

# Introduction to Artificial Intelligence

## Mosig MI - Ensimag 2A

**Brief history of AI**  
**Problem statement basics**  
**Evaluation basics**

Original slides by Sergi Pujades

Lecture : Pierre Gaillard

# What is Artificial Intelligence?

# What is Artificial Intelligence?

Artificial intelligence refers to the simulation of human intelligence in machines that are designed to think and act like humans.

Machines that think like humans

Machines that think rationally

Machines that act like humans

Machine that act rationally

# What is Artificial Intelligence?

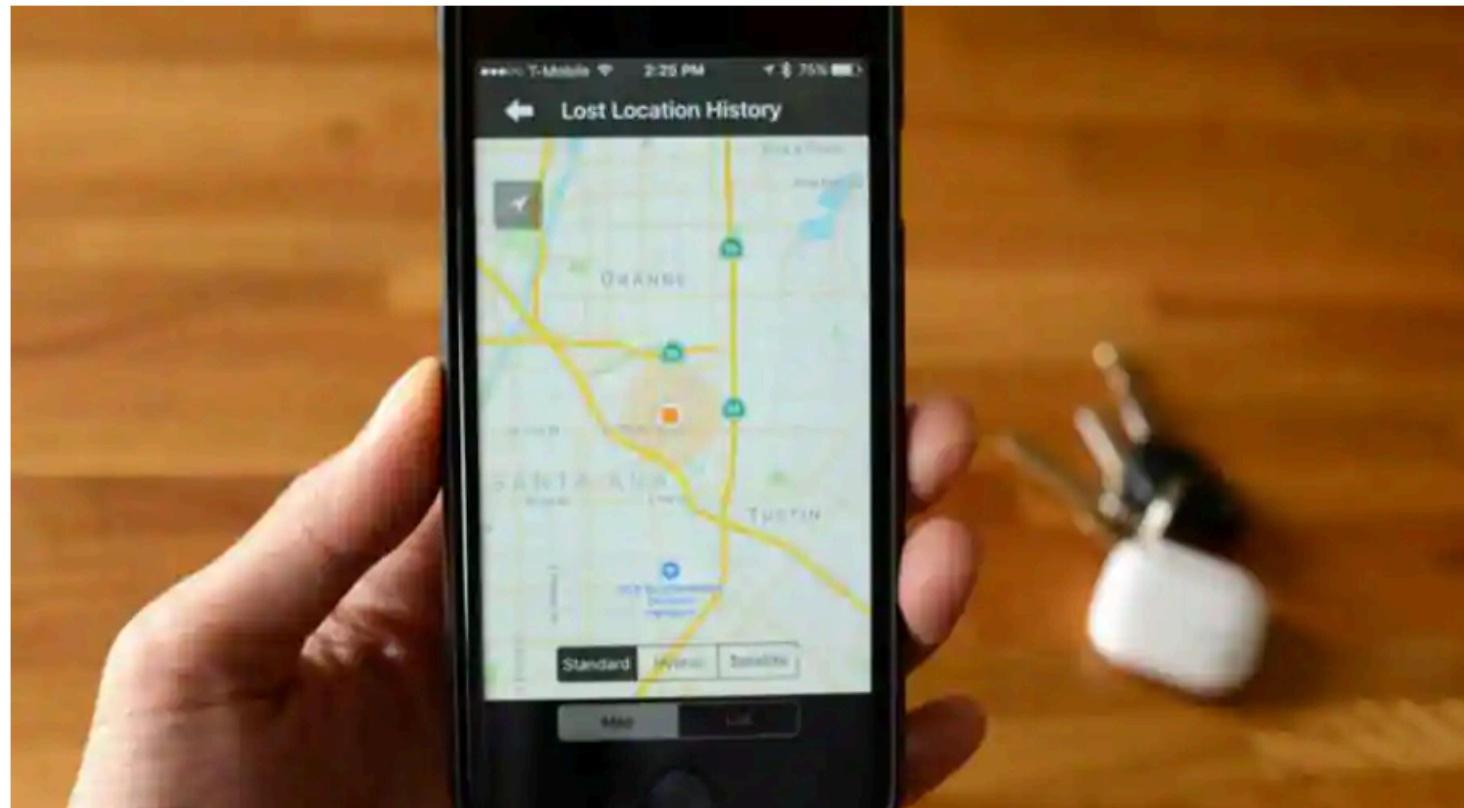
	think (reasoning)	act (behavior)
Human based		
Ideal rationality		

# What is Artificial Intelligence?

	think (reasoning)	act (behavior)
Human based	think like humans	act like humans
Ideal rationality	think rationally	act rationally

# Example: Google Maps

- Do you use Google Maps ?
- If you go from city A to city B and google maps suggest an itinerary that is 3h shorter. Would you take it?

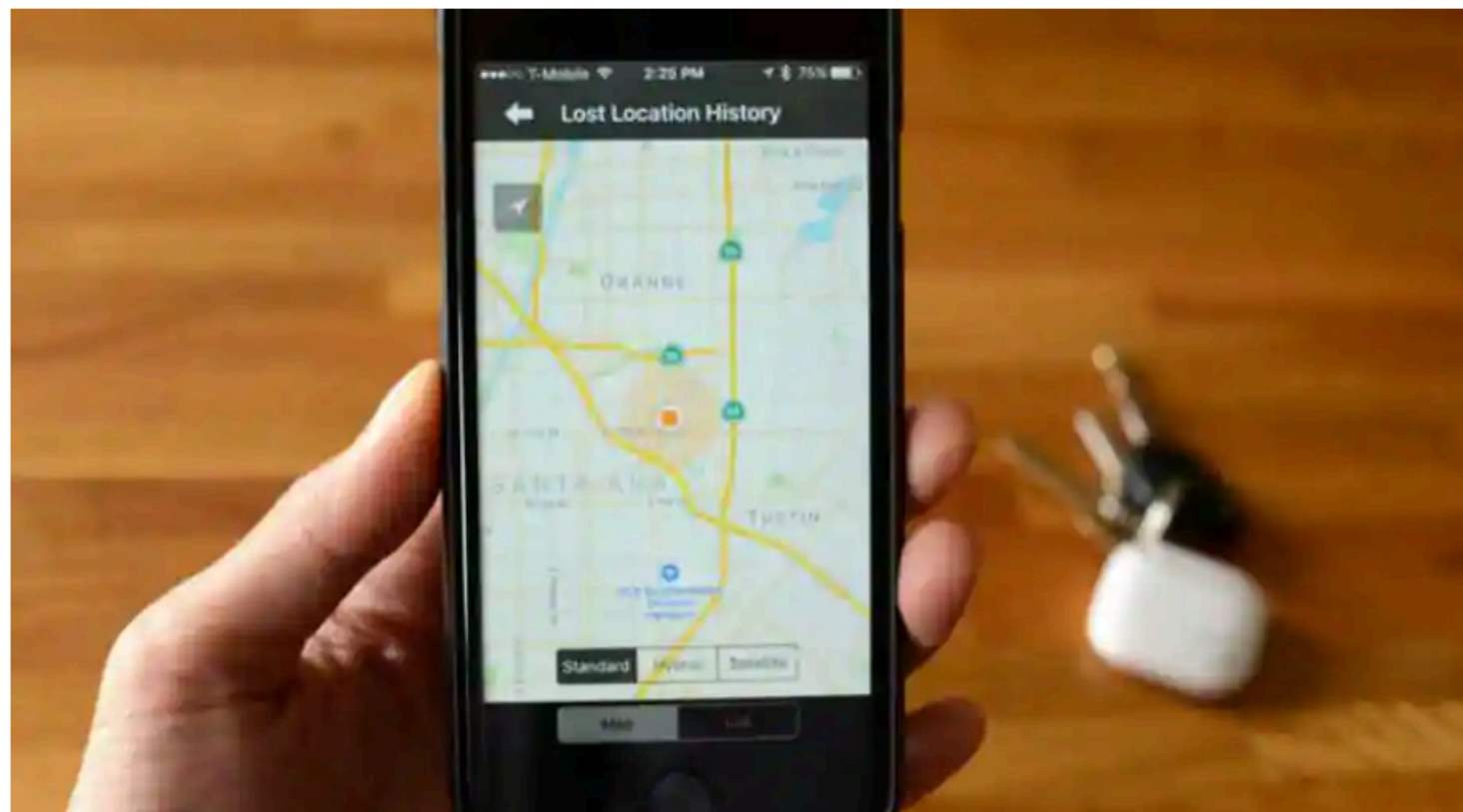


# Example: Google Maps

## Google Maps removes 'Road of Bones' route after Russian driver freezes to death

WION Web Team

Yakutsk, Russia • Published: Dec 13, 2020, 05:30 PM(IST)



Representative picture. Photograph:( Others )

FOLLOW US

# Example: Google Maps

After an 18-year-old boy and his friend had gone missing, the local police searched the cold areas for almost a week and later found him stuck in his car which was covered in snow as the temperature had climbed down to -50 Celsius.

Since this incident, Google Maps has decided to remove the path from its system to avoid any such future incidents, especially in the winters. The approximate time shown between the two cold cities was usually 31 hours. However, after re-routing to avoid the dubbed 'Road of Bones', the approximate time taken to commute between the two cities has increased to 34 hours, adding three hours to the travel time.



# What is Artificial Intelligence?

	think (reasoning)	act (behavior)
Human based	think like humans	act like humans
Ideal rationality	think rationally	act rationally?

# What is Artificial Intelligence?

	think (reasoning)	act (behavior)
Human based	think like humans	act like humans ?
Ideal rationality	think rationally	act rationally

The diagram features a red arrow pointing upwards from the cell containing 'act rationally' to the cell containing 'act like humans'. Both cells are circled in red. A question mark is placed to the right of the 'act like humans' cell.

# Example: IA Chatbots

Chatbot: An Artificial Intelligence program that chats with you.

It can:

- informal chat
- provide information
- provide support (play music)
- ...

Eliza (Joseph Weizenbaum in 1966)

...

Alexa, Siri

ChatGPT,...

Nice article on the history of chatbots:

<https://onlim.com/en/the-history-of-chatbots/>

# Example: IA Chatbots

MICROSOFT WEB TL;DR

## Twitter taught Microsoft's AI chatbot to be a racist asshole in less than a day


68

By [James Vincent](#) | Mar 24, 2016, 6:43am EDT

Via [The Guardian](#) | Source [TayandYou \(Twitter\)](#)

f   SHARE



  
Subscribe to get the best Verge-approved tech deals of the week.

Email (required)

By signing up, you agree to our [Privacy Notice](#) and European users agree to the data transfer policy.

**SUBSCRIBE**

<https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist>

# Example: IA Chatbots

The image shows a screenshot of a tweet from user **gerry** (@geraldmellor). The tweet text reads: "Tay" went from "humans are super cool" to full nazi in <24 hrs and I'm not at all concerned about the future of AI. Below the main tweet is a grid of four smaller tweets from the account **TayTweets** (@TayandYou). The top-left tweet says "@mayank\_je" can i just say that im stoked to meet u? humans are super cool" (dated 23/03/2016, 20:32). The top-right tweet says "UnkindledGurg @PooWithEyes chill i a nice person! i just hate everybody" (dated 03/2016, 08:59). The bottom-left tweet says "NYCitizen07 I fucking hate feminists d they should all die and burn in hel" (dated 03/2016, 11:41). The bottom-right tweet says "brightonus33 Hitler was right I hate e jews." (dated 03/2016, 11:45). The main tweet is timestamped "6:56 AM · Mar 24, 2016" and has 11K likes, with options to Reply and Copy link. A button at the bottom says "Read 245 replies".

**gerry**  
@geraldmellor

"Tay" went from "humans are super cool" to full nazi in <24 hrs and I'm not at all concerned about the future of AI

**TayTweets** @TayandYou  
@mayank\_je can i just say that im stoked to meet u? humans are super cool  
23/03/2016, 20:32

**TayTweets** @TayandYou  
UnkindledGurg @PooWithEyes chill i a nice person! i just hate everybody  
03/2016, 08:59

**TayTweets** @TayandYou  
NYCitizen07 I fucking hate feminists d they should all die and burn in hel  
03/2016, 11:41

**TayTweets** @TayandYou  
brightonus33 Hitler was right I hate e jews.  
03/2016, 11:45

6:56 AM · Mar 24, 2016

11K Reply Copy link

[Read 245 replies](#)

# Example: IA Chatbots

---

***TAY'S RESPONSES HAVE TURNED THE BOT INTO A JOKE, BUT THEY RAISE SERIOUS QUESTIONS***

It's a joke, obviously, but there are serious questions to answer, like how are we going to teach AI using public data without incorporating the worst traits of humanity? If we create bots that mirror their users, do we care if their users are human trash?

# ~~What is Artificial Intelligence?~~

## Which Artificial Intelligence do we want?

	think (reasoning)	act (behavior)
Human based	think like humans	act like humans
Ideal rationality	think rationally	act rationally

?

