UNIVERSITÀ
POLITECNICA
DELLE MARCHE

Facoltà di Ingegneria
Corso di Laurea in Ingegneria Informatica e dell'Automazione

Un nuovo framework di classificazione ordinale basato su deep learning per la stratificazione della gravità della Covid-19 in ecografie polmonari

A novel ordinal deep learning classification framework for lung ultrasound Covid-19 ranking

Candidato:
Conti Edoardo

Relatore:
Prof. Zingaretti Primo

Correlatori:
Fiorentino Maria Chiara, PhD
Rosati Riccardo, PhD

Anno Accademico 2022-2023

UNIVERSITÀ
POLITECNICA
DELLE MARCHE

Facoltà di Ingegneria
Corso di Laurea in Ingegneria Informatica e dell'Automazione

Un nuovo framework di classificazione ordinale basato su deep learning per la stratificazione della gravità della Covid-19 in ecografie polmonari

A novel ordinal deep learning classification framework for lung ultrasound Covid-19 ranking

Candidato:
Conti Edoardo

Relatore:
Prof. Zingaretti Primo

Correlatori:
Fiorentino Maria Chiara, PhD
Rosati Riccardo, PhD

Anno Accademico 2022-2023

Università Politecnica delle Marche
Facoltà di Ingegneria
Corso di Laurea in Ingegneria Informatica e dell'Automazione
Via Brecce Bianche – 60131 Ancona (AN), Italy

ad Alessia ♡

Ringraziamenti

Vorrei esprimere la mia sincera gratitudine alle persone che mi hanno sostenuto lungo l'intero percorso di studi fino alla realizzazione di questa tesi.

In primo luogo, un caloroso ringraziamento va alla mia famiglia, a mio padre Graziano, mia madre Donatella e mia sorella Debora, per il costante sostegno, interesse ed incoraggiamento. Grazie specialmente per avermi concesso la possibilità di intraprendere questo percorso universitario iniziato oramai diversi anni fa, tutto quello che verrà sarà sicuramente anche merito vostro.

Questo lavoro è stato reso possibile innanzitutto grazie al relatore Prof. Primo Zingaretti, il quale ha offerto l'opportunità di impegnarmi in un progetto perfettamente in sintonia con le mie passioni accademiche. Un sentito riconoscimento va anche ai correlatori, Riccardo e Maria Chiara, il cui prezioso supporto ha elevato costantemente la qualità dell'elaborato, dimostrando una disponibilità invidiabile e contribuendo a rendere il percorso di laurea non solo ricco di competenze, ma anche estremamente piacevole.

La mia immensa gratitudine va ovviamente ai miei amici Lorenzo e Andrian, con i quali ho condiviso anche quest'ultimo percorso accademico. Per di più questa volta anche con l'esperienza di convinvenza, condividendo risate, imprevisti e lunghe sessioni di studio per esami e progetti, creando ricordi, che vi assicuro, rimarranno indelebili nella mia memoria. Quest'esperienza è stata uno dei capitoli più belli della mia vita, che mi lascierà per sempre una nostalgia infinita, grazie ragazzi.

Infine, dedico questo traguardo speciale alla mia fidanzata Alessia, perché nessun altro come lei è stato così essenziale in questo percorso. Durante quei giorni stressanti, già settimane prima degli esami, quando cominciavo a sentirmi sopraffatto, eri sempre lì a faticare per riportarmi alla realtà. Nonostante tutto, non ti sei mai tirata indietro e hai continuato a credere in me. Durante questo percorso ne sono successe tante, con momenti alti ma altri molto bassi, e sono certo che senza di te tutto questo non sarebbe mai stato possibile. Ti amo.

Ancona, Febbraio 2024

Conti Edoardo

Abstract

In the context of medical diagnosis, especially in the Covid-19 pneumonia, medical imaging plays an important role. Lung ultrasound (LUS) comes out as a valuable diagnostic technology for the early detection of pulmonary pathologies. This research focuses on how Deep Learning (DL) can contribute to the automation of medical diagnosis, with a specific emphasis on classifying LUS frames using Convolutional Neural Networks (CNN). The University of Trento provides the ICLUS-DB, a lung ultrasound dataset. It includes a 4-level scoring system reflecting the severity ranking of Covid-19 pneumonia, highlighting the intrinsic ordinal nature of LUS data. This aspect has sparked interest in investigating the possibility of achieving more accurate results by leveraging the ordinal nature of the data through the implementation of specific methodologies to optimize the classification of LUS frames. In contrast, the State-of-the-art is oriented towards solving this problem with nominal classification methods, not penalizing errors between distant classes, a relevant aspect for medical implications. In order to fill this gap in the literature, the main goal is to compare a baseline approach, represented by the ResNet18 CNN, with two distinct ordinal approaches: the Ordinal Binary Decomposition (OBD), based on the decomposition of the ordinal problem into a set of binary tasks, and the Cumulative Link Model (CLM), a probabilistic method to predict the probabilities of groups of contiguous categories. Both ordinal approaches use ResNet18 as a feature extractor and exploit dedicated loss functions, including Quadratic Weighted Kappa (QWK). Furthermore, as traditional stratified holdout methods proved insufficient for obtaining reliable results, a robust evaluation framework was implemented. The design of a dedicated Cross-Validation (CV) procedure for ICLUS-DB is a valuable contribution, addressing the challenges of its nature and yielding generalized and robust results. The results confirm the positive contribution of ordinal approaches, especially in ordinal metrics, highlighting a significant increase in average values across all aspects, particularly in Accuracy 1-Off (up to 96.6%), QWK index (up to 73.1%), and Spearman's coefficient (up to 74.4%). The analysis of ROC curves, confusion matrices, and saliency maps underline the advantage of ordinal approaches in capturing pathological details in LUS frames. The ablation study, conducted on all components of the Deep Neural Network (DNN), provides further insights, demonstrating the effectiveness of the proposed components.

Sommario

Nell'ambito della diagnostica medica, soprattutto nel contesto della polmonite da Covid-19, l'imaging medico riveste un ruolo fondamentale. In particolare, l'ecografia polmonare (LUS) emerge come una valida tecnologia diagnostica per la rivelazione precoce di patologie polmonari. Questa ricerca si concentra su come il Deep Learning (DL) possa contribuire all'automatizzazione della diagnosi medica, focalizzandosi sulla classificazione di frame LUS mediante l'utilizzo di reti neurali convoluzionali (CNN). In questo contesto, l'Università di Trento mette a disposizione il dataset di ecografie polmonari ICLUS-DB. Questo comprende un sistema di scoring a 4 livelli che riflette la gerarchia della gravità delle patologie polmonari, sottolineando la natura ordinale intrinseca dei dati LUS. Questo aspetto ha suscitato l'interesse di indagare sulla possibilità di ottenere risultati più accurati sfruttando la natura ordinale dei dati, attraverso l'implementazione di metodologie specifiche per ottimizzare la classificazione dei frame LUS. Contrariamente, lo stato dell'arte è orientato a risolvere questo problema con metodi di classificazione nominali, non penalizzando l'errore tra classi distanti, un aspetto rilevante per le implicazioni mediche. Nell'ottica di colmare questo gap nella letteratura, l'obiettivo principale è quindi quello di confrontare un approccio di riferimento, rappresentato dalla CNN ResNet18, con due approcci ordinali distinti: l'Ordinal Binary Decomposition (OBD), basato sulla decomposizione del problema ordinale in un insieme di task binari, e il Cumulative Link Model (CLM), un metodo probabilistico per prevedere le probabilità di gruppi di categorie contigue. Entrambi gli approcci ordinali utilizzano ResNet18 come estrattore di feature e sfruttano funzioni di loss dedicate, tra cui il Quadratic Weighted Kappa (QWK). Inoltre, dato che i tradizionali metodi di holdout stratificato non sono risultati sufficienti per ottenere risultati affidabili, è stato implementato un framework robusto per la valutazione. La progettazione di una procedura di Cross-Validation (CV) ad-hoc per ICLUS-DB è un contributo alla letteratura che permette di affrontare le sfide della sua natura, ottenendo risultati generalizzati e robusti. I risultati confermano il contributo positivo degli approcci ordinali, specialmente nelle metriche ordinali, evidenziando un aumento significativo dei valori medi sotto ogni aspetto, nello specifico in Accuracy 1-Off (fino al 96.6%), indice QWK (fino al 73.1%) e coefficiente di Spearman (fino al 74.4%). L'analisi delle curve ROC, delle matrici di confusione e delle mappe di salienza sottolineano il vantaggio degli approcci ordinali nel catturare dettagli patologici nei frame LUS. Lo studio d'ablazione, condotto su tutte le componenti dell'architettura neurale, ha fornito ulteriori insights dimostrando l'efficacia dei componenti proposti.

Indice

Elenco delle figure

Elenco delle tabelle

Capitolo 1

Introduzione

La pandemia globale di Covid-19 ha rapidamente trasformato la salute pubblica mondiale, emergendo come una delle sfide moderne più pressanti [1]. L'ampia diffusione del virus ha portato ad una carenza di capacità di testing e forniture mediche, accentuata dalla bassa sensibilità del test di *Reverse Transcription Polymerase Chain Reaction (RT-PCR)* e da un'elevata incidenza di falsi negativi nella diagnosi di Covid-19 [2]. La tomografia computerizzata del torace (CT) è stata considerata come una potenziale opzione data la sua rapidità di risposta ma presenta purtroppo notevoli svantaggi. In questo contesto, la necessità di identificare alternative valide per la diagnosi era diventata imperativa.

L'ecografia polmonare *(Lung Ultrasound, LUS)* è emersa come una valida tecnica di valutazione dell'impatto della polmonite da Covid-19, specialmente per la sua applicabilità in contesti *Point-of-Care (PoC)* [3]. L'analisi LUS si concentra su quelli che in letteratura vengono definiti artefatti visivi, che nelle ecografie polmonari si presentano principalmente in due forme: orizzontali e verticali [4]. Quest'ultimi sono quelli di maggiore interesse in quanto indicano la formazione di trappole acustiche causate da una deaerazione locale della superficie polmonare, ergo associabile a diverse patologie polmonari [4]. Un protocollo di acquisizione standardizzato, coadiuvato da un approccio semi-quantitativo mediante un sistema di punteggio a quattro livelli, è stato proposto al fine di standardizzare un metodo riproducibile fornendo un approccio globalmente uniforme all'analisi LUS dei pazienti affetti da Covid-19 [5]. Nel contesto di questa standardizzazione, emerge il rilevante contributo del progetto *Italian Covid-19 Lung Ultrasound (ICLUS)*, un'iniziativa guidata anche dagli stessi ricercatori che hanno concepito il protocollo menzionato. Il dataset ICLUS-DB rappresenta quindi una risorsa fondamentale, offrendo dati di ecografie polmonari raccolti seguendo rigorosamente le linee guida stabilite.

L'interpretazione basata su artefatti visivi ha reso l'analisi soggettiva e suscettibile a errori [6]. Perciò il lavoro in questione si colloca in un contesto in cui l'uso di LUS è essenziale ma richiede un'approfondita valutazione. Di conseguenza, con l'intento di agevolare i professionisti del settore in questa analisi, sono state proposte soluzioni che impiegano l'Intelligenza Artificiale *(Artificial Intelligence, AI)* adottando nello specifico tecniche di Apprendimento Profondo *(Deep Learning, DL)* per automatizzare il processo di valutazione fornendo un Sistema di Supporto Decisionale *(Decision*

Support System, DSS) agli addetti ai lavori [7][8]. L'introduzione di tali tecnologie offre opportunità per sviluppi significativi che possono apportare contributi rilevanti alla diagnostica per immagini automatizzata, con la stratificazione della gravità come esempio paradigmatico. La stratificazione della gravità della polmonite da Covid-19 implica una classificazione graduale da lieve a severa. Questa complessità sottolinea l'importanza di assegnare accuratamente i punteggi e di ridurre le misclassificazioni importanti, poiché errori significativi influenzano direttamente la gestione clinica, evitando, ad esempio, la sottovalutazione di casi più gravi.

Il progetto mira quindi a stabilire un approccio metodologico robusto per lo sviluppo di soluzioni basate sul Deep Learning per un task di classificazione ordinale di ecografie pleuro-polmonari. In particolare, il lavoro implementa due approcci ordinali distinti, i quali saranno oggetto di confronto con l'approccio tradizionale nominale. Questo paragone mira a valutare concretamente il contributo pratico e l'efficacia di tali soluzioni nel contesto specifico del problema indicato.

1.1 Medical Imaging per la polmonite da Covid-19

La diagnostica per immagini riveste un ruolo cruciale nella valutazione e diagnosi di patologie mediche. Le tecniche di imaging medico consentono una visualizzazione non invasiva degli organi e dei tessuti interni del corpo, fornendo informazioni dettagliate agli addetti ai lavori. Nel contesto delle patologie polmonari, l'impiego di tecniche di medical imaging è di particolare rilevanza. Le malattie polmonari possono presentare una varietà di caratteristiche che possono essere catturate attraverso diverse modalità di imaging, offrendo così una visione completa e dettagliata dello stato polmonare.

Le principali tecniche di imaging polmonare includono la tomografia computerizzata (CT), la radiografia del torace (CXR) e l'ecografia polmonare (LUS). La tomografia computerizzata fornisce immagini dettagliate a sezioni sottili dei polmoni, rivelando con precisione lesioni e alterazioni anatomiche. La radiografia del torace, più convenzionale, offre una visione più ampia ma meno dettagliata dei polmoni. Entrambe queste tecniche, tuttavia, comportano l'esposizione a radiazioni ionizzanti.

L'ecografia polmonare rappresenta una valida alternativa e sebbene sia stata tradizionalmente utilizzata per la valutazione di patologie acute, la sua applicazione nella diagnosi e gestione della polmonite da Covid-19 è emersa come un'area di crescente interesse negli ultimi anni [4][9][10].

1.1.1 Ecografia polmonare (LUS)

L'ecografia polmonare è una tecnica di imaging medico non invasiva che si basa sull'uso di onde sonore ad alta frequenza per generare immagini in tempo reale della struttura polmonare. Questa metodologia offre numerosi vantaggi, tra cui l'assenza di radiazioni ionizzanti, la possibilità di eseguire esami ripetuti senza rischi significativi, e la sua applicabilità in ambienti di pronto soccorso o in situazioni critiche (POCT).

Questo aspetto è ulteriormente avvalorato da uno studio recente focalizzato sulla tubercolosi polmonare (PTB), il quale dimostra che la LUS rappresenta un valido strumento diagnostico capace di fornire le informazioni diagnostiche desiderabili in modo equivalente a quanto rilevabile con tomografia computerizzata (CT) del torace e radiografia toracica (CXR) [10]. La tecnologia impiegata comprende un trasduttore che emette onde sonore a frequenze ultrasoniche (generalmente nell'intervallo di 3-15 MHz) che penetrano nei tessuti polmonari. Quando queste onde incontrano una superficie in cui ci sono differenze di densità tra i tessuti, una parte di esse viene riflessa e captata dal trasduttore. L'analisi di queste onde riflesse consente la creazione di immagini dettagliate dei polmoni [11]. Per eseguire ecografie polmonari esistono diverse sonde ecografiche, ognuna caratterizzata da specifiche applicazioni e caratteristiche. Le principali si dividono in sonde convesse e lineari (Figura 1.1). Le prime (3,5-5 MHz) offrono una visione panoramica e sono adatte per la valutazione generale dei polmoni, mentre quelle lineari (9-12 MHz) forniscono immagini ad alta risoluzione, ideali per indagare aree specifiche. Tuttavia, la scelta tra le due dipende anche dall'applicazione clinica e dalla conformazione del paziente. Le sonde convesse sono più adatte per pazienti obesi o con difficoltà respiratorie, mentre le sonde lineari, seppur offrendo una visione più dettagliata, possono risultare limitate in determinati contesti [11]. Un elemento essenziale nelle LUS è la linea pleurica, un foglio sottile e iperecogeno che delimita il polmone dai tessuti circostanti. Appare come una linea nitida e riflettente sulla superficie pleurica, posizionandosi tra il rivestimento della pleura parietale e il tessuto polmonare (Figura 1.1).

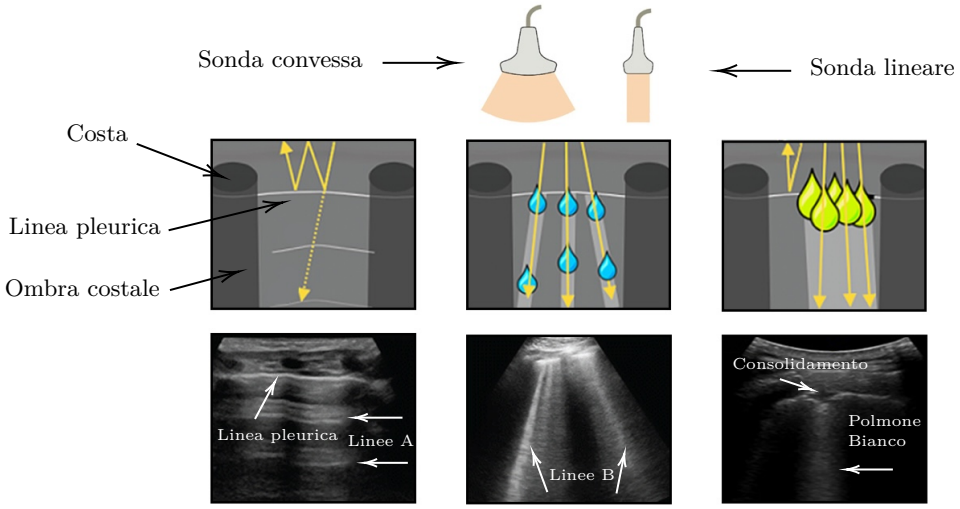


Figura 1.1: Ecografie polmonari: sonde, anatomia e artefatti.

Un altro aspetto critico nella lettura delle ecografie polmonari è la comprensione degli artefatti, dei riverberi che appaioni nelle immagini. Esistono due tipologie di artefatti, quelli orizzontali che sono il risultato delle riflessioni ripetute delle onde ultrasoniche tra la linea pleurica e la superficie della sonda (*Linee A*), indicando

una condizione di polmoni completamente aerati. Quelli verticali invece derivano dalla formazione di trappole acustiche a seguito della deaerazione parziale della superficie polmonare (*Linee B* e *"Polmone Bianco"*), e possono essere associati a l'alterazione della conformazione della linea pluerica, consolidamenti subpleurici o altre anomalie (Figura 1.1). A seguito di un'analisi più approfondita, risulta cruciale distinguere l'importanza degli artefatti nelle immagini LUS. Gli artefatti orizzontali, come le linee A, e gli artefatti verticali, come le Linee B e il polmone bianco, sono da considerarsi echi sonografici. D'altro canto la linea pleurica e i consolidamenti subpleurici sono invece caratteristiche anatomiche reali [8]. Nell'interpretazione delle immagini LUS le A-lines potrebbero non apportare informazioni significative, mentre le caratteristiche anatomiche e artefatti verticali forniscono importanti indicazioni sullo stato dei polmoni, rappresentando manifestazioni di alterazioni patologiche.

1.2 Problemi e sfide di dominio

La valutazione delle ecografie polmonari da parte del personale medico è intrinsecamente soggettiva e suscettibile a variabili interpretative [6]. Infatti la natura visiva delle immagini LUS introduce una serie di sfide pratiche che i professionisti devono affrontare ai fini di una diagnosi accurata. La comprensione degli artefatti, la variabilità inter-osservatore e la necessità di standardizzare le procedure di acquisizione sono solo alcune delle difficoltà incontrate nell'analisi manuale delle ecografie. La soggettività di tale approccio aumenta l'importanza di sottolineare la necessità di metodologie oggettive e supportate da tecnologie per migliorare la precisione e la coerenza delle valutazioni.

L'introduzione di tecnologie di Deep Learning in questo dominio, sembrano emergere come valide soluzioni per affrontare le problematiche indicate precedentemente. Tuttavia, l'impiego di modelli neurali, nello specifico quelli basati sulle Reti Neurali Convoluzionali *(Convolutional Neural Network, CNN)*, introduce ulteriori sfide nell'analisi automatizzata delle ecografie polmonari. Le CNN sono addestrate per riconoscere pattern geometrici nelle immagini, ma le ecografie, in particolare quelle polmonari, possono mancare di strutture geometriche chiare e facilmente riconoscibili [12]. La complessità delle immagini ecografiche, caratterizzate da artefatti e dalla presenza di pattern non lineari, può rendere difficile per le CNN effettuare analisi precise. La necessità di adattare le reti neurali a una vasta gamma di varianti delle immagini e di sviluppare architetture capaci di catturare dettagli sottili diventa imperativa per garantire risultati affidabili.

Inoltre, la stratificazione della gravità della polmonite da Covid-19 costituisce una sfida unica in questo contesto, definito classificazione ordinale. I problemi di classificazione ordinale sono quei compiti di classificazione in cui le etichette sono ordinabili in una scala categorica [13], una caratteristica che si integra perfettamente con il sistema di scoring standardizzato a quattro livelli utilizzato per valutare la gravità della condizione. Questa stretta compatibilità formale fornisce una solida

base teorica per il lavoro di tesi proposto, sottolineando la rilevanza e la motivazione intrinseca nel cercare di migliorare l'efficacia della classificazione della gravità polmonare attraverso approcci specificamente adattati ai dati ordinali.

La natura ordinale della valutazione comporta la necessità di classificare i livelli di gravità in modo graduale, da lieve a severo. Questo pone un'enfasi significativa sulla corretta attribuzione dei punteggi per evitare situazioni di sovrastima o sottostima, specialmente riducendo al minimo errori che potrebbero avere conseguenze cliniche significative. Un esempio concreto in questo contesto è evidenziato da uno studio sulla Broncopneumopatia Cronica Ostruttiva (BPCO) in Svezia [14], che ha dimostrato che ridurre la sottodiagnosi può modificare i fattori di rischio, migliorare il trattamento medico e le strategie di autogestione nelle prime fasi della malattia. Applicando questa lezione al contesto della polmonite da Covid-19, evitare misclassificazioni notevolmente distanti tra i diversi livelli di gravità emerge come un elemento cruciale che impatta positivamente sulla gestione clinica e sull'esito della malattia.

1.3 Obiettivi

Lo scopo principale di questo lavoro è la valutazione dell'efficacia di tecniche che tengono conto dell'ordinalità dei dati, applicate ad un task di classificazione della gravità della polmonite da Covid-19 nelle ecografie polmonari. L'obiettivo chiave è dimostrare che l'applicazione di approcci specifici per dati ordinali può fornire un contributo significativo rispetto alle metodologie di classificazione nominale tradizionali, anche attraverso l'utilizzo di metriche più adatte a valutare la bontà dei risultati in contesti ordinali. Un ulteriore motivazione per questo lavoro è la mancanza, nella letteratura scientifica, di approcci ordinali specifici sul dominio dell'analisi LUS, suggerendo che esplorare tale prospettiva rappresenta un'opportunità significativa per colmare questo vuoto.

Al fine di raggiungere tale obiettivo, sarà fondamentale sviluppare un framework di classificazione robusto, in grado di gestire in modo efficace il flusso di lavoro fornendo una base solida per la valutazione comparativa delle metodologie impiegate.

Parallelamente, si mira a confrontare il framework proposto con lo Stato dell'Arte (*State-of-the-Art, SOTA*), seppur con le limitazioni del caso. Questo confronto è utile per valutare la validità e l'efficacia del nuovo approccio rispetto alla letteratura esistente.

Il resto della tesi è organizzato nei seguenti capitoli: nei *Related Work* viene fatta una rassegna dei lavori presenti in letteratura che hanno portato dei contributi in questo dominio e che sono stati utili per lo sviluppo dell'elaborato, in *Materiali e Metodi* vengono discusse le strategie individuate per risolvere le problematiche descritte e le architetture di modelli neurali adottati; nei *Risultati e Discussioni* vengono riportati i risultati sperimentali supportati da riflessioni scaturite dalla loro analisi, seguiti in fine dalle *Conclusioni*.

Capitolo 2

Related Work

Il Deep Learning (DL) ha ottenuto risultati significativi in svariati compiti di Computer Vision (CV), spaziando dalla classificazione e riconoscimento di oggetti fino alla segmentazione semantica. Questo ha motivato un crescente impiego del DL nelle applicazioni mediche, come ad esempio la rilevazione di polmonite tramite raggi X del torace [15] o anche la segmentazione di immagini biomediche [16]. Tali lavori pionieristici indicano che, con la disponibilità di dati, il DL può svolgere un ruolo importante nell'assistenza e nell'automazione delle diagnosi preliminari, apportando notevole rilevanza alla comunità medica. Tuttavia, in letteratura non sono presenti ricerche che integrino nello specifico la classificazione ordinale con le ecografie polmonari. Questo vuoto motiva l'approccio innovativo proposto in questa tesi. Al fine di colmare questa lacuna, il presente capitolo fornisce un background dettagliato sulla classificazione di immagini ad ultrasuoni, in ambito LUS e sugli approcci ordinali, preparando il terreno per il contributo di questo lavoro che unisce entrambi i domini.

2.1 Classificazione di immagini ad ultrasuoni

Sono oramai diversi anni che si sta assistendo ad un aumento delle attività di ricerca nel campo dell'utilizzo delle reti neurali profonde per l'analisi delle immagini ecografiche (consultare [17] e le relative citazioni). Questo trend evidenzia l'entusiasmo crescente nella comunità scientifica riguardo alle potenzialità offerte dalle reti neurali nell'ambito della diagnostica ecografica. Le reti neurali convoluzionali (CNN) sono una classe di reti neurali profonde (DNN) del mondo DL che sono state impiegate con successo ad una varietà di applicazioni nel dominio delle immagini ad ultrasuoni. Ad esempio un lavoro di Brown et al. [18] ha utilizzato una CNN multi-scala per classificare le malattie epatiche da immagini ad ultrasuoni, raggiungendo ottimi risultati nella classificazione di epatite cronica, cirrosi e cancro al fegato. Stessa conclusione per la diagnosi di calcoli renali da immagini ad ultrasuoni come pubblicato da Alkurdy et al. [19]. Attraverso l'impiego della tecnologia CNN, sfruttando le *features* di una DNN differente [20] è stato possibile diagnosticare i calcoli renali mediante ecografie con una notevole sensibilità e specificità. Ulteriori esempi

non mancherebbero per avvalorare il successo del DL nel campo delle immagini ecografiche e nello specifico anche delle ecografie polmonari (LUS).

2.1.1 Classificazione dei marcatori del Covid-19 nelle LUS

Il paper *"Deep Learning for Classification and Localization of COVID-19 Markers in Point-of-Care Lung Ultrasound"* di Roy et al. del 2020 [7] rappresenta un contributo significativo nello sviluppo di metodologie avanzate per la classificazione e la localizzazione dei marcatori di COVID-19 nelle ecografie polmonari PoC (Point-of-Care LUS). In questo studio, viene illustrato l'utilizzo del Deep Learning per supportare gli operatori sanitari nella rilevazione dei pattern di imaging associati a Covid-19 attraverso LUS PoC. I ricercatori si sono concentrati su tre compiti specifici nell'immagine LUS: classificazione basata su frame, valutazione a livello di video e segmentazione di artefatti patologici. Il primo compito, quello di riferimento e maggiore rilevanza per questa tesi, coinvolge la classificazione di ciascun frame di una sequenza di immagini LUS in uno dei quattro livelli di gravità della malattia, secondo il sistema di punteggio precedentemente definito. La valutazione a livello di video mira a prevedere un punteggio per l'intera sequenza di frame sulla stessa scala di valutazione. Infine, la segmentazione comporta la classificazione a livello di pixel degli artefatti patologici presenti in ciascun frame. Questo studio contribuisce significativamente allo stato dell'arte (*State-of-the-Art, SOTA*) nell'analisi automatica delle immagini LUS, fornendo un DSS al personale medico nella diagnosi delle patologie legate a Covid-19. I principali contributi includono la proposta di una versione estesa e completamente annotata del database ICLUS-DB, con etichette sulla scala a quattro livelli. Il dataset comprende anche un sottoinsieme di immagini LUS annotate a livello di pixel, utile per lo sviluppo e la valutazione di metodi di segmentazione semantica. Inoltre, gli autori introducono un'architettura di deep learning innovativa (Figura 2.1) in grado di prevedere il punteggio associato a una singola immagine LUS e di identificare regioni contenenti artefatti patologici in modo debolmente supervisionato. Utilizzando una *Spatial Transformers Network (STN)* e una funzione di perdita di regressione ordinale (*Soft Ordinal Regression Loss, SORD*), la rete raggiunge la localizzazione dei pattern della malattia per una stima robusta del punteggio associato. L'architettura descritta nella figura è una rete neurale convoluzionale a due stadi. La prima è la fase di localizzazione, dove la rete di trasformazioni spaziali (STN) viene utilizzata per prevedere due trasformazioni, θ_1 e θ_2, che vengono applicate all'immagine di input. Le trasformazioni ruotano e scalano l'immagine in modo da evidenziare i possibili artefatti patologici. Poi segue la fase di classificazione, dove una rete CNN viene applicata alle immagini trasformate x_1 e x_2 per generare le previsioni finali che sono costrette ad essere uguali per non compromettere la coerenza (dato che si tratta della stessa immagine di partenza). In fine in fase di training viene sfruttata la funzione di perdita *SORD* per tenere in considerazione la codifica in termini di distanza dello scoring.

