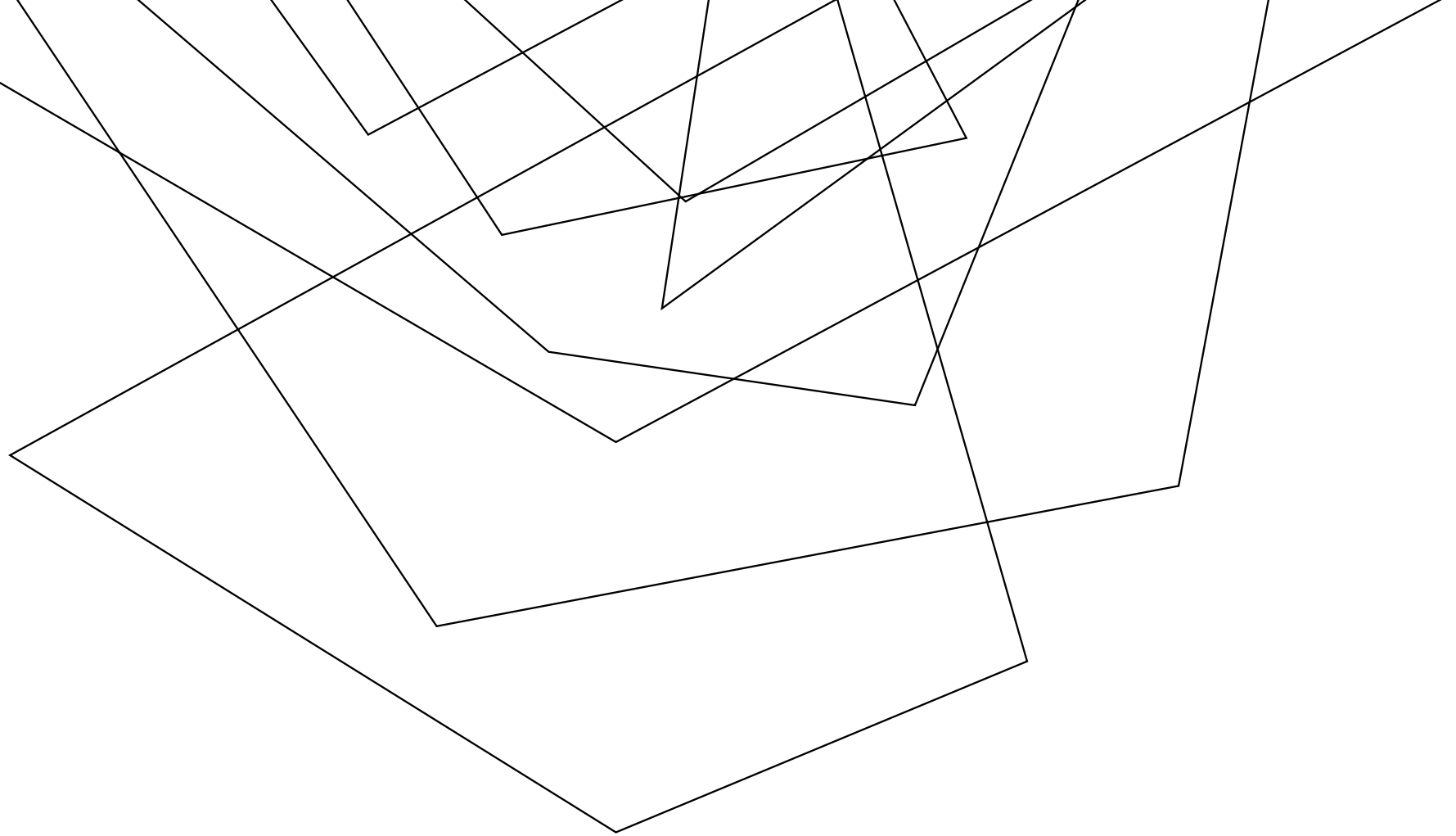


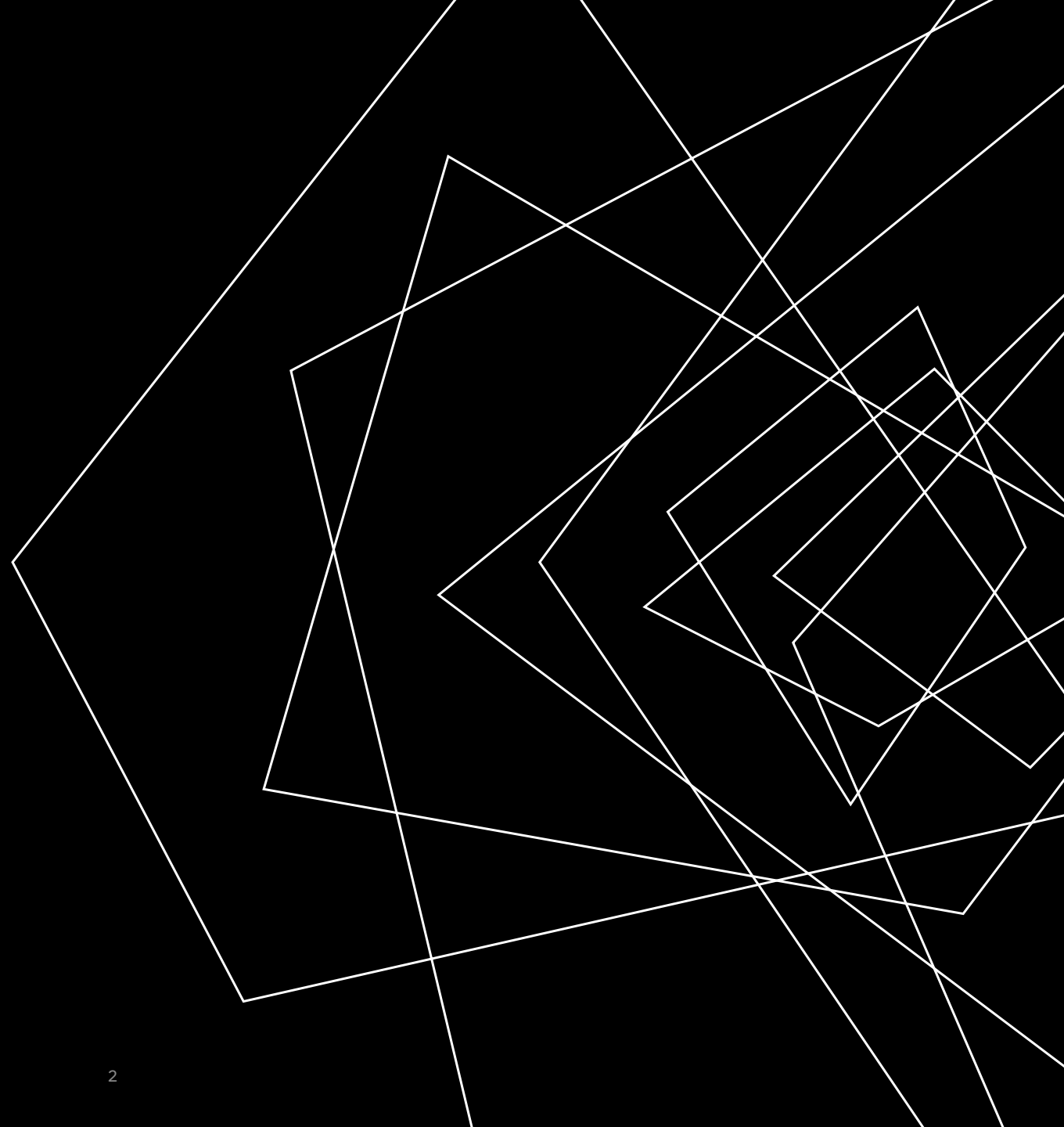
ARCHITECTURAL INTELLIGENCE

DigitalFUTURES Summer 2022

ABOUT

Samer Rahmeh (Sam)

- Co-Founder of Cali Technology Solutions (Jordan)
- Chief Technology Officer of SAVIR Virtual Learning & Robotics (Saudi Arabia)
- ML Engineer/Deep Learning Specialist. (MDFT Europe)
- CODE Architect [Architectural Programmer] (Philadelphia University)



WORKSHOP TOPICS

MACHINE LEARNING

Basics of Machine Learning, Linear Regression, Supervised/Unsupervised Learning

DEEP LEARNING

Basics of Labeled Dataset Creation, Training Deep Neural Networks, SAVE/LOAD ML Models.

AUGMENTED REALITY + AI

PREVIEW/MONITOR Artificial Neural Network using AR

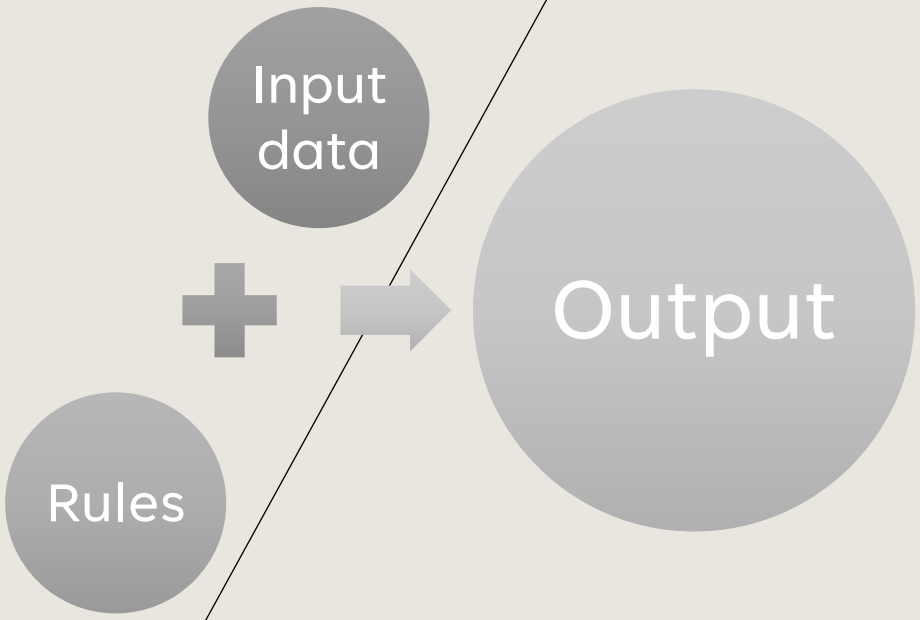
FRACTAL DESIGN

