UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

Rilevamento delle anomalie per test sequenziali applicati ad autenticazione a livello fisico

Relatore:
Prof. Stefano Tomasin

Laureando:
Francesco Pizzolato

ANNO ACCADEMICO: 2023/2024
Data di laurea: 25/09/2024

Indice

Sommario

L'obiettivo di questa tesi è l'analisi delle problematiche legate alla Physical Layer Authentication (PLA) applicata al canale acustico subacqueo (UWAC), un modello complesso e variabile a causa delle caratteristiche fisiche e morfologiche dell'ambiente sottomarino. In particolare, ci siamo concentrati sulla rilevazione di attacchi di impersonificazione tramite tecniche di quickest detection, introducendo il test CUSUM. Poiché la distribuzione statistica delle osservazioni dell'attaccante non è nota tale metodo non è applicabile, perciò abbiamo sviluppato l'algoritmo CUSUM UWAC, assumendo una distribuzione uniforme per l'attaccante e utilizzando distribuzioni gaussiane per emulare il trasmettitore legittimo. Abbiamo testato l'algoritmo su distribuzioni gaussiane complete, troncate e normalizzate, sia per campioni singoli che multidimensionali e gaussiane saturate, confrontando le differenti strategie utilizzate al ricevitore. Infine, abbiamo sviluppato un modello basato su reti neurali LSTM per risolvere il problema della anomaly detection, ottenendo buone performance nella classificazione delle anomalie rispetto agli algoritmi statistici tradizionali.

Capitolo 1

Introduzione

L'argomento che viene discusso e analizzato all'interno di questa tesi è l'autenticazione di messaggi a livello fisico.
La "Physical Layer Authentication" (PLA) è una tecnica di sicurezza che mira a a verificare l'identità di un trasmettitore in una rete di comunicazione direttamente attraverso le caratteristiche fisiche del segnale. Questa tecnica di autenticazione è differente rispetto alle tradizionali tecniche a livello di rete attraverso password o chiavi crittografiche, ma sfrutta proprietà del segnale ricevuto per autenticare l'entità trasmittente. Un esempio di proprietà di un segnale fisico che possono essere sfruttate per PLA possono essere il fading (variazioni del segnale causate da fenomeni di riflessione o diffrazione) o caratteristiche spettrali come variazioni di potenza, frequenza o fase. Tutte queste proprietà sono molto complesse da riprodurre artificialmente e generalmente rendono un segnale univoco.
La PLA permette di difendersi da attacchi di *spoofing*, ovvero impersonificazione, dove un attaccante cerca di imitare il segnale di un trasmettitore

legittimo.
D'altra parte, però, questa tecnica risulta essere molto sensibile all'ambiente (riflessioni, ostacoli, interferenze) e alle proprietà del canale utilizzato dalle comunicazioni della stessa rete.
Il generico meccanismo coinvolto nella PLA è il seguente:

- Ricezione e misurazione del segnale e delle proprietà fisiche coinvolte nell'analisi
- Confronto con un modello
- Decisione di autenticazione in relazione ai risultati offerti dal modello

In particolare, la nostra ricerca si concentrerà sulla PLA applicata al canale acustico sottomarino (UWAC, Underwater Acoustic Channel).
L'obiettivo che ci siamo posti è quello di confrontare modelli differenti di analisi del canale acustico sottomarino, basandoci su tecniche e modelli inizialmente di carattere statistico, come il test CUSUM. Gli algoritmi implementati verranno testati su scenari differenti, in relazione anche al comportamento del ricevitore legittimo.
Il secondo capitolo si occuperà di introdurre in maniera più approfondita il problema su cui vogliamo effettuare le analisi, fornendo i principi fondamentali su cui queste tecniche si basano, oltre a descrivere tutte le problematiche legate alla loro applicazione al canale sottomarino; nel terzo capitolo verranno descritte le soluzioni implementative realizzate in relazione al comportamento del ricevitore, mentre nell'ultimo verranno confrontati e analizzati i risultati ottenuti dagli algoritmi descritti precedentemente.

Capitolo 2

Descrizione del problema

In questo capitolo ci occuperemo di descrivere proprietà e principi fondamentali relativi alla quickest detection, con riferimenti a scenari reali quali i canali acustici subacquei. Si tratta di un argomento importante nell'ambito delle comunicazioni, l'obiettivo è quello di controllare l'autenticità dei messaggi ricevuti. E' un aspetto fondamentale per mantenere la sicurezza dei dati, delle reti e dei sistemi soprattutto in contesti in cui la protezione di tali si ritiene cruciale, come quello sottomarino, per motivi civili e militari.
Mentre molti approcci di autenticazione utilizzano tecniche di crittografia (che rimane uno strumento fondamentale di sicurezza collocato a livello applicativo della manipolazione di informazioni), in questa tesi ci concentreremo prevalentemente sulla Physical Layer Security (PLS), investigando il problema della PLA, che invece fonda i suoi principi sull'analisi fisica e statistica dei segnali che vengono captati da un ricevitore. L'obiettivo di quickest detection è quello di rilevare il cambiamento dello stato di questi ultimi o di un evento che li coinvolge nel minor tempo possibile, minimizzando gli

errori di falso allarme, mancato rilevamento e ritardo di segnalazione. Possiamo descriverlo come problema di ottimizzazione formulato da Pollack [1]. Definiamo una variabile v_n che definisce una funzione legata all' n-esima osservazione rappresentata da una finestra di ricezioni, un paramentro λ (una soglia definita dall'utente) e il *detected change time* $t^* \triangleq \min\{t_n \mid v_n > \lambda\}$. Il nostro obiettivo e' il seguente:

$$\min_{\hat{t}} \quad \sup_{t^* \geq 1} \mathbb{E}_v[\hat{t} - t^* \mid \hat{t} \geq t^*] \ , \ subj.to \, \mathbb{E}_\infty[t^*] \geq \beta \tag{2.1}$$

E' ragionevole pensare, inoltre, che l'attacco si protragga per un periodo relativamente lungo, nel tentativo di compromettere irreversibilmente il sistema. Consideriamo perciò uno scenario in cui il difensore inizialmente stia ricevendo segnali da un trasmettitore noto, e che da un determinato istante t un attaccante invii dei segnali per simulare un trasmettitore legittimo in maniera continuativa. Ciò verra rappresentato da un vettore di N elementi $X = \{x[0], ..., x[N-1]\}$ che descrive un insieme di osservazioni. A partire da questa finestra di osservazioni il nostro obiettivo sarà determinare nel minor tempo possibile che l'attacco è iniziato. Nel contesto di studio di sistemi in evoluzione, infatti, risulta sempre più efficiente mantenere un approccio sequenziale piuttosto che *single-shot*, ovvero considerare una sequenza di input rispetto ad un singolo campione. Ciò può essere particolarmente utile quando i dati precedenti forniscono insight preziosi per la previsione o l'ottimizzazione delle prestazioni future. Notiamo anche che si tratta di un'analisi dinamica di un sistema in evoluzione, perciò un approccio sequenziale che considera una finestra di osservazioni può risultare molto più efficace.

Purtroppo, ci sono delle criticità non trascurabili quando affrontiamo il tema di *quickest detection* associato all' UWAC. Definire un approccio statistico efficiente risulta estremamente complicato se calato nello scenario in cui ci siamo posti. Un modello affidabile della comunicazione sottomarina è molto complesso da definire per la variabilità degli scenari presenti nelle varie aree geografiche e soprattutto per le proprietà fisiche del mezzo di propagazione dei segnali acustici, l'acqua. Il comportamento di questo mezzo è strettamente dipendente da parametri quali temperatura, salinità, profondità e pressione che influenzano ad esempio la velocità di propagazione di tali segnali [1].
Inoltre, bisogna considerare che affrontando un attacco al nostro ricevitore, il nemico cercherà di mantenere sempre la propria imprevedibilità e segretezza, evitando di condividere in qualsiasi modo le proprie tecniche e informazioni riguardanti la statistica della sua trasmissione. Gli unici dati certamente validi a nostra disposizione saranno quindi quelli riguardanti la comunicazione tra trasmettitori e ricevitori legittimi, dai quali cercheremo di ipotizzare un modello risolutivo per determinare la natura della trasmissione.

2.1 CUSUM Test

Consideriamo un ricevitore che, a partire da uno stato legittimo in cui i segnali ricevuti sono esclusivamente autentici, identificato dall'ipotesi H_0, subisce un attacco prolungato da parte di un trasmettitore nemico che tenta di impersonare il trasmettitore legittimo, passando al secondo stato H_1. Per quanto enunciato precedentemente, il nostro target sarà quello di identificare

nel minor tempo possibile questo cambiamento di condizione del ricevitore. Si tratta quindi di un problema di *change detection*.
Una delle tecniche basate su approccio statistico più efficaci a questo fine è il CUSUM Test (Cumulative Sum Test). Si tratta di un test sequenziale che permette di individuare cambiamenti di stato non immediatamente evidenti in un insieme di dati. Il principio di base del test CUSUM è calcolare la somma cumulativa di una generica funzione dell'input corrente e un valore di riferimento. Se i dati deviano dal valore di riferimento, la somma cumulativa si allontana da esso. Questa deviazione crescente può indicare un cambiamento nel processo. A partire da queste valutazioni viene definita una soglia che rappresenta un valore significativamente diverso da quello di riferimento per cui si conclude che è avvenuto un cambiamento di stato nel sistema.

