



UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica  
Strumenti Formali per la Bioinformatica

RELAZIONE DI PROGETTO

**Un approccio deep learning per la classificazione delle varie tipologie di forme tumorali tramite uso delle espressioni genetiche, individuando potenziali biomarker**

DOCENTI:

Prof.ssa Clelia De Felice  
Prof.ssa Rosalba Zizza  
Prof. Rocco Zaccagnino

STUDENTI:

Nicola Pagliara 0522501413  
[n.pagliara1@studenti.unisa.it](mailto:n.pagliara1@studenti.unisa.it)  
Gaetano Antonucci 0522500941  
[g.antonucci2@studenti.unisa.it](mailto:g.antonucci2@studenti.unisa.it)

## Abstract

L'analisi differenziale è la parte più significativa dell'analisi delle RNA-Seq. I metodi convenzionali di solito mettono in corrispondenza i sample tumorali con i sample normali provenienti dalla stessa tipologia di tumore. Tali metodi, però, potrebbero fallire nel differenziare i diversi tipi di tumore siccome non possono sfruttare la conoscenza proveniente da altre tipologie di tumore. Il Pan-Cancer Atlas fornisce informazioni molto ampie sulle 33 classi prevalenti di tumori che possono essere utilizzate come conoscenza base per generare biomarker specifici a seconda della classe tumorale. In questo lavoro abbiamo dapprima replicato il lavoro svolto nel paper [1] inglobando i dati ad alta dimensionalità delle RNA-Seq in immagini 2D al fine di usare una rete neurale convoluzionale (CNN) per eseguire la classificazione delle 33 classi tumorali. L'accuracy finale ottenuta è stata del 94.74%. Abbiamo poi provato, come variante a quanto svolto in [1], a modificare i parametri di addestramento della CNN al fine di migliorarne le performance ma abbiamo ottenuto un'accuracy complessiva del 94.83% che è inferiore a quanto ottenuto da [1]. Tuttavia abbiamo ottenuto dei miglioramenti per singola coorte tumorale, come ad esempio ESCA, LGG e LUAD. In seguito, seguendo quanto fatto, abbiamo sfruttato anche noi l'idea di Guided Grad-CAM [2] e abbiamo generato, per ogni classe, le heatmap significative per tutti i geni. Tramite Pathway Enrichment Analysis, sui geni che hanno mostrato un'alta intesità nelle heatmap, è stato possibile validare che tali geni sono correlati a specifiche path tumorali dimostrando che è possibile utilizzare tale lavoro per la ricerca di potenziali biomarker.

# Indice

<b>Elenco delle figure</b>	<b>iii</b>
<b>Elenco delle tabelle</b>	<b>iv</b>
<b>1 Introduzione</b>	<b>1</b>
1.0.1 Pan-Cancer Atlas . . . . .	2
1.1 Next Generation Sequencing e RNA-Seq . . . . .	2
1.2 RNA-Seq vs Microarray DNA . . . . .	3
1.3 Funzionamento di RNA-Seq . . . . .	3
1.4 RNA-Seq per la ricerca sul cancro . . . . .	3
<b>2 Lavori correlati</b>	<b>4</b>
2.1 Metodi di Machine Learning e Deep Learning per la classificazione . . . . .	4
2.2 Metodi di Visualizzazione delle Deep Neural Network . . . . .	4
<b>3 Metodologia Applicata</b>	<b>6</b>
3.1 Workflow . . . . .	6
3.2 Dati utilizzati . . . . .	6
3.2.1 RSEM: RNA-Seq Expectation Maximization . . . . .	8
3.3 Fase di Preprocessing . . . . .	8
3.4 Classificazione delle immagini tramite Neural Network . . . . .	9
3.4.1 Convolutional Neural Network . . . . .	10
3.4.2 Convolutional Neural Network Layers . . . . .	11
3.4.3 Funzioni d'attivazione . . . . .	15
3.4.4 Funzioni di perdita . . . . .	17
3.4.5 Ottimizzatori . . . . .	17
3.4.6 Hyperparameter tuning . . . . .	19
3.5 Generazione delle Heatmap . . . . .	21
3.5.1 Gradient Classification Activation Map . . . . .	21
3.5.2 Guided BackPropagation . . . . .	22
3.5.3 Guided GradCam . . . . .	22
3.6 Validazione . . . . .	23
3.6.1 Enrichment Pathway Analysis . . . . .	23
<b>4 Risultati Sperimentali</b>	<b>26</b>
4.1 Valutazione Classificazione . . . . .	26
4.1.1 Caso Binario . . . . .	26
4.1.2 Caso Ternario . . . . .	26
4.1.3 Caso Generale . . . . .	28
4.2 Heat-Map Generate . . . . .	28
4.2.1 Caso Binario . . . . .	29
4.2.2 Caso Ternario . . . . .	30
4.2.3 Caso Generale . . . . .	30
4.3 Validazione dei percorsi biologici dei top genes . . . . .	30
4.3.1 Caso Binario . . . . .	32
4.3.2 Caso Ternario . . . . .	35
4.3.3 Caso Generale . . . . .	36

<b>5 VarNet: Variant Convolutional Neural Network</b>	<b>49</b>
5.1 VarNet vs Net: Less is more? . . . . .	49
5.1.1 Assunzioni e Obiettivi VarNet . . . . .	49
5.1.2 Schema della rete: VarNet vs Net . . . . .	50
5.1.3 Learning blocks: VarNet vs Net . . . . .	50
5.1.4 Risultati training: VarNet vs Net . . . . .	51
5.2 Valutazione Classificazione . . . . .	51
5.3 Heatmap Generation . . . . .	52
5.4 Validazione biologica . . . . .	53
<b>6 Discussione</b>	<b>67</b>
<b>7 Conclusioni e Sviluppi Futuri</b>	<b>68</b>
<b>8 Ringraziamenti</b>	<b>69</b>
<b>Bibliografia</b>	<b>69</b>
<b>A Manutenzione ed Evoluzione DL-tumor based approach</b>	<b>75</b>
A.1 Pipeline Progetto . . . . .	75
A.2 Moduli e Modifiche Apportate al Codice Esistente . . . . .	76
A.2.1 Raw Data . . . . .	76
A.2.2 Preprocessing . . . . .	77
A.2.3 Training e Testing . . . . .	78
A.2.4 Performance evaluation . . . . .	79
A.2.5 Heatmaps generation . . . . .	80
A.2.6 Biological evaluation . . . . .	82
A.2.7 Moduli di Supporto . . . . .	83
A.3 Setup dell'esperimento . . . . .	83

## Elenco delle figure

1	Il workflow del nostro metodo (ripreso da [1]) . . . . .	6
2	Un esempio di immagini 2D incorporate dalla classe 1 alla classe 33. . . . .	9
3	L'architettura della rete neurale convoluzionale . . . . .	9
4	Confronto archittture CNN e FC [3] . . . . .	10
5	Esempio di movimento di un filtro $3 \times 3$ con stride 1. [4] . . . . .	12
6	Risultato dell'applicazione dello stride. [4] . . . . .	12
7	Effetto dello zero padding sull'immagine di input. [4] . . . . .	13
8	Funzionamento della convoluzione [5] . . . . .	13
9	Layer Convoluzionale [5] . . . . .	14
10	Applicazione Max Pooling [5] . . . . .	14
11	Diagramma generale sulle funzioni d'attivazione [3] . . . . .	15
12	Esempi di funzioni d'attivazione: (a) Sigmoid, (b) Tanh, (c) ReLU . . . . .	16
13	La funzione Leaky ReLU . . . . .	17
14	Overview di Guided Grad CAM [2] . . . . .	23
15	La matrice di confusione del caso binario . . . . .	27
16	La matrice di confusione del caso ternario . . . . .	27
17	La matrice di confusione del caso generale . . . . .	30
18	Heatmaps del caso binario . . . . .	30
19	Heatmaps del caso ternario . . . . .	31
20	Alcuni esempi di heatmap. Ogni colonna rappresenta il risultato di una fold. Nella prima riga ci sono le heatmap del tipo di tumore BLCA, nella seconda di LGG e nella terza di PRAD. . . . .	31
21	I cambi di intensità nelle heatmap per ogni classe. Si può notare come alcune classi condividano lo stesso pattern nei cambi di intensità. . . . .	32
22	Composizione della Rete VarNet . . . . .	50
23	Confusion Matrix di VarNet . . . . .	52
24	Alcuni esempi di heatmap legate a VarNet. Ogni colonna rappresenta il risultato di una fold. Nella prima riga ci sono le heatmap del tipo di tumore BLCA, nella seconda di LGG e nella terza di PRAD. . . . .	52
25	I cambi di intensità nelle heatmap per ogni classe. Si può notare come alcune classi condividano lo stesso pattern nei cambi di intensità. . . . .	54
26	Pipeline del progetto . . . . .	75
27	Modulo Raw Data . . . . .	76
28	Modulo di Preprocessing . . . . .	77
29	Modulo di Training e Testing . . . . .	78
30	Modulo di Valutazione delle Performance . . . . .	79
31	Modulo di Generazione delle Heatmap . . . . .	80
32	Modulo di Validazione Biologica . . . . .	82
33	Fold di output. . . . .	84

## Elenco delle tabelle

1	Numero campioni di RNA-Sequence tumorali . . . . .	7
2	Iperparametri più comuni . . . . .	20
3	Performance del caso binario . . . . .	26
4	Risultati del caso binario per coorte tumorale . . . . .	26
5	Performance del caso ternario . . . . .	28
6	Risultati del caso ternario per coorte tumorale . . . . .	28
7	Performance del metodo utilizzato . . . . .	28
8	Accuracy per coorte tumorale (calcolo effettuato su GPU e su dataset con oversampling). . . . .	29
9	Risultati della Pathways Analysis sui primi 400 geni per le coorti DLBC e UCS ( $P < 10^{-3}$ ) . . . . .	33
10	Risultati della Pathways Analysis sui primi 400 geni per le coorti BLCA, CESC e LGG ( $P < 10^{-3}$ ) . . . . .	35
11	Risultati della Pathways Analysis sui primi 400 geni per ogni tipo di tumore ( $P < 10^{-3}$ ) . . . . .	37
12	Tabella del training GPU si dataset oversampled di VarNet . . . . .	49
13	Tabella del training GPU si dataset oversampled di Net . . . . .	49
14	Performance del metodo utilizzato con VarNet . . . . .	52
15	Accuracy per coorte tumorale (calcolo effettuato su CPU e su dataset senza oversampling). . . . .	53
16	Risultati della Pathways Analysis sui primi 400 geni per ogni tipo di tumore ( $P < 10^{-3}$ ) . . . . .	54
17	Specifiche Hardware & Software della Macchina Utilizzata . . . . .	83

# 1 Introduzione

Grazie al miglioramento delle tecniche di sequenziamento con l'avvento dei Next Generation Sequencing methods, si è avuto un progressivo miglioramento in accuratezza ed efficienza anche per quanto riguarda l'analisi del genoma umano. Un problema sempre presente anche ai giorni nostri è quello di comprendere in maniera completa e approfondita le cause dei vari tumori. A tal fine, The Cancer Genome Atlas, si è occupato del sequenziamento e della gestione di un grande volume di tessuti tumorali e, inoltre, ha analizzato oltre 11.000 tumori appartenenti alle 33 forme più diffuse di cancro. A partire da tali analisi è stata incentivata la creazione del Pan-Cancer Atlas, il quale, come riportato nella loro home page "fornisce una comprensione unica, completa, approfondita e interconnessa di come, dove e perché nascono i tumori nell'uomo". Il Pan-Cancer Atlas è di fatto una risorsa essenziale per lo sviluppo di nuovi trattamenti nella ricerca della medicina personalizzata, in quanto, per ogni sample tumorale, è possibile accedere ai suoi *RNA-Seq expression data*. Tali dati sono un vantaggio quando si cerca di identificare i potenziali biomarker per ogni tipo di tumore.

Finora, per tentare di scoprire i potenziali biomarker, molte analisi cercavano di trovare i geni che si sono espressi differentemente. Tali analisi, però, non tengono conto dell'espressione dei geni in altri tipi di tumore e solo legate al tipo di tumore e/o ai dati specifici che si stanno analizzando correntemente. Di fatti, molti modelli vanno a simulare l'espressione dei geni ma falliscono miseramente quando si usa il modello su altri dati o su altri tipi di tumore. Da tali fallimenti è dunque nata l'esigenza di progettare un metodo che possa includere nelle analisi la conoscenza proveniente da più classi tumorali.

Da un altro punto di vista, la classificazione dei tumori usando i dati genomici può contribuire ad una diagnosi più veloce e più accurata ma, di fatti, raramente viene usata in quanto è difficile lavorare con dataset genomici ad alta dimensionalità. Per rendere più chiaro questo punto, basti pensare che nel Pan-Cancer Atlas gli *mRNA-Seq gene expression data* contengono informazioni a partire da oltre 20 mila geni, molti dei quali non sono coinvolti nello sviluppo dei tumori. Tali geni sono dunque weak feature quando si lavora con tecniche di Machine Learning. Usare metodi classici di ML come KNN non è fattibile in quanto essi fallirebbero a causa della cosiddetta "curse of dimension". È da notare che, se anche la classificazione dei tumori è ancora agli inizi, la classificazione e il riconoscimento di immagini è invece molto sviluppata e in tale ambito è usato ampiamente il deep learning. Molte architetture come ResNet [6] e Inception [7] hanno mostrato performance eccellenti nella classificazione multi-classe di immagini. Inoltre, per comprendere come lavorano le deep neural network, nel campo della computer vision sono stati sviluppati metodi come la deep Taylor Decomposition, la layer-wise decomposition e Grad-CAM [2]. Tali metodi generano heatmap delle immagini in input al fine di indicare quanto ogni pixel ha contribuito alla classificazione. Dal momento che i *gene expression data* contengono oltre 10 mila geni è promettente utilizzare le deep neural network per ottenere una buona accuracy. Allo stesso tempo, se si suppone che l'importanza di un gene sia paragonabile a quanto si è espresso (e dunque a quanto ha contribuito alla classificazione), si possono prendere in prestito i metodi di interpretazione delle deep neural network per scoprire i top gene di ogni tumore. A tale scopo, ad ogni gene sarà associato un confidence score, e i geni con uno score alto saranno considerati come top gene e dunque potenziali biomarker dal momento che la loro esistenza ha influito in maniera considerevole nella classificazione.

A partire dal lavoro svolto da Lyu e Haque [1], dapprima abbiamo provveduto a creare il dataset su cui lavorare a partire dai *raw data*. Poi, abbiamo replicato il loro lavoro andando a filtrare i geni nei sample che presentavano una varianza piccola. Successivamente, abbiamo incorporato gli expression data ad alta intensità (10381x1) in immagini 2D (102x102) per adattarli ai layer convoluzionali. In seguito, abbiamo utilizzato la rete convoluzionale di Lyu e Haque unitamente alla 10-fold cross validation per testare le performance. Con la rete addestrata, abbiamo seguito al pari degli autori, l'idea della Gradient-weighted Class Activation Mapping (Guided Grad-CAM [2]) e sono state generate le heat-map per tutte le classi mostrando i pixel più significativi (geni). Tramite Pathway Enrichment Analysis, abbiamo poi validato, come in [1], che la scelta dei top gene effettuata è biologicamente

significativa per i tumori corrispondenti e ciò dimostra che il lavoro di partenza è valido. Per agevolarci nel compito abbiamo dapprima effettuato dei test su un numero ristretto di classi tumorali. Dapprima abbiamo lavorato su un caso binario (classi DLBC e UCS), poi su un caso ternario (BLCA, CESC e LGG) e infine sul caso generale (tutte e 33 le classi tumorali).

### 1.0.1 Pan-Cancer Atlas

The Cancer Genome Atlas (TCGA), un programma di riferimento per la genomica del cancro, ha caratterizzato molecolarmente oltre 20.000 campioni primari di cancro e campioni normali appaiati che coprono 33 tipi di cancro. Questo sforzo congiunto tra il National Cancer Institute (NCI) e il National Human Genome Research Institute è iniziato nel 2006 riunendo ricercatori di diverse discipline e molteplici istituzioni. The Cancer Genome Atlas (TCGA) ha contribuito a stabilire l'importanza della genomica del cancro<sup>1</sup>, ha trasformato la nostra comprensione del cancro e ha persino iniziato a cambiare il modo in cui la malattia viene trattata clinicamente. L'impatto va ancora oltre poi, raggiungendo le tecnologie sanitarie e scientifiche, la biologia computazionale e altri campi di ricerca. Dopo 12 anni, con i contributi di oltre 11.000 pazienti e l'incredibile impegno di migliaia di ricercatori, il TCGA ha prodotto una serie di dati di incommensurabile valore. Questi dati<sup>2</sup> rimangono a disposizione del pubblico come riferimento affidabile che potrà essere sfruttato per molti anni. Da questi sforzi nasce il Pan-Cancer Atlas, una raccolta di analisi trasversali sul cancro che ne approfondiscono temi generali, tra cui: i modelli di origine cellulare, i processi oncogenici e le signaling pathway. Grazie all'analisi di oltre 11.000 tumori provenienti da 33 delle forme tumorali più diffuse, il Pan-Cancer Atlas fornisce una comprensione unica, completa, approfondita e interconnessa di come, dove e perché nascono i tumori nell'uomo. Come punto di riferimento unico e unificato, il Pan-Cancer Atlas è una risorsa essenziale per lo sviluppo di nuovi trattamenti nella ricerca della medicina di precisione, il progetto è stato reso pubblico con tutti i dati dopo il suo completamento nel 2018.

## 1.1 Next Generation Sequencing e RNA-Seq

Come affermato in [8] con le tecniche Next Generation Sequencing e quelle NGS-based RNA sequencing (RNA-Seq) si è avuto un progresso tecnologico che ha consentito agli scienziati di andare oltre quelli che erano i limiti dei metodi tradizionali. Inizialmente i ricercatori guardavano il 90% del genoma umano unicamente come "junk DNA" mentre ora esso è apprezzato per il suo ruolo di controllo sui geni espressi includendo anche informazioni quali: le regioni in cui si sono espressi, in quale momento e in che entità. Grazie ai GWAS (Genome-Wide Associations Studies) è stato rilevato che la maggior parte delle varianti identificate sono presenti nelle regioni non-codificant del DNA e ciò sottolinea quanto sia importante la *gene expression e regulation* nel meccanismo di una malattia. RNA-Seq rivela il trascrittoma completo (full transcriptome), non solo pochi trascritti selezionati. L'RNA-Seq fornisce visibilità all'interno di cambiamenti nella gene expression che prima non erano rilevabili e consente la caratterizzazione di forme multiple di RNA non codificante. Grazie all'autentico potere di scoperta del rilevamento imparziale dell'RNA, l'RNA-Seq è rapidamente emerso come l'approccio più importante per la profilazione del trascrittoma ad alto throughput e attualmente è uno dei tool più potenti e significativi.

---

<sup>1</sup>Il cancro è un gruppo di malattie causate da cambiamenti nel DNA che alterano il comportamento delle cellule, provocando una crescita incontrollabile a livello molecolare e causando gravi danni al sistema cellulare. Queste anomalie possono assumere diverse forme, tra cui mutazioni del DNA, riarrangiamenti, soppressioni, amplificazioni e aggiunta o rimozione di marcature chimiche. Questi cambiamenti possono indurre le cellule a produrre quantità anomale di particolari proteine o a produrre proteine malformate che non funzionano come dovrebbero. Spesso, una combinazione di diverse alterazioni genomiche portano alla formazione di un cancro. Per risolvere e capire al meglio questi cambiamenti genomici si utilizzano dati clinici che descrivono la risposta dei pazienti al trattamento del cancro, esperimenti di laboratorio che utilizzano linee cellulari e organismi modello e tecniche di analisi dei big data.

<sup>2</sup>2.5 petabytes

L'analisi dell'espressione genica con RNA-Seq è considerata uno strumento fondamentale per scoprire i meccanismi del cancro e aiutare la ricerca sulle malattie genetiche. L'RNA-Seq fornisce anche una visione dei trascritti non codificanti e ne illumina il ruolo nelle malattie complesse. Gli avanzamenti nell'RNA-Seq hanno permesso ai ricercatori di esaminare i dettagli dello sviluppo del cancro e delle malattie infettive a livello di singola cellula con un contesto a livello di tessuto.

## 1.2 RNA-Seq vs Microarray DNA

L'RNA-Seq è un potente metodo basato sul sequenziamento che cattura uno spettro completo e informativo dei dati di espressione genica. A differenza dei microarray, RNA-Seq non richiede l'uso di sonde (probe) predefinite e dunque i set di dati sono imparziali e consentono una progettazione sperimentale priva di ipotesi. Tale strumento si rivela fondamentale negli studi di *transcript discovery* e di varianti che non sarebbero possibili con l'utilizzo di metodi tradizionali che prevedono che l'esperimento abbia un target noto.

L'RNA-Seq offre una copertura più fine del trascrittoma e una minore variabilità tecnica rispetto ai microarray, con la capacità di rilevare una percentuale più elevata di geni espressi in modo differenziato, in particolare sui geni a bassa abbondanza.

## 1.3 Funzionamento di RNA-Seq

L'RNA-Seq conta le singole *sequence read* allineate ad una *reference sequence* per generare conteggi discreti di read. I ricercatori, aumentando o diminuendo il numero di sequence read (*coverage depth*), possono regolare con precisione la sensibilità di un esperimento al fine di soddisfare i diversi obiettivi di studio. La natura quantitativa di questo processo e la capacità di controllare i livelli di copertura supportano un intervallo dinamico estremamente ampio, con valori di espressione assoluti piuttosto che relativi.

## 1.4 RNA-Seq per la ricerca sul cancro

L'RNA-Seq si è rivelato uno strumento essenziale per misurare in maniera diretta le conseguenze funzionali delle mutazioni dei geni. Dalle 46 mutazioni circa contenute nel cancro medio, ne servono solo da 5 a 8 per dare inizio al tumore. Il solo profilo genomico è insufficiente per differenziare queste "driver mutation" dalle "passenger mutation", ovvero quelle mutazioni che non influenzano l'inizio o la progressione del cancro. Combinando le misurazioni dei modelli di espressione genica e le conseguenze delle mutazioni tramite RNA-Seq è possibile differenziare, su larga scala e in maniera imparziale, i fattori cruciali per la progressione del cancro e ciò consente una modellazione più approfondita e accurata dello stesso. Il rilevamento delle fusioni geniche è particolarmente significativo per la ricerca sul cancro, poiché il 20% di tutti i tumori umani presenta traslocazioni e fusioni geniche. La maggior parte delle fusioni geniche ha un impatto significativo sulla tumorigenesi e una forte associazione con il fenotipo morfologico, rendendole utili come potenziali marker diagnostici e prognostici.

La parte restante di questa relazione è organizzata come segue: nella sezione 2 vengono passati a rassegna i lavori correlati alla classificazione dei tumori e alla visualizzazione delle deep neural network. Nella sezione 3 vengono descritti i dati utilizzati, le metodologie applicate per la classificazione dei tumori e la procedura per generare le heatmap. Nella sezione 4 vengono discussi i risultati ottenuti e la validazione tramite Pathway Enrichment Analysis dei top gene estratti dalle heatmap. Infine, nella sezione 5 presentiamo quella che è stata la nostra modifica al lavoro di Lyu e Haque [1].

## 2 Lavori correlati

### 2.1 Metodi di Machine Learning e Deep Learning per la classificazione

Molti paper scientifici di questi ultimi anni si soffermano maggiormente sull'individuazione dei top gene per una singola tipologia di tumore mentre altri si limitano ad effettuare una classificazione binaria (identificazione) dei tumori usando i *gene expression data*. Il paper [9], invece, ha effettuato una ricerca sulla classificazione di multipli tipi di tumore usando tecniche classiche di Machine Learning. Gli autori di tale lavoro hanno applicato iterativamente il metodo GA/KNN al fine di generare, ad ogni iterazione, un sottoinsieme di geni (le feature) da passare a KNN per testarne l'accuracy. Con questo procedimento hanno ottenuto un'accuracy del 90% su 31 classi tumorali e sono riusciti a generare un insieme di top gene valido per tutti i tipi di tumore analizzati. Per quanto questo sia un metodo robusto e alla fine riesca ad ottenere un feature set ottimale esso richiede molte iterazioni e la dimensione del feature set è fissata a priori. Inoltre, usando un unico feature set, viene trascurata quella che è l'eterogeneità tra i differenti tipi di tumore e non si vanno a considerare quelli che sono i top gene specifici di una determinata classe, tenendo conto che i top gene potrebbero variare molto da classe a classe. Analizzando, invece, tecniche di Deep Learning, nel paper [10] gli autori hanno usato gli stacked auto-encoder per estrarre feature di alto livello dai valori di espressione dei geni. Tali feature vengono date in input ad una ANN a singolo layer per decidere se il sample è un tumore oppure no. L'accuracy di tale modello ha raggiunto il 94%. Questo metodo, tuttavia, ha una struttura estremamente complessa dal momento che non è un metodo end-to-end. Per identificare i top gene di BRCA, dapprima le matrici dei pesi di ogni layer dell'auto-encoder vengono moltiplicate per ottenere i pesi stimati di ogni gene dell'input layer e poi estraggono i top gene facendone il fit con la distribuzione normale. L'idea di tale lavoro è molto simile a quella seguita successivamente da [1] usando però metodi di visualizzazione delle DNN. La differenza sta nel fatto che in [10] hanno fatto corrispondere l'importanza dei geni alle feature di alto livello, mentre in [1] hanno fatto corrispondere l'importanza dei geni alla loro contribuzione nella classificazione.

### 2.2 Metodi di Visualizzazione delle Deep Neural Network

Le Deep Neural Network sono spesso descritte come delle "black box" perché non è così ovvio e scontato il perché una rete prenda determinate decisioni. Con il crescente utilizzo delle DNN anche in ambito medico, però, è diventato cruciale comprendere come una DNN arrivi a prendere una determinata decisione. A seguito di tale esigenza sono state sviluppate diverse tecniche, tra cui, la Deep Taylor Decomposition e la layer-wise relevance propagation (LRP) [11, 12]. Tali tecniche sono progettate per interpretare la DNN tramite backpropagation. Una maniera conservativa di eseguire LRP è quella di redistribuire l'input di ogni neurone all'indietro a tutti i suoi predecessori ugualmente layer dopo layer e, quando viene raggiunto il layer di input, la decomposizione risulta effettuata. Tale metodo però va ad inserire dei vincoli sulla rete in quanto richiede una proprietà di relevance conservation tra i layer. Esiste una tecnica invece che non va a modificare la rete di partenza e può essere applicata a qualsiasi rete neurale. Tale tecnica è la Guided Grad-CAM [2], una combinazione di Guided back-propagation e Grad-CAM. In questo lavoro, così come fatto da [1], si userà tale tecnica per tener traccia dei pixel significativi (i geni) in input al fine di poter estrarre i top gene. La localization map di Grad-CAM è generata dapprima calcolando l'activation map nella forward propagation e poi calcolando la gradient map nella backward propagation. Ciò è così fatto dal momento che, più profondo è il layer convoluzionale, più alto è il livello delle feature che esso contiene. Dal momento che, una volta che i dati raggiungono i layer fully-connected, tutti i dati spaziali vengono persi, Grad-CAM costruisce la localization map dell'ultimo layer convoluzionale e la ridimensiona alla dimensione dell'input. Tuttavia, come affermato in [2], la heatmap mostra l'importanza di ogni classe ma la risoluzione è bassa a causa del processo di ridimensionamento. Per risolvere questo problema, è stato usato Guided backpropagation, un metodo di pixel-space gradient visualization, per raffinare la heatmap. L'output di Guided backpropagation è il gradiente di ogni pixel nell'immagine di input e

può essere ottenuto attraverso backpropagation. Il risultato finale di Guided Grad-CAM può, quindi, essere ottenuto moltiplicando i risultati di Guided backpropagation e di Grad-CAM.

### 3 Metodologia Applicata

#### 3.1 Workflow

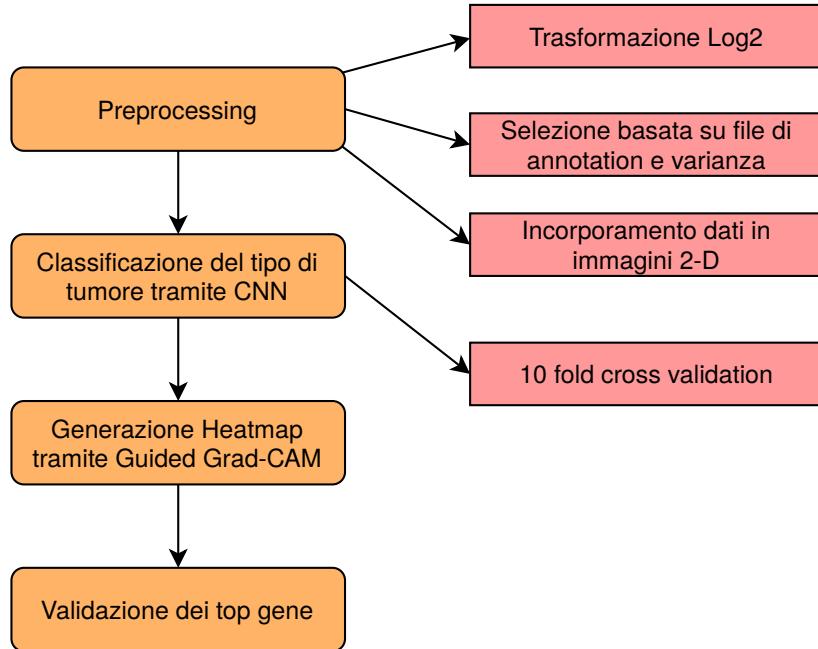


Figura 1: Il workflow del nostro metodo (ripreso da [1])

Il workflow del nostro metodo ricalca in buona parte quello di [1] ed è mostrato in Figura 1:

1. Unione dei dati grezzi per ottenere i dataset su cui operare (tale passo non è riportato nella Figura 1 in quanto è stato utilizzato solo per il test binario e il test ternario e dunque non viene effettuato in quello che è il test principale di questo lavoro).
2. Preprocessing sui dati che rappresentano le espressioni genetiche di vari genomi.
3. Classificazione delle varie tipologie di tumori tramite l'uso di una rete neurale convoluzionale.
4. Valutazione delle performance della rete neurale.
5. Generare le heatmap per ogni classe tumorale e selezionare i geni corrispondenti ai pixel con la massima intensità.
6. Convalidare i percorsi biologici dei geni selezionati.

#### 3.2 Dati utilizzati

Sono stati utilizzati i **normalized-level3 RNA-Seq gene expression** data di 33 tipi di tumore presenti nel Pan-Cancer Atlas. Un elenco dettagliato di tutti i 33 tipi di tumore e del corrispondente numero di campioni è riportato nella Tabella 1 a pagina 7. I dati contengono 10.267 campioni di tumore su 20.531 campioni totali. I dati sono stati scaricati da *Firehose*[13], che è un tool sviluppato dal TGCA network e gestito dal Broad Institute. *Firehose* è "una serie di tool e pipeline sviluppati per processare ed analizzare vari tipi di dati genomici e proteici su vasta scala" [14].

Per il caso binario (DLBC e UCS) e per il caso ternario (BLCA, CESC e LGG) abbiamo ricreato il dataset di partenza unendo i raw data delle classi prese in esame, abbiamo effettuato data cleaning sui dati in quanto erano presenti dei sample non appartenenti al tessuto tumorale e abbiamo aggiunto le label per distinguere le classi. Per il caso generale, invece, abbiamo deciso di utilizzare il dataset messo a disposizione da [1] che è stato creato a partire dagli stessi dati.

Tabella 1: Numero campioni di RNA-Sequence tumorali.

