THE UNIVERSITY OF MELBOURNE

Computer Vision COMP90086

Workshop Week 2

Thi Xuan May Le

thixuanmay.le@unimelb.edu.au

Tutorial with May

Wednesday, 10:00 AM at PAR-379-B1-117-Digital Learning Space (PC) (32)

Friday, 14:00 PM at PAR-379-B1-116-Digital Learning Space (PC) (32)

Email: **thixuanmay.le@unimelb.edu.au**

Contacting us

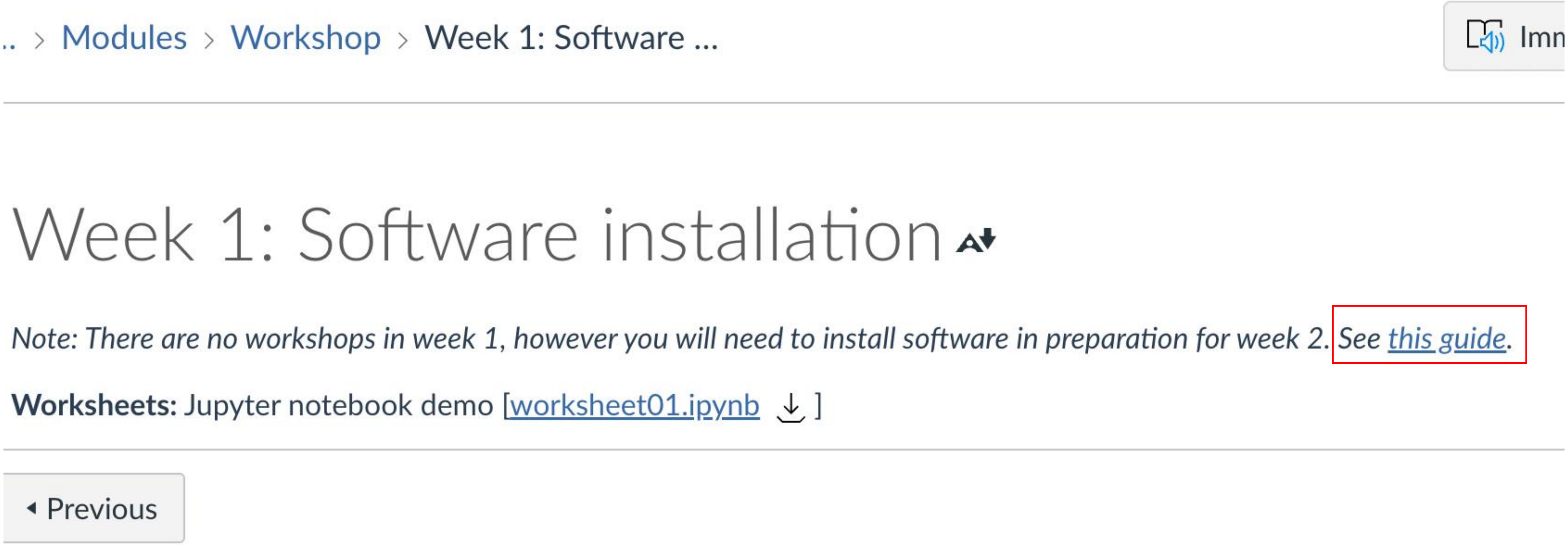
- General inquiries: Ed forum on LMS

"Remember class, there is no such thing as a, STUPID QUESTION"
*Teacher
CHALLENGE ACCEPTED

- Personal/private concerns: Email the instructors

Dr. Kris Ehinger kris.ehinger@unimelb.edu.au

Software installation



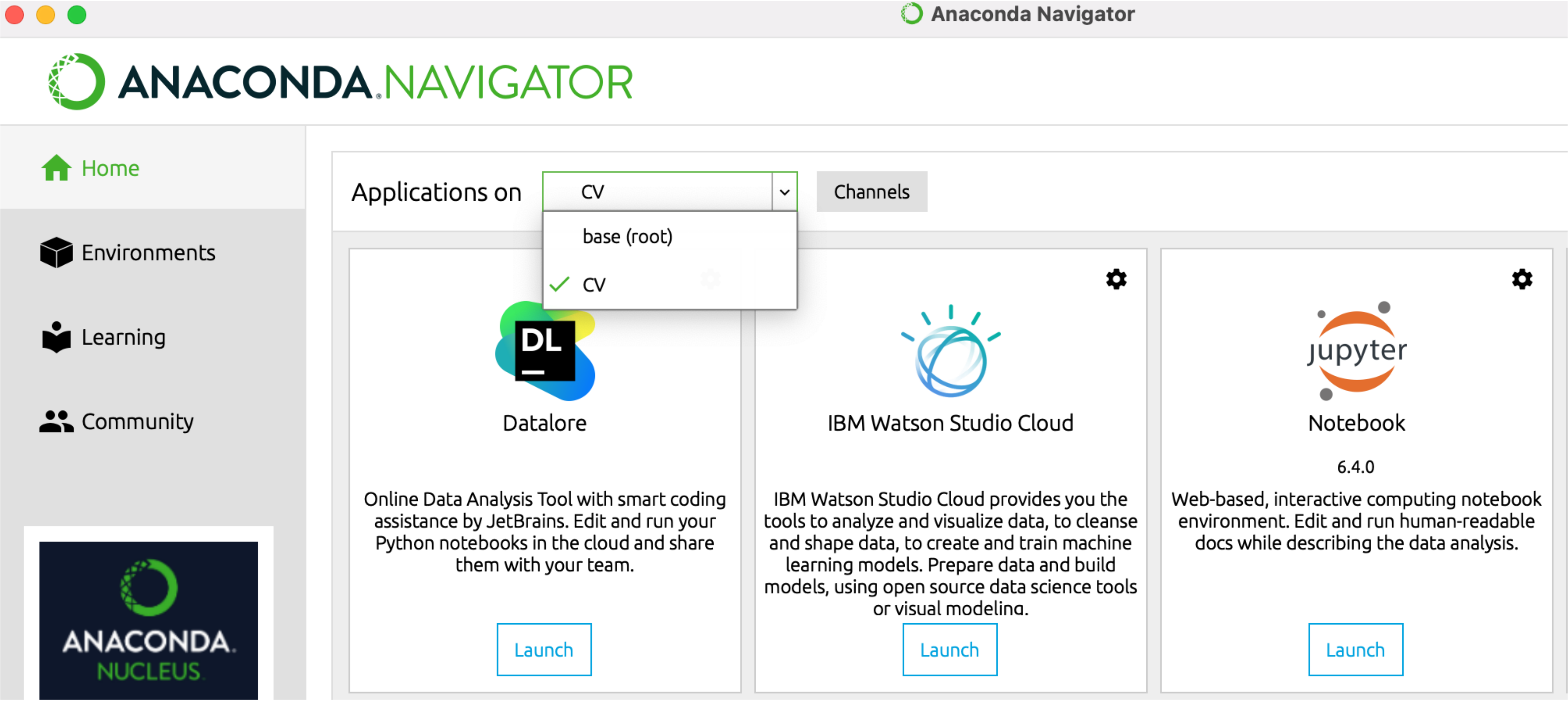
.. › Modules › Workshop › Week 1: Software ...

