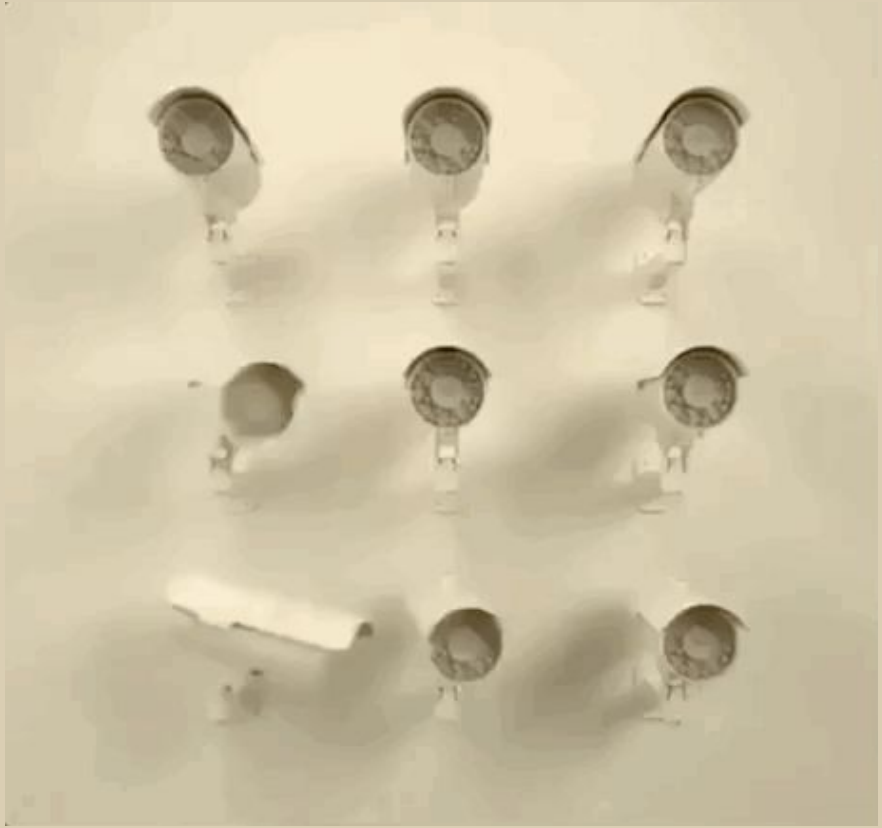
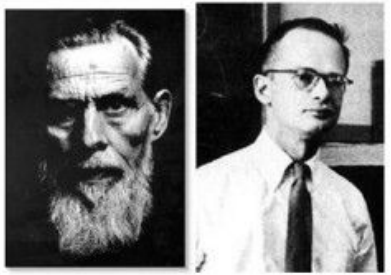
The Fundamentals of Machine Learning

Lesson #02

Table of Contents



World War II (1939 - 1945)



1943

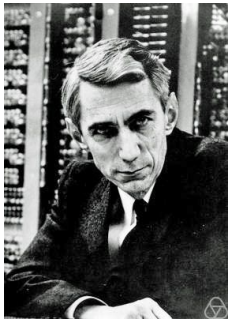
A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY

WARREN S. MCCULLOCH and WALTER H. PITTS

Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic. It is found that the behavior of every net can be described in these terms, with the addition of more complicated logical means for nets containing circles; and that for any logical expression satisfying certain conditions, one can find a net behaving in the fashion it describes. It is shown that many particular choices among possible neurophysiological assumptions are equivalent, in the sense that for every net behaving under one assumption, there exists another net which behaves under the other and gives the same results, although perhaps not in the same time. Various applications of the calculus are discussed.

INTRODUCTION

THEORETICAL neurophysiology rests on certain cardinal assumptions. The nervous system is a net of neurons, each having a



1950

Philosophical Magazine, Ser.7, Vol. 41, No. 314 - March 1950.

XXII. Programming a Computer for Playing Chess[1]

By CLAUDE E. SHANNON

Bell Telephone Laboratories, Inc., Murray Hill, N.J.[2]

[Received November 8, 1949]

1. INTRODUCTION

This paper is concerned with the problem of constructing a computing routine or "program" for a modern general purpose computer which will enable it to play chess. Although perhaps of no practical importance, the question is of theoretical interest, and it is hoped that a satisfactory solution of this problem will act as a wedge in attacking other problems of a similar nature and of greater significance. Some possibilities in this direction are: -

(1)Machines for designing filters, equalizers, etc.
(2)Machines for designing relay and switching circuits.
(3)Machines which will handle routing of telephone calls based on the individual circumstances rather than by fixed patterns.
(4)Machines for performing symbolic (non-numerical) mathematical operations.
(5)Machines capable of translating from one language to another.
(6)Machines for making strategic decisions in simplified military operations.
(7)Machines capable of orchestrating a melody.
(8)Machines capable of logical deduction.

1950

A. M. Turing (1950) Computing Machinery and Intelligence. *Mind 49:* 433-460.

COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

1. The Imitation Game

I propose to consider the question, "Can machines think?" This should begin with definitions of the meaning of the terms "machine" and "think." The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous, If the meaning of the words "machine" and "think" are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, "Can machines think?" is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.

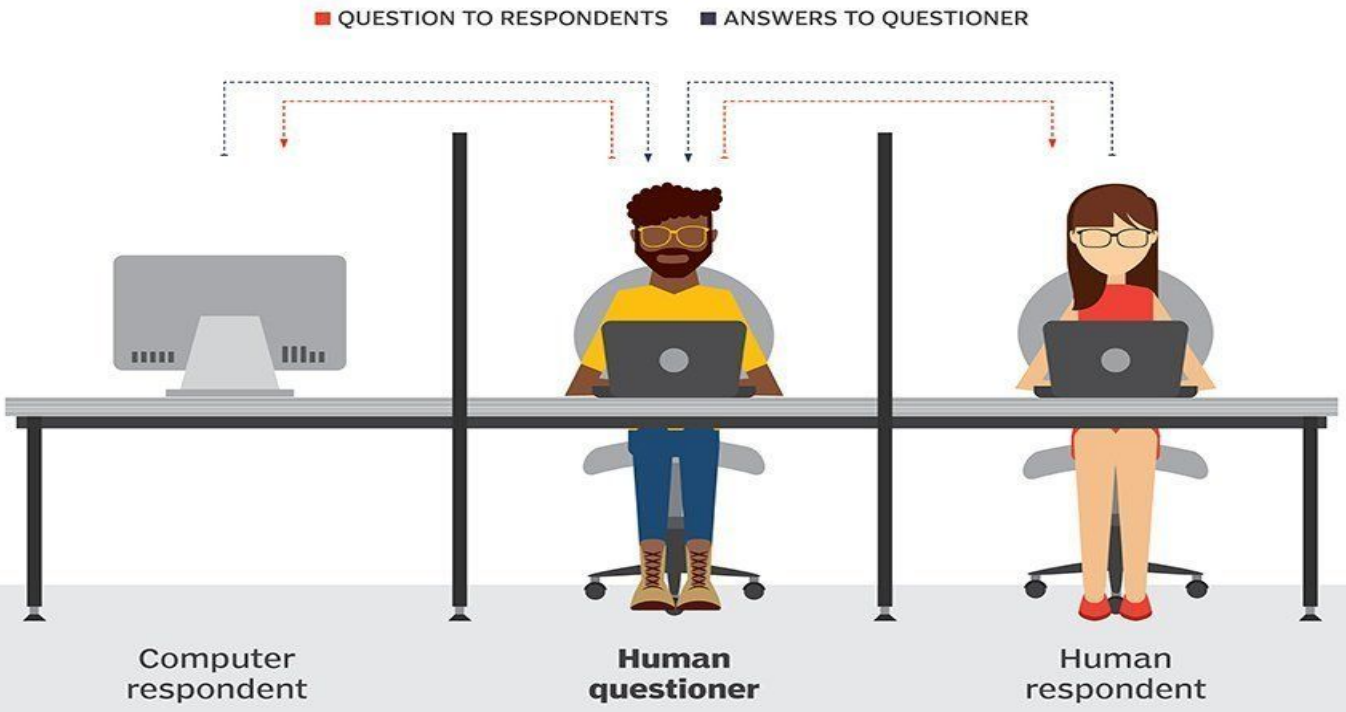
The new form of the problem can be described in terms of a game which we call the 'imitation game." It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in a room apart front the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either "X is A and Y is B" or "X is B and Y is A." The interrogator is allowed to put questions to A and B thus:

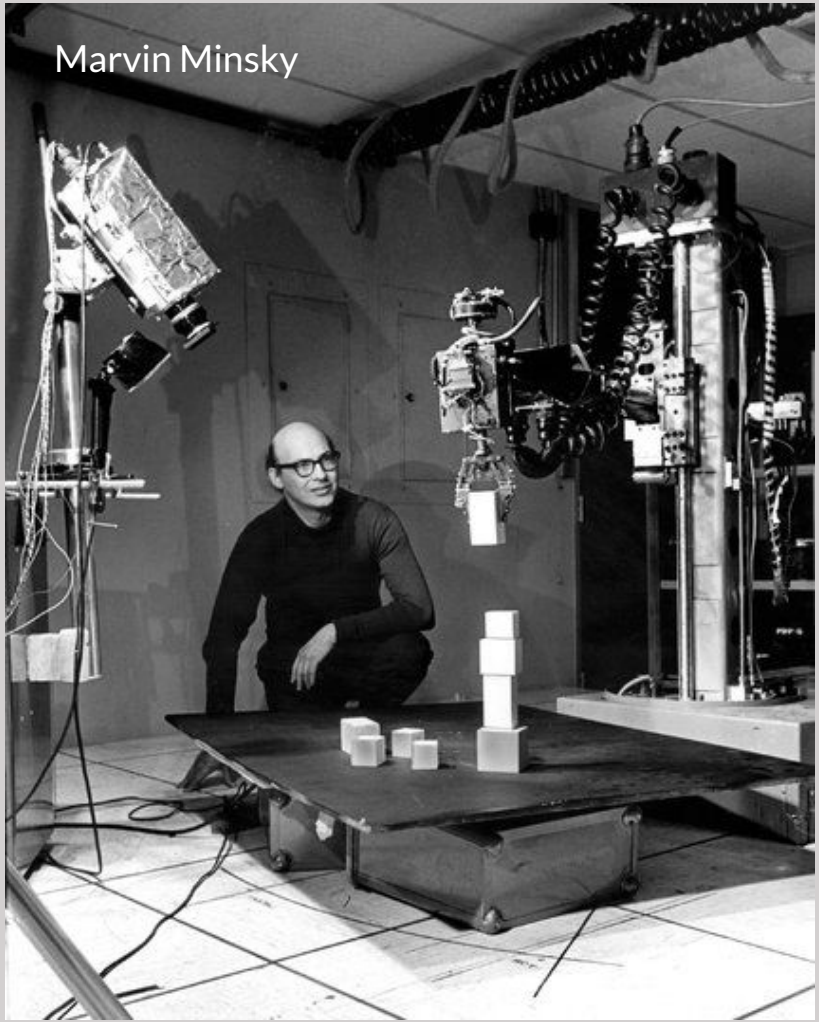
C: Will X please tell me the length of his or her hair?

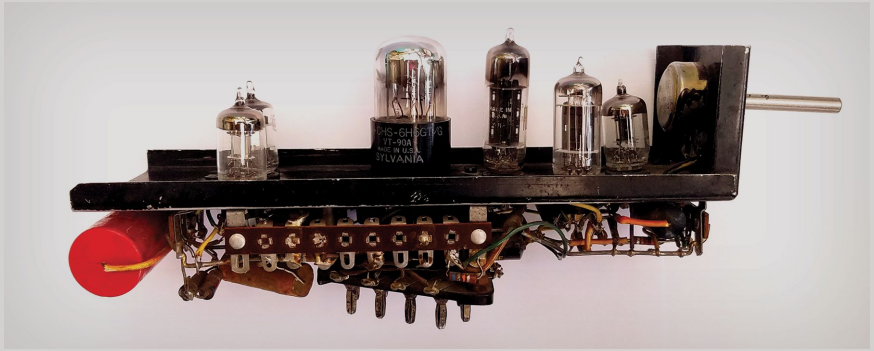
Now suppose X is actually A, then A must answer. It is A's object in the game to try and

Turing test

During the Turing test, the human questioner asks a series of questions to both respondents. After the specified time, the questioner tries to decide which terminal is operated by the human respondent and which terminal is operated by the computer.