Classe tumorale	Coorte	Numero Campioni
Adrenocortical carcinoma	ACC	79
Bladder urothelial carcinoma	BLCA	408
Breast invasive carcinoma	BRCA	1093
Cervical and endocervical cancers	CESC	304
Cholangiocarcinoma	CHOL	36
Colon adenocarcinoma	COAD	457
Lymphoid Neoplasm Diffuse Large B-cell Lymphoma	DLBC	48
Esophageal carcinom	ESCA	184
Glioblastoma multiforme	GBM	160
Head and Neck squamous cell carcinoma	HNSC	520
Kidney Chromophobe	KICH	66
Kidney renal clear cell carcinoma	KIRC	533
Kidney renal papillary cell carcinoma	KIRP	290
Acute Myeloid Leukemia	LAML	179
Brain Lower Grade Glioma	LGG	516
Liver hepatocellular carcinoma	LIHC	371
Lung adenocarcinoma	LUAD	515
Lung squamous cell carcinoma	LUSC	501
Mesothelioma	MESO	87
Ovarian serous cystadenocarcinoma	OV	304
Pancreatic adenocarcinoma	PAAD	178
Pheochromocytoma and Paraganglioma	PCPG	179
Prostate adenocarcinoma	PRAD	497
Rectum adenocarcinoma	READ	166
Sarcoma	SARC	259
Skin Cutaneous Melanoma	SKCM	469
Stomach adenocarcinoma	STAD	415
Testicular Germ Cell Tumors	TGCT	150
Thyroid carcinoma	THCA	501
Thymoma	THYM	120
Uterine Corpus Endometrial Carcinoma	UCEC	545
Uterine Carcinosarcoma	UCS	57
Uveal Melanoma	UVM	80
<b>Totale campioni</b>		<b>10267</b>

### 3.2.1 RSEM: RNA-Seq Expectation Maximization

I dati utilizzati nell'ambito di questo progetto sono stati ottenuti dal FireHose Broad GDAC<sup>3</sup>. Questi ultimi sono ottenuti dall'applicazione di un software denominato RSEM [15] così definito: RSEM è un pacchetto software di facile utilizzo per quantificare le abbondanze di geni e isoforme da dati RNA-Seq single-end o paired-end. RSEM, inoltre, è stato progettato per lavorare con letture allineate a sequenze di trascritti, anziché a letture allineate a sequenze di genomi interi. L'utilizzo di allineamenti a livello di trascrizione ha come vantaggio quello di consentire facilmente l'analisi di campioni di specie senza genomi sequenziati ma con un trascrittoma decentemente caratterizzato (magari attraverso l'assemblaggio del trascrittoma RNA-Seq [16, 17]). Infine, la lunghezza totale di tutti i possibili trascritti è spesso molto inferiore alla lunghezza del genoma, consentendo un allineamento più rapido a livello di trascritto. RSEM produce stime di abbondanza, intervalli di credibilità al 95%, file di visualizzazione, può anche simulare i dati RNA-Seq, e a differenza di altri strumenti esistenti, il software non richiede un genoma di riferimento. Pertanto, in combinazione con un assemblatore di trascrittomi de novo, RSEM consente una quantificazione accurata dei trascritti per le specie prive di genomi sequenziati. Su set di dati simulati e reali, RSEM ha prestazioni superiori o paragonabili ai metodi di quantificazione che si basano su un genoma di riferimento. Sfruttando la capacità di RSEM di utilizzare efficacemente le letture a mappatura ambigua, è stato dimostrato che le stime accurate dell'abbondanza a livello genico si ottengono al meglio con un gran numero di short read single-end. D'altra parte, le stime delle frequenze relative delle isoforme all'interno di singoli geni possono essere migliorate con l'uso di letture paired-end, a seconda del numero di possibili forme di splice per ciascun gene. Inoltre RSEM possiede diversi allineatori di sequenze genomiche come Bowtie, Bowtie2 e STAR. Gli input di tali strumenti risultano quindi essere le sequenze di RNA prodotte dalla piattaforma genomica Illumina sottoforma di file formato fasta o fastaq. Oltre a ciò, viene dato anche un file contenente l'annotazione genetica che RSEM userà come riferimento per stimare il gene expression level. Dopo l'allineamento delle read tramite uno degli algoritmi sopra citati, RSEM calcola le stime di abbondanza ML utilizzando l'algoritmo Expectation-Maximization (EM) per il suo modello statistico. Sono disponibili diverse opzioni per specificare il modello utilizzato da RSEM, che deve essere personalizzato in base al protocollo RNA-Seq che ha prodotto le letture in ingresso. Ad esempio, se viene utilizzato un protocollo specifico per ogni filamento, è necessario specificare l'opzione **-strand-specific**. Altrimenti, si presume che una lettura abbia la stessa probabilità di provenire dalla direzione *senso* o *antisenso*. L'output principale di RSEM consiste di due file: uno per le stime a livello di isoforma e l'altro per le stime a livello di gene. Le stime di abbondanza sono fornite in termini di due misure. La prima è una stima del numero di frammenti derivati da una determinata isoforma o gene. La seconda, è la frazione stimata di trascritti costituita da una determinata isoforma o gene. Questa misura può essere utilizzata direttamente come valore compreso tra zero e uno o può essere moltiplicata per  $10^6$  per ottenere una misura in termini di trascritti per milione (TPM).

## 3.3 Fase di Preprocessing

I dati contengono il conteggio normalizzato delle read per ogni gene, ma l'intervallo dei valori è enorme e alcuni valori sono più piccoli di 1. Perciò, come [1], abbiamo applicato prima una trasformazione usando  $y = \log_2(x+1)$  per ridurre la scala. Poi abbiamo posto a 0 tutti i valori inferiori a 1, poiché è molto probabile che si tratti di rumore. Successivamente, abbiamo confrontato i geni con

<sup>3</sup>Con sede a Cambridge, nel Massachusetts, il Broad Institute è stato fondato nel 2004 per realizzare la promessa della medicina genomica, tre anni dopo il completamento del Progetto Genoma Umano, che gli scienziati del Broad hanno contribuito a creare e guidare. Le nostre origini sono radicate nella genomica e, dal 2004, la ricerca biomedica e la conoscenza si sono ampliate, così come noi. I nostri ricercatori sono profondamente collaborativi, lanciano agilmente progetti innovativi e ad alto rischio su ogni scala, acquisiscono conoscenze sui meccanismi biologici delle malattie, inventano nuove tecnologie, costruiscono e implementano strumenti computazionali, sviluppano nuove terapie da portare in clinica, fanno da tutor e formano la prossima generazione di scienziati oltre che condividere apertamente i dati e gli strumenti per consentire progressi in ogni settore della società.

un file di annotazione (aggiornato al 03 aprile 2018) scaricato dall'NCBI. Circa 1000 geni non sono stati trovati nel file di annotazione, quindi sono stati rimossi. Abbiamo poi utilizzato una soglia di varianza pari a 1,19 per filtrare i geni i cui livelli di espressione rimangono quasi invariati tra i campioni. Il numero di geni è stato così ulteriormente ridotto a 10.381. Questa soglia è stata scelta considerando sia l'accuratezza della classificazione sia l'integrità dell'insieme di geni. Una soglia più alta ridurrà maggiormente le "weak feature", in modo da migliorare l'accuratezza della classificazione. Una soglia più bassa, invece, permette di mantenere un maggior numero di geni, in modo da non perdere i reali biomarker specifici del gene. Inoltre, per comodità nella fase successiva, abbiamo voluto che il numero di geni filtrati fosse vicino a un numero quadrato. Per dare in pasto i dati alla rete neurale convoluzionale, essi sono stati incorporati in immagini 2D. I primi geni sono ordinati in base al numero di cromosoma, perché è più probabile che geni adiacenti interagiscano tra loro. Poi i dati sono stati rimodellati da una matrice  $10.381 \times 1$  in un'immagine  $102 \times 102$  aggiungendo alcuni zeri all'ultima riga dell'immagine. Quindi tutte le immagini vengono normalizzate per assicurarsi che l'intervallo sia  $[0, 255]$ . Le immagini generate delle diverse classi sono mostrate nella Figura 2 a pagina 9.

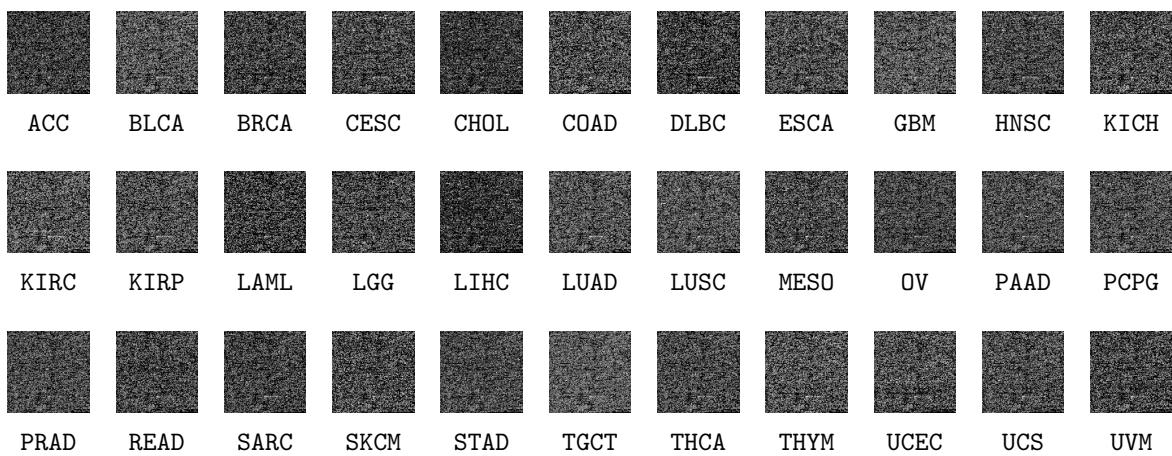


Figura 2: Un esempio di immagini 2D incorporate dalla classe 1 alla classe 33.

### 3.4 Classificazione delle immagini tramite Neural Network

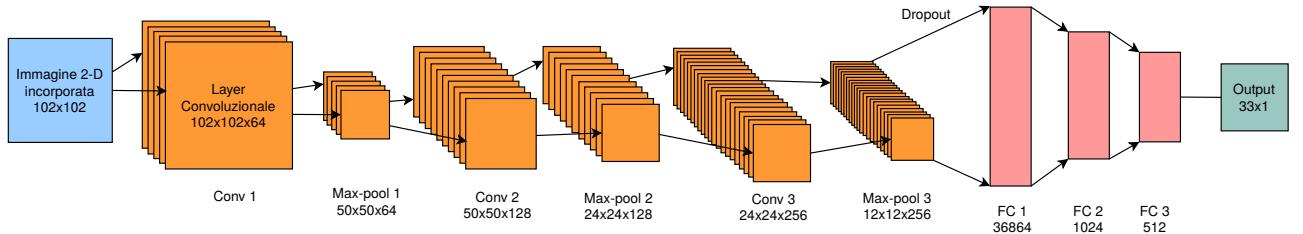


Figura 3: L'architettura della rete neurale convoluzionale

A causa dello squilibrio nel set di dati, alcune classi hanno pochi campioni, quindi la rete neurale non dovrebbe contenere troppi strati al fine di evitare un grave overfitting. Alla fine abbiamo utilizzato la rete neurale convoluzionale di [1] composta da tre layer convoluzionali e tre layer completamente connessi, come mostrato in Figura 3. Il primo layer convoluzionale "conv1" contiene 64 filtri diversi, mentre il secondo e il terzo layer convoluzionale contengono rispettivamente 128 e 256 filtri. Il layer di max-pooling e il layer di batch normalization sono posti immediatamente dopo ogni layer convoluzionale. Prima di entrare nel layer fully-connected viene aggiunto un layer di drop-out, il cui tasso di drop-out è del 25%. Le dimensioni dei tre layer fully-connected sono 36.864, 1.024 e 512 rispettivamente. Abbiamo scelto la Cross Entropy come funzione di perdita e l'ottimizzatore Adam

per aggiornare i pesi. Abbiamo utilizzato la 10 fold cross validation per addestrare la rete neurale convoluzionale e testarne le prestazioni.

### 3.4.1 Convolutional Neural Network

La CNN è una rete neurale feedforward che è in grado di estrarre caratteristiche dai dati, tramite connessioni locali e un pattern di pesi replicati in ogni layer nascosto della rete. Questo pattern è definito kernel ed il processo di applicazione del kernel sull'input è detto convoluzione. Diversamente dai metodi classici di estrazione delle caratteristiche [18, 19], le CNN non hanno bisogno di estrarre caratteristiche manualmente. L'architettura della CNN si ispira alla percezione visiva [20], quindi imita il funzionamento del sistema oculare umano. In questa visione possiamo entrare nel dettaglio di questa struttura, affermando che un neurone biologico del cervello che fa da feedback alla corteccia visiva può corrispondere ad un neurone artificiale che compone uno strato della rete; i layer che applicano la convoluzione usano i kernel che fungono da recettori e possono catturare varie caratteristiche della realtà; le funzioni di attivazione possono simulare il funzionamento di trasmissione tra vari gruppi di neuroni, imponendo che solo i segnali elettrici neurali, superiori ad una certa soglia, possono essere trasmessi al gruppo successivo o al singolo neurone. Le funzioni di perdita e gli ottimizzatori sono strumenti che permettono al sistema CNN di imparare e migliorarsi per raggiungere gli obiettivi che ci aspettiamo. Rispetto al layer fully connected (FC), come in Figura 4, le CNN possiedono molti

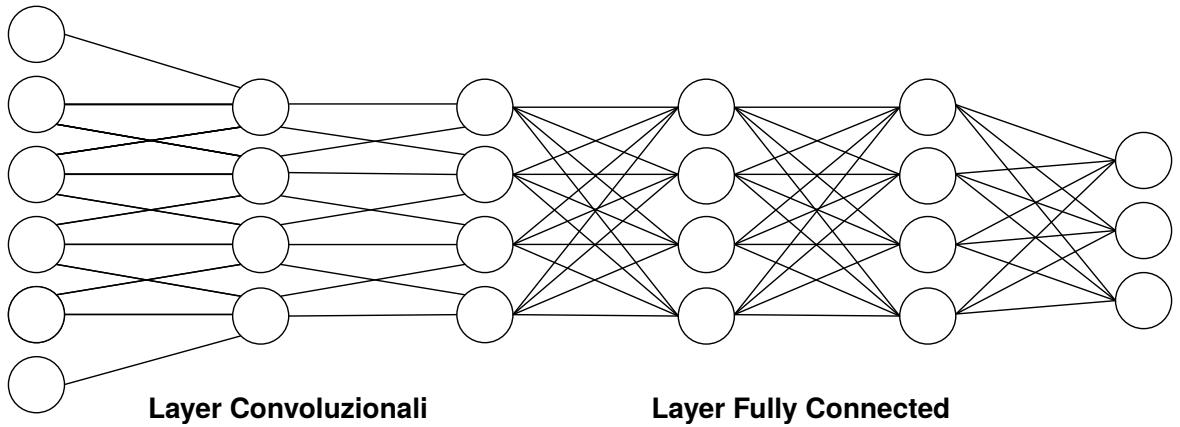


Figura 4: Confronto archittture CNN e FC [3]

vantaggi:

1. **Connessioni locali:** Ogni neurone non è più collegato a tutti i neuroni del layer precedente, ma solo ad un piccolo numero di neuroni. Ciò è utile a ridurre i parametri ed accelerare la convergenza del modello.
2. **Ripartizione del peso:** un gruppo di connessioni possono condividere gli stessi pesi e ciò riduce ulteriormente i parametri.
3. **Riduzione della dimensione di downsampling:** il principio della correlazione locale dell'immagine applicata dall'operazione di convoluzione serve a filtrare la dimensione dell'immagine (downsample dell'immagine) conservando le informazioni più complesse che devono essere processate da uno strato più profondo. Il downsample rimuove anche le caratteristiche banali consentendo al modello di avere un minor numero di parametri negli strati più profondi.

Queste tre caratteristiche accattivanti rendono le CNN uno degli algoritmi più rappresentativi nel campo del deep learning. Il termine "convoluzione" si riferisce ad un processo matematico lineare tra matrici. I layer convoluzionali, non lineari, di pooling e fully connected sono solo alcuni tra i molti

layer che compongono una CNN. Nei layer di una CNN si fa distinzione tra layer con parametri e layer senza parametri. I layer con parametri sono quelli convoluzionali e quelli fully connected. Quelli senza parametri sono i layer di pooling e i layer non lineari. Nel sottoparagrafo seguente analizzeremo come agiscono i differenti layer durante la classificazione effettuata con una CNN.

### 3.4.2 Convolutional Neural Network Layers

Le CNN sono composte da vari strati che hanno la capacità di comporre l'input tramite l'uso di funzioni non lineari e di associarlo ad una target class. I principali componenti base per la costruzione di una CNN sono:

**Convolutional layer** Il layer di convoluzione è il layer più semplice ma allo stesso tempo più importante di una CNN. Fondamentalmente applica il kernel oppure moltiplica o la matrice dei pixel generata per l'immagine o l'oggetto dato in input per produrre una mappa di attivazione per l'immagine fornita. Il vantaggio principale della mappa di attivazione è che memorizza tutte le caratteristiche distintive di una data immagine riducendo allo stesso tempo la quantità di dati da elaborare. La matrice con cui i dati sono convogliati è un rilevatore di caratteristiche che fondamentalmente è un insieme di valori con cui la rete è compatibile. Diverse versioni dell'immagine vengono generate utilizzando diversi valori del rilevatore di funzionalità. Inoltre il modello CNN è anche addestrato con backpropagation per accertare l'errore minimo in ogni layer e, a seconda dell'errore più basso ottenuto dal layer, si impone la profondità, ossia il numero di filtri, e il padding. La Figura 8 mostra come funziona la convoluzione. Questo passaggio comporta la convoluzione della matrice contenente i dati dell'immagine e quindi il rilevatore di caratteristiche che ci dà una mappa di attivazione o una mappa caratteristica. Ciò che accade nella convoluzione è che i valori su posizioni identiche nei dati e nella mappa delle caratteristiche, cioè i valori che hanno valore 1 o più di 1, vengono mantenuti mentre il resto viene rimosso. La matrice dei dati dell'immagine viene confrontata  $3 \times 3$  alla volta. La dimensione del rilevatore di caratteristiche varia a seconda del tipo di CNN utilizzato. Per esempio ci sono versioni di CNN che utilizzano filtri  $5 \times 5$  o anche filtri in scala  $7 \times 7$ . La convoluzione segue la seguente formula:

$$(f^*g)(t) = \int_{-\infty}^{\infty} f(\beta) g(t - \beta) d\beta$$

che mira a mostrare come una funzione modifica la forma dell'altra. Nell'immagine 9 si nota che i dati generati per questa immagine sono stati modificati usando il filtro per generare la mappa di attivazione. L'insieme di tutte le feature map create con vari filtri formano un livello di convoluzione. Altri fattori che influiscono sulla potenzialità di un strato convoluzionale sono lo **Stride** e il **Padding**:

**Stride** In realtà, per le CNN esistono altre possibilità per diminuire ulteriormente i parametri in un layer convoluzionale, impedendo anche effetti collaterali dovuti a questa diminuzione. Una di queste opzioni è lo stride. Si assume per semplicità questa ipotesi di base che afferma che il nodo del livello successivo abbia molte sovrapposizioni con i suoi vicini quando guardano le regioni di una immagine. Possiamo manipolare queste sovrapposizioni tra i campi recettivi che compongono un layer convoluzionale gestendo lo stride. La Figura 5, è un esempio che inizia con una data immagine  $7 \times 7$ . Se spostiamo il filtro un nodo alla volta, possiamo avere solo un'uscita  $5 \times 5$ . Si noti che l'uscita delle tre matrici di sinistra in Figura 5, hanno una sovrapposizione (un totale di tre tra quelle centrali e di tre tra quelle di destra). Tuttavia, se ci muoviamo e facciamo ogni movimento con stride 2, allora l'uscita sarà  $3 \times 3$ . In parole povere, non solo il fenomeno della sovrapposizione, ma anche la dimensione della output image sarà ridotto [21].

$$O = 1 + \frac{N - F}{S} \quad (1)$$

L'Equazione (1), formalizza quanto detto in precedenza, afferma che data l'immagine di dimensione  $N \times N$  e la dimensione del filtro data come  $F \times F$ , la dimensione dell'output  $O$  ed è mostrato in Figura 6 dove  $N$  è la dimensione dell'immagine d'input,  $F$  è la dimensione del filtro, e  $S$  è la dimensione del passo.

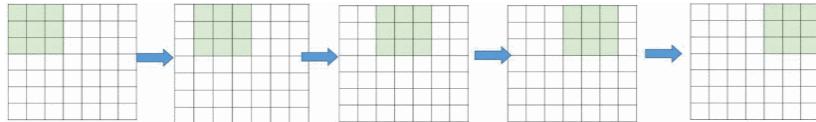


Figura 5: Esempio di movimento di un filtro  $3 \times 3$  con stride 1. [4]

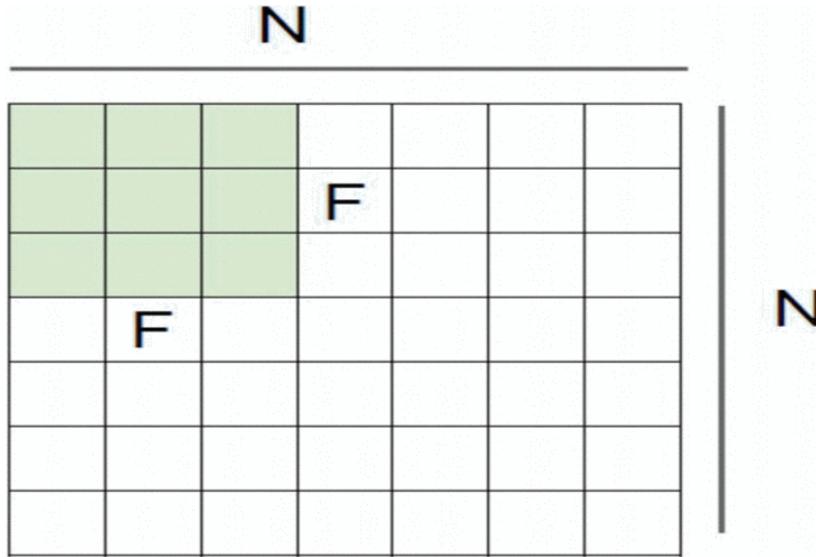


Figura 6: Risultato dell'applicazione dello stride. [4]

**Padding** Uno degli svantaggi della fase di convoluzione è la perdita di informazioni che potrebbero esistere sul bordo dell'immagine, poiché durante la convoluzione i pixel del bordo sono visti ed usati meno volte rispetto a quelli del centro, dato che il filtro li osserva solo quando si muove attraverso l'immagine. Un metodo molto semplice ma efficace per risolvere il problema, è usare zero-padding. L'altro vantaggio di zero-padding è quello di gestire la dimensione dell'output. Per esempio, in Figura 5, con  $N = 7$  e  $F = 3$  e stride 1, l'uscita sarà  $5 \times 5$  (che si restringe da un ingresso  $7 \times 7$ ). Tuttavia, aggiungendo uno zero-padding, l'output sarà  $7 \times 7$ , che è esattamente lo stesso dell'input originale (il valore attuale di  $N$  diventa 9, usata nella (1). L'equazione modificata che include zero-padding è la (2).

$$O = 1 + \frac{N + 2P - F}{S} \quad (2)$$

dove  $P$  è il numero dei livelli di riempimento applicati dallo zero-paddding (ad esempio  $P = 1$  in Figura 7). Questa idea di riempimento ci aiuta a impedire che la dimensione dell'uscita di rete si restrin ga con la profondità. Pertanto, è possibile avere un numero qualsiasi di reti convoluzionali profonde [21].

**Pooling Layer** Il pooling è un passo importante per ridurre ulteriormente le dimensioni della mappa di attivazione, mantenendo solo le caratteristiche importanti e riducendo l'invarianza spaziale. Questo a sua volta riduce il numero di caratteristiche apprendibili dal modello. Ciò aiuta a risolvere il problema del sovrardimensionamento. Il pooling consente alla CNN di incorporare tutte le diverse dimensioni di un'immagine in modo che la rete riconosca con successo l'oggetto dato anche se la sua forma è inclinata o è presente con un'angolazione diversa. Ci sono vari tipi

0	0	0	0	0	0	0	0	0
0								0
0								0
0								0
0								0
0								0
0								0
0								0
0	0	0	0	0	0	0	0	0

Figura 7: Effetto dello zero padding sull'immagine di input. [4]

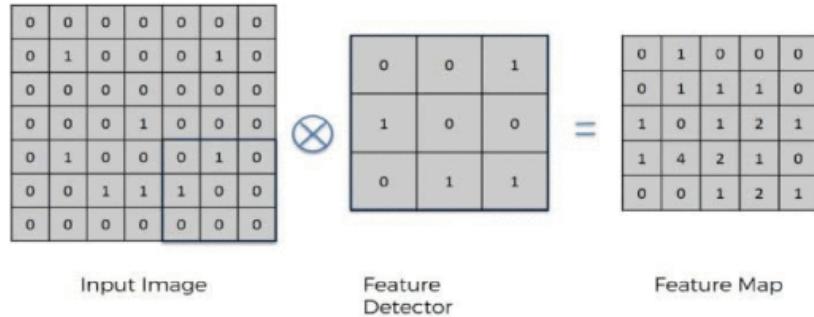


Figura 8: Funzionamento della convoluzione [5]

di pooling come il Max Pooling, l’Average Pooling, lo Stochastic Pooling e lo Spatial Pyramid Pooling. Tra questi, il più popolare è il Max Pooling che prende il valore più alto da ogni sub matrice della mappa di attivazione e forma una matrice separata. In questo modo si assicura che le caratteristiche apprendibili rimangano limitate nel numero, preservando allo stesso tempo le caratteristiche chiave di qualsiasi immagine. Il Max Pooling è solitamente fatto usando un filtro  $2 \times 2$ . La Figura 10 mostra il funzionamento di tale layer.

**Batch Normalization layer** La Batch Normalization (BatchNorm) [22, 5] è stata probabilmente una delle innovazioni architettonali di maggior successo nel deep learning e anche se la sua efficacia è indiscutibile, non abbiamo una ferma comprensione del perché questo è il caso. In generale, BatchNorm è un meccanismo che mira a stabilizzare la distribuzione (su un mini lotto) di input ad un determinato livello di rete durante l’addestramento. Ciò si ottiene aumentando la rete con strati aggiuntivi che impostano i primi due momenti (media e varianza) della distribuzione di ciascuna attivazione rispettivamente a 0 (zero) e 1 (uno). Poi, gli ingressi normalizzati batch sono in genere anche in scala e spostati sulla base di parametri addestrabili per preservare l’espressività del modello. Questa normalizzazione viene applicata prima della non-linearietà del layer precedente. Una delle motivazioni chiave per lo sviluppo di BatchNorm è stata la riduzione del cosiddetto Spostamento Covariato Interno (Internal Covariate Shift). Questa riduzione è stata ampiamente considerata come il fulcro del successo di BatchNorm. Ioffe e Szegedy [22] descrivono l’ICS come "il fenomeno in cui la distribuzione degli input a un livello della rete cambia a causa di un aggiornamento dei parametri dei livelli precedenti". Questo cambiamento porta ad un costante spostamento del problema di formazione di base e si ritiene quindi che abbia un effetto negativo sul processo di training.

**Drop-out Layer** Drop-out [23] è un metodo per regolarizzare le reti neurali, proposto per evitare

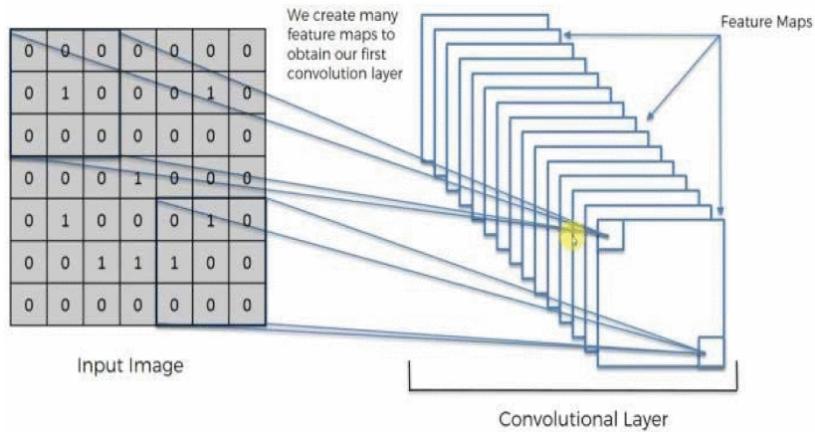


Figura 9: Layer Convoluzionale [5]

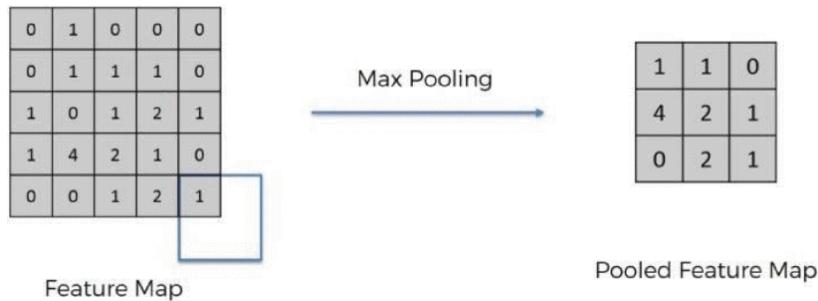


Figura 10: Applicazione Max Pooling [5]

l'over-fitting. È un metodo di regolarizzazione<sup>4</sup> che imposta stocasticamente a zero le attivazioni delle unità nascoste per ogni caso di addestramento al momento della fase di training. Questo interrompe il co-adattamento dei rilevatori di funzionalità poiché le unità degli strati nascosti selezionati dal drop-out non possono influenzare le restanti unità. Un altro modo per interpretare il drop-out è che produce una forma molto efficiente di media del modello in cui il numero di modelli addestrati è esponenziale in quello delle unità, e questi modelli condividono gli stessi parametri. Il drop-out ha anche ispirato altri metodi stocastici come lo Stochastic Pooling [24] e DropConnect [25]. Anche se il Drop-out è noto per funzionare bene in layer fully connected di reti neurali convoluzionali il suo effetto non è stato ancora ben documentato [25, 24, 26]. Il Drop-out è una nuova tecnica di regolarizzazione che è stata più recentemente impiegata nel deep learning. È simile al bagging [27], in cui un insieme di modelli sono addestrati su diversi sottoinsiemi degli stessi dati di allenamento. Al momento della prova, le previsioni dei diversi modelli sono mediate insieme. Nel bagging, ogni modello ha parametri indipendenti e tutti i membri sarebbero addestrati individualmente. Nel caso di formazione tramite drop-out, ci sono esponenzialmente molti modelli eventualmente da addestrare ma non lo sono tutti in maniera palese, poiché questi ultimi condividono gli stessi parametri. In realtà, il numero di modelli addestrati tramite drop-out non è più grande di  $\gamma \times \epsilon$ , dove  $\gamma$  è il numero di esempi di formazione, ed  $\epsilon$  è il numero di epoche usate nel training. Questo è molto più piccolo del numero di modelli eventualmente addestrati che è pari a  $2^n$  (dove  $n$  è il numero di unità nascoste in una rete neurale feed-forward). Pertanto, la stragrande maggioranza dei modelli non sono esplicitamente addestrati al momento della formazione.

<sup>4</sup>Un metodo di regolarizzazione è spesso formalmente definito come un metodo di inversione a seconda di un singolo parametro reale  $\alpha \geq 0$  che produce una famiglia di soluzioni approssimative  $f(\alpha)$  con le seguenti due proprietà: in primo luogo, per  $\alpha$  grande abbastanza la soluzione regolarizzata  $f(\alpha)$  è stabile a fronte di perturbazioni o rumore nei dati (a differenza della soluzione generalizzata) e, in secondo luogo, per  $\alpha$  che va a zero è recuperata la soluzione generalizzata non regolarizzata:  $f(\alpha) = f() + \alpha \rightarrow 0$

**Fully Connected Layer** Questo è il layer che viene utilizzato alla fine della rete neurale. Generalmente la matrice dei pesi viene appiattita prima di essere passata ai neuroni. È difficile seguire i dati dopo questo punto a causa della presenza di molte unità nascoste con peso variabile all'uscita di ogni neurone. Tutto il ragionamento e il calcolo sui dati ed il raggiungimento di una target class associato al nostro input, viene delegato a questa tipologia di layer.

### 3.4.3 Funzioni d'attivazione

Le CNN possono sfruttare diverse funzioni di attivazione per esprimere funzioni complesse. Simile alla funzione del modello del neurone del cervello umano, la funzione di attivazione qui è un'unità che determina quale informazione dovrebbe essere trasmessa al neurone seguente. Ogni neurone nella rete neurale accetta il valore di uscita dei neuroni dal livello precedente come input e passa il valore elaborato al livello successivo. In una rete neurale multilayer, la funzione di attivazione è posta tra due layer e la struttura è mostrata nella figura 11. In Figura 11,  $x_i$  rappresenta la funzione di input;

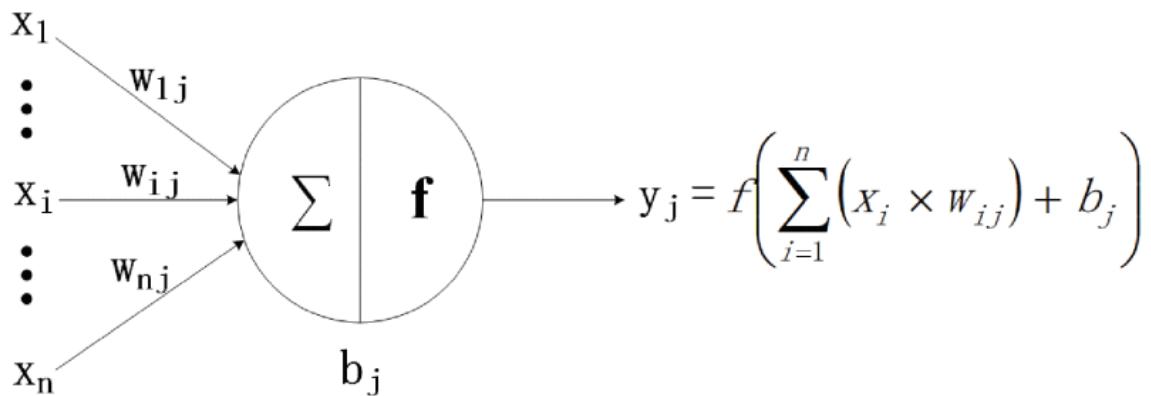


Figura 11: Diagramma generale sulle funzioni d'attivazione [3]

$n$  sono le caratteristiche in input al neurone  $j$  allo stesso tempo;  $w_{ij}$  rappresenta il valore di peso della connessione tra la funzione di input  $x_i$  e il neurone  $j$ ;  $b_j$  rappresenta lo stato interno del neurone  $j$ , che è il valore di bias; e  $y_j$  è l'uscita del neurone  $j$ .  $f()$  è la funzione di attivazione, che può essere la funzione sigmoide, la funzione Tanh [28] o l'unità lineare rettificata (ReLU) [29]. Se non viene utilizzata una funzione di attivazione o viene utilizzata una funzione lineare, l'input di ciascun livello sarà una funzione lineare dell'output del livello precedente. In questo caso, He et al. [30] dimostrano che non importa quanti layer ha la rete neurale, l'output sarà sempre una combinazione lineare dell'input, il che significa che i livelli nascosti non hanno effetto. Questa situazione rappresenta il modello del perceptron [31, 32], che ha capacità di apprendimento limitata. Per questo motivo, le funzioni non lineari sono introdotte come funzioni di attivazione. Teoricamente, reti neurali profonde con funzioni di attivazione non lineari possono approssimare qualsiasi funzione, il che migliora notevolmente la capacità delle reti neurali di adattarsi ai dati. La funzione sigmoide è una delle più tipiche funzioni di attivazione non lineari con una forma S complessiva (vedere Figura 12(a)). Quando il valore  $x$  si avvicina a 0, il gradiente diventa più ripido. La funzione sigmoide può mappare un numero reale nell'intervallo (0, 1), pertanto, può essere utilizzata per problemi di classificazione binaria. Inoltre, i modelli Senet [33] e MobileNet v3 [34] devono trasformare il valore di uscita di ogni layer in (0, 1) per poter usare il loro meccanismo di attenzione. In questo contesto la sigmoide è la funzione più appropriata da implementare. Diversamente dalla sigmoide, la funzione Tanh [28] (vedere Figura 12(b)) può mappare un numero reale nel range (-1, 1). Poiché il valore medio dell'output di Tanh è 0, può raggiungere una sorta di normalizzazione. Questo rende l'apprendimento di nuove complessità più facili per lo strato successivo. Oltre Tanh, esiste anche ReLU [29] (vedere Figura 12(c)) che risulta essere un'altra funzione di attivazione efficace. Quando  $x$  è inferiore a 0, il suo valore di funzione è

0; quando  $x$  è maggiore o uguale a 0, il suo valore di funzione è  $x$  stesso. Rispetto alla funzione sigmoide e alla funzione Tanh, un vantaggio significativo dell'utilizzo della funzione ReLU è che può accelerare l'apprendimento. Sigmoide e Tanh mentre calcolano i derivati coinvolgono nel calcolo una funzione esponenziale che include la divisione. Il derivato di ReLU, invece, è una costante. Inoltre, nella funzione sigmoide e Tanh, se il valore di  $x$  è troppo grande o troppo piccolo, il gradiente della funzione è piccolo, il che può far convergere lentamente la funzione. Tuttavia, quando  $x$  è inferiore a 0, il derivato di ReLU è 0, e quando  $x$  è maggiore di 0, la derivata è 1; quindi, si può ottenere un effetto di convergenza ideale. AlexNet [26], il miglior modello in ILSVRC-2012, utilizza ReLU come funzione di attivazione del modello basato su CNN perché mitiga il problema della scomparsa del gradiente quando la rete è profonda e verifica che l'uso di ReLU superi il sigma nelle reti profonde.

