CS 480

Introduction to Artificial Intelligence

November 22, 2022

Announcements / Reminders

- **Final Exam: December 1st!**
  - **Ignore Registrar date for CS 480**
  - **Online section 02: please contact Mr. Charles Scott (scott@iit.edu) to make arrangements if necessary**

- **End of semester course evaluation: opened**

- **Quiz #04: due on Monday (11/28)**
- **Written Assignment #04: due on Wednesday (11/30)**

Plan for Today

- **Casual Introduction to Machine Learning**

Main Machine Learning Categories

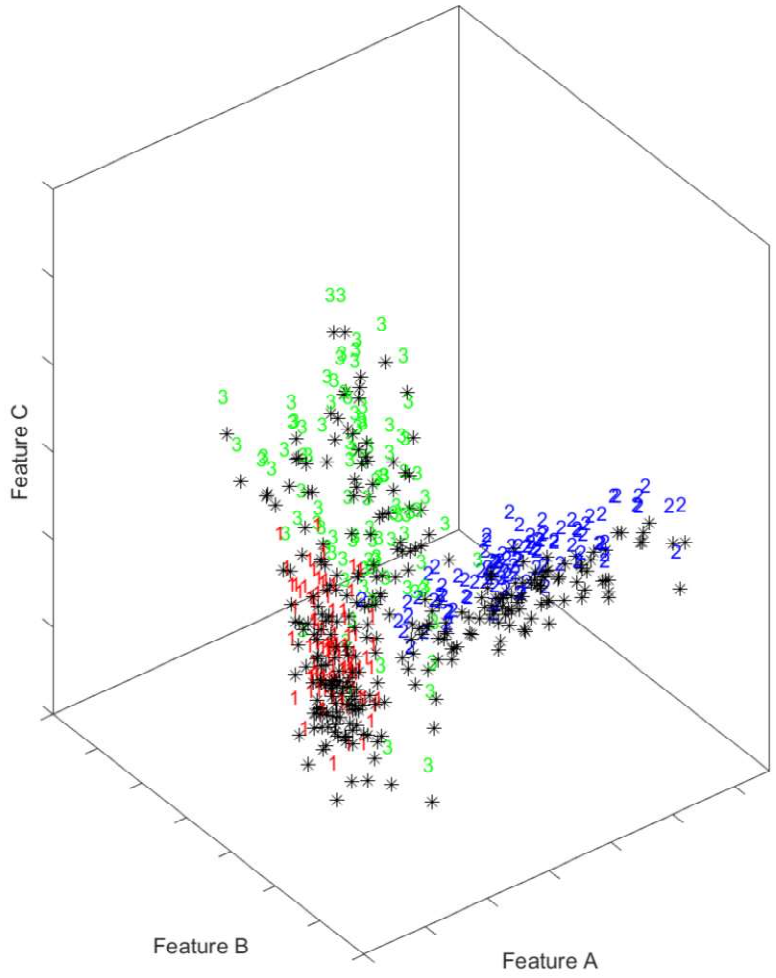
Supervised learning	Unsupervised learning	Reinforcement learning
Supervised learning is one of the most common techniques in machine learning. It is based on **known relationship(s) and patterns within data** (for example: relationship between inputs and outputs). Frequently used types: **regression**, and **classification**.	**Unsupervised learning** involves finding underlying patterns within data. Typically used in **clustering** data points (similar customers, etc.)	Reinforcement learning is inspired by behavioral psychology. It is **based on a rewarding / punishing an algorithm**. Rewards and punishments are based on algorithm's action within its environment.

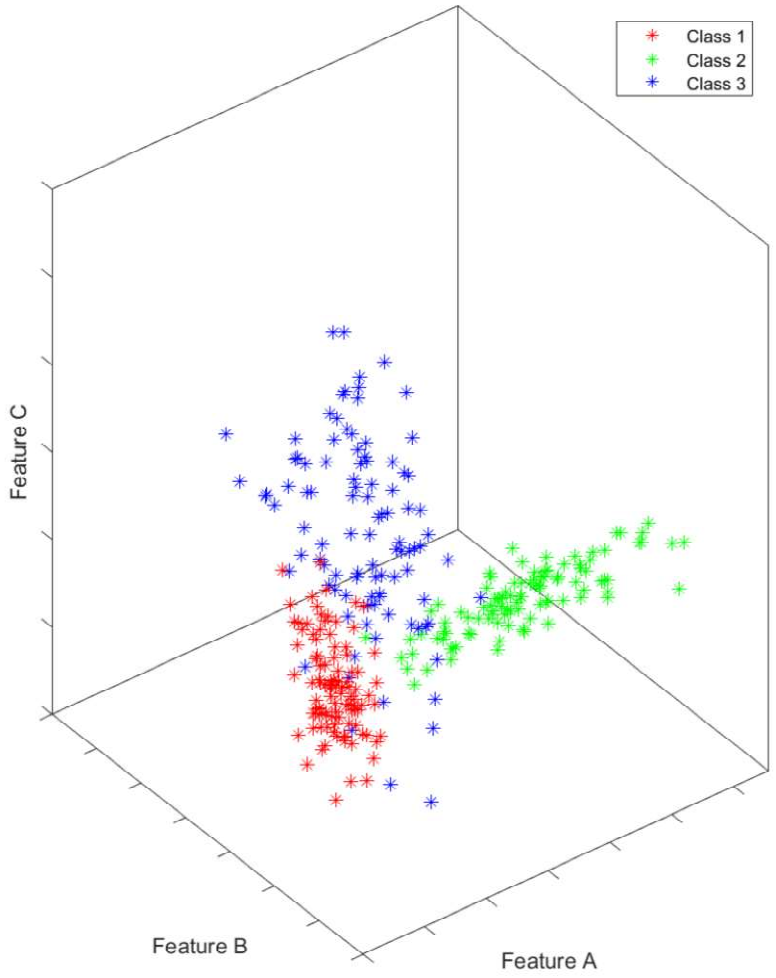
Training / Validation / Test Sets

In order to create the best model possible, given some (relatively large) data set, we should divide it into:

- **training set: to train candidate models**
- **validation set: to evaluate candidate models and pick the best one**
- **test set: to do the final evaluation of the model**

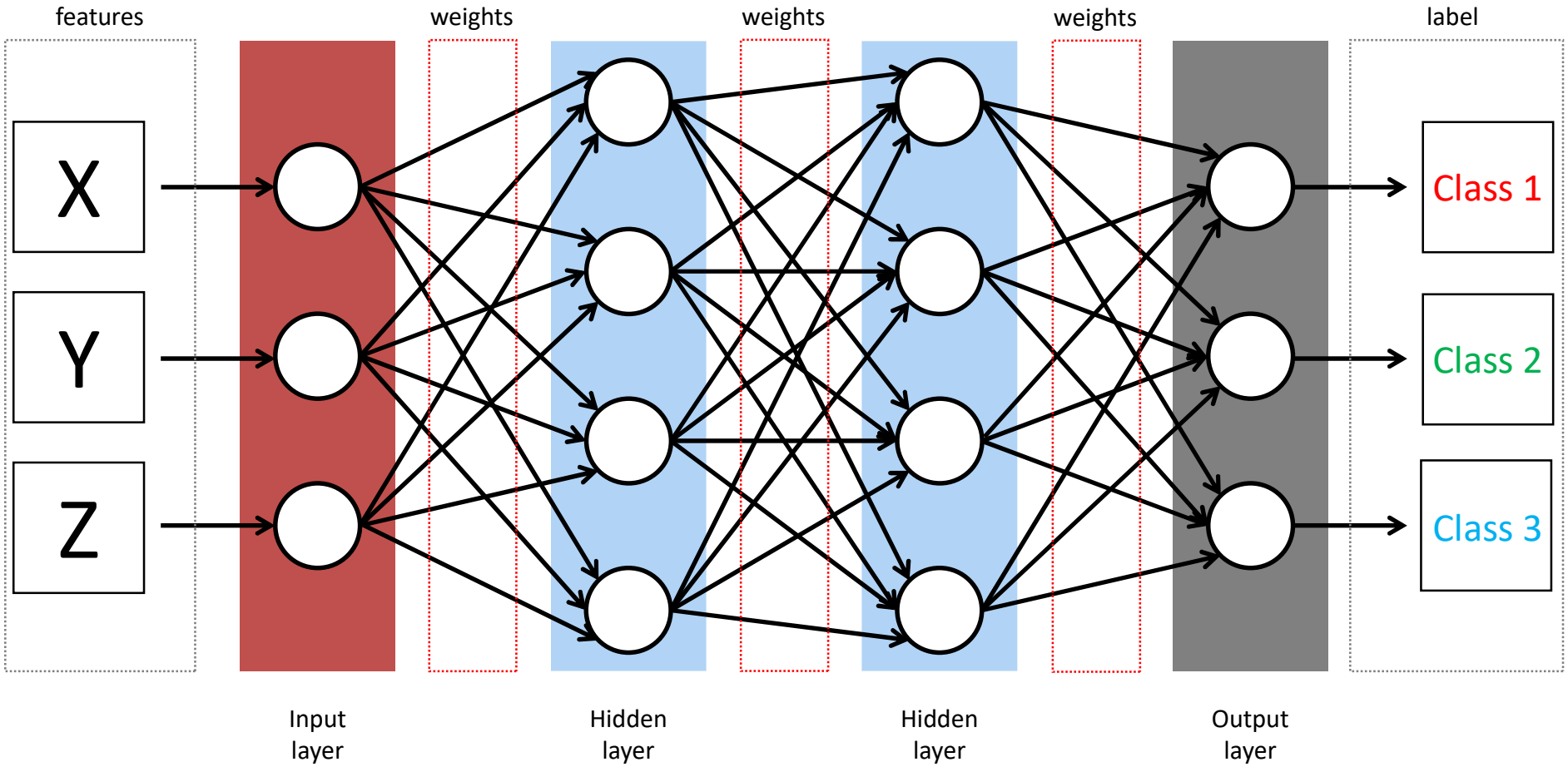
Data Set: Labeled Data





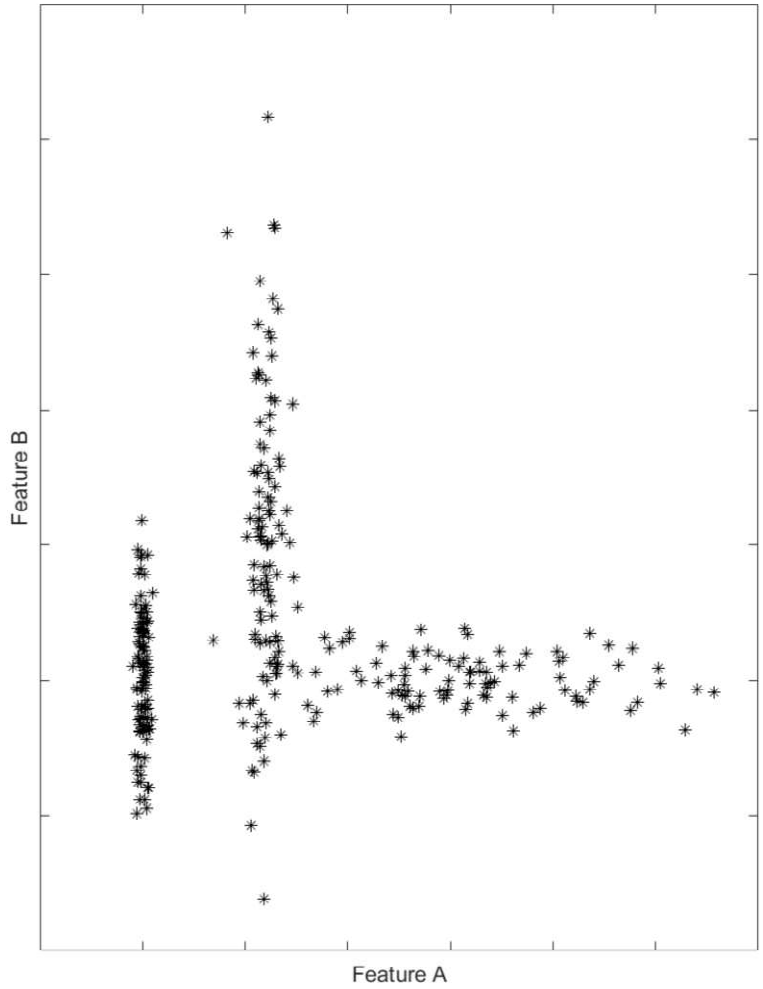
ANN: Supervised Learning

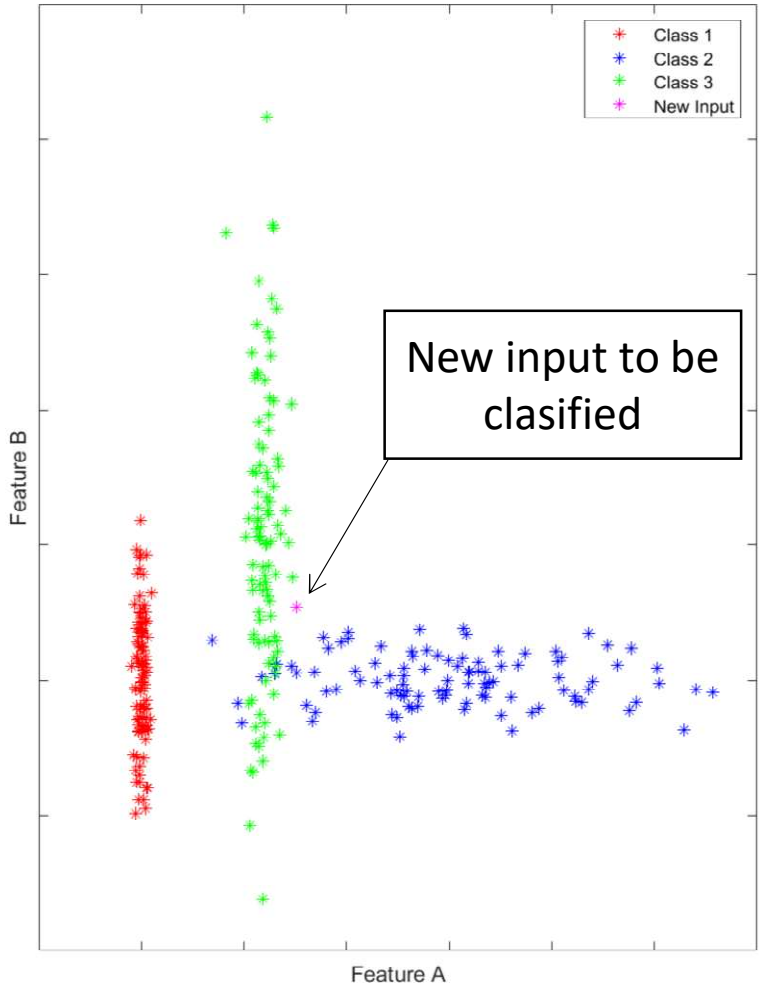
In order to work properly a classifier **needs to be trained** first with **labeled data**.



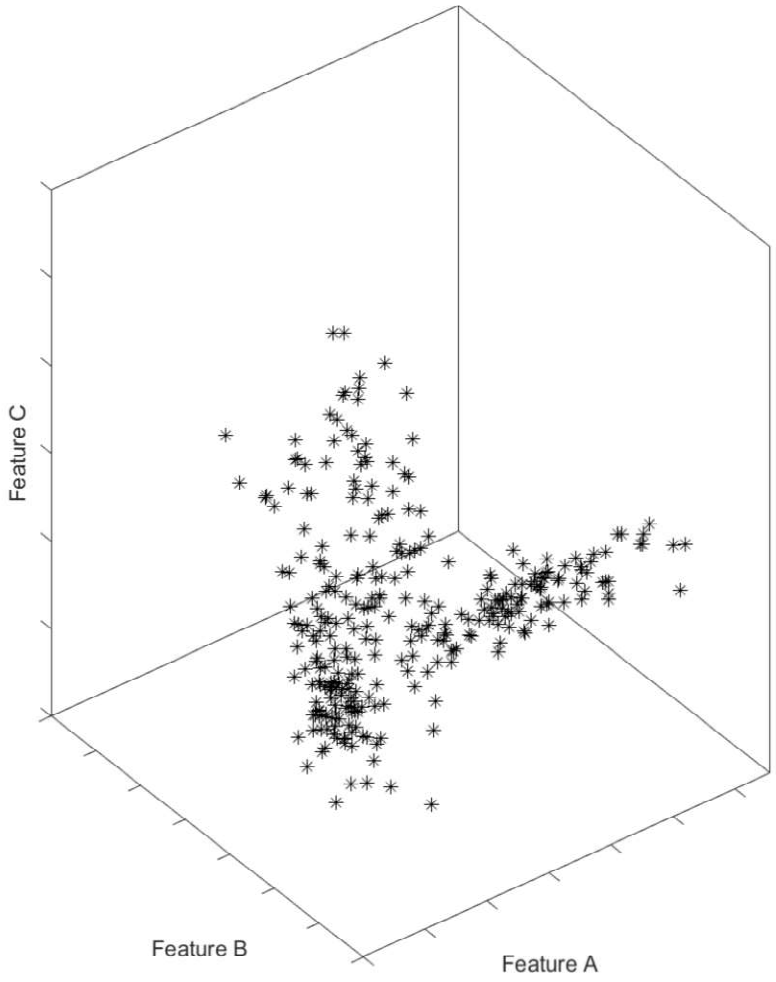
Training will **adjust all the weights** within this artificial neural network.

