



**Politecnico
di Torino**

Politecnico di Torino

Corso di Laurea Magistrale in Ingegneria Biomedica

A.a. 2025/2026

Sessione di laurea Luglio 2026

Classificazione basata su machine learning del DLBCL nei sottotipi ABC e GCB a partire da immagini istologiche

Relatore
Prof. Massimo Salvi

Candidato
Eleonora Catania Cecere

Correlatore
Prof. Nicola Michielli

Abstract

Il linfoma diffuso a grandi cellule B (DLBCL) è il più frequente tra i linfomi non-Hodgkin e si distingue nei sottotipi germinal center B-cell-like (GCB), caratterizzato da un profilo di espressione genica tipico delle cellule B del centro germinativo, e activated B-cell-like (ABC), con un profilo simile a quello delle cellule B post-germinative o attivate. Il sottotipo ABC presenta una prognosi peggiore, con una sopravvivenza globale a cinque anni del 45% rispetto all'80% del sottotipo GCB. Poiché esistono terapie specifiche in grado di migliorare la prognosi del sottotipo ABC, la classificazione del DLBCL risulta fondamentale. Questa tesi ha come obiettivo lo sviluppo di due classificatori basati su machine learning per la distinzione dei sottotipi ABC e GCB a partire da immagini istologiche digitalizzate: uno allenato con label ottenute tramite il metodo immunoistochimico di Hans e uno con label ottenute mediante NanoString. Il dataset include 74 pazienti con DLBCL, ciascuno con un numero variabile di region of interest (ROI). Sono stati innanzitutto segmentati i nuclei delle immagini tramite una rete neurale convoluzionale, ottenendo le maschere nucleari, le quali sono state preprocessate per consentire l'estrazione di feature a livello nucleare, di ROI e di paziente. Tra le feature nucleari sono state considerate misure di densità ottenute dalle distanze tra nuclei e feature descrittive delle relazioni spaziali tra nuclei, ricavate dal minimum spanning tree (MST), implementato in una versione interamente connessa e in una versione tagliata, ottenuta rimuovendo gli archi superiori a 35 pixel. L'MST tagliato è stato utilizzato anche per ricavare feature intra-cluster ed inter-cluster a livello di ROI. A livello di ROI sono stati inoltre estratti l'indice di Morisita e il coefficiente di correlazione di Pearson come misure di co-localizzazione di cellule tumorali e linfociti in sub-ROI di una stessa ROI; tali indici sono stati calcolati anche considerando tutte le ROI di uno stesso paziente, producendo feature a livello di paziente. Le feature nucleari e di ROI sono state aggregate a livello paziente tramite statistiche descrittive (media, deviazione standard e percentili), mentre per le feature nucleari sono stati applicati anche i metodi cell feature level co-occurrence matrix (CFLCM) e bag of words (BoW), basati rispettivamente sulla co-occorrenza dei livelli di feature tra nuclei vicini e sulla distribuzione di statistiche calcolate su sub-ROI all'interno di un dizionario visivo. Le fasi di feature selection e classificazione sono state eseguite separatamente per le due label. Dopo una riduzione della dimensionalità mediante rimozione di feature altamente correlate, sono state selezionate, tramite random forest (RF), le 10 feature con i valori SHAP più elevati e le 10 con il miglior GINI index. Le 20 feature selezionate per ciascuna label sono state utilizzate per allenare il relativo classificatore RF mediante 5-fold cross-validation. Il classificatore allenato con label Hans ha ottenuto un'accuratezza media del 95% nei set di training e validation e del 72% nel test set, mentre quello basato su label NanoString ha riportato un'accuratezza media del 96% nei set di training e validation e del 77% nel test set.

Indice

Elenco delle figure	IV
Elenco delle tabelle	VI
1 Introduzione	1
1.1 Neoplasie ematolinfoidi	1
1.2 Diffuse large B-cell lymphoma	3
1.3 Metodi per la classificazione dei sottotipi ABC e GCB	4
1.4 Obiettivi	5
1.5 Struttura della tesi	6
2 Materiali e metodi	7
2.1 Descrizione del dataset	7
2.2 Descrizione dell'ambiente virtuale	9
2.3 Preprocessing	10
2.4 Feature extraction	11
2.4.1 Feature a livello nucleare	11
2.4.2 Feature a livello di ROI	17
2.4.3 Feature a livello di paziente	23
2.5 Feature aggregation	24
2.5.1 Cell Feature Level Co-occurrence Matrix (CFLCM)	25
2.5.2 Bag of Words (BoW)	30
2.6 Feature selection	33
2.7 Classification	39
2.7.1 Costruzione dei fold	39
2.7.2 Nested cross-validation	41
2.7.3 Random Forest e ottimizzazione degli iperparametri	42
3 Risultati	45
3.1 Dataset e distribuzione delle classi	45
3.2 Metriche di valutazione	46

3.3	Risultati del classificatore Hans	46
3.3.1	Prestazioni per partizione di test	46
3.3.2	Confusion matrix aggregata	49
3.3.3	Curve ROC	49
3.4	Risultati del classificatore Nanostring	50
3.4.1	Prestazioni per partizione di test	50
3.4.2	Confusion matrix aggregata	53
3.4.3	Curve ROC	54
3.5	Sintesi dei risultati	54
4	Discussione e conclusioni	56
4.1	Confronto tra i due classificatori	56
4.1.1	Andamento delle metriche per partizione	58
4.1.2	Distribuzione delle probabilità predette	58
4.2	Interpretazione in relazione al sistema di etichettatura	59
4.3	Confronto con la letteratura	60
4.4	Considerazioni metodologiche: handcrafted features vs. deep learning	62
4.5	Considerazioni sui metodi di aggregazione delle feature	62
4.6	Conclusioni	65
5	Limitazioni e sviluppi futuri	67
5.1	Limitazioni	67
5.1.1	Sbilanciamento del dataset	67
5.1.2	Limitazioni legate ai metodi di labelling	67
5.1.3	Limitazioni della rete di segmentazione nucleare	68
5.1.4	Costruzione del vocabolario visivo nel metodo BoW	68
5.1.5	Feature selection eseguita sull'intero dataset	69
5.2	Sviluppi futuri	69
	Bibliografia	73

Elenco delle figure

1.1	Pipeline dello sviluppo dei sistemi di classificazione. Le fasi in azzurro sono comuni a entrambe le label; le fasi in arancione sono eseguite separatamente per Hans e NanoString.	6
2.1	ROI numero 6 del paziente 22_L_3305_A1A_EE. Come tutte le ROI del dataset, l'immagine è colorata con H&E, che colora i nuclei cellulari in blu-viola e il citoplasma e la matrice extracellulare in rosa	7
2.2	Maschera nucleare della ROI 6 del paziente 22_L_3305_A1A_EE prima (a) e dopo (b) la rimozione dei nuclei con centroidi nella fascia perimetrale di 10 μm (40 pixel).	10
2.3	Confronto tra l'MST full (a sinistra) e l'MST cut (a destra)	13
2.4	Rappresentazione schematica delle feature di densità nucleare . (a) neighbors_mu : vengono conteggiati i nuclei pieni che ricadono entro il cerchio di raggio $\mu = 100$ pixel; i nuclei con bordo grigio sono esclusi dal conteggio. (b) distance_k : le linee tratteggiate collegano il nucleo centrale ai due vicini più prossimi; la linea continua indica la distanza dal k -esimo vicino ($k = 3$), posizionato sulla circonferenza, il cui raggio corrisponde al valore della feature	16
2.5	Schema dei metodi di aggregazione applicati alle feature a livello di nucleo e a livello di ROI, per ottenere feature a livello di paziente.	25
2.6	Schema della costruzione della matrice CFLCM.	27
2.7	Schema della nested 5-fold cross-validation	42
3.1	Metriche di classificazione del classificatore Hans (media su 5 fold). Le barre d'errore rappresentano l'intervallo di confidenza al 95% ($95\% \text{ CI} = 1.96 \times \text{SD}/\sqrt{5}$).	48
3.2	Matrice di confusione aggregata su tutte le partizioni di test del classificatore Hans.	49
3.3	Curve ROC per ciascuna partizione di test del classificatore Hans, con curva media (linea continua) e banda ± 1 SD (area ombreggiata). La linea tratteggiata rappresenta il classificatore casuale ($\text{AUC} = 0.5$).	50

3.4	Metriche di classificazione del classificatore Nanostring (media su 5 fold). Le barre d'errore rappresentano l'intervallo di confidenza al 95% ($95\% \text{ CI} = 1.96 \times \text{SD}/\sqrt{5}$).	52
3.5	Matrice di confusione aggregata su tutte le partizioni di test del classificatore NanoString.	53
3.6	Curve ROC per ciascuna partizione di test del classificatore Nanostring, con curva media (linea continua) e banda $\pm 1 \text{ SD}$ (area ombreggiata). La linea tratteggiata rappresenta il classificatore casuale ($\text{AUC} = 0.5$).	54
4.1	Confronto delle metriche di classificazione tra il classificatore Hans (blu) e il classificatore Nanostring (rosso). Le barre rappresentano la media su 5 fold; le barre d'errore rappresentano il 95% CI ($1.96 \times \text{SD}/\sqrt{5}$).	57
4.2	Andamento delle metriche di classificazione per ciascuna partizione di test (sinistra: classificatore Hans; destra: classificatore Nanostring).	58
4.3	Distribuzione delle probabilità predette di appartenenza alla classe ABC, separata per classe reale (verde: GCB; blu/rosso: ABC), per il classificatore Hans (sinistra) e Nanostring (destra). La linea tratteggiata verticale indica la soglia decisionale a 0.5.	59

Elenco delle tabelle

2.1	Numero di ROI disponibili per ciascun paziente nel dataset iniziale (75 pazienti). Il paziente 20_L_3267_EE, con una sola ROI, è stato escluso dall'analisi.	8
2.2	Feature estratte a livello di singolo nucleo dalla struttura dell'MST. Le feature sono calcolate su entrambe le varianti del grafo e distinte dal suffisso <i>_full</i> o <i>_cut</i> nel nome della feature. Nella colonna <i>Degree min.</i> è indicato il numero minimo di archi incidenti sul nodo necessario per il calcolo; in caso contrario la feature viene posta a zero.	15
2.3	Feature di densità nucleare estratte a livello di singolo nucleo tramite k-d tree. I parametri $\mu = 100$ pixel e $k = 3$ sono stati selezionati tramite tuning sulla correlazione di Pearson con le label.	17
2.4	Feature a livello di ROI estratte dall'MST cut. Le feature calcolate a livello di singolo cluster sono state aggregate a livello di ROI tramite somma (total), media (average), minimo (minimum) o massimo (maximum). Tutte le distanze sono espresse in μm	19
2.5	Feature di colocalizzazione spaziale a livello di ROI, calcolate per le tre combinazioni di parametri selezionate. La soglia di area (thr) distingue nuclei DLBCL da linfociti; il lato della finestra (w) definisce la dimensione della griglia di suddivisione della ROI.	23
2.6	Suffissi delle combinazioni CFLCM applicati alle feature nucleari, ottenuti dal prodotto tra le 3 definizioni di vicinato e le 6 metriche di Haralick. Il nome della feature aggregata è ottenuto concatenando il nome della feature di base con il suffisso corrispondente, separati da un trattino basso.	30
2.7	Feature selezionate per il classificatore Hans, con valori di Gini importance, SHAP importance e correlazione di Pearson con la label. Ord.: ordine di selezione.	34
2.8	Feature selezionate per il classificatore Nanostring, con valori di Gini importance, SHAP importance e correlazione di Pearson con la label. Ord.: ordine di selezione.	35

2.9	Feature selezionate per il classificatore Hans dopo l'applicazione del filtro di correlazione, con valori di Gini importance, SHAP importance e correlazione di Pearson con la label. Ord.: ordine di selezione.	37
2.10	Feature selezionate per il classificatore Nanostring dopo l'applicazione del filtro di correlazione, con valori di Gini importance, SHAP importance e correlazione di Pearson con la label. Ord.: ordine di selezione.	38
2.11	Fold 1 (14 paz., 530 ROI). Hans: 9 NON-GC, 5 GC. Nanostring: 6 ABC, 8 Non-ABC. Coerenti: 11; non coerenti: 3.	40
2.12	Fold 2 (16 paz., 520 ROI). Hans: 11 NON-GC, 5 GC. Nanostring: 8 ABC, 8 Non-ABC. Coerenti: 13; non coerenti: 3.	40
2.13	Fold 3 (15 paz., 495 ROI). Hans: 10 NON-GC, 5 GC. Nanostring: 7 ABC, 8 Non-ABC. Coerenti: 12; non coerenti: 3.	40
2.14	Fold 4 (15 paz., 530 ROI). Hans: 10 NON-GC, 4 GC. Nanostring: 8 ABC, 7 Non-ABC. Coerenti: 12; non coerenti: 3.	40
2.15	Fold 5 (14 paz., 530 ROI). Hans: 10 NON-GC, 4 GC. Nanostring: 7 ABC, 7 Non-ABC. Coerenti: 11; non coerenti: 3.	41
3.1	Distribuzione dei campioni nelle partizioni di test per i due classificatori. I valori indicano il numero totale di campioni nella partizione, con la ripartizione tra classe ABC e classe GCB.	46
3.2	Risultati della nested cross-validation per il classificatore Hans. Per ogni fold esterno sono riportati gli iperparametri selezionati e le accuracy nelle tre fasi della procedura.	47
3.3	Prestazioni del classificatore Hans per ciascuna partizione di test. TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative.	47
3.4	Risultati della nested cross-validation per il classificatore Nanostring.	51
3.5	Prestazioni del classificatore Nanostring per ciascuna partizione di test. TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative.	51
4.1	Confronto delle metriche di classificazione (media $\pm$ SD su 5 fold) tra il classificatore Hans e il classificatore Nanostring. In grassetto il valore migliore per ciascuna metrica.	56
4.2	Feature selezionate per il classificatore Hans, in ordine di selezione. In grassetto le feature a livello di singolo nucleo.	63
4.3	Feature selezionate per il classificatore Nanostring, in ordine di selezione. In grassetto le feature a livello di singolo nucleo.	64

Capitolo 1

Introduzione

1.1 Neoplasie ematolinfoidi

Le neoplasie ematolinfoidi comprendono un ampio gruppo di tumori originati dalle cellule ematopoietiche e vengono oggi classificate in modo gerarchico sulla base della linea cellulare di origine, dello stadio di maturazione, delle caratteristiche morfologiche, immunofenotipiche e genetiche. In questo schema, la distinzione primaria è tra neoplasie mieloidi e linfoidi, come riportato sia nella World Health Organization 5th edition Classification che nell' International Consensus Classification (ICC) [1], [2] .

Le neoplasie mieloidi originano da proliferazioni clonali di cellule staminali ematopoietiche del midollo osseo, che possono manifestarsi con un aumento del numero delle piastrine, dei globuli rossi o dei globuli bianchi ed eventualmente con fibrosi del midollo osseo ed emopoiesi extramidollare [3].

Le neoplasie linfoidi, invece, si verificano quando i linfociti si differenziano in modo neoplastico, diventando cancerosi e accumulandosi in un linfonodo o diffondendosi nel midollo osseo, nel sangue, nella milza o in altri organi [4].

La diagnosi di linfoma prevede una valutazione morfologica, esami immunoistochimici e di citometria di flusso e, quando necessario, esami molecolari. L'agoaspirato non è adeguato come esame biotico, poichè non fornisce sufficiente tessuto, è più appropriata invece una biopsia incisionale o escissionale, o, in alternativa, una biopsia con ago tranciante [5].

L'anamnesi del paziente include anche età, sesso, storia di precedenti neoplasie maligne, presenza di febbre e sudorazioni notturne, perdita ingiustificabile di peso, valutazione delle dimensioni di linfonodi accessibili, milza e fegato [5].

Prima della scelta della terapia, è fondamentale effettuare la stadiazione tramite tomografia ad emissione di positroni (PET), in combinazione con la tomografia computerizzata (CT); La classificazione di Lugano distingue i linfomi in stadio

limitato e avanzato: i primi due stadi corrispondono a stadio limitato, mentre gli ultimi due a stadio avanzato; lo stadio I indica il coinvolgimento di una sola stazione linfonodale o di un singolo sito extranodale, lo stadio II il coinvolgimento di due o più stazioni linfonodali dallo stesso lato del diaframma, eventualmente con una limitata estensione ad un sito extranodale contiguo, mentre gli stadi III e IV indicano rispettivamente il coinvolgimento di linfonodi e/o milza su entrambi i lati del diaframma ed il coinvolgimento di siti extranodali come midollo osseo, polmoni o fegato [5].

Le neoplasie linfoidi possono essere ulteriormente suddivise in neoplasie linfoidi derivanti dai linfociti B e neoplasie linfoidi derivanti dai linfociti T o dalle cellule natural killer (NK) [1].

Tra le neoplasie linfoidi, quelle della linea B sono molto varie e vengono suddivise in numerosi sottogruppi, come le neoplasie delle cellule B mature, i linfomi di Hodgkin e le neoplasie plasmacellulari [1].

I linfomi di Hodgkin si distinguono dagli altri linfomi per la presenza delle cellule di Reed-Sternberg, grandi cellule neoplastiche binucleate fondamentali per diagnosi di questo tipo particolare di linfoma; esse sono derivanti da cellule B del centro germinativo, ma con un fenotipo B modificato; l'altro elemento caratteristico è un microambiente composto da linfociti, istiociti, monociti, plasmacellule ed eosinofili fortemente reattivo [3].

A tutte le altre tipologie di linfoma si fa riferimento come linfomi non-Hodgkin, i quali sono molto più frequenti dei linfomi di Hodgkin, rappresentando ogni anno circa il 4% di tutti i nuovi casi di cancro ed il 3% di tutte le morti per cancro negli Stati Uniti [6]. La diagnosi e il trattamento dei linfomi non-Hodgkin sono più complessi anche a causa della loro eterogeneità biologica e clinica [3]. I pazienti con linfomi non-Hodgkin devono essere seguiti regolarmente per monitorare la malattia e aggiustare il trattamento di conseguenza [7]. La manifestazione clinica più frequente dei linfomi non-Hodgkin è l'ingrossamento non doloroso dei linfonodi. Tuttavia, il coinvolgimento extranodale è presente in oltre un terzo dei casi e può interessare vari organi, come lo stomaco, la cute, il sistema nervoso centrale o il testicolo [8].

La classificazione delle neoplasie ematolinfoidi è cambiata nel tempo. Oggi, la diagnosi richiede una combinazione di morfologia, immunoistochimica, studi molecolari e dati clinici. L'obiettivo è quello di definire con maggiore precisione la prognosi e il comportamento biologico della malattia. Questo approccio integrato aiuta a fornire un trattamento più personalizzato e efficace per i pazienti [2].

1.2 Diffuse large B-cell lymphoma

Il linfoma diffuso a grandi cellule B è il sottotipo più comune di linfoma non-Hodgkin, costituendone circa il 30-40% [1]. Il trattamento standard per il linfoma diffuso a grandi cellule B è la poliemoterapia R-CHOP, che ottiene una risposta completa nel 60-70% dei pazienti e una sopravvivenza globale a 5 anni del 60% nei pazienti a basso rischio. Tuttavia, i fattori prognostici negativi, come la presenza di "double-hit" lymphomas o lo status di "double-expressor", possono influire sulla prognosi [9].

A seconda della cell of origin (COO) il DLBCL si suddivide in due sottocategorie: la sottocategoria germinal center B-cell like (GCB) mostra un profilo di espressione genica simile a quello delle cellule B germinali; il sottotipo activated B-cell like (ABC) presenta invece un profilo di espressione genica tipico delle cellule B post-germinative o attivate. Questi due sottotipi si differenziano per le vie biologiche che coinvolgono il ciclo cellulare, la proliferazione, la risposta immunitaria e l'apoptosi [10].

L'ABC DLBCL riporta una prognosi peggiore, uno studio ha riscontrato una progression-free survival (PFS) a 5 anni del 31%, contro una PFS a 5 anni del 76% per il GCB DLBCL ed un'overall survival (OS) a 5 anni del 45%, contro l'80% del GCB DLBCL [11].

Oltre che per prevedere la possibile prognosi, la determinazione del COO è fondamentale anche per la scelta delle terapie; la maggior parte delle nuove terapie in fase di trial sembrano infatti avere effetto in particolare sull'ABC DLBCL, attaccando le vie del B-cell receptor (BCR) o le vie a valle di quest'ultima, tra queste terapie rientrano gli inibitori del proteasoma, gli agenti immunomodulatori e gli inibitori della segnalazione del BCR [12], [13], [14].

Tali terapie inducono tuttavia effetti collaterali più severi rispetto all'R-CHOP. Ad esempio pazienti con linfoma mantellare precedentemente non trattato hanno mostrato più frequentemente effetti collaterali se trattati con bortezomib - un inibitore del proteasoma - in combinazione con rituximab, ciclofosfamide, doxorubicina e prednisone (BzR-CAP) rispetto ai pazienti a cui era stato somministrato l'RCHOP: sono state registrate incidenze più elevate di trombocitopenia di grado ≥ 3 (56.7% vs 5.8%) e di infezione da herpes zoster (6.7% vs 1.2%), inoltre per il 22.5% dei pazienti in cura con BzR-CAP è risultata necessaria una trasfusione di piastrine, contro il 2.9% dei pazienti curati con R-CHOP [15].

La classificazione del sottotipo di DLBCL risulta quindi fondamentale per valutare la somministrazione di terapie aggiuntive all'R-CHOP per i pazienti con sottotipo ABC, a causa della sua prognosi più sfavorevole e della sua maggiore aggressività e per evitare al contrario di somministrare tali terapie aggiuntive ai pazienti con sottotipo GCB, i quali ne beneficerebbero molto meno, a costo di ulteriori effetti collaterali.

1.3 Metodi per la classificazione dei sottotipi ABC e GCB

Il gold standard, nonché il primo metodo utilizzato per distinguere i due principali sottotipi di DLBCL, è quello dei gene-expression profiles (GEP) tramite microarray di cDNA, attraverso cui vengono differenziati i sottotipi ABC, in cui si ha una sostanziale attivazione del gene NF- κ B, il GCB, l'unico in cui sono stati osservati due eventi oncogenici ricorrenti nel DLBCL, t(14;18), e l'amplificazione del locus c-rel sul cromosoma 2p ed il type 3 dei non classificabili [16]. La procedura per eseguire la GEP è tuttavia lunga, complessa e costosa, necessitando inoltre di tessuti freschi o congelati, dunque non è sempre sostenibile includerla nella pratica clinica [17].

A causa di tali complicazioni sono stati sviluppati dei metodi alternativi di classificazione, come gli algoritmi immunoistochimici eseguiti su tissue microarray (TMA). Essi si basano sull'analisi di marker proteici espressi su tessuto fissato in formalina e incluso in paraffina (FFPE) e perciò più semplice da adoperare nella pratica clinica. Tra gli algoritmi immunoistochimici quello di Hans è stato raccomandato nel 2016 nella revisione della classificazione delle neoplasie linfoidi da parte della world health organization (WHO), la quale approva anche l'utilizzo di altri algoritmi [17]. L'algoritmo di Hans utilizza le proteine cluster of differentiation 10 (CD10), B-cell lymphoma 6 (BCL6), Multiple Myeloma Oncogene 1 (MUM1), Forkhead box protein P1 (FOXP1), Cyclin D2 e B-cell lymphoma 2 (BCL2). In particolare i marker proteici del centro germinativo CD10 e BCL6 risultano essere rappresentativi del profilo GCB, mentre il MUM1, indice di uno stadio più avanzato della differenziazione delle cellule B, del profilo non-GCB (ABC); l'algoritmo ottiene inoltre una sensibilità del 71% per il gruppo GCB e dell'88% per il gruppo non-GCB, utilizzando la GEP con cDNA microarray come gold standard, rispetto a cui mostra una migliore capacità predittiva dell'OS a 5 anni [18]. Studi successivi che utilizzano come gold standard la GEP Lymph2Cx, ossia un metodo molecolare più moderno e accurato, riportano tuttavia una concordanza complessiva del 72%, con una concordanza dell'85% per i casi GCB e solo del 58% per i non-GCB [19].

Tra gli algoritmi immunoistochimici, quello di Choi risulta essere il più predittivo per il sottotipo e per l'OS se confrontato con la GEP, ma è il più complesso da utilizzare, poichè sfrutta il Germinal Center B-cell Expressed Transcript 1 (GCET1) ed il Forkhead box P1 (FOXP1), anticorpi che non vengono adoperati in molti laboratori [20].

Successivamente è stato sviluppato un metodo alternativo per effettuare la GEP, ovvero la tecnologia NanoString, la quale è eseguibile su campioni FFPE e si basa sull'analisi di un gruppo ristretto di geni. La piattaforma NanoString consente di leggere digitalmente l'abbondanza trascrizionale, generando un punteggio per la

classificazione del sottotipo. Recentemente è stato anche sviluppato il Lymph2Cx, un assay di GEP per la subclassificazione del DLBCL; sfruttando la tecnologia NanoString, i ricercatori sono partiti da 93 geni indicativi per la distinzione tra ABC e GCB e hanno riscontrato che bastavano 15 geni core più 15 geni housekeeping per ottenere un'accuratezza di classificazione molto simile a quella della GEP completa, con una misclassificazione del 2%, contro un tasso di errore variabile tra il 6% e il 17% a seconda dell'algoritmo immunoistochimico utilizzato (9% per Hans, 6% per Tally, 17% per Choi) [21], [22].

