

UNIVERSITÀ DELLA CALABRIA



DIPARTIMENTO DI BIOLOGIA, ECOLOGIA E SCIENZE DELLA TERRA

---

DOTTORATO DI RICERCA IN SCIENZE DELLA VITA  
XXIX CICLO

**INNOVATIVI MULTI CLASSIFICATORI CON  
APPLICAZIONE AI SISTEMI DI SUPPORTO  
ALLE DECISIONI CLINICHE**

Settore Scientifico Disciplinare MAT/09

COORDINATORE:

PROF. MARCELLO CANONACO

SUPERVISORI:

PROF. DOMENICO CONFORTI

ING. ROSITA GUIDO

DOTTORANDA:

ING. MARIA CARMELA GROCCIA

# Indice

<b>Introduzione</b>	<b>i</b>
<b>1 Il processo decisionale</b>	<b>1</b>
1.1 Decisioni Individuali . . . . .	2
1.1.1 Approccio razionale . . . . .	2
1.1.2 Approccio basato sulla razionalità limitata . . . . .	3
1.2 Decisioni Organizzative . . . . .	4
1.3 Processo decisionale in ambito biomedico e clinico . . . . .	8
<b>2 Sistemi di Supporto alle Decisioni Cliniche</b>	<b>13</b>
2.1 Tipologie di Sistemi di Supporto alle Decisioni . . . . .	15
2.1.1 Knowledge Based CDSS . . . . .	15
2.1.2 Non Knowledge Based CDSS . . . . .	18
2.2 Architettura dei CDSS . . . . .	19
2.2.1 Approccio Service Oriented Architecture . . . . .	20
2.3 Principali CDSS in letteratura . . . . .	24
2.4 Progettazione e sviluppo di CDSS . . . . .	27
<b>3 Standard di interoperabilità</b>	<b>31</b>
3.1 HL7 . . . . .	32
3.1.1 HL7 Clinical Document Architecture . . . . .	33
3.1.2 HL7 Decision Support Service . . . . .	34

---

3.1.3	HL7 Virtual Medical Record for Clinical Decision Support . . . . .	36
3.1.4	HL7 FHIR . . . . .	38
3.2	OMG . . . . .	40
3.2.1	UML . . . . .	40
3.2.2	MDA . . . . .	41
3.3	HSSP . . . . .	42
3.4	IHE . . . . .	44
<b>4</b>	<b>Multi Classificatori</b>	<b>47</b>
4.1	Principali Classificatori di base . . . . .	49
4.2	Topologia dei Multi Classificatori . . . . .	62
4.3	Progettazione del pool iniziale di classificatori . . . . .	63
4.4	Progettazione della strategia di combinazione . . . . .	65
4.5	Fasi di sviluppo di Multi Classificatori . . . . .	66
4.6	Stato dell'arte dei Multi Classificatori . . . . .	67
4.7	Selezione dinamica di classificatori . . . . .	69
4.7.1	Stato dell'arte delle tecniche di selezione dinamica . . . . .	70
<b>5</b>	<b>Approccio proposto</b>	<b>75</b>
5.1	Architettura proposta . . . . .	75
5.2	Approccio proposto per la progettazione di Multi Classificatori	78
5.2.1	Algoritmo 1 . . . . .	85
5.2.2	Algoritmo 2 . . . . .	86
5.2.3	Algoritmo 3 . . . . .	87
<b>6</b>	<b>Validazione sperimentale</b>	<b>91</b>
6.1	Dataset . . . . .	91
6.1.1	Wisconsin Diagnostic Breast Cancer Dataset . . . . .	92
6.1.2	Wisconsin Breast Cancer Dataset . . . . .	93
6.1.3	Wisconsin Prognostic Breast Cancer Dataset . . . . .	95

---

6.1.4	Cleveland Heart Disease Dataset . . . . .	97
6.1.5	Mammographic Mass Dataset . . . . .	99
6.1.6	Dermatology Dataset . . . . .	101
6.2	Classificatori . . . . .	104
6.3	Protocollo Sperimentale . . . . .	105
6.3.1	10-fold cross validation . . . . .	106
6.3.2	Tuning dei parametri . . . . .	108
6.4	Risultati e discussione . . . . .	112
6.4.1	Risultati su Wisconsin Diagnostic Breast Cancer Dataset	113
6.4.2	Risultati su Wisconsin Breast Cancer Dataset . . . . .	119
6.4.3	Risultati su Wisconsin Prognostic Breast Cancer Dataset	125
6.4.4	Risultati su Cleveland Dataset . . . . .	130
6.4.5	Risultati su Mammographic Mass Dataset . . . . .	135
6.4.6	Risultati su Dermatology Dataset . . . . .	135
6.4.7	Considerazioni finali . . . . .	137
<b>7</b>	<b>Sviluppo del prototipo</b>	<b>149</b>
7.1	Interfaccia utente . . . . .	150
7.2	Strato di interoperabilità . . . . .	151
7.3	CDSS . . . . .	158
	<b>Conclusioni e Sviluppi Futuri</b>	<b>161</b>
	<b>Bibliografia</b>	<b>166</b>



# Elenco delle figure

2.1	Sistema di Supporto alle Decisioni - DSS . . . . .	14
2.2	Schema architetturale di un Knowledge Base CDSS . . . . .	16
2.3	Esempio di architettura SOA . . . . .	22
2.4	Fasi di progettazione e sviluppo di un CDSS . . . . .	29
3.1	Interazione tra un client e un CDSS attraverso il servizio standard HL7 DSS . . . . .	35
4.1	Architettura generale di un Multi Classificatore . . . . .	48
4.2	Rete neurale con singolo neurone . . . . .	49
4.3	Esempio di rete neurale multistrato . . . . .	51
4.4	Esempio di dati linearmente separabili . . . . .	53
4.5	Esempio di dati non linearmente separabili . . . . .	54
4.6	Esempio di mappatura . . . . .	56
4.7	Esempio di classificazione 3-NN . . . . .	59
4.8	Esempio di albero decisionale . . . . .	61
4.9	Topologie di Multi Classificatore . . . . .	62
4.10	Fasi di sviluppo di un Multi Classificatore . . . . .	67
4.11	Regioni di competenza diverse per lo stesso set di dati . . . . .	70
5.1	Architettura di un KM . . . . .	77
5.2	Architettura proposta per CDSS . . . . .	78
5.3	Schema generale del processo di classificazione . . . . .	79

5.4	Confronto tra una regione locale costruita mediante l'algoritmo 5NN e una costruita mediante l'uso della soglia $R$ . . . . .	82
5.5	Fasi del processo di selezione dinamica . . . . .	83
6.1	WDBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA . . . . .	116
6.2	WDBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA . . . . .	117
6.3	WDBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote . . . . .	118
6.4	WBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA . . . . .	122
6.5	WBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA . . . . .	123
6.6	WBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote . . . . .	124
6.7	WPBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA . . . . .	127
6.8	WPBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA . . . . .	128
6.9	WPBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote . . . . .	129
6.10	Cleveland Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA . . . . .	132
6.11	Cleveland Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA . . . . .	133
6.12	Cleveland Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote . . . . .	134
6.13	Mammographic Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA . . . . .	141

---

6.14	Mammographic Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA . . . . .	142
6.15	Mammographic Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote . . . . .	143
6.16	Dermatology Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA . . . . .	145
6.17	Dermatology Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA . . . . .	146
6.18	Dermatology Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote . . . . .	147
7.1	Interfaccia web-based per il Breast Cancer CDSS . . . . .	150
7.2	Interfaccia web-based in caso di diagnosi di nodulo maligno . . . . .	151
7.3	Interfaccia web-based in caso di diagnosi di nodulo benigno . . . . .	152
7.4	Principali interazioni tra l'interfaccia utente, lo strato di interoperabilità e il CDSS . . . . .	153
7.5	Contratto di servizio definito per l'operazione <i>evaluate</i> della <i>Evaluation Interface</i> dello standard HL7 DSS . . . . .	154
7.6	Modellazione della conversazione tra erogatore e fruitore del servizio . . . . .	155
7.7	Modellazione della coreografia tra erogatore e fruitore del servizio lato Consumer . . . . .	156
7.8	Modellazione della coreografia tra erogatore e fruitore del servizio lato Provider . . . . .	157
7.9	Principali interazioni tra i componenti principali dell'architettura software del CDSS . . . . .	159
7.10	Esempio file XML di richiesta per il CDSS . . . . .	160
7.11	Esempio file XML di risposta del CDSS . . . . .	160





# Elenco delle tabelle

6.1	Dataset . . . . .	92
6.2	Caratteristiche del WDBC Dataset . . . . .	93
6.3	Caratteristiche del WBC Dataset . . . . .	94
6.4	Caratteristiche del WPBC Dataset . . . . .	96
6.5	Caratteristiche del Cleveland Heart Disease Dataset . . . . .	98
6.6	Caratteristiche del Mammographic Mass Dataset . . . . .	100
6.7	Caratteristiche del Dermatology Dataset: dodici attributi che derivano dalla valutazione clinica . . . . .	102
6.8	Caratteristiche del Dermatology Dataset: ventidue attributi che derivano dall'esame istopatologico . . . . .	103
6.9	Parametri principali settabili in Weka per i classificatori usati .	109
6.10	Parametri settabili in Weka per Kernel Polinomiale e Gaussiano	110
6.11	Funzioni Weka per calcolare la distanza . . . . .	110
6.12	Set dei parametri ottimali per SMO, MLP e NB . . . . .	111
6.13	Risultati su WDBC Dataset in termini di accuratezza. Per i tre Algoritmi proposti si specifica la versione basata su Recall (Ver Rec) e Accuratezza (Ver Acc) . . . . .	115
6.14	Risultati su WBC Dataset . . . . .	121
6.15	Risultati su WPBC Dataset . . . . .	126
6.16	Risultati su Cleveland Dataset . . . . .	131
6.17	Risultati su Mammographic Mass Dataset . . . . .	140
6.18	Risultati su Dermatology Dataset . . . . .	144



# Introduzione

Le tecnologie dell'informazione (IT) sono utilizzate ormai in quasi tutti i rami della sanità. La cartella clinica elettronica, le prescrizioni on-line, le immagini digitali provenienti da esami strumentali, la memorizzazione e il recupero delle informazioni dei pazienti sono ormai compiti ben noti agli operatori sanitari e sono stati negli anni implementati con vari gradi di successo. Mentre in passato ci si è molto concentrati sullo sviluppo di soluzioni software orientate alla conservazione e alla gestione dei dati clinici, un ruolo centrale nei sistemi sanitari di tutto il mondo viene ricoperto, negli ultimi anni, dai sistemi di supporto alle decisioni cliniche (CDSS). I processi decisionali nella pratica clinica possono risultare in molti contesti piuttosto complicati. Generalmente, il comportamento del medico nell'assumere decisioni si basa sui principi fondamentali della metodologia clinica e, nel suo complesso, il processo decisionale si configura come una successione di fasi diverse e molto articolate. Tali fasi, proprio per la loro intrinseca natura, possono essere rese più obiettive, accurate e affidabili attraverso l'ausilio di metodologie e tecnologie quantitative, opportunamente implementabili in sistemi computerizzati. Un CDSS è una complessa piattaforma informatica interattiva progettata per aiutare gli operatori sanitari a prendere decisioni, senza però mai sostituirsi a essi. L'obiettivo complessivo riguarda, ovviamente, la possibilità di migliorare la qualità dell'assistenza e della cura dei pazienti riducendo il più possibile gli errori e la scarsa appropriatezza degli interventi.

I sistemi di supporto alle decisioni sono in grado, infatti, di aiutare gli

operatori sanitari ad accedere a informazioni e conoscenze complesse e altamente specializzate non sempre accessibili in maniera tempestiva attraverso le linee guida cartacee e dare un contributo significativo alla qualità e alla sicurezza delle decisioni. I CDSS con la loro velocità e la loro vasta conoscenza possono sicuramente dare un contributo importante all'attuale contesto socio-sanitario internazionale che risulta sempre più caratterizzato da eterogeneità degli attori (pazienti, cittadini, operatori sanitari, ambulatori, ospedali, strutture amministrative), complessità dei processi, integrazione di informazioni provenienti da fonti diverse. Quello attuale è un sistema sanitario che sente sempre più la necessità di garantire un'ampia accessibilità alle informazioni a tutti gli operatori del settore, sia all'interno che all'esterno dell'azienda sanitaria in cui questi operano, attraverso soluzioni che facilitino lo scambio dei dati e l'interoperabilità dei servizi applicativi. Tipicamente i CDSS operano in background e forniscono agli operatori sanitari supporto decisionale in tutte le procedure medico-cliniche sotto forma di suggerimenti, avvisi o promemoria. Lo sviluppo di CDSS negli ultimi anni si è avvalso sempre più del contributo fornito da approcci di apprendimento automatico (Machine Learning) e Data Mining. L'approccio tradizionale allo sviluppo di tali sistemi prevede l'addestramento di classificatori diversi applicati allo stesso problema e l'implementazione nel CDSS solo di quello più accurato. Nel tempo però, si è visto come questo approccio presenta diversi limiti specie quando i dati a disposizione per l'addestramento sono pochi. In questi casi, infatti, non sempre si riesce a identificare in modo affidabile il classificatore migliore; anzi, potrebbe accadere di scegliere addirittura il peggiore tra tutti quelli addestrati. La tendenza degli ultimi anni invece è usare approcci basati su sistemi multi classificatore (MCS). Un sistema multi classificatore è un sistema che addestra sullo stesso insieme di dati più classificatori (detti classificatori di base) i quali cooperano, in base a un criterio prestabilito, al fine di prendere la decisione finale. L'idea alla base di questi sistemi è quella di combinare i vantaggi dei singoli classificatori per ottenere risultati migliori. A tal fine, è importante usare classificatori diversi tra loro e che presentano un

certo grado di complementarità. Nelle applicazioni reali risulta comunque difficile soddisfare questa ipotesi e per evitare in parte il problema viene usato un approccio noto come Dynamic Classifiers Selection (Selezione Dinamica di Classificatori). La tecnica consiste nel selezionare per ciascun campione di test i classificatori più competenti all'interno di una regione locale, detta regione di competenza, definita nello spazio degli attributi.

La tesi propone contributi innovativi fortemente orientati alle applicazioni reali, ovvero:

1. Un'architettura software per CDSS basata su un insieme di moduli di conoscenze cliniche detti Knowledge Modules (KM)
2. Approcci metodologici basati su tecniche di Machine Learning (ML) per il disegno e lo sviluppo dei KM come Sistemi Multi Classificatori che utilizzano la tecnica della selezione dinamica

L'architettura software per CDSS si basa su tre componenti fondamentali: (1) un insieme di KM, (2) un controllore strategico che ha il compito di selezionare, istanziare e orchestrare i KM necessari per elaborare il più efficace supporto decisionale e (3) un interfaccia per gestire le comunicazioni con l'esterno. Tale architettura offre, inoltre, soluzioni atte a facilitare lo scambio di dati e l'interoperabilità dei servizi del settore sanitario attraverso uno strato software di interoperabilità basato su standard internazionali. A prescindere dal tipo di attività di supporto decisionale, l'obiettivo è quello di progettare CDSS che possano interagire agevolmente con gli altri componenti del sistema sanitario e che possano essere consultati in qualsiasi momento e da qualsiasi luogo attraverso un browser o un'applicazione su smartphone. In questa architettura, ciascun KM è indipendente dagli altri e può essere sviluppato e implementato sia secondo approcci deduttivi che induttivi. La tesi propone la realizzazione di KM secondo approcci induttivi. Nello specifico, ciascun KM è implementato come un sistema multi classificatore che usa la tecnica della selezione dinamica. Il metodo di selezione dinamica proposto tiene conto sia delle istanze mal classificate da ciascun classificatore di base

all'interno della regione locale che delle prestazioni generali di ognuno. La regione locale è definita in modo dinamico per ciascun nuovo campione di test attraverso un algoritmo K-Nearest Neighbor (KNN) di tipo adattivo. In generale, la selezione del classificatore all'interno della regione locale avviene in due fasi. Durante la prima fase si escludono i classificatori che hanno erroneamente classificato una o più istanze all'interno della regione mentre nella seconda fase viene scelto il classificatore più competente tra tutti quelli rimasti. La tesi propone MCS basati su tre differenti approcci di selezione dinamica. Tutti gli approcci sono stati testati su dataset pubblici che si riferiscono a importanti problemi decisionali in ambito biomedico e clinico. È importante notare che seppur testati in ambito biomedico e clinico, gli approcci proposti sono estremamente generali e applicabili a qualsiasi ambito di interesse.

Il lavoro è organizzato come segue. Il Capitolo 1 introduce i concetti fondamentali del processo decisionale con particolare riferimento ai processi decisionali in ambito clinico. Il Capitolo 2 riassume le caratteristiche principali dei CDSS; presenta la loro organizzazione architetturale generale con particolare riferimento all'architettura orientata ai servizi (SOA). Il Capitolo 3 presenta una panoramica delle organizzazioni e degli standard di integrazione e interoperabilità più usati nel settore sanitario internazionale. Il Capitolo 4 riassume le caratteristiche principali dei MCS e illustra l'approccio basato sulla selezione dinamica. Il Capitolo 5 descrive l'architettura software per CDSS e gli approcci metodologici proposti per il disegno e lo sviluppo di MCS. Il Capitolo 6 presenta i risultati ottenuti dalla validazione sperimentale mentre il Capitolo 7 illustra l'implementazione di un prototipo per CDSS.

# Capitolo 1

## Il processo decisionale

La decisione è la scelta di intraprendere un'azione, tra più alternative considerate (opzioni), da parte di un individuo o di un gruppo (decisore). Tipicamente, in un reale contesto applicativo, la decisione prende forma e si realizza nel corso di un complesso processo decisionale caratterizzato da un'articolata evoluzione di eventi e di attività. Un modello decisionale è un costrutto analitico che individua gli elementi essenziali di tale processo: il decisore, i suoi attributi cognitivi, le attività di ricerca delle soluzioni, le modalità ed i criteri della scelta. Si parla, quindi, di processi decisionali in relazione a tutto quello che succede dal momento in cui nasce un problema al momento in cui viene definita la soluzione. Lo studio dei processi decisionali ha prodotto diversi modelli e identificato diverse modalità di interazione tra i soggetti coinvolti. In base al numero di attori coinvolti le decisioni possono essere di due tipi:

- Decisioni individuali, assunte da un solo individuo
- Decisioni collettive, assunte per un gruppo da un individuo, dal gruppo stesso o da un altro gruppo



## 1.1 Decisioni Individuali

Per le decisioni individuali sono stati definiti due approcci di risoluzione: approccio razionale e approccio della razionalità limitata. Il primo rappresenta il caso ideale di come le decisioni dovrebbero essere prese; il secondo rappresenta i casi più comuni in cui le decisioni sono soggette a vincoli di tempo e di risorse. Spesso, infatti, le decisioni devono essere prese molto velocemente e poiché si dispone solo di tempo limitato non si può valutare ogni obiettivo, problema o alternativa.

### 1.1.1 Approccio razionale

L'approccio razionale alla decisione individuale enfatizza la necessità di analisi sistematiche del problema, seguite da una scelta e dalla realizzazione, lungo una sequenza logica precisa. Il processo decisionale può essere articolato in otto fasi:

- Monitoraggio dell'ambiente: controllo delle informazioni interne ed esterne che indicheranno deviazioni dal comportamento programmato o accettabile
- Definizione del problema: identificazione dei dettagli essenziali del problema
- Individuazione degli obiettivi della decisione: quali risultati dovrebbero essere ottenuti come conseguenza di una decisione
- Comprensione del problema: si cerca di capire la causa del problema. La comprensione della causa permette di individuare la linea di azione appropriata
- Ricerca e sviluppo di possibili soluzioni alternative: definizione di tutte le diverse opzioni possibili per ottenere gli obiettivi desiderati

- Valutazione delle alternative: si valuta la probabilità di successo di ogni singola soluzione in base alle esperienze personali o mediante l'uso di tecniche statistiche
- Scelta dell'alternativa migliore: questa fase rappresenta il nucleo del processo decisionale. Dall'analisi del problema, degli obiettivi e delle alternative si seleziona la decisione con le maggiori possibilità di successo
- Attuazione dell'alternativa scelta: la decisione presa viene portata a termine.

### 1.1.2 Approccio basato sulla razionalità limitata

Il concetto di razionalità limitata è associata a processi decisionali “intuitivi”, la cui soluzione si basa sull'esperienza e il giudizio personale piuttosto che sulla logica o il ragionamento. Secondo il concetto di razionalità limitata introdotta da Simon [77] come base per la modellazione matematica del processo decisionale, i decisori sono limitati e ostacolati da una serie di vincoli [33, 69]. Tra questi la limitata capacità della mente umana, l'incertezza e la complessità del problema, la quantità e la disponibilità di informazioni, la criticità della decisione e la tempistica. A causa di questi limiti i decisori raramente ricercano soluzioni ideali che massimizzano i risultati, si limitano invece a scegliere soluzioni in grado di dare opzioni soddisfacenti ma non ottimali. La soluzione a un problema perciò è sempre determinata da ciò che il decisore conosce e ciò che non conosce. In queste condizioni diventa pertanto irrazionale voler prendere una decisione secondo quanto prescritto dai modelli più semplici della teoria della scelta razionale. In base alla teoria della razionalità limitata poiché il fine cui si tende è trovare una soluzione soddisfacente, chi deve risolvere il problema decisionale, stabilisce quella che dovrebbe essere la qualità della soluzione che si aspetta di raggiungere. Ciò in genere è determinato dall'esperienza passata e dal bagaglio cognitivo che il decisore porta con sé. Le aspirazioni dei decisori tendono ad alzarsi quan-

do la ricerca di alternative migliori si rivela sufficientemente semplice e ad abbassarsi qualora la ricerca diventi improduttiva.

## 1.2 Decisioni Organizzative

In relazione al grado di complessità possiamo distinguere le decisioni organizzative in:

- Decisioni strutturate (o programmate): sono impiegate per problemi noti e che si presentano di frequente in cui, le alternative decisionali sono chiare e le loro conseguenze sono prevedibili con elevato grado di accuratezza
- Decisioni non strutturate (o non programmate): si riferiscono a problematiche nuove, in cui le alternative decisionali sono incerte e per le quali l'esperienza accumulata o i metodi decisionali razionali si rilevano inefficaci
- Decisioni semi-strutturate: si riferiscono a situazioni in cui alcune fasi sono strutturate mentre altre non lo sono

La ricerca relativa alle decisioni in ambito organizzativo ha identificato quattro approcci di risoluzione:

- Approccio delle scienze manageriali. Nasce in ambito militare e si basa sull'utilizzo di metodi matematici e statistici. Si può applicare quando:
  - Le decisioni sono definite in modo completo, univoco e logicamente strutturato
  - Le variabili in gioco sono note e misurabili
  - I rapporti causa effetto tra le variabili e il risultato sono chiari

Con l'aiuto delle tecnologie informatiche, le scienze manageriali permettono di trattare problemi che non possono essere trattati dall'uomo per

la loro complessità, per la loro dimensione o per il numero di variabili e vincoli coinvolti

- **Modello Carnegie.** Elaborato da Cyert, March e Simon presso la Carnegie-Mellon University, considera come le decisioni organizzative coinvolgano più attori e siano l'espressione di diverse politiche. Secondo il modello Carnegie, le decisioni organizzative non vengono prese da un singolo decisore ma da un coalizione. Le coalizioni sono unioni temporanee di soggetti che condividono i medesimi obiettivi, la medesima percezione dei problemi e le stesse linee d'azione. Poiché il potere della coalizione è maggiore del potere del singolo esse permettono di raggiungere obiettivi altrimenti irraggiungibili. In generale le coalizioni nascono per due motivi:
  - I soggetti hanno limiti cognitivi e di tempo (razionalità limitata), per cui non dispongono di tutte le risorse (ad esempio informazioni e competenze professionali) per poter delineare in modo sufficiente il problema, le possibili alternative decisionali e gli effetti attesi di tali alternative
  - Gli obiettivi dei soggetti coinvolti nella decisione possono essere contrastanti; l'azione politica dei membri della coalizione permette (tipicamente di negoziazione) di giungere ad un consenso

Il processo decisionale secondo il modello di Carnegie è un processo lungo che coinvolge in discussioni e negoziazioni grandi risorse in termini di tempo e di persone coinvolte; proprio per l'onerosità dell'ottenimento del consenso, la prima alternativa che lo soddisfa viene generalmente adottata e, quando possibile, formalizzata in procedure e politiche che permettono di evitare il ricomponimento della coalizione qualora un problema simile si dovesse presentare. Tuttavia ogni volta che una decisione non programmata si ripresenta, ritorna indispensabile la formazione di una nuova coalizione.

- Modello del processo decisionale incrementatale. Elaborato da Mintzberg, pone la sua attenzione sulla struttura del processo decisionale, quindi sulla sequenza di attività che vengono svolte per identificare il problema e per scegliere una soluzione. Si basa sull'idea che le grandi decisioni sono il frutto della combinazione di tante piccole decisioni. Mintzberg ha evidenziato che, per decisioni complesse il processo decisionale è composto da diversi momenti decisionali pertanto il processo decisionale incrementale può essere diviso in tre fasi:
  - Identificazione: è la fase di definizione e comprensione del problema
  - Sviluppo: vengono individuate possibili soluzioni al problema diagnosticato; tali soluzioni possono essere completamente nuove e progettate per l'occasione, oppure possono essere già state utilizzate ed incorporate in procedure o norme. Il percorso di sviluppo non è univoco; ad esempio, è possibile che in prima battuta venga ricercata una soluzione già implementata in passato che non si rivela interamente applicabile al problema reale e che, dopo alcuni loop di progettazione, possa essere considerata applicabile. La costruzione di una soluzione ai problemi decisionali avviene per cicli di "prove ed errori" e implica a sua volta che vengano prese delle decisioni
  - Selezione: vengono valutate possibili soluzioni e viene scelta quell'alternativa che risolve meglio il problema decisionale

Lo step finale riguarda l'accettazione formale della soluzione scelta. Ogni fase produce dei risultati che alimentano le attività successive e solo raramente il processo decisionale evolve in maniera sequenziale. Generalmente infatti è difficile, e in alcuni casi impossibile, completare una fase e passare alla successiva senza considerare l'effetto che essa produce su altri elementi del processo. Ogni nuovo dato raccolto può suggerire l'esistenza di una nuova alternativa che va valutata attenta-

mente. Esistono dunque diversi punti decisionali in cui il processo può seguire un arresto o l'innescare di un loop. Per tale motivo il processo decisionale può estendersi per orizzonti temporali molto lunghi.

- Il modello del garbage can. Il modello è stato sviluppato per descrivere la struttura del processo decisionale in quelle situazioni che convivono con un elevato grado d'incertezza. Si distingue dagli altri modelli in quanto non riguarda una singola decisione, ma la gestione di decisioni multiple. La caratteristica peculiare del modello consiste nel fatto che il processo decisionale non è visto come una sequenza di passi che inizia con un problema e termina con una soluzione. Il garbage can fa riferimento a quattro concetti chiave:

- Problemi: manifestano uno scostamento tra uno stato desiderato e lo stato attuale. I problemi possono essere affrontati adottando una soluzione oppure no. Nel primo caso non sempre la soluzione proposta risolve il problema in modo soddisfacente
- Soluzioni potenziali: i decisori propongono l'adozione di idee. Tali idee e le motivazioni sottostanti fluiscono e possono essere adottate indipendentemente dall'esistenza di problemi a cui applicarle
- Partecipanti: sono soggetti che prendono parte alla decisione, ciascuno caratterizzato da propri problemi, valori, modalità percettive, soluzioni, sensibilità e cultura
- Opportunità di scelta: circostanze in cui viene presa la decisione

Le conseguenze del modello sono:

- Le soluzioni possono essere proposte anche quando i problemi non esistono
- Le scelte sono fatte senza risolvere i problemi
- I problemi possono persistere senza essere risolti

Secondo il modello garbage can è altrettanto probabile che si prendano decisioni sia a partire dalla soluzione sia a partire dal problema. Il decision making diventa simile a una “pattumiera” nella quale i problemi, le soluzioni e le preferenze di individui e coalizioni si mescolano e si contendono l’azione e l’attenzione dell’organizzazione. La scelta delle alternative dipende da quale coalizione o manager riesce a imporre la propria visione.

In situazioni di incertezza, il modello garbage can spesso si combina con il modello di Carnegie e il modello incrementale.

### 1.3 Processo decisionale in ambito biomedico e clinico

In ambito biomedico e clinico, il processo decisionale è essenzialmente il processo che porta un operatore sanitario a scegliere l’azione da intraprendere tra più alternative in condizioni di incertezza. L’incertezza varia a seconda dei casi. Un paziente, infatti, potrebbe presentare sintomi ben precisi e una storia clinica nota per cui la definizione della diagnosi risulterebbe semplice e immediata oppure, presentare una serie di sintomi riconducibili a condizioni cliniche diverse per cui fare una diagnosi risulterebbe più complicato. Il processo decisionale si configura come una successione di fasi diverse e molto articolate. Tutte le fasi del processo sono caratterizzate da una funzione fondamentale che consiste, essenzialmente, nell’identificare, acquisire, interpretare ed elaborare tutto un corredo di dati dal paziente. Tipicamente, a parte quelle situazioni nelle quali la gravità clinica richiede provvedimenti di emergenza, il rapporto tra medico e paziente inizia con l’anamnesi che consiste nell’identificare e acquisire tutti i sintomi e i dati che potrebbero portare alla formulazione di una diagnosi. Tali dati possono essere riferiti direttamente dal paziente, acquisiti mediante l’esame obiettivo o eventualmente acquisiti da esami strumentali e di laboratorio. Il processo prosegue

con la formulazione della diagnosi dell'eventuale condizione patologica e con la scelta e la pianificazione dell'intervento terapeutico migliore per il suo trattamento; termina con l'analisi e la previsione dell'evoluzione della malattia nel tempo (prognosi) al fine di assumere eventualmente interventi correttivi.

Ciò che risulta cambiato negli ultimi anni riguarda la quantità di informazioni che il medico ha a disposizione e che deve gestire. In generale, è il paziente stesso a fornire più informazioni perché è mediamente più istruito, informato e attento alla propria salute. La diagnostica strumentale e la diagnostica per immagini forniscono inoltre molte più informazioni rispetto al passato. Il tutto è inserito in un contesto caratterizzato dalla necessità di operare opportune assunzioni tenendo conto di inevitabili e scarsamente controllabili fattori d'incertezza. Un aspetto non trascurabile inoltre, è la necessità di adeguare i tempi del processo decisionale alle reali esigenze del paziente e in particolare, alla gravità clinica della situazione e alle priorità di intervento. Ciò che un medico in genere fa è decidere quali informazioni raccogliere, quali test prescrivere, come interpretare e integrare queste informazioni nelle ipotesi diagnostiche. Nella pratica clinica i medici si avvalgono generalmente di linee guida cliniche. La maggior parte di esse sono sviluppate utilizzando un metodo specifico che incorpora i principi della medicina basata sulle prove e delle raccomandazioni espresse da un gruppo di esperti. Molte linee guida definiscono regole di tipo "se, allora" (a esempio, se il paziente ha la febbre, allora utilizzare antipiretici). Le regole più complesse e articolate sono generalmente strutturate sotto forma di algoritmo. Regole e algoritmi sono, di solito, semplici e facili da usare. Le linee guida comunque non tengono conto del grado di incertezza insita nei risultati dei test, della probabilità di successo di un trattamento, dei relativi rischi e benefici di ogni azione. Per incorporare l'incertezza e il valore dei risultati nel processo decisionale clinico, i medici devono spesso applicare principi decisionali quantitativi e analitici. Una delle strategie più usate consiste nella costruzione di un'ipotesi seguita dalla sua verifica. La costruzione dell'ipotesi comporta l'individuazione delle principali possibilità diagnostiche (diagnosi



differenziale) che potrebbero essere la causa del quadro clinico del paziente. A ogni possibile diagnosi è idealmente assegnato un certo grado di probabilità. Il medico inizialmente verifica un'ipotesi già durante l'esame obiettivo e l'anamnesi, ponendo domande o facendo esami specifici che supportino o rifiutino una diagnosi sospetta. Test diagnostici vengono utilizzati quando persistono incertezze dopo l'anamnesi e l'esame obiettivo. I risultati dei test possono modificare ulteriormente le probabilità di diagnosi diverse.

Sebbene esistano dei modelli matematici quantitativi in grado di guidare il processo decisionale clinico, i medici usano raramente calcoli formali per prendere decisioni nella loro pratica giornaliera. Nelle situazioni semplici e/o comuni, spesso prendono le decisioni in maniera informale; la diagnosi scaturisce dal riconoscimento delle caratteristiche del quadro clinico e i test e le terapie vengono stabiliti sulla base della pratica abituale. Per esempio, nel corso di un'epidemia d'influenza, un adulto sano che presenta febbre, dolori diffusi e tosse è probabile che venga considerato come affetto da influenza e che gli venga, quindi, prescritta solo un'appropriata terapia sintomatica. Tale tipologia di analisi e riconoscimento è efficiente e facile da mettere in pratica, ma può essere fonte di errore perché altre possibilità diagnostiche e terapeutiche non vengono, in tal modo, prese in considerazione in maniera seria e sistematica. Per esempio, un paziente con segni e sintomi d'influenza e ridotta saturazione arteriosa di  $O_2$  potrebbe avere una polmonite batterica e richiedere quindi antibiotici. Tale ragionamento informale dunque è spesso fallace e può provocare svariati tipi di errori detti errori cognitivi. Gli errori cognitivi possono essere dovuti principalmente a due fattori: a un'errata valutazione della probabilità di malattia oppure a una mancata seria considerazione delle altre possibili diagnosi. Errori medici possono essere causati inoltre dalla grande quantità di dati che gli operatori sanitari devono elaborare. Infatti, sebbene essi siano in grado di assumere decisioni complesse e di valutare, in termini qualitativi, i benefici e i rischi delle varie opzioni nelle situazioni semplici o comuni, spesso la quantità dei dati e delle informazioni che devono elaborare eccede le loro capacità cognitive. Sistemi di supporto

alle decisioni specifici possono aiutare a ridurre tali errori e migliorare di conseguenza la qualità delle cure e l'appropriatezza degli interventi.

L'approfondita analisi del generale contesto medico-clinico porta, tipicamente, ad interpretare le decisioni medico-cliniche tramite uno schema logico del tipo "causa-effetto". In tal senso, il processo decisionale medico-clinico può essere rappresentato e formalizzato attraverso l'integrazione di due schemi inferenziali:

- Deduttivo: basato sulla conoscenza dei risultati scientifici prodotti, dei principi biomedici e clinici su cui si basano i fenomeni biologici, delle relazioni funzionali tra le varie condizioni fisio-patologiche
- Induttivo: basato sull'esperienza e sulla pratica clinica, sulla conoscenza indotta ed estratta dalla casistica

Entrambi sono facilmente implementabili in sistemi computerizzati.



## Capitolo 2

# Sistemi di Supporto alle Decisioni Cliniche

In generale, un Sistema di Supporto alle Decisioni (Decision Support System - DSS) è una complessa piattaforma informatica interattiva che supporta l'utente finale (tipicamente il decisore del contesto di riferimento) durante il processo decisionale, integrando ed elaborando dati e modelli computazionali. Concettualmente un DSS può essere concepito come un'articolata architettura che integra al suo interno conoscenza di dominio opportunamente codificata, dati, modelli matematici e computazionali. Come mostrato in Figura 2.1, un DSS è un sistema che in base alle informazioni acquisite in ingresso è in grado restituire un output dato dalla opportuna combinazione di tutti questi elementi. L'output può essere una semplice informazione sul dominio di riferimento, un avviso, un promemoria o il suggerimento di una scelta da prendere tra tante possibili alternative.

Il disegno e lo sviluppo di un DSS nasce dalla definizione e organizzazione dello specifico processo decisionale che si intende supportare. In particolare, la determinazione dell'insieme di servizi e funzionalità che il DSS dovrà erogare si basa sull'approfondita analisi del contesto di riferimento e dell'articolazione dei complessivi flussi di attività che vi si svolgono, con specifica attenzione alle attività di natura decisionale. In tal senso, l'applicazione e

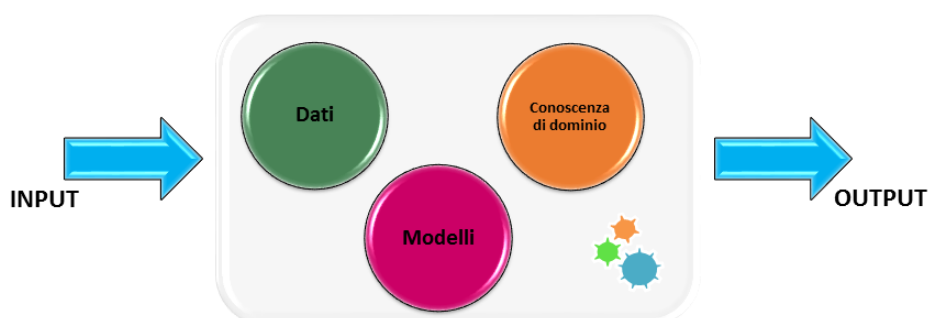


Figura 2.1: Sistema di Supporto alle Decisioni - DSS

l'integrazione di un appropriato DSS al contesto di riferimento può e deve contribuire a migliorare lo svolgimento dei flussi di attività e le prestazioni complessive in relazione alla qualità dei servizi resi.

Un sistema di supporto alle decisioni cliniche (Clinical Decision Support Systems - CDSS) fornisce supporto decisionale agli operatori sanitari in tutte le procedure medico-cliniche. L'obiettivo complessivo riguarda, ovviamente, la possibilità di incidere sostanzialmente nel migliorare la qualità dell'assistenza e della cura dei pazienti riducendo il più possibile gli errori e la scarsa appropriatezza degli interventi. In generale, un CDSS assiste gli operatori sanitari nelle loro attività decisionali fornendo un insieme organizzato di strumenti con cui è possibile affrontare la soluzione del problema decisionale. Fornisce inoltre, conoscenze cliniche e informazioni relative ai pazienti, intelligentemente filtrate e presentate al momento opportuno.

Numerose sono le attività cliniche in cui un CDSS può essere utilizzato. L'elenco che segue mostra alcune delle applicazioni più comuni:

- Diagnosi precoci, previsioni di eventi acuti e di peggioramento delle condizioni cliniche di un paziente
- Valutazione prognostica, inquadramento del rischio, stadiazione della malattia e delle condizioni di severità
- Pianificazione e gestione della terapia, con personalizzazione dei trattamenti

- Gestione del follow-up
- Supporto al “patient empowerment”
- Alerts/Reminder: messaggi unidirezionali basati su acquisizione ed elaborazione automatica di dati (ad esempio in ambienti di monitoraggio in remoto)
- Consultazione Interattiva:
  - Dialogo con l’utente
  - Raccomandazioni sulla base di specifiche richieste
  - Analisi/esplorazione di dati
- Supporto Clinico:
  - Indicazioni sull’applicazione di linee guide e protocolli clinici
  - Aggiornamenti periodici sulle condizioni del paziente e sulla progressione della malattia
  - Consultazione specializzata in ambienti con scarse risorse mediche o non clinicamente controllati

## 2.1 Tipologie di Sistemi di Supporto alle Decisioni

In letteratura i CDSS sono classificati in due categorie principali: sistemi basati sulla conoscenza (knowledge-based CDSS) e sistemi non basati sulla conoscenza (nonknowledge-based CDSS) [9].

### 2.1.1 Knowledge Based CDSS

I sistemi knowledge-based rappresentano la conoscenza e i fatti di un dominio clinico attraverso un linguaggio di descrizione della conoscenza (detto

anche vocabolario) comprensibile e sfruttabile dai sistemi informatici [13]. Il vocabolario viene utilizzato da un sistema di ragionamento automatico, che nella maggior parte dei casi, usa regole di tipo “IF-THEN”, per fare inferenza. I sistemi basati sulla conoscenza sono generalmente composti da tre elementi principali: la Base di Conoscenza, il Motore di Inferenza ed il meccanismo che gli permette di comunicare con l’utente (Interfaccia Utente) [9]. La Figura 2.2 mostra le interazioni tra questi componenti.

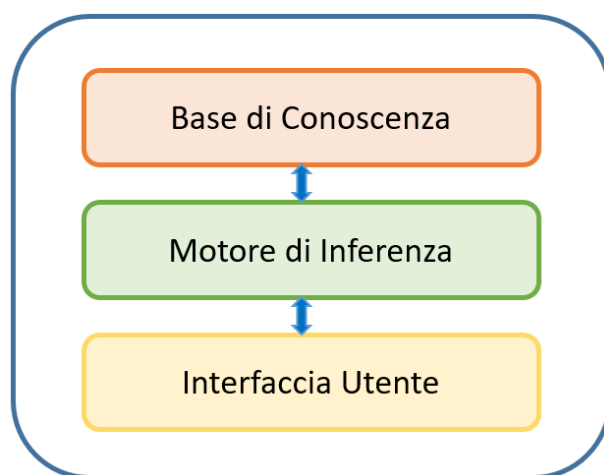


Figura 2.2: Schema architetturale di un Knowledge Base CDSS

La Base di Conoscenza contiene informazioni opportunamente codificate del dominio clinico di riferimento. Il Motore di Inferenza combina le regole della base di conoscenza con i dati del paziente. L’Interfaccia Utente gestisce la comunicazione con l’utente finale. Permette al sistema di mostrare i risultati all’utente e di ricevere da questi i dati su cui effettuare le valutazioni. Sono di tipo knowledge-based i CDSS basati su ontologie, linee guida cliniche e logica Fuzzy.

- **CDSS basati su ontologie.** Un’ontologia è una rappresentazione formale di un certo dominio in un formato che è comprensibile da sistemi computerizzati. Essa descrive un insieme di concetti (come ad esempio, pazienti, malattie, terapie) e le relazioni che tra questi intercorrono. In

[68] viene presentato un CDSS basato su ontologia per la gestione dei pazienti affetti da malattie croniche. In base alla malattia cronica da cui il paziente è affetto, una prima serie di regole seleziona la parte di ontologia che si riferisce alla malattia in esame mentre una seconda serie di regole usa questa sotto-ontologia per adattare piani di intervento che descrivono trattamenti generali allo specifico paziente.

- **CDSS basati su linee guida.** I sistemi basati sulla conoscenza sono particolarmente adatti allo sviluppo di linee guida cliniche automatizzate in quanto implementano in un linguaggio di programmazione la conoscenza da esse estratta. Choi et al. [18] hanno sviluppato un CDSS basato su linea guida per supportare gli operatori sanitari nello screening e nella gestione della depressione. Un approccio basato su linee guida è usato anche in [58] per migliorare la gestione dei pazienti affetti da patologie multiple. Il sistema è in grado di fornire informazioni estratte dalle linee guida attraverso una piattaforma web ed è direttamente integrato con le informazioni dei pazienti contenute in cartelle cliniche elettroniche.
- **CDSS basati su logica Fuzzy.** La logica Fuzzy [64] è un metodo probabilistico che cerca di emulare il ragionamento umano in quelle applicazioni reali dove il ragionamento è approssimato a causa di informazioni imprecise o mancanti. Al contrario di altri metodi in cui le variabili di uscita possono assumere solo due valori, ad esempio “vero” o “falso”, le variabili nella logica fuzzy possono avere diversi “gradi di verità”, con un peso che indica la forza della connessione. In [30] hanno utilizzato la Logica Fuzzy per valutare lo stato di salute delle persone affette da sclerosi multipla. Gli autori hanno sviluppato regole di tipo IF-THEN per valutare la normalità/anormalità dei tessuti cerebrali sulla base di caratteristiche che possono assumere valori in un set di valori possibili.



### 2.1.2 Non Knowledge Based CDSS

I CDSSs non basati sulla conoscenza implementano con metodologie inferenziali statistiche e metodi di ML l'esperienza diretta e la conoscenza inductivamente estraibile dalle casistiche retrospettive. In particolare, esse prevedono una fase di addestramento iniziale, su un set di dati composto da dati rappresentativi del paziente (ad esempio sintomi o risultati di laboratorio e informazioni sulla presenza o l'assenza della malattia) in base ai quali, il sistema apprende come completare un compito specifico e, una fase in cui il sistema analizza nuovi casi per fare inferenza (ad esempio prevedono lo stato di salute del paziente o se una certa malattia è presente o meno).

In generale, non esiste un metodo di apprendimento perfetto che si comporta bene su tutti gli insiemi di dati. Può accadere infatti, che per un dato problema clinico, un certo metodo ottenga un'elevata accuratezza diagnostica mentre un altro metodo fallisca completamente.

Nella ML, l'apprendimento può essere di due tipi: supervisionato e non supervisionato. Nell'apprendimento supervisionato il sistema viene addestrato con un insieme di esempi di cui si conosce l'etichetta della variabile di output, nell'apprendimento non supervisionato invece, non c'è una variabile di risposta a "supervisionare" il processo di apprendimento e l'obiettivo è quello di trovare strutture significative e modelli descrittivi dagli esempi analizzati. Metodi quali le Support Vector Machine e le Reti Neurali, sono metodi di apprendimento supervisionato mentre le tecniche di Clustering sono basate su un apprendimento non supervisionato. Esiste anche un apprendimento semi-supervisionato che è dato dalla combinazione di elementi delle due tecniche di apprendimento supervisionato e non supervisionato.

