MACHINE LEARNING
Supervised Approaches

Team: A. Bit-Monnot, E. Chanthery, M-J. Huguet, P. Leleux, M. Siala

https://moodle.insa-toulouse.fr/course/view.php?id=1790

Topics planned

Volume: 6 CM – 3 TP

1. Introduction on AI and supervised learning (E. Chanthery)
    - Learning process and assessment
    - A brief focus on a basic tool for learning: Gradient descent
2. Learning using Artificial Neural Networks (P. Leleux)
3. Learning using Interpretable Machine Learning Models (M. Siala)

Practical sessions

- Artificial Neural Networks
- Decision Trees

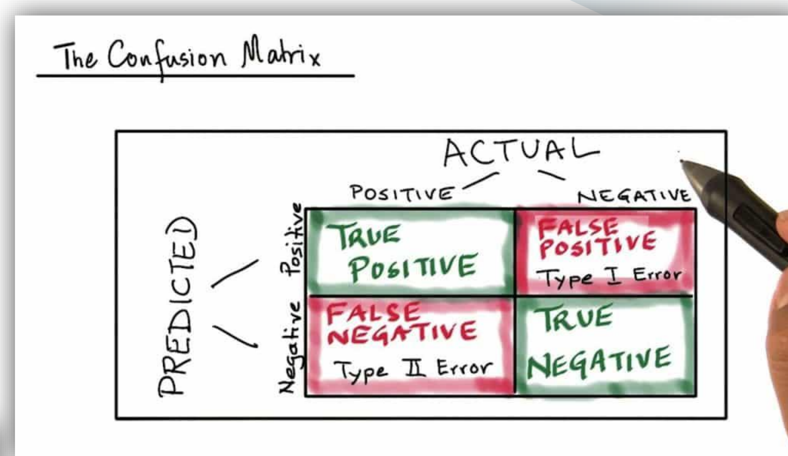
Assessment: 1 quizz on moodle, 1 report on labs

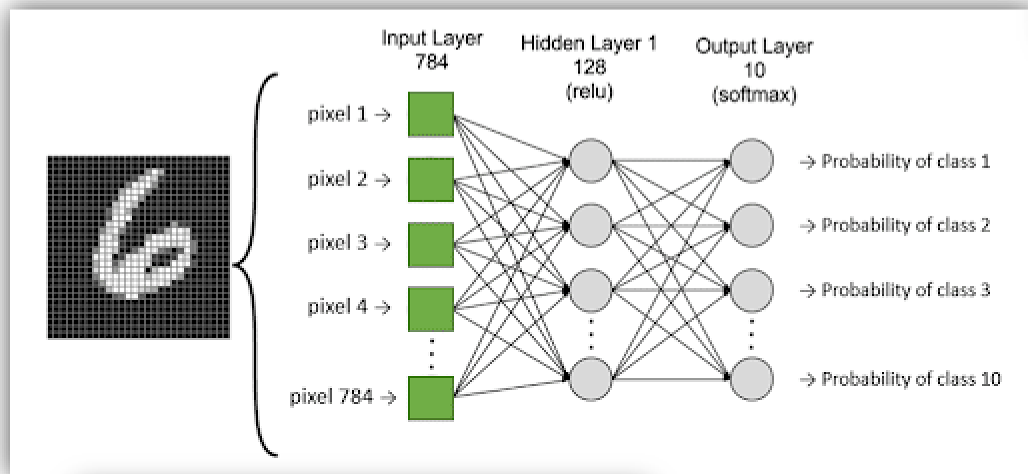
At the end of this module, the student will have understood and be able to explain:

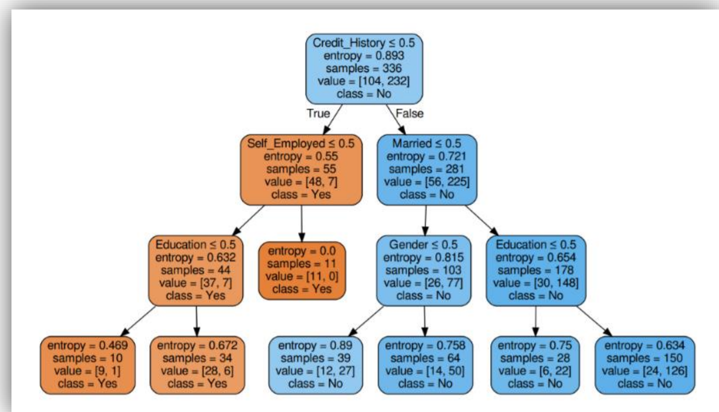
- The **characteristics of supervised learning problems** (data sets, classification / regression, learning process, evaluation of learning models)
- the **main basic methods and algorithms** to deal with these problems (neural networks and interpretable models)

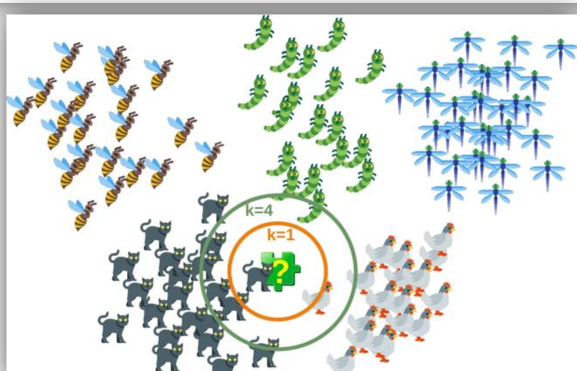
A short teaser: what you will be able to do after this course

- set up a learning process
- use the algorithms implemented in existing libraries
- adapt and develop your own algorithms
- present and explain the results of learning algorithms
- program ML in Python









Our research interests

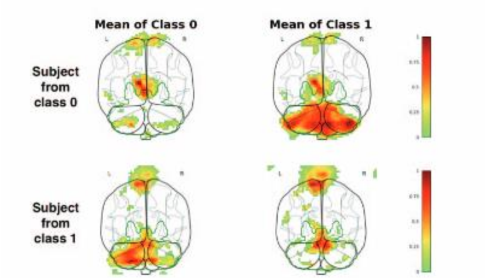


Figure 7: Visualization of the absolute distance between the activation maps from a random subject of both classes and the mean activation map on the training dataset for cerebellum and putamen abnormal-induced data. The green line contours putamen and cerebellum areas

Villain, Edouard, et al. "Visual interpretation of CNN decision-making process using Simulated Brain MRI." *2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE, 2021.

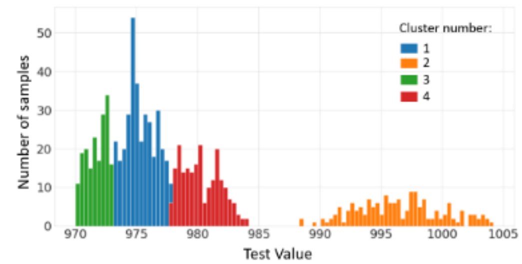


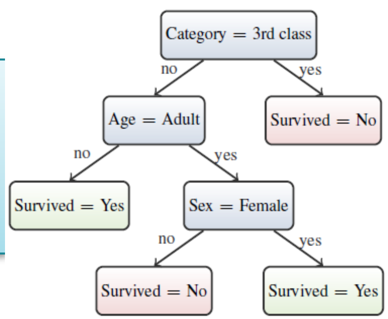
Figure 7. Histogram of the classes of the period number 3

Alexandre Gaffet, et al. Data-Driven Capability-based Health Monitoring Method for Automotive Manufacturing. *EUROPEAN CONFERENCE OF THE PROGNOSTICS AND HEALTH MANAGEMENT SOCIETY (PHM Europe)*, PHM society, Jun 2021, Turin (virtual), Italy

Data-based diagnosis and prognosis

Model learning for health monitoring

Interpretable methods
- Decision trees
- rules



IF	Age = Adult ∧ Sex ≠ Female	THEN	Survived = No
ELSE IF	Category ≠ 3rd class	THEN	Survived = Yes
		ELSE	Survived = No

Combinatory methods (MaxSAT) for interpretable models

Hu, Hao, et al. "Learning optimal decision trees with maxsat and its integration in adaboost." *IJCAI-PRICAI 2020, 29th International Joint Conference on Artificial Intelligence and the 17th Pacific Rim International Conference on Artificial Intelligence*. 2020.

Combinatory problems (Branch and Bound) for interpretable **fair** models

```
if [capitalGain>=5095.5] then [high]
else if [1881.5<=capitalLoss<=1978.5] then [high]
else if [education:hs_grad AND capitalLoss<=1534.0] then [low]
else if [occupation:whiteCollar AND hoursPerWeek>=40.5] then [high]
else [low]
```

Error=0,183
Unf=0,066

```
if [education:hs_grad AND hoursPerWeek>=40.5] then [low]
else if [35.5<=age<=61.5 AND occupation:professional] then [high]
else if [capitalGain>=7073.5] then [high]
else [low]
```

Error=0,208
Unf=0,0036

FairCORELS, an Open-Source Library for Learning Fair Rule Lists. U. Aivodji, J. Ferry, S. Gambs, M-J. Huguet, M. Siala - ACM International Conference on Information and Knowledge Management (CIKM), November 1-5, 2021

Introduction on AI and Supervised Learning

INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

What is AI?
What is Machine Learning?
What is Supervised Learning?

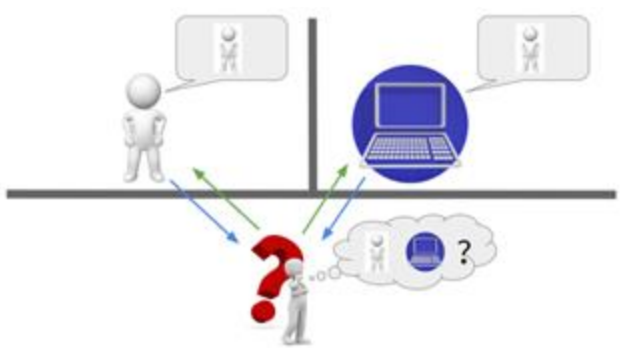
Université de Toulouse

What is AI?

