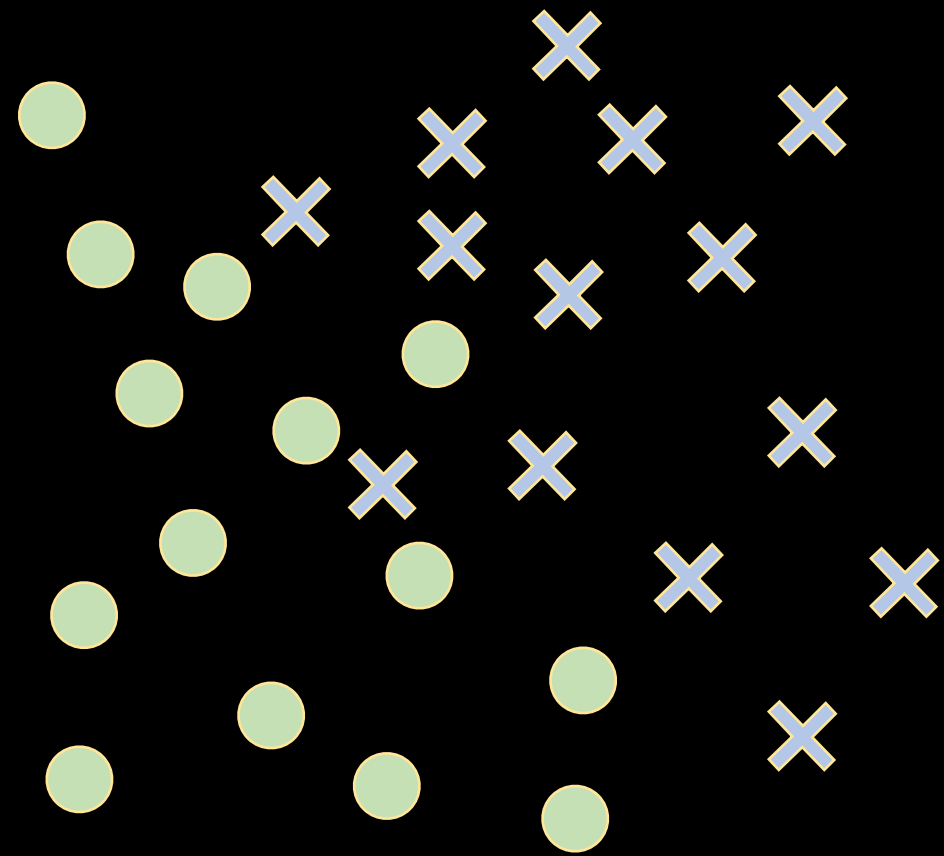
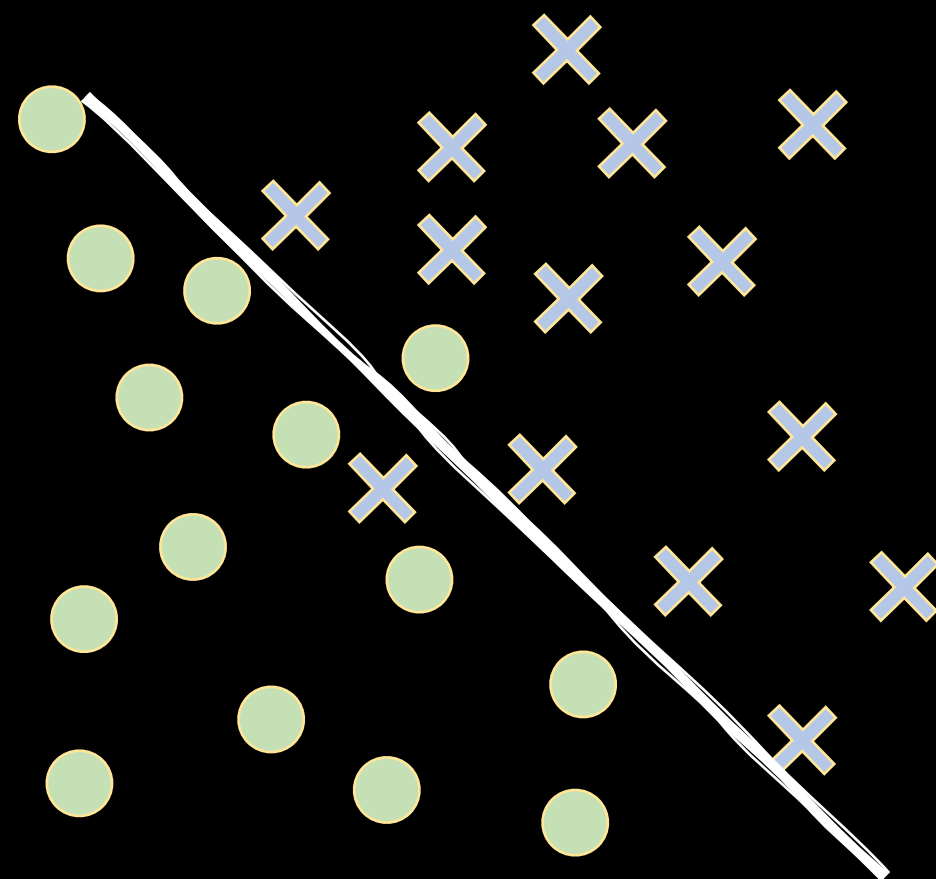


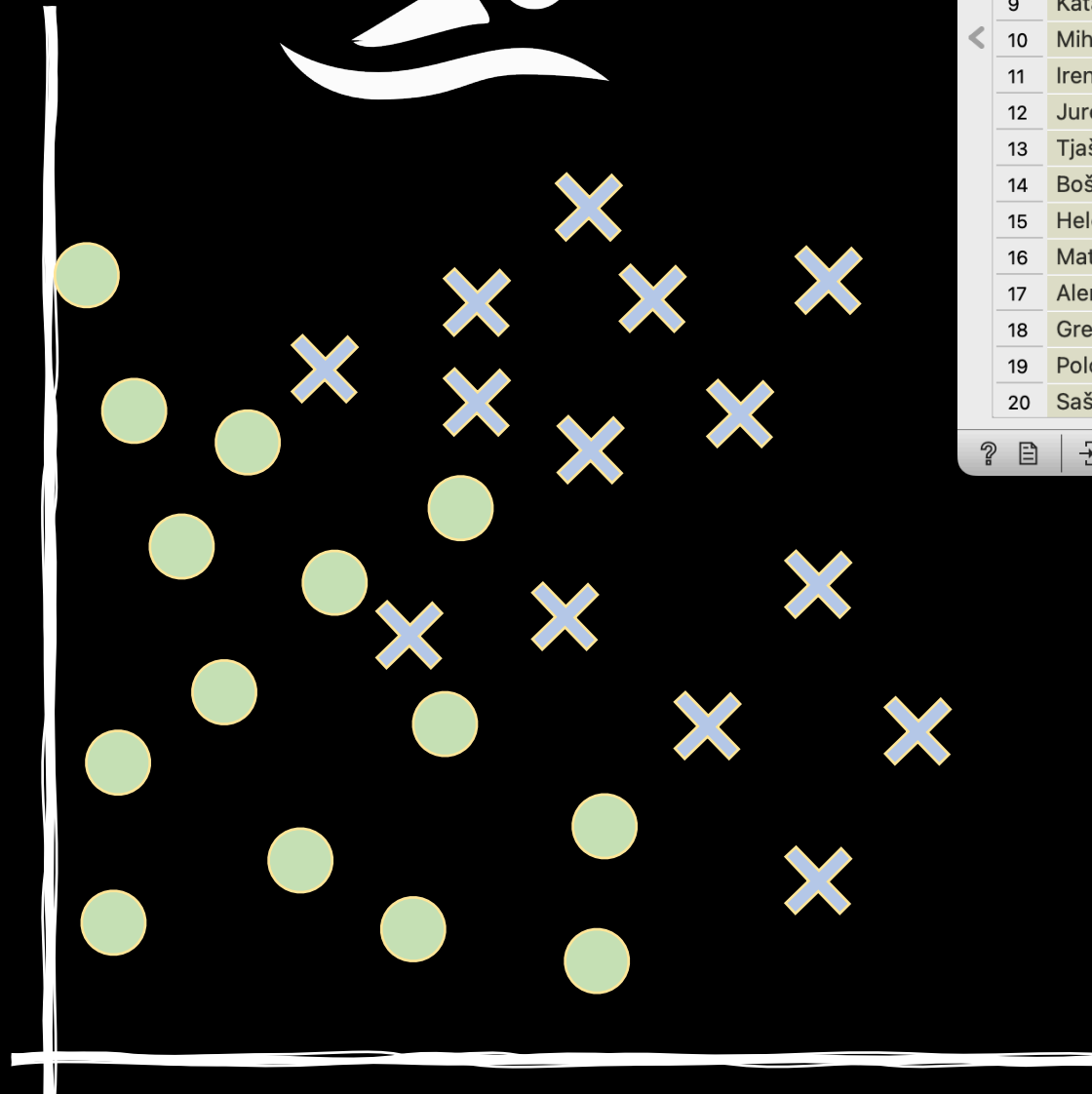






Data Table					
	name	exercise	sleep	activity	
1	Ana	2	8	bathing	
2	Marko	4	7	bathing	
3	Petra	3	6	bathing	
4	Janez	5	6	bathing	
5	Maja	9	6	bathing	
6	Andrej	5	5	bathing	
7	Nika	2	5	bathing	
8	Luka	5	4	bathing	
9	Katarina	4	3	bathing	
10	Miha	6	4	bathing	
11	Irena	9	3	bathing	
12	Jure	8	4	bathing	
13	Tjaša	11	3	bathing	
14	Boštjan	7	5	bathing	
15	Helena	4	9	swimming	
16	Matej	4	8	swimming	
17	Alenka	7	8	swimming	
18	Gregor	10	9	swimming	
19	Polona	9	8	swimming	
20	Sašo	10	7	swimming	

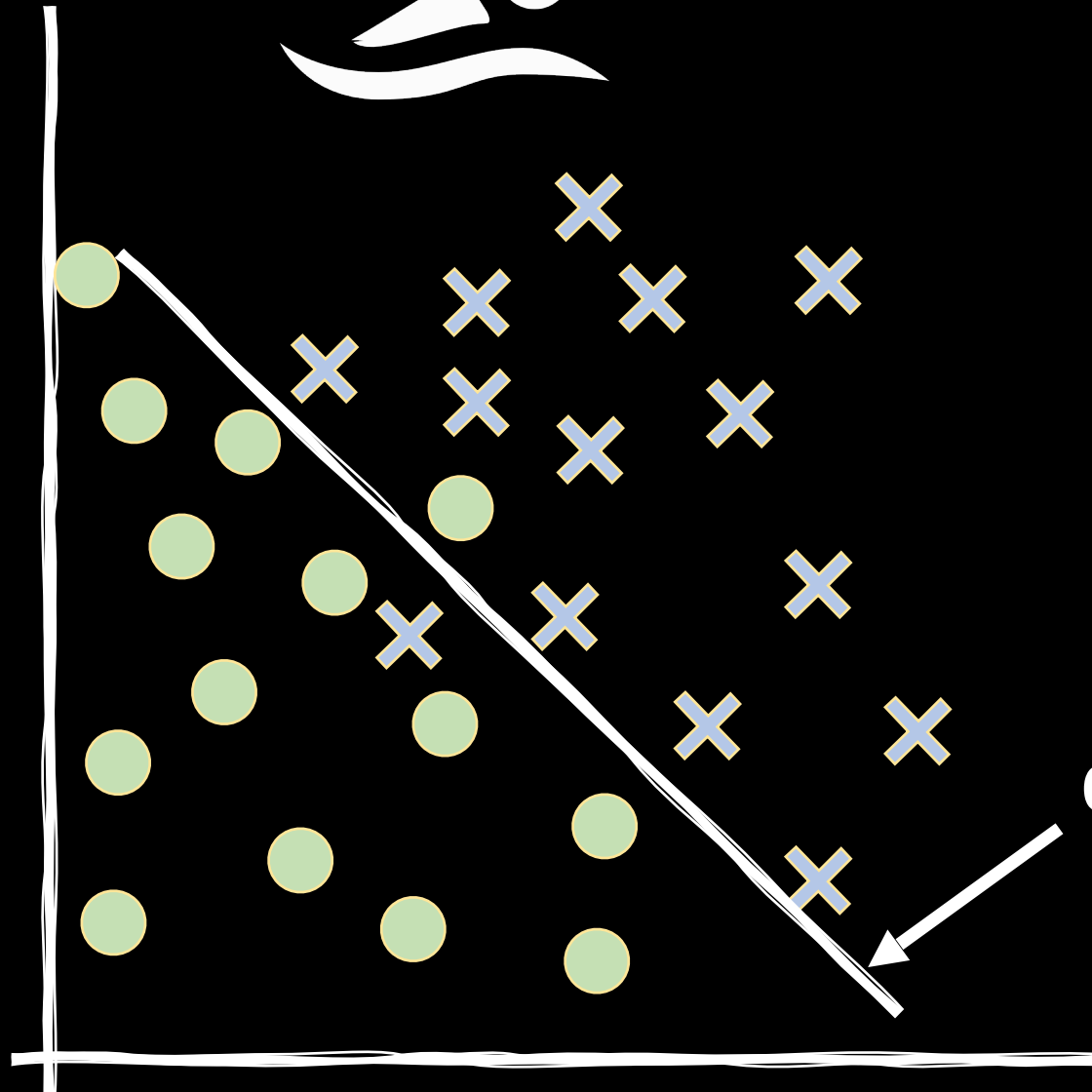
sleep



exercise

	name	exercise	sleep	activity
1	Ana	2	8	bathing
2	Marko	4	7	bathing
3	Petra	3	6	bathing
4	Janez	5	6	bathing
5	Maja	9	6	bathing
6	Andrej	5	5	bathing
7	Nika	2	5	bathing
8	Luka	5	4	bathing
9	Katarina	4	3	bathing
10	Miha	6	4	bathing
11	Irena	9	3	bathing
12	Jure	8	4	bathing
13	Tjaša	11	3	bathing
14	Boštjan	7	5	bathing
15	Helena	4	9	swimming
16	Matej	4	8	swimming
17	Alenka	7	8	swimming
18	Gregor	10	9	swimming
19	Polona	9	8	swimming
20	Sašo	10	7	swimming

