



UNIVERSITÀ DEGLI STUDI DI SALERNO  
DIPARTIMENTO DI INFORMATICA

Strumenti Formali per la Bioinformatica

---

# Classificazione delle Sequenze della Proteina Spike del SARS-CoV-2 Tramite Rappresentazione FCGR e Reti Neurali Convoluzionali

---

*Studenti:*

Girolamo Martina - 0522501680  
Nappi Severino - 0522501681  
Ragozzini Emanuele - 0522502039

*Docenti:*

De Felice Clelia  
Zaccagnino Rocco  
Zizza Rosalba

Aprile 2025

## Abstract

I *coronavirus* sono una sottofamiglia di virus responsabili di diverse patologie negli animali, talvolta in grado di attaccare l'essere umano. Nel 2019, la sindrome respiratoria acuta grave da SARS-CoV-2 ha dato origine a una pandemia globale che ha causato la morte di milioni di persone. Come tutti i virus, moltiplicandosi nell'organismo ospite, il SARS-CoV-2 subisce mutazioni nel suo patrimonio genetico, con particolare frequenza nella proteina *Spike*, rendendola la regione più studiata per la classificazione delle varianti.

Nell'era dell'informazione genomica, l'analisi delle sequenze virali è diventata essenziale per comprendere l'evoluzione del SARS-CoV-2 e identificare le varianti emergenti di rilevanza epidemiologica. In questo contesto, è stato sviluppato un approccio innovativo per la classificazione delle sequenze virali basato sull'utilizzo della *Frequency Chaos Game Representation* (FCGR), che permette di trasformare le sequenze genomiche in rappresentazioni visive idonee all'elaborazione tramite modelli di deep learning.

Le immagini FCGR sono state analizzate mediante reti neurali convoluzionali (CNN), note per la loro capacità di riconoscere pattern complessi all'interno di dati visivi. I modelli addestrati si sono dimostrati in grado di classificare accuratamente le sequenze genomiche secondo i principali *clade* del virus, raggiungendo livelli elevati di accuratezza e F1-score, prossimi al 98%. Una volta validato il modello, è stata condotta un'analisi di interpretabilità utilizzando gli *SHAP values* (SHapley Additive exPlanations), al fine di identificare le regioni dell'immagine maggiormente influenti nella classificazione.

Dal momento che ogni pixel della rappresentazione FCGR è associato a specifici *k-mer* della sequenza genomica, le aree evidenziate dagli SHAP values risultano direttamente riconducibili a pattern nucleotidici potenzialmente significativi a livello biologico. Questa informazione è stata sfruttata per visualizzare in modo preciso e coerente i *k-mer* più rilevanti all'interno dell'immagine, mettendo in evidenza la loro posizione e la loro potenziale funzione discriminante nel contesto delle varianti virali. Tale approccio ha permesso di combinare efficacemente accuratezza predittiva e interpretabilità biologica, aprendo nuove prospettive per l'analisi delle sequenze genomiche mediante tecniche di deep learning.

# Contents

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Virus: caratteristiche biologiche e modalità di trasmissione . . . . .	1
1.2	Covid-19 . . . . .	2
1.2.1	Struttura biologica . . . . .	3
1.2.2	Sintomatologia e modalità di trasmissione . . . . .	3
1.3	Mutazioni del virus . . . . .	4
1.3.1	Le varianti del Covid-19 . . . . .	5
1.4	Algoritmi di codifica delle sequenza amminoacidiche . . . . .	5
1.4.1	Chaos Game Representation . . . . .	6
1.4.2	Frequency Chaos Game Representation . . . . .	6
1.5	Deep Learning e Neural Networks . . . . .	7
1.6	Reti neurali convoluzionali: Residual Neural Network . . . . .	10
1.7	SHAP Values e interpretabilità dei modelli . . . . .	12
<b>2</b>	<b>Metodologia Utilizzata</b>	<b>13</b>
2.1	Raccolta e Preprocessamento dei Dati . . . . .	13
2.2	Progettazione e Addestramento del Modello di Deep Learning . . . . .	15
2.2.1	Architettura del Modello e Parametri di Addestramento . . . . .	16
2.3	Interpretazione del Modello tramite SHAP Values . . . . .	17
<b>3</b>	<b>Risultati e Considerazioni</b>	<b>18</b>
3.1	Risultati con $k = 7$ . . . . .	18
3.2	Risultati con $k = 9$ . . . . .	23
3.3	Impatto della lunghezza dei k-mer . . . . .	28
3.4	Considerazioni generali . . . . .	28

# 1 Introduzione

A partire dal 2020, la diffusione su scala globale del virus SARS-CoV-2 ha profondamente modificato la quotidianità e le abitudini dell'intera popolazione mondiale, dando origine a una crisi sanitaria senza precedenti. Di fronte al continuo emergere di nuove varianti, diventa sempre più importante comprendere a fondo i meccanismi biologici alla base dell'evoluzione del virus, così da poter intervenire in maniera tempestiva ed efficace nella prevenzione, nella diagnosi e nella cura.

In questo contesto, i concetti teorici coinvolti risultano essere molteplici e articolati, spaziando dal livello molecolare, con l'analisi della struttura e delle mutazioni del genoma virale, fino a quello computazionale, dove entrano in gioco rappresentazioni matematiche, algoritmi di apprendimento automatico e tecniche di interpretabilità dei modelli predittivi.

## 1.1 Virus: caratteristiche biologiche e modalità di trasmissione

I virus sono entità biologiche che vivono e si riproducono come parassiti dell'organismo ospitante. Quando non si trovano nella fase dell'infezione o all'interno di una cellula infetta, i virus esistono in forma di particelle indipendenti e inattive. Queste particelle virali, note anche come virioni, sono costituite da due o tre parti:

1. il materiale genetico costituito da DNA o RNA, lunghe molecole che contengono le informazioni genetiche;
2. un rivestimento proteico, chiamato capsid, che circonda e protegge il materiale genetico;
3. e in alcuni casi un involucro esterno formato da uno strato di lipidi, che circonda il rivestimento proteico, detto pericapsid.

Pertanto, i virus vengono trasportati passivamente finché le molecole presenti sul rivestimento virale esterno si attaccano a specifici "recettori" presenti sulla superficie della cellula ospite. Quest'interazione permette alla particella virale di introdurre il proprio materiale genetico all'interno della cellula ospite e di sfruttarne i "sistemi" per produrre le proteine necessarie a costruire nuove copie del virus (*replicazione*). Quindi, ne consegue che la capacità infettiva di ogni virus dipende fortemente dalla specie e dallo specifico tessuto con cui entrano in contatto. Soltanto alcuni virus, in questo senso, possono causare malattie sia nell'uomo, che in alcuni animali (zoonosi), mentre ancor meno sono quelli capaci di infettare sia animali, che vegetali. Dal punto di vista chimico, i virus sono costituiti, da un lato, fino ed oltre il 90% da proteine che svolgono ruoli funzionali e strutturali come, ad esempio, la replicazione degli acidi nucleici, dall'altro, per il restante 1-15%, da DNA o RNA<sup>1</sup>. La classificazione dei virus è principalmente basata sulla natura e la struttura del loro genoma e il loro metodo di replicazione, ma non in base alle malattie che causano<sup>2</sup>. Pertanto, i virus si distinguono in virus a DNA e virus a RNA; sia i virus a DNA che i virus a RNA

---

<sup>1</sup>I virus dotati di pericapsid (o envelope) hanno anche lipidi (10-30%) e glicoproteine.

<sup>2</sup>International Committee on Taxonomy of Viruses (ICTV), 2021 release.

possono avere filamenti singoli o doppi di materiale genetico. I virus a RNA a singolo filamento sono ulteriormente suddivisi in RNA virus a polarità (+) e polarità (-). I virus a RNA a senso positivo possiedono un genoma a singolo filamento di RNA che può fungere da RNA messaggero (mRNA) ed essere direttamente tradotto per produrre una sequenza aminoacidica. I virus a RNA a senso negativo possiedono un genoma a singolo filamento che prima deve sintetizzare un antigenoma a senso positivo complementare, che viene poi utilizzato per produrre RNA messaggero virale. Di solito, i virus a DNA si replicano nel nucleo della cellula ospite mentre i virus a RNA in genere si replicano nel citoplasma. La distinzione tra i virus a DNA e virus a RNA è molto importante: l'RNA è una molecola molto più instabile del DNA e consente ai virus di mutare più facilmente, evolvendo, quindi, ad una velocità decisamente maggiore rispetto a quelli a DNA. Di conseguenza, i virus a RNA riescono ad eludere più facilmente la difesa del sistema immunitario, delle terapie e dei vaccini. Le modalità di trasmissione dei virus sono svariate: possono infettare per via aerea, alimentare, attraverso rapporti sessuali o attraverso vettori (soprattutto insetti come le zanzare). I virus respiratori, come ad esempio l'influenza o il raffreddore, si diffondono attraverso le goccioline di saliva o di secrezioni, prodotte con tosse e starnuti di persone con l'infezione in corso. Altri virus sono in grado di contagiare per via parenterale o attraverso contatti tra le mucose o con sangue e altri fluidi corporei, come ad esempio epatite B, C e HIV. I virus a trasmissione oro-fecale si contraggono con l'ingestione di cibo, acqua o altro contaminato da materiale fecale (es. poliomielite o rotavirus). Esistono agenti virali, poi, di provenienza quasi prettamente alimentare, come epatite A ed epatite E. Anche gli animali domestici possono trasmettere virus: il caso più famoso è probabilmente quello della rabbia. In ogni caso, la presenza di un indebolimento generale dell'organismo o un'immunodepressione può facilitare le infezioni da virus e peggiorarne il decorso.

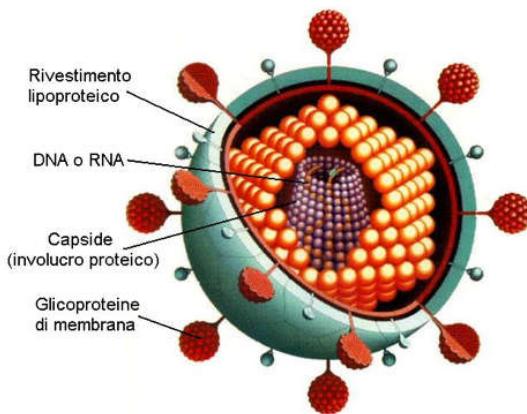


Figure 1: Struttura di un virus

## 1.2 Covid-19

Il SARS-CoV-2 è un ceppo virale della specie coronavirus correlato alla SARS facente parte del genere Betacoronavirus (famiglia Coronaviridae), sottogenere Sarbecovirus ed è il settimo coronavirus riconosciuto in grado di infettare esseri umani.

### 1.2.1 Struttura biologica

I Betacoronavirus sono virus a RNA a singolo filamento positivo che infettano i mammiferi e di cui i pipistrelli e i roditori sono considerati riserve virali<sup>3</sup>. Il termine "betacoronavirus" deriva dal greco antico (bêta, "la seconda lettera dell'alfabeto greco"), e (korónē, "ghirlanda"), che significa corona. Infatti, come altri coronaviruses, SARS-CoV-2 presenta quattro proteine strutturali, note come: proteina *S* (spike o spinula), *E* (involucro), *M* (membrana) e *N* (nucleocapside); la proteina *N* contiene il genoma dell'RNA mentre le proteine *S*, *E* e *M* creano insieme il capsid virale. La proteina spike è quella che permette al virus di attaccarsi alla membrana della cellula ospite e attribuisce la tipica morfologia a "corona". Il genoma del SARS-CoV-2 è formato da 29.881 nucleotidi di cui l'89% identici a quelli del SARS-like-CoVZXC21, diffuso nei pipistrelli, e l'82% identici a quelli del SARS-CoV; tuttavia solo il 40% degli amminoacidi coincide con quelli dei coronavirus legati alla SARS.