I concetti alla base dei principali metodi di apprendimento supervisionato sono illustrati nel Capitolo 4.

## 2.2 Architettura dei CDSS

L'architettura software è l'organizzazione di base di un sistema. Essa descrive le componenti del sistema, le relazioni che intercorrono tra di esse e che intercorrono tra il sistema e l'ambiente esterno. Definisce inoltre, i principi che ne guidano il progetto e l'evoluzione. Lo scopo principale è quello di scomporre il sistema in sottosistemi e quindi progettare e realizzare più componenti distinte che interagiscono tra loro. Shaw and Garlan [75] descrivono infatti, l'architettura software di un sistema complesso come un insieme di componenti computazionali (o semplicemente componenti) unitamente alla descrizione delle interazioni tra questi componenti (i connettori). L'architettura software permette ai progettisti di ragionare circa le capacità che un sistema deve fornire per soddisfare determinati requisiti e di conseguenza suggerisce un progetto per la costruzione e la composizione del sistema [8, 36].

Generalmente, un'architettura software viene descritta attraverso un Linguaggio di Descrizione delle Architetture Software (Architecture Description Languages - ADL). Esistono diversi ADL quali ad esempio AADL, ACME, Rapide, Wright [12]. Un ADS è un linguaggio formale (di tipo grafico, testuale o entrambi) di supporto alla modellazione di architetture software. Anche se esistono decine di ADL, la loro popolarità e il loro utilizzo da parte di professionisti è molto bassa. Di solito, si preferisce descrivere l'architettura software attraverso il linguaggio UML (Unified Modeling Language), che è diventato standard OMG e offre una grande varietà di concetti per la definizione della struttura e il comportamento atteso di un sistema software. Anche se è non pienamente qualificato come ADL, l'UML è lo standard industriale de facto. In letteratura si possono trovare esempi di CDSS basati su organizzazioni architettonali differenti come in El-Sappagh and El-Masri [28] dove è proposta un'architettura distribuita basata su Service Oriented Architecture che è in grado di usare i dati provenienti dal Fascicolo Sanitario Elettronico (FSE) e di interagire con diversi database clinici e basi di conoscenza.

Per quanto riguarda l'interazione con l'ambiente esterno, i primi CDSS, come quello di de Dombal et al. [22] per la diagnosi differenziale del dolore addominale, erano tutti sistemi stand-alone, ovvero non avevano interazioni con altri componenti esterni al sistema. I loro utenti dovevano accedere al sistema attraverso la propria interfaccia di front-end e inserire i dati del paziente in esame. Negli anni, con l'evoluzione delle tecnologie informatiche anche le architetture e le metodologie di progettazione e implementazione dei CDSS si sono evolute di conseguenza. Da subito i CDSS sono stati integrati con i sistemi informativi delle aziende sanitarie che li utilizzavano. Tra questi, il sistema HELP [78] integrato nel flusso di lavoro (workflow) dell'LDS Hospital di Salt Lake City, Utah. Vista la difficoltà di comunicazione tra una struttura e l'altra dovuta alla mancanza di dati e informazioni cliniche standardizzate, tali sistemi erano comunque limitati all'utilizzo da parte di una singola organizzazione sanitaria. Solo con l'avvento di organismi di standardizzazione quali HL7 ad esempio, sono stati sviluppati CDSS in grado di comunicare e condividere la conoscenza con altri sistemi. SEBASTIAN [47] e SAGE [81] sono stati i primi CDSS orientati ai servizi e basati su architetture distribuite. Un'architettura distribuita, prevede che l'elaborazione delle informazioni sia distribuita su più calcolatori anziché centralizzata su una singola macchina. Architetture software distribuite più recenti per CDSS sono state proposte in [93] e [27]. Negli ultimi anni, l'uso di un'architettura orientata ai servizi (Service Oriented Architecture - SOA) è stata identificata come uno tra gli approcci più promettenti per migliorare l'assistenza sanitaria e agevolare l'uso di CDSS. L'iniziativa OpenCDS [5] attualmente sta lavorando su una serie di strumenti open source per la condivisione di CDSS tramite servizi.

### 2.2.1 Approccio Service Oriented Architecture

L'approccio SOA (Service Oriented Architecture), può essere sintetizzato intuitivamente nell'idea che la collaborazione tra i soggetti (utenti, sistemi, organizzazioni) può e deve essere pensata come scambio di servizi. Un ser-

vizio è un'attività svolta da un soggetto erogatore, produttiva di risultati che hanno valore d'uso per un altro soggetto (il fruitore del servizio), eseguita secondo le modalità definite da un contratto di servizio sottoscritto da entrambe le parti. Il contratto di servizio definisce:

1. Le funzionalità del servizio, cioè cosa l'erogatore fa per il fruitore
2. Le modalità di interazione tra erogatore e fruitore
3. Le caratteristiche prestazionali del servizio, ad esempio performance, affidabilità e qualità dei dati
4. Le esigenze e i vincoli di sicurezza del servizio come autenticazione e autorizzazione.

Generalmente, l'approccio SOA non è legato ad alcuna specifica tecnologia ma definisce un insieme di proprietà che i servizi devono avere per poter collaborare in un ambiente eterogeneo. Principalmente, un servizio:

- deve poter essere ricercato e recuperato in maniera dinamica attraverso la sua interfaccia
- deve essere indipendente dal contesto o dallo stato di altri servizi
- deve essere indipendente dall'implementazione, infatti non è necessario conoscere come un servizio è stato realizzato ma solo quali sono le funzionalità che mette a disposizione
- l'accesso deve avvenire in maniera indipendente rispetto all'allocazione del servizio

L'approccio SOA si basa su tre entità che interagiscono tra loro:

- Service Provider
- Service Consumer
- Service Registry

Il Service Provider è l'entità che implementa il servizio e lo rende disponibile. Il Service Registry possiede informazioni, quali URL e modalità di accesso, su tutti i servizi disponibili. Il Service Consumer è l'entità che utilizzerà il servizio. La Figura 2.3 riporta le interazioni tra queste entità. Un servizio, per poter essere trovato da altre entità che vogliono utilizzarlo, deve essere reso visibile sulla rete. A tal fine, il Service Provider comunica al Service Registry le informazioni relative al servizio affinché vengano memorizzate. Nel momento in cui un Service Consumer dovrà utilizzare un servizio farà richiesta al Service Registry delle informazioni ad esso relative. Con queste informazioni il Service Consumer potrà comunicare direttamente con il Service Provider ed utilizzare il servizio richiesto.

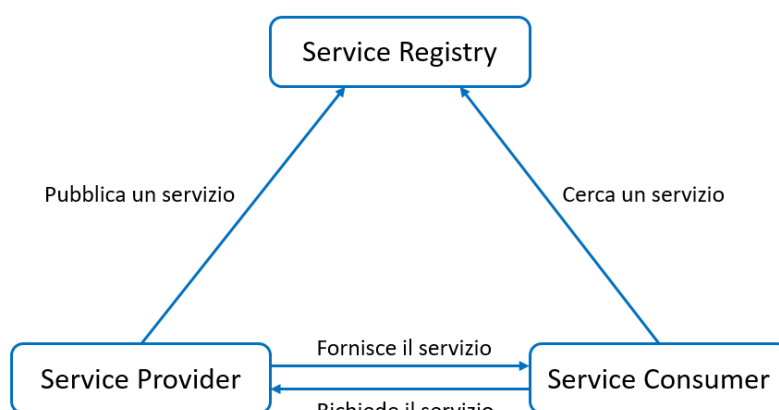


Figura 2.3: Esempio di architettura SOA

Per un CDSS, i vantaggi di un approccio SOA sono diversi:

- **Accesso a una conoscenza più ampia:** un CDSS può avere l'accesso e dunque utilizzare più basi di conoscenza o diverse basi di dati da cui estrarre conoscenza. Organizzazioni specializzate in grado di sviluppare e mantenere linee guida cliniche o casistiche di dati reali possono esporle come servizi alle organizzazioni sanitarie e ai CDSS che ne faranno richiesta

- **Riduzione dei costi:** SOA consente la riduzione dei costi grazie al riutilizzo di servizi CDSS esistenti, riducendo al minimo il tempo di sviluppo del software e semplificandone la manutenzione
- **Accesso più veloce alle informazioni:** SOA consente l'implementazione di nuove funzionalità per i CDSS in modo veloce e trasparente all'utente finale

Oltre all'approccio generale basato su service registry, esistono altri approcci architetturali per realizzare sistemi basati su SOA. In Loya et al. [57] sono stati individuate diverse architetture generali per lo sviluppo di CDSS basati su tale approccio. Di seguito un elenco delle architetture più utilizzate per lo sviluppo di tali sistemi:

- point to point
- enterprise service bus (ESB)
- service registry
- coreography
- orchestration

La comunicazione di tipo point to point è l'approccio più utilizzato nello sviluppo di CDSS basati su SOA. Utilizzando questo approccio di comunicazione, ogni interazione tra le applicazioni (ad esempio, tra un erogatore di servizi e un fruitore di servizi) viene progettata, realizzata e gestita in maniera personalizzata.

L'ESB è un approccio architetturale che permette la comunicazione tra applicazioni attraverso un bus in grado di acquisire dati in qualsiasi formato e da qualsiasi fonte e instradarli verso le applicazioni appropriate. In contrasto con la comunicazione point-to-point, con questo tipo di architettura i messaggi passano attraverso l'ESB che funge da intermediario tra il fornitore del servizio e l'utilizzatore del servizio.

L'architettura basata su service registry è l'architettura classica precedentemente descritta. Il service registry è un componente architetturale che memorizza le informazioni relative a ciascun servizio (ad esempio, la descrizione, le politiche di accesso, l'url, le versioni) e pertanto i fruitori possono trovare i servizi che soddisfano i loro requisiti interrogando semplicemente il registro.

Nelle configurazioni basate su coreografia, la coreografia definisce come ogni servizio interagisce con gli altri. Definisce cosa viene trasmesso e quando viene trasmesso, le attività e gli obblighi tra le parti. Tutti i partecipanti sono posti allo stesso livello favorendo quindi una maggior propensione alla collaborazione.

In una configurazione basata su orchestrazione invece, c'è un solo servizio che gestisce tutti gli altri definendo le modalità delle loro interazioni. Non tutti i partecipanti all'orchestrazione hanno dunque la stessa importanza in quanto il processo di interazione è controllato da uno solo dei partecipanti.

La maggior parte dei CDSS basati su SOA usano una configurazione basata su coreografia.

## 2.3 Principali CDSS in letteratura

Gli sforzi per automatizzare l'assistenza sanitaria hanno avuto inizio alla fine degli anni '50. Sebbene i primi tentativi di offrire servizi sanitari attraverso l'uso di sistemi computerizzati sono apparsi alla fine del 1950 [54], i primi veri e propri prototipi di CDSS sono stati implementati e utilizzati nella pratica clinica a partire dal 1970 [22, 35, 76]. Inizialmente essi affrontavano solo tipi semplici di decisioni cliniche quali ad esempio, riconoscere se il risultato di un test di laboratorio è fuori dal range normale di valori, o se un farmaco ha una pericolosa interazione con un altro. Il loro impatto sulla pratica clinica non è stato da subito quello previsto. I problemi che hanno impedito l'introduzione e la diffusione di tali sistemi sono stati diversi, dalle limitazioni delle basi scientifiche a disposizione, alle difficoltà logistiche che

gli sviluppatori incontrarono nel tentativo di incoraggiare i medici a utilizzare e accettare sistemi che non erano ben integrati nel loro normale flusso di lavoro.

Sistemi di supporto alle decisioni cliniche introdotti negli anni '70 sono: AAHELP [22], un sistema di supporto alla diagnosi differenziale del dolore addominale acuto e basato sul ragionamento bayesiano; INTERNIST-I [61] un sistema di supporto alla diagnosi nell'ambito della medicina generale interna (1974) e MYCIN [76], un sistema esperto basato su regole per la diagnosi e il trattamento delle infezioni nel sangue.

Negli anni '80 sono stati commercializzati invece sistemi di supporto alle decisioni di successo come DXplain [34] e QMR [61]. DXplain è un sistema di supporto alle decisioni basato sul ragionamento bayesiano che produce una sorta di graduatoria di tutte le possibili diagnosi che potrebbero spiegare (o essere associate) determinate manifestazioni cliniche. Per ciascuna diagnosi fornisce inoltre, il motivo per il quale dovrebbe essere considerata. QMR è un sistema di supporto decisionale di tipo diagnostico, con una vasta base di conoscenza che contiene informazioni su diverse malattie, sulle caratteristiche di ognuna e sulle correlazioni che tra queste possono esistere.

L'avvento del World Wide Web alla fine degli anni '90 ha cambiato radicalmente le organizzazioni sanitarie e di conseguenza anche l'architettura dei sistemi usati in questo contesto. Per la prima volta, infatti, si è iniziato a pensare di poter accedere a una variegata collezione di risorse informative in modo uniforme utilizzando un comune browser. Sono nati così i primi CDSS ai quali si poteva accedere tramite il web. È degli anni '90 un sistema che supporta i medici nella prescrizione di antibiotici e altri farmaci per la cura delle infezioni [70].

Molti CDSS sviluppati e commercializzati nei primi anni 2000 sono ancora in uso. Tra questi LISA [45] per la diagnosi della leucemia linfoblastica acuta infantile e RETROGRAM [82] per la diagnosi e il trattamento dell'HIV. Entrambi sono attualmente utilizzati nella pratica clinica e sono in continuo sviluppo e aggiornamento. ATHENA (Assessment and Treatment



of Hypertension: Evidence-Based Automation) invece, è un CDSS sviluppato per supportare gli operatori sanitari nella gestione dell'ipertensione nell'ambito della medicina di base [40]. Il sistema, basato su regole, implementa le linee guida cliniche del settore. In [32] uno dei primi CDSS di tipo interattivo. Esso è in grado di interagire con gli utenti e apprendere automaticamente nuove conoscenze dalle esperienze di quest'ultimi. La sua base di conoscenza risulta essere dinamica e auto evolutiva, contrariamente a quanto avviene nei CDSS convenzionali che utilizzano di solito una base statica di conoscenze e che trascurano i feedback dai pazienti. Anche in [60] è stato presentato un sistema di supporto che facilita la comunicazione tra infermieri, team di pronto intervento e medici attraverso un'interfaccia web. Il motore computazionale combina tecniche di ML e regole decisionali.

Nel corso degli anni, molti sono i CDSS sviluppati per supportare gli operatori sanitari nella scelta e definizione dei trattamenti terapeutici. La somministrazione di farmaci è, infatti, una delle pratiche cliniche di routine più comuni per i medici ed è fondamentale per la salute dei pazienti. Questa attività comprende un processo decisionale in cui il medico deve decidere la dose e la frequenza delle somministrazioni di un farmaco sia sulla base delle caratteristiche dei pazienti che della propria esperienza clinica. I CDSS aiutano a rendere questo processo più rapido, poco costoso, non invasivo e molto più accurato. Tra gli altri si ricorda il Drug Administration Decision Support System (DADSS) [92]. Il CDSS aiuta medici e infermieri a calcolare la dose di farmaco ottimale da prescrivere per il trattamento di una certa malattia. Il sistema è basato su tecniche di ML e in particolare su Support Vector Machine. Diverse applicazioni sanitarie inoltre, sono state sviluppate per il monitoraggio dei pazienti nel loro ambiente domestico. Lavori recenti mostrano che la maggior parte delle applicazioni in tale contesto forniscono ai pazienti delle linee guida convenzionali e alcuni promemoria per quanto riguarda i farmaci da prendere e le attività fisiche da seguire. Rientra in questa categoria Smart CDSS [44], un sistema intelligente in grado di fornire ai pazienti consigli in base al loro stato di salute giornaliero.

Nell'attuale contesto clinico e sanitario, sempre più spesso i CDSS vengono messi a disposizione degli utilizzatori attraverso applicazioni su smartphone. Il sistema chiamato PTT (Partial Thromboplastin Time) Advisor [74] è tra questi. Esso supporta i medici nella gestione del follow-up di pazienti con PTT prolungato.

Altri CDSS sviluppati di recente sono: [83] e [39]. Il CDSS in [83] supporta gli utenti nella definizione, nella gestione e nella valutazione di studi clinici attraverso la creazione di modelli rischio-beneficio. In [39] viene presentato un sistema basato su tecniche di ML per il monitoraggio e la diagnosi delle infezioni che si possono facilmente contrarre in un contesto ospedaliero. Il sistema usa diverse tecniche di apprendimento automatico al fine di:

- estrarre automaticamente la conoscenza contenuta in documenti clinici sia strutturati che non strutturati
- integrare la conoscenza degli operatori del settore
- generare dinamicamente nuove conoscenze fornendo una motivazione semplice e comprensibile per ciascuna decisione suggerita

## 2.4 Progettazione e sviluppo di CDSS

La progettazione di un CDSS richiede una forte interazione tra i progettisti e i responsabili decisionali (decision maker). L'obiettivo finale di entrambe le figure professionali è quello di formalizzare la conoscenza, i meccanismi e i processi decisionali. I decision maker devono identificare e definire con precisione la propria esperienza e competenza, che è fatta di conoscenza del dominio, regole euristiche, meccanismi di ragionamento ed espedienti di semplificazione. I progettisti, invece, hanno il compito di comprendere tali conoscenze mediante un colloquio continuo con i decision maker e trovare la formalizzazione più adatta ad esse. Poichè, è generalmente riconosciuto che un CDSS non è mai completo in quanto la conoscenza ed il modello di base non lo sono mai, è sempre possibile migliorarlo. Il sistema inoltre, può essere

soggetto a errori, quindi è necessario aggiungere man mano nuove conoscenze e modelli per migliorare le sue prestazioni.

Le fasi di progettazione e sviluppo di un CDSS seguono generalmente un processo incrementale come mostrato in Figura 2.4. L'evoluzione del sistema segue un processo costituito da continui perfezionamenti e revisioni, riformulazione di concetti e modelli e riorganizzazione delle conoscenze che porta ai suoi raffinamenti successivi. Le fasi principali del processo sono le seguenti:

- **Identificazione:** è finalizzata all'analisi e rappresentazione del processo decisionale medico-clinico che si vuole supportare e dunque, all'individuazione degli obiettivi generali (requisiti) del CDSS
- **Concettualizzazione:** è incentrata: (1) sull'identificazione delle capacità generali (funzionalità) che il CDSS deve avere per soddisfare i requisiti, (2) sulla caratterizzazione dei concetti chiave e delle relazioni tra gli elementi di conoscenza definiti nella prima fase, (3) nell'identificazione dei moduli indipendenti che devono essere sviluppati per affrontare il problema
- **Formalizzazione:** è dedicata alla selezione degli strumenti necessari per sviluppare le funzionalità del CDSS. In particolare, è la fase dedicata alla progettazione e allo sviluppo dei modelli decisionali basati sulla conoscenza del dominio medico-clinico, alla formalizzazione della conoscenza di dominio, al disegno della strategia di controllo
- **Attuazione:** è la fase di sviluppo delle componenti del CDSS
- **Test:** è la fase in cui si valutano le prestazioni del sistema

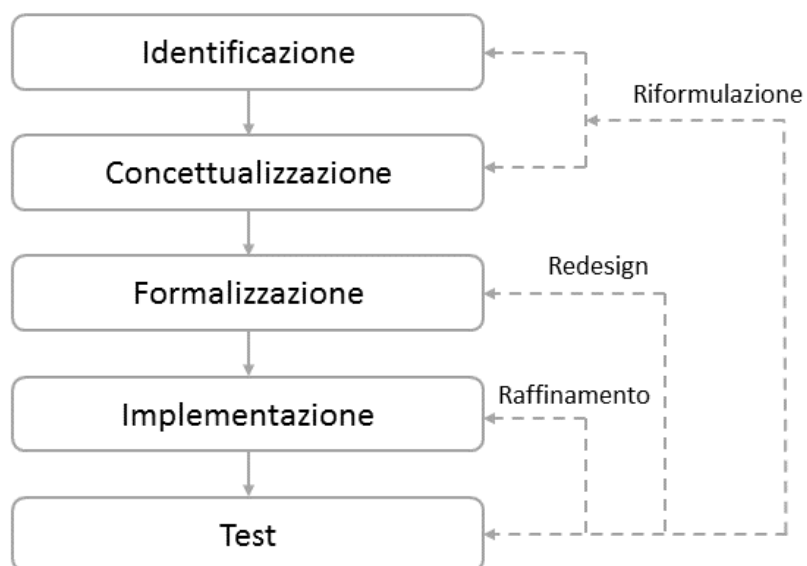


Figura 2.4: Fasi di progettazione e sviluppo di un CDSS



## Capitolo 3

# Standard di interoperabilità

Importante nello sviluppo di CDSS è l'interoperabilità applicativa e semantica con altri sistemi del settore sanitario. A prescindere dal tipo di supporto decisionale, un CDSS è un sistema che richiede l'accesso tempestivo a dati e conoscenze mediche memorizzati in altri sistemi e di conseguenza, questo implica una certa interazione con gli altri componenti del sistema sanitario. Problemi di integrazione e di interoperabilità si sono iniziati ad affrontare e risolvere già dalla fine degli anni '90, con l'introduzione progressiva di standard tecnologici quali DICOM (Digital Imaging and COmmunications in Medicine) e con la sempre maggiore conformità a profili di integrazione dettati da iniziative quali IHE (Integrating Healthcare Enterprises) [2]. Al contrario dei primi standard che proponevano sempre casi d'uso specifici, sistemi e transazioni precise e non generalizzabili, gli standard definiti negli ultimi anni sono orientati ai servizi e sono basati sui concetti di cooperazione applicativa e interoperabilità semantica. A livello internazionale, gli organismi di standardizzazione principali sono HL7 (Health Level Seven International) e OMG (Object Management Group). Entrambe hanno affrontato il problema generale dell'interoperabilità nel settore socio-sanitario in modo innovativo nel progetto internazionale HSSP (Healthcare Services Specification Program).

### 3.1 HL7

HL7 (Health Level Seven International) [1] è un'organizzazione non-profit, fondata nel 1987 accreditata allo sviluppo di standard ANSI (American National Standards Institute) e dedicata a fornire un framework e un insieme di standard per lo scambio, l'integrazione, e la condivisione di tutte quelle informazioni a sostegno alla pratica clinica e alla gestione, all'erogazione e alla valutazione dei servizi sanitari. L'obiettivo principale di HL7 è quello di creare standard per il settore sanitario che consentano di aumentare l'interoperabilità tra i sistemi così da migliorare la qualità della cura, ottimizzare il flusso di lavoro, ridurre l'ambiguità e migliorare il trasferimento delle conoscenze tra tutti gli stakeholder del settore, compresi i fornitori dell'assistenza sanitaria, le agenzie governative, le comunità dei fornitori, i pazienti e gli operatori sanitari (medici, specialisti, ecc.). "Level Seven" si riferisce al settimo livello (Application layer) del modello standard ISO (International Organization for Standardization) / OSI (Open Systems Interconnection OSI) [31], uno standard nato alla fine degli anni '70 che propone un modello di riferimento per l'interconnessione di sistemi aperti. Gli standard di HL7 sono raggruppati in sette differenti categorie (o sezioni) di riferimento:

- **Sezione 1: Primary Standards.** Standard utilizzati per l'integrazione, l'interoperabilità e la conformità dei sistemi. Gli standard HL7 più utilizzati appartengono a questa categoria
- **Sezione 2: Foundational Standards.** Gli standard di questa categoria definiscono gli strumenti fondamentali e i blocchi costruttivi che gli implementatori devono gestire per creare i loro artefatti
- **Sezione 3: Clinical and Administrative Domains.** Standard per la messaggistica, la memorizzazione e la gestione di documenti clinici. Tali standard sono generalmente implementati dopo che i Primary Standards sono stati attuati dall'organizzazione che li intende utilizzare

- **Sezione 4: EHR Profiles.** Gli standard che fanno parte di questa categoria forniscono modelli funzionali e profili che consentono di gestire le cartelle cliniche elettroniche (electronic health record - EHR)
- **Sezione 5: Implementation Guides.** Contiene le linee guida per l'implementazione di standard esistenti
- **Sezione 6: Rules and References.** Specifiche tecniche, strutture di programmazione e linee guida per lo sviluppo di standard e di software
- **Sezione 7: Education & Awareness.** Questa sezione, contiene gli standard che sono ancora in versione bozza, i cosiddetti Draft Standards for Trial Use (DSTU), i progetti correnti di HL7 e tutte quelle risorse e strumenti utili per l'approfondimento, l'utilizzo e l'adozione degli standard HL7

In generale, HL7 gestisce l'intero ciclo di vita di definizione di uno standard, tra cui lo sviluppo, l'adozione, il riconoscimento del mercato, l'utilizzo e l'aderenza. L'uso commerciale degli standard HL7 richiede un abbonamento di membership all'associazione. Solo i membri possono accedere gratuitamente agli standard. Tutti gli altri possono acquistarli direttamente da HL7 o da ANSI. HL7 ha una rete di affiliati in tutto il mondo. Essi sono formalmente i responsabili della localizzazione dello standard. La sezione italiana di HL7 si è formata nel 2003 ed è responsabile della localizzazione degli standard nella realtà italiana.

### 3.1.1 HL7 Clinical Document Architecture

Clinical Document Architecture (CDA) è uno standard ANSI di markup sviluppato da HL7 per definire la struttura dei documenti clinici, quali ad esempio le lettere di dimissione ospedaliera o referti di pronto soccorso. Ciascun documento in formato standard CDA è costituito da una parte testuale obbligatoria (che garantisce l'interpretazione umana del contenuto del documento) e parti strutturate opzionali (per l'elaborazione software). La parte



strutturata si basa su sistemi di codifica standard (come SNOMED-CT (Systematized Nomenclature of Medicine – Clinical Terms) [6] e LOINC (Logical Observation Identifiers Names and Codes) [3]) per rappresentare i concetti. Lo standard garantisce che i dati dei pazienti possono essere creati e letti da qualsiasi EHR (Electronic Health Record) del sistema sanitario. CDA specifica dunque la sintassi e fornisce un quadro di riferimento per definire nel dettaglio la semantica di un documento clinico.

### 3.1.2 HL7 Decision Support Service

Lo standard HL7 DSS (Decision Support Service) è uno standard di HL7 che si basa sulla normativa OMG DSS definita nell'ambito del progetto HSSP. Definisce un'interfaccia standard di accesso e di gestione di applicazioni di supporto alla decisione, indipendentemente dal formato dei dati, dai meccanismi di inferenza e dalle tecnologie d'implementazione di tali applicazioni. In pratica, sottoponendo a un'applicazione i dati del paziente in formato standard e attraverso l'interfaccia del servizio HL7 DSS, è possibile ottenere, sempre attraverso tale interfaccia, le indicazioni diagnostiche e le prescrizioni terapeutiche fornite dall'applicazione per quel paziente anch'esse in un formato standard. Un servizio HL7 DSS può essere concettualmente inteso come il custode di uno o più CDSS. Un client del servizio, quando richiede la valutazione dei dati di un paziente, specifica il CDSS (o i CDSS) che intende utilizzare e invia i dati che intende valutare. Ovviamente, il client è in grado di conoscere, attraverso opportune funzioni che il servizio mette a disposizione, i CDSS disponibili in consultazione, i dati (e il formato) richiesti da ciascuno di essi nonché il significato dei dati che saranno restituiti come risultato delle valutazioni. La Figura 3.1 mostra un esempio di interazione tra un applicazione client e un provider del servizio.

Quando un'applicazione client ha bisogno di valutare i dati di uno specifico paziente, recupera i dati dai propri archivi, li fornisce al servizio e richiede che essi vengano valutati utilizzando i CDSS specificati. A valutazione ultimata il client riceve i risultati prodotti dai CDSS specificati ed è così in grado

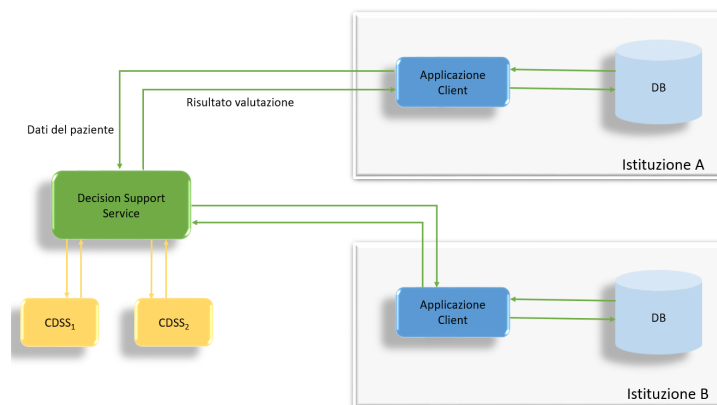


Figura 3.1: Interazione tra un client e un CDSS attraverso il servizio standard HL7 DSS

di analizzare e utilizzare i risultati ottenuti. La realizzazione di un CDSS è un processo che può essere difficile, lungo e costoso poiché si tratta di tradurre in un linguaggio eseguibile conoscenze cliniche e sanitarie che possono essere molto complesse. Lo standard HL7 DSS, fornendo l'accesso alle conoscenze nel loro formato eseguibile attraverso un'interfaccia standard, non risolve la difficoltà di progettazione di tali sistemi, ma può permettere una diffusione globale del loro uso. Senza la definizione di uno standard, il fruitore del servizio dovrebbe implementare interfacce specifiche prima di poter utilizzare le conoscenze messe a disposizione da CDSS (e quindi da fornitori) diversi. Con l'adozione dello standard, invece, la stessa interfaccia può essere usata per accedere e quindi per consultare CDSS differenti. Analogamente, la mancanza di uno standard consentirebbe ai fornitori di CDSS di raggiungere solo una piccola parte dei loro potenziali clienti. Inoltre, poiché il servizio è in grado di integrare applicazioni esistenti attraverso la definizione di opportuni wrapper, permette a un fornitore di CDSS di continuare a usare gli approcci che ritiene più opportuni e già implementati per lo sviluppo delle sue applicazioni. Il servizio infatti, non definisce il modo in cui la conoscenza deve essere rappresentata. Un client del servizio standard può essere potenzialmente qualsiasi soggetto che desidera implementare, mantenere o

semplicemente usare sistemi di supporto alle decisioni. Il vantaggio fondamentale di questo standard è che è altamente sicuro poiché non vi è alcuna necessità da parte del servizio di sapere chi viene valutato infatti, nessun dato identificativo del paziente è necessario per le valutazioni. Il servizio inoltre, non ha bisogno di accedere agli archivi di dati del client garantendo così un elevato livello di sicurezza. È il client che controlla tutte le chiamate lettura / scrittura ai propri repository clinici. Il servizio è composto da tre interfacce: Metadata Discovery, Query ed Evaluation. La MetadataDiscovery Interface mette a disposizione un insieme di operazioni che permettono a un client di identificare i meta dati del servizio o dei CDSS che gestisce come ad esempio profili, caratteristiche e formato dei dati scambiati. La Query Interface fornisce operazioni per l'individuazione e l'analisi dei CDSS di interesse. Come ad esempio, identificare quali sono i dati necessari a uno specifico CDSS per la valutazione. La Evaluation Interface infine, mette a disposizione quelle funzioni che permettono di valutare i dati dei pazienti attraverso i CDSS di interesse. Lo standard può essere implementato attraverso due diversi profili di conformità. Ciascun profilo definisce l'elenco delle operazioni supportate.

### 3.1.3 HL7 Virtual Medical Record for Clinical Decision Support

Lo standard HL7 Virtual Medical Record for Clinical Decision Support (vMR-CDS), di seguito indicato solo con vMR, è un modello usato per rappresentare i dati clinici utilizzati e prodotti da CDSS. In particolare, attraverso il formato standard vMR si possono rappresentare dati anagrafici, storia clinica di un paziente, le inferenze ottenute attraverso i CDSS (suggerimenti di diagnosi o suggerimenti di esami clinici e di laboratorio).

Generalmente, organizzazioni diverse utilizzano modelli differenti per rappresentare i dati dei loro pazienti. Di conseguenza, i dati definiti secondo il modello di una certa organizzazione non possono essere direttamente utilizzati da un'altra organizzazione che usa un modello diverso.

L'obiettivo principale dello standard vMR dunque, è far in modo che organizzazioni differenti usino un modello comune per rappresentare i dati dei pazienti così che le informazioni scambiate possano essere usate senza ulteriori conversioni. Il modello può essere usato inoltre, anche per implementare le regole decisionali usate nei motori computazionali dei CDSS.

Rispetto allo standard CDA, i documenti in formato vMR risultano essere più semplici e di immediata lettura da parte di un esperto avendo meno livelli di nidificazione e, omettendo tutti quegli elementi che possono essere necessari ai fini della documentazione ma non necessari ai fini dell'uso del documento da parte di un CDSS.

Il vMR può essere usato per modellare e strutturare una grande varietà di concetti clinici comuni, tra cui:

- I dati clinici di un paziente documentati durante il corso di una visita, in base al risultato di un test o di una procedura diagnostica, oppure sulla base dell'anamnesi
- Le diagnosi, le allergie, le intolleranze o il verificarsi di un evento avverso per il paziente come ad esempio la reazione a un farmaco, cadute, infezioni
- La storia clinica di un paziente
- Le prescrizioni mediche suggerite dai clinici o dai CDSS. Ad esempio la prescrizione di un esame di laboratorio (esami del sangue, analisi delle urine), la prescrizione di un esame strumentale (Radiografia, Risonanza Magnetica), la prescrizione di una dieta alimentare o la prescrizione di farmaci
- I piani interdisciplinari di cura, ovvero una raccolta dei problemi, degli obiettivi e degli interventi per affrontare uno o più problemi di salute e per guidare la risoluzione degli stessi
- La lista di tutte le prescrizioni attive per un certo paziente cioè quelle prescrizioni che non sono ancora state eseguite dal paziente.

In generale, i dati possono provenire da fonti strutturate (ad esempio documenti clinici in formato CDA) e non strutturate. Fonti non strutturate spesso includono informazioni in forma narrativa, quali:

- Annotazioni di cura da parte di medici e infermieri
- Note sulla procedura clinica da seguire (invasiva e non invasiva)
- Note sull'interpretazione di studi di imaging o di altri studi clinici, interpretazioni di esami radiologi o di risultati di test diagnostici

Se le informazioni che riguardano il paziente e che devono essere valutate attraverso un CDSS, provengono da fonti strutturate queste possono essere convertite nel modello vMR in maniera quasi automatica. Se invece le informazioni provengono da fonti non strutturate è necessario effettuare una vera e propria mappatura dei concetti. Per sfruttare appieno le potenzialità del modello vMR, sono in via di definizione diverse risorse in specifiche aggiuntive. In particolare:

- Templates che vincolano il vMR e i suoi componenti al fine di garantire l'interoperabilità
- Approcci di implementazione specifici per la piattaforma, in particolare XML

#### 3.1.4 HL7 FHIR

FHIR® (Fast Healthcare Interoperability Resources) è uno standard per lo scambio di informazioni clinico-sanitarie creato da HL7 come percorso evolutivo di altri standard HL7 tra cui CDA. La prima versione è stata pubblicata in formato draft nel febbraio 2014, attualmente è in fase di ballot (votazioni), e dovrebbe consolidarsi entro il 2017. FHIR combina le migliori caratteristiche di altri standard HL7 e sfrutta i più recenti standard web, focalizzandosi in particolare sulla semplicità di sviluppo, manutenibilità ed estendibilità delle soluzioni. Le soluzioni FHIR possono essere costruite a

partire da un insieme di componenti modulari detti *Risorse*. Tali Risorse, possono essere facilmente assemblate per realizzare sistemi clinico-sanitari anche complessi. FHIR offre supporto per l'utilizzo in un'ampia varietà di contesti: applicazioni mobile, applicazioni cloud-based, condivisione dei dati fra sistemi clinici, comunicazione con server all'interno di sistemi sanitari pubblici e molto altro. La sfida maggiore da affrontare per gli standard sanitari è la gestione della variabilità associata a processi assistenziali diversi. Generalmente questo comporta che, nel corso del tempo, una specifica venga arricchita da più campi, aggiungendo gradualmente costi e complessità delle implementazioni risultanti. L'alternativa si basa su estensioni personalizzate, ma esse tendono a creare molti problemi implementativi. FHIR definisce invece, un semplice framework per estendere e adattare le risorse esistenti. Fornisce un'alternativa ad un approccio document-centric, esponendo l'accesso alle informazioni attraverso servizi: le risorse, come ad esempio il paziente o un referto diagnostico, possono essere trovate, accedute e manipolate direttamente attraverso un URL ( Uniform Resource Locator) di accesso. Le risorse in FHIR sono classificate in sei sezioni:

1. **Clinical:** risorse che rappresentano il contenuto di un record clinico
2. **Identification:** entità di supporto coinvolte nel processo assistenziale come ad esempio pazienti (Patient) e dispositivi (Device)
3. **Workflow:** risorse coinvolte nella gestione del processo assistenziale come appuntamenti per visite o esami diagnostici (Appointment, Order, Encounter)
4. **Financial:** risorse a supporto di fatturazione e pagamenti
5. **Conformance:** risorse per la gestione delle specifiche, dello sviluppo e del testing delle soluzioni FHIR ad esempio, Conformance e ImplementationGuide
6. **Infrastructure:** Funzionalità generali e risorse per i requisiti interni di FHIR

## 3.2 OMG

L'Object Management Group (OMG<sup>®</sup>) [4] è un consorzio internazionale senza fini di lucro, fondato nel 1989 per la definizione di standard tecnologici. Gli standard OMG sono guidati dai fornitori. Principalmente il consorzio sviluppa standard di integrazione per una vasta gamma di tecnologie e un'ampia gamma di settori. OMG, non definisce per le sue specifiche un piano di implementazione e ciò le rende immediatamente utilizzabili. Il consorzio durante l'anno organizza quattro incontri tecnici durante i quali membri e non membri hanno la possibilità di collaborare, conoscere standard tecnologici prodotti dall'associazione e partecipare a speciali eventi informativi sui trend attuali e i temi di interesse. OMG mantiene rapporti con decine di altre organizzazioni, tra cui ISO (che pubblica molti standard OMG senza effettuare ulteriori modifiche) e HL7. Ospita inoltre, organizzazioni quali il Cloud Standards Customer Council<sup>™</sup>(CSCC<sup>™</sup>) e il Consortium for IT Software Quality<sup>™</sup>(CISQ<sup>™</sup>) rispettivamente organizzazioni che mirano a favorire la condivisione delle informazioni e la qualità del software. Sono standard OMG gli standard di modellazione UML e Model Driven Architecture (MDA<sup>®</sup>).

### 3.2.1 UML

UML (Unified Modeling Language) è un linguaggio di modellazione e specifica basato sul paradigma object-oriented. Il nucleo del linguaggio (UML 1) fu definito nel 1996 da Grady Booch, Jim Rumbaugh e Ivar Jacobson. Il linguaggio nacque con l'intento di unificare approcci precedenti raccogliendo le migliori prassi nel settore e definendo così uno standard industriale unificato. La notazione UML è semi-grafica e semi-formale; un modello UML è costituito da una collezione organizzata di diagrammi correlati, costruiti componendo elementi grafici (con significato formalmente definito), elementi testuali formali, ed elementi di testo libero. Ha una semantica molto precisa e un grande potere descrittivo che lo rende un linguaggio di modellazione general purpose.

### 3.2.2 MDA

MDA (Model Driven Architecture) è un approccio d'ingegneria del software che si propone di rappresentare i requisiti e le specifiche di sistemi software come modelli formali a diversi livelli di astrazione (concettuale, logico e implementabile), e di mappare questi modelli con le implementazioni del sistema. L'iniziativa è stata lanciata da OMG nel 2001. L'approccio MDA è sintetizzabile in sei punti, che possono essere associati a sei parole chiave:

**Model** In MDA, il Model (Modello) di un sistema (hardware, software, organizzazione) è una rappresentazione formale delle funzioni, della struttura e del comportamento del sistema. Una rappresentazione è detta formale se è basata su un linguaggio che possiede una sintassi e una semantica formalmente definite (com'è UML), e quindi permette l'analisi, la validazione e l'inferenza meccanica.

**Platform** Una piattaforma (platform) è una collezione di componenti, sistemi e tecnologie che forniscono un insieme di funzionalità attraverso interfacce e modelli d'uso. Di fatto, è la base su cui vengono costruiti i sistemi software. I sistemi software che si basano su una piattaforma possono utilizzarne le funzionalità senza conoscerne i dettagli d'implementazione. L'insieme di tecnologie Web services (XML, XSD, SOAP, WSDL), per esempio, costituisce una piattaforma per un'architettura di servizi. Tale piattaforma ha la caratteristica specifica di garantire l'interoperabilità tra i sistemi software che la usano anche se, le loro piattaforme d'implementazione non sono interoperabili.

**Computation Independent Model (CIM)** È un modello formale dei termini, dei concetti (vocabolario) e delle regole strutturali e operative che organizzano, presiedono e regolano le attività, i processi e le strutture di un'organizzazione e/o di una comunità. In genere è espresso in linguaggio naturale.



**Platform Independent Model (PIM)** È il modello logico funzionale, strutturale e comportamentale di un sistema software, eventualmente distribuito, che non contiene informazioni specifiche sulla piattaforma utilizzata per realizzarlo. Esso funge da specifica formale per gli architetti, i progettisti e gli sviluppatori, ed è interamente espresso in UML.

**Platform Specific Model (PSM)** È il modello strutturale e comportamentale di un sistema software, eventualmente distribuito, totalmente “consapevole” della piattaforma specifica su cui è realizzato, in termini sia architetturali sia tecnologici, ed è espresso nel linguaggio proprio alla piattaforma.

**Mapping** Il mapping in MDA, è la specifica di un metodo ed eventualmente di un algoritmo per trasformare gli elementi di un modello in elementi di un altro modello. Una mappatura è espressa in termini di associazioni, vincoli e regole, e consente, per esempio, la trasformazione da un PIM a un PSM, previa la scelta di una piattaforma.

### 3.3 HSSP

HSSP (Healthcare Services Specification Program) è un progetto internazionale avviato nel 2005 da HL7 e OMG. Il progetto combina le competenze di informatica sanitaria e di interoperabilità di HL7 con le competenze di OMG sui sistemi distribuiti e la loro architettura. All’origine dell’iniziativa vi è la consapevolezza che l’accordo sulla rappresentazione e sul significato dei dati clinico-sanitari non è sufficiente a garantire l’interoperabilità delle attività dei soggetti nei processi clinico-sanitari, e che l’adozione dell’approccio SOA consente un passo decisivo verso l’interoperabilità effettiva. Il programma internazionale ha come obiettivo generale la standardizzazione, sia a livello funzionale che tecnico, dei servizi “generici” che sono coinvolti in praticamente tutti i processi socio-sanitari e che in essi svolgono un ruolo chiave. L’obiettivo generale del progetto HSSP è quindi, quello di ottenere

l'interoperabilità dei processi socio-sanitari a livello locale, trans-regionale e trans-nazionale, attraverso l'immersione di tali processi in un'architettura di servizi. Il livello cui si pongono le specifiche HSSP è particolarmente sfidante, perché HSSP accetta pragmaticamente il carattere totalmente distribuito del sistema sanitario globale, come premessa della definizione e realizzazione di un'architettura di servizi globale per la sanità. In particolare, l'obiettivo di HSSP è quello di garantire l'interoperabilità delle funzioni di base, ricorrenti nei processi socio-sanitari, mediante la standardizzazione delle modalità di erogazione e fruizione di tali funzioni, prescindendo dalle loro implementazioni interne, da parte dei soggetti partecipanti, e dal loro uso nei processi socio-sanitari. Un tale approccio consente di far evolvere i processi, disegnati come combinazioni di scambi di servizi standardizzati, a costi molto ridotti, mettendo in opera la flessibilità richiesta dalla rapida evoluzione scientifica e organizzativa del settore socio-sanitario. Il progetto ha individuato sei servizi generici da standardizzare. Di seguito le specifiche definite per ognuno:

- **Resource Location and Updating Service (RLUS).** Fornisce un insieme di interfacce che permettono l'accesso e la gestione di informazioni socio-sanitarie. I dati socio-sanitari possono essere localizzati, recuperati e modificati indipendentemente dal loro formato, dalle esigenze e dai vincoli di sicurezza, dal meccanismo con cui tali dati sono messi a disposizione dal fornitore del servizio. La specifica fornisce dunque una soluzione per gestire l'interoperabilità del trattamento dei dati contenuti nel fascicolo sanitario elettronico distribuito
- **Identity Cross-Reference Service (IXS).** La specifica fornisce soluzioni per garantire l'interoperabilità delle anagrafiche dei pazienti, degli operatori, delle strutture, dei dispositivi, e in generale delle entità presenti nel sistema clinico-sanitario. Permette di allocare un identificativo unico a ciascuna entità e simmetricamente di ritrovare l'identificativo sulla base di proprietà e caratteristiche della stessa
- **Common Terminology Service 2 (CTS2).** Favorisce l'interopera-

bilità della gestione di terminologie e codifiche eterogenee di tipo clinico e sanitario

- **Healthcare and Community Services Provider Directory (HC-SPD)**. Permette a organizzazioni e professionisti di rendere pubblici i servizi da loro offerti e di conseguenza, favorisce l'interoperabilità della ricerca in una directory contenente informazioni relative ad organizzazioni e professionisti che erogano servizi socio-clinico-sanitari secondo criteri predefiniti
- **Privacy, Access and Security Services (PASS)**. Definisce un insieme di servizi di gestione della privacy, del consenso all'accesso di dati sensibili, di gestione dell'autenticazione
- **Decision Support Service (DSS)**. Definisce un'interfaccia standard di accesso e di gestione di applicazioni di supporto alla decisione, indipendentemente dal formato dei dati, dai meccanismi di inferenza e dalle tecnologie d'implementazione di tali applicazioni. Definisce dunque, le specifiche per l'interoperabilità dell'accesso ai CDSS

### 3.4 IHE

IHE [2] è un'iniziativa internazionale avviata negli Stati Uniti nel 1998 ad opera di RSNA (Radiological Society of North America) e HIMSS (Healthcare Information and Management Systems Society) con lo scopo di supportare lo sviluppo dell'integrazione tra sistemi informativi sanitari e assicurare che tutte le informazioni richieste per le decisioni cliniche siano corrette e facilmente reperibili da parte delle varie figure professionali coinvolte. L'obiettivo dell'iniziativa IHE è quello di definire in maniera chiara come gli standard esistenti (in particolare DICOM e HL7) devono essere utilizzati dai diversi sistemi informativi per realizzare un'integrazione tra loro. Come è noto, in ciascuna struttura sanitaria esistono numerosi sistemi informativi distinti, che gestiscono i dati anagrafici, clinici e diagnostici del paziente. Tali sistemi

che hanno spesso bisogno di condividere informazioni, pur utilizzando protocolli standard di comunicazione, spesso non sono in grado di scambiarsi efficientemente dati, in quanto gli standard stessi possono presentare conflitti interpretativi e un'ampia scelta di opzioni. A tal fine IHE non definisce un vero e proprio standard di comunicazione, ma costruisce un linguaggio univoco che elimina le possibili ambiguità contenute negli standard esistenti. L'iniziativa IHE dagli Stati Uniti si è diffusa in tutto il mondo con la creazione di diversi comitati IHE nazionali e regionali, tra cui il Comitato IHE Italia. L'approccio IHE alla standardizzazione parte dalla definizione di casi d'uso specifici ovvero da specifiche modalità di collaborazione tra soggetti clinico-sanitari e cerca di individuare e di formalizzare tutte le fasi di uno specifico processo clinico-sanitario a livello internazionale e nazionale. Definisce inoltre, le architetture software per il trattamento delle informazioni e l'automazione dei processi (workflow) a supporto di tali casi d'uso. Le linee guida per l'implementazione degli standard sono chiamate *Profili IHE*. L'approccio basato sull'astrazione da pratiche locali specifiche mostra però due punti deboli:

1. Le modifiche dei processi di lavoro provocate dall'adozione degli standard devono essere accettate da soggetti che non le hanno prodotte
2. Il numero dei profili tende a crescere vertiginosamente ed è potenzialmente illimitato visto l'ampliamento progressivo dell'approccio a nuovi domini clinico-sanitari e quindi a nuovi casi d'uso

I profili di integrazione proposti da IHE riguardano comunque per lo più la standardizzazione dei processi clinici.