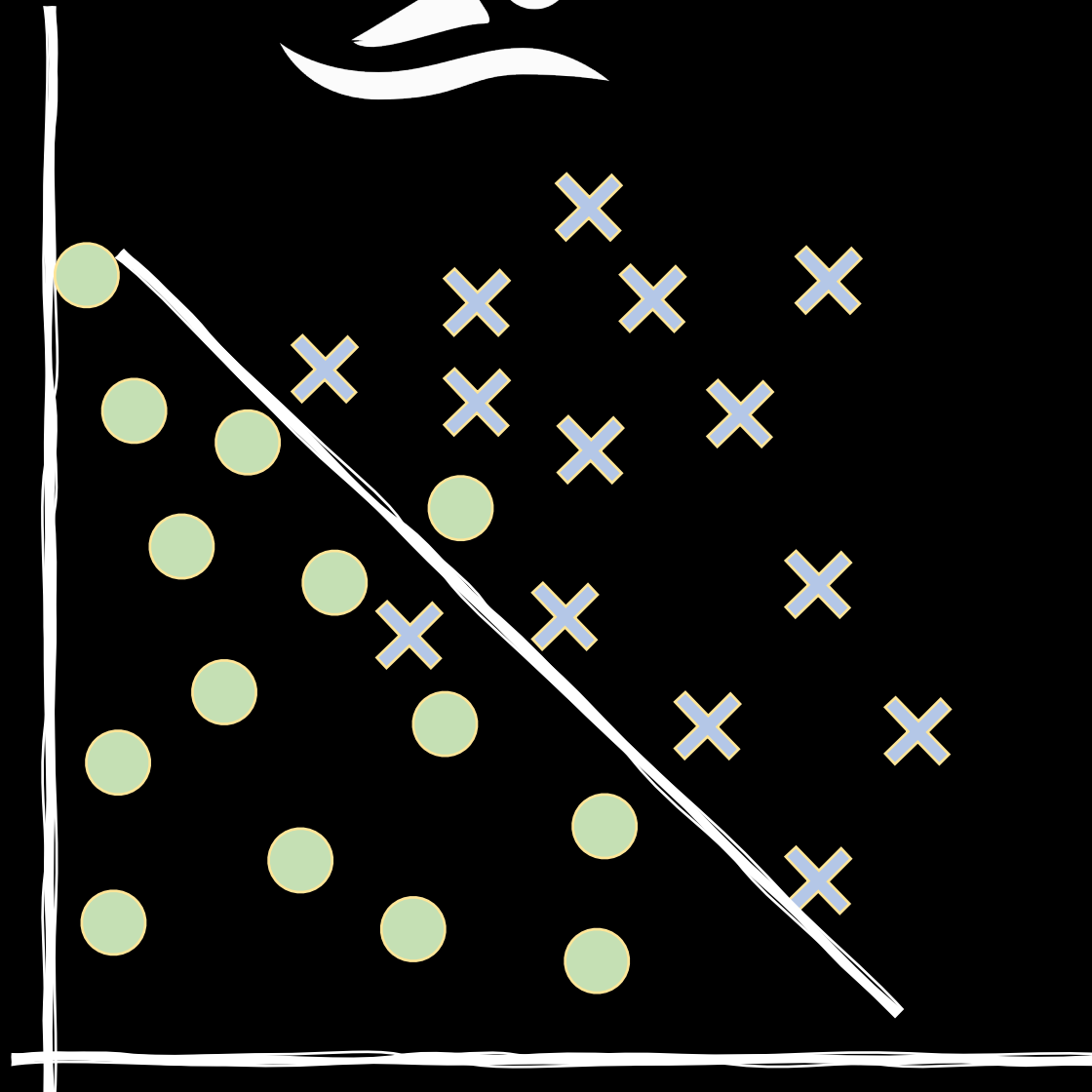
sleep



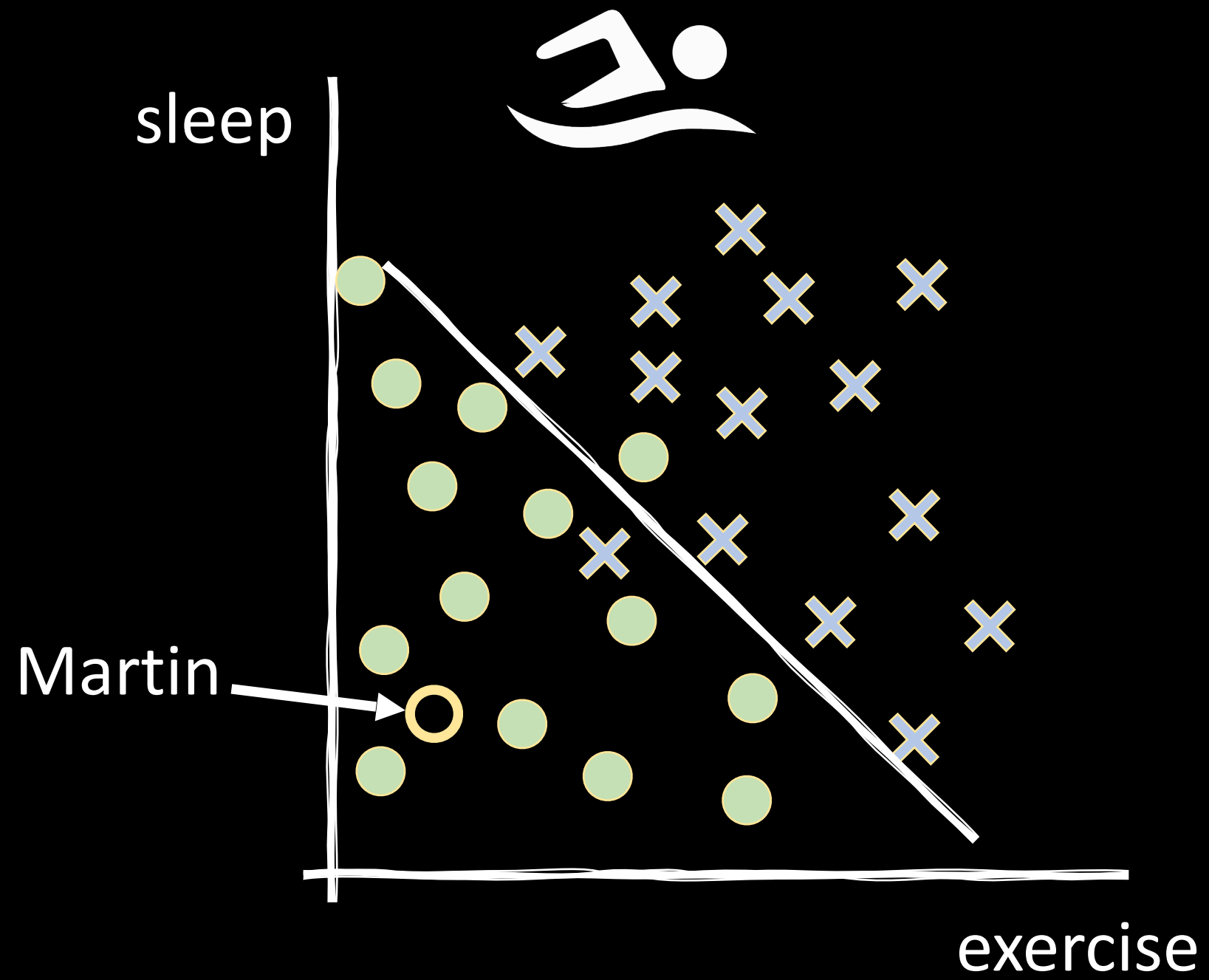
decision boundary

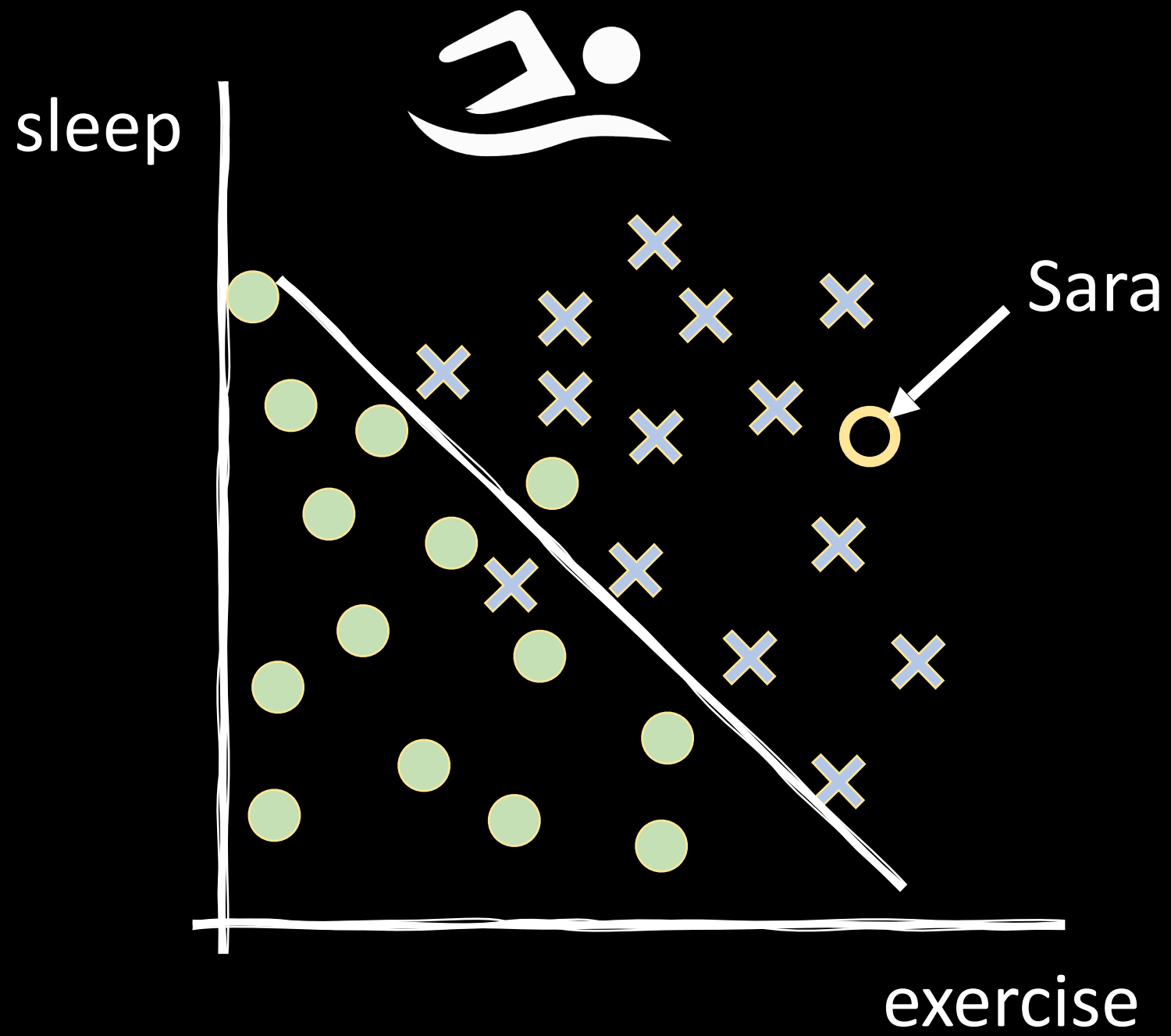
exercise

sleep

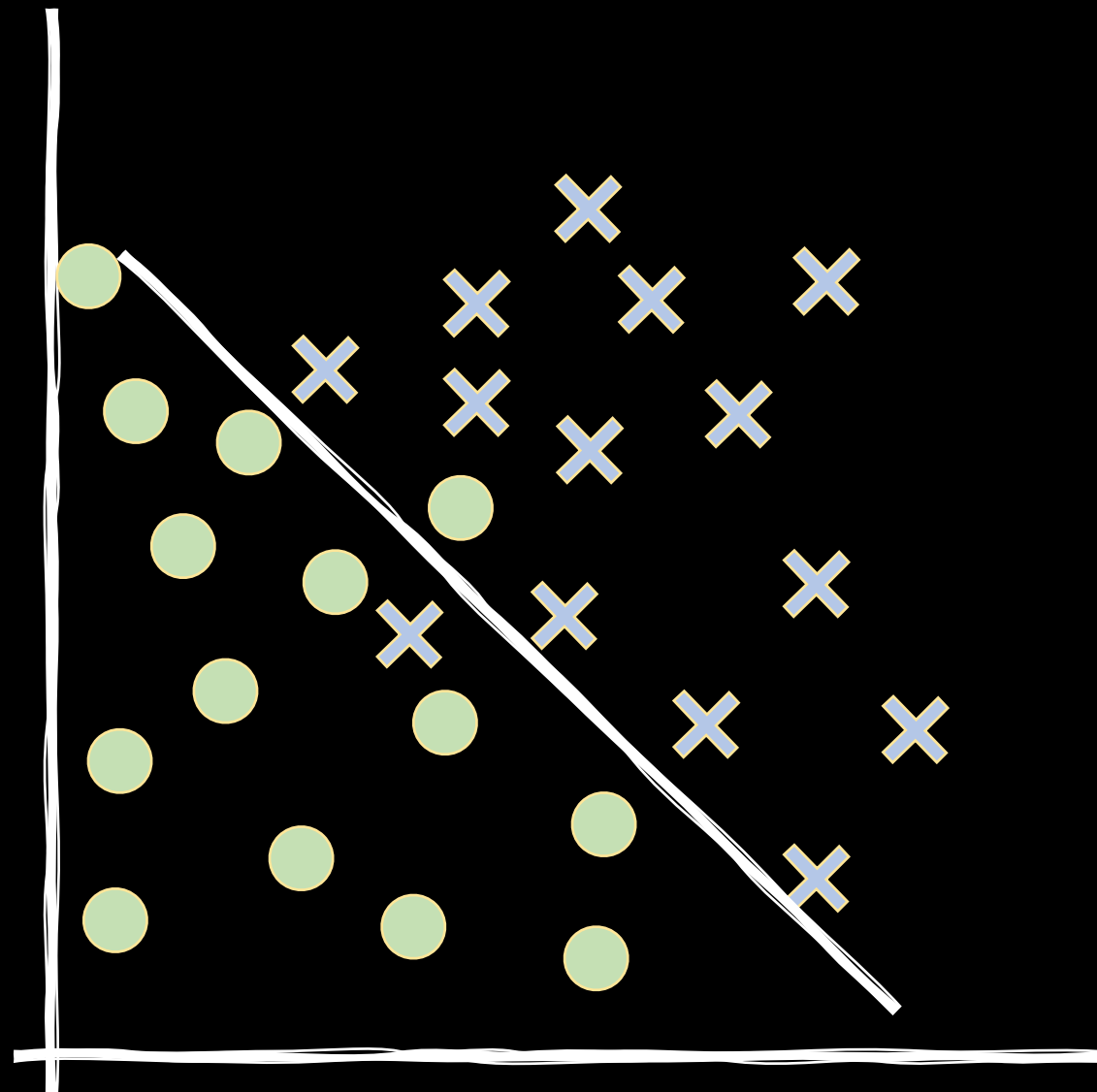


exercise





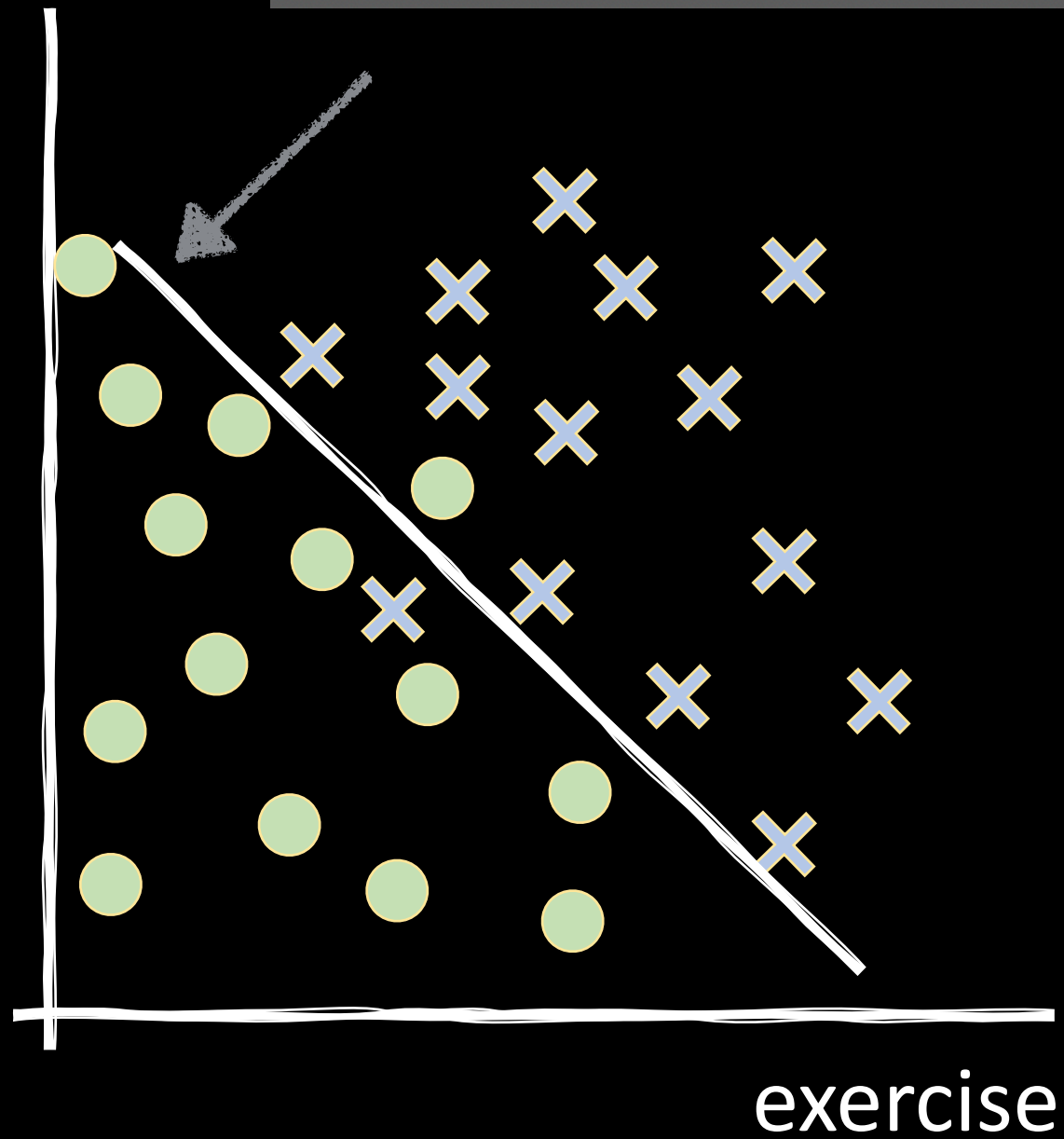
sleep



exercise

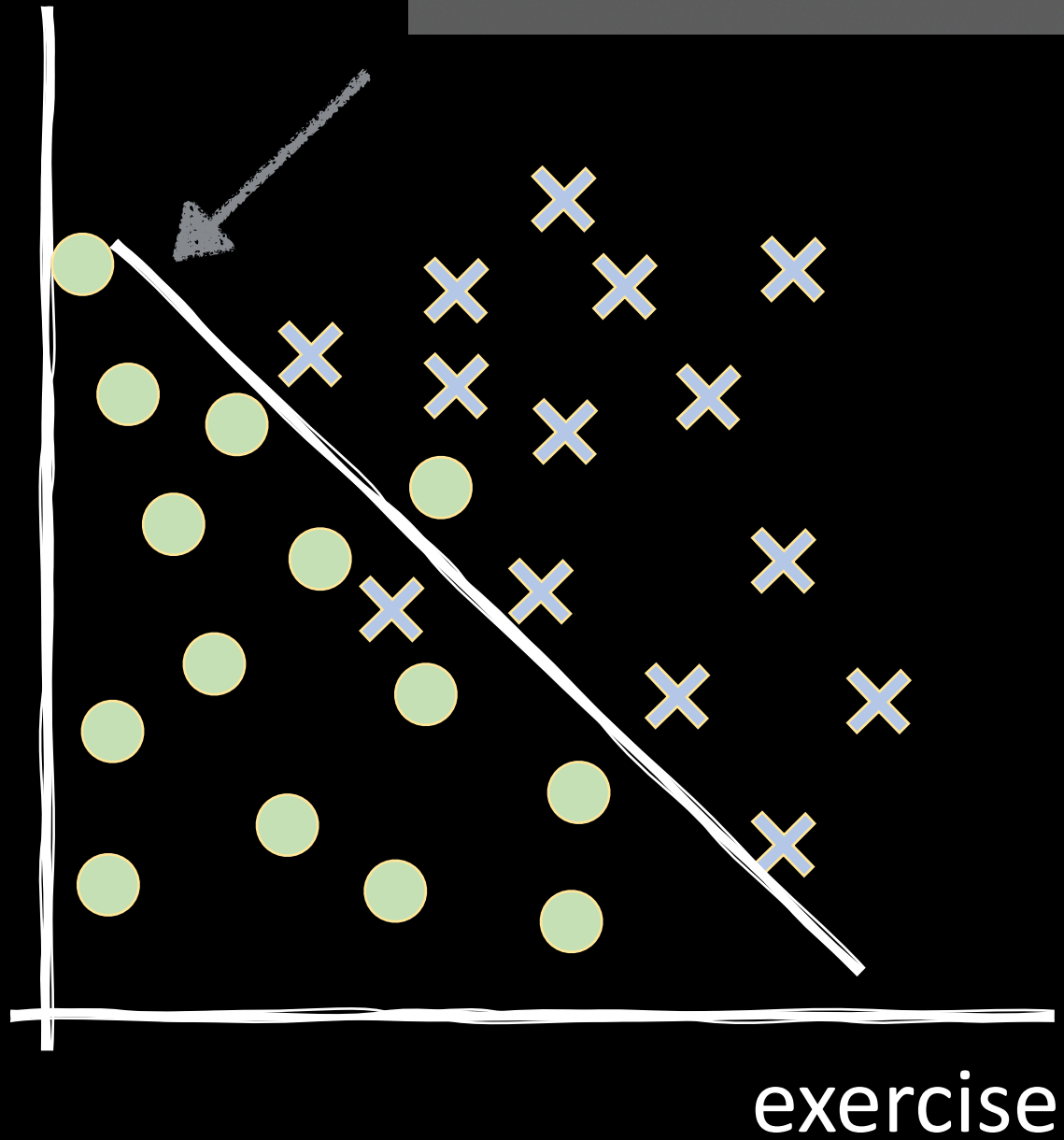
sleep

$$w_0 + w_1 \times \text{exercise} + w_2 \times \text{sleep} = 0$$



sleep

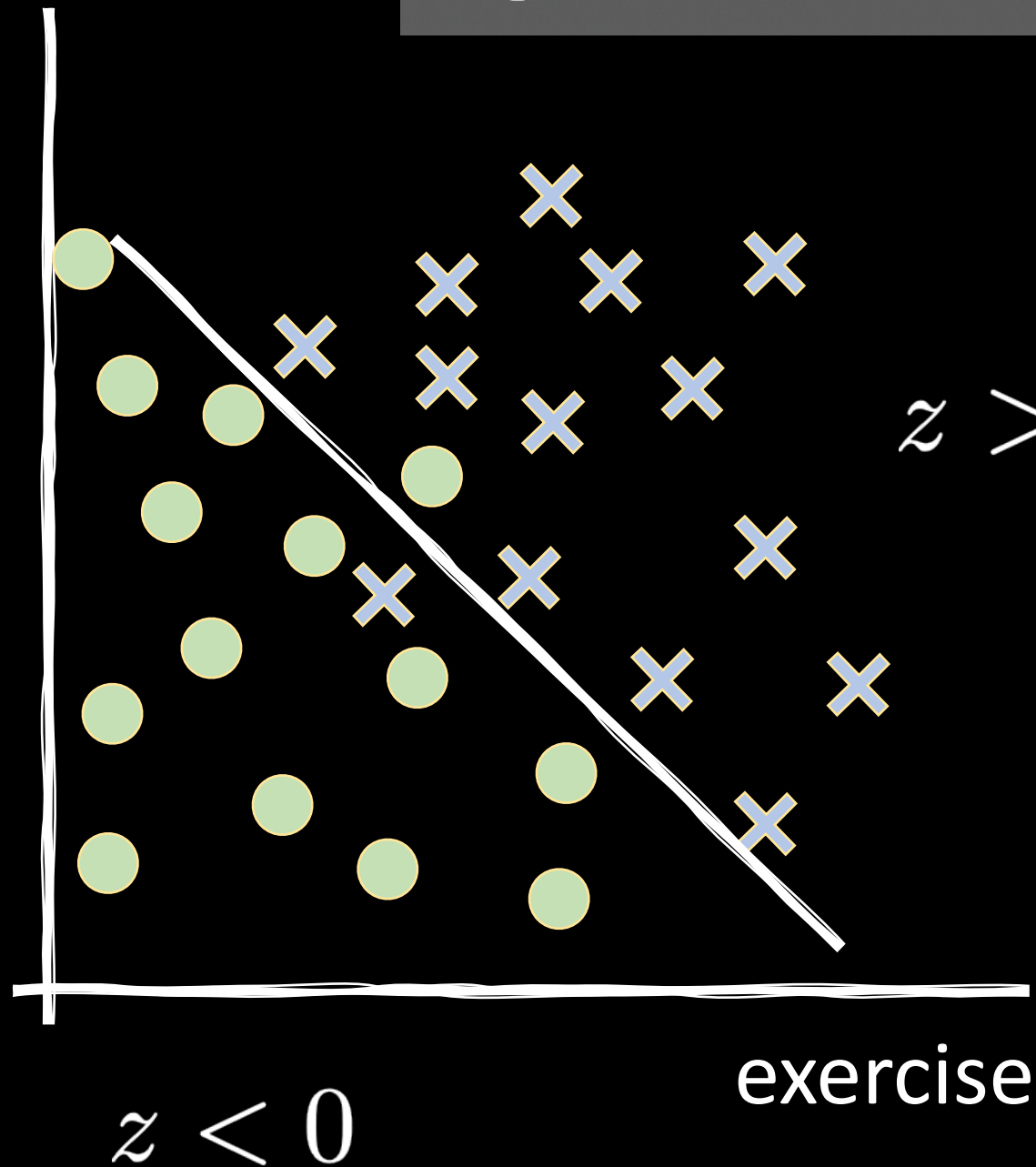
$$w_0 + w_1 \times \text{exercise} + w_2 \times \text{sleep} = z$$