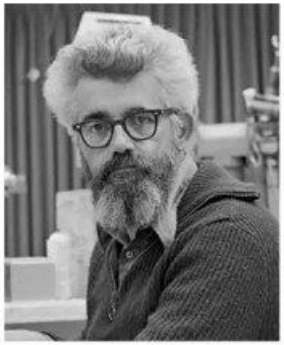


Stochastic Neural Analog Reinforcement Calculator (SNARC) 1951

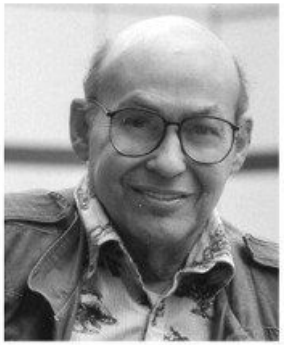
Arthur Samuel (1952)

The Samuel Checkers playing Program was among the world's first successful self-learning programs

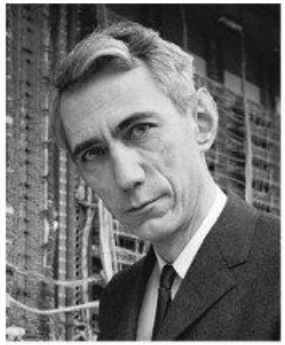
1956 Dartmouth Conference: The Founding Fathers of AI



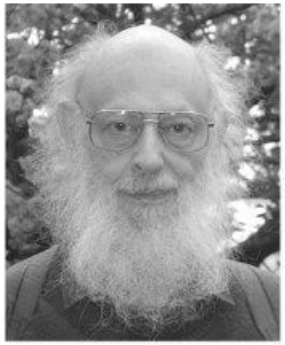
John MacCarthy



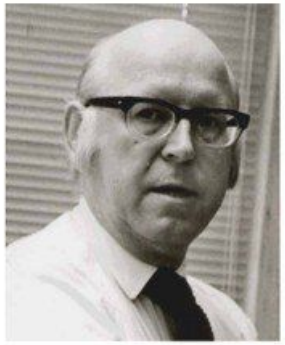
Marvin Minsky



Claude Shannon



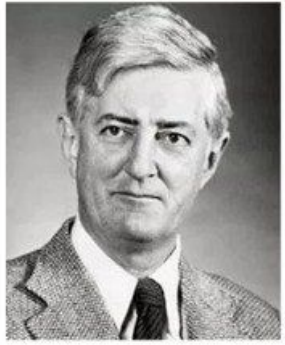
Ray Solomonoff



Alan Newell

Herbert Simon

Arthur Samuel



Oliver Selfridge

Nathaniel Rochester

Trenchard More

(2006) From left: Trenchard More, John McCarthy, Marvin Minsky, Oliver Selfridge, and Ray Solomonoff.

Dartmouth Conference, 1956


ARPA
ADVANCED RESEARCH PROJECTS AGENCY

1958

Psychological Review
Vol. 65, No. 6, 1958

THE PERCEPTRON: A PROBABILISTIC MODEL FOR INFORMATION STORAGE AND ORGANIZATION IN THE BRAIN[1]

F. ROSENBLATT

Cornell Aeronautical Laboratory

If we are eventually to understand the capability of higher organisms for perceptual recognition, generalization, recall, and thinking, we must first have answers to three fundamental questions:

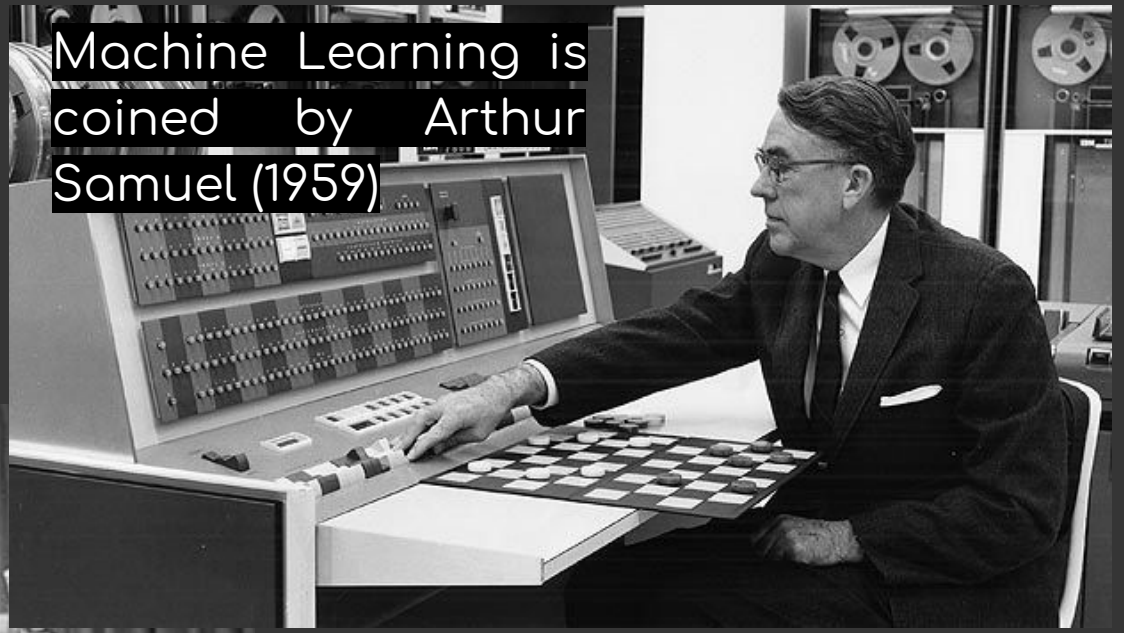
1. How is information about the physical world sensed, or detected, by the biological system?
2. In what form is information stored, or remembered?
3. How does information contained in storage, or in memory, influence recognition and behavior?

The first of these questions is in the province of sensory physiology, and is the only one for which appreciable understanding has been achieved. This article will be concerned primarily with the second and third questions, which are still subject to a vast amount of speculation, and where the few relevant facts currently supplied by neurophysiology have not yet been integrated into an acceptable theory.

With regard to the second question, two alternative positions have been maintained. The first suggests that

and the stored pattern. According to this hypothesis, if one understood the code or "wiring diagram" of the nervous system, one should, in principle, be able to discover exactly what an organism remembers by reconstructing the original sensory patterns from the "memory traces" which they have left, much as we might develop a photographic negative, or translate the pattern of electrical charges in the "memory" of a digital computer. This hypothesis is appealing in its simplicity and ready intelligibility, and a large family of theoretical brain models has been developed around the idea of a coded, representational memory (**2**, **3**, **9**, **14**). The alternative approach, which stems from the tradition of British empiricism, hazards the guess that the images of stimuli may never really be recorded at all, and that the central nervous system simply acts as an intricate switching network, where retention takes the form of new connections, or pathways, between centers of activity. In many of the more recent developments of this position (Hebb's "cell assembly," and Hull's "cortical anticipatory goal response," for example) the "re-

LISP: The language that made AI possible

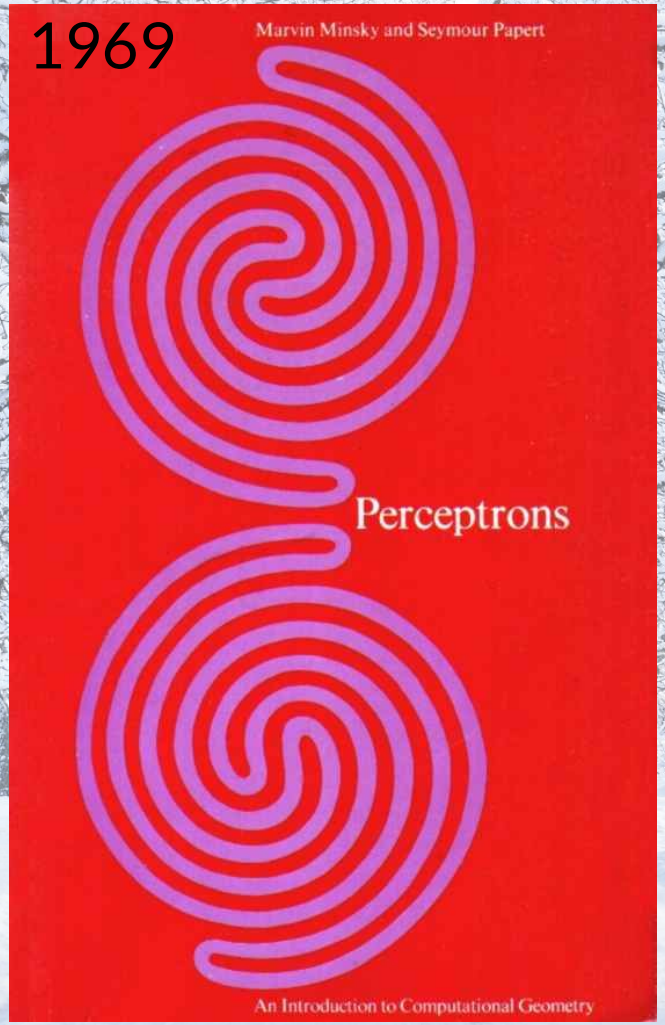


"Machine Learning: field of study that gives computers the ability to learn without being explicitly programmed"

- 1960
  - one Defense computer mistakenly identified the moon as an incoming missile which understandably caused great consternation
- 1961
  - IBM declared separate departments for software and hardware, meaning pure programmers officially would have a declared place to develop programs and environments
  - In the academic sphere, universities began granting the first degrees in Computer Science.
- 1962
  - during the Cuban Missile crisis, when communications were blocked for several days.
- 1963
  - DARPA funding in 1963, MIT created a new research group Project MAC (Mathematics and Computation) - 2M U$

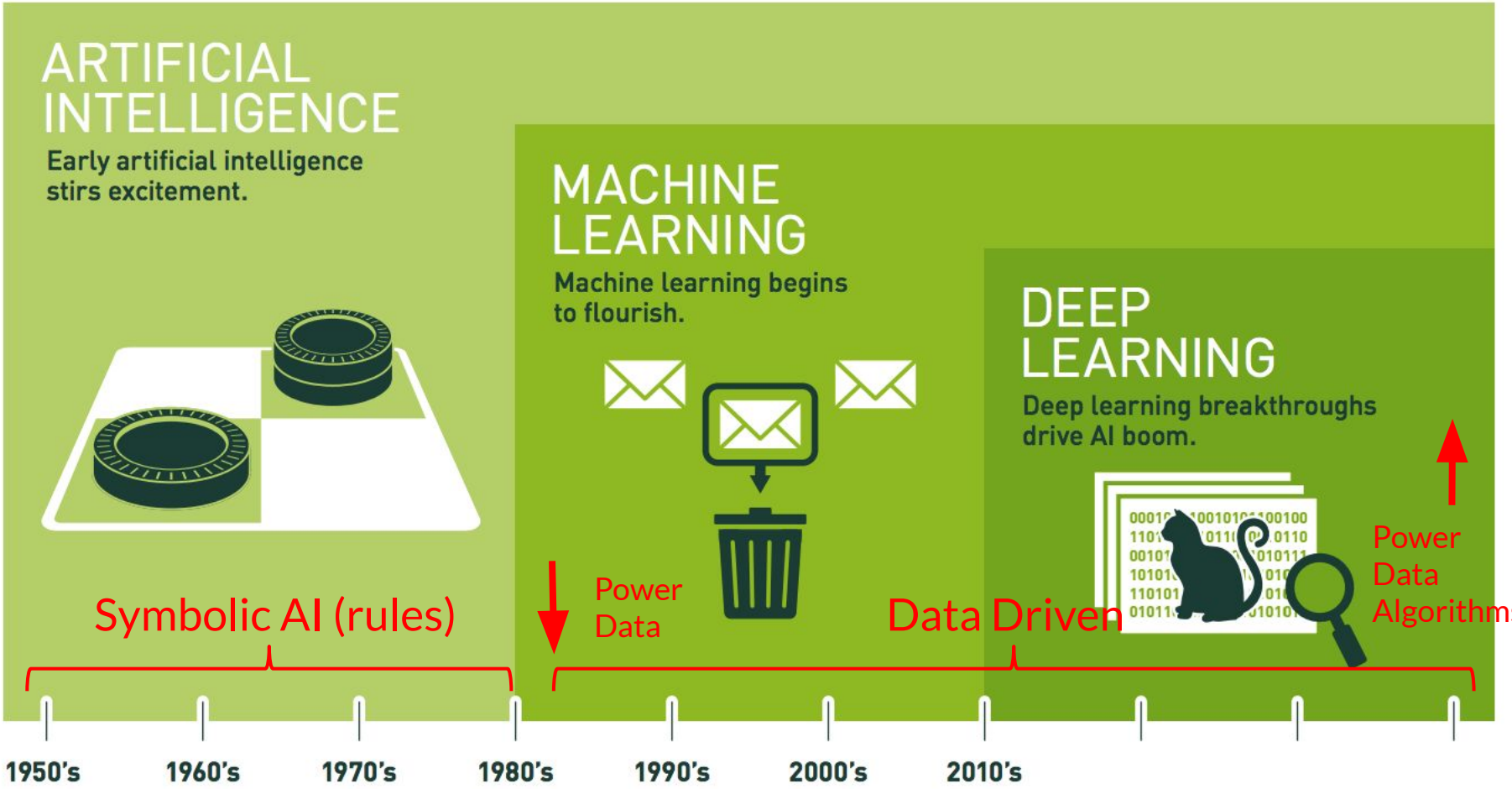
First chabot - Eliza (1964, Joseph Weizenbaum, MIT AI Lab)

