

Abstract Patterns and Features

AI Systems Implementation

George Langrudi
Based on slides by Gareth Howells

Can machines be intelligent?

- Is this intelligent?
 1. Yes!
 2. Not at the moment, but maybe when Machine Learning technology improves
 3. It'll only ever have a basic understanding of the world, it'll never compare to humans
 4. Don't be daft, it's made of plastic!
- Why?

The screenshot shows a dark-themed ChatGPT 3.5 interface. At the top, it says "ChatGPT 3.5". Below that, a user message from "You" asks, "Briefly, Can machines be intelligent?". The AI response from "ChatGPT" explains that machines can exhibit intelligence through AI, mentioning learning, reasoning, problem-solving, perception, and language understanding. It notes that current machine intelligence is narrow and task-specific, lacking general cognitive abilities. The AI concludes by stating that while advances in AI push boundaries, achieving human-level intelligence remains a topic of research.

ChatGPT 3.5

You

Briefly, Can machines be intelligent?

2 / 2

ChatGPT

Yes, machines can exhibit a form of intelligence through artificial intelligence (AI). AI involves the development of algorithms and systems that enable machines to perform tasks that typically require human intelligence, such as learning, reasoning, problem-solving, perception, and language understanding. However, the intelligence demonstrated by machines is currently narrow and task-specific, lacking the general cognitive abilities and understanding that humans possess. Advances in AI continue to push the boundaries of machine intelligence, but whether machines can achieve a level of intelligence comparable to human intelligence remains a topic of ongoing research and debate.

undo redo

Lecture Overview

- What are features?
- Choosing features
- Deep dive on Moments
 - What is a Moment?
 - How are they extracted?
- This week's class work

BUT FIRST!

- How would you rate your AI knowledge?
 - I have a good understanding of the theory
 - I understand the principles, but not the practical details
 - This all sounds confusing, I'm absolutely terrified, what have I signed up for?
 - Please stop asking us questions and get on with it...
- Fine...

Features

- What is a feature?
- In essence, values that describe an object, task, or domain
 - Abstract away the raw data
 - Reduce the *dimensionality* of the problem!

Features

- Features could be:
 - Raw, quantitative data
 - Numerical data, boolean values, vectors, even images
 - Direct features
 - Edge detection, detected circles/ellipses, Spectrograms
 - Abstract features
 - Region textures, *Moments*

Features

Garbage in, Garbage out

- Problems with data = problems with classification
 - Insufficient training data
 - Unrepresentative training data
 - Irrelevant features
 - Poor quality data
- Aim for high quality, representative data, with relevant features
 - Easier said than done...

