



Università degli Studi di Salerno

Corso di Statistica e Analisi dei Dati

# **Analisi statistica di caratteristiche e distribuzioni in un dataset di siti web legittimi e di phishing**

Autore: Nicola Frugieri

Matricola: 0522501966

# Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Analisi del dataset</b>	<b>3</b>
2.1	Prime analisi e studio dell'URL . . . . .	3
2.2	Studio del TLD . . . . .	7
2.3	Offuscamento dell'URL . . . . .	20
2.4	Studio dei caratteri degli URL . . . . .	24
2.5	L'impiego degli URL shortener . . . . .	37
2.6	Frequenza e distribuzione dei caratteri . . . . .	38
2.7	Impiego del protocollo HTTPS . . . . .	40
2.8	Caratteristiche del codice HTML . . . . .	41
2.8.1	Righe di codice . . . . .	41
2.8.2	Titolo della pagina . . . . .	44
2.8.3	Caratteristiche estetiche ed informazioni . . . . .	52
2.8.4	Campi per immissione di dati e riferimenti bancari . . . . .	54
2.8.5	Redirect, riferimenti esterni e popup . . . . .	55
2.8.6	Codice CSS e JS . . . . .	59
2.8.7	Altre colonne . . . . .	60
<b>3</b>	<b>Classificazione degli URL con SVM</b>	<b>61</b>
3.1	Scelta delle feature e pre-processing dei dati . . . . .	61
3.2	Creazione del modello . . . . .	62
3.3	Visualizzazione dei dati . . . . .	63
<b>4</b>	<b>LLM e dati sintetici</b>	<b>66</b>
4.1	Obiettivo . . . . .	66
4.2	LLM utilizzato e prompt . . . . .	66
4.2.1	Lunghezza dell'URL e del dominio . . . . .	67
4.2.2	Confronto dei TLD . . . . .	71
4.3	Punti deboli del modello . . . . .	72
4.4	Conclusioni . . . . .	76

# 1 Introduzione

Il phishing è una tecnica di ingegneria sociale utilizzata per ingannare gli utenti e indurli a fornire informazioni sensibili, come credenziali di accesso, dati bancari o numeri di carte di credito.

Solitamente, i cybercriminali mascherano le proprie comunicazioni da fonti affidabili, come banche, istituzioni finanziarie o siti web conosciuti, con l'obiettivo di convincere le vittime a cliccare su link o scaricare allegati dannosi.

Questo fenomeno rappresenta una delle minacce più diffuse nel campo della sicurezza informatica, dato che i truffatori continuano ad affinare le proprie tecniche per rendere i siti e le email di phishing sempre più simili a quelli legittimi.

L'analisi statistica dei dati sui siti di phishing e legittimi, come quelli riguardanti l'uso di tecniche di offuscamento e altre caratteristiche distinte, ci permette di comprendere meglio questo fenomeno e può contribuire allo sviluppo di metodi efficaci di rilevamento e prevenzione, ad esempio creando modelli di machine learning in grado di identificare questi siti malevoli.

## 2 Analisi del dataset

Il dataset PhiUSIIL contiene 134.850 URL legittimi e 100.945 URL di phishing, per un totale di 235.795, ottenuti da fonti affidabili come PhishTank, OpenPhish e MalwareWorld. I dati

Categoria	Numero di URL
Phishing	100.945
Legittimi	134.850

Table 1: Tabella degli URL

presenti sono divisi in 56 colonne, principalmente numerici. In particolare, possiamo trovare:

- **Caratteristiche URL:** Lunghezza del dominio, presenza di caratteri offuscati, numero di sottodomini, e probabilità del TLD.
- **Caratteristiche HTML:** Linee di codice più lunghe, presenza di favicon, campi nascosti, o elementi come pulsanti di invio.
- **Caratteristiche derivate:** Continuity rate dei caratteri, punteggio di corrispondenza tra URL e titolo della pagina.

Nell'ambito di questo progetto, il dataset era inizialmente diviso in dieci file .csv. Questi sono stati dunque riuniti in un unico dataframe in R, chiamato *combined\_dataset*.

L'ultima colonna, denominata *label*, indica la legittimità dell'URL: assume valore 0 per gli URL di phishing e valore 1 per gli URL legittimi. Questa colonna sarà utilizzata per confrontare gli altri dati del dataset.

### 2.1 Prime analisi e studio dell'URL

Le prime colonne del dataset sono riferite alla stringa dell'URL, in particolare possiamo porre la nostra attenzione su URLLength, DomainLength e TLD. Le prime due si riferiscono rispettivamente alla lunghezza dell'URL e del dominio. Possiamo ipotizzare che gli URL di phishing abbiano una lunghezza media maggiore, a causa di alcune strategie che il phisher utilizza per ingannare gli utenti. Spesso, infatti, chi crea siti di phishing imita siti web legittimi e cerca dunque di imitare gli URL di questi ultimi. Un tipico modo per farlo è inserendo altre parole intorno all'URL legittimo (Es. se <https://www.poste.it/> è l'URL legittimo, <https://www.poste-italiane.it/> potrebbe essere un esempio di URL di phishing).

In particolare, queste parole aggiuntive possono essere termini per infondere fiducia, come "secure-login", "security-check" o simili.

Inoltre, spesso vengono usati sottodomini o parametri aggiuntivi e query che allungano notevolmente l'URL. Infine, potrebbe essere una strategia per non far leggere l'URL all'utente, che vedendo una stringa più lunga potrebbe essere distratto dalle parole presenti. Andiamo dunque ad analizzare la lunghezza degli URL e del dominio:

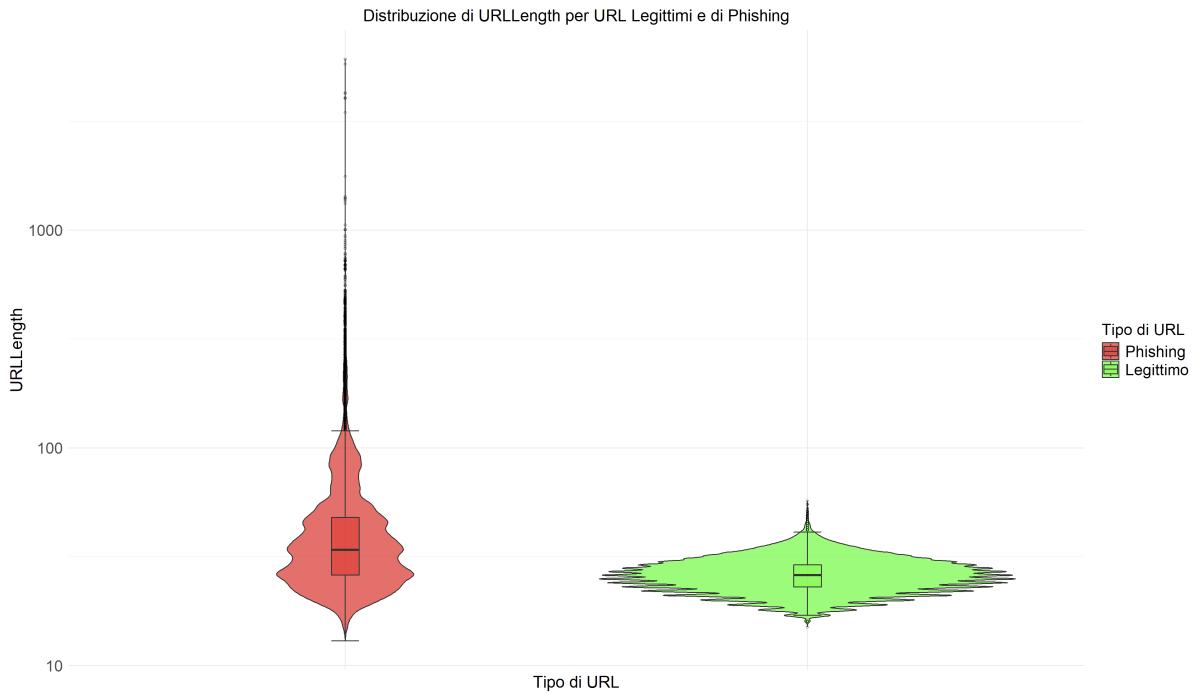


Figura 1: Confronto della distribuzione della lunghezza degli URL.

Come si può vedere dal grafico in figura 1, gli URL di phishing tendono ad avere una lunghezza maggiore. Nel grafico sono state mostrate sia la distribuzione completa delle lunghezze che quella dei valori più frequenti. Sebbene infatti alcuni URL di phishing arrivino a migliaia di caratteri, con il valore più alto che è 6097, la maggiore concentrazione si trova tra 13 (il minimo) e 130, in cui vi sono ben 98315 URL (97,4% degli URL di phishing). Nell'intervallo 500-1000 ricadono 104 URL di phishing, mentre superati i 1000 caratteri, soltanto 19 URL sono presenti. Gli URL legittimi, invece, sono completamente concentrati tra 15, il valore minimo, e 57, il massimo.

La scelta di utilizzare questo particolare tipo di grafico è dovuta alla variabilità e alla dispersione dei dati, da dover confrontare sempre tra URL legittimi e di phishing. In questo modo, è possibile visualizzare facilmente sia la distribuzione che le statistiche dei dati; inoltre sono indicati anche gli outlier. La scala logaritmica (o pseudologaritmica in alcuni casi, a causa dei valori 0) è stata scelta a causa della grande variabilità che i dati assumono, come è possibile vedere ad esempio dal grafico della lunghezza degli URL.

Per quanto riguarda la lunghezza del dominio, invece, questa risulta più contenuta, sebbene sia comunque mediamente maggiore per gli URL di phishing; il grafico in ogni caso è molto simile a quello della lunghezza degli URL. I domini si attestano tra i 4 e i 110 caratteri per gli URL di phishing, mentre quelli degli URL legittimi tra 8 e 50. Questo indica che non è il dominio a rappresentare la parte più lunga degli URL di phishing, ma, come detto precedentemente, la causa è da ricercarsi in altre componenti dell'URL.

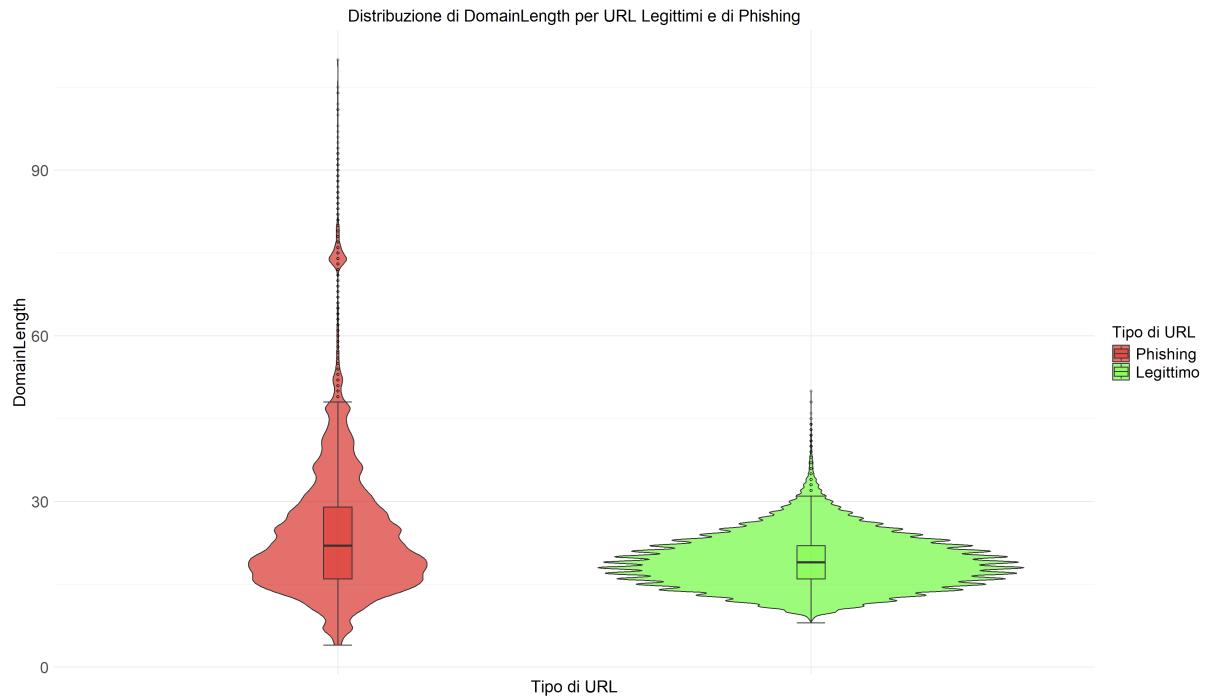


Figura 2: Grafico della distribuzione della lunghezza dei domini degli URL

Di seguito le statistiche della lunghezza degli URL e dei domini divisi per legittimità:

Statistiche	Siti di Phishing	Siti Legittimi
Min URLLength	13,00	15,00
1° Quart. URLLength	26,00	23,00
Mediana URLLength	34,00	26,00
Media URLLength	45,72	26,23
3° Quart. URLLength	48,00	29,00
Max URLLength	6097,00	57,00
Dev. st. URLLength	61,15	4,82
Min DomainLength	4,00	8,00
1° Quart. DomainLength	16,00	16,00
Mediana DomainLength	22,00	19,00
Media DomainLength	24,47	19,23
3° Quart. DomainLength	29,00	22,00
Max DomainLength	110,00	50,00
Dev. st. DomainLength	12,20	4,82

Table 2: Statistiche sulla lunghezza degli URL e dei domini per siti di phishing e legittimi

Come detto precedentemente, spesso gli URL di phishing ricorrono all'utilizzo di sottodomini per ingannare gli utenti, sfruttando parole o termini che possano infondere sicurezza e fiducia, oppure per imitare semplicemente l'URL originale.

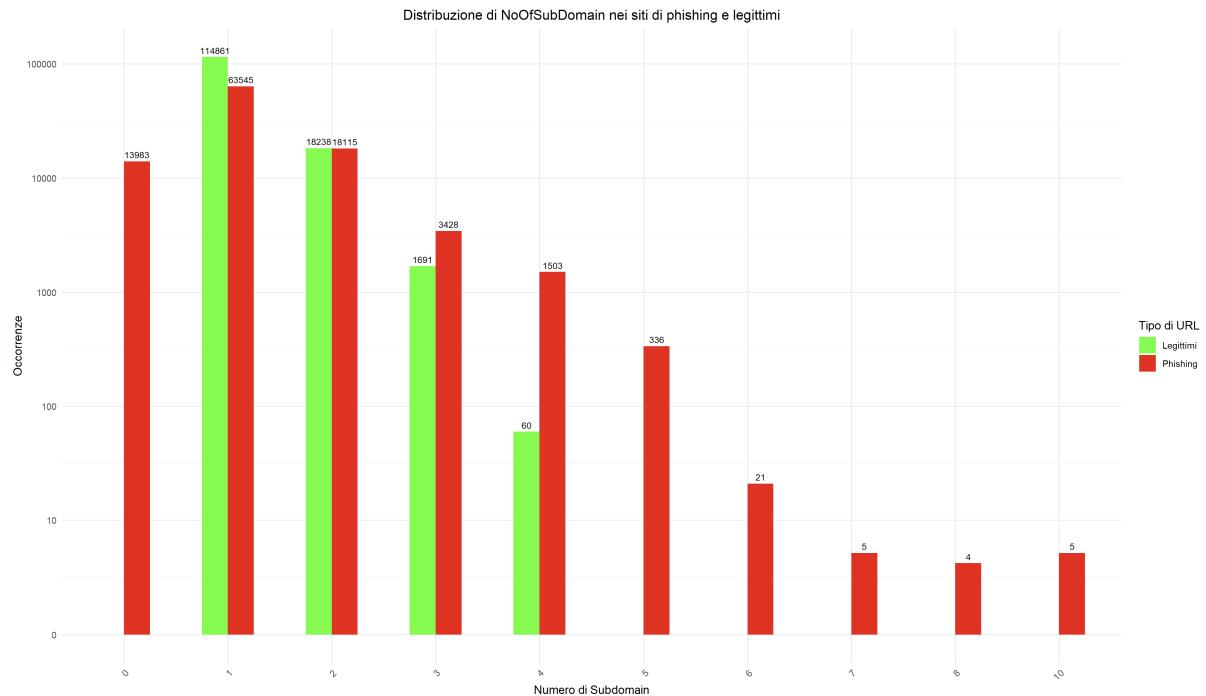


Figura 3: Grafico a barre che mostra il confronto della distribuzione del numero di sottodomini

Come evidenziato dal grafico (Figura 3), gli URL legittimi tendono a utilizzare principalmente un solo sottodominio, con la maggior parte dei casi compresa tra uno e quattro sottodomini.

Anche gli URL di phishing presentano un picco nell'utilizzo di un solo sottodominio, ma mostrano una distribuzione più ampia, variando principalmente da zero a cinque sottodomini, con alcuni casi che arrivano fino a dieci.

Dal dataset si evince che gli URL legittimi hanno solitamente almeno un dominio (Figura 3), mentre molti URL di phishing possono non averne affatto.

La colonna `URLSimilarityIndex` è invece una feature calcolata dai creatori del dataset. È un indice che misura la somiglianza tra un URL di origine (`source URL`) e un URL di destinazione (`target URL`). L'idea alla base di questo indice è di verificare quanto un URL somigli a un sito legittimo presente nella lista dei 10 milioni di siti legittimi più popolari. Se un URL ha un USI di 100, significa che è un URL legittimo. Se il valore è vicino a 100, potrebbe essere un sito di phishing che cerca di assomigliare a un sito famoso.

Una criticità della feature è che non è associata ad una colonna contenente gli URL sorgente, che dovrebbero essere quelli legittimi, dunque non possiamo capire se, come per altre colonne che vedremo in seguito, l'indice calcolato ha errori o anomalie.

Gli URL legittimi hanno ovviamente tutti valore 100, dunque prenderemo in considerazione solo quelli di phishing. Questa variabile non ci fornisce molte informazioni, ma possiamo farci un'idea di quanto generalmente gli URL di phishing tendano ad imitare URL legittimi, sostituendo alcuni caratteri per ingannare gli utenti:

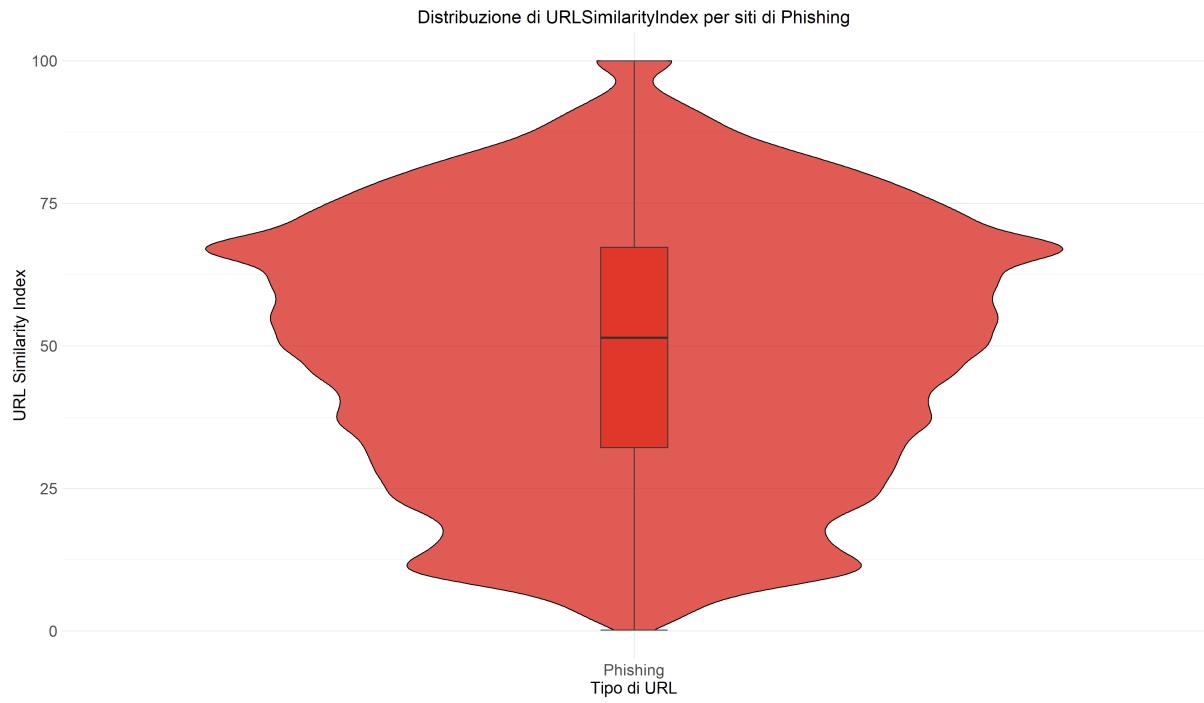


Figura 4: Grafico a violino che mostra la distribuzione della variabile URLSimilarityIndex negli URL di phishing

La distribuzione appare abbastanza omogenea, con un picco tra 60 e 75 e una contrazione vicino a 100, come ci si aspetterebbe. Non è chiaro come però possano esserci degli URL (786) con un punteggio di 100, nonostante siano fraudolenti. Questo lascia pensare che possano esserci alcuni siti web segnalati come fraudolenti nonostante siano legittimi, oppure che il punteggio sia sbagliato.