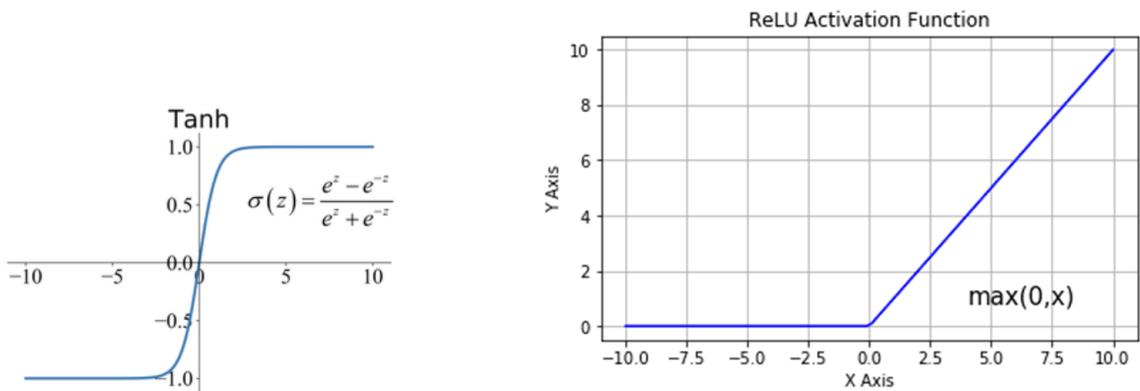
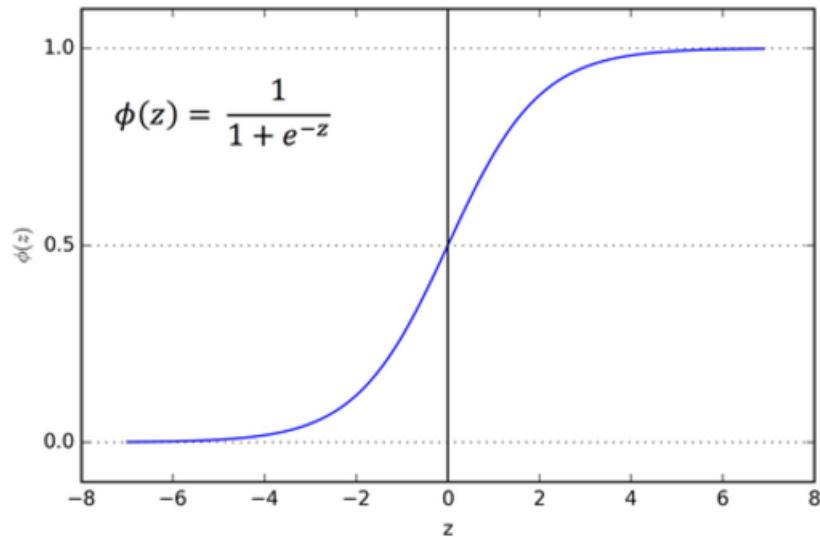


Figura 12: Esempi di funzioni d'attivazione: (a) Sigmoid, (b) Tanh, (c) ReLU

Sulla base della discussione precedente, notiamo che ReLU non include il limite superiore. In pratica, possiamo impostare un limite superiore, come ReLU6 [35].

Tuttavia un difetto di ReLU è che, quando  $x$  è minore di 0, il gradiente di ReLU è 0. Ciò significa che l'errore retropropagato verrà moltiplicato per 0, con il risultato che non verrà passato alcun errore al livello precedente. In questo scenario, i neuroni possono essere considerati inattivati o morti. Pertanto, vengono proposte alcune versioni migliorate. Un esempio è Leaky ReLU (Figura 13) che riesce a ridurre l'inattivazione dei neuroni. Quando  $x$  è minore di 0, l'errore retropropagato è  $x/a$ , invece di zero, dove  $a$  è un parametro fisso nell'intervallo  $(1, +\infty)$ .

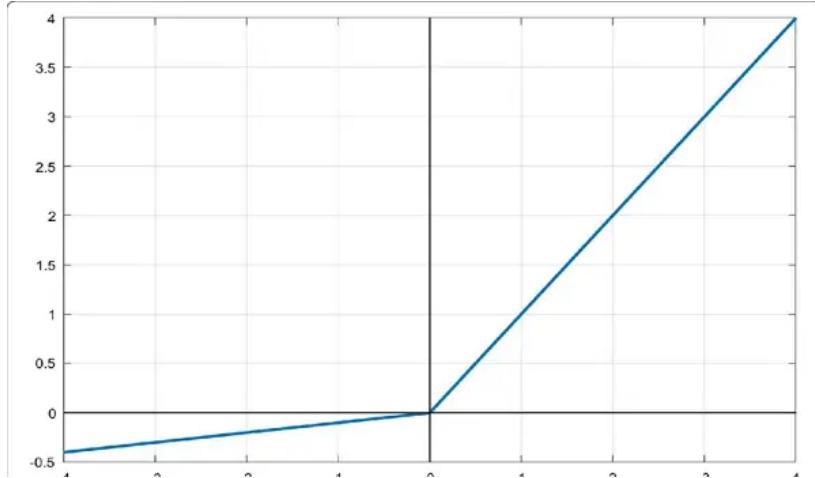


Figura 13: La funzione Leaky ReLU

#### 3.4.4 Funzioni di perdita

La funzione di perdita o la funzione di costo è sfruttata per calcolare la distanza fra il valore previsto ed il valore reale. La funzione di perdita è usata solitamente come criterio imparante del problema di ottimizzazione. La funzione di perdita può essere utilizzata con CNN per affrontare i problemi di regressione e i problemi di classificazione, il cui obiettivo è quello di minimizzare la funzione stessa. Le funzioni di perdita comunemente usate includono l'errore assoluto medio (MAE), l'errore quadratico medio (MSE), l'entropia trasversale e così via. Nelle CNN, ci sono molte funzioni di perdita da gestire quando si tratta di un task di classificazione. La più conosciuta ed usata è la Cross-Entropy, che è usata per valutare la differenza tra la distribuzione di probabilità ottenuta dall'allenamento corrente e la distribuzione effettiva. Questa funzione confronta la probabilità prevista con il valore di uscita effettivo (0 o 1) in ogni classe e calcola il valore di penalità in base alla distanza da loro. La penalità è logaritmica, pertanto, la funzione fornisce una penalità minore (0.1 o 0.2) per differenze più piccole tra le due distribuzioni e una penalità più grande (0.9 o 1.0) per differenze più grandi. La Cross-Entropy loss function è anche chiamata perdita di Softmax, poiché viene usata nelle reti convoluzionali in combinazione con un layer Softmax. Ad esempio, AlexNet [26], Inception v1 [7] e Resnet [6] utilizzano la Cross-Entropy loss function come funzione di perdita nei loro articoli originali e ciò li ha aiutati a raggiungere risultati all'avanguardia. Tuttavia, la Cross-Entropy loss function ha alcuni difetti: essa si preoccupa solo della correttezza della classificazione ma non del grado di compattezza all'interno della stessa classe o del margine tra classi diverse. Una delle varianti della Cross-Entropy è la Large-Margin Softmax loss [36]. Lo scopo di questa funzione è di massimizzare la densità interna tra campioni di una stessa classe e massimizzare la distanza tra classi diverse. La Large-Margin Softmax loss aggiunge un margine tra classi diverse e introduce la regolarità di tale margine attraverso l'angolo della matrice del peso di vincolo. Tale funzione è stata utilizzata per face recognition [36], emotion recognition [37], speaker verification [38], e così via.

#### 3.4.5 Ottimizzatori

Nelle CNN, spesso abbiamo bisogno di ottimizzare le funzioni non convesse. I metodi matematici richiedono un'enorme potenza di calcolo, pertanto, gli ottimizzatori vengono utilizzati nel processo di allenamento per ridurre al minimo la funzione di perdita e per ottenere parametri di rete ottimali in un tempo accettabile. Gli algoritmi di ottimizzazione comuni includono momentum, Root-mean-square prop (RMSprop), stima del momento adattivo (Adam), e così via. Ciascuno dei precedenti

algoritmi si basa sulla discesa del gradiente e ne esistono tre tipi che si possono utilizzare per allenare i modelli di CNN:

1. Batch Gradient Descent (BGD)
2. Stochastic Gradient Descent (SGD)
3. mini-BGD (MBGD)

Il BGD indica che un intero lotto di dati deve essere calcolato per ottenere un gradiente per ogni aggiornamento al fine di garantire la convergenza all'ottimo globale del piano convesso e all'ottimo locale del piano non convesso. Tuttavia, BGD è troppo lento da usare perché deve essere calcolato il gradiente medio dei campioni interi del lotto. Inoltre, può essere difficile per i dati che non sono adatti per il calcolo in memoria. BGD, quindi, è difficilmente utilizzato nella pratica nella formazione di modelli basati su CNN. Al contrario, SGD utilizza un solo campione per ogni aggiornamento. È evidente che il tempo di SGD per ogni aggiornamento è notevolmente inferiore rispetto a BGD perché soltanto un gradiente del campione totale viene usato per il calcolo. In questo caso, SGD è adatto per l'apprendimento online [39]. Tuttavia, SGD è rapidamente aggiornato con varianza elevata e ciò causa una forte oscillazione nella funzione di perdita. Da un lato, l'oscillazione del calcolo può far saltare il calcolo del gradiente dall'ottimo locale e infine raggiungere un punto migliore. D'altra parte, SGD non può mai convergere a causa dell'oscillazione infinita. In termini di tasso di convergenza, supponendo che la deviazione standard di ciascun campione per la distribuzione reale sia  $\sigma$ , la deviazione standard di  $n$  campioni è  $\frac{\sigma}{\sqrt{n-1}}$ . Quando usiamo i campioni per stimare il gradiente, un campione introduce una deviazione standard pari a  $\sigma$ , ma usandone  $n$  ciò non fa diminuire linearmente la deviazione standard. L'unica cosa che aumenta è la quantità di calcolo sugli  $n$  campioni e l'aumento è lineare. Sulla premessa di questa stessa quantità di calcolo, la velocità di convergenza dell'utilizzo dell'intero set di campioni è molto più lenta rispetto all'utilizzo di un piccolo numero di campioni. In altre parole, per convergere nello stesso punto ottimale, quando si utilizza l'intero set di allenamento, anche se il numero di iterazioni è piccolo, il tempo di ogni iterazione è lungo. Quindi, il costo totale di tempo è maggiore rispetto ad un piccolo numero di campioni per più iterazioni multiple.

Teoricamente, quanto affermato prima è vero, ossia minore è il numero di campioni, più veloce sarà il tasso di convergenza quando si utilizza una CPU single-core. Tuttavia, quando si utilizzano le GPU per l'addestramento, a causa del gran numero di core ed all'eccellente capacità di calcolo parallelo della GPU, ci vuole lo stesso tempo sia per calcolare un campione sia per calcolarne decine o centinaia. Pertanto, nella pratica ingegneristica basata su BGD e SGD, MBGD è stato proposto e utilizzato frequentemente. Esso combina i vantaggi di BGD e SGD e la dimensione del batch dipende dalla memoria della GPU e dal core di calcolo. MBGD utilizza un piccolo lotto di campioni per ogni aggiornamento in modo che possa non solo eseguire la discesa del gradiente in modo più efficiente di BGD, ma anche ridurre la varianza rendendo la convergenza più stabile.

Molti modelli classici della CNN utilizzano MBDG per addestrare le loro reti, come AlexNet [26], VGG [40], inception v2 [7], Resnet [26] e DenseNet [41]. È stata inoltre utilizzata in FaceNet [42], DeepID [43] e DeepID2 [44]. Sulla base di MBGD, sono stati proposti una serie di algoritmi efficaci per l'ottimizzazione per accelerare il processo di formazione del modello. Qui di seguito mostriamo alcuni ottimizzatori che si basano MBDG:

- L'algoritmo di Adagrad [45] è un altro algoritmo di ottimizzazione basato sui gradienti. Può adattare dinamicamente il learning rate ai parametri delle reti, cioè esegue piccoli aggiornamenti (se ad esempio, sussiste un basso learning rate) per caratteristiche che si mostrano con alta frequenza ed esegue aggiornamenti più ampi (cioè, con un alto learning rate) per quelle non frequenti. Pertanto, è adatto per l'elaborazione di dati sparsi. Uno dei principali vantaggi di Adagrad è che non è necessario regolare manualmente il tasso di apprendimento. Nella maggior parte dei casi, si usa solo 0.01 come learning rate predefinito [46]. Ad esempio FaceNet [42] utilizza Adagrad come ottimizzatore in allenamento.

- L'algoritmo di Adadelta [47], che è un'estensione dell'Adagrad, è progettato per ridurre il suo learning rate come una funzione monotona decrescente. Non si limita ad accumulare tutti i gradienti al quadrato, ma imposta una finestra di dimensioni fisse per limitare il numero di gradienti al quadrato accumulati. Allo stesso tempo, la somma dei gradienti è definita ricorsivamente come la media di decadimento di tutti i gradienti precedentemente quadrati, piuttosto che memorizzare direttamente i gradienti precedentemente quadrati. L'algoritmo Adadelta è sfruttato in molti compiti come speech recognition e sentence classification [48, 49].
- L'algoritmo RMSprop [50] è progettato per risolvere il problema del tasso di apprendimento radicalmente decrescente nell'algoritmo Adagrad. MobileNet [51], versione iniziale v3 [52] e versione iniziale v4 [53] hanno ottenuto i migliori modelli utilizzando RMSprop.
- Un altro ottimizzatore utilizzato di frequente è la stima adattiva del momento (Adam) [54]. È essenzialmente un algoritmo formato combinando la quantità di moto con l'RMSprop. Adam memorizza sia la media di decadimento esponenziale dei gradienti quadrati passati, come l'algoritmo Adadelta, sia la media di decadimento esponenziale medio dei gradienti passati, come l'algoritmo di quantità di moto. La pratica ha dimostrato che l'algoritmo Adam funziona bene su molti problemi ed è applicabile a molte strutture diverse di CNN [55, 56].

L'algoritmo AdaMax [54] è una variante di Adam che semplifica l'intervallo di confine del tasso di apprendimento ed è stato utilizzato per addestrare i modelli CNN di [56] e [57].

La stima di Adam accelerata da Nesterov (Nadam) [58] è una combinazione di Adam e NAG. Nadam ha un vincolo più forte sul tasso di apprendimento e un impatto diretto sull'aggiornamento del gradiente. Nadam è utilizzato in molti compiti come la classificazione del livello del suono in un luogo [59, 60].

### 3.4.6 Hyperparameter tuning

Quando si costruisce una CNN, oltre a selezionare la funzione di attivazione, la funzione di perdita e l'ottimizzatore, dobbiamo anche regolare molti altri iperparametri che influiscono notevolmente sulle prestazioni del modello. Come è noto, non esiste un insieme fisso di iperparametri in grado di garantire una soluzione ottimale per tutto il tempo. Per cui, un insieme di esperienze e buone regole pratiche sono significative durante l'hyperparameter tuning. Un modello di deep neural network (DNN) apprende i valori appropriati delle sue variabili di configurazione, cioè i pesi di connessione e il bias, dai dati di allenamento e, tali variabili, sono chiamate parametri [61]. Il processo di determinazione dei loro valori dai dati è noto come formazione o apprendimento del modello. Tuttavia, ci sono alcuni parametri di alto livello noti come "iperparametri" i cui valori non possono essere appresi dai dati [62]. Alcuni degli iperparametri importanti per le architetture DNN includono il numero di layer, il numero di nodi, il tasso di apprendimento, il tasso di regolarizzazione, la funzione di perdita, la funzione di attivazione, il metodo di ottimizzazione, il metodo di valutazione, ecc. L'addestramento di una DNN è un processo relativamente ingombrante poiché queste ultime devono essere addestrate con una grande quantità di dati per imparare i loro parametri con precisione [63, 64]. Anche una singola istanza del problema di formazione della rete neurale potrebbe richiedere giorni o addirittura settimane per allenarsi. Ciò influenza anche le reti neurali convoluzionali (CNN) distribuite per riconoscere le immagini e video [65]. Quando la CNN tenta di riconfigurare i suoi parametri dopo l'introduzione di piccoli rumori, si rischia l'over-fitting e il rallentamento della convergenza del modello [66]. Allo stesso modo, l'addestramento delle reti neurali ricorrenti (RNN) è un compito difficile poiché hanno un'architettura complessa e soffrono di problemi di gradienti che svaniscono [67]. Le prestazioni di formazione delle DNN e la loro struttura dipendono fortemente dalle scelte dei valori degli iperparametri [68]. Perciò, decidere i valori ottimali degli iperparametri è essenziale per esprimere il massimo potenziale dei modelli DNN [66]. La Tabella 2 a pagina 20 mostra come alcuni iperparametri comuni influenzano le prestazioni dei modelli.

Tabella 2: Iperparametri più comuni

Iperparametri	Descrizione
<b>Learning rates</b>	Il tasso di apprendimento si riferisce alla dimensione del passo di aggiornamento dei pesi di rete. Può essere costante o variabile. Come accennato nella sottosezione Ottimizzatori 3.4.5, gli algoritmi di ottimizzazione determinano diversi tassi di apprendimento. Per rendere la discesa in pendenza migliore, il valore della velocità di apprendimento dovrebbe essere impostato in un intervallo appropriato. Se la velocità di apprendimento è troppo piccola, la velocità di convergenza sarà lenta; se è troppo grande, i parametri oscilleranno avanti e indietro su entrambi i lati della soluzione ottimale.
<b>Epoch</b>	L'epoca si riferisce al numero di volte che tutto l'insieme di addestramento è dato in input alla rete neurale per il training. Quando il divario di accuratezza fra l'insieme di addestramento e l'insieme di convalida è piccolo, l'epoca corrente è considerata appropriata. Altrimenti, se il divario diminuisce, significa che l'epoca è troppo piccola, con conseguente under-fitting; se il divario aumenta, l'epoca è troppo grande, con conseguente over-fitting.
<b>Mini-batch size</b>	La dimensione del lotto mini è il numero di campioni inviati al modello in ogni allenamento. Nel processo di ottimizzazione della rete, avere la dimensione del batch troppo piccola significa che il numero di campioni immessi nella rete è troppo piccolo, cioè non rappresentativo, e il rumore aumenta di conseguenza, il che rende la convergenza della rete molto difficoltosa. La dimensione del lotto troppo grande rende la direzione del gradiente quasi stabile e ciò rende la discesa del gradiente rapida verso un punto ottimale locale o sella locale.
<b>Number of conv layers e conv kernels</b>	Ogni livello Conv di solito contiene caratteristiche di livello diverso. Gli strati superficiali possono rilevare le caratteristiche dei bordi, le caratteristiche locali e altre caratteristiche a basso livello dell'immagine, mentre gli strati profondi possono rilevare le caratteristiche globali. Di solito, le reti con più livelli e kernel hanno la capacità di rappresentare caratteristiche più complesse, ma nel frattempo, sono più difficili da addestrare.
<b>Size of conv kernels</b>	Sulla premessa dello stesso campo ricettivo, più piccolo è il kernel di convoluzione, meno sono i parametri e meno complessità computazionale è richiesta. In particolare, quando la dimensione del kernel di convoluzione è maggiore di 1, può aumentare il campo ricettivo. Se si usa il kernel di convolutione con dimensione pari, anche se il padding viene aggiunto simmetricamente, non possono essere mantenute invariate la dimensione della mappa delle caratteristiche di input e la dimensione della mappa delle caratteristiche di output .

### 3.5 Generazione delle Heatmap

Le heatmap sono generate sulla base dell'idea di Guided Grad-CAM [2]. Come premessa bisogna usare solo il dataset di training sulla rete CNN già addestrata, in seguito, per ogni campione di dati, si è riportato la mappa di attivazione dell'ultimo layer convoluzionale durante il processo di forward passing. Poi, attraverso il processo di backpropagation, abbiamo registrato i gradienti specifici dell'etichetta di ogni neurone dell'ultimo layer convoluzionale. Questi gradienti rappresentano il contributo dei neuroni ai risultati della classificazione. La Grad-CAM viene calcolata utilizzando una somma ponderata delle mappe di attivazione che viene poi ridimensionata alle dimensioni dell'input. Per migliorare ulteriormente l'accuratezza della localizzazione, i gradienti del layer di input sono moltiplicati con la Grad-CAM, generando infine la Guided Grad-CAM per ciascun campione, che può essere visualizzata come una heatmap. Successivamente, è stata calcolata la media di tutte le heatmap della stessa classe e, dopo la normalizzazione, sono state ottenute le heatmap per ogni classe. L'intensità di ciascun pixel rappresenta il significance score del gene che corrisponde al contributo dato da quest'ultimo alla classificazione. In realtà, questo processo di generazione delle heatmap potrebbe essere impiantato nell'addestramento della rete neurale convoluzionale poiché non influisce sul processo di addestramento e di test. La computazione è, però, minore con il metodo a due fasi descritto precedentemente, poiché la Grad-CAM richiede che tutti i campioni passino una sola volta attraverso la rete neurale.

#### 3.5.1 Gradient Classification Activation Map

Un certo numero di opere precedenti hanno affermato che rappresentazioni più profonde in una CNN catturano costrutti visivi di livello superiore [69, 70]. Inoltre, le caratteristiche convoluzionali mantengono naturalmente le informazioni spaziali che vengono perse nei layer fully connected, quindi possiamo aspettarci che gli ultimi layer convoluzionali abbiano il miglior compromesso tra semantica di alto livello e informazioni spaziali dettagliate. I neuroni in questi layer cercano nell'immagine informazioni semantiche specifiche della classe (ad esempio, parti di oggetti). Grad-CAM usa le informazioni di gradiente che fluiscono nell'ultimo strato convoluzionale della CNN per capire l'importanza di ogni neurone per una decisione di interesse. Anche se la nostra tecnica è molto generica e può essere utilizzata per visualizzare qualsiasi attivazione in una rete profonda, in questo lavoro ci concentriamo sulla spiegazione delle decisioni che la rete può prendere eventualmente.

Come mostrato in Figura 14 a pagina 14, al fine di ottenere la mappa di localizzazione classe-discriminante Grad-CAM del layer  $L$  definita come  $L_{\text{Grad-CAM}}^c \in \mathbb{R}^{u \times v}$  di larghezza  $u$  e altezza  $v$  per qualsiasi classe  $c$ , in primo luogo calcoliamo il gradiente del punteggio per la classe  $c$ ,  $y^c$  (prima del Softmax), rispetto alla mappa delle caratteristiche  $A^k$  di uno strato convoluzionale, cioè  $\frac{\partial y^c}{\partial A^k}$ . Questi gradienti che scorrono indietro sono raggruppati nella media globale per ottenere i pesi  $\alpha_k^c$  dell'importanza dei neuroni:

$$\alpha_k^c = \underbrace{\frac{1}{Z} \sum_i \sum_j}_{\text{global average pooling}} \underbrace{\frac{\partial y^c}{\partial A_{ij}^k}}_{\text{gradients via backprop}} \quad (1)$$

Questo peso  $\alpha_k^c$  rappresenta una linearizzazione parziale della rete profonda a valle di  $A$ , e cattura l'importanza della mappa caratteristica  $k$  per una classe target  $c$ .

Eseguiamo una combinazione ponderata di mappe di attivazione in avanti e applichiamo ReLU per ottenere:

$$L_{\text{Grad-CAM}}^c = \text{ReLU} \left( \underbrace{\sum_k \alpha_k^c A^k}_{\text{linear combination}} \right) \quad (2)$$

Si noti che ciò comporta una heatmap grossolana delle stesse dimensioni delle mappe delle caratteristiche convoluzionali ( $14 \times 14$  nel caso degli ultimi strati convoluzionali della rete VGG [40] e AlexNet [26]). Gli studi di ablazione e più visualizzazioni di Grad-CAM possono essere trovati in [71]. In generale,  $y^c$  non deve essere il punteggio di classe prodotto da una classificazione di immagine tramite CNN. Potrebbe essere un'attivazione differenziabile che include parole da una didascalia o la risposta a una domanda.

### 3.5.2 Guided BackPropagation

Parte del potere d'interpretazione visiva di Guided Grad-CAM è ottenuta utilizzando l'algoritmo di Guided Backpropagation basato sul gradiente (GBP) [72], che si concentra sui pixel rilevanti delle immagini, responsabili della previsione della tipologia tumorale, fornendo un punteggio di confidenza a ciascun pixel. L'algoritmo GBP esegue una propagazione in avanti attraverso il modello in cui l'input viene passato attraverso la funzione di attivazione ReLU, come descritto nell'Equazione 3. Qui l'input  $z_i^l$  rappresenta il punteggio di attribuzione per ogni timestamp. In questo caso, l'ingresso  $z_i^l$  rappresenta l'uscita dell' $i^{th}$  neurone dell' $l^{th}$  strato. Durante la propagazione all'indietro, le saliency map ottenute con il filtro di convoluzione passano solo i valori che sono stati positivi durante il passaggio in avanti e tagliano il resto dei valori. Ciò è visibile nell'Equazione 4. Il segnale di errore  $E_i^{l+1}$  rappresenta l'errore proveniente dall' $i^{th}$  neurone dello strato  $(l+1)^{th}$ . Il GBP modifica la propagazione all'indietro classica, aggiungendo un'ulteriore condizione: passano solo i valori che sono stati positivi durante la propagazione in avanti e all'indietro. Dunque, dato questo vincolo viene generato un nuovo segnale di errore che tiene conto sia di  $z_i^l$  sia di  $E_i^{l+1}$  prima di passare l'errore al livello precedente, come descritto nell'Equazione 5. Pertanto, il gradiente è guidato dall'input dello strato precedente e dal segnale di errore dello strato successivo. I gradienti retropropagati evidenziano solo i pixel che influenzano fortemente la diagnosi sulla tipologia di coorte tumorale, mentre la retropropagazione convenzionale non maschera le voci negative durante la propagazione all'indietro. Il GBP calcola la versione vincolata del gradiente rispetto all'input, mantenendo costante la matrice dei pesi della rete  $\theta$  e utilizzando ReLU. Essendo una tecnica interpretabile a posteriori, GBP non influenza la capacità decisionale del modello.

$$z_i^{l+1} = \text{ReLU}(z_i^l) = \max(z_i^l, 0) \quad (3)$$

$$E_i^l = E_i^{l+1} \forall (z_i^l > 0), \text{ where } E_i^{l+1} = \frac{\delta z^{out}}{\delta z_i^{l+1}} \quad (4)$$

$$E_i^l = E_i^{l+1} \forall (z_i^l > 0) \text{ and } (E_i^{l+1} > 0) \quad (5)$$

### 3.5.3 Guided GradCam

Mentre le visualizzazioni Grad-CAM sono class-discriminative e localizzano bene le regioni più rilevanti in un'immagine, esse non hanno la capacità di mostrare l'importanza a grana fine come i metodi di visualizzazione gradient pixel-space transformation (ad es. guided backpropagation e deconvolution). Per esempio, in Figura 14 a pagina 23, si può notare come nell'immagine indicata con **Grad-CAM**, vengano facilmente localizzate le regioni rilevanti del gatto; tuttavia, non è chiaro dalle basse risoluzioni della heatmap perché la rete prevede questa particolare istanza come "gatto tigre". Al fine di combinare gli aspetti migliori di entrambi, si fondono entrambi questi metodi di visualizzazione, Guided Backpropagation e Grad-CAM, tramite moltiplicazione puntiforme ( $L_{Grad-CAM}^c$  prima di essere moltiplicata con i gradienti generati da GBP viene ridimensionata alla risoluzione dell'immagine di ingresso utilizzando interpolazione bi-lineare). Nella Figura 14 si può notare nell'immagine etichettata come **Guieded Grad-CAM** come la heatmap generata con il procedimento descritto in

precedenza sia ad alta risoluzione e classe discriminante (ossia quando la classe di interesse è "gatto tigre", oltre che identificare importanti caratteristiche del "gatto tigre", come strisce, orecchie a punta e gli occhi mostra anche la corretta tipologia di animale, cioè "gatto tigre" e non un'altra classe). La sostituzione di Guided Backpropagation con Deconvolution, in base a quanto descritto sopra, dà risultati simili. Di fatti in [1] hanno constato, però, che Deconvolution aveva degli artefatti: un vincolo sull'architettura della rete presa in analisi e che le visualizzazioni erano rumorose rispetto a Guided Backpropagation che erano generalmente meno rumorose. Per tali motivi, quindi, viene scelta come tecnica di gradient pixel-space transformation, Guided Backpropagation al posto di Deconvolution.

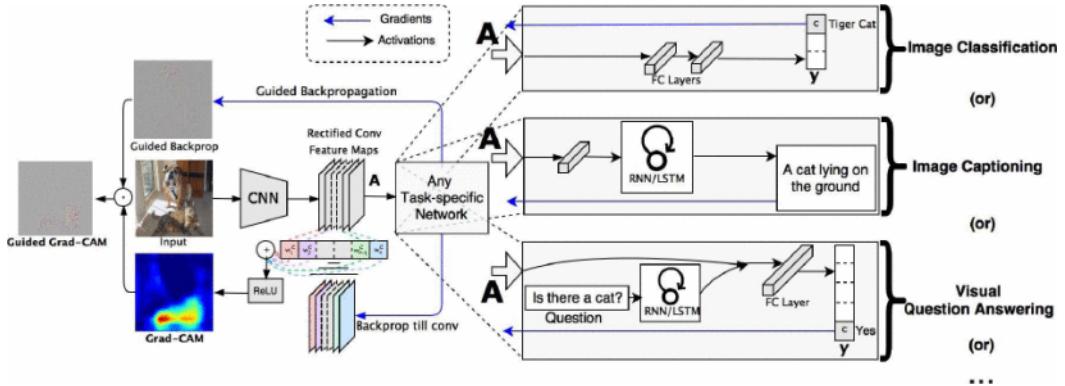


Figura 14: Overview di Guided Grad CAM [2]

### 3.6 Validazione

I geni principali sono stati selezionati in base alla classifica dei significance score nelle heatmap. Abbiamo applicato l'analisi funzionale a questi top gene per dimostrare ulteriormente che i geni sono specifici per il tumore e che sono potenziali biomarker. Nella prima fase, abbiamo scelto i primi 400 geni più importanti di ciascun tipo di tumore per effettuare la pathway analysis (analisi dei percorsi biologici), cercando di scoprire se i percorsi significativamente arricchiti (significantly enriched pathway) sono correlati al tumore corrispondente.