Per la classificazione dei due sottotipi di DLBCL sono stati inoltre sviluppati dei sistemi basati sull'intelligenza artificiale: sono state utilizzate reti neurali artificiali per predire l'OS ed il sottotipo molecolare del DLBCL a partire da 730 geni, in particolare il sottotipo molecolare, predetto sulla base dell'assay Lymph2Cx, ha riportato un'area under curve (AUC) pari a 1 [23]. Un altro esempio di sistema di intelligenza artificiale, partendo da immagini istologiche trattate con ematossilina e eosina (H&E) digitalizzate, estrae feature da esse tramite encoder basati su convolutional neural network (CNN) e transformer preaddestrati e classifica i campioni in ABC o GCB attraverso una variante di multiple instance learning con meccanismo di attenzione, raggiungendo un'AUC media dell'87.4% su tre fold [24].

1.4 Obiettivi

La tesi ha come obiettivo l'implementazione di due classificatori basati su machine learning, per la distinzione dei sottotipi ABC e GCB del DLBCL, a partire da whole slide images (WSI) colorate con ematossilina e eosina (H&E): uno allenato tramite le label GCB e non-GCB, ottenute dal metodo immunoistochimico di Hans e uno addestrato secondo le label ABC e non-ABC, ricavate col metodo NanoString. Tale sistema consentirebbe la classificazione automatizzata dei sottotipi di DLBCL direttamente dalle scannerizzazioni delle sezioni istologiche, senza necessità di analisi molecolari, configurandosi come un sistema di supporto alla decisione per i medici durante il processo di diagnosi. L'utilizzo di metodi di machine learning e l'estrazione di feature handcrafted con intrinseco significato fisico, porterebbe, in caso di applicazione in ambito clinico, ad una maggiore fiducia del personale medico verso i classificatori, rispetto ai classificatori deep learning, i quali estraggono feature più astratte direttamente dai dati grezzi [25].

1.5 Struttura della tesi

Il progetto di tesi ha previsto inizialmente il preprocessing delle maschere nucleari delle immagini istologiche, prodotte dalla rete di segmentazione, al fine di migliorare la successiva estrazione di feature nucleari a livello di singolo nucleo, a livello di ROI e a livello di paziente. Anche le feature a livello nucleare e a livello di ROI sono state in seguito portate a livello di paziente, mediante la fase di aggregazione. La restante parte della pipeline è stata eseguita separatamente per le label Hans e NanoString: per ogni label, sono state rimosse le feature altamente ridondanti, creando un dataset con dimensionalità ridotta, il quale è stato dato in input alla fase di selezione di feature, effettuata tramite RF, considerando i valori del GINI index e dello SHapley additive explanation value (SHAP). Le feature finali selezionate per ciascuna label sono state utilizzate nel relativo classificatore RF, implementato con 5-fold cross-validation, per assegnare a ciascun paziente la classe GCB o non-GCB nel caso di Hans e ABC o non-ABC nel caso di NanoString. In figura 1.1 viene riportato lo schema della pipeline complessiva.

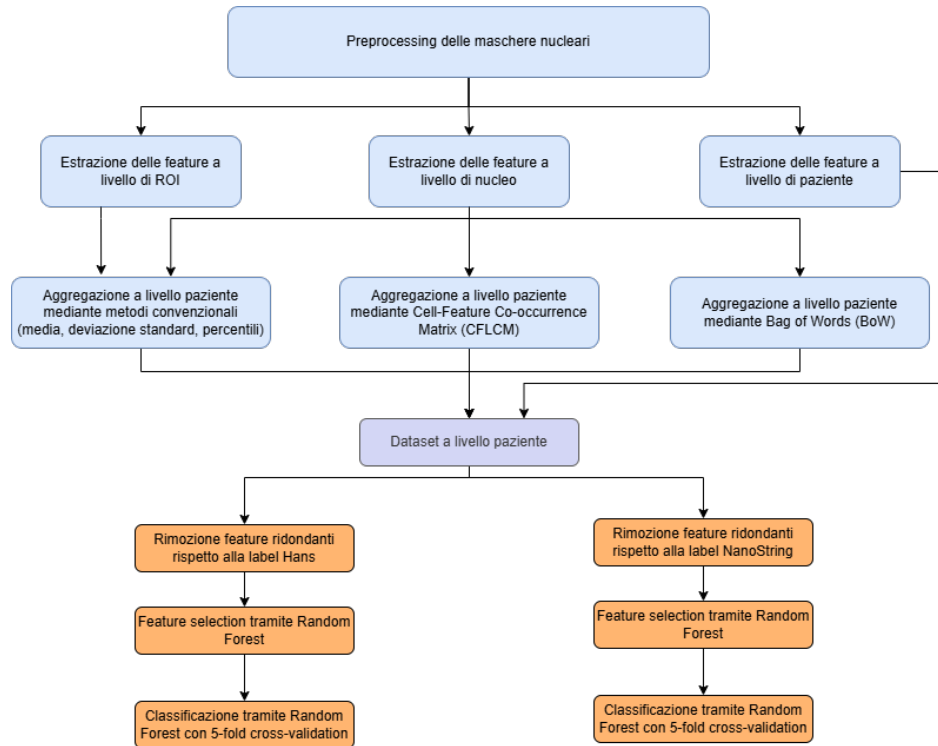


Figura 1.1: Pipeline dello sviluppo dei sistemi di classificazione. Le fasi in azzurro sono comuni a entrambe le label; le fasi in arancione sono eseguite separatamente per Hans e NanoString.

Capitolo 2

Materiali e metodi

2.1 Descrizione del dataset

Il dataset iniziale includeva ROI appartenenti a 75 diversi pazienti con DLBCL, ricavate da whole slide images (WSI) colorate con ematossilina e eosina e digitalizzate con scanner Aperio in formato SVS ad ingrandimento 40x, corrispondente ad una risoluzione di $0.25\mu\text{m}/\text{pixel}$. Il materiale è stato fornito dalla federazione italiana linfomi (FIL), in collaborazione con l'istituto europeo di oncologia (IEO) di Milano. In figura 2.1 viene riportato un esempio di ROI.

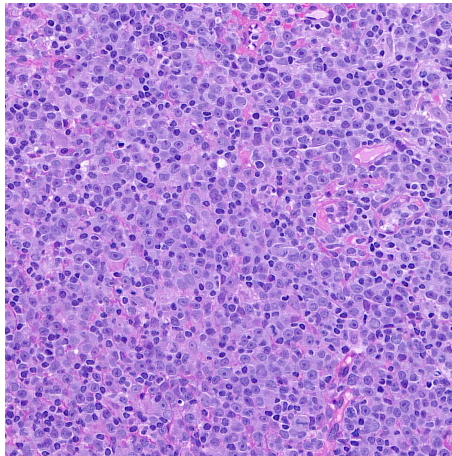


Figura 2.1: ROI numero 6 del paziente 22_L_3305_A1A_EE. Come tutte le ROI del dataset, l'immagine è colorata con H&E, che colora i nuclei cellulari in blu-viola e il citoplasma e la matrice extracellulare in rosa

I pazienti avevano ROI di dimensioni 1500x1500 pixel e in numerosità variabile, indicata nella tabella 2.1.

Tabella 2.1: Numero di ROI disponibili per ciascun paziente nel dataset iniziale (75 pazienti). Il paziente 20_L_3267_EE, con una sola ROI, è stato escluso dall’analisi.

ID paziente	N. ROI	ID paziente	N. ROI
20_L_1001_A1_EE	40	22_L_3461_A1A_EE	40
20_L_1020_A1_EE	40	22_L_3305_A1A_EE	40
20_L_1146_A1_EE	40	22_L_3189_A1A_EE	40
20_L_2078_EE	40	22_L_3188_A1A_EE	40
20_L_2050_EE	40	22_L_317_A1A_EE	40
20_L_2270_A1_EE	40	22_L_2739_A3A_EE	20
20_L_2902_B1_EE	5	22_L_2421_A1E_EE	40
20_L_2702_A1_EE	40	22_L_2271_A1A_EE	40
20_L_3267_EE	1	22_L_1486_A1A_EE	40
20_L_3128_C1_EE	40	22_L_1400_A1A_EE	20
20_L_382_EE	40	22_L_1380_A1A_EE	40
20_L_521_EE	40	22_L_1292_A1A_EE	40
21_L_894_A1_EE	40	22_L_1176_A1A_EE	40
21_L_995_A1_EE	40	23_L_770_A1A_EE	10
21_L_914_A1_EE	40	23_L_533_A1A_EE	40
21_L_695_A1_EE	40	23_L_299_A1A_EE	40
21_L_3295_A1_EE	10	23_L_298_A1A_EE	40
21_L_421_EE	40	23_L_2779_A1G_EE	40
21_L_2589_A1_EE	40	23_L_2716_A1B_EE	10
21_L_2523_A1_EE	10	23_L_2606_A1A_EE	40
21_L_2426_A1_EE	40	23_L_2514_A1B_EE	40
21_L_2213_A1_EE	40	23_L_242_A1A_EE	40
21_L_2195_EE	40	23_L_2288_A1B_EE	40
21_L_2143_A1_EE	40	23_L_2126_A1J_EE	40
21_L_2110_A1_EE	40	23_L_2125_A1B_EE	10
21_L_2094_EE	40	23_L_1937_A1D_EE	40
21_L_2033_A1_EE	30	23_L_1850_A1A_EE	40
21_L_1936_A1_EE	40	23_L_1374_A1L_EE	10
21_L_1931_EE	10	23_L_1289_A1A_EE	40
21_L_1649_A1_EE	40	23_L_1078_A1A_EE	40
21_L_1463_A1_EE	40	24_L_817_A1A_EE	20
21_L_1080_A1_EE	40	24_L_669_B1B_EE	40
22_L_806_A1A_EE	30	24_L_643_A1A_EE	10
22_L_782_A1A_EE	40	24_L_444_A1A_EE	40
22_L_682_A1A_EE	40	24_L_281_A1A_EE	40
22_L_564_A1A_EE	40	24_L_131_EE	40
22_L_563_A1A_EE	40	24_L_130_A3B_EE	40
22_L_3473_A1A_EE	40		

Il paziente con ID 20_L_3267_EE è stato rimosso, poichè una sola ROI non sarebbe stata sufficiente per le analisi successive, che sono state quindi condotte su 74 pazienti.

La segmentazione dei nuclei è stata effettuata mediante una rete di deep learning già sviluppata e addestrata nell'ambito del progetto Dipat3Care, in una fase precedente all'inizio del presente lavoro di tesi. La rete è stata sviluppata utilizzando il framework MMSegmentation e un'architettura basata su backbone Swin Transformer Tiny [10.48550/arXiv.2103.14030], scelto per la sua efficacia nei compiti di segmentazione semantica e per la capacità di modellare in modo efficace sia le informazioni locali sia il contesto globale dell'immagine. Il modello è stato addestrato su patch RGB di dimensione 320×320 pixel, estratte dalle immagini istologiche annotate. Il problema è stato formulato come un task di segmentazione a tre classi, corrispondenti a sfondo, bordo del nucleo e interno del nucleo. La classe bordo è stata introdotta per favorire una migliore separazione tra nuclei adiacenti o parzialmente sovrapposti. La classe background, invece, include tutte le aree non riconducibili alle strutture nucleari annotate.

Durante la fase di addestramento sono state applicate procedure standard di pre-processing e tecniche di data augmentation, tra cui rotazioni, ribaltamenti e leggere variazioni cromatiche, con l'obiettivo di aumentare la robustezza del modello rispetto alla variabilità morfologica e cromatica tipica delle immagini istopatologiche. L'ottimizzazione del modello è stata effettuata mediante l'algoritmo Adam. L'addestramento è stato condotto per un massimo di 100 epoche, adottando una strategia di early stopping per interrompere il training in assenza di ulteriori miglioramenti sulle metriche di validazione. Come funzione obiettivo è stata utilizzata la Dice loss, scelta per la sua efficacia nei problemi di segmentazione e per la capacità di gestire in modo appropriato lo sbilanciamento tra classi.

Al termine della fase di inferenza, le mappe di segmentazione generate dal modello sono state utilizzate per ricostruire le maschere nucleari finali, le quali hanno costituito il dataset di partenza del presente lavoro di tesi.

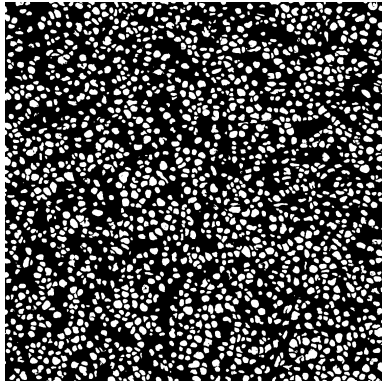
2.2 Descrizione dell'ambiente virtuale

Gli step di preprocessing, feature extraction, feature aggregation, feature selection e classificazione sono stati eseguiti mediante script implementati in linguaggio Python (versione 3.9.18). Tutti i codici sono stati sviluppati ed eseguiti all'interno di un ambiente virtuale dedicato, creato tramite Anaconda, utilizzando l'ambiente di sviluppo integrato (IDE) Spyder (versione 5.1.5).

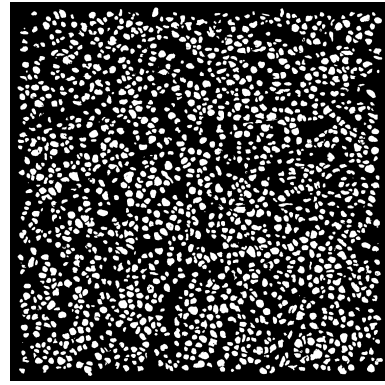
2.3 Preprocessing

Tutte le feature estratte - feature a livello di singolo nucleo, a livello di ROI e a livello di paziente - sono state calcolate a partire dalle maschere nucleari delle ROI. La rete di segmentazione ha tuttavia riconosciuto alcuni nuclei parzialmente sovrapposti o strettamente adiacenti come un unico nucleo, il che avrebbe portato ad un'alterazione delle feature estratte.

Per ridurre tale bias, prima della fase di feature extraction è stato eseguito il preprocessing delle maschere nucleari. Le maschere utilizzate sono le maschere erose prodotte dalla rete di segmentazione, in cui è già stato applicato un kernel di erosione morfologica per favorire la separazione di nuclei adiacenti. Sono stati inoltre rimossi i nuclei localizzati in prossimità dei bordi delle immagini e quindi potenzialmente troncati: in particolare, sono stati eliminati tutti i nuclei il cui centroide si trovava a una distanza inferiore a $10\text{ }\mu\text{m}$ (40 pixel) da uno qualsiasi dei quattro bordi dell'immagine, poiché tali nuclei avrebbero introdotto un bias nelle feature estratte. In Figura 2.2 viene mostrato il risultato di questa operazione.



(a) Maschera originale (1886 nuclei)



(b) Dopo rimozione nuclei ai bordi

Figura 2.2: Maschera nucleare della ROI 6 del paziente 22_L_3305_A1A_EE prima (a) e dopo (b) la rimozione dei nuclei con centroide nella fascia perimetrale di $10\text{ }\mu\text{m}$ (40 pixel).

Poiché l'erosione morfologica comporta una sistematica sottostima dell'area nucleare, il calcolo di quest'ultima è stato eseguito separatamente: ciascun nucleo è stato isolato tramite bounding box e dilatato singolarmente per mezzo di un elemento strutturale circolare di raggio pari a 4 pixel, a seguito di uno zero-padding per evitare artefatti ai bordi dell'oggetto. Per consentire il calcolo delle aree, i nuclei isolati e dilatati sono stati temporaneamente ricomposti in una maschera artificiale (*puzzle mask*), mantenendo una separazione spaziale di un pixel tra oggetti adiacenti al fine di evitarne la fusione durante la procedura di labeling.

Le aree nucleari così ottenute, espresse in μm^2 , sono state utilizzate per escludere gli oggetti con area superiore a $450 \mu\text{m}^2$, valore troppo elevato per essere attribuibile a un singolo nucleo e riconducibile ad agglomerati nucleari erroneamente segmentati come unico oggetto dalla rete. La maschera finale, ottenuta applicando alla maschera dopo crop anche la rimozione dei nuclei sopra soglia, è stata utilizzata per l'estrazione dei centroidi nucleari e per il calcolo di tutte le feature basate sulle relazioni spaziali tra nuclei. Su questa stessa maschera filtrata è stata ripetuta la procedura di isolamento e dilatazione singola dei nuclei, i cui risultati sono stati utilizzati per il calcolo delle feature morfologiche a livello di singolo nucleo.

2.4 Feature extraction

2.4.1 Feature a livello nucleare

Le feature estratte a livello di singolo nucleo includono feature spaziali derivate dal grafo Minimum Spanning Tree (MST) e feature di densità nucleare calcolate tramite ricerca del vicino più prossimo con struttura k-d tree. Per ciascuna ROI, l'MST e la k-d tree sono stati costruiti considerando l'insieme dei centroidi nucleari appartenenti a quella ROI, e le feature sono state calcolate per ogni singolo nucleo in essa contenuto. Al termine dell'elaborazione di tutte le ROI di un paziente, le feature nucleari sono state salvate nel rispettivo file Excel, organizzato in un numero di fogli pari al numero di ROI del paziente, ciascuno contenente le feature dei nuclei della ROI corrispondente.

Feature spaziali estratte dall'MST

L'MST è un grafo che collega dei vertici tramite degli archi pesati, nello specifico l'MST contiene $V-1$ archi, con V il numero di vertici, e collega tutti i vertici senza creare cicli e minimizzando il peso totale degli archi [26].

Gli algoritmi usati per la costruzione dell'MST sono l'algoritmo di Kruskal e quello di Prim; essi sono entrambi algoritmi di tipo greedy, che si basano sull'aggiunta progressiva di archi di peso minimo che collegano componenti ancora separate del grafo, rispettando dunque la cut property, la quale garantisce che l'arco di peso minimo che collega due componenti separate è un arco sicuro e può essere incluso in un MST [26].

Nell'algoritmo di Kruskal inizialmente ogni vertice costituisce una componente connessa separata; successivamente tutti i possibili archi di collegamento tra due nuclei vengono ordinati in ordine crescente di peso. L'algoritmo aggiunge dunque progressivamente gli archi di peso minore, che collegano due componenti distinte, evitando la formazione di cicli. La foresta di vertici separati viene perciò progressivamente trasformata in un unico albero di copertura, finché non si raggiunge un

numero di archi pari a $V-1$ [26].

L'algoritmo di Prim costruisce invece un MST mediante una strategia greedy che espande progressivamente un unico albero. A partire da un vertice iniziale scelto arbitrariamente, a ogni iterazione viene selezionato l'arco di peso minimo che collega un vertice già appartenente all'albero a un vertice non ancora incluso. Il nuovo vertice e il relativo arco vengono quindi aggiunti alla soluzione, estendendo l'albero fino a coprire tutti i vertici del grafo. La correttezza dell'algoritmo deriva dalla Cut Property, secondo la quale l'arco di peso minimo che attraversa il taglio tra i vertici già inclusi e quelli ancora esterni costituisce un safe edge e può essere aggiunto senza compromettere l'ottimalità della soluzione [26].

Storicamente l'MST è stato utilizzato per la formazione di cluster mediante la rimozione degli archi più lunghi [27]. Questo grafo viene inoltre adattato da diversi anni all'istopatologia, considerando i centroidi nucleari come vertici e analizzando così l'organizzazione spaziale delle cellule tumorali [28].

Le feature spaziali a livello nucleare calcolate nel presente lavoro di tesi fanno parte delle feature strutturali locali descritte nell'articolo [29] e sono state applicate a due varianti dell'MST: MST full, ossia l'albero unico formato da $V-1$ archi e che consente di raggiungere a partire da un vertice, qualsiasi altro vertice attraverso un percorso formato da uno o più archi, e l'MST cut, ottenuto rimuovendo gli archi al di sopra di una certa lunghezza dall'MST full.

Per la costruzione di entrambe le varianti di MST, la distanza euclidea tra i centroidi dei nuclei ha rappresentato il peso degli archi; inizialmente per ogni ROI sono state dunque calcolate dalle maschere le coordinate x ed y dei centroidi nucleari tramite la funzione `regionprops` della libreria `python skimage.measure`, successivamente è stata calcolata la matrice delle distanze euclidee tra tutte le coppie di centroidi nucleari della ROI, ricavata mediante le funzioni `pdist` e `squareform` della libreria `python scipy.spatial.distance`, implementando la formula in (2.1).

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.1)$$

La matrice delle distanze euclidee tra i centroidi è stata data in input alla funzione `nx.from_numpy_array` della libreria `networkx`, la quale ha restituito l'albero con tutti i centroidi connessi fra loro, a sua volta passato alla funzione `nx.minimum_spanning_tree` della stessa libreria, che ha fornito l'MST, mantenendo solo gli archi che permettevano di non avere cicli e di minimizzare il peso totale, tramite algoritmo di Kruskal.

I precedenti passaggi hanno permesso di ottenere l'MST full, la costruzione dell'MST cut ha previsto in aggiunta la rimozione degli archi di lunghezza superiore ai 35 pixel, formando dei cluster separati di nuclei; la soglia di 35 pixel è stata scelta osservando il numero di cluster che venivano così creati nelle maschere del

dataset, cercando un compromesso tra la formazione di un numero eccessivamente basso di cluster e quella di numerosi centroidi isolati. In figura 2.3 viene mostrato il confronto tra MST full ed MST cut.

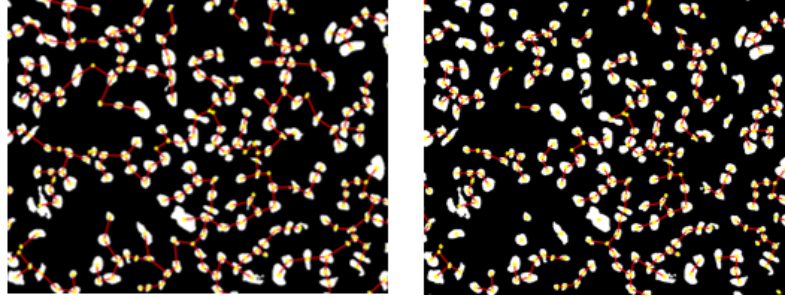


Figura 2.3: Confronto tra l'MST full (a sinistra) e l'MST cut (a destra)

Le feature descritte nell'articolo di Bilgin et al. venivano calcolate sul cell-graph, un grafo costruito collegando due nodi se la loro distanza euclidea è inferiore ad una soglia prefissata e che, a differenza dell'MST, è costituito da diverse componenti separate e può contenere cicli. Tra le feature strutturali locali sono state perciò selezionate esclusivamente quelle adattabili al grafo MST. Dopo la costruzione dell'MST, la funzione `nx.all_pairs_dijkstra_path_length()` è stata utilizzata per calcolare le lunghezze dei cammini minimi tra tutte le coppie di nodi, considerando come peso degli archi la distanza euclidea tra i centroidi. A partire da tali distanze sono state ricavate le feature di eccentricity, definita come il più lungo cammino minimo da un nucleo a qualsiasi altro nucleo raggiungibile nel grafo, e di effective eccentricity 90, ottenuta considerando anziché il massimo, il novantesimo percentile delle distanze, riducendo così l'influenza di eventuali valori estremi. Le misure di centralità closeness e betweenness sono state invece calcolate rispettivamente tramite le funzioni `nx.closeness centrality()` e `nx.betweenness centrality()`. La closeness centrality quantifica quanto un nodo sia mediamente vicino agli altri nodi del grafo e per ogni nodo v è definita, mediante la formula (2.2), come l'inverso della somma di tutti i cammini minimi che partono da v .

$$C_C(v) = \frac{n - 1}{\sum_{u \neq v} d(v, u)} \quad (2.2)$$

Mentre la betweenness centrality misura l'importanza di un nodo come elemento di collegamento all'interno del grafo. Formalmente, essa è definita come la somma, su tutte le possibili coppie di nodi s e t , del rapporto tra il numero di cammini minimi che attraversano il nodo v e il numero totale di cammini minimi che collegano s e t

e la sua formula viene riportata in (2.3)

$$C_B(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (2.3)$$

dove σ_{st} rappresenta il numero totale di cammini minimi tra i nodi s e t , mentre $\sigma_{st}(v)$ indica il numero di tali cammini che attraversano il nodo v . Valori elevati di betweenness identificano nodi che fungono da collegamento tra diverse regioni del grafo e che risultano quindi particolarmente rilevanti per la connettività globale della struttura. Nel caso specifico dell'MST, essendo il grafo un albero, tra ogni coppia di nodi esiste un unico cammino e tale cammino coincide necessariamente con il cammino minimo. Di conseguenza risulta sempre $\sigma_{st} = 1$ e la definizione precedente si semplifica nella formula (2.4)

$$C_B(v) = \sum_{s \neq v \neq t} \sigma_{st}(v) \quad (2.4)$$

Poiché $\sigma_{st}(v)$ può assumere solamente i valori 0 oppure 1, la betweenness centrality corrisponde al numero di coppie di nodi il cui cammino minimo attraversa il nodo v . Nell'ambito dell'analisi dell'MST dei centroidi nucleari, valori elevati di betweenness individuano quindi nuclei che svolgono il ruolo di ponte tra differenti regioni della rete spaziale.

Per ogni nucleo sono inoltre state estratte alcune caratteristiche locali direttamente dalla struttura dell'MST. Il grado (degree) è stato ottenuto tramite `G.degree(i)` e rappresenta il numero di archi incidenti sul nodo, ossia il numero di nuclei direttamente connessi ad esso.

A partire dal grado è stata inoltre definita la feature endpoint, che assume valore 1 quando il nodo possiede un solo vicino e rappresenta quindi una foglia dell'albero, e valore 0 negli altri casi.