Week 1: Software installation

Note: There are no workshops in week 1, however you will need to install software in preparation for week 2. See this guide.

Worksheets: Jupyter notebook demo [worksheet01.ipynb]

Activate CV environment



Applications on CV

- base (root)
- ✓ CV

Datalore

Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.

IBM Watson Studio Cloud

IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.

Notebook

6.4.0

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

What is Numpy?

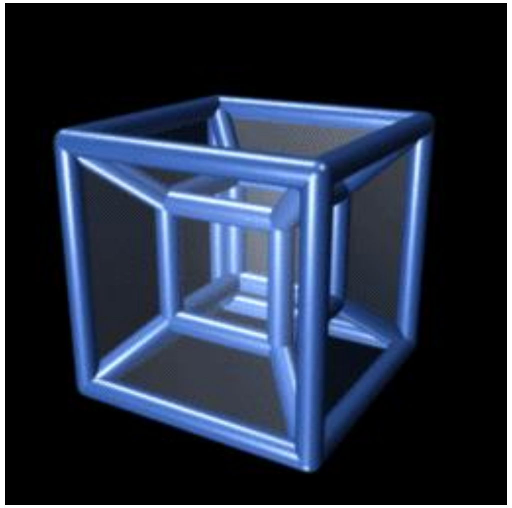


*Image: wikipedia

- A scientific computation package
- Offers many functions and utilities to work with N-Dimension arrays

