Università degli Studi di Verona

DIPARTIMENTO DI INFORMATICA
Corso di Laurea in Bioinformatica

Approcci di classificazione per la diagnosi di steatosi epatica non alcolica in pazienti diabetici

Laureando:
Simonetti Andrea

Relatore:
Bicego Manuele

Correlatori:
Danese Elisa
Salvagno Gian Luca

Anno Accademico 2020–2021

Indice

Sommario

Questo studio si colloca nell'ambito della *Pattern Recognition* ed in particolare ne prevede l'applicazione in campo biomedico. L'obiettivo di questa analisi è quello di caratterizzare un insieme di pazienti diabetici, affetti o meno da steatosi epatica non alcolica (NAFLD), ciascuno dei quali è descritto da una serie di parametri chimici e clinici. Tra questi, particolare attenzione è prestata ai livelli degli acidi biliari, ottenuti tramite una tecnica innovativa basata su LC-MS. I dati a disposizione ci sono stati forniti dal gruppo della Professoressa Elisa Danese e del Professor Gian Luca Salvagno e provengono da 216 pazienti dell'ospedale di B.go Trento a Verona.

Durante lo studio sono state impiegate diverse tecniche, a partire da quelle di riduzione della dimensionalità (Principal Component Analisys, T-distributed Stochastic Neighbor Embedding), per poi passare a quelle di classificazione (K-Nearest-Neighbor, Support Vector Machines, Random Forest) ed infine a quelle di Feature Selection (Least Absolute Shrinkage and Selection Operator), al fine di diagnosticare la presenza o assenza della patologia NAFLD a partire dai livelli degli acidi biliari. Nel corso delle sperimentazioni ci si è ritrovati a dover fronteggiare livelli di accuratezza decisamente ridotti. Adottando differenti approcci si è tentato di migliorarli rimuovendo quei pazienti che potevano essere fonte di errore, ma senza ottenere risultati promettenti. Quello che quindi si propone in questo rapporto è una pipeline di lavoro generale, applicabile in contesti simili a quello illustrato nei capitoli che seguono.

Capitolo 1

Introduzione

In questo capitolo si vuol fornire al lettore il contesto biologico entro cui si inserisce questa analisi. Si parlerà in breve della patologia sulla quale ci si è concentrati nel corso dello studio e delle misurazioni che giocano il ruolo più importante nelle attività descritte nei capitoli che seguono.

1.1 NAFLD: la patologia

La *steatosi epatica non alcolica* (Non-alcoholic Fatty Liver Disease, NAFLD) è la causa più comune di disturbi cronici a carico del fegato nella popolazione dei paesi occidentali. Si prevede inoltre che diventerà, dal 2030, la causa più frequente di trapianto di fegato [1].

Per ragioni non del tutto conosciute, questo disordine colpisce l'uomo in misura maggiore della donna, la sua incidenza è infatti valutata intorno al 30-40% nella popolazione maschile, contro il 15-20% in quella femminile. Questa raggiunge il 70% se si considerano solo i pazienti malati di diabete di tipo 2 [1].

Questa patologia si manifesta a diversi livelli di gravità prevalentemente a carico del fegato, con steatosi, steatoepatite, talvolta fibrosi, cirrosi e, nei casi più gravi, cancro. La patogenesi è multifattoriale e non del tutto chiara. Un fattore chiave sembra essere l'accumulo di lipidi, che porta alla perdita della corretta funzionalità da parte degli adipociti. A seguito di questo fenomeno vengono rilasciate elevate quantità di acidi grassi nel sistema circolatorio, che vengono poi assorbiti dal fegato causandone l'affaticamento e i disordini a cui si è accennato sopra. Oltre a fattori genetici, anche le abitudini alimentari e lo stile di vita possono incidere fortemente sullo sviluppo di questa patologia [2].

Nel corso degli ultimi decenni, si è mostrato che il NAFLD non si ritrova confinato esclusivamente a livello epatico, ma si manifesta come disturbo multi-sistema in grado di colpire organi diversi dal fegato e molti pathways regolatori nell'organismo. Ad esempio, la presenza di NAFLD aumenta il rischio di insorgenza di diabete di tipo 2, nonché di

malattie a carico del sistema cardiovascolare e dell'apparato renale. Sebbene il quadro patologico primario di questa malattia riguardi prevalentemente il fegato, la causa principale di morte dovuta al NAFLD è attribuibile ai disordini a carico del sistema cardiocircolatorio [1].

Questa sua potenzialità di dare vita ad un quadro patologico molto grave, unita all'ampia diffusione ed all'aumento della sua frequenza nella popolazione, rendono la diagnosi rapida della patologia uno strumento molto potente ed un importante obiettivo di ricerca.

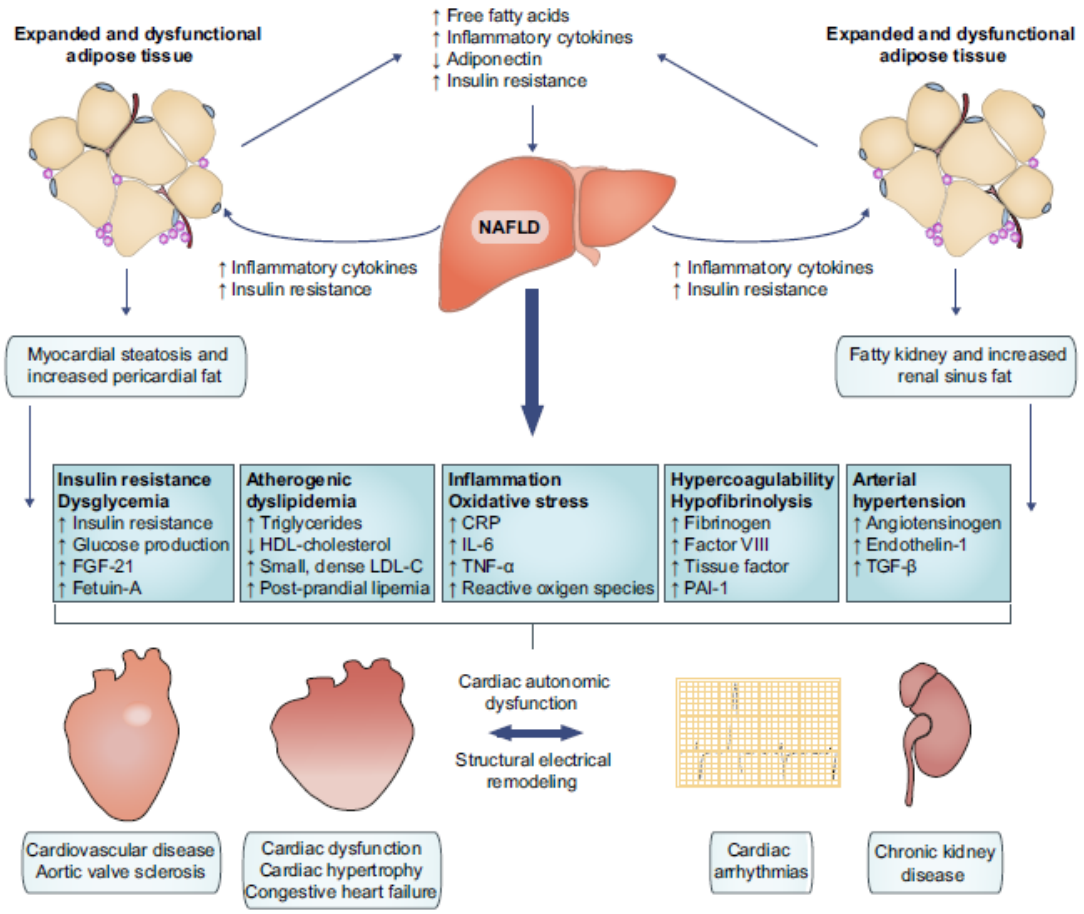

Figura 1.1: Il NAFLD e gli effetti sull'organismo [1].

1.2 Gli acidi biliari e la loro determinazione

Gli acidi biliari, maggiori costituenti della bile, sono molecole anfipatiche prodotte dagli epatociti a partire dal colesterolo. Ricordiamo che le molecole anfipatiche sono caratterizzate dalla presenza contemporanea di regioni idrofobiche ed idrofiliche. Le aree idrofiliche degli acidi biliari rivestono un ruolo protettivo per le cellule del fegato, mentre le regioni idrofobe possono essere citotossiche e causa di stress ossidativo, inducendo disfunzione mitocondriale, la formazione di specie dell'ossigeno molto reattive e, talvolta, apoptosi (morte cellulare) e necrosi [2].

Gli acidi biliari primari (acido colico CA, acido chenodeossicolico CDCA) sono sintetizzati nel fegato tramite ossidazione del colesterolo. Dopo la coniugazione con taurina o glicina, gli acidi biliari coniugati sono secreti nella cistifellea e successivamente passano nell'intestino tenue attraverso il dotto biliare. Circa il 95% di questi verrà poi riassorbito e riportato al fegato, dove ne verranno ripristinate la forma e le funzionalità originali [3].

Queste molecole svolgono nella bile il ruolo di emulsionanti per i grassi ingeriti, permettendo un migliore assorbimento dei lipidi e delle vitamine liposolubili da parte dell'intestino. Essi giocano inoltre un ruolo critico nella regolazione del livello di lipidi epatici e del metabolismo del glucosio, agendo come modulatori di una particolare famiglia di recettori intracellulari (NRS) [2].

Studi pregressi [2] hanno mostrato che in pazienti affetti da insulino-resistenza e diabete di tipo 2 i livelli plasmatici degli acidi biliari sono elevati. Questa alterazione sembra avere effetti negativi sul microbioma, l'insieme di virus, batteri e funghi che popolano il nostro organismo e che svolgono importanti funzioni nel metabolismo degli aminoacidi e nella biosintesi di vitamine. Il microbioma regola inoltre l'assunzione di grassi e l'insulino-resistenza e quindi riveste un ruolo chiave nella patogenesi del NAFLD.