Mediante il metodo `G.neighbors(i)` sono stati individuati tutti i nuclei adiacenti al nodo considerato, mentre le lunghezze degli archi incidenti sono state recuperate dagli attributi di peso del grafo. Su tali lunghezze sono state quindi calcolate la media, la deviazione standard, e la skewness utilizzando rispettivamente le funzioni `numpy.mean()`, `numpy.std()`, e `scipy.stats.skew()`. Queste feature descrivono la distribuzione delle distanze tra il nucleo analizzato e i suoi vicini diretti, consentendo di caratterizzare non solo la connettività locale ma anche la regolarità o eterogeneità della disposizione spaziale dei nuclei circostanti. Tutte le distanze sono state convertite da pixel a micrometri mediante il fattore di conversione mpp (microns per pixel), pari a 0.2498. Le feature spaziali a livello nucleare estratte dall'MST sono riassunte nella tabella 2.2

Tabella 2.2: Feature estratte a livello di singolo nucleo dalla struttura dell’MST. Le feature sono calcolate su entrambe le varianti del grafo e distinte dal suffisso `_full` o `_cut` nel nome della feature. Nella colonna *Degree min.* è indicato il numero minimo di archi incidenti sul nodo necessario per il calcolo; in caso contrario la feature viene posta a zero.

Feature	Descrizione	Degree min.
degree	Numero di archi incidenti sul nodo (vicini diretti nell’MST)	—
endpoint	Vale 1 se il nodo è una foglia dell’albero (degree= 1), 0 altrimenti	—
eccentricity	Lunghezza in μm del cammino minimo più lungo verso qualsiasi altro nodo raggiungibile	≥ 1
effective_eccentricity_90	90° percentile in μm delle lunghezze dei cammini minimi verso tutti gli altri nodi	≥ 1
closeness	Closeness centrality: inverso della somma dei cammini minimi dal nodo verso tutti gli altri	≥ 1
betweenness	Betweenness centrality: numero di coppie di nodi il cui unico cammino minimo attraversa il nodo	≥ 2
mean_edge_length	Media in μm delle lunghezze degli archi incidenti sul nodo	≥ 1
std_edge_length	Deviazione standard in μm delle lunghezze degli archi incidenti	≥ 2
skew_edge_length	Asimmetria della distribuzione delle lunghezze degli archi incidenti	≥ 3

Feature di densità nucleare

Come feature nucleari sono state estratte anche le feature di densità nucleare `density neighbors` e `density distance` descritte in [30], che a sua volta fa riferimento a [31]. La feature `density neighbors` è definita per ogni nucleo come il numero di centroidi nucleari che ricadono in un cerchio di raggio μ centrato nel centroide di quel nucleo, viceversa la feature `density distance` è definita per ogni nucleo come il raggio che deve avere un cerchio centrato nel suo centroide, affinché al suo interno ricadano k altri centroidi. In [31] la feature `density neighbors` è stata calcolata per μ pari a 10, 20, 30, 40 e 50 pixel, mentre la feature `density distance` per valori di k pari a 3, 5 e 7.

Nel presente lavoro di tesi le due tipologie di feature di densità nucleare sono state calcolate costruendo un `k-d tree` a partire dai centroidi nucleari di ciascuna ROI

tramite la classe `cKDTree` della libreria SciPy; un k-d tree organizza gli elementi nello spazio bidimensionale (in questo caso i centroidi nucleari) mediante una suddivisione ricorsiva basata sulle coordinate spaziali, consentendo di escludere a priori alcuni elementi dalla ricerca di vicinato e rendendo dunque quest'ultima più rapida e meno onerosa computazionalmente. La feature density neighbors è stata denominata **neighbors_mu** ed è stata ottenuta tramite la funzione `tree.query_ball_point()`, che restituisce l'insieme dei punti contenuti all'interno di un cerchio di raggio μ centrato sul nucleo considerato; il numero di vicini viene quindi calcolato come la cardinalità dell'insieme restituito, sottraendo un'unità per escludere il nucleo stesso. La seconda feature, denominata **distance_k**, è stata invece calcolata come la distanza tra il centroide del nucleo centrale e quello del k-esimo vicino più prossimo. Per ciascun centroide viene eseguita una ricerca dei k+1 vicini mediante la funzione `tree.query()`, che include il punto stesso, il quale compare sempre alla distanza nulla; poiché le distanze vengono restituite in ordine crescente, l'ultima distanza dell'array corrisponde alla distanza dal k-esimo centroide più vicino. Tale valore viene successivamente convertito da pixel a micrometri mediante il fattore di scala mpp (microns per pixel), pari a 0.2498.

In Figura 2.4 è riportata una rappresentazione schematica del calcolo delle due feature di densità nucleare.

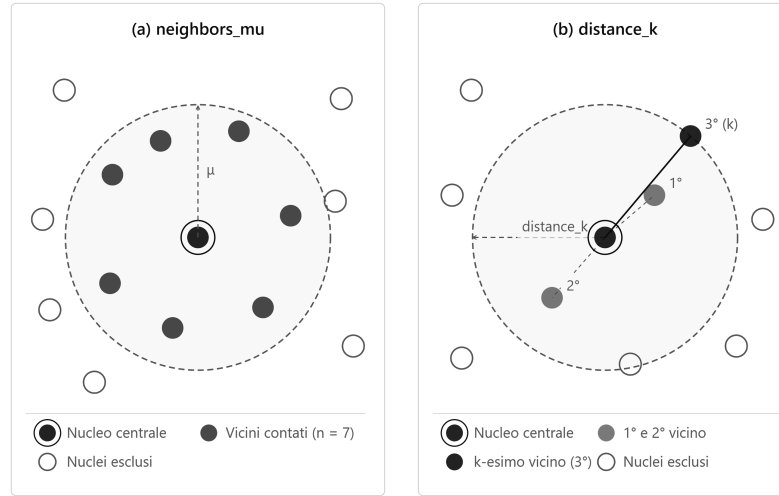


Figura 2.4: Rappresentazione schematica delle feature di densità nucleare .
 (a) **neighbors_mu**: vengono conteggiati i nuclei pieni che ricadono entro il cerchio di raggio $\mu = 100$ pixel; i nuclei con bordo grigio sono esclusi dal conteggio.
 (b) **distance_k**: le linee tratteggiate collegano il nucleo centrale ai due vicini più prossimi; la linea continua indica la distanza dal k-esimo vicino ($k = 3$), posizionato sulla circonferenza, il cui raggio corrisponde al valore della feature

Le due feature forniscono informazioni complementari sulla densità locale dei nuclei: **neighbors_mu** quantifica il numero di cellule presenti all'interno di una regione di dimensione fissa, mentre **distance_k** misura l'estensione spaziale necessaria per includere un numero prefissato di nuclei circostanti.

Per entrambi i parametri μ e k è stato effettuato un tuning, testando diversi valori e scegliendo quelli con cui si ottenevano i più alti valori di coefficiente di correlazione di Pearson della feature, aggregata a livello di paziente tramite media e deviazione standard, con le label. Per il parametro μ sono stati testati valori di lunghezze in pixel da 10 a 100 con passo 10, mentre per il parametro k sono stati testati i valori 3, 5 e 7. Per μ il valore che consentiva di ottenere la più elevata correlazione della feature aggregata con le label è risultato pari a 100, mentre per k a 3. Nella tabella 2.3 vengono riportati i nomi delle feature di densità nucleare e le relative descrizioni riassuntive.

Tabella 2.3: Feature di densità nucleare estratte a livello di singolo nucleo tramite k-d tree. I parametri $\mu = 100$ pixel e $k = 3$ sono stati selezionati tramite tuning sulla correlazione di Pearson con le label.

Feature	Descrizione
neighbors_mu	Numero di centroidi nucleari contenuti in un cerchio di raggio $\mu = 100$ pixel centrato nel centroide del nucleo considerato, calcolato tramite <code>tree.query_ball_point()</code>
distance_k	Distanza in μm tra il centroide del nucleo considerato e quello del k -esimo vicino più prossimo ($k = 3$), calcolata tramite <code>tree.query()</code>

2.4.2 Feature a livello di ROI

Le feature estratte a livello di ROI includono feature spaziali e di clustering calcolate a partire dall'MST cut dei centroidi nucleari e feature di colocalizzazione spaziale tra nuclei DLBCL e linfociti calcolate tramite il coefficiente di correlazione di Pearson e l'indice di Morisita-Malely. Per ciascuna ROI, tutte le feature sono state calcolate considerando l'insieme dei centroidi nucleari appartenenti a quella ROI. Al termine dell'elaborazione di tutte le ROI di un paziente, le feature sono state salvate nel rispettivo file Excel, organizzato in un unico foglio in cui ogni riga corrisponde a una ROI del paziente.

Feature spaziali estratte dall'MST

Le feature descritte in questa sezione derivano dal lavoro di Bhattacharjee et al. [32] e sono state calcolate sull'MST cut, la cui costruzione è stata descritta nel Paragrafo 2.4.1. La rimozione degli archi di lunghezza superiore a 35 pixel determina la formazione di cluster isolati di centroidi nucleari. A partire da tali cluster sono state estratte feature volte a caratterizzare sia la compattezza interna dei cluster sia le relazioni spaziali tra cluster differenti.

La prima feature considerata è il numero di cluster, definito come il numero totale di componenti connesse presenti nell'MST cut. Valori elevati indicano una frammentazione della distribuzione nucleare, mentre valori ridotti corrispondono a una disposizione più compatta dei nuclei.

Per ogni cluster sono state inoltre calcolate le feature intra-cluster e l'area del cluster, definita come la somma delle aree dei nuclei che lo compongono:

$$A_c = \sum_{i=1}^{N_c} a_i \quad (2.5)$$

Le feature intra-cluster descrivono la distribuzione spaziale dei nuclei all'interno dello stesso cluster. In particolare, sono state considerate le distanze euclidee tra tutte le coppie di nuclei appartenenti al cluster, dalle quali sono state ricavate la distanza totale (ottenuta dalla somma delle distanze), media, minima e massima. Sono state inoltre considerate le lunghezze degli archi dell'MST appartenenti al cluster, ottenendo la distanza totale, minima e massima degli archi interni. Le feature calcolate a livello di singolo cluster sono state aggregate a livello di ROI mediante operatori statistici quali somma, media, minimo e massimo.

Le feature inter-cluster descrivono invece le relazioni spaziali tra cluster differenti. A tale scopo è stata calcolata la distanza euclidea tra i centroidi dei cluster, definita come

$$d(C_i, C_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2.6)$$

Da tali distanze sono state ricavate le misure totale, media, minima e massima. Sono state inoltre considerate le lunghezze degli archi dell'MST originale che collegavano cluster differenti prima dell'applicazione del taglio, ottenendo anche in questo caso le misure totale, media, minima e massima.

Nel presente lavoro i cluster sono stati identificati mediante la funzione `nx.connected_components()` della libreria NetworkX, applicata all'MST cut. Il numero di cluster è stato quindi ottenuto come cardinalità dell'insieme delle componenti connesse individuate.

Le distanze euclidee intra-cluster e inter-cluster sono state calcolate mediante le funzioni `pdist()` e `squareform()` della libreria `scipy.spatial.distance`. Innanzitutto è stata definita una funzione capace di restituire tutte le distanze euclidee tra coppie di centroidi nucleari, che è stata utilizzata per il calcolo delle distanze euclidee

tra coppie di centroidi dello stesso cluster, dalle quali sono state successivamente ricavate le statistiche di interesse (somma, media, minimo e massimo). La stessa procedura è stata applicata ai centroidi dei cluster per il calcolo delle distanze euclidee inter-cluster.

L'area di ciascun cluster è stata ottenuta definendo la funzione `extract_cluster_areas()`, che somma le aree dei nuclei appartenenti alla stessa componente connessa. Le lunghezze degli archi intra-cluster sono state estratte dai sottografi ottenuti mediante `G_cut.subgraph(nodes)`, mentre le lunghezze degli archi inter-cluster sono state ricavate identificando, nell'MST originale, gli archi che collegavano nodi appartenenti a cluster differenti nell'MST dopo il taglio. Tutte le distanze sono state infine convertite da pixel a micrometri mediante il fattore di conversione $mpp=0.2498 \mu/\text{pixel}$.

Nella tabella 2.4 vengono riassunte le feature spaziali estratte dall'MST cut a livello di ROI.

Tabella 2.4: Feature a livello di ROI estratte dall'MST cut. Le feature calcolate a livello di singolo cluster sono state aggregate a livello di ROI tramite somma (total), media (average), minimo (minimum) o massimo (maximum). Tutte le distanze sono espresse in μm .

Gruppo	Nome feature (codice)	Descrizione
Numero cluster	number_of_clusters	Numero totale di componenti connesse nell'MST cut
Area cluster	total_cluster_area	Somma delle aree di tutti i cluster
	average_cluster_area	Media delle aree dei cluster
	minimum_area_of_a_cluster	Area del cluster più piccolo
	maximum_area_of_a_cluster	Area del cluster più grande
Intra-cluster (archi MST)	total_intra_cluster_total_MST_distance	Somma della lunghezza totale degli archi MST interni per cluster
	average_intra_cluster_total_MST_distance	Media della lunghezza totale degli archi MST interni per cluster
	total_intra_cluster_minimum_MST_distance	Somma della lunghezza minima degli archi MST interni per cluster
	average_intra_cluster_minimum_MST_distance	Media della lunghezza minima degli archi MST interni per cluster
	total_intra_cluster_maximum_MST_distance	Somma della lunghezza massima degli archi MST interni per cluster

continua nella pagina successiva

Tabella 2.4 – continuazione

Gruppo	Nome feature (codice)	Descrizione
	average_intra_cluster_maximum_MST_distance	Media della lunghezza massima degli archi MST interni per cluster
Intra-cluster (coppie nuclei)	total_intra_cluster_nucleus_to_nucleus_total_distance	Somma della distanza totale tra tutte le coppie di nuclei per cluster
	average_intra_cluster_nucleus_to_nucleus_total_distance	Media della distanza totale tra tutte le coppie di nuclei per cluster
	average_intra_cluster_nucleus_to_nucleus_average_distance	Media della distanza media tra tutte le coppie di nuclei per cluster
	total_intra_cluster_nucleus_to_nucleus_minimum_distance	Somma della distanza minima tra nuclei dello stesso cluster
	average_intra_cluster_nucleus_to_nucleus_minimum_distance	Media della distanza minima tra nuclei dello stesso cluster
	total_intra_cluster_nucleus_to_nucleus_maximum_distance	Somma della distanza massima tra nuclei dello stesso cluster
	average_intra_cluster_nucleus_to_nucleus_maximum_distance	Media della distanza massima tra nuclei dello stesso cluster
Inter-cluster (centroidi)	inter_cluster_centroid_to_centroid_total_distance	Somma delle distanze euclidee tra centroidi di cluster diversi
	inter_cluster_centroid_to_centroid_average_distance	Media delle distanze euclidee tra centroidi di cluster diversi
	inter_cluster_centroid_to_centroid_minimum_distance	Distanza minima tra centroidi di cluster diversi
	inter_cluster_centroid_to_centroid_maximum_distance	Distanza massima tra centroidi di cluster diversi

continua nella pagina successiva

Tabella 2.4 – continuazione

Gruppo	Nome feature (codice)	Descrizione
Inter-cluster (archi MST)	inter_cluster_total_MST_ distance	Somma delle lunghezze degli archi MST originale tra cluster diversi
	inter_cluster_average_MS T_distance	Media delle lunghezze degli archi MST originale tra cluster diversi
	inter_cluster_minimum_ MST_distance	Lunghezza minima degli archi MST originale tra cluster diversi
	inter_cluster_maximum_ MST_distance	Lunghezza massima degli archi MST originale tra cluster diversi

Feature spaziali di colocalizzazione

La distribuzione spaziale delle cellule immunitarie in relazione alle cellule tumorali nel microambiente tumorale (TME) costituisce un fattore di rilevanza clinica e prognostica nel DLBCL. Studi recenti hanno dimostrato che la prossimità spaziale tra popolazioni cellulari diverse — in particolare tra linfociti T citotossici e cellule tumorali B — è associata a outcome clinici significativamente differenti, con pattern di co-localizzazione o reciproco evitamento che riflettono il grado di risposta immunitaria anti-tumorale [33, 34]. La quantificazione di tali pattern spaziali a partire da immagini istologiche rappresenta pertanto un approccio promettente per estrarre informazioni biologicamente rilevanti sul rapporto tra sistema immunitario e cellule neoplastiche.

Le feature di colocalizzazione spaziale implementate nel presente lavoro derivano dalla metodologia proposta da Maley et al. [35], che utilizza misure ecologiche per descrivere il grado di co-localizzazione tra cellule tumorali e cellule immunitarie in immagini istopatologiche. In particolare, il coefficiente di correlazione di Pearson e l'indice di Morisita sono stati utilizzati per quantificare il grado di associazione spaziale tra nuclei classificati come DLBCL e nuclei classificati come linfociti.

Per replicare tale approccio, i nuclei sono stati preliminarmente suddivisi nelle due categorie sulla base della loro area: i nuclei con area superiore a una soglia prefissata sono stati classificati come DLBCL, mentre quelli con area inferiore sono stati classificati come linfociti. Successivamente, ciascuna ROI è stata suddivisa in una griglia di finestre quadrate di lato prefissato e, per ogni finestra, è stato conteggiato il numero di nuclei appartenenti alle due classi, memorizzando i risultati nei vettori `counts_dlbcl` e `counts_lymph`. Il coefficiente di Pearson è stato calcolato mediante la funzione `pearsonr()` del modulo `scipy.stats`, mentre per il calcolo dell'indice di Morisita-Maley è stata definita la funzione `morisita_maley()`, che normalizza i conteggi ottenendo le frequenze relative p_i e q_i e applica direttamente la formula dell'indice di sovrapposizione. In entrambi i casi, quando almeno una delle due

distribuzioni presentava deviazione standard nulla, il valore della feature è stato posto pari a zero per evitare risultati numericamente non definiti.

Il coefficiente di correlazione di Pearson misura il grado di correlazione lineare tra le distribuzioni spaziali delle due popolazioni cellulari:

$$r = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^N (y_i - \bar{y})^2}} \quad (2.7)$$

dove x_i e y_i rappresentano rispettivamente il numero di nuclei DLBCL e di linfociti presenti nella finestra i , mentre $\bar{x}$ e $\bar{y}$ sono le corrispondenti medie. Valori positivi indicano che le due popolazioni tendono a concentrarsi nelle stesse regioni della ROI, valori negativi indicano una segregazione spaziale, mentre valori prossimi a zero suggeriscono assenza di correlazione.

L'indice di Morisita-Malely misura il grado di sovrapposizione spaziale tra le due popolazioni cellulari ed è definito come:

$$M = \frac{2 \sum_{i=1}^N p_i q_i}{\sum_{i=1}^N p_i^2 + \sum_{i=1}^N q_i^2} \quad (2.8)$$

dove

$$p_i = \frac{x_i}{\sum_{j=1}^N x_j} \quad q_i = \frac{y_i}{\sum_{j=1}^N y_j} \quad (2.9)$$

rappresentano le frequenze relative delle due popolazioni cellulari nella finestra i . L'indice assume valori compresi tra 0 e 1, con valori elevati che indicano una forte co-localizzazione spaziale tra DLBCL e linfociti e valori bassi che indicano una distribuzione spaziale differente delle due popolazioni.

Poiché le finestre ai margini della ROI possono avere area inferiore a quella teorica, sono state considerate esclusivamente le finestre con area effettiva pari ad almeno il 66% dell'area nominale, per ridurre possibili bias nei conteggi cellulari. Tale valore percentuale di area nominale è stato scelto per far sì che anche le finestre ai bordi avessero un'area minima pari all'area della finestra più piccola considerata da Maley et al. [35].

Le analisi sono state ripetute per diverse combinazioni di soglia di area nucleare e dimensione della finestra. La soglia di area è stata variata tra 15 e 60 μm^2 con passo di 5 μm^2 , mentre il lato della finestra è stato variato tra 40 e 100 μm con passo di 10 μm . Le configurazioni più informative sono state selezionate sulla base della correlazione delle feature aggregate a livello di paziente con le label cliniche disponibili. Sono risultate più informative le combinazioni: soglia di area 50 μm^2

con lato finestra $60 \mu\text{m}$, soglia di area $50 \mu\text{m}^2$ con lato finestra $70 \mu\text{m}$, e soglia di area $60 \mu\text{m}^2$ con lato finestra $70 \mu\text{m}$; entrambe le tipologie di feature sono state calcolate con tutte e tre le combinazioni di parametri. Nella tabella 2.5 vengono riassunte tutte le feature di colocalizzazione spaziale calcolate a livello di ROI.

Tabella 2.5: Feature di colocalizzazione spaziale a livello di ROI, calcolate per le tre combinazioni di parametri selezionate. La soglia di area (**thr**) distingue nuclei DLBCL da linfociti; il lato della finestra (**w**) definisce la dimensione della griglia di suddivisione della ROI.

Combinazione	Nome feature (codice)	Descrizione
Soglia $50 \mu\text{m}^2$ Finestra $60 \mu\text{m}$	pearson_thr50_w60	Coefficiente di correlazione di Pearson tra i conteggi di nuclei DLBCL e linfociti nelle finestre
	morisita_thr50_w60	Indice di Morisita-Maley tra i conteggi di nuclei DLBCL e linfociti nelle finestre
Soglia $50 \mu\text{m}^2$ Finestra $70 \mu\text{m}$	pearson_thr50_w70	Coefficiente di correlazione di Pearson tra i conteggi di nuclei DLBCL e linfociti nelle finestre
	morisita_thr50_w70	Indice di Morisita-Maley tra i conteggi di nuclei DLBCL e linfociti nelle finestre
Soglia $60 \mu\text{m}^2$ Finestra $70 \mu\text{m}$	pearson_thr60_w70	Coefficiente di correlazione di Pearson tra i conteggi di nuclei DLBCL e linfociti nelle finestre
	morisita_thr60_w70	Indice di Morisita-Maley tra i conteggi di nuclei DLBCL e linfociti nelle finestre

2.4.3 Feature a livello di paziente

Feature di colocalizzazione a livello di paziente

Oltre all'estrazione delle feature di colocalizzazione a livello di singola ROI, è stata effettuata un'analisi a livello di paziente al fine di sfruttare congiuntamente le informazioni provenienti da tutte le regioni disponibili. In questo caso i conteggi dei nuclei DLBCL e dei linfociti ottenuti nelle finestre di tutte le ROI appartenenti al medesimo soggetto sono stati concatenati in due vettori unici, mantenendo la distinzione tra le due popolazioni cellulari.

Successivamente, il coefficiente di Pearson e l'indice di Morisita-Malely sono stati calcolati direttamente su tali vettori aggregati, applicando le stesse procedure descritte per l'analisi a livello di ROI. In questo modo le feature ottenute descrivono il grado complessivo di associazione spaziale tra nuclei DLBCL e linfociti nell'intero insieme delle ROI del paziente, anziché all'interno di una singola regione istologica. Le due tipologie di feature sono state estratte per le stesse tre combinazioni di parametri individuate come più informative a livello di singola ROI.

2.5 Feature aggregation

Le feature estratte nei paragrafi precedenti sono caratterizzate da differenti livelli di granularità. Le feature ROI-level, essendo già calcolate a livello di regione istologica, sono state aggregate a livello di paziente mediante metriche statistiche descrittive tradizionali, nello specifico: media, deviazione standard, 10°, 25°, 50°, 75°, e 90° percentile.

Le feature nucleus-level, invece, essendo disponibili per ogni singolo nucleo, sono state aggregate sia mediante le statistiche classiche che tramite approcci in grado di preservare informazioni relative all'eterogeneità e all'organizzazione spaziale delle popolazioni cellulari. In particolare, oltre alle aggregazioni statistiche convenzionali, sono stati utilizzati il Cell Feature Level Co-occurrence Matrix (CFLCM) e il metodo Bag of Words (BoW). Tali tecniche consentono di descrivere aspetti complementari delle distribuzioni delle feature nucleari: il CFLCM cattura relazioni spaziali tra nuclei caratterizzati da valori vicini o distanti di una determinata feature, mentre il BoW descrive la variabilità delle ROI attraverso una rappresentazione basata su pattern ricorrenti. In figura 2.3 vengono riassunti i metodi di aggregazione applicati alle specifiche tipologie di feature.

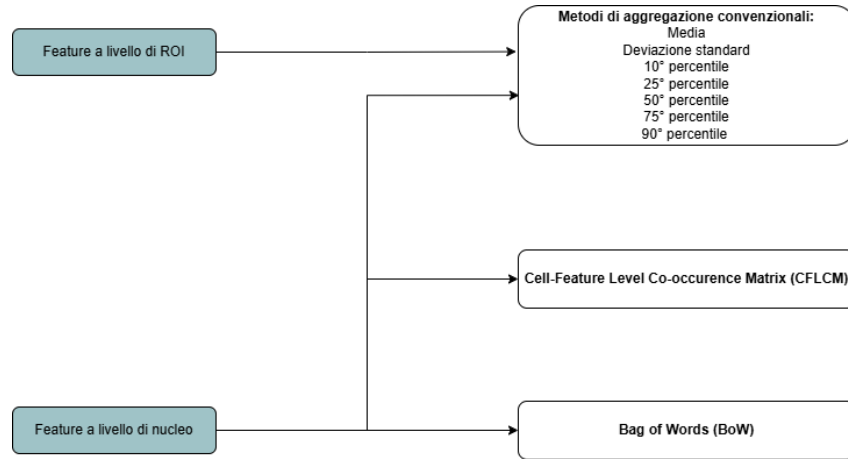


Figura 2.5: Schema dei metodi di aggregazione applicati alle feature a livello di nucleo e a livello di ROI, per ottenere feature a livello di paziente.

Durante la fase di aggregazione delle feature a livello di nucleo è stato applicato un filtro per escludere, dall'aggregazione a livello paziente delle feature `eccentricity`, `effective_eccentricity_90`, `closeness`, `betweenness`, `mean_edge_length`, `std_edge_length` e `skew_edge_length`, i nuclei per i quali il grado non raggiungeva il requisito minimo necessario alla definizione della feature e per i quali il relativo valore era stato quindi posto a zero.

