K-Nearest Neighbours and Introduction to Python

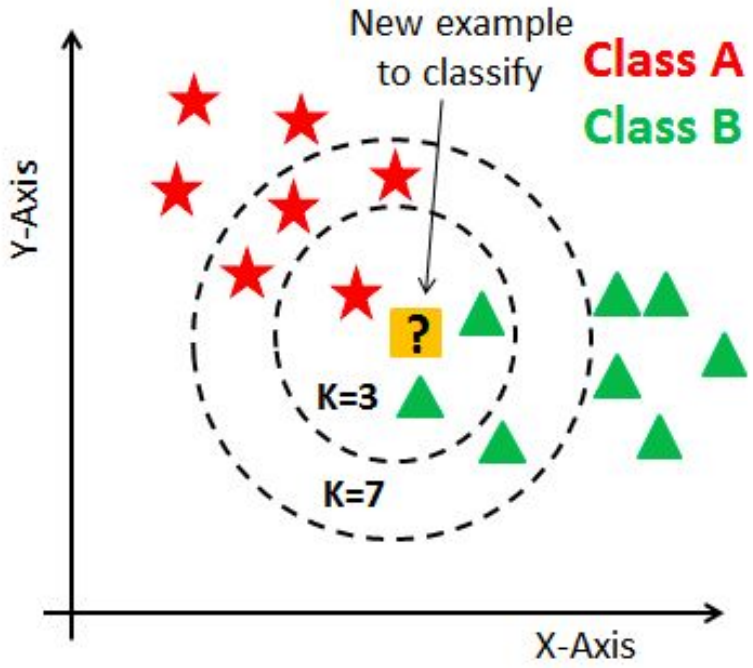
ACM AI | Intro to Machine Learning: Beginner Track #5

Slides: tinyurl.com/f20btrack5
Attendance code: **coco**
Discord: bit.ly/ACMdiscord

acm.ai

K-Nearest Neighbours Classification (KNN)

What is KNN?



- Goal: classify the new data point based on how its neighbours are classified
- One of the simplest **supervised** ML algorithms
- Observe that the outputs are discrete
  - Either class A or B

Intuition Behind KNN

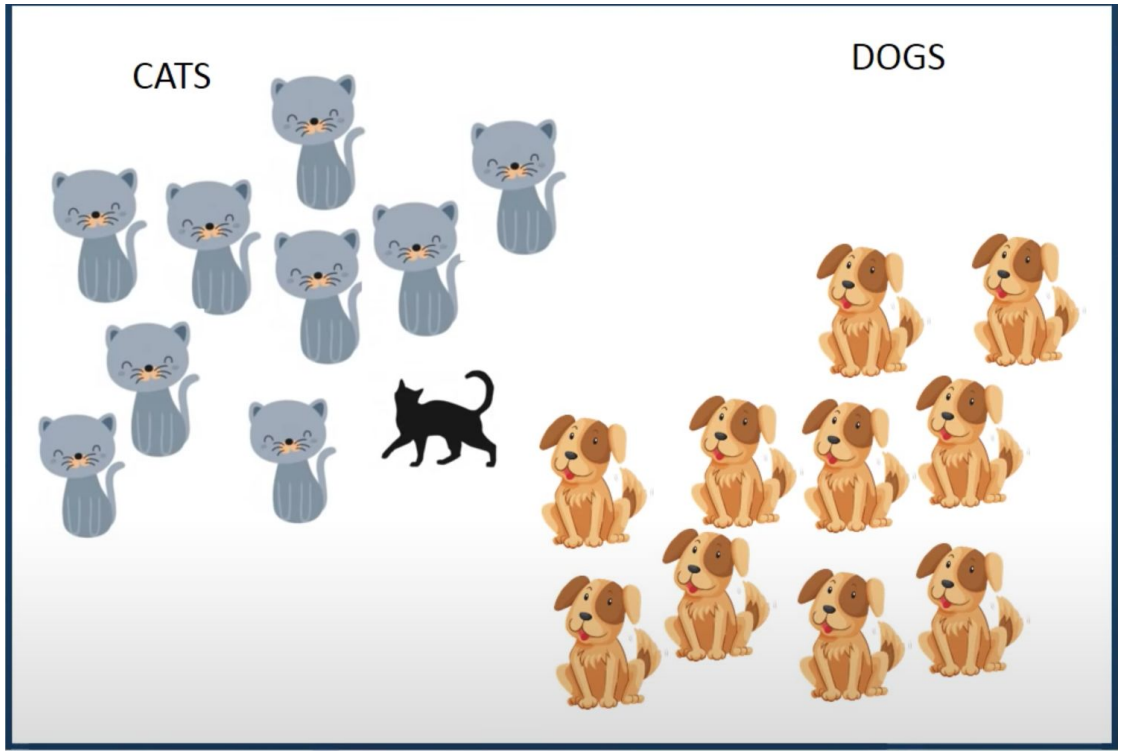
Feature Differences between Cats and Dogs