La quantificazione degli acidi biliari nei pazienti considerati nello studio è effettuata tramite l'applicazione di una tecnica innovativa [3] basata sull'impiego della cromatografia liquida accoppiata alla spettrometria di massa (LC-MS). La separazione delle diverse molecole avviene tramite una corsa cromatografica (LC) della durata di circa 7 minuti, durante i quali viene impostato un gradiente per la fase mobile. Questa è costituita da due soluzioni diverse: soluzione A (acqua, formiato di ammonio, acido formico) e soluzione B (acetonitrile+isopropanolo 1:1, formiato di ammonio, acido formico), che vengono mescolate secondo un rapporto che varia durante la corsa. I vari acidi biliari del campione di partenza separati vengono poi quantificati tramite spettrometria di massa. Questa è una tecnica all'avanguardia, in quanto permette di rilevare i livelli di 15 acidi biliari contemporaneamente e non necessita di elevate quantità di campione (è sufficiente iniettare nel sistema cromatografico 10 μL di campione).

1.3 Obiettivo della Tesi

L'obiettivo dello studio è quello di confermare l'esistenza di una relazione tra i livelli degli acidi biliari e la presenza della patologia NAFLD. Quello che si vuole realizzare è, pertanto, un modello di classificazione che sia in grado di prevedere la presenza della malattia a partire dal *pattern* (insieme di misurazioni) di composizione in acidi biliari del paziente in esame. Per raggiungere lo scopo si utilizzeranno tecniche di Pattern Recognition/Machine Learning, metodologie spesso basate sul paradigma dell'apprendimento da esempi. Queste tecniche analizzano l'insieme di dati in input (pattern) per rispondere ad una domanda tipicamente legata al concetto di categoria o classe (Che tipo di oggetto sto osservando? Esiste un oggetto di un dato tipo nell'insieme considerato?). Possiamo vedere come, in maniera molto riduttiva, lo scopo del lavoro si riduca, preso un pattern di misurazioni di un paziente, al trovare una risposta alla domanda: il paziente appartiene alla categoria "NAFLD positivo", oppure no?

Capitolo 2

Background

Lo scopo di questo capitolo è quello di fornire al lettore le basi teoriche per la comprensione di tutto ciò che verrà illustrato nei capitoli successivi e che costituisce il cuore dello studio. Come anticipato, il lavoro si colloca nel contesto della Pattern Recognition, che permette di rispondere a quesiti legati al concetto di categoria e che prevede il susseguirsi di varie attività: dall'acquisizione, al preprocessing, all'eventuale riduzione della dimensionalità e Feature Selection, per finire poi con classificazione e validazione. Si illustrano di seguito brevemente le fasi appena descritte.

2.1 Acquisizione

La prima fase di un qualsiasi studio è senza dubbio l'acquisizione dei dati. In particolare, è necessario eseguire un campionamento a partire dalla popolazione totale che si sta studiando. Il campione generato deve essere il più possibile rappresentativo della popolazione e deve inoltre essere vario e quindi contenere oggetti appartenenti a tutte le categorie secondo le quali si vuol discriminare. A questo punto va generato un pattern (insieme di misurazioni, anche dette *features*) per ogni oggetto campionato dalla popolazione. Le misurazioni possono essere di diverso tipo (valori numerici, stringhe, vettori, grafi ecc.) e provenienti da fonti (e.g. sensori) differenti. Nel caso di questo studio in particolare il campionamento viene eseguito sulla popolazione dei soggetti affetti da diabete, e ognuno dei pazienti selezionati è rappresentato da una serie di misure chimiche e cliniche.

2.2 Preprocessing

Dopo aver campionato un certo insieme di dati riguardanti un problema ed aver costruito dei pattern, una delle primissime attività che possono essere svolte sul dataset è il *preprocessing*. In questo paragrafo tuttavia ci si concentrerà solamente sulla *standardizzazione*,

lasciando ad un altro paragrafo il compito di discutere la riduzione della dimensionalità, che tuttavia fa parte anch'essa delle tecniche di preprocessing.

Un primo importante concetto da introdurre è quello di scala. Una certa misurazione (numero) può infatti relazionarsi in modo diverso con altre a seconda del range di valori che tale misura può assumere. Per meglio comprendere, si può immaginare come due valori, 10 e 12 ad esempio, possano essere considerati molto simili in una scala [0,100], e al tempo stesso molto diversi, se considerati in una scala [10,13]. La standardizzazione dei dati produce dati "senza dimensionalità", in cui tutta la conoscenza su scala e locazione viene persa, per ottenere dati in un formato standard, indipendenti dalla scala.

Solitamente le tecniche di standardizzazione prevedono che ogni insieme di valori relativi ad una certa feature venga traslato di una certa quantità e poi scalato di un certo fattore. In generale, considerando X l'insieme dei dati originale, il dataset standardizzato X^* è così specificato:

$$x^*_{ji} = \frac{x_{ji} - L_j}{M_j}$$

dove x^*_{ji} è la j-esima feature dell'i-esimo oggetto dopo la standardizzazione, x_{ji} è la j-esima feature dell'i-esimo oggetto prima della standardizzazione, L_j rappresenta l'entità della traslazione della j-esima feature e M_j è il fattore di scala della j-esima feature. Considerando una nuvola di punti, il valore L_j ne provoca uno spostamento lungo la direzione j, mentre il valore M_j una deformazione (scala l'insieme) lungo la dimensione j-esima.

Una tecnica di standardizzazione molto utilizzata in generale e in questo studio, è la *Z-Score standardization*, in cui

$$x^*_{ji} = \frac{x_{ji} - \bar{x}_j}{\sigma_j}$$

dove $\bar{x}_j$ e σ_j sono, rispettivamente, media e deviazione standard calcolate lungo la j-esima feature. L'effetto di questa tecnica è quello di generare un insieme di dati in cui ogni feature è caratterizzata dall'avere media nulla e deviazione standard unitaria (possiamo immaginare la nuvola di punti relativi ad ogni singola feature centrata nell'origine degli assi del sistema di riferimento adottato e di forma perfettamente sferica con raggio unitario).

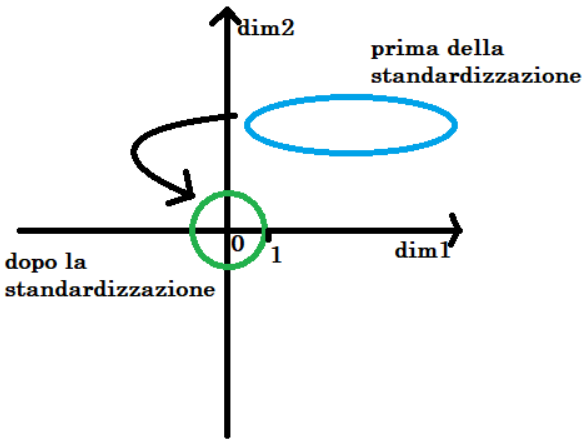

Figura 2.1: La Z-Score standardization.

2.3 Riduzione della dimensionalità

Come anticipato, gli algoritmi di *riduzione della dimensionalità* si inseriscono nella più ampia famiglia delle tecniche di preprocessing. A differenza di quelle di standardizzazione, queste procedure non agiscono alterando la forma secondo cui sono distribuiti i valori delle varie features, ma vanno a modificare il pattern nella sua definizione, diminuendo il numero di features che lo compongono.

Talvolta queste attività si rendono necessarie ad esempio per ridurre i tempi di elaborazione. Con dataset ad elevata dimensionalità, c'è il rischio poi di incorrere nella *curse of dimensionality*, fenomeno per cui i modelli di classificazione iniziano a perdere in capacità di generalizzazione, la capacità di classificare correttamente oggetti non presenti nell'insieme di addestramento.

Le tecniche di riduzione della dimensionalità servono proprio ad evitare questo, permettono di ottenere una nuova rappresentazione dei dati più compatta, in cui ogni oggetto è rappresentato da un numero di misurazioni più basso rispetto a quello originale.

Vengono descritte brevemente 2 tecniche di riduzione della dimensionalità usate:

- *Principal Component Analysis* (PCA) [4], una tecnica classica di riduzione della dimensionalità ampiamente sfruttata in molti contesti medici. La tecnica trasforma linearmente le variabili andando ad estrarre le direzioni a massima varianza;

- *T-distributed Stochastic Neighbor Embedding* (TSNE) [5], un algoritmo avanzato di riduzione della dimensionalità che prevede una trasformazione dei dati di tipo non lineare: modella i punti in modo che oggetti vicini nello spazio originale risultino vicini nello spazio a dimensionalità ridotta, e oggetti lontani risultino lontani, cercando di preservare la struttura locale.

2.4 Classificazione

Classificare un oggetto significa attribuirlo ad una *categoria*. Un sistema di classificazione è un sistema di decisione, che attribuisce ad un certo elemento la classe di appartenenza. Per costruire un classificatore spesso si utilizza il cosiddetto paradigma dell'*apprendimento da esempi*. Un sistema di classificazione viene costruito usando un insieme di oggetti (training set) di cui si conosce la reale classe di appartenenza. L'obiettivo è quello di generare un modello in grado di generalizzare, cioè di classificare correttamente anche oggetti che non appartengono al training set.

Un classificatore può essere visto come una funzione $y = f(x)$ che ritorna un valore y discreto (una delle possibili classi o categorie) a partire dal pattern x. Esistono diversi modi di stimare questa funzione f: nel paragrafo seguente vedremo alcune possibilità utilizzate in questa Tesi.

2.4.1 Tecniche di classificazione

Nel corso delle analisi, si sono impiegati tre modelli di classificazione molto famosi, si cercherà ora di spiegare brevemente in cosa consistono e quali sono il loro punti di forza e debolezza.

K-Nearest Neighbor

Il *K-Nearest Neighbor* (KNN) [6] rientra nella famiglia dei classificatori generativi non parametrici, in cui quindi i parametri del modello vengono stimati assumendo che la forma delle funzioni densità di probabilità secondo cui si distribuiscono non sia nota.

Questo modello di classificazione si basa sul concetto di similarità. L'idea è quella di classificare un punto sulla base della classe degli oggetti che gli "assomigliano", cioè che si trovano ad una *distanza* molto bassa.

Il funzionamento del classificatore, dato X un insieme di esempi etichettati (il training set), è il seguente:

- dato un punto x_0 da classificare, si calcola l'insieme U dei k punti in X più vicini ad x_0 secondo una certa metrica;
- si calcola la classe C più frequente all'interno dell'insieme U;
- si assegna x_0 a C.

