

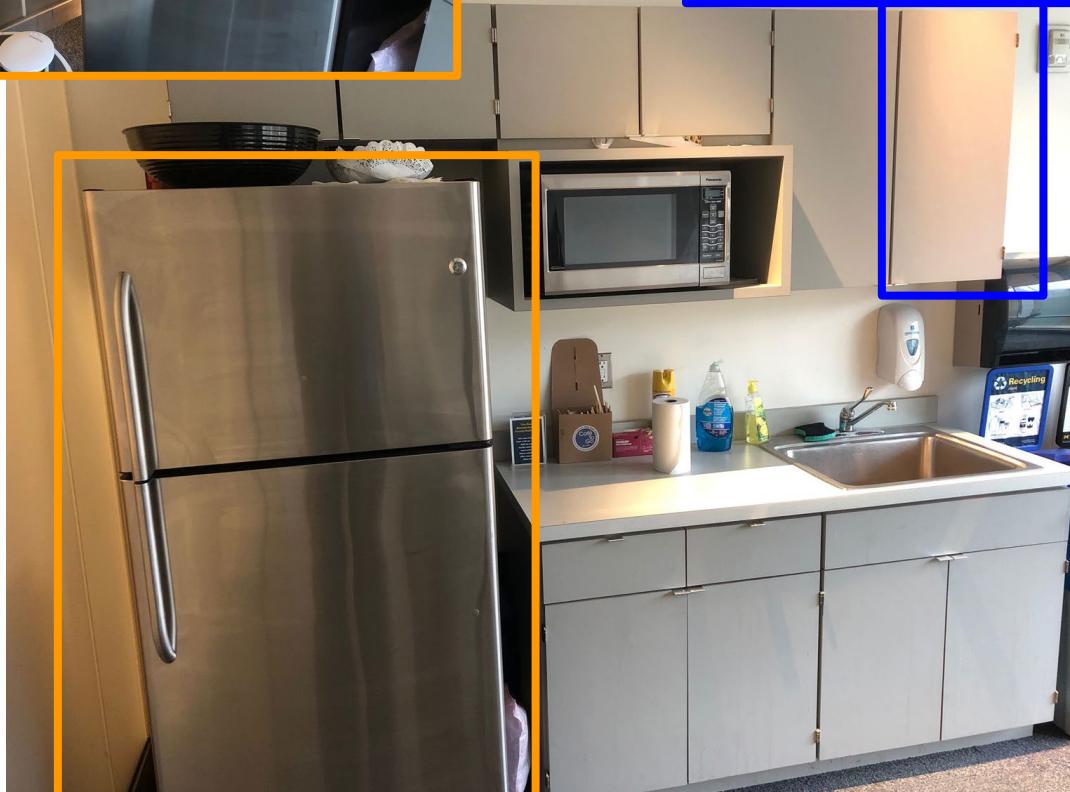
Machine Learning & Image Classification

ROB 102: Introduction to AI & Programming

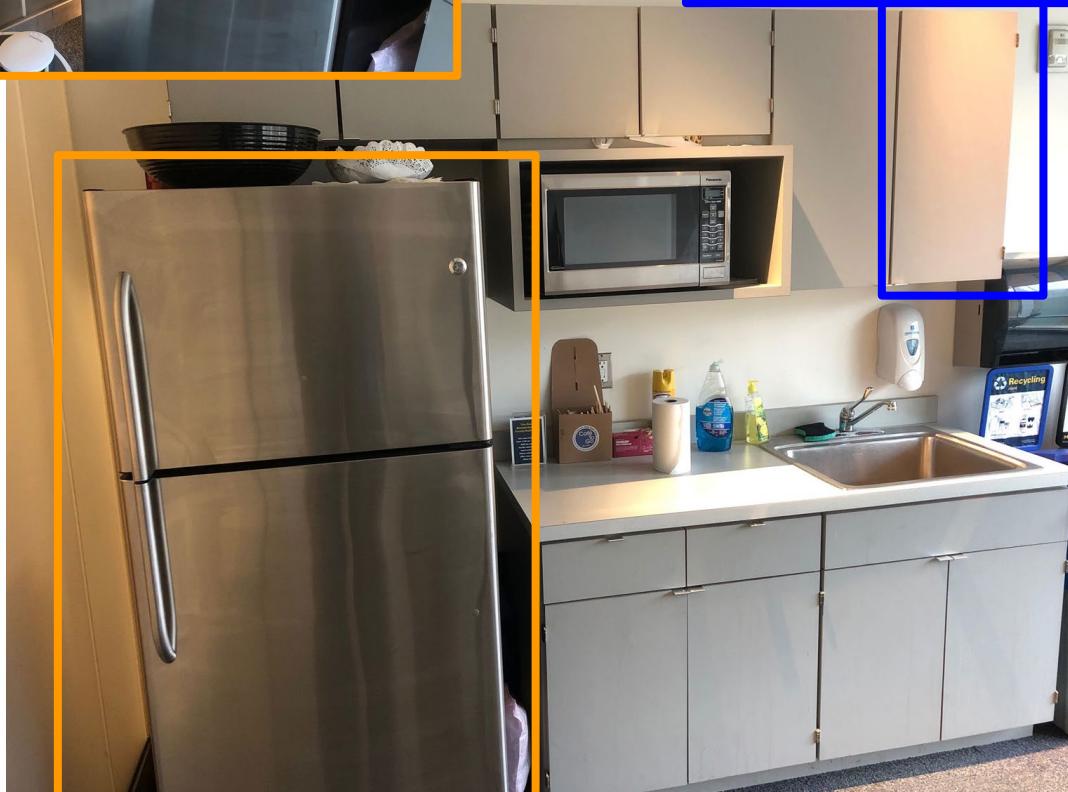
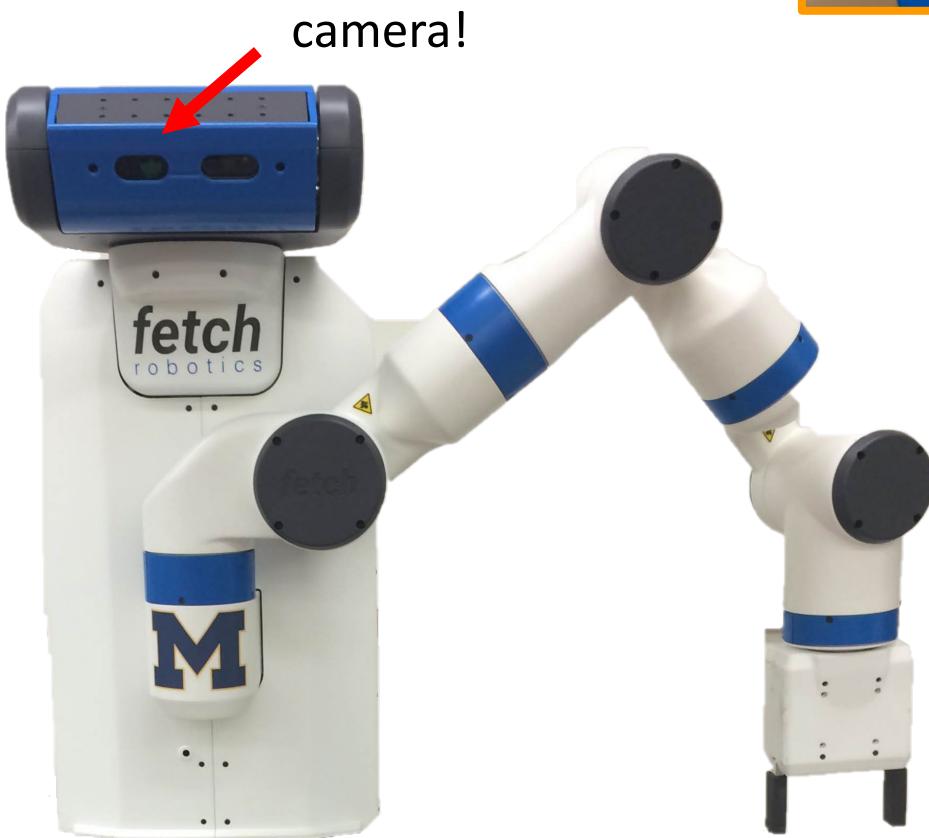
Lecture 10

