



# The K-Nearest Neighbors

---

This Photo by Unknown Author is licensed under CC BY



Is that a dog?



No dear, you can  
differentiate  
between a cat  
and a dog based  
on their  
characteristics

### CATS



Sharp Claws, uses to climb

Smaller length of ears

Meows and purrs

Doesn't love to play around

### DOGS



Dull Claws

Bigger length of ears

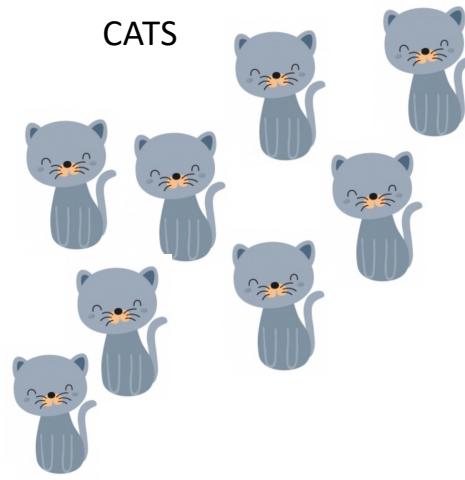
Barks

Loves to run around

No dear, you can  
differentiate  
between a cat  
and a dog based  
on their  
characteristics

Sharpness of claws →

CATS



DOGS



Length of ears →

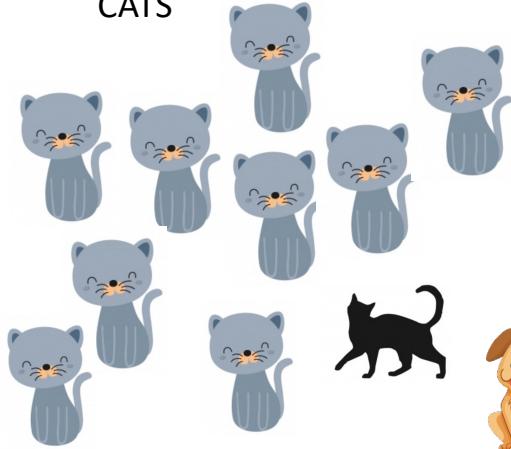
No dear, you can  
differentiate  
between a cat  
and a dog based  
on their  
characteristics



Now tell me if it  
is a cat or a dog?

Sharpness of claws →

CATS



DOGS



Length of ears →

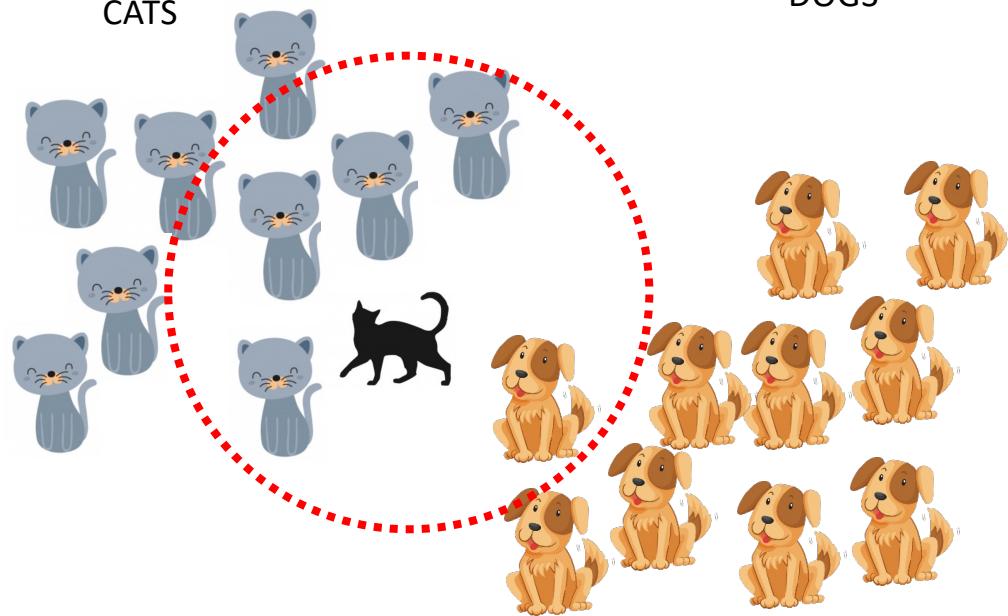
Now tell me if  
it's a cat or a  
dog?



It's features are  
more like cats, it  
must be a cat!