distance to decision boundary

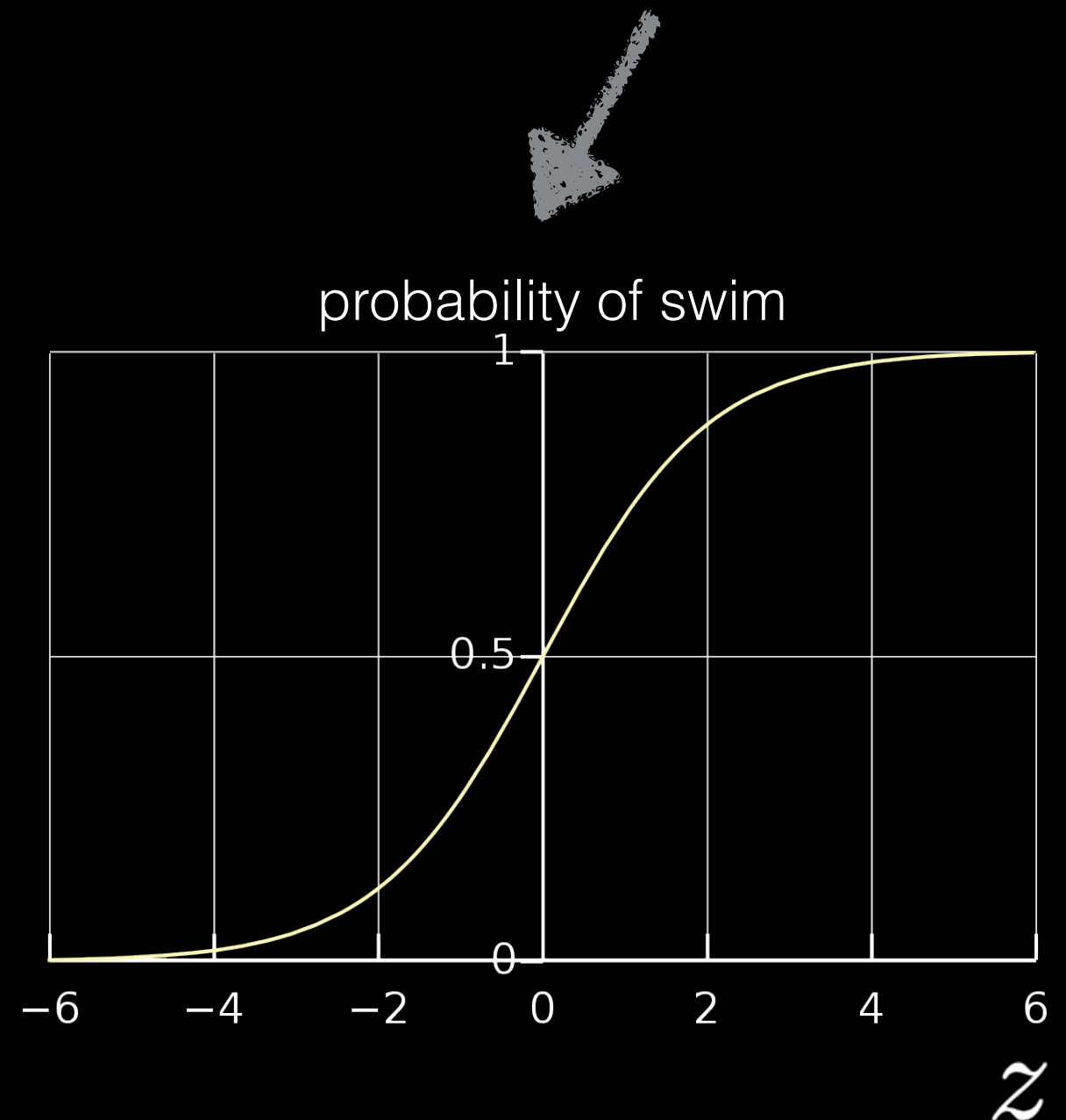
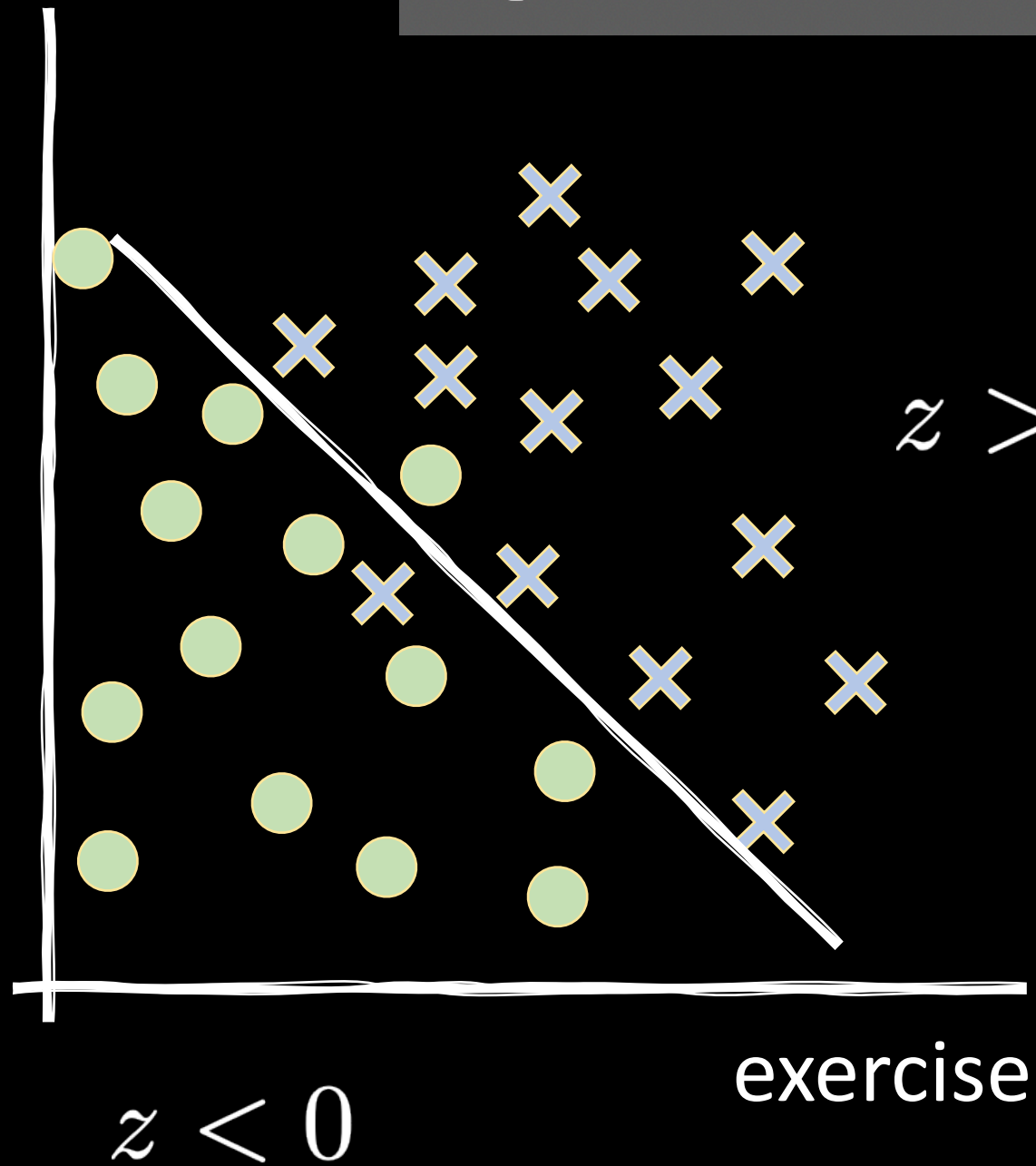
$$w_0 + w_1 \times \text{exercise} + w_2 \times \text{sleep} = z$$

sleep

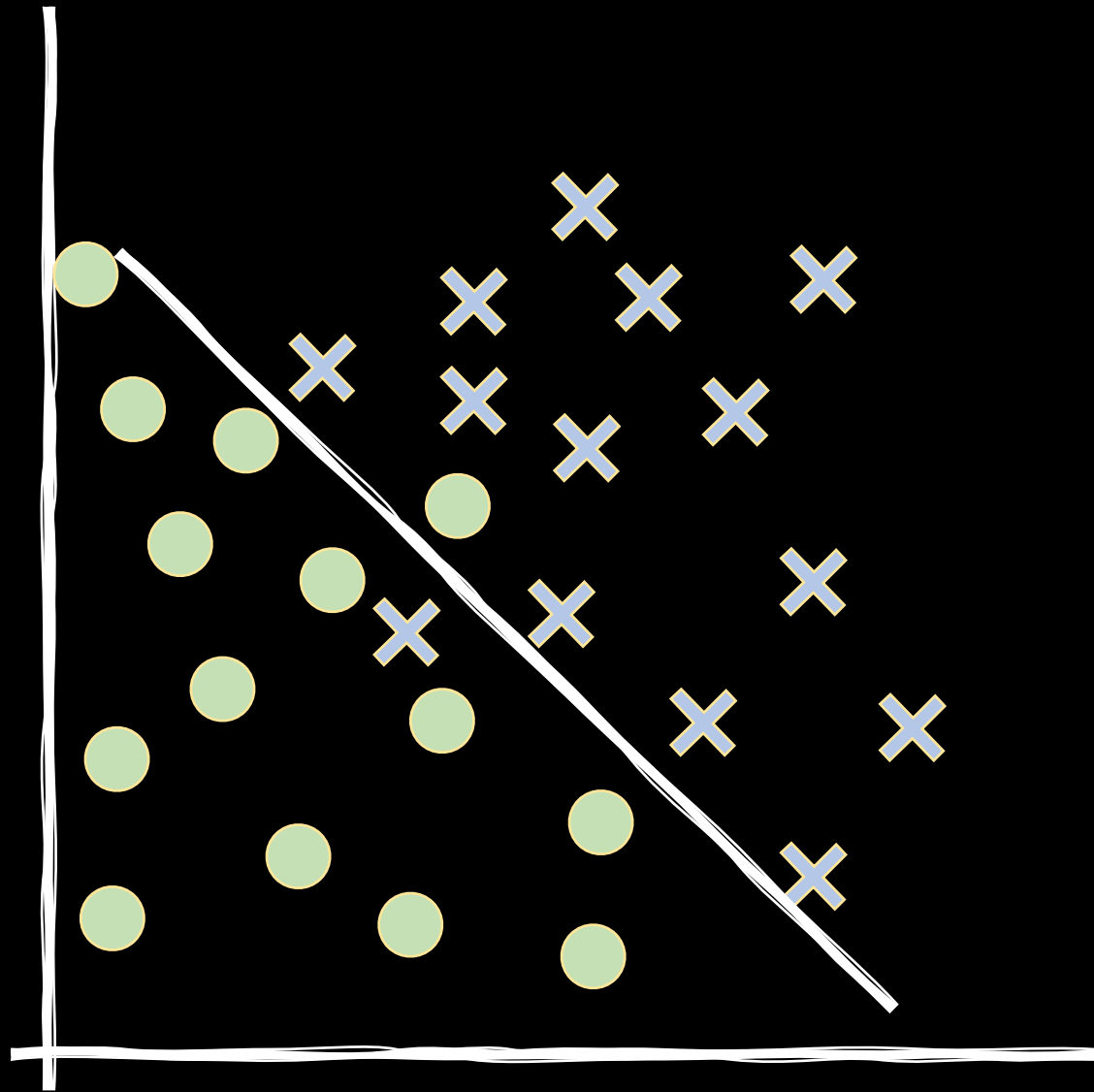


sleep

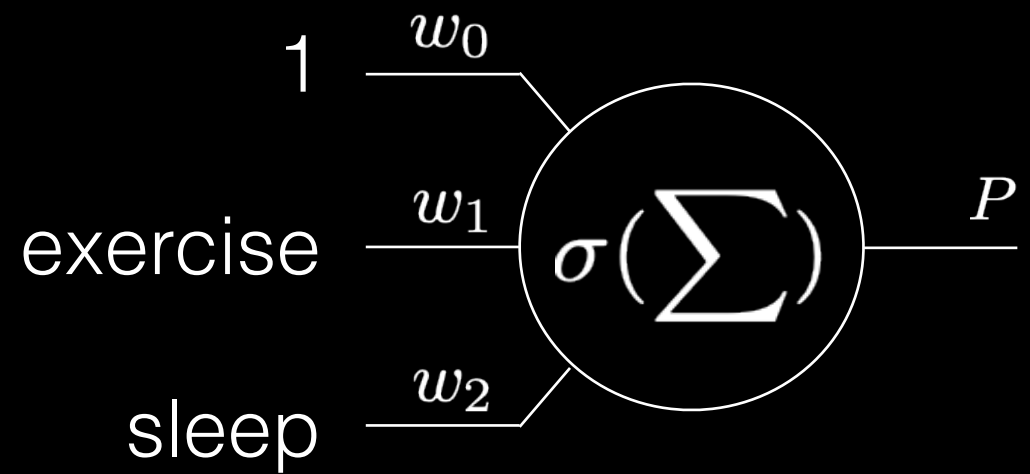
$$w_0 + w_1 \times \text{exercise} + w_2 \times \text{sleep} = z$$