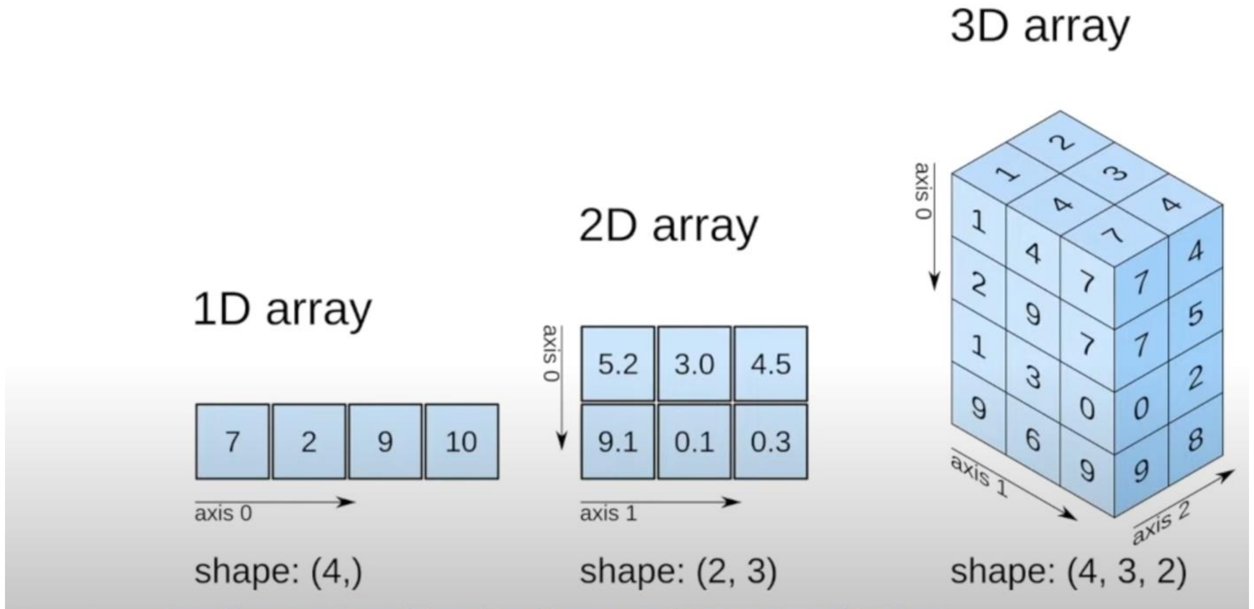
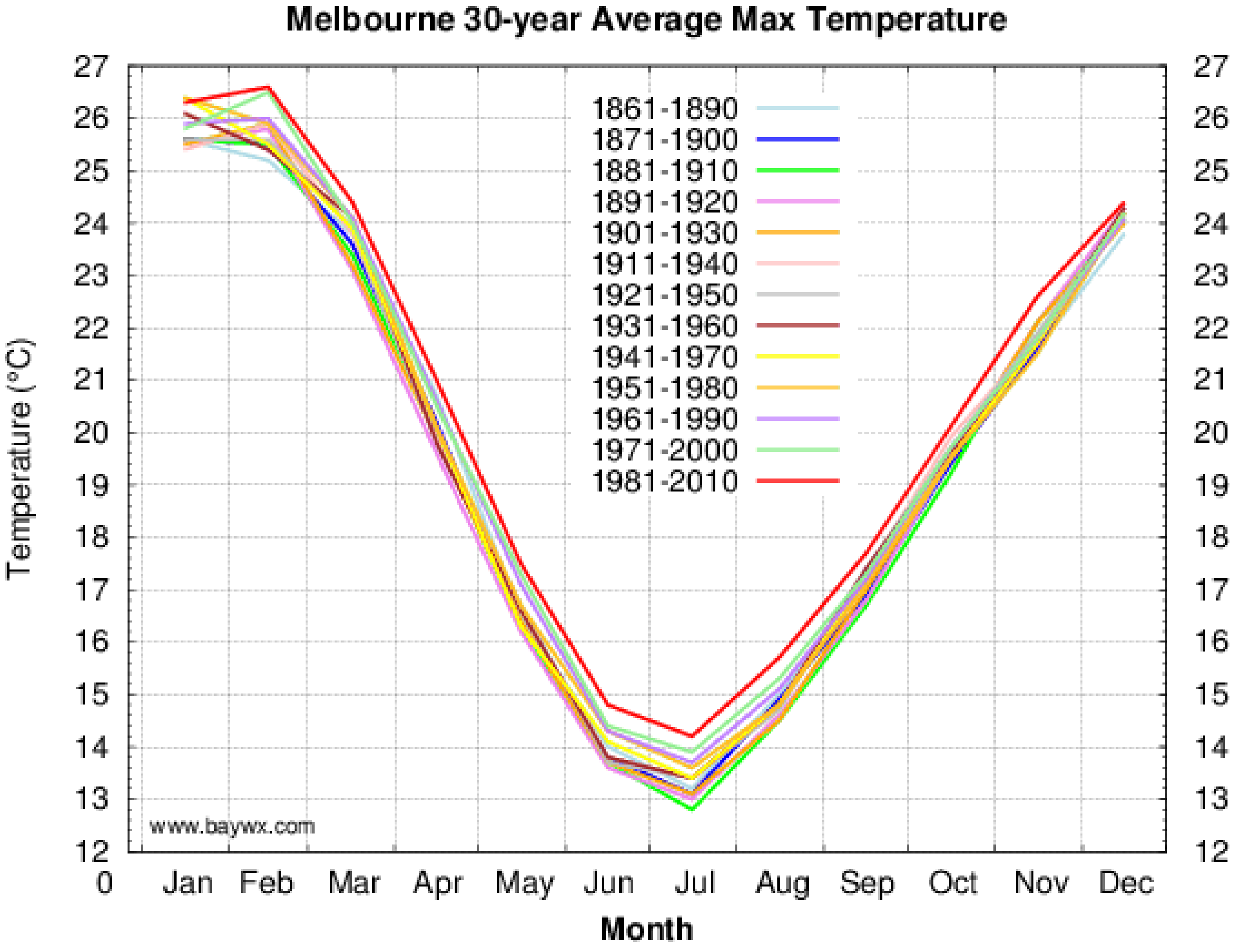


Image Source: https://www.oreilly.com/library/view/elegant-scipy/9781491922927/ch01.html

- Largely used by other libraries

Plot

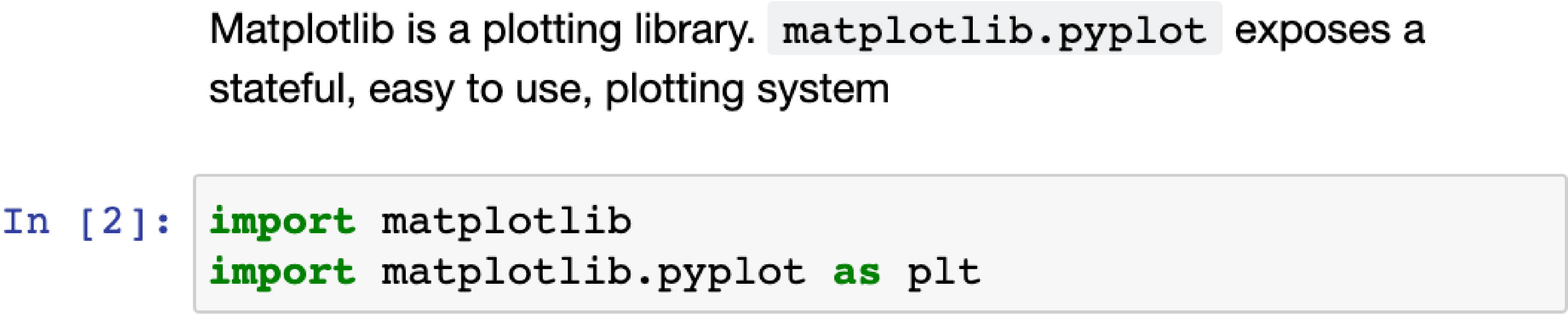
VectorStock®
VectorStock.com/21328676


Melbourne 30-year Average Max Temperature
Temperature (°C)
Month
27
26
25
24
23
22
21
20
19
18
17
16
15
14
13
12
0
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
1861-1890
1871-1900
1881-1910
1891-1920
1901-1930
1911-1940
1921-1950
1931-1960
1941-1970
1951-1980
1961-1990
1971-2000
1981-2010
www.baywx.com

Matplotlib

Matplotlib is an open-source plotting library able to produce high-quality graphs and charts.