Sharp of claws →

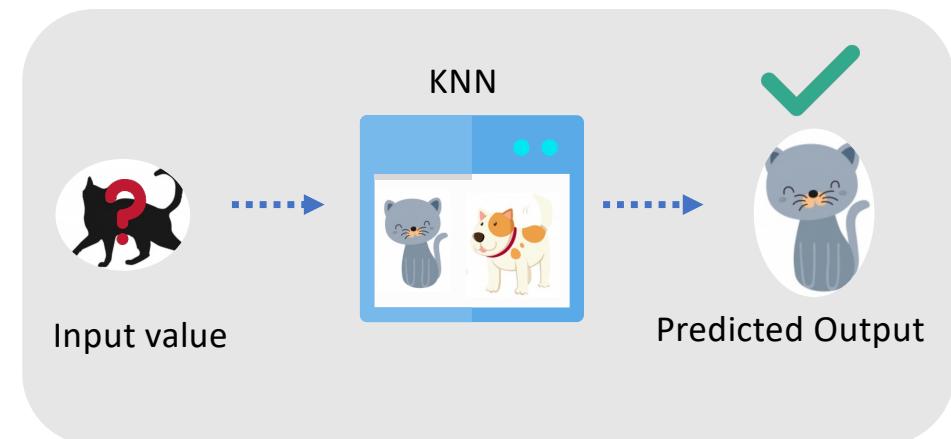
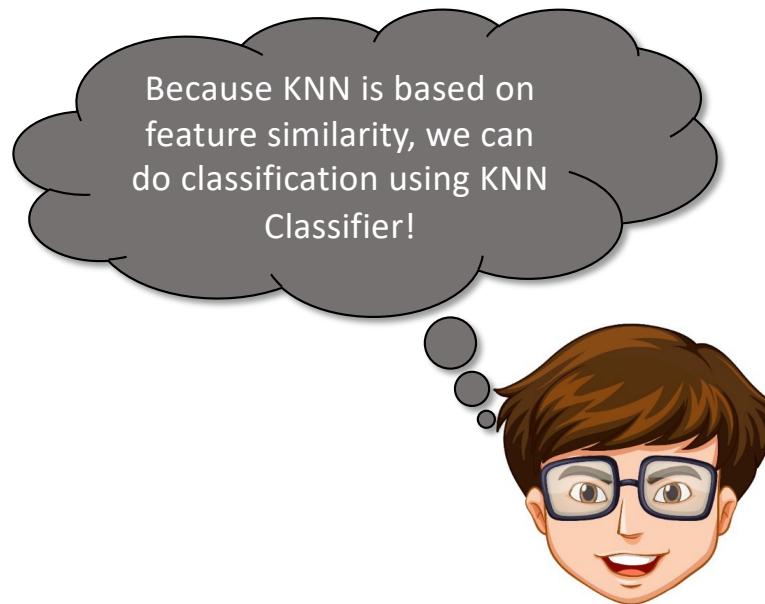
CATS



DOGS

Length of ears →

## Why KNN?



# What is KNN Algorithm?

KNN – K Nearest Neighbors, is one of the simplest **Supervised** Machine Learning algorithm mostly used for