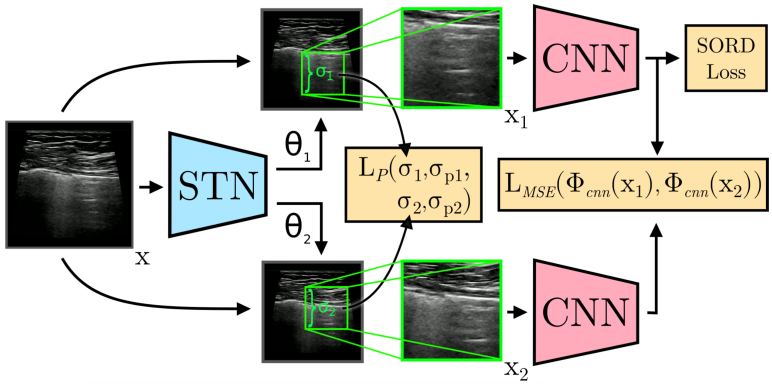


Figura 2.1: Architettura che unisce CNN, Reg-STN e funzione di perdita SORD per la previsione del punteggio basata su frame LUS di Roy et al.

Un altro contributo rilevante, anche se di meno interesse specifico, è l'introduzione di un approccio semplice e leggero basato su *uninorms* per aggregare le previsioni a livello di frame e stimare il punteggio associato a una sequenza video. Gli autori affrontano anche il problema della localizzazione automatica degli artefatti patologici, valutando le prestazioni di metodologie di segmentazione semantica avanzate derivate da architetture completamente convoluzionali.

Arrivando ai risultati dalla ricerca, questi sono stati valutati utilizzando il dataset ICLUS-DB curato dal Laboratorio di Ecografia di Trento. Innanzitutto si fa notare che la sostituzione della loss tradizionale Cross-Entropy (CE) con la loss SORD per la regressione ordinale migliora chiaramente le prestazioni. Si riscontra che l'aggiunta di STN provoca una diminuzione dell'F1-score a causa dei parametri addestrabili aggiunti (pari a quelli della CNN) e l'assenza di una regolarizzazione. Tuttavia, STN presenta due effetti positivi: fornisce localizzazioni debolmente supervisionate e consente l'uso di una regolarizzazione basata sulla consistenza, che è molto vantaggiosa in termini di prestazioni. Il modello completo, che incorpora il modulo STN, la loss SORD e la proposta loss di consistenza, raggiunge un F1-score del 65.1% (dal 61.6% ottenuto con sola architettura CNN e CE), superando tutti i modelli di confronto. Inoltre, i ricercatori fanno notare che la maggior parte della confusione del modello sia causata dal rumore presente sia nei frame che nelle etichette. Si suppone anche che questa incertezza sia dovuta alla soggettività dell'annotazione e alla presenza di frame ambigui. Ciò accade principalmente quando la sonda d'acquisizione è in movimento, causando una transizione da un punteggio a un altro. Per avvalorare questa ipotesi, tramite l'esperimento di valutazione numero due nel quale vengono eliminati i frame vicini ai punti di transizione, si evince che rimuovere i frame ambigui dal set di test riduce drasticamente la quantità di errori del modello, indipendentemente dall'architettura, convalidando empiricamente l'ipotesi sull'etichettatura rumorosa.

2.1.2 Integrazione della conoscenza di dominio LUS nelle DNN

Il secondo contributo proviene dal paper *"A Framework for Integrating Domain Knowledge into Deep Networks for Lung Ultrasound, and its Applications to COVID-19"* di Frank et al. del 2022 [8]. Gli autori presentano un framework finalizzato all'addestramento di reti neurali basato sull'aggiunta di maschere aggiuntive ai dati grezzi. Per raggiungere questo obiettivo, si propone di arricchire esplicitamente l'input del modello con conoscenze specifiche del dominio. In particolare, viene suggerito di informare il modello su importanti caratteristiche anatomiche e artefatti sonografici. Attraverso una fase di pre-elaborazione, vengono rilevate la linea pleurica e gli artefatti verticali (come le Linee B, "polmone bianco", ecc.). Queste informazioni specifiche del dominio, estratte automaticamente, vengono poi fornite come canali di input aggiuntivi, analogamente ai canali di colore RGB nelle immagini naturali, a un modello di DL insieme al frame LUS grezzo. Tali canali specifici del dominio consentono al modello di focalizzarsi meglio su caratteristiche rilevanti e reperti tipici di questo dominio specifico.

La Figura 2.2 illustra l'approccio descritto, la parte superiore mostra un esempio di frame LUS in input e i canali di artefatti verticali e linea pleurica rilevati automaticamente. La concatenazione risultante di queste maschere e del frame di input grezzo viene utilizzata come input per un modello DNN standard (come indicato nella parte inferiore). Fornire esplicitamente al modello questa conoscenza di dominio estratta automaticamente consente l'uso di architetture standard per la classificazione di immagini e di raffinarle rapidamente ed efficientemente per ottenere prestazioni elevate su dati LUS. Il framework consente quindi di addestrare DNN anche quando sono disponibili solo diversi migliaia di esempi di addestramento. Inoltre, rende fattibile l'addestramento di un singolo modello DNN in grado di gestire frame LUS acquisiti sia da sonde convessi che lineari.

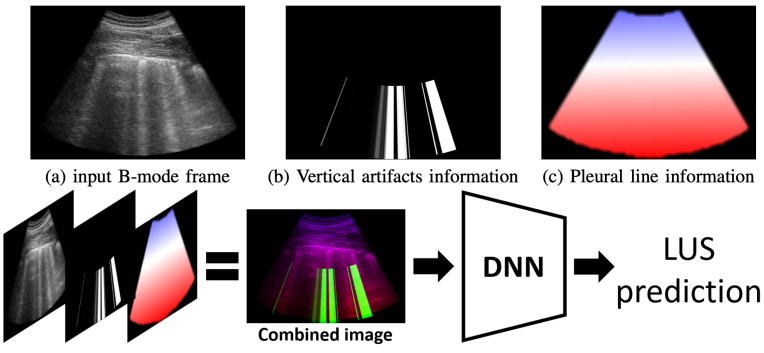


Figura 2.2: Framework per l'integrazione della conoscenza di dominio LUS nelle reti neurali profonde (DNN) di Frank et al.

Il framework viene valutato, sia tramite un task classificazione di frame LUS che in uno di segmentazione semantica. Come per il lavoro di Roy et al. [7], anche questo paper ha valutato i risultati sul dataset ICLUS riuscendo ad ottenere risultati persino

migliori. Il framework è stato utilizzato per raffinare un modello ResNet-18 al fine di classificare ogni frame in base al suo punteggio di gravità annotato. Utilizzando la stessa architettura (ResNet-18), il modello di Frank et al. ha ottenuto un punteggio F1 del 68.8%, rispetto al 62.2% ottenuto dalla stessa architettura, ma senza l'uso esplicito degli artefatti verticali e delle informazioni sulla linea pleurica. Inoltre, il modello progettato supera l'architettura CNN-Reg-STN proposta da Roy et al. specificamente progettata per dati LUS, definendo il nuovo stato dell'arte. I ricercatori hanno anche osservato che è più vantaggioso incorporare la conoscenza del dominio come canali di input aggiuntivi piuttosto che come design di architetture neurali profonde. Il metodo GradCAM [21] è stato impiegato per ispezionare visivamente le previsioni dei modelli ResNet-18. Queste visualizzazioni suggeriscono che la rete è in grado di guidare efficacemente il modello nelle regioni rilevanti del frame, è in grado di esaminare la linea pleurica e le regioni sospette all'interno della cavità polmonare, prestando meno attenzione al tessuto sopra la linea pleurica.

2.2 Classificazione ordinale nella diagnostica per immagini

Negli anni, il concetto di classificazione ordinale (o regressione ordinale) è stato diffuso come un modo per sfruttare informazioni extra in un problema di classificazione in cui è presente un ordinamento naturale delle classi [13]. La classificazione ordinale, oltre ad essere stata applicata con successo in diverse aree, ha persino dimostrato di superare la prospettiva classica nominale nelle applicazioni mediche, come nella stima della progressione di Alzheimer [22], o nel trapianto di fegato [23] o anche nella diagnosi del melanoma [24]. I task di classificazione ordinale sono diversi dalla regressione perché la distanza tra i valori della variabile dipendente (la classe) è generalmente sconosciuta. La situazione più comune nella regressione ordinale è che le categorie derivano dalla discretizzazione di una variabile latente, che è esattamente il caso dei diversi score della stratificazione della polmonite.

I modelli a soglia sono un approccio popolare per questo compito. Si presume che esista una variabile latente continua sottostante, da cui derivano i diversi ranghi impostando determinate soglie. In questo contesto, sia il valore della variabile latente che le soglie devono essere apprese dai dati. Il pioneristico *Proportional Odds Model (POM)* [25] rientra nel framework del *Cumulative Link Model (CLM)* [26], un metodo probabilistico per prevedere le probabilità di gruppi di categorie contigue, tenendo conto della scala ordinale. In un recente lavoro di Vargas et al. [27] si propone un modello di rete neurale convoluzionale per la classificazione ordinale, sfruttando la tecnologia CNN a monte di funzioni di collegamento ordinali probabilistiche e combinando tali modelli con una funzione di perdita che tiene conto della distanza tra le categorie. La sperimentazione include diverse funzioni di collegamento, confrontate tramite analisi statistica su diversi dataset, e dimostra che questi modelli migliorano i risultati rispetto a un modello nominale e superano altre proposte robuste presenti

in letteratura. Questo definirà uno dei due approcci che verrano implementati in questa tesi per risolvere il problema posto.

Altri approcci ordinali consistono nella decomposizione del problema ordinale in un insieme di problemi binari (*Ordinal Binary Decomposition, OBD*). A volte queste decomposizioni sono risolte da un insieme di modelli diversi, come nel modello di utilità lineare a cascata [28]. In altri casi sono modellate da diversi output dello stesso modello sottostante [29]. Tutti i metodi OBD presentano inevitabilmente la stessa sfida: combinare i risultati di tutte le decomposizioni in una singola classificazione finale. Un'eccellente ricerca pubblicata nel 2021, condotta nel contesto della valutazione dei danni neurologici in pazienti affetti dalla malattia di Parkison (PD) [30], mostra ancora una volta come l'impiego di tecniche ordinali migliorino le prestazioni rispetto ai metodi nominali. Questo articolo propone un modello OBD basato su CNN 3D per valutare il livello dei danni neurologici nei pazienti affetti da PD. Dato che le CNN necessitano di ampi set di dati per ottenere prestazioni accettabili, viene adattato un metodo di aumento dei dati per lavorare con dati spaziali (OGO-SP-β). La valutazione dei diversi metodi è stata effettuata su un nuovo dataset di immagini 3D fornito dal *Hospital Universitario 'Reina Sofía' (Córdoba, Spagna)*. Per queste ragioni, si è scelto di incorporare non solo un modello Cumulative Link Model (CLM), ma anche un approccio OBD (Ordinal Binary Decomposition), al fine di ottimizzare la classificazione delle lesioni polmonari nelle ecografie polmonari.

2.2.1 Contestualizzazione dei metodi ordinali nelle immagini LUS

I metodi ordinali sfruttano quindi la natura ordinata delle classi per migliorare gli algoritmi di apprendimento e, al contempo, penalizzano la magnitudine degli errori di classificazione tramite l'utilizzo di funzioni di loss dedicate. Ad esempio, in questo caso specifico, confondere uno score 0 con uno score 1 non dovrebbe essere considerato lo stesso che confonderlo con uno score 3. Quando si cerca di valutare lo score di una polmonite leggendo un'immagine LUS, si nota che esiste una struttura ordinata tra le classi. Classificare un'ecografia polmonare coinvolge l'osservazione di artefatti e caratteristiche anatomiche, tra cui la linea pleurica, le linee B, il *"white lung"* e i consolidamenti subpleurici. Tali punteggi della polmonite sono strettamente correlati alla presenza e alle condizioni di questi fattori (Figura 2.3). Ad esempio, una linea pleurica regolare e l'assenza di artefatti verticali significativi possono indicare uno score basso. Mentre, uno score più alto potrebbe essere suggerito da una perdita di regolarità nella linea pleurica, che appare dentellata, e altre condizioni specificate precedentemente. Nell'ambito CLM, questi fattori sono i giusti candidati per modellare la variabile latente introdotta poc'anzi, in grado di identificare la gravità della condizione polmonare. Inoltre, va notato che l'Ordinal Binary Decomposition (OBD) è anch'esso applicabile a questo contesto. Tale approccio si adatta particolarmente bene a situazioni in cui la variabile dipendente è ordinale, appunto come nel caso del sistema di scoring standardizzato a quattro

livelli. OBD suddivide l'output ordinale in una serie di problemi binari, consentendo una gestione efficace e precisa delle diverse categorie di gravità. Questa flessibilità lo rende una scelta appropriata e motivata per il caso in esame. Questi approcci di valutazione riflettono la natura ordinale della stratificazione della gravità della polmonite, consentendo di adottare tali metodologie specifiche per dati ordinali al fine di migliorare la precisione della classificazione del danno della polmonite da Covid-19.

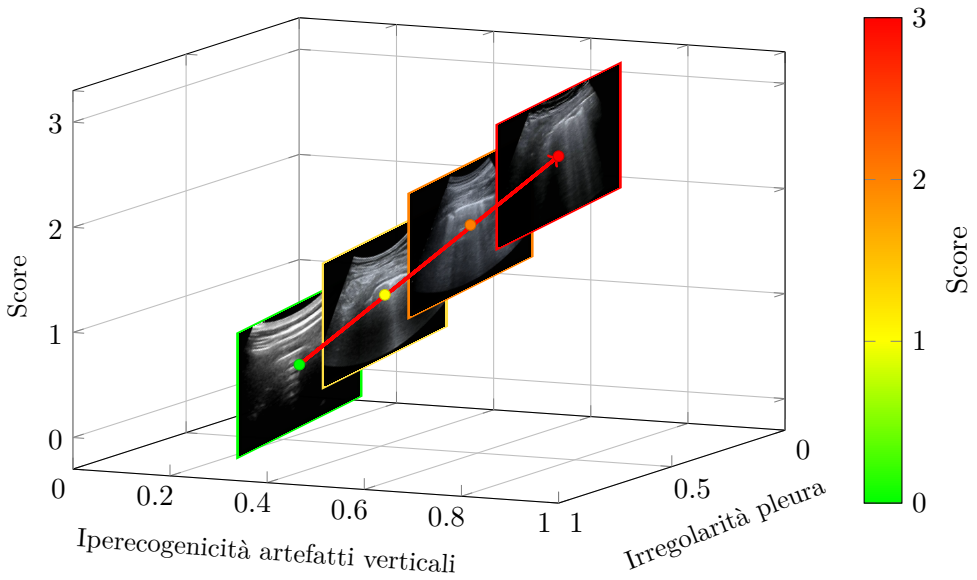


Figura 2.3: Relazione ordinale tra artefatti verticali, pleura e score nei frame LUS.

Capitolo 3

Materiali e Metodi

Il capitolo dedicato ai Materiali e Metodi costituisce il cuore operativo dell'elaborato, presentando le metodologie utilizzate per affrontare la classificazione ordinale delle immagini ecografiche polmonari. In particolare, vengono delineate le strategie implementate attraverso l'applicazione di reti neurali convoluzionali (CNN) e approcci ordinali specifici, quali il Cumulative Link Model (CLM) e l'Ordinal Binary Decomposition (OBD). In questo capitolo, saranno esaminati in dettaglio i passaggi fondamentali del processo metodologico, dal trattamento del dataset, all'implementazione delle reti neurali con approccio nominale e ordinale, fino alla definizione del framework nel suo complesso, inteso come flusso di lavoro globale del progetto.

3.1 Dataset ICLUS

L'*Italian COVID-19 Lung Ultrasound DataBase (ICLUS-DB)* è un progetto del Dipartimento di Ingegneria e Scienza dell'Informazione dell'Università di Trento coordinato da Libertario Demi. I dati sono stati raccolti da cinque centri clinici italiani: BresciaMEd di Brescia, Valle del Serchio General Hospital di Lucca, Fondazione Policlinico Universitario A. Gemelli IRCCS di Roma, Fondazione Policlinico Universitario San Matteo IRCCS di Pavia e Tione General Hospital di Tione [7]. Attualmente il dataset è composto da 277 video di ecografie polmonari (LUS) appartenenti a 35 pazienti, per un totale di esattamente 58.924 frame (Figura 3.1). Per l'acquisizione dei video sono state utilizzate diverse apparecchiature e quindi sia sonde lineari (13.364 frame) che convesse (45.560 frame), a seconda delle esigenze. Dei 35 pazienti totali, 17 sono risultati positivi al Covid-19, 4 erano sospetti, e 14 erano individui sani e asintomatici. Per valutare la progressione della patologia, è stato creato un sistema di scoring a 4 livelli, con punteggi da 0 a 3 [5]. Il punteggio 0 indica una superficie polmonare sana, mentre i punteggi successivi indicano anomalie crescenti, con il punteggio 3 associato a un'area iperecogena definita *"white lung"* (letteralmente "polmone bianco"). Dei 58.924 frame, il 34% è stato etichettato come score 0, il 24% score 1, il 32% score 2 e il 10% come score 3.

Per garantire un'annotazione oggettiva, il processo è stato suddiviso in 4 livelli, con diversi professionisti coinvolti. Nella prima fase viene assegnato il punteggio frame per frame da quattro studenti universitari con competenze comprovate in ambito

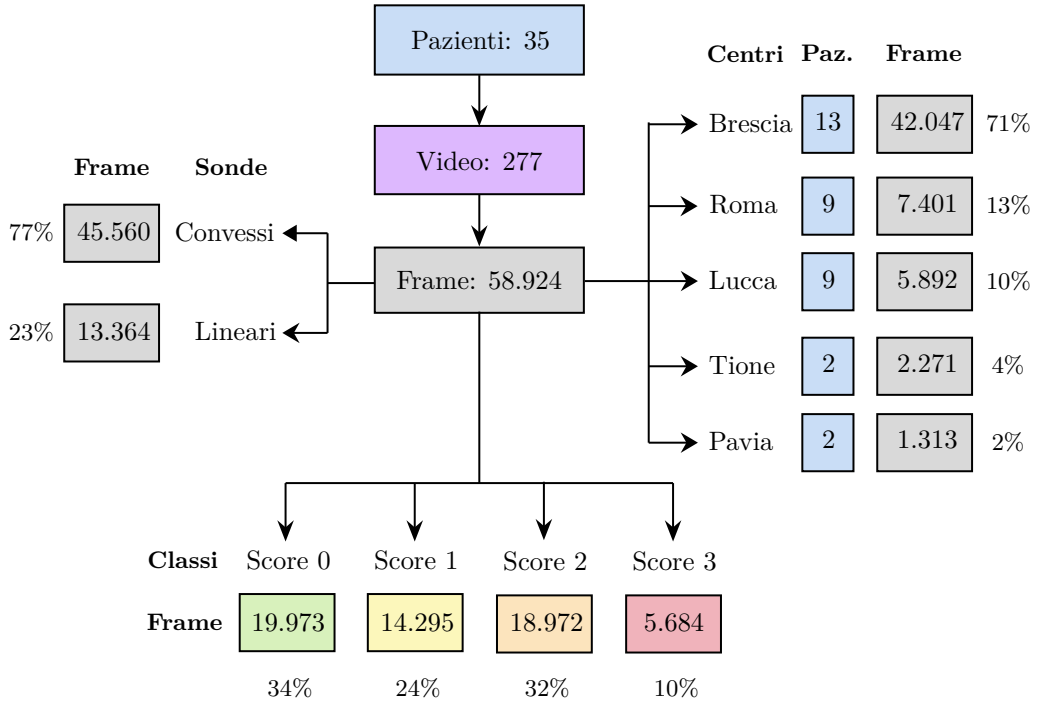


Figura 3.1: Schema omnicomprensivo del dataset ICLUS-DB.

ecografico, successivamente i punteggi assegnati vengono rivalutati da uno studente di dottorato (fase 2) e da un ingegnere biomedico con oltre 10 anni di esperienza in LUS (fase 3). Infine il quarto livello di validazione è costituito dall'accordo tra medici con più di 10 anni di esperienza in LUS. Da notare che l'accordo medio tra gli operatori è stato del 67%. Questi dati costituiscono la base della tesi e riflettono una vasta gamma di condizioni cliniche nei pazienti affetti da Covid-19.

In linea generale, si fa notare che ciascun centro medico ha contribuito alla diversificazione del dataset con distribuzioni di dati leggermente differenziate (Figura 3.2). Ad esempio, i centri medici di Pavia e Brescia hanno fornito dati più uniformemente distribuiti tra i diversi punteggi rispetto gli altri.

Nello specifico la Figura 3.2 mostra la distribuzione degli score a livello di paziente di ogni centro medico del dataset. In generale, come anticipato, si può osservare che la distribuzione degli score differisce tra i centri medici. Per di più alcuni pazienti mostrano una distribuzione abbastanza uniforme dei quattro score. Nello specifico, i centri medici di Pavia e Brescia sono quelli con la distribuzione più uniforme degli score, prevedibilmente con una maggiore concentrazione sui primi 3 gradi del punteggio. Lucca e Roma sono maggiormente composti da pazienti sani e asintomatici e per una piccola misura anche da dati contenenti gli score più gravi. Infine, Tione è l'unico centro medico che contiene esclusivamente pazienti sani. Queste differenze potrebbero essere dovute a fattori quali la popolazione del centro medico, le pratiche mediche utilizzate o la disponibilità di risorse mediche.

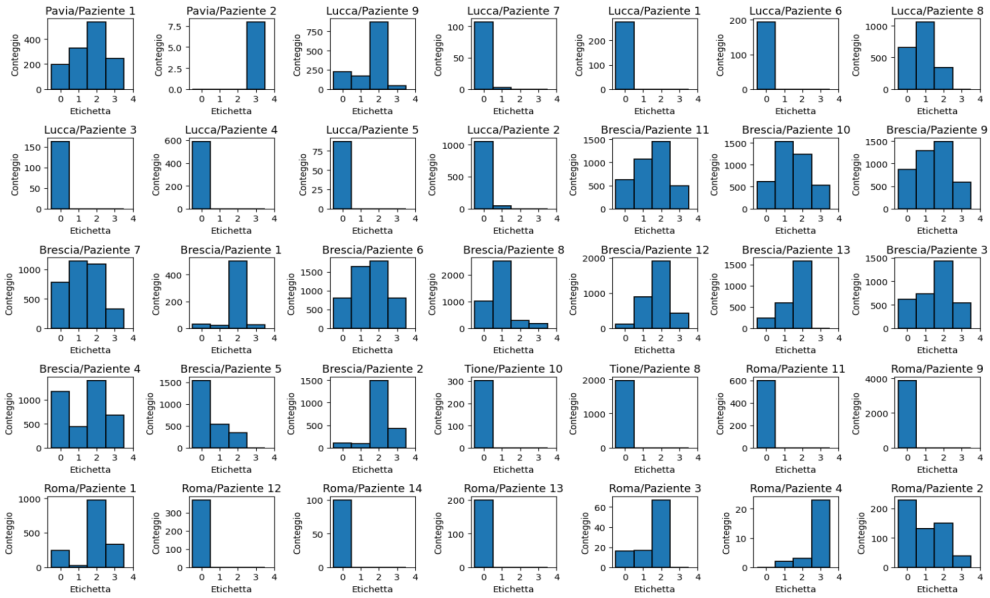


Figura 3.2: Distribuzione degli score di ogni paziente per ogni centro medico.

3.1.1 Sistema di scoring e natura ordinale intrinseca

La standardizzazione del protocollo di acquisizione, oltre che definire i punti e i movimenti specifici per eseguire l'ecografia polmonare, comprendere anche il sistema di punteggio per valutare le immagini ottenute classificandole in base a quattro livelli di gravità della condizione polmonare. In Figura 3.3 si riportano un paio di esempi per ogni punteggio per rendere apprezzabile come ogni gravità risulta riconoscibile nelle due tipologie di sonde ecografiche.

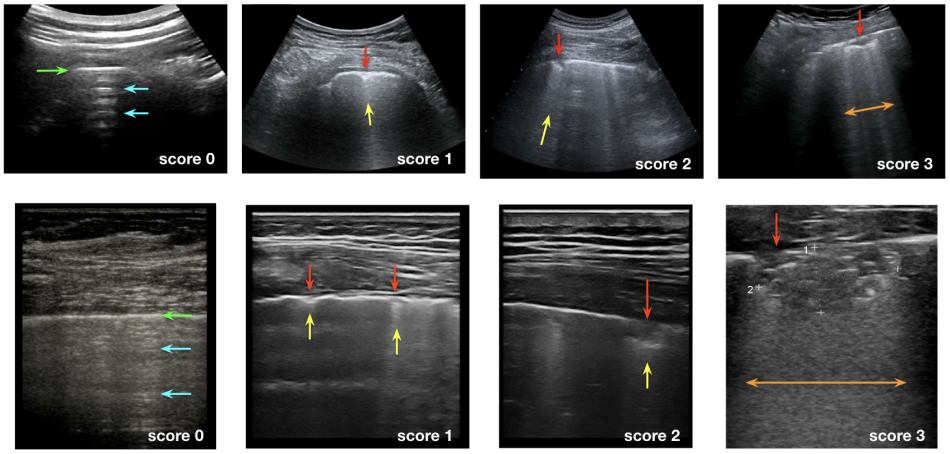


Figura 3.3: Sistema di scoring ICLUS a quattro livelli: esempi convessi (prima riga) ed esempi da sonde lineari (seconda riga) per ciascun punteggio.

La prima riga di immagini della Figura 3.3 mostra esempi di frame convessi, mentre la seconda frame lineari. Le linee verdi indicano la linea pleurica, quelle azzurre

gli artefatti orizzontali (linee A), quelle rosse evidenziano le irregolarità della linea pleurica o nei casi più gravi dei consolidamenti subpleurici, quelle gialle gli artefatti verticali (linee B) ed infine le linee arancioni mostrano il pattern "white lung" negli stadi avanzati. Questo sistema di punteggio fornisce una valutazione dettagliata della gravità della condizione polmonare, consentendo una classificazione accurata delle immagini ottenute attraverso l'ecografia polmonare standardizzata.

Gli score vengono quindi assegnati in base alle informazioni estratte dall'ecografia, di seguito si riportano le linee guida che definiscono ciascun punteggio:

- `Score 0`: la linea pleurica appare continua e regolare, potrebbe essere accompagnata da artefatti orizzontali noti come linee A. Questa categoria rappresenta uno stato di salute polmonare normale.
- `Score 1`: la linea pleurica è discontinua e sotto di essa possono essere visibili aree verticali bianche. Questo indica i primi segni di anormalità polmonare.
- `Score 2`: evidenzia una linea pleurica interrotta, con piccole o grandi aree consolidate (aree più scure) e aree bianche sotto l'area consolidata (linee B), indicando uno stato patologico più avanzato.
- `Score 3`: l'area esaminata mostra un'estesa zona di polmone bianco denso, con consolidamenti più estesi, coprendo almeno il 50% della linea pleurica. Questo rappresenta uno stato di gravità significativo.

Il sistema di scoring presenta dunque una chiara ordinalità, in cui le categorie sono naturalmente ordinate e con distanze non note per definizione. Ogni livello di punteggio, dal 0 al 3, rappresenta uno stadio progressivamente più grave della condizione polmonare. Ad esempio in termini insiemistici, lo score 3 è inclusivo di tutte le caratteristiche presenti negli score 2 e 1, creando una struttura gerarchica in cui ogni informazione diagnostica contenuta in uno score inferiore è implicitamente inclusa nello score superiore (`score 3` $\subseteq$ `score 2` $\subseteq$ `score 1`). Questa natura ordinata riflette la progressione della gravità della patologia, fornendo una base solida per l'applicazione di metodologie specifiche per dati ordinali nella valutazione delle immagini LUS e garantendo una classificazione accurata e dettagliata delle condizioni polmonari.