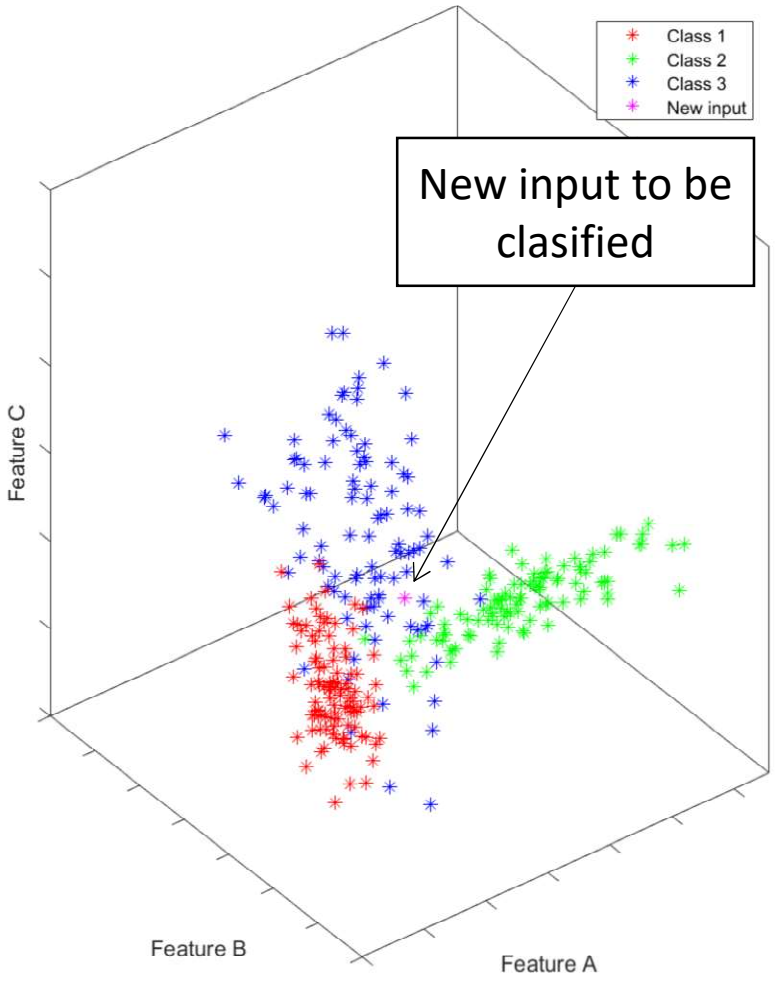
Supervised Learning: New Input



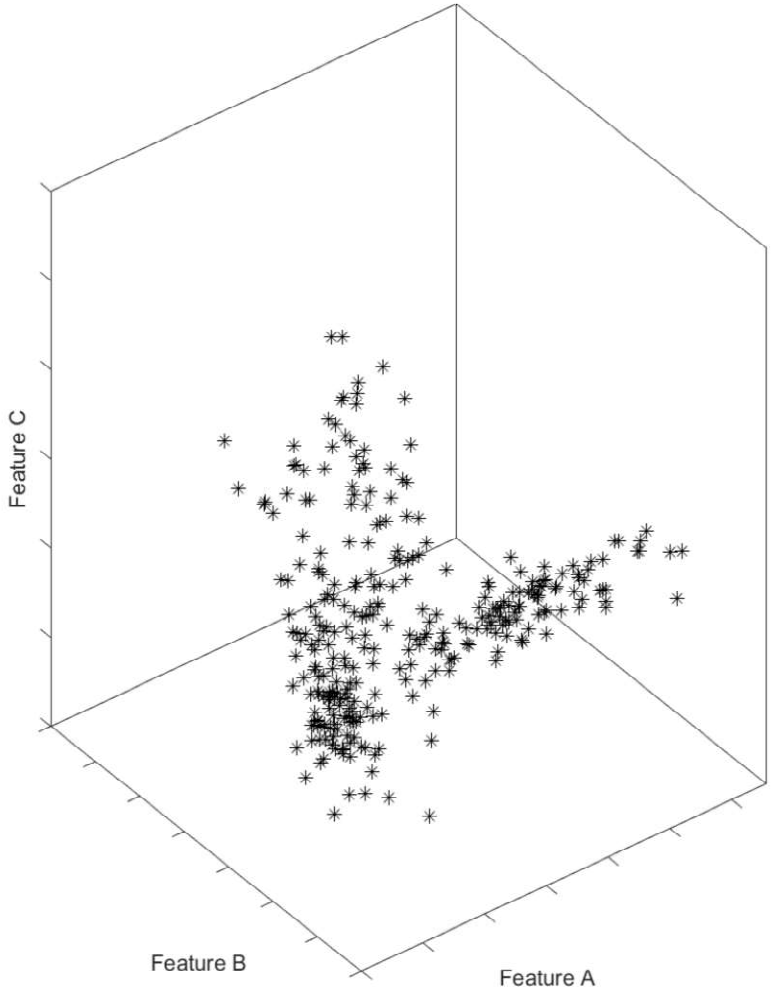


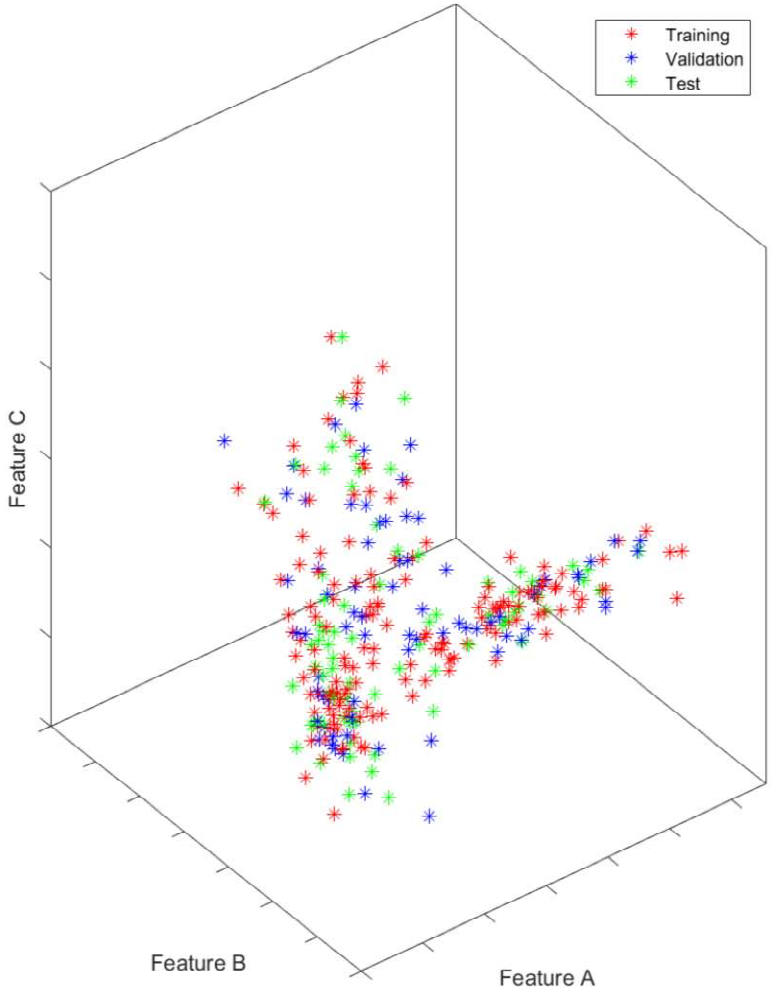
Supervised Learning: New Input



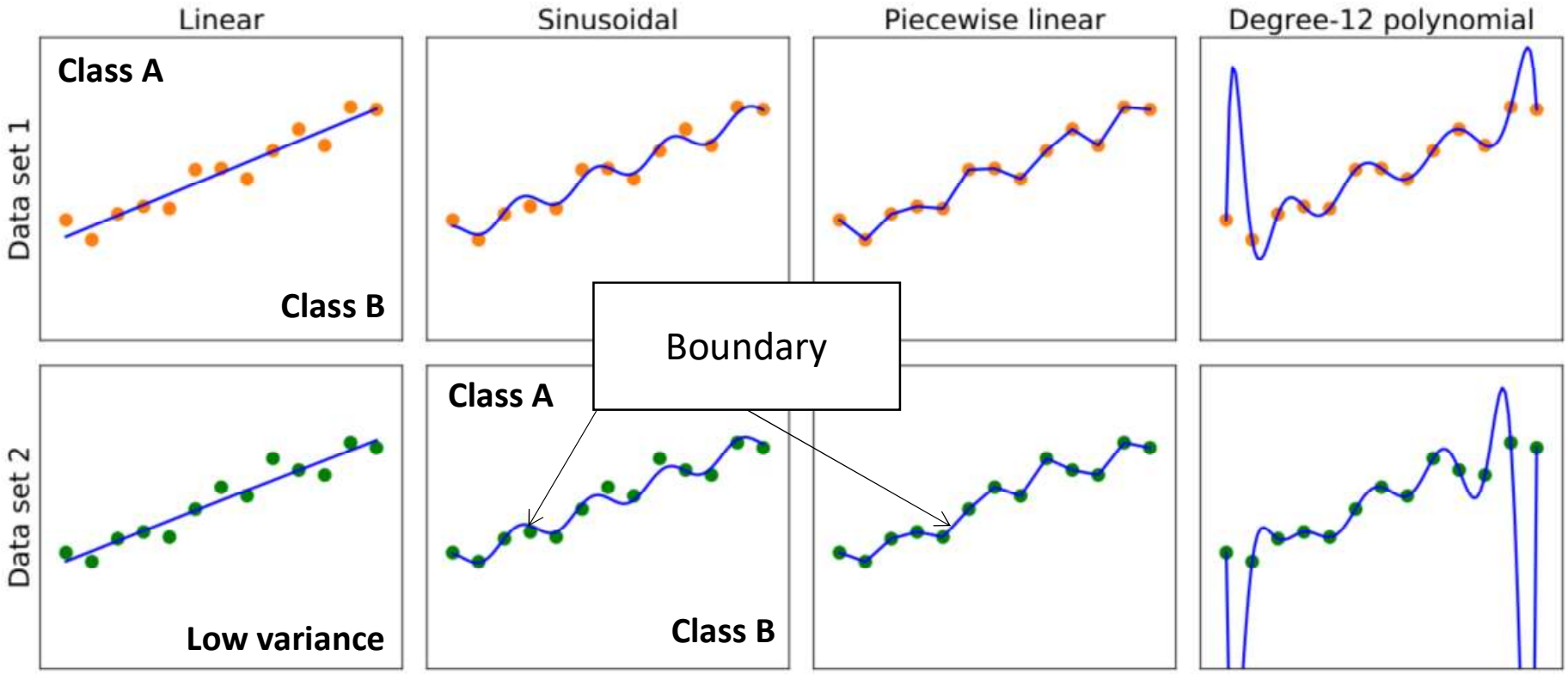


Training / Validation / Test Sets

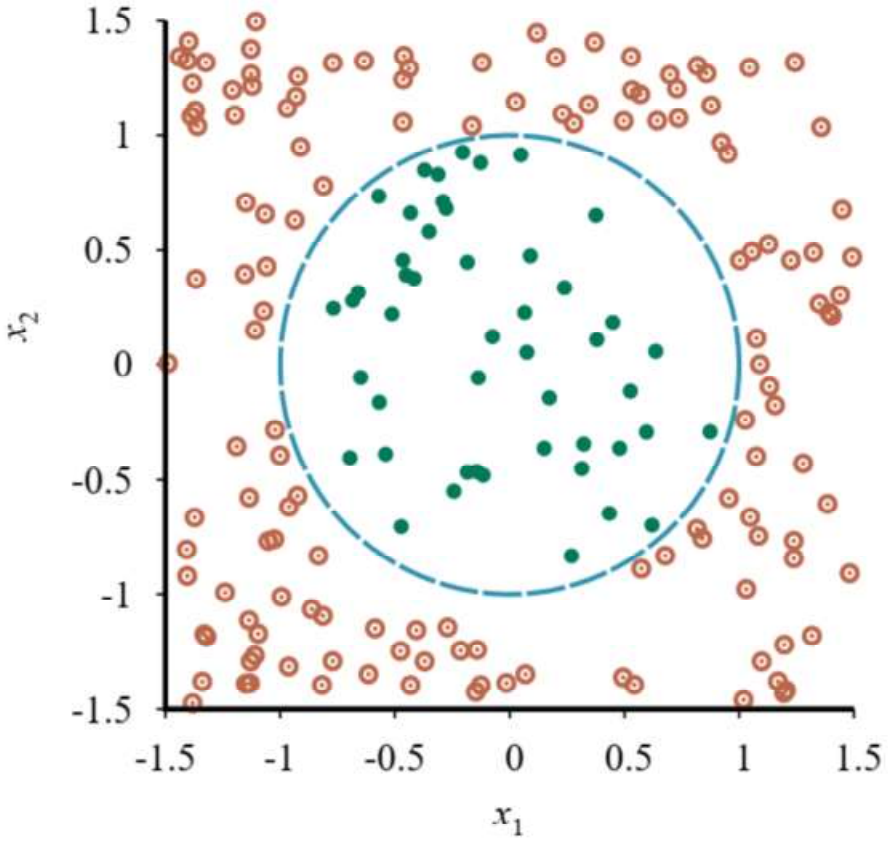


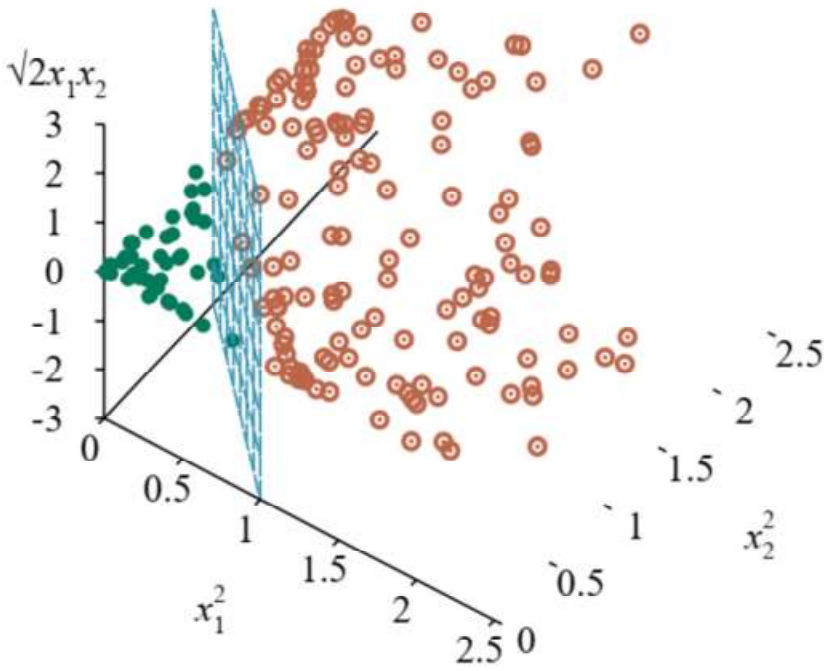


Hypothesis: Decision "Boundary"



Hypothesis: Classification "Boundary"





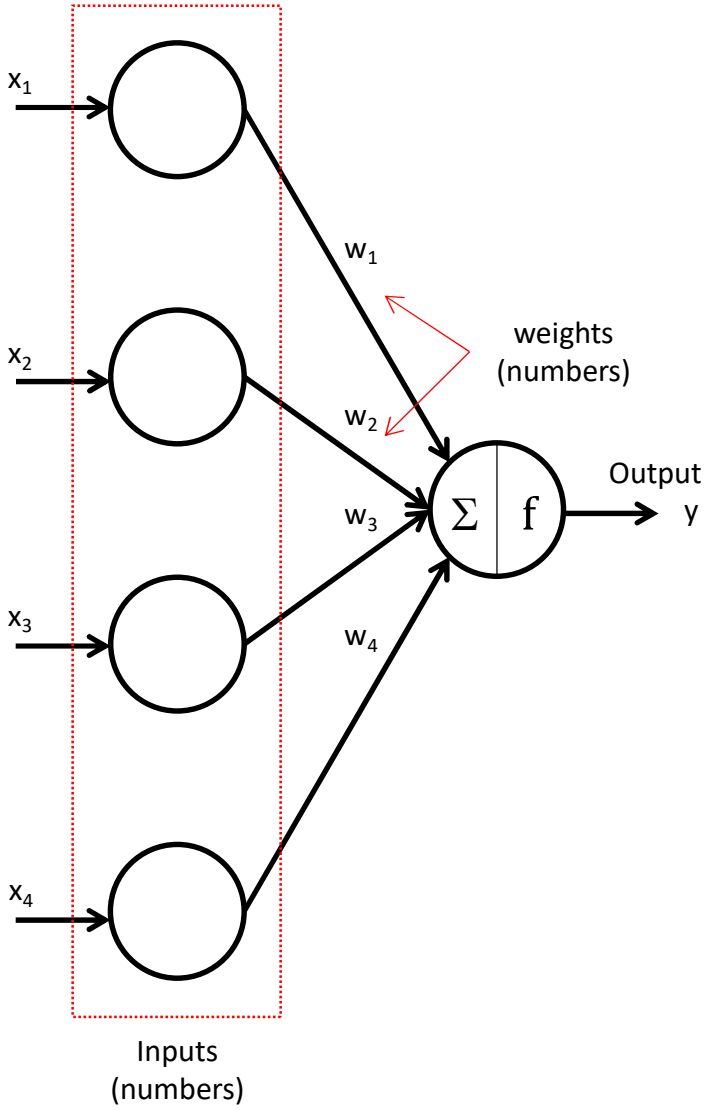
Classifier Evaluation: Confusion Matrix

		Predicted class		
		Positive	Negative	
Actual class	Positive	True Positive (TP)	False Negative (FN) Type II Error	Sensitivity $\frac{TP}{TP+FN}$
	Negative	False Positive (FP) Type I Error	True Negative (TN)	Specificity $\frac{TN}{TN+FP}$
		Precision $\frac{TP}{TP+FP}$	Negative Predictive Value $\frac{TN}{TN+FN}$	Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

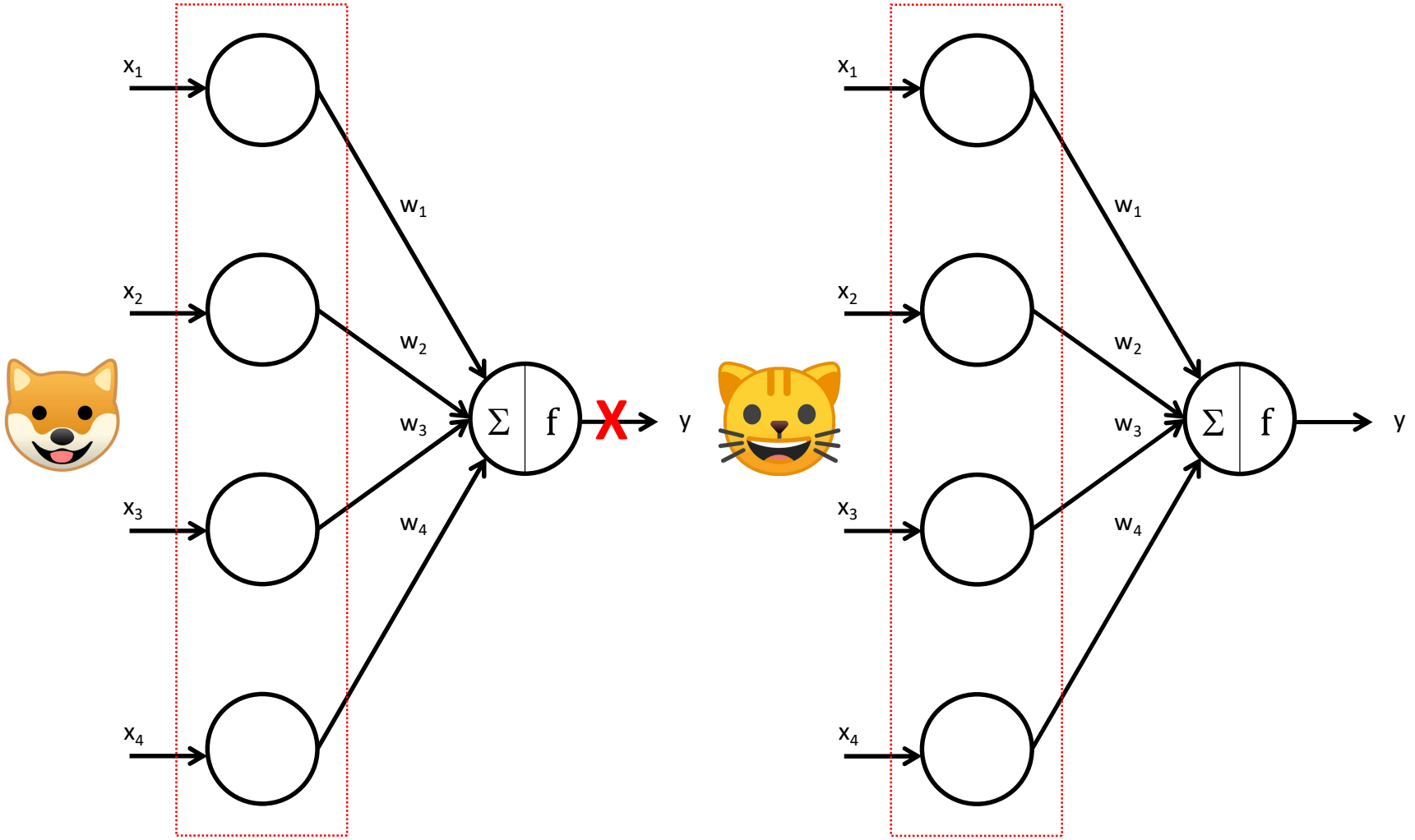
Artificial Neuron (Perceptron)

A (**single-layer**) **perceptron** is a model of a biological neuron. It is made of the following components:

- inputs x_i - numerical values representing information
- weights w_i - numerical values representing how "important" corresponding input is
- weighted sum: $\Sigma\, w_i * x_i$
- activation function f that decides if the neuron "fires"