2.5.1 Cell Feature Level Co-occurrence Matrix (CFLCM)

Il CFLCM trae ispirazione dalla Gray-Level Co-occurrence Matrix (GLCM), introdotta da Haralick et al. [36] come strumento per la caratterizzazione della texture nelle immagini digitali. La GLCM è una matrice quadrata in cui l'elemento in posizione (i, j) rappresenta il numero di volte in cui due pixel a distanza e orientazione prefissate presentano rispettivamente intensità i e intensità j ; da essa vengono poi derivate statistiche scalari — le metriche di Haralick — che descrivono proprietà della tessitura quali contrasto, omogeneità, correlazione ed entropia. Il CFLCM trasla questo stesso principio dal livello dei pixel al livello delle cellule [37, 38]: anziché considerare le intensità di pixel adiacenti, considera i valori di una feature morfologica nucleare (ad esempio l'area o l'eccentricità) di nuclei vicini secondo una definizione di vicinato spaziale. L'elemento in posizione (i, j) della CFLCM conta quindi il numero di coppie di nuclei vicini in cui il primo nucleo presenta il livello di feature i e il secondo il livello j . Analogamente alla GLCM, dalla CFLCM vengono estratte le stesse metriche di Haralick, che in questo contesto non descrivono più la texture dell'immagine a livello di pixel, bensì l'eterogeneità morfologica e l'organizzazione spaziale dei nuclei all'interno del tessuto. Il Cell

Feature Level Co-occurrence Matrix (CFLCM) è una tecnica proposta da Saito et al. [37] e successivamente applicata da Xu et al. [38] per quantificare l'eterogeneità morfologica e l'organizzazione spaziale delle cellule in immagini istopatologiche. A differenza delle aggregazioni statistiche tradizionali, che sintetizzano esclusivamente la distribuzione dei valori di una feature, il CFLCM considera simultaneamente il valore della feature e la disposizione spaziale delle cellule, permettendo di caratterizzare fenomeni di eterogeneità tissutale che non emergerebbero dall'analisi delle sole statistiche descrittive.

Per ciascuna feature nucleare, i valori osservati sull'intero dataset vengono inizialmente convertiti in un insieme di livelli discreti. Nel presente lavoro le feature continue sono state quantizzate mediante suddivisione uniforme in 16 livelli dell'intervallo compreso tra il valore minimo e il valore massimo della feature, mentre per le feature discrete, come il grado e l'indicatore di endpoint dell'MST, ogni valore osservato è stato associato direttamente a un livello distinto, al fine di non ottenere matrici di co-occorrenza eccessivamente sparse ed alterare dunque le feature da esse estratte.

Successivamente viene costruita una matrice quadrata $M \in \mathbb{R}^{N_g \times N_g}$, dove N_g rappresenta il numero di livelli discreti della feature. L'elemento $M(i, j)$ conta il numero di coppie di centroidi nucleari spazialmente correlati per cui il primo nucleo appartiene al livello i e il secondo al livello j . La matrice viene costruita in modo simmetrico incrementando contemporaneamente le celle (i, j) e (j, i) .

Le relazioni di vicinato considerate nel presente lavoro sono le tre formulate da Saito et al. [37]. La prima modalità, denominata *nearest*, associa a ciascun nucleo esclusivamente il vicino più prossimo, purché la distanza tra i due centroidi non superi cinque volte il diametro nucleare medio globale. La seconda modalità, denominata *circle*, considera tutte le coppie formate dal nucleo di riferimento con i nuclei i cui centroidi ricadono all'interno di un cerchio centrato sul centroide del nucleo di riferimento e avente raggio pari a cinque volte il diametro nucleare medio globale. La terza modalità, denominata *lattice*, suddivide la ROI in una griglia regolare di celle quadrate aventi lato pari a due volte il diametro nucleare medio globale e considera associate tutte le coppie di nuclei i cui centroidi appartengono alla stessa cella della griglia.

Nel presente lavoro, analogamente a quanto effettuato per le feature di colocalizzazione, le celle di bordo con area inferiore al 66% dell'area teorica sono state escluse dall'analisi per limitare possibili bias dovuti alla presenza di regioni parziali.

Per ogni feature nucleare e per ogni paziente sono state quindi costruite tre differenti matrici CFLCM, corrispondenti alle configurazioni *nearest*, *circle* e *lattice*. Ciascuna matrice fornisce una rappresentazione differente delle relazioni spaziali tra nuclei e consente di caratterizzare l'eterogeneità tissutale a differenti scale di vicinato.

È opportuno sottolineare che, a differenza della GLCM classica in cui la matrice viene calcolata separatamente per diverse direzioni spaziali prefissate (tipicamente

0° , 45° , 90° e 135°) al fine di catturare l'anisotropia della tessitura, le tre definizioni di vicinato adottate nel CFLCM sono intrinsecamente isotropiche: il vicinato *nearest* considera il vicino più prossimo indipendentemente dalla direzione, il vicinato *circle* include tutti i nuclei entro un cerchio, e il vicinato *lattice* raggruppa i nuclei per cella di griglia senza privilegiare alcuna direzione. Di conseguenza, per ciascuna coppia feature–vicinato viene costruita un'unica matrice CFLCM, dalla quale le metriche di Haralick vengono estratte una sola volta, senza necessità di mediare su direzioni multiple come avviene nella procedura standard per la GLCM.

Lo schema complessivo della procedura di costruzione del CFLCM è illustrato in Figura 2.6.

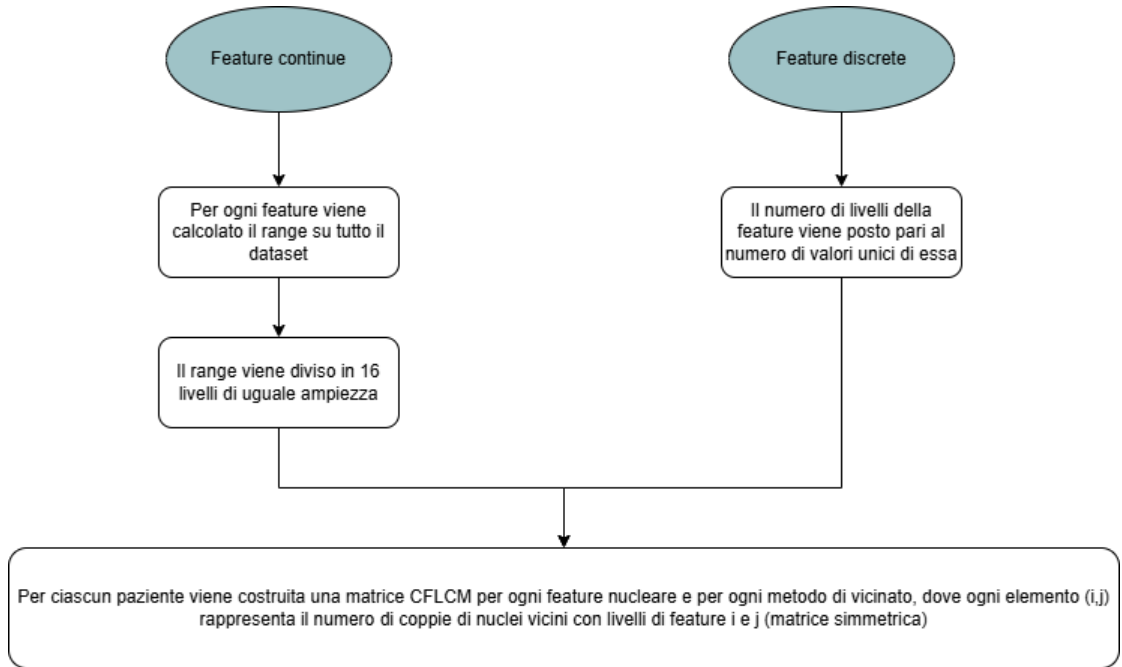


Figura 2.6: Schema della costruzione della matrice CFLCM.

Le matrici CFLCM sono state costruite separatamente per ciascuna ROI e successivamente sommate tra tutte le ROI appartenenti allo stesso paziente, ottenendo una rappresentazione complessiva dell'organizzazione spaziale della feature a livello paziente.

Prima dell'estrazione dei descrittori tessiturali, le matrici CFLCM sono state normalizzate dividendo ciascun elemento per la somma totale dei valori all'interno degli elementi della matrice (totale delle co-occorrenze), ottenendo così una distribuzione di probabilità congiunta. Tale passaggio è necessario affinché le statistiche di Haralick possano essere interpretate come misure derivate da una distribuzione di probabilità discreta [36]. Tali descrittori consentono di quantificare differenti

aspetti dell'eterogeneità cellulare, quali uniformità, complessità, dispersione dei livelli della feature e correlazione tra nuclei spazialmente vicini.

A partire da ciascuna matrice CFLCM normalizzata sono stati estratti sei descrittori tessiturali di Haralick [36], ampiamente utilizzati per caratterizzare la distribuzione spaziale dei livelli di grigio dei pixel in radiomica. Tali descrittori consentono di quantificare differenti aspetti dell'eterogeneità cellulare, quali uniformità, complessità, dispersione dei livelli della feature e correlazione tra nuclei spazialmente vicini. L'*Angular Second Moment* (ASM) misura il grado di uniformità della matrice ed è definito come:

$$ASM = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P(i, j)^2 \quad (2.10)$$

valori elevati indicano una distribuzione concentrata in poche combinazioni di livelli e quindi una maggiore omogeneità. L'*Entropy* quantifica invece il grado di disordine della distribuzione ed è definita come:

$$Entropy = - \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P(i, j) \log P(i, j) \quad (2.11)$$

assumendo valori maggiori in presenza di elevata eterogeneità. Il *Contrast* misura la variabilità locale della feature tra nuclei associati spazialmente ed è calcolato come:

$$Contrast = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - j)^2 P(i, j) \quad (2.12)$$

valori elevati indicano frequenti associazioni tra nuclei appartenenti a livelli molto differenti. La *Homogeneity* valuta invece quanto la distribuzione sia concentrata lungo la diagonale principale della matrice, ovvero quanto nuclei vicini tendano ad avere valori simili della feature:

$$Homogeneity = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{P(i, j)}{1 + (i - j)^2} \quad (2.13)$$

La *Correlation* misura la dipendenza lineare tra i livelli osservati nelle coppie di nuclei associate:

$$Correlation = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{(i - \mu_i)(j - \mu_j)}{\sigma_i \sigma_j} P(i, j) \quad (2.14)$$

dove μ_i , μ_j , σ_i e σ_j rappresentano rispettivamente le medie e le deviazioni standard marginali della matrice normalizzata. Infine, la *Difference Entropy* viene calcolata sulla distribuzione delle differenze tra livelli adiacenti:

$$P_{x-y}(k) = \sum_{|i-j|=k} P(i, j) \quad (2.15)$$

$$DifferenceEntropy = - \sum_{k=0}^{N_g-1} P_{x-y}(k) \log P_{x-y}(k) \quad (2.16)$$

e fornisce una misura dell'eterogeneità delle differenze tra nuclei spazialmente correlati. Nel complesso, tali descrittori consentono di caratterizzare differenti aspetti dell'organizzazione spaziale della feature considerata, descrivendo uniformità, disordine, contrasto locale, similarità tra nuclei vicini e dipendenza statistica tra i livelli osservati.

Il metodo CFLCM fornisce quindi 18 metodi di aggregazione per le feature a livello di nucleo; i relativi suffissi che sono stati concatenati ai nomi delle feature a seguito dell'aggregazione vengono riportati nella tabella 2.6

Tabella 2.6: Suffissi delle combinazioni CFLCM applicati alle feature nucleari, ottenuti dal prodotto tra le 3 definizioni di vicinato e le 6 metriche di Haralick. Il nome della feature aggregata è ottenuto concatenando il nome della feature di base con il suffisso corrispondente, separati da un trattino basso.

Vicinato	Suffisso	Metrica di Haralick
Nearest neighbor	nearest_diff_entropy	Entropia differenziale
	nearest_ASM	Angular Second Moment (ASM)
	nearest_contrast	Contrasto
	nearest_homogeneity	Omogeneità
	nearest_correlation	Correlazione
	nearest_entropy	Entropia
Circolare	circle_diff_entropy	Entropia differenziale
	circle_ASM	Angular Second Moment (ASM)
	circle_contrast	Contrasto
	circle_homogeneity	Omogeneità
	circle_correlation	Correlazione
	circle_entropy	Entropia
A griglia	lattice_diff_entropy	Entropia differenziale
	lattice_ASM	Angular Second Moment (ASM)
	lattice_contrast	Contrasto
	lattice_homogeneity	Omogeneità
	lattice_correlation	Correlazione
	lattice_entropy	Entropia

2.5.2 Bag of Words (BoW)

Come descritto da Powell et al. [39], il paradigma Bag of Words (BoW) consente di rappresentare quantitativamente l'eterogeneità morfologica di un tessuto trasformando le informazioni estratte da numerose regioni locali in una rappresentazione compatta a livello di paziente. Analogamente a quanto avviene nell'analisi automatica dei documenti testuali, in cui un testo viene descritto tramite la frequenza delle parole che contiene, nel contesto istopatologico un paziente viene descritto mediante la frequenza di differenti pattern morfologici osservati nel tessuto.

L'approccio prevede innanzitutto la suddivisione delle immagini in patch e l'estrazione, per ciascuna di esse, di feature quantitative derivate dai nuclei contenuti al loro interno. Le feature nucleari vengono successivamente aggregate all'interno della singola patch mediante opportune statistiche riassuntive, ottenendo un vettore descrittivo della regione considerata. Ogni patch rappresenta pertanto un'osservazione

nello spazio delle feature.

L'obiettivo del metodo è identificare automaticamente gruppi di patch caratterizzate da proprietà morfologiche simili, denominati *visual words*. A tale scopo, i vettori descrittivi di tutte le patch appartenenti al training set vengono sottoposti a clustering non supervisionato mediante algoritmo K-Means, che suddivide lo spazio delle feature in K regioni distinte. Il centroide di ciascun cluster rappresenta una visual word del vocabolario, mentre l'insieme dei K centroidi costituisce il dizionario morfologico utilizzato per descrivere i pazienti.

Una volta costruito il vocabolario, ogni patch viene assegnata alla visual word più vicina nello spazio delle feature. Per ciascun paziente viene quindi costruito un istogramma che riporta il numero di patch appartenenti a ciascuna visual word:

$$h_k = \sum_{i=1}^N I(c_i = k) \quad (2.17)$$

dove N rappresenta il numero totale di patch del paziente, c_i indica la visual word assegnata alla patch i ed $I(\cdot)$ è la funzione indicatrice che assume valore 1 quando la condizione è verificata e 0 altrimenti.

L'istogramma risultante descrive quindi la composizione morfologica complessiva del tessuto, quantificando la frequenza con cui compaiono i diversi pattern identificati dal clustering. Come osservato da Powell et al. [39], tale rappresentazione consente di catturare l'eterogeneità istologica del paziente mediante un numero limitato di variabili, facilmente utilizzabili come input per modelli di classificazione o di predizione della prognosi.

Nel presente lavoro è stato sviluppato un adattamento di quanto descritto da [39], applicato alle feature nucleari descritte nel Paragrafo 2.4.1. L'intera procedura è stata eseguita separatamente per ciascuna feature, generando quindi un dizionario visivo indipendente per ogni descrittore nucleare. In questo modo il metodo non cattura associazioni tra feature differenti, ma quantifica esclusivamente l'eterogeneità spaziale della singola caratteristica considerata. I

Inizialmente le ROI da 1500×1500 pixel già disponibili sono state utilizzate direttamente come unità di analisi, senza alcuna ulteriore suddivisione in patch: ciascuna ROI costituiva pertanto una singola osservazione nello spazio delle feature, dalla quale venivano estratti i quattro descrittori statistici della feature nucleare considerata. Successivamente è stata sperimentata una suddivisione delle ROI in patch quadrate di dimensioni fisicamente comparabili a quelle adottate da Powell et al. [39]: poiché gli autori utilizzano patch 256×256 pixel a ingrandimento $10\times$ con una risoluzione di circa $1 \mu\text{m}/\text{pixel}$, corrispondenti a un'area di circa $256 \times 256 \mu\text{m}^2$, nel presente studio sono state utilizzate patch di lato 850 pixel (circa $212 \mu\text{m}$) con overlap di 200 pixel (circa $50 \mu\text{m}$), generando quattro patch

sovrapposte per ROI. Tuttavia, le feature BoW calcolate con questa configurazione hanno mostrato una correlazione con le label cliniche inferiore rispetto a quella ottenuta utilizzando le ROI come unità di analisi; si è pertanto scelto di adottare quest'ultima configurazione, impiegando direttamente le ROI come patch nell'intera procedura BoW.

Per ciascuna ROI sono stati successivamente calcolati quattro descrittori statistici della feature nucleare considerata:

- media;
- deviazione standard;
- mediana;
- numero di nuclei validi presenti nella patch.

Tali quantità sono state estratte mediante la funzione personalizzata `extract_patch_features()`, che utilizza le funzioni `np.mean()`, `np.std()` e `np.median()` della libreria NumPy. Nel caso di ROI prive di nuclei validi, tutti i descrittori sono stati posti pari a zero; nel caso di un solo nucleo, media e mediana coincidono con il valore osservato mentre la deviazione standard viene posta pari a zero.

L'insieme delle feature ROI-level di tutti i pazienti è stato quindi utilizzato per costruire il dizionario visivo. Data la disponibilità di soli 74 pazienti, si è scelto di utilizzare l'intero dataset per stimare i cluster, aumentando così la stabilità della procedura di clustering. Le feature delle ROI sono state preliminarmente standardizzate mediante la classe `StandardScaler` della libreria `scikit-learn`, che trasforma ciascuna variabile in modo da avere media nulla e deviazione standard unitaria:

$$z = \frac{x - \mu}{\sigma} \quad (2.18)$$

dove μ e σ rappresentano rispettivamente media e deviazione standard calcolate sull'insieme delle ROI. Le ROI standardizzate sono state successivamente raggruppate tramite la classe `KMeans` della libreria `scikit-learn`, utilizzando $K = 15$ cluster. Ciascun cluster definisce una visual word del dizionario morfologico e viene rappresentato dal proprio centroide nello spazio delle feature ROI-level.

Una volta costruito il vocabolario, ogni ROI di ciascun paziente è stata assegnata alla visual word più vicina mediante il metodo `predict()` della classe `KMeans`. Per ogni paziente è stato quindi costruito un istogramma normalizzato delle frequenze di appartenenza ai diversi cluster:

$$h_k = \frac{n_k}{\sum_{j=1}^K n_j} \quad (2.19)$$

dove n_k rappresenta il numero di ROI assegnate alla visual word k .

Infine, come misura sintetica dell'eterogeneità morfologica del paziente, è stata calcolata l'entropia di Shannon dell'istogramma mediante la funzione `entropy()` del modulo `scipy.stats`:

$$H = - \sum_{k=1}^K h_k \log(h_k) \quad (2.20)$$

Valori elevati di entropia indicano una distribuzione relativamente uniforme delle patch tra numerose visual words e quindi una maggiore eterogeneità morfologica, mentre valori bassi indicano la predominanza di poche categorie morfologiche e una maggiore omogeneità del tessuto. Per ogni feature nucleare è stata quindi ottenuta una singola feature patient-level denominata **BoW_entropy**, successivamente utilizzata nelle analisi di classificazione.

2.6 Feature selection

Le aggregazioni descritte nel paragrafo precedente hanno permesso di ottenere un dataset di sole feature a livello di paziente, a partire da esso è stata effettuata la feature selection, eseguita separatamente per le due label diagnostiche disponibili nel dataset, ovvero la classificazione Hans e la classificazione Nanostring. Per ciascuna delle due label è stata inizialmente effettuata una selezione diretta delle feature utilizzando un classificatore Random Forest e due differenti misure di importanza: la Gini importance e la SHAP importance.

La classificazione è stata effettuata mediante la classe `RandomForestClassifier` della libreria `scikit-learn`, utilizzando 300 alberi decisionali (`n_estimators = 300`), criterio di suddivisione basato sull'indice di Gini (`criterion = "gini"`), profondità massima pari a 8 (`max_depth = 8`) e numero minimo di campioni per foglia pari a 6 (`min_samples_leaf = 6`). Ad ogni nodo dell'albero viene selezionata la variabile che massimizza la riduzione dell'impurità di Gini:

$$G = 1 - \sum_{c=1}^C p_c^2 \quad (2.21)$$

dove p_c rappresenta la proporzione di campioni appartenenti alla classe c . La riduzione di impurità prodotta da una feature viene accumulata su tutti i nodi e su tutti gli alberi della foresta; la normalizzazione finale di tali contributi fornisce la cosiddetta Gini importance, ottenuta nel codice tramite l'attributo `feature_importances_` del modello addestrato.

Accanto alla Gini importance è stata calcolata anche la SHAP importance mediante

la libreria **SHAP**. In particolare è stato utilizzato l'oggetto **TreeExplainer**, specificamente progettato per modelli ad albero. Per ogni paziente e per ogni feature vengono calcolati i corrispondenti valori SHAP, che quantificano il contributo della feature alla predizione del modello rispetto ad una predizione media di riferimento. Formalmente, il valore SHAP della feature i è definito come:

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} [f(S \cup \{i\}) - f(S)] \quad (2.22)$$

dove F rappresenta l'insieme completo delle feature, S un suo sottoinsieme e $f(\cdot)$ l'output del modello. Nel presente lavoro l'importanza globale di una feature è stata ottenuta come media del valore assoluto dei corrispondenti valori SHAP:

$$I_{SHAP}(i) = \frac{1}{N} \sum_{j=1}^N |\phi_{ij}| \quad (2.23)$$

dove N indica il numero di pazienti.

Le feature sono state successivamente ordinate in base alla SHAP importance. Sono state selezionate automaticamente le prime dieci feature della classifica SHAP; ulteriori feature sono state poi aggiunte seguendo l'ordinamento della Gini importance fino ad ottenere un insieme finale di venti feature distinte. Tale procedura è implementata mediante la funzione personalizzata `get_top_features_no_duplicates()`. Per ciascuna feature selezionata è stato inoltre calcolato il coefficiente di correlazione di Pearson rispetto alla label considerata mediante la funzione `pearsonr()` della libreria `scipy.stats`. Le tabelle 2.7 e 2.8 riportano le feature selezionate per ciascuna classificazione insieme ai corrispondenti valori di Gini importance, SHAP importance e correlazione di Pearson con il target.

Tabella 2.7: Feature selezionate per il classificatore Hans, con valori di Gini importance, SHAP importance e correlazione di Pearson con la label. Ord.: ordine di selezione.

Ord.	Feature	Gini	SHAP	Pearson
1	pearson_thr60_w70	0.0276	0.0129	-0.4271
2	pearson_thr60_w70_p25	0.0210	0.0109	-0.3013
3	pearson_thr50_w60	0.0217	0.0108	-0.4306
4	pearson_thr50_w70	0.0217	0.0099	-0.4277
5	closeness_cut_circle_contrast	0.0184	0.0098	0.3503
6	mean_edge_length_full_circle_contrast	0.0155	0.0094	-0.3207
7	mean_edge_length_full_lattice_entropy	0.0117	0.0072	-0.2582

continua nella pagina successiva

Tabella 2.7 – continuazione

Ord.	Feature	Gini	SHAP	Pearson
8	skew_edge_length_full_std	0.0100	0.0072	-0.1897
9	minimum_area_of_a_cluster_p25	0.0120	0.0070	0.1232
10	pearson_thr50_w60_p50	0.0124	0.0067	-0.3741
11	endpoint_full_std	0.0122	0.0048	0.2964
12	endpoint_full_mean	0.0114	0.0048	0.2917
13	mean_edge_length_full_nearest_contrast	0.0101	0.0061	-0.3002
14	closeness_cut_circle_diff_entropy	0.0100	0.0058	0.3585
15	closeness_full_lattice_diff_entropy	0.0097	0.0030	0.1627
16	minimum_area_of_a_cluster_p90	0.0093	0.0045	0.2413
17	mean_edge_length_full_circle_diff_entropy	0.0091	0.0054	-0.3093
18	pearson_thr50_w60_mean	0.0091	0.0047	-0.3416
19	minimum_area_of_a_cluster_std	0.0086	0.0038	0.2642
20	pearson_thr60_w70_mean	0.0081	0.0044	-0.3026

Tabella 2.8: Feature selezionate per il classificatore Nanostring, con valori di Gini importance, SHAP importance e correlazione di Pearson con la label. Ord.: ordine di selezione.

Ord.	Feature	Gini	SHAP	Pearson
1	pearson_thr50_w60_std	0.0244	0.0134	-0.3421
2	pearson_thr50_w70_std	0.0241	0.0130	-0.4013
3	total_cluster_area_p10	0.0202	0.0116	0.4433
4	closeness_cut_lattice_entropy	0.0176	0.0115	0.3712
5	pearson_thr50_w70_mean	0.0198	0.0098	-0.3743
6	pearson_thr60_w70_p75	0.0170	0.0093	-0.4114
7	pearson_thr60_w70	0.0156	0.0082	-0.4399
8	minimum_area_of_a_cluster_std	0.0146	0.0082	0.3489
9	degree_full_circle_ASM	0.0133	0.0081	-0.3962
10	betweenness_full_lattice_entropy	0.0107	0.0068	-0.3340
11	number_of_clusters_std	0.0133	0.0063	0.3064
12	effective_eccentricity_90_full_lattice_contrast	0.0097	0.0058	0.3523
13	mean_edge_length_full_lattice_entropy	0.0093	0.0052	-0.3069
14	area_micron_circle_contrast	0.0088	0.0037	0.0991
15	eccentricity_full_circle_contrast	0.0083	0.0055	0.2916

continua nella pagina successiva

Tabella 2.8 – continuazione

Ord.	Feature	Gini	SHAP	Pearson
16	std_edge_length_full_nearest_ent ropy	0.0082	0.0055	-0.3441
17	inter_cluster_centroid_to_centroi d_maximum_distance_p90	0.0079	0.0037	0.1087
18	std_edge_length_full_lattice_entr opy	0.0079	0.0050	-0.3506
19	mean_edge_length_full_nearest_c orrelation	0.0078	0.0043	0.4106
20	skew_edge_length_full_lattice_co ntrast	0.0075	0.0027	0.1244

L'analisi dei risultati ottenuti mediante la procedura descritta ha evidenziato due criticità principali. In primo luogo, molte delle feature selezionate differivano esclusivamente per specifici parametri di estrazione (ad esempio differenti dimensioni di finestra, differenti soglie), suggerendo una forte ridondanza informativa. In secondo luogo, sia i valori di Gini importance sia quelli di SHAP importance risultavano generalmente bassi e distribuiti su un elevato numero di feature, senza evidenziare descrittori chiaramente dominanti.