Definiamo il problema di autenticazione come un problema di *binary hypothesis testing*, in cui le due ipotesi H_0 e H_1 costituiscono i casi legittimo e sotto attacco. Formalizziamo il test introducendo una variabile v_n, funzione dell'input corrente e quelli passati, un paramentro λ (una soglia definita dall'utente per una determinata probabilità di falso allarme) e *detected change time*

$$t^* \triangleq \min\{t_n \mid v_n > \lambda\}. \tag{2.2}$$

Il nostro obiettivo è quello di minimizzare il tempo tra l'inizio effettivo dell'attacco e l'istante in cui esso viene riconosciuto. Più precisamente, note le densità di probabilità del segnale ricevuto x_n in relazione al canale UWAC rispetto alle due ipotesi $p(x_n|H_0)$, $p(x_n|H_1)$, l'algoritmo CUSUM all'istante

t_n opera nel modo seguente:

- Calcola il Likelihood Ratio (LR)

$$l_n = \log \frac{p(x_n|H_1)}{p(x_n|H_0)} \tag{2.3}$$

- Aggiorna la variabile $v_n = max\{v_{n-1} + l_n, 0\}$, con $v_0 = 0$,
- Verifica $v_n > \lambda$: se vero, viene lanciato un allarme e consideriamo $t^* = t_n$ come il valore dell'istante di detection

2.1.1 CUSUM UnderWater Acoustic Channel (UWAC)

Abbiamo analizzato lo stato dell'arte per quanto riguarda la tematica dell'autenticazione a livello fisico e in generale quella del *binary hypothesis testing*. Sottolineiamo nuovamente che l'UWAC presenta non poche problematiche nella sua caratterizzazione. È complesso quindi ottenere un modello affidabile del canale subacqueo. Inoltre, nello scenario proposto dall'algoritmo CUSUM (dimostrato ottimo quando i campioni rilevati sono indipendenti e identicamente distribuiti), la statistica dei casi legittimo e sotto attacco devono essere note a priori per calcolare l'LR (2.1.1). Ciò a livello pratico è impossibile considerando che ci stiamo difendendo da un attaccante nemico, che non condividerà mai la propria strategia d'attacco. Nella pratica perciò il test CUSUM non può essere applicato nella sua totalità. Nella nostra analisi sarà comunque fondamentale definirne una sua variante come riferimento per

avvicinarsi il più possibile alle prestazioni di CUSUM.

Non conoscendo alcun dettaglio riguardo la trasmissione con cui l'attaccante agisce, ipotizziamo che la distribuzione illegittima sia di tipo uniforme, con supporto nell'intervallo $[\mu_0 - k\sigma_0, \mu_0 + k\sigma_0]$, dove k è un valore intero positivo scelto arbitrariamente, ovvero ipotizziamo:

$$\hat{p}(x_n|H_1) = \begin{cases} \frac{1}{2k\sigma_0} & \text{se } \mu_0 - k\sigma_0 \leq x \leq \mu_0 + k\sigma_0 \\ 0 & \text{altrimenti} \end{cases}$$

Mantenendo le stesse proprietà definite dell'algoritmo classico, l'algoritmo CUSUM UWAC all'istante t_n effettua le seguenti operazioni:

- Calcola l'LR

$$l_n = \log \frac{\hat{p}(x_n|H_1)}{p(x_n|H_0)} \tag{2.4}$$

- Aggiorna la variabile $v_n = max\{v_{n-1} + l_n, 0\}$, con $v_0 = 0$,
- Verifica $v_n > \lambda$: se vero, viene lanciato un allarme e consideriamo $t^* = t_n$ come il valore dell'istante di detection.

Intuitivamente, l'utilizzo dell'indice l_n permette di accumulare valori positivi quando la distribuzione associata ad H_1 è dominante rispetto ad H_0, mentre nella situazione opposta definisce dei valori negativi (a causa della presenza della funzione logaritmo, negativa per valori appartenenti all'intervallo $[0, 1]$), che sommati alla variabile cumulativa v_n evitano che la soglia λ

venga oltrepassata. La variabile v_n viene azzerata se assume valori negativi per evitare che la somma cumulativa si allontani negativamente dalla soglia, evitando che il test non riconosca velocemente un attacco.

Capitolo 3

Risoluzione del problema

Nel capitolo precedente abbiamo introdotto la problematica legata all'autenticazione a livello fisico applicata al canale acustico subacqueo e tutte le criticità legate ad esso. In particolare, abbiamo evidenziato che lo sviluppo di algoritmi basati sul test CUSUM utilizzando opportune ipotesi possono risultare comunque interessanti nonostante non si conosca la distribuzione associata alla trasmissione illegittima di un potenziale attaccante.
In questo capitolo verranno descritti gli algoritmi sviluppati basati proprio sul tale test per poi andare ad introdurre un nuovo paradigma risolutivo che sfrutterà le reti neurali. Nel momento in cui le tecniche basate su modelli statistici potrebbero essere di difficile applicazione nel sistema analizzato, l'ipotetica soluzione di interesse potrebbe essere quella dell'utilizzo delle reti neurali.
Nel nostro caso, come approfondiremo nel capitolo successivo, il processo di detection dei segnali illegittimi viene effettuato assumendo che questi siano distribuiti secondo una distribuzione uniforme: ipotizziamo che i segnali

non legittimi non abbiano una struttura statistica ben definita o prevedibile, ma si distribuiscano uniformemente su un certo intervallo di possibili valori. Questa ipotesi ci permette di identificare i segnali illegittimi senza la necessità di conoscere a priori la loro statistica precisa. Il modello viene quindi addestrato a riconoscere i segnali legittimi, mentre quelli che si discostano in modo significativo da questo comportamento vengono classificati come potenzialmente illegittimi.
Tutte le soluzioni proposte a seguire sono state realizzate in linguaggio Python.

3.1 Algoritmi CUSUM UWAC

3.1.1 Algoritmo CUSUM UWAC con distribuzioni non troncate

Per applicare il test CUSUM è necessaria la conoscenza di entrambe le distribuzioni delle due segnalazioni, ma ciò non è possibile a causa dello scenario che abbiamo ipotizzato nello sviluppo di tali soluzioni: per il calcolo dell'LR all'interno del test CUSUM abbiamo quindi assunto che la distribuzione della segnalazione illegittima sia di tipo uniforme con supporto nell'intervallo $[\mu_0 - k\sigma_0, \mu_0 + k\sigma_0]$, dove μ_0 rappresenta la media e σ_0 la deviazione standard associate alla distribuzione legittima, mentre k è un numero intero strettamente positivo. La segnalazione legittima invece è una gaussiana completa. Per valutare l'LR è necessario che i sample appartengano al supporto della distribuzione uniforme, perciò in questa situazione la scelta implementativa

è stata quella di segnalare un'anomalia per ogni sample ricevuto esterno al supporto a prescindere dal fatto che sia legittimo o illegittimo: per valori elevati di k, infatti, la probabilità che un sample legittimo sia esterno a tale supporto è estremamente bassa, perciò le occorrenze in cui si otterrebbero dei falsi allarmi sono limitate.

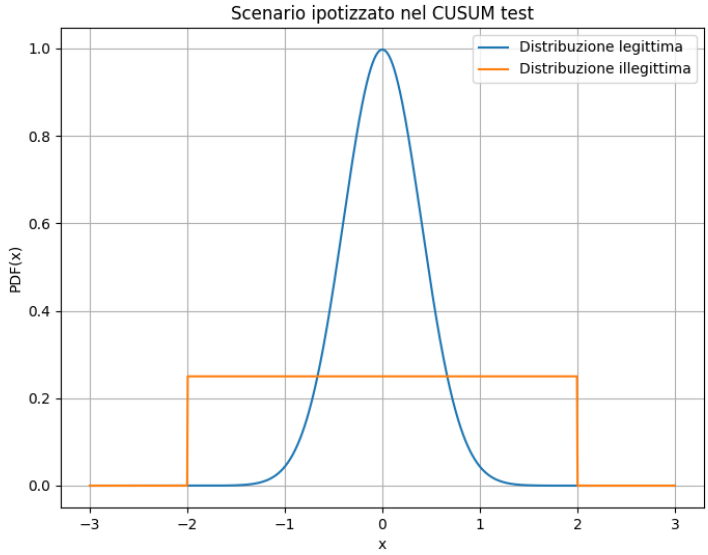


Figura 3.1: Il valore dell'LR associato ad un determinato sample viene calcolato all'interno dell'algoritmo CUSUM in relazione alle distribuzioni mostrate in figura, ipotizzando che la segnalazione dell'attaccante segua una distribuzione di tipo uniforme.

3.1.2 Algoritmo CUSUM UWAC con distribuzioni troncate e normalizzate

Come introdotto nel capitolo precedente, in questa variante alternativa del test CUSUM non sono più note entrambe le densità di probabilità legate ai

due stati come nelle ipotesi teoriche precedenti, ma solamente quella relativa allo stato H_0, ovvero quello in cui la trasmissione è legittima. In particolare in questo caso si tratta di una distribuzione gaussiana troncata di media μ_0 e deviazione standard σ_0, con supporto nell'intervallo $[\mu_0 - k\sigma_0, \mu_0 + k\sigma_0]$ con $k > 0$ e normalizzata attraverso un fattore di scala $c > 0$.

$$p(x_n|H_0) = \frac{c}{\sigma_0\sqrt{2\pi}} \cdot \exp\left(-\frac{(x-\mu_0)^2}{2\sigma_0^2}\right) \cdot \text{rect}\left(\frac{x-\mu_0}{2k\sigma_0}\right) \tag{3.1}$$

L'idea descritta precedentemente è quella di progettare la distribuzione della segnalazione legittima in funzione del supporto associato alla distribuzione uniforme che emula la segnalazione illegittima. Come abbiamo visto precedentemente, il supporto è definito dai valori associati alla variabile k e la deviazione standard σ_0, perciò è sufficiente manipolare tale variabili per definire la dimensione del supporto. La distribuzione associata alla segnalazione legittima risulta quindi essere una gaussiana ma in questo caso troncata rispetto all'intervallo stabilito. Dal punto di vista computazionale le differenze rispetto all'algoritmo precedente sono legate alla definizione delle sequenze di testing: per emulare la segnalazione legittima vengono considerati i sample in uno specifico intervallo, scartando quelli esterni (al contrario del caso precedente che generava tutti i sample appartenenti alla distribuzione). Inoltre perchè tale funzione risulti essere una funzione densità di probabilità è necessario normalizzarla rispetto alla superficie sottesa da essa: ciò viene effettuato attraverso il calcolo di un fattore di scala $c > 1$, ottenuto valutando l'integrale della gaussiana nel supporto definito precedentemente che andrà a moltiplicare il valore della PDF valutata rispetto ad un determinato sample.