1969


Marvin Minsky and Seymour Papert
Perceptrons
An Introduction to Computational Geometry


AI Winter
(1970-1980)

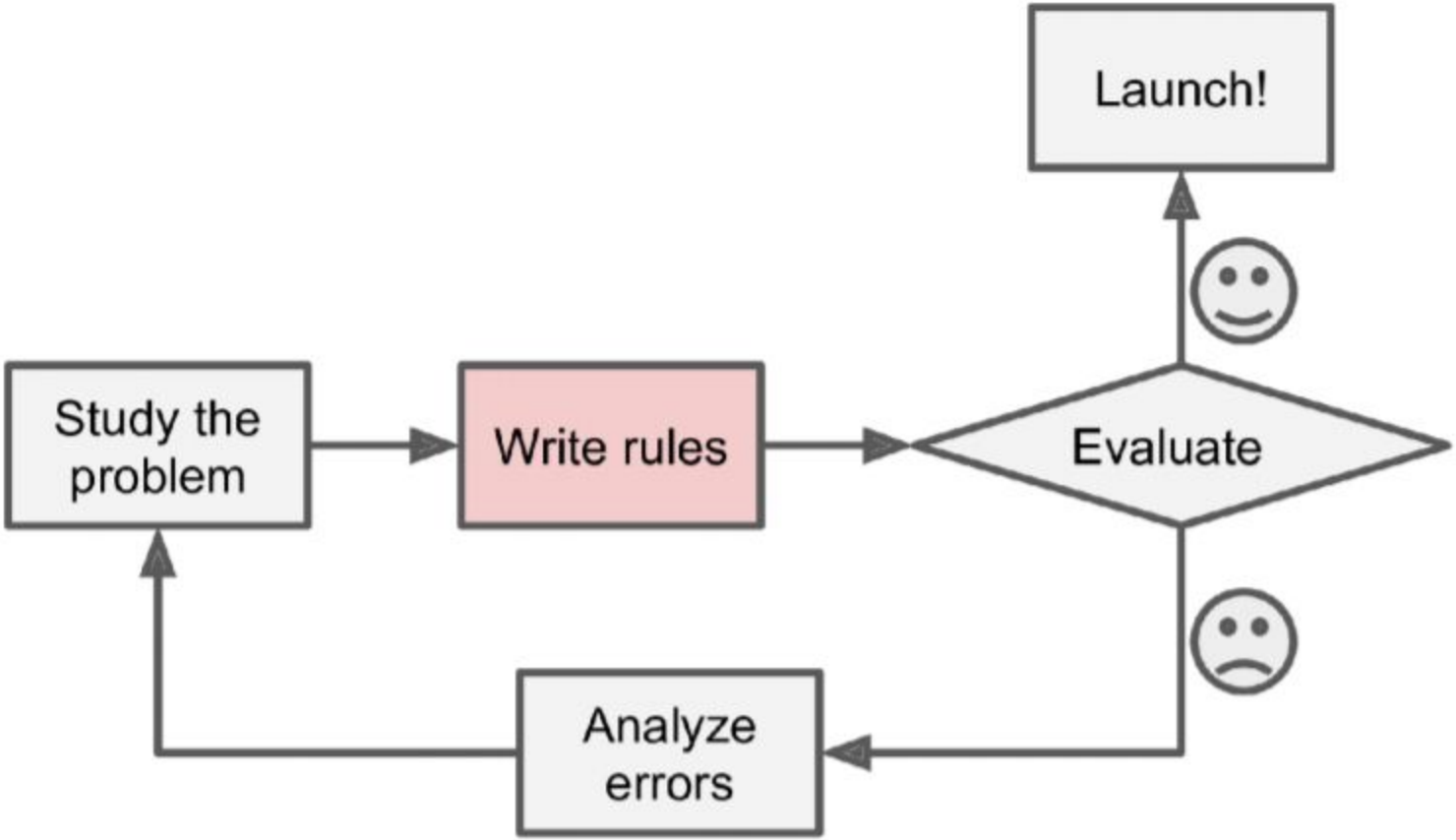
How do they related to each other?



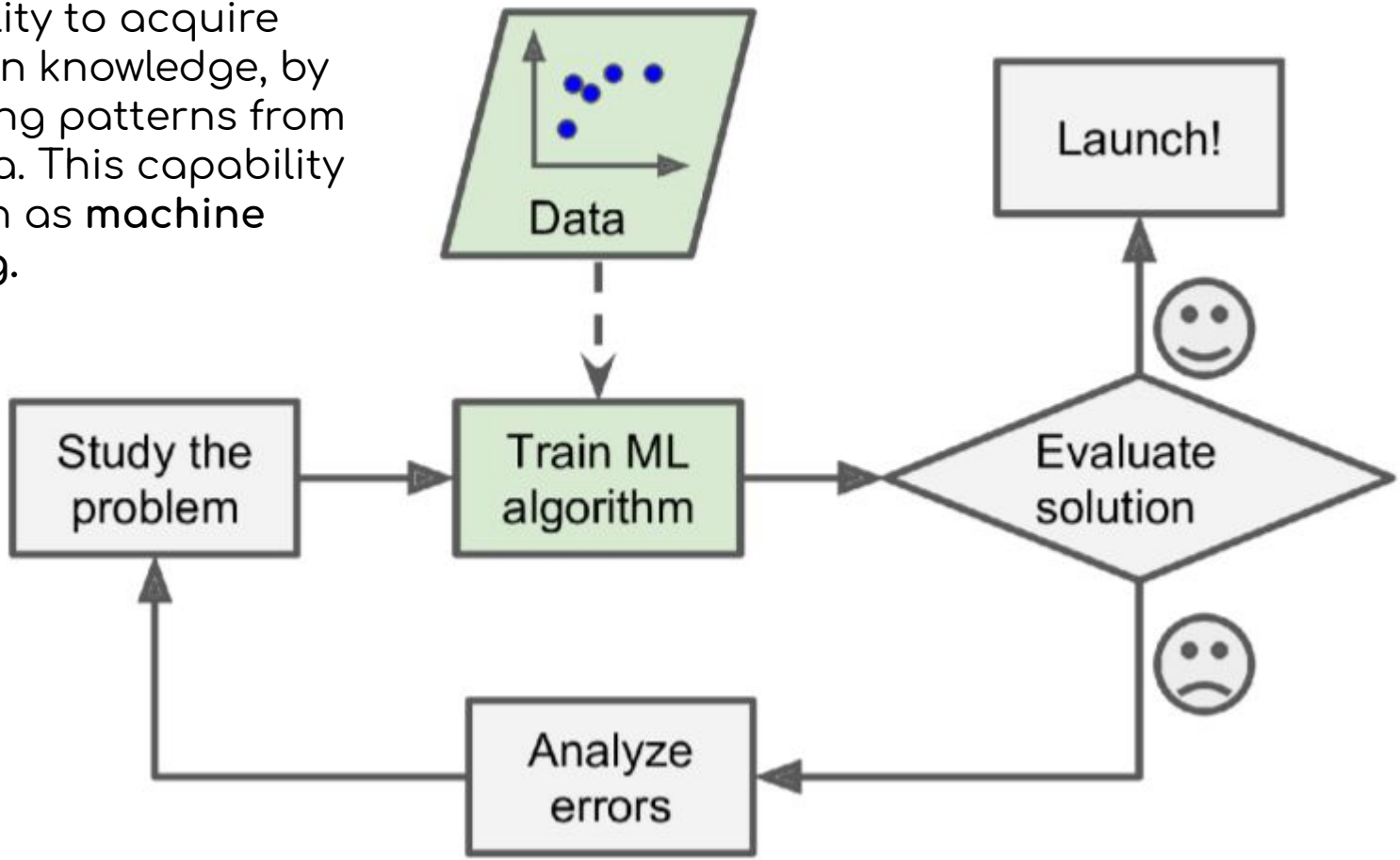

NVIDIA CORP · 1M · Cboe BZX
539.20 Máx. 589.07 Mín. 468.17 Fch 494.32 −40.66 (−7.60%)
494.30
0.41
494.71
Vol 244.615M
NVIDIA
USD
600.00
560.00
520.00
494.32
13d 2h
440.00
400.00
360.00
320.00
280.00
240.00
200.00
160.00
120.00
80.00
40.00
0.00
-40.00
2008
2010
2012
2014
2016
2018
2020

MACHINE LEARNING

A NEW PROGRAMMING PARADIGM

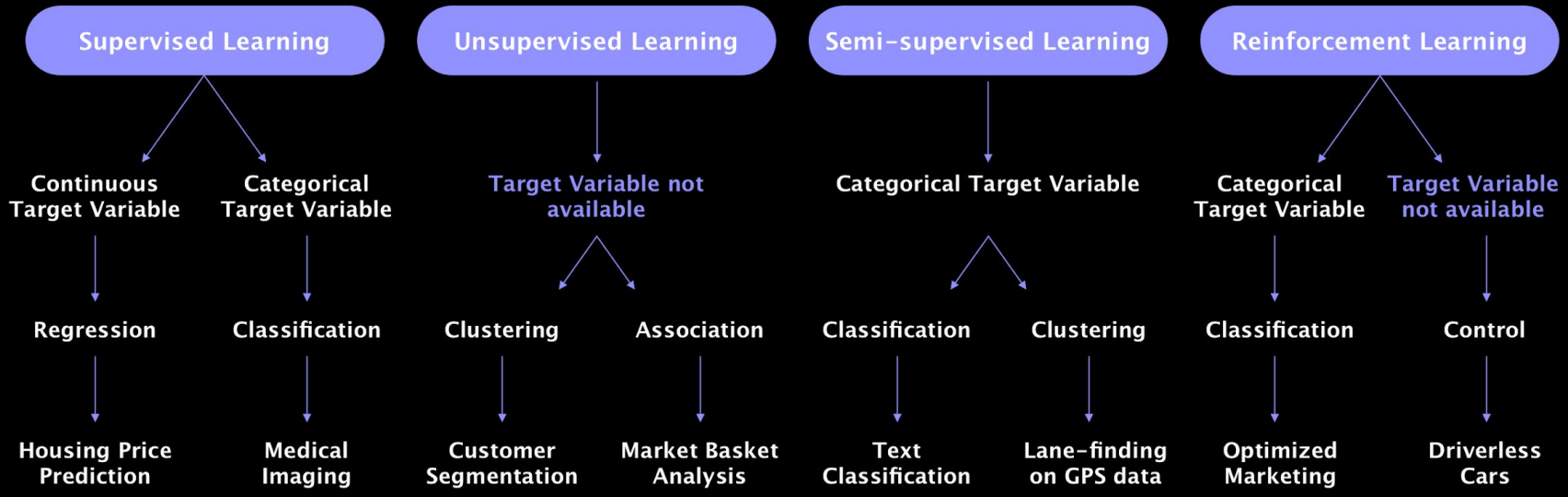

Launch!
Study the problem
Write rules
Evaluate
Analyze errors

The ability to acquire their own knowledge, by extracting patterns from raw data. This capability is known as **machine learning.**




It's time to
move on...