3.1.2 Acquisizione e fasi della gestione del dataset

Il dataset è stato originariamente fornito in formato MATLAB (`.mat`), che è un formato di file proprietario non compresso. Ciascun video ecografico nel set di dati `ICLUS-DB` è stato etichettato in base al sistema di scoring discusso precedentemente ed è rappresentato da tre file:

- il video originale, opportunamente editato per rimuovere le informazioni personali del paziente, è fornito nel formato MATLAB (`.mat`). I dati sono contenuti

nella variabile `"frames"`, una matrice 4D delle dimensioni $(R \times C \times 3 \times F)$, dove R e C indicano rispettivamente il numero di righe e colonne in un singolo frame, 3 rappresenta il numero di canali e F è il numero di frames nel video.

- la matrice dei punteggi è anch'essa fornita in formato MATLAB. In questo caso i dati sono contenuti nella variabile `"Score matrix"`, una matrice 2D delle dimensioni $3 \times F$ (Figura 3.4), dove 3 rappresenta i punteggi da 0 a 3, e F è il numero di frames nel video. In altre parole, ogni frame è descritto da un vettore (3×1) che indica punteggio con cui è stato etichettato. Quindi Il vettore $(0, 0, 0)$ indica che il frame è etichettato come Score 0, $(1, 0, 0)$ come Score 1, $(1, 1, 0)$ come Score 2 ed infine se un frame ha label $(1, 1, 1)$ indica che appartiene alla categoria Score 3.

First row	0	1	1	1
Second row	0	0	1	1
Third Row	0	0	0	1
Score	0	1	2	3

Figura 3.4: Matrice di codifica dello scoring ICLUS.

- per completare il dataset, viene fornito un video `.avi` che mostra la versione finale del dataset etichettato. Ogni frame può essere circondato da un rettangolo colorato in base all'etichettatura associata.

Questo formato era inefficiente per l'elaborazione dei dati, quindi è stato convertito in `HDF5`, che è un altro formato di file ma aperto e gerarchico. Questo processo è stato progettato per leggere i file `.mat` e raccogliere i dati relativi ai frame e ai target (scores). Sono anche stati implementati dei controlli mirati sulla forma dei dati per garantire l'integrità del dataset. Per quanto riguarda gli score, viene eseguito un controllo sulla forma del punteggio, verificando se è in accordo con le aspettative. Ad esempio, si controlla se ogni frame ha un punteggio associato e se la forma rispetta le dimensioni attese. Inoltre, vengono effettuati controlli dimensionali per verificare se il numero di score per ogni frame è corretto. Se la forma non è riconosciuta o non rispetta le aspettative, lo score in analisi e il video corrispondente vengono esclusi dalla conversione. Analogamente, per i video, vengono verificate le forme dei frame. Inoltre, vengono effettuati controlli sulla presenza di un singolo frame o di più frame nel video, assicurandosi che le dimensioni siano adeguate per l'elaborazione successiva. Durante la fase di caricamento, viene anche eseguito un controllo sulla coerenza tra dati video e score per assicurarsi che entrambi siano disponibili prima di procedere con l'analisi. Questi controlli mirati assicurano che il dataset sia correttamente strutturato e pronto per essere utilizzato nelle fasi successive del progetto.

La fase di conversione crea un file `HDF5` e iterativamente salva i dati dei video e dei target dal dataset Matlab. Per ciascun video, vengono create le strutture di gruppo necessarie nel file `HDF5`, come il macrogruppo della sonda (`convex` o `linear`), il gruppo del singolo video e i sottogruppi per frame e target. Inoltre, vengono salvate anche le informazioni sul paziente, il centro medico, il tipo di sonda e il file di origine.

I dati vengono quindi salvati nei relativi sottogruppi. Il processo può essere interrotto e ripreso in un secondo momento, salvando un checkpoint che memorizza l'indice corrente dei frame e dei video elaborati. In caso di interruzione, il processo riprende dal punto in cui si è interrotto. L'intero processo di conversione viene monitorato e visualizzato attraverso una barra di avanzamento che fornisce informazioni sul numero di video elaborati, la dimensione del file in corso di creazione e il numero di frame elaborati. Il processo di conversione richiede approssimativamente 1 ora e 30 minuti, generando un file con una dimensione di circa 35 GB.

Tuttavia, durante i primi tentativi di addestrare reti neurali sul formato `HDF5`, le prestazioni non raggiungevano le aspettative prefissate, con dei tempi d'inferenza di circa 30 minuti ad epoca. Di fronte a questa sfida, è stato eseguito un profiling di `TensorFlow` per ottimizzare le prestazioni.

Come soluzione, si è deciso di migrare al formato `TFRecordDataset`. Questo ha permesso di strutturare il nuovo dataset organizzando i dati in file per ogni video contenente tutti i frame e score associati, dividendo tutti i video su due livelli, per paziente e per centro medico. Il peso complessivo è ora di circa 5 GB, e l'utilizzo di una pipeline di input basato su `TFRecord` ha notevolmente migliorato le prestazioni, riducendo il tempo di addestramento a circa 2 minuti per epoca, un miglioramento del 93% rispetto al formato HDF5. La prima fase di conversione è stata eseguita su *Google Colab* utilizzando `PyTorch`. Successivamente, la seconda fase è stata effettuata in locale su un MacBook Air M1 passando all'ambiente `TensorFlow`. Ulteriori dettagli sulla configurazione dell'ambiente di sviluppo verranno forniti nella Sezione 3.6.

3.1.3 Preprocessing e Data Augmentation

Nella prima fase di gestione del dataset è emersa una notevole eterogeneità nelle risoluzioni dei video a causa delle diverse strumentazioni utilizzate nei cinque centri medici coinvolti. Un'analisi delle risoluzioni dei video ha evidenziato la necessità di standardizzare le dimensioni dei frame. Di conseguenza, tutti i frame sono stati ridimensionati a una risoluzione comune di $224 \times 224 \times 3$ per garantire uniformità nel set di dati. Inoltre, al fine di applicare una standardizzazione completa, è stato effettuato un processo di normalizzazione su tutti i frame. Questo processo consiste nella divisione di ciascun valore per 255, al fine di ottenere un intervallo normalizzato compreso tra 0 e 1.

La *Data Augmentation*, una nota tecnica per migliorare la generalizzazione dei modelli di Deep Learning, è stata implementata in modalità online, conosciuta come *Online Data Augmentation*. In questo contesto, la data augmentation si riferisce a trasformazioni applicate alle immagini durante l'addestramento dei modelli per diversificarne il set di dati, promuovendo così una maggiore robustezza e generalizzazione. L'implementazione online indica che queste trasformazioni vengono applicate alle batch di dati direttamente durante il processo di addestramento. Le trasformazioni utilizzate sono state selezionate sulla base dello stato dell'arte [7], con alcune variazio-

ni introdotte come conseguenza di risultati empirici. Ciascuna trasformazione è stata applicata con una probabilità del 50%, introducendo così variabilità nell'applicazione di tali trasformazioni ma anche evitando di impattare eccessivamente il contenuto informativo. Le trasformazioni includono il ritaglio centrale casuale, la modifica di luminosità e contrasto, il flip orizzontale e la rotazione casuale (Tabella 3.1). Nello specifico, il ritaglio casuale effettua uno zoom tra il $70 - 90\%$ della porzione centrale dei frame, potenzialmente focalizzando l'analisi sulla parte informativa. La variazione di luminosità, che altera appunto l'intensità dei pixel, è stata implementata con lo scopo di evidenziare determinate parti più o meno luminose con l'obiettivo di enfatizzare quelle che sono le componenti anatomiche ed artefatti ecografici. Le rimanenti trasformazioni includono il flip orizzontale e la rotazione casuale entro un intervallo di -23° a +23°.

Trasformazione	Parametri	Probabilità
Random crop	minval=0.7 ; maxval=0.9	$\mathbb{P} = 0.5$
Random brightness	minval=-0.1; maxval=0.15	
Random horizontal flip	-	
Random rotation	limit=(-23, 23)	

Tabella 3.1: Trasformazioni e parametri dell'Online Data Augmentation.

3.2 Reti Neurali Convoluzionali (CNN)

Le Reti Neurali Convoluzionali (*Convolutional Neural Network, CNN*) costituiscono una componente fondamentale nell'ambito della visione artificiale e del Machine Learning (ML) applicato al medical imaging [12], rivestendo un ruolo cruciale nella risoluzione di compiti complessi legati alle immagini. Questa tecnologia è particolarmente adatta per l'estrazione di informazioni significative dalle immagini, grazie alla sua capacità di apprendimento automatico delle caratteristiche attraverso strati convolutivi. Una CNN è progettata per riconoscere pattern spaziali e gerarchie di features nelle immagini, simulando in modo efficace il processo di percezione visiva umana.

La fase convolutiva, costituita appunto dalle CNN, è essenziale per il progetto, poiché è responsabile dell'estrazione di features semantiche dai frame delle ecografie polmonari. Questa fase consente alla rete di "imparare" rappresentazioni significative dalle LUS, identificando dettagli e strutture rilevanti. Nello specifico, l'aspettativa è quella di far apprendere ai moduli convolutivi la posizione spaziale delle componenti anatomiche dell'immagine e dei vari artefatti che si possono presentare. Le features estratte costituiranno in seguito l'input per la successiva fase di classificazione, che dipenderà dal tipo di approccio impiegato, nominale e ordinale per l'appunto.

Nonostante il focus principale del progetto sia sulla classificazione ordinale, la fase convolutiva è indispensabile poiché fornisce un meccanismo per comprendere la

complessità delle immagini e catturare informazioni semantiche. Questo approccio permette di ottenere una migliore generalizzazione del modello e di affrontare il problema di classificazione in modo più accurato, sfruttando le relazioni spaziali presenti nelle immagini.

3.2.1 ResNet18 come Feature Extractor

Le *"Residual Network"* (o anche *ResNet*), si sono affermate come una delle architetture più rilevanti e influenti nelle reti neurali profonde [31]. *ResNet18* prende il suo nome dalla combinazione di *Residual Network* e dal numero di strati (o layers) totali nella rete, appunto 18. La sua architettura si basa sul concetto di *blocchi residuali*, che ha introdotto un approccio alla progettazione delle reti neurali in grado di mitigare il problema della scomparsa del gradiente. Questo problema si verifica quando la retropropagazione del gradiente attraverso numerosi strati di una rete risulta difficoltosa, limitando la capacità di apprendimento del modello. La peculiarità di ResNet è l'uso di quelle che vengono definite *skip connections*, che consentono al gradiente di "saltare" alcuni strati durante la retropropagazione. Questa tecnica permette al modello di imparare gli errori residui più facilmente e favorisce una formazione più stabile di reti profonde. La sua progettazione mira quindi a mantenere un equilibrio tra complessità computazionale e prestazioni, fornendo una soluzione efficace per problemi di classificazione di immagini.

Rispetto ai modelli alternativi, ResNet18 si distingue per la sua capacità di addestrare reti più profonde senza incorrere in problemi di degradazione del modello. Questo modello è noto per la sua efficienza e la sua capacità di apprendimento su dataset di grandi dimensioni, contribuendo così a migliorare le prestazioni in una varietà di compiti di visione artificiale. La sua struttura semplificata rispetto a varianti più complesse, come ad esempio ResNet50, la rende particolarmente adatta per applicazioni in cui la complessità computazionale e il numero di parametri del modello possono influenzare negativamente le prestazioni, soprattutto quando la profondità della rete non è essenziale per ottenere risultati migliori.

Il codice implementato per ResNet18 riflette fedelmente la sua architettura originale, come illustrato nella Tabella 3.2. Il modello è costituito da uno strato di convoluzione iniziale (`conv1`) con kernel 7×7 e 64 filtri, seguito da una `Batch Normalization (BN)` e un'attivazione `ReLU`. Successivamente, è presente uno strato di Max Pooling (`max pool`) con un kernel 3×3 e uno stride di 2 per ridurre le dimensioni spaziali dell'immagine. Gli strati successivi compongono blocchi residuali (da `conv2_x` a `conv5_x`) che includono in ordine: convoluzione 2D, batch normalization e attivazione ReLU ripetute 2 volte e le skip connections per agevolare il flusso del gradiente. La rete termina con uno strato di Global Average Pooling 2D (`average pool`), che riduce la dimensione spaziale delle feature a un vettore di output $1 \times 1 \times 512$. Questi strati costituiscono la parte convolutiva dei modelli utilizzati in questa tesi, come motivato nella sezione precedente.

Strato	Dimensione in uscita	ResNet18
conv1	$112 \times 112 \times 64$	7×7, 64, stride 2
max pool	$56 \times 56 \times 64$	3×3 max pool, stride 2
conv2_x	$56 \times 56 \times 64$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$
conv3_x	$28 \times 28 \times 128$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$
conv4_x	$14 \times 14 \times 256$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$
conv5_x	$7 \times 7 \times 512$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$
average pool	$1 \times 1 \times 512$	7×7 global average pool

Tabella 3.2: Architettura convoluzionale della ResNet18 implementata.

3.3 Architettura implementativa unificata

L'architettura delle rete neurali impiegate in questo lavoro è stata progettata e implementata con un'approccio modulare, garantendo flessibilità e adattabilità alle esigenze specifiche degli esperimenti condotti. Infatti, la classe designata per rappresentare i modelli neurali, sia per l'approccio nominale che ordinale, offre una struttura unificata. Questa classe comprende la fase di estrazione delle caratteristiche con diverse opzioni di backbone impiegabili (es. ResNet18, ResNet50, VGG16, ecc.). Per lo studio in questione si è in ogni caso deciso di concentrarsi sull'uso esclusivo di ResNet18 per allinearsi con lo stato dell'arte, sia in configurazione *"from scratch"* (pesi inizializzati randomicamente) che preaddestrata su *ImageNet (IN1k)*. Quest'ultimo approccio, noto come *Transfer Learning*, mira a sfruttare le conoscenze apprese da una vasta gamma di immagini eterogenee per migliorare le prestazioni del modello su specifici compiti di classificazione.

Successivamente, la classe incorpora il modulo di classificazione, che può essere configurato in modo flessibile per adattarsi ai diversi contesti affrontati. Per l'approccio nominale viene utilizzato un modulo tradizionale con *Average pooling*, layer denso completamente connesso (*Fully Connected, FC*), *Dropout* (opzionale) e uscita *Softmax*. Per l'approccio ordinale, la classe prevede l'integrazione di modelli di classificazione ordinale come OBD (Ordinal Binary Decomposition) o CLM (Cumulative Link Model), argomenti approfonditi nelle sezioni dedicate della tesi. Questo approccio modulare permette una gestione agevole e coesa delle diverse configurazioni di modelli, semplificando il processo di sperimentazione e adattamento alle specifiche esigenze di ogni scenario di classificazione.

Per riassumere, come mostrato in Figura 3.5, le architetture delle reti implementate

condividono la stessa struttura per quanto riguarda la parte convoluzionale. Tuttavia, è importante sottolineare che le reti vengono addestrate in modo seriale e distinto, ciascuna con il proprio processo di addestramento convoluzionale. Le reti differiscono quindi nel modo in cui l'output finale viene calcolato, permettendo così un approccio personalizzato alla fase successiva della classificazione ordinale o nominale.

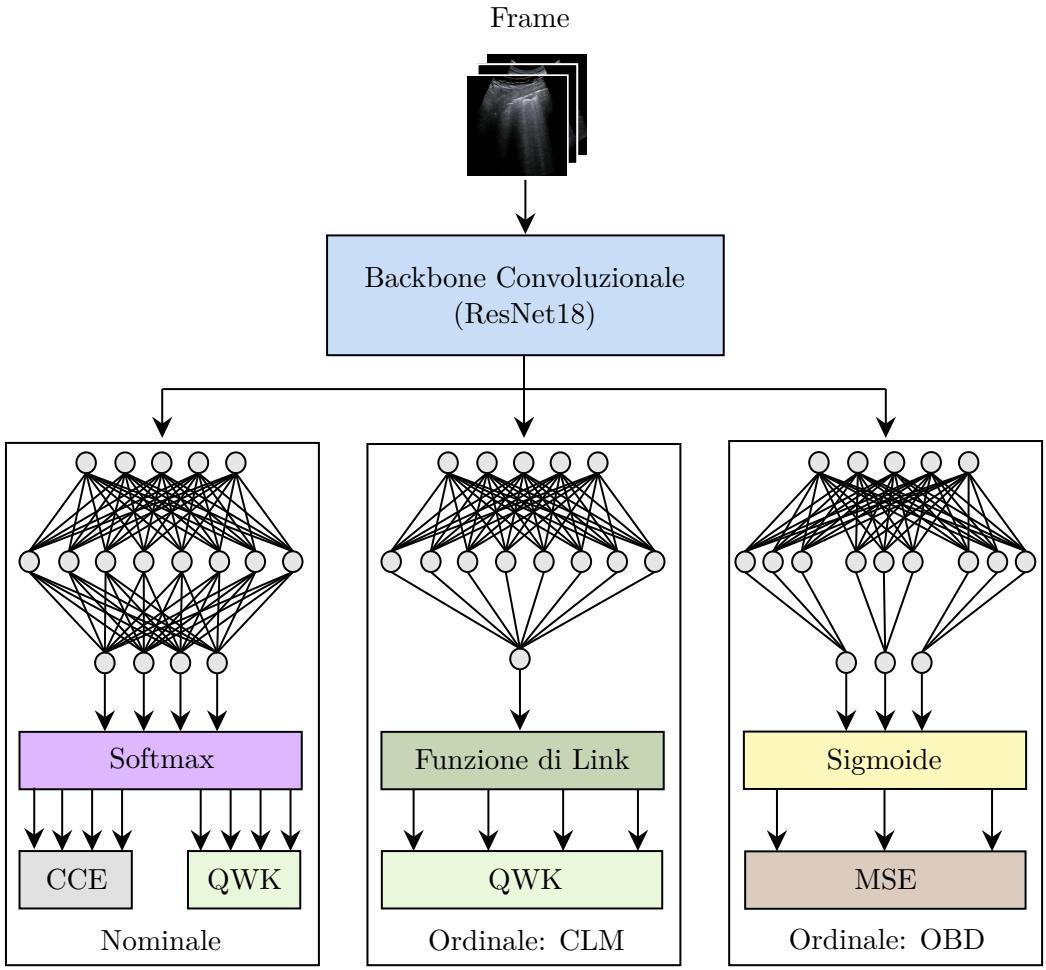


Figura 3.5: Confronto delle tre reti neurali implementate con architettura unificata che comprende la stessa backbone convoluzionale e come moduli di classificazione, da sinistra a destra: architettura nominale di riferimento (utilizzando come funzione di loss sia CCE che QWK), CLM e OBD.

3.4 Rete di classificazione con approccio nominale

Il modello, seguendo un approccio di classificazione nominale, integra la parte convolutiva della rete neurale convoluzionale (CNN) come suo estrattore di caratteristiche primario. Questa componente costituisce il nucleo iniziale del modello e assume un ruolo cruciale nella cattura di schemi e tratti distintivi presenti nelle immagini, come discusso in precedenza.

Una volta che le caratteristiche significative sono state estratte dall'ultimo blocco convolutivo, l'output subisce il processo di `Global Average Pooling (GAP)`. Questa operazione è fondamentale per ridurre la dimensionalità delle feature map bidimensionali a un vettore unidimensionale, rappresentando in modo più compatto le informazioni salienti. Il GAP consiste nel calcolare la media dei valori lungo ciascuna dimensione delle feature map, generando un singolo valore per ogni canale. Questo vettore risultante, noto come vettore di pooling globale, contiene informazioni essenziali su ciascuna feature map.

Successivamente, il vettore ottenuto attraverso il GAP passa attraverso uno strato denso completamente connesso (`Fully Connected, FC`) con un numero predefinito di neuroni (di default 1000). Questo strato denso svolge un importante ruolo nel combinare e interpretare le caratteristiche estratte, preparandole per la fase di classificazione.

Se impostato, viene applicato uno strato di `Dropout`, il quale aiuta a prevenire l'overfitting del modello durante la fase di addestramento, disattivando casualmente alcuni neuroni durante ogni iterazione. La probabilità di disattivazione è determinata dal parametro di *dropout rate*, specificato durante la configurazione dell'esperimento (approfondito nella Sezione 3.6.1).

Infine, uno strato denso finale con funzione di attivazione `Softmax` viene applicato per ottenere le probabilità delle quattro classi, corrispondenti ai quattro livelli di scoring di `ICLUS-DB`. La funzione softmax è una scelta comune per problemi di classificazione nominale multiclasse, in quanto converte l'output del modello in una distribuzione di probabilità su tutte le classi, consentendo di attribuire a ciascuna classe la probabilità di appartenenza.

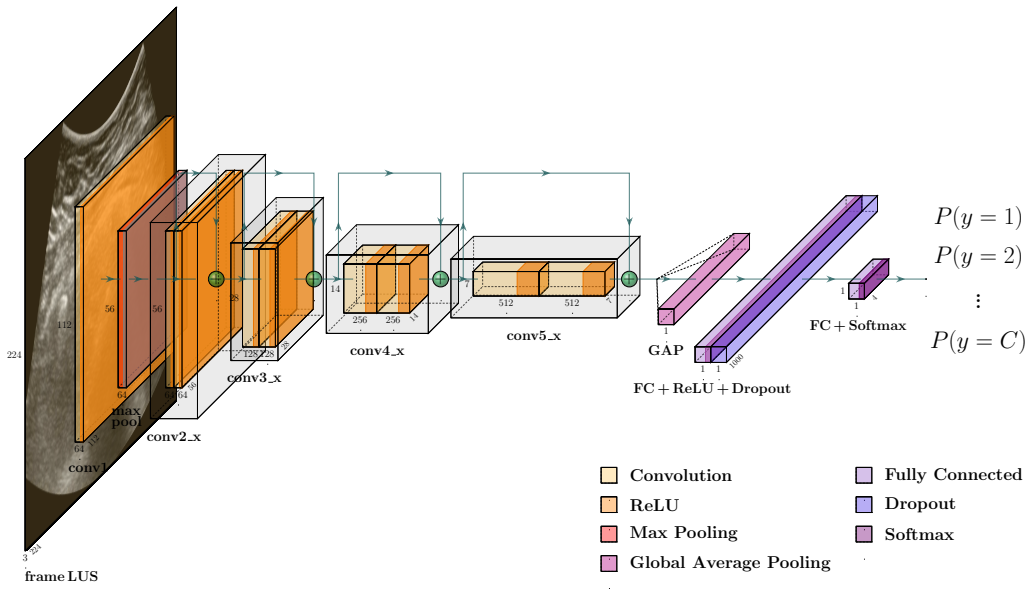


Figura 3.6: Architettura nominale costituita dalla rete convoluzionale ResNet18 e una testa di classificazione con layer Fully Connected e uscita Softmax.

Inoltre, nell'ambito delle reti neurali, la funzione di `loss` è il componente che misura la discrepanza tra le previsioni del modello ($\hat{y}$) e le etichette di ground truth (y). L'obiettivo è minimizzare questa discrepanza durante il processo di allenamento per migliorare le prestazioni del modello. In questo approccio nominale alla classificazione multiclasse del dataset `ICLUS-DB`, la funzione di loss utilizzata è la `Categorical Cross Entropy (CCE)`. Questa funzione è definita dalla seguente espressione matematica:

$$L_{CCE}(\hat{y}, y) = -\sum_{i=1}^{C} y_i \log(\hat{y_i}) \ ,$$

dove C è il numero di classi che indicano la dimensione dell'output (4 in questo caso), $\hat{y_i}$ è l'i-esimo valore scalare dell'output del modello e y_i è il valore obiettivo corrispondente (ground truth).

Inoltre, al fine di valutare l'impatto delle funzioni di loss ordinate, è stata anche eseguita un'analisi sperimentale con questa architettura nominale ma utilizzando la loss `Quadratic Weighted Kappa (QWK)`. Questa particolare configurazione dell'architettura di riferimento è stata testata per condurre uno studio d'ablazione (Sezione 4.6) e valutare le prestazioni della rete anche nell'affrontare un compito di classificazione ordinale. La funzione di perdita QWK sarà maggiormente approfondita nella Sezione 3.5.1 dedicata al modello ordinale CLM.

3.5 Reti di classificazione con approcci ordinali

I moduli di classificazione ordinale sono stati integrati nell'architettura unificata (Sezione 3.3) seguendo lo stesso approccio adottato per la testa di classificazione nominale. Due metodi ordinali distinti, il Cumulative Link Model (CLM) e l'Ordinal Binary Decomposition (OBD) sono stati sviluppati in modo separato, mantenendo la modularità del modello. Questi approcci consentono di gestire la classificazione ordinale, considerando l'ordine intrinseco tra le etichette di classe.

3.5.1 Cumulative Link Model (CLM) con funzione di loss QWKc

Per affrontare l'architettura di un modello CLM è necessario definire la formulazione del problema per poi fornire i formalismi utili a definire i passaggi necessari per modellare la soluzione ordinale al problema. Nel contesto di un problema di classificazione ordinale, l'obiettivo è predire l'etichetta y di un vettore di input x, dove $x \in X \subseteq \mathbb{R}^K$ e $y \in Y = \{C_1, C_2, \ldots, C_Q\}$, ovvero x appartiene a uno spazio di input K-dimensionale e y è in uno spazio di etichette di Q diverse classi. Per risolvere questo problema è necessario ricercare una funzione $r : X \to Y$ per prevedere le etichette di nuovi pattern, dato un set di addestramento di N campioni, $D = \{(x_i, y_i), i = 1, \ldots, N\}$.

Come già affrontato nella Sezione 3.1.1, le etichette del dataset `ICLUS-DB` presentano un ordinamento qualitativo naturale: $C_1 \prec C_2 \prec \ldots \prec C_Q$ (nel caso in questione 4 classi totali). I metodi CLM, qui affrontati, si basano sulle probabilità cumulative e introducono un insieme di soglie che separano lo spazio di output in Q classi contigue diverse tenendo conto della scala ordinale. In questo modo, vengono stimate le probabilità cumulative $P(y \preceq C_q|x)$, che possono essere direttamente correlate alle probabilità standard con le seguenti equazioni:

$$P(y \preceq C_q|x) = P(y = C_1|x) + \ldots + P(y = C_q|x) \tag{3.1}$$

$$P(y = C_q|x) = P(y \preceq C_q|x) - P(y \preceq C_{q-1}|x) \tag{3.2}$$

con $q = 2, \ldots, Q-1$, considerando che:

$$P(y = C_1|x) = P(y \preceq C_1|x) \quad \text{e} \quad P(y \preceq C_Q|x) = 1 \tag{3.3}$$

La prima equazione (Eq. 3.1) afferma che la probabilità che l'etichetta y sia minore o uguale a C_q è ottenuta sommando le probabilità di tutte le etichette da C_1 a C_q. In altre parole, è la probabilità cumulativa che l'etichetta sia C_q o meno. Invece la seconda equazione (Eq. 3.2) dice che la probabilità che l'etichetta sia esattamente C_q è ottenuta sottraendo la probabilità cumulativa che l'etichetta sia minore o uguale a C_{q-1} dalla probabilità cumulativa che l'etichetta sia minore o uguale a C_q. In sintesi, rappresenta la differenza tra le probabilità cumulative corrispondenti alle etichette C_q e C_{q-1}. Le assunzioni necessarie affinchè siano valide le equazioni precedenti (Eq. 3.3) affermano: che la probabilità di y di essere esattamente C_1 è uguale alla probabilità cumulativa di y di essere minore o uguale a C_1 e che la probabilità cumulativa che y sia minore o uguale a C_Q è sempre 1, poiché non ci sono etichette superiori a C_Q in un problema ordinale. In sostanza, queste equazioni forniscono una struttura matematica per rappresentare e calcolare le probabilità relative alle diverse etichette in un problema di classificazione ordinale, sfruttando la relazione ordinale tra le classi.

Il modello si ispira alla nozione di variabile latente, dove $f(x)$ rappresenta una mappatura monodimensionale ed è definita come segue: $y^* = f(x)^* = f(x) + \epsilon$, dove ϵ è la componente casuale dell'errore. La scelta più comune per la distribuzione di probabilità di ϵ è la funzione logistica (che è la funzione predefinita per POM), ma possono esserne utilizzate di diverse. L'etichetta C_q viene predetta se e solo se $f(x) \in [b_{q-1}, b_q]$, dove la funzione f e $b = (b_0, b_1, \ldots, b_{Q-1}, b_Q)$ devono essere determinate dai dati. Si assume che $b_0 = -\infty$ e $b_Q = +\infty$, quindi la retta reale definita da $f(x)$, con $x \in X$, è divisa in Q intervalli consecutivi. Ogni intervallo corrisponde a una categoria. I vincoli $b_1 \leq b_2 \leq \ldots \leq b_{Q-1}$ assicurano che $P(y \preceq C_q|x)$ aumenti con q [25].