Il KNN è una tecnica molto semplice, intuitiva e flessibile. Funziona per diverse tipologie di dati, non solo vettoriali, in quanto richiede solamente che si definisca una misura di distanza. Solitamente è accurata e non richiede che vengano gestiti molti parametri, l'unico importante è il valore k, la cui scelta è comunque cruciale. Tuttavia ha

anche degli svantaggi, come ad esempio il fatto che tutti i punti del training set devono essere mantenuti in memoria, oppure che per decidere quale classe assegnare ad un nuovo oggetto vengono presi in considerazione solo k punti. Inoltre spesso risente dei problemi di scala e richiede una definizione di distanza adeguata.

Support Vector Machines

Le *Support Vector Machines* (SVM) [7] sono classificatori discriminativi impiegati diffusamente perché molto potenti e veloci.

Nascono come classificatori binari, sono quindi in grado di discriminare tra due sole classi e per farlo dividono lo spazio in due regioni grazie ad un iperpiano (una retta in 2D, un piano in 3D). Le SVM ricavano l'iperpiano ottimale seguendo un *approccio geometrico* e non probabilistico, cercando di massimizzare il *margine*. Il margine μ dell'iperpiano di separazione h è definito come la distanza minima fra le due classi. I punti che si trovano a tale distanza sono definiti *Support Vectors*.

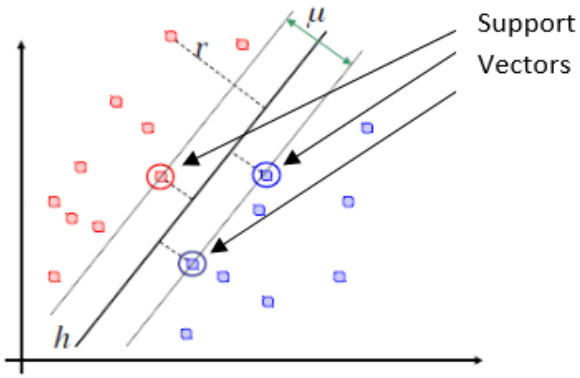

Figura 2.2: Il confine di decisione e i Support Vectors.

È possibile anche gestire dati non linearmente separabili, per cui non esiste quindi un iperpiano che produca una perfetta distinzione delle classi, introducendo nel modello le cosiddette *slack variables*, che consentono la classificazione errata di qualche punto.

Ci sono poi casi in cui un iperpiano rappresenterebbe una risposta troppo semplicistica. Una possibile soluzione può essere quella di proiettare i punti in uno spazio a dimensione maggiore dove questi possano essere separati più facilmente. Questa soluzione genera però due criticità: la curse of dimensionality e la scelta del mapping, che vengono entrambe risolte con l'utilizzo del *kernel trick*. Senza entrare nei dettagli matematici, possiamo dire che il kernel trick permette di lavorare in spazi a dimensionalità enorme, potenzialmente infinita, utilizzando gli stessi algoritmi della variante di base e senza incorrere nel problema della curse of dimensionality.

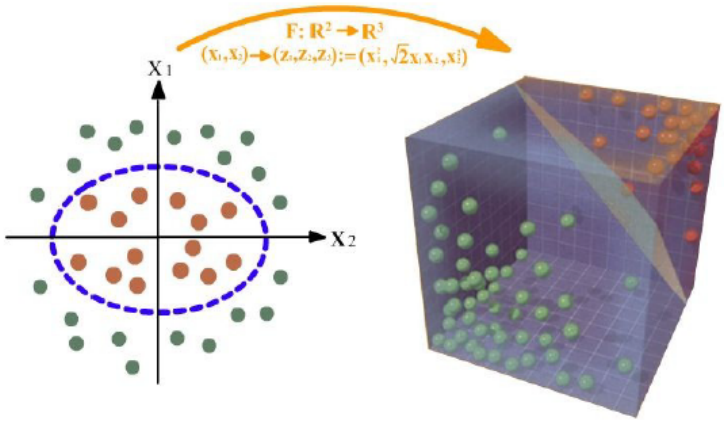

Figura 2.3: Possibili vantaggi dell'aumento della dimensionalità.

Per questo genere di classificatori ci sono più parametri da gestire: la funzione kernel che si vuole utilizzare, con gli eventuali parametri specifici, cui si aggiunge il costo C delle classificazioni errate. Valori di C bassi permetteranno la classificazione errata di un maggior numero di punti, ottenendo soluzioni più smooth, ma probabilmente più grossolane.

Ricapitolando, le SVM sono classificatori con una interpretazione geometrica semplice, la soluzione del training è ottimale e necessitano solamente dei Support Vectors per la definizione dell'iperpiano di separazione, e quindi per la rappresentazione dell'intero modello. D'altro canto però la scelta dei parametri è cruciale, il training può essere oneroso in termini di tempo e non è incrementale, quindi se viene fornito un nuovo punto per il training set l'addestramento va ricominciato dall'inizio.

Random Forest

Le *Random Forest* (RF) [8] sono un metodo di classificazione molto potente che sfrutta un insieme di alberi di decisione [9] binari, detto ensemble. Le Random Forest combinano più alberi di decisione e assegnano un determinato oggetto alla classe che è stata selezionata il maggior numero di volte. In un albero di decisione sono presenti dei test associati a nodi e l'esito dei test stabilisce verso quale dei nodi figli proseguire. Si continua la discesa fino a raggiungere i nodi foglia, nell'ultimo livello, dove è memorizzata la classe da assegnare sulla base del percorso seguito da un particolare oggetto dato in input.

In figura si mostra un semplice esempio di albero di decisione. Y' e Y'' rappresentano le classi e y l'oggetto. A, B e C sono delle generiche features che caratterizzano l'oggetto y sulle quali vengono effettuati i test.

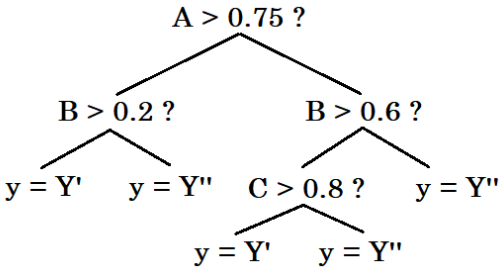

Figura 2.4: Esempio di albero di decisione.

2.5 Feature Selection

Fanno parte della *Feature Selection* tutte quelle tecniche che permettono di selezionare un sottoinsieme delle features che sia maggiormente rappresentativo per gli oggetti che compongono la realtà che stiamo modellando.

Sono quindi tecniche di riduzione della dimensionalità. Diversamente da quelle presentate nella sezione 2.3, che considerano tutte le features e le trasformano, le tecniche di Feature Selection ne selezionano solo alcune sulla base di un certo criterio di ottimalità, eliminando quelle ridondanti, non informative o addirittura dannose.

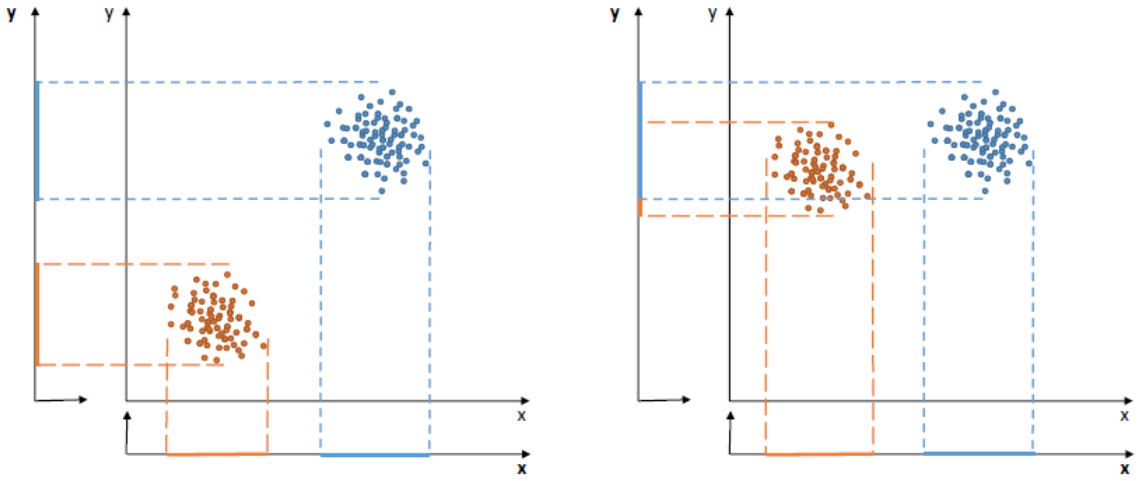

Figura 2.5: (Sinistra) In questo esempio, le features x e y sono ridondanti, entrambe forniscono la stessa informazione nel discriminare i due cluster e quindi una sola delle due è realmente necessaria. (Destra) In quest'altro esempio, la feature y può essere considerata come irrilevante perchè non fornisce informazione utile alla separazione dei due insiemi di oggetti, che infatti risultano sovrapposti se valutati rispetto alla sola variabile y, che potrebbe quindi essere rimossa.

In questa Tesi abbiamo utilizzato una sola tecnica di FS: LASSO [10].

Il *Least Absolute Shrinkage and Selection Operator* (LASSO) è una tecnica di FS molto utilizzata e fa parte della famiglia dei *wrapper*, che sfruttano dei classificatori per attribuire un punteggio ai sottoinsiemi di features selezionati.

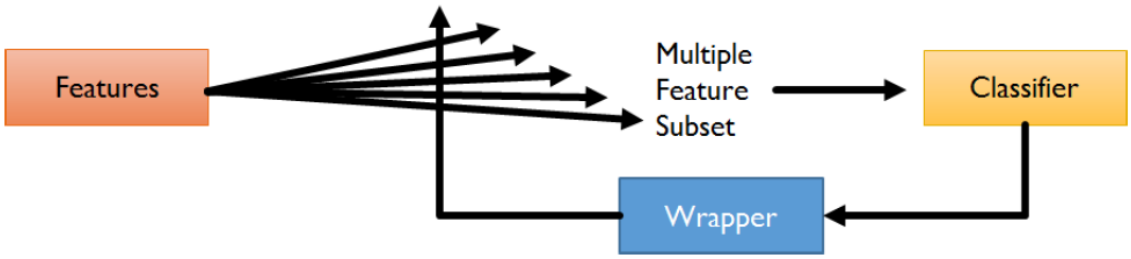

Figura 2.6: Schematizzazione di un wrapper.