## 2.2 Studio del TLD

Una componente fondamentale dell'URL è il TLD, ovvero il dominio di primo livello, la parte finale della stringa, dopo l'ultimo punto. Possono essere principalmente di due tipi:

- **Nazionali:** usati da uno stato o una dipendenza territoriale, come `.it` o `.fr`.
- **Generici:** usati da aziende e organizzazioni di vario tipo. Il più famoso è `.com`. Ne esistono poi alcuni usati da governi o enti governativi come `.gov` o `.edu`

Solitamente, i phisher utilizzano determinati TLD per diversi motivi. Alcuni TLD offrono registrazioni a costi molto bassi o addirittura gratuite (es. `.tk`, `.ga`, `.ml`, `.cf`, `.gq`). Questo permette agli attaccanti di creare molti domini in modo economico, anche se vengono rapidamente bloccati. Inoltre, molti di questi TLD consentono registrazioni anonime, rendendo difficile identificare i responsabili. I TLD gestiti da registrar con politiche di controllo meno rigorose sono più vulnerabili all'abuso. I cybercriminali sfruttano infatti registrar che non verificano adeguatamente l'identità degli utenti o non monitorano attività sospette. Spesso, alcuni TLD regionali come `.cn` (Cina), `.ru` (Russia) o `.br` (Brasile) sono abusati, poiché alcuni registrar locali sono meno restrittivi. Alcuni TLD hanno un'alta disponibilità di domini con parole chiave popolari. Per esempio, un attaccante potrebbe registrare facilmente URL che imitano quelli legittimi, o che utilizzano parole che infondono fiducia, in TLD meno popolari, mentre su

TLD come .com, questi nomi sarebbero già occupati.

Infine, TLD come .com, .net o .org sono anch'essi frequentemente usati per phishing poiché sono familiari agli utenti e quindi più fidati; non deve dunque sorprendere se la maggior parte degli URL di phishing utilizza questi TLD.

Il dataset contiene URL con 695 TLD diversi, da un primo conteggio. Tuttavia, alcuni URL sono degli indirizzi IP, a cui è stato identificato come TLD l'ultimo gruppo di cifre. Non conoscendo la risoluzione DNS di questi URL, li scarteremo dal conteggio e dall'analisi del TLD. Inoltre, alcuni URL contengono anche la porta di rete a seguito del TLD, ad esempio

"[http://o8899ff.livewireremote.com:4000/p2u9==\\$/d@deltech-cs.com](http://o8899ff.livewireremote.com:4000/p2u9==$/d@deltech-cs.com)"; ciò ha causato la registrazione di TLD aggiuntivi come ".com:4000", invece di registrare gli URL con TLD ".com"; sono stati dunque normalizzati scartando la porta e includendoli nel conteggio dei TLD relativi. Dopo questa pulizia dei dati, il numero di TLD effettivi risulta essere **569**, con .com che è il più utilizzato, con 112.565 URL, quasi la metà (47,74%) dell'intero dataset (235.795 URL totali).

Il dataset fornisce una colonna, denominata `TLDLength`, che può darci un'idea di come gli URL di phishing utilizzino spesso TLD meno noti, più lunghi dei classici TLD a due o tre caratteri.

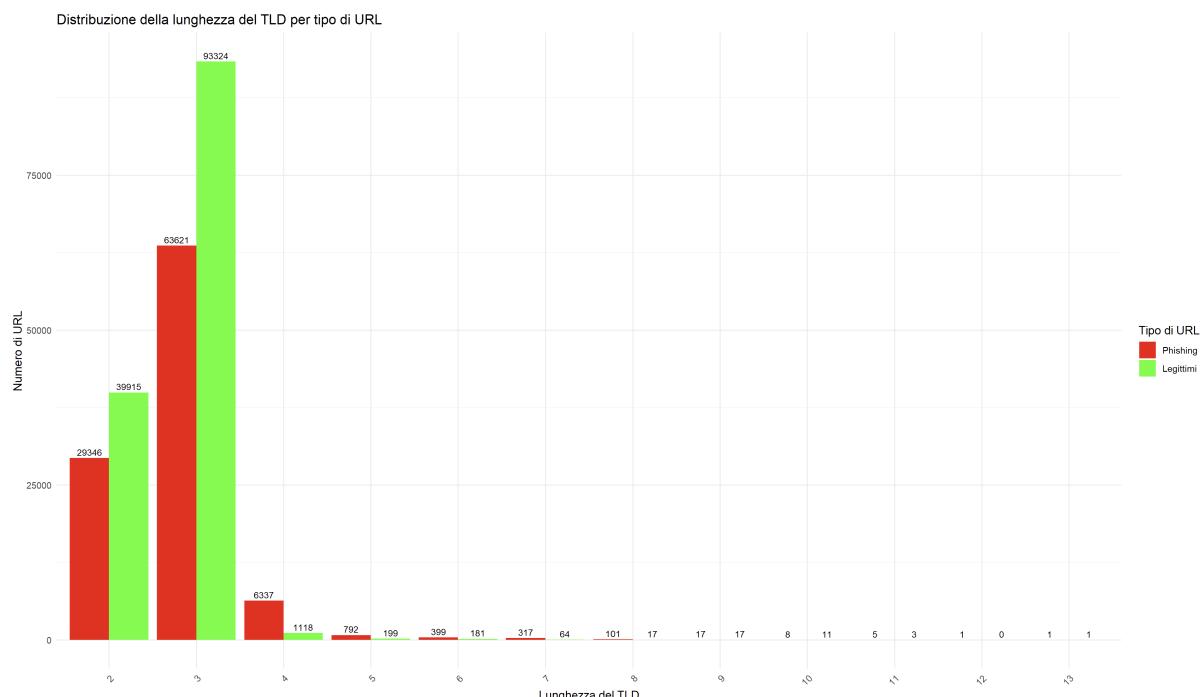


Figura 5: Grafico a barre della lunghezza dei TLD, confrontato per URL legittimi e di phishing

Sebbene dunque la maggior parte degli URL di phishing tenda comunque ad utilizzare i TLD più comuni, una buona fetta utilizza TLD di lunghezza maggiore, spesso poco utilizzati dai siti web legittimi.

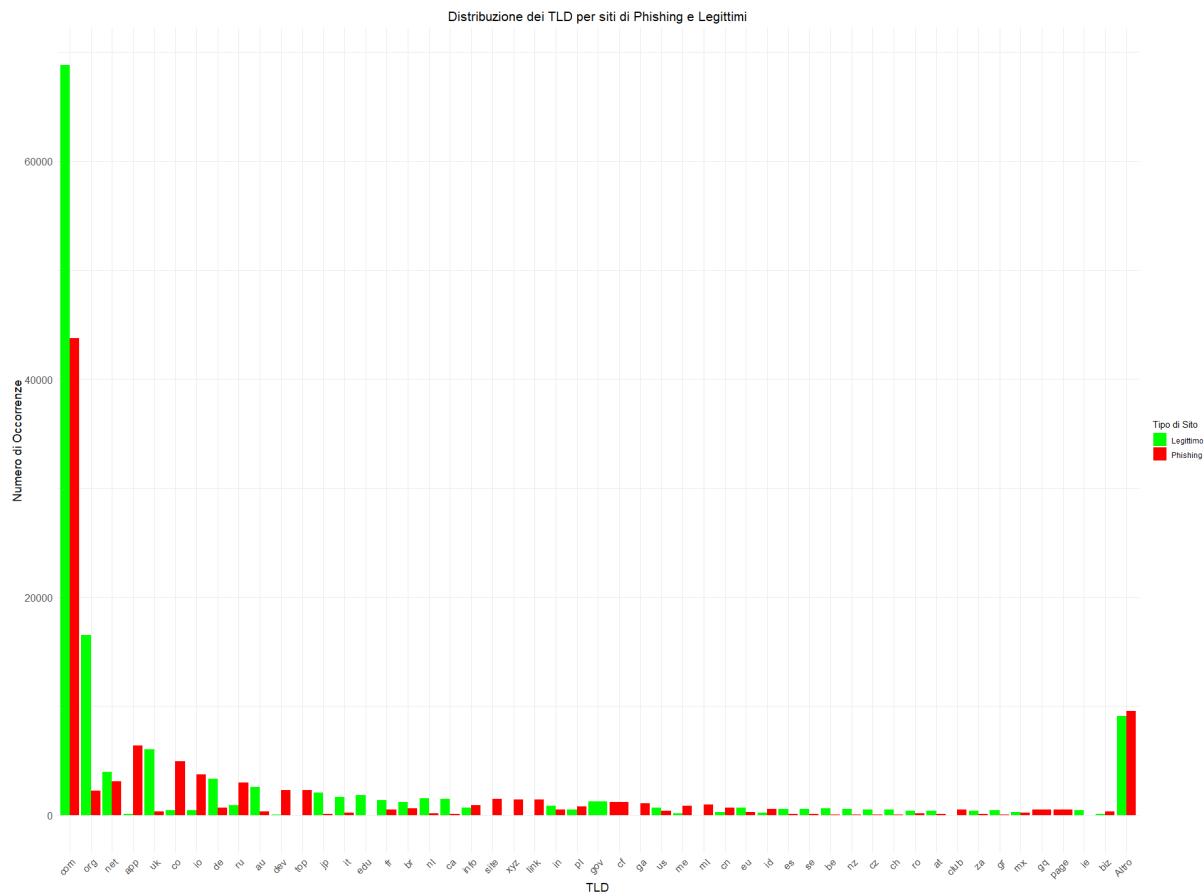


Figura 6: Grafico a barre dei primi 50 TLD più utilizzati con confronto tra URL legittimi e di phishing + Altro

Dal grafico 6 possiamo notare immediatamente come la maggior parte degli URL ricada nei primi 50 TLD, in particolare `.com` è usato per la maggior parte degli URL legittimi, ma anche di phishing.

Ciò che salta all'occhio è la distribuzione molto concentrata dei TLD. Infatti, i primi 50 TLD hanno in totale 216.531 URL, rappresentando il 91,83% del dataset.

Distribuzione dei primi 15 TLD tra gli URL + Altro

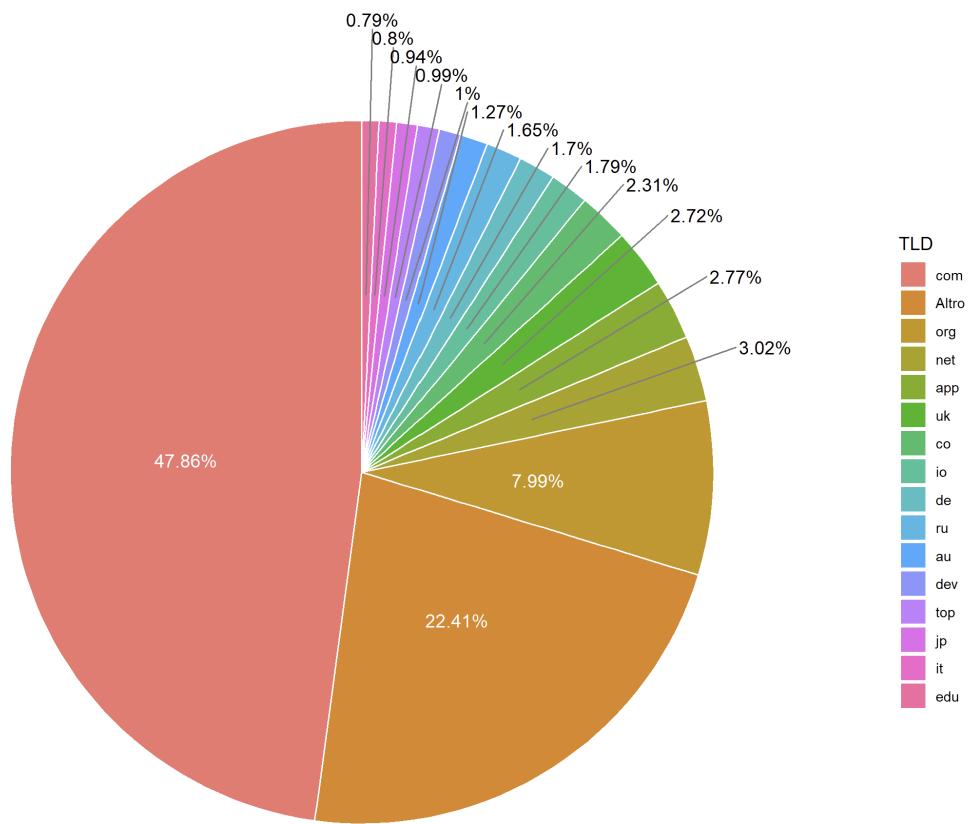


Figura 7: Grafico a torta della distribuzione dei primi 15 TLD

Distribuzione dei TLD tra gli URL di Phishing

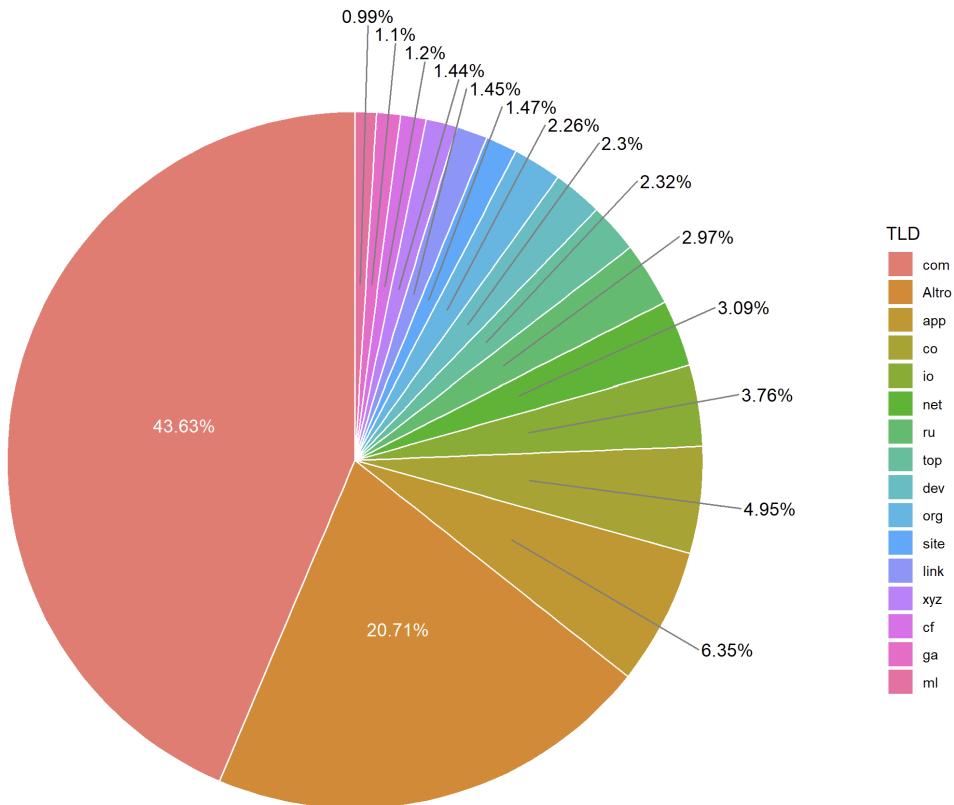


Figura 8: Grafico a torta della distribuzione degli URL di phishing dei primi 15 TLD + Altro

La concentrazione nel dataset è importante: il valore medio di URL per TLD è 413 ma il valore mediano è 8.

Min	Q1	Media	Mediana	Moda	Q3	Max	Dev. Standard
1	2	233.34 (60.76%)	7 (63.30%)	1	34	43780	2172.41

Table 3: Statistiche degli URL di phishing per TLD. Media e mediana degli URL di phishing per TLD sono state riportate anche in forma percentuale.

Ricavando un barplot dei TLD, ordinato in base agli URL di phishing, possiamo studiare quali siano questi TLD maggiormente usati in maniera maliziosa. Sebbene `.com` risultino essere comunque il primo, notiamo come in valori assoluti, i TLD più utilizzati siano comunque quelli maggiormente colpiti dal fenomeno degli URL di phishing.



Figura 9: Grafico a barre dei TLD con più URL di phishing in termini assoluti

È interessante notare, tuttavia, che ci sono dei TLD che hanno una maggioranza di URL di phishing, o addirittura vengono usati esclusivamente per il phishing.

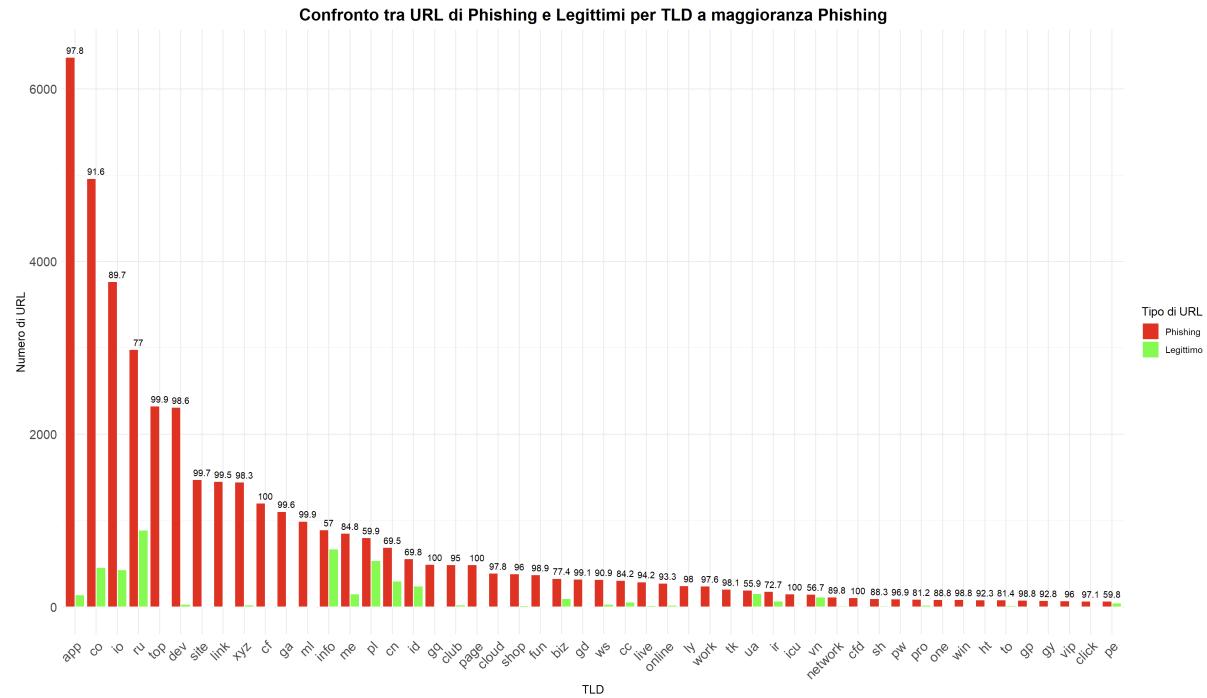


Figura 10: Grafico a barre dei TLD con più URL di phishing rispetto a quelli legittimi. Il valore su ogni barra indica la percentuale di URL di phishing rispetto al totale per ogni TLD.

Il dataset contiene, inoltre, una colonna denominata `TLDLegitimateProb`, che dovrebbe indicare la probabilità di un TLD di essere più o meno utilizzato da siti web legittimi. Tuttavia, è poco chiara la sua efficacia, in quanto il suo valore è calcolato prendendo la lista dei top 10 milioni di siti web e contando le occorrenze dei TLD; dopodiché è stato diviso il numero di occorrenze di ogni TLD per il numero totale di occorrenze di tutti i TLD. Secondo i creatori del dataset, dunque, un maggior `TLDLegitimateProb` potrebbe indicare un URL legittimo, mentre un punteggio basso potrebbe indicare un URL fraudolento.