In questo lavoro vengono considerate tre diverse funzioni di collegamento per la distribuzione di probabilità di ϵ come mostrato in Figura 3.7 (da Vargas et al. [27]):

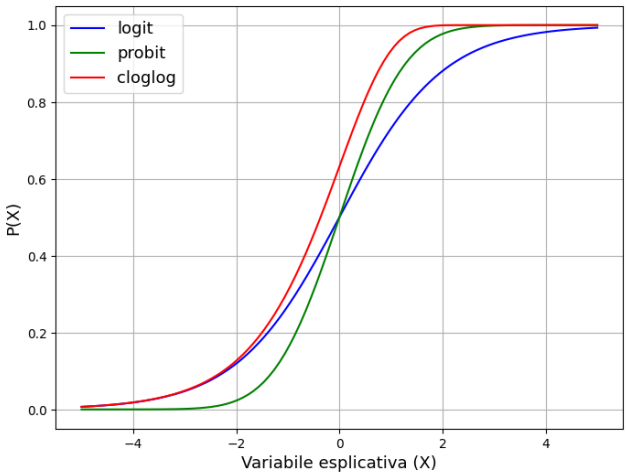


Figura 3.7: Funzioni di collegamento implementate nel modello CLM.

- `logit`: funzione logista utilizzata per POM.

$$P(y \preceq C_q|x) = \frac{1}{1 + e^{-(b_q - f(x))}}, \quad q = 1, \ldots, Q-1,$$

- `probit`: l'inverso della funzione di distribuzione cumulativa normale standard (`cdf`).

$$P(y \preceq C_q|x) = \Phi(b_q - f(x)), \quad q = 1, \ldots, Q-1,$$

  dove Φ rappresenta la funzione di distribuzione cumulativa normale standard.

- `clog-log`: prende una risposta limitata all'intervallo (0,1) e la converte in un valore nell'intervallo $(-\infty, +\infty)$ (come le trasformazioni logit e probit).

$$P(y \preceq C_q|x) = 1 - e^{-e^{b_q - f(x)}}, \quad q = 1, \ldots, Q-1,$$

Si fa notare che la funzione di collegamento `complementary log-log` è asimmetrica. In questo modo, quando la distribuzione dei dati forniti non è simmetrica nell'intervallo [0,1] e aumenta lentamente a valori bassi o moderati ma aumenta rapidamente vicino a 1, i modelli logit e probit non sono appropriati, mentre clog-log può portare a risultati migliori. Se la distribuzione delle caratteristiche nei frame LUS non segue una distribuzione simmetrica nell'intervallo [0,1], significa che alcune caratteristiche o pattern possono variare più rapidamente o lentamente rispetto ad altre. Quindi, se ci sono situazioni in cui i frame LUS mostrano caratteristiche che aumentano rapidamente vicino a 1 (ad esempio, in condizioni specifiche o casi limite), la funzione clog-log potrebbe essere più adatta. Questo perché la clog-log è in grado di gestire asimmetrie e aumenti rapidi in modo più flessibile rispetto alle funzioni logit e probit, che sono simmetriche.

In questo lavoro, la struttura probabilistica dei CLM è proposta come funzione di collegamento per la rete neurale convoluzionale (CNN). Ciò può essere ottenuto definendo un nuovo tipo di layer di output alternativo allo standard layer softmax. In questo modo, il layer di output proposto trasforma la proiezione monodimensionale, precedentemente denominata $f(x)$, in un insieme di probabilità. $f(x)$ è stimato da una trasformazione non lineare dell'insieme di feature apprese dai layer precedenti (Figura 3.8). Per applicare ottimizzatori non vincolati assicurando che $b_1 \leq b_2 \leq \ldots \leq b_{Q-1}$, è possibile ridefinire le soglie. Infatti tutte le soglie possono essere derivate dalla prima nel seguente modo:

$$b_q = b_1 + \sum_{j=1}^{q-1} \alpha_q^2, \quad q = 2, \ldots, Q,$$

dove b_1 è un parametro di apprendimento corrispondente alla prima soglia, α è un vettore di parametri apprendibili utilizzati per ottenere il resto delle soglie, e Q è il numero di classi, che nel caso della classificazione dei danni polmonari da Covid-19 è quattro.

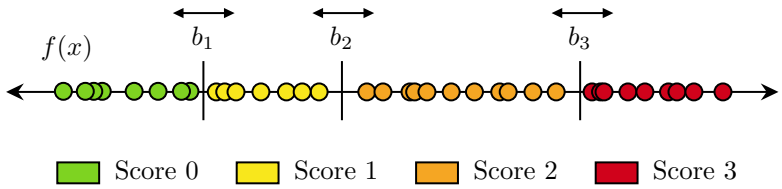


Figura 3.8: Proiezione della variabile latente modellata dall'insieme di feature apprese dai layer convoluzionali nello spazio 1D $f(x)$ partizionato da $Q-1$ soglie.

Per implementare il modulo di classificazione CLM, è stata creata una classe personalizzata estendendo la classe `Layer` di `Keras`. Questa classe incorpora il modello del *Proportional Odds Model (POM)* e utilizza la libreria `TensorFlow Probability` per gestire la distribuzione normale. Di seguito viene fornita una descrizione ad alto livello di come è stata implementata la classe. La costruzione del layer è guidata dai parametri specifici del modello, come il numero di classi (`num_classes`), la funzione di collegamento (`link_function`), l'opzione di utilizzo del parametro regolarizzatore τ (`use_tau`) e i parametri per le soglie (`thresholds_a` e `thresholds_b`).

A tempo d'inferenza, i valori dei pesi e dei parametri vengono utilizzati per convertire le proiezioni dei dati in probabilità cumulative. Come descritto sopra, sono state implementate tre funzioni di collegamento, ognuno delle quali modella in modo specifico la relazione tra le proiezioni e le probabilità cumulative.

Per collegare il modulo CLM alla parte convoluzionale, è stato mantenuto l'approccio di architettura unificata (Figura 3.9). Il risultato della rete convolutiva rappresentato dal layer `GAP` è passato attraverso il layer densamente connesso standard (`FC`), seguito dal layer di regolarizzazione `Dropout` (con tasso variabile dipendente dall'esperimento). Successivamente l'output corrente viene passato ad un secondo ed

ultimo strato `FC` caratterizzato da un singolo neurone in quanto fornisce la proiezione del modello in uno spazio unidimensionale. Il layer di output è il `CLM`, il modello a soglia implementato come funzione di collegamento. In mezzo all'ultimo layer completamente connesso e il CLM è stato inserito un layer di batch normalization (`BN`) che è risultata fondamentale per il corretto funzionamento. Questo processo è stato integrato nella struttura della rete neurale unificata, garantendo che il modulo CLM sia coerente con le caratteristiche della parte convoluzionale come mostrato in Figura 3.9.

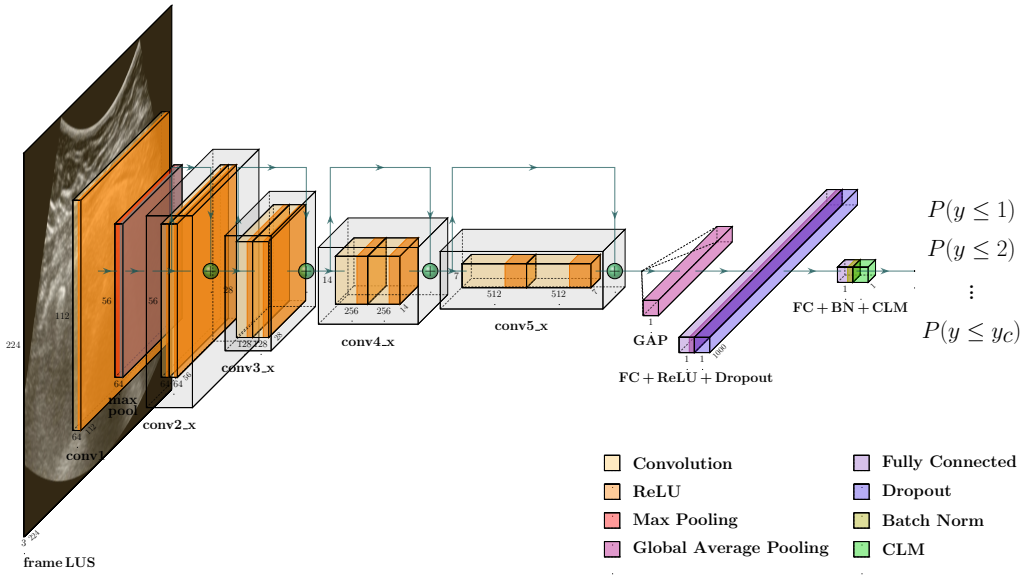


Figura 3.9: Architettura ordinale costituita dalla backbone convoluzionale ResNet18 e una testa di classificazione che incorpora il modulo CLM.

Al fine di migliorare le prestazioni del modello ordinale, la struttura CLM nel layer di output è combinata con la versione continua della funzione di loss `QWK` [32]. L'indice di Kappa è una metrica ben nota che misura l'accordo tra due valutatori diversi. Il Kappa ponderato (WK) [33] si basa sull'indice Kappa e assegna diversi pesi ai diversi tipi di disaccordi in base a una matrice di pesi. È utile per valutare le prestazioni nei problemi ordinali, poiché attribuisce un peso maggiore agli errori che sono più lontani dalla classe corretta.

La funzione di loss QWK è definita come segue:

$$QWK = 1 - \frac{\sum_{i,j}^{N} \omega_{i,j} O_{i,j}}{\sum_{i,j}^{N} \omega_{i,j} E_{i,j}}$$

dove N è il numero di campioni valutati, ω è la matrice di penalizzazione (in questo caso, sono considerati pesi quadratici $\omega_{i,j} = (i-j)^2/(C-1)^2$, $\omega_{i,j} \in [0,1]$), O è la matrice di confusione, $E_{i,j} = \frac{O_{i,\cdot} O_{\cdot,j}}{N}$, $O_{i,\cdot}$ è la somma della i-esima riga e $O_{\cdot,j}$ è la somma della j-esima colonna.

Il QWK definito sopra non può essere utilizzato come funzione di perdita per l'algoritmo di ottimizzazione poiché non è continuo. Tuttavia, è stato ridefinito dal lavoro di de la Torre et al. [32] in termini di probabilità delle predizioni:

$$QWK_c = \frac{\sum_{k=1}^{N}\sum_{q=1}^{Q}\omega_{t_k,q}P(y=C_q|x_k)}{\sum_{i=1}^{Q}\frac{N_i}{N}\sum_{j=1}^{Q}\left(\omega_{i,j}\sum_{k=1}^{N}P\left(y=\mathcal{C}_j \mid \mathbf{x}_k\right)\right)}$$

Dove $QWK_c \in [0,2]$, x_k e t_k sono i dati di input e la vera classe del k-esimo campione, Q è il numero di classi, N è il numero di campioni, N_i è il numero di campioni della i-esima classe, $P(y=C_q|x_k)$ è la probabilità che il k-esimo campione appartenga alla classe C_q (stimata utilizzando la struttura CLM), e $\omega_{i,j}$ sono gli elementi della matrice di penalizzazione ($\omega_{i,j} = (i-j)^2/(C-1)^2$). Questo passaggio è necessario per rendere la funzione di perdita minimizzabile utilizzando un algoritmo basato sulla discesa del gradiente, requisito mandatorio per poter essere sfruttata in una rete neurale profonda.

3.5.2 Ordinal Binary Decomposition (OBD)

Il secondo approccio ordinale implementato è stato originariamente presentato da Frank e Hall [34], si tratta dell'Ordinal Binary Decomposition (OBD). La formulazione del metodo prevede di scomporre il problema in $Q-1$ sottoproblemi decisionali binari. Ciascun problema $q \in Q$ consiste nel decidere se $y > C_q$ condizionato a x $1 \leq q < Q$. Questo richiederebbe normalmente $Q-1$ modelli diversi, ognuno che risolva uno di questi sottoproblemi binari. Questo implicherebbe di calcolare a priori ogni probabilità $p_q = P(y=C_q)$ basata sui modelli ottenuti e quindi selezionare la classe con la probabilità più alta. Le probabilità individuali vengono calcolate come funzione delle probabilità cumulative, $P(y > C_q)$, stimate dai modelli binari:

$$p_1 = P(y=C_1) = 1 - P(y > C_1),$$

$$p_q = P(y=C_q) = P(y > C_{q-1}) - P(y > C_q) \;\; \forall 1 < q < Q,$$

$$p_Q = P(y=C_Q) = P(y > C_{Q-1}).$$

Tuttavia, sono associati diversi problemi a questo approccio poiché le uscite delle diverse decomposizioni non vengono combinate nello stesso processo di addestramento. Ciò significa che le assunzioni di probabilità di base, che includono l'ordine delle probabilità ($P(y > C_q) \geq P(y > C_q + 1)$), la positività delle probabilità ($p_q \geq 0$) e la somma delle probabilità che deve essere uguale a 1 ($\sum_q p_q = 1$) non sono necessariamente soddisfatte, il che può portare a incongruenze nelle previsioni del modello. Per comprendere meglio il fenomeno, si consideri il task di classificazione LUS in quattro categorie: Score 0, Score 1, Score 2 e Score 3. L'approccio OBD decomporrebbe questo problema in tre sottoproblemi binari: (1) $\mathbb{P}(y > Score\,0 \,|\, x)$, (2) $\mathbb{P}(y > Score\,1 \,|\, x)$ e (3) $\mathbb{P}(y > Score\,2 \,|\, x)$. Le criticità emergeranno nel momento

in cui si vorranno combinare le previsioni di questi sottoproblemi distinti per ottenere la previsione finale della classe. Poiché i modelli binari sono addestrati separatamente come da approccio originale [34], l'ordine delle probabilità potrebbe non essere coerente, creando situazioni in cui il modello crede erroneamente che una classe sia più probabile di un'altra. Ad esempio, tradotte le uscite cumulative in probabilità standard, si potrebbe avere una previsione di Score 1 con una probabilità più alta di Score 2, cosa che è per definizione incoerente con l'ordine naturale delle classi. Per superare questa limitazione, viene proposto un compromesso: addestrare un singolo modello convoluzionale per poi essere alimentato in più blocchi completamente connessi, ognuno risolvendo un singolo sottoproblema di classificazione binaria [30]. In questo modo l'addestramento può essere compiuto in parallelo per ogni modulo di classificazione. L'uscita di ciascuno dei $Q-1$ blocchi completamente connessi ha una funzione di attivazione sigmoidale rappresentante la probabilità $o_k = P(y > C_k|x) \in (0,1)$. Una rappresentazione grafica della decomposizione binaria la si può apprezzare con lo Schema 3.10, che mostra le tre uscite di un modello OBD a quattro classi $C_0, C_2, \ldots, C_3$ come nel caso in questione.

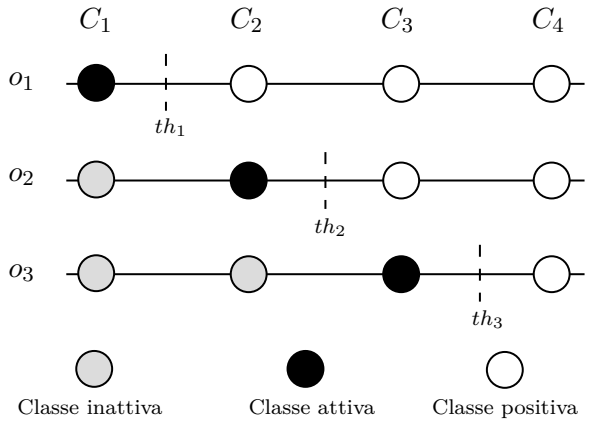


Figura 3.10: Decomposizione binaria del task di classificazione ordinale LUS a quattro classi in tre sottoproblemi binari.

Le probabilità cumulative sono rappresentate dalle uscite o_1, o_2 e o_3. L'uscita o_1 rappresenta l'esito del primo sottoproblema binario, che confronta la classe C_1 con le restanti tre. A loro volta, le proiezioni o_2 e o_3 rappresentano le ultime due uscite che confrontano rispettivamente le classi C_2 e C_3 con le restanti. Questa è una decomposizione ordinale poiché le classi adiacenti sono raggruppate insieme (tramite le soglie th_q raffigurate come linee tratteggiate verticali) ad eccezione della prima classe, che, per definizione del problema, presenta una relazione differente con le altre classi. I valori delle uscite indicano la probabilità cumulativa, nella forma descritta da o_k, che un campione x appartenga alla classe attiva C_q (disco nero) o quelle successive $C_j \; \forall \, j > q$ (dischi bianchi), ignorando le precedenti (disco grigio). Ad esempio, un campione con un valore elevato di o_2 avrà un'alta probabilità di appartenere alla classe C_3 (Score 2) oppure alla classe C_4 (Score 3).

Successivamente per ottenere le probabilità finali, viene utilizzata un'opzione più stabile proposta recedentemente da Barbero-Gómez et al. [35] basata sulla funzione decisionale del framework `Error-Correcting Output Codes (ECOC)`. Il corretto codice di uscita ideale per ogni classe è considerato come le coordinate di un vertice di un ipercubo in $Q-1$ dimensioni. Ad esempio, per un problema ordinale a 4 classi come quello affrontato in questa tesi, le classi C_1, C_2, C_3, C_4 sarebbero associate ai codici $(0,0,0)$, $(1,0,0)$, $(1,1,0)$ e $(1,1,1)$, rispettivamente. In questo modo, tutte le uscite del modello sono considerate per la classificazione e per decidere a quale classe appartiene un campione x, viene selezionata la classe con il codice più vicino secondo una qualche funzione di distanza d:

$$\hat{y} = \arg \min_{1 \leq q \leq Q} d(o, c_q)$$

dove $o = (o_1, o_2, \ldots, o_{Q-1})$ è il vettore dei valori di output, e c_q è il vettore di codice associato alla classe C_q. Il processo è illustrato con un esempio in Figura 3.11 tramite la rappresentazione grafica dell'ipercubo 3D del framework ECOC. Il vertice più vicino a o, il vettore delle probabilità cumulative estratte con OBD, è $v(C_2)$, ergo il campione x verrà assegnato alla classe corrispondente: $\hat{y} = C_2$.

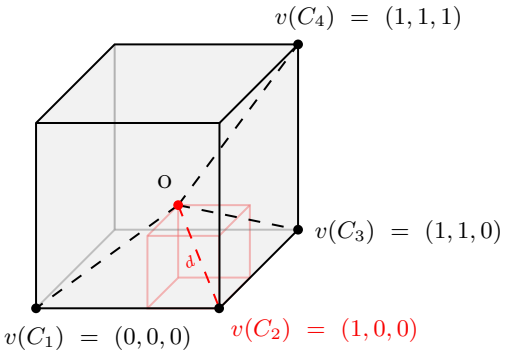


Figura 3.11: Rappresentazione 3D dell'ipercubo nel framework ECOC che mostra il vettore di output del modello per un campione (punto rosso), i vettori ideali delle classi (linee tratteggiate) e la distanza minore che assegna l'etichetta (linea tratteggiata rossa).

Il criterio di ottimizzazione globale della rete è la funzione di loss `Mean Squared Error (MSE)` definita come di seguito:

$$\ell(x_i) = \frac{1}{Q-1} \sum_{k=1}^{Q-1} \left(1\{y_i > C_k\} - P(y_i > C_k | x_i)\right)^2$$

La formulazione può sembrare diversa dalla classica forma che coinvolge direttamente i valori predetti ($\hat{y}_i$) e i valori di ground truth (y_i). Tuttavia, questa formulazione è comune nelle situazioni in cui la previsione non è un valore continuo, ma una distribuzione di probabilità.

Durante la classificazione di nuovi campioni, viene invece utilizzata la norma L_2 come metrica di distanza d, poiché si allinea con il criterio di ottimizzazione MSE:

$$\hat{y} = \arg \min_{1 \leq q \leq Q} ||o - c_q||^2$$

Per implementare l'Ordinal Binary Decomposition (OBD), il modello è strutturato sfruttando una rete neurale convoluzionale come backbone di features extractor, seguita da uno strato di layer aggiuntivi che compongono l'architettura OBD. La funzione Python principale che implementa il metodo, riceve l'output dell'ultimo layer convoluzionale come input e costruisce gli strati densi che si diramano definendo i vari sottoclassificatori (come mostrato in Figura 3.12). Questi strati includono uno layer completamente connesso (`FC`) per ogni coppia di classi ordinate (a eccezione della prima classe), strati di `Dropout` per evitare l'overfitting, uno strato di output denso per ciascuna coppia di classi e strati di `Batch Normalization (BN)` per migliorare la stabilità del modello. Le funzioni di attivazione `Leaky ReLU (LReLU)` e `Sigmoid` sono applicate per gestire non linearità e generare probabilità comprese tra 0 e 1. L'output finale del modello è una concatenazione delle probabilità di appartenenza alle classi ordinate. OBD affronta l'ordinamento delle classi adiacenti, consentendo al modello di gestire la complessità delle relazioni ordinali e restituire il vettore d'uscita o che include le probabilità cumulative calcolate. Così come per il CLM, il modulo OBD è stato integrato nella rete neurale unificata, compatibile con le backbone convoluzionali disponibili.

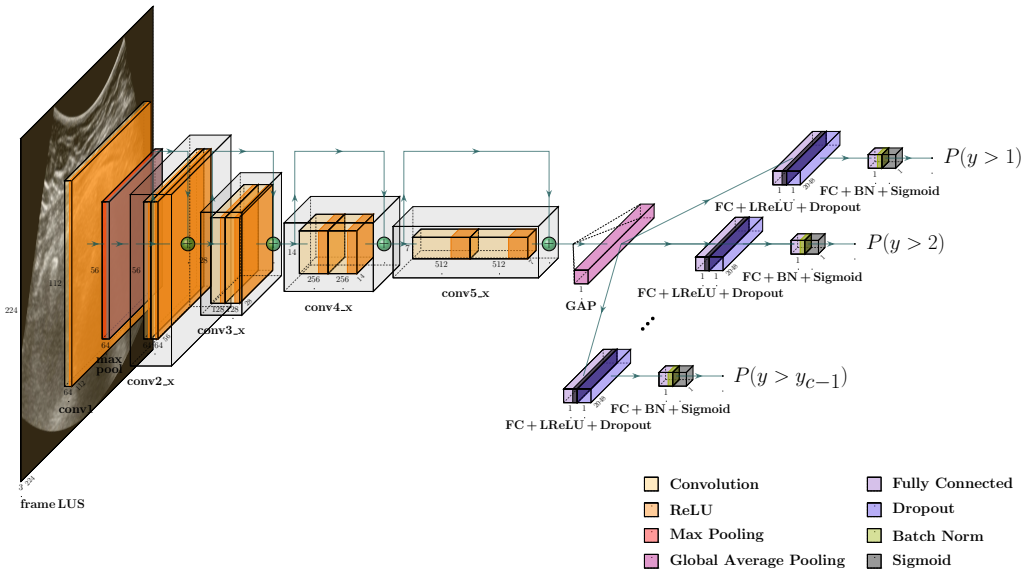


Figura 3.12: Architettura ordinale costituita dalla backbone convoluzionale ResNet18 e decomposizione binaria OBD.

3.6 Framework sperimentale

Il framework proposto è stato progettato e sviluppato per automatizzare il processo di valutazione dei modelli neurali sul dataset `ICLUS-DB`. Si basa su un approccio sistematico, basato sull'esecuzione di quelli che vengono definiti *"esperimenti"*, dove ogni *esperimento* rappresenta una configurazione specifica di un modello neurale. Gli esperimenti sono definiti attraverso un file di configurazione `JSON`, nei quali sono specificate le impostazioni per il training e la valutazione dei modelli. Queste impostazioni includono il modello della rete neurale da impiegare tra CLM, OBD o ResNet nominale di riferimento (nel caso degli approcci ordinali anche la backbone convoluzionale), il numero di epoche con cui effettuare l'allenamento, l'ottimizzatore utilizzato, gli iperparametri, la funzione di loss e altro ancora. La definizione degli esperimenti verrà approfondita a dovere nella Sottosezione 3.6.1.

La fase di *Training* è eseguita mediante `Cross-Validation (CV)`, garantendo un flusso di lavoro robusto evitando il fenomeno dell'*overfitting*. Durante questa fase, vengono generate le matrici di confusione, che costituiscono una griglia delle predizioni rispetto ai veri valori, consentendo una valutazione approfondita degli errori del modello nelle diverse classi del problema. Insieme a questo sono estratte e salvate anche le mappe di attivazione `GradCAM`, che forniscono una visualizzazione a mappa di calore delle regioni dei frame LUS in cui il modello si focalizza durante l'apprendimento. Inoltre, vengono generati i grafici dell'alleamento che rapportano la loss in funzione delle epoche, offrendo una rappresentazione visiva dellà bontà del processo di apprendimento del modello.

Dopo la fase di training, segue quella di *Testing*, dove il modello viene testato sul fold di test e le performance sono valutate attraverso la generazione della matrice di confusione. Le metriche di valutazione, sia nominali che ordinali, sono calcolate e salvate in un file CSV dedicato. Questo approccio consente una chiara tracciabilità delle prestazioni di ogni modello, agevolando il confronto tra diverse configurazioni e strategie di training. Il framework è progettato per supportare l'esecuzione seriale di numerosi esperimenti in modo efficiente. I risultati generati da ciascun modello vengono successivamente elaborati attraverso un'apposito script dedicato all'aggregazione, mirato a sintetizzare un resoconto unificato organizzato per modelli neurali. Per ciascun modello, vengono calcolate le medie e le deviazioni standard di ogni metrica calcolata, fornendo così una panoramica statistica completa delle performance di ciascun approccio. Questa aggregazione semplifica l'analisi comparativa tra i diversi modelli testati alla ricerca di evidenze che mostrino che gli approcci ordinali siano effettivamente in grado di portare un contributo positivo alla classificazione dei danni da Covid-19 nelle ecografie polmonari.

Il framework (Figura 3.13) è stato concepito per automatizzare completamente tutte le fasi coinvolte, garantendo un processo di valutazione efficiente e riproducibile. La sua struttura modulare e scalabile è progettata per consentire l'espansione del framework, facilitando l'inclusione di nuovi modelli senza particolari complicazione.

Questa flessibilità rappresenta un elemento chiave per adattare il framework a eventuali sviluppi futuri nel campo della classificazione di frame LUS.

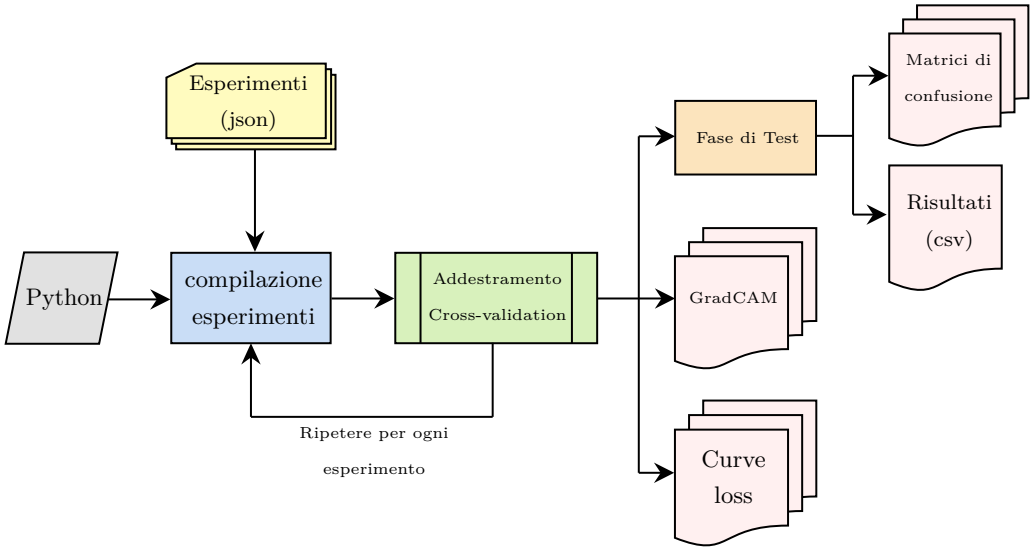


Figura 3.13: Framework proposto per automatizzare il processo di addestramento e valutazione dei modelli neurali.