1950's : idea : do not define AI, but test it → Turing test intended to test whether or not a machine has the ability to imitate human intelligence

1956: AI as a scientific field (conference at Darmouth College)

"The main components of an AI *system should be knowledge, reasoning, natural language understanding and learning."* A. Turing



empirique	théorique
Systems that think like humans	**Systems that think rationally**
"The exciting new effort to make computers think ... *machines with minds,* in the full and literal sense." (Haugeland, 1985) "[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning ..." (Bellman, 1978)	"The study of mental faculties through the use of computational models." (Chamiak and McDermott, 1985) "The study of the computations that make it possible to perceive, reason, and act." (Winston, 1992)
Systems that act like humans	**Systems that act rationally**
"The art of creating machines that perform functions that require intelligence when performed by people." (Kurzweil, 1990) "The study of how to make computers do things at which, at the moment, people are better." (Rich and Knight, 1991)	"Computational Intelligence is the study of the design of intelligent agents." (Poole *et al.*, 1998) "AI ...is concerned with intelligent behavior in artifacts." (Nilsson, 1998)

Figure 1.1 Some definitions of artificial intelligence, organized into four categories.

▶ *AIMA 2nd edition, p. 2*

Traditional programming paradigm vs ML

The traditional programing paradigm

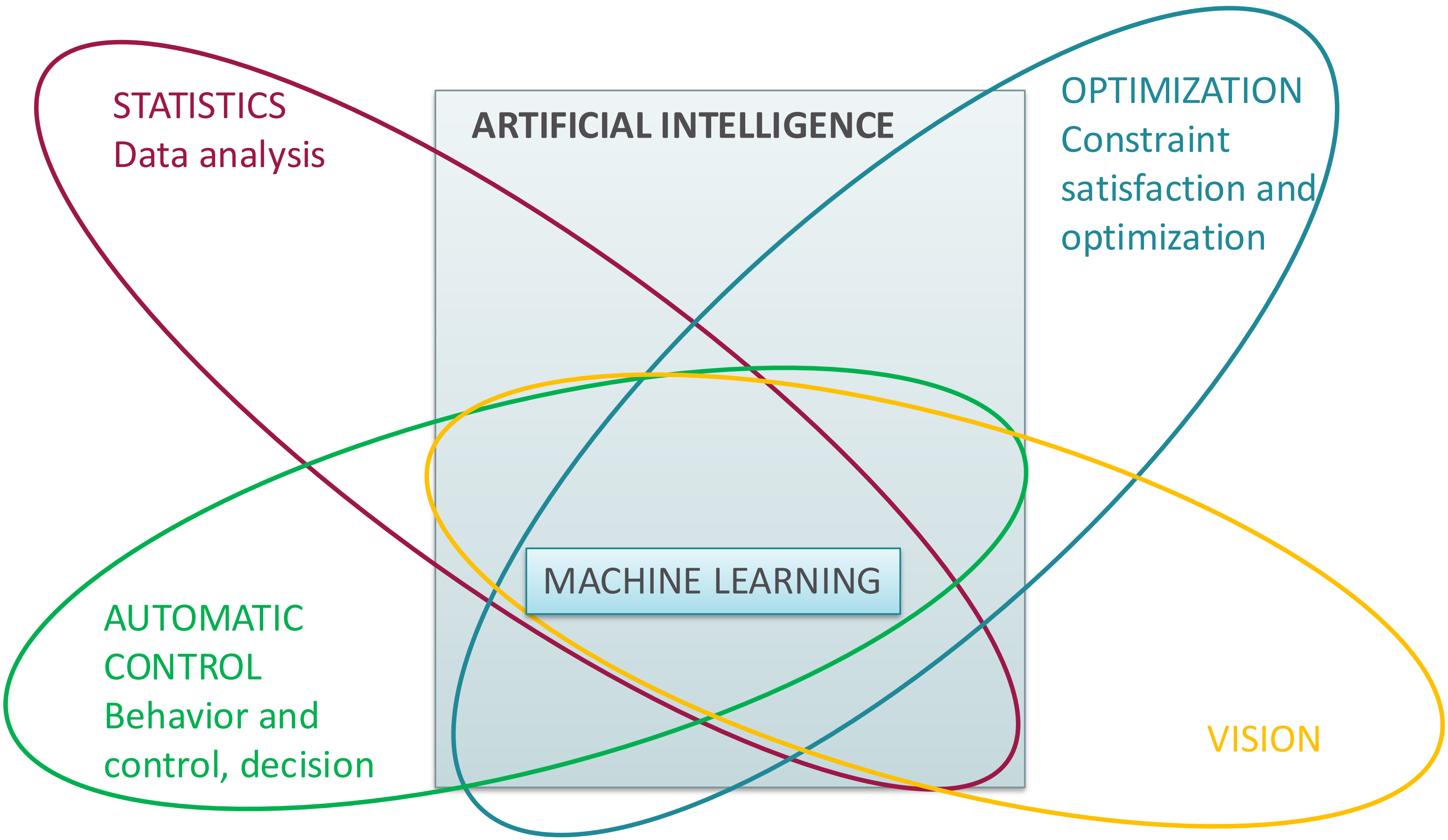
program + data = outputs

Machine Learning

data + outputs = program

"Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed". Arthur Samuel (1959)

The connection between fields: AI in your cursus



Artificial Intelligence

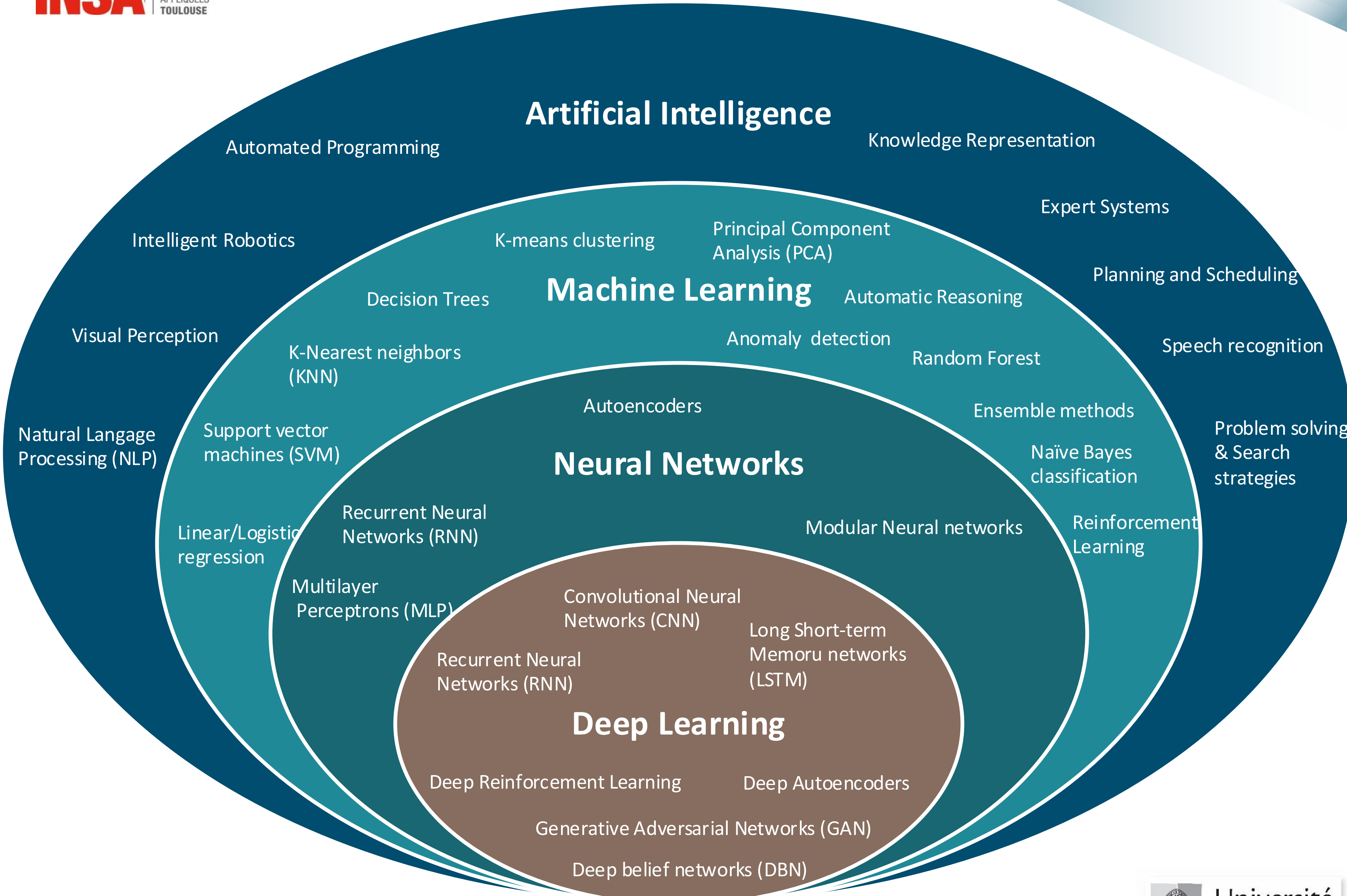
Machine Learning

- Supervised Learning
- Semi-supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Transfert Learning
- Deep Learning

- Natural Language Processing (NLP)
- Reasoning
- Planning
- Knowledge Representation and Reasoning
- Vision
- Motion and Manipulation
- Multiagent Systems
- General Intelligence

Machine Learning is a subset of Artificial Intelligence. The term Artificial Intelligence is often used wrongly (buzzword in the sense of global intelligence).