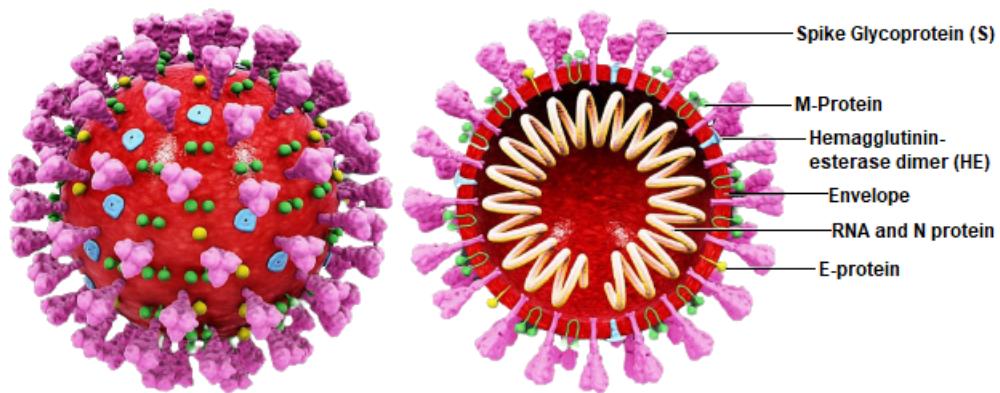


Figure 2: Struttura del SARS-CoV-2

### 1.2.2 Sintomatologia e modalità di trasmissione

Coloro che sono infetti possono risultare asintomatici o presentare alcuni sintomi come febbre, tosse o respiro corto o il più comune raffreddore. Vomito, diarrea o sintomi respiratori superiori (ad es. starnuti, naso che cola, mal di gola), congiuntivite ed eruzioni cutanee sono meno frequenti. La perdita dell'olfatto con la conseguente alterazione del senso del gusto può associarsi agli altri sintomi descritti oppure può rappresentare l'unico sintomo presente. I casi possono, tuttavia, progredire in peggio evolvendo in polmonite, insufficienza multiorgano, fino a portare al

<sup>3</sup>insieme degli organi, o degli organismi, in cui la specie stessa può sopravvivere in forma latente per un periodo prolungato in attesa che si ripresentino le condizioni per aggredire gli organi o gli organismi in cui può proliferare; wikipedia fonte: [https://it.wikipedia.org/wiki/Riserva\\_virale](https://it.wikipedia.org/wiki/Riserva_virale).

decesso nei soggetti più vulnerabili. In generale, i polmoni sono gli organi più colpiti da COVID-19 perché il virus accede alle cellule ospiti tramite l'enzima 2 (ACE2), che è più abbondante nelle cellule alveolari di tipo II dei polmoni. Il COVID-19 si trasmette per via aerea, molto spesso tramite le goccioline respiratorie. Infatti, al fine di limitarne la trasmissione devono essere prese precauzioni, come mantenere la distanza interpersonale di almeno 1,5 metri e tenere comportamenti corretti sul piano dell'igiene personale (lavare e disinfeccare periodicamente le mani, starnutire o tossire in un fazzoletto o nell'incavo del gomito, indossare mascherine e guanti) e ambientale (rinnovare spesso l'aria negli ambienti chiusi aprendo le finestre e mantenere gli ambienti molto puliti). Il periodo di incubazione varia da 2 a 14 giorni con un periodo mediano stimato di incubazione tra i 5 e i 6 giorni. Numerosi sono i test disponibili per assicurare la positività (o negatività) al COVID-19. Per quanto riguarda, invece, la gestione e il trattamento della malattia non esistono ancora farmaci o terapie in grado di apportare un definitivo o effettivo beneficio clinico. Numerosi sono gli studi in merito. Inoltre, sono in fase di sperimentazione oltre 76 vaccini specifici per questa malattia, di cui:

- 40 in fase I (studio delle caratteristiche del farmaco);
- 17 in fase II (studio sulla sicurezza del farmaco);
- 13 in fase III (studio sull'efficacia del farmaco);
- 6 (approvati per uso limitato) in fase IV (studio di farmaco-sorveglianza).

Attualmente, dunque, il trattamento consiste principalmente nell'isolare il paziente per prevenire la diffusione del contagio e nel curare la sintomatologia sviluppata dal singolo individuo.

### 1.3 Mutazioni del virus

Ogni processo evolutivo ha come base delle mutazioni che possono verificarsi casualmente ogni volta che una cellula o un virus si replica. Le mutazioni creano, all'interno di una popolazione, variazioni che consentono alla selezione naturale di amplificare i tratti che aiutano le creature a prosperare. I virus, anche se tecnicamente non considerati forme di vita, mutano ed evolvono infettando le cellule degli organismi ospite e replicandosi. Le modifiche del codice genetico del virus che ne derivano possono aiutarlo a passare più facilmente da un essere umano all'altro oppure ad eludere le difese del sistema immunitario. Le varianti risultanti da questo processo evolutivo sono inserite all'interno di diversi gruppi in funzione delle loro caratteristiche e del grado di preoccupazione suscitato. A questo proposito, è possibile distinguere:

- **Varianti di preoccupazione (VOC):** Queste varianti presentano evidenze di una maggiore trasmissibilità, una severità della malattia aumentata, una significativa riduzione dell'efficacia dei vaccini, o una ridotta capacità di neutralizzazione da parte di anticorpi derivanti da infezioni precedenti o dalla vaccinazione. Le VOC richiedono azioni immediate a livello di salute pubblica, come l'intensificazione della sorveglianza, la modifica dei vaccini o l'introduzione di nuove misure di controllo.

- **Varianti di interesse** (VOI): Sono varianti che presentano mutazioni con caratteristiche genetiche che potrebbero influenzare la trasmissibilità, la gravità della malattia, l'efficacia dei vaccini o la risposta ai trattamenti. Le VOI sono sottoposte a monitoraggio in quanto potrebbero evolvere in varianti più preoccupanti.
- **Varianti sotto monitoraggio** (VUM): Sono varianti con mutazioni che potrebbero rappresentare un rischio, ma per le quali l'evidenza scientifica non è ancora sufficiente per classificarle come VOI o VOC. Queste varianti vengono attentamente monitorate per eventuali segnali che possano indicare un impatto significativo.

### 1.3.1 Le varianti del Covid-19

Con varie decine di migliaia di sequenziamenti del genoma del SARS-CoV-2, dalla fine del 2019 si sono prodotte migliaia di varianti. Tutte queste varianti possono essere raggruppate in gruppi più grandi come lignaggi o clade. Non esiste una nomenclatura standard ma quella più dettagliata e maggiormente condivisa nella letteratura scientifica è la nomenclatura PANGO<sup>4</sup>. Sono, in aggiunta, utilizzate altre nomenclature per raggruppare le varianti:

- GISAID<sup>5</sup>, che a fine 2021 ha codificato 11 clade globali (S, O, L, V, G, GH, GK, GR, GRY, GRA, GV) su 6.900.000 sequenze comunicate.
- Nextstrain<sup>6</sup>, che a fine 2021 ha codificato 23 clade (19A–B, 20A–20J e 21A–H).

Ogni autorità sanitaria, nazionale o sovranazionale, può inoltre istituire un proprio sistema di classificazione. Le principali varianti di interesse (VOI), sono state etichettate dall'Organizzazione Mondiale della Sanità (OMS) come Alfa, Beta, Gamma, Delta e Omicron. Tutte queste varianti, spesso identificate impropriamente anche col nome del paese in cui sono state sequenziate per prime o si sono particolarmente diffuse, rispetto al virus originale di Wuhan, presentano delle consistenti mutazioni che possono tradursi tanto in una maggiore rapidità di diffusione ed elusione del sistema immunitario quanto in nuovi o diversi sintomi o livelli di letalità. In aggiunta, la potenziale comparsa di una variante SARS-CoV-2 potrebbe richiedere la modifica dei vaccini in quanto potrebbe essere moderatamente o completamente resistente alla risposta anticorpale attualmente ottenuta con i vaccini già approvati ed largamente utilizzati.

## 1.4 Algoritmi di codifica delle sequenze amminoacidiche

In letteratura sono stati proposti diversi approcci di Machine Learning ai fini di classificare e/o clusterizzare le diverse specie di Coronavirus, sfruttando le numerose sequenze genomiche raccolte durante la pandemia. Di seguito, si andranno a definire formalmente i presupposti e gli scopi dei principali algoritmi di CGR e FCGR utilizzati per la suddetta finalità.

---

<sup>4</sup><https://cov-lineages.org/index.html>

<sup>5</sup><https://gisaid.org/>

<sup>6</sup><https://nextstrain.org/>

#### 1.4.1 Chaos Game Representation

L'algoritmo di Chaos Game Representation è stato ideato e approfondito dal matematico M. F. Barnsley [1]. Inizialmente, fu sviluppato per altre finalità ma, nel 1990 [2], si propose il suo utilizzo per il sequenziamento del DNA, dando così il via alla sua applicazione nell'ambito della bioinformatica. Infatti, grazie alle sue proprietà, esso è stato utilizzato per diversi scopi come, ad esempio, nel confronto tra sequenze senza allineamento, filogenesi e come codifica per Machine Learning. In sintesi, la finalità principale è quella di creare una mappa che, partendo da una rappresentazione unidimensionale, trasformi una sequenza in una rappresentazione a spazio dimensionale maggiore, tipicamente bidimensionale. In particolare, preso un punto casuale di partenza  $S$  ed un vertice  $V_1$  si calcola la distanza media tra i due. Il risultato è il punto  $P_1$ . Questo processo si ripete utilizzando poi  $P_1$  come punto di partenza al posto di  $S$ . Il secondo punto  $P_2$  è ottenuto come distanza media tra  $P_1$  ed un secondo vertice  $V_2$  casualmente selezionato.

Formalmente l'algoritmo utilizzato per la codifica di sequenze di DNA/RNA è definito nel seguente modo [3]:

Sia  $S = s_1 \dots s_n \in \{A, C, G, T\}$  una sequenza. Quindi, la codifica CGR della sequenza  $S$  è la rappresentazione bidimensionale della coppia ordinata  $(x_n, y_n)$ , definita iterativamente come:

$$(x_i, y_i) = \frac{1}{2}((x_{i-1}, y_{i-1}) + g(s_i)), \quad if \quad i \geq 1 \quad (1)$$

dove  $(x_0, y_0) = (0,0)$  e

$$g(s_i) = \begin{cases} (1, 1) & s_i = A \\ (-1, 1) & s_i = C \\ (-1, -1) & s_i = G \\ (1, -1) & s_i = T \end{cases}$$

Le basi azotate mancanti possono creare problematiche poiché la codifica della funzione  $g(\cdot)$  non è definita in tal caso. Al fine di ovviare a ciò, i ricercatori dell'Università di Milano-Bicocca utilizzano nel loro lavoro, per la classificazione dei clade, la nozione di matrice di frequenza CGR che permette di analizzare i  $k$ -meri anziché stringhe di arbitraria lunghezza.

#### 1.4.2 Frequency Chaos Game Representation

La Frequency Chaos Game Representation è stata sviluppata nel 2000 da Karlin e Brendel ed è stata utilizzata in una varietà di studi in bioinformatica, tra cui la predizione delle funzioni delle proteine e l'identificazione di regioni codificant nei genomi batterici.

