

Dawn of robots

An Artificial Intelligence story



Eduardo Matallanas

AI Lead @ Plain Concepts

- Knowmad interested in:
 - AI applied to industry
 - Research in robotics & ML
 - Data lover
 - Love films and series



@ematde



ematallanas@plainconcepts.com



What is AI?





Platforms

Boston Dynamics



SpotMini



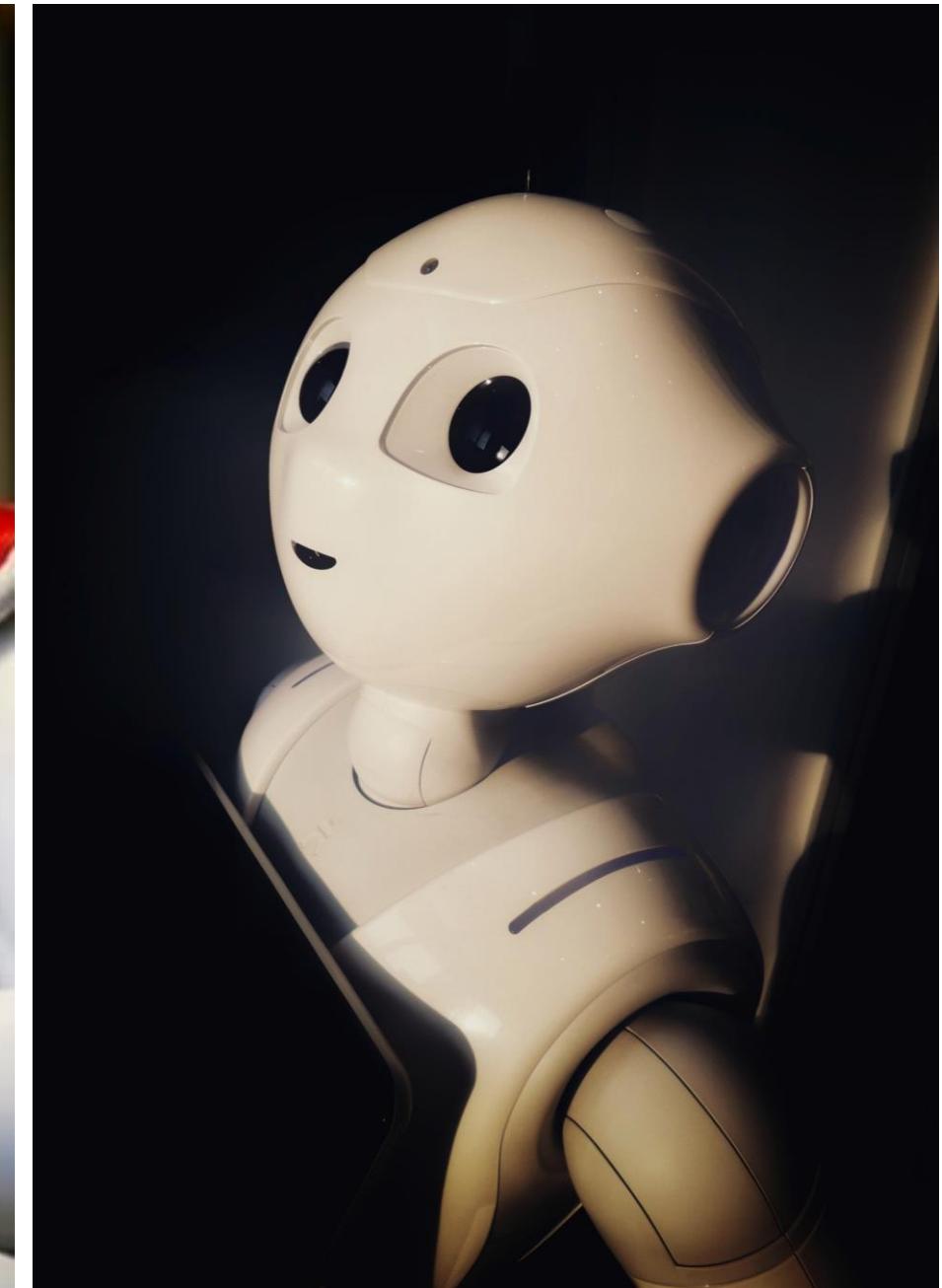
Spot

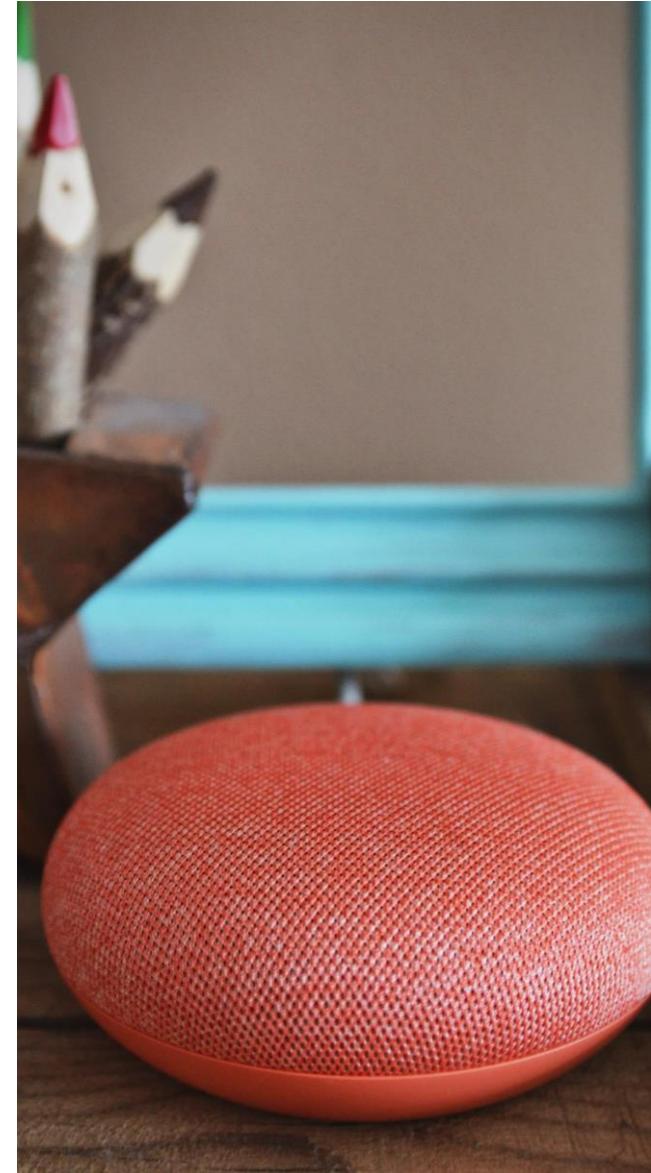


Atlas



Handle





SOURCE: GOOGLE





Artificial Intelligence

- More than simply code
- Solve hard human tasks
- Capacity to learn from the environment
- Intelligent Agents

“

Any system that interacts with the environment, perceives it, learns from it, and takes actions to achieve successfully its goals and tasks through flexible adaptation.

```
1 <?php header( 'Content-Type: text/html; charset=UTF-8' ); ?>
2 <?php language_attributes(); ?>
3 <?php wp_head(); ?>
4 <?php wp_title( '|', true, 'right' ); ?>
5 <?php bloginfo( 'charset' ); ?>
6 <?php bloginfo( 'viewport' ); ?>
7 <?php bloginfo( 'url' ); ?>
8 <?php bloginfo( 'description' ); ?>
9 <?php bloginfo( 'generator' ); ?>
10 <?php bloginfo( 'pingback_url' ); ?>
11 <?php echo get_template_directory_uri(); ?>
12 <?php echo get_stylesheet_directory_uri(); ?>
13 <?php echo get_stylesheet(); ?>
14 <?php echo get_header(); ?>
15 <?php echo get_sidebar(); ?>
16 <?php echo get_footer(); ?>
17 <?php echo get_search_form(); ?>
18 <?php echo get_calendar(); ?>
19 <?php echo get_calendar('j'); ?>
20 <?php echo get_calendar('y'); ?>
21 <?php echo get_calendar('yy'); ?>
22 <?php echo get_calendar('yyj'); ?>
23 <?php echo get_calendar('yyjy'); ?>
24 <?php echo get_calendar('yyjyj'); ?>
25 <?php echo get_calendar('yyjyjy'); ?>
26 <?php echo get_calendar('yyjyjyj'); ?>
27 <?php echo get_calendar('yyjyjyjy'); ?>
28 <?php echo get_calendar('yyjyjyjyj'); ?>
29 <?php echo get_calendar('yyjyjyjyjy'); ?>
30 <?php echo get_calendar('yyjyjyjyjyj'); ?>
31 <?php echo get_calendar('yyjyjyjyjyjy'); ?>
32 <?php echo get_calendar('yyjyjyjyjyjyj'); ?>
33 <?php echo get_calendar('yyjyjyjyjyjyjy'); ?>
34 <?php echo get_calendar('yyjyjyjyjyjyjyj'); ?>
35 <?php echo get_calendar('yyjyjyjyjyjyjyjy'); ?>
```



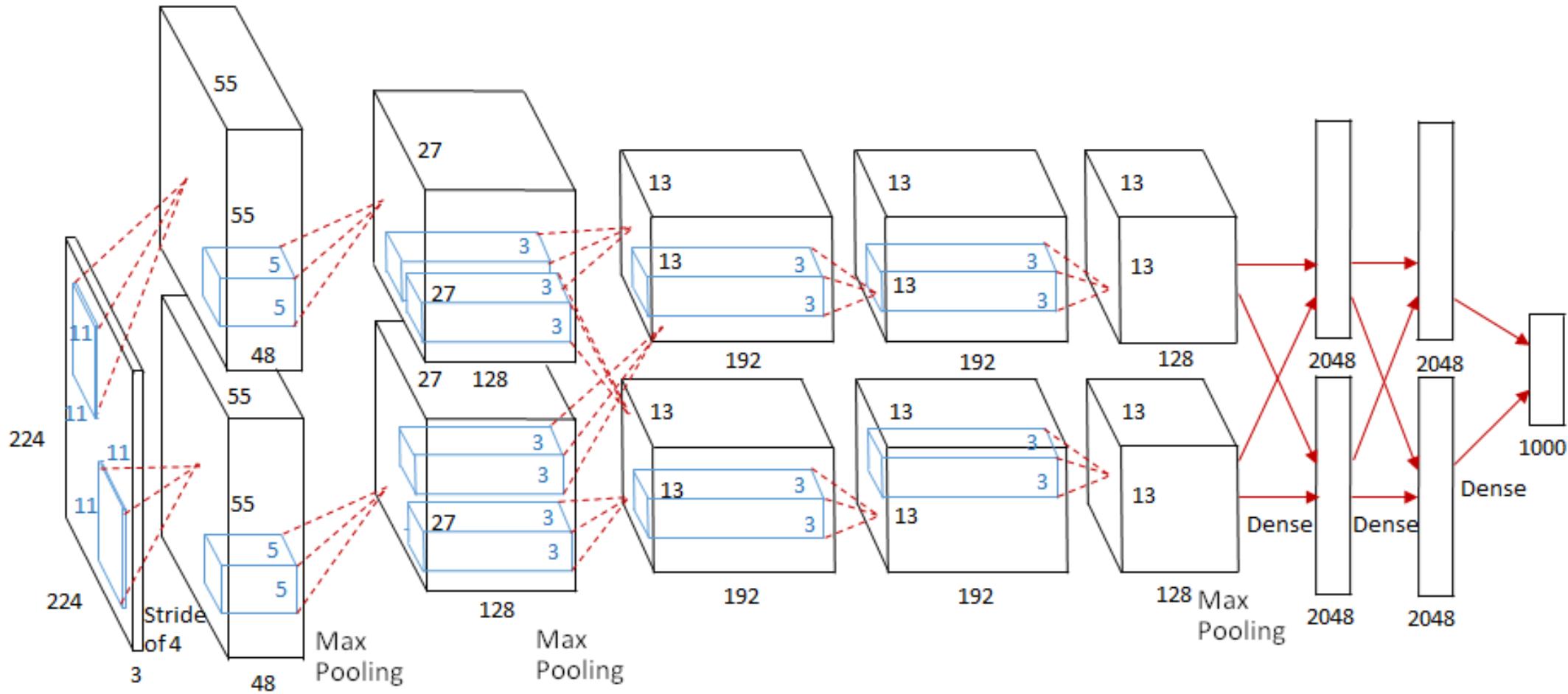
Things that you will learn today (hopefully)

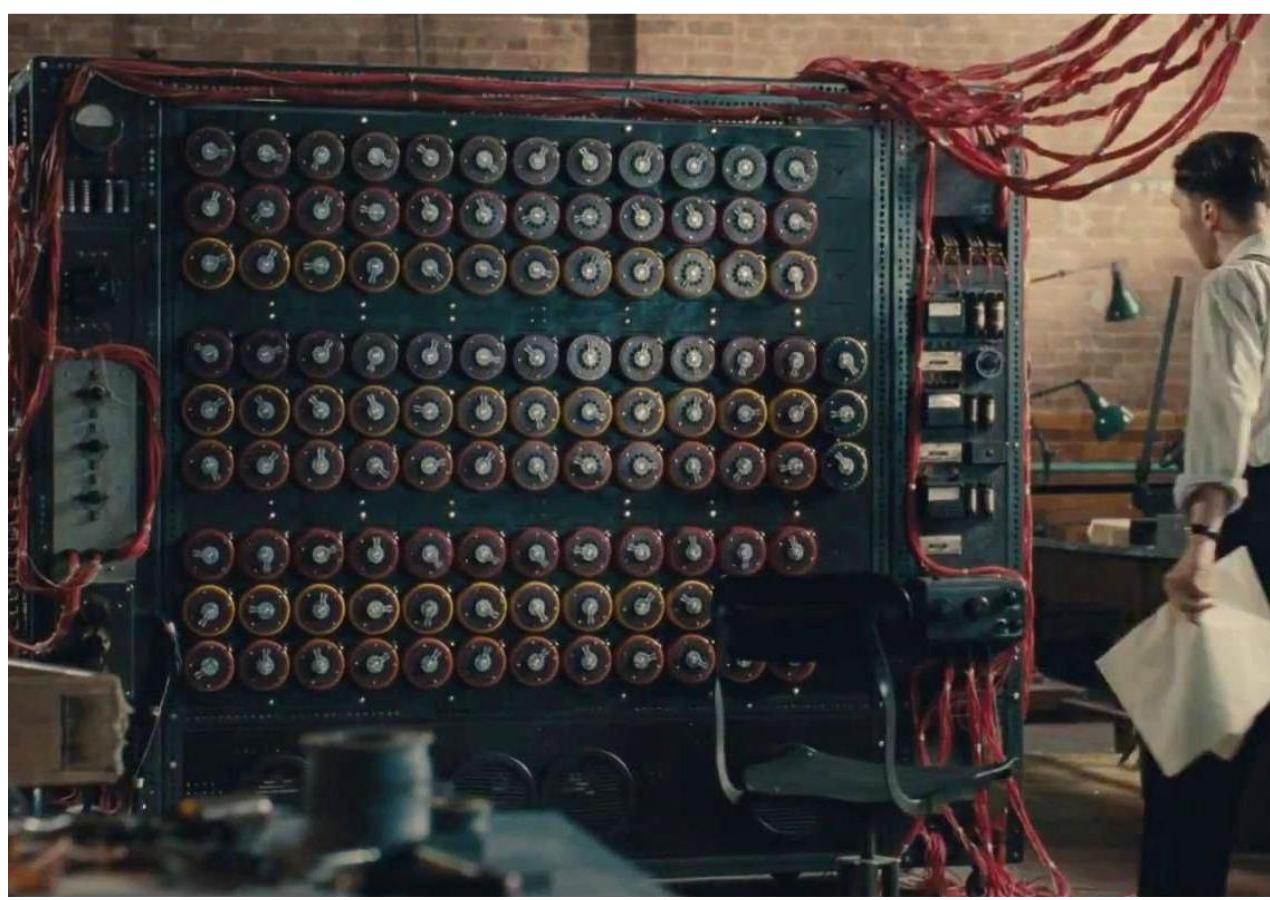
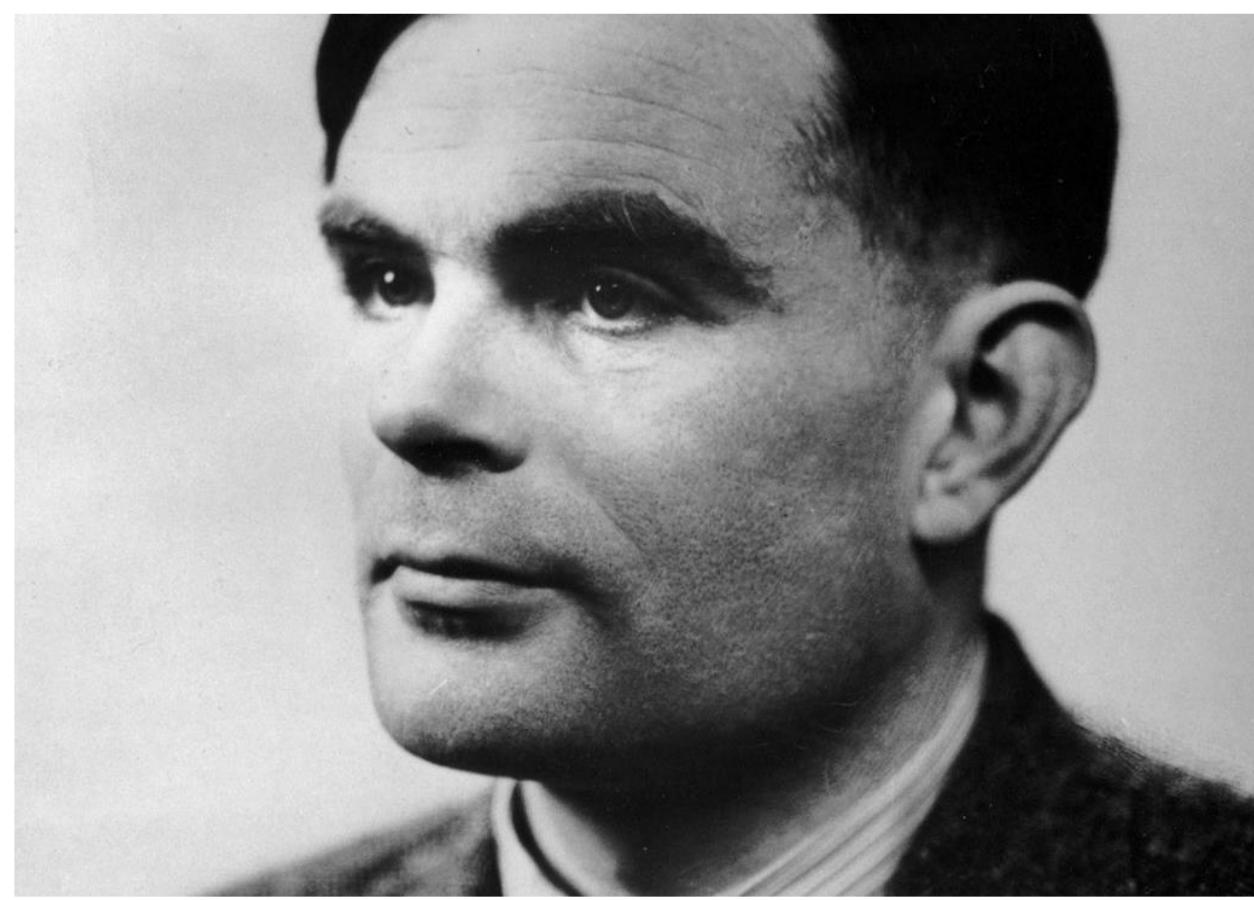
1. What is AI
2. History of AI
3. Why is important to know about it
4. Problems that face the AI
5. How to face a AI project

An AI history review



WHAT IS THIS THING?