# Capitolo 4

## Multi Classificatori

Nell'ambito dell'apprendimento supervisionato, il problema della classificazione è sicuramente il più importante. La classificazione è il processo che assegna nuove istanze a classi e categorie che contengono entità dello stesso tipo. Un Classificatore è un algoritmo, in grado di eseguire la classificazione: esso estrae dall'insieme di dati su cui è addestrato un modello che utilizza successivamente per classificare nuovi casi. Si definisce dataset un insieme  $D = \{x_1, x_2, \dots, x_n\}$  di  $n$  istanze. Ogni istanza è una collezione  $x = \{a_1, a_2, \dots, a_m, c\}$  costituita da  $m$  attributi e da un'etichetta di classe  $c$ . L'etichetta di classe è il valore che ci si aspetta come esito del processo di classificazione per l'istanza  $x$ ; appartiene a un insieme predefinito di valori detti appunto classi. Generalmente, in un processo di classificazione supervisionata, l'insieme di dati iniziale viene suddiviso in tre insiemi disgiunti: training set, validation set e test set. Il training set viene utilizzato per costruire e addestrare uno o più classificatori distinti. Il validation set viene utilizzato per valutare le prestazioni che ciascun classificatore ha nel riconoscere nuove istanze. Il test set contiene invece le istanze a cui si vuole assegnare un'etichetta. L'approccio tradizionale alla classificazione prevede che, per uno stesso problema, si addestrino più algoritmi di classificazione e si scelga quello più accurato. Tale approccio presenta però diversi limiti specie in quelle situazioni in cui si hanno pochi dati a disposizione per l'addestra-

mento. In questi casi, spesso non si riesce a individuare con un certo grado di affidabilità il classificatore migliore anzi, potrebbe accadere di scegliere il peggiore tra tutti quelli addestrati. La tendenza degli ultimi anni è quella di usare Sistemi Multi Classificatore (MCS). Un MCS è un sistema che addestra sullo stesso insieme di dati più classificatori detti classificatori di base, i quali cooperano (in base a un criterio prestabilito) al fine di prendere la decisione finale. L'architettura generale di un MCS è mostrata in Figura 4.1.

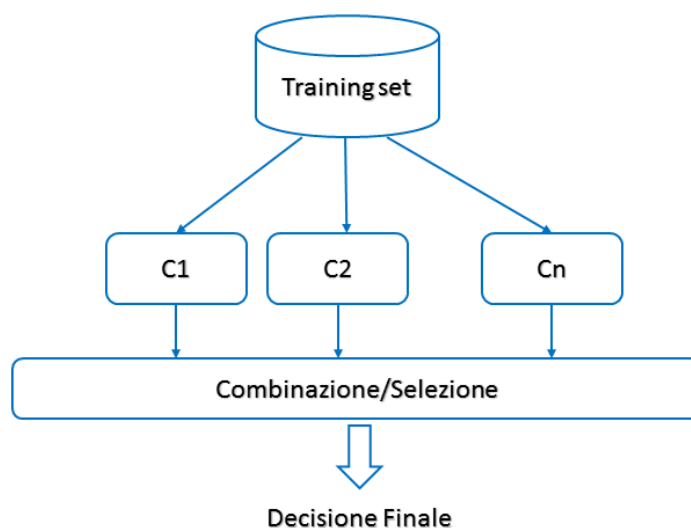


Figura 4.1: Architettura generale di un Multi Classificatore

Tipicamente in un sistema multi classificatore ogni singolo classificatore fornisce in output la propria decisione riguardo l'istanza in esame. Le decisioni dei singoli classificatori vengono fuse e/o combinate a un qualche livello della catena di classificazione prima di prendere la decisione finale. I classificatori addestrati possono avere tutti la stessa importanza oppure, a ognuno può essere associato un peso in base a come essi si comportano ad esempio durante la fase di addestramento. Gli aspetti fondamentali nella progettazione di un MCS sono sostanzialmente tre:

1. Topologia del sistema: come interconnettere i singoli classificatori
2. Progettazione del pool iniziale di classificatori

## 3. Progettazione della strategia di combinazione

## 4.1 Principali Classificatori di base

In ML esistono diversi metodi di apprendimento supervisionato finalizzati alla costruzione di classificatori. Di seguito i concetti alla base delle metodologie principali:

**Rete Neurale Artificiale.** Una rete neurale artificiale (Artificial Neural Network, ANN) è un modello matematico che simula il processo di apprendimento del cervello umano utilizzando una rete di strati interconnessi di neuroni artificiali [65]. Sono state introdotte da Rumelhart et al. [71] nel 1986. Una ANN può essere vista come una funzione matematica non lineare che trasforma un insieme di variabili indipendenti  $x = (a_1, \dots, a_m)$ , ingressi della rete, in un insieme di variabili dipendenti  $c = (c_1, \dots, c_k)$ , uscite della rete. La forma di queste funzioni dipende dalla struttura interna della rete e da un insieme di valori  $w = (w_1, \dots, w_m)$ , detti pesi. La funzione della rete può essere scritta nella forma  $c = f(x; w)$ , dove  $c$  è una funzione di  $x$  parametrizzata da  $w$ . Il modello matematico di un singolo neurone è quello rappresentato in Figura 4.2.

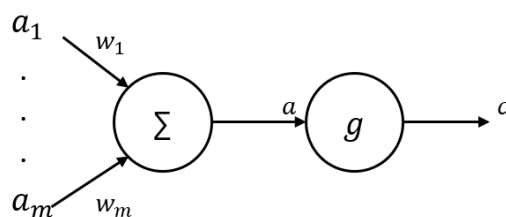


Figura 4.2: Rete neurale con singolo neurone

Un neurone (o Perceptron) può essere visto come una funzione non lineare che trasforma le variabili di ingresso  $a_1, \dots, a_m$  nella variabile



di uscita  $c$ . Nel modello di Figura 4.2 si effettua la somma ponderata degli ingressi usando come pesi i valori  $w_1, \dots, w_m$ , ottenendo così

$$a = \sum_1^m w_i a_i + w_0 \quad (4.1)$$

dove il parametro  $w_0$  viene chiamato bias. L'uscita  $c$  è ottenuta applicando una trasformazione non lineare  $g()$ , chiamata funzione di attivazione:  $c = g(a)$ .

Esistono varie categorie di rete neurale. Quelle più usate sono sicuramente le reti di tipo feedforward. In tali reti l'informazione si muove in un'unica direzione e non ci sono cicli.

- **Multilayer Perceptron.** Le reti neurali multistrato (Multilayer Perceptron - MLP) si compongono di più nodi di elaborazione (perceptron) descritti dall'equazione 4.1. Tipicamente una rete neurale multistrato è costituita da uno strato di ingresso,  $L$  strati nascosti e uno strato di uscita. Gli strati nascosti sono quelli che apprendono i modelli dai dati. Come mostrato in Figura 4.3, una rete neurale richiede solitamente alcuni input e produce uno o più uscite impiegando un algoritmo di apprendimento incrementale che calcola e modifica man mano la forza delle connessioni tra l'ingresso, l'uscita e i livelli nascosti della rete.

Il processo addestramento di una rete neurale può essere molto oneroso dal punto di vista computazionale e consiste nella ricerca dei valori dei parametri  $w$  che minimizzano un'opportuna funzione di errore. Un aspetto fondamentale nella progettazione delle MLP è dato dalla scelta del numero di strati nascosti e dalla scelta del numero di elementi per ogni strato. Entrambi i parametri, infatti, determinano il grado di complessità della rete e di conseguenza anche il tipo di problemi che essa è in grado di risolvere. Le reti neurali rappresentano al giorno d'oggi una delle architetture più diffuse per la soluzione di problemi di classificazione. In ambito

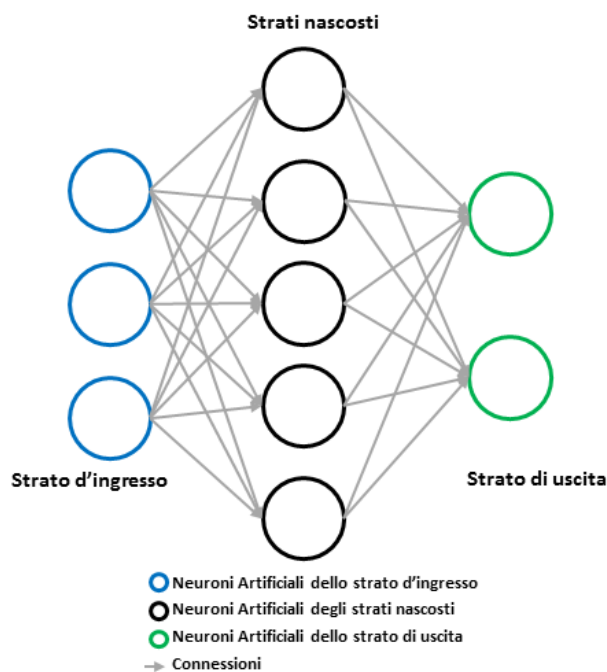


Figura 4.3: Esempio di rete neurale multistrato

clinico Lin et al. [56] utilizzano una rete neurale per prevedere i risultati del trapianto di rene.

**Support Vector Machine (SVM).** Concettualmente le SVM sono simili alle ANN. Esse sono strumenti semplici da analizzare matematicamente ma che consentono di realizzare sistemi complessi. Sono particolarmente adatte a compiti di classificazione molto difficili [62]. Una SVM rappresenta i dati come punti nello spazio, in modo che i dati di categorie distinte (per esempio malattie presenti e assenti) siano divisi da un gap evidente il più ampio possibile. Gli esempi da classificare vengono mappati nello stesso spazio. La SVM prevede la loro appartenenza a una categoria in base al lato del piano in cui essi ricadono.

- **SVM Lineari.** Il caso più semplice da analizzare è quello in cui i dati di addestramento (training set) sono linearmente separabili. Nel caso semplice di due classi, una SVM per risolvere il problema

determina un iperpiano che massimizza il margine di separazione tra le classi. Da un punto di vista formale il problema è un problema di ottimizzazione soggetto a vincoli di uguaglianza. Dato un insieme di dati di addestramento  $D$ :

$$D = \{(x_i, c_i) | x_i \in \mathbb{R}^m, c_i \in \{-1, 1\}\}_n^{i+1} \quad (4.2)$$

Si vuole trovare l'iperpiano che massimizza il margine tra i punti aventi  $c_i = 1$  da quelli aventi  $c_i = -1$ . Indicando con  $w$  il vettore normale all'iperpiano e  $b$  l'intercetta all'origine, tutti i punti che giacciono sull'iperpiano separatore soddisfano l'equazione:

$$wx - b = 0 \quad (4.3)$$

Si suppone cioè che tutti i dati del training set soddisfino le seguenti condizioni:

$$wx_i - b \geq 1 \quad \text{se } c_i = +1 \quad (4.4)$$

$$wx_i - b \leq -1 \quad \text{se } c_i = -1 \quad (4.5)$$

che possono essere rappresentate attraverso la seguente disuguaglianza:

$$c_i(wx_i + b) - 1 \geq 0 \quad \forall i = 1, \dots, n \quad (4.6)$$

In riferimento alla Figura 4.4 i punti che soddisfano l'uguaglianza nell'Equazione 4.4 giacciono tutti sull'iperpiano  $H_1 : wx_i + b = 1$  mentre i punti per cui vale l'uguaglianza nell'Equazione 4.5 giacciono sull'iperpiano  $H_2 : wx_i + b = -1$ .  $H_1$  e  $H_2$  risultano essere paralleli e nello spazio che li separa non cadono punti del training set. La distanza tra i due piani, detta *margine* è uguale a  $2/\|w\|$ . Tutti i punti che si trovano sui due iperpiani  $H_1$  e  $H_2$  sono di fondamentale importanza per le SVM poiché la loro rimozione modifica la soluzione trovata. Tali punti sono detti *vettori di supporto*.

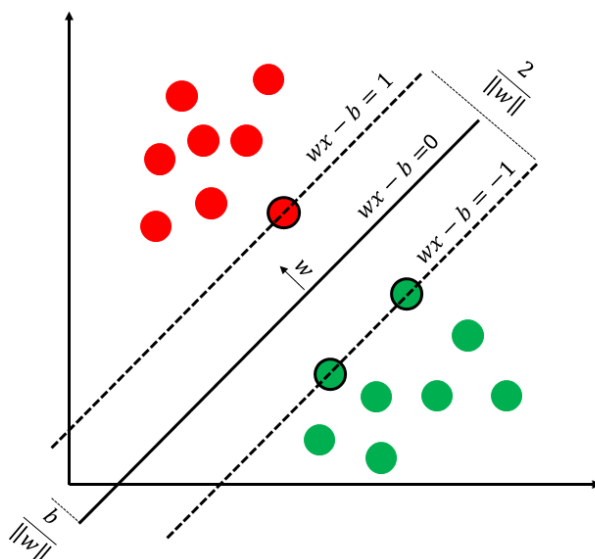


Figura 4.4: Esempio di dati linearmente separabili

L'algoritmo che le SVM utilizzano per trovare i due parametri ( $w$  e  $b$ ) del separatore ottimo consiste nel rendere massimo il margine appena definito. Un modo può essere quello di cercare i due iperpiani che minimizzano  $\|w\|^2$ , con  $w$  e  $b$  soggetti ai vincoli dell'Equazione 4.3. Il problema di minimo vincolato da risolvere diventa un problema di programmazione quadratica e può essere formalizzato nel modo seguente:

$$\begin{cases} \min_{w,b} \frac{1}{2} \|w\|^2 \\ c_i(wx_i + b) \geq +1 \quad \forall i = 1, \dots, n \end{cases}$$

Quando sono presenti dei campioni anomali che si trovano nel semipiano sbagliato e la classificazione binaria tramite iperpiani non può essere eseguita in maniera corretta, per estendere il metodo di risoluzione visto in precedenza si utilizzano delle variabili dette variabili slack  $\xi_i \geq 0$ . Tali variabili, come mostrato in Figura 4.5, si usano essenzialmente per misurare la distanza tra un valore

anomalo e gli altri punti della classe di appartenenza . Si suppone quindi che i dati del training set rispettino i seguenti vincoli:

$$wx_i - b \geq 1 - \xi_i \quad \text{se } c_i = +1 \quad (4.7)$$

$$wx_i - b \leq -1 + \xi_i \quad \text{se } c_i = -1 \quad (4.8)$$

Le due condizioni possono essere rappresentate attraverso un'unica disuguaglianza:

$$c_i(wx_i + b) \geq 1 - \xi \quad \forall i \quad (4.9)$$

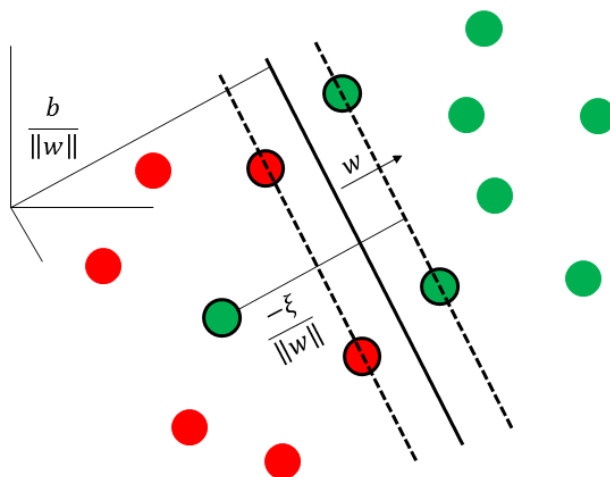


Figura 4.5: Esempio di dati non linearmente separabili

Per ogni campione anomalo, la corrispondente  $\xi_i$  deve essere più grande dell'unità. La funzione da minimizzare diventa:

$$\frac{1}{2} \|w\|^2 + C \left( \sum_i \xi_i \right)^k \quad (4.10)$$

$C$  e  $k$  sono parametri scelti a priori.  $C$  rappresenta il peso che si vuole attribuire agli errori. Ad un alto valore di  $C$  corrisponde un elevato peso degli errori commessi. La Figura 4.5 riporta un

esempio. Il piano viene determinato tramite i vettori di supporto (punti cerchiati) e ha distanza  $-\frac{b}{\|w\|}$  dall'origine. Ogni punto che risulta essere mal classificato, invece, ha distanza  $-\frac{\xi_i}{\|w\|}$  dalla sua classe di appartenenza. Terminata la fase di apprendimento in cui, grazie alle tecniche descritte, si ottengono i parametri dell'iperpiano ottimo di separazione, le SVM procedono alla classificazione vera e propria di nuovi dati. Per raggiungere lo scopo in genere si utilizza la seguente funzione di decisione:

$$f(x) = \text{sign}(wx + b) \quad (4.11)$$

- **SVM Non Lineari.** Nel caso in cui i dati di addestramento non possono essere separati in maniera lineare ma è necessario usare ad esempio una funzione quadratica, cubica o addirittura di grado superiore, il metodo usato dalle SVM consiste nel mappare i dati iniziali in uno spazio di dimensione superiore. La tecnica è nota come tecnica degli spazi immagine (o feature spaces). Per la mappatura si utilizza una funzione  $\Phi : \mathbb{R}^m \rightarrow \mathbb{R}^p$  con  $p \geq m$ . La Figura 4.6 riporta un esempio di due classi non linearmente separabili che diventano separabili linearmente in seguito alla mappatura attraverso la funzione  $\Phi$ . Nel nuovo spazio sarà quindi possibile trovare i parametri di un iperpiano lineare che le separa. Il procedimento utilizzato è lo stesso di quello usato per dati linearmente separabili. Nel problema di programmazione quadratica da risolvere in  $\mathbb{R}^p$  i dati del training set appaiono solo sotto forma di prodotto scalare  $x_i \cdot x_j$ . In questi casi, la tecnica degli spazi immagine è particolarmente interessante. Infatti, nello spazio  $\mathbb{R}^p$  non si devono trovare esplicitamente  $\Phi(x_i)$  e  $\Phi(x_j)$  ma basta calcolare il loro prodotto scalare  $\Phi(x_i) \cdot \Phi(x_j)$ . Per rendere semplice quest'ultimo calcolo, che in spazi di dimensioni elevate diventa molto complicato, si utilizza una funzione detta *kernel* che

restituisce direttamente il prodotto scalare delle immagini:

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \quad (4.12)$$

In generale, i kernel più utilizzati sono:

**Lineare:**  $k(x_i, x_j) = x_i \cdot x_j$

**Polinomiale:**  $k(x_i, x_j) = (x_i \cdot x_j)^d$

**Gaussian Radial Basis function:**  $k(x_i, x_j) = \exp \frac{-(|x_i - x_j|^2)}{2\sigma^2}$

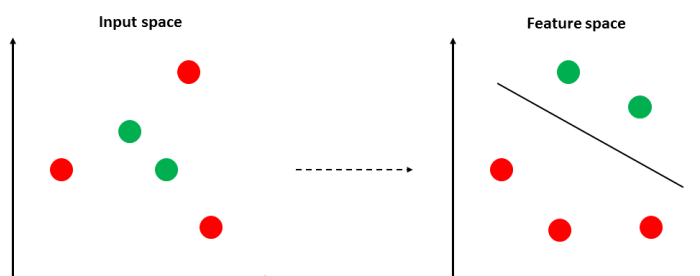


Figura 4.6: Esempio di mappatura

Le SVM sono state introdotte da Vladimir Vapnik e dai suoi collaboratori nel 1992 ed estese nel 1995 [84]. La prima versione era dotata solo di un kernel lineare. Funzioni di kernel non lineari sono state introdotte più tardi. In ambito clinico BH et al. [10] utilizzano una SVM per prevedere l'insorgenza della nefropatia diabetica.

**Naive Bayes.** Il classificatore Naive Bayes [41] è un classificatore probabilistico basato sul teorema di Bayes. Sia dato un insieme di possibili classi  $C$  e una serie di attributi  $A_1, A_2, \dots, A_m$ . Indicando con  $(c, a_1, \dots, a_m)$  dei particolari valori assunti dalle variabili, la classificazione ottimale sarà quella per cui la probabilità di una certa classe, data una serie di valori assunti dagli attributi, è massima:

$$\max P(C = c | A_1 = a_1 \wedge A_2 = a_2 \wedge \dots \wedge A_m = a_m) \quad (4.13)$$

Per il teorema di Bayes si ha che questa probabilità è uguale a:

$$\max \frac{P(A_1 = a_1 \wedge A_2 = a_2 \wedge \dots \wedge A_m = a_m) | C = c}{P(A_1 = a_1 \wedge A_2 = a_2 \wedge \dots \wedge A_m = a_m)} P(C = c) \quad (4.14)$$

La probabilità a priori di avere una data classe  $c$  si può stimare facilmente dal training set, a patto che sia rappresentativo dei dati analizzati, mentre il denominatore dell'equazione è irrilevante, dato che è lo stesso per tutte le classi dell'insieme  $C$  in quanto da esse indipendente. Si tratta quindi di stimare la probabilità condizionata al numeratore per ciascuno degli attributi. La stima è molto complicata a causa delle dipendenze tra attributi e anche con poche feature rischia di diventare computazionalmente irrealizzabile. Per ovviare al problema si fa un'ipotesi piuttosto naive sulle feature, ovvero, che le feature siano completamente indipendenti. In questo modo si ha che:

$$P(A_1 = a_1 \wedge A_2 = a_2 \wedge \dots \wedge A_m = a_m \wedge C = c) \quad (4.15)$$

e quindi che

$$P((A_1 = a_1 \wedge A_2 = a_2 \wedge \dots \wedge A_m = a_m) | C = c) \quad (4.16)$$

Ciò equivale a

$$P(A_1 = a_1 | C = c) P(A_2 = a_2 | C = c) \dots P(A_m = a_m | C = c) \quad (4.17)$$

A questo punto si può ricavare tutto il necessario dal training set:

$$P(A_j = a_j | C = c) = \frac{\text{count}(A_j = a_j \wedge C = c)}{\text{count}(C = c)} \quad (4.18)$$

In questo modo, tutti i dati per calcolare le probabilità a posteriori  $P(C = c | A_1 = a_1)$  sono disponibili. Il metodo sceglie la classe che presenta la probabilità maggiore. L'ipotesi naive riduce notevolmente la complessità computazionale. Il vantaggio principale del classificatore Naive Bayes è infatti, proprio la sua rapidità di utilizzo. Tale metodo è in grado di gestire con facilità dati con molti attributi e ha bisogno



solo di un piccolo training set per stimare con precisione i parametri. Le assunzioni sull'indipendenza delle feature sono il più delle volte irrealistiche, ma ciononostante le prestazioni di questi classificatori sono piuttosto buone se ben addestrati. In [66] il metodo è usato per la diagnosi di malattie cardiache.

**K- Nearest Neighbour (KNN).** Il classificatore KNN [50] è uno dei metodi di classificazione più importanti e semplici presenti in letteratura. L'idea alla base del metodo è molto semplice: i nuovi esempi vengono classificati in base alla classe dei loro vicini. Per il classificatore non è prevista una fase di addestramento e tutto è fatto a runtime. Data una nuova istanza  $x$  da classificare, l'algoritmo cerca i  $k$  esempi del training set che sono più simili a  $x$ . Assegna a  $x$  l'etichetta che ricorre più frequentemente tra le  $k$  etichette più vicine. Il parametro  $k$  è un intero positivo, tipicamente non molto grande e dispari nella classificazione binaria, per evitare di ritrovarsi in situazioni di parità. La scelta di  $k$  dipende dalle caratteristiche e dalla tipologia dei dati: grandi valori di  $k$  riducono l'effetto rumore nella classificazione (ossia la classificazione è meno influenzata da istanze "strane") ma rendono i confini tra le classi meno definiti. Considerando solo le etichette dei  $k$  oggetti vicini, se una classe predomina sensibilmente sull'altra l'errore di classificazione aumenta. In questa circostanza può essere utile associare dei pesi alle istanze vicine in modo che l'algoritmo dia la "giusta" importanza alla distanza tra l'oggetto considerato e i vicini, senza lasciarsi influenzare totalmente dalla quantità. Per la ricerca dei vicini, l'algoritmo KNN usa una misura di similarità. Tipicamente la distanza più usata è la distanza euclidea. La Figura 4.7 mostra un esempio di classificazione binaria KNN con  $K=3$ .

Gli elementi del dataset appartengono alla *classe O* oppure alla *classe X*. In base a come sono disposti gli elementi nello spazio delle feature, assegnare la classe all'elemento  $q_1$  risulta semplice poiché tutti e tre i suoi vicini appartengono alla stessa classe (*classe O*). Di conseguenza,

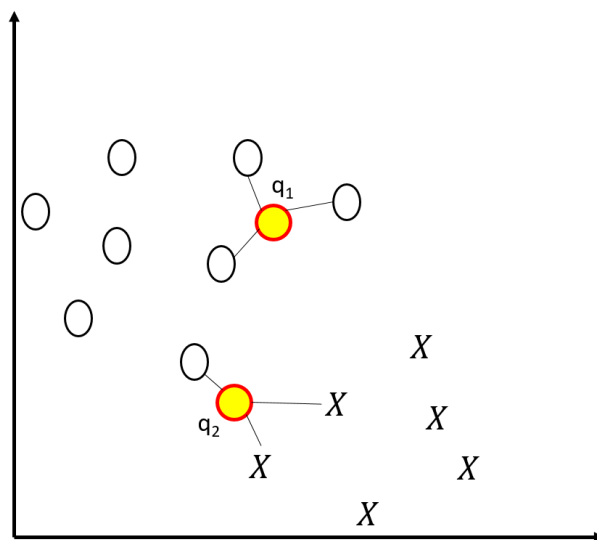


Figura 4.7: Esempio di classificazione 3-NN

anch'esso verrà assegnato alla *classe O*. La situazione di  $q_2$  invece è più complicata poiché due suoi vicini sono di *classe X* mentre uno è di classe *O*. Questa situazione può essere risolta dall'algoritmo in due modi differenti: (1) si calcola la distanza tra  $q_2$  e ciascuno dei suoi 3 vicini, quindi si assegna  $q_2$  alla classe del “vicino più vicino”, oppure (2) usando una tecnica di assegnamento a votazione  $q_2$  verrà assegnato alla classe che ha più elementi tra i suoi vicini. In [55] il metodo è usato per la diagnosi di metastasi linfonodali nel carcinoma gastrico.

**Alberi di decisione** . Gli alberi di decisione Kohavi and Quinlan [49] rappresentano un metodo supervisionato per la costruzione di un modello che mira alla previsione del valore di una variabile di risposta (target) in funzione di un insieme di variabili indipendenti (input). Il principio su cui il metodo si basa è la costruzione di un albero. Dal punto di vista formale, un albero è un insieme finito di elementi detti nodi; il nodo da cui si diramano i successivi nodi viene detto radice. L'insieme dei nodi, a esclusione del nodo radice può essere suddiviso in  $h$  insiemi distinti, detti sotto-alberi. Un nodo viene chiamato padre rispetto ai nodi che

esso genera, mentre viene denominato figlio rispetto al nodo da cui discende. I nodi finali sono detti foglie. Il metodo etichetta i nodi con il nome degli attributi mentre gli archi (i rami dell'albero) con i possibili valori che l'attributo può assumere. Ogni foglia rappresenta il valore di una possibile etichetta per la variabile di uscita. Un nuovo esempio viene classificato seguendo il percorso lungo l'albero che dalla radice porta a un nodo foglia. La strategia adottata dagli alberi decisionali per la costruzione della struttura ad albero a partire da un training set è nota come la strategia divide et impera [3]. Ogni nodo interno in un albero di decisione corrisponde a un test su un particolare attributo. Solitamente si confronta un attributo con una costante. Per classificare un nuovo esempio, esso viene instradato lungo l'albero effettuando lungo il percorso tutta una serie di test. Quando si raggiunge un nodo foglia, l'esempio viene classificato secondo la classe definita dalla foglia. Se l'attributo che viene testato è di tipo nominale, il numero di figli è di solito il numero di possibili valori che l'attributo può assumere.

Esistono due principali categorie di alberi di decisione: alberi di classificazione e alberi di regressione. I primi, sono alberi di decisione usati per predire variabili di tipo categorico mentre i secondi sono usati per predire variabili di tipo quantitativo. In generale, la costruzione di un qualunque albero include tre passi principali:

- Selezione di una regola di splitting per ogni nodo: ciò significa determinare le variabili e il corrispondente valore di soglia, che saranno usate per partizionare il data set a ogni nodo
- Determinare quali nodi sono da intendersi terminali: per ogni nodo bisogna decidere quando continuare con gli splits, quando fermarsi, quando considerare il nodo come terminale. Senza un'adeguata regola, si corre il rischio di costruire alberi troppo grandi con una piccola capacità di generalizzazione, oppure alberi troppo piccoli che invece approssimano male i dati

- Assegnare le etichette ad ogni nodo terminale

Criteri di controllo sono usati per limitare la crescita degli alberi. In tale ambito rientra anche la fase di pruning che consiste nell'ottenere da un albero il più piccolo sottoalbero che non comprometta l'accuratezza della classificazione. Un ramo o un sottoalbero che l'utilizzatore giudica irrilevante perché ha un numero esiguo di casi, può essere rimosso effettuando di fatto un'operazione di "potatura". L'algoritmo C4.5, nato come successore di ID3 [49] è l'algoritmo di albero decisionale più diffuso.

Il vantaggio principale degli alberi di decisione consiste nella visualizzazione dei dati attraverso cui, gli utenti possono comprendere rapidamente la struttura complessiva dei dati e come gli attributi afferiscono alle varie classi. In Figura 4.8, l'albero decisionale usato in Candelieri et al. [14] per la diagnosi precoce di condizioni di decompensazione in pazienti con scompenso cardiaco.

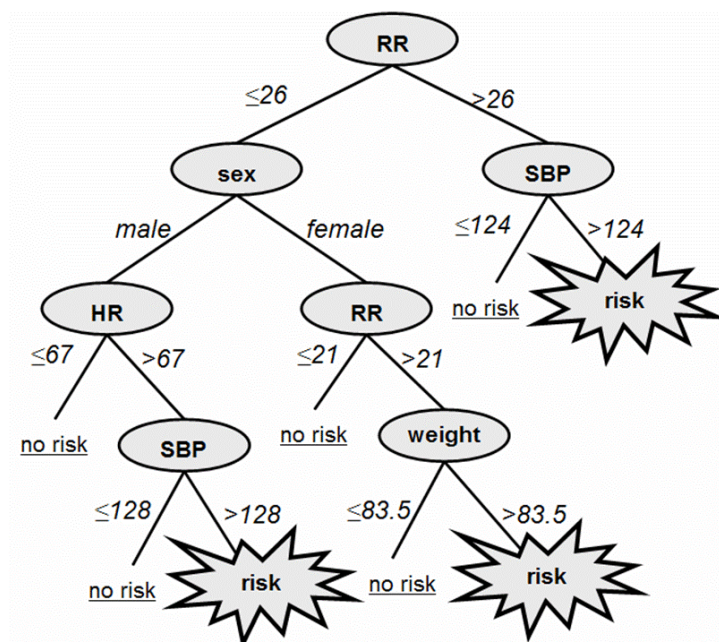


Figura 4.8: Esempio di albero decisionale

## 4.2 Topologia dei Multi Classificatori

Come riportato in Figura 4.9 le topologie principali per MCS sono tre: a cascata, parallela e ibrida. In una configurazione a cascata, i classificatori sono posizionati uno di seguito all'altro. In questo modo, il risultato generato da un certo classificatore viene usato come ingresso del successivo e così via fino all'ultimo della catena. Lo svantaggio principale che si riscontra nell'usare questa topologia è l'incapacità dei classificatori di correggere gli errori commessi dai classificatori precedenti. In una configurazione parallela invece, i classificatori operano in parallelo e in maniera indipendente sui dati. Solo alla fine delle elaborazioni i loro risultati vengono combinati per prendere la decisione finale. Spesso, elementi delle due topologie vengono combinate in una topologia di tipo ibrido.

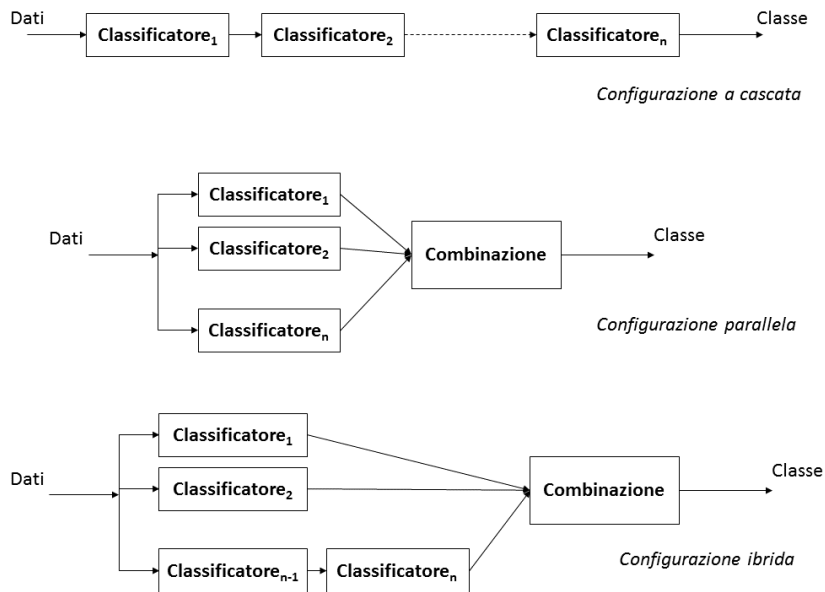


Figura 4.9: Topologie di Multi Classificatore

In letteratura si trovano anche MCS basati su un'altra topologia detta topologia condizionale [53]. In base a essa si addestrano più classificatori in parallelo, si seleziona un primo classificatore e se questo classificatore non riesce a identificare correttamente i dati di test è possibile selezionarne un altro

per eseguire l'operazione di classificazione. La selezione del classificatore successivo può avvenire sia attraverso una decisione di tipo statico che attraverso una decisione presa in base ai valori ottenuti mediante l'uso del classificatore primario. Il processo può continuare finché ci sono classificatori disponibili oppure finché tutti i dati risultano correttamente classificati. Un aspetto difficile di tale implementazione è la valutazione dei fallimenti e dei successi di un classificatore. Inoltre, la metodologia può diventare eccessivamente complicata quando il numero di classificatori disponibili aumenta.

### 4.3 Progettazione del pool iniziale di classificatori

In generale, nella progettazione di un MCS un concetto chiave è quello della diversità. Alla base vi è la necessità di includere classificatori caratterizzati da un'elevata precisione e che siano diversi tra loro. Purtroppo, il problema di come misurare la diversità dei classificatori è ad oggi un problema aperto e molto studiato in letteratura. Misurare la diversità non è semplice poiché non esiste una definizione formale generalmente accettata. Si dice che due classificatori sono diversi se commettono errori su campioni diversi. Uno studio completo sulle misure di diversità tra classificatori è stato affrontato in [52]. Le misure sono definite sia per coppie di classificatori che per insiemi più grandi. Misure di diversità per coppie di classificatori sono:

- Q-statistic
- Correlation coefficient
- Disagreement measure

Misure di diversità per insiemi più grandi invece sono:

- Entropy measure
- Kohavi-Wolpert variance

- Measurement of interrater agreement
- Measure of difficulty
- Generalized diversity
- Coincident failure diversity

In letteratura, per classificatori ensemble, l'errore di classificazione comprende due componenti:

- errore dovuto alla distorsione (bias)
- errore dovuto alla varianza

In tal modo, i metodi di classificazione possono essere diversificati considerando varianza e distorsione [24, 25]: nelle reti neurali all'aumentare dei nodi nascosti aumenta la varianza mentre diminuisce la distorsione; nei metodi KNN, all'aumentare del valore di  $k$  aumenta la distorsione mentre diminuisce la varianza; negli alberi decisionali la profondità definisce la varianza e solitamente si procede alla potatura per poterla controllare.

Per problemi di classificazione, è intuitivo pensare che aumentando la diversità si dovrebbe avere una precisione più alta del sistema combinato anche se, non vi è alcuna prova formale di questa dipendenza come confermato dalla vasta gamma di risultati sperimentali presentati, ad esempio, in [11]. In generale, al fine di progettare classificatori diversi tra loro si possono seguire come linee guida diverse strategie:

- **Diversificazione degli ingressi:** i classificatori vengono addestrati su insiemi di dati differenti. È possibile usare:
  - Partizioni diverse dello stesso insieme di dati
  - Caratteristiche (attributi) diverse per ciascun classificatore
- **Diversificazione delle uscite:** la diversità dei classificatori può essere ottenuta facendo in modo che ciascuno di essi sia progettato per riconoscere solo alcune classi del problema. Il metodo prevede principalmente

di scomporre un problema di classificazione multi-classe in una serie di problemi di classificazione binarie [23] [80] e addestrare dunque, per lo stesso problema più classificatori combinando successivamente le loro decisioni. I metodi più popolari per combinazioni di classificatori binari sono: OAO (uno-contro-uno) e OAA (uno-contro-tutti) [26]

- **Diversificazione dei modelli:** si usano classificatori eterogenei tra loro in modo da poter sfruttare i vantaggi di ciascuno

## 4.4 Progettazione della strategia di combinazione

I metodi di combinazione si dividono in metodi di combinazione basati sulla confidenza e non basati sulla confidenza. La confidenza è espressa come un peso (o una probabilità) associato all'uscita del classificatore. Nei metodi non basati sulla confidenza, tutti i classificatori hanno la stessa importanza per cui, ogni singolo classificatore fornisce in uscita la propria decisione. Tali decisioni vengono combinate secondo una certa tecnica prefissata. Una delle tecniche più usate è la cosiddetta majority vote rule [17] in cui, ogni classificatore vota per una classe e l'istanza da classificare viene assegnata a quella maggiormente votata. Nei metodi di combinazione basati sulla confidenza invece, ogni singolo classificatore fornisce in uscita un vettore  $s$ -dimensionale in cui l' $i$ -esimo elemento indica la probabilità di appartenenza della nuova istanza alla classe  $i$ -esima. I vettori possono essere combinati tra loro usando varie funzioni di classificazione (somma, media, prodotto, max, min). Il metodo più usato è quello della somma. Si esegue cioè la somma vettoriale dei diversi vettori di confidenza classificando l'esempio di test sulla base dell'elemento maggiore. In generale le funzioni di classificazione possono essere di vario tipo [67]:

- Metodi di combinazione lineare: fanno uso di funzioni di combinazione lineari come somma o prodotto



- Metodi di combinazioni non lineari: stilano una classifica delle uscite dei classificatori addestrati. Un esempio è la tecnica del voto a maggioranza
- Metodi statistici: metodi come la tecnica Dempster-Shafer [91] e la combinazione Bayesiana
- Metodi computazionalmente intelligenti: l'uso di paradigmi intelligenti come le reti neurali e gli algoritmi genetici

Metodi come lo Stacking [87] invece, utilizzano come strategia di combinazione un meta-classificatore. Con la tecnica dello stacking si addestrano generalmente in parallelo, più classificatori; le loro uscite sono usate per addestrare un altro classificatore detto appunto meta-classificatore, che è quello che prenderà la decisione finale.

## 4.5 Fasi di sviluppo di Multi Classificatori

In generale, lo sviluppo di un MCS prevede tre fasi:

- **Fase di generazione.** In questa fase viene definito secondo una precisa strategia, l'insieme dei classificatori di base. L'idea è quella di ottenere classificatori diversi e particolarmente accurati sull'insieme di dati considerato. I classificatori di base possono essere sia omogenei (tutti dello stesso tipo) che eterogenei (algoritmi diversi).
- **Fase di selezione.** La fase di selezione consiste nel selezionare i classificatori più competenti sull'insieme di dati considerato. Il processo di selezione può essere di due tipi:
  - **Statico.** Negli approcci statici la selezione viene eseguita durante la fase di addestramento del sistema (fase di training) indipendentemente dai singoli campioni da classificare. Il classificatore o i classificatori selezionati vengono poi utilizzati per la classificazione di tutti i campioni del test set

- **Dinamico.** L'approccio di selezione dinamica consiste nel selezionare un classificatore diverso per ciascun nuovo campione del test set. Tale approccio si basa sul presupposto che ciascun classificatore di base è esperto in una differente regione locale dello spazio delle feature. Solo i classificatori che raggiungono un certo livello di competenza, secondo un certo criterio prestabilito, vengono utilizzati. Lavori recenti presenti in letteratura dimostrano che le tecniche di selezione dinamica sono uno strumento efficace per problemi di classificazione in cui la dimensione dei dati di training è piccola [15] e [16]
- **Fase di integrazione.** La fase di integrazione consiste nell'applicare il classificatore o i classificatori selezionati a ciascun campione del test set ed eventualmente combinare le loro risposte per l'assegnazione dell'etichetta

Le fasi di selezione e integrazione possono essere facoltative. Ci possono essere, infatti, casi in cui tutti i modelli generati vengono usati per prendere la decisione finale e altri in cui ne viene selezionato uno solo. Le fasi di sviluppo di un MCS sono mostrate graficamente in Figura 4.10.

