==============================] - 0s 24ms/step - loss: 4.9910 - mae: 5.4747
Epoch 86/100
```

```
Epoch 86/100
10/10 [==============================] - 0s 23ms/step - loss: 5.5814 - mae: 6.0642
Epoch 87/100
10/10 [==============================] - 0s 26ms/step - loss: 7.3240 - mae: 7.8139
Epoch 88/100
10/10 [==============================] - 0s 29ms/step - loss: 7.9318 - mae: 8.4220
Epoch 89/100
10/10 [==============================] - 0s 33ms/step - loss: 41.4157 - mae: 41.9151
Epoch 90/100
10/10 [==============================] - 0s 28ms/step - loss: 72.6887 - mae: 73.1870
Epoch 91/100
10/10 [==============================] - 0s 32ms/step - loss: 72.8214 - mae: 73.3214
Epoch 92/100
10/10 [==============================] - 0s 31ms/step - loss: 90.7138 - mae: 91.2138
Epoch 93/100
10/10 [==============================] - 0s 33ms/step - loss: 102.8926 - mae: 103.3911
Epoch 94/100
10/10 [==============================] - 0s 24ms/step - loss: 64.7180 - mae: 65.2159
Epoch 95/100
```

```
plt.semilogx(history.history["lr"], history.history["loss"])
plt.axis([1e-8, 1e-4, 0, 60])
```

```
(1e-08, 0.0001, 0.0, 60.0)
```




```
tf.keras.backend.clear_session()
tf.random.set_seed(51)
np.random.seed(51)
train_set = windowed_dataset(x_train, window_size=60, batch_size=100, shuffle_buffer=shuffle_
model = tf.keras.models.Sequential([
  tf.keras.layers.Conv1D(filters=32, kernel_size=5, strides=1, padding="causal", activation="
  tf.keras.layers.LSTM(64, return_sequences=True),
  tf.keras.layers.LSTM(64, return_sequences=True),
  tf.keras.layers.Dense(30, activation="relu"),
  tf.keras.layers.Dense(10, activation="relu"),
  tf.keras.layers.Dense(1),
  tf.keras.layers.Lambda(lambda x: x * 400)
])
```

```
optimizer = tf.keras.optimizers.SGD(lr=5e-5, momentum=0.9)
```