

*****Draft*****

TensorFlow for Raspberry Pi 4 64 bit and Raspberry Pi Pico 10/04/22

*****Draft*****

One of the first videos on Tensorflow that I watched was

video link <https://www.youtube.com/watch?v=tPYj3fFJGjk&t=339s>

TensorFlow 2.0 Complete Course - Python Neural Networks for Beginners Tutorial

About the Author

The author of this course is Tim Ruscica, otherwise known as “Tech With Tim” from his educational programming YouTube channel. Tim has a passion for teaching and loves to teach about the world of machine learning and artificial intelligence.

This has led to a journey of learning.

Spent many hours working on C/C++ using the <https://github.com/raspberrypi/pico-tflmicro>.

This repository provided and older version of Tensorflow Lite for the Raspberry Pi Pico.

Since then I have managed to update <https://github.com/develone/devel-pico-tflmicro> to early Sep 2022,

Installed Tensorflow 2.9 on a 64 bit Raspberry Pi 4 4Gb.
10/3/2022

video link <https://youtu.be/QLZWQlg-Pk0>

How to Install Tensorflow 2 on a Raspberry Pi

<https://www.samwestby.com/tutorials/rpi-tensorflow.html>

Sam Westby

This has led to several videos on Game On! - Flatbuffers

```
uname -m  
aarch64
```

```
python3 -V  
Python 3.9.2
```

```
cat /etc/os-release  
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"  
NAME="Debian GNU/Linux"  
VERSION_ID="11"
```

```
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
```

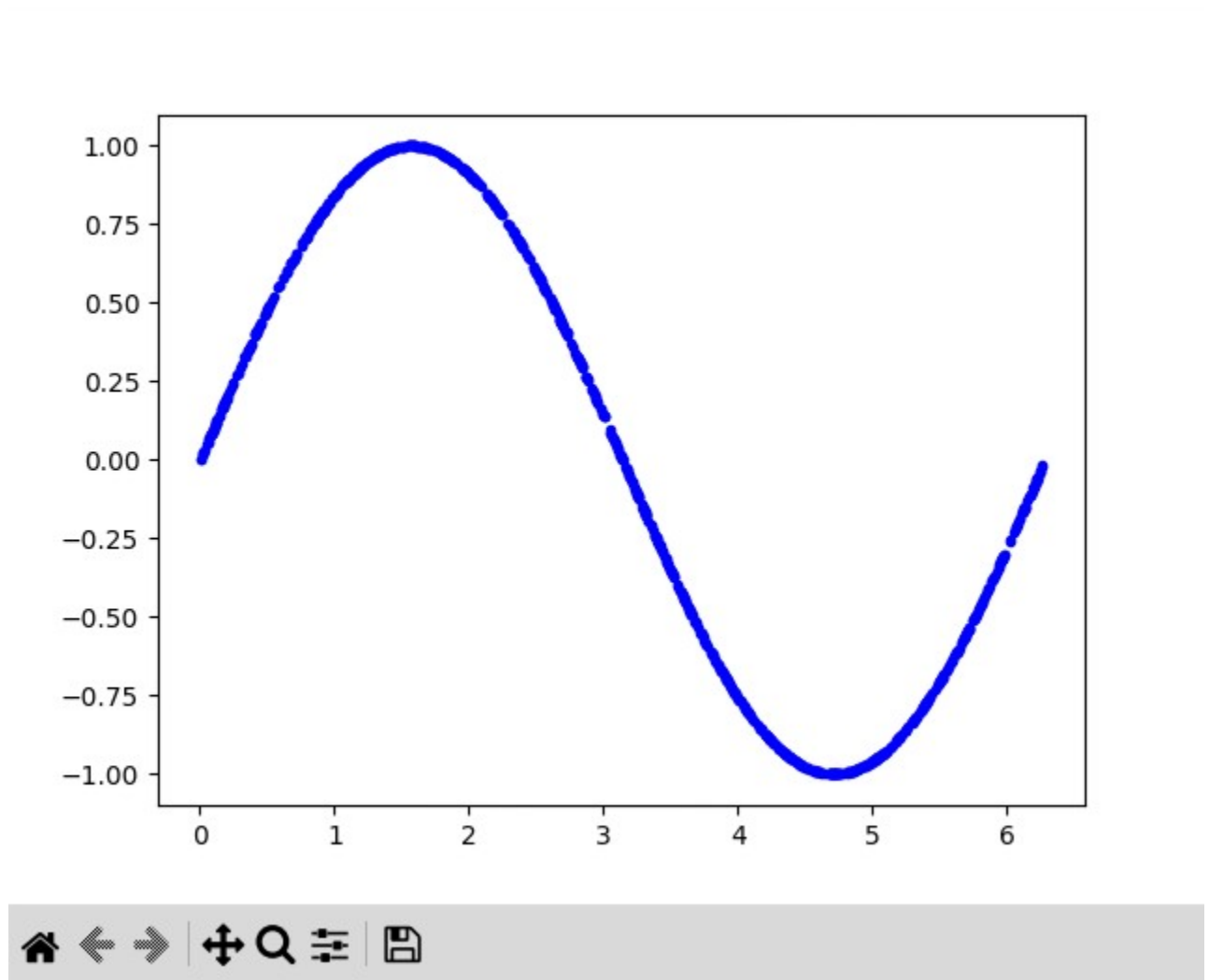
```
sudo apt-get install --yes libssl-dev zlib1g-dev libbz2-dev libreadline-dev libsqlite3-dev llvm
libncurses5-dev libncursesw5-dev xz-utils tk-dev libgdbm-dev lzma lzma-dev tcl-dev libxml2-dev
libxmlsec1-dev libffi-dev liblzma-dev wget curl make build-essential openssl
```

```
sudo apt-get install -y libhdf5-dev libc-ares-dev libeigen3-dev gcc gfortran libgfortran5 libatlas3-base
libatlas-base-dev libopenblas-dev libopenblas-base libblas-dev liblapack-dev cython3 libatlas-base-dev
openmpi-bin libopenmpi-dev python3-dev
```