- Sharp claws (used for climbing)
- Shorter ears
- Meow

- Dull claws
- Longer ears
- Bark

Graphical Representation

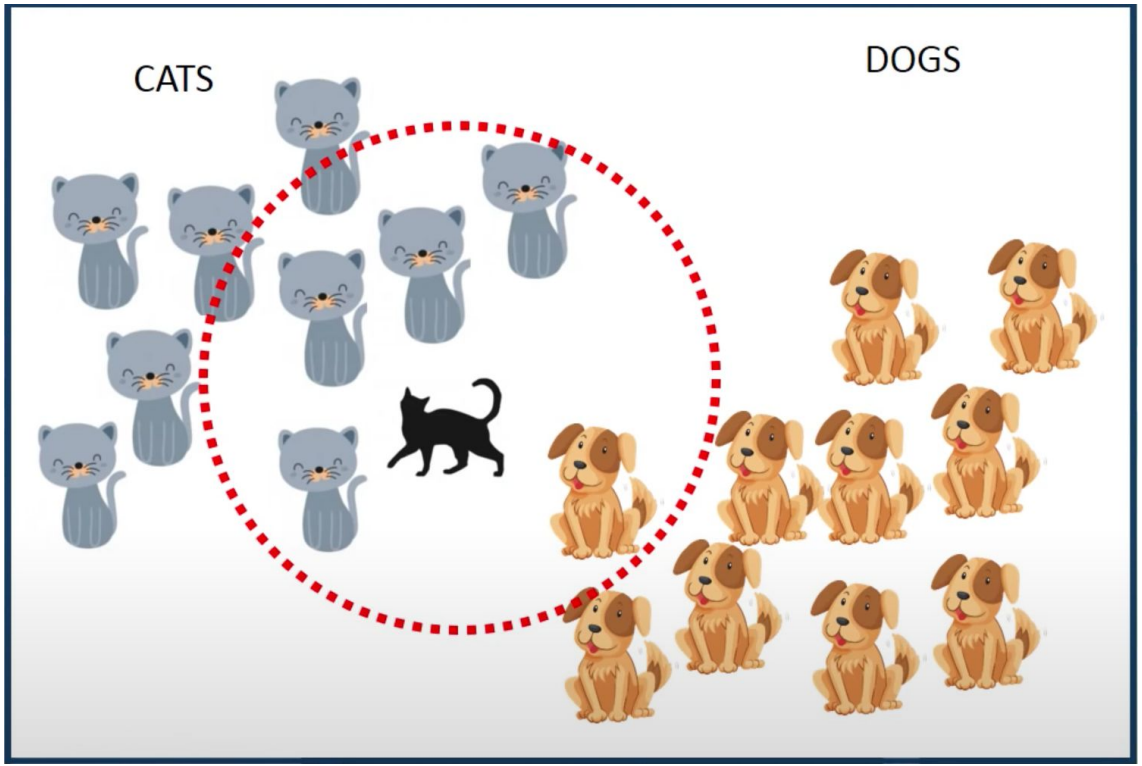
Sharpness of claws



Length of ears

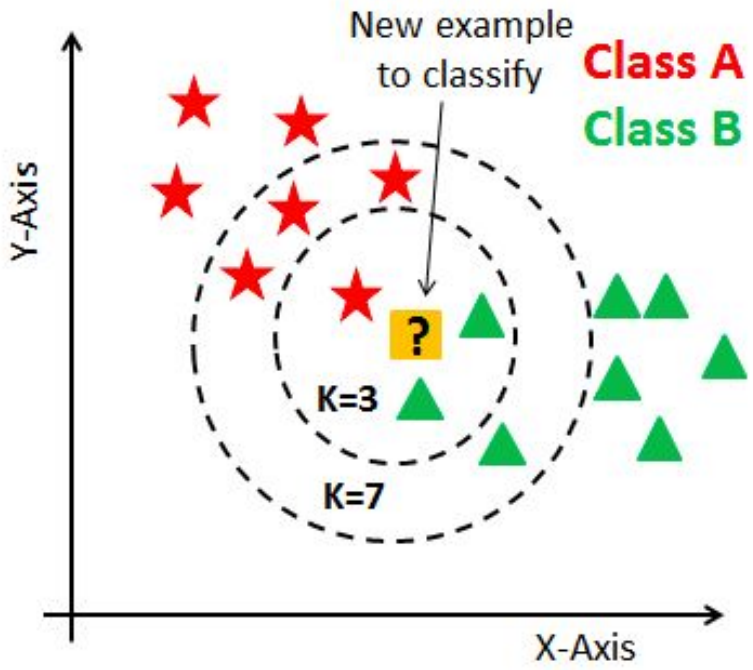
Is it a Cat or a Dog?

Sharpness of claws