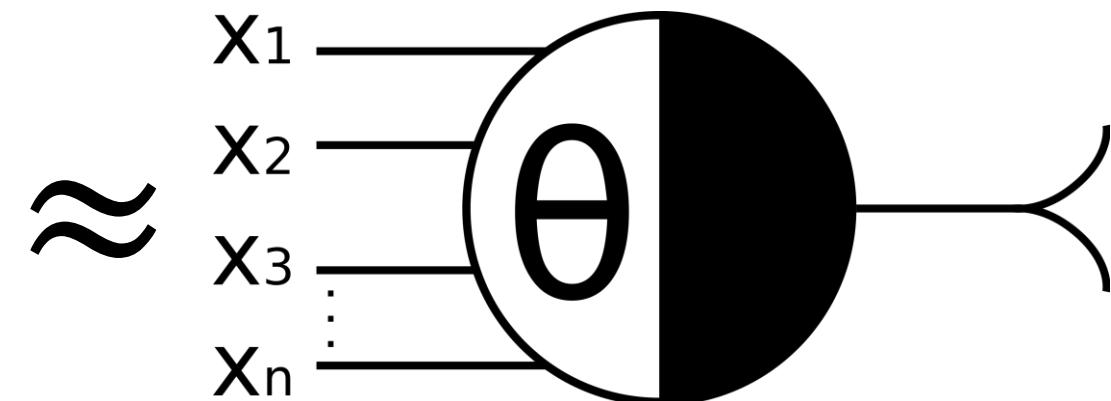
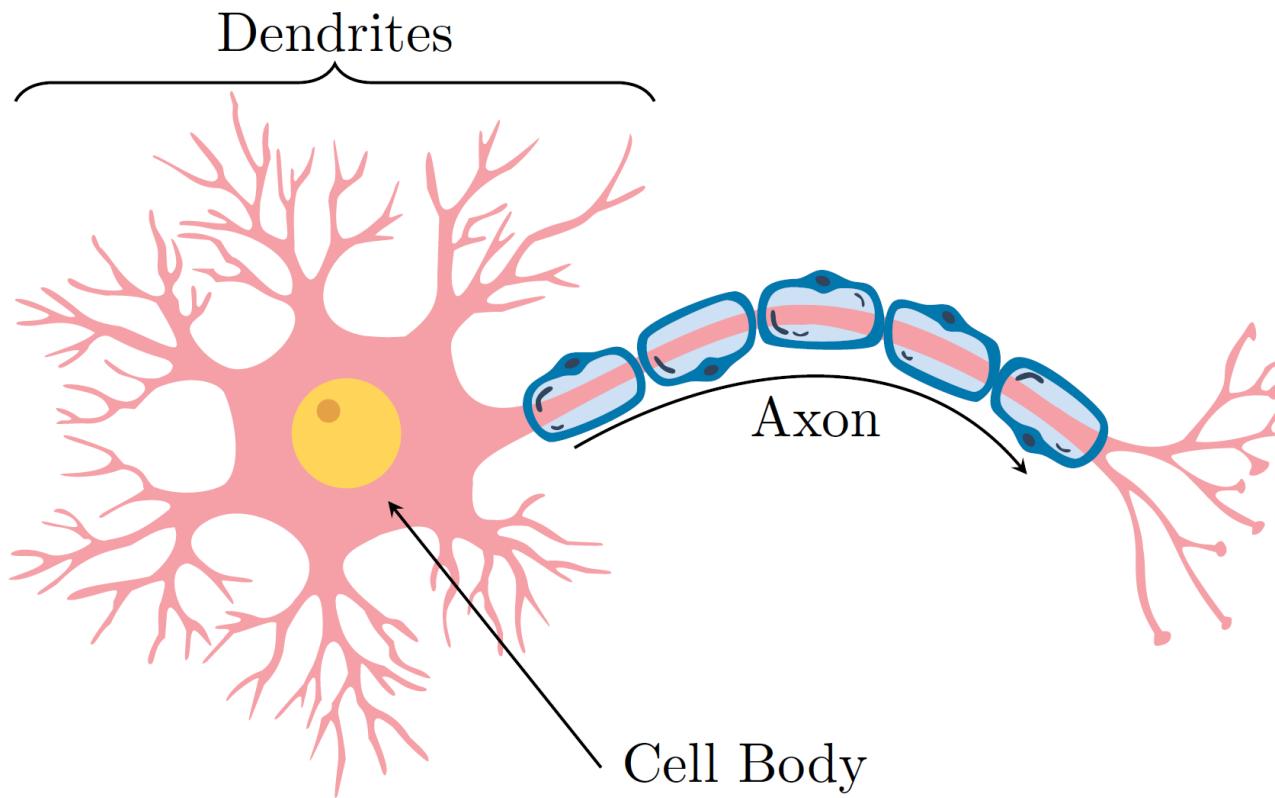




1936: Alan Turing

Turing Machine → Manipulate symbols and simulate mathematical deduction

1943: McCulloch & Pitts Neuron





50s: Al rocks

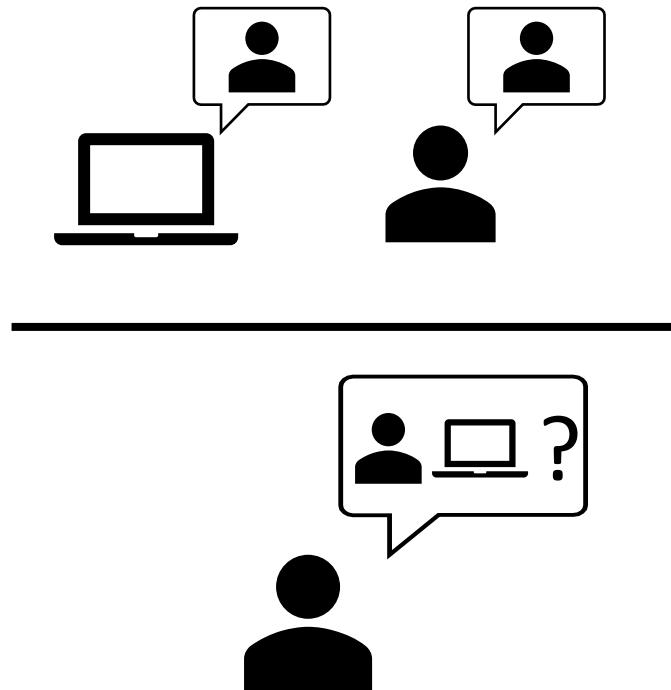
50s: AI Revolution

1949: Hebb rule

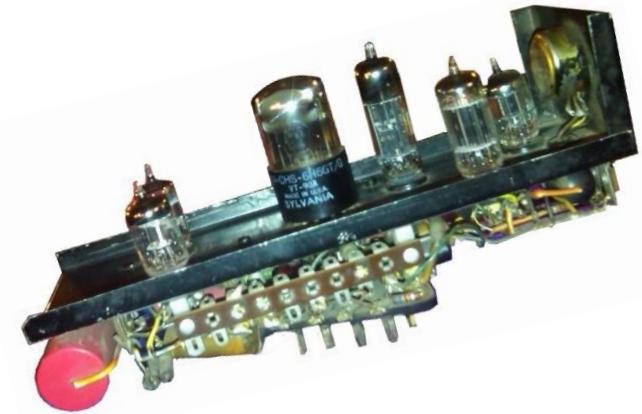


$$\Delta w_i = \eta \cdot x_i \cdot y_i$$

1950: Turing Test



1951: SNARC



Stochastic Neural Analog
Reinforcement Computer



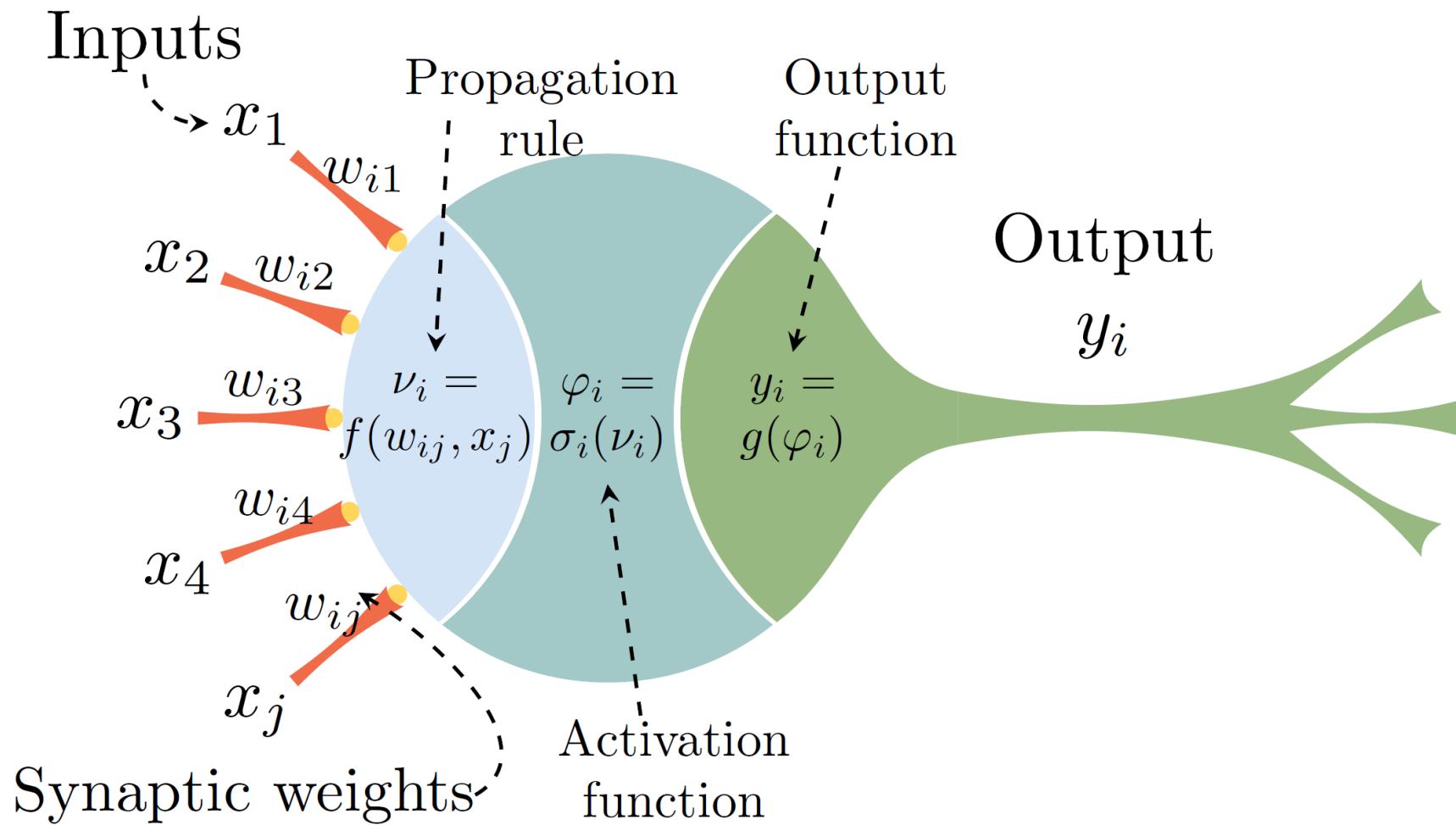
Dartmouth Conference



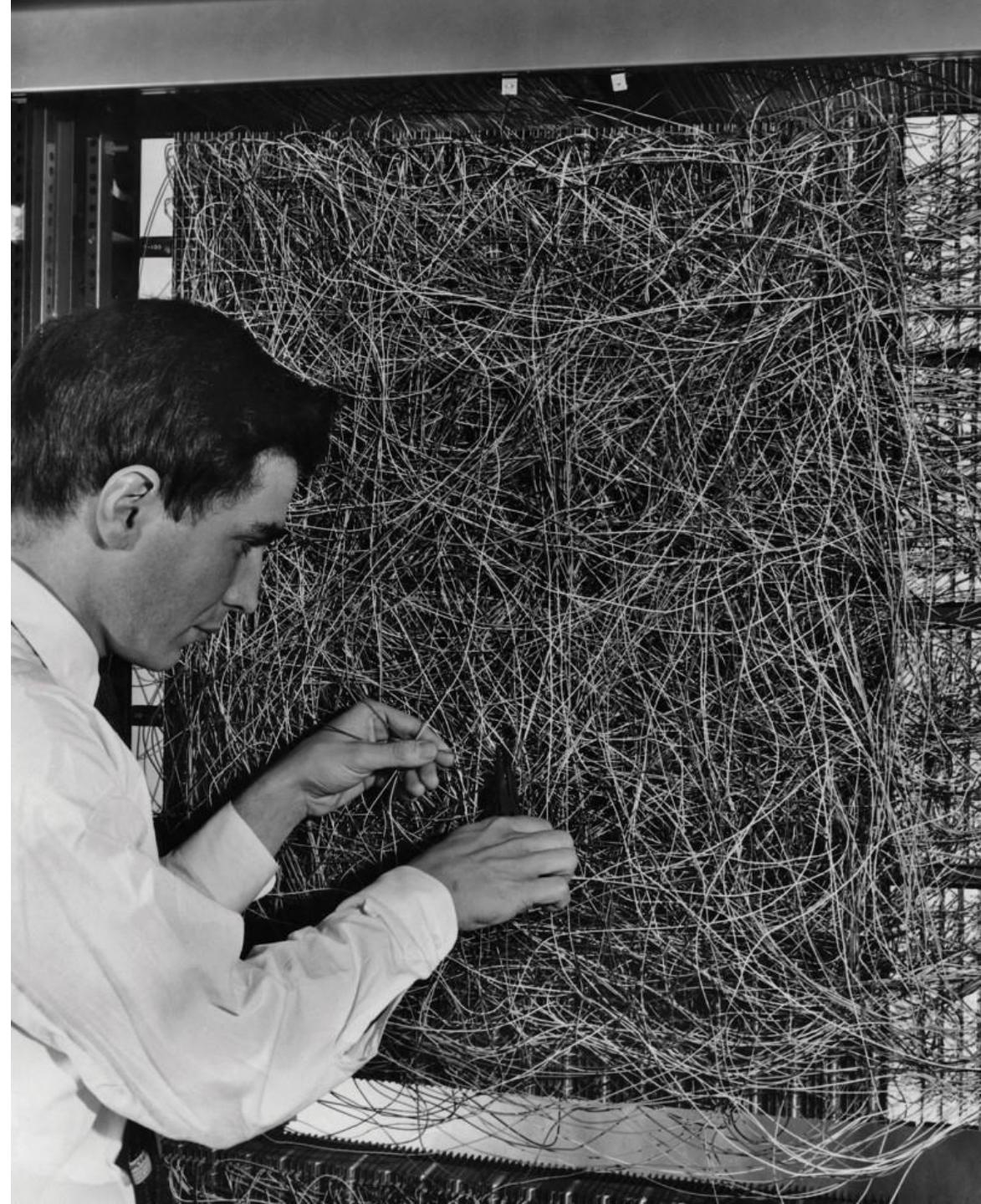
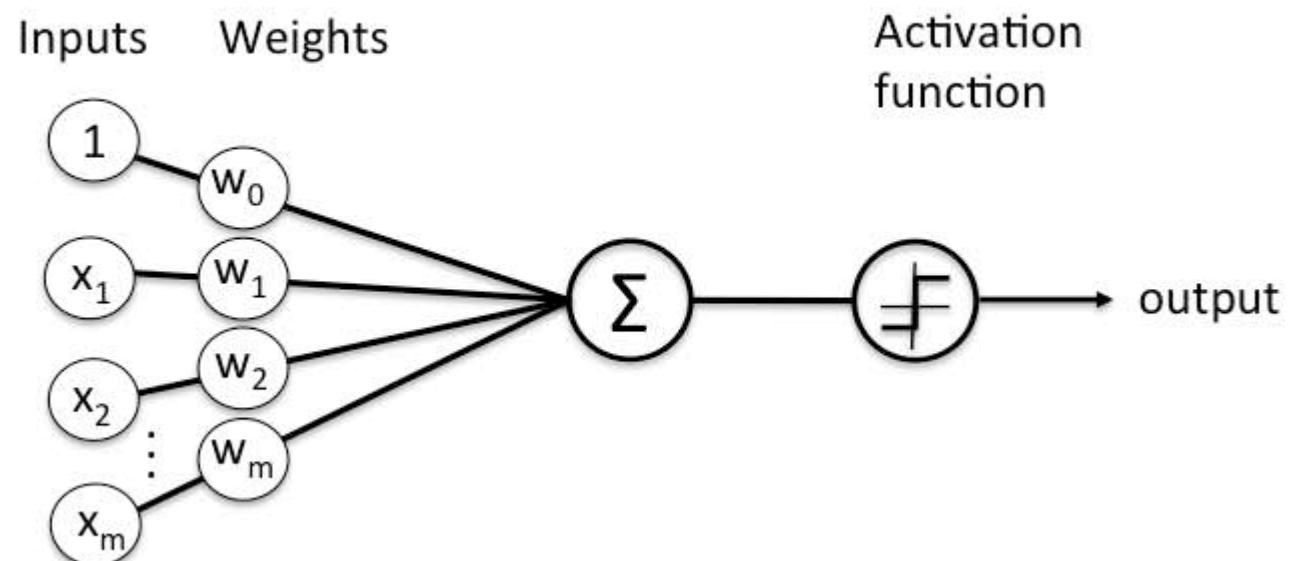