L'FCGR è considerata una variante/astrazione del Chaos Game Representation (CGR). La frequenza di ogni possibile parola di lunghezza  $k$  nella sequenza di DNA o proteine viene calcolata e rappresentata in un diagramma tridimensionale. Il grafico è costituito da una griglia di punti che rappresentano le possibili parole di lunghezza  $k$  e la loro posizione dipende dalle frequenze relative di ciascuna parola nella sequenza.

L'algoritmo di Frequency Chaos Game Representation usato per la codifica di sequenze di DNA/RNA è definito formalmente nel modo seguente [3]:

Sia  $S = s_1 \dots s_n \in \{A, C, G, T, N\}^*$  una sequenza e sia  $k$  un intero. Allora la matrice di frequenza del Chaos Game Representation, in breve FCGR, della sequenza  $S$  è una matrice bidimensionale  $F = (a_{i,j})$ ,  $1 \leq i,j \leq 2^k$  con  $(i,j) \in N$  di dimensione  $2^k \times 2^k$ . Per ciascun  $k$ -mer  $b \in \{A, C, G, T\}^k$ , abbiamo un elemento  $a_{i,j}$  nella matrice  $F$  che è uguale al numero di occorrenze di  $b$  come sottostringa di  $s$ . Inoltre, la posizione  $(i, j)$  di tale elemento è calcolata come segue:

$$i = 2^k - \lceil 2^{k-1}(x+1) \rceil + 1$$

$$y = \lceil 2^{k-1}(y+1) \rceil$$

dove  $(x, y)$  è la codifica CGR per il  $k$ -mer  $b$ .

Si noti che l'FCGR è definito per una sequenza di DNA con nucleotidi sconosciuti, indicati con  $N$ , dove  $k$ -mers con una  $N$  sono semplicemente esclusi nel processo di conteggio, mentre la codifica CGR è ben definita solo quando tutti i nucleotidi sono noti.

## 1.5 Deep Learning e Neural Networks

Il Deep Learning, in italiano “apprendimento profondo”, è un segmento di ricerca del Machine Learning (*ML*) e del più ampio ambito di studi dell’intelligenza artificiale (*AI*), in cui gli algoritmi di reti neurali artificiali sono modellati per funzionare come l’apparato cerebrale umano, imparando da grandi quantità di dati. Questa tecnologia, permette di migliorare l’automazione e le attività analitiche. La maggior parte delle persone utilizza il deep learning ogni giorno durante la navigazione su Internet o l’uso dei telefoni cellulari senza, tuttavia, riconoscerne la presenza. Tra le numerose altre applicazioni, il deep learning viene utilizzato, infatti, per generare didascalie per i video YouTube, per eseguire il riconoscimento vocale sui telefoni e gli altoparlanti intelligenti, per consentire il riconoscimento facciale delle fotografie e per consentire la guida autonoma delle automobili. Formalmente il deep learning comprende un insieme di tecniche basate su reti neurali artificiali organizzate in diversi strati, dove ogni strato calcola i valori per quello successivo affinché l’informazione venga elaborata in maniera sempre più completa.

In particolare, le reti neurali artificiali (in inglese *artificial neural network*, abbreviato in *ANN* o anche come *NN*) sono reti strutturate come neuroni artificiali che consentono di implementare azioni complesse tipiche della cognizione umana.

La relazione tra input e output sulla quale si poggia il metodo di addestramento del deep learning non è di tipo diretto, come invece accade nell’apprendimento automatico, quanto piuttosto costituito da livelli intermedi nascosti. Posto ciò, è possibile identificare già una prima distinzione tra reti neurali e deep learning. Infatti, una rete neurale di base potrebbe avere uno o due livelli nascosti, mentre una rete di deep learning potrebbe avere decine o addirittura centinaia di livelli intermedi. L’aumento del numero di livelli e di nodi diversi può aumentare la precisione di una rete. Tuttavia, più livelli possono anche significare che un modello richiederà più parametri e risorse computazionali. Formalmente, il concetto di rete neurale si pone

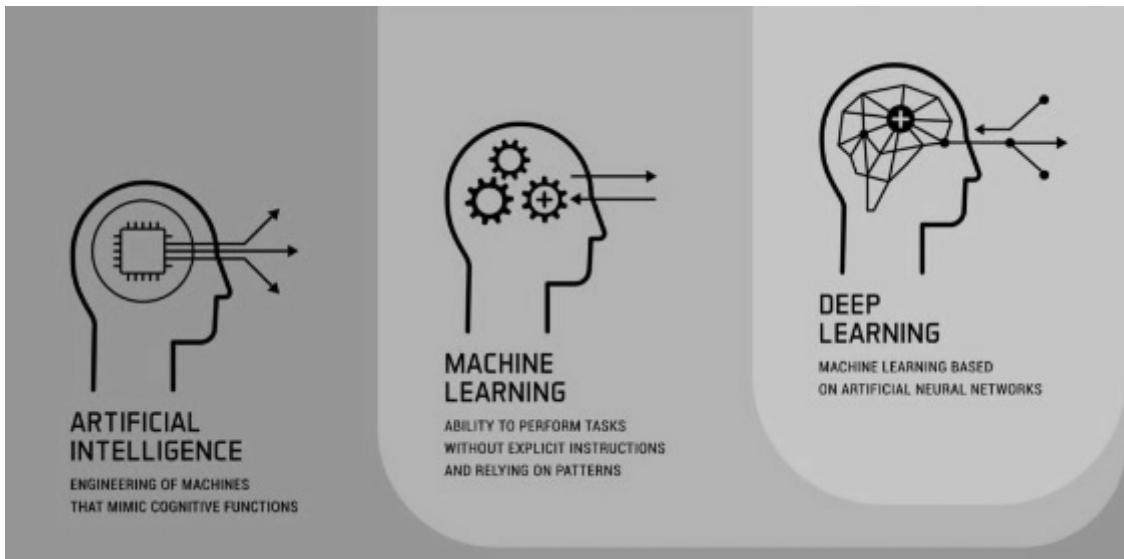


Figure 3: Rami dell’Intelligenza Artificiale

perché una funzione  $f(x)$  è definita come una composizione di altre funzioni  $G(x)$ , che possono a loro volta essere ulteriormente definite come composizione di altre funzioni. Graficamente, questo può essere comodamente rappresentato come una struttura di reti, con le frecce raffiguranti le dipendenze tra variabili:

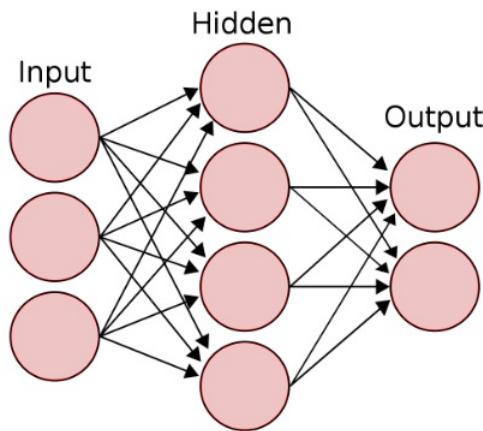


Figure 4: Rete Neurale Artificiale

Nel dettaglio, la figura 5 esemplifica una decomposizione della funzione  $f$ , con dipendenze tra le variabili indicate dalle frecce. Queste possono essere interpretate in due modi:

1. Il primo punto di vista è di tipo funzionale: l’ingresso  $x$  è trasformato in un vettore a 3-dimensioni, che viene poi trasformato in un vettore bi-dimensionale  $g$ , che è poi finalmente trasformato in  $f$ . Questo punto di vista è più comunemente riscontrato nel contesto dell’ottimizzazione.
2. Il secondo punto di vista è di tipo probabilistico: la variabile casuale  $F=f(G)$  dipende dalla variabile casuale  $G=g(H)$ , che dipende da  $H=h(X)$ , che dipende

a sua volta dalla variabile casuale  $X$ . Questo punto di vista è più comunemente riscontrato nel contesto dei modelli grafici.

I due punti di vista sono in gran parte equivalenti. In entrambi i casi, per questa particolare architettura di rete, i componenti dei singoli strati sono indipendenti l'uno dall'altro (ad esempio, le componenti di  $g$  sono indipendenti l'una dall'altra, dato il loro ingresso  $h$ ). Questo, naturalmente, permette un certo grado di parallelismo nella costruzione del sistema. Reti di questo tipo vengono comunemente chiamate "feedforward", perché il loro è un grafo aciclico diretto, ovvero, in altre parole, non esiste alcun percorso che inizi da un nodo e ritorni allo stesso nodo.

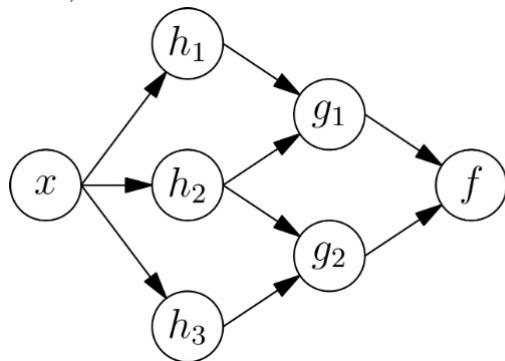


Figure 5: Diagramma di dipendenza di una rete neurale "feedforward"

Nel campo delle reti neurali, il deep learning è stato introdotto attraverso la definizione delle cosiddette reti neurali profonde (deep neural network, *DNN*). Il principio di funzionamento è lo stesso delle reti neurali classiche, con la differenza che esse sono caratterizzate da un numero elevato di strati di calcolo, a loro volta basati su un numero altrettanto elevato di livelli, tale da richiedere uno sforzo computazionale significativo per ottenere uno scenario simile alle connessioni neurali proprie del cervello umano.

Il deep learning, quindi, classifica le informazioni attraverso livelli di reti neurali, che hanno un set di input che riceve dati grezzi. Ad esempio, se una rete neurale è addestrata con immagini di uccelli, può essere utilizzata per riconoscere le immagini degli uccelli. Più livelli consentono risultati più precisi, come distinguere un corvo da una cornacchia, rispetto a distinguere un corvo da un pollo. Le reti neurali consolidate, che si basano su algoritmi di deep learning, hanno diversi livelli nascosti tra i nodi di input e di output, il che significa che sono in grado di eseguire classificazioni di dati più complesse. In base a ciò, risulta intuitivo che un algoritmo di deep learning deve essere addestrato con grandi moli di dati, e che, proporzionalmente, più dati riceve, più accurato sarà nel predire correttamente l'output. Riprendendo l'esempio precedente, sarà necessario alimentare l'algoritmo con migliaia di immagini di uccelli prima di poter classificare accuratamente nuove immagini di uccelli.

