

Machine Learning Workshop



The Colab Notebook

Insert notebook link here.

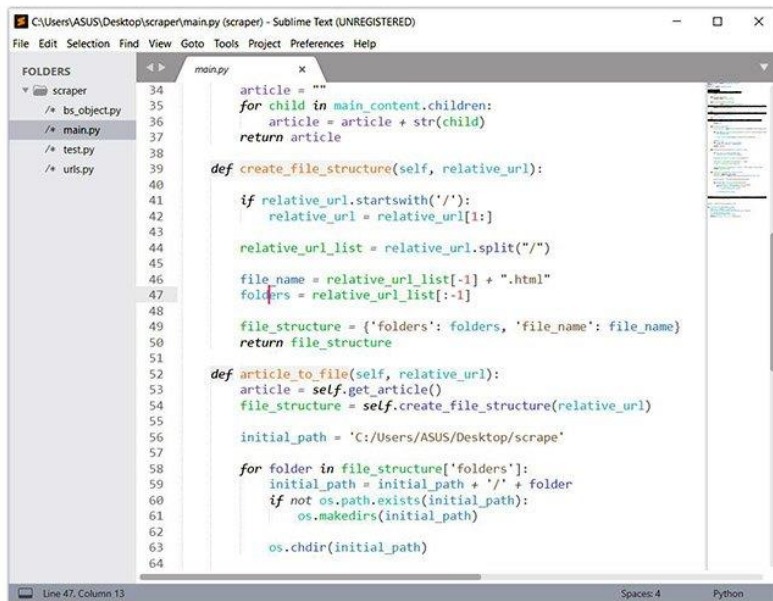
<https://go.dscuoan.nz/mlcolab>

https://colab.research.google.com/drive/1cBszWcDJ_P_9oe4K89GUhKjVMV5MxDXem?usp=sharing

colab

What is Google Colab?

An introduction to the running theme of thievery



The screenshot shows a Sublime Text editor window titled "C:\Users\ASUS\Desktop\scraper\main.py (scraper) - Sublime Text (UNREGISTERED)". The editor displays a Python script with the following code:

```
34 article = ""
35 for child in main_content.children:
36     article = article + str(child)
37 return article
38
39 def create_file_structure(self, relative_url):
40
41     if relative_url.startswith('/'):
42         relative_url = relative_url[1:]
43
44     relative_url_list = relative_url.split("/")
45
46     file_name = relative_url_list[-1] + ".html"
47     folders = relative_url_list[:-1]
48
49     file_structure = {'folders': folders, 'file_name': file_name}
50     return file_structure
51
52 def article_to_file(self, relative_url):
53     article = self.get_article()
54     file_structure = self.create_file_structure(relative_url)
55
56     initial_path = 'C:/Users/ASUS/Desktop/scraper'
57
58     for folder in file_structure['folders']:
59         initial_path = initial_path + '/' + folder
60         if not os.path.exists(initial_path):
61             os.makedirs(initial_path)
62
63         os.chdir(initial_path)
64
```

The left sidebar shows a file explorer with the following structure:

- scraper
 - bs_object.py
 - main.py
 - test.py
 - urls.py

The status bar at the bottom indicates "Line 47, Column 13", "Spaces: 4", and "Python".



What is Google Colab?

An introduction to the running theme of thievery



```

1  article = ""
2  for child in main_content.children:
3      article = article + str(child)
4  return article
5
6 def create_file_structure(self, relative_url):
7
8     if relative_url.startswith("/"):
9         relative_url = relative_url[1:]
10
11     relative_url_list = relative_url.split("/")
12
13     file_name = relative_url_list[-1] if ".html"
14     folders = relative_url_list[:-1]
15
16     file_structure = {}
17     folders, file_name = file_name
18     return file_structure
19
20 def article_to_file(self, relative_url):
21     article = self.get_article(relative_url)
22     file_structure = self.create_file_structure(relative_url)
23
24     initial_path = "C:/Users/ASUS/Desktop/scanme/"
25
26     for folder in file_structure["folders"]:
27         initial_path = initial_path + "/" + folder
28     if not os.path.exists(initial_path):
29         os.makedirs(initial_path)
30
31     os.chdir(initial_path)

```



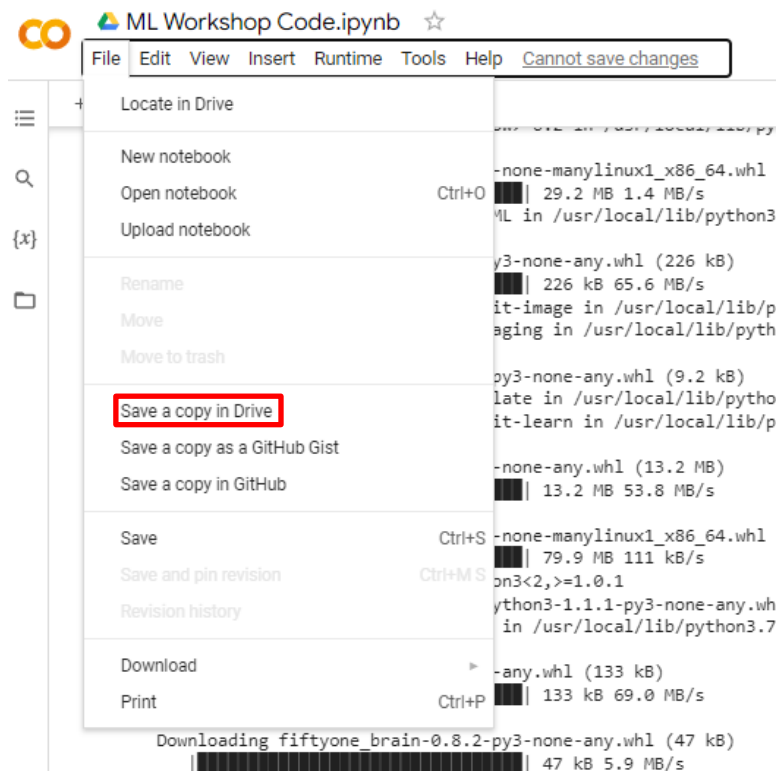
Disclaimer

The speaker may mention concepts and words such as “stealing” or “theft.”