Il progetto è stato sviluppato integralmente in `Python`, un linguaggio di programmazione altamente noto, particolarmente flessibile e adatto all'ambito dell'apprendimento automatico. Per quanto riguarda l'ambiente di sviluppo, inizialmente durante le fasi di acquisizione e conversione del dataset, è stato impiegato `PyTorch`. Tuttavia, successivamente il progetto è passato a `Tensorflow` con `Keras`, utilizzando la versione `2.11.0`. Questa scelta è stata motivata dalla necessità di adattarsi all'ambiente d'esecuzione e di supportare la retrocompatibilità, un aspetto fondamentale in un contesto open source. Per lo sviluppo del codice, è stato utilizzato `Visual Studio Code (VSCode)`, un ambiente di sviluppo integrato (IDE) con funzionalità avanzate di debugging e una vasta gamma di estensioni. Durante la fase iniziale di gestione del dataset, anche `Google Colab` è stato impiegato per sfruttare la potenza di calcolo offerta dalla piattaforma. Il codice sorgente è accessibile tramite un repository open source su GitHub [1]. Quanto all'esecuzione delle reti neurali, il progetto ha beneficiato dell'infrastruttura del cluster del gruppo *VRAI (Vision Robotics Artificial Intelligence*) dell'*Università Politecnica delle Marche (UNIVPM)*. Questo cluster conta 2 processori `Intel Xeon Silver 4214`, quindi 12 core per socket e 2 thread per core, offrendo un totale di 48 CPU. Inoltre, la presenza della GPU dedicata `NVIDIA GeForce RTX 2080 Ti` ha notevolmente contribuito a tempi di inferenza efficienti, permettendo una gestione ottimale delle complesse operazioni richieste durante l'addestramento e la valutazione dei modelli neurali.

[1] `https://github.com/edoardo-conti/iclus-ordinal-classification`

3.6.1 Configurazione degli esperimenti

L'architettura sperimentale del framework per la valutazione dei modelli neurali nel contesto del dataset `ICLUS-DB` si basa su una robusta configurazione degli esperimenti, dove ogni aspetto del processo di addestramento, validazione e testing è attentamente definito. Questo approccio fornisce una struttura flessibile e personalizzabile per esplorare diverse modalità di apprendimento e valutare le prestazioni dei modelli.

Gli esperimenti sono codificati in un singolo file `JSON` che racchiude una lista di definizioni. Ogni definizione di esperimento all'interno della lista costituisce una specifica configurazione di un modello neurale, che può essere ordinale o nominale, a seconda delle necessità dell'analisi.

Per ogni esperimento sono implementati una serie di parametri configurabili (Tabella 3.3), per la maggior parte condivisi tra gli esperimenti ad eccezione di quelli dipendenti dall'architettura della rete neurale. Innanzitutto, il parametro `nn_model` identifica il modello neurale da utilizzare tra le opzioni disponibili come ResNet18, OBD (Ordinal Binary Decomposition) o CLM (Cumulative Link Model). La scelta del modello incide ovviamente sull'approccio impiegato, ordinale o nominale. Per OBD e CLM è possibile specificare la backbone convoluzionale mediante `nn_backbone`, determinando così la struttura di partenza del modello prima di applicare la decomposizione binaria ordinale oppure il modulo CLM. Passando ai parametri dipendenti dall'architettura, per quanto riguarda OBD è possibile specificare la profondita dei layer fully connected (FC) per ogni testa di classificazione (`hidden_size`). Per il Cumulative Link Model è invece possibile scegliere la funzione di collegamento (`link_function`) e se attivare o meno il contributo τ (`use_tau`). Passando ai parametri condivisi tra gli esperimenti, il numero di fold per la cross-validation è definito da `folds`, consentendo una robusta validazione incrociata per valutare l'efficacia del modello su diverse suddivisioni del dataset. Parametri come `epochs`, `batch_size`, e `dropout` influenzano il processo di addestramento, definendo rispettivamente il numero di epoche, la dimensione della batch di frame LUS fornita ad ogni step e la probabilità di dropout durante il training. La presenza o assenza di Online Data Augmentation (`augmentation`) può impattare notevolmente la capacità del modello di generalizzare i pattern nei dati di test, con importanti implicazioni sulla sua affidabilità in scenari reali. Generalmente questo parametro è sempre attivo a meno di situazioni limite particolari. La scelta della funzione di loss (`loss`) e delle metriche di valutazione (`metrics`) determina gli obiettivi di apprendimento del modello e le metriche attraverso cui misurare le sue prestazioni. La funzione di loss dipende dall'approccio e dalla rete. Invece in ottica di comparazione dei modelli, le metriche da testare è auspicabile che siano le stesse per tutti gli esperimenti (se calcolabili), in modo da avere tutti i metri di paragone possibili. La selezione dell'ottimizzatore (`optimizer`) e la definizione del tasso di apprendimento (`learning_rate`) sono aspetti critici per l'efficace convergenza del modello durante il processo di ottimizzazione. Ulteriori parametri, come `weight_decay` e `momentum`, sono personalizzabili e

forniscono ulteriori strumenti per regolare altri aspetti dell'apprendimento.

In questo modo, la configurazione degli esperimenti offre un ampio margine di personalizzazione, consentendo una dettagliata modellazione delle configurazioni dei modelli neurali in base alle esigenze specifiche della valutazione sul dataset `ICLUS-DB`. Riassumendo, questi sono tutti i parametri personalizzabili per definire un esperimento:

Parametri	**Descrizione**	**Note**
`nn_model`	modello della rete neurale	-
`nn_backbone`	CNN come feature extractor	solo per CLM e OBD
`folds`	numeri di fold per la CV	-
`epochs`	epoche per l'addestramento	-
`batch_size`	dimensione della batch LUS	-
`dropout`	tasso di dropout	-
`hidden_size`	unità degli strati densi (FC)	solo per OBD
`link_function`	funzione di collegamento	solo per CLM
`use_tau`	regolarizzatore POM	solo per CLM
`augmentation`	attivazione data augmentation	-
`loss`	loss da minimizzare	-
`metrics`	metriche da misurare	-
`optimizer`	ottimizzatore della rete	-
`learning_rate`	tasso d'apprendimento	-
`weight_decay`	regolarizzatore pesi	-
`momentum`	acceleratore gradiente	-

Tabella 3.3: Parametri configurabili implementati per definire un "esperimento".

3.6.2 Cross-Validation ad-hoc per ICLUS-DB

La Cross-Validation (CV) rappresenta una metodologia fondamentale nell'ambito del Machine Learning, mirando a fornire una valutazione robusta e generalizzata delle prestazioni dei modelli neurali. In questo contesto, la CV è essenziale per mitigare le sfide legate alle variazioni nei dati e alla loro eterogeneità, consentendo una stima accurata delle capacità predittive dei modelli.

Una prima motivazione che ha spinto ad adottare una validazione incrociata sta proprio nel *gap* in letteratura, legato appunto ai metodi di validazione dei risultati. Lo stato dell'arte della classificazione di ecografie polmonari ha adottato

approcci di holdout semplici, suddividendo il dataset in set di allenamento e test [7][8]. Tale mancanza è rilevante in quanto potrebbe non fornire una valutazione completa e generalizzata delle prestazioni del modello, poiché i risultati possono variare considerevolmente a seconda dello split dei dati. L'indisponibilità di uno split specifico e le indicazioni vaghe in letteratura hanno ulteriormente motivato l'approccio indicato, offrendo allo stesso tempo affidabilità e generalizzazione.

Inoltre, la decisione di implementare la CV è stata motivata da un secondo fattore emerso durante le prime fasi della ricerca: la variabilità significativa dei risultati ottenuti da differenti split dei dati, pur facendo sempre attenzione ad ottenere distribuzioni il più simili possibile a quella del dataset completo. Questa instabilità può essere attribuita alla eterogeneità nella spartizione dei frame tra i set di training, testing e validation, derivante dalla diversità dovuta dai centri medici e di conseguenza dagli strumenti d'acquisizione utilizzati da essi. Una tale variabilità nei risultati poteva portare a conclusioni non robuste e poco generalizzabili.

Quindi, la Cross-Validation è stata scelta come approccio metodologico in grado di superare questa sfida. Il processo di CV, implementato con attenzione ai dettagli e alle peculiarità del dataset `ICLUS-DB`, mira a garantire che le analisi siano intrinsecamente robuste, indipendenti dalle possibili variabilità nei dati. Questo approccio, intrinsecamente *split-agnostic*, rappresenta un contributo alla letteratura esistente, superando le limitazioni evidenziate nei metodi precedenti nel contesto della classificazione di ecografie polmonari.

L'applicazione della Cross-Validation nel dataset `ICLUS-DB` si scontra con sfide significative a causa della sua struttura gerarchica e della necessità di evitare il fenomeno del *Data Leakage*. Questo si riferisce al rischio di includere dati appartenenti allo stesso paziente in set diversi. Questa situazione comprometterebbe la capacità del modello di generalizzare a nuovi dati, poiché si troverebbe ad affrontare casi simili a quelli utilizzati nell'addestramento. Ciò potrebbe portare a valutazioni ottimistiche delle prestazioni del modello, distorcendo la sua effettiva capacità di adattamento a nuovi pazienti e compromettendo la validità delle analisi sperimentali. Per evitare il data leakage, è pertanto essenziale adottare procedure di suddivisione dei dati che preservino l'indipendenza tra le osservazioni nei set di addestramento e test. Il dataset può infatti essere suddiviso a diversi livelli, ciascuno con le proprie caratteristiche:

- `Frame-level`: l'uso diretto di frame LUS, l'unità di dati più fine, è ovviamente problematico in quanto rende il rischio di data leakage altamente probabile;
- `Hospital-level`: questo livello di aggregazione assicura che i frame di uno stesso paziente rimangano all'interno dello stesso centro medico, separando quest'ultimi tra i vari set;
- `Patient-level`: aggregando i frame di uno stesso paziente, la CV garantisce che il modello venga testato su dati non visti appartenenti a pazienti differenti;

- `Video-level`: quest'ultima suddivisione, che aggrega tutti i frame nella sequenza d'acquisizione d'appartenenza, non assicura che dati dello stesso paziente non finiscano su set diversi indicando che è non è sufficiente come metodo di aggregazione.

La natura aggregata del dataset `ICLUS-DB`, suddivisibile a diversi livelli gerarchici, introduce sfide significative nello sviluppo di uno splitting accurato e stratificato.

In primo luogo, l'approccio tradizionale di campionamento casuale di una percentuale specifica di dati (ad esempio, il 70% dei frame per il *training set*) diventa impraticabile a causa della necessità di mantenere l'integrità dei dati almeno a di livello di paziente. In altre parole, non è possibile selezionare casualmente un certo numero di frame LUS dal dataset, poiché devono essere considerati i gruppi d'appartenenza (pazienti).

In secondo luogo, l'obiettivo di realizzare uno splitting stratificato, che conservi la distribuzione delle classi del dataset completo, diventa una sfida non banale. La necessità di rispettare le aggregazioni e garantire che ciascun set contenga una rappresentazione bilanciata di classi patologiche specifiche richiede un'attenta progettazione della procedura di splitting.

Di seguito lo schema di Cross-Validation attentamente concepito *ad-hoc* per il task di classificazione dei danni da Covid-19 tramite ecografie polmonari:

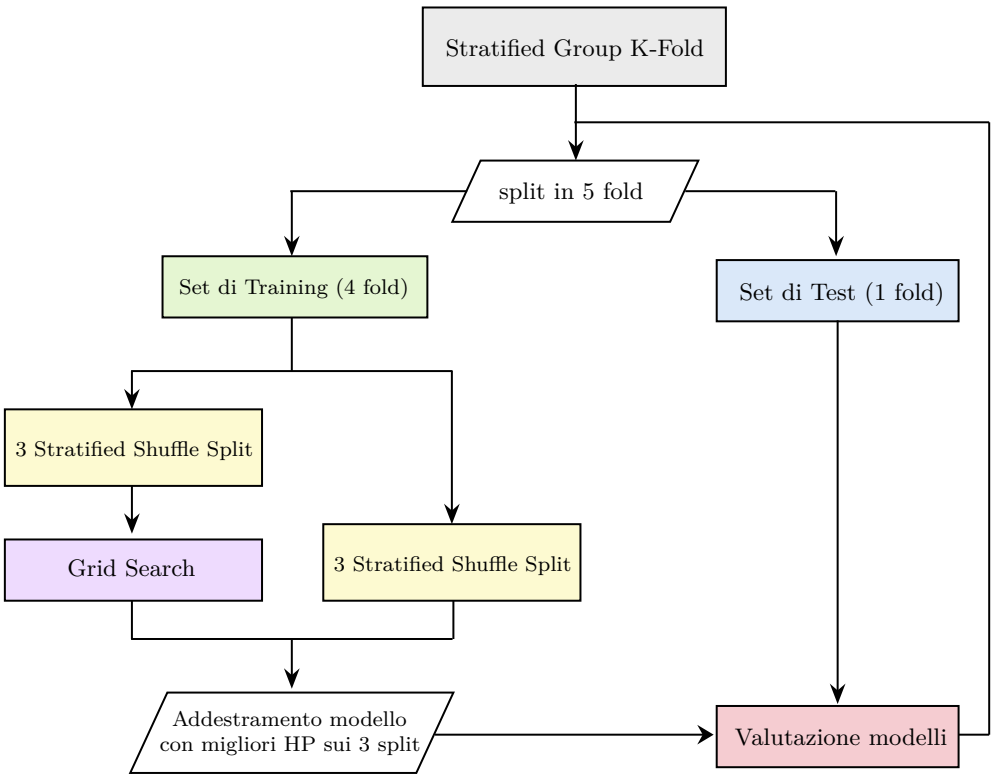


Figura 3.14: Schema di Cross-Validation progettato ad-hoc per il dataset ICLUS.

Per approfondire lo schema mostrato in Figura 3.14 si affronta il processo suddividendolo in fasi distinte, ognuna delle quali svolge un ruolo fondamentale nell'assicurare un'analisi affidabile e robusta dei modelli proposti.

1. `Stratified Group K-Fold`: i dati da partizionare sono i 277 video da dividere in 5 fold, 4 formano il *training set* e 1 il *test set*. I pazienti fungono da gruppi per la suddivisione. Le etichette vengono campionate per ogni video in modo da supportare la stratificazione;

2. `Stratified Shuffle Split`: in ciascun set di training, i video associati ai pazienti sono sottoposti ad un holdout stratificato 85/15 a livello di paziente. Ciò permette di creare un *set di validation* (15% dei pazienti) e di training (85% dei pazienti) in modo coerente. Vengono generati 6 split diversi, 3 per il Grid Search e i restanti 3 dedicati alla la fase di training;

3. `Grid Search`: viene eseguita una ricerca a griglia per determinare la combinazione ottimale di *iperparametri*, tenendo conto dello spazio specifico definito per l'esperimento (Sezione 3.6.3). Ogni configurazione è allenata per 10 epoche, valutando il `Mean Absolute Error (MAE)` come metrica di riferimento. Infine calcolando il MAE medio tra i fold viene selezionata la combinazione di iperparametri che ha minimizzato l'`Average MAE (AMAE)`;

4. `Training`: in questa fase la rete viene allenata con la combinazione di iperparametri ottimale sui 3 set ad essa dedicati, generati tramite Stratified Shuffle Split;

5. `Evaluation`: i modelli addestrati vengono valutati sul *test set*, che corrisponde al fold di test estratto all'inizio mediante lo Stratified Group K-Fold.

In ottica di riproducibilità, i vari split del dataset effettuati durante la fase di Cross-Validation sono configurabili attraverso l'utilizzo di un `seed`. L'impostazione di un seme consente di rendere gli splitting deterministici e ripetibili. Ciò significa che, utilizzando lo stesso seed in diversi esperimenti, la cross-validation genererà sempre gli stessi split. Questo approccio è fondamentale per garantire che i vari modelli siano testati sulle stesse configurazioni di dati durante gli esperimenti, eliminando la variabilità derivante dai diversi split. Tale rigore metodologico contribuisce significativamente alla coerenza e affidabilità delle comparazioni tra i modelli e i risultati ottenuti.

Date le restrizioni introdotte per la suddivisione, lo Stratified Group K-Fold si è rivelato essere la strategia più efficace. La partizione avviene a livello di video (`Video-level`), dove i 277 video ecografici devono essere suddivisi in un numero K di fold, in questo caso pari a 5. Ciascun fold rappresenta una porzione del dataset contenente diversi video appartenenti a diversi pazienti.

Per gestire il componente *"Group"* dello Stratified Group K-Fold e rispettare il requisito che evita il fenomeno del data leakage, viene creata una lista di 277 elementi,

dove ogni posizione corrisponde a un video specifico e ogni elemento rappresenta il paziente a cui appartiene il video associato. L'introduzione del concetto *"Stratified"* è finalizzata a garantire la conservazione della distribuzione delle etichette, inizialmente assegnate a livello di frame, durante la suddivisione. Questo coinvolge il calcolo di uno score rappresentativo per ciascun video, ottenuto mediante la *moda* degli score dei frame appartenenti a quel video.

Combinando la suddivisione a livello di video, l'assegnazione di gruppi e lo stratified sampling, si ottiene una procedura complessa, ma robusta, per generare fold stratificati e, al contempo, rispettare la suddivisione dei pazienti, evitando così il rischio di data leakage (Figura 3.15).

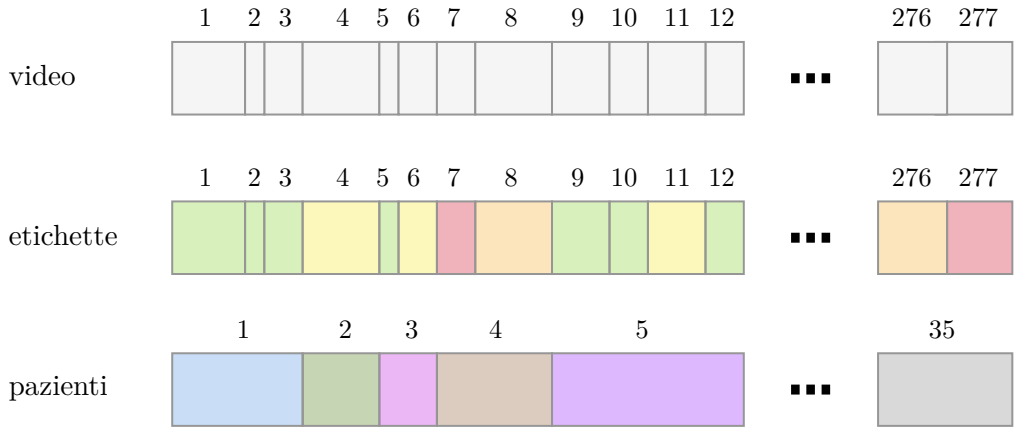


Figura 3.15: Schematizzazione dello Stratified Group K-Fold per la CV.

Per quanto riguarda invece il secondo approccio di *splitting*, per dividere il training set (formato dall'unione dei 4 fold) in modo da ricavare anche il set di validation, questo prevedeva un classico holdout random con una suddivisione del 90/10 a livello di paziente. Tuttavia, tale approccio poteva risultare in divisioni fortemente sbilanciate e, in casi estremi, generare set di dati privi delle label minoritarie. Per superare questi problemi, si è optato per lo Stratified Shuffle Split, un'alternativa più avanzata che consente di mantenere la stratificazione e garantire distribuzioni bilanciate delle label. Poiché non esiste un metodo "Stratified *Group* Shuffle Split", è necessario tornare a livello di paziente. Tuttavia, questo processo comporta la sfida di campionare un paziente con un singolo punteggio basato sugli score di tutti i frame di tutti i suoi video, introducendo ovvi errori di approssimazione. Lo Stratified Shuffle Split implementato opera nel seguente modo: dato un paziente P con un insieme di video $V = \{V_1, V_2, ..., V_m\}$, dove ciascun video V_i è composto da un numero variabile di frame classificati con uno score da 0 a 3, si calcola il suo `bincount` complessivo BC_P nel seguente modo:

$$BC_P = \sum_{i=1}^{m} BC_{V_i}$$

dove BC_{V_i} è il `bincount` del video V_i (vettore 4×1 del conteggio delle classi).

Successivamente, il BC_P viene ponderato con il `class weight` del dataset completo CW, ovvero la distribuzione originale delle classi codificata in un vettore di pesi (più è alto un valore e minore è la rappresentazione della classe corrispettiva nel dataset), tramite una moltiplicazione punto a punto:

$$BC_{P,w} = BC_P \odot CW$$

Questa operazione è volta a conferire maggiore peso alle classi minoritarie, prevenendo così la situazione in cui non siano assegnati almeno due pazienti per ogni score, garantendo una suddivisione equa tra i due set risultanti.

Infine, l'estrazione della classe rappresentativa C_P avviene attraverso un'operazione di arg max:

$$C_P = \arg\max BC_{P,w}$$

dove arg max restituisce l'indice dell'elemento massimo nel vettore ponderato $BC_{P,w}$, che corrisponde alla classe (score) con cui verrà campionato il paziente P per proseguire con la stratificazione degli split randomici. Questo ha permesso di formulare un metodo che generasse training set e validation set di dimensioni il più possibile congrue a quanto impostato e con una distribuzione delle classi il più comparabile possibile a quella del dataset `ICLUS-DB`.

3.6.3 Tuning degli iperparametri con Grid Search

Questa fase, inclusa nel processo di CV per ogni fold di ogni esperimento eseguito, è volta a individuare la combinazione ottimale di iperparametri per massimizzare le performance dei modelli. La ricerca richiede la definizione dello *spazio degli iperparametri* (Tabella 3.4), che si riferisce alla specifica dei range e dei valori che saranno esplorati per ciascun parametro. Per la configurazione di ResNet18, vengono esplorati vari valori per la dimensione delle batch (`Batch Size, BS`), il tasso di apprendimento (`Learning Rate, LR`) e il dropout (`Dropout Rate, DR`). Successivamente, i due modelli ordinali presentano i propri parametri insieme a quelli definiti per ResNet18. L'Ordinal Binary Decomposition, in particolare, condivide gli stessi parametri di ResNet18, aggiungendo il numero di neuroni completamente connessi degli strati nascosti da dividere per ogni sottoproblema (`Hidden Size`). Questo parametro definisce il numero totale di neuroni per l'intera architettura. Poiché l'approccio OBD coinvolge Q-1 classificatori (dove Q rappresenta il numero di classi del problema, in questo caso 4), si specifica il numero di neuroni che sarà suddiviso per 3. Il risultato ottenuto rappresenterà il numero di neuroni Fully Connected (FC) per ciascuno degli strati connessi di ogni sottoclassificatore. Infine per CLM, oltre ai parametri di ResNet18, si esplorano diverse funzioni di collegamento (`Link Function`) e l'utilizzo del parametro regolarizzatore τ (`Tau`). Riassumendo vengono esplorati 3 parametri per ResNet18, 4 per OBD e 5 per CLM.

Modelli	**Parametri**		
`ResNet18`	**Batch Size**	**Learning Rate**	**Dropout Rate**
	$[8, 16, 32, 64]$	$[1e^{-2}, 1e^{-3}, 1e^{-4}, \text{CDR}]$	$[0.0, 0.1, 0.2, 0.3, 0.4]$
`OBD`	`ResNet18 HPs` +	**Hidden Size**	
		$[768, 1536, 3072, 6144]$	
`CLM`	`ResNet18 HPs` +	**Link Function**	**Tau**
		[logit, probit, cloglog]	[true, false]

Tabella 3.4: Spazio degli iperparametri esplorato nella fase di Grid Search all'interno della CV per ogni modello.

Una nota riguardo al valore `CDR` del parametro `Learning Rate (LR)`. Si tratta di un metodo di scheduling del learning rate noto come `Cosine Decay Restart`. Implementato con Keras, utilizza un approccio che regola dinamicamente il tasso di apprendimento durante l'addestramento della rete neurale. La funzione segue una forma cosinusoidale, diminuendo gradualmente il learning rate con il passare delle epoche. Quando viene raggiunto un certo numero di epoche, il processo "riparte" e ricomincia da capo, creando cicli di adattamento del tasso di apprendimento. Il learning rate iniziale è fissato a $1e^{-3}$, con il primo decadimento programmato al 30% del numero totale di *steps*. Le ripartenze riprendono con un LR pari al 90% del valore precedente. Questo metodo può contribuire a migliorare la convergenza della rete neurale e per questo motivo è stato esplorato il suo contributo.

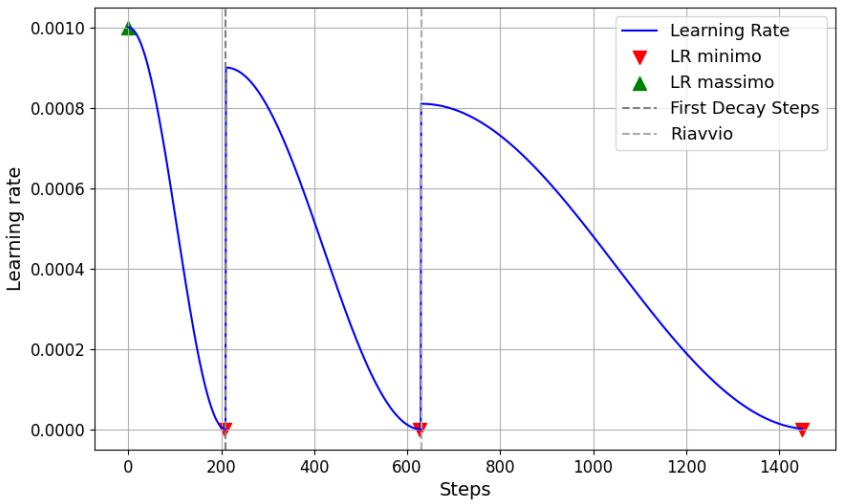


Figura 3.16: Schema del processo di scheduling del Learning Rate (LR) con Cosine Decay Restart (CDR).

Sottraendo i parametri esplorati nella fase di grid search dai parametri che definiscono gli esperimenti (Tabella 3.3), si identificano i seguenti parametri che rimangono costanti durante questa fase di ottimizzazione e condivisi da tutti gli esperimenti condotti:

- Numero di fold: 5
- Numero di epoche: 100
- Data Augmentation: Online, definita nella Sottosezione 3.1.3
- Metriche di valutazione: definite nella Sottosezione 3.6.4
- Ottimizzatore: `Stochastic Gradient Descent (SGD)`
- Weight decay: $1e^{-6}$
- Momentum: 0.9

Il numero ottimale di fold per la suddivisione dell'intero dataset a livello video è 5, poiché si adatta meglio alla proporzione ideale di frame per ciascun set. Sebbene 3 fosse praticabile, sperimentalmente 5 ha dimostrato di produrre distribuzioni più adeguate. Nella scelta dell'ottimizzatore, invece, si è preferito lo *Stochastic Gradient Descent*, anche se sono state esplorate alternative come *Adam*. L'utilizzo di Adam tendeva a convergere frequentemente verso minimi locali, richiedendo un adattamento dei parametri che risultava oneroso. Inoltre, le impostazioni di weight decay e momentum sono ignorate quando il learning rate è impostato con lo scheduling CDR. L'obiettivo della fase di Grid Search è individuare, per ciascun fold, la combinazione di iperparametri che minimizza l'AMAE, assicurando prestazioni ottimali in ogni configurazione. La tabella risultante riporta i valori specifici di ciascun iperparametro che conducono a tale minimizzazione, evidenziando la configurazione più comune scelta per l'addestramento vero e proprio dei modelli. È essenziale evidenziare che, poiché si tratta di un processo di K-Fold Cross-Validation con 5 fold, la tabella riflette la combinazione di parametri migliore più frequente osservata tra i diversi fold. Questa scelta tiene conto delle possibili variazioni nei dati che possono contribuire a leggere differenze nelle combinazioni ottimali di iperparametri tra i vari fold.

Modelli	**Parametri**					
	BS	**LR**	**DR**	**Hidden Size**	**Link Function**	**Tau**
`ResNet18`	32	CDR	0.2	-	-	-
`OBD`	32	CDR	0.3	6144	-	-
`CLM`	32	$1e^{-2}$	0.3	-	cloglog	true

Tabella 3.5: Combinazione di iperparametri più comune che minimizza l'AMAE nelle fasi di Grid Search all'interno della CV per ogni modello.

3.6.4 Metriche di valutazione

Questa sottosezione illustra le metriche di valutazione utilizzate per valutare le prestazioni di ciascun modello nel set di test. Le metriche selezionate forniranno un quadro dettagliato delle capacità predittive dei diversi approcci, facilitando la comparazione e la valutazione dei risultati ottenuti in ogni esperimento.

Metriche nominali

Nel contesto della classificazione multiclasse, il *Correct Classification Rate (CCR)* solitamente rappresenta il criterio più rilevante. Il CCR misura la percentuale di campioni correttamente classificati rispetto al numero totale di campioni nel dataset. Un CCR più alto indica una migliore precisione nella classificazione dei campioni ed è definito come di seguito:

$$\mathrm{CCR}\,(\uparrow) = \frac{1}{N}\sum_{i=1}^{N} 1\left\{\widehat{y}_i = y_i\right\},$$

dove N è il numero totale di campioni nel set di test, $\hat{y}_i$ è l'etichetta predetta per il campione e y_i è la vera classe d'appartenenza.