Basics of 4^{th} Dimensional shapes, Understanding Bicomplex numbers, Parallel Computing (GPU) [Real-Time Simulation]

WHAT IS MACHINE LEARNING?

In traditional coding, the programmer provides the rules (written in a programming language) that act on the input data.

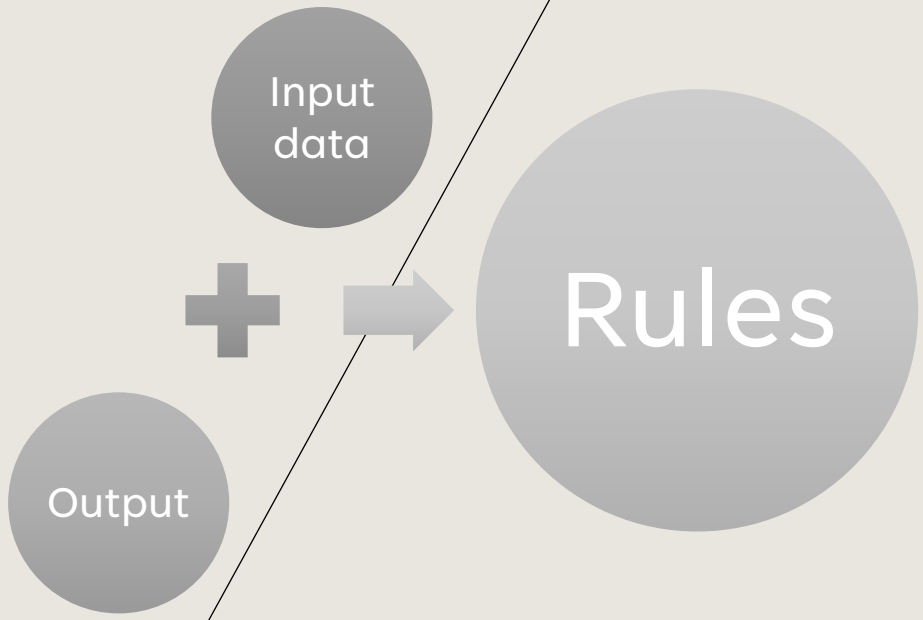
However, it is very difficult to provide rules when the input data contains so many situations or scenarios that need to be taken into consideration.



WHAT IS MACHINE LEARNING?

In machine learning, we provide input data and output, while the compute figures out the rules.

We **teach** the machine to find patterns within data.



EXAMPLES

Say we want to write a code that can tell if an image contains a cat or a dog

Traditional way

We have to make the code inclusive for every possible shape, breed, position, etc...