Rest assured, what we are doing is 100% legal.

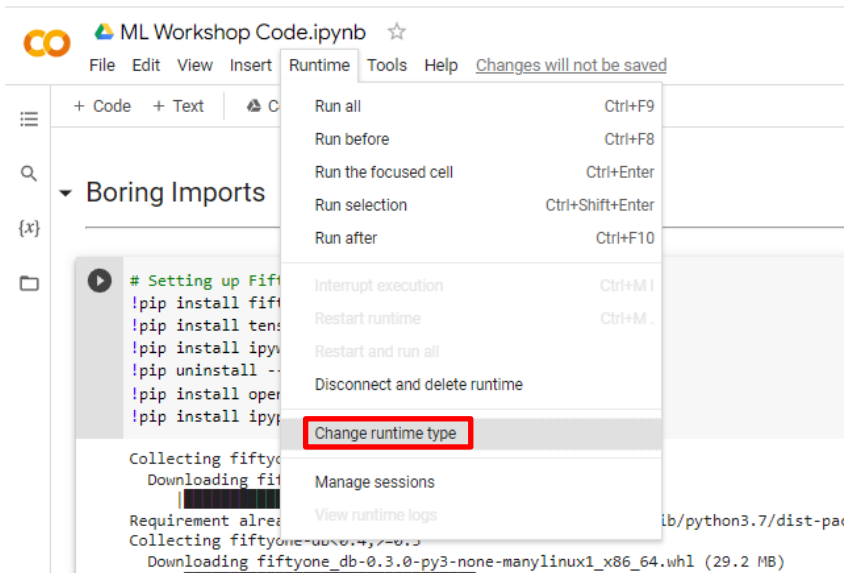
The Colab Notebook

Step One: Stealing the notebook



The Colab Notebook

Step Two: Stealing a GPU



Notebook settings

Hardware accelerator

GPU 

To get the most out of Colab, avoid using a GPU unless you need one. [Learn more](#)

☐ Background execution

Want your notebook to keep running even after you close your browser? [Upgrade to Colab Pro+](#)

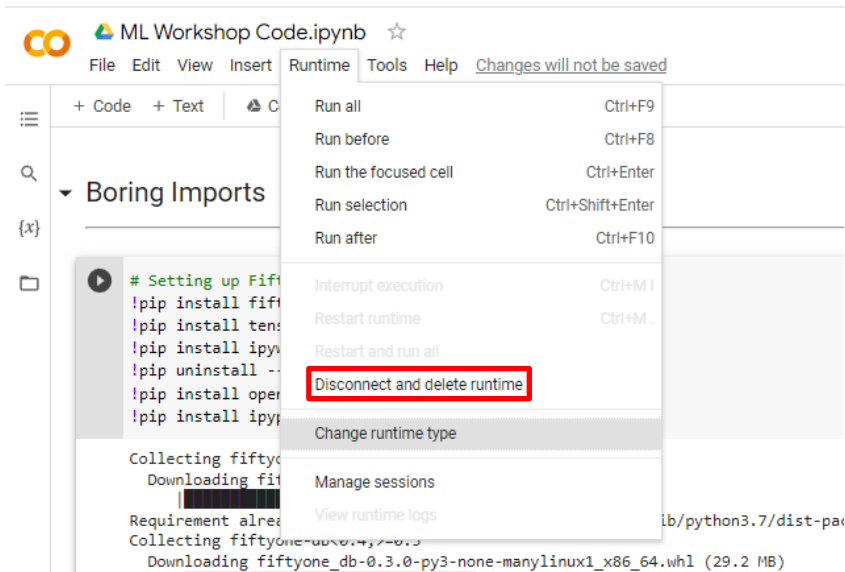
☐ Omit code cell output when saving this notebook