Machine Learning Types



https://litslink.com/blog/an-introduction-to-machine-learning-algorithms

SUPERVISED LEARNING

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease
Yes	Yes	205	Yes
No	Yes	180	Yes
Yes	No	210	Yes
Yes	Yes	167	Yes
No	Yes	156	No
No	Yes	125	No
Yes	No	168	No
Yes	Yes	172	No

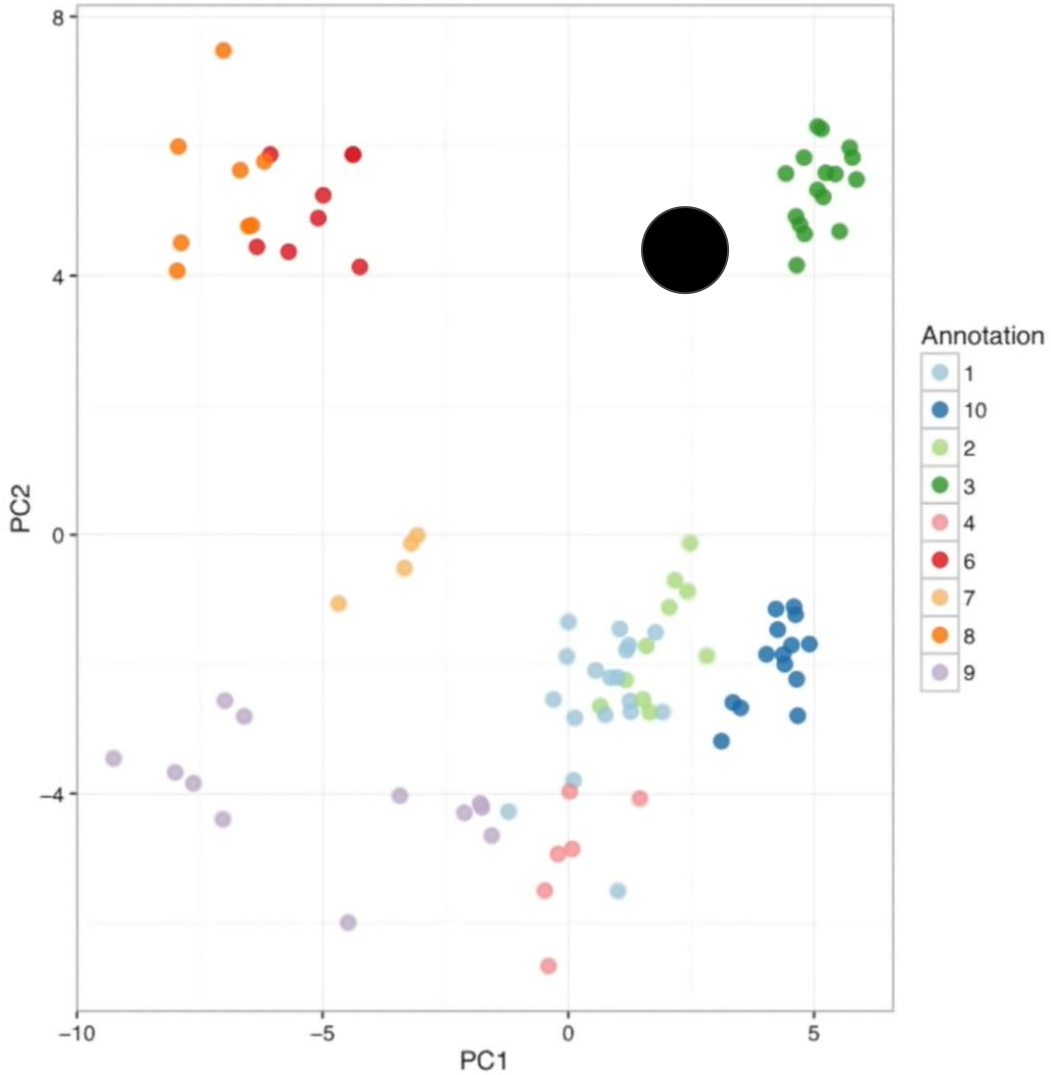
Chest Pain	Blocked Arteries	Patient Weight
Yes	No	200

Height (m)	Favorite Color	Gender	Weight (kg)
1.6	Blue	Male	88
1.6	Green	Female	76
1.5	Blue	Female	56
1.8	Red	Male	73
1.5	Green	Male	77
1.4	Blue	Female	57

Height (m)	Favorite Color	Gender
1.83	Yellow	Male

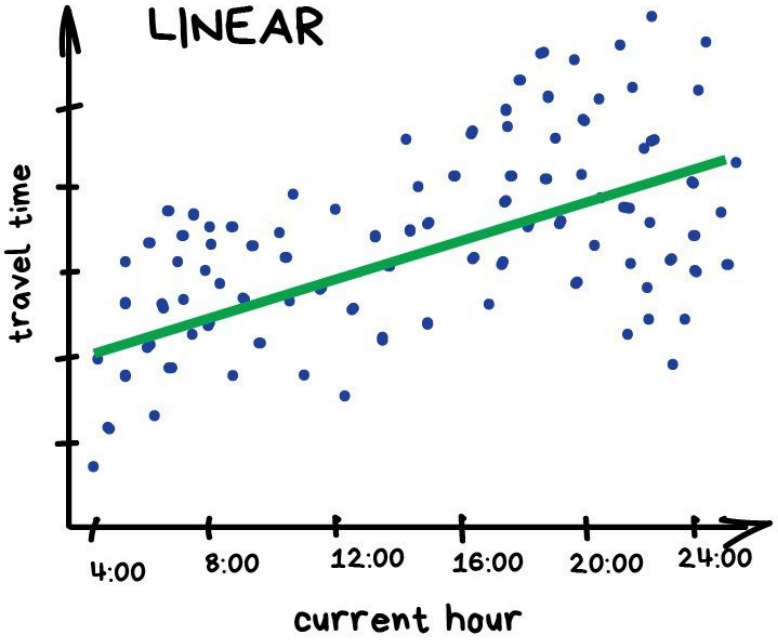
AutoML

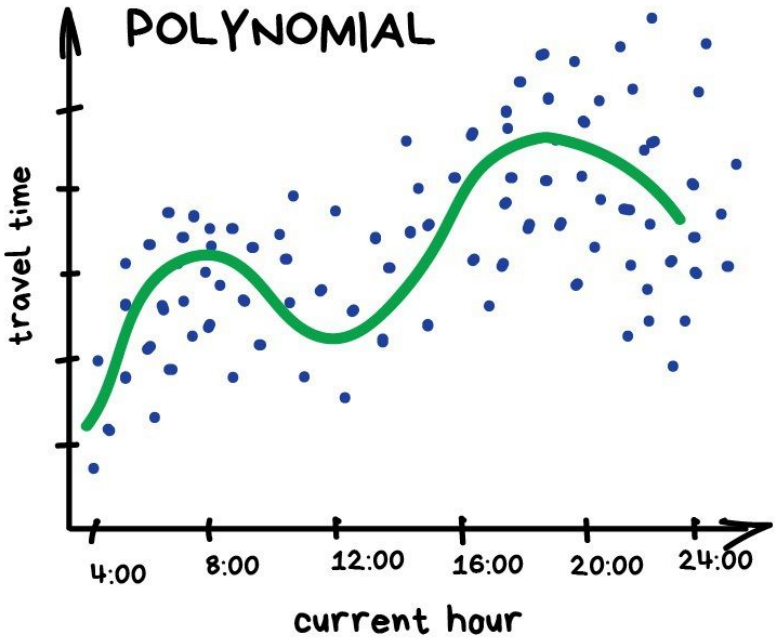
K-Nearest Neighbors (KNN)
Linear Regression
Logistic Regression
Support Vector Machines (SVM)
Decision Trees
Random Forests
Neural Networks
Deep Learning
Ensemble
XGBoost



K-Nearest Neighbors (KNN)