Cat


Dog

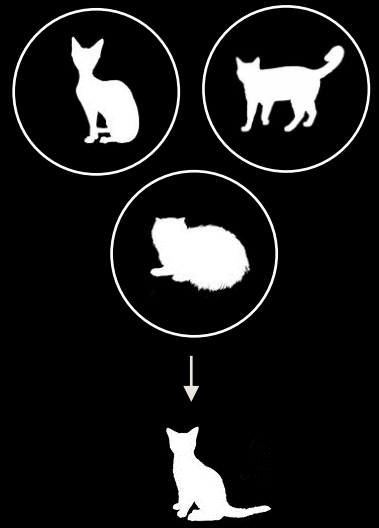

Cat?

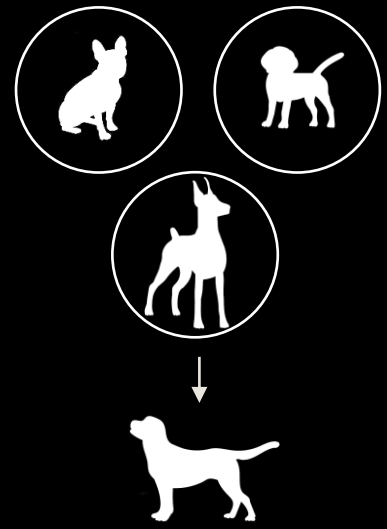

D...og?


Help...?

Machine learning

We provide the machine with images for cats and dogs. The machine learns to identify and differentiate between them.


Cat


Dog

EXAMPLES

WHAT IS THE RELATIONSHIP BETWEEN X AND Y?

X = 0, 1, 2, 3, 4, 5

Y = 1, 3, 5, 7, 9, 11

EXAMPLES

WHAT IS THE RELATIONSHIP BETWEEN X AND Y?

X = 0, 1, 2, 3, 4, 5

Y = 1, 3, 5, 7, 9, 11

Human brain:

$Y = 2x + 1$

EXAMPLES

Machine:

1 neuron

Feeds network with single x to predict y

```
model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1])])
model.compile(optimizer='sgd', loss='mean_squared_error')
xs = np.array([0.0, 1.0, 2.0, 3.0, 4.0, 5.0], dtype=float)
ys = np.array([1.0, 3.0, 5.0, 7.0, 9.0, 11.0], dtype=float)
model.fit(xs, ys, epochs=500)
Print(model.predict([7.0])
```

→ Defines model (trained neural network)

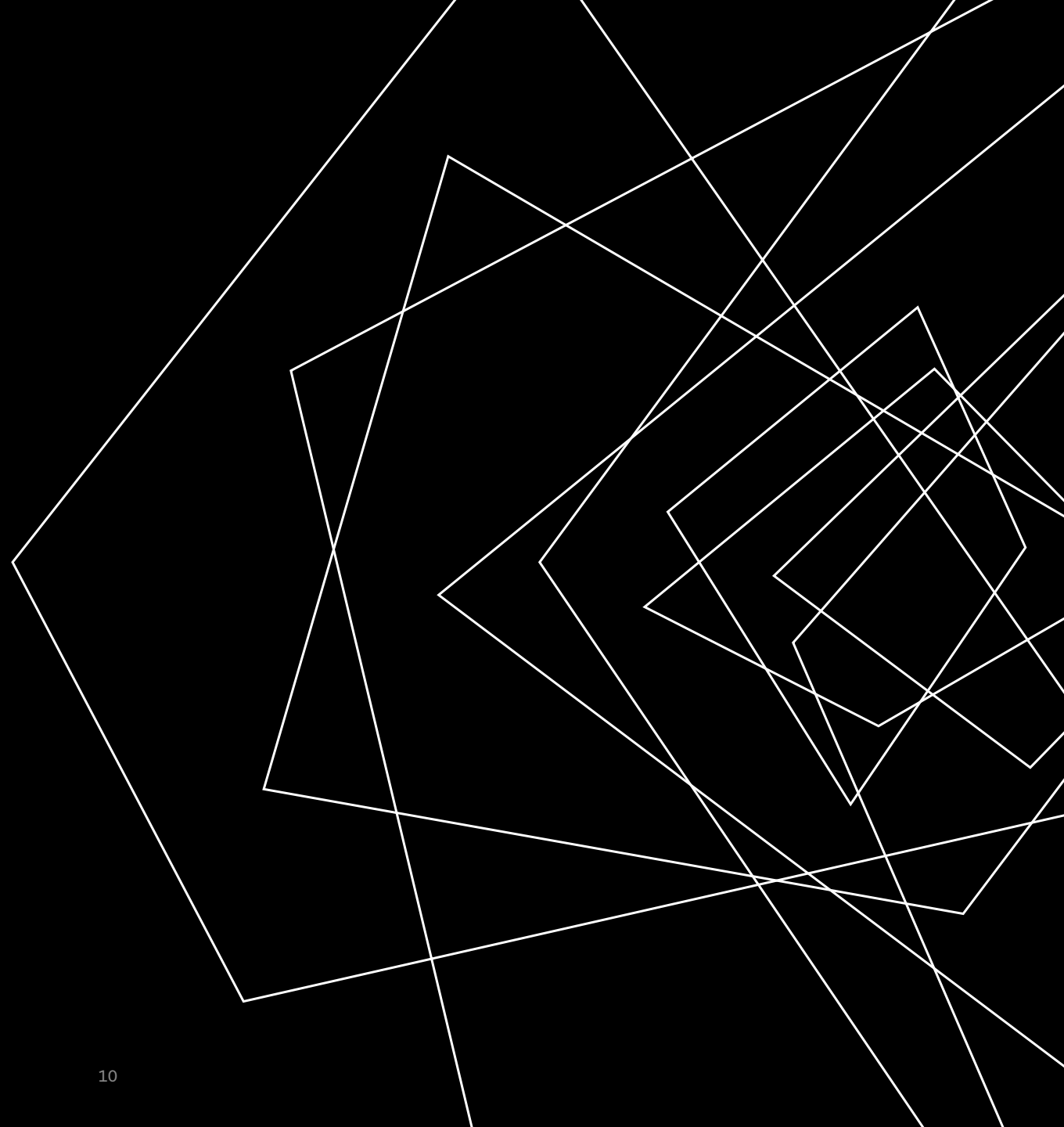
→ Loss: calculates how good the prediction is
optimizer: generates a new enhanced prediction

→ Fit xs to ys
Repeat 500 times

→ Predict y for x = 7.0

What do you think the prediction will be?