Tuttavia, in presenza di sbilanciamento delle classi, il CCR potrebbe perdere di significato. Infatti, in scenari con classi minoritarie, il CCR potrebbe essere dominato dalle performance sulla classe maggioritaria. Pertanto si calcola anche l'*F1-Score*, che è definito come la media armonica tra *Precision* e *Recall*, fornendo una misura complessiva delle prestazioni bilanciata tra Falsi Positivi (FP) e Falsi Negativi (FN). Per calcolare l'F1-Score di un set di dati multiclasse, viene utilizzata una tecnica *one-vs-all (OvA)* per calcolare i punteggi individuali per ogni classe nel set di dati. Nello specifico la *micro-average* consiste nel calcolare la precisione e la recall per ciascuna classe e quindi combinare questi risultati per ottenere una misura complessiva. In termini matematici, il *Micro-averaged F1-Score* è calcolato come di seguito:

$$\text{Micro-averaged F1-Score}\,(\uparrow) = \frac{\sum_{i=1}^{C} 2 \cdot \mathrm{Precision}_i \cdot \mathrm{Recall}_i}{\sum_{i=1}^{C} \mathrm{Precision}_i + \mathrm{Recall}_i},$$

dove C è il numero totale di classi nel problema di classificazione, $\mathrm{Precision}_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i+\mathrm{FP}_i}$ è la precisione della classe i e $\mathrm{Recall}_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i+\mathrm{FN}_i}$ è la recall della classe i.

Per gli stessi motivi, anche la sensibilità minima *(Minimum Sensitivity, MS)* può essere monitorata come metrica sensibile allo sbilanciamento delle classi. La metrica MS valuta la sensibilità minima tra tutte le classi, ovvero la probabilità di un risultato positivo condizionato al fatto che il campione sia veramente positivo (*True Positive Rate, TPR*), evidenziando così la classe con la performance più bassa. L'equazione è così definita:

$$\mathrm{MS}\,(\uparrow) = \min\left\{S_c = \frac{O_{cc}}{O_c}, \quad c = 1, \ldots, C\right\},$$

dove O è la matrice di confusione, C è il numero di classi nel problema, O_{cc} rappresenta il numero di campioni della classe c correttamente classificati, O_c è il totale dei campioni della classe c e S_c è la sensibilità della classe c.

Metriche ordinali

Le metriche CCR, F1-Score e MS non considerano quanto ogni previsione si discosti dalla verità, essendo progettate principalmente per problemi di classificazione nominale, dove tutti gli errori sono penalizzati allo stesso modo. Infatti, per i problemi di classificazione ordinale, come quello affrontato in questa tesi, sono più adatte metriche che tengono conto della distanza della predizione dalla realtà, in cui un errore di una classe è più accettabile di un errore di due classi. Quindi le metriche ordinali implementate sono le *Accuracy K-Off (Acc. 1-Off e Acc. 2-Off)*, il coefficiente di correlazione di *Spearman* (r_s) e l'indice di Kappa quadratico pesato *(Quadratic Weighted Kappa, QWK)*. Nello specifico, le metriche Acc. 1-Off e Acc. 2-Off valutano rispettivamente la percentuale di campioni etichettati con una classe distante massimo 1 o 2 dalla verità e sono definibili con un'unica formulazione:

$$\text{Acc. K-Off}\,(\uparrow) = \frac{1}{N}\sum_{i=1}^{N} 1\{\hat{y}_i \in K_i\}, \quad K_i = 1, \ldots, C-1\,,$$

dove N è il numero totale di campioni nel set di test, $\hat{y}_i$ è l'etichetta predetta per il campione i, K_i rappresenta l'insieme delle etichette corrette entro una distanza k dalla verità per il campione i e $1\{\hat{y}_i \in K\}$ è una funzione che restituisce 1 se $\hat{y}_i$ è contenuto in K e 0 altrimenti.

Il coefficiente di Spearman r_s è una misura statistica che valuta quanto bene l'ordine delle previsioni del modello si allinea con l'ordine delle etichette reali. La sua formula coinvolge la covarianza delle posizioni ordinali delle coppie di dati, normalizzata dai prodotti delle deviazioni standard delle posizioni ordinate delle due variabili ed è così definita:

$$\mathrm{r_s}\,(\uparrow) = \frac{\mathrm{Cov}(y, \widehat{y})}{\sigma_y \sigma_{\hat{y}}}\,,$$

dove $Cov(y, \hat{y})$ è la covarianza tra le verità e le previsioni e σ_y e $\sigma_{\hat{y}}$ sono le deviazioni standard.

L'indice QWK è invece una metrica di concordanza che considera la possibilità di accordo casuale tra le etichette previste e quelle effettive. Questo indice è particolarmente utile in contesti di classificazione ordinale in cui l'accordo su classi distanti dovrebbe essere penalizzato maggiormente rispetto all'accordo su classi più vicine. La sua formula coinvolge la somma dei pesi quadratici delle differenze tra le previsioni e le etichette reali, normalizzata dalla somma totale dei pesi. Gli accordi perfetti e le discordie casuale sono entrambi considerati nella sua valutazione. La formulazione

matematica è di seguito fornita:

$$\text{QWK}\,(\uparrow) = 1 - \frac{\sum_{i=1}^{Q}\sum_{j=1}^{Q} w_{ij}O_{ij}}{\sum_{i=1}^{Q}\sum_{j=1}^{Q} w_{ij}E_{ij}}\,,$$

dove w_{ij} è il costo del disaccordo quando $y = C_i$ e $\hat{y} = C_j$ ($w_{ij} = |i - j|$), O_{ij} è l'accordo osservato ed E_ij è l'accordo atteso dovuto dal caso. In sintesi, mentre il coefficiente di correlazione di Spearman valuta la relazione monotona tra variabili ordinali, l'indice di Kappa Quadratico Pesato misura l'accordo tra le previsioni del modello e le etichette reali considerando pesi quadratici per differenze specifiche.

Metriche per la misura dell'errore

È importante tenere sotto controllo anche metriche che valutano l'errore. In particolare, il *Mean Absolute Error (MAE)* che considera la differenza assoluta tra le previsioni e le etichette senza distinzione di direzione, fornendo così una valutazione robusta delle prestazioni del modello rispetto alla natura ordinale delle classi. Definito come di seguito:

$$\text{MAE}\,(\downarrow) = \frac{1}{N}\sum_{i=1}^{N} |\widehat{y}_i - y_i|\,,$$

dove N è il numero totale di campioni nel set di test, $\hat{y}_i$ è l'etichetta predetta per il campione e y_i è la vera classe d'appartenenza.

Allo stesso modo, l'utilizzo del *Root Mean Squared Error (RMSE)* può aggiungere ulteriori dettagli sulla dispersione degli errori, anche se è fondamentale notare che il MAE è spesso preferito in contesti ordinali per la sua interpretazione più intuitiva. La formula per il calcolo dell'RMSE è la seguente:

$$\text{RMSE}\,(\downarrow) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

dove N rappresenta il numero totale di campioni, y_i è il valore vero del campione i e $\hat{y}_i$ è il valore predetto per il campione i.

Successivamente, le performance dei modelli vengono valutate anche attraverso le Curve ROC *(Receiver Operating Characteristic)* e l'area sotto la curva *(Area-Under-the-Curve, AUC)*. Le Curve ROC illustrano la capacità discriminante di un classificatore, visualizzando il trade-off tra la sensibilità (TPR) e il tasso di falsi positivi (FPR). L'AUC-ROC rappresenta la misura dell'efficacia complessiva del modello. Nel contesto della classificazione multiclasse, viene adottata la strategia uno-contro-tutti *(One-vs-the-Rest, OvR)*, considerando separatamente ciascuna classe rispetto alle altre e calcolando infine la media AUC dalle curve ottenute. Un'area sotto la curva (AUC) più ampia indica una maggiore capacità predittiva del modello. Nell'ottica di questa analisi, si cerca di ottenere curve ROC che siano più vicine

posibile all'angolo in alto a sinistra del grafico, riflettendo un'elevata sensibilità e specificità nella classificazione delle diverse categorie di score LUS.

Un ulteriore strumento sono le Matrici di Confusione (*Confusion Matrix*), utili per valutare le performance di un modello di classificazione. Queste matrici mostrano il numero di predizioni corrette e erronee per ciascuna classe, consentendo di identificare gli errori specifici compiuti dal modello. Nelle matrici di confusione, soprattutto in contesti di classificazione ordinale, l'obiettivo è stringere la diagonale, concentrando le predizioni vicino alla verità, e ridurre gli errori distanti. Questo approccio mira a minimizzare gli errori di classificazione che comportano distanze elevate dalla *ground truth*, attribuendo una maggiore importanza alla corretta assegnazione delle classi più prossime. Nell'analisi in questione, sono state generate matrici di confusione per ogni run di valutazione su ciascun fold di test, totalizzando 15 matrici disponibili per ogni esperimento. Questo ci ha fornito una panoramica dettagliata della capacità discriminante dei modelli in diverse configurazioni.

Infine, sono state estratte anche le Mappe di Salienza con metodo `GradCAM` *(Gradient-weighted Class Activation Mapping)*, un'importante risorsa nell'interpretazione delle reti neurali convoluzionali. Queste mappe di calore consentono di visualizzare le regioni dell'immagine che hanno influenzato maggiormente le decisioni della rete durante il processo d'apprendimento. Le GradCAM sono state estratte durante l'addestramento delle reti, rilevando queste attivazioni di classe su dati non visti a intervalli regolari. Questa procedura fornisce l'opportunità di esaminare come le varie reti focalizzano l'attenzione su particolari regioni del frame dell'ecografia polmonare durante il processo di apprendimento.

Capitolo 4

Risultati e Discussioni

In questo capitolo vengono riportati nel dettaglio gli esiti delle sperimentazioni condotte. Attraverso una valutazione accurata dei risultati sperimentali si vuole mettere in luce le prestazioni dei modelli proposti in relazione agli obiettivi prefissati. Le metriche di valutazione utilizzate costituiscono le misure di paragone per confrontare le prestazioni dei diversi approcci implementati.

Ogni esperimento è stato eseguito utilizzando la procedura di K-Fold Cross-Validation (CV) descritta nella Sezione 3.6.2. All'interno di ogni fold (5 per ogni esperimento), è stata eseguita una fase di Grid Search per l'ottimizzazione degli iperparametri. Durante questa fase, vengono lanciati degli addestramenti della durata fissa di 10 epoche per ogni combinazione di iperparametri estratti dallo spazio predefinito degli iperparametri da esplorare. La complessità computazionale è notevole, poiché l'individuazione di un set ottimale richiede la valutazione di numerose combinazioni. Questa complessità aumenta ulteriormente per le architetture ordinali, dato che includono una personalizzazione aggiuntiva rispetto al modello nominale di riferimento.

Il tempo d'inferenza medio di un'epoca dell'architettura ResNet18 nominale, eseguita sulla piattaforma descritta alla fine della Sezione 3.6, varia approssimativamente tra gli $80 - 90$ secondi. Tale intervallo temporale aumenta fino a raggiungere i 2 minuti e 10 secondi nel caso del modello CLM, passando attraverso i 2 minuti del modello OBD. Questo rende la fase di Grid Search un processo notevolmente *time-consuming* rispetto all'effettivo addestramento dei modelli. La fase di training comporta invece l'allenamento di ogni modello sui 3 split stratificati randomizzati per 100 epoche, con possibilità di interruzione precoce in caso di stagnamento della loss.

Complessivamente, ogni esperimento comprende 15 esecuzioni (3 per ciascuno dei 5 fold totali), generando un totale di 60 run per tutti i modelli confrontare tra cui: ResNet18 con CCE e QWK, OBD e CLM. Tale approccio permette di raccogliere dati da diverse configurazioni di splitting dei risultati, fornendo una visione completa delle prestazioni e permettendo di calcolare media ($\overline{x}$) e deviazione standard (σ) delle metriche.

4.1 Prestazioni predittive con backbone addestrata da zero

Nella Tabella 4.1 vengono presentati i risultati sperimentali dei modelli allenati con backbone convoluzionale ResNet18 addestrata da zero (*from scratch*), ovvero con pesi inizializzati randomicamente. La tabella riporta tutte le metriche valutate durante la fase di test, includendo media e deviazione standard come pedice per ciascuna di esse. Con questa configurazione, l'impiego della loss QWK nel modello nominale ha generato risultati insoddisfacenti e di conseguenza sono stati omessi. Un'ipotesi, che sarà poi confermata, è che l'ottimizzazione ordinale del modello ResNet18 nominale senza l'ausilio di pesi preallenati potrebbe faticare a convergere a minimi globali. Tale tendenza suggerisce che l'uso di un criterio d'ottimizzazione ordinale in una ResNet richieda la stabilizzazione fornita da pesi preallenati per raggiungere prestazioni ottimali.

	`ResNet18` (`CCE`)	`OBD` (`MSE`)	`CLM` (`QWK`)
$\overline{\textbf{CCR}}_{SD}$ (↑)	$0.5498_{0.0425}$	$0.5530_{0.0463}$	$\mathbf{0.5735_{0.0753}}$
$\overline{\textbf{F1-Score}}_{SD}$ (↑)	$0.5436_{0.0454}$	$0.5484_{0.0544}$	$\mathbf{0.5697_{0.0826}}$
$\overline{\textbf{1-Off}}_{SD}$ (↑)	$0.9122_{0.0413}$	$0.9476_{0.0325}$	$\mathbf{0.9524_{0.0375}}$
$\overline{\textbf{2-Off}}_{SD}$ (↑)	$0.9943_{0.0062}$	$0.9967_{0.0059}$	$\mathbf{0.9981_{0.0025}}$
$\overline{\textbf{QWK}}_{SD}$ (↑)	$0.6203_{0.0828}$	$0.6579_{0.0742}$	$\mathbf{0.7034_{0.0869}}$
$\overline{\textbf{Spearman}}_{SD}$ (↑)	$0.6324_{0.0869}$	$0.6773_{0.0709}$	$\mathbf{0.7294_{0.0761}}$
$\overline{\textbf{MS}}_{SD}$ (↑)	$0.3140_{0.1196}$	$0.2997_{0.1246}$	$\mathbf{0.3290_{0.1287}}$
$\overline{\textbf{MAE}}_{SD}$ (↓)	$0.5435_{0.0624}$	$0.5025_{0.0581}$	$\mathbf{0.4758_{0.1027}}$
$\overline{\textbf{RMSE}}_{SD}$ (↓)	$0.8572_{0.0855}$	$0.7837_{0.0794}$	$\mathbf{0.7526_{0.1119}}$

Tabella 4.1: Risultati della classificazione ottenuti dai modelli neurali su `ICLUS-DB` con backbone ResNet18 *from scratch*. Le medie migliori sono evidenziate in grassetto, con le deviazioni standard (*SD*) indicate come pedice delle medie.

Analizzando i risultati partendo da quelle che sono le metriche nominali, `CCR` e `F1-Score` forniscono un'indicazione generale delle performance complessive dei modelli. In generale, i risultati mostrano che ResNet18 nominale, OBD e CLM hanno ottenuto valori comparabili con le medie attorno al 55% per entrambe le metriche, ad eccezzione di CLM che guadagna un leggero miglioramento di circa 2 punti percentuali arrivando al 57.3% in CCR e 56.9% in F1-Score. Il tutto con una deviazione standard (*SD*) relativamente bassa, indicando una coerenza nelle prestazioni.

Le metriche più rilevanti per la classificazione ordinale sono però l'`Accuracy 1-Off` e `2-Off`, l'indice `QWK` e il coefficiente di `Spearman` (r_S). Queste metriche valutano

la capacità del modello di assegnare correttamente le etichette tenendo conto della natura ordinale. In questo contesto, sia l'Ordinal Binary Decomposition (OBD) che il Cumulative Link Model (CLM) hanno mostrato prestazioni superiori rispetto al modello di riferimento, indicando una migliore gestione della natura ordinale della classificazione. Infatti, considerando il risultato iniziale ottenuto dalla ResNet18 con Softmax e loss Cross-Entropy Categorica (CCE), che costituisce il punto di riferimento nominale o "baseline", si osserva un punteggio del 91.2% per la metrica Accuracy 1-Off. Gli approcci ordinali migliorano questa prestazione, portandola fino al 95.2% grazie all'utilizzo del CLM. Per quanto riguarda l'Accuracy 2-Off, non si evidenziano miglioramenti sostanziali, poiché la metrica già ottiene buoni risultati con il modello di base, ma si nota comunque un lieve incremento con l'adozione di approcci ordinali. Ciò suggerisce che l'implementazione di metodologie che considerano la natura ordinale del problema contribuisce a ridurre la discrepanza tra gli score predetti e quelli reali. Ad ulteriore conferma di ciò, analizzando le metriche QWK e Spearman, la baseline registra rispettivamente il 62% e il 63.2%. Il primo miglioramento è introdotto dall'OBD, che incrementa le metriche di circa il $3-4\%$, raggiungendo rispettivamente 65.7% e 67.7%. Un notevole salto in avanti è evidenziato dal CLM, che ottiene il 70% per il QWK e il 72.9% per il coefficiente di Spearman. Questo rappresenta un notevole miglioramento rispetto alla baseline dell'8% per QWK e fino all'8.8% per il coefficiente di Spearman. È importante notare che, anche in questo caso, le deviazioni osservate sono coerenti con quelle rilevate per le metriche nominali, confermando la solidità e la coerenza dei risultati.

La metrica per la sensibilità minima (`MS`) è stata esaminata per valutare se i modelli lasciano indietro la classe minoritaria. ResNet18 e CLM hanno mostrato una MS comparabile e discretamente buona rapportata al dominio, suggerendo una maggiore sensibilità alla classe meno frequente. Al contrario, OBD ha ottenuto valori di MS inferiori, indicando una potenziale tendenza a trascurare le classi minoritarie, seppur si tratti comunque di una differenza poco marcata. In questo caso la deviazione standard è maggiore per tutti e tre gli esperimenti rispetto alle metriche precedenti, indicando come la sensibilità sia una misurazione influenzabile dallo splitting dei dati. D'altronde è intuibile dato che in questo ambito, variare i frame nei set di training, validation e test alterando la distribuzione delle classi possa influire sulla sensibilità dei modelli.

Le metriche `MAE` e `RMSE` forniscono un'indicazione sugli errori di classificazione. Entrambe le metriche sono state valutate per ciascun modello. Inoltre, il MAE, pur essendo una metrica comunemente associata alla misura dell'errore, può essere considerato anch'esso come una metrica ordinale, poiché tiene conto delle distanze tra le previsioni e i valori effettivi. CLM ha dimostrato di ottenere risultati inferiori in termini di MAE e RMSE rispetto a ResNet18 e OBD. Questo indica una maggiore precisione nel posizionare le predizioni nelle classi corrette. La deviazione standard più elevata per CLM può suggerire una maggiore variabilità nelle prestazioni tra i fold, ma la media più bassa implica una migliore performance complessiva.

Oltre alla presentazione tabellare dei risultati, è stato possibile sfruttare il numero significativo di esecuzioni per ogni esperimento per generare anche dei boxplot delle metriche (Figura 4.1) per una panoramica più dettagliata delle misure di tendenza centrale, dispersione e distribuzione dei dati:

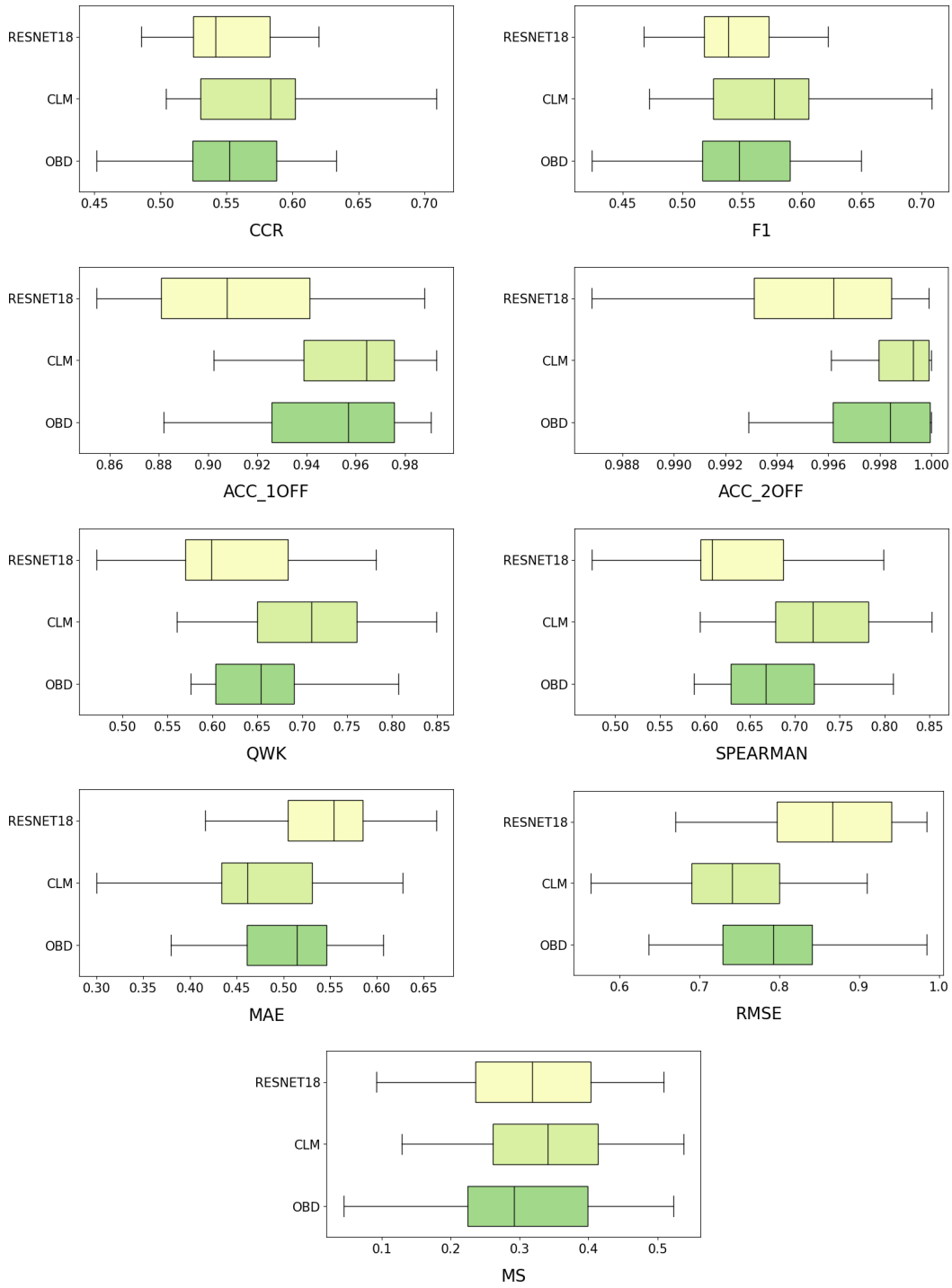


Figura 4.1: Riepilogo grafico dei risultati sperimentali sotto forma di boxplot.

Riassumendo, l'approccio nominale mostra prestazioni complessive discrete e sufficientemente in linea con le aspettative dettatate dalla letteratura. I modelli ordinali, OBD e CLM, si distinguono invece per la loro effettiva capacità di affrontare la natura ordinale del problema. La differenza nelle prestazioni tra OBD e CLM potrebbe essere attribuita alla complessità intrinseca del task. Perciò per quanto riguarda i risultati con backbone non pre-allenata, CLM emerge come il miglior modello sotto ogni aspetto valutato, segnando i risultati medi migliori in ogni metrica. È importante notare che il modello OBD registra prestazioni superiori rispetto all'approccio nominale in quasi tutti gli aspetti, tranne che nella misura della MS, con una differenza di pochi punti percentuali. Tuttavia, questo risultato potrebbe indicare che, con la scalabilità, tale architettura potrebbe trascurare leggermente la classe minoritaria. Infine, le metriche ordinali indicano che OBD e CLM superano la metodologia nominale nelle applicazioni di classificazione ordinale, sottolineando l'importanza di adottare approcci specializzati in contesti di classificazione complessi come questo.

4.2 Impatto del Transfer Learning sulle prestazioni predittive

Successivamente si passa ai risultati ottenuti mantenendo ResNet18 come backbone convoluzionale, ma in questo caso pre-addestrata su ImageNet (IN1k). L'impiego del Transfer Learning ha fornito l'opportunità di addestrare con successo l'architettura di baseline utilizzando anche la funzione di loss ordinale QWK. I risultati sono riportanti nella Tabella 4.2:

	ResNet18 (CCE)	ResNet18 (QWK)	OBD (MSE)	CLM (QWK)
$\overline{\textbf{CCR}}_{SD}$ (↑)	$0.5701_{0.0646}$	$0.5711_{0.0733}$	$\mathbf{0.6094_{0.0639}}$	$0.5951_{0.0355}$
$\overline{\textbf{F1-Score}}_{SD}$ (↑)	$0.5642_{0.0701}$	$0.5620_{0.0737}$	$0.5817_{0.0852}$	$\mathbf{0.5943_{0.0440}}$
$\overline{\textbf{1-Off}}_{SD}$ (↑)	$0.9396_{0.0285}$	$0.9516_{0.0225}$	$0.9539_{0.0124}$	$\mathbf{0.9662_{0.0149}}$
$\overline{\textbf{2-Off}}_{SD}$ (↑)	$0.9874_{0.0251}$	$0.9960_{0.0033}$	$0.9950_{0.0058}$	$\mathbf{0.9982_{0.0029}}$
$\overline{\textbf{QWK}}_{SD}$ (↑)	$0.6529_{0.1088}$	$0.6811_{0.0622}$	$0.7199_{0.0534}$	$\mathbf{0.7310_{0.0378}}$
$\overline{\textbf{Spearman}}_{SD}$ (↑)	$0.6702_{0.1012}$	$0.7037_{0.0581}$	$0.7390_{0.0463}$	$\mathbf{0.7444_{0.0383}}$
$\overline{\textbf{MS}}_{SD}$ (↑)	$0.3288_{0.1138}$	$0.2158_{0.1847}$	$0.2311_{0.1161}$	$\mathbf{0.3503_{0.1240}}$
$\overline{\textbf{MAE}}_{SD}$ (↓)	$0.5027_{0.0896}$	$0.4772_{0.0694}$	$0.4516_{0.0585}$	$\mathbf{0.4404_{0.0411}}$
$\overline{\textbf{RMSE}}_{SD}$ (↓)	$0.8123_{0.1209}$	$0.7569_{0.0569}$	$0.7641_{0.0493}$	$\mathbf{0.7163_{0.0467}}$

Tabella 4.2: Risultati della classificazione ottenuti dai modelli neurali su ICLUS-DB con backbone ResNet18 pre-allenata su ImageNet. Le medie migliori sono evidenziate in grassetto, con le deviazioni standard (*SD*) indicate come pedice delle medie.

L'analisi dei risultati ottenuti partendo con la rete convolutiva pre-addestrata rivela un generale miglioramento nelle prestazioni di tutti i modelli, riconfermando l'efficacia del Transfer Learning nel dominio LUS. Tuttavia, un aspetto degno di nota è l'incremento della dispersione dei risultati evidenziato nei boxplot riportati in seguito (Figura 4.2), che potrebbe indicare una maggiore variabilità nei risultati ottenuti da differenti run.