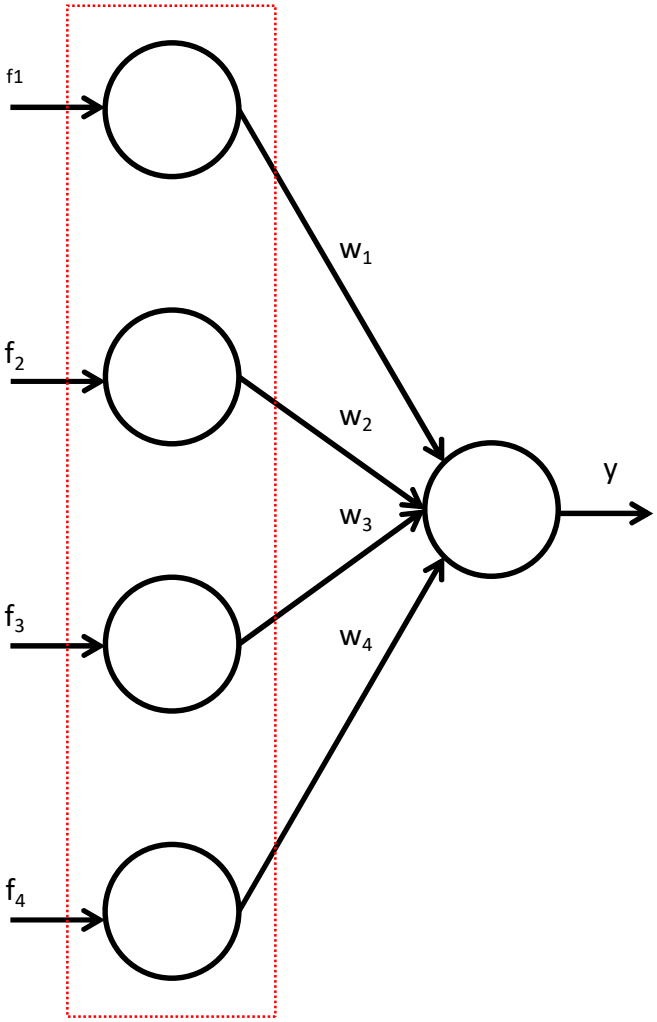
Single-layer Perceptron as a Classifier

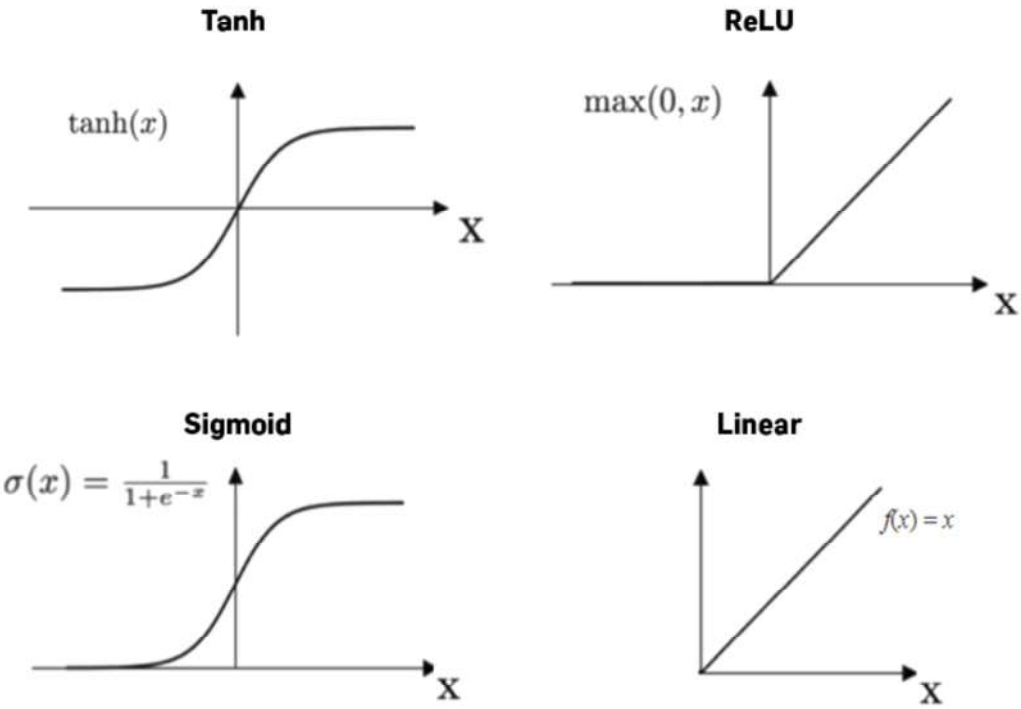


$\Sigma\, w_i * x_i < 0 \rightarrow f = 0$ → **NOT CAT**

$\Sigma\, w_i * x_i \geq 0 \rightarrow f = 1$ → **CAT**

Selected Activation Functions

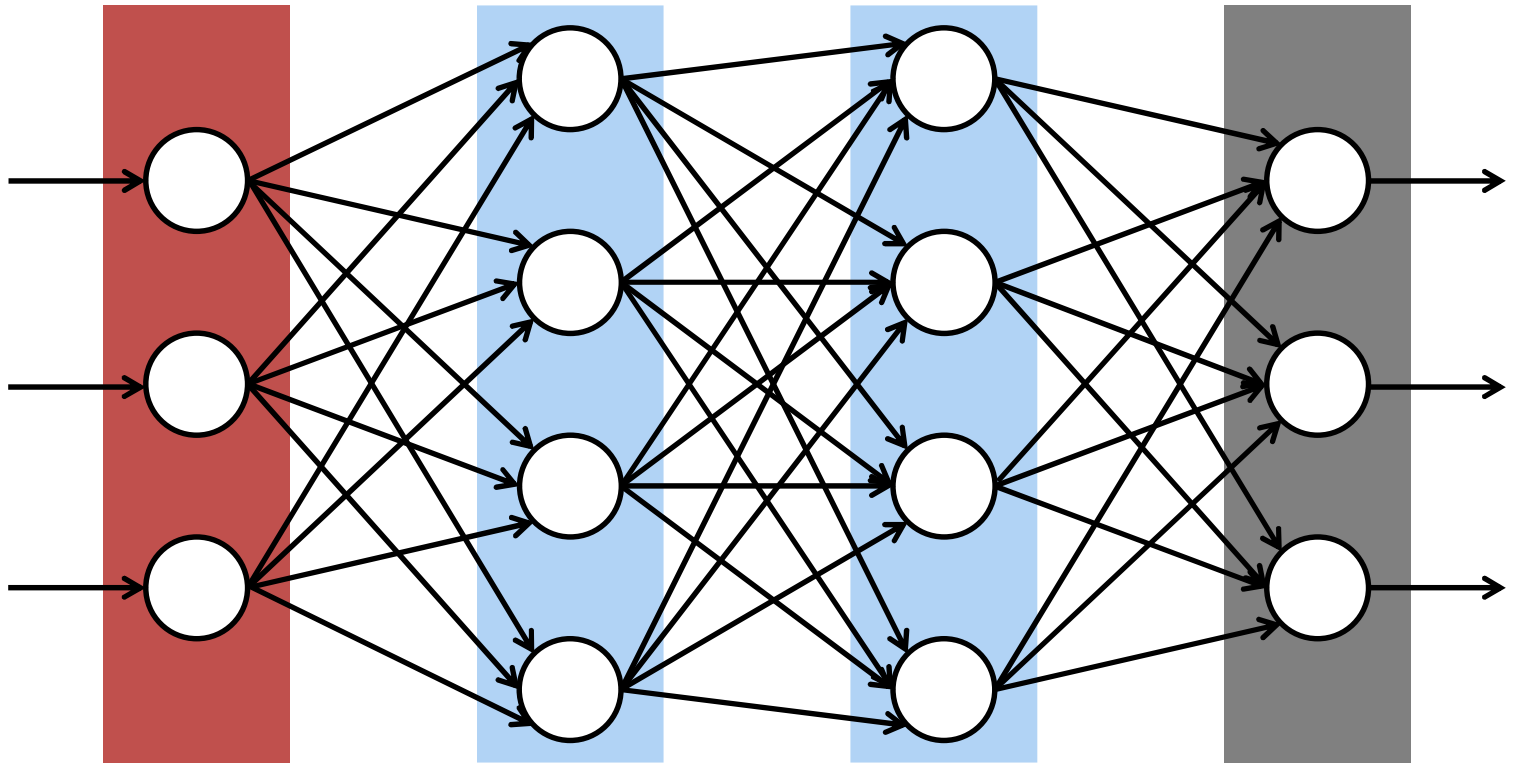




ReLU: Rectified Linear Unit

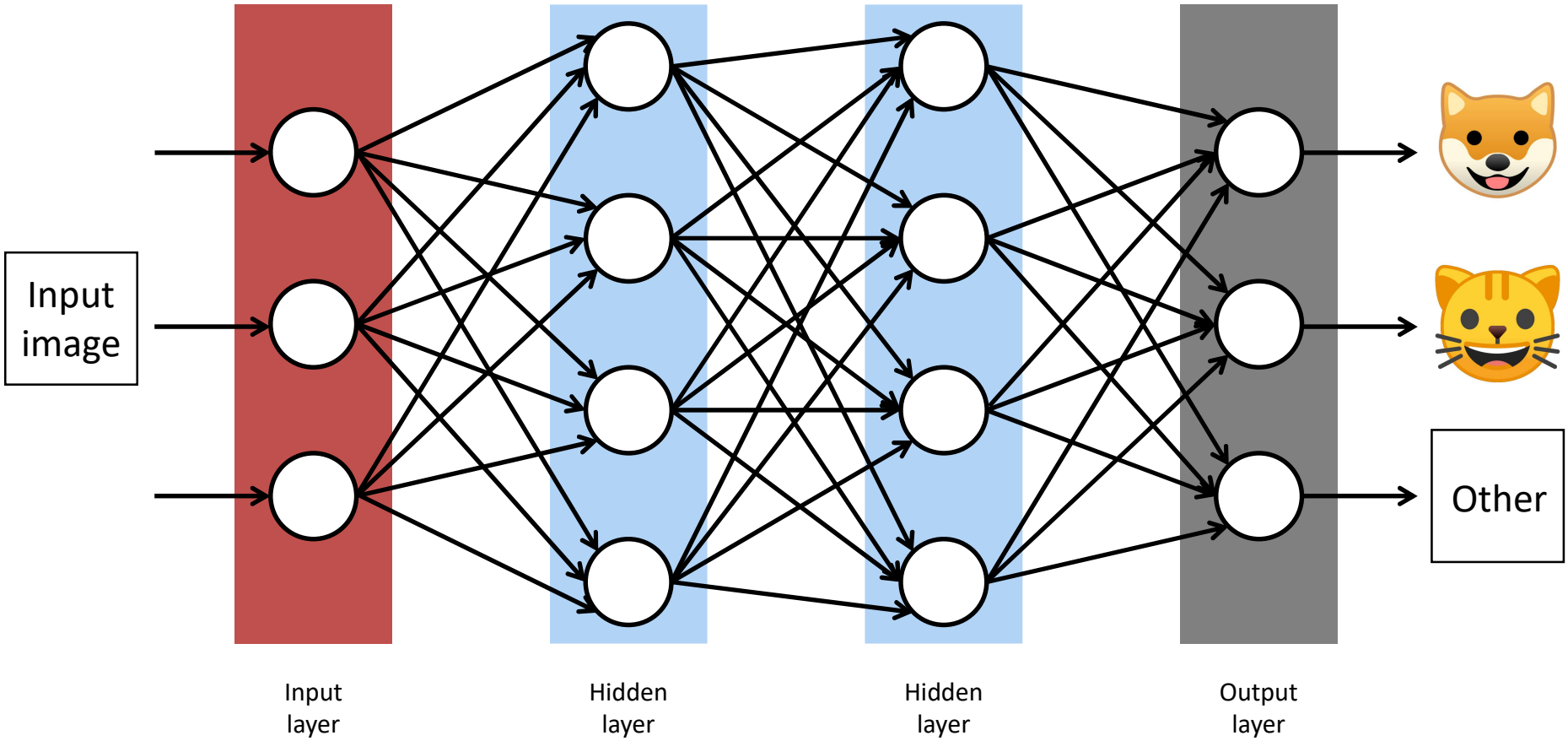
Artificial Neural Network (ANN)

An artificial neural network is made of **multiple artificial neuron layers**.

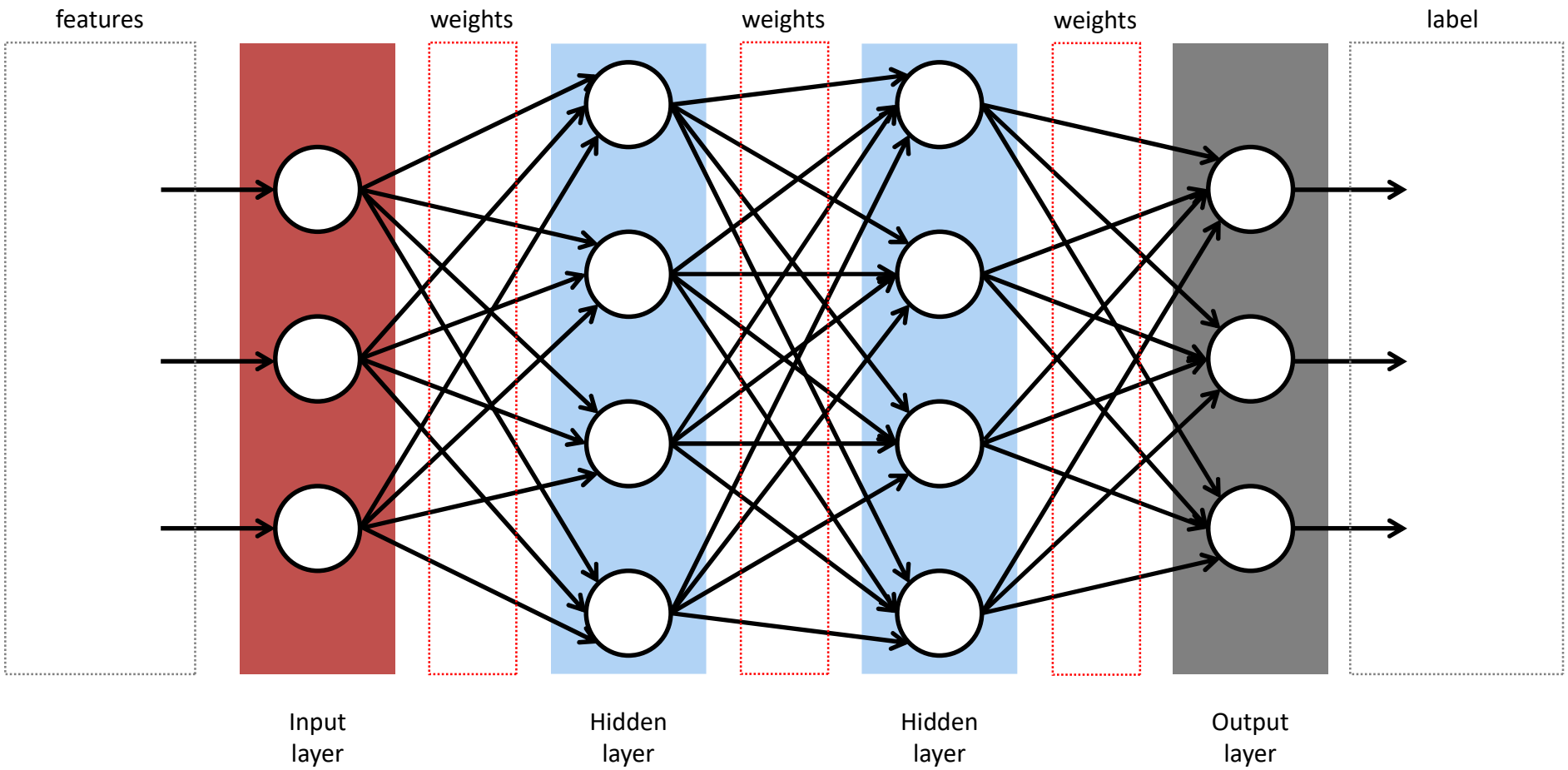


ANN as an Image Classifier

An artificial neural network can be used as a **classifier** as well.

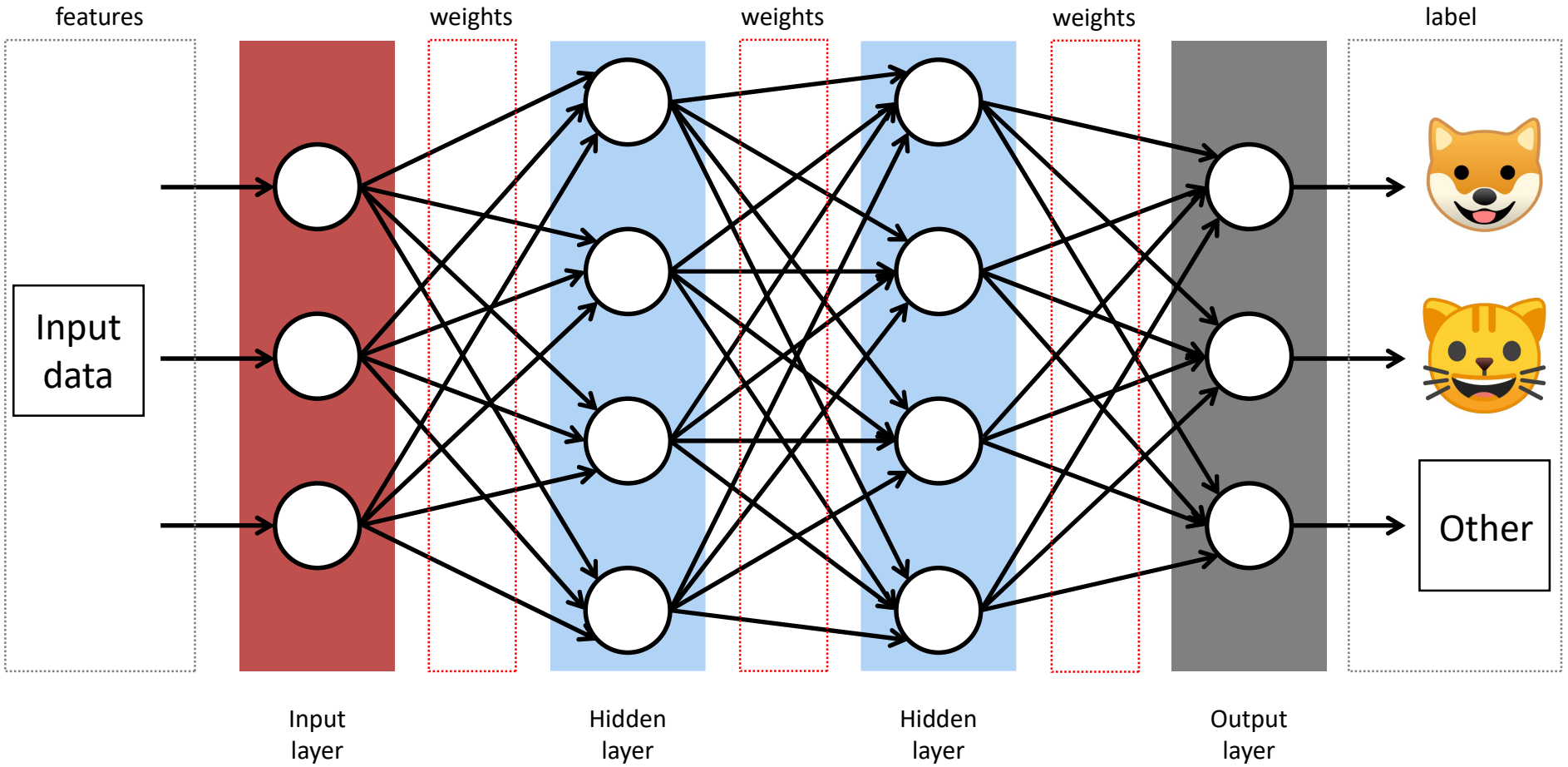


ANN as a Classifier



ANN: Supervised Learning

In order to work properly a classifier **needs to be trained** first with **labeled data**.



Training will **adjust all the weights** within this artificial neural network.

Training Data: Features + Labels

Typically input data will be represented by a **limited set of features**.

Features:
Wheels: 4
Weight: 8 tons
Passengers: 1

Label:
Truck

Features:
Wheels: 6
Weight: 8 tons
Passengers: 1

Label:
Truck

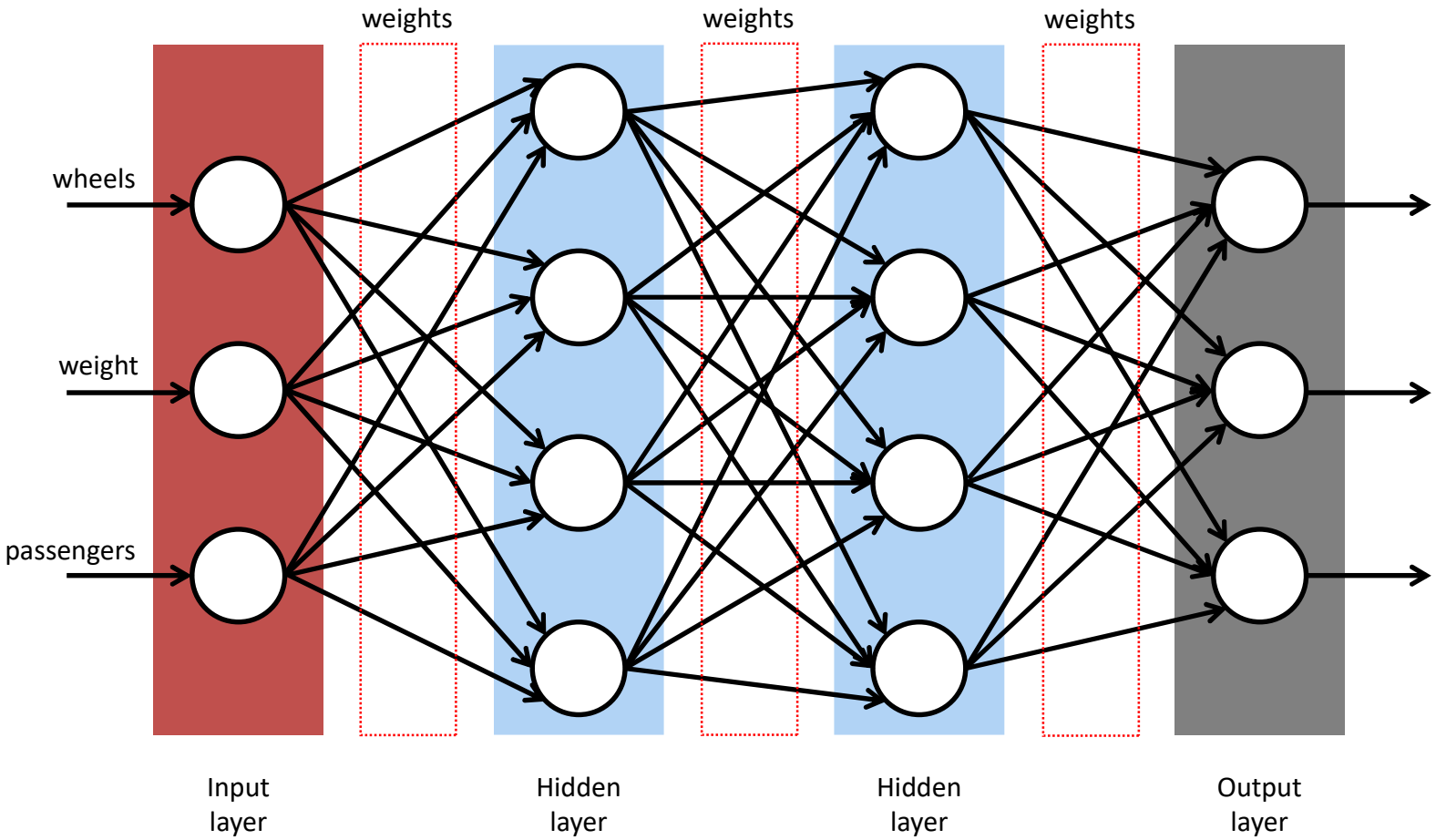
Features:
Wheels: 4
Weight: 1 ton
Passengers: 4

Label:
Car

Features:
Wheels: 4
Weight: 2 tons
Passengers: 4

Label:
Car

ANN: Supervised Learning



Training Data: Images + Labels

A classifier **needs to be "shown" thousands of labeled examples to learn**.

Label: BUS

Label: CAR

Label: BRIDGE

Label: PALM

Label: TRAFFIC LIGHT

Label: TAXI

Label: CROSSWALK

Label: CHIMNEY

Label: MOTORCYCLE

Label: STREET SIGN

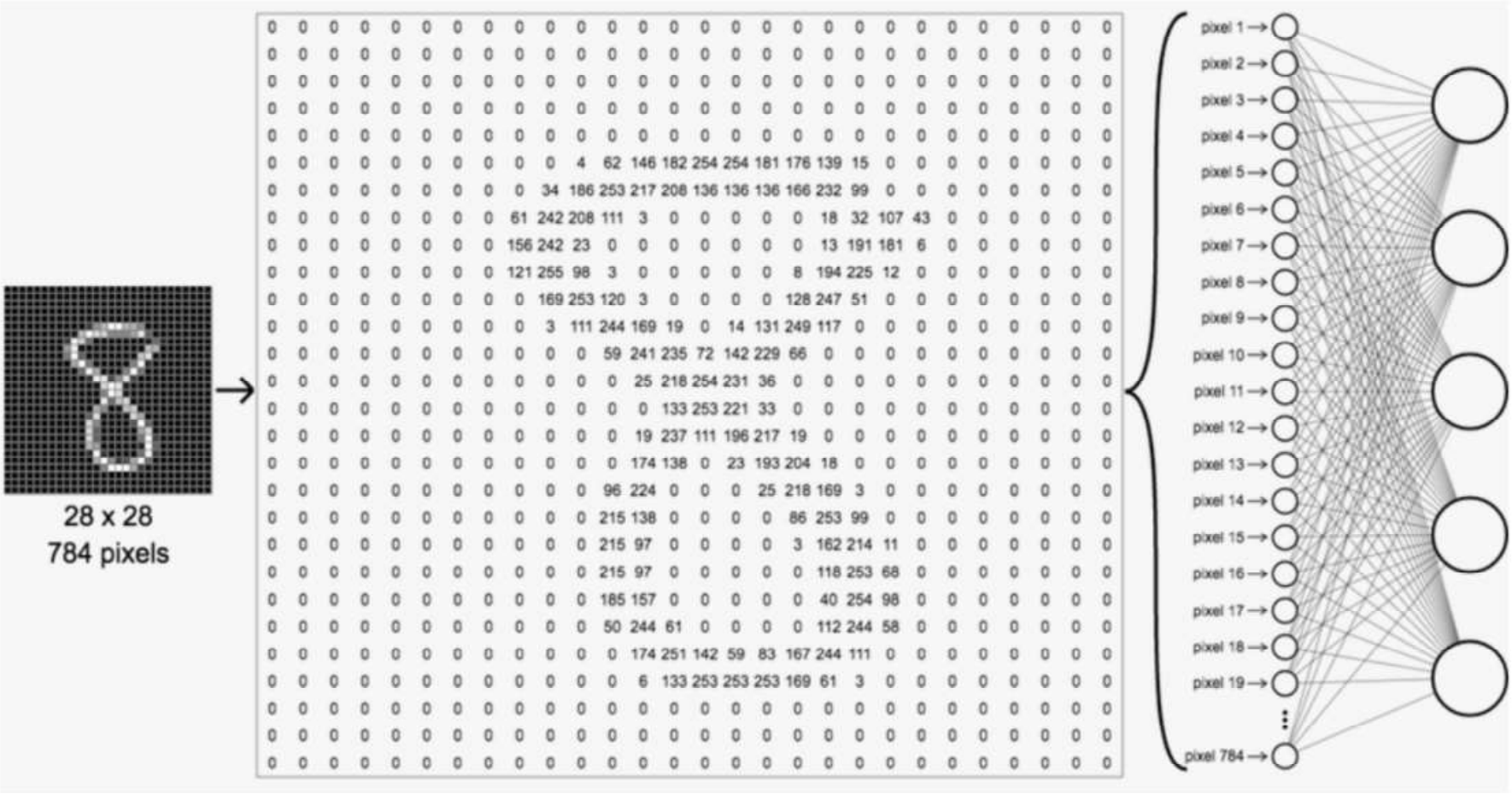
Label: HYDRANT

Label: BICYCLE

Note how some images are "incomplete" and "flawed".

Digit Image as ANN Feature Set

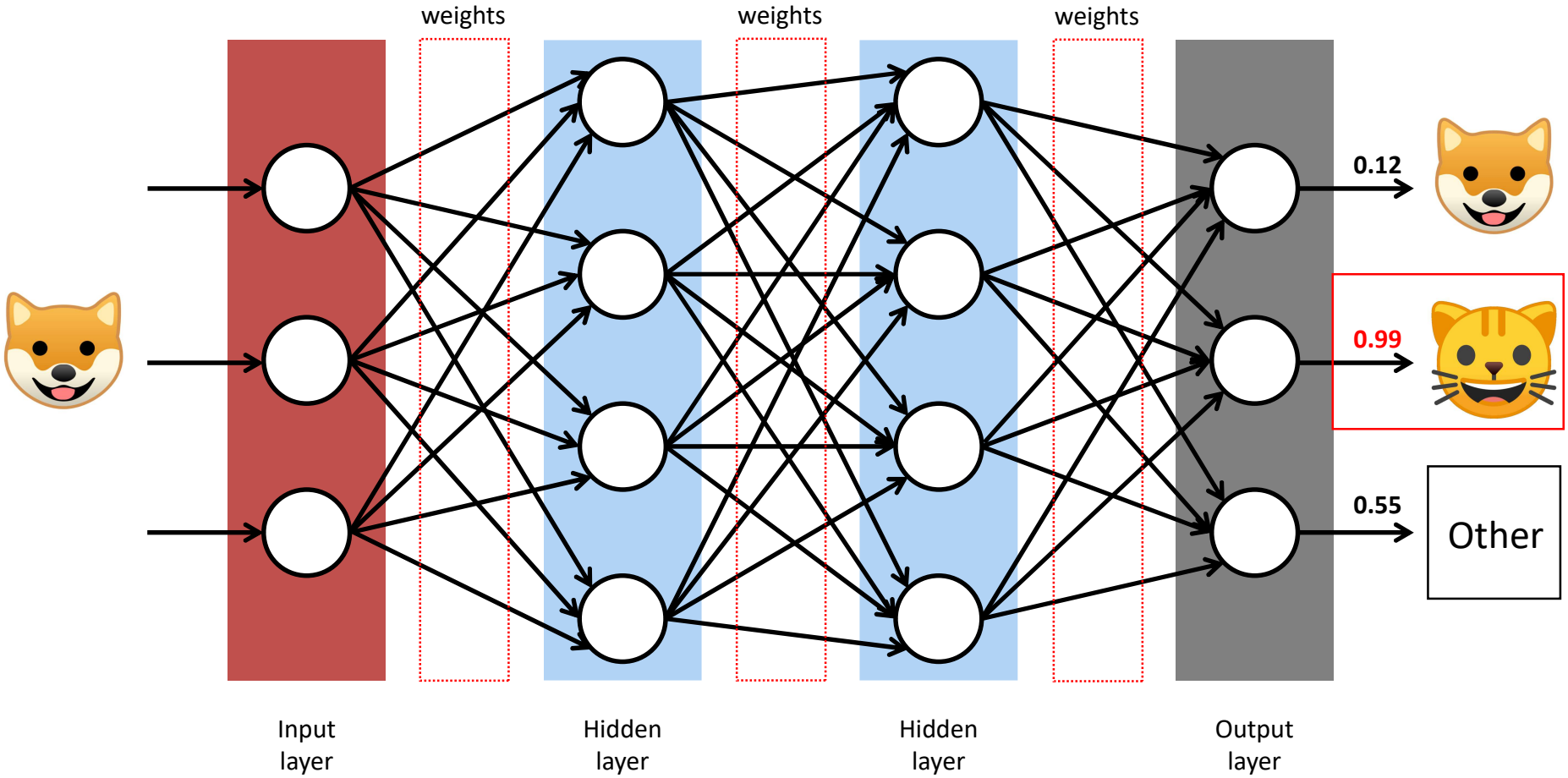
Individual features need to be “extracted” from an image. An image is numbers.