AI

1957: Rosenblatt Perceptron

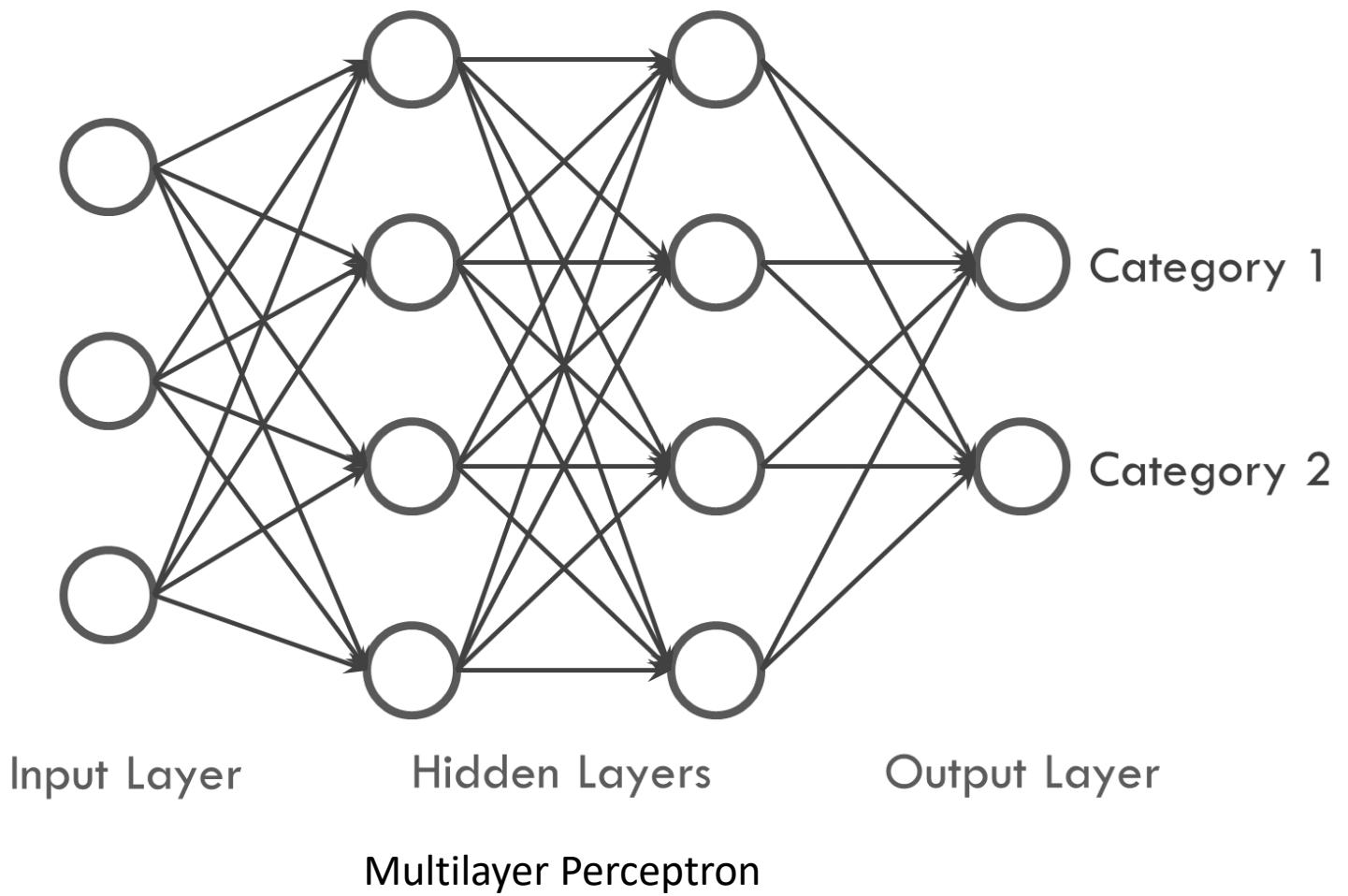


Perceptron



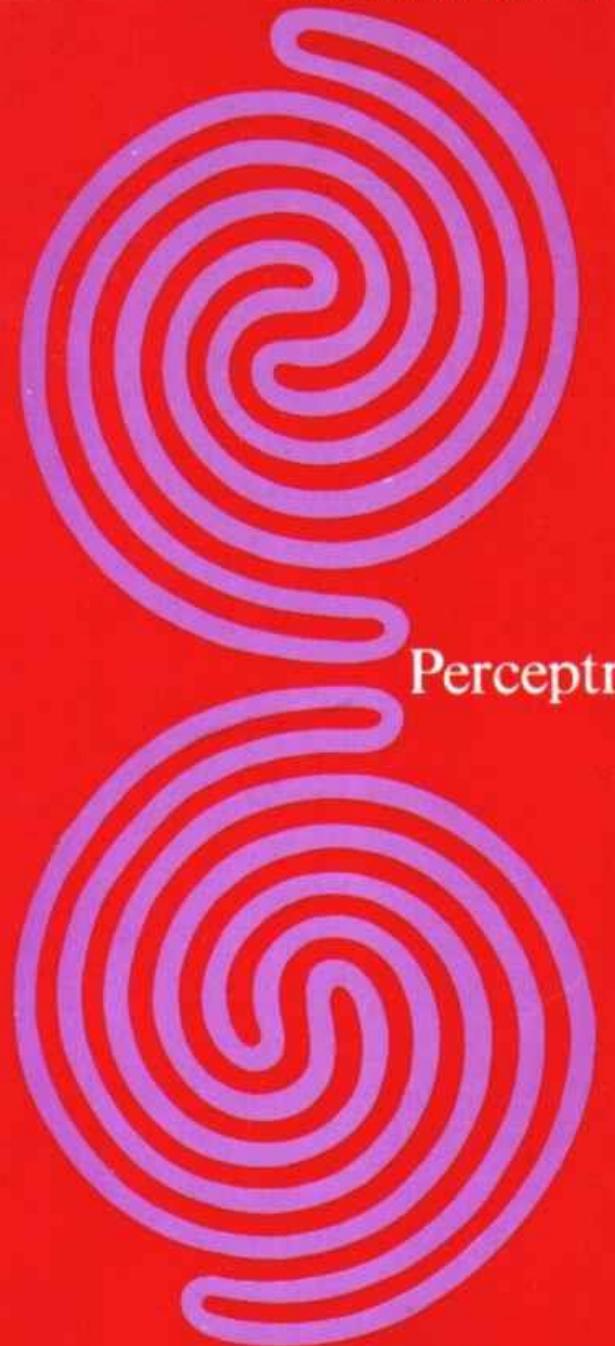
60s: Applications and Structures

- 1960: Widrow and Hoff → LMS & ADALINE
- 1965: MLP

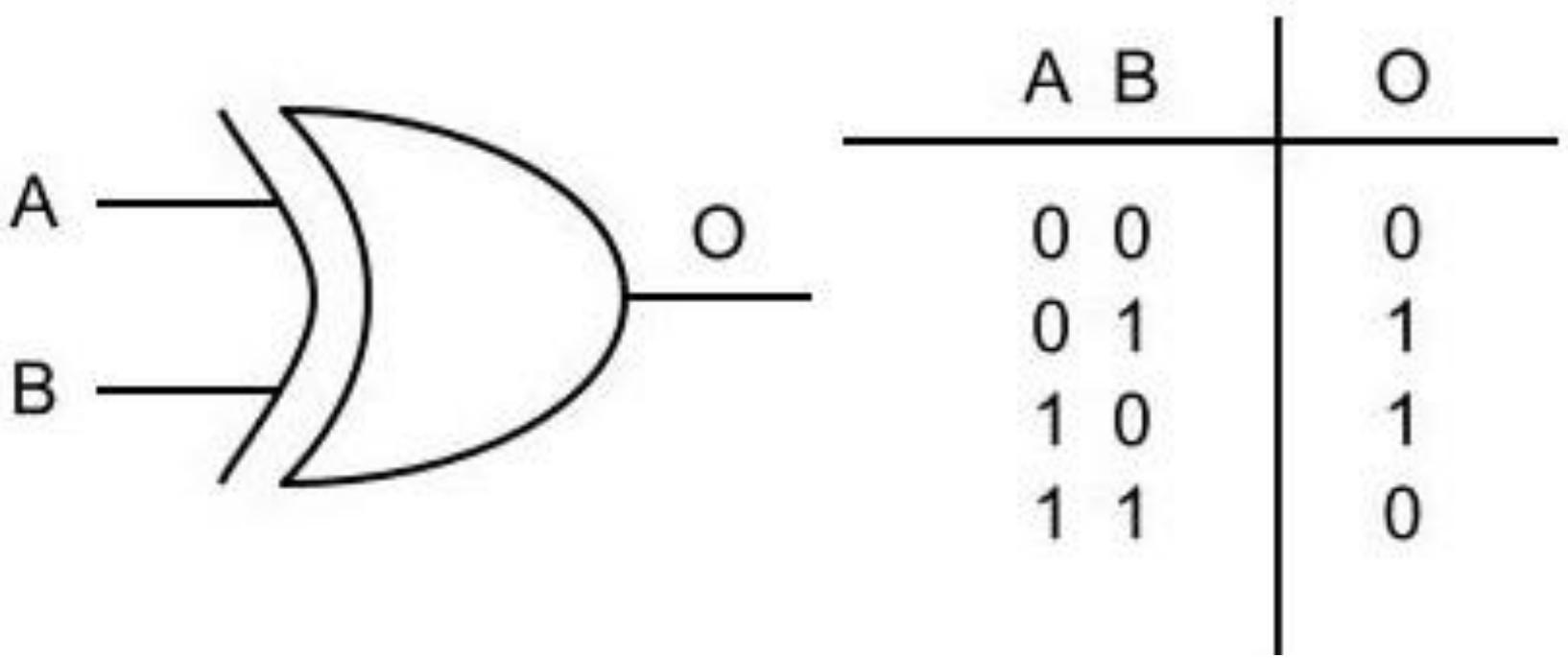


70s

WINTER
IS COMING

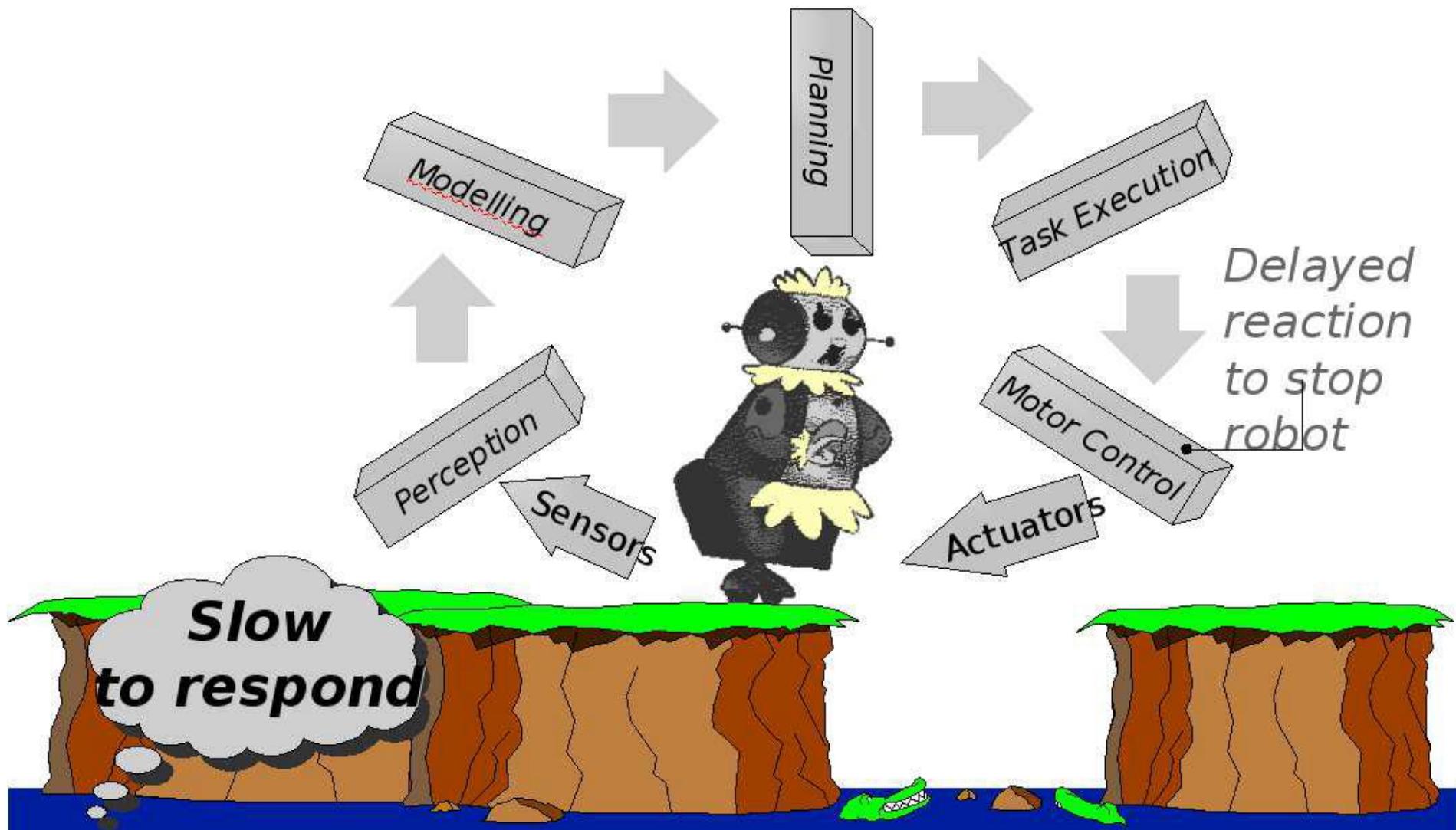


Perceptrons

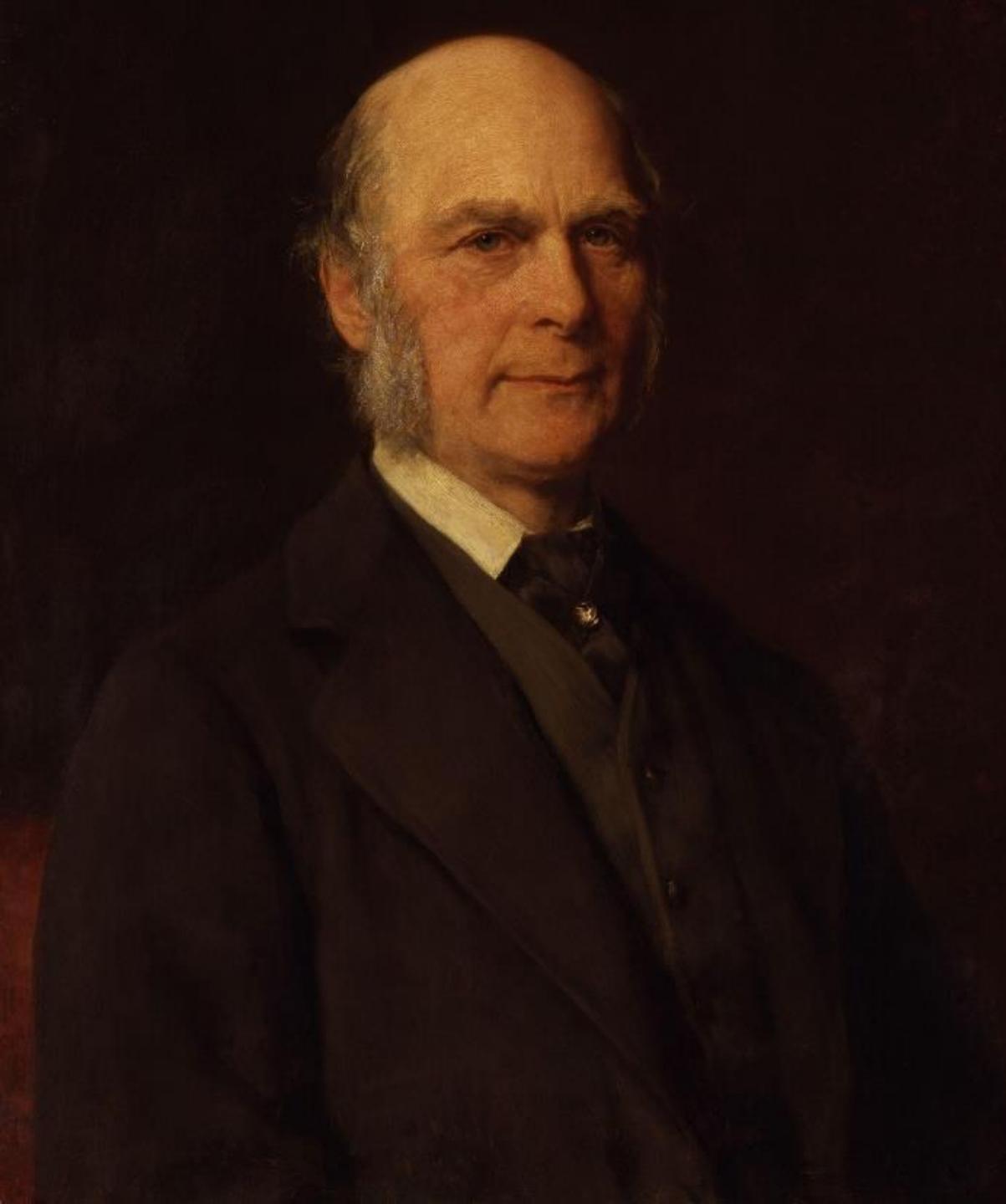


Expert Systems?