Cancel

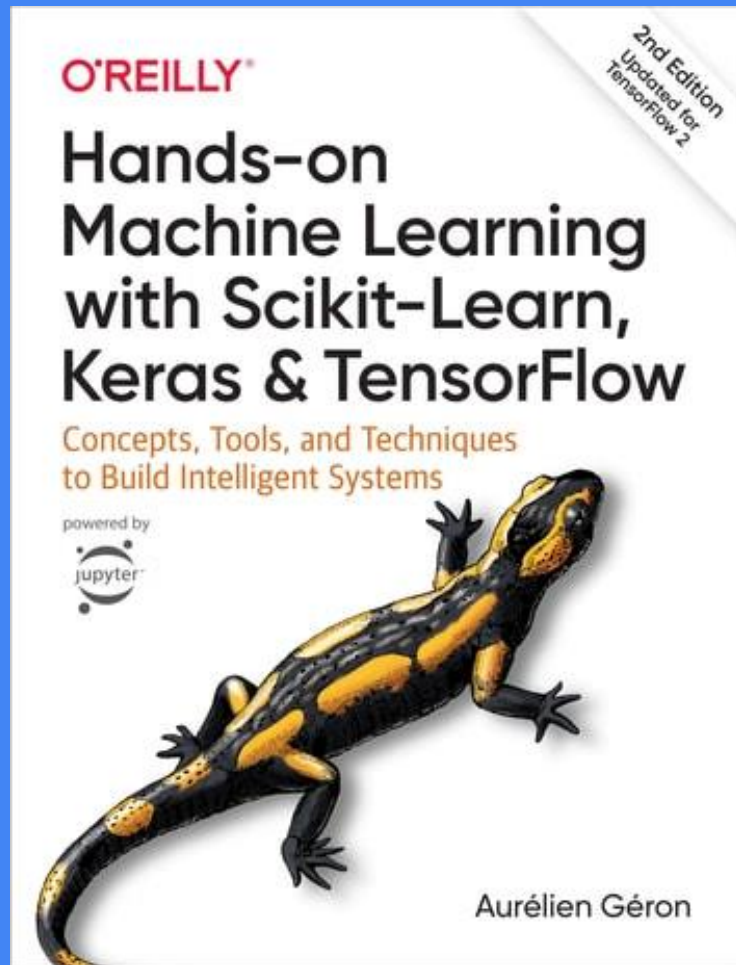
Save

The Colab Notebook

Step Three: resetting the notebook



Generally Good Practical Resource



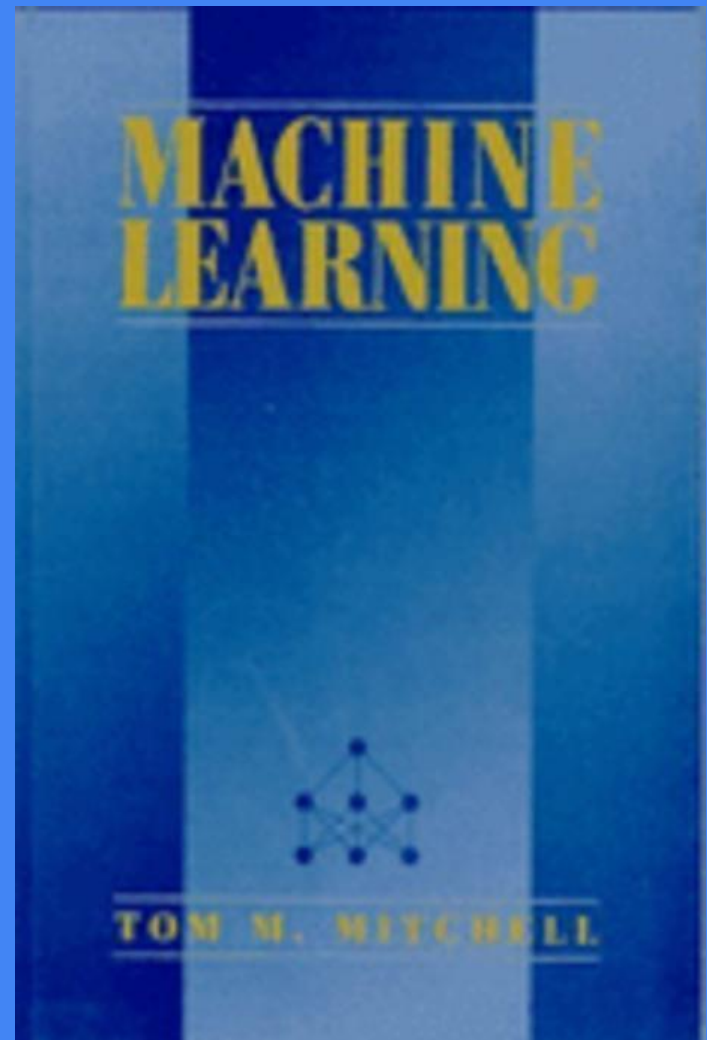
A Shameless Plug



Aurélien Geron

Speaker Event
25th May

Generally Good
Theoretical Resource



What this workshop will go through

Very practical focused

Impossible to teach the ins-and-outs of theory in a one hour workshop

Further resources will be provided for your own discretion

Don't worry about not understanding ALL the concepts or ALL the code!

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

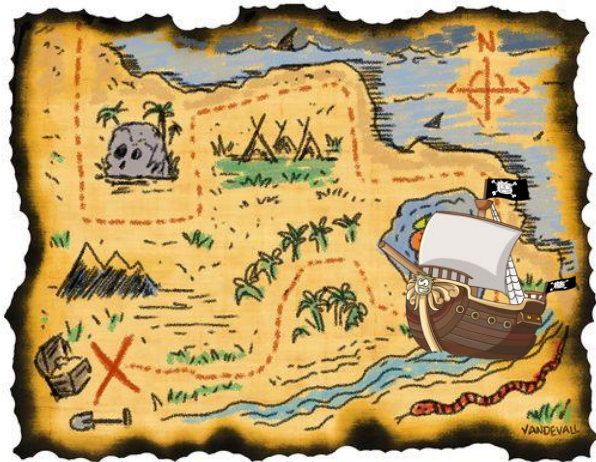
Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

What this workshop will cover:

The goal is to build something and have fun!



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

What this workshop will cover:

The goal is to build something and have fun!



What we intend to explore

Look at the big picture

Get the data

Discover and visualise the data to gain insights

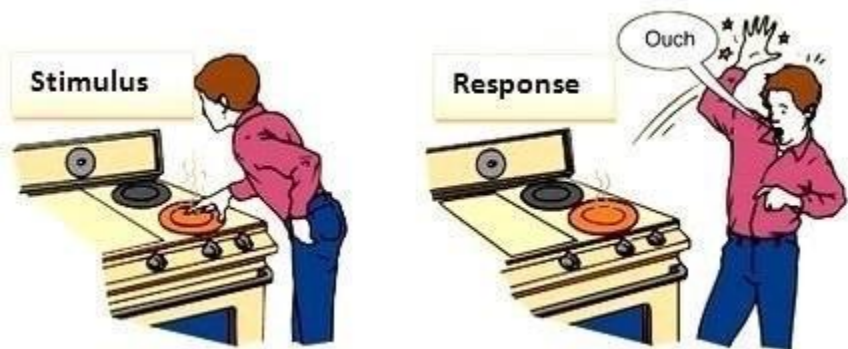
Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

What is machine learning?



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

What is machine learning?

"A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with experience E ."

- Mitchell, T. M. (1997). Machine learning. New York: McGraw-Hill.

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

When to use machine learning?

Have a view of what you're trying to achieve.

Good places to start*:

- Can the rules be coded?
- Can the task be scaled?

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

When to use machine learning? An example:



Life insurance:

- The rules can be coded
- The problem can scale

Solution: NOT a machine-learning algorithm!

Look at the big picture

Get the data

Discover and visualise the data to gain insights

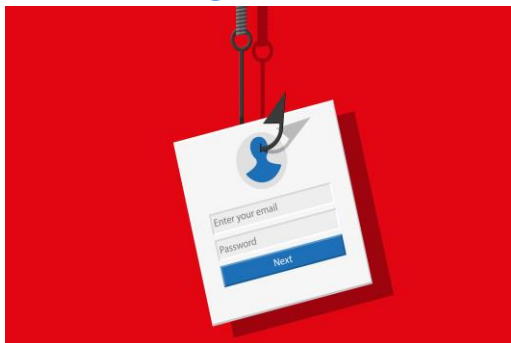
Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

When to use machine learning? An example:



Phishing emails

- The rules cannot be coded
- The problem can scale

Solution: A machine-learning algorithm!