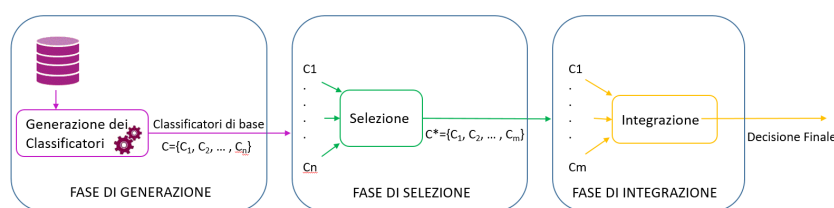


Figura 4.10: Fasi di sviluppo di un Multi Classificatore

## 4.6 Stato dell'arte dei Multi Classificatori

Il concetto di MCS è stato presentato per la prima volta da Chow [19] nel 1965. Il lavoro definisce una serie di condizioni per ottenere l'ottimalità

della decisione congiunta di un insieme di classificatori binari indipendenti con pesi adeguatamente definiti. In [21] viene presentato un MCS che combina un classificatore KNN e un classificatore lineare. Il metodo, individua innanzitutto la regione dello spazio degli attributi in cui i due classificatori sono in disaccordo; il classificatore KNN è usato per classificare gli esempi che ricadono in tale regione mentre quello lineare per tutti gli altri esempi. In [42] gli autori propongono un MCS costituito da un ensemble di reti neurali, mentre in [91] un MCS che utilizza la tecnica del voto a maggioranza per il riconoscimento della grafia. Un ensemble di classificatori è di per se un multi classificatore che combina le predizioni di un insieme di modelli predittivi. Una recente review su MCS [90] mostra come i MCS sono utilizzati in diversi settori tra cui anche quello clinico. Lavori recenti in questo campo sono: [59], [29], [72]. Il MCS presentato in [29] usa una tecnica di combinazione basata su Stacked generalization [88]. La tecnica segue fundamentalmente un'architettura a strati. A livello 0, i classificatori sono addestrati utilizzando l'insieme di dati originale e ogni classificatore fa una previsione per un'istanza test. Gli strati successivi ricevono come input le previsioni del livello precedente, e all'ultimo livello, un unico classificatore, chiamato meta-classificatore, prende la decisione finale. L'andamento complessivo non solo dipende dai singoli classificatori utilizzati a livello 0, ma anche dalla corretta selezione dei classificatori in altri livelli. L'approccio proposto riduce il numero di classificatori usati per prendere la decisione finale mediante un algoritmo evolutivo, che si basa sulla stima di algoritmi di distribuzione (EDA). L'approccio è stato testato su dataset clinici. AptaCDSSE [59] è un CDSS per la diagnosi di malattie cardiovascolari che usa più ensemble di classificatori. Ciascun ensemble è costituito da classificatori omogenei tra loro (tutti dello stesso tipo). Essi sono resi differenti gli uni dagli altri settando per loro parametri differenti. Ogni ensemble è addestrato con le repliche del training set ottenute tramite booststrapping. Le decisioni degli ensemble sono combinate con la tecnica del voto a maggioranza. In [72] invece, gli autori combinano con la tecnica del voting cinque diversi classificatori (in gruppi di due, tre e quattro per

volta) per fare diagnosi di tumore al seno.

Nelle applicazioni reali, risulta difficile soddisfare il vincolo di avere classificatori diversi tra loro e, per evitare in parte il problema, anziché combinare i risultati ottenuti da più classificatori, si usa la tecnica della selezione dinamica (Dynamic Classifiers Selection – DCS). L’idea alla base del metodo è quella di non combinare le decisioni dei singoli classificatori ma far classificare direttamente un’istanza al modello che ha maggiore probabilità di assegnarla alla classe corretta.

## 4.7 Selezione dinamica di classificatori

Quello della selezione dinamica di un insieme di classificatori è un importante tema di ricerca nell’ambito della ML. Essa gioca un ruolo fondamentale nella progettazione e implementazione di MCS. La tecnica consiste nel selezionare per ciascun campione di test i classificatori più competenti all’interno di una regione locale, detta appunto regione di competenza, definita all’interno dello spazio degli attributi. Solitamente la regione di competenza è costituita dalle istanze che si trovano vicino all’esempio da classificare. La maggior parte delle tecniche descritte in letteratura utilizza come criterio di selezione l’accuratezza dei classificatori di base su tale regione. Nella maggior parte dei casi inoltre, la regione di competenza viene definita utilizzando l’algoritmo KNN [37], [89], [38]. Solo in alcuni casi viene definita invece mediante tecniche di clustering [51]. La regione di competenza definita attraverso l’algoritmo KNN risulta costituita dai  $k$  campioni più vicini a quello da classificare. La stima dell’accuratezza locale dipende dal numero di istanze all’interno nella regione di competenza (quindi dalla scelta del parametro  $k$ ), dalla forma della regione considerata (quindi dalla metrica usata per misurare le distanze) e non ultimo, dalla densità di istanze nella regione. L’obiettivo è quello di selezionare il classificatore “migliore”, ovvero quello che si comporta meglio sui vicini dell’esempio da classificare. A seconda di come la regione di competenza viene costruita, un certo classificatore può compor-

tarsi in maniera diversa. Un esempio è mostrato in Figura 4.11. Lo spazio delle feature è rappresentato dalle istanze correttamente classificate indicate con il simbolo  $\square$ , e dalle istanze mal classificate indicate con il simbolo  $\triangle$ , da un certo classificatore. Nel caso raffigurato a sinistra, il classificatore

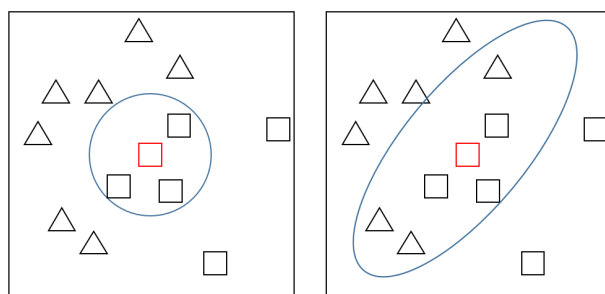


Figura 4.11: Regioni di competenza diverse per lo stesso set di dati

prenderà sicuramente la decisione corretta in quanto tutti i vicini all'interno della regione considerata, sono stati da esso correttamente classificati. Nel caso raffigurato a destra invece, farà la scelta sbagliata poiché la regione di competenza è costituita per lo più da istanze che sono state mal classificate. In generale, il numero di vicini da considerare (parametro  $k$  dell'algoritmo) viene deciso per tentativi secondo la tecnica "trial and error". La metrica più usata per misurare la distanza tra i campioni è la metrica euclidea. La regione di competenza può essere definita sia sul training set che sul validation set.

#### 4.7.1 Stato dell'arte delle tecniche di selezione dinamica

In letteratura sono stati definiti diversi criteri per misurare il livello di competenza dei classificatori di base. In [89] la tecnica conosciuta come DCS-LA OLA (Dynamic Classifier Selection - Local Accuracy - Overall Local Accuracy), valuta l'accuratezza di ciascun classificatore di base come la percentuale di istanze del training set correttamente classificate che ricadono nella regione locale. Il classificatore che ottiene l'accuratezza più alta è considerato il più competente. Sempre in [89] la tecnica DCS-LA-LCA

(Dynamic Classifier Selection -Local Accuracy - Local Class Accuracy) stima invece l'accuratezza rispetto a una sola classe di output. In questo caso, l'accuratezza viene definita come la percentuale di istanze del training set appartenenti a una certa classe, che ricadono all'interno della regione locale e che sono correttamente classificate da un certo classificatore. La classe considerata è la classe assegnata dal classificatore in esame all'istanza test. Anche in questo caso, il classificatore che ottiene l'accuratezza più alta è considerato il più competente per l'istanza test che si sta considerando. In [37] gli autori propongono due approcci per stimare la competenza dei classificatori utilizzando una rappresentazione probabilistica. In entrambi i casi la regione locale è costituita dai  $k$  vicini dell'esempio test considerato. Nel primo metodo denominato metodo "A Priori" il classificatore più competente viene selezionato in base alla sua accuratezza nella regione locale senza considerare la classe assegnata dallo stesso all'istanza test. L'accuratezza viene stimata come la probabilità di un certo classificatore di riuscire a classificare correttamente l'esempio in esame  $\Psi$  appartenente a una classe  $\omega$ . La probabilità viene inoltre ponderata in base alla distanza euclidea  $d_i$  tra l' $i$ -esima istanza  $\Psi_i$  della regione locale e l'esempio in esame  $\Psi$  secondo la seguente espressione:

$$\hat{p}(\text{correct}_j) = \frac{\sum_{i=1}^K \hat{P}_j(\omega_l | \Psi_i \in \omega_l) \cdot \delta_i}{\sum_{i=1}^K \delta_i} \quad (4.19)$$

dove  $\omega_l$  è la classe  $l$ ,  $\hat{P}_j(\omega_l | \Psi_i \in \omega_l)$  è una misura dell'accuratezza del classificatore  $C_j$  per l'istanza  $\Psi_i$ , e  $\delta_i = 1/d_i$ .

Nel metodo denominato "A Posteriori" invece, si prende in considerazione l'informazione relativa all'etichetta assegnata all'esempio test dal classificatore in esame. La formula presa in considerazione è la seguente:

$$\hat{p}(\text{correct}_j | C_j(\Psi_i) = \omega_l) = \frac{\sum_{\Psi_i \in \omega_l} \hat{P}_j(\omega_l | \Psi_i) \cdot \delta_i}{\sum_{i=1}^K \hat{P}_j(\omega_l | \Psi_i) \cdot \delta_i} \quad (4.20)$$

In [37] viene definito un algoritmo basato su entrambi i metodi di selezione. L'algoritmo, fa uso di una soglia: se l'accuratezza di un certo clas-

sificatore è inferiore alla soglia fissata non viene preso in considerazione. Il metodo presentato in [38] ], noto con il nome di DS MCB (Dynamic Selection Multi Classifier Behaviour), definisce prima la regione di competenza, costituita dai  $k$  vicini dell'istanza da classificare nel training set. Successivamente, vengono selezionati i campioni su cui i classificatori di base, nella regione di competenza, hanno un comportamento simile a quello che hanno sull'istanza test. Tali campioni vengono usati per selezionare il classificatore che ha l'accuratezza maggiore rispetto agli altri. L'accuratezza di ciascun classificatore è misurata come la percentuale di istanze che ricadono nella regione locale, correttamente classificate. Se il classificatore selezionato è sufficientemente migliore degli altri in base a un valore di soglia definito, viene usato per classificare il nuovo campione, altrimenti si usa la tecnica del voto a maggioranza per combinare le uscite di tutti i classificatori e prendere la decisione finale. Un'interessante review di metodi di selezione dinamica di classificatori è presentata in [46].

In letteratura, oltre a metodiche che permettono di selezionare un solo classificatore che assegna poi l'etichetta all'esempio test, sono definiti approcci per selezionare più sottoinsiemi di ensemble di classificatori. Un metodo interessante è stato proposto in [16]. Il metodo è basato sul concetto di organizzazioni multistadio (MO) e, l'obiettivo principale è quello di definire una funzione di fusione multistrato che si adatti dinamicamente ad ogni campione di prova. L'idea alla base del metodo, già presentata in [73] è quello di (1) generare un primo insieme di classificatori  $C'$ ; (2) attraverso un processo di ottimizzazione creare a partire da questo, un insieme  $C''$  costituito da ensemble di classificatori; (3) in base alle caratteristiche dell'istanza test selezionare da  $C''$  l'ensemble che produrrà la decisione finale. In [16] sono descritti altri due diversi algoritmi: DSA-M e DSA-C mentre, Nabiha et al. propongono in [63] un'estensione del metodo DCS-LA proposto in [89] per sistemi basati su ensemble di classificatori. Il metodo, calcola l'affidabilità di ogni classificatore sulla base della matrice di confusione ottenuta sul validation set da ciascuno. Un altro approccio interessante è stato proposto

in [48] in due versioni: KNORA-ELIMINATE e KNORA-UNION. L'algoritmo KNORA-ELIMINATE data un'istanza test cerca innanzitutto all'interno del validation set i suoi  $k$  vicini. Solo i classificatori che hanno classificato correttamente tutti i  $k$  vicini entrano a far parte dell'ensemble che assegnerà l'etichetta alla nuova istanza. Nel caso in cui non esiste un classificatore in grado di classificare correttamente tutti i  $k$  vicini, si decrementa  $k$  fino a quando si trova un classificatore che riesce a classificarli correttamente tutti. Se nemmeno con  $k=1$  si riesce a trovare un classificatore con queste caratteristiche, dall'insieme dei  $k$  vicini iniziale si seleziona il classificatore che ha l'accuratezza maggiore e con esso tutti gli altri classificatori che riescono a classificare correttamente la stessa quantità di istanze. In KNORA-UNION invece, si cercano innanzitutto i  $k$  vicini dell'istanza da classificare. Tutti i classificatori che hanno classificato correttamente almeno una delle  $k$  istanze entrano a far parte dell'ensemble. Secondo questo algoritmo, un classificatore può entrare a far parte dell'ensemble più volte se classifica bene più di una istanza e di conseguenza, può esprimere la sua decisione più volte per una stessa istanza test. Nell'articolo, gli autori danno anche una versione pesata dei due algoritmi: KNORA-ELIMINATE-W e KNORA-UNION-W. In entrambi i casi, ciascun voto nell'ensemble viene pesato attraverso la distanza euclidea tra il vicino e l'istanza test.

Il metodo in [85] usa per selezionare i classificatori che entreranno a far parte dell'ensemble sia l'accuratezza locale complessiva (OLA) che l'accuratezza locale sulla classe (LCA) calcolata sull'insieme dei vicini dell'istanza test all'interno del validation set. Il metodo si compone di due fasi. Durante la prima fase, vengono cercati i  $k$  vicini dell'istanza test all'interno del validation set e quindi, viene calcolata per ciascun classificatore che ha classificato correttamente le istanze vicine, la OLA su questo insieme. Si evita in questo modo di calcolare la OLA per tutti i classificatori di base risparmiando tempo e risorse. Su tali classificatori viene fatta una prima selezione. L'articolo in esame propone due diverse strategie: DESCOLA-ELIMINATE e DESCOLA-UNION. In DESCOLA-ELIMINATE vengono selezionati solo



i classificatori che hanno la OLA più alta rispetto a tutti gli altri mentre in DESCOLA–UNION vengono selezionati tutti i classificatori che hanno un OLA maggiore di zero, ovvero vengono selezionati i classificatori che classificano correttamente almeno un vicino. Durante la seconda fase, tutti i classificatori assegnano un’etichetta all’istanza test e quindi viene calcolata l’ambiguità dei classificatori (secondo una determinata equazione) rispetto agli ensemble con l’obiettivo di organizzare i classificatori preselezionati in ensemble in cui l’ambiguità è zero. Ciò significa avere un pieno consenso tra i classificatori all’interno di ogni ensemble dal momento che tutti i classificatori prevedono la stessa classe per l’istanza test. Infine, dopo aver organizzato i classificatori in ensemble, viene selezionato quello più adatto per la particolare istanza test considerata. A tale scopo l’algoritmo calcola la LCA di ogni ensemble rispetto alla classe predetta. Ovviamente viene selezionato l’ensemble che presenta la LCA più alta. Metodi più recenti sono quelli presentati in [43], [20].

# Capitolo 5

## Approccio proposto

La tesi propone:

1. Un'architettura software per CDSS basata su un insieme di moduli di conoscenze cliniche detti Knowledge Modules (KM)
2. Approcci metodologici basati su tecniche di ML per il disegno e lo sviluppo dei KM come Sistemi Multi Classificatori che utilizzano la tecnica della selezione dinamica

La prima parte del capitolo descrive l'architettura software mentre la seconda descrive gli approcci metodologici definiti per la progettazione di Multi Classificatori basati su una nuova tecnica di selezione dinamica.

### 5.1 Architettura proposta

L'organizzazione architetturale per CDSS proposta in questa tesi si basa su tre componenti distinte ma cooperanti:

- un'interfaccia
- un controllore strategico e
- un insieme di moduli di conoscenze cliniche (KM)

L'interfaccia gestisce la comunicazione tra il CDSS e l'utente finale. Per utente finale non si intende solo il medico o l'operatore sanitario. Un utente è una qualsiasi entità esterna che interagisce con il CDSS per ottenere i suoi servizi ad esempio un altro CDSS, una cartella clinica elettronica o qualsiasi altro componente e servizio del sistema sanitario. La funzione principale dell'interfaccia è quella di mettere a disposizione degli utilizzatori le proprie funzionalità, ricevere i dati da valutare e restituire i risultati al chiamante.

Il controllore strategico ha il ruolo di selezionare, istanziare e orchestrare (in base a un criterio prefissato) i KM necessari per definire il più efficace supporto decisionale. Il modulo ha inoltre la responsabilità di acquisire e gestire tutte le richieste che tramite l'interfaccia arrivano al CDSS e di inoltrarli quindi ai KM. Il controllore strategico inoltra poi la decisione finale verso l'interfaccia in modo da poter essere inoltrata all'utente.

Ciascun KM è un motore computazionale in grado di valutare i dati dei pazienti e fornire informazioni circa il loro stato di salute. Ciascun KM è indipendente dagli altri e può essere sviluppato e implementato sia secondo approcci deduttivi che induttivi. Nell'implementazione prodotta, l'architettura di ciascun KM è quella di un MCS in configurazione parallela. Tutti i classificatori vengono addestrati e validati in parallelo e in maniera indipendente sui dati. Alla fine del processo di addestramento e validazione, per ciascuna nuova istanza da classificare viene selezionato un solo classificatore che prenderà la decisione finale. In Figura 5.1 l'organizzazione architetturale semplificata di un KM.

Per il Controllore Strategico possono essere definite diverse politiche di gestione. Esso ad esempio, a seconda dei casi, potrebbe decidere di consultare uno solo dei KM tra tutti quelli in grado di risolvere uno specifico problema decisionale, oppure si potrebbe decidere di consultare più KM in parallelo per poi scegliere il risultato più affidabile tra tutti quelli elaborati. L'architettura del CDSS così per come è stata concepita facilita dunque l'integrazione e la cooperazione tra KM attraverso il Controllore Strategico. Si potrebbe infatti pensare di realizzare KM che siano in qualche modo in relazione tra loro:

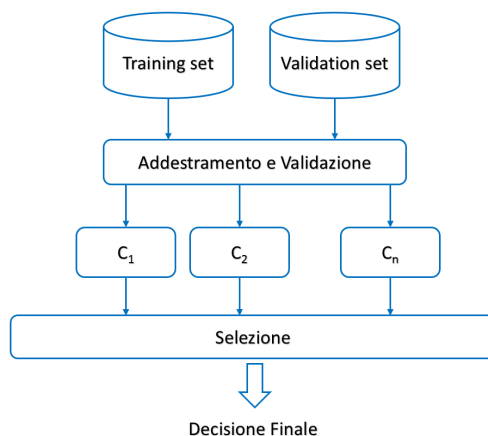


Figura 5.1: Architettura di un KM

un KM potrebbe utilizzare come ingresso i risultati di un altro KM prima di prendere la decisione finale. Tali relazioni potrebbero essere gestite dal Controllore Strategico.

In base alla caratterizzazione generale definita dall'approccio SOA, il CDSS interagisce con i suoi utilizzatori attraverso uno strato di interoperabilità basato su standard internazionali. In particolare, interagisce attraverso un servizio web che implementa lo standard HL7 (Health Level Seven) DSS (Decision Support Service). Tale standard garantisce al CDSS l'interoperabilità operativa con gli altri servizi e applicazioni del sistema sanitario. Esso, infatti, definisce un'interfaccia standard di accesso e di gestione di applicazioni di supporto alla decisione, indipendentemente dal formato dei dati, dai meccanismi di inferenza e dalle tecnologie d'implementazione di tali applicazioni. L'interazione tra lo strato di interoperabilità e gli altri servizi del sistema sanitario è definita attraverso coreografia. Lo standard è descritto in dettaglio nel Capitolo 4. La figura 5.2 mostra una rappresentazione grafica dell'architettura così definita.

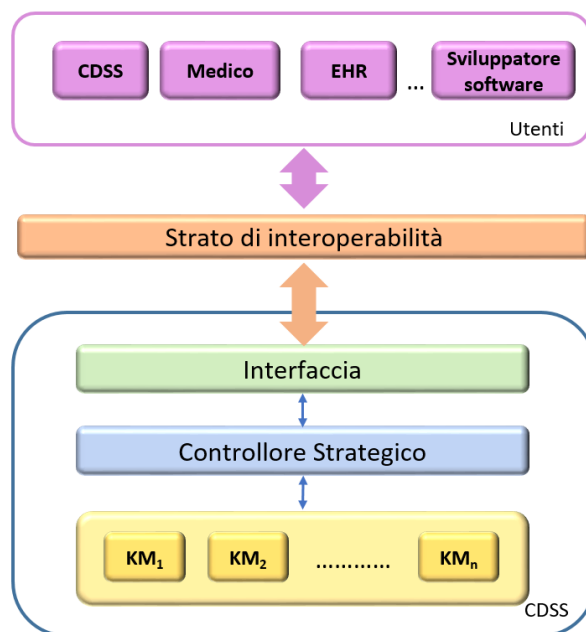


Figura 5.2: Architettura proposta per CDSS

## 5.2 Approccio proposto per la progettazione di Multi Classificatori

In base all'architettura definita nella Sezione 5.1, il CDSS si compone di vari KM. Ciascun KM è disegnato e sviluppato come un MCS che utilizza la tecnica della selezione dinamica per determinare quale, tra i classificatori di base addestrati, è quello più adatto a classificare una certa istanza test. Sono stati definiti tre diversi metodi di selezione dinamica.

L'approccio di classificazione proposto segue lo schema generale di un processo di classificazione supervisionata. L'insieme di dati iniziale viene suddiviso in tre insiemi disgiunti: training set, validation set e test set. Il training set viene utilizzato per costruire e addestrare i classificatori di base. Il validation set viene utilizzato per valutare le prestazioni che ciascun classificatore di base ha nel riconoscere nuove istanze. Il test set contiene invece le istanze a cui si vuole assegnare un'etichetta. Lo schema generale del processo di classificazione è mostrato in Figura 5.3.

## 5.2 Approccio proposto per la progettazione di Multi Classificatori 79

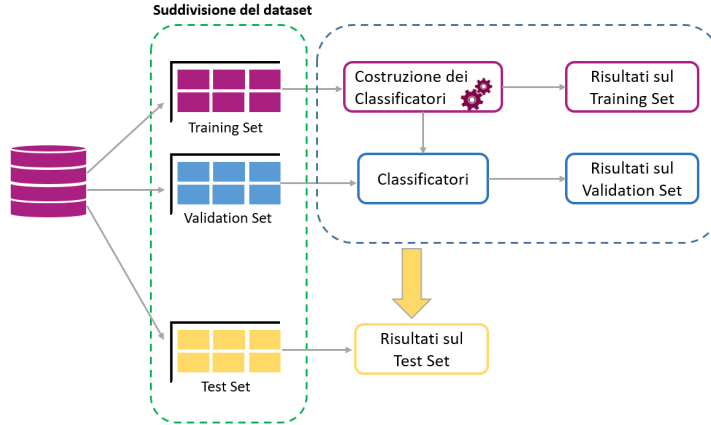


Figura 5.3: Schema generale del processo di classificazione

In generale, il metodo di selezione dinamica proposto in questa tesi tiene conto sia delle istanze mal classificate da ciascun classificatore di base all'interno della regione locale che delle prestazioni generali di ognuno. L'algoritmo *Classification Algorithm* riassume i passi principali dell'approccio proposto.

Sia  $D = \{I_1, I_2, \dots, I_n\}$  un dataset costituito da  $n$  istanze divise in  $h$  classi e sia  $I$  un'istanza test. Sia  $TS$  il training set,  $VS$  il validation set e  $TE$  il test set. Sia  $C = \{C_1, C_2, \dots, C_m\}$  un insieme di  $m$  classificatori di base. L'approccio proposto restituisce l'etichetta di classe predetta per  $I$ . Tutti i classificatori di base vengono addestrati con i dati contenuti nel training set. Al termine della fase di addestramento si valutano le prestazioni di ciascun classificatore usando le istanze del validation set. Le prestazioni di ciascun classificatore valutate sia sul training che sul validation set vengono quindi utilizzate nella fase di selezione del miglior classificatore. Le istanze contenute nel test set vengono considerate una per volta. Per ciascuna istanza test  $I$  viene individuata una regione locale nel validation set; il classificatore più competente è poi selezionato in tale regione. Il classificatore più competente è quello che presenta prestazioni migliori. Eventuali conflitti, ovvero la presenza di più classificatori con le stesse prestazioni, sono stati risolti secondo tre approcci diversi che hanno portato alla definizione di tre differenti ver-

sioni dell’algoritmo di selezione. La prima, utilizza le performance ottenute dai classificatori di base sia sul training che sul validation set. La seconda, utilizza solo le performance ottenute sul validation set mentre la terza è un ibrido tra le precedenti due. Ciascuna versione è descritta in dettaglio nel seguito.

---

### Classification Algorithm

---

```

1: function CLASSIFICATION_ALGORITHM( $C, TS, VS, TE, Option$ ) ▷ where  $C$  is a pool of base classifiers;  $TS$  the training
   set;  $VS$  the validation set;  $TE$  the test set;  $Option = (TRAINING+VALIDATION, VALIDATION, HYBRID)$ 
   Output:  $c_t^*$ , the most promising classifier for each testing sample  $I$  in  $Te$ 
2:   for each classifier  $c$  in  $C$  do
3:     Train  $c$  on  $TS$ 
4:   end for
5:   for each classifier  $c$  in  $C$  do
6:     Validate  $c$  on  $VS$ 
7:   end for
8:   for each test instance  $I$  in  $TE$  do
9:     CALL Compute the Local Region
10:    if  $Option == TRAINING+VALIDATION$  then
11:      CALL Select the best classifiers based on validation and training results and obtain  $c_t^*$ 
12:    else if  $Option == VALIDATION$  then
13:      CALL Select the Best Classifiers based on Validation Results and obtain  $c_t^*$ 
14:    else if  $Option == HYBRID$  then
15:      CALL Select the Best Classifiers based on hybrid Method and obtain  $c_t^*$ 
16:    end if
17:    Use  $c_t^*$  to classify  $I$ 
18:  end for
19: end function

```

---

La regione locale, in cui viene valutata la competenza di ciascun classificatore di base, è definita in modo dinamico per ciascuna nuova istanza test attraverso un algoritmo K-Nearest Neighbor (KNN) di tipo adattivo. Mentre nella versione tradizionale l’algoritmo KNN per costruire la regione locale usa sempre lo stesso numero prefissato di vicini (indicato con  $k$ ) per tutte le istanze test, nella versione adattativa l’esecuzione dell’algoritmo può essere interrotta nel momento in cui vengono soddisfatti determinati criteri. Algoritmi KNN adattivi differenti sono stati presentati in Wang et al. [86], Cruz et al. [20], Sun and Huang [79]. La scelta del valore ottimale di  $k$  per l’algoritmo KNN nella versione tradizionale non è semplice. La scelta di tale parametro dipende, infatti, da diversi fattori: dai dati che si devono trattare, dalla esperienza dell’utente, dal numero di esperimenti condotti. Solitamente si utilizza un approccio di tipo “trial and error”. In questa tesi, per ciascuna istanza test, il numero di vicini viene scelto in modo dinamico.

## 5.2 Approccio proposto per la progettazione di Multi Classificatori 81

L'algoritmo adattativo definito in questa tesi, infatti, mantiene nella regione locale solo le istanze che hanno una distanza dall'istanza test inferiore a una certa soglia. In base alla definizione della soglia si possono avere regioni più o meno grandi e considerare istanze più o meno vicine all'istanza test. Tale soglia viene calcolata anch'essa in maniera dinamica considerando la massima e la minima distanza tra l'istanza test e tutte le altre istanze nel validation set. La soglia è definita dalla formula seguente:

$$R = \begin{cases} (R_{max} - R_{min}) / 2 & \text{se } R_{max} > 3R_{min} \\ R_{min} & \text{altrimenti} \end{cases} \quad (5.1)$$

dove  $R_{max}$  e  $R_{min}$  sono rispettivamente la distanza massima e la distanza minima tra l'istanza test e tutte le altre istanze dell'insieme. In questo modo, la regione locale viene costruita considerando le istanze del validation set che sono all'interno di una ipersfera di raggio  $R$ . In base all'Equazione 5.1, il raggio dell'ipersfera più piccolo è  $R_{min}$ ; il raggio più grande è  $\frac{R_{max}}{2}$ . Quest'ultimo caso si verifica quando  $R_{min}$  è prossimo a zero. Ciascuna istanza è rappresentata come un vettore in cui ogni elemento si riferisce a un attributo specifico. La somiglianza tra due istanze viene valutata considerando la distanza euclidea tra i due vettori che le rappresentano. Valori grandi della distanza indicano istanze lontane tra loro e quindi dissimili, valori piccoli invece indicano istanze vicine e quindi simili. In Figura 5.4 è riportato un esempio di regioni locali diverse costruite sullo stesso set di dati. Le istanze con lo stesso colore appartengono alla stessa classe. L'istanza test si distingue dalle altre perché è quella evidenziata con un riquadro.

A sinistra, la regione è definita in termini di 5-NN. In questo caso, è facile osservare che solo due istanze sono posizionate vicino all'istanza test. Le altre istanze, si trovano all'interno della regione ma sono lontane da essa e nel caso specifico, appartengono anche a classi diverse. A destra invece, la regione è definita dalle istanze che si trovano nella circonferenza di raggio  $R$ . Si può osservare che in questo caso si considerano meno istanze rispetto al caso precedente e che queste sono più vicine all'istanza test considerata. La probabilità che esse appartengano alla stessa classe dunque è più alta. Come



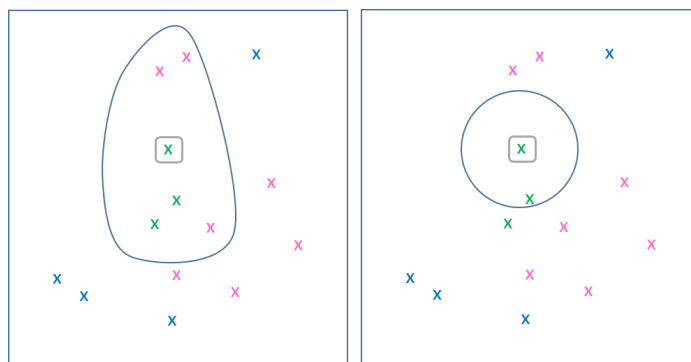


Figura 5.4: Confronto tra una regione locale costruita mediante l'algoritmo 5NN e una costruita mediante l'uso della soglia  $R$

è possibile osservare dalla Figura 5.4, l'uso di una soglia  $R$  definita dall'Equazione 5.1 può essere utile soprattutto nei casi in cui le istanze test si trovano in regioni a bassa densità. L'algoritmo *Compute the Local Region* mostra i passi principali per la definizione della regione locale mentre l'algoritmo *Compute the Threshold* i passi principali per il calcolo della soglia.

---

#### Compute the Local Region

---

```

1: function COMPUTE THE LOCAL REGION( $I, VS$ )                                ▷ where  $I$  is a test instance;  $VS$  is the validation set
   Output: The Local Region  $S$ 
2:    $S = \emptyset$ 
3:   CALL Compute the threshold and obtain  $R$ 
4:   for each  $I_i$  in  $VS$  do
5:     if Euclidean Distance between  $I$  and  $I_i \leq R$  then
6:        $S = S \cup I_i$ 
7:     end if
8:   end for
9:   return  $S$ 
10: end function

```

---

La selezione del classificatore migliore per  $I$  avviene in due fasi. In Figura 5.5 una schematizzazione del processo di selezione. Nella prima fase, si escludono i classificatori che hanno erroneamente classificato una o più istanze all'interno della regione locale. L'idea di base è che, se un classificatore effettua una previsione non corretta per uno o più vicini di una certa istanza, allora la probabilità che esso classifichi erroneamente anche l'istanza test è alta. Pertanto, in fase di classificazione questo classificatore non verrà utilizzato. Nella seconda fase, viene selezionato il classificatore più competente tra

## 5.2 Approccio proposto per la progettazione di Multi Classificatori 83

---

### Compute the Threshold

---

```
1: function COMPUTE THE THRESHOLD( $I, VS$ )                                 $\triangleright$  where  $I$  is a test instance;  $VS$  is the validation set
   Output: The threshold  $R$ 
2:   Let  $E$  an empty array of  $n$  real value
3:   for each instance  $I_i$  in  $VS$  do
4:      $E[i]$  is the Euclidean Distance between  $I$  and  $I_i$ 
5:   end for
6:    $R_{max} = \text{argmax}(E)$ 
7:    $R_{min} = \text{argmin}(E)$ 
8:    $R = (R_{max} - R_{min})/2$ 
9:   if  $R < R_{min}$  then
10:     $R = R_{min}$ 
11:   end if
12:   return  $R$ 
13: end function
```

---

tutti quelli rimasti. Il classificatore più competente è quello che presenta prestazioni migliori. In questa fase, si considerano separatamente le prestazioni di ciascun classificatore. Sono stati definiti due diversi criteri di competenza: uno è basato sulla recall mentre l'altro è basato sull'accuratezza. La recall, fornisce informazioni circa il numero di istanze che appartengono a una certa classe e che vengono correttamente classificate da un classificatore. L'accuratezza invece, fornisce informazioni circa il numero di istanze che vengono correttamente etichettate da un classificatore, indipendentemente dalla classe di appartenenza.

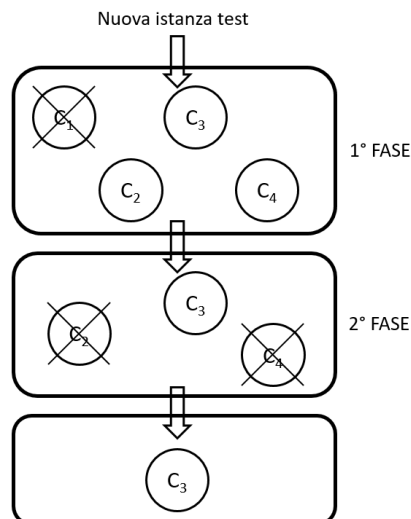


Figura 5.5: Fasi del processo di selezione dinamica

**Criterio basato sulla recall** Si individua la classe maggiormente presente tra le istanze contenute nella regione locale; quindi si valuta la recall dei classificatori sulla classe individuata. Si seleziona il classificatore con la recall più alta.

**Criterio basato sull'accuratezza** Si seleziona il classificatore con la massima accuratezza sul set di dati considerato

Al termine della prima fase di selezione si possono verificare tre casi distinti:

1. **Rimane un solo classificatore di base.** Questo classificatore sarà utilizzato per assegnare l'etichetta all'istanza test.
2. **Rimangono più classificatori di base.** Si seleziona il classificatore più competente per l'istanza  $I$  tra i classificatori rimasti applicando uno dei due criteri di competenza definiti in precedenza (basato su recall o basato su accuratezza).
3. **Non rimane nessun classificatore di base.** Il classificatore più competente per l'istanza  $I$  viene selezionato tra tutti i classificatori di base addestrati. Come nel caso precedente, per selezionare il classificatore più competente si può utilizzare il criterio di competenza basato sulla recall oppure quello basato sull'accuratezza.

Nei casi 2 e 3 possono verificarsi situazioni di indecisione (o conflitti). I conflitti (ovvero più classificatori hanno le stesse prestazioni) sono stati risolti secondo tre approcci diversi che hanno portato alla definizione di tre differenti versioni dell'algoritmo di selezione: Algoritmo 1, Algoritmo 2 e Algoritmo 3.

L'approccio proposto non prevede che i classificatori di base assegnino un'etichetta all'istanza test prima della fase di selezione al contrario di quanto accade, ad esempio, negli approcci presentati da Woods et al. [89]: in tali approcci, infatti, viene prima eseguita una fase di classificazione da parte di tutti i classificatori di base dell'istanza test; l'etichetta da assegnare all'istanza è quella predetta dai classificatori solo se essi sono concordi, altrimenti si

## 5.2 Approccio proposto per la progettazione di Multi Classificatori 85

procede alla fase di selezione del migliore classificatore attraverso la costruzione della regione locale. L'onere computazionale dell'approccio proposto è maggiore rispetto ai metodi di Woods et al. [89] nei casi in cui i classificatori sono concordi nel classificare un'istanza test mentre è inferiore nei casi in cui i classificatori sono discordi. È bene comunque evidenziare che l'onere computazionale è trascurabile nel caso di dataset di piccole/medie dimensioni, come quelli utilizzati nel capitolo successivo per testare l'approccio.

### 5.2.1 Algoritmo 1

L'Algoritmo 1, per ogni istanza test, seleziona il classificatore più competente utilizzando le performance che i classificatori di base hanno ottenuto sia sul training che sul validation set. L'algoritmo durante la prima fase di selezione, esclude i classificatori che etichettano erroneamente una o più istanze della regione locale. A questo punto si possono presentare i tre casi descritti precedentemente. Nei casi 2 e 3 i conflitti vengono risolti nei modi seguenti, a seconda che si utilizzi la recall o l'accuratezza come criterio di competenza.

**Caso in cui si utilizzi il criterio basato sulla recall :** Il criterio viene applicato prima sul validation set; si considera la classe maggiormente presente tra le istanze della regione locale. Quindi si valuta la recall dei classificatori su questa classe individuata considerando le istanze del validation set . Viene selezionato il classificatore con la recall più alta. La presenza di eventuali conflitti, ovvero più classificatori hanno la stessa recall (la massima ottenuta sul validation set) sulla classe considerata, si risolve considerando la recall di questi classificatori sul training set. Nello specifico, si riapplica lo stesso criterio e si seleziona anche in questo caso il classificatore con la recall più alta. I conflitti sul training set vengono risolti invece selezionando il classificatore che presenta l'errore medio assoluto più basso in fase di addestramento.

**Caso in cui si utilizzi il criterio basato sull'accuratezza :** Si seleziona il classificatore che presenta l'accuratezza più alta sul validation set. I conflitti, ovvero più classificatori hanno la stessa accuratezza (la massima ottenuta sul validation set), vengono risolti sulla base dell'accuratezza che questi classificatori hanno ottenuto sul training set. Si seleziona anche in questo caso, il classificatore con l'accuratezza più alta. I conflitti sul training set vengono risolti invece, selezionando il classificatore con l'errore medio assoluto più basso ottenuto in fase di addestramento

### 5.2.2 Algoritmo 2

L'Algoritmo 2 seleziona il classificatore più competente per ciascuna istanza test utilizzando solo le performance ottenute dai classificatori di base sul validation set. Come per l'Algoritmo 1, si escludono innanzitutto i classificatori che hanno erroneamente etichettato una o più istanze nella regione locale. I conflitti che si possono presentare nei casi 2 e 3 vengono risolti nei modi seguenti, a seconda che si utilizzi il criterio basato sulla recall o sull'accuratezza.

**Caso in cui si utilizzi il criterio basato sulla recall:** Si individua la classe maggiormente presente tra le istanze della regione locale quindi si considera la recall che i classificatori hanno ottenuto su questa classe sul validation set. Si seleziona il classificatore con la recall più alta. I conflitti, ovvero più classificatori hanno la stessa recall (la massima ottenuta sul validation set), vengono risolti selezionando il classificatore che sul validation set ha l'errore medio assoluto più basso

**Caso in cui si utilizzi il criterio basato sull'accuratezza:** Si seleziona tra i classificatori quello che sul validation set ha ottenuto l'accuratezza più alta. I conflitti, ovvero più classificatori hanno la stessa accuratezza (la massima ottenuta sul validation set) vengono risolti selezionando il classificatore con l'errore medio assoluto più basso sul validation set

### 5.2.3 Algoritmo 3

L'Algoritmo 3 è un ibrido tra gli Algoritmi 1 e 2 in quanto usa per la gestione dei conflitti elementi usati in entrambi i metodi. Come per gli Algoritmi 1 e 2, durante la prima fase si escludono i classificatori che hanno erroneamente classificato alcune istanze nella regione locale. Successivamente, si considera l'insieme dei classificatori rimasti. I conflitti vengono risolti nel modo seguente:

- Rimangono più classificatori. I conflitti vengono gestiti come nell'Algoritmo 2.
- Non rimane nessun classificatore di base. I conflitti vengono gestiti come nell'Algoritmo 1.

Anche in questo caso, si può usare come criterio di competenza quello basato sulla recall o quello basato sull'accuratezza.

---

**Algoritmo 1**


---

```

1: function SELECT THE BEST CLASSIFIERS BASED ON VALIDATION AND TRAINING RESULTS( $I, VS, TS, C, S, Option$ ) ▷ Where
   I is an instance of the test set; VS - Validation Set; TS - Training Set; C is a pool of classifiers; S the local region;
   Option = (RECALL or ACCURACY)
   Output:  $C^*$  the most promising classifier for  $I$ 
2:  $P = C; B = 0; B_1 = 0$ 
3: Exclude from P the classifiers that have misclassified some instances in S
4: if P contains one classifier then
5:    $C^* = P$ 
6: end if
7: if P is empty AND (Option == RECALL) then
8:   Compute the class h of majority instances in S
9:   Put in B the classifiers from C with the highest Recall on h in VS
10:  if B contains more classifiers then
11:    Put in  $B_1$  the classifiers with the best Recall on h in TS
12:    if  $B_1$  contains more classifiers then
13:       $C^* =$  the classifier with the best Mean Absolute Error on TS in  $B_1$ 
14:    else
15:       $C^* = B$ 
16:    end if
17:  else
18:     $C^* = B$ 
19:  end if
20: end if
21: if (P is empty) AND (Option == ACCURACY) then
22:   Put in B the classifiers from C with the highest Accuracy in VS
23:   if B contains more classifiers then
24:     Put in  $B_1$  the classifiers from B with the best Accuracy in TS
25:     if  $B_1$  contains more classifiers then
26:        $C^* =$  the classifier with the best Mean Absolute Error on TS in  $B_1$ 
27:     else
28:        $C^* = B_1$ 
29:     end if
30:   else
31:      $C^* = B$ 
32:   end if
33: end if
34: if (P contains more classifiers) AND (Option == RECALL) then
35:   Compute the class h of majority instances in S
36:   Put in B the classifiers from P with the highest Recall on h in VS
37:   if B contains more classifiers then
38:     Put in  $B_1$  the classifiers from B with the best Recall on h in TS
39:     if  $B_1$  contains more classifiers then
40:        $C^* =$  the classifier from with the best Mean Absolute Error on TS in  $B_1$ 
41:     else
42:        $C^* = B_1$ 
43:     end if
44:   else
45:      $C^* = B$ 
46:   end if
47: end if
48: if (P contains more classifiers) AND (Option == ACCURACY) then
49:   Put in B the classifiers from P with the highest Accuracy in VS
50:   if B contains more classifiers then
51:     Put in  $B_1$  the classifiers from P with the highest Accuracy in TS
52:     if  $B_1$  contains more classifiers then
53:        $C^* =$  the classifier from with the best Mean Absolute Error on TS in  $B_1$ 
54:     else
55:        $C^* = B_1$ 
56:     end if
57:   else
58:      $C^* = B$ 
59:   end if
60: end if
61: return  $C^*$ 
62: end function

```

---

## 5.2 Approccio proposto per la progettazione di Multi Classificatori 89

---

---

### Algoritmo 2

```
function SELECT THE BEST CLASSIFIERS BASED ON VALIDATION RESULTS(I, VS, C, S, Option) ▷ Where I is an instance of
the test set; VS - Validation Set; C is a pool of classifiers; S the local region; Option = (RECALL or ACCURACY)
  Output: C* the most promising classifier for I
2:  P = C; B = 0; B1 = 0
  Exclude from P the classifiers that have misclassified some instances in S
4:  if P contains one classifier then
      C* = P
6:  end if
  if P is empty AND (Option == RECALL) then
8:    Compute the class h of majority instances in VS
    Put in B the classifiers from C with the highest Recall on h in VS
10:   if B contains more classifiers then
      C* = the classifier with the best Mean Absolute Error on VS in B
12:   else
      C* = B
14:   end if
  end if
  if (P is empty) AND (Option == ACCURACY) then
16:    Put in B the classifiers from C with the highest Accuracy in VS
18:    if B contains more classifiers then
      C* = the classifier from with the best Mean Absolute Error on VS in B
20:    else
      C* = B
22:    end if
  end if
  if P contains more classifiers AND (Option == RECALL) then
24:    Compute the class h of majority instances in S
26:    Put in B the classifiers from P with the highest Recall on h in VS
    if B contains more classifiers then
28:      C* = the classifier from with the best Mean Absolute Error on VS in B
    else
30:      C* = B
    end if
  end if
  if (P contains more classifiers) AND (Option == ACCURACY) then
34:    Put in B the classifiers from P with the highest Accuracy in VS
    if B contains more classifiers then
36:      C* is the classifier from with the best Mean Absolute Error on VR in B
    else
38:      C* = B
    end if
  end if
40:  return C*
42: end function
```

---



---

**Algoritmo 3**


---

```

1: function SELECT THE BEST CLASSIFIERS BASED ON HYBRID METHOD(I, VS, TS, C, S, Option) ▷ Where I is an
   instance of the test set; VS - Validation Set; TS - Training Set; C is a pool of classifiers; S the local region; Option =
   (RECALL or ACCURACY) Output:  $C^*$  the most promising classifier for I
2:    $P = C; B = 0; B_1 = 0$ 
3:   Exclude from P the classifiers that have misclassified some instances in VS
4:   if P contains one classifier then
5:      $C^* = P$ 
6:   end if
7:   if P is empty AND (Option == RECALL) then
8:     Compute the class h of majority instances in S
9:     Put in B the classifiers from C with the highest Recall on h in VS
10:    if B contains more classifiers then
11:      Put in  $B_1$  the classifiers with the best Recall on h in TS
12:    if  $B_1$  contains more classifiers then
13:       $C^* =$  the classifier with the best Mean Absolute Error on TS in  $B_1$ 
14:    else
15:       $C^* = B$ 
16:    end if
17:  else
18:     $C^* = B$ 
19:  end if
20: end if
21: if (P is empty) AND (Option == ACCURACY) then
22:   Put in B the classifiers from C with the highest Accuracy in VS
23:   if B contains more classifiers then
24:     Put in  $B_1$  the classifiers from B with the best Accuracy in TS
25:   if  $B_1$  contains more classifiers then
26:      $C^* =$  the classifier from with the best Mean Absolute Error on TS in  $B_1$ 
27:   else
28:      $C^* = B_1$ 
29:   end if
30: else
31:    $C^* = B$ 
32: end if
33: end if
34: if (P contains more classifiers) AND (Option == RECALL) then
35:   Compute the class h of majority instances in VS
36:   Put in B the classifiers from P with the highest Recall on h in VS
37:   if B contains more classifiers then
38:      $C^* =$  the classifier from with the best Mean Absolute Error on VS in B
39:   else
40:      $C^* = B$ 
41:   end if
42: end if
43: if (P contains more classifiers) AND (Option == ACCURACY) then
44:   Put in B the classifiers from P with the highest Accuracy in VS
45:   if B contains more classifiers then
46:      $C^* =$  the classifier from with the best Mean Absolute Error on VS in B
47:   else
48:      $C^* = B$ 
49:   end if
50: end if
51: return  $C^*$ 
52: end function

```

---

# Capitolo 6

## Validazione sperimentale

Il capitolo descrive il protocollo sperimentale e i risultati ottenuti applicando a dataset pubblici gli approcci di classificazione descritti nel Capitolo 5. In particolare, vengono presentati e sintetizzati tutti i test eseguiti su ciascun dataset e le valutazioni che ne sono scaturite, fornendo un'analisi oggettiva della metodologia utilizzata. I test eseguiti mirano a verificare la bontà degli approcci proposti in termini di accuratezza di classificazione. I dataset utilizzati si riferiscono a importanti problemi decisionali in ambito medico-clinico. Gli approcci sono stati implementati in JAVA utilizzando le API (Application Programming Interface) del software WEKA (Waikato Environment for Knowledge Analysis). Il Capitolo è organizzato come segue. Nella Sezione 9.1 una panoramica dei dataset mentre nella Sezione 9.2 una panoramica dei classificatori utilizzati. La Sezione 9.3 descrive il protocollo sperimentale utilizzato per l'esecuzione dei test. La Sezione 9.4 mostra invece i risultati ottenuti sui vari dataset per ciascun insieme di classificatori di base utilizzato.