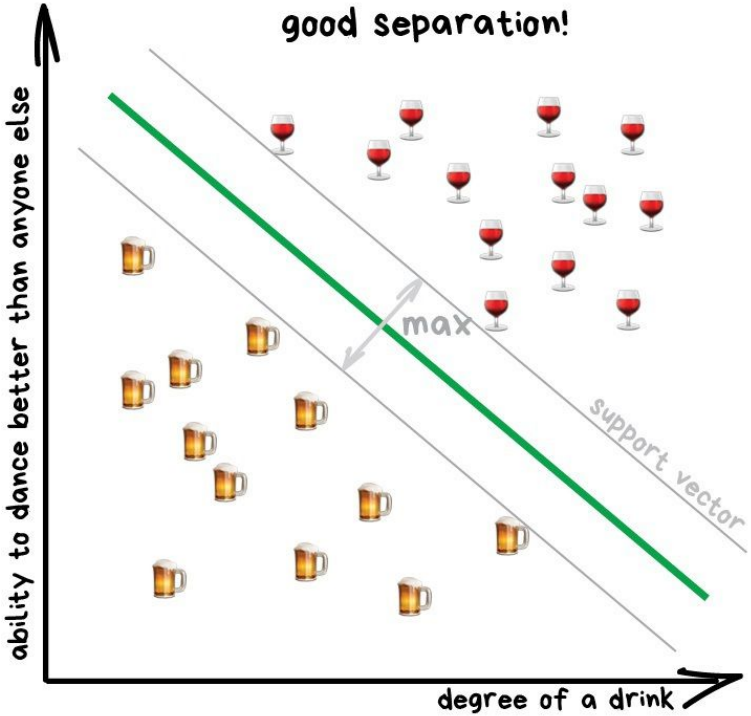
PREDICT TRAFFIC JAMS

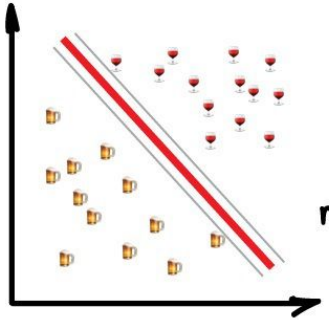


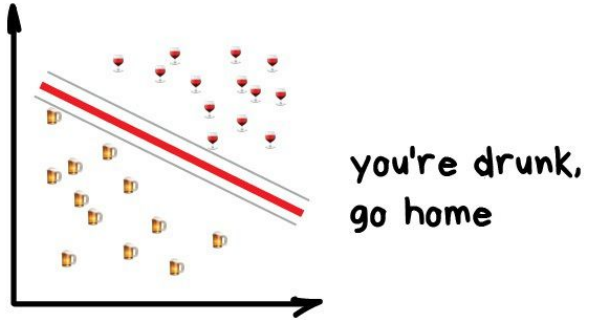


REGRESSION

SEPARATE TYPES OF ALCOHOL

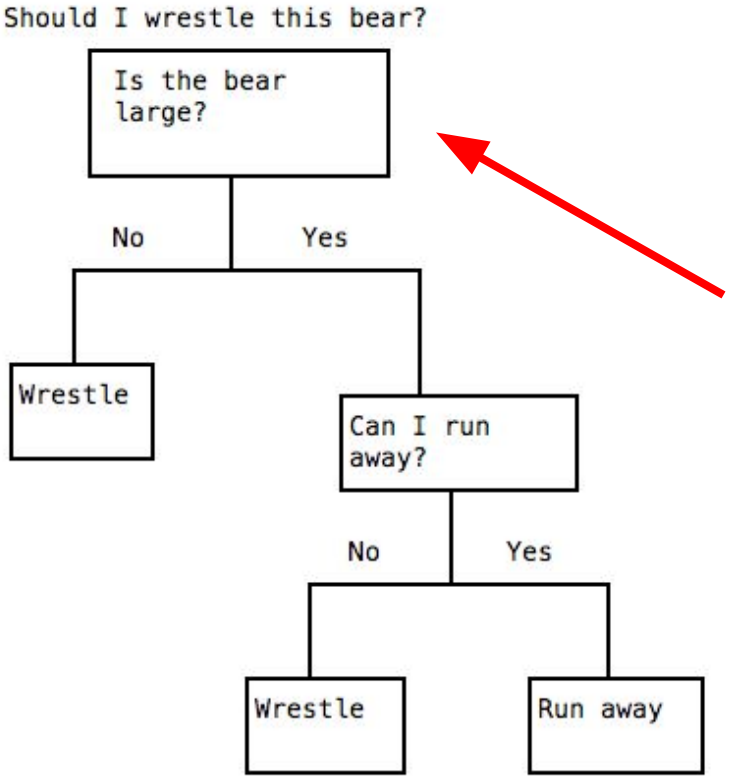




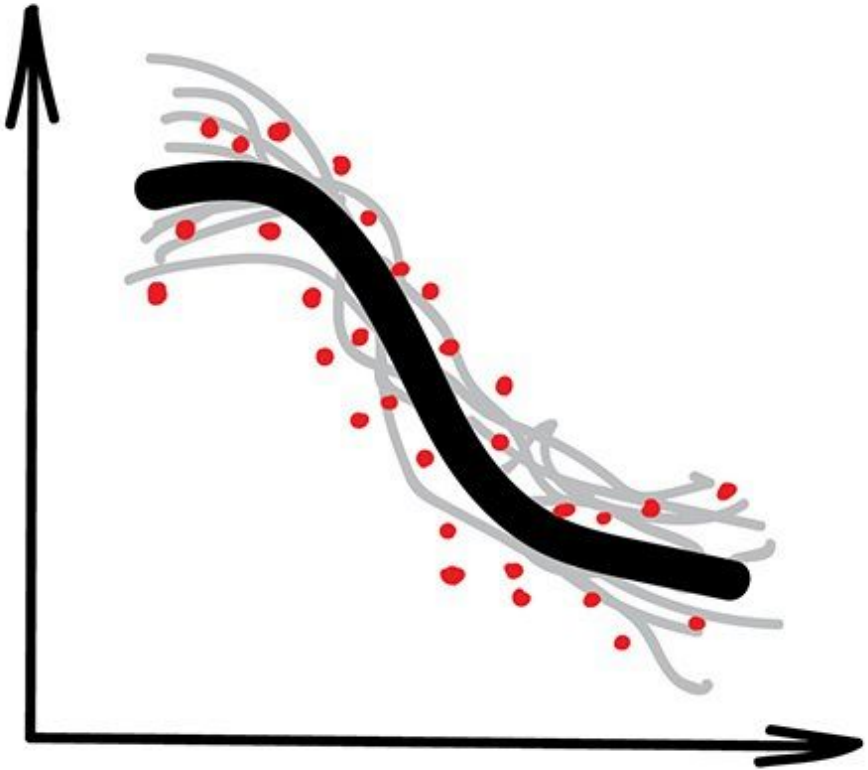


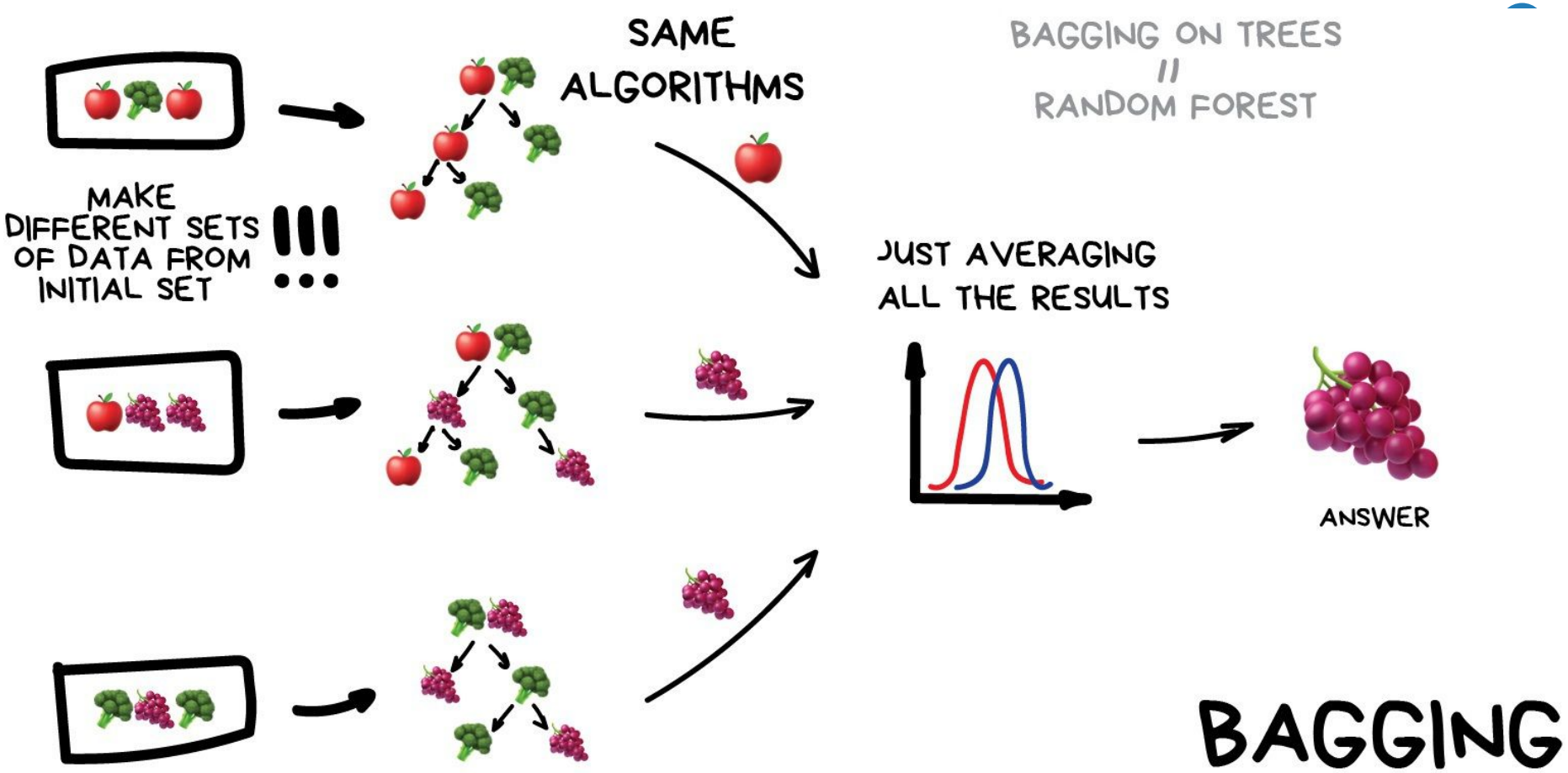
SUPPORT VECTOR MACHINE

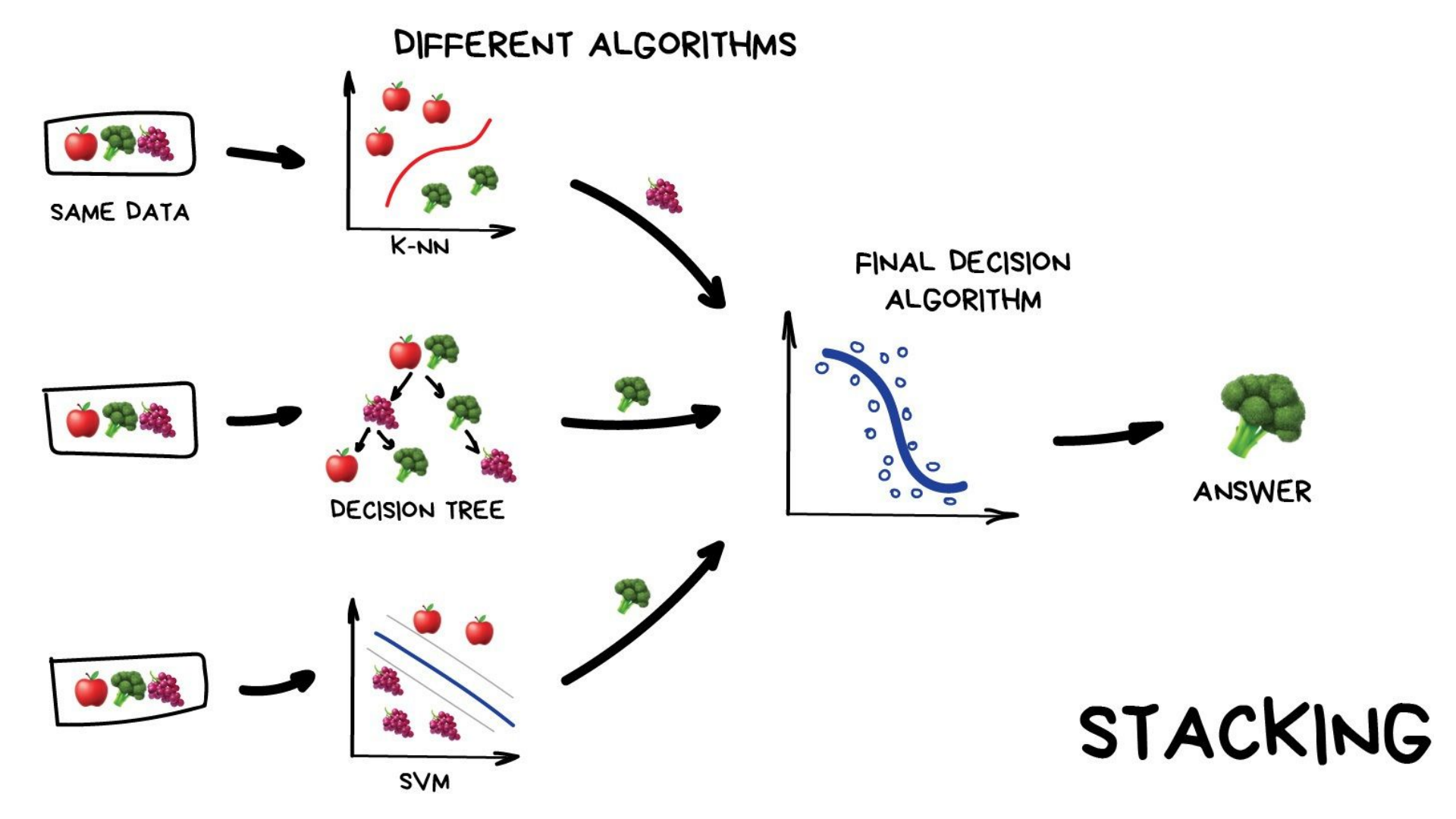
Decision Tree (classification)

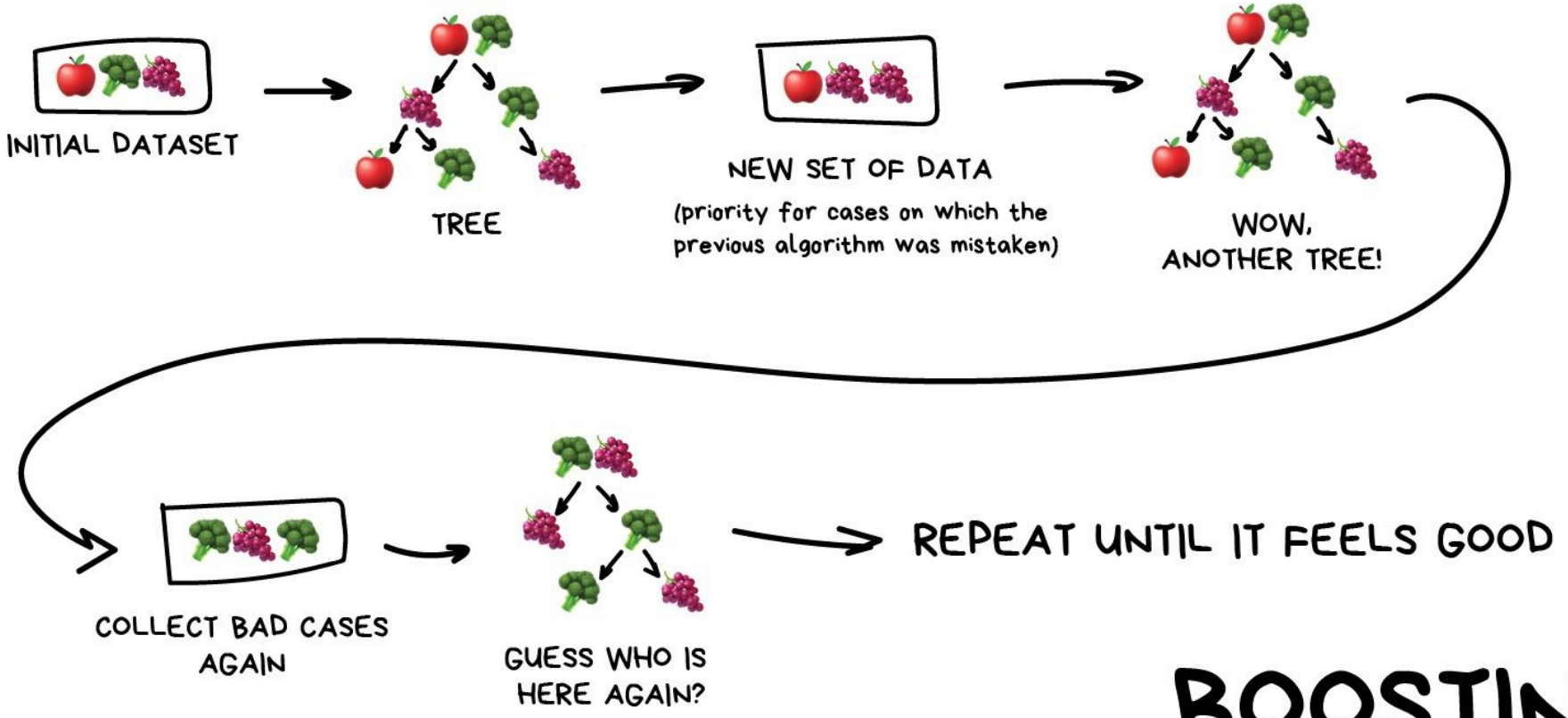


Bear name	Size	Escape possible?	Action
Yogi	Small	No	Wrestle
Winnie	Small	Yes	Wrestle
Baloo	Large	Yes	Run away
Gentle Ben	Large	No	Wrestle