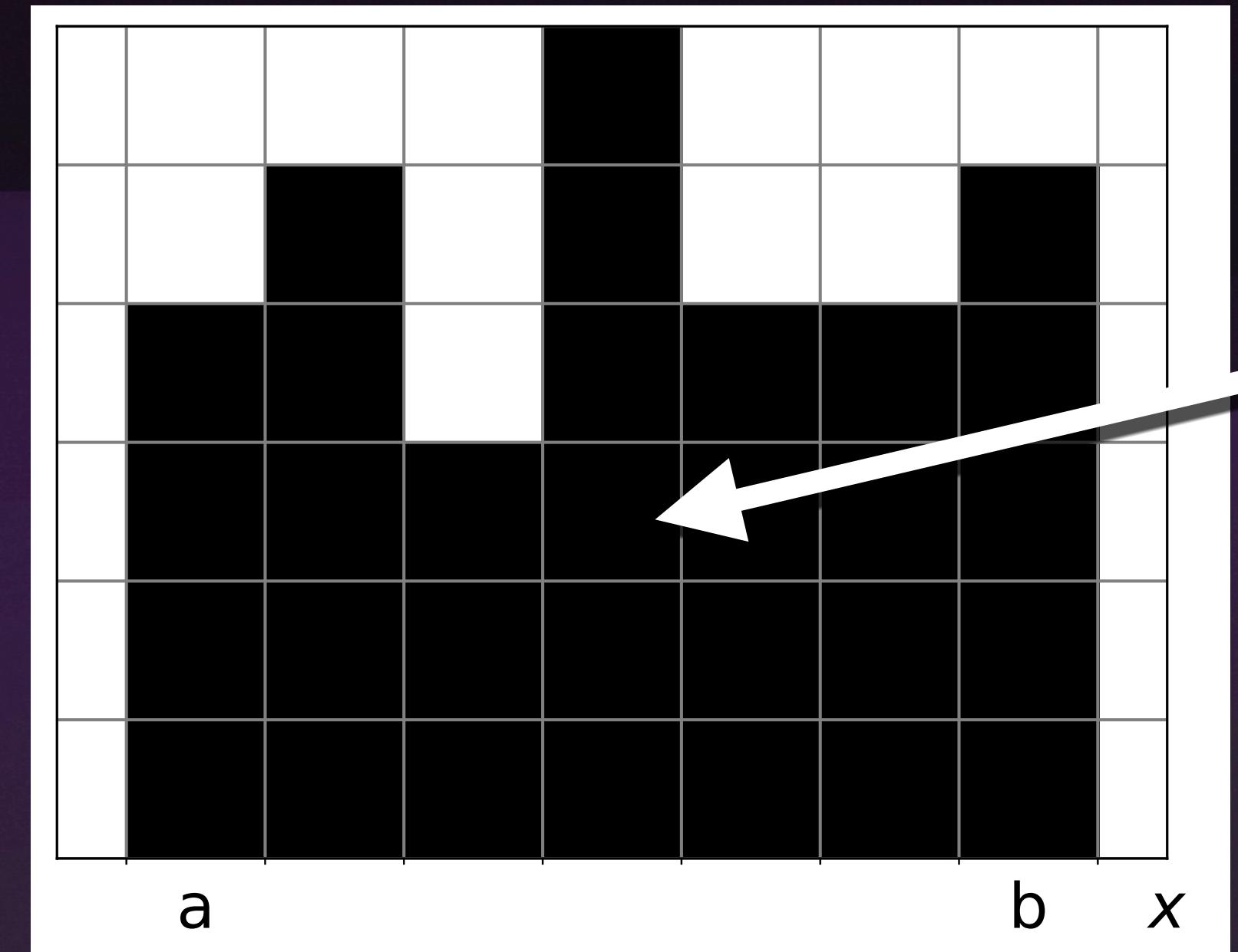
Moments

- Moments are an abstract feature that characterises the shape of objects and regions
- The $(p, q)^{th}$ moment of an image $f(x, y)$ is given by:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx . dy$$

- Where $p, q = 0, 1, 2, \dots, \infty$
- $f(x, y)$ is the image intensity (value) as point (x, y)

OK, that looked horrible



$$\int_a^b f(x)dx = \text{Area}$$

- If x is made from discrete intervals, then $\int_a^b f(x)dx$ becomes $\sum_a^b f(x)$
- Because the area is the sum of each column ($= 4 + 5 + 3 + 6 + 4 + 4 + 5 = 31$ in this case)

Order of a Moment

- The order of a moment is the sum of p and q : $(p + q)$
- Therefore there is just one zero order moment:
 - m_{00}
- Two first order moments
 - m_{10}, m_{01}
- Three second order moments
 - m_{20}, m_{02}, m_{11}
- And so on...

Calculating Moments

- Assume a binary image
 - Each pixel is either 0 (white) or 1 (black)
- Moment:
$$m_{pq} = \sum_x \sum_y x^p y^q \cdot f(x, y)$$
- But $f(x, y)$ is either 0 or 1
- If $f(x, y)$ is 0, then $x^p y^q \cdot f(x, y) = 0$
- If $f(x, y)$ is 1, then $x^p y^q \cdot f(x, y) = x^p y^q$