Ipotizzando inoltre che in ingresso al ricevitore i sample esterni al supporto della distribuzioni uniforme utilizzata nel CUSUM UWAC vengano scartati, anche la distribuzione illegittima viene troncata e normalizzata, così che l'algoritmo CUSUM UWAC riceva in ingresso solamente sample appartenenti al supporto dell'uniforme: ciò permette sempre di valutare tutti i sample calcolando l'indice LR.

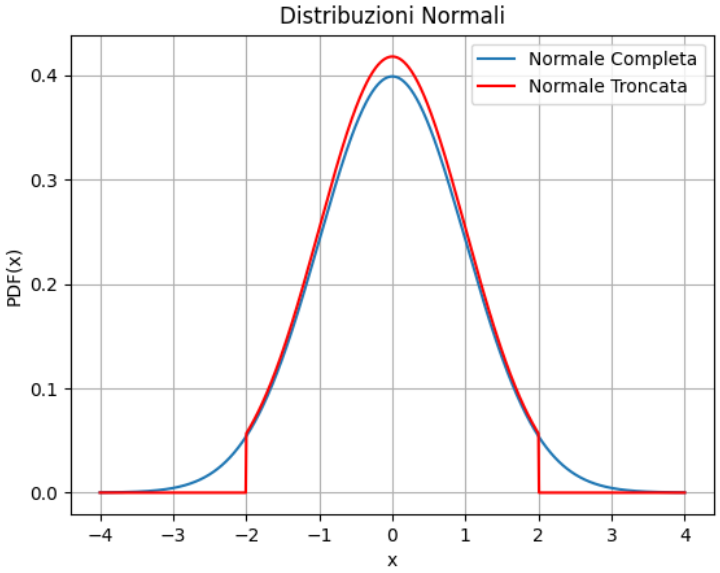


Figura 3.2: Funzioni densità di probabilità di due distribuzioni gaussiane troncata e non con stessa media e deviazione standard; possiamo notare come la distribuzione normale troncata sia scalata rispetto alla distribuzione non troncata.

A partire dalle proprietà della distribuzione gaussiana completa, per il calcolo dell'LR si utilizza una routine specifica che permette di calcolare il valore della PDF legittima troncata attraverso il prodotto con questo fattore di scala c. Non è necessario effettuare questa valutazione per la segnalazione nemica in quanto il test CUSUM UWAC utilizza una distribuzione uniforme

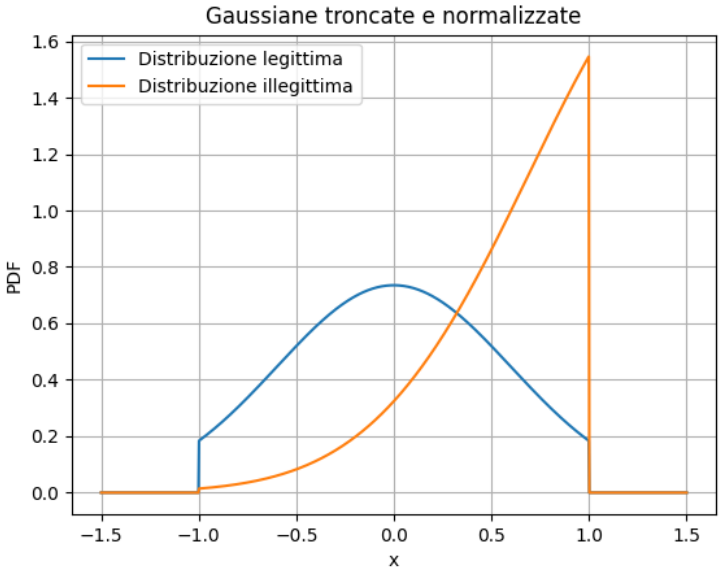


Figura 3.3: In questo caso il dataset genera i sample legittimi e illegittimi a partire da una distribuzione gaussiana troncata e normalizzata.

pre valutare l'LR.

$$\int_{-\infty}^{+\infty} \frac{1}{\sigma_0\sqrt{2\pi}} e^{-\frac{(x-\mu_0)^2}{2\sigma_0^2}} \, dx = 1 \tag{3.2}$$

$$\int_{\mu_0-k\sigma_0}^{\mu_0+k\sigma_0} \frac{1}{\sigma_0\sqrt{2\pi}} e^{-\frac{(x-\mu_0)^2}{2\sigma_0^2}} \, dx \neq 1 \tag{3.3}$$

$$c\int_{\mu_0-k\sigma_0}^{\mu_0+k\sigma_0} \frac{1}{\sigma_0\sqrt{2\pi}} e^{-\frac{(x-\mu_0)^2}{2\sigma_0^2}} \, dx = 1 \tag{3.4}$$

$$c = \frac{1}{\int_{\mu_0-k\sigma_0}^{\mu_0+k\sigma_0} \frac{1}{\sigma_0\sqrt{2\pi}} e^{-\frac{(x-\mu_0)^2}{2\sigma_0^2}} \, dx} \tag{3.5}$$

Il vantaggio di utilizzare questo approccio al ricevitore è il fatto che definendo

la segnalazione legittima in funzione del supporto legato alla statistica illegittima si evita la possibilità di valutare a priori come illegittimi dei sample esterni al supporto, eventualità che risulterebbe essere piuttosto probabile se si utilizzassero delle distribuzioni gaussiane complete soprattutto nel momento in cui l'intervallo della distribuzione illegittima ha un supporto piccolo, cioè per valori piccoli della variabile k. Per valori di k molto bassi, infatti, la probabilità di scartare sample legittimi esterni al supporto sarebbe molto elevata e ciò causerebbe un degrado notevole nelle prestazioni della segnalazione. Se tali campioni anzichè essere scartati venissero etichettati immediatamente come illegittimi si otterrebbe una PFA molto elevata.

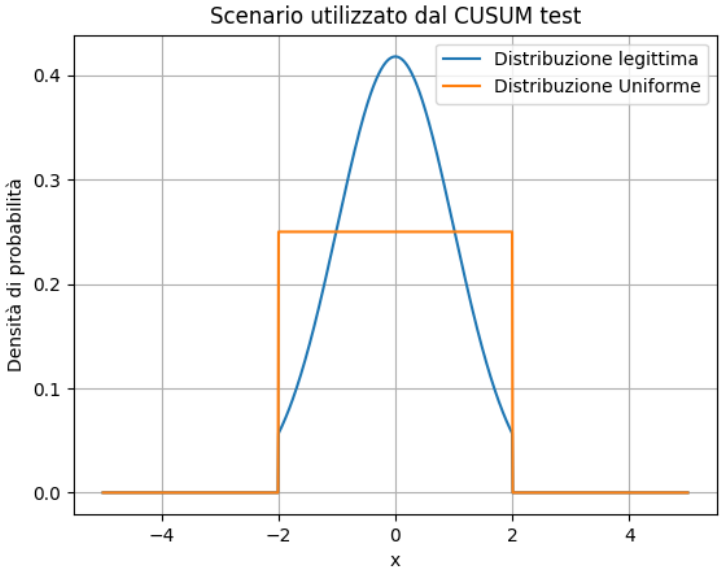


Figura 3.4: Il valore dell'LR associato ad un determinato sample viene calcolato all'interno dell'algoritmo CUSUM in relazione alle distribuzioni mostrate in figura, ipotizzando che la segnalazione dell'attaccante segua una distribuzione di tipo uniforme. In questa situazione, però, nessun sample legittimo è esterno all'intervallo di validità della distribuzione uniforme.

3.1.3 Algoritmo CUSUM UWAC con sample multidimensionale

Nelle soluzioni proposte precedentemente abbiamo considerato sample unidimensionali nel calcolo dell'LR nel CUSUM test, ovvero sample costituiti da una singola ricezione. E' possibile applicare il test CUSUM anche su osservazioni multidimensionali considerando sample che realizzano un vettore di osservazioni, andando quindi a modellare scenari in cui il ricevitore estrae più caratteristiche da ciascun segnale ricevuto (es. potenza e doppler). Intuitivamente considerando un vettore di elementi per il calcolo dell'LR sarebbe possibile ottenere performance migliori in termini di falso allarme, missed detection e ritardo di segnalazione, poichè si prende in considerazione una finestra più ampia di osservazioni.
Il calcolo dell'LR all'interno dell'algoritmo CUSUM è stato modificato in funzione delle nuove proprietà del dataset.

$$l_n = \log \frac{p(\boldsymbol{x_n}|\tilde{H}_1)}{p(\boldsymbol{x_n}|H_0)} \tag{3.6}$$

Supponendo che i campioni ricevuti siano i.i.d e che la dimensione del vettore sia descritta dalla variabile t, possiamo dedurre che

$$\begin{aligned} l_n &= \log \frac{\prod_{j=0}^{t-1} p(x_{n,j}|\tilde{H}_1)}{\prod_{j=0}^{t-1} p(x_{n,j}|H_0)} \\ &= \sum_{j=0}^{t-1} \log \frac{p(x_{n,j}|\tilde{H}_1)}{p(x_{n,j}|H_0)} \\ &= \sum_{j=0}^{t-1} l_{n,j} \end{aligned} \tag{3.7}$$

Ciò significa che l'LR associato al sample multidimensionale n-esimo è pari alla somma degli LR unidimensionali calcolati sulle singole osservazioni del vettore.

3.1.4 Algoritmo CUSUM UWAC con distribuzioni troncate e saturate

Analogamente a quanto introdotto nella sezione 3.1.2, un'ulteriore soluzione potrebbe essere quella di saturare le distribuzioni associate alle segnalazioni in ingresso al ricevitore. Saturare una generica funzione su un intervallo $[a, b]$ implica che per tutti i punti esterni a tale intervallo una funzione assuma i valori corrispondenti agli estremi, ovvero:

$$f(x) = \begin{cases} f(a), & \text{se } x < a, \\ f(x), & \text{se } a \leq x \leq b, \\ f(b), & \text{se } x > b. \end{cases} \tag{3.8}$$

Utilizzare un approccio simile per le distribuzioni legittima e illegittima permette, allo stesso modo della tecnica di normalizzazione utilizzata precedentemente, di poter valutare l'LR di ogni singolo campione ricevuto in relazione al supporto della distribuzione uniforme utilizzata nell'algoritmo CUSUM UWAC senza dover nè scartare sample a priori nè considerarli direttamente illegittimi.