Questo metodo parte da una regressione lineare e applica un processo di regolarizzazione che penalizza i coefficienti delle variabili di regressione azzerando quelli delle feature meno rilevanti. La selezione è una conseguenza di questo processo in quanto vengono mantenute solamente quelle features che al termine hanno coefficiente diverso da zero.

2.6 Validazione

Una volta generato un modello di classificazione è necessario quantificare la sua capacità di generalizzare e cioè la sua *accuratezza*, che viene così definita:

$$accuracy = \frac{n_c}{n_t}$$

dove n_c rappresenta il numero di soggetti classificati correttamente e n_t il numero totale di soggetti nel testing set.

La tendenza di un sistema di classificazione, in fase di addestramento, è quella di memorizzare e quindi specializzarsi sugli oggetti che gli vengono presentati. Questa specializzazione, a livelli estremi, viene definita *overtraining*, e conduce alla perdita di capacità di generalizzazione. Se si vuole definire un livello di accuratezza che rispecchi le vere capacità del classificatore, è bene non testare quest'ultimo su oggetti impiegati durante il suo addestramento. Le soluzioni sono due: campionare nuovi dati da poter usare per testare il classificatore, o suddividere il dataset in due insiemi:

- *training set*, con cui effettuare l'addestramento del modello;
- *testing set*, con cui testare e validare il modello.

Solitamente la strategia più utilizzata è quella di dividere in due il dataset: questa tecnica è chiamata *cross validation* e ne esistono diverse varianti.

Una variante molto comune per dividere l'insieme di dati, quando non si hanno a disposizione elevate quantità di soggetti, è quella che viene definita *Leave-One-Out* (LOO).

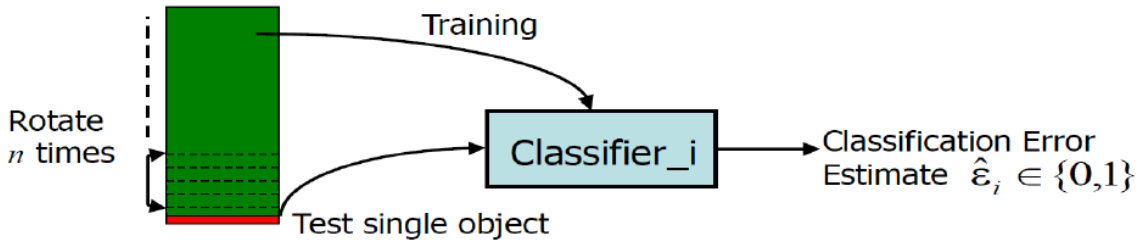

Figura 2.7: Il protocollo Leave-One-Out.

Consideriamo un insieme di n elementi $X = \{x_i\}$, con $1 \leq i \leq n$. Ad ogni oggetto x_i dell'insieme è associata la classe l_i (classe reale di appartenenza).

Il protocollo LOO è strutturato come segue:

- si sceglie un oggetto x_i;
- si utilizza $X \setminus \{x_i\}$ (i restanti elementi dell'insieme) per addestrare il classificatore;
- si assegna l'oggetto x_i alla classe a_i predetta dal modello.

Si ripete per ogni possibile x_i e l'accuratezza risultante si calcola come

$$accuracy = \frac{\sum_{i=1}^{n} c_i}{n}$$

Dove c_i è così calcolato:

$$c_i = \begin{cases} 1 & \text{se } a_i = l_i \\ 0 & \text{altrimenti} \end{cases}$$

Capitolo 3

Pipeline proposta

In questo capitolo si vuole presentare una pipeline di lavoro generica applicabile in studi di classificazione che si svolgono in contesti analoghi a quello presentato nel capitolo 1. Nei paragrafi successivi si discuteranno nel dettaglio le attività relative alle diverse fasi dello studio, ognuna delle quali può essere considerata come un raffinamento delle precedenti al fine di risolvere eventuali problematche insorte. L'ordine con cui sono presentate è quindi quello in cui sono state, e andrebbero, effettuate.

3.1 Prima fase: Visualizzazione

In una classica analisi di classificazione, come in questo caso, la prima strada che è possibile percorrere è quella di ricercare una naturale predisposizione degli oggetti al raggruppamento in classi distinte.

Occorre quindi visualizzare l'intero dataset in un grafico 2D/3D. Osservando le rappresentazioni è possibile valutare il grado di separazione delle diverse classi, caratteristica che fungerà da punto di partenza per analisi successive.

Nell'ambito dello studio, questa è quindi un'analisi di tipo esplorativo, in quanto ci fornisce una prima informazione circa la propensione dei dati a definire classi ben distinguibili, dandoci idea di quali saranno i modelli di classificazione più adatti a descrivere il problema.

Perché il dataset sia visualizzabile, può rendersi necessario ridurne la dimensionalità, adottando ad esempio le tecniche descritte nel capitolo precedente. Un numero di osservazioni (features) maggiore di 3 non è, per ovvie ragioni, visualizzabile.

3.2 Seconda fase: Diverse rappresentazioni e classificazione

Una volta completata la prima fase esplorativa occorre procedere allo sviluppo del modello di classificazione vero e proprio.

Prima di passare all'addestramento di classificatori occorre definire la rappresentazione degli oggetti. Chiaramente, esistono differenti modi per rappresentare i dati a nostra disposizione e rappresentazioni diverse dello stesso insieme di oggetti possono portare a classificatori diversi, permettendo magari una migliore e maggiore separazione delle classi.

Nel caso specifico dello studio in esame abbiamo individuato 4 differenti rappresentazioni basate sugli acidi biliari:

ORIGINALE , costituita dall'insieme di tutti gli acidi biliari. Nel nostro studio ne abbiamo usati 14 (Tabella 3.1);

Feature	Significato
ba_TUDCA	Acido tauroursodeossicolico
ba_GUDCA	Acido glicoursodeossicolico
ba_GCA	Acido glicolico
ba_TCDCA	Acido taurochenodeossicolico
ba_TDCA	Acido taurodeossicolico
ba_UDCA	Acido ursodeossicolico
ba_CA	Acido colico
ba_GCDCA	Acido glicochenodeossicolico
ba_HDCA	Acido iodeossicolico
ba_GDCA	Acido glicodeossicolico
ba_CDCA	Acido chenodeossicolico
ba_GLCA	Acido glicolitocolico
ba_DCA	Acido deossicolico
ba_TCA	Acido taurocolico

Tabella 3.1: Acidi biliari.

AGGREGAZIONE1 (o AGGR1), generata da 4 somme di acidi biliari (le 4 coppie presentate sono generate a partire dallo stesso substrato)

GCA + TCA
GUDCA + TUDCA
GDCA + TDCA
GLCA + TLCA

AGGREGAZIONE2 (o AGGR2), generata da 2 rapporti tra acidi biliari, coniugati e non coniugati

GCA+TCA / CA+CDCA (acidi biliari primari)
GUDCA+TUDCA+GDCA+TDCA+GLCA+TLCA /
UDCA+DCA+LCA (acidi biliari secondari)

AGGREGAZIONE3 (o AGGR3), generata dalla concatenazione delle due precedenti.

A queste è possibile aggiungerne altre, quelle fornite dall'applicazione di tecniche di Feature Selection che, di fatto, ci forniscono un insieme di dati diverso da quello originale, in quanto costituito da un sottoinsieme delle features di partenza. In particolare, sfruttando le tecniche di Feature Selection si sono prodotti due insiemi di dati: il primo in cui sono mantenute solo le prime 5 features (LASSO 5 features) ed il secondo in cui sono mantenute le prime 10 (LASSO 10 features).

Una volta ricavate le diverse rappresentazioni, il passo successivo è quello di definire uno o più protocolli di addestramento e validazione di un modello di classificazione. Questo va a costituire il cuore dell'analisi. In particolare, per ogni rappresentazione vengono utilizzati diversi classificatori e per ognuno di questi viene stimata l'accuratezza di predizione tramite il protocollo LOO. In questo modo sarà possibile stabilire se:

i) esiste la possibilità di discriminare tra pazienti sani e malati;

ii) qual è la rappresentazione migliore per l'insieme di pazienti a nostra disposizione;

iii) qual è il modello di classificazione migliore per risolvere il problema in esame.

In questa fase la scala è un fattore rilevante. Come anticipato nel capitolo 2, da questa criticità nasce la prassi, prima di procedere all'addestramento di un classificatore, di scalare il dataset applicando tecniche di data standardization menzionate sopra. Questa attività permette di renderci indipendenti dalla differente scala nella quale possono essere rappresentate le diverse features.

3.3 Terza fase: Raffinamento delle analisi

In questa sezione sono riportate 3 strategie la cui applicazione è suggerita quando dalle fasi precedenti non si ottengono risultati soddisfacenti, al fine di migliorare il modello di classificazione prodotto.

Analisi degli errori

La prima strategia che si riporta è quella che abbiamo definito, in modo molto generale, "analisi degli errori".

Obiettivo di questa analisi è quello di individuare un eventuale sottoinsieme di pazienti, non casuale, su cui il modello di classificazione è più accurato. Si vuol quindi definire per quali tipologie di soggetti il classificatore avrà una migliore capacità di generalizzazione.

Il primo passaggio è quello di ricavare, per ogni soggetto e diversi metodi di classificazione, la frequenza di attribuzione errata della classe. Possiamo quindi vedere se esiste un gruppo di pazienti su cui tutti i classificatori sbagliano oppure su cui tutti i classificatori fanno giusto, e cercare di caratterizzarli. Per far questo, i pazienti vengono separati in due insiemi: quelli con frequenza d'errore inferiore ad un certo livello finiranno nel primo gruppo, gli altri nel secondo. Questi due nuovi gruppi vengono analizzati al fine di trovare un modo di caratterizzarli. Ad esempio, è possibile calcolare, per ognuno dei parametri chimici e clinici che di solito sono disponibili per i soggetti di uno studio, la distanza tra gli insiemi di misurazioni nei due gruppi di soggetti individuati. In questo modo è possibile vedere se esiste un parametro chimico/clinico critico (e.g. tutti i pazienti con AST > 7 vengono classificati erroneamente). Una possibile alternativa, più complessa, consiste nell'applicare tecniche di Feature Selection all'insieme dei parametri chimici e clinici in modo da trovare un eventuale insieme di misurazioni che ci permetta di descrivere e separare i due raggruppamenti.