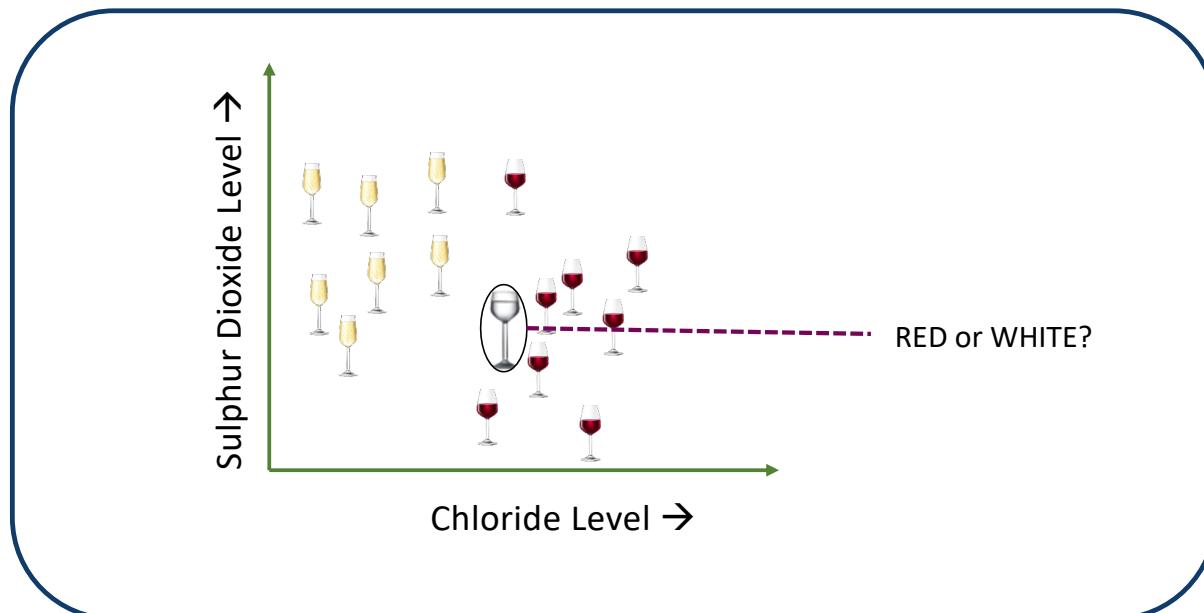
## Classification



It classifies a data point based on how its neighbors are classified

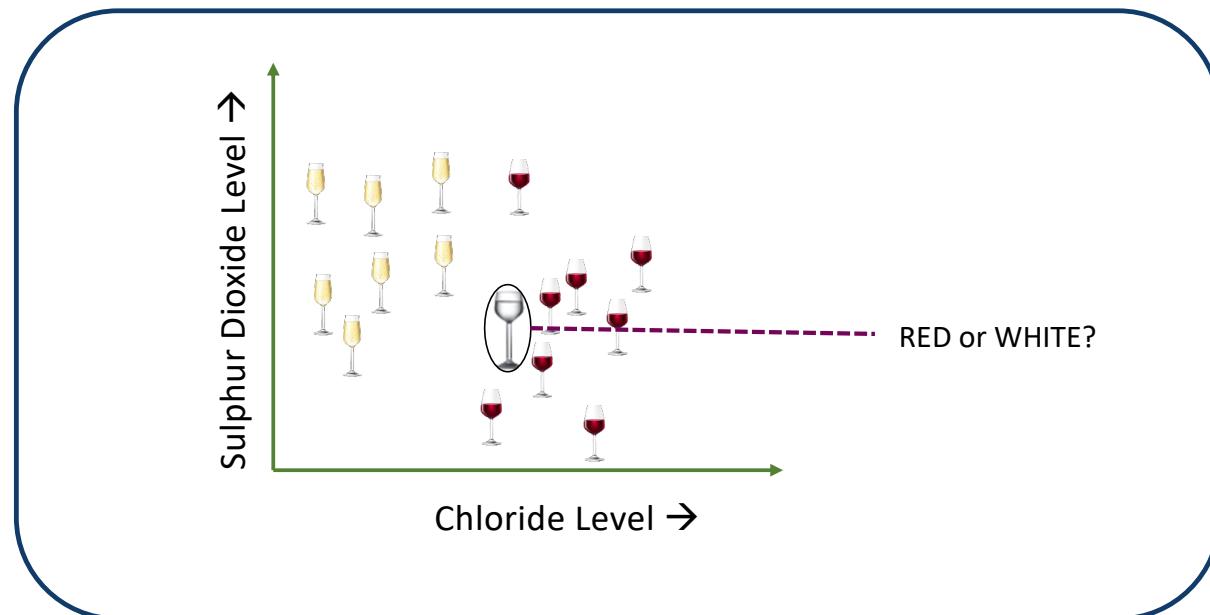
# What is KNN Algorithm?

KNN stores all available cases and classifies new cases based on a similarity measure



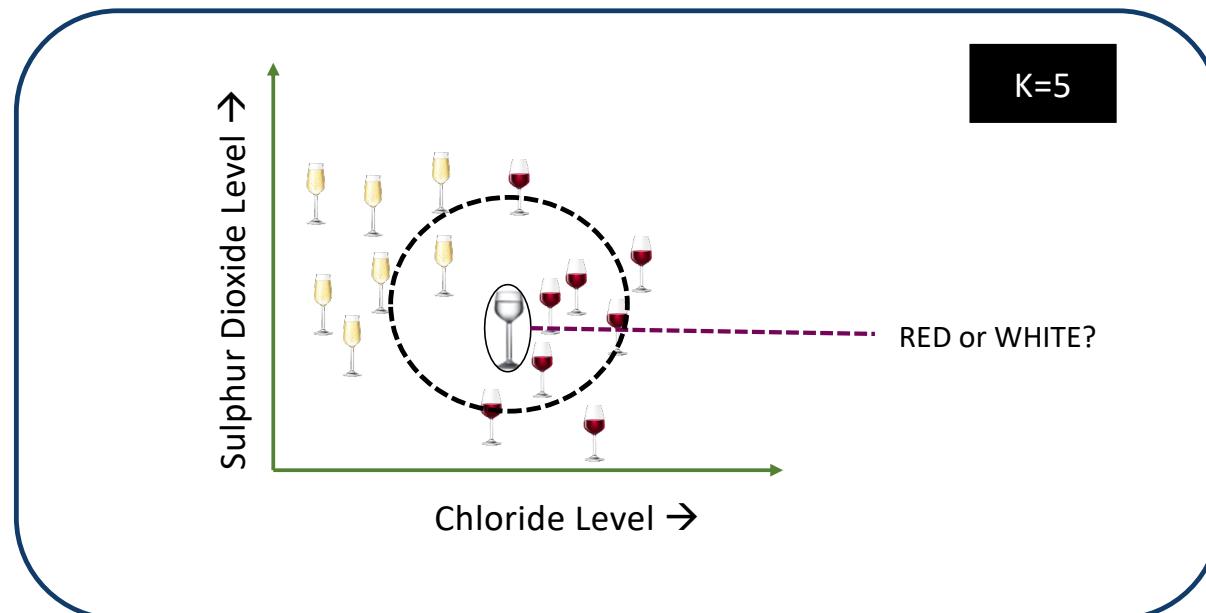
# What is KNN Algorithm?