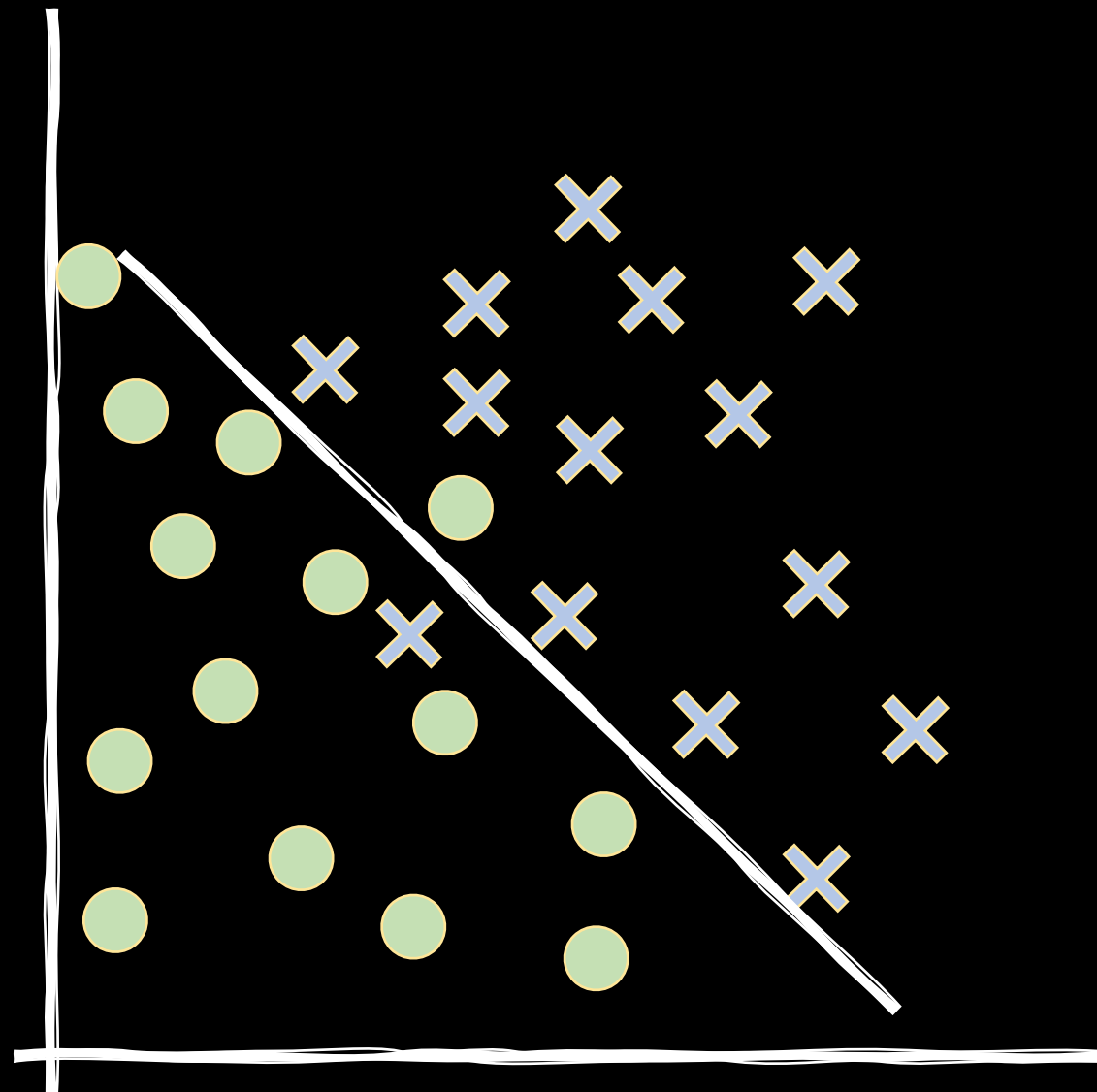
sleep



exercise

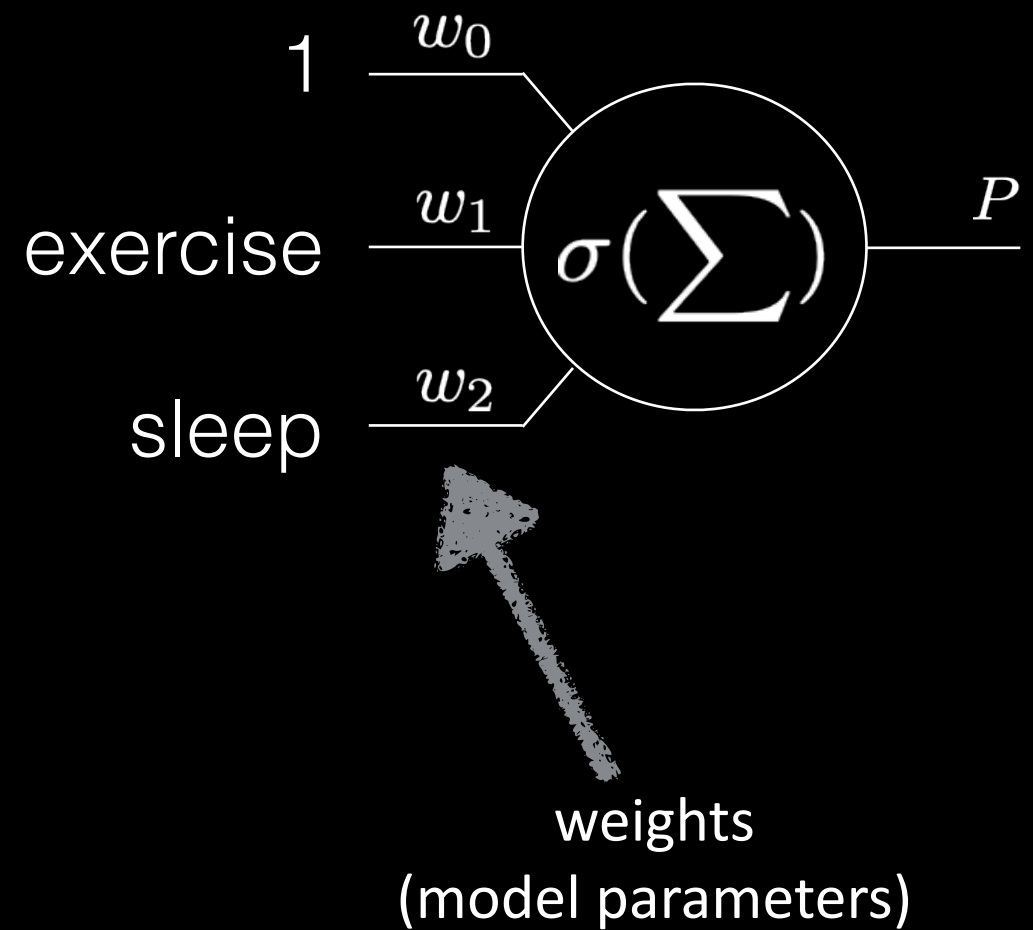


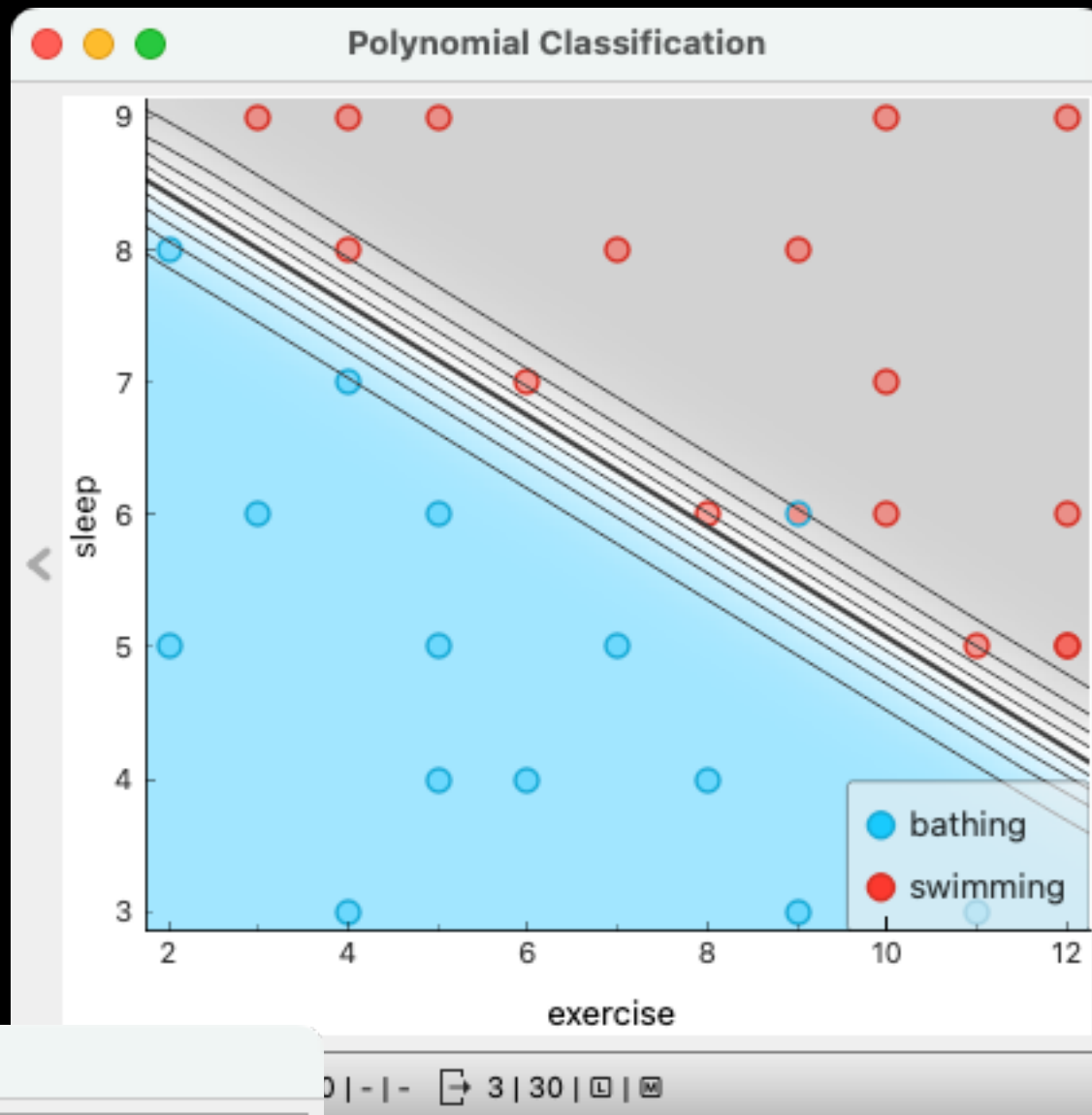
sleep



exercise

logistic regression





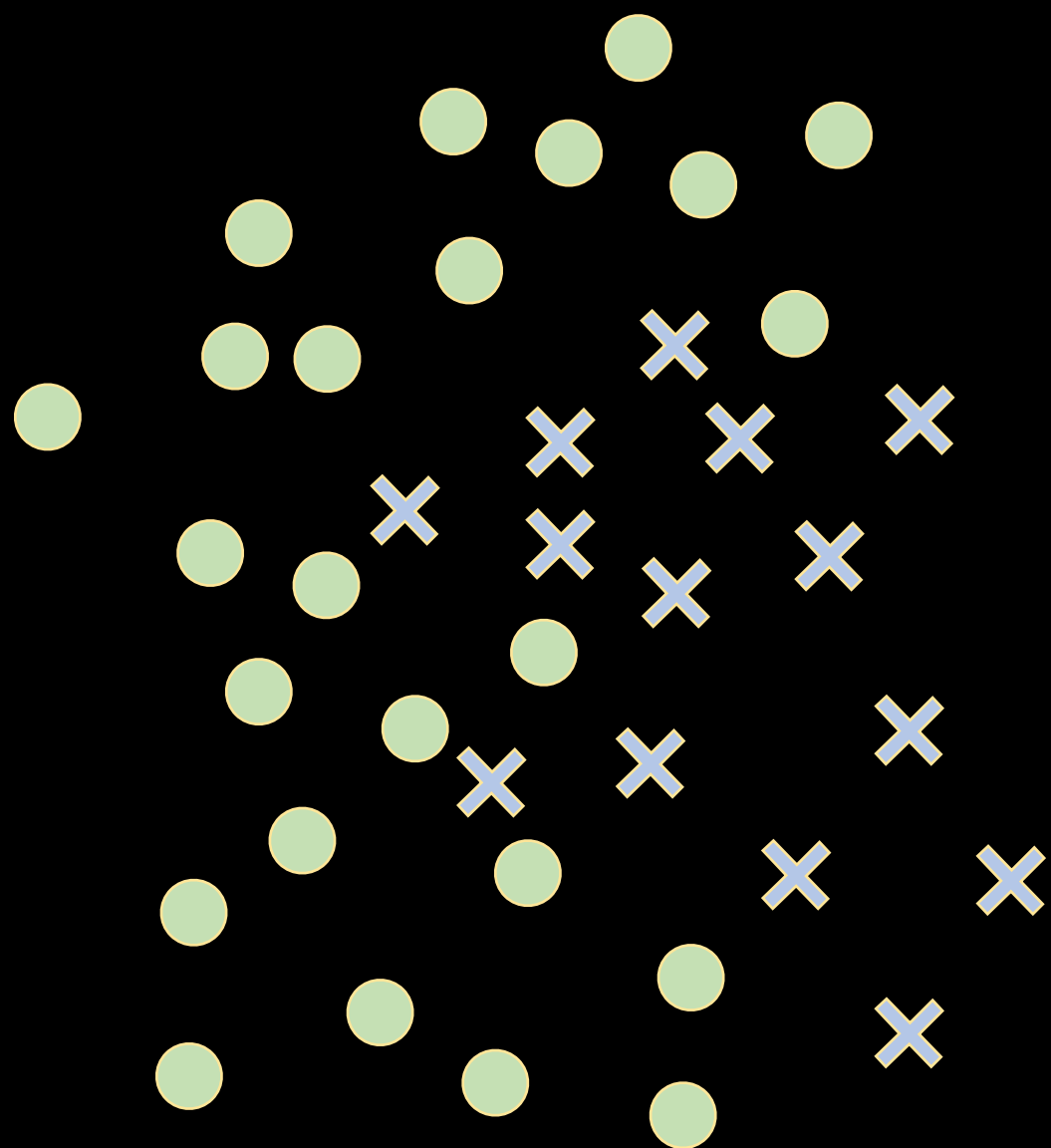
Data Table (1)

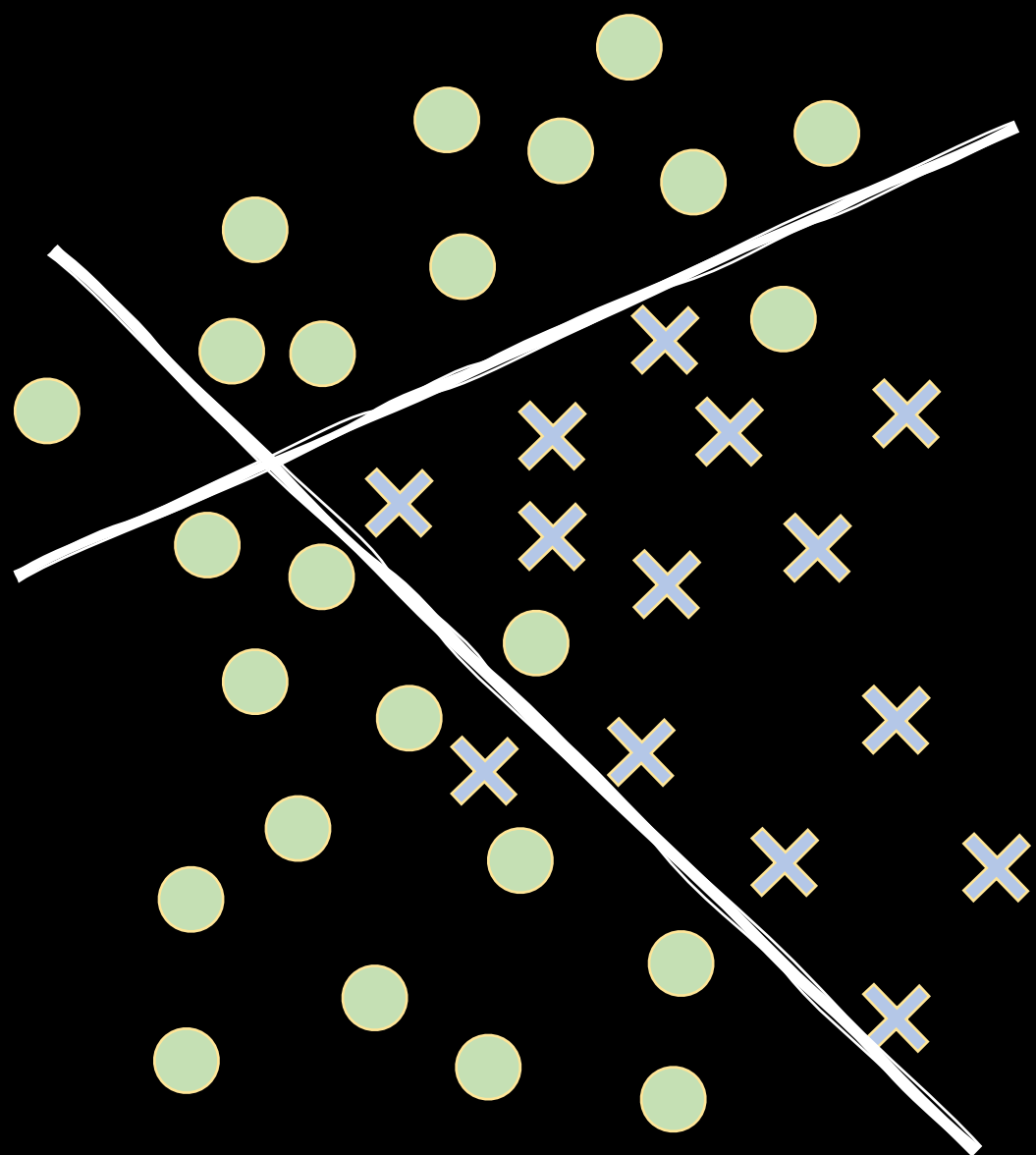
	name	swimming
1	intercept	-15.6318
2	exercise	0.800022
3	sleep	1.63898

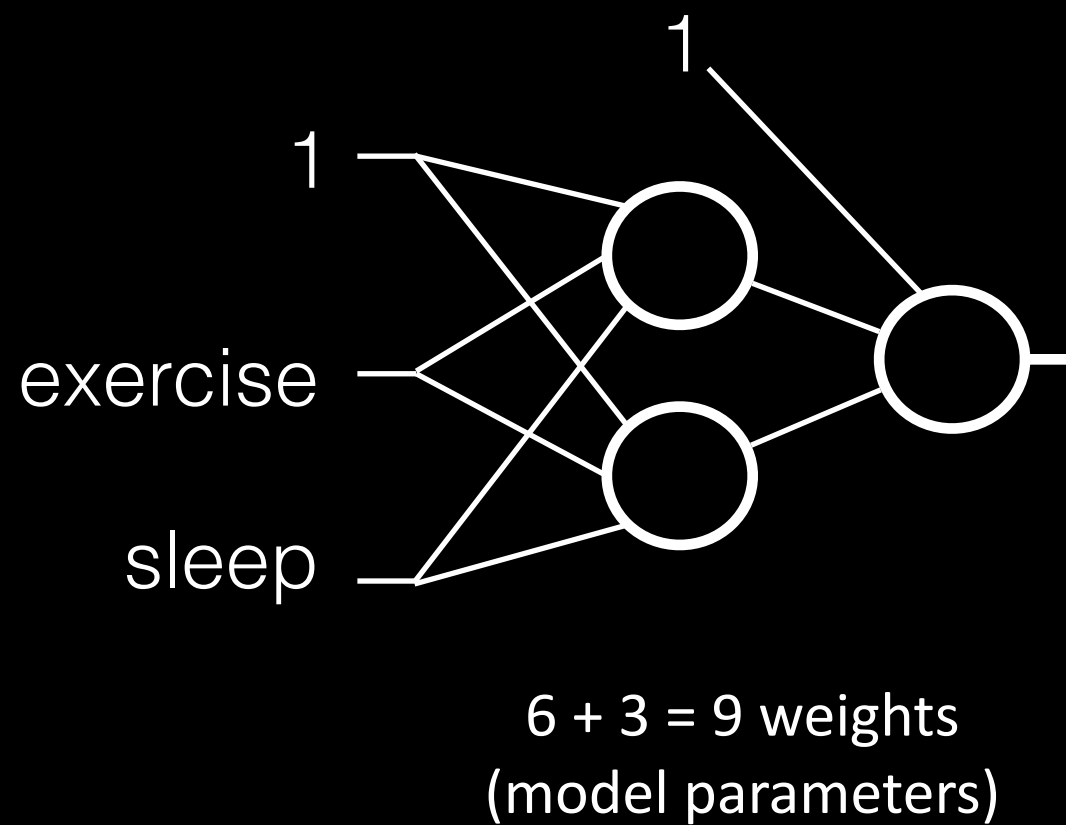
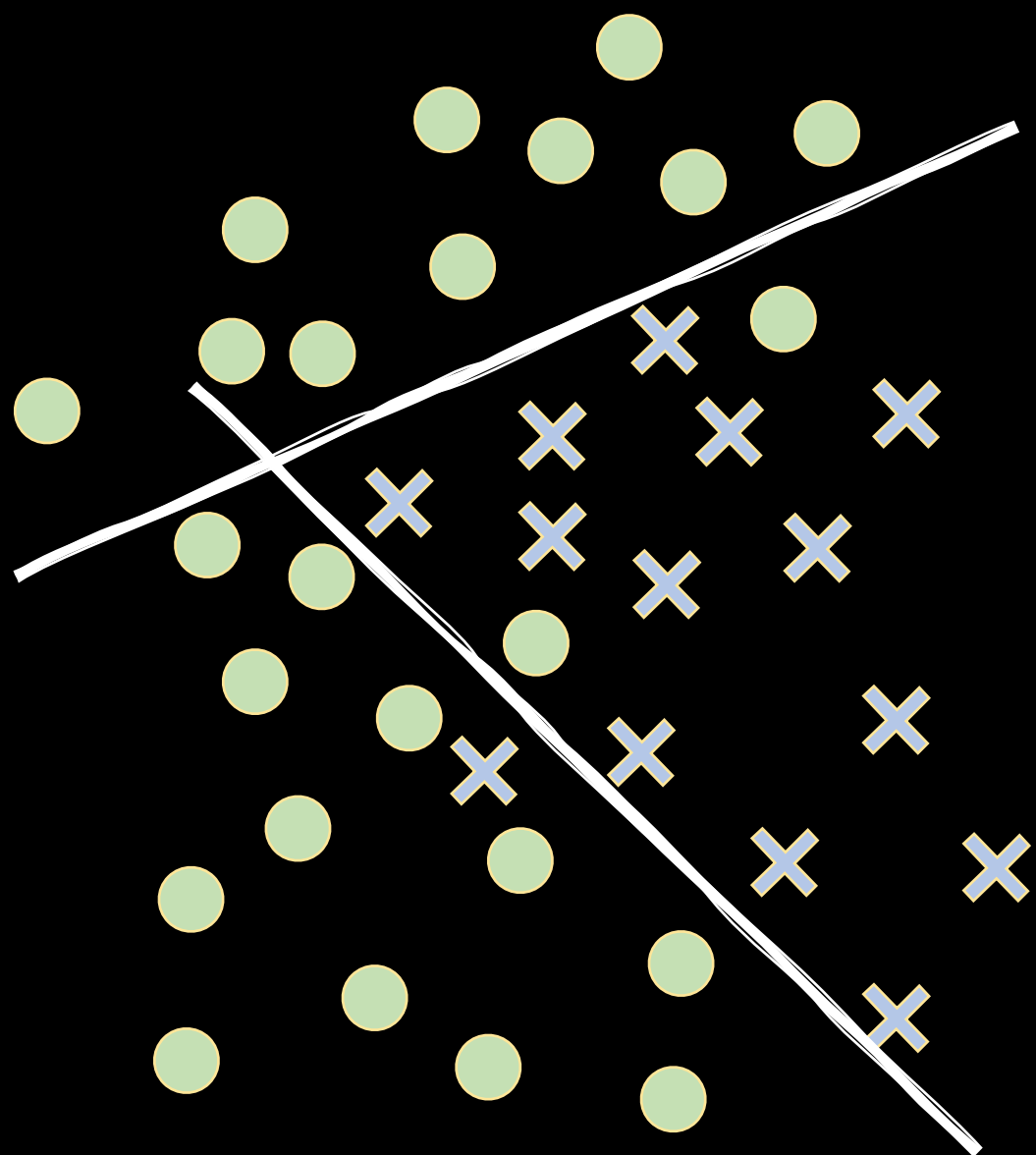
Data Table (2)

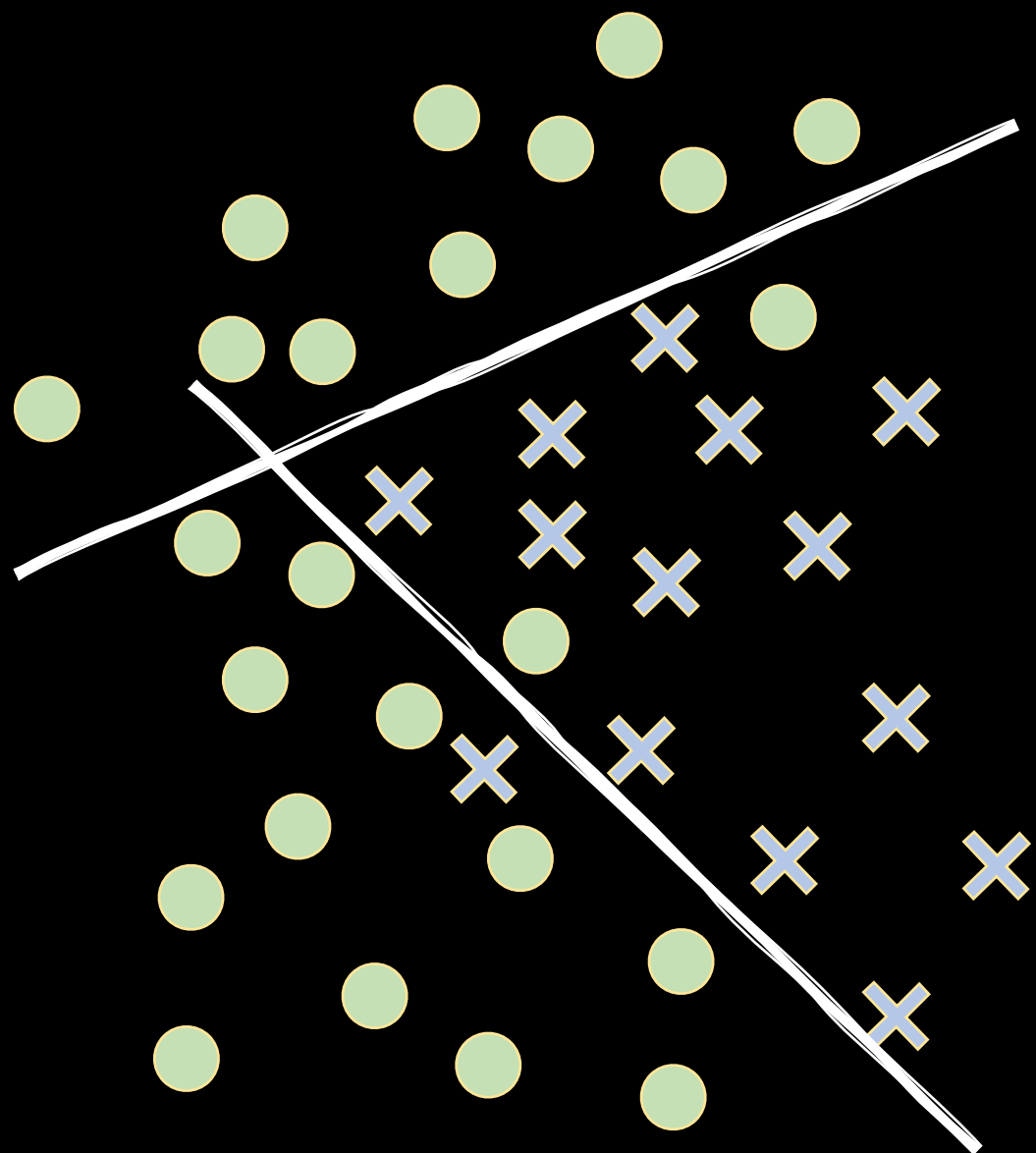
	activity	name	exercise	sleep
11	bathing	Irena	9	3
12	bathing	Jure	8	4
13	bathing	Tjaša	11	3
14	bathing	Boštjan	7	5
15	swimming	Helena	4	9
16	swimming	Matej	4	8
17	swimming	Alenka	7	8
18	swimming	Gregor	10	9
19	swimming	Polona	9	8
20	swimming	Saša	10	7

$$\begin{array}{l}
 1 \\
 \text{exercise} \\
 \text{sleep}
 \end{array}
 \begin{array}{l}
 -15.6 \\
 0.80 \\
 1.64
 \end{array}
 \begin{array}{c}
 \sigma(\Sigma)
 \end{array}
 \begin{array}{l}
 P
 \end{array}$$