It offers a large set of plot types (e.g., histogram, scatter, line, 3D and more), and uses NumPy arrays to handle data.



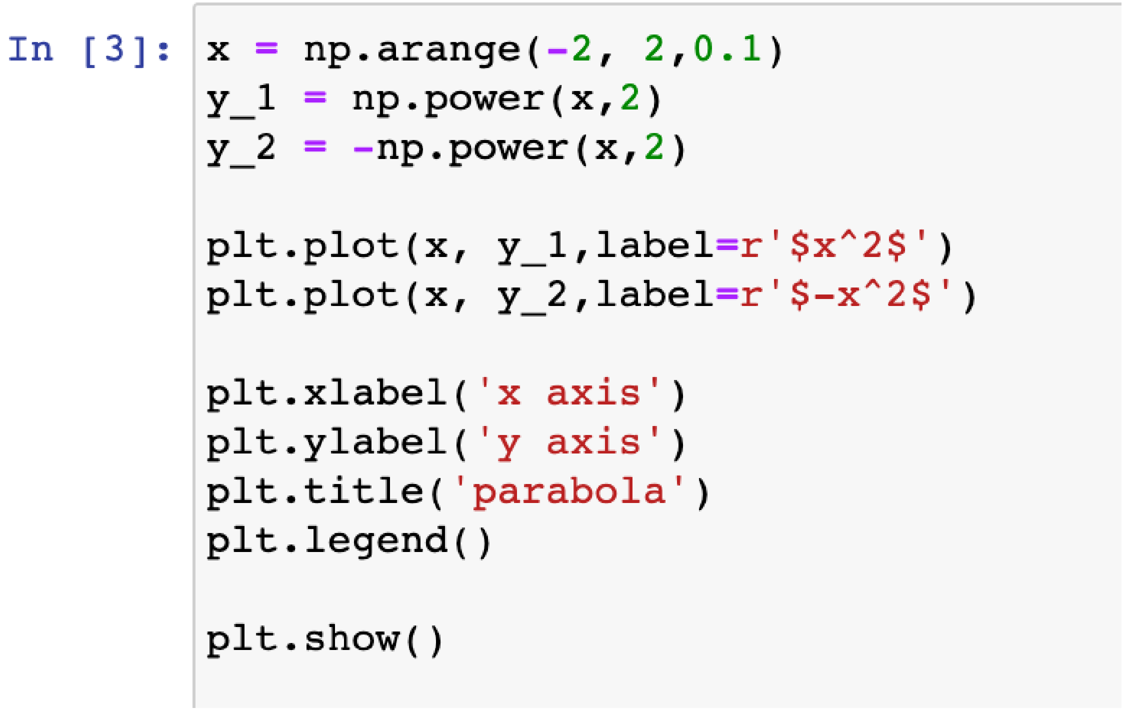
Matplotlib is a plotting library. `matplotlib.pyplot` exposes a stateful, easy to use, plotting system

In [2]:

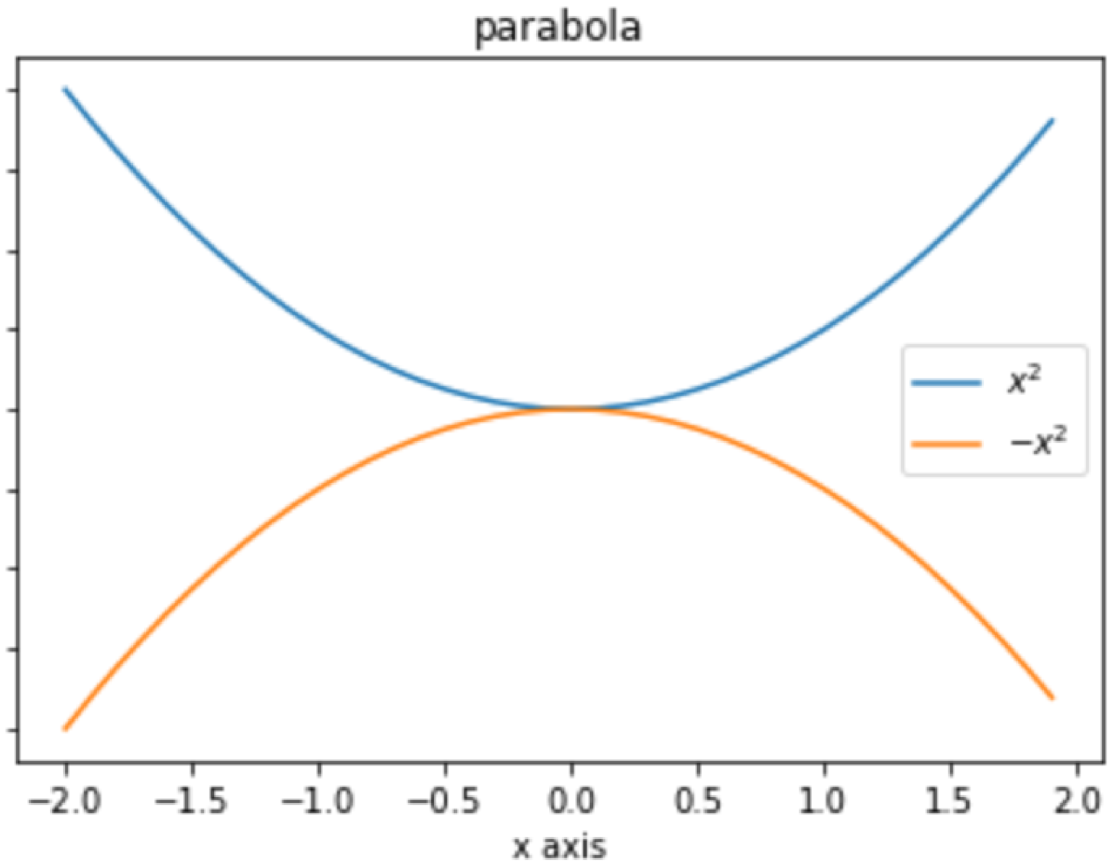
```
import matplotlib
import matplotlib.pyplot as plt
```