#### 3.6.1 Enrichment Pathway Analysis

La Pathway Analysis (PA), nota anche come analisi di arricchimento funzionale, sta rapidamente diventando uno degli strumenti più importanti della ricerca Omics<sup>5</sup>. Lo scopo principale degli strumenti di PA è di analizzare i dati ottenuti dalle tecnologie ad alta-capacità di lavorazione, individuare i gruppi relativi ai geni legati alle alterazioni nei campioni di caso rispetto ad un campione di controllo. In questo modo, i metodi di PA cercano di superare il problema della dimensionalità introdotta dal dover interpretare grandi liste contenenti geni importanti, ma isolati dal contesto biologico, che sono l'output principale dell'analisi dei dati ad alta-capacità di lavorazione prodotti dalla maggior parte delle tecniche biologiche che si basano sull'analisi funzionale, come l'analisi di espressione differenziale. I metodi PA invece forniscono significato ai dati biologici sperimentali ad alto rendimento (High Throughput Biological Data, in breve HTBD) facilitando così l'interpretazione e la successiva generazione di ipotesi. Ciò è stato ottenuto sulla base dell'accoppiamento delle conoscenze biologiche esistenti provenienti da banche dati con test statistici, analisi matematiche e algoritmi computazionali. I metodi di PA possiedono una vasta gamma di applicazioni nella ricerca fisiologica e biomedica. Questi metodi mirano ad aiutare il ricercatore a scoprire quali temi biologici, e quali biomolecole, sono cruciali per comprendere i fenomeni in studio, dato gli HTBD analizzati. A sua volta, gli indizi che

<sup>5</sup>Con il termine OMICS si intendono tutte quelle scienze biologiche il cui nome termina con il suffisso -omics, ad esempio genomics, transcriptomics, proteomics, metabolomics.

fornisce una PA consentono al ricercatore di generare nuove ipotesi, progettare esperimenti successivi e convalidare ulteriormente i loro risultati. I metodi di PA hanno aiutato i ricercatori nell'identificazione dei ruoli biologici dei geni candidati, selezionati per progettare nuove terapie per il cancro, aggirando i danni collaterali alle cellule sane [73]. Ci sono diversi elementi necessari per eseguire una PA. Prima di tutto, sono necessari dati quantitativi rappresentativi della biologia cellulare che sono generati con l'uso delle tecnologie di Omics come: RNA-microarrays, tandem mass spectrometry e RNA-sequencing. In secondo luogo, un approccio in grado di analizzare una tale quantità sostanziale di dati è obbligatorio. La biologia dei sistemi è un campo di ricerca emergente che consente lo studio degli organismi viventi come sistemi, opponendosi ad approcci riduzionisti [74, 75], utilizzando i dati Omics come input principale delle sue analisi. In terzo luogo, le conoscenze biologiche molecolari memorizzate in basi di dati sono necessarie per l'analisi da eseguire, guidando i metodi PA per cercare le relazioni tra i dati Omics generati e temi biologici noti. Infine, il potere computazionale necessario per realizzare PA consiste principalmente nella computazione necessaria ad eseguire il test statistico dei temi biologici vs. i dati, e alla complessità degli altri algoritmi matematici che cercano di estrarre le relazioni tra i dati e le conoscenze precedenti. Un altro esempio è la determinazione della somiglianza e della diversità, a livello molecolare, tra gruppi di campioni, come nel confronto tra linee cellulari e campioni tumorali [76]. Questo tipo di analisi può aiutare i ricercatori a comprendere i fenomeni di eterogeneità in diversi contesti di ricerca. Un altro esempio ancora è l'uso di metodi PA per esaminare la funzione biologica dei moduli genici. I ricercatori hanno dovuto convalidare insiemi di geni pensati per essere correlati tra loro, come nell'analisi dei geni che fluttuano in risposta alle variazioni naturali, come le stagioni [77]. Anche se tutte queste applicazioni hanno avuto successo in obiettivi specifici, l'uso di metodi di PA può essere ampia e complessa come la creatività del loro utente. Gli sforzi nella strutturazione della conoscenza biologica sulle pathway hanno provocato la generazione delle basi di dati delle pathway (PDBs), anche denominate "basi di conoscenza," che condensano la conoscenza biologica corrente delle interazioni molecolari nelle raccolte di dati delle pathway. I PDBs di solito recuperano e strutturano dati da fonti diverse. Generalmente, le prove sperimentali sono curate dalla letteratura e le analisi computazionali sono effettuate dal progetto stesso per inferire le funzioni possibili delle biomolecole omologhe. Inoltre, viene generalmente fatto un riferimento incrociato dei dati tra database simili. Ad esempio, le annotazioni del database Reactome [78] sono curate manualmente dalla letteratura da biologi esperti in collaborazione con il loro staff editoriale, e cross-referenced con diverse altre risorse, come letteratura primaria, e altri PDBs correlati [79]. Aggiunte e correzioni ai PDBs sono effettuate periodicamente, aumentando così la qualità e la copertura delle loro conoscenze biologiche. Alcuni database sono in grado di aggiornare le proprie informazioni in modo frequente, per mantenere il passo con le nuove scoperte. Ad esempio il database KEGG [80] aggiorna i suoi dati su base settimanale, ma altri PDBs lo fanno meno spesso, come Gene Ontology [81], che aggiorna i suoi dati su base mensile. Tuttavia, alcuni PDBs non riescono ad aggiornare le loro informazioni in modo regolare, quindi diventano obsoleti nel corso del tempo, eppure vengono utilizzati dagli utenti di strumenti di PA, quindi si suggerisce cautela nell'uso di dati PDB obsoleti. Quando si esegue la PA è necessario considerare la qualità dei dati nei PDBs e, come ulteriore fattore, è necessario considerare anche la copertura offerta dai PDBs. Si tratta della proporzione di componenti biologici aggregati descritti in tutti i percorsi di un PDB rispetto a un elenco di riferimento di componenti. Ad esempio, uno dei più completi PDB pubblici compositi, Pathway Commons [82], con le informazioni aggregate di 22 PDBs, ha attualmente una copertura di 17.439 simboli genici sui 39.241 accettati, pari a circa il 45% dei simboli totali dei registri ufficiali del Comitato per la nomenclatura del genoma HUGO<sup>6</sup>. L'utilizzo delle informazioni più aggiornate e complete è importante per l'estrazione ottimale di informazioni dai dati sperimentali. Raggiungere questo obiettivo non è solo un invito agli utenti a utilizzare i migliori PDB disponibili, ma è anche un invito ai potenziali contributori a migliorare la copertura e la conoscenza contenuta nei database. La maggior parte delle banche dati pubbliche incoraggia sforzi di collaborazione aperti, con la corrispondente revisione dei dati condivisi. Tutti questi miglioramenti

<sup>6</sup>Progetto approvato dal Comitato per la nomenclatura del genoma (HGNC)

nei PDBs porteranno, in conclusione, ad una più rapida scoperta della conoscenza e ad una più rapida applicazione delle conoscenze biologiche.

## 4 Risultati Sperimentali

Nella seguente sezione mostriamo i risultati ottenuti dalle nostre sperimentazioni e le comparazioni con il modello di riferimento in [1].

### 4.1 Valutazione Classificazione

Usando la 10-fold cross validation, è stata calcolata la media totale dell'accuracy, la media dell'accuracy per ogni classe, la media del precision score  $P$ , la media del recall score  $R$  e la media dell'f1 score  $F1$ .

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F1 = \frac{2PR}{P + R} \quad (1)$$

Nei paragrafi successivi mostriamo i risultati ottenuti per i test condotti.

#### 4.1.1 Caso Binario

Per il caso binario, è stato necessario modificare la soglia di varianza utilizzata nella fase di preprocesing al fine di estrarre un numero di feature (i geni) che fosse più o meno congruo a quello individuato da [1] in modo da non dover modificare la dimensione delle immagini e da ottenere in maniera più precisa possibile i potenziali biomarker. La soglia utilizzata in questo caso è stata di 0.9552 e ci ha permesso di avere 10381 geni. Le performance ottenute da questo test sono riportate in Tabella 3 a pagina 26 mentre l'accuracy divisa per coorte tumorale è riportata in Tabella 4 a pagina 26. Per completezza, abbiamo generato anche la matrice di confusione riportata in Figura 15 dalla quale si evince che non ci sono stati errori di classificazione.

Tabella 3: Performance del caso binario

Metodo	Accuracy	Precision	Recall	F1-score
CNN	100%	100%	100%	100%

Tabella 4: Risultati del caso binario per coorte tumorale

Classe tumorale	Coorte	Numero Campioni	Accuratezza ns. metodo	Accuratezza Riferimento
Lymphoid Neoplasm Diffuse Large B-cell Lymphoma	DLBC	48	1.00	1.00
Uterine Carcinosarcoma	UCS	57	1.00	0.81
<b>Totale campioni</b>		<b>105</b>		

#### 4.1.2 Caso Ternario

Come per il caso binario, anche per il caso ternario è stato necessario modificare la soglia di varianza utilizzata nella fase di preprocessing per l'estrazione delle feature (i geni). La soglia utilizzata in questo caso è stata di 0.9869 e ci ha permesso di avere 10382 geni. Le performance ottenute in questo test sono riportate in Tabella 5 a pagina 28 mentre l'accuracy divisa per coorte tumorale è riportata in Tabella 6 a pagina 28. Per completezza, abbiamo generato anche la matrice di confusione riportata in Figura 16 dalla quale si evince che BLCA è stata identificata al 97.56% correttamente e solo nel 2.44% è stata identificata come CESC; CESC è stata sempre identificata correttamente; LGG è stata identificata correttamente al 98.04% mentre è stata identificata all'1.96% come BLCA.

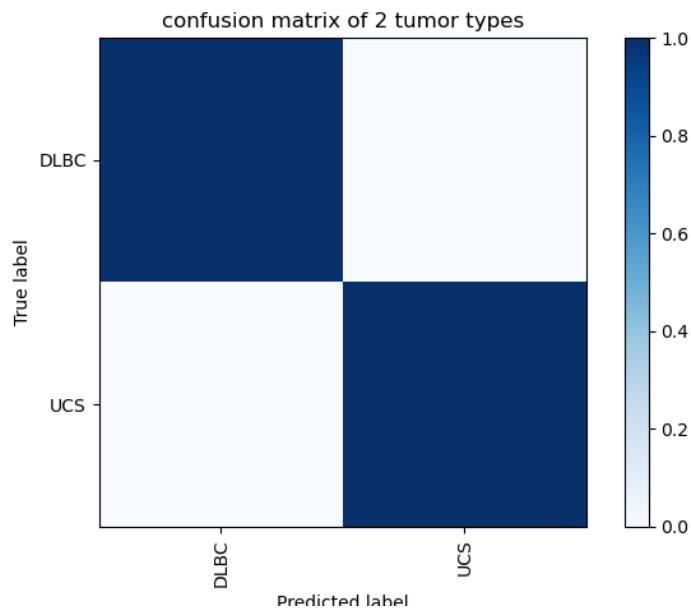


Figura 15: La matrice di confusione del caso binario

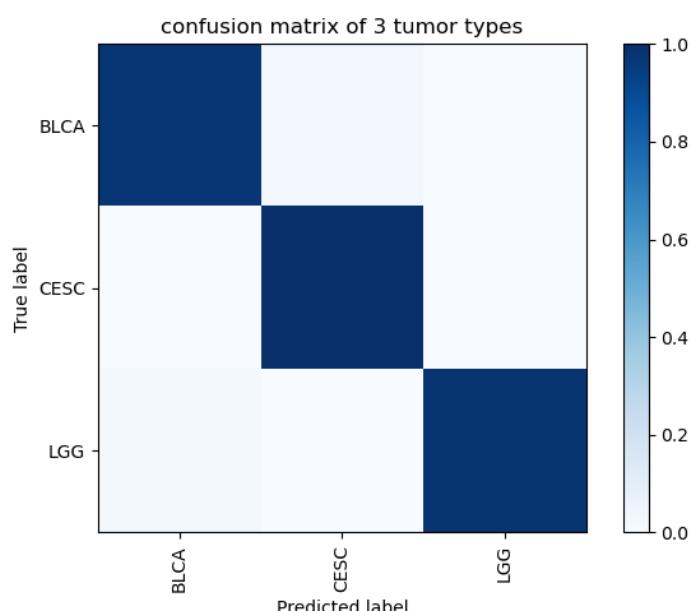


Figura 16: La matrice di confusione del caso ternario

Tabella 5: Performance del caso ternario

Metodo	Accuracy	Precision	Recall	F1-score
CNN	98.37%	98.45%	98.37%	98.37%

Tabella 6: Risultati del caso ternario per coorte tumorale

Classe tumorale	Coorte	Numero Campioni	Accuratezza ns. metodo	Accuratezza Variante	Accuratezza Riferimento
Bladder urothelial carcinoma	BLCA	408	0.98		0.97
Cervical and endocervical cancers	CESC	304	1.00		0.93
Brain Lower Grade Glioma	LGG	516	0.98		0.98
<b>Totale campioni</b>			<b>1.228</b>		

#### 4.1.3 Caso Generale

Le performance ottenute nel test generale usando la rete neurale di [1] sono riportati nella Tabella 7 a pagina 28 mentre una comparazione dell'accuracy per ogni classe tumorale è mostrata in Tabella 8 a pagina 29. Inoltre, abbiamo anche generato la matrice di confusione così come mostrato in Figura 17. Dalla matrice di confusione si può notare che la maggior parte delle classi sono classificate correttamente, tuttavia ci sono alcune classificazioni errate:

1. i campioni READ sono per lo più sono classificati come COAD e ciò potrebbe essere dovuto alla vicinanza delle locazioni spaziali dei due tumori;
2. alcuni campioni di CHOL sono classificati erroneamente come LIHC a causa dei pochi campioni presenti per la classe CHOL;
3. alcuni campioni di ESCA sono classificati erroneamente come STAD;
4. alcuni campioni di UCS sono classificati erroneamente come UCEC.

Delle possibili motivazioni per gli errori di classificazione (3) e (4) sono riportate nel paragrafo 5.2 in quanto si sono riscontrate anche durante l'utilizzo della rete VarNet.

Tabella 7: Performance del metodo utilizzato

Metodo	Accuracy	Precision	Recall	F1-score
CNN	95.79%	95.95%	95.79%	95.57%

Dal momento che il nostro metodo, al pari di [1] si prefissava di classificare i tumori e allo stesso tempo individuare i potenziali biomarker, sono stati mantenuti molti geni nella fase di preprocessing.

## 4.2 Heat-Map Generate

Nelle seguenti sottosezioni saranno mostrati degli esempi di heatmap generate per i vari casi di test effettuati. Le zone indicate da un contorno rosso nelle immagini indicano le zone con i pixel più luminosi, ossia i geni che hanno contribuito maggiormente alla classificazione.

Tabella 8: Accuracy per coorte tumorale (calcolo effettuato su GPU e su dataset con oversampling).

Coorte	Accuratezza ns. metodo	Accuratezza Variante	Accuratezza Riferimento
ACC	<b>1.00</b>	<b>1.00</b>	0.95
BLCA	<b>0.98</b>	<b>0.98</b>	0.97
BRCA	<b>1.00</b>	<b>1.00</b>	0.99
CESC	<b>0.97</b>	<b>0.97</b>	0.93
CHOL	<b>0.60</b>	<b>0.60</b>	0.56
COAD	<b>1.00</b>	<b>0.97</b>	0.95
DLBC	1.00	1.00	1.00
ESCA	<b>0.75</b>	<b>0.85</b>	0.77
GBM	<b>1.00</b>	<b>1.00</b>	0.94
HNSC	0.98	<b>1.00</b>	0.98
KICH	<b>0.67</b>	<b>0.44</b>	0.87
KIRC	<b>0.87</b>	<b>0.87</b>	0.95
KIRP	<b>0.94</b>	<b>0.94</b>	0.93
LAML	1.00	1.00	1.00
LGG	0.98	<b>1.00</b>	0.98
LIHC	<b>1.00</b>	<b>1.00</b>	0.97
LUAD	<b>0.91</b>	<b>0.97</b>	0.95
LUSC	<b>0.89</b>	<b>0.89</b>	0.91
MESO	<b>1.00</b>	<b>0.89</b>	0.94
OV	<b>0.97</b>	<b>1.00</b>	0.99
PAAD	<b>0.95</b>	<b>0.95</b>	0.97
PCPG	1.00	1.00	1.00
PRAD	1.00	1.00	1.00
READ	<b>0.09</b>	<b>0.09</b>	0.35
SARC	<b>0.92</b>	<b>0.85</b>	0.97
SKCM	<b>0.96</b>	<b>0.96</b>	0.98
STAD	<b>0.93</b>	<b>0.98</b>	0.96
TGCT	<b>1.00</b>	<b>1.00</b>	0.99
THCA	1.00	1.00	1.00
THYM	<b>1.00</b>	<b>0.92</b>	0.99
UCEC	<b>1.00</b>	<b>0.95</b>	0.96
UCS	<b>0.83</b>	<b>0.67</b>	0.81
UVM	<b>1.00</b>	<b>1.00</b>	0.99

#### 4.2.1 Caso Binario

Alcuni esempi di heatmap generate nel caso binario sono mostrate nella Figura 18 a pagina 30. Si può notare che esiste una certa similarità nelle immagini e in particolare, per DLBC, è possibile notare lo

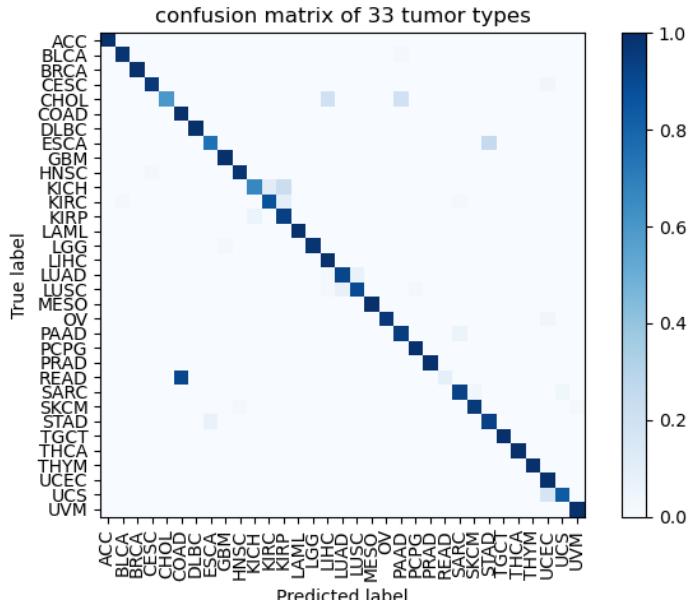


Figura 17: La matrice di confusione del caso generale

stesso pattern per le fold 2, 3, 4 e 9.

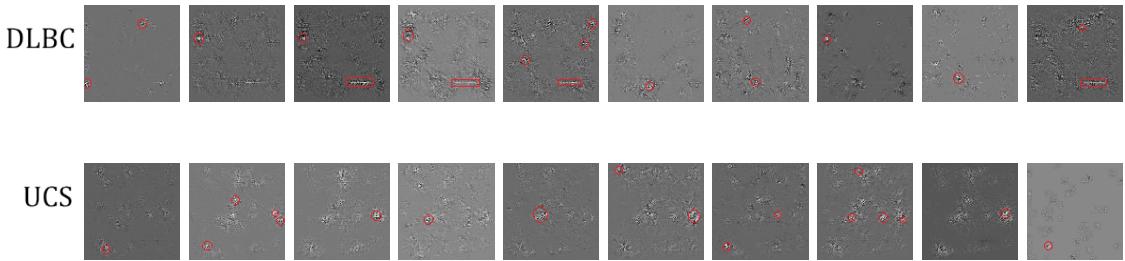


Figura 18: Heatmaps del caso binario

#### 4.2.2 Caso Ternario

Alcuni esempi di heatmap generate nel caso ternario sono mostrate nella Figura 19 a pagina 31. Anche in questo caso si può notare che esiste una certa similarità nelle immagini ma purtroppo non è presente nessun pattern evidente.

#### 4.2.3 Caso Generale

Le heatmap generate per ogni classe mostrano una similarità tra le 10 fold e mostrano un pattern distinto quando comparate tra classi. Alcuni esempi sono mostrati nella Figura 20 a pagina 31. Nell'esempio, ogni riga rappresenta le heatmap di diverse fold (da sinistra a destra sono dalla fold 1 alla fold 10). Anche se ci sono alcune differenze tra le diverse fold, esiste un pattern chiaro tra di esse.

### 4.3 Validazione dei percorsi biologici dei top genes

Come si può notare dalla Figura 21 a pagina 32, l'intensità dei primi 100 geni decresce rapidamente, mentre l'intensità dei geni successivi decresce in maniera uniforme. Tale comportamento è stato constatato anche in [1]. Inoltre, quando l'intensità è bassa, il tasso di decrescita cresce di nuovo. Se,

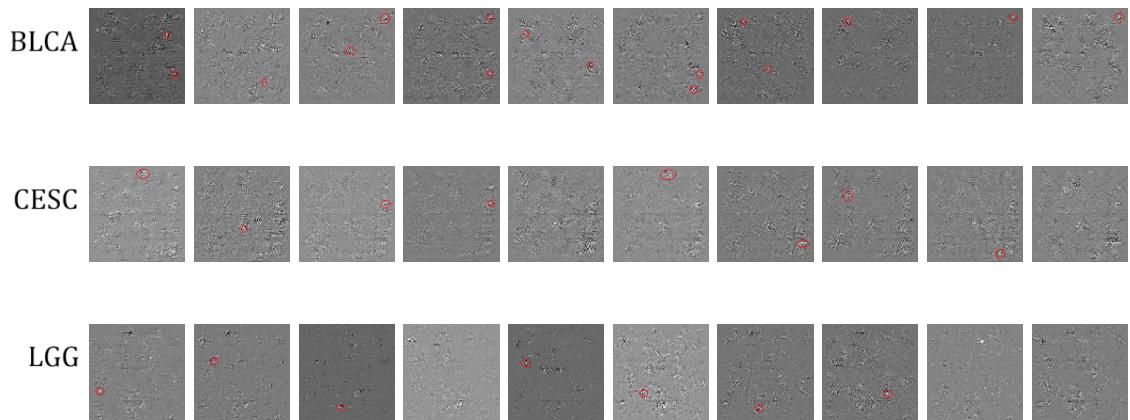


Figura 19: Heatmaps del caso ternario

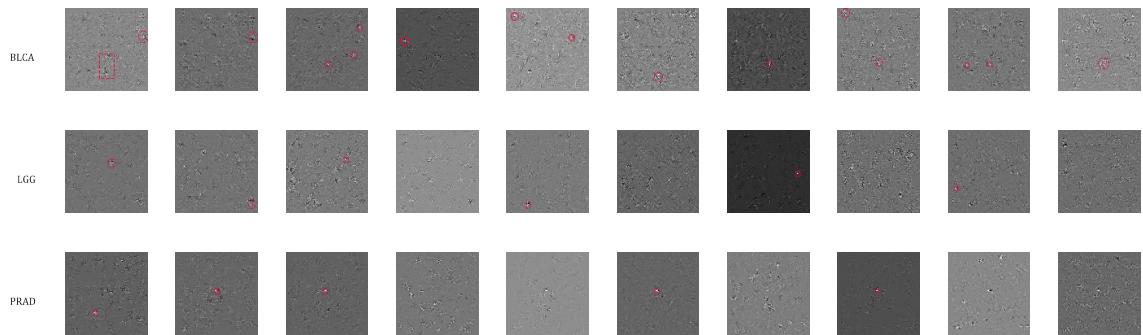


Figura 20: Alcuni esempi di heatmap. Ogni colonna rappresenta il risultato di una fold. Nella prima riga ci sono le heatmap del tipo di tumore BLCA, nella seconda di LGG e nella terza di PRAD.

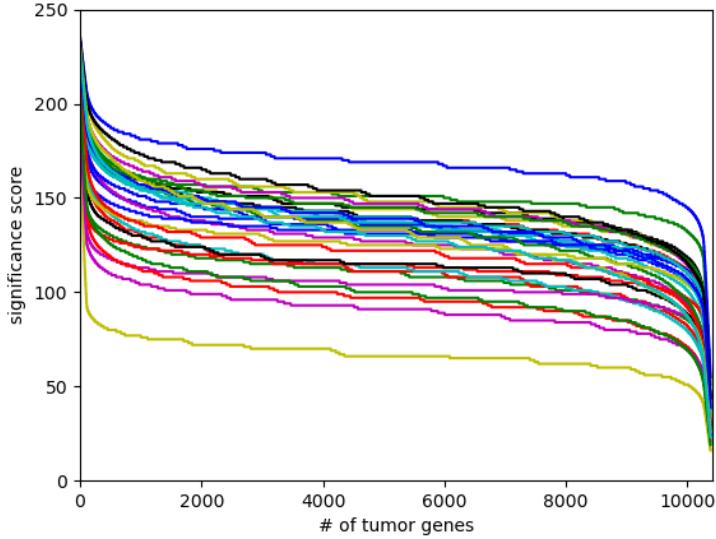


Figura 21: I cambi di intensità nelle heatmap per ogni classe. Si può notare come alcune classi condividano lo stesso pattern nei cambi di intensità.

come hanno fatto in [1], si assume che un'intensità più alta implica una *significance* più alta, dal momento che la curva delle intensità nei primi 400 geni cambia in maniera più evidente (ossia è più grande) rispetto alle altre migliaia di geni, allora è lecito scegliere i primi 400 geni come query per effettuare la pathway analysis. In aggiunta, facendo una comparazione con il numero totale di geni considerati (10381), la scelta di 400 geni è consistente col fatto che il numero di biomarker dovrebbe essere basso. Per effettuare la KEGG<sup>7</sup> pathway analysis è stato utilizzato il tool DAVID<sup>8</sup>. In seguito, basandoci sui risultati già ottenuti da [1] e facendo una sommaria attività di literature review, limitata alla nostra conoscenza attuale, abbiamo provato a trovare le relazioni che sussistono tra le pathway individuate e i tipi di tumore. Le pathway individuate, con un P-value inferiore a  $10^{-3}$ , per i casi di test effettuati, sono mostrate nelle successive sezioni. Nelle Tabelle che seguiranno si utilizzerà la seguente legenda: i valori di questo colore sono le nuove pathway rilevate dalle nostre analisi con il tool DAVID, i valori con questo background indicano le path che trovano riscontro anche nel lavoro di Lyu e Haque [1] e i valori con questo background indicano quelle pathway che sono specifiche della coorte tumorale.

#### 4.3.1 Caso Binario

Come si può notare dalla Tabella che segue, le pathway ottenute in questo caso binario, a differenza delle stesse classi tumorali riportate nella sezione 4.3, sono molte di più e, a differenza di quanto rilevato in [1] nel 2018, sono presenti anche malattie di origini recente (ad es. il Corona Virus). Per UCS inoltre, si può notare che sono presenti quattro pathway tumorali note: Wnt signaling pathway, PI3K-Akt signaling pathway, Hippo signaling pathway e TGF-beta signaling pathway, tre delle quali (Wnt, Hippo e TGF-beta) scompaiono nel caso generale descritto nella sezione 4.3.

Tali risultati sono ottenuti in quanto il numero di classi in esame è molto piccolo e dunque, al momento della classificazione, non viene sfruttata la conoscenza di ulteriori classi tumorali e in sede di generazione delle heatmap, e quindi di attribuzione del significance score ai geni (dal momento che si scopre quali geni hanno contribuito di più alla classificazione), il valore che viene assegnato ai geni

<sup>7</sup>Kyoto Encyclopedia of Genes and Genomes

<sup>8</sup><https://david.ncifcrf.gov/home.jsp>

è diverso e quindi ne consegue che anche la selezione dei top 400 effettuata in questo caso è diversa rispetto al caso generale.

Tabella 9: Risultati della Pathways Analysis sui primi 400 geni per le coorti DLBC e UCS ( $P < 10^{-3}$ )

Tumore	Pathway correlata	P value	
	ID	Nome	
DLBC	hsa05169	<a href="#">Epstein-Barr virus infection</a>	1.43e-16
	hsa05166	<a href="#">Human T-cell leukemia virus 1 infection</a>	1.01e-13
	hsa04659	<a href="#">Th17 cell differentiation</a>	1.28e-13
	hsa04658	<a href="#">Th1 and Th2 cell differentiation</a>	3.13e-13
	hsa04062	<a href="#">Chemokine signaling pathway</a>	3.10e-12
	hsa05330	<a href="#">Allograft rejection</a>	9.20e-12
	hsa04612	Antigen processing and presentation	5.07e-11
	hsa05140	Leishmaniasis	1.94e-10
	hsa04940	Type I diabetes mellitus	2.02e-10
	hsa05332	<b>Graft-versus-host disease</b>	9.73e-10
	hsa05145	Toxoplasmosis	1.02e-09
	hsa05416	Viral myocarditis	1.72e-09
	hsa04064	<b>NF-kappa B signaling pathway</b>	1.98e-09
	hsa05321	Inflammatory bowel disease	2.09e-09
	hsa04380	Osteoclast differentiation	3.82e-09
	hsa04668	TNF signaling pathway	4.10e-09
	hsa05340	Primary immunodeficiency	7.56e-09
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	4.56e-08
	hsa05200	Pathways in cancer	8.57e-08
	hsa05323	Rheumatoid arthritis	1.18e-07
	hsa05163	<a href="#">Human cytomegalovirus infection</a>	1.19e-07
	hsa05320	Autoimmune thyroid disease	1.26e-07
	hsa04672	Intestinal immune network for IgA production	1.57e-07
	hsa05150	Staphylococcus aureus infection	2.47e-07
	hsa05167	<a href="#">Kaposi sarcoma-associated herpesvirus infection</a>	3.78e-07
	hsa04514	<b>Cell adhesion molecules</b>	4.20e-07
	hsa04662	<b>B cell receptor signaling pathway</b>	4.29e-07
	hsa04060	Cytokine-cytokine receptor interaction	4.36e-07
	hsa05417	Lipid and atherosclerosis	4.78e-07
	hsa04640	Hematopoietic cell lineage	4.99e-07
	hsa05310	Asthma	7.56e-07
	hsa05152	Tuberculosis	8.38e-07
	hsa04145	Phagosome	1.48e-06
	hsa04933	<a href="#">AGE-RAGE signaling pathway in diabetic complications</a>	2.16e-06
	hsa05130	<a href="#">Pathogenic Escherichia coli infection</a>	3.86e-06
	hsa05170	<a href="#">Human immunodeficiency virus 1 infection</a>	4.80e-06
	hsa04210	Apoptosis	6.52e-06
	hsa05171	Coronavirus disease - COVID-19	1.05e-05
	hsa05162	Measles	1.06e-05
	hsa05205	<a href="#">Proteoglycans in cancer</a>	1.10e-05
	hsa04611	Platelet activation	1.98e-05
	hsa05418	<a href="#">Fluid shear stress and atherosclerosis</a>	2.83e-05

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04670	Leukocyte transendothelial migration	3.00e-05
	hsa05142	Chagas disease	3.17e-05
	hsa04926	Relaxin signaling pathway	4.36e-05
	hsa04015	Rap1 signaling pathway	4.60e-05
	hsa05164	Influenza A	6.03e-05
	hsa05215	Prostate cancer	1.12e-04
	hsa04625	C-type lectin receptor signaling pathway	1.27e-04
	hsa05202	Transcriptional misregulation in cancer	1.58e-04
	hsa05161	Hepatitis B	2.46e-04
	hsa04218	Cellular senescence	2.74e-04
	hsa04010	MAPK signaling pathway	2.82e-04
	hsa04620	Toll-like receptor signaling pathway	3.39e-04
	hsa04660	T cell receptor signaling pathway	3.39e-04
	hsa04610	Complement and coagulation cascades	4.06e-04
	hsa04928	Parathyroid hormone synthesis, secretion and action	4.52e-04
	hsa05132	Salmonella infection	5.33e-04
	hsa05133	Pertussis	6.18e-04
	hsa05219	Bladder cancer	6.71e-04
	hsa04151	PI3K-Akt signaling pathway	8.15e-04
	hsa05203	Viral carcinogenesis	9.87e-04
UCS	hsa04512	ECM-receptor interaction	1.14e-13
	hsa05165	Human papillomavirus infection	3.44e-12
	hsa04510	Focal adhesion	1.42e-11
	hsa05200	Pathways in cancer	3.14e-10
	hsa05205	Proteoglycans in cancer	9.14e-09
	hsa04360	Axon guidance	1.11e-08
	hsa04810	Regulation of actin cytoskeleton	2.26e-08
	hsa04310	Wnt signaling pathway	9.72e-08
	hsa04151	PI3K-Akt signaling pathway	1.24e-07
	hsa04974	Protein digestion and absorption	1.48e-07
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	1.61e-07
	hsa04514	Cell adhesion molecules	5.23e-06
	hsa04390	Hippo signaling pathway	1.36e-05
	hsa04520	Adherens junction	3.02e-05
	hsa04670	Leukocyte transendothelial migration	4.25e-05
	hsa05224	Breast cancer	5.20e-05
	hsa05217	Basal cell carcinoma	6.10e-05
	hsa05222	Small cell lung cancer	6.35e-05
	hsa04530	Tight junction	7.02e-05
	hsa05410	Hypertrophic cardiomyopathy	1.30e-04
	hsa04015	Rap1 signaling pathway	1.54e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	2.46e-04
	hsa04350	TGF-beta signaling pathway	3.00e-04
	hsa05166	Human T-cell leukemia virus 1 infection	5.11e-04
	hsa04928	Parathyroid hormone synthesis, secretion and action	5.99e-04
	hsa05225	Hepatocellular carcinoma	6.83e-04
	hsa04270	Vascular smooth muscle contraction	7.20e-04

Continua nella prossima pagina

Tumore	Pathway correlata	P value
	ID	Nome
	hsa04934	Cushing syndrome
	hsa05414	Dilated cardiomyopathy

#### 4.3.2 Caso Ternario

Anche per il caso ternario, così come accaduto per il binario, se si fa un confronto con quanto riportato nella sezione 4.3, è possibile notare che il numero di pathway rilevate per le stesse classi tumorali è maggiore. Come già indicato nel caso binario, tali risultati sono dovuti al fatto che sono prese in esame poche classi tumorali e quindi i significance score dei geni sono diversi rispetto al caso generale. Non è presente un grande differenza con i risultati del caso generale, per cui vale quanto detto nella sezione 4.3 per le stesse classi tumorali.

