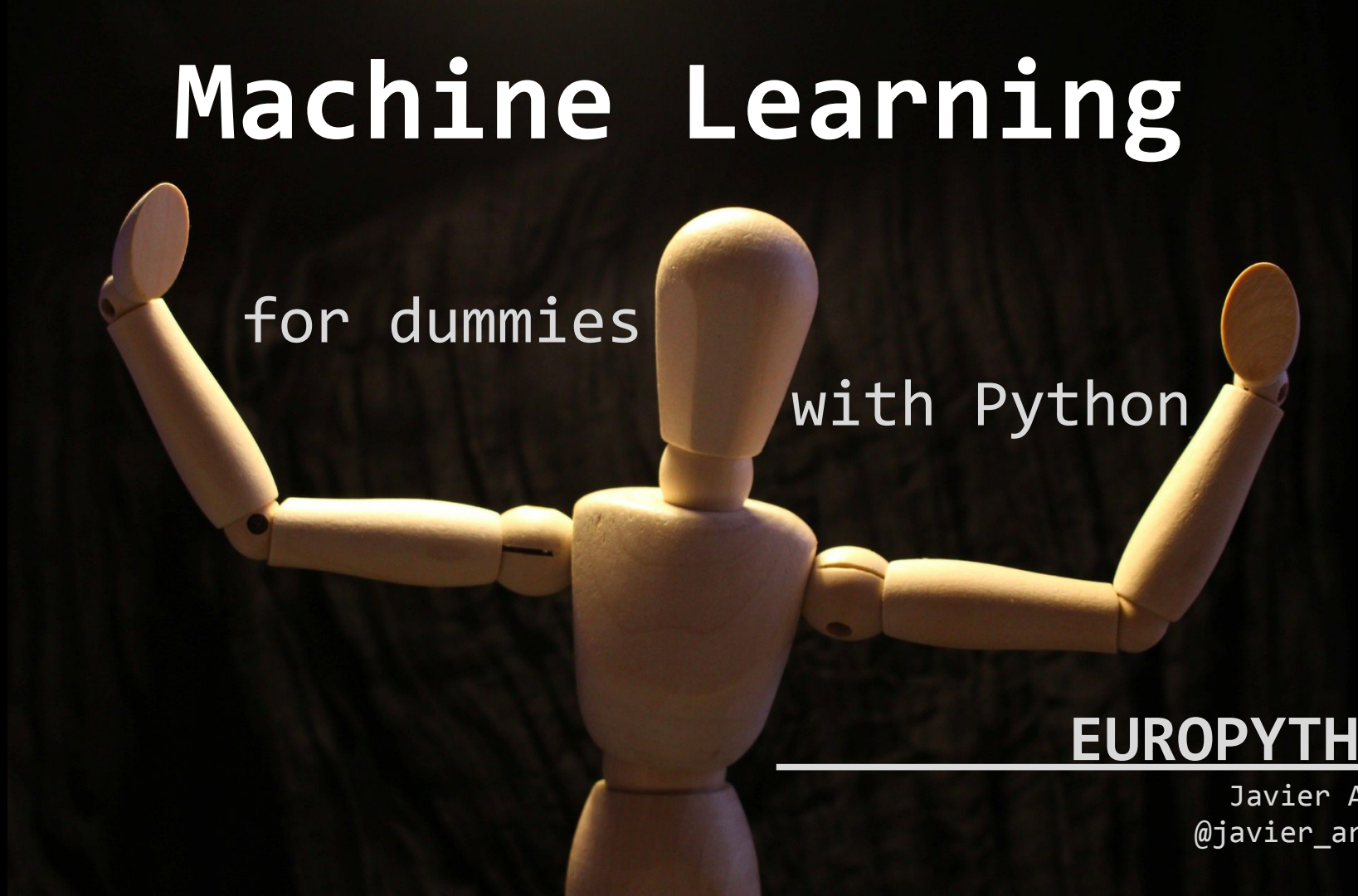
Machine Learning

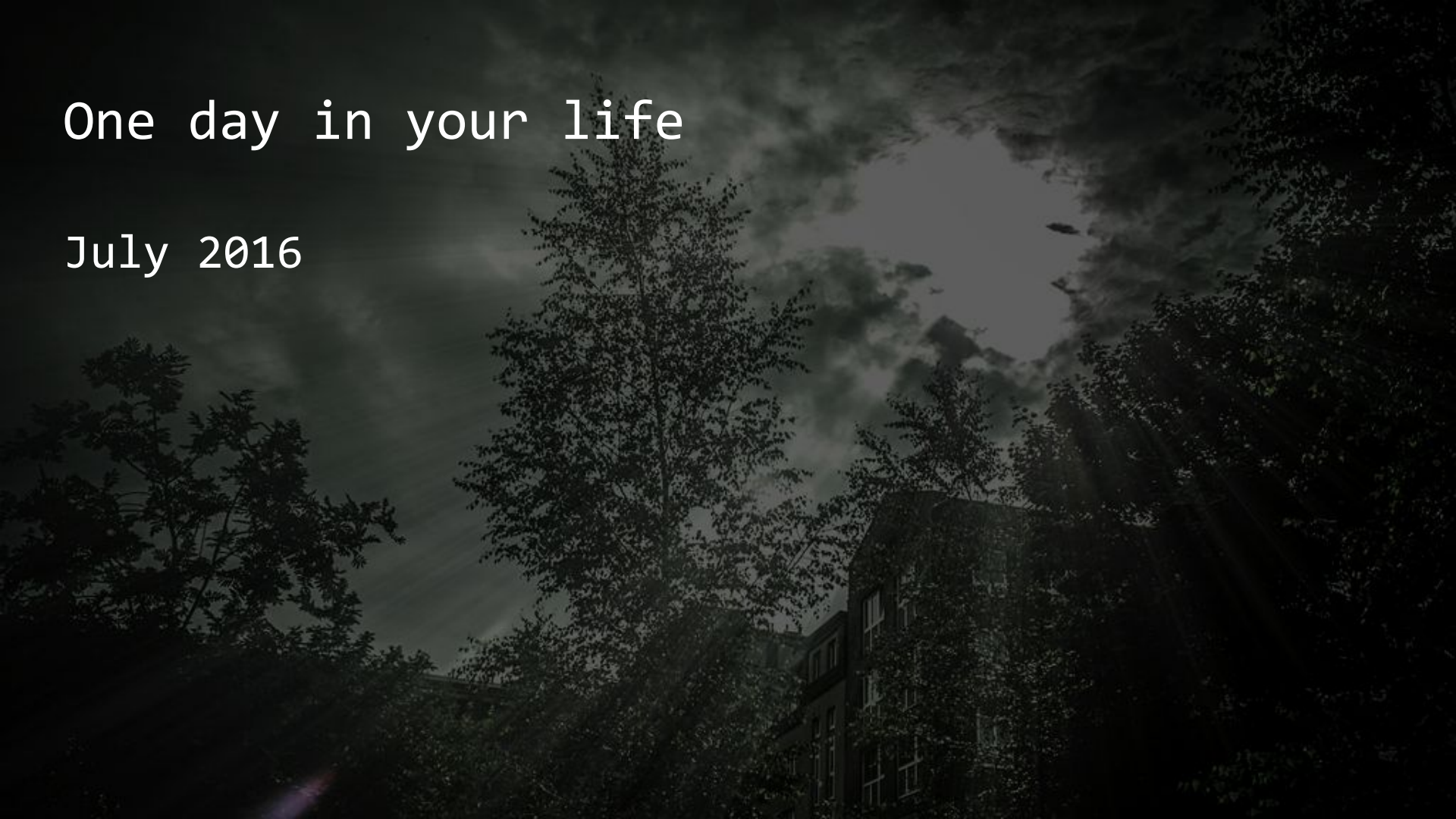


for dummies

with Python

EUROPYTHON

Javier Arias
@javier_arilos


One day in your life
July 2016

One day in your life

Time to leave the office

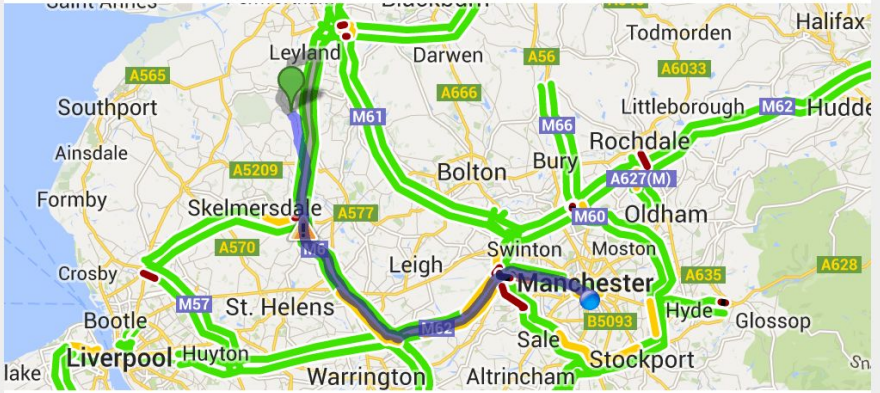
google now next meeting

Search, or say “Ok Google”