Dal dataset, tuttavia, la correlazione non risulta così forte. Per studiarla, è stata calcolata la proporzione di URL di phishing per ogni TLD per valutare il comportamento di `TLDLegitimateProb` al variare della percentuale di URL di phishing. A questo proposito è stata ricavata la variabile `PhishingRatio`, calcolando il rapporto tra gli URL di phishing e il numero totale di URL per un determinato TLD.

### Pearson's product-moment correlation

```
data: tld_summary$TLDLegitimateProb and tld_summary$PhishingRatio
t = -0.55086, df = 567, p-value = 0.5819
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
-0.1051255  0.0591823
sample estimates:
cor
-0.02312776
```

Figura 11: Test della correlazione tra TLDLegitimateProb e la percentuale di URL di phishing per ogni TLD.

Il valore di correlazione è molto vicino a zero, indicando nessuna correlazione significativa tra TLDLegitimateProb e *PhishingRatio*. Un valore negativo significherebbe che all'aumentare di TLDLegitimateProb, la percentuale di URL di phishing tende a diminuire, ma qui l'effetto è praticamente inesistente. Il P-value ( $p = 0.5819$ ) indica la probabilità di ottenere questi risultati se la correlazione vera fosse zero (ipotesi nulla). Con un p-value molto alto ( $> 0.05$ ), non possiamo rifiutare l'ipotesi nulla, ovvero non c'è evidenza statistica che le due variabili siano correlate. Il valore  $t$  misura la forza della relazione tra le variabili. Un valore basso indica una relazione molto debole.

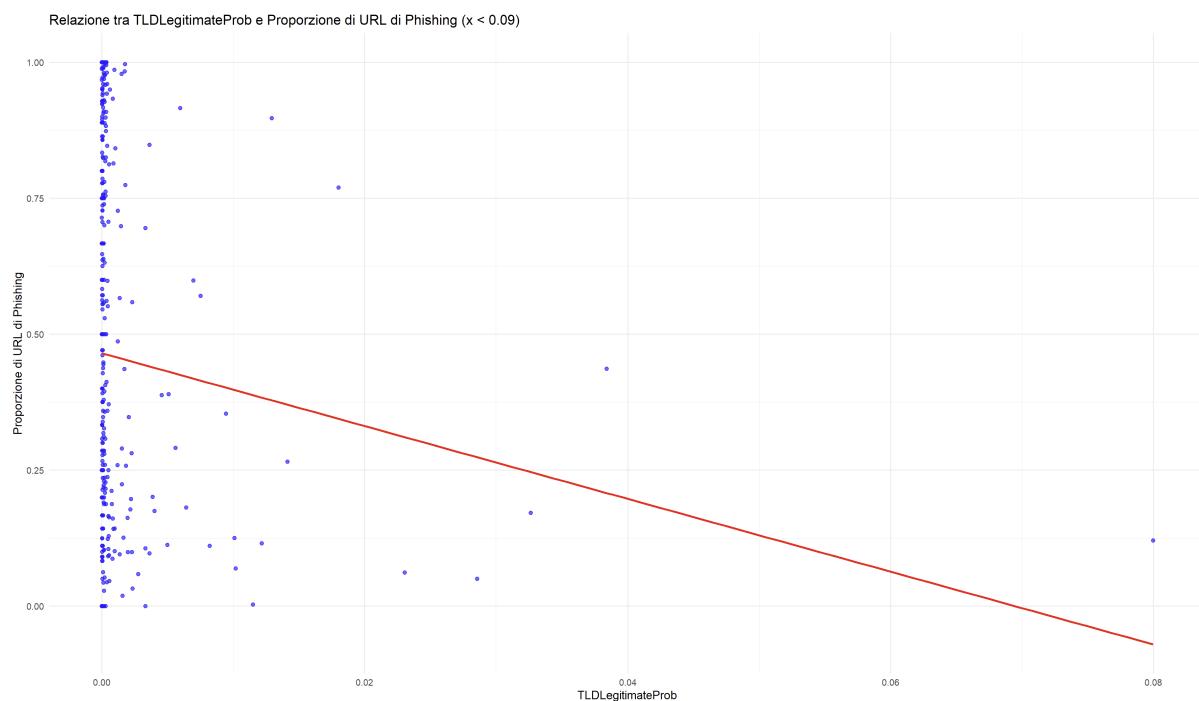


Figura 12: Scatterplot che mostra la correlazione tra TLDLegitimateProb e la percentuale di URL di phishing per ogni TLD. Per una migliore chiarezza, l'asse x è stata limitata a 0.09, in quanto tutti i valori sono compresi tra 0 e 0.09, tranne .com, che è l'unico TLD con una TLDLegitimateProb di 0.52.

Come si evince dal grafico in Figura 12, molti TLD, pur avendo una bassissima percentuale di URL di phishing, risultano avere una TLDLegitimateProb pari o vicina a 0. Inoltre, tutti i TLD possiedono una TLDLegitimateProb compresa tra 0 e 0.09, fatta eccezione per .com, con

0.52, considerabile dunque un outlier. Questa grande differenza è però dovuta alla frequenza di utilizzo di `.com` rispetto agli altri TLD, che vengono penalizzati per il minor utilizzo. A causa di ciò, anche TLD usati solo per URL legittimi, come `.gov`, o quasi, risultano avere una probabilità di legittimità molto bassa.

<b>Rank</b>	<b>TLD</b>	<b>TLDLegitimateProb</b>	<b>PhishingRatio</b>	<b>Count</b>
1	com	0.523	0.389	112565
2	org	0.0800	0.121	18793
3	net	0.0384	0.437	7097
4	de	0.0327	0.172	3996
5	uk	0.0286	0.0504	6395
6	jp	0.0230	0.0617	2219
7	ru	0.0180	0.770	3875
8	fr	0.0141	0.266	1860
9	io	0.0129	0.897	4201
10	it	0.0122	0.116	1887
11	edu	0.0115	0.00269	1861
12	ca	0.0102	0.0694	1614
13	au	0.0101	0.125	2979
14	br	0.00944	0.354	1846
15	nl	0.00820	0.111	1727
16	info	0.00751	0.570	1566
17	pl	0.00696	0.599	1340
18	es	0.00641	0.182	705
19	co	0.00598	0.916	5422
20	eu	0.00562	0.291	980
21	in	0.00508	0.389	1379
22	ch	0.00498	0.112	560
23	us	0.00456	0.388	1108
24	se	0.00402	0.175	704
25	at	0.00388	0.201	517
26	me	0.00364	0.848	1006
27	cz	0.00363	0.0971	587
28	gov	0.00333	0	1276
29	cn	0.00332	0.695	993
30	be	0.00332	0.106	696

Table 4: Classifica dei primi 30 TLD ordinati per TLDLegitimateProb.

La Tabella 4 ci mostra come la maggior parte dei TLD abbiano una TLDLegitimateProb molto bassa, nonostante il bassissimo rapporto di URL di phishing, come ad esempio i TLD `.edu` e `.gov`, che pur avendo un *PhishingRatio* rispettivamente di 0,0027 e 0, risultano avere una TLDLegitimateProb di 0,0115 e 0,0033, a causa del minor numero di occorrenze, posizionandosi al di sotto di `.ru` o `.io`, ad esempio, due dei TLD più comuni ma al contempo più utilizzati per il phishing.

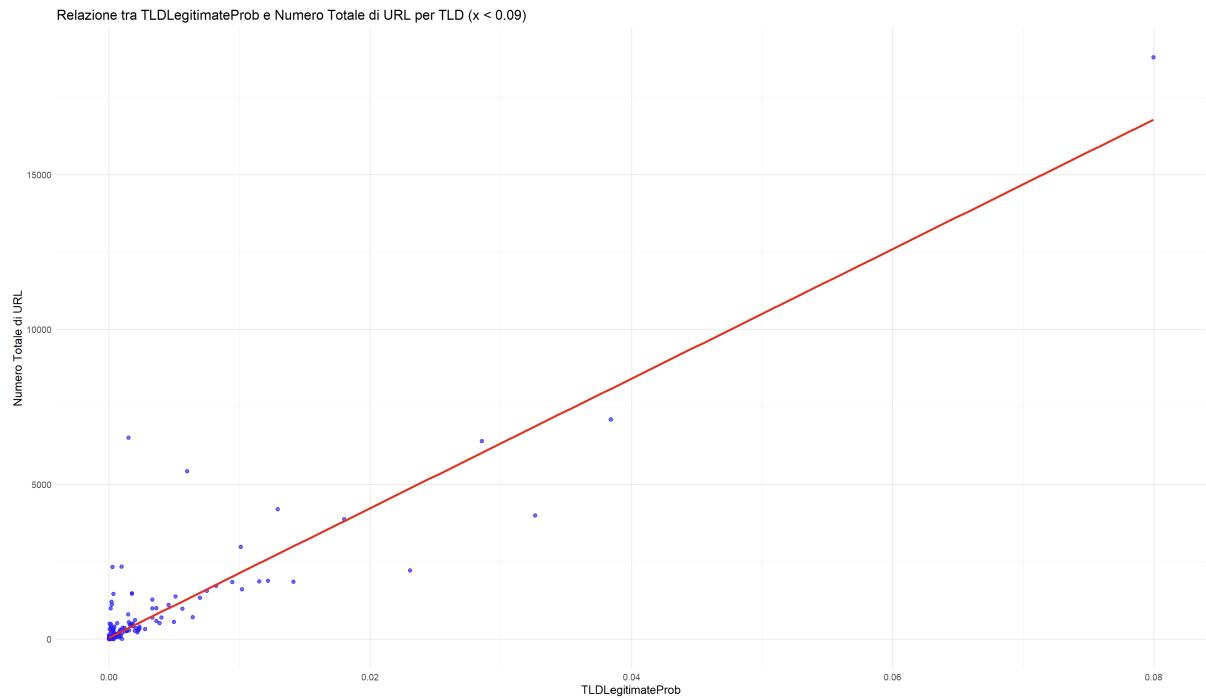


Figura 13: Scatterplot che mostra la correlazione tra TLDLegitimateProb e il numero totale di URL per ogni TLD. Per una migliore chiarezza, l'asse x è stata limitata a 0.09, in quanto tutti i valori sono compresi tra 0 e 0.09, tranne *.com*, che è l'unico TLD con una TLDLegitimateProb di 0.52.

Questo è dovuto al modo in cui questa variabile viene ottenuta, in quanto il suo valore dipende solo ed esclusivamente dalla frequenza di utilizzo di ogni TLD e non da un loro maggiore impiego per attività illecite.

Nel dataset non sono presenti altre colonne riferite ai TLD, tuttavia è lecito pensare che, come detto anche all'inizio del capitolo, che il basso costo e l'assenza di controlli siano il motivo principale che porta utenti malintenzionati ad usare questi domini per attività di phishing. Inoltre, molti TLD utilizzati per attività illecite sono domini di primo livello nazionali di paesi più poveri o con regolamentazioni deboli, come la Russia o gli stati dell'Africa o del Centro America. I motivi sono da ricercarsi sempre nella facilità di accesso, negli scarsi controlli e nella libertà di utilizzo dei nomi in domini meno affollati.

Escludendo *.com*, il TLD più usato per il phishing risulta essere *.app*. Questo è un TLD generico gestito da Google, pensato per applicazioni o, in generale, prodotti software.

Oltre ai TLD generici, nella vetta della classifica sono presenti TLD nazionali utilizzati per il phishing, che risultano essere `.co` (Colombia), `.io` (Territorio Britannico dell'Oceano Indiano) e `.ru` (Russia), che occupano rispettivamente il secondo, il terzo e il quarto posto.

Il numero di TLD con percentuale di phishing superiore al 50% è 237, ovvero il 41,65% sul totale. Di seguito mostriamo la distribuzione per fasce di URL di phishing:

Table 5: Numero e percentuale di TLD con 0 URL per categoria

Descrizione	Numero	Percentuale (%)
TLD con 0 URL di phishing	139	24,43
TLD con 0 URL legittimi	96	16,87

Table 6: Distribuzione dei TLD in base alla percentuale di URL di phishing

Fascia	Numero di TLD	Percentuale (%)
0-10%	165	29.00
10-20%	42	7.38
20-30%	47	8.26
30-40%	29	5.10
40-50%	15	2.64
50-60%	50	8.79
60-70%	24	4.22
70-80%	26	4.57
80-90%	31	5.45
90-100%	140	24.60

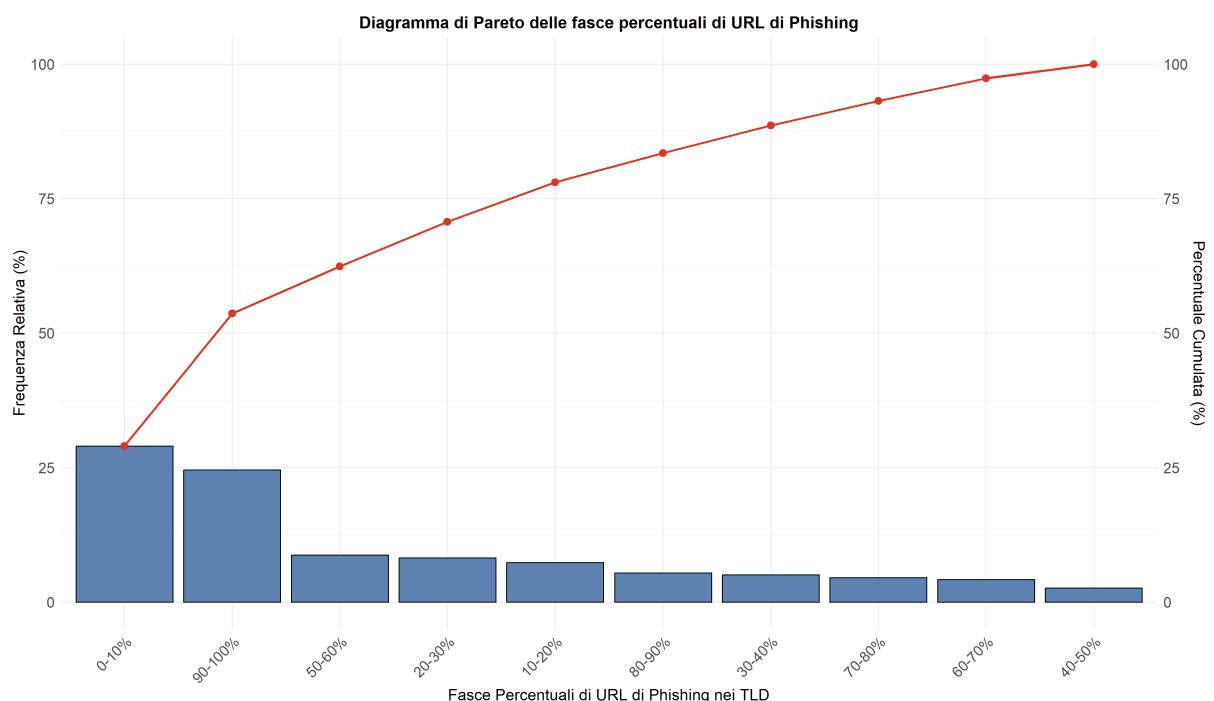


Figura 14: Diagramma di Pareto delle fasce percentuali di phishing nei TLD

Come si può notare dalla tabella 6 e dal grafico in Figura 14 la maggior parte dei TLD ricade nelle fasce estreme, 0-10% e 90-100%. Queste riflettono una combinazione di fattori: le dinamiche di utilizzo dei TLD, la regolamentazione dei registrar, le strategie degli attori malevoli e la distribuzione naturale delle registrazioni legittime rispetto a quelle fraudolente. Analizzare queste tendenze può aiutare a comprendere meglio il comportamento dei phisher e migliorare le misure di prevenzione. I cybercriminali tendono a preferire TLD meno noti, di recente introduzione o meno utilizzati dai siti legittimi, poiché hanno meno probabilità di essere bloccati da blacklist o filtri di sicurezza.

Questo porta ad una polarizzazione dove certi TLD diventano “hub” per attività di phishing, mentre altri rimangono quasi interamente sicuri.

Dal punto di vista geografico, il dataset può darci un’idea di quali TLD vengano maggiormente utilizzati per il phishing. La mappa in figura 15 mostra la percentuale di URL di phishing per ciascun TLD nazionale.

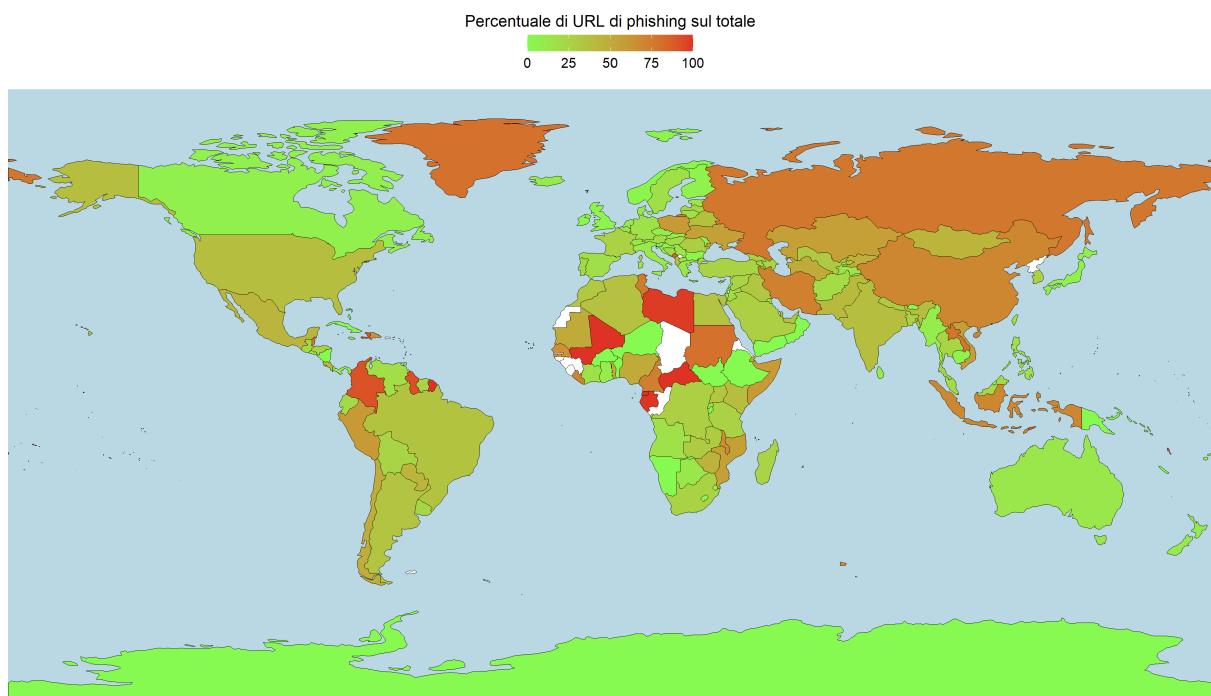


Figura 15: Mappa della percentuale di URL di phishing nei TLD nazionali. I paesi senza dati sono in bianco

Tuttavia, è importante notare che i TLD con un basso numero di URL totali potrebbero non avere un’adeguata rappresentanza statistica nel dataset. Questo può portare a valori di phishing che non sono rappresentativi della realtà, con alcune aree che appaiono significativamente più vulnerabili o più sicure a causa della scarsità di dati.