Matplotlib

Plotting



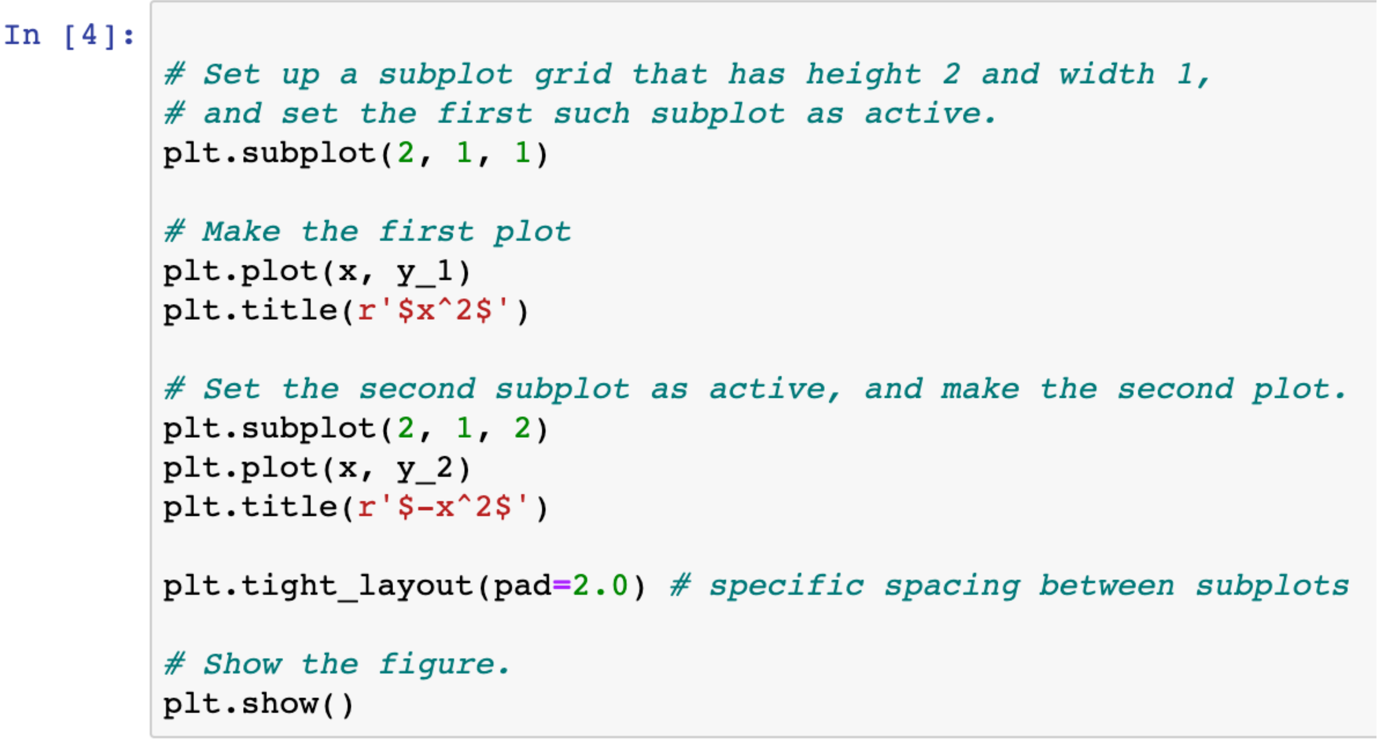
```
In [3]: x = np.arange(-2, 2,0.1)
        y_1 = np.power(x,2)
        y_2 = -np.power(x,2)

        plt.plot(x, y_1,label=r'$x^2$')
        plt.plot(x, y_2,label=r'$-x^2$')

        plt.xlabel('x axis')
        plt.ylabel('y axis')
        plt.title('parabola')
        plt.legend()

        plt.show()
```