Eliminazione degli outliers

Lo scopo di questa operazione, come il nome lascia intendere, è quello di rintracciare ed eliminare dal dataset tutti gli *outliers*, ovvero quegli elementi anomali e quindi non rappresentativi, che potrebbero "confondere" il modello che stiamo cercando di realizzare.

Cruciale in questa fase è il criterio adottato per stabilire se un certo oggetto sia un outlier o meno, un criterio troppo permissivo porterà ad includere nel modello dati anomali, uno troppo restrittivo porterà alla possibile esclusione di soggetti rilevanti. In questo caso sono stati scartati tutti quei soggetti il cui livello di acidi biliari totale (la somma di tutti i livelli) era inferiore a $q_1 - (1.5 \times (q_3 - q_1))$ o superiore a $q_3 + (1.5 \times (q_3 - q_1))$, dove q_1 e q_3 sono rispettivamente il primo e terzo quartile (25° e 75° percentile).

Bilanciamento delle classi

Un'altra attività che potrebbe portare giovamento allo studio consiste nel tentare di bilanciare l'insieme dei dati. Nel dataset infatti, i soggetti potrebbero non essere equamente distribuiti all'interno delle varie classi. Questo squilibrio fa si che, in fase di addestramento, una delle categorie rivesta un ruolo di maggior rilievo e possa influenzare il modello risultante.

Prendiamo in considerazione un insieme di dati suddiviso in due classi, la prima e più numerosa composta da m soggetti, la seconda invece da n soggetti. Per poter bilanciare il dataset sarà necessario eliminare un numero di soggetti dalla prima classe pari a $m-n$, in modo da ottenere due classi caratterizzate da un ugual numero di pattern.

La scelta cruciale qui sta in quali soggetti della classe più popolosa eliminare.

Si sono seguite due alterative: una prima in cui il gruppo di pattern da eliminare viene selezionato in modo casuale, e una seconda in cui, ad ogni iterazione LOO, si eliminano dal training set gli oggetti della classe più popolosa basandosi sulla loro distanza dai soggetti della classe meno numerosa.

Capitolo 4

Validazione sperimentale

Nel capitolo precedente sono stati illustrati i passi di una pipeline generale, che può essere considerata valida per risolvere questo genere di problematica. Qui di seguito vengono riportate le analisi relative all'applicazione della pipeline ai dati a nostra disposizione e i risultati a cui hanno condotto.

4.1 Il dataset

Per la realizzazione di questo studio sono stati messi a disposizione i dati relativi a 216 pazienti. Per ogni paziente si conoscono il sesso, l'età, 17 misurazioni in ambito chimico/clinico e i livelli relativi a 14 acidi biliari (Tabella 3.1). I dati a nostra disposizione sono stati raccolti dal professor Giovanni Targher e i suoi collaboratori nell'ambito di uno studio condotto presso il reparto di endocrinologia dell'ospedale di B.go Trento a Verona. Successivamente sono stati processati dal gruppo della Professoressa Elisa Danese e del Professor Gian Luca Salvagno, da cui ci sono poi stati forniti nell'ambito di una collaborazione della quale questo rapporto fornisce una descrizione.

La presenza della patologia NAFLD in ogni soggetto riveste per noi un ruolo cruciale. La variabile può assumere due differenti valori: 0 nel caso in cui il paziente non risulti affetto da tale patologia, 1 altrimenti. L'obiettivo del lavoro è quello di predire la presenza della patologia NAFLD, quindi la variabile di cui si è parlato rappresenta la classe di appartenenza del soggetto. La prima attività svolta sul dataset è quindi l'eliminazione di tutti quei soggetti che non possiedono una misurazione in questo campo, cioè per cui non è possibile sapere se siano o meno affetti da NAFLD.

Il risultato di questa operazione è un nuovo dataset, costituito da 160 pazienti: 76 femmine, delle quali 61 affette da NAFLD (80.3%), e 84 maschi, di cui 51 affetti da NAFLD (60.7%).

4.2 Attività sperimentali e risultati

Di seguito verranno riportati alcuni dettagli circa gli esperimenti svolti. Verranno inoltre mostrati i principali risultati ottenuti durante il corso delle sperimentazioni.

Tutti gli esperimenti sono stati svolti nell'ambiente di calcolo MATLAB.

4.2.1 Visualizzazione

La base di partenza per lo studio, come anticipato in precedenza, sono i livelli dei 14 acidi biliari a disposizione per ogni paziente. Per poter procedere alla visualizzazione del dataset occorre ridurre la dimensionalità di ogni pattern da 14 (numero acidi biliari) a 2/3. In particolare vengono applicate le due tecniche di riduzione della dimensionalità illustrate nel capitolo 2: PCA e TSNE. L'insieme viene ridotto a 3 e 2 dimensioni, utilizzando entrambi gli algoritmi. In questa fase sono quindi prodotti 4 plot, illustrati di seguito.

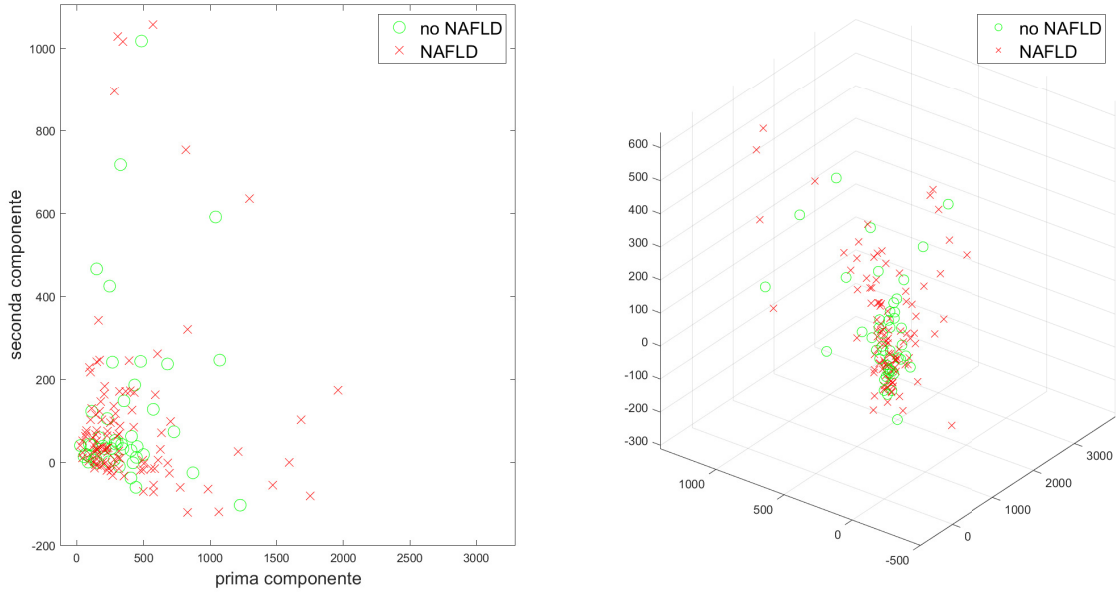

Figura 4.1: (Sinistra) Dettaglio del risultato di riduzione a 2 dimensioni tramite PCA. (Destra) Risultato della riduzione a 3 dimensioni tramite PCA.

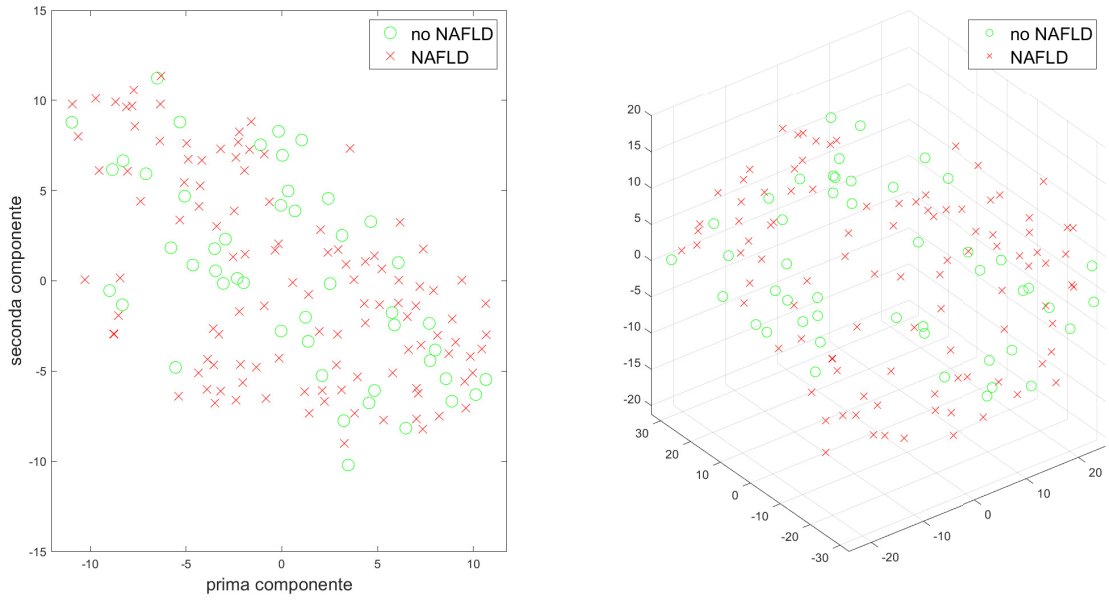

Figura 4.2: (Sinistra) Risultato della riduzione a 2 dimensioni tramite TSNE. (Destra) Risultato della riduzione a 3 dimensioni tramite TSNE.

Come è possibile notare, non sembra sussistere una separazione netta delle due classi. Sia per quanto riguarda la riduzione della dimensionalità mediante PCA, sia per l'algoritmo TSNE le due classi si presentano sovrapposte. Da questi primi tentativi non sembra quindi possibile fissare un confine di decisione che permetta di discriminare tra le due categorie.