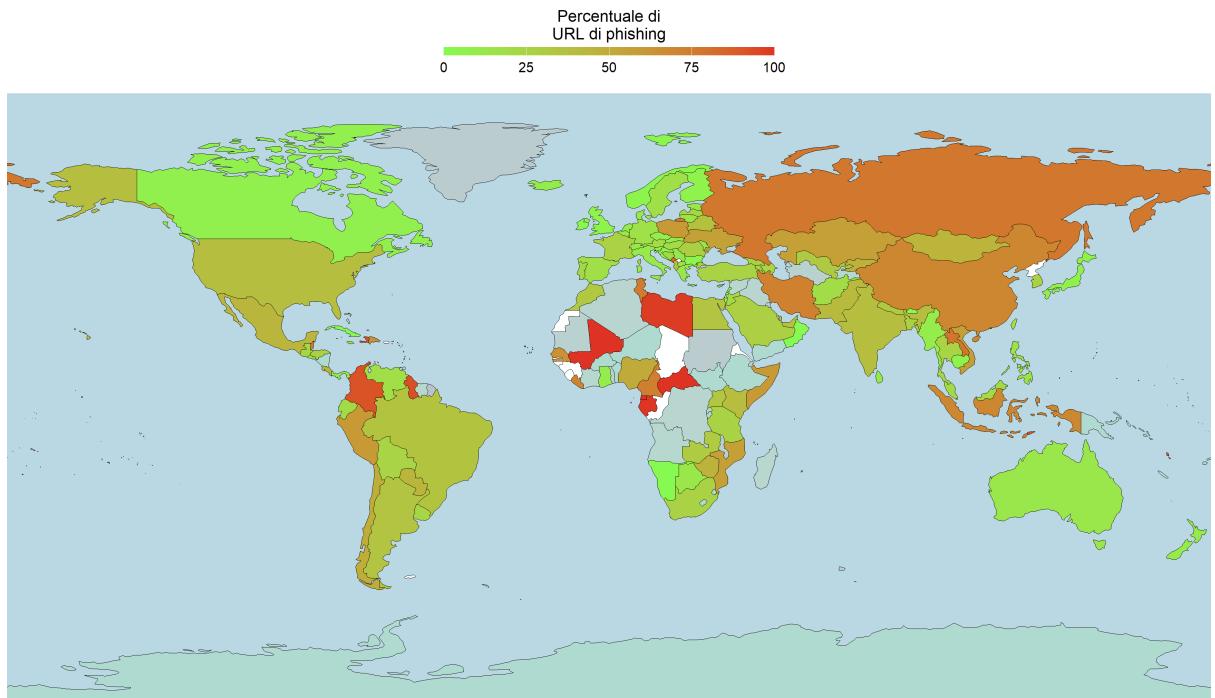


Figura 16: Mappa della percentuale di URL di phishing nei TLD nazionali. I paesi con TLD che hanno meno di 10 URL nel dataset sono grigie

## 2.3 Offuscamento dell'URL

L'offuscamento degli URL è una tecnica utilizzata, spesso in contesti di phishing e cybercriminalità, per nascondere la reale destinazione di un link o per renderlo più difficile da riconoscere come dannoso. L'obiettivo è ingannare gli utenti o superare i sistemi di sicurezza automatizzati. Tuttavia, l'offuscamento può anche essere usato per motivi legittimi, come la protezione della privacy o il mascheramento di parametri tecnici.

Nel dataset sono presenti tre colonne riferite all'offuscamento: HasObfuscation, NoOfObfuscatedChar, ObfuscationRatio. Sono presenti 485 URL offuscati, tutti di phishing: non sono presenti URL legittimi che presentano offuscamento.

Tra gli URL che presentano offuscamento possiamo osservare queste statistiche per quanto riguarda il numero di caratteri offuscati:

Media	Mediana	Moda	Varianza	Deviazione Standard	Minimo	Massimo
12,09	6	3	1568,93	39,61	3	447

Table 7: Statistiche sul numero di caratteri offuscati negli URL che presentano offuscamento

<b>Media</b>	<b>Mediana</b>	<b>Moda</b>	<b>Varianza</b>	<b>Deviazione Standard</b>	<b>Minimo</b>	<b>Massimo</b>
0,0673	0,056	0,056	0,0026	0,0507	0,006	0,348

Table 8: Statistiche sul rapporto di offuscamento negli URL che presentano offuscamento

Gli URL offuscati risultano essere mediamente molto più lunghi rispetto agli altri URL:

<b>Categoria</b>	<b>Media</b>	<b>Moda</b>	<b>Mediana</b>	<b>Minimo</b>	<b>Massimo</b>
URLLength offuscati	181.98	215	109	31	4274
URLLength phishing	45.72	26	34	13	6097
URLLength legittimi	26.23	25	26	15	57

Table 9: Statistiche descrittive della lunghezza degli URL, suddivise per categoria.

Più nello specifico, la distribuzione della lunghezza è così definita:

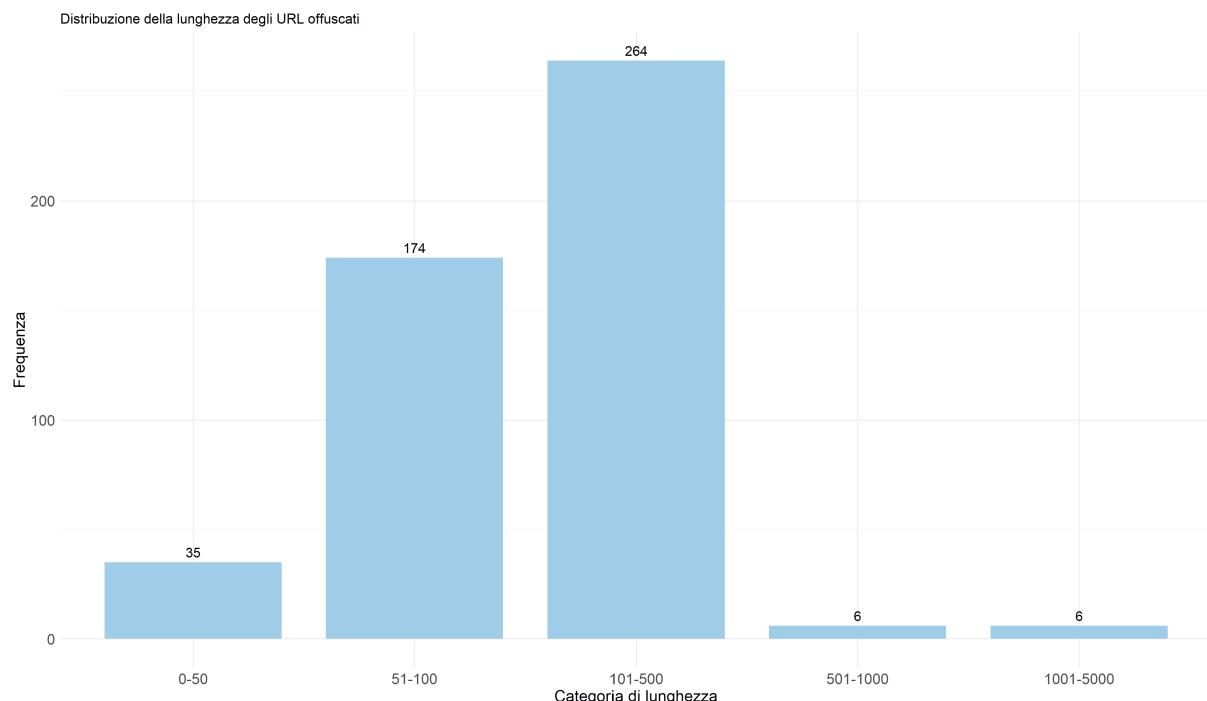


Figura 17: Barplot che mostra la distribuzione degli URL offuscati in base alle fasce di lunghezza

Proviamo a studiare la correlazione tra lunghezza dell'URL e offuscamento, per capire se gli URL più lunghi vogliono nascondere parametri o altre componenti, oppure sono altre le motivazioni che portano ad allungare gli URL.

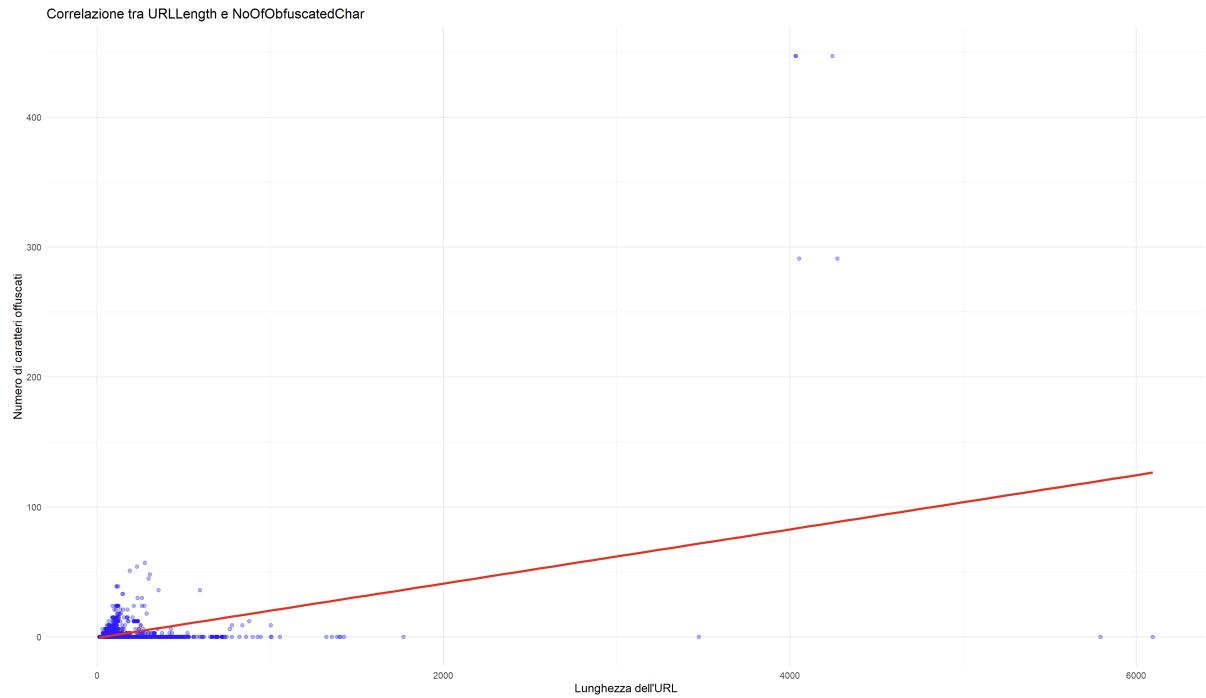


Figura 18: Scatterplot che mostra la correlazione tra la lunghezza dell'URL e i caratteri offuscati.

Il coefficiente di correlazione risulta di 0,46, indicando una correlazione positiva ma non fortissima. Tuttavia, questo dato è influenzato particolarmente dagli outlier. Inoltre, nel grafico, si nota una forte densità in basso a sinistra. Osservando meglio quei valori, possiamo osservare un andamento irregolare:

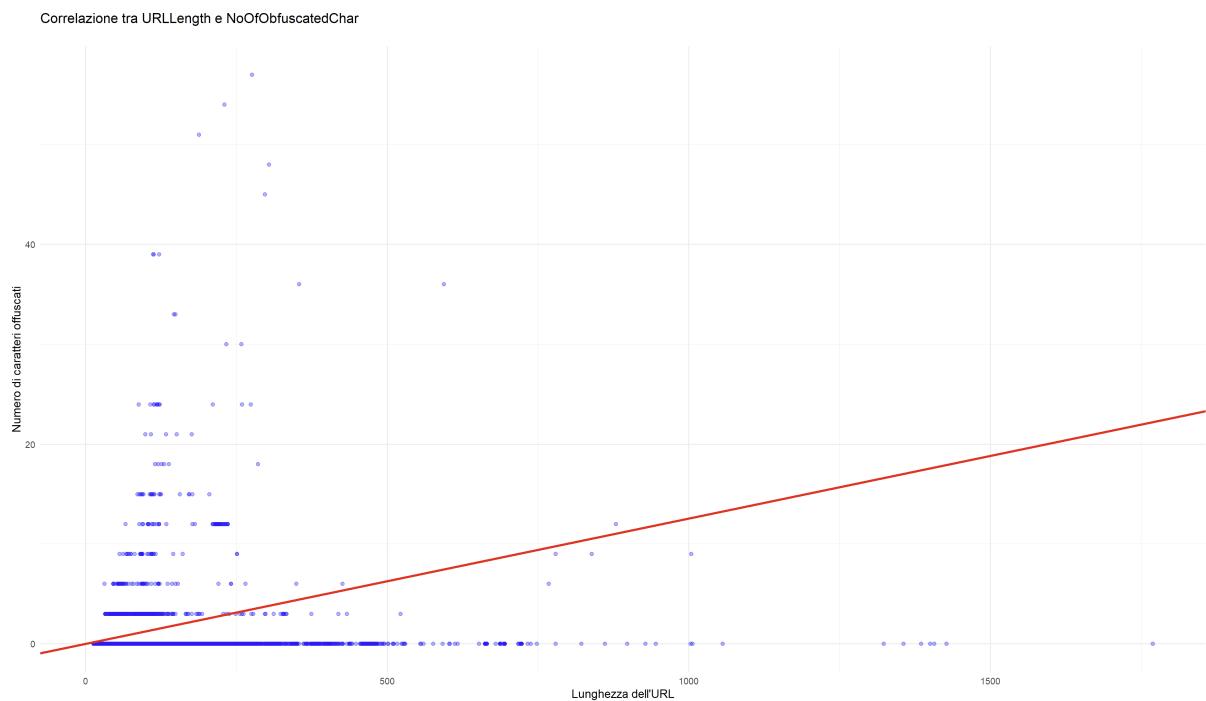


Figura 19: Scatterplot che mostra la relazione tra la lunghezza dell'URL e i caratteri offuscati per URL inferiori a 2000 caratteri.

Se escludessimo gli URL con lunghezza superiore a 2000 caratteri, il coefficiente di

correlazione sarebbe di 0,14.

Sebbene l'offuscamento possa portare ad un aumento del numero di caratteri all'interno dell'URL, studiando la correlazione tra queste due variabili non otteniamo un andamento ben definito: il grafico infatti non mostra una correlazione lineare molto evidente, ma solo una debolissima correlazione positiva.

Utilizzando il comando `cor.test()`, possiamo ottenere diversi dati sulla correlazione ottenuta:

- **t**: il valore del test t di Student.
- **p-value**: probabilità associata al test.
- **confidence interval**: l'intervallo di confidenza per la correlazione.
- **cor**: il valore del coefficiente di correlazione.

```
> cor.test(filtered_data_limited$URLLength, filtered_data_limited$NoOfObfuscatedChar, method = "pearson")
Pearson's product-moment correlation

data: filtered_data_limited$URLLength and filtered_data_limited$NoOfObfuscatedChar
t = 6.1962, df = 478, p-value = 1.249e-09
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
0.1877522 0.3535433
sample estimates:
cor
0.2726707
```

Figura 20: Risultati di `cor.test(df$URLLength, df$NoOfObfuscatedChar, method = "pearson")`

L'output ottenuto è così descrivibile:

- **p-value**: questo valore indica la probabilità, per una ipotesi supposta vera, di ottenere risultati ugualmente o meno compatibili di quelli osservati durante il test, con la suddetta ipotesi. Poiché il p-value è pari a 1,249e-09, (prossimo allo 0 e inferiore al livello di significatività convenzionale di 0,05), si può rifiutare l'ipotesi nulla e dunque esiste una correlazione statisticamente significativa tra le due variabili.
- **t e df**: rispettivamente valore del test t di Student e gradi di libertà, che confermano la significatività della differenza dei dati rispetto all'ipotesi nulla.
- **Intervallo di confidenza**: questo valore indica la probabilità che il valore della correlazione sia compreso nell'intervallo specificato. Dunque abbiamo una certa fiducia (95%) che il coefficiente di correlazione sia compreso tra 0,1878 e 0,3535, confermando la correlazione significativa ma debole.

Questi risultati indicano che, sebbene esista una correlazione tra la lunghezza dell'URL e l'offuscamento dei caratteri, essa risulta debole. Pertanto, anche se l'offuscamento potrebbe avere un certo impatto sulla lunghezza, ed è associato a URL più lunghi, è probabile che altre variabili abbiano un'influenza più significativa su questa caratteristica.

## 2.4 Studio dei caratteri degli URL

Il dataset fornito contiene delle colonne che riguardano il tipo di caratteri presenti. In particolare abbiamo:

- NoOfLettersInURL
- LetterRatioInURL
- NoOfDigitsInURL\*
- DigitRatioInURL\*
- NoOfEqualsInURL
- NoOfQMarkInURL
- NoOfAmpersandInURL
- NoOfOtherSpecialCharsInURL
- SpacialCharRatioInURL\*

\*'Digits' e 'Spacial' sono dei tipo per 'Digits' e 'Special'

<b>Colonna</b>	<b>Media</b>	<b>Median</b> a	<b>Moda</b>	<b>DS</b>	<b>Min</b>	<b>Max</b>
NoOfLettersInURL	12.93	13	12	4.78	1	44
LetterRatioInURL	0.48	0.480	0.500	0.10	0.056	0.772
NoOfDigitsInURL	0.05	0	0	0.35	0	8
DigitRatioInURL	0.002	0	0	0.01	0	0.348
NoOfEqualsInURL	0.000	0	0	0.000	0	0
NoOfQMarkInURL	0.000	0	0	0	0	0
NoOfAmpersandInURL	0.000	0	0	0.000	0	0
NoOfOtherSpecialCharsInURL	1.24	1	1	0.50	0	7
SpacialCharRatioInURL	0.048	0.042	0.040	0.02	0.000	0.231

Table 10: Statistiche per gli URL legittimi.

<b>Colonna</b>	<b>Media</b>	<b>Median</b> a	<b>Moda</b>	<b>DS</b>	<b>Min</b>	<b>Max</b>
NoOfLettersInURL	28,11	20	16	42,60	0	5191
LetterRatioInURL	0,57	0,577	0,50	0,14	0,000	0,926
NoOfDigitsInURL	4,33	0	0	17,87	0	2011
DigitRatioInURL	0,06	0,000	0,00	0,10	0,000	0,684
NoOfEqualsInURL	0,15	0	0	1,42	0	176
NoOfQMarkInURL	0,07	0	0	0,29	0	4
NoOfAmpersandInURL	0,06	0	0	1,28	0	149
NoOfOtherSpecialCharsInURL	3,80	3	1	5,00	1	499
SpacialCharRatioInURL	0,08	0,080	0,045	0,04	0,005	0,397

---

Table 11: Statistiche per gli URL di phishing.

Di seguito alcuni grafici per visualizzare la distribuzione di suddette caratteristiche tra gli URL, confrontando per legittimità:

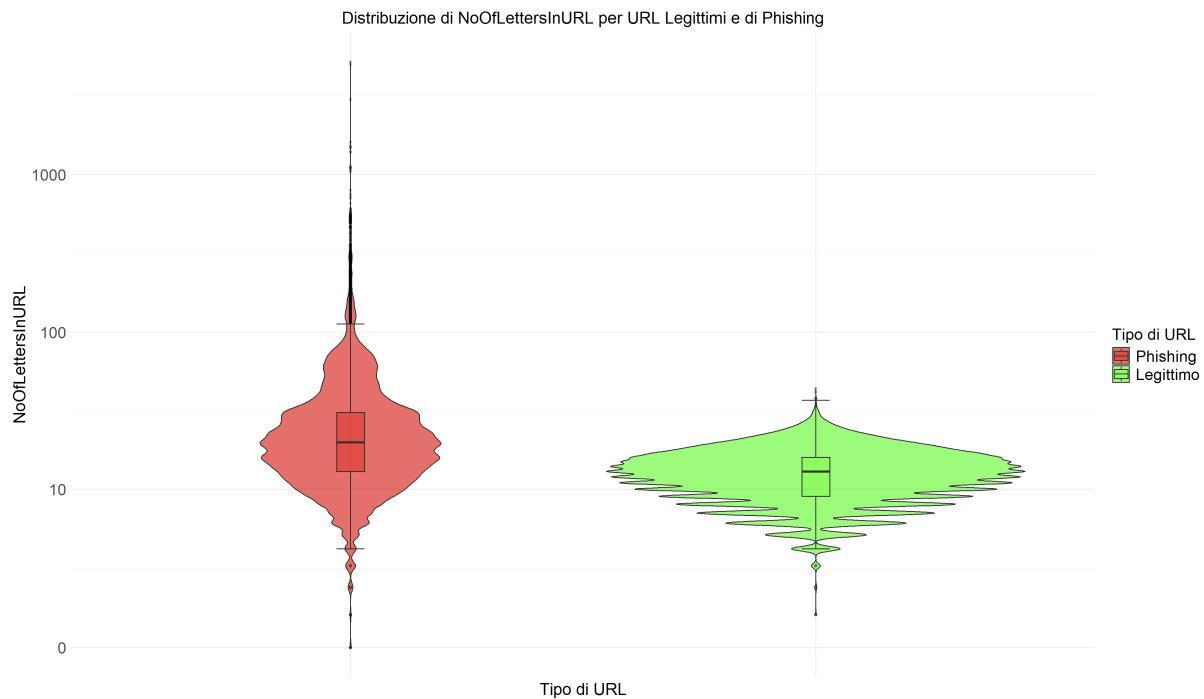


Figura 21: Confronto della distribuzione delle lettere negli URL legittimi e di phishing.

