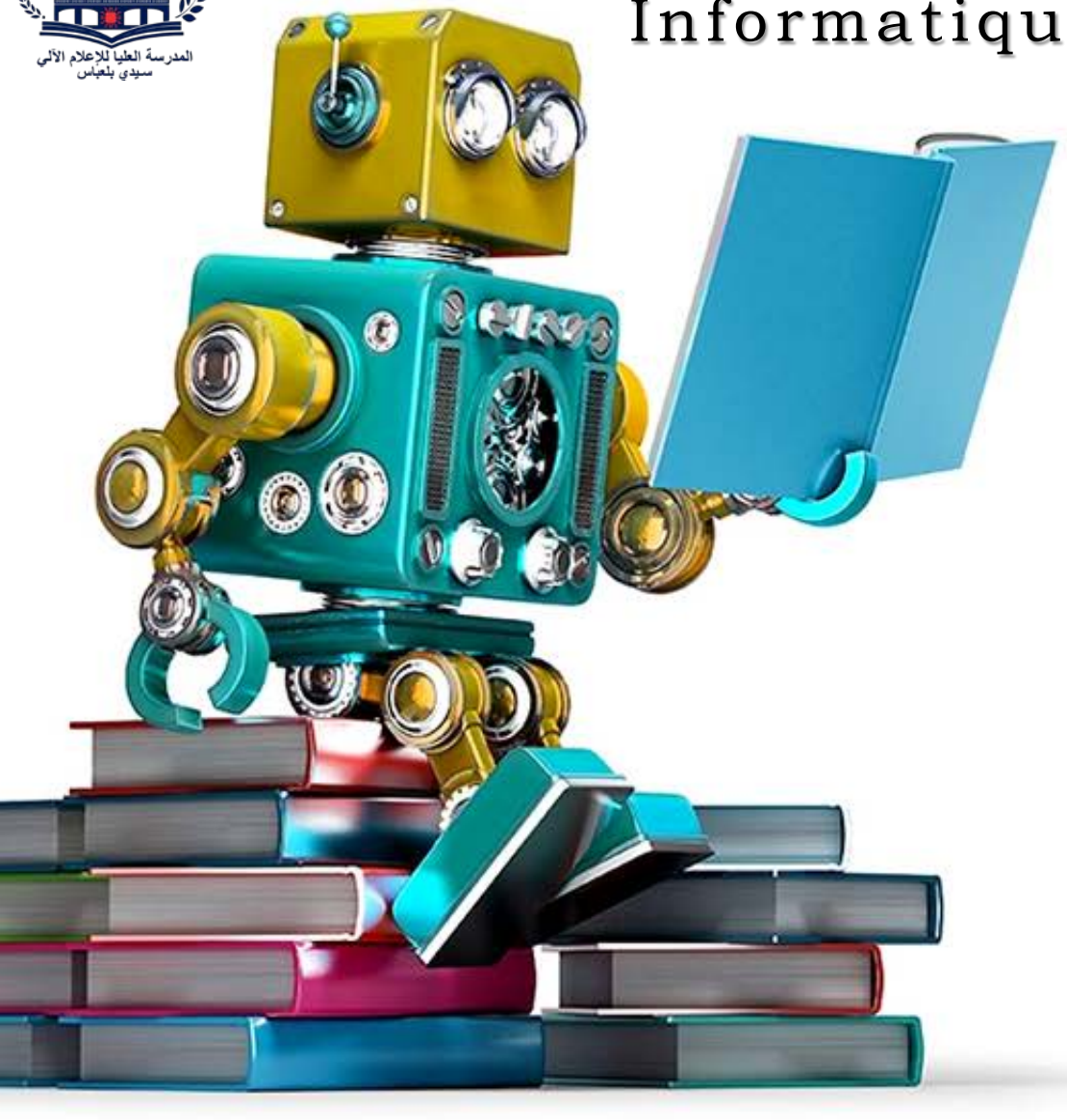
ESI-SBA - École Supérieure en Informatique 08-MAI-1945

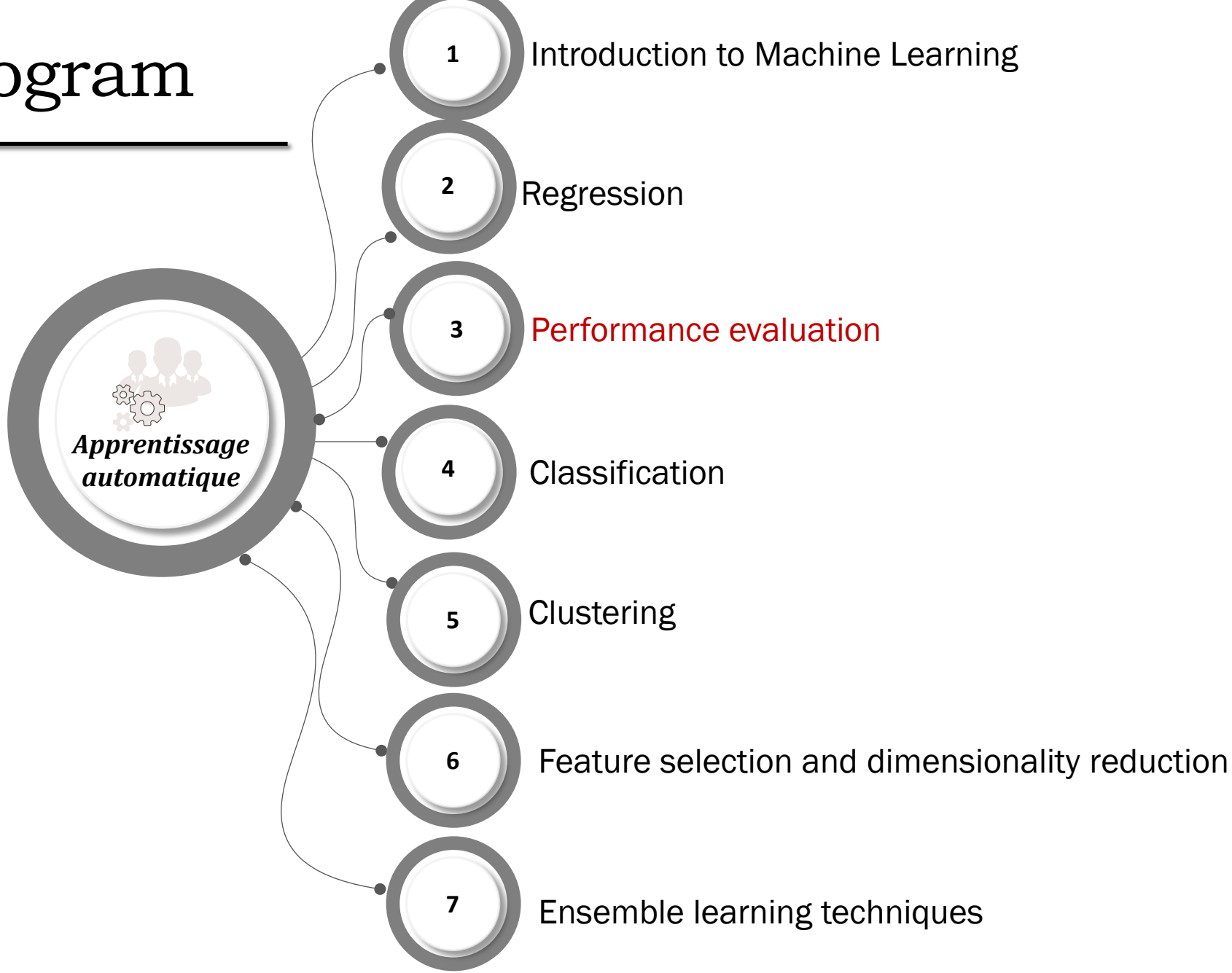


Machine Learning

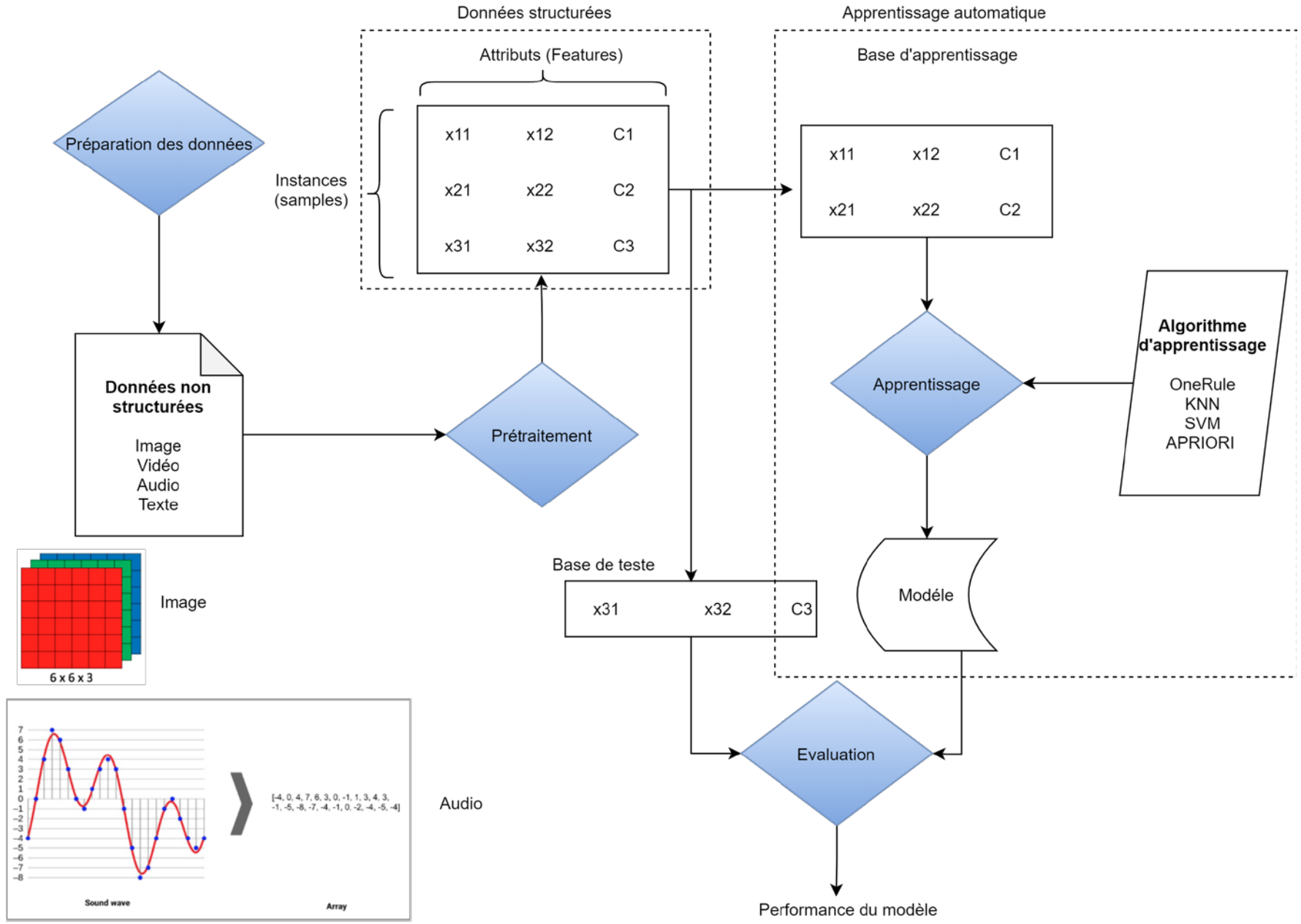
Intelligence Artificielle et Sciences de Données (IASD)

DR N. DIF

Program


Apprentissage automatique
1
Introduction to Machine Learning
2
Regression
3
Performance evaluation
4
Classification
5
Clustering
6
Feature selection and dimensionality reduction
7
Ensemble learning techniques

Rappel

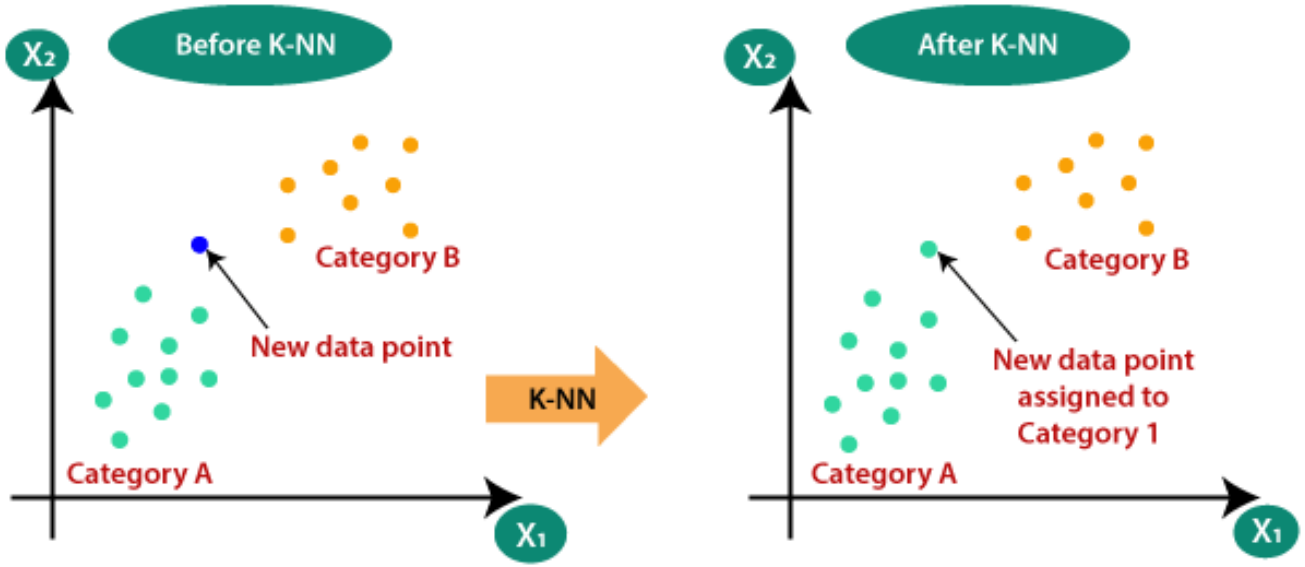

Données structurées
Attributs (Features)
Instances (samples)
x11 x12 C1
x21 x22 C2
x31 x32 C3
Préparation des données
Données non structurées
Image
Vidéo
Audio
Texte
Prétraitement
Apprentissage automatique
Base d'apprentissage
x11 x12 C1
x21 x22 C2
Apprentissage
Algorithme d'apprentissage
OneRule
KNN
SVM
APRIORI
Modéle
Base de teste
x31 x32 C3
Evaluation
Performance du modèle
Image
6 x 6 x 3
Audio
Sound wave
Array

1. The k-nearest neighbors (knn)

IM LAZY

1.1. Definition

- Instance-based supervised algorithm.
- It can be used for both Regression and Classification problems.
- Belongs to the category of lazy learners (does not generates a model for prediction).
- Based on distances to select the closest samples to the instance to predict.
- K presents the number of close instances to be selected.
- Selects the predicted class through a majority voting procedure.



From : https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

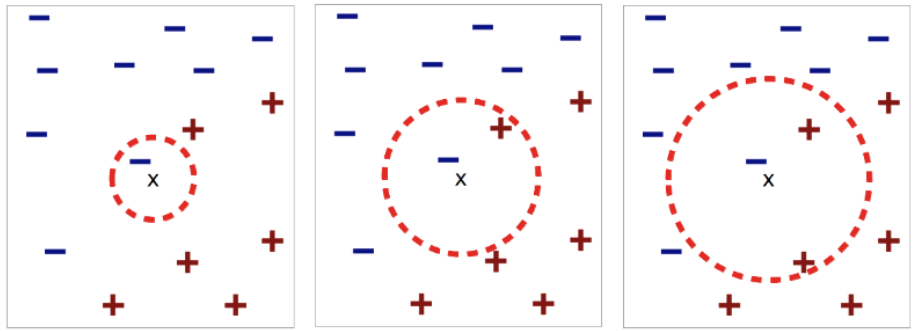
Does the KNN algorithm considered eager learners ?

1. THE K-NEAREST NEIGHBOURS (KNN)

1.2. Algorithm

KNN Algorithm

Consider the dataset D = $\{x_i \mid y_i\}$, $i\epsilon\{1,..n\}. y_i\epsilon\{c_1, \dots, c_k\}$.
Consider S as an sample to predict.
Convert categorical into numerical data.
Normalize all data using scaling techniques
Select the number K of the neighbors
For each sample S_i in the dataset D
    # S presents the sample to predict
    Compute the distance between S_i and S
Select the K nearest samples to S that minimize the distance.
Compute the number of nearest samples in each class c .
Assign the class that contains the largest number of nearest samples (majority voting) to S.



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

1. THE K-NEAREST NEIGHBOURS (KNN)

1.3. The distance

Consider two points, X and Y, each with coordinates (x1, .., xn) and (y1, .., yn) within a parameter space $\mathbb{R}^n$.

Manhattan distance : $\sum_{i=1}^{n} |X_i - Y_i|$

Euclidien distance : $\sqrt{\sum_{j=1}^{n} (X_i - Y_i)^2}$

Compute the Manhattan and the Euclidean distance between the two samples s1 = {1.2, 0.5, 15} and s1 = {1.8, 0.6, 17}

1. THE K-NEAREST NEIGHBOURS (KNN)

1.4. Preprocessing

Does the KNN handle categorical values ?

Impute missing (NaN) values.

Does the KNN handle categorical values ?

When the data contains different types of features (numerical and categorical), there are two approaches that can be used :

1. Convert categorical data into numerical representation
2. Calculate the Euclidean distance for numeric data and calculate hamming distance for categorical data, and then combine both distances.

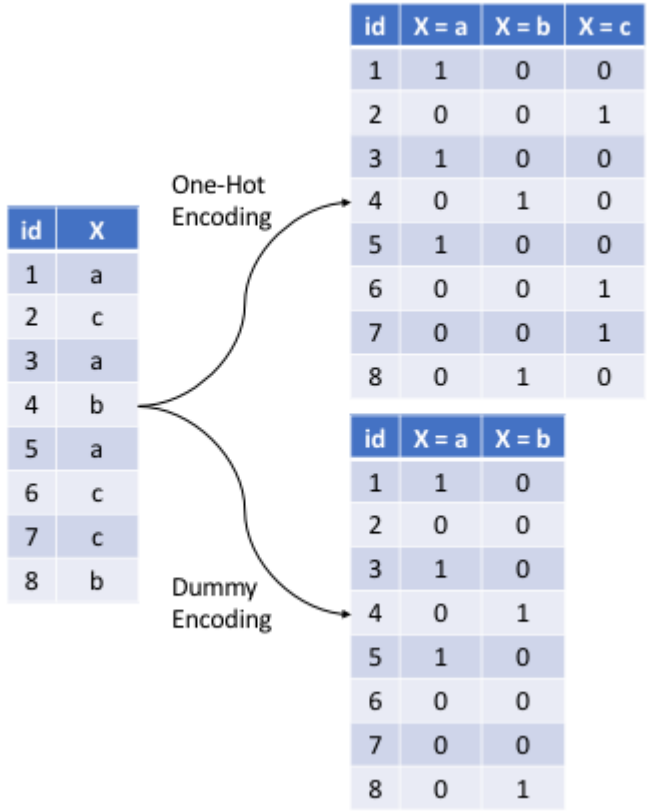
1. THE K-NEAREST NEIGHBOURS (KNN)

1.4. Preprocessing : Convert categorical data into numerical representation

Convert categorical data into numerical representation : there are three common approaches : ordinal encoding, one-hot encoding, and dummy variable encoding

1. **Ordinal encoding :** recommended for categorical variables that can be ordered. By default, it will assign integers to labels in a specific order. Example : Red : 1, Green : 2, Blue : 3.

2. **One-hot encoding :** for categorical variables where no ordinal relationship exists. In such method, one binary features will be created for each value. Each category is mapped with a binary variable containing either 0 or 1. Here, 0 represents the absence, and 1 represents the presence of that category
3. **Dummy variable encoding :** The dummy encoding is a small improvement over one-hot-encoding. Dummy encoding uses N-1 features to represent N labels/categories.



1. THE K-NEAREST NEIGHBOURS (KNN)

1.4. Preprocessing : Data normalization

In general, distance based algorithms are **influenced by the scale of the variables**. For instance, the distance $\in \{0,1\}$ for symbolic values transformed by the one-hot encoding technique, while, for numerical values, the distance $\in [0, +\infty]$. This will give a **high weight to variables with higher magnitude** (numerical values in our case). As a solution, all variable are transformed to the same scale. Normalization is among the exploited scaling techniques for this problem. Among the well known **normalization techniques** : z-score normalization, and Min-Max normalization.

- Min-Max normalization: it scales the values of a feature to a range between 0 and 1. $V_{ijnorm} = \frac{V_{ij} - V_{imin}}{V_{imax} - V_{imin}}$, V_{ij} presents the value j of feature A_i, V_{imin} and V_{imax} re minimal and maximal values of the feature A_i, and V_{ijnorm} is the normalized value of V_{ij}.

- Z-score normalization: This technique scales the values of a feature to have a mean of 0 and a standard deviation of 1. $V_{ijnorm} = \frac{V_{ij} - \overline{A_i}}{\sigma_{A_j}}$

Predict the class of : Sunny 66 76 Weak

Outlook	Temperature	Humidity	Wind	PlayTennis
Sunny	85	85	Weak	No
Sunny	80	90	Strong	No
Overcast	83	78	Weak	Yes
Rain	70	96	Weak	Yes
Rain	68	80	Weak	Yes
Rain	65	70	Strong	No
Overcast	64	65	Strong	Yes
Sunny	72	95	Weak	No
Sunny	69	70	Weak	Yes
Rain	75	80	Weak	Yes
Sunny	75	70	Strong	Yes
Overcast	72	90	Strong	Yes
Overcast	81	75	Weak	Yes
Rain	71	80	Strong	No

1. THE K-NEAREST NEIGHBOURS (KNN)

1.5. Case Study: Play Tennis Dataset

1- Preprocessing : Ordinal Encoding for Outlook and Wind, and Min- Max normalisation for Temperature and Humidity.
For outlook : Sunny = 0; Overcast = 1;Rain = 2.
For wind : Weak = 0, Strong = 1.
Predict the class of : Sunny 66 76 Weak ➔ 0 0.095 0.35 0 using the Manhattan distance

Outlook	Temperature	Humidity	Wind	PlayTennis
Sunny	85	85	Weak	No
Sunny	80	90	Strong	No
Overcast	83	78	Weak	Yes
Rain	70	96	Weak	Yes
Rain	68	80	Weak	Yes
Rain	65	70	Strong	No
Overcast	64	65	Strong	Yes
Sunny	72	95	Weak	No
Sunny	69	70	Weak	Yes
Rain	75	80	Weak	Yes
Sunny	75	70	Strong	Yes
Overcast	72	90	Strong	Yes
Overcast	81	75	Weak	Yes
Rain	71	80	Strong	No

Encoding

Outlook	Temperature	Humidity	Wind
1	85	85	1
1	80	90	2
2	83	78	1
3	70	96	1
3	68	80	1
3	65	70	2
2	64	65	2
1	72	95	1
1	69	70	1
3	75	80	1
1	75	70	2
2	72	90	2
2	81	75	1
3	71	80	2

1. THE K-NEAREST NEIGHBOURS (KNN)

1.5. Case Study: Play Tennis Dataset

Normalization

Outlook	Temperature	Humidity	Wind
0	1	0.65	0
0	0.76	0.81	1
0.5	0.90	0.42	0
1	0.29	1.00	0
1	0.19	0.48	0
1	0.05	0.16	1
0.5	0.00	0.00	1
0	0.38	0.97	0
0	0.24	0.16	0
1	0.52	0.48	0
0	0.52	0.16	1
0.5	0.38	0.81	1
0.5	0.81	0.32	0
1	0.33	0.48	1

Outlook	Temperature	Humidity	Wind	Distance
0	1	0.65	0	**1.20**
0	0.76	0.81	1	2.12
0.5	0.90	0.42	0	1.38
1	0.29	1.00	0	1.84
1	0.19	0.48	0	1.23
1	0.05	0.16	1	2.24
0.5	0.00	0.00	1	1.95
0	0.38	0.97	0	**0.90**
0	0.24	0.16	0	**0.33**
1	0.52	0.48	0	1.56
0	0.52	0.16	1	1.62
0.5	0.38	0.81	1	2.24
0.5	0.81	0.32	0	1.24
1	0.33	0.48	1	2.37

1. THE K-NEAREST NEIGHBOURS (KNN)

2.6. Disadvantages

- High prediction time: is linear to the number of samples in the training dataset.
- High storage capacity, as the training dataset must be loaded into RAM during prediction, which can be problematic for embedded vision systems if the training dataset is large.

2. NAIVE BAYES

IM NAIVE

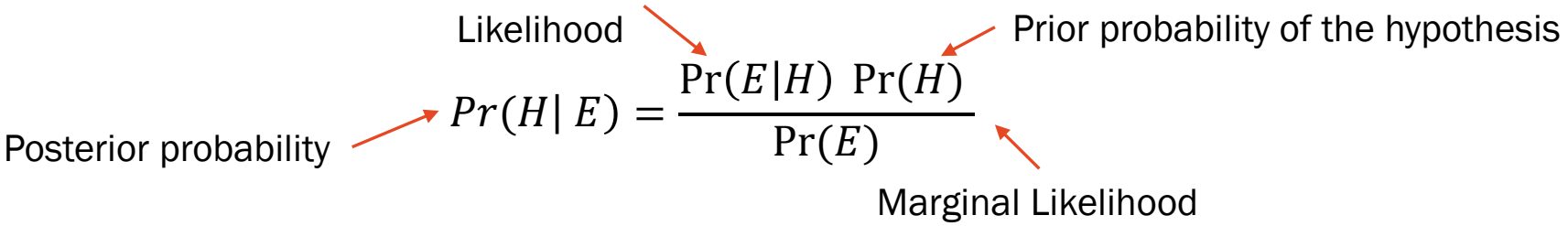
2.1. Definition

- A probabilistic classifier used in supervised learning.
- Based on Bayes' theorem.
- Generally used for solving high-dimensional problems, such as text classification problems.
- Naïve ? Because, the Bayes' theorem assumes that all features are independent.

2. NAIVE BAYES

2.1. The bayes rule

Given an hypothesis H and an evidence E, the bayes rule is defined as follow :



Posterior probability $(Pr(H|E))$ **:** the probability of the hypothesis H given the observed event E (not directly computable).
Likelihood $(\Pr(E|H))$ **:** the probability of the evidence E given that the hypothesis H is true.
Prior probability of the hypothesis $(\Pr(H))$ **:** the probability of the hypothesis before observing the evidence.
Marginal Likelihood $(\Pr(E))$**:** the probability of the evidence under all hypotheses.

In machine learning, it can be reformulated as follow : $Pr(C_k|X) = \frac{\Pr(X|C_k)\ \Pr(C_k)}{\Pr(X)}$

Where X = $[x_1, x_2, \ldots, x_m]$, with m the number of features, X present the sample to predicts and C_k is a class, k$\in \{1, \ldots, c\}$, c is the number of classes.
Example : Pr (Yes|Rain Mild High Strong) .

2. NAIVE BAYES

2.2. Compute probabilities for discerte values

$$Pr(C_k|X) = \frac{\Pr(X|C_k)\ \Pr(C_k)}{\Pr(X)}$$

$$Pr(C_k|X) = \frac{\prod_{i=1}^{m} Pr(x_i|C_k)\ .\Pr(C_k)}{\Pr(X)}$$

For discrete values : $Pr(x_{ij}|C_k) = \frac{|x_{ij_k}|}{n_k}$, $|x_{ij_k}|$ presents the number of samples with the value x_j for the i^{th} features that belongs to the class C_k. n_k is the number of samples associated to the class k.

Outlook	Temperature	Humidity	Wind	Play Tennis
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

Example : $Pr(Outlook = Sunny|No) = \frac{3}{5}$

$$Pr(Temperature = Mild|Yes) = \frac{4}{9}$$

2. NAIVE BAYES

2.3. Gaussian Naive Bayes for continuous values

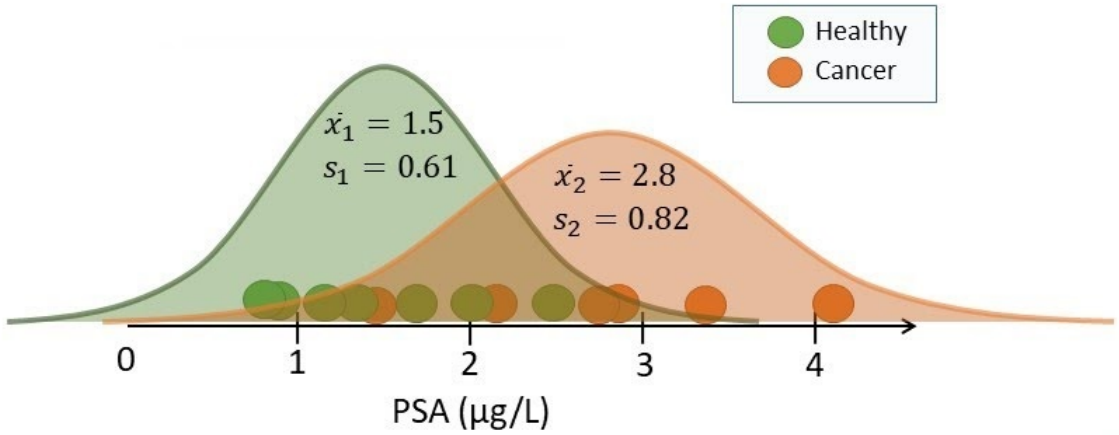
For continuos values, there are two possibilities :

1- Discretization : convert continuous values into discrete values.

2- Use the Gaussian Naïve Bayes variant.

- The Gaussian Naïve Bayes considers that the numerical values of a feature A_i follow a normal or Gaussian distribution.

Status	PSA
Cancer	4.1
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8



2. NAIVE BAYES

2.3. Gaussian Naive Bayes for continuous values

- The Gaussian Naïve Bayes compute the probability of a numerical value v_{ij} based on the probability density function :

$$P(v_{ij}\,|C_k) = \frac{1}{\sigma_k\sqrt{2\pi}}\, e^{-\frac{1}{2}(\frac{v_{ij}-\mu_k}{\sigma_k})^2}$$

σ_k and μ_k present the sample standard deviation and the mean of values that belongs to the feature A_i and the class k (ps : for each class σ_k and μ_k are computed.

$$\sigma_k = \sqrt{\frac{\sum_{j=1}^{n}|v_{ij}-\mu_k|^2}{n-1}}$$

$\mu_k = \frac{\sum_{j=1}^{n} v_{ij}}{n}$, n is the number of samples that belongs to the class C_k

2. NAIVE BAYES

2.4. PREDICTION

To predict the class of an instance X, after predicting the final posterior probabilities , their values can be standardized between 0 and 1 :

$\Pr\_normalized(C_k|X) = \frac{\frac{\Pr(X|C_k)\ \Pr(C_k)}{\Pr(X)}}{\frac{\sum_{i=1}^{c} P(X|C_i)}{\Pr(X)}}$, c presents the number of classes.

The class that maximize the probability is selected :

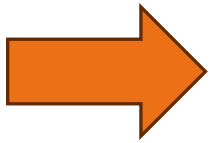
$$C_{max} = argmax_{c_k \epsilon C}\ \Pr\_normalized(C_k|X)$$

2. NAIVE BAYES

2.5. Case Study: Play Tennis Dataset

1- Train the model (compute probabilities)

Outlook	Temperature	Humidity	Wind	Play Tennis
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No



Feature	Values	Probabilities	
		No	Yes
Outlook	Sunny	3/5	2/9
	Overcast	0/5	4/9
	Rain	2/5	3/9
Temperature	Hot	2/5	2/9
	Mild	2/5	4/9
	Cool	1/5	3/9
Humidity	High	4/5	3/9
	Normal	1/5	6/9
Wind	Strong	3/5	3/9
	Weak	2/5	6/9
Play	Yes	9/14	
	No	5/14	

2. NAIVE BAYES

2.5. Case Study: Play Tennis Dataset

2- Predict the class of the sample : Rain Mild High Strong

$$\Pr(\text{Yes} \mid \text{Rain Mild High Strong}) = \frac{\Pr(\text{Rain Mild High Strong} \mid \text{Yes}\ )\ \Pr(Yes)}{\Pr(\text{Rain Mild High Strong})}$$

$$\Pr(\text{Yes} \mid \text{Rain Mild High Strong}) = \frac{\Pr(\text{Rain} \mid \text{Yes})\ \Pr(\text{Mild} \mid \text{Yes})\ \Pr(\text{High} \mid \text{Yes})\ \Pr(\text{Strong}) \mid \text{Yes}\ )\Pr(Yes)}{\Pr(\text{Rain Mild High Strong})}$$

$$\Pr(\text{Yes} \mid \text{Rain Mild High Strong}) = \frac{\frac{3}{9} * \frac{4}{9} * \frac{3}{9} * \frac{3}{9} * \frac{9}{14}}{\Pr(\text{Rain Mild High Strong})} = \frac{\mathbf{0.0106}}{\Pr(\text{Rain Mild High Strong})}$$

$$\Pr(\text{No} \mid \text{Rain Mild High Strong}) = \frac{\frac{2}{5} * \frac{2}{5} * \frac{4}{5} * \frac{3}{5} * \frac{5}{14}}{\Pr(\text{Rain Mild High Strong})} = \frac{\mathbf{0.0274}}{\Pr(\text{Rain Mild High Strong})}$$

$$\text{Pr\_normalized}(\text{Yes} \mid \text{Rain Mild High Strong}) = \frac{\mathbf{0.0106}}{\mathbf{0.0106} + 0.0274} = 15\%$$

$$\text{Pr\_normalized}(\text{No} \mid \text{Rain Mild High Strong}) = \frac{0.0274}{\mathbf{0.0106} + 0.0274} = 85\%$$

According to the obtained results, the predicted class of Rain Mild High Strong is $C_{max} = \text{No}$

2. NAIVE BAYES

2.6. Laplace Smoothing

In Naive Bayes, the probability of a feature can neutralize the significance of the probabilities of the other features if it has never been seen in that class during training . As a solution, Laplace smoothing is used to address the problem of zero probabilities for certain events in the training data :

For instance : $\Pr(\text{No} \mid \text{Overcast Mild High Strong}) = \frac{\frac{0}{5} * \frac{2}{5} * \frac{4}{5} * \frac{3}{5} * \frac{5}{14}}{\Pr(\text{Rain Mild High Strong})} = \mathbf{0}$

- The probability of the sample : Overcast Mild High Strong | No is always Regardless of the values of other probabilities due to Pr(Overcast| No)!!!
- **Solution :** Laplace smoothing.
- The new probabilities based on Laplace smoothing are computed as follow :

$new\_P(A_i = v_{ij} | C_k) = \frac{\left|x_{ij_k}\right| + 1}{n_k + s_i}$, s_i is the number of possible values for A_i .

2. NAIVE BAYES

2.7. Case Study: Play Tennis Dataset (numerical version) with mixed Naive Bayes.

Outlook			**Temperature**		Humidity		**Wind**			**Play**	
Val	Yes	No	Yes	No	Yes	No	Val	Yes	No		
Sunny	2	3	83 70 68 64 69 75 75 72 81	85 80 65 72 71	86 96 80 65 70 80 70 90 75	85 90 70 95 91	Strong	3	3	Yes	9
Overcast	4	0					Weak	6	2	No	5
Rainy	3	2									
Sunny	2/9	3/5	$\mu = 73$	$\mu = 74.6$	$\mu = 79.1$	$\mu = 86.2$	Strong	3/9	3/5	Yes	9
Overcast	4/9	0/5	$\sigma = 6.2$	$\sigma = 7.9$	$\sigma = 10.2$	$\sigma = 9.7$	Weak	6/9	2/9	No	5
Rainy	3/9	2/5									

Example : if température = 66 then $P(66|\ yes) = \frac{1}{6.2\sqrt{2*\ 3.14}} e^{\frac{-1}{2}(\frac{66-73}{6.2})^2} = 0.036$

2. NAIVE BAYES

2.8. Conclusion

Advantages

It handle missing values by ignoring the probability of the value.

Low computational complexity during prediction.

Disadvantages

It assumes that all attributes are independent but in real life this is not true, because there are some features that are highly correlated.