Matplotlib

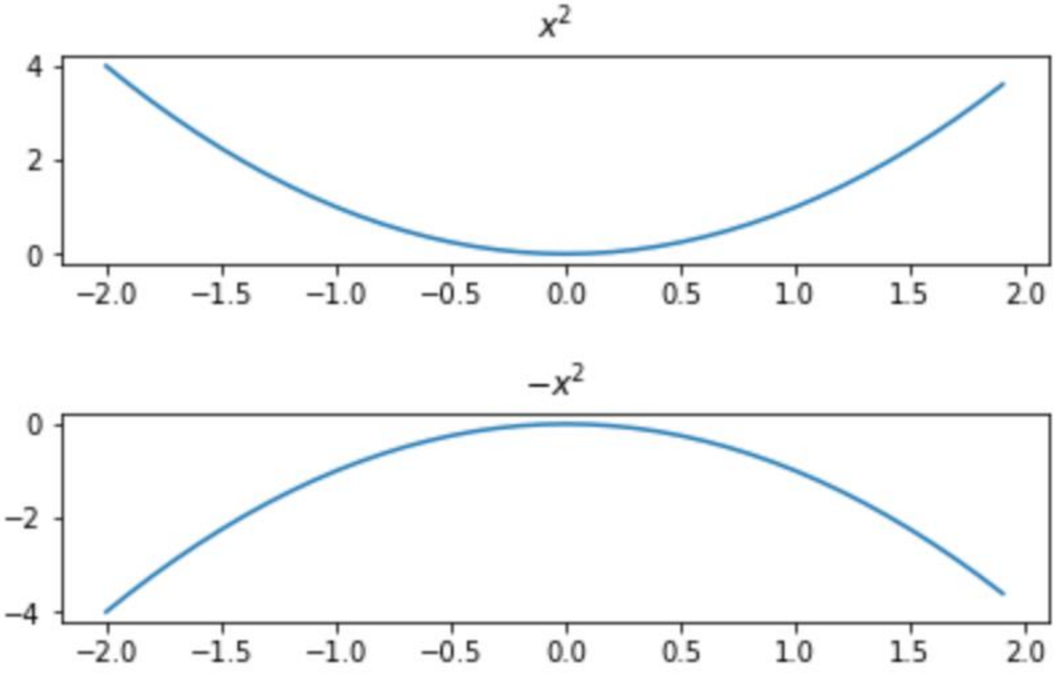
Subplots

put different plots in the same figure



In [4]:

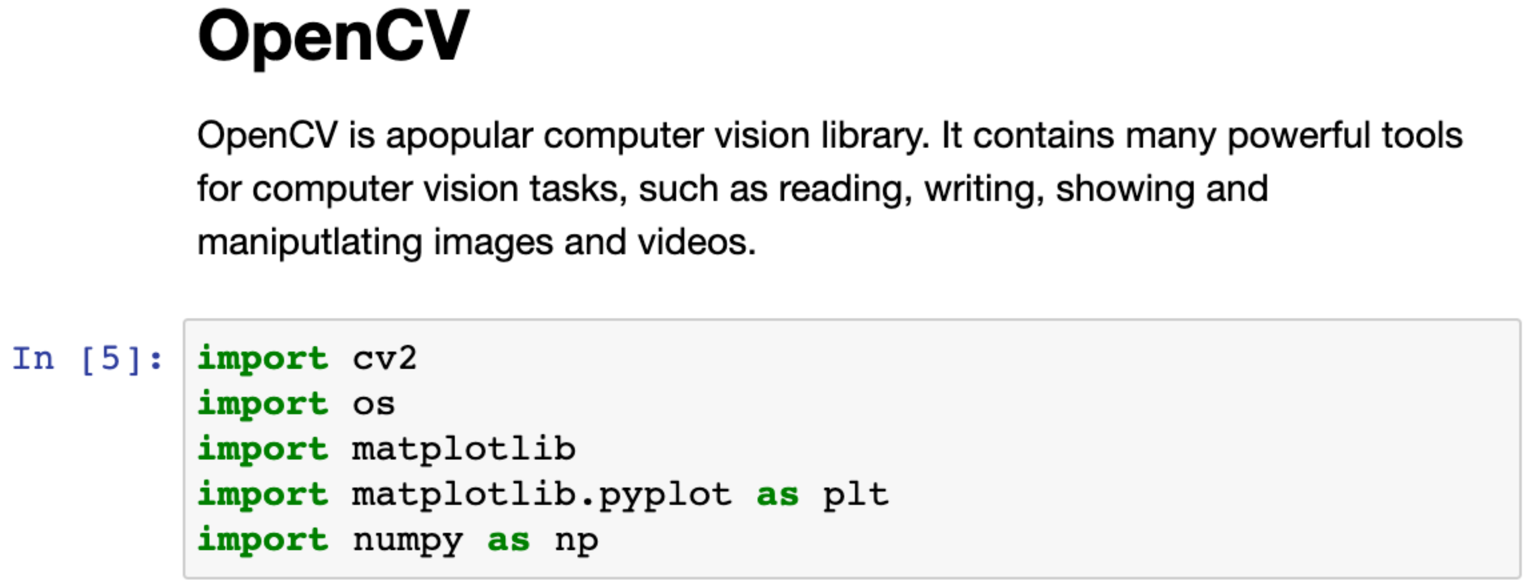
```
# Set up a subplot grid that has height 2 and width 1,
# and set the first such subplot as active.
plt.subplot(2, 1, 1)

# Make the first plot
plt.plot(x, y_1)
plt.title(r'$x^2$')

# Set the second subplot as active, and make the second plot.
plt.subplot(2, 1, 2)
plt.plot(x, y_2)
plt.title(r'$-x^2$')

plt.tight_layout(pad=2.0) # specific spacing between subplots

# Show the figure.
plt.show()
```



OpenCV

OpenCV is an open-source Computer Vision library. It allows to develop complex Computer Vision and Machine Learning applications fast, offering a wide set of functions.

In Python, OpenCV and NumPy are strictly related.