Per ridurre la ridondanza del dataset e migliorare l'interpretabilità del processo di selezione è stata quindi introdotta una fase preliminare di preselezione basata sulla correlazione tra feature. Tale procedura è stata implementata mediante la funzione personalizzata `greedy_correlation_filter()`, applicata separatamente alle due label Hans e NanoString.

In una prima fase è stata calcolata, per ogni feature, la correlazione di Pearson assoluta rispetto alla label binaria mediante la funzione `pearsonr()` del modulo `scipy.stats`. Successivamente è stata costruita la matrice di correlazione assoluta feature-feature mediante il metodo `corr()` della libreria `pandas`. Per ogni coppia di feature con correlazione superiore ad una soglia prefissata T pari a 0.8, le due feature sono state confrontate sulla base della rispettiva correlazione assoluta con la label. La feature caratterizzata dal valore più elevato di coefficiente di correlazione di Pearson con la label è stata mantenuta, mentre l'altra è stata eliminata. In caso di uguaglianza dei valori di correlazione con la label, la procedura eliminava la prima feature incontrata. L'algoritmo è stato applicato in maniera greedy fino all'analisi completa di tutte le coppie di feature.

Sul dataset così filtrato è stata quindi ripetuta l'intera procedura di feature selection basata su Random Forest e SHAP.

Le Tabelle 2.9 e 2.10 riportano le feature selezionate dopo l'applicazione del filtro di correlazione.

Tabella 2.9: Feature selezionate per il classificatore Hans dopo l'applicazione del filtro di correlazione, con valori di Gini importance, SHAP importance e correlazione di Pearson con la label. Ord.: ordine di selezione.

Ord.	Feature	Gini	SHAP	Pearson
1	pearson_thr50_w60_p50	0.0542	0.0276	-0.3741
2	closeness_cut_circle_diff_entropy	0.0625	0.0275	0.3585
3	pearson_thr50_w60	0.0541	0.0245	-0.4306
4	maximum_area_of_a_cluster_p10	0.0299	0.0196	0.1833
5	total_cluster_area_p10	0.0358	0.0179	0.3885
6	skew_edge_length_full_std	0.0344	0.0173	-0.1897
7	skew_edge_length_cut_nearest_diff_entropy	0.0223	0.0121	0.1752
8	inter_cluster_minimum_MST_distance_p50	0.0199	0.0117	-0.2159
9	inter_cluster_minimum_MST_distance_p75	0.0259	0.0111	-0.2636
10	minimum_area_of_a_cluster_p10	0.0173	0.0109	0.2089
11	closeness_full_lattice_contrast	0.0267	0.0097	0.1784
12	closeness_cut_circle_correlation	0.0215	0.0100	0.2757
13	inter_cluster_centroid_to_centroid_total_distance_p90	0.0178	0.0066	-0.0791
14	morisita_thr50_w60_std	0.0176	0.0072	-0.2309
15	inter_cluster_minimum_MST_distance_p25	0.0162	0.0070	-0.2761
16	pearson_thr50_w60_p10	0.0161	0.0043	-0.2259
17	inter_cluster_average_MST_distance_std	0.0159	0.0088	-0.2049
18	mean_edge_length_cut_BoW_entropy	0.0156	0.0069	0.0357
19	eccentricity_full_BoW_entropy	0.0151	0.0065	-0.2077
20	closeness_cut_lattice_contrast	0.0149	0.0055	0.3026

Tabella 2.10: Feature selezionate per il classificatore Nanostring dopo l'applicazione del filtro di correlazione, con valori di Gini importance, SHAP importance e correlazione di Pearson con la label. Ord.: ordine di selezione.

Ord.	Feature	Gini	SHAP	Pearson
1	pearson_thr50_w70_std	0.0928	0.0493	-0.4013
2	total_cluster_area_p10	0.0582	0.0314	0.4433
3	pearson_thr60_w70	0.0593	0.0302	-0.4399
4	mean_edge_length_full_nearest_correlation	0.0484	0.0267	0.4106
5	inter_cluster_minimum_MST_distance_p50	0.0383	0.0236	-0.2449
6	inter_cluster_average_MST_distance_std	0.0399	0.0226	-0.2171
7	pearson_thr60_w70_p75	0.0377	0.0206	-0.4114
8	mean_edge_length_full_circle_correlation	0.0447	0.0204	0.2261
9	area_micron_p10	0.0273	0.0154	0.2784
10	skew_edge_length_full_std	0.0236	0.0121	-0.0971
11	skew_edge_length_cut_lattice_contrast	0.0207	0.0067	0.1666
12	closeness_cut_std	0.0203	0.0114	0.3886
13	inter_cluster_minimum_MST_distance_mean	0.0199	0.0086	-0.2239
14	average_intra_cluster_nucleus_to_nucleus_minimum_distance_p10	0.0182	0.0093	-0.2348
15	morisita_thr60_w70_mean	0.0164	0.0092	0.3209
16	skew_edge_length_cut_BoW_entropy	0.0153	0.0098	-0.1905
17	total_intra_cluster_nucleus_to_nucleus_minimum_distance_std	0.0144	0.0061	0.3538
18	betweenness_full_circle_contrast	0.0124	0.0044	-0.2146
19	inter_cluster_centroid_to_centroid_total_distance_std	0.0122	0.0040	0.1207
20	degree_cut_nearest_ASM	0.0107	0.0037	0.1220

Rispetto alla feature selection effettuata sul dataset completo, si osserva una maggiore eterogeneità dei descrittori selezionati e una riduzione della ridondanza tra feature appartenenti alla stessa famiglia. Inoltre, i valori di Gini importance e SHAP importance risultano frequentemente superiori, suggerendo una maggiore

capacità discriminante delle feature sopravvissute alla fase di preselezione. È opportuno precisare che la feature selection è stata eseguita sull'intero dataset prima della procedura di classificazione con cross-validation, anziché essere integrata all'interno di ciascun fold di training come richiederebbe una procedura rigorosa. Tale scelta, sebbene introduca un rischio di data leakage che può produrre stime ottimistiche delle prestazioni [40], è stata dettata da due ragioni pratiche. In primo luogo, la ridotta numerosità campionaria (74 pazienti). In secondo luogo, l'elevato costo computazionale della fase di aggregazione delle feature (in particolare CFLCM e BoW) ha reso proibitivo il ricalcolo completo della pipeline all'interno di ogni split. Le metriche di prestazione riportate devono pertanto essere considerate come stime potenzialmente ottimistiche, e il confronto con studi che hanno adottato una procedura di cross-validation completamente annidata deve tenere conto di questa differenza metodologica, come discusso nel Capitolo 5.

2.7 Classification

2.7.1 Costruzione dei fold

Prima della fase di classificazione, i 74 pazienti sono stati suddivisi in cinque fold mutuamente esclusivi. La suddivisione è stata effettuata in modo da mantenere il più possibile bilanciata la distribuzione delle classi per entrambe le label considerate, Hans e Nanostring, evitando che uno specifico fold contenesse una prevalenza eccessiva di una determinata classe. Tale procedura è particolarmente importante nei dataset di dimensioni ridotte, poiché riduce la variabilità delle stime di performance tra le diverse partizioni del campione [41].

Si è inoltre bilanciato, per quanto possibile, il numero di ROI nei vari fold, per non avere fold con una quantità di informazione eccessivamente minore rispetto ad altri.

Per documentare la composizione dei fold vengono riportate le tabelle 2.11, 2.12, 2.13, 2.14 e 2.15 contenenti i pazienti assegnati a ciascun gruppo, il loro numero di ROI e la distribuzione delle classi secondo entrambe le etichette di riferimento. La colonna *Coerenza* indica se le due label Hans e Nanostring concordano nella classificazione del paziente: i pazienti *non coerenti* sono quelli per cui l'algoritmo IHC di Hans e la profilazione molecolare NanoString assegnano il paziente a sottotipi molecolari differenti, riflettendo la nota limitazione di concordanza tra i due metodi di classificazione. I medesimi fold sono stati successivamente utilizzati in tutte le fasi di addestramento, selezione degli iperparametri e valutazione finale dei modelli.

Tabella 2.11: Fold 1 (14 paz., 530 ROI).
Hans: 9 NON-GC, 5 GC. Nanostring:
6 ABC, 8 Non-ABC. Coerenti: 11; non
coerenti: 3.

Paziente	Hans	Nanostring	Coer.	ROI
20_L_1001_A1_EE	NON-GC	Non-ABC	No	40
20_L_1146_A1_EE	NON-GC	ABC	Si	40
20_L_2050_EE	GC	Non-ABC	Si	40
21_L_2213_A1_EE	GC	Non-ABC	Si	40
21_L_2094_EE	GC	Non-ABC	Si	40
21_L_1080_A1_EE	NON-GC	ABC	Si	40
22_L_564_A1A_EE	NON-GC	ABC	Si	40
22_L_2421_A1E_EE	NON-GC	ABC	Si	40
22_L_1380_A1A_EE	GC	Non-ABC	Si	40
22_L_1292_A1A_EE	NON-GC	Non-ABC	No	40
23_L_2779_A1G_EE	NON-GC	ABC	Si	40
23_L_242_A1A_EE	GC	Non-ABC	Si	40
23_L_2126_A1J_EE	NON-GC	ABC	Si	40
23_L_2125_A1B_EE	NON-GC	Non-ABC	No	10

Tabella 2.12: Fold 2 (16 paz., 520 ROI).
Hans: 11 NON-GC, 5 GC. Nanostring:
8 ABC, 8 Non-ABC. Coerenti: 13; non
coerenti: 3.

Paziente	Hans	Nanostring	Coer.	ROI
20_L_1020_A1_EE	NON-GC	Non-ABC	No	40
20_L_2078_EE	GC	Non-ABC	Si	40
20_L_3128_C1_EE	GC	Non-ABC	Si	40
21_L_914_A1_EE	NON-GC	ABC	Si	40
21_L_2523_A1_EE	NON-GC	ABC	Si	10
21_L_894_A1_EE	NON-GC	ABC	Si	40
21_L_1931_EE	NON-GC	ABC	Si	10
22_L_806_A1A_EE	GC	Non-ABC	Si	30
22_L_782_A1A_EE	NON-GC	ABC	Si	40
22_L_317_A1A_EE	NON-GC	Non-ABC	No	40
22_L_2739_A3A_EE	NON-GC	ABC	Si	20
23_L_770_A1A_EE	GC	Non-ABC	Si	10
23_L_298_A1A_EE	NON-GC	ABC	Si	40
23_L_2606_A1A_EE	NON-GC	Non-ABC	No	40
24_L_130_A3B_EE	GC	Non-ABC	Si	40
21_L_2426_A1_EE	NON-GC	ABC	Si	40

Tabella 2.13: Fold 3 (15 paz., 495 ROI).
Hans: 10 NON-GC, 5 GC. Nanostring:
7 ABC, 8 Non-ABC. Coerenti: 12; non
coerenti: 3.

Paziente	Hans	Nanostring	Coer.	ROI
20_L_2902_B1_EE	NON-GC	Non-ABC	No	5
20_L_2702_A1_EE	GC	Non-ABC	Si	40
20_L_382_EE	GC	Non-ABC	Si	40
21_L_995_A1_EE	NON-GC	ABC	Si	40
21_L_3295_A1_EE	GC	Non-ABC	Si	10
21_L_1936_A1_EE	NON-GC	ABC	Si	40
22_L_3189_A1A_EE	NON-GC	Non-ABC	No	40
22_L_3188_A1A_EE	NON-GC	ABC	Si	40
22_L_1400_A1A_EE	NON-GC	ABC	Si	20
23_L_299_A1A_EE	NON-GC	ABC	Si	40
23_L_1289_A1A_EE	GC	Non-ABC	Si	40
23_L_1078_A1A_EE	NON-GC	ABC	Si	40
24_L_817_A1A_EE	NON-GC	ABC	Si	20
24_L_444_A1A_EE	NON-GC	Non-ABC	No	40
24_L_281_A1A_EE	GC	Non-ABC	Si	40

Tabella 2.14: Fold 4 (15 paz., 530 ROI).
Hans: 10 NON-GC, 4 GC. Nanostring:
8 ABC, 7 Non-ABC. Coerenti: 12; non
coerenti: 3.

Paziente	Hans	Nanostring	Coer.	ROI
20_L_2270_A1_EE	NON-GC	ABC	Si	40
21_L_421_EE	NON-GC	ABC	Si	40
21_L_2143_A1_EE	NON-GC	ABC	Si	40
21_L_2110_A1_EE	NON-GC	Non-ABC	No	40
21_L_2033_A1_EE	NON-GC	ABC	Si	30
21_L_1463_A1_EE	GC	Non-ABC	Si	40
22_L_563_A1A_EE	NON-GC	Non-ABC	No	40
22_L_3461_A1A_EE	GC	Non-ABC	Si	40
22_L_3305_A1A_EE	NON-GC	ABC	Si	40
22_L_1486_A1A_EE	NON-GC	ABC	Si	40
23_L_533_A1A_EE	NON-GC	ABC	Si	40
23_L_2514_A1B_EE	GC	Non-ABC	Si	40
23_L_1850_A1A_EE	GC	Non-ABC	Si	40
23_L_1374_A1L_EE	NON-GC	ABC	Si	10
24_L_643_A1A_EE	NON-GC	Non-ABC	No	10

Tabella 2.15: Fold 5 (14 paz., 530 ROI).
Hans: 10 NON-GC, 4 GC. Nanostring:
7 ABC, 7 Non-ABC. Coerenti: 11; non
coerenti: 3.

Paziente	Hans	Nanostring	Coer.	ROI
20_L_521_EE	NON-GC	ABC	Sì	40
21_L_695_A1_EE	GC	Non-ABC	Sì	40
21_L_2589_A1_EE	NON-GC	Non-ABC	No	40
21_L_2195_EE	NON-GC	ABC	Sì	40
21_L_1649_A1_EE	GC	Non-ABC	Sì	40
22_L_682_A1A_EE	NON-GC	ABC	Sì	40
22_L_3473_A1A_EE	NON-GC	ABC	Sì	40
22_L_2271_A1A_EE	NON-GC	ABC	Sì	40
22_L_1176_A1A_EE	GC	Non-ABC	Sì	40
23_L_2716_A1B_EE	NON-GC	ABC	Sì	10
23_L_2288_A1B_EE	GC	Non-ABC	Sì	40
23_L_1937_A1D_EE	NON-GC	Non-ABC	No	40
24_L_669_B1B_EE	NON-GC	Non-ABC	No	40
24_L_131_EE	NON-GC	ABC	Sì	40

2.7.2 Nested cross-validation

La classificazione e la valutazione delle prestazioni sono state eseguite mediante una procedura di nested 5-fold cross-validation, applicata separatamente alle classificazioni Hans e Nanostring utilizzando, in ciascun caso, il sottoinsieme di feature selezionate durante la fase di feature selection.

L'impiego della nested cross-validation è raccomandato nei dataset di piccole dimensioni quando è necessario eseguire contemporaneamente selezione degli iperparametri e valutazione delle prestazioni del classificatore, poiché consente di ottenere una stima meno ottimistica dell'errore di generalizzazione rispetto ad una cross-validation tradizionale [42].

Ad ogni iterazione uno dei cinque fold viene utilizzato come test set esterno, mentre i quattro fold rimanenti costituiscono il training-validation set. Il fold esterno non partecipa in alcun modo alla scelta degli iperparametri e viene utilizzato esclusivamente per la valutazione finale del modello.

All'interno del training-validation set viene eseguita una seconda cross-validation a quattro fold. In ciascuna iterazione interna, uno dei quattro fold viene utilizzato come validation set mentre gli altri tre costituiscono il training set. Per ogni combinazione di iperparametri viene quindi addestrato un modello sui fold di training e valutata l'accuracy sul fold di validazione. Il processo viene ripetuto ruotando il validation fold, ottenendo quattro valori di accuracy che vengono mediati per stimare la qualità della configurazione considerata.

La combinazione di iperparametri che massimizza l'accuracy media di validazione viene selezionata come configurazione ottimale per il fold esterno corrente.

Una volta identificati gli iperparametri migliori, il modello finale viene addestrato utilizzando tutti i quattro fold del training-validation set e successivamente valutato

sul fold di test esterno, ottenendo una stima indipendente delle prestazioni. In figura 4.1 viene riportato lo schema della nested cross-validation applicata.

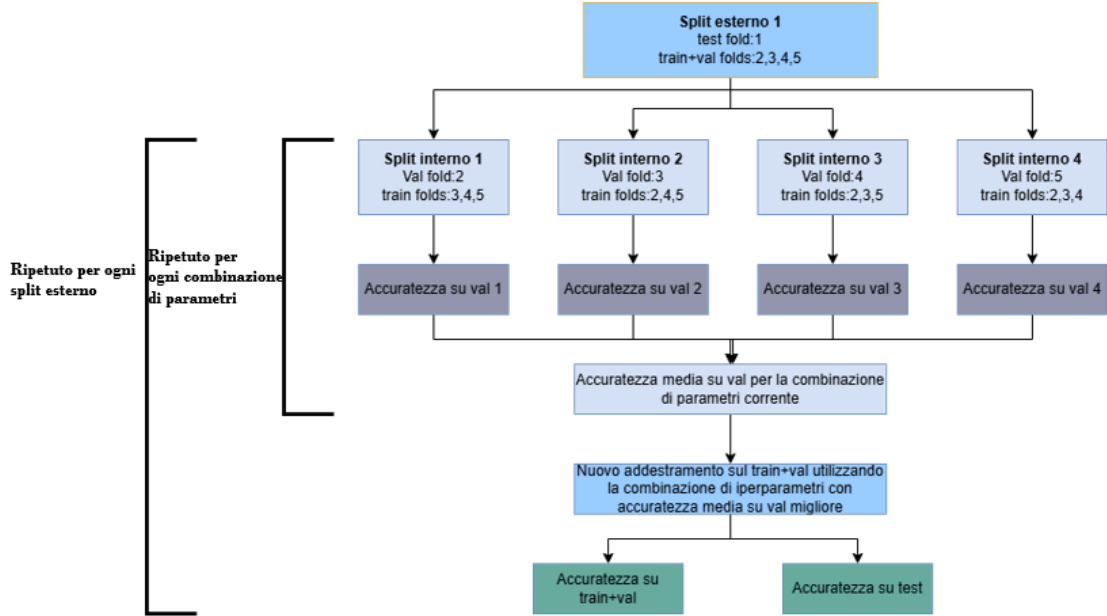


Figura 2.7: Schema della nested 5-fold cross-validation

2.7.3 Random Forest e ottimizzazione degli iperparametri

La classificazione è stata effettuata mediante l'algoritmo Random Forest, implementato tramite la classe `RandomForestClassifier` della libreria `scikit-learn`. Il Random Forest è un metodo di ensemble che combina le predizioni di molteplici alberi decisionali costruiti su differenti sottoinsiemi dei dati e delle feature. L'approccio riduce il rischio di overfitting tipico dei singoli alberi decisionali e risulta particolarmente adatto a dataset caratterizzati da un elevato numero di feature rispetto al numero di campioni disponibili [43].

Nel presente lavoro sono stati ottimizzati i seguenti iperparametri:

- **Numero di alberi**

$$n_{\text{estimators}} \in \{100, 300\}$$

Questo parametro controlla il numero di alberi che compongono la foresta. Un numero maggiore di alberi riduce generalmente la varianza del modello, al prezzo di un maggiore costo computazionale [43].

- **Profondità massima**

$$\text{max_depth} \in \{3, 5, 8\}$$

Limita la profondità massima degli alberi decisionali. Valori relativamente bassi sono frequentemente raccomandati nei dataset con pochi campioni per ridurre il rischio di overfitting [44].

- **Numero minimo di campioni in una foglia**

`min_samples_leaf` $\in \{3,6\}$

Impedisce la creazione di foglie contenenti un numero troppo ridotto di osservazioni. Valori maggiori producono modelli più regolarizzati e meno sensibili al rumore [45].

- **Numero minimo di campioni necessari per effettuare uno split**

`min_samples_split` $\in \{5,10,15\}$

Questo parametro stabilisce il numero minimo di osservazioni che un nodo deve contenere affinché possa essere ulteriormente suddiviso. Valori elevati limitano la crescita degli alberi e riducono la probabilità di apprendere strutture specifiche del training set, contribuendo a contenere l'overfitting nei dataset di piccole dimensioni [44, 45].

La scelta di valori relativamente conservativi per la profondità massima degli alberi, per il numero minimo di campioni nelle foglie terminali e per il numero minimo di campioni richiesti per effettuare uno split è stata motivata dalla ridotta numerosità campionaria (74 pazienti), che espone facilmente i modelli ad albero al rischio di sovra-adattamento ai dati di training, producendo stime eccessivamente ottimistiche delle prestazioni.

Inoltre, durante l'addestramento è stata utilizzata l'opzione

`class_weight = balanced`

che assegna automaticamente pesi maggiori alla classe meno rappresentata, compensando eventuali squilibri tra le classi [46].

Il parametro `bootstrap` è stato impostato a `False`, consentendo a ciascun albero di utilizzare l'intero training set disponibile ad ogni split della cross-validation. Tale scelta è stata adottata per limitare l'ulteriore riduzione della numerosità campionaria che il campionamento bootstrap comporterebbe in un dataset di dimensioni ridotte, in cui ciascun campione bootstrap include in media circa il 63.2% dei campioni originali [43].

Per ogni fold esterno e per ciascuna label sono stati salvati:

- gli iperparametri ottimali identificati dalla cross-validation interna;
- le accuracy ottenute sui quattro validation fold interni;
- l'accuracy media di validazione;
- l'accuracy sul training-validation set;
- l'accuracy sul test fold esterno;
- il modello finale addestrato, salvato mediante la funzione `joblib.dump()`;
- le predizioni sul test set esterno, comprensive della probabilità stimata di appartenenza alla classe positiva ottenuta tramite il metodo `predict_proba()`.

Al termine delle cinque iterazioni esterne è stato quindi ottenuto un insieme di cinque modelli e cinque valutazioni indipendenti per ciascuna delle due label considerate, consentendo di stimare la capacità di generalizzazione del classificatore sull'intero dataset.

Capitolo 3

Risultati

In questo capitolo vengono presentati i risultati ottenuti dai due classificatori RF allenati con label ricavate tramite classificazione immunoistochimica secondo l'algoritmo di Hans e tramite profilazione genica tramite piattaforma NanoString. In entrambi i casi, la classe positiva è rappresentata dal sottotipo ABC (corrispondente al fenotipo NON-GCB secondo la classificazione immunoistochimica), mentre la classe negativa corrisponde al sottotipo GCB (corrispondente al fenotipo non-ABC della classificazione NanoString).

La valutazione delle prestazioni è stata condotta mediante una procedura di validazione incrociata a cinque partizioni (*5-fold cross-validation*). Per ciascuna partizione, il modello è stato addestrato sulle quattro partizioni rimanenti e testato sulla partizione esclusa, garantendo che ogni campione fosse incluso nel set di test esattamente una volta. Le metriche di classificazione sono state calcolate separatamente per ciascuna partizione di test e successivamente aggregate.

3.1 Dataset e distribuzione delle classi

La coorte utilizzata comprende complessivamente 74 pazienti con DLBCL. Per la label *Hans*, la distribuzione delle classi risulta sbilanciata: 51 pazienti appartengono al sottotipo ABC/NON-GCB e 23 al sottotipo GCB, con un rapporto di classe di circa 2.2:1 a favore della classe positiva. Per la label *Nanostring*, la distribuzione è sostanzialmente bilanciata: 36 pazienti sono classificati come ABC e 38 come GCB, con un rapporto di classe prossimo a 1:1.

La procedura di 5-fold cross-validation ha suddiviso il dataset in cinque partizioni di dimensione approssimativamente equivalente (range: 14–16 pazienti per partizione), mantenendo all'interno di ciascuna partizione la proporzione originale delle classi. La distribuzione dei pazienti nelle partizioni di test è riportata in Tabella 3.1.

Tabella 3.1: Distribuzione dei campioni nelle partizioni di test per i due classificatori. I valori indicano il numero totale di campioni nella partizione, con la ripartizione tra classe ABC e classe GCB.

Classificatore	Partizione di test				
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Hans (totale / ABC / GCB)	14 / 9 / 5	16 / 11 / 5	15 / 10 / 5	15 / 11 / 4	14 / 10 / 4
Nanostring (totale / ABC / GCB)	14 / 6 / 8	16 / 8 / 8	15 / 7 / 8	15 / 8 / 7	14 / 7 / 7

3.2 Metriche di valutazione

Le prestazioni dei classificatori sono state quantificate mediante le seguenti metriche, calcolate a partire dalla confusion matrix ottenuta su ciascuna partizione di test, dove TP (True Positive), TN (True Negative), FP (False Positive) e FN (False Negative) si riferiscono alla classe ABC come classe positiva:

- **Accuracy:** $Acc = \frac{TP+TN}{TP+TN+FP+FN}$
- **Sensitivity** (Recall, True Positive Rate): $Sens = \frac{TP}{TP+FN}$
- **Specificity** (True Negative Rate): $Spec = \frac{TN}{TN+FP}$
- **Positive Predictive Value** (Precision): $PPV = \frac{TP}{TP+FP}$
- **Negative Predictive Value:** $NPV = \frac{TN}{TN+FN}$
- **F1-score:** $F_1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$
- **Area Under the ROC Curve** (AUC): calcolata a partire dalle probabilità predette dal modello per la classe positiva.