But, what is K?



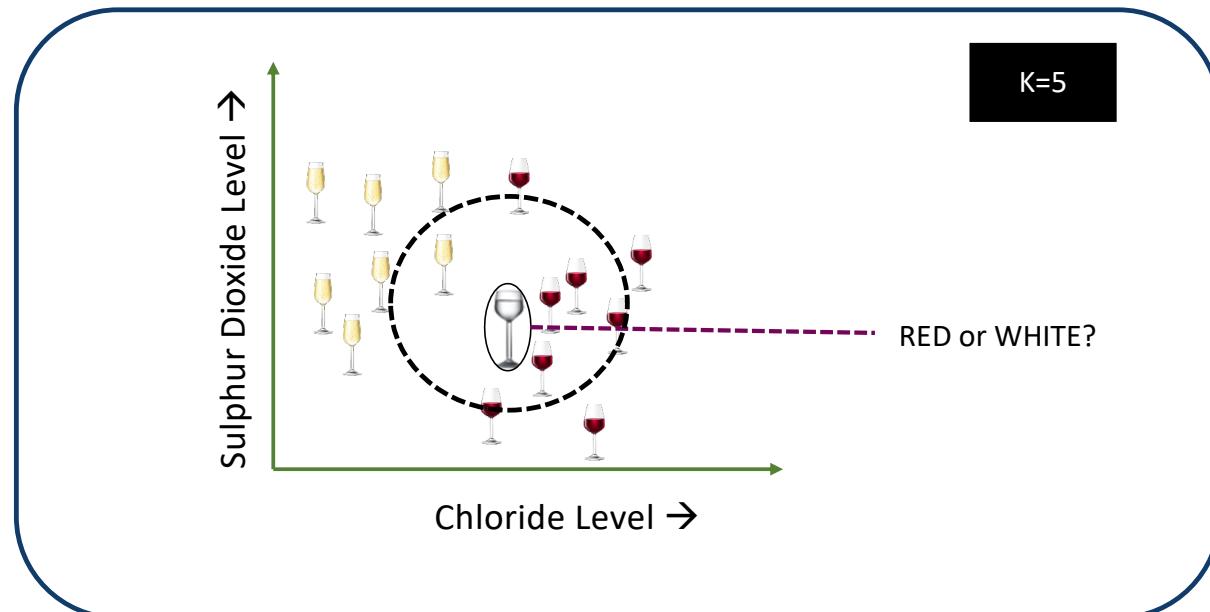
# What is KNN Algorithm?

*k* in **KNN** is a parameter that refers to the number of nearest neighbors to include in the majority voting process



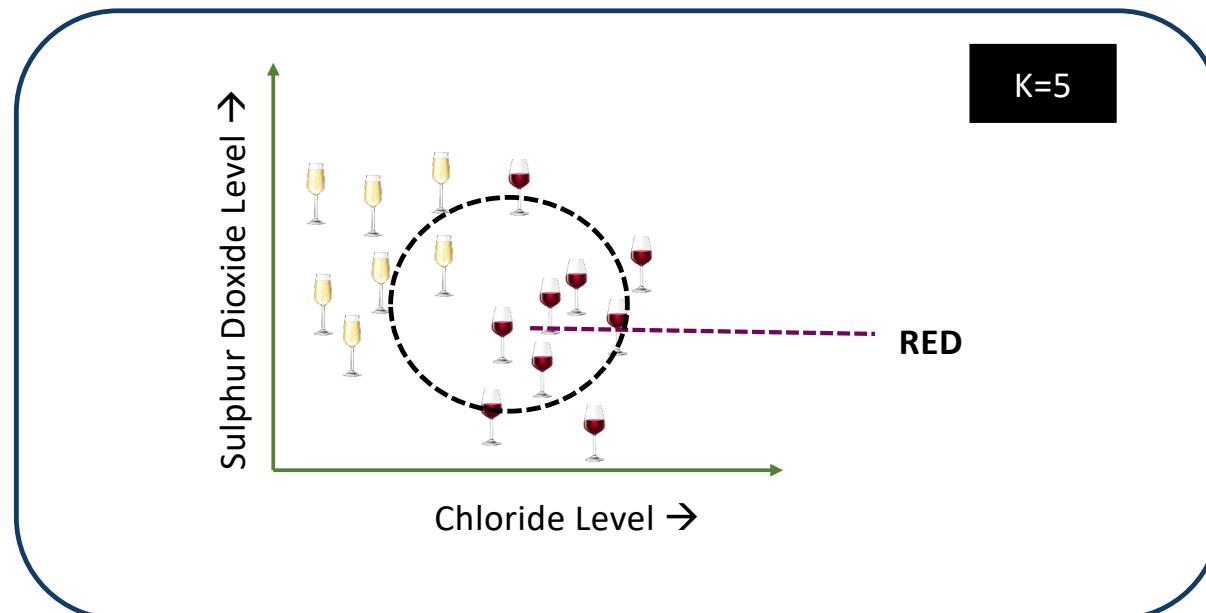
# What is KNN Algorithm?

