UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II

PH.D. THESIS

IN

INFORMATION AND COMMUNICATION TECHNOLOGY FOR HEALTH

DATA FOR HEALTH – MACHINE LEARNING ELABORATION DATA FOR URO-ONCOLOGY

SUPERVISED MACHINE LEARNING METHODOLOGIES FOR BLADDER CANCER PROGRESSION RISK CLASSIFICATION AND CLINICAL PATIENT MANAGEMENT

BY

LUCA SCAFURI

TUTORS: PROF. NICOLA PASQUINO

PROF. ALFREDO MARINELLI

COORDINATOR: PROF. DANIELE RICCIO
XXXVI CICLE

SCUOLA POLITECNICA E DELLE SCIENZE DI BASE - DIPARTIMENTO DI INGEGNERIA ELETTRICA E TECNOLOGIE DELL'INFORMAZIONE

Dedication:

Il faro

È liberta, è voglia di giungere

Che illumina ma non abbaglia

Che guida ma non influenza

Che ispira ma non costringe

Che ti accarezza ma non ti sfiora

Che ti accoglie ma non ti soffoca

Che ti indica ma non ti parla

Che dà ma non pretende

Che è presenza ma non è presente

Che è vita ma non è in vita

Il faro

E' libertà, è voglia di giungere

Con te, accanto a te

Mamma

DATA FOR HEALTH – MACHINE LEARNING

ELABORATION DATA FOR URO-ONCOLOGY

Supervised machine learning methodologies for bladder cancer progression risk classification and clinical patient management

Ph.D. Thesis presented

for the fulfillment of the Degree of Doctor of Philosophy

in Information and Communication Technology for Health (ICTH)

by

Dr. Luca Scafuri

October 2023

Approved as to style and content by

Prof. Sergio Rapuano, Advisor

Prof. Francesco Lamonaca, Co-advisor

Candidate's declaration

I hereby declare that this thesis submitted to obtain the academic degree of Philosophiæ Doctor (Ph.D.) in Information and Communication Technology for Health (ICTH) is my own unaided work, that I have not used other than the sources indicated, and that all direct and indirect sources are acknowledged as references. Parts of this dissertation have been published in international journals and/or conference articles (see list of the author's publications at the end of the thesis).

Napoli, July 16, 2023

Dr. Luca Scafuri

Abstract

In this thesis work, supervised learning methodologies were compared for the classification of the risk of progression of bladder cancer. In particular, three supervised learning algorithms were applied (Decision Tree, Random Forest and Naive Bayes) and the accuracy results were compared both considering the data set composed of all the variables and the data set containing a lower number of variables through the use of a dimensionality reduction technique (Feature Selection).

The data refers to a sample of 111 patients and both qualitative and quantitative variables were considered as input: sex, body mass index, smoking, family history, age, muscular invasiveness of the tumor, dimensions of the bladder wall, number and dimensions of lymph nodes, number and size of liver and bone lesions, etc. These variables were used to predict the risk of progression of bladder cancer, which is divided into four classes from 1 to 4: Low (risk 1), Medium-low (risk 2), Medium-high (risk 3), High (risk 4). The main goal is to give the medical oncologist objective support for the diagnosis of disease progression, a diagnosis which is not always easy to carry out, especially in the case of minimal and/or subtle disease progression, thus avoiding: exposing the patient to unnecessary side effects resulting from the use of a drug that is starting to lose its effectiveness, and to spend the budget relating to healthcare costs unnecessarily.

From the analysis of the results it can be seen that the predictions for the risk of progression of bladder cancer are satisfactory for the three algorithms used. Comparing the predictions obtained, the dimensionality reduction technique is reliable, the loss of accuracy is balanced by the advantage of having a lower number of predictors and this represents an advantage for the computational effort of the algorithm.

Keywords:

Bladder Cancer, Artificial Intelligence, Machine Learning, Classification, Progression Risk.

Sintesi in lingua italiana

In questo lavoro di tesi sono state confrontate metodologie di apprendimento supervisionato per la classificazione del rischio di progressione del tumore della vescica. In particolare sono stati applicati tre algoritmi di apprendimento supervisionato (Decision Tree, Random Forest e Naive Bayes) e sono stati confrontati i risultati delle accuracy sia considerando il data set composto da tutte le variabili, sia il data set contenente un numero di variabili inferiori attraverso l'utilizzo di una tecnica di riduzione della dimensionalità (Feature Selection).

I dati fanno riferimento ad un campione di 111 pazienti e sono state considerate come input sia variabili qualitative che quantitative: sesso, indice di massa corporea, fumo, familiarità, età, invasività muscolare del tumore, dimensioni della parete della vescica, numero e dimensioni di linfonodi, numero e dimensioni di lesioni epatiche e ossee, etc. Tali variabili sono state utilizzate per la previsione del rischio di progressione del tumore della vescica, che è diviso in quattro classi da 1 a 4: Basso (rischio 1), Medio-basso (rischio 2), Medio-alto (rischio 3), Alto (rischio 4). Il main goal è quello di dare all'oncologo medico un supporto oggettivo alla diagnosi di progressione di malattia, diagnosi che non è sempre facile da effettuare soprattutto in caso di minima e/o sfumata progressione di malattia, evitando così di: esporre il paziente ad inutili effetti collaterali derivanti dall'utilizzo di un farmaco che sta iniziando a perdere la sua efficacia, e di spendere inutilmente il budget relativo ai costi della sanità.

Dall'analisi dei risultati si può notare come le previsioni per il rischio di progressione del tumore della vescica siano soddisfacenti per i tre algoritmi utilizzati. Confrontando le previsioni ottenute, la tecnica della riduzione della dimensionalità risulta affidabile, la perdita di accuracy è bilanciata dal vantaggio di avere un numero inferiore dei predittori e questo rappresenta un vantaggio per lo sforzo computazionale dell'algoritmo.

Parole chiave:

Cancro della Vescica, Intelligenza Artificiale, Apprendimento Automatico, Classificazione, Rischio di Progressione.

Acknowledgements

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Contents

List of Acronyms:

The following acronyms are used throughout the thesis

Uro TC: Urological Computed Tomography

TAC: Computed axial tomography

CT: Computed tomography

TCC: transitional cell carcinoma

TNM: Tumor Node Metastasis

TUR: Transurethral resection

Re-TUR: Repetition Transurethral resection

MRI: Magnetic Resonance Imaging

G: Grading

WHO: World Health Organization

PUNLMP: Papillary urothelial neoplasm of low malignant potential

CIS: In situ carcinoma

BCG: Bacillus Calmette–Guérin

AI: Artificial intelligence

PCA: Principal Component Analysis

RMSE: Root Mean Square Error

Loocv: Leave-one-out-cross-validation

DB Scan: density-based spatial clustering of applications with noise

Eps: epsilon

MinPts: minimum points

CART: Classification And Regression Tree

Oob: out of bag

TP: True positive

TN: True negative

FP: False Positive

FN: False negative

List of Figures:

List of Tables:

List of Symbols

The following symbols are used within the thesis

h	set of nodes
S	subtrees
T	generic tree
l	left node
r	right node
P	probability of the event
u(t)	subspace associated with the final node
j	set possible classes
D	date set
N	number of observations
G	Gini index
$\sum$	summation
$p(j\vert t)$	relative frequency of class j at node t
t	child node
H(s)	entropy
F	frequency
R(t)	misclassified observations
M(t)	number of observations that do not belong to the class

N	total number of observations
r(t)	pr of misclassification
$\tilde{T}$	cardinality of the final node set
α	complexity parameter
x	training set
$fbag$	average of tree predictions
P(C|X)	posterior probability
P(X|C)	likelihood
$\prod$	production
$f(x\ \|\mu,\sigma2)$	gaussian probability
π	pi
μ	average
$\hat{\sigma}$	estimated standard deviation

INTRODUCTION

Within this thesis work, a classification of the risk of progression of bladder cancer was carried out. To do this, a statistical analysis was carried out and supervised machine learning methodologies were applied.

The bladder is a hollow, unequal and median muscular organ of the pelvis, its purpose is to collect the urine produced by the kidneys, which reaches it through the ureters and to allow it to be expelled outwards through the urethra.

Bladder cancer is a pathology that results quite frequently from the malignant transformation of the cells that cover its internal surface, the urothelium, i.e. the transitional epithelium that comes into contact with urine and lines the urinary tract from the renal calyces up to the urethra.

The data set used is made up of 111 patients and is the result of my daily clinical activity at the Complex Oncology Operating Unit of the "A. Tortora" of Pagani (SA).

18 input variables, both qualitative and quantitative, were considered for predicting the risk of bladder cancer progression.

The qualitative variables were transformed into dummy variables with the "one hot" coding so in total the number of predictors was equal to 22.

Given the large number of input variables, a dimensionality reduction technique, Feature Selection, intrinsic to the Random Forest algorithm, was applied. This technique was compared with the data set composed of all the variables in such a way as to highlight the advantage of providing a lower number of variables as input to the algorithm.

The selection of features is a technique that involves the reduction of the predictors, Random Forest was used as the intrinsic algorithm which selects the variables through the Gini index. For the prediction of the risk of progression, 5 variables were selected.

Three classification algorithms were subsequently applied: Decision Tree, Random Forest and Naive Bayes.

The predictions of the algorithms on the two sets of different variables were then compared: the one composed of all the variables and the one composed of the variables selected with Feature Selection.

The classification metric considered is Accuracy which is given by the ratio between the number of exact predictions and the total number of predictions made. It gives an idea of how the algorithm worked and is therefore used to evaluate the performance of the model.

The accuracies of the three algorithms applied to the two sets of different variables for predicting the risk of bladder cancer progression were compared in such a way as to identify the algorithm that provided better performance.

The objective of this thesis work is to apply machine learning algorithms in the medical field, in such a way as to give objective support to the diagnosis.

The analysis was carried out with the help of the R software and its R studio interface.

CHAPTER 1

BLADDER CANCER

1.1 BLADDER CANCER

The bladder is a hollow, unequal and median muscular organ of the pelvis, it is responsible for collecting urine which, produced by the kidneys, reaches it through the ureters. From the bladder, urine is periodically expelled outward through the urethra. The urethra passes through a urogenital diaphragm made up of striated muscles subjected to voluntary control, or external sphincter. The process of emitting urine is called urination and consists of the periodic emptying of the bladder by means of an automatic reflex of the spinal cord which stimulates the contraction of the detrusor muscle, a smooth muscle band that forms a layer of the bladder.

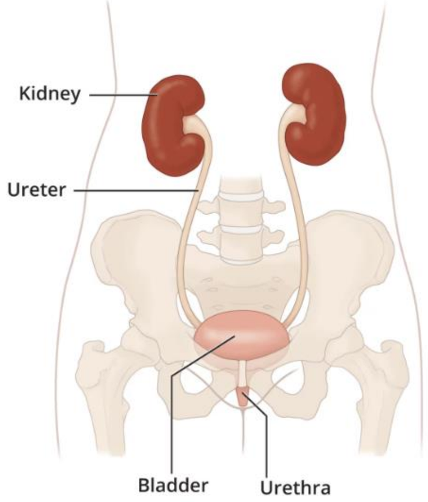


Figure 1: Urinary system

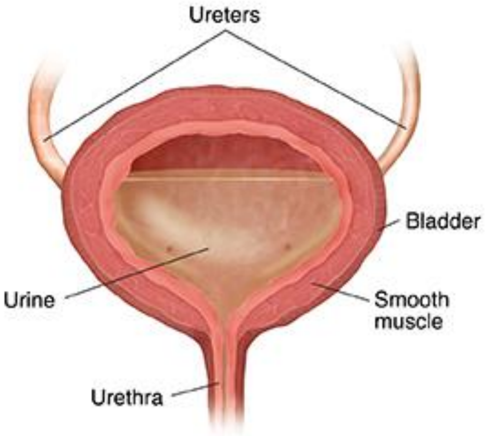


Figure 2: Bladder

The bladder is made up of several layers of tissue:

- *Internal layer:* it is composed of the mucosa with a lining tissue (urothelium) to protect the bladder from urine. When the bladder is empty, the urothelium is composed of numerous folds, which relax as the bladder fills with urine.

- *Layer of connective tissue*: it is called the lamina propria, it is located below the urothelium and is rich in nerve endings, blood vessels and lymphatic vessels, which bring sensitivity, nourishment and oxygen to the entire bladder.

- *Muscle tissue*: it is made up of three superimposed layers which are not clearly distinct from each other and do not have a uniform thickness. Controls the dilation and contraction of the bladder during urination.

- *Adipose tissue*: it is the outermost layer, which covers the muscle layer.

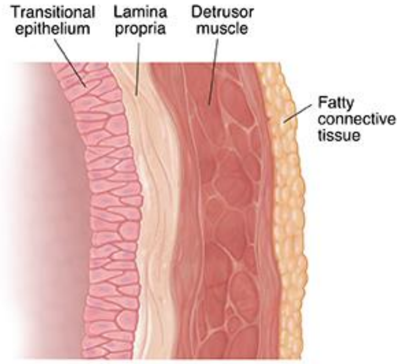


Figure 3: Layers of the bladder wall

Bladder cancer is due to the malignant transformation of the cells that line the internal surface of the bladder itself. It is a pathology that derives quite frequently from the epithelial cells that make up the internal lining of the bladder, i.e. the urothelium, i.e. the transitional epithelium that comes into contact with urine and lines the urinary tract from the renal calyces to the urethra . The bladder is the most frequent site of neoplasms originating from the transitional epithelium (80% of cases), while rarely bladder cancer can be an adenocarcinoma or a squamous cell carcinoma.

In fact, there are different types of bladder tumors:

• transitional cell carcinoma, the most common, which arises in the cells that make up the internal lining of the organ;

- primary squamous carcinoma, rarer, which affects squamous cells and seems particularly linked to parasitic infections;
- adenocarcinoma, very rare, which begins in the cells of the glands present in the bladder.

In fact, bladder lesions are almost always pathologies of the urothelium and can be superficial or infiltrative. The prognosis of bladder cancer is given by the combination of histological differentiation and degree of infiltration, associated with the lymph node status and the presence of distant metastases. The genesis of bladder cancer is due, as in other tumors, to genetic mutations. The p53 protein acts as a cell cycle controller, protecting the cell from excess mitosis and blocking cells that reproduce abnormally by inducing apoptosis (cell death). The mutation of the gene that translates the p53 protein (tumor suppressor gene), present on chromosome 17, has been frequently found in oncology, and is held responsible for blocking the mechanisms of induction of cellular apoptosis, resulting in uncontrolled growth and division of the cells that thus give rise to cancer.

1.1.1 Epidemiology

In 2018, 27,110 new cases of bladder cancer were recorded in Italy, in particular 21,500 among men and 5,600 among women, equal to 7% of all incident tumors. In urology it is second only to prostate cancer [1].
At the time of diagnosis, 70% of tumors are superficial, while the remaining 30% have muscular infiltration. Of the patients treated with radical cystectomy, approximately 57% have a disease infiltrating the muscles already at the time of diagnosis, while the remaining 43% develop this condition at a later time, despite the treatments carried out [2].

Age: Bladder cancer is most common between the ages of 60 and 70 and is three times more common in men than in women. It is the fourth most common cancer in men with percentages of 11% in the 50-69 age group and 12% after the age of 70. In women, however, it is less frequent and is responsible for 1% of female tumors in the same age group [1].

Geographical area: men have higher incidence values in the center and south compared to the northern regions; in women, however, the values are lower [1].

Mortality: in 2015 (Istat) 5,641 deaths from bladder cancer were recorded in Italy, equal to 3% of cancer deaths [1].

Survival: 5-year survival is 79%, with no significant differences between men and women. It decreases with age, is equal to 96% in young people (< 45 years) and reduces up to 66% in those aged 75+ [3].

Prevalence: in 2018 in Italy, over 269,000 people were estimated to be alive with a previous diagnosis of bladder cancer. Rates per 100,000 inhabitants are on average higher in the north than in the south.

1.1.2 Risk factors

There are some factors that can increase the risk of bladder cancer, for example:

- smoking, due to the chemicals that accumulate in smokers' urine [4].
- exposure to chemical substances: arsenic and products used in the processing of rubber, leather, paint and in the textile industry.

- drugs used in the treatment of cancer: cyclophosphamide.

- exposure to radiation following radiotherapy treatment in the pelvic region.

- chronic inflammation of the bladder: urinary infections or cystitis, caused for example by parasites widespread in some Middle Eastern countries (Schistosomiasis).

- family history: presence of cases of bladder cancer in the family.

- the Caucasian population is more affected than the others.

- dietary factors: the incidence of bladder cancer is lower in subjects who consume abundant quantities of fruit and vegetables.

1.2 DIAGNOSTIC FRAMEWORK

The diagnostic flow is shown below, i.e. the procedure that is initiated when a bladder neoplasm is suspected. All this will be covered in detail in the following paragraphs.

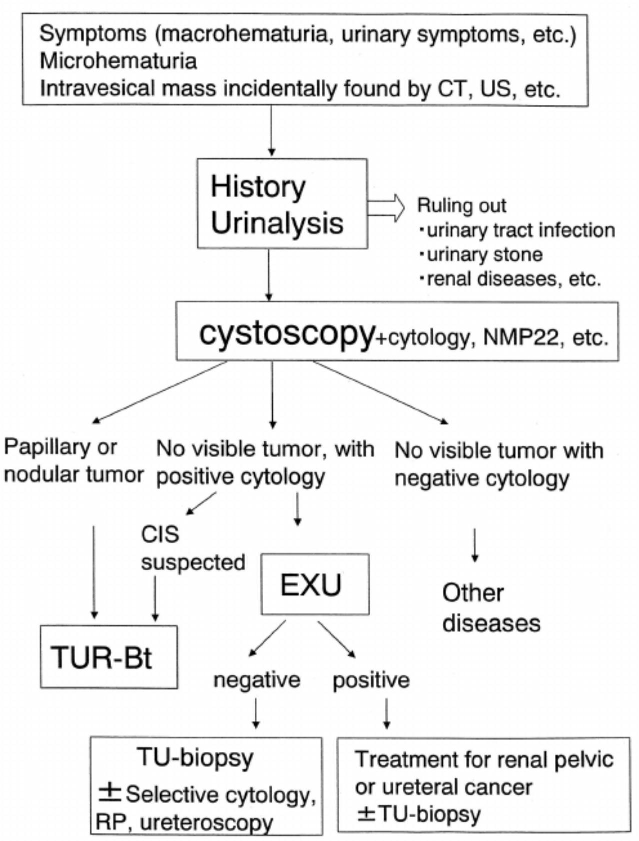

Symptoms (macrohematuria, urinary symptoms, etc.)
Microhematuria
Intravesical mass incidentally found by CT, US, etc.
History
Urinalysis
Ruling out
·urinary tract infection
·urinary stone
·renal diseases, etc.
cystoscopy+cytology, NMP22, etc.
Papillary or nodular tumor
No visible tumor, with positive cytology
No visible tumor with negative cytology
CIS suspected
EXU
Other diseases
TUR-Bt
negative
positive
TU-biopsy
±Selective cytology, RP, ureteroscopy
Treatment for renal pelvic or ureteral cancer
±TU-biopsy

Figure 4: Diagnostic flow

1.3 SYMPTOMS

The symptoms of bladder cancer are very similar to those of other diseases that affect the urinary system. They range from the presence of blood in the urine to the burning sensation in the bladder when pressing on the abdomen, from difficulty urinating to the ease with which infections are contracted. The main and often only initial sign of the disease is the presence of gross hematuria.

Superficial forms of tumor rarely manifest themselves with only irritative disorders such as urinary urgency, frequency, stranguria, which are instead frequent in patients with carcinoma in situ. The size or clinical stage of the neoplasm does not correlate with the extent and characteristics of the hematuria.

The later the diagnosis and the more advanced the tumor, the lower back pain may be present [5].

1.4 DIAGNOSIS

1.4.1 Urinary cytology

The diagnosis of urothelial neoplasia is carried out through cytological examination of the urinary sediment. The sediment allows us to examine the desquamation of the urothelium and detects the possible presence of neoplastic cells. It is a low-cost, non-invasive test characterized by high sensitivity in high-grade tumors, low sensitivity in low-grade tumors and high specificity [6].

The correct preparation and preparation of the urinary sediment, the number of exfoliated cells, the experience of the examiner and the possible presence of infections, stones or previous instillations in the bladder tract influence the outcome of the cytological examination.

The adequate and correct conservation of the urinary sample are essential factors for a correct cytological diagnosis, 30mL represents the optimal volume for a reliable outcome. When collecting urine, the use of an alcoholic fixative is also recommended to avoid cell degradation and therefore the validity of the test itself [7].

1.5 IMAGING DIAGNOSTICS

1.5.1 Ultrasound

When there is a suspicion of urothelial pathology, the first-instance investigation is ultrasound which has an overall accuracy of between 80-95% [8] and a very high specificity.

Ultrasound is a non-invasive diagnostic method which, using ultrasound (sound waves) emitted by particular probes placed on the patient's skin, allows you to visualize organs, glands, blood vessels, subcutaneous structures and also muscle and tendon structures in numerous parts of the body.

During the ultrasound, the area to be examined is moistened with a special non-toxic gel, which allows for better transmission of ultrasound through the human body. Specifically, it allows you to obtain a real-time image of internal organs, blood vessels, structures and substructures of the body, in search of diagnostic data.

The ultrasound suspicion is normally investigated with an endoscopic examination.

1.5.2 CT

Uro-CT (Urological Computed Tomography) is carried out when clinical suspicion persists or in the case of positive urinary cytology after a negative endoscopic examination.

CT is an imaging diagnostic technique that allows you to examine every part of the body. It is a radiological examination in which data is collected by the passage of various X-ray beams in the affected area and are reprocessed by a computer in order to reconstruct a three-dimensional image of the different types of tissues. The acronym TAC, for computerized "axial" tomography, still exists but is no longer in use, as long ago the examination was conducted along a single axis, with sections perpendicular to the length of the body. Today there are more modern multilayer machines and computed tomography is no longer just axial, but the images are acquired with a spiral technique which allows three-dimensional images to be obtained. The term TAC is therefore now obsolete.

It is done like this:

The x-ray tube, which emits the Sometimes, to obtain better images of the vasculature (arterial and venous) of organs and tissues, an iodine-based contrast medium is used, which is commonly injected intravenously. The injection can cause a rather intense sensation of heat but which fades quickly [9].

CT can only document the macroscopic involvement of the perivesical fat and nearby organs, while it is not able to evaluate the extension of microscopic initial stages of growt.

It is used more than urography (radiological examination), both for the diagnosis of any localization of disease affecting the upper excretory system, and in the definition of local infiltration of the urinary bladder.

It is used in staging (search for possible lymph node and/or organ metastases) before radical cystectomy surgery.

In fact, it is the only test that allows lymph node staging of the disease (category N of the TNM).

In the evaluation of wall infiltration (category T) the pre-eminent role is played by the staging TUR. If infiltration of the muscle layer is present, CT may confirm increased wall thickness or show increased density of perivesical fat which raises the suspicion of extension outside the bladder wall (T3a). Macroscopic invasion of the perivesical fat (T3b) is generally demonstrated by examination with good diagnostic accuracy.

However, CT per se cannot discriminate with sufficient reliability the degree of muscle infiltration, or distinguish between the T2a and T2b categories. If there is clinical suspicion of an infiltrating bladder lesion already at the time of diagnosis, it is advisable to perform a CT examination before endoscopic resection. CT allows you to visualize the lymph nodes that show an increase in volume and define their morphology, any skeletal or soft tissue metastases of the pelvis as well as, when extended to the abdomen, secondary abdominal localizations. It also provides a comprehensive visualization of the kidney and upper excretory tracts.

CT is therefore used to detect the tumor lesion in the bladder, measure its size, determine to what extent the tumor has spread to surrounding or distant tissues and, above all, to monitor the effects of radio or chemotherapy treatment over time.

1.5.3 MRI

Magnetic Resonance Imaging (MRI) provides diagnostic information that is comparable to CT. It is a type of scan that is used to look at the bones, tissues and organs inside the body. It uses strong magnetic fields and radio waves to create extremely detailed images.

The advantages of MRI compared to CT consist in the possibility of defining and distinguishing the different tissues and organs which in CT have the same density and which are therefore indistinguishable. In clinical practice, however, CT is preferred due to its speed of use and ease of access.

As regards lymph node involvement, both techniques are able to evaluate the volumetric and morphological trend.