Tabella 10: Risultati della Pathways Analysis sui primi 400 geni per le coorti BLCA, CESC e LGG ( $P < 10^{-3}$ )

Tumore	Pathway correlata	P value
	ID	Nome
BLCA	hsa04512	ECM-receptor interaction
	hsa04510	Focal adhesion
	hsa05205	Proteoglycans in cancer
	hsa05204	Chemical carcinogenesis - DNA adducts
	hsa04514	Cell adhesion molecules
	hsa00980	Metabolism of xenobiotics by cytochrome P450
	hsa04145	Phagosome
	hsa00983	Drug metabolism - other enzymes
	hsa00982	Drug metabolism - cytochrome P450
	hsa00140	Steroid hormone biosynthesis
CESC	hsa05200	Pathways in cancer
	hsa04926	Relaxin signaling pathway
	hsa00040	Pentose and glucuronate interconversions
	hsa05165	Human papillomavirus infection
	hsa04360	Axon guidance
	hsa04933	AGE-RAGE signaling pathway in diabetic complications
	hsa04514	Cell adhesion molecules
	hsa04512	ECM-receptor interaction
	hsa05169	Epstein-Barr virus infection
	hsa04612	Antigen processing and presentation
	hsa04530	Tight junction
	hsa05200	Pathways in cancer
	hsa05330	Allograft rejection
	hsa04940	Type I diabetes mellitus

Continua nella prossima pagina

Tumore	Pathway correlata		P value
	ID	Nome	
	hsa05146	Amoebiasis	3.19e-05
	hsa05145	Toxoplasmosis	5.64e-05
	hsa05205	<b>Proteoglycans in cancer</b>	<b>6.76e-05</b>
	hsa04659	Th17 cell differentiation	8.51e-05
	hsa04145	Phagosome	1.22e-04
	hsa04658	Th1 and Th2 cell differentiation	1.56e-04
	hsa04640	Hematopoietic cell lineage	1.66e-04
	hsa05323	Rheumatoid arthritis	1.83e-04
	hsa00330	Arginine and proline metabolism	2.05e-04
	hsa05133	Pertussis	2.78e-04
	hsa05140	Leishmaniasis	3.31e-04
	hsa05418	Fluid shear stress and atherosclerosis	3.44e-04
	hsa04218	Cellular senescence	4.56e-04
	hsa04670	Leukocyte transendothelial migration	5.31e-04
	hsa04657	IL-17 signaling pathway	5.92e-04
	hsa04810	Regulation of actin cytoskeleton	6.94e-04
LGG	hsa04010	<b>MAPK signaling pathway **</b>	<b>1.26e-08</b>
	hsa04724	Glutamatergic synapse	4.80e-08
	hsa04727	GABAergic synapse	1.05e-06
	hsa04514	Cell adhesion molecules	5.96e-06
	hsa04713	Circadian entrainment	6.16e-06
	hsa04728	Dopaminergic synapse	1.58e-05
	hsa04360	Axon guidance	3.27e-05
	hsa05200	Pathways in cancer	1.04e-04
	hsa04725	Cholinergic synapse	1.08e-04
	hsa05032	Morphine addiction	1.69e-04
	hsa04015	Rap1 signaling pathway	3.59e-04
	hsa04014	Ras signaling pathway	5.50e-04
	hsa04371	Apelin signaling pathway	6.37e-04
	hsa05031	Amphetamine addiction	7.16e-04
	hsa05205	Proteoglycans in cancer	8.80e-04
	hsa04926	Relaxin signaling pathway	9.72e-04

#### 4.3.3 Caso Generale

Come è possibile notare dalla Tabella che segue, a differenza di [1], sono state trovate pathway per tutte le classi tumorali ma solo 17 hanno mostrato almeno una pathway collegata allo specifico tipo di tumore. I geni contenuti in tali pathway possono essere considerati biomarker specifici del tumore. Per le classi rimanenti, che non hanno mostrato pathway specifiche correlate alla coorte, sono state individuate diverse pathway concorrenti. Ad esempio, per le classi ACC, KICH, LGG, PCPG e UVM è stata rilevata la MAPK signaling pathway che è una nota pathway tumorale che si occupa della sopravvivenza delle cellule. Analizzando i geni correlati a questa pathway, si può notare che essi non sono gli stessi per tutte le classi sopramenzionate e dunque tali geni possono essere considerati candidati biomarker. Stesso discorso può essere fatto per le classi BRCA, CHOL, ESCA, GBM, KIRP, LUSC, PCPG e SARC, che pur non avendo mostrato pathway specifiche hanno mostrato la PI3K-Akt signaling che è una nota pathway tumorale per la sopravvivenza delle cellule proprio come la

MAPK signaling pathway. ESCA, GBM e KICH hanno mostrato anche la Hippo signaling pathway che si occupa di controllare le dimensioni degli organi regolando la proliferazione cellulare, l'apoptosi e l'auto-rinnovamento delle cellule staminali e la cui disregolazione è nota contribuisca allo sviluppo dei cancri. Un'altra considerazione può essere fatta per ESCA, KICH, KIRP, SARC, SKCM, TGCT, THCA e UCS che hanno mostrato come pathway correlate ECM-receptor e Focal Adhesion e, anche in questo caso, analizzando i geni correlati a tali pathway, è possibile notare che essi non coincidono per tutte le classi e quindi sono dei potenziali biomarker. Una nota a parte meritano la classi CHOL, KICH e READ che, seppur non sono state omesse dai risultati per completezza, la loro valenza è relativa in quanto: READ viene ignorata completamente in quanto l'accuracy ottenuta è troppo bassa e CHOL e KICH hanno un numero troppo limitato di campioni.

Tabella 11: Risultati della Pathways Analysis sui primi 400 geni per ogni tipo di tumore ( $P < 10^{-3}$ )

Tumore	Pathway correlata ID	Nome	P value
ACC	hsa04010	MAPK signaling pathway	2.58e-07
	hsa04015	Rap1 signaling pathway	2.65e-05
	hsa04512	ECM-receptor interaction	3.77e-05
	hsa04976	Bile secretion	1.33e-04
	hsa04115	p53 signaling pathway	3.75e-04
	hsa04610	Complement and coagulation cascades	5.56e-04
	hsa05144	Malaria	7.06e-04
BLCA	hsa04512	ECM-receptor interaction	9.98e-08
	hsa05205	Proteoglycans in cancer	2.15e-07
	hsa04514	Cell adhesion molecules	7.48e-06
	hsa04510	Focal adhesion	1.01e-05
	hsa05150	Staphylococcus aureus infection	7.54e-05
	hsa04270	Vascular smooth muscle contraction	1.26e-04
	hsa04668	TNF signaling pathway	1.64e-04
	hsa05200	Pathways in cancer	2.11e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	4.00e-04
	hsa04913	Ovarian steroidogenesis	5.06e-04
BRCA	hsa05165	Human papillomavirus infection	9.06e-04
	hsa05166	Human T-cell leukemia virus 1 infection	9.85e-04
	hsa04915	Estrogen signaling pathway	2.38e-08
	hsa04512	ECM-receptor interaction	2.44e-08
	hsa04151	<b>PI3K-Akt signaling pathway *</b>	4.61e-06
	hsa04927	Cortisol synthesis and secretion	8.11e-06
	hsa04928	Parathyroid hormone synthesis, secretion and action	2.10e-05
	hsa04934	Cushing syndrome	3.31e-05
	hsa04510	Focal adhesion	4.96e-05
	hsa05205	Proteoglycans in cancer	8.05e-05
ESCA	hsa05165	Human papillomavirus infection	8.86e-05
	hsa03320	<b>PPAR signaling pathway *</b>	8.95e-05
	hsa04514	Cell adhesion molecules	1.03e-04
	hsa04145	Phagosome	1.20e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.51e-04
	hsa05200	Pathways in cancer	5.14e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
Tumore	hsa01522	Endocrine resistance	7.22e-04
	hsa04926	Relaxin signaling pathway	7.76e-04
	hsa04540	Gap junction	9.62e-04
CESC	hsa05200	Pathways in cancer	8.37e-07
	hsa04115	p53 signaling pathway	4.64e-06
	hsa05205	<b>Proteoglycans in cancer</b>	<b>5.71e-05</b>
	hsa05222	Small cell lung cancer	8.00e-05
	hsa04512	ECM-receptor interaction	1.14e-04
	hsa04110	Cell cycle	1.54e-04
	hsa05146	Amoebiasis	1.57e-04
	hsa04080	Neuroactive ligand-receptor interaction	1.79e-04
	hsa04915	Estrogen signaling pathway	2.85e-04
	hsa04934	Cushing syndrome	4.46e-04
CHOL	hsa05144	Malaria	5.78e-04
	hsa04512	ECM-receptor interaction	3.94e-10
	hsa04610	Complement and coagulation cascades	1.92e-08
	hsa00340	Histidine metabolism	6.82e-06
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	8.03e-06
	hsa04514	Cell adhesion molecules	1.59e-05
	hsa04015	Rap1 signaling pathway	2.18e-05
	hsa04020	Calcium signaling pathway	3.36e-05
	hsa04510	Focal adhesion	3.57e-05
	hsa04974	Protein digestion and absorption	4.25e-05
	hsa04950	Maturity onset diabetes of the young	4.90e-05
	hsa04540	Gap junction	6.49e-05
	hsa04080	Neuroactive ligand-receptor interaction	8.23e-05
	hsa05146	Amoebiasis	9.96e-05
	hsa04060	Cytokine-cytokine receptor interaction	1.37e-04
	hsa04928	Parathyroid hormone synthesis, secretion and action	1.92e-04
	hsa04062	Chemokine signaling pathway	2.71e-04
COAD	hsa05414	Dilated cardiomyopathy	6.96e-04
	hsa04530	Tight junction	8.58e-04
	hsa04151	<b>PI3K-Akt signaling pathway **</b>	<b>8.82e-04</b>
	hsa04640	Hematopoietic cell lineage	6.01e-06
	hsa04145	Phagosome	1.01e-05
	hsa04060	Cytokine-cytokine receptor interaction	1.18e-05
	hsa04974	Protein digestion and absorption	1.37e-05
	hsa04512	ECM-receptor interaction	1.73e-05
	hsa04080	Neuroactive ligand-receptor interaction	4.12e-05
	hsa04514	Cell adhesion molecules	5.28e-05
DLBC	hsa05332	Graft-versus-host disease	1.96e-04
	hsa05323	Rheumatoid arthritis	9.33e-04
	hsa04940	Type I diabetes mellitus	7.84e-11
	hsa05330	<b>Allograft rejection</b>	<b>1.85e-10</b>
DLBC	hsa05332	<b>Graft-versus-host disease</b>	<b>3.28e-10</b>
	hsa04640	Hematopoietic cell lineage	5.58e-10

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04659	<b>Th17 cell differentiation</b>	6.90e-10
	hsa04672	Intestinal immune network for IgA production	2.72e-09
	hsa04145	Phagosome	2.08e-08
	hsa05321	Inflammatory bowel disease	2.47e-08
	hsa05150	Staphylococcus aureus infection	4.84e-08
	hsa04658	<b>Th1 and Th2 cell differentiation</b>	5.35e-08
	hsa05323	Rheumatoid arthritis	7.22e-08
	hsa05416	Viral myocarditis	7.58e-08
	hsa05140	Leishmaniasis	1.01e-07
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	1.52e-07
	hsa04514	<b>Cell adhesion molecules</b>	1.78e-07
	hsa05320	Autoimmune thyroid disease	4.97e-07
	hsa04610	Complement and coagulation cascades	1.50e-06
	hsa04062	<b>Chemokine signaling pathway</b>	2.65e-06
	hsa05169	Epstein-Barr virus infection	5.57e-06
	hsa05166	Human T-cell leukemia virus 1 infection	9.70e-06
	hsa04060	Cytokine-cytokine receptor interaction	1.04e-05
	hsa05133	Pertussis	1.29e-05
	hsa04662	<b>B cell receptor signaling pathway</b>	1.83e-05
	hsa05310	Asthma	1.91e-05
	hsa04380	Osteoclast differentiation	2.74e-05
	hsa05152	Tuberculosis	1.20e-04
	hsa05145	Toxoplasmosis	1.51e-04
	hsa04612	Antigen processing and presentation	1.89e-04
	hsa04064	<b>NF-kappa B signaling pathway</b>	2.26e-04
	hsa04621	NOD-like receptor signaling pathway	4.09e-04
	hsa05130	Pathogenic Escherichia coli infection	4.96e-04
	hsa04115	p53 signaling pathway	5.14e-04
	hsa04625	C-type lectin receptor signaling pathway	5.46e-04
	hsa04928	Parathyroid hormone synthesis, secretion and action	7.52e-04
ESCA	hsa04512	ECM-receptor interaction	1.38e-12
	hsa05414	Dilated cardiomyopathy	5.86e-10
	hsa05412	Arrhythmic right ventricular cardiomyopathy	2.79e-09
	hsa04510	Focal adhesion	2.85e-08
	hsa05410	Hypertrophic cardiomyopathy	3.42e-08
	hsa05165	Human papillomavirus infection	2.32e-07
	hsa05200	Pathways in cancer	2.31e-06
	hsa04022	cGMP-PKG signaling pathway	1.55e-05
	hsa04151	PI3K-Akt signaling pathway *	2.25e-05
	hsa05146	Amoebiasis	6.56e-05
	hsa05222	Small cell lung cancer	9.19e-05
	hsa04934	Cushing syndrome	1.02e-04
	hsa05205	Proteoglycans in cancer	1.50e-04
	hsa04360	Axon guidance	2.40e-04
	hsa04550	Signaling pathways regulating pluripotency of stem cells	2.72e-04
	hsa04390	Hippo signaling pathway	2.99e-04
	hsa04916	Melanogenesis	4.06e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
Tumore	hsa02010	ABC transporters	7.09e-04
	hsa05224	Breast cancer	9.80e-04
GBM	hsa04060	Cytokine-cytokine receptor interaction	4.80e-10
	hsa04080	Neuroactive ligand-receptor interaction	8.72e-10
	hsa04062	Chemokine signaling pathway	2.85e-06
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	7.48e-06
	hsa05323	Rheumatoid arthritis	1.50e-05
	hsa04015	Rap1 signaling pathway	3.59e-05
	hsa05205	Proteoglycans in cancer	4.06e-05
	hsa04512	ECM-receptor interaction	4.60e-05
	hsa04510	Focal adhesion	5.20e-05
	hsa05144	Malaria	6.52e-05
	hsa05032	Morphine addiction	8.44e-05
	hsa05200	Pathways in cancer	1.93e-04
	hsa04724	Glutamatergic synapse	2.69e-04
	hsa04020	Calcium signaling pathway	5.21e-04
	hsa04974	Protein digestion and absorption	6.78e-04
	hsa04151	<b>PI3K-Akt signaling pathway **</b>	7.09e-04
HNSC	hsa04390	Hippo signaling pathway	7.86e-04
	hsa04371	Apelin signaling pathway	7.87e-04
	hsa04270	Vascular smooth muscle contraction	8.89e-04
	hsa04512	ECM-receptor interaction	5.74e-12
	hsa05414	<b>Dilated cardiomyopathy</b>	8.53e-09
	hsa05410	Hypertrophic cardiomyopathy	2.73e-08
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	2.10e-07
	hsa04510	Focal adhesion	3.46e-05
	hsa05146	Amoebiasis	5.55e-05
	hsa04151	PI3K-Akt signaling pathway	5.93e-05
	hsa04020	Calcium signaling pathway	6.03e-05
	hsa04713	Circadian entrainment	6.61e-05
	hsa05165	Human papillomavirus infection	1.01e-04
	hsa05150	Staphylococcus aureus infection	1.58e-04
	hsa04915	Estrogen signaling pathway	2.79e-04
	hsa04060	Cytokine-cytokine receptor interaction	3.67e-04
KICH	hsa04926	Relaxin signaling pathway	4.78e-04
	hsa04921	Oxytocin signaling pathway	8.48e-04
	hsa04724	Glutamatergic synapse	8.86e-04
	hsa05200	Pathways in cancer	1.25e-06
	hsa04940	Type I diabetes mellitus	6.03e-06
	hsa04015	Rap1 signaling pathway	1.42e-05
	hsa04510	Focal adhesion	2.06e-05
	hsa04916	Melanogenesis	2.42e-05
	hsa04390	Hippo signaling pathway	2.90e-05
	hsa05205	Proteoglycans in cancer	3.50e-05
	hsa04512	ECM-receptor interaction	4.17e-05
	hsa05217	Basal cell carcinoma	4.29e-05

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04640	Hematopoietic cell lineage	4.61e-05
	hsa04010	MAPK signaling pathway	9.56e-05
	hsa04060	Cytokine-cytokine receptor interaction	1.06e-04
	hsa05226	Gastric cancer	1.13e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.50e-04
	hsa04310	<b>Wnt signaling pathway *</b>	1.79e-04
	hsa05225	Hepatocellular carcinoma	2.96e-04
	hsa04020	Calcium signaling pathway	4.55e-04
	hsa04934	Cushing syndrome	5.54e-04
	hsa05321	Inflammatory bowel disease	6.77e-04
	hsa00410	beta-Alanine metabolism	8.03e-04
	hsa05224	Breast cancer	9.13e-04
KIRC	hsa04514	Cell adhesion molecules	5.72e-08
	hsa04060	Cytokine-cytokine receptor interaction	1.70e-07
	hsa04640	Hematopoietic cell lineage	8.25e-07
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	3.35e-06
	hsa04512	ECM-receptor interaction	6.44e-06
	hsa04610	<b>Complement and coagulation cascades</b>	1.28e-05
	hsa05332	Graft-versus-host disease	1.10e-04
	hsa04510	Focal adhesion	1.57e-04
	hsa04015	Rap1 signaling pathway	2.23e-04
	hsa04080	Neuroactive ligand-receptor interaction	3.87e-04
	hsa04940	Type I diabetes mellitus	5.56e-04
	hsa04360	Axon guidance	5.71e-04
	hsa04151	PI3K-Akt signaling pathway *	6.52e-04
	hsa03320	PPAR signaling pathway	7.84e-04
	hsa05205	Proteoglycans in cancer	9.41e-04
	hsa05144	Malaria	9.42e-04
KIRP	hsa04512	ECM-receptor interaction	7.01e-13
	hsa04976	Bile secretion	1.98e-07
	hsa04510	Focal adhesion	1.26e-06
	hsa04974	Protein digestion and absorption	1.82e-06
	hsa05146	Amoebiasis	4.60e-06
	hsa04360	Axon guidance	3.32e-05
	hsa04928	Parathyroid hormone synthesis, secretion and action	8.28e-05
	hsa00260	Glycine, serine and threonine metabolism	1.22e-04
	hsa02010	ABC transporters	1.33e-04
	hsa04915	Estrogen signaling pathway	1.34e-04
	hsa04151	<b>PI3K-Akt signaling pathway *</b>	3.46e-04
	hsa00053	Ascorbate and aldarate metabolism	4.11e-04
	hsa04015	Rap1 signaling pathway	4.43e-04
	hsa04640	Hematopoietic cell lineage	4.49e-04
	hsa00340	Histidine metabolism	5.43e-04
	hsa00010	Glycolysis / Gluconeogenesis	6.20e-04
	hsa00650	Butanoate metabolism	6.85e-04
	hsa05414	Dilated cardiomyopathy	7.00e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04514	Cell adhesion molecules	7.19e-04
	hsa00620	Pyruvate metabolism	8.10e-04
LAML	hsa04640	Hematopoietic cell lineage	1.99e-15
	hsa05200	Pathways in cancer	3.52e-08
	hsa05140	<b>Leishmaniasis</b>	3.95e-08
	hsa04380	Osteoclast differentiation	5.34e-08
	hsa04514	Cell adhesion molecules	3.93e-07
	hsa05321	Inflammatory bowel disease	8.93e-07
	hsa05143	African trypanosomiasis	9.97e-07
	hsa05332	Graft-versus-host disease	1.69e-06
	hsa05340	Primary immunodeficiency	8.45e-06
	hsa04940	Type I diabetes mellitus	1.22e-05
	hsa04060	Cytokine-cytokine receptor interaction	1.97e-05
	hsa05414	Dilated cardiomyopathy	2.26e-05
	hsa04658	Th1 and Th2 cell differentiation	2.79e-05
	hsa04062	Chemokine signaling pathway	4.92e-05
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	5.12e-05
	hsa05310	Asthma	6.28e-05
	hsa04970	Salivary secretion	7.93e-05
	hsa04659	Th17 cell differentiation	8.71e-05
	hsa04672	<b>Intestinal immune network for IgA production</b>	8.74e-05
	hsa05323	Rheumatoid arthritis	9.67e-05
	hsa04015	Rap1 signaling pathway	1.13e-04
	hsa04072	Phospholipase D signaling pathway	1.24e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.37e-04
	hsa05150	Staphylococcus aureus infection	1.71e-04
	hsa05330	Allograft rejection	1.76e-04
	hsa04611	Platelet activation	2.12e-04
	hsa04145	Phagosome	2.19e-04
	hsa04010	MAPK signaling pathway	2.19e-04
	hsa04670	Leukocyte transendothelial migration	2.43e-04
	hsa04750	Inflammatory mediator regulation of TRP channels	2.45e-04
	hsa05410	Hypertrophic cardiomyopathy	3.87e-04
	hsa05144	Malaria	4.05e-04
	hsa04662	B cell receptor signaling pathway	6.09e-04
	hsa04625	C-type lectin receptor signaling pathway	6.67e-04
	hsa05418	Fluid shear stress and atherosclerosis	8.10e-04
LGG	hsa04512	ECM-receptor interaction	2.13e-06
	hsa04510	Focal adhesion	3.42e-06
	hsa04974	Protein digestion and absorption	1.83e-05
	hsa04640	Hematopoietic cell lineage	2.42e-05
	hsa04015	Rap1 signaling pathway	1.20e-04
	hsa04020	Calcium signaling pathway	1.96e-04
	hsa04724	Glutamatergic synapse	3.25e-04
	hsa04010	<b>MAPK signaling pathway **</b>	4.16e-04
	hsa04145	Phagosome	4.22e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
Tumore	hsa02010	ABC transporters	5.18e-04
	hsa04972	Pancreatic secretion	7.57e-04
	hsa04080	Neuroactive ligand-receptor interaction	9.97e-04
LIHC	hsa04610	Complement and coagulation cascades	2.48e-15
	hsa05150	Staphylococcus aureus infection	1.08e-08
	hsa03320	PPAR signaling pathway	3.42e-06
	hsa04216	Ferroptosis	4.49e-06
	hsa04979	Cholesterol metabolism	8.20e-06
	hsa04940	Type I diabetes mellitus	9.71e-06
	hsa00140	Steroid hormone biosynthesis	1.55e-05
	hsa04950	Maturity onset diabetes of the young	3.21e-05
	hsa04514	Cell adhesion molecules	6.27e-05
	hsa05332	Graft-versus-host disease	1.28e-04
	hsa05330	Allograft rejection	1.46e-04
	hsa05323	Rheumatoid arthritis	1.97e-04
	hsa04612	Antigen processing and presentation	2.22e-04
LUSC	hsa00830	Retinol metabolism	2.29e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	2.65e-04
	hsa04060	<b>Cytokine-cytokine receptor interaction **</b>	2.74e-04
	hsa04062	<b>Chemokine signaling pathway **</b>	2.99e-04
	hsa05321	Inflammatory bowel disease	3.57e-04
LUAD	hsa04610	Complement and coagulation cascades	1.82e-07
	hsa04145	Phagosome	4.00e-06
	hsa04514	Cell adhesion molecules	5.46e-05
	hsa05150	Staphylococcus aureus infection	6.39e-05
	hsa05205	Proteoglycans in cancer	3.05e-04
	hsa05416	Viral myocarditis	4.79e-04
MESO	hsa04512	ECM-receptor interaction	1.39e-08
	hsa04060	Cytokine-cytokine receptor interaction	1.81e-07
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	3.53e-07
	hsa04514	Cell adhesion molecules	4.09e-06
	hsa04640	Hematopoietic cell lineage	3.00e-05
	hsa04976	Bile secretion	1.11e-04
	hsa04151	<b>PI3K-Akt signaling pathway **</b>	3.13e-04
	hsa05200	Pathways in cancer	3.68e-04
	hsa04610	Complement and coagulation cascades	5.04e-04
	hsa04610	Complement and coagulation cascades	2.56e-09

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
Tumore	hsa04390	Hippo signaling pathway	1.33e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	1.34e-04
	hsa04145	Phagosome	1.51e-04
	hsa04915	Estrogen signaling pathway	2.49e-04
	hsa04621	NOD-like receptor signaling pathway	4.07e-04
	hsa05030	Cocaine addiction	5.50e-04
	hsa05165	Human papillomavirus infection	6.73e-04
	hsa05033	Nicotine addiction	6.76e-04
	hsa04015	Rap1 signaling pathway	8.79e-04
	hsa04060	Cytokine-cytokine receptor interaction	9.22e-04
OV	hsa04350	TGF-beta signaling pathway	9.52e-04
	hsa04360	Axon guidance	1.30e-05
	hsa04514	Cell adhesion molecules	2.59e-05
	hsa05200	Pathways in cancer	1.34e-04
PAAD	hsa045205	Proteoglycans in cancer	9.19e-04
	hsa04974	Protein digestion and absorption	6.26e-11
	hsa04512	ECM-receptor interaction	4.23e-08
	hsa04080	Neuroactive ligand-receptor interaction	2.22e-07
	hsa04972	Pancreatic secretion	5.47e-07
	hsa04950	Maturity onset diabetes of the young	5.65e-07
	hsa04610	Complement and coagulation cascades	1.21e-06
	hsa04510	Focal adhesion	7.62e-06
	hsa05414	Dilated cardiomyopathy	3.86e-05
	hsa04024	cAMP signaling pathway	5.05e-05
	hsa04911	Insulin secretion	1.17e-04
	hsa04670	Leukocyte transendothelial migration	3.59e-04
	hsa04060	Cytokine-cytokine receptor interaction	3.62e-04
	hsa04015	Rap1 signaling pathway	4.55e-04
PCPG	hsa04976	Bile secretion	5.47e-04
	hsa04978	Mineral absorption	9.30e-04
	hsa04510	Focal adhesion	3.73e-07
	hsa04514	Cell adhesion molecules	7.34e-07
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.42e-05
	hsa04512	ECM-receptor interaction	2.73e-05
	hsa05205	Proteoglycans in cancer	4.34e-05
	hsa04010	<b>MAPK signaling pathway **</b>	4.36e-05
	hsa04940	Type I diabetes mellitus	4.52e-05
	hsa05200	Pathways in cancer	5.98e-05
	hsa05416	Viral myocarditis	1.17e-04
	hsa04080	Neuroactive ligand-receptor interaction	1.27e-04
	hsa05332	Graft-versus-host disease	1.34e-04
	hsa05145	Toxoplasmosis	1.39e-04
	hsa05032	Morphine addiction	1.44e-04
	hsa04062	Chemokine signaling pathway	1.60e-04
	hsa04727	GABAergic synapse	2.64e-04
	hsa04060	Cytokine-cytokine receptor interaction	3.03e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04926	Relaxin signaling pathway	3.55e-04
	hsa04145	Phagosome	3.66e-04
	hsa04610	Complement and coagulation cascades	4.01e-04
	hsa04724	Glutamatergic synapse	4.62e-04
	hsa04925	Aldosterone synthesis and secretion	5.00e-04
	hsa04020	Calcium signaling pathway	5.94e-04
	hsa05330	Allograft rejection	6.02e-04
	hsa04151	<b>PI3K-Akt signaling pathway **</b>	6.18e-04
	hsa04612	Antigen processing and presentation	6.42e-04
	hsa05410	Hypertrophic cardiomyopathy	8.04e-04
	hsa05414	Dilated cardiomyopathy	8.77e-04
PRAD	hsa04514	Cell adhesion molecules	1.12e-08
	hsa05205	Proteoglycans in cancer	2.34e-05
	hsa04940	Type I diabetes mellitus	6.91e-05
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	7.51e-05
	hsa04010	MAPK signaling pathway	2.19e-04
	hsa04060	Cytokine-cytokine receptor interaction	2.41e-04
	hsa04270	<b>Vascular smooth muscle contraction</b>	2.72e-04
	hsa04640	Hematopoietic cell lineage	4.10e-04
	hsa02010	ABC transporters	4.70e-04
	hsa05150	Staphylococcus aureus infection	6.03e-04
	hsa05130	Pathogenic Escherichia coli infection	6.11e-04
READ	hsa04080	Neuroactive ligand-receptor interaction	9.20e-09
	hsa04514	Cell adhesion molecules	1.60e-07
	hsa04510	Focal adhesion	1.28e-06
	hsa05146	Amoebiasis	1.66e-06
	hsa04060	Cytokine-cytokine receptor interaction	1.67e-06
	hsa04512	ECM-receptor interaction	1.92e-06
	hsa04015	Rap1 signaling pathway	2.16e-06
	hsa04020	Calcium signaling pathway	1.06e-04
	hsa04014	Ras signaling pathway	1.33e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	2.10e-04
	hsa05032	Morphine addiction	2.88e-04
	hsa05200	Pathways in cancer	3.21e-04
	hsa04270	Vascular smooth muscle contraction	5.22e-04
	hsa04611	Platelet activation	6.47e-04
	hsa04610	Complement and coagulation cascades	7.99e-04
	hsa05205	Proteoglycans in cancer	9.23e-04
	hsa05144	Malaria	9.32e-04
SARC	hsa04512	ECM-receptor interaction	9.92e-08
	hsa05205	Proteoglycans in cancer	1.33e-06
	hsa05410	Hypertrophic cardiomyopathy	2.26e-06
	hsa04510	Focal adhesion	4.53e-06
	hsa05414	Dilated cardiomyopathy	2.60e-05
	hsa04514	Cell adhesion molecules	2.83e-05
	hsa04151	PI3K-Akt signaling pathway	1.08e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa05200	Pathways in cancer	1.64e-04
	hsa04020	Calcium signaling pathway	1.98e-04
	hsa04610	Complement and coagulation cascades	3.00e-04
	hsa05416	Viral myocarditis	5.43e-04
	hsa04974	Protein digestion and absorption	6.30e-04
	hsa05144	Malaria	7.01e-04
SKCM	hsa04512	ECM-receptor interaction	2.05e-09
	hsa04514	Cell adhesion molecules	1.05e-06
	hsa04510	Focal adhesion	1.40e-05
	hsa04974	Protein digestion and absorption	4.75e-05
	hsa04970	Salivary secretion	1.20e-04
	hsa04145	Phagosome	2.35e-04
	hsa04360	Axon guidance	2.50e-04
	hsa04640	Hematopoietic cell lineage	4.15e-04
	hsa05205	Proteoglycans in cancer	4.35e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	4.89e-04
	hsa04062	Chemokine signaling pathway	7.83e-04
	hsa05416	Viral myocarditis	8.52e-04
	hsa05202	Transcriptional misregulation in cancer	8.72e-04
STAD	hsa04080	Neuroactive ligand-receptor interaction	1.29e-07
	hsa04060	Cytokine-cytokine receptor interaction	1.70e-05
	hsa04020	Calcium signaling pathway	5.87e-04
TGCT	hsa04514	Cell adhesion molecules	5.25e-07
	hsa04940	Type I diabetes mellitus	5.73e-07
	hsa05410	Hypertrophic cardiomyopathy	6.61e-07
	hsa04512	ECM-receptor interaction	1.32e-06
	hsa05414	Dilated cardiomyopathy	3.05e-06
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	9.50e-06
	hsa04974	Protein digestion and absorption	1.47e-05
	hsa05416	Viral myocarditis	4.79e-05
	hsa05205	Proteoglycans in cancer	7.82e-05
	hsa04510	Focal adhesion	9.72e-05
	hsa04145	Phagosome	1.22e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	1.62e-04
	hsa05332	Graft-versus-host disease	1.75e-04
	hsa05130	Pathogenic Escherichia coli infection	2.45e-04
	hsa04540	Gap junction	3.13e-04
	hsa04261	Adrenergic signaling in cardiomyocytes	4.38e-04
	hsa04062	Chemokine signaling pathway	5.34e-04
	hsa05330	Allograft rejection	7.56e-04
	hsa04550	Signaling pathways regulating pluripotency of stem cells	7.75e-04
	hsa04975	Fat digestion and absorption	8.41e-04
THCA	hsa04514	Cell adhesion molecules	3.21e-08
	hsa04512	ECM-receptor interaction	1.32e-07
	hsa04510	Focal adhesion	5.79e-06
	hsa05205	Proteoglycans in cancer	1.00e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa05146	Amoebiasis	1.08e-04
	hsa05144	Malaria	1.50e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.93e-04
	hsa05410	Hypertrophic cardiomyopathy	2.03e-04
	hsa04151	PI3K-Akt signaling pathway	3.40e-04
	hsa05200	Pathways in cancer	3.41e-04
	hsa04670	Leukocyte transendothelial migration	3.45e-04
	hsa04020	Calcium signaling pathway	3.83e-04
THYM	hsa04514	Cell adhesion molecules	5.87e-10
	hsa04672	Intestinal immune network for IgA production	1.43e-08
	hsa05150	Staphylococcus aureus infection	4.04e-08
	hsa04640	Hematopoietic cell lineage	3.30e-07
	hsa04940	Type I diabetes mellitus	4.25e-06
	hsa05416	Viral myocarditis	5.22e-06
	hsa04145	Phagosome	6.94e-06
	hsa04658	Th1 and Th2 cell differentiation	1.82e-05
	hsa05321	Inflammatory bowel disease	2.28e-05
	hsa04612	Antigen processing and presentation	5.83e-05
	hsa05332	Graft-versus-host disease	5.86e-05
	hsa05323	Rheumatoid arthritis	6.42e-05
	hsa04015	Rap1 signaling pathway	1.32e-04
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	1.37e-04
	hsa04659	Th17 cell differentiation	1.63e-04
	hsa04810	Regulation of actin cytoskeleton	1.65e-04
	hsa04512	ECM-receptor interaction	1.74e-04
	hsa05330	Allograft rejection	2.56e-04
	hsa04510	Focal adhesion	3.46e-04
	hsa04060	Cytokine-cytokine receptor interaction	6.09e-04
	hsa04151	PI3K-Akt signaling pathway	8.45e-04
	hsa05340	<b>Primary immunodeficiency</b>	9.58e-04
	hsa04020	Calcium signaling pathway	9.76e-04
UCEC	hsa05205	Proteoglycans in cancer	1.59e-06
	hsa04670	Leukocyte transendothelial migration	2.05e-05
	hsa04530	<b>Tight junction</b>	5.91e-05
	hsa04020	Calcium signaling pathway	6.01e-05
	hsa04514	Cell adhesion molecules	6.29e-05
	hsa04940	Type I diabetes mellitus	1.84e-04
	hsa05230	Central carbon metabolism in cancer	2.17e-04
	hsa05200	Pathways in cancer	2.29e-04
	hsa04512	ECM-receptor interaction	2.62e-04
	hsa05416	Viral myocarditis	2.64e-04
	hsa04611	Platelet activation	2.65e-04
	hsa05165	Human papillomavirus infection	2.71e-04
	hsa05145	Toxoplasmosis	2.73e-04
	hsa05332	Graft-versus-host disease	5.42e-04
	hsa04360	Axon guidance	6.07e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04550	Signaling pathways regulating pluripotency of stem cells	6.19e-04
	hsa05130	Pathogenic Escherichia coli infection	7.20e-04
	hsa05140	Leishmaniasis	8.01e-04
	hsa05418	Fluid shear stress and atherosclerosis	8.32e-04
	hsa04659	Th17 cell differentiation	9.04e-04
	hsa05414	Dilated cardiomyopathy	9.47e-04
	hsa04612	Antigen processing and presentation	9.48e-04
	hsa04270	Vascular smooth muscle contraction	9.90e-04
UCS	hsa04512	ECM-receptor interaction	2.44e-08
	hsa05410	Hypertrophic cardiomyopathy	6.86e-06
	hsa04360	Axon guidance	1.67e-05
	hsa04080	Neuroactive ligand-receptor interaction	1.98e-05
	hsa04974	Protein digestion and absorption	3.58e-05
	hsa05414	Dilated cardiomyopathy	7.21e-05
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	1.08e-04
	hsa05205	Proteoglycans in cancer	1.54e-04
	hsa05200	Pathways in cancer	1.57e-04
	hsa04510	Focal adhesion	1.97e-04
	hsa04151	PI3K-Akt signaling pathway	2.06e-04
	hsa04145	Phagosome	3.79e-04
	hsa04550	Signaling pathways regulating pluripotency of stem cells	5.53e-04
	hsa05165	Human papillomavirus infection	8.07e-04
	hsa04024	cAMP signaling pathway	8.55e-04
	hsa04010	MAPK signaling pathway	9.99e-04
UVM	hsa04940	Type I diabetes mellitus	4.06e-05
	hsa04540	Gap junction	6.89e-05
	hsa05145	Toxoplasmosis	1.19e-04
	hsa04010	MAPK signaling pathway **	1.20e-04
	hsa05032	Morphine addiction	1.25e-04
	hsa04020	Calcium signaling pathway	1.35e-04
	hsa04530	Tight junction	1.37e-04
	hsa04974	Protein digestion and absorption	1.62e-04
	hsa04510	Focal adhesion	1.91e-04
	hsa04512	ECM-receptor interaction	1.92e-04
	hsa04916	Melanogenesis	2.91e-04
	hsa05414	Dilated cardiomyopathy	3.12e-04
	hsa05140	Leishmaniasis	4.74e-04
	hsa05330	Allograft rejection	5.54e-04
	hsa04612	Antigen processing and presentation	5.72e-04
	hsa05165	Human papillomavirus infection	6.13e-04
	hsa04145	Phagosome	6.59e-04
	hsa04970	Salivary secretion	9.90e-04

