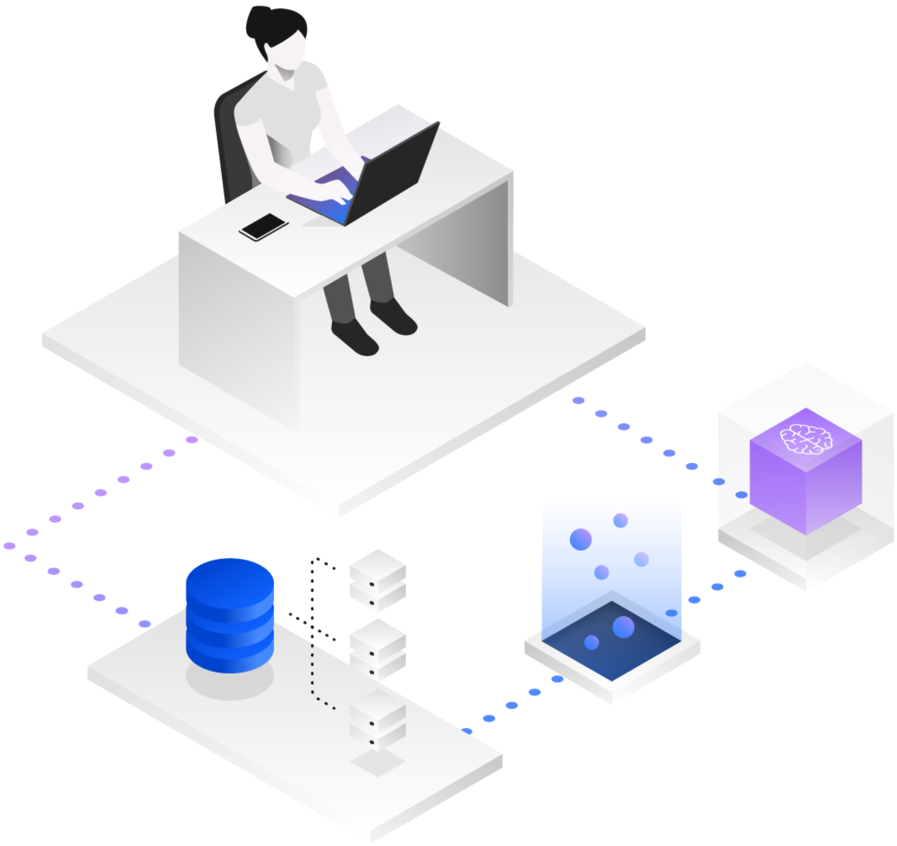
Digital Developer Conference

Data & AI

Early Forests Fire Detection via Machine Learning

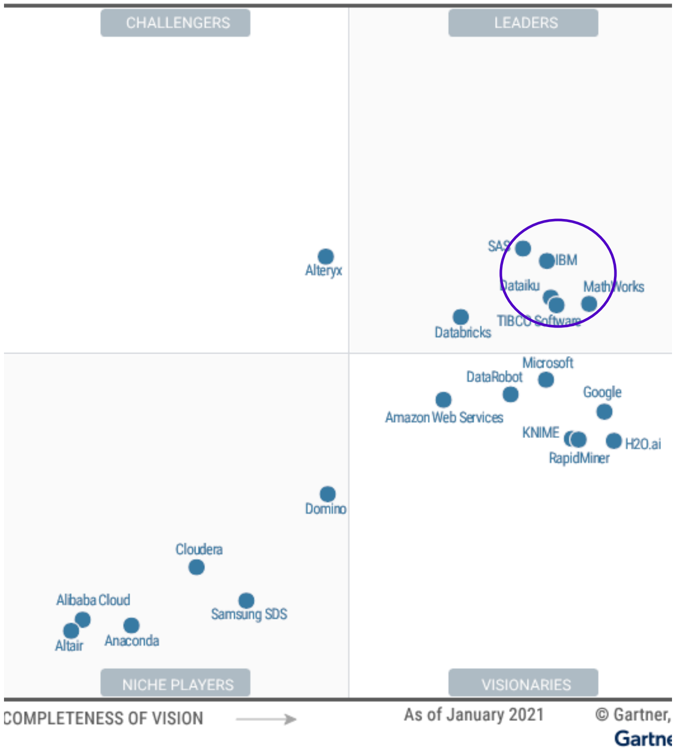
Speakers: Graciana Puentes,
University of Buenos Aires,
National Research Council,
CONICET-ARGENTINA
gracianapuentes@gmail.com



AI & Machine Learning:

Machine learning is a branch of Artificial Intelligence (AI) which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

Through the use of statistical methods, algorithms are trained to make classifications or predictions, uncovering key insights within data.