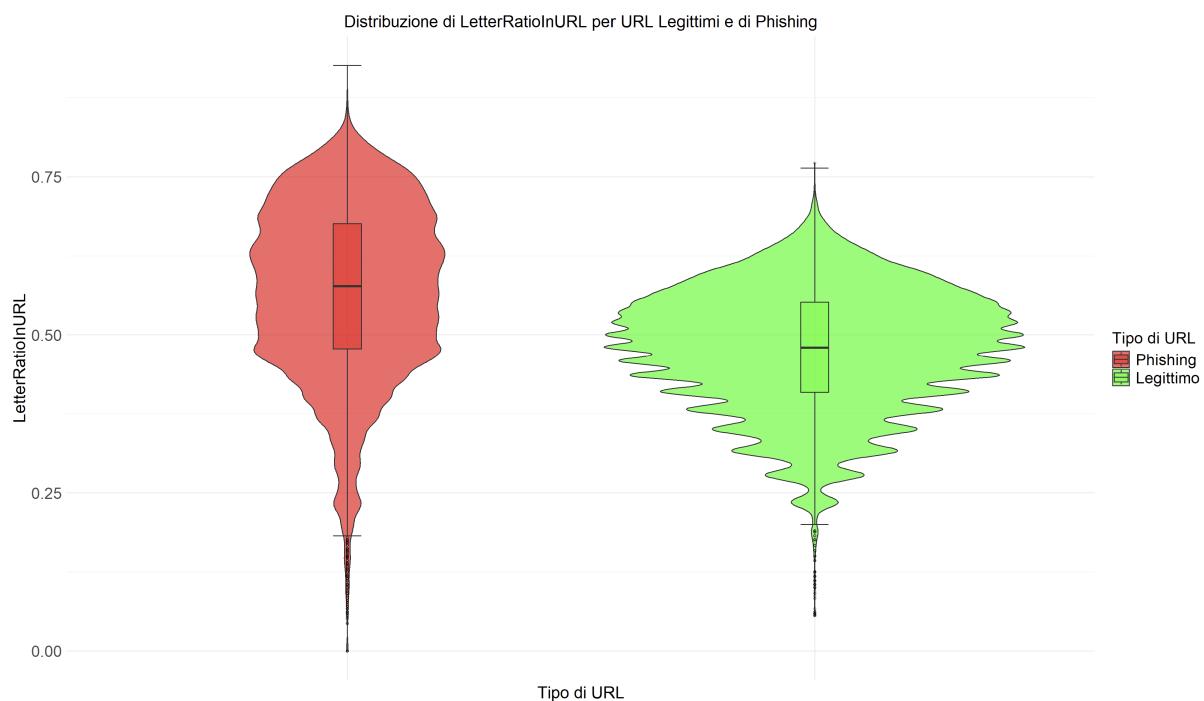


Figura 22: Confronto della distribuzione del rapporto di lettere negli URL legittimi e di phishing.

I grafici 21 e 22 si riferiscono al confronto sulla quantità di lettere negli URL. Il primo mostra la distribuzione delle lettere come numero assoluto, mentre il secondo mostra la distribuzione del rapporto tra il numero di lettere presenti nell'URL e la sua lunghezza.

Per il primo grafico è stata utilizzata una scala logaritmica (pseudologaritmica, in quanto sono presenti i valori in corrispondenza di 0), a causa della grande ampiezza del range di valori assunti, con una concentrazione però estremamente limitata. Come prevedibile, il grafico mostra una forte somiglianza con quello relativo alla lunghezza degli URL (Figura 1), poiché le lettere costituiscono la componente principale della loro lunghezza, come dimostra anche il grafico del rapporto di lettere. Gli URL di phishing, essendo mediamente più lunghi, contengono un numero maggiore di lettere rispetto a quelli legittimi, come si può notare comunque dalle tabelle 10 e 11. Inoltre, presentano una distribuzione più ampia e valori estremamente elevati nei casi più estremi. Come possiamo notare dagli outlier, questi valori estremi si distribuiscono dalla lunghezza di 200 fino al massimo di 5191 per gli URL di phishing.

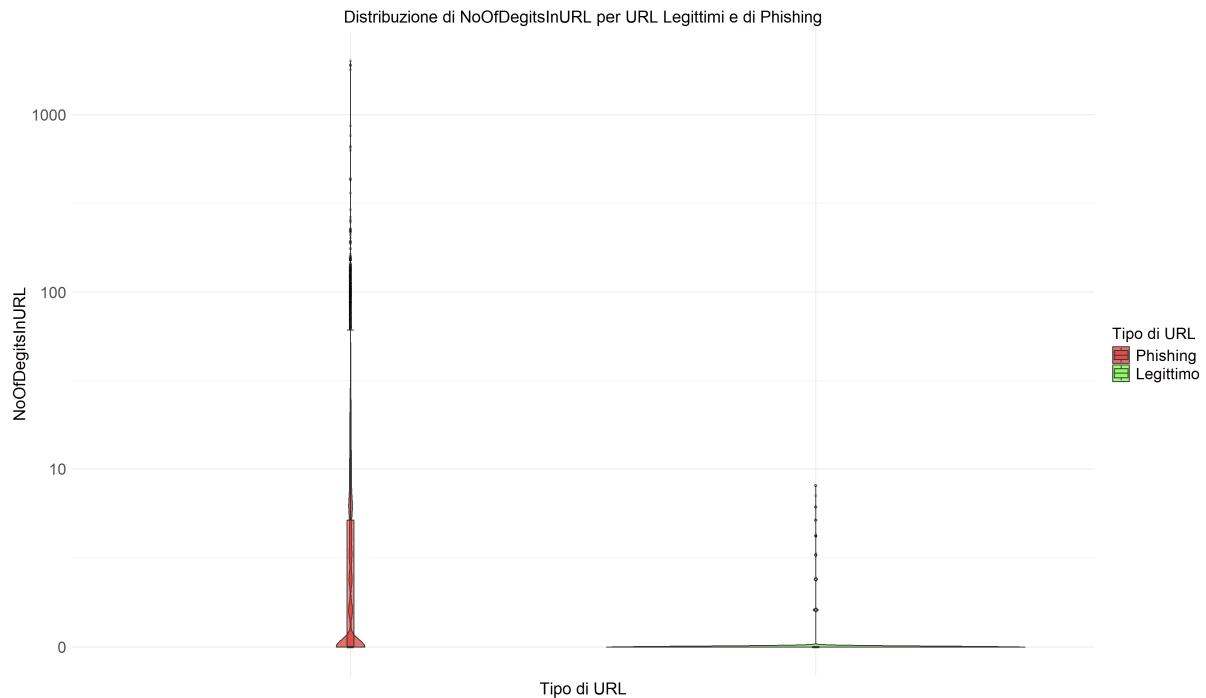


Figura 23: Confronto della distribuzione delle cifre negli URL legittimi e di phishing.

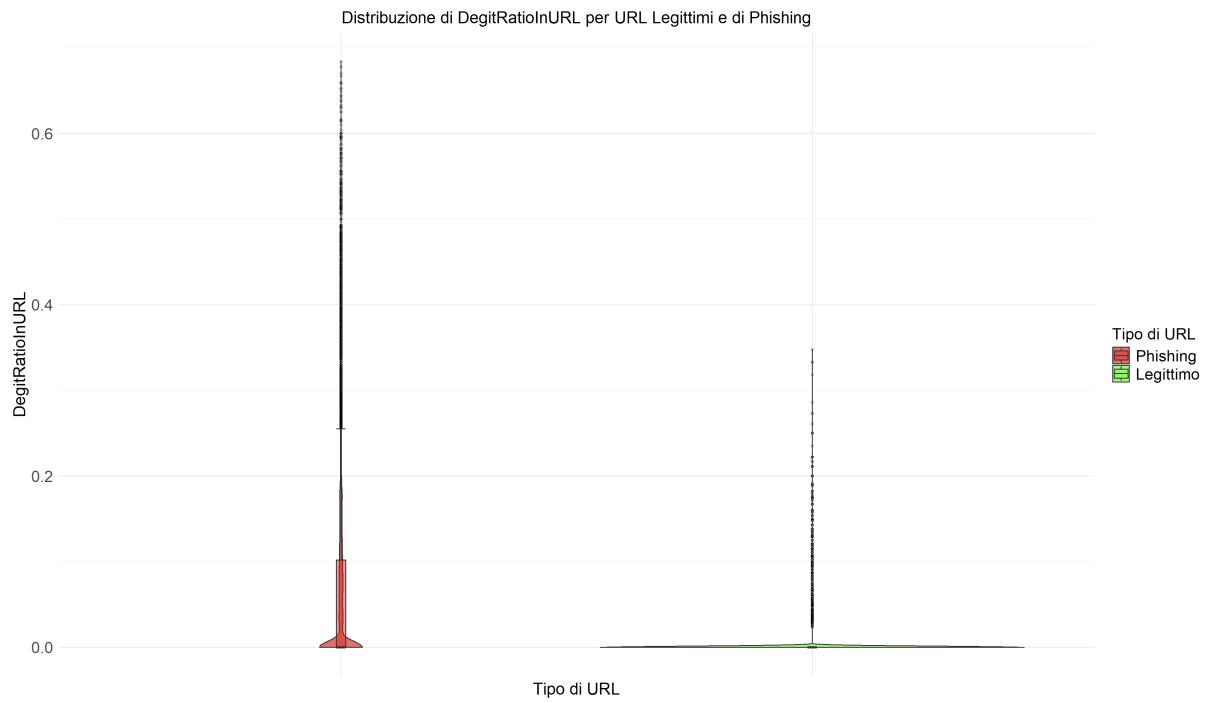


Figura 24: Confronto della distribuzione del rapporto di cifre negli URL legittimi e di phishing.

Discorso simile alle lettere vale per le cifre, i cui grafici 23 e 24 mostrano una distribuzione concentrata sul valore 0, ma con differenze sostanziali tra gli URL di phishing e quelli legittimi.

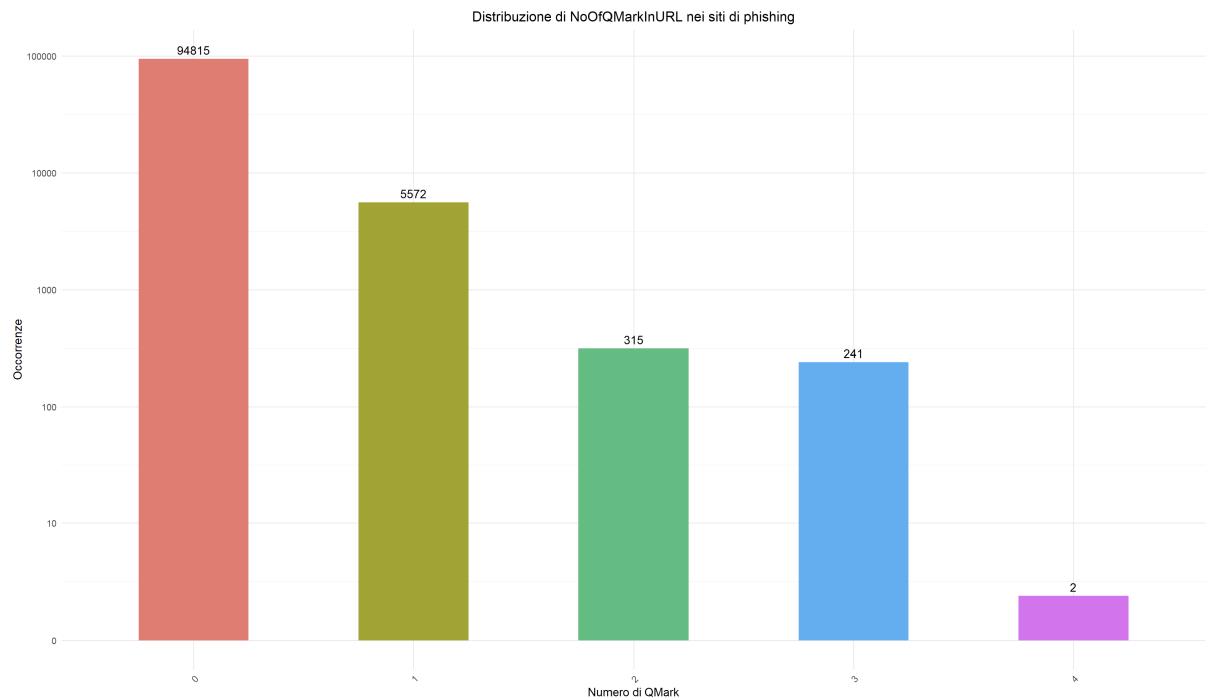


Figura 25: Distribuzione dei punti interrogativi negli URL di phishing.

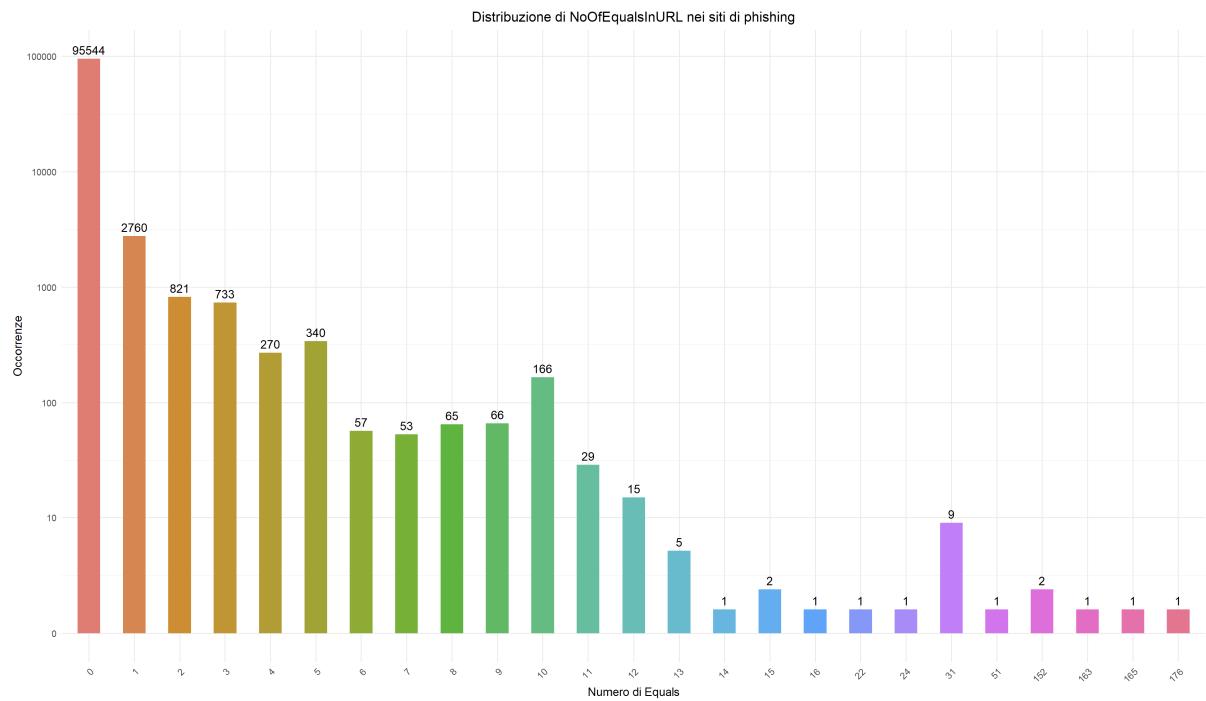


Figura 26: Distribuzione di '=' negli URL di phishing.

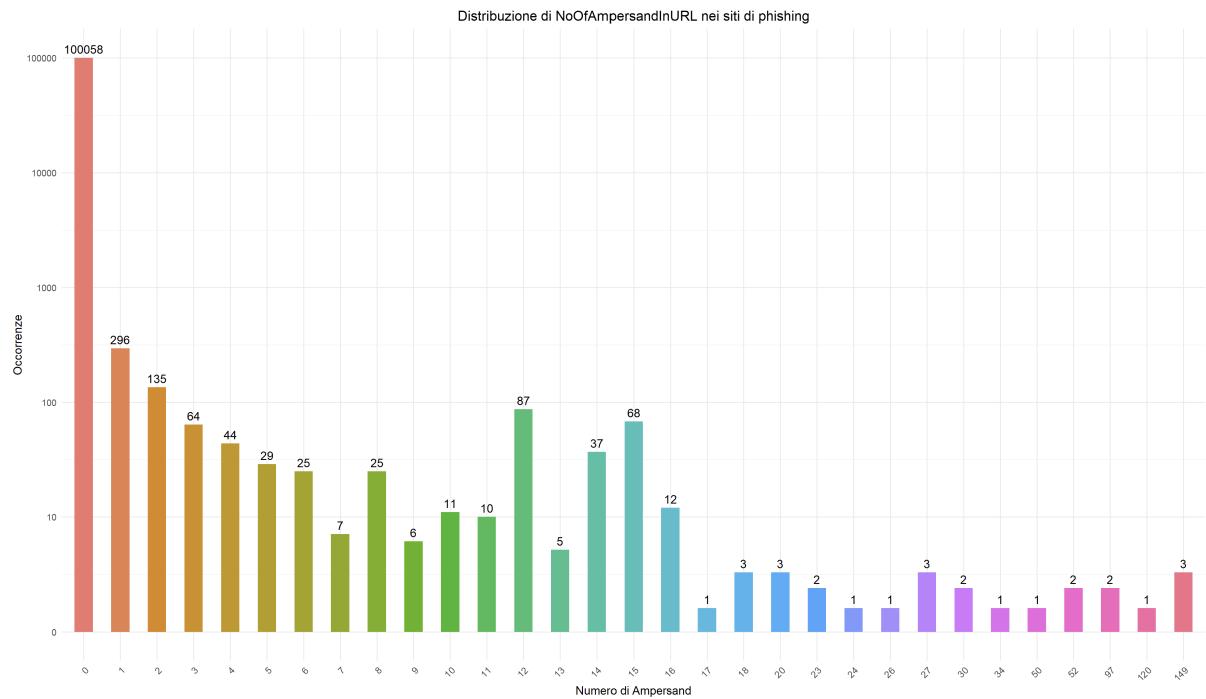


Figura 27: Distribuzione di '=' negli URL di phishing.

Per quanto riguarda i caratteri '?' (Figura 25), '=' (Figura 26) e '&' (Figura 27), questi hanno una caratteristica in comune: tutti e tre non sono presenti negli URL legittimi, ma solo in quelli fraudolenti.

La distribuzione dei punti interrogativi assume soltanto quattro valori, da 0 a 4, con il 93,93% del totale rappresentato da 0.

I caratteri '=' e '&' presentano invece una distribuzione molto simile tra loro, quasi

completamente concentrata sul valore 0 (rispettivamente 94,65% e 99,12%), ma comunque più ampia.

<b>Carattere</b>	<b>Numero di URL</b>	<b>Percentuale sugli URL di phishing (%)</b>
?	6130	6.07
=	5401	5.35
&	887	0.88

Table 12: Numero di URL e percentuale che hanno almeno un'occorrenza di ogni carattere

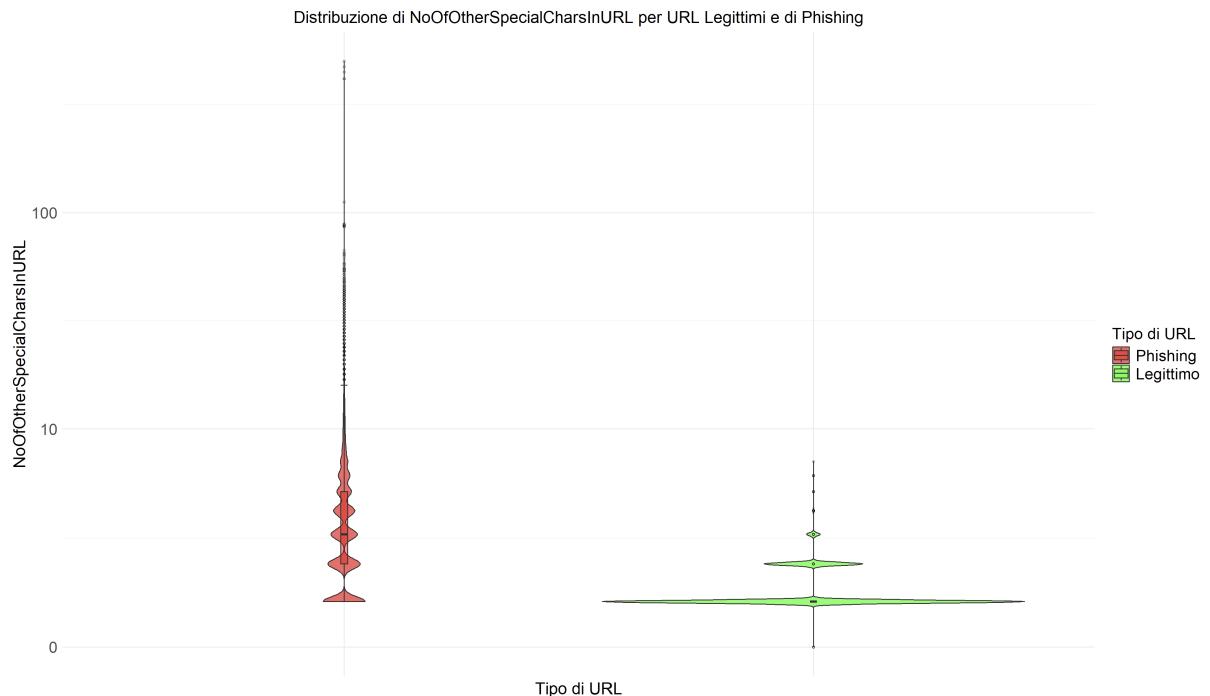


Figura 28: Confronto della distribuzione di altri caratteri speciali negli URL legittimi e di phishing.