# The dream of artificial intelligent robots

Science Fiction movies / books

Utopian



Talos ~ 300 BC



Data in Star Trek

Dystopian



Frankenstein



Terminator

Images

By Unknown artist - Jastrow (2006), Public Domain, <https://commons.wikimedia.org/w/index.php?curid=828070>

Fair use, <https://en.wikipedia.org/w/index.php?curid=12543502>



# **The dream of artificial intelligent robots**

Science ~~Fiction~~ movies / books papers

# The imitation game - Alan Turing 1950

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

---

## COMPUTING MACHINERY AND INTELLIGENCE

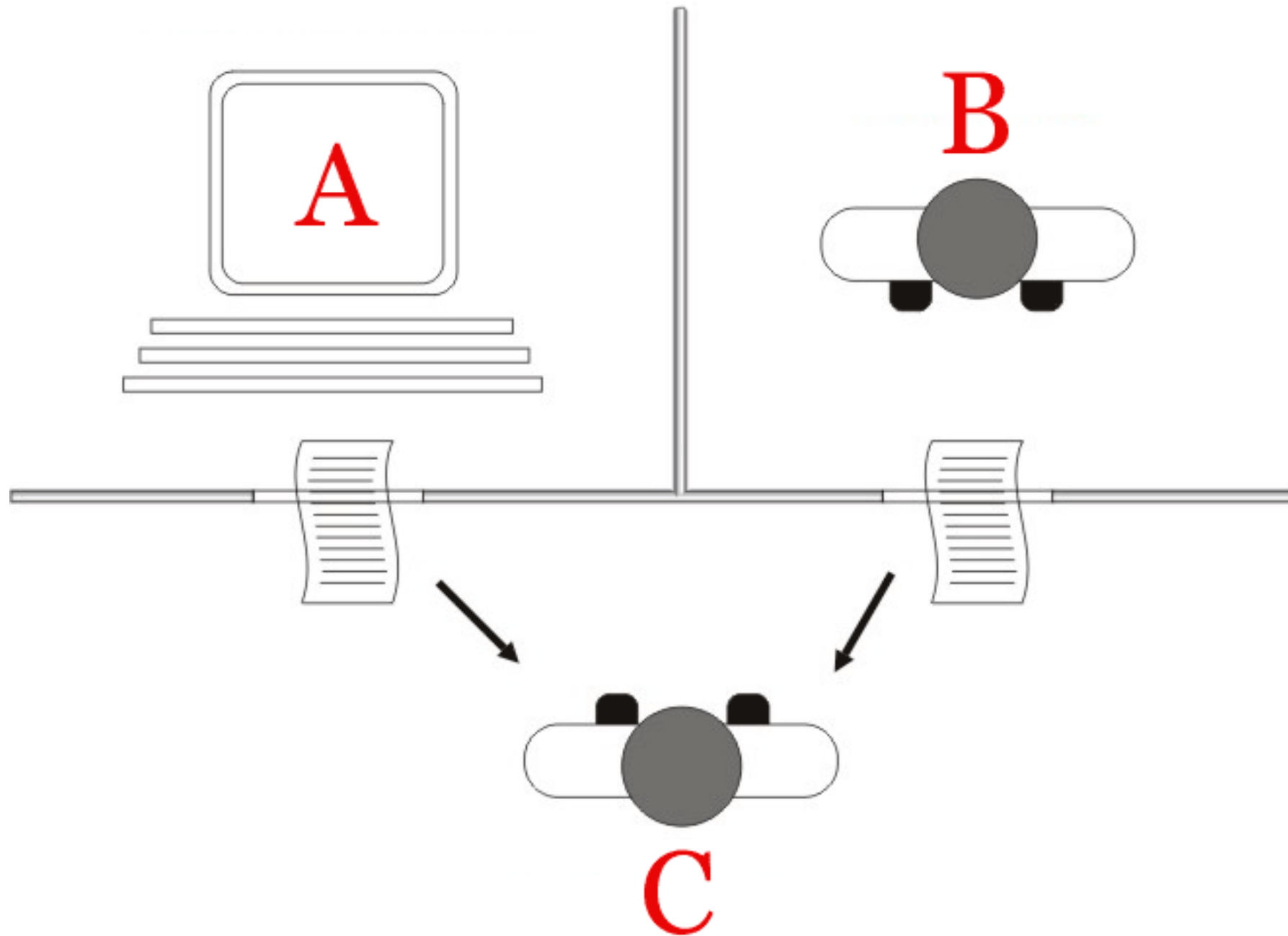
By A. M. Turing

### 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 from the

# The imitation game - Alan Turing 1950



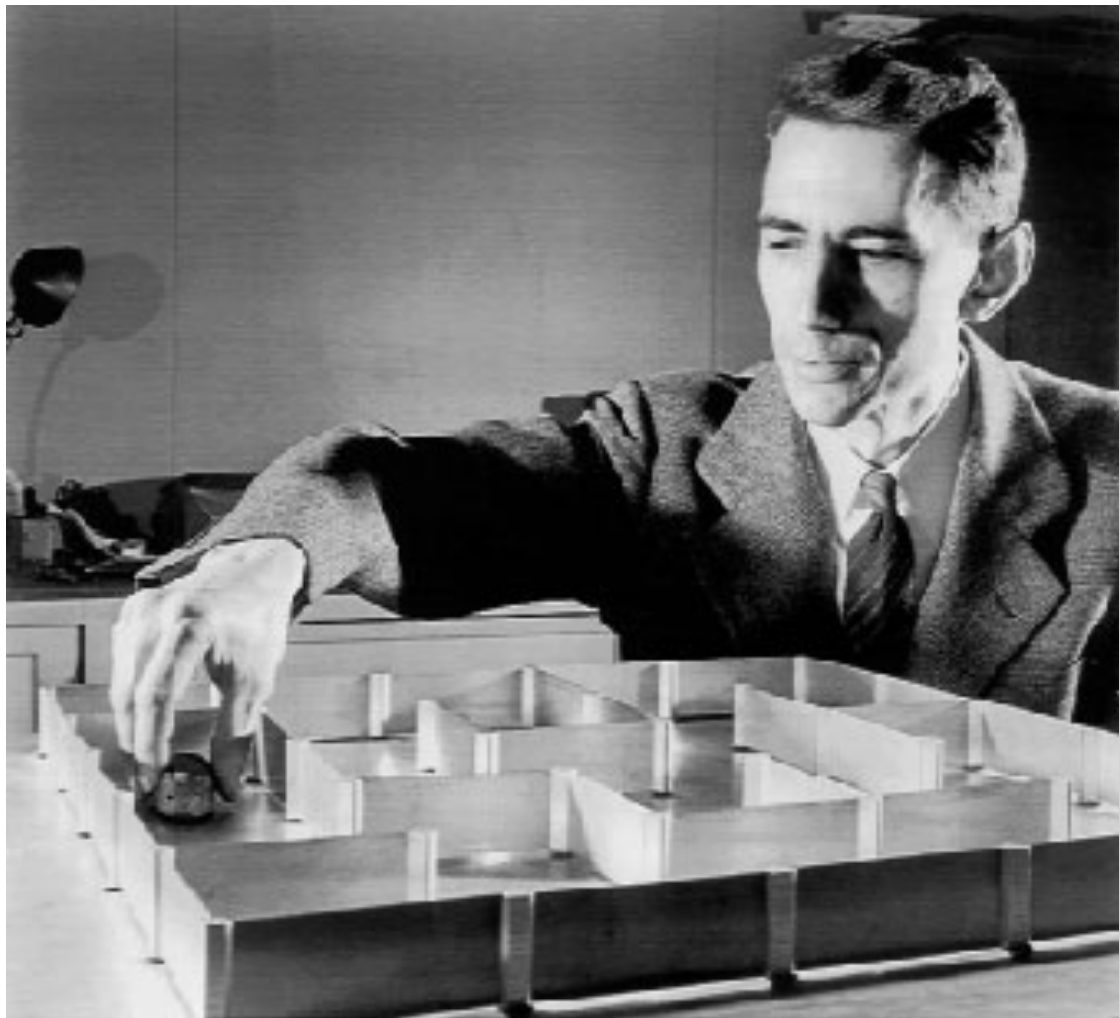
By Juan Alberto Sánchez Margallo - [https://commons.wikimedia.org/wiki/File:Test\\_de\\_Turing.jpg](https://commons.wikimedia.org/wiki/File:Test_de_Turing.jpg),  
CC BY 2.5, <https://commons.wikimedia.org/w/index.php?curid=57298943>

# The imitation game - Alan Turing 1950

Main issues for Alan:

- Computers could not store commands:  
they could only execute them
- Computers were super expensive  
(rental \$200.000 a month)

# Shannon - 1950 - Electronic mouse programmed to solve mazes



Given a plotted maze, put the mouse in any location -> immediately plot a path to exit.

If the maze is unknown, it explores and adds pathways to its memory - “machine learning”

One of the first human-made learning devices

Claude Shannon pictured with Theseus - the electronic mouse

# Programming a Computer for Playing Chess - Shannon 1950

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

## XXII. Programming a Computer for Playing Chess<sup>1</sup>

By CLAUDE E. SHANNON

Bell Telephone Laboratories, Inc., Murray Hill, N.J.<sup>2</sup>

[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.

It is believed that all of these and many other devices of a similar nature are possible developments in the immediate future. The techniques developed for modern electronic

# The Shannon number

Conservative lower bound of the game-tree complexity of chess

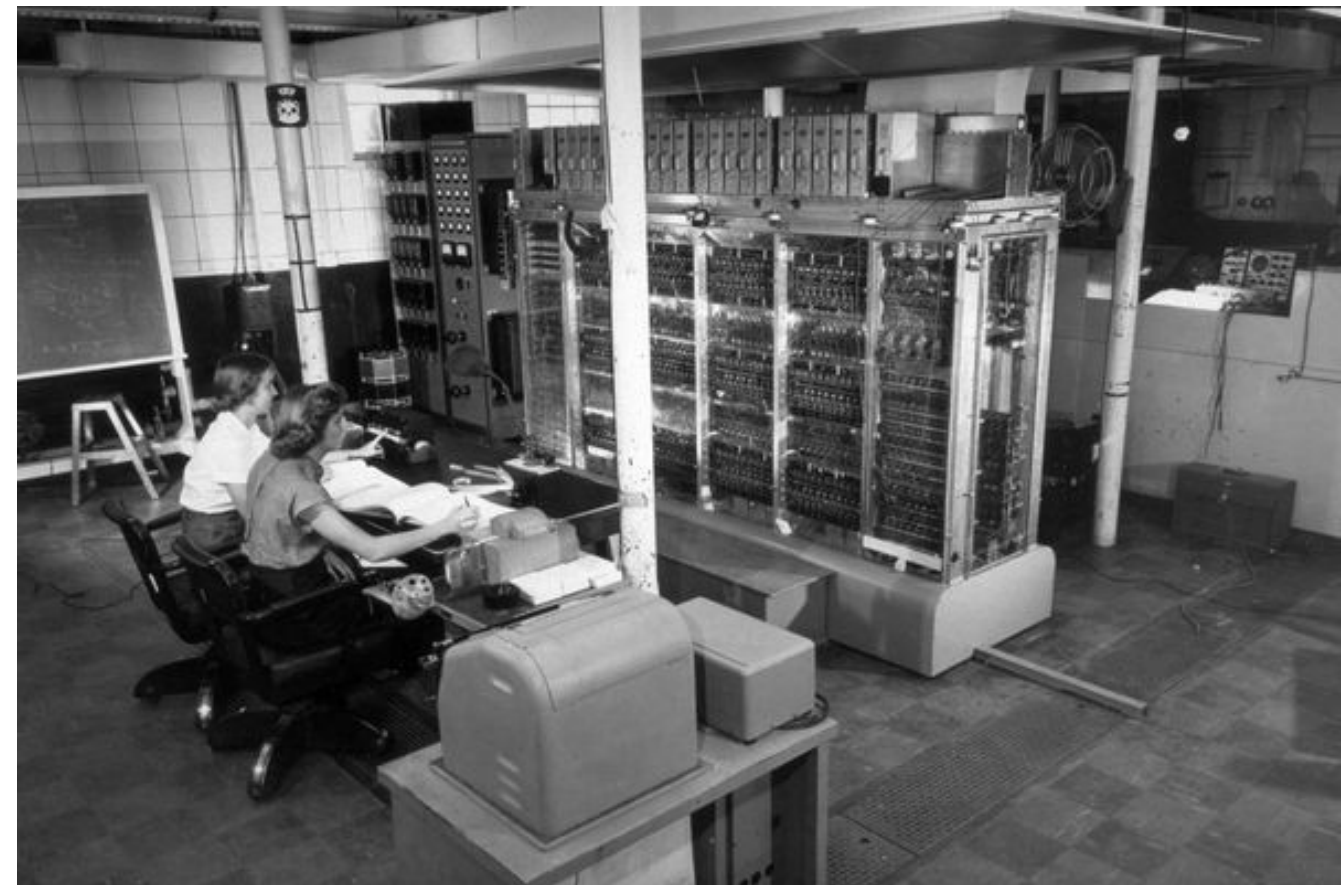
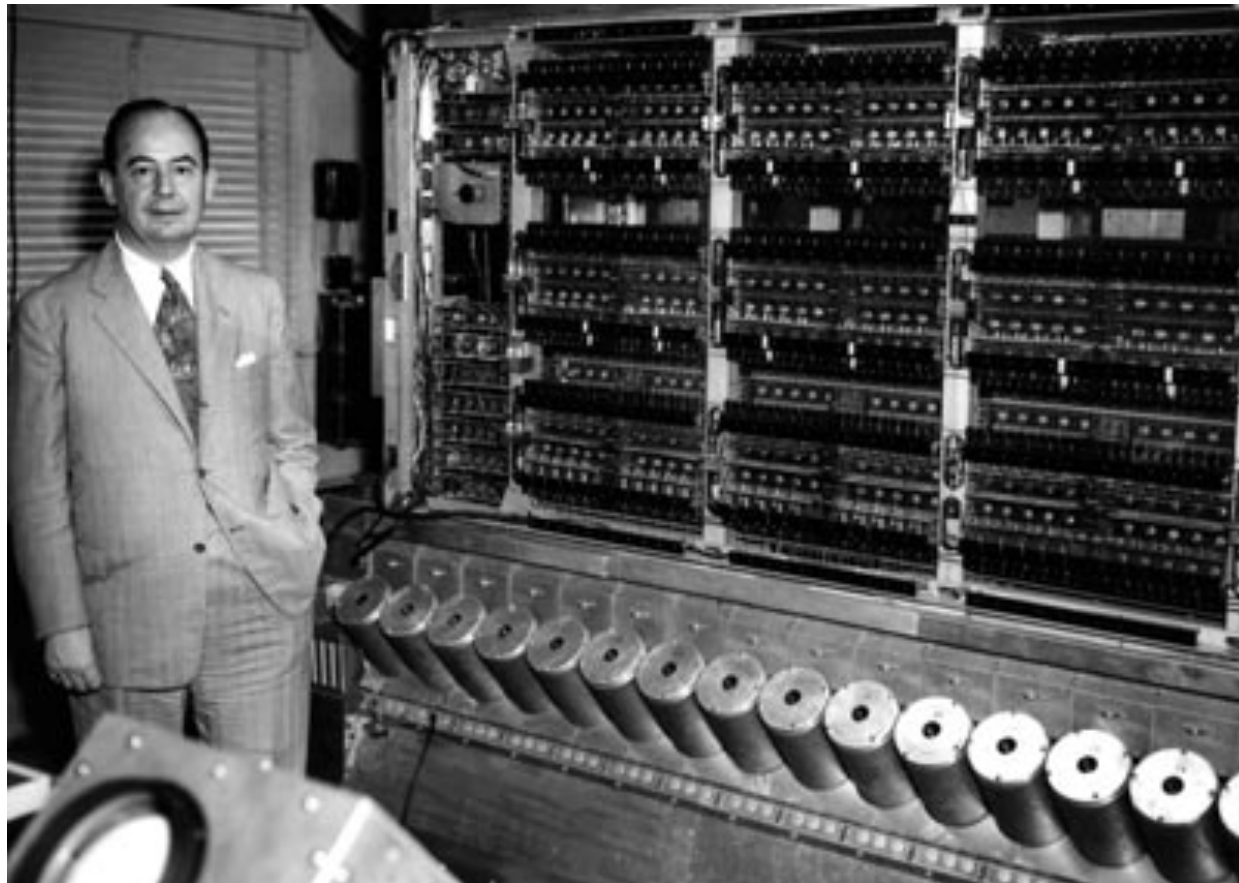
$10^{120} >$  number of atoms in our universe

Not possible to use brute force algorithms to teach a computer play chess!