As of January 2021

© Gartner,

IBM named leader in 2021 GartnerMagic Quadrant
For Data Science and Machine Learning Platforms

Machine Learning vs Deep Learning vs NN

Deep learning and Machine Learning tend to be used interchangeably. Machine Learning, Deep Learning, and Neural Networks are all sub-fields of Artificial Intelligence. However, Deep Learning is actually a sub-field of Machine Learning, and Neural Networks is a sub-field of Deep Learning.

Deep Learning **automates** much of the feature extraction process, eliminating much of the manual human intervention required and enabling the use of larger data sets.

You can think of Deep Learning as "*scalable machine learning*" as Lex Fridman notes.

Lex Fridman

https://deeplearning.mit.edu/

Deep Learning in one slide

- **What is it?**

Extract useful patterns from data sets

- **How?**

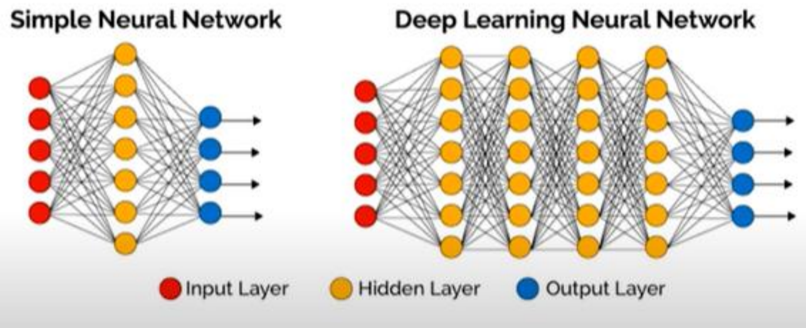
Using Deep Neural Networks and Optimization

- **How (practically)?**

Python + TensorFlow + TorchVision (and friends)

- **Hard Part**

Good questions + Good Data



Applications

- Pattern recognition
- Image Classification
- Text transcription
- Machine Translation
- Medical Diagnosis
- Digital Assistants
- Game playing
- Etc, etc

Deep Learning in one slide

- **What is it?**

Extract useful patterns from data sets

- **How?**

Using Deep Neural Networks and Optimization

- **How (practically)?**

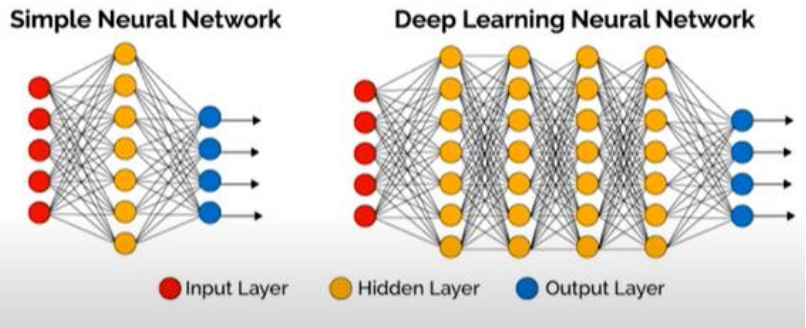
Python + TensorFlow + TorchVision (and friends)

- **Hard Part**

Good questions + Good Data

- **What is a Good Question?**

Can we use Deep Learning to solve the most important question of the 21st Century ….



Applications

- Pattern recognition
- Image Classification
- Text transcription
- Machine Translation
- Medical Diagnosis
- Digital Assistants
- Game playing
- Etc, etc

Climate Change: The biggest challenge of 21st Century

Climate change is the long-term alteration of temperature and weather patterns in a place, causing weather patterns to be less predictable.

Climate change has also been connected with other damaging weather events such as more frequent and more intense hurricanes, floods, downpours, winter storms, and **wildfires**.

Photograph credit National Geographic

Wildfires

A wildfire is an uncontrolled fire that burns in the wildland vegetation, often in rural areas. Wildfires can burn in forests, grasslands, savannas, and other ecosystems, and have been doing so for hundreds of millions of years.

While many plants and animals need and benefit from wildfires, climate change has left some ecosystems more susceptible to flames, especially in the southwest United States and the Amazonas (largest ecosystem in the planet)

Photograph credit Reuters

Wildfires

A wildfire is an uncontrolled fire that burns in the wildland vegetation, often in rural areas. Wildfires can burn in forests, grasslands, savannas, and other ecosystems, and have been doing so for hundreds of millions of years.