La scelta di riportare sensitivity e specificity separatamente, anziché affidarsi alla sola accuracy, è motivata dallo sbilanciamento delle classi presente nel dataset del classificatore Hans, che potrebbe rendere l'accuracy una misura fuorviante delle reali capacità discriminative del modello.

3.3 Risultati del classificatore Hans

3.3.1 Prestazioni per partizione di test

La tabella 3.2 riporta, per ciascun fold esterno, gli iperparametri ottimali identificati dalla cross-validation interna, l'accuracy media sui fold di validazione interni (*val*),

l'accuracy sul training set completo dopo il riaddestramento (*train+val*) e l'accuracy sul fold di test esterno (*test*) per il classificatore Hans.

Tabella 3.2: Risultati della nested cross-validation per il classificatore Hans. Per ogni fold esterno sono riportati gli iperparametri selezionati e le accuracy nelle tre fasi della procedura.

Fold	n_est	depth	min_leaf	min_split	Acc val	Acc train+val	Acc test
1	300	3	6	15	0.770	0.867	0.786
2	100	3	3	15	0.760	0.879	0.812
3	100	5	3	5	0.746	0.949	0.667
4	100	5	3	5	0.764	0.949	0.733
5	100	3	6	5	0.785	0.850	0.714
Media					0.765	0.899	0.742
SD					0.013	0.043	0.052

In tabella 3.3 vengono invece riportate le metriche di valutazione calcolate su ciascuna partizione di test e le rispettive media e deviazione standard.

Tabella 3.3: Prestazioni del classificatore Hans per ciascuna partizione di test. TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative.

Fold	TP	TN	FP	FN	Acc	Sens	Spec	PPV	NPV	AUC
1	9	2	3	0	0.786	1.000	0.400	0.750	1.000	0.956
2	10	3	2	1	0.812	0.909	0.600	0.833	0.750	0.873
3	8	2	3	2	0.667	0.800	0.400	0.727	0.500	0.600
4	9	2	2	2	0.733	0.818	0.500	0.818	0.500	0.795
5	10	0	4	0	0.714	1.000	0.000	0.714	—	0.950
Media					0.742	0.905	0.380	0.769	0.688	0.835
SD					0.058	0.096	0.228	0.054	0.239	0.147

Il classificatore Hans ha ottenuto un'accuracy media di 0.742 ± 0.058 e un F1-score medio di 0.828 ± 0.042 . La sensitivity media risulta elevata (0.905 ± 0.096), indicando che il modello riconosce correttamente la grande maggioranza dei casi ABC. In due partizioni (fold 1 e fold 5) è stata raggiunta una sensitivity pari a 1.0, ovvero tutti i campioni ABC nella partizione di test sono stati correttamente classificati. L'AUC media è pari a 0.835 ± 0.147 , con un valore di picco di 0.956 nel fold 1. La specificità media è risultata notevolmente più bassa rispetto alla sensitivity (0.380 ± 0.228), con valori che variano da 0.000 (fold 5, dove nessun campione GCB è stato correttamente classificato) a 0.600 (fold 2). Questa asimmetria tra

sensitivity e specificity riflette la tendenza del modello a classificare i campioni in favore della classe maggioritaria (ABC), fenomeno coerente con lo sbilanciamento del dataset. Il fold 3 ha mostrato le prestazioni complessivamente più basse, con accuracy pari a 0.667 e AUC pari a 0.600. Le metriche medie sono visualizzate in Figura 3.1.

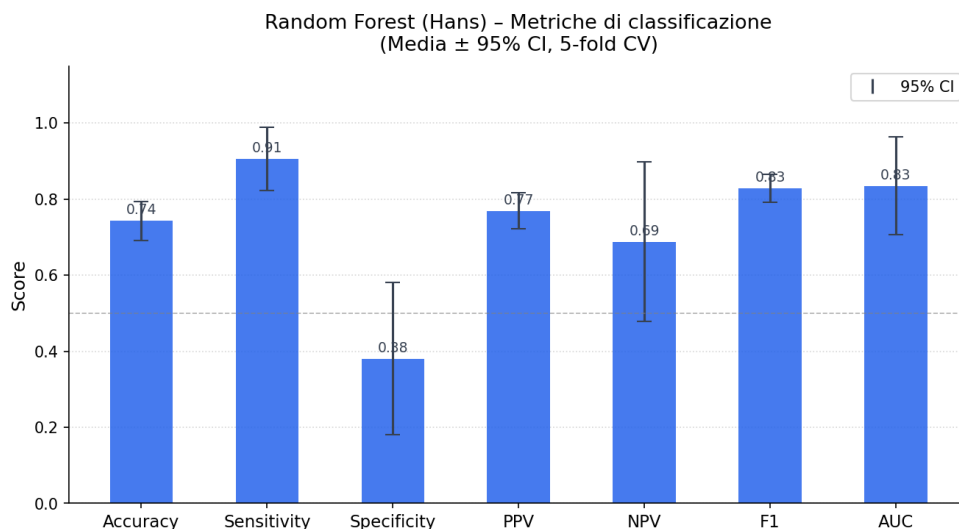


Figura 3.1: Metriche di classificazione del classificatore Hans (media su 5 fold). Le barre d'errore rappresentano l'intervallo di confidenza al 95% ($95\% \text{ CI} = 1.96 \times \text{SD}/\sqrt{5}$).

3.3.2 Confusion matrix aggregata

La confusion matrix ottenuta aggregando le predizioni su tutte e cinque le partizioni di test è riportata in Figura 3.2. Complessivamente, su 74 campioni, il classificatore ha correttamente identificato 46 casi ABC su 51 (TP) e 9 casi GCB su 23 (TN). Sono stati registrati 5 falsi negativi (casi ABC classificati come GCB) e 14 falsi positivi (casi GCB classificati come ABC).

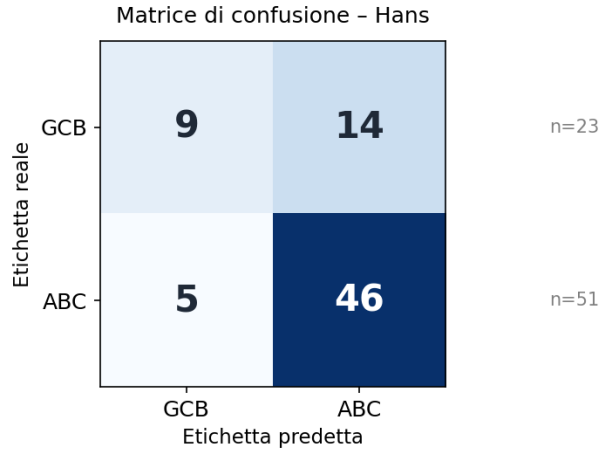


Figura 3.2: Matrice di confusione aggregata su tutte le partizioni di test del classificatore Hans.

3.3.3 Curve ROC

Le curve ROC per ciascuna partizione di test del classificatore Hans, unitamente alla curva media e alla banda di deviazione standard, sono visualizzate in Figura 3.3. La curva media presenta un andamento regolare con AUC pari a 0.835 ± 0.147 . Si osserva una marcata variabilità tra le partizioni: la curva del fold 3 si discosta notevolmente dalle altre, mantenendosi prossima alla diagonale casuale (AUC = 0.600), mentre le curve dei fold 1 e fold 5 mostrano un'eccellente capacità discriminativa (AUC di 0.956 e 0.950, rispettivamente).

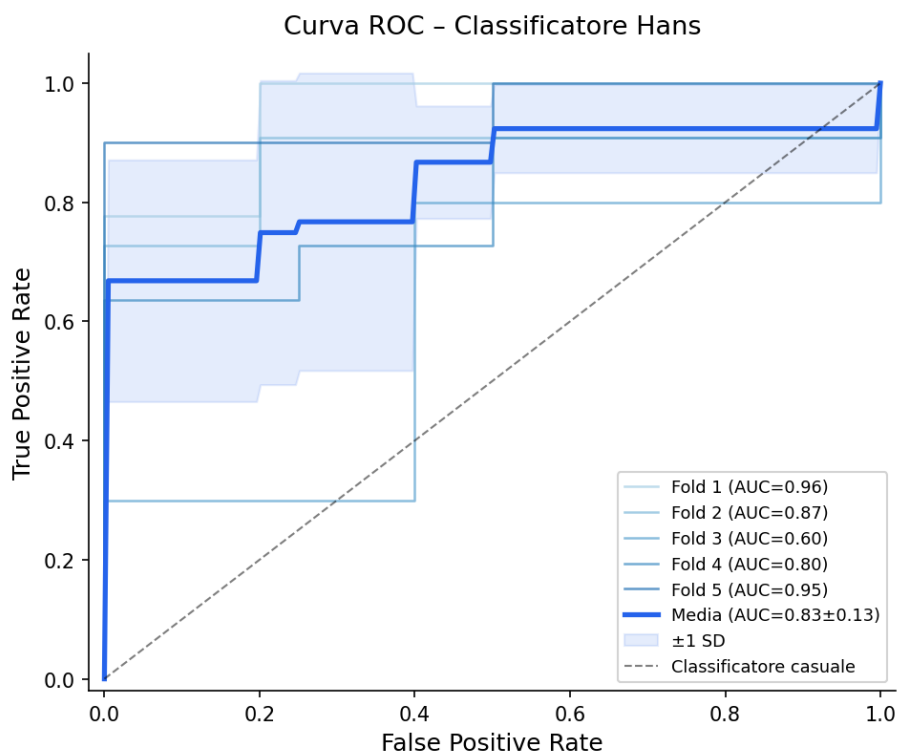


Figura 3.3: Curve ROC per ciascuna partizione di test del classificatore Hans, con curva media (linea continua) e banda ± 1 SD (area ombreggiata). La linea tratteggiata rappresenta il classificatore casuale ($AUC = 0.5$).

3.4 Risultati del classificatore Nanostring

3.4.1 Prestazioni per partizione di test

La tabella 3.4 riporta i risultati della nested cross-validation per il classificatore Nanostring.

Tabella 3.4: Risultati della nested cross-validation per il classificatore Nanostring.

Fold	n_est	depth	min_leaf	min_split	Acc val	Acc train+val	Acc test
1	100	3	3	5	0.832	0.933	0.857
2	100	3	3	15	0.792	0.879	0.812
3	300	5	6	5	0.781	0.915	0.733
4	300	5	3	15	0.712	0.966	0.733
5	300	3	3	5	0.781	0.950	0.786
Media					0.779	0.929	0.784
SD					0.038	0.031	0.048

Le metriche di valutazione complete del classificatore NanoString sono riportate in tabella 3.5.

Tabella 3.5: Prestazioni del classificatore Nanostring per ciascuna partizione di test. TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative.

Fold	TP	TN	FP	FN	Acc	Sens	Spec	PPV	NPV	AUC
1	5	7	1	1	0.857	0.833	0.875	0.833	0.875	0.875
2	6	7	1	2	0.812	0.750	0.875	0.857	0.778	0.938
3	6	5	3	1	0.733	0.857	0.625	0.667	0.833	0.875
4	7	4	3	1	0.733	0.875	0.571	0.700	0.800	0.857
5	6	5	2	1	0.786	0.857	0.714	0.750	0.833	0.776
Media					0.784	0.835	0.732	0.761	0.824	0.864
SD					0.053	0.050	0.140	0.082	0.037	0.058

Il classificatore Nanostring ha raggiunto un'accuracy media di 0.784 ± 0.053 e un F1-score medio di 0.792 ± 0.031 . Rispetto al classificatore Hans, si osserva un miglioramento sia in termini di accuracy che di equilibrio tra sensitivity e specificity. La sensitivity media è pari a 0.835 ± 0.050 , mentre la specificity media è 0.732 ± 0.140 , con un divario tra le due metriche notevolmente ridotto rispetto al classificatore Hans. Il PPV medio è 0.761 ± 0.082 e il NPV medio è 0.824 ± 0.037 , con una variabilità inter-fold contenuta per entrambe le misure.

L'AUC media è pari a 0.864 ± 0.058 , leggermente superiore a quella del classificatore Hans (0.835), con una deviazione standard sensibilmente inferiore, indicativa di una maggiore stabilità delle prestazioni tra le partizioni. Il valore di AUC più alto è stato osservato nel fold 2 (0.938), mentre il fold 5 ha prodotto il valore più basso (0.776), che rimane comunque ben al di sopra del livello di casualità.

Le metriche medie sono visualizzate in Figura 3.4.

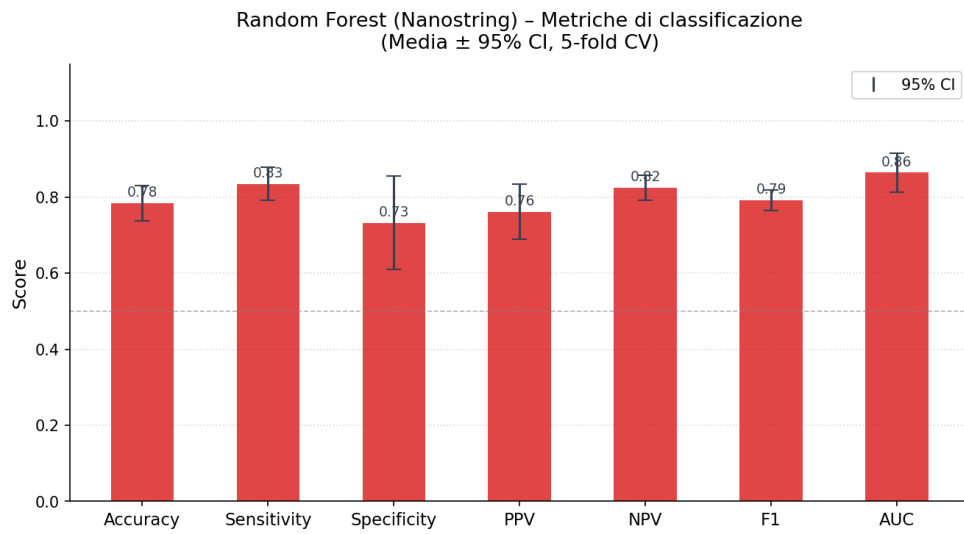


Figura 3.4: Metriche di classificazione del classificatore Nanostring (media su 5 fold). Le barre d'errore rappresentano l'intervallo di confidenza al 95% (95% CI = $1.96 \times \text{SD} / \sqrt{5}$).

3.4.2 Confusion matrix aggregata

La confusion matrix aggregata del classificatore Nanostring è riportata in Figura 3.5. Complessivamente, il modello ha correttamente classificato 30 casi ABC su 36 (TP) e 28 casi GCB su 38 (TN). I falsi negativi (casi ABC classificati come GCB) sono stati 6, mentre i falsi positivi (casi GCB classificati come ABC) sono stati 10. Rispetto al classificatore Hans, la distribuzione degli errori risulta più bilanciata tra le due classi, a fronte di un dataset di partenza con classi anch'esse più equilibrate.

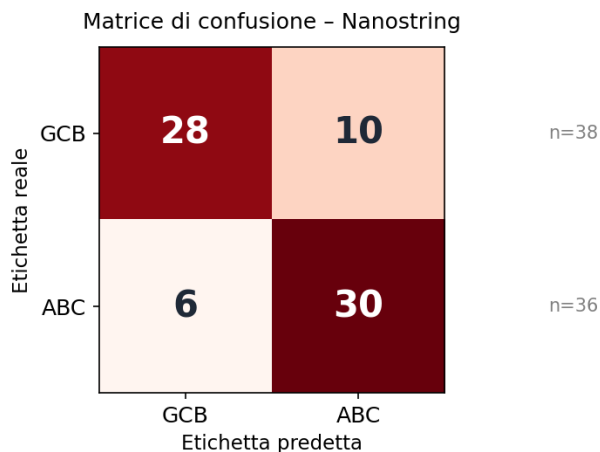


Figura 3.5: Matrice di confusione aggregata su tutte le partizioni di test del classificatore NanoString.

3.4.3 Curve ROC

Le curve ROC del classificatore NanoString sono riportate in Figura 3.6. A differenza del classificatore Hans, si osserva una minore dispersione tra le partizioni: la banda di deviazione standard attorno alla curva media risulta più stretta, a indicare una maggiore consistenza delle prestazioni. Tutte le partizioni producono curve che si mantengono chiaramente al di sopra della diagonale casuale, con valori di AUC compresi tra 0.776 e 0.938. L'AUC media è pari a 0.864 ± 0.058 .

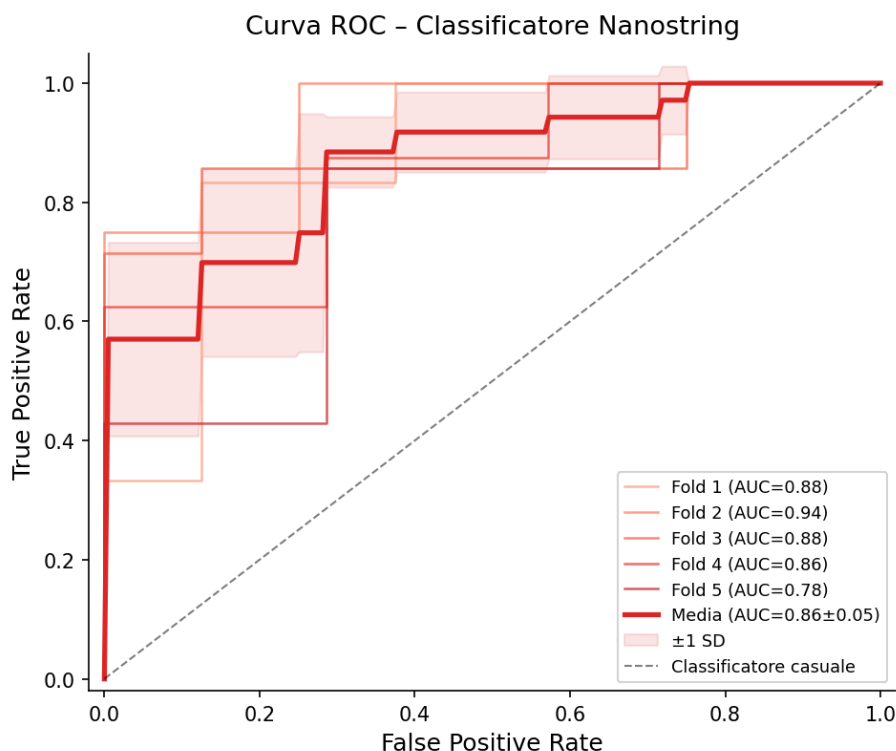


Figura 3.6: Curve ROC per ciascuna partizione di test del classificatore Nanostring, con curva media (linea continua) e banda ± 1 SD (area ombreggiata). La linea tratteggiata rappresenta il classificatore casuale ($AUC = 0.5$).

3.5 Sintesi dei risultati

In sintesi, entrambi i classificatori Random Forest hanno dimostrato capacità discriminative superiori al caso ($AUC > 0.8$ in entrambi i casi) nella classificazione del sottotipo molecolare del DLBCL a partire da feature estratte da maschere nucleari di immagini H&E. Le prestazioni principali dei due modelli sono riepilogate di seguito:

- Il classificatore **Hans** ha raggiunto un'accuracy media di 0.742 ± 0.058 , una sensitivity media di 0.905 ± 0.096 , una specificity media di 0.380 ± 0.228 e un'AUC media di 0.835 ± 0.147 .
- Il classificatore **Nanostring** ha raggiunto un'accuracy media di 0.784 ± 0.053 , una sensitivity media di 0.835 ± 0.05 , una specificity media di 0.732 ± 0.14 e un'AUC media di 0.864 ± 0.058 .
- Il classificatore Nanostring mostra prestazioni più bilanciate tra sensitivity e specificity, nonché una minore variabilità inter-fold rispetto al classificatore Hans.
- Il classificatore Hans evidenzia una tendenza sistematica verso la classe positiva (ABC), attribuibile allo sbilanciamento del dataset di addestramento.

Capitolo 4

Discussione e conclusioni

In questo capitolo vengono discussi i risultati ottenuti dai due classificatori Random Forest per la sottotipizzazione ABC/GCB del DLBCL a partire da feature hand-crafted estratte da maschere nucleari di immagini istologiche H&E. I risultati sono interpretati alla luce delle proprietà intrinseche dei due sistemi di etichettatura utilizzati, confrontati con le prestazioni riportate in letteratura per metodi analoghi e alternativi, e inseriti nel contesto clinico e metodologico più ampio.

4.1 Confronto tra i due classificatori

Un confronto diretto delle metriche medie dei due classificatori è riportato in Tabella 4.1 e visualizzato in Figura 4.1.

Tabella 4.1: Confronto delle metriche di classificazione (media $\pm$ SD su 5 fold) tra il classificatore Hans e il classificatore Nanostring. In grassetto il valore migliore per ciascuna metrica.

Metrica	Hans RF	Nanostring RF
Accuracy	0.742 $\pm$ 0.058	0.784 $\pm$ 0.053
Sensitivity	0.905 $\pm$ 0.096	0.835 $\pm$ 0.050
Specificity	0.380 $\pm$ 0.228	0.732 $\pm$ 0.140
PPV	0.769 $\pm$ 0.054	0.761 $\pm$ 0.082
NPV	0.688 $\pm$ 0.239	0.824 $\pm$ 0.037
F1-score	0.828 $\pm$ 0.042	0.792 $\pm$ 0.031
AUC	0.835 $\pm$ 0.147	0.864 $\pm$ 0.058

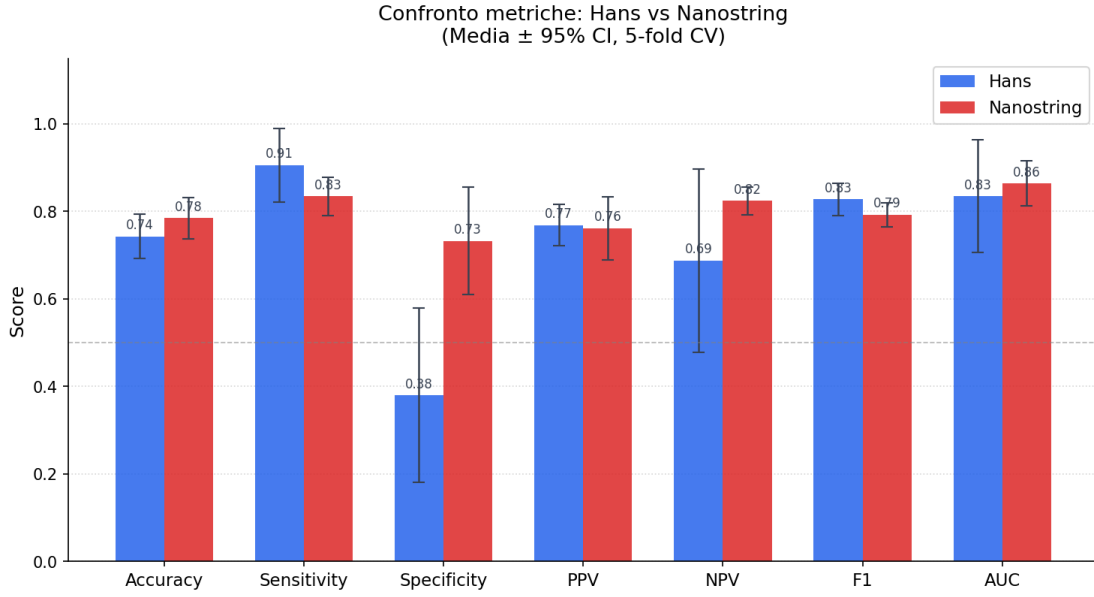


Figura 4.1: Confronto delle metriche di classificazione tra il classificatore Hans (blu) e il classificatore Nanostring (rosso). Le barre rappresentano la media su 5 fold; le barre d'errore rappresentano il 95% CI ($1.96 \times SD/\sqrt{5}$).

Il classificatore Nanostring supera il classificatore Hans in accuracy (+0.042), specificity (+0.352), NPV (+0.136) e AUC (+0.029). Il classificatore Hans presenta una sensitivity media superiore (0.905 contro 0.835) e un F1-score leggermente più alto (0.828 contro 0.792), sebbene quest'ultimo valore possa essere influenzato dallo sbilanciamento delle classi. Il vantaggio più marcato del classificatore Nanostring riguarda la specificità (+0.352) e la stabilità delle prestazioni tra le partizioni, evidenziata da deviazioni standard tendenzialmente inferiori.

4.1.1 Andamento delle metriche per partizione

L'andamento delle principali metriche al variare della partizione di test è illustrato in Figura 4.2.

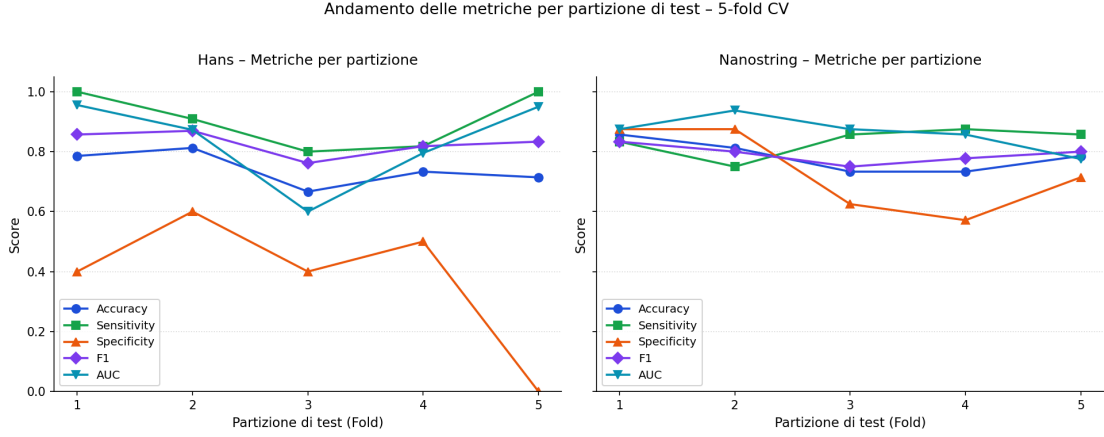


Figura 4.2: Andamento delle metriche di classificazione per ciascuna partizione di test (sinistra: classificatore Hans; destra: classificatore Nanostring).