2021/11/17

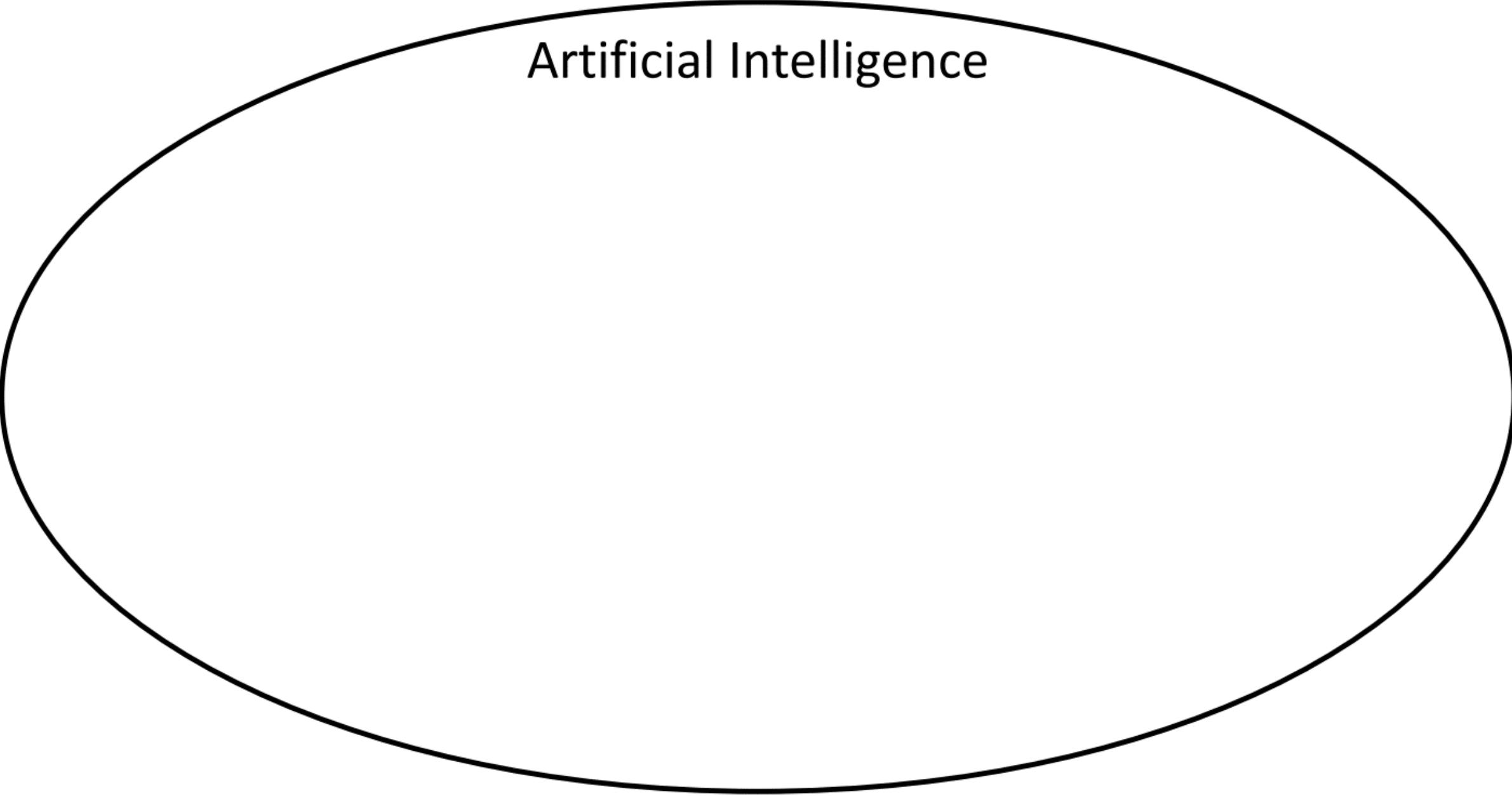
How can we make a robot see?



How can we make a robot see?



ROB 102 (Project 4): We will perform a **computer vision** task using **machine learning** algorithms.



Artificial Intelligence

Artificial Intelligence

Machine Learning

Computer
Vision

Artificial Intelligence

Computer
Vision

Machine Learning

Deep
Learning

ROB 102
Project 4

Artificial Intelligence

Robotics

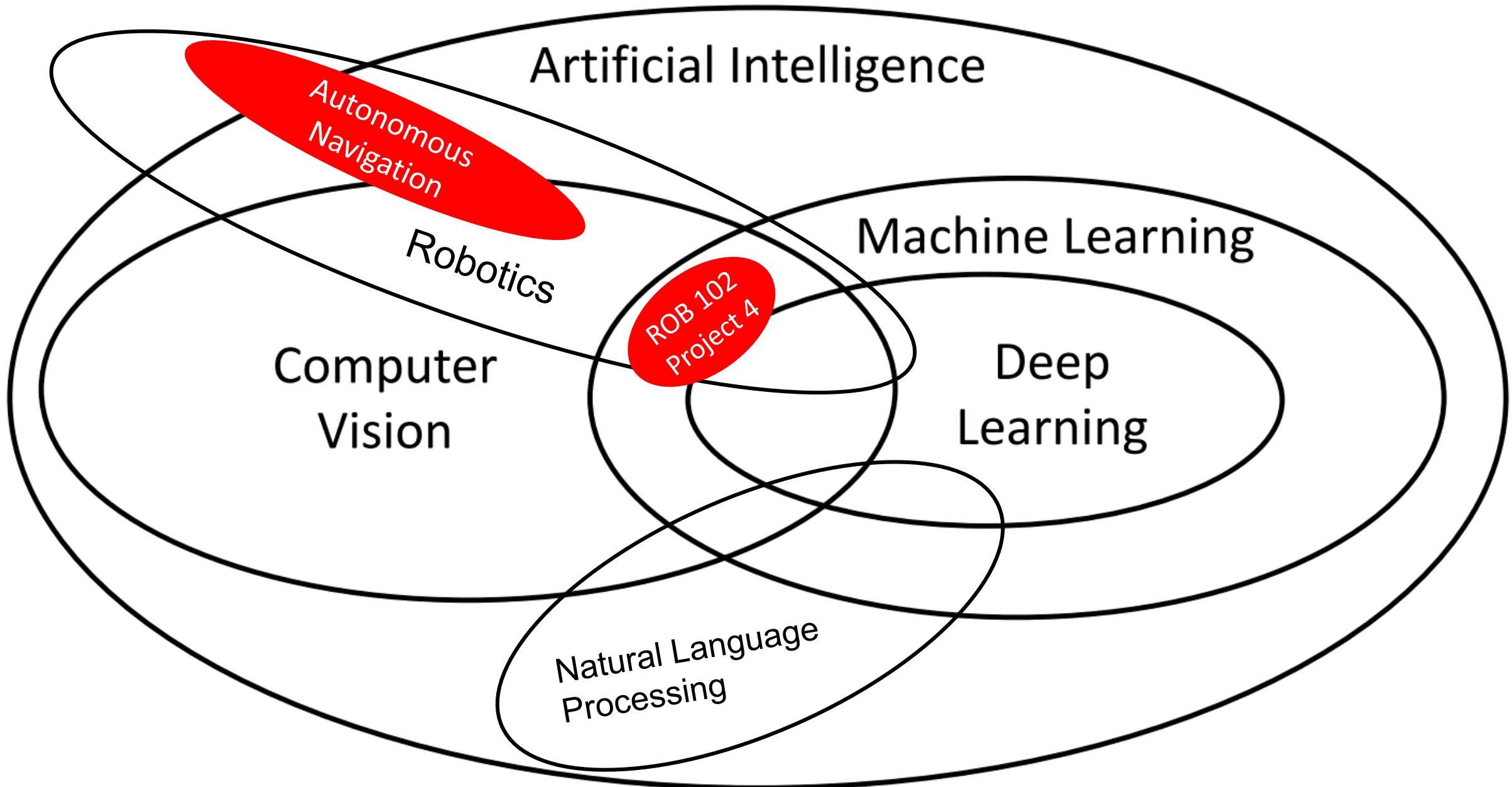
Computer
Vision

Machine Learning

Deep
Learning

Natural Language
Processing

ROB 102
Project 4



Goal for ROB 102 Project 4: Perform image classification with machine learning algorithms.

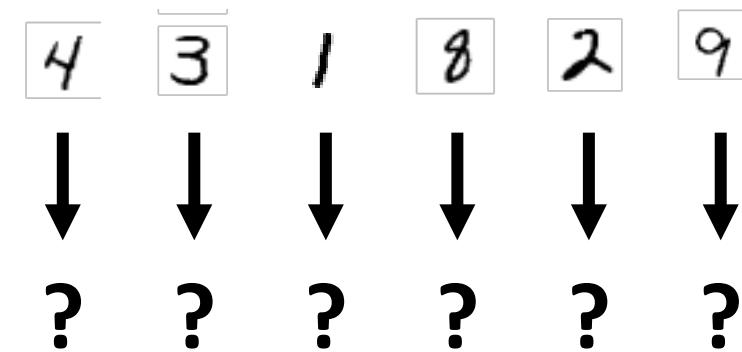
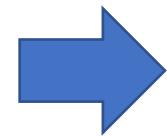


Image: CC4.0 ([link](#))

Machine Learning Terminology

Supervised
Learning

Unsupervised
Learning

Reinforcement
Learning

Supervised Learning

Given input data X and labels for that data y , learn a function to perform prediction on new data:

$$f(X) = y.$$

Supervised Learning

IMAGENET Large Scale Visual Recognition Challenge

The Image Classification Challenge:

1,000 object classes

1,431,167 images

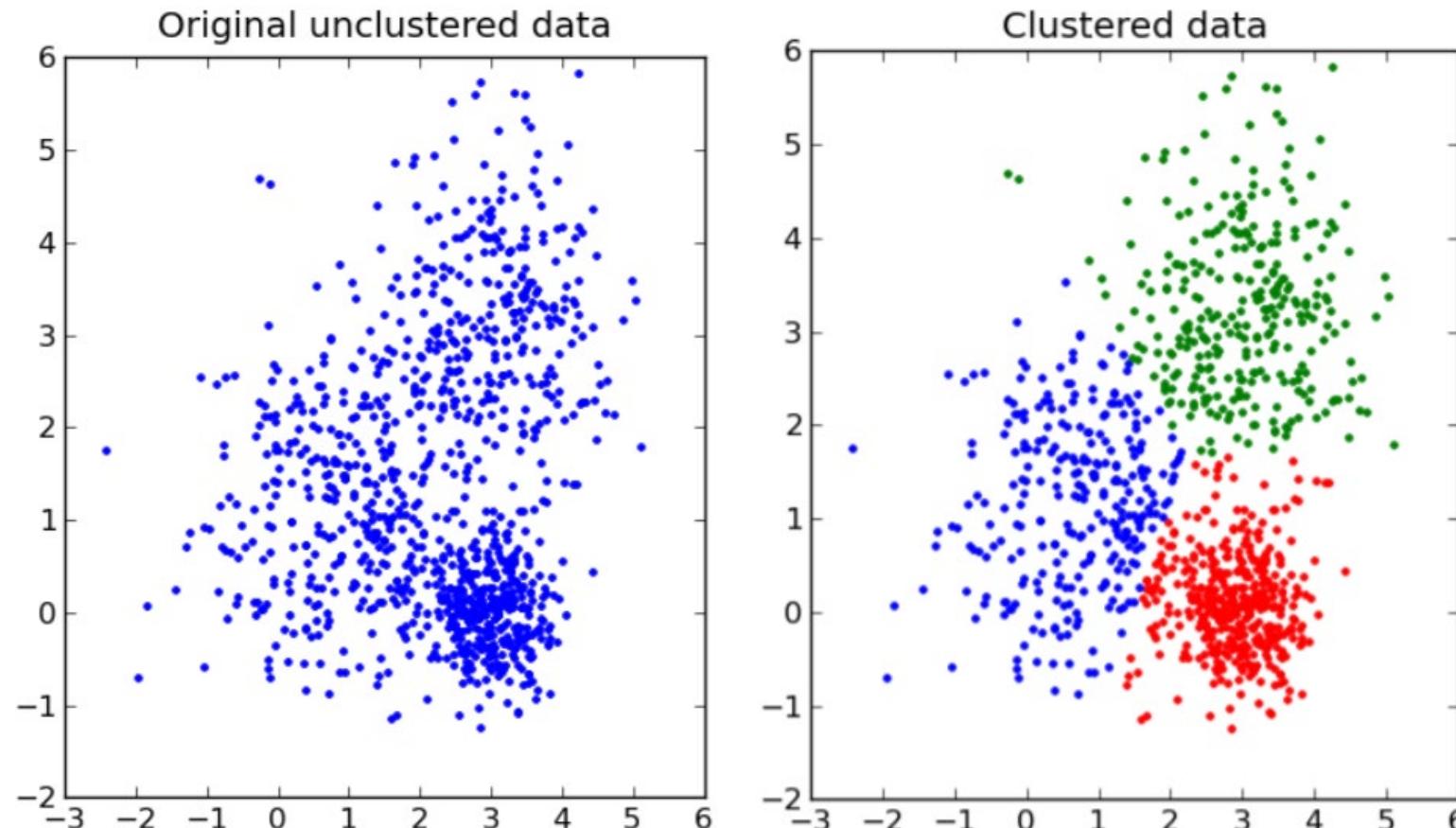


Output:
Scale
T-shirt
Steel drum
Drumstick
Mud turtle

Deng et al, 2009
Russakovsky et al. IJCV 2015

Unsupervised Learning

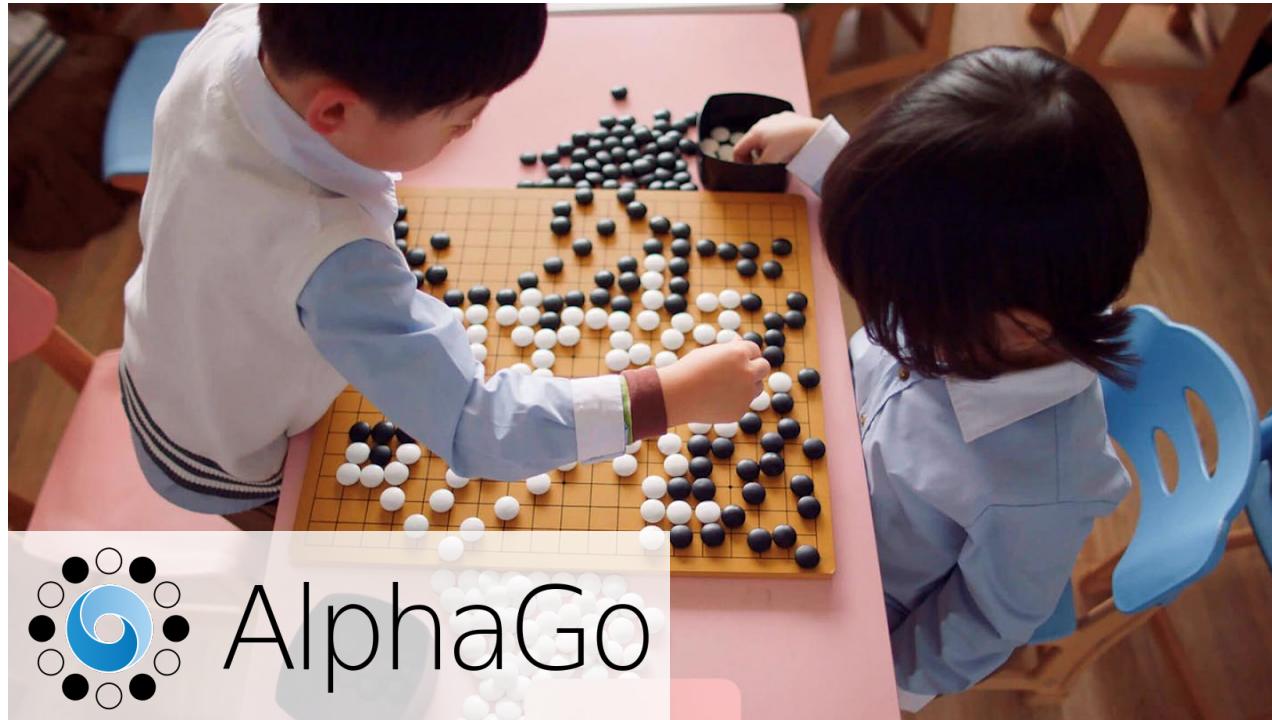
Given unlabeled data, learn a function over the input.



Source: Pineau, 2017 ([link](#))

Reinforcement Learning

An agent learns a policy that maximizes a reward.



Beat the world Go champion in 2017.

Reinforcement Learning

Outperforming humans at Atari games



Agent57: Outperforming the human Atari benchmark. [Puigdomènech, 2020]

Machine Learning Terminology

Supervised
Learning

Image Classification

Unsupervised
Learning

Reinforcement
Learning

Machine Learning Terminology

Supervised
Learning

Image Classification

ROB 102

Unsupervised
Learning

Reinforcement
Learning

What is in this picture?



Image: Alastair Pollock Photography/Getty Images

“shark”

What is in this picture?



Image: Alastair Pollock Photography/Getty Images



Image: Benjamin Lowy/The New York Times

“shark”

“shark”

What is in this picture?



Image: Alastair Pollock Photography/Getty Images



Image: Benjamin Lowy/The New York Times



Image: [WeRateDogs](#)

“shark”

“shark”

What is in this picture?



Image: Alastair Pollock Photography/Getty Images



Image: Benjamin Lowy/The New York Times



Image: WeRateDogs

“shark”

“shark”

“shark”

“dog”

Solve the following equation:

$$10230823.5849 \times 3729.2201 + 19420186 = ?$$

Humans are very good at reasoning about images:



“shark”



“shark”



“shark”
“dog”

Computers are very good at doing arithmetic:

$$10230823.5849 \times 3729.2201 + 19420186 = ?$$

Goal for ROB 102 Project 4: Perform image classification with machine learning algorithms.

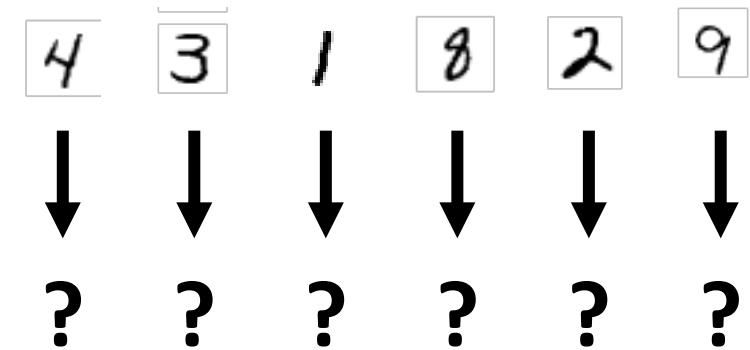
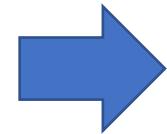
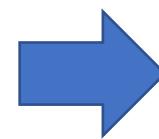


Image: CC4.0 ([link](#))

Goal for ROB 102 Project 4: Perform image classification with machine learning algorithms.