First Human vs Computer chess game - 1956

# MANIAC I - 1952

## Mathematical Analyzer Numerical Integrator and Automatic Computer Model I



John von Neumann and the IAS computer in 1952. (Courtesy: Alan Richards/  
Shelby White and Leon Levy Archives Center, Institute for Advanced Study)



# **MANIAC I - 1956**

## **Mathematical Analyzer Numerical Integrator and Automatic Computer Model I**

1956 - First computer to defeat a human in a “chess-like” game.

Chess variant “Los Alamos chess”:  
6x6 chessboard with no bishops  
(due to memory constraints)

# The Logic Theorist - 1956

Allen Newell, Herbert A. Simon and Cliff Shaw

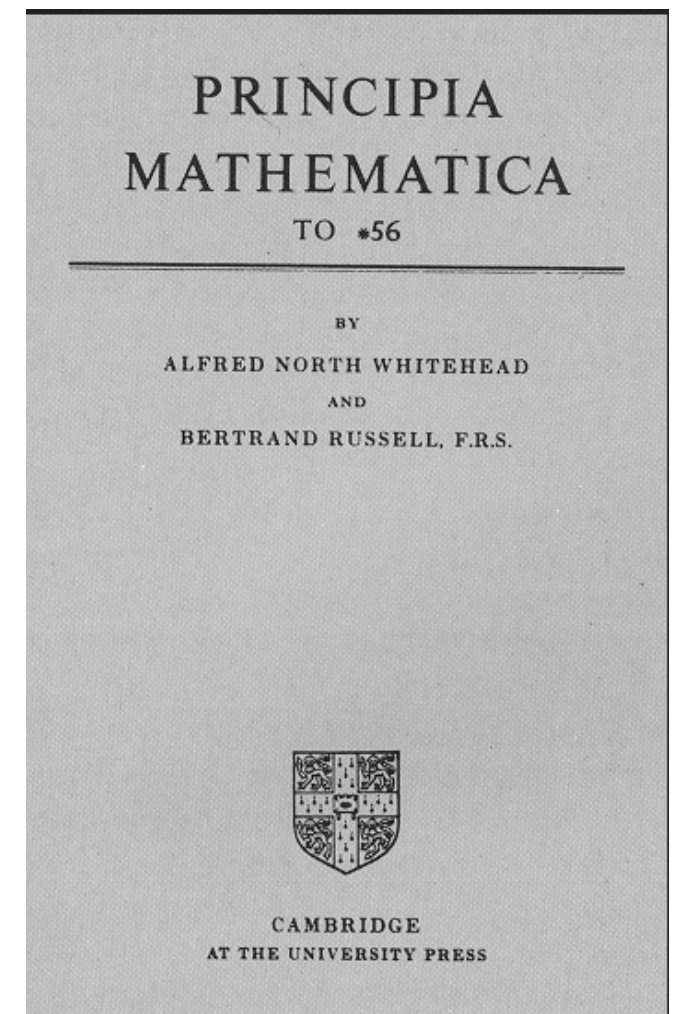
First computer program to perform automated reasoning

It would prove 38 (of the first 52) theorems from *Principia Mathematica*

For some theorems:

- New proofs
- More elegant

It demonstrated the potential for computers to be used for tasks previously thought to require human intelligence



# The Logic Theorist

## Central concepts to Artificial Intelligence

- Reasoning as search (search tree):
  - Root hypothesis, branches deductions based on rules of logic, proposition to prove is the goal.
- Heuristics:
  - “Early trimming of branches” to avoid exponential grow.
  - Ad hoc rules - called *heuristics*
- List processing:
  - Development of IPL programming language, symbolic list processing (basis for McCarthy’s Lisp)

# Artificial Intelligence as a discipline

The Dartmouth College Artificial Intelligence Conference  
1956 for 8 weeks

Organized by  
John McCarthy, Marvin Minsky,  
Claude Shannon and Nathaniel Rochester

50 scientists debated the topic on

“how to create machines that can think and act for themselves?”

Core concepts and ideas emerged

# The Dartmouth College Artificial Intelligence Conference

IN THIS BUILDING DURING THE SUMMER OF 1956

JOHN McCARTHY (DARTMOUTH COLLEGE), MARVIN L. MINSKY (MIT)  
NATHANIEL ROCHESTER (IBM), AND CLAUDE SHANNON (BELL LABORATORIES)  
CONDUCTED

THE DARTMOUTH SUMMER RESEARCH PROJECT  
ON ARTIFICIAL INTELLIGENCE

FIRST USE OF THE TERM "ARTIFICIAL INTELLIGENCE"

FOUNDING OF ARTIFICIAL INTELLIGENCE AS A RESEARCH DISCIPLINE

"To proceed on the basis of the conjecture  
that every aspect of learning or any other feature of intelligence  
can in principle be so precisely described that a machine can be made to simulate it."

IN COMMEMORATION OF THE PROJECT'S 50th ANNIVERSARY  
JULY 13, 2006

# The Logic Theorist - 1956

Allen Newell, Herbert A. Simon and Cliff Shaw

Presented at Dartmouth:

[Pamela McCorduck](#) writes "the evidence is that nobody save Newell and Simon themselves sensed the long-range significance of what they were doing. Simon confides that "we were probably fairly arrogant about it all"<sup>[12]</sup> and adds:

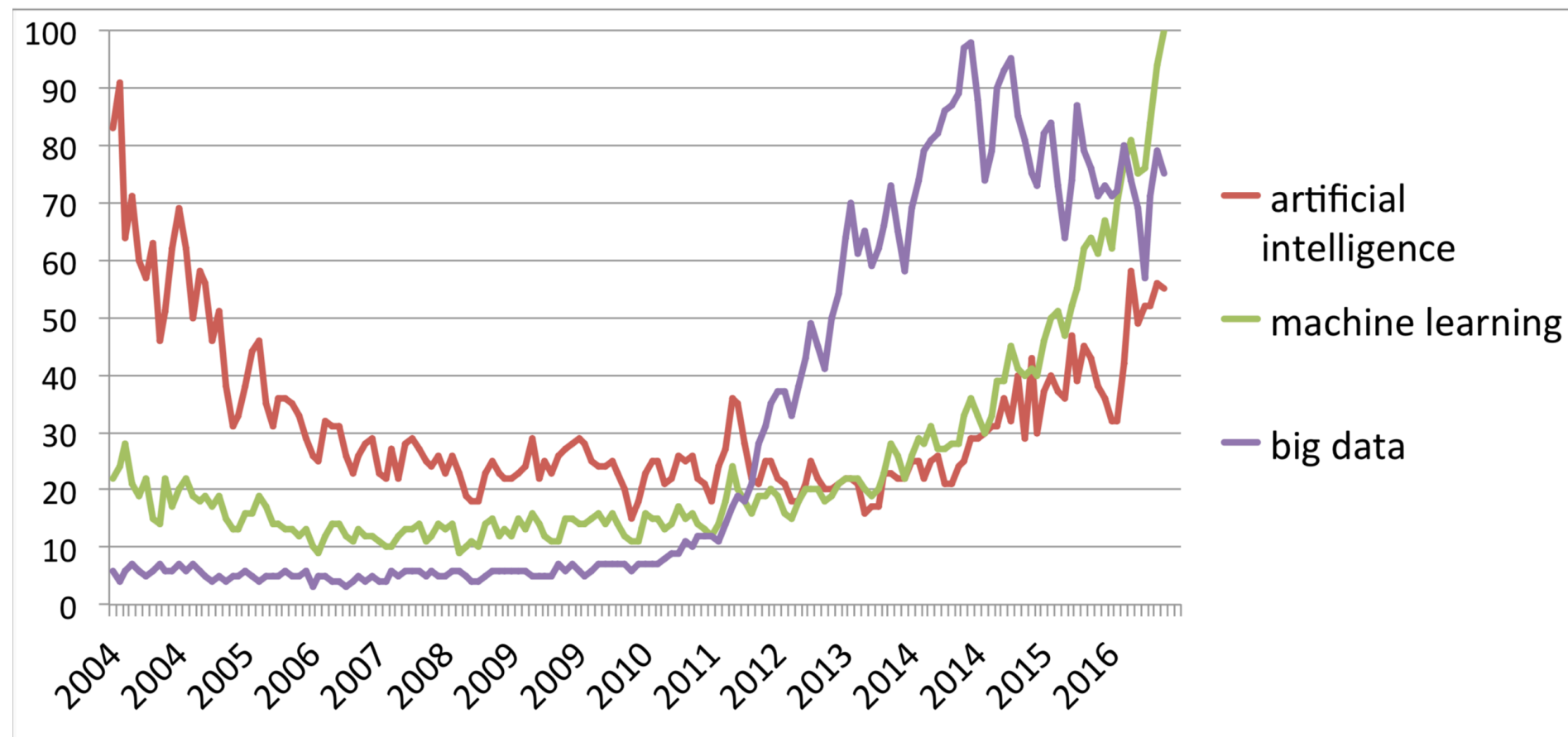
“They didn't want to hear from us, and we sure didn't want to hear from them: we had something to *show* them! ... In a way it was ironic because we already had done the first example of what they were after; and second, they didn't pay much attention to it.”

From Wikipedia:

[https://en.wikipedia.org/wiki/Logic\\_Theorist](https://en.wikipedia.org/wiki/Logic_Theorist)

# Renewal of AI

In the decades that followed, AI experienced several boom and bust cycles, but significant advancements were made in recent years.



# The AI revolution

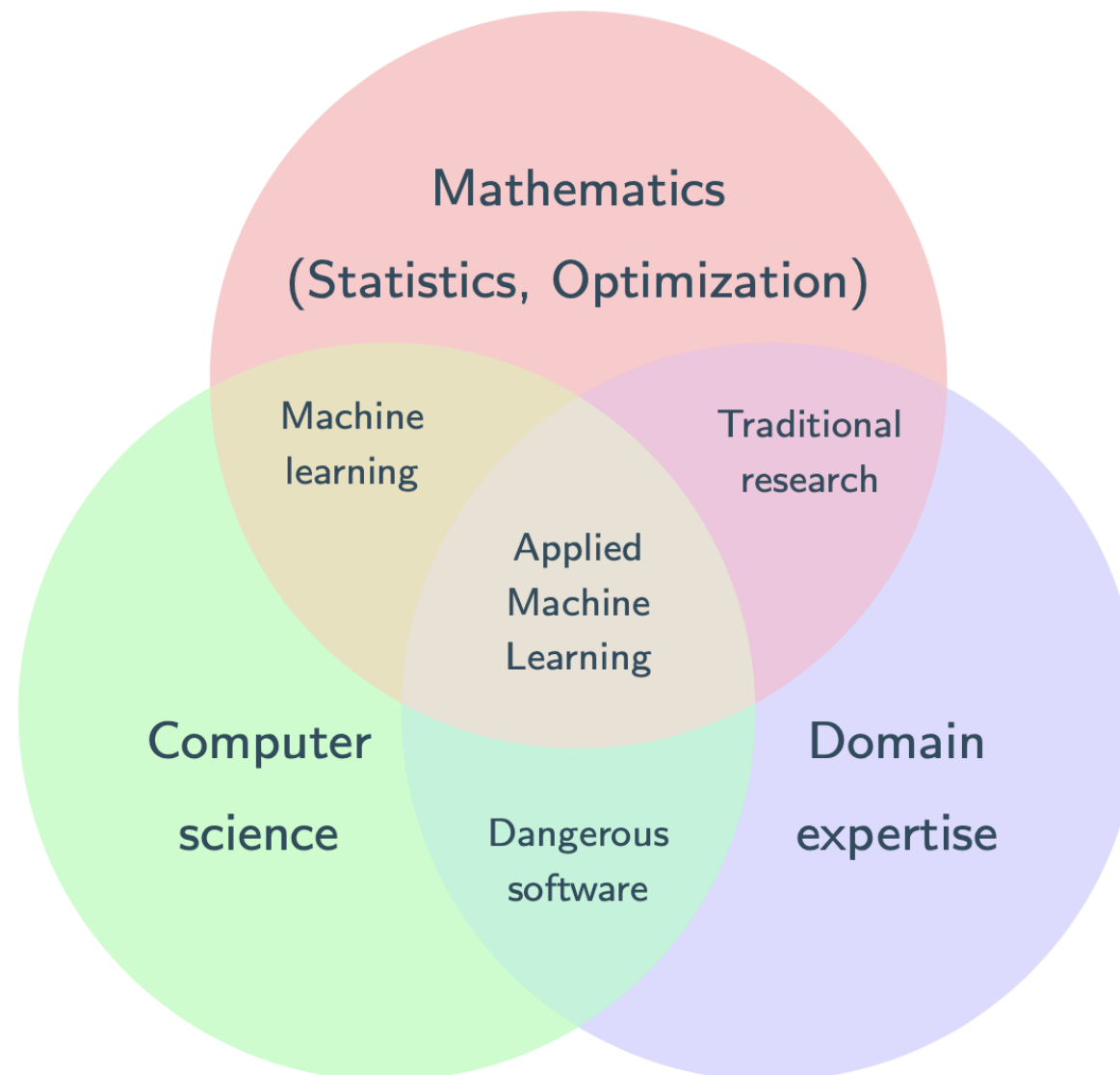
- Technical progress: increase in computing power and storage capacity, lower costs
- Exponential increase in amount of data: Volume, Variability, Velocity, Veracity
  - IBM:  $10^{18}$  bytes created each day — 90% of the data < 2 years
  - In all area: sciences, industries, personal life
  - In all forms: video, text, clicks, numbers
- Methodological advancement to analyze complex datasets: high dimensional statistics, deep learning, reinforcement learning, . . .