While many plants and animals need and benefit from wildfires, climate change has left some ecosystems more susceptible to flames, especially in the southwest United States and the Amazonas (largest ecosystem in the planet!)

What can AI and Machine Learning do to tackle Cilmate Change?

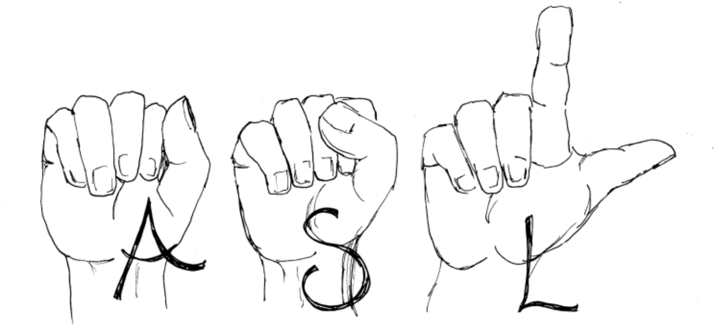
Proposal: Train a Deep Nueral Network to spot Early Forest Fires

Photograph credit Reuters

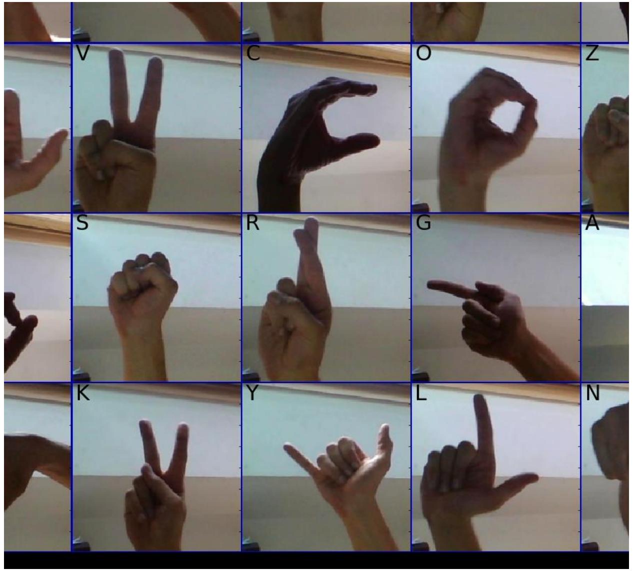
Image Classifier

Based on Notebook wrtitten by

Mostafa Abdelaleem, IBM Develper

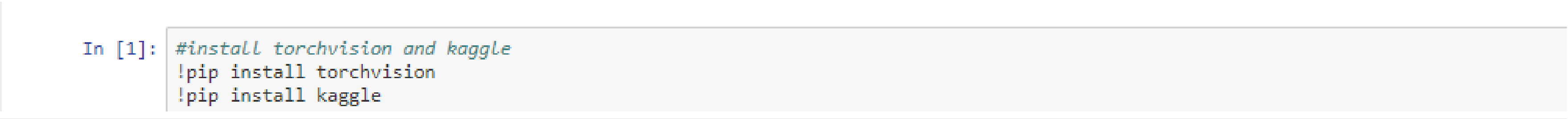


1- Obtain dataset from Kaggle.
2- Explore data and define transformers
3- Define a Deep Learning classifier
4- Train the classifier we defined.
5- Test the model we trained.



- **Python 3.7** environment
- **CPU Environment** train Deep Neural Network
- **Watson Studio Environments**

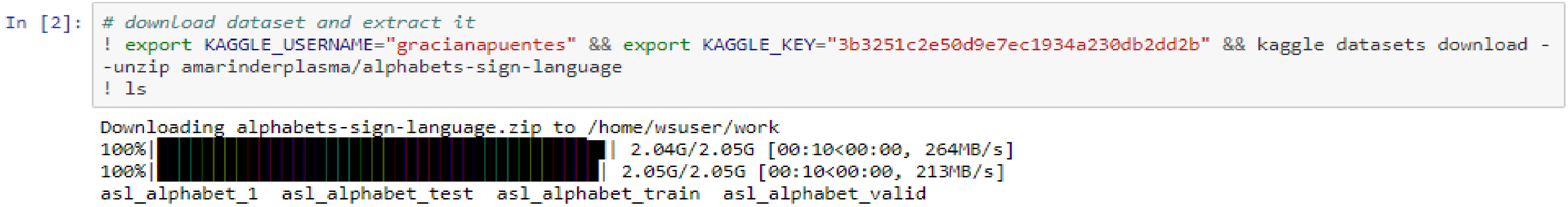
1-Packages: Kaggle CLI and TorchVision Packages



```
In [1]: #install torchvision and kaggle
        !pip install torchvision
        !pip install kaggle
```