Image: CC4.0 ([link](#))



4 3 / 8 2 9

↓ ↓ ↓ ↓ ↓ ↓

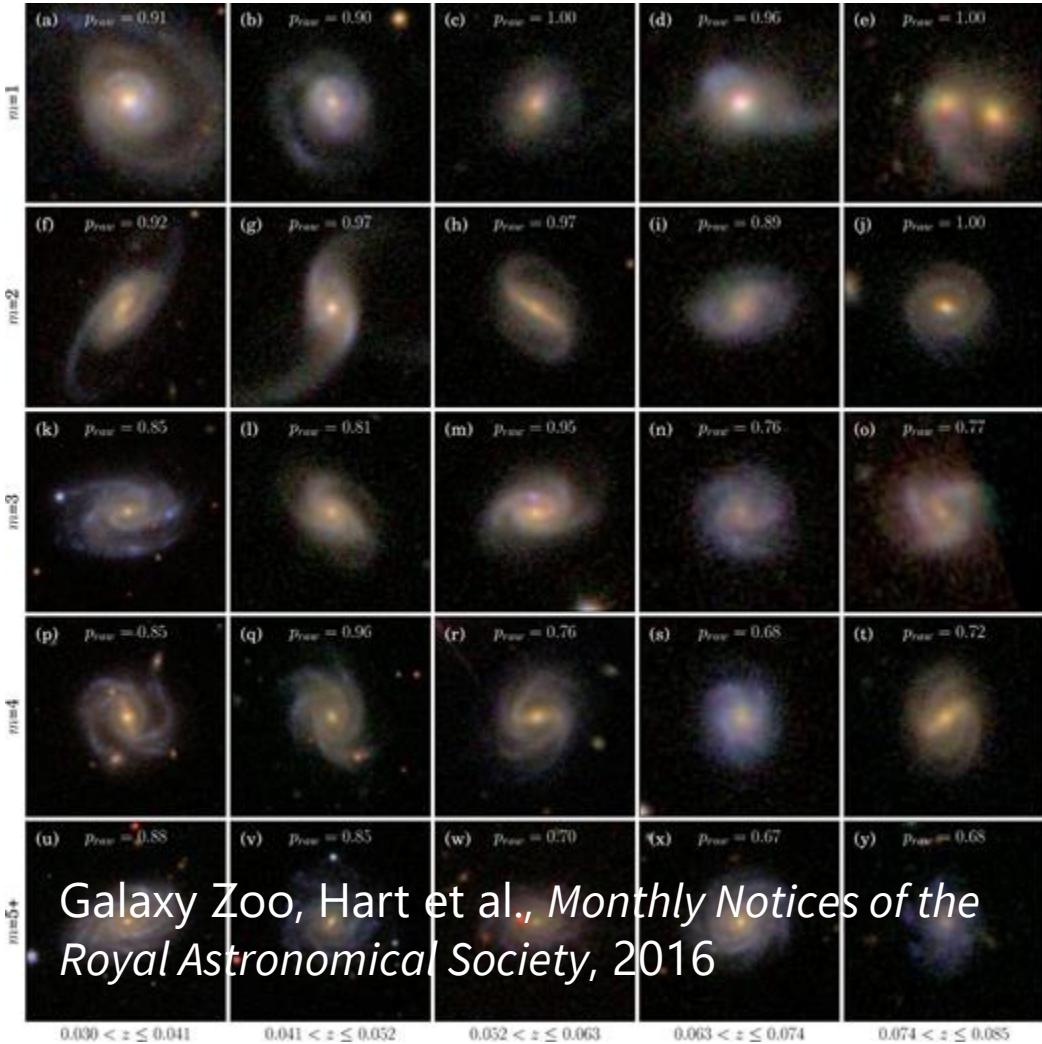
$$10230823.5849 \times 3729.2201 + 1942018$$

⋮

↓ ↓ ↓ ↓ ↓ ↓

? ? ? ? ? ?

Image Classification



Galaxy Classification

Skin Cancer Detection



Esteva et al., Nature, 2017

Image: Matt Young ([link](#))

Handwriting classification

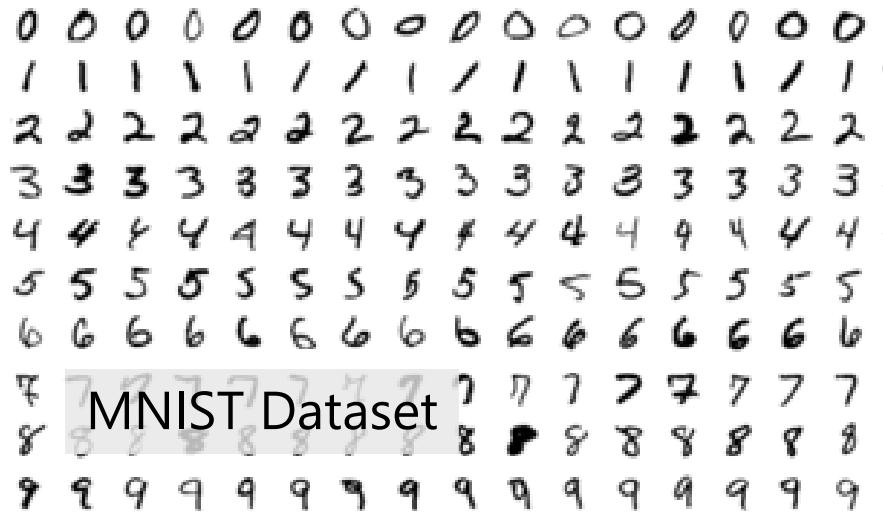


Image Classification: A Building Block

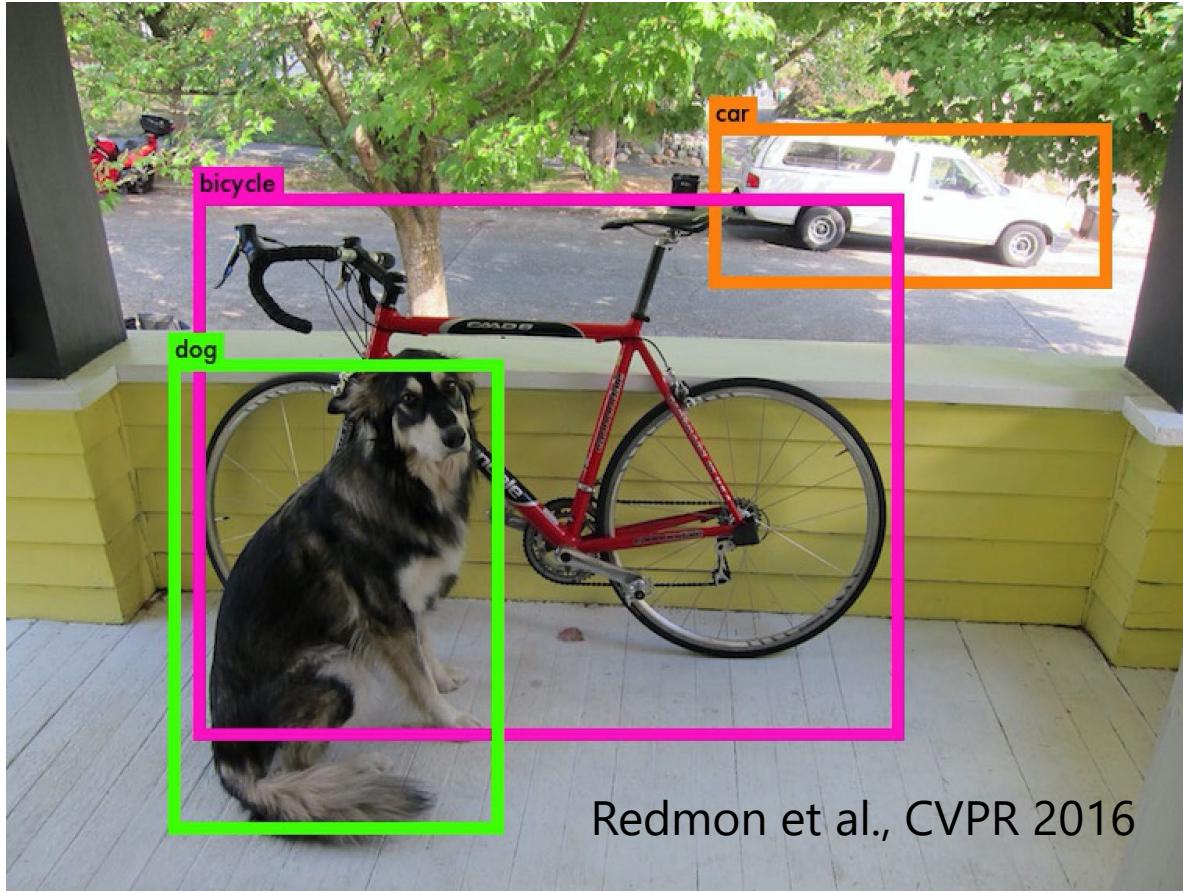


Image Recognition



Image Segmentation

How does a computer process an image?



097	097	097	097	097	097	097	097	097	096	097	097	096	096	096
100	100	100	100	100	100	101	101	102	102	101	100	100	100	099
105	105	105	105	105	105	105	103	102	102	101	103	104	105	
109	109	109	109	109	110	107	118	145	132	120	112	106	103	
113	113	113	112	112	113	110	129	160	160	164	162	157	151	
118	117	118	123	119	118	112	125	142	134	135	139	139	175	
123	121	125	162	166	157	149	153	160	151	150	146	137	168	
127	127	125	168	147	117	139	135	126	147	147	149	156	160	
133	130	150	179	145	132	160	134	150	150	111	145	126	121	
138	134	179	185	141	090	166	117	120	153	111	153	114	126	
144	151	188	178	159	154	172	147	159	170	147	185	105	122	
152	157	184	183	142	127	141	133	137	141	131	147	144	147	
130	147	185	180	139	131	154	121	140	147	107	147	120	128	
035	102	194	175	149	140	179	128	146	168	096	163	101	125	

An image is a grid of pixels. Each pixel has a numerical value.