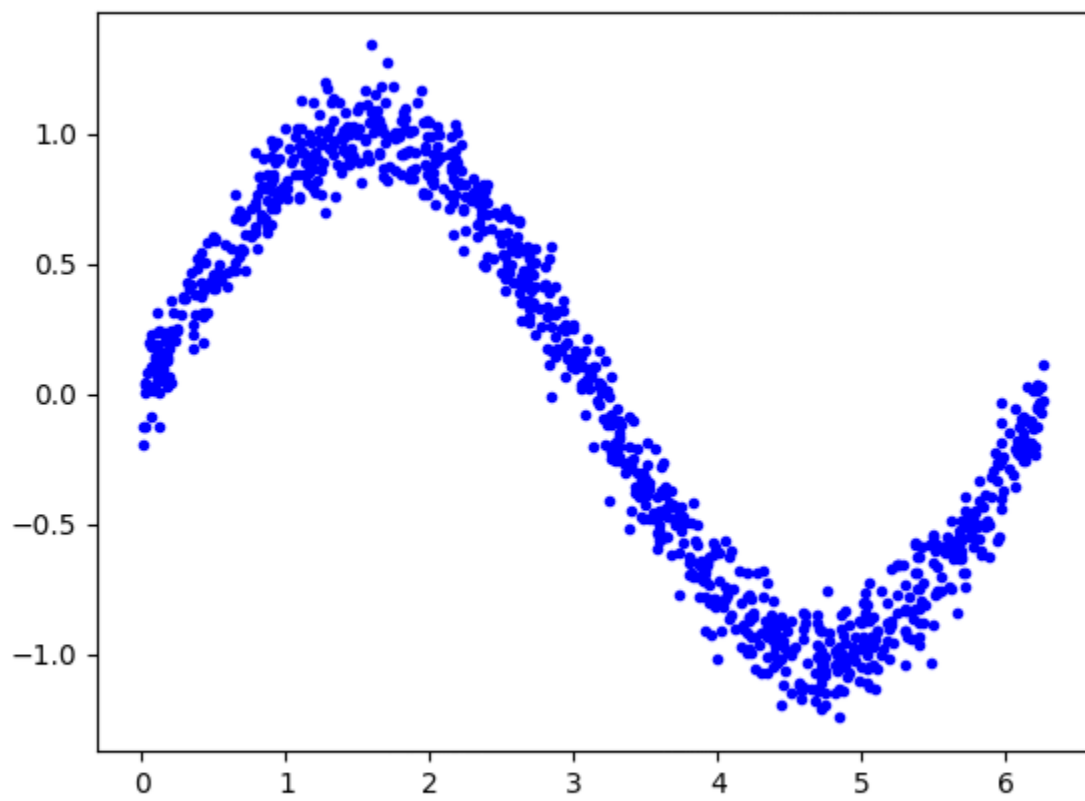
numpy was used to generate a 1000 point sin.

```
x_values = np.random.uniform(low=0, high=2*math.pi, size=SAMPLES).astype(np.float32)
```