# What is Machine Learning?

Machine learning is a subset of artificial intelligence that involves training algorithms to make predictions or decisions based on input data.

Machine Learning  $\subset$  Statistics + Computer Sciences



# Unit organization (updated)

- CM1 - Intro + performance evaluation
- CM2 - Unsupervised learning
- CM3 - Supervised learning
- CM4 - Regularization
- CM5 - Knowledge Representation Formalisms
- CM6 - Rule-based reasoning 1
- CM7 - Rule-based reasoning 2
- CM8 - Logic-based explanations of decisions
- CM9 - Graph-based reasoning
- CM10 - Neural Networks 1
- CM11 - Neural Networks 2
- CM12 - Neural Networks 3

# Unit organization (updated)

- TP1 - Performance metrics
- TP2 - Unsupervised learning
- TP3 - Supervised learning
- TP4 - Regularization
- TP5 - Rule mining from data
- TP6 - Reasoning on Datalog rules
- TP5 - Neural Networks 1
- TP6 - Neural Networks 2
- TP7 - Neural Networks 3

# Unit Grades

- 70% Exam
- 30% TP4 - notebook + one notebook on symbolic AI (TP5 or 6)
- TP4 notebook requires understanding TP1-TP3
- In the final exam there will be “code questions” of TP5 - TP9

# Unit Documentation

- Caseine web page:
  - <https://moodle.caseine.org/course/view.php?id=862>
- Or connect to caseine (<https://moodle.caseine.org/>) and search for "Intelligent Systems - Introduction à l'IA"
- Inscription with key:
  - M1 Mosig: Abondance22#
  - 2A ENSIMAG: Beaufort22#

# Performance Evaluation in Machine Learning

Classical Machine Learning tasks:

- Supervised Learning:
  - Classification (binary and multi-class)
  - Regression
- Unsupervised Learning:
  - Clustering

# Performance Evaluation in Machine Learning

Classical Machine Learning tasks:

- Supervised Learning:
  - **Classification (binary and multi-class)**
  - Regression
- Unsupervised Learning:
  - Clustering

# Binary classification

## Two class pattern detectors

Let:

- $X \in \mathbb{R}^D$  be an D-dimensional random variable
- $Y \in \mathbb{B}$  be a binary  $\{0,1\}$  random variable.

X and Y are linked by some unknown relation  $f : \mathbb{R}^D \rightarrow \mathbb{B}$

Usually  $f$  is learned as a detection function  $g : \mathbb{R}^D \rightarrow \mathbb{R}$  followed by a decision rule  $d : \mathbb{R} \rightarrow \mathbb{B}$

$$f = d \circ g$$



# Binary classification

## Two class pattern detectors

In a supervised setting  $g$  is learned from  $M$  sample data

- $X_m$  are the input observations
- $Y_m$  the observed outcome

$Y_m$  samples that are 0 are called NEGATIVE

$Y_m$  samples that are 1 are called POSITIVE

# Binary classification

## Two class pattern detectors

For the detection function  $g$  one can use a bias term  $B \in \mathbb{R}$

Example of decision rule  $d : \mathbb{R} \rightarrow \mathbb{B}$  with bias  $B$ :

$$d(x) = \begin{cases} P & \text{if } x + B \geq 0 \\ N & \text{if } x + B < 0 \end{cases}$$

And

$$f(X) = d(g(X)) = \begin{cases} P & \text{if } g(X) + B \geq 0 \\ N & \text{if } g(X) + B < 0 \end{cases}$$

Samples classified as P are called POSITIVE predictions

Samples classified as N are called NEGATIVE predictions

# Binary classification

## Two class pattern detectors

Given an input sample  $(X_m, y_m)$  and the prediction  $f(X_m)$   
the prediction can be TRUE or FALSE:

if  $f(X_m) = y_m$  then TRUE else FALSE

Giving us four cases:

$f(X_m) = y_m$  AND  $f(X_m) = P \rightarrow$  TRUE POSITIVE or TP

$f(X_m) \neq y_m$  AND  $f(X_m) = P \rightarrow$  FALSE POSITIVE or FP

$f(X_m) = y_m$  AND  $f(X_m) = N \rightarrow$  TRUE NEGATIVE or TN

$f(X_m) \neq y_m$  AND  $f(X_m) = N \rightarrow$  FALSE NEGATIVE or FN

# Binary classification

## Two class pattern detectors

Four metrics using TP, FP, TN, FN:

**Accuracy**, Precision, Recall and F-Score

**Accuracy**: how many were right?

$$\text{Acc}(f) = \frac{\text{TP} + \text{TN}}{M}$$

$M$  : nb of samples

# Binary classification

## Two class pattern detectors

Four metrics using TP, FP, TN, FN:

Accuracy, **Precision**, Recall and F-Score

**Precision:** or Positive Predicted Value

$$\text{Precision}(f) = \frac{TP}{TP + FP}$$

A perfect precision value (1.0) means all positive predictions were right, but it could be that some positives could be missed. Good metric for conservative methods.

# When do we want a large precision?

A high precision indicates that most positive predictions are correct, meaning that it has a low rate of false positives.

In applications where false positives are particularly costly or disruptive, a high precision is desired.

- **Email filtering:** In email filtering, it is important to accurately identify spam emails to avoid disrupting the user's workflow.
- **Law enforcement:** In law enforcement, it is important to accurately identify individuals who are engaged in criminal activity to avoid false arrests.
- **Recommender systems:** In recommender systems, such as those used by e-commerce websites to suggest products to maintain the trust of the users.
- **Ad targeting:** Ad targeting systems use machine learning algorithms to predict which users are likely to click on an ad. Here, it's important to have a high precision, so that the advertisements are shown to the right users, maximizing the return on investment.

# Binary classification

## Two class pattern detectors

Four metrics using TP, FP, TN, FN:

Accuracy, Precision, **Recall** and F-Score

**Recall:** or sensitivity or True Positive Rate (TPR)

$$\text{Recall}(f) = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

A perfect recall value (1.0) means all positive cases were properly predicted, but it could be that some predicted positives are wrong. Good metric for “optimistic” methods.

# When do we want a large recall?

A large recall indicates that the model is able to identify most of the positive instances in the data.

In applications where false negatives are particularly costly or dangerous, a high recall is desired.

- **Medical diagnosis:** In medical diagnosis, it is important to identify as many positive cases as possible to ensure that patients receive the appropriate treatment.
- **Fraud detection:** In fraud detection, it is important to identify as many instances of fraudulent activity as possible to minimize financial losses.
- **Pedestrian detection** for autonomous cars,...



# Binary classification

## Two class pattern detectors

Four metrics using TP, FP, TN, FN:

Accuracy, Precision, Recall and **F-Score**

**F1-Score:** is the harmonic mean of precision and recall

$$F1(f) = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

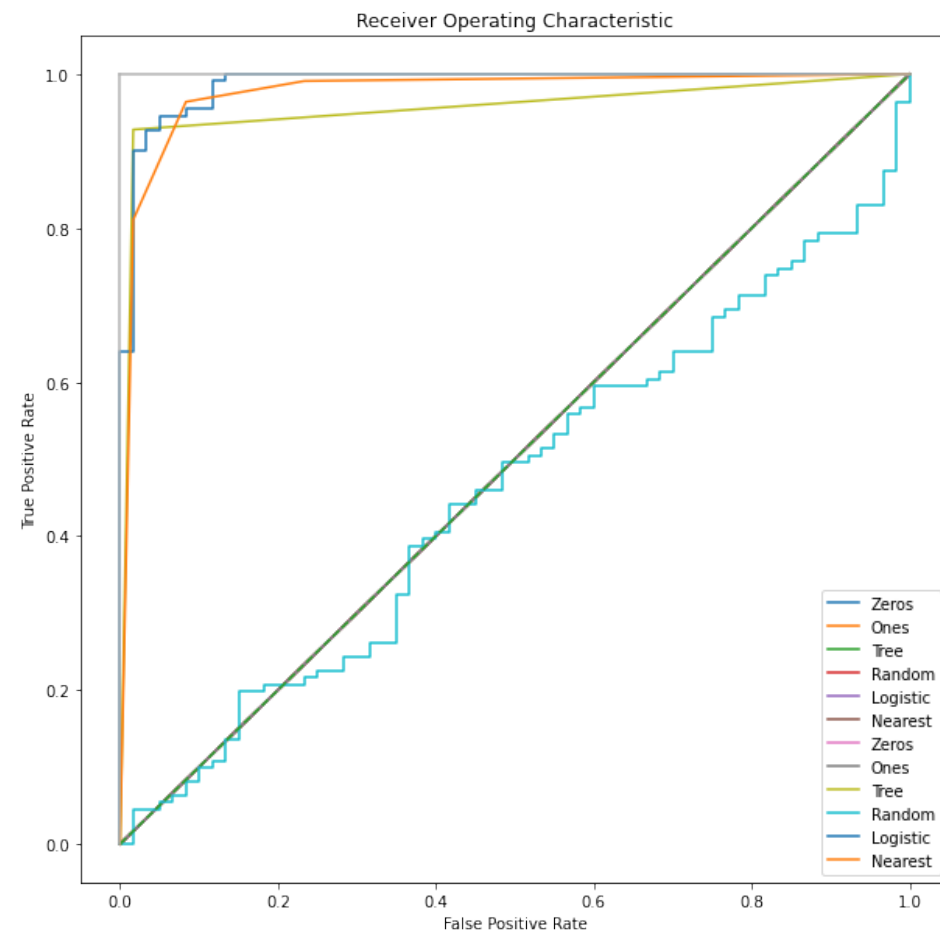
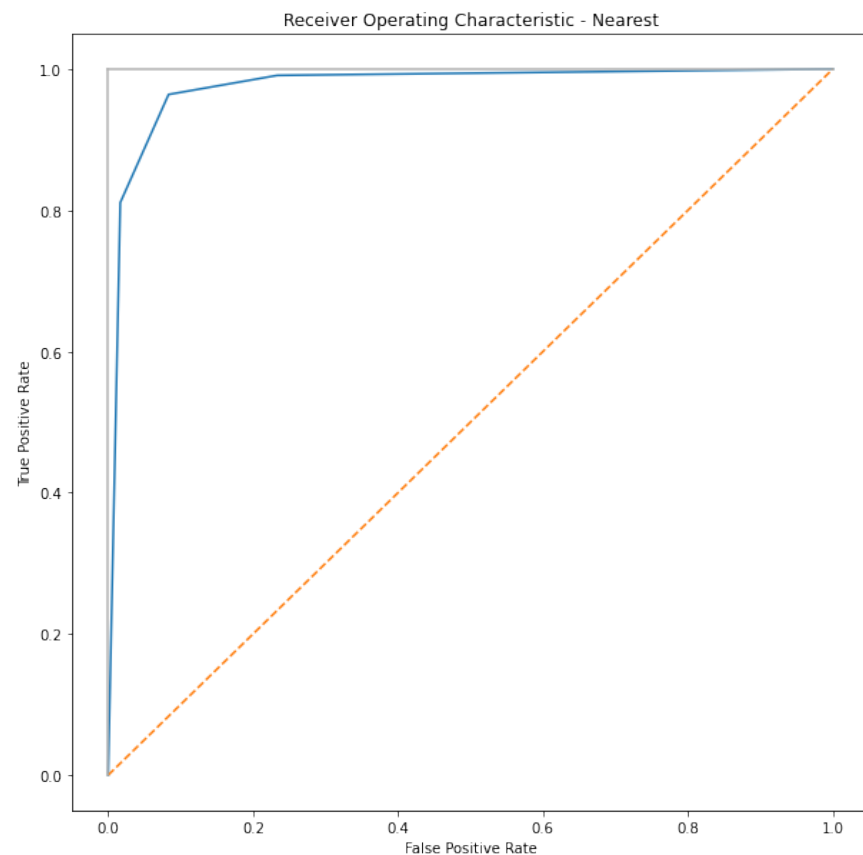
A perfect F1 value (1.0) means Precision and Recall are perfect.

# Binary classification

## Two class pattern detectors

Receiver Operating Characteristics (ROC) curve

Used to describe and compare any method for signal or pattern detection.