hudl

52 mins to Home

Normal traffic on M6



Navigate / 52 mins via M6

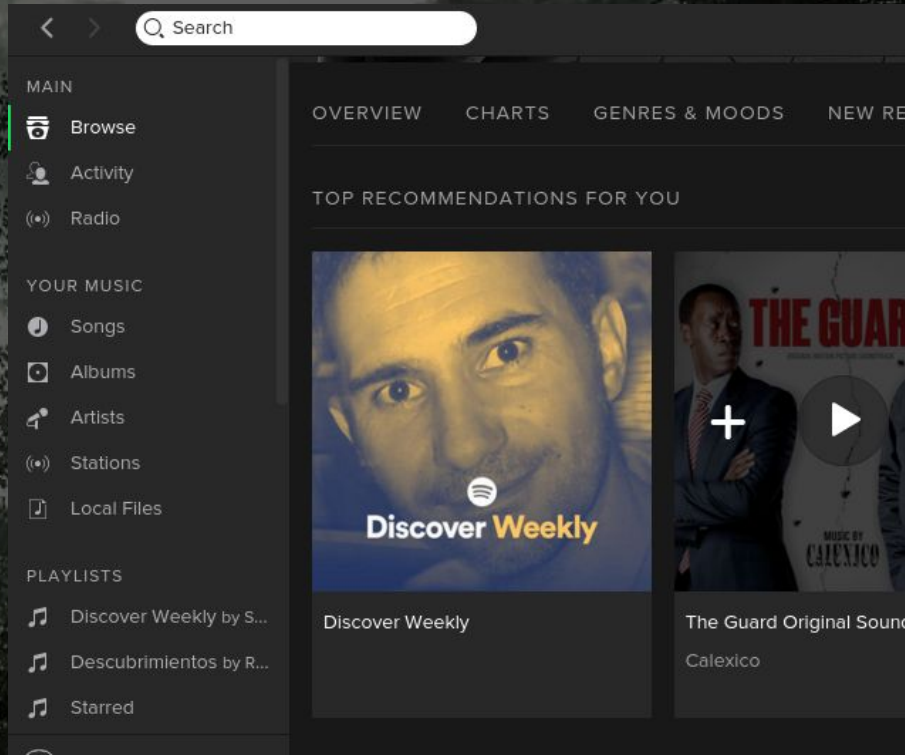
Alternative route / 1 hr 7 min via M6

One day in your life

Tesla autopilot

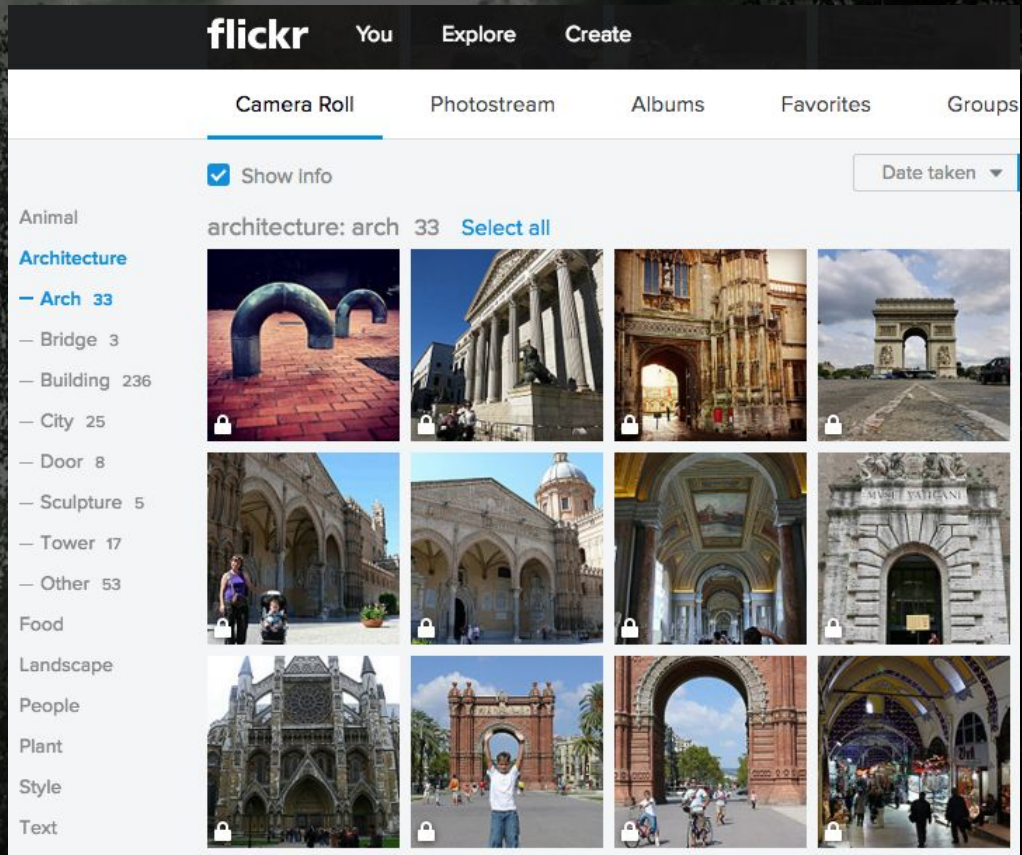
One day in your life

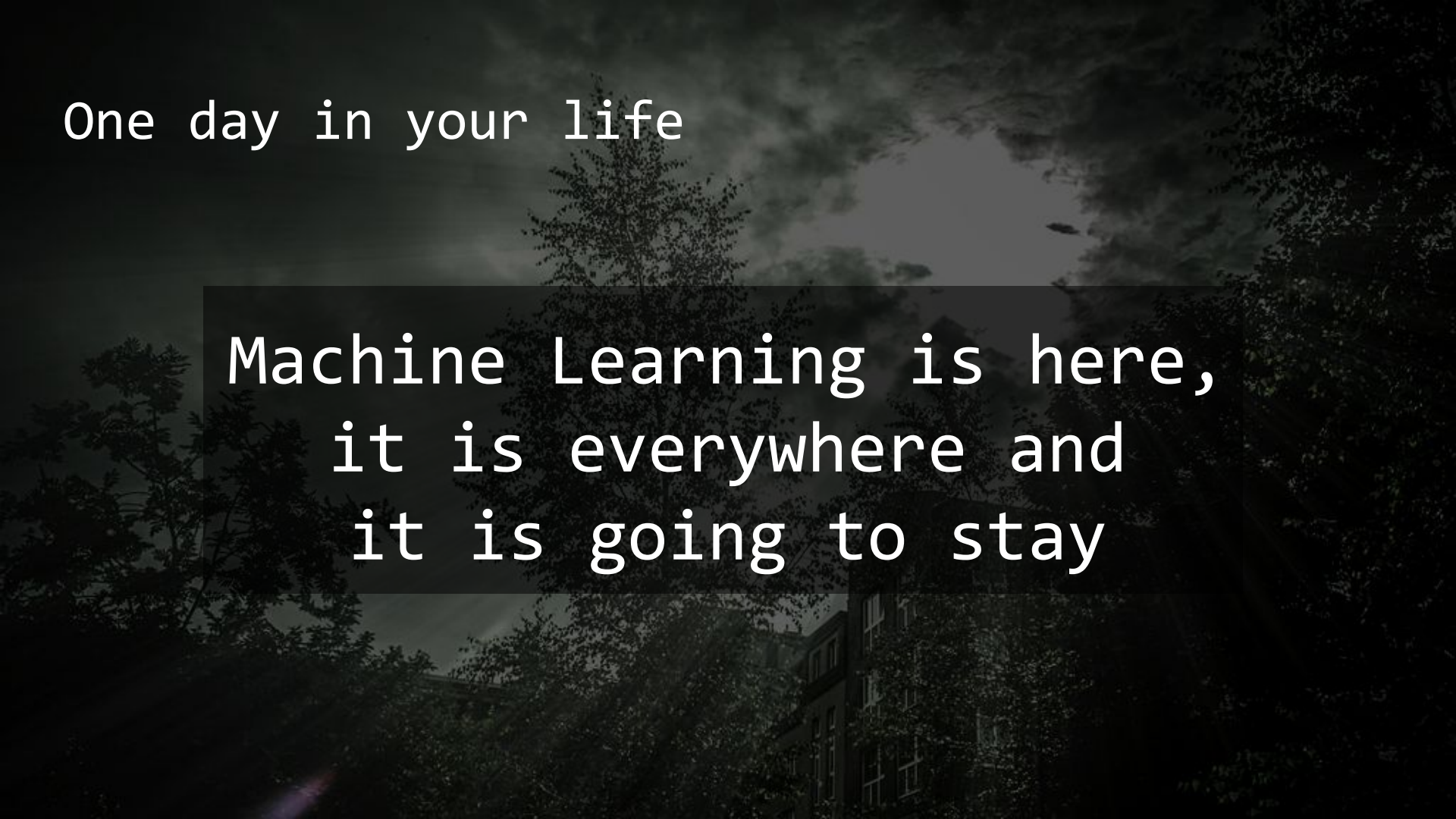
Playing music



One day in your life

Your photos organized




One day in your life