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

The rules cannot be coded:

Suspicious sender

Poor English

And many more!

From: Netflix <rahma-cakupuyje-vakangenlaaywa@bihvgh.com>

Date: September 14, 2020 at 6:05:32 AM GMT+2

To: [REDACTED]

Subject: Re: Update Payment Subscription - We can't authorize payment September 13, 2020.

Order Number : 38443246

The Netflix logo is displayed in its characteristic red, bold, sans-serif font.

Update current billing information

Hi,

Unfortunately, we cannot authorize your payment for the next billing cycle of your subscription, Netflix was unable to receive a payment because the financial institution rejected the monthly charge.

TRY AGAIN PAYMENT

Obviously we'd love to have you back. if you change your mind, simply restart your membership and update your payment to enjoy all the best TV shows & movies without interruption.

- Netflix Team

Your task:

I want to find something from my room*



*Not my room

Quick and dirty checklist:

- The rules on searching for an item cannot be coded
- The problem can scale

Solution: A machine-learning algorithm!

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

What I'm trying to find



Get the data:

Highly dependent on the big picture identified in the previous step!

Let's say I want to find a coin



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

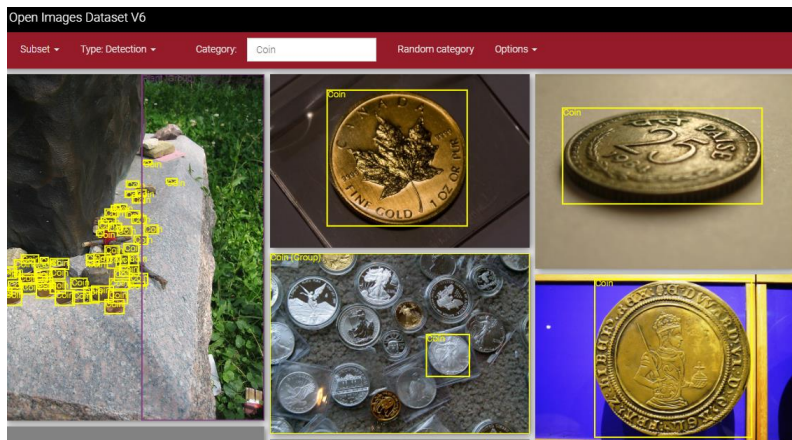
Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Get the data for the task:

We will be using Google Open Images as our dataset



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

FiftyOne

We will be using FiftyOne to handle importing the dataset



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Import FiftyOne:

```
[ ] import fiftyone as fo
import fiftyone.zoo as foz
import random

desired_class = "Coin"

dataset = foz.load_zoo_dataset(
    "open-images-v6",
    label_types=["detections"],
    classes=[desired_class.capitalize()],
    max_samples=250,
    shuffle=True,
    seed = 5,
    num_workers = 20,
)
```

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Viewing our desired_class

✓
9s



```
from fiftyone import ViewField as F
```

```
data_view = dataset.filter_labels("detections", (F("label") == desired_class) & (F("IsGroupOf") == False))
```

```
# Visualize the dataset in the FiftyOne App
```

```
session = fo.launch_app(view=data_view)
```



Viewing our desired_class

FiftyOne open-images-v6-250 Have a team?

Sample tags Label tags Labels Other fields Show

FilterLabels detections True False x

Filters 391 samples

SAMPLE TAGS

- ☐ train 221
- ☐ test 126
- ☐ validation 44

LABEL TAGS

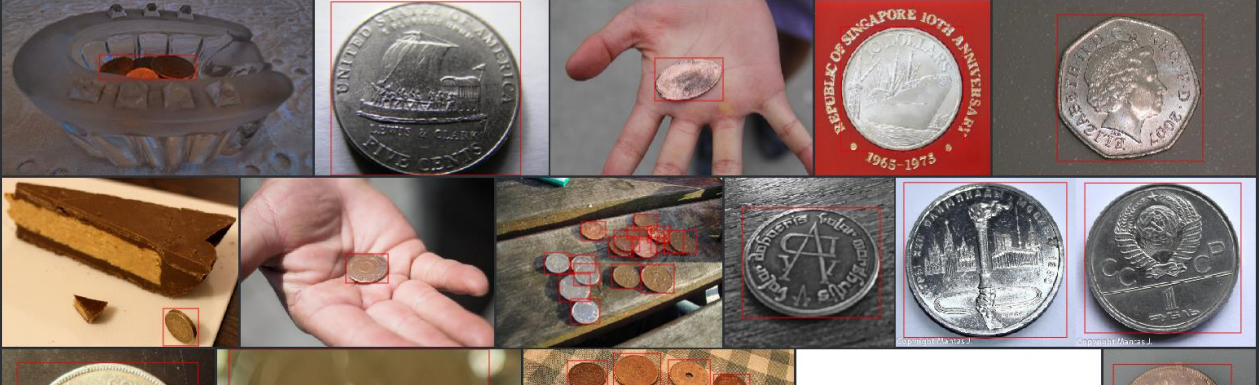
No label tags

LABELS 1 ✓

- ☒ detections 837

OTHER FIELDS

- ☐ id 391
- ☐ filepath 391
- ☐ open_images_id 391



Determining our model:

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

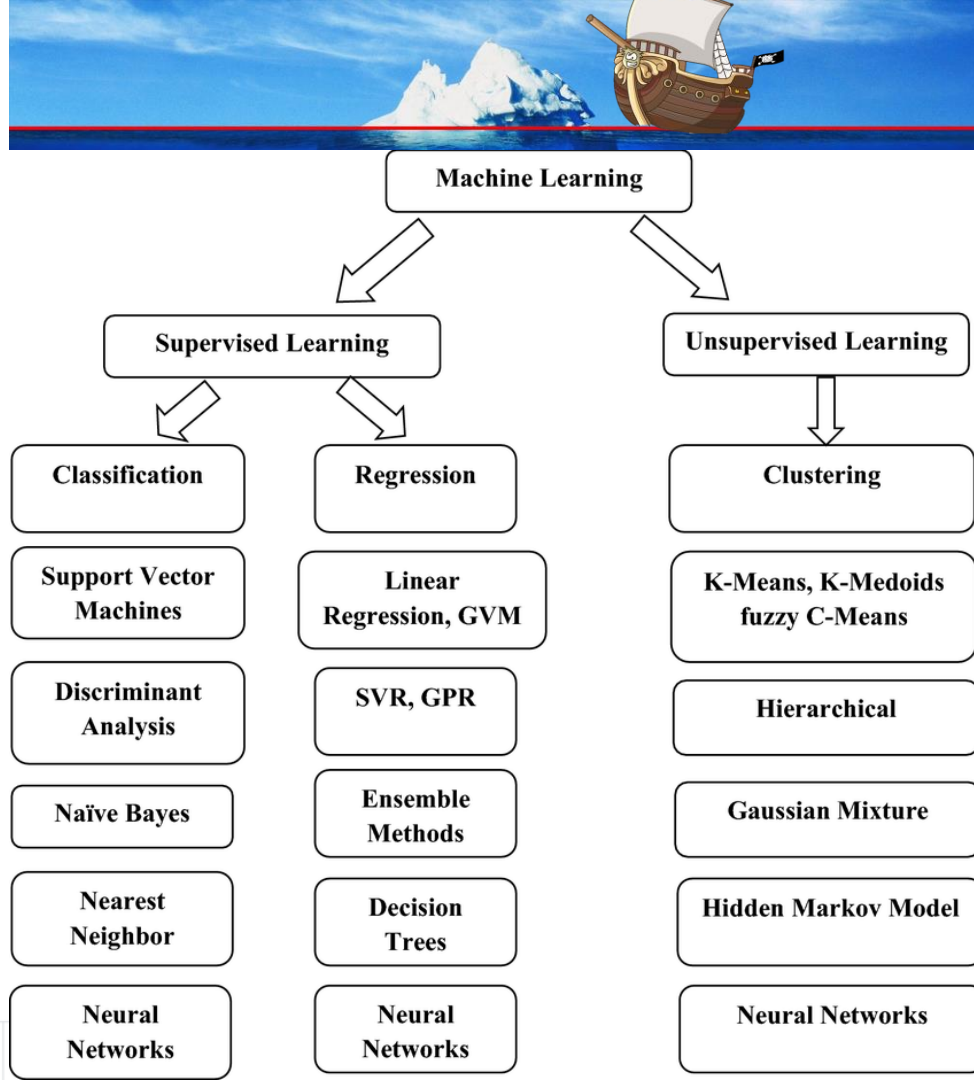
Launch, monitor and maintain your system

Determining which algorithm to use:



Google Search

- Understand for what types of data the algorithm should be used for
- Understand when to use said algorithm
- Worry about the “what” rather than the “how”.



Supervised Learning

An overly simple analogy



COIN



COIN



COIN

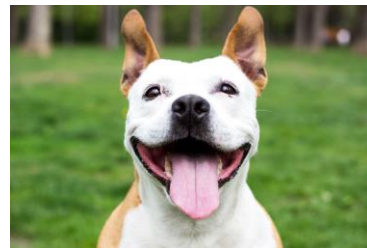


NO COIN

X A very large number

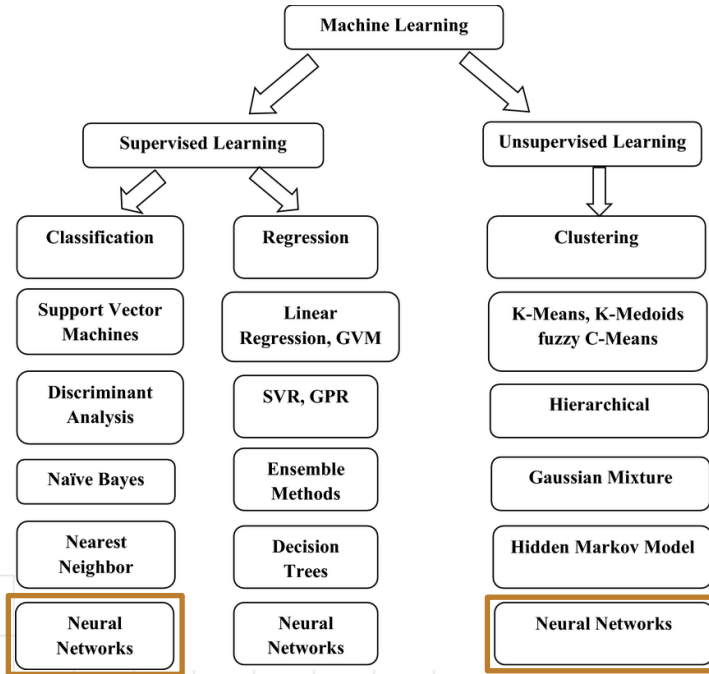
Unsupervised Learning

An overly simple analogy



X A very large number

Disclaimer



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Determining which algorithm to use:

A lot of resources to gather data, train and monitor an algorithm

The solution is thievery! (Stealing a pre-trained algorithm)



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Disclaimer

Content from here on out will be focused on **NEURAL NETWORKS** and will be specific to **YOLOV5**

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Determining which algorithm to use:



detect images in pictures

All

Images

News

Videos

Shopping

More

Tools

About 291,000,000 results (0.52 seconds)

[https://stackoverflow.com/questions/58441411-is-there-a-way-to-detect-pictures-in-images](https://stackoverflow.com/questions/58441411/is-there-a-way-to-detect-pictures-in-images)

[is there a way to detect pictures in images? - Stack Overflow](https://stackoverflow.com/questions/58441411/is-there-a-way-to-detect-pictures-in-images)

7/12/2019 — I want to know if there is a way to **detect pictures** in **images**. For example, the task is to recognize an apple and a **photo** of an apple in ...