A data point is classified by majority votes from its 5 nearest neighbors



# What is KNN Algorithm?

Here, the unknown point would be classified as red, since 4 out of 5 neighbors are red



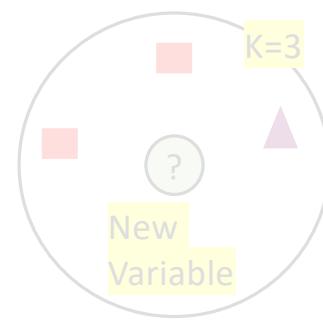


How do we  
choose K

## How do we choose the factor ‘k’?

---

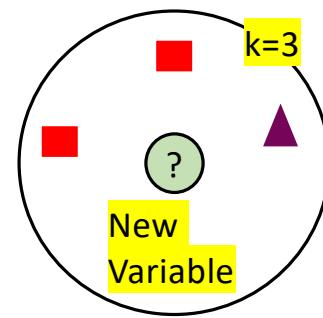
KNN Algorithm is based on **feature similarity**: Choosing the right value of  $k$  is a process called parameter tuning, and is important for better accuracy



## How do we choose the factor 'k'?

---

KNN Algorithm is based on **feature similarity**: Choosing the right value of  $k$  is a process called parameter tuning, and is important for better accuracy

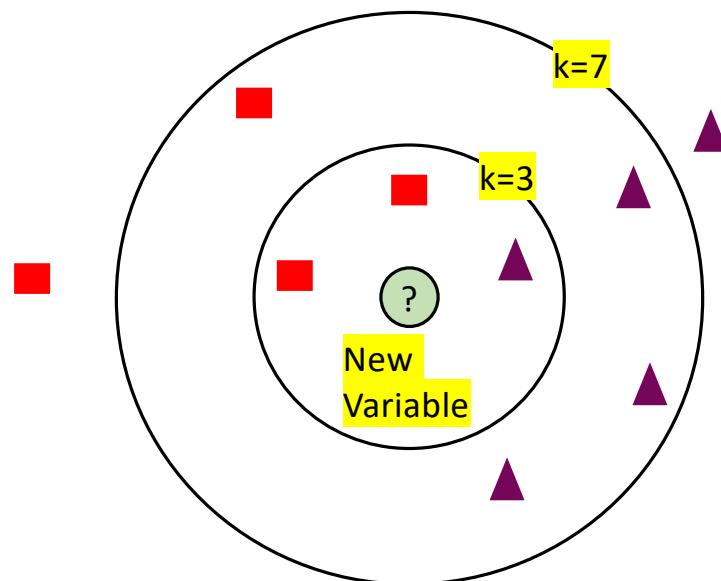


So at  $k=3$ , we can classify '?' as



## How do we choose the factor 'k'?

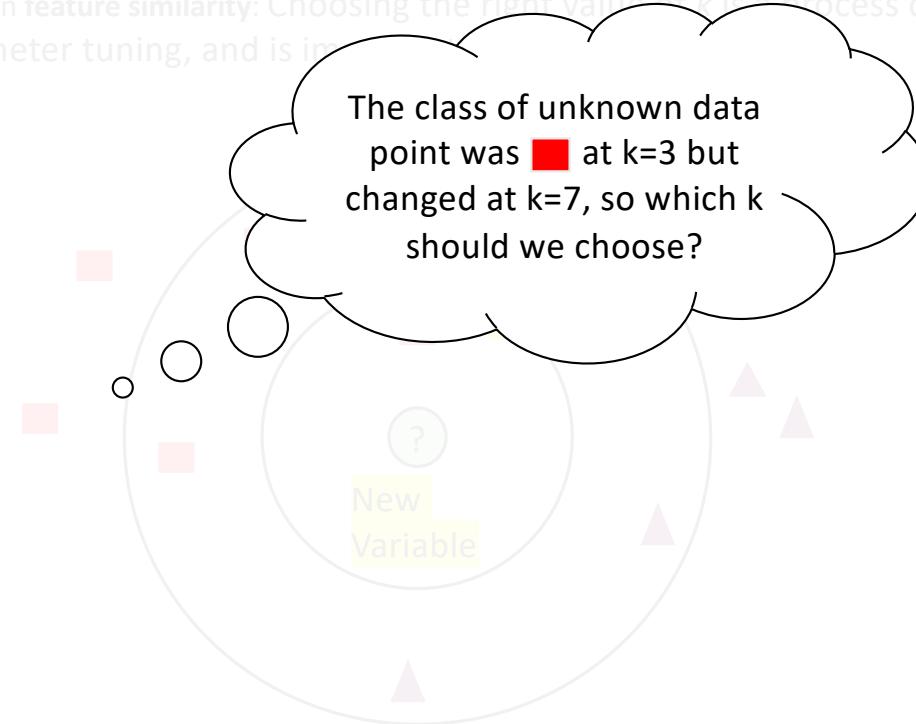
KNN Algorithm is based on **feature similarity**: Choosing the right value of  $k$  is a process called parameter tuning, and is important for better accuracy



But at  $k=7$ , we classify '?' as

## How do we choose the factor 'k'?

KNN Algorithm is based on feature similarity: Choosing the right value of  $k$  is a process called parameter tuning, and is important.