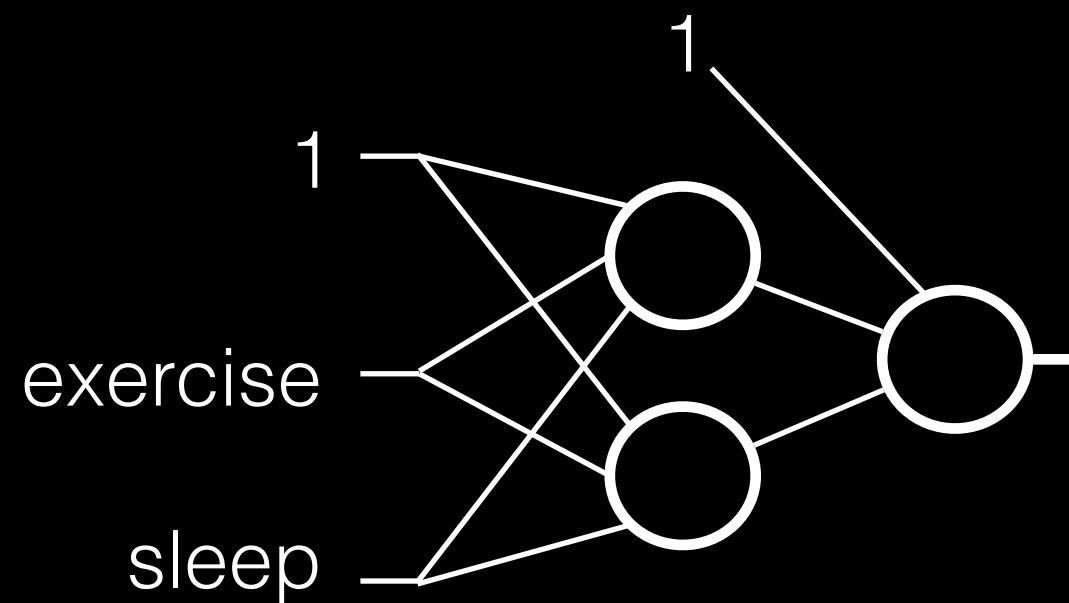




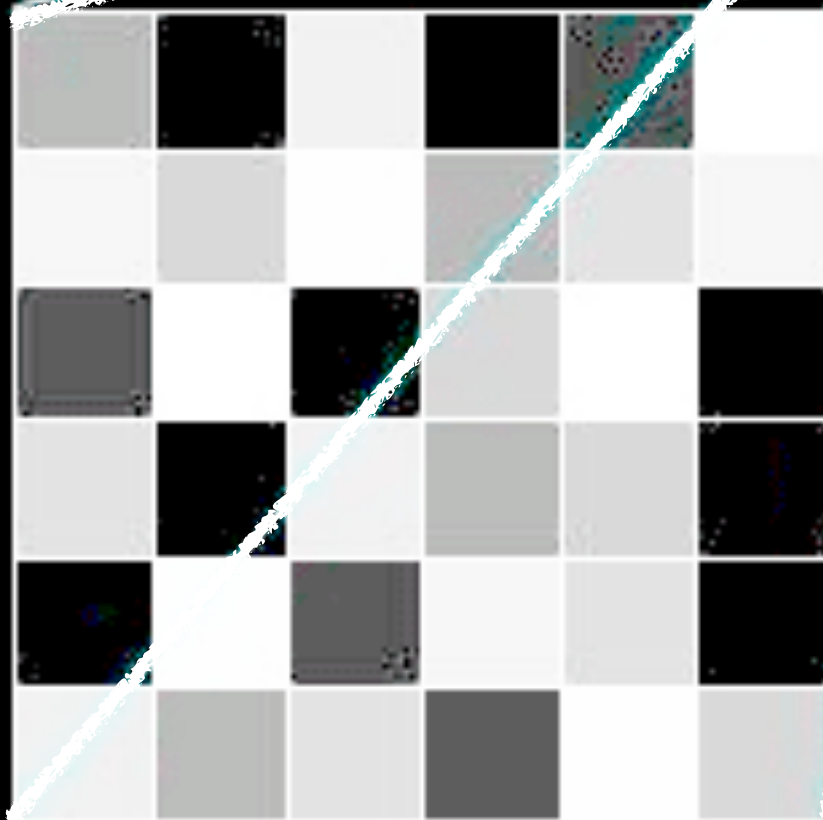




neural network



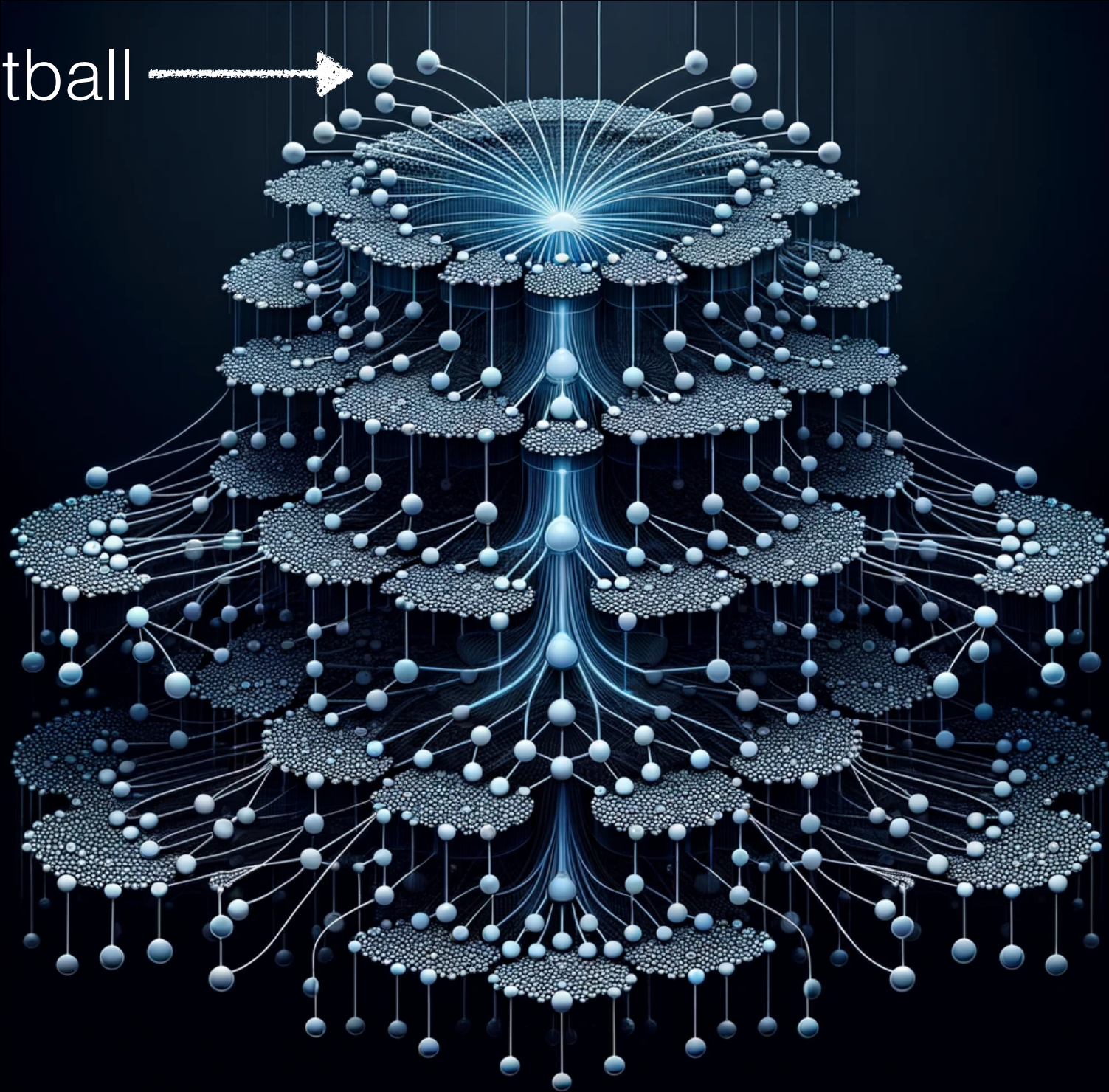
$6 + 3 = 9$ weights
(model parameters)



170	238	85	255	221	0
68	136	17	170	119	68
221	0	238	136	0	255
119	255	85	170	136	238
238	17	221	68	119	255
85	170	119	221	17	136

output (e.g., image classification)

basketball →



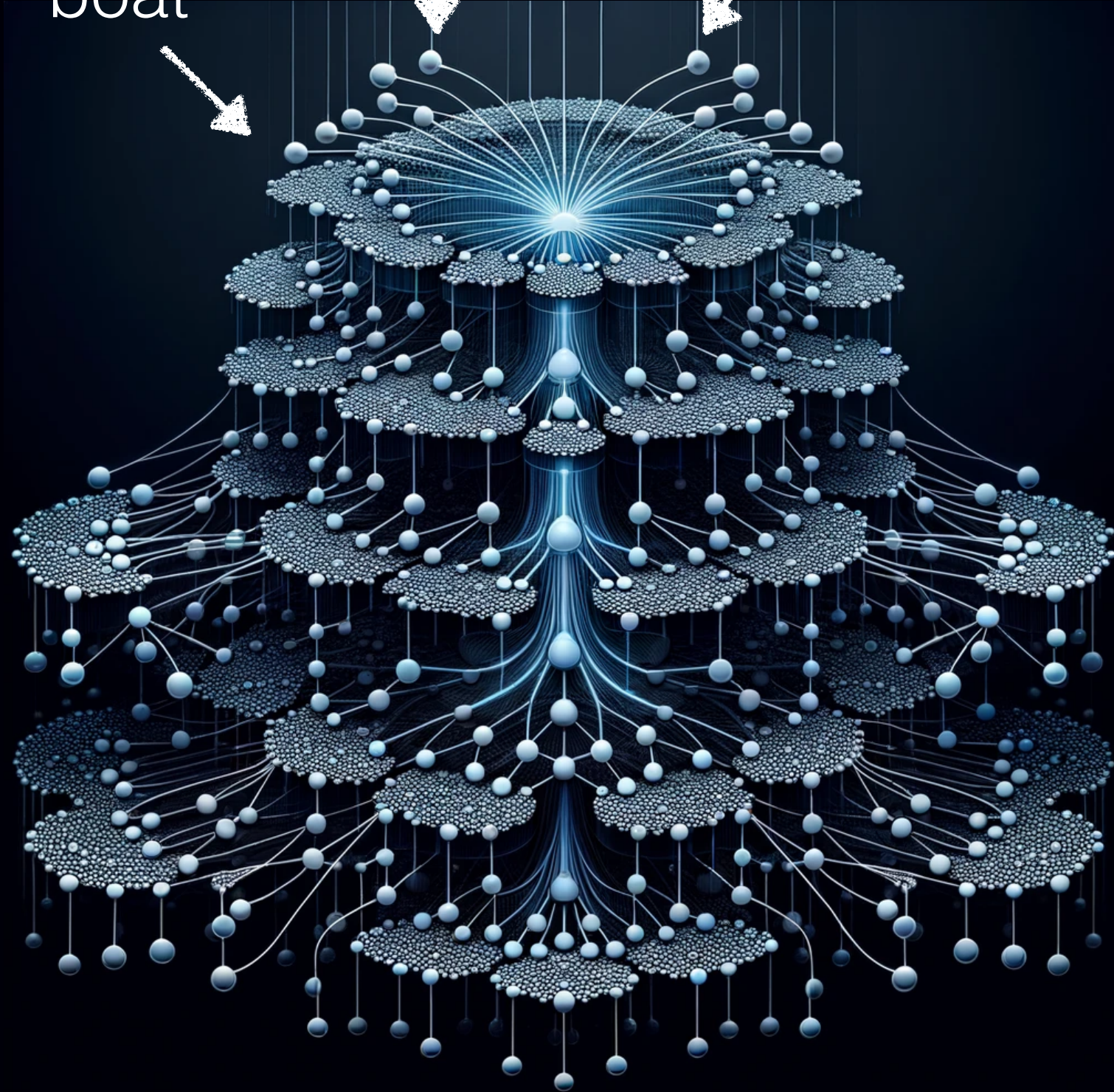
input (numbers, e.g. pixels from images)