2 answers · 0 votes: I don't think this could be done using only opencv. You should look in to a ...

How do I find an **image** contained within an **image**? - Stack ... 21 Oct 2011

How to recognize object in **photo** with flutter? - Stack Overflow 13 Jan 2021

How to **detect** a specific **picture** inside a video frame (or ... 30 May 2019

Is there any algorithm that allows me to identify if a **photo** was ... 9 Sept 2019

[More results from stackoverflow.com](#)



Developer Student Clubs

is there a way to detect pictures in images?

Asked 2 years, 4 months ago · Modified 4 months ago · Viewed 245 times



-2



I want to know if there is a way to detect pictures in images. For example, the task is to recognize an apple and a photo of an apple in given image. Is it possible with OpenCV or any other way?

opencv

image-processing

computer-vision

computer-science

object-recognition



Share Follow

asked Dec 7, 2019 at 4:04



Kundan

user

11

are you searching for some specific pictures or generally for any pictures? What kind of images does your application expect to see? – Micka Dec 8, 2019 at 22:30

Add a comment

2 Answers

Sorted by: Highest score (default)



0



I don't think this could be done using only opencv. You should look in to a deep learning approach. There are many models trained for object detection. eg:- YOLO model

Share Follow

answered Dec 8, 2019 at 17:40



chamith mawela

36

7

Our training algorithm



- YoloV5 is a supervised, neural network algorithm
- The algorithm is used to detect things in images
- The algorithm requires images and labels for inputs

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Installing YoloV5

▼ Installing YoloV5

```
✓ [4] # Clone YOLO-V5, change to its directory and install dependencies  
ss !git clone https://github.com/ultralytics/yolov5  
    %cd yolov5  
    !pip install -r requirements.txt
```

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Configuring the yaml file

```
✓ [5] # Constructing the yaml file
0s import yaml

yaml_file = dict(
    path = "../dataset/FiftyOne",
    train = "/content/dataset/train/images",
    val = "/content/dataset/validate/images",
    test = "/content/dataset/test/images",
    nc = 1,
    names = [desired_class],
)

with open(r'/content/yolov5/data/FiftyOne.yaml', 'w') as file:
    documents = yaml.dump(yaml_file, file)
    print(yaml_file)
```

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Creating the dataset directories

```
✓ 1s ▶ # Create dataset directories
!mkdir /content/dataset
!mkdir /content/dataset/train /content/dataset/train/images /content/dataset/train/labels
!mkdir /content/dataset/validate /content/dataset/validate/images /content/dataset/validate/labels
!mkdir /content/dataset/test /content/dataset/test/images /content/dataset/test/labels
```



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

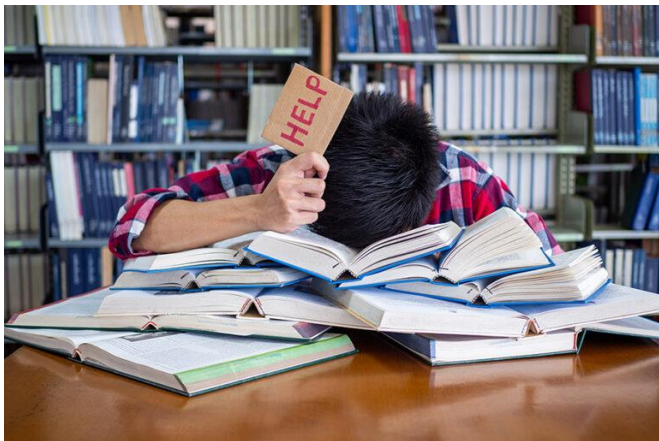
Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

The train, validate and test set

Train set:



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

The train, validate and test set

Validate set:



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

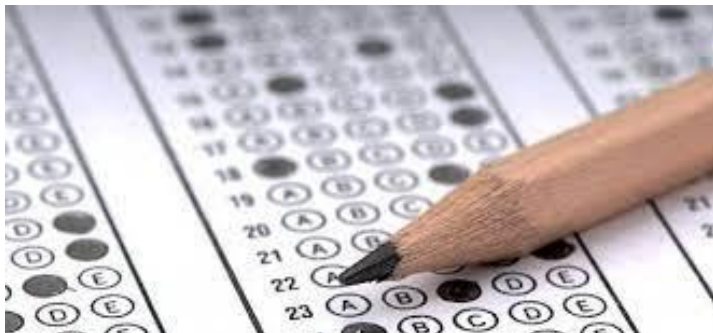
Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

The train, validate and test set

Test set:



Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Export the dataset into directories

```
[7] # Export dataset into select directories
import os
import shutil

export_dir = "/content"
dataset_type = fo.types.VOLOv5Dataset
label_field = "detections"

data_view.export(
    export_dir=export_dir,
    dataset_type=dataset_type,
    label_field=label_field,
)

for filename in os.listdir("/content/images/val"):
    random_value = random.randrange(10)
    if random_value == 0:
        filename = filename[:filename.index(".")]
        shutil.move("/content/images/val/{}.jpg".format(filename), "/content/dataset/test/images/{}.jpg".format(filename))
        shutil.move("/content/labels/val/{}.txt".format(filename), "/content/dataset/test/labels/{}.txt".format(filename))
    elif random_value == 1:
        filename = filename[:filename.index(".")]
        shutil.move("/content/images/val/{}.jpg".format(filename), "/content/dataset/validate/images/{}.jpg".format(filename))
        shutil.move("/content/labels/val/{}.txt".format(filename), "/content/dataset/validate/labels/{}.txt".format(filename))
    else:
        filename = filename[:filename.index(".")]
        shutil.move("/content/images/val/{}.jpg".format(filename), "/content/dataset/train/images/{}.jpg".format(filename))
        shutil.move("/content/labels/val/{}.txt".format(filename), "/content/dataset/train/labels/{}.txt".format(filename))
```

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Training the model

✓
4m

```
[8] # Train the model  
!python train.py --img 640 --batch 32 --epochs 10 --data /content/yolov5/data/FiftyOne.yaml --weights yolov5n.pt --single-cls
```