### 6.1 Dataset

I sei dataset utilizzati per testare gli approcci proposti sono pubblici e disponibili su UCI [7]. Tutti i dataset si riferiscono a importanti problemi

decisionali dell'ambito medico-clinico, come di seguito descritto. Un riepilogo è mostrato in Tabella 6.1.

Tabella 6.1: Dataset

Dataset	Num. Istanze	Num. Attributi	Num. Classi
WDBC	569	30	2
WBC	699	9	2
WPBC	198	33	2
Cleveland	303	13	2
Mammographic Mass	961	5	2
Dermatology	366	34	6

### 6.1.1 Wisconsin Diagnostic Breast Cancer Dataset

Il Wisconsin Diagnostic Breast Cancer Dataset (WDBC) contiene dati di pazienti che si sono sottoposte a esame di agoaspirato di un nodulo mammario (Fine Needle Aspiration - FNA) ed è stato collezionato presso la University of Wisconsin Hospitals, Madison dal Dr. William H. Wolberg. Il FNA è una speciale biopsia che utilizza un ago sottile per prelevare cellule all'interno di un'area anomala del corpo umano. L'esame consiste nell'inserimento di un ago sottile (solo leggermente più spesso di quello delle normali siringhe) nel seno, fino a raggiungere il nodulo, da dove si aspira parte del contenuto da esaminare in laboratorio. Dopo l'acquisizione, i campioni prelevati vengono analizzati al microscopio e alcune immagini del microscopio vengono digitalizzate. Le feature che il dataset contiene sono state estratte da tali immagini digitalizzate. Esse descrivono le caratteristiche dei nuclei delle cellule presenti nelle immagini. Il dataset viene utilizzato per diagnosticare se un nodulo al seno è di natura benigna (per esempio una cisti) o se si tratta di un tumore. Si compone di 569 istanze (357 casi di noduli benigni e 212 maligni). Ogni istanza contiene 30 feature a valori reali oltre all'etichetta di classe. Per ogni nucleo cellulare sono state osservate dieci caratteristiche e per ciascuna caratteristica sono state calcolate la media, l'errore standard, e il valore peggiore (o più grande) osservato. Le caratteristiche misurate sono riportate in Tabella 6.2. Il dataset non contiene valori mancanti.

Tabella 6.2: Caratteristiche del WDBC Dataset

Attributo	Descrizione	Tipo	Valori
Radius	Raggio del nucleo cellulare	reale	-
Texture	Trama del nucleo misurata come deviazione standard dei valori in scala di grigi	reale	-
Perimeter	Perimetro del nucleo cellulare	reale	-
Area	Area del nucleo cellulare	reale	-
Smoothness	Scorrevolezza dei nuclei	reale	-
Compactness	Compattezza	reale	-
Concavity	Concavità	reale	-
Concave points	Punti Concavi	reale	-
Symmetry	Simmetria	reale	-
Fractal dimension	Dimensione frattale	reale	-
Diagnosis	Diagnosi	Binario	Benign=B Malignant=M

### 6.1.2 Wisconsin Breast Cancer Dataset

Il Wisconsin Breast Cancer Dataset (WBC) contiene dati di pazienti che si sono sottoposte a esame di agoaspirato di un nodulo mammario (Fine Needle Aspiration - FNA) ed è stato collezionato presso la University of Wisconsin Hospitals, Madison dal Dr. William H. Wolberg. Le feature sono state estratte dalle immagini digitalizzate durante l'analisi al microscopio dei campioni prelevati durante l'esame FNA. Esse descrivono, infatti, le caratteristiche dei nuclei delle cellule presenti nell'immagine. Il dataset viene utilizzato per distinguere i noduli maligni da quelli benigni. Si compone di 699 istanze (458 casi benigni e 241 casi maligni). Ogni nucleo cellulare è descritto attraverso nove feature a valori reali più l'attributo di classe. Le caratteristiche misurate sono riassunte in Tabella 6.3. Il dataset ha qualche valore mancante per alcune feature.

Tutti i parametri considerati sono importanti per determinare se le cellule sono tumorali o meno. Infatti, le cellule benigne tendono ad essere raggruppate in mono strati, mentre le cellule tumorali sono spesso raggruppate in

Tabella 6.3: Caratteristiche del WBC Dataset

Attributo	Descrizione	Tipo	Valori
Clump Thickness	Spessore degli strati delle cellule	Ordinale	1-10
Uniformity of Cell Size	Uniformità della dimensione della cellula	Ordinale	1-10
Uniformity of Cell Shape	Uniformità della forma della cellula	Ordinale	1-10
Marginal Adhesion	Adesione marginale	Ordinale	1-10
Single Epithelial Cell Size	Dimensione di una singola cellula epiteliale	Ordinale	1-10
Bare Nuclei	Bare Nuclei	Ordinale	1-10
Bland Chromatin	Bland Cromatina	Ordinale	1-10
Normal Nucleoli	Nucleoli Normali	Ordinale	1-10
Mitoses	Mitosi	Ordinale	1-10
Diagnosis	Diagnosi	Binario	Benign=B Malignant=M

multistrato (Clump Thickness). Le cellule malate tendono a variare in dimensione e forma (Uniformity of Cell Size /Shape) al contrario di quelle sane. Mentre, le cellule sane tendono ad aderire l'una con l'altra (Marginal Adhesion), quelle tumorali tendono a perdere questa capacità. Le cellule epiteliali che risultano essere significativamente ampliate, potrebbero appartenere a una cellula maligna. Bare nuclei è un termine usato per nuclei che non sono circondati da citoplasma e sono in genere visti come tumori benigni. La Bland cromatina descrive una texture uniforme del nucleo che viene tipicamente osservata in cellule benigne. Nelle cellule tumorali la bland cromatina tende invece ad essere più grossolana. I nucleoli normali sono piccole strutture osservate nel nucleo: nelle cellule normali il nucleolo, se visibile, è di solito molto piccolo; nelle cellule tumorali invece i nucleoli diventano più grandi e, a volte, ce ne sono più di uno. Infine, la mitosi è il processo in cui la cellula si divide e si replica formando due cellule figlie identiche. I patologi determinano il tipo di tumore contando il numero di mitosi.

### 6.1.3 Wisconsin Prognostic Breast Cancer Dataset

Il Wisconsin Prognostic Breast Cancer Dataset (WPBC) è stato collezionato presso la University of Wisconsin Hospitals, Madison dal Dr. William H. Wolberg. Ogni istanza contiene i dati di follow-up di pazienti affette da carcinoma mammario invasivo con nessuna evidenza di metastasi al momento della diagnosi. Il dataset viene utilizzato per prevedere se il tumore potrebbe ripresentarsi oppure no. Si compone di 198 istanze (151 casi in cui il tumore non si è ripresentato e 47 in cui si è ripresentato). Ciascuna istanza è costituita da 33 caratteristiche a valori reali più la prognosi. Come per i WBC Dataset e il WDBC Dataset, ogni istanza descrive le caratteristiche del nucleo cellulare presente nell'immagine digitalizzata durante l'esame al microscopio dei campioni estratti attraverso l'esame FNA. Per ogni nucleo cellulare sono state osservate, così come per il WDBC, dieci caratteristiche e per ciascuna di esse sono stati calcolati il valore medio, errore standard e valore peggiore (o più grande) osservato. Rispetto al dataset WDBC, nel WPBC sono state osservate due ulteriori caratteristiche: il diametro del tumore asportato (misurato in centimetri) e lo stato dei linfonodi ovvero, il numero di linfonodi ausiliari positivi osservati al momento dell'intervento chirurgico. Il dataset riporta inoltre, l'intervallo temporale trascorso tra l'intervento chirurgico e la ricomparsa del tumore nel caso in cui esso si sia ripresentato oppure l'intervallo temporale tra l'intervento chirurgico e l'ultima visita se il tumore non si è ripresentato. Le caratteristiche misurate sono riportate in Tabella 6.4. Il dataset contiene valori mancanti.

Tabella 6.4: Caratteristiche del WPBC Dataset

Attributo	Descrizione	Tipo	Valori
Time	Intervallo temporale	reale	
Radius	Raggio del nucleo cellulare	reale	-
Texture	Trama del nucleo cellulare	reale	-
Perimeter	Perimetro del nucleo cellulare	reale	-
Area	Area del nucleo cellulare	reale	-
Smoothness	Scorrevolezza dei nuclei	reale	-
Compactness	Compattezza	reale	-
Concavity	Concavità	reale	-
Concave points	Punti Concavi	reale	-
Symmetry	Simmetria	reale	-
Fractal dimension	Dimensione frattale	reale	-
Tumor size	Diametro del tumore asportato	reale	
Lymph node status	Numero di linfonodi ausiliari positivi osservati al momento dell'intervento chirurgico	reale	
Outcome	Prognosi	Binario	Recurr=R NonRecurr=N

#### 6.1.4 Cleveland Heart Disease Dataset

Il Cleveland Heart Disease Database è stato collezionato presso il Medical Center di Long Beach e la Cleveland Clinic Foundation dal Dr. Robert Detrano. Il database contiene 303 istanze e viene utilizzato per diagnosticare la presenza o l'assenza di una malattia cardiaca. Ogni istanza contiene 13 attributi più la diagnosi. Gli attributi sono descritti in Tabella 6.5. Sei istanze presentano attributi mancanti.



Tabella 6.5: Caratteristiche del Cleveland Heart Disease Dataset

Attributo	Descrizione	Tipo	Valori
Age	Età in anni	intero	
Sex	Sesso	nominale	0: Female 1: Male
Cp	Tipo di dolore al petto	nominale	1: typical angina 2: atypical angina 3: non-anginal pain 4: asymptomatic
Trestbps	Pressione del sangue a riposo misurata in mm Hg	reale	
Chol	Colesterolo sierico misurato in mg/dl	reale	
Fbs	Glicemia a digiuno > 120 mg/dl	nominale	0: False 1: True
Restecg	Risultati elettrocardiografici a riposo	nominale	0: normale 1: onda ST-T anormale 2: ipertrofia ventricolare sinistra probabile o certa in base ai criteri di Este
Thalach	Massima frequenza cardiaca raggiunta	reale	
Exang	Angina indotta da esercizio	nominale	0: No 1: Yes
Oldpeak	Depressione ST indotta da esercizio fisico rispetto al riposo		1: pendenza alta
Slope	Pendenza del picco di esercizio del tratto ST	nominale	2: nessuna pendenza 3: pendenza bassa
Ca	Numero di grossi vasi (0-3) colorato da fluoroscopia	intero	
Thal	Frequenza cardiaca	nominale	Normale=3 Difetto fisso=6 Difetto reversibile=7
Diagnosis	Stato della malattia	Binario	0: restringimento del diametro <50% 1: restringimento del diametro > 50%

### 6.1.5 Mammographic Mass Dataset

Il Mammographic Mass Dataset è stato collezionato presso l'Institute of Radiology, Gynaecological Radiology, University Erlangen-Nuremberg. Il dataset contiene i dati estratti dall'analisi di mammografie digitali raccolte tra il 2003 e il 2006 e viene usato per predire se una massa tumorale potrebbe essere benigna o maligna. I dati sono estratti da immagini mammografiche analizzate attraverso il sistema di reporting BI-RADS (Breast Imaging Reporting and Data System). Il tool BI-RADS, sviluppato dall'American College of Radiology, è stato progettato principalmente per standardizzare i referti della mammografia e ridurre la confusione nelle interpretazioni delle immagini. Il Mammographic Mass Dataset è costituito da 961 istanze (516 con diagnosi di massa benigna e 445 con diagnosi di massa maligna). Ciascuna istanza è descritta attraverso 5 attributi più la diagnosi. Tutti gli attributi sono riportati in Tabella 6.6. Alcune istanze presentano attributi mancanti. La valutazione BI-RADS assume valori da 1 (assolutamente benigna) a 5 (altamente suggestivo di malignità). Tali valori sono stati assegnati attraverso un processo a doppia revisione da parte dei medici. Il dataset contiene inoltre, l'età del paziente e tre attributi ottenuti attraverso il tool BI-RADS: mass shape, mass margin e mass density (forma, bordo e densità della massa).

Tabella 6.6: Caratteristiche del Mammographic Mass Dataset

Attributo	Descrizione	Tipo	Valori
BI-RADS assessment	Valutazione con BI-RADS	Ordinale	Da 1 a 5
Age	Età del paziente in anni	Intero	
Mass Shape	Forma della massa tumorale	Nominale	round=1 oval=2 lobular=3 irregular=4
Mass Margin	Bordo della massa tumorale	Nominale	circumscribed=1 microlobulated=2 obscured=3 ill-defined=4 spiculated=5
Mass Density	Densità della massa tumorale	Ordinale	high=1 iso=2 low=3 fat-containing=4
Severity	Etichetta di classe	Binario	benign=0 malignant=1

### 6.1.6 Dermatology Dataset

Il dermatology database è stato collezionato presso le Università Gazi e Bilkent e contiene i dati di pazienti affetti da malattie eritemato-desquamative. Tutte le malattie di questo gruppo condividono le caratteristiche cliniche di eritema e desquamazione, con pochissime differenze. Di solito, per la diagnosi è necessaria una biopsia. Ma poiché queste malattie condividono anche molte caratteristiche istopatologiche a volte, anche con la biopsia purtroppo l'incertezza rimane. Non solo, molto spesso allo stadio iniziale una malattia può mostrare le caratteristiche di un'altra e mostrare le caratteristiche proprie solo nelle fasi successive. Il database si compone di 366 istanze. Ogni istanza contiene 34 attributi. 12 attributi si riferiscono alla valutazione clinica dei pazienti mentre 22 derivano dall'esame istopatologico eseguito sui campioni di pelle prelevati. Al contrario degli altri dataset considerati, in questo caso la classificazione è multi-classe infatti, la diagnosi differenziale viene fatta tra sei diverse malattie: psoriasi, dermatite seborroica, lichen planus, pitiriasi rosea, dermatite cronica, e pitiriasi rubra pilaris . Gli attributi sono riportati in Tabella 6.7 e in Tabella 6.8. Il dataset contiene 8 valori mancanti per l'attributo età.

L'attributo family history (familiarità) assume il valore 1 se una di queste malattie è stata osservata anche tra i familiari, 0 altrimenti. L'attributo età rappresenta semplicemente l'età del paziente. Ogni altra caratteristica (clinica e istopatologica) assume un valore nel range da 0-3 dove 0 indica che l'attributo non era presente, 3 indica che era presente con il valore più alto possibile mentre 1 e 2 indicano valori intermedi.

Tabella 6.7: Caratteristiche del Dermatology Dataset: dodici attributi che derivano dalla valutazione clinica

Attributo	Descrizione	Tipo	Valori
Erythema	Eritema	intero	0, 1, 2, 3
Scaling	Ridimensionamento	intero	0, 1, 2, 3
Definite borders	Confini definiti	intero	0, 1, 2, 3
Itching	Prurito	intero	0, 1, 2, 3
Koebner phenomenon	Fenomeno di Koebner	intero	0, 1, 2, 3
Polygonal papules	Papule poligonali	intero	0, 1, 2, 3
Follicular papules	Papule follicolari	intero	0, 1, 2, 3
Oral mucosal involvement	Coinvolgimento della mucosa orale	intero	0, 1, 2, 3
Knee and elbow involvement	Coinvolgimento del gomito e del ginocchio	intero	0, 1, 2, 3
Scalp involvement	Coinvolgimento del cuoio capelluto	intero	0, 1, 2, 3
Family history	Familiarità	intero	0, 1
Age	Età	intero	

Tabella 6.8: Caratteristiche del Dermatology Dataset: ventidue attributi che derivano dall'esame istopatologico

Attributo	Descrizione	Tipo	Valori
Melanin incontinence	Melanina accumulata negli strati superficiali dell'epidermide	intero	0, 1, 2, 3
Eosinophils in the infiltrate	Infiltrati eosinofili	intero	0, 1, 2, 3
PNL infiltrate	Infiltrati PNL	intero	0, 1, 2, 3
Fibrosis of the papillary dermis	Fibrosi del derma papillare	intero	0, 1, 2, 3
Exocytosis	Esocitosi	intero	0, 1, 2, 3
Acanthosis	Acanthosi (ispessimento dell'epidermide)	intero	0, 1, 2, 3
Hyperkeratosis	Ipercheratosi	intero	0, 1, 2, 3
Parakeratosis	Paracheratosi	intero	0, 1, 2, 3
Clubbing of the rete ridges	Clubbing delle creste	intero	0, 1, 2, 3
Elongation of the rete ridges	Allungamento delle creste	intero	0, 1, 2, 3
Thinning of the suprapapillary epidermis	Assottigliamento dell'epidermide soprapapillare	intero	0, 1, 2, 3
Spongiform pustule	Pustule spongiformi	intero	0, 1, 2, 3
Munro microabcess	Micro ascesso di Munro (strato corneo dell'epidermide)	intero	0, 1, 2, 3
Focal hypergranulosis	Iper granulosi focale	intero	0, 1, 2, 3
Disappearance of the granular layer	Scomparsa dello strato granulare	intero	0, 1, 2, 3
Vacuolisation and damage of basal layer	Vacuolizzazione e danneggiamento dello strato basale	intero	0, 1, 2, 3
Spongiosis	Spongiosi	intero	0, 1, 2, 3
Saw-tooth appearance of retes	Aspetto a dente di sega	intero	0, 1, 2, 3
Follicular horn plug	Spina corno follicolare	intero	0, 1, 2, 3
Perifollicular parakeratosis	Paracheratosi follicolare	intero	0, 1, 2, 3
Inflammatory mononuclear infiltrate	Infiltrato mononucleare infiammato	intero	0, 1, 2, 3
Band-like infiltrate	Infiltrato dermico a forma di fascia	intero	0, 1, 2, 3
class	Etichetta di classe	Nominale	1: Psoriasis 2: Seboric dermatitis 3: Lichen planus 4: Pityriasis rosea 5: Chronic dermatitis 6: Pityriasis rubra pilaris

## 6.2 Classificatori

Gli insiemi di classificatori di base usati nei test sono stati costruiti a partire da un set iniziale di cinque classificatori:

- **Naive Bayes (NB)**
- **Multilayer Perceptron (MLP)**
- **Support Vector Machine (SMO)**
- **K-Nearest Neighbour (Ibk)**
- **Albero di decisione (J48)**

In particolare, si è scelto di considerare classificatori ampiamente usati in diversi compiti di classificazione e che usano per l'apprendimento approcci metodologici differenti. Le implementazioni dei classificatori usate sono quelle fornite dal software WEKA. In base allo specifico compito di classificazione sono state usate sia SVM con kernel lineari che con kernel gaussiani. In Weka questo classificatore è chiamato SMO (Sequential Minimal Optimization). Come albero di decisione è stato utilizzato il J48 che implementa l'algoritmo C4.5 di Kohavi and Quinlan [49]. Il J48 usa l'Information Gain per selezionare gli attributi che determinano i test sui nodi interni dell'albero. L'Information Gain rappresenta la riduzione attesa di entropia conseguente al partizionamento degli esempi in accordo al test scelto. Per prendere la decisione viene utilizzato l'attributo con l'Information Gain più alto. Il J48 è in grado di gestire sia attributi continui che discreti, dati di training con i valori degli attributi mancanti e attributi con costi differenti. Inoltre, fornisce un'opzione che permette la potatura degli alberi in seguito alla loro creazione. In Weka il classificatore K-Nearest Neighbour è chiamato Ibk (Instance-based method with parameter k).

## 6.3 Protocollo Sperimentale

Come già ampiamente discusso nei capitoli precedenti, gli approcci proposti seguono un processo di classificazione supervisionato. Ciascun dataset è stato suddiviso in tre insiemi disgiunti: training set, validation set e test set. Il training set è stato utilizzato per costruire e addestrare i classificatori di base. Il validation set per valutare le prestazioni che ciascun classificatore di base ha nel riconoscere nuove istanze mentre il test set contiene le istanze a cui si vuole assegnare un'etichetta. Sui dataset non è stato eseguito alcun processo di selezione degli attributi (feature selection). Tutti i dataset invece sono stati sottoposti a una procedura di normalizzazione delle feature nel range [0-1]. Per ogni dataset sono stati settati i parametri ottimali di ciascun classificatore e su ognuno sono stati testati diversi set di classificatori di base. In particolare, i cinque classificatori sono stati testati in gruppi di due, tre e quattro. Il protocollo sperimentale è basato su 10-fold cross validation stratificata. La regione di competenza di ciascun classificatore è stata costruita sul validation set. Le prestazioni degli approcci proposti sono state confrontate con quelle di altri approcci noti in letteratura. In particolare, sono stati considerati il metodo DCS LA (Dynamic Classifier Selection Local Accuracy) in entrambe le versioni di LCA (Local Class Accuracy) e OLA (Overall Local Accuracy) riportati entrambi in Woods et al. [89] e il metodo di voto a maggioranza (Vote) Cha Zhang [17]. Poiché non sono stati trovati in letteratura lavori in cui le due tecniche basate su DCS LA sono state implementate e testate in campo medico sui dataset di riferimento, si è provveduto all'implementazione delle tecniche in java e all'esecuzione dei test. Tutti i metodi di riferimento sono stati testati nella versione originale dell'algoritmo utilizzando la 10 fold cross validation. Per i classificatori di base sono stati impostati gli stessi parametri utilizzati per gli altri approcci. Per quanto riguarda il metodo DCS LA entrambe le versioni dell'algoritmo sono state applicate diverse volte ad uno stesso dataset variando ogni volta il numero di vicini utilizzati per la costruzione della regione di competenza. In generale, il range considerato è 1-20. Tale range varia comunque a seconda



delle istanze contenute nel validation set di ciascun dataset. Al fine di confrontare tali tecniche note con gli approcci proposti in questa tesi, per ciascun set di classificatori sono state considerate le prestazioni migliori ottenute da ciascun algoritmo nel range considerato. Tale procedura è risultata essere necessaria per avere un confronto più equo possibile tra i metodi DCS LA e gli approcci proposti dal momento che, i metodi DCS LA utilizzano per la costruzione della regione di competenza l'algoritmo KNN nella versione tradizionale. Per lo stesso motivo, la regione di competenza è stata definita anche per le tecniche note sul validation set.

### 6.3.1 10-fold cross validation

La fold cross validation è una tecnica statistica utilizzata in ML e Data Mining per evitare il problema dell'overfitting. Si parla di overfitting (o eccessivo adattamento) quando un modello si adatta eccessivamente ai dati su cui viene addestrato ma non è in grado di generalizzare e quindi di riconoscere in maniera adeguata dati che non ha già visto. In generale, quando si parla di classificazione si divide l'insieme dei dati in due parti: training set e test set. Il training set viene utilizzato per la costruzione del modello mentre il test set viene utilizzato per la successiva validazione. Per ottenere campioni il più possibili rappresentativi, i due insiemi devono essere estratti in maniera casuale. Tipicamente i 2/3 del dataset costituiscono il training set e 1/3 costituisce il test set. Con la  $k$  fold cross validation si creano più partizioni casuali dello stesso dataset e si ripete su ciascuna di essa la procedura di training e test set. In particolare, si divide il dataset in  $k$  partizioni di dimensioni uguali e indipendenti tra loro. A ogni ripetizione della procedura, una partizione viene usata come test set mentre le restanti come training set. Nella fold cross validation stratificata i sottoinsiemi sono partizionati in modo che la distribuzione degli oggetti rispetto alla classe da predire in ogni fold, è approssimativamente la stessa riscontrabile nei dati iniziali.

In questa tesi, per tutti gli esperimenti, è stata utilizzata la 10-fold cross validation stratificata. Il dataset è stato inizialmente suddiviso in maniera

casuale in 10 parti di dimensioni uguali. Ad ogni ripetizione (10 in totale) una partizione è stata usata come test e 9 come training set. Il 50 % del test set è stato utilizzato come validation set. Nello specifico, sulla partizione test è stata applicata una 2 fold cross validation in modo da dividere questo insieme in validation set e test set. La partizione è stata dunque divisa in maniera casuale in due parti uguali e ad ogni ripetizione (2 in totale) una parte è stata usata come test e l'altra come validation set.

### 6.3.2 Tuning dei parametri

La maggior parte degli algoritmi di classificazione possiede diversi parametri che ne regolano la fase di addestramento e di conseguenza il funzionamento. Come nel caso della scelta dell'algoritmo di classificazione, anche la scelta dei parametri ottimali può influire pesantemente sulle prestazioni del nostro sistema. La scelta dei parametri del classificatore risulta quindi di primaria importanza. Nel caso specifico, per la scelta dei parametri ottimali di ciascun algoritmo di classificazione su uno specifico dataset è stato usato il meta-classificatore di WEKA chiamato `CVParameterSelection`.

Nella Tabella 6.9 sono riportati i parametri principali di ogni classificatore che è possibile settare in Weka e i corrispondenti valori di default e/o valori ammissibili. I parametri dei kernel di una SVM sono riportati nella Tabella 6.10.

Nella Tabella 6.12 sono riportati i parametri settati per ogni classificatore su ciascun dataset. Per i parametri non specificati sono stati usati i valori di default.

Tabella 6.9: Parametri principali settabili in Weka per i classificatori usati

Classificatore	Parametro	Descrizione	Valore di default
SMO	-C	Costante di penalizzazione	1
	-N	Trasformazioni nei dati	0: normalizza 1 : standardizza 2 : nessuno dei due
	-L	Parametro di tolleranza	$1.0e - 3$
	-P	Valore di Epsilon per l'errore di arrotondamento	$1.0E - 12$
	-V	Numero di folds per la cross validation	-1
	-K	Tipo di kernel	PolyKernel
MLP	-L	Tasso di apprendimento per BP. Valore compreso tra 0 e 1	0.03
	-M	Momentum per BP. Valore compreso tra 0 e 1	0.02
	-N	Numero di epoche dell'addestramento	500
	-V	Dimensione del validation set. Valore compreso tra 0 e 100	0
	-E	Numero di errori consecutivi ammessi per la convalida del test. Valore $> 0$	20
	-B	Filtro NominalToBinary	Non settato
	-H	Numero di strati nascosti della rete	a
	-C	Se settato, le classi di tipo numerico non sono normalizzate	Non settato
	-I	Gli attributi non sono normalizzati	Non settato
-R	Non consente il reset della rete	Non settato	
NB	-K	Stimatore di densità del kernel per gli attributi numerici o distribuzione normale	Non settato
	-D	Discretizzazione supervisionata per attributi numerici	Non settato
J48	-U	Se settato, albero non potato	Non settato
	-C	Soglia di confidenza per la potatura dell'albero	0.25
	-M	Numero minimo di istanze usate per foglia	2
	-R	Se settato permette di ridurre gli errori di potatura	Non settato
	-N	Numero di fold usati per ridurre l'errore di potatura	3
	-B	Se settato l'albero usa solo split binari	Non settato
-Q	Seme usato per i dati casuali	1	
Ibk	-I	Pesa i vicini con l'inverso della loro distanza. Si usa quando $k > 1$	Non settato
	-F	Il peso dei vicini è calcolato come $1 -$ la loro distanza. Si usa quando $k > 1$	Non settato
	-K	Numero di vicini utilizzati nella classificazione	1
	-E	Minimizza l'errore quadratico medio piuttosto che usare l'errore medio assoluto quando si utilizza l'opzione -X con la previsione numerica	Non settato
	-W	Numero massimo di istanze di training mantenute. Le istanze di formazione vengono eliminate con politica FIFO	0= nessuna
	-X	Seleziona il numero di vicini più vicini tra 1 e il valore di K specificato utilizzando la valutazione hold-one-out sui dati di addestramento. Si usa quando $k > 1$	Non settato
	-A	Specifica l'algoritmo di ricerca per i vicini da usare	LinearNNSearch

Tabella 6.10: Parametri settabili in Weka per Kernel Polinomiale e Gaussiano

Parametro	Descrizione	Valore di default
-C	Dimensione della memoria cache	250007
-E	Esponente del kernel polinomiale: $k(x_1, x_2) = (x_1 \cdot x_2 + 1)^E$	1
G	Esponente del kernel gaussiano: $k(x_1, x_2) = \exp(-\frac{\ x_1 - x_2\ ^2}{2\sigma^2})$	0.01

Tabella 6.11: Funzioni Weka per calcolare la distanza

Funzioni di distanza per la ricerca dei vicini	
weka.core.EuclideanDistance	Distanza euclidea
weka.core.Manhattan Distance	Distanza di Manhattan
weka.core.ChebyshevDistance	Distanza di Chebyshev

Tabella 6.12: Set dei parametri ottimali per SMO, MLP e NB

Dataset	SMO			MLP			Naive Bayes			J48			IBk			
	C	N	E	L	M	I	G	K	D	C	N	K	I	F	A	Dist
	(Poly K) (Gaus. K)															
WDBC	5	set	-	0.08	0.4	0.8	0.08	-	-	-	4	4	-	-	Lin	Euc
WBC	1	set	-	0.05	0.1	0.2	0.05	set	-	-	4	4	-	-	Lin	Euc
Cleveland	1	set	-	0.08	1	0.6	0.08	-	set	0.1	-	6	set	Lin	Man	Man
Mamm. Mass	9.75	set	2	-	0.2	0.4	-	-	-	0.25	-	9	-	-	Lin	Man
WPBC	10	set	-	0.09	0.1	1	0.09	-	set	-	4	7	-	-	Lin	Euc
Dermatology	3.5	set	1	-	0.2	0.1	-	-	-	-	3	3	-	-	Lin	Euc

## 6.4 Risultati e discussione

In questa sezione si descrive la campagna di sperimentazione eseguita sui sei dataset descritti precedentemente. In particolare si confrontano i risultati ottenuti tramite gli approcci proposti sia con i risultati ottenuti attraverso i metodi DCS LA, nelle versioni LCA e OLA [89], che con quelli ottenuti applicando il metodo di voto a maggioranza [17].

Si è osservato che tutte e tre le versioni dell'algoritmo proposto hanno prestazioni simili. In generale, le tre versioni si comportano meglio su alcuni dataset. Sui dataset WBC, WDBC, WPBC e Dermatology le prestazioni sono da ritenersi soddisfacenti. Su dataset come Cleveland e Mammographic Mass si sono registrate per molti insiemi di classificatori di base prestazioni solo leggermente inferiori a DCS LA LCA, DCS LA OLA e Vote.

In base allo specifico problema decisionale i KM di un CDSS saranno quindi costituiti solo da quei MCS che da un confronto dei risultati presentano le performance migliori. Nel Capitolo 7 è illustrato lo sviluppo di un prototipo di CDSS per la diagnosi del tumore al seno basato sul WDBC Dataset. Sulla base dei risultati ottenuti su questo dataset, infatti, si è scelto di implementare il Multi Classificatore NB-SMO basato sull'Algoritmo 2. Tale Multi Classificatore è tra quelli con le prestazioni più alte pari al 98.24 % .

Di seguito sono discussi in dettaglio i risultati ottenuti su ogni singolo dataset. Le Tabelle 6.13-6.17 riportano i risultati in termini di accuratezza ottenuti dai vari metodi. Su ciascuna riga è specificato:

- l'insieme dei classificatori di base considerato
- l'accuratezza ottenuta con l'Algoritmo 1, l'Algoritmo 2 e l'Algoritmo 3 sia nella versione che usa la recall come criterio di competenza che nella versione che usa l'accuratezza
- l'accuratezza ottenuta dal metodo DCS LA nelle due versioni LCA e OLA
- l'accuratezza ottenuta con il metodo del Voto a maggioranza

Più in dettaglio, le prime dieci righe riportano i risultati ottenuti considerando un insieme costituito da due classificatori di base; le successive dieci riportano i risultati ottenuti con un insieme costituito da tre classificatori di base mentre, le ultime cinque i risultati ottenuti con insiemi costituiti da quattro classificatori di base. Per valutare le performance dei diversi metodi in funzione degli insiemi di classificatori di base utilizzati si confrontano i risultati ottenuti dai vari approcci su ciascuna riga: i valori migliori di accuratezza ottenuti sono evidenziati in grassetto. Nelle seguenti sezioni verranno confrontati per tutti i dataset:

1. I risultati ottenuti tramite l'Algoritmo 1, l'Algoritmo 2 e l'Algoritmo 3, nella versione che usa la recall come criterio di competenza, con i risultati ottenuti dall'algoritmo DCS LA-LCA. In questo caso, tutti gli approcci utilizzano come criterio di competenza le prestazioni dei classificatori di base su una singola etichetta di classe
2. I risultati ottenuti tramite l'Algoritmo 1, l'Algoritmo 2 e l'Algoritmo 3, nella versione che usa l'accuratezza come criterio di competenza, e i risultati dell'algoritmo DCS LA-OLA. In questo caso, tutti gli approcci utilizzano come criterio di competenza l'accuratezza seppure calcolata su regioni differenti
3. I risultati ottenuti da Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con la tecnica del Vote

#### 6.4.1 Risultati su Wisconsin Diagnostic Breast Cancer Dataset

La Tabella 6.13 riporta i risultati in termini di accuratezza ottenuti nella fase di sperimentazione sul dataset WDBC.

Si confrontano innanzitutto, i risultati ottenuti tramite l'Algoritmo 1, l'Algoritmo 2 e l'Algoritmo 3, nella versione che usa la recall come criterio di competenza, con i risultati ottenuti dall'algoritmo DCS LA-LCA. Per quanto



riguarda l'Algoritmo 1, la tabella mostra che in 11 insiemi su 25 si ottengono prestazioni uguali a quelle ottenute con il metodo DCS LA-LCA usando gli stessi set di classificatori di base. L'Algoritmo 2 e l'Algoritmo 3 ottengono invece, entrambi in 12 insiemi su 25 prestazioni uguali al metodo DCS LA-LCA. Il grafico di Figura 6.1 riporta tale confronto.

Si confrontano successivamente, i risultati ottenuti tramite l'Algoritmo 1, l'Algoritmo 2 e l'Algoritmo 3, nella versione che usa l'accuratezza come criterio di competenza, e i risultati dell'algorithm DCS LA-OLA. Con l'Algoritmo 1, in 7 insiemi su 25 si registrano prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-OLA usando gli stessi classificatori di base. In particolare, in 4 insiemi su 7 si registrano prestazioni migliori. Gli insiemi di classificatori NB-Ibk e J48-NB-Ibk permettono di ottenere i valori di accuratezza più alti sull'intero dataset tra tutti gli algoritmi che usano gli stessi classificatori di base. L'Algoritmo 2 in 12 insiemi su 25 ottiene prestazioni migliori o uguali rispetto al metodo DCS LA-OLA. In 6 insiemi su 12 ottiene prestazioni migliori. Con gli insiemi costituiti da SMO-Ibk-NB e J48-MLP-NB si registrano i valori di accuratezza più alti per il dataset WDBC tra tutti gli algoritmi che usano gli stessi classificatori di base. Anche l'Algoritmo 3 in 12 insiemi su 25 ottiene prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-OLA e in particolare, in 6 insiemi su 12 si registrano prestazioni migliori. Con gli insiemi costituiti da SMO-Ibk-NB e J48-MLP-NB si registrano i valori di accuratezza migliori ottenuti sull'intero dataset tra tutti gli algoritmi che usano gli stessi classificatori di base. Il grafico di Figura 6.2 riporta tale confronto.

Infine, i risultati ottenuti con i tre algoritmi nelle due versioni sono confrontati con i risultati del Vote. Complessivamente, rispetto a tale tecnica, in 15 insiemi su 25 almeno uno degli algoritmi proposti ottiene prestazioni uguali o migliori. Il grafico di Figura 6.3 riporta tale confronto.

In generale, si osserva che con gli insiemi di classificatori di base NB-Ibk, SMO-Ibk-NB, J48-MLP-NB, J48-NB-Ibk gli approcci proposti permettono di ottenere l'accuratezza migliore sul dataset WDBC.

Tabella 6.13: Risultati su WDBC Dataset in termini di accuratezza. Per i tre Algoritmi proposti si specifica la versione basata su Recall (Ver Rec) e Accuratezza (Ver Acc)

Classificatori	Alg. 1		Alg. 2		Alg. 3		Best DCS		Vote
	Ver Rec	Ver Acc	Ver Rec	Ver Acc	Ver Rec	Ver Acc	LA LCA	LA OLA	
NB-SMO	98,24	98,24	98,24	98,24	98,24	98,24	98,24	98,24	98,24
MLP-SMO	97,36	97,36	98,24	98,24	98,24	98,24	98,24	98,24	98,24
J48-SMO	97,89	97,54	97,54	97,54	97,54	97,54	98,07	97,72	97,54
Ibk-SMO	97,54	97,54	98,24	98,07	98,24	98,07	98,24	98,24	98,24
MLP-NB	97,36	97,36	97,54	97,54	97,54	97,54	97,54	97,36	96,31
MLP-J48	96,84	96,84	96,84	96,84	96,84	96,84	96,84	97,19	96,66
MLP-Ibk	97,54	97,36	97,19	97,01	97,19	97,01	97,54	97,36	97,54
J48-NB	94,38	94,38	94,55	94,55	94,55	94,55	94,9	95,08	93,32
J48-Ibk	97,19	97,01	95,96	95,96	95,96	95,96	97,19	96,13	95,61
NB-Ibk	97,19	97,72	97,01	97,01	97,01	97,01	97,19	96,31	93,5
SMO-Ibk-NB	97,54	97,54	97,89	98,07	97,89	98,07	97,89	97,54	97,89
SMO-Ibk-MLP	97,54	97,54	98,24	98,07	98,24	98,07	98,24	98,07	98,24
SMO-Ibk-J48	97,01	97,01	97,54	97,36	97,54	97,36	97,72	97,54	98,24
SMO-J48-MLP	96,84	96,84	97,54	97,54	97,54	97,54	97,72	97,54	97,89
SMO-J48-NB	97,72	97,54	96,66	97,54	96,66	97,54	97,89	96,49	96,13
SMO-MLP-NB	97,36	97,36	98,24	98,24	98,24	98,24	98,24	97,89	97,89
J48-MLP-NB	96,84	96,84	95,61	97,01	95,61	97,01	96,84	95,61	95,43
J48-MLP-Ibk	96,84	96,84	96,49	96,49	96,49	96,49	96,84	97,54	97,89
J48-NB-Ibk	96,66	97,01	95,78	95,61	95,78	95,61	96,66	96,49	95,08
MLP-NB-Ibk	97,54	97,36	97,01	97,19	97,01	97,19	97,54	97,36	97,19
SMO-Ibk-NB-MLP	97,54	97,54	98,24	98,07	98,24	98,07	98,24	98,07	97,72
SMO-Ibk-NB-J48	97,01	97,01	97,01	97,36	97,01	97,36	97,01	97,36	97,01
SMO-J48-MLP-NB	96,84	96,84	97,54	97,54	97,54	97,54	97,54	97,72	97,54
SMO-J48-MLP-Ibk	96,84	97,01	97,54	97,36	97,54	97,36	97,54	97,89	98,07
J48-MLP-NB-Ibk	96,84	96,84	96,31	96,66	96,31	96,66	96,84	97,36	95,96

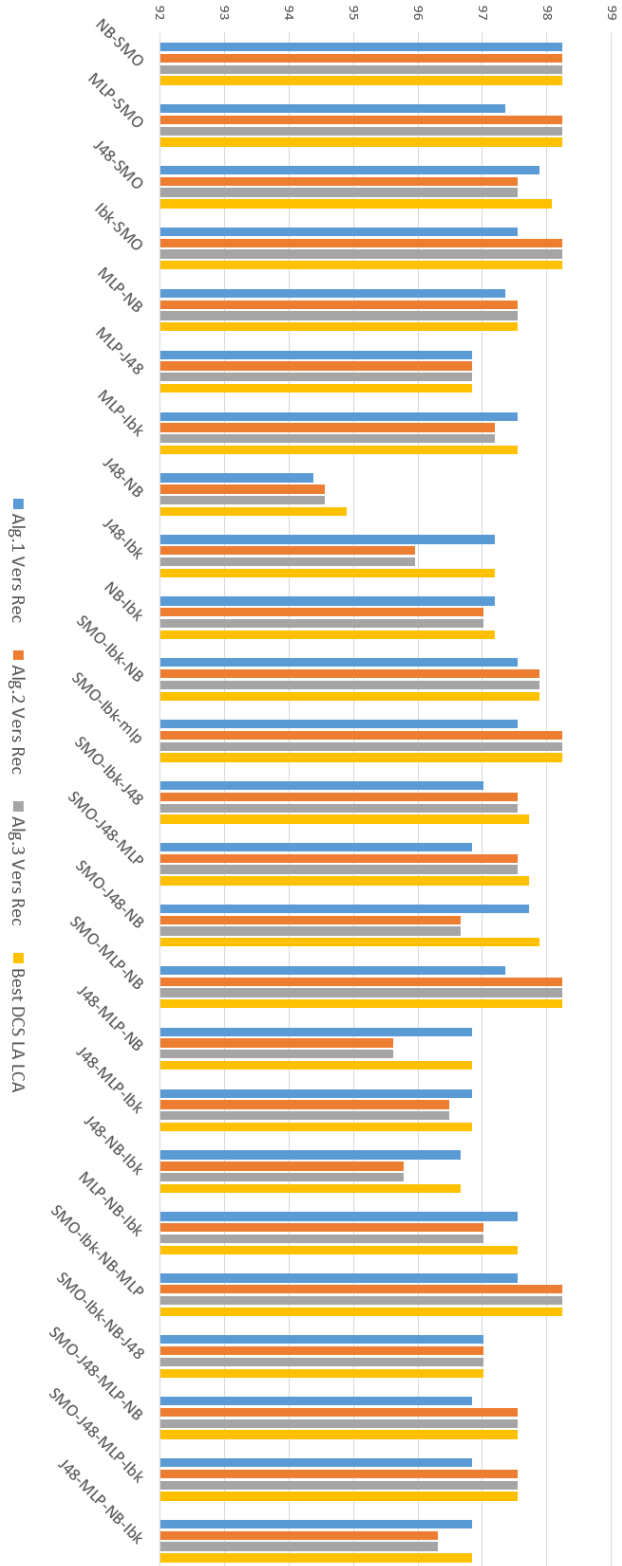


Figura 6.1: WDBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA

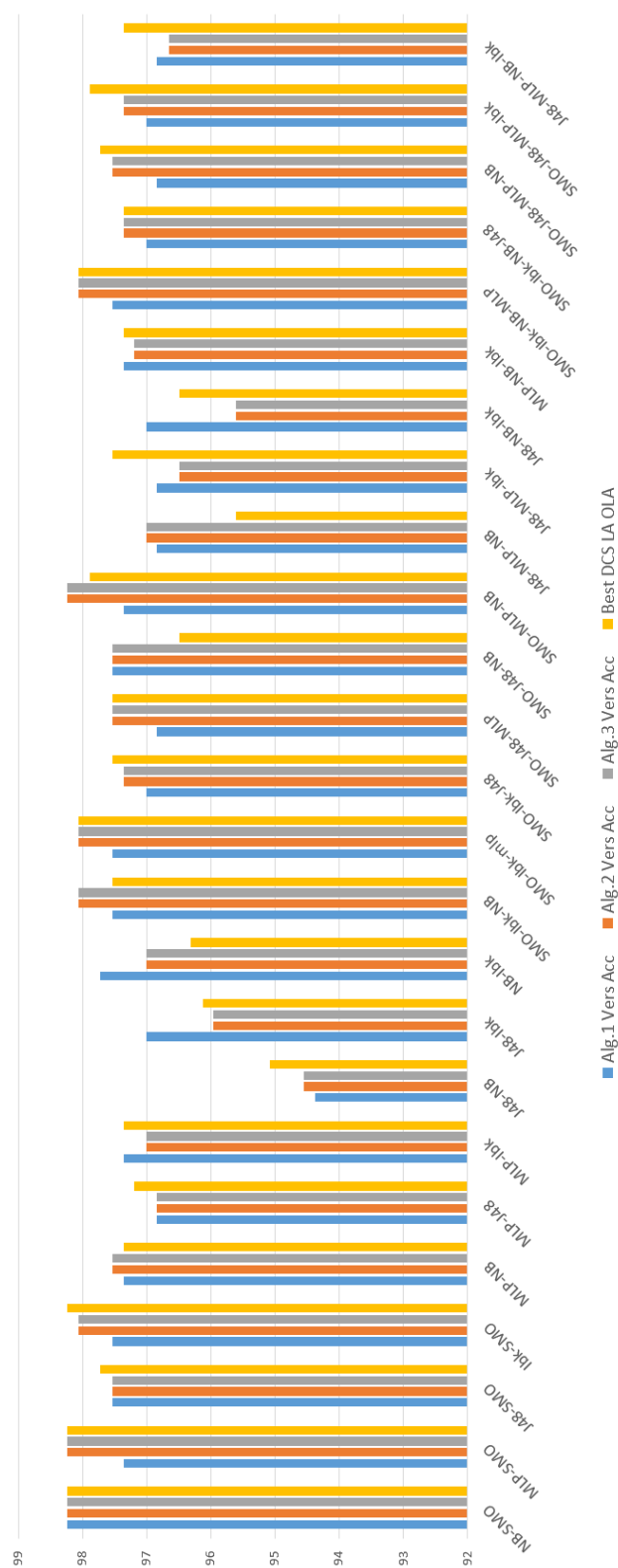


Figura 6.2: WDBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA

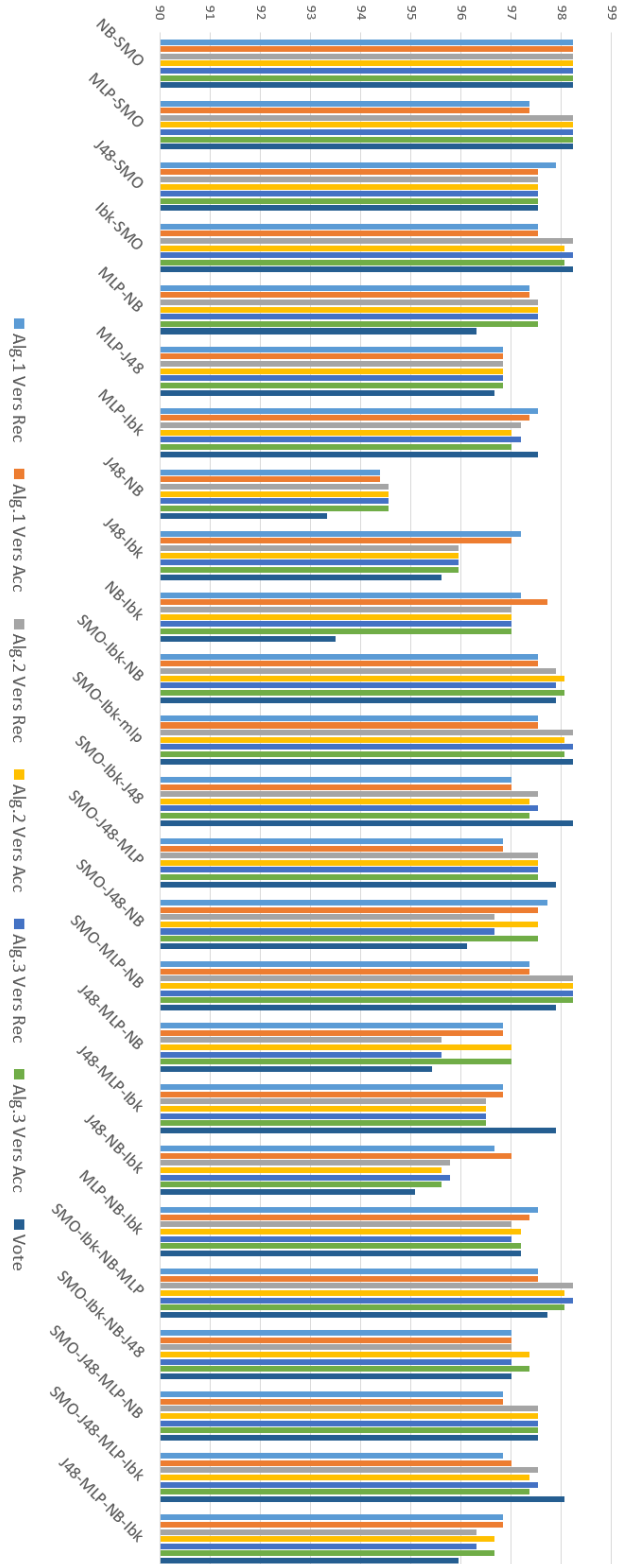


Figura 6.3: WDBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote

### 6.4.2 Risultati su Wisconsin Breast Cancer Dataset

La Tabella 6.14 riporta i risultati in termini di accuratezza di classificazione ottenuti sul dataset WBC.