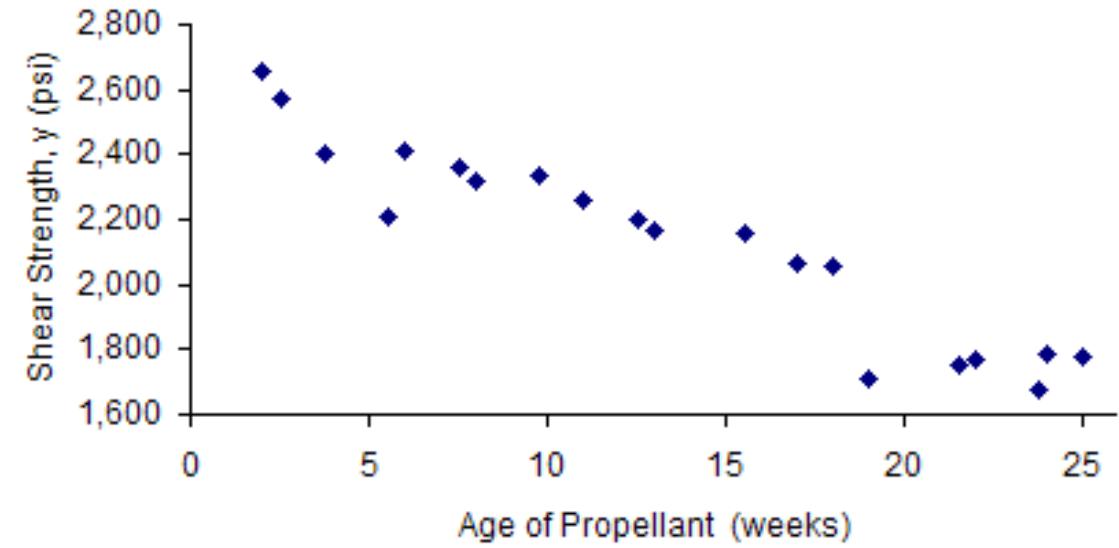




Fitz Patrick (1996)

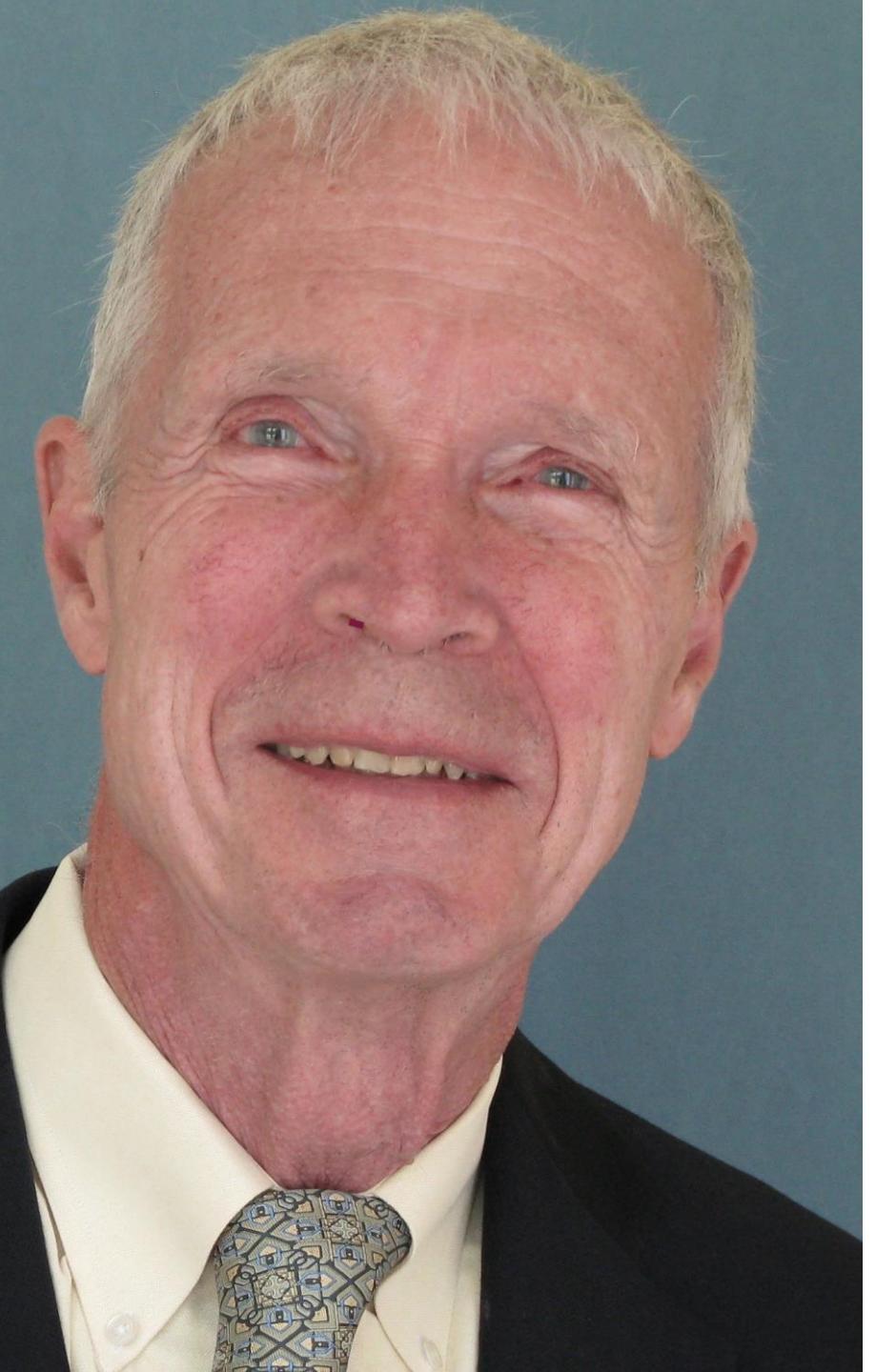
A portrait painting of James Clark Ross, a British naval officer and explorer. He is shown from the chest up, wearing a dark brown naval uniform with a white collar and a gold chain around his neck. His hair is white and receding. He has a serious expression and is looking slightly to the right.

Machine Learning

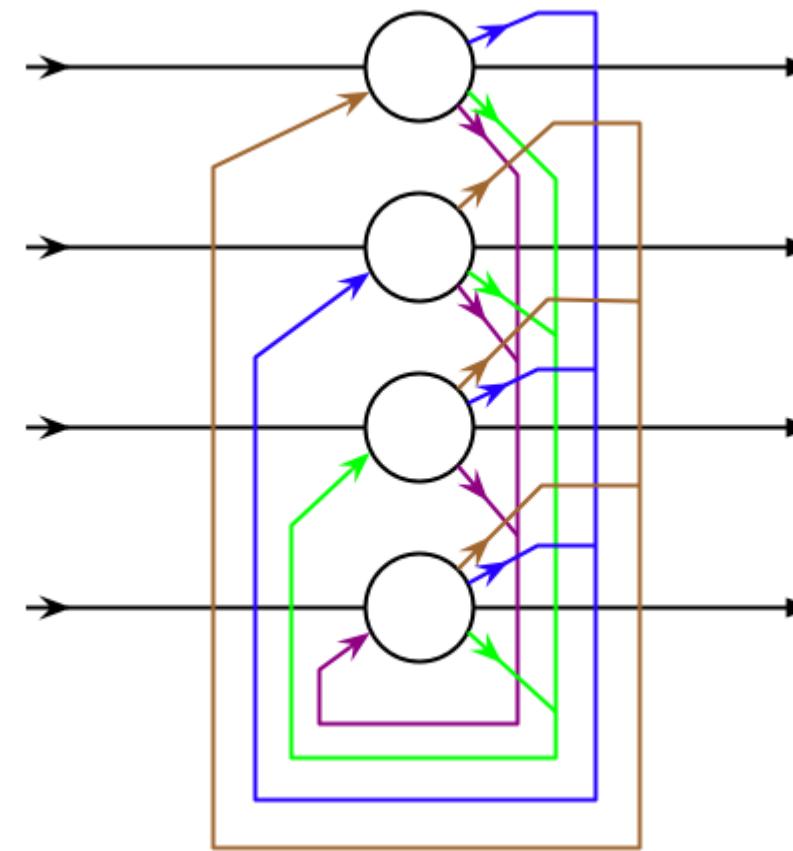




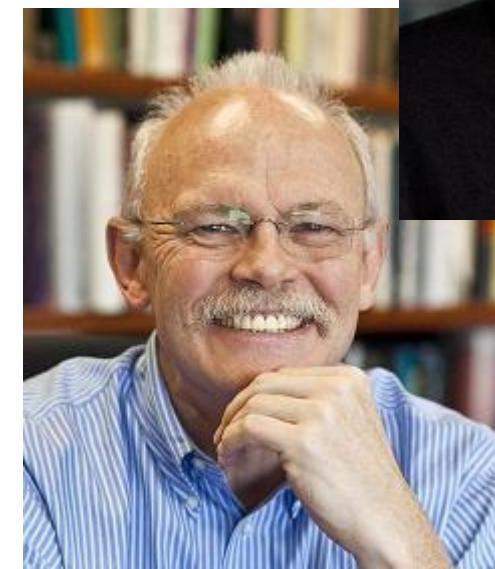
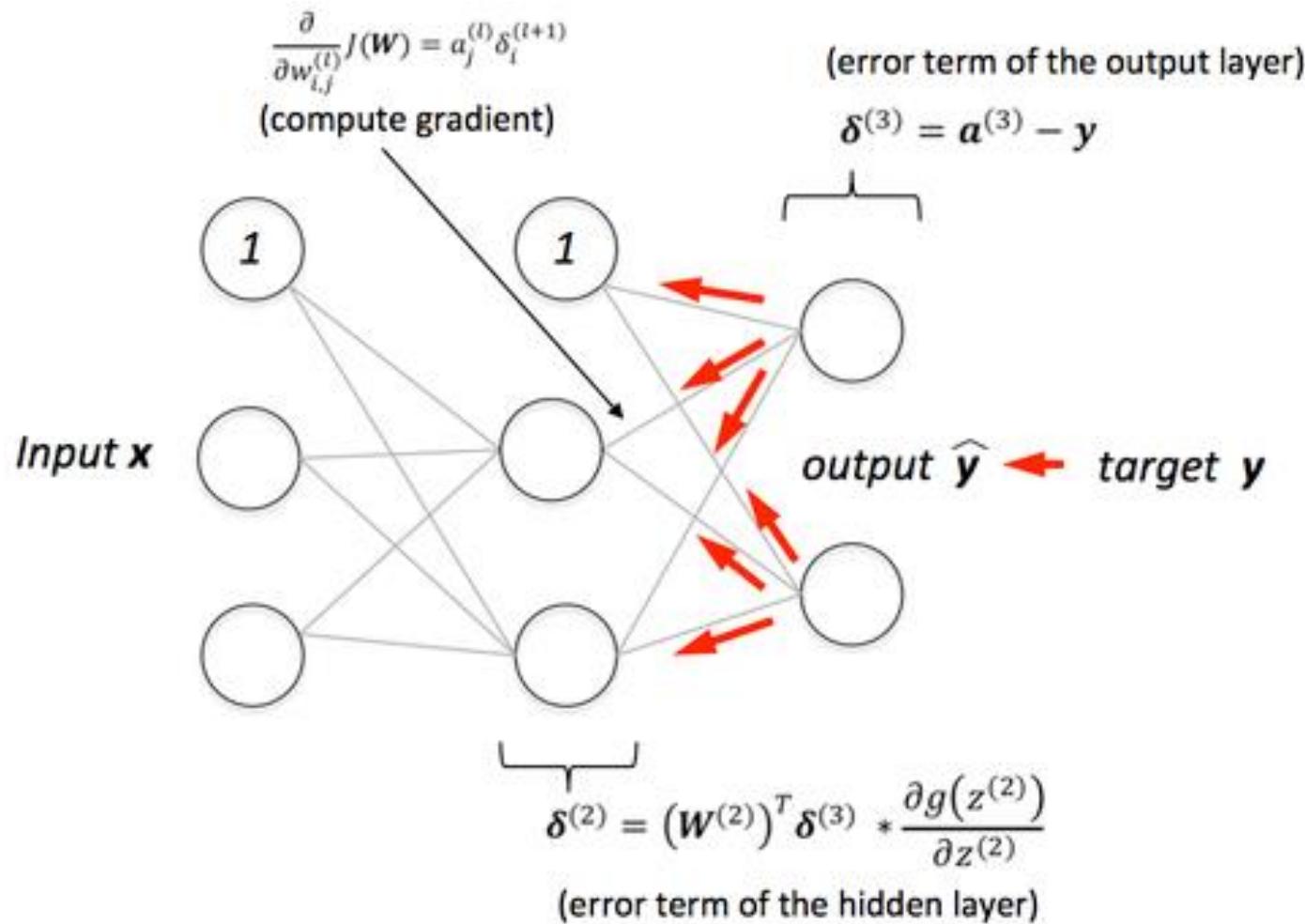
80s: I'll
be back

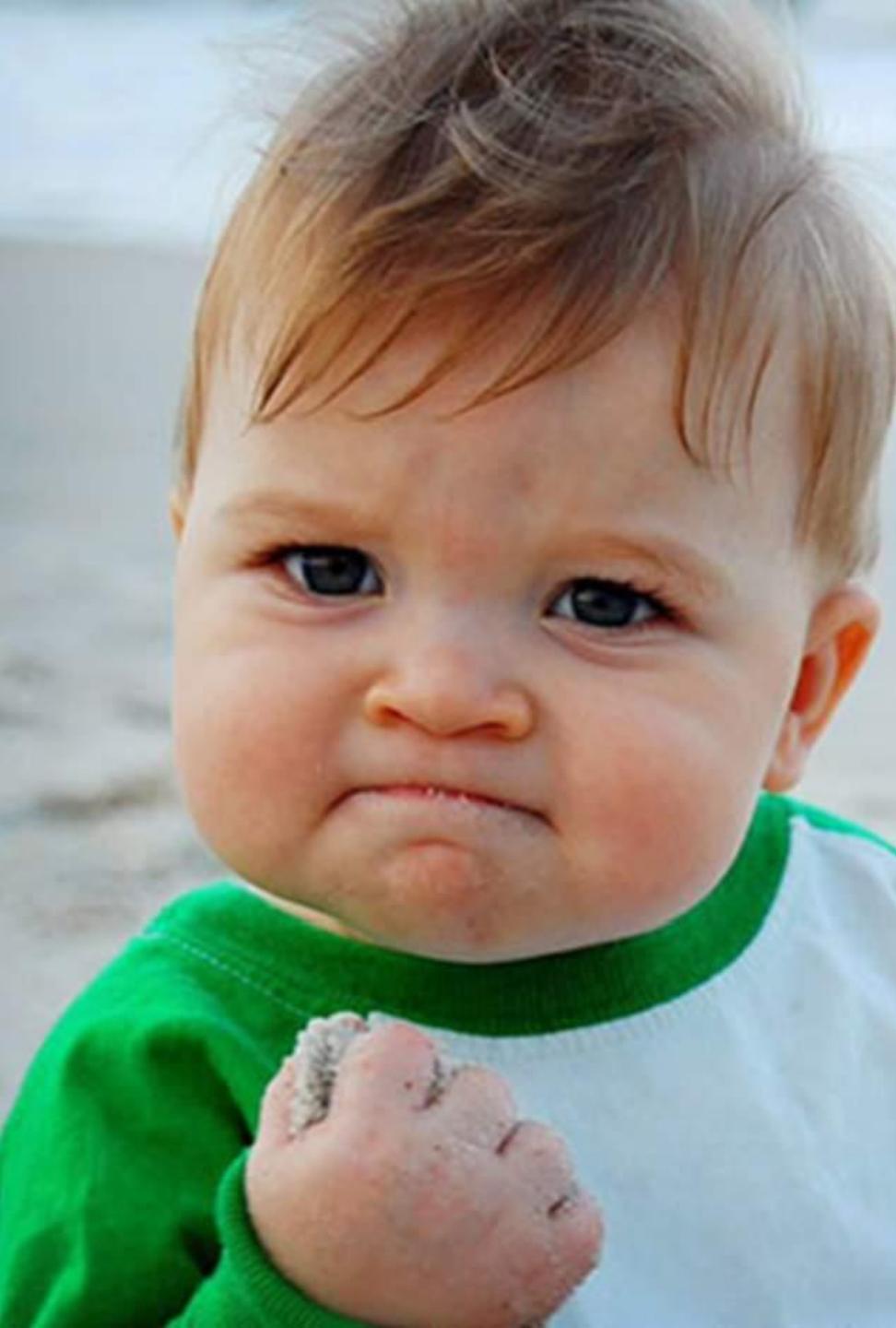


Dynamic structures



Backpropagation





AI Race to Success

- A brave new world:
 - New structures: RBF,SVMs, etc.
 - Application in different sectors:
 - Forecasting
 - Motor control
 - Finance
 - Neuro-Hardware