Ensemble Methods


BAGGING ON TREES
=
RANDOM FOREST
SAME
ALGORITHMS
MAKE
DIFFERENT SETS
OF DATA FROM
INITIAL SET
!!!
JUST AVERAGING
ALL THE RESULTS
ANSWER
BAGGING


DIFFERENT ALGORITHMS
SAME DATA
K-NN
DECISION TREE
SVM
FINAL DECISION ALGORITHM
ANSWER
STACKING

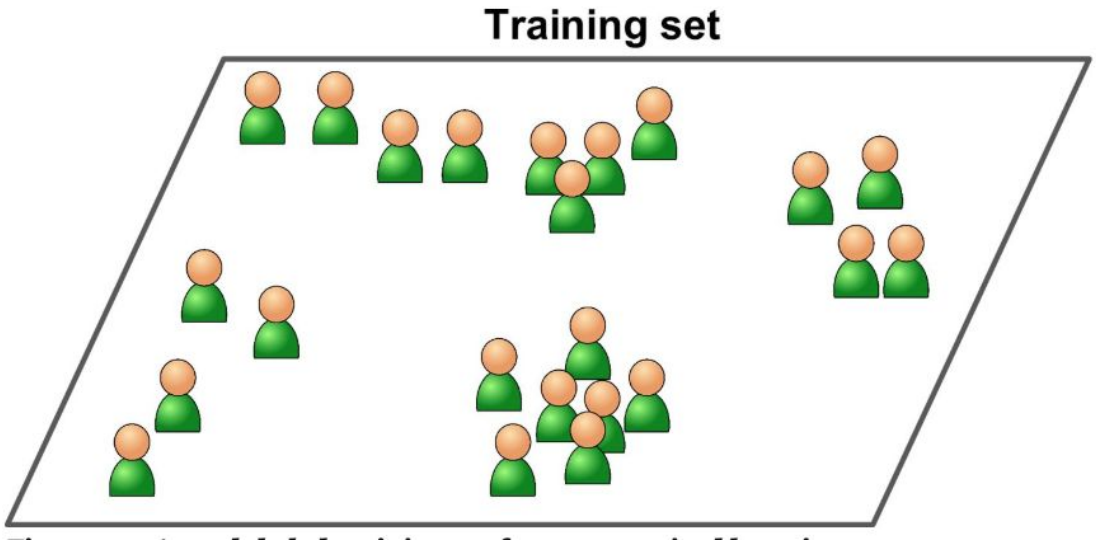

INITIAL DATASET
TREE
NEW SET OF DATA
(priority for cases on which the previous algorithm was mistaken)
WOW, ANOTHER TREE!
REPEAT UNTIL IT FEELS GOOD
GUESS WHO IS HERE AGAIN?
COLLECT BAD CASES AGAIN

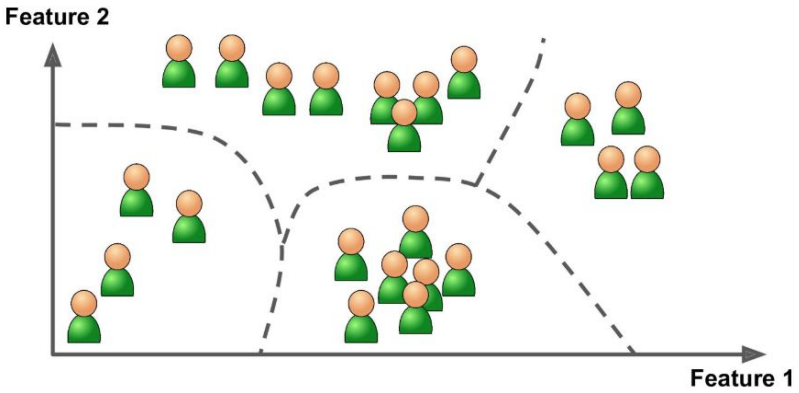
BOOSTING

UNSUPERVISED LEARNING

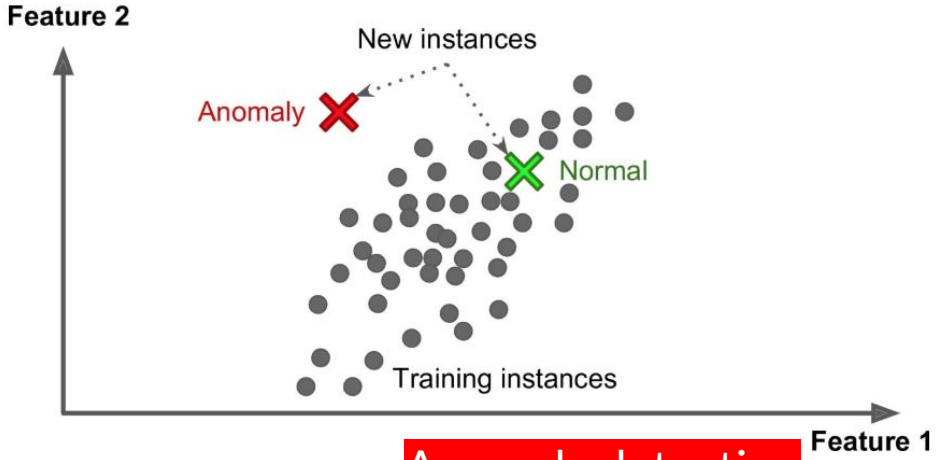
Unsupervised Learning

In unsupervised learning, as you might guess, the training data is unlabeled. The system tries to learn without a teacher.

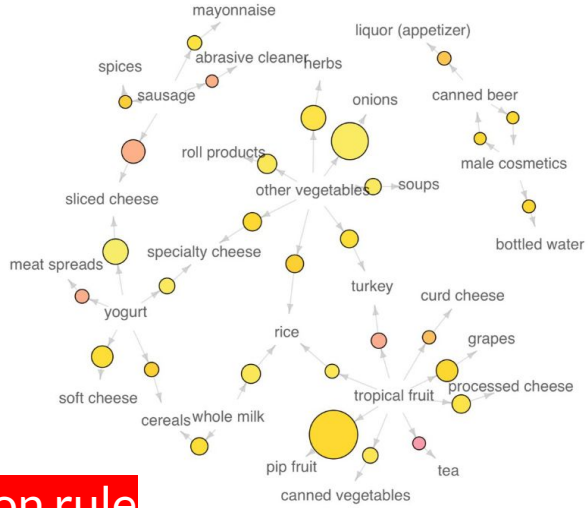




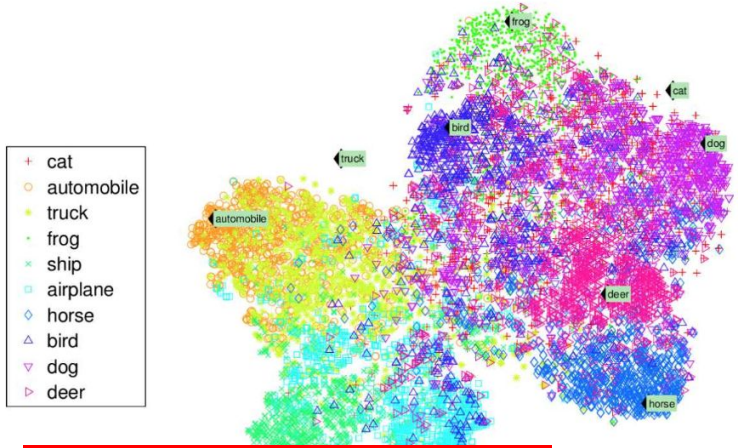
Clustering



Anomaly detection



Association rule



Visualization highlighting

Dimensionality reduction

- Principal Component Analysis (PCA)
- Kernel PCA
- Locally-Linear Embedding (LLE)
- T-distributed Stochastic Neighbor Embedding (t-SNE)

Anomaly detection

- ICA anomaly detection
- Dictionary learning

Clustering & Time Series Clustering

- K-means
- DBSCAN
- Hierarchical Clustering

Complex Network Analysis

Recommender System

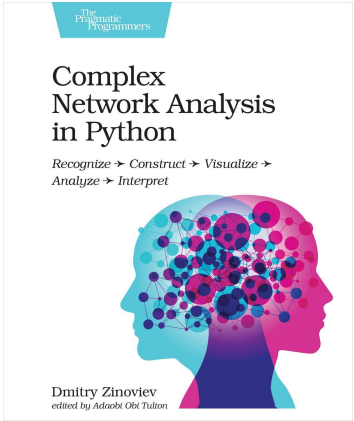
- Restricted Boltzmann Machines

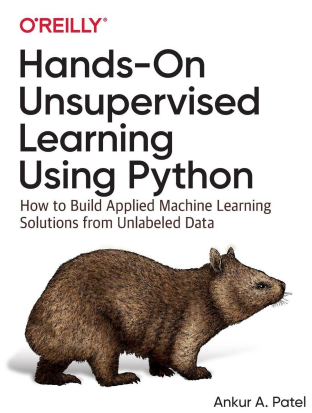
Feature Detection

- Deep Belief Networks

Generative Adversarial Networks

- GAN
- Deep Convolutional GANs





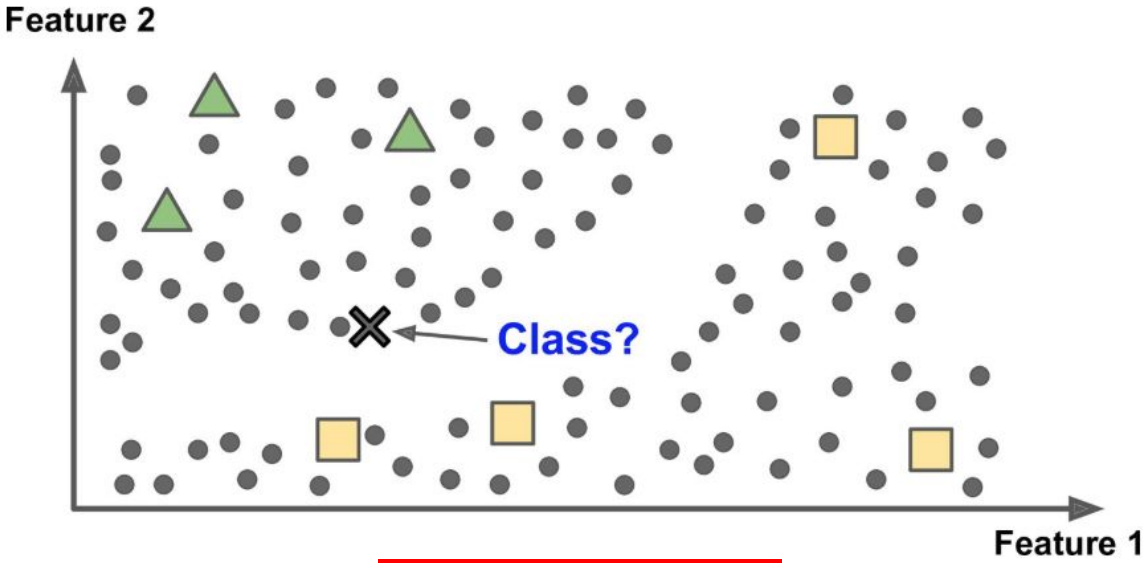
SEMI-SUPERVISED LEARNING

Semi-supervised Learning

Semi-supervised learning algorithms are trained on a combination of labeled and unlabeled data.