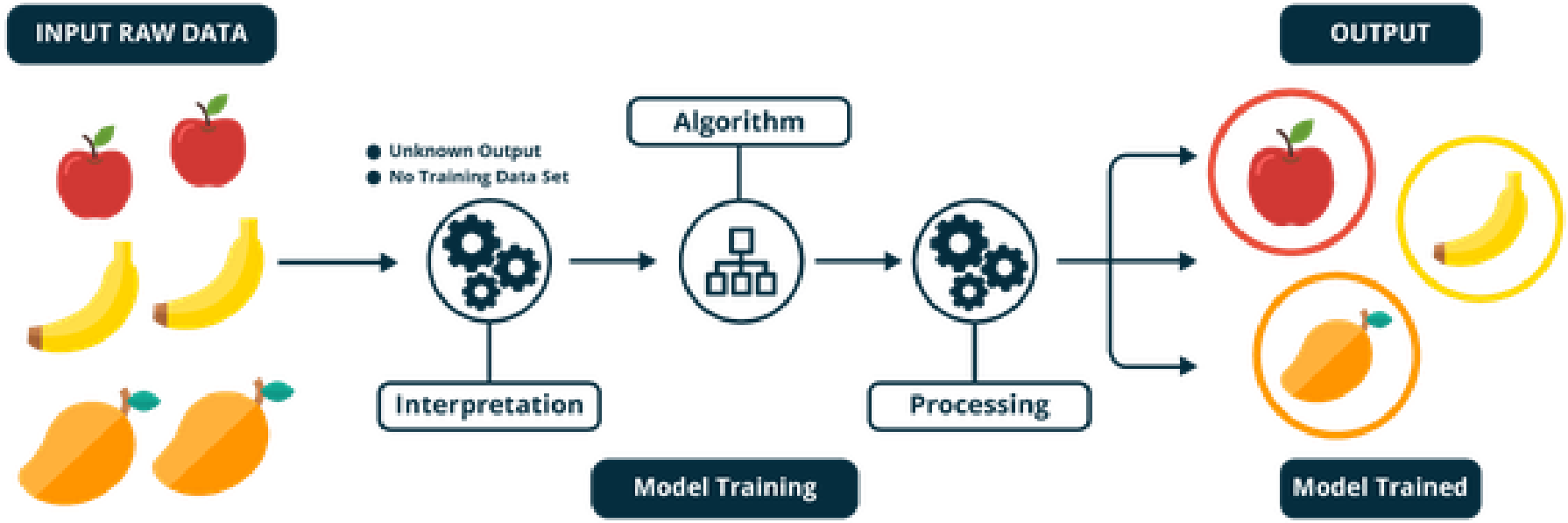
CODE_0.GH



WHAT IS UNSUPERVISED LEARNING?

a type of machine learning that learns patterns from untagged data. The hope is that through mimicry, which is an important mode of learning in people, the machine is forced to build a compact internal representation of its world and then generate imaginative content from it.

EXAMPLE OF UNSUPERVISED LEARNING



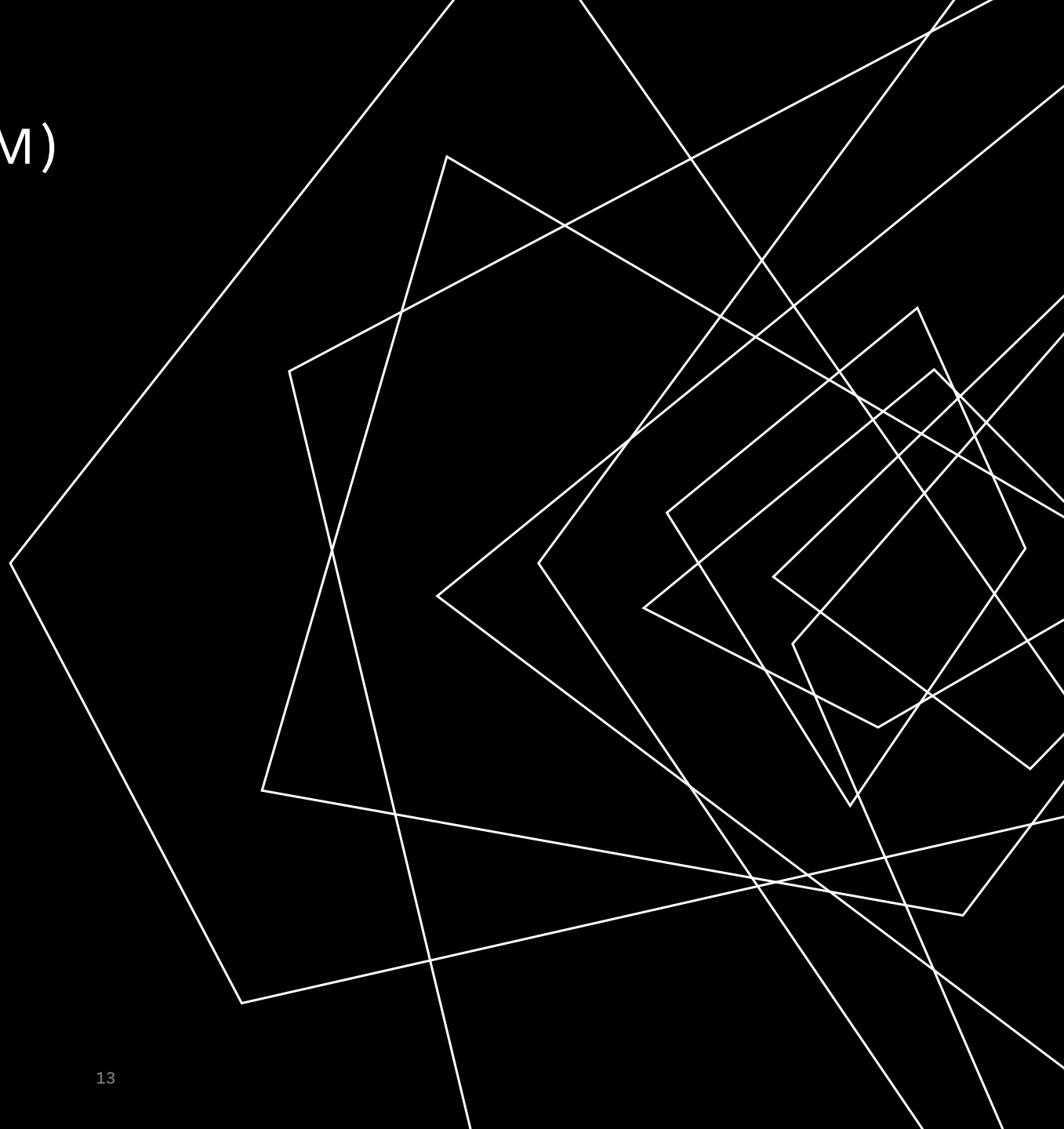
SELF-ORGANIZING MAPS (SOM)