GARRY
KASPAROV

DEEP
BLUE



\$2,000

ken



\$5,000

WATSON

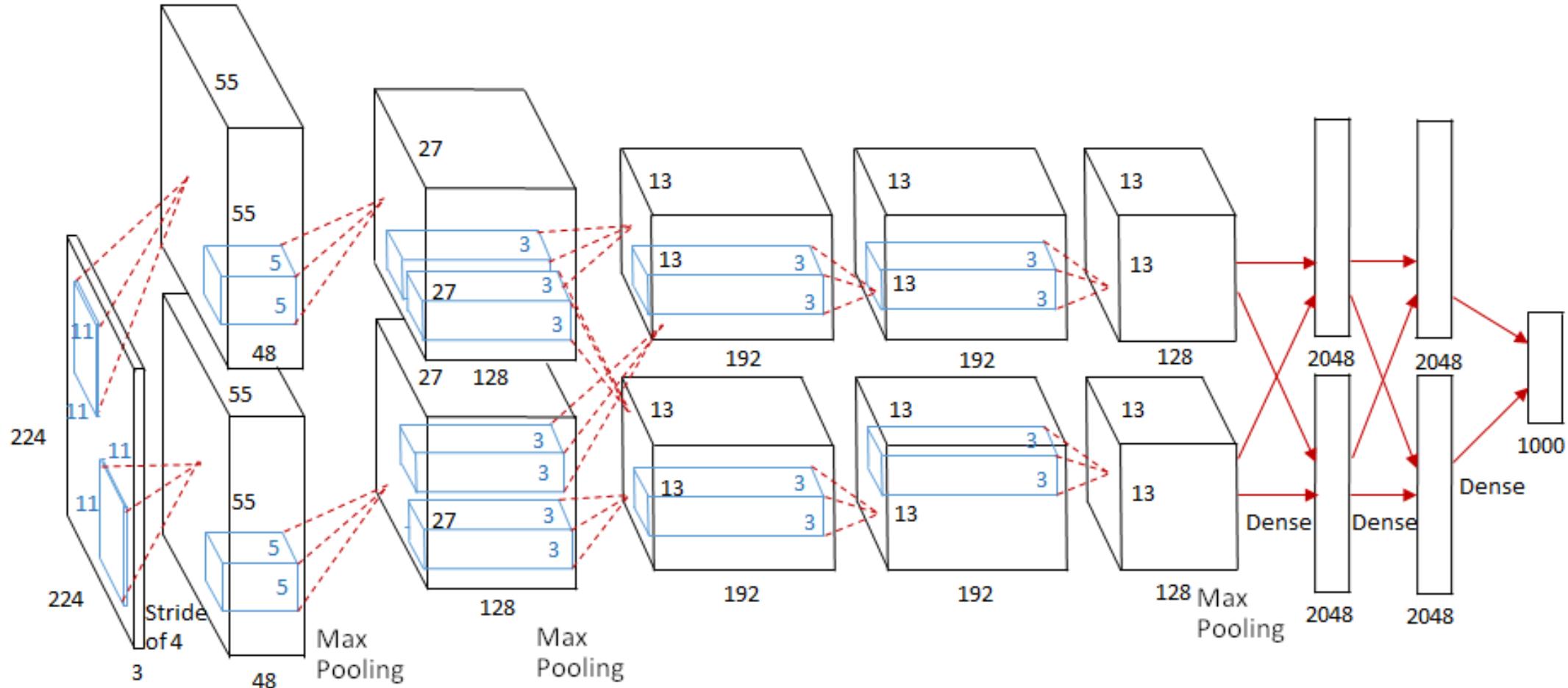


\$5,000

BRAD

DEEP LEARNING

A Machine Learning technique



AlexNet



Google DeepMind Challenge Match

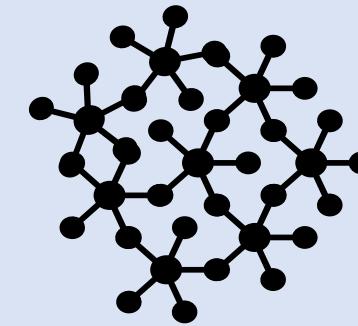
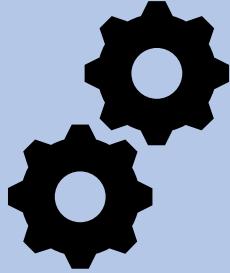
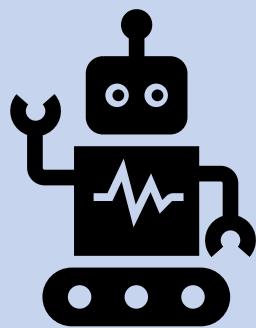
8 - 15 March 2016

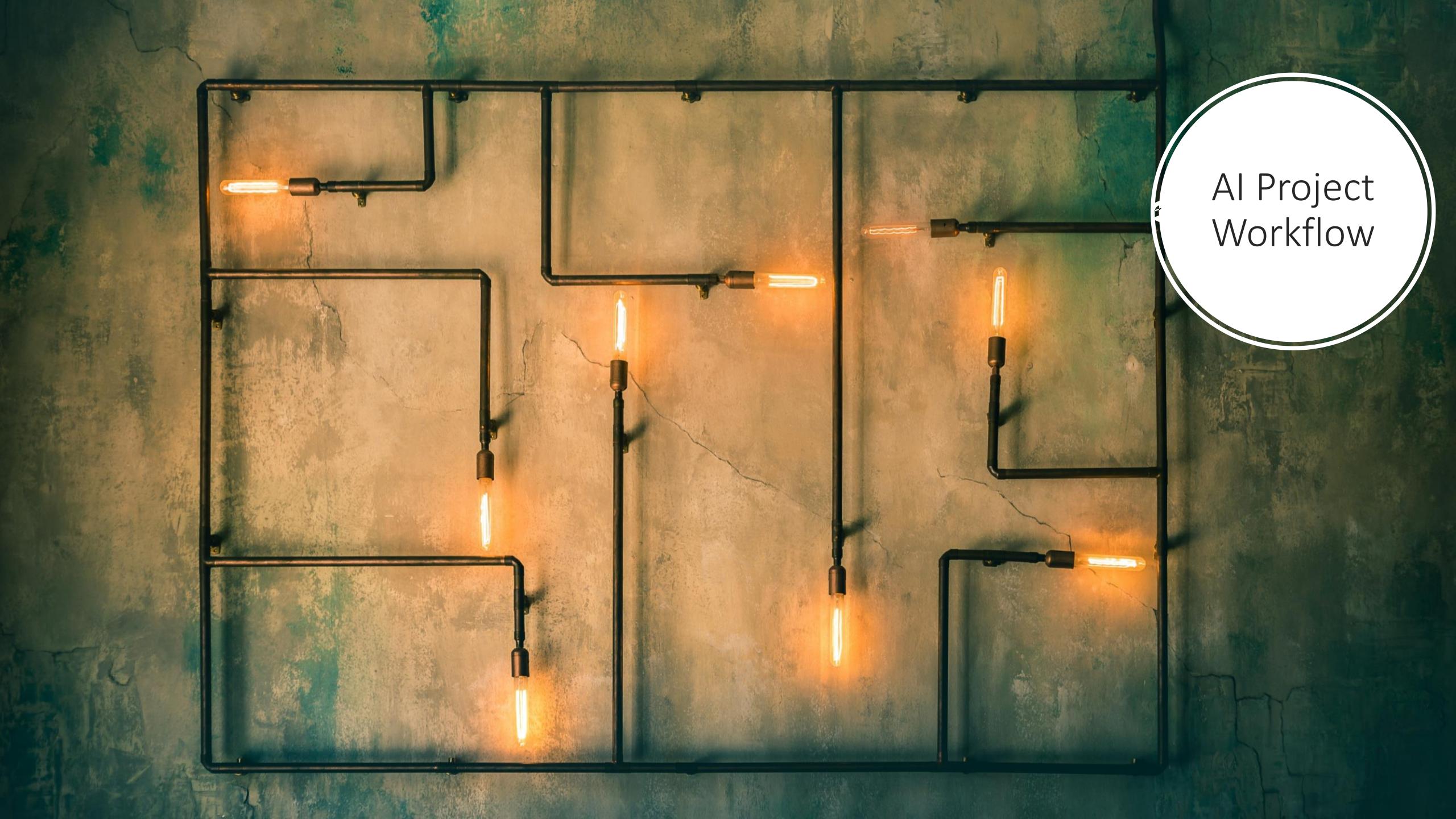


Artificial Intelligence

Machine learning

Deep learning





AI Project
Workflow

Why using it?

Non-linear
solution

Adaptability

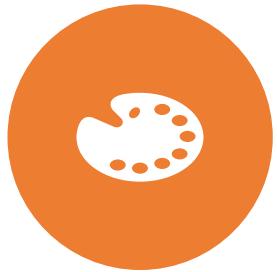
Generalization

Fault tolerance

Task
decomposition

Scalability

Some problems



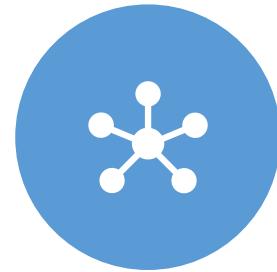
COMPLEX DESIGNS



ITERATIVE PROCESS

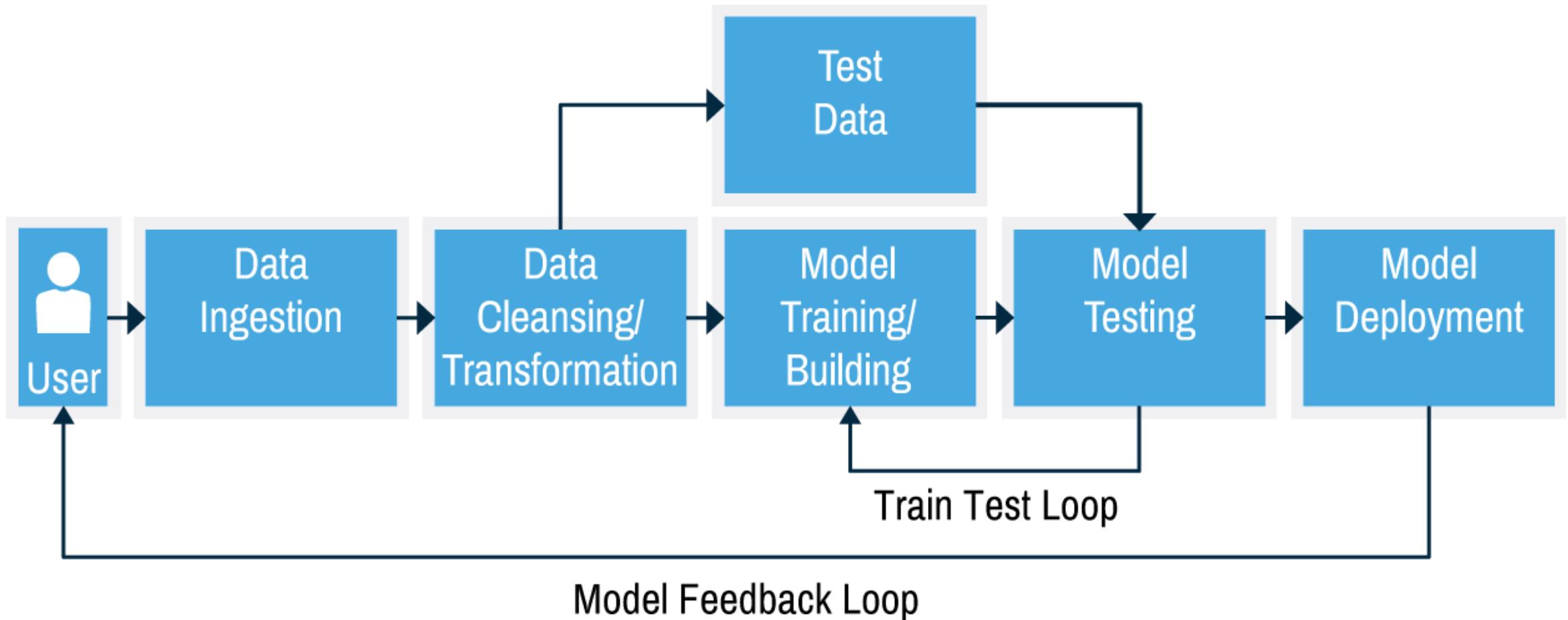


LACK OF A GENERAL
PROCEDURE



LOTS OF PARAMETER
TO ADJUST

Workflow



Gather data

- Different data sources
- The nature of the data is important
- **Which data is really important to build your model?**





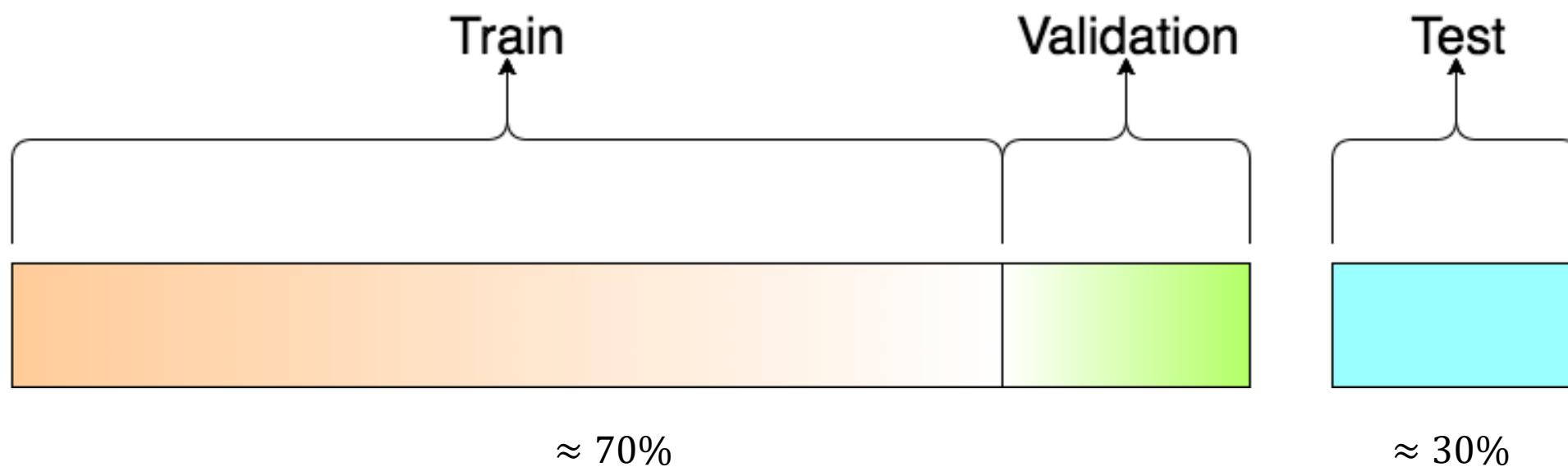
The importance of
data preprocessing

Why preprocessing?

- Cleansing
- Instance selection
- Normalization
- Transformation
- Feature extraction



Split data





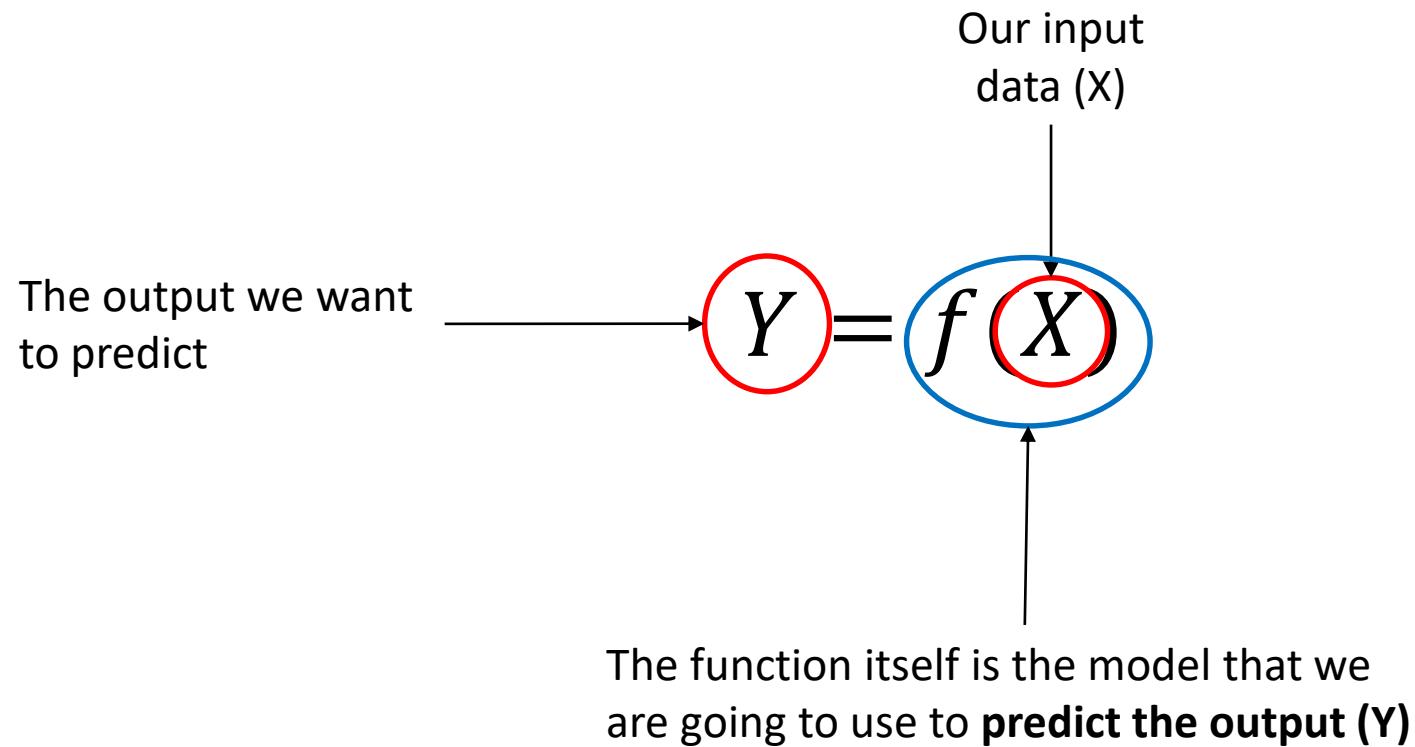
How to train your
model

Hold up wait a minute



Let me put some science in it

What is an AI model?





What do I choose?

- How is my input?
 - Few records, real time data, continuous, discrete, etc.
- What I want to represent?
 - Prediction, approximate my data, classification, etc.
- Which algorithm adjust better to my scenario?
 - RNN, decision tree, SVM, CNN, etc.
- How to train my model?
 - Learning? → Supervised, unsupervised, reinforcement, etc.
 - Tuning? → Genetic algorithms, simulated annealing, etc.