Riducendo la dimensionalità del dataset, di fatto, creiamo una rappresentazione semplificata della realtà descritta. E' naturale quindi presumere che parte dell'informazione veicolata dai dati vada persa. Un modo di valutare l'entità di questa perdita è calcolare quella che viene definita "varianza spiegata". Per fare ciò si sfruttano gli autovalori calcolati nell'esecuzione della PCA, che forniscono il peso di ogni singola feature. A partire da questi autovalori è quindi possibile realizzare il grafico della varianza spiegata, riportato di seguito.

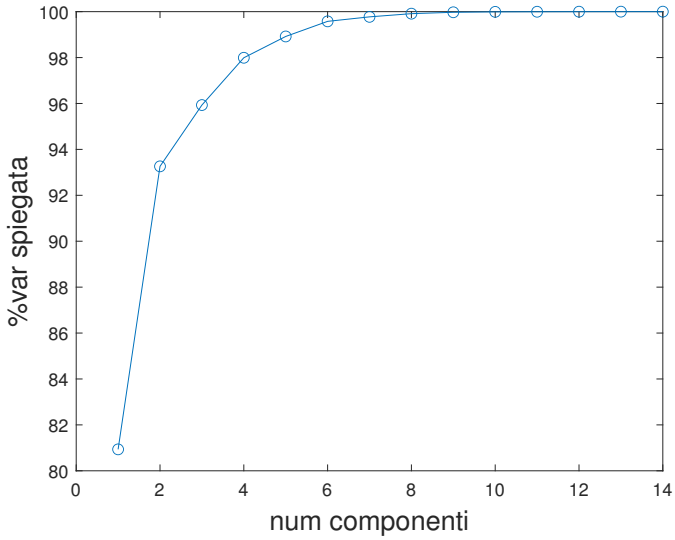

Figura 4.3: Varianza spiegata.

Come è possibile notare, sia con 2 che con 3 features viene spiegato più del 90% della varianza. Possiamo quindi ipotizzare che non ci sia molta perdita di informazione nella proiezione e che quindi sarà difficile creare un classificatore.

4.2.2 Classificazione

Come anticipato nel capitolo precedente, i dati possono essere rappresentati in scale differenti, il che può talvolta influenzare negativamente l'addestramento del classificatore. Per questo motivo si è deciso di procedere con due differenti tecniche di standardizzazione:

- Z-Score standardization, di cui si è discusso nel capitolo 2;
- somma1, con la quale abbiamo portato a 1 la somma totale dei livelli degli acidi biliari di ogni paziente. L'idea su cui ci si è basati è quella per cui sia più importante la relazione tra i diversi acidi biliari piuttosto che la loro effettiva presenza.

Queste due tecniche verranno applicate ad ogni diversa rappresentazione prima di procedere all'addestramento e validazione dei modelli di classificazione. In particolare abbiamo considerato le seguenti rappresentazioni (si veda il capitolo 3 per i dettagli):

- ORIGINALE;
- AGGR1;
- AGGR2;
- AGGR3;
- LASSO 5 features;
- LASSO 10 features.

Una volta standardizzato il dataset, si procede all'addestramento dei classificatori.

La nostra scelta è ricaduta su 3 modelli in particolare: **KNN**, **SVM**, **RF**.

Il primo è stato scelto perché intuitivo e veloce, soprattutto per dataset non troppo estesi, il secondo invece per la sua efficacia dimostrata in molti settori, il terzo per la potenza e la capacità di lavorare in modo efficace anche in presenza di un gran numero di features. Per quanto riguarda il KNN sono state effettuate 15 prove, ogni volta variando il parametro k, il numero dei vicini considerati. Abbiamo quindi ottenuto 15 livelli di accuratezza, per k che varia tra 1 e 30 con passo 2, dei quali è stato selezionato il valore massimo. Nel caso delle SVM abbiamo agito in modo analogo, selezionando il kernel rbf (*Radial Basis Function*) e variando di volta in volta il parametro C. Anche in questo caso sono poi stati selezionati i risultati migliori. Molto meno articolata è stata invece la scelta delle parametrizzazioni per le RF, delle quali è stato modificato solamente il numero di alberi di decisione da considerare, portandolo a 500. Il resto dei parametri è lasciato all'opzione di default. Per la validazione di ognuno dei modelli realizzati è stato implementato il protocollo LOO.

Di seguito si riportano i valori di accuratezza di classificazione per ogni diversa rappresentazione, per ognuno dei preprocessing, suddivisi per modello di classificazione utilizzato (Tabella 4.1).

Viene anche riportato il *No Information Rate* (NIR), un dato di confronto, che aiuta a stabilire la significatività di un valore di accuratezza ottenuto. Il NIR rappresenta l'accuratezza di un classificatore che attribuisce ogni oggetto alla classe più frequente nel training set.

Sono stati effettuati inoltre degli esperimenti sfruttando una funzione di ottimizzazione dei parametri che MATLAB mette a disposizione. Viene in questo caso testato un elevato numero di parametrizzazioni diverse, fino ad ottenere quella che minimizza una certa funzione d'errore calcolata internamente. Questo tipo di ottimizzazione può tuttavia portare ad overtraining, oltre ad essere onerosa in termini di tempo e risorse computazionali. Di

seguito si riportano i valori di accuratezza ottenuti tramite protocollo di validazione LOO (Tabella 4.2).

RAPPRESENTAZIONE	PREPROCESSING	NIR (%)	KNN (%)	SVM (%)	RF (%)
Originale	Z-Score	70	71.88	70	68.75
	Somma1	70	70.63	70	67.50
Aggr1	Z-Score	70	70	70	59.38
	Somma1	70	69.38	70	59.38
Aggr2	Z-Score	70	70	70	65
	Somma1	70	70	70	65.63
Aggr3	Z-Score	70	70	70	61.88
	Somma1	70	68.75	70	67.50
FS (LASSO 5 features)	Z-Score	70	70.63	70	63.75
	Somma1	70	70	70	70
FS (LASSO 10 features)	Z-Score	70	70.63	70	67.50
	Somma1	70	70	70	63.75

Tabella 4.1: Accuratezza di classificazione.

RAPPRESENTAZIONE	NIR (%)	KNN (%)	SVM (%)	RF (%)
Originale	70	71.88	68.13	65.63
Aggr1	70	68.13	70	61.25
Aggr2	70	71.25	70	62.50
Aggr3	70	70	70	61.88
FS (LASSO 5 features)	70	69.38	69.38	62.50
FS (LASSO 10 features)	70	70	70	68.13

Tabella 4.2: Accuratezza di classificazione con ottimizzazione automatica dei parametri.

RF e SVM producono risultati scadenti, qualche risultato sopra il NIR è invece ottenuto sfruttando il KNN. Da notare inoltre che non esiste una grossa differenza tra le varie rappresentazioni considerate. In generale i risultati non sono soddisfacenti, quasi tutti si discostano minimamente dal NIR e molti vi si posizionano anche al di sotto.

4.2.3 Analisi sul dataset

Analisi degli errori

Visti gli scarsi risultati ottenuti nelle analisi della sezione precedente, abbiamo provato a capire se esiste una regolarità negli errori (e.g. se vegono sbagliati sempre i soggetti con un parametro clinico particolare) Per fare questo, si sono confrontate le previsioni di 3 gruppi di classificatori, per un totale di 45 modelli differenti, con le seguenti parametrizzazioni:

- KNN $k = \{1, 3, 5, \ldots, 29\}$;
- SVM $C = \{0.01, 0.1, 0.2, 0.5, 0.8, 1, 2, 5, 10, 15, 20, 50, 100, 200, 500\}$;
- RF $NumAlberi = \{80, 110, 140, \ldots, 500\}$.

Il risultato è l'immagine riportata alla pagina seguente (in orizzontale), così strutturata:

$$Img(i,j) = \begin{cases} 1 & \text{se il paziente j viene correttamente classificato dal modello i} \\ 0 & \text{altrimenti} \end{cases}$$

con i indice di riga e j indice di colonna.

Le diverse parametrizzazioni sono riportate nell'ordine indicato sopra.

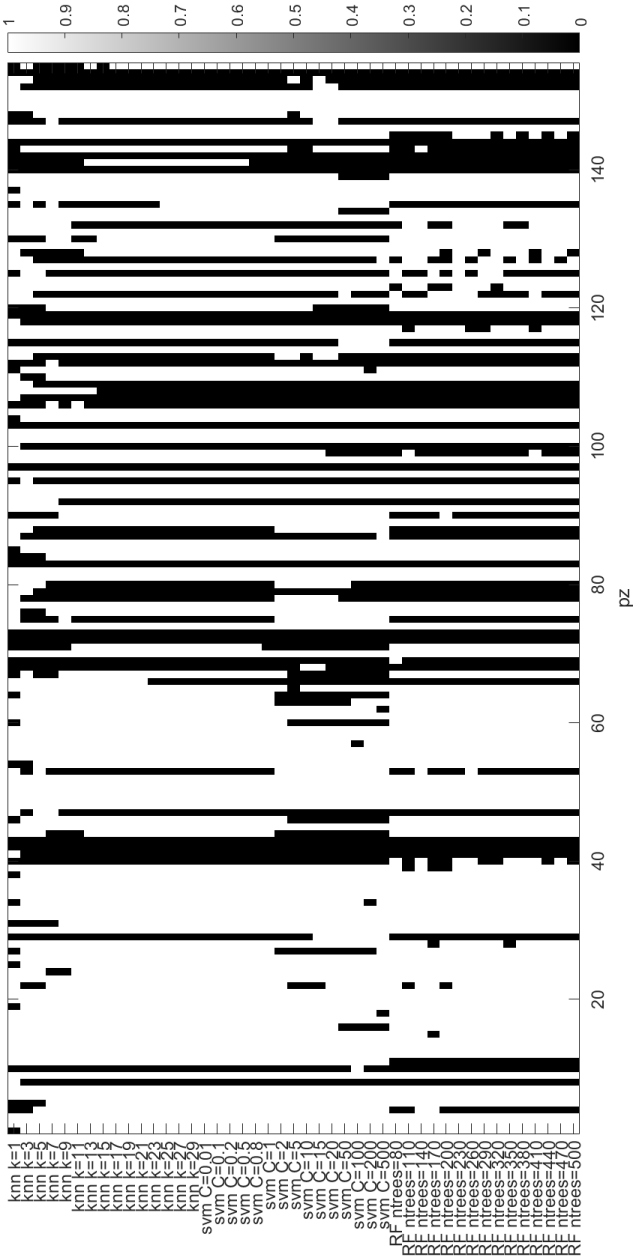

Figura 4.4: Errori di classificazione. I pazienti sono disposti per colonna, un pixel nero indica classificazione errata da parte del modello alla riga corrispondente, un pixel bianco indica invece classificazione corretta.