1.6 ENDOSCOPIC DIAGNOSTICS

Urological endoscopy is a procedure that allows you to directly view the urinary tract or renal calico-pielic cavities, ureters, bladder and urethra, detecting any pathologies. Various rigid and flexible fiber optic systems are used. The instrument used is called an endoscope.

In particular, cystoscopies are endoscopic examinations performed on an outpatient basis that allow direct visualization of the bladder and urethra. The examination is performed through the use of fiber optic instruments to the end of which a small camera is connected which, during the examination, are introduced from the urethra to the bladder for the identification of lesions-pathologies affecting the bladder or lower urinary tract.

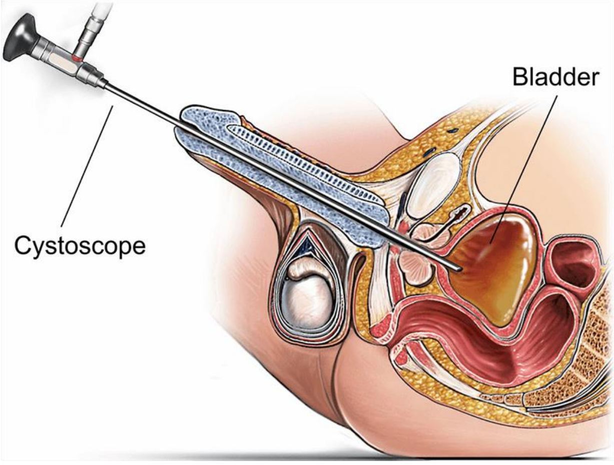


Figure 5: Cystoscopy

The report must report in detail the position, number, size and appearance of the bladder neoplasm(s) that were found to be abnormal with respect to the bladder mucosa.

Cystoscopy also allows you to take a small sample of tumor tissue, which will be used to carry out histological examination in the laboratory. The histological examination will confirm the diagnosis of bladder cancer and provide information on the specific characteristics of the tumor.

1.7 STAGING AND GRADING

Bladder tumors are classified by stage, which is a measure of the size and spread of the tumor, and by grade, which is a measure of the differences between cancer cells and healthy cells. Taking into account both stage and grade, tumors can be classified in this way:

- Non-muscle invasive bladder cancer, when the tumor is confined to the transitional epithelium (stage Ta and Tis) or to the submucosa (stage T1);

- Muscle invasive bladder cancer when it has invaded the muscle layer of the bladder or has spread to surrounding tissues (stage T2-T3);

- Advanced or metastatic bladder cancer when the tumor has invaded the pelvis, abdominal wall or other organs. Bladder cancer tends to metastasize to the lymph nodes, lungs, liver and bones (stage T4).

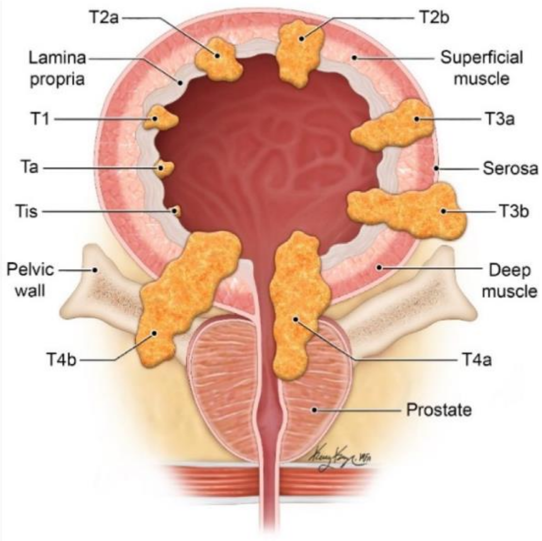


Figure 6: Stadium

Staging is a fundamental moment for describing how large a tumor is and how much it has spread compared to the original site of development.

Cancer cells behave very differently from healthy cells as they grow and multiply in a disorderly manner, and do not die as and when they should. In this way, a tumor mass is formed which, unlike a healthy tissue, tends to grow in volume, and whose cells can detach and migrate, through the lymphatic system and/or the blood flow, to other parts of the body, giving rise to metastases. .

Staging defines in which phase of this process the tumor is found, and is therefore a fundamental aspect of the diagnosis, since the prognosis of the disease and the most appropriate type of treatment to adopt can depend on these characteristics [9].

If the tumor is, for example, localized in a single location and is small, local treatment such as surgery or radiotherapy can be curative. In cases where, however, it is extended to other sites, a local intervention is normally not enough: it may be necessary to resort to systemic treatments, i.e. those capable of having effects on the whole body, such as chemotherapy or other more innovative pharmacological treatments (for example molecularly targeted therapies or immunotherapies).

Staging systems detect:

- The size of the primary tumor;
- Involvement of the lymph nodes;
- The presence (and number) of metastases, i.e. tumor cells that have migrated through the blood from the primary site to other organs.

From the combination of these elements a very detailed description of the tumor and its extension can be obtained.

The most common staging system is the so-called "TNM" system, an English acronym that stands for “Tumour, Node, Metastasis” [10].

A number is associated with each of the letters that make up the acronym:

- **T**, refers to the size of the primary tumor: the scale goes from 1, which identifies the smallest tumors, to 4 for the largest ones.

- **N**, indicates whether the cancer has spread to the lymph nodes, it can have a value ranging from 0 (no lymph nodes involved) to 3 (many lymph nodes involved).

- **M**, stands for metastasis, can have a value of 0 (if the tumor has remained limited to its primary site) or 1 (when the tumor has spread to other parts of the body).

In particular, metastasis is the phenomenon through which tumor cells move from the area in which they were formed to another part of the body. Metastatic cells move away from the primary tumor, travel in the blood or lymph vessels, and form a new secondary tumor in other organs or tissues.

The characteristic that distinguishes a benign tumor from a malignant one is represented by its ability to form metastases. The cells with metastatic capacity that multiply in the organ of origin manage, over time, to break the barriers of the tissue, until they reach the nearest lymph nodes, which are real "control stations" that have the task of blocking the passage of foreign or dangerous molecules. If the metastatic cells pass the lymph node filter, they enter the lymphatic circulation and can even reach areas very far from their tumor of origin.

From the lymphatic circulation these cells can also pass into the blood circulation, thanks to the numerous communication routes between the two systems. Sometimes cancer cells can enter directly into blood vessels by crossing their walls. If metastatic cells survive the attack of the immune system, they can reach a new location where they can reproduce and give rise to a new tumor.

There are many organs that can become the site of metastases: the most common sites are the liver and lung, as they are very vascularised, i.e. they have a large number of incoming and outgoing blood vessels and are therefore more likely to be crossed by tumor cells circulating; the liver also performs a "filter" function of the blood which can favor metastases. Other common sites of metastasis are bones and the brain.

TNM staging is illustrated in detail below:

Table 1: Primary tumor

T – Primary tumor	
Tx	Not enough material
T0	No evidence of disease
Tis	Carcinoma in situ: flat tumor
Ta	Does not infiltrate the submucosal layer
T1	Infiltrating the submucosal layer
T2a	Infiltrating the first half of the muscularis layer
T2b	Infiltrating the second half of the muscularis layer
T3a	Microscopic infiltration of peri-vesical fat
T3b	Macroscopic infiltration of peri-vesical fat
T4a	Infiltration of nearby organs: prostate, seminal vesicles, uterus, vagina
T4b	Infiltration of the pelvic and/or abdominal wall

Table 2: Lymph nodes

N – Lymph nodes	
Nx	Lymph nodes not assessable
N0	Absence of metastases in regional lymph nodes
N1	Metastasis in a single lymph node of an endopelvic station (obturators, internal iliacs, external iliacs, presacrals)
N2	Metastasis to two or more endopelvic lymph nodes
N3	Metastases to lymph nodes located proximal to the common iliac artery

Tabella 1: Metastasis

M – Distant metastasis	
Mx	Distant metastases not assessable
M0	Absence of distant metastases
M1	Presence of distant metastases

The TNM classification can be grouped into stages based on the extent of the tumor within the body. There are four stages:

- Stage I: T1-2 N0 M0

- Stage II: T1-2 N1 M0, T3 N0 M0

- Stage III: T1-2 N2-3 M0, T3 N1-3 M0, T4 N0-3 M0

- Stage IV: T1-4 N0-3 M1

Examples of bladder tumors are shown below:

D.G.R., 74 year old male, cancerous lesion of the bladder wall, stage I, protruding into the lumen.

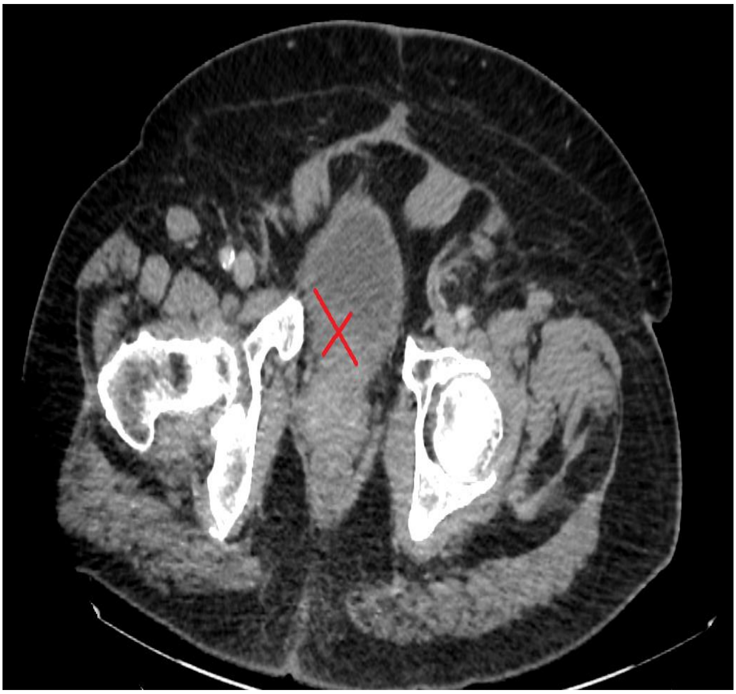


Figure 7: Bladder cancer, stage I

A.N., 77 year old male, cancerous lesion of the bladder wall, stage II

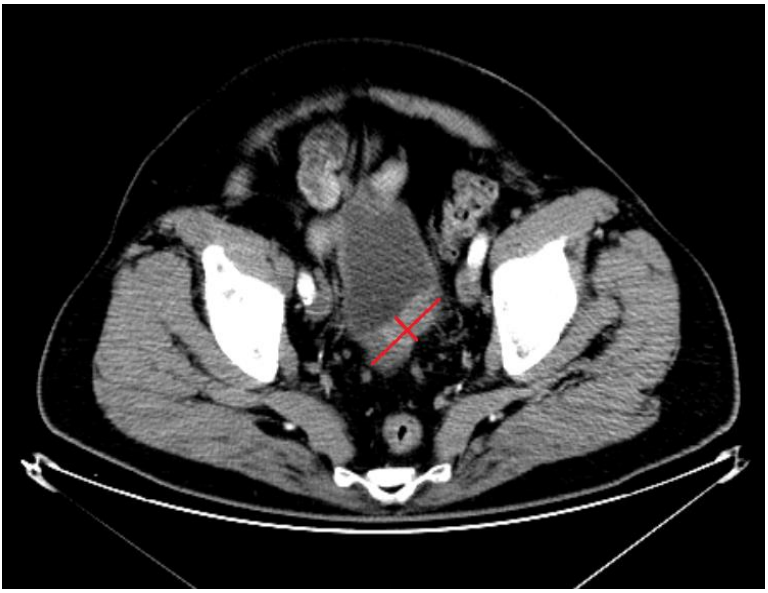


Figure 8: Bladder cancer, stage II

In addition to the TNM classification, bladder carcinomas are classified according to different grades using the 1973 WHO (World Health Organization) criteria, updated in 2004 and subsequently in 2016. The histological grade corresponds to cellular differentiation. It is useful for classifying the tumor and identifying the most appropriate treatment. It represents the "aggressive potential" of the tumor.

The grade of the tumor is identified based on the appearance of the cells, therefore whether they are more or less different compared to the original appearance. According to the WHO 1973 classification, different grades can be distinguished:

- G1, slightly differentiated cells
- G2, moderately differentiated cells
- G3, poorly or non-differentiated cells

The most recent classification introduces papillary urothelial neoplasm with low degree of malignant potential (PUNLMP), which histologically is characterized by the absence of cytological aspects of malignancy showing cells of normal appearance but with a papillary configuration. The WHO 2016 classification [11], with regards to grading, reproduces the WHO 2004 classification [12], eliminates the intermediate grade (G2) and distinguishes exclusively between low-grade and high-grade neoplasms.

Table 4: Histological grade

WHO/AFIP 1973	WHO 2004 E WHO 2016
Papilloma	Papilloma

TCC grade 1	PUNLMP
TCC grade 1	Low-grade urothelial carcinoma
TCC grade 2	Low- or high-grade urothelial carcinoma
TCC grade 3	High-grade urothelial carcinoma

There is therefore a difference between stage and grade: the first indicates how large a tumor is and how much it has spread in the body, the second describes how strong the abnormal characteristics of the tumor cells are. The higher the grade, the more the tumor cells are different from healthy ones and are destined to grow and spread quickly in the body.

To find out the grade of the tumor, a part of it is taken during a biopsy and is subsequently observed under a microscope.

Low-grade tumors have cells that are very similar to healthy ones and tend to grow slowly. In high-grade tumors the cells differ greatly in morphological characteristics from those of normal tissues and tend to grow and spread rapidly.

However, there is a strong correlation between the stage and grade of the tumor. Almost all superficial tumors are low-grade while most muscle-invasive tumors are high-grade. From a clinical point of view, it seems appropriate to keep both classifications, TNM and grade, in the report in order to allow better prognostic stratification.

1.8 INFILTRATING AND NON-INFILTRATING TUMORS

Urothelial cell carcinoma is the most common bladder neoplasm [11] and can be divided into:

- Non-infiltrating neoplasm
- Infiltrating neoplasm

Non-infiltrating tumors are those of stage Ta, which have a papillary architecture and are divided as follows:

- Papillary urothelial neoplasm of low malignant potential
- Low-grade papillary urothelial carcinoma
- High grade papillary urothelial carcinoma

Infiltrating urothelial neoplasms are those that have a T stage > or equal to T1. The T1 stage invades the mucosa while the T2 stage invades the muscularis layer. This type of neoplasm is high grade, but it is still advisable to report the grading in the report as there may be exceptionally different cases.

CIS (carcinoma in situ), on the other hand, is a rare neoplasm to be considered separately. It is devoid of papillary architecture and by definition of high grade. It is a neoplasm with poorly differentiated transition cells and limited to the urothelium. This type of neoplasm is observed in association, simultaneously with or following the development of papillary and/or invasive urothelial neoplasms [11], to which the presence of CIS confers greater capacity for progression. It is a flat lesion that does not have papillary structures and is composed of malignant and often non-cohesive cells.

1.9 TREATMENT

The therapeutic approach for bladder cancer differs depending on the muscular invasiveness. In particular, there are three categories:

- Non-muscle invasive
- Muscle-invasive
- Advanced

After the diagnosis of a bladder tumor, which occurs through cystoscopy, the next step is represented by endoscopic TUR surgery.

The TUR has a dual purpose:

- Removes the tumor completely, if possible
- Define the stage and identify the degree of cellular differentiation

In the presence of a superficial bladder tumor, TUR has a therapeutic role, while in the case of an infiltrating neoplasm it has no curative value, but allows precise staging of the tumor. The histological examination of the pieces taken with the TUR allows us to distinguish tumors of low grade or high degree of malignancy, superficial or infiltrating the muscles and is an examination at the basis of the choice of subsequent treatment [1].

1.9.1 Treatment of non-muscle invasive disease

Endoscopic trans-urethral resection (Tur)

The first level therapeutic approach for all non-muscle invasive bladder tumors is endoscopic resection. This surgery involves the complete removal of the tumor, including the implant base and the margins surrounding the exophytic portion. It is advisable to carry out the operation with local-regional anesthesia.

In particular, the following stand out:

- Tumors > 2cm in size: resect and collect the exophytic portion and implant base with related perilesional margins separately, avoiding damage from electrocautery

- Tumors < 2cm in size: they can be removed “en bloc” by including the exophytic part and muscular layer corresponding to the implant base in a single sample. It allows the en bloc removal of the entire neoplasm with a particular technique [13].

Endoscopic surgery can be followed by:

- intravesical instillations of chemotherapy drugs, such as mitomycin C, or immunotherapeutics, such as bacillus Calmette-Guérin (BCG), for non-muscle-invasive forms. In fact, TUR is usually followed by immediate bladder instillation with BCG to reduce the risk of recurrence. BCG acts in such a way as to provoke an immune reaction in such a way as to prevent the onset of relapses which occur with a frequency rate of 30-60% and which require monitoring of the patient for at least 5 years.

- cystectomy and/or radiotherapy, often in combination with chemotherapy for muscle-invasive forms.

Re-Tur

RE-TUR is the second endoscopic transurethral resection. In 33-53% of stage Ta, T1 tumors a residual neoplasm was found. A second resection is therefore recommended (usually between 2-6 weeks after the first operation) in case of incomplete resection, absence of tunica muscularis, high-grade Ta or T1 neoplasm at the first resection [14].

1.9.2 Treatment of muscle-invasive disease: Radical cystectomy

If following TUR the tumor is infiltrating, another type of operation is necessary.

The standard treatment needed for muscle-invasive tumors (T2-T3) is radical cystectomy. The bladder, prostate and seminal vesicles will be removed in men and the bladder, uterus and appendages in women. In both sexes, the lymph nodes are also removed up to the bifurcation of the aorta.

1.9.3 Treatment of metastatic disease

However, chemotherapy treatment is provided for all those patients who develop metastatic cancer at the time of diagnosis or for all those who undergo radical cystectomy for muscle-invasive bladder cancer and develop a local or distant recurrence of the disease [14].

Urothelial carcinoma of the bladder is a chemosensitive disease, therefore chemotherapy is the treatment of choice in patients with advanced or metastatic disease.

The standard treatment is chemotherapy regimens containing cisplatin.

Radiotherapy can also be carried out in association with chemotherapy or in place of surgery in patients suffering from bladder cancer and inoperable due to comorbidities.

1.10 RISK OF RECURRENCE OR PROGRESSION

A peculiar characteristic of this type of tumor is the tendency to recur, that is, to reappear some time after complete removal, even in completely different bladder areas. This happens because the urothelium of patients affected by this disease widely presents alterations that predispose the formation of the tumor.

Instead, we speak of tumor progression when a superficial carcinoma of the bladder changes biological behavior (possibly even after an initial complete removal) and begins to infiltrate the muscular wall of the bladder.

It is therefore very important to carry out adequate monitoring of patients over time even after tumor removal due to these two characteristics.

All the parameters useful for quantifying these risks can be obtained from the patient's clinical history, from the macroscopic description of the tumor before removal and

from the microscopic characteristics indicated in the histological report performed after removal.
The most important factors to consider are [15]:

- The rate of tumor recurrence: a neoplasm that has occurred several times in the past is potentially more dangerous than a first-episode disease.

- The number of tumors: in some cases the bladder neoplasm can be single, other times it can be multiple. The presence of more than eight tumors is correlated with a higher risk of recurrence and progression.

- The diameter of the tumor: the limit that differentiates the most at-risk forms from the less problematic ones is around 3 centimetres.

- The stage of infiltration of the bladder wall: the papillary forms limited to the urothelium alone (defined as Ta in the TNM classification) are distinguished from those which also reach the first subepithelial tissues (T1).

- The degree of cellular differentiation: describes how much the tumor cells, in their histological cellular appearance, differ from the normal bladder urothelium; provides for a subdivision into 3 groups: G1, G2 and G3 (where G3 represents the most undifferentiated and dangerous form).

The possible associated presence of carcinoma in situ (CIS): this is a particular form of flat bladder tumor which, although still limited to the urothelium alone, is made up of particularly malignant cells and with a high tendency to progress towards infiltrating tumor forms.

CHAPTER 2

MACHINE LEARNING

2.1 MACHINE LEARNING

Machine Learning, translated into Italian as automatic learning, is a subset of artificial intelligence (AI) and represents the ability of machines to learn without having been explicitly and previously programmed.

It is a data analysis method that automates the building of analytical models and is based on the idea that systems can learn from data, identify patterns autonomously and make decisions with minimal human intervention.

Arthur Lee Samuel, a pioneering American scientist in the field of Artificial Intelligence, first coined the term in 1959 although, to date, the most accredited definition by the scientific community is that provided by another American, Tom Michael Mitchell, director of the Machine department Learning from Carnegie Mellon University:

«a program is said to learn from experience E with reference to some class of tasks T and with performance measurement

P, if its performance in task T, as measured by P, improves with experience E» [16]

According to Mitchell, learning occurs when program performance improves after carrying out a task or completing an action. Machine Learning therefore allows you to learn when there is experience. The main goal is for a machine to be able to carry out inductive reasoning and therefore generalize from its own experience.

From an IT point of view, the program is provided "only" with data sets which are processed through algorithms developing its own logic to carry out the function, the activity, the requested task, therefore you will not write a program to order the machine what to do, but it will develop the rules itself. The computational analysis of machine learning algorithms and their performance is a branch of theoretical computer science called learning theory [17].

A generic function y=f(x) solves a problem and is the programmer detailing how f(x) works. With machine learning, generic mathematical and statistical algorithms are used which, exposed to a specific series of data in an initial phase defined as "training" and passing through a second phase of evaluation of the results with optimization of the parameters, autonomously derive the function able to identify the most probable value of y in a different series of data, possibly indicating a degree of confidence in the estimate. In practice, it derives the function f(x) on its own.

The system composed of trained algorithm, data and operational parameters is called a model.

In summary, therefore, a Machine Learning model constantly learns from experience rather than performing a task following explicit and defined rules. A system based on predetermined rules will perform a task the same way every time, whether for better or worse, while the performance of a machine learning system can be improved through learning by providing the algorithm with more data.

The peculiarity of machine learning models is therefore that of being very effective in identifying common characteristics or trends in huge data sets, taking into consideration a number of variables that no human being can be able to evaluate, or even notice.

Depending on the nature of the “signal” used for learning or the “feedback” available to the learning system, machine learning tasks are classified into three broad categories:

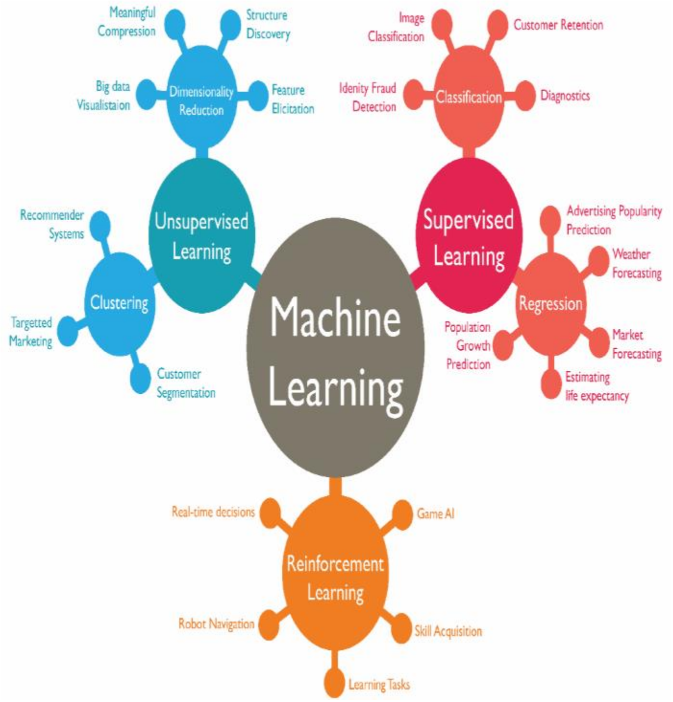


Figure 9: Types of learning

- ***Supervised learning:*** In this category, data sets are provided to the computer, as input and information related to the desired results. The goal is for the system to identify a general rule that connects input data with output data in such a way as to reuse this rule for other similar tasks. For example, it can be applied in the medical field to predict the response to certain therapies.

- ***Unsupervised learning:*** in this second category the system is only provided with data sets without any indication of the desired result. The purpose of this second learning method is to "trace" hidden patterns and models, that is, to identify a logical structure in the inputs without them being previously labeled. For example it can be applied for grouping animal species.