For a greyscale image, the value corresponds to the intensity of that pixel.

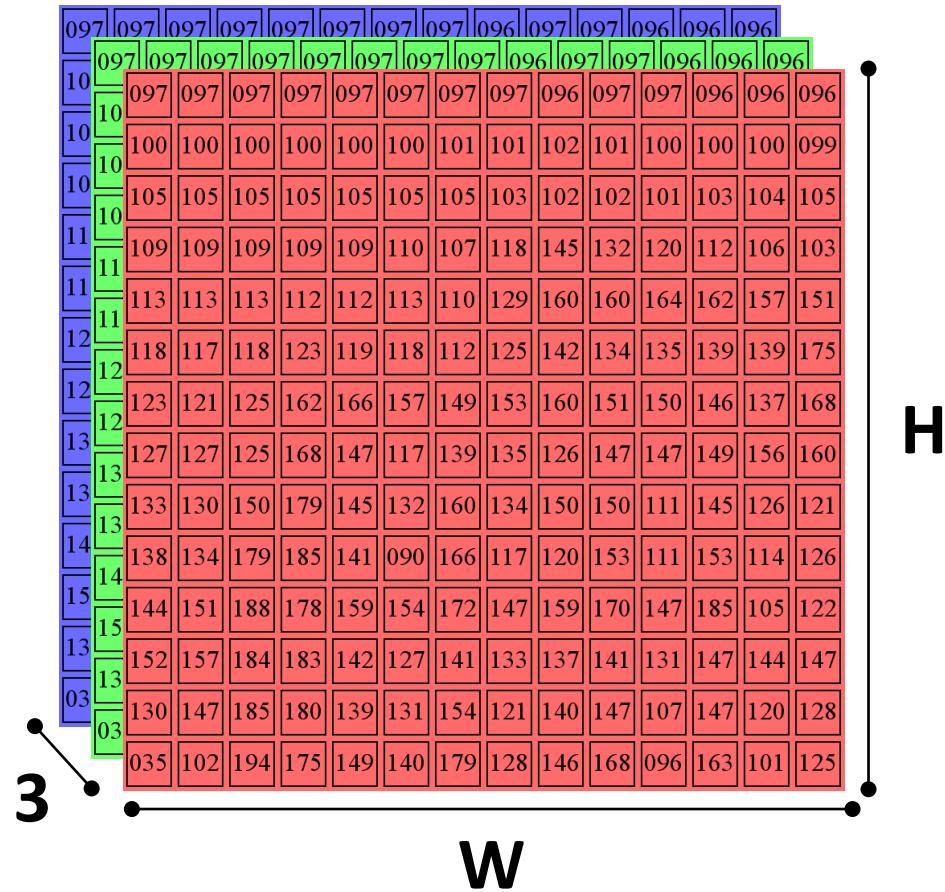
How does a computer process an image?



097	097	097	097	097	097	097	097	097	096	097	097	096	096	096
100	100	100	100	100	100	101	101	102	102	101	100	100	100	099
105	105	105	105	105	105	105	103	102	102	101	103	104	105	
109	109	109	109	109	110	107	118	145	132	120	112	106	103	
113	113	113	112	112	113	110	129	160	160	164	162	157	151	
118	117	118	123	119	118	112	125	142	134	135	139	139	175	
123	121	125	162	166	157	149	153	160	151	150	146	137	168	
127	127	125	168	147	117	139	135	126	147	147	149	156	160	
133	130	150	179	145	132	160	134	150	150	111	145	126	121	
138	134	179	185	141	090	166	117	120	153	111	153	114	126	
144	151	188	178	159	154	172	147	159	170	147	185	105	122	
152	157	184	183	142	127	141	133	137	141	131	147	144	147	
130	147	185	180	139	131	154	121	140	147	107	147	120	128	
035	102	194	175	149	140	179	128	146	168	096	163	101	125	

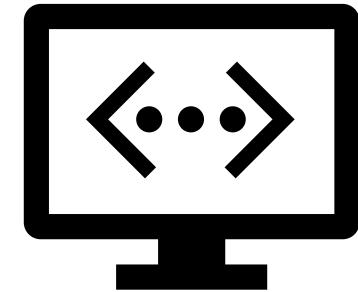
The image has size HxW.

How does a computer process an image?



A color can be represented by 3 values at each pixel: **red**, **green**, **blue** (**RGB**)
A full-color image can be represented by $H \times W \times 3$ numbers.

We could try to code up a classifier...



Catari 2600

Challenges: Viewpoint Variation

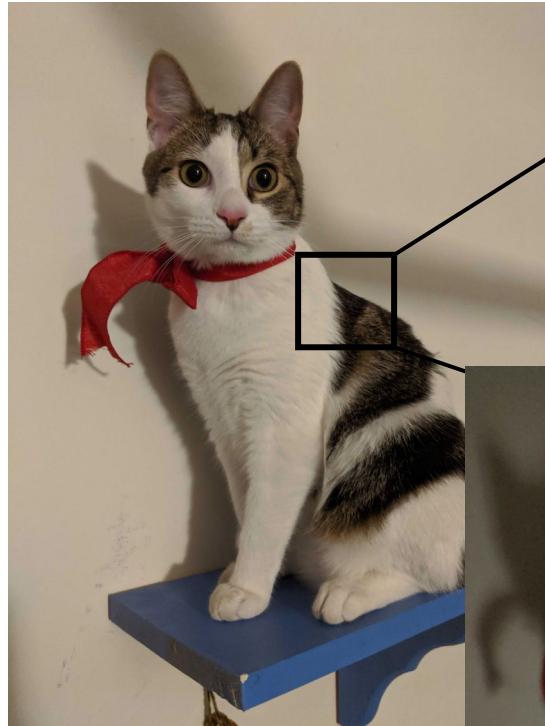
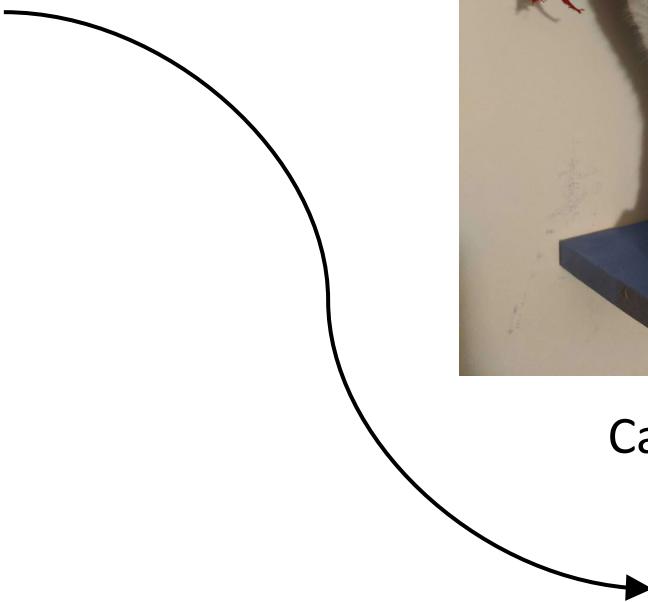


Catari 2600

097	097	097	097	097	097	097	097	097	096	097	097	096	096
100	100	100	100	100	100	100	101	101	102	101	100	100	100
105	105	105	105	105	105	105	105	103	102	102	101	103	104
109	109	109	109	109	110	107	118	145	132	120	112	106	
113	113	113	112	112	113	110	129	160	160	164	162	157	
118	117	118	123	119	118	112	125	142	134	135	139	139	
123	121	125	162	166	157	149	153	160	151	150	146	137	
127	127	125	168	147	117	139	135	126	147	147	149	156	
133	130	150	179	145	132	160	134	150	150	111	145	126	
138	134	179	185	141	090	166	117	120	153	111	153	114	
144	151	188	178	159	154	172	147	159	170	147	185	105	

Pixels change values when
the camera moves

Challenges: Viewpoint Variation



Catari 2600



097	097	097	097	097	097	097	097	097	096	097	097	096	096
100	100	100	100	100	100	100	101	101	102	101	100	100	100
105	105	105	105	105	105	105	103	102	102	101	103	104	
109	109	109	109	109	110	107	118	145	132	120	112	106	
113	113	113	112	112	113	110	129	160	160	164	162	157	
118	117	118	123	119	118	112	125	142	134	135	139	139	
123	121	125	162	166	157	149	153	160	151	150	146	137	
127	127	125	168	147	117	139	135	126	147	147	149	156	
									179	145	132	160	134
									185	141	090	166	117
									178	159	154	172	147
									159	170	147	185	105

Pixels change values when
the camera moves

Challenges: Variation within Classes



This image is [CC0 1.0](#) public domain

Challenges: Background Clutter



[This image is CCO 1.0 public domain](#)



[This image is CCO 1.0 public domain](#)