Kohonen Network

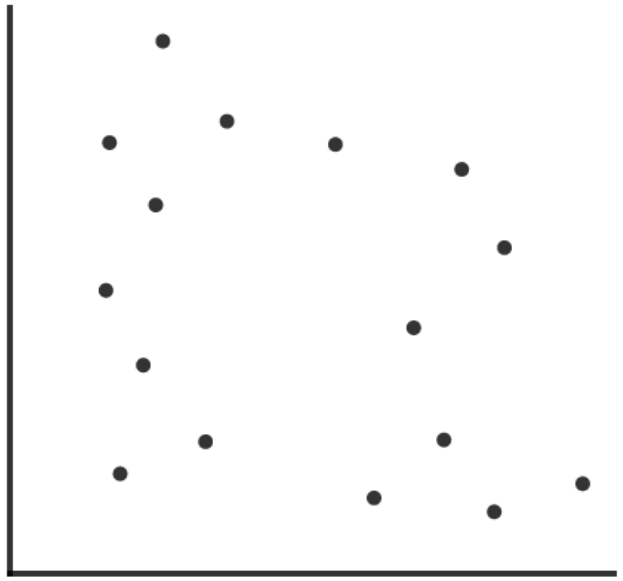
is a computational method for the visualization and analysis of high-dimensional data, especially experimentally acquired information.

How does it work?

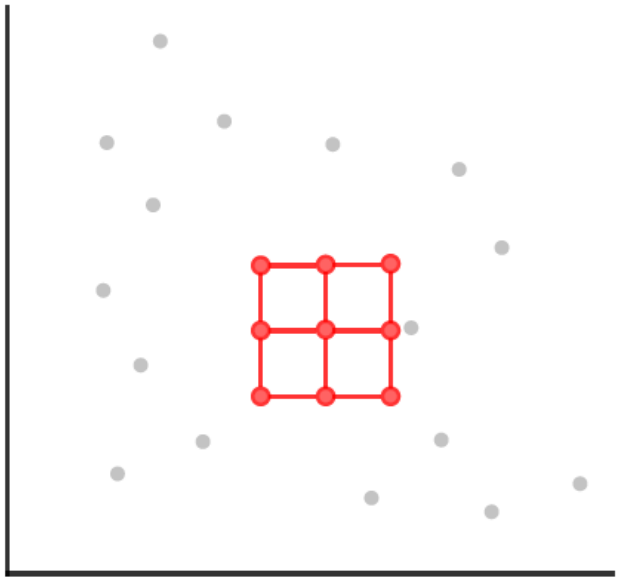
- fixed grid topology between neurons.
- neighborhood connections modified through a neighborhood function (gaussian or Mexican hat).
- each input represents a data vector of n dimensions
- each neuron is also reprinted in this n-dimensional space



SELF-ORGANIZING MAPS (SOM)

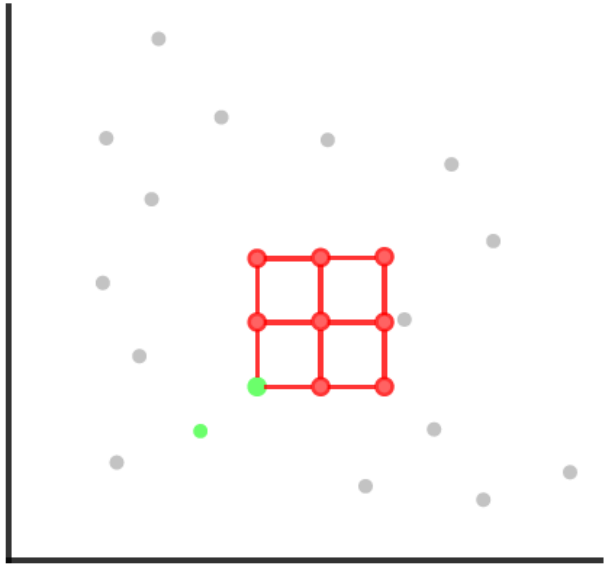


unlabeled data set
(2D)

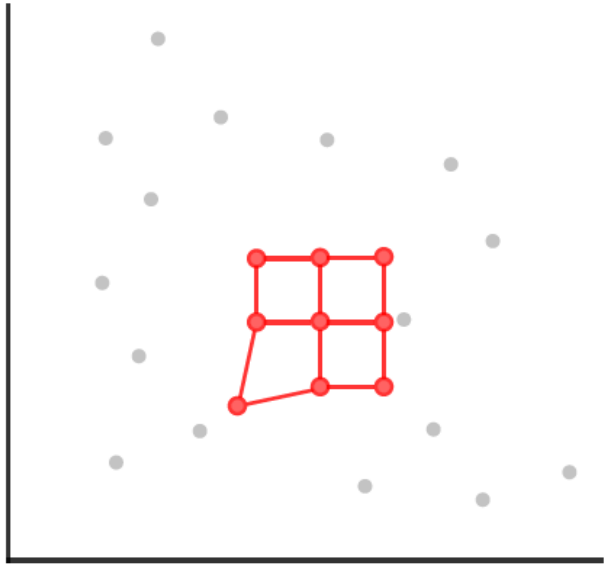


- defined neural network topology, e.g.:
  3x3 network of two dimensions

SELF-ORGANIZING MAPS (SOM)

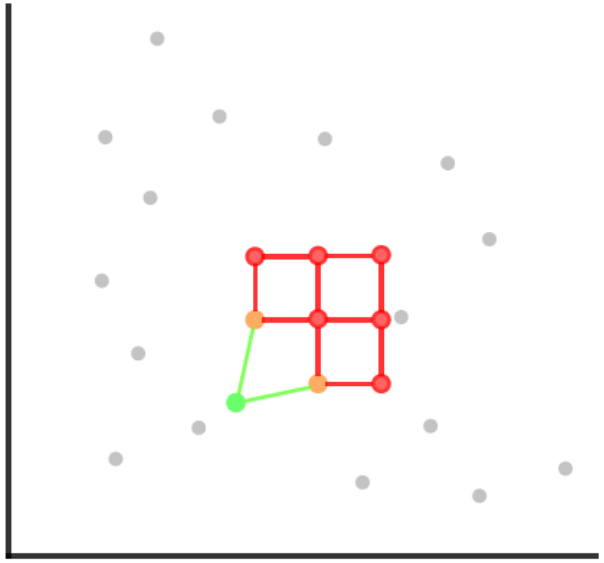


- start comparing data samples to neurons in random order
- find closest fit in set of neurons (winner neuron)