Machine Learning is here,
it is everywhere and
it is going to stay

About this presentation

Why Machine Learning (ML) matters

A journey on Machine Learning

Some ML technologies and resources

Some basic ML concepts, with code samples

Machine Learning is the next big thing

Are machines already intelligent?

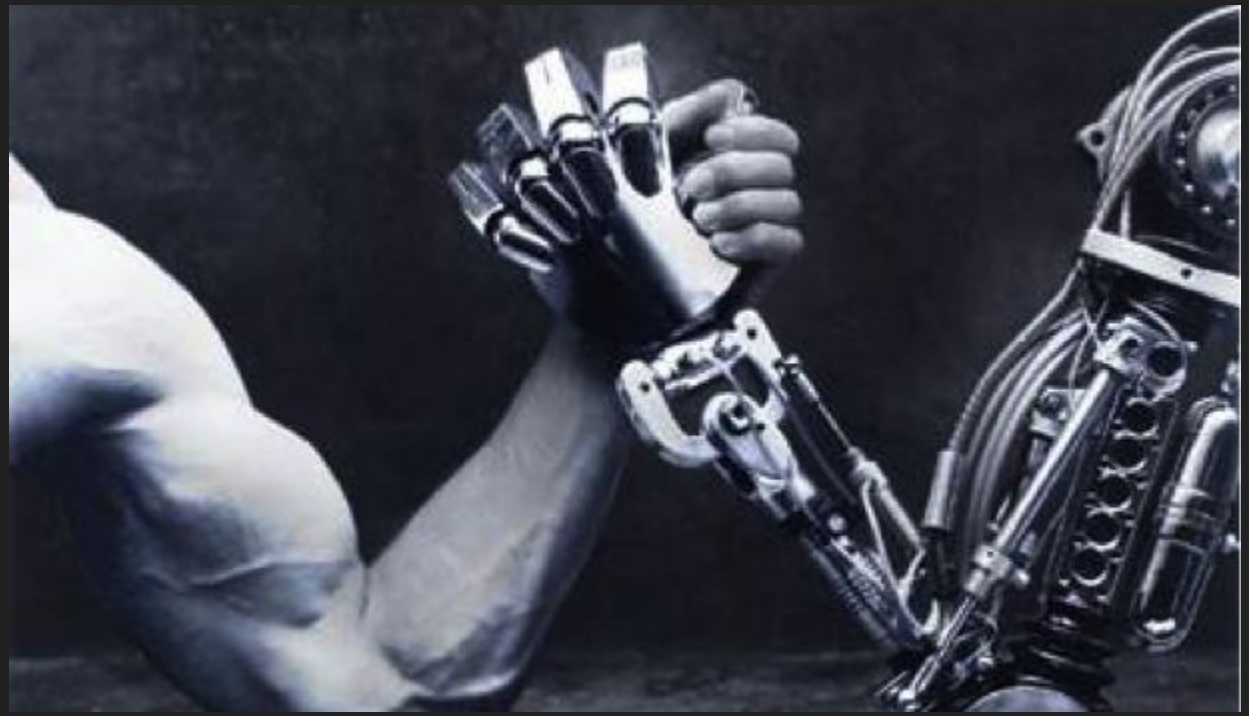
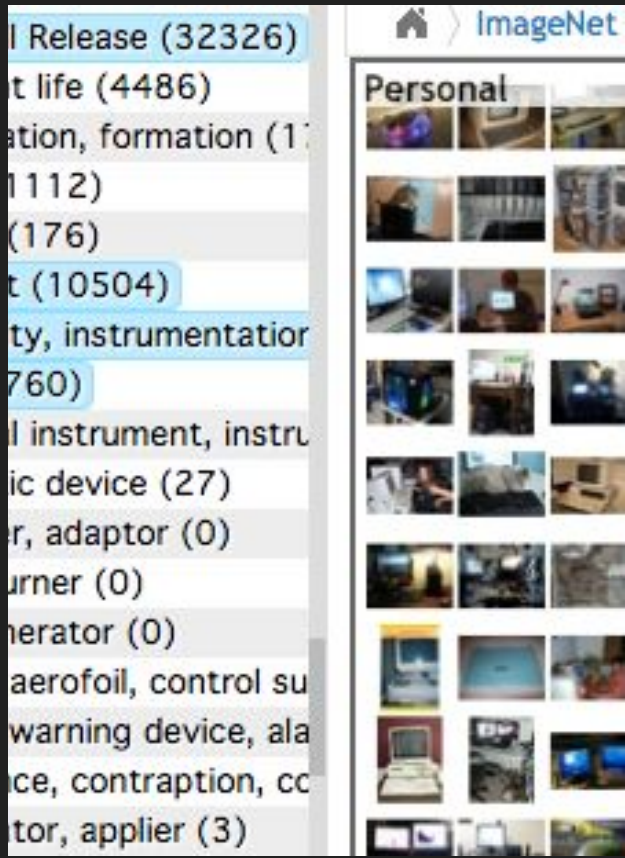