Not all AI systems involve machine learning: ex Deep Blue executes the alpha-beta search algorithm: it is not ML


Artificial Intelligence
Automated Programming
Knowledge Representation
Expert Systems
Intelligent Robotics
Planning and Scheduling
Visual Perception
Speech recognition
Natural Langage Processing (NLP)
Problem solving & Search strategies
Machine Learning
K-means clustering
Principal Component Analysis (PCA)
Decision Trees
Automatic Reasoning
K-Nearest neighbors (KNN)
Anomaly detection
Random Forest
Support vector machines (SVM)
Ensemble methods
Naïve Bayes classification
Linear/Logistic regression
Reinforcement Learning
Neural Networks
Autoencoders
Recurrent Neural Networks (RNN)
Modular Neural networks
Multilayer Perceptrons (MLP)
Deep Learning
Convolutional Neural Networks (CNN)
Long Short-term Memoru networks (LSTM)
Recurrent Neural Networks (RNN)
Deep Reinforcement Learning
Deep Autoencoders
Generative Adversarial Networks (GAN)
Deep belief networks (DBN)

Some applications of Machine learning

- Diagnostic support systems
- High stake decision-making systems
- Games
- Pattern recognition: email spam detection, fingerprint/face detection and matching
- System control: self-driving cars (Uber, Tesla), automatic control, sort (post office)

  Automatic translation (initiated during the war) (google translate, deepl)
- Voice synthesizer, smart assistants (Apple Siri, Amazon Alexa…)
- Finance/industry: cost of living forecasting, stock predictions
- Sports prediction, product recommendation (Netflix, Amazon…)
- Drug design, medical diagnoses (EEC and ECG analysis)

…

The broad categories of ML

Machine Learning

- Supervised Learning
- Semi-supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Transfer Learning
- Deep Learning

Supervised Learning

Supervised Learning

- Labeled data
- Direct feedback
- Predict outcome/future

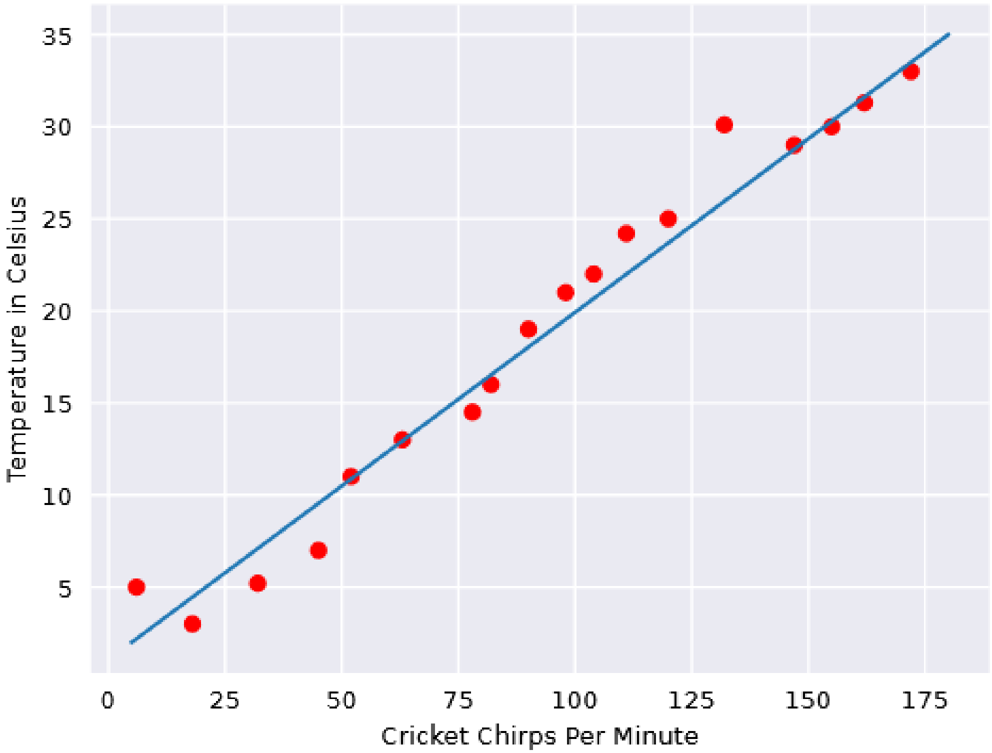
Source: Raschka and Mirjalily (2019). *Python Machine Learning, 3rd Edition*

Input: a set of data for which we know the class (classification) said to be **annotated/labeled** with their outputs or the result of the function (regression): these values are called targets or labeled data.

Goal: an algorithm/model that can predict from new data x*→y* once it has been "trained" by $(x1, y1), (x2, y2), (x3, y3)$,...

Supervised Learning (1): regression

Target (dependent variable, output)

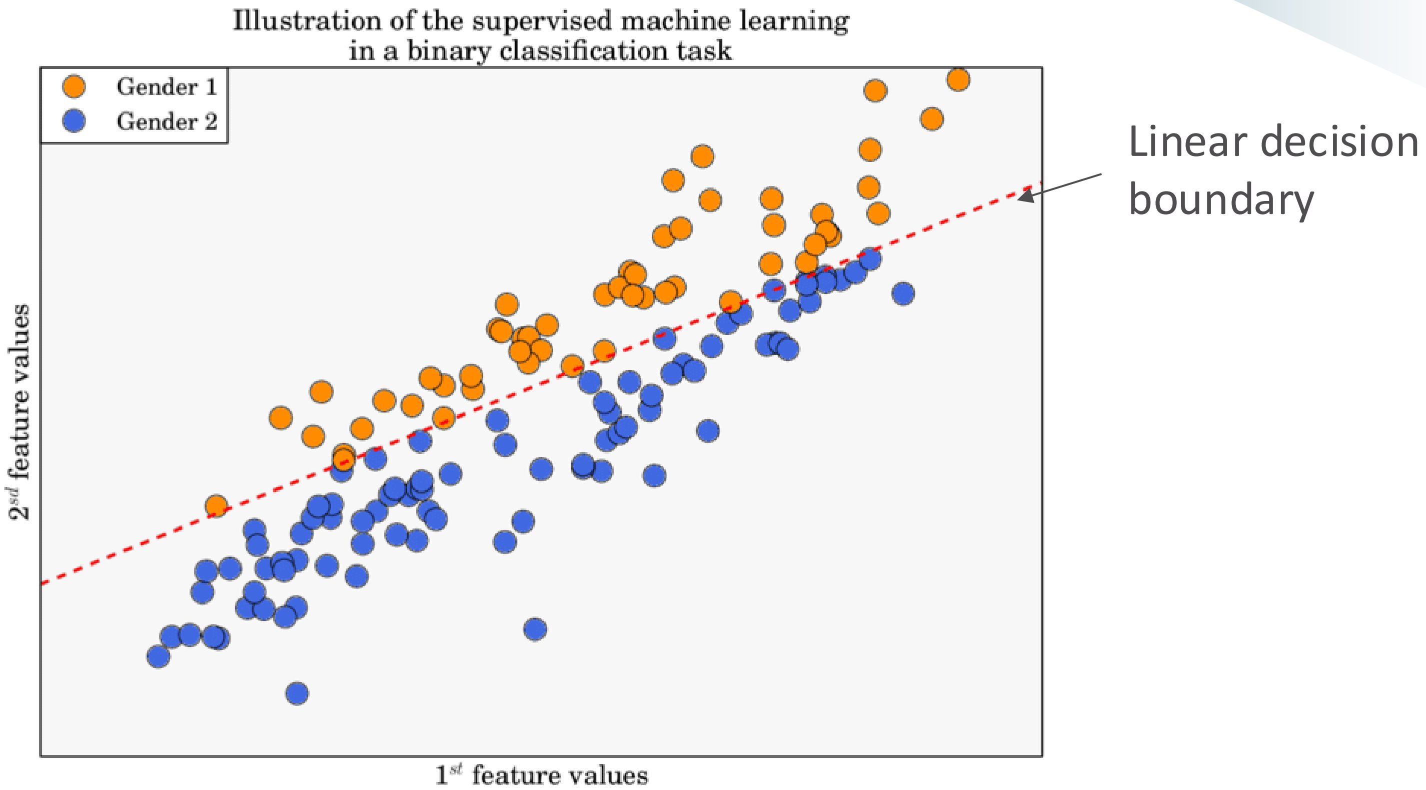


Feature (input, observation)

To **infer** (predict) the temperature for a new chirps-per-minute value, just substitute the value into the blue model.