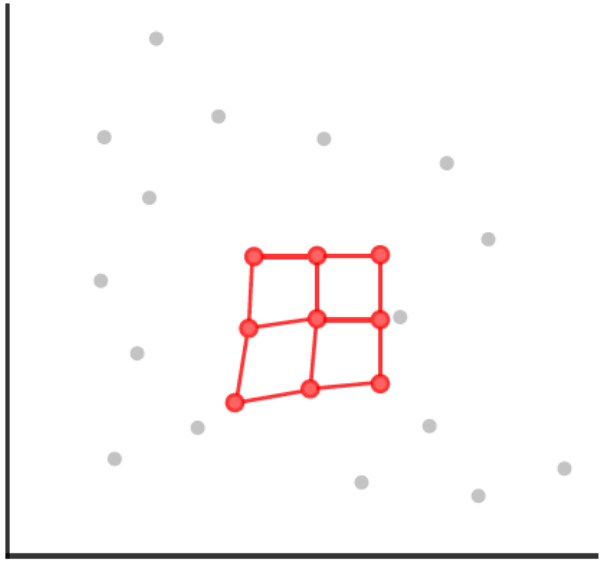


- adjust winner neurons position to respective data sample according to predefined *learning rate*

SELF-ORGANIZING MAPS (SOM)

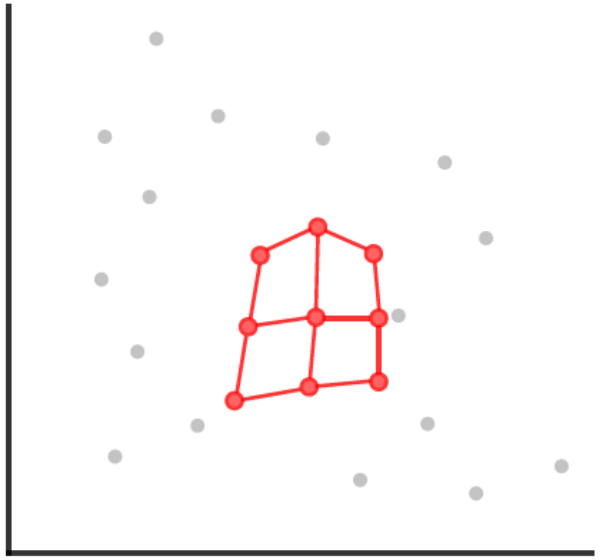


- spread out the position update information to the winner neurons to-polical neighbors

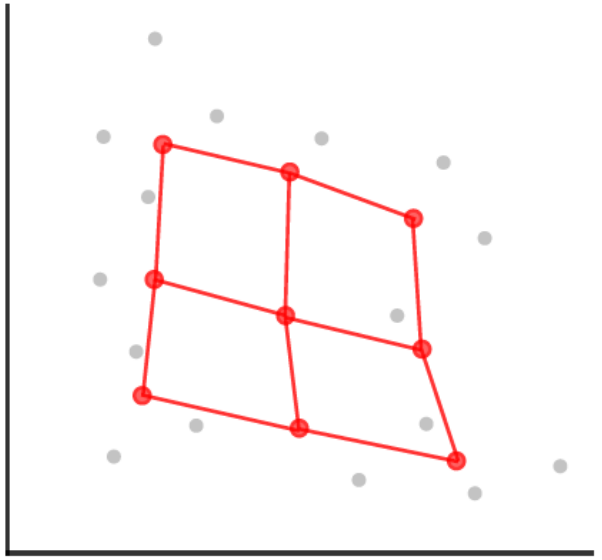


- adjust neighbor neuron positions according to pre-defined neighborhood function

SELF-ORGANIZING MAPS (SOM)