3.2 Algoritmo implementato attraverso rete neurale

Un approccio differente a quelli presentati nelle sezioni precedenti è quello dell'utilizzo di una rete neurale per effettuare l'autenticazione.
L'idea fondamentale è quella di sostituire il calcolo dell'LR statistico con un modello che sia in grado di valutare i campioni che vengono ricevuti in input dal sistema. Questa valutazione verrà poi utilizzata per segnalare una potenziale anomalia.

3.2.1 Struttura del modello

L'analisi che dobbiamo svolgere può essere definita come una sorta di classificazione binaria associata ai sample ricevuti in input: il modello dovrà valutare la legittimità o meno di ogni campione ricevuto. Per far ciò è stato implementato un modello di rete neurale LSTM (Long Short Term Memory) sequenziale composta da due strati, progettata per risolvere un problema di classificazione binaria. Le reti neurali LSTM sono efficaci per l'apprendimento di sequenze temporali. In fase di training, i sample legittimi sono stati etichettati con il valore 0, mentre quelli potenzialmente illegittimi con il valore 1.

L'algoritmo implementato analizza l'output ottenuto dal modello per ogni sample appartenente alla sequenza: nel momento in cui viene rilevato il primo sample potenzialmente illegittimo della sequenza l'algoritmo segnala un'anomalia restituendo l'istante di detection, ovvero la posizione del campione

all'interno della sequenza.

Capitolo 4

Risultati

Questa sezione è dedicata all'analisi dei risultati ottenuti dalle soluzioni sviluppate nel capitolo precedente, sia utilizzando l'approccio statistico basato sull'algoritmo CUSUM che l'utilizzo di modelli basati su reti neurali.

4.1 Generazione del dataset

Per verificare le performance e la correttezza degli algoritmi implementati è stato fondamentale prima di tutto definire la struttura delle sequenze generate per poi definire gli indicatori di performance per la successiva analisi. Per simulare il cambiamento di stato da comunicazione legittima a illegittima sono state generate 5000 sequenze di lunghezza 40 elementi. Per rappresentare le due segnalazioni i campioni della sequenza vengono generati secondo due distribuzioni gaussiane ognuna delle quali descritta da media μ_0, μ_1 (sempre differente tra le due distribuzioni) e deviazione standard σ_0, σ_1. L'istante del cambiamento di stato da legittimo a illegittimo viene generato in ma-

niera casuale secondo una distribuzione uniforme a partire dal decimo al trentesimo sample della sequenza: i primi dieci, per costruzione, appartengono sempre alla comunicazione legittima mentre gli ultimi 10 appartengono sempre alla distribuzione illegittima così da avere un numero sufficiente di sample illegittimi per poter testare l'efficacia dei nostri algoritmi. Oltre alla sequenza generata, per ognuna di esse è necessario memorizzare una variabile `change_time` che rappresenta l'indice di cambiamento di stato con cui analizzeremo il ritardo di rilevamento dell'algoritmo.

Per quanto riguarda gli indicatori implementati per l'analisi delle performance, sono state utilizzate le seguenti metriche: la probabilità di falso allarme (PFA) ovvero la frazione di sequenze in relazione al totale in cui viene rilevata un'anomalia dall'algoritmo prima che essa sia effettivamente realizzata all'interno delle sequenze del dataset ed il ritardo di rilevamento (DELAY), ovvero il numero di sample di ritardo con cui viene rilevata un'anomalia rispetto all'inizio dell'attacco.

Tutte queste proprietà sono state valutate in funzione del valore che viene associato alla soglia λ che influenza le prestazioni di questi algoritmi. In generale una soglia bassa predilige una PMD e un DELAY di segnalazione molto bassi a discapito della PFA che invece risulta essere elevata; al contrario con valori elevati della soglia si ottiene il risultato opposto, cioè valori alti della PMD e DELAY e bassi di PFA. Le variabili manipolate nei vari test sono i parametri delle distribuzioni gaussiane, la larghezza del supporto di validità della distribuzione uniforme utilizzata dall'algoritmo CUSUM (che dipende dalla costante moltiplicativa k) e la soglia λ.

4.2 Scenari

Per verificare l'efficienza delle soluzioni proposte all'interno della tesi sono stati utilizzati due scenari differenti che rappresentano le due distribuzioni della segnalazione legittima e non. Il primo scenario coinvolge due distribuzioni gaussiane (opportunamente troncate o saturate in relazione all'algoritmo sviluppato): la distribuzione legittima ha media $\mu_0 = 0$ e deviazione standard $\sigma_0 = 0.4$, mentre la distribuzione illegittima è centrata in $\mu_1 = 1$, sempre con deviazione standard $\sigma_1 = 0.4$. Questo scenario risulta essere il più complesso da valutare, infatti la sovrapposizione delle due segnalazioni è molto marcata. Il secondo scenario invece risulta essere più semplice, ma rimane comunque un caso significativo per le performance che verranno valutate; la prima distribuzione è invariata, mentre in questo caso quella illegittima ha media $\mu_1 = 2$ (è più lontana rispetto al caso precedente) e stessa deviazione standard dello scenario precedente per cui la sovrapposizione tra le due è minore. Nei vari algoritmi implementati tali distribuzioni vengono opportunamente troncate, normalizzate o saturate in relazione alla strategia implementata dal ricevitore.

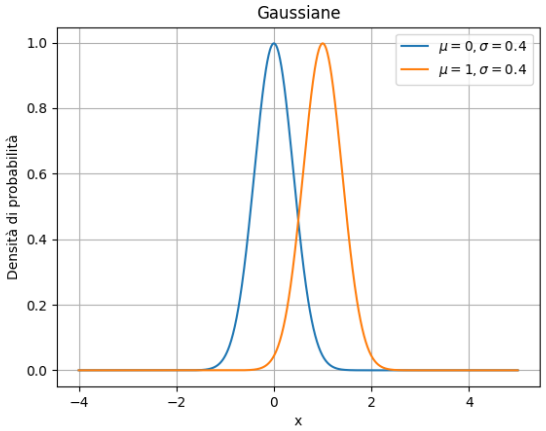


Figura 4.1: Primo scenario: la distribuzione blu è associata alla segnalazione legittima. Questa situazione risulta essere molto critica, infatti le due distribuzioni sono estremamente ravvicinate tra loro.

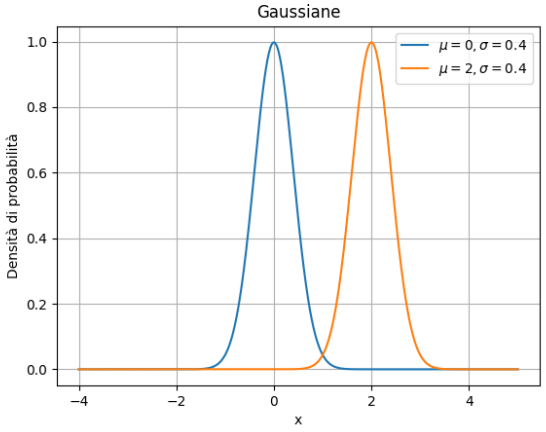


Figura 4.2: Secondo scenario: la distribuzione blu è associata alla segnalazione legittima. Questa situazione risulta essere più favorevole, infatti le due distribuzioni sono più distanziate fra loro.

4.3 Algoritmo CUSUM UWAC con distribuzioni non troncate

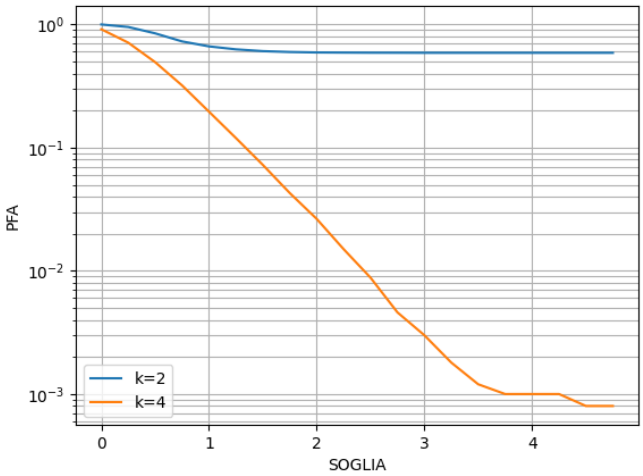


Figura 4.3: PFA in funzione della soglia per $k = 2$ e $k = 4$, primo scenario. Possiamo notare come la porzione di sequenze per $k = 2$ in cui viene rilevato un falso allarme sia molto alta a causa sia della vicinanza delle due distribuzioni sia della lunghezza ridotta del supporto di definizione della distribuzione uniforme utilizzata nel CUSUM. Per $k = 4$ (il supporto raddoppia), le prestazioni sono estremamente migliorate.

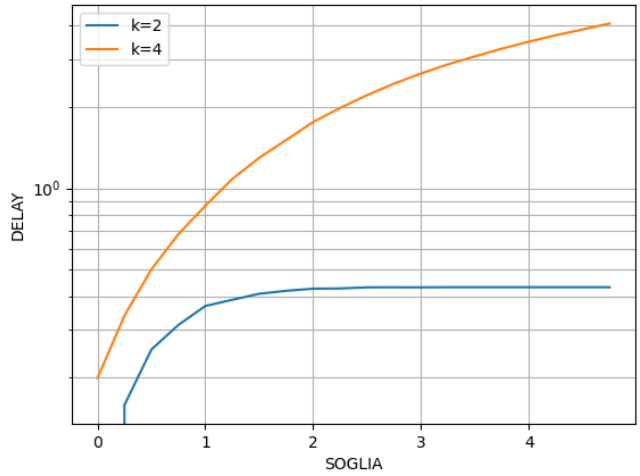


Figura 4.4: DELAY in funzione della soglia per $k = 2$ e $k = 4$, primo scenario. Possiamo notare come per $k = 2$ il DELAY medio sia inferiore rispetto a $k = 4$. Il motivo è dato dal valore che assume la funzione densità di probabilità uniforme nei due casi: per $k = 2$ è maggiore rispetto a $k = 4$, perciò l'LR assume valori più elevati e supera la soglia più velocemente.