## 5 VarNet: Variant Convolutional Neural Network

Tabella 12: Tabella del training GPU si dataset oversampled di VarNet

Fold ID	Accuracy	Epoche	Tempo
0	94%	200	14:00:34
1	95%	200	14:05:03
2	95%	189	13:22:22
3	95%	200	14:06:19
4	95%	200	14:06:19
5	95%	200	14:06:44
6	93%	200	14:06:45
7	95%	200	14:07:14
8	95%	200	14:07:05
9	96%	200	14:07:50

Tabella 13: Tabella del training GPU si dataset oversampled di Net

Fold ID	Accuracy	Epoche	Tempo
0	94%	44	2:40:24
1	95%	131	8:00:44
2	95%	49	2:56:24
3	95%	50	3:00:12
4	95%	200	12:10:15
5	94%	200	12:00:00
6	94%	43	2:34:48
7	95%	49	2:57:09
8	95%	39	2:21:24
9	96%	51	2:30:36

In questa sezione mostriamo l'applicazione del workflow che ha portato alla progettazione di VarNet ed il suo confronto con Net. In primo luogo descriviamo VarNet nel suo complesso. Dopodiché mostriamo i risultati della rete e le sue differenze con Net nei punti fondamentali del progetto: classificazione, interpretabilità delle decisioni prese in base al task assegnato e validazione dei biomarker estratti.

### 5.1 VarNet vs Net: Less is more?

Nella seguente sezione spieghiamo come viene organizzata e strutturata la rete VarNet. Dapprima mostriamo quali sono i goal interessati dallo sviluppo di VarNet, poi estraiamo le ipotesi di base che servono alla rete per raggiungere tali scopi, infine mostriamo come vengono effettuate le principali modifiche che hanno portato alla nascita di VarNet.

#### 5.1.1 Assunzioni e Obiettivi VarNet

Analizzando la metodologia applicata alla rete CNN di riferimento (per ulteriori dettagli consultare la sezione 3.4) ed i risultati ottenuti sulla pipeline dell'intero progetto (cfr. 4) ci siamo posti la seguente domanda:

Quali sono i punti salienti che possono rendere l'individuazione dei biomarker più efficiente, affidabile e veloce?

Da questa domanda sono stati estratti, usando la notazione espressa nella sezione A, i moduli che offrivano i migliori servizi per raggiungere tale scopo, raggruppati in 2 macro-categorie legate al processo di sviluppo di un progetto che usa il Deep Learning:

1. **Dati Genomici:** di questa categoria fanno parte i moduli **Raw Data**, **Preprocessing** e **Biological Validation**. Questi si occupano di trattare i *gene expression data*, che rappresentano il dominio di applicazione del nostro problema. Si potrebbe, quindi, pensare di migliorarne le qualità o cambiare il metodo con cui vengono validati i risultati biologici ottenuti dal modello, oppure si potrebbero utilizzare diversi progetti di raccolta ed analisi di ulteriori campioni biologici ampliando la conoscenza di FireHose. La complessità di tale operazioni può essere demandata ad ulteriori indagini e quindi tale categoria non sarà scelta per raggiungere lo scopo espresso in precedenza.
2. **Modello Deep Learning:** in questa categoria ci sono i moduli **Training & Test**, **Heatmaps Generation** e **Performance Evaluation**. Essi trattano tutti gli aspetti fondamentali per il modello di Deep Learning scelto (cfr. 3.4.1), dall'addestramento della rete alla sua valutazione ed interpretabilità decisionale.

Da quanto espresso in precedenza abbiamo deciso di concentrare i nostri sforzi sul modello DL usato nel lavoro di Lyu e Haque [1], concentrandoci sui seguenti due punti:

1. **Architettura:** la composizione degli strati nascosti della rete rappresenta il primo punto focale che ci permette di migliorare le performance del modello.
2. **Learning Blocks:** li definiamo come i componenti fondamentali che modellano tutti gli aspetti dell'apprendimento necessari all'efficacia della rete feedforward. Essi sono: *Activation Function*, *Loss Function* e *Optimizer*.

### 5.1.2 Schema della rete: VarNet vs Net

La Figura 22 mostra l'architettura di VarNet, confrontandola con Net (Figura 3 a pagina 9) la prima differenza notabile è che VarNet contiene un livello di profondità in più osservando le dovute precauzioni sul dataset (cfr 3.4). In VarNet, dunque, viene aggiunto prima del layer di drop-out un quarto layer convoluzionale "conv4" contenente 512 filtri e posti immediatamente dopo di esso, un layer di Max-Pooling e Batch Normalization. Oltre ciò cambiano le dimensioni del primo layer fully-connected che risulta essere di 18432. Il resto delle reti è fatta come Net. Una seconda differenza tra le reti è basata sul numero di parametri e dal peso complessivo occupato sul disco: NET: Total-params: 38,778,755, Total-Size (MB): 160.65 e VARNET: Total-params: 17,415,555, Total-Size (MB): 80.78. Concludiamo dicendo che la modifica è stata indotta dalla parziale applicazione del principio fondamentale dietro il design di VGG [40], cercando di sfruttare il numero di filtri, andando ad aumentare la profondità dei livelli in cui ci sono i layer convoluzionali e cercando di mitigare il problema di rendere meno time-consuming il modello.

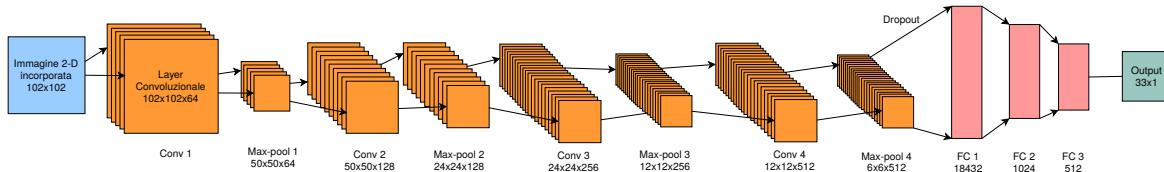


Figura 22: Composizione della Rete VarNet

### 5.1.3 Learning blocks: VarNet vs Net

Illustriamo ora le differenze tra le reti nei rispettivi learning blocks precedentemente definiti:

**Activation Function:** Per cercare di risolvere il problema di ReLU (cfr. 3.4.3), viene scelta come alternativa SeLU (Scaled Exponential Linear Units) che è una funzione di attivazione che induce l'auto-normalizzazione. Le attivazioni delle reti neurali che usano SELU convergono automaticamente a una media zero e ad una varianza unitaria. La SeLU è descritta come segue:

$$f(x) = \begin{cases} \lambda x & \text{if } x > 0 \\ \lambda(e^x - 1) & \text{if } x \leq 0 \end{cases} \quad (2)$$

I principali vantaggi sono: (1) SeLU non avrà il problema di retropropagare gradienti molto piccoli. (2) SeLU non soffre di momenti inattivi o morti. Uno svantaggio è che esistono pochi research paper che mostrano risultati efficienti di SeLU su diverse architetture.

**Loss Function:** Per sopperire alla mancanza della Cross-Entropy (cfr. 3.4.4) viene utilizzata la Multi Margin Loss che può essere intesa come il tentativo di assicurarsi che il punteggio per la classe corretta sia superiore alle altre classi di almeno un certo margine  $\delta > 0$ , altrimenti si incorre in una perdita che comporta una penalità al modello.

**Optimizer:** Si usa Nadam, che combina Adam con NAG (Nesterov Accelerated Gradient che è un ottimizzatore SGD basato sulla quantità di moto che "guarda avanti" a dove i saranno parametri, per calcolare il gradiente ex post piuttosto che ex ante), e che rispetto ad Adam risulta essere migliore nei task di ottimizzazione.

#### 5.1.4 Risultati training: VarNet vs Net

Osservando le Tabelle 12 a pagina 49 e 13 a pagina 49 si può fare la seguente osservazione: a parità prestazionali di GPU si nota che il numero di epoche ed i tempi su singola fold di VarNet sono più alti di Net. Questo è dovuto alla ridotta **batch size** rispetto a quella di Lyu e Haque [1] dovuta alla scarsità di memoria disponibile per la GPU in uso. Oltre ciò, la causa si addurre anche a come i learning blocks di VarNet suppongano principalmente che la sua loss function sia influenzata maggiormente dalla dispersività dei dati, usati in numero ridotto nel lotto impiegato per la fase di training (cfr. 3.4.5).

## 5.2 Valutazione Classificazione

Dalla matrice di confusione si può notare che la maggior parte delle classi sono classificate correttamente, tuttavia ci sono alcune classificazioni errate:

1. I campioni READ sono stati per lo più erroneamente classificati in COAD e ciò potrebbe essere dovuto ai pochi campioni di READ rispetto a quelli di COAD;
2. alcuni campioni di ESCA (25%) sono classificati erroneamente come STAD, questo potrebbe essere causato da alcuni campioni di ESCA i quali rientrano nella tipologia tumorale di carcinoma e che vengono visti dal modello come adenocarcinomi<sup>9</sup>. Ciò avviene perché la forma di adenocarcinoma si è presentata con una frequenza maggiore nei campioni ESCA rispetto alla sua forma di carcinoma dato che i campioni raccolti dal progetto FireHose provengono dagli Stati Uniti e questa forma per ESCA risulta essere la più vista nella sua popolazione;
3. alcuni campioni di UCS (33,33%) sono classificati erroneamente come UCEC e ciò può essere causato dalla difficoltà del modello a discriminare correttamente una particolare caratteristica che emerge nei campioni UCS: essi sono legati ad un tipologia particolare di cancro cioè il carcinosarcoma, questo significa, in ambito biologico, che osservando i campioni di tale coorte tumorale al microscopio si vede che a livello di proprietà istologiche, essi mostrano caratteristiche

---

<sup>9</sup>La principale differenza tra le due tipologie è la seguente: il carcinoma è un tumore maligno (cancro) che prende origine dalle cellule che compongono il tessuto epiteliale mentre l'adenocarcinoma è un tumore maligno (cancro) che prende origine dalle cellule che compongono un tipo di tessuto epiteliale specifico, ossia quello ghiandolare. L'adenocarcinoma, in sintesi, è un tipo di carcinoma.

sia del tumore carcinoma endometriale sia del sarcoma. Per cui in alcuni campioni questa manifestazione è avvenuta in maniera sbilanciata facendo generare l'errore di classificazione con la coorte UCEC che è un sotto-tipologia di carcinoma specifico della zona endometriale dell'utero.

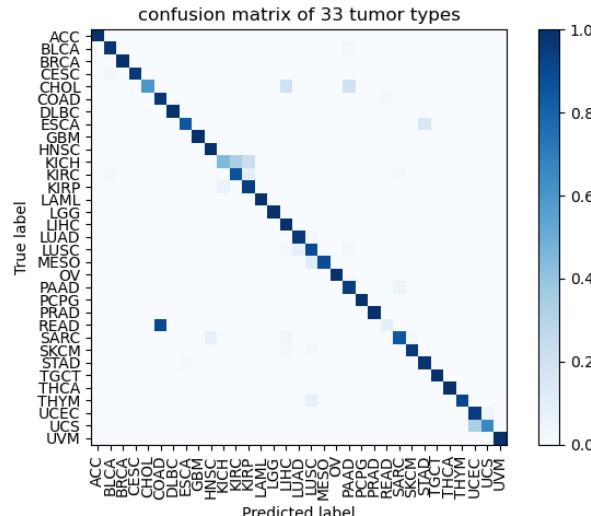


Figura 23: Confusion Matrix di VarNet

Tabella 14: Performance del metodo utilizzato con VarNet

Metodo	Accuracy	Precision	Recall	F1-score
CNN	94.83%	94.66%	94.83%	94.45%

### 5.3 Heatmap Generation

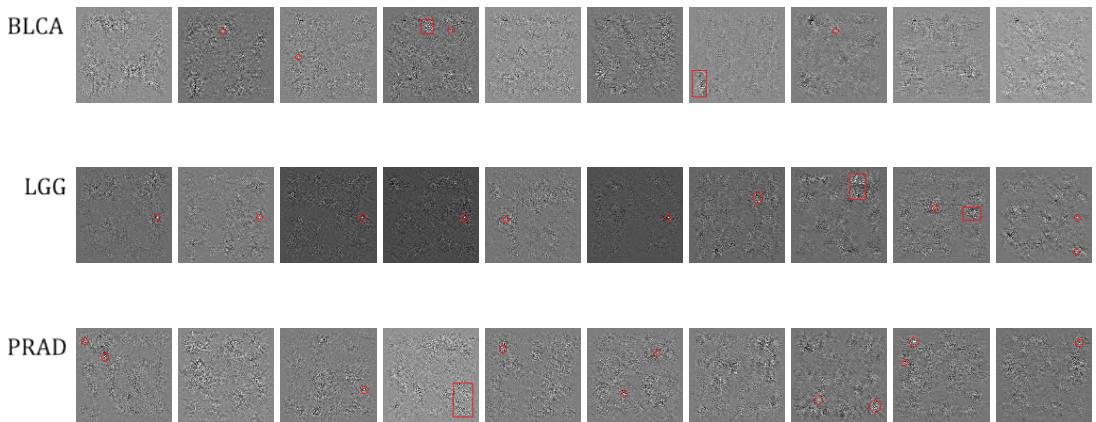


Figura 24: Alcuni esempi di heatmap legate a VarNet. Ogni colonna rappresenta il risultato di una fold. Nella prima riga ci sono le heatmap del tipo di tumore BLCA, nella seconda di LGG e nella terza di PRAD.

Confrontando le Figure 24 a pagina 52 e 20 a pagina 31 si nota questa principale differenza data dal fatto che VarNet ha un hidden layer in più rispetto a Net. Ciò porta a dire che nelle Guided Grad-CAM di VarNet si notano molti più pattern visibili internamente alle fold e tra fold diverse.

Tabella 15: Accuracy per coorte tumorale (calcolo effettuato su CPU e su dataset senza oversampling).

Coorte	Accuratezza ns. metodo	Accuratezza Variante	Accuratezza Riferimento
ACC	<b>1.00</b>	<b>1.00</b>	0.95
BLCA	0.98	0.98	0.97
BRCA	0.99	0.99	0.99
CESC	<b>0.97</b>	<b>0.97</b>	0.93
CHOL	<b>1.00</b>	<b>0.80</b>	0.56
COAD	0.97	0.97	0.95
DLBC	1.00	1.00	1.00
ESCA	<b>0.70</b>	0.75	0.77
GBM	0.94	0.94	0.94
HNSC	0.98	0.98	0.98
KICH	<b>0.89</b>	<b>0.89</b>	0.87
KIRC	0.97	0.97	0.95
KIRP	<b>0.88</b>	<b>0.88</b>	0.93
LAML	1.00	1.00	1.00
LGG	1.00	1.00	0.98
LIHC	0.95	0.95	0.97
LUAD	<b>0.91</b>	<b>0.91</b>	0.95
LUSC	0.91	<b>0.89</b>	0.91
MESO	<b>0.89</b>	<b>0.89</b>	0.94
OV	1.00	1.00	0.99
PAAD	0.95	0.95	0.97
PCPG	1.00	1.00	1.00
PRAD	1.00	1.00	1.00
READ	<b>0.09</b>	<b>0.09</b>	0.35
SARC	1.00	1.00	0.97
SKCM	1.00	1.00	0.98
STAD	0.96	0.96	0.96
TGCT	1.00	1.00	0.99
THCA	1.00	1.00	1.00
THYM	1.00	1.00	0.99
UCEC	1.00	1.00	0.96
UCS	<b>0.67</b>	<b>0.67</b>	0.81
UVM	1.00	1.00	0.99

#### 5.4 Validazione biologica

Come si può notare nella Figura 25 a pagina 54, i risultati ottenuti utilizzando la rete neurale VarNet sono del tutto paragonabili a quelli della rete neurale Net e dunque vale quanto già descritto nella

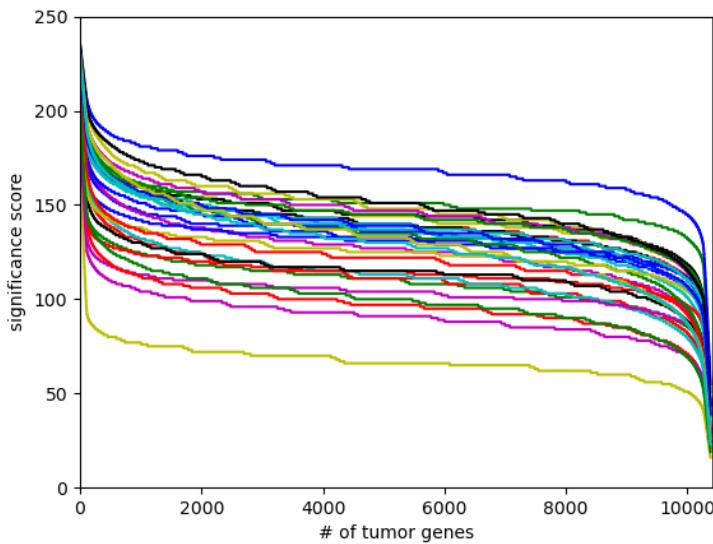


Figura 25: I cambi di intensità nelle heatmap per ogni classe. Si può notare come alcune classi condividano lo stesso pattern nei cambi di intensità.

sezione 4.3. Stesso discorso vale per le pathway biologiche rilevate: esse sono del tutto identiche a quelle rilevate a seguito dei test con la rete Net, così come riportato nella sezione 4.3. Nelle Tabella che segue si utilizzerà la seguente legenda: i valori di questo colore sono le nuove pathway rilevate dalle nostre analisi con il tool DAVID, i valori con questo background indicano le path che trovano riscontro anche nel lavoro di Lyu e Haque [1] e i valori con questo background indicano quelle pathway che sono specifiche della coorte tumorale.

Tabella 16: Risultati della Pathways Analysis sui primi 400 geni per ogni tipo di tumore ( $P < 10^{-3}$ )

Tumore	Pathway correlata ID	Nome	P value
ACC	hsa04010	MAPK signaling pathway	2.58e-07
	hsa04015	Rap1 signaling pathway	2.65e-05
	hsa04512	ECM-receptor interaction	3.77e-05
	hsa04976	Bile secretion	1.33e-04
	hsa04115	p53 signaling pathway	3.75e-04
	hsa04610	Complement and coagulation cascades	5.56e-04
	hsa05144	Malaria	7.06e-04
BLCA	hsa04512	ECM-receptor interaction	9.98e-08
	hsa05205	Proteoglycans in cancer	2.15e-07
	hsa04514	Cell adhesion molecules	7.48e-06
	hsa04510	Focal adhesion	1.01e-05
	hsa05150	Staphylococcus aureus infection	7.54e-05
	hsa04270	Vascular smooth muscle contraction	1.26e-04
	hsa04668	TNF signaling pathway	1.64e-04
	hsa05200	Pathways in cancer	2.11e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	4.00e-04
	hsa04913	Ovarian steroidogenesis	5.06e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
Tumore	hsa05165	Human papillomavirus infection	9.06e-04
	hsa05166	Human T-cell leukemia virus 1 infection	9.85e-04
BRCA	hsa04915	Estrogen signaling pathway	2.38e-08
	hsa04512	ECM-receptor interaction	2.44e-08
	hsa04151	<b>PI3K-Akt signaling pathway *</b>	4.61e-06
	hsa04927	Cortisol synthesis and secretion	8.11e-06
	hsa04928	Parathyroid hormone synthesis, secretion and action	2.10e-05
	hsa04934	Cushing syndrome	3.31e-05
	hsa04510	Focal adhesion	4.96e-05
	hsa05205	Proteoglycans in cancer	8.05e-05
	hsa05165	Human papillomavirus infection	8.86e-05
	hsa03320	<b>PPAR signaling pathway *</b>	8.95e-05
CESC	hsa04514	Cell adhesion molecules	1.03e-04
	hsa04145	Phagosome	1.20e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.51e-04
	hsa05200	Pathways in cancer	5.14e-04
	hsa01522	Endocrine resistance	7.22e-04
	hsa04926	Relaxin signaling pathway	7.76e-04
	hsa04540	Gap junction	9.62e-04
	hsa05200	Pathways in cancer	8.37e-07
	hsa04115	p53 signaling pathway	4.64e-06
	hsa05205	<b>Proteoglycans in cancer</b>	5.71e-05
CHOL	hsa05222	Small cell lung cancer	8.00e-05
	hsa04512	ECM-receptor interaction	1.14e-04
	hsa04110	Cell cycle	1.54e-04
	hsa05146	Amoebiasis	1.57e-04
	hsa04080	Neuroactive ligand-receptor interaction	1.79e-04
	hsa04915	Estrogen signaling pathway	2.85e-04
	hsa04934	Cushing syndrome	4.46e-04
	hsa05144	Malaria	5.78e-04
	hsa04512	ECM-receptor interaction	3.94e-10
	hsa04610	Complement and coagulation cascades	1.92e-08
	hsa00340	Histidine metabolism	6.82e-06
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	8.03e-06
	hsa04514	Cell adhesion molecules	1.59e-05
	hsa04015	Rap1 signaling pathway	2.18e-05
	hsa04020	Calcium signaling pathway	3.36e-05
	hsa04510	Focal adhesion	3.57e-05
	hsa04974	Protein digestion and absorption	4.25e-05
	hsa04950	Maturity onset diabetes of the young	4.90e-05
	hsa04540	Gap junction	6.49e-05
	hsa04080	Neuroactive ligand-receptor interaction	8.23e-05
	hsa05146	Amoebiasis	9.96e-05
	hsa04060	Cytokine-cytokine receptor interaction	1.37e-04
	hsa04928	Parathyroid hormone synthesis, secretion and action	1.92e-04
	hsa04062	Chemokine signaling pathway	2.71e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa05414	Dilated cardiomyopathy	6.96e-04
	hsa04530	Tight junction	8.58e-04
	hsa04151	<b>PI3K-Akt signaling pathway **</b>	8.82e-04
COAD	hsa04640	Hematopoietic cell lineage	6.01e-06
	hsa04145	Phagosome	1.01e-05
	hsa04060	Cytokine-cytokine receptor interaction	1.18e-05
	hsa04974	Protein digestion and absorption	1.37e-05
	hsa04512	ECM-receptor interaction	1.73e-05
	hsa04080	Neuroactive ligand-receptor interaction	4.12e-05
	hsa04514	Cell adhesion molecules	5.28e-05
	hsa05332	Graft-versus-host disease	1.96e-04
	hsa05323	Rheumatoid arthritis	9.33e-04
DLBC	hsa04940	Type I diabetes mellitus	7.84e-11
	hsa05330	<b>Allograft rejection</b>	1.85e-10
	hsa05332	<b>Graft-versus-host disease</b>	3.28e-10
	hsa04640	Hematopoietic cell lineage	5.58e-10
	hsa04659	Th17 cell differentiation	6.90e-10
	hsa04672	Intestinal immune network for IgA production	2.72e-09
	hsa04145	Phagosome	2.08e-08
	hsa05321	Inflammatory bowel disease	2.47e-08
	hsa05150	Staphylococcus aureus infection	4.84e-08
	hsa04658	Th1 and Th2 cell differentiation	5.35e-08
	hsa05323	Rheumatoid arthritis	7.22e-08
	hsa05416	Viral myocarditis	7.58e-08
	hsa05140	Leishmaniasis	1.01e-07
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	1.52e-07
	hsa04514	Cell adhesion molecules	1.78e-07
	hsa05320	Autoimmune thyroid disease	4.97e-07
	hsa04610	Complement and coagulation cascades	1.50e-06
	hsa04062	<b>Chemokine signaling pathway</b>	2.65e-06
	hsa05169	Epstein-Barr virus infection	5.57e-06
	hsa05166	Human T-cell leukemia virus 1 infection	9.70e-06
	hsa04060	Cytokine-cytokine receptor interaction	1.04e-05
	hsa05133	Pertussis	1.29e-05
	hsa04662	<b>B cell receptor signaling pathway</b>	1.83e-05
	hsa05310	Asthma	1.91e-05
	hsa04380	Osteoclast differentiation	2.74e-05
	hsa05152	Tuberculosis	1.20e-04
	hsa05145	Toxoplasmosis	1.51e-04
	hsa04612	Antigen processing and presentation	1.89e-04
	hsa04064	<b>NF-kappa B signaling pathway</b>	2.26e-04
	hsa04621	NOD-like receptor signaling pathway	4.09e-04
	hsa05130	Pathogenic Escherichia coli infection	4.96e-04
	hsa04115	p53 signaling pathway	5.14e-04
	hsa04625	C-type lectin receptor signaling pathway	5.46e-04
	hsa04928	Parathyroid hormone synthesis, secretion and action	7.52e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
ESCA	hsa04512	ECM-receptor interaction	1.38e-12
	hsa05414	Dilated cardiomyopathy	5.86e-10
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	2.79e-09
	hsa04510	Focal adhesion	2.85e-08
	hsa05410	Hypertrophic cardiomyopathy	3.42e-08
	hsa05165	Human papillomavirus infection	2.32e-07
	hsa05200	Pathways in cancer	2.31e-06
	hsa04022	cGMP-PKG signaling pathway	1.55e-05
	hsa04151	PI3K-Akt signaling pathway *	2.25e-05
	hsa05146	Amoebiasis	6.56e-05
	hsa05222	Small cell lung cancer	9.19e-05
	hsa04934	Cushing syndrome	1.02e-04
	hsa05205	Proteoglycans in cancer	1.50e-04
	hsa04360	Axon guidance	2.40e-04
	hsa04550	Signaling pathways regulating pluripotency of stem cells	2.72e-04
	hsa04390	Hippo signaling pathway	2.99e-04
	hsa04916	Melanogenesis	4.06e-04
GBM	hsa02010	ABC transporters	7.09e-04
	hsa05224	Breast cancer	9.80e-04
	hsa04060	Cytokine-cytokine receptor interaction	4.80e-10
	hsa04080	Neuroactive ligand-receptor interaction	8.72e-10
	hsa04062	Chemokine signaling pathway	2.85e-06
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	7.48e-06
	hsa05323	Rheumatoid arthritis	1.50e-05
	hsa04015	Rap1 signaling pathway	3.59e-05
	hsa05205	Proteoglycans in cancer	4.06e-05
	hsa04512	ECM-receptor interaction	4.60e-05
	hsa04510	Focal adhesion	5.20e-05
	hsa05144	Malaria	6.52e-05
	hsa05032	Morphine addiction	8.44e-05
	hsa05200	Pathways in cancer	1.93e-04
	hsa04724	Glutamatergic synapse	2.69e-04
	hsa04020	Calcium signaling pathway	5.21e-04
	hsa04974	Protein digestion and absorption	6.78e-04
HNSC	hsa04151	<b>PI3K-Akt signaling pathway **</b>	7.09e-04
	hsa04390	Hippo signaling pathway	7.86e-04
	hsa04371	Apelin signaling pathway	7.87e-04
	hsa04270	Vascular smooth muscle contraction	8.89e-04
	hsa04512	ECM-receptor interaction	5.74e-12

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
KICH	hsa04713	Circadian entrainment	6.61e-05
	hsa05165	Human papillomavirus infection	1.01e-04
	hsa05150	Staphylococcus aureus infection	1.58e-04
	hsa04915	Estrogen signaling pathway	2.79e-04
	hsa04060	Cytokine-cytokine receptor interaction	3.67e-04
	hsa04926	Relaxin signaling pathway	4.78e-04
	hsa04921	Oxytocin signaling pathway	8.48e-04
	hsa04724	Glutamatergic synapse	8.86e-04
KICH	hsa05200	Pathways in cancer	1.25e-06
	hsa04940	Type I diabetes mellitus	6.03e-06
	hsa04015	Rap1 signaling pathway	1.42e-05
	hsa04510	Focal adhesion	2.06e-05
	hsa04916	Melanogenesis	2.42e-05
	hsa04390	Hippo signaling pathway	2.90e-05
	hsa05205	Proteoglycans in cancer	3.50e-05
	hsa04512	ECM-receptor interaction	4.17e-05
	hsa05217	Basal cell carcinoma	4.29e-05
	hsa04640	Hematopoietic cell lineage	4.61e-05
	hsa04010	MAPK signaling pathway	9.56e-05
	hsa04060	Cytokine-cytokine receptor interaction	1.06e-04
	hsa05226	Gastric cancer	1.13e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.50e-04
	hsa04310	Wnt signaling pathway *	1.79e-04
	hsa05225	Hepatocellular carcinoma	2.96e-04
	hsa04020	Calcium signaling pathway	4.55e-04
	hsa04934	Cushing syndrome	5.54e-04
KIRC	hsa05321	Inflammatory bowel disease	6.77e-04
	hsa00410	beta-Alanine metabolism	8.03e-04
	hsa05224	Breast cancer	9.13e-04
	hsa04514	Cell adhesion molecules	5.72e-08
	hsa04060	Cytokine-cytokine receptor interaction	1.70e-07
	hsa04640	Hematopoietic cell lineage	8.25e-07
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	3.35e-06
	hsa04512	ECM-receptor interaction	6.44e-06
	hsa04610	<b>Complement and coagulation cascades</b>	1.28e-05
	hsa05332	Graft-versus-host disease	1.10e-04
	hsa04510	Focal adhesion	1.57e-04
	hsa04015	Rap1 signaling pathway	2.23e-04
	hsa04080	Neuroactive ligand-receptor interaction	3.87e-04
KIRP	hsa04940	Type I diabetes mellitus	5.56e-04
	hsa04360	Axon guidance	5.71e-04
	hsa04151	PI3K-Akt signaling pathway *	6.52e-04
	hsa03320	PPAR signaling pathway	7.84e-04
	hsa05205	Proteoglycans in cancer	9.41e-04
	hsa05144	Malaria	9.42e-04
KIRP	hsa04512	ECM-receptor interaction	7.01e-13