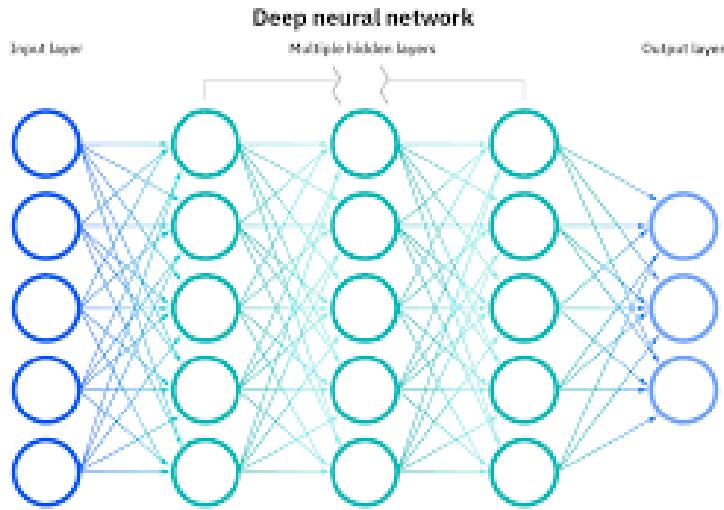


Figure 6: Deep Neural Network

## 1.6 Reti neurali convoluzionali: Residual Neural Network

Le reti neurali convoluzionali (*ConvNet* o *CNN*) si distinguono dagli altri tipi di reti neurali per le loro prestazioni superiori con le immagini, input vocali e segnali audio. Contengono tre tipi di livello principali, ovvero:

1. **Livello convoluzionale:** è l'elemento costitutivo principale di una CNN ed è il punto in cui si verifica la maggior parte dei calcoli. Richiede pochi componenti, ovvero dati di input, un filtro e una mappa delle funzioni. Questo processo è noto come convoluzione.
2. **Livello di pooling:** definito anche sottocampionamento, esegue la riduzione della dimensionalità, ovvero del numero di parametri nell'input. In modo simile al livello convoluzionale, l'operazione di pooling applica un filtro sull'intero input, ma la differenza è che a questo filtro non è associato alcun peso. Invece, il kernel applica una funzione di aggregazione ai valori all'interno del campo ricettivo, popolando l'array di output.
3. **Livello completamente connesso (FC, Fully-Connected):** esegue l'attività di classificazione in base alle funzioni estratte tramite i livelli precedenti e i loro diversi filtri.

Dopo la prima architettura basata su CNN (*AlexNet*) che ha vinto il concorso ImageNet 2012, tutte le successive architetture sono state sviluppate basandosi sull'utilizzo di più livelli di una rete neurale profonda al fine di ridurre il tasso di errore. Questo procedimento, in realtà, funziona solo per un numero contenuto di livelli. Infatti, quando il numero di livelli aumenta, si presenta un problema abbastanza comune nell'ambito dell'apprendimento profondo vale a dire il cosiddetto gradiente<sup>7</sup> che svanisce/esplode (Vanishing/Exploding gradient). Tale problematica fa sì che il gradiente diventi 0 (Vanishing) o troppo grande (Exploding) ovvero

---

<sup>7</sup>è la variazione di lunghezza che una grandezza subisce da un punto all'altro dello spazio lungo

che, all'aumentare del numero dei livelli, aumenta anche il tasso di errore del set di addestramento e del set di test.

Al fine di risolvere il problema del gradiente che svanisce/esplode, la Residual Neural Network, abbreviata Resnet, introduce il concetto di Residual Blocks o blocchi residui. In questa architettura, infatti, si utilizza una tecnica chiamata skip connection che collega le attivazioni di un livello a ulteriori livelli saltandone alcuni intermedi, formando così un blocco residuo. I resnet vengono realizzati impilando insieme questi blocchi residui.

Il vantaggio dell'aggiunta della skip connection è che se un livello compromette le prestazioni dell'architettura, quest'ultimo verrà saltato grazie alla regolarizzazione<sup>8</sup>. Quindi, questo si traduce nell'addestrare una rete neurale molto profonda senza i problemi causati dal gradiente che svanisce/esplode. Formalmente, in un modello di rete neurale multi-livello, si consideri una sottorete formata da un certo numero (ad esempio 2 o 3) di layers sovrapposti, come mostrato nella figura 7. Si indichi la funzione eseguita da questa sottorete come  $H(x)$ , dove  $x$  costituisce l'input della sottorete stessa.

Il risultato,  $y$ , può allora essere rappresentato come:

$$y = F(x) + x$$

La funzione  $F(x)$  è spesso rappresentata da una moltiplicazione di matrici interlacciata con funzioni di attivazione e operazioni di normalizzazione. Questa sottorete è denominata "blocco residuo". Una rete residua profonda viene costruita sovrapponendo una serie di blocchi residui.

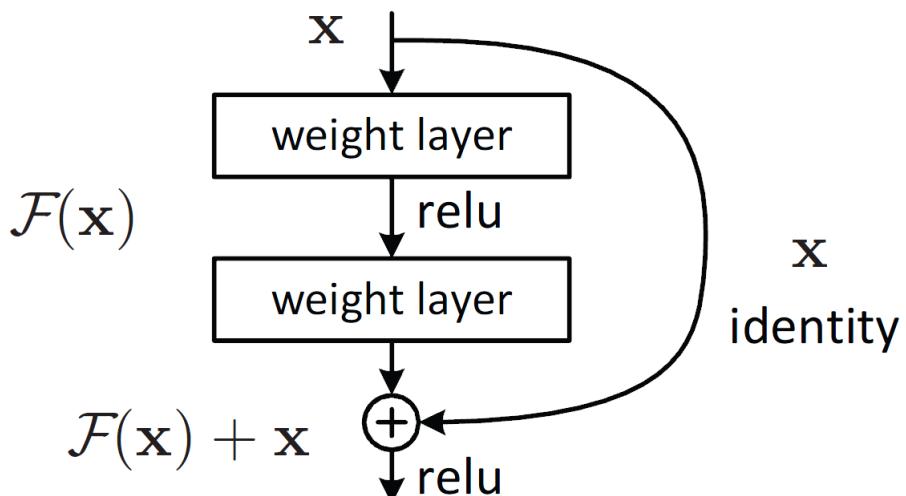


Figure 7: Un blocco residuale in una ResNet. Si saltano due livelli.

---

una certa direzione.

<sup>8</sup>la regolarizzazione implica l'introduzione di ulteriore informazione allo scopo di risolvere un problema mal condizionato o per prevenire l'eccessivo adattamento; fonte wikipedia: [https://it.wikipedia.org/wiki/Regolarizzazione\\_\(matematica\)](https://it.wikipedia.org/wiki/Regolarizzazione_(matematica))

## 1.7 SHAP Values e interpretabilità dei modelli

Uno degli aspetti più rilevanti nello sviluppo di modelli di deep learning, in particolare nel contesto bioinformatico, è la possibilità di comprendere quali caratteristiche influenzino le decisioni prese dal modello. Questa esigenza diventa ancora più cruciale quando si lavora con dati biologici, in cui l'interpretabilità può tradursi in conoscenze utili dal punto di vista medico e molecolare.

A tal fine, una delle tecniche più diffuse ed efficaci è rappresentata dagli **SHAP values** (SHapley Additive exPlanations), un approccio basato sulla teoria dei giochi cooperativi. L'idea alla base di SHAP è quella di assegnare a ciascun input una contribuzione numerica alla predizione effettuata dal modello, analoga al contributo di un giocatore al risultato finale in un gioco cooperativo. In particolare, ogni feature (o regione dell'immagine, nel caso di input visivi come le FCGR) riceve un valore che quantifica quanto essa influisce, positivamente o negativamente, sull'output del modello.

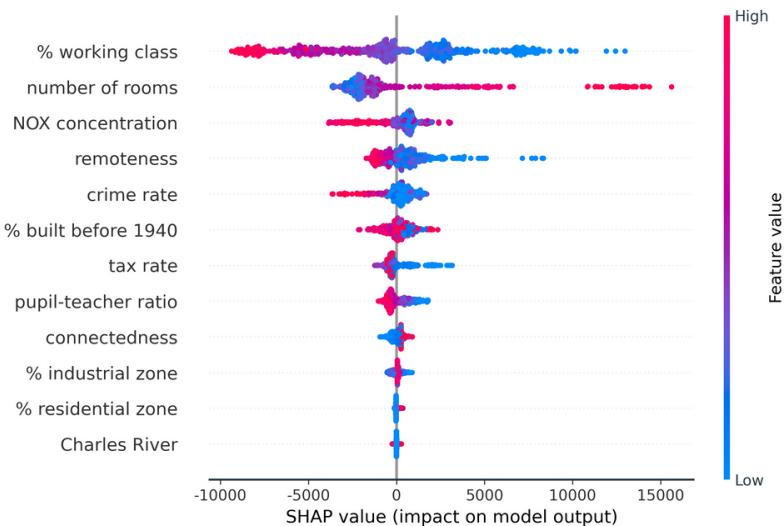


Figure 8: Esempio di SHAP Values

Applicando questa tecnica alle immagini ottenute dalle sequenze genomiche del SARS-CoV-2, è stato possibile visualizzare in modo chiaro e interpretabile le aree che il modello considera maggiormente informative. Le zone con contributo positivo alla predizione vengono evidenziate, ad esempio, in rosso.

Questa visualizzazione, oltre a migliorare la comprensione del comportamento del modello, consente anche un'analisi biologica più profonda: i k-mer localizzati nelle aree più influenti possono infatti essere messi in relazione con mutazioni note o con tratti distintivi di specifici lignaggi o clade. In alcuni casi, le zone evidenziate si sono rivelate coerenti con mutazioni chiave già note nella letteratura scientifica, confermando la capacità del modello non solo di classificare correttamente, ma anche di apprendere pattern biologicamente significativi.

In definitiva, l'integrazione degli SHAP values all'interno del flusso di lavoro ha permesso di colmare il divario tra l'efficacia predittiva del deep learning e la necessità di trasparenza e interpretabilità, offrendo uno strumento potente sia per la validazione dei modelli che per l'esplorazione di nuove ipotesi biologiche.

## 2 Metodologia Utilizzata

Il presente lavoro si propone di sviluppare un approccio innovativo per la classificazione dei clade del SARS-CoV-2, integrando tecniche di bioinformatica, rappresentazioni grafiche delle sequenze genomiche e modelli di deep learning. La metodologia adottata si articola in diverse fasi, che vanno dalla raccolta e preprocessamento dei dati genomici fino all'interpretazione dei risultati ottenuti tramite tecniche di explainable AI.

Di seguito, vengono descritte in dettaglio le principali fasi del processo.

### 2.1 Raccolta e Preprocessamento dei Dati

Il dataset utilizzato per lo svolgimento dell'esperimento è stato ottenuto dalla piattaforma internazionale **GISaid** (Global Initiative on Sharing All Influenza Data), una delle principali risorse per la raccolta e condivisione di sequenze genomiche virali, in particolare relative al SARS-CoV-2.

Ai fini dell'analisi, sono state selezionate esclusivamente **sequenze genomiche complete**, ovvero sequenze la cui lunghezza fosse superiore a **29.000 basi azotate**, al fine di garantire un'informazione nucleotidica il più possibile esaustiva e priva di frammentazioni. È stata inoltre applicata una soglia di **alta copertura**, includendo solo quelle sequenze che presentano una percentuale di nucleotidi ambigui ("bad nucleotides", rappresentati con il carattere 'N') compresa tra lo 0.05% e l'1%. Questa scelta è stata motivata dalla volontà di bilanciare la qualità dei dati con la necessità di disporre di un numero sufficiente di campioni per l'addestramento del modello.

Un ulteriore filtro ha riguardato l'origine del campione, considerando **esclusivamente le sequenze provenienti da ospiti umani** (campo Host uguale a "Human"), al fine di mantenere l'omogeneità biologica del dataset e focalizzare l'analisi sul comportamento del virus all'interno della popolazione umana.

Infine, per la fase sperimentale, si è scelto di concentrare l'attenzione su due specifici clade virali: il clade **V** e il clade **GK**, selezionati per la loro rilevanza epidemiologica e distribuzione nel tempo, e per valutare l'efficacia del modello nel discriminare tra varianti con caratteristiche genetiche distinte.