Calculating Moments

- Therefore, for a binary image, $m_{pq} = \sum_x \sum_y x^p y^q$
 - Where \sum is taken over all black elements
 - m_{00} = Total number of black pixels in the image
 - Since x^0 is always 1 and y^0 is always 1 and $1 \times 1 = 1$
 - $m_{10} = \sum x$ (the sum of all the x coordinates of the black pixels)
 - Since x^1 is equal to x and y^0 is always 1 and $x \times 1 = x$
 - $m_{01} = \sum y$ (the same of all the y coordinates of the black pixels)
 - Since x^0 is always 1 and y^1 is equal to y and $1 \times y = y$

Moments

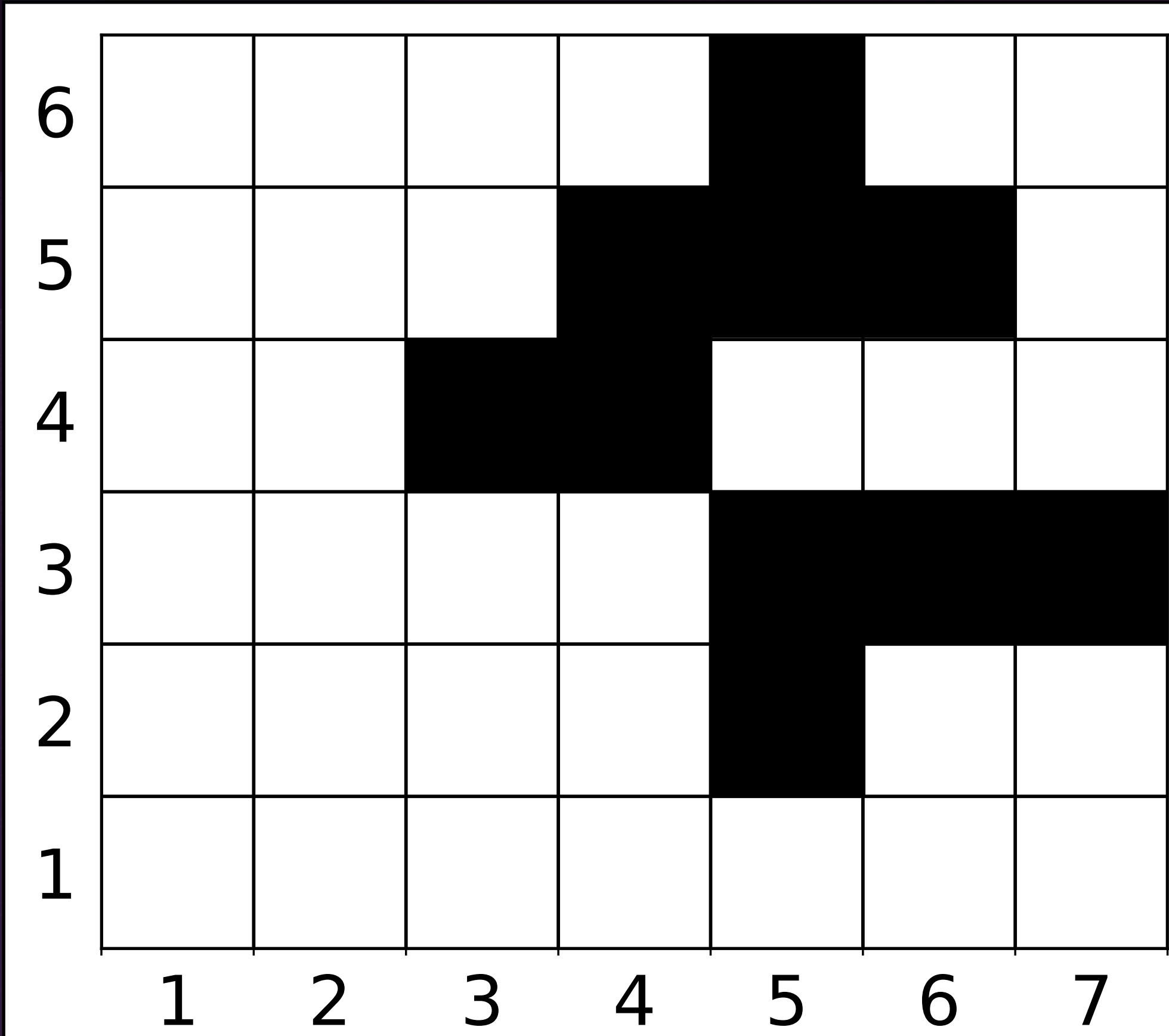
So, to recap...