Implement a Model

- Custom Solution → Too many frameworks
- Scikit-learn
 - linear and logistic regressions, SVM, Naive Bayes, gradient boosting, clustering, k-means, etc.
- Deep learning Frameworks



Caffe

Chainer

DL4J
Deeplearning4j

K
KERAS

Microsoft
CNTK

MatConvNet

MINERVA

mxnet

Purine

TensorFlow

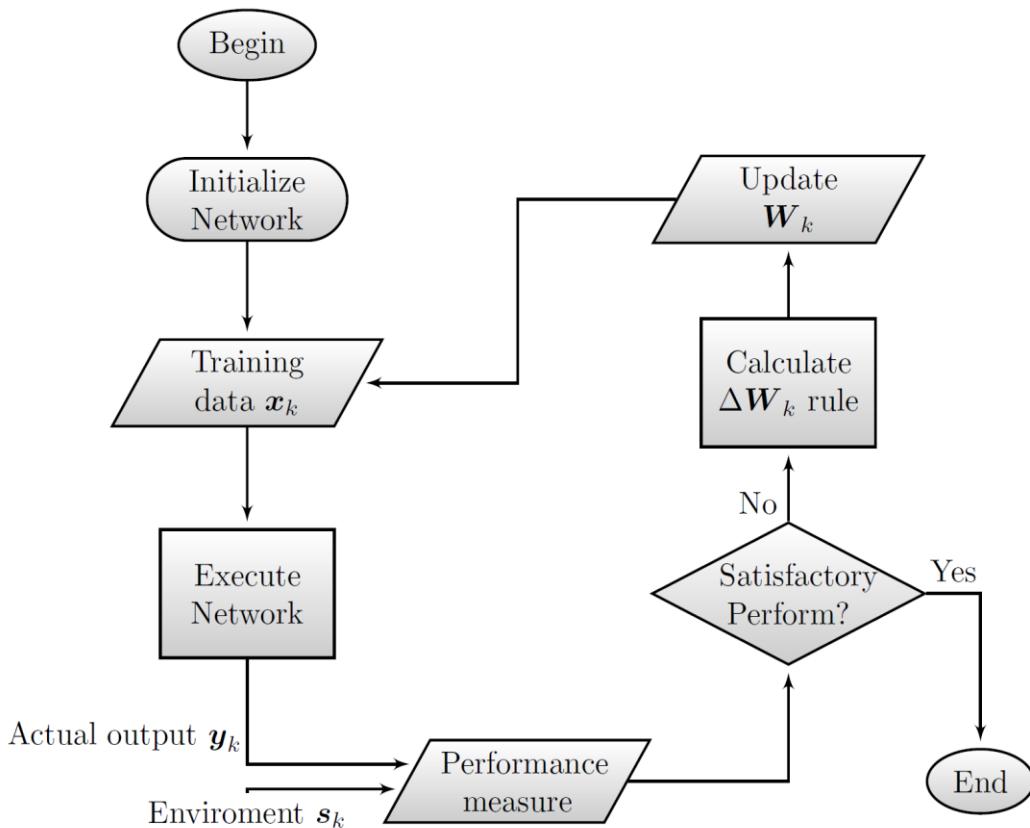
theano

torch



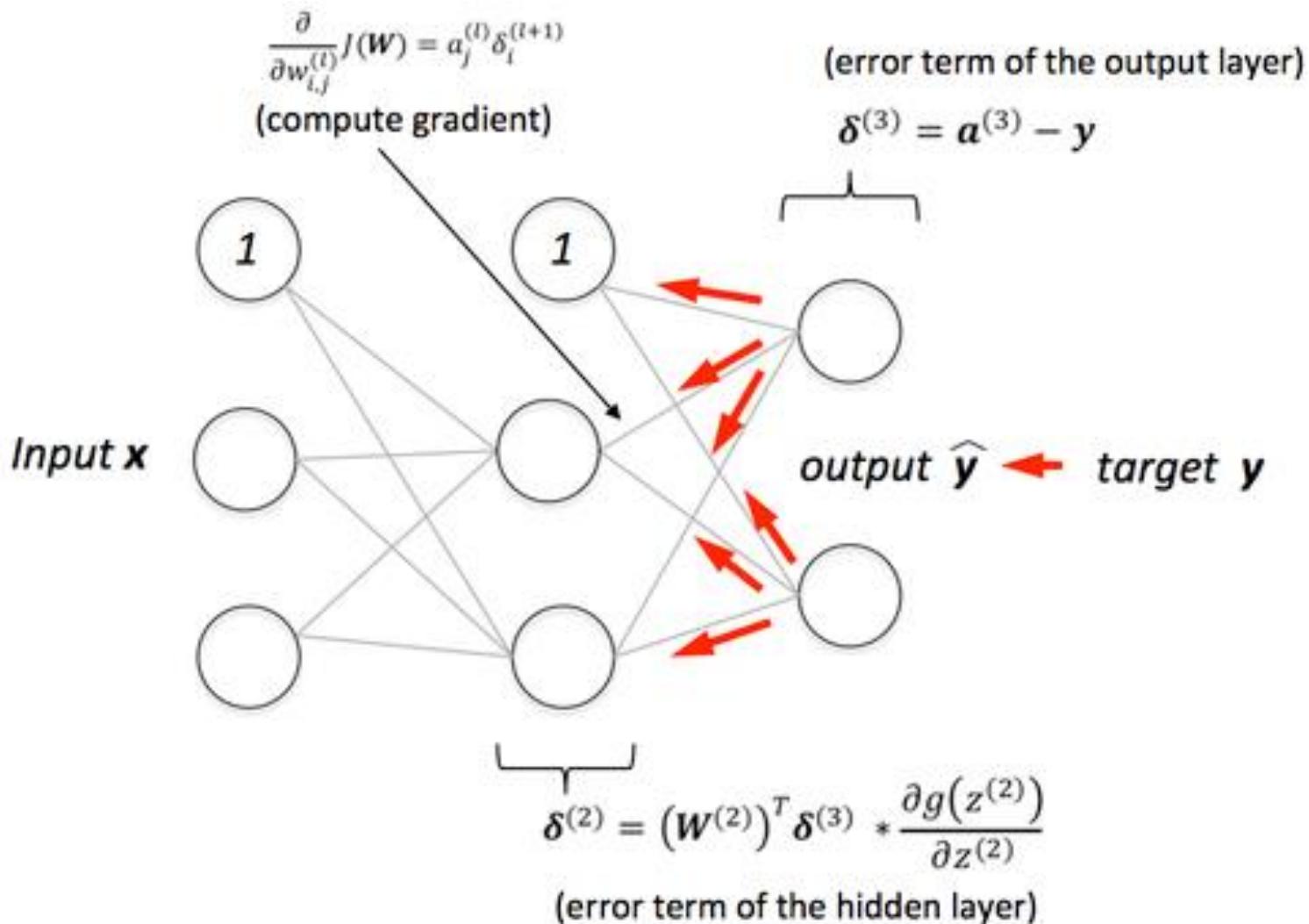
Training a Model

- Learning / Tuning



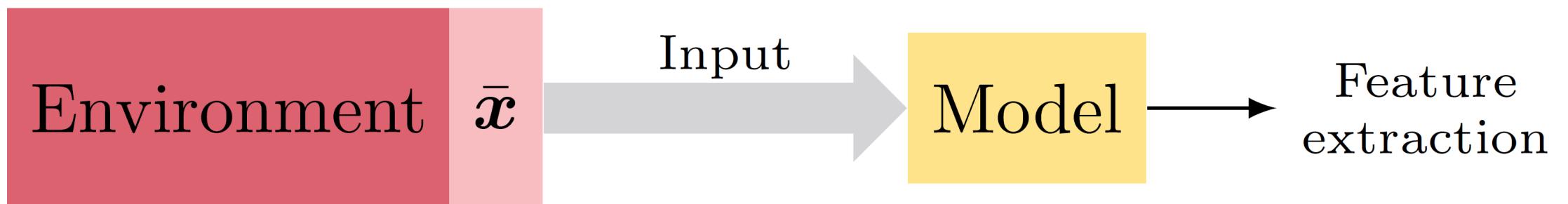
Supervised Learning

- Known:
 - Inputs
 - Target output
- Error is measured
- E.g.
 - LMS
 - Backpropagation



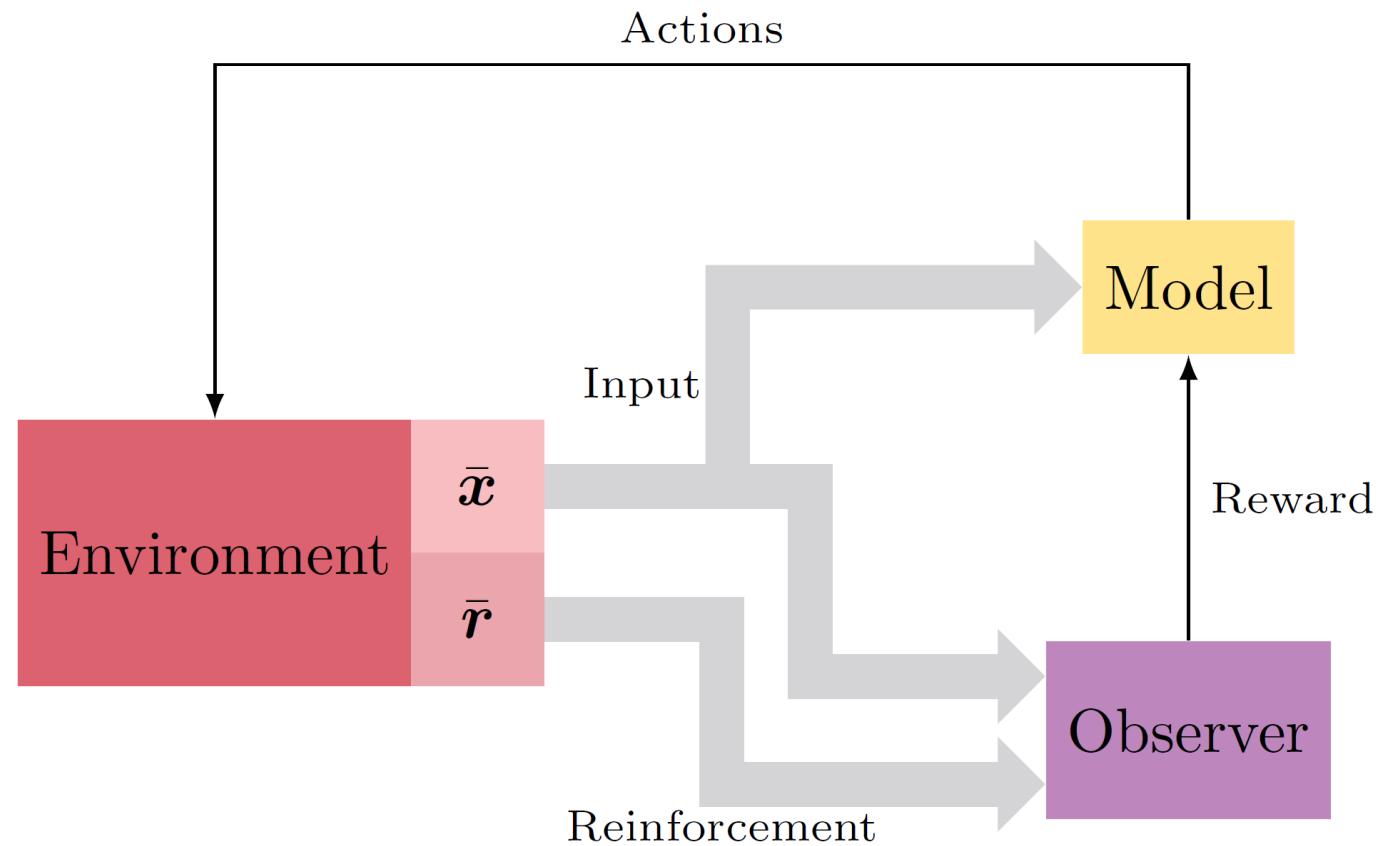
Unsupervised Learning

- Model the probability density of the inputs
- Learn without a teacher
- Known as self-organization
- E.g: Hebb rule $\Delta w_i = \eta \cdot x_i \cdot y_i$



Reinforcement Learning

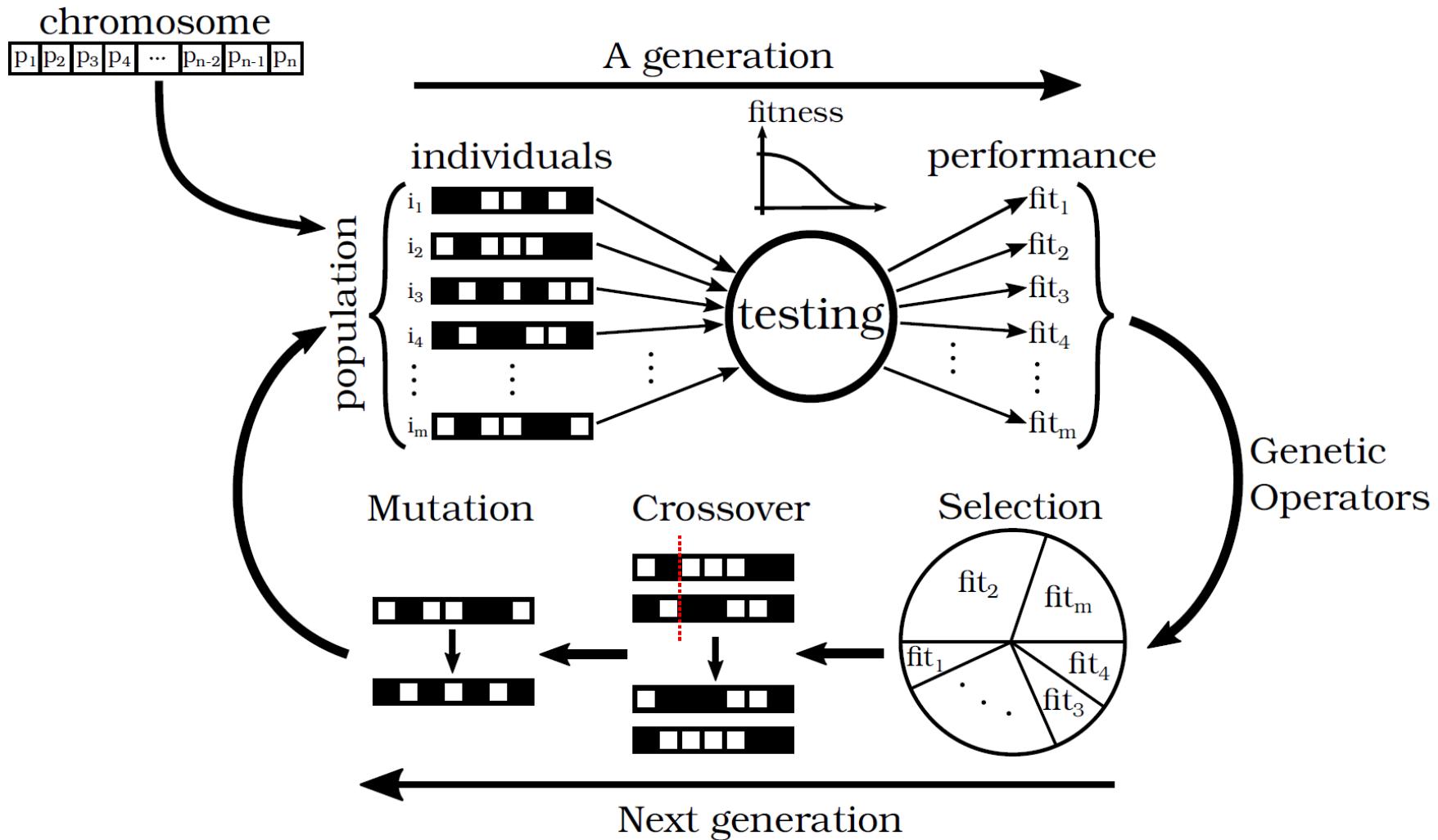
- Train through environment
- Model output influence its behaviour
- Maximize the reward of the environment not a target
 - how right or wrong



Reinforcement Learning



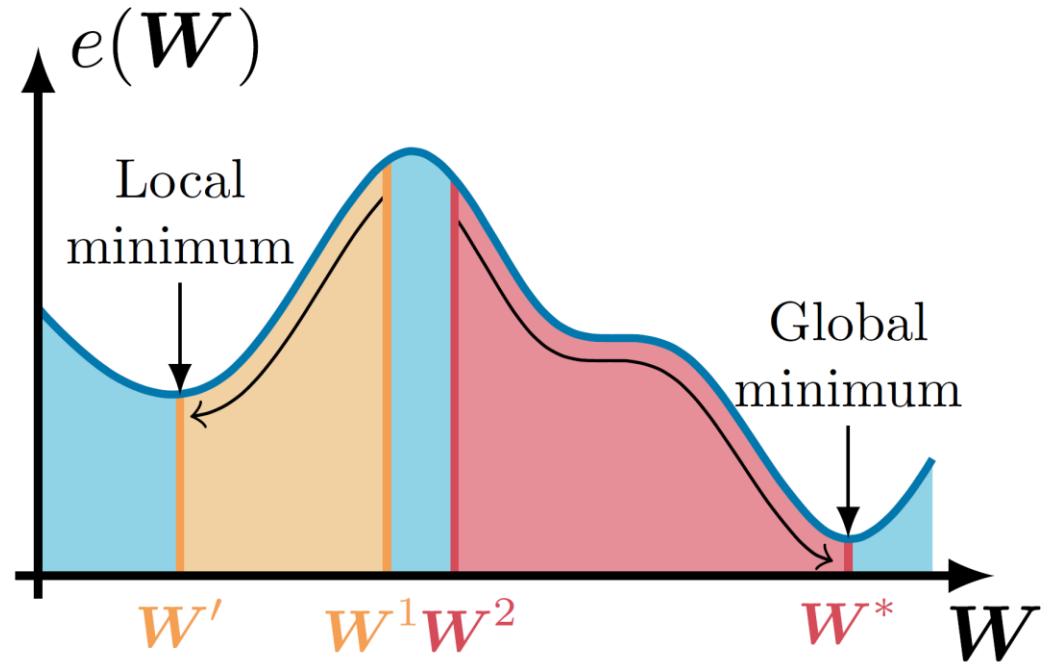
Tuning: GA

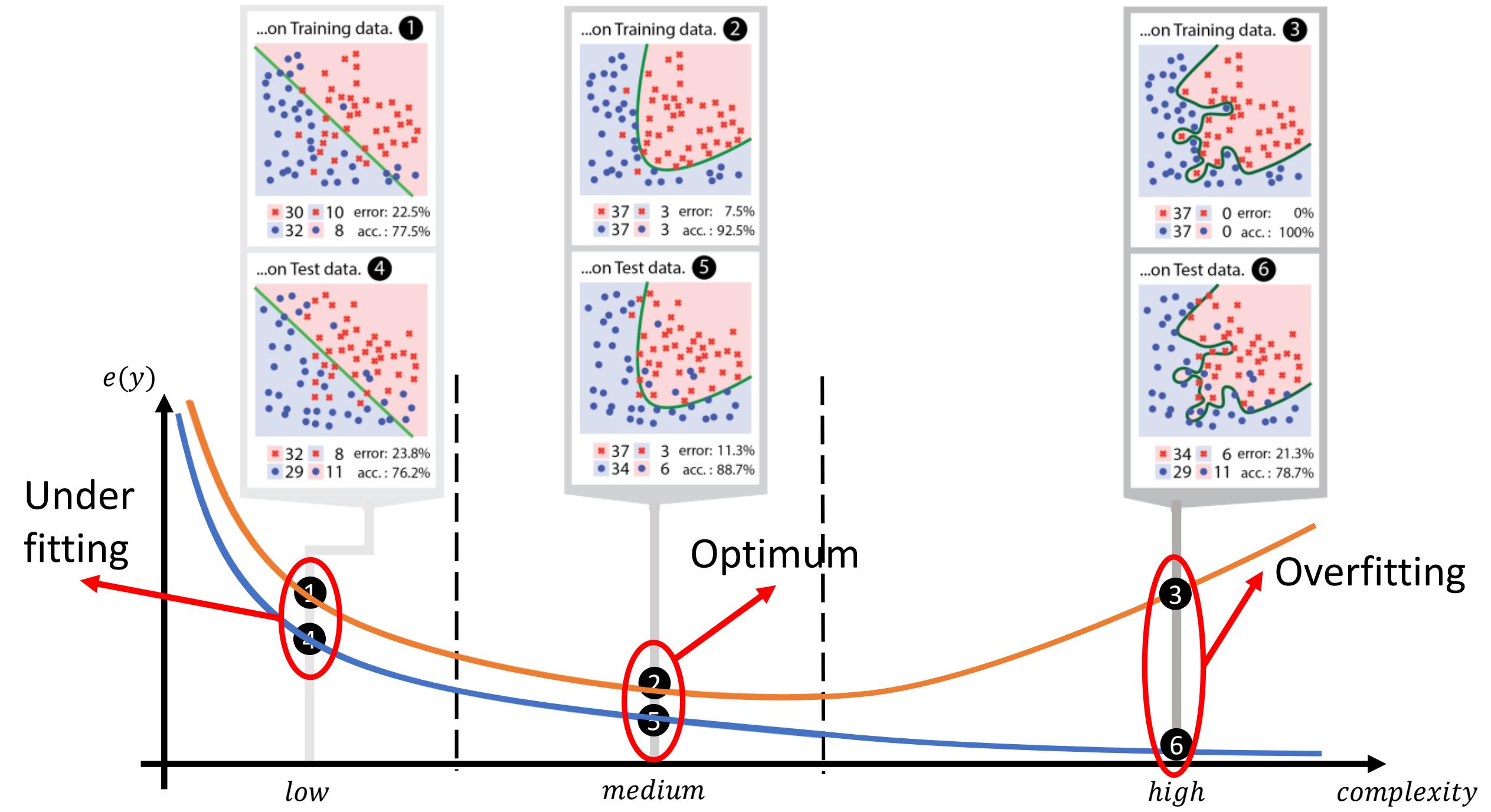


When stop training

- Found the minimum error

Watch out!!