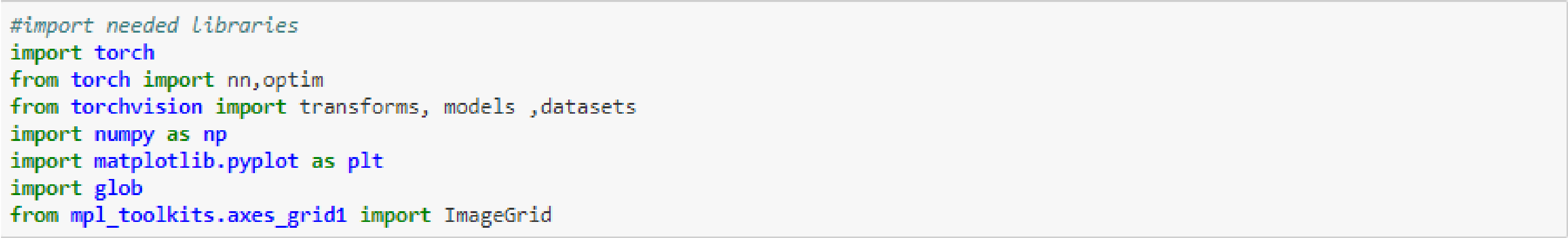
Download and Unzip Data

- Create Account in Kaggle
- Use Kaggle Directory unzip a **phylake1337/fire-dataset**
- https://www.kaggle.com/phylake1337/fire-dataset?select=fire_dataset



```
In [2]: # download dataset and extract it
        ! export KAGGLE_USERNAME="gracianapuentes" && export KAGGLE_KEY="3b3251c2e50d9e7ec1934a230db2dd2b" && kaggle datasets download -
        -unzip amarinderplasma/alphabets-sign-language
        ! ls

        Downloading alphabets-sign-language.zip to /home/wsuser/work
        100%|████████████████████████████████████████| 2.04G/2.05G [00:10<00:00, 264MB/s]
        100%|███████████████████████████████████████| 2.05G/2.05G [00:10<00:00, 213MB/s]
        asl_alphabet_1  asl_alphabet_test  asl_alphabet_train  asl_alphabet_valid
```

Import Libraries



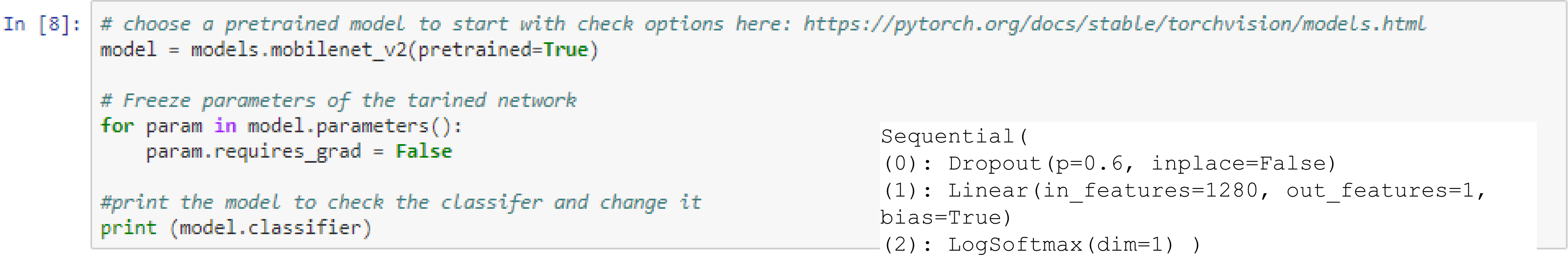
```
#import needed libraries
import torch
from torch import nn,optim
from torchvision import transforms, models ,datasets
import numpy as np
import matplotlib.pyplot as plt
import glob
from mpl_toolkits.axes_grid1 import ImageGrid
```

2- Load images to data loaders

Create data loaders to feed our model with data during the training phase.

We will use ImageFolder from TorchVision to load our data.