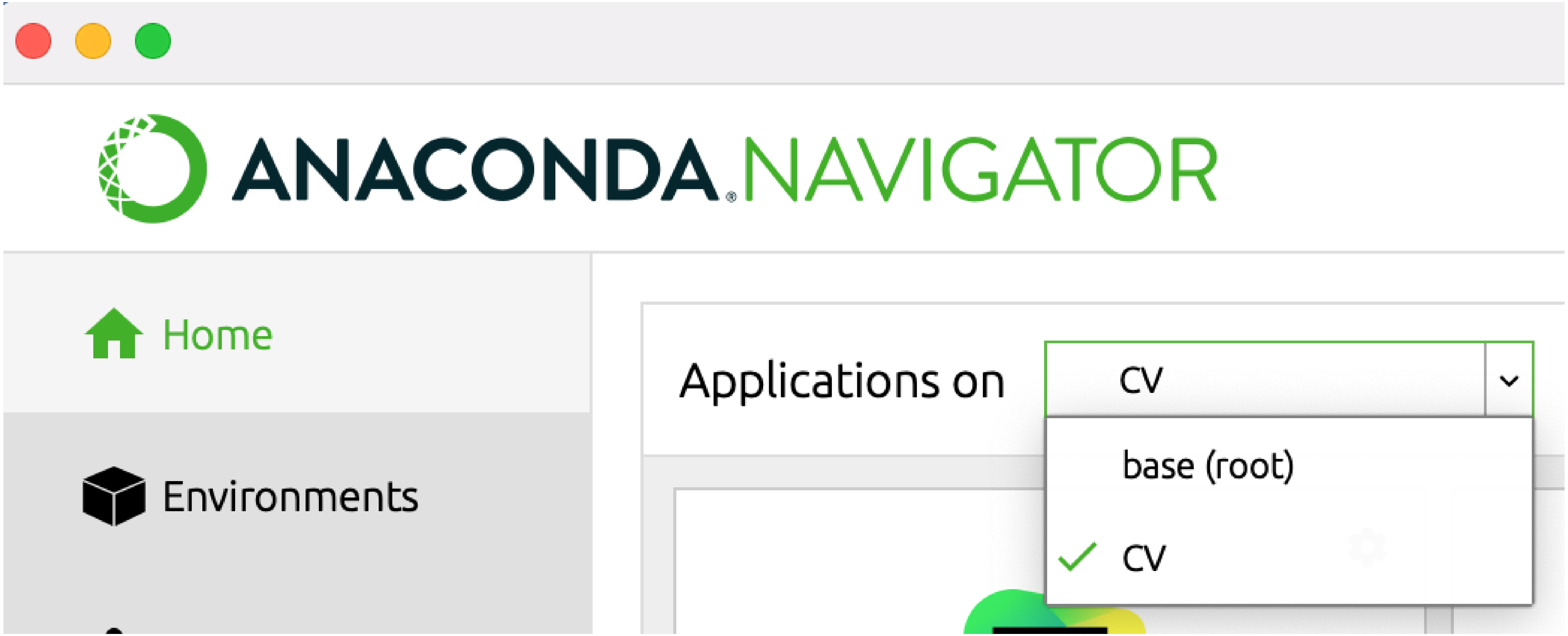
OpenCV

OpenCV is apopular computer vision library. It contains many powerful tools for computer vision tasks, such as reading, writing, showing and maniputlating images and videos.

In [5]:

```python
import cv2
import os
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
```

OpenCV



ANACONDA NAVIGATOR

Home

Environments

Applications on CV

base (root)