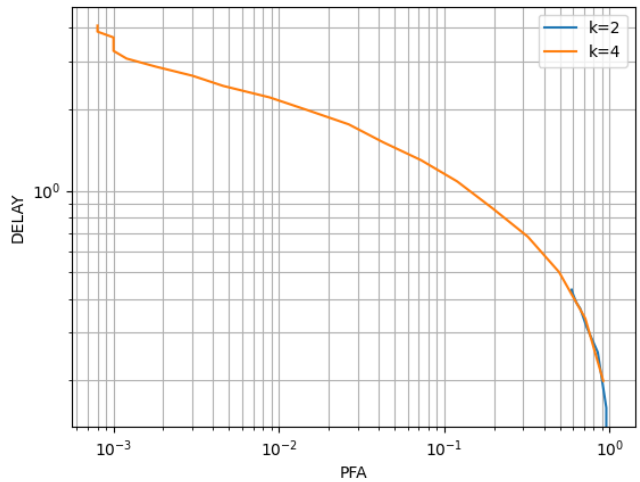


Figura 4.5: Curva PFA-DELAY Tradeoff per $k = 2$,$k = 4$, primo scenario. Per gli stessi valori di PFA le curve sono sovrapposte, ma come abbiamo visto nei grafici precedenti per $k = 4$ vengono offerte prestazioni di PFA più efficienti.

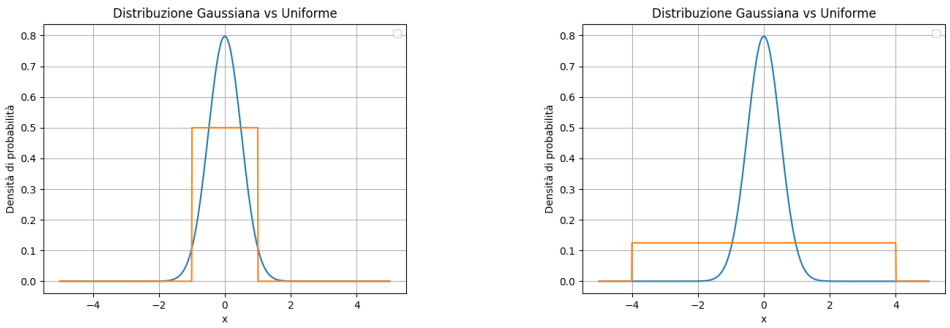
4.3.1 Osservazioni

Il primo algoritmo che abbiamo considerato utilizza un dataset che genera delle distribuzioni gaussiane complete sia per la segnalazione legittima che per quella illegittima, ma presenta delle limitazioni piuttosto importanti soprattutto per valori del supporto della distribuzione uniforme bassi (cioè per valori di k non elevati). In questo contesto, per come è stato realizzato l'algoritmo, nel momento in cui si riscontra un sample non appartenente al dominio della distribuzione uniforme definito esso viene immediatamente riconosciuto come sample illegittimo, in quanto esterno all'intervallo predisposto. Ciò causa un innalzamento piuttosto significativo della probabilità di falso allarme (PFA), in quanto è possibile che vengano scartati campioni legittimi. Ciò si nota in maniera estremamente marcata per intervalli molto piccoli del supporto, poichè la probabilità che un sample legittimo sia al di fuori di tale intervallo risulta molto elevata (è sufficiente calcolare la differenza tra 1 e l'integrale della distribuzione legittima su tale supporto per valutare la probabilità che un sample amico venga scartato).

$$P(x < \mu_0 - k\sigma_0 \vee x > \mu_0 + k\sigma_0) = 1 - \int_{\mu_0 - k\sigma_0}^{\mu_0 + k\sigma_0} \frac{1}{\sigma_0\sqrt{2\pi}} e^{-\frac{(x-\mu_0)^2}{2\sigma_0^2}} dx. \quad (4.1)$$

Per valori di k più elevati, al contrario, la probabilità di scartare campioni legittimi è inferiore perciò l'algoritmo è più efficace per lo meno in termini di probabilità di falso allarme. Un altro aspetto da prendere in considerazione per valutare le performance dell'algoritmo CUSUM è che la larghezza del supporto della distribuzione uniforme (che nel calcolo dell'LR rappresenta la

segnalazione ipotizzata illegittima) influenza il valore della PDF stessa. Per valori bassi di k, il valore della PDF valutata nel supporto sarà maggiore rispetto a valori di k più elevati. Ciò fa sì che il valore della funzione densità di probabilità uniforme sia maggiore rispetto alla distribuzione legittima per gran parte del supporto definito, anche per intervalli in cui la segnalazione legittima genera dei sample piuttosto vicini alla media della distribuzione. Ciò determina l'aumento della somma cumulativa nell'algoritmo CUSUM anche per sample la cui probabilità di essere legittimi è piuttosto elevata, sollevando un'anomalia. Ciò spiega l'elevata PFA per $k = 2$ rispetto al test valutato con $k = 4$.



(a) k=2 (b) k=8

Figura 4.6: Differenza fra funzioni densità di probabilità uniforme e gaussiana in relazione alla variabile k. Nel primo caso vengono scartati molti sample esterni e il valore dell'uniforme è più elevato rispetto alla seconda immagine.

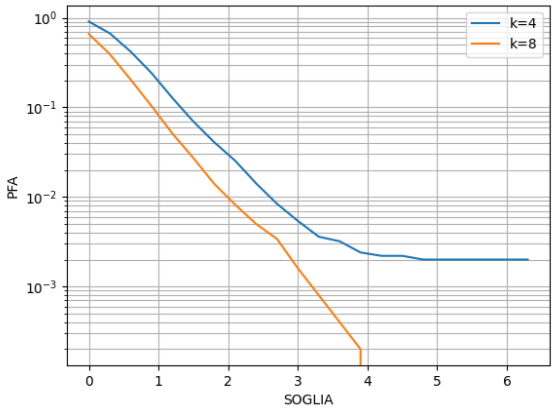


Figura 4.7: PFA in funzione della soglia per $k = 4$, $k = 8$, secondo scenario. Possiamo notare come per $k = 8$ la PFA sia minore a parità di soglia fino ad annullarsi, mentre per $k = 4$ si stabilizza.

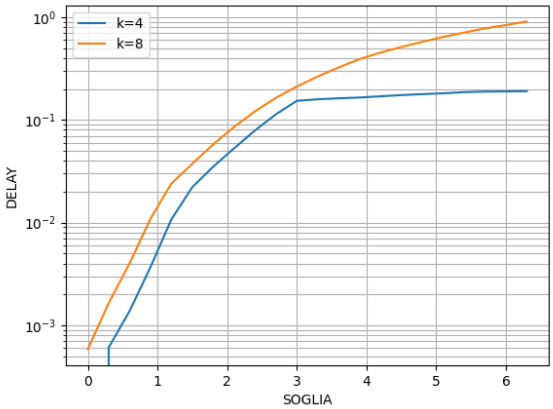


Figura 4.8: DELAY in funzione della soglia per $k = 4$, $k = 8$, secondo scenario. Come motivato nella sezione 4.3.1, il delay per $k = 4$ è minore rispetto a $k = 8$.

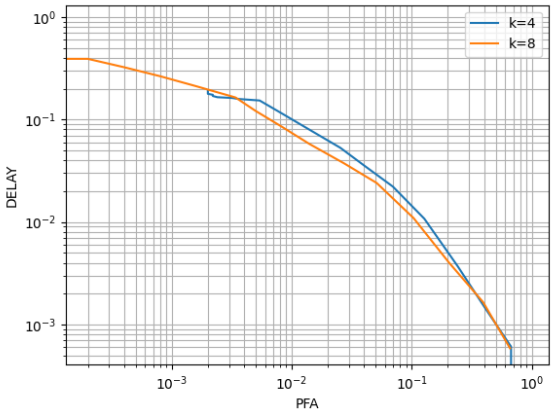


Figura 4.9: Curva di PFA-DELAY Tradeoff per $k = 4$, $k = 8$, secondo scenario.

4.3.2 Osservazioni

Grazie all'allontanamento della distribuzione associata alla segnalazione illegittima e ad un elevato valore di k questo scenario ha offerto delle prestazioni significativamente migliori rispetto al caso precedente. Questo è dovuto all'allontanamento della distribuzione illegittima rispetto al valore medio di quella legittima. Nonostante ciò questi dati rimangono comunque significativi in quanto questo scenario non è assolutamente banale infatti le distribuzioni presentano valori medi piuttosto simili ed una deviazione standard elevata, che a livello pratico causa un *allargamento* delle funzioni densità di probabilità. E' Il primo scenario in realtà a risultare un modello estremamente sfavorevole e complesso da analizzare da questo algoritmo, in quanto la sovrapposizione delle due distribuzioni è molto marcata.
Possiamo notare inoltre come in entrambi gli scenari le curve di PFA-DELAY Tradeoff in relazione a k siano pressochè sovrapposte. Ciò significa che non c'è un tradeoff effettivamente migliore per valori differenti di k, ma per $k = 8$ vengono offerte performance di PFA superiori.