Length of ears

How does KNN actually work?

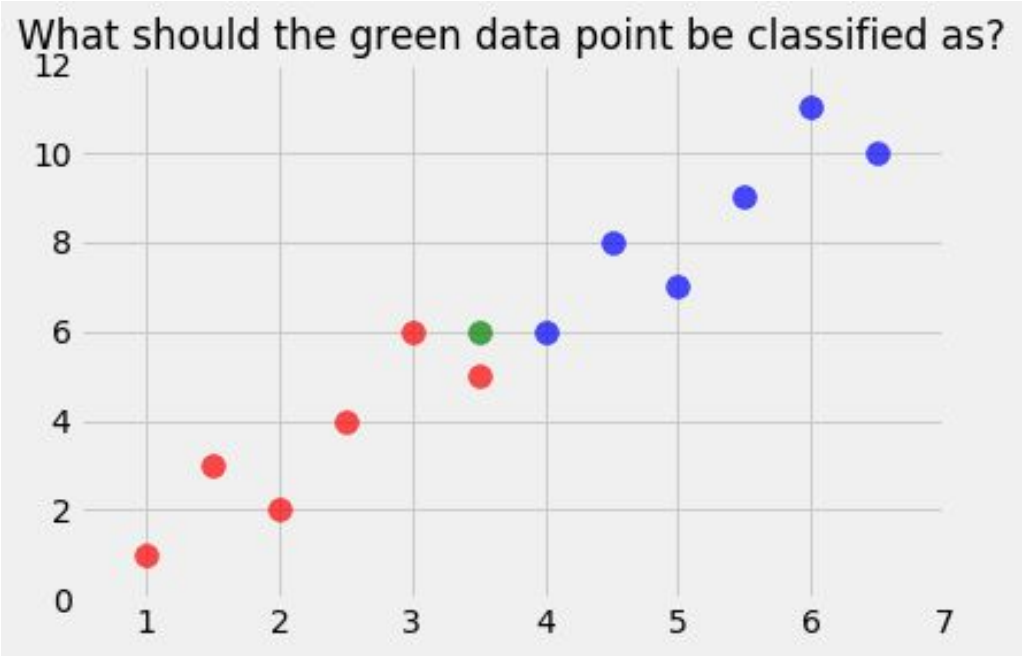


- Start with a labelled dataset
- Classify new data points based on how their neighbours are classified
- k is a parameter that refers to the number of nearest neighbours to include in the voting process

