Pattern Recognition

Presentation

Benmoussa Younes

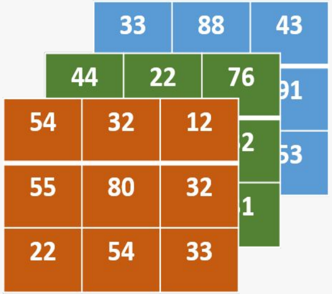
Table of contents

Introduction

A pattern is Everything around in this digital world

Introduction

In Computer Science, a pattern is represented using vector feature values



Pattern Recognition

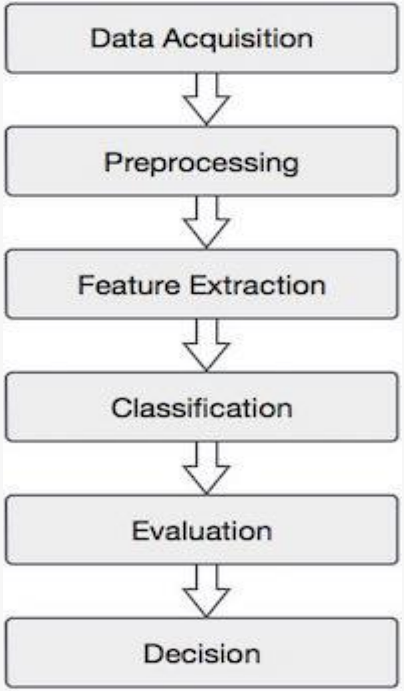
Pattern Recognition

Pattern recognition is a broader term that refers to the process of
➡ identifying and discovering patterns or regularities in data.

Pattern recognition is a technique to classify input data into classes or objects by recognizing patterns or feature similarities

pattern recognition looks for a “most likely” pattern to classify all information provided.

Pattern recognition schematic



Example of pattern recognition applications

Problem Domain	Application	Input Pattern	Pattern Classes
Bioinformatics	Sequence analysis	DNA/Protein sequence	Known types of genes/ patterns
Data mining	Searching for meaningful patterns	Points in multi-dimensional space	Compact and well-separated clusters
Document classification	Internet search	Text document	Semantic categories (e.g., business, sports, etc.)
Document image analysis	Reading machine for the blind	Document image	Alphanumeric characters, words
Industrial automation	Printed circuit board inspection	Intensity or range image	Defective / non-defective nature of product
Multimedia database retrieval	Internet search	Video clip	Video genres (e.g., action, dialogue, etc.)
Biometric recognition	Personal identification	Face, iris, fingerprint	Authorized users for access control
Remote sensing	Forecasting crop yield	Multispectral image	Land use categories, growth pattern of crops
Speech recognition	Telephone directory enquiry without operator assistance	Speech waveform	Spoken words

Fundamentals

- Data
- Characteristic vector
- Selection of features
- Class

Fundamentals

Data

They represent all the information relating to a subject or object of study.

Characteristic vector

one or more characteristic data or properties that have been grouped, structured and coded

Selection of features

A "good" feature vector is representative and discriminating

Fundamentals

class

A class
teaches about a set of objects or samples of the same
kind and having the same vector
descriptor as a description model

Pattern Recognition Methods

TS

There are two main approaches to pattern recognition:

- **Statistical Methods** : feature extraction produces numerical values .
- **Stochastic Methods** : The extraction of primitives produces symbolic values and relations that are the subject of structural or syntactic analysis.

Statistical Methods for pattern recognition.

Statistical Methods

Parametric methods

the classification of a method
as parametric
depends entirely
on the presumptions made
about a population

Non-parametric methods

Non-parametric methods
are not population-dependent. Here, no
fixed set
of parameters is available, and no
distribution (normal distribution, etc.)
is available.