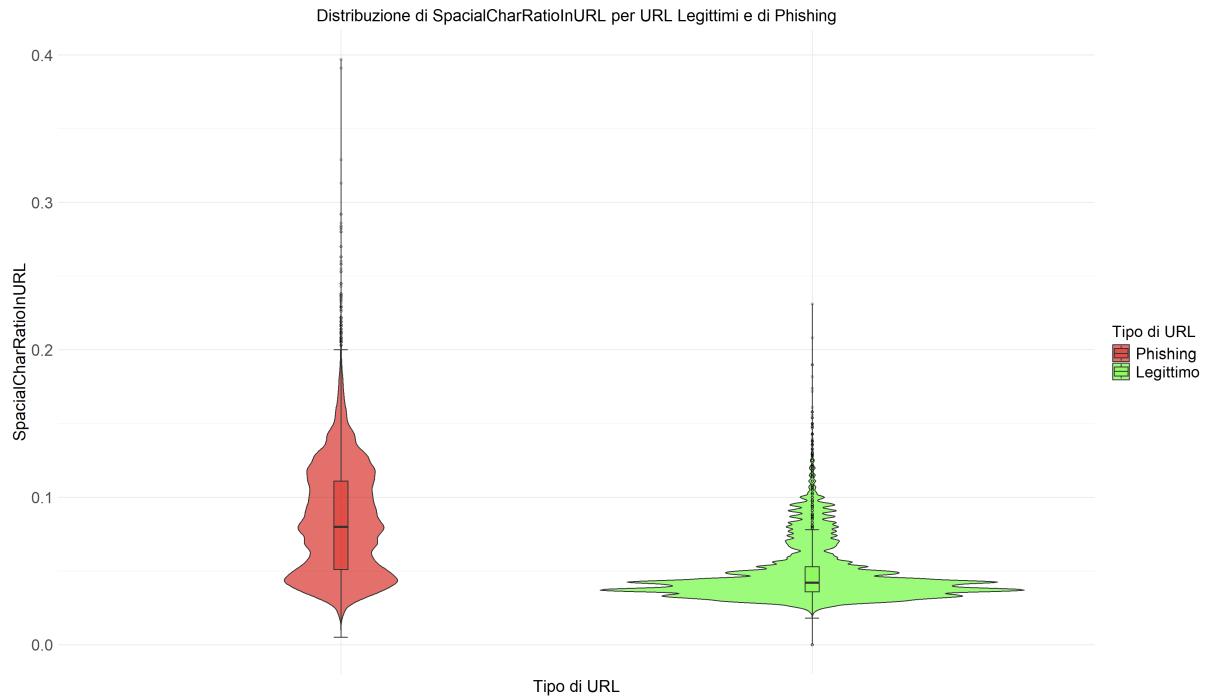


Figura 29: Confronto della distribuzione del rapporto di tutti i caratteri speciali negli URL di phishing.

Per la colonna `NoOfOtherSpecialCharsInURL`, vi sono alcune precisazioni da fare. Non è ben chiaro quali siano i caratteri presi in considerazione in questa variabile, ma è lecito pensare che possa includere tutti quelli che non siano una lettera, un numero o uno dei caratteri speciali sopra riportati. Tuttavia, la distribuzione di questi altri caratteri nel dataset lascia pensare che sia stato conteggiato anche uno dei due punti che separano il sottodominio o il dominio di primo livello dal dominio, ma non entrambi. Vediamo le distribuzioni:

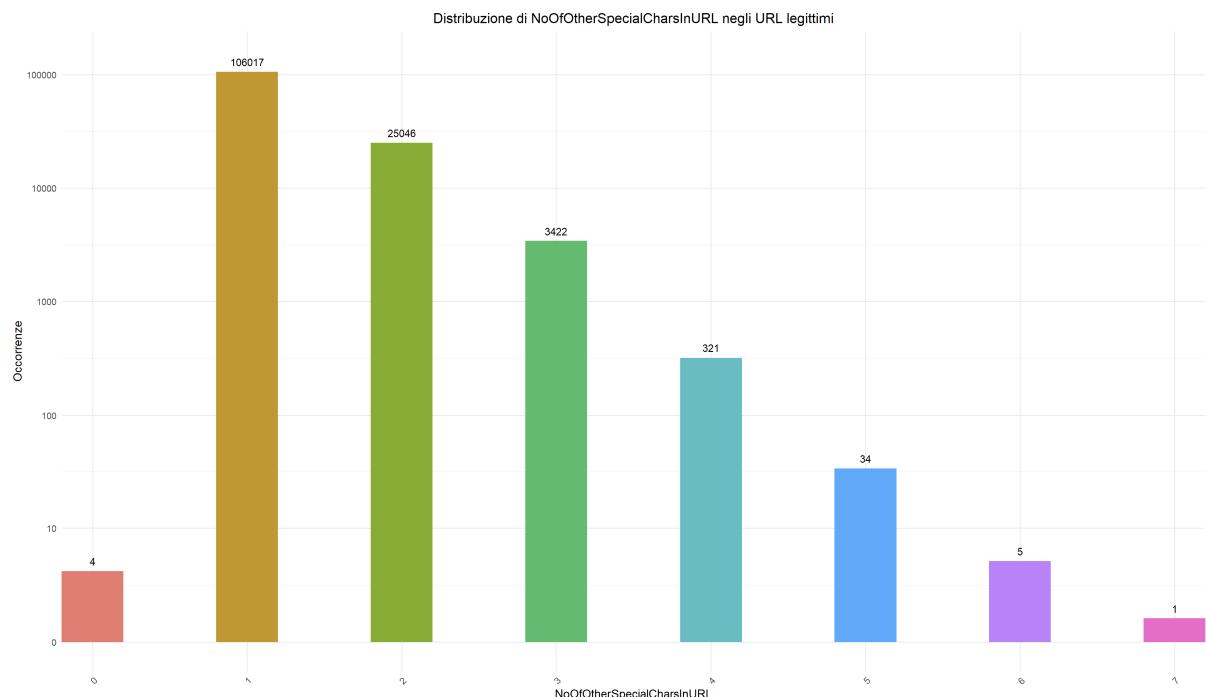


Figura 30: Distribuzione di `NoOfOtherSpecialCharsInURL` negli URL legittimi

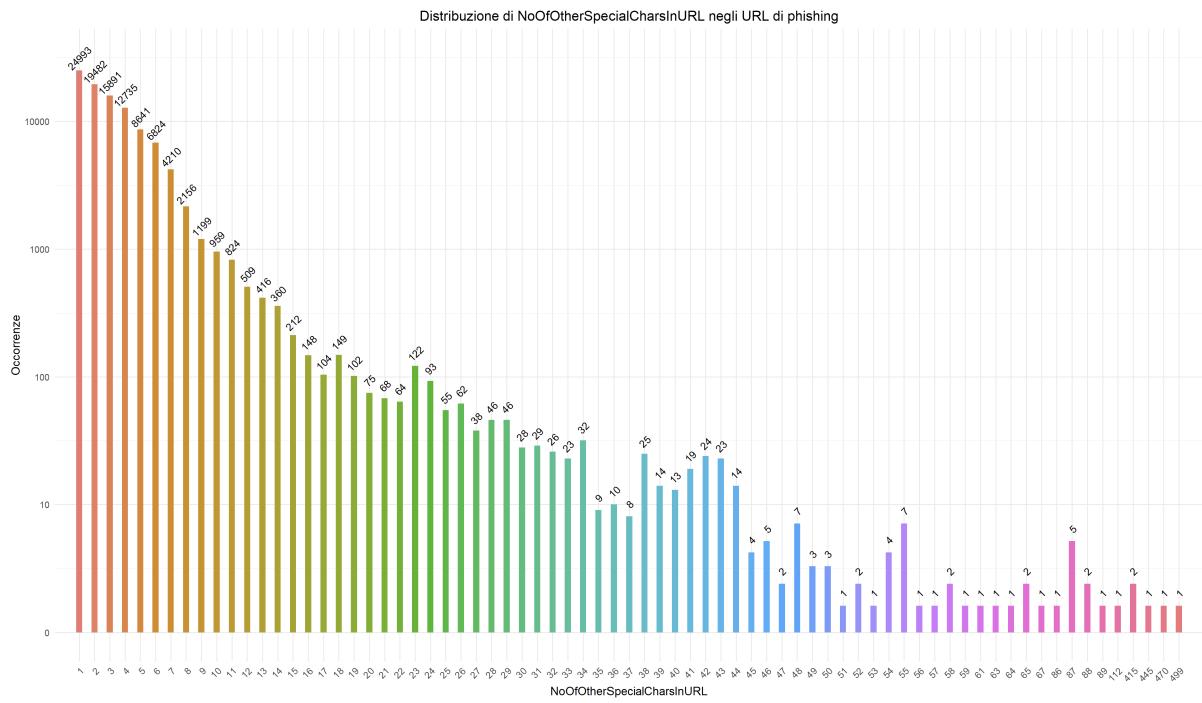


Figura 31: Distribuzione di NoOfOtherSpecialCharsInURL negli URL di phishing

Come si vede nelle Figure 30 e 31, buona parte degli URL risultano avere un carattere speciale diverso da quelli specificati, tuttavia, controllando queste righe, notiamo che non vi sono caratteri speciali a parte i due punti che delimitano il dominio. Non è chiaro quale dei due venga conteggiato. Ad esempio, URL come "www.southbankmosaics.com" risulta avere un carattere speciale, mentre "www.uni-mainz.de" risulta averne due, uno dei quali è probabilmente il trattino. Per utilizzare questa variabile in maniera ottimale, bisogna considerare i valori superiori a 1 (che dunque risulta essere la normalità) in quanto i punti che delimitano il dominio sono presenti nella quasi totalità degli URL. Per qualche motivo, quattro URL legittimi risultano avere 0 caratteri speciali diversi, nonostante siano simili a quelli che ne possiedono uno solo.

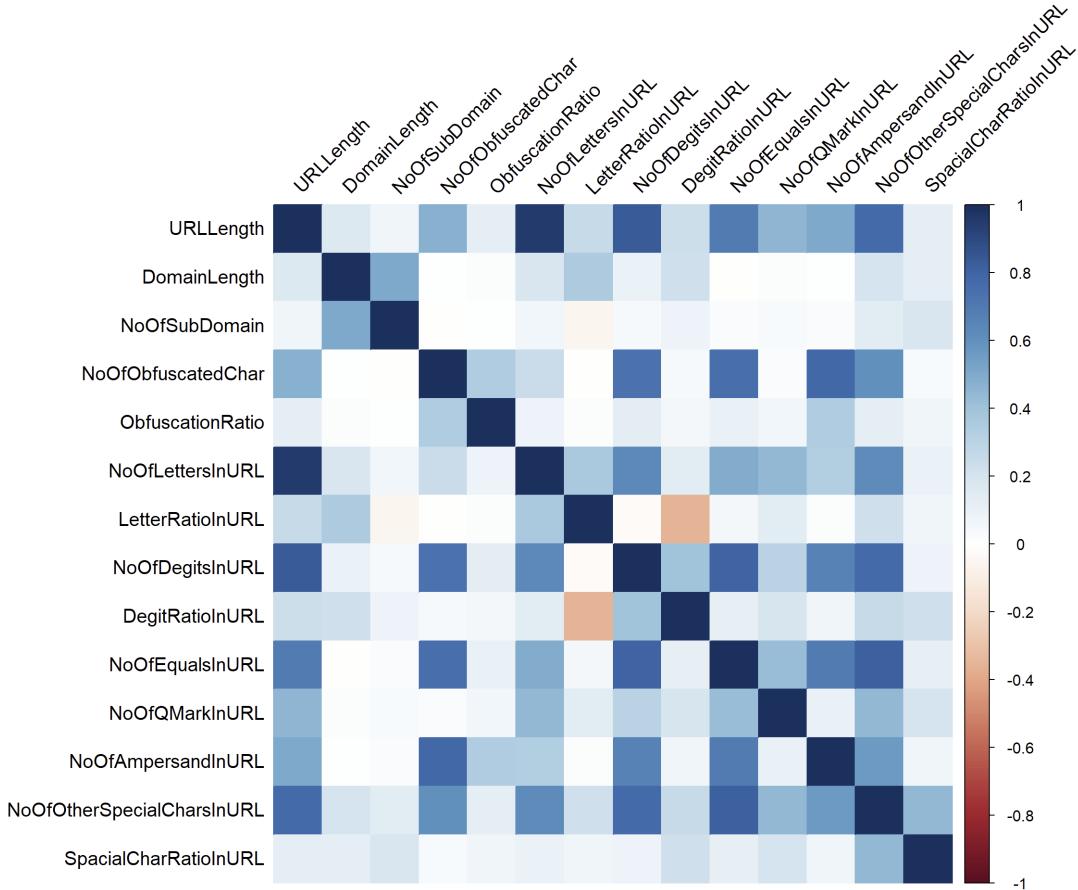


Figura 32: Matrice di correlazione tra le variabili riguardanti i caratteri negli URL di phishing.

Dalla matrice di correlazione in Figura 32 notiamo correlazioni che non sorprendono, come quelle tra i vari caratteri speciali e la lunghezza degli URL o i numeri. I caratteri speciali, infatti, in particolare '=' , indicano parametri passati all'URL, che ne aumentano in maniera sostanziale la lunghezza. Inoltre, questi parametri sono spesso numerici o presentano comunque numeri al loro interno, facendone aumentare la presenza.

È invece interessante notare la correlazione tra `NoOfObfuscatedChar` e le variabili `NoOfDigitsInURL`, `NoOfEqualsInURL`, `NoOfAmpersandInURL`. Innanzitutto, possiamo osservare quanto segue: Come si può notare dalla Tabella 13, gli URL che presentano

<b>Caratteristica <math>\geq 1</math></b>	<b>Numero di URL</b>	<b>HasObfuscation = 1</b>	<b>Entrambe</b>
<code>NoOfAmpersandInURL</code>	887	485	485
<code>NoOfEqualsInURL</code>	5401		219
<code>NoOfDigitsInURL</code>	51461		485

Table 13: Numero di URL con caratteristiche specifiche che e offuscamento

offuscamento sono 485 e tutti contengono almeno un '&' ed almeno un '=', mentre 219 URL su 485 offuscati presentano almeno un numero.

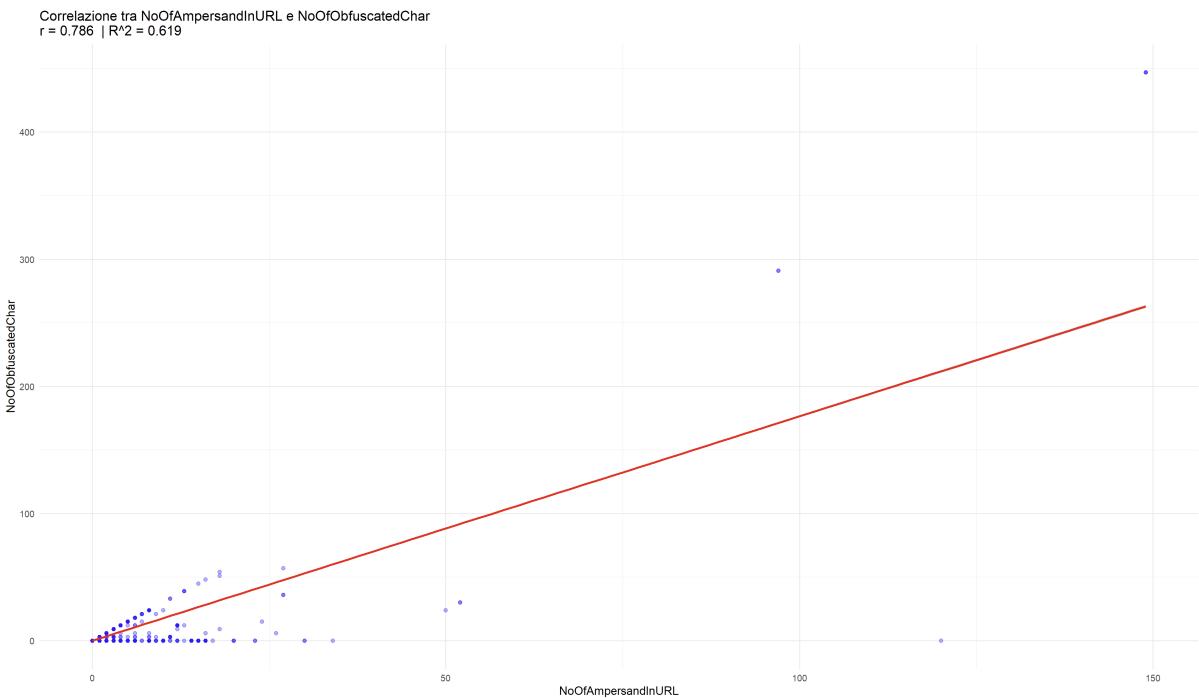


Figura 33: Scatterplot che mostra la correlazione tra NoOfAmpersandInURL e NoOfObfuscatedChar negli URL di phishing.

La correlazione tra NoOfAmpersandInURL e NoOfObfuscatedChar sembra forte, tuttavia è molto probabilmente influenzata dagli outlier, considerando anche il coefficiente di determinazione di 0,619. Tutti i valori sono concentrati al di sotto di 60, con soltanto 6 valori al di sopra, rispettivamente due a 97, uno a 120 e tre a 149. Provando a rimuovere questi valori e ricalcolare la correlazione, otteniamo un coefficiente di correlazione pari a 0,55 e un coefficiente di determinazione di 0,302, dunque una correlazione presente ma non fortissima. Potremmo tuttavia ipotizzare che vi sia un qualche tipo di correlazione non lineare.

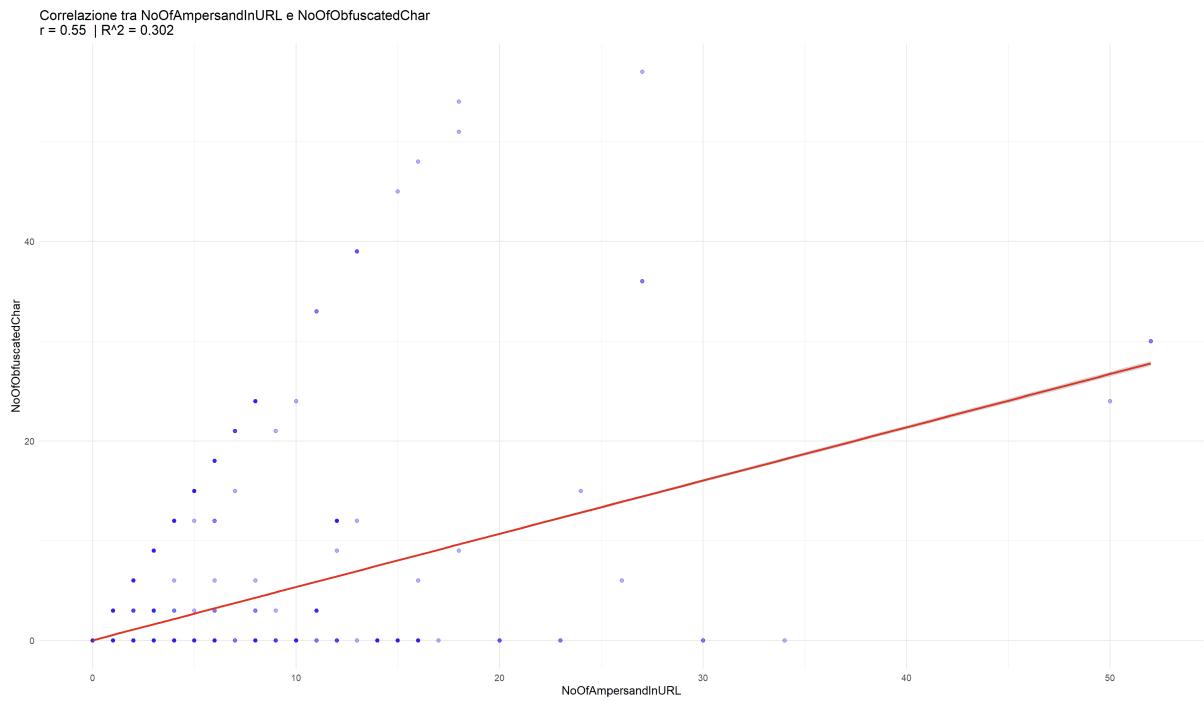


Figura 34: Scatterplot che mostra la correlazione tra NoOfAmpersandInURL e NoOfObfuscatedChar negli URL di phishing senza outlier.