Image-net challenge

2015: machines outperform people

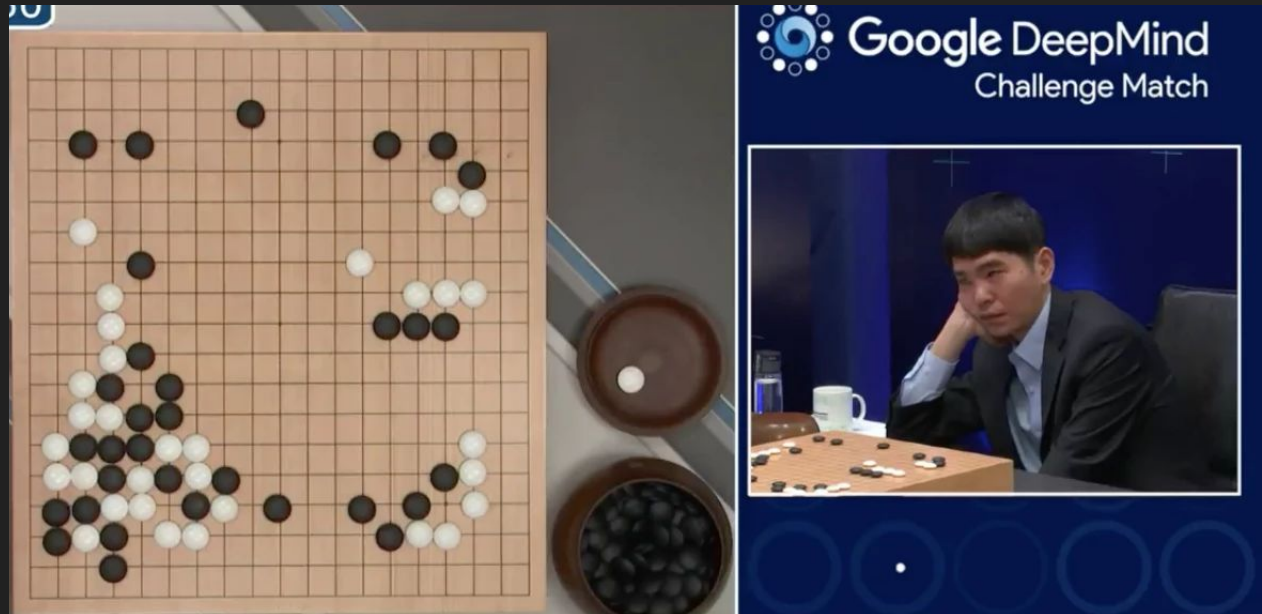


Chess

1997: Deepblue defeats Kasparov

Game of Go

2016: AlphaGo wins world champion Lee Sedol




TERMINATOR
RISE OF THE MACHINES


The journey

Learning about ML

MOOC - Massive Open Online Courses

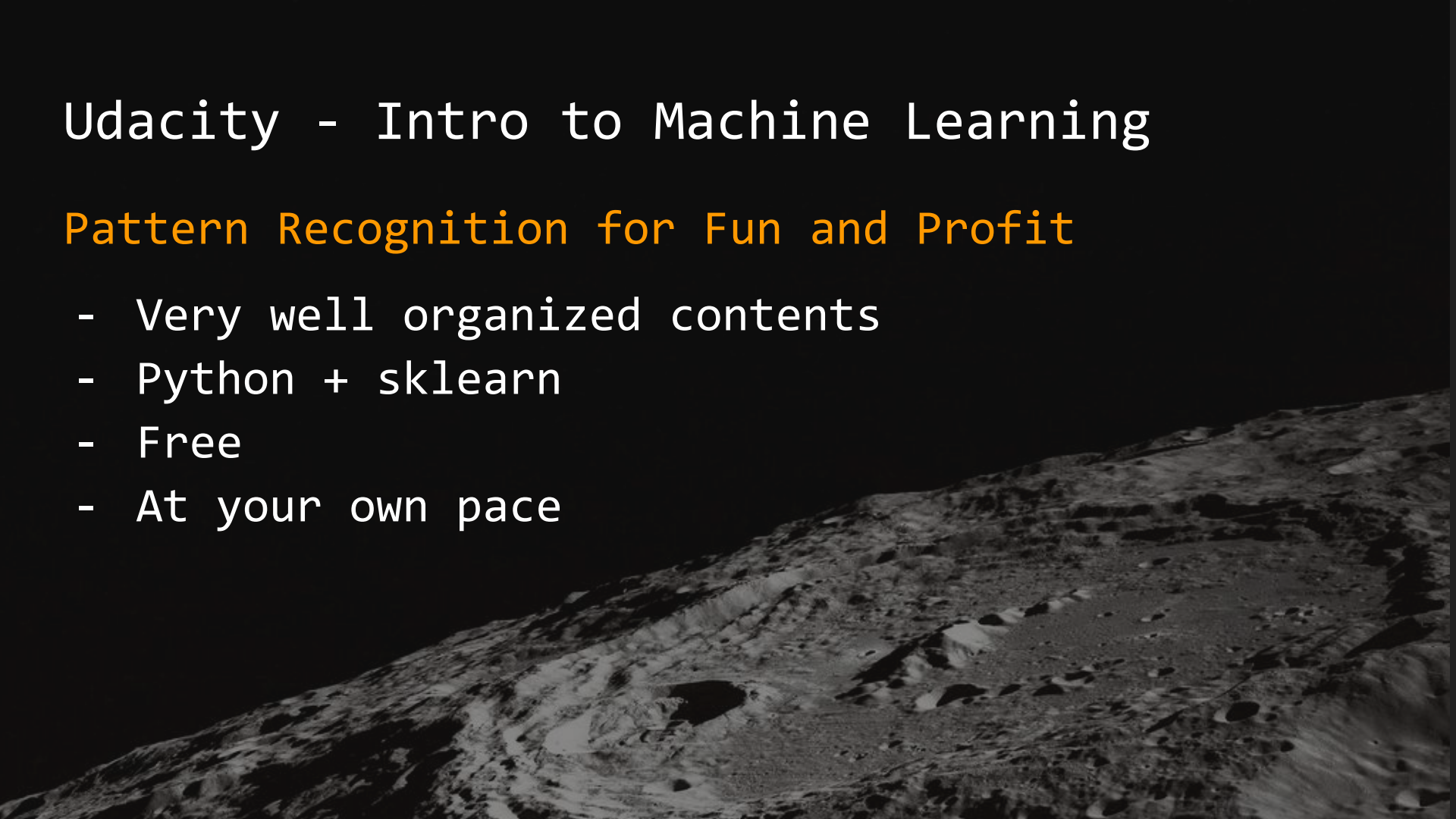
Contents by the best universities and companies