boat

dog

house



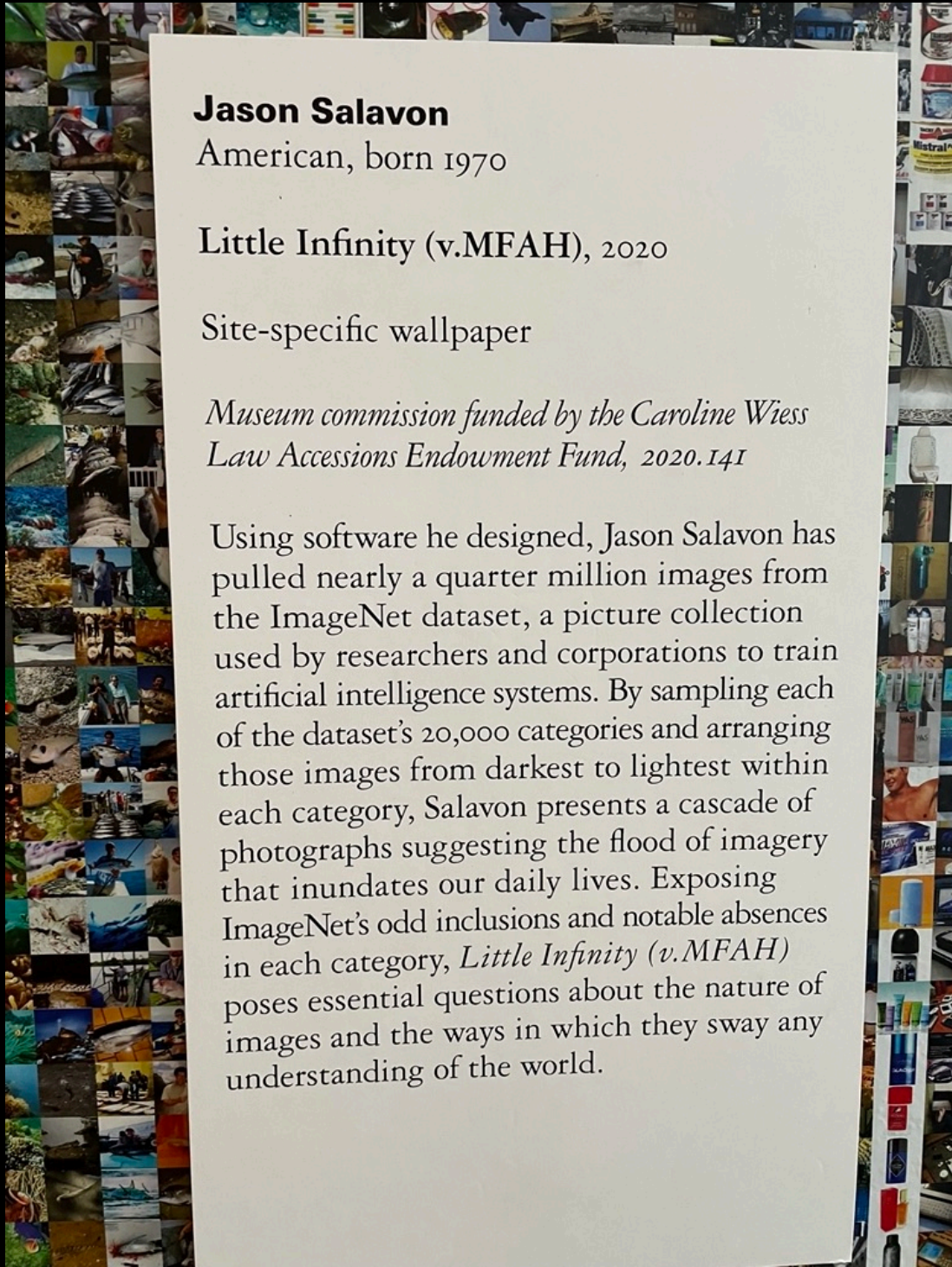












Jason Salavon

American, born 1970

Little Infinity (v.MFAH), 2020

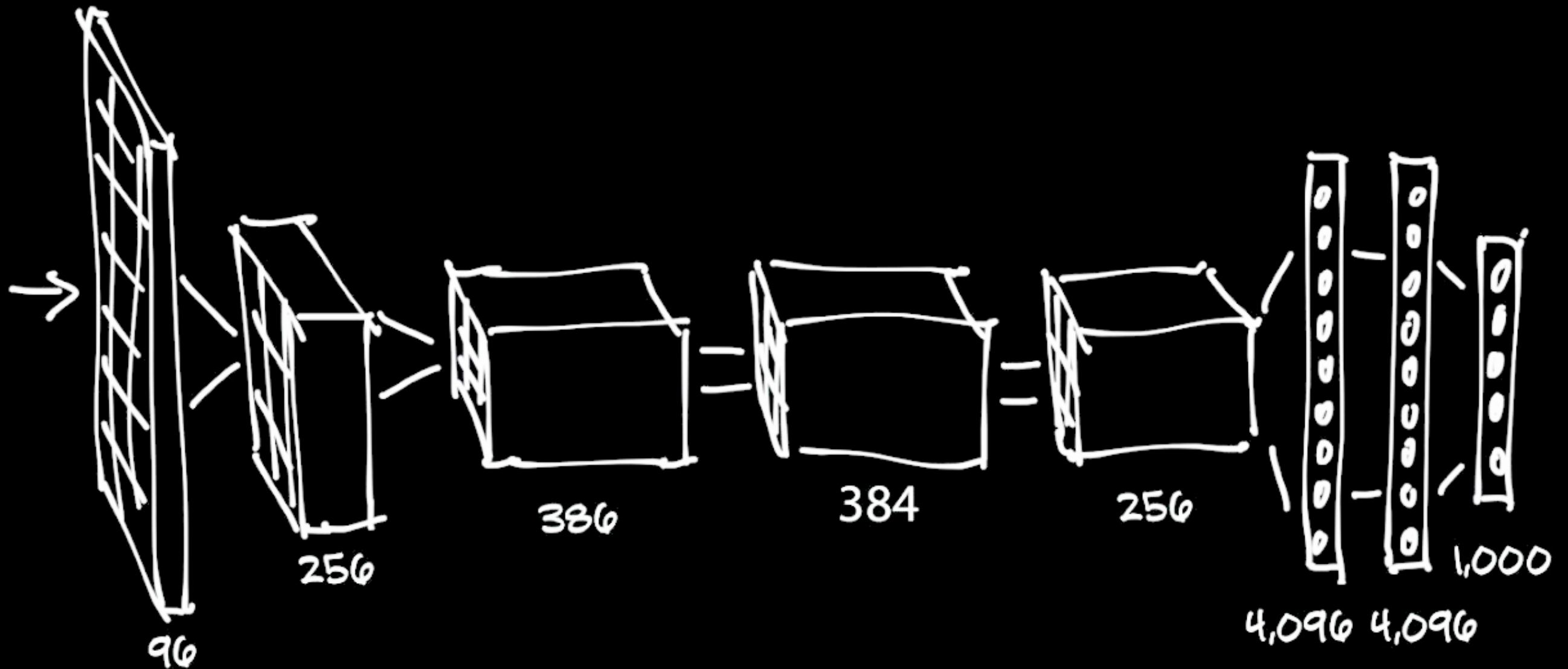
Site-specific wallpaper

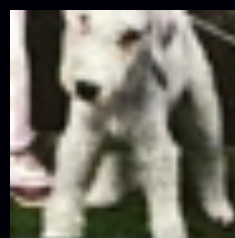
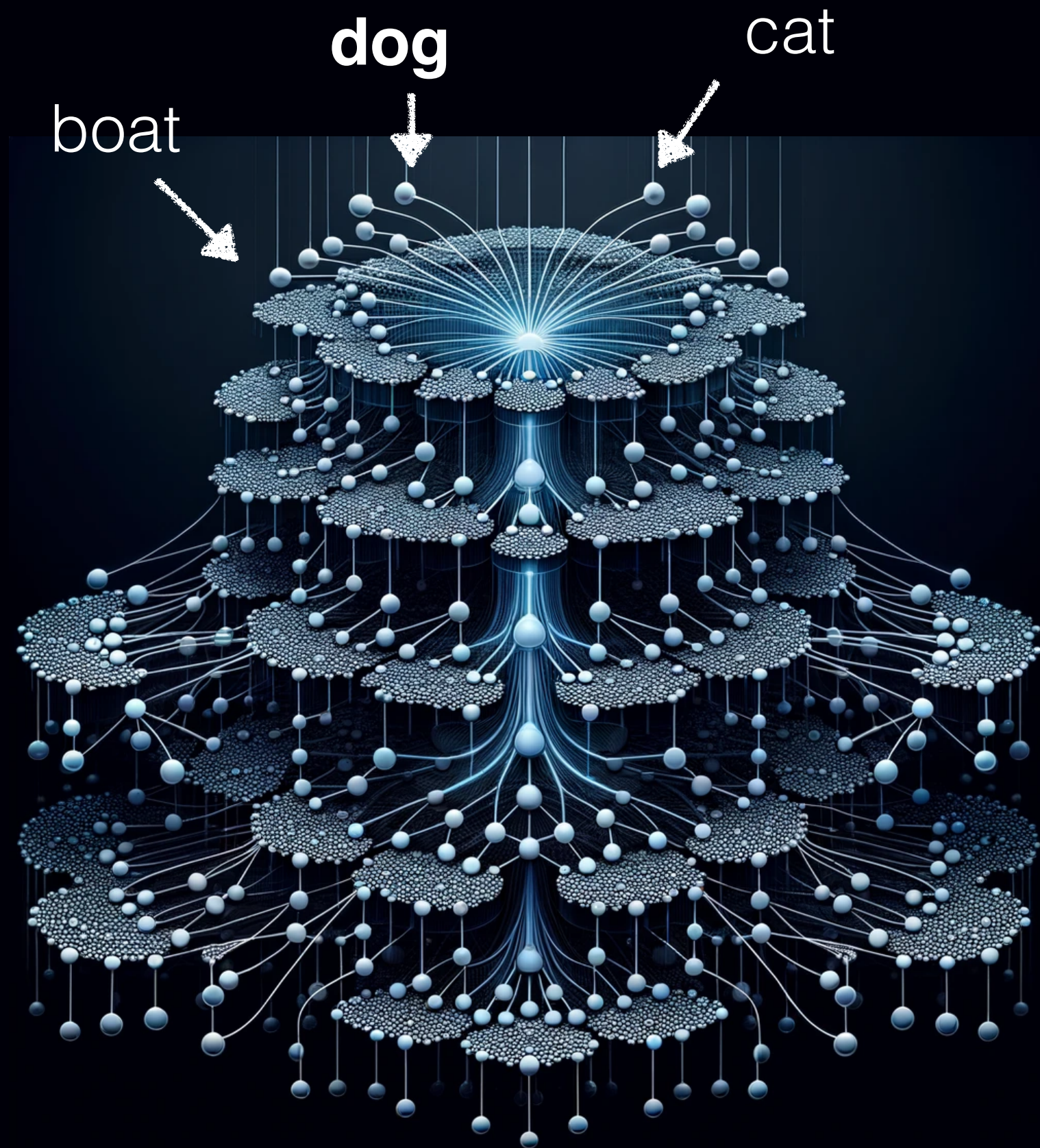
*Museum commission funded by the Caroline Wiess
Law Accessions Endowment Fund, 2020. 141*

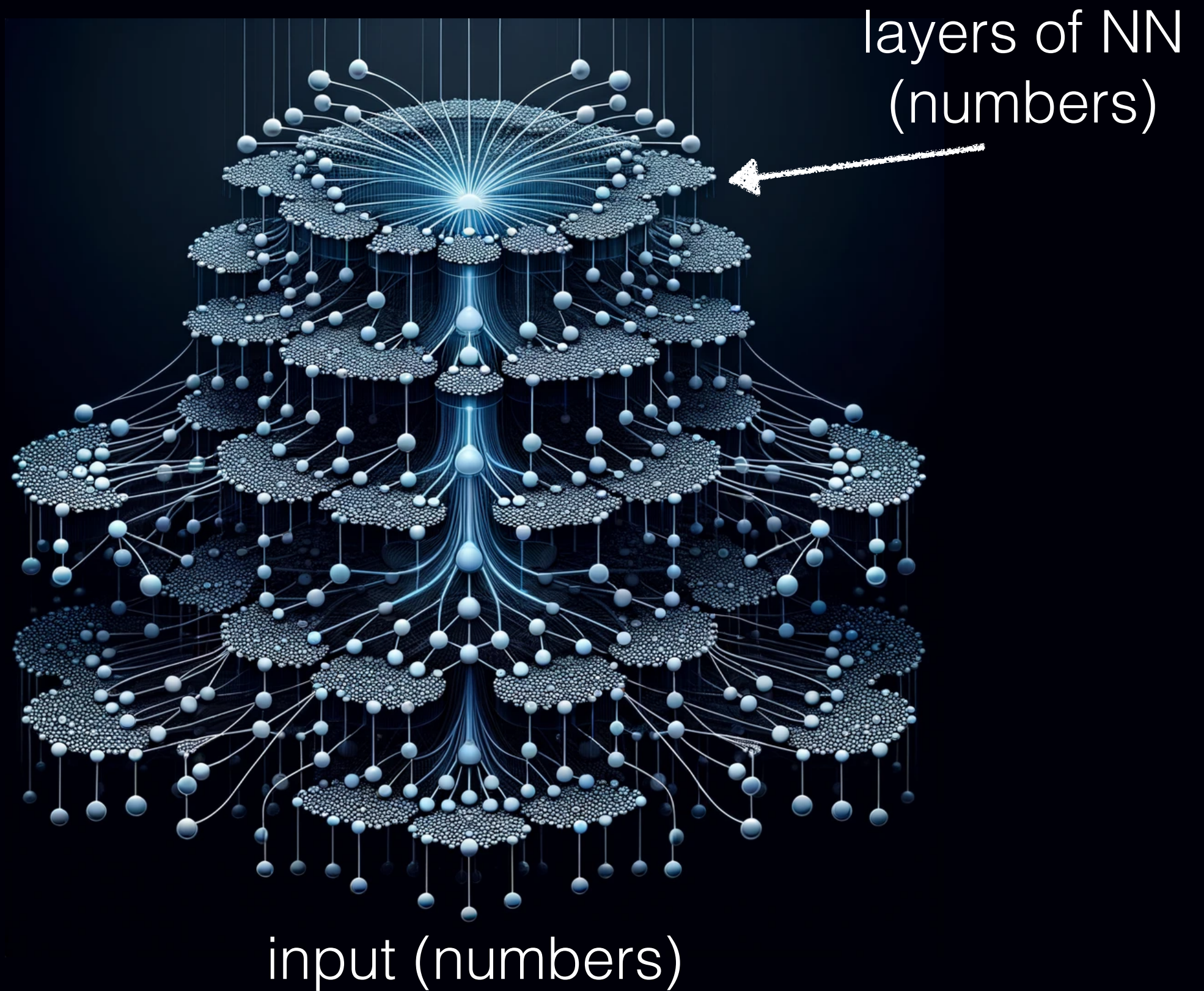
Using software he designed, Jason Salavon has pulled nearly a quarter million images from the ImageNet dataset, a picture collection used by researchers and corporations to train artificial intelligence systems. By sampling each of the dataset's 20,000 categories and arranging those images from darkest to lightest within each category, Salavon presents a cascade of photographs suggesting the flood of imagery that inundates our daily lives. Exposing ImageNet's odd inclusions and notable absences in each category, *Little Infinity (v.MFAH)* poses essential questions about the nature of images and the ways in which they sway any understanding of the world.

AlexNet (Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton)

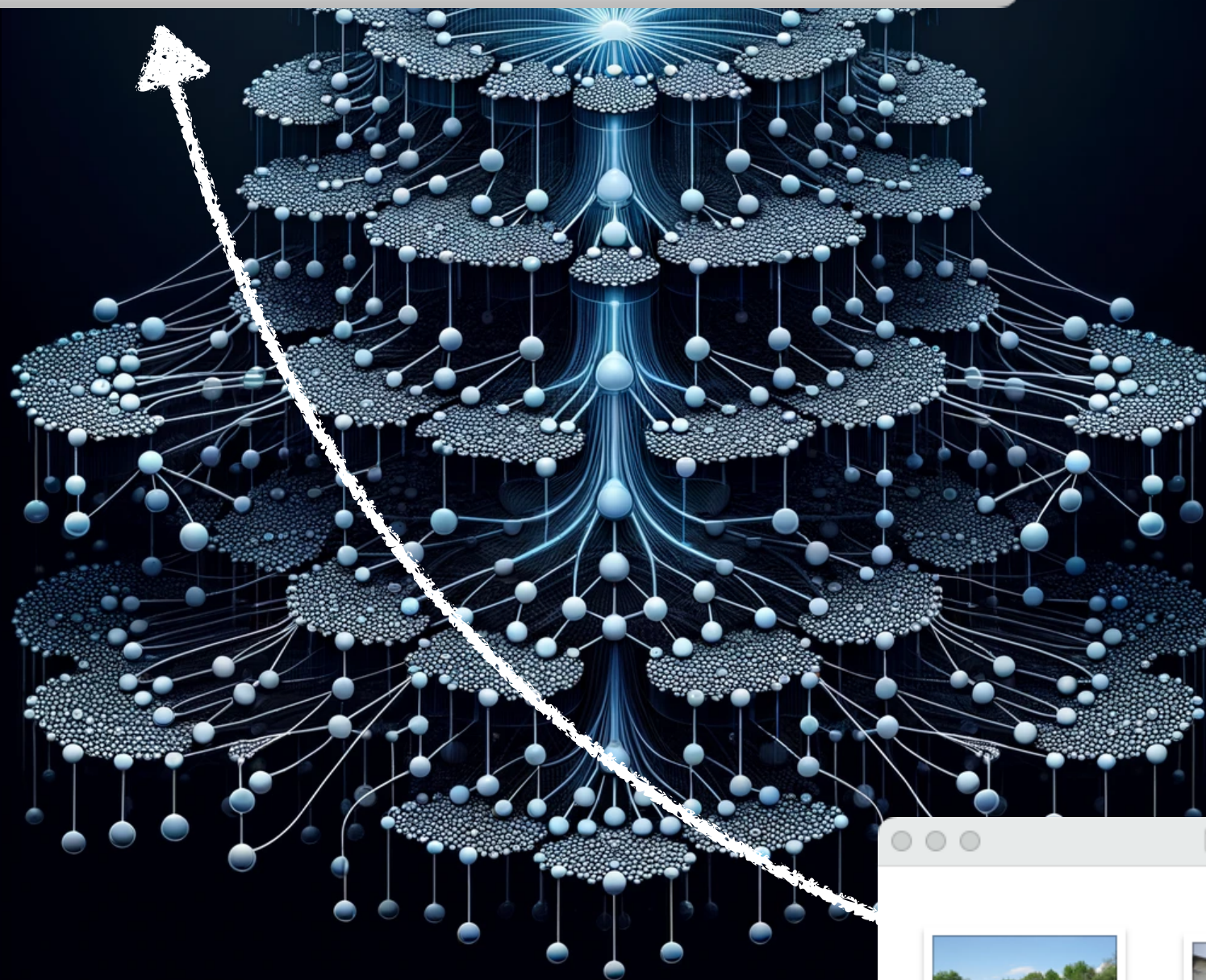
60 million parameters



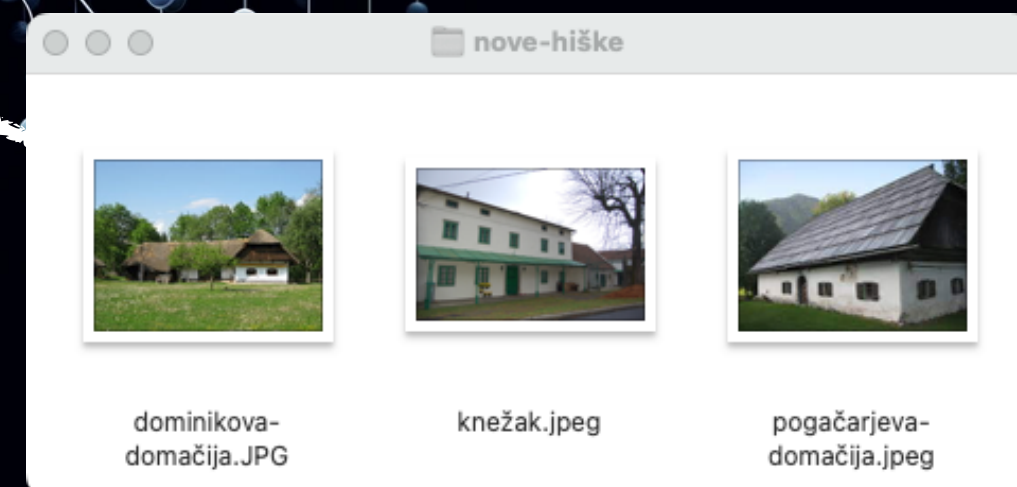


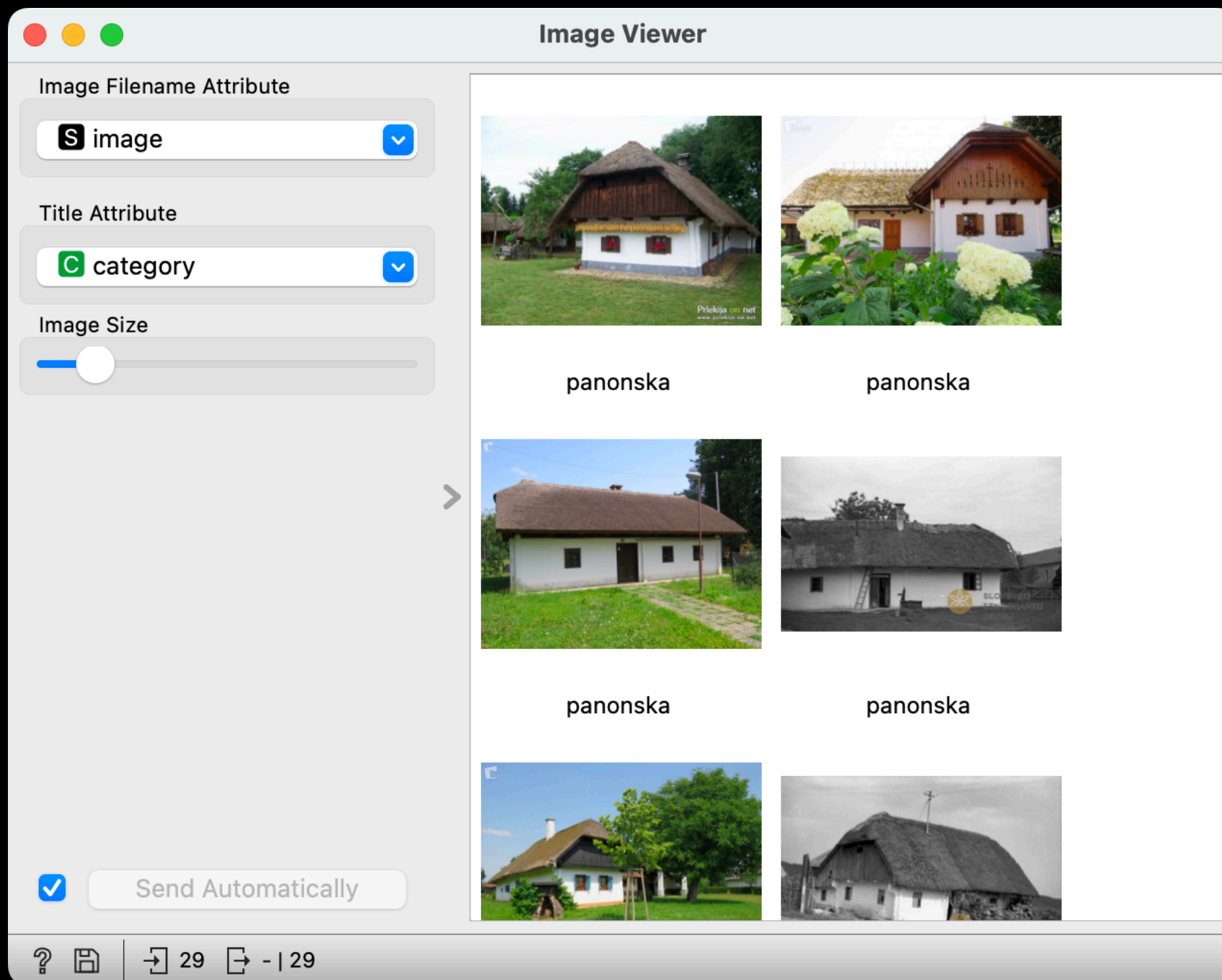


hidden	image name	n0 True	n1 True	n2 True	n3 True	n4 True
1	pogačarjeva...	0.422006	0.659342	0.255844	0.165216	0.303353
2	knežak	0.312856	0.0879217	0.603873	0.0636521	0.248142
3	dominikova-...	0.392473	0.0192729	0.456797	0.320302	0.0885926