Nel classificatore Hans, il fold 3 si distingue come la partizione più critica, con accuracy (0.667), AUC (0.600) e F1-score (0.762) sensibilmente inferiori rispetto alle altre partizioni, mentre la specificity mostra la maggiore variabilità inter-fold (range: 0.000–0.600). Nel classificatore Nanostring non si osservano fold particolarmente critici: le metriche mostrano una distribuzione più uniforme, con l'accuracy che varia tra 0.733 e 0.857 e la sensitivity nel range 0.750–0.875. La deviazione standard dell'F1-score (± 0.031) è la più bassa tra tutte le metriche e tra i due classificatori, confermando la maggiore stabilità del classificatore Nanostring tra le partizioni.

4.1.2 Distribuzione delle probabilità predette

La Figura 4.3 mostra la distribuzione delle probabilità predette dal modello per la classe ABC, separatamente per i campioni appartenenti alla classe reale ABC (positivi) e alla classe reale GCB (negativi), aggregando le predizioni su tutte le partizioni di test.

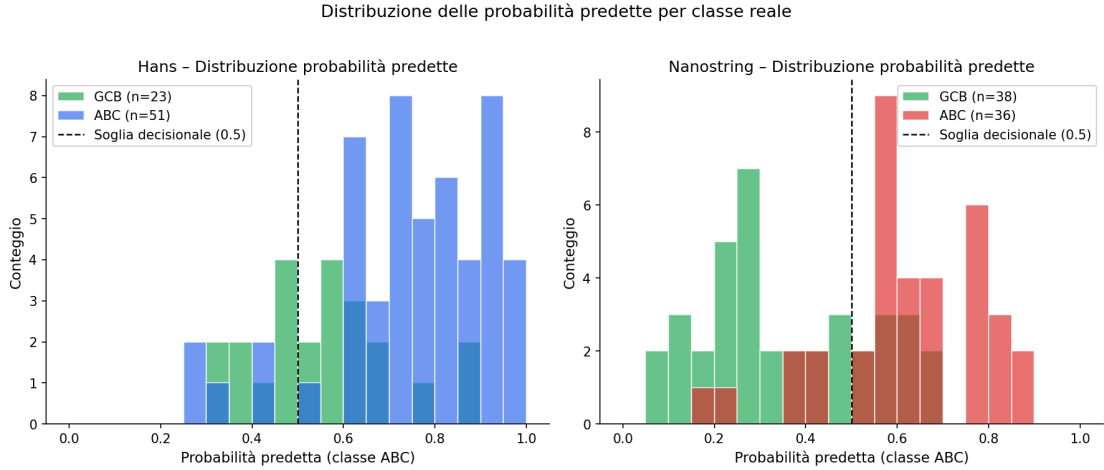


Figura 4.3: Distribuzione delle probabilità predette di appartenenza alla classe ABC, separata per classe reale (verde: GCB; blu/rosso: ABC), per il classificatore Hans (sinistra) e Nanostring (destra). La linea tratteggiata verticale indica la soglia decisionale a 0.5.

Nel classificatore Hans, i campioni ABC ($n = 51$) mostrano una distribuzione delle probabilità predette spostata verso valori elevati, con la maggioranza dei campioni che supera la soglia decisionale di 0.5. I campioni GCB ($n = 23$) presentano invece una distribuzione più dispersa e con sovrapposizione non trascurabile nella regione superiore alla soglia, coerentemente con l'elevato numero di falsi positivi ($n = 14$) osservato nella confusion matrix aggregata.

Nel classificatore Nanostring, le distribuzioni delle due classi appaiono meglio separate. I campioni ABC ($n = 36$) tendono a concentrarsi nella regione a probabilità elevata, mentre i campioni GCB ($n = 38$) mostrano una distribuzione spostata verso valori inferiori a 0.5, con una sovrapposizione limitata nella zona di confine. Questo pattern è coerente con i valori più equilibrati di sensitivity (0.835) e specificity (0.732) osservati per questo classificatore.

4.2 Interpretazione in relazione al sistema di etichettatura

I due classificatori sviluppati differiscono esclusivamente nel sistema di riferimento utilizzato per l'assegnazione delle etichette di classe. Questo disegno sperimentale consente di isolare l'effetto del ground truth e della distribuzione delle classi nel dataset sulla capacità classificativa del modello.

Il classificatore Nanostring ha mostrato prestazioni sistematicamente più bilanciate e stabili rispetto al classificatore Hans. La distribuzione delle probabilità predette

(Figura 4.3) conferma questa interpretazione: nel classificatore Hans i campioni GCB mostrano una distribuzione dispersa con sovrapposizione nella regione superiore alla soglia decisionale di 0.5, mentre nel classificatore Nanostring le due distribuzioni sono meglio separate. Il pattern di alta sensitivity e bassa specificity del classificatore Hans è coerente con lo sbilanciamento del dataset (51 ABC vs 23 GCB, rapporto 2.2:1): dal momento che, in presenza di classi sbilanciate, i modelli Random Forest tendono a privilegiare la classe maggioritaria [44].

Questi risultati sono inoltre coerenti con quanto riportato in letteratura sulla concordanza tra la classificazione IHC secondo Hans e i metodi molecolari. Studi che utilizzano come riferimento la GEP Lymph2Cx riportano una concordanza complessiva dell'algoritmo di Hans del 72%, con una concordanza dell'85% per i casi GCB e solo del 58% per i casi non-GCB [11]. La misclassificazione intrinseca dell'algoritmo di Hans introduce quindi del rumore nelle etichette di addestramento, che si traduce in un modello meno accurato e meno stabile. Al contrario, il Lymph2Cx NanoString, sviluppato da Scott et al. su campioni FFPE con una misclassificazione del 2% rispetto alla GEP su tessuto fresco [21], fornisce un ground truth molto più affidabile, il che si riflette in un modello con prestazioni più equilibrate e riproducibili.

4.3 Confronto con la letteratura

I risultati ottenuti possono essere posti a confronto con quelli riportati da altri sistemi di classificazione automatica del DLBCL basati su immagini istologiche H&E, che rappresentano il contesto metodologico più direttamente comparabile con il presente lavoro. È tuttavia necessario segnalare una limitazione metodologica che impone cautela nell'interpretazione delle prestazioni riportate: a causa della ridotta numerosità della coorte (74 pazienti), la feature selection è stata eseguita sull'intero dataset anziché esclusivamente sul training set di ciascun fold, come richiederebbe una procedura di validazione incrociata rigorosa. Questa scelta introduce un rischio di data leakage, in quanto le feature selezionate possono riflettere parzialmente le caratteristiche dei campioni di test, portando a una sovrastima delle prestazioni reali del modello. Le metriche riportate devono quindi essere considerate come stime ottimistiche, e il confronto con i risultati della letteratura deve tenere conto di questa differenza procedurale.

Gupta et al. [24] hanno sviluppato un modello di deep learning basato su un framework di Multiple Instance Learning (MIL) con meccanismo di attenzione, utilizzando encoder basati su reti convoluzionali (ConvNeXT_Tiny) e Transformer preaddestrati su immagini istologiche. Il modello ha raggiunto un'AUC media di $87.4 \pm 5.7\%$ in una validazione incrociata a tre fold. Rispetto al classificatore Nanostring sviluppato nel presente lavoro ($AUC = 0.864 \pm 0.058$), le prestazioni

sono comparabili, nonostante la sostanziale differenza di complessità architetture: il modello di Gupta et al. si avvale di encoder profondi preaddestrati su grandi dataset e di un meccanismo di aggregazione a livello di slide, mentre il presente lavoro si basa esclusivamente su feature morfologiche e spaziali handcrafted estratte da maschere nucleari, senza alcun addestramento di reti neurali. Lo stesso studio riporta, come termine di confronto, che Yuce et al. avevano ottenuto con un modello di deep learning cell-based un'AUC di 0.73 sul test set e di 0.67 sul holdout set [47]. Questo suggerisce che la segmentazione nucleare, l'estrazione di feature a livello di singola cellula e la classificazione mediante machine learning possano fornire un segnale discriminativo rilevante, anche a confronto con approcci deep learning di maggiore complessità computazionale.

Il confronto con modelli basati su dati molecolari deve invece essere interpretato diversamente, poiché questi operano su una modalità di dati intrinsecamente più informativa rispetto alle immagini H&E. Carreras et al. [23], utilizzando reti neurali artificiali su un pannello di 730 geni (nCounter), hanno riportato un'AUC di 1.0 per la predizione del sottotipo molecolare secondo Lymph2Cx. Questi risultati confermano che le informazioni molecolari restano il gold standard per la classificazione COO e che i metodi basati su immagini H&E, indipendentemente dall'architettura utilizzata, affrontano un problema intrinsecamente più difficile per la sottilità delle differenze morfologiche tra i due sottotipi [24]. Tuttavia, a differenza dei test molecolari, le immagini istologiche H&E sono parte integrante della pratica patologica standard: le sezioni colorate con ematossilina-eosina vengono prodotte routinariamente per ogni biopsia e sono già disponibili in forma digitalizzata in un numero crescente di centri grazie alla diffusione degli scanner per whole slide imaging. I test molecolari richiedono invece protocolli aggiuntivi, infrastrutture specializzate e costi significativamente superiori, che ne limitano la disponibilità soprattutto in contesti con risorse limitate [24]. Un sistema di classificazione automatica basato su immagini H&E potrebbe quindi avere un impatto pratico rilevante come strumento di supporto diagnostico, in particolare laddove l'accesso alla profilazione molecolare non sia garantito.

In questo contesto, i valori di AUC ottenuti nel presente lavoro (0.835 per Hans e 0.864 per Nanostring) risultano in linea o superiori a quelli riportati da altri metodi computazionali su H&E per lo stesso task, pur con le importanti differenze metodologiche di cui si deve tenere conto nella comparazione (dimensione dei dataset, strategia di validazione, sistema di etichettatura).

4.4 Considerazioni metodologiche: handcrafted features vs. deep learning

L’approccio adottato nel presente lavoro, basato sull’estrazione di feature handcrafted da maschere nucleari, si distingue concettualmente dai metodi di deep learning end-to-end che operano direttamente sui pixel delle immagini. Come discusso da Rundo e Militello [25], i due approcci presentano vantaggi e limiti complementari. Le feature handcrafted offrono piena interpretabilità biologica: ogni feature corrisponde a una misura morfologica, texturale o spaziale definita e può essere ricondotta a caratteristiche biologicamente significative. Al contrario, i modelli di deep learning operano su rappresentazioni latenti difficilmente interpretabili, anche quando si utilizzano meccanismi di attenzione per localizzare le regioni discriminative. In ambito clinico, l’interpretabilità del modello costituisce un requisito non secondario, sia per la validazione scientifica dei risultati che per la fiducia del clinico nello strumento.

Un ulteriore vantaggio dell’approccio handcrafted è la ridotta dipendenza dalla dimensione del dataset di addestramento: i modelli Random Forest su feature estratte manualmente possono produrre risultati significativi anche con coorti di dimensioni limitate come quella utilizzata nel presente lavoro (74 pazienti), mentre i modelli di deep learning richiedono tipicamente dataset molto più ampi per evitare l’overfitting. Il dataset di Gupta et al. [24], ad esempio, include 115 pazienti.

D’altra parte, le feature handcrafted sono dipendenti dalla qualità della segmentazione nucleare a monte: errori sistematici nella segmentazione delle maschere si propagano direttamente alle feature estratte e quindi al classificatore. I modelli di deep learning su patch o su WSI, operando direttamente sui pixel, bypassano questo passaggio, pur introducendo altri tipi di bias legati alla scelta dell’architettura e alla strategia di pre-addestramento.

4.5 Considerazioni sui metodi di aggregazione delle feature

Un ulteriore aspetto rilevante emerso dall’analisi delle feature selezionate riguarda i metodi di aggregazione utilizzati per sintetizzare le distribuzioni di feature calcolate a livello di singolo nucleo o di singola ROI in un valore rappresentativo a livello di paziente. Oltre ai metodi statistici classici (media, deviazione standard e percentili), esclusivamente per le feature calcolate a livello di nucleo, sono stati impiegati due metodi di aggregazione sperimentali: il Bag of Words (BoW), ed il Cell Feature Level Co-occurrence Matrix, (CFLCM).

Le 20 feature selezionate per ciascun classificatore sono riportate in Tabella 4.2 (classificatore Hans) e in Tabella 4.3 (classificatore Nanostring), in grassetto le

feature calcolate a livello di singolo nucleo, le uniche per cui i metodi di aggregazione sperimentali BoW e CFLCM risultano applicabili; le restanti feature sono calcolate a livello di ROI o direttamente di paziente.

Tabella 4.2: Feature selezionate per il classificatore Hans, in ordine di selezione. In grassetto le feature a livello di singolo nucleo.

Ordine	Feature
1	pearson_thr50_w60_p50
2	closeness_cut_circle_diff_entropy
3	pearson_thr50_w60
4	maximum_area_of_a_cluster_p10
5	total_cluster_area_p10
6	skew_edge_length_full_std
7	skew_edge_length_cut_nearest_diff_entropy
8	inter_cluster_minimum_MST_distance_p50
9	inter_cluster_minimum_MST_distance_p75
10	minimum_area_of_a_cluster_p10
11	closeness_full_lattice_contrast
12	closeness_cut_circle_correlation
13	inter_cluster_centroid_to_centroid_total_distance_p90
14	morisita_thr50_w60_std
15	inter_cluster_minimum_MST_distance_p25
16	pearson_thr50_w60_p10
17	inter_cluster_average_MST_distance_std
18	mean_edge_length_cut_BoW_entropy
19	eccentricity_full_BoW_entropy
20	closeness_cut_lattice_contrast

Tabella 4.3: Feature selezionate per il classificatore Nanostring, in ordine di selezione. In grassetto le feature a livello di singolo nucleo.

Ordine	Feature
1	pearson_thr50_w70_std
2	total_cluster_area_p10
3	pearson_thr60_w70
4	mean_edge_length_full_nearest_correlation
5	inter_cluster_minimum_MST_distance_p50
6	inter_cluster_average_MST_distance_std
7	pearson_thr60_w70_p75
8	mean_edge_length_full_circle_correlation
9	area_micron_p10
10	skew_edge_length_full_std
11	skew_edge_length_cut_lattice_contrast
12	closeness_cut_std
13	inter_cluster_minimum_MST_distance_mean
14	average_intra_cluster_nucleus_to_nucleus_minimum_distance_p10
15	morisita_thr60_w70_mean
16	skew_edge_length_cut_BoW_entropy
17	total_intra_cluster_nucleus_to_nucleus_minimum_distance_std
18	betweenness_full_circle_contrast
19	inter_cluster_centroid_to_centroid_total_distance_std
20	degree_cut_nearest_ASM

Tra le 20 feature selezionate per il classificatore Hans, 5 sono aggregate tramite CFLCM e 2 tramite BoW, per un totale di 7 feature (35%) ottenute con metodi sperimentali, contro 12 feature (60%) aggregate con i metodi statistici classici. Per il classificatore Nanostring, la proporzione è analoga: 5 feature CFLCM/Haralick e 1 feature BoW, per un totale di 6 feature (30%) sperimentali contro 13 feature (65%) classiche.

Questo confronto, tuttavia, include nel denominatore anche feature calcolate a livello di paziente o di ROI, per le quali i metodi BoW e CFLCM non sono applicabili. Un confronto più informativo si ottiene quindi restringendo l'analisi alle sole feature a livello nucleare. Per il classificatore Hans, 8 delle 20 feature selezionate sono calcolate a livello di nucleo, e, di queste, 7 (87.5%) sono aggregate tramite i metodi sperimentali, contro 1 sola (12.5%) aggregata con metodi classici (**skew_edge_length_full_std**). Per il classificatore Nanostring, anch'esso con 8 feature a livello nucleare su 20 selezionate, 5 (62.5%) sono aggregate tramite metodi sperimentali e 3 (37.5%) tramite metodi classici (**area_micron_p10**, **skew_edge_length_full_std** e **closeness_cut_std**).

Ristretta al solo livello nucleare, quindi, la proporzione di feature aggregate con metodi sperimentali risulta nettamente superiore rispetto a quella calcolata sul totale delle feature selezionate, e nel caso del classificatore Hans i metodi BoW e CFLCM risultano addirittura predominanti rispetto ai metodi classici. Questo risultato indica che, quando la feature selection può scegliere liberamente tra più modalità di aggregazione per una stessa distribuzione a livello di nucleo, tende a privilegiare i metodi sperimentali, suggerendo che l'informazione catturata da BoW e CFLCM sia più discriminativa, in questo contesto, rispetto alle statistiche classiche.

Questi risultati indicano che i metodi di aggregazione sperimentali, pur essendo numericamente meno rappresentati rispetto ai metodi classici se considerati sul totale delle feature selezionate, contribuiscono alla capacità predittiva del modello con un'importanza comparabile, e a livello nucleare con una frequenza di selezione superiore, rispetto a quella delle feature aggregate con media, deviazione standard e percentili. Il fatto che vengano selezionate feature BoW e diverse feature CFLCM in entrambi i classificatori, in proporzione elevata tra le feature nucleari, suggerisce che l'informazione catturata da questi metodi non sia ridondante rispetto a quella delle statistiche descrittive classiche, ma aggiunga un contributo discriminativo complementare, verosimilmente legato alla struttura spaziale e all'eterogeneità locale della distribuzione delle feature nucleari, aspetti che le statistiche classiche non sono in grado di catturare.

4.6 Conclusioni

Il presente lavoro ha riscontrato, in classificatori automatici basati su feature morfologiche e spaziali handcrafted estratte da immagini istologiche H&E, una discreta capacità di discriminazione tra i sottotipi molecolari ABC e GCB del DLBCL, senza ricorrere ad analisi molecolari o a modelli di deep learning.

Entrambi i classificatori Random Forest sviluppati hanno prodotto risultati superiori al caso, con AUC media superiore a 0.8 in entrambi i casi. Il classificatore addestrato con le label molecolari NanoString ha mostrato prestazioni più bilanciate e stabili (accuracy 0.784, specificity 0.732, AUC 0.864) rispetto al classificatore addestrato con le etichette IHC secondo Hans (accuracy 0.742, sensitivity 0.905, specificity 0.380, AUC 0.835). La differenza di prestazioni tra i due classificatori può essere attribuita in larga misura al forte sbilanciamento delle classi nel dataset di partenza per quanto riguarda la classificazione IHC di Hans ed in secondo luogo alla migliore qualità del ground truth molecolare (NanoString) rispetto a quello immunoistochimico (Hans).

Un ulteriore risultato di interesse riguarda i metodi di aggregazione delle feature nucleari: la feature selection ha privilegiato, tra le feature a livello di nucleo, quelle

aggregate tramite i metodi sperimentali CFLCM e BoW rispetto alle statistiche classiche, con una proporzione dell'87.5% per il classificatore Hans e del 62.5% per il classificatore Nanostring. Questo suggerisce che tali metodi catturino un'informazione spaziale complementare rispetto alle sole statistiche descrittive di primo e secondo ordine, e che il loro impiego nell'aggregazione delle feature nucleari costituisca un contributo metodologico rilevante del presente lavoro.

Questi risultati suggeriscono che le informazioni morfologiche e spaziali a livello nucleare contenute nelle immagini H&E siano parzialmente informative rispetto al sottotipo molecolare del DLBCL, e che un approccio basato su feature handcrafted possa costituire una alternativa interpretabile ed efficiente ai metodi di deep learning, in particolare in contesti con dataset di dimensioni ridotte.

Capitolo 5

Limitazioni e sviluppi futuri

In questo capitolo vengono discusse le principali limitazioni metodologiche del presente lavoro, sia quelle ereditate dalle fasi di acquisizione e di segmentazione delle immagini istologiche, sia quelle introdotte nelle fasi di aggregazione delle feature, di feature selection e di validazione dei modelli. Vengono inoltre proposte alcune direzioni di sviluppo futuro volte a superare tali limitazioni e a estendere la metodologia adottata.

5.1 Limitazioni

5.1.1 Sbilanciamento del dataset

Il dataset utilizzato per il classificatore Hans presenta uno sbilanciamento marcato tra le classi (51 pazienti ABC/NON-GCB contro 23 GCB, rapporto 2.2:1), che ha avuto un impatto diretto sulle prestazioni del modello, in particolare sulla specificity media (0.380 ± 0.228), risultata sistematicamente più bassa e più variabile rispetto alla sensitivity (0.905 ± 0.096). Come discusso nel capitolo precedente, in presenza di classi sbilanciate i modelli Random Forest tendono a privilegiare la classe maggioritaria, penalizzando il riconoscimento della classe minoritaria. Sebbene il dataset del classificatore Nanostring sia sostanzialmente bilanciato (36 ABC vs 38 GCB), la dimensione complessiva della coorte (74 pazienti) resta comunque ridotta, il che limita ulteriormente la possibilità di costituire sottocampioni di addestramento e validazione realmente rappresentativi della variabilità della popolazione clinica.

5.1.2 Limitazioni legate ai metodi di labelling

Le etichette di classe utilizzate per l'addestramento dei due classificatori derivano da sistemi di riferimento con un grado di affidabilità intrinsecamente diverso. La classificazione immunoistochimica secondo Hans, pur essendo ampiamente adottata

nella pratica clinica per la sua semplicità di esecuzione, presenta una concordanza limitata con il gold standard molecolare: studi che utilizzano come riferimento la GEP Lymph2Cx riportano una concordanza complessiva del 72%, con una concordanza dell'85% per i casi GCB ma di solo il 58% per i casi non-GCB [11]. Questo significa che una parte non trascurabile delle etichette utilizzate per addestrare il classificatore Hans può essere errata rispetto al reale sottotipo molecolare del paziente, introducendo rumore nel processo di addestramento e limitando la capacità del modello di apprendere un'associazione affidabile tra le feature morfologiche e spaziali estratte e il sottotipo biologico. Anche le etichette NanoString, sebbene caratterizzate da una concordanza molto più elevata con la GEP completa (misclassificazione del 2% secondo Scott et al. [21]), non sono comunque esenti da un margine di incertezza residua, che si ripercuote inevitabilmente sulla qualità del ground truth utilizzato nel presente lavoro.

5.1.3 Limitazioni della rete di segmentazione nucleare

Le maschere nucleari utilizzate come base per l'estrazione delle feature sono state prodotte da una rete di segmentazione sviluppata in una fase precedente nell'ambito del progetto Dipat3Care, basata su un'architettura con backbone Swin Transformer Tiny e formulata come task di segmentazione a tre classi (sfondo, bordo del nucleo, interno del nucleo), con l'introduzione della classe bordo specificamente volta a favorire la separazione di nuclei adiacenti o parzialmente sovrapposti. Nonostante questo accorgimento architetturale, la segmentazione prodotta non è esente da errori: in particolare, nuclei fortemente sovrapposti o in stretta adiacenza vengono in alcuni casi erroneamente identificati dalla rete come un singolo nucleo anziché come entità distinte. Questo fenomeno, introduce una distorsione sistematica nelle feature estratte e, di conseguenza, nelle feature aggregate a livello di paziente che da esse derivano. Poiché tale errore di segmentazione non è stato quantificato sistematicamente nel presente lavoro, non è possibile escludere che parte della capacità discriminativa, o della mancanza di essa, osservata nei classificatori sia in parte attribuibile a queste imprecisioni a monte della pipeline.

5.1.4 Costruzione del vocabolario visivo nel metodo BoW

Nel metodo di aggregazione Bag of Words (BoW) utilizzato nel presente lavoro, la costruzione del vocabolario visivo, ossia l'identificazione dei cluster rappresentativi, è stata effettuata utilizzando l'insieme di tutte le ROI disponibili, anziché limitarsi alle sole ROI appartenenti ai pazienti del set di training di ciascuno split della cross-validation. Tuttavia la letteratura sul Bag of Words indica esplicitamente che il vocabolario visivo deve essere costruito a partire dalle sole immagini del training set, per poi essere applicato, senza ulteriori adattamenti, alle immagini del test

set [48]. Costruire il vocabolario sull'intero dataset, includendo quindi anche le ROI dei pazienti che in un determinato fold sarebbero stati assegnati al test set, introduce una forma indiretta di contaminazione tra training e test, poiché i centri dei cluster sono influenzati anche dai dati di test. Questa scelta è stata dettata dalla ridotta numerosità campionaria della coorte e dall'onerosità computazionale che avrebbe comportato la ricostruzione del vocabolario, l'aggregazione delle feature e la feature selection separatamente all'interno di ciascuno dei cinque split della cross-validation.

5.1.5 Feature selection eseguita sull'intero dataset

Per le stesse ragioni esposte al punto precedente, la feature selection è stata eseguita sull'intero dataset, anziché esclusivamente sul set di training di ciascun fold della cross-validation. Questa pratica costituisce una nota fonte di data leakage ampiamente documentata in letteratura: Demircioğlu [40] ha dimostrato empiricamente, su dieci dataset radiomici pubblici, che l'applicazione scorretta della feature selection prima della cross-validation introduce un bias positivo fino a 0.15 in AUC-ROC, 0.29 in AUC-F1 e 0.17 in accuracy rispetto alla procedura corretta, in cui la feature selection viene ripetuta autonomamente all'interno di ciascun fold di addestramento. Le metriche di prestazione riportate nel presente lavoro devono quindi essere considerate come stime potenzialmente ottimistiche delle reali capacità predittive dei modelli, e il confronto con studi che hanno correttamente integrato la feature selection all'interno della procedura di cross-validation deve tenere conto di questa differenza metodologica.