Udacity, Coursera, EdX

Udacity - Intro to Machine Learning

Pattern Recognition for Fun and Profit

- Very well organized contents
- Python + sklearn
- Free
- At your own pace



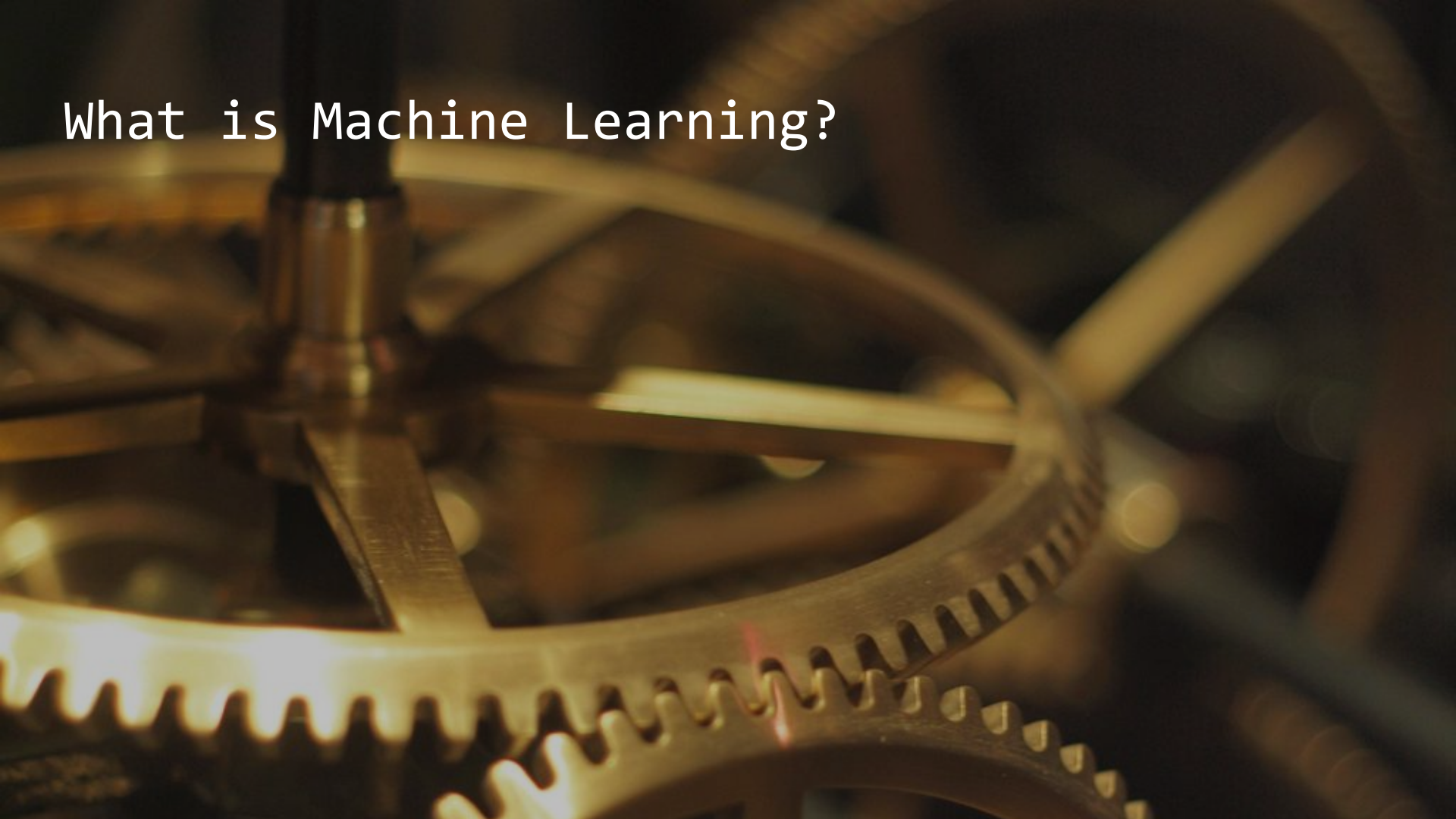
Udacity - Intro to Machine Learning

Pattern Recognition for Fun and Profit

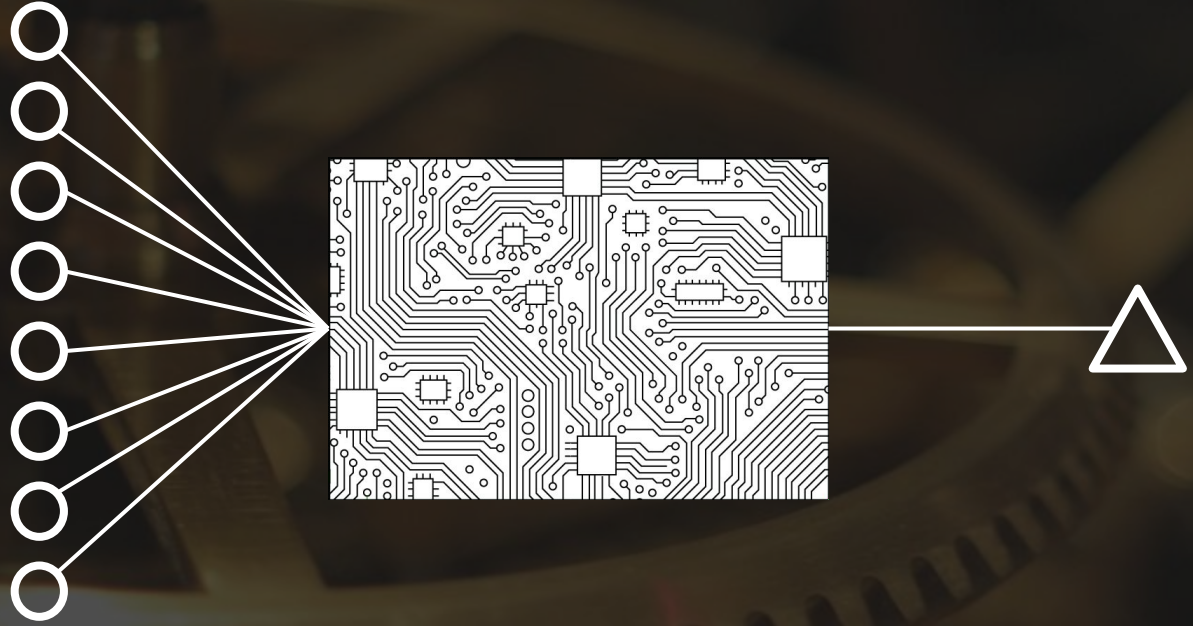
Udacity - Intro to Machine Learning

Pattern Recognition for Fun and Profit

What is Machine Learning?



Solving a complex problem



features
(data)