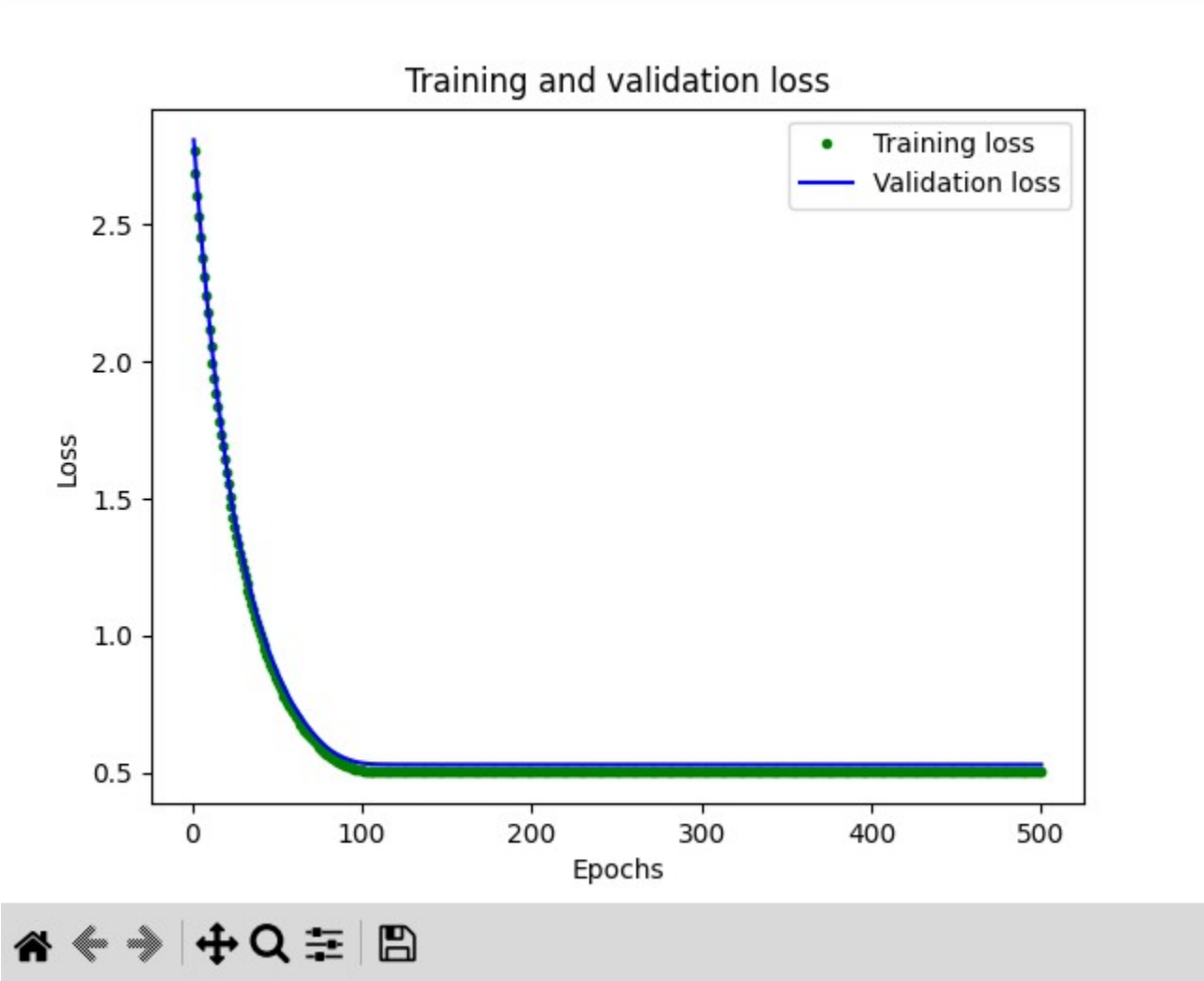
```
y_values = np.sin(x_values).astype(np.float32)
```