5.2 Sviluppi futuri

A partire dalle limitazioni discusse, è possibile delineare diverse direzioni di sviluppo per lavori futuri.

Una prima direzione riguarda la correzione delle problematiche metodologiche relative alla costruzione del vocabolario BoW e alla feature selection, integrando entrambe le fasi all'interno di ciascuno split della cross-validation, in modo da eliminare la contaminazione tra training e test e ottenere stime di prestazione non distorte. Questo richiederebbe, in linea di principio, un dataset di dimensioni maggiori rispetto a quello attualmente disponibile, per garantire una numerosità sufficiente in ciascun fold di training anche dopo l'esclusione dei dati di test.

Una seconda direzione consiste nell'esplorazione sistematica dei parametri dei metodi di aggregazione sperimentali. Per il metodo CFLCM, possono essere indagati l'effetto della dimensione del raggio nel criterio di selezione del vicinato *circle*, della dimensione del lato della finestra nel criterio *lattice*, nonché il numero di livelli utilizzati per la discretizzazione delle feature continue su cui vengono poi calcolate

le metriche di Haralick. Per il metodo BoW, è opportuno indagare l'effetto del numero di cluster (dimensione del vocabolario visivo) e della dimensione delle patch sulla capacità discriminativa delle feature risultanti. Si potrebbe inoltre considerare la sperimentazione di metodi alternativi di feature selection e di classificazione, non limitandosi al Random Forest e ai metodi di selezione attualmente impiegati (importanza Gini e SHAP), ma esplorando ad esempio metodi di selezione basati su tecniche di regolarizzazione (LASSO, Elastic Net) o altri classificatori, quali Support Vector Machine, Gradient Boosting o reti neurali, al fine di verificare la robustezza dei risultati rispetto alla scelta dell'algoritmo di apprendimento.

Si potrebbe inoltre valutare l'estensione della metodologia verso approcci di deep learning, qualora si rendesse disponibile una coorte di dimensioni sufficientemente ampie da consentire l'addestramento di modelli a maggiore capacità parametrica senza incorrere in overfitting. Come discusso nel capitolo precedente, i metodi di deep learning end-to-end, pur presentando una minore interpretabilità rispetto alle feature handcrafted, offrono il vantaggio di apprendere automaticamente rappresentazioni discriminative direttamente dai pixel delle immagini, senza dipendere dalla qualità di un passaggio di segmentazione esplicito a monte, e hanno mostrato in letteratura prestazioni comparabili o superiori a quelle ottenute nel presente lavoro [24].

Un ulteriore possibile sviluppo futuro riguarda l'integrazione di tecniche di explainable AI (XAI) e di quantificazione dell'incertezza, particolarmente rilevanti qualora la metodologia venga estesa verso modelli di deep learning come discusso al punto precedente. Da un lato, tecniche di XAI quali i meccanismi di attenzione tipici dei modelli di Multiple Instance Learning, mappe di salienza (Grad-CAM) o metodi di interpretabilità locale (LIME, SHAP) permetterebbero di identificare quali ROI del paziente abbiano maggiormente contribuito alla predizione del sottotipo molecolare, fornendo al patologo un riscontro visivo direttamente verificabile sulle regioni di tessuto ritenute più informative dal modello.

Dall'altro lato, l'integrazione di tecniche di uncertainty quantification permetterebbe di associare a ciascuna predizione una stima della confidenza del modello, distinguendo i casi in cui la classificazione è affidabile da quelli in cui il modello risulta incerto e per i quali sarebbe opportuno un approfondimento diagnostico aggiuntivo da parte del patologo. Tra le tecniche più diffuse a questo scopo si colloca il Monte Carlo dropout, proposto da Gal e Ghahramani [49], che consente di approssimare l'inferenza bayesiana mantenendo attivo il dropout anche in fase di inferenza ed eseguendo molteplici forward pass stocastici per lo stesso input: la variabilità tra le predizioni ottenute fornisce una stima dell'incertezza epistemica del modello. Tale approccio, applicabile naturalmente a un'estensione del lavoro verso architetture di deep learning, risulterebbe complementare alle tecniche di XAI sopra descritte, fornendo al clinico non solo un'indicazione di quali regioni abbiano contribuito alla predizione, ma anche una misura della affidabilità della

predizione stessa.

Un'ulteriore direzione di sviluppo riguarda l'ampliamento del set di feature considerate. Il presente lavoro si è focalizzato prevalentemente su feature spaziali derivate dall'organizzazione dei nuclei nel tessuto (MST, clustering, co-localizzazione). In prospettiva futura sarebbe di interesse integrare sistematicamente anche feature morfologiche a livello di singolo nucleo — quali forma, elongazione, irregolarità del contorno e compattezza — e feature di texture intranucleari, calcolate ad esempio tramite GLCM direttamente sull'intensità dei pixel all'interno di ciascun nucleo. L'integrazione di tali descrittori potrebbe arricchire la rappresentazione delle caratteristiche biologicamente rilevanti dei nuclei tumorali e potenzialmente migliorare la capacità discriminativa del classificatore.

Infine, un ulteriore sviluppo futuro potrebbe riguardare l'estensione della coorte a centri multipli, al fine di valutare la generalizzabilità del modello rispetto alla variabilità di acquisizione e colorazione tipica di dataset multi-istituzionali.

Dichiarazione sull'utilizzo di strumenti di intelligenza artificiale generativa

Nel corso della redazione della presente tesi magistrale sono stati impiegati strumenti di intelligenza artificiale generativa al fine di migliorare la qualità linguistica del testo e garantire uno stile più coerente con il registro accademico richiesto. In particolare, ChatGPT (OpenAI) e Claude (Anthropic) sono stati utilizzati per migliorare il linguaggio formale e scientifico, per supportare la spiegazione di concetti complessi e per la generazione di rappresentazioni grafiche esteticamente curate a partire dai risultati ottenuti nel presente lavoro.

Tali strumenti non sono stati impiegati per generare contenuto tecnico o scientifico originale, né per sostituire il ragionamento critico necessario allo sviluppo dell'argomentazione della tesi. La scelta delle feature estratte e dei metodi di aggregazione adottati è frutto di un'autonoma e approfondita ricerca bibliografica condotta sulle fonti scientifiche elencate nella sezione Riferimenti bibliografici. L'implementazione della pipeline, le scelte metodologiche applicate, la valutazione e la discussione dei risultati e delle relative limitazioni, nonché le proposte di sviluppi futuri, costituiscono elaborazioni originali e autonome.

Tutto il contenuto tecnico e scientifico della presente tesi è stato sviluppato autonomamente, sulla base di fonti accademiche affidabili quali articoli scientifici e testi di riferimento, elencati nella sezione Riferimenti bibliografici.

Bibliografia

- [1] Rita Alaggio et al. «The 5th edition of the World Health Organization Classification of Haematolymphoid Tumours: Lymphoid Neoplasms». In: *Leukemia* 36.7 (2022), pp. 1720–1748. ISSN: 0887-6924. DOI: 10.1038/s41375-022-01620-2. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC9214472/> (visitato il 17/05/2026) (cit. alle pp. 1–3).
- [2] Elias Campo et al. «The International Consensus Classification of Mature Lymphoid Neoplasms: a report from the Clinical Advisory Committee». In: *Blood* 140.11 (set. 2022), pp. 1229–1253. ISSN: 0006-4971. DOI: 10.1182/blood.2022015851. URL: <https://doi.org/10.1182/blood.2022015851> (visitato il 14/05/2026) (cit. alle pp. 1, 2).
- [3] *Linfoma di Hodgkin - Oncologia*. it. URL: <https://www.msmanuals.com/it/professionale/oncologia/linfomi/linfoma-di-hodgkin> (visitato il 17/05/2026) (cit. alle pp. 1, 2).
- [4] *Panoramica sui linfomi - Disturbi del sangue*. it. URL: <https://www.msmanuals.com/it/casa/disturbi-del-sangue/linfomi/panoramica-sui-linfomi> (visitato il 17/05/2026) (cit. a p. 1).
- [5] Bruce D. Cheson, Richard I. Fisher, Sally F. Barrington, Franco Cavalli, Lawrence H. Schwartz, Emanuele Zucca e T. Andrew Lister. «Recommendations for Initial Evaluation, Staging, and Response Assessment of Hodgkin and Non-Hodgkin Lymphoma: The Lugano Classification». In: *J Clin Oncol* 32.27 (set. 2014), pp. 3059–3067. ISSN: 0732-183X. DOI: 10.1200/JCO.2013.54.8800. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC4979083/> (visitato il 17/05/2026) (cit. alle pp. 1, 2).
- [6] Rebecca L. Siegel, Kimberly D. Miller, Nikita Sandeep Wagle e Ahmedin Jemal. «Cancer statistics, 2023». en. In: *CA: A Cancer Journal for Clinicians* 73.1 (2023). _eprint: <https://acsjournals.onlinelibrary.wiley.com/doi/pdf/10.3322/caac.21763> pp. 17–48. ISSN: 1542-4863. DOI: 10.3322/caac.21763. URL: <https://onlinelibrary.wiley.com/doi/abs/10.3322/caac.21763> (visitato il 18/05/2026) (cit. a p. 2).

-
- [7] H. Tilly e M. Dreyling. «Diffuse large B-cell non-Hodgkin's lymphoma: ESMO Clinical Practice Guidelines for diagnosis, treatment and follow-up». In: *Annals of Oncology* 21.Suppl 5 (2010), pp. v172–v174. DOI: 10.1093/annonc/mdq203 (cit. a p. 2).
- [8] *Non-Hodgkin Lymphoma*. URL: <https://www.ncbi.nlm.nih.gov/books/NBK559328/> (visitato il 30/06/2026) (cit. a p. 2).
- [9] *Linfomi non-Hodgkin - Ematologia e oncologia*. it. URL: <https://www.msdmannuals.com/it/professionale/ematologia-e-oncologia/linfomi/linfomi-non-hodgkin> (visitato il 01/05/2026) (cit. a p. 3).
- [10] S. Blenk, J. Engelmann, M. Weniger, J. Schultz, M. Dittrich, A. Rosenwald, H.K. Müller-Hermelink, T. Müller e T. Dandekar. «Germinal Center B Cell-Like (GCB) and Activated B Cell-Like (ABC) Type of Diffuse Large B Cell Lymphoma (DLBCL): Analysis of Molecular Predictors, Signatures, Cell Cycle State and Patient Survival». EN. In: *Cancer Inform* 3 (gen. 2007). Publisher: SAGE Publications Ltd STM, p. 117693510700300004. ISSN: 1176-9351. DOI: 10.1177/117693510700300004. URL: <https://doi.org/10.1177/117693510700300004> (visitato il 02/05/2026) (cit. a p. 3).
- [11] Gonzalo Gutiérrez-García et al. «Gene-expression profiling and not immunophenotypic algorithms predicts prognosis in patients with diffuse large B-cell lymphoma treated with immunochemotherapy». eng. In: *Blood* 117.18 (mag. 2011), pp. 4836–4843. ISSN: 1528-0020. DOI: 10.1182/blood-2010-12-322362 (cit. alle pp. 3, 60, 68).
- [12] Kieron Dunleavy et al. «Differential efficacy of bortezomib plus chemotherapy within molecular subtypes of diffuse large B-cell lymphoma». eng. In: *Blood* 113.24 (giu. 2009), pp. 6069–6076. ISSN: 1528-0020. DOI: 10.1182/blood-2009-01-199679 (cit. a p. 3).
- [13] Grzegorz S Nowakowski, Annalisa Chiappella, Thomas E Witzig, Michele Spina, Randy D Gascoyne, Lei Zhang, Jocelyne Flament, Jacqueline Repici e Umberto Vitolo. «ROBUST: Lenalidomide-R-CHOP versus placebo-R-CHOP in previously untreated ABC-type diffuse large B-cell lymphoma». In: *Future Oncol* 12.13 (lug. 2016), pp. 1553–1563. ISSN: 1479-6694. DOI: 10.2217/fon-2016-0130. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC5551933/> (visitato il 03/05/2026) (cit. a p. 3).
- [14] Wyndham H. Wilson et al. «Targeting B cell receptor signaling with ibrutinib in diffuse large B cell lymphoma». eng. In: *Nat Med* 21.8 (ago. 2015), pp. 922–926. ISSN: 1546-170X. DOI: 10.1038/nm.3884 (cit. a p. 3).
- [15] *Bortezomib Accord / European Medicines Agency (EMA)*. en. Mag. 2018. URL: <https://www.ema.europa.eu/en/medicines/human/EPAR/bortezomib-accord> (visitato il 04/05/2026) (cit. a p. 3).

- [16] Andreas Rosenwald e Louis M Staudt. «Gene Expression Profiling of Diffuse Large B-Cell Lymphoma». In: *Leukemia & Lymphoma* 44.sup3 (nov. 2003). Publisher: Taylor & Francis _eprint: <https://doi.org/10.1080/10428190310001623775> S41–S47. ISSN: 1042-8194. DOI: 10.1080/10428190310001623775. URL: <https://doi.org/10.1080/10428190310001623775> (visitato il 05/05/2026) (cit. a p. 4).
- [17] Steven H. Swerdlow et al. «The 2016 revision of the World Health Organization classification of lymphoid neoplasms». In: *Blood* 127.20 (mag. 2016), pp. 2375–2390. ISSN: 0006-4971. DOI: 10.1182/blood-2016-01-643569. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC4874220/> (visitato il 09/05/2026) (cit. a p. 4).
- [18] Christine P. Hans et al. «Confirmation of the molecular classification of diffuse large B-cell lymphoma by immunohistochemistry using a tissue microarray». en. In: *Blood* 103.1 (gen. 2004). Publisher: American Society of Hematology, pp. 275–282. ISSN: 0006-4971. DOI: 10.1182/blood-2003-05-1545. URL: <https://dx.doi.org/10.1182/blood-2003-05-1545> (visitato il 09/05/2026) (cit. a p. 4).
- [19] Maysaa Abdulla et al. «Cell-of-origin determined by both gene expression profiling and immunohistochemistry is the strongest predictor of survival in patients with diffuse large B-cell lymphoma». In: *American Journal of Hematology* 95.1 (gen. 2020), pp. 57–67. ISSN: 0361-8609. DOI: 10.1002/ajh.25666 (cit. a p. 4).
- [20] Paul N. Meyer et al. «Immunohistochemical Methods for Predicting Cell of Origin and Survival in Patients With Diffuse Large B-Cell Lymphoma Treated With Rituximab». In: *J Clin Oncol* 29.2 (gen. 2011), pp. 200–207. ISSN: 0732-183X. DOI: 10.1200/JCO.2010.30.0368. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC3058275/> (visitato il 12/05/2026) (cit. a p. 4).
- [21] David W. Scott et al. «Determining cell-of-origin subtypes of diffuse large B-cell lymphoma using gene expression in formalin-fixed paraffin-embedded tissue». In: *Blood* 123.8 (feb. 2014), pp. 1214–1217. ISSN: 0006-4971. DOI: 10.1182/blood-2013-11-536433. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC3931191/> (visitato il 13/05/2026) (cit. alle pp. 5, 60, 68).
- [22] Grzegorz S. Nowakowski, Tatyana Feldman, Lisa M. Rimsza, Jason R. Westin, Thomas E. Witzig e Pier Luigi Zinzani. «Integrating precision medicine through evaluation of cell of origin in treatment planning for diffuse large B-cell lymphoma». In: *Blood Cancer Journal* 9.6 (2019), p. 48. DOI: 10.1038/s41408-019-0208-6 (cit. a p. 5).

- [23] Joaquim Carreras et al. «Artificial Neural Networks Predicted the Overall Survival and Molecular Subtypes of Diffuse Large B-Cell Lymphoma Using a Pancancer Immune-Oncology Panel». en. In: *Cancers* 13.24 (gen. 2021). Publisher: Multidisciplinary Digital Publishing Institute, p. 6384. ISSN: 2072-6694. DOI: 10.3390/cancers13246384. URL: <https://www.mdpi.com/2072-6694/13/24/6384> (visitato il 11/06/2026) (cit. alle pp. 5, 61).
- [24] Ravi Kant Gupta et al. «Classification and Morphological Analysis of Dlbcl Subtypes in H&E-Stained Slides». In: *2024 IEEE 24th International Conference on Bioinformatics and Bioengineering (BIBE)*. ISSN: 2471-7819. Nov. 2024, pp. 1–8. DOI: 10.1109/BIBE63649.2024.10820463. URL: <https://ieeexplore.ieee.org/abstract/document/10820463> (visitato il 11/06/2026) (cit. alle pp. 5, 60–62, 70).
- [25] Leonardo Rundo e Carmelo Militello. «Image biomarkers and explainable AI: handcrafted features versus deep learned features». In: *Eur Radiol Exp* 8 (nov. 2024), p. 130. ISSN: 2509-9280. DOI: 10.1186/s41747-024-00529-y. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC11576747/> (visitato il 12/06/2026) (cit. alle pp. 5, 62).
- [26] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest e Clifford Stein. *Introduction to Algorithms*. 4^a ed. Cambridge, MA: The MIT Press, 2022. ISBN: 9780262046305 (cit. alle pp. 11, 12).
- [27] J. C. Gower e G. J. S. Ross. «Minimum Spanning Trees and Single Linkage Cluster Analysis». In: *Journal of the Royal Statistical Society Series C: Applied Statistics* 18.1 (mar. 1969). _eprint: https://academic.oup.com/jrsssc/article-pdf/18/1/54/48611887/jrsssc_18_1_54.pdf, pp. 54–64. ISSN: 0035-9254. DOI: 10.2307/2346439. URL: <https://doi.org/10.2307/2346439> (cit. a p. 12).
- [28] K. Kayser e H. Stute. «Minimum Spanning Tree - Voronoi's Tessellation - Johnson-Mehl Diagrams - Human Lung Carcinoma». In: *Pathology - Research and Practice* 185.5 (dic. 1989), pp. 729–734. ISSN: 0344-0338. DOI: 10.1016/S0344-0338(89)80228-6. URL: <https://www.sciencedirect.com/science/article/pii/S0344033889802286> (visitato il 18/06/2026) (cit. a p. 12).
- [29] Cemal Cagatay Bilgin, Shayoni Ray, Banu Baydil, William P. Daley, Melinda Larsen e Bülent Yener. «Multiscale Feature Analysis of Salivary Gland Branching Morphogenesis». In: *PLoS One* 7.3 (mar. 2012), e32906. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0032906. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC3293912/> (visitato il 02/07/2026) (cit. a p. 12).
- [30] Andrew Lagree et al. «Assessment of Digital Pathology Imaging Biomarkers Associated with Breast Cancer Histologic Grade». In: *Curr Oncol* 28.6 (ott. 2021), pp. 4298–4316. ISSN: 1198-0052. DOI: 10.3390/curroncol28060366.

- URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC8628688/> (visitato il 02/07/2026) (cit. a p. 15).
- [31] Scott Doyle, Shannon Agner, Anant Madabhushi, Michael Feldman e John Tomaszewski. «Automated grading of breast cancer histopathology using spectral clustering with textural and architectural image features». In: *2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. ISSN: 1945-8452. Mag. 2008, pp. 496–499. DOI: 10.1109/ISBI.2008.4541041. URL: <https://ieeexplore.ieee.org/document/4541041> (visitato il 02/07/2026) (cit. a p. 15).
- [32] Subrata Bhattacharjee, Kobiljon Ikromjanov, Kouayep Sonia Carole, Nuwan Madusanka, Nam-Hoon Cho, Yeong-Byn Hwang, Rashadul Islam Sumon, Hee-Cheol Kim e Heung-Kook Choi. «Cluster Analysis of Cell Nuclei in H&E-Stained Histological Sections of Prostate Cancer and Classification Based on Traditional and Modern Artificial Intelligence Techniques». In: *Diagnostics (Basel)* 12.1 (dic. 2021), p. 15. ISSN: 2075-4418. DOI: 10.3390/diagnostics12010015. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC8774423/> (visitato il 02/07/2026) (cit. a p. 18).
- [33] Laura Ardighieri et al. «Different spatial distribution of inflammatory cells in the tumor microenvironment of ABC and GBC subgroups of diffuse large B cell lymphoma». In: *Frontiers in Oncology* (2021). DOI: 10.3389/fonc.2021.716848 (cit. a p. 21).
- [34] Ayoma D. Attygalle et al. «Spatial analysis of the tumor microenvironment in diffuse large B-cell lymphoma reveals clinically relevant cell interactions and recurrent cellular neighborhoods». In: *Blood Advances* (2022). DOI: 10.1182/bloodadvances.2022007523 (cit. a p. 21).
- [35] Carlo C. Maley, Konrad Koelble, Rachael Natrajan, Athena Aktipis e Yinyin Yuan. «An ecological measure of immune-cancer colocalization as a prognostic factor for breast cancer». en. In: *Breast Cancer Res* 17.1 (set. 2015), p. 131. ISSN: 1465-542X. DOI: 10.1186/s13058-015-0638-4. URL: <https://doi.org/10.1186/s13058-015-0638-4> (visitato il 02/07/2026) (cit. alle pp. 21, 22).
- [36] Robert M. Haralick, Karthikeyan Shanmugam e Its'hak Dinstein. «Textural Features for Image Classification». In: *IEEE Transactions on Systems, Man, and Cybernetics* SMC-3.6 (1973). ISSN: 0018-9472. DOI: 10.1109/TSMC.1973.4309314 (cit. alle pp. 25, 27, 28).
- [37] Akira Saito, Yasushi Numata, Takuya Hamada, Tomoyoshi Horisawa, Eric Cosatto, Hans Peter Graf, Masahiko Kuroda e Yoichiro Yamamoto. «A novel method for morphological pleomorphism and heterogeneity quantitative measurement: Named cell feature level co-occurrence matrix». In: *Journal*

- of Pathology Informatics* 7 (2016), p. 36. DOI: 10.4103/2153-3539.189699 (cit. alle pp. 25, 26).
- [38] Naoto Tokuyama et al. «Prediction of non-muscle invasive bladder cancer recurrence using machine learning of quantitative nuclear features». In: *Modern Pathology* 35 (2022), pp. 533–538. DOI: 10.1038/s41379-021-00955-y (cit. alle pp. 25, 26).
- [39] Reid Trenton Powell, Adriana Olar, Shivali Narang, Ganesh Rao, Erik Sulman, Gregory N. Fuller e Arvind Rao. «Identification of Histological Correlates of Overall Survival in Lower Grade Gliomas Using a Bag-of-words Paradigm: A Preliminary Analysis Based on Hematoxylin & Eosin Stained Slides from the Lower Grade Glioma Cohort of The Cancer Genome Atlas». In: *Journal of Pathology Informatics* 8 (2017), p. 9. DOI: 10.4103/jpi.jpi_43_16 (cit. alle pp. 30, 31).
- [40] Aydin Demirciğlu. «Measuring the bias of incorrect application of feature selection when using cross-validation in radiomics». In: *Insights into Imaging* 12 (2021), p. 172. DOI: 10.1186/s13244-021-01115-1 (cit. alle pp. 39, 69).
- [41] Ron Kohavi. «A study of cross-validation and bootstrap for accuracy estimation and model selection». In: *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* 14 (1995), pp. 1137–1143 (cit. a p. 39).
- [42] Sudhir Varma e Richard Simon. «Bias in error estimation when using cross-validation for model selection». In: *BMC Bioinformatics* 7 (2006), p. 91. DOI: 10.1186/1471-2105-7-91 (cit. a p. 41).
- [43] Leo Breiman. «Random Forests». In: *Machine Learning* 45.1 (2001), pp. 5–32. ISSN: 0885-6125. DOI: 10.1023/A:1010933404324 (cit. alle pp. 42, 43).
- [44] Trevor Hastie, Robert Tibshirani e Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2^a ed. New York: Springer, 2009. DOI: 10.1007/978-0-387-84858-7 (cit. alle pp. 43, 60).
- [45] Pierre Geurts, Damien Ernst e Louis Wehenkel. «Extremely randomized trees». In: *Machine Learning* 63.1 (2006), pp. 3–42. DOI: 10.1007/s10994-006-6226-1 (cit. a p. 43).
- [46] Fabian Pedregosa et al. «Scikit-learn: Machine Learning in Python». In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (cit. a p. 43).

- [47] Anil Yuce, Jacob Gildenblat, Samaneh Abbasi-Sureshjani, Camille Laurent e Konstanty Korski. «PB2377: PREDICTING CELL OF ORIGIN FROM DIGITIZED IMAGES OF HEMATOXYLIN AND EOSIN-STAINED SLIDES OF DIFFUSE LARGE B-CELL LYMPHOMAS USING A CELL-BASED DEEP-LEARNING MODEL». en-US. In: *HemaSphere* 7.S3 (ago. 2023), e5900850. ISSN: 2572-9241. DOI: 10.1097/01.HS9.0000976224.59008.50. URL: https://journals.lww.com/hemasphere/fulltext/2023/08003/pb2377__predicting_cell_of_origin_from_digitized.2230.aspx (visitato il 06/07/2026) (cit. a p. 61).
- [48] Stephen O'Hara e Bruce A. Draper. «Introduction to the Bag of Features Paradigm for Image Classification and Retrieval». In: *arXiv preprint arXiv:1101.3354* (2011) (cit. a p. 69).
- [49] Yarin Gal e Zoubin Ghahramani. «Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning». In: *International Conference on Machine Learning*. 2016, pp. 1050–1059 (cit. a p. 70).