something

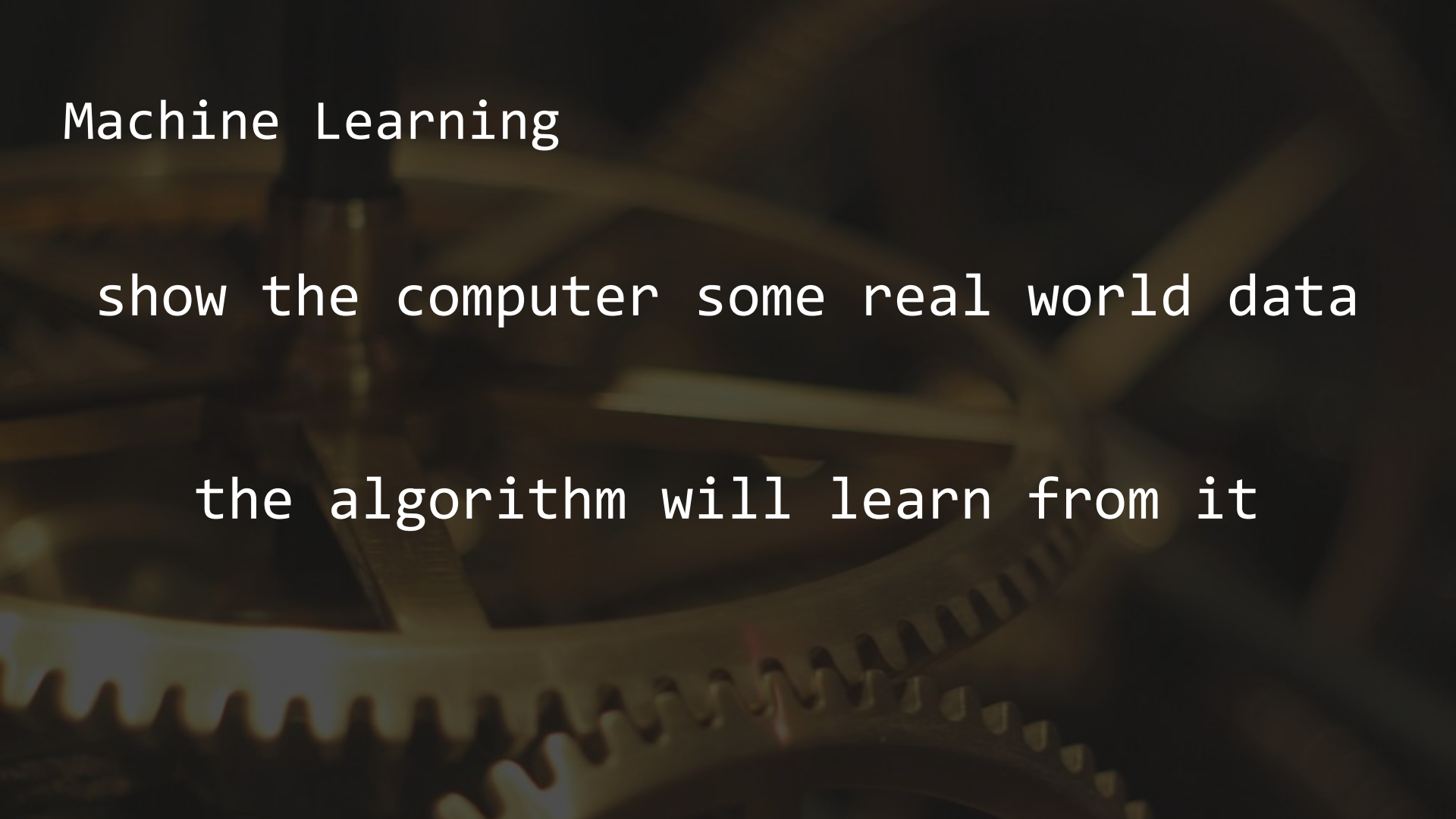
prediction

First approach, programming

tell the computer what to do in very tiny steps

First approach, programming

programming does not scale for very complex problems...


Machine Learning
show the computer some real world data
the algorithm will learn from it

Machine Learning, implications

we can train computers to do things we do not know how to do


CAN YOU ELABORATE THAT?

ML example: character recognition

Not-MNIST dataset

Thousands of 28x28 grayscale images with labels

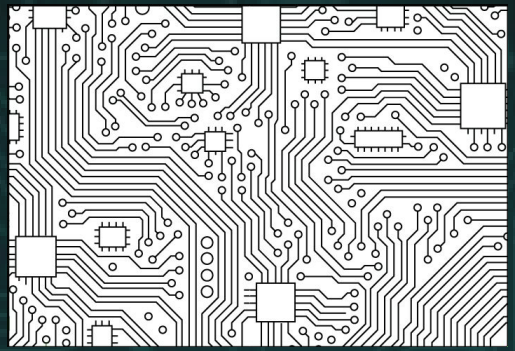
ML step 1: get samples (training data)

features x 1000s

F G F J

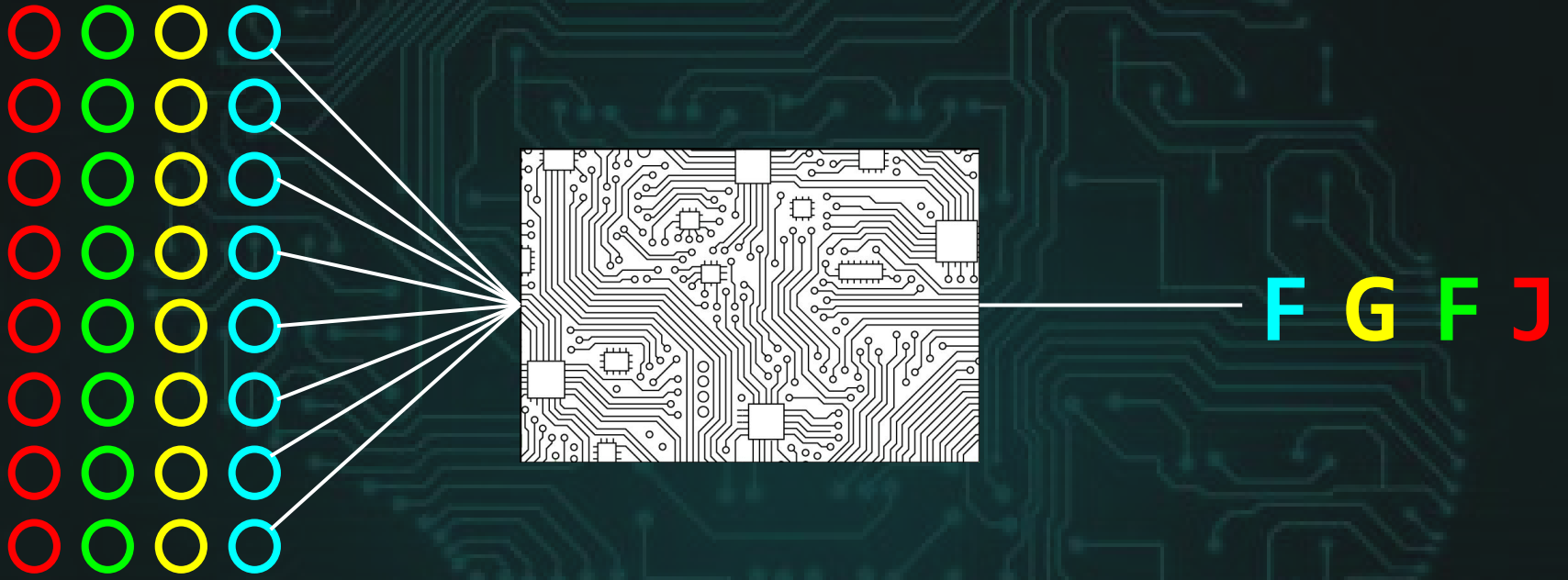
labels x 1000s

ML step 2: choose an algorithm



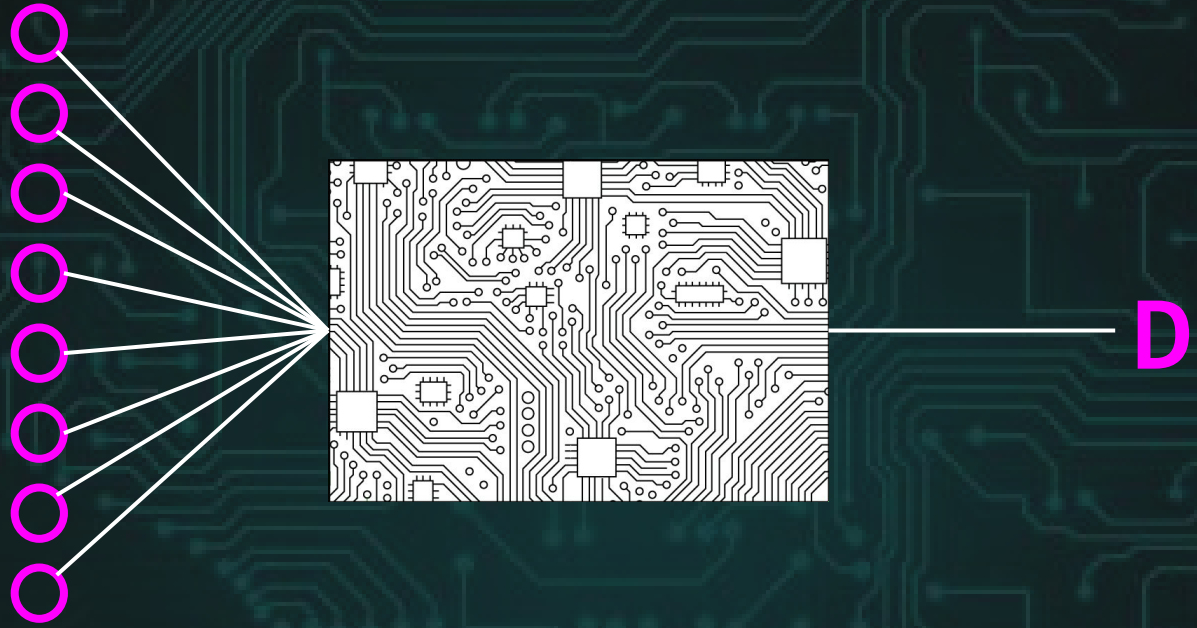
Linear regression
Support Vector Mach.
k-Means
Decision Trees
Random Forests
Neural networks
Convolutional NN
Naive Bayes

ML step 3: train your algorithm



features x 1000s ML algorithm labels x 1000s

ML, last step: getting predictions



features
(data)

ML algorithm

prediction

Tricky Question

How good are our predictions?