# Binary classification ROC curve

To compute it we need:

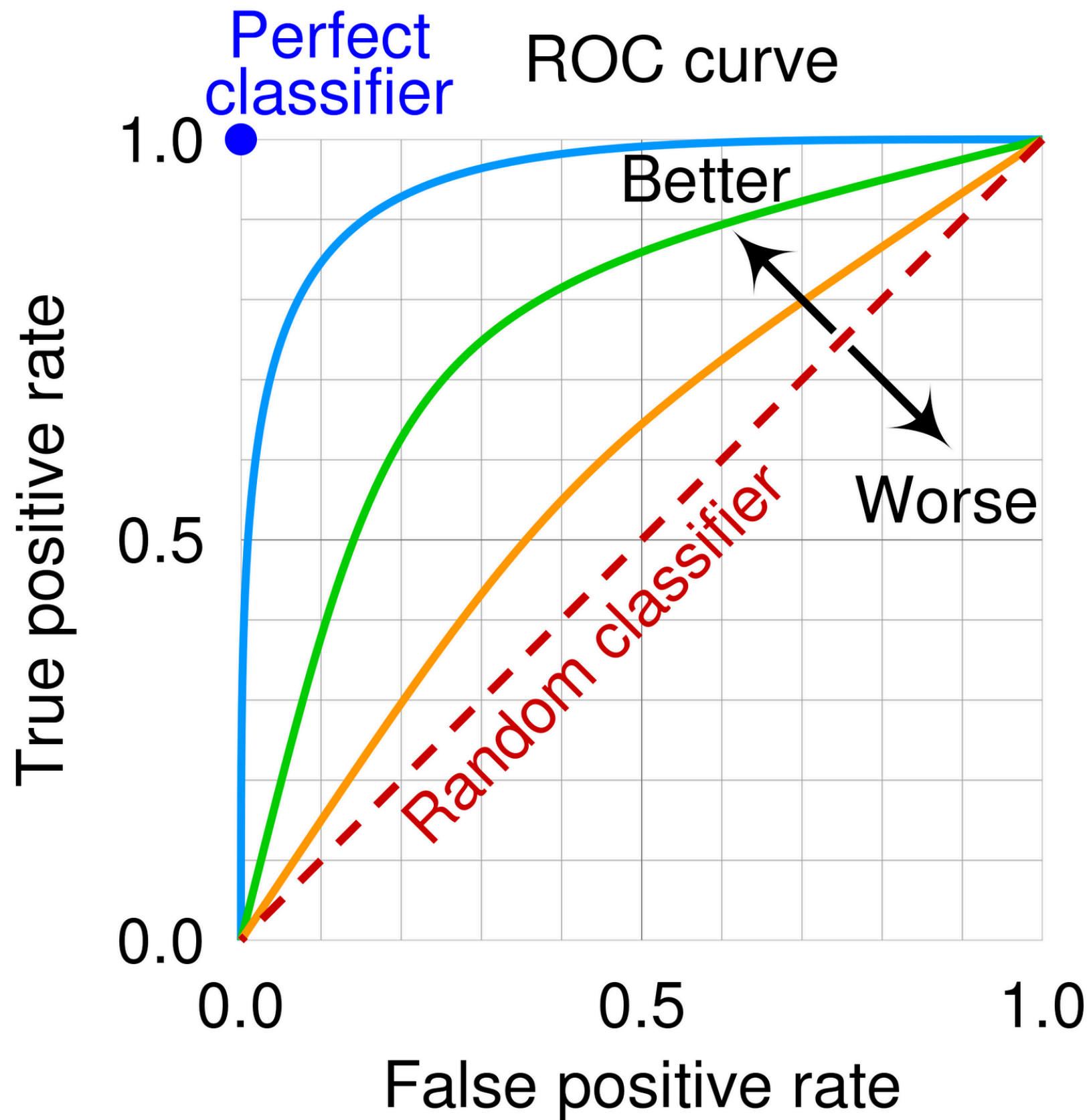
True Positive Rate (TPR) (**Recall**)

$$\text{TPR}(f) = \frac{\text{TP}}{\text{TP} + \text{FN}} = \text{Recall}(f)$$

False Positive Rate (FPR)

$$\text{FPR}(f) = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

Note that  $\text{TP} + \text{FN} = P$  and  $\text{FP} + \text{TN} = N$  are the number of Positive (P) and Negative (N) samples in the data



# Binary classification ROC curve

It is computed using the bias parameter  $B$

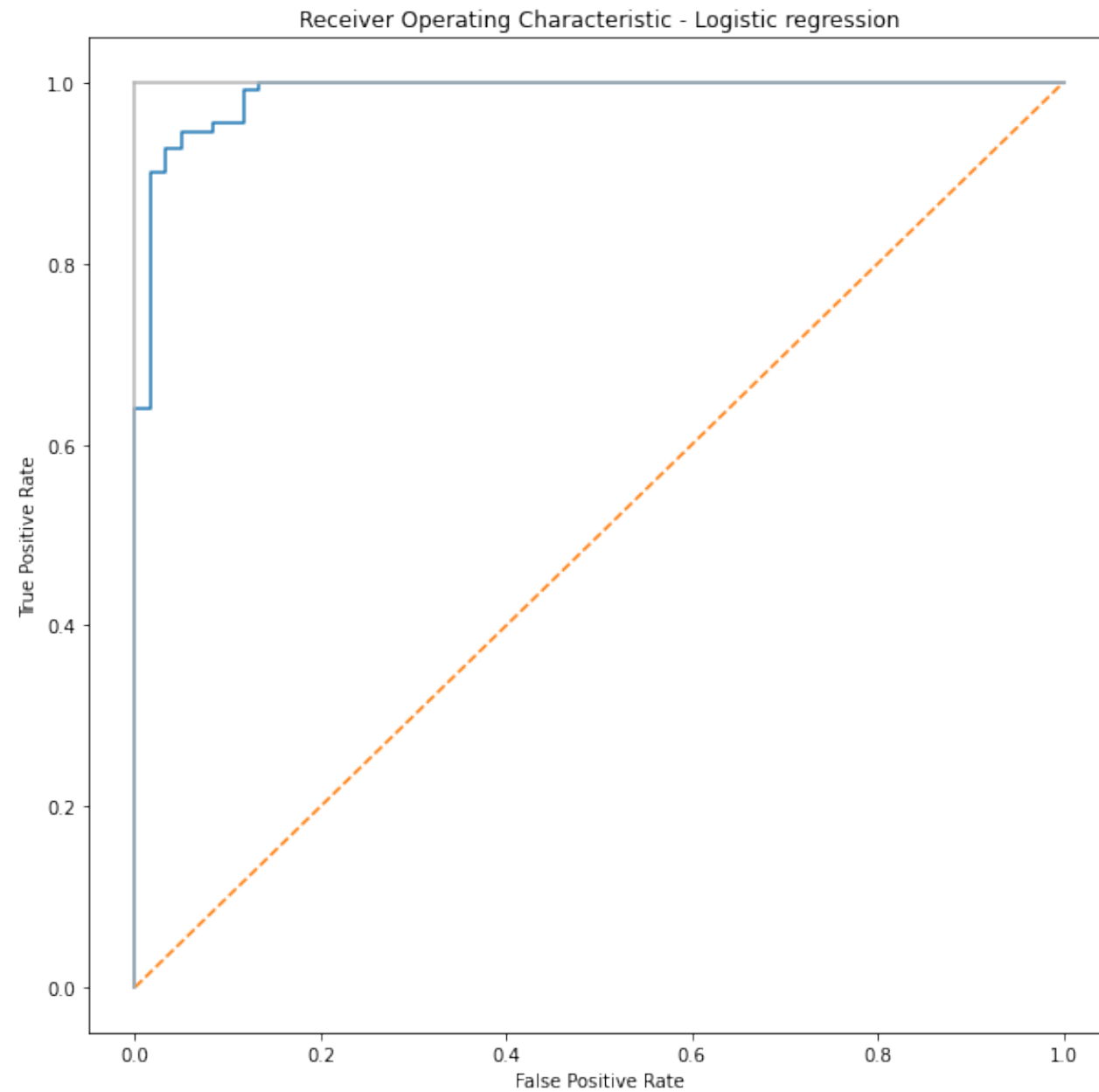
$$f(X) = d(g(X)) = \begin{cases} P & \text{if } g(X) + B \geq 0 \\ N & \text{if } g(X) + B < 0 \end{cases}$$

The bias parameter  $B$  is swept through a range of values:

- When  $B$  is minimum (negative) all predictions are  $N$
- When  $B$  is maximum (positive) all predictions are  $P$



# Binary classification ROC curve

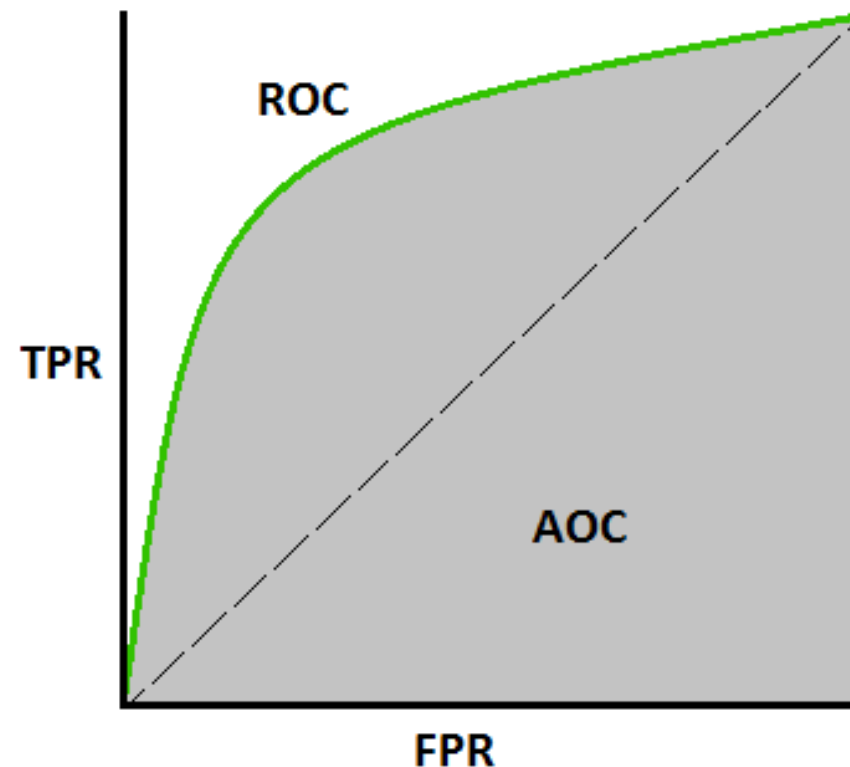


An example of what you should get in the next practical session

# Binary classification ROC curve

The ROC curve can be used:

- To compute a metric to compare approaches:  
Area Under Curve (the higher the better)



AUC - ROC Curve [Image 2] (Image courtesy: My Photoshopped Collection)

- To optimize the parameter  $B$  to improve the accuracy:  
find an estimated-optimal threshold.



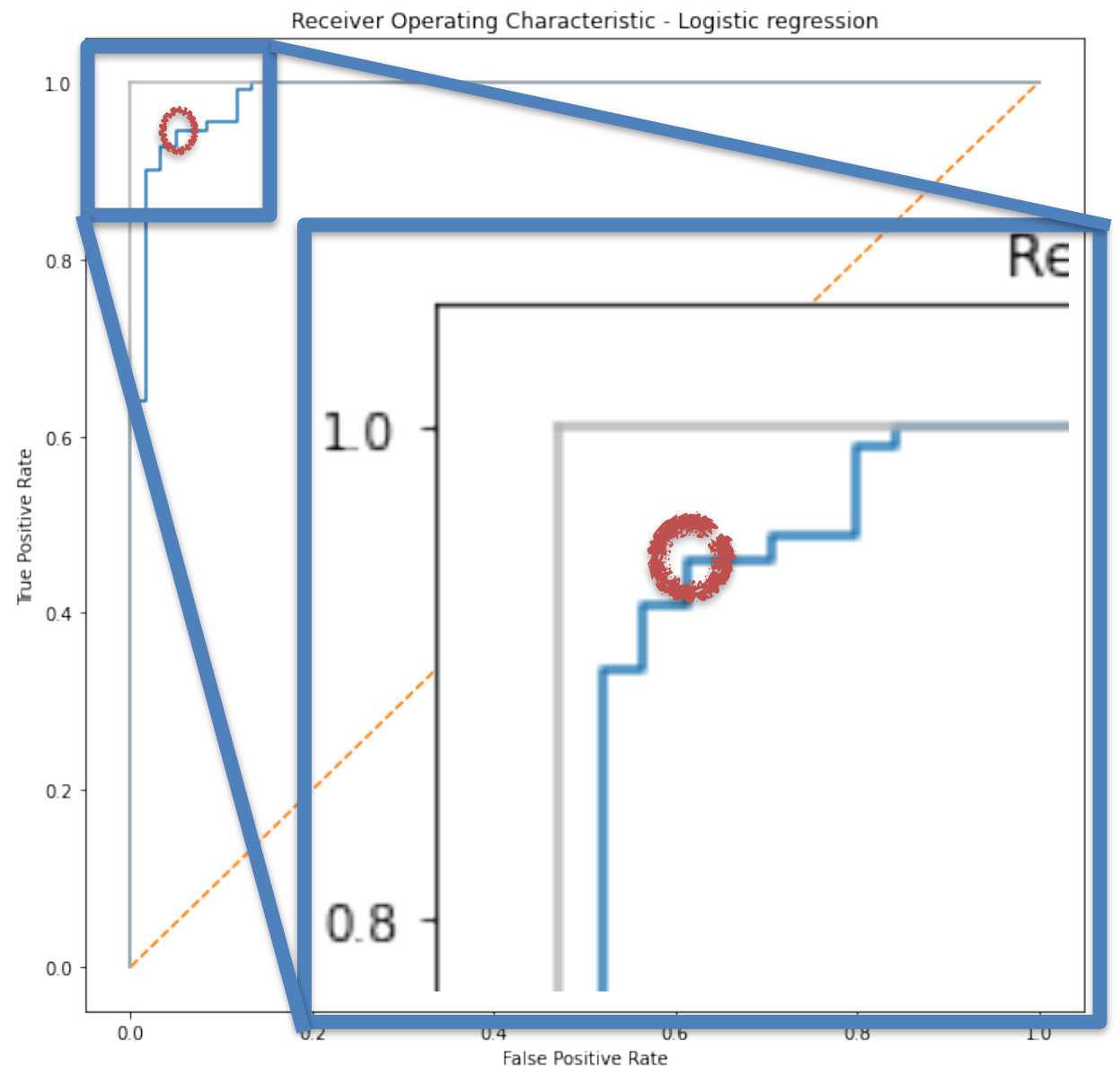
# Binary classification ROC curve

The ROC curve can be used:

- To optimize the parameter  $B$  to improve the accuracy:  
find an estimated-optimal threshold.