Performance evaluation

Precision

Recall

Accuracy

F1 Score

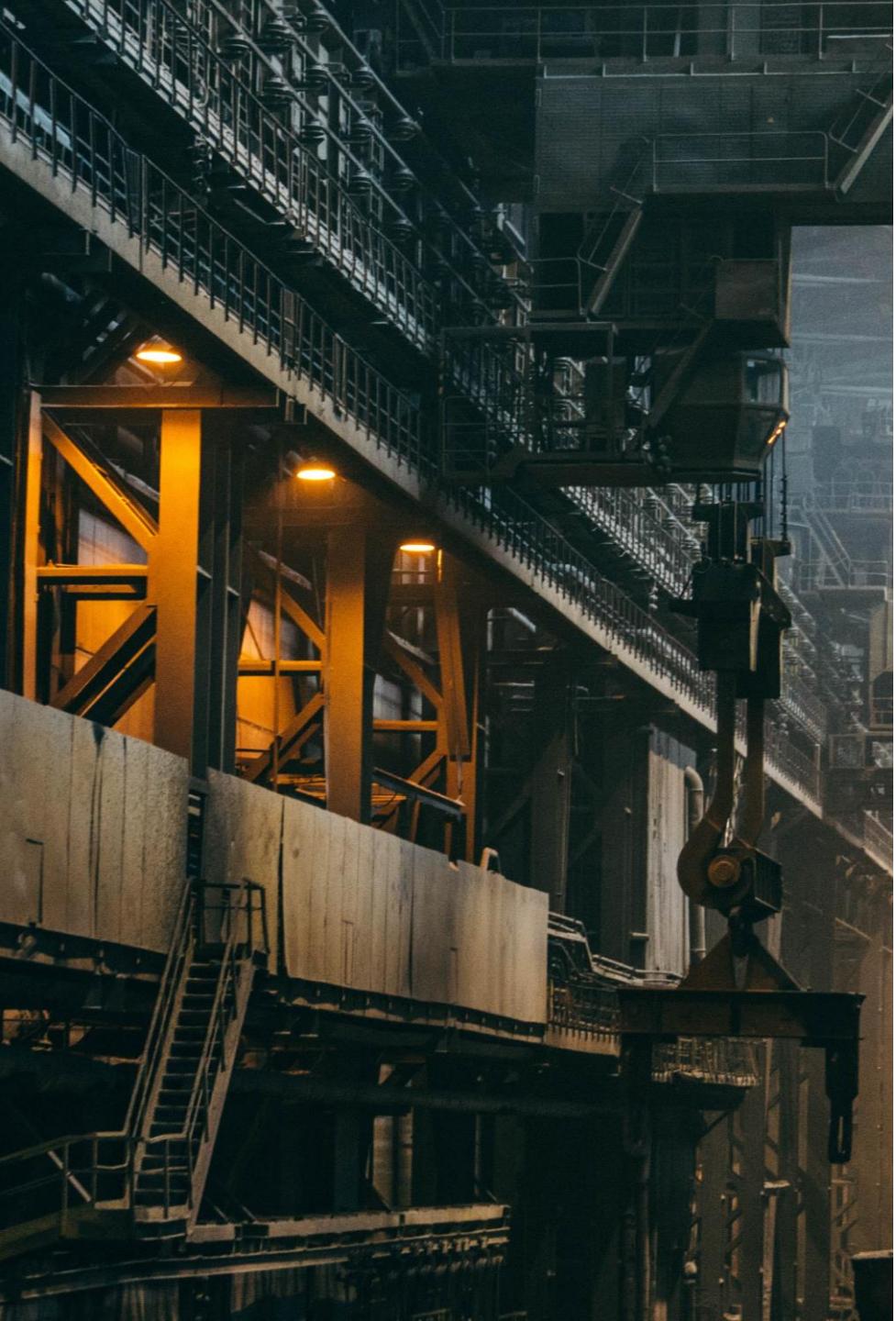
$$\frac{TP}{TP + FP}$$

$$\frac{TP}{TP + FN}$$

$$\frac{TP + TN}{P + N}$$

$$2 \cdot \frac{P \cdot R}{P + R}$$

There are lots more metrics that you can use!!



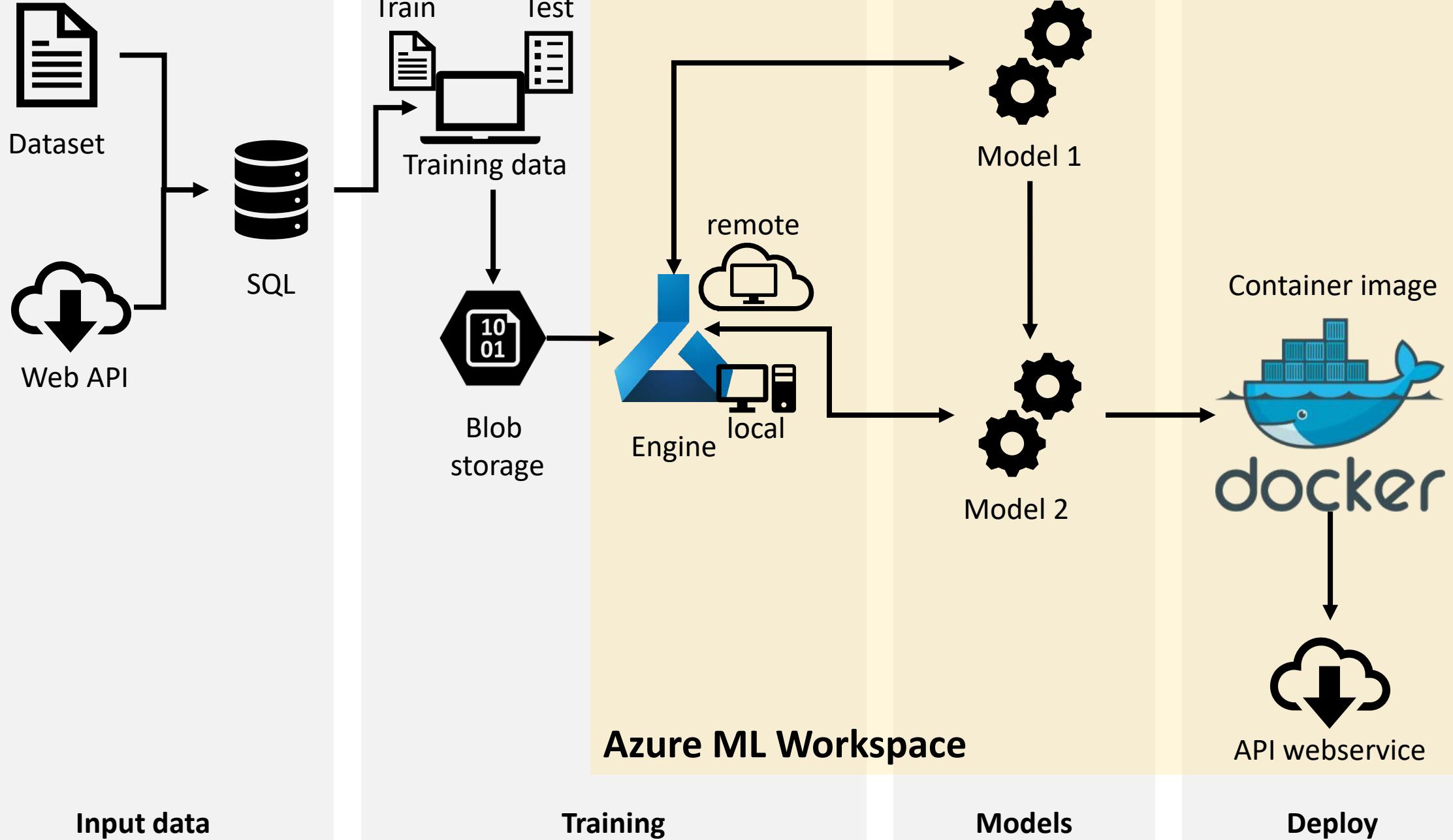
Production



Azure Machine Learning



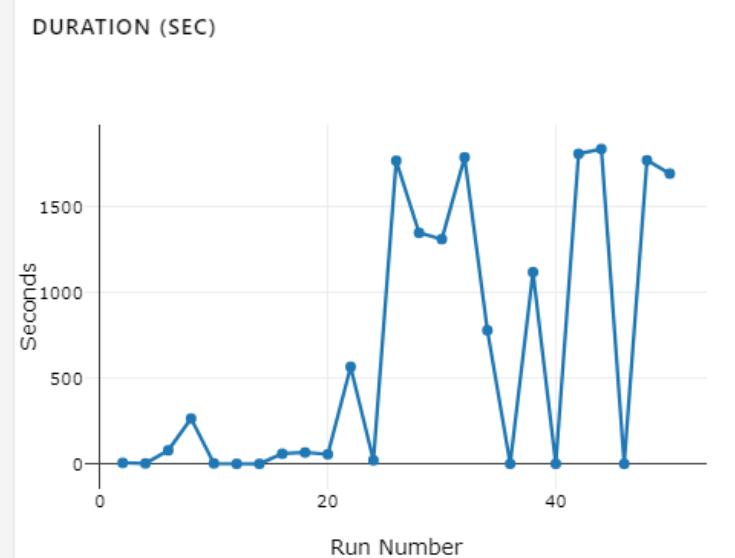
Amazon SageMaker



classifier | All Runs Report

[+ Add Chart](#) [Filter](#) [Get Link](#) [Refresh](#)

PREPARING	RUNNING
0	0
COMPLETED	FAILED
7	18



RUN NUMBER ↑↓	STATUS ↑↓	CREATED TIME (UTC) ↑↓	DURATION (SECONDS) ↑↓	ACCURACY ↑↓	TAGS ↑↓
42	Completed	10/30/2018, 12:21:37 PM	1810.439		
44	Failed	10/31/2018, 2:40:02 PM	1838.426		
46	Failed	10/31/2018, 3:40:15 PM	0.127		
48	Completed	10/31/2018, 3:42:05 PM	1772.63		
50	Completed	11/7/2018, 12:24:12 PM	1694.762		

ATTRIBUTES	
Status	Completed
Created	11/6/2018, 4:00:00 PM
Duration	0:25:43
Target	devindicationsvm
Run Id	embedding_1541516393251
Run Number	70
Script Name	train.py

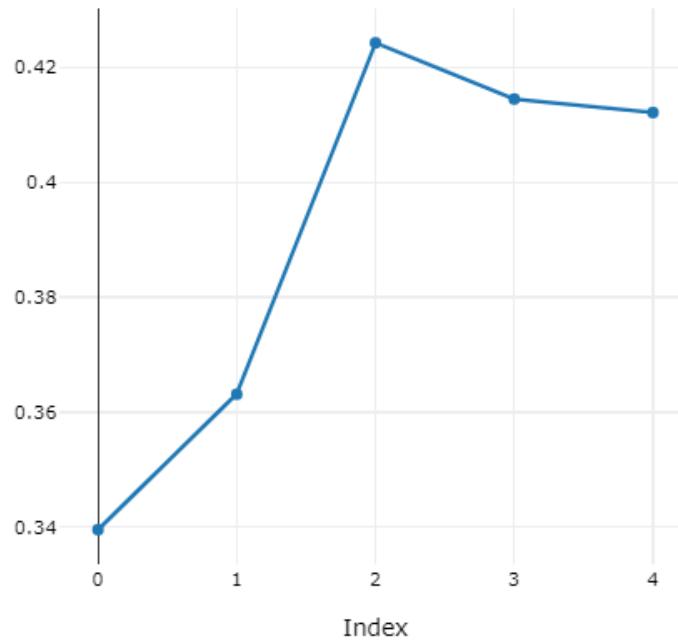
TRACKED METRICS	
Learning Rate	0.05
Epochs	5
Embedding Type	skipgram
Dimension	300
Pretrained Vectors	wiki.es.vec

Training Metrics

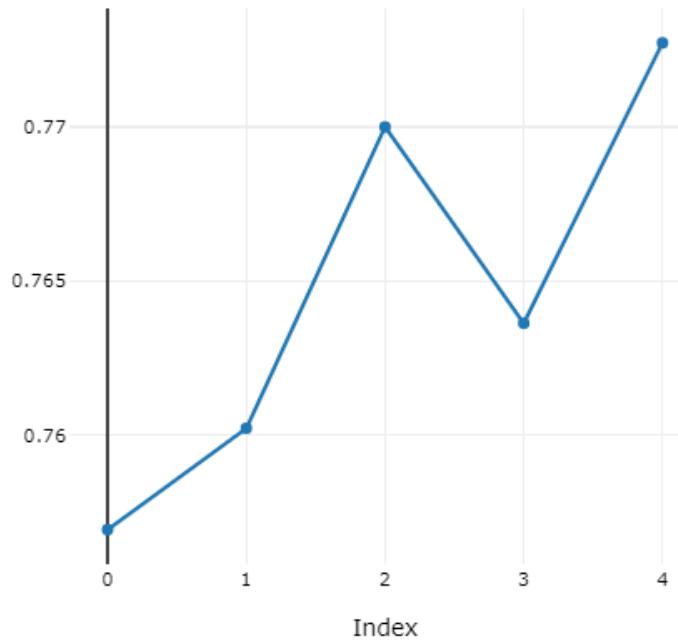
Performance

CHARTS

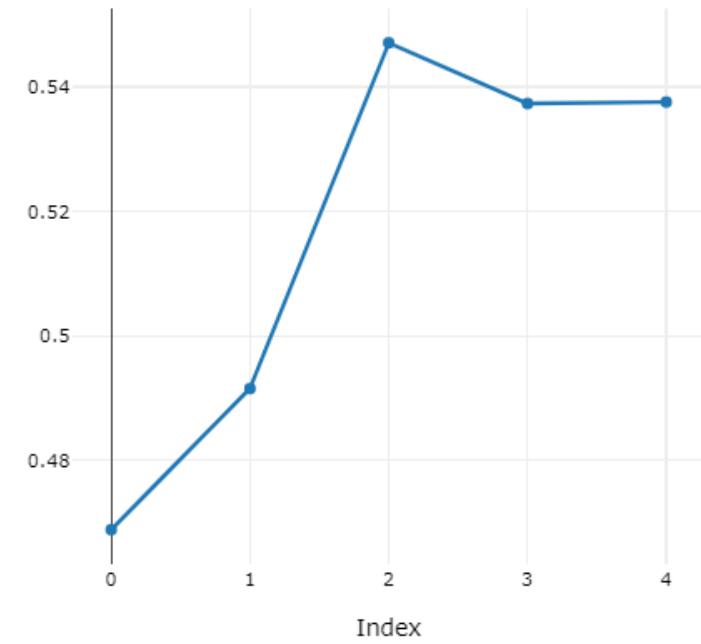
Recall



Precision



F1 score





- Open source platform
- Integrated within Databricks
- Can publish to different flavours
 - Azure machine learning
 - Amazon Sagemaker
 - H₂O.ai
 - Local Python function
 - Etc.