4.4 Algoritmo CUSUM UWAC con distribuzioni troncate e normalizzate

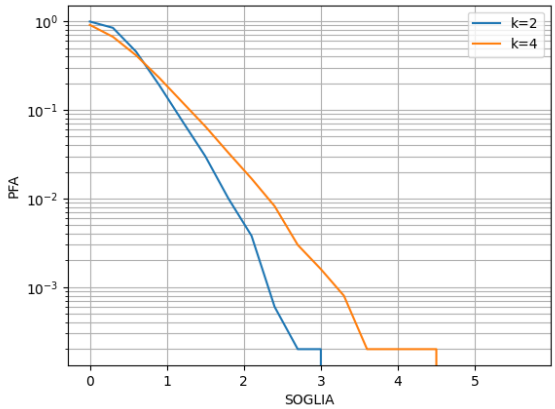


Figura 4.10: PFA in funzione della soglia per $k = 2$, $k = 4$, primo scenario. La PFA risulta essere minore a parità di soglia a quella valutata nella sezione precedente grazie all'introduzione della distribuzione legittima troncata rispetto al dominio dell'uniforme.

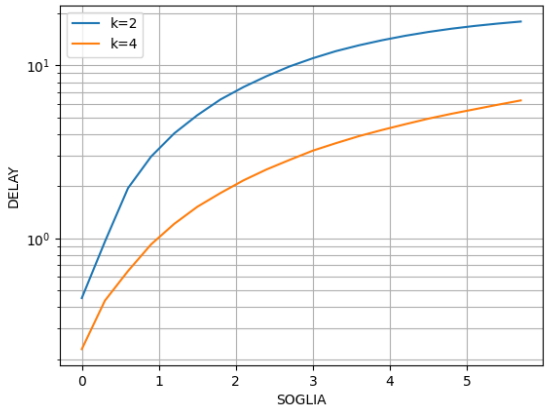


Figura 4.11: DELAY in funzione della soglia per $k = 2$, $k = 4$, primo scenario.

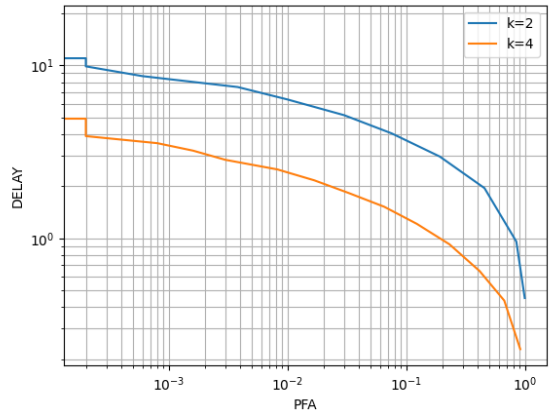


Figura 4.12: Curva di PFA-DELAY Tradeoff per $k = 2$, $k = 4$, primo scenario.

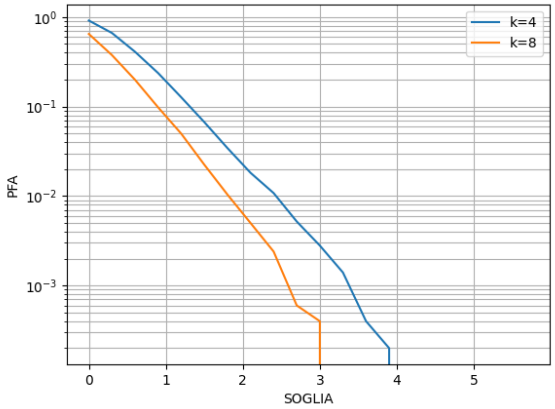


Figura 4.13: PFA in funzione della soglia per $k = 4$, $k = 8$, secondo scenario. La PFA diminuisce leggermente rispetto allo scenario precedente testato sullo stesso algoritmo.

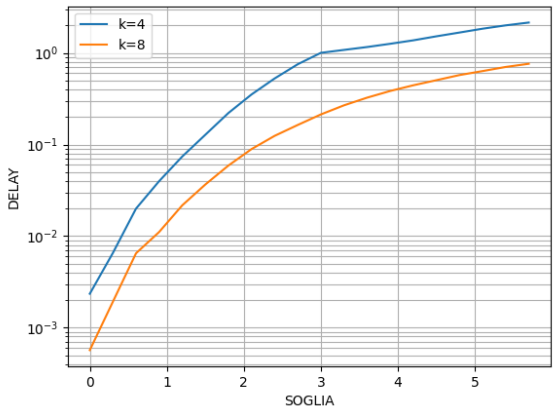


Figura 4.14: DELAY in funzione della soglia per $k = 4$, $k = 8$, secondo scenario.

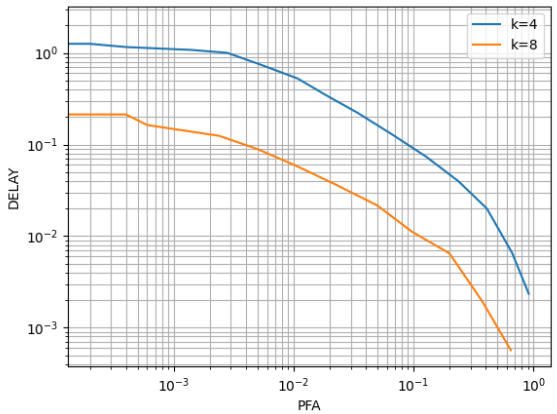


Figura 4.15: Curva di PFA-DELAY Tradeoff in funzione per $k = 4$, $k = 8$, secondo scenario.

4.4.1 Osservazioni

La principale differenza rispetto al contesto analogo associato all'algoritmo precedente riguarda il miglioramento delle performance in relazione alla PFA. Infatti, se nel caso precedente i sample esterni al supporto definito per il calcolo dell'LR (sia legittimi che illegittimi) venivano considerati tutti illegittimi e veniva segnalata un'anomalia, in questo caso tutti i sample appartengono esclusivamente al supporto di definizione della distribuzione uniforme essendo le distribuzioni gaussiane troncate e normalizzate. Ciò permette di evitare la segnalazione di anomalie in relazione a potenziali sample legittimi che invece vengono generati a partire da una gaussiana completa. L'utilizzo di un supporto più ampio rimane comunque una scelta migliore in relazione alle prestazioni generali e il tradeoff fra PFA e DELAY.

Tutti i valori analizzati in questa sezione per il valore di $k = 8$ risultano essere pressochè uguali rispetto al caso precedente per l'analogo valore di k. Il motivo per cui ciò avviene è che nel caso delle distribuzioni non troncate, la probabilità di generare sample legittimi che vengono scartati dall'algoritmo è quasi nulla. Infatti, il fattore di scala che viene definito nell'algoritmo che utilizza una distribuzione legittima troncata tende proprio a 1. Ciò fa si che per tale valore di k, i due casi siano praticamente identici fra loro, grazie al fatto che pochissimi sample legittimi vengono scartati dal primo dei due algoritmi.

4.5 Algoritmo CUSUM UWAC con sample multidimensionale

Applicando il teorema del CUSUM ad un sample multidimensionale, nonostante la forzatura dell'ipotesi associata alla distribuzione illegittima, le performance generali sono notevolmente migliorate. Ciò è dovuto al calcolo dell'LR relativo ai sample multidimensionali, ovvero vettori di molteplici osservazioni, come somma dei singoli LR valutati su ogni elemento dell' n-esimo sample (3.7).

Ciò fa sì che l'LR assuma valori più elevati per sample valutati come illegittimi e molto bassi per campioni considerati legittimi, ottenendo prestazioni migliori in termini di ritardo di segnalazione.

Per quanto riguarda la struttura delle sequenze anche in questo contesto ha subito delle modifiche, modificando quello sviluppato nella sezione precedente. Ogni sample viene descritto non più da una singola osservazione ma da un vettore di tre elementi ognuno dei quali appartiene ad una delle due distribuzioni legittima o illegittima (rappresentate da distribuzioni gaussiane).

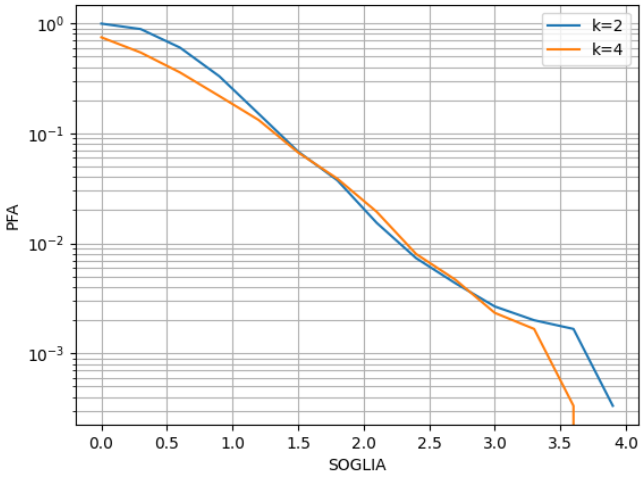


Figura 4.16: PFA in funzione della soglia per $k = 2$, $k = 4$, primo scenario.

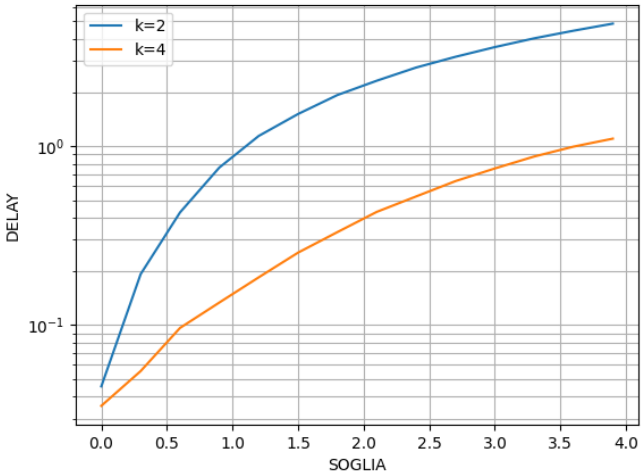


Figura 4.17: DELAY in funzione della soglia per $k = 2$, $k = 4$, primo scenario.