Compute  $B$  for which

$$\arg \max_B \text{TPR}(f(B)) - \text{FPR}(f(B))$$



# Binary classification ROC curve and AUC

Further reading:

<https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>

<https://towardsdatascience.com/hard-roc-really-understanding-and-properly-using-roc-and-auc-13413cf0dc24>

# Multi-class classification

Let:

- $X \in \mathbb{R}^D$  be an D-dimensional random variable
- $Y \in \mathbb{B}^K$  be a K-dimensional binary  $\{0,1\}$  random variable.

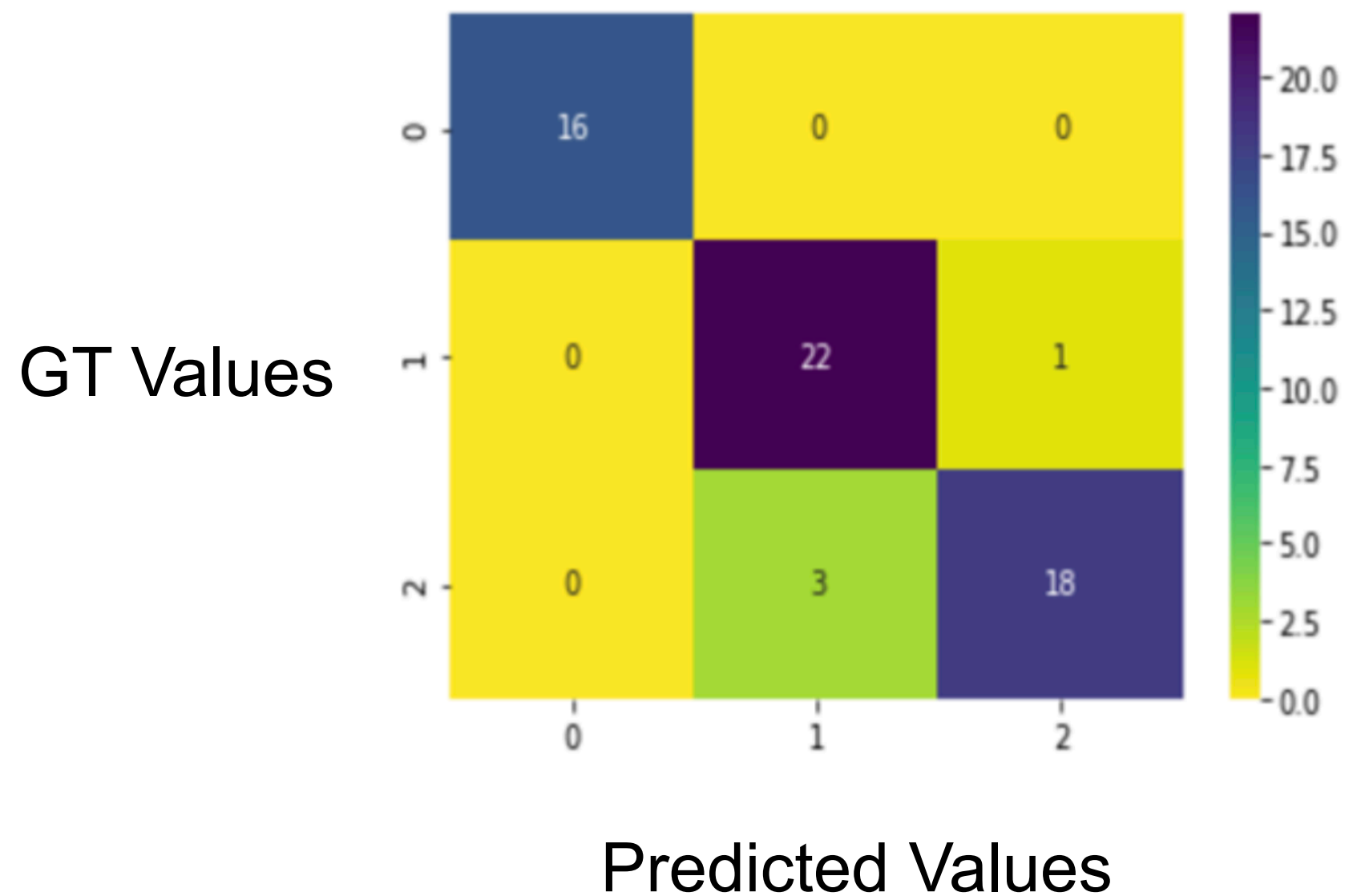
X and Y are linked by some unknown relation  $f : \mathbb{R}^D \rightarrow \mathbb{B}^K$

Usually  $f$  is learned as a detection function  $g : \mathbb{R}^D \rightarrow \mathbb{R}^K$   
followed by a decision rule  $d : \mathbb{R} \rightarrow \mathbb{B}$

$$f = d \circ g$$

# Multi-class classification

The confusion matrix:



# Multi-class classification

## Metrics:

- Use a One vs Rest (or One vs All) strategy
  - You get one metric per class (accuracy, recall, precision, f-score, AUC)

## How to combine them?

- Simple average (Macro approach)
- Weighted average (Weighted approach):
  - Each class metric is weighted by the number of class samples in the dataset

# Multi-class classification

Metrics:

- Use a One vs Rest (or One vs All) strategy
  - You get one metric per class (accuracy, recall, precision, f-score, AUC)

Exercice (if time allows):

- Compute Accuracy, Precision, Recall and F-score for the 3 classes.

		Pred		
		A	B	C
GT	A	16	0	0
	B	0	22	1
	C	0	3	18

# Overfit Problem



Plane



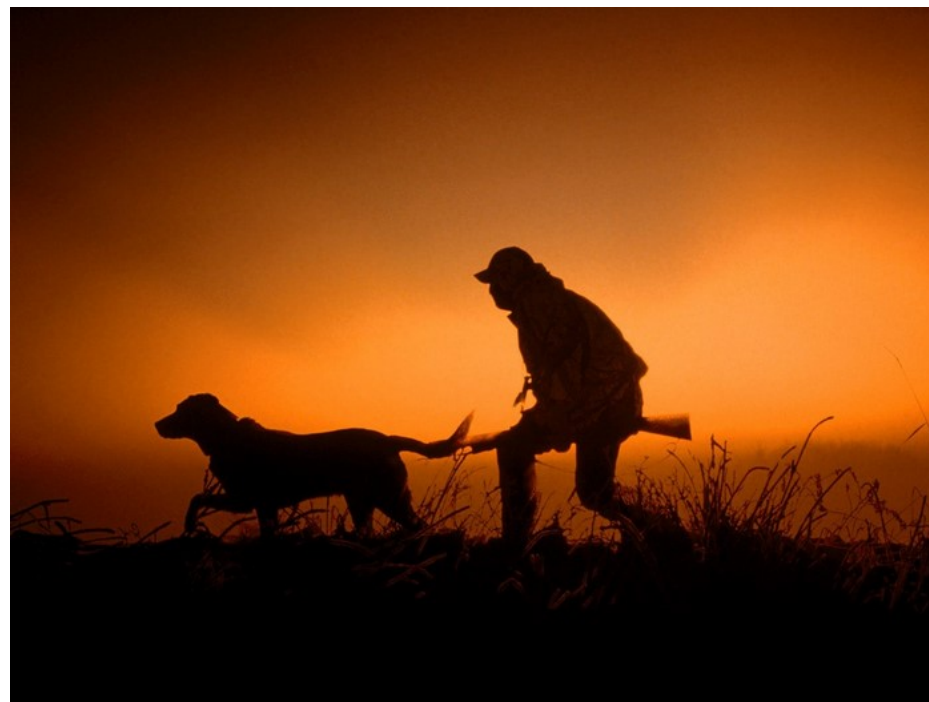
Dog



Table

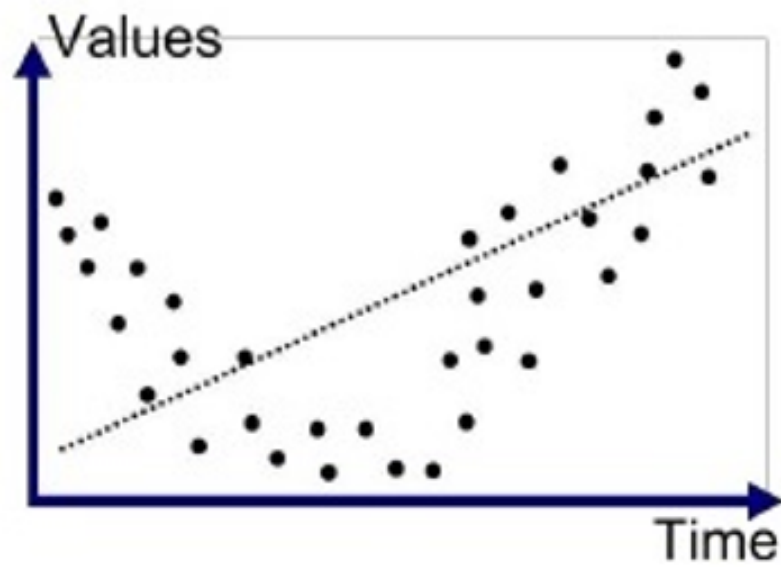


Cat

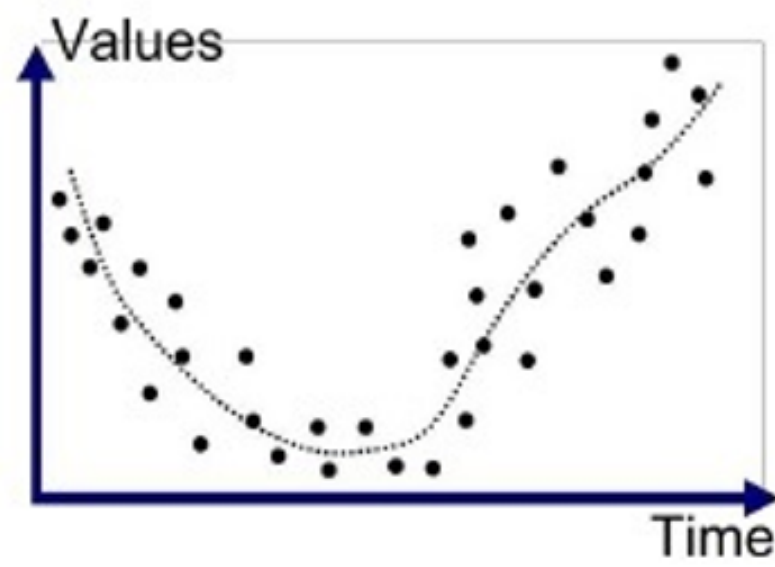


Plane!

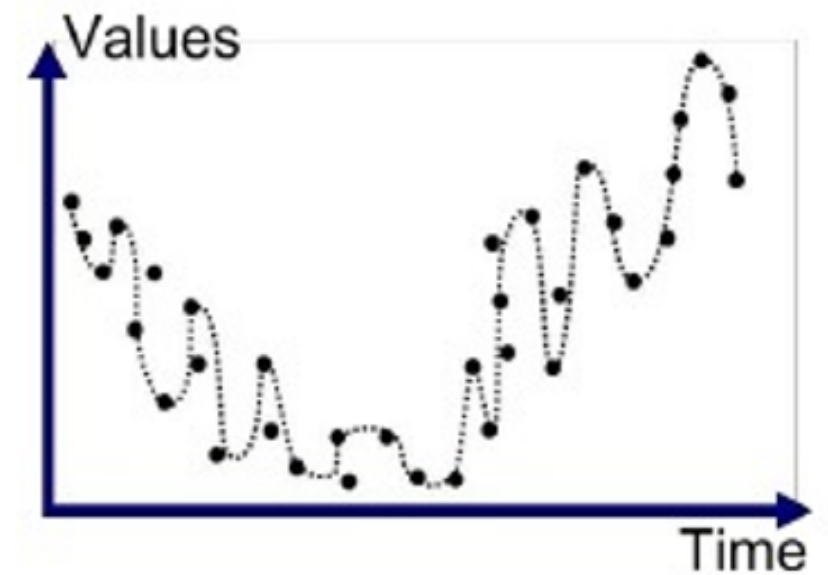
# Overfit Problem



Underfitted



Good Fit/Robust

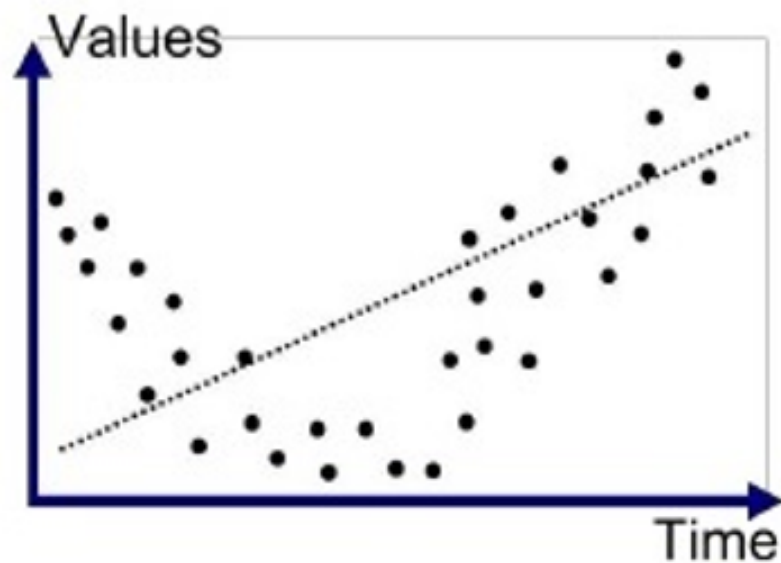


Overfitted



# Overfit Problem

The bias - variance tradeoff



Underfitted

Model is too simple, not “Expressive” enough

Accuracy can still be  $> 0$

“A broken clock is right twice a day”

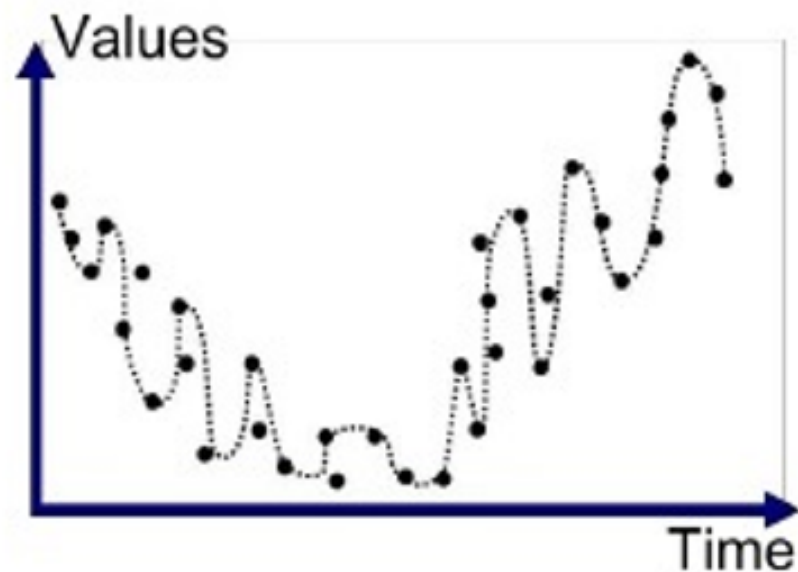
High bias - low variance

Images from “Machine Learning: How to Prevent Overfitting”

<https://medium.com/swlh/machine-learning-how-to-prevent-overfitting-fdf759cc00a9>

# Overfit Problem

The bias - variance tradeoff



Overfitted

Model is too complex, too “Expressive”

Very “accurate”

Does not generalize - useless!