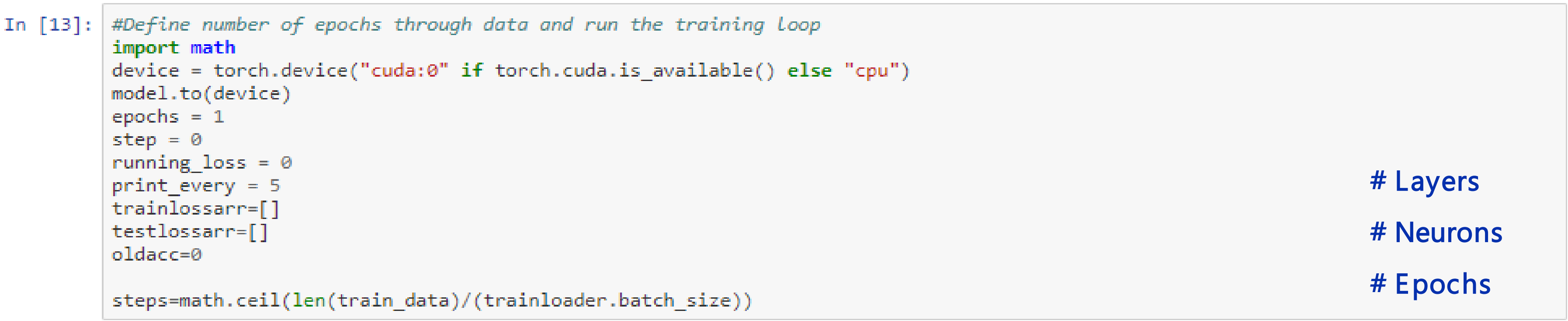
3- Customize Neural Network using Classifier



```
In [8]: # choose a pretrained model to start with check options here: https://pytorch.org/docs/stable/torchvision/models.html
        model = models.mobilenet_v2(pretrained=True)

        # Freeze parameters of the tarined network
        for param in model.parameters():
            param.requires_grad = False

        #print the model to check the classifer and change it
        print (model.classifier)
```

```
Sequential(
(0): Dropout(p=0.6, inplace=False)
(1): Linear(in_features=1280, out_features=1,
bias=True)
(2): LogSoftmax(dim=1) )
```

4- Configure Training parameters



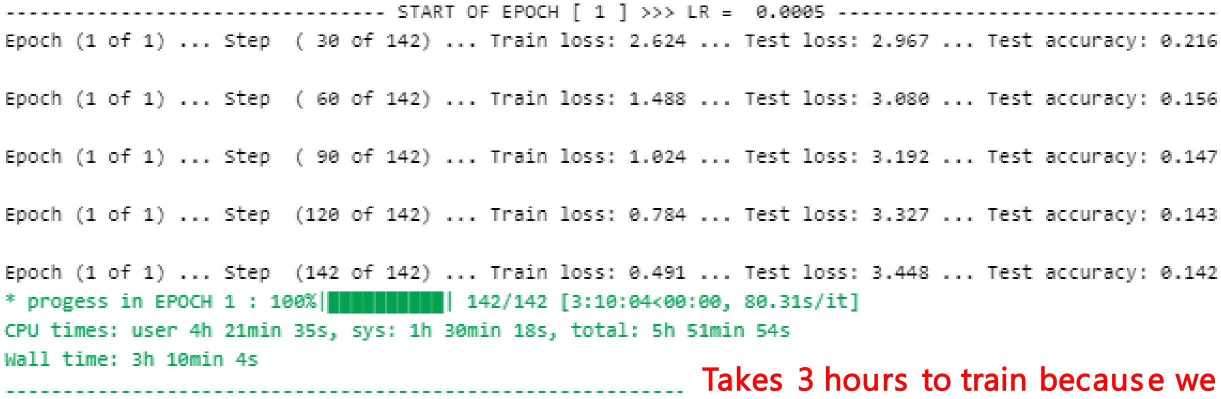
```
In [13]: #Define number of epochs through data and run the training loop
         import math
         device = torch.device("cuda:0" if torch.cuda.is_available() else "cpu")
         model.to(device)
         epochs = 1
         step = 0
         running_loss = 0
         print_every = 5
         trainlossarr=[]
         testlossarr=[]
         oldacc=0

         steps=math.ceil(len(train_data)/(trainloader.batch_size))
```