Challenges: Lighting Changes



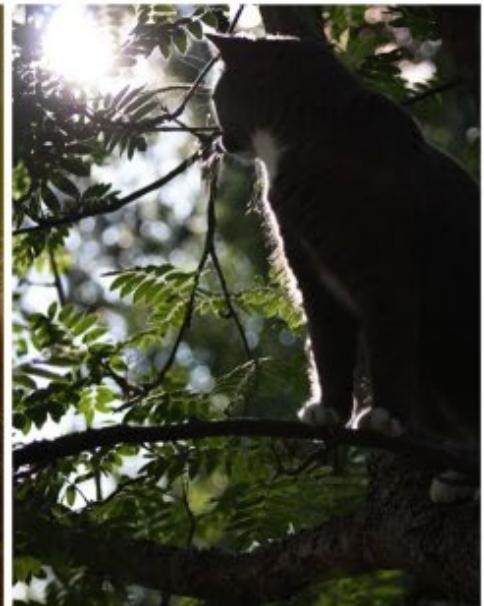
[This image is CC0 1.0 public domain](#)



[This image is CC0 1.0 public domain](#)



[This image is CC0 1.0 public domain](#)

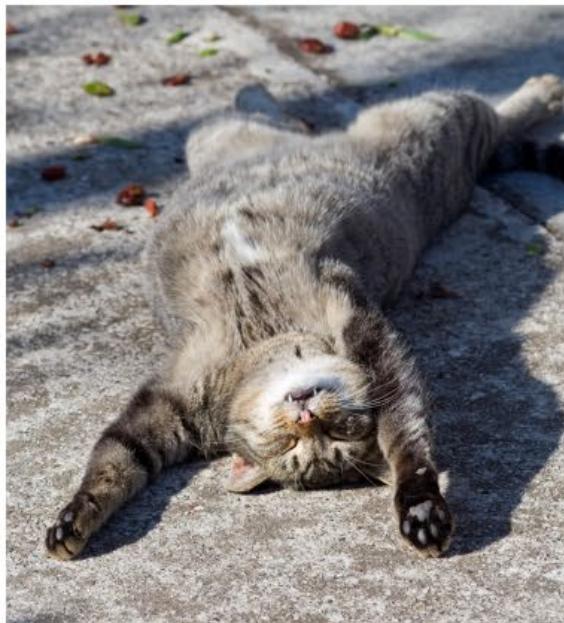


[This image is CC0 1.0 public domain](#)

Challenges: Deformation



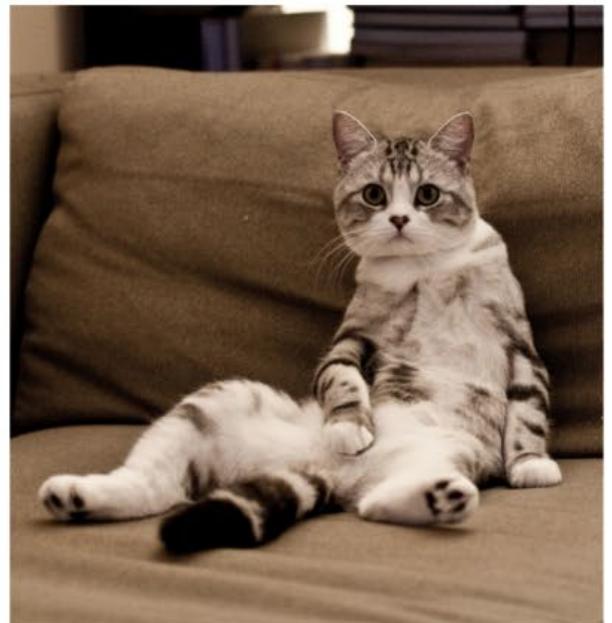
This image by [Umberto Salvagnin](#) is
licensed under [CC-BY 2.0](#)



This image by [Umberto Salvagnin](#) is
licensed under [CC-BY 2.0](#)



This image by [sare bear](#) is licensed
under [CC-BY 2.0](#)



This image by [Tom Thai](#) is licensed
under [CC-BY 2.0](#)

Challenges: Occlusion



[This image is CC0 1.0 public domain](#)



[This image is CC0 1.0 public domain](#)



[This image by jonsson is licensed under CC-BY 2.0](#)

Data-Driven Approach

Idea: Use (lots & lots of) data to learn to classify images.

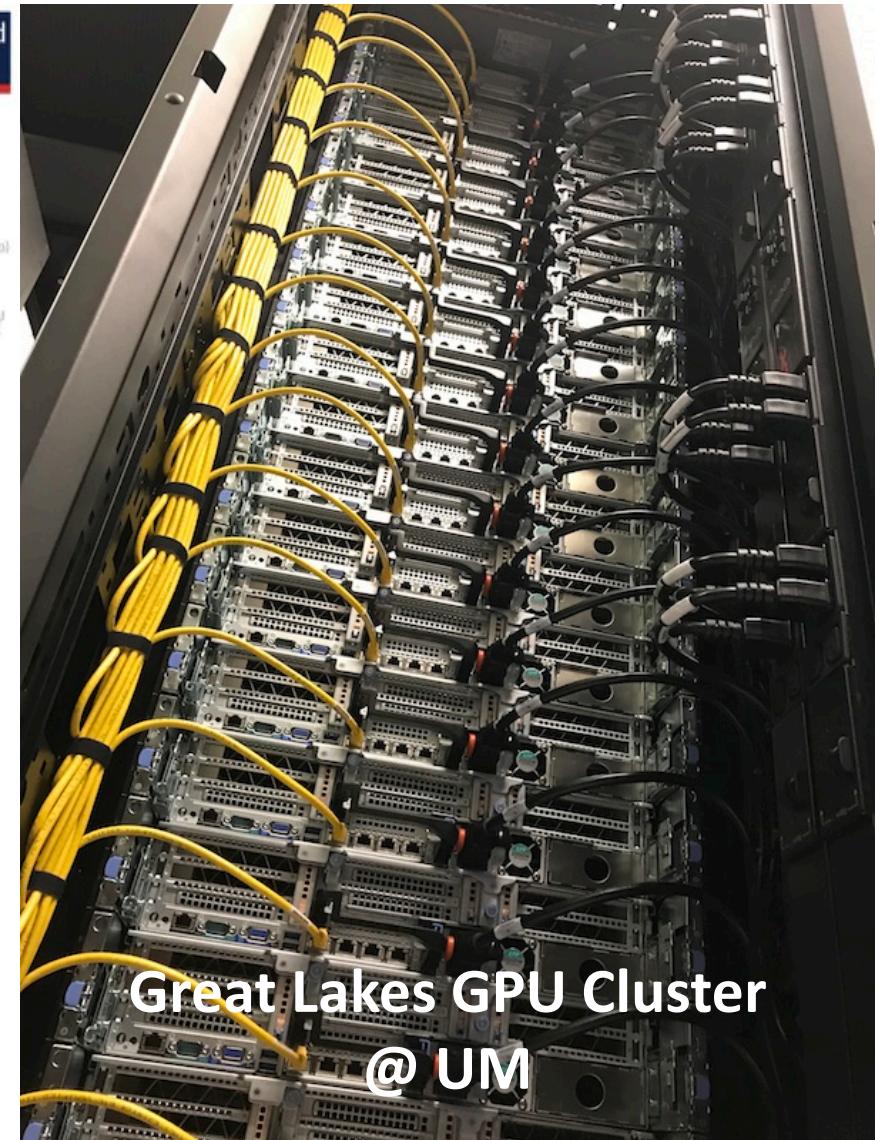
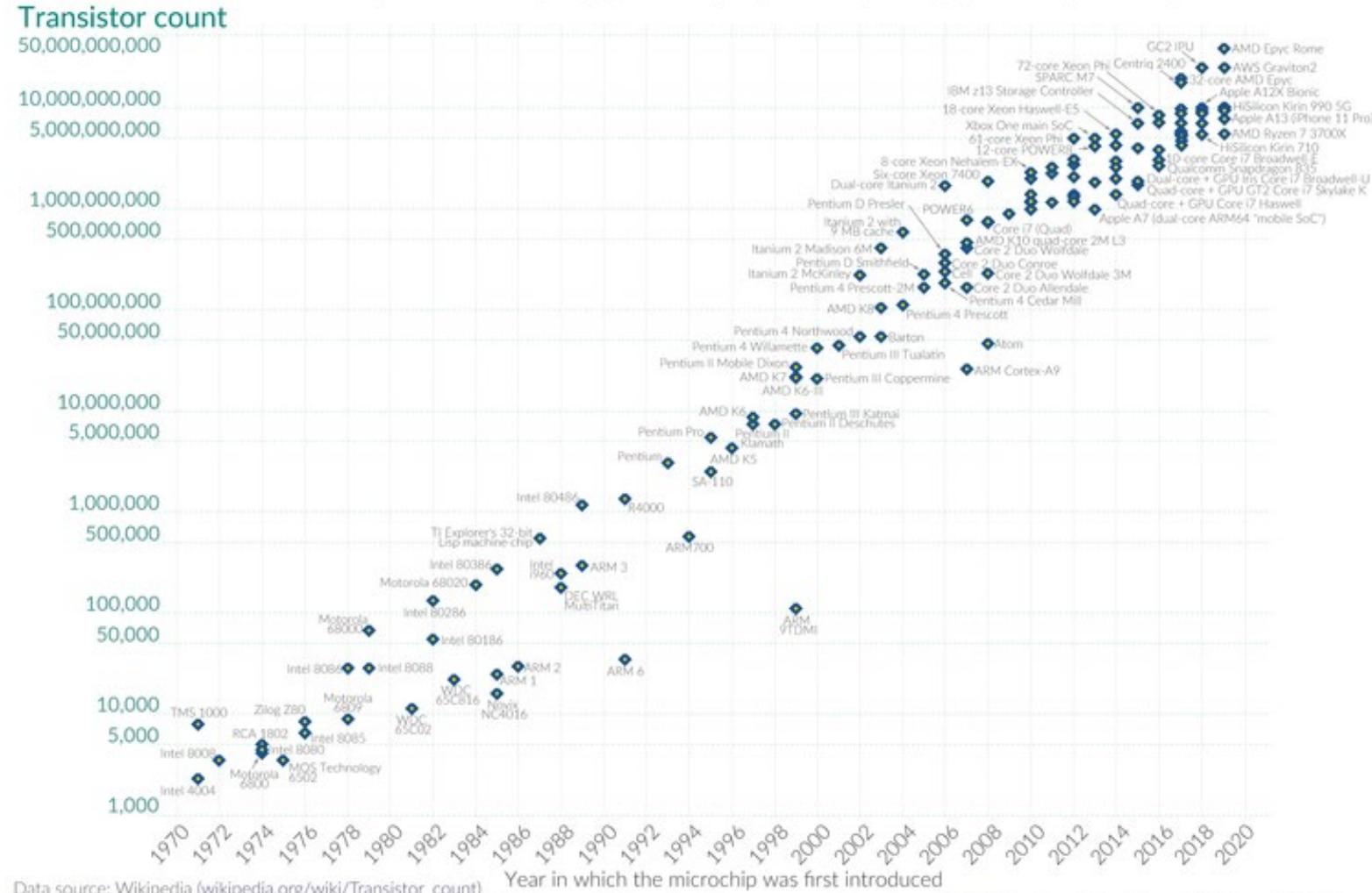
1. Get a bunch of labelled data.
2. Use Supervised Learning to train a classifier.
3. Use the classifier to label new images.