A questo punto si calcolano le frequenze di attribuzione errata per ogni paziente e si generano due nuovi insiemi di dati: quello dei soggetti a cui viene attribuita la classe errata più dell'80% delle volte, e quello che contiene tutti i rimanenti. Per ogni feature clinico/chimica, viene poi mostrato il range nei due gruppi per vedere se esiste una caratterizzazione (Figura 4.5).

Vista la sovrapposizione dei due insiemi di soggetti nella totalità delle features, e quindi l'impossibilità di trovarne una discriminante, si è tentato, applicando le tecniche di Feature Selection sopra descritte, di trovare un gruppo di features che riuscisse a descrivere bene i due insiemi che stiamo studiando. In parole povere, si è cercato di trovare un sottoinsieme dei parametri chimici e clinici che fosse in grado di "spiegare" perché un certo soggetto venisse classificato in modo errato, in modo da poter definire un classificatore adeguato almeno per una certa categoria di soggetti.

Questa analisi non ha riportato alcun risultato interessante, pertanto non verrà approfondita ulteriormente.

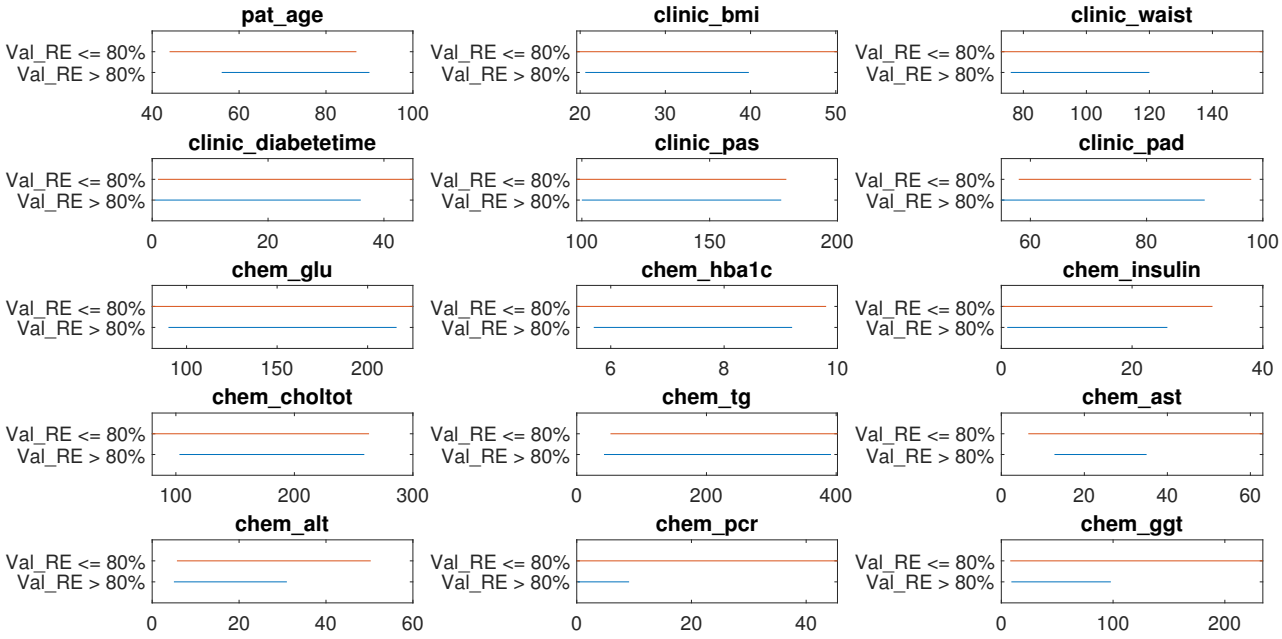

Figura 4.5: Distribuzione dei valori dei parametri chimici e clinici. In alto i pazienti con frequenza di attribuzione errata (RE) inferiore o uguale all'80%, in basso quelli dei pazienti con frequenza di attribuzione errata maggiore dell'80%.

Eliminazione degli outliers

In questo caso l'obiettivo è quello di eliminare gli outliers, per vedere se, eliminando queste osservazioni "anomale", siamo in grado di addestrare un classificatore adeguato. Per la definizione degli outliers è stata presa in considerazione, per ogni paziente, la somma s dei livelli di tutti gli acidi biliari e, sulla base di questo, sono state realizzate 2 soglie. Come descritto nel capitolo precedente, la prima soglia viene posta a 2600, ultimo valore in grado di soddisfare il vincolo $s < q_3 + (1.5 \times (q_3 - q_1))$, con s somma degli acidi biliari, q_3 e q_1 rispettivamente il primo e terzo quartile, mentre la seconda, un po' più permissiva, è posta ad un livello circa il doppio rispetto alla prima, a 5000.

A questo punto, vengono creati due differenti insiemi di dati "filtrati": uno da cui sono stati rimossi tutti i soggetti con $s > 2600$ e un secondo da cui sono stati rimossi tutti i soggetti con $s > 5000$. Una volta ottenuti questi due dataset, si è proceduto a ripetere le prime due fasi della pipeline: visualizzazione e classificazione. Di seguito si riportano i grafici ottenuti nei tentativi di visualizzazione (Figura 4.6) e i risultati di classificazione con KNN, SVM (Figura 4.7) e RF (Tabella 4.3). Per quanto riguarda il dataset filtrato con la soglia a 2600 il NIR è circa al 69.7%, mentre nel caso del filtrato a 5000 il NIR si trova lievemente al di sotto, al 68.1%. Anche in questo caso sembra evidente come eliminare gli outliers non porti ad un incremento dell'accuratezza.

	Filtrato s<2600	Filtrato s<5000	Dataset originale
Accuratezza %	64.58	66.45	65

Tabella 4.3: Confronto di accuratezza con RF.

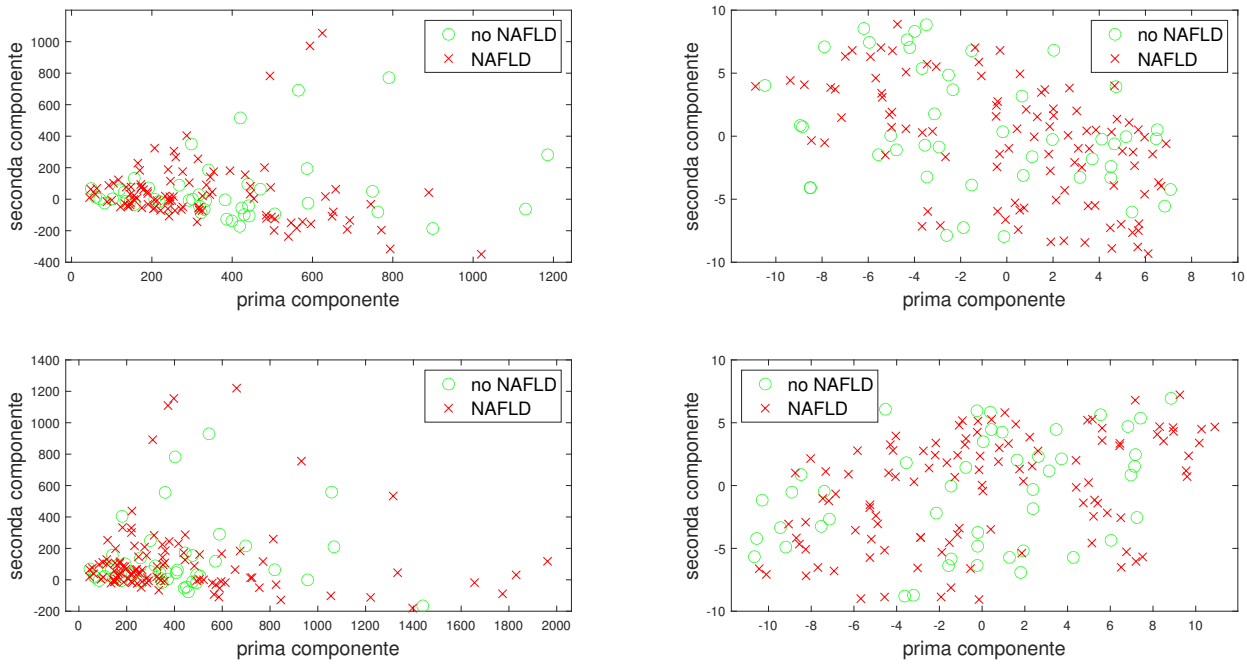

Figura 4.6: (Sinistra in alto) PCA su filtrato soglia 2600. (Destra in alto) TSNE su filtrato soglia 2600. (Sinistra in basso) PCA su filtrato soglia 5000. (Destra in basso) TSNE su filtrato soglia 5000.