- The process of labeling massive amounts of data for supervised learning is often prohibitively time-consuming and expensive.
- What's more, too much labeling can impose human biases on the model.
- That means including lots of unlabeled data during the training process actually tends to improve the accuracy of the final model while reducing the time and cost spent building it

Semi-supervised Learning



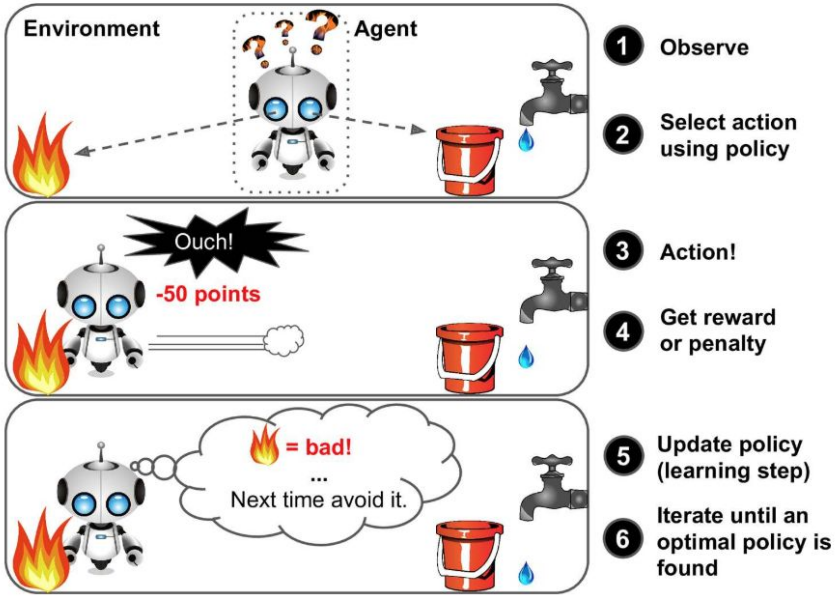
Web Page Classification
Google Photos

REINFORCEMENT LEARNING

Reinforcement Learning

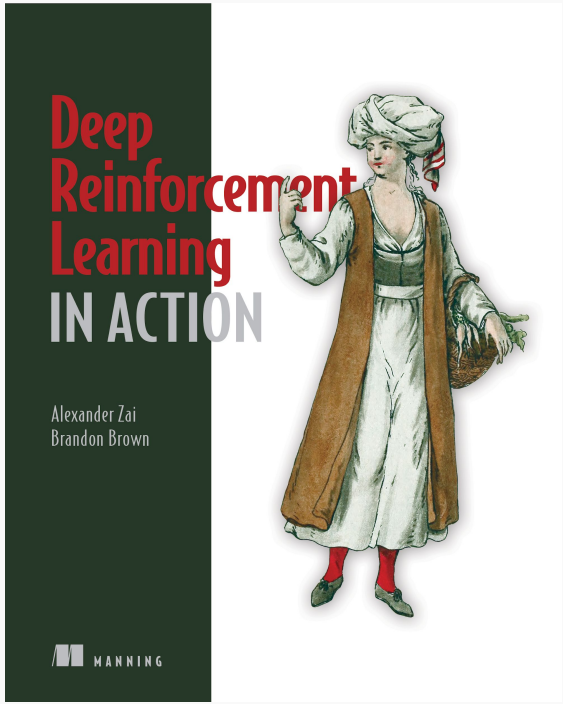
The learning system, called an agent in this context, can :

- Observe the environment
- Select and perform actions
- Get rewards in return (positive or negative)
- Learn by self what is the best strategy, called a policy

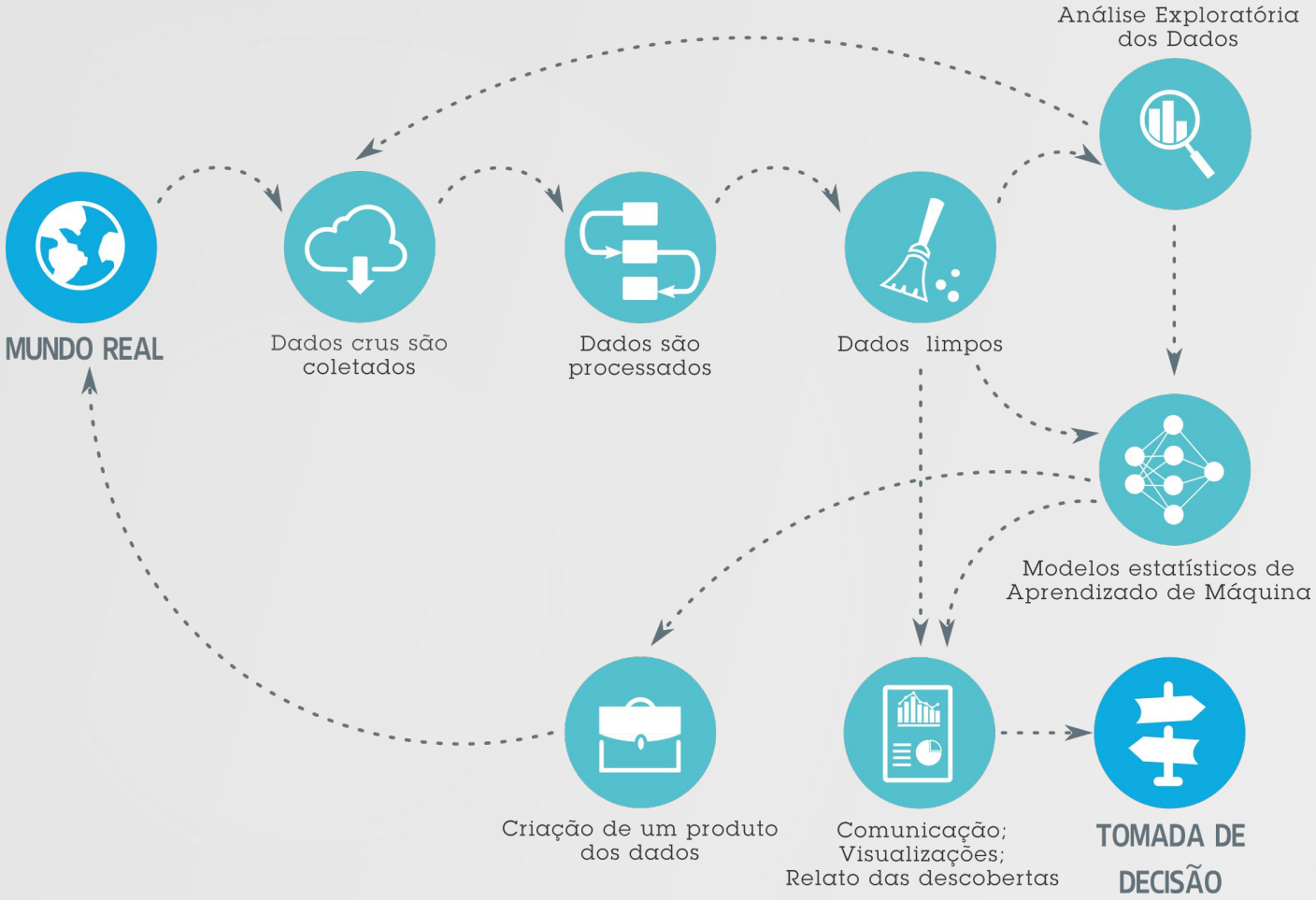


http://bit.do/playmari

- Q-Learning, Markov Decision Process
- State-Action-Reward-State-Action (SARSA)
- Deep Q Network (DQN)
- Deep Deterministic Policy Gradient (DDPG)



MAIN CHALLENGES OF MACHINE LEARNING



Gisliany Alves et al *Uma abordagem orientada a dados para a criação de um indicador de habitabilidade baseado na API da UBER*, 2019.

Titanic: Machine Learning from Disaster

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Train - Dev - Test Sets

Making good choices in how you set up your training, development, and test sets can make a huge difference in helping you quickly find a good high performance neural network.

Holdout
Cross-Validation
Validation
Development

Previous ML era

- 70/30
- 60/20/20

Big Data era

- 98/1/1
- 99.5/0.25/0.25
- 99.5/0.4/0.1

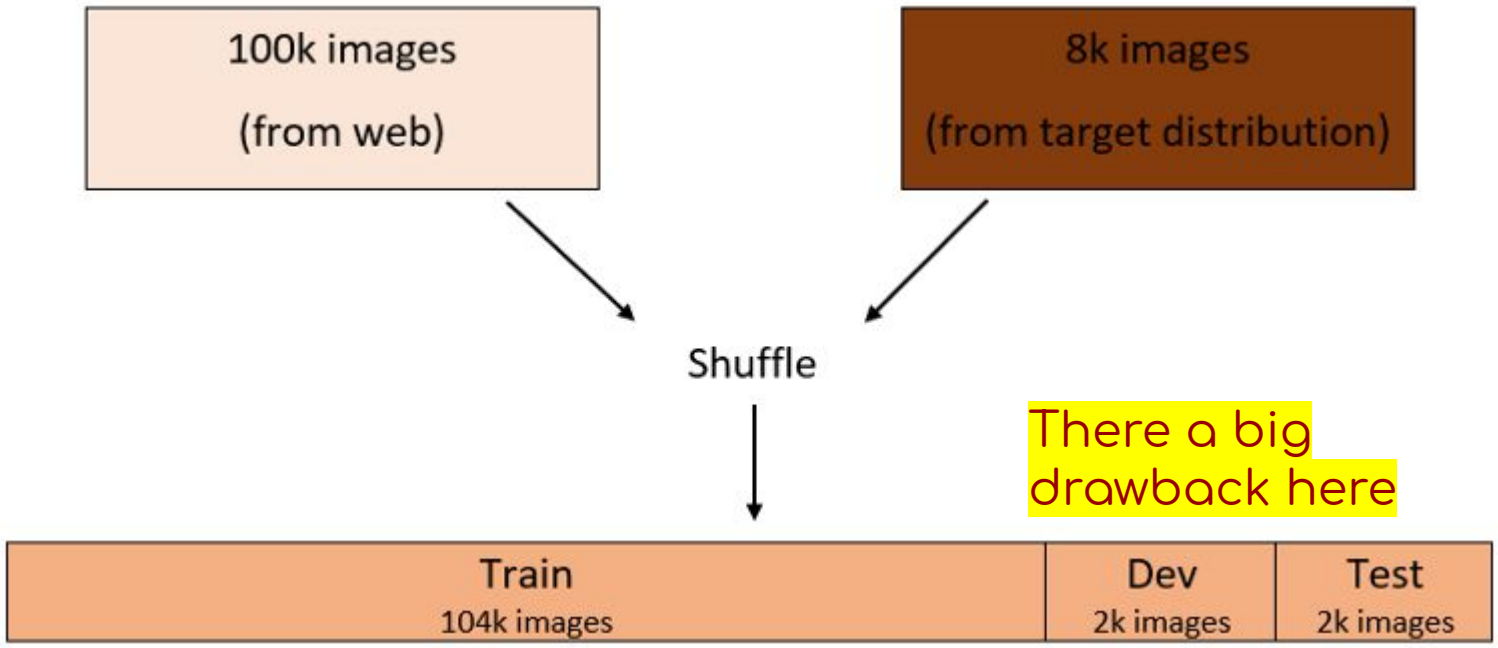
Mismatched train/test distribution

Scenario: say you are building a cat-image classifier application that determines if an image is of a cat or not. The application is intended for users in rural areas who can take pictures of animals by their mobile devices for the application to classify the animals for them.