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04976	Bile secretion	1.98e-07
	hsa04510	Focal adhesion	1.26e-06
	hsa04974	Protein digestion and absorption	1.82e-06
	hsa05146	Amoebiasis	4.60e-06
	hsa04360	Axon guidance	3.32e-05
	hsa04928	Parathyroid hormone synthesis, secretion and action	8.28e-05
	hsa00260	Glycine, serine and threonine metabolism	1.22e-04
	hsa02010	ABC transporters	1.33e-04
	hsa04915	Estrogen signaling pathway	1.34e-04
	hsa04151	PI3K-Akt signaling pathway *	3.46e-04
	hsa00053	Ascorbate and aldarate metabolism	4.11e-04
	hsa04015	Rap1 signaling pathway	4.43e-04
	hsa04640	Hematopoietic cell lineage	4.49e-04
	hsa00340	Histidine metabolism	5.43e-04
	hsa00010	Glycolysis / Gluconeogenesis	6.20e-04
	hsa00650	Butanoate metabolism	6.85e-04
	hsa05414	Dilated cardiomyopathy	7.00e-04
	hsa04514	Cell adhesion molecules	7.19e-04
	hsa00620	Pyruvate metabolism	8.10e-04
LAML	hsa04640	Hematopoietic cell lineage	1.99e-15
	hsa05200	Pathways in cancer	3.52e-08
	hsa05140	<b>Leishmaniasis</b>	3.95e-08
	hsa04380	Osteoclast differentiation	5.34e-08
	hsa04514	Cell adhesion molecules	3.93e-07
	hsa05321	Inflammatory bowel disease	8.93e-07
	hsa05143	African trypanosomiasis	9.97e-07
	hsa05332	Graft-versus-host disease	1.69e-06
	hsa05340	Primary immunodeficiency	8.45e-06
	hsa04940	Type I diabetes mellitus	1.22e-05
	hsa04060	Cytokine-cytokine receptor interaction	1.97e-05
	hsa05414	Dilated cardiomyopathy	2.26e-05
	hsa04658	Th1 and Th2 cell differentiation	2.79e-05
	hsa04062	Chemokine signaling pathway	4.92e-05
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	5.12e-05
	hsa05310	Asthma	6.28e-05
	hsa04970	Salivary secretion	7.93e-05
	hsa04659	Th17 cell differentiation	8.71e-05
	hsa04672	<b>Intestinal immune network for IgA production</b>	8.74e-05
	hsa05323	Rheumatoid arthritis	9.67e-05
	hsa04015	Rap1 signaling pathway	1.13e-04
	hsa04072	Phospholipase D signaling pathway	1.24e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.37e-04
	hsa05150	Staphylococcus aureus infection	1.71e-04
	hsa05330	Allograft rejection	1.76e-04
	hsa04611	Platelet activation	2.12e-04
	hsa04145	Phagosome	2.19e-04
	hsa04010	MAPK signaling pathway	2.19e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
LGG	hsa04670	Leukocyte transendothelial migration	2.43e-04
	hsa04750	Inflammatory mediator regulation of TRP channels	2.45e-04
	hsa05410	Hypertrophic cardiomyopathy	3.87e-04
	hsa05144	Malaria	4.05e-04
	hsa04662	B cell receptor signaling pathway	6.09e-04
	hsa04625	C-type lectin receptor signaling pathway	6.67e-04
	hsa05418	Fluid shear stress and atherosclerosis	8.10e-04
LIHC	hsa04512	ECM-receptor interaction	2.13e-06
	hsa04510	Focal adhesion	3.42e-06
	hsa04974	Protein digestion and absorption	1.83e-05
	hsa04640	Hematopoietic cell lineage	2.42e-05
	hsa04015	Rap1 signaling pathway	1.20e-04
	hsa04020	Calcium signaling pathway	1.96e-04
	hsa04724	Glutamatergic synapse	3.25e-04
	hsa04010	<b>MAPK signaling pathway **</b>	4.16e-04
	hsa04145	Phagosome	4.22e-04
	hsa02010	ABC transporters	5.18e-04
LUAD	hsa04972	Pancreatic secretion	7.57e-04
	hsa04080	Neuroactive ligand-receptor interaction	9.97e-04
	hsa04610	Complement and coagulation cascades	2.48e-15
	hsa05150	Staphylococcus aureus infection	1.08e-08
	hsa03320	PPAR signaling pathway	3.42e-06
	hsa04216	Ferroptosis	4.49e-06
	hsa04979	Cholesterol metabolism	8.20e-06
	hsa04940	Type I diabetes mellitus	9.71e-06
	hsa00140	Steroid hormone biosynthesis	1.55e-05
	hsa04950	Maturity onset diabetes of the young	3.21e-05
LUSC	hsa04514	Cell adhesion molecules	6.27e-05
	hsa05332	Graft-versus-host disease	1.28e-04
	hsa05330	Allograft rejection	1.46e-04
	hsa05323	Rheumatoid arthritis	1.97e-04
	hsa04612	Antigen processing and presentation	2.22e-04
	hsa00830	Retinol metabolism	2.29e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	2.65e-04
	hsa04060	<b>Cytokine-cytokine receptor interaction **</b>	2.74e-04
	hsa04062	<b>Chemokine signaling pathway **</b>	2.99e-04
	hsa05321	Inflammatory bowel disease	3.57e-04
LUSC	hsa04610	Complement and coagulation cascades	1.82e-07
	hsa04145	Phagosome	4.00e-06
	hsa04514	Cell adhesion molecules	5.46e-05
	hsa05150	Staphylococcus aureus infection	6.39e-05
	hsa05205	Proteoglycans in cancer	3.05e-04
	hsa05416	Viral myocarditis	4.79e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04514	Cell adhesion molecules	4.09e-06
	hsa04640	Hematopoietic cell lineage	3.00e-05
	hsa04976	Bile secretion	1.11e-04
	hsa04151	<b>PI3K-Akt signaling pathway **</b>	3.13e-04
	hsa05200	Pathways in cancer	3.68e-04
	hsa04610	Complement and coagulation cascades	5.04e-04
MESO	hsa04610	Complement and coagulation cascades	2.56e-09
	hsa04080	Neuroactive ligand-receptor interaction	1.26e-07
	hsa04512	ECM-receptor interaction	3.08e-07
	hsa05144	Malaria	6.56e-07
	hsa05150	<b>Staphylococcus aureus infection</b>	7.07e-07
	hsa04020	Calcium signaling pathway	5.55e-06
	hsa04974	Protein digestion and absorption	3.15e-05
	hsa04510	Focal adhesion	3.79e-05
	hsa05205	Proteoglycans in cancer	1.31e-04
	hsa04390	Hippo signaling pathway	1.33e-04
	hsa04514	Cell adhesion molecules	1.33e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	1.34e-04
	hsa04145	Phagosome	1.51e-04
	hsa04915	Estrogen signaling pathway	2.49e-04
	hsa04621	NOD-like receptor signaling pathway	4.07e-04
	hsa05030	Cocaine addiction	5.50e-04
	hsa05165	Human papillomavirus infection	6.73e-04
	hsa05033	Nicotine addiction	6.76e-04
	hsa04015	Rap1 signaling pathway	8.79e-04
	hsa04060	Cytokine-cytokine receptor interaction	9.22e-04
	hsa04350	TGF-beta signaling pathway	9.52e-04
OV	hsa04360	Axon guidance	1.30e-05
	hsa04514	Cell adhesion molecules	2.59e-05
	hsa05200	Pathways in cancer	1.34e-04
	hsa05205	Proteoglycans in cancer	9.19e-04
PAAD	hsa04974	Protein digestion and absorption	6.26e-11
	hsa04512	ECM-receptor interaction	4.23e-08
	hsa04080	Neuroactive ligand-receptor interaction	2.22e-07
	hsa04972	Pancreatic secretion	5.47e-07
	hsa04950	Maturity onset diabetes of the young	5.65e-07
	hsa04610	Complement and coagulation cascades	1.21e-06
	hsa04510	Focal adhesion	7.62e-06
	hsa05414	Dilated cardiomyopathy	3.86e-05
	hsa04024	cAMP signaling pathway	5.05e-05
	hsa04911	Insulin secretion	1.17e-04
	hsa04670	Leukocyte transendothelial migration	3.59e-04
	hsa04060	Cytokine-cytokine receptor interaction	3.62e-04
	hsa04015	Rap1 signaling pathway	4.55e-04
	hsa04976	Bile secretion	5.47e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04978	Mineral absorption	9.30e-04
PCPG	hsa04510	Focal adhesion	3.73e-07
	hsa04514	Cell adhesion molecules	7.34e-07
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.42e-05
	hsa04512	ECM-receptor interaction	2.73e-05
	hsa05205	Proteoglycans in cancer	4.34e-05
	hsa04010	<b>MAPK signaling pathway **</b>	4.36e-05
	hsa04940	Type I diabetes mellitus	4.52e-05
	hsa05200	Pathways in cancer	5.98e-05
	hsa05416	Viral myocarditis	1.17e-04
	hsa04080	Neuroactive ligand-receptor interaction	1.27e-04
	hsa05332	Graft-versus-host disease	1.34e-04
	hsa05145	Toxoplasmosis	1.39e-04
	hsa05032	Morphine addiction	1.44e-04
	hsa04062	Chemokine signaling pathway	1.60e-04
	hsa04727	GABAergic synapse	2.64e-04
	hsa04060	Cytokine-cytokine receptor interaction	3.03e-04
	hsa04926	Relaxin signaling pathway	3.55e-04
	hsa04145	Phagosome	3.66e-04
	hsa04610	Complement and coagulation cascades	4.01e-04
	hsa04724	Glutamatergic synapse	4.62e-04
	hsa04925	Aldosterone synthesis and secretion	5.00e-04
	hsa04020	Calcium signaling pathway	5.94e-04
	hsa05330	Allograft rejection	6.02e-04
	hsa04151	<b>PI3K-Akt signaling pathway **</b>	6.18e-04
	hsa04612	Antigen processing and presentation	6.42e-04
	hsa05410	Hypertrophic cardiomyopathy	8.04e-04
	hsa05414	Dilated cardiomyopathy	8.77e-04
PRAD	hsa04514	Cell adhesion molecules	1.12e-08
	hsa05205	Proteoglycans in cancer	2.34e-05
	hsa04940	Type I diabetes mellitus	6.91e-05
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	7.51e-05
	hsa04010	MAPK signaling pathway	2.19e-04
	hsa04060	Cytokine-cytokine receptor interaction	2.41e-04
	hsa04270	<b>Vascular smooth muscle contraction</b>	2.72e-04
	hsa04640	Hematopoietic cell lineage	4.10e-04
	hsa02010	ABC transporters	4.70e-04
	hsa05150	Staphylococcus aureus infection	6.03e-04
	hsa05130	Pathogenic Escherichia coli infection	6.11e-04
READ	hsa04080	Neuroactive ligand-receptor interaction	9.20e-09
	hsa04514	Cell adhesion molecules	1.60e-07
	hsa04510	Focal adhesion	1.28e-06
	hsa05146	Amoebiasis	1.66e-06
	hsa04060	Cytokine-cytokine receptor interaction	1.67e-06
	hsa04512	ECM-receptor interaction	1.92e-06
	hsa04015	Rap1 signaling pathway	2.16e-06

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa04020	Calcium signaling pathway	1.06e-04
	hsa04014	Ras signaling pathway	1.33e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	2.10e-04
	hsa05032	Morphine addiction	2.88e-04
	hsa05200	Pathways in cancer	3.21e-04
	hsa04270	Vascular smooth muscle contraction	5.22e-04
	hsa04611	Platelet activation	6.47e-04
	hsa04610	Complement and coagulation cascades	7.99e-04
	hsa05205	Proteoglycans in cancer	9.23e-04
	hsa05144	Malaria	9.32e-04
SARC	hsa04512	ECM-receptor interaction	9.92e-08
	hsa05205	Proteoglycans in cancer	1.33e-06
	hsa05410	Hypertrophic cardiomyopathy	2.26e-06
	hsa04510	Focal adhesion	4.53e-06
	hsa05414	Dilated cardiomyopathy	2.60e-05
	hsa04514	Cell adhesion molecules	2.83e-05
	hsa04151	PI3K-Akt signaling pathway	1.08e-04
	hsa05200	Pathways in cancer	1.64e-04
	hsa04020	Calcium signaling pathway	1.98e-04
	hsa04610	Complement and coagulation cascades	3.00e-04
	hsa05416	Viral myocarditis	5.43e-04
	hsa04974	Protein digestion and absorption	6.30e-04
	hsa05144	Malaria	7.01e-04
SKCM	hsa04512	ECM-receptor interaction	2.05e-09
	hsa04514	Cell adhesion molecules	1.05e-06
	hsa04510	Focal adhesion	1.40e-05
	hsa04974	Protein digestion and absorption	4.75e-05
	hsa04970	Salivary secretion	1.20e-04
	hsa04145	Phagosome	2.35e-04
	hsa04360	Axon guidance	2.50e-04
	hsa04640	Hematopoietic cell lineage	4.15e-04
	hsa05205	Proteoglycans in cancer	4.35e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	4.89e-04
	hsa04062	Chemokine signaling pathway	7.83e-04
	hsa05416	Viral myocarditis	8.52e-04
	hsa05202	Transcriptional misregulation in cancer	8.72e-04
STAD	hsa04080	Neuroactive ligand-receptor interaction	1.29e-07
	hsa04060	Cytokine-cytokine receptor interaction	1.70e-05
	hsa04020	Calcium signaling pathway	5.87e-04
TGCT	hsa04514	Cell adhesion molecules	5.25e-07
	hsa04940	Type I diabetes mellitus	5.73e-07
	hsa05410	Hypertrophic cardiomyopathy	6.61e-07
	hsa04512	ECM-receptor interaction	1.32e-06
	hsa05414	Dilated cardiomyopathy	3.05e-06
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	9.50e-06
	hsa04974	Protein digestion and absorption	1.47e-05

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa05416	Viral myocarditis	4.79e-05
	hsa05205	Proteoglycans in cancer	7.82e-05
	hsa04510	Focal adhesion	9.72e-05
	hsa04145	Phagosome	1.22e-04
	hsa04061	Viral protein interaction with cytokine and cytokine receptor	1.62e-04
	hsa05332	Graft-versus-host disease	1.75e-04
	hsa05130	Pathogenic Escherichia coli infection	2.45e-04
	hsa04540	Gap junction	3.13e-04
	hsa04261	Adrenergic signaling in cardiomyocytes	4.38e-04
	hsa04062	Chemokine signaling pathway	5.34e-04
	hsa05330	Allograft rejection	7.56e-04
	hsa04550	Signaling pathways regulating pluripotency of stem cells	7.75e-04
	hsa04975	Fat digestion and absorption	8.41e-04
THCA	hsa04514	Cell adhesion molecules	3.21e-08
	hsa04512	ECM-receptor interaction	1.32e-07
	hsa04510	Focal adhesion	5.79e-06
	hsa05205	Proteoglycans in cancer	1.00e-04
	hsa05146	Amoebiasis	1.08e-04
	hsa05144	Malaria	1.50e-04
	hsa04933	AGE-RAGE signaling pathway in diabetic complications	1.93e-04
	hsa05410	Hypertrophic cardiomyopathy	2.03e-04
	hsa04151	PI3K-Akt signaling pathway	3.40e-04
	hsa05200	Pathways in cancer	3.41e-04
	hsa04670	Leukocyte transendothelial migration	3.45e-04
	hsa04020	Calcium signaling pathway	3.83e-04
THYM	hsa04514	Cell adhesion molecules	5.87e-10
	hsa04672	Intestinal immune network for IgA production	1.43e-08
	hsa05150	Staphylococcus aureus infection	4.04e-08
	hsa04640	Hematopoietic cell lineage	3.30e-07
	hsa04940	Type I diabetes mellitus	4.25e-06
	hsa05416	Viral myocarditis	5.22e-06
	hsa04145	Phagosome	6.94e-06
	hsa04658	Th1 and Th2 cell differentiation	1.82e-05
	hsa05321	Inflammatory bowel disease	2.28e-05
	hsa04612	Antigen processing and presentation	5.83e-05
	hsa05332	Graft-versus-host disease	5.86e-05
	hsa05323	Rheumatoid arthritis	6.42e-05
	hsa04015	Rap1 signaling pathway	1.32e-04
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	1.37e-04
	hsa04659	Th17 cell differentiation	1.63e-04
	hsa04810	Regulation of actin cytoskeleton	1.65e-04
	hsa04512	ECM-receptor interaction	1.74e-04
	hsa05330	Allograft rejection	2.56e-04
	hsa04510	Focal adhesion	3.46e-04
	hsa04060	Cytokine-cytokine receptor interaction	6.09e-04
	hsa04151	PI3K-Akt signaling pathway	8.45e-04

Continua nella prossima pagina

Tumore	Pathway correlata ID	Nome	P value
	hsa05340	<b>Primary immunodeficiency</b>	9.58e-04
	hsa04020	Calcium signaling pathway	9.76e-04
UCEC	hsa05205	Proteoglycans in cancer	1.59e-06
	hsa04670	Leukocyte transendothelial migration	2.05e-05
	hsa04530	<b>Tight junction</b>	5.91e-05
	hsa04020	Calcium signaling pathway	6.01e-05
	hsa04514	Cell adhesion molecules	6.29e-05
	hsa04940	Type I diabetes mellitus	1.84e-04
	hsa05230	Central carbon metabolism in cancer	2.17e-04
	hsa05200	Pathways in cancer	2.29e-04
	hsa04512	ECM-receptor interaction	2.62e-04
	hsa05416	Viral myocarditis	2.64e-04
	hsa04611	Platelet activation	2.65e-04
	hsa05165	Human papillomavirus infection	2.71e-04
	hsa05145	Toxoplasmosis	2.73e-04
	hsa05332	Graft-versus-host disease	5.42e-04
	hsa04360	Axon guidance	6.07e-04
	hsa04550	Signaling pathways regulating pluripotency of stem cells	6.19e-04
	hsa05130	Pathogenic Escherichia coli infection	7.20e-04
	hsa05140	Leishmaniasis	8.01e-04
	hsa05418	Fluid shear stress and atherosclerosis	8.32e-04
	hsa04659	Th17 cell differentiation	9.04e-04
	hsa05414	Dilated cardiomyopathy	9.47e-04
	hsa04612	Antigen processing and presentation	9.48e-04
	hsa04270	Vascular smooth muscle contraction	9.90e-04
UCS	hsa04512	ECM-receptor interaction	2.44e-08
	hsa05410	Hypertrophic cardiomyopathy	6.86e-06
	hsa04360	Axon guidance	1.67e-05
	hsa04080	Neuroactive ligand-receptor interaction	1.98e-05
	hsa04974	Protein digestion and absorption	3.58e-05
	hsa05414	Dilated cardiomyopathy	7.21e-05
	hsa05412	Arrhythmogenic right ventricular cardiomyopathy	1.08e-04
	hsa05205	Proteoglycans in cancer	1.54e-04
	hsa05200	Pathways in cancer	1.57e-04
	hsa04510	Focal adhesion	1.97e-04
	hsa04151	PI3K-Akt signaling pathway	2.06e-04
	hsa04145	Phagosome	3.79e-04
	hsa04550	Signaling pathways regulating pluripotency of stem cells	5.53e-04
	hsa05165	Human papillomavirus infection	8.07e-04
	hsa04024	cAMP signaling pathway	8.55e-04
	hsa04010	MAPK signaling pathway	9.99e-04
UVM	hsa04940	Type I diabetes mellitus	4.06e-05
	hsa04540	Gap junction	6.89e-05
	hsa05145	Toxoplasmosis	1.19e-04
	hsa04010	MAPK signaling pathway **	1.20e-04
	hsa05032	Morphine addiction	1.25e-04

Continua nella prossima pagina

Tumore	Pathway correlata		P value
	ID	Nome	
	hsa04020	Calcium signaling pathway	1.35e-04
	hsa04530	Tight junction	1.37e-04
	hsa04974	Protein digestion and absorption	1.62e-04
	hsa04510	Focal adhesion	1.91e-04
	hsa04512	ECM-receptor interaction	1.92e-04
	hsa04916	Melanogenesis	2.91e-04
	hsa05414	Dilated cardiomyopathy	3.12e-04
	hsa05140	Leishmaniasis	4.74e-04
	hsa05330	Allograft rejection	5.54e-04
	hsa04612	Antigen processing and presentation	5.72e-04
	hsa05165	Human papillomavirus infection	6.13e-04
	hsa04145	Phagosome	6.59e-04
	hsa04970	Salivary secretion	9.90e-04

## 6 Discussione

La scoperta dei biomarker specifici per ogni tumore è essenziale al fine di migliorare i popolari test genomici attuali. Con la saliva o con il sangue, questi test possono indicare le possibilità stimate dei diversi tumori per ogni individuo. L'essenzialità è dovuta al fatto che non è garantito che i biomarker generati dall'analisi differenziale siano specifici per il determinato tipo di tumore e dal fatto che alcuni tumori potrebbero condividere gli stessi biomarker. Per evitare questa problematica, in [1] hanno progettato un metodo che sfrutta la conoscenza di diversi tipi di tumore e che va a cercare i geni che possono essere usati per differenziarli. In questo lavoro abbiamo replicato il lavoro di [1] e dunque abbiamo utilizzato due reti neurali convoluzionali per effettuare la classificazione dei dati genomici. La prima rete usata è quella progettata da [1], mentre la seconda è una sua variante ispirata a VGG [40] il cui scopo era di migliorare le performance generali (purtroppo non riuscendoci) e migliorare l'accuracy per le singole coorti tumorali. Tale miglioramento è stato riscontrato, ad esempio, per OV e STAD.

La ricerca in ambito di computer vision si è sviluppata velocemente e molti dei metodi sono stati progettati per risolvere i problemi usando le deep neural network. Uno dei problemi presenti è che i dati genomici di solito hanno un'alta dimensionalità mentre molte delle architetture di deep learning sono per immagini 2D. In questo lavoro abbiamo mostrato che il metodo sviluppato da [1], nel quale si inglobano in maniera naïve i dati genomici (i geni) su ogni pixel di un'immagine 2D in base all'ordinamento cromosomico, è valido e che le performance sono eccellenti tranne che per alcuni tipi di tumore. In base a tali risultati è auspicabile che molti altri metodi di deep learning possano essere applicati ai dati genomici in futuro.

## 7 Conclusioni e Sviluppi Futuri

In questo lavoro ci eravamo posti come obiettivo quello di replicare nella sua interezza il lavoro svolto in [1] e di provare a migliorarlo andando ad agire sulle varie parti che lo compongono. Dapprima ci siamo occupati di scrivere le parti di codice mancanti (tutte le modifiche sono documentate nell'Appendice A), poi abbiamo eseguito dei test sia su casi ridotti sia sul caso completo e infine abbiamo provato a migliorare le performance utilizzando un'altra rete neurale come descritto nella sezione 5. È stato un lavoro lungo e difficile in quanto è stato necessario scrivere ex novo alcune parti di codice ed è stato necessario aggiornare il codice per meglio sfruttare i recenti miglioramenti nelle librerie utilizzate. In conclusione, seppur utilizzando la stessa rete del riferimento all'inizio, abbiamo ottenuto sia delle performance generali migliori sia performance migliori per alcune coorti tumorali specifiche. Inoltre, questo lavoro è stato fonte di ispirazione per lo studio ed è stato fondamentale per comprendere quanto il lavoro dei bioinformatici possa essere d'aiuto alla ricerca medica senza mai ovviamente poterla sostituire. Sviluppi futuri di questo lavoro sicuramente potranno migliorare le performance magari utilizzando un modello di deep learning diverso (ad esempio utilizzando modelli pretrained come VGG, AlexNet o Inception) oppure utilizzando reinforcement learning o i transformers. Un'altra strada percorribile è quella di fare un tuning ancora più preciso di quelli che sono stati i parametri di addestramento utilizzati al fine di trovare una configurazione migliore di quella attuale.

## 8 Ringraziamenti

Ci teniamo a ringraziare il Prof. Rocco Zaccagnino per averci dato la possibilità di lavorare a questo progetto in quanto eravamo desiderosi di poter lavorare manipolando le gene expression e per averci fornito ciò di cui avevamo bisogno per poter iniziare e le Prof.sse Clelia De Felice e Rosalba Zizza per averci fornito le nozioni teoriche necessarie a comprendere quanto indispensabile per lavorare con i dati genomici.

## Riferimenti bibliografici

- [1] Boyu Lyu and Anamul Haque. Deep learning based tumor type classification using gene expression data. In *Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics*, pages 89–96, 2018.
- [2] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.
- [3] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):6999–7019, Dec 2022.
- [4] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Azawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–6, 2017.
- [5] Arohan Ajit, Koustav Acharya, and Abhishek Samanta. A review of convolutional neural networks. In *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, pages 1–5, Feb 2020.
- [6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [7] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- [8] Illumina Inc. High-impact discovery through gene expression and regulation research, 2021.
- [9] Yuanyuan Li, Kai Kang, Juno M Krahn, Nicole Croutwater, Kevin Lee, David M Umbach, and Leping Li. A comprehensive genomic pan-cancer classification using the cancer genome atlas gene expression data. *BMC genomics*, 18:1–13, 2017.
- [10] Padideh Danaee, Reza Ghaeini, and David A Hendrix. A deep learning approach for cancer detection and relevant gene identification. In *Pacific symposium on biocomputing 2017*, pages 219–229. World Scientific, 2017.
- [11] Sebastian Bach, HHI Fraunhofer, Alexander Binder, EDU Sg, and Wojciech Samek. Deep taylor decomposition of neural networks. [n.d.].
- [12] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

- [13] <http://gdac.broadinstitute.org/>, [n.d.].
- [14] <https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga/using-tcga/tools>, [n.d.].
- [15] Bo Li and Colin N Dewey. Rsem: accurate transcript quantification from rna-seq data with or without a reference genome. *BMC bioinformatics*, 12:1–16, 2011.
- [16] Gordon Robertson, Jacqueline Schein, Readman Chiu, Richard Corbett, Matthew Field, Shaun D Jackman, Karen Mungall, Sam Lee, Hisanaga Mark Okada, Jenny Q Qian, et al. De novo assembly and analysis of rna-seq data. *Nature methods*, 7(11):909–912, 2010.
- [17] Manfred G Grabherr, Brian J Haas, Moran Yassour, Joshua Z Levin, Dawn A Thompson, Ido Amit, Xian Adiconis, Lin Fan, Rakimra Raychowdhury, Qiandong Zeng, et al. Full-length transcriptome assembly from rna-seq data without a reference genome. *Nature biotechnology*, 29(7):644–652, 2011.
- [18] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):2037–2041, Dec 2006.
- [19] Tony Lindeberg. Scale invariant feature transform. 2012.
- [20] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *The Journal of physiology*, 160(1):106, 1962.
- [21] Keiron O’Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.
- [22] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.
- [23] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.
- [24] Matthew D Zeiler and Rob Fergus. Stochastic pooling for regularization of deep convolutional neural networks. *arXiv preprint arXiv:1301.3557*, 2013.
- [25] Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, and Rob Fergus. Regularization of neural networks using dropconnect. In *International conference on machine learning*, pages 1058–1066. PMLR, 2013.
- [26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.
- [27] Leo Breiman. Bagging predictors. *Machine learning*, 24:123–140, 1996.
- [28] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [29] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.
- [30] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

- [31] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.
- [32] Frank Rosenblatt. Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Technical report, Cornell Aeronautical Lab Inc Buffalo NY, 1961.
- [33] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.
- [34] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019.
- [35] Alex Krizhevsky and Geoff Hinton. Convolutional deep belief networks on cifar-10. *Unpublished manuscript*, 40(7):1–9, 2010.
- [36] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. *arXiv preprint arXiv:1612.02295*, 2016.
- [37] Lianzhi Tan, Kaipeng Zhang, Kai Wang, Xiaoxing Zeng, Xiaojiang Peng, and Yu Qiao. Group emotion recognition with individual facial emotion cnns and global image based cnns. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 549–552, 2017.
- [38] Yi Liu, Liang He, and Jia Liu. Large margin softmax loss for speaker verification. *arXiv preprint arXiv:1904.03479*, 2019.
- [39] David Saad. *On-line learning in neural networks*. Cambridge University Press, 1999.
- [40] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [41] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.
- [42] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.
- [43] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1891–1898, 2014.
- [44] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. *Advances in neural information processing systems*, 27, 2014.
- [45] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.
- [46] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131, 2018.
- [47] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

- [48] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. *Advances in neural information processing systems*, 28, 2015.
- [49] Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [50] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8):2, 2012.
- [51] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.
- [52] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.
- [53] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.
- [54] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [55] Shikhar Sharma, Ryan Kiros, and Ruslan Salakhutdinov. Action recognition using visual attention. *arXiv preprint arXiv:1511.04119*, 2015.
- [56] Michel JAM Van Putten, Sebastian Olbrich, and Martijn Arns. Predicting sex from brain rhythms with deep learning. *Scientific reports*, 8(1):3069, 2018.
- [57] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive separable convolution. In *Proceedings of the IEEE international conference on computer vision*, pages 261–270, 2017.
- [58] Timothy Dozat. Incorporating nesterov momentum into adam. 2016.
- [59] Dat Quoc Nguyen and Karin Verspoor. Convolutional neural networks for chemical-disease relation extraction are improved with character-based word embeddings. *arXiv preprint arXiv:1805.10586*, 2018.
- [60] Alexander Schindler, Thomas Lidy, and Andreas Rauber. Multi-temporal resolution convolutional neural networks for acoustic scene classification. *arXiv preprint arXiv:1811.04419*, 2018.
- [61] Nasser M Nasrabadi. Pattern recognition and machine learning. *Journal of electronic imaging*, 16(4):049901, 2007.
- [62] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2011.
- [63] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.
- [64] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

- [65] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.
- [66] Mengchen Liu, Jiaxin Shi, Zhen Li, Chongxuan Li, Jun Zhu, and Shixia Liu. Towards better analysis of deep convolutional neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):91–100, 2017.
- [67] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks, 2012.
- [68] Felix Dobslaw. A parameter-tuning framework for metaheuristics based on design of experiments and artificial neural networks. *World Academy of Science, Engineering and Technology, International Journal of Aerospace and Mechanical Engineering*, 64:213–216, 2010.
- [69] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.
- [70] Aravindh Mahendran and Andrea Vedaldi. Visualizing deep convolutional neural networks using natural pre-images. *International Journal of Computer Vision*, 120:233–255, 2016.
- [71] Ramprasaath R Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? *arXiv preprint arXiv:1611.07450*, 2016.
- [72] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.
- [73] Ori Folger, Livnat Jerby, Christian Frezza, Eyal Gottlieb, Eytan Ruppin, and Tomer Shlomi. Predicting selective drug targets in cancer through metabolic networks. *Molecular systems biology*, 7(1):501, 2011.
- [74] LH Hartwell, JJ Hopfield, and S Leibler. Murray aw. *From molecular to modular cell biology. Nature*, 402, 1999.
- [75] Hiroaki Kitano. Computational systems biology. *Nature*, 420(6912):206–210, 2002.
- [76] Laura M Heiser, Anguraj Sadanandam, Wen-Lin Kuo, Stephen C Benz, Theodore C Goldstein, Sam Ng, William J Gibb, Nicholas J Wang, Safiyyah Ziyad, Frances Tong, et al. Subtype and pathway specific responses to anticancer compounds in breast cancer. *Proceedings of the National Academy of Sciences*, 109(8):2724–2729, 2012.
- [77] Xaqin Castro Dopico, Marina Evangelou, Ricardo C Ferreira, Hui Guo, Marcin L Pekalski, Deborah J Smyth, Nicholas Cooper, Oliver S Burren, Anthony J Fulford, Branwen J Hennig, et al. Widespread seasonal gene expression reveals annual differences in human immunity and physiology. *Nature communications*, 6(1):7000, 2015.
- [78] Imre Vastrik, Peter D'Eustachio, Esther Schmidt, Geeta Joshi-Tope, Gopal Gopinath, David Croft, Bernard de Bono, Marc Gillespie, Bijay Jassal, Suzanna Lewis, et al. Reactome: a knowledge base of biologic pathways and processes. *Genome biology*, 8:1–13, 2007.
- [79] Hiroyuki Ogata, Susumu Goto, Kazushige Sato, Wataru Fujibuchi, Hidemasa Bono, and Minoru Kanehisa. Kegg: Kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 27(1):29–34, 1999.

- [80] Minoru Kanehisa. A database for post-genome analysis. *Trends in genetics: TIG*, 13(9):375–376, 1997.
- [81] K Dolinski, S Dwight, J Eppig, M Harris, D Hill, L Issel-Tarver, A Kasarskis, S Lewis, JC Matese, JE Richardson, et al. Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nat Genet*, 25(1):2529Attri, 2000.
- [82] Ethan G Cerami, Benjamin E Gross, Emek Demir, Igor Rodchenkov, Özgür Babur, Nadia Anwar, Nikolaus Schultz, Gary D Bader, and Chris Sander. Pathway commons, a web resource for biological pathway data. *Nucleic acids research*, 39(suppl\_1):D685–D690, 2010.