So at  $k=3$ , we can classify '?' as



---

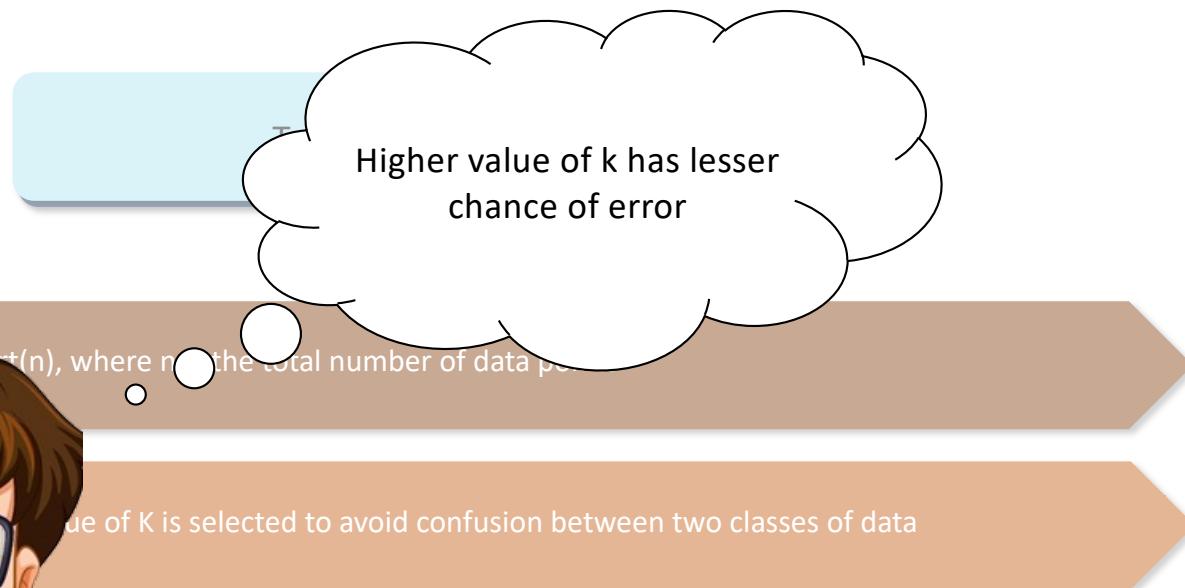
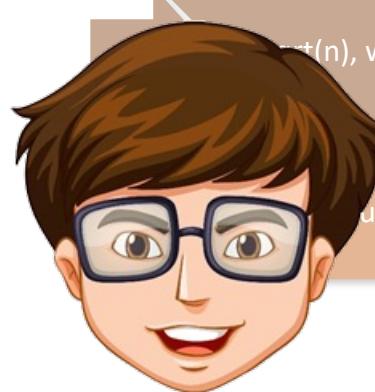
## How do we choose the factor 'k'?

---

To choose a value of k:

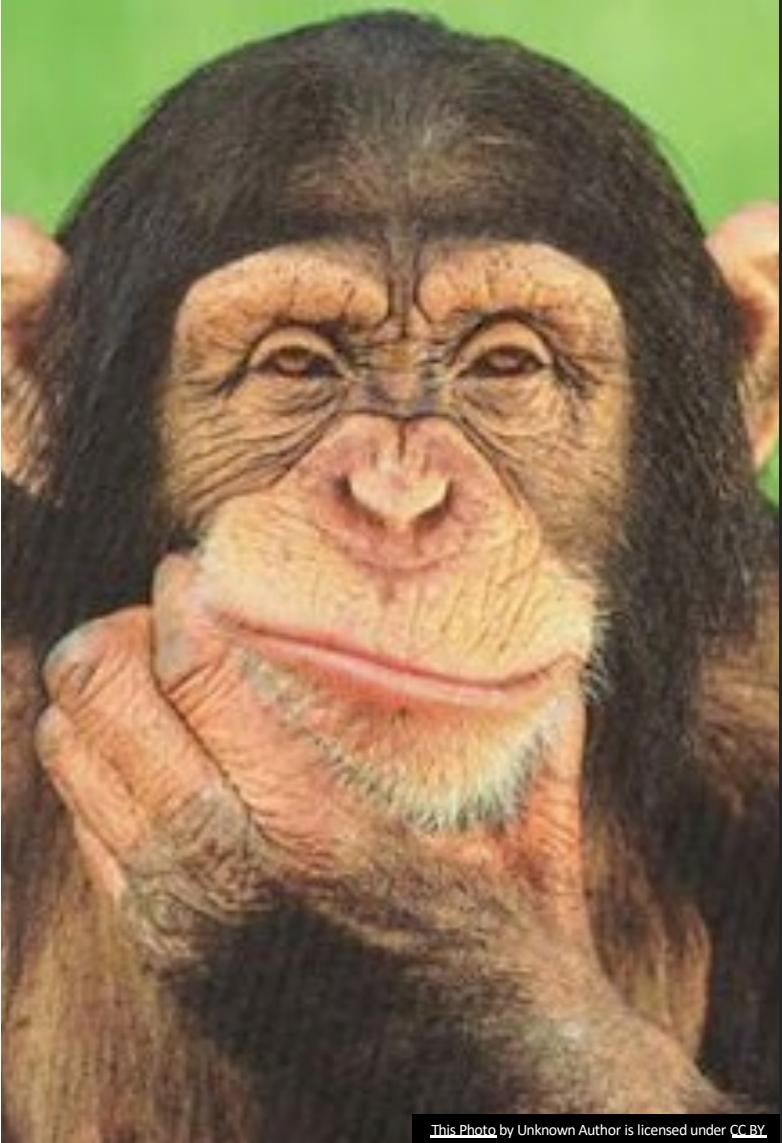
- $\text{Sqrt}(n)$ , where n is the total number of data points
- Odd value of K is selected to avoid confusion between two classes of data