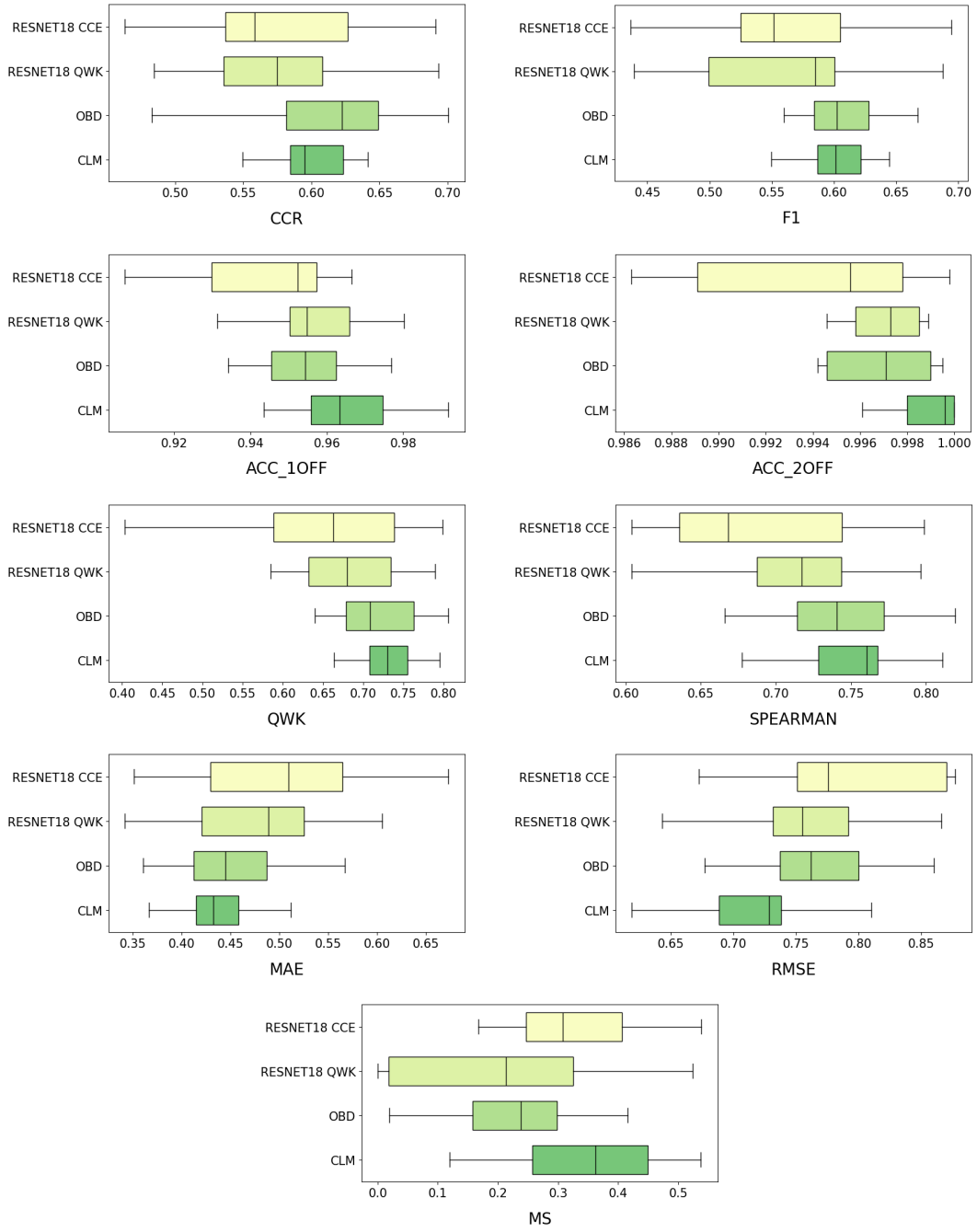


Figura 4.2: Riepilogo grafico dei risultati sperimentali con backbone pre-allenata su ImageNet sotto forma di boxplot.

I miglioramenti delle metriche nominali, come CCR e F1, sono costanti rispetto alla configurazione precedente con la ResNet18 allenata da zero. Nello specifico OBD si guadagna il primo posto toccando un CCR medio di quasi il 61% seguito da CLM, ResNet18 con QWK ed in fine il modello baseline. Invece, per quanto riguarda le metriche ordinali, come QWK e Spearman, anche queste mostrano un effettivo miglioramento, riconfermando l'efficacia dell'impiego di un criterio d'ottimizzazione ordinale. In particolare, l'implementazione della funzione di loss QWK nella ResNet18 pre-addestrata introduce miglioramenti in tutte nelle metriche ordinali valutate. Si segnalano incrementi del +1.5%, +0.9%, +2.8% e +3.4% rispettivamente per le metriche Acc. 1-Off, Acc. 2-Off, QWK e Spearman. È interessante notare che la sensibilità minima (MS) risulta notevolmente compromessa rispetto alla configurazione con CCE (21.6% contro 32.9%), suggerendo che l'utilizzo di QWK come funzione di loss potrebbe penalizzare la classe minoritaria in questo specifico modello. Questa osservazione solleva considerazioni importanti sulla gestione delle classi sottorappresentate e sull'adeguatezza del criterio di ottimizzazione nella contestualizzazione ordinale del problema utilizzando il modello di riferimento. OBD, seppur portando dei miglioramenti generalizzati rispetto a ResNet18 con QWK sotto diversi aspetti, purtroppo soffre dello stesso problema legato alla sensibilità minima bassa, segnando un 23.1% (−9.8% rispetto la baseline).

Nuovamente il CLM si conferma come il modello con prestazioni mediamente superiori. Eccezione fatta per il CCR, il Cumulative Link Model si distingue come il migliore in ogni aspetto, raggiungendo in media il 59.4% nell'F1-Score. Nei confronti delle metriche ordinali, CLM mostra un incremento di oltre 1 punto percentuale nella Accuracy 1-off rispetto ad OBD, un risultato molto positivo. Ottiene il 73.1% e il 74.4% rispettivamente in QWK e Spearman, segnado i valori medi misurati più alti per queste metriche. Finalmente torna a migliorare anche la sensibilità minima (MS) raggiungendo il 35%, presentando quindi un distacco notevole rispetto a ResNet18 con loss QWK e OBD e superando il modello di riferimento (+2.1%). Analogamente, si registra un ulteriore miglioramento negli errori, confermando il contributo positivo del pre-allenamento della parte convolutiva dei modelli neurali.

4.3 Curve AUC-ROC

Sono state estratte le curve `ROC` *(Receiver Operating Characteristic)* rappresentative per ogni modello dai risultati provenienti dallo stesso fold e split interno (Stratified Shuffle Split), assicurando così che le predizioni dei modelli fossero valutate sugli stessi insiemi di dati in ottica di *"fair comparison"*. È importante sottolineare che è possibile generare curve ROC per ogni run di evaluation, ma sono stati scelti a campione un particolare fold e split per tutti i modelli. Quindi per garantire un confronto equo tra gli approcci e ottenere risultati rappresentativi, è stato scelto il secondo split del primo fold come base per la generazione delle curve ROC, delle matrici di confusione e delle mappe di salienza con il metodo GradCAM. Questa selezione è motivata

dalla considerazione che, in generale, gli splitting e i fold generati durante la cross-validation erano sostanzialmente equivalenti ai fini di rappresentazione, eccezione fatta dai casi outlier. Pertanto, la scelta del fold 1 e split 2 rappresenta una decisione pragmatica e accettabile per garantire la coerenza nei confronti tra gli approcci considerati.

Analizzando le curve ROC dei modelli senza pre-addestramento, emerge chiaramente che le classi più complesse, dove le performance declinano rispetto alle restanti, sono la classe 0 ma soprattutto la classe 1. La prima ha un intervallo di valori `AUC` *(Area Under the Curve)* che variano da 84% a 88%. In particolare, la classe 1 è quella che si distingue maggiormente presentando i valori AUC minimi, compresi tra il 69% e il 73%. D'altro canto, le classi 2 e 3 mostrano valori di AUC più elevati, variando tra il 90% e il 98%, dove i valori più alti sono concentrati nella classe 3. Il CLM è il modello con i valori AUC migliori in modo consistente tra le varie classi, seguito da OBD che ha risultati uguali se non maggiori al modello di riferimento ad eccezione della casistica della classe 1.

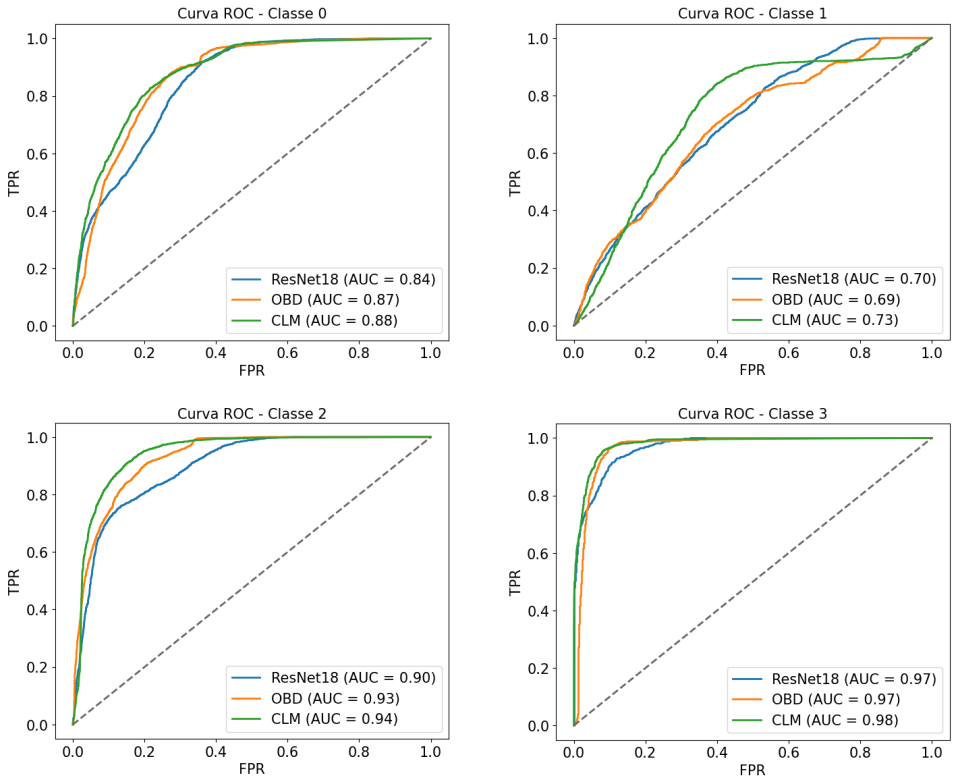


Figura 4.3: Curve AUC-ROC rappresentative per ogni classe estratte da specifici fold e split che confrontano i modelli con backbone *from scratch*.

Passando alle curve ROC ricavate dallo stesso fold e split ma delle run con modelli pre-allenati, le informazioni derivanti sono pressoché simili a quelli precedenti. Nello specifico la baseline ResNet18 con CCE, presenta i valori più bassi di AUC. Nel confronto, ResNet18 con QWK è generalmente superiore al modello di riferimento e

comparabile o inferiore rispetto ai modelli ordinali specializzati. Nell'analisi degli approcci ordinali, risulta evidente che sia OBD che CLM mantengono un livello di performance costantemente superiore rispetto agli altri modelli in quasi tutte le situazioni esaminate. È interessante notare che l'unico contesto in cui entrambi gli approcci registrano una sottoperformance è nella classe 0, dove i risultati si collocano leggermente al di sotto di quelli ottenuti da ResNet18 con loss ordinale. Un'analisi più approfondita rivela che, in particolare nella classe 3, i modelli ordinali contribuiscono in modo significativo, determinando un distacco significativo dalla baseline e persino dalla ResNet18 con loss QWK.

Le curve ROC, nel loro complesso, confermano la solidità e le prestazioni superiori degli approcci ordinali specializzati, con particolare rilevanza per OBD e CLM, nel contesto della classificazione ordinale delle patologie polmonari da Covid-19. Questi risultati consolidano l'efficacia di tali metodologie nell'affrontare la complessità e la natura ordinale intrinseca dei dati LUS, offrendo una prospettiva promettente per l'automatizzazione della diagnosi medica in questo ambito.

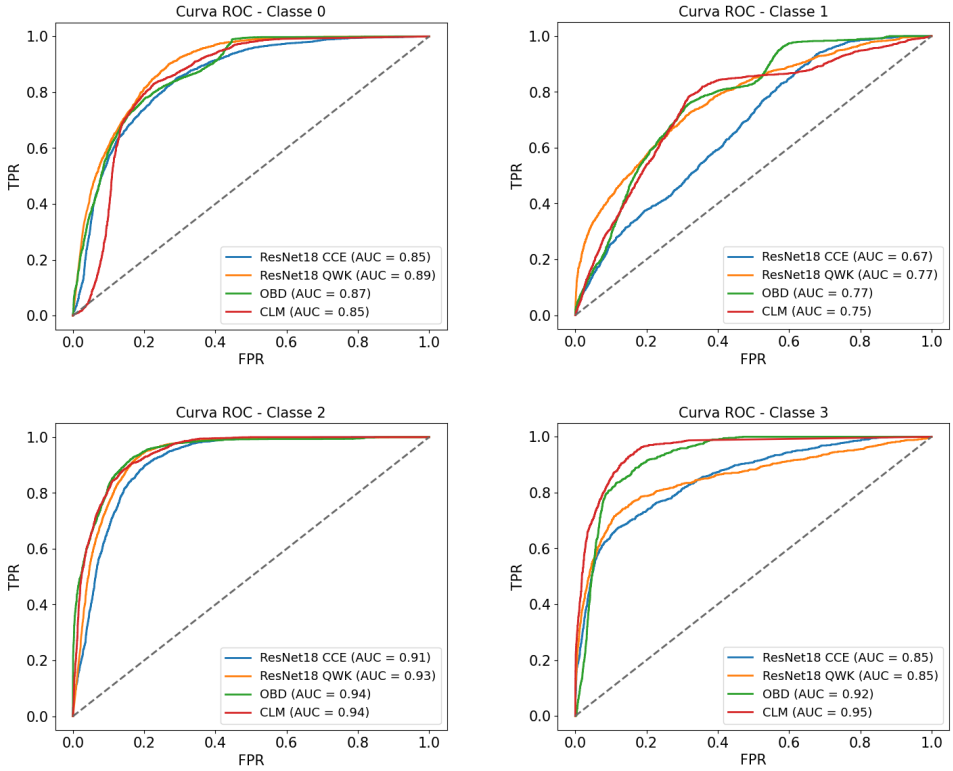
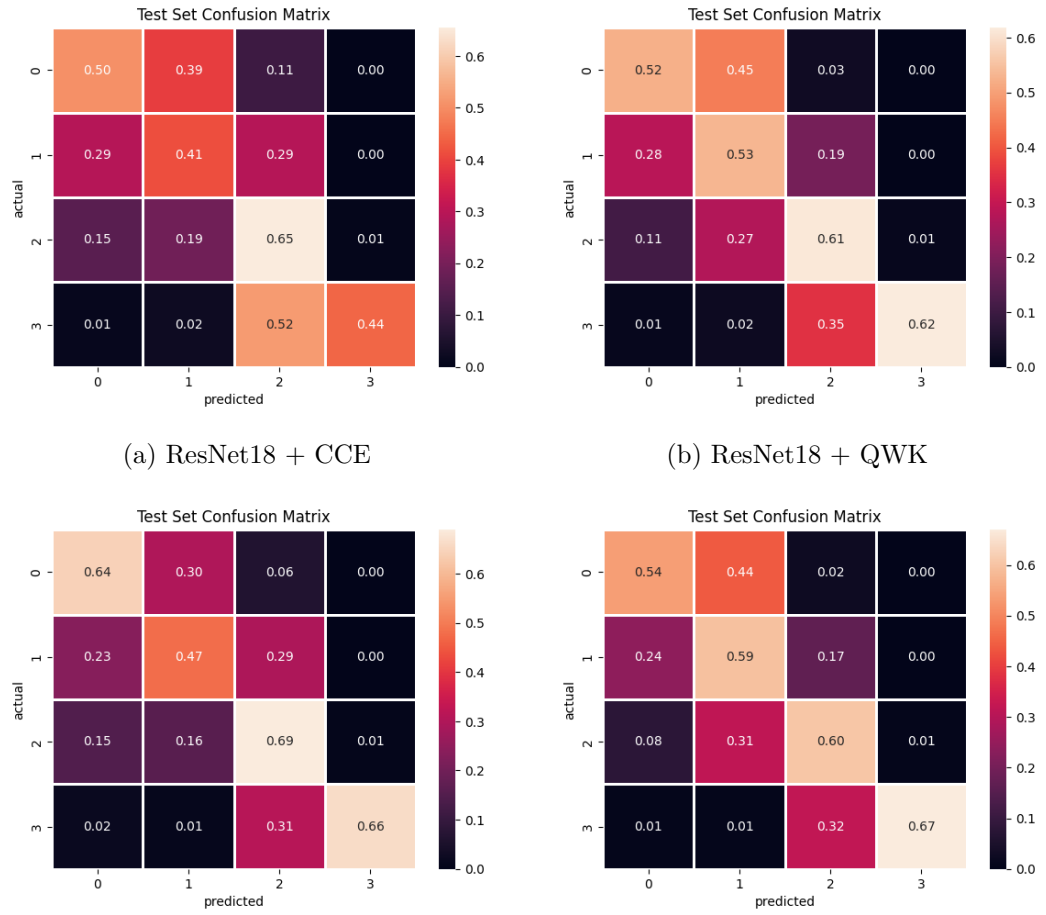


Figura 4.4: Curve AUC-ROC rappresentative per ogni classe estratte da specifici fold e split che confrontano i modelli con backbone pre-addestrata.

4.4 Comparazione approcci tramite Matrici di Confusione

Nel contesto della classificazione ordinale, le matrici di confusione (Figura 4.5) rivestono un ruolo ancora più importante rispetto ai contesti nominali. Questo perchè l'obiettivo principale è "stringere" tutte le predizioni il più possibile verso la diagonale, indicando che il modello effettua correttamente le classificazioni senza commettere errori di distanza significativa dalla verità. Pertanto, una maggiore concentrazione nella zona della diagonale rappresenta un indicatore positivo della capacità del modello di assegnare correttamente le classi secondo la loro gerarchia ordinale. Analogamente alla procedura seguita per ottenere le curve ROC, le matrici di confusione sono calcolate utilizzando le predizioni di uno specifico fold e split, precisamente dal fold 1 e split 2. Questo garantisce che i risultati siano confrontati basandosi sugli stessi insiemi di dati per tutti gli approcci considerati. Inoltre sono stati presi come riferimento solamente i modelli pre-addestrati, perchè sufficienti a livello informativo e le sezioni precedenti hanno assodato il contributo positivo del transfer learning.



(a) ResNet18 + CCE

(b) ResNet18 + QWK

(c) OBD

(d) CLM

Figura 4.5: Confronto tra approcci nominali e ordinali tramite matrici di confusione.

Dal confronto tra matrici si evince che il modello di riferimento, costituito da ResNet18 con loss CCE (Figura 4.5a), ha ottenuto risultati non ottimali e presenta alcune problematiche significative. Ad esempio, dalla prima riga emergono errori nella classificazione del 11% dei frame come score 2 quando, in realtà, erano score 0 (errore di distanza 2). Di particolare rilievo è però il notevole 15% di score 2 classificati erroneamente come score 0, comunque un errore di distanza 2 ma che sottolinea un fenomeno di sottodiagnosi, ergo potenzialmente rischioso nella pratica clinica. Inoltre, nelle fasi avanzate della patologia, si osserva che la maggior parte degli score 3 è stata valutata come score 2, indicando una classificazione non auspicabile.

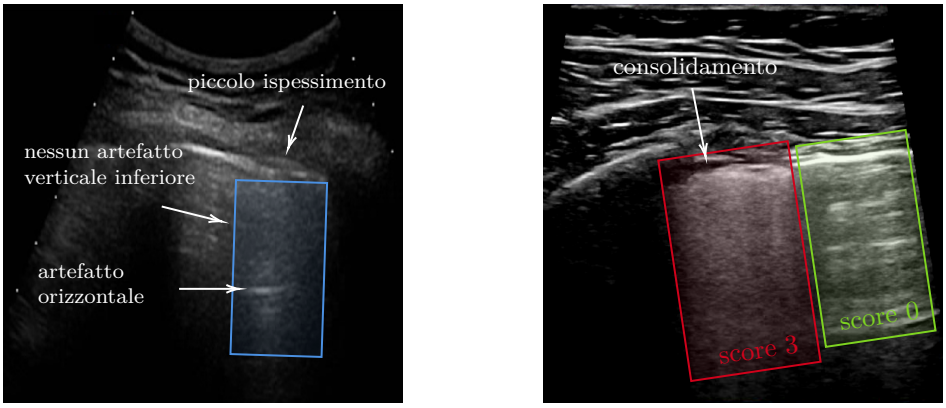
Passando alla ResNet18 nominale ma con loss ordinale QWK (Figura 4.5b), si notano i primi miglioramenti con una riduzione sensibile degli errori di sovradiagnosi, un leggero seppur presente miglioramento anche nella sottodiagnosi e un'inversione dei risultati di classificazione per la classe 3, tutti effetti positivi.

Per quanto riguarda gli approcci ordinali, OBD (Figura 4.5c) mostra risultati simili alla ResNet18 con QWK, migliorando notevolmente la classificazione della classe negativa (score 0) e aumentando significativamente le predizioni corrette degli score 2 e 3. Tuttavia, si osserva una diminuzione delle performance nella classe 1, fenomeno già preannunciato dalle curve ROC dato che questa classe è risultata essere la più complessa da gestire. Si precisa nuovamente che questa è una matrice risultante da un singolo campionamento delle 15 run effettuate con OBD, per tanto non è da considerarsi come risultato generalizzato per il modello (così come gli altri), dato che è applicato ad uno specifico fold e split. Quindi il fenomeno osservato potrebbe cambiare in altre condizioni di splitting.

Infine, il CLM (Figura 4.5d) migliora ulteriormente i risultati riducendo drasticamente gli errori sia di sovradiagnosi che di sottodiagnosi, portando questi ultimi al valore minimo dell'8%. CLM supera ogni risultato nella diagonale della ResNet18 con QWK e risolve i compromessi dell'OBD, a discapito di pochi punti percentuali persi nelle classi maggioritarie, confermandosi come il modello con le migliori prestazioni medie in classificazione ordinale.

Tuttavia, è importante sottolineare che una considerevole porzione della confusione dei modelli potrebbe derivare dall'interpretabilità intrinseca del dataset, causata dal rumore presente sia nei singoli frame sia nella modalità di *labeling* adottata. Si ipotizza che questa incertezza sia attribuibile alla soggettività delle annotazioni e alla presenza di frame ambigui di transizione, come evidenziato inizialmente nel lavoro di Roy et al. [7]. In particolare, negli esempi illustrati in Figura 4.6, l'immagine a sinistra (Figura 4.6a) mostra un frame classificato come Score 1, anche se le differenze rispetto allo Score 0 sono quasi impercettibili. Sebbene la linea pleurica sembri interrotta, potrebbe trattarsi semplicemente di un ispessimento. Mancano soprattutto gli artefatti verticali, mentre sono visibili le linee A, segno di una buona riflettività. D'altra parte, l'immagine a destra (Figura 4.6b) presenta un frame con caratteristiche valide per due score completamente diversi, ossia Score 0 e Score 3. Nella parte centrale dell'immagine, si osserva il consolidamento subpleurico e il

white lung, ma a destra è presente una porzione di frame perfettamente coerente con lo Score 0, caratterizzata da una linea pleurica regolare e artefatti orizzontali. Pertanto, tra gli errori riportati dalle matrici di confusione, emergono situazioni che non possono essere considerate errori reali del modello, ma che derivano dalle limitazioni intrinseche del dataset.



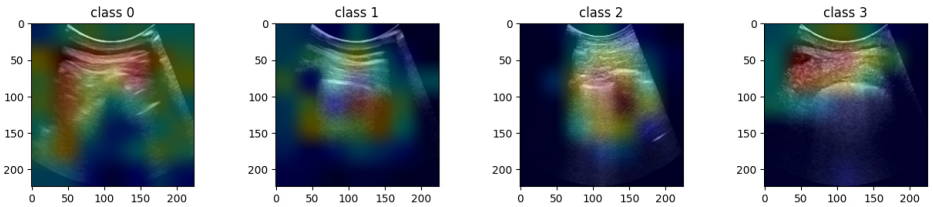
(a) Score 1 (b) Score 3

Figura 4.6: Esempi di complessità interpretativa del dataset che contribuisce alla confusione dei modelli neurali.

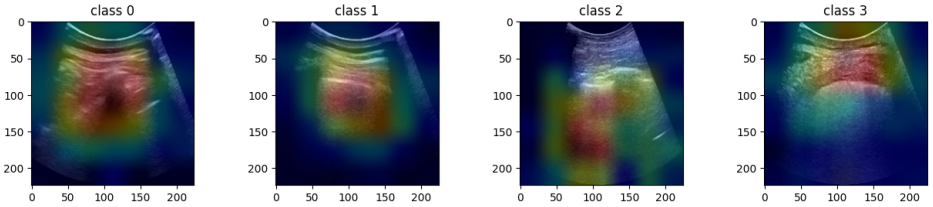
4.5 Analisi delle Mappe di Salienza con metodo GradCAM

Le mappe di salienza generate attraverso il metodo GradCAM *(Gradient-weighted Class Activation Mapping)* forniscono una visione approfondita della regione dell'immagine che ha influenzato maggiormente le decisioni del modello durante il processo di classificazione. La tecnica GradCAM si basa sul gradiente dell'output rispetto agli strati convoluzionali della rete. Innanzitutto, si calcolano i gradienti dell'output rispetto alla feature map dell'ultimo strato convoluzionale. Successivamente, questi gradienti vengono pesati in base all'importanza di ciascuna feature map per la classe di interesse. Infine, si aggregano le feature map ponderate per ottenere la mappa di salienza, evidenziando le regioni dell'input che hanno influenzato maggiormente la predizione della rete per una determinata classe. Come per le altre analisi condotte, i risultati saranno basati sui modelli pre-addestrati, garantendo una valutazione dettagliata delle performance su uno specifico fold e split (fold 1 e split 2). Inoltre, le mappe di calore sono state applicate su frame LUS non utilizzati nel processo di addestramento, in più provenendo dallo stesso fold e split è garantità l'uniformità tra i modelli utilizzando sempre lo stesso set di frame.

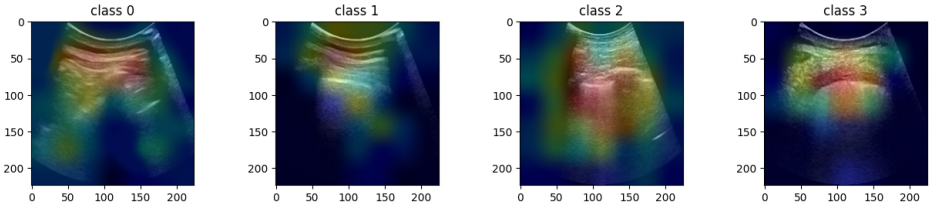
Le mappe sono disposte in una griglia (Figura 4.7), in cui le righe corrispondono ai diversi modelli considerati, mentre le colonne rappresentano le classi del problema (quattro in totale, associate agli score da 0 a 3).



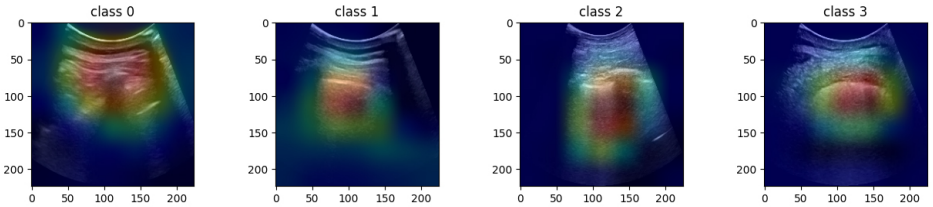
(a) ResNet18 + CCE



(b) ResNet18 + QWK



(c) OBD



(d) CLM

Figura 4.7: Mappe di attivazione estratte tramite il metodo GradCAM per ciascun modello e ogni classe del problema.

Analizzando le mappe di salienza del modello di riferimento (Figura 4.7a), si osserva che per la classe 0, il modello si concentra sulla zona al di sopra della linea pleurica e presenta una distribuzione troppo diffusa, estendendosi eccessivamente fino alla parte inferiore del frame (colori tendenti al verde e al giallo). La classe 1 è completamente trascurata, con poche e sparse attivazioni, confermando l'incapacità del modello di riconoscere questa classe, come già evidenziato dalle curve ROC e dalle matrici di confusione. La classe 2 è gestita in modo migliore, con attenzione all'interruzione della linea pleurica, anche se le regioni più attive (colori più tendenti al rosso) non corrispondono alle linee B essendo decentrate sul lato sinistro. Tuttavia, per la classe 3, il modello mostra una gestione inefficace, concentrandosi su una zona limitrofa alla linea pleurica e ignorando completamente il white lung.

I miglioramenti iniziali emergono con il modello nominale che sfrutta la loss ordinale QWK (Figura 4.7b). Nella classe 0, le attivazioni mostrano un'intensità comparabile rispetto a quelle ottenute con loss CCE, con una possibile enfatizzazione eccessiva nella zona scura circolare centrale. In più si osserva un notevole miglioramento nella classe 1, con attivazioni più chiare e concentrate che identificano in modo più preciso la linea pleurica. Per la classe 2, le attivazioni più intense si dirigono verso le linee b, rappresentando un progresso rispetto al modello precedente, anche se ulteriori miglioramenti sono ancora possibili dato che la linea pleurica non è ancora rilevata in modo integrato agli artefatti verticali. Un cambiamento significativo è evidente anche nella classe 3, con attivazioni più focalizzate sulla regione limitrofa della pleura, indicando una maggiore consapevolezza; tuttavia, la presenza di white lung continua a non essere rilevata.