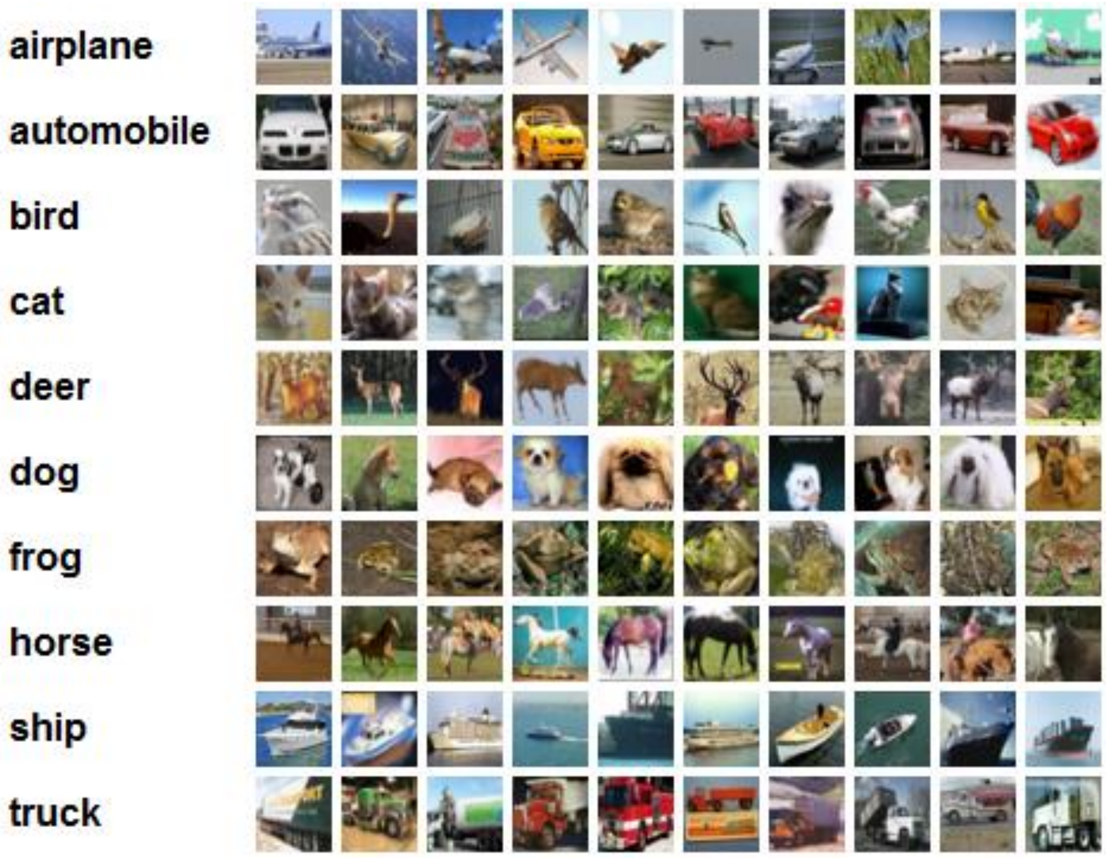
Source: https://developers.google.com/machine-learning/crash-course/descending-into-ml/linear-regression

Supervised Learning (2): classification



Kawala, François. (2015). Activity prediction in social-networks.

Supervised Learning (3): classification



Input data: labeled images
Target: photo category

Unsupervised Learning

Unsupervised Learning

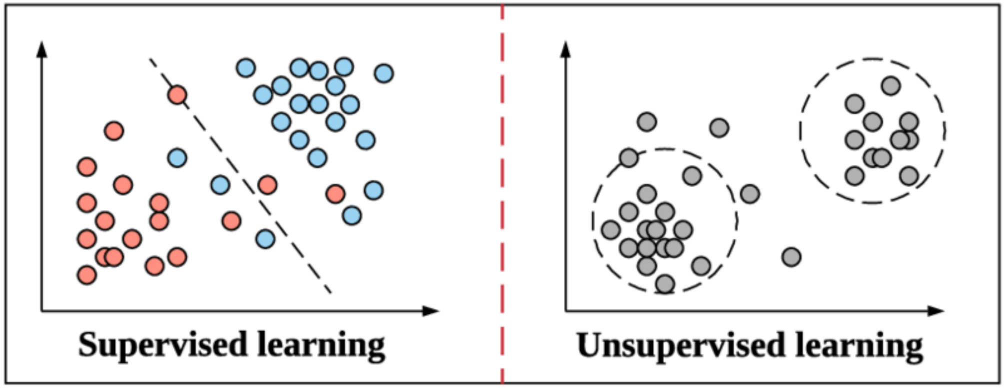
- No labels/targets
- No feedback
- Find hidden structure in data

(Won't cover this in this course)
But in 5th year! (SIEC)

Source: Raschka and Mirjalily (2019). *Python Machine Learning, 3rd Edition*

Inputs: unlabeled data, the targets are unknown

Goal: group the data $x1, x2, x3, , \dots$ by similarity and discover the relationship with structural latent variables: $xi \rightarrow yi$



INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

(Won't cover this in this course)

Semisupervised Learning

- Some training examples contain outputs, but some do not
- Use the labeled training subset to label the unlabeled portion of the training set → model training

- A recent development, promising research trend in deep learning
- Useful if pre-trained models for transfer learning are not available

Reinforcement Learning

- Based on an experience/reward cycle
- Decision process that improves performance with each iteration, reward system
- The "dopamine" effect

Université de Toulouse

Data

Key ML terminology

Supervised learning ML systems learn how to combine input to produce useful predictions on never-before-seen data

Labels A label is the thing we're predict y

Ex: the future price of house, the kind of animal shown in a picture, the meaning of an audio clip

Features/attributes/variables A feature is an input variable.

A simple ML model uses a single feature x; a more sophisticated ML model could use millions of features: $x_1, \ldots, x_N$

Examples/samples/instances/point/vector An example is a particular instance of data $\boldsymbol{x}$, it is made up of attributes

It is assume the data set consists of N **samples**

Examples

housingMedianAge (feature)	totalRooms (feature)	totalBedrooms (feature)	medianHouseValue (label)
15	5612	1283	66900
19	7650	1901	80100
17	720	174	85700
14	1501	337	73400
20	1454	326	65500

Labeled examples from a data set containing information about housing prices in California

housingMedianAge (feature)	totalRooms (feature)	totalBedrooms (feature)
42	1686	361
34	1226	180
33	1077	271

Unlabeled examples about housing prices in California

iquiz

sepal_length	sepal_width	petal_length	petal_width	Iris_class
5	2	3.5	1	versicolor
6	2.2	4	1	versicolor
6.2	2.2	4.5	1.5	versicolor
6	2.2	5	1.5	virginica
4.5	2.3	1.3	0.3	setosa
5.5	2.3	4	1.3	versicolor
6.3	2.3	4.4	1.3	versicolor
5	2.3	3.3	1	versicolor
4.9	2.4	3.3	1	versicolor
5.5	2.4	3.8	1.1	versicolor
5.5	2.4	3.7	1	versicolor
5.6	2.5	3.9	1.1	versicolor
6.3	2.5	4.9	1.5	versicolor
5.5	2.5	4	1.3	versicolor
5.1	2.5	3	1.1	versicolor
4.9	2.5	4.5	1.7	virginica
6.7	2.5	5.8	1.8	virginica
5.7	2.5	5	2	virginica
6.3	2.5	5	1.9	virginica
5.7	2.6	3.5	1	versicolor
5.5	2.6	4.4	1.2	versicolor
5.8	2.6	4	1.2	versicolor

Categorical value

What does the green line represent?

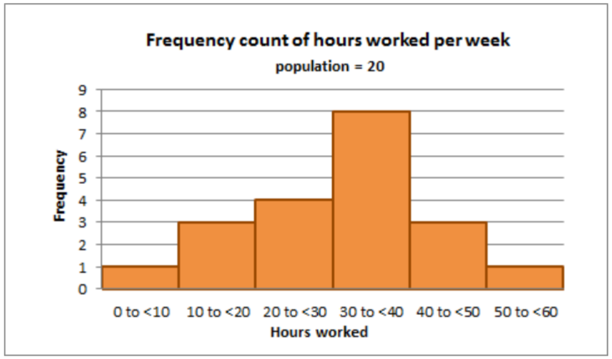
1. Attributes
2. One sample
3. Several samples

iquizz

Suppose you want to developed a supervised ML model to predict whether a given email is “spam” or “not spam”. Which of the following statements is true?

1. Word in the subject header will make good labels
2. Unlabeled example will be used to train the model
3. Emails not marked as “spam” or “not spam” are unlabeled examples
4. By hypothesis, all the labels applied to examples are reliable

Basic types of data

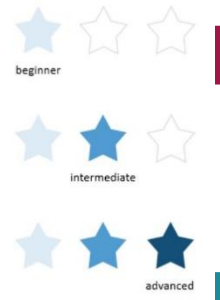


Tabular data

Quantitative/Numerical data

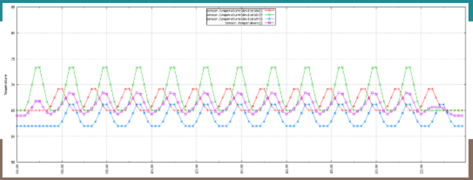
- Can be counted, data are exact numbers, but they are not ordered
- Ex: house prices, speed, frequency
- Can be **continuous** (temperature, speed) or **discrete/binary** (number of cycles)
- Special mention for **time** and **interval**

Qualitative/Categorical data



- Can't be counted, represents characteristics
- Ex: gender, color, team
- Can take numerical values that do not have mathematical meaning
- Can be **nominal** (not ordered, ex: gender, color) or **ordinal** (small<medium<large)
- could the class label

Time series data



- A sequence of numbers collected at regular intervals over some period of time
- Values are ordered : there is a first data point and a last data point collected
- Ex: the voltage value during 10 sec

Text

Video

Audio

Image

https://medium.com/ml-research-lab/chapter-2-data-and-its-different-types-3dfebcbb4dbe

Other classification for quantitative data

- **Transactional logs**: record a **specific event**.

  Ex: record an open command, plus date and time

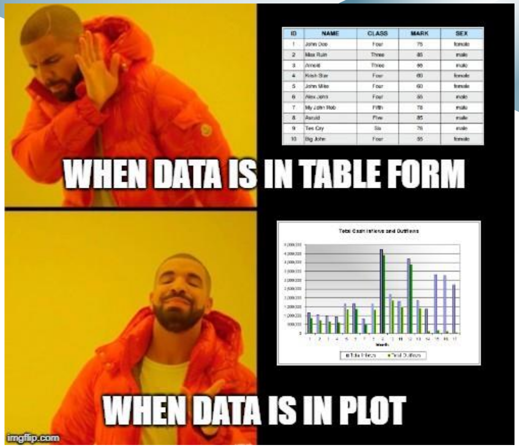
- **Attribute data:** contain snapshots of information.

  Ex: user demographics

- **Aggregate statistics:** create an attribute from multiple transactional logs.