Si confrontano i risultati degli Algoritmi 1, 2 e 3, nella versione che usa la recall come criterio di competenza, con quelli dell'algoritmo DCS LA-LCA. Per quanto riguarda l'Algoritmo 1 la tabella mostra che in 6 insiemi su 25 si hanno prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA usando gli stessi insiemi di classificatori di base. In particolare, in 3 insiemi su 6 l'Algoritmo 1 ha prestazioni migliori. Il valore di accuratezza ottenuto utilizzando come set di classificatori di base MLP-NB è il migliore ottenuto sull'intero dataset tra tutti i metodi che usano lo stesso set di classificatori. Per quanto riguarda l'Algoritmo 2, in 9 insiemi su 25 si registrano prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA e in particolare, in 3 insiemi su 9 l'Algoritmo 2 ha prestazioni migliori. Utilizzando come set di classificatori di base quello costituito da MLP-SMO o quello costituito da MLP-NB si registrano i valori di accuratezza più alti ottenuti su l'intero dataset tra tutti gli algoritmi che usano questo insieme di classificatori di base. Per quanto riguarda l'Algoritmo 3 la tabella mostra che in 9 insiemi su 25 si hanno prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA e in particolare, in 3 insiemi su 9 si registrano prestazioni migliori. Con l'insieme MLP-SMO si registrano i valori di accuratezza migliori ottenuti su l'intero dataset tra tutti gli algoritmi che usano lo stesso set di classificatori. Il confronto dei metodi è riportato nel grafico di Figura 6.4.

Si confrontano successivamente, i risultati degli Algoritmi 1, 2 e 3 nella versione che usa l'accuratezza come criterio di competenza con i risultati dell'algoritmo DCS LA-OLA. Per quanto riguarda l'Algoritmo 1, in 5 insiemi su 25 si hanno prestazioni migliori a quelle ottenute con il metodo DCS LA-OLA. Il valore di accuratezza ottenuto utilizzando l'insieme di classificatori di base NB-SMO è il migliore ottenuto su l'intero dataset tra tutti gli algoritmi che usano lo stesso set di classificatori. L'Algoritmo 2, in 8 insiemi

su 25 ha prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-OLA. In 4 insiemi su 8 si registrano prestazioni migliori. I valori di accuratezza ottenuti con gli insiemi MLP-SMO e SMO-MLP-NB sono i migliori ottenuti su l'intero dataset tra tutti gli algoritmi che usano gli stessi insiemi di classificatori di base. Per quanto riguarda l'Algoritmo 3 la tabella mostra che in 7 insiemi su 25 si hanno prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-OLA. I valori di accuratezza ottenuti con gli insiemi SMO-MLP e SMO-MLP-NB sono i più alti ottenuti sul dataset WBC tra tutti gli algoritmi che usano lo stesso set di classificatori. Il confronto dei metodi è riportato nel grafico di Figura 6.5.

Infine, si confrontano tutte le versioni dell'approccio proposto con la tecnica del Vote. In 6 insiemi su 25 almeno uno degli algoritmi proposti ottiene prestazioni uguali o migliori. Il confronto dei metodi è riportato nel grafico di Figura 6.6.

In generale, a parità di insiemi di classificatori di base, con gli insiemi costituiti da NB-SMO, MLP-NB, MLP-Ibk e J48-NB gli approcci proposti permettono di ottenere l'accuratezza migliore sul dataset WBC.

Tabella 6.14: Risultati su WBC Dataset

Classificatori	Alg. 1		Alg. 2		Alg. 3		Best DCS		Vote
	Ver Rec	Ver Acc	Ver Rec	Ver Acc	Ver Rec	Ver Acc	LA LCA	LA OLA	
NB-SMO	97,28	<b>97,57</b>	97,28	97,42	97,28	97,42	97,28	97,28	97,28
MLP-SMO	96,85	96,85	<b>97,28</b>	<b>97,28</b>	<b>97,28</b>	<b>97,28</b>	97,14	<b>97,28</b>	<b>97,28</b>
J48-SMO	95,85	96,71	96,57	96,71	96,57	96,71	<b>97,14</b>	97	96,71
Ibk-SMO	97	96,85	97	96,85	97	96,85	97	<b>97,28</b>	<b>97,28</b>
MLP-NB	97	97,28	<b>97,42</b>	<b>97,42</b>	<b>97,42</b>	<b>97,42</b>	96,85	96,85	97,28
MLP-J48	95,42	96,28	95,85	96,28	95,85	96,28	<b>96,85</b>	96,57	96,14
MLP-Ibk	<b>97,14</b>	97	96,85	96,85	96,85	96,85	96,85	96,85	96,71
J48-NB	96,71	<b>97</b>	96,85	<b>97</b>	96,85	<b>97</b>	96,57	95,71	96,71
J48-Ibk	96,14	96,42	96,14	96,42	96,14	96,42	96,42	95,57	<b>96,57</b>
NB-Ibk	97	97	97,42	97,28	97,42	97,28	97,42	<b>97,57</b>	<b>97,57</b>
SMO-Ibk-NB	96,85	97,14	97	97,28	97	97,28	97,28	<b>97,71</b>	97,28
SMO-Ibk-MLP	97	97	97	96,85	97	96,85	97	<b>97,28</b>	<b>97,28</b>
SMO-Ibk-J48	96,14	96,57	96,42	96,42	96,42	96,42	97,28	<b>97,42</b>	97,28
SMO-J48-MLP	95,42	96,14	96,42	96,71	96,42	96,71	97	97,14	<b>97,28</b>
SMO-J48-NB	96,57	96,71	97	96,85	97	96,85	<b>97,28</b>	<b>97,28</b>	<b>97,28</b>
SMO-MLP-NB	97	97,28	97,14	<b>97,42</b>	97,14	<b>97,42</b>	97,28	<b>97,42</b>	<b>97,42</b>
J48-MLP-NB	96,28	96,42	96,71	96,85	96,71	96,85	97,42	97,14	<b>97,57</b>
J48-MLP-Ibk	96,14	96,42	95,85	96,28	95,85	96,28	<b>97,42</b>	97,28	97,28
J48-NB-Ibk	96,42	96,57	96,57	96,71	96,57	96,71	97,28	97,28	<b>97,42</b>
MLP-NB-Ibk	97	97	<b>97,28</b>	<b>97,28</b>	<b>97,28</b>	<b>97,28</b>	<b>97,28</b>	<b>97,28</b>	<b>97,28</b>
SMO-Ibk-NB-MLP	96,85	97,14	97	97,28	97	97,28	97,14	<b>97,57</b>	97,42
SMO-Ibk-NB-J48	96,42	96,71	96,57	96,71	96,71	96,71	97,28	<b>97,42</b>	<b>97,42</b>
SMO-J48-MLP-NB	96,28	96,57	96,71	96,85	96,85	96,85	97	97,14	<b>97,28</b>
SMO-J48-MLP-Ibk	96,14	96,71	96,14	96,42	96,28	96,42	97	97,14	<b>97,28</b>
J48-MLP-NB-Ibk	96,42	96,57	96,57	96,71	96,57	96,71	97,42	97,14	<b>97,57</b>



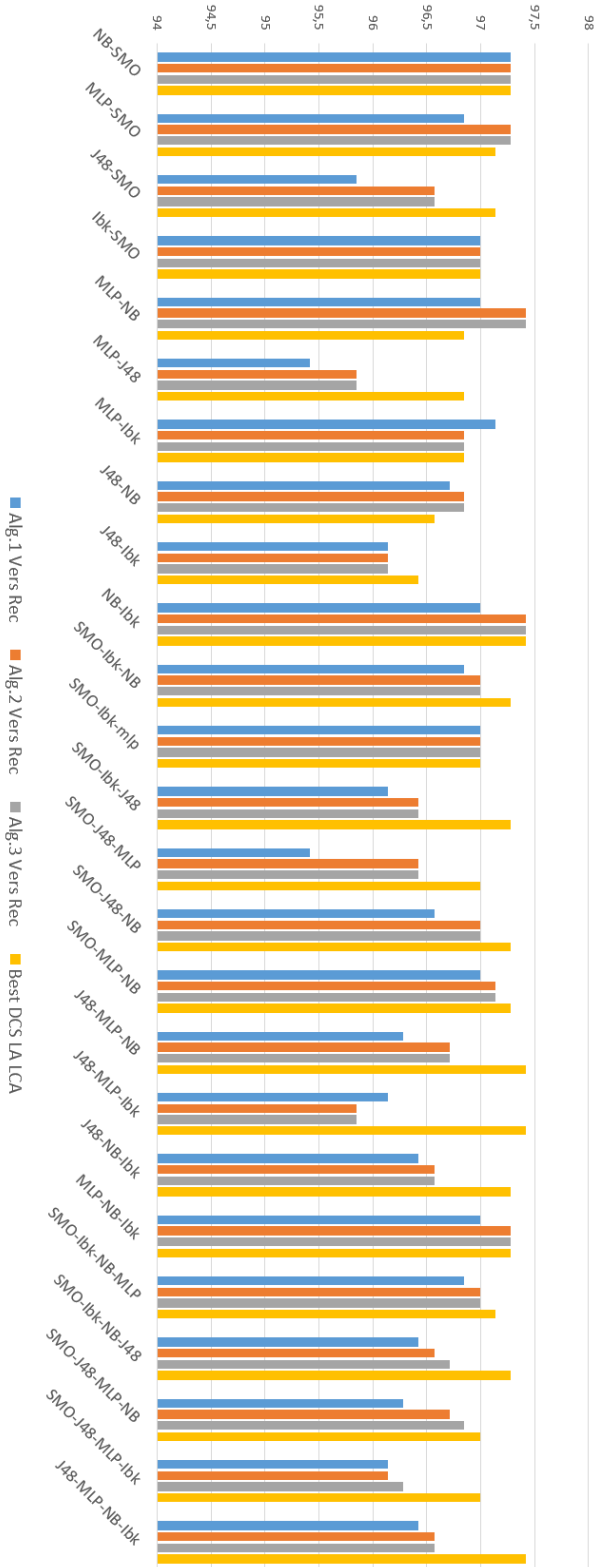


Figura 6.4: WBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA



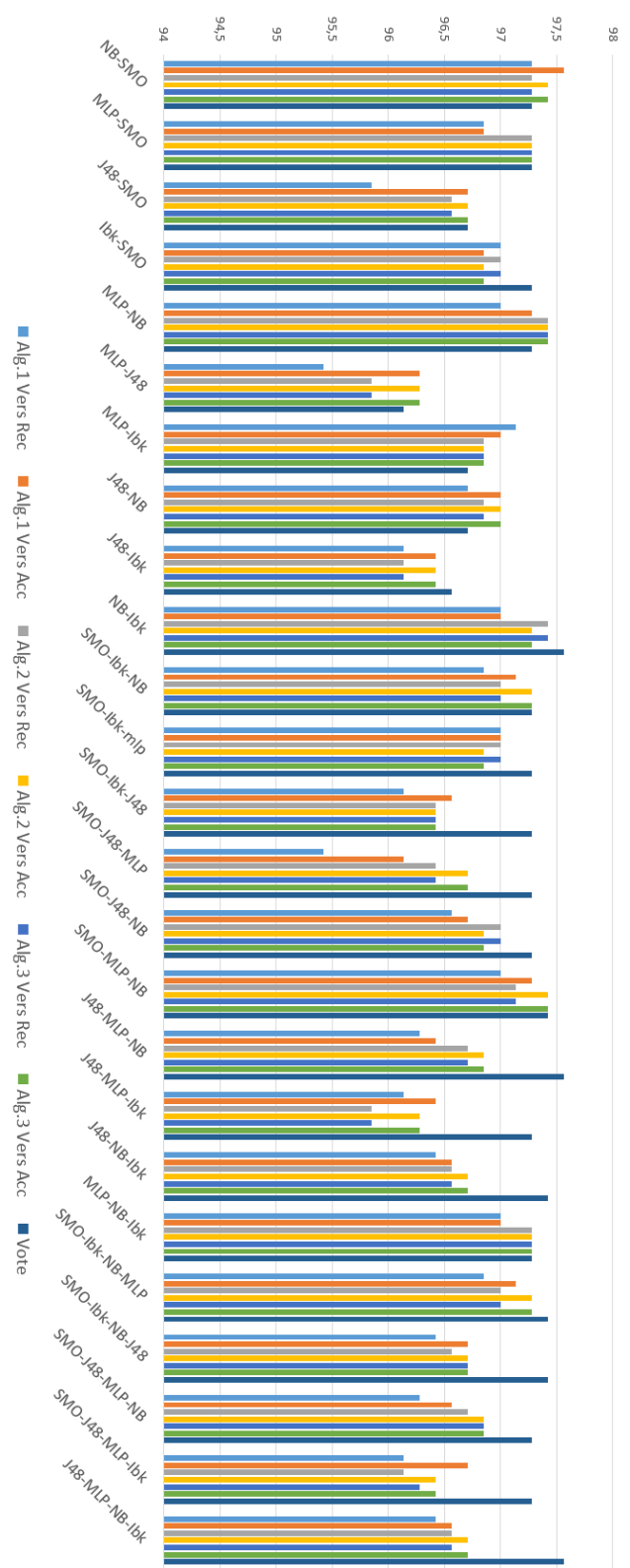


Figura 6.6: WBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote

### 6.4.3 Risultati su Wisconsin Prognostic Breast Cancer Dataset

La Tabella 6.15 riporta i risultati in termini di accuratezza ottenuti nella fase di sperimentazione sul WPBC Dataset.

Dal confronto dei risultati degli Algoritmi 1, 2 e 3 nella versione che usa la recall come criterio di competenza con quelli dell'algoritmo DCS LA-LCA, si può osservare che l'Algoritmo 1 in 21 insiemi su 25 ha prestazioni migliori rispetto a tale metodo. L'Algoritmo 2 ottiene in 22 insiemi su 25 prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA. In 20 insiemi su 22 ha prestazioni migliori. Anche l'Algoritmo 3 ottiene in 22 insiemi su 25 prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA. Anch'esso in 20 insiemi su 22 ha prestazioni migliori. Con gli insiemi di classificatori di base Ibk-SMO, MLP-Ibk e MLP-NB-Ibk, l'Algoritmo 1 ha le prestazioni più alte sul WPBC Dataset rispetto agli altri metodi con gli stessi classificatori di base.

Dal confronto dei risultati degli Algoritmi 1, 2 e 3, nella versione che usa l'accuratezza come criterio di competenza, con quelli dell'algoritmo DCS LA-OLA si può osservare che l'Algoritmo 1 in 14 insiemi su 25 ha prestazioni migliori o uguali a quelle ottenute con tale metodo. In particolare, in 8 insiemi su 14 si registrano prestazioni migliori. L'Algoritmo 2 ha prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-OLA in 19 insiemi su 25. In 7 insiemi su 19 ha prestazioni migliori. L'Algoritmo 3 ha in 10 insiemi su 25 prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-OLA e in particolare, in 5 insiemi su 10 ha prestazioni migliori.

Dal confronto con la tecnica del Vote, in 14 insiemi su 25 almeno uno degli algoritmi proposti ottiene prestazioni migliori.

I grafici in Figura 6.7 , Figura 6.8 e Figura 6.9 riportano tali confronti.

Infine, a parità di classificatori di base, in 11 insiemi (Ibk-SMO, MLP-Ibk, SMO-Ibk-MLP, SMO-J48-MLP, SMO-MLP-NB, J48-MLP-Ibk, J48-NB-Ibk, MLP-NB-Ibk, SMO-Ibk-NB-MLP, SMO-J48-MLP-NB e SMO-J48-MLP-Ibk) gli approcci proposti ottengono l'accuratezza migliore sul WPBC Dataset.

Tabella 6.15: Risultati su WPBC Dataset

Classificatori	Alg. 1		Alg. 2		Alg. 3		Best DCS LA LCA	Best DCS LA OLA	Vote
	Ver Rec	Ver Acc	Ver Rec	Ver Acc	Ver Rec	Ver Acc			
NB-SMO	78,79	79,8	80,3	79,8	79,8	79,8	78,28	80,3	81,31
MLP-SMO	78,79	79,8	80,3	79,8	79,8	79,8	78,28	80,3	81,31
J48-SMO	80,3	<b>81,31</b>	<b>81,31</b>	<b>81,31</b>	<b>81,31</b>	<b>81,31</b>	79,8	<b>81,31</b>	<b>81,31</b>
Ibk-SMO	<b>81,82</b>	81,31	81,31	81,31	79,8	79,8	81,31	81,31	81,31
MLP-NB	75,76	75,76	75,76	75,76	<b>76,26</b>	<b>76,26</b>	<b>76,26</b>	<b>76,26</b>	<b>76,26</b>
MLP-J48	75,76	76,26	75,76	75,76	75,76	75,76	76,26	<b>76,77</b>	<b>76,77</b>
MLP-Ibk	<b>77,78</b>	77,27	<b>77,78</b>	76,77	76,77	76,26	76,26	76,77	76,26
J48-NB	75,25	75,76	75,76	75,76	75,76	75,76	75,76	<b>77,27</b>	76,77
J48-Ibk	<b>77,78</b>	<b>77,78</b>	<b>77,78</b>	<b>77,78</b>	<b>77,78</b>	<b>77,78</b>	76,26	<b>77,78</b>	76,26
NB-Ibk	77,27	77,27	<b>77,78</b>	<b>77,78</b>	76,77	76,77	76,26	76,26	<b>77,78</b>
SMO-Ibk-NB	78,28	79,29	80,3	80,3	79,8	79,8	76,77	80,3	<b>81,31</b>
SMO-Ibk-MLP	78,79	79,29	<b>80,81</b>	80,3	79,8	79,29	76,77	80,3	79,29
SMO-Ibk-J48	80,3	80,81	81,31	81,82	80,3	81,31	77,27	<b>81,82</b>	81,31
SMO-J48-MLP	78,79	79,8	<b>80,3</b>	79,8	<b>80,3</b>	79,8	76,26	79,29	77,27
SMO-J48-NB	78,28	79,8	79,8	79,8	80,3	79,8	76,26	79,29	<b>81,31</b>
SMO-MLP-NB	78,28	79,8	79,8	<b>80,3</b>	79,8	79,8	76,26	78,28	76,26
J48-MLP-NB	75,25	75,76	75,25	75,76	75,76	75,76	76,26	<b>76,77</b>	75,76
J48-MLP-Ibk	<b>77,78</b>	77,27	77,78	76,77	76,77	76,26	75,25	76,77	75,76
J48-NB-Ibk	77,27	77,27	77,78	<b>78,28</b>	76,77	76,77	75,25	76,77	76,26
MLP-NB-Ibk	<b>77,27</b>	76,77	<b>77,27</b>	76,77	76,77	76,26	76,26	76,77	76,26
SMO-Ibk-NB-MLP	78,28	79,29	80,3	<b>80,81</b>	79,8	79,29	76,77	79,29	77,78
SMO-Ibk-NB-J48	78,28	79,29	80,3	80,3	80,3	79,8	76,77	80,3	<b>81,82</b>
SMO-J48-MLP-NB	78,28	79,8	79,8	<b>80,3</b>	<b>80,3</b>	79,8	75,25	78,79	77,27
SMO-J48-MLP-Ibk	78,79	79,29	<b>80,81</b>	80,3	80,3	79,29	76,77	80,3	77,27
J48-MLP-NB-Ibk	<b>77,27</b>	<b>77,27</b>	<b>77,27</b>	<b>77,27</b>	76,77	76,26	76,26	<b>77,27</b>	75,76

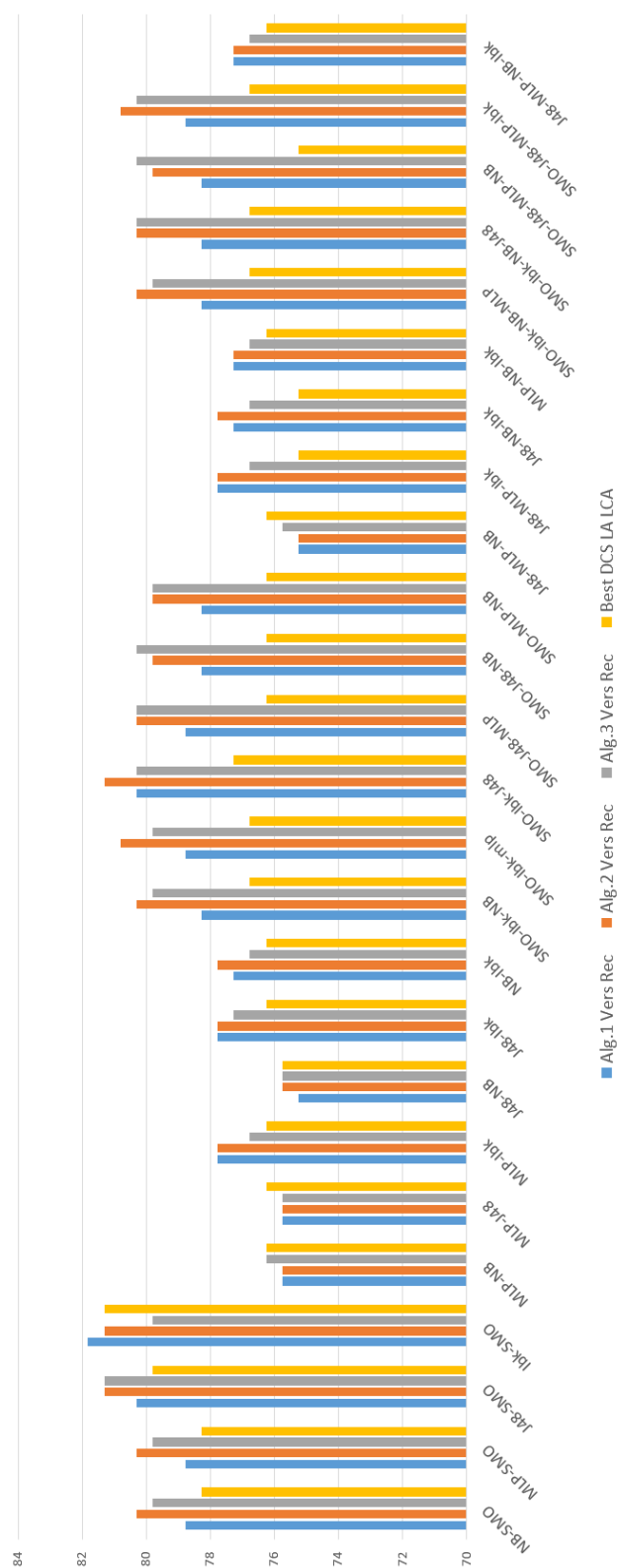


Figura 6.7: WPBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA

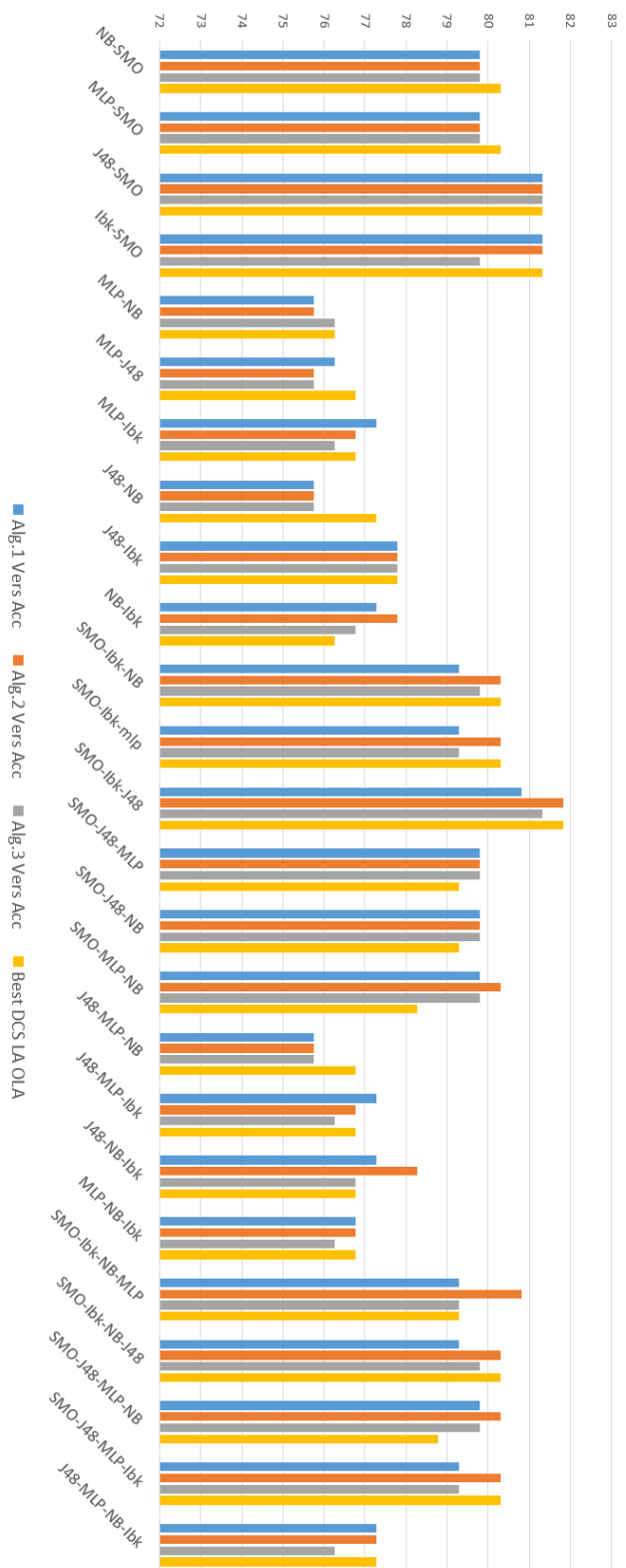


Figura 6.8: WPBC Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA





#### 6.4.4 Risultati su Cleveland Dataset

La Tabella 6.16 riporta i risultati in termini di accuratezza ottenuti nella fase di sperimentazione sul Cleveland Dataset.

Dal confronto dei risultati degli Algoritmi 1, 2 e 3, nella versione che usa la recall come criterio di competenza, con quelli dell'algoritmo DCS LA-LCA, si può osservare che l'Algoritmo 1 in 6 insiemi su 25 ha prestazioni migliori o uguali rispetto a tale metodo. In particolare, in 5 insiemi su 6 si registrano prestazioni migliori. L'Algoritmo 2 ottiene in 6 insiemi su 25 prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA. Solo con l'insieme di classificatori di base costituito da J48-MLP-NB ottiene prestazioni migliori. L'Algoritmo 3 ottiene in 6 insiemi su 25 prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA e in particolare, in 4 insiemi ha prestazioni migliori.

Dal confronto dei risultati gli Algoritmi 1, 2 e 3 nella versione che usa l'accuratezza come criterio di competenza con quelli dell'algoritmo DCS LA-OLA si può osservare che tutti gli algoritmi hanno prestazioni leggermente inferiori a tale tecnica. Solo con un insieme di classificatori di base si ottengono valori di accuratezza uguali a quelli ottenuti con il metodo DCS LCA OLA.

Dal confronto con la tecnica del Vote, in 16 insiemi su 25 almeno uno degli algoritmi proposti ottiene prestazioni migliori.

I grafici in Figura 6.10, Figura 6.11 e Figura 6.12 riportano tali confronti.

A parità di classificatori di base, i valori di accuratezza ottenuti dagli approcci proposti utilizzando gli insiemi J48-NB e MLP-NB sono i migliori ottenuti sul Cleveland Dataset.

Tabella 6.16: Risultati su Cleveland Dataset

Classificatori	Alg. 1		Alg. 2		Alg. 3		Best DCS		Vote
	Ver Rec	Ver Acc	Ver Rec	Ver Acc	Ver Rec	Ver Acc	LA LCA	LA OLA	
NB-SMO	83,17	83,5	83,17	83,5	83,17	83,5	84,16	<b>84,82</b>	83,83
MLP-SMO	82,51	83,17	82,51	83,5	83,17	83,5	85,15	<b>85,48</b>	83,83
J48-SMO	80,2	81,52	83,17	82,84	<b>83,83</b>	82,84	83,5	<b>83,83</b>	83,17
Ibk-SMO	85,15	84,49	84,82	84,82	84,49	85,15	85,15	<b>85,48</b>	83,83
MLP-NB	81,52	81,52	81,52	82,18	82,18	82,51	<b>83,5</b>	82,84	81,19
MLP-J48	81,19	80,53	81,19	80,53	80,86	80,53	82,18	<b>83,83</b>	81,52
MLP-Ibk	83,83	83,5	83,5	83,83	83,5	83,83	83,5	<b>84,49</b>	80,86
J48-NB	80,2	80,86	81,85	81,85	<b>82,51</b>	81,85	82,18	81,85	81,19
J48-Ibk	82,51	82,18	82,51	81,85	82,51	81,85	<b>83,5</b>	82,18	79,87
NB-Ibk	83,17	83,5	83,83	83,5	84,16	84,49	83,83	<b>85,15</b>	84,16
SMO-Ibk-NB	83,83	83,5	83,17	83,83	82,84	84,16	85,15	<b>85,81</b>	84,49
SMO-Ibk-MLP	85,15	84,16	83,83	84,16	83,5	84,16	83,83	<b>85,48</b>	83,17
SMO-Ibk-J48	83,17	83,17	83,17	83,5	82,84	83,83	<b>84,16</b>	84,16	82,51
SMO-J48-MLP	81,52	82,84	81,52	82,18	81,85	82,51	82,51	<b>83,5</b>	82,84
SMO-J48-NB	79,54	80,2	81,52	82,51	82,18	82,51	83,5	<b>84,16</b>	83,17
SMO-MLP-NB	81,52	82,18	80,86	82,51	81,52	82,51	<b>84,16</b>	84,16	83,83
J48-MLP-NB	81,52	81,85	<b>83,17</b>	81,52	<b>83,17</b>	81,52	82,18	82,84	80,86
J48-MLP-Ibk	83,17	82,51	82,51	82,51	82,51	82,51	82,51	<b>84,16</b>	80,53
J48-NB-Ibk	81,85	82,51	83,5	82,51	83,83	83,5	<b>84,16</b>	83,83	81,85
MLP-NB-Ibk	83,83	82,84	83,5	83,17	83,17	83,83	84,82	<b>85,15</b>	82,18
SMO-Ibk-NB-MLP	84,49	83,17	81,52	83,17	80,86	83,17	83,83	<b>85,81</b>	83,83
SMO-Ibk-NB-J48	81,85	82,84	82,18	83,17	82,18	83,5	83,5	<b>84,82</b>	84,16
SMO-J48-MLP-NB	80,53	81,85	80,86	82,18	81,52	82,18	83,17	<b>84,82</b>	83,5
SMO-J48-MLP-Ibk	83,83	83,17	83,17	83,17	82,84	83,17	83,17	<b>85,15</b>	82,51
J48-MLP-NB-Ibk	83,17	82,51	83,17	82,18	82,84	82,51	83,83	<b>85,15</b>	81,85

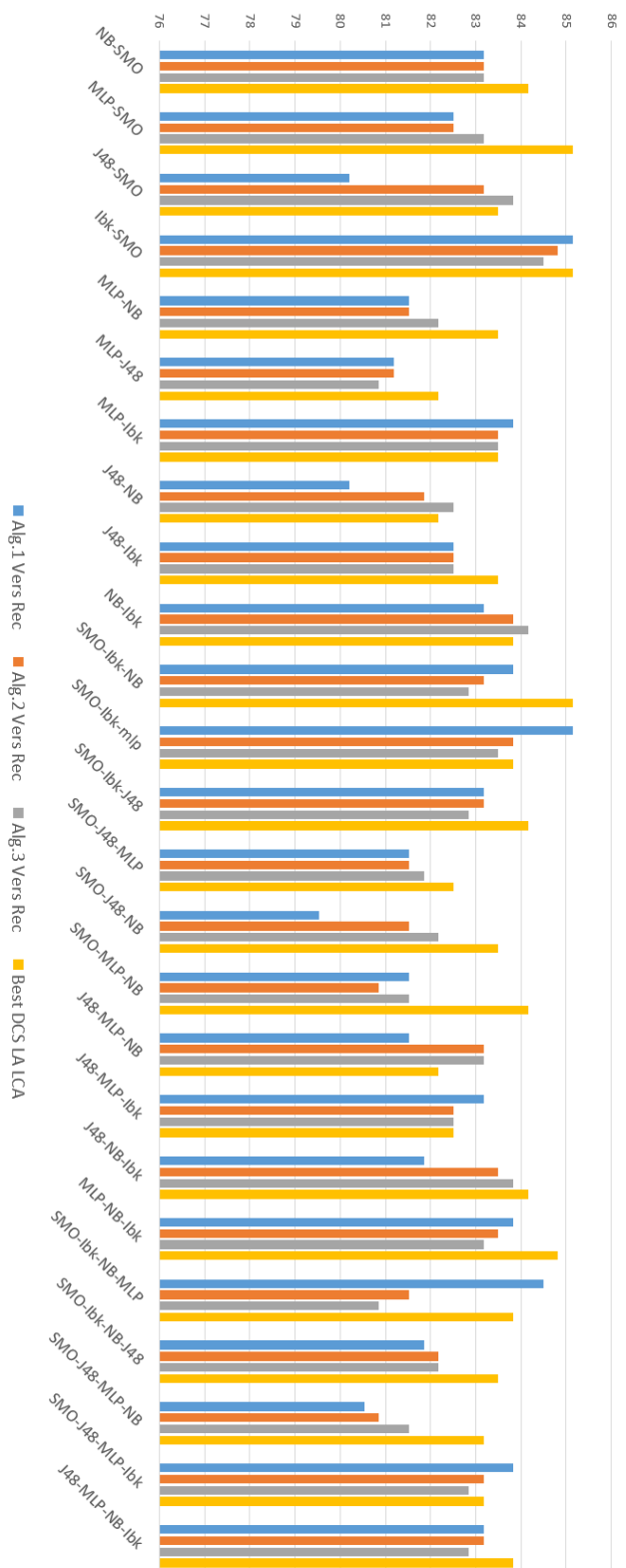


Figura 6.10: Cleveland Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA

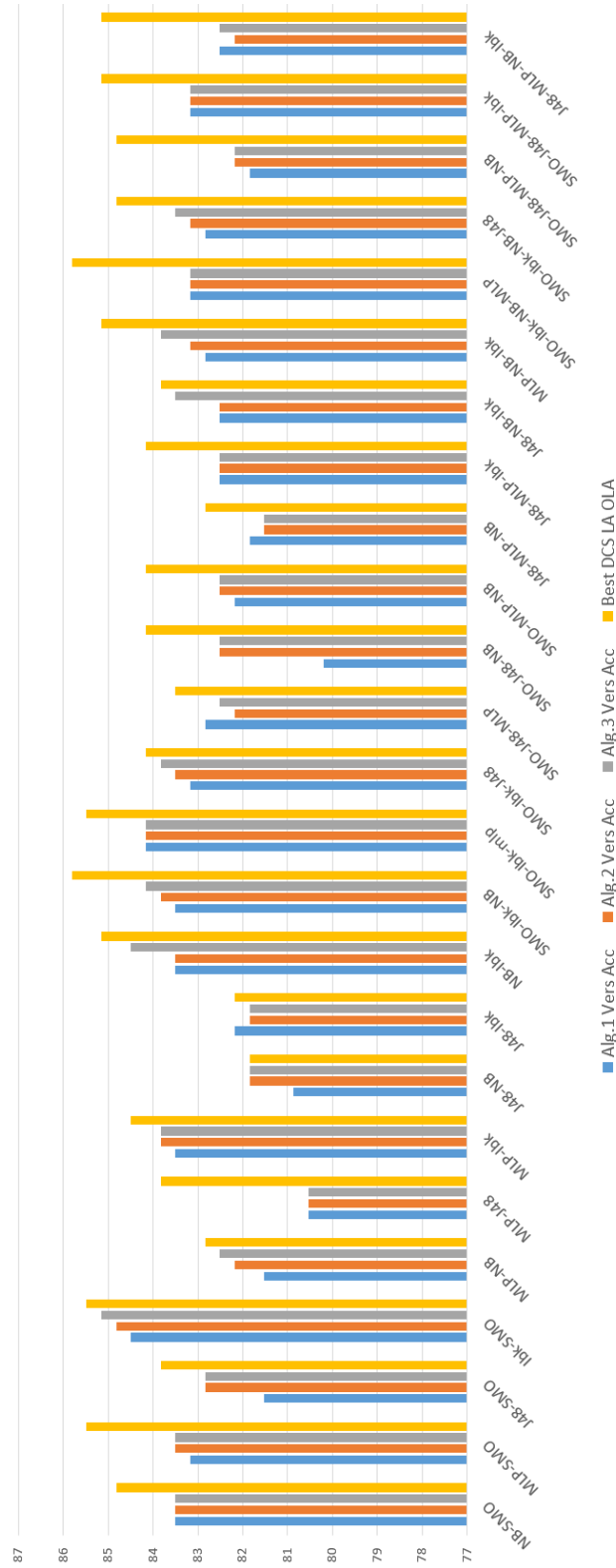


Figura 6.11: Cleveland Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS LA OLA

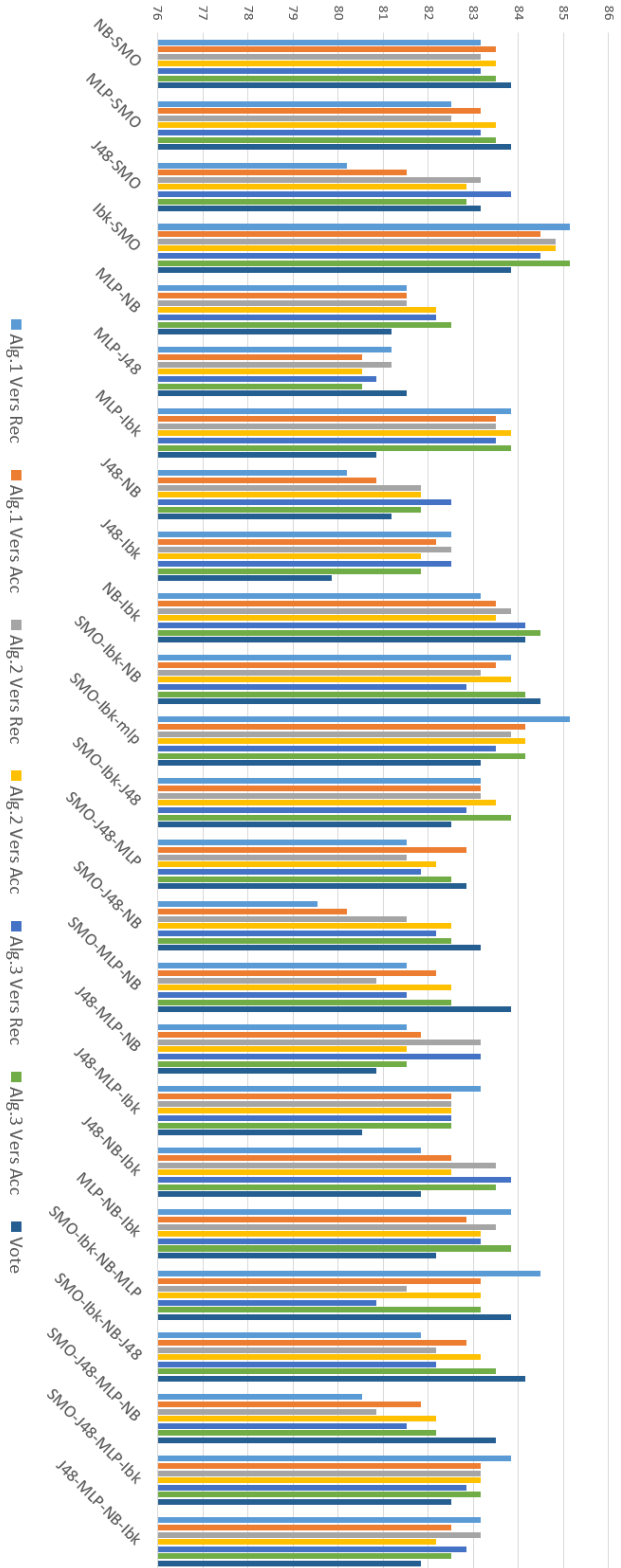


Figura 6.12: Cleveland Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote

### 6.4.5 Risultati su Mammographic Mass Dataset

La Tabella 6.17 riporta i risultati in termini di accuratezza ottenuti nella fase di sperimentazione sul Mammographic Mass Dataset.