This will take lots and lots of data and compute power.

Moore's Law: The number of transistors on microchips doubles every two years

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.

Our World
in Data



Great Lakes GPU Cluster @ UM

Data source: Wikipedia ([wikipedia.org/wiki/Transistor_count](https://en.wikipedia.org/w/index.php?title=Transistor_count&oldid=1000000000))

OurWorldInData.org – Research and data to make progress against the world's largest problems.

Licensed under CC-BY by the authors Hannah Ritchie and Max Roser

Computers have gotten very fast.

IMAGENET Large Scale Visual Recognition Challenge

The Image Classification Challenge:
1,000 object classes
1,431,167 images



Output:
Scale
T-shirt
Steel drum
Drumstick
Mud turtle

Deng et al, 2009
Russakovsky et al. IJCV 2015

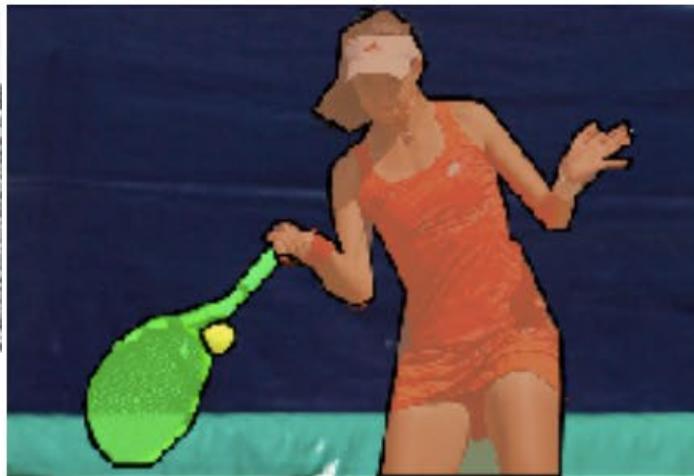
Huge open-source datasets are increasingly common.

IMAGENET Large Scale Visual Recognition Challenge

The Image Classification Challenge:
COCO 2020 Object Detection Task

Output:
Scale

COCO Dataset ([link](#))

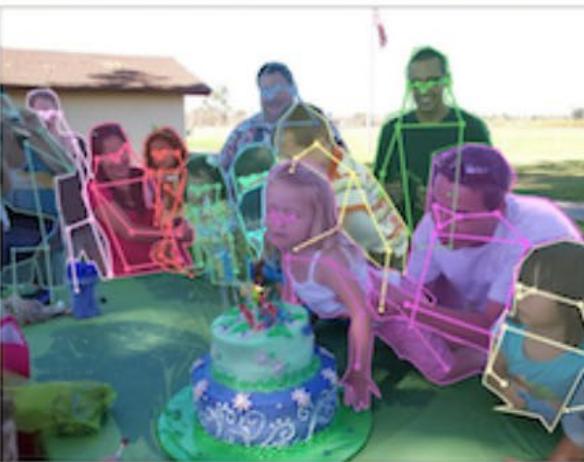
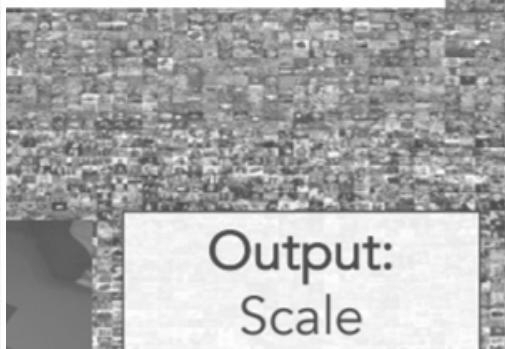


Huge open-source datasets are increasingly common.

COCO 2020 Keypoint Detection Task

COCO Dataset ([link](#))

Challenge



Huge open-source datasets are increasingly common.

COCO 2020 Keypoint Detection Task



Huge open-source datasets are increasingly common.

The MNIST dataset of images

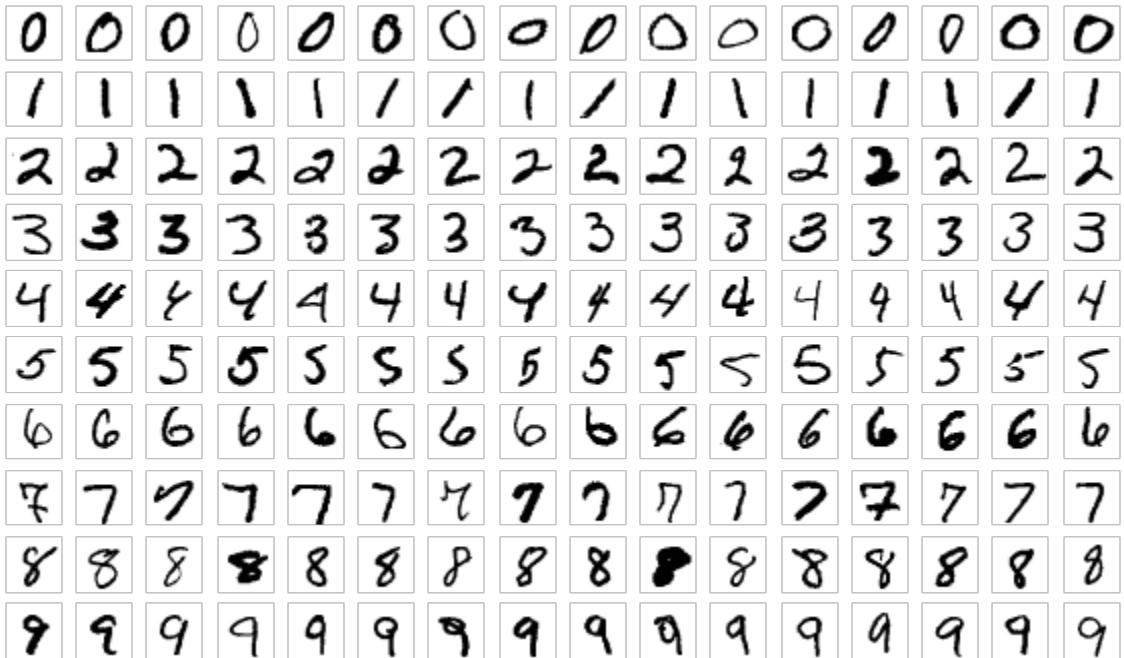


Image: CC4.0 ([link](#))

MNIST contains 70k images of handwritten digits. All of them are labelled as a digit from 0 to 9.

The images are 28x28 pixels (tiny!).

Machine Learning Algorithm

Training time:

Learn a prediction model by optimizing over a labeled dataset.

Testing time:

Use the model to perform prediction on new data.

Machine Learning Algorithm

Training time:

Learn a prediction model by optimizing over a labeled dataset.

Testing time:

Use the model to perform prediction on new data.

We have to pick the data, the model, and the optimization method.

Training & Testing Datasets

Training set: Labelled data used for training a machine learning algorithm.

Test set: Data used to test the accuracy of the machine learning algorithm.

- Usually smaller than the training set
- Also has labels, only used for measuring how good the algorithm is (no cheating!!)
- We don't look at this during training, so we are testing on images the algorithm has never seen before.

In MNIST, there are 60k training images and 10k test images.

Project 4: Machine Learning

Implement three machine learning algorithms to classify images from the MNIST dataset.