The Tools

The Tools: Python

- Opensource
- Expressive
- Interpreted, dynamically typed
- Widely used many different problems
- Batteries included: Notebook, Libraries

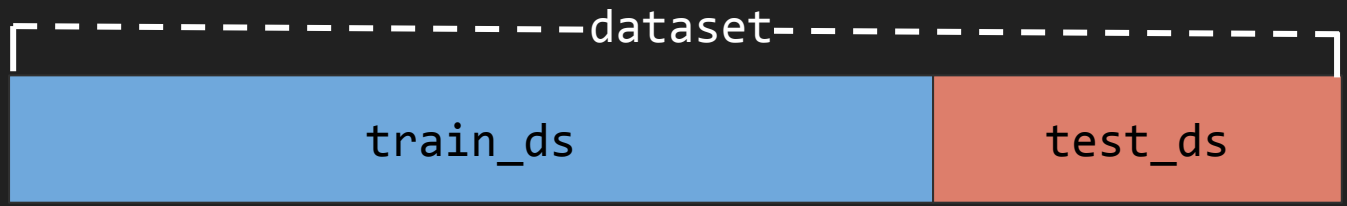
The Tools: sklearn

- Opensource, Python
- Wonderful documentation
- Support to full ML lifecycle:
  - Feature engineering
  - Algorithms
  - Validation
  - Datasets

A summary of ML process

- Get features (with labels)
- Choose and configure an algorithm
- Train your algorithm
- Do predictions
- Validate your results

train your model




```
tr_ds, _, tr_lbl, _ = train_test_split(dataset, labels,
                                       train_size=size,
                                       random_state=17)
clf = LogisticRegression()
clf.fit(tr_ds, tr_lbl) # fit with train dataset and train labels
```

make predictions

```
pred = clf.predict(test_dataset)
```

How good are our predictions?

accuracy

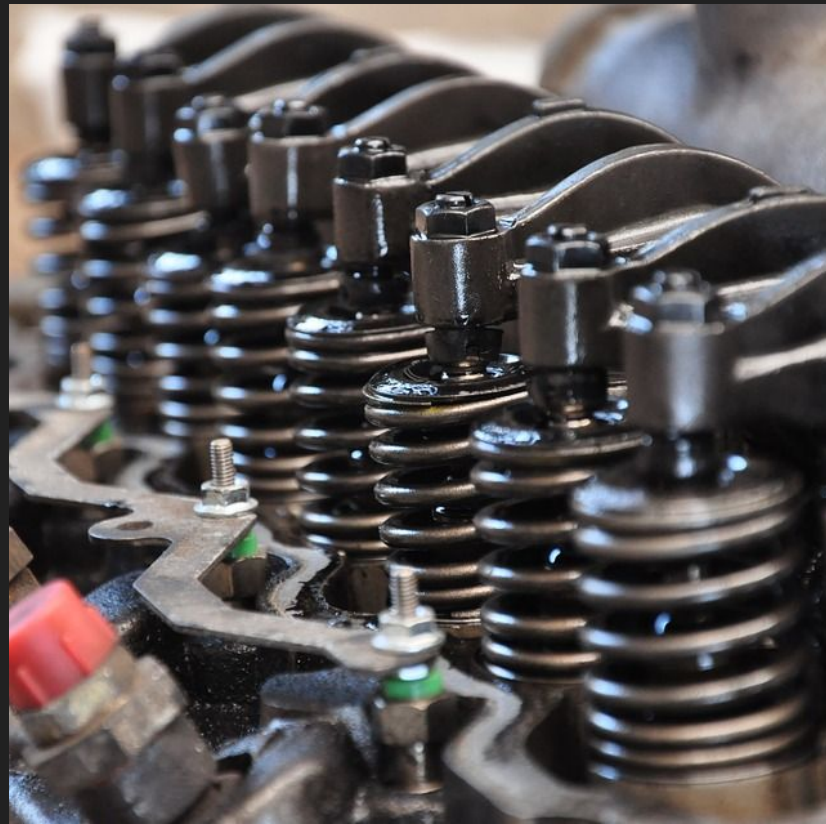
```
test_predicions = clf.predict(test_dataset)
acc = accuracy_score(test_labels, test_predictions)
```

89% accuracy

Improving prediction results

Training data

Algorithm + config



THE TWO STATES OF EVERY PROGRAMMER

I AM A GOD.

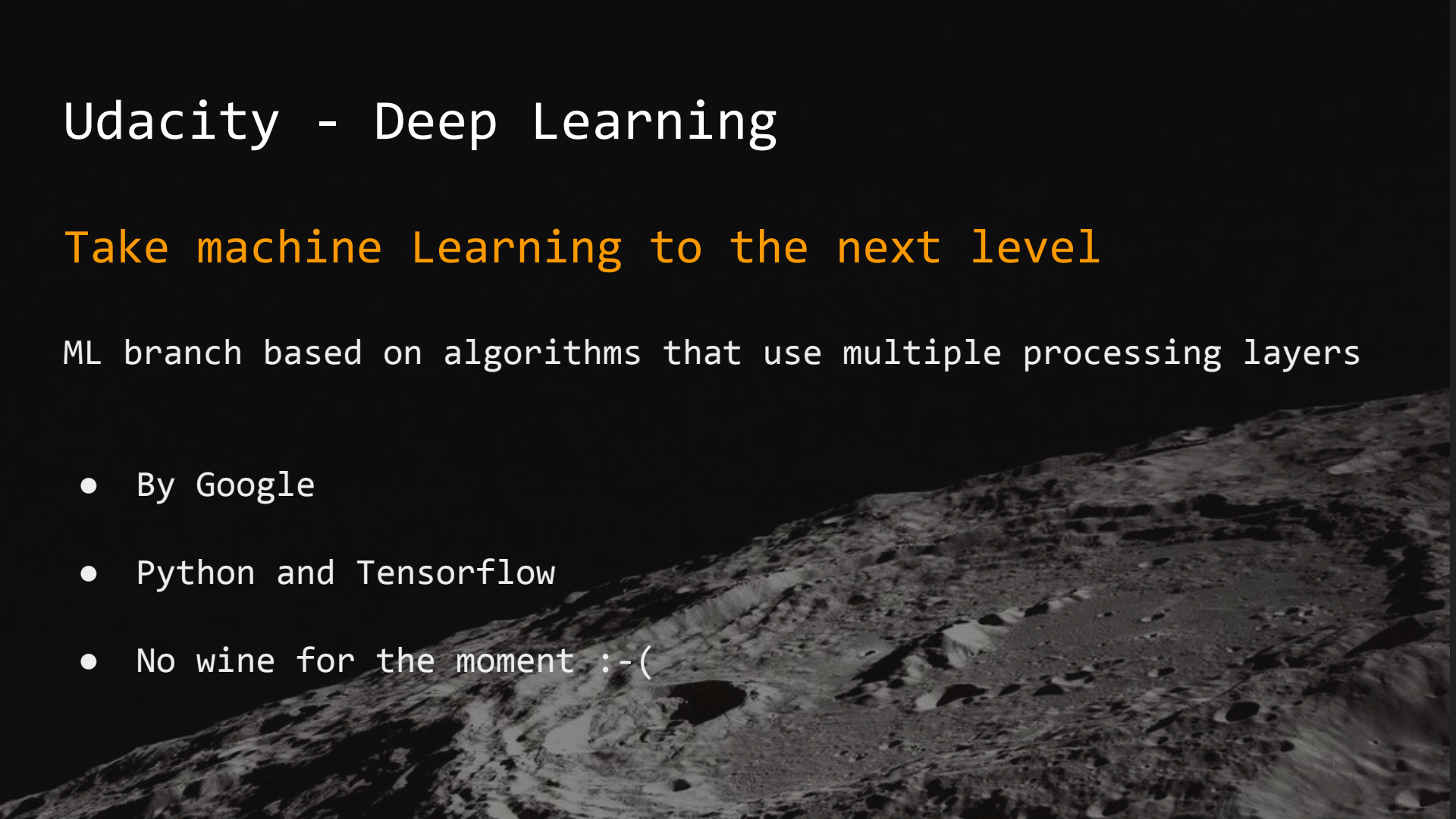
I HAVE NO IDEA WHAT I'M DOING.