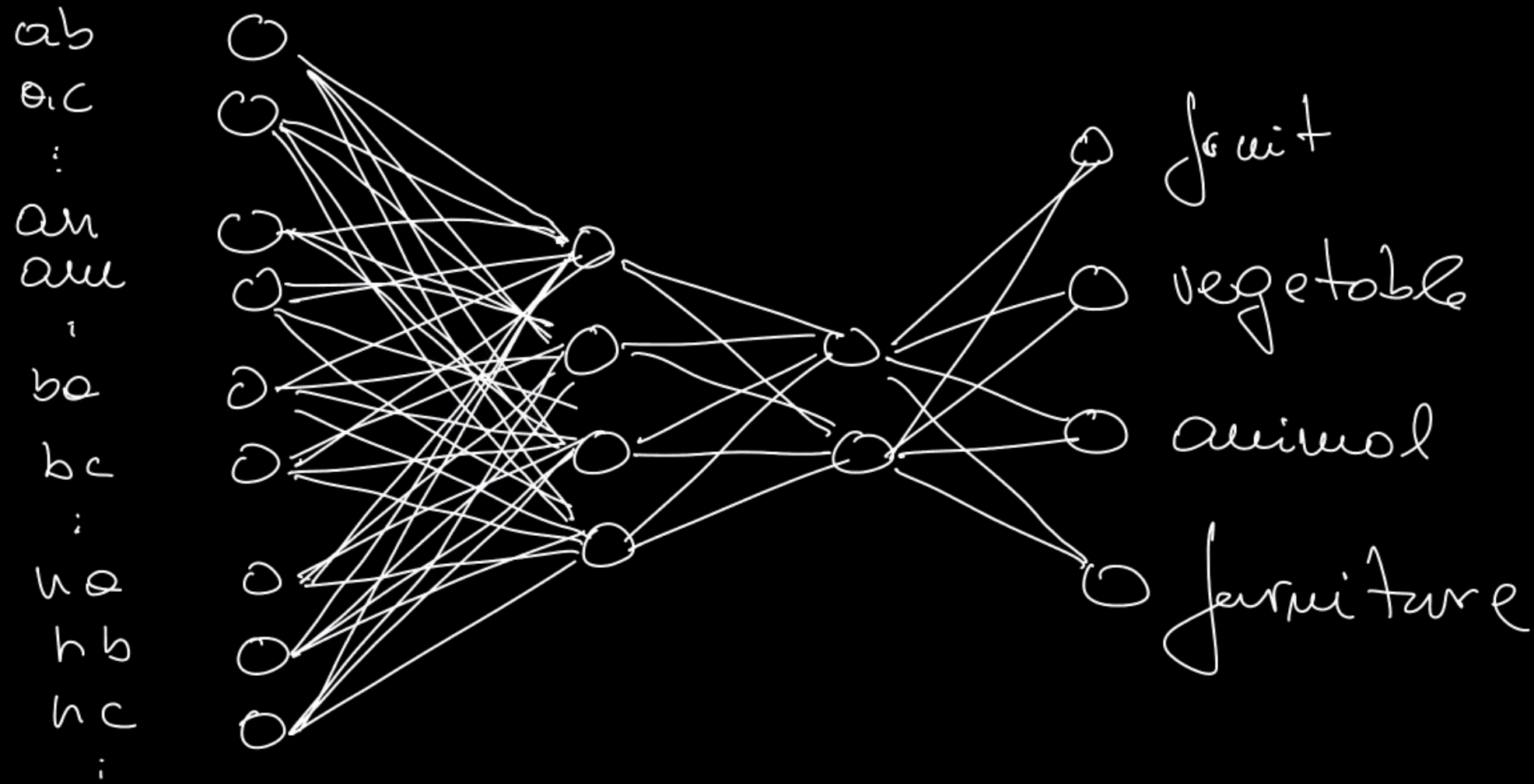
embedding into a vector space



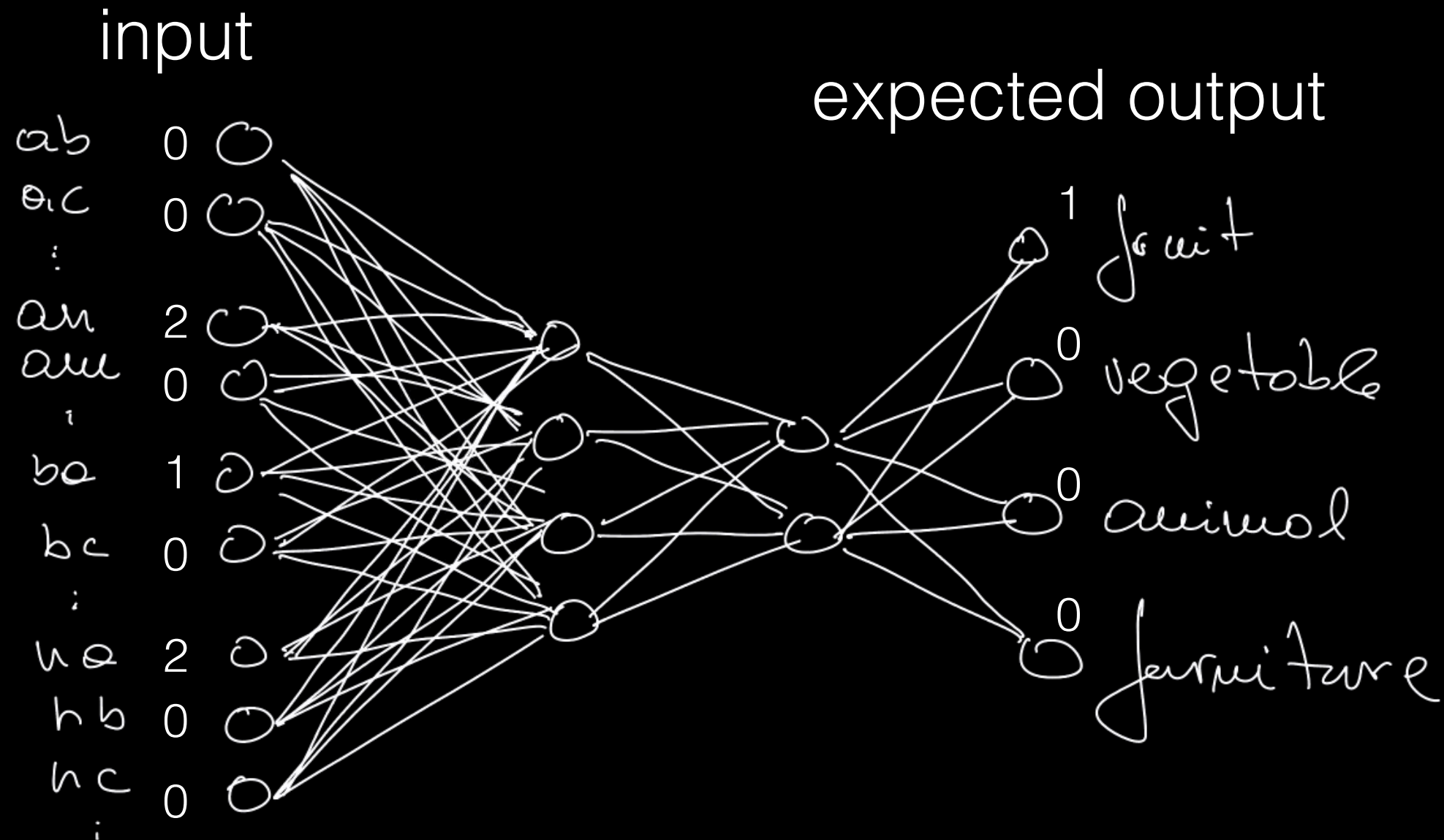


Data Table	
	words
1	Apple
2	Arrow
3	Banana
4	Bat
5	Bed
6	Blanket
7	Book
8	Bowl
9	Bread
10	Bridge
11	Camera
12	Car
13	Cat
14	Cello

words - n-grams - neural network - classification



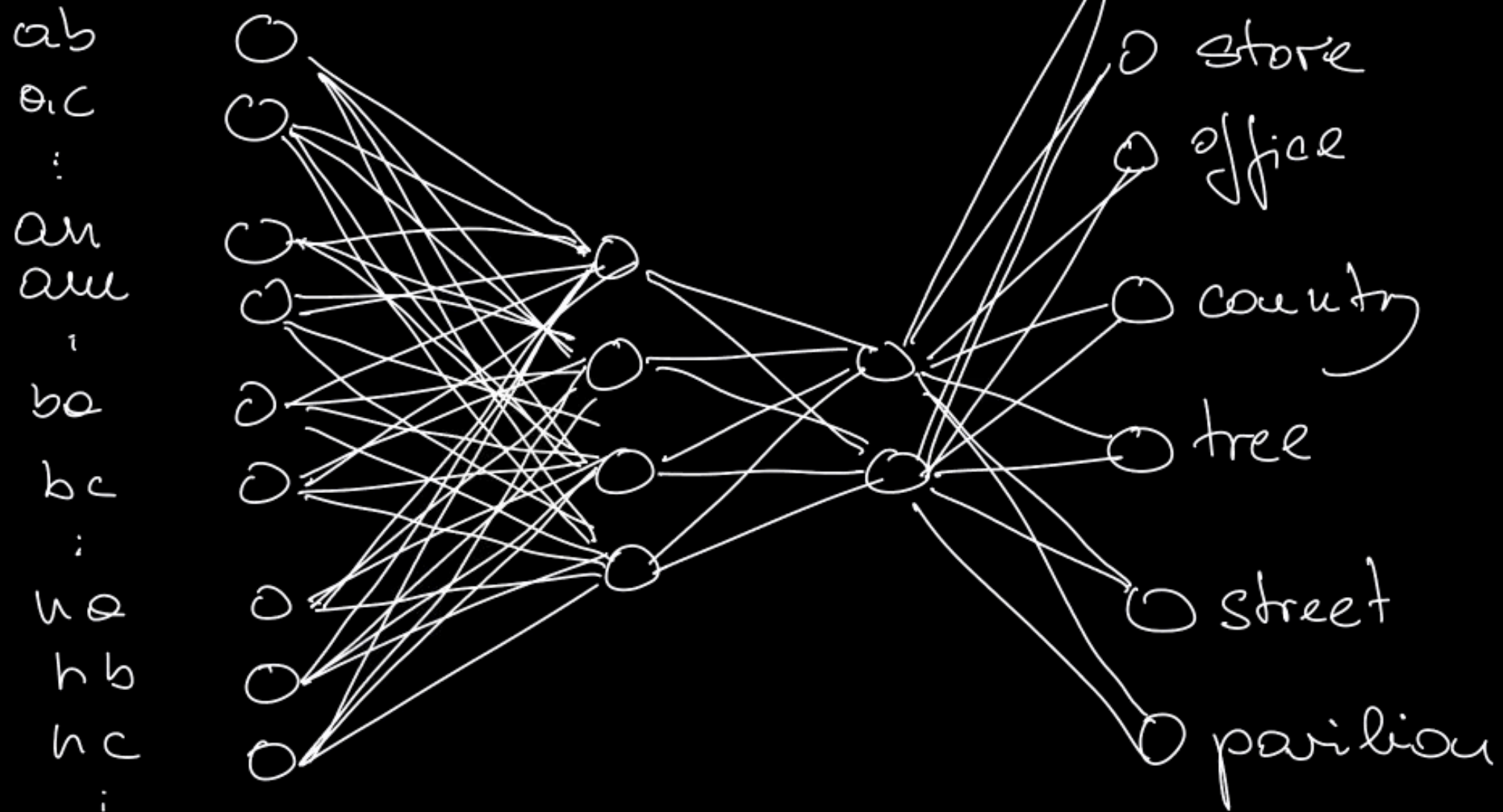
words - n-grams - neural network - classification



banana – ba, an, na, an, na

encoding of
input word

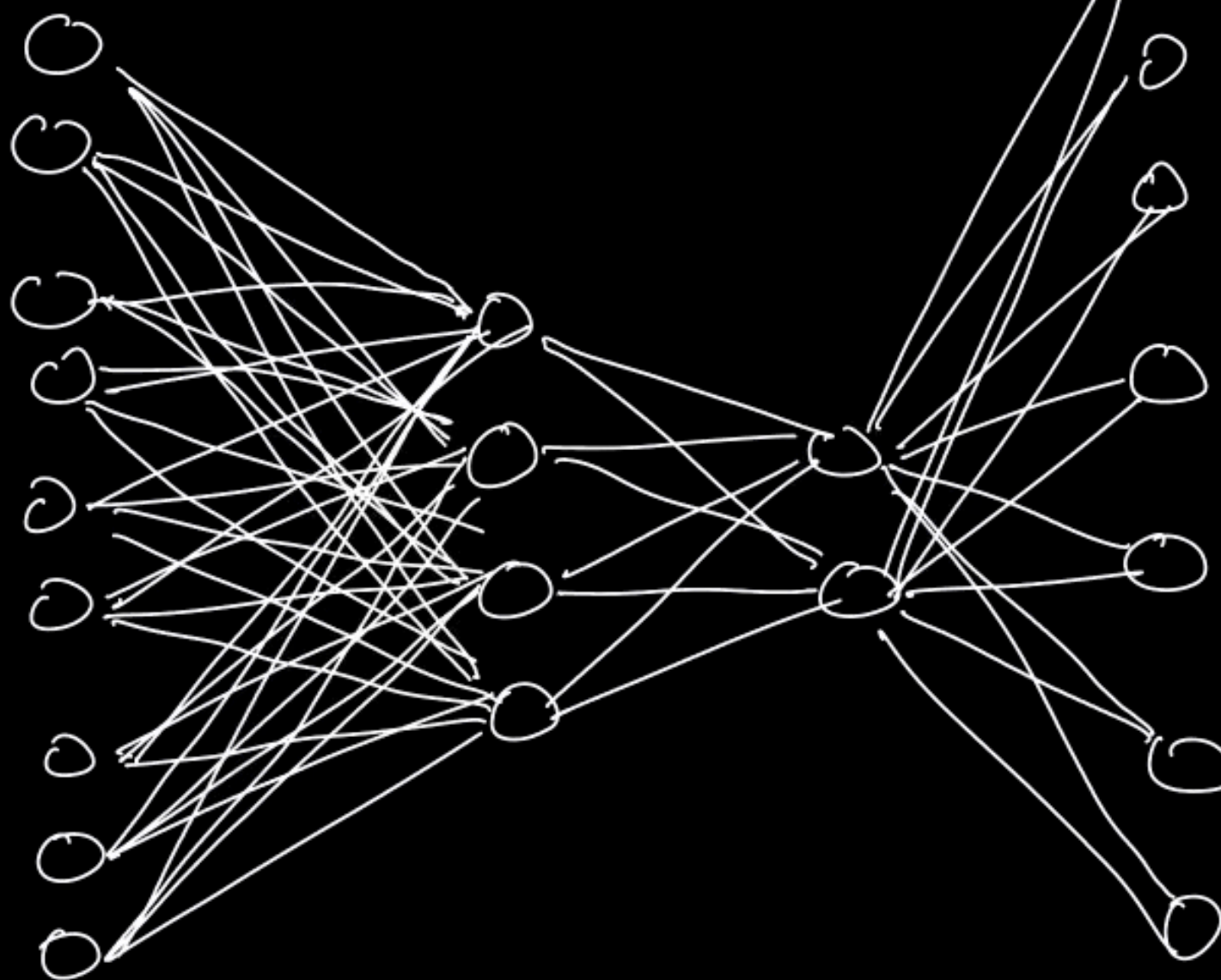
prediction
of next word



encoding of
input string

prediction
of next token

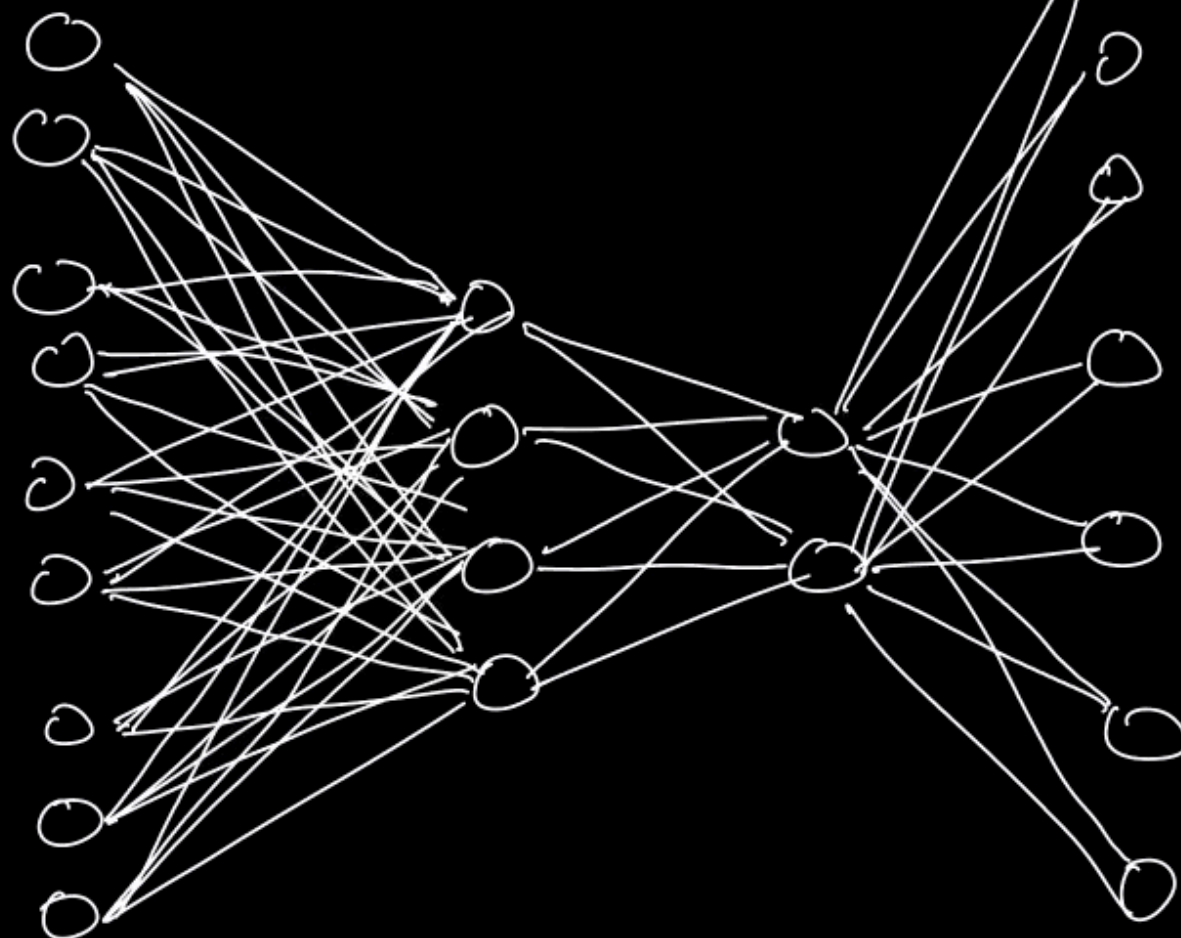
ab
bc
:
an
au
i
ba
bc
:
ba
bb
bc
i



ab
ac
:
au
au
:
be
bc
bcd
:

encoding of
input string

ab
bc
:
an
au
:
ba
bc
:
ba
bb
bc
:
:



prediction
of next token

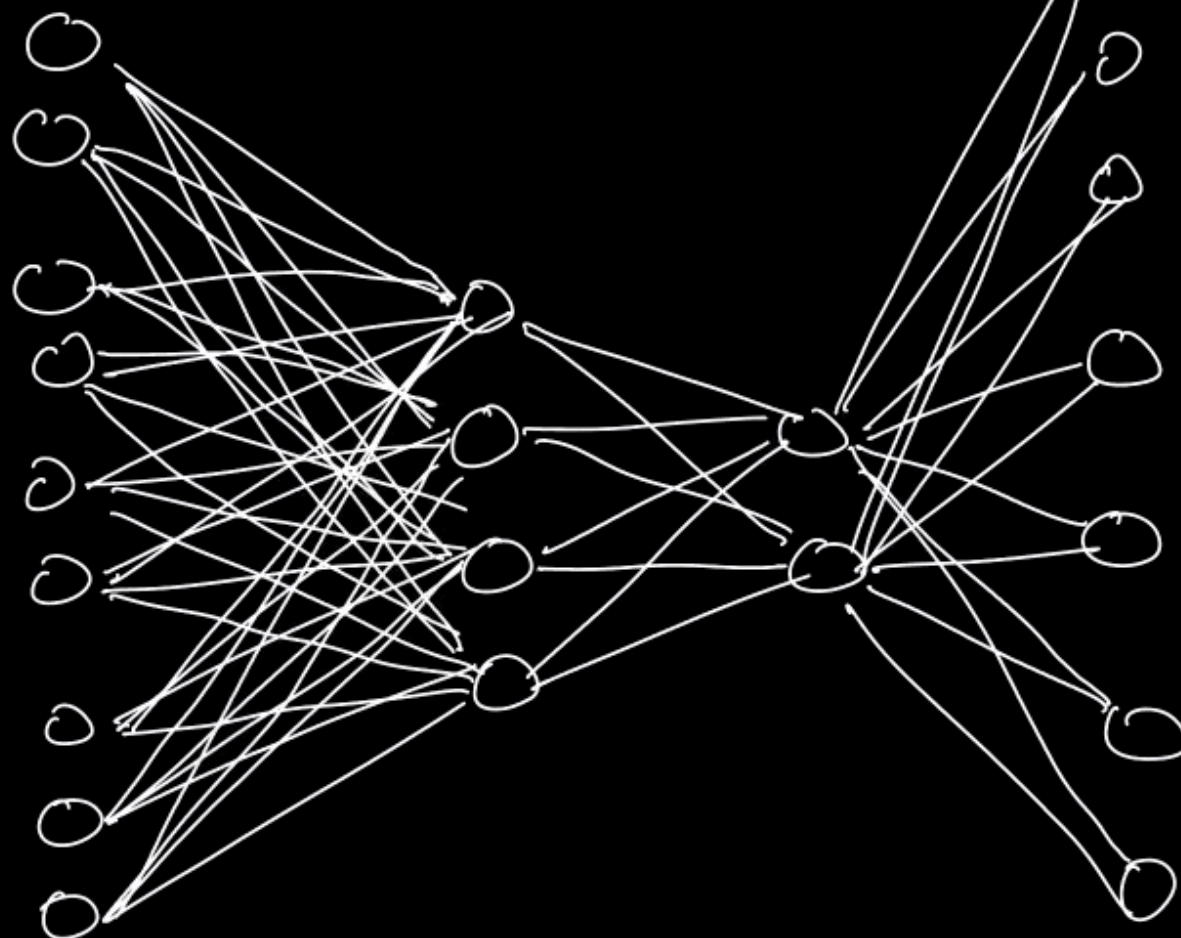
ab
ac
:
au
aen
:
be
bc
bcd
:
:

She added a banana to her ...