## A Manutenzione ed Evoluzione DL-tumor based approach

Nelle sezioni seguenti andremo a mostrare la struttura della repository del progetto e quali sono state le principali modifiche e gli errori riscontrati dapprima per clonare il progetto originale di [1] e poi per mettere in piedi quella che è stata la nostra variante alla rete da loro utilizzata.

### A.1 Pipeline Progetto

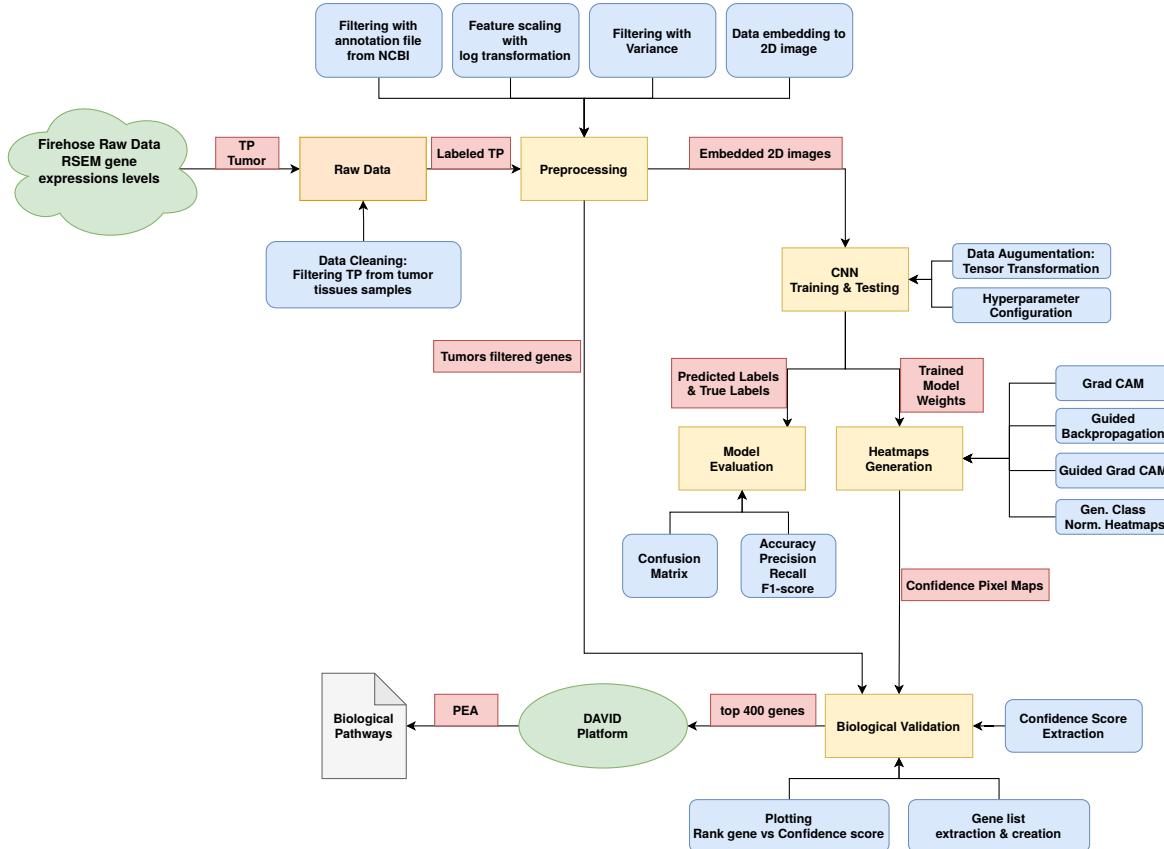


Figura 26: Pipeline del progetto

Il progetto si compone di 6 moduli (come si può vedere in Figura 26) a pagina 75:

**Raw Data** A questo modulo vengono dati in input i dati grezzi scaricati da Firehose. I dati grezzi sono sotto forma di file di testo semplice e divisi per classe tumorale. Il modulo si occupa di aggiungere la label di classe ai sample di una specifica classe tumorale per tutte le classi tumorali coinvolte. In output viene restituito un file csv unico comprensivo dei sample di tutte le classi tumorali sotto test.

**Preprocessing** Tale modulo prende in input il file csv (il dataset) creato dal modulo raw data e il file di annotation con cui vengono confrontati i geni. Vengono effettuati le trasformazioni così come descritto nella sezione 3 e vengono create le immagini (l'output di questo modulo) contenenti i dati preprocessati che saranno date in pasto alla CNN.

**Training and Testing** Questo modulo prende in input le immagini generate dal modulo di preprocessing, ossia i dati d'input sono file .png divisi in una fold di training ed una di testing per 10 volte in base alla 10-cross fold validation. Il modulo applica a queste immagini un servizio di data augmentation per trasformare le immagini in tensori che sono comprensibili dal modello; in seguito configura gli iperparametri necessari al processo di training. In conclusione il modulo avvia una sessione di testing con un test set valutando un'accuracy iniziale per ogni fold. Il

modulo avrà multipli output: i pesi ottenuti dal processo di trainng, le etichette predette e le etichette reali.

**Performance evaluation** Tale modulo prende in input i vettori contenenti le etichette predette dal modello e quelle reali del Test Set. Usando le metriche di valutazione descritte nella sezione 4 valuta le performance del modello che poi vengono salvate in un’immagine .png per la matrice di confusione ed in un file csv per le metriche di valutazione.

**Heatmaps generation** Questo modulo prende in input i file pytorch che contengono i pesi ottenuti dal modulo di training. Da questi ultimi il modulo genera delle heatmap che permettono di interpretare visivamente i processi decisionali interni al modello. Tali mappe saranno prima generate per i singoli campioni dopodiché verranno usate per generare delle heatmap legate all’intera coorte tumorale. L’output di tale modulo saranno le stesse heatmap che contengono i contribuiti di ogni gene alla classificazione.

**Biological validation** Tale modulo prende come input l’insieme delle feature (ossia i geni selezionati nella fase di preprocessing) e i confidence score generati a partire dalle confidence pixel map generate dal modulo di heatmaps generation. L’output, invece, consiste del plot che mostra l’importanza dei geni selezionati e delle liste dei top 400 gene per ogni classe tumorale per ogni fold. Queste ultime vengono poi unite in un’unica lista per classe tumorale e tale lista viene sottomessa alla piattaforma DAVID per eseguire la Pathway Enrichment Analysis.

## A.2 Moduli e Modifiche Apportate al Codice Esistente

Quando abbiamo clonato la repository GitHub di [1] ci siamo accorti subito che purtroppo alcune parti del codice erano mancanti e che la documentazione era davvero molto scarsa o del tutto assente. Abbiamo quindi passato in rassegna i vari moduli per capire come procedere e abbiamo convenuto che la cosa più ovvia da fare fosse quella di seguire il workflow.

### A.2.1 Raw Data

Questo modulo non era stato previsto da [1] ed è stato da noi inserito al fine di comprendere a

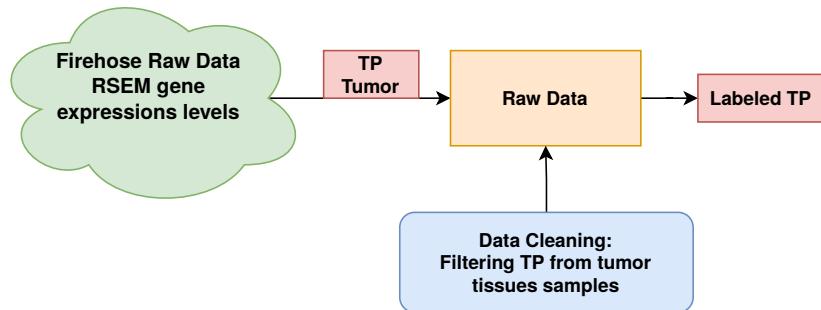


Figura 27: Modulo Raw Data

più basso livello come è stato creato il dataset utilizzato per i vari test. La prima cosa che abbiamo constatato, esaminando i dati grezzi scaricati da FireHose, è che per alcune coorti tumorali il numero di sample dichiarato non corrispondeva perché erano presenti dei sample in più. Analizzando meglio le tabelle riepilogative di FireHose abbiamo capito che tali sample in più erano quelli non classificati come TP (Solid Primary Tumor) e dunque andavano rimossi dal dataset finale in quanto non utili ai fini della classificazione. Per effettuare questo passo di data cleaning, quindi, è stato necessario recuperare gli ID specifici dei sample da rimuovere. Inizialmente avevamo pensato di utilizzare un processo di web scraping per automatizzare il processo ma non è stato possibile procedere in tal senso per via della complessità della pagina html e per via del tempo necessario ad eseguire tale operazione.

Abbiamo quindi deciso di eseguire tale operazione manualmente solo per il caso binario e ternario con lo scopo ultimo di confermare le nostre ipotesi. Dopo aver eseguito tale operazione, abbiamo provveduto ad aggiungere, per ogni sample, le etichette necessarie all'identificazione della corretta coorte tumorale così come fatto anche in [1]. Tali etichette sono numeri interi nel range  $[0, n - 1]$  dove  $n$  è il numero di classi tumorali sotto test. Questo vincolo sulle etichette è presente per via della funzione `CrossEntropyLoss` del modulo di Training e Testing che altrimenti solleverebbe errore.

Il modulo raw data, dunque, si limita ad unire in un unico file csv i sample dei file di dati grezzi delle varie coorti tumorali sotto test preventivamente ripuliti ed etichettati. Nel dataset finale, quindi, compariranno solo i sample mRNASeq contrassegnati con TP (Solid Primary Tumor). La struttura di questo modulo può essere osservata in Figura 27.

### A.2.2 Preprocessing

Nello script di preprocessing erano fornite solo alcune funzione ma perlopiù scarsamente documentate

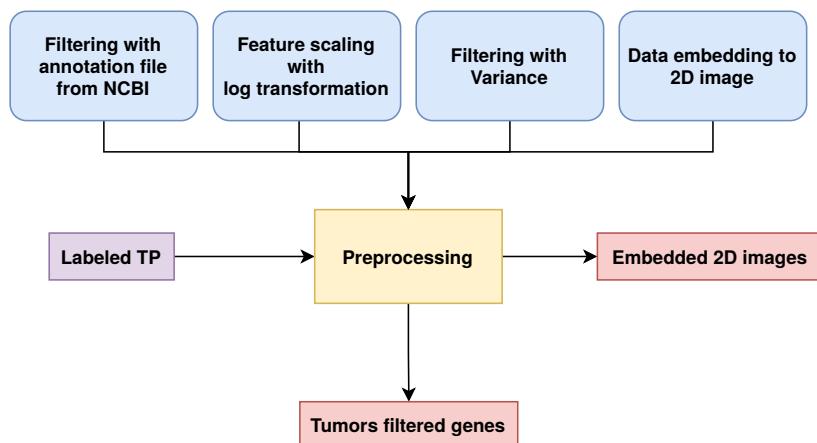


Figura 28: Modulo di Preprocessing

o con documentazione completamente assente. Inoltre non era presente una funzione `main` che consentisse di avviare l'intero processo. Abbiamo, quindi, dapprima studiato nel dettaglio il funzionamento di ogni funzione per comprenderne tutti i passi. Dopodiché è stato facile dedurre anche l'ordine di chiamata, capire quali erano i parametri attuali da passare alle funzioni e scrivere la funzione `main` (poi spezzata in diverse sotto-funzioni) che consente di eseguire tutto il processo. Per semplificarcì il lavoro e per avere un maggior controllo sull'esecuzione abbiamo modificato le firme di alcune delle funzioni al fine poter modificare agilmente parametri che in [1] avevano impostato come costanti (ad esempio la threshold per la selezione delle feature). Una panoramica generale del modulo è presente in Figura 28.

Nello specifico, le funzioni fornite da [1] erano le seguenti:

`feature_selection` che si occupa di estrarre le feature rilevanti (i geni) dal dataset. Dapprima viene fatto un confronto con un file di annotation in modo da rimuovere i geni non pertinenti, poi i geni vengono selezionati in base alla soglia di varianza impostata. La lista restituita è una lista di `official gene symbol`.

`sort_feature` che si occupa di ordinare i geni in base al numero cromosomico, in quanto è più probabile che geni vicini interagiscano tra loro.

`kfold_split` che si occupa di dividere il dataset tramite n-fold cross validation. In questo caso specifico la divisione è fatta solo in dataset di training e dataset di testing per 10 fold e non in training, testing e validation.

`embedding_2d` che si occupa di fare l'embedding dei dati nelle immagini 2D. Durante l'esecuzione di questa funzione così come fornita da [1] veniva sollevato un warning sulla conversione Lossy<sup>10</sup> delle immagini. Abbiamo provveduto a rimuovere tale warning tramite la funzione `img_as_ubyte` di `skimage` che consente di mappare i valori dei pixel direttamente nel range [0, 255].

`over_sampling` per ovviare allo sbilanciamento tra classi tramite SVMSMOTE. Una nota a parte merita questa funzione in quanto è stata aggiornata per poter utilizzare la nuova versione della libreria `imbalanced-learn`. In particolare abbiamo impostato i seguenti parametri:

`sampling_strategy='minority'` al fine effettuare il resampling solo delle classi minoritarie  
`random_state=42` per controllare la randomizzazione dell'algoritmo. 42 è usato come seme del generatore.

`k_neighbors=5` I nearest neighbor utilizzati per definire il vicinato di campioni da utilizzare per generare i campioni sintetici.

`n_job=32` il numero di core della CPU utilizzati durante il ciclo di cross-validation.

Per rendere più agevoli e chiari i passi eseguiti e l'ordine di chiamata delle varie funzioni, sono state da noi aggiunte le seguenti funzioni:

- `run_preprocessing`: che si occupa di avviare effettivamente il preprocessing chiamando le funzioni descritte precedentemente nell'ordine corretto impostando in maniera appropriata i parametri attuali.
- `binary_test_preprocessing`: che si occupa semplicemente di impostare le path corrette per eseguire il test sul caso binario, creare la folder per gli output e passare la threshold corretta per la selezione dei geni.
- `ternary_test_preprocessing`: come la funzione precedente ma per il caso ternario.
- `general_test_preprocessing`: come la funzione precedente ma per il caso generale.
- `main`: per avviare l'intero modulo.

### A.2.3 Training e Testing

Nello script di training e testing non erano fornite funzioni e c'erano le seguenti definizioni di classe:

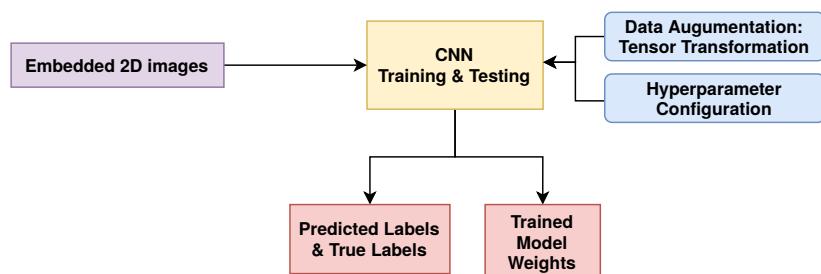


Figura 29: Modulo di Training e Testing

- `TumorDatasetTrain`
- `TumorDatasetTest`
- `ToTensor`

<sup>10</sup>Conversione con perdita di informazioni

- **Net**

I nostri primi sforzi si sono dunque concentrati nel comprendere il funzionamento di tali classi e capire dove venissero utilizzate. Le classi `TumorDatasetTrain` e `TumorDatasetTest` modellano i dataset di training e di testing e si occupano della lettura delle immagini generate nella fase di preprocessing e della loro trasformazione in tensori, la classe `ToTensor` si occupa di trasformare i campioni estratti in tensori da mandare al modello mentre la classe `Net` modella la rete neurale convoluzionale utilizzata. Siccome tali classi saranno utilizzate anche nella fase successiva, abbiamo deciso di metterle in un file diverso, chiamato `classes.py` in modo da richiamarle agilmente e al fine di eliminare la ridondanza e gli errori che ne potevano derivare in seguito alle nostre modifiche. Originariamente le classi `TumorDatasetTrain` e `TumorDatasetTest` facevano riferimento ad un esecuzione su un cluster di nodi, quindi abbiamo deciso di rinominarle in `LocalTumorDatasetTrain` e `LocalTumorDatasetTest` in modo da rendere più evidente che il nostro test è stato eseguito su un'unica macchina e non in un cluster.

I problemi che abbiamo constatato sono stati i seguenti:

- l'API `Variable` utilizzata da [1] era deprecata e dunque il codice è stato modificato per aggiornarlo alla versione corrente di `pytorch`;
- è stato necessario introdurre la One-Hot encoding nella fase di training con la rete Net (seppur gli autori non avevano previsto tale passo) in quanto la funzione di loss sollevava un'eccezione. Con VarNet, invece, non è necessario utilizzarla in quanto la funzione di loss è stata modificata;
- abbiamo dovuto modificare il codice in quanto non è stato possibile utilizzare `pytorch` in combinazione con CUDA così com'era. È stato necessario ridurre drasticamente la batch size (da 500 previsto dagli autori a 90) in quanto la memoria GPU disponibile non era sufficiente.

Si può osservare la struttura di questo modulo in Figura 29.

#### A.2.4 Performance evaluation

Nello script di valutazione delle performance gli autori si limitavano ad eseguire il plotting della

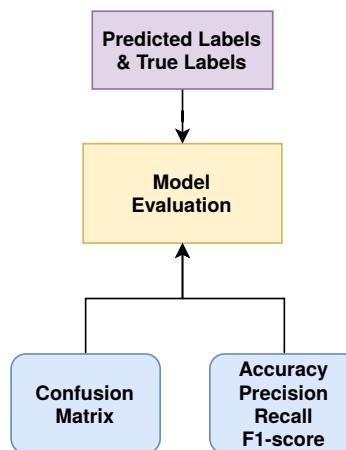


Figura 30: Modulo di Valutazione delle Performance

matrice di confusione e a calcolare le metriche complessive (Accuracy, Precision, Recall e F1-score).

Tale script è stato integrato prevedendo la possibilità di:

- salvare l'accuracy per ogni coorte tumorale
- salvare le righe della matrice di confusione al fine di avere un quadro più chiaro in fase di analisi

- salvare tutti i risultati in un file formato `csv`
- salvare l'immagine della matrice di confusione.

La struttura di questo modulo è presente in Figura 30.

#### A.2.5 Heatmaps generation

Il modulo per la generazione delle heatmap, mostrato in Figura 31, era stato inserito nella repository

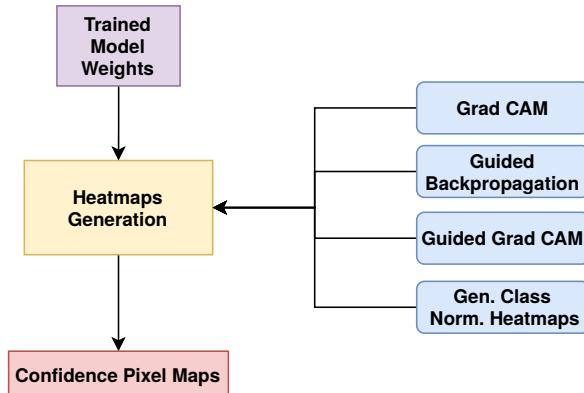


Figura 31: Modulo di Generazione delle Heatmap

GitHub di [1] ed è stato analizzato nel nostro lavoro per poter comprendere la tecnica di visualizzazione Guided Grad-CAM (cfr. 3). Lo script di riferimento era `Guided_GradCam.py` (era presente anche una versione che include l'uso di CUDA per il supporto con GPU). Abbiamo quindi concentrato i nostri primi sforzi sull'analizzare e comprendere le classi elencate di seguito e i metodi messi a disposizione:

- `Net`
- `PropagationBase`
- `GradCAM`
- `BackPropagation`
- `GuidedBackPropagation`

In seguito applicando il workflow descritto nella metodologia applicata, abbiamo compreso l'integrazione, le chiamate e la gerarchia che potevano sussistere tra tali classi. Così come indicato nella precedente lista, la classe `Net`, modella la nostra rete neurale CNN, che in questo caso sarà utilizzata dopo l'addestramento così da poter registrare le mappe di attivazione ed i gradienti che costituiscono i componenti principali per generare Guided Grad-CAM.

La classe `PropagationBase`, modella le operazioni di base per cui sarà usata la classe precedente cioè la forward pass per le feature map è la backward pass per i gradienti. Queste operazioni saranno applicate a tutti i layer della rete. Oltre ciò definisce anche un metodo che modella la base per una hook function<sup>11</sup>. Per concludere tale classe sarà il padre di tutte le altre visto che implementa funzioni vitali allo svolgimento del processo di generazione delle heatmaps. La prima classe specializzata da `PropagationBase` è `Grad-CAM`, questa classe rappresenta la prima vera componente per costruire Guided Grad-CAM cioè la Gradient Classification Activation Map, modellando tutte le operazioni necessarie alla sua generazione. Esegirà dapprima la combinazione lineare delle activation map ed la media ponderata dei gradienti normalizzati, poi questo risultato sarà dato in pasto alla ReLU per

<sup>11</sup>Pytorch usa le hook function per eseguire una funzione durante l'operazione di backward oppure durante il processo di forward pass

ottenere la nostra heatmap. Infine, per ottenere i componenti della combinazione lineare, la classe **Grad-CAM** definirà la hook function del padre per registrare le feature map di ogni layer durante il processo di forward pass ed i gradienti durante il processo di backward.

La classe **BackPropagation**, esprime le operazioni necessarie allo svolgimento dell'operazione di backpropagation con cui si stima l'errore di ogni layer della rete ad ogni epoca della fase di training ereditata dal padre, le componenti per interagire con gli strati della rete, oltre che la possibilità di definire una hook function apposita.

In conclusione la classe **GuidedBackPropagation**, specializzerà la classe **BackPropagation**, introducendo quindi tutti i vincoli necessari per guidare la backpropagation tramite l'input ed il gradiente per cui la classe definirà la hook function ereditata da **BackPropagation**. Quest'ultima si accerterà che tutti i moduli della CNN siano istanze di ReLU, e in caso affermativo tutti gradienti negativi registrati dal modulo stesso vengono rimossi. Da questo lavoro di analisi dello script così com'era sono risaltati alcuni problemi:

1. Si è notato una ridondanza nell'uso della classe **Net** (usata sia qui che nel modulo di training & testing) .
2. La mancanza di una funzione **main**, o di uno script principale dove poter evocare le classi ed i metodi espressi.
3. Il problema di conversione lossy riscontrata nel modulo di Preprocessing, ha reso necessario una gestione delle immagini a contributo nullo, vale a dire con gradiente nullo .
4. Viene sollevato un warning sull'uso errato della notazione array per accedere all'attributo **data** della classe **Tensor**.
5. Un altro problema è legato all'uso scorretto della **register\_backward** per quando si hanno multipli nodi foglia della rete computazionale che conservano il gradiente dell'istanza calcolata con la backward..
6. L'API **Variable** usata [1] per calcolare il gradiente nel rispetto del tensore d'input, risulta essere deprecata..
7. Non risulta presente nello script una funzione o una classe che permetta l'operazione di generare una heatmaps generale che rappresenti la visione completa che ha avuto il modello sull'intera coorte tumorale.

Data la mole di problemi riscontrati abbiamo deciso di eseguire un'operazione di refactoring, generando lo script **genes\_heatmaps.py**. Questo script è strutturato per esprimere i servizi offerti dal modulo e nel frattempo risolve i problemi precedentemente esposti tramite le seguenti funzioni:

- **init\_dataset\_heatmaps**: Questa funzione inizializza il dataset di train che deve essere passato alla rete addestrata.
- **init\_component\_heatmaps**: Il metodo assolve al compito di instanziare gli oggetti che modellano i componenti fondamentali per la generazione di tutte le heatmaps. Qui viene risolto il problema 1, spostando la classe **Net** nel modulo di supporto A.2.7. In seguito vengono richiamato ed inizializzate passandogli il modello addestrato anche le classi **GRAD-CAM** e **GUIDEDBACKPROPAGATION**.
- **generate\_heatmaps**: La funzione si occupa di generare i tensori che rappresentano le regioni di Grad-CAM e i gradienti generati dal processo di GuidedBackPropagation, infine richiama le operazioni di salvataggio per entrambe le heatmaps. Il suddetto modulo risolve il problema 4, sostituendo la relativa notazione con **.data** aggiornato alla versione corrente di Pytorch. Mentre il problema 6 è risolto modificando il codice per aggiornarlo alla versione corrente di Pytorch, sono stati rimossi tutti i riferimenti sostituendoli con l'attributo della classe **Tensor**

.`requires_grad`. Oltre ciò viene risolto anche il problema 5, sostituendo la funzione presente con la funzione `register_full_backward`, che a differenza della precedente versione richiama il corpo della hook function se è solo se vengono computati i tensori d'output prodotti da un layer della rete.

- `save_GradCam`: Questo servizio offre la possibilità di salvare la heatmap gradCam per una sua visualizzazione in un file .png dopo essere stato scalato nel range [0, 255].
- `save_Guided_Gcam`: Questo servizio offre la possibilità di salvare la heatmap GuidedGrad-CAM: qui avviene la moltiplicazione tra le regioni di Grad-CAM ed i gradienti di GuidedBackProgration, dopo ciò i valori sono normalizzati prima di essere salvati in un file .png.
- `filter_null_img_Guided_GradCam`: L'obiettivo di questa funzione è la risoluzione del problema 3, tale funzione si accerta prima che l'immagine in esame abbiano contributo nullo dopodiché elimina quest'ultima. Si applica tale filtro solo per le heatmap di GuidedGrad-CAM.
- `filter_null_img_GradCam`: Come la precedente con l'unica differenza che si applica tale filtro solo per le heatmap di Grad-CAM.
- `gen_avg_norm_GradCam`: L'obiettivo del servizio è la risoluzione del problema 7: esso estrae tutte le heatmap generate dai vari campioni classificati, dopodiché converte il tensore dell'immagine e lo aggiunge ad un unico numpy array in seguito viene eseguita una media su tale array e viene normalizzato nel range [0, 255] prima di essere salvato.
- `gen_avg_norm_GuidedGcam`: Come la precedente con l'unica differenza che l'array che rappresenta il tensore dell'immagine viene convertito a `uint8` per facilitare le operazioni matematiche dato che il canale dei colori viene estratto sulla scala del grigio.
- `main`: Risolve il problema 2, permettendo di invocare tutte le funzioni precedentemente descritte eseguendo quindi il caso generale basta sulla 10-SCV.

#### A.2.6 Biological evaluation

Per il processo di validazione è stato creato uno script ex-novo in quanto in [1] non era previsto.

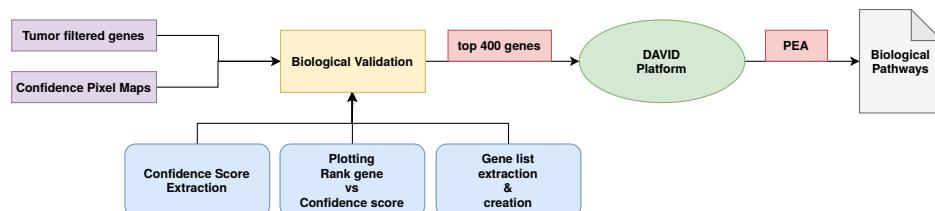


Figura 32: Modulo di Validazione Biologica

Tale script si occupa della generazione del plot che mostra i cambi di intensità nelle heatmap per ogni classe e della generazione delle liste di geni da sottomettere al tool DAVID<sup>12</sup> per procedere alla validazione dei risultati ottenuti tramite Pathway Enrichment Analysis. Il tool DAVID mette a disposizione delle API e dei Web Services (entrambi non aggiornati) ed era nostra intenzione utilizzarli per automatizzare il processo di validazione. Purtroppo non ci è stato possibile in quanto sia le API, sia i Web Services non consentono di utilizzare liste di **official gene symbol**. La validazione è stata dunque eseguita manualmente. L'unico limite che abbiamo dovuto rispettare è stato quello di avere una lista di geni con al massimo 3000 valori. La lista da sottomettere è stata generata nel modo seguente: sono stati dapprima estratti i top 400 gene da ogni folder e poi, tramite la struttura dati `set` di python è stata creata un'unica lista, senza ripetizioni, dei geni significativi di ogni coorte tumorale. È possibile osservare la struttura di questo modulo nella Figura 32.

<sup>12</sup><https://david.ncifcrf.gov/home.jsp>

### A.2.7 Moduli di Supporto

Sono stati da noi creati due moduli di supporto per quelle funzioni e quelle classi richiamate in diverse parti del codice. In [1] avevano optato per la ridefinizione delle classi all'interno dei vari script in cui venivano utilizzate. Noi, invece, al fine di ridurre al minimo gli errori che potevano scaturire a seguito delle nostre modifiche e per ridurre la ridondanza, abbiamo deciso di definirle in due script esterni. Tali script sono:

- `support.py`: in cui vengono definite le funzioni di utilità per la cancellazioni delle immagini a contributo nullo e la funzione che ci consente di ottenere il confidence score necessario all'estrazione dei top gene.
- `classes.py`: in cui vengono definite le varie classi utilizzate sia dallo script di training e testing sia dallo script di generazione delle heatmap.

## A.3 Setup dell'esperimento

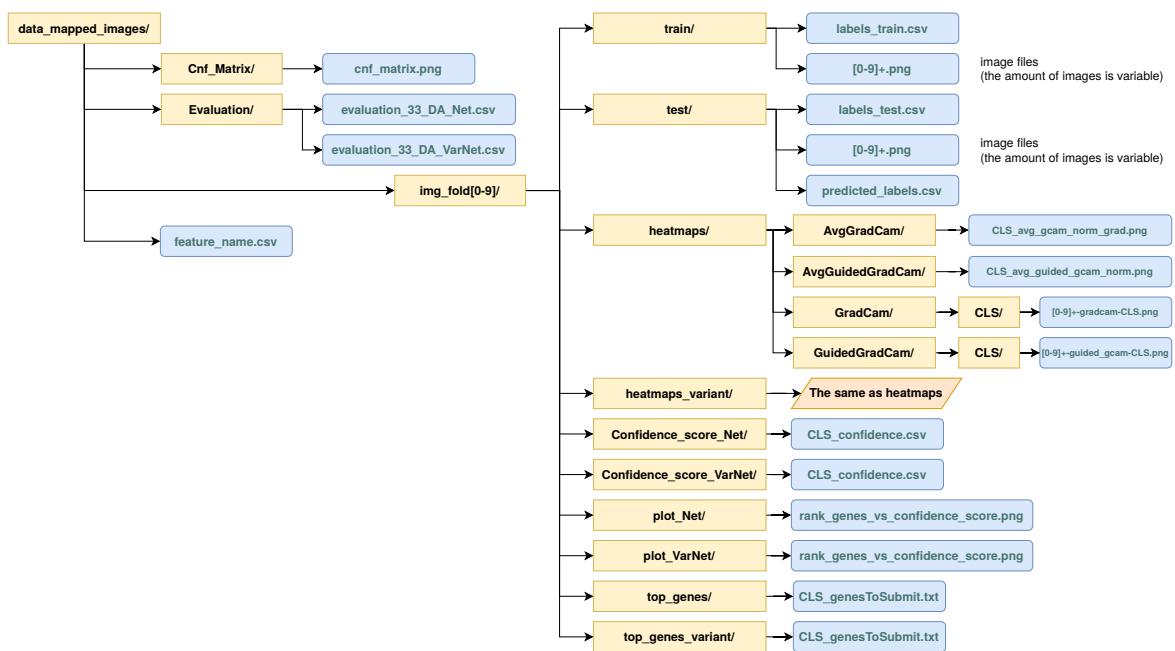
I test sono stati svolti su un server HP con le caratteristiche riportate in Tabella 17 a pagina 83 e i passi svolti sono stati i seguenti:

1. creazione di un virtual environment tramite conda (con python 3.9)
2. installazione delle librerie necessarie al funzionamento del codice (tra parentesi tonda la versione da noi utilizzata):
  - numpy (1.23.5)
  - pandas (1.5.2)
  - pytorch (1.13.1) con CUDA (ver. 11.6)
  - matplotlib (3.5.3)
  - scikit-learn (1.2.1)
  - scikit-image (0.19.3)
  - imbalanced-learn (0.10.1)
  - opencv (4.7.0)

Come IDE abbiamo utilizzato PyCharm Community edition e i risultati sono stati gestiti come riportato in Figura 33 a pagina 84.

Tabella 17: Specifiche Hardware & Software della Macchina Utilizzata

Sistema Operativo	Ubuntu 21.10 (64-bit)
Processore	Intel(R) Xeon(R) Gold 5218 CPU @ 2.30Ghz (x32)
RAM	270 GB
Scheda Grafica	NVIDIA Quadro RTX 4000 (8 GiB di memoria dedicata) [CUDA ver. 11.6]



Where there is CLS, it means "for all CLS in CLASSES"

CLASSES = ['ACC', 'BLCA', 'BRCA', 'CESC', 'CHOL', 'COAD', 'DLBC', 'ESCA', 'GBM', 'HNSC', 'KICH', 'KIRC', 'KIRP', 'LAML', 'LGG', 'LIHC', 'LUAD', 'LUSC', 'MESO', 'OV', 'PAAD', 'PCPG', 'PRAD', 'READ', 'SARC', 'SKCM', 'STAD', 'TGCT', 'THCA', 'THYM', 'UCEC', 'UCS', 'UVM']

Figura 33: Fold di output.