## How do we choose the factor 'k'?



Value of K is selected such that  $K = \sqrt{n}$ , where n is the total number of data points.

Value of K is selected to avoid confusion between two classes of data



When do we use  
KNN?

This Photo by Unknown Author is licensed under CC BY

## When do we use KNN Algorithm?



We can use KNN when

Data is labeled



Dog

## When do we use KNN Algorithm?



We can use KNN when

Data is labeled



Dog

Data is noise free

Weight(x2)	Height(y2)	Class
51	167	Underweight
62	182	one-fourty
69	176	23
64	173	hello kitty
65	172	Normal

Noise

## When do we use KNN Algorithm?



We can use KNN when

Dataset is small



Because KNN is a 'lazy learner' i.e. doesn't learn a discriminative function from the training set

Data is labeled



Dog

Data is noise free

Weight(x2)	Height(y2)	Class
51	167	Underweight
62	182	one-fourty
69	176	23
64	173	hello kitty
65	172	Normal

Noise

---

## How does KNN Algorithm work?

## How does KNN Algorithm work?



Consider a dataset having two variables: height (cm) & weight (kg) and each point is classified as Normal or Underweight

Weight(x2)	Height(y2)	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 Algorithm work?



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

57 kg

170 cm

?



Assuming, we don't  
know how to  
calculate BMI!

---

## How does KNN Algorithm work?

---

To find the nearest neighbors, we will calculate Euclidean distance

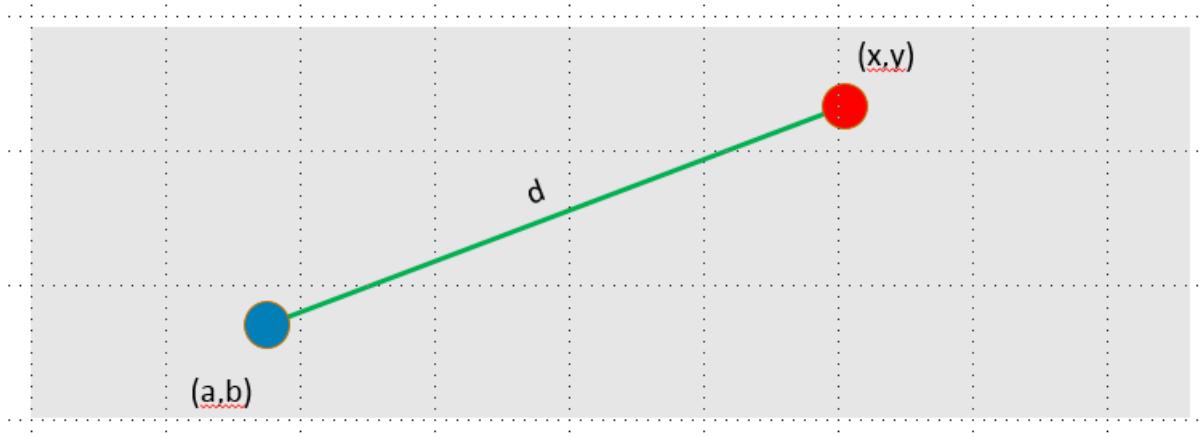


But, what is  
Euclidean distance?