A batch

Our dataset

Training the model

✓
4m

```
[8] # Train the model  
!python train.py --img 640 --batch 32 --epochs 10 --data /content/yolov5/data/FiftyOne.yaml --weights yolov5n.pt --single-cls
```



Training the model

✓
4m

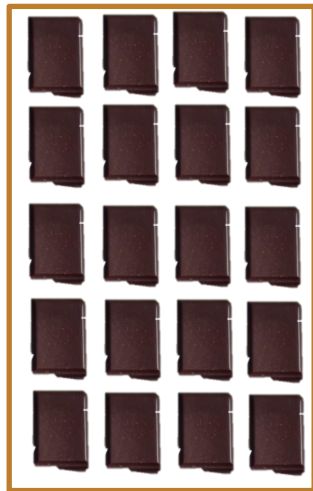
```
[8] # Train the model  
!python train.py --img 640 --batch 32 --epochs 10 --data /content/yolov5/data/FiftyOne.yaml --weights yolov5n.pt --single-cls
```



Training the model

✓
4m

```
[8] # Train the model  
!python train.py --img 640 --batch 32 --epochs 10 --data /content/yolov5/data/FiftyOne.yaml --weights yolov5n.pt --single-cls
```



X 10

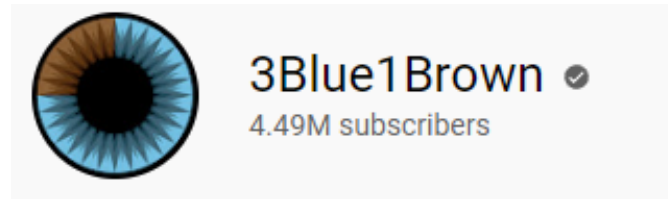
Utilise YoloV5's algorithm with pre-trained weights

```

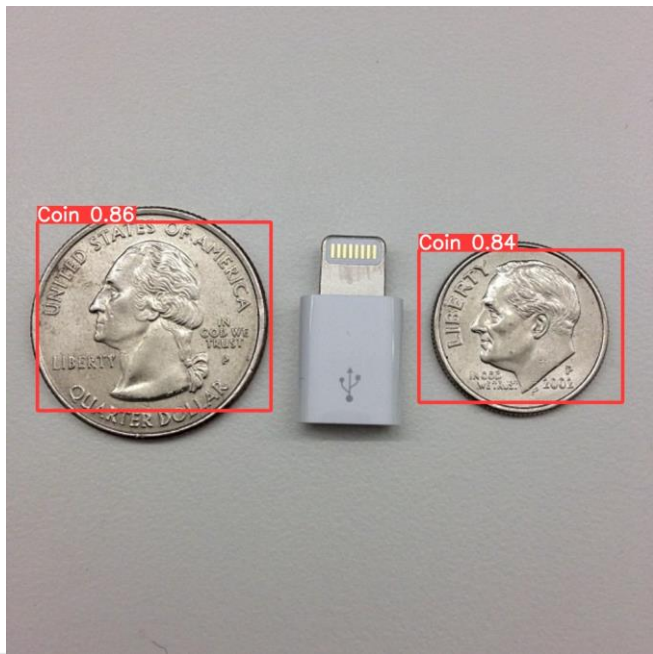
0      from n  params module arguments
1      -1 1    1760 models.common.Conv [3, 16, 6, 2, 2]
2      -1 1    4672 models.common.Conv [16, 32, 3, 2]
3      -1 1    4800 models.common.C3 [32, 32, 1]
4      -1 1    18560 models.common.Conv [32, 64, 3, 2]
5      -1 2    29184 models.common.C3 [64, 64, 2]
6      -1 1    73984 models.common.Conv [64, 128, 3, 2]
7      -1 3    156928 models.common.C3 [128, 128, 3]
8      -1 1    295424 models.common.Conv [128, 256, 3, 2]
9      -1 1    296448 models.common.C3 [256, 256, 1]
10     -1 1    164608 models.common.SPPF [256, 256, 5]
11     -1 1    33024 models.common.Conv [256, 128, 1, 1]
12     -1 1      0 torch.nn.modules.upsampling.Upsample [None, 2, 'nearest']
13     [-1, 6] 1      0 models.common.Concat [1]
14     -1 1    90880 models.common.C3 [256, 128, 1, False]
15     -1 1    8320 models.common.Conv [128, 64, 1, 1]
16     -1 1      0 torch.nn.modules.upsampling.Upsample [None, 2, 'nearest']
17     [-1, 4] 1      0 models.common.Concat [1]
18     -1 1    22912 models.common.C3 [128, 64, 1, False]
19     -1 1    36992 models.common.Conv [64, 64, 3, 2]
20     [-1, 14] 1      0 models.common.Concat [1]
21     -1 1    74496 models.common.C3 [128, 128, 1, False]
22     -1 1    147712 models.common.Conv [128, 128, 3, 2]
23     [-1, 10] 1      0 models.common.Concat [1]
24     -1 1    296448 models.common.C3 [256, 256, 1, False]
25     [17, 20, 23] 1    8118 models.yolo.Detect [1, [[10, 13, 16, 30, 33, 23], [30, 61, 62, 45, 59, 119], [116, 90, 156, 198, 373, 326]], [64, 128, 256]]
Model summary: 270 layers, 1765270 parameters, 1765270 gradients, 4.2 GFLOPs