# Layers

# Neurons

# Epochs

5- Start the Training Loop….



```
---------------------------------- START OF EPOCH [ 1 ] >>> LR =  0.0005 ----------------------------------
Epoch (1 of 1) ... Step  ( 30 of 142) ... Train loss: 2.624 ... Test loss: 2.967 ... Test accuracy: 0.216

Epoch (1 of 1) ... Step  ( 60 of 142) ... Train loss: 1.488 ... Test loss: 3.080 ... Test accuracy: 0.156

Epoch (1 of 1) ... Step  ( 90 of 142) ... Train loss: 1.024 ... Test loss: 3.192 ... Test accuracy: 0.147

Epoch (1 of 1) ... Step  (120 of 142) ... Train loss: 0.784 ... Test loss: 3.327 ... Test accuracy: 0.143

Epoch (1 of 1) ... Step  (142 of 142) ... Train loss: 0.491 ... Test loss: 3.448 ... Test accuracy: 0.142
* progess in EPOCH 1 : 100%|██████████| 142/142 [3:10:04<00:00, 80.31s/it]
CPU times: user 4h 21min 35s, sys: 1h 30min 18s, total: 5h 51min 54s
Wall time: 3h 10min 4s
------------------------------------------------------------
```

Takes 3 hours to train because we are running on CPU …

6-Test Model After Trainning

```
Classification

Predicted Class: Non Fire Image
```

Out[1]

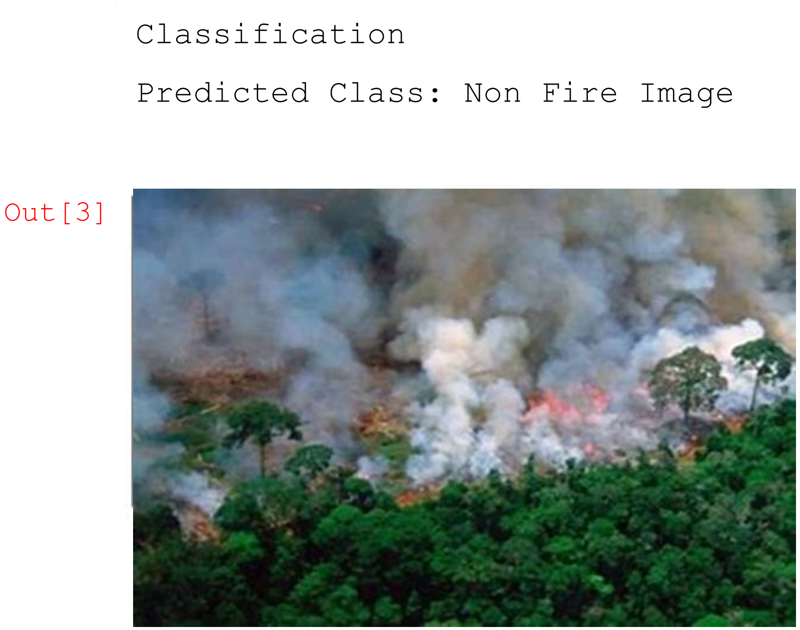
True-

```
Classification

Predicted Class: Fire Image
```

Out[2]

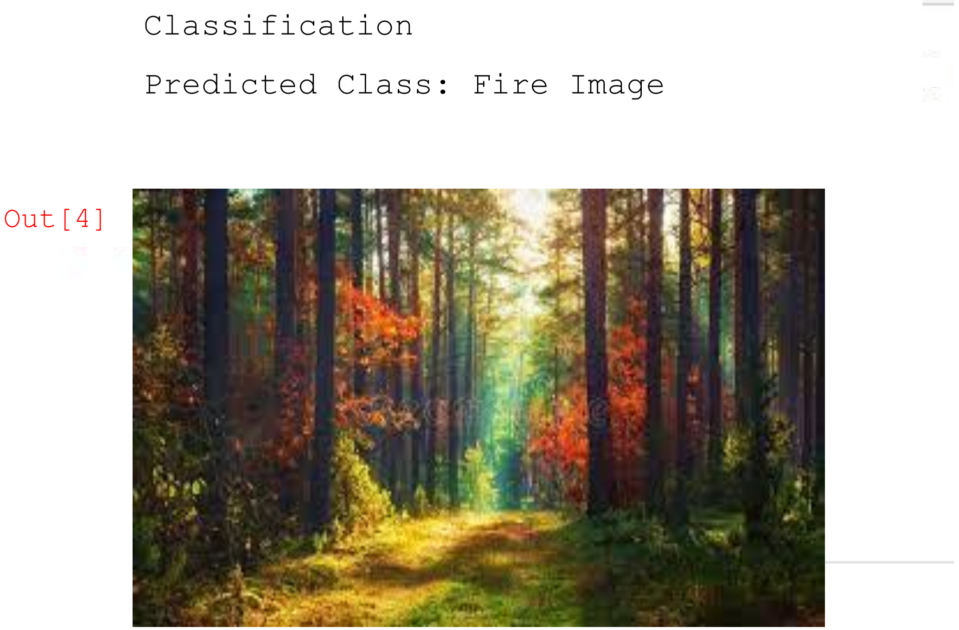
True+

6-Test Model After Trainning



```
Classification

Predicted Class: Non Fire Image

Out[3]
```