- ***Reinforcement learning:*** the system input is a goal to be achieved. The system initially knows the objective but is unable to achieve it, because it does not have a dataset of examples for training, nor a prior knowledge base. The surrounding environment is observed, which is subsequently transformed into a feature vector X. Each combination of elements of the vector is a different state of the environment. The algorithm learns and adapts to environmental changes through an evaluation system, which establishes a reward if the action performed is correct, or a penalty in the opposite case. The goal is to maximize the reward received, without announcing the path to take.

> The reinforcement function measures the degree of success of an action or decision, compared to a predetermined objective, which can be a reward or a penalty. For example, it can be applied in the case of a robot that must learn to move within a path.

Then there is **semi-supervised learning**, which represents a middle ground between unsupervised and supervised learning. It consists of problems that have the majority of input data, but only a portion of them are labeled. It is in fact considered as a "hybrid" model since the computer is provided with an incomplete data set for learning; some of these inputs are characterized by their respective outputs (as in supervised learning), while others are devoid of them (as in unsupervised learning). The objective is the same: to identify rules and functions for solving problems, as well as models and data structures useful for achieving certain objectives. Each of these models solves different types of problems: supervised machine learning problems are divided into **classification** and **regression** problems.

Classification: is a supervised learning problem that requires making a choice between two or more classes to attribute to the data. The algorithm is trained in such a way as to recognize the categories it belongs to through a training dataset. In each example, the machine is provided with both the variables that describe the environment, i.e. the inputs, and a label to indicate the result, i.e. the output. The system processes the examples with the aim of looking for a general rule which is called a model.

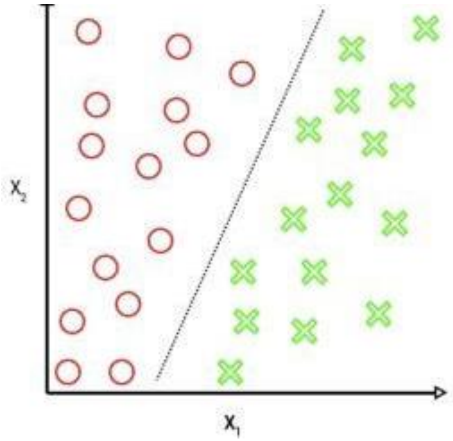


Figure 10: Classification

Regression: is a supervised learning problem that requires the model to predict a numerical value. The difference with classification is that the output to be predicted has a continuous domain. With this type of model, the relationship between the dependent variables and the independent variables is established through a line that more or less represents the relationship between the two variables.

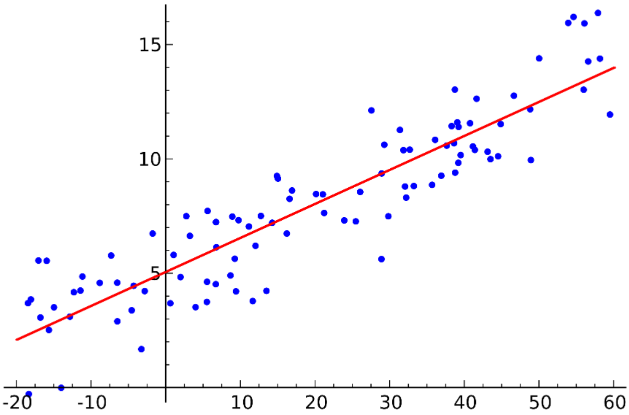


Figure 1: Regression

In unsupervised machine learning, however, problems of **clustering** or **grouping**, **association** (search for common sequences of objects, such as coffee and milk) and **dimensionality reduction** are distinguished. The machine will have to organize the information in its possession, the input data, in an intelligent way and learn from them which are the best results for the different situations that arise.

Clustering: is an unsupervised learning problem that requires the model to find groups of data points that have similarities to each other. The algorithm learns if and when it identifies a relationship between the data. The data is not categorized, but a rule is extracted that groups the input data according to characteristics that are obtained from the data itself. The most popular algorithm is K-Means Clustering. There are others such as DbScan or the hierarchical algorithm.

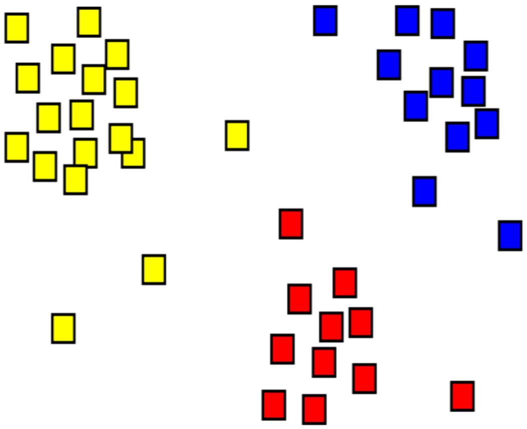


Figure 2: Clustering

Dimensionality reduction: is an unsupervised learning problem that requires the model to eliminate or combine variables that have no major impact on the final result. The learning algorithm eliminates irrelevant data (noise) and combines redundant (correlated) information to focus the analysis on those where a pattern emerges. This is often used in combination with classification or regression. Examples of algorithms that involve dimensionality reduction are: the decision tree, random forest, the removal or combination of variables with high correlation, Backward Feature Elimination, Forward, Feature Selection and PCA.

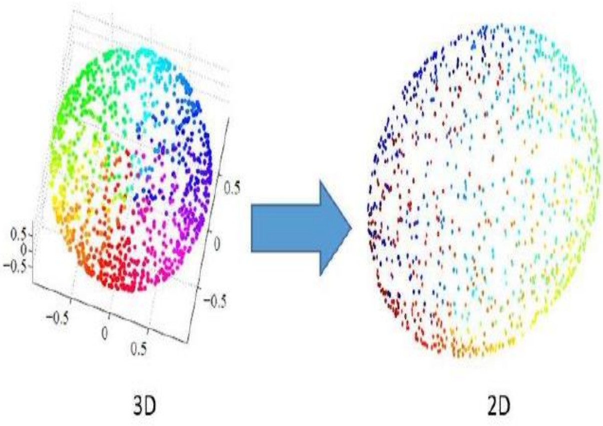


Figure 13: Dimensionality reduction

The following table represents the various types of learning and the related problems they solve:

Table 5: Type of problem vs learning

Type of problem/ learning	**Classification**	**Grouping**	**Regression**
Supervised	X		X
Unsupervised		X	
Partially supervised	X		X
For reinforcement			X

Finally, there are artificial **neural networks** that use some algorithms for learning inspired by the structure, functioning and connections of biological neural networks (i.e. those of human beings). In the case of so-called multilayer neural networks, we then enter the field of Deep Learning. Deep learning is based on different levels of representation, corresponding to hierarchies of factor or concept characteristics, where high-level concepts are defined on the basis of low-level ones. It is called "deep" because the machine's training takes place in the hidden layers of a deep neural network. The network is composed of the initial input layer, the final output layer and hidden intermediate layers called hidden layers, which are usually two.

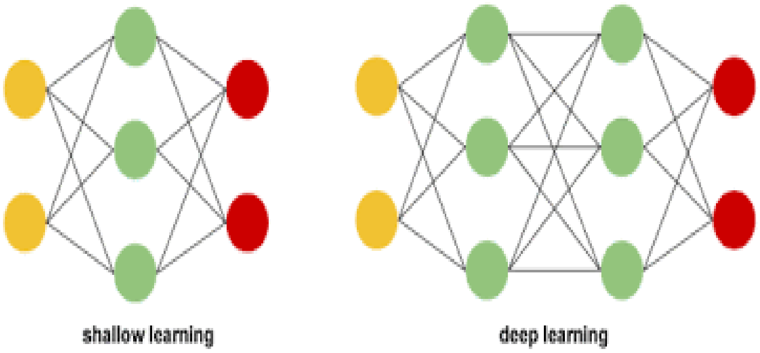


Figure 14: Simple network vs multilayer network

The network works like this: it receives the input data in the input layer, each layer calculates the values of the next layer and the last layer, called the output layer, returns the final result. Deep learning is used in character and image recognition, in the driverless driving sector, computer vision, natural language recognition and for automatic machine learning in general.

Very often machine learning is combined with the concept of data mining: a set of techniques and methodologies that aim to extract useful information from large quantities of data. Data mining mainly focuses on exploratory data analysis and therefore it can be said to mainly use the philosophy of unsupervised learning [18].

2.1.1 Supervised learning

Supervised Learning is a machine learning technique through which a model is built starting from labeled training data, with which predictions are made on unavailable or future data. In this type of learning, based on discrete class labels, you have a task based on classification techniques in the case in which the output signals present discrete values or regression in the case in which the output signals present continuous values [19].

The objective is therefore to develop a system that is able to predict the output values with respect to the inputs already having available the input-output pairs used to train the model for the knowledge of the interactions between them.

One of the most accredited definitions is the one written by Adam Geitgey:

«In supervised learning the resolution work is left to the computer. Once the mathematical function that led to solving a specific set of problems has been understood, it will be possible to reuse the function to answer any other similar problem» [20].

The success of supervised learning algorithms obviously depends on the number of experiences E, i.e. the number of inputs and outputs, that are provided to them, since the greater E the more effective the algorithms will be.
Examples of this type of algorithms, in particular classification ones, will be discussed in detail in chapter 4.

Classification

Classification is a supervised learning technique. Such a model must be able to correctly fit the input data and above all to be able to correctly predict class labels from data that it has never seen. The goal is to build models with good generalization capabilities.
The data is divided into:

- *training set:* set of data whose membership classes are specified
- *test set:* set of input data, which are also labeled, which is not used for training but for evaluating the performance of the classifier

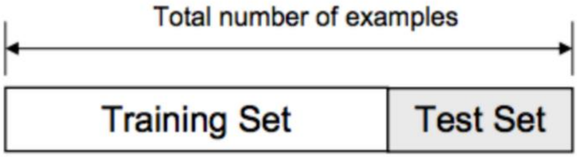
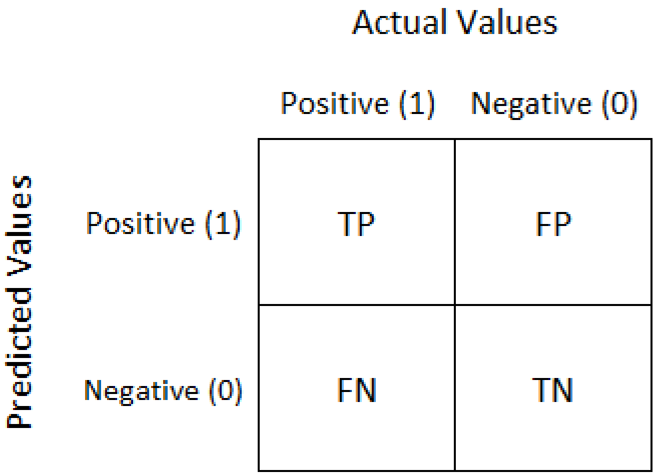


Figure 3: Train set e test set

The algorithm is trained on train data and evaluated on test data through types of metrics, for example the accuracy which represents the ratio between correct predictions and the total predictions.

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

The performance of the classifier is then evaluated taking into account the count of correctly predicted and incorrectly predicted test set data. These results are organized into a confusion matrix.



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 4: Confusion Matrix

It is possible to distinguish two types of classifiers:

- *linear*, which are simple and fast.

- *non-linear*, which are more precise but slower to process.

Regression

In regression the objective is always to find a relationship between the input and output variables. The forecast model produces a numerical estimate as output.

Given a predictor variable x and a response y, a line or curve is identified and drawn to decrease the distance between the points and the line or curve itself.

Taking the slope and the intersection point as a reference, it is possible to predict from these data a target variable to be used as a reference for new ones.

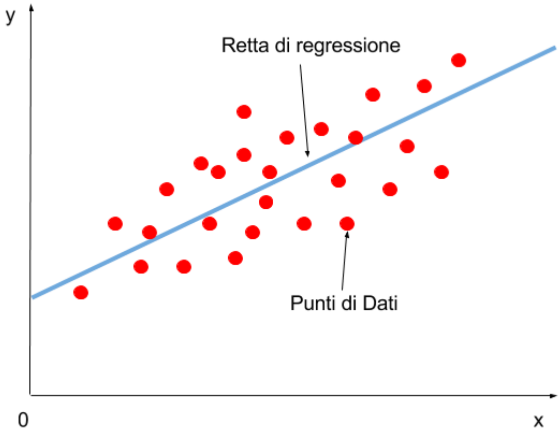


Figure 5: Regression

The objective is not to find the line that passes through all the points, but the one that is close to satisfying all the points in such a way as to have a good generalization of the model.

If the curve passed perfectly through all points, the model would adapt too much to the train data and therefore would have a poor performance on the prediction of new data.

It is possible to distinguish two types of regressors:

- *linear*: the algorithm is very simple and fast. The estimate is a straight line or plane.
- *non-linear* (polynomial): the learning algorithm is slower. Typically, the accuracy of the prediction model is higher than linear regressors because the prediction is a curve.

The same things said previously regarding the division of data into training sets and validation sets also apply to these types of algorithms. The difference compared to classifiers is the type of validation metric that is used, since in this case, for example, RMSE which is the mean square error can be used.

$$RMSE = \frac{\sqrt{\Sigma(y_i - \ y_{p,i})}^2}{m}$$

Where:

y_i is the true output

y_(p,i) is the predicted output

m is the number of elements chosen

In the case of regression the objective is to minimize the value of RMSE.

2.1.2 Overfitting, underfitting and cross validation

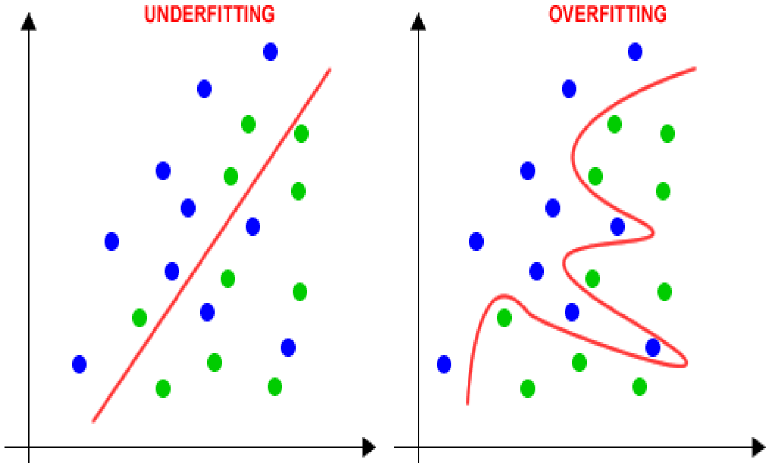


Figure 6: Overfitting vs underfitting

Overfitting and underfitting are two typical machine learning problems in which the model achieves poor performance after training, due to different reasons:

- *Underfitting* is a problem that occurs when the algorithm is based on few parameters. The model suffers from excessive discrepancy (high bias). It's too simple.

- *Overfitting* is a problem that occurs when the algorithm is based on too many parameters. In these cases, the variance becomes high, because the model is too sensitive to the training data.

We therefore speak of underfitting when the model is sized in such a way that the result of the predictions both in the test phase and in the training phase returns very high errors, or when the predictions in the train phase return a greater error than that in the test.

Instead, we talk about overfitting when the model returns optimal results in the training phase while returning much worse results in the testing phase, this means that the model has not generalized well from the learning data to the "invisible" data, i.e. those that it has never view.

The approach to determining whether a predictive model is subject to underfitting or overfitting the training data is to examine the prediction error on the train and test sets.

As a general rule, the more training data you feed the model, the less likely it is to overfit, as more data will result in the model being more accurate, while reducing the chance of overfitting.

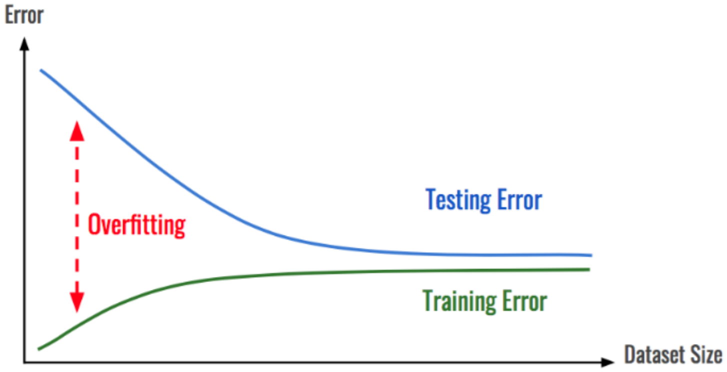


Figure 7: Overfitting

In order to counteract the phenomenon of overfitting, a powerful preventive measure is used: cross-validation. It uses the initial training data to generate multiple splits of mini-test sets in order to refine the model [21]. Through the validation set approach, the data set is divided into train and test: the train set is used to train the model, while the test set is used to evaluate the performance. By doing this, the model is tested on data it has never seen before.

It is not important to have a model with 100% accuracy on the train set if it is not capable of making a prediction on new data. By only subdividing the data set into train and test, you choose only one of the possible subdivision methods, therefore the model could be affected in terms of the significance of the data chosen, since they could be less than representative from the point of view of the desired phenomenon to describe. In fact, the accuracy depends on the initial split and to be sure of having the combination that leads to the best result it is necessary to introduce Cross-Validation. There are various cross-validation techniques such as:

- *Holdout*: The simplest type of data validation as it does a single split, but its evaluation can have high variance because of this, as doing a single split may cause the model not to generalize well.

- *Loocv* (Leave-one-out-cross-validation): the number of subdivisions is equal to the number of observations in the dataset. In one part there is a single observation, i.e. the test data, and in the other part all the other observations from the training dataset.

- *K-fold CV*: the most widespread validation that allows you to define k subdivisions of the dataset.

- *Stratified Cross Validation*: in which in each subdivision the distribution of samples between the classes is kept constant. Stratification is a technique where we reorder the data such that each fold has a good representation of the entire dataset.

- *ShuffleSplit*: a hybrid method between the holdout method and k-fold validation.

The k-fold CV technique will be covered in detail:

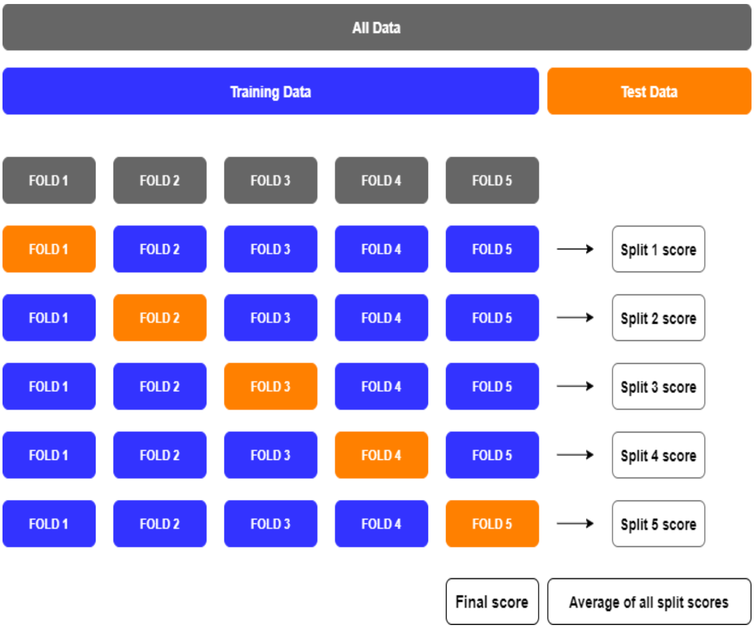


Figure 8: K-fold cross validation

In standard k-fold cross validation, the training set is divided into a number k of randomly drawn, disjoint subsamples or segments, which will be used in the training phase. The model will then make predictions on the k-th segment (for example, using the k-1 segments as model elements) and the error will be evaluated.

This means that each time, one of the k subsets is used as the test set, while the other k-1 subsets form the training set. At the end of the procedure, in the case of regression an average of the errors is calculated to provide a measure of the stability of the model (i.e. how well the model predicts new instances), for a classification model the predicted classes and the actual classes are instead compared through the accuracy which is the ratio between the correct number of predictions and the total number of elements in the validation set.

This technique guarantees a well-generalized model and a score that is not affected by the initial splitting decision. If the training of the model were done on the entire data set, the algorithm's score would be poor and generalized, since it would not be able to adapt well to the new data.

Cross-validation also allows you to tune the model's hyperparameters using only the original training set. This allows you to keep the test set as a truly invisible dataset for evaluating the final model.

2.1.3 Unsupervised learning

Unsupervised learning is a branch of machine learning that requires the system to be provided with a series of input data, which will be reclassified according to common characteristics to carry out reasoning and predictions on subsequent inputs. These types of algorithms work by comparing data and looking for similarities and differences. Compared to supervised learning, the machine does not know the classes a priori, but is only provided with unlabeled examples. An example is clustering, also called group analysis, is a set of multivariate data analysis techniques aimed at selecting and grouping homogeneous elements in a data set.

It consists of a set of methods aimed at grouping objects into homogeneous classes, in fact a cluster is a set of objects that present similarities between them, while dissimilarities with objects in other clusters. The input of a clustering algorithm consists of a sample of elements, while the output is given by a certain number of clusters in which the elements of the sample are divided based on a similarity measure, in such a way as to look for regularity in the available data.

Types of clustering

It is possible to make a classification of the various clustering techniques:

- Partitional vs hierarchical

In partitional clustering, group membership is defined through the distance from a representative point of the cluster, the centroid. When the desired number of clusters is fixed a priori, an example of an algorithm is k-means.

In hierarchical clustering, a partition hierarchy is constructed characterized by a number of groups that are visible through a tree structure.

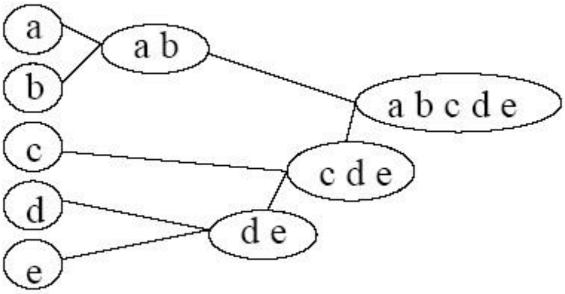


Figure 21: Hierarchical clustering

- Exclusive vs Non-exclusive

In exclusive clustering each element is assigned to a single cluster and therefore to each single group. The final clusters have no elements in common. This approach is called hard clustering. In non-exclusive clustering each element can belong to more than one cluster and with different degrees of membership. It is also called fuzzy clustering precisely because it uses fuzzy logic. The degree of membership is a number between 0 and 1, where 0 represents the non-belonging of that data to that cluster, while 1 represents the total belonging of the object to that cluster [22].

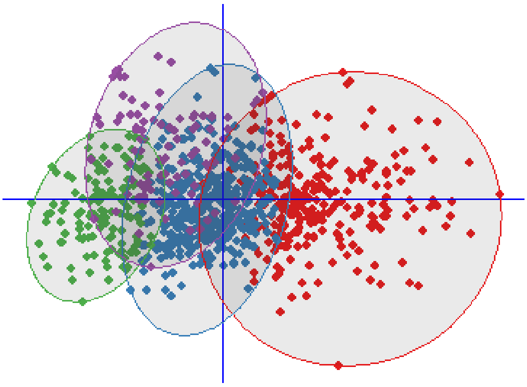


Figure 22: Non-exclusive clustering

- Complete vs Partial

Full clustering assigns each object to a cluster, while partial clustering does not. The rationale for a partial cluster is that some objects in a dataset may not belong to well-defined groups, for example some noise, hence outliers.

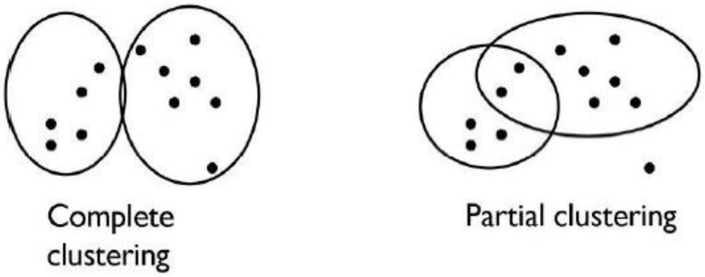


Figure 23: Complete vs partial clustering

- Agglomerative vs divided

Depending on the cluster generation technique, you can divide the algorithms like this:

- Agglomerative clustering algorithms (bottom-up)

They start by inserting each object in the set into its own cluster and then grouping them iteratively until a specific condition is reached (e.g. desired number of clusters). Each cluster contains a single point and at each iteration the “nearest” clusters are merged, until a single large cluster is obtained.