### Generazione delle Immagini tramite Frequency Chaos Game Representation (FCGR)

La *Frequency Chaos Game Representation (FCGR)*, descritta già in precedenza, è una tecnica bioinformatica avanzata impiegata per rappresentare graficamente sequenze di DNA, sfruttando un approccio non lineare che rende visibili proprietà strutturali e frattali delle sequenze nucleotidiche. A differenza della Chaos Game Representation (CGR) classica, che costruisce un'immagine tracciando i singoli punti su coordinate bidimensionali, la FCGR fornisce una discretizzazione dello spazio in una griglia predefinita, ottenendo una matrice che riflette la **frequenza di comparsa dei k-mer** nella sequenza.

Il processo si fonda su una suddivisione ricorsiva dello spazio in **quattro quadranti principali**, ciascuno dei quali rappresenta una base azotata (Adenina, Citosina,

Guanina, Timina). A partire dal centro dell’immagine, ogni nucleotide viene mappato muovendosi verso il quadrante corrispondente e riducendo progressivamente la scala del sistema di riferimento. Ripetendo questo processo per l’intera sequenza, si ottiene una distribuzione spaziale dei k-mer, codificata come valori numerici che indicano la frequenza relativa all’interno di ciascun quadrante.

La rappresentazione risultante è una **matrice bidimensionale di dimensione**  $2^k \times 2^k$ , dove  $k$  indica la lunghezza del k-mer analizzato. Il numero di celle della griglia (quadranti) è dato dalla formula:

$$q = 2^{2k}$$

Ad esempio, una griglia  $16 \times 16$  (ovvero 256 quadranti) rappresenta tutti i possibili k-mer di lunghezza 4. Al contrario, per determinare il valore di  $k$  a partire da un dato numero di quadranti  $q$ , si può applicare la formula inversa:

$$k = \frac{\log_2(q)}{2}$$

Questa trasformazione consente di visualizzare le sequenze in **scala di grigi**, dove le aree più scure indicano una maggiore frequenza di determinati k-mer, mentre le aree chiare o bianche evidenziano k-mer rari o assenti. In tal modo, le “zone vuote” (*buchi bianchi*) nell’immagine rappresentano sequenze sottorappresentate o mancanti, permettendo di individuare pattern ricorrenti, regioni altamente conservate oppure segnali di eventi biologicamente rilevanti, come duplicazioni geniche o motivi ripetitivi.

La potenzialità della FCGR risiede nella sua capacità di fornire una visione d’insieme compatta e interpretabile della composizione di una sequenza, risultando particolarmente utile in combinazione con tecniche di **deep learning** per l’analisi comparativa e la classificazione di varianti genomiche su larga scala.

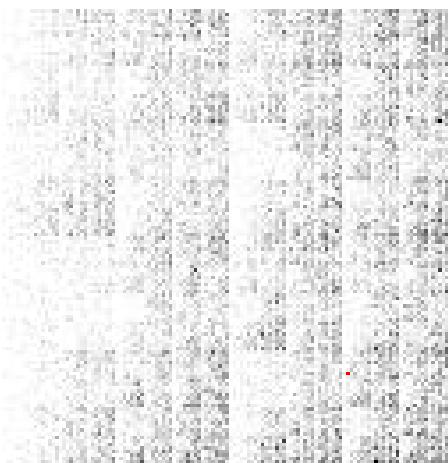


Figure 9: Esempio di immagine generata tramite Frequency Chaos Game Representation (FCGR) di una sequenza genomica del SARS-CoV-2. Le aree più scure indicano la presenza ricorrente di specifici k-mer, mentre i “buchi bianchi” evidenziano sequenze sottorappresentate.

## 2.2 Progettazione e Addestramento del Modello di Deep Learning

Le reti neurali convoluzionali, note come *Convolutional Neural Networks* (CNN), rappresentano una classe di modelli particolarmente efficaci nel riconoscimento e nella classificazione di immagini. La loro architettura è progettata per apprendere gerarchie di caratteristiche visive: i livelli iniziali rilevano pattern semplici (es. bordi e angoli), che vengono progressivamente combinati in strutture sempre più complesse nei livelli successivi.

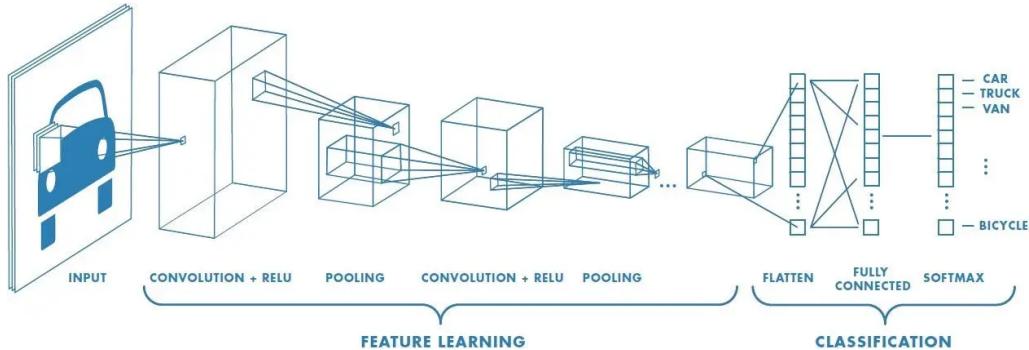


Figure 10: Esempio di architettura di un CNN

Questo approccio riflette la natura compositiva delle immagini reali ed è alla base delle elevate prestazioni delle CNN. I principi fondamentali su cui si basano queste reti sono: **campi recettivi locali**, **pesi condivisi** e **operazioni di pooling**.

A differenza delle reti completamente connesse, nelle CNN ciascun neurone è connesso solo a una porzione ristretta dell'immagine di input, chiamata *campo recettivo locale*. Tale campo può essere immaginato come una finestra che si muove sull'immagine: ogni posizione della finestra corrisponde a un neurone del layer successivo, responsabile dell'elaborazione di quella specifica regione.

Ogni filtro convoluzionale è rappresentato da una matrice di pesi (o kernel) che viene fatta scorrere sull'immagine. In ciascuna posizione, i valori dei pixel vengono moltiplicati per i corrispondenti pesi del filtro e sommati, eventualmente con l'aggiunta di un bias. Il risultato è una *feature map*, che indica la presenza e l'intensità di una determinata caratteristica. Ogni filtro produce la propria feature map, e l'insieme di tutte queste costituisce l'output del layer convoluzionale.

Per migliorare l'efficienza computazionale e ridurre la dimensionalità, vengono impiegati gli **strati di pooling**, tipicamente con l'operazione di *max-pooling*. Questa tecnica seleziona il valore massimo da una regione locale della feature map, mantenendo le informazioni più rilevanti e rendendo la rete più robusta a piccole traslazioni o distorsioni dell'immagine.

Successivamente, l'output convoluzionale viene “appiattito” tramite uno strato **Flatten** e passato a uno o più *fully connected layer*. Tali livelli permettono di integrare le caratteristiche estratte nei layer precedenti e produrre la classificazione finale.

### 2.2.1 Architettura del Modello e Parametri di Addestramento

Per la classificazione dei **Clade V** e **Clade GK**, è stata progettata una rete neurale convoluzionale ad hoc. Il processo di addestramento ha seguito le seguenti fasi:

- **Caricamento e Preprocessing dei Dati:** i dati consistono in un file CSV con i nomi delle immagini e le etichette corrispondenti ai clade. Le immagini sono state:
  - Caricate da directory separate;
  - Ridimensionate a  $128 \times 128$  pixel;
  - Convertite in array numerici monodimensionali.
- Le etichette testuali sono state trasformate in formato *one-hot encoded*.
- **Definizione dell'Input:** il modello accetta in input immagini di dimensione  $128 \times 128$ , e prevede due classi in output (classificazione binaria).
- **Architettura della Rete:**
  - Più strati Conv2D con 4 filtri e kernel di dimensione  $8 \times 8$  (cioè  $2^{\text{level}}$ );
  - **Stride** impostata a 8;
  - Funzione di attivazione ReLU;
  - Normalizzazione con **BatchNormalization** dopo ogni layer convoluzionale;
  - Pooling con finestra  $2 \times 2$  (operazione di **MaxPooling2D**).
- **Strato Finale:**
  - **Flatten** per convertire l'output 2D in un vettore 1D;
  - Strato **Dense** con 2 unità e attivazione **softmax** per produrre una distribuzione di probabilità sulle due classi.
- **Compilazione e Addestramento:**
  - Funzione di perdita: **binary\_crossentropy**;
  - Ottimizzatore: **Adam** con learning rate iniziale pari a 0.01;
  - **Batch size:** 64.

Questa configurazione è stata scelta per bilanciare efficienza computazionale, accuratezza e capacità di generalizzazione del modello, garantendo un'adeguata distinzione tra le immagini appartenenti ai due clade. Inoltre, al fine di ridurre la complessità computazionale e facilitare l'addestramento della rete, le immagini inizialmente caricate in formato RGB (quindi con 3 canali di colore) sono state convertite in **scala di grigi**. Questa trasformazione ha permesso di ridurre la rappresentazione da un array tridimensionale  $128 \times 128 \times 3$  a una matrice bidimensionale  $128 \times 128$ , corrispondente a un solo canale. In termini computazionali, ciò equivale a passare da una rappresentazione tridimensionale a una più semplice, mantenendo le informazioni essenziali per la classificazione tra *Clade V* e *Clade GK*.

## 2.3 Interpretazione del Modello tramite SHAP Values

Una volta completato l’addestramento del modello, è fondamentale interrogarsi su come esso giunga alle sue decisioni. Questo passaggio è essenziale per garantire trasparenza e interpretabilità, specialmente in ambiti critici come l’analisi genomica, dove la comprensione dei meccanismi decisionali può rivelare informazioni biologiche rilevanti.

Per questo motivo, è stato adottato un approccio interpretativo basato sugli **SHAP Values** (SHapley Additive exPlanations). Gli SHAP rappresentano una tecnica di interpretabilità del modello fondata sulla teoria dei giochi cooperativi. Il principio alla base consiste nel quantificare l’importanza di ogni feature in input, nel nostro caso i singoli pixel dell’immagine FCGR (Frequency Chaos Game Representation), rispetto alla predizione finale effettuata dal modello.

Nel contesto della nostra rete neurale convoluzionale, applicare i valori SHAP alle immagini FCGR consente di determinare visivamente quali aree dell’immagine – e quindi, indirettamente, quali regioni della sequenza genomica – abbiano avuto un impatto maggiore nella classificazione tra i due *clade*, ovvero *Clade V* e *Clade GK*.

Poiché ogni posizione nell’immagine FCGR rappresenta un particolare  $k$ -mer (una sottosequenza di lunghezza  $k$ ), i valori SHAP non si limitano a fornire una visione "pixel-based", ma assumono anche una chiara interpretazione biologica: evidenziano i  $k$ -mers più informativi per la rete nella distinzione tra i due *clade*. Le zone dell’immagine che presentano valori SHAP particolarmente elevati corrispondono infatti a combinazioni di nucleotidi ricorrenti e distinctive che caratterizzano specifici gruppi virali.

In tali visualizzazioni, le regioni con valore positivo elevato (colori caldi) indicano aree che hanno favorito la predizione verso una determinata classe.

Tale approccio ha permesso non solo di verificare la coerenza interna del modello – osservando pattern simili all’interno delle immagini appartenenti allo stesso *clade* – ma anche di correlare direttamente le regioni ad alta attivazione con particolari sequenze di  $k$ -mers. Questo risultato suggerisce che la rete neurale ha effettivamente imparato a riconoscere segnali distintivi nel genoma virale, associati a particolari strutture o regioni genomiche caratteristiche dei *clade V* e *GK*.