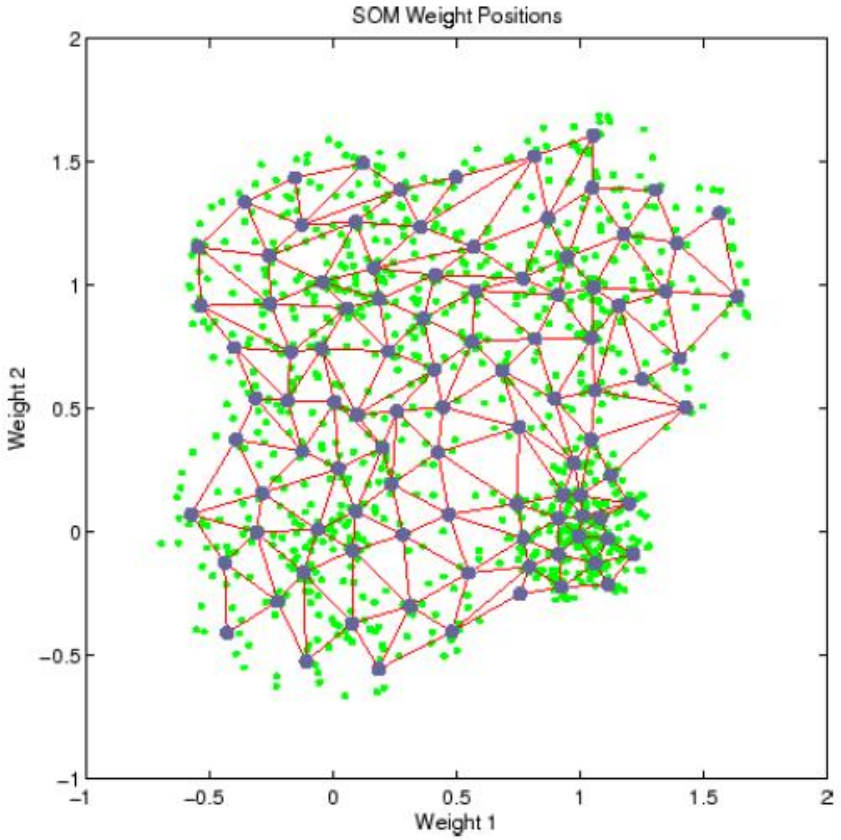


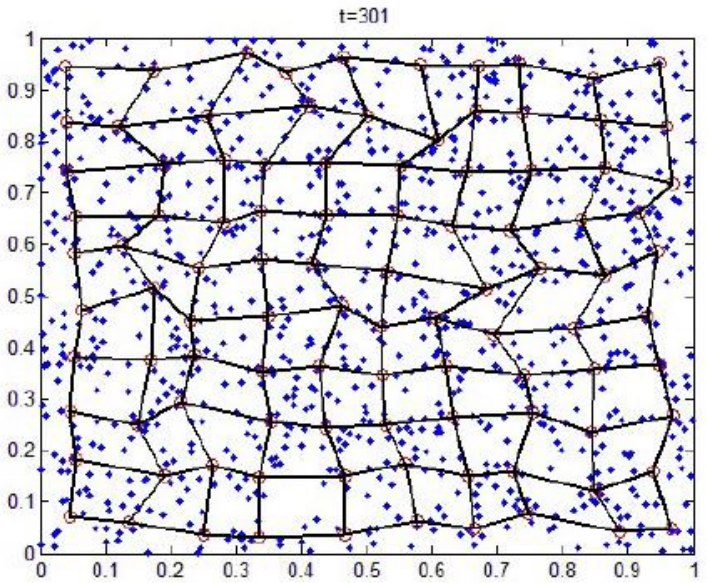
- repeat for different data sample
- repeat repeat repeat ....



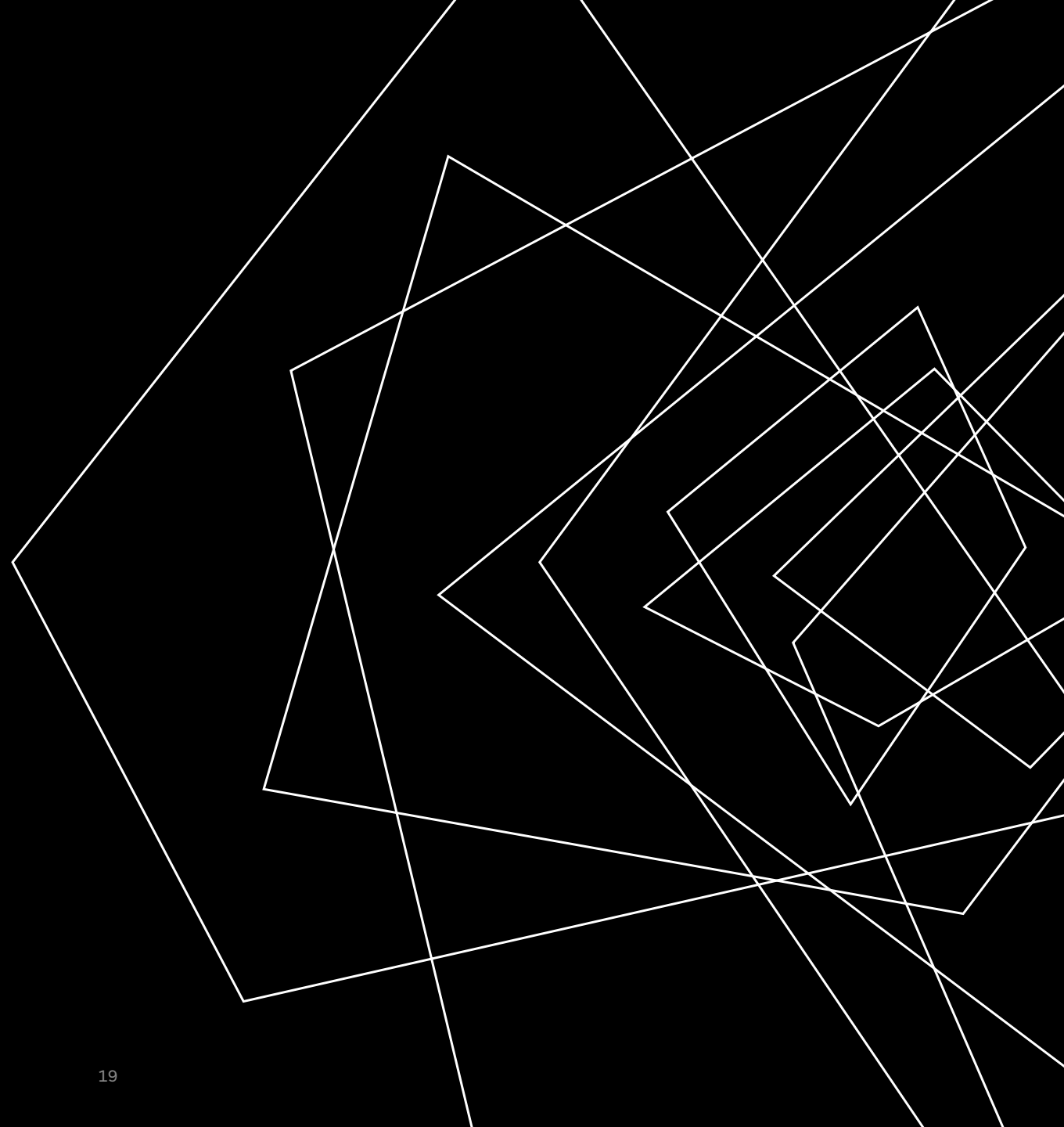
- network topology will (hopefully) fit and approximate the data