Udacity - Deep Learning

Take machine Learning to the next level

ML branch based on algorithms that use multiple processing layers

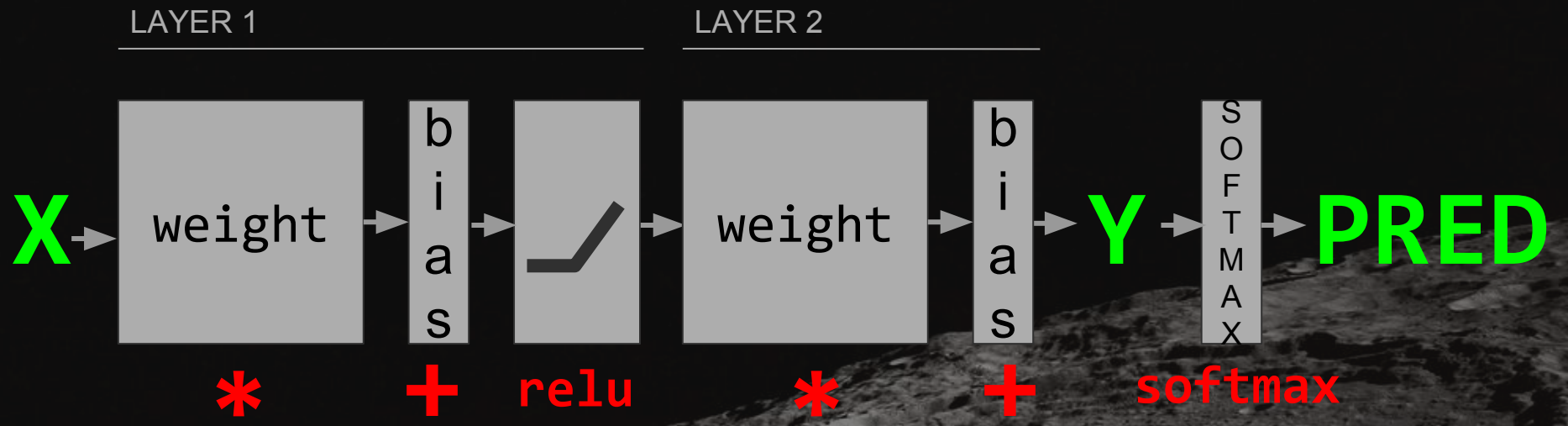
- By Google
- Python and Tensorflow
- No wine for the moment :-(



The Tools: TensorFlow

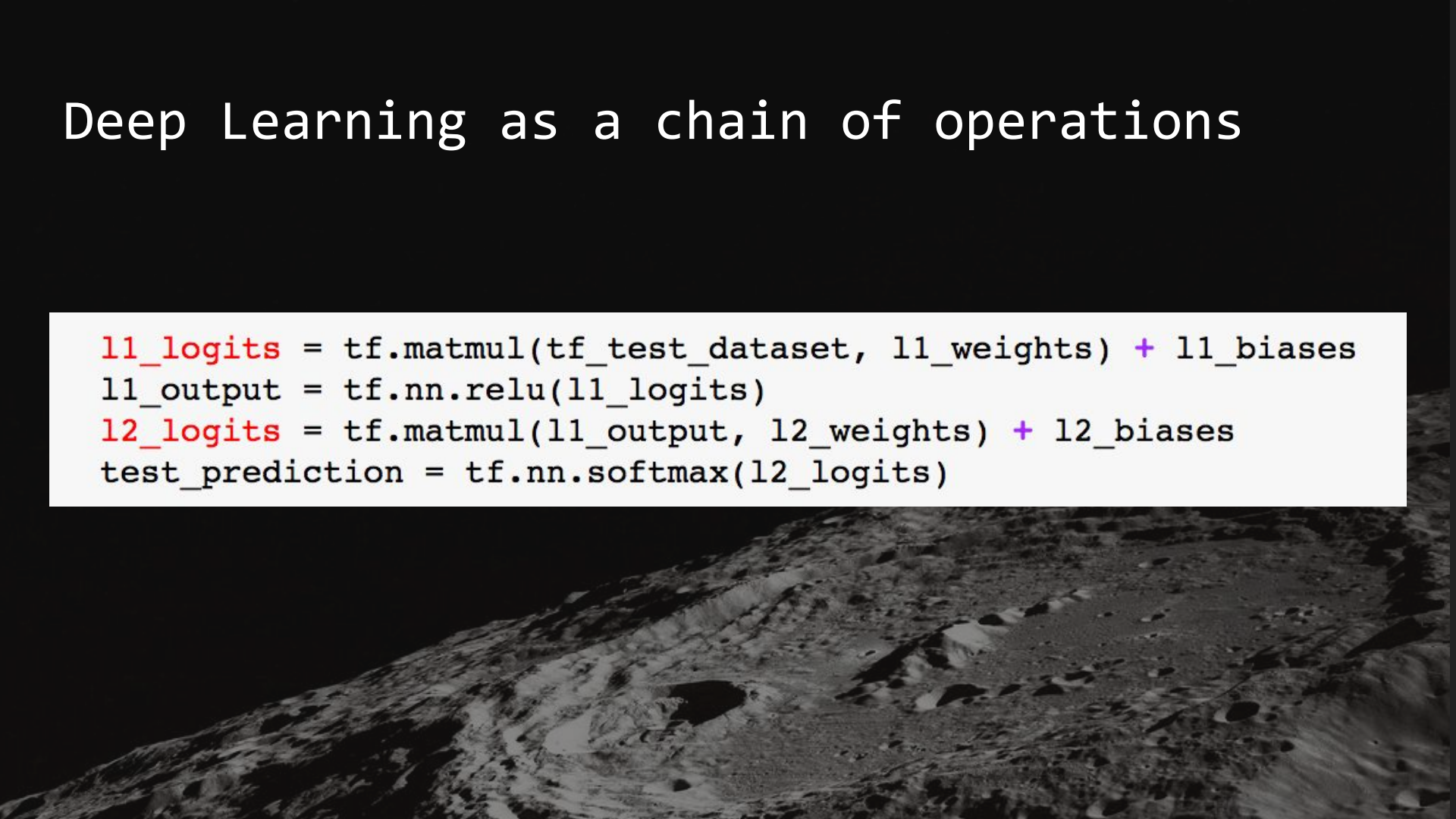
- Opensource, Python
- Deep Learning
- Data flow graphs.
  - Nodes: mathematical operations
  - Edges: Tensors, multidimensional arrays

Simplest Neural Network



Deep Learning as a chain of operations

```
l1_logits = tf.matmul(tf_test_dataset, l1_weights) + l1_biases
l1_output = tf.nn.relu(l1_logits)
l2_logits = tf.matmul(l1_output, l2_weights) + l2_biases
test_prediction = tf.nn.softmax(l2_logits)
```



Let’s recap


THE BIG
BANANA
COFFS HARBOUR


scikit
learn


VIA 9GAG.COM

Thank you for your attention