  Ex: average of a signal value

Data vizualization



- Why visualization? For understandability, for intuition, for explainability
- How? → there are very good packages for visualization

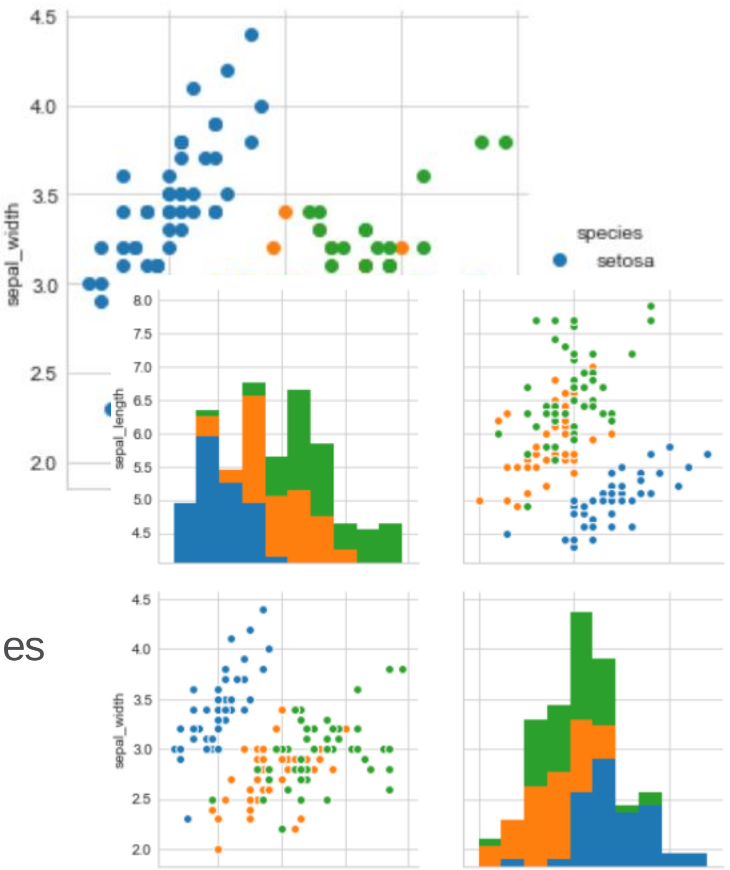
  Common languages: R (more a statistical language), Python (widely used for ML and data science)

  Packages: scikit, matplotlib, seaborn…
- How to work with big dimensionality? Dimensionality reduction techniques (PCA, **TSNE**, LDA…)

https://towardsdatascience.com/data-visualization-for-machine-learning-and-data-science-a45178970be7

Different types of analysis

- Univariate analysis: plot a single feature to analyze its properties
  - Box plot
  - Violin plot
  - Distribution plot
  - Joint plot
- Bivariate analysis: compare exactly 2 features
  - Scatter plot
  - Bar chart
  - Line plot
- Multivariate analysis: compare more than 2 features
  - Pair plot: a nxn figure with



Pair plot

https://towardsdatascience.com/data-visualization-for-machine-learning-and-data-science-a45178970be7

Quality and size of the data sets

- The **quality** and **size of the data set** matter much more than which shiny algorithm you use

Google Translate

The Google Translate team has more training data than they can use. Rather than tuning their model, the team has earned bigger wins by using the best features in their data.

"Interesting-looking" errors are typically caused by the data. Faulty data may cause your model to learn the wrong patterns, regardless of what modeling techniques you try.

https://developers.google.com/machine-learning/data-prep

The size of the data set

- As a rough rule of thumb, your model should **train on at least an order of magnitude more examples than trainable parameters**.
- What is “a lot” of data?

Data set	Size (number of examples)
Iris flower data set	150 (total set)
MovieLens (the 20M data set)	20,000,263 (total set)
Google Gmail SmartReply	238,000,000 (training set)
Google Books Ngram	468,000,000,000 (total set)
Google Translate	trillions

- Amount of data depends on the complexity of the “true” function

  If the true function is simple, a small amount of data is enough for a learning algorithm with high bias and low variance

  If the true function is complex, a very large amount of data will be necessary and the algorithm should be with low bias and high variance

Quality of a data set

- **Rule 1**: **be pragmatic** → a quality set is one that accomplishes its intended task
- **3 quality criteria:**
  - Reliability
  - Feature representation
  - Minimizing skew

Reliability (fiabilité)

Represents how much you can trust the data

- Depends on the **label errors** (ex: if the data are labeled by humans → mistakes)
- Depends on the **noise** on the features (ex: GPS measurements)
- Depends if the data are **appropriate** for the problem (ex: bias)

Examples:

- Omitted values in a data base
- Duplicated examples
- Bad labels
- Bad feature values (sensor failures for example)

In your machine learning project, how much time will you typically spend on data preparation and transformation?

1. Less than half time of the project
2. More than half time of the project

Feature representation

- How is data shown to the model?

- Should you **normalize** numeric values? Are the features scaled to similar ranges?

  Methods that employs distance are sensitive to this → SVM, KNN perform poorly

- How should you handle **outliers**?
- Is there some redundancy/interaction in the data or not ?

  → If each of the features makes an independent contribution to the output

  linear regression, SVM, naive Bayes and distance-based algorithms (KNN) perform well

  → If there are complex interactions among features

  Linear regression, distance based methods will perform poorly, decision trees and neural networks work better

De quand date l'Intelligence Artificielle?

1. Des 10 dernières années
2. Des années 2000
3. Des années 70
4. Des années 50

Quelle approche d'apprentissage est utilisée lorsque l'algorithme s'améliore en fonction de son expérience antérieure?

1) Apprentissage supervisé
2) Apprentissage non supervisé
3) Apprentissage par renforcement
4) Apprentissage semi-supervisé

Quelle est la différence entre l'apprentissage supervisé et l'apprentissage non supervisé?

- 1) La présence ou l'absence d'un modèle
  2) La présence ou l'absence d'étiquettes dans les données d'entraînement
  3) La complexité des algorithmes utilisés
  4) Le nombre de layers dans le modèle

Minimizing skew (distorsion): training vs prediction

- Sometimes we get great results offline and very bad results online

→ It is a training/serving skew (distorsion apprentissage/reconnaissance)

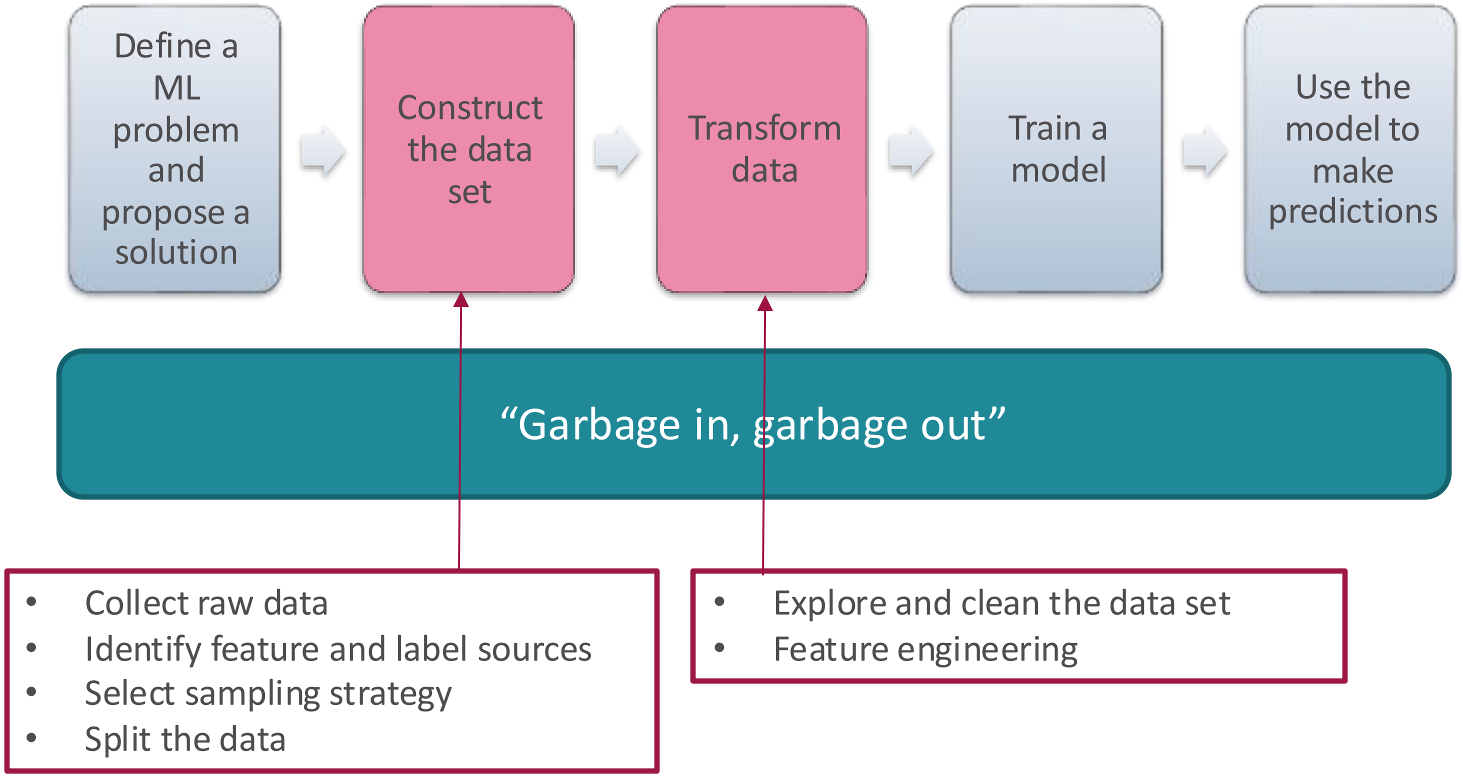
→ Always consider what data is available to your model **at prediction time**. During training, use only the features that you'll have available in serving, and make sure your training set is representative of your serving traffic.