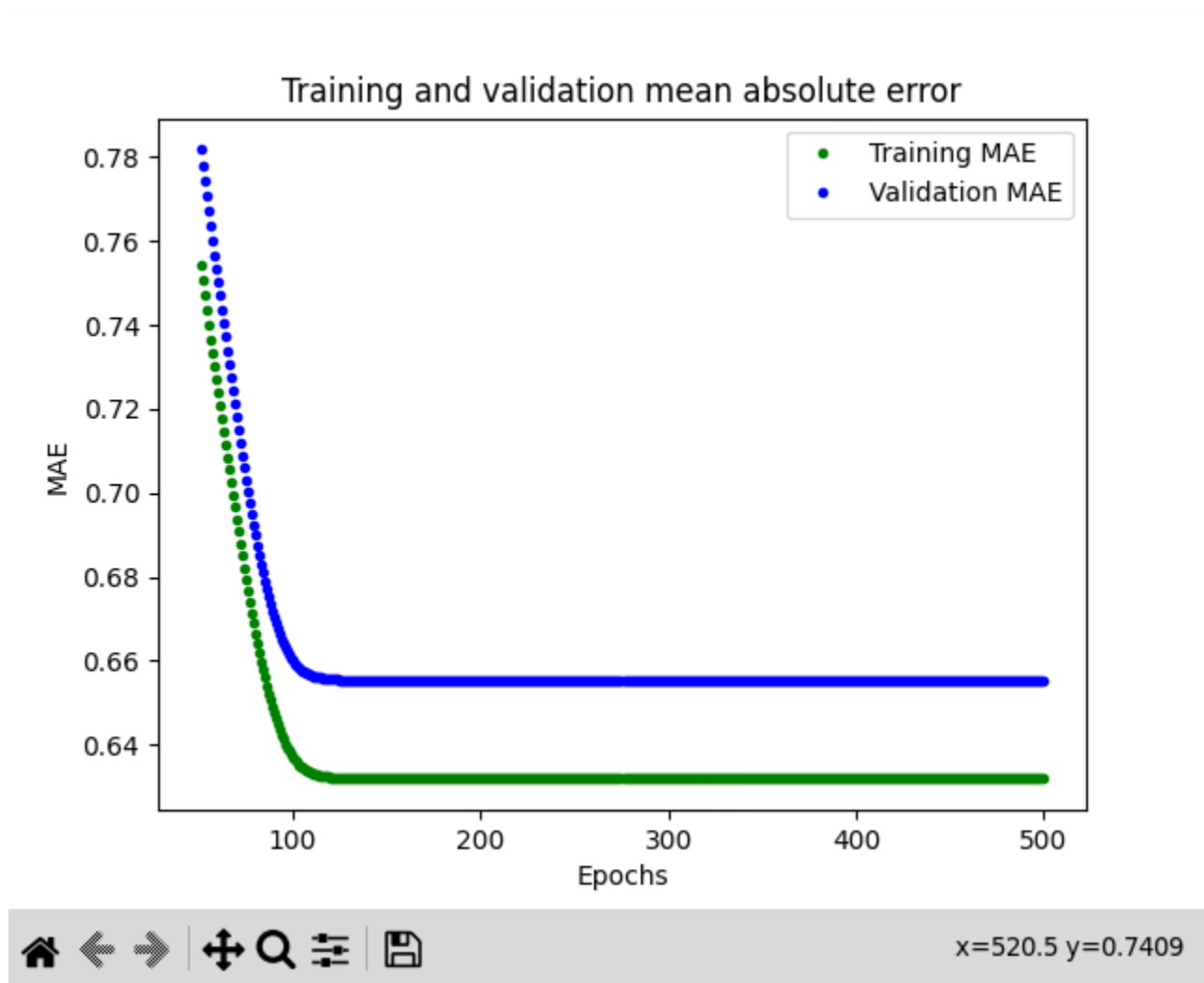
signal with noise.



Xx



xxx



xxx

Training and validation loss