False-



```
Classification

Predicted Class: Fire Image

Out[4]
```

False+

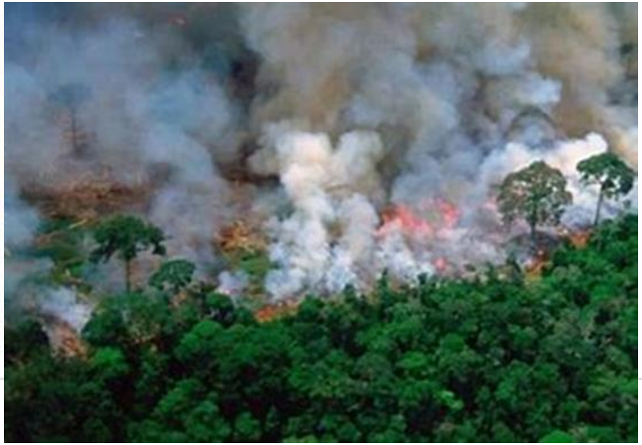
6-Test Model AfterTrainning

```
Classification

Predicted Class: Non Fire Image
```

```
Out[5]
```

```
Classification

Predicted Class: Non Fire Image
```

```
Out[6]
```



Accuracy: (True+ + True-)/(N+ +N-)

Detection rate: True+/N+

False Alarm rate: False-/N-

Accuracy ~ 82%

Detection rate ~ 70%

False Alarm rate ~ 9%

Preliminary

Results

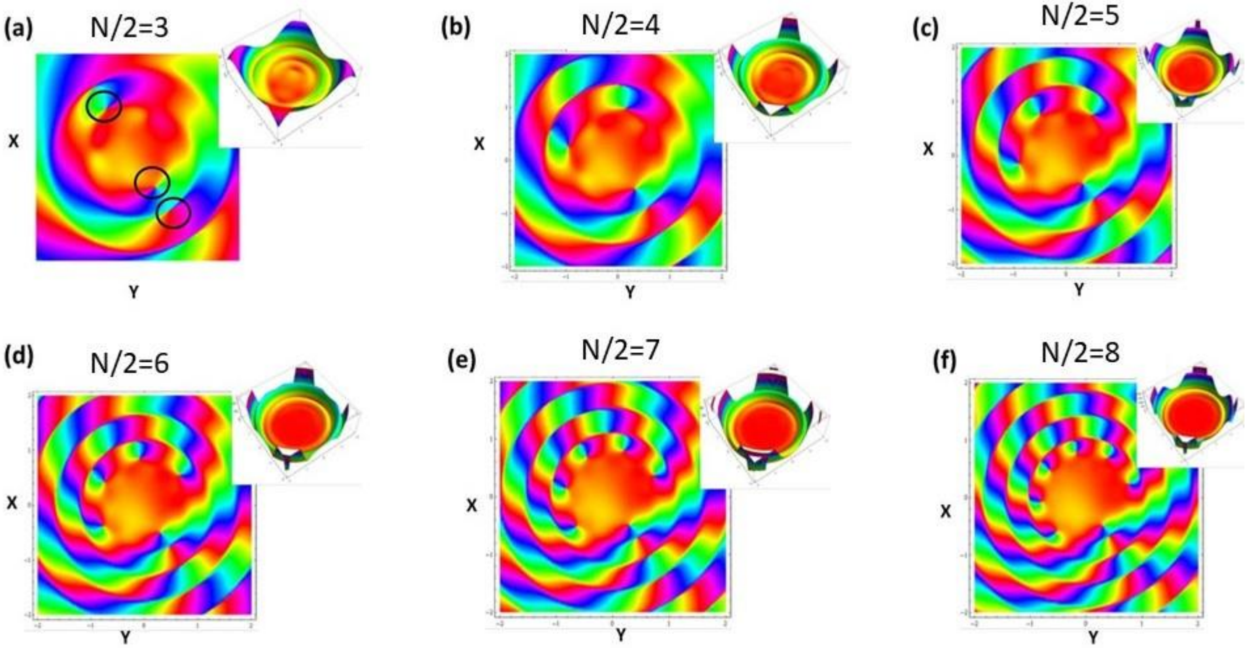
Outlook

- Improve efficiency running on GPU
- Improve accuracy using larger sets of data (from National Geographyc)
- Test other predefined classifiers
- Software integrated in Automated Surveillance Cameras
- Remotely connected to the Cloud
- Controlled via a phone or tablet applications