Lo stesso accade per le altre due variabili:

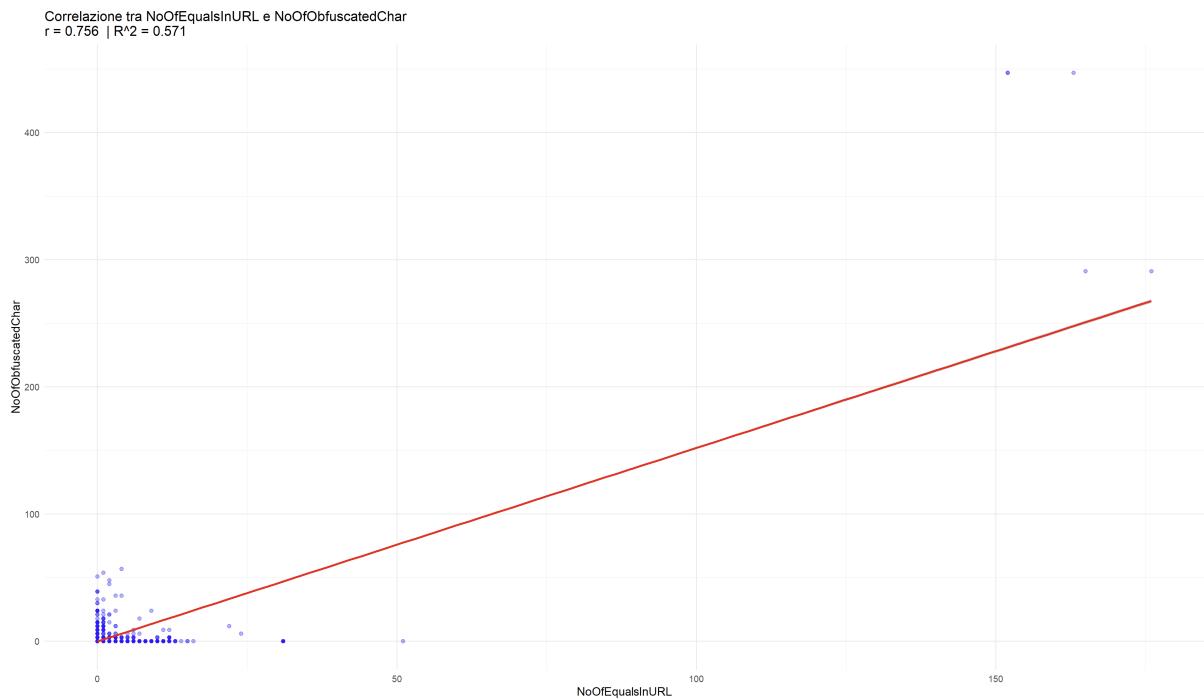


Figura 35: correlazione tra NoOfEqualsInURL e NoOfObfuscatedChar negli URL di phishing

Figura 36: Confronto correlazione con e senza outlier

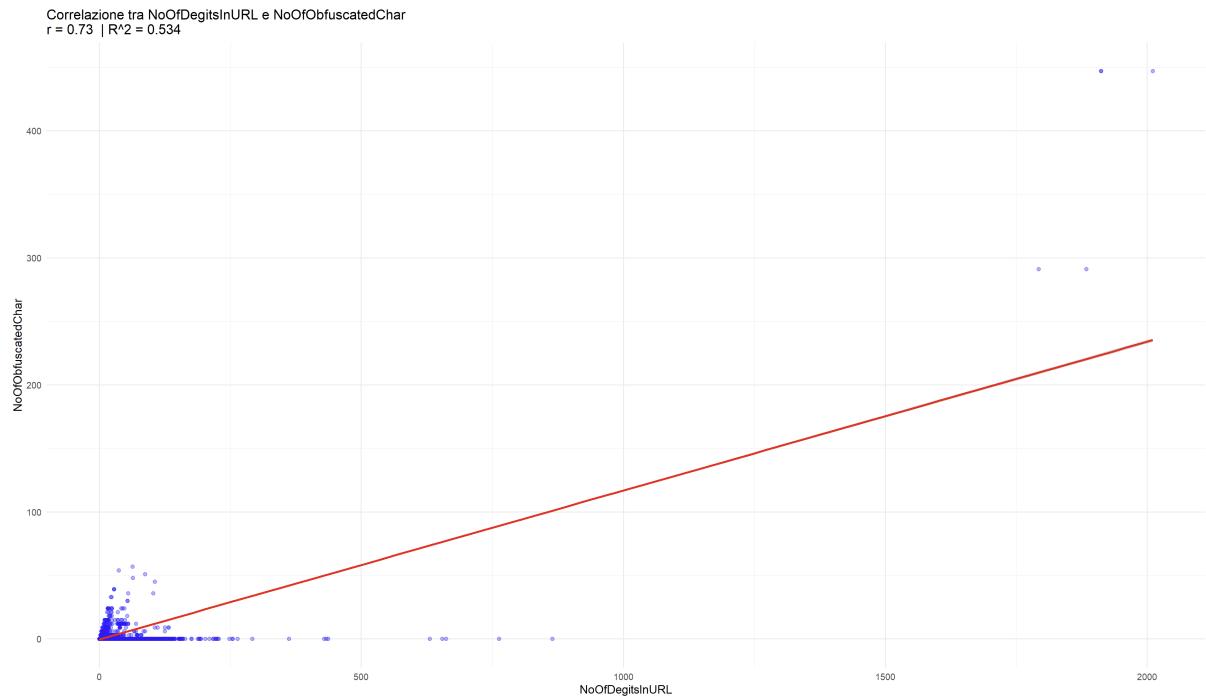


Figura 37: correlazione tra NoOfDigitsInURL e NoOfObfuscatedChar negli URL di phishing

Figura 38: Confronto correlazione con e senza outlier

I coefficienti di correlazione passano rispettivamente da 0,756 a 0,057 per NoOfEquals e da 0,73 a 0,129 per NoOfDigits. Dunque, nonostante tutti gli URL offuscati presentino almeno un '=' e un numero, mentre circa il 45% contiene almeno un '&', la maggior presenza di questi caratteri speciali non implica automaticamente un maggior numero di caratteri offuscati.

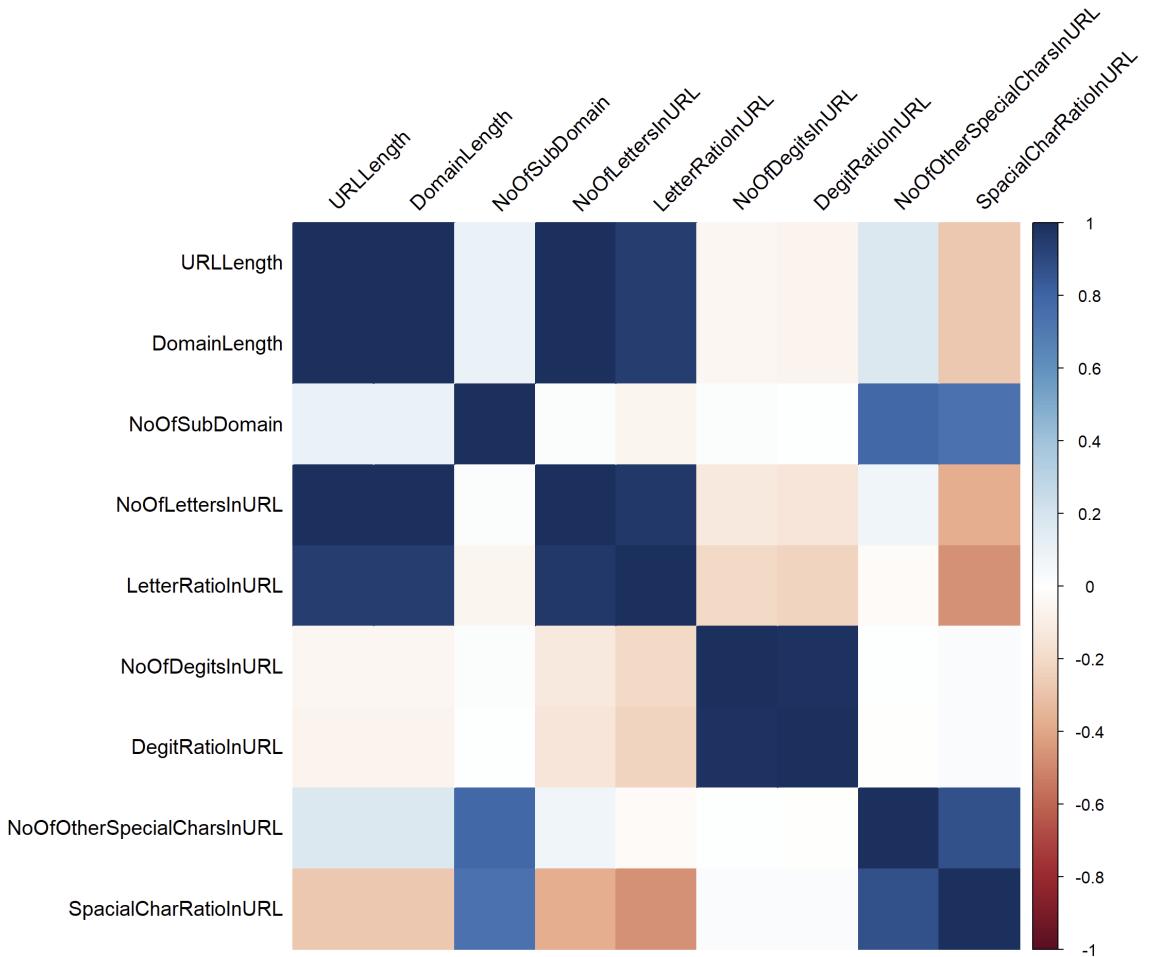


Figura 39: Matrice di correlazione tra le variabili riguardanti i caratteri negli URL legittimi. Sono stati escluse le variabili con valori nulli.

La matrice di correlazione in Figura 32 mostra la correlazione tra le variabili che riguardano la conformazione dell'URL. Dal grafico si può notare che molte delle variabili sono correlate tra loro positivamente, soprattutto con la lunghezza dell'URL.

In particolare, però, possiamo vedere come i caratteri speciali siano correlati tra loro e con il numero di caratteri offuscati, lasciando ipotizzare che questi caratteri siano proprio quelli utilizzati per l'offuscamento. L'unica variabile non correlata tra queste è NoOfQMarkInURL, probabilmente perché i punti interrogativi sono utilizzati per l'aggiunta di parametri di richiesta piuttosto che per offuscare l'URL. L'unica eccezione nella correlazione è rappresentata dalle variabili DigitRatioInURL e LetterRatioInURL, che presentano una moderata correlazione inversa, a causa degli URL composti da indirizzi IP.

In Figura 39 è invece mostrata la matrice di correlazione per le variabili riguardanti i caratteri negli URL legittimi. Le variabili NoOfQMarkInURL, ObfuscationRatio, NoOfEqualsInURL, NoOfQMarkInURL e NoOfAmpersandInURL sono state escluse, in quanto non presenti negli URL legittimi. La matrice mostra in questo caso che la quantità di lettere nell'URL ha una correlazione inversa con numeri e caratteri speciali, probabilmente perché devono condividere una più limitata quantità di caratteri, essendo gli URL legittimi più corti.

## 2.5 L'impiego degli URL shortener

Gli URL shortener sono strumenti che permettono di abbreviare link lunghi in URL più compatti, rendendo più facile la condivisione e la memorizzazione. Questi strumenti sono ampiamente utilizzati nelle piattaforme social, nei messaggi di testo, nelle email e in altri contesti digitali, dove la lunghezza dei link può essere un problema, ad esempio in spazi limitati come Twitter o nei messaggi SMS. I servizi di URL shortening creano una versione più breve di un URL, che può essere usata senza compromettere l'accessibilità del contenuto a cui punta il link.

Tuttavia, gli URL shortener sono spesso sfruttati anche per scopi dannosi, tra cui il phishing. Poiché l'URL abbreviato nasconde l'indirizzo web effettivo, gli utenti non sono in grado di verificare facilmente la legittimità del sito a cui il link punta. I criminali informatici utilizzano URL shortener per mascherare indirizzi web fraudolenti che simulano siti legittimi, ingannando così le vittime. Questo è particolarmente pericoloso in contesti come le email di phishing o i messaggi sui social media, dove un URL che sembra inoffensivo può in realtà indirizzare l'utente a un sito progettato per rubare credenziali, installare malware o eseguire altre attività dannose.

Nel nostro dataset sono presenti URL abbreviati tramite questi strumenti. Per trovarli, è stata utilizzata una lista dei più famosi URL shorteners<sup>1</sup>, da utilizzare per il confronto con la colonna Domain nel dataset. Sono stati trovati in totale 1024 URL abbreviati, tutti di phishing. Secondo il dataset, questi non sono URL offuscati, come solitamente vengono considerati; infatti, nessuno di loro ha la voce HasObfuscation positiva.

Distribuzione degli URL Shortener

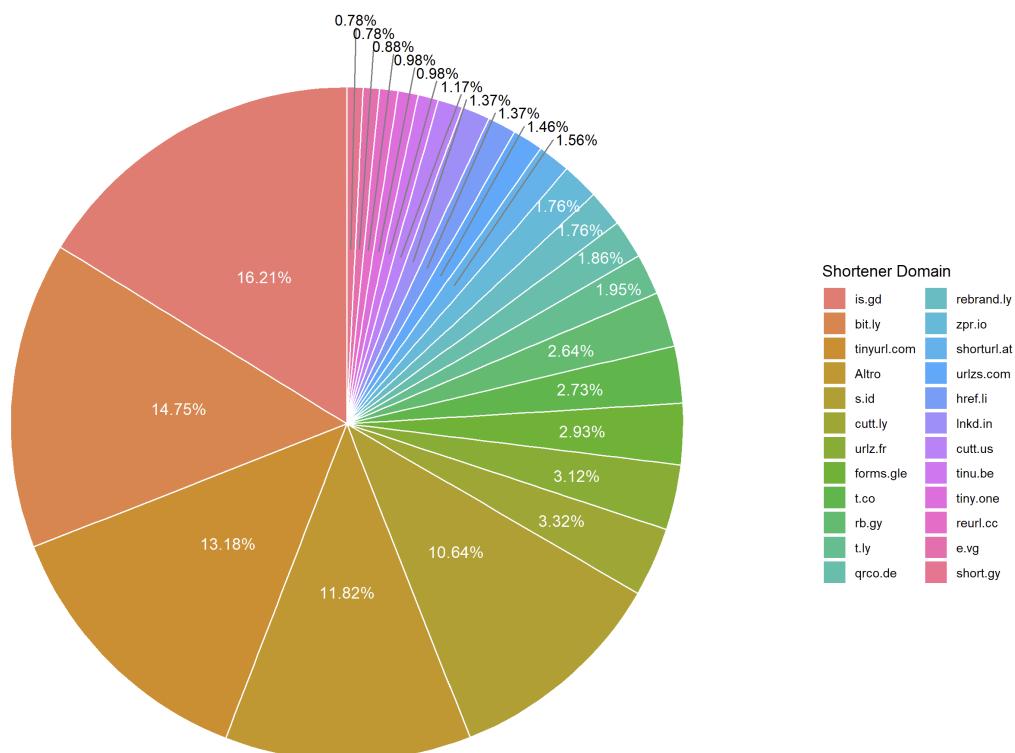


Figura 40: Grafico a torta che mostra la distribuzione degli URL shortener.

<sup>1</sup><https://github.com/PeterDaveHello/url-shorteners/blob/master/list>

## 2.6 Frequenza e distribuzione dei caratteri

Un'importante caratteristica degli URL legittimi è l'utilizzo di parole intere, non interrotte da simboli o numeri e con un'alta frequenza delle lettere dell'alfabeto più comuni, come le vocali e le consonanti che solitamente le accompagnano; al contrario, gli URL di phishing presentano spesso URL con sequenze di caratteri interrotte. Questo può essere dovuto a strategie per ingannare gli utenti, ad esempio sostituendo una lettera con un carattere simile, come altre lettere (facendo dunque aumentare la frequenza di quella determinata lettera), numeri o simboli simili. Inoltre, gli URL di pagine fraudolente possono avere una maggiore occorrenza di caratteri come 'x', 'y', 'z' e altre lettere meno frequenti nelle parole comuni.

Il dataset fornisce due variabili, CharContinuationRate e URLCharProb.

La prima misura la sequenza più lunga di caratteri alfabetici, numerici e speciali in un URL e la confronta con la lunghezza totale dell'URL. Identifica le sequenze più lunghe di caratteri alfabetici (es. "abc", "xyz", numeri (es. "123", "456") e caratteri speciali (es. "-", "...\_") e ne somma la lunghezza; divide poi il valore ottenuto per la lunghezza totale dell'URL. Un alto valore indica blocchi chiari di lettere, dunque è più probabile che l'URL sia legittimo, mentre un valore basso indica caratteri mescolati, più diffuso tra gli URL di phishing.

La seconda, invece, misura quanto un URL sia composto da caratteri alfabetici e numerici comuni nei siti legittimi rispetto a quelli più tipici nei siti di phishing. Come accennato prima, molti siti di phishing utilizzano nomi di dominio non significativi o con errori tipografici (typosquatting) per sembrare autentici. Ad esempio, un attaccante potrebbe registrare "paypa1.com" invece di "paypal.com". Per individuare questa strategia, si analizza la distribuzione dei caratteri negli URL. La variabile è stata calcolata come segue: sono stati analizzati 10 milioni di URL legittimi ed è stata calcolata la probabilità di ogni lettera (a-z) e numero (0-9). Sono stati poi analizzati 7 milioni di URL di phishing per verificare se la distribuzione dei caratteri fosse diversa. Infine, è stata calcolata URLCharProb con la formula  $\sum_n \frac{prob(URL[char_i])}{n}$ , dove  $prob(URL[char_i])$  è la probabilità di ogni carattere in base alla distribuzione ottenuta prima dai 10 milioni di URL legittimi, mentre  $n$  è il numero totale di caratteri nell'URL.