MLflow

127.0.0.1:5000/#/ Jules

mlflow GitHub Docs

Experiments

Default

Default

Experiment ID: 0

Artifact Location: /Users/jules/jsd-mlflow-examples/keras/binary_classifier_nn/mlruns/0

Search Runs:

metrics.rmse < 1 and params.model = "tree"

Search

Filter Params:

alpha, lr

Filter Metrics:

rmse, r2

Clear

9 matching runs

Compare Selected

Download CSV

Date	User	Source	Version	Parameters				Metrics					
				epochs	hidden_layers	loss_function	output	average_acc	average_loss	binary_acc	binary_loss	validation_acc	validation_loss
2018-08-13 15:13:54	jules	main_nn.py	abod1f	20	3	mse	32	0.878	0.09	0.977	0.025	0.885	0.025
2018-08-13 14:34:43	jules	main_nn.py	abod1f	30	3	binary_crossentropy	32	0.866	0.441	0.992	0.035	0.879	0.035
2018-08-13 09:12:03	jules	main_nn.py	abod1f	20	1	binary_crossentropy	16	0.883	0.304	0.937	0.212	0.89	0.212
2018-08-10 15:11:24	jules	main_nn.py	abod1f	20	1	binary_crossentropy	16	0.883	0.304	0.937	0.212	0.89	0.212
2018-08-10 15:09:02	jules	main_nn.py	abod1f	20	1	binary_crossentropy	16	0.883	0.304	0.937	0.212	0.89	0.212
2018-08-10 14:41:29	jules	main_nn.py	abod1f	20	1	binary_crossentropy	16	0.883	0.304	0.937	0.212	0.89	0.212
2018-08-10 14:39:25	jules	main_nn.py	abod1f	20	1	binary_crossentropy	16	0.883	0.304	0.937	0.212	0.89	0.212
2018-08-09 14:54:40	jules	main_nn.py	e3c9ae	15	3	mse	32	0.881	0.089	0.96	0.04	0.887	0.04
2018-08-09 14:53:49	jules	main_nn.py	e3c9ae	20	1	binary_crossentropy	16	0.883	0.304	0.937	0.212	0.89	0.212



Applications

Image

- Image Classification
- Object Detection
- Synthetical Generation of Images

Sound

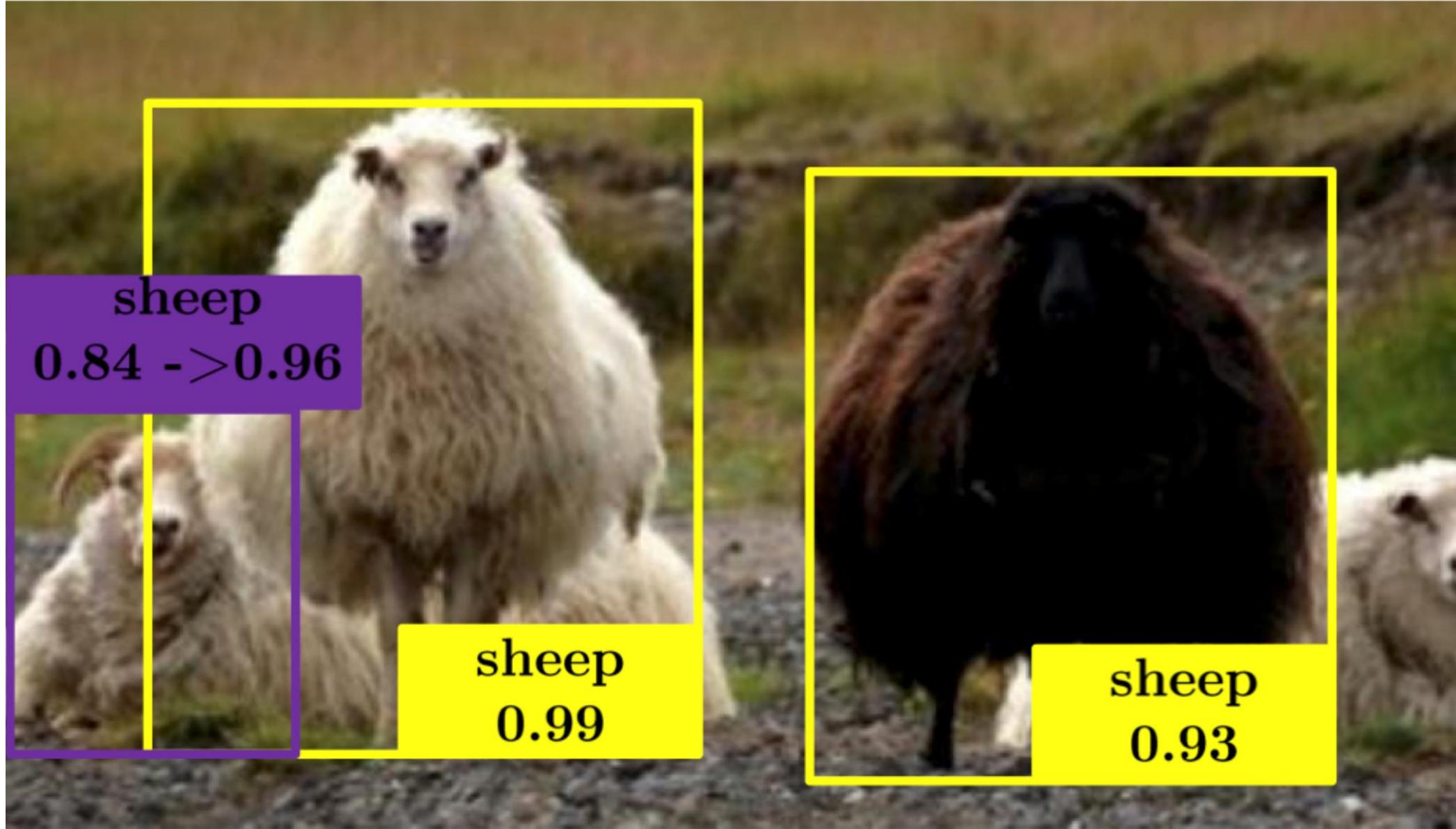
- Fraud Detection
- Defect Enhancement
- Style Transfer

Text

- Information Retrieval
- Knowledge Extraction
- Sentiment Analysis
- Style Transfer

Signal Analysis

- Time Series Analysis



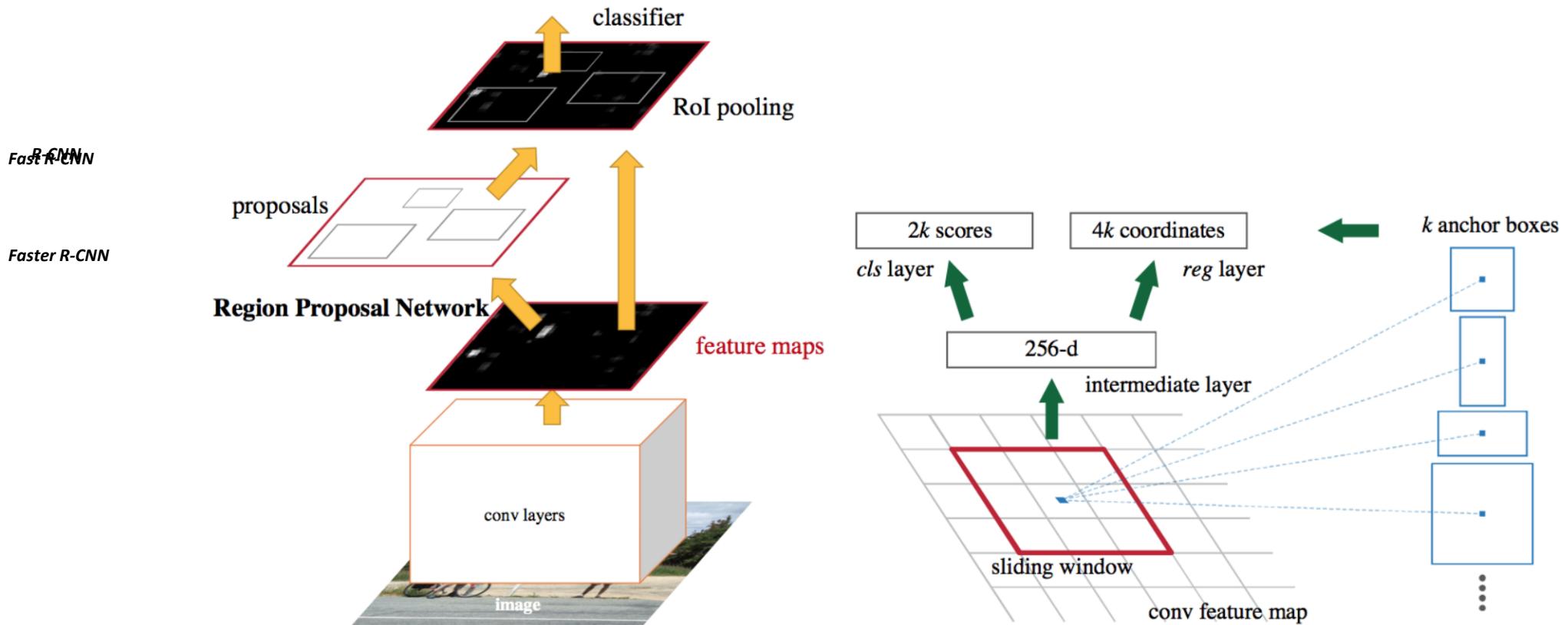






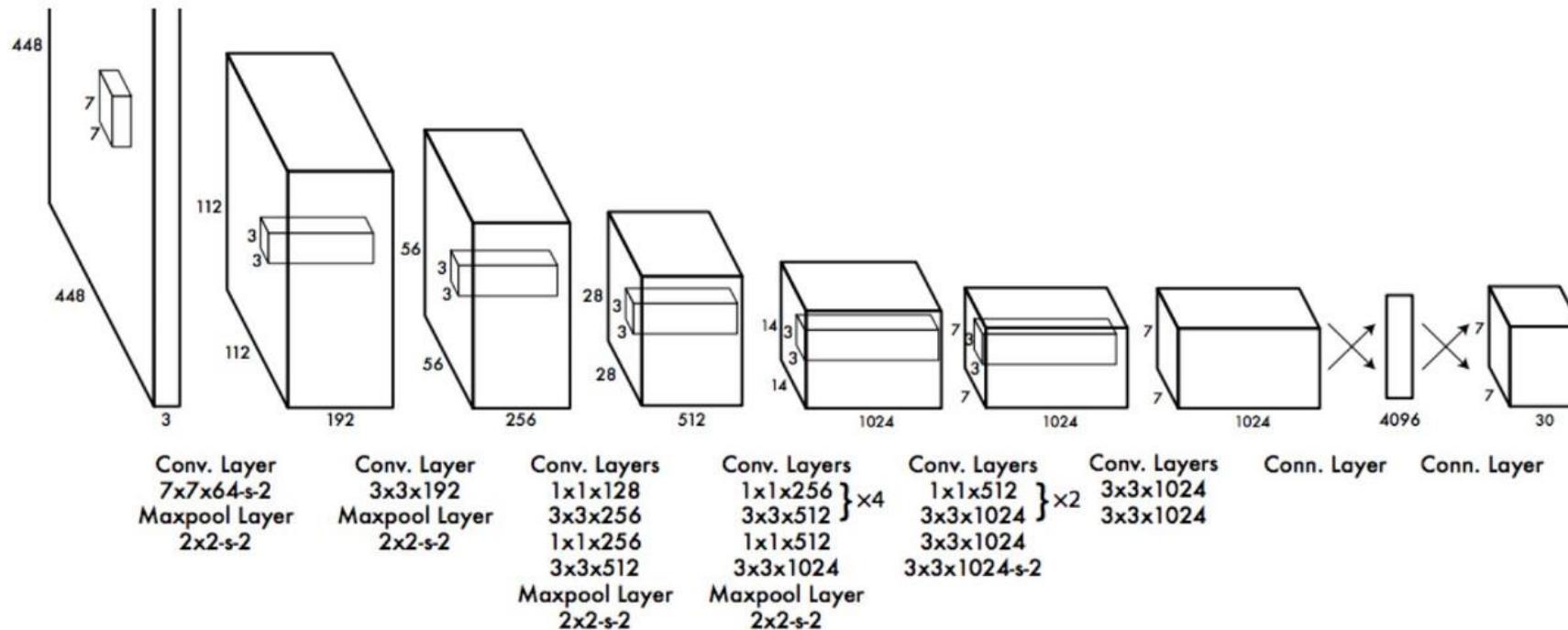
Detección de Objetos

- Métodos detección de objetos



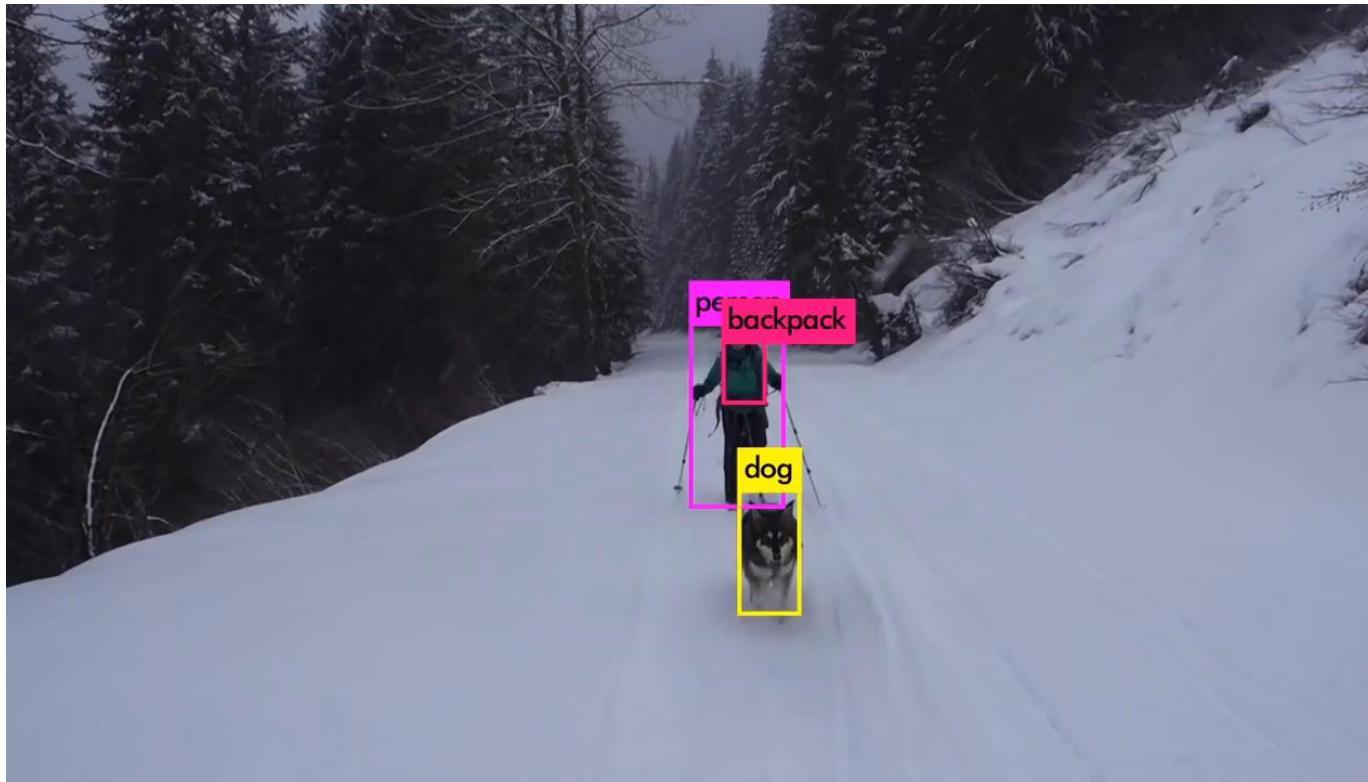
YOLO (You Only Look Once)

- Enfoque muy distinto para la detección de objetos



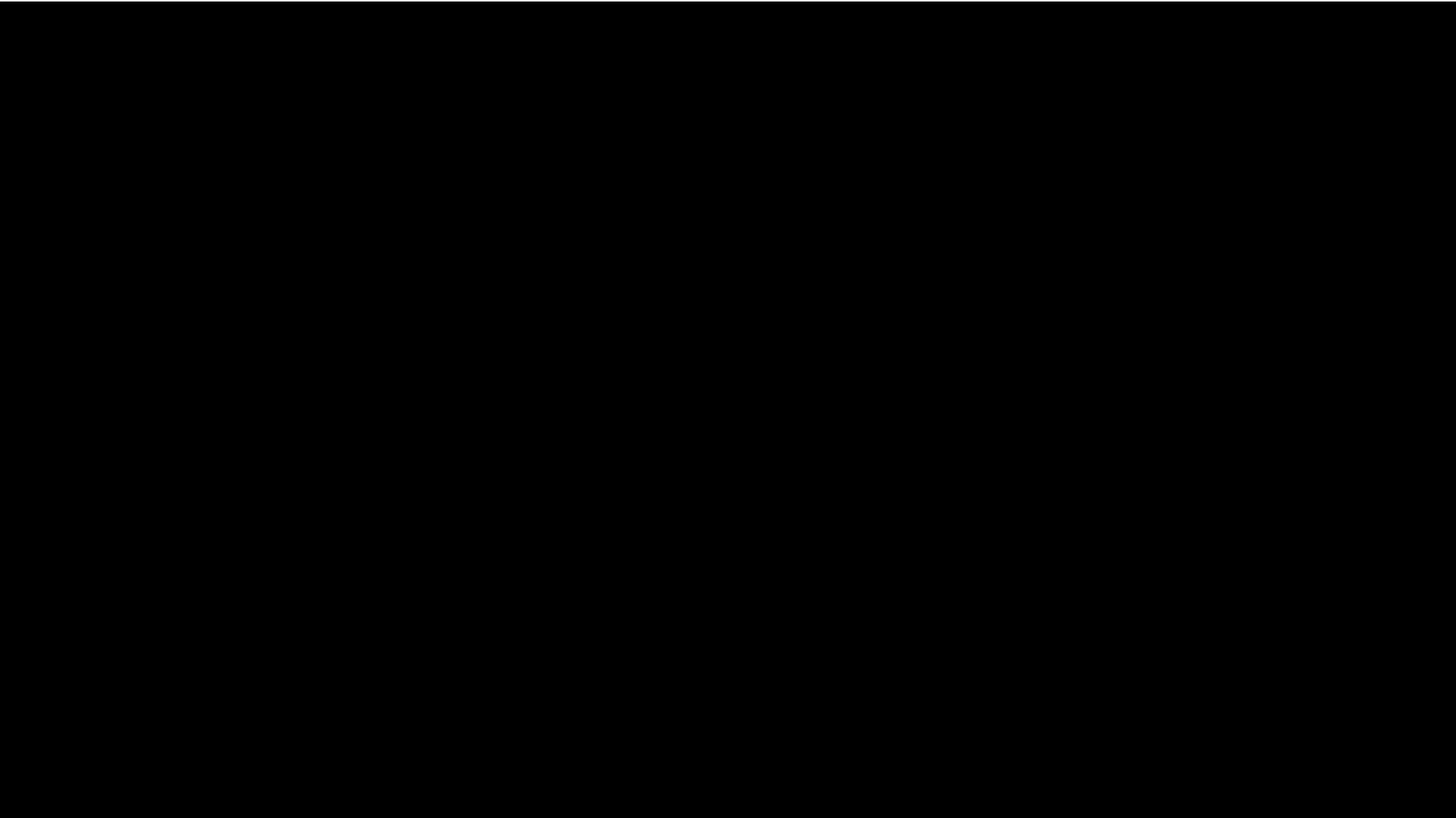
YOLO v3

- Extremadamente rápido
- Ideal para escenarios de tiempo real.



One more thing ...

Fake Videos





Conclusions

- Learn from the past
 - There is no need to reinvent the wheel
 - Know your tools
 - Identify different frameworks
-
- Time to code!!!

THANK YOU





Now is your turn