✓ CV

OpenCV

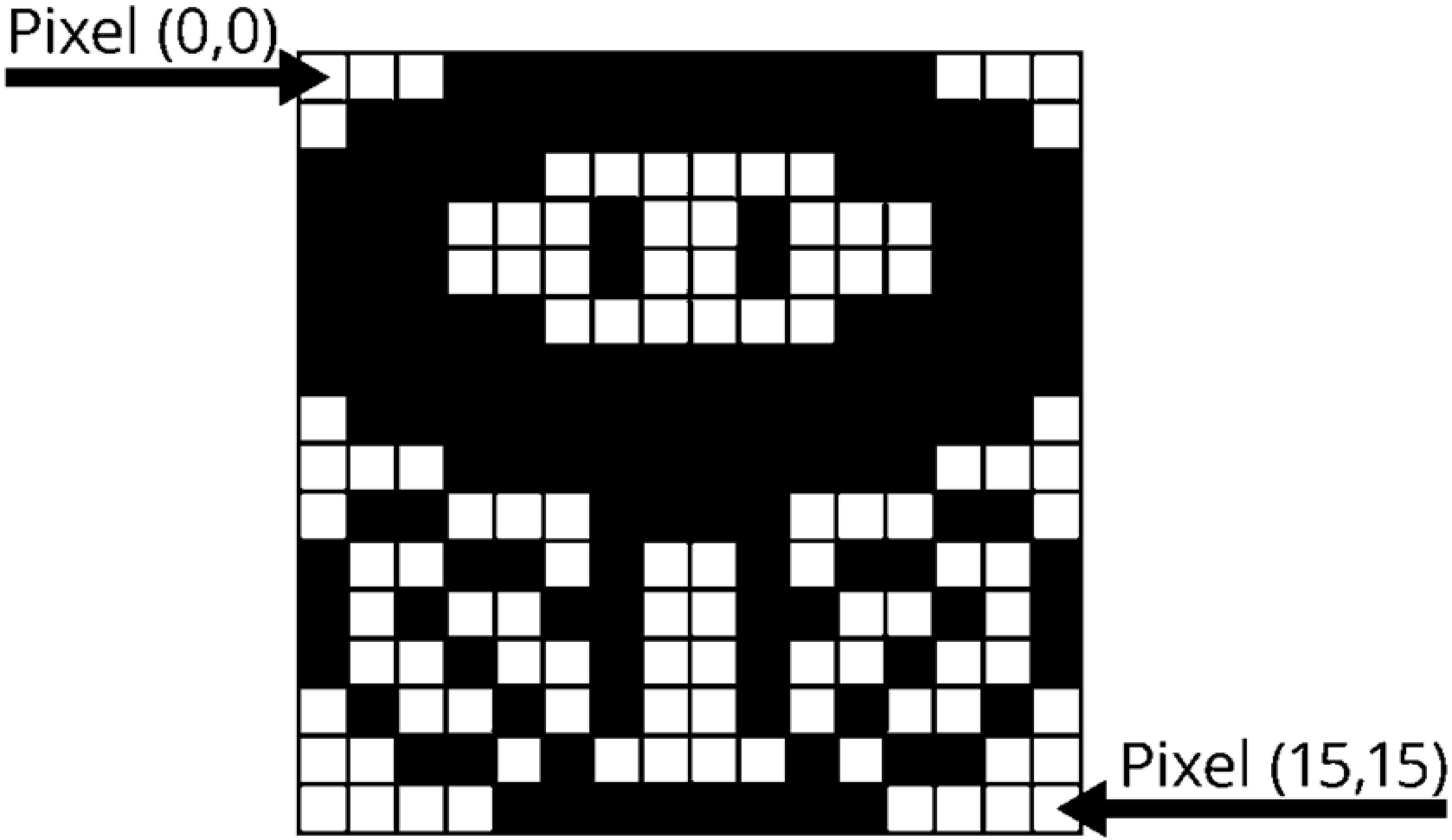


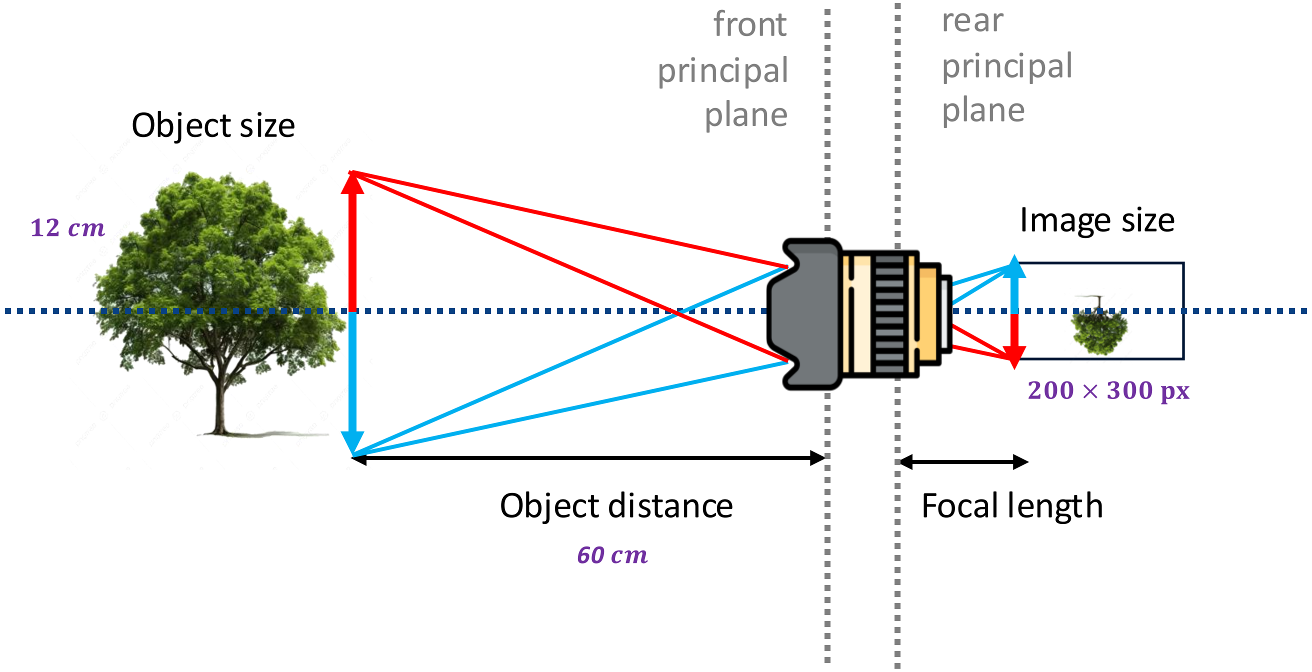
*image: https://www.oreilly.com/library/view/practical-computer-vision/9781449337865/ch04.html

A camera with a sensor size of 20 x 30 mm (which corresponds to a 200 x 300 pixel image) is aligned with a flat surface. An object which is 12 cm tall is placed at a distance of 60 cm from the camera to produce the image shown below. The top of the object is exactly aligned with the top edge of the image (the object is 100px high in the image). You can assume the optical centre of the camera is in the centre of the image. What is the focal length of the camera?

$$\frac{h_i}{f} = \frac{h_o}{d}$$

Where:

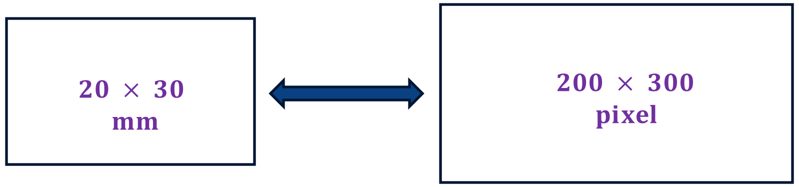
- $h_o = 120$ mm (real object height)
- $d = 60\ cm = 600mm$ (distance to object)
- $h_i =$ image height on sensor (measured)
- $d =$ focal length (distance from pinhole to sensor)



From the description:

Sensor height: 20 mm corresponds to 200 pixels.
So, **pixel size**: 20 mm / 200 pixels = 0.1 mm per pixel.
Image height of object in pixels: 100 pixels.
Image height on sensor: $h_i = 100 * 0.1 = 10$ mm

$$f = \frac{h_i \times d}{h_o} = \frac{10 \times 600}{120} = 50\ mm$$



Today's materials

Please access with UniMelb account.