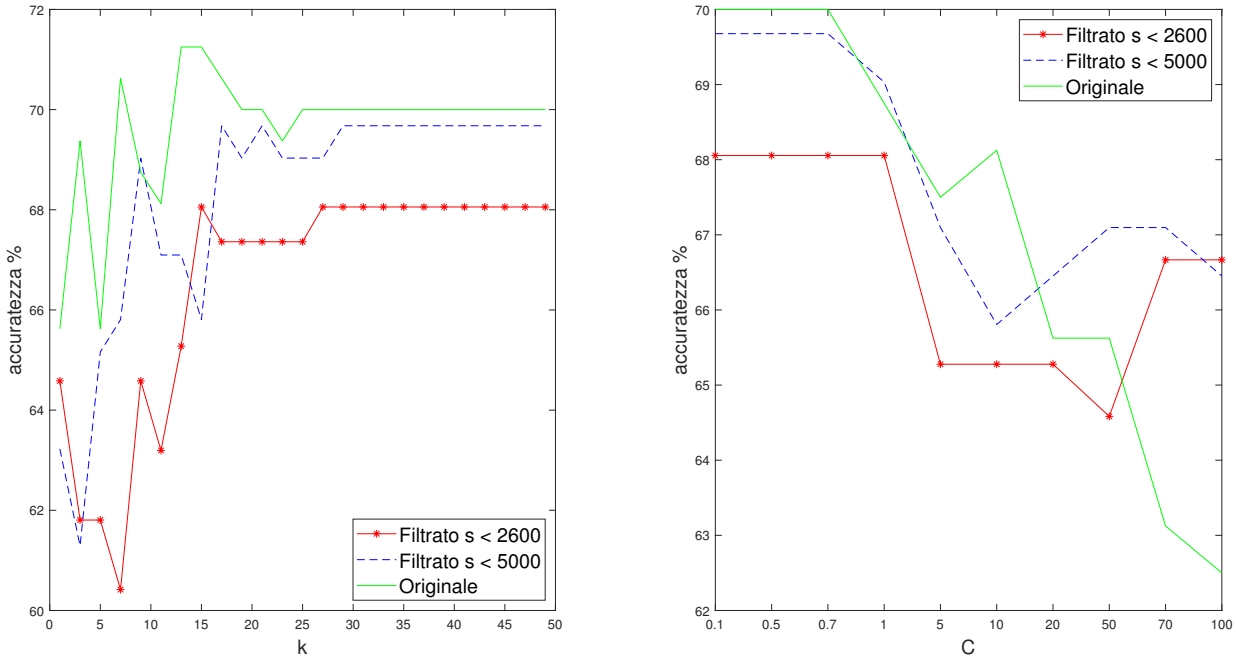

Figura 4.7: (Sinistra) Accuratezza KNN a confronto: i due filtrati e l'originale. (Destra) Accuratezza SVM a confronto: i due filtrati e l'originale.

Bilanciamento delle classi

Come ultimo tentativo si è provato ad eliminare un'altra delle possibili cause del cattivo funzionamento di un classificatore: lo sbilanciamento delle classi. In particolare, per portare le due classi ad esser equamente rappresentate sono state seguite due differenti strategie: una random e una basata sulla distanza.

Ricordiamo che, per poter far si che le due classi siano ugualmente popolate, è necessario eliminare alcuni dei soggetti appartenenti alla categoria più numerosa, nel nostro caso quella dei pazienti malati. Le due strategie riportate differiscono proprio nel modo in cui i pazienti da eliminare vengono selezionati.

Un primo tentativo può essere appunto quello di selezionare i soggetti da rimuovere in modo del tutto casuale. Questa strategia di bilanciamento priva di criterio viene ripetuta, valutando di volta in volta l'accuratezza risultante e cercando di rintracciare dei parametri chimici e clinici in grado di giustificare la selezione di quel certo gruppo di oggetti. Purtroppo questa non ha prodotto risultati su cui valesse la pena soffermarsi.

Per quanto riguarda la seconda strategia, al fine di testare il classificatore su tutti i soggetti del dataset esaminato, si è proceduto in questo modo: ad ogni iterazione LOO, si è bilanciato il training set eliminando i pazienti della classe più popolata che si trovavano alla distanza (euclidea) più bassa dai soggetti dell'altra classe. Adottando questo approccio il nostro obiettivo era quello di allontanare i membri delle due classi sulla base del profilo degli acidi biliari. Pare logico infatti supporre che tra sani e malati debba esistere una differenza nei livelli degli acidi biliari, e che questa differenza possa manifestarsi a diversi livelli di entità (concetto di distanza). La nostra intenzione era quindi quella di eliminare quei pazienti malati che possiedono un quadro clinico che si avvicina maggiormente a quello di un paziente sano. Di seguito viene riportato il risultato di classificazione con un modello KNN, al variare del parametro k (Figura 4.8). In questo caso, dato che le classi si presentano ugualmente rappresentate a seguito del bilanciamento, il NIR si posiziona al 50%.

Dati i risultati decisamente scarsi, non si è approfondita ulteriormente questa analisi.

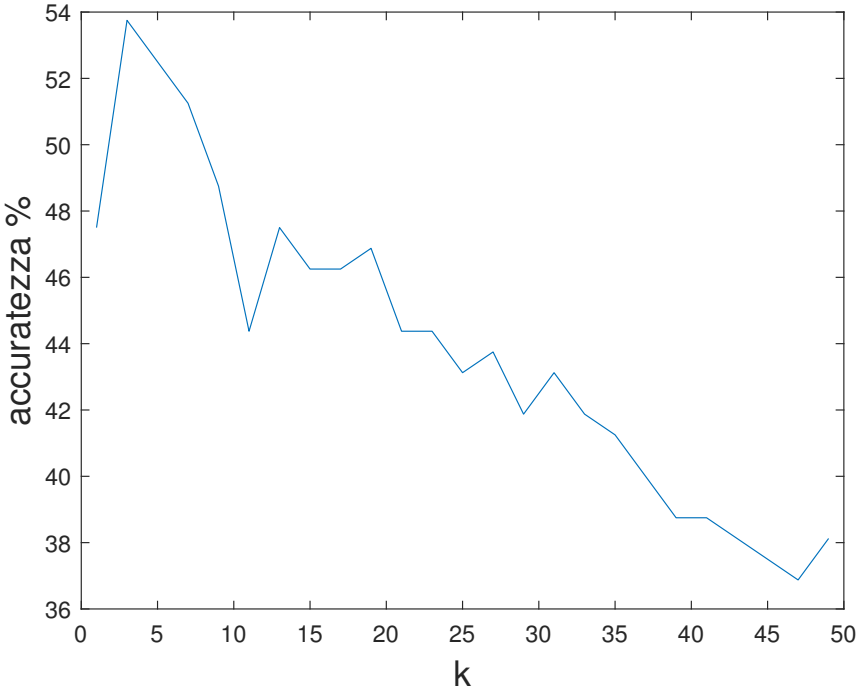

Figura 4.8: Accuratezza risultante con KNN dopo il bilanciamento.

Capitolo 5

Conclusioni

Il lettore potrà facilmente osservare che i risultati del lavoro svolto e descritto nei capitoli precedenti non sono quelli sperati. L'ipotesi di partenza era, come specificato all'inizio di questo rapporto, che ci fosse una correlazione tra i livelli degli acidi biliari e lo stato patologico dei pazienti. Questa ipotesi, seppur sostenuta da evidenze biologiche molto forti, non è stata confermata, né tantomeno smentita, dall'applicazione della pipeline qui descritta.

Possiamo escludere che la causa di questo mancato successo possa essere la presenza di misurazioni (features) rumorose o in grado di confondere i modelli di classificazione testati, poiché in tal caso le tecniche di Feature Selection avrebbero permesso di ovviare al problema. Con le analisi sul dataset riportate nel paragrafo 3.3 (e 4.2.3) si è indagato più nel dettaglio quale potesse essere la causa dell'elevato errore che affligge i classificatori presi in considerazione, cercando nello specifico di trovare un sottoinsieme di pazienti su cui l'errore commesso fosse più contenuto e un modo per discriminare questi soggetti sfruttando i parametri chimici e clinici. I risultati ottenuti ci portano ad ipotizzare che i dati a nostra disposizione non siano stati raccolti nelle condizioni ottimali per il raggiungimento dell'obiettivo prefissato e che quindi in questo risieda la causa dell'errore di generalizzazione diffuso.

Il lascito di questa serie di sperimentazioni non è, come ci si aspettava, un prototipo efficiente di classificazione. Il vero valore aggiunto è invece la pipeline di elaborazione, che può fungere come strumento potente dal quale partire nell'impostare strategie di lavoro per lo studio di realtà affini a quella descritta.

Appendice A

Elenco delle features

Feature	Significato
clinic_NAFLD	Classe di appartenenza NAFLD
pat_age	Età del paziente
clinic_bmi	Indice di massa corporea
clinic_waist	Circonferenza vita
clinic_diabetetime	Tempo trascorso dalla diagnosi del diabete
clinic_pas	Pressione sistolica
clinic_pad	Pressione diastolica
chem_glu	Glucosio
chem_hba1c	Emoglobina glicata
chem_insulin	Insulina
chem_choltot	Colesterolo totale
chem_tg	Trigliceridi
chem_ast	Transaminasi glutammico-ossalacetica
chem_alt	Alanina amino transferasi
chem_pcr	Proteina C reattiva
chem_ggt	Gamma-glutamil transferasi

Tabella A.1: Parametri chimici e clinici.

Bibliografia

[1] Christopher D. Byrne, and Giovanni Targher. *NAFLD: A multisystem disease*. Journal of Hepatology vol. 62, 2015.

[2] Vania Cruz-Ramón, Paulina Chinchilla-López, Oscar Ramírez-Pérez, and Nahum Méndez-Sánchez. *Bile Acids in Nonalcoholic Fatty Liver Disease: New Concepts and Therapeutic Advances*. Annals of Hepatology vol. 16, November 2017.

[3] Elisa Danese, Davide Negrini, Mairi Pucci, Simone De Nitto, Davide Ambrogi, Simone Donzelli, Patricia M.-J. Lievens, Gian Luca Salvagno, and Giuseppe Lippi. *Bile Acids Quantification by Liquid Chromatography–Tandem Mass Spectrometry: Method Validation, Reference Range, and Interference Study*. MDPI diagnostics 10, 2020.

[4] Konstantinos Koutroumbas, and Sergios Theodoridis. *Pattern Recognition*. Academic press, 2003.

[5] Laurens van der Maaten, and Geoffrey Hinton. *Visualizing Data using t-SNE*. J. Machine Learning Research 9, 2008.

[6] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classifcation*. Wiley-Interscience, 2001.

[7] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[8] Leo Breiman. *Random Forests*. Machine Learning 45, 2001.

[9] Leo Breiman, Jerome Friedman, Charles J. Stone, and Richard A. Olshen. *Classification and Regression Trees*. Chapman and Hall/CRC, 1984.

[10] Robert Tibshirani. *Regression Shrinkage and Selection via the Lasso*. Journal of the Royal Statistical Society. Series B, Vol. 58, No. 1, 1996.