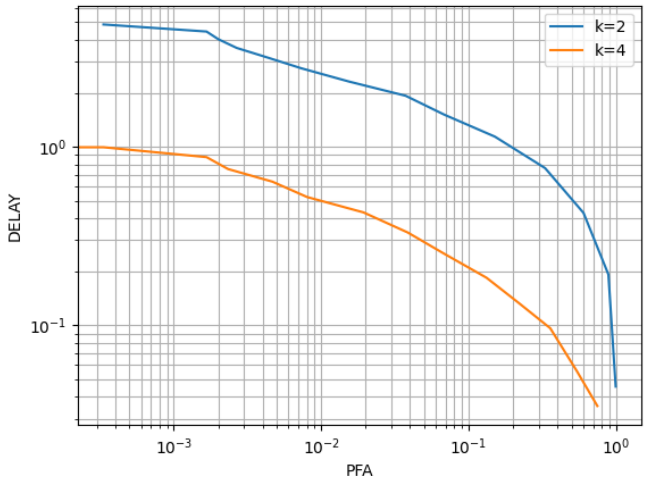


Figura 4.18: Curva di PFA-DELAY Tradeoff per $k = 2$, $k = 4$, primo scenario.

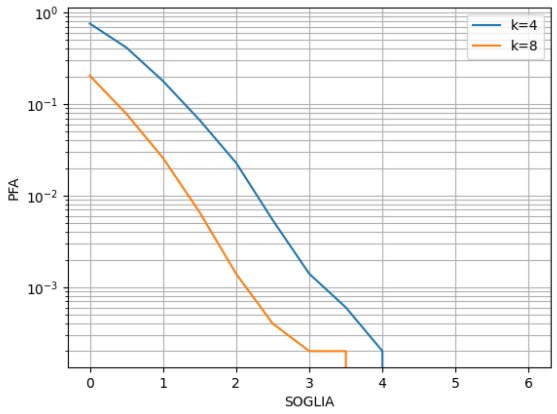


Figura 4.19: PFA in funzione della soglia per $k = 4$, $k = 8$, secondo scenario.

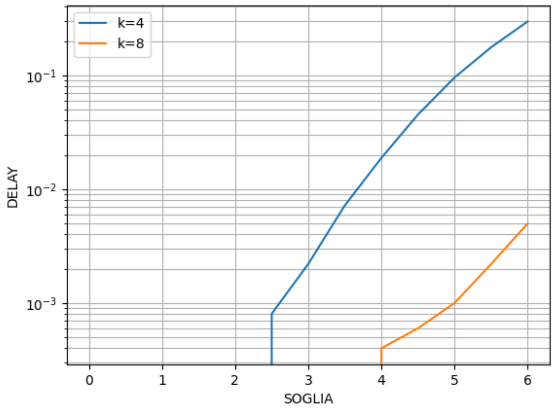


Figura 4.20: DELAY in funzione della soglia per $k = 4$, $k = 8$, secondo scenario.

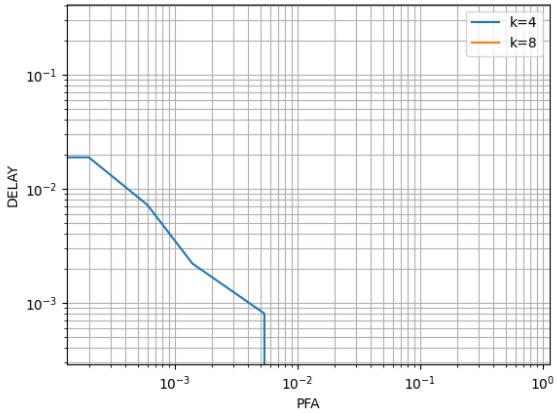


Figura 4.21: Curva di PFA-DELAY Tradeoff in funzione della soglia per $k = 4$, $k = 8$, secondo scenario. Possiamo notare che per $k = 8$ per valori non nulli di PFA il DELAY di segnalazione sia nullo, ragione per cui la curva di PFA-DELAY Tradeoff per tale valore di k non è rappresentata.

4.5.1 Osservazioni

Quello che possiamo notare dagli indicatori rappresentati nei grafici appena visualizzati è, a parità di scenario, un miglioramento del DELAY di segnalazione rispetto alle tecniche precedenti applicate allo stesso scenario. La ragione va ricercata nel calcolo dell'LR di un sample multidimensionale come somma dei singoli LR valutati su ogni campione del vettore che permette di aumentare più velocemente la somma cumulativa.

4.6 Algoritmo CUSUM UWAC con distribuzioni troncate e saturate

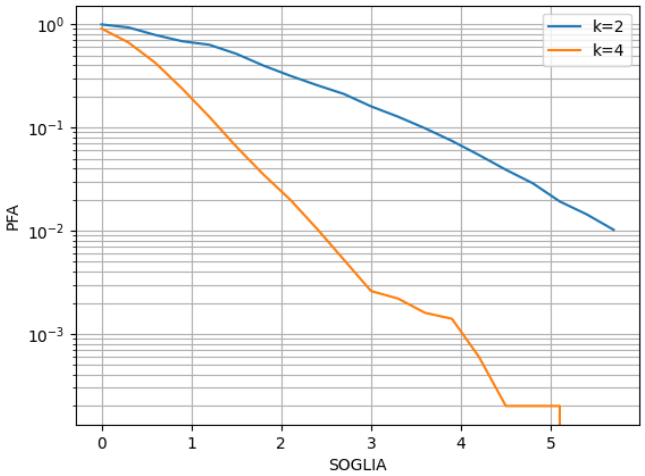


Figura 4.22: PFA in funzione della soglia per $k = 2$, $k = 4$, primo scenario.

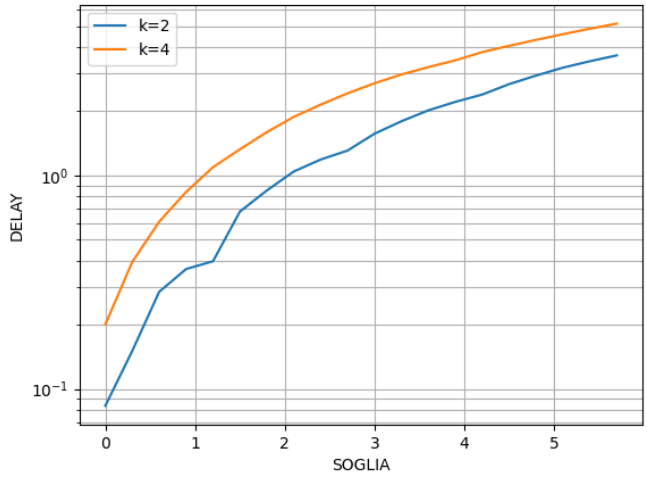


Figura 4.23: DELAY in funzione della soglia per $k = 2$, $k = 4$, primo scenario.

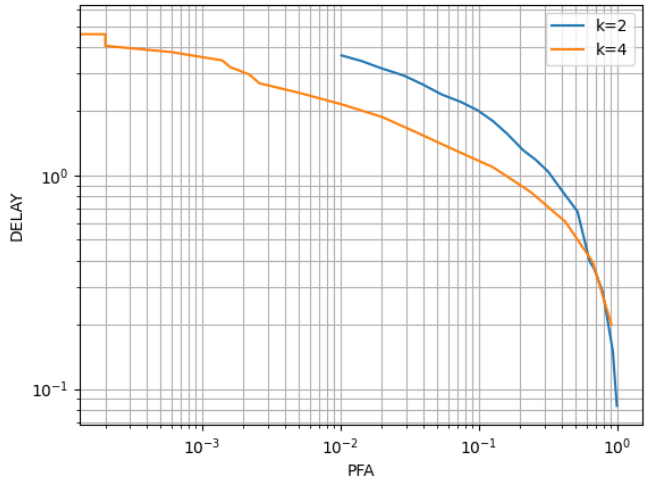


Figura 4.24: Curva di PFA-DELAY Tradeoff, primo scenario.

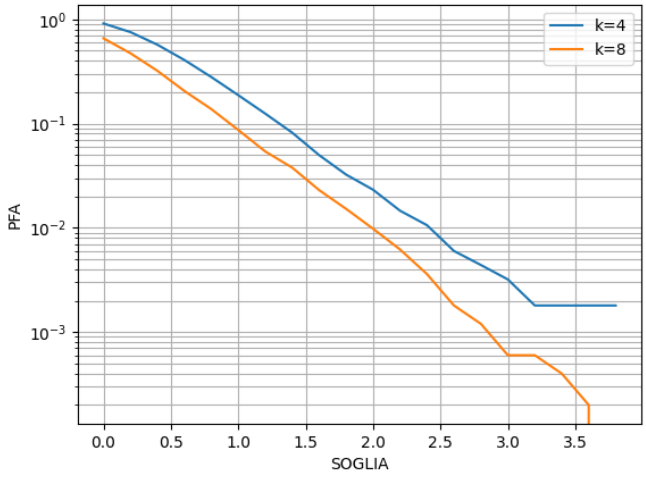


Figura 4.25: PFA in funzione della soglia per $k = 4$, $k = 8$, secondo scenario.

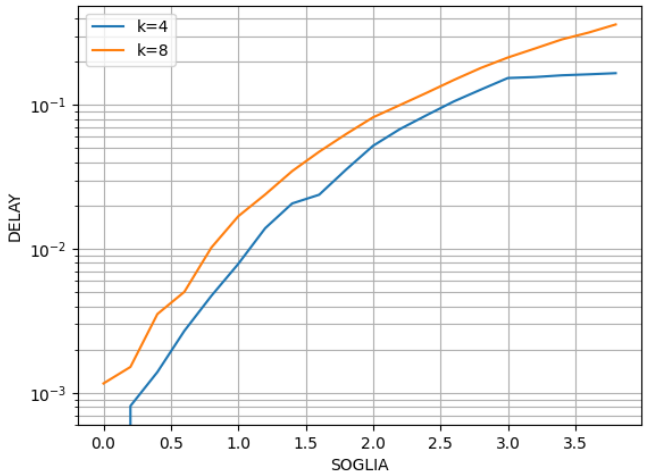


Figura 4.26: DELAY in funzione della soglia per $k = 4$, $k = 8$, secondo scenario.