... mo

encoding of
input string

ab
bc
:
an
au
:
ba
bc
:
ba
bb
bc
:



prediction
of next token

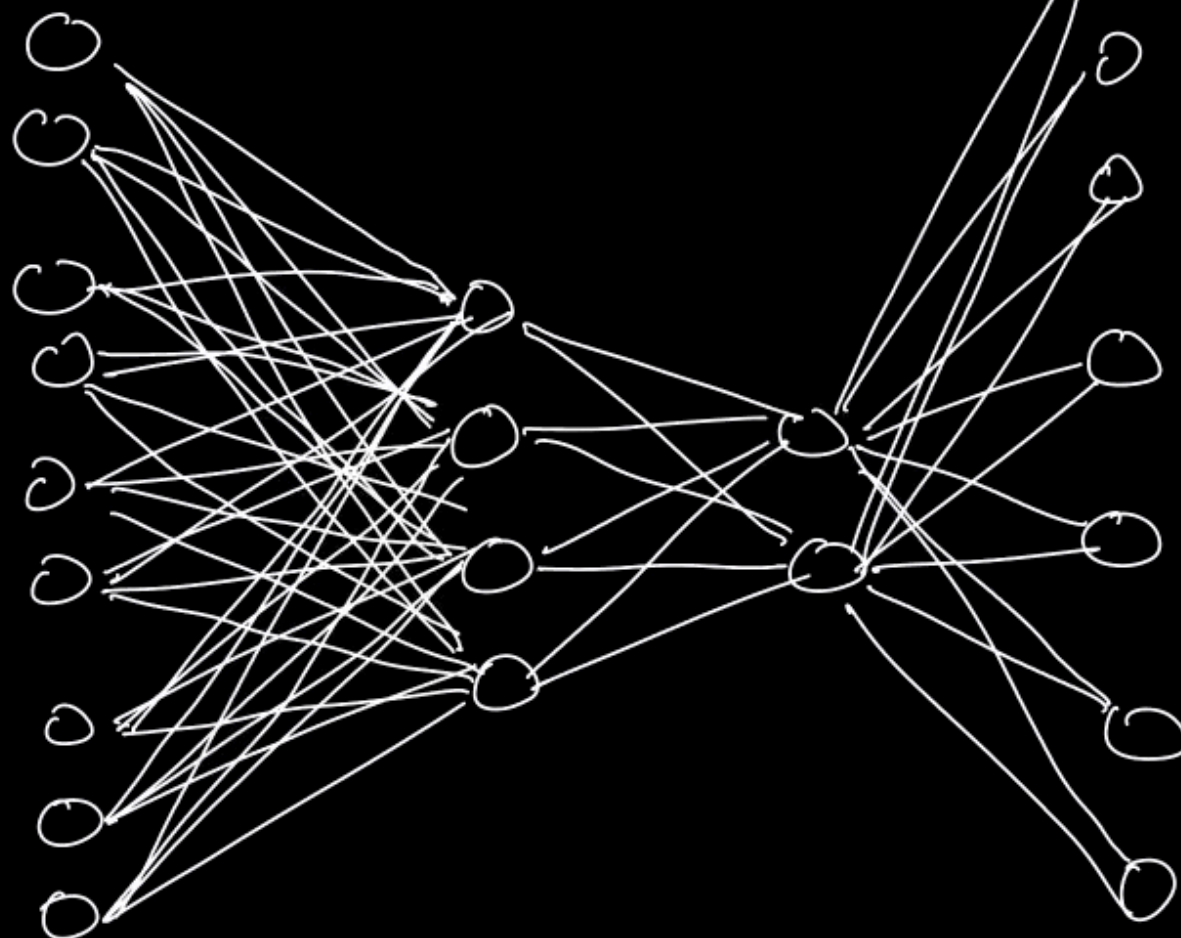
ab
ac
:
au
aen
:
be
bc
bcd
:

She added a banana to her mo...

... rn

encoding of
input string

ab
bc
:
an
au
:
ba
bc
:
ba
bb
bc
:
:



prediction
of next token

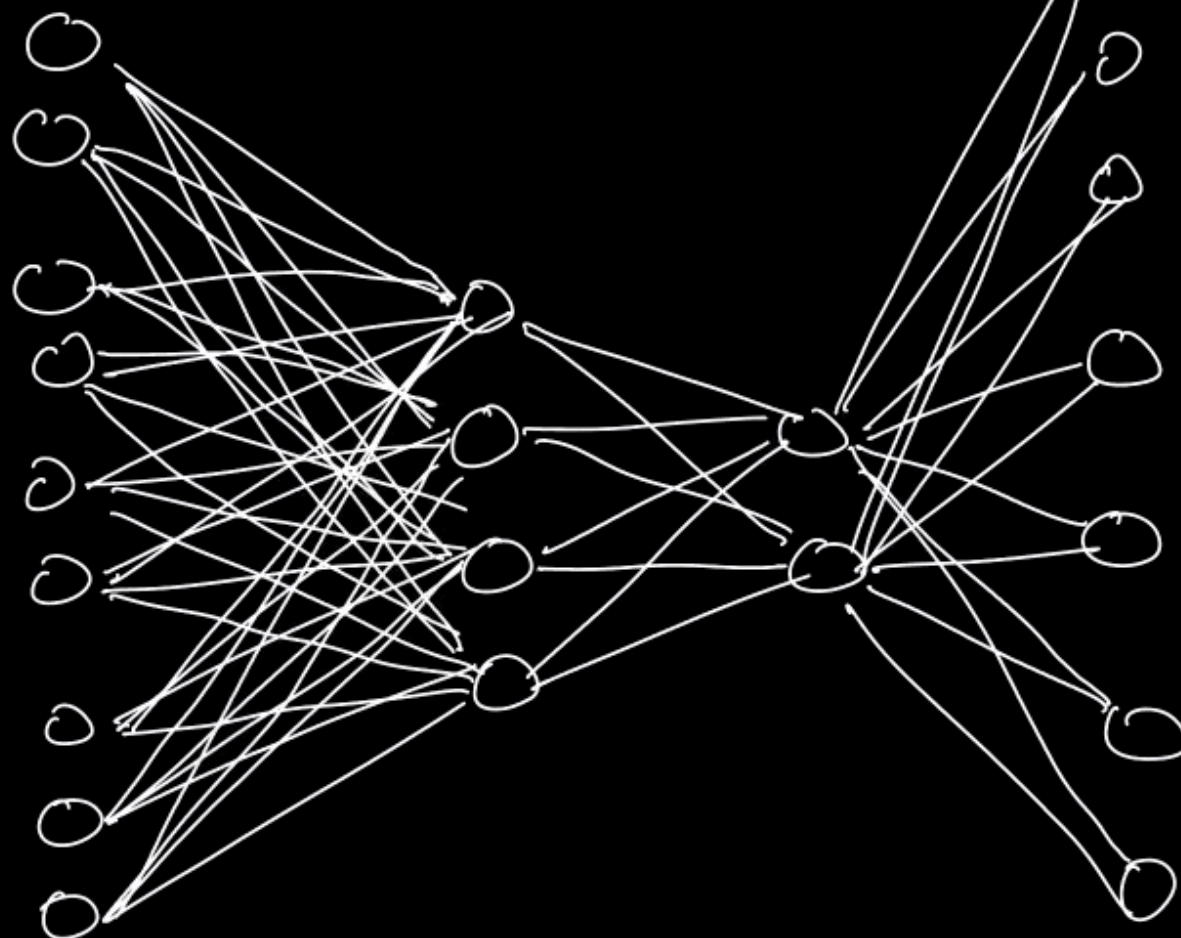
ab
ac
:
au
aau
:
ba
bc
ba
bb
bc
:
:

She added a banana to her morn...

... in

encoding of
input string

ab
bc
:
an
au
:
ba
bc
:
ba
bb
bc
:
:



prediction
of next token

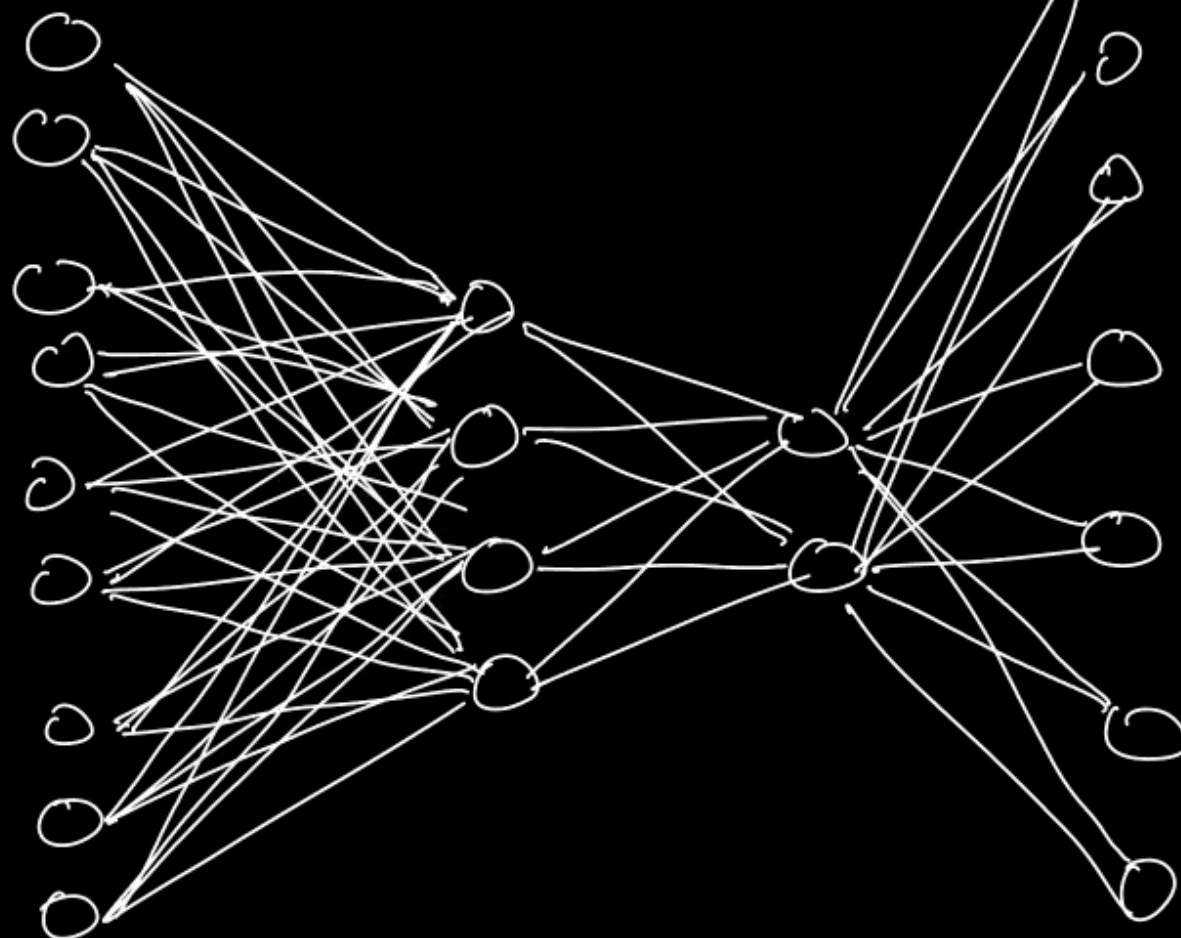
ab
ac
:
au
aau
:
ba
bc
ba
bb
bc
:
:

She added a banana to her mornin...

... g

encoding of
input string

ab
bc
:
ac
au
:
ba
bc
:
ba
ba
ba
:



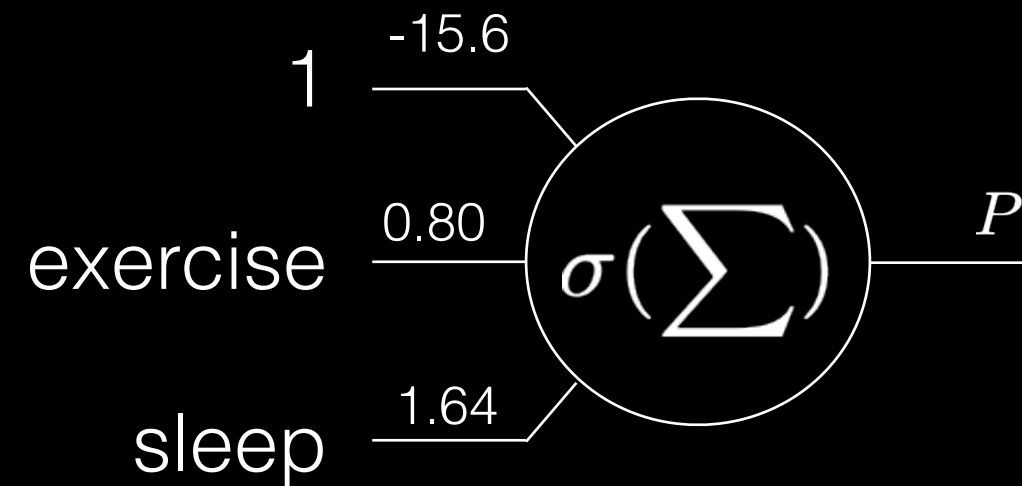
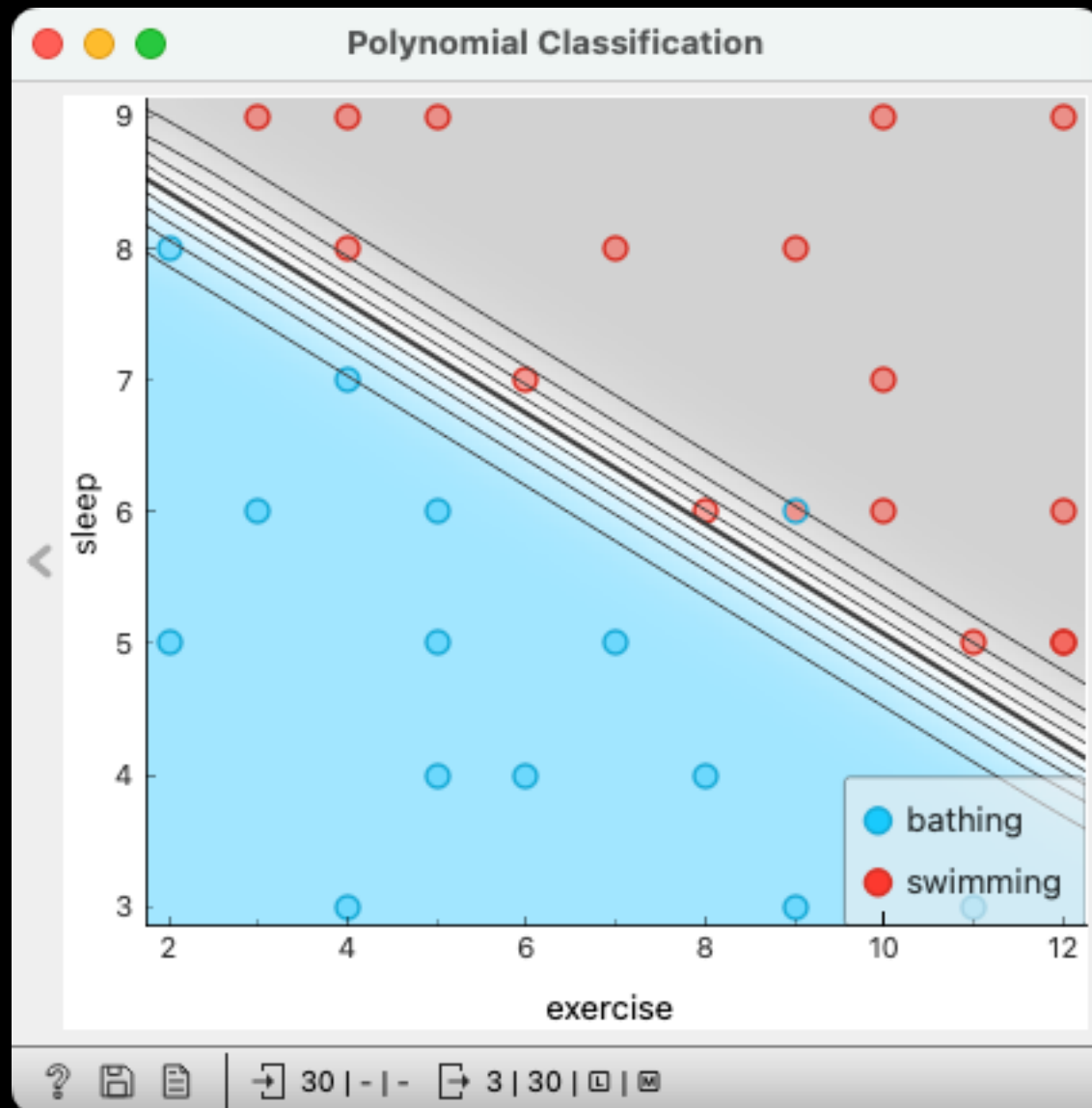
prediction
of next token

ab
ac
:
au
au
:
ba
bc
ba
ba
:

She added a banana to her morning smoothie.

Generative AI and LLM

- Context
- Encoding of context (sequence of tokens)
- (Iterative) output prediction (sequence of tokens)



Our swimming model had 3 parameters!

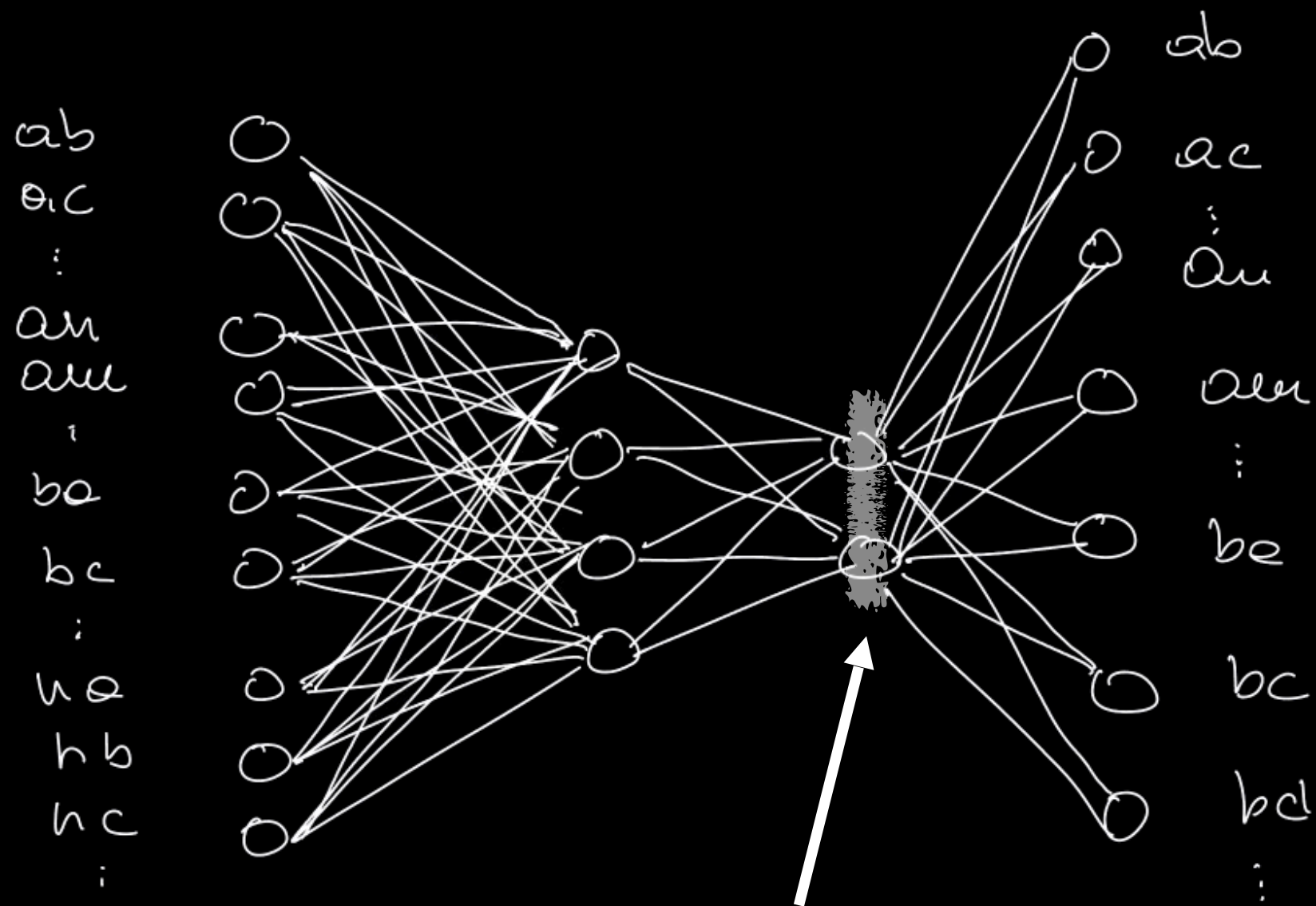
GPT-3.5 has 175 billion parameters!

GPT-4 has 1.76 trillion parameters!

That's 1,760,000,000,000!

ChatGPT

- A chat bot that uses trained LLM for text generation
- Not an AI!
- It's web-based interface is useful but rather very limited, the more interesting things to come are underneath



just numbers
let us see if they are useful