The more closely the training task matches the prediction task, the better the ML system will perform

Prediction data sources: online vs offline

- Online: latency is concern, the system must generate input quickly
- Offline: no computation restrictions

Data preparation



Caution:

- this process is a typical process, not ideal for every process
- The process is not always sequential (more data needed, modification of the feature set even after training)

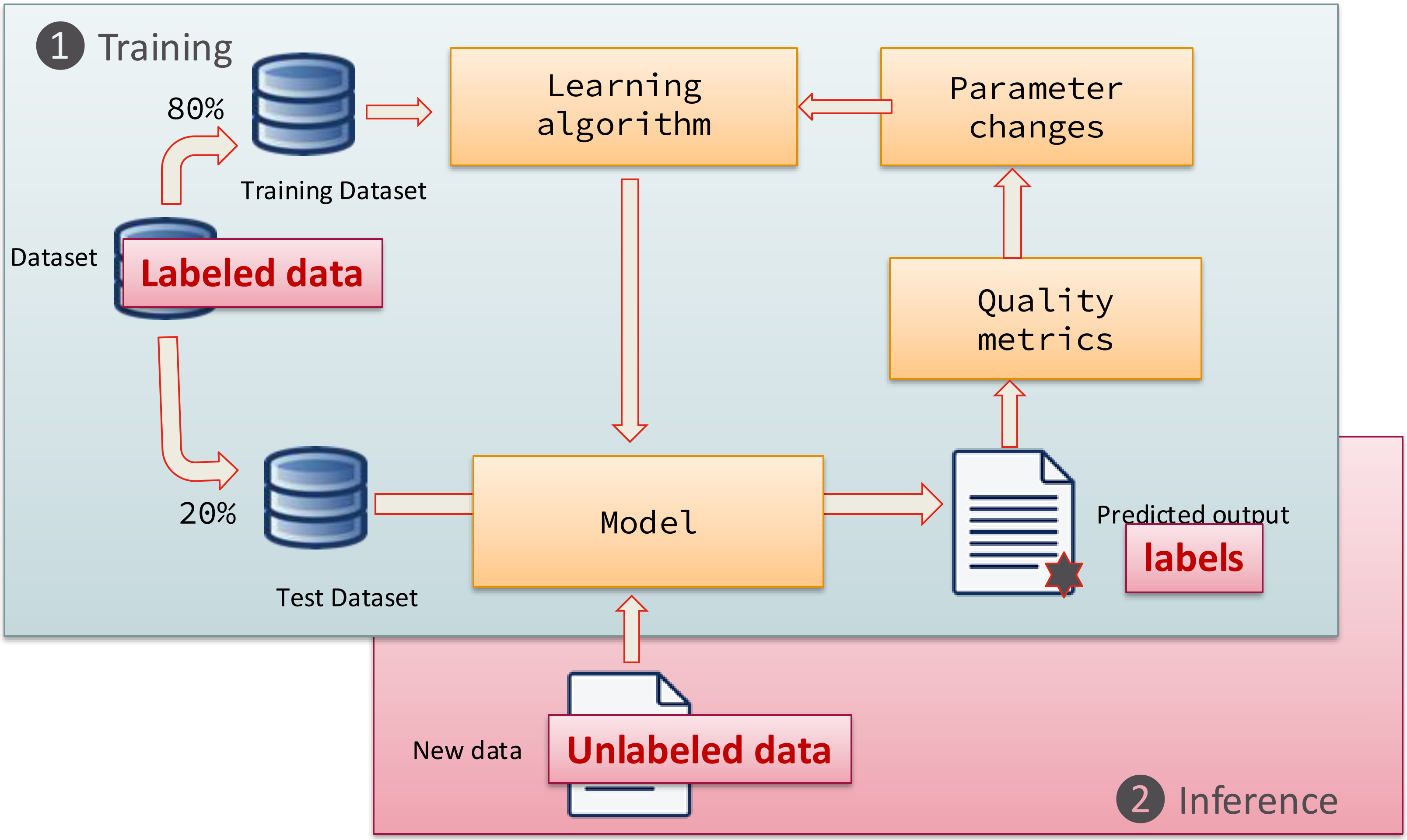
https://developers.google.com/machine-learning/data-prep

The ML workflow

The Machine Learning Workflow

A machine learning problem has different specific elements:

- The data (training data but also new data)
- The specific task to be accomplished (predict, recommend, decide something, etc.)
- The learning algorithm itself
- The error analysis (or measurement of the model performance)



Suppose we must choose between two possible ways to fit some data. How do we choose between them?

→ Simple solution: try to fit the data as closely as possible.

Problem: the generalization to new measurements

→ Solution: evaluate models by **testing them on a new data set** (the “test set”), distinct from the training set. Model validation: estimating the reliability of a model

Cross-validation is a method for estimating the reliability of a model based on a sampling technique.

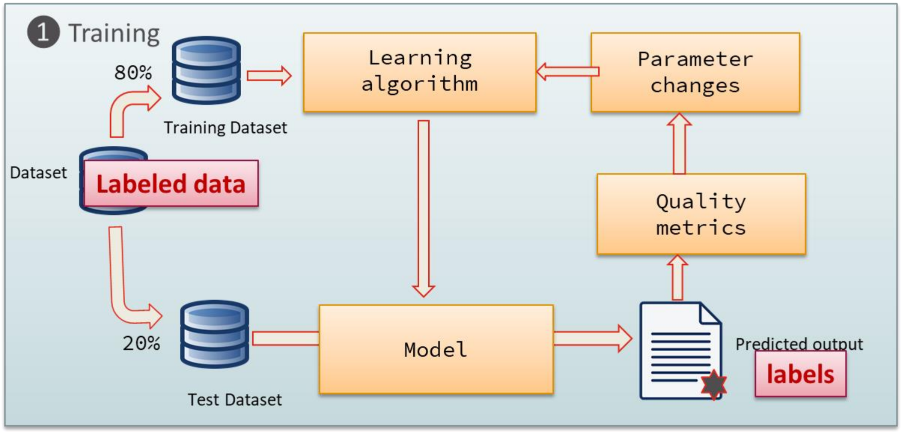
3 methods:

1. "testset validation" or "hold-out validation"
2. "k-fold cross-validation”
3. "leave-one-out cross-validation" (LOOCV).

“testset validation” or “hold-out validation”

The simplest method:

1. Partition the data randomly into a **training set** (usually > 60%) and a **validation set (hold-out set)**
2. For a set of chosen values for hyperparameters, learn a model on the training set
3. Compute the model’s error on the validation set (see metrics)
4. Pick the best hyperparameter which has the smallest validation set error and retrain the model



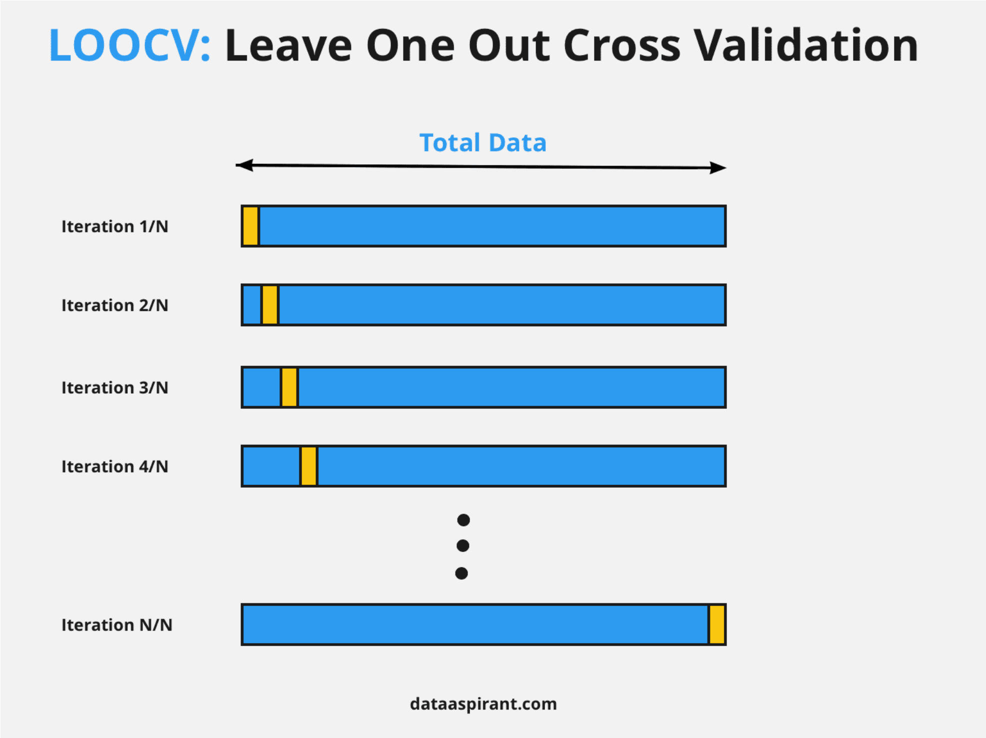
k-fold cross-validation

1. Randomly partition the training data into **K sets of equal size**
2. **Run the learning algorithm K times**: each time, a different one of the K sets is deemed the test set, and the model is trained on the remaining K-1 sets
3. The hyperparameter score is the average of the error across the K tests
4. **Pick the best hyperparameters** and **retrain** the model

Ashfaque, Johar & Iqbal, Amer. (2019). Introduction to Support Vector Machines and Kernel Methods.