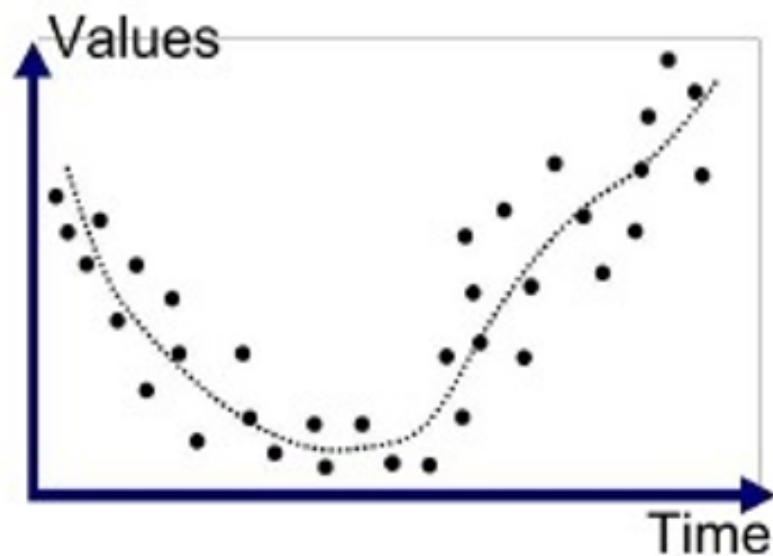
Low bias - high variance

Images from “Machine Learning: How to Prevent Overfitting”

<https://medium.com/swlh/machine-learning-how-to-prevent-overfitting-fdf759cc00a9>

# Overfit Problem

The bias - variance tradeoff



Good Fit/Robust

Tradeoff between bias and variance?

Relatively accurate

Relatively good generalization properties

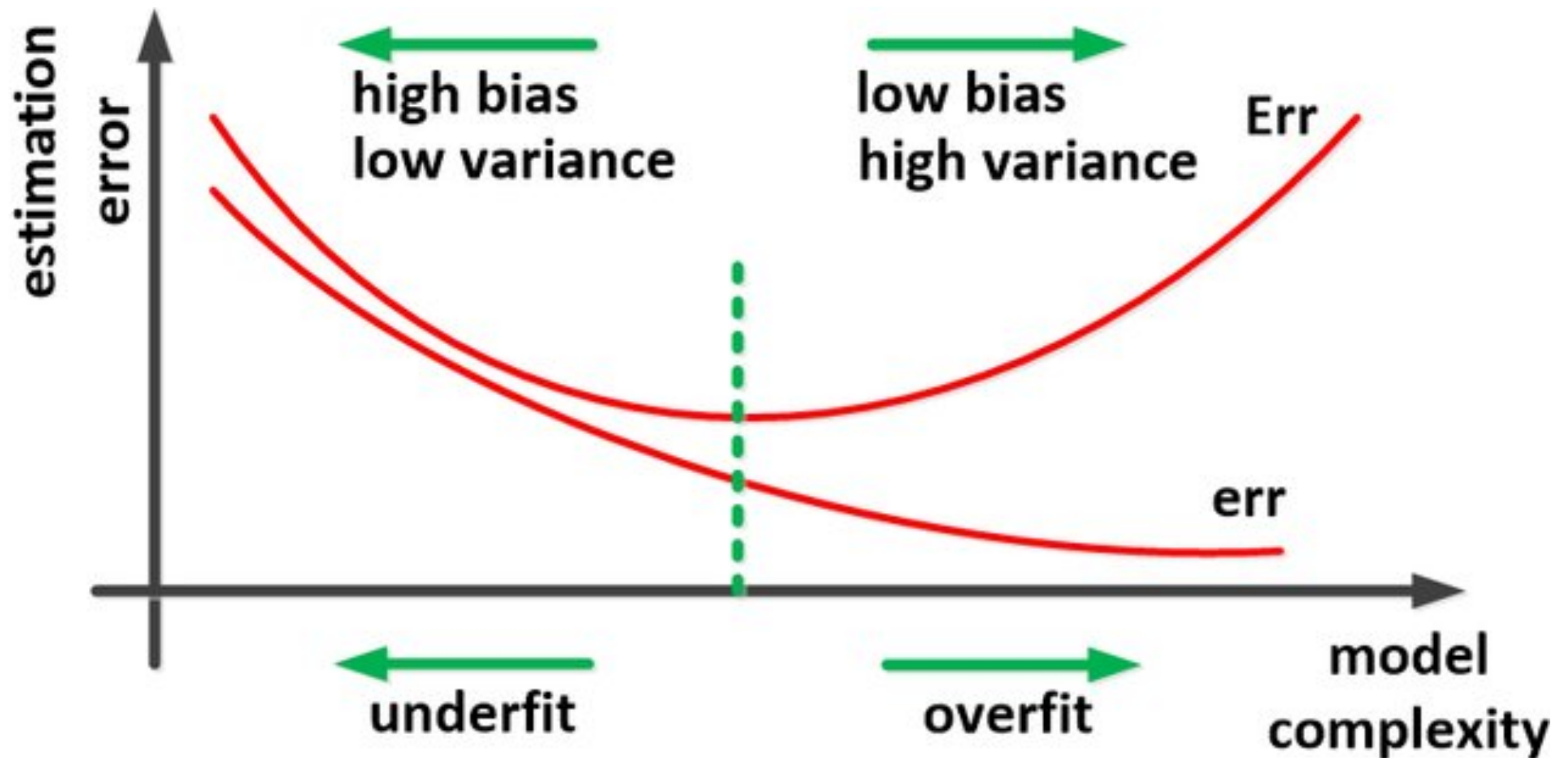
moderate bias - moderate variance

Images from "Machine Learning: How to Prevent Overfitting"

<https://medium.com/swlh/machine-learning-how-to-prevent-overfitting-fff759cc00a9>

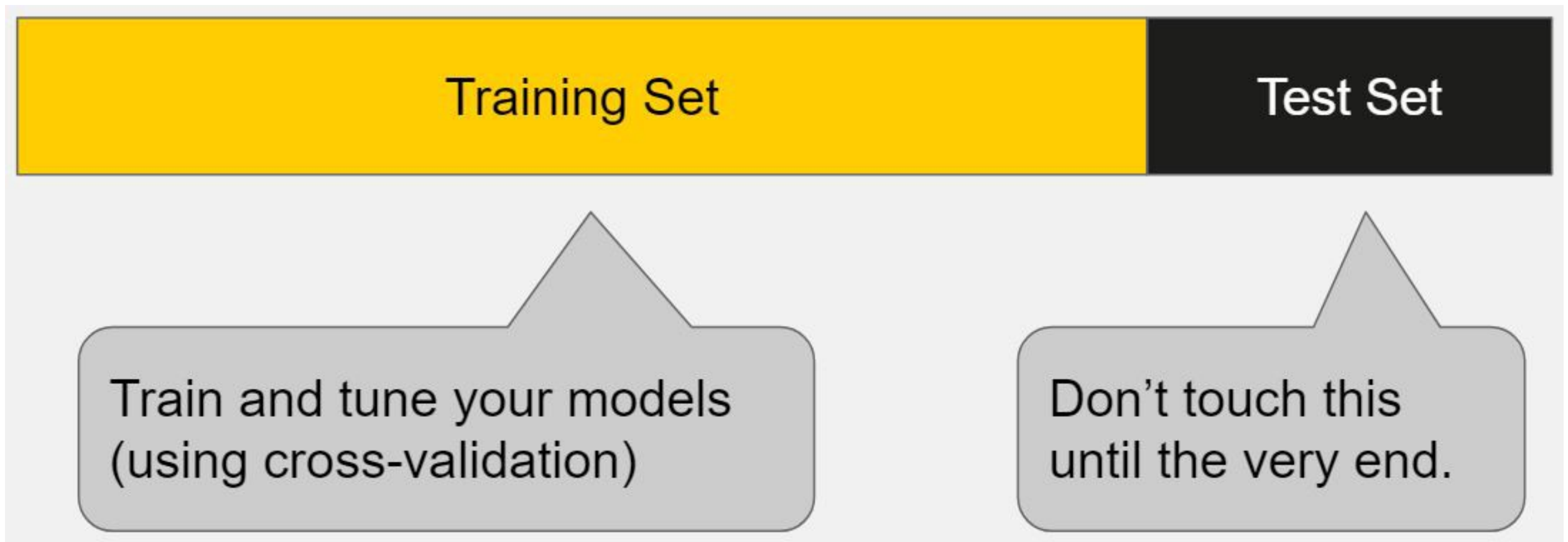
# Overfit Problem

The bias - variance tradeoff



Ghojogh, Benjamin, and Mark Crowley. "The theory behind overfitting, cross validation, regularization, bagging, and boosting: tutorial." *arXiv preprint arXiv:1905.12787* (2019).