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx . dy$$

- m is shorthand for moment
 - m_{pq} means the pq^{th} moment
 - $f(x, y)$ means the intensity of the pixel at (x, y)
 - For a binary image, we can simplify to:
- $$m_{pq} = \sum_x \sum_y x^p y^q$$

Example

- A binary image has black pixels at the following coordinates



x	3	7	4	5	5	6	5	6	5	4
y	4	3	5	2	3	5	6	3	5	4

Example

x	3	7	4	5	5	6	5	6	5	4
y	4	3	5	2	3	5	6	3	5	4

$$m_{pq} = \sum_x \sum_y x^p y^q \quad m_{11} = \sum_x \sum_y xy$$

$$\begin{aligned} \bullet &= (3 \times 4) + (7 \times 3) + (4 \times 5) + (5 \times 2) + (5 \times 3) \\ &+ (6 \times 5) + (5 \times 6) + (6 \times 3) + (5 \times 5) + (4 \times 4) = 197 \end{aligned}$$

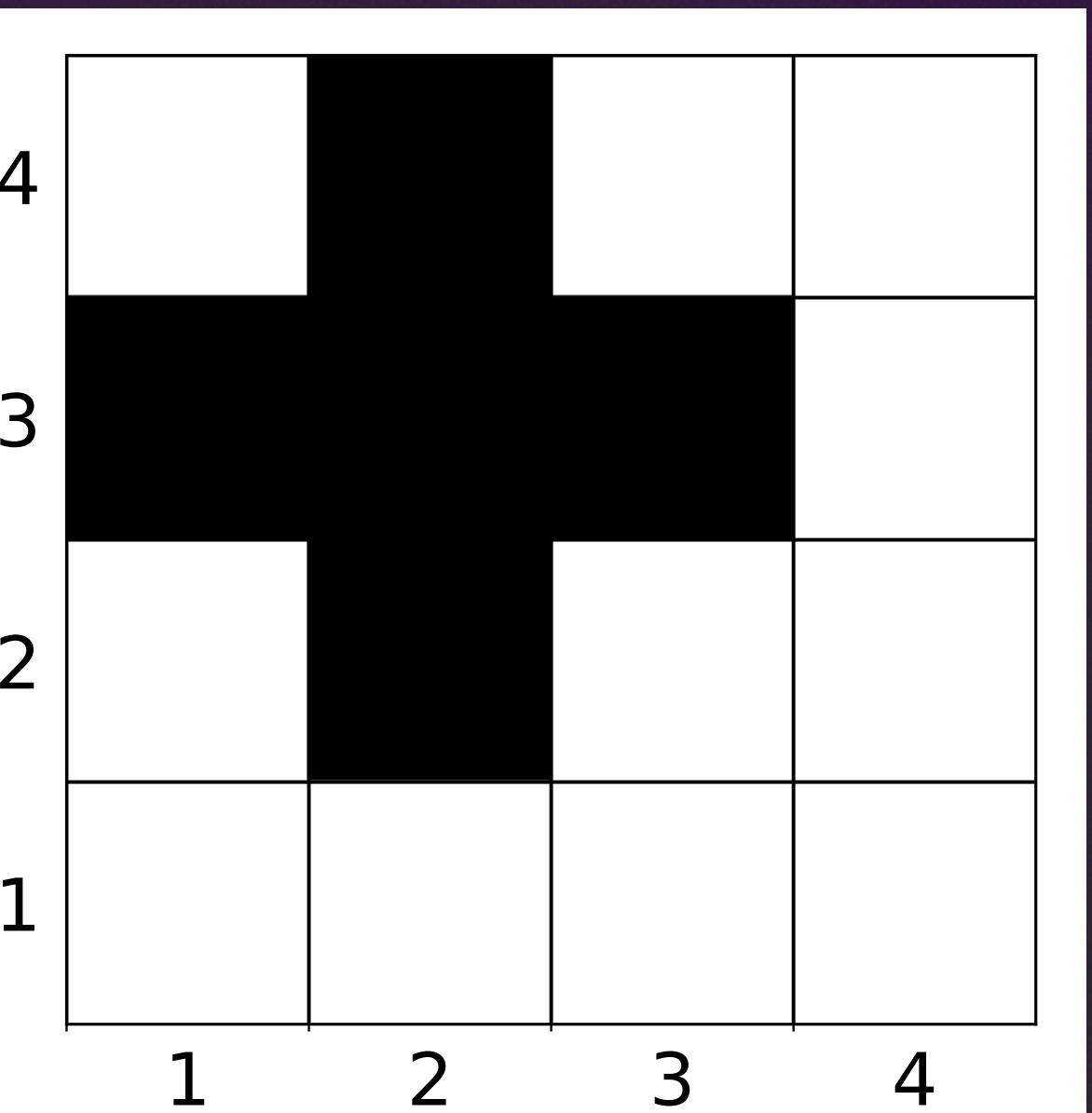
Example

Another One

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y)$$

$$\begin{aligned} m_{20} &= ((1^2 \times 3^0) + (2^2 \times 2^0) + (2^2 \times 3^0) + (2^2 \times 4^0) + (3^2 \times 3^0)) \\ &= ((1^2 \times 1) + (2^2 \times 1) + (2^2 \times 1) + (2^2 \times 1) + (3^2 \times 1)) \\ &= (1^2 + 2^2 + 2^2 + 2^2 + 3^2) \\ &= (1 + 4 + 4 + 4 + 9) \\ &= 22 \end{aligned}$$

$$\begin{aligned} m_{22} &= ((1^2 \times 3^2) + (2^2 \times 2^2) + (2^2 \times 3^2) + (2^2 \times 4^2) + (3^2 \times 3^2)) \\ &= ((1 \times 9) + (4 \times 4) + (4 \times 9) + (4 \times 16) + (9 \times 9)) \\ &= (9 + 16 + 36 + 64 + 81) \\ &= 206 \end{aligned}$$



Normalised Example

Un-Normalised

- A: $m_{20} = (1^2 + 2^2 + 2^2 + 2^2 + 3^2)$
 $= 22$

- B: $m_{20} = (2^2 + 3^2 + 3^2 + 3^2 + 4^2)$
 $= 47$

- Same shapes, but different values!

A:

4				
3				
2				
1				
	1	2	3	4

B:

4				
3				
2				
1				
	1	2	3	4

Normalised Moments

- Currently, the moments we extract are relative to the location of an object...
- But what if we don't always know where the object will be in an image?

$$M_{pq} = \sum (x - \bar{x})^p (y - \bar{y})^q$$

- Where \bar{x} is the average x coordinate of all black pixels, and \bar{y} is the average y coordinate of black pixels