"leave-one-out cross-validation" (LOOCV)

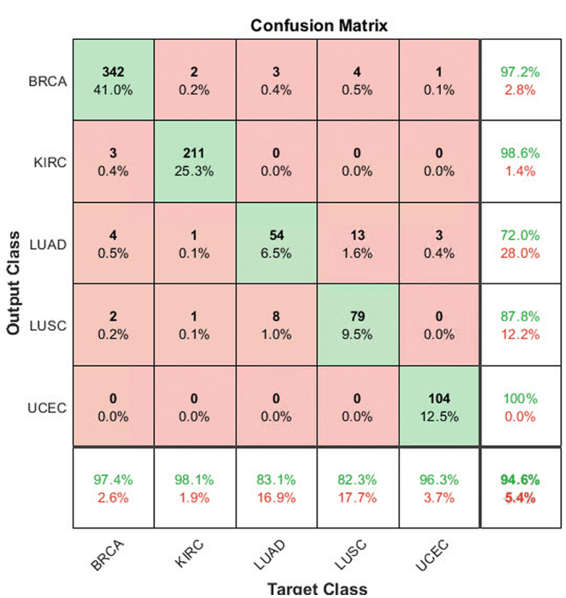
- K-fold cross validation with $K = M - 1$, with M the number of data points



Evaluate a ML algorithm with the confusion matrix

Confusion Matrix: describes the complete performance of the model

Predicted class		True class	
		Positive	**Negative**
	Positive	True positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)



Metrics to evaluate a ML algorithm

- **Classification Accuracy**: ratio of number of correct predictions to the total number of input samples.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

		True class	
		Positive	Negative
Predicted class	Positive	True positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Works well if there are equal number of samples belonging to each class.

Predicted class		True class	
		Positive	Negative
	Positive	True positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

- **Sensitivity/Recall (proba de detection)**: True Positive Rate, corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all positive data points.

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

Ex: Rapid COVID-19 antigen-test :
97,3% sensitivity

- **Specificity**: True Negative Rate, corresponds to the proportion of negative data points that are correctly considered as negative, with respect to all negative data points.

$$TNR = \frac{TN}{FP + TN}$$

Ex: Rapid COVID-19 antigen-test :
100% specificity

For example, if the sensitivity is 100% and the specificity is 50%, this means that all infected people will be detected as positive, however, many people who are not infected will be mistakenly identified as positive (false positives).

- **False Positive Rate**, corresponds to the proportion of negative data points that are mistakenly considered as positive, with respect to all negative data points.

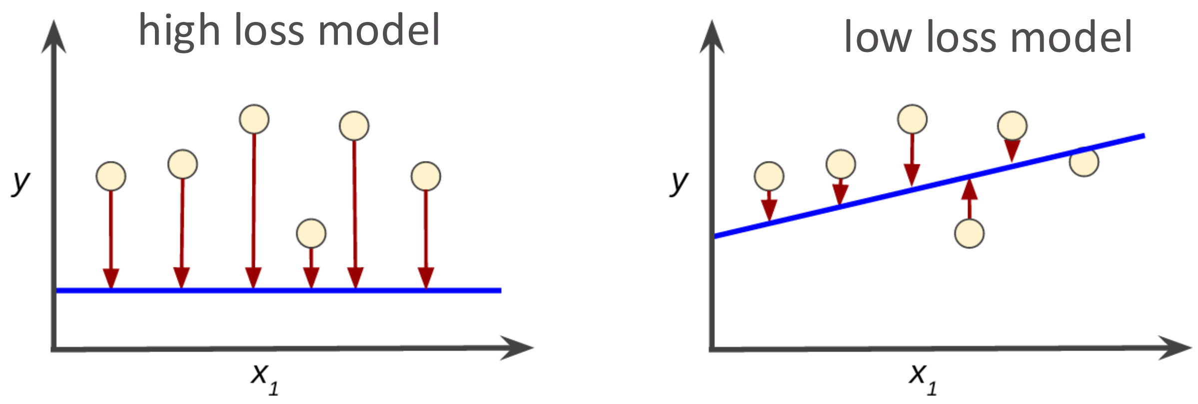
$$FPR = \frac{FP}{FP + TN}$$

Loss - Intuition

Loss is a number indicating how bad the model's prediction was on a single example.

If the model's prediction is perfect, the loss is zero; otherwise, the loss is greater.

The goal of training a model is to find a set of weights and biases that have **low** loss, on average, across all examples.



blue lines represent predictions

L1 and L2 loss

- $\boldsymbol{L_1}$ **loss: Least Absolute Deviation** (LAD) is used to minimize the error which is the sum of the all the **absolute** differences between the true value and the predicted value.

$$L_1 = \sum_{i=1}^{n} |\hat{y}_i - \mathrm{yi}|$$

- $\boldsymbol{L_2}$ **loss: Least Square Error** is used to minimize the error which is the sum of the all the **squared** differences between the true value and the predicted value.

$$L_2 = \sum_{i=1}^{n} (\hat{y}_i - \mathrm{yi})^2$$

Generally, L_2 is preferred in most of the cases (except when outliers are present in the dataset)

Note : It is possible to compute the L_1 loss or the L_2 loss for a <u>single example</u>

Regression loss functions

- **Mean Absolute Error (MAE) (empirical L_1 loss):** average of the difference between the original values yi and the predicted values $\hat{y}_i$. No info about the direction of the error (under or over prediction)

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |\hat{y}_i - \mathrm{yi}|$$

- **Mean Squared Error (MSE) Loss (empirical L_2 loss):** takes the average of the square of the difference between the original values and the predicted values. The goal is to reduce MSE

Advantages: easier to compute a gradient, whereas the MAE requires complicated computations

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (\hat{y}_i - \mathrm{yi})^2$$

- **Root Mean Square Error (RMSE)** $$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (\hat{y}_i - \mathrm{yi})^2}$$

Binary/Multi-class Classification Loss functions

- **Binary Cross-entropy**: measures the performance of a classification model whose output is a probability value between 0 and 1. Cross-entropy loss **increases as the predicted probability value deviates from the actual label**. *Binary Cross Entropy is the negative average of the log of corrected predicted probabilities*.

$$-\frac{1}{N}\sum_{i=1}^{N}(\log(p_i))$$

ID	Actual	Predicted probabilities	Corrected Probabilities	Log
ID6	1	0.94	0.94	-0.0268721464
ID1	1	0.90	0.90	-0.0457574906
ID7	1	0.78	0.78	-0.1079053973
ID8	0	0.56	0.44	-0.3565473235
ID2	0	0.51	0.49	-0.30980392
ID3	1	0.47	0.47	-0.3279021421
ID4	1	0.32	0.32	-0.4948500217
ID5	0	0.10	0.90	-0.0457574906

- **Log loss**: the same but does not computes

corrected probabilities

$$Log\ loss = \frac{1}{N}\sum_{i=1}^{N} -(y_i \log(p_i) + (1-y_i)\log(1-p_i))$$

- **Binary Cross-Entropy for Multi-class classification**, N numbers of rows, M number of classes

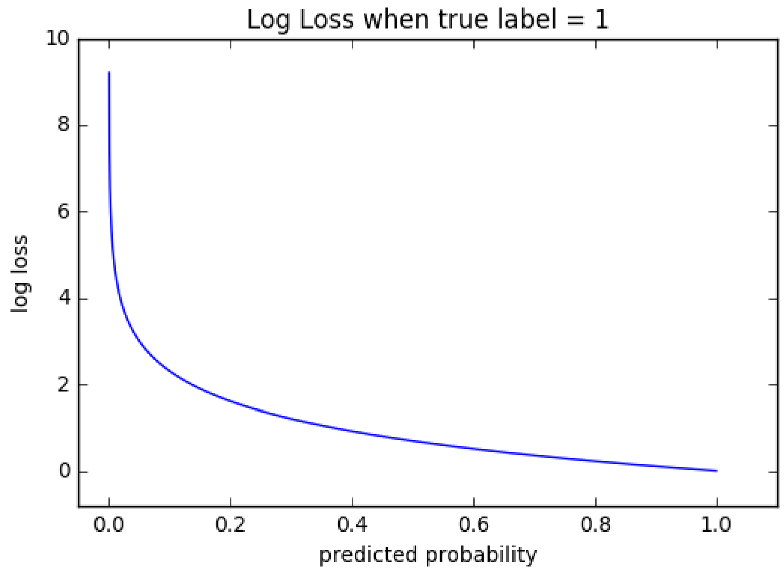
$$Log\ loss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} log(p_{ij})$$

https://ml-cheatsheet.readthedocs.io/en/latest/loss_functions.html

iquizz

My model predicts a probability of 0,012 when the actual observation label is 1.