- Divisive (top-down) clustering algorithms

They start by placing all the objects in the set into a single cluster and then iteratively separate it into smaller clusters until a specific condition is reached. At each step a cluster is selected based on a measurement, and it is divided into smaller clusters. Usually a minimum number of points is set below which the cluster is not further subdivided.

Types of algorithms

- K-means

K-means is an unsupervised learning algorithm that solves the clustering problem, finding a fixed number of clusters in a data set.

The number of clusters is chosen a priori, before the initialization of the algorithm and the final result is influenced by the choice of the initial centroids, as the k-means does not guarantee the achievement of the global optimum but can settle at an optimal point local, i.e. the best cluster configuration that can be achieved given the initial conditions. The data are grouped based on the presence or absence of a certain similarity, based on the characteristics of each [21].

For each cluster, a point is defined, the centroid, which represents the midpoint of the cluster.

The objective of the algorithm is to minimize an objective function, i.e. the total intra-group variance: the sum of the squared Euclidean distances (total within sum of squares) between each element belonging to the data set and the corresponding centroids must be minimized. The best cluster is in fact the one that has the minimum distance between its members and the maximum distance with the members of the other clusters. It is an iterative algorithm, meaning that it repeatedly carries out some of its phases:

- Initialization: the input parameters to execute the algorithm are defined. The k points are positioned in space, which represent the initial centroids.

- Cluster assignment: each data point is assigned to the closest cluster (or centroid);

- Update of the centroid position: recalculates the exact point of the centroid and consequently modifies its position.

The algorithm ends when the position of the centroids does not change, therefore when a point of convergence is reached such that there are no more changes in the clusters.

The stop condition is represented by one of the following options:

- no data points change clusters;
- the sum of distances is reduced to a minimum;
- a maximum number of iterations is reached.

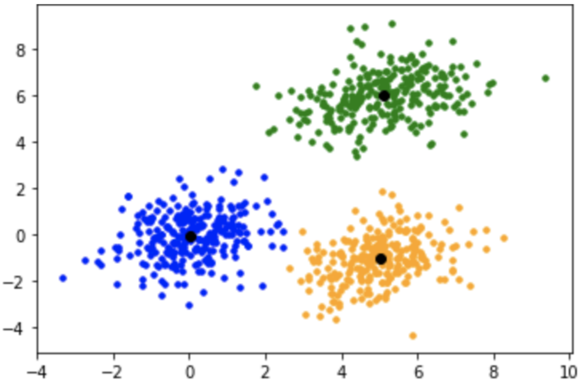


Figure 9: K-means

- Hierarchical algorithm

Hierarchical clustering is an agglomerative algorithm which, unlike k-means, does not need the a priori identification of the optimal number of clusters [21].

It consists of several phases:

1. Initially each element of the data set forms a separate cluster
2. At each iteration the most similar clusters are merged, gradually expanding the similarity threshold
3. The iteration ends when all the objects are merged into a single cluster, where all the elements are considered similar.

It is necessary to use specific metrics (Euclidean distance, Manhattan distance) and linking criteria, which identify the dissimilarity between two sets of elements.

The clusters are represented through a dendogram, or tree graph, as shown in the following figure:

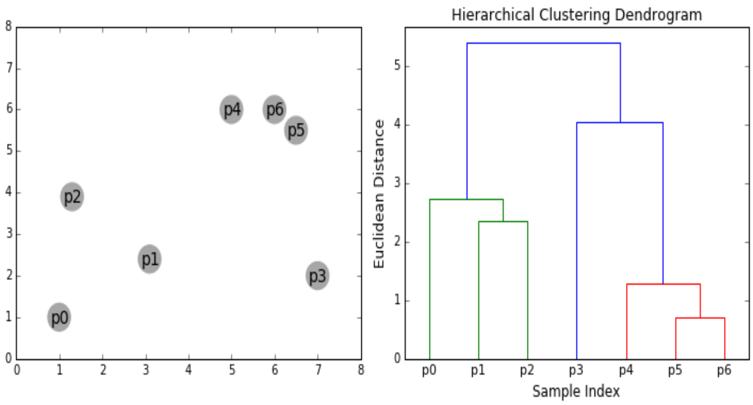


Figure 25: Hierarchical algorithm

The "distance" between the clusters is shown on the ordinate axis and the various input data are shown on the horizontal axis.

At the first iteration when each point constitutes a cluster, there will be n clusters, so to calculate the distance just take into account the Euclidean distance between the elements precisely because there will be n elements and n clusters. In the following iterations, when clusters consisting of multiple elements begin to form, the distance between the various elements of the clusters can be calculated in various ways: there are 5 linkage criteria.

These linkage criteria between clusters are based and differentiated on the concept of difference between clusters: single linkage method, complete linkage method, average linkage method, Ward's method, centroid method.

The number of clusters is defined by the user and is represented by the cutting point of the tree, called the cut-off point. It is identified by seeing a sharp change in the length of the dendogram. In the figure, if the cut is made at a distance of 3, we can see that there are 3 clusters: p0, p1,p2 – p3 – p4,p5,p6.

- DB-Scan

DB-SCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based approach, which identifies high- and low-density regions [21].

It is necessary to first identify the concept of density: in the center-based approach in which the density of a point in the data set is estimated by counting the number of points within a specific radius, called Eps. The point itself is included within the radius.

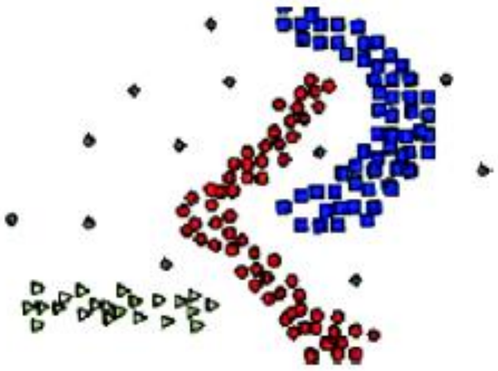


Figure 10: DB-scan

In DB-Scan the points are classified as core points, border points and noise points. Points that are close to each other are grouped together based on a distance measurement (usually the Euclidean distance) and a minimum number of points. To identify these points it is necessary to define:

- Eps, i.e. the radius of the hypersphere, which specifies how close the points must be to each other to be considered part of a cluster. It is therefore considered as the minimum distance that two points must have to be considered close.
- MinPoint, i.e. the minimum number of points to form a dense region.

Based on these two values the points are classified as follows:

- Core points: points that have at least MinPts within a distance of Eps.
- Border points: points that have a lower density than MinPts, but there is a core point nearby.
- Noise points: points that are neither core nor border.

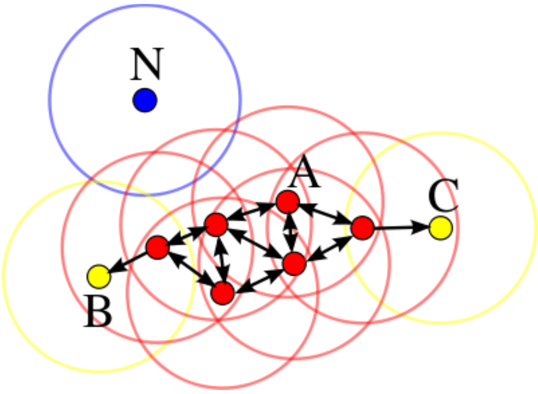


Figure 11: Core, border, and noise points

In the figure, once the radius Eps and MinPts = 4 have been defined, point A and the other red points are Core, the yellow points are Border and point N, in blue, is a Noise.

The algorithm works like this:

1. Once Eps and MinPts have been established, all points are classified as core, border or noise.
2. Noise points are eliminated.
3. Core points that are closer than Eps are merged and form a cluster.
4. Border points are assigned to the closest cluster.

CHAPTER 3

DIMENSIONALITY REDUCTION TECHNIQUES

3.1 FEATURE SELECTION

A characteristic is an individual measurable property of the observed process. Using a set of features, any machine learning algorithm can perform classification. In recent years in machine learning or pattern recognition applications, the feature domain has expanded from tens to hundreds of variables or features used in such applications. Several techniques are developed to address the problem of reducing irrelevant and redundant variables that are a burden for challenging tasks. Feature selection is therefore the process of reducing the number of input variables when developing a predictive model:

"Feature selection focuses primarily on removing uninformative or redundant predictors from the model." [26]

The goal of feature selection is therefore to select a subset of variables from the input in such a way as to reduce the effects of noise or irrelevant variables and still provide good prediction results [27].

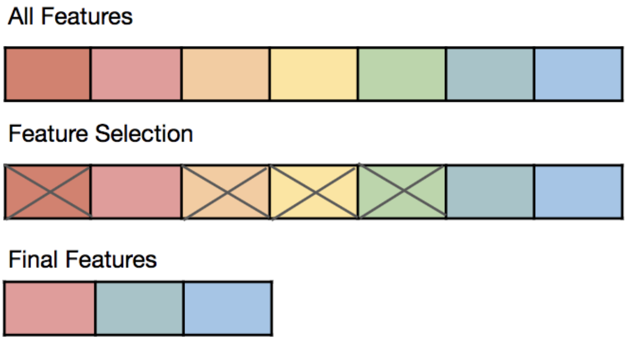


Figure 28: Feature selection

When there are a large number of features, the machine learning model you are training can suffer in performance. Within the data set, some input variables may be positively or negatively correlated with each other, or some are less informative than others, and as a result poorer results are obtained than one might expect. One way to overcome this problem is to use a Feature Selection method.

The advantages that can be obtained from this technique, but in general from any dimensionality reduction technique are:

- Reduced overfitting: less redundant data means fewer opportunities to make decisions based on noise.
- Improved accuracy: Less misleading data means modeling accuracy improves.
- Reduced training time: Fewer data points reduce algorithm complexity and algorithms train faster.

3.1.1 Types of techniques

Feature selection is the process in which those features that contribute most to the prediction of the output are automatically or manually selected. It is therefore desirable, without losing too much information from the initial dataset, to reduce the number of input variables both to reduce the computational cost of modeling and to improve the performance of the model, which could become overfitting, but also in terms of time. for training.

There are two main feature selection techniques:

- *unsupervised*: these are methods in which the destination variable is not used
- *supervised*: these are methods in which the destination variable is used.

Unsupervised methods remove redundant variables through the correlation coefficient. Supervised methods, on the other hand, can be divided into:

- *Wrappers*: look for well-performing feature subsets. They are based on greedy algorithms (algorithms that make at each step the choice that at that moment seems the best, locally optimal, in the hope of obtaining a globally optimal solution), which aim to find the best possible combination of features that translate into the model with the best performance. The algorithm is trained using a subset of features in an iterative manner. This will be computationally expensive and often impractical in case of exhaustive search.

- *Filter*: Select subsets of items based on their relationship to the goal. They allow you to select features from a dataset independently for any machine learning algorithm. They are based only on the characteristics of these variables, so the features are filtered from the data before learning begins. They use statistical measures, to evaluate the correlation or dependence between input variables which can be filtered to choose the most relevant characteristics. Measures must be carefully chosen based on the data type of the input variable and the output or response variable. This method evaluates the relationship between each input variable and the target variable through the use of statistics and the selection of those input variables that have the strongest relationship with the target variable. They are fast and effective methods, although the choice of statistical measures depends on the data type of both input and output variables. As such, it can be difficult for a machine learning practitioner to select an appropriate statistical measure for a dataset when performing filter-based feature selection. An example of statistical measures is Chi-Square, Pearson correlation coefficient, Fisher Score.

- *Intrinsic:* These are algorithms that perform automatic feature selection during training. Intrinsic methods complete the feature selection process within the construction of the machine learning algorithm itself. In other words, they perform feature selection during model training, which is why they are called built-in methods. A learning algorithm takes advantage of its own variable selection process and simultaneously performs feature selection with classification/regression. (e.g. Decision Trees and Random Forest). In the next paragraph they will be explored in detail.

3.1.2 Built-in methods

Decision trees and therefore also random forests fall into the category of embedded methods. The built-in methods are simple, fast, and combine the qualities of filter and wrapper methods. They are implemented by algorithms that have their own built-in feature selection methods.

They have the following advantages:

- They are extremely precise
- They generalize better
- They are interpretable

Random forests are one of the most popular machine learning algorithms. They are remarkably successful because they generally provide good predictive performance, low overfitting and easy interpretability. This interpretability is given by the fact that it is simple to derive the importance of each variable in the tree's decision, therefore, it is easy to calculate how much each variable is contributing to the decision. Random forests consist of aggregation of decision trees, each of which is built on a random extraction of observations from the dataset and a random extraction of features. Not all trees see all features or all observations, and this ensures that trees are decorrelated to each other and therefore less prone to overfitting.

The calculation of the importance of the variable, necessary to subsequently carry out the selection of the variables, varies depending on the problem being treated:

- For classification, the measure of impurity is the Gini impurity or information gain/entropy.
- For regression the measure of impurity is the variance.

During training, you can calculate how much each characteristic reduces impurity. The more a characteristic decreases impurity, the more important it is. The decrease in impurity from each feature can be averaged across trees to determine the final importance of the variable.

Intuitively, features that are selected at the top of trees are in general more important than features that are selected at the final nodes of trees, since higher splits generally lead to greater information gains.

The important features not only provide an overview of the features with high weight and frequently used by the model but also the features that are slowing down our model. Therefore, importance scores are useful for selecting features to eliminate (lower scores) or those to keep (higher scores).

Variable importance is usually followed by variable selection. The difference between using the model with all the variables or with a subset of them can be evaluated in terms of accuracy. To decide the number of features to choose, one should find a number such that neither too few nor too many features are used in the model. The selection of variables is a compromise between the loss of complexity and the gain in execution speed.

The feature importance in a classification problem is then calculated as the decrease in node impurity weighted by the probability of reaching that node. The node probability can be calculated by the number of samples reaching the node, divided by the total number of samples. The higher the value, the more important the feature. The importance of a node j for each decision tree is calculated through the Gini index. A more detailed explanation on the calculation of this index will be covered in the next chapter when explaining the algorithm.

CHAPTER 4

SUPERVISED ALGORITHMS

4.1 DECISION TREE

Decision trees are a representation of learning in machine learning. They are methods widely used in supervised learning, therefore they are part of that type of algorithms in which the label of the input data to the model is provided.

The decision tree is a very popular tool for classification and prediction. It has a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents a test result, and each leaf node (terminal node) contains a class label. It is in fact built through an algorithmic approach that identifies different ways to divide a data set based on different conditions. Decision rules are generally in the form of if-then-else statements. The deeper the tree, the more complex the rules and the more suitable the model. It is a type of non-parametric algorithm, meaning there is no underlying assumption about the distribution of the data. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from data characteristics. There is an important difference within the family of decision trees:

- There are decision trees used to predict categorical variables and are called classification trees. They predict a class through a voting system where the majority class within a leaf node wins. For example: (is the patient who presents certain electrocardiogram values at risk of having a heart attack or not?).

- Decision trees used to predict quantitative variables are known in the literature as regression trees. They provide a numerical value calculated based on the distribution of the target within a node [31].

From a formal point of view, a tree is a finite set of elements called nodes; the node from which the subsequent nodes branch off is called the root. The set of nodes, excluding the root node, can be divided into h distinct sets, S_1....S_h called subtrees; a node is called parent with respect to the nodes it generates, while it is called child with respect to the node from which it descends. The final nodes are called leaves. The threshold values that divide the units of a given node are called split [32].

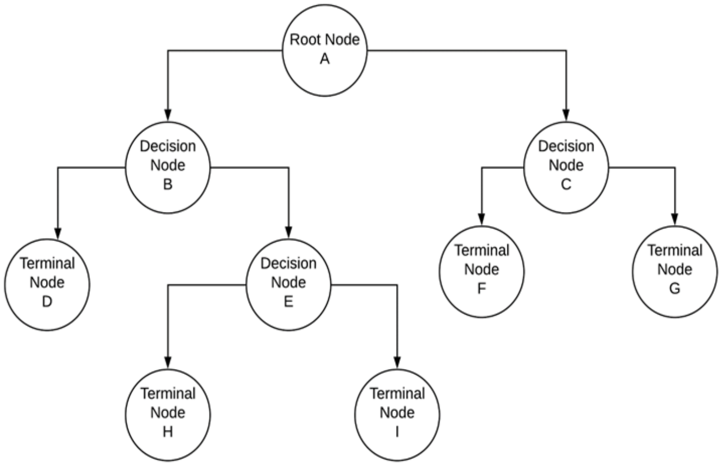


Figure 29: Type of nodes

It is therefore possible to define:

• Root node: represents the sample or population which is divided into further homogeneous groups

• Division: process of dividing nodes into two subnodes

• Decision node: when a subnode divides into further subnodes based on a certain condition, it is called a decision node precisely because it represents a decision that is satisfied

• Leaf or terminal node: secondary nodes that represent a result and therefore do not divide further

Connecting these different nodes is what we call "branches". Nodes and branches can be used over and over in any number of combinations to create trees of varying complexity. A very important concept in decision trees is information gain: to split nodes using a condition (the most informative feature) it is necessary to define an objective function that can be optimized. In the decision tree algorithm, the information gain at each split is maximized. There are several impurity measures that are used to measure information gain such as, for example, Gini impurity or entropy, which will be covered in the following paragraphs.

The structure of a decision tree can become very complicated, especially if the dataset has a large number of input variables and an output variable with different classes. In such situations, letting the tree "grow" without establishing a limit of any kind can make the tree obtained difficult to interpret, creating a large number of rules, and therefore losing its predictive power. There are, therefore, control criteria that limit the growth of trees which are based either on the maximum number of rules obtainable from the classification or on the maximum "depth" reachable by the tree or even on the minimum number of observations that must be present in each node to be able to carry out the division (splitting) in that node.

This area also includes the tree pruning phase which will be treated in detail later and which consists in obtaining the smallest subtree from a tree, which in fact does not compromise the accuracy of the classification/prediction made possible by the tree.

A branch or subtree that the user judges to be irrelevant because it has a small number of cases can be removed by effectively carrying out a "pruning" operation. For example, the CART algorithm, (Classification And Regression Trees, the most widespread algorithm for building classification and regression trees) carries out the pruning phase by simply checking whether the improvement in accuracy justifies the additional presence of other nodes. The path that leads to the construction of decision trees is certainly not simple; however, in the next paragraphs the methodology that allows the construction of the trees will be described.

4.1.1 Preliminary notions

We define a set of primary notions for building a binary classification tree:

Consider a dependent variable Y which presents J modes if qualitative or is divided into J classes if quantitative and the p explanatory variables, X_1…. X_p, quantitative or qualitative, recorded on N statistical units. The construction of a decision tree can be considered as a stepwise procedure through which the N units or observations are progressively divided, according to an optimization criterion, into a series of disjoint subgroups, which present a greater level of homogeneity within them compared to the initial set.

At each step of the process the heterogeneity of the groups is reduced compared to the previous step. At the end, the leaves of the tree present such a degree of homogeneity that they can be attributed to one of the J starting classes.

By building a tree, therefore, it is possible to identify a rule that allows new observations to be classified into one of the J classes of the variable Y.

A tree is defined as a set T of positive integers, together with two functions l (·) and r(·) (denoting the left and right nodes respectively). Each member contained in T identifies a node of the tree and for each node the following two rules apply: For every t ∈T, if l(t) = 0 or r(t) = 0 we are in the presence of a final node, if instead l(t) > 0 or r(t) > 0 then the node is internal. Apart from the root, the smallest integer t = 1 in T, there is a unique parent s ∈ T for every node; a subtree is a non-empty subset T_1 of T, together with two functions l_1 and r_1 for which:

$$l_1\ (t) = \begin{cases} l(t) & if\ l(t) \in\ T_1 \\ 0 & otherwise \end{cases}$$

$$r_1\ (t) = \begin{cases} r(t) & if\ r(t) \in\ T_1 \\ 0 & otherwise \end{cases}$$

And so that T_1, l_1 (·) and r_1 (·) form a tree.

Let (T)̃ be the set of final nodes and {u(t), t ∈ (T)̃} a partition of the data space R^p. This implies that u(t) is a subspace associated with the final node.

Let J = 1…J be the set of possible class labels and j(t) a generic element of the set.

A classification tree is therefore formed by a tree T, together with the class labels {j(t), t ∈ (T)̃} and the partition { u(t), t

∈ (T ̃) }. Therefore associated with each final node there is a region of the data space, belonging to a certain class.

A classification tree is constructed using a certain data set D = {(x_i,y_i) i=1….n} where y_i is the class corresponding to x_i.

If we indicate with N(t) the number of observations of our data set D for which x_i ∈ u(t) and with N_j (t) the number of observations for which x_i ∈ u(t) and y_i = j, then ∑j Nj (t)=N(t), then we indicate with p(·) the probability of the event contained between the round brackets, we can define the following estimate of p(x ∈u(t)).

$$p(t) = \frac{N(t)}{N} \tag{4.1}$$

Instead, an estimate of p(y = j | x ∈ u(t)) is:

$$p(j|t) = \frac{N_j(t)}{N(t)} \tag{4.2}$$

Furthermore, we can define, by setting t_L=l(t) and t_R=r(t), the following estimates of p(x ∈u(t_L) | x ∈ u(t)) and p(x ∈u(t_R) | x ∈ u(t)):

$$p_L = \frac{p(t_L)}{p(t)} \tag{4.3}$$

$$p_R = \frac{p(t_R)}{p(t)} \tag{4.4}$$

At this point at node t the label j if:

$$p(j|t) = \ max_i \, p(i|t) \tag{4.5}$$

Then each node t is assigned the corresponding label j, depending on the proportion of observations in each class in u(t).

4.1.2 The phases of building a tree

The construction of any tree includes three steps:

- select a splitting rule for each node; this involves determining those variables, together with the respective threshold value, that will be used to partition the data set at each node.

- determine which are the terminal nodes; for each node it is necessary to decide when to continue with the splits, when to stop and consider the node as a terminal and subsequently assign it a label. Without an appropriate rule, there is a risk of building trees that are too large with little generalization capacity, or trees that are too small and which instead poorly approximate the data.

- assign labels to each terminal node, for example minimizing the expected value of misclassification.

Each of the three phases is briefly illustrated below:
The central phase of the procedure is the subdivision of the units belonging to a node and consequently the choice of the criterion on the basis of which to carry out this division.
The splitting criterion consists in the calculation of a statistical index, which allows you to select the best partition corresponding to each individual predictor, among all the possible ones; among all the predictors, the best one will be selected in relation to the chosen heterogeneity reduction criterion. The goodness of this criterion must be consistent, i.e. the initial set must be divided into groups that are as homogeneous as possible internally and as heterogeneous as

possible between them. Generally the best split is sought by analyzing all the explanatory variables.

The construction of a tree is a recursive procedure, it is therefore necessary to define one or more stop rules, upon occurrence of which the process stops. The desirable properties of a stopping rule are simplicity and discriminatory power.

Based on the first property, between two stopping rules, the one that determines the smaller tree is chosen since it is more readable when interpreting the results.

The second property concerns the need to obtain trees capable of distinguishing statistical units belonging to different classes in the most effective way possible. The two properties are evidently opposed and difficult to reconcile.

The best-known construction methods use stopping rules based on the minimum number of terminal nodes or the maximum depth that the tree can reach. The CART methodology is innovative, which carries out a "pruning" phase of the less significant branches, after having built the maximum size tree. A general algorithm for a decision tree can be described as follows:

1. Choose the best feature, that is, the one that best divides or separates the data.

2. Ask the relevant question.
3. Follow the response path.
4. Go back to step 1 until you get to the answer.

After building the tree it is necessary to establish which class it corresponds to each final node. Three cases can be distinguished:

- the leaf includes cases belonging to only one class; the leaf is then assigned the label corresponding to the units that are part of it, according to the unanimity rule;

- in the leaf there are statistical units of different classes, but one of these has a higher frequency than the others; according to the majority rule, the class of the leaf corresponds to the one with maximum frequency;

- the units of the leaf belong to different classes with the same frequency; in this case you fall into a zone of indecision.

After assigning a class to each single final node, it is possible to proceed with the classification of new cases unrelated to the sample used to build the tree.