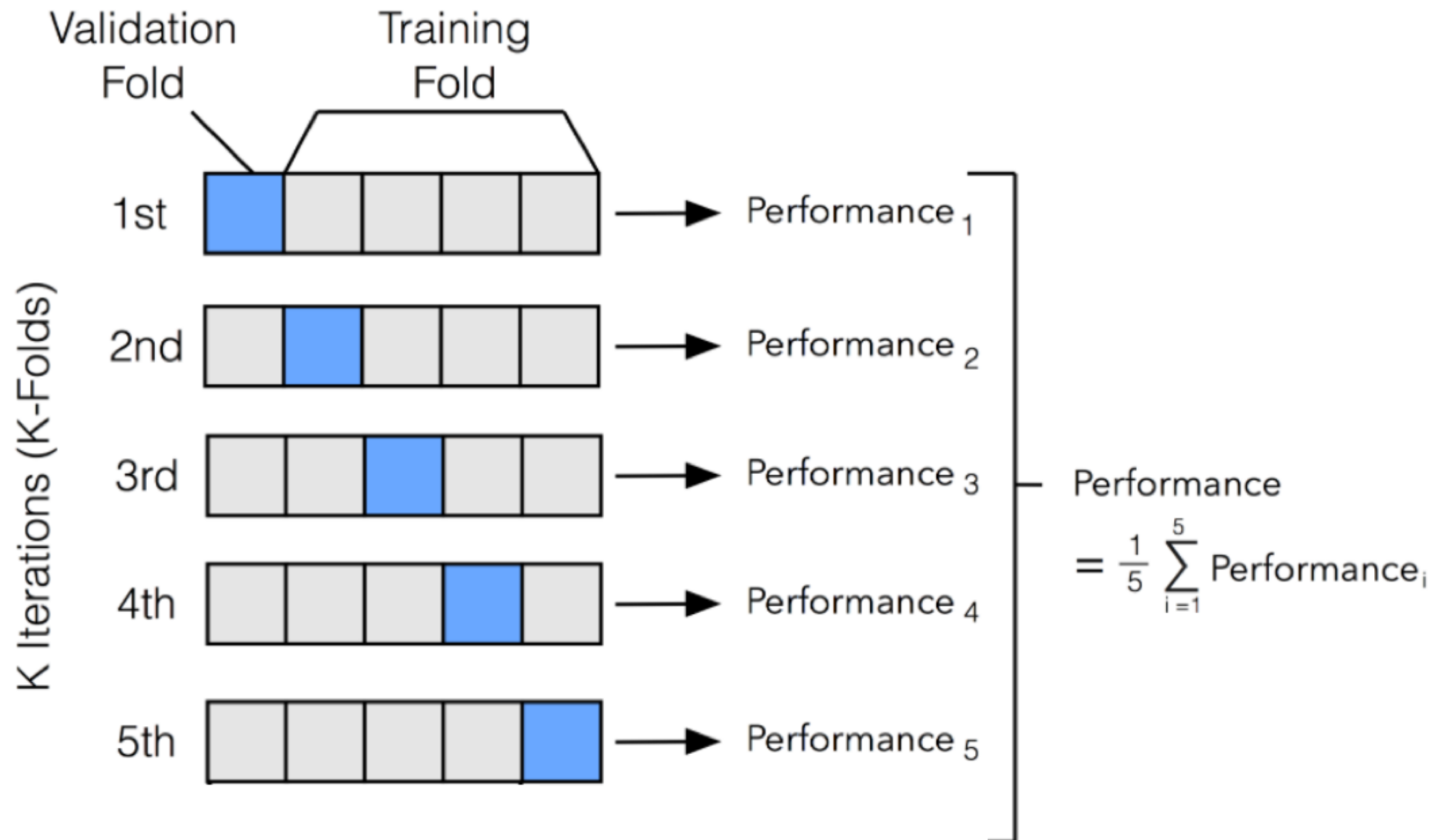
# Avoiding Overfit



# Avoiding Overfit

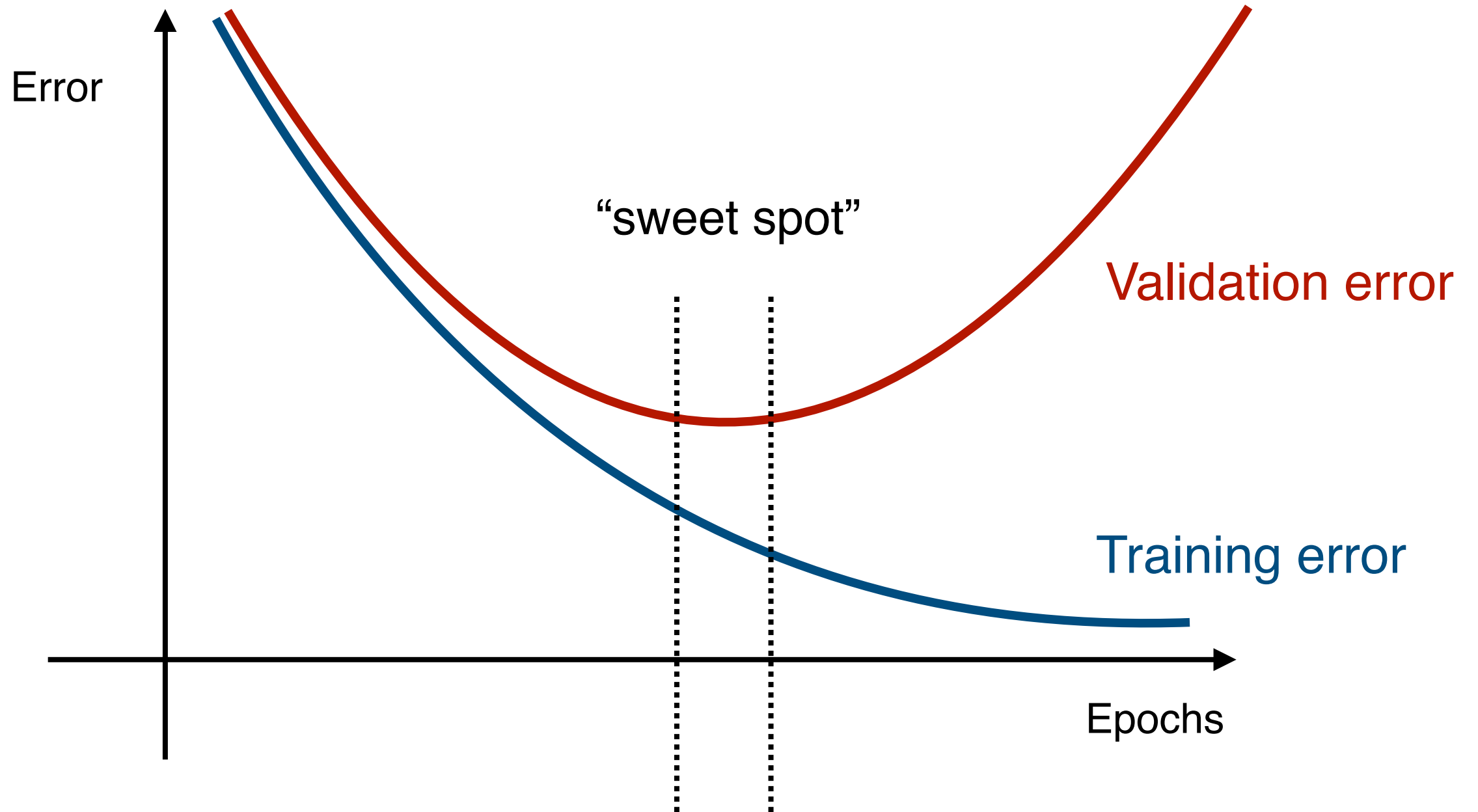
Validation set - super important!

Cross-validation (if possible)



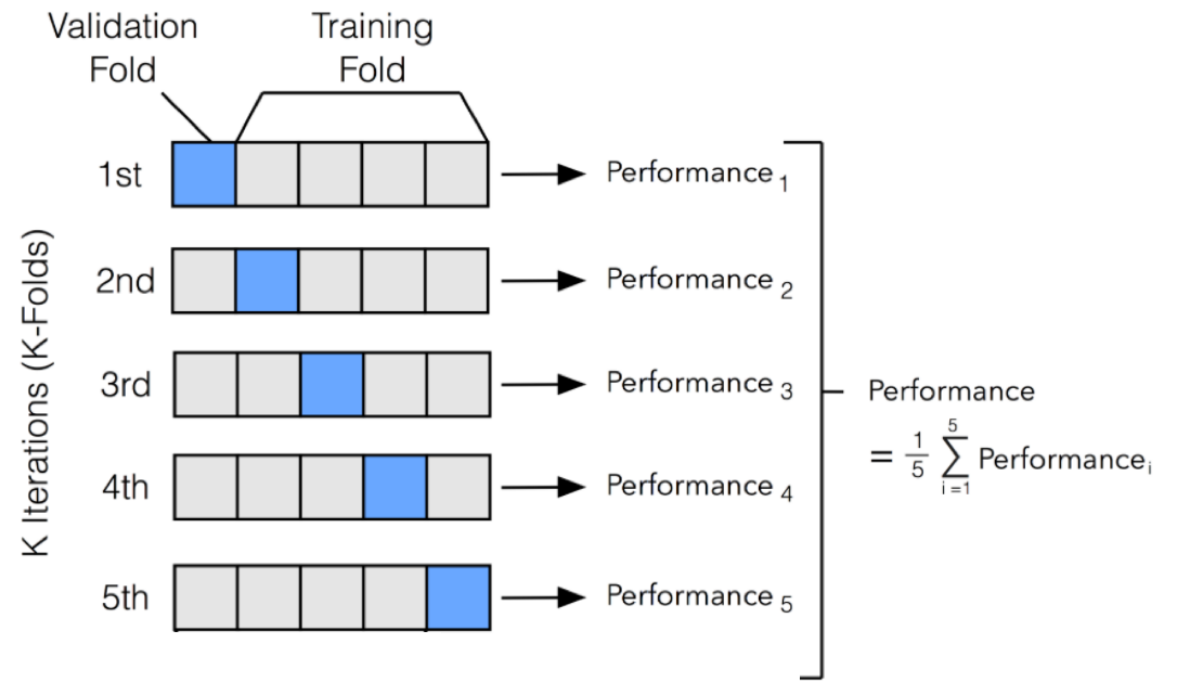
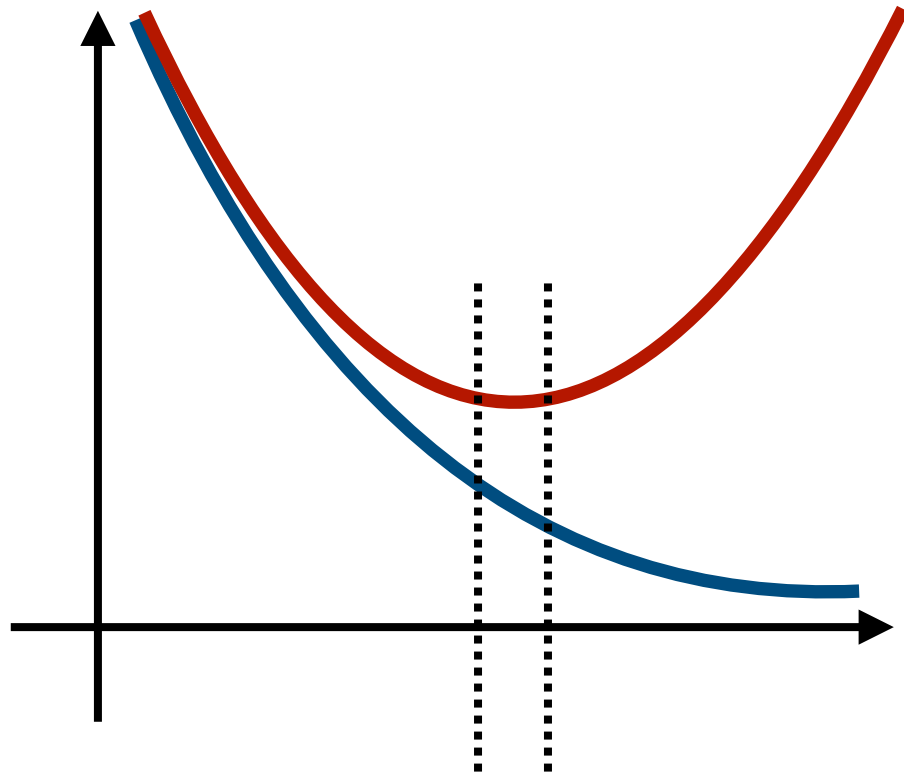
# Avoiding Overfit

Early stop



No convergence properties!

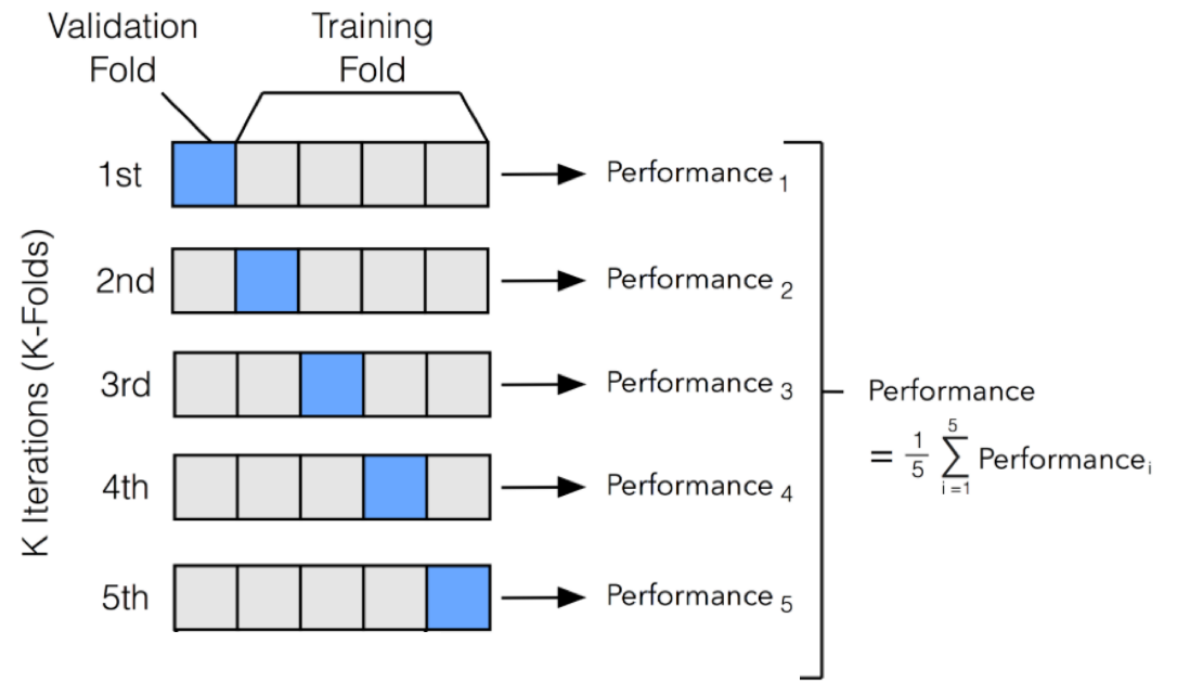
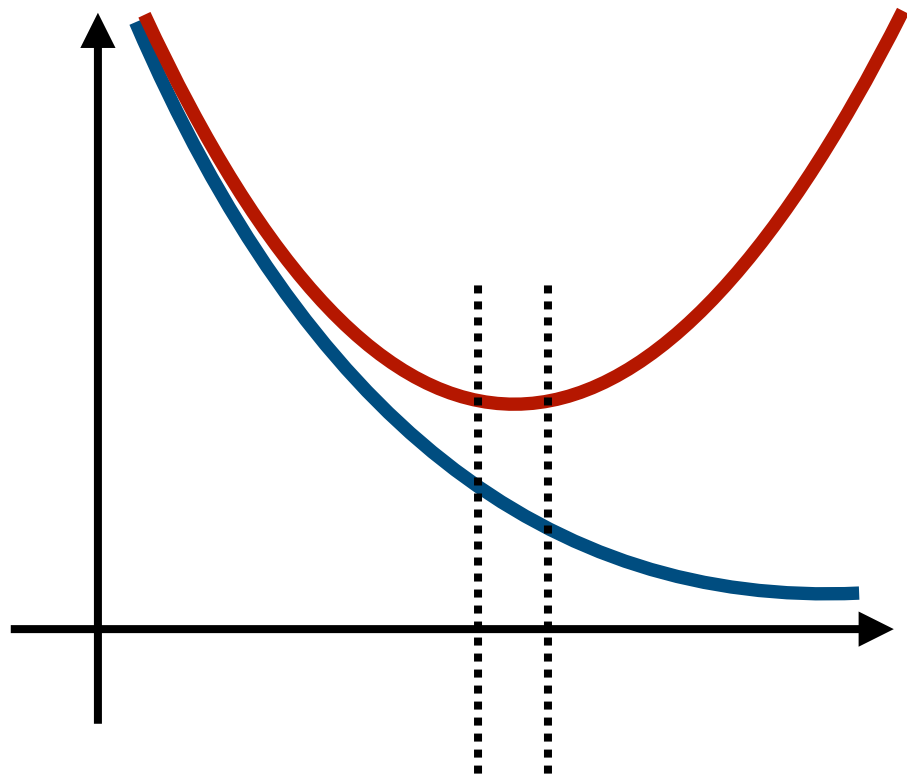
# A note on accuracy



The same method will have different results in different folds  
-> what do you think is happening?



# A note on accuracy



Evaluate on easy “naive” baselines (underfit)