Quick Poll

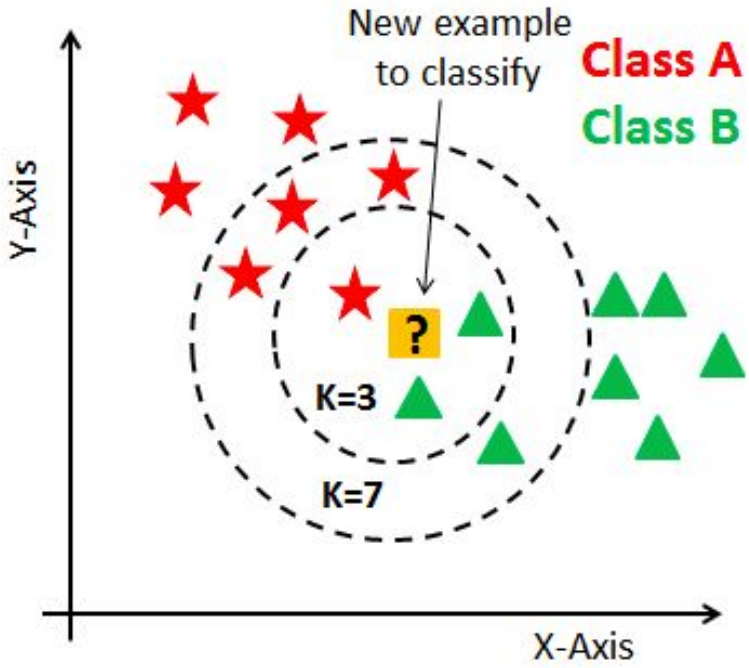
What should the green data point be classified as? Choose k = 3

a. Red
b. Blue
c. Purple
d. None of the above



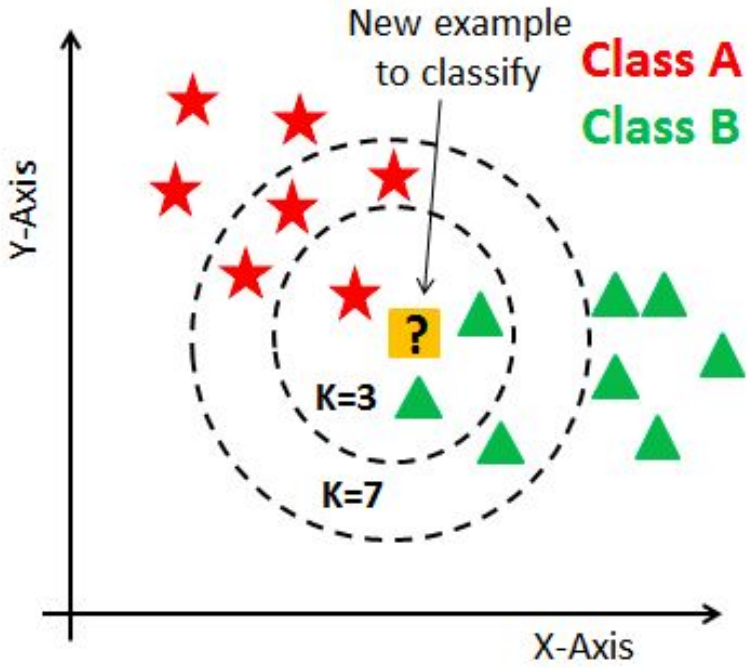
acm.ai

How do we choose the factor ‘K’?