Méthodes paramétriques	Méthodes non paramétriques
Les méthodes paramétriques utilisent un nombre fixe de paramètres pour créer le modèle.	Les méthodes non paramétriques utilisent le nombre flexible de paramètres pour construire le modèle.
Méthodes paramétriques supposées être une distribution normale.	Il n'y a pas de distribution supposée dans les méthodes non paramétriques.
Les méthodes paramétriques peuvent bien fonctionner dans de nombreuses situations, mais leurs performances sont optimales (supérieures) lorsque la répartition de chaque groupe est différente.	Les méthodes non paramétriques peuvent bien fonctionner dans de nombreuses situations, mais leurs performances sont optimales (supérieures) lorsque la répartition de chaque groupe est la même.
Ces méthodes sont plus rapides en termes de calcul que les méthodes non paramétriques.	Ces méthodes sont plus lentes que les méthodes paramétriques.
L'analyse paramétrique consiste à tester les moyennes des groupes.	Une analyse non paramétrique consiste à tester les médianes.
Exemples : modèle naïf de Bayes, etc.	**Exemples** : KNN, modèle d'arbre de décision, etc.

Naive Bayes Classifier

Naïve Bayes is a probabilistic machine learning algorithm that can be used in a wide variety of classification tasks.

the naive name is used because it assumes that the functionalities entering the model are independent of each other.

- Spam filtering - Document classification - Sentiment prediction

Naive Bayes Classifier Example

Imagine we received a normal messages from friends.

And we also received Spam
(unwanted messages that are usually scams or unsolicited advertisements) . . .

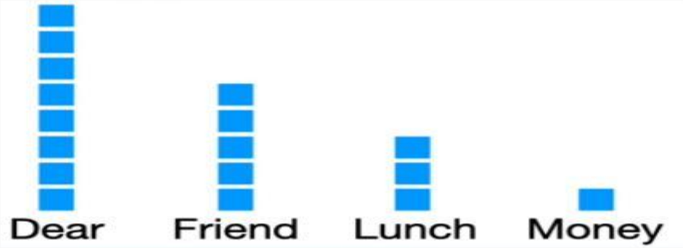
We wanted to filter out the Spam messages.

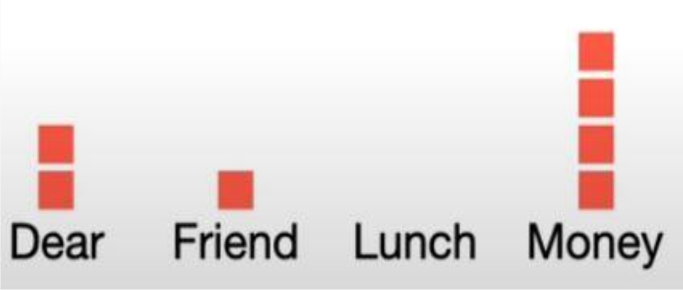
Naive Bayes Classifier Example

So, the first thing we do is make a histogram of all the words that occur in the normal messages

Naive Bayes Classifier Example

So, the first thing we do is make a histogram of all the words that occur in the normal messages

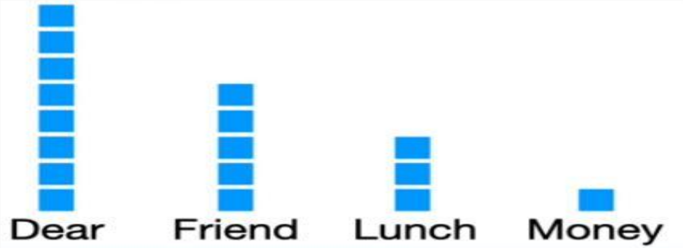


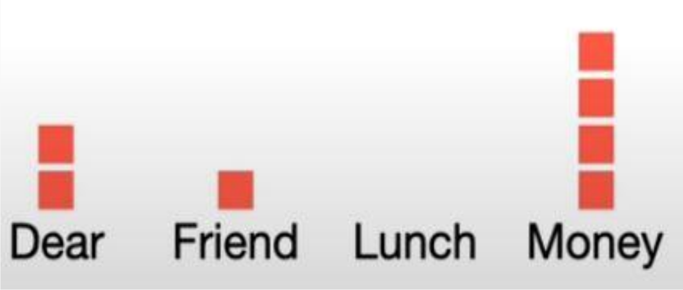


Naive Bayes Classifier Example

So, the first thing we do is make a histogram of all the words that occur in the normal messages

We can use the histogram to calculate the probabilities Of seeing each word

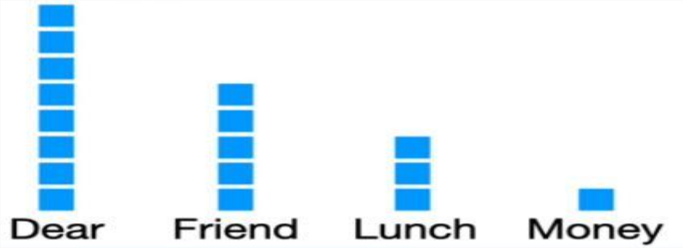


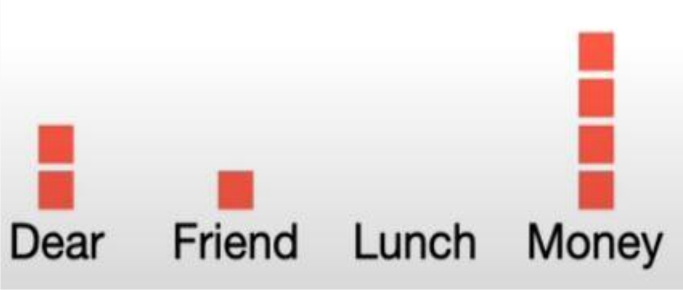


Naive Bayes Classifier Example

So, the first thing we do is make a histogram of all the words that occur in the normal messages

We can use the histogram to calculate the probabilities Of seeing each word





Total number of words in all of

→ The normal messages : 17

→ The Spam messages : 7

Naive Bayes Classifier Example

Normal Messages

8	5	3	1
Dear	Friend	Lunch	Money

$P(\text{Dear} \mid \text{Normal}) = 8 / 17 = 0.47$

$P(\text{Friend} \mid \text{Normal}) = 5 / 17 = 0.29$

$P(\text{Lunch} \mid \text{Normal}) = 3 / 17 = 0.18$

$P(\text{Money} \mid \text{Normal}) = 1 / 17 = 0.06$

Dear 8 time on the Nm
17 total number of worlds
On the normal message

Naive Bayes Classifier Example

Spam Messages →

Dear	Friend	Lunch	Money
2	1	0	4

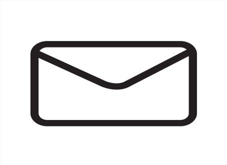
$P(\text{Dear} \mid \text{Spam}) = 2/7 = 0.29$

$P(\text{Friend} \mid \text{Spam}) = 1/7 = 0.14$

$P(\text{Lunch} \mid \text{Spam}) = 0/7 = 0.00$

$P(\text{Money} \mid \text{Spam}) = 4/7 = 0.57$

Now, Imagine that we got a new message that said :



Dear Friend

And we want to decide if it is a normal message or spam .

Now, Imagine that we got a new message that said :

Dear Friend

And we want to decide if it is a normal message or spam .

$$P\,(\,Y \mid \textbf{Dear Friend}\,)$$

normal spam

For what value of y

$$P (Y = y \mid X = (x1, x2, \ldots, xn))$$

Is maximum

Dear Friend

But … $P(Y \mid X)$ is hard to find !

Bayes' rule

Bayes' rule is a way of using **P(X|Y)**, known from the training data set, to find **P(Y|X)**.

- X represents features

P(X|Y)

Data Set

P (Dear) = 10 / 24 = 0.41	P (Dear | Normal) = 0.47	P (Dear | Spam) = 0.29
P (Friend) = 6 /24 = 0.25	P (Friend | Normal) = 0.29	P (Friend | Spam) = 0.14
P (Lunch) = 3 /24 = 0.125	P (Lunch | Normal) = 0.18	P (Lunch | Spam) = 0.00
P (Money) = 5/24 = 0.208	P (Money | Normal) = 0.06	P (Money | Spam) = 0.57

Bayes' rule

→ X represent features
→ Y represent labels (classes)

$$P(Y|X) = P(X|Y) \times P(Y) \,/\, P(X)$$

Bayes' rule

Likelihood

$$P(Y|X) = P(X|Y) \times P(Y) \, / \, P(X)$$

Evidence (“ states of features “

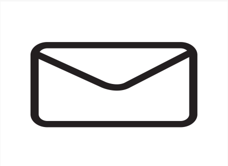
Probability of that after considering some evidence

Posterior

Prior

Probability of an event before considering any evidence

Now, Imagine that we got a new message that said :



Dear Friend

And we want to decide if it is a normal message or spam .

- For Example since 8 of the 12 messages are normal messages ,

Our initial guess will be 0.67

1. We calculate The Initial guess $P(N) = 8 / 8+4 = 0.67$

! The Initial guess That we observe a normal Message is called a **Prior probability** event

2. Now, we multiply that initial guess by the probability that the word **Dear** occurs in a normal message . . .

$$P(N) \times P(\textbf{Dear} \mid N)$$

2. Now, we multiply that initial guess by the probability that the word **Dear** occurs in a normal message . . .

$$P(N) \times P(\textbf{Dear} \mid N)$$

3. And the probability that the word **Friend** occurs in a normal message

$$P(N) \times P(\textbf{Dear} \mid N) \times P(\textbf{Friend} \mid N)$$

$P(N) \times P(\textbf{Dear} \mid N) \times P(\textbf{Friend} \mid N)$

0.67 x 0.47 x 0.29 = 0.09

$$P(N) \times P(\textbf{Dear} \mid N) \times P(\textbf{Friend} \mid N)$$

$$\mathbf{0.67 \times 0.47 \times 0.29 = 0.09}$$

4. Now we need to calculate the probability of the Evidence

$$P(\text{Evidence}) = P(\textbf{Dear}) \times P(\textbf{Friend})$$

$$P(\textbf{Evidence}) = P(\textbf{Dear}) \times P(\textbf{Friend})$$

$$\textbf{0.41} \times \textbf{0.25} = \textbf{0.1025}$$

$$P(N) \times P(\textbf{Dear} \mid N) \times P(\textbf{Friend} \mid N) \,/\, P(\textbf{Dear}) \times P(\textbf{Friend})$$

$$\textbf{0.09} \,/\, \textbf{0.1025} = \textbf{0.8780487}$$

Probability That the message is a normal message is → 0.87

→ Let do the same for the spam message

P (S) x P (**Dear** | S) x P (**Friend** | S)

0.33 x 0.29 x 0.14 = 0.01

→ Let do the same for the spam message

P (S) x P (**Dear** | S) x P (**Friend** | S)

0.33 x 0.29 x 0.14 = 0.01

P (**Evidence**) = P (**Dear**) x P (**Friend**)

0.41 x 0.25 = 0.1025

P (S) x P (**Dear** | S) x P (**Friend** | S) / P (**Dear**) x P (**Friend**)

0.01 / 0.1025 = 0.1025

$$P(S) \times P(\textbf{Dear} \mid S) \times P(\textbf{Friend} \mid S) \;/\; P(\textbf{Dear}) \times P(\textbf{Friend})$$

$$\mathbf{0.01 \;/\; 0.1025 = 0.1025}$$

Probability That the message is a spam message is → 0.1

P (N | **Dear Friend**) - 0.87

P (S | **Dear Friend**) - 0.1

0.87 > 0.1 So we will decide that it a normal Message

Let look at slightly more complicated example

Let try to classify this message : **Launch Money Money Money**

Since the message contain the word **Money 4 time**

It clear that the message is a spam (from the previous probability results)

→ Let confirm this by doing the math !

Calculating the score for a normal message

$$P(N) \times P(\textbf{Launch} \mid N) \times P(\textbf{Money} \mid N)^{\wedge} 4 = 0.000002$$

Calculating the score for a normal message

$$P(N) \times P(\textbf{Launch} \mid N) \times P(\textbf{Money} \mid N)^{\wedge}4 = 0.000002$$

$$P(\textbf{Evidence}) = P(\textbf{Launch}) \times P(\textbf{Money})$$

$$= 0.125 \times 0.208 = 0.026$$

$\rightarrow$ $0.000002 / 0.026 = 0.00007692307$

Calculating the score for a normal message

$$P(N) \times P(\textbf{Launch} \mid N) \times P(\textbf{Money} \mid N)^4 = 0.000002$$

However, when we do the same calculation for spam

We get 0 !

$$P(S) \times P(\textbf{Launch} \mid S) \times P(\textbf{Money} \mid S)^4 = 0$$

This is Because the probability we see **Launch** in **spam** is **0**, since it was not in **the training data**

P (S) x P (**Launch** | S) x P (**Money** | S) ^ 4 = 0

P (Dear | Spam) = 2 / 7 = 0.29

P (Friend | Spam) = 1 / 7 = 0.14

P (Lunch | Spam) = 0 / 7 = 0.00

P (Money | Spam) = 4 / 7 = 0.57

And when we plugin **0** for the P (**Launch** | **S**) ...

P (S) x P (**Launch** | S) x P (**Money** | S) ^ 4 = 0

P (Dear | Spam) = 2 / 7 = 0.29

P (Friend | Spam) = 1 / 7 = 0.14

P (Lunch | Spam) = 0 / 7 = 0.00

P (Money | Spam) = 4 / 7 = 0.57

… then it does not matter what value we picked from our initial guess that the message was Spam

$$P(S) \times P(\textbf{Launch} \mid S) \times P(\textbf{Money} \mid S)^{4} = 0$$

$P(\text{Dear} \mid \text{Spam}) = 2/7 = 0.29$

$P(\text{Friend} \mid \text{Spam}) = 1/7 = 0.14$

$P(\text{Lunch} \mid \text{Spam}) = 0/7 = 0.00$

$P(\text{Money} \mid \text{Spam}) = 4/7 = 0.57$

$\mathbf{P(S) = 0.33}$

… and it doesn't matter what the probability is it that we see **Money** given that the message **Spam** . . .

$$P(S) \times 0 \times P(\textbf{Money} \mid S)^{4} = 0$$

$P(\text{Dear} \mid \text{Spam}) = 2/7 = 0.29$

$P(\text{Friend} \mid \text{Spam}) = 1/7 = 0.14$

$P(\text{Lunch} \mid \text{Spam}) = 0/7 = 0.00$

$P(\text{Money} \mid \text{Spam}) = 4/7 = 0.57$

…Because anything times 0 → 0

$$P(S) \times 0 \times P(\textbf{Money} \mid S)^4 = 0$$

P (Dear | Spam) = 2 / 7 = 0.29

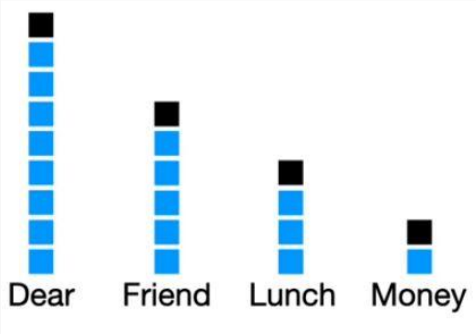
P (Friend | Spam) = 1 / 7 = 0.14

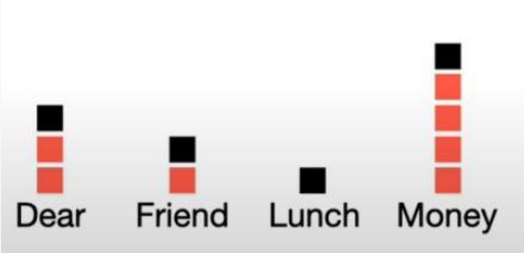
P (Lunch | Spam) = 0 / 7 = 0.00

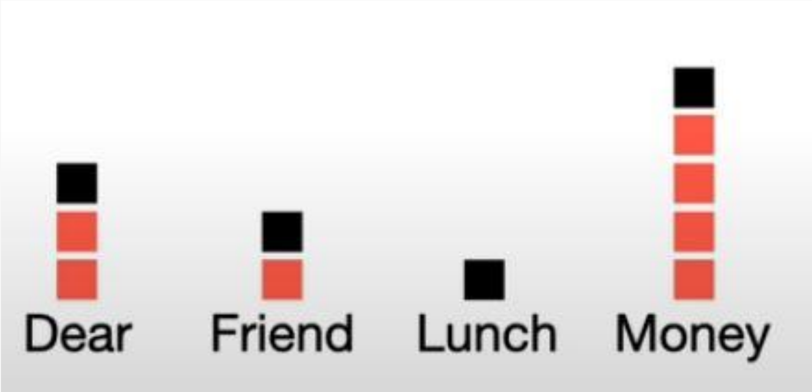
P (Money | Spam) = 4 / 7 = 0.57

To work Around this Problem, people usually add 1 count, represented by a black box, to each word in the histograms.

To work Around this Problem, people usually add 1 count, represented by a black box, to each word in the histograms.







The extra counts
that we added

P (**Lunch** | **Spam**) = 1 / 7+**4** **= 0.09**

Total of number of
words in spam

Notes

→ Adding counts (black boxes) to each word does not change our initial guess
That a message is normal, spam ....

$P(N) = 0.67$ $\qquad$ $P(S) = 0.33$

Notes

→ Adding counts (black boxes) to each word does not change our initial guess That a message is normal, spam ....

P (**N**) = 0.67 P (**S**) = 0.33

→ Because adding a count to each word did not change the number of messages in the training Dataset that are normal (8) ... Or the number of message that are spam

Lunch Money Money Money Money Money

Now when we calculate the
scores for this messages ...

$P(N) \times P(\text{Lunch} \mid N) \times P(\text{Money} \mid N)$ ^ 4 = 0.00001

$P(S) \times P(\text{Lunch} \mid S) \times P(\text{Money} \mid S)$ ^ 4 = 0.00122

Lunch Money Money Money Money Money

Now when we calculate the
scores for this messages ...

$P(N) \times P(\text{Lunch} \mid N) \times P(\text{Money} \mid N)^4 = 0.00001$

$P(S) \times P(\text{Lunch} \mid S) \times P(\text{Money} \mid S)^4 = 0.00122$

... We classify the message as **spam**

Naive Bayes Classifier Example

	woman	Man	Total
Teacher	8	12	20
Student	32	48	80
Total	40	60	100

With the tabulation below of 100 people, what is the conditional probability that a certain
certain member of the school is a "teacher" given that he is a "man"?

Decision Tree

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression.

The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

A tree can be seen as a piecewise constant approximation.

k nearest neighbor

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point

Stochastic methods for pattern recognition.

Hidden Markov model

Hidden Markov Models (HMMs) are the most popular recognition algorithm for pattern recognition. Hidden Markov Models are mathematical representations of the stochastic process, which produces a series of observations based on previously stored data.

Conclusion

pattern recognition is a versatile and powerful tool with diverse advantages across multiple industries and research fields.

Its ability to detect patterns and extract meaningful information from data fuels progress in artificial intelligence, making it an essential component in developing innovative solutions to complex real-world problems.

Thanks!

Thank you for your attention,
Do you have any questions ?