Normalised Example

Normalised

$$M_{20} = ((1 - 2)^2 + (2 - 2)^2 + (2 - 2)^2 + (2 - 2)^2 + (3 - 2)^2)$$

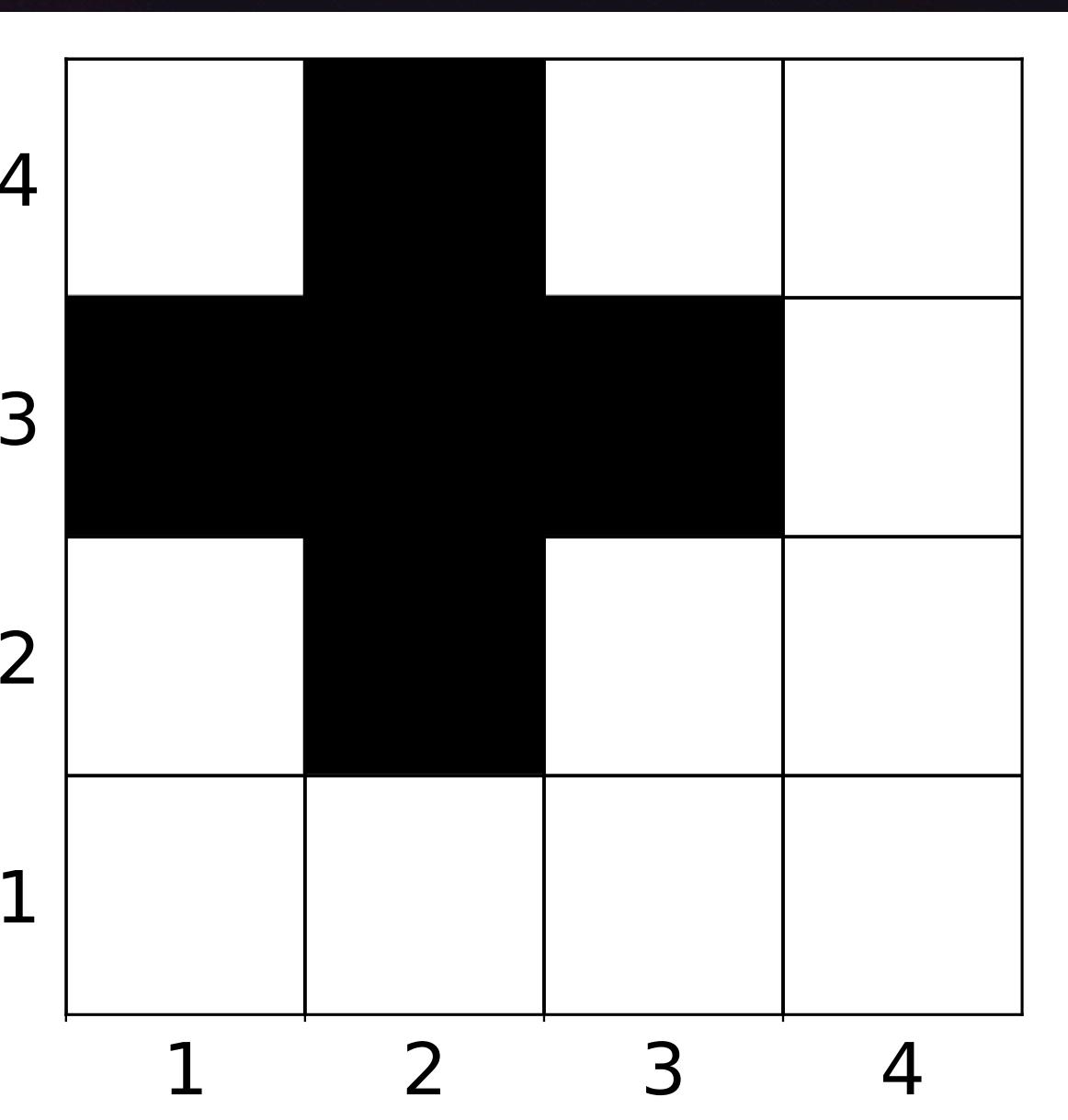
- A:
= $(1 + 0 + 0 + 0 + 1)$
= 2

$$M_{20} = ((2 - 3)^2 + (3 - 3)^2 + (3 - 3)^2 + (3 - 3)^2 + (4 - 3)^2)$$

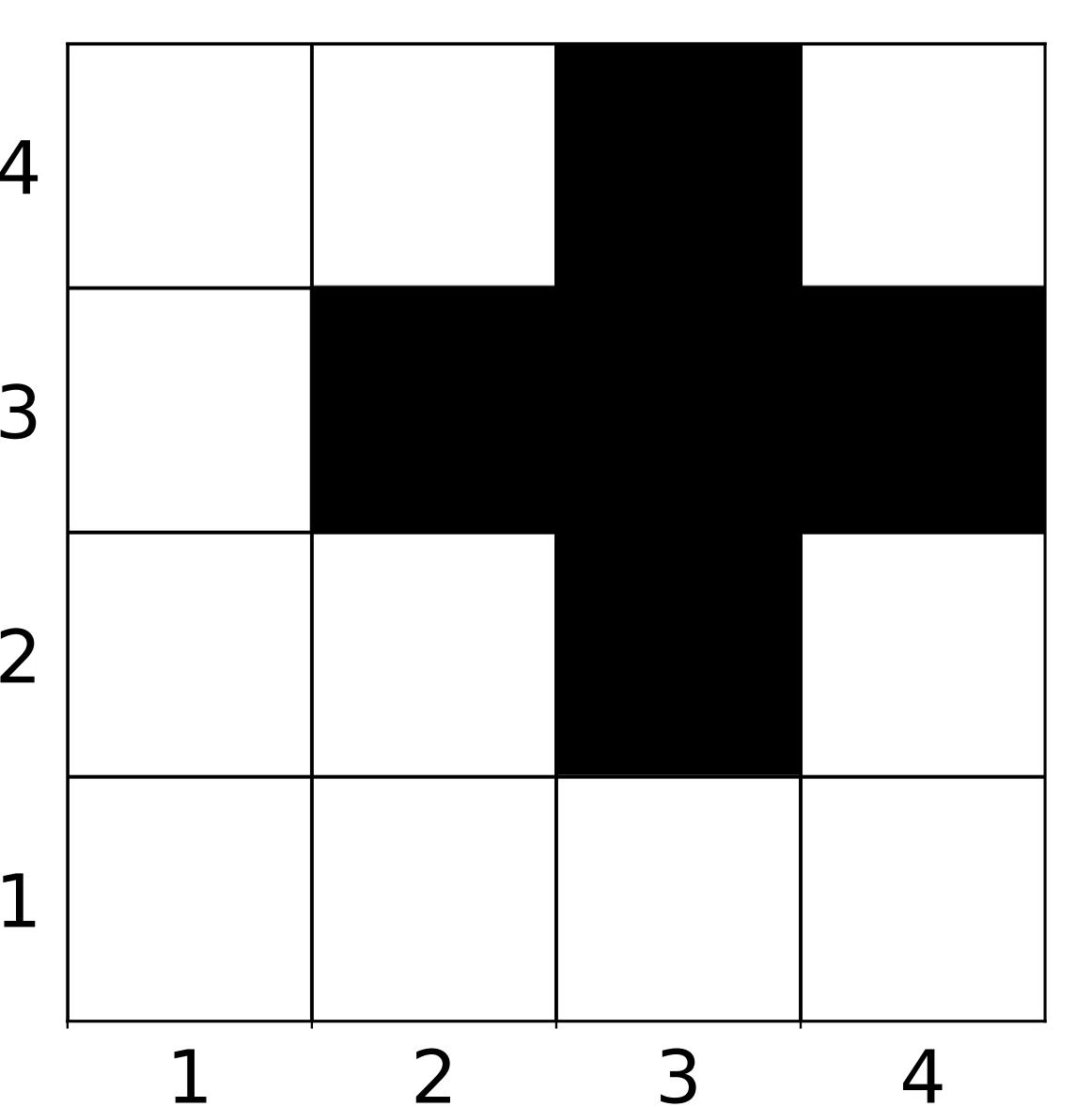
- B:
= $(1 + 0 + 0 + 0 + 1)$
= 2

- Same shape, different locations = Same value!

A:



B:



Recap

- This: $m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx . dy$
- Becomes this: $m_{pq} = \sum_x \sum_y x^p y^q f(x, y)$
- And to normalise it, use this: $M_{pq} = \sum (x - \bar{x})^p (y - \bar{y})^q$
- Now... your turn to implement it!

Up next

- Classes on Thursday/Friday
 - Implementing Moments in Python (Jupyter), and performing classification with them.
- Next week's lecture
 - Classification!