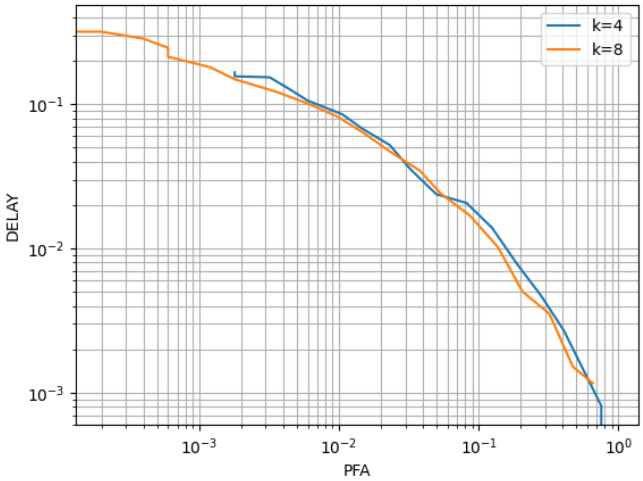


Figura 4.27: Curva di PFA-DELAY Tradeoff, secondo scenario.

4.6.1 Osservazioni

Dall'analisi dei risultati precedenti possiamo notare degli aspetti interessanti e differenti che questa soluzione offre rispetto alla normalizzazione delle distribuzioni in ingresso al ricevitore. Innanzitutto la PFA, a parità di scenario e valore della variabile k risulta essere peggiore, e ciò è parecchio evidente per valori di k bassi. La ragione per cui ciò avviene è probabilmente legata al calcolo dell'LR nel valore di saturazione. Per supporti più stretti, la probabilità di saturazione è molto elevata sia per la distribuzione legittima che illegittima, e ciò rende complicata la valutazione dei sample che vengono saturati agli estremi del supporto, considerando che il valore della funzione densità di probabilità valutata in saturazione risulta essere elevato, rendendo meno accurata la valutazione di tali sample nell'algoritmo CUSUM UWAC, sollevando più falsi allarmi.
Inoltre, il DELAY di segnalazione risulta essere più elevato per valori di k maggiori, al contrario di ciò che accade nel test effettuato con le distribuzioni normalizzate: ciò accade perchè nel calcolo dell'LR la distribuzione uniforme varia in relazione al valore di k, ma la distribuzione legittima all'interno dell'intervallo di saturazione rimane invariata non subendo normalizzazioni. Per valori elevati del supporto il ritardo di segnalazione è molto simile a quello rilevato utilizzando in input delle distribuzioni gaussiane troncate e normalizzate. Al contrario questo approccio sembra essere vincente rispetto al precedente per valori inferiori del supporto, sempre per il fatto che il fattore di scala delle distribuzioni normalizzate è più elevato e nel calcolo dell'LR domina rispetto alla distribuzione uniforme.

Un altro aspetto interessante lo si riscontra nell'analisi delle curve di PFA-DELAY Tradeoff, dove possiamo vedere che in questo contesto per entrambi gli scenari risultano essere molto simili nonostante i valori di k differenti. Questo è dettato probabilmente dal fatto che il numero di sample in saturazione non varia molto tra i due valori di k (soprattutto per il secondo scenario), e le distribuzioni valutate nell'intervallo stabilito non cambiano.

4.7 Algoritmo implementato con rete neurale

4.7.1 Struttura del modello

La rete è stata implementata attraverso la libreria TensorFlow ed è costituita da un primo strato LSTM con 64 neuroni seguito da un secondo strato completamente connesso con un singolo neurone con funzione di attivazione sigmoide, ideale per produrre un risultato compreso tra 0 e 1 per valutare ogni singolo campione. Successivamente, l'output fornito dal modello viene sogliato con un valore compreso sempre tra 0 e 1 per etichettare i sample analizzati: al di sopra della soglia i campioni vengono considerati potenzialmente illegittimi. La rete è addestrata utilizzando l'ottimizzatore Adam con la funzione di perdita *binary crossentropy*, specifica per compiti di classificazione binaria, e la metrica di *accuracy* per monitorare le performance. Il dataset viene suddiviso in training set (90%) e test set (10%) mediante la funzione `train_test_split()`, con un processo di addestramento che prevede 10 epoche e un `batch_size` pari a 10.

4.7.2 Dataset di training

Come già accennato nel capitolo precedente, il dataset di training è stato realizzato attraverso la definizione di 10000 sequenze da 40 elementi, ognuna delle quali presenta un istante in cui avviene il cambiamento di stato tra comunicazione legittima e illegittima. Questo valore viene generato attraverso una distribuzione uniforme appartenente all'intervallo $[10, 30]$, ciò per avere un numero simile di sample legittimi e illegittimi. Il modello, comunque, riceverà in input singoli sample e non intere sequenze. La distribuzione legittima risulta essere una distribuzione gaussiana, mentre quella illegittima in fase di training è una distribuzione uniforme appartenente ad un intervallo definito in fase di progettazione (in questo caso $[-5, 5]$). La scelta della distribuzione uniforme è stata presa in relazione alle ipotesi utilizzate nei precedenti algoritmi statistici, per poterne eseguire un confronto il più accurato possibile, oltre al fatto che non abbiamo alcuna informazioni sulla segnalazione illegittima.

Tabella 4.1: Confronto con algoritmo CUSUM UWAC con distribuzioni non troncate, secondo scenario.

Metriche	**Rete neurale**	**CUSUM UWAC**
PFA	$7,20 \times 10^{-3}$	$\approx 7 \times 10^{-3}$
DELAY (Tempo di Ritardo)	$5,62 \times 10^{-3}$	$\approx 10^{-1}$

4.7.3 Osservazioni

La rete neurale LSTM ha offerto delle prestazioni molto interessanti. Nonostante la PFA sia molto bassa il DELAY rimane comunque competitivo ($5,6 \times 10^{-2}$). Ciò significa che nella maggioranza delle sequenze considerate il primo sample illegittimo viene valutato dal modello come tale. Inoltre possiamo notare dalla figura 4.27 che la metrica del ritardo di segnalazione a parità di PFA sia migliore rispetto all'algoritmo CUSUM UWAC con distribuzioni non troncate, algoritmo che riceve in input un dataset dalla struttura analoga a quello utilizzato per analizzare il modello realizzato con rete neurale, ovvero delle distribuzioni gaussiane complete per entrambe le segnalazioni legittima e illegittima.

Bisogna considerare però che il modello realizzato attraverso una rete neurale non offre un vero e proprio tradeoff tra PFA e DELAY: le performance rimangono le stesse se la struttura del modello non varia. L'utilizzo dell'algoritmo CUSUM UWAC al contrario offre la possibilità di prediligere metriche differenti: variando solamente il valore della soglia ma mantenendo la stessa struttura dell'algoritmo è possibile ottenere tradeoff differenti, e ciò può essere sicuramente un vantaggio se si vuole favorire una metrica rispetto alle

altre, ad esempio mantenendo una PFA molto bassa a discapito di un DELAY elevato.

Capitolo 5

Conclusioni

In questa tesi abbiamo analizzato le problematiche legate alla Physical Layer Authentication (PLA) applicata all'*Underwater Acoustic Channel* (UWAC). Essa si occupa di autenticare la provenienza di un segnale trasmetto attraverso un canale in relazione alle proprietà fisiche legate alla segnalazione stessa e al canale che viene utilizzato. Il canale acustico subacqueo risulta essere un modello molto complesso da analizzare e modellare poichè il comportamento risulta essere molto variabile in relazione a proprietà fisiche del mezzo di propagazione e caratteristiche morfologiche delle aree in cui avviene la comunicazione, oltre alla presenza di fenomeni come rifrazione o interferenze che possono essere presenti in tale ambiente. Innanzitutto, abbiamo introdotto delle tecniche di carattere statistico per risolvere il problema della *quickest detection*, il cui obiettivo è quello di rilevare attacchi di spoofing da parte di un trasmettitore illegittimo nel minor tempo possibile. A tal proposito abbiamo introdotto il teorema CUSUM, una tecnica dimostrata ottima in ottica di rilevamento di anomalie nel momento in cui si conoscono le distri-

buzioni associate ad entrambe le segnalazioni in cui i campioni sono i.i.d.. Nel contesto in cui ci siamo posti, però, non è possibile applicare questa tecnica in quanto la distribuzione dell'attaccante non è nota al difensore. Per poter utilizzare tale approccio abbiamo introdotto quindi il CUSUM UWAC: non conoscendo la distribuzione illegittima abbiamo ipotizzato che l'attaccante si comporti secondo una distribuzione uniforme in un intervallo fissato a priori in fase di progettazione. In relazione al comportamento del ricevitore legittimo, infine, abbiamo ipotizzato algoritmi differenti per confrontare approcci differenti al problema, definendo degli scenari di trasmissione basati su distribuzioni gaussiane per entrambe le trasmissioni.

In particolare, sono stati realizzati test dell'algoritmo CUSUM UWAC su distribuzioni gaussiane complete, per poi ipotizzare che il ricevitore ignorasse i sample esterni al supporto della distribuzione uniforme progettato così da poter valutare ogni singolo campione ricevuto. L'algoritmo CUSUM UWAC è stato testato su distribuzioni gaussiane troncate e normalizzate sia per sample singoli che sample multidimensionali (ovvero vettori di campioni ricevuti in ingresso al ricevitore). Abbiamo effettuato un test utilizzando distribuzioni gaussiane saturate in relazione al supporto della distribuzione illegittima ipotizzata nell'algoritmo CUSUM.

Infine, abbiamo cercato di risolvere il problema della *anomaly detection* sviluppando un algoritmo realizzato attraverso una rete neurale LSTM che fosse in grado di effettuare classificazione binaria sui sample per rilevare anomalie all'interno delle sequenze, che ha riscontrato ottime performance in relazione agli algoritmi statistici sviluppati precedentemente.

Bibliografia

[1] L. Cardillo, "On the quickest detection problem for authentication in underwater acoustic channels," *University of Padua*, 2023.