Dal confronto dei risultati degli Algoritmi 1, 2 e 3, nella versione che usa la recall come criterio di competenza, con quelli dell'algoritmo DCS LA-LCA, si può osservare che l'Algoritmo 1 in 3 insiemi su 25 ha prestazioni migliori o uguali. In particolare, in 2 insiemi su 3 si registrano prestazioni migliori. L'Algoritmo 2 ottiene solo in 4 insiemi prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA e in uno solo dei tre, ha prestazioni migliori. Le prestazioni dell'Algoritmo 3 sono su questo dataset sempre inferiori a quelle ottenute con il metodo DCS LA-LCA.

Dal confronto dei risultati degli Algoritmi 1, 2 e 3, nella versione che usa l'accuratezza come criterio di competenza, con quelli dell'algoritmo DCS LA-OLA si può osservare che gli Algoritmi 1 e 2 riescono ad ottenere in un solo caso prestazioni migliori mentre l'Algoritmo 3 ha sempre valori di accuratezza inferiori.

Dal confronto con la tecnica del Vote, in 14 insiemi su 25 almeno uno degli approcci proposti ottiene prestazioni migliori.

I grafici in Figura 6.13, Figura 6.14 e Figura 6.15 riportano tali confronti.

A parità di classificatori di base, i valori di accuratezza ottenuti con gli insiemi di classificatori di base SMO-Ibk-NB, J48-MLP-Ibk e SMO-J48-MLP-Ibk sono i migliori ottenuti sul Mammographic Mass Dataset.

### 6.4.6 Risultati su Dermatology Dataset

La Tabella 6.18 riporta i risultati in termini di accuratezza ottenuti nella fase di sperimentazione sul Dermatology Dataset.

Dal confronto dei risultati degli Algoritmi 1, 2 e 3, nella versione che usa la recall come criterio di competenza, con quelli dell'algoritmo DCS LA-LCA, si può osservare che l'Algoritmo 1 in 15 insiemi su 25 ha prestazioni migliori o uguali rispetto a tale tecnica. In particolare, in 7 insiemi su 15 si registrano

prestazioni migliori. I valori di accuratezza ottenuti utilizzando i classificatori NB-SMO, J48-Ibk e SMO-Ibk-NB sono i migliori ottenuti sull'intero dataset tra tutti gli algoritmi che usano questi insiemi di classificatori di base. L'Algoritmo 2 ottiene in 12 insiemi su 25 prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA e in particolare, in 9 insiemi su 12 ha prestazioni migliori. I valori di accuratezza ottenuti con questo algoritmo utilizzando gli insiemi di classificatori di base MLP-Ibk, NB-Ibk, SMO-Ibk-MLP, SMO-MLP-NB e SMO-Ibk-NB-MLP sono i migliori sul Dermatology Dataset. L'Algoritmo 3 ha in 12 insiemi su 25 prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-LCA. In particolare, in 9 insiemi su 12 si registrano prestazioni migliori. I valori di accuratezza ottenuti utilizzando gli insiemi di classificatori di base MLP-Ibk, NB-Ibk, SMO-Ibk-MLP, SMO-MLP-NB, MLP-NB-Ibk e SMO-Ibk-NB-MLP sono i migliori ottenuti per questi insiemi di classificatori. Il grafico di Figura 6.16 mostra tali confronti.

Dal confronto dei risultati degli Algoritmi 1, 2 e 3, nella versione che usa l'accuratezza come criterio di competenza, con quelli dell'algoritmo DCS LA-OLA si può osservare che l'Algoritmo 1 in 9 insiemi su 25 ha prestazioni migliori o uguali a quelle ottenute con tale metodo. In particolare, in 2 insiemi su 9 si registrano prestazioni migliori. L'Algoritmo 2 in 10 insiemi su 25 ha prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-OLA. In particolare, in 8 insiemi su 10 si registrano prestazioni migliori. I valori di accuratezza ottenuti con questo algoritmo utilizzando gli insiemi MLP-Ibk, NB-Ibk, SMO-Ibk-MLP, SMO-MLP-NB, MLP-NB-Ibk e SMO-Ibk-NB-MLP sono i migliori ottenuti sull'intero dataset per questi insiemi di classificatori. L'Algoritmo 3 ha in 10 insiemi su 25 prestazioni migliori o uguali a quelle ottenute con il metodo DCS LA-OLA. In particolare, in 8 insiemi su 10 ha prestazioni migliori. I valori di accuratezza ottenuti con questo algoritmo utilizzando i classificatori MLP-NB, MLP-Ibk, NB-Ibk, SMO-Ibk-MLP, SMO-MLP-NB, MLP-NB-Ibk e SMO-Ibk-NB-MLP sono i migliori ottenuti sul Dermatology Dataset per questi insiemi di classificatori. Il grafico di

Figura 6.17 mostra tali confronti.

Dal confronto con la tecnica del Vote, in 17 insiemi su 25 almeno uno degli algoritmi proposti ottiene prestazioni migliori. Il grafico di Figura 6.18 mostra tali confronti.

In generale, a parità di classificatori di base, con gli insiemi NB-SMO, MLP-NB, MLP-Ibk, J48-NB, J48-Ibk, NB-Ibk, SMO-Ibk-NB, SMO-Ibk-MLP, SMO-MLP-NB, MLP-NB-Ibk e SMO-Ibk-NB-MLP gli approcci proposti ottengono l'accuratezza più alta sul Dermatology Dataset.

### 6.4.7 Considerazioni finali

Sul dataset WDBC le prestazioni dei tre algoritmi proposti, in entrambe le versioni, sono da ritenersi soddisfacenti. L'accuratezza di classificazione ottenuta è in generale molto alta. Ciò è dovuto a diversi fattori. I classificatori di base sono abbastanza diversi tra loro; nella maggior parte dei casi, sono poche le istanze mal classificate da tutti i classificatori dell'insieme. Il dataset contiene un numero sufficientemente grande di istanze per cui anche il validation set, usato per valutare le prestazioni dei classificatori di base è abbastanza grande. Le istanze del dataset WDBC sono inoltre caratterizzate da attributi ben separati tra le classi. Istanze appartenenti alla stessa classe hanno attributi simili e nello spazio degli attributi sono vicine tra loro. La distanza euclidea riesce pertanto a cogliere le differenze tra le due classi e la regione locale, nella maggior parte dei casi, risulta essere interamente costituita da istanze che appartengono alla stessa classe dell'istanza test.

Anche sul WBC Dataset le prestazioni degli approcci proposti sono buone e confrontabili con quelli di DCS LA LCA, DCS LA OLA e Vote. Il dataset rispetto al WDBC è costituito da un numero maggiore di istanze. In questo dataset però istanze appartenenti a classi diverse hanno in molti casi attributi simili. Quello che accade è che nella regione locale spesso vengono incluse istanze che non appartengono alla classe a cui appartiene invece l'istanza test. Ciò porta gli algoritmi proposti a scartare alcuni classificatori di base che invece andrebbero considerati per prendere la decisione finale.



Per quanto riguarda il WPBC Dataset le prestazioni ottenute dagli approcci proposti sono da ritenersi soddisfacenti anche se leggermente inferiori, in termini di accuratezza, rispetto a quelle ottenute su WDBC e WBC Dataset. Il dataset WPBC è costituito, infatti, da poche istanze e le classi sono molto sbilanciate. Nonostante ciò i classificatori di base sono abbastanza diversi tra loro e commettono errori su istanze differenti. Per tale motivo la gestione dei conflitti viene risolta al meglio dai tre algoritmi proposti. In questo caso sarebbe stato utile avere un validation set costituito da più istanze in modo da poter valutare meglio le prestazioni dei classificatori di base.

Le prestazioni ottenute sul Cleveland Dataset dai tre algoritmi proposti se confrontate con i risultati ottenuti dai metodi DCS LA LCA, DCS LA OLA e Vote sono per molti insiemi di classificatori di base leggermente inferiori. In questo caso, istanze appartenenti a classi diverse hanno valori molto simili degli attributi che le caratterizzano. La regione locale spesso contiene istanze che non appartengono alla classe a cui appartiene l'istanza test. Ciò porta gli algoritmi proposti a scartare alcuni classificatori di base che invece andrebbero considerati per prendere la decisione finale. I risultati ottenuti da tutti e tre gli algoritmi proposti sono comunque incoraggianti. Tutti gli approcci infatti riescono a gestire e a risolvere bene i conflitti facendo registrare prestazioni molto simili alle altre tecniche considerate.

Anche sul Mammographic Mass Dataset le prestazioni ottenute dai tre algoritmi proposti se confrontate con i risultati ottenuti dai metodi DCS LA LCA, DCS LA OLA e Vote sono per molti insiemi di classificatori di base leggermente inferiori. Anche in questo caso, istanze appartenenti a classi diverse hanno valori molto simili degli attributi che le caratterizzano. La regione locale, per ciascuna istanza test, contiene istanze appartenenti a classi diverse. Ciò porta gli algoritmi proposti a scartare alcuni classificatori di base che invece andrebbero considerati per prendere la decisione finale. Non solo, in questo caso i classificatori di base non sono sufficientemente diversi tra loro (molti commettono errori sulle stesse istanze) per cui, i conflitti non vengono risolti nel migliore dei modi.

---

Il Dermatology Dataset è l'unico dataset multi classe considerato in questa sessione di sperimentazione. I risultati sono molto incoraggianti e ci suggeriscono di testare il funzionamento degli algoritmi su altri dataset di questo tipo. Sebbene molte istanze che appartengono a classi diverse hanno attributi molto simili, la distanza euclidea riesce a cogliere bene le differenze tra le sei classi. In generale, il numero di istanze nel validation set è sufficiente per valutare le prestazioni dei classificatori di base per cui i conflitti vengono gestiti al meglio dai tre approcci proposti.

Tabella 6.17: Risultati su Mammographic Mass Dataset

Classificatori	Alg. 1		Alg. 2		Alg. 3		Best DCS LA LCA	Best DCS LA OLA	Vote
	Ver Rec	Ver Acc	Ver Rec	Ver Acc	Ver Rec	Ver Acc			
NB-SMO	82,1	81,58	82	81,37	81,69	81,89	82,21	<b>82,31</b>	81,79
MLP-SMO	81,37	80,85	81,37	80,85	81,27	80,96	<b>82,1</b>	81,79	81,79
J48-SMO	82,31	81,27	82,1	81,17	81,79	81,17	<b>82,41</b>	82,1	81,58
Ibk-SMO	81,48	81,37	81,48	81,37	81,58	81,48	<b>82,1</b>	81,58	81,79
MLP-NB	81,79	81,69	82	81,69	81,79	81,69	82,1	82,31	<b>82,41</b>
MLP-J48	82,52	81,58	82,52	81,58	81,79	81,58	<b>82,62</b>	82,52	82,41
MLP-Ibk	81,17	80,85	81,17	80,85	80,96	81,17	81,58	81,48	<b>81,69</b>
J48-NB	82,52	81,27	82,52	81,27	82,41	80,96	82,62	82,52	<b>82,93</b>
J48-Ibk	81,58	80,85	81,58	80,85	81,06	80,54	82,21	<b>82,31</b>	81,06
NB-Ibk	82,41	81,58	82,73	81,58	82,62	81,58	<b>83,04</b>	82,41	81,48
SMO-Ibk-NB	81,79	<b>82,21</b>	81,89	81,89	82	81,89	82,1	82,1	81,89
SMO-Ibk-MLP	81,27	80,65	81,17	80,75	81,17	80,75	<b>81,79</b>	81,69	81,58
SMO-Ibk-J48	81,89	81,17	81,89	81,27	81,69	80,75	<b>82,31</b>	81,58	81,69
SMO-J48-MLP	82,52	80,85	82,52	80,85	81,37	80,65	82,52	<b>82,73</b>	82
SMO-J48-NB	82,21	80,85	82,1	80,96	81,89	80,75	82,83	83,04	<b>83,14</b>
SMO-MLP-NB	81,79	81,48	81,79	81,58	81,27	81,48	82,31	<b>82,73</b>	82,1
J48-MLP-NB	82,31	81,06	82,31	81,27	82	80,85	<b>83,04</b>	<b>83,04</b>	82,62
J48-MLP-Ibk	<b>82,41</b>	80,96	82,41	81,37	81,37	81,27	82,31	82,31	82,1
J48-NB-Ibk	82,1	80,33	82,1	80,54	81,89	80,33	82,83	82,62	<b>83,14</b>
MLP-NB-Ibk	81,89	81,06	82,1	81,37	81,89	81,37	<b>82,31</b>	82,1	82
SMO-Ibk-NB-MLP	82	81,69	81,89	81,79	81,48	81,48	<b>82,1</b>	81,79	81,89
SMO-Ibk-NB-J48	82,1	80,85	82,21	81,27	82,1	80,75	<b>82,21</b>	81,89	82,1
SMO-J48-MLP-NB	82	80,65	82	80,85	81,48	80,54	82,41	<b>82,93</b>	82,31
SMO-J48-MLP-Ibk	<b>82,31</b>	80,54	82,21	80,85	81,27	80,44	82,21	82,21	81,58
J48-MLP-NB-Ibk	82,31	80,33	82,31	81,06	82	80,65	<b>82,73</b>	82,52	82,21



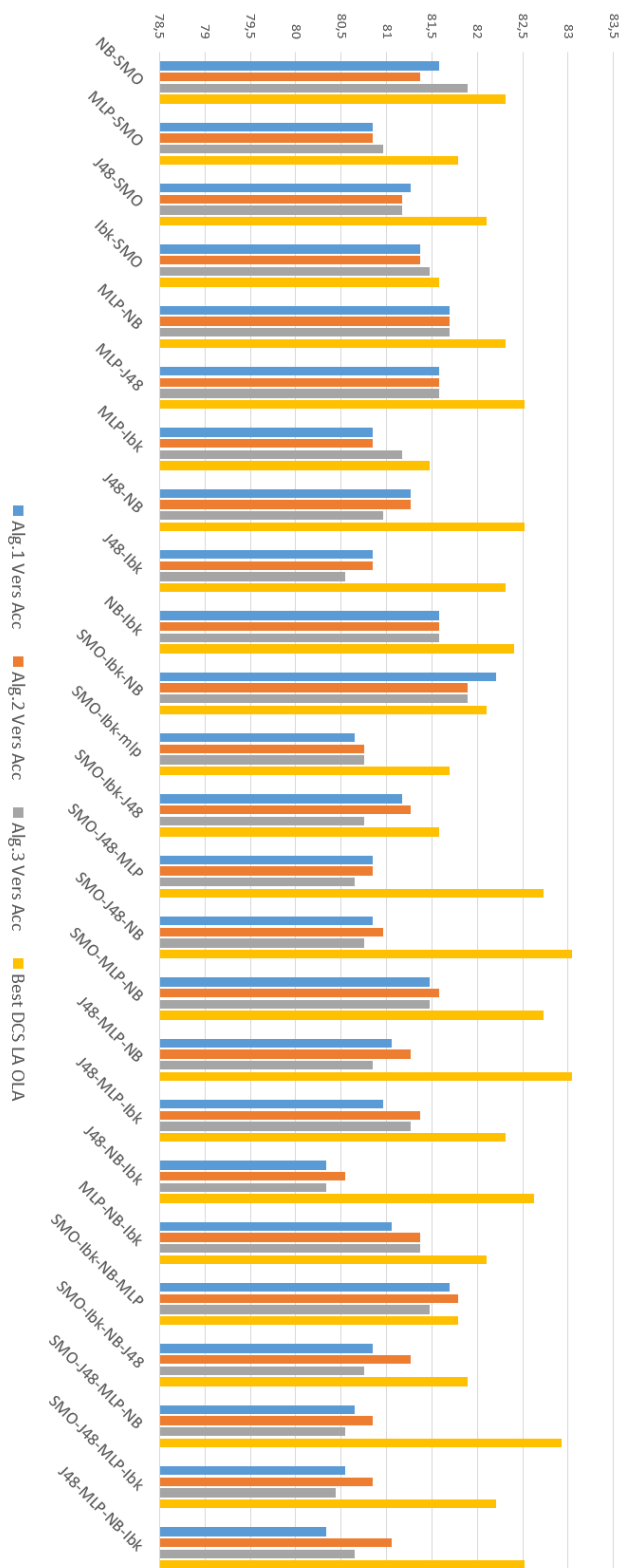


Figura 6.14: Mammographic Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS IA OLA

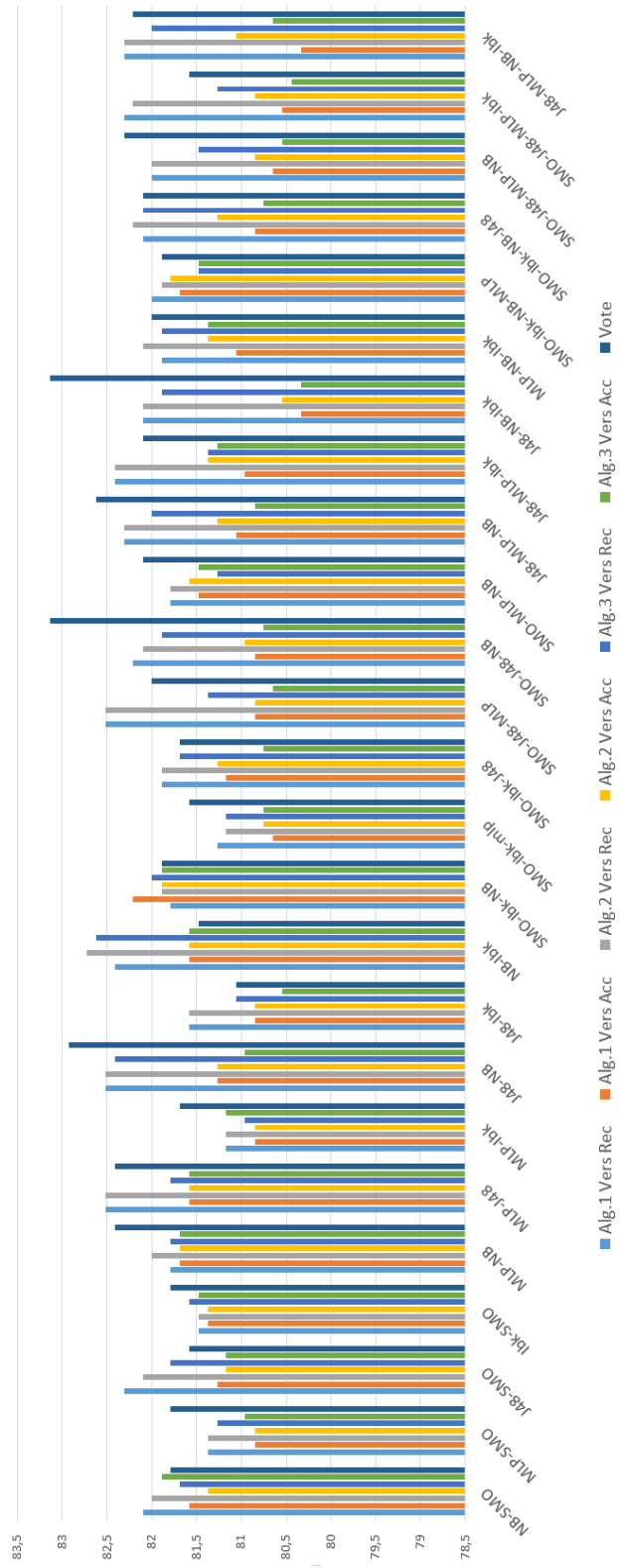


Figura 6.15: Mammographic Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote

Tabella 6.18: Risultati su Dermatology Dataset

Classificatori	Alg. 1		Alg. 2		Alg. 3		Best DCS LA LCA	Best DCS LA OLA	Vote
	Ver Rec	Ver Acc	Ver Rec	Ver Acc	Ver Rec	Ver Acc			
NB-SMO	<b>98,36</b>	98,09	98,09	97,81	98,09	97,81	98,09	98,09	97,54
MLP-SMO	98,36	98,36	98,36	98,36	98,36	98,36	98,36	<b>98,63</b>	98,09
J48-SMO	96,99	97,54	95,9	95,63	95,9	95,63	<b>98,36</b>	97,81	96,45
Ibk-SMO	<b>98,09</b>	<b>98,09</b>	96,99	97,27	96,99	97,27	<b>98,09</b>	<b>98,09</b>	96,99
MLP-NB	98,36	98,36	99,18	99,18	99,18	<b>99,18</b>	98,36	98,36	97,54
MLP-J48	<b>98,36</b>	98,09	97,81	97,54	97,81	97,54	<b>98,36</b>	<b>98,36</b>	97,81
MLP-Ibk	98,36	98,36	<b>98,63</b>	<b>98,63</b>	<b>98,63</b>	<b>98,63</b>	98,36	98,36	96,72
J48-NB	<b>98,63</b>	97,54	97,27	97,27	97,27	97,27	96,17	96,45	98,09
J48-Ibk	<b>98,09</b>	96,45	95,9	95,9	95,9	95,9	96,17	96,17	96,45
NB-Ibk	96,99	96,72	<b>97,54</b>	<b>97,54</b>	<b>97,54</b>	<b>97,54</b>	97,27	97,27	97,27
SMO-Ibk-NB	<b>98,36</b>	98,09	97,54	97,81	97,54	97,81	97,81	98,09	97,54
SMO-Ibk-MLP	98,36	98,36	<b>98,63</b>	<b>98,63</b>	<b>98,63</b>	<b>98,63</b>	98,09	98,36	96,99
SMO-Ibk-J48	97,81	97,54	95,9	95,9	95,9	95,9	<b>98,09</b>	97,81	97,54
SMO-J48-MLP	98,36	98,09	97,81	97,54	97,81	97,54	<b>98,91</b>	<b>98,91</b>	<b>98,91</b>
SMO-J48-NB	98,09	97,81	97,27	97,27	97,27	97,27	98,63	98,63	<b>98,91</b>
SMO-MLP-NB	98,09	98,36	<b>99,18</b>	<b>99,18</b>	<b>99,18</b>	<b>99,18</b>	98,36	98,63	98,09
J48-MLP-NB	98,63	98,36	98,91	98,63	98,91	98,63	<b>99,45</b>	<b>99,45</b>	<b>99,45</b>
J48-MLP-Ibk	98,09	98,09	98,09	97,81	98,09	97,81	<b>98,91</b>	<b>98,91</b>	98,63
J48-NB-Ibk	98,63	96,99	96,99	97,27	96,99	97,27	98,36	98,63	<b>98,91</b>
MLP-NB-Ibk	98,63	98,36	<b>99,18</b>	<b>99,18</b>	<b>99,18</b>	<b>99,18</b>	97,54	98,36	98,09
SMO-Ibk-NB-MLP	98,09	98,36	<b>99,18</b>	<b>99,18</b>	<b>99,18</b>	<b>99,18</b>	98,36	98,63	98,09
SMO-Ibk-NB-J48	98,09	97,81	97,27	97,27	97,27	97,27	98,09	98,09	<b>98,91</b>
SMO-J48-MLP-NB	98,36	98,36	98,91	98,63	98,91	98,63	98,36	98,63	<b>99,45</b>
SMO-J48-MLP-Ibk	<b>98,36</b>	98,09	98,09	97,81	98,09	97,81	<b>98,36</b>	<b>98,36</b>	<b>98,36</b>
J48-MLP-NB-Ibk	98,63	98,36	<b>98,91</b>	98,63	<b>98,91</b>	98,63	<b>98,91</b>	98,63	98,63

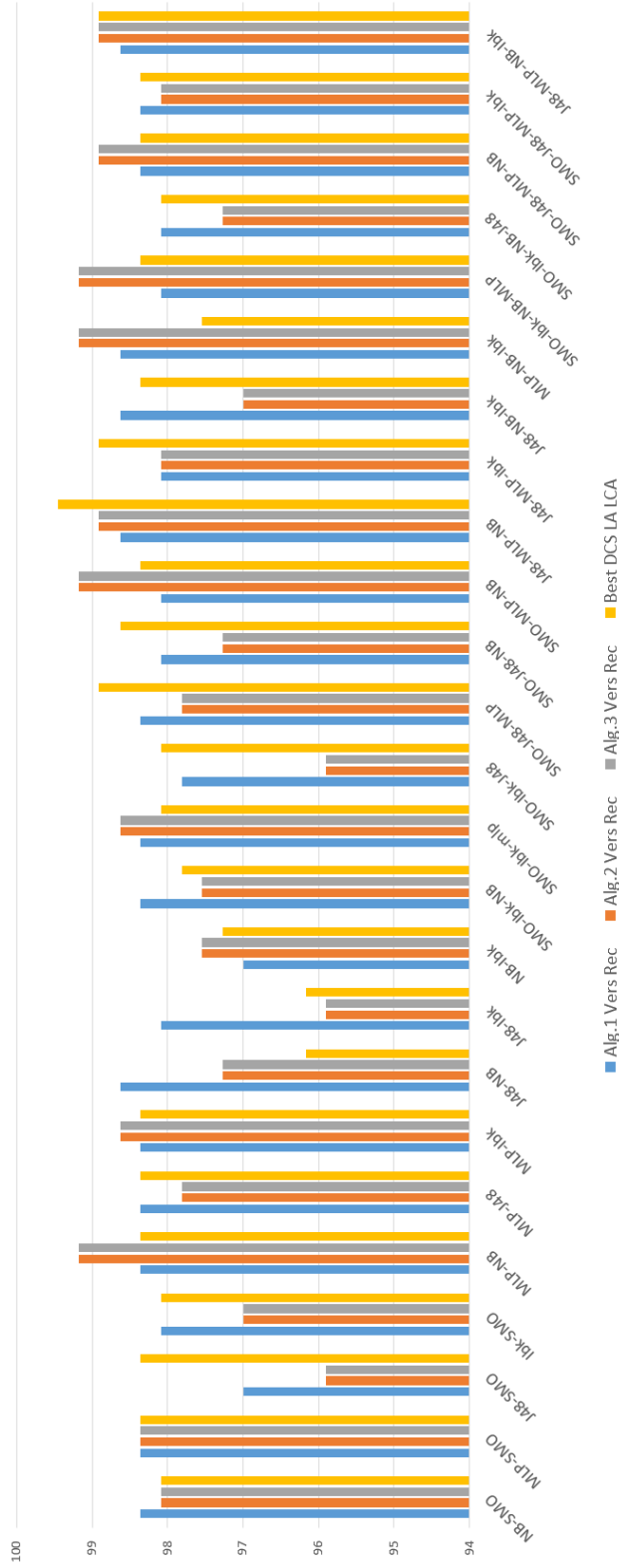


Figura 6.16: Dermatology Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Recall con Best DCS LA LCA



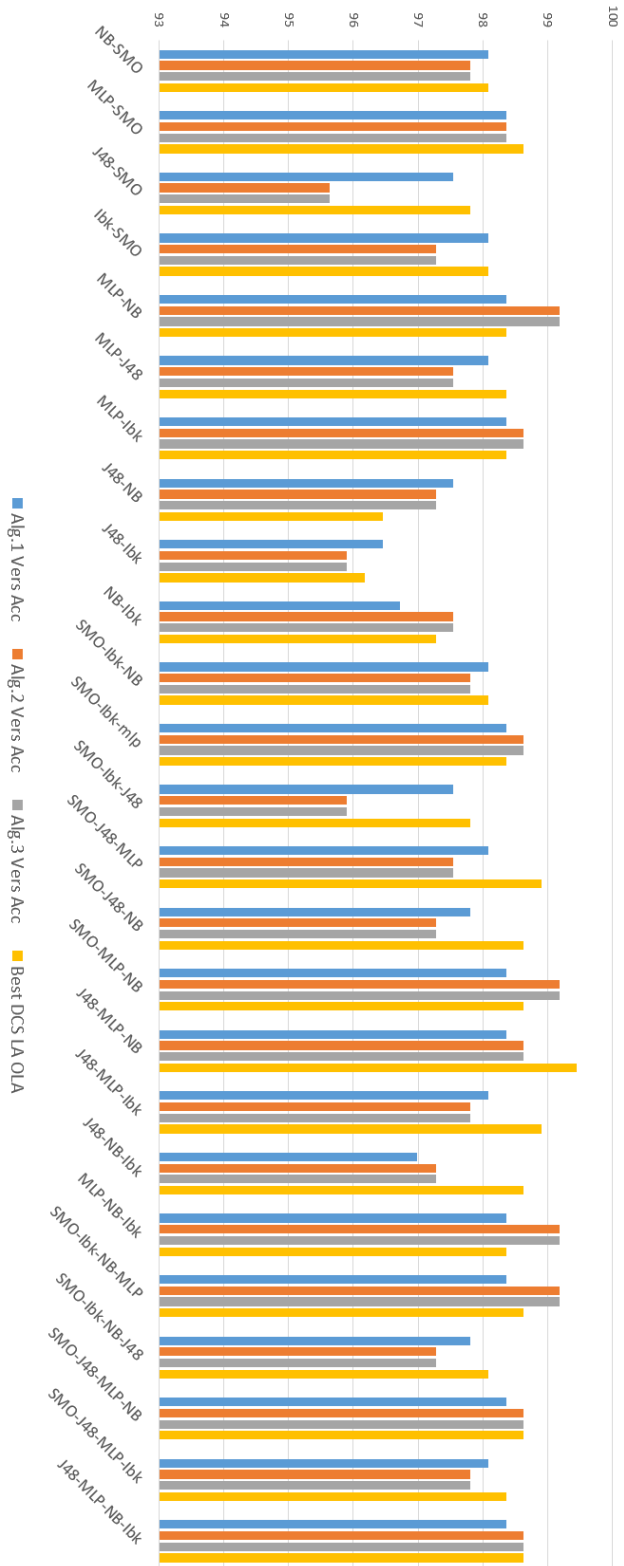


Figura 6.17: Dermatology Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in versione Accuratezza con Best DCS IA OLA

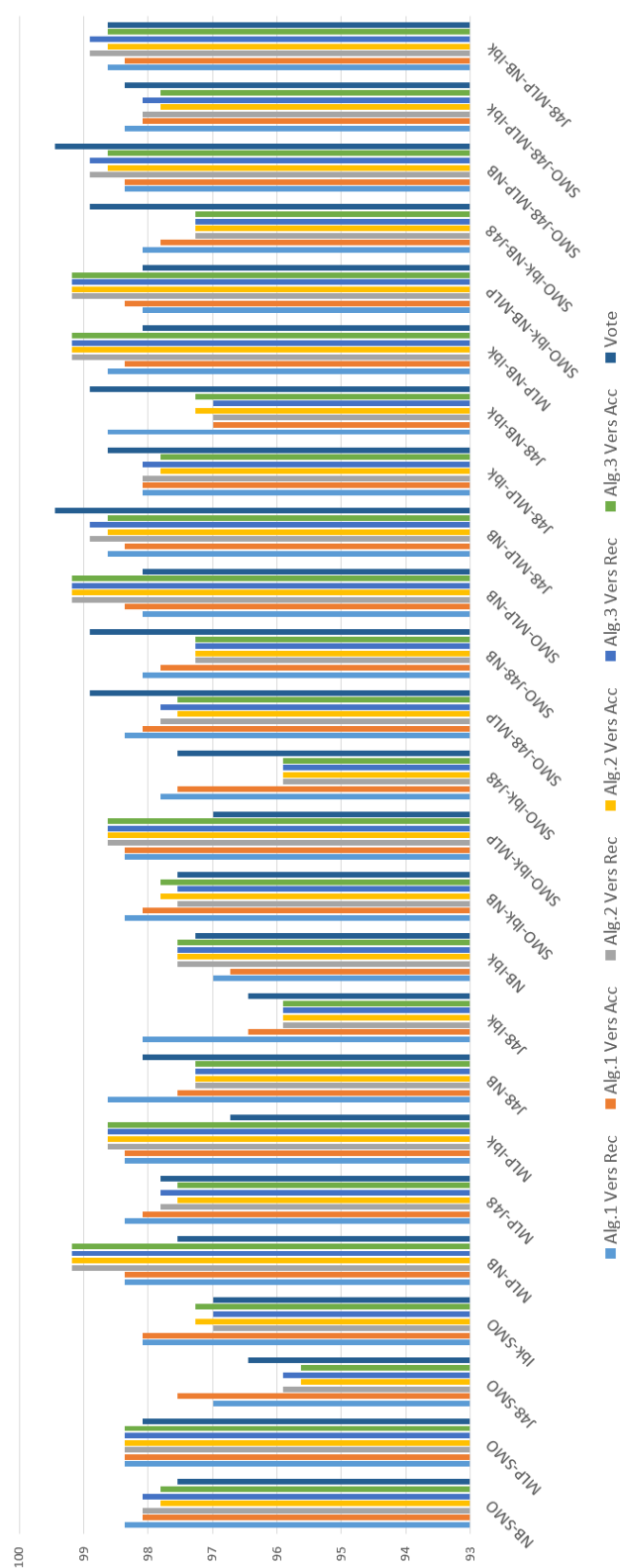


Figura 6.18: Dermatology Dataset: Confronto tra Algoritmo 1, Algoritmo 2 e Algoritmo 3 in entrambe le versioni con Vote



# Capitolo 7

## Sviluppo del prototipo

A scopi puramente dimostrativi è stato implementato, sulla base degli approcci proposti, un prototipo di CDSS per la diagnosi del tumore al seno. Il Multi Classificatore è stato addestrato con i dati del Wisconsin Diagnosis Breast Cancer Dataset (WDBC). Come descritto nella Sezione 7.1.1, i dati si riferiscono a pazienti che si sono sottoposte a esame di agoaspirato di un nodulo mammario. Ciascuna istanza descrive le caratteristiche dei nuclei cellulari presenti nei campioni prelevati durante l'esame. Il dataset viene usato per distinguere noduli benigni da quelli maligni. Il Multi Classificatore è stato interamente implementato in Java attraverso l'uso dell'ambiente di sviluppo Eclipse e importando le API (Application Programming Interface) del software WEKA (Waikato Environment for Knowledge Analysis). Per lo sviluppo dell'interfaccia web è stato utilizzato invece HLML5. Il prototipo è stato implementato unicamente per scopi di ricerca, di conseguenza l'interfaccia manca di alcune funzionalità che saranno sicuramente implementate in versioni successive. L'intero prototipo inoltre, necessita senza dubbio di un'ulteriore fase di ingegnerizzazione per essere presentato come strumento di ricerca stabile ed affidabile.

## 7.1 Interfaccia utente

Al fine di testare il funzionamento del sistema, è stata implementata un'interfaccia utente web-based. Uno screenshot della versione corrente è mostrato in Figura 7.1. L'interfaccia permette l'inserimento dei dati necessari

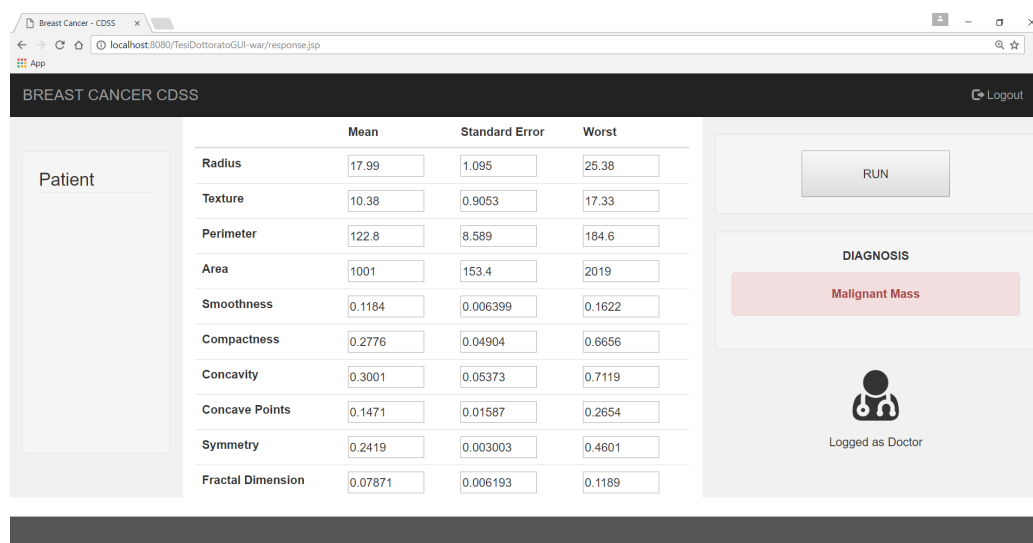
	Mean	Standard Error	Worst
Radius	<input type="text"/>	<input type="text"/>	<input type="text"/>
Texture	<input type="text"/>	<input type="text"/>	<input type="text"/>
Perimeter	<input type="text"/>	<input type="text"/>	<input type="text"/>
Area	<input type="text"/>	<input type="text"/>	<input type="text"/>
Smoothness	<input type="text"/>	<input type="text"/>	<input type="text"/>
Compactness	<input type="text"/>	<input type="text"/>	<input type="text"/>
Concavity	<input type="text"/>	<input type="text"/>	<input type="text"/>
Concave Points	<input type="text"/>	<input type="text"/>	<input type="text"/>
Symmetry	<input type="text"/>	<input type="text"/>	<input type="text"/>
Fractal Dimension	<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 7.1: Interfaccia web-based per il Breast Cancer CDSS

al KM per la formulazione di un suggerimento di diagnosi. In particolare, per ogni nucleo cellulare è previsto l'inserimento delle dieci caratteristiche che descrivono le istanze del WDBC Dataset e per ciascuna l'inserimento di media, errore standard e valore peggiore (o più grande) osservato. In questa versione dimostrativa dell'interfaccia utente è previsto l'inserimento manuale dei dati. L'obiettivo comunque è quello di estendere tale versione in modo da permettere l'interfacciamento del CDSS direttamente con la cartella clinica elettronica. Attraverso questa funzionalità, i dati potranno essere acquisiti in maniera automatica dal sistema con un notevole risparmio di tempo da parte dell'utente e una maggiore semplicità di utilizzo.

Il CDSS si attiva alla pressione del pulsante RUN. Il suggerimento di diagnosi viene mostrato sull'interfaccia con uno sfondo di colore verde se i

dati valutati suggeriscono la presenza di un nodulo benigno, di colore rosso invece se suggeriscono la presenza di un nodulo maligno. Gli screenshot dei due possibili suggerimenti di diagnosi sono mostrati in Figura 7.2 e Figura 7.3.



The screenshot shows a web browser window with the URL `localhost:8080/TesiDottoratoGUI-war/response.jsp`. The page title is "BREAST CANCER CDSS" and it includes a "Logout" link. The interface is divided into three main sections:

- Patient:** A placeholder area on the left.
- Input Table:** A table with columns "Mean", "Standard Error", and "Worst". Each cell contains a numerical value and a text input field.
- Output Section:** A "RUN" button, a "DIAGNOSIS" label, a red box containing the text "Malignant Mass", a doctor icon, and the text "Logged as Doctor".

	Mean	Standard Error	Worst
Radius	17.99	1.095	25.38
Texture	10.38	0.9053	17.33
Perimeter	122.8	8.589	184.6
Area	1001	153.4	2019
Smoothness	0.1184	0.006399	0.1622
Compactness	0.2776	0.04904	0.6656
Concavity	0.3001	0.05373	0.7119
Concave Points	0.1471	0.01587	0.2654
Symmetry	0.2419	0.003003	0.4601
Fractal Dimension	0.07871	0.006193	0.1189

Figura 7.2: Interfaccia web-based in caso di diagnosi di nodulo maligno

## 7.2 Strato di interoperabilità

Dal punto di vista implementativo, la pressione del pulsante RUN attiva una chiamata AJAX al web service che espone il CDSS. Il web service fa parte dello strato di interoperabilità che implementa lo standard HL7 DSS. A livello generale lo strato di interoperabilità riceve le richieste da parte del consumer, le valida e, in caso di verifica positiva le invia al CDSS che si intende consultare. Lo strato di interoperabilità è responsabile anche del confezionamento delle risposte e dell'invio di queste ultime al richiedente. Dal punto di vista architetturale lo strato di interoperabilità si compone oltre che del web service di un componente che funge da broker. Il broker, in questo caso, si

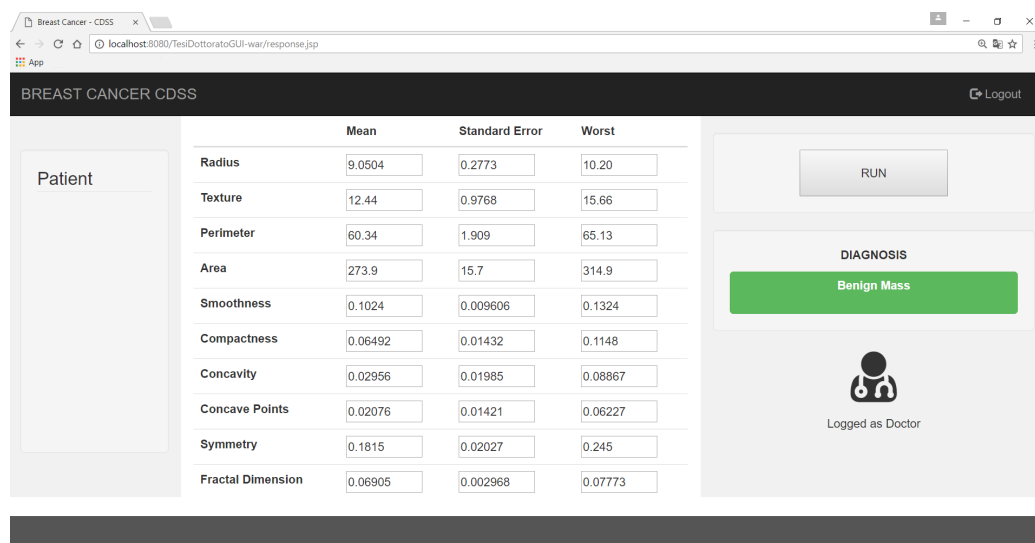


Figura 7.3: Interfaccia web-based in caso di diagnosi di nodulo benigno

occupa di trattare la richiesta inviata dal web service, incaricandosi di localizzare e recuperare prima il CDSS che il consumer intende consultare e poi di verificare se tutti i dati, che quel CDSS utilizza per fare le sue valutazioni, sono stati forniti e se sono nel formato corretto. Poiché lo standard prevede che tutti i dati dei pazienti valutati da un CDSS devono essere codificati nel formato Base64, il broker si occupa anche di decodificare il payload di ogni richiesta e codificare il payload di ogni risposta. Solo in seguito a tali verifiche il broker invia i dati al CDSS per il processamento. Ovviamente, nel caso in cui le verifiche non vanno a buon fine, il broker restituisce al chiamante la relativa eccezione. A valutazione ultimata il CDSS inoltra la risposta, ovvero il suggerimento di diagnosi, verso lo strato di interoperabilità che a sua volta lo inoltra verso l'interfaccia web per la visualizzazione dell'utente. In Figura 7.4 sono mostrate le principali interazioni tra l'interfaccia utente, lo strato di interoperabilità e il CDSS.