- KNN is based on **feature similarity**
  - Choosing the right value of k is a process of parameter tuning and is important for better accuracy
- Here’s an example of a dilemma
  - At k = 3, we classify ? as a triangle
  - But at k = 7, we classify ? as a star
- So which one is correct?

How do we choose the factor ‘K’?



- In general, if we pick a k that is too low, the results can be easily skewed by outliers
- If we pick a k that is too high, it is too expensive computationally to process
- A general rule of thumb for choosing k
  - k = sqrt(n), where n is the total number of data points
  - **Odd** value of k to avoid a potential tie in the voting process

When can we use KNN?

- When data is noise-free
  - Example of noisy data:

Weight (kg)	Height (cm)	Class
51	167	Underweight
62	182	one-fourty
69	176	23
64	173	Hello kitty
65	172	Normal

- When the dataset is relatively small
  - Because KNN is a lazy learner
  - I.e. It doesn't learn a discriminative function from the training dataset

Quick Poll

If there are 170 data points in your KNN algorithm, what k value should you pick?

a. 11
b. 12
c. 13
d. 14

Case Study

How does KNN work in practice?

- Consider a dataset having 2 variables: height and weight, and each data point is classified as Normal or Underweight

Weight (kg)	Height (cm)	Class
51	167	Underweight
62	182	Normal
69	176	Normal
64	173	Normal
65	172	Normal
56	174	Underweight
58	169	Normal
57	173	Normal
55	170	Normal

How does KNN work in practice?

- On the basis of the given data, we have to classify the below set as Normal or Underweight using KNN

57 kg	170 cm	?

- To find the nearest neighbours, we will use Euclidean distance

THE DISTANCE FORMULA

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- Assume x corresponds to weight and y corresponds to height

How does KNN work in practice?

- We will calculate the Euclidean distance of unknown data from all the points in the dataset
  - $dist(d1) = \sqrt{(170 - 167)^2 + (57 - 51)^2}$ ~=6.7
  - $dist(d2) = \sqrt{(170 - 182)^2 + (57 - 62)^2}$ ~=13
    - So on and so forth

Weight (kg)	Height (cm)	Class
51	167	Underweight
62	182	Normal
69	176	Normal
64	173	Normal
65	172	Normal
56	174	Underweight
58	169	Normal
57	173	Normal
55	170	Normal

How does KNN work in practice?

- Here, we have calculated the Euclidean distance of all unknown data point from all the points as shown, where (x1, y1) = (57,170) whose class we have to classify.
- Total number of data points (including the unknown data point) = 10
- K = sqrt(10) ~= 3

Weight (x2)	Height (y2)	Class	Euclidean distance
51	167	Underweight	6.7
62	182	Normal	13
69	176	Normal	13.4
64	173	Normal	7.6
65	172	Normal	8.2
56	174	Underweight	4.1
58	169	Normal	1.4
57	173	Normal	3
55	170	Normal	2

Recap of KNN

- Start with a labelled dataset, along with a new sample/data point
- We select k entries from our dataset closest to that new sample
- We find the most common classification of these entries
- This is the classification we give to the new sample

Introduction to Python

Google Colab

- Use the link on the slide to access a copy of the notebook so you can work along with us!

- tinyurl.com/btrackpython

Worksheet

- If you want some extra practice, we've put together a bunch of cool problems for you to work on whenever you like.
- tinyurl.com/btrackpythonws
- Feel free to reach out to us if you get stuck on any problem!

  (StackOverflow is also a great resource for debugging code)

Thank you! We'll see you next week!

Please fill out our feedback form:
tinyurl.com/f20btrack5fb

Next week: Data Analytics

numpy and pandas (who doesn't love pandas?)

Today's event code: **coco**

FB group: facebook.com/groups/uclaacmai

Github: github.com/uclaacmai/beginner-track-fall-2020

acm.ai