By applying the tree classification rule, each individual case falls into a leaf and is labeled according to the class assigned to the corresponding leaf.

4.1.3 Construction of a classification tree

We now describe the CART (Classification And Regression Trees) methodology, introduced by Breiman, Friedman, Olshen and Stone, in 1984, one of the best-known algorithms for building binary trees, i.e. trees in which only two branches correspond to each node.

It is divided into two phases [33]:

1. generation of the tree.
2. pruning of the tree.

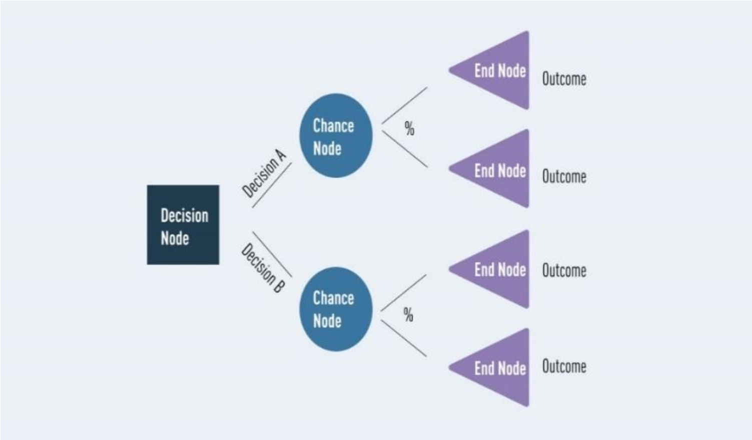


Figure 30: Decision tree splitting

In the building phase, the objective is to increase the size of the tree, in terms of arcs and nodes, in such a way as to define splitting rules, capable of identifying homogeneous classes in terms of output. The natural conclusion of this phase is a particularly "thick" tree, a characteristic which may not be a positive factor as overfitting problems could arise. In the pruning phase, i.e. pruning, branches that do not add significant information value to the tree are eliminated since a particularly thick tree may not be a particularly reliable structure in predictive terms.

With the first phase, complete trees are often obtained, but quite complex, in the sense that they are made up of a very large number of nodes. For this reason, in order for the tree to be effectively useful in classifying objects and providing effective rules, it is necessary to prune the redundancy of the tree, eliminating the less significant branches through the pruning technique.

4.1.4 Splitting rules

It is necessary to define a splitting rule to decide which variable should be used at a certain node to divide the sample into subgroups, and to decide the threshold value to use. To choose the attribute and the related splitting, particular dispersion indices of the values of the categorical class attribute are used:

- Gini index
- Information gain

During the tree generation phase, we start from the totality of the observations belonging to the training set and begin a binary division into classes. First of all, the method according to which the splits are to be carried out must be established; this method is based on the definition of impurity which is a function of the fraction of observations classified in each class. For classification problems the impurity function that can be used is the Gini coefficient which provides an indication of how "pure" the leaf nodes are. At a certain node t of the tree under construction, and with respect to the corresponding partition of the training dataset, the Gini impurity index is defined as:

$$G(t) = 1 - \sum_j [p(j|t)]^2 \qquad (4.6)$$

Where p(j | t) is the relative frequency of class j at node t.

The Gini impurity measures how often any element in the dataset will be mislabeled when randomly labeled. The minimum value of the Gini Index is 0. This happens when the node is pure, this means that all the elements contained in the node are of a unique class. Therefore, this node will not be split again. The optimal split is chosen from the features with a lower Gini index, since the homogeneity of the node is higher. The Gini index measures the impurity of the dataset corresponding to node t and presents:

- A maximum value of (1- 1/n_c) when records are equally distributed across all classes.
- A minimum value of 0, when all records belong to a single class.

The total impurity of the generic T-tree is as follows:

$$G(T) = \sum_{t \in \tilde{T}} G(t)p(t) \tag{4.7}$$

One measure of the goodness of a split is the change in the impurity function; at each node t we choose the split s* that maximizes the decrease in impurities:

$$\Delta G(s^*, t) = G(t) - (\, G(t_L)p_L + G(t_R)p_R) \tag{4.8}$$

The optimal split s* will split each node t into t_L and t_R , with p_L proportion of cases of t going into t_L and the remaining proportion p_R into t_R.

At each node, the decision tree searches among the features for the value to split them on, which results in the greatest reduction in Gini Impurity. In the various phases, the algorithm will choose splitter attributes among those present in the training set and will try different values so that the impurity function is minimized in each node. Minimizing this function coincides with finding the attributes and the respective threshold value in the various phases, which make the most correct classification possible and which therefore should provide greater information in that phase. The procedure is iterative and will stop when it is no longer possible to reduce the impurity function by manipulating the choice of attributes and/or their threshold value. Breiman et al. (1984) have shown that split selection that maximizes the impurity decrease (4.8) is equivalent to split selection that minimizes the total impurity of the tree. This means that the local optimization criterion of a classification tree is equivalent to its global optimization. So when a node t'is split into k partitions (children), the quality of the split is calculated as

$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i) \qquad (4.9)$$

Where:
n_i = number of records in the partition of child i.
n = number of records in the dataset at node t.
n_i/n constitutes the weight of the Gini value.

Given the dataset associated with node t, we choose the attribute that provides the smallest GINI split (t) to partition the dataset.

In classification trees, the Gini index is used to calculate the impurity of a data partition. In CART, binary divisions are performed, then the Gini index will be calculated as the weighted sum of the resulting partitions, and the division with the smallest Gini index will be selected.

Similarly, the attribute that maximizes the gain of the Gini index can be considered, i.e. the one that gives a greater decrease, since it is necessary to consider the difference between the Gini index of the parent node and the child nodes.

$$GINI_{gain} = G(t) - GINI_{split} \tag{4.10}$$

When training a decision tree, the attribute that provides the smallest GINI split is chosen to split the node. To obtain information gain for an attribute, the weighted impurities of the branches are subtracted from the original impurity, i.e. that of the parent. The best split can also be chosen by maximizing the Gini gain. So in summary, the impurity measure can be minimized, or the information gain can be maximized, which is the impurity of the parent node minus the weighted average impurity of its child nodes. The best split is the feature and threshold that produces the greatest information gain in the node. Example of calculating the Gini split for a partition with k = 2:

Table 2: Gini Split

	N3	**N4**
C1	5	1
C2	2	4
Gini Split = 0,371		

Gini(N3) = $1-\left(\frac{5}{7}\right)^2-\left(\frac{2}{7}\right)^2 = 0{,}408$

Gini(N4) = $1-\left(\frac{1}{5}\right)^2-\left(\frac{4}{5}\right)^2 = 0{,}320$

Gini Split = $\frac{7}{12} * 0{,}408 + \frac{5}{12} * 0{,}320 = 0{,}371$

Another measure that represents the impurity of the node is **entropy**. In general, if the observations are classified into j classes, the entropy is defined as:

$$H(S) = -\sum_j F_j log_2 F_j \tag{4.11}$$

Where F_j=p(j│t) is the relative frequency of class j at node t.

It measures the disorder of a node and has a value:

- Maximum, equal to log (n_c), when the records are equally distributed among all classes.
- Minimum, equal to 0, when all records belong to a single class.

Entropy is a measure of the impurity of the observations that are considered for the construction of decision trees; a high entropy value expresses the inhomogeneity that characterizes the data space, i.e. greater difficulty in assigning each observation to its own class on the basis of the attributes that characterize the class: the higher the entropy, the more difficult it will be to identify the attributes that actually characterize the classes.

In general, starting from a situation of maximum disorder in which H(S) =1 or from any high entropy value, a partition of the data carried out with respect to a certain attribute X would lead to a new value H'(S) such that result H'(S) ≤ H(S) and therefore to a release of entropy. This includes the concept of information gain, defined as the decrease in entropy obtained by partitioning the data with respect to a certain attribute. If we indicate with H(S) the initial entropy value and with H (S, A) the entropy value after partitioning the data based on the X attribute, the information gain, which we will indicate with G, is given by :

$$G = H(S) - H(S, X) \qquad (4.12)$$

This quantity is greater the greater the decrease in entropy after partitioning the data with the X attribute. Therefore, one criterion for choosing the nodes of a possible classification tree consists in choosing from time to time the X attribute that gives a greater decrease in entropy or which similarly maximizes the information gain. The information gain has very high values corresponding to attributes that are highly informative and therefore help to identify the class to which the observations belong. Often, however, the more informative the attributes are, the more they lose generality; for example, in the database of a telephone company, the tax code field is highly informative, therefore it has a high information gain value, since it certainly identifies the user, but it is not generalizable at all. The ideal is therefore to identify highly informative fields with a good degree of generalization. Information gain is a statistical property that measures how much a given attribute separates training examples based on their target classification. It is calculated as the decrease in entropy after the dataset is split on an attribute, and then calculates the difference between the entropy before splitting and the average entropy after splitting the dataset based on the attribute values data. Building a decision tree consists of finding an attribute that returns the maximum information gain and minimum entropy. When a node t is divided into k partitions (children), the quality of the division is calculated as information Gain:

$$GAIN_{split} = Entropy(t) - \left(\sum_{i=1}^{k} \frac{n_i}{n} Entropy(i) \right) \quad (4.13)$$

Where:

n_i = number of partition (child) records i.

n = number of records in the dataset at node t.

n_i/n constitutes the weight of the various Entropy(i).

It is therefore necessary to choose the split that achieves the greatest reduction, therefore the one that maximizes the information gain.

Generalising, given a node p with records belonging to k classes and its partitioning into n child nodes, the overall impurity of the split is given by the following formula, where meas () is one of the measures introduced:

$$Impurity_{split} = \sum_{i=1}^{n} \frac{m_i}{m} meas(i) \qquad (4.14)$$

Where:

m = number of records in parent p

m_i = number of records in child i

To determine the best split, the gain of the split is considered which is given by the difference between the purity value before the split, therefore that of the parent node and after the split, therefore that of the child nodes. The internal workings of both methods are very similar, in fact neither metric produces a more accurate tree than the other and both are used to calculate the split after each new split. Comparing them both though, Gini Impurity is more efficient than entropy in terms of computing power. As can be seen from the graph, the entropy first increases to 1 and then begins to decrease, while the Gini index rises to 0.5 and then begins to decrease, therefore requiring less computing power.

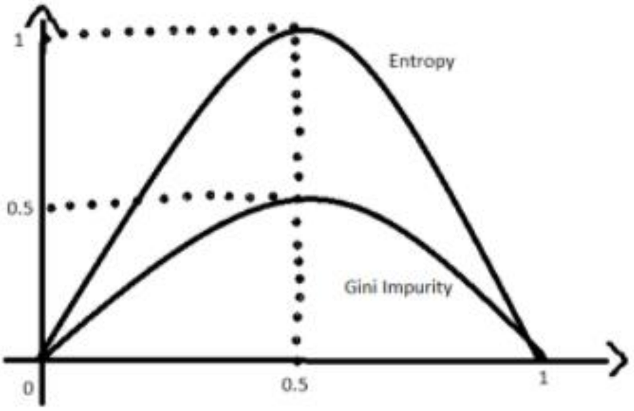


Figure 31: Entropy vs Gini

The entropy range is therefore between 0 and 1 while the Gini Impurity range is between 0 and 0.5.

From a computational point of view, entropy is more complex since it uses logarithms and consequently the calculation of the Gini Index will be faster.

4.1.5 Depth of the tree

After building the tree and choosing the splitting rule, it is necessary to define its depth.

The fundamental question to ask is: how deep, i.e. complex, must the tree be? If an overly complex tree is grown it risks over-fitting to the training data resulting in poor generalization performance. As a result, a balance must be struck between depth and tree complexity to optimize predictive performance on unseen future data.

To find this balance, there are two main approaches: early stopping and pruning.

- Stopping early explicitly limits tree growth. There are various ways to limit the growth of the tree, but there are two most common approaches which are to limit the depth of the tree to a certain level (max_depth) or to limit the minimum number of observations (min_n) allowed in any terminal node: When you limit the depth of the tree, it stops splitting after a certain depth. The lower the tree, the lower the variance of the predictions. When you limit the minimum terminal node size (for example, leaf nodes must contain at least 10 observations for predictions) you decide not to split intermediate nodes that contain too few data points. If the minimum terminal node size is too low, such as a value of 1, this results in high variance and poor generalization. On the other hand, high values limit further subdivisions, thus reducing the variance.

- Pruning: An alternative to explicitly specifying the depth of a decision tree is to grow a very large and complex tree and then prune it back to find an optimal subtree. We find the optimal substructure using a cost complexity parameter (α) that penalizes an objective function that we will discuss in the next paragraph.

4.1.6 Pruning

After choosing the splitting rule, it is possible to grow the tree by successive bipartitions of the nodes, but it is necessary to establish a criterion for stopping; we could possibly continue until each final node contains only one observation, but as mentioned previously this would lead to very large trees, which would have overfitting problems. In some situations, stopping rules don't work well. An alternative method for building a decision tree model is to first grow a large tree, and then prune it to the optimal size by removing nodes that provide the least additional information. There are two types of pruning:

- pre-pruning: Use chi-square tests or multiple comparison adjustment methods to prevent the generation of insignificant branches.

- post-pruning: This is used after generating a complete decision tree to remove branches in a way that improves the overall classification accuracy when applied to the validation dataset.

Pruning is therefore a general technique that can be applied to both regression and classification trees that reduces the size of decision trees by removing sections of the tree that are of little importance. Pruning reduces the complexity of the final model and therefore improves predictive accuracy by reducing overfitting. The metric used for tree pruning is the misclassification rate, if the goal is prediction accuracy. We define the following notation:

Let *R(t)* be the real number associated with node t of a given tree T. If t is a final node, for which t $\in \tilde{T}$ must therefore hold, (so let $\tilde{T}$ be the set of final nodes) then *R(t)* represents the proportion of misclassified observations; indicating with *M(t)* the number of observations in *u(t)* (data subspace associated with a final node) that do not belong to the class associated with that final node and with N the total number of observations, then:

$$R(t) = \frac{M(t)}{N} \tag{4.15}$$

Let the quantity *R(t)* be given by:

$$R(t) = r(t)p(t) \tag{4.16}$$

Where *r(t)* is the estimate by resubstitution of the probability of misclassification given by:

$$r(t) = 1 - \; max_j p(j|t) \tag{4.17}$$

While *p(t)* and p(j|t) are given by formulas 4.1 and 4.2

Then the misclassification rate associated with the generic T-tree is:

$$R(T) = \; \sum_{t \, \in \tilde{T}} R(t) \tag{4.18}$$

This value can be estimated, as well as by re-substitution, also by means of a test set independent of the training set and with cross-validation. Pruning works like this: initially it is necessary to construct the maximal tree T_{max} whose final nodes are made up of cases belonging to the same class or at most a single case.

We then select the subtrees that can be obtained by cutting Tmax at certain points and estimate the misclassification rate of the different subtrees appropriately, in this way the tree is "pruned".

Finally, we choose the subtree that provides the best estimate of *R(T)*. The number of possible subtrees can be very high, even if the leaves are relatively few, so selective pruning is necessary, i.e. a method that allows identifying a sequence of subtrees of decreasing size T_{max},T_1,T_2… …{t_1} where {t_1} is the tree consisting only of the root node. Each subtree belonging to the optimal sequence is the best compared to subtrees having the same number of nodes.

In order to identify the optimal sequence, the measurement is defined for each node:

$$R_\alpha(t) = R(t) + \alpha \qquad (4.19)$$

with α real number.

So for every tree T ≤ T_{max}, the following is a cost-complexity function:

$$R_\alpha(t) = \sum_{t \in \tilde{T}} R_\alpha(t) = \ R(T) + \ \alpha\left|\tilde{T}\ \right| \qquad (4.20)$$

In a classification problem *R(T)* is the estimated misclassification rate, $\tilde{T}$ is the cardinality of the set of final nodes $\tilde{T}$ and α is a non-negative real number called the complexity parameter. This parameter can be considered as the penalty connected to large trees, whereby between two trees having the same value of *R(T)* the one with the lowest number of nodes is selected. Once the value of α has been fixed, we search for that subtree T(α) ≤ T_{max} such that:

$$R_\alpha(T(\alpha)) = \ min_{T\ \leq T_{max}} R_\alpha(t) \qquad (4.21)$$

At each split we obtain an estimate of the misclassification rate for each node, *R(T)*, which decreases at each split. In the terminal node α is zero, it increases if we go back towards the root node.

As α increases, the weight of complexity increases to balance the error reduction due to the lesser presence of decision nodes.

If α is small then the penalty associated with having a large number of nodes is small, so T(α) will be large; if α increases, T(α) will have an increasingly smaller number of final nodes until, for a sufficiently high value of α, the ideal subtree will be the one formed by the root alone. When α grows, the tree with R(α)minimum is a smaller subtree than the original tree, therefore, the sequence of resulting subtrees is nested.

Each subtree is connected to a value of the complexity parameter α so we can accept the subtree as the one with "minimum complexity".

Although α belongs to the field of real numbers, the number of subtrees of T_{max} is always finite, so with pruning we obtain a finite sequence of subtrees with a number of final nodes decreasing as α increases. It can be shown (Breiman and others 1984) that for each α there exists a unique minimizing subtree, so the optimal sequence of subtrees $T_{max},T_1,T_2\ldots\ldots\{t_1\}$ is uniquely determined.

The subtrees belonging to the optimal sequence are then compared using an estimate of the misclassification rate, and the subtree chosen will be the one for which this estimate is minimum. The choice of the best subtree is influenced by the *R(T)* estimator used. In fact, if the estimate by resubstitution of *R(T)* is used, the selected subtree will be the most complex one, i.e. T_1. It is therefore necessary to resort to more accurate estimates of the misclassification rate: two other possible ways to estimate *R(T)* are estimation by test set and estimation by cross-validation.

4.1.7 Advantages and disadvantages of the decision tree

Decision trees have the following advantages [34]:

- They are easy to understand and interpret: Compared to other models, it is easier to visualize, explain and apply the results of a decision tree, even to people without a data science background.
- Require little data preparation: Unlike other techniques, decision trees are more resistant to outliers.
- Built-in feature selection: If a feature is not useful, it will not show much in the decision tree model. The hierarchy of a decision tree model reflects the importance of features. The features above are more informative. For less informative features, we can potentially remove them in subsequent runs.
- Generalization ability: they can be stored compactly and are capable of efficiently classifying new observations.

Decision trees have the following disadvantages:

- Relatively unstable: tree structure is less robust than linear/logistic regression. If another sample of data were taken from the same population, the tree could have large differences and even different prediction results. This is due to its hierarchical nature. Small differences in the training dataset can lead to different splits at the top, and these differences affect all child nodes. These differences add up for deeper nodes. This can be alleviated by clustering trees (e.g., random forests or gradient-boosting decision trees), at the sacrifice of being more difficult to interpret.
- More prone to overfitting: Decision tree models are more likely to have overfitting problems. Most of the time, you need to set stopping conditions, prune trees, and use cross-validation techniques to avoid this problem.

4.2 RANDOM FOREST

The Random Forest algorithm is a supervised learning methodology. It is a versatile machine learning method, capable of tackling both classification and regression tasks. It is widely used in some industries:

- Banking: the banking sector is made up of a large number of users, not all of whom are always honest. To determine whether the customer is loyal or fraudulent, with the help of a Random Forest algorithm, it is possible to determine whether the customer is fraudulent or not.

- Medicine: Medicines require a complex combination of specific chemicals. Therefore, to identify the large combination of drugs, random forest can be used. With the help of machine learning algorithm, it has become easier to detect and predict sensitivity to drugs in combination with others. In certain cases it is then easier to identify a patient's disease by analyzing their medical records.

- Stock Market: When you want to know the behavior of the stock market, with the help of this algorithm, you can analyze the trend of stocks, so as to show the expected loss or profit that can be produced while purchasing stocks. a certain title.

- E-commerce.

It is an ensemble model, which uses **bagging** as an ensemble method and the **decision tree** as an individual model [35].
Bagging, or bootstrap aggregation, is a technique used to reduce the variance of an estimated prediction function. It works especially well with high-variance, low-distortion procedures, such as trees. Several estimators are constructed independently and the predictions are averaged taking into account that the combined estimator is often better than any single estimator as it will have low variance.
The idea behind the algorithm is that by training each tree on different samples, although each tree may have high variance compared to a particular set of training data, overall the entire forest will have lower variance but not at the cost of increase distortion. This means that a random forest combines many decision trees into a single model to obtain a more accurate and stable prediction. Individually, the predictions made by the decision trees may not be accurate, but combined together, they will be closer to the result on average.

The final result returned by the Random Forest is nothing more than the average of the numerical result returned by the different trees in the case of a regression problem, or the class returned by the largest number of trees, which is called "committee", in the case of the Random Forest was used to solve a classification problem.

4.2.1 Operation of the algorithm

The random forest consists of a large number of individual decision trees that operate as an ensemble. Every single tree in the random forest outputs a class prediction, and the class with the most votes becomes our model's prediction. The low correlation between trees is the key, since thanks to this effect the trees “protect” each other from their individual errors (as long as they do not all constantly err in the same direction). While some trees may be wrong, many others will be right, so as a group the trees are able to move in the correct direction. Random forest uses two key concepts that give it its name random:

- **Random sampling** of training data points when building trees:

During training, each tree learns from a random sample of data points. Samples are drawn with replacement, known as bootstrapping, which means that some samples will be used multiple times in a single tree. Since decision trees are very sensitive to the data on which they are trained, small changes to the training set can result in significantly different tree structures, so each individual tree samples the data randomly with replacement, thus forming different trees. This process is known as bagging.

With the bagging technique you don't split the training data into smaller blocks and train each tree on a different block, rather, if we have a sample of size N, we are still feeding each tree with a training set of size N. Instead of the original training data, we take a random sample of size N with replacement. For example, if our training data were [1, 2, 3, 4, 5, 6], we could give one of our trees the following list [1, 2, 2, 3, 6, 6]. Note that both lists are of length six, and that "2" and "6" are both repeated in the randomly selected training data we give to our tree (because we sample with replacement).

- **Random subsets** of features considered when splitting nodes:

In a decision tree, when you split a node, every possible characteristic is considered and the one that produces the greatest separation between observations in the left node compared to those in the right node is chosen. In contrast, each tree in a random forest can only choose from a random subset of variables. This forces even more variation between trees in the model and ultimately results in less correlation between trees and more diversification.

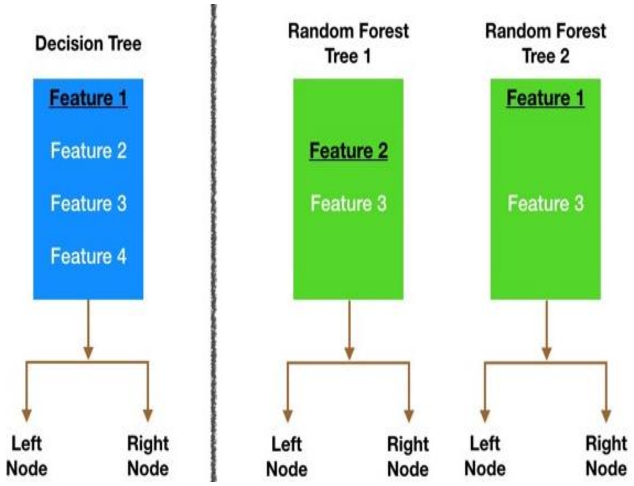


Figure 32: Decision Tree vs Random Forest

Let's take an example: in the image above, the traditional decision tree (in blue) can select between all four features when deciding how to split the node. He decides to use feature 1 (black and underlined) because it divides the data into groups that are as separate as possible.

In the random forest, however, in this example there are only two trees. Random forest tree 1 only considers features 2 and 3 (randomly selected) for its node-splitting decision. We know from our traditional decision tree (in blue) that Feature 1 is the best feature for splitting, but Tree 1 can't see Feature 1, so it's forced to go with Feature 2 (black and underlined). Tree 2, on the other hand, can only see features 1 and 3, so it is able to select feature 1.

So, with random forest, we end up with trees that are not only trained on different datasets (thanks to bagging), but also use different features to make decisions. In practice, random forest combines hundreds or thousands of decision trees, trains each on a slightly different set of observations, splitting the nodes in each tree by considering a limited number of features.

The value of the number of variables taken in each tree is set to √(n_features) for classification, while it is set to (n_features)/3 for regression.