In definitiva, l’integrazione degli SHAP Values nel processo di interpretazione ha fornito un potente strumento non solo per aprire la “scatola nera” della rete neurale, ma anche per estrarre conoscenza biologica utile dai dati, restituendo una mappa visiva che collega pixel,  $k$ -mers e predizione finale.

### 3 Risultati e Considerazioni

Al termine della fase di progettazione e addestramento del modello, è stato avviato un processo sistematico di sperimentazione volto a valutare l'efficacia della rete neurale convoluzionale sviluppata, nonché a esplorarne le potenzialità in termini di accuratezza e generalizzazione nel compito di classificazione tra le sequenze appartenenti al *Clade V* e al *Clade GK*.

Le attività sperimentali si sono svolte con l'obiettivo non solo di confermare la validità del modello, ma anche di comprendere a fondo l'impatto delle diverse scelte progettuali – tra cui la dimensione del parametro  $k$ , la struttura dell'architettura convoluzionale, e le tecniche di preprocessamento delle immagini FCGR – sul comportamento complessivo del sistema di classificazione. In particolare, l'attenzione è stata rivolta alla valutazione dell'accuratezza, della robustezza e della stabilità del modello attraverso test iterativi su diversi dataset di input, mantenendo sempre il focus sull'interpretabilità dei risultati, resa possibile tramite l'analisi con i valori SHAP.

Per garantire tempi di addestramento contenuti e prestazioni computazionali adeguate, tutte le sperimentazioni sono state eseguite su una macchina ad alte prestazioni, appositamente configurata per supportare carichi di lavoro intensivi legati al deep learning.

Le sperimentazioni sono state condotte variando il valore di  $k$ , parametro fondamentale nella generazione delle immagini FCGR. In particolare, si è scelto di confrontare le prestazioni del modello su due configurazioni distinte: una con  $k = 7$ , e una con  $k = 9$ . Questa scelta ha permesso di osservare come l'aumento della granularità nell'analisi dei  $k$ -mers influenzi la qualità della rappresentazione visiva dell'immagine, la capacità della rete di cogliere pattern significativi e, in ultima analisi, la precisione della classificazione.

Tutte le immagini, inizialmente in formato RGB, sono state convertite in scala di grigi durante il processo di preprocessing. Questa trasformazione ha ridotto lo spazio dei dati da una rappresentazione tridimensionale a una monodimensionale, semplificando la struttura delle immagini senza penalizzare la capacità del modello di riconoscere caratteristiche significative. Tale scelta ha contribuito a migliorare l'efficienza computazionale, riducendo la complessità del modello e i tempi di addestramento, mantenendo al contempo un buon livello di accuratezza nei risultati.

Infine, i risultati ottenuti sono stati analizzati sia in termini quantitativi (metriche di classificazione), sia in termini qualitativi, mediante le heatmap generate con i valori SHAP, che hanno permesso di osservare la forte correlazione tra le regioni attivate della rete e i  $k$ -mers più influenti per la classificazione. Questo ha fornito ulteriori conferme dell'efficacia dell'approccio adottato, sia sotto il profilo computazionale che in termini di interpretabilità biologica del modello.

#### 3.1 Risultati con $k = 7$

L'utilizzo di  $k = 7$  ha permesso la costruzione di immagini FCGR di dimensione  $128 \times 128$ , offrendo un buon equilibrio tra risoluzione e quantità di informazione. Durante il processo di addestramento del modello, è stata monitorata l'accuratezza

sui dati di validazione per valutare la capacità di generalizzazione dell’architettura proposta. Come si può osservare nella Figura 11, il valore della *Validation Accuracy* ha mostrato un rapido incremento sin dalle primissime epoche, raggiungendo valori superiori al 98% già dopo pochi cicli di addestramento. Successivamente, il comportamento dell’accuratezza si è mantenuto relativamente stabile, con leggere fluttuazioni fisiologiche dovute alla natura stocastica dell’ottimizzazione.

In particolare, si nota come a partire dalla decima epoca l’accuratezza tenda a stabilizzarsi attorno al 99%, senza evidenti segni di overfitting o decadimenti improvvisi. Questo andamento conferma la bontà del modello, suggerendo una corretta capacità di apprendere le caratteristiche rilevanti dei dati senza incorrere in fenomeni di sovra-allenamento. Inoltre, la presenza di minime variazioni nelle ultime epoche evidenzia un processo di training regolare e ben controllato.

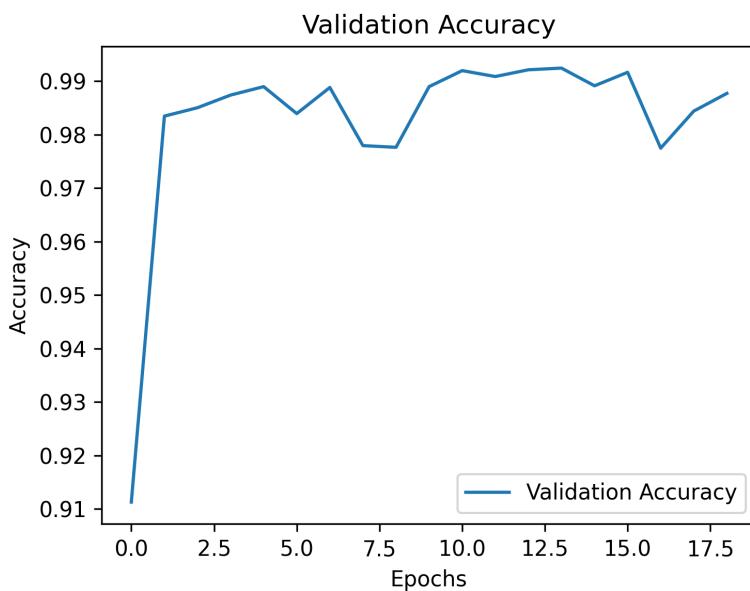


Figure 11: Andamento della Validation Accuracy durante le epoche di addestramento.

Le prestazioni ottenute dal modello, con accuracy pari al 94%, sono riportate nella Tabella 1. Le metriche si riferiscono al comportamento globale del classificatore, e risultano valide per entrambi i clade considerati.

Per quanto riguarda il **clade V**, il modello ha raggiunto una *Precision* pari a 0.90, una *Recall* molto elevata di 0.99 e un *F1-Score* complessivo di 0.95, calcolati su un totale di 1042 sequenze. Questi valori indicano che il modello è stato estremamente efficace nel riconoscere correttamente la maggior parte delle sequenze appartenenti a questo clade, con un bassissimo numero di falsi negativi, come evidenziato dall’elevatissimo valore di Recall.

Nel caso del **clade GK**, invece, si osserva una *Precision* molto alta pari a 0.99, a fronte di una *Recall* leggermente inferiore (0.88) e un *F1-Score* di 0.93, su un totale di 941 sequenze testate. Ciò suggerisce che il modello tende a commettere meno falsi positivi rispetto al clade V, pur avendo una leggera difficoltà nel riconoscere tutte le sequenze appartenenti a questo gruppo.

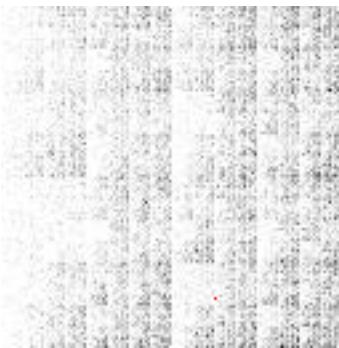
Nel complesso, il modello ha dimostrato ottime performance su entrambi i cladi, con valori di Precision, Recall e F1-Score sempre superiori al 90%, confermando l'efficacia dell'approccio adottato per la classificazione delle sequenze.

Table 1: Metriche di classificazione del modello per  $k = 7$

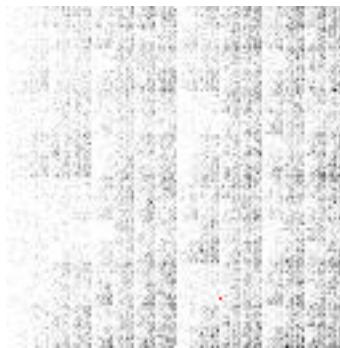
Test	Precision	Recall	F1-Score	Support
Clade V	0.90	0.99	0.95	1042
Clade GK	0.99	0.88	0.93	941

### Visualizzazione SHAP per il Clade V

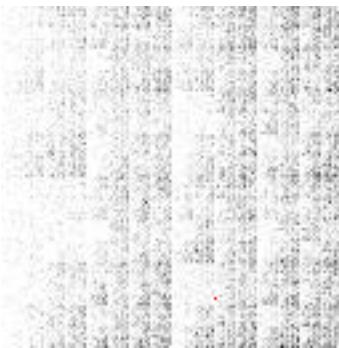
Le immagini FCGR riportate di seguito evidenziano le regioni più influenti per il Clade V in relazione ai 24, 50, 100 e 200  $k$ -mers più importanti.



(a) Top 24  $k$ -mers



(b) Top 50  $k$ -mers



(c) Top 100  $k$ -mers



(d) Top 200  $k$ -mers

Figure 12: Tutti i marker presenti nel clade V al variare di N.

Tutte le immagini in figura 12 non presentano variazione, pertanto, nella figura seguente si è evidenziata la regione interessata:

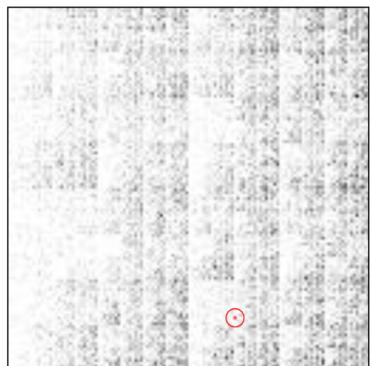
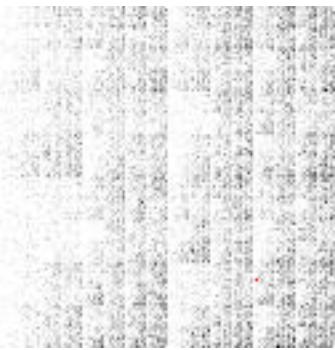


Figure 13: Visualizzazione in scala di grigi dell’immagine generata a partire dai  $k$ -mers del Clade V con  $k = 7$ . Il cerchio rosso evidenzia la posizione del  $k$ -mer più ricorrente e influente secondo l’analisi dei valori SHAP: ***GGTCAT***.

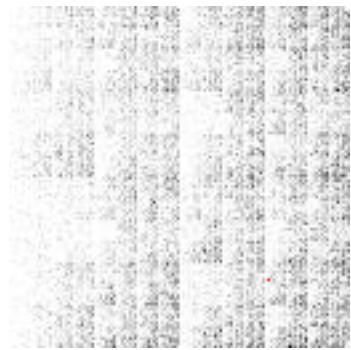
È stato osservato che l’applicazione dell’approccio Reverse & Complement alle sequenze non ha comportato miglioramenti significativi nelle prestazioni del modello.

#### Visualizzazione SHAP per il Clade GK

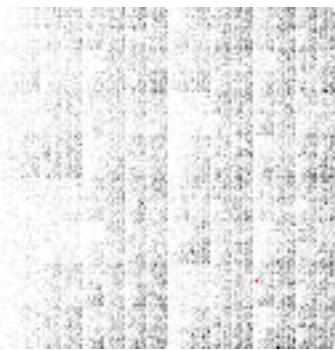
Le immagini FCGR riportate di seguito evidenziano le regioni più influenti per il Clade GK in relazione ai 24, 50, 100 e 200  $k$ -mers più importanti.