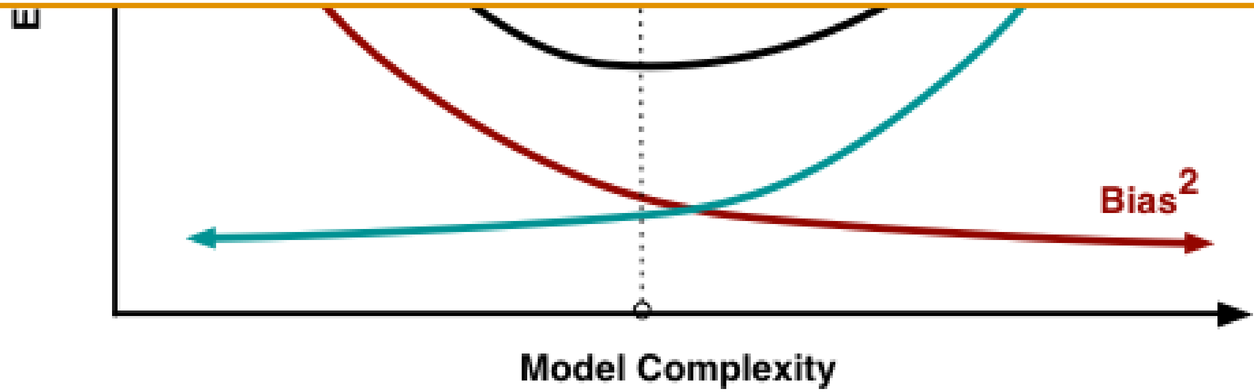
1. The value of the log loss will be close to 1
2. The value of the log loss will be greater than 1
3. The value of the log loss will be close to 0
4. The value of the log loss will be lower than 0

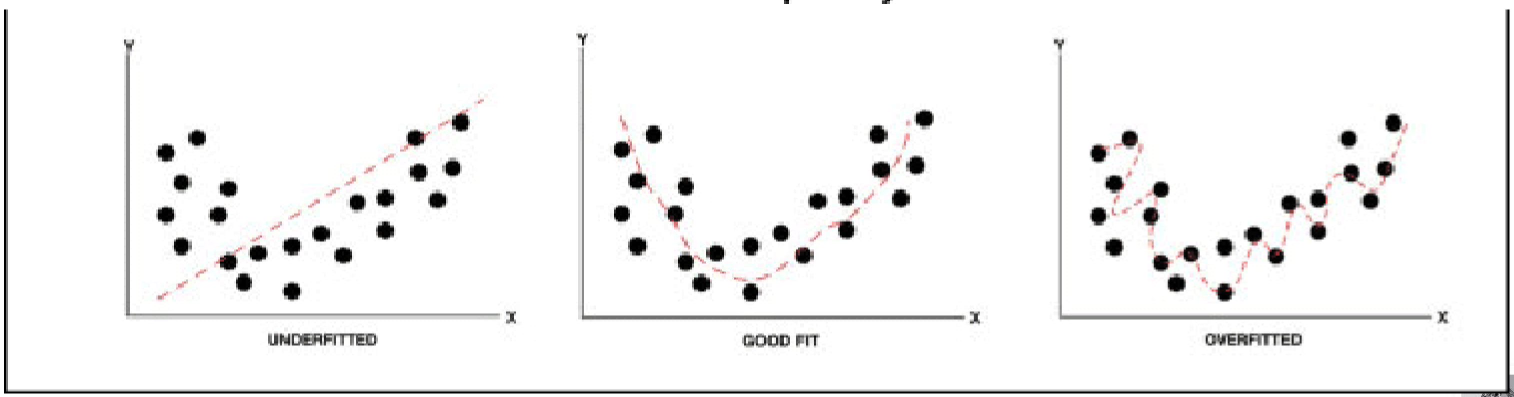


Bias-variance tradeoff

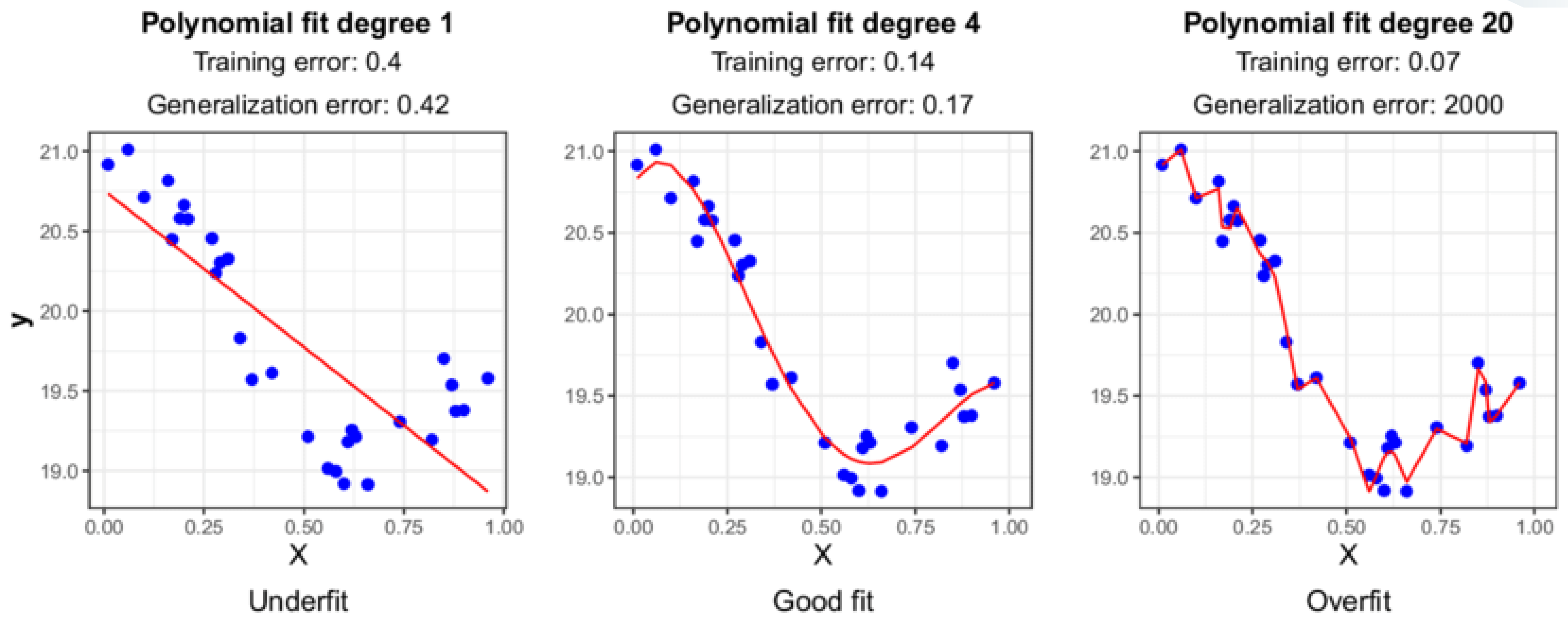
Bias: Let T be a statistic used to estimate a parameter θ . If $E(T) = \theta + b(\theta)$ then $b(\theta)$ is called the bias of the statistic T. $E(T)$ represents the expected value of the statistics T.

Variance: The variance is the square of the standard deviation σ, it can be seen as a measure of dispersion



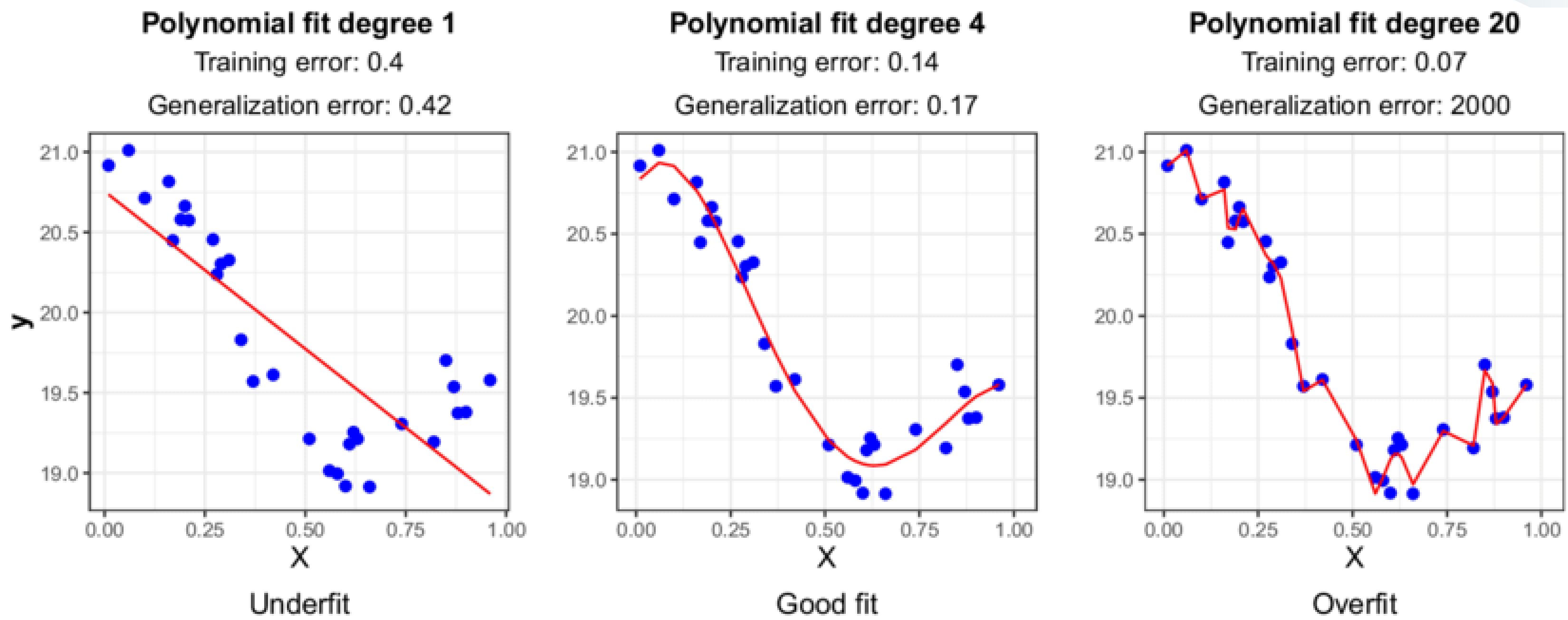


Overfitting- Underfitting



- Attempting to fit the data too carefully leads to **overfitting (low bias, high variance)**
  - If the desired output values are often incorrect (human errors or sensor noise), the learning algorithm attempts to find a function that exactly matches the training examples
  - If the model has too many parameters
- **Solutions:** early stopping, detecting and removing the noisy training examples prior training, smoothness assumption (either with the model directly, or with regularization terms in the optimization criterion)

Overfitting- Underfitting



- When a model is not complex enough to accurately capture relationships between features and target variables, it leads to **underfitting (high bias, low variance)**
- **Solutions:**

  Decrease regularization term used to reduce the variance

  Increase the duration of training

  Feature selection

  Increase the size of the model (more hidden neurons, add more trees…)

Supervised learning: a guiding tour