4.2.2 Algorithm parameters

The Random Forest algorithm has three main hyperparameters, which must be set before training. These include node size, number of trees, and number of features sampled:

- n_trees, which is the number of trees the algorithm builds before taking the maximum grade or taking the averages of the predictions. In general, a larger number of trees increases performance and makes more precise predictions stable, but it also slows down the computation.

- mtry, which is the maximum number of features that the random forest considers to split a node.

- min_n, which determines the minimum number of leaves required to split an internal node.

The Random Forest algorithm can be used for both classification and regression problems.
It is made up of a collection of decision trees, and each tree in the set is composed of a sample of data taken from a

training set with replacement, called a bootstrap sample. Of that training sample, one-third is set aside as test data, known as the out-of-bag (oob) sample [36].

Another example of randomness is injected through feature bagging, adding more diversity to the dataset and reducing the correlation between decision trees. Depending on the type of problem, the prediction determination will vary. For a regression task, the individual decision trees will be averaged, and for a classification task, a majority vote, that is, the most frequent categorical variable, will produce the predicted class. Finally, the oob sample is then used for cross-validation.

4.2.3 Phases of the algorithm

The algorithm involves four phases:

1. Random samples are selected from a given data set (bagging).

2. A decision tree is constructed for each data sample with respective randomly chosen variables and a prediction result is obtained from each decision tree.

3. A vote is taken for each predicted outcome.

4. The prediction result with the most votes is selected as the final prediction.

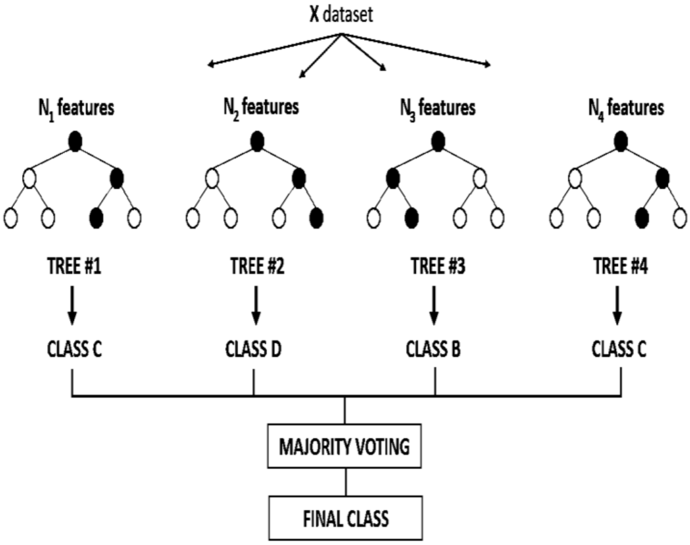


Figure 33: Random Forest

4.2.4 Bagging

Let's analyze the bagging procedure mentioned previously in detail, given a training set :

For b=1…..B:

Example, with replacement, n training examples from call them X_b and Y_b.

Train a classification or regression tree f_b on X_b and Y_b.

After training, you can make predictions for unseen samples x^' by averaging the predictions from all individual regression trees across x':

$$f_{bag} = \frac{1}{B}\sum_{i=1}^{B} f_i\,(x') \tag{4.22}$$

or with the majority of votes in the case of classification trees [21].

This bootstrapping procedure leads to better model performance because it decreases the variance of the model, without increasing the bias. This means that while the predictions of a single tree are highly sensitive to noise in its training set, the average of many trees is not, as long as the trees are uncorrelated. Simply training many trees on a single training set would give highly correlated trees (or even the same tree many times, if the training algorithm is deterministic); bootstrap sampling is a way to decorrelate trees by showing them different training sets. The number of samples/trees, B, is a parameter that can be freely chosen. Typically, a few hundred to several thousand trees are used, depending on the size and nature of the training set. An optimal number of B-trees can be found using cross-validation or by looking at the out-of-bag error i.e. the average prediction error over each training sample bootstrap. When bootstrap aggregation is performed, two independent sets are created. One set, the bootstrap sample, consists of data chosen to be "in the bag" through sampling with replacement. The out-of-bag set consists of all data not chosen in the sampling process. When this process is repeated, for example when creating a random forest, many bootstrap examples and Oob sets are created. Oob sets can be aggregated into one dataset, but each sample is considered out-of-bag only for trees that do not include it in their bootstrap sample. Training and testing error tend to stabilize after a certain number of trees have been adapted.

4.2.5 Importance of characteristics

Through the Random Forest algorithm it is possible to intrinsically calculate the importance of features through the Gini impurity index which is defined as the average Gini decrease in node impurities over all trees in the forest (arises from the fact that the index of Gini impurity for a given parent node is greater than the value of that measure for its two child nodes). Variable importance determines how that variable contributed to reducing a node's impurity during the learning phase and can then be calculated based on a measure of how the node's impurity decreases [37]. Random forests provide built-in support for selecting the right variables as the algorithm tracks how often each descriptor is used by the trees in the forest and how many training data points are affected by the decision within a tree. This information can be compiled into a characteristic number that reflects the importance of a variable. The variable importance is calculated separately for each class, and in addition, the overall importance for all classes is also calculated. The results can be used to purge the descriptor list. Intuitively, such a measure of feature importance should give more weight to a variable that is used near the root while giving less importance to those used at the bottom of trees in a forest. For a classification task, for each variable X_i used in a decision tree forest, you can use the sum of mutual information weighted by the relative sample size of the node where that variable is used. In this way, variables that appear in many "larger" nodes (i.e. closer to the root) will be more important than the others. The importance value is represented by a score, indicated as "Mean Decrease

Gini", which represents the contribution of each characteristic to the homogeneity of the data. The average decrease in the Gini index is highest for the most important characteristic, which will be used first during division. The overall average decrease in Gini importance for each feature is then calculated as the ratio of the sum of the number of splits in all trees that include the feature to the number of samples it splits.

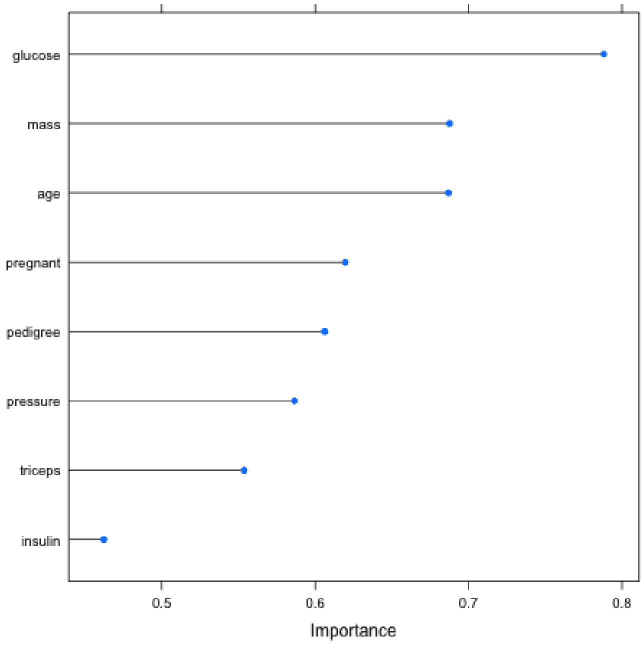


Figure 34: Importance of features

In the example above, you can see how the most important variables for predicting diabetes are glucose, body mass index and age.

4.2.6 Advantages and disadvantages Random Forest

The algorithm has the following advantages [38]:

- It is one of the most accurate learning algorithms available. For many datasets, it produces a highly accurate classifier.
- Works efficiently on large databases.
- Can handle thousands of input variables without deleting variables.
- Provides estimates of which variables are important in classification.
- It has an effective method to estimate missing data and maintains accuracy when a large portion of data is missing.
- Requires less training time than other algorithms.
- Reduced risk of overfitting: Decision trees run the risk of overfitting as they tend to tightly fit all samples within the training data. However, when

there are a large number of decision trees in a random forest, the classifier does not fit the model because averaging uncorrelated trees reduces the overall variance and prediction error.

The algorithm has the following disadvantages:

- Random forests have been observed to overfit some datasets with noisy classification/regression tasks.
- Although random forest can be used for both classification and regression tasks, it is no longer suitable for regression tasks.
- For data that includes categorical variables with different numbers of levels, random forests are biased in favor of those attributes with more levels. Therefore, variable importance scores from random forest are not reliable for this type of data.

4.3 DIFFERENCE BETWEEN DECISION TREE AND RANDOM FOREST

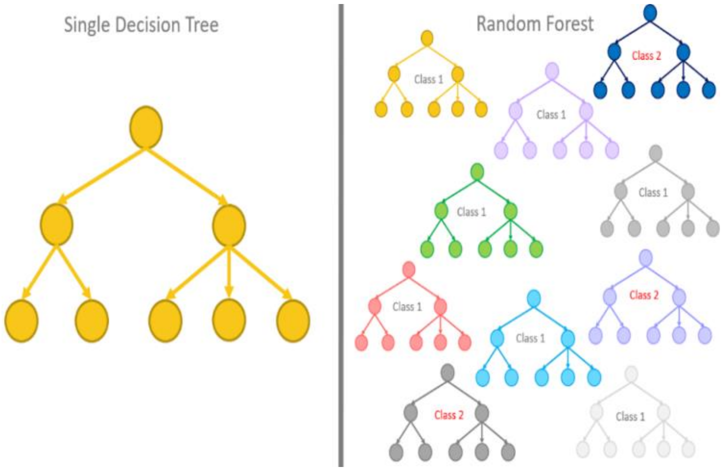


Figure 35: Single tree vs groups of trees

Although random forest is a collection of decision trees, there are some differences. If you feed a training dataset with features and labels into a decision tree, a set of rules will be formulated, which will be used to make predictions. In the random forest algorithm, however, observations and characteristics are randomly selected to build different decision trees and the results are then averaged. Furthermore, decision trees with greater depths are more prone to overfitting and therefore result in greater variance in the model. This single tree gap is explored by the Random Forest model since the original training data are random samples obtained by the replacement method. These samples are also known as bootstrap samples. Each of these trees is trained separately on these bootstrap examples, and the final result of the ensemble model is determined by counting a majority vote from all decision trees. This concept is known as Bagging or Bootstrap Aggregation. The bagging concept reduces variance without changing the bias of the complete ensemble.

4.4 NAIVE BAYES

The Naive Bayes classifier is a probabilistic supervised learning algorithm that is based on Bayes' theorem from which it takes its name. In machine learning we are often interested in choosing, whatever it is, the best hypothesis (C) from the given data (X). In a classification problem, our hypothesis (C) could be the class to assign for a new data instance (X) that we want to predict. The Naive Bayes classifier finds various applications such as:

- Real-time prediction: It is a fast learning classifier. Therefore, it could be used to make real-time predictions, for example, in weather forecasting, where there is still room for improvement and artificial intelligence will certainly make a strong contribution in this sense.

- Text classification, spam filtering and sentiment analysis: Naive Bayes classifiers are often used in text classification, spam filtering (in order to identify junk mail) and sentiment analysis (in social analytics media, to identify customers' positive and negative feelings).

- Recommendation system: Naive Bayes classifier and collaborative filtering together build a recommendation system that uses machine learning and data mining techniques to filter invisible information and predict whether or not a user would like a certain resource.

Bayes' Theorem is then used to determine the preferable hypothesis with the data already available. This theorem provides a way to calculate the probability of a hypothesis based on prior knowledge.

A naive Bayesian model is easy to build, without complicated iterative parameter estimations making it particularly useful for very large datasets. Despite its simplicity, it often works very well and is widely used because it often outperforms more sophisticated classification methods. In statistics and probability theory, Bayes' theorem describes the probability of an event, based on prior knowledge of conditions that might be related to the event. It is used to understand conditional probability.

The theorem essentially allows a hypothesis to be updated every time new evidence is introduced. The equation that represents it is the following:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \tag{4.23}$$

Where: P indicates probability

P(C|X) is the probability of event C (hypothesis) occurring given that X (evidence) occurred. It is also defined as **posterior probability**, with reference to the experiment in which event.

P(X|C) is the probability that event X (evidence) occurs given that C (hypothesis) has occurred. It is also defined as likelihood or likelihood with reference to what would likely result if C were true.

P(X) is the probability of event X occurring. Obviously it is of fundamental importance that this event occurs, otherwise we would be dividing by 0.

P(C) is the probability of event C occurring. It is also called "a priori probability" because it is the knowledge we have of the value of C before looking at the observables X, in fact it is not conditioned on any other event [39].

The transition from a priori to a posteriori probability constitutes the updating of the evaluation in light of the further information constituted by the fact that event C has occurred. Using Bayes' theorem, we can find the probability of C happening, given that X has occurred, where X is the evidence and C is the hypothesis. The assumption made here is that the predictors/characteristics are independent. That is, the presence or absence of a particular characteristic does not affect the other. So he is called naive for this reason. The feature independence approach is used to “separate” multiple tests and treat them as independent.

Therefore, the probability

$$P(X_1, X_2, X_3 \dots \dots X_n | C)$$

can be translated into:

$$P(X_1|C) * P(X_2|C) * P(X_3|C) \dots\dots * \ P(X_n|C)$$

that is to say:

$$P(X_1, X_2, X_3 \dots\dots X_n|C) = \textstyle\prod_n P(X|C) \qquad (4.24)$$

Therefore:

$$P(C|X) = \frac{\prod_n P(X|C)\ P(C)}{P(X)} \qquad (4.25)$$

Furthermore, since the denominator does not depend on the category, the classification is conducted by maximizing the numerator, with the hypothesis stated previously. The class with the highest posterior probability is the result of the prediction.

Naive Bayes is often described using categorical data because it is easy to describe and calculate using ratios. According to this approach, it is sufficient to calculate probabilities. Categorical variables take values that are names or labels. The breed of a dog (e.g., collie, shepherd, terrier) or the color of a ball are examples of categorical variables. On the contrary, quantitative variables are numerical and can be divided into discrete and continuous. Discrete quantitative variables represent a measurable quantity. For example, when we talk about the population of a city, we mean the number of people in the city (a measurable attribute of the city). The population is, in this case, a discrete quantitative variable. Instead, continuous quantitative variables are not finite numbers, but real numbers such as the weight or height of a man. A more useful version of the algorithm supports continuous numerical data and assumes that the values of each feature are normally distributed (i.e., fall somewhere on a bell curve). In this case, Naive Bayes can be extended to real-valued attributes, most commonly assuming a Gaussian or normal distribution. So when the predictors take on a continuous value and are not discrete, it is assumed that these values are sampled from a Gaussian distribution:

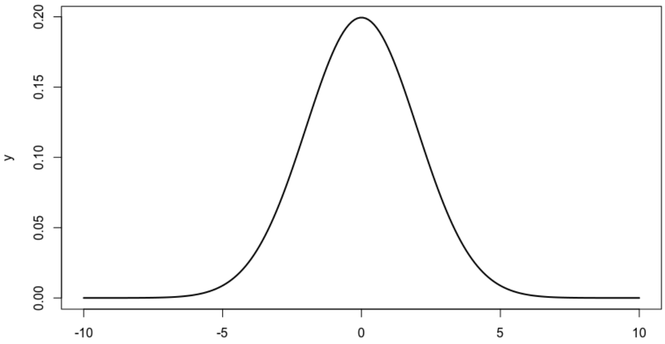


Figure 36: Gaussian distribution

$$f(x\,|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \qquad (4.26)$$

According to this assumption it is sufficient to find the mean and standard deviation of each probability for each attribute and for each individual class. They are calculated as follows:

$$\bar{\mu} = \frac{1}{n}\sum\nolimits_{i=1}^{n} x_i \qquad (4.27)$$

$$\hat{\sigma} = \sqrt{\frac{\sum_{i=1}^{n}(\,x_i - \bar{\mu})^2}{n-1}} \qquad (4.28)$$

This extension of the model is called Gaussian Naive Bayes. Other functions can be used to estimate the distribution of the data, but the Gaussian (or normal distribution) is the simplest to use since you only need to estimate the mean and standard deviation from the training data.

By substituting these values into the Gaussian probability density function (also called Gaussian Probability Density Function) a probability is obtained which allows the various class probabilities to be obtained. The highest class probability value thus obtained represents the class to be associated with the new instance that you want to categorize.

The algorithm follows the following steps:

- Calculating class probability: Class probabilities are simply the frequencies of instances that belong to each class divided by the total number of instances.
- Calculation of conditional probability: Bayes' theorem is applied to determine the conditional probabilities of the characteristics of the problem.
- Make a decision: the probability is calculated to predict the class to which the new instance belongs, respecting the verification of the independence of the characteristics. The final decision is identified in the class that obtains the highest probability value.

4.4.1 Advantages and Disadvantages Naive Bayes

The algorithm has the following advantages:

- Ease of use.
- Robust at isolated noise points.
- Classifiers handle missing data by not considering the event during calculations.
- Despite being a simple and dated algorithm, it still solves some classification problems very well today with reasonable efficiency.
- Works well in multi-class predictions and in case of categorical input variables. If you analyze continuous numerical variables it is possible, as seen, to use the normal distribution to predict new instances and make further hypotheses. Tends not to consider irrelevant attributes.
- The training of the model is much simpler than other algorithms, it is based exclusively on the theorems of probabilistic calculation and does not involve an iterative learning phase with gradient descent, but rather the simple construction of a table relating to the conditional probability starting from the training sets [40].

The algorithm has the following disadvantages:

- Requires knowledge of all problem data. Especially simple and conditional probabilities.
- The algorithm provides a "naïve" approximation of the problem because it does not consider the correlation between the characteristics of the instance. In real life, it is almost impossible to obtain a completely independent set of predictors.
- If the categorical variable is not observed in the training dataset, the model will assign a probability of zero and will not be able to make a prediction. This event is often known as zero frequency, and smoothing techniques such as Laplace estimation can still be used to resolve it [40].

CHAPTER 5

EXPERIMENTAL ANALYSIS AND RESULTS

5.1 DATA ANALYSIS

In this thesis work, supervised learning methodologies were compared for the classification of the risk of progression of bladder cancer. In particular, three supervised learning algorithms were applied and the accuracy results were compared both considering the data set composed of all the variables and the data set containing a smaller number of variables through the use of a dimensionality reduction technique.

The analysis carried out was carried out with the R software and its RStudio interface. It is an open source software and programming language for statistical computing and graphics. Data refers to a sample of 111 patients and both qualitative and quantitative variables were considered as input:

- Sex
- Body mass index
- Smoke

- Familiarity
- Age
- Muscle invasiveness of the tumor
- Width of the bladder wall
- Length of the bladder wall
- Number of lymph nodes
- Lymph node width max
- Lymph node length max
- Number of liver lesions
- Width of liver lesion max
- Length of liver lesion max
- Number of lung lesions
- Width of the lung lesion max
- Length of lung lesion max
- Number of bone lesions

The variables listed above were used to predict the risk of bladder cancer progression, which is divided into four classes from 1 to 4:

- Low (risk 1)
- Medium-low (risk 2)
- Medium-high (risk 3)
- High (risk 4)

The qualitative variables were transformed into dummy variables with the coding "one hot", then the variables with the wording "yes" and "no" were transformed into numerical variables with values 0 and 1, in such a way as to make the sample as there are also numerical variables. In total, therefore, the number of predictors is equal to 22.

For data analysis, an additional "age classes" column was created to have a representation of the sample under examination. 4 age classes were created, respectively 40-49, 50-59, 60-69, 70-80.

From the following graph you can see how there is no statistical correlation between age classes and risk of tumor progression, in fact any age can present any type of risk.

In particular, it is highlighted that the sample in question is made up of 60% risk 4 data, therefore there is a polarization.

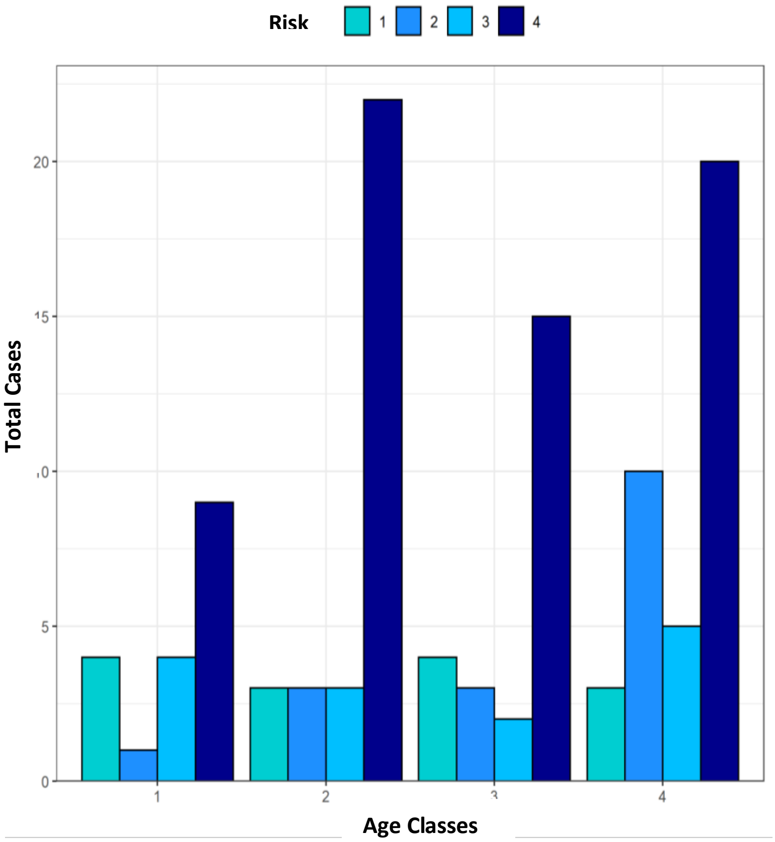


Figure 37: Risk and age classes

Before applying the algorithms, the data were appropriately normalized due to the presence of categorical variables and numerical variables.

Subsequently, the data was divided into train and test, in particular 81 observations were used for the train set, to train the model, and 30 observations for the test set, for verification.

After an appropriate study on classification algorithms, the following were applied: Random Forest, Decision Tree and Naïve Bayes.

The same train and test splits were used for the 3 different algorithms in order to make the comparison consistent.

During the training phase of a model, the goal is to achieve a level of generalization that allows the model to perform well on new and unseen data.

K-fold cross validation was applied to make the predictions more reliable and less biased.

Each algorithm accepted for Naïve Bayes has parameters to be optimized, therefore for parameter tuning the k-fold was used, with k = 5, repeated 3 times, stratified on the output variable, and the values of the parameters that return a higher average accuracy.

The number of input variables is equal to 22, therefore, given the modest data set and a high number of variables, a dimensionality reduction technique was used to compare the results of the accuracy of the algorithms considering a lower number of variables provided as input.

The technique that is most used for this type of data, since it is also poorly correlated with each other, is Feature Selection. It is a method intrinsic to the RF algorithm and uses the Gini index as a verification of the most important variables, which represents a measure of impurity.

Each algorithm was applied both on the data set composed of all the variables and on the one composed of a smaller number of variables, in such a way as to highlight the advantage of providing a smaller number of variables as input at the "cost" of having a small loss of precision.

The algorithms were appropriately compared with each other in order to verify which one returns a greater accuracy value and therefore a more reliable classification.

5.2 FEATURE SELECTION

Given the large number of variables, the correlation matrix was analyzed in order to subsequently choose the most appropriate technique to reduce the number of variables.

The figure below shows the correlation matrix of the variables. Blue circles indicate a positive correlation, while red circles indicate a negative correlation. The darker the colors the greater the correlation.

The data set in question is composed of variables that do not present such a high level of correlation, except for a few, therefore the dimensionality reduction technique is Feature Selection.