```

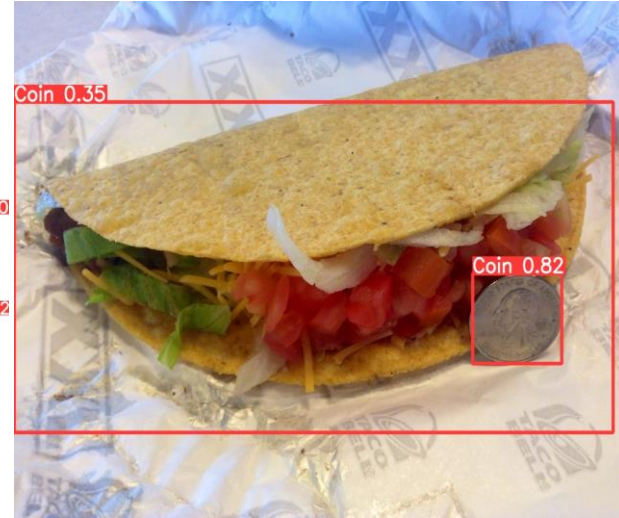
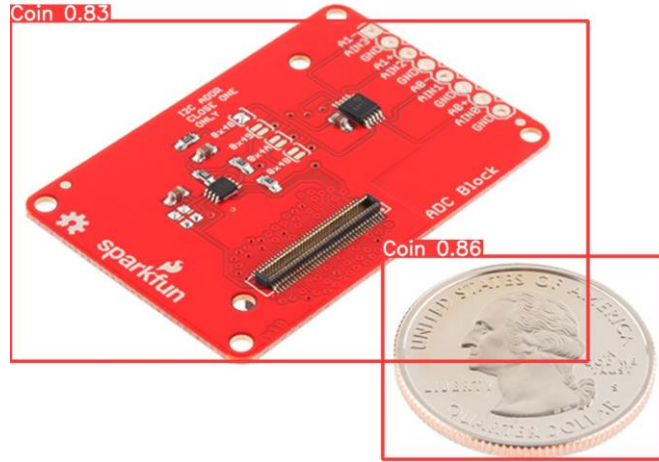
Additional Resources



Detecting coins!



Whoops!



Fine-tune your model

```
[ ] # View tensorboard
    %load_ext tensorboard
    %tensorboard --logdir runs/train
```

Look at the big picture

Get the data

Discover and visualise the data to gain insights

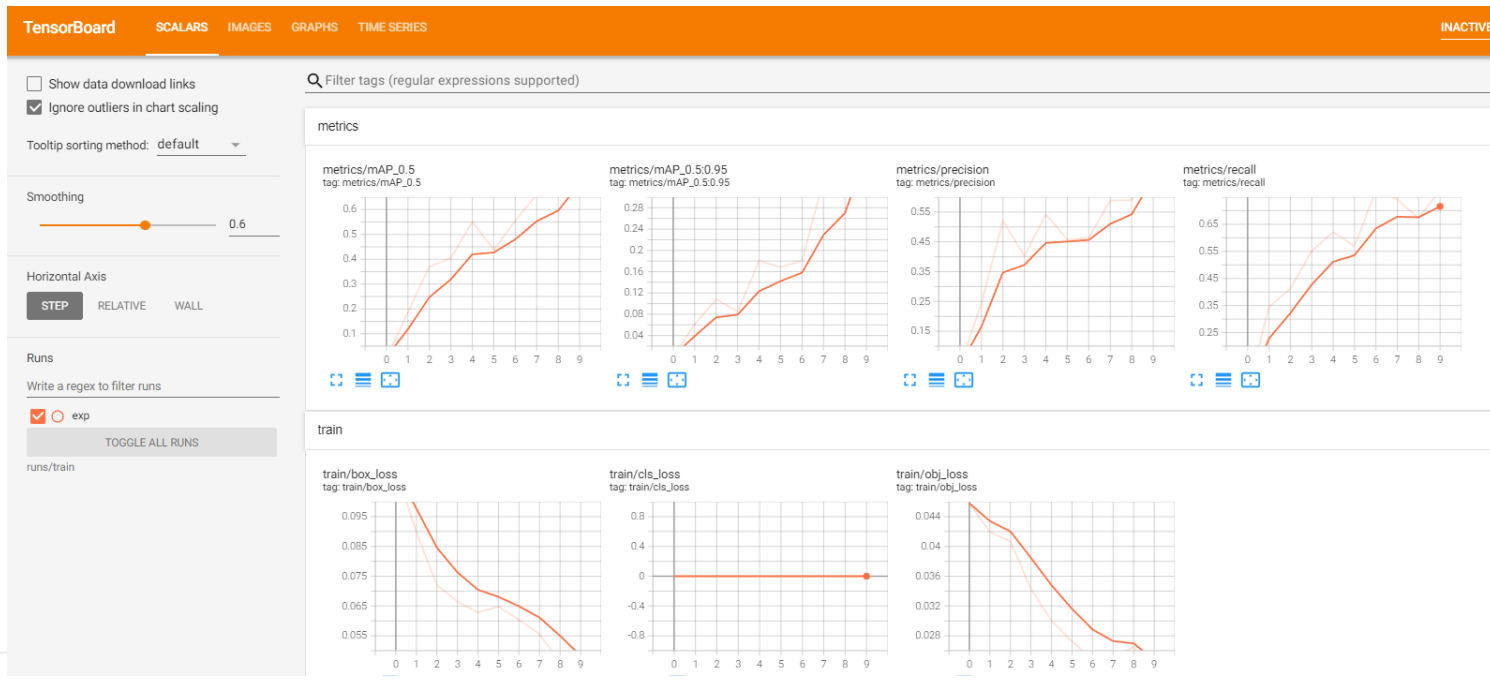
Prepare data for machine-learning algorithms

Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Fine-tune your model



Launch, monitor and maintain your system

Shifting more towards software design

Look at the big picture

Get the data

Discover and visualise the data to gain insights

Prepare data for machine-learning algorithms

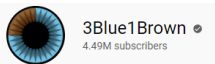
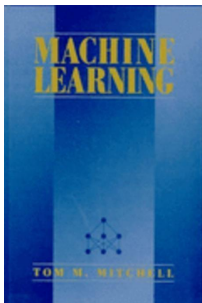
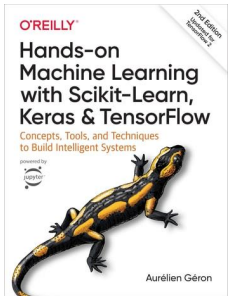
Select the model and train it

Fine-tune your model

Launch, monitor and maintain your system

Where next?

THEORY



PRACTICE

+ kaggle +

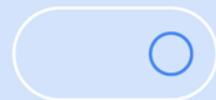
EXPLORATION

Google
StackExchange



Developer Student Clubs

Breaktime!



Thank you!