Lo sviluppo dello strato di interoperabilità ha seguito l'intero ciclo di progettazione e sviluppo previsto dall'approccio MDA. L'attività di model-

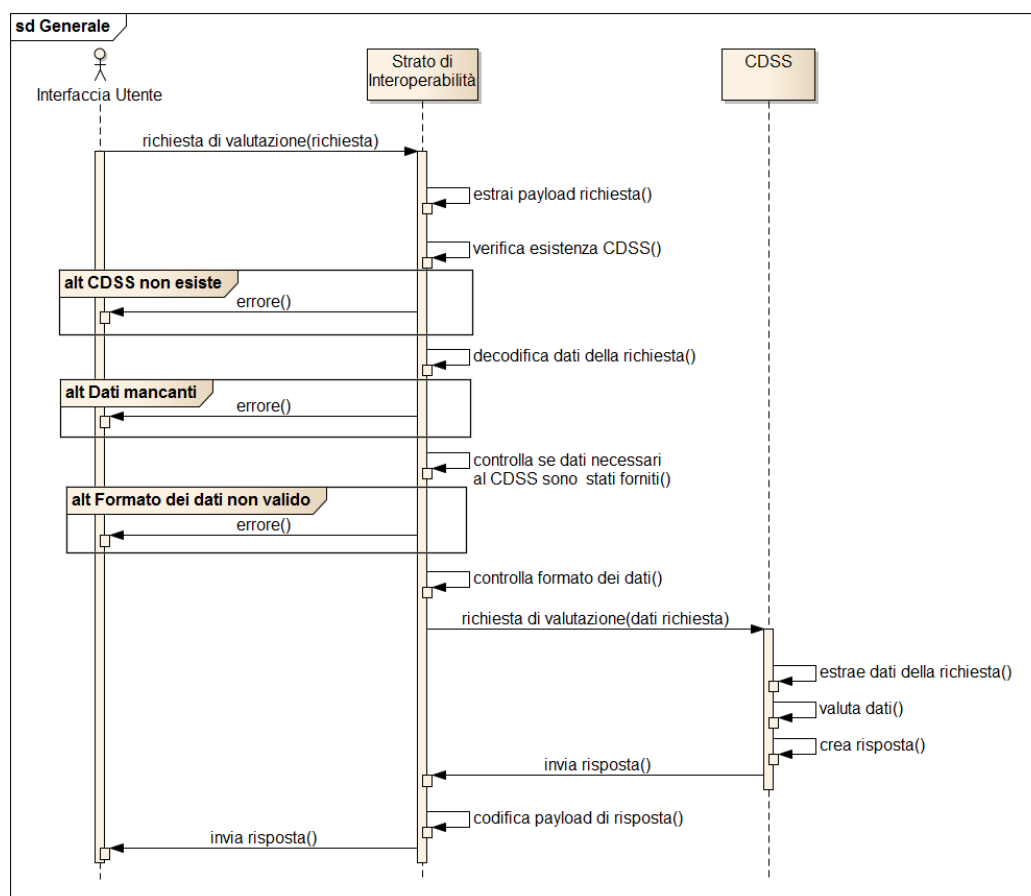


Figura 7.4: Principali interazioni tra l'interfaccia utente, lo strato di interoperabilità e il CDSS

lazione delle specifiche tecniche fornite dallo standard si è tradotta nella definizione del Service PIM: la modellazione del contratto di servizio tra un CDSS Consumer ed un CDSS Provider. In Figura 7.5 è riportato il contratto di servizio definito per l'operazione *evaluate* della *Evaluation Interface* che è stata successivamente implementata su piattaforma Web Service. Il Contratto di servizio (Service Contract) specifica come gli erogatori e i fruitori del servizio interagiscono tra loro. Per definire la conversazione tra erogatori e fruitori del contratto di servizio sono stati usati gli UML State Machine, mentre per definire l'organizzazione delle attività eseguite dalle parti del con-



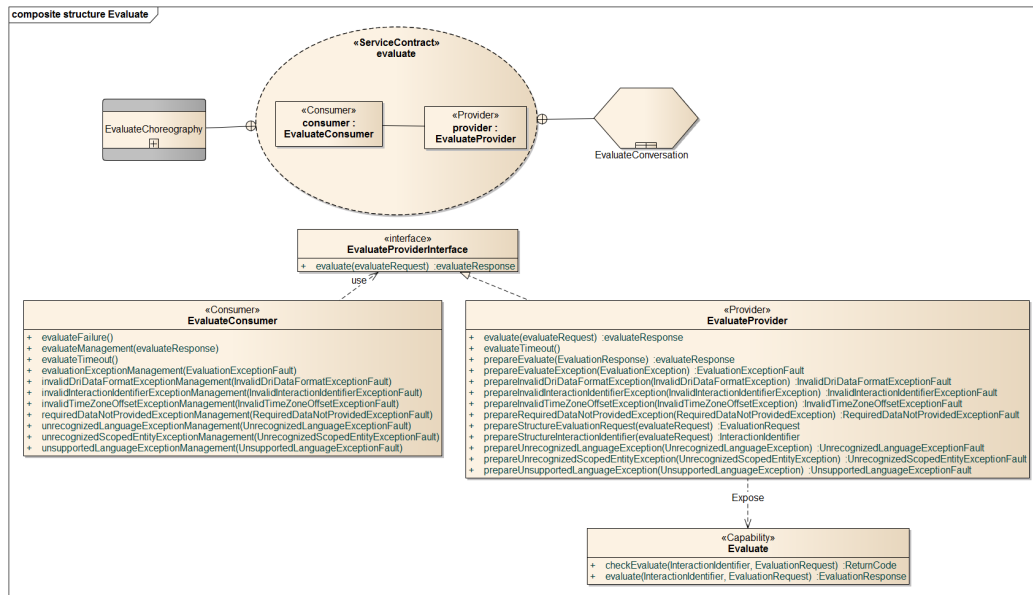


Figura 7.5: Contratto di servizio definito per l'operazione *evaluate* della *Evaluation Interface* dello standard HL7 DSS

tratto di servizio sono stati usati gli UML Activity Diagram. Attraverso gli UML State Machine, il comportamento è modellato come un attraversamento di un grafo di nodi di stato, interconnessi mediante uno o più archi di transizione che vengono attivati dall'invio di una serie di occorrenze di eventi. In Figura 7.6 la modellazione della conversazione tra erogatori e fruitori del servizio. L'utente del servizio (il Consumer) fornisce alcuni parametri e richiede l'esecuzione dell'operazione *evaluate ()*; l'erogatore del servizio (il Provider) prende in carico la richiesta e fornisce al Consumer una *evaluate-Response()* se tutti i controlli sui dati in essa contenuti sono andati a buon fine. In caso contrario restituisce la relativa eccezione. Nello specifico, la conversazione fa uso dei seguenti stati:

- *evaluate*: l'utente del servizio fornisce alcuni parametri e richiede l'esecuzione dell'operazione del servizio
- *evaluateResponse*: l'erogatore del servizio invia i risultati ottenuti dal-

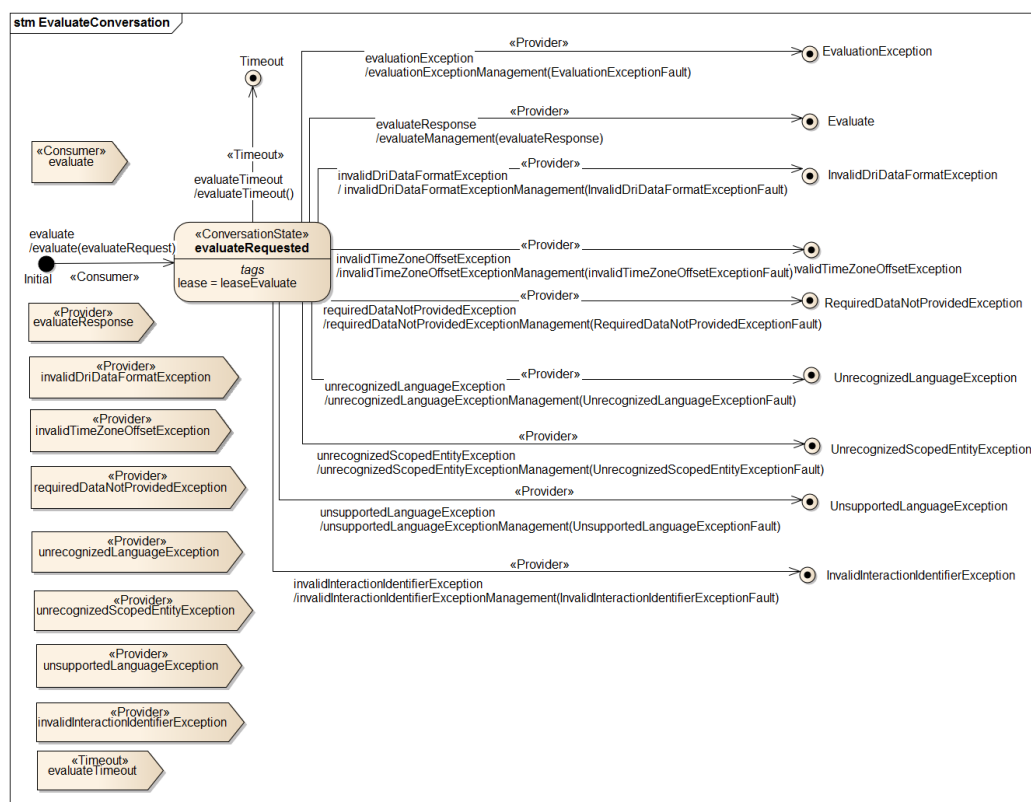


Figura 7.6: Modellazione della conversazione tra erogatore e fruitore del servizio

l'esecuzione dell'operazione del servizio

- `invalidDriTimeZoneOffsetException`: l'erogatore del servizio non esegue l'operazione visto che il `TimeZoneOffset` della richiesta non è valido
- `dataNotProvidedException`: l'erogatore del servizio non esegue l'operazione poiché alcuni dati necessari per eseguire l'operazione non sono stati forniti
- `unrecognizedLangageException`: l'erogatore del servizio non esegue l'operazione poiché la lingua dell'utente non è riconosciuta dal servizio

- unrecognizedScopingEntityException: l'erogatore del servizio non esegue l'operazione poiché l'entità non è riconosciuta dal servizio
- unsupportedLanguageException: l'erogatore del servizio non esegue l'operazione poiché la lingua dell'utente non è supportata dal servizio
- invalidInteractionIdentifierException: l'erogatore del servizio non esegue l'operazione poiché l'id che identifica la transizione non è valido
- evaluateTimeout: il fruitore del servizio lancia un evento di timeout se trascorso un determinato periodo di tempo non ottiene alcuna risposta

In Figura 7.7 e Figura 7.8 la modellazione della coreografia tra il consumer ed il provider dell'operazione di servizio *evaluate()*. Il consumer in seguito

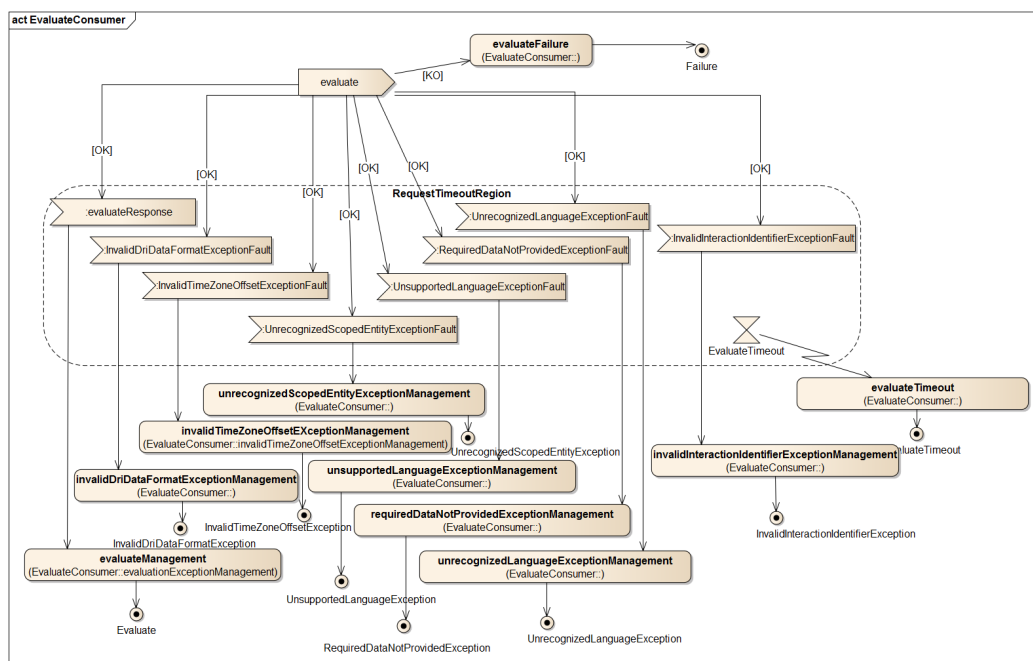


Figura 7.7: Modellazione della coreografia tra erogatore e fruitore del servizio lato Consumer

all'invio della richiesta può ricevere dal provider una risposta oppure una delle

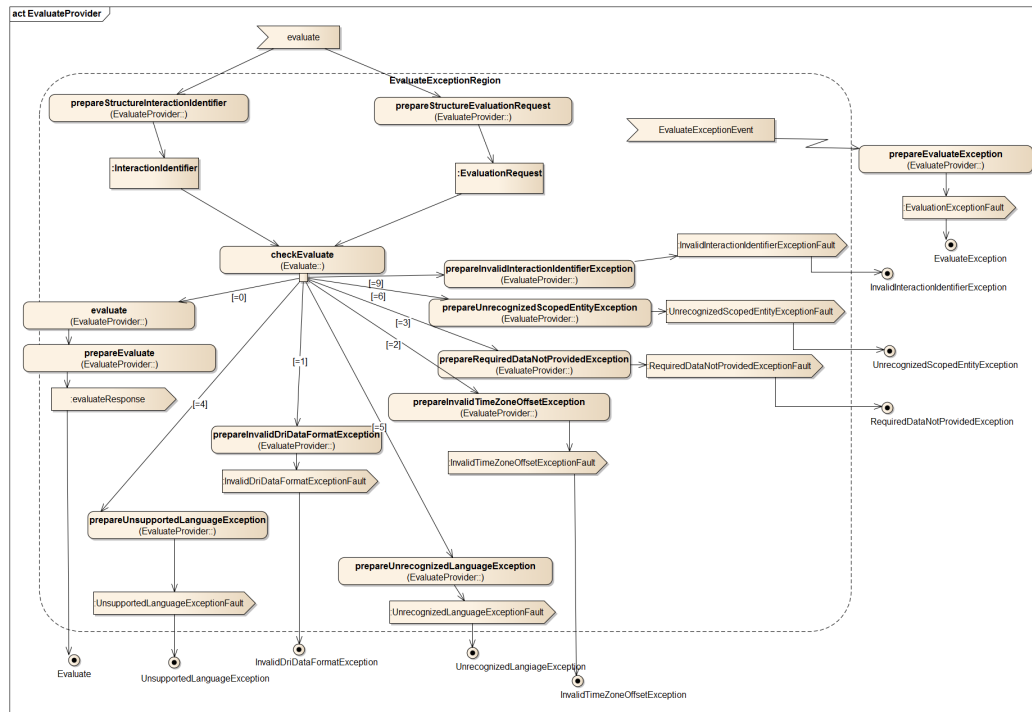


Figura 7.8: Modellazione della coreografia tra erogatore e fruitore del servizio lato Provider

tante eccezioni previste. In ogni caso si occuperà di gestirle. Può lanciare e gestire un evento di timeout se non ottiene una risposta dal provider in un lasso di tempo prestabilito. Può inoltre gestire il fallimento dell'operazione dovuto a qualsiasi altro motivo verificatosi durante l'esecuzione. Il provider invece, in seguito alla ricezione di una richiesta dell'operazione di servizio estrae innanzitutto le strutture dati che compongono la richiesta e verifica tutti i dati in esso contenuti. Se tutti i dati sono corretti valuta i dati relativi al paziente, prepara la risposta e la invia al consumer. Se invece viene riscontrato qualche errore prepara e invia le relative eccezioni.

### 7.3 CDSS

In Figura 7.9 sono riportati i componenti dell'architettura software del CDSS e le interazioni che tra questi intercorrono. L'interfaccia del CDSS espone verso l'esterno le funzionalità che esso mette a disposizione dei suoi utilizzatori. In questo caso espone una funzione che permette di valutare le caratteristiche del WDBC Dataset. L'input della funzione è un file XML che contiene tali dati. Il Controllore Strategico acquisisce la richiesta, estrae dal file XML i valori su cui effettuare le valutazioni e li invia al KM. Il KM implementa il Multi Classificatore basato sull'approccio definito dall'Algoritmo 2 e usa l'insieme di classificatori di base costituito da una Support Vector Machine e un classificatore di tipo Bayesiano (NB-SMO). A valutazione ultimata il Controllore Strategico acquisisce il suggerimento di diagnosi prodotto dal KM, crea un file XML di risposta e lo inoltra verso l'interfaccia per il successivo invio allo strato di interoperabilità prima e all'interfaccia utente subito dopo. In Figura 7.10 un esempio di file XML usato come input dal CDSS mentre in Figura 7.11 un esempio di file XML di risposta prodotto dal CDSS.

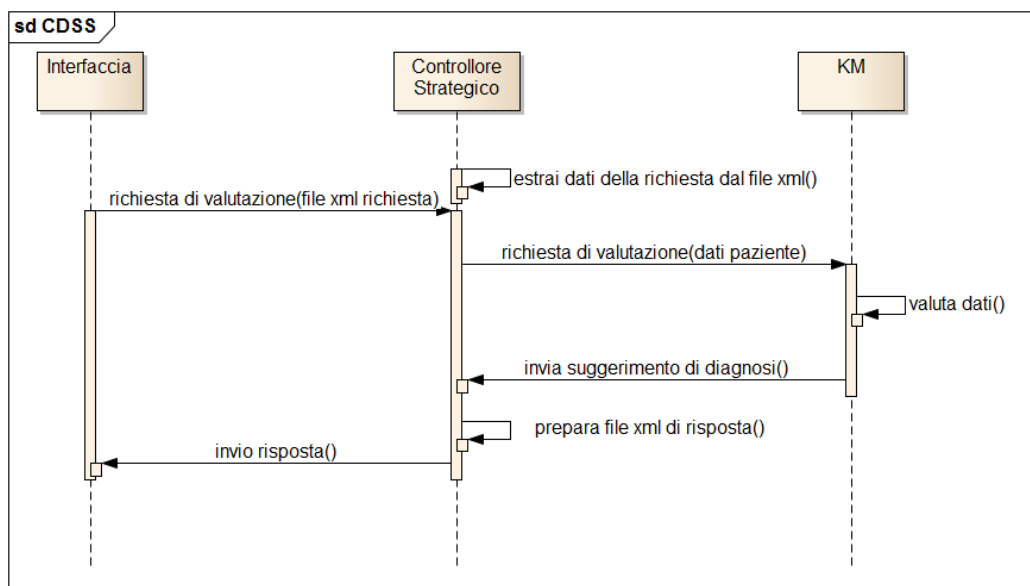


Figura 7.9: Principali interazioni tra i componenti principali dell'architettura software del CDSS

```

<dss:request xmlns:dss="http://www.omg.org/spec/CDSS/201012/dss">
  <radius_Mean>17.99</radius_Mean>
  <radius_StandardError>1.095</radius_StandardError>
  <radius_Worst>25.38</radius_Worst>
  <texture_Mean>10.38</texture_Mean>
  <texture_StandardError>0.9053</texture_StandardError>
  <texture_Worst>17.33</texture_Worst>
  <perimeter_Mean>122.8</perimeter_Mean>
  <perimeter_StandardError>8.589</perimeter_StandardError>
  <perimeter_Worst>184.6</perimeter_Worst>
  <area_Mean>1001</area_Mean>
  <area_StandardError>153.4</area_StandardError>
  <area_Worst>2019</area_Worst>
  <smoothness_Mean>0.1184</smoothness_Mean>
  <smoothness_StandardError>0.006399</smoothness_StandardError>
  <smoothness_Worst>0.1622</smoothness_Worst>
  <compactness_Mean>0.2776</compactness_Mean>
  <compactness_StandardError>0.04904</compactness_StandardError>
  <compactness_Worst>0.6656</compactness_Worst>
  <concavity_Mean>0.3001</concavity_Mean>
  <concavity_StandardError>0.05373</concavity_StandardError>
  <concavity_Worst>0.7119</concavity_Worst>
  <concavePoint_Mean>0.1471</concavePoint_Mean>
  <concavePoint_StandardError>0.01587</concavePoint_StandardError>
  <concavePoint_Worst>0.2654</concavePoint_Worst>
  <simmetry_Mean>0.2419</simmetry_Mean>
  <simmetry_StandardError>0.003003</simmetry_StandardError>
  <simmetry_Worst>0.4601</simmetry_Worst>
  <fractalDimension_Mean>0.07871</fractalDimension_Mean>
  <fractalDimension_StandardError>0.006193</fractalDimension_StandardError>
  <fractalDimension_Worst>0.1189</fractalDimension_Worst>
</dss:request>

```

Figura 7.10: Esempio file XML di richiesta per il CDSS

```

<ris:response xmlns:ris="http://www.omg.org/spec/CDSS/201012/risposta">
  <massType>BENIGN MASS</massType>
</ris:response>

```

Figura 7.11: Esempio file XML di risposta del CDSS

# Conclusioni e Sviluppi Futuri

La tesi di dottorato propone contributi innovativi fortemente orientati alle applicazioni reali sia in termini di architettura software per CDSS che, di approcci metodologici, basati su tecniche di Machine Learning (ML), per il disegno e lo sviluppo di Sistemi Multi Classificatori basati sulla tecnica della selezione dinamica.

L'architettura software per CDSS si basa su tre componenti fondamentali distinte ma cooperanti:

- Un interfaccia per gestire le comunicazioni con l'esterno
- Un controllore strategico che ha il compito di selezionare, istanziare e orchestrare i KM necessari per elaborare il più efficace supporto decisionale
- Un insieme di KM

Ciascun KM è un motore computazionale in grado di valutare i dati dei pazienti e fornire informazioni circa il loro stato di salute. Ciascun KM è indipendente dagli altri e può essere sviluppato e implementato sia secondo approcci deduttivi che induttivi. L'architettura del CDSS così per come è stata concepita facilita l'integrazione e la cooperazione tra KM attraverso il Controllore Strategico. Propone, infatti, un modello di CDSS capace di sfruttare più modelli computazionali e di scegliere in maniera intelligente lo strumento più adatto rispetto ai dati ricevuti in input. Avere un'architettura di questo tipo consente di mantenere a livelli elevati le caratteristiche di flessibilità, sia in termini di funzionalità che di ampiezza nella copertura



del dominio clinico di riferimento. L'architettura offre inoltre soluzioni atte a facilitare lo scambio di dati e l'interoperabilità dei servizi del settore sanitario attraverso uno strato software di interoperabilità basato su standard internazionali. Attraverso l'adozione dello standard il CDSS è ricercabile e utilizzabile da chiunque volesse farne uso.

La tesi propone inoltre la realizzazione di ciascun KM come un MCS. Nello specifico, sono stati definiti tre differenti approcci per la modellazione di MCS basati sulla tecnica della selezione dinamica. L'idea alla base degli approcci proposti è quella di non combinare le decisioni dei singoli classificatori ma far classificare direttamente un'istanza al modello che ha maggiore probabilità di assegnarla alla classe corretta. I MCS utilizzano una tecnica di selezione dinamica appositamente definita. Il metodo di selezione dinamica proposto in questa tesi tiene conto sia delle istanze mal classificate da ciascun classificatore di base all'interno della regione locale che delle prestazioni generali di ognuno. La regione locale su cui viene calcolata la competenza di ciascun classificatore di base è definita in modo dinamico per ciascun nuovo campione di test attraverso un algoritmo K-Nearest Neighbor (KNN) di tipo adattivo. L'algoritmo adattivo definito in questa tesi, mantiene nella regione locale solo le istanze che hanno una distanza dall'istanza test inferiore a una certa soglia. In questo modo si cerca di considerare solo le istanze che sono molto vicine all'istanza test sotto l'ipotesi che istanze con attributi simili dovrebbero essere vicine tra loro e dovrebbero appartenere alla stessa classe. La selezione del classificatore più competente all'interno della regione locale, avviene in due fasi. Durante la prima fase, si escludono i classificatori che hanno erroneamente classificato una o più istanze all'interno della regione. Nella seconda fase, viene scelto il classificatore più competente tra tutti quelli rimasti. Il classificatore più competente è quello che presenta prestazioni migliori. Sono state definite due diverse strategie: una utilizza come indice di performance la recall mentre l'altra utilizza l'accuratezza. I conflitti (ovvero più classificatori hanno le stesse prestazioni) sono stati risolti secondo tre approcci diversi che hanno portato alla definizione di tre differenti versioni

dell'algoritmo di classificazione. La prima versione dell'algoritmo risolve i conflitti utilizzando le performance che i classificatori di base hanno sia sul training che sul validation set, la seconda utilizza solo le performance dei classificatori di base sul validation set mentre la terza combina elementi delle due in una versione di tipo ibrido. Gli approcci proposti sono stati testati su dataset pubblici che si riferiscono a importanti problemi decisionali in ambito biomedico e clinico. Tutti gli esperimenti sono stati eseguiti sulla base della 10-fold cross validation stratificata. Per ogni dataset sono stati settati i parametri ottimali di ciascun classificatore e su ognuno sono state testati diversi insiemi di classificatori di base. In particolare, i classificatori che costituiscono gli insiemi sono stati estratti dal seguente set di metodi: Support Vector Machine, Multilayer Perceptron, Naïve Bayes, Decision Tree, K Nearest Neighbours. Tali classificatori sono stati utilizzati in gruppi di due, tre e quattro per volta. Complessivamente, i risultati sperimentali mostrano che l'accuratezza ottenuta su tutti i dataset è confrontabile e, in alcuni casi, migliore rispetto a quella ottenuta utilizzando altre tecniche proposte in letteratura. In generale le tre versioni dell'algoritmo hanno prestazioni simili. Le prestazioni variano in base al dataset considerato.

In futuro l'architettura del sistema verrà ulteriormente ampliata; lo strato di interoperabilità favorirà oltre all'interoperabilità applicativa anche l'interoperabilità semantica tra il CDSS e le altre applicazioni del sistema sanitario. Il CDSS potrà così essere utilizzato da una più vasta gamma di utenti e organizzazioni sanitarie.

Saranno implementati KM in grado di cooperare tra loro. In particolare, saranno implementati KM capaci di utilizzare come input per le loro valutazioni i risultati prodotti da altri KM. Essi saranno in grado di istanziare in maniera automatica altri KM per ottenere da questi i dati e le informazioni utili a completare il loro compito. Ciascun KM sarà dunque anche in grado di fornire i risultati delle sue valutazioni ai KM che ne faranno richiesta. Ovviamente, ciascun KM sarà in grado di completare il compito decisionale indipendentemente dagli altri (ad esempio nel caso di problemi decisionali

più semplici) ma può sfruttare i risultati di altri KM nel caso di problemi decisionali più complessi.

Un CDSS così costituito potrà essere sicuramente utile nei casi di diagnosi molto complesse che prevedono la valutazione di un numero molto elevato di ipotesi diagnostiche. Un CDSS capace di consultare e integrare i risultati di più motori computazionali potrà quasi certamente essere più affidabile rispetto ad altri sistemi che si basano sui risultati di un solo modello.

Sviluppi futuri riguarderanno anche il Controllore Strategico; sarà implementata all'interno del Controllore Strategico una nuova politica di gestione dei KM. Il Controllore Strategico sarà in grado di consultare in parallelo più KM integrando successivamente i loro suggerimenti di diagnosi nella decisione finale. Esso, in base alla complessità della decisione consulterà prima un certo numero di KM e qualora ritenesse tali decisioni non soddisfacenti o contrastanti tra loro potrà decidere di consultarne altri costruiti per gli stessi scopi.

In futuro, saranno utilizzate nuove metodologie di classificazione. Si utilizzeranno pertanto nuovi classificatori di base. Sono previste, infatti, nuove sessioni sperimentali con insiemi di classificatori di base diversi da quelli già usati in questa sessione.

Ulteriori sviluppi riguarderanno inoltre la costruzione della regione locale. In particolare:

- Saranno testati nuovi criteri di arresto dell'algoritmo KNN adattativo
- Saranno usate altre metriche per calcolare la similarità delle istanze nella regione. Si è pensato di usare metriche note, quali la distanza di Manhattan e la similarità basata su kernel, di definire metriche ad hoc e di utilizzare metriche diverse in zone diverse dell'input space. Inoltre, poiché le prestazioni di molti algoritmi di apprendimento automatico dipendono dalla metrica di distanza utilizzata anche nello spazio delle feature, saranno utilizzati approcci di "metric distance learning" per apprendere la metrica di distanza più opportuna in relazione al classificatore di base utilizzato.

Infine, per quanto riguarda l'implementazione del prototipo, l'interfaccia utente sarà dotata di nuove funzionalità; è stato già previsto l'interfacciamento con la cartella clinica elettronica per l'acquisizione automatica dei dati del paziente. È previsto inoltre lo sviluppo di un'interfaccia mobile.



# Bibliografia

- [1] HL7-<http://www.hl7.org/>.
- [2] IHE-<http://www.ihe-italy.org/>.
- [3] Loinc-<https://loinc.org/>.
- [4] OMG-<http://www.omg.org/>.
- [5] OpenCDS - <http://www.opencds.org/>.
- [6] Snomed. URL <http://www.snomed.org/snomed-ct>.
- [7] UC I Learning Repository - <https://archive.ics.uci.edu/ml/>.
- [8] Robert Allen, Remi Douence, and David Garlan. *Specifying and analyzing dynamic software architectures*, volume 1382, pages 21–37. Lisbon, Portugal, March 28, 1998.
- [9] Eta S. Berner. *Clinical Decision Support Systems: Theory and Practice*. Springer Publishing Company, Incorporated, 2nd edition, 2010. ISBN 1441922237, 9781441922236.
- [10] Cho BH, Yu H, Kim K-W, Kim TH, Kim IY, and Kim SI. Application of irregular and unbalanced data to predict diabetic nephropathy using visualization and feature selection methods. *Artif Intell Med*, 42(1): 37–53, 2008.

- [11] Yaxin Bi. The impact of diversity on the accuracy of evidential classifier ensembles. *International Journal of Approximate Reasoning*, 53(4):584–607, 2012.
- [12] Stefan Bjornander. Architecture description languages. Technical report.
- [13] R.J. Brachman and H.J. Levesque. *Knowledge representation and reasoning*. Elsevier/Morgan Kaufmann Publishers, San Francisco, CA, USA, 2004.
- [14] Antonio Candelieri, Domenico Conforti, Angela Sciacqua, and Francesco Perticone. Knowledge discovery approaches for early detection of decompensation conditions in heart failure patients. In *Intelligent Systems Design and Applications, 2009. ISDA'09. Ninth International Conference on*, pages 357–362. IEEE, 2009.
- [15] Paulo R Cavalin, Robert Sabourin, and Ching Y Suen. Logid: An adaptive framework combining local and global incremental learning for dynamic selection of ensembles of HMMs. *Pattern Recognition*, 45(9):3544–3556, 2012.
- [16] Paulo R Cavalin, Robert Sabourin, and Ching Y Suen. Dynamic selection approaches for multiple classifier systems. *Neural Computing and Applications*, 22(3):673–688, 2013.
- [17] Yunqian Ma Cha Zhang. *Ensemble Machine Learning*. Springer-Verlag, New York, 2012. ISBN 978-1-4899-8817-1.
- [18] Jeeyae Choi, Leanne M. Currie, Dongwen Wang, and Suzanne Bakken. Encoding a clinical practice guideline using guideline interchange format: A case study of a depression screening and management guideline. *International Journal of Medical Informatics*, 76, Supplement 2: S302–S307, 2007. ISSN 1386-5056. Nursing Informatics 2006 Special Issue.

- 
- [19] C. K. Chow. Statistical independence and threshold functions. *IEEE Transactions on Electronic Computers*, EC-14(1):66–68, 1965.
- [20] Rafael M. O. Cruz, Robert Sabourin, and George D. C. Cavalcanti. Prototype selection for dynamic classifier and ensemble selection. *Neural Computing and Applications*, pages 1–11, 2016.
- [21] B. V. Dasarathy and B. V. Sheela. A composite classifier system design: Concepts and methodology. *Proceedings of the IEEE*, 67(5):708–713, 1979.
- [22] F. T. de Dombal, D. J. Leaper, J. R. Staniland, A. P. McCann, and Jane C. Horrocks. Computer-aided diagnosis of acute abdominal pain. *BMJ*, 2(5804):9–13, 1972.
- [23] T. G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995.
- [24] Pedro Domingos. A unified bias-variance decomposition and its applications. In *In Proc. 17th International Conf. on Machine Learning*, pages 231–238. Morgan Kaufmann, 2000.
- [25] Pedro Domingos. A unified bias-variance decomposition for zero-one and squared loss. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 564–569. AAAI Press, 2000.
- [26] Kaibo Duan, S. Sathiya Keerthi, Wei Chu, Shirish Krishnaji Shevade, and Aun Neow Poo. *Multi-category Classification by Soft-Max Combination of Binary Classifiers*, pages 125–134. Springer Berlin Heidelberg, 2003.
- [27] Shaker H. El-Sappagh and Samir El-Masri. A distributed clinical decision support system architecture. *Journal of King Saud University - Computer and Information Sciences*, 26(1):69–78, 2014.



- [28] Shaker H. El-Sappagh and Samir El-Masri. A distributed clinical decision support system architecture. *Journal of King Saud University - Computer and Information Sciences*, 26(1):69–78, 2014.
- [29] Jae-Hong Eom, Sung-Chun Kim, and Byoung-Tak Zhang. Aptacdss-e: A classifier ensemble-based clinical decision support system for cardiovascular disease level prediction. *Expert Systems with Applications*, 34(4):2465–2479, 2008. ISSN 0957-4174.
- [30] I Esposito De Falco and G De Pietro. An evolutionary-fuzzy dss for assessing health status in multiple sclerosis disease. *Int J Med Inform*, 80(12):e245–54, 2011.
- [31] R. Fantacci. *Reti di Telecomunicazioni: Fondamenti e Tecnologie Internet*. 2014. ISBN 9788874887712.
- [32] Iram Fatima, Muhammad Fahim, Donghai Guan, Young-Koo Lee, and Sungyoung Lee. Socially interactive cdss for u-life care. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication, ICUIMC '11*, pages 1–8, New York, NY, USA, 2011. ACM.
- [33] Antonio Foglio. *Decisioni di marketing e problem solving*. Franco Angeli, 2015.
- [34] Barnett G, Cimino JJ, Hupp JA, and Hoffer EP. Dxpain: An evolving diagnostic decision-support system. *JAMA*, 258(1):67–74, 1987.
- [35] RM Gardner, RO Crapo, AH Morris, and ML Beus. Computerized decision-making in the pulmonary function laboratory. *Respiratory care*, 27(7):799–816, July 1982. ISSN 0020-1324.
- [36] David Garlan. Software architecture: A roadmap. In *Proceedings of the Conference on The Future of Software Engineering, ICSE '00*, pages 91–101, New York, NY, USA, 2000. ACM.

- [37] G. Giacinto and F. Roli. Methods for dynamic classifier selection. In *Proceedings 10th International Conference on Image Analysis and Processing*, pages 659–664, 1999.
- [38] Giorgio Giacinto, Fabio Roli, and Giorgio Fumera. *Selection of Classifiers Based on Multiple Classifier Behaviour*, pages 87–93. Springer Berlin Heidelberg, 2000.
- [39] HJ Gmez-Vallejo, B Uriel-Latorre, M Sande-Meijide, B Villamarn-Bello, Reyes Pavon, F Fdez-Riverola, and Daniel Glez-Pena. A case-based reasoning system for aiding detection and classification of nosocomial infections. *Decision Support Systems*, 84:104–116, 2016.
- [40] Mary K Goldstein, Brian B Hoffman, Robert W Coleman, Mark A Musen, Samson W Tu, A Advani, R Shankar, and M O’connor. Implementing clinical practice guidelines while taking account of changing evidence: Athena dss, an easily modifiable decision-support system for managing hypertension in primary care. In *Proceedings of the AMIA Symposium*, page 300. American Medical Informatics Association, 2000.
- [41] Jiawei Han, Jian Pei, and Micheline Kamber. *Data Mining Concepts and Techniques*. 2006. ISBN 139781558609013.
- [42] L. K. Hansen and P. Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10): 993–1001, 1990.
- [43] C. Hou, Y. Xia, Z. Xu, and J. Sun. Learning classifier competence based on graph for dynamic classifier selection. In *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pages 1164–1168, 2016.
- [44] Maqbool Hussain, Wajahat Ali Khan, Muhammad Afzal, and Sungyong Lee. *Smart CDSS for Smart Homes*, pages 266–269. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

- [45] Bury J, Hurt C, Roy A, Bradburn M, Cross S, Fox J, and Saha V. A quantitative and qualitative evaluation of lisa, a decision support system for chemotherapy dosing in childhood acute lymphoblastic leukaemia. *Stud Health Technol Inform*, 107:197–201, 2004.
- [46] Alceu S. Britto Jr., Robert Sabourin, and Luiz E.S. Oliveira. Dynamic selection of classifiers- a comprehensive review. *Pattern Recognition*, 47(11):3665–3680, 2014.
- [47] Kensaku Kawamoto and David F Lobach. Design, implementation, use, and preliminary evaluation of SEBASTIAN, a standards-based web service for clinical decision support. *AMIA. Annual Symposium proceedings*, pages 380–384, 2005.
- [48] Albert H.R. Ko, Robert Sabourin, Alceu Souza Britto, and Jr. From dynamic classifier selection to dynamic ensemble selection. *Pattern Recognition*, 41(5):1718–1731, 2008.
- [49] R. Kohavi and J.R. Quinlan. Decision tree discovery. In W. Klogen and J.M. Zytkow, editors, *Handbook of Data Mining and Knowledge Discovery*, pages 267–276. Oxford University Press, New York, 2002.
- [50] Igor Kononenko and Matjaž Kukar. *Machine learning and data mining: introduction to principles and algorithms*. Horwood Publishing, 2007.
- [51] L. I. Kuncheva. Clustering-and-selection model for classifier combination. In *Knowledge-Based Intelligent Engineering Systems and Allied Technologies, 2000. Proceedings. Fourth International Conference on*, volume 1, pages 185–188, 2000.
- [52] Ludmila I. Kuncheva and Christopher J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207, 2003.
- [53] Louisa Lam. *Classifier Combinations: Implementations and Theoretical Issues*, pages 77–86. Springer, Berlin, Heidelberg, 2000.

- [54] S. Ledley, Lee B. Lusted, and Robert S. Ledley. Reasoning foundations of medical diagnosis. *Science*, pages 9–21, 1959.
- [55] Chao Li, Shuheng Zhang, Huan Zhang, Lifang Pang, Kinman Lam, Chun Hui, and Su Zhang. Using the k-nearest neighbor algorithm for the classification of lymph node metastasis in gastric cancer. *Computational and mathematical methods in medicine*, 2012, 2012.
- [56] RS Lin, Horn SD, Hurdle JF, and Goldfarb-Rumyantzev AS. Single and multiple time-point prediction models in kidney transplant outcomes. *J Biomed Inform*, 41(6):944–52, 2008.
- [57] Salvador Rodriguez Loya, Kensaku Kawamoto, Chris Chatwin, and Vojtech Huser. Service oriented architecture for clinical decision support: a systematic review and future directions. *Journal of medical systems*, 38(12):140, 2014.
- [58] Alicia Martinez-Garcia, Alberto Moreno-Conde, Francisco Jodar-Sanchez, Sandra Leal, and Carlos Parra. Sharing clinical decisions for multimorbidity case management using social network and open-source tools. *Journal of Biomedical Informatics*, 46(6):977 – 984, 2013.
- [59] Iñigo Mendiáldua, Andoni Arruti, Ekaitz Jauregi, Elena Lazkano, and Basilio Sierra. Classifier subset selection to construct multi-classifiers by means of estimation of distribution algorithms. *Neurocomputing*, 157: 46–60, 2015.
- [60] Yuan MJ, Finley GM, Long J, Mills C, and Johnson RK. Evaluation of user interface and workflow design of a bedside nursing clinical decision support system. *Interact J Med Res.*, 2(1):00, 2013.
- [61] J. D. Myers. The background of internist i and qmr. In *Proceedings of ACM Conference on History of Medical Informatics*, HMI '87, pages 195–197, New York, NY, USA, 1987. ACM. ISBN 0-89791-248-9.

- [62] Cristianini N and Shawe-Taylor J. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.
- [63] A. Nabiha and F. Nadir. New dynamic ensemble of classifiers selection approach based on confusion matrix for arabic handwritten recognition. In *2012 International Conference on Multimedia Computing and Systems*, pages 308–313, 2012.
- [64] Hung T. Nguyen and Elbert A. Walker. *A First Course in Fuzzy Logic, Third Edition*. Chapman & Hall/CRC, 2005. ISBN 1584885262.
- [65] Picton P. *Neural Networks*. Palgrave Macmillan, 2000. ISBN 9780333802878.
- [66] Shadab Adam Pattekari and Asma Parveen. Prediction system for heart disease using naïve bayes. *International Journal of Advanced Computer and Mathematical Sciences*, 3(3):290–294, 2012.
- [67] Romesh Ranawana and Vasile Palade. Multi-classifier systems: Review and a roadmap for developers. *Int. J. Hybrid Intell. Syst.*, 3(1):35–61, January 2006.
- [68] David Riano, Francis Real, Joan Albert Lopez-Vallverdu, Fabio Campana, Sara Ercolani, Patrizia Mecocci, Roberta Annicchiarico, and Carlo Caltagirone. An ontology-based personalization of health-care knowledge to support clinical decisions for chronically ill patients. *Journal of Biomedical Informatics*, 45(3):429–446, 2012.
- [69] Angelo Kinicki Robert Kreitner. *Comportamento organizzativo. Dalla teoria all'esperienza*. Apogeo, 2008.
- [70] Evans RS, Pestotnik SL, Classen DC, Clemmer TP, Weaver LK, J.F. Orme JF, and et al. A computer-assisted management program for antibiotics and other antiinfective agents. *N Eng J Med*, 338:232–8, 1998.

- [71] David E. Rumelhart, James L. McClelland, and CORPORATE PDP Research Group, editors. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. MIT Press, Cambridge, MA, USA, 1986. ISBN 0-262-68053-X.
- [72] Gouda I. Salama, M. B. Abdelhalim, and Magdy Abd elghany Zeid. Breast cancer diagnosis on three different datasets using multi-classifiers. Technical report, J. of Comput. and Inform. Technology, 2012.
- [73] Eulanda M. Dos Santos, Robert Sabourin, and Patrick Maupin. A dynamic overproduce-and-choose strategy for the selection of classifier ensembles. *Pattern Recognition*, 41(10):2993–3009, 2008. ISSN 0031-3203.
- [74] Thomas G Savel, Brian A Lee, Greg Ledbetter, Sara Brown, Dale La-Valley, Julie Taylor, and Pam Thompson. Ptt advisor: A cdc-supported initiative to develop a mobile clinical laboratory decision support application for the ios platform. *Online journal of public health informatics*, 5(2):215, 2013.
- [75] Mary Shaw and David Garlan. *Software Architecture: Perspectives on an Emerging Discipline*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1996. ISBN 0-13-182957-2.
- [76] EH Shortliffe, R Davis, SG Axline, BG Buchanan, CC Green, and SN Cohen. Computer-based consultations in clinical therapeutics: explanation and rule acquisition capabilities of the mycin system. *Computers and biomedical research*, 8(4):303–320, 1975.
- [77] Herbert Simon. *Administrative behavior; a study of decision-making processes in administrative organization*. New York, 1947.
- [78] Dean F. Sittig, Nathan L. Pace, Reed M. Gardner, Eduardo Beck, and Alan H. Morris. Implementation of a computerized patient advice system

- using the help clinical information system. *Computers and Biomedical Research*, 22(5):474–487, 1989.
- [79] Shiliang Sun and Rongqing Huang. An adaptive k-nearest neighbor algorithm. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on*, volume 1, pages 91–94. IEEE, 2010.
- [80] D. M. J. Tax and R. P. W. Duin. Using two-class classifiers for multiclass classification. In *Object recognition supported by user interaction for service robots*, volume 2, pages 124–127 vol.2, 2002.
- [81] Samson W. Tu, James R. Campbell, Julie Glasgow, Mark A. Nyman, Robert McClure, James McClay, Craig Parker, Karen M. Hrabak, David Berg, Tony Weida, James G. Mansfield, Mark A. Musen, and Robert M. Abarbanel. The SAGE guideline model: Achievements and overview. *Journal of the American Medical Informatics Association*, 14(5):589–598, 9 2007. ISSN 1067-5027.
- [82] Cristina Tural, Lidia Ruiz, Christopher Holtzer, Jonathan Schapiro, Pompeyo Viciano, Juan González, Pere Domingo, Charles Boucher, C Rey-Joly, Bonaventura Clotet, et al. Clinical utility of hiv-1 genotyping and expert advice: the havana trial. *Aids*, 16(2):209–218, 2002.
- [83] Gert Van Valkenhoef, Tommi Tervonen, Tijs Zwinkels, Bert De Brock, and Hans Hillege. Addis: a decision support system for evidence-based medicine. *Decision Support Systems*, 55(2):459–475, 2013.
- [84] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.
- [85] L. M. Vriesmann, A. S. Britto, L. S. Oliveira, A. L. Koerich, and R. Saborin. Combining overall and local class accuracies in an oracle-based

- method for dynamic ensemble selection. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7, 2015.
- [86] Jigang Wang, Predrag Neskovic, and Leon N Cooper. Improving nearest neighbor rule with a simple adaptive distance measure. *Pattern Recognition Letters*, 28(2):207–213, 2007.
- [87] Ian Witten, Eibe Frank, and Mark Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. 2011. ISBN 9780080890364.
- [88] David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.
- [89] K. Woods, W. P. Kegelmeyer, and K. Bowyer. Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):405–410, 1997.
- [90] Michal Wozniak, Manuel Grana, and Emilio Corchado. A survey of multiple classifier systems as hybrid systems. *Information Fusion*, 16: 3–17, 2014. ISSN 1566-2535. Special Issue on Information Fusion in Hybrid Intelligent Fusion Systems.
- [91] L. Xu, A. Krzyzak, and C. Y. Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(3):418–435, 1992.
- [92] W You, A Simalatsar, N Widmer, and G De Micheli. A drug administration decision support system. *IEEE International Conference on Bioinformatics and Biomedicine Workshops*, pages 122–129, 2012.
- [93] Shifeng Zhang and Steve Goddard. A software architecture and framework for web-based distributed decision support systems. *Decision Support Systems*, 43(4):1133–1150, 2007.