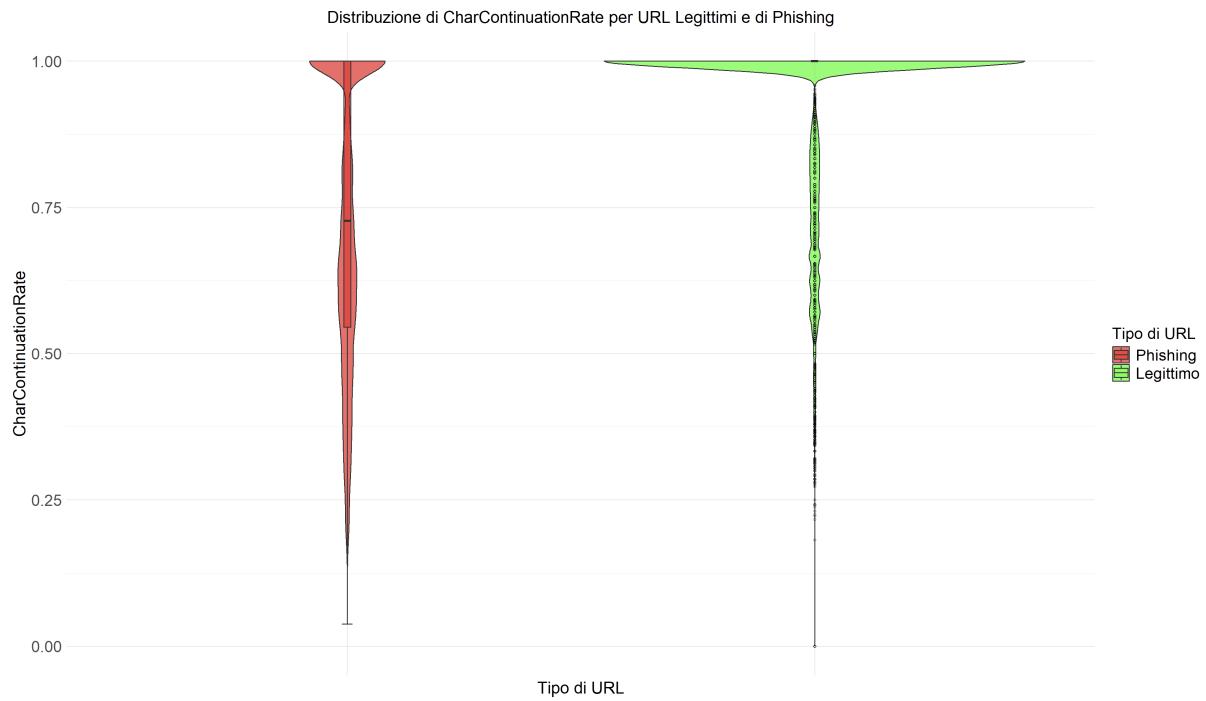


Figura 41: Distribuzione di CharContinuationRate.

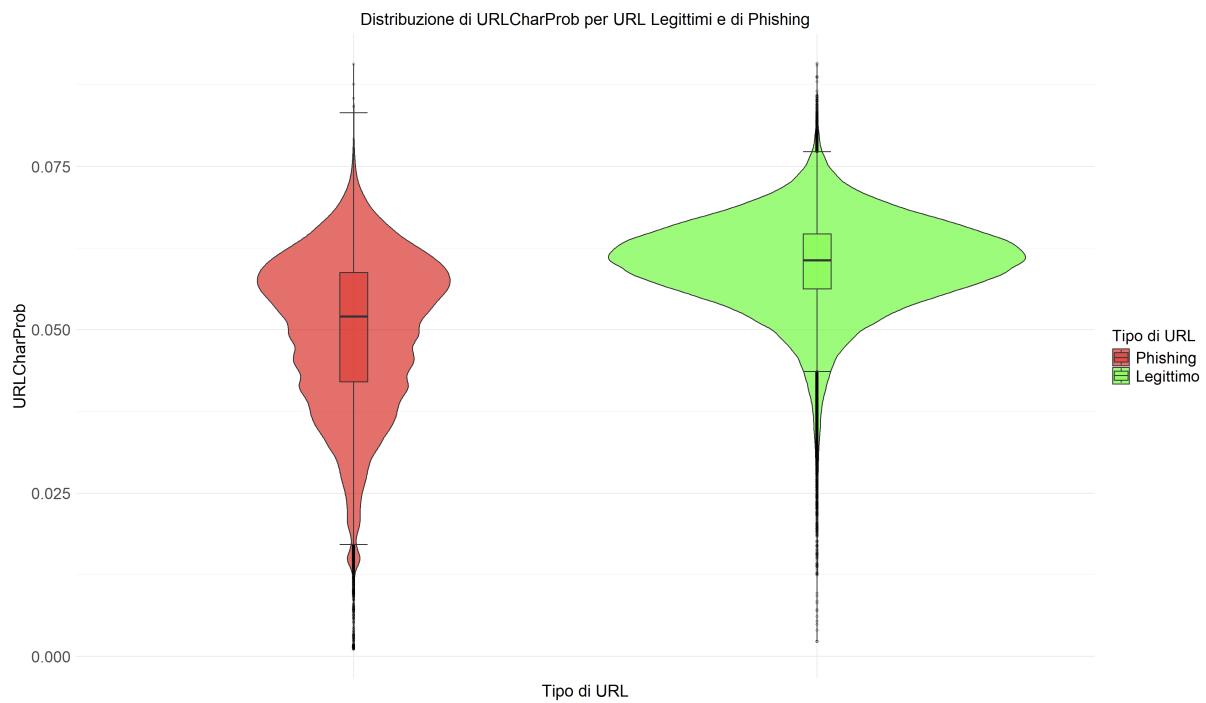


Figura 42: Distribuzione di URLCharProb.

Come previsto, in Figura 41 vediamo che gli URL di phishing tendono ad avere valori di CharContinuationRate più bassi e variabili, con una distribuzione più ampia e alcuni valori estremamente bassi. Gli URL legittimi mostrano invece una distribuzione più ristretta, concentrata su valori più alti (vicini a 1), indicando che mantengono sequenze più lunghe di caratteri omogenei (lettere, numeri o simboli consecutivi). Questo suggerisce che gli URL legittimi seguono schemi più coerenti, mentre quelli di phishing tendono a mischiare caratteri

più frequentemente, creando pattern irregolari.

Per quanto riguarda URLCharProb, gli URL di phishing assumono valori mediamente più bassi, con una distribuzione più concentrata verso lo 0.025 - 0.05. Gli URL legittimi tendono invece ad avere valori più alti, con una distribuzione più concentrata intorno a 0.05 - 0.075. Questo conferma che gli URL legittimi utilizzano caratteri considerati più comuni nelle loro strutture, mentre gli URL di phishing spesso contengono caratteri insoliti o meno frequenti.

## 2.7 Impiego del protocollo HTTPS

L'HTTPS (HyperText Transfer Protocol Secure) è una versione sicura del protocollo HTTP utilizzato per trasferire dati tra un browser e un server web. La sicurezza di HTTPS è garantita grazie a:

- Crittografia: i dati scambiati tra il browser e il server sono cifrati per prevenire intercettazioni.
- Autenticazione: HTTPS utilizza certificati digitali (come i certificati SSL/TLS) per garantire che il server sia autentico.
- Integrità: i dati trasmessi non possono essere modificati o alterati durante il trasferimento senza essere rilevati.

Questa sicurezza aggiuntiva rende HTTPS uno standard indispensabile per i siti web moderni, specialmente quelli che gestiscono informazioni sensibili come login, password, dati personali o bancari.

Sebbene HTTPS sia stato progettato per aumentare la sicurezza, i phisher hanno trovato modi per sfruttare il protocollo nei loro attacchi. Negli ultimi anni, un numero crescente di URL di phishing utilizza HTTPS per apparire legittimi.

I motivi per cui i siti di phishing possano utilizzare il protocollo HTTPS sono diversi. Il primo è sicuramente la falsa sicurezza per l'utente: molti utenti associano il simbolo del lucchetto (visibile accanto agli URL HTTPS) ad un sito affidabile. I phisher sfruttano questa percezione per ingannare le vittime.

Un altro motivo potrebbe essere la maggiore facilità di accesso ai certificati: ottenere certificati SSL/TLS, ottenibili tramite alcuni servizi, anche gratuiti, utilizzando proprio il protocollo HTTPS. Anche i siti di phishing che utilizzano HTTPS possono dunque richiedere questi certificati.

Infine, è un modo per evitare il rilevamento: i sistemi di rilevamento automatico basati su URL tendono a penalizzare maggiormente i siti senza HTTPS, quindi i phisher lo utilizzano per aggirare tali controlli.

Ad oggi utilizzare questo protocollo sicuro è diventato uno standard, dunque i siti di phishing si sono adeguati, ottenendo in questo modo la fiducia da parte degli utenti e dei sistemi di protezione, poiché rispettano tale standard; tuttavia, l'impiego di questo protocollo non garantisce l'affidabilità del sito.

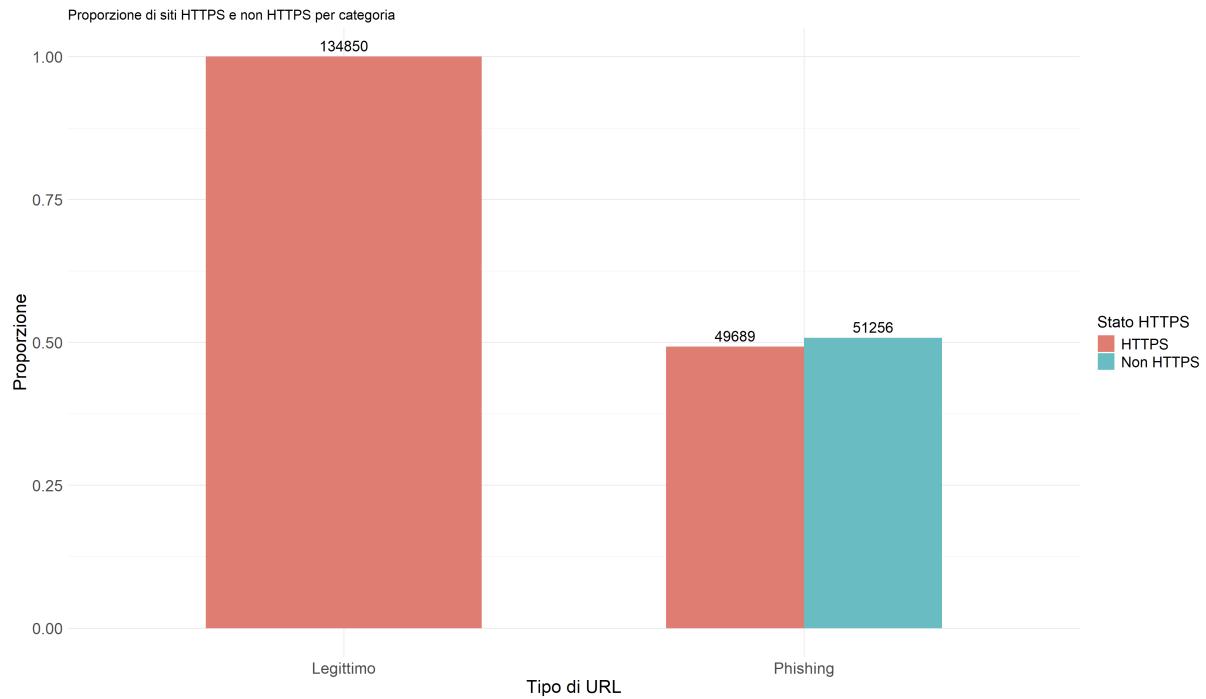


Figura 43: Distribuzione del protocollo HTTPS tra gli URL, divisi per categoria.

## 2.8 Caratteristiche del codice HTML

Il dataset possiede diverse caratteristiche HTML che possono essere utilizzate per identificare siti web di phishing analizzando il codice HTML delle pagine. Queste caratteristiche includono lunghezza del codice, presenza o assenza di alcune caratteristiche (come favicon, immagini, titolo, informazioni sul copyright), responsività del sito, presenza di reindirizzamenti, di contenuti interattivi, presenza di parole chiave, codice JavaScript e link.

### 2.8.1 Righe di codice

Le prime variabili da prendere in considerazione sono quelle che riguardano la lunghezza del codice HTML dei siti relativi agli URL. In particolare abbiamo `LineOfCode`, il numero di righe presenti, e `LargestLineLength`, ovvero la lunghezza della riga più lunga.

Categoria	Min	1° Qu.	Mediana	Media	3° Qu.	Max	Dev. St.
Legittimi	102	613	1105	1947,49	2088	442666	4348,20
Phishing	2	2	12	65,73	90	29687	195,92

Table 14: Statistiche descrittive di `LineOfCode` per i siti legittimi e i siti di phishing.

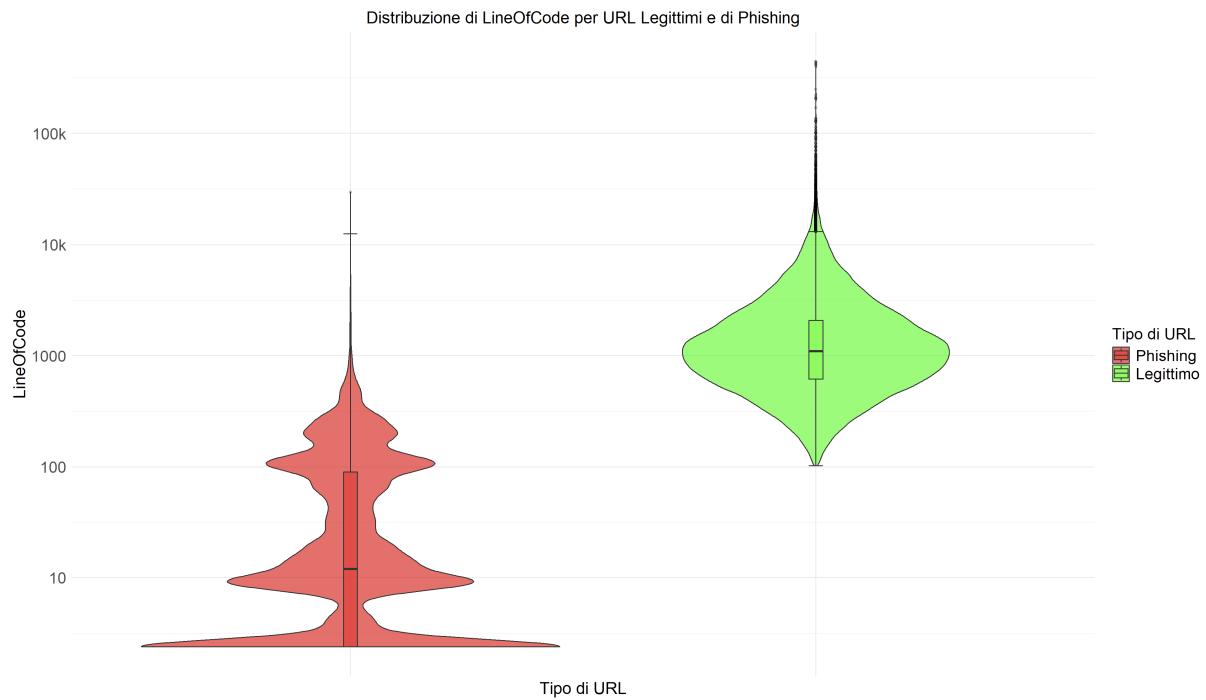


Figura 44: Distribuzione del numero di linee di codice HTML presente negli URL, diviso per URL legittimi e di phishing.

Categoria	Min	1° Qu.	Mediana	Media	3° Qu.	Max	Dev. St.
Legittimi	84	875	3066	7375,92	9381	636010	12778,2
Phishing	22	59	138	20021,45	869	13975732	231951,9

Table 15: Statistiche descrittive di LargestLineLength per i siti legittimi e i siti di phishing.

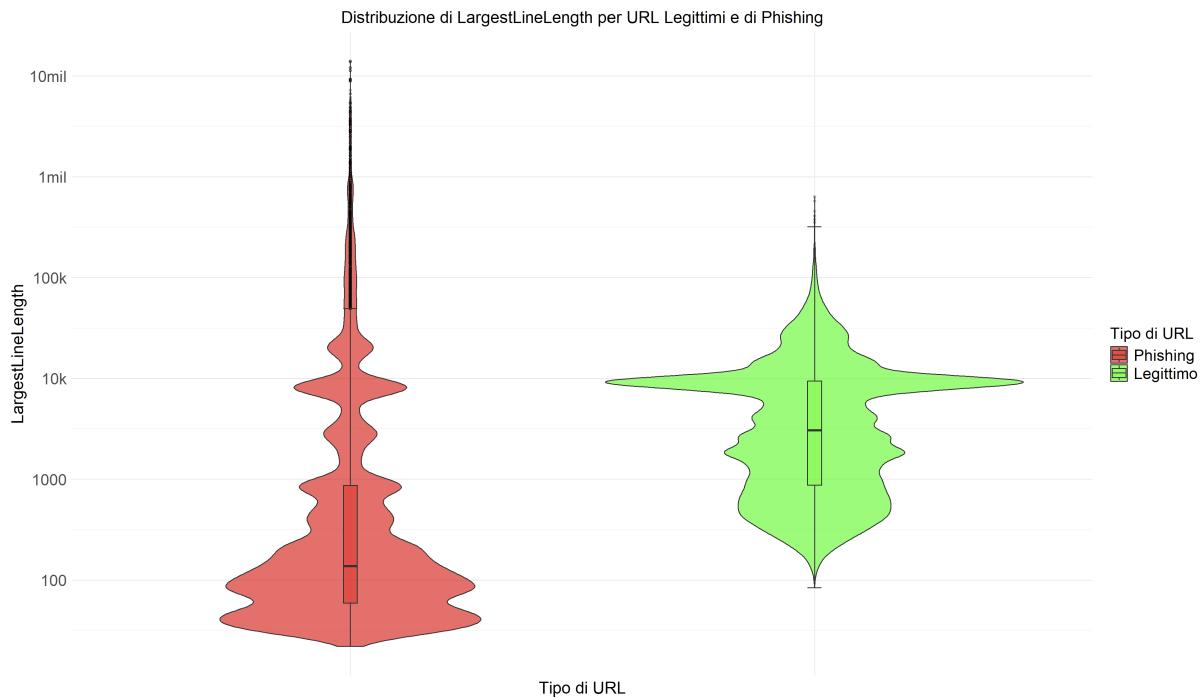


Figura 45: Distribuzione della riga di codice più lunga presente negli URL, diviso per URL legittimi e di phishing.

Come mostrano i grafici nelle Figure 44 e 45, i siti di phishing hanno in media meno righe di codice HTML, circa due ordini di grandezza in meno. Lo stesso si può dire per le righe più lunghe: negli URL di phishing la maggior parte tende ad avere una lunghezza massima inferiore, anche se in questo caso sono presenti valori estremi che superano anche il milione di righe, fino a raggiungere il massimo di 13975732.

Questa differenza nella lunghezza del codice può essere influenzata da diversi fattori. Innanzitutto, gli URL legittimi sono progettati per fornire funzionalità complete e offrire contenuti basati sullo scopo del sito, spesso massimizzando la qualità dell'esperienza utente. Questo porta ad avere un codice più complesso, più curato e con più elementi, oltre che ottimizzazioni come media query per rendere più responsivo il sito. Al contrario, i siti di phishing sono spesso progettati per imitare solo visualmente il sito legittimo, dunque potrebbero essere omesse importanti parte del codice. Ad esempio, alcuni pulsanti o altri elementi interattivi potrebbero essere solo elementi grafici a cui è stato rimosso il codice di scripting originale. Interne parti di un sito di phishing potrebbero addirittura essere semplici immagini e non vero e proprio codice. I valori massimi più alti per `LargestLineLength` nei siti di phishing potrebbero derivare da codice offuscato o inserimento di script pesanti. I criminali possono usare tecniche di codifica che concatenano lunghe stringhe di testo in un'unica riga per nascondere script dannosi.

Inoltre, alcuni siti di phishing possono incorporare script di tracciamento o codice copiato che genera righe molto lunghe.

Le differenze in questi dati sono una conseguenza degli obiettivi opposti dei siti di phishing rispetto ai siti legittimi. I siti di phishing, infatti, tendono ad avere meno righe di codice e righe più corte perché sono progettati per essere rapidi, semplici e focalizzati su obiettivi specifici (come rubare credenziali). Implementare funzionalità complesse, ad esempio, potrebbe addirittura essere controproducente per i phisher, in quanto gli utenti potrebbero distrarsi con parti del sito che non portano ad inserire le loro credenziali, perdendo preziose opportunità. I siti legittimi, invece, hanno maggiore complessità e funzionalità avanzate, che portano ad un numero maggiore di righe di codice e lunghezze più consistenti, poiché devono offrire la

migliore esperienza possibile.

### 2.8.2 Titolo della pagina

Il titolo di una pagina web è il testo visualizzato nella barra del titolo del browser o nella scheda di navigazione quando si apre una pagina. È definito all'interno del codice HTML della pagina usando il tag <title> e serve a descrivere brevemente il contenuto della pagina.

Il dataset contiene le colonne

- HasTitle: per indicare la presenza o l'assenza di un titolo,
- URLTitleMatchScore: che misura la corrispondenza tra il titolo della pagina web e il contenuto previsto dal suo URL;
- DomainTitleMatchScore: simile a URLTitleMatchScore ma utilizzando il dominio;
- Title: il titolo vero e proprio della pagina. