(https://colab.research.google.com/github/sergejhorvat/TensorFlow-Data-and-Deployment-Specialization/blob/master/Device-based%20Models%20with%20TensorFlow/Week%201/Examples/TFLite_Week1_Linear_Regression.ipynb)

***Copyright 2018 The TensorFlow Authors.***

In [0]:
```
#@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.
```

# Running TFLite models

| | |
|---|---|
|  Run in Google Colab (https://colab.research.google.com/github/lmoroney/dlaicourse/blob/master/TensorFlow%20Deployment/Course%202%20-%20TensorFlow%20Lite/Week%201/Examples/TFLite_Week1_Linear_Regression.ipynb) |  View source on GitHub (https://github.com/lmoroney/dlaicourse/blob/master/TensorFlow%20Deployment/Course%202%20-%20TensorFlow%20Lite/Week%201/Examples/TFLite_Week1_Linear_Regression.ipynb) |

## Setup

In [0]:
```
try:
    %tensorflow_version 2.x
except:
    pass
```

In [0]:
```
import pathlib
import numpy as np
import matplotlib.pyplot as plt

import tensorflow as tf

print('\u2022 Using TensorFlow Version:', tf.__version__)
```

## Create a Basic Model of the Form y = mx + c

In [0]:
```
# Create a simple Keras model.
x = [-1, 0, 1, 2, 3, 4]
y = [-3, -1, 1, 3, 5, 7]

model = tf.keras.models.Sequential([
        tf.keras.layers.Dense(units=1, input_shape=[1])
])

model.compile(optimizer='sgd',
              loss='mean_squared_error')

model.fit(x, y, epochs=200)
```

## Generate a SavedModel

In [0]:
```
export_dir = 'saved_model/1'
tf.saved_model.save(model, export_dir)
```

## Convert the SavedModel to TFLite

Saved model can be used on Androind, iOS and Linux embedded systems.
Check filesystem to see all files generated.

In [0]:
```
# Convert the model.
converter = tf.lite.TFLiteConverter.from_saved_model(export_dir)
tflite_model = converter.convert()
```

In [0]:
```
tflite_model_file = pathlib.Path('model.tflite')
tflite_model_file.write_bytes(tflite_model)
```

# Initialize the TFLite Interpreter To Try It Out

In [0]:
```
# Load TFLite model and allocate tensors.
interpreter = tf.lite.Interpreter(model_content=tflite_model)
interpreter.allocate_tensors()

# Get input and output tensors.
input_details = interpreter.get_input_details()
output_details = interpreter.get_output_details()
```

In [0]:
```
# Test the TensorFlow Lite model on random input data.
input_shape = input_details[0]['shape']
inputs, outputs = [], []
for _ in range(100):
    input_data = np.array(np.random.random_sample(input_shape), dtype=np.float32)
    interpreter.set_tensor(input_details[0]['index'], input_data)

    interpreter.invoke()
    tflite_results = interpreter.get_tensor(output_details[0]['index'])

    # Test the TensorFlow model on random input data.
    tf_results = model(tf.constant(input_data))
    output_data = np.array(tf_results)

    inputs.append(input_data[0][0])
    outputs.append(output_data[0][0])
```

## Visualize the Model

In [0]:
```
%matplotlib inline

plt.plot(inputs, outputs, 'r')
plt.show()
```

## Download the TFLite Model File

If you are running this notebook in a Colab, you can run the cell below to download the tflite model to your local disk.

**Note**: If the file does not download when you run the cell, try running the cell a second time.

In [0]:
```
try:
    from google.colab import files
    files.download(tflite_model_file)
except:
    pass
```

# Command-line usage

If Python code is not available for generating the model, but have acces to saved model file. Converter cli commands to convert to TF Lite model:

In [0]:
```
# Saving with the CLI from a SavedModel
tflite_converter --output_file=model.tflite --saved_model_dir=/tmp/saved_model_dir

# Saving with the command-line from a Keras model
tflite_convert --output_file=model.tflite --keras_model_file=model.h5
```