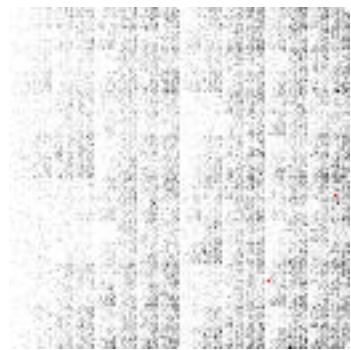
(a) Top 24  $k$ -mers



(b) Top 50  $k$ -mers



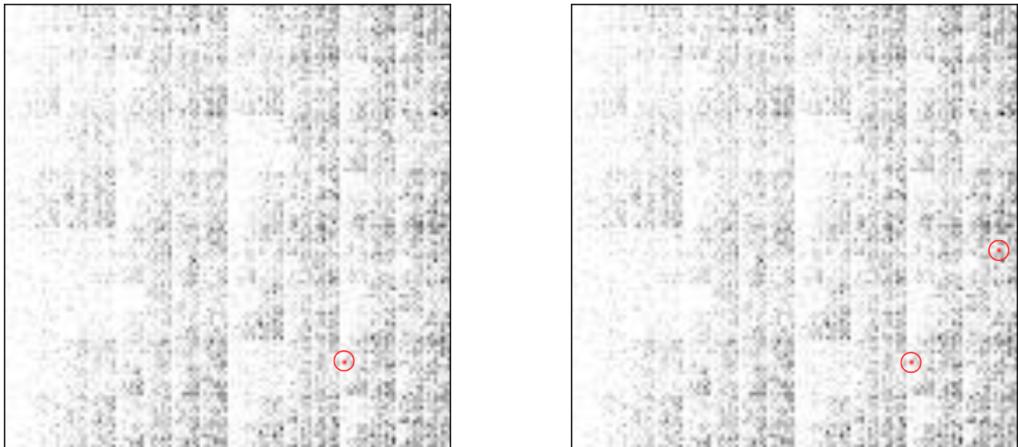
(c) Top 100  $k$ -mers



(d) Top 200  $k$ -mers

Figure 14: Tutti i marker presenti nel clade GK al variare di N.

Come si può osservare nella figura 14, le immagini relative ai valori  $n = 24, 50$  e  $100$  non mostrano variazioni rilevanti nella distribuzione dei valori SHAP. Tuttavia, un cambiamento significativo emerge con  $n = 200$ , motivo per cui nella figura successiva è stata messa a paragone la regione maggiormente influenzata in ambo le variazioni.



(a) Top 24, 50, 100  $k$ -mers

(b) Top 200  $k$ -mers

Figure 15: Confronto tra i marker più rilevanti: (a) mostra i marker individuati per il Clade GK, caratterizzati da una maggiore densità in regioni specifiche dell’immagine: **ACACCTT**; (b) evidenzia i marker associati al Clade GK, con  $N$  pari a 200: **ACACCTT**, **TCAGGGT**.

Per il *Clade GK* è stato applicato anche l’approccio di Reverse & Complement alle sequenze, il quale ha permesso l’identificazione di nuovi marker rilevanti. In particolare, per valori di  $N = 24, 50, 100$ , sono stati individuati due marker ricorrenti: **ACACCCT** e **ACACCTT**. Al crescere del numero di sequenze considerate ( $N = 200$ ), i marker rilevanti risultano raddoppiati, includendo anche **TCAGGGT** e **TAACATC**, oltre ai precedenti. Questo risultato suggerisce che l’incremento del campione, combinato con la strategia di reverse complement, può evidenziare regioni informative altrimenti trascurate.

### 3.2 Risultati con $k = 9$

Con  $k = 9$ , l’immagine FCGR ottenuta ha dimensioni pari a  $512 \times 512$ , incrementando significativamente il livello di dettaglio disponibile per la rete.

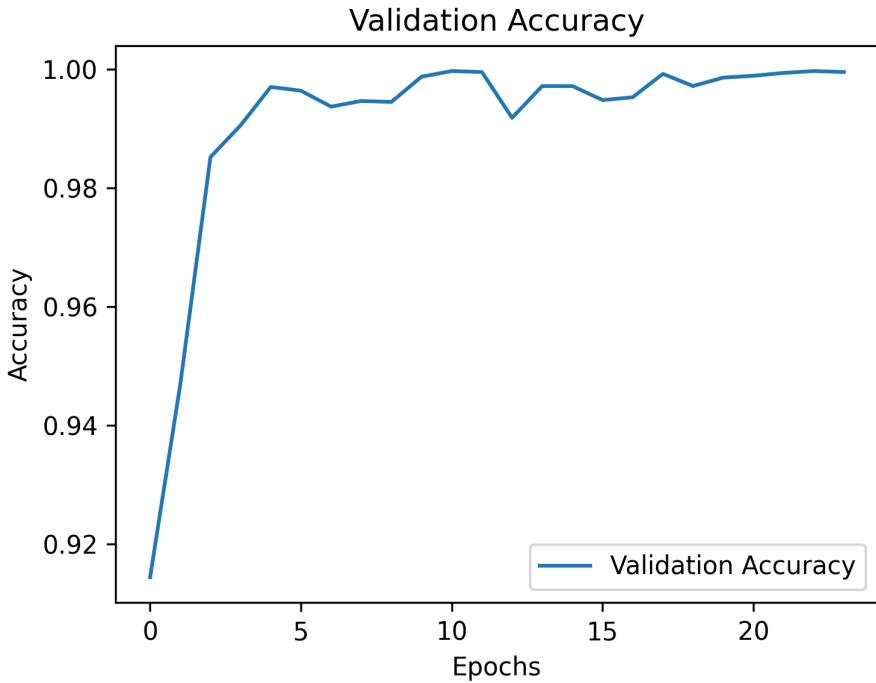


Figure 16: Andamento della *Validation Accuracy* durante l’addestramento con  $k = 9$ .

La Figura 16 mostra l’andamento della **Validation Accuracy** durante il processo di addestramento del modello, utilizzando  $k = 9$  nella fase di estrazione dei marker.

Dal grafico si osserva un rapido incremento dell’accuratezza già nelle prime epoche, passando da un valore iniziale di circa 0,91 fino a superare la soglia dello 0,98 in sole 3 epoche. Successivamente, il modello continua a migliorare la propria capacità predittiva, raggiungendo un’accuratezza prossima al 100% dopo circa 5 epoche. Nei cicli successivi, il valore di *Validation Accuracy* si stabilizza, mantenendosi costantemente molto elevato, con leggere oscillazioni trascurabili, indice di un modello ben generalizzato e privo di fenomeni evidenti di overfitting.

Nel complesso, l’andamento della curva testimonia l’elevata capacità del modello di apprendere rapidamente le caratteristiche distintive dei dati, ottenendo prestazioni estremamente solide già in una fase iniziale del training.

Le metriche globali ottenute dal modello, con accuracy pari al 92%, sono riassunte nella Tabella 2 e, analogamente al caso precedente, restano invariate tra i due clade.

Per il **Clade V**, il modello raggiunge un valore di *Recall* perfetto pari a 1, indicando che è stato in grado di identificare correttamente tutte le istanze appartenenti a questa classe. Tuttavia, la *Precision* è leggermente inferiore (0,87), suggerendo che alcuni esempi predetti come appartenenti a Clade V erano in realtà appartenenti all’altra classe. Il risultato complessivo di *F1-Score* per Clade V è pari a 0,93.

Per quanto riguarda il **Clade GK**, si osserva una situazione complementare: la *Precision* è massima (1), mentre il *Recall* si attesta a 0,84. Questo significa che tutte le istanze classificate come Clade GK sono corrette, ma il modello non riesce a riconoscere il 16% circa delle vere istanze di questa classe. L’F1-Score risultante è

di 0,91.

Nel complesso, il modello dimostra elevate prestazioni su entrambe le classi, anche se con una leggera asimmetria nella capacità di riconoscere tutti gli esempi di Clade GK.

Table 2: Metriche di classificazione del modello per  $k = 9$

Test	Precision	Recall	F1-Score	Support
Clade V	0.87	1	0.93	1042
Clade GK	1	0.84	0.91	941

### Visualizzazione SHAP per il Clade V

Per quanto riguarda il *Clade V*, i risultati ottenuti con  $k = 9$  non sono stati particolarmente incoraggianti. In tutte le prove effettuate, infatti, non è emerso alcun marker degno di nota che potesse essere considerato rilevante o ricorrente all'interno delle sequenze analizzate. Anche cercando di migliorare l'analisi con l'aggiunta dell'approccio Reverse & Complement, non si sono osservati cambiamenti significativi o nuove informazioni utili. Questo suggerisce che, almeno per questo valore di  $k$ , non sembrano esserci pattern chiari o regioni delle sequenze in grado di distinguere efficacemente il Clade V.

### Visualizzazione SHAP per il Clade GK

Le immagini FCGR riportate di seguito evidenziano le regioni più influenti per il Clade GK in relazione ai 24, 50, 100 e 200  $k$ -mers più importanti.

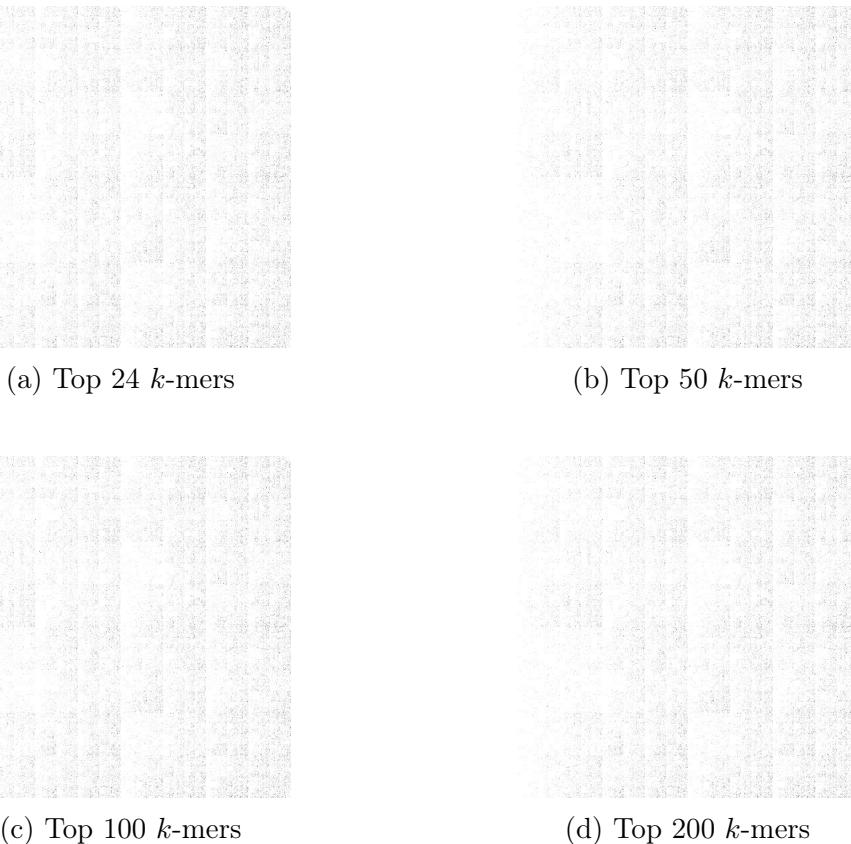


Figure 17: Tutti i marker presenti nel clade GK al variare di  $N$ .