Per quanto riguarda il modello OBD (Figura 4.7c), emerge una precisione nella classe negativa, concentrandosi principalmente nella regione della linea pleurica, anche se continuano a presentarsi attivazioni troppo sparse. Si osserva una difficoltà significativa nella classificazione della classe 1 con risultati pessimi, simile al modello di riferimento. Le attivazioni risultano scarse, prevalentemente concentrate nella parte superiore del frame, dove la natura convessa introduce errori di interpretazione dovuti ai bordi luminosi superiori che catturano erroneamente l'attenzione della rete. In aggiunta, il modello OBD mostra un peggioramento rispetto alla sua versione nominale con loss ordinale. Il panorama si modifica invece nelle classi più gravi, evidenziando una migliore performance nella classe 2. Infatti le attivazioni si concentrano nella discontinuità della linea pleurica ma anche integrando in modo efficace le linee b sottostanti, dimostrando un'interpretazione accurata di questa classe. Anche nello score 3 si osserva un notevole miglioramento, con la salienza posizionata esattamente sulla linea pleurica rotta e persino un fascio rosso che delinea il white lung, evidenziando una maggiore consapevolezza e il riconoscimento di dettagli patologici determinanti per una corretta classificazione.

Si conclude l'analisi delle mappe di salienza focalizzandosi sul CLM (Figura 4.7d). Per quanto riguarda la classe 1, il Cumulative Link Model presenta attivazioni accurate, con una leggera estensione della zona calda rispetto all'Ordinary Binary Decomposition (OBD), indicando una performance leggermente inferiore a quest'ultimo ma comunque debitamente centrata. Tuttavia, conferma le tendenze precedentemente osservate, poiché le mappe di attivazione per gli score da 1 a 3 sono dettagliate ed esplicative. Per la classe 1, il CLM non solo identifica accuratamente la posizione della linea pleurica ma concentra l'attenzione su di essa in modo più nitido, eliminando efficacemente le attivazioni nelle zone circostanti. Un comportamento simile è riscontrato anche per la classe 2, dove il modello pulisce le attivazioni nelle aree limitrofe, concentrandosi sulla linea pleurica e sopratutto sugli artefatti verticali. Infine, per lo score 3, il CLM supera gli altri modelli, evidenziando attivazioni prevalentemente intense, con una chiara identificazione della rottura della linea pleurica e un marcato rilievo dell'estensione del polmone bianco nell'ascissa.

4.6 Studio di Ablazione

Lo studio di ablazione rappresenta un approccio metodologico utile per comprendere il contributo individuale di specifiche componenti o parametri in un modello neurale. In questo contesto, l'ablazione permette di isolare gli effetti dell'architettura della testa di classificazione (nominale vs ordinale) e dell'utilizzo della funzione di loss (nominale vs ordinale). Saranno analizzati quattro scenari distinti, ciascuno allenato con la medesima architettura convoluzionale, evidenziando le differenze di performance derivanti dalle scelte di modellazione effettuate.

L'obiettivo principale di questo studio è identificare se l'adozione di un modello ordinale e l'utilizzo di una loss ordinale contribuiscono in modo significativo alle prestazioni della rete, isolando l'impatto della struttura della testa di classificazione.

I risultati saranno analizzati per valutare eventuali miglioramenti nelle metriche di valutazione, fornendo così indicazioni sulla validità e l'efficacia delle configurazioni proposte. A tale scopo, verranno valutati i risultati medi dell'F1-Score e dell'indice QWK riportati in Tabella 4.2 delle seguenti configurazioni di modelli, tutti allenati con la stessa backbone convolutiva pre-addestrata:

A. `ResNet18 + CCE`: modello nominale con loss nominale

   In questa configurazione, il modello utilizza un'architettura ResNet18 nominale con uscita Softmax tradizionale per la testa di classificazione e la funzione di loss nominale Categorical Cross-Entropy (CCE). Questo scenario serve come riferimento per valutare le performance di base della rete;

B. `ResNet18 + QWK`: modello nominale con loss ordinale

   In questa variante, il modello conserva l'architettura nominale per la testa di classificazione ma adotta una funzione di loss ordinale (QWK). L'obiettivo è comprendere se la semplice introduzione di una loss ordinale è sufficiente per migliorare le prestazioni;

C. `OBD + MSE`: modello ordinale con loss ordinale

   In questo caso, il modello implementa un'architettura ordinale per la testa di classificazione (OBD) e utilizza una loss function ordinale (MSE). Quindi si vuole valutare se e quanto un approccio ordinale puro influisce positivamente sulle prestazioni;

D. `CLM + QWK`: modello ordinale alternativo con loss ordinale

   Nella quarta configurazione, il modello adotta sia un'architettura ordinale alternativa (CLM) che una funzione di loss ordinale (QWK) diversa rispetto al terzo caso. L'obiettivo è determinare se varie architetture ordinali portano a risultati distinti.

La Tabella 4.3 riporta i risultati dello studio di ablazione condotto sui moduli di classificazione e loss impiegate nel modello ResNet18 pre-addestrato. Le colonne

indicate come "Modulo classificatore" rappresentano le configurazioni della testa di classificazione, con "Softmax", "OBD" e "CLM" che corrispondono ai tre diversi approcci. Le colonne "Loss" indicano le funzioni di loss utilizzate, tra cui CCE, QWK e MSE, con "Nom." e "Ord." rispettivamente come abbreviazioni di Nominale e Ordinale. Le colonne "Metriche" riportano i risultati medi delle metriche di valutazione, con F1 e QWK utilizzati come principali indicatori.

Modello	**Modulo classificatore**			**Loss**		**Metriche**	
	Softmax	**OBD**	**CLM**	**Nom.**	**Ord.**	**F1**	**QWK**
`ResNet18 + CCE`	✓			✓		$\overline{56.4\%}$	$\overline{65.3\%}$
`ResNet18 + QWK`	✓				✓	$\overline{56.2\%}$	$\overline{68.1\%}$
`OBD + MSE`		✓			✓	$\overline{58.2\%}$	$\overline{72.0\%}$
`CLM + QWK`			✓		✓	$\overline{\mathbf{59.4}\%}$	$\overline{\mathbf{73.1}\%}$

Tabella 4.3: Studio di ablazione sugli approcci (Softmax, OBD e CLM) e loss impiegate (CCE, QWK e MSE) dal modello ResNet18 pre-addestrato, per valutare l'impatto dei componenti dedicati a dati ordinali.

Utilizzando come classificatore l'uscita Softmax con loss nominale (ResNet18 + CCE), il modello ha raggiunto un F1-Score medio del 56.4% e un QWK medio del 65.3%, valori che verranno utilizzati come riferimento. L'introduzione della loss ordinale QWK ha portato ad un miglioramento della metrica omonima raggiungendo il 68.1%, lasciando invece quasi invariato l'F1. Quindi si può dedurre che un'operazione di semplice sostituzione della loss ordinale porta in questo contesto un incremento di quasi il 3%, senza sacrificare performance in termini puramente nominali.

Successivamente, passando all'approccio ordinale OBD con loss MSE (quindi ordinale), si è osservato un ulteriore miglioramento nelle performance. Il modello ha registrato un miglioramento dell'F1 fino al 58.2% e un QWK del 72.0%. Questo indica che l'utilizzo di un modello ordinale dedicato ha contribuito positivamente alla gestione della natura ordinale dei dati. Inoltre il distacco segnato, rispetto a quanto ResNet18 + QWK aveva fatto in confronto alla baseline evidenzia un miglioramento più marcato (+3.9% contro +2.8%).

Infine, l'approccio CLM, anch'esso basato sulla loss QWK, ha dimostrato di essere il più efficace nell'affrontare il problema della classificazione ordinale. Il modello ha ottenuto prestazioni superiori, con un F1 medio del 59.4% e un QWK medio del 73.1%. Quindi si registra un miglioramento rispetto al modello baseline del 3% per l'F1 e fino al 7.8% nella metrica QWK. Questi risultati indicano che l'implementazione di un approccio specifico per la classificazione ordinale, come CLM, può portare a notevoli miglioramenti rispetto agli approcci nominali.

4.7 Confronto con lo Stato dell'Arte

Il confronto con lo Stato dell'Arte *(State-Of-The-Art, SOTA)* nel contesto della classificazione dei danni polmonari da Covid-19 sulle LUS trova riferimento principalmente nei lavori di Roy et al. [7] e Frank et al. [8]. Tuttavia, va sottolineato fin da subito che un confronto diretto non è praticabile per due motivi. In primo luogo, i lavori di riferimento effettuano una valutazione dei modelli mediante holdout, fornendo un singolo valore di prestazione. Questo elaborato, invece, opta per una metodologia basata su cross-validation (CV), risultando in un valore medio piuttosto che uno puntuale. Pertanto, il confronto si limita alla comparazione tra il valore specifico dell'F1-Score riportato dal SOTA e quelli medi ottenuti in questo lavoro.

In secondo luogo, va notato che nella letteratura attuale mancano studi che affrontino la classificazione ordinale nel contesto del dataset `ICLUS`. Di conseguenza, i risultati presentati costituiscono il primo tentativo di applicare approcci di classificazione ordinale a questo specifico dominio. Tale assenza di riferimenti limita la possibilità di confrontare direttamente i risultati delle metriche ordinali con lavori precedenti.

Al fine di tentare di fornire un contesto comparativo, vengono riportate le performance del modello di riferimento, equivalente a quello impiegato nel lavoro di Roy et al. (ResNet18). Successivamente, tali risultati sono confrontati con quelli ottenuti attraverso l'implementazione di metodi ordinali allo scopo di valutare l'efficacia di tali approcci rispetto al modello di riferimento. Inoltre, tra i lavori di Roy et al. e Frank et al., si considera solamente il primo, dato che il secondo propone miglioramenti nel contesto della classificazione nominale, cercando di aumentare l'F1-Score. Poiché l'obiettivo della presente ricerca è focalizzato sulla classificazione ordinale, si riporta esclusivamente il risultato di Roy et al. relativo al modello baseline.

Modello	**F1-Score** (%)	**QWK** (%)
`CNN + CE` (Roy et al.)	61.6	-
`ResNet18 + SORD` (Roy et al.)	**62.2**	-
`ResNet18 + CCE` (∗)	$\overline{56.4}$	$\overline{65.3}$
`ResNet18 + QWK` (∗)	$\overline{56.2}$	$\overline{68.1}$
`OBD + MSE` (∗)	$\overline{58.2}$	$\overline{72.0}$
`CLM + QWK` (∗)	$\overline{59.4}$	$\overline{\mathbf{73.1}}$

Tabella 4.4: Confronto con lo stato dell'arte nella classificazione di frame LUS su dataset ICLUS introducendo i risultati di approcci ordinali. I modelli contrassegnati con l'asterisco (∗) sono quelli proposti in questa ricerca.

La Tabella 4.4 mostra come già dal SOTA si evince che il modello `ResNet18 +`

`SORD` (loss utilizzata in compiti di regressione ordinale) supera la `CNN + CE` in termini di F1-Score. Infatti la CNN base implementata da Roy et al. (non meglio specificata) con loss Cross-Entropy (CE) segna 61.2%. Dallo stesso lavoro ResNet18 con loss ordinale SORD raggiunge invece il 62.2%.

Tuttavia, il modello di riferimento implementato in questo lavoro `ResNet18 + CCE` imposta i valori di baseline per le metriche F1 e QWK, rispettivamente con una media del 56.4% (-4.8% rispetto al SOTA) e del 65.3%.

L'osservazione di valori medi di baseline inferiori rispetto al SOTA può essere attribuita al metodo di valutazione più robusto adottato in questa tesi. La Cross-Validation (CV) utilizzata consiste in diversi splitting del dataset, permettendo di valutare più configurazioni possibili e generare risultati più generalizzati. In contrasto, il SOTA ha utilizzato un holdout e misurato le performance su un singolo split, generando valori puntuali e dipendenti dalla particolare suddivisione del dataset. Questa differenza metodologica spiega la variazione nei valori medi, che possono essere sia inferiori che superiori rispetto al SOTA, come evidenziato anche dai boxplot. Tale approccio di valutazione più completo contribuisce a una comprensione più approfondita delle prestazioni dei modelli in diverse configurazioni del dataset.

Continuando il confronto, l'introduzione della loss ordinale nello stesso modello nominale lascia inalterate le performance in termini di F1 ma è sufficiente per portare un incremento di quasi il 3% in QWK. L'Ordinal Binary Decomposition (OBD) con loss basata su MSE è il primo modello a compiere un balzo in entrambe le direzioni, incrementando rispettivamente le metriche del $+1.8\%$ e del $+6,7\%$ rispetto alla baseline. La metodologia `CLM + QWK` implementata in questa ricerca emerge come il modello ordinale più performante, superando in entrambe le metriche ogni modello precedente, con un F1-Score medio del 59.4% e un QWK medio del 73.1%. In conclusione, l'impiego di metodi ordinali, come OBD e CLM, evidenzia il potenziale di miglioramento nella classificazione di frame LUS su dataset ICLUS, contribuendo significativamente al campo della diagnostica di frame LUS mediante reti neurali ordinali.

Capitolo 5

Conclusioni

Il presente elaborato ha affrontato la sfida della classificazione ordinale di frame di immagini ecografiche polmonari (LUS) nel contesto del dataset `ICLUS-DB`. L'obiettivo era raggiungere una classificazione accurata e interpretabile con un'attenzione particolare alla riduzione delle misclassicazioni più gravi, fondamentale per l'assistenza medica basata sull'intelligenza artificiale. La natura ordinale degli score con cui sono stati etichettati i frame ha aperto la strada all'adozione di metodologie ordinali per gestire la gerarchia degli stessi. Inoltre, l'eterogeneità del dataset ha enfatizzato l'importanza di strategie di valutazione robuste in grado di riuscire ad estrarre risultati *split-agnostic*, ovvero generalizzati e non dipendenti dalla divisione dei pazienti nei vari set.

L'implementazione di reti neurali convoluzionali (CNN), nello specifico del modello ResNet18, ha fornito una solida base di partenza in comune allo stato dell'arte. L'adozione di funzioni di loss ordinali e sopratutto di architetture ordinali come l'Ordinal Binary Decomposition (OBD) e il Cumulative Link Model (CLM) hanno permesso di catturare le relazioni ordinali in modo più sensibile. Il Transfer Learning, attraverso l'utilizzo della backbone convoluzionale ResNet18 pre-addestrata su ImageNet, ha dimostrato di apportare miglioramenti significativi alle prestazioni dei modelli.

Il processo di sviluppo non è stato privo di sfide. La gestione del *class imbalance* ha richiesto l'implementazione di strategie di pesatura delle classi e l'adozione di metriche sensibili a tale disequilibrio. L'introduzione di Cross-Validation con K-Fold e split stratificati interni è risultata una novità non affrontata in letteratura in questo dominio. Le fasi di Grid Search interne a ciascun fold hanno portato all'esplorazione dello spazio degli iperparametri su diverse configurazioni del dataset, offrendo come risultato i parametri più adeguati e generalizzati sul dataset per massimizzare i risultati di classificazione dei modelli neurali. I contributi principali di questa tesi includono l'introduzione di metodologie ordinali specifiche nel contesto LUS e l'impiego di una tecnica di Cross-Validation progettata ad-hoc per le sfide proposte dall'`ICLUS-DB`.

Il confronto sistematico tra approcci nominali e ordinali ha evidenziato le potenzialità di quest'ultimi nel gestire la natura gerarchica degli score. I risultati ottenuti sono stati valutati attraverso la misurazione di diverse metriche. Per quanto riguarda il contesto nominale si è misurato il classico CCR e l'F1-Score. Invece le metriche

ordinali implementate comprendono Accuracy 1-Off e 2-Off, il Quadratic Weighted Kappa (QWK) e il rank di Spearman (r_S). Per quanto riguarda le misure dell'errore si è optato per il Mean Absolute Error (MAE) e il Root Mean Squared Error (RMSE). In fine, per monitorare il class imbalance si è sfruttata la sensiblità minima (MS), monitorando le performance sulla classe minoritaria. L'analisi dettagliata delle curve ROC ha approfondito la comprensione delle performance in diverse condizioni di scoring. Le matrici di confusione sono risultate utili per confermare come i metodi ordinali siano in grado di ridurre sensibilmente la sovradiagnosi e sopratutto la sotto-diagnosi. Le analisi delle mappe di salienza con il metodo GradCAM hanno offerto un'ulteriore chiave di interpretazione del processo decisionale della rete. Queste sottolineano la capacità di questi modelli ordinali di individuare in modo più meticoloso rispetto alla baseline le regioni con contenuto patologico dell'immagine, contribuendo a una maggiore interpretabilità. L'esecuzione di uno studio di ablazione ha permesso di isolare l'impatto delle componenti ordinali sui risultati. Il confronto con lo stato dell'arte, seppur non direttamente possibile, ha confermato la validità degli approcci ordinali proposti, fornendo un nuovo punto di riferimento per le performance di modelli ordinali nel contesto del dataset `ICLUS-DB`.

Inoltre, da notare che, a differenza di quanto riscontrato in un caso nella letteratura dei metodi ordinali [30], l'impiego di queste tecniche nel dominio LUS non solo non compromette le performance in F1-Score, ma addirittura le migliora.

Un punto chiave emerso da questa ricerca è quindi l'efficacia sostanziale degli approcci ordinali, evidenziata dai significativi miglioramenti nelle metriche, tra cui Accuracy 1-Off, QWK e Spearman. Questi risultati si traducono in una percentuale notevolmente aumentata di frame, e quindi pazienti, classificati in modo più vicino alla realtà. Questo è di particolare rilevanza in ambito medico, in quanto ridurre gli errori a distanza elevata nella area inferiore della diagonale della matrice di confusione, che corrispondono agli errori sottovalutazione, è fondamentale. La riduzione di tali errori di sottodiagnosi rappresenta un progresso significativo, indicando che il modello è in grado di assegnare gli score con attenzione per evitare di sottovalutare la gravità della patologia. Questo ha impatti pratici notevoli, consentendo il riconoscimento tempestivo delle classi più gravi e il conseguente avvio di interventi preventivi. Tale capacità di migliorare la precisione nelle diagnosi è un passo avanti rilevante nell'applicazione clinica dei modelli di apprendimento automatico nel contesto della LUS.

In sintesi, questa tesi fornisce un contributo all'applicazione di metodi ordinali nella classificazione di frame LUS del dataset ICLUS, dimostrando la loro utilità nell'incrementare la precisione diagnostica. Le implicazioni pratiche di questa miglioria sono determinanti per garantire un'assistenza medica più efficace e tempestiva, aprendo la strada a ulteriori sviluppi nell'integrazione di approcci avanzati nelle procedure diagnostiche cliniche.

5.1 Limitazioni dello studio

Va sottolineato che questo studio presenta alcune limitazioni intrinseche. In primo luogo, nonostante gli sforzi mirati a garantire un'analisi completa e accurata, l'assenza di un confronto diretto con lavori precedenti basati su classificazione ordinale su dataset ICLUS può limitare la bontà delle conclusioni. I risultati ottenuti, seppur promettenti, necessitano di ulteriori convalide ed eventualmente confronti con altre metodologie presenti in letteratura. Va inoltre sottolineato che la mancanza di accesso al criterio di splitting del dataset utilizzato nei lavori precedenti, in particolare nello stato dell'arte, impedisce un confronto diretto dei risultati ottenuti con l'architettura baseline. Questo aspetto rappresenta una limitazione intrinseca nell'analisi comparativa dei risultati ottenuti in questa ricerca con quelli presenti in letteratura. Inoltre, la scelta di adottare la ResNet18 come unica backbone di tutti gli approcci per allinearsi con lo stato dell'arte potrebbe rappresentare un vincolo nel valutare appieno il potenziale di altre architetture di reti neurali. L'espansione della ricerca includendo diverse architetture potrebbe fornire una panoramica più completa delle possibilità di modellazione per la classificazione ordinale di frame LUS. Un'altra limitazione riguarda la varianza nei risultati dovuta alle diverse configurazioni degli splitting durante la cross-validation. Pur avendo implementato un approccio controllato e robusto, i risultati mostrano una varianza consistente che sembrerebbe essere associata alla suddivisione dei pazienti nei vari set. Questo aspetto richiede un'attenzione ulteriore nell'interpretazione dei risultati e suggerisce la necessità di valutare approcci di mitigazione dell'impatto di questa variabilità nei futuri studi.

5.2 Sviluppi futuri

Un'area di possibile interesse futuro riguarda l'indagine delle metodologie di splitting dei lavori precedenti. Questo potrebbe permettere un confronto più approfondito con i risultati attuali e contribuire ad una comprensione più chiara delle variabili coinvolte nel processo di valutazione dei modelli, cercando di ottenere dettagli sui criteri utilizzati nei lavori dello stato dell'arte per una valutazione più approfondita. Un ulteriore percorso di sviluppo potrebbe essere l'integrazione di informazioni di dominio nel processo di classificazione, seguendo l'esempio della ricerca di Frank et al. dello stato dell'arte. Incorporare queste informazioni potrebbe migliorare ulteriormente le performance ordinali, concentrando l'attenzione della rete sulle caratteristiche patologiche nei frame LUS. Per concludere, considerando gli avanzamenti nell'ambito delle reti neurali, un'ulteriore direzione di sviluppo potrebbe riguardare l'esplorazione di architetture basate su Transformer, eventualmente implementando approcci Multi-Task. L'integrazione di queste tecnologie potrebbe consentire di sfruttare al massimo metodologie avanzate, portando a un ulteriore progresso nello stato dell'arte della classificazione di frame LUS.

Bibliografia

[1] C. Pollard, M. Morran, and A. Kalinoski. The covid-19 pandemic: a global health crisis. *Physiological genomics*, 52, 09 2020.

[2] S. Woloshin, N. Patel, and A. S. Kesselheim. False negative tests for sars-cov-2 infection — challenges and implications. *New England Journal of Medicine*, 383(6):e38, 2020. PMID: 32502334.

[3] F. Mojoli, B. Bouhemad, S. Mongodi, and D. Lichtenstein. Lung ultrasound for critically ill patients. *American Journal of Respiratory and Critical Care Medicine*, 199, 2019.

[4] Gino Soldati, Marcello Demi, Andrea Smargiassi, Riccardo Inchingolo, and Libertario Demi. The role of ultrasound lung artifacts in the diagnosis of respiratory diseases. *Expert Review of Respiratory Medicine*, 13, 2019.

[5] G. Soldati, A. Smargiassi, R. Inchingolo, D. Buonsenso, T. Perrone, D. F. Briganti, S. Perlini, E. Torri, A. Mariani, E. E. Mossolani, F. Tursi, F. Mento, and L. Demi. Proposal for international standardization of the use of lung ultrasound for covid-19 patients; a simple, quantitative, reproducible method. *Journal of ultrasound in medicine : official journal of the American Institute of Ultrasound in Medicine*, 2020.

[6] L. Demi. Lung ultrasound: The future ahead and the lessons learned from covid-19. *The Journal of the Acoustical Society of America*, 148, 2020.

[7] S. Roy, W. Menapace, S. Oei, B. Luijten, E. Fini, C. Saltori, I. Huijben, N. Chennakeshava, F. Mento, A. Sentelli, E. Peschiera, R. Trevisan, G. Maschietto, E. Torri, R. Inchingolo, A. Smargiassi, G. Soldati, P. Rota, A. Passerini, R. J.G. Van Sloun, E. Ricci, and L. Demi. Deep learning for classification and localization of covid-19 markers in point-of-care lung ultrasound. *IEEE Transactions on Medical Imaging*, 39, 2020.

[8] O. Frank, N. Schipper, M. Vaturi, G. Soldati, A. Smargiassi, R. Inchingolo, E. Torri, T. Perrone, F. Mento, L. Demi, M. Galun, Y. C. Eldar, and S. Bagon. Integrating domain knowledge into deep networks for lung ultrasound with applications to covid-19. *IEEE Transactions on Medical Imaging*, 41(3):571–581, 2022.

[9] R. Raheja, M. Brahmavar, D. Joshi, and D. Raman. Application of lung ultrasound in critical care setting: A review. *Cureus*, 2019.

[10] F. Giannelli, D. Cozzi, E. Cavigli, I. Campolmi, F. Rinaldi, S. Giachè, P. Giorgio Rogasi, V. Miele, and M. Bartolucci. Lung ultrasound (lus) in pulmonary tuberculosis: correlation with chest ct and x-ray findings. *Journal of Ultrasound*, 25, 2022.

[11] F. Fichera, M. Nicotra, and I. Paolini. Pocus del polmone. *Rivista Società Italiana di Medicina Genarale e delle Cure Primarie*, 27, 2020.

[12] A. W. Salehi, S. Khan, G. Gupta, B. Ibrahimm A., A. Almjally, H. Alsolai, T. Siddiqui, and A. Mellit. A study of cnn and transfer learning in medical imaging: Advantages, challenges, future scope. *Sustainability (Switzerland)*, 15, 2023.

[13] P. A. Gutiérrez, M. Pérez-Ortiz, J. Sánchez-Monedero, F. Fernández-Navarro, and C. Hervás-Martínez. Ordinal regression methods: Survey and experimental study. *IEEE Transactions on Knowledge and Data Engineering*, 28(1):127–146, 2016.

[14] M. Axelsson, H. Backman, B. I. Nwaru, C. Stridsman, L. Vanfleteren, L. Hedman, P. Piirilä, J. Jalasto, A. Langhammer, H. Kankaanranta, M. Rådinger, L. Ekerljung, E. Rönmark, and A. Lindberg. Underdiagnosis and misclassification of copd in sweden – a nordic epilung study. *Respiratory Medicine*, 217, 2023.

[15] P. Rajpurkar et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *CoRR*, abs/1711.05225, 2017.

[16] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9351, 2015.

[17] R. J.G. Van Sloun, R. Cohen, and Y. C. Eldar. Deep learning in ultrasound imaging. *Proceedings of the IEEE*, 108, 2020.

[18] H. Che, L. G. Brown, D. J. Foran, J. L. Nosher, and I. Hacihaliloglu. Liver disease classification from ultrasound using multi-scale cnn. *International Journal of Computer Assisted Radiology and Surgery*, 16, 2021.

[19] N. H. Alkurdy, D. K. Aljobouri, and Z. K. Wadi. Ultrasound renal stone diagnosis based on convolutional neural network and vgg16 features. *International Journal of Electrical and Computer Engineering*, 13, 2023.

[20] J. Ramesh and R. Manavalan. Prostate ultrasound image classification using cnn-bilstm. *Indian Journal of Computer Science and Engineering*, 12, 2021.

[21] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128, 2020.

[22] O. M. Doyle, E. Westman, A. F. Marquand, P. Mecocci, B. Vellas, M. Tsolaki, I. Kłoszewska, H. Soininen, S. Lovestone, S. C.R. Williams, and A. Simmons. Predicting progression of alzheimer's disease using ordinal regression. *PLoS ONE*, 9, 2014.

[23] M. Dorado-Moreno, M. Pérez-Ortiz, P. A. Gutiérrez, R. Ciria, J. Briceño, and C. Hervás-Martínez. Dynamically weighted evolutionary ordinal neural network for solving an imbalanced liver transplantation problem. *Artificial Intelligence in Medicine*, 77, 2017.

[24] J. Sánchez-Monedero, M. Pérez-Ortiz, A. Sáez, P. A. Gutiérrez, and C. Hervás-Martínez. Partial order label decomposition approaches for melanoma diagnosis. *Applied Soft Computing Journal*, 64, 2018.

[25] P. McCullagh. Regression models for ordinal data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 42, 1980.

[26] Alan Agresti. *Analysis of ordinal categorical data*, volume 656. John Wiley & Sons, 2010.

[27] V. M. Vargas, P. A. Gutiérrez, and C. Hervás-Martínez. Cumulative link models for deep ordinal classification. *Neurocomputing*, 401, 2020.

[28] H. Wu, H. Lu, and S. Ma. A practical svm-based algorithm for ordinal regression in image retrieval. *Proceedings of the ACM International Multimedia Conference and Exhibition*, 2003.

[29] F. Cheng, Z. Wang, and G. Pollastri. A neural network approach to ordinal regression. *Proceedings of the International Joint Conference on Neural Networks*, 2008.

[30] J. Barbero-Gómez, P. A. Gutiérrez, V. M. Vargas, J. A. Vallejo-Casas, and C. Hervás-Martínez. An ordinal cnn approach for the assessment of neurological damage in parkinson's disease patients. *Expert Systems with Applications*, 182, 2021.

[31] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-December, 2016.

[32] J. de la Torre, D. Puig, and A. Valls. Weighted kappa loss function for multi-class classification of ordinal data in deep learning. *Pattern Recognition Letters*, 105, 2018.

[33] A. Ben-David. Comparison of classification accuracy using cohen's weighted kappa. *Expert Systems with Applications*, 34, 2008.

[34] E. Frank and M. Hall. A simple approach to ordinal classification. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2167, 2001.

[35] J. Barbero-Gómez, P. A. Gutiérrez, and C. Hervás-Martínez. Error-correcting output codes in the framework of deep ordinal classification. *Neural Processing Letters*, 55, 2023.