## How does KNN Algorithm work?

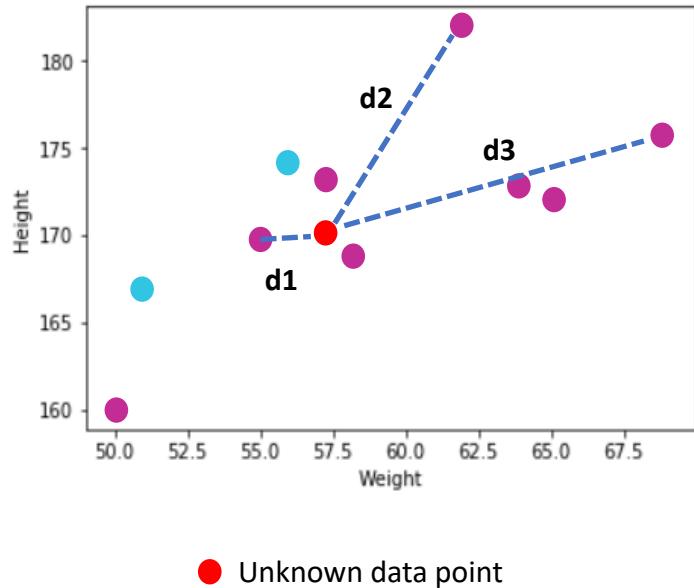
According to the **Euclidean distance** formula, the **distance** between two points in the plane with coordinates  $(x, y)$  and  $(a, b)$  is given by:

$$\text{dist}(d) = \sqrt{(x - a)^2 + (y - b)^2}$$



# How does KNN Algorithm work?

Let's calculate it to understand clearly:



$$\text{dist(d1)} = \sqrt{(170-167)^2 + (57-51)^2} \approx 6.7$$

$$\text{dist(d2)} = \sqrt{(170-182)^2 + (57-62)^2} \approx 13$$

$$\text{dist(d3)} = \sqrt{(170-176)^2 + (57-69)^2} \approx 13.4$$

Similarly, we will calculate Euclidean distance of unknown data point from all the points in the dataset

## How does KNN Algorithm work?

Hence, we have calculated the Euclidean distance of unknown data point from all the points as shown:

Where  $(x_1, y_1) = (57, 170)$  whose class we have to classify

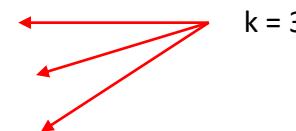
Weight(x2)	Height(y2)	Class	Euclidean Distance
51	167	Underweight	6.7
62	182	Normal	13
69	176	Normal	13.4
64	173	Normal	7.6
65	172	Normal	8.2
56	174	Underweight	4.1
58	169	Normal	1.4
57	173	Normal	3
55	170	Normal	2

## How does KNN Algorithm work?

Now, lets calculate the nearest neighbor at k=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

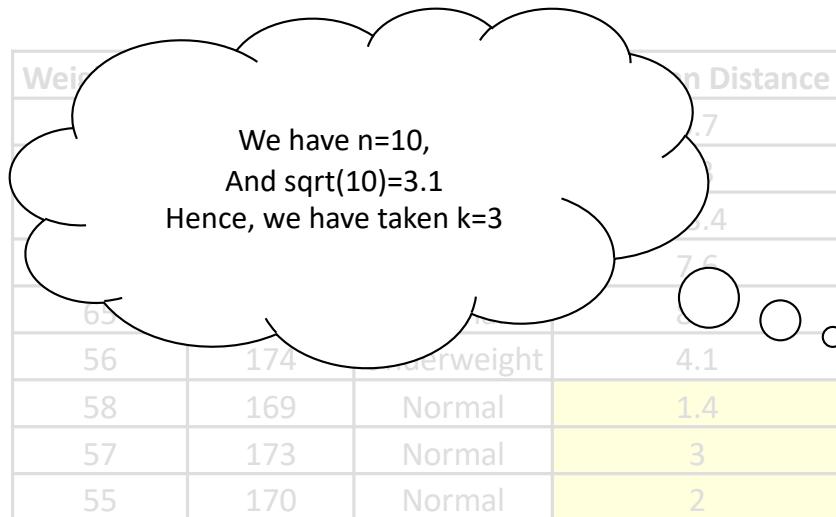
k = 3



57 kg	170 cm	?
-------	--------	---

# How does KNN Algorithm work?

Now, lets calculate the nearest neighbor at k=3



57 kg	170 cm	?
-------	--------	---

## How does KNN Algorithm work?



Class	Euclidean Distance
Underweight	6.7
Normal	13
Normal	13.4
Normal	7.6
Normal	8.2
Underweight	4.1
Normal	1.4
Normal	3
Normal	2

k = 3

Three red arrows point from the text "k = 3" to the three "Normal" entries in the table, specifically highlighting the rows with distances 1.4, 3, and 2.

So, majority neighbors are pointing towards 'Normal'

Hence, as per KNN algorithm the class of (57, 170) should be 'Normal'

---

## Recap of KNN

---



### Recap of KNN

- A positive integer  $k$  is specified, along with a new sample
- We select the  $k$  entries in our database which are closest to the new sample
- We find the most common classification of these entries
- This is the classification we give to the new sample