Nel caso del *Clade GK*, l'analisi condotta utilizzando  $k = 9$  ha restituito risultati decisamente più interessanti rispetto a quanto osservato per il *Clade V*. In particolare, ciò che è emerso con chiarezza è che ogni variazione del parametro  $N$  ha portato all'identificazione di marker differenti, a testimonianza di come la dimensione della sequenza analizzata possa influenzare in maniera significativa l'output del modello. In altre parole, al crescere di  $N$ , cambiano anche le regioni della sequenza che risultano maggiormente influenti per la classificazione. I diversi marker individuati per ciascun valore di  $N$  sono mostrati nella figura seguente, dove è possibile visualizzare chiaramente come le regioni influenti si spostino o si modifichino.

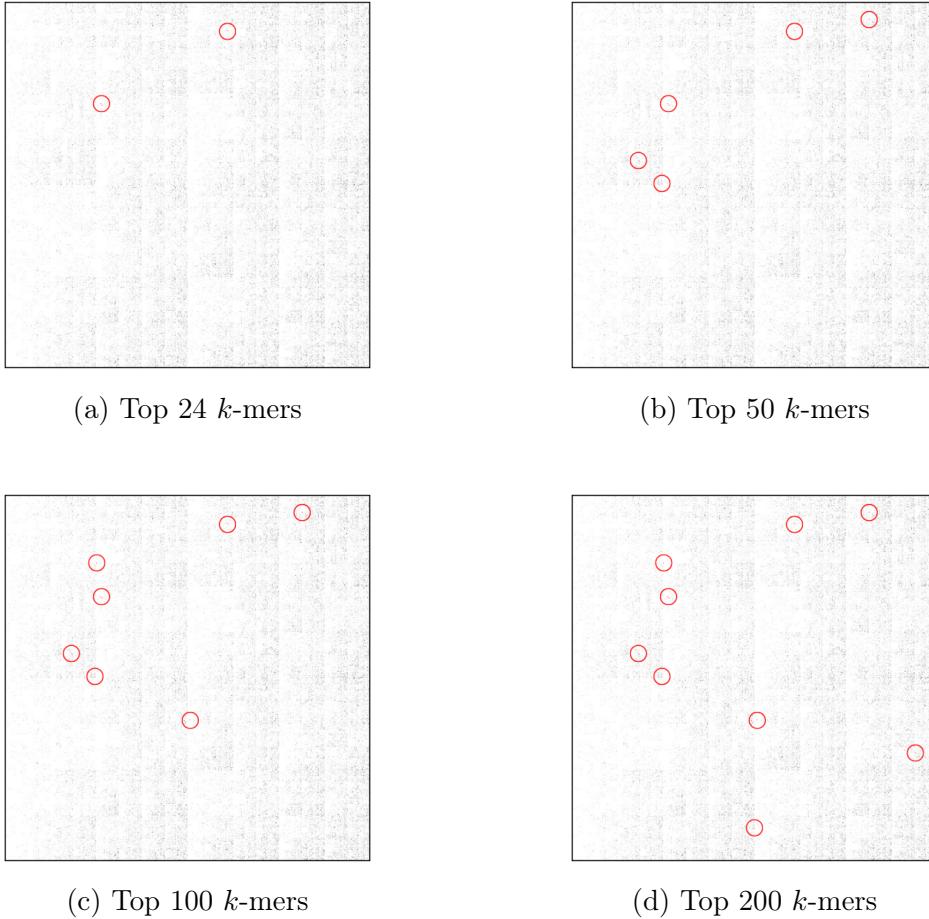


Figure 18: Evoluzione dei marker più influenti individuati per il *Clade GK* con  $k = 9$ , al variare del parametro  $N$ . (a) Per  $N = 24$ , emergono tre marker principali: **TTTACCCCTC**, **TCTACCCCTC** e **TTTCGTCCG**, (b) aumentando a  $N = 50$ , si osservano tre nuovi marker che si aggiungono ai precedenti: **TCTTTCTAC**, **TCTTTTTAC** e **TTTGTCCGG**, (c) con  $N = 100$  vengono individuati due ulteriori marker: **AGCAAACCT** e **GTTTTGTCC**, infine, (d) per  $N = 200$ , il modello riesce ad evidenziare altri due marker distintivi: **AGGGTGTAA** e **AAACCTTGT**, portando a dieci il numero complessivo di sequenze ricorrenti associate a questo clade.

L’incremento di  $N$  ha dunque permesso di affinare progressivamente la rilevazione dei marker, mostrando come una maggiore profondità nella rappresentazione della sequenza permetta di catturare segnali più complessi e potenzialmente informativi.

L’applicazione dell’approccio Reverse & Complement alle sequenze relative al clade GK, con  $k = 9$ , ha portato all’individuazione di un insieme crescente di marker al variare del valore di  $N$ .

Nel dettaglio, i marker individuati per ciascun valore di  $N$  sono i seguenti:

- Per  $N = 24$ :  
**TTTACCCCTC**, **TCTACCCCTC**, **TTTCGTCCG**
- Per  $N = 50$ :  
**TTTACCCCTC**, **TCTACCCCTC**, **TTTCGTCCG**, **TCTTTCTAC**,

## TCTTTTAC, TTTGTCCGG

- Per  $N = 100$ :  
**TTTACCCTC, TCTACCCTC, TTTCGTCCG, TCTTTCTAC, TCTTTTAC, TTTGTCCGG, AGCAAACCT, GTTTGTCC, TTTGCTACC**
- Per  $N = 200$ :  
**TTTACCCTC, TCTACCCTC, TTTCGTCCG, TCTTTCTAC, TCTTTTAC, TTTGTCCGG, AGCAAACCT, GTTTGTCC, TTTGCTACC, AGGGTGTAA, AACCTTGT**

Come si può osservare, all'aumentare della dimensione della sequenza in input, cresce anche il numero di marker distinti individuati. In particolare, per  $N = 200$  si raggiunge il livello massimo di dettaglio, con un totale di undici marker, evidenziando l'efficacia dell'approccio Reverse & Complement nell'enfatizzare segnali informativi altrimenti latenti nelle sequenze genomiche.

### 3.3 Impatto della lunghezza dei k-mer

- **k=7**: Il modello ha mostrato un buon equilibrio tra precisione e recall per entrambi i clade. In particolare, per il Clade V, si è osservata una precisione di 0.90 e una recall di 0.99, mentre per il Clade GK, la precisione è stata di 0.99 e la recall di 0.88.
- **k=9**: Con l'aumento della lunghezza dei k-mer, si è notato un miglioramento della recall per il Clade V, che ha raggiunto un valore di 1.00, a fronte di una leggera diminuzione della precisione (0.87). Per il Clade GK, la precisione è rimasta elevata (1.00), ma la recall è diminuita a 0.84.

Questi risultati suggeriscono che l'aumento della lunghezza dei k-mer può influenzare diversamente le metriche di classificazione per ciascun clade, migliorando una metrica a scapito dell'altra.

### 3.4 Considerazioni generali

La scelta della lunghezza dei k-mer rappresenta un compromesso tra precisione e recall. Un valore di k più basso tende a fornire un equilibrio tra le due metriche, mentre un valore più alto può migliorare una metrica specifica a discapito dell'altra.

È importante notare che, secondo studi precedenti, non esiste una lunghezza di k-mer universalmente ottimale per la classificazione di sequenze, e le prestazioni possono variare a seconda del metodo utilizzato e del tipo di dati.

In conclusione, la selezione della lunghezza dei k-mer dovrebbe essere guidata dagli obiettivi specifici dell'analisi e dalle caratteristiche dei dati disponibili. Un valore di k più basso può essere preferibile quando si desidera un equilibrio tra precisione e recall, mentre un valore più alto può essere scelto per ottimizzare una metrica specifica.

# Conclusioni e sviluppi futuri

Il lavoro svolto ha permesso di approfondire il potenziale dell'integrazione tra tecniche di machine learning, rappresentazioni visive delle sequenze genomiche e strumenti di interpretabilità come gli SHAP values, per l'identificazione di marker distintivi tra differenti clade virali. Il cuore dell'approccio ha previsto la trasformazione delle sequenze in immagini, successivamente analizzate da una rete neurale per individuare pattern ricorrenti e regioni informative, con l'obiettivo di differenziare i clade V e GK.

Nel corso della sperimentazione sono stati presi in considerazione due diversi valori di  $k$  per la generazione dei k-mers ( $k = 7$  e  $k = 9$ ), e per ciascuno di essi sono state testate sequenze di lunghezza variabile ( $N = 24, 50, 100, 200$ ). Questa scelta ha permesso di osservare come la granularità e la quantità di informazione fornita al modello influenzino in maniera significativa l'efficacia nella rilevazione di marker.

I risultati ottenuti per il clade GK si sono dimostrati particolarmente interessanti: non solo il modello è stato in grado di individuare marker specifici e ricorrenti già con basse lunghezze di sequenza, ma l'introduzione dell'approccio *Reverse & Complement* ha ulteriormente arricchito il quadro, rivelando marker aggiuntivi che non erano stati rilevati in fase iniziale. Questo suggerisce che la simmetria intrinseca del DNA e la sua duplice lettura possono giocare un ruolo chiave nell'identificazione di segnali nascosti e potenzialmente discriminanti.

Diversamente, per il clade V, l'analisi ha restituito risultati più deboli. Né la variazione dei parametri, né l'applicazione del *Reverse & Complement* hanno portato all'evidenza di marker chiari o ripetibili. Questa assenza potrebbe essere dovuta a una minore variabilità interna alle sequenze di questo clade, oppure potrebbe indicare che la strategia adottata necessiti di ulteriori ottimizzazioni per riuscire a cogliere segnali più sottili.

In generale, si è osservato come il valore di  $k$  influenzi significativamente la natura e la quantità dei marker identificati. Mentre  $k = 7$  ha mostrato una certa stabilità, soprattutto per il clade V,  $k = 9$  ha fornito risultati più ricchi, in particolare nel caso del clade GK, anche se a costo di una maggiore complessità computazionale e interpretativa.

## Sviluppi futuri

Guardando al futuro, il lavoro qui presentato apre a numerose possibilità di approfondimento. Innanzitutto, si potrebbe sperimentare con architetture di rete più sofisticate, ad esempio modelli basati su convoluzioni 3D o su meccanismi di attenzione (come quelli utilizzati nei Transformer), in grado di catturare relazioni più complesse all'interno delle sequenze genomiche.

Un altro ambito di potenziale sviluppo riguarda l'integrazione di ulteriori fonti di informazione, come dati strutturali o funzionali relativi alle regioni codificanti, che potrebbero fornire un contesto biologico più ricco alla classificazione. Inoltre, sarà fondamentale validare i marker individuati su dataset esterni o reali, per verificarne l'effettiva rilevanza biologica e la generalizzabilità.

Infine, il metodo proposto potrebbe essere adattato ad altri casi d'uso, come la classificazione di varianti patogene, l'analisi filogenetica o la predizione di resistenze farmacologiche. In tal senso, questo approccio si propone come un punto di partenza

per una nuova classe di strumenti computazionali interpretabili, capaci di coniugare rigore matematico e significato biologico.

## References

- [1] Barnsley Michael F. *"Fractals Everywhere: New Edition."*. Dover Publications, 2012.
- [2] H.Joel Jeffrey. Chaos game representation of gene structure. *Nucleic Acids Research*, 18(8):2163–2170, 04 1990. URL <https://doi.org/10.1093/nar/18.8.2163>.
- [3] Jorge Avila Cartes, Santosh Anand, Simone Ciccolella, Paola Bonizzoni, and Gianluca Della Vedova. Accurate and fast clade assignment via deep learning and frequency chaos game representation. *GigaScience*, 12, 12 2022. URL <https://doi.org/10.1093/gigascience/giac119>.