1. Nearest Neighbors
2. Linear Classifier
3. Neural Network

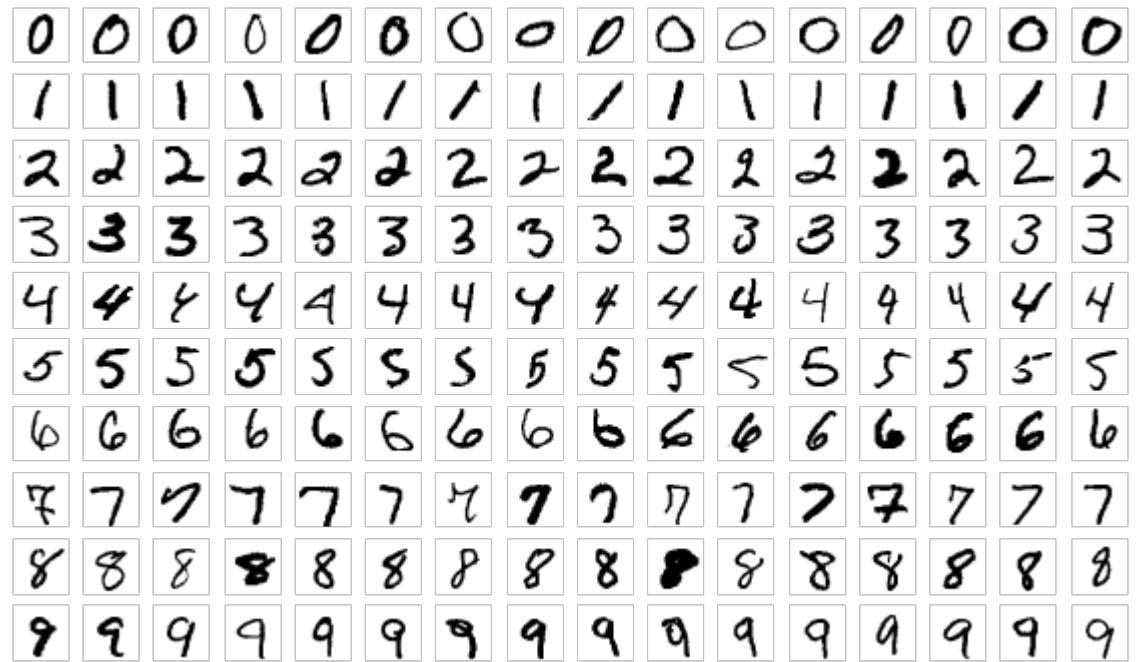
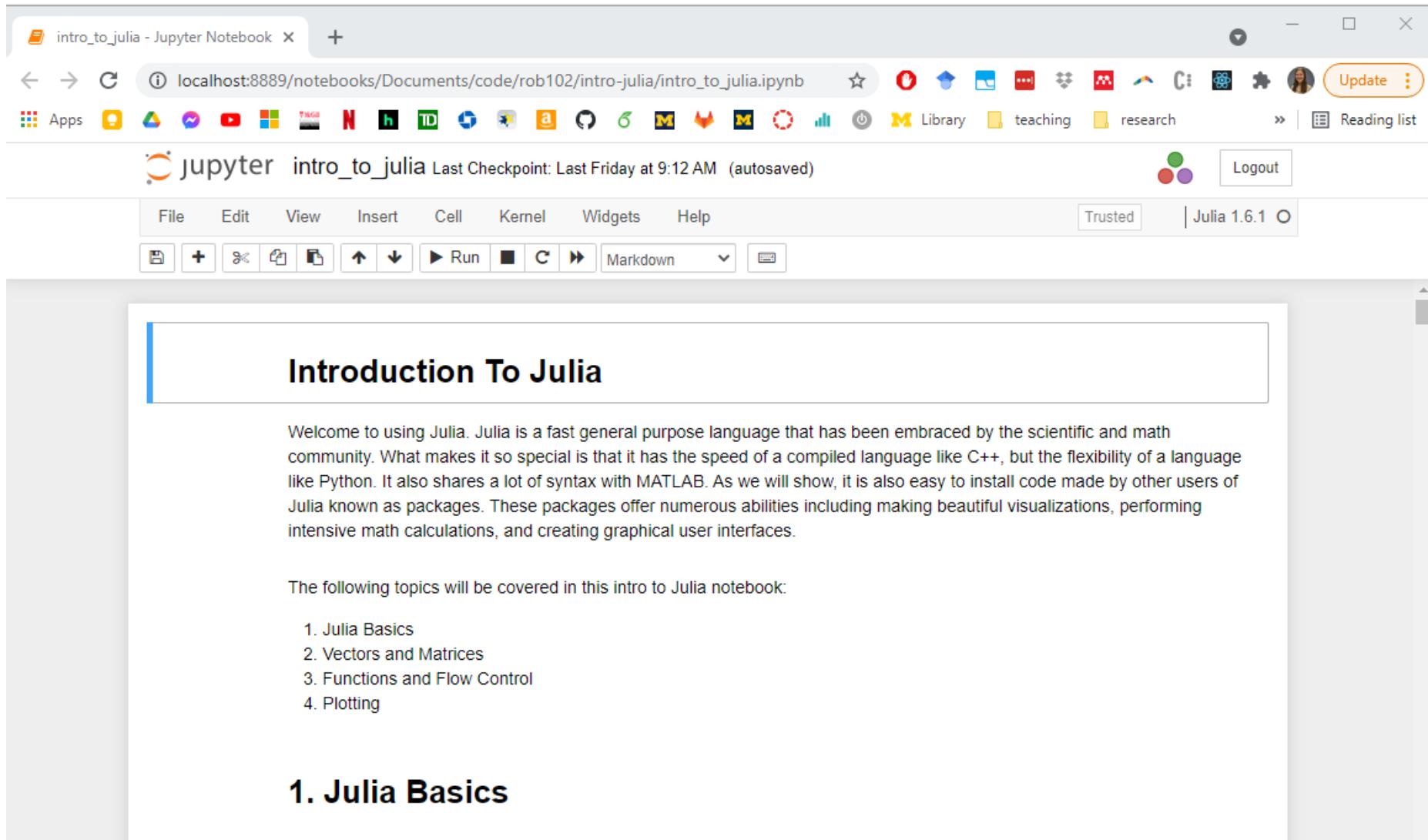


Image: CC4.0 ([link](#))

Julia & Jupyter Notebooks



The screenshot shows a Jupyter Notebook interface running in a web browser. The title bar of the browser window reads "intro_to_julia - Jupyter Notebook". The address bar shows the URL "localhost:8889/notebooks/Documents/code/rob102/intro-julia/intro_to_julia.ipynb". The browser toolbar includes standard icons for back, forward, search, and refresh. Below the toolbar, there's a row of colored icons representing various apps. The main content area is titled "jupyter intro_to_julia Last Checkpoint: Last Friday at 9:12 AM (autosaved)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. A status bar at the bottom indicates "Trusted" and "Julia 1.6.1". The central content pane displays the first section of a notebook:

Introduction To Julia

Welcome to using Julia. Julia is a fast general purpose language that has been embraced by the scientific and math community. What makes it so special is that it has the speed of a compiled language like C++, but the flexibility of a language like Python. It also shares a lot of syntax with MATLAB. As we will show, it is also easy to install code made by other users of Julia known as packages. These packages offer numerous abilities including making beautiful visualizations, performing intensive math calculations, and creating graphical user interfaces.

The following topics will be covered in this intro to Julia notebook:

1. Julia Basics
2. Vectors and Matrices
3. Functions and Flow Control
4. Plotting

1. Julia Basics

Next time...

Implement three machine learning algorithms to classify images from the MNIST dataset.

1. **Nearest Neighbors** ← Next lecture!
2. Linear Classifier
3. Neural Network

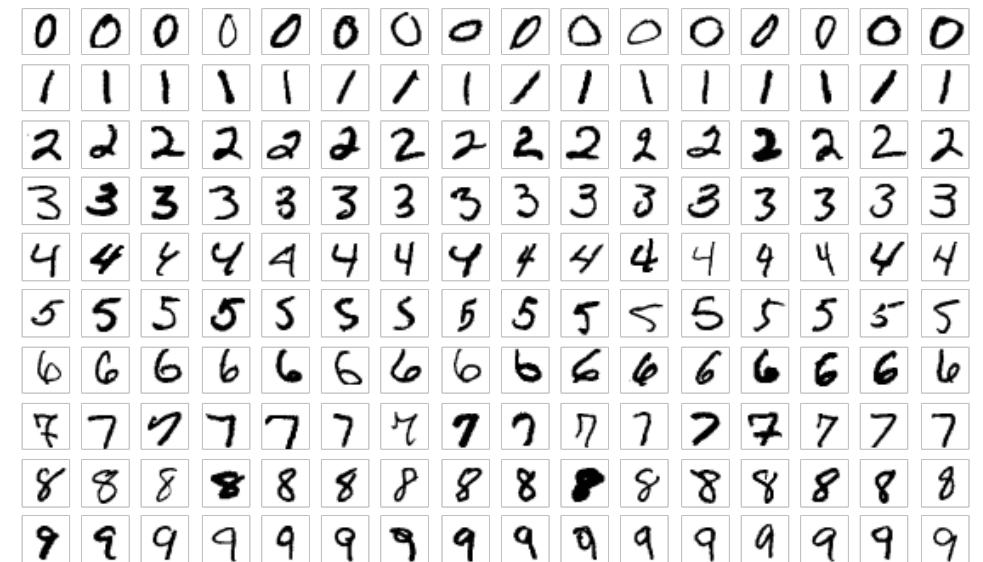


Image: CC4.0 ([link](#))