Other applications of Deep Learning
Classification of Quantum Vortex States

Quantum Vortex Order
Depends on the
# singularites

Deep Learning can
enable classification
w/ minimal
human Intervention



G. Puentes *et al.*, accepted for publilcation in Frontires in Physics 2021

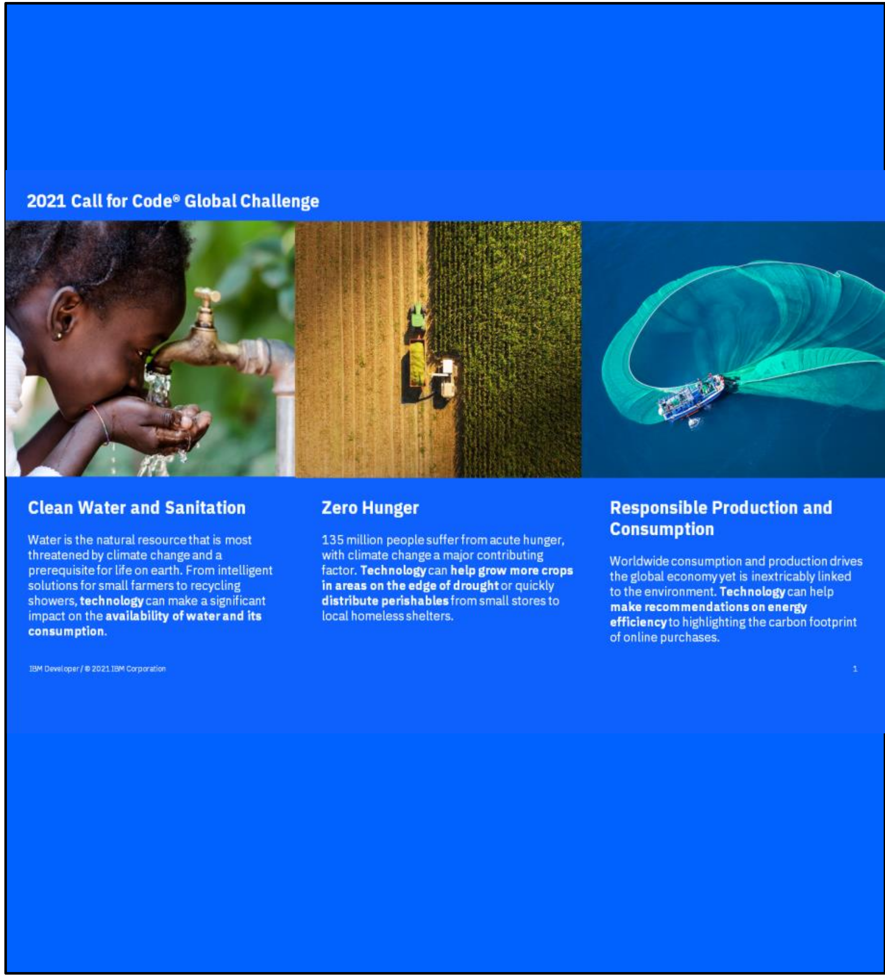
Centre for
Quantum
Technologies

Take away

- Deep Learning requires
  Good questions + Good Data

- Exploit AI to solve the most pressing issue of the 21 Century ....
  - CLIMATE CHANGE !

- IBM Cloud/Watson Studios is the perfect envirnment to get up to speed with Machine Learning !



2021 Call for Code® Global Challenge

Clean Water and Sanitation

Water is the natural resource that is most threatened by climate change and a prerequisite for life on earth. From intelligent solutions for small farmers to recycling showers, **technology** can make a significant impact on the **availability of water and its consumption.**

Zero Hunger

135 million people suffer from acute hunger, with climate change a major contributing factor. **Technology** can **help grow more crops in areas on the edge of drought** or quickly **distribute perishables** from small stores to local homeless shelters.

Responsible Production and Consumption

Worldwide consumption and production drives the global economy yet is inextricably linked to the environment. **Technology** can help **make recommendations on energy efficiency** to highlighting the carbon footprint of online purchases.

IBM Developer / © 2021 IBM Corporation

1

Thank you