Source: https://nikolanews.com/not-just-introduction-to-convolutional-neural-networks-part-1/

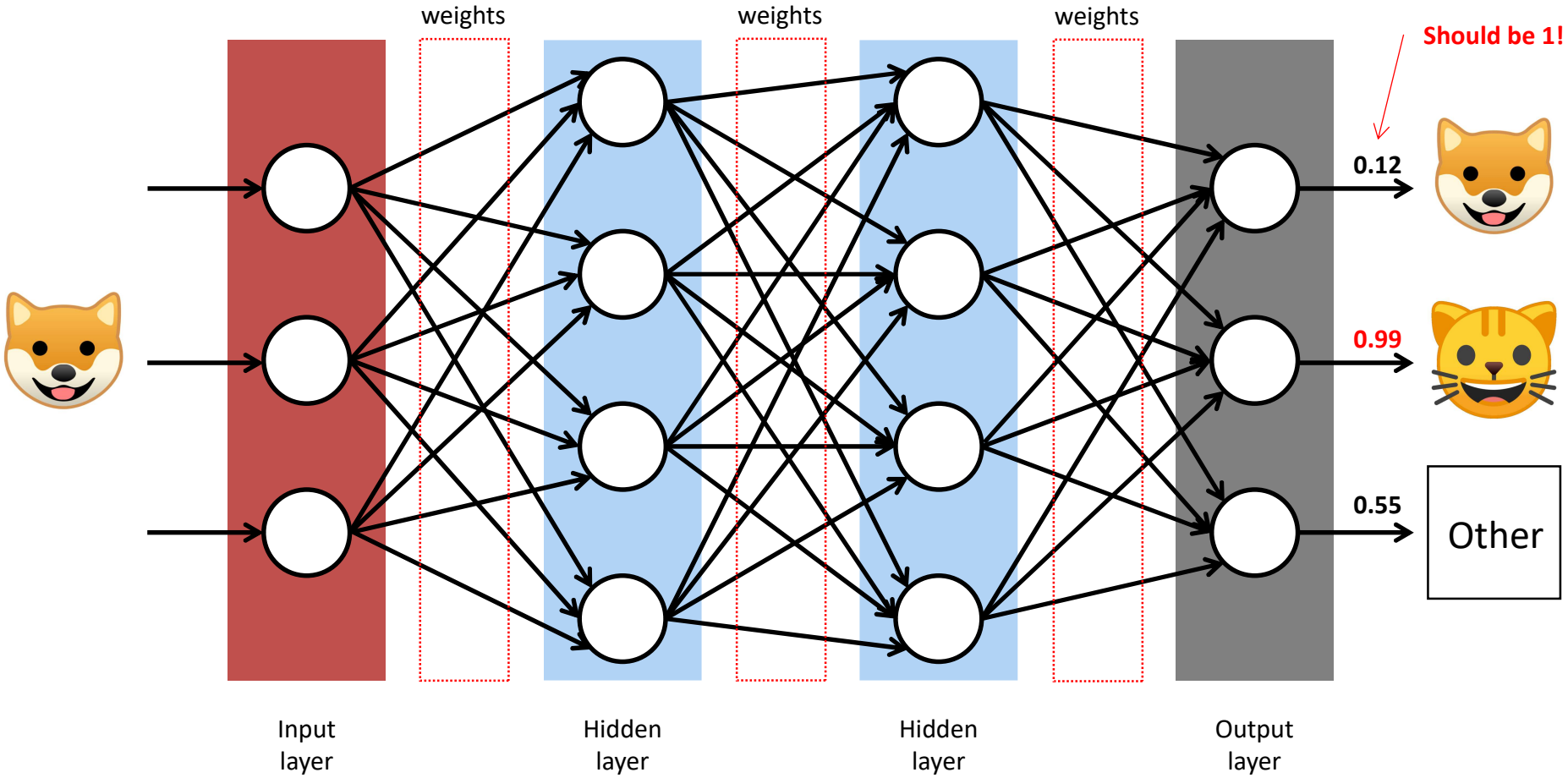
ANN: Supervised Learning

An **untrained classifier** will **NOT label input data correctly**.



ANN: Training

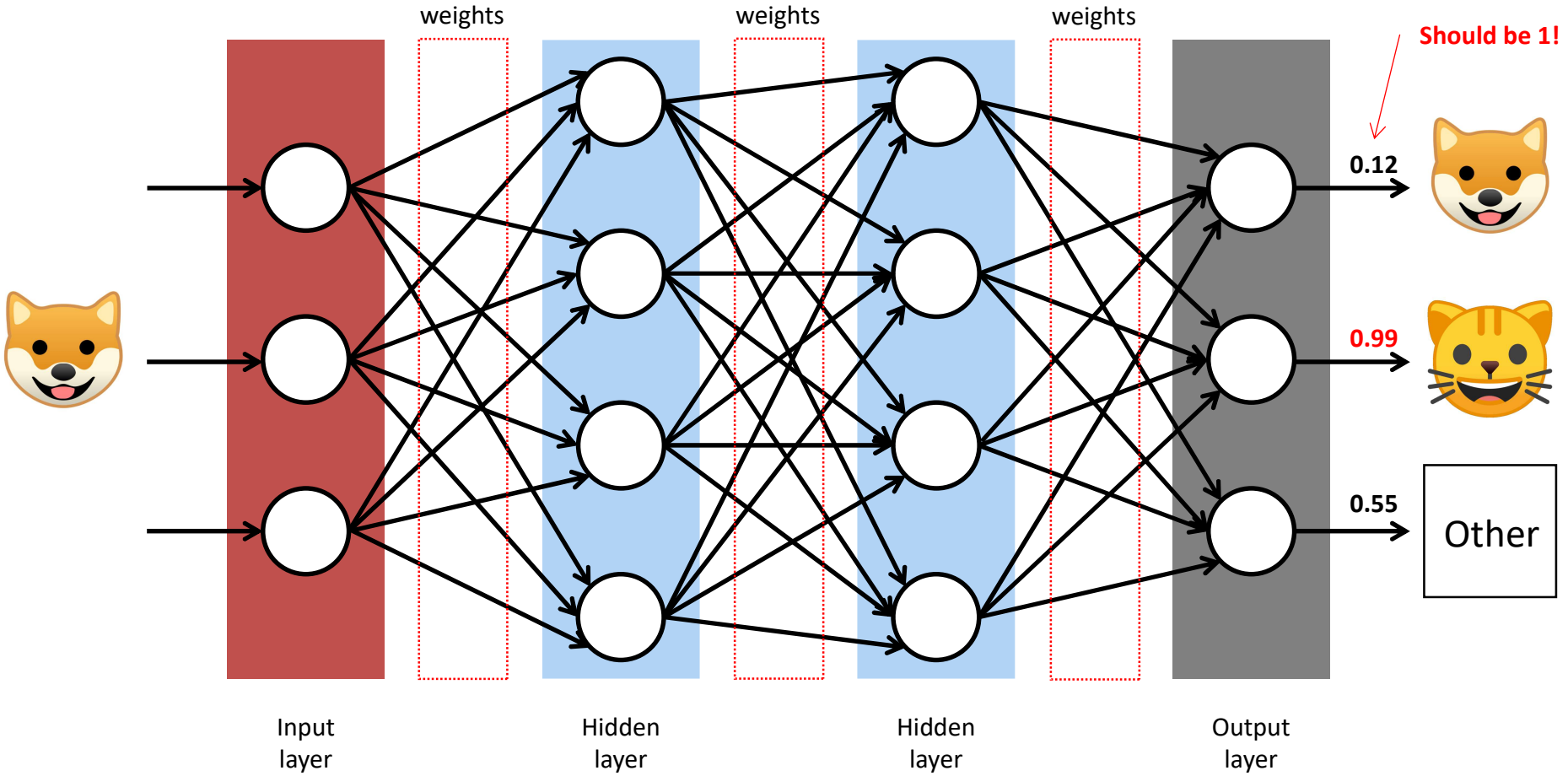
Given: input data  and it's corresponding **expected** label: DOG calculate "error".



"Error" = 0.88. Go back and **adjust all the weights** to ensure it is lower next time.

ANN: Training

Show data / label pair:  / DOG.



Correct all the weights. Repeat many times.

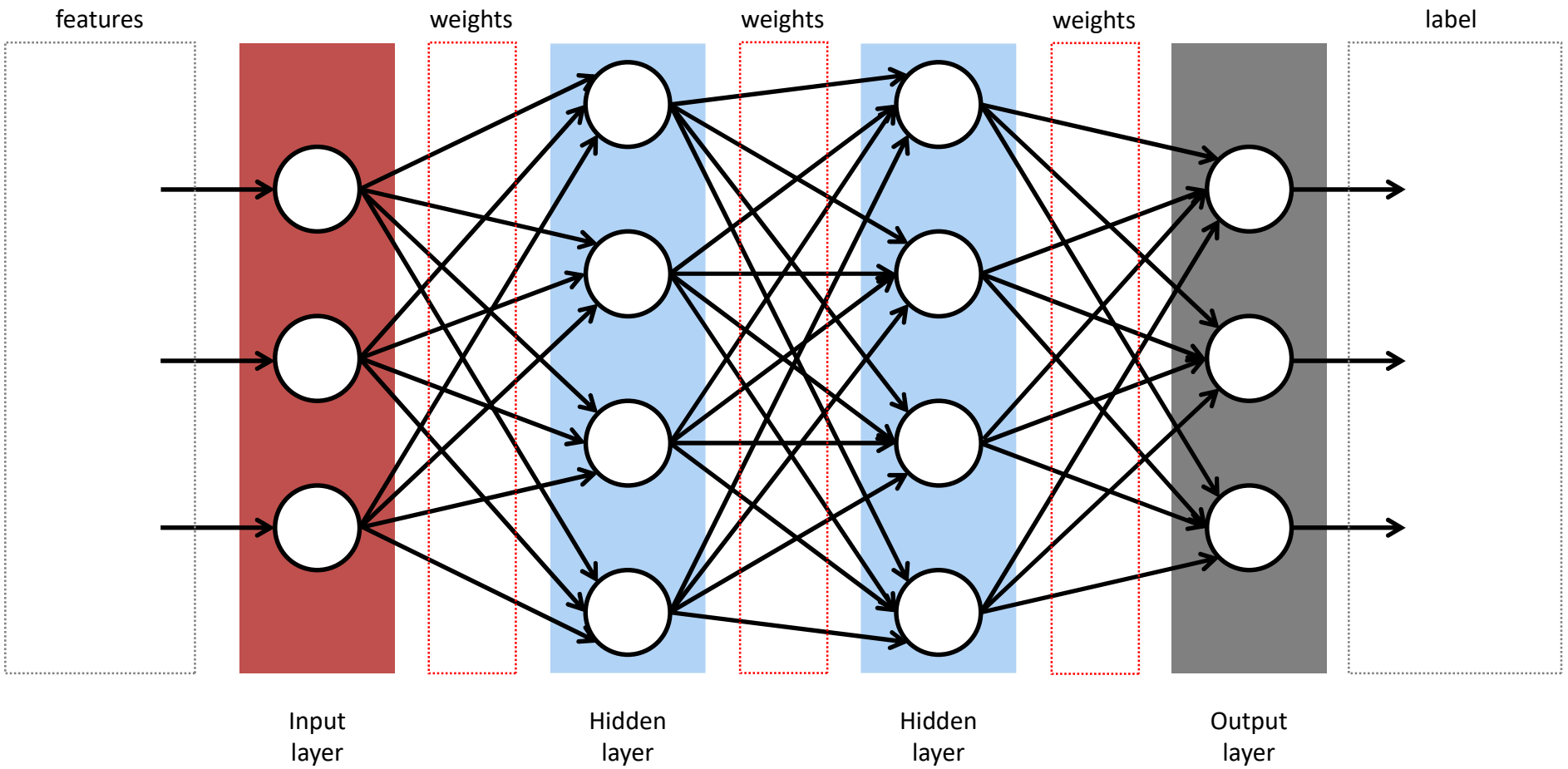
Exercise: ANN Demo

http://playground.tensorflow.org/

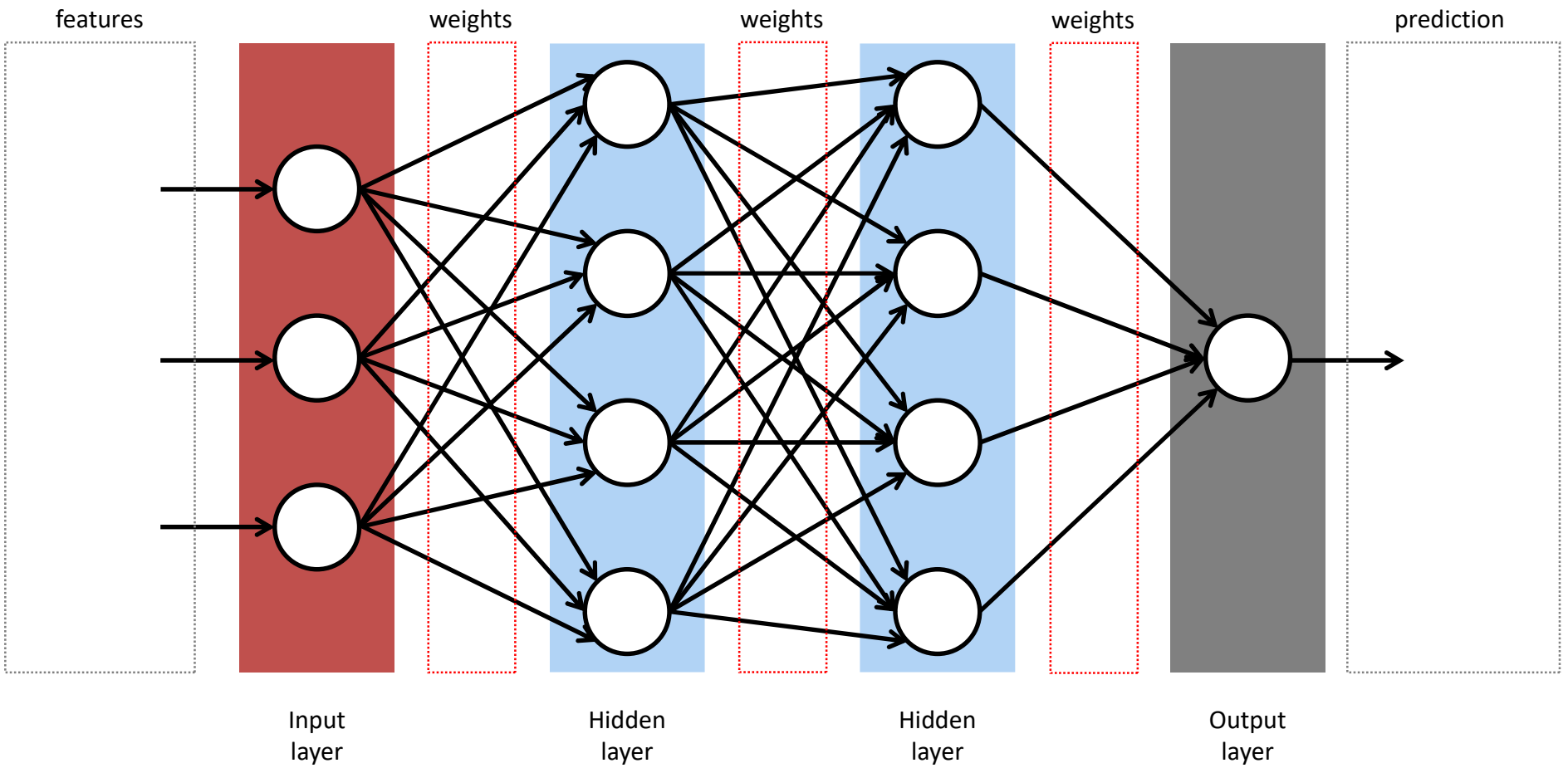
Exercise: Train a Classifier!

https://teachablemachine.withgoogle.com/

ANN for Classification

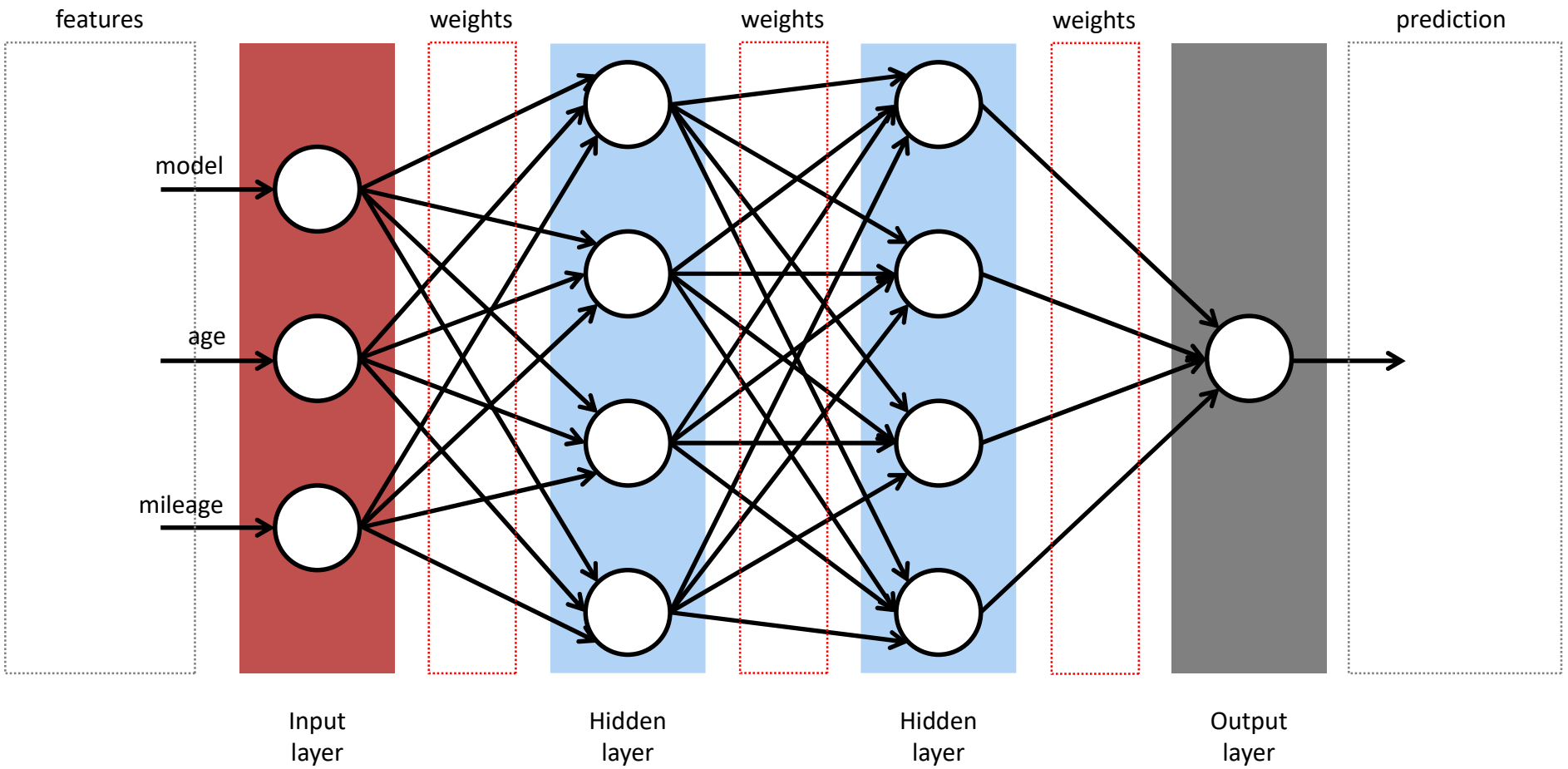


ANN for Regression



ANN for Regression: Used Car Price

Used car price predictor: train it first with used **car data - price** pairs.



Unsupervised Learning

What is Unsupervised Learning?

Idea:

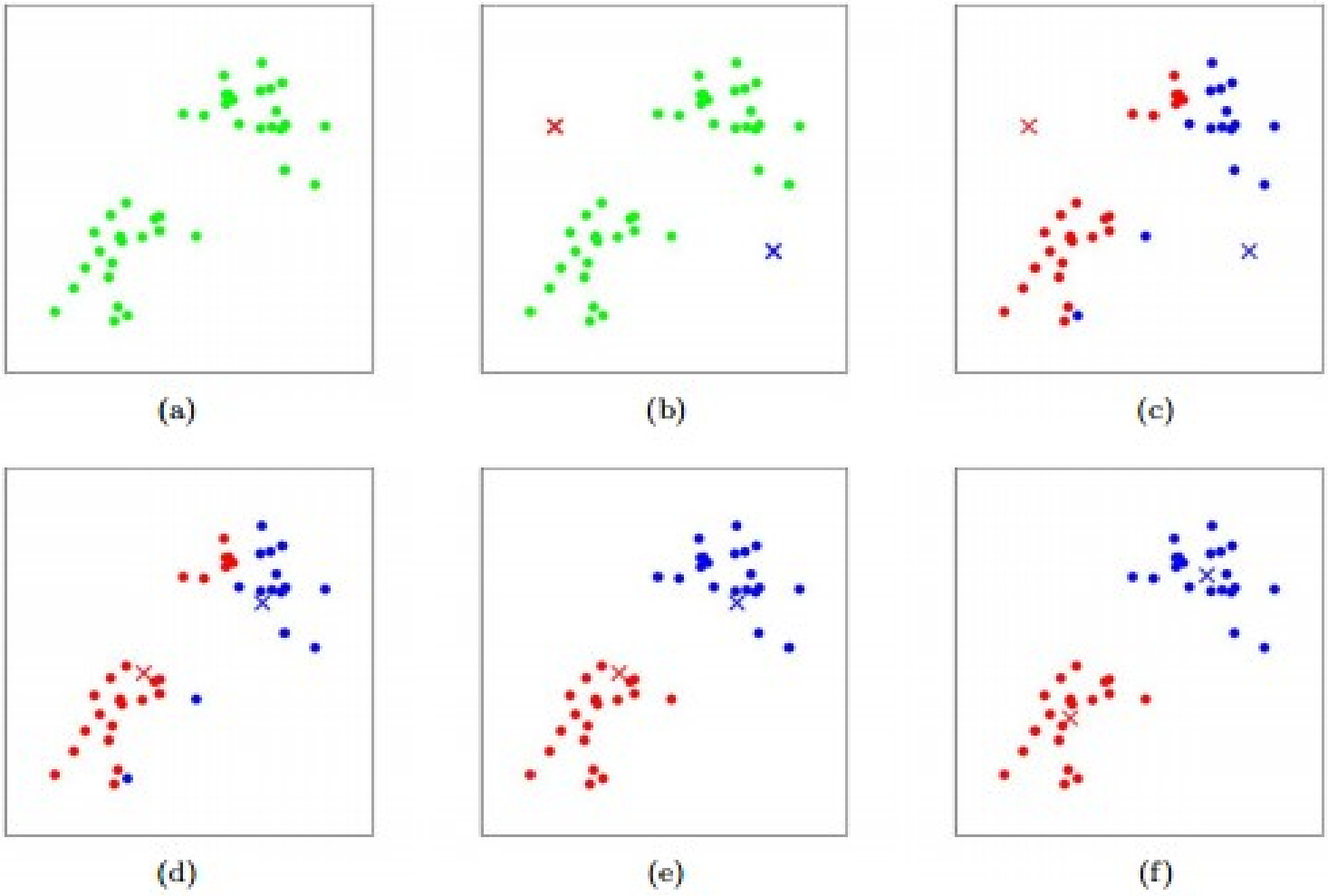
Unsupervised learning involves finding underlying patterns within data. Typically used in **clustering** data points (similar customers, etc.).