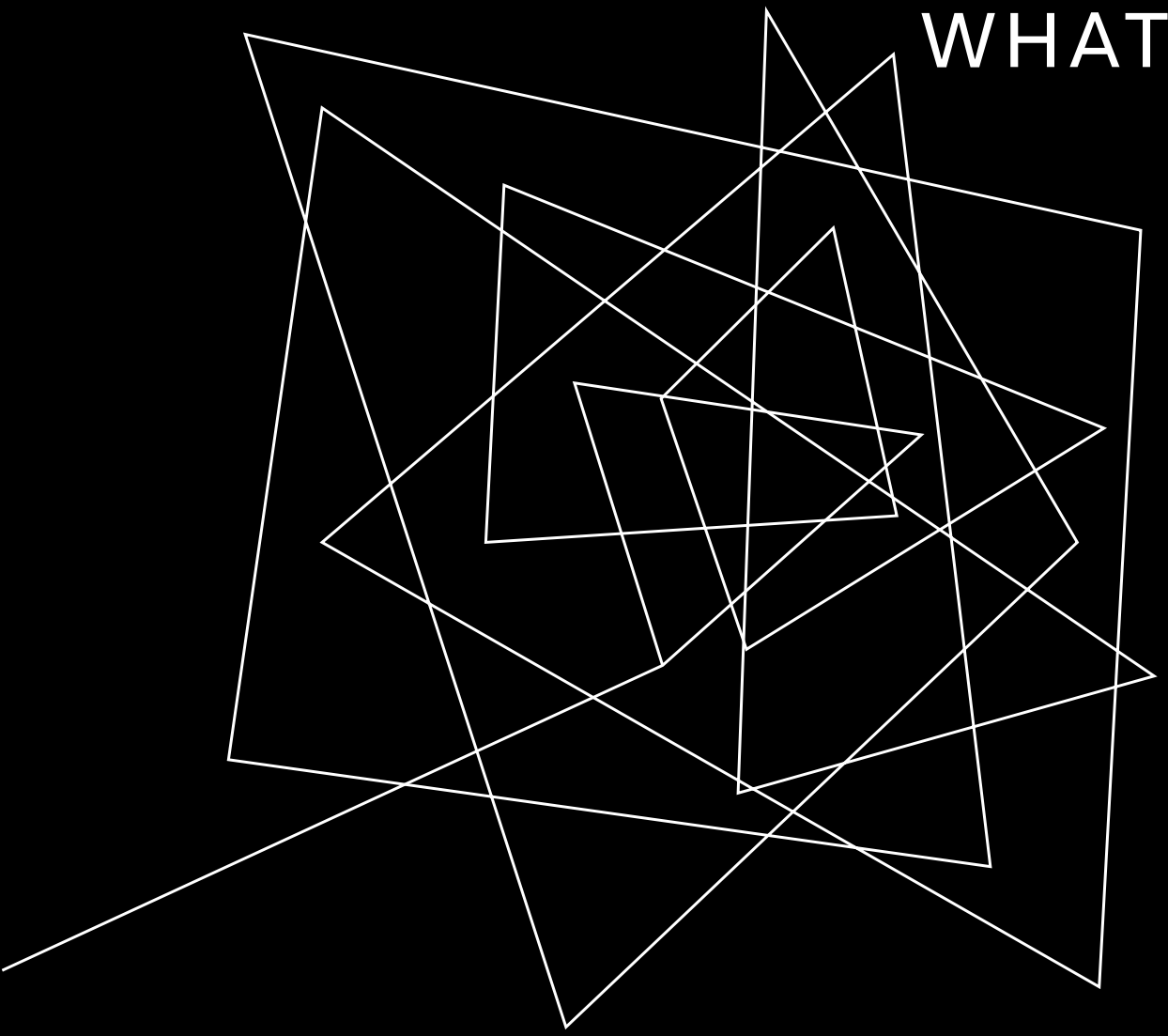
SELF-ORGANIZING MAPS (SOM)





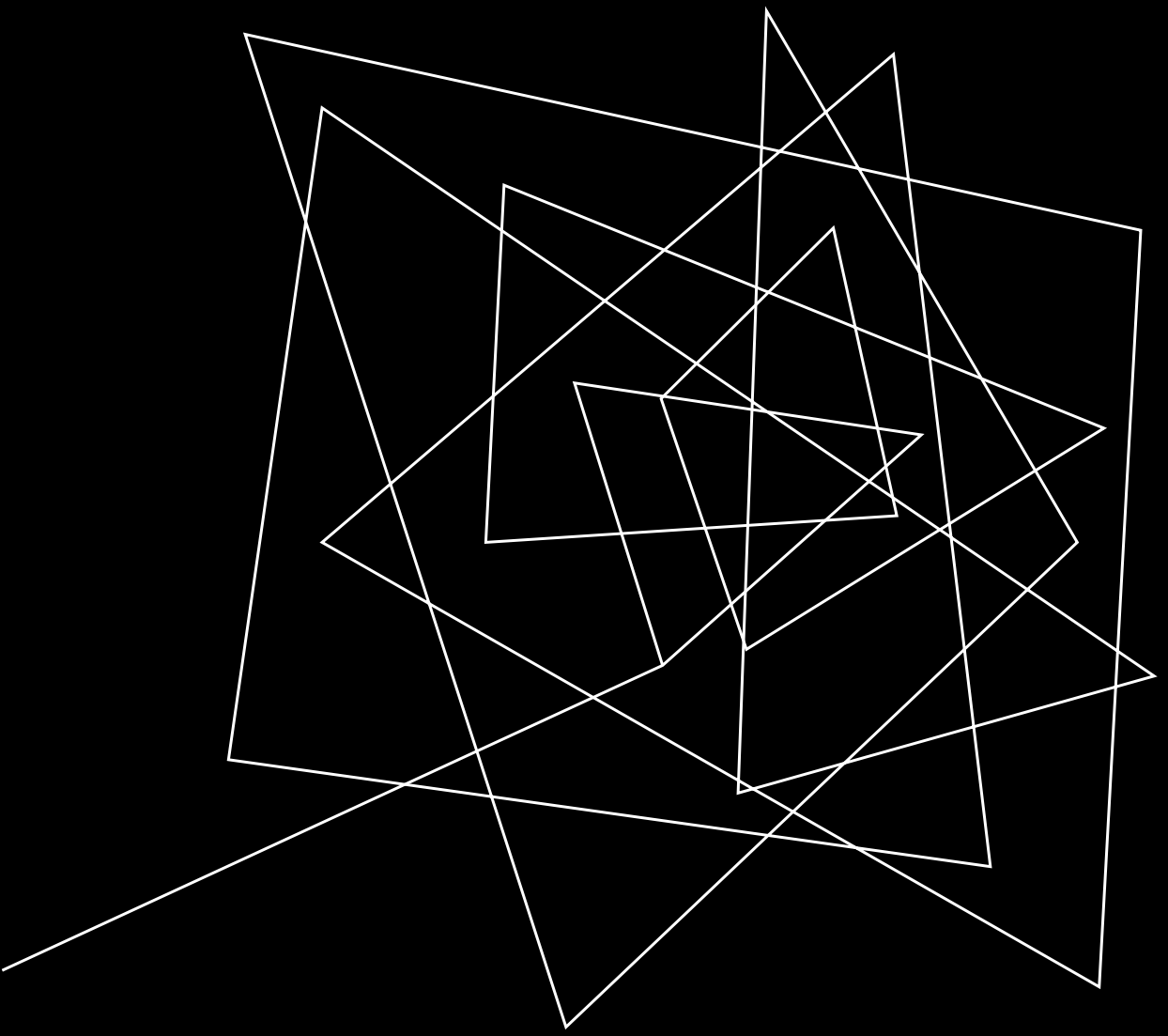
CODE_1.GH
CODE_2.GH





WHAT HAVE WE LEARNED TODAY?

HOW TO EMBED IT IN ARCHITECTURE?



THANK YOU ☺