Scraped from Web Pages
100k images

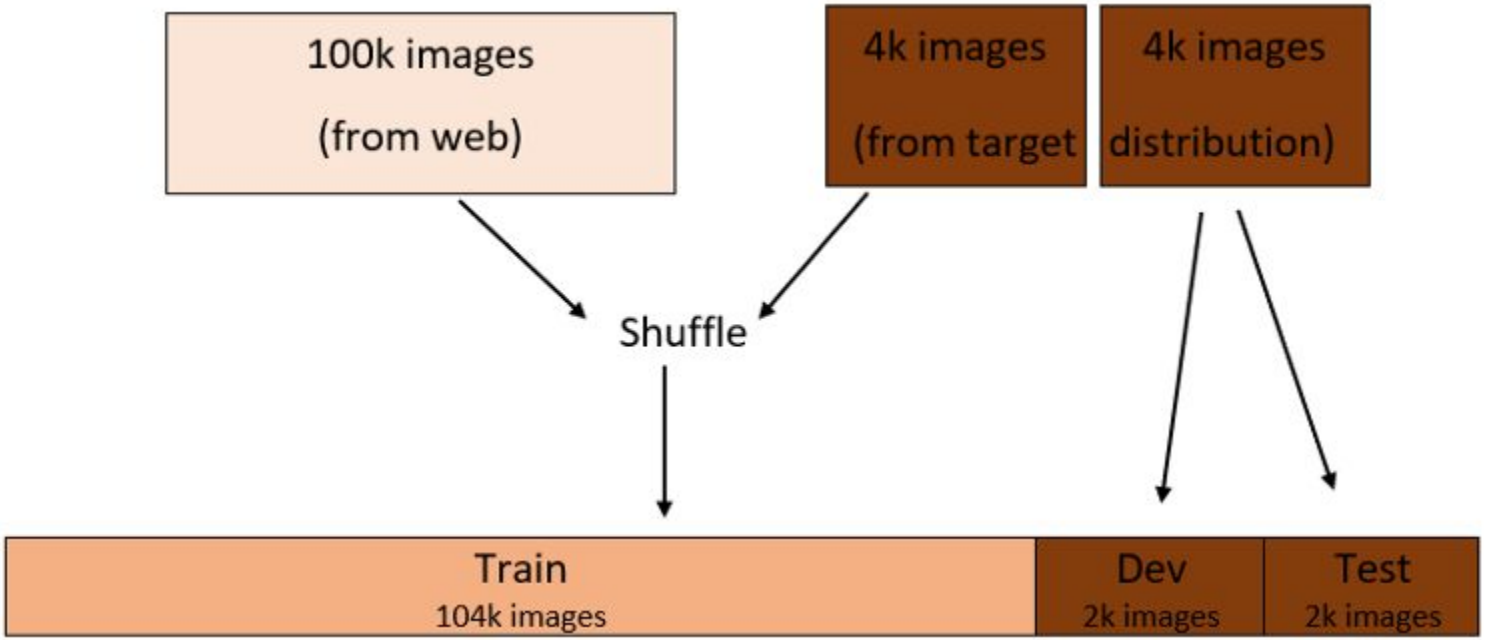
Collected from Mobile Devices
<<target distribution>>
8k images

A possible option: shuffling the data



only 148 images come from the target distribution

A better option

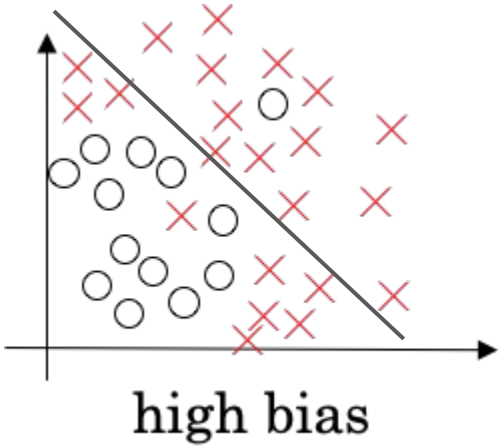


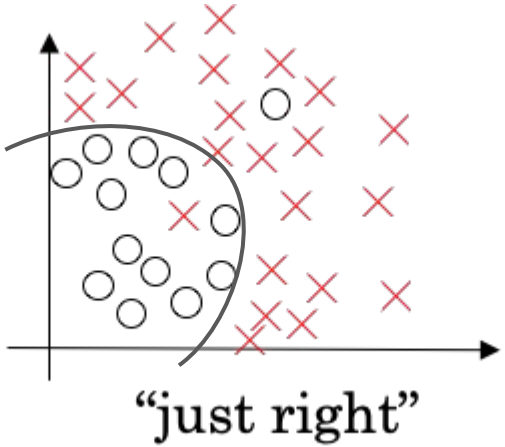
Rule of the thumb

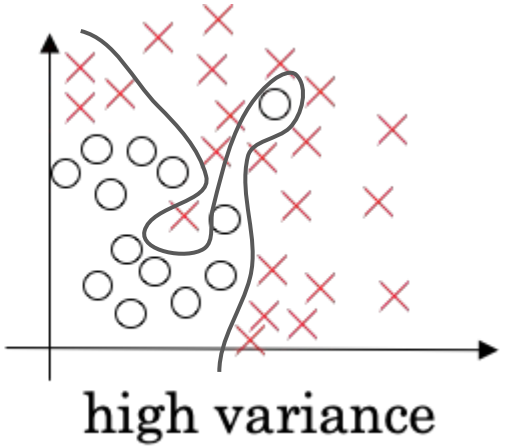
>> make sure that the dev and test sets come from the same distribution

Not having a test set might be okay. (Only dev set)

Bias vs Variance







Underfitting

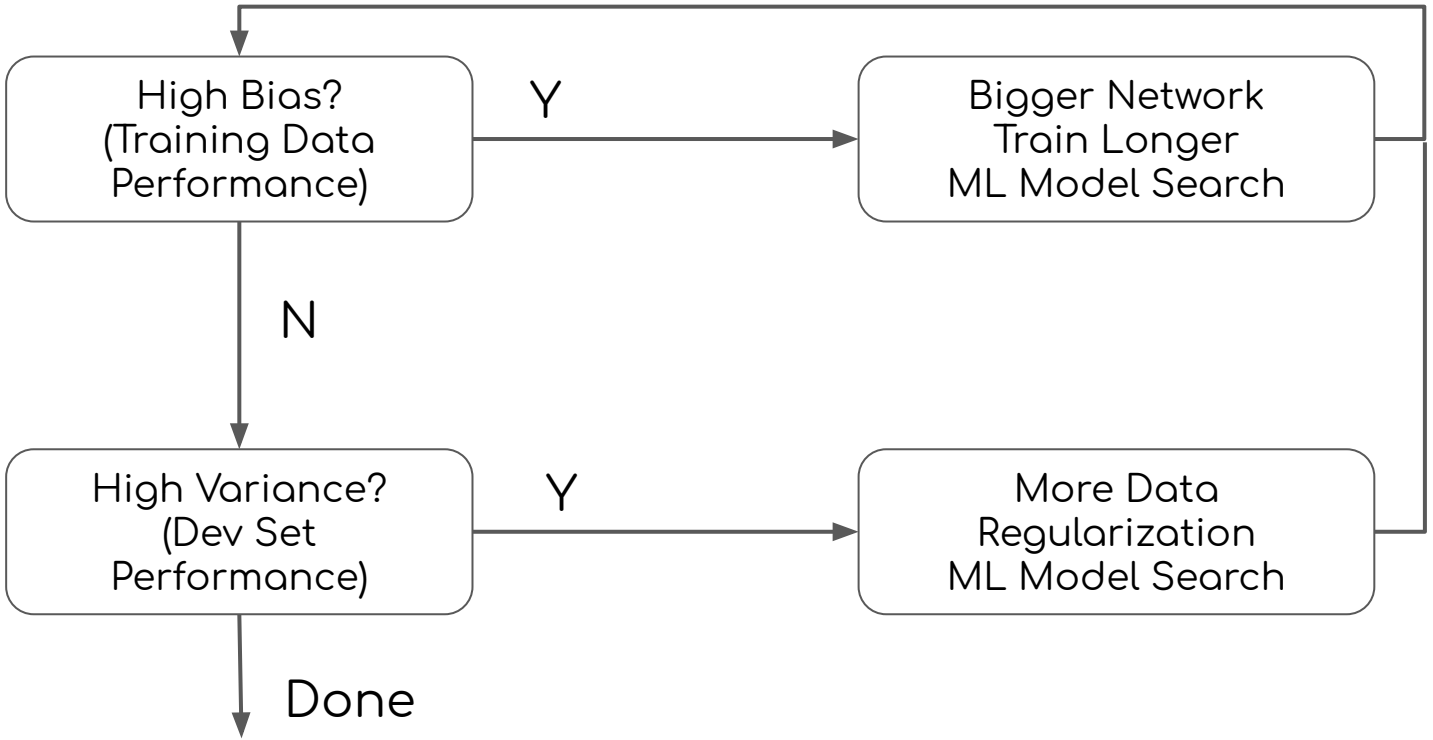
Overfitting

Bias vs Variance

Cat Classification

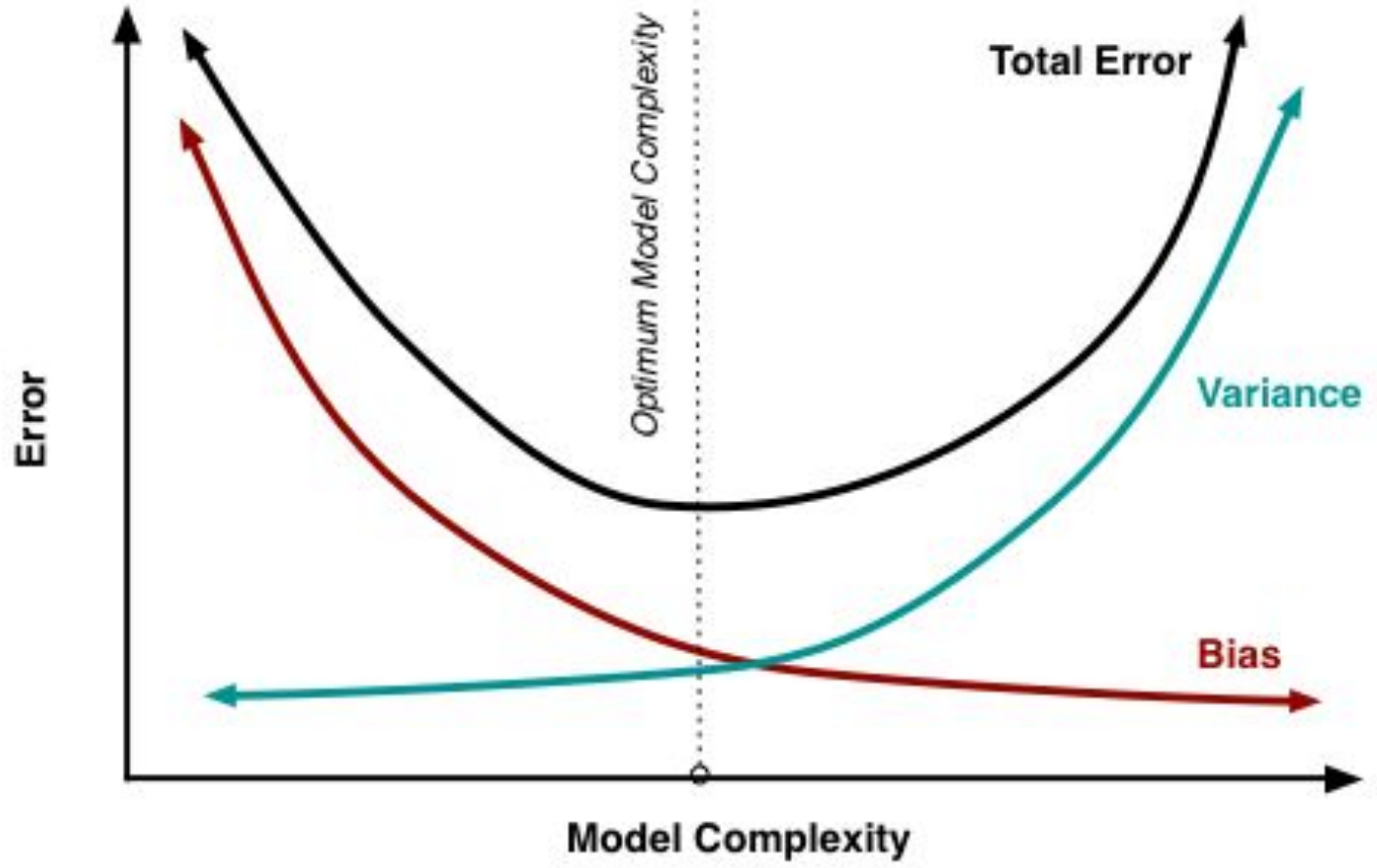
	Scenario #01	Scenario #02	Scenario #03	Scenario #04
Train Set Error	1%	15%	15%	0.5%
Dev Set Error	16%	16%	30%	1%
	Low Bias High Variance	High Bias Low Variance	High Bias High Variance	Low Bias Low Variance

Basic Recipe for Machine Learning



Underfitting

Overfitting



What tools will we use?


scikit
learn


matplotlib


colab


seaborn


python™


plotly

Next ....

https://www.imd.ufrn.br/
https://github.com/ivanovitchm/machinelearning2020.2