In other words:

- there is some structure (groups / clusters) in data (for example: customer information)
- we don't know what it is (**= no labels!**)
- unsupervised learning tries to discover it

Unsupervised Learning: K-Means Clustering

K-Means Clustering: The Idea

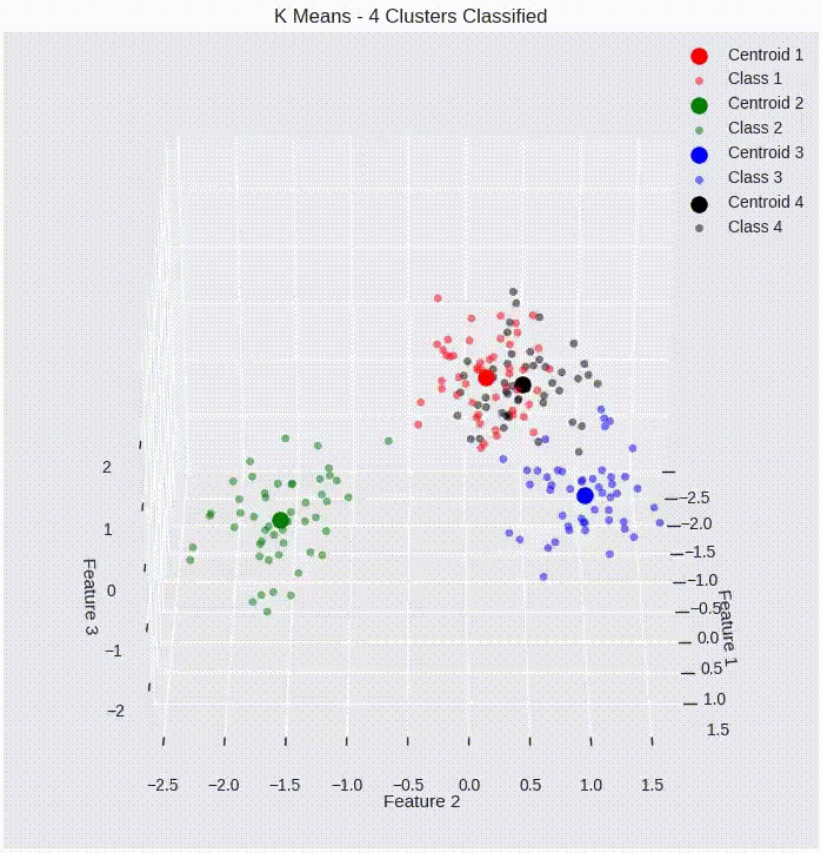


Source: https://stanford.edu/~cpiech/cs221/handouts/kmeans.html

Exercise: K-Means Clustering

https://lalejini.com/my_empirical_examples/KMeansClusteringExample/web/kmeans_clustering.html

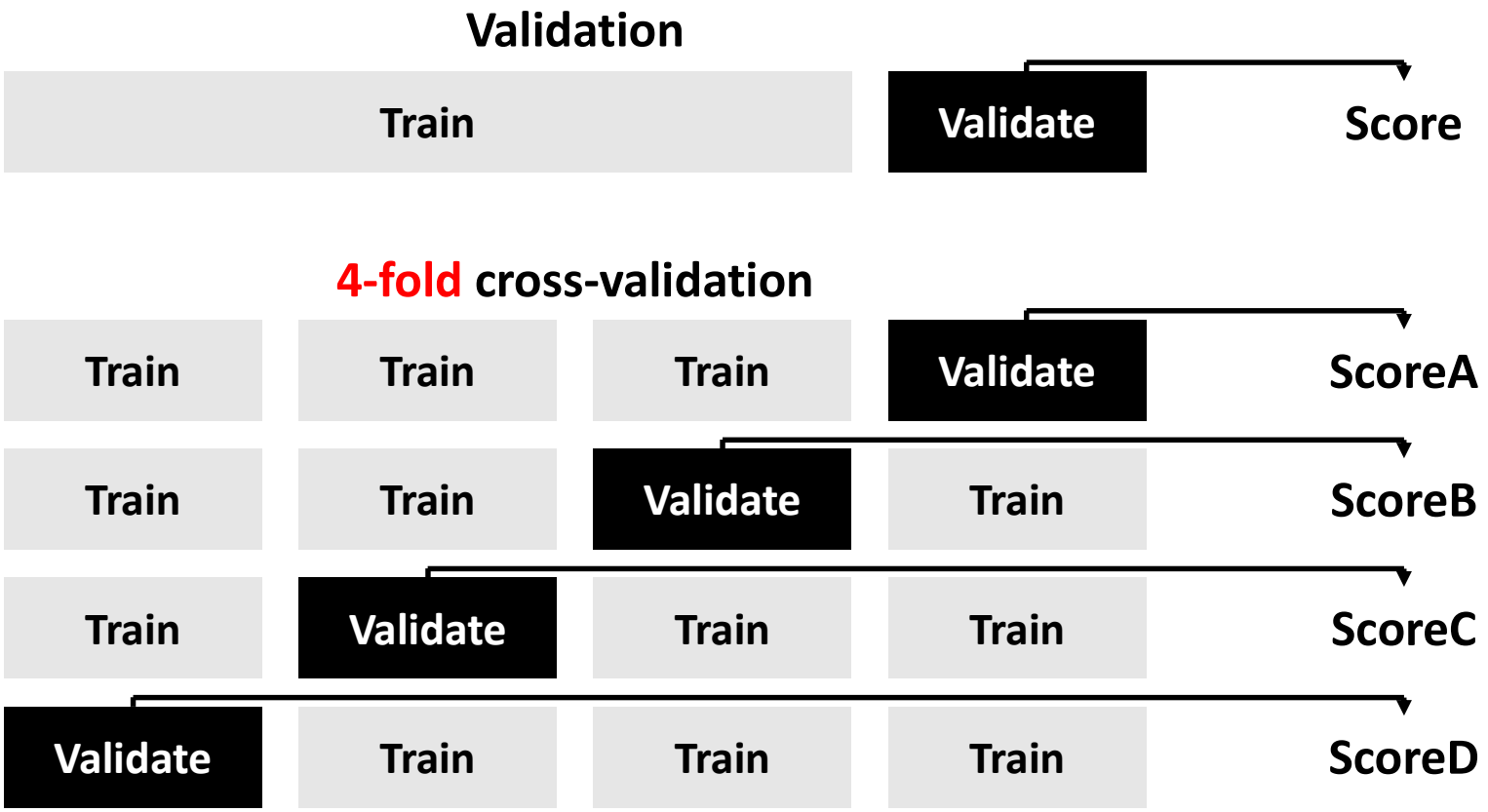
3D K-Means Clustering Visualized



Source: https://github.com/Gautam-J/Machine-Learning

Where Would You Use Clustering?

K-Fold Cross-Validation



$$\text{Score} = \frac{\text{ScoreA} + \text{ScoreB} + \text{ScoreC} + \text{ScoreD}}{4}$$

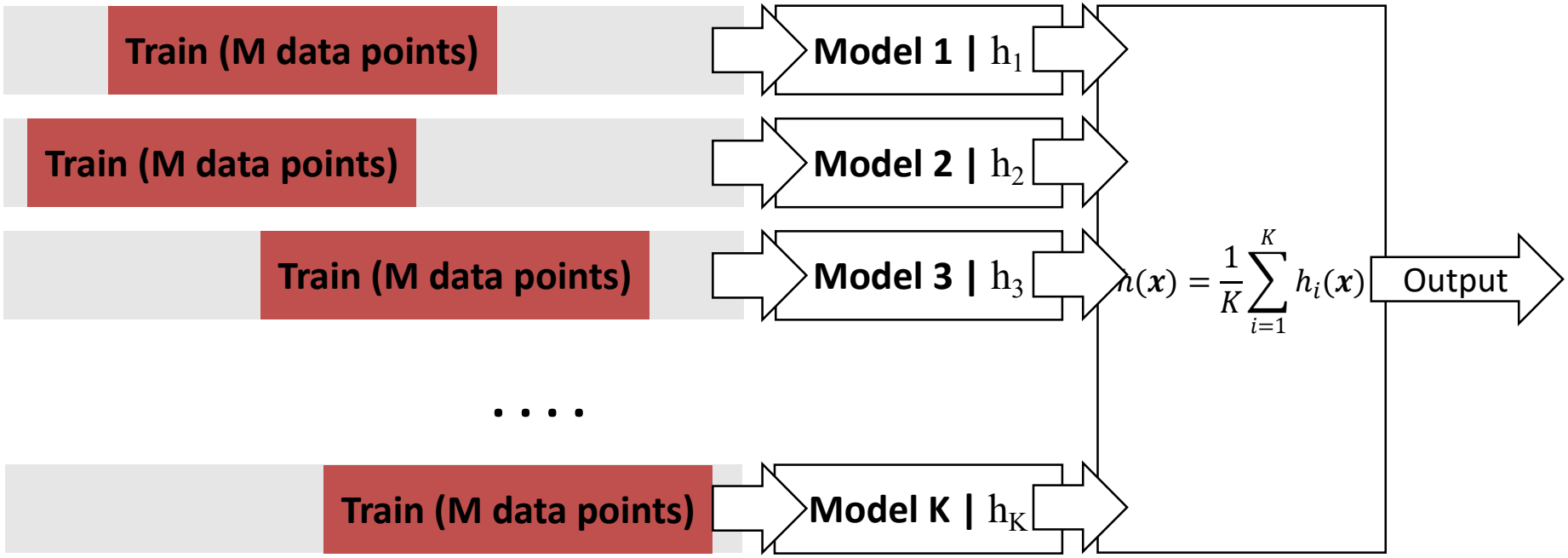
Ensemble Learning

In ensemble learning we are creating a collection (an ensemble) of hypotheses (models) h_1, h_2, ..., h_N and combine their predictions by averaging, voting, or another level of machine learning. Indvidual hypotheses (models) are based models and their combination is the ensemble model.

- **Bagging**
- **Boosting**
- **Random Trees**
- **etc.**

Bagging: Regression

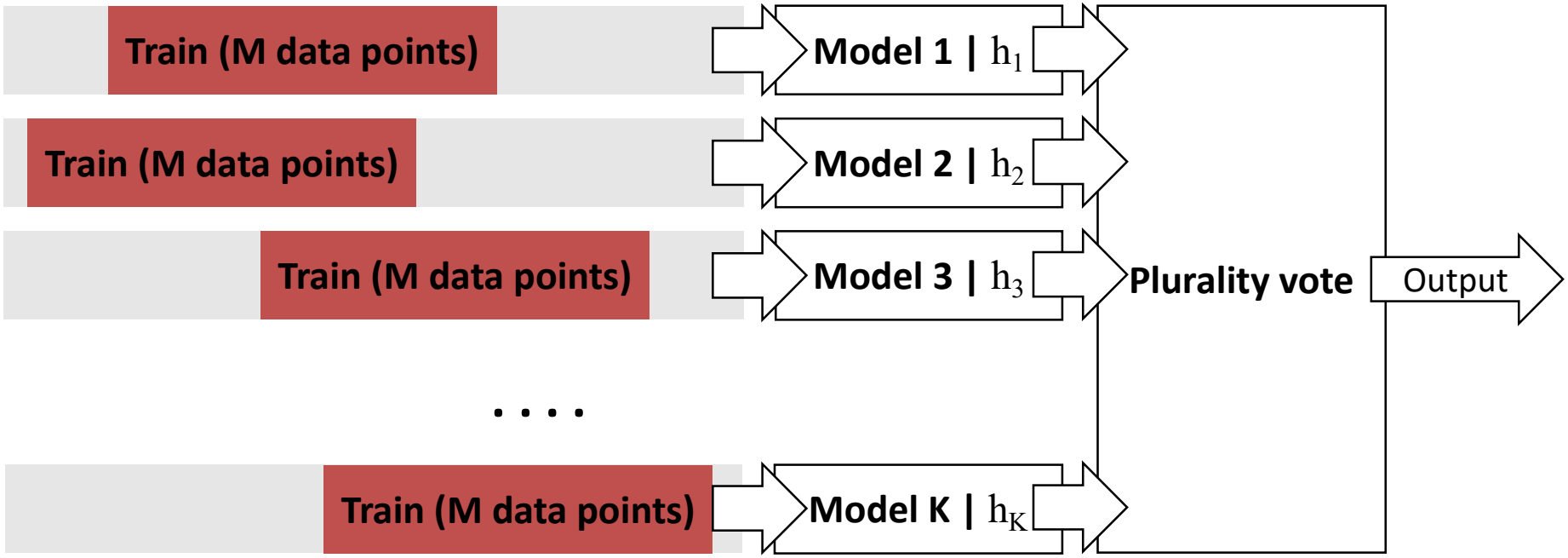
In bagging we generate K training sets by sampling with replacement from the original training set.



Bagging tends to reduce variance and helps with smaller data sets.

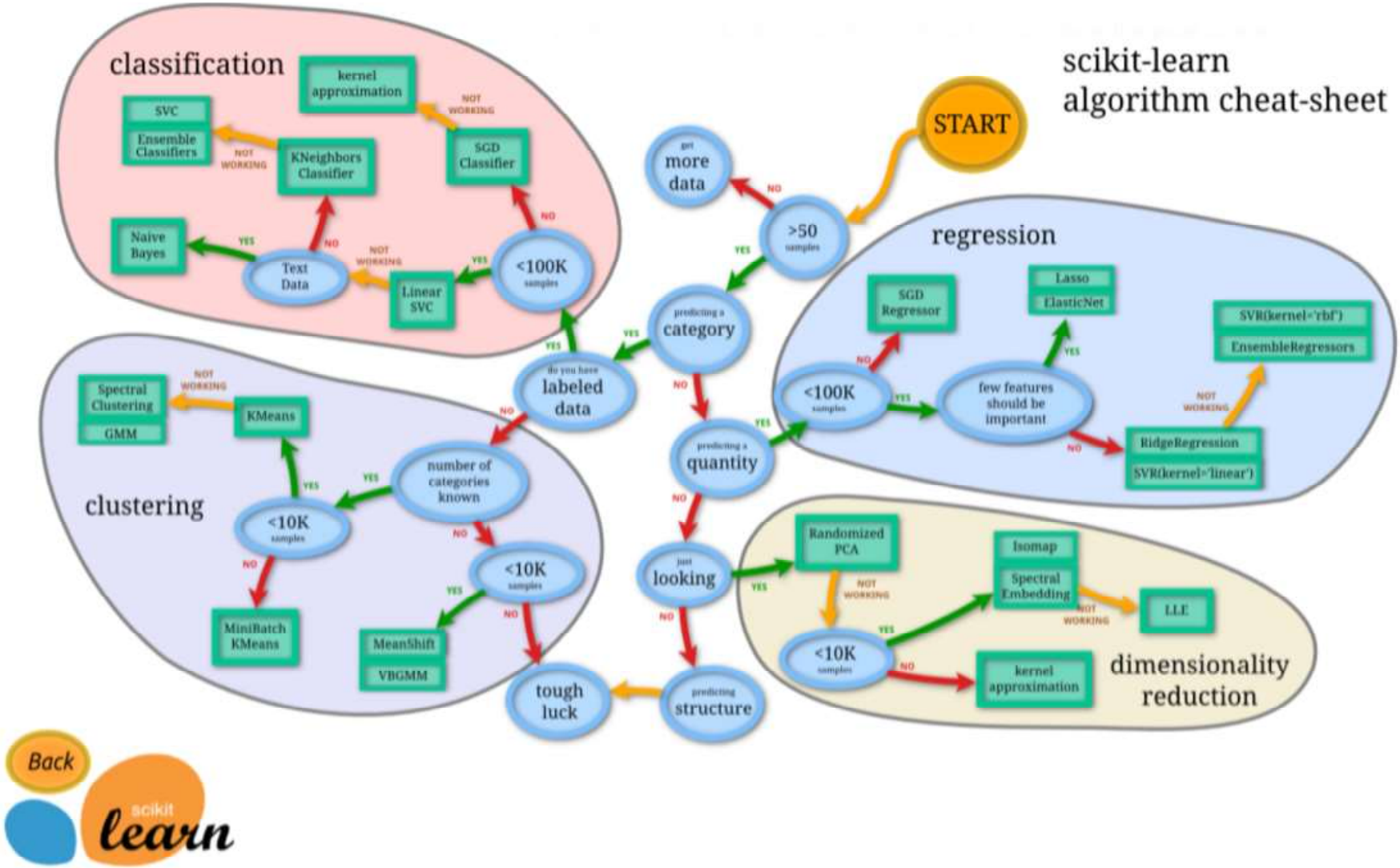
Bagging: Classification

In bagging we generate K training sets by sampling with replacement from the original training set.



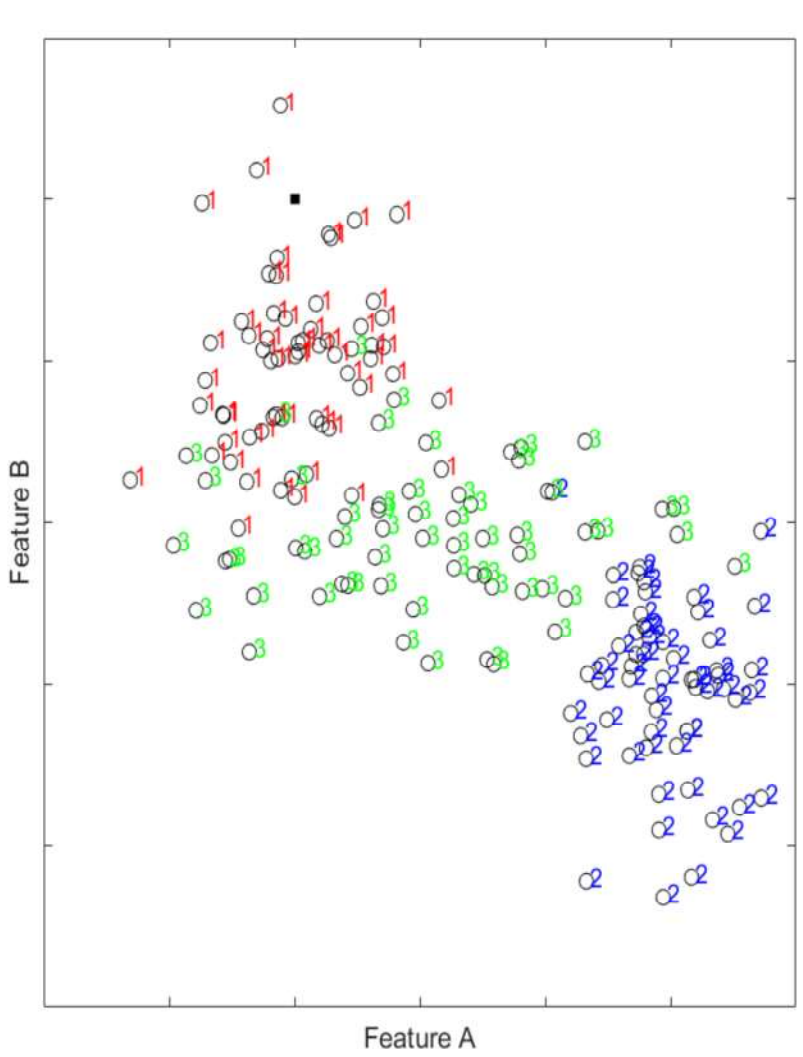
Bagging tends to reduce variance and helps with smaller data sets.

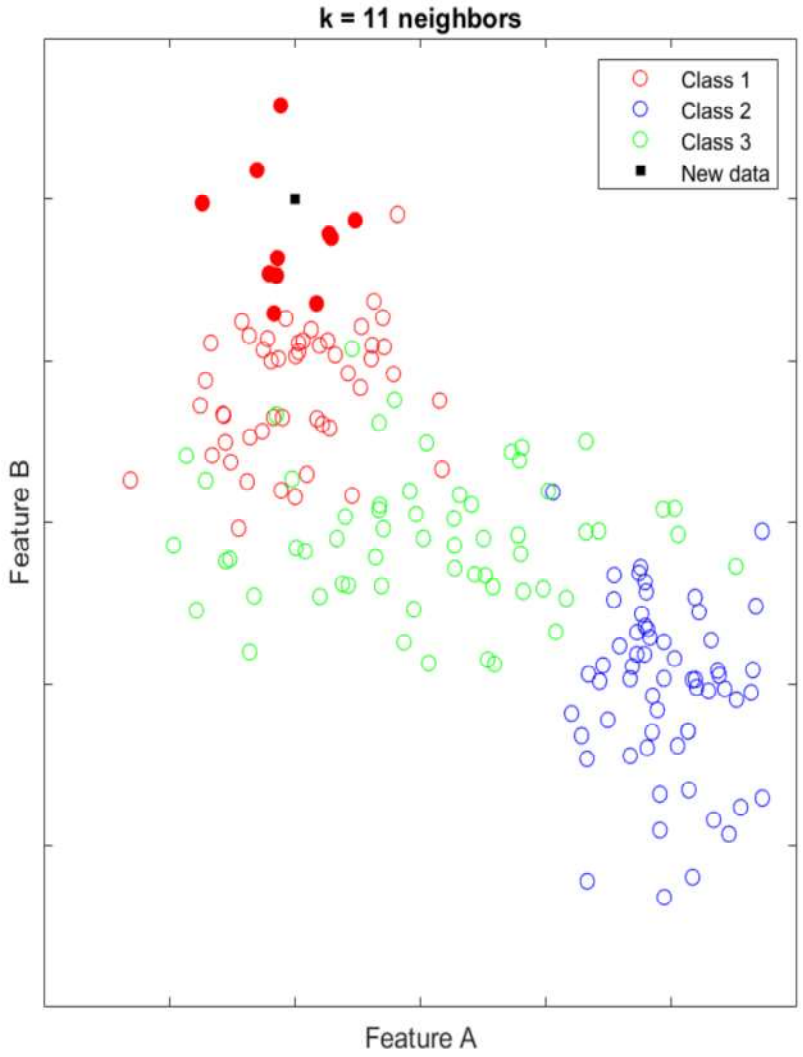
scikit-learn Algorithm Cheat Sheet



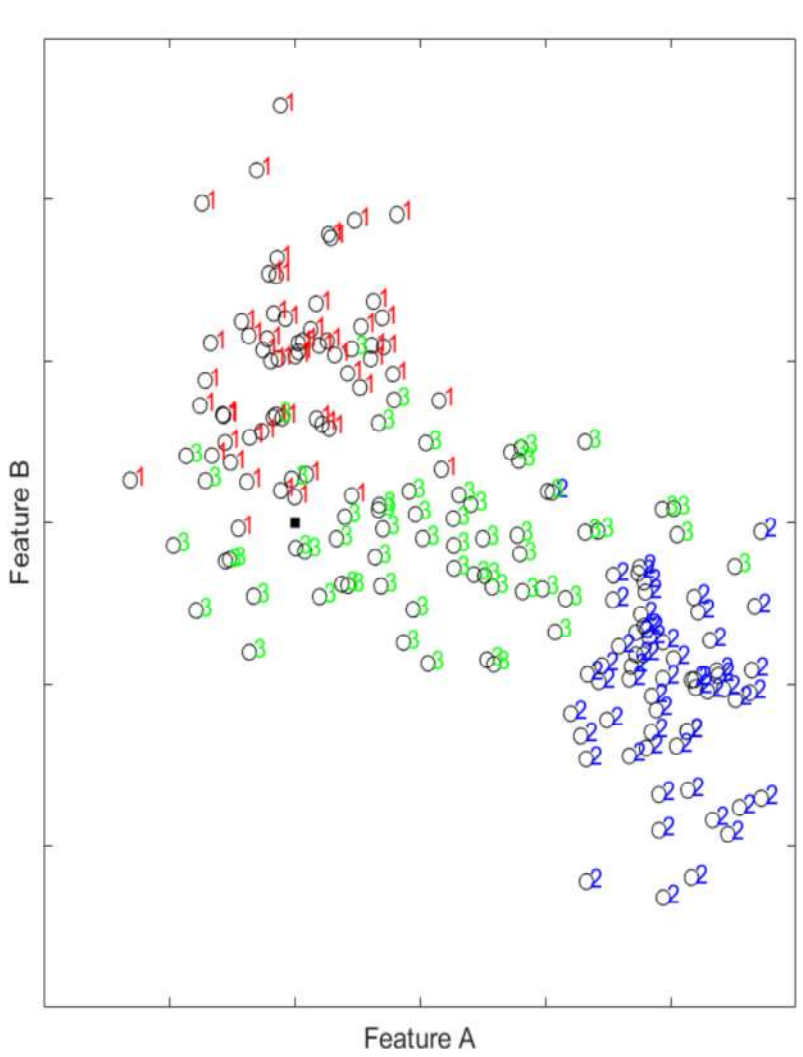
Source: https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

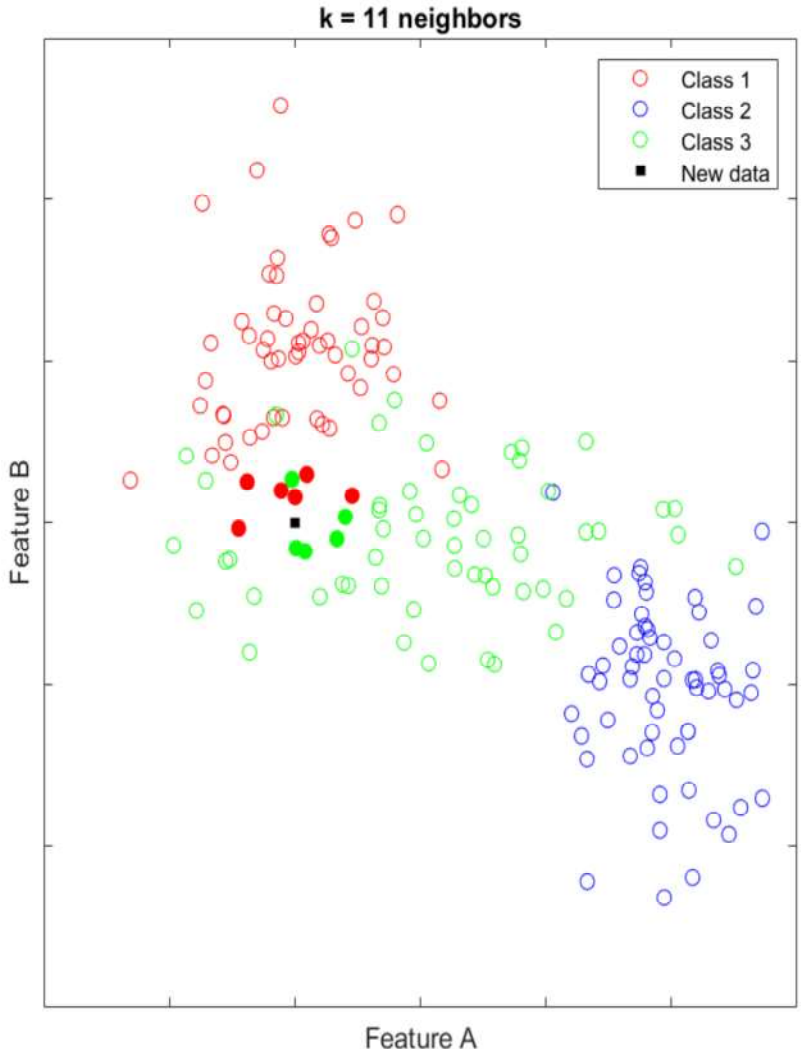
k = 11 Nearest Neighbors



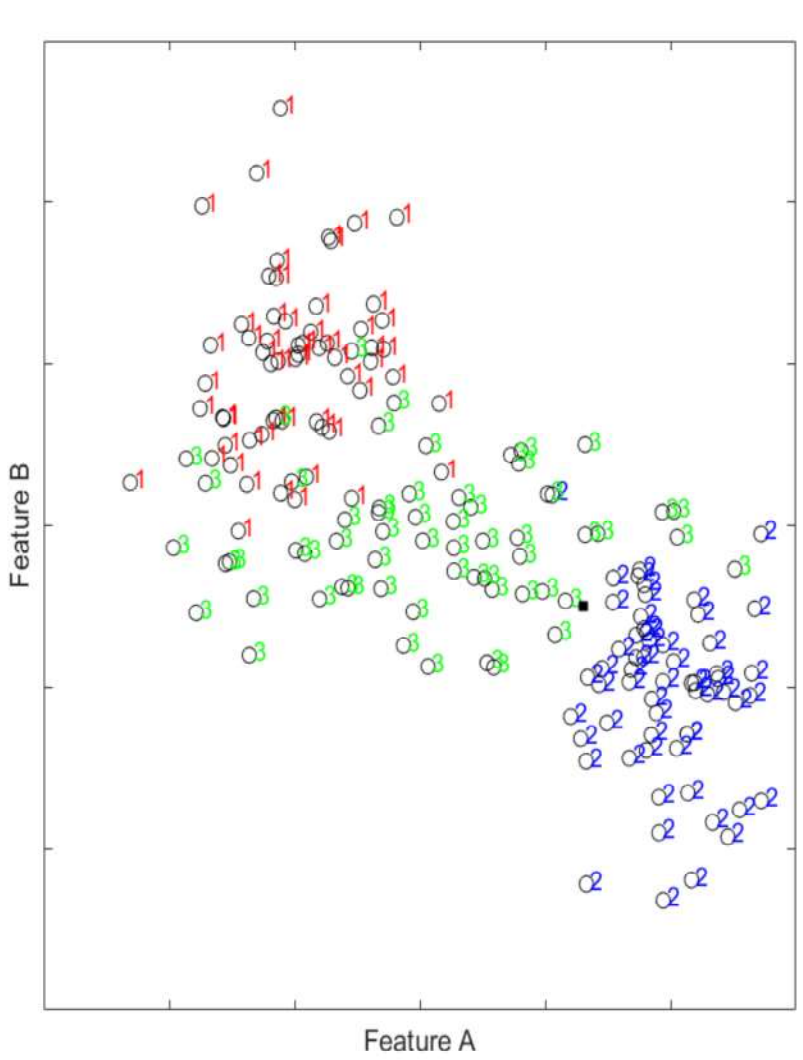


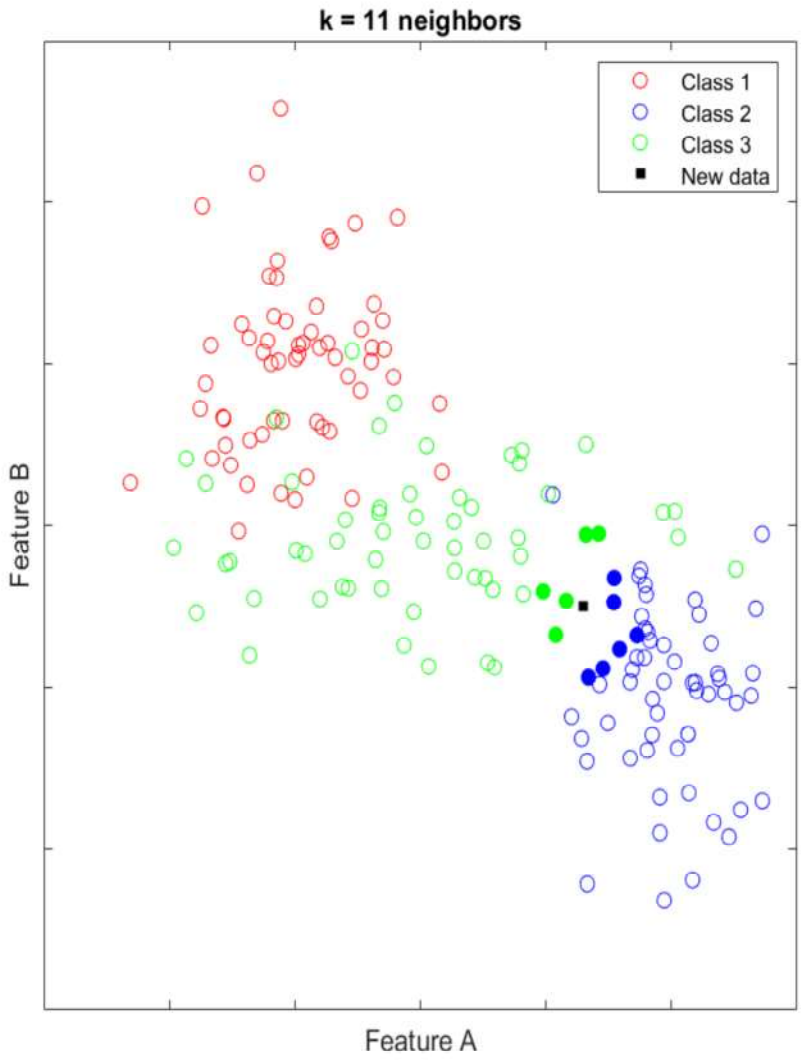
k = 11 Nearest Neighbors



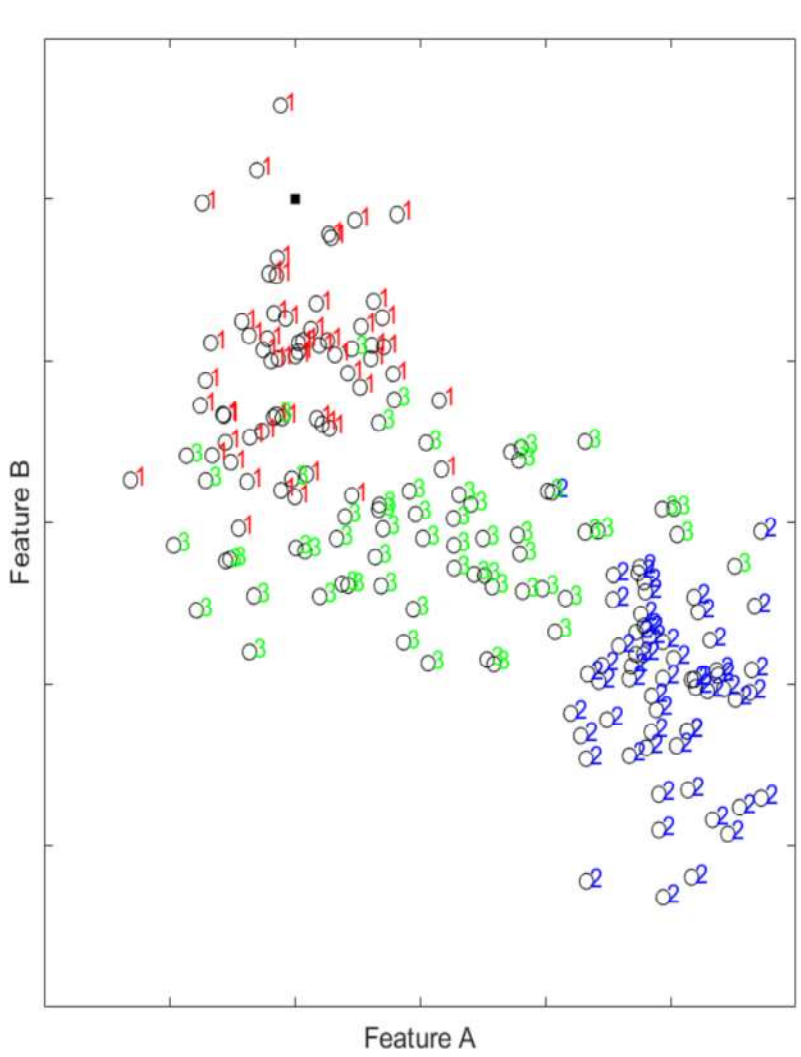


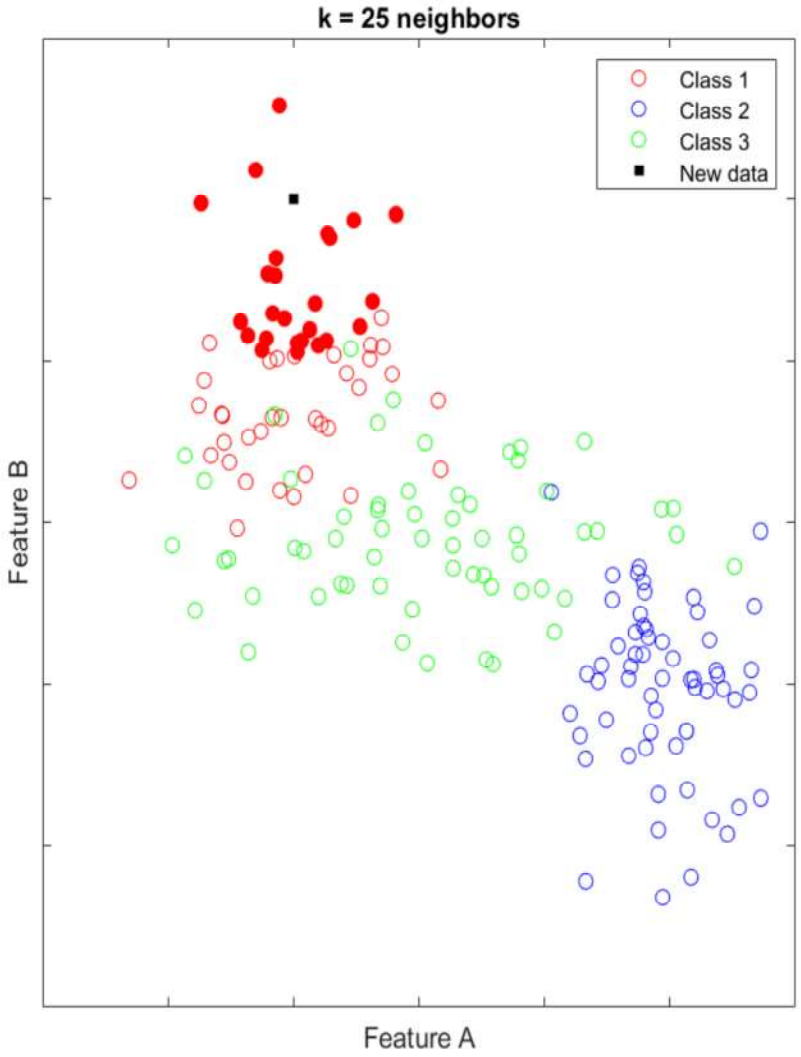
k = 11 Nearest Neighbors



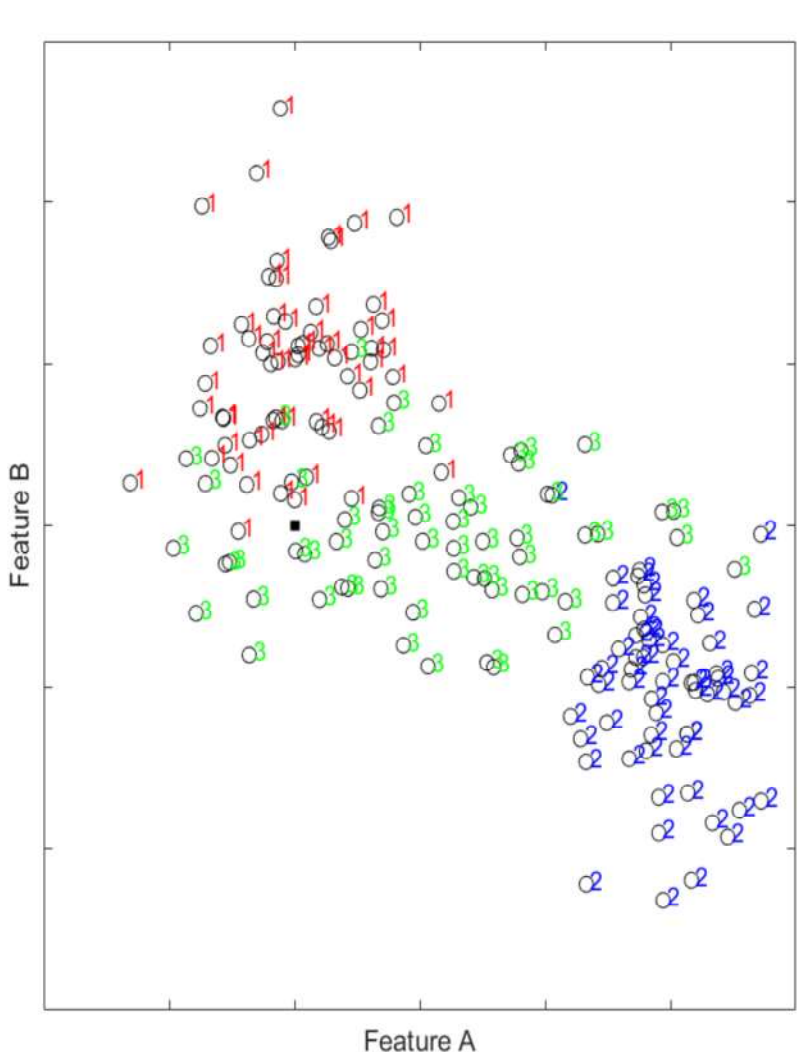


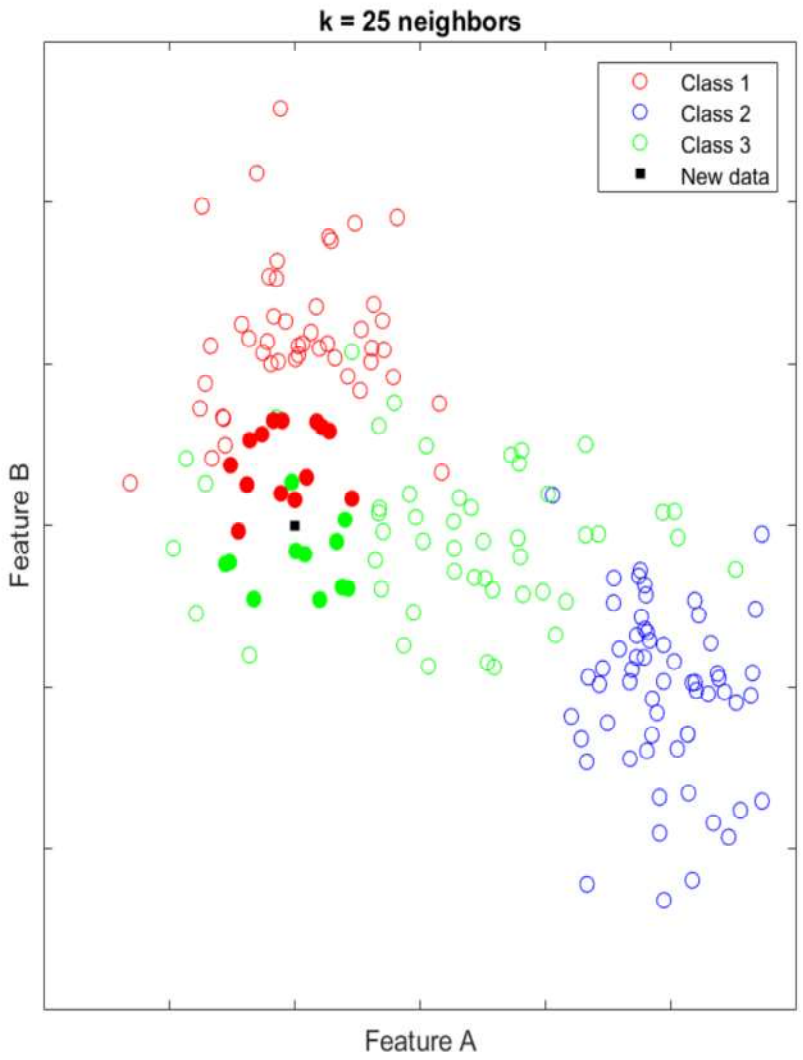
k = 25 Nearest Neighbors



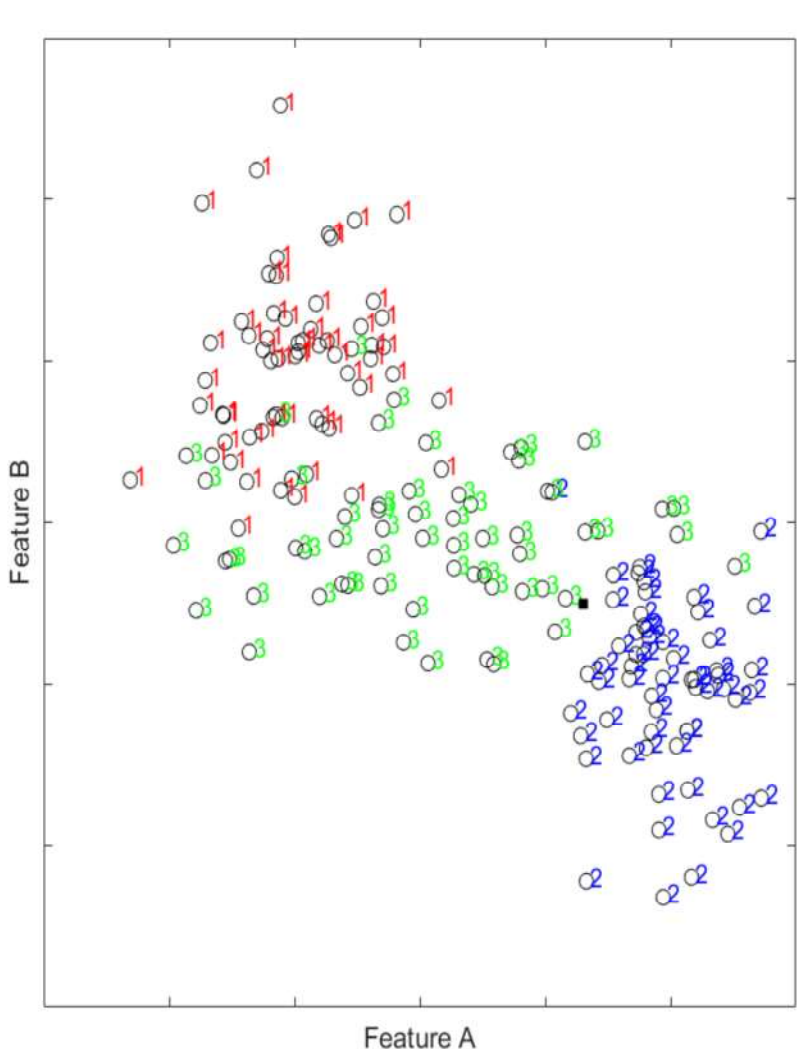


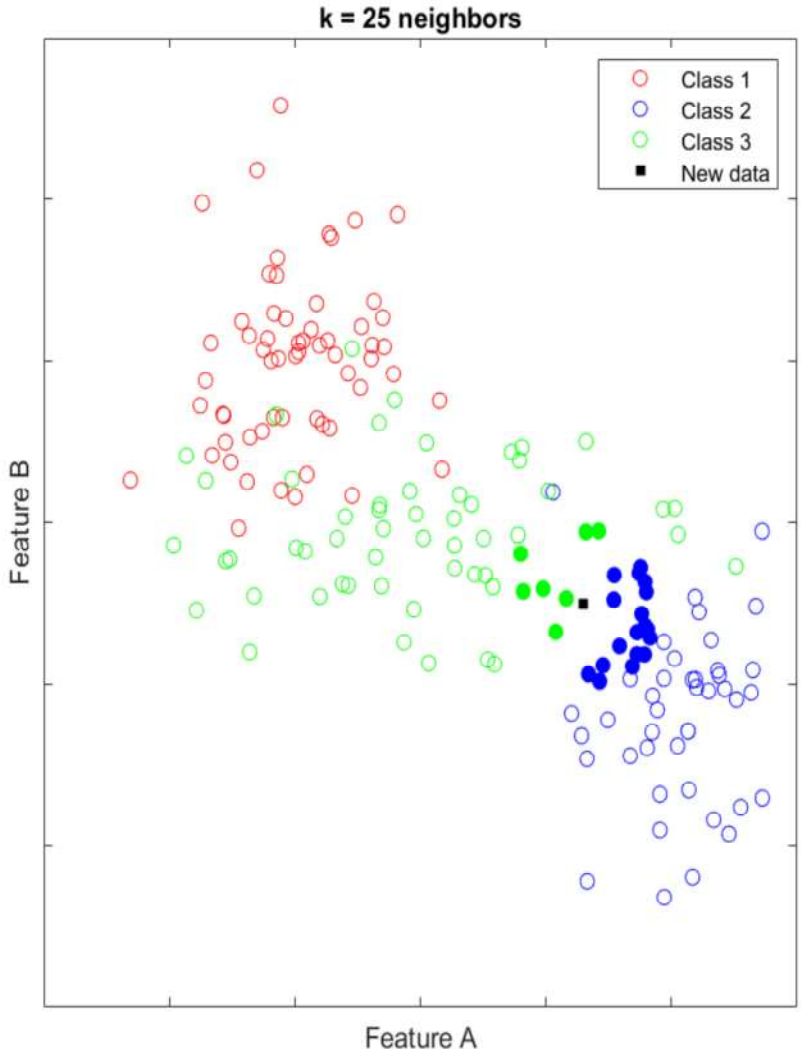
k = 25 Nearest Neighbors



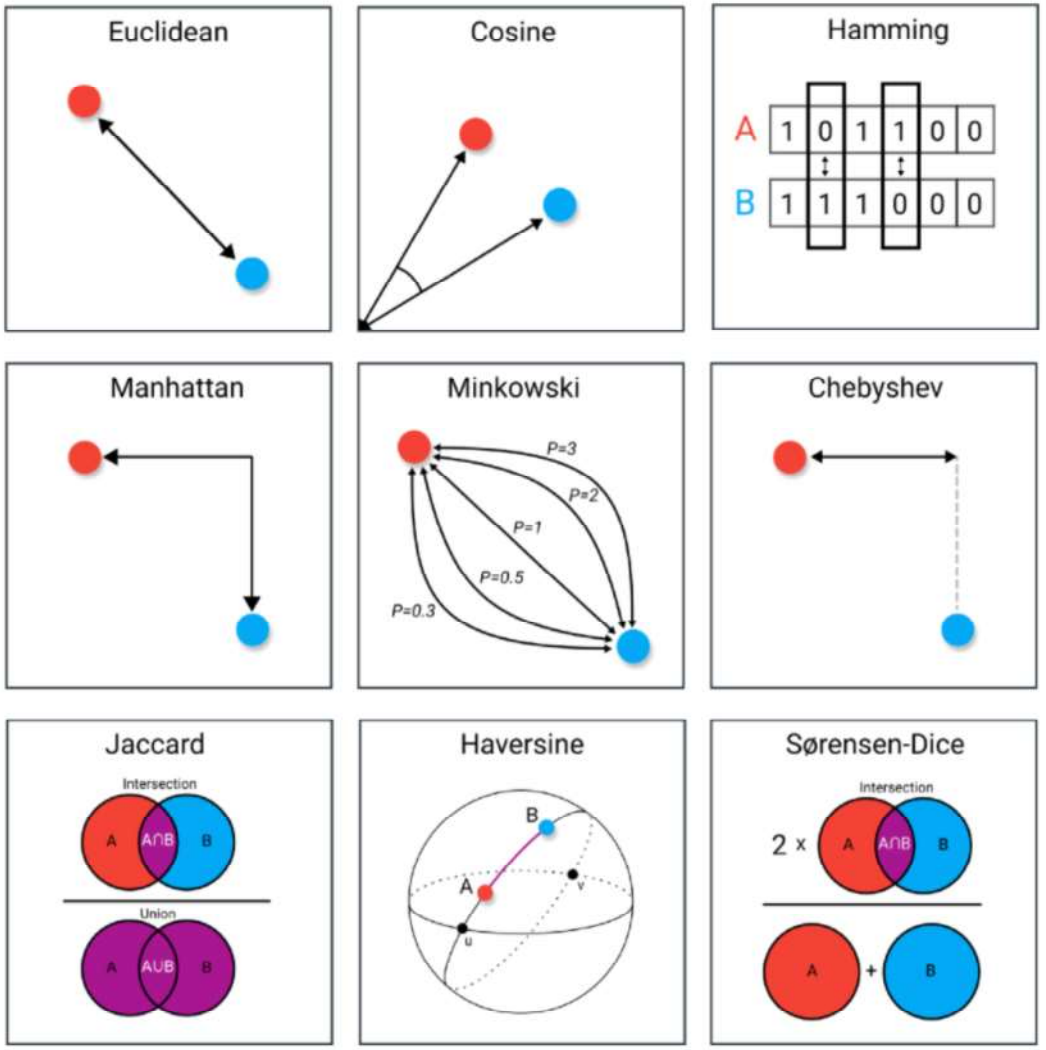


k = 25 Nearest Neighbors





Distance Measures



Source: https://towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa

Practical ML: Feature Engineering

- One-hot encoding

  red = [1, 0, 0]

  yellow = [0, 1, 0]

  green = [0, 0, 1]

- Binning / Bucketing
- Normalization
- Dealing with missing data / features

Reinforcement Learning (RL)

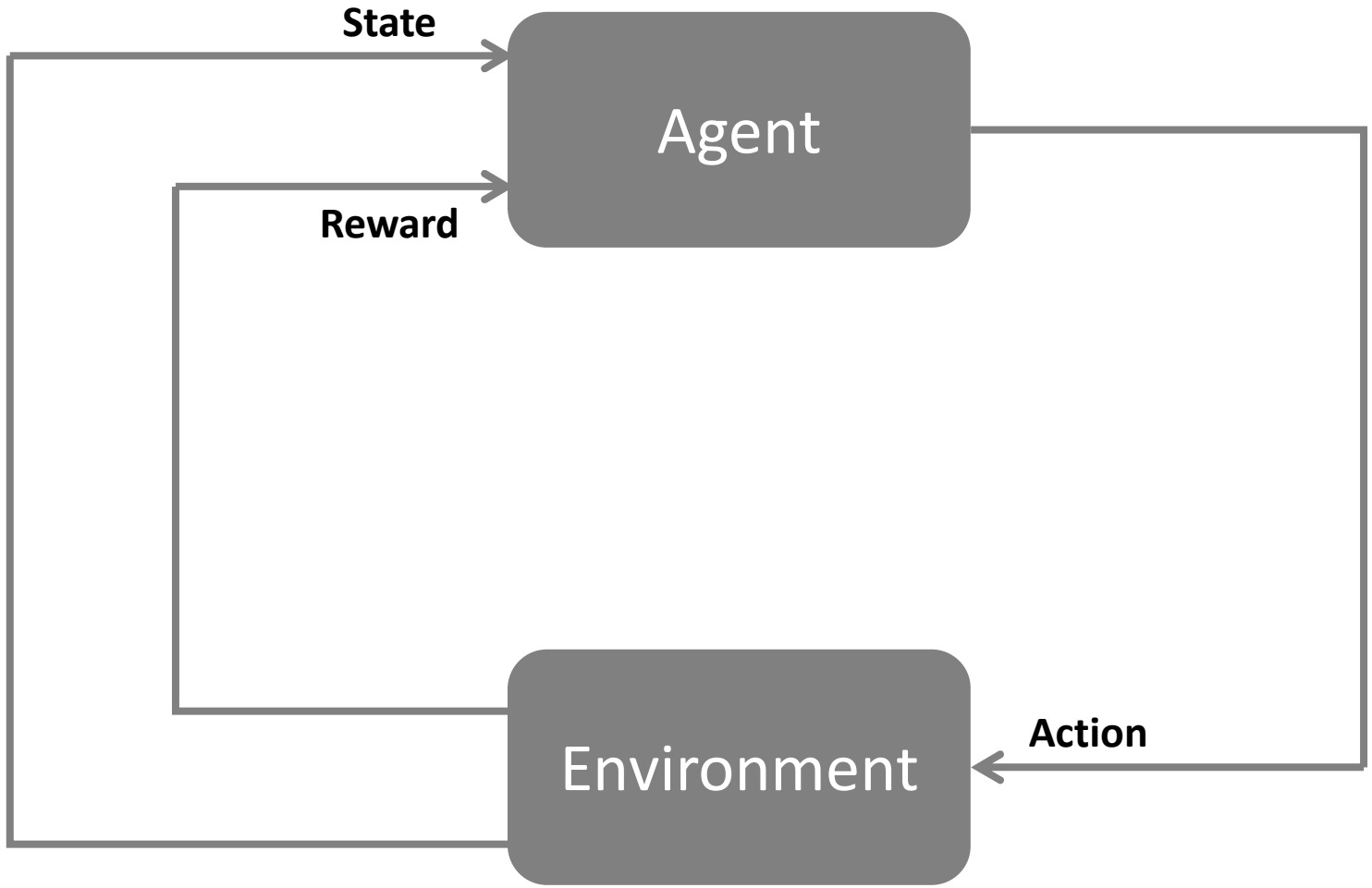
What is Reinforcement Learning?

Idea:

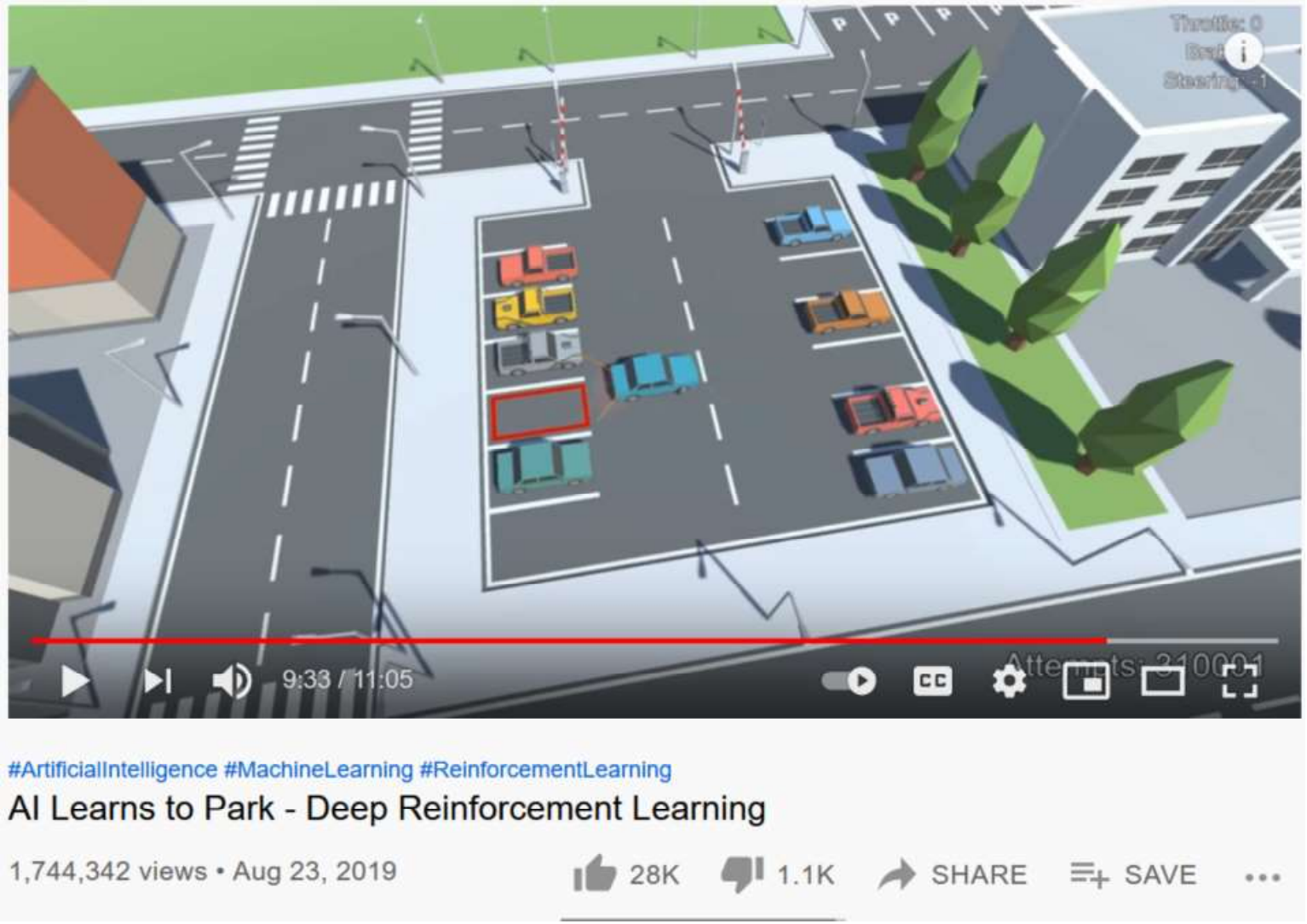
Reinforcement learning is inspired by behavioral psychology. It is **based on a rewarding / punishing an algorithm**.

Rewards and punishments are based on algorithm's action within its environment.

RL: Agents and Environments

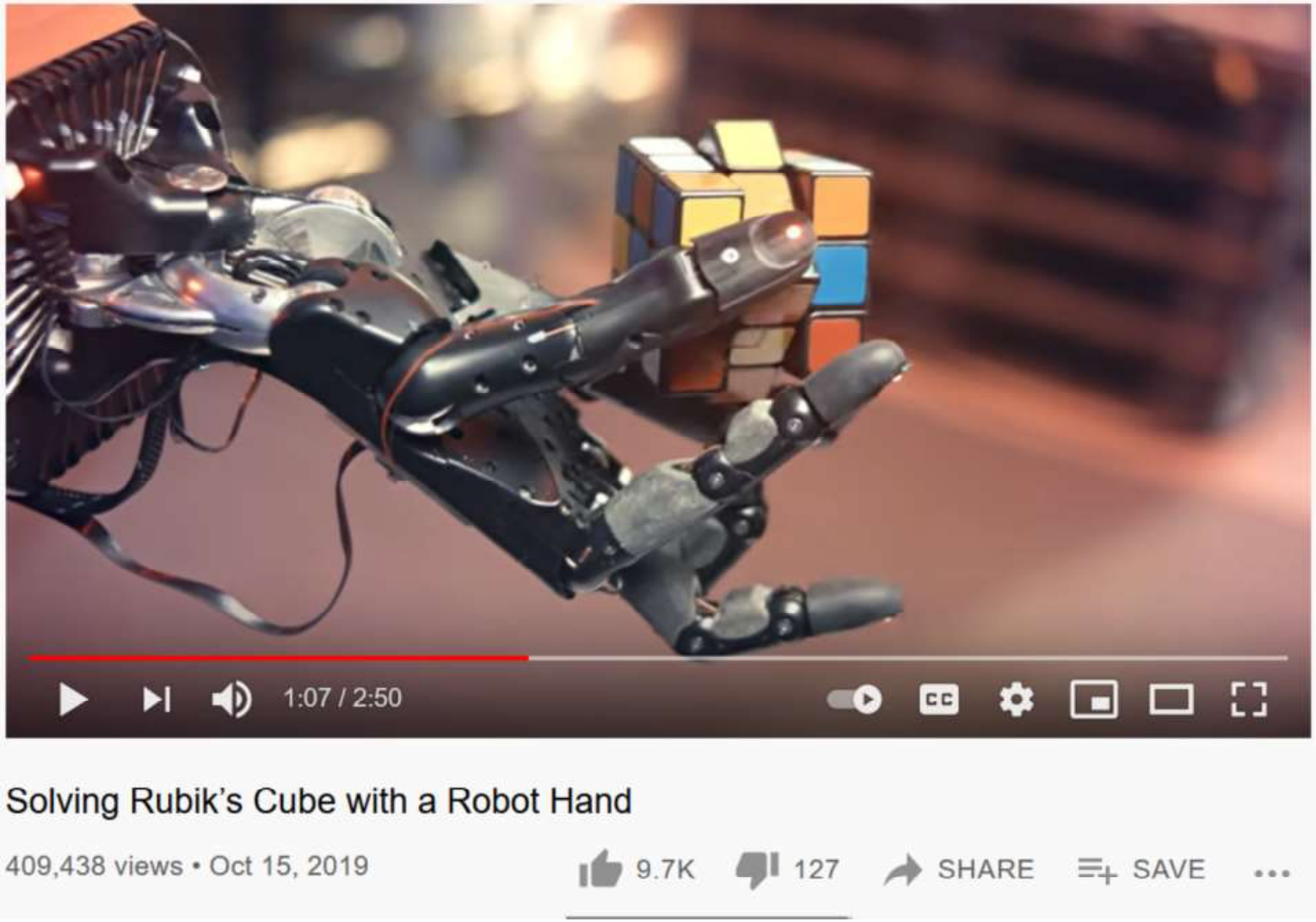


Reinforcement Learning in Action



Source: https://www.youtube.com/watch?v=VMp6pq6_QjI

Reinforcement Learning in Action



Source: https://www.youtube.com/watch?v=x4O8pojMF0w

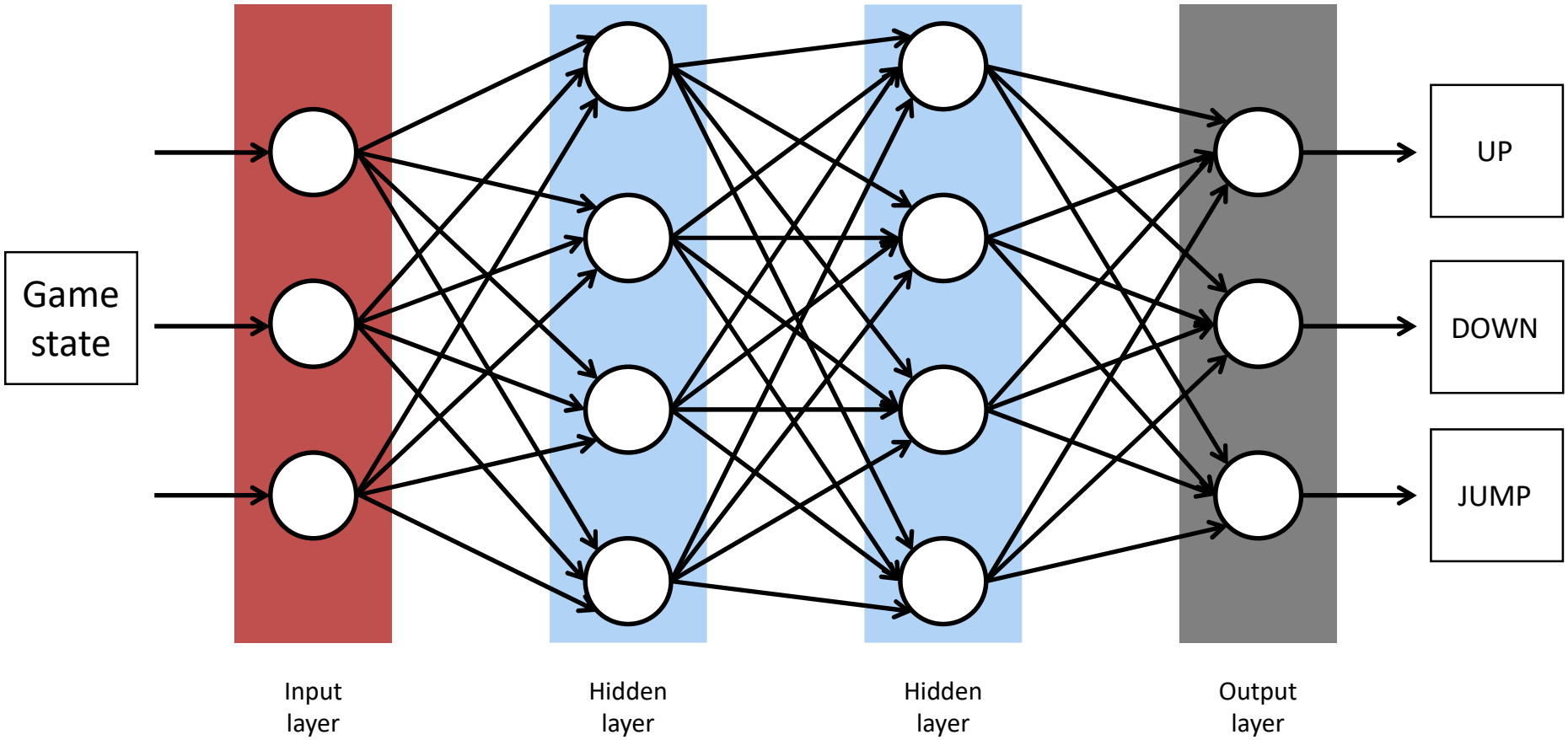
Reinforcement Learning in Action

Source: https://www.youtube.com/watch?v=kopoLzvh5jY

Reinforcement Learning in Action

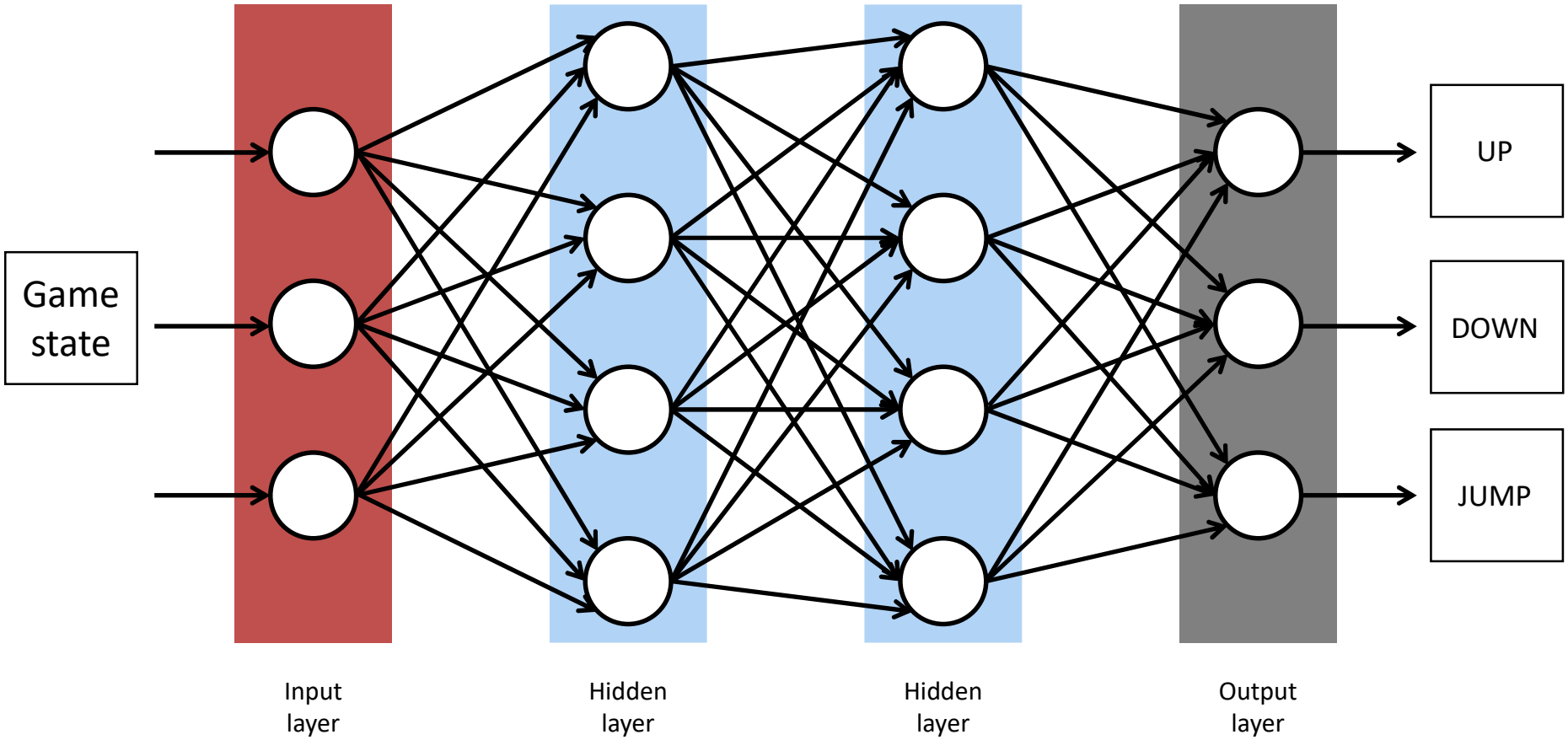
Source: https://www.youtube.com/watch?v=Tnu4O_xEmVk

ANN for Simple Game Playing



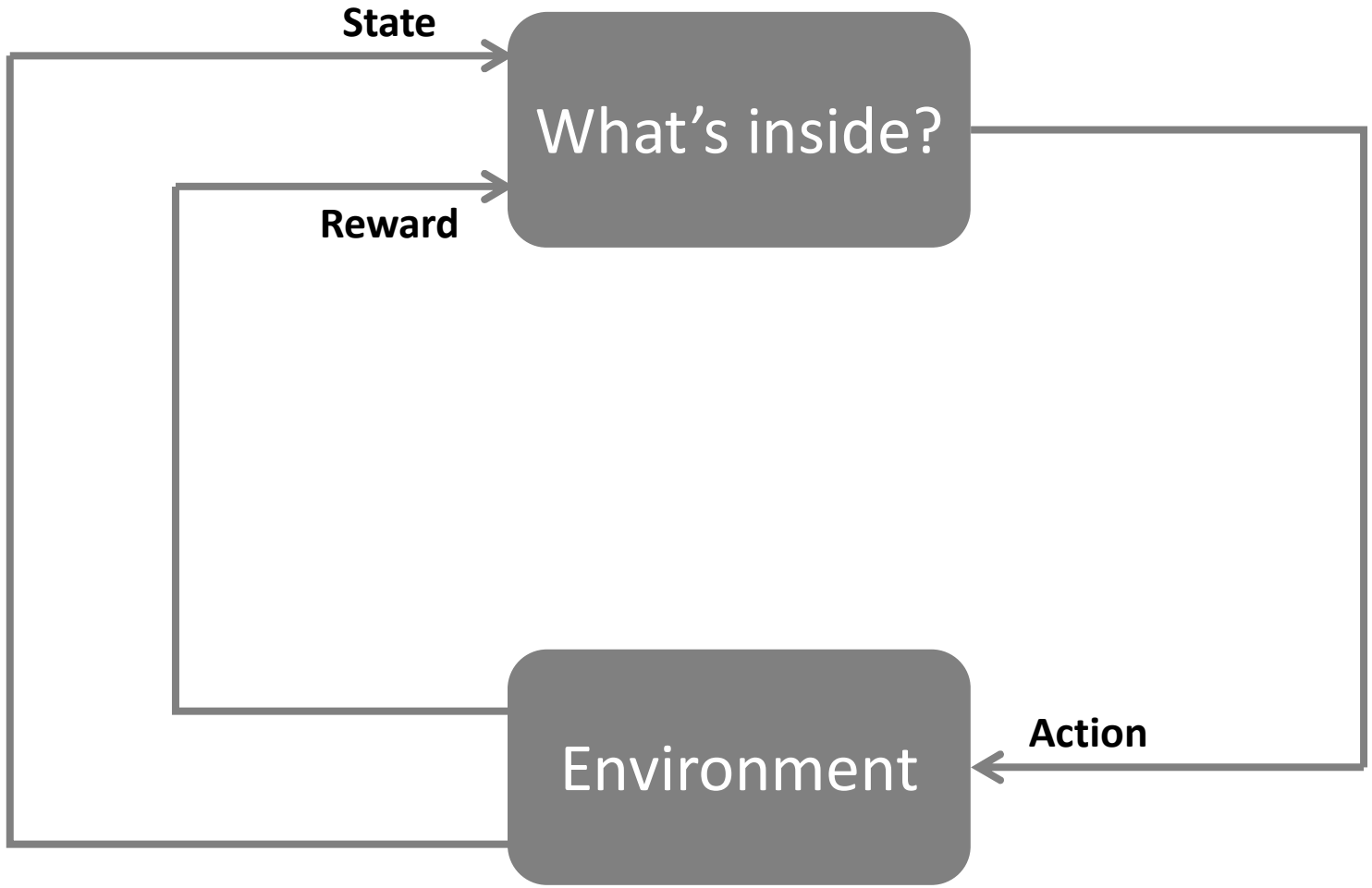
ANN for Simple Game Playing

Current game is an input. Decisions (UP/DOWN/JUMP) are rewarded/punished.

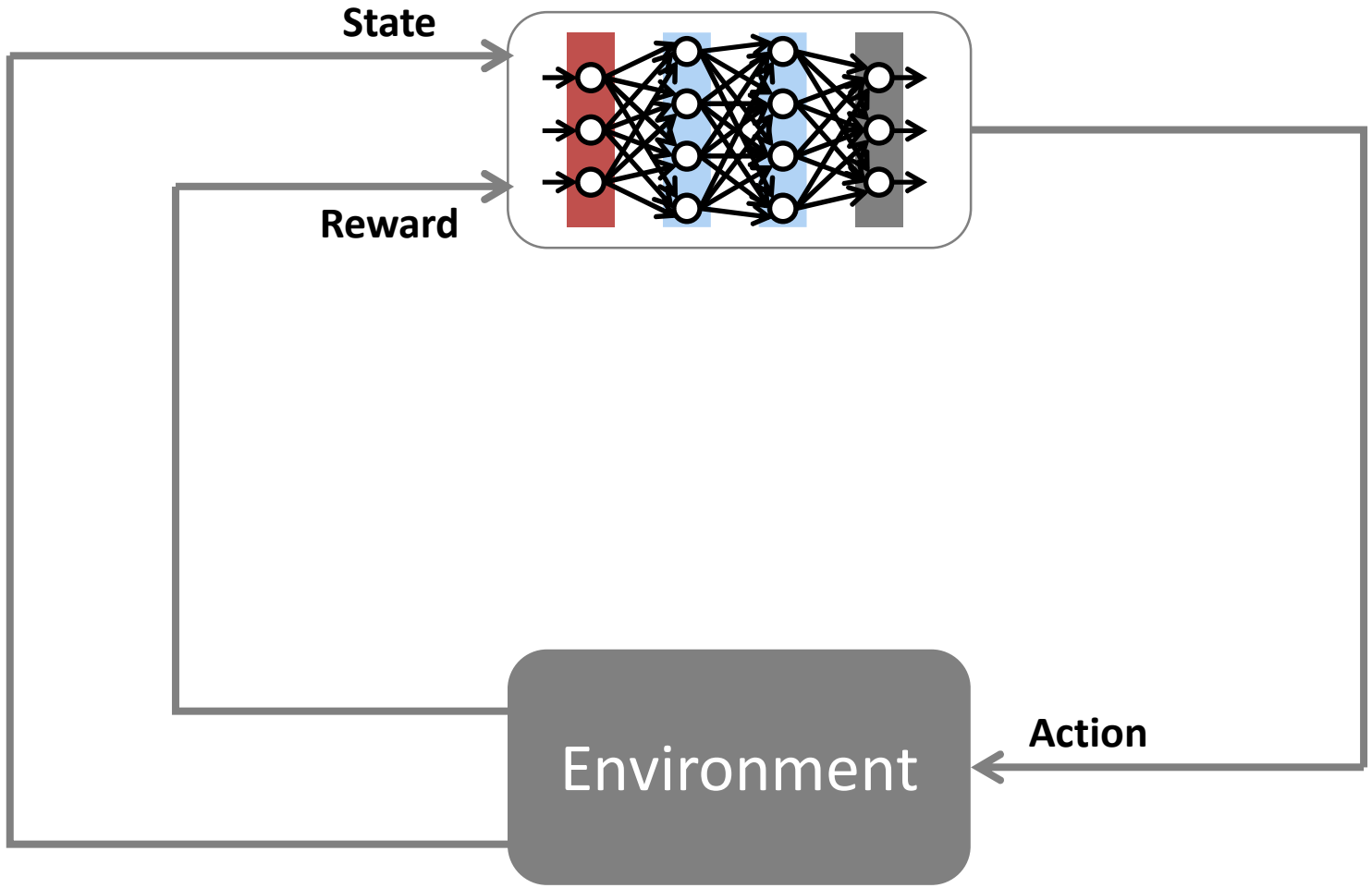


Correct all the weights using Reinforcement Learning.

RL: Agents and Environments

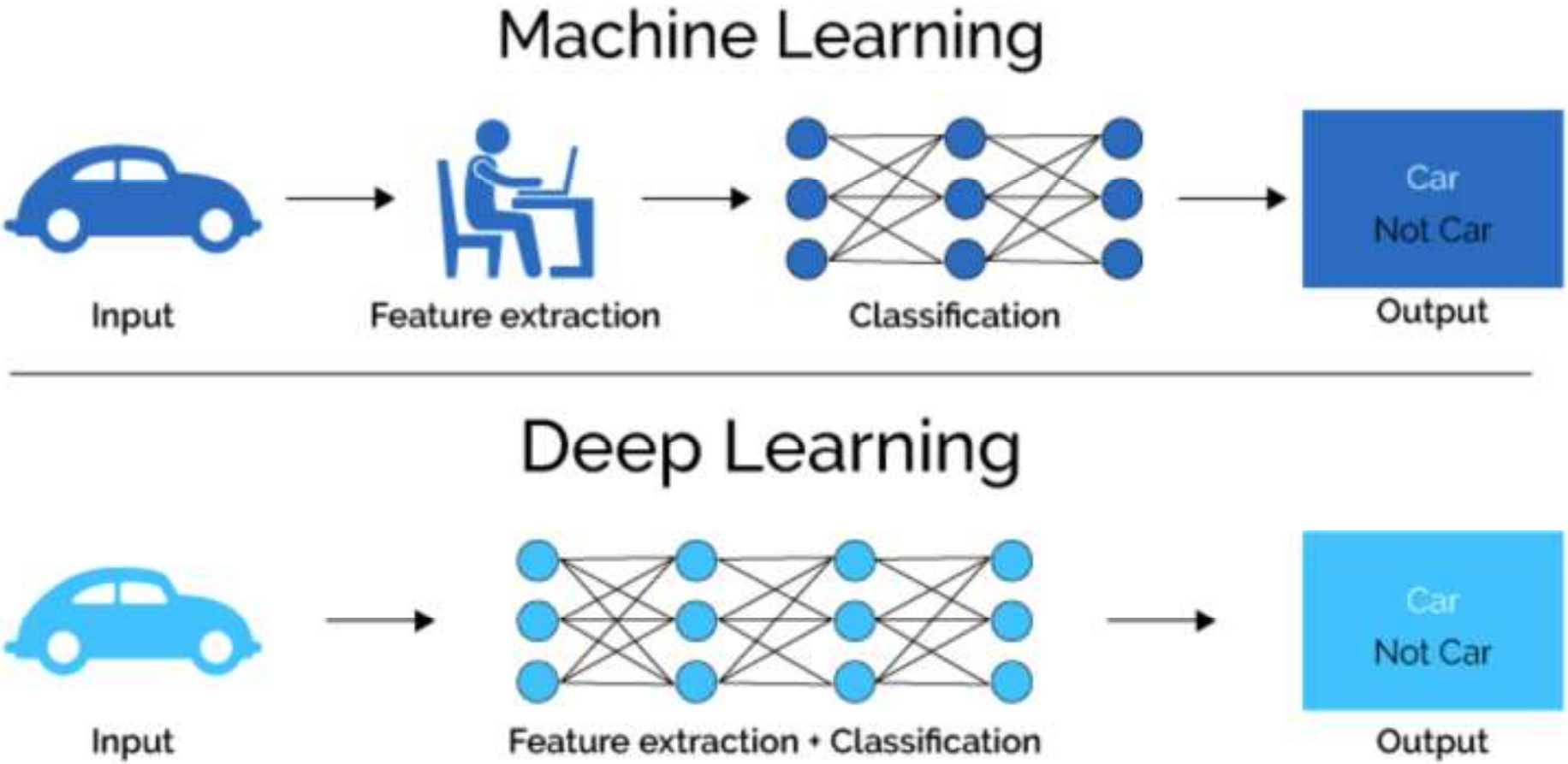


RL: Agents and Environments



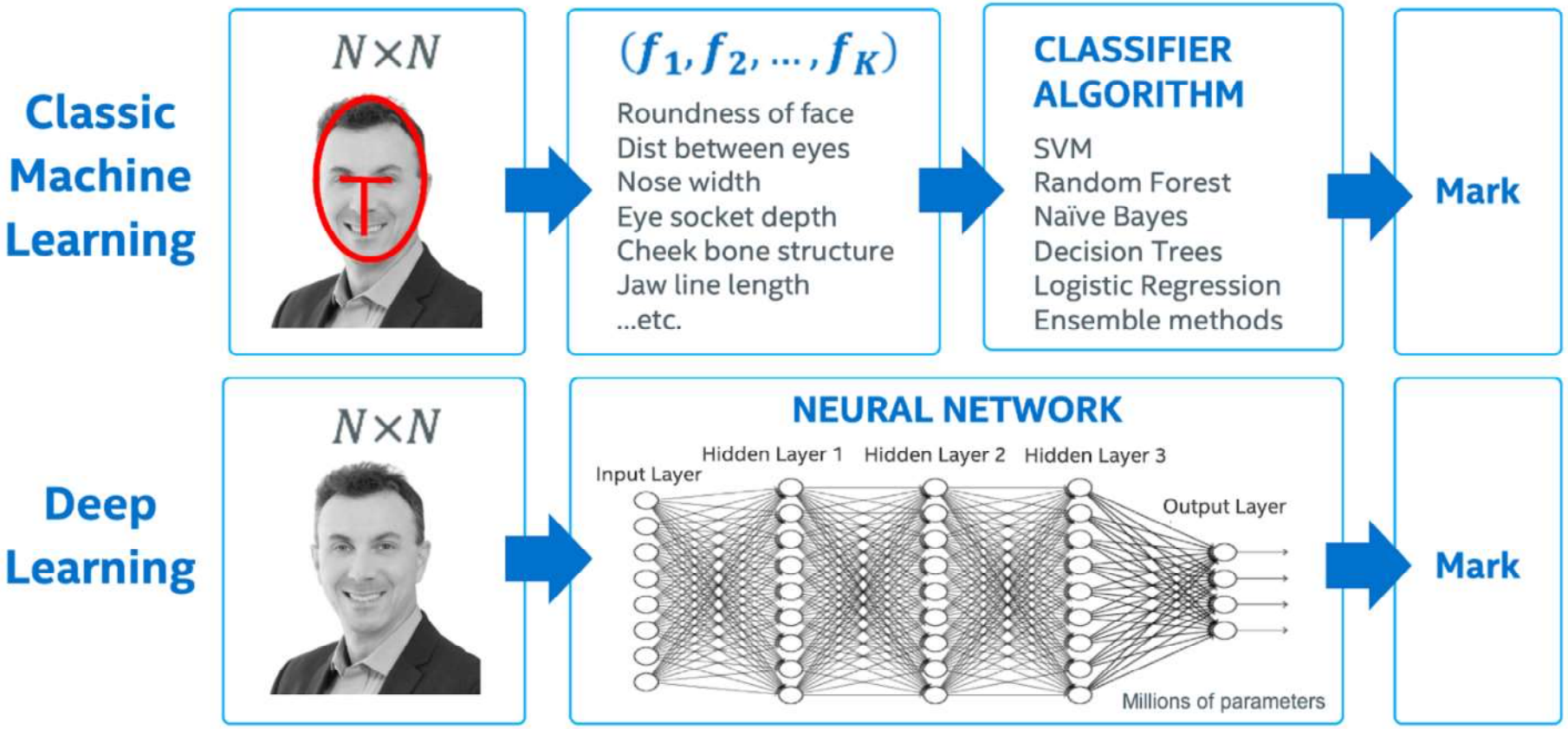
Deep Learning

Machine Learning vs. Deep Learning



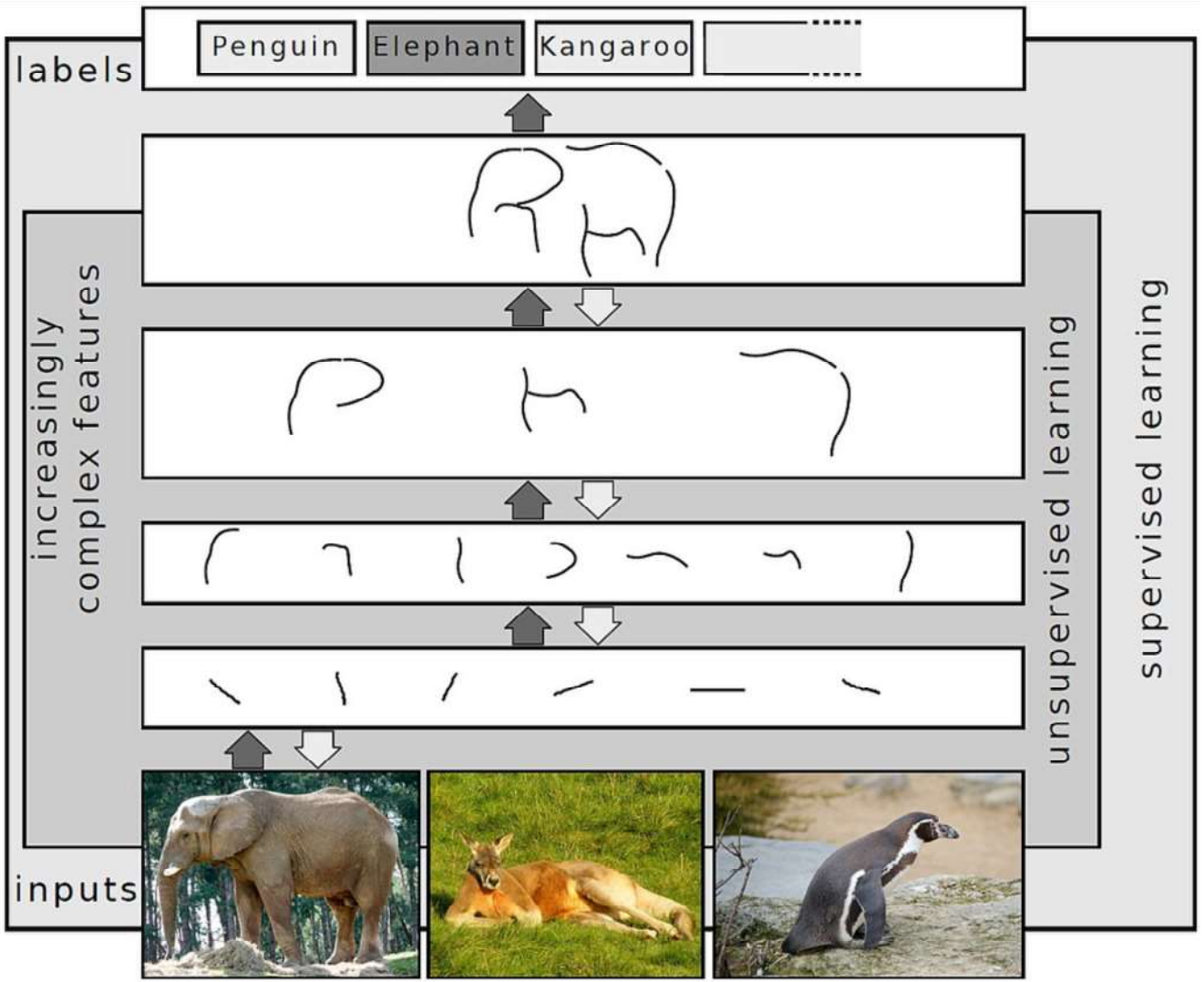
Source: https://www.quora.com/What-is-the-difference-between-deep-learning-and-usual-machine-learning

Machine Learning vs. Deep Learning



Source: https://www.intel.com/content/www/us/en/artificial-intelligence/posts/difference-between-ai-machine-learning-deep-learning.html

Deep Learning: Feature Extraction



Source: https://en.wikipedia.org/wiki/Deep_learning

Exercise: Object Recognition

https://braneshop.com.au/object-detection-in-the-browser.html

(you can try it on your smartphone)

Exercise: Image Colorizer

https://deepai.org/machine-learning-model/colorizer

Exercise: Deep Learning

https://www.handwriting-generator.com/