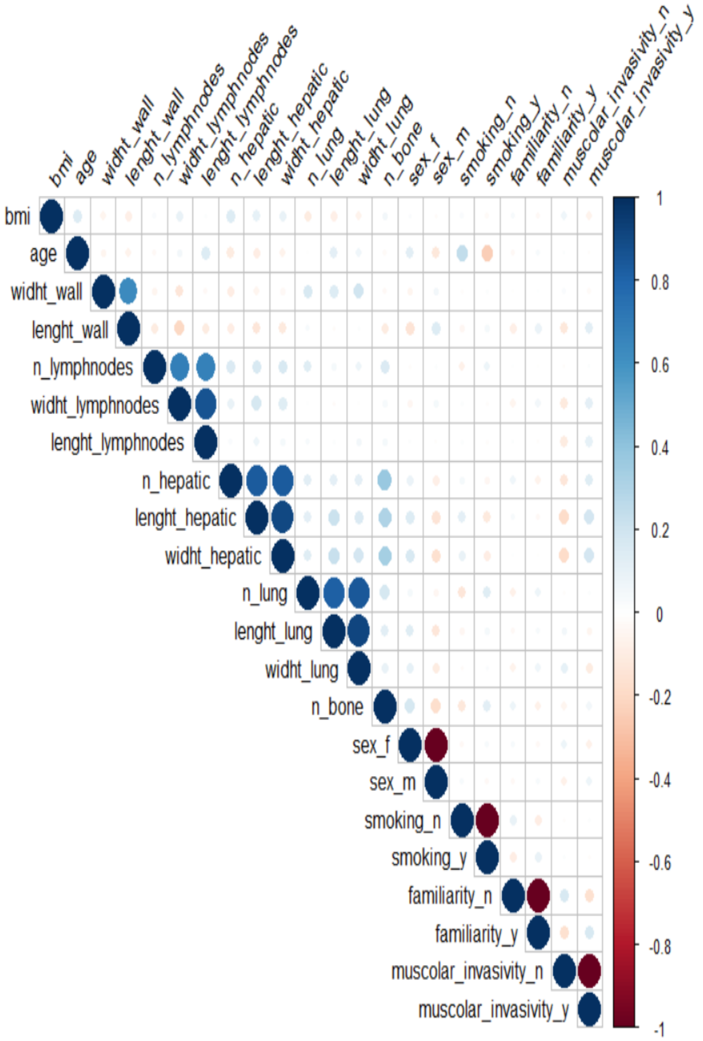


Figure 38: Correlations of variables

Among the different Feature Selection techniques, as mentioned in chapter 3, the one intrinsic to the Random

Forest algorithm which uses the Gini index for the importance of the variables was selected. In the following figures, the value of the Gini index of the input variables for risk prediction has been ordered in decreasing order.

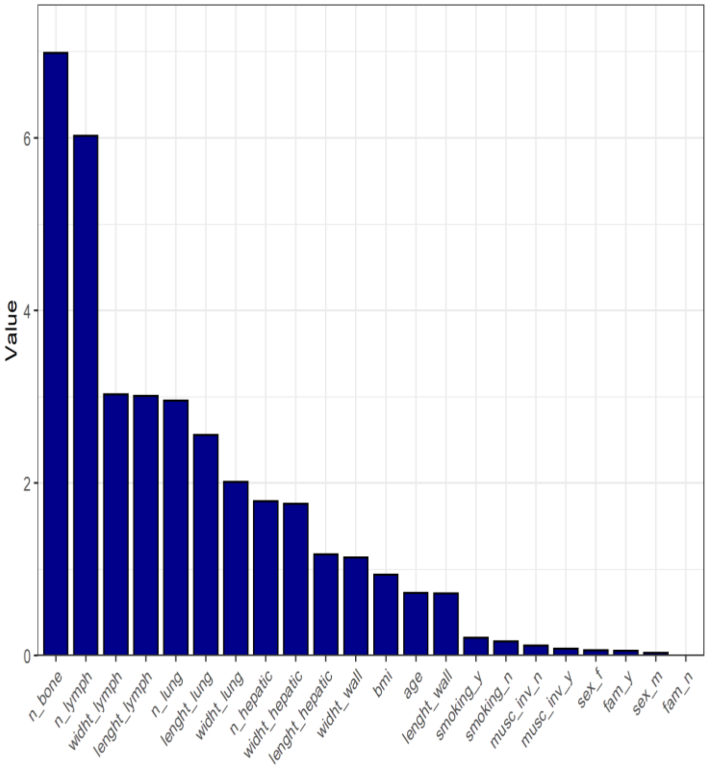


Figure 39: Gini

From the analysis of the results it can be seen that the most significant variable is the number of bone lesions, followed by the number of lymph nodes. Categorical variables, on the other hand, are the least significant for predicting cancer risk.

Since there is no threshold value to select the number of optimal variables, we proceeded by trial and error taking into account the accuracies obtained in the train and test set predictions.

Assuming a number of 300 trees, predictions of the train set and the test set were made taking into account 5,10,15 variables. The results obtained are the following:

Table 7: Feature selection

RISK	Train	Test
5 variables	93,01 %	100 %
10 variables	94,64 %	100 %
15 variables	91,32 %	100 %

At parity or with a minimum loss of accuracy, it is better to take into consideration a smaller number of variables since it is necessary to find a trade-off between the simplicity of the problem and computational effort.

The top 5 variables were selected for prediction.

5.3 APPLICATION OF ALGORITHMS

Three supervised methodologies were applied: Decision Tree, Random Forest and Naive Bayes.

The algorithms were compared with each other taking into consideration 2 data sets of different variables:

- The one containing all the variables.
- The one containing the variables selected with Feature Selection.

The first necessary thing to do is to divide the data set into train sets and test sets. 81 data were selected in the train set, therefore for the model training phase, 30 data for the test data set, therefore for verification.

The same splits were used for the 3 different algorithms, in order to make the comparison consistent.

K-fold cross validation was applied to make the predictions more reliable and less biased. In particular, the k fold repeated three times was applied, with k = 5 and stratified on the output.

The package used for the application is tidymodels.

5.3.1 Decision Tree

The first algorithm that was used is Decision Tree.

The three optimized parameters are:

- **tree depth**: the maximum depth of a tree.
- **min_n**: The minimum number of data points in a node needed to divide it further.
- **cost complexity**: the parameter α which is the penalty connected to large trees.

Tuning of these three parameters occurred through three-fold and stratified k-fold cross-validation. The parameters that give a better level of accuracy have been selected. Parameter tuning was carried out for the two data sets taken into consideration. The following graph shows the three parameters optimized as a function of the average accuracy obtained from the cross-validation.

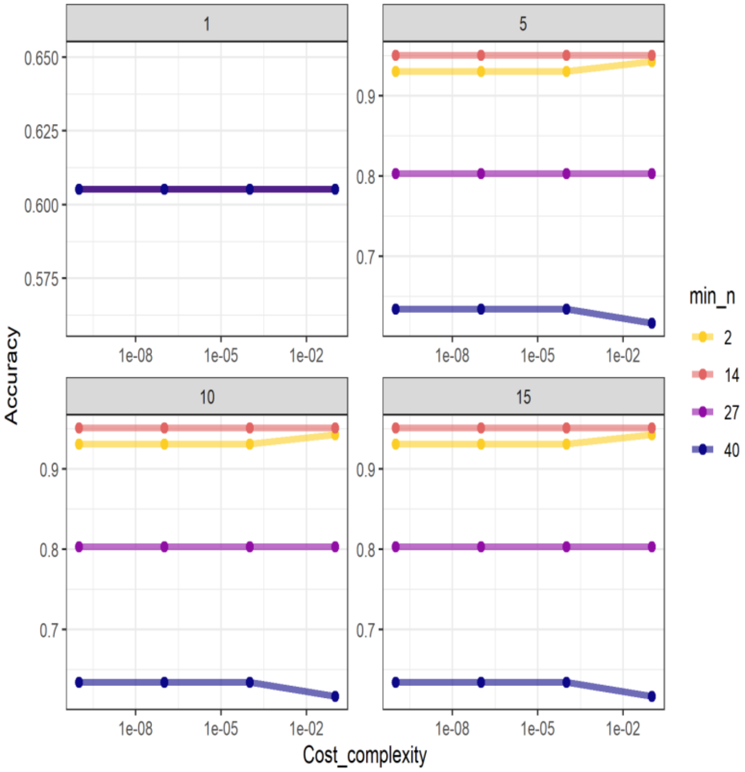


Figure 40: Risk Accuracy

The selected parameters are those that give the best accuracy. The tuning of the parameters is also shown below for the data set composed of a smaller number of variables by carrying out the same procedure defined previously.

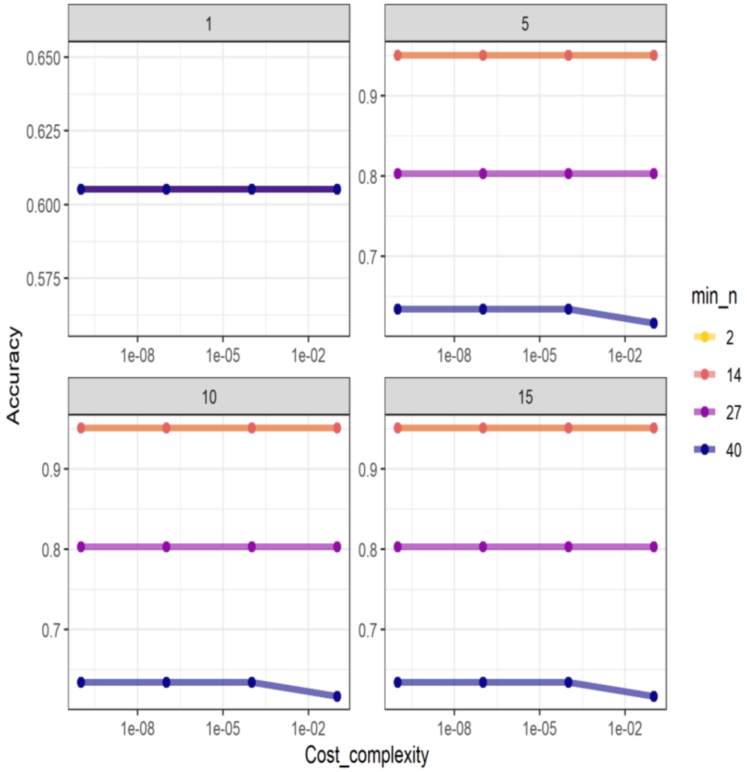


Figure 41: Accuracy FS risk

The following tables summarize the parameters selected during tuning, therefore those that determined the best level of average accuracy.

Table 8: Tuning risk parameters (Decision Tree)

RISK	***cost complexity***	***tree depth***	***min_n***
All variables	10^{-10}	5	14
Feature selection	10^{-10}	5	14

The results obtained are the following:

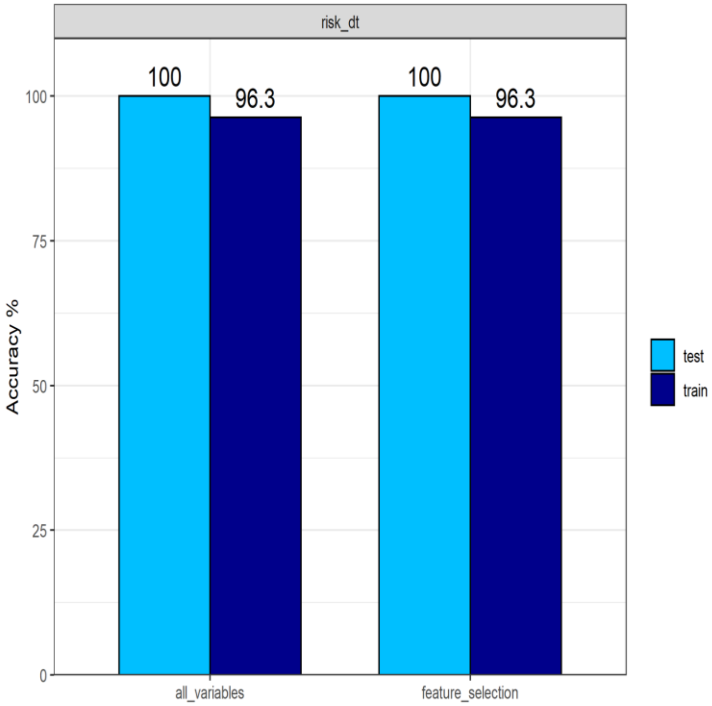


Figure 42: Accuracy Decision Tree

From the analysis of the results it can be seen that the forecast is satisfactory. The accuracy result is the same for both data sets, this implies that the 5 variables selected in the reduced data set perfectly approximate the initial model. There are no losses in accuracy, but a notable advantage in computational complexity since the initial 22 variables have been reduced to 5.

5.3.2 Random Forest

A further algorithm that has been applied is Random Forest, which consists of the aggregation of multiple decision trees.

The parameters that have been optimized are:

- **n_tree**: the number of decision trees.
- **mtry**: The maximum number of features that the random forest considers to split a node.
- **min_n**: The minimum number of data points in a node needed to divide it further.

The number of trees increases the complexity of the model, it is possible to evaluate that from a certain point onwards the prediction error of random forests remains constant, so the lowest number of trees is chosen.

Once the number of trees was fixed, 300 for the data set composed of all the variables and 100 for the one with a lower number of variables, two parameters were optimized: mtry and min_n. The tuning was carried out through the cross validation method, in particular using the k-fold with k = 5, repeated 3 times and with stratification. A double tuning of the parameters was carried out to improve the accuracy of the model: from the first tuning for the prediction of both outputs, 20 models were considered which are given by the combination of the two parameters as a function of accuracy. The two parameters have different ranges since the minimum number of points necessary to further divide the node depends on the number of data in possession, while mtry depends on the number of variables within the data set. Subsequently, further tuning was carried out by decreasing the range of possible values of the two variables in such a way as to consider a greater number of combinations, since previously 20 models were considered based on all the possible values of the two variables. The optimal values chosen are those that give a better level of accuracy.

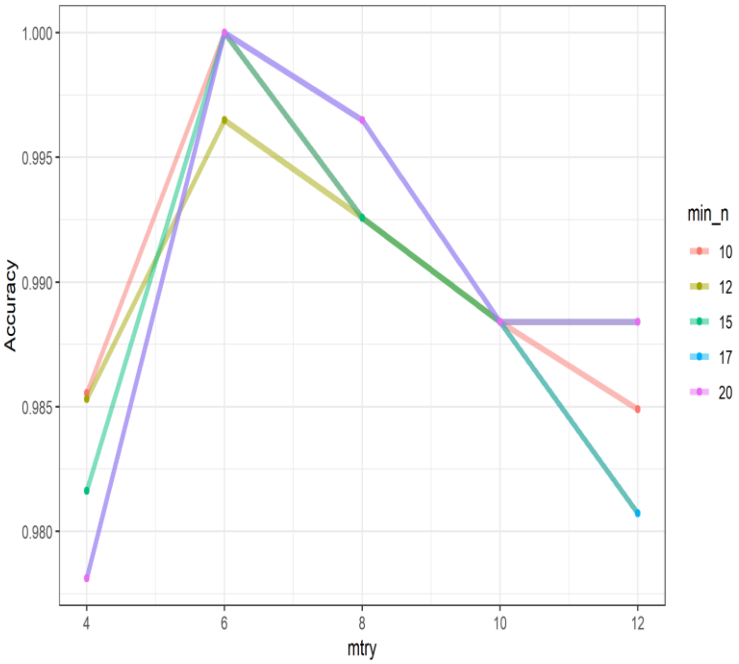


Figure 43: Accuracy risk

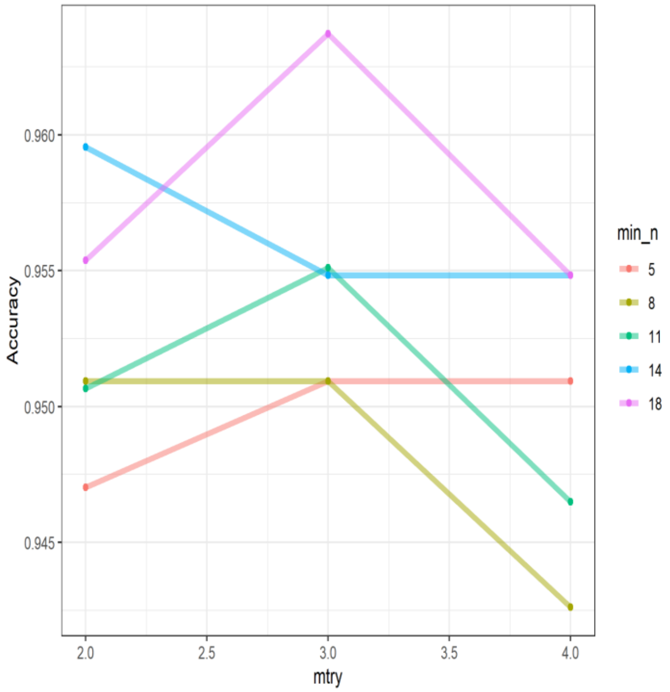


Figure 44: Accuracy FS stage

Table 9: Tuning of risk parameters (Random Forest)

RISK	***n_tree***	***mtry***	***min_n***
All variables	300	6	17
Feature selection	100	3	18

The following results were obtained:

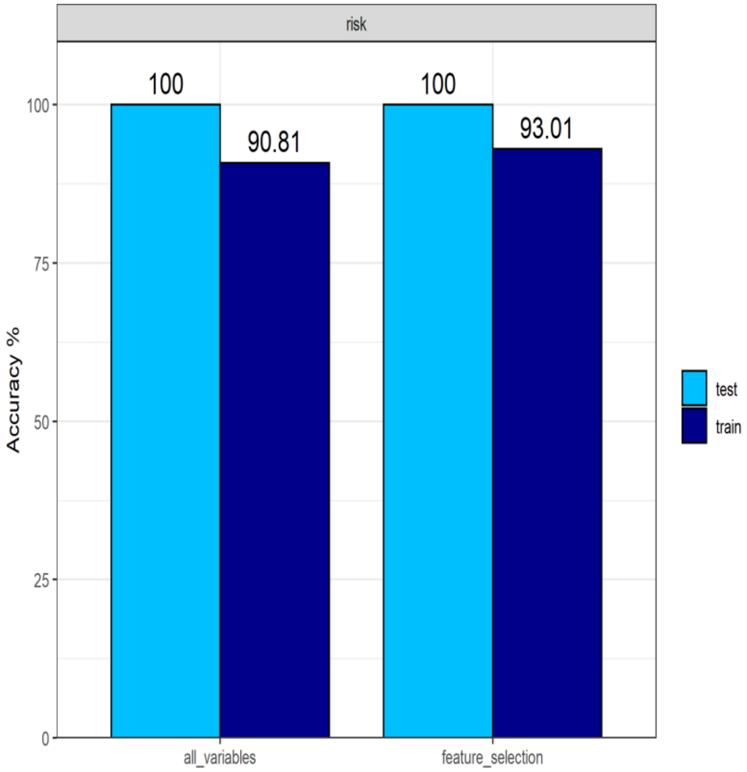


Figure 45: Accuracy Random Forest

From the analysis carried out, it can be seen that the model has a greater error on the train set than that of the test set. However, the results are satisfactory since there is a low error rate in the train set and zero in the test set.

The Feature Selection technique is valid because by reducing the number of variables to 5 there is even an increase in accuracy for the train set. The complexity of the algorithm is reduced and performance increases, therefore the variables selected through the Gini indices are adequate.

5.3.3 Naive Bayes

The last algorithm used is Naive Bayes, named after the theorem from which it takes its name. The Naive Bayes classifier is a probabilistic classifier and assigns each observation to the class that returns the highest posterior probability value. The application of this algorithm is very simple since it does not involve any tuning of the parameters, but only the calculation of the marginal and conditional probabilities. The results obtained on the train set and on the test set are the following:

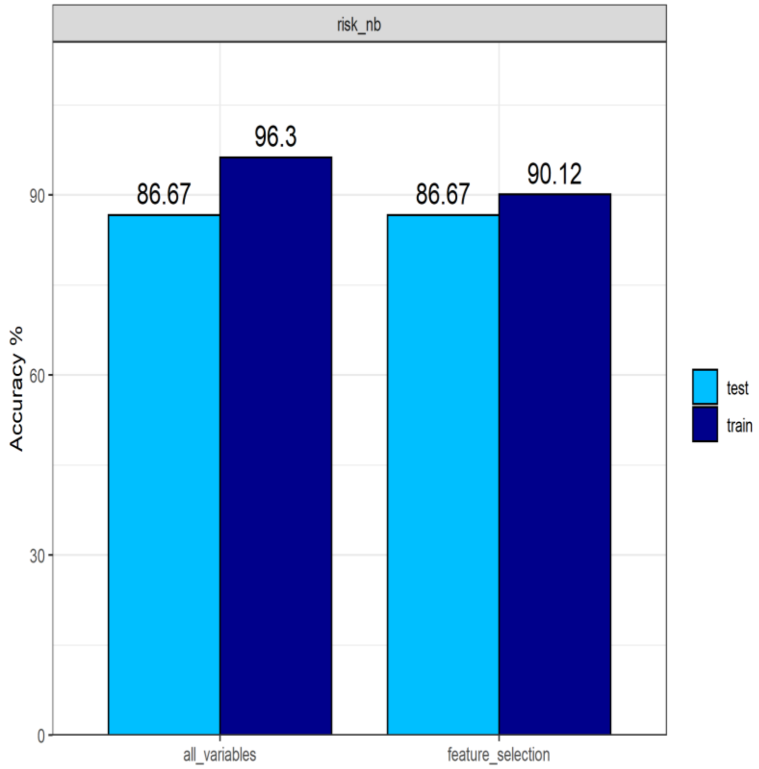


Figure 46: Accuracy Naive Bayes

The prediction of the risk of progression of bladder cancer presents high values for the data set composed of all the variables, while lower values for the other data set, but still satisfactory. Considering the data set composed of a significantly lower number of variables, there is a small loss of accuracy in the train set, while nothing in the test set, therefore the feature selection technique optimally approximates the initial data set.

CHAPTER 6

CONCLUSIONS AND FUTURE DEVELOPMENTS

6.1 CONCLUSIONS

The supervised algorithms that have been used return important results which are summarized in the following figure. In conclusion, within this work a dimensionality reduction technique was applied which was compared with the data set containing all the variables and three supervised algorithms were applied.

The objective was to give the medical oncologist objective support in the diagnosis of disease progression, a diagnosis that is not always easy to make, especially in the case of minimal and/or subtle progression (oligoprogression) of the disease. Carrying out a diagnosis of bladder cancer progression in good time is of fundamental importance for the medical oncologist and for the patient's life as it is thus possible to avoid continuing to administer a drug to the patient (often very expensive, on the order of thousands of euros every 2 weeks) which is exhausting its effectiveness and which perhaps brings more side effects to the patient than benefits, as well as avoiding a further decline in the patient's clinical conditions which perhaps will not allow, in the short term, to subject him to a new line of therapy (treatment), whether chemotherapy or radiotherapy. And since the lines of therapy are not infinite (for bladder cancer today there are 4/5 those universally recognized by the scientific community) and do not last indefinitely (on average from 4 to 8 months each), it is necessary that the oncologist understand best when a treatment is no longer effective and therefore when it is necessary to change it. This is why having objective support for the diagnosis of disease progression can be of great use to the medical oncologist in cases of not so clear progression of the tumor (an eventuality which often represents the majority of cases).

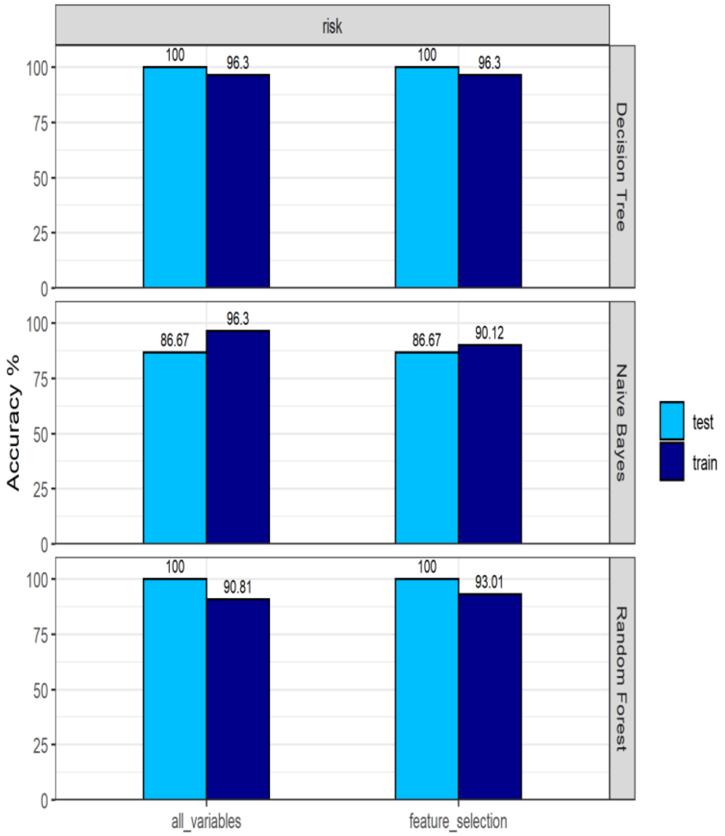


Figure 47: Comparison of algorithms

From the analysis of the results it can be seen that the predictions for the risk of progression of bladder cancer are satisfactory for all three algorithms. Analyzing in detail, we can see how the Decision Tree algorithm gives similar results both considering a number of variables equal to 22 and a number of variables equal to 5. This implies a notable advantage regarding the computational effort of the algorithm.

The Random Forest algorithm presents slightly inferior results compared to the Decision Tree. The results are satisfactory in both cases and considering the data set composed of a smaller number of variables there is an improvement in accuracy in the train set.

The last algorithm applied, namely Naive Bayes, presents good results, but inferior to the other two algorithms. Also in this case the dimensionality reduction technique is valid.

Comparing the predictions obtained, the dimensionality reduction technique is reliable, the loss of accuracy is balanced by the advantage of having a lower number of predictors and this represents an advantage for the computational effort of the algorithm. In one algorithm there was no loss, while in another the prediction improved.

In conclusion, it can be seen that the prediction of the risk of progression of bladder cancer is satisfactory taking into account the modest data set. From the analysis carried out, the Decision Tree algorithm is the one that gives better results than the other two models taken into consideration. Probably the simplest model to apply is Naive Bayes since it is not necessary to tune the parameters, but rather it is necessary to calculate only the marginal and conditional probabilities, so the time required to run the algorithm is the lowest of all.

The most complex algorithm is therefore Random Forest since it is composed of an aggregation of trees, while the Decision Tree is made up of a single tree.

In conclusion, it can be stated that the three algorithms have provided reliable results for predicting the risk of progression and this constitutes an important basis for future applications.

6.2 FUTURE DEVELOPMENTS

Machine learning will be an important support for medicine in the future. Previous experience and data from millions of patients can help doctors make faster and more accurate diagnoses in such a way as to facilitate the understanding of the etiopathogenetic mechanisms underlying diseases and to predict the risk of onset and/or progression of a pathology in time useful for perhaps preventing it or better treating it through the use of the most suitable therapy for that individual patient, with his/her medical history and comorbidities, with his/her concomitant medications and his/her clinical and/or laboratory alterations.

For this reason it is easy to hypothesize how various and interesting the developments that could be obtained through:

- the addition of new variables, such as comorbidities, any genetic mutations (which cannot be interpreted with the naked eye), instrumental and laboratory data, etc.
- the use of a neural network for classification.
- the use of a neural network for the extraction of features for the analysis of radiological images, in order to provide objective support to the doctor where a critical decision needs to be made.

The advantage of using machine learning models is due to the fact that we can work with a huge amount of data, which our brain would not be able to imagine and without incurring human errors due to inattention or tiredness. Having said this, obviously the human doctor-patient relationship is essential and the evolution of medicine towards the digital world can only provide an improvement in the process of diagnosis, treatment and patient care, reducing human errors, lowering mortality rates and costs in healthcare, but it will never be able to totally replace the critical thinking of the doctor's human mind.

Bibliography

[1] AIOM-AIRTUM-Fondazione-AIOM-PASSI. I numeri del cancro in Italia 2018. Intermedia Editore. Settembre 2018

[2] Antoni S, Ferlay J, Soerjomataram I et al. Bladder cancer incidence and mortality: a global overview and recent trends. Eur Urol 2017 Jan

[3] AIRTUM Working Group. La sopravvivenza dei pazienti oncologici in Italia. Epidemiologia e Prevenzione. Suppl. 1 n.2. Marzo-aprile 2017

[4] Negri E, La Vecchia C. Epidemiology and prevention of bladder cancer. Eur J Cancer Prev 2001

[5] Shirodkar SP, Lokeshwar VB. Bladder tumor markers: from hematuria to molecular diagnostics: where do we stand? Expert Rev Anticancer Ther. 2008 Jul;8(7): 1111-23

[6] Yafi FA, Brimo F, Steingberg J, Aprikian AG, Tanguay S, Kassouf W. Prospective analysis of sensitivity and specificity of urinary cytology and other urinary biomarkers for bladder cancer. Urol Oncol 2015

[7] Lokershwar VB, Habuchi T, Grossman HB, et al. Bladder tumor markers beyond cytology specimens: the role of volume and repeat void upon predictive values for high-grade urothelial carcinoma. Cancer Cytopathology 2016

[8] Pavlica P., Gaudiano C, Barozzi L. Sonography of the bladder World J Urol 2004

[9] AIOM Associazione Italiana Oncologia Medica – Linee guida Tumori dell'Urotelio – Edizione 2021 https://www.iss.it/documents/20126/8403839/LG-459-AIOM_Urotelio

[10] Sobin DH, Wittekind Ch, eds. In: TNM Classification of Malignant Tumours. 6th ed. New York: Wiley Liss, 2002

[11] The 2016, WHO Classification of Tumours of the Urinary System and Male Genital Organs. Edited by J. Eble et al. IARC Lyon 2004

[12] The 2004, WHO Classification of Tumours of the Urinary System and Male Genital Organs. Edited by J. Eble et al. IARC Lyon 2004

[13] Steinberg RL, Thomas LJ, O'Donnel MA. Bacillus Calmette-Guèrin (BCG) Treatment Failures in Non Muscle Invasive Bladder Cancer: What Truly Constitutes Unresponsive Disease. Bladder Cancer. 2015

[14] AIOM. Linee guida carcinoma della vescica. Edizione 2015

[15] Fradet Y, Aprikian A, Dranitsaris G, Siemens R, Tsihlias J, Fleshner N. Does prolonging the time to bladder cancer surgery affect long-term cancer control: a systematic review of the literature. Can J Urol 2006

[16] Tom M. Mitchell, «The Discipline of Machine Learning», July 2006.
https://www.cs.cmu.edu/~tom/pubs/MachineLearning.pdf

[17] Christopher M. Bishop, «Pattern recognition and Machine Learning,» 2006, Springer

[18] Friedman JH. , «Data Mining and Statistics: What's the connection?», Computing Science and Statistics, 1998.
https://www.researchgate.net/profile/Muhammad-Iqbal-116/publication/337324537_Data_Mining_and_Statistics_What's_the_Connection/links/5dd2607a299bf1b74b4b790c/Data-Mining-and-Statistics-Whats-the-Connection.pdf

[19] T. Hastie, R. Tibshirani e J. H. Friedman, The elements of statistical learning: data mining, inference and prediction, 2018, second edition Springer.
https://hastie.su.domains/Papers/ESLII.pdf

[20] Adam Geitgey, Machine learning is fun, May 5, 2014
https://medium.com/@ageitgey/machine-learning-is-fun-80ea3ec3c471

[21] G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction To Statistical Learning, Springer, 2013

[22] Hirota K., Pedrycz W., Fuzzy computing for data mining, IEEE, pages1575 – 1600, (Sept 1999) doi 10.1109/5.784240

[23] I. Jolliffe. Principal Component Analysis Springer, 2002

[24] Kiran Parte «Understanding the entirety of the PCA algorithm in bits and pieces», Analytics Vidhya, 2020.

[25] Witten D., James G., Hastie T., Tibshirani R., “An introduction to statistical learning‟, Springer, 2013

[26] Chandrashekar G., Sahin F., A survey on feature selection methods, (2014) 40(1) 16-28;
https://doi.org/10.1016/j.compeleceng.2013.11.024

[27] I. Guyon , A. Elisseeff, Un'introduzione alla selezione di variabili e funzioni J Mach Res, 2003

[28] E. Alpaydin, Introduction to Machine Learning, The MIT Press (2004)

[29] MH Law , M. rio AT Figueiredo , AK Jain Selezione simultanea delle caratteristiche e raggruppamento utilizzando modelli misti , IEEE Trans Pattern Anal Mach Intell , 26 (2004)

[30] G. John, R. Kohavi, and K. Pfleger. “Irrelevant Features and the Subset Selection Problem,” In Proceedings of the Eleventh International Conference on Machine Learning, 1994

[31] Mantovani R., Horvart T., Cerri R., An empirical study on hyperparameter tuning of decision trees, 2019 https://doi.org/10.48550/arXiv.1812.02207

[32] Song Y., Lu Y., Decision tree methods: applications for classification and prediction. (2015) 27(2) 130-135 doi: 10.11919/j.issn.1002-0829.215044

[33] Rajeev Rastogi and Kyuseok Shim, PUBLIC: A Decision Tree Classifier that Integrates Building and Pruning. Proccedings of the 24th VLDB Conference, New York, USA,. 1998.

[34] Tom Michael Mitchell, Machine Learning – Chapter 3, Decision Tree Learning. McGraw-Hill Education, 1997.

[35] Breiman, L. (1996a). Bagging predictors. Machine Learning 26(2), 123–140

[36] Witten D., James G., Hastie T., Tibshirani R., "An introduction to statistical learning‟, Springer, 2013

[37] Breiman L., Random forests. Machine Learning 45, pages5–32 (2001)

https://link.springer.com/article/10.1023/A:1010933404324

[38] G. Biau, E. Scornet, A random forest guided tour, 25, pages197–227 (2016)

https://link.springer.com/article/10.1007/s11749-016-0481-7

[39] G. Webb, L. Liu, X. Ma, S. Chen: A novel selective naïve Bayes algorithm, 2020, (Elsevier)

https://doi.org/10.1016/j.knosys.2019.105361

[40] L. Thot, A. Kocsor, J. Csirik, On naïve bayes in speech recognition, 2005, Volume: 15, Issue: 2, page 287-294,

International Journal of Applied Mathematics and Computer Science

http://matwbn.icm.edu.pl/ksiazki/amc/amc15/amc15211.pdf

Author's publications

List of scientific publications during the PhD, e.g.:

First-line systemic therapy for metastatic castration-sensitive prostate cancer: an updated systematic review with novel findings

Matteo Ferro, Giuseppe Lucarelli, Felice Crocetto, Pasquale Dolce, Antonio Verde, Evelina La Civita, Silvia Zappavigna, Ottavio de Cobelli, Giuseppe Di Lorenzo, Bianca Arianna Facchini, **Luca Scafuri**, Livia Onofrio, Angelo Porreca, Gian Maria Busetto, Guru Sonpavde, Michele Caraglia, Michele Klain, Daniela Terracciano, Sabino De Placido, Carlo Buonerba

Crit Rev Oncol Hematol.2020 Dec 11;103198.
doi: 10.1016/j.critrevonc.2020.103198 PMID: 33316417

The use of chest ultrasonography in suspected cases of COVID-19 in the emergency department

Enrico Allegorico, Carlo Buonerba, Giorgio Bosso, Antonio Pagano, Giovanni Porta, Claudia Serra, Pasquale Dolce, Valentina Minerva, Ferdinando Dello Vicario, Concetta Altruda, Paola Arbo, Teresa Russo, Chiara De Sio, Nicoletta Franco, Gianluca Ruffa, Cinzia Mormile, Francesca Cannavacciuolo, Valentina Mercurio, Gelsomina Gervasio, Giuseppe Di Costanzo, Alfonso Ragozzino, **Luca Scafuri**, Gaetano Facchini, Fabio Numis

Future Sci OA. 2020 Nov 30;7(1):FSO635.
doi: 10.2144/fsoa-2020-0127. PMID: 33432268

Perspective: Cancer Patient Management Challenges During the COVID-19 Pandemic.

Terracciano D, Buonerba C, **Scafuri L**, De Berardinis P, Calin GA, Ferrajoli A, Fabbri M, Cimmino A.

Front Oncol. 2020 Aug 18;10:1556. doi: 10.3389/fonc.2020.01556. eCollection 2020.PMID: 32984015

Contralateral prophylactic mastectomy in male breast cancer: where do we stand?

Antonella Sciarra, Carlo Buonerba, Giuseppe Di Lorenzo, **Luca Scafuri**

Future Sci OA. 2021 Jul 2;7(8):FSO746.

PMID: 34295542 PMCID: PMC8288221 doi: 10.2144/fsoa-2021-0071. eCollection 2021 Sep.

Three vs. Four Cycles of Neoadjuvant Chemotherapy for Localized Muscle Invasive Bladder Cancer Undergoing Radical Cystectomy: A Retrospective Multi-Institutional Analysis

Matteo Ferro, Ottavio de Cobelli, Gennaro Musi, Giuseppe Lucarelli, Daniela Terracciano, Daniela Pacella, Tommaso Muto, Angelo Porreca, Gian Maria Busetto, Francesco Del Giudice, Francesco Soria, Paolo Gontero, Francesco Cantiello, Rocco Damiano, Fabio Crocerossa, Abdal Rahman Abu Farhan, Riccardo Autorino, Mihai Dorin Vartolomei, Matteo Muto, Michele Marchioni, Andrea Mari,

Luca Scafuri, Andrea Minervini, Nicola Longo, Francesco Chiancone, Sisto Perdona, Pietro De Placido, Antonio Verde, Michele Catellani, Stefano Luzzago, Francesco Alessandro Mistretta, Pasquale Ditonno, Vincenzo Francesco Caputo, Michele Battaglia, Stefania Zamboni, Alessandro Antonelli, Francesco Greco, Giorgio Ivan Russo, Rodolfo Hurle, Nicolae Crisan, Matteo Manfredi, Francesco Porpiglia, Giuseppe Di Lorenzo, Felice Crocetto, Carlo Buonerba

Front Oncol. 2021 May 11 PMID: 34046347 PMCID: PMC8144638 DOI: 10.3389/fonc.2021.651745

Immune checkpoint inhibitors in penile cancer

Carlo Buonerba, **Luca Scafuri**, Ferdinando Costabile, Bruno D’Ambrosio, Simona Gatani, Pasquale Verolino, Rossella Di Trolio, Vincenzo Cosimato, Antonio Verde & Giuseppe Di Lorenzo

Future Sci OA. 2021 Aug; 7(7): FSO714.
PMCID: PMC8256326 PMID: 34258024
DOI: 10.2144/fsoa-2021-0044

Assessment of Total, PTEN -, and AR-V7 + Circulating Tumor Cell Count by Flow Cytometry in Patients with Metastatic Castration-Resistant Prostate Cancer Receiving Enzalutamide

Giuseppe Di Lorenzo, Silvia Zappavigna, Felice Crocetto, Mario Giuliano, Dario Ribera, Rocco Morra, **Luca Scafuri**, Antonio Verde, Dario Bruzzese, Simona Iaccarino, Ferdinando Costabile, Livia Onofrio, Martina Viggiani, Alessandro Palmieri, Pietro De Placido, Antonella Lucia Marretta, Erica Pietroluongo, Amalia Luce, Marianna Abate, Zahrasadat Navaeiseddighi, Vincenzo Francesco Caputo, Giuseppe Celentano, Nicola Longo, Matteo Ferro, Franco Morelli, Gaetano Facchini, Michele Caraglia, Sabino De Placido, Carlo Buonerba

Clinical Genitourinary Cancer - 2021 Apr 03
PMID: 33958297 DOI: 10.1016/j.clgc.2021.03.021

A risk-group classification model in patients with bladder cancer under neoadjuvant cisplatin-based combination chemotherapy

Matteo Ferro, Giuseppe Lucarelli, Ottavio de Cobelli, Pasquale Dolce, Daniela Terracciano, Gennaro Musi, Angelo Porreca, Gian Maria Busetto, Francesco Del Giudice, Francesco Soria, Paolo Gontero, Francesco Cantiello, Rocco Damiano, Fabio Crocerossa, Abdal Rahman Abu Farhan, Riccardo Autorino, Mihai Dorin Vartolomei, Michele Marchioni, Andrea Mari, Andrea Minervini, Nicola Longo, Giuseppe Celentano, Francesco Chiancone, Sisto Perdonà, Paola Del Prete, Pasquale Ditonno, Michele Battaglia, Stefania Zamboni, Alessandro Antonelli, Francesco Greco, Giorgio Ivan Russo, Rodolfo Hurle, Nicolae Crisan, Matteo Manfredi, Francesco Porpiglia, Dario Ribera, Pietro De Placido, Sergio Facchini, **Luca Scafuri**, Antonio Verde, Giuseppe Di Lorenzo, Vincenzo Cosimato, Angelo Luciano, Vincenzo Francesco Caputo, Felice Crocetto, Carlo Buonerba

Future Oncol. 2021 Oct;17(30):3987-3994.
PMID: 34278815 doi: 10.2217/fon-2020-1298. Epub 2021 Jul 19.

COVID-19 and prostate cancer: a complex scenario with multiple facets

Felice Crocetto, Luciana Buonerba, **Luca Scafuri**, Vincenzo Caputo, Biagio Barone, Antonella Sciarra, Antonio Verde, Armando Calogero, Carlo Buonerba, Giuseppe Di Lorenzo

Future Sci OA 2021 Oct;8(1):FSO. doi: 10.2144/fsoa-2021-0113. Epub 2021 Nov 23.
PMID: 34898669 PMCID: PMC8621215 DOI: 10.2144/fsoa-2021-0113

Kaempferol, Myricetin and Fisetin in Prostate and Bladder Cancer: A Systematic Review of the Literature

Felice Crocetto, Erika di Zazzo, Carlo Buonerba, Achille Aveta, Savio Domenico Pandolfo, Biagio Barone, Francesco Trama, Vincenzo Francesco Caputo, **Luca Scafuri**, Matteo Ferro, Vincenzo Cosimato, Ferdinando Fusco, Ciro Imbimbo, Giuseppe Di Lorenzo

Nutrients. 2021 Oct 23;13(11):3750. doi: 10.3390/nu13113750.
PMID: 34836005 PMCID: PMC8621729

Does perioperative systemic therapy represent the optimal therapeutic paradigm in organ-confined, muscle-invasive urothelial carcinoma?

Luca Scafuri, Antonella Sciarra, Felice Crocetto, Matteo Ferro, Carlo Buonerba, Francesco Ugliano, Germano Guerra, Roberto Sanseverino, and Giuseppe Di Lorenzo

Future Sci OA. 2021 Oct; 7(9): FSO770.

doi: 10.2144/fsoa-2021-0092 PMCID: PMC8558855 PMID: 34737893

Fisetin as an adjuvant treatment in prostate cancer patients receiving androgen-deprivation therapy

Giuseppe Di Lorenzo, **Luca Scafuri**, Ferdinando Costabile, Liuba Pepe, Anna Scognamiglio, Felice Crocetto, Germano Guerra, Carlo Buonerba

Future Sci OA. 2022 Feb 11;8(3):FSO784. doi: 10.2144/fsoa-2022-0002. eCollection 2022 Mar.

PMID: 35251698 PMCID: PMC8890115 DOI: 10.2144/fsoa-2022-0002

Immune Checkpoint Inhibitors as a Neoadjuvant/Adjuvant Treatment of Muscle-Invasive Bladder Cancer: A Systematic Review

Biagio Barone, Armando Calogero**,** **Luca Scafuri**, Matteo Ferro, Giuseppe Lucarelli, Erika Di Zazzo, Enrico Sicignano, Alfonso Falcone, Lorenzo Romano, Luigi De Luca, Francesco Oliva, Benito Fabio Mirto, Federico Capone, Ciro Imbimbo, Felice Crocetto

Cancers (Basel) 2022 May 21;14(10):2545. doi: 10.3390/cancers14102545

PMID: 35626149 PMCID: PMC9139497 DOI: 10.3390/cancers14102545

The Effect of Vaccination against COVID-19 in Cancer Patients: Final Results of the COICA Trial

Giuseppe Di Lorenzo, Concetta Ingenito, Bruno D'Ambrosio, Chiara Ranieri, Michela Rosaria Iuliucci, Mario Iervolino, Ferdinando Primiano, Luciana Buonerba, Giuseppina Busto, Claudia Ferrara, Annamaria Libroia, Gianluca Ragone, Ferdinando De Falco, Ferdinando Costabile, Pietro Fimiani, Francesco Ugliano, Emilio Leo, Giandomenico Roviello, **Luca Scafuri**, Carlo Buonerba

Oncology 2022;100(9):512-518. doi: 10.1159/000525962. Epub 2022 Jul 11.

PMID: 35817009 DOI: 10.1159/000525962

The Impact of Routine Molecular Screening for SARS-CoV-2 in Patients Receiving Anticancer Therapy: An Interim Analysis of the Observational COICA Study

Di Lorenzo G · Iervolino M · Primiano F · D'Ambrosio M · Ingenito C · Buonerba L · Busto G · Ferrara C · Libroia A · Ragone G · De Falco F · Costabile F · Fimiani P · Ugliano F · Ranieri C · Leo E · Roviello G · **Scafuri L** · Guerra G · Buonerba C

Oncology 2022;100:505–511 2022, Vol.100, No. 9 September 2022

PMID: 34839299 DOI: 10.1159/000521086

Mediterranean Diet as a Supportive Intervention in Cancer Patients: Current Evidence and Future Directions

Roberta Rubino, Michela Rosaria Iuliucci, Simona Gatani, Arianna Piscosquito, Bruno D'Ambrosio, Concetta Ingenito, **Luca Scafuri**, Carlo Buonerba and Giuseppe Di Lorenzo

Curr. Oncol. 2022, 29(10), 7579-7582 Published: 11 October 2022

PMID: 36290874 PMCID: PMC9600992 DOI: 10.3390/curroncol29100597

Predictors of Efficacy of Immune Checkpoint Inhibitors in Patients With Advanced Urothelial Carcinoma: A Systematic Review and Meta-Analysis

Matteo Ferro, Felice Crocetto, Sabin Tataru, Biagio Barone, Pasquale Dolce, Giuseppe Lucarelli, Guru Sonpavde, Gennaro Musi, Alessandro Antonelli, Alessandro Veccia, Daniela Terracciano, Gian Maria Busetto, Francesco Del Giudice, Michele Marchioni, Luigi Schips, Francesco Porpiglia, Cristian Fiori, Giuseppe Carrieri, Francesco Lasorsa, Antonio Verde, **Luca Scafuri** , Carlo Buonerba, Giuseppe Di Lorenzo

Clin Genitourin Cancer. 2023 May 27;S1558-7673(23)00137-4.

PMID: 37419854 DOI: 10.1016/j.clgc.2023.05.017

PREVES: A Population-Based Survey Focused on Cancer and Nutrition

Giuseppe Di Lorenzo, Concetta Ingenito, Mario Iervolino, Gennaro Sosto, Primo Sergianni , Ferdinando Primiano, Arianna Piscosquito, Michela Rosaria Iuliucci, Roberta Rubino, Simona Gatani , Francesco Ugliano, **Luca Scafuri**, Ferdinando Costabile, Bruno D'Ambrosio, Alessandra D'Antonio, Antonio Crescenzo, Francesca Cappuccio, Carlo Buonerba

Oncology. 2023;101(6):369-374. doi: 10.1159/000530626. Epub 2023 Apr 13.

PMID: 37054688 DOI: 10.1159/000530626

Nature's hidden gem: quercitrin's promising role in preventing prostate and bladder cancer

Benito Fabio Mirto, **Luca Scafuri**, Enrico Sicignano, Ciro De Luca, Pasquale Angellotto, Giuseppe Di Lorenzo, Daniela Terracciano, Carlo Buonerba, Alfonso Falcone

Future Sci OA. 2023 May 16;9(6):FSO867. doi: 10.2144/fsoa-2023-0041. eCollection 2023 Jul.

PMID: 37228856 PMCID: PMC10203909 DOI: 10.2144/fsoa-2023-0041

Comparing cardiovascular adverse events in cancer patients: A meta-analysis of combination therapy with angiogenesis inhibitors and immune checkpoint inhibitors versus angiogenesis inhibitors alone

Felice Crocetto, Matteo Ferro, Carlo Buonerba, Luca Bardi, Pasquale Dolce, **Luca Scafuri**, Benito Fabio Mirto, Antonio Verde, Antonella Sciarra, Biagio Barone, Armando Calogero, Caterina Sagnelli, Gian Maria Busetto, Francesco Del Giudice, Simone Cilio, Guru Sonpavde, Rossella Di Trolio, Giuseppe Luca Della Ratta, Gabriele Barbato, Giuseppe Di Lorenzo

Crit Rev Oncol Hematol. 2023 Aug;188:104059. doi: 10.1016/j.critrevonc.2023.104059. Epub 2023 Jun 21.

PMID: 37353178 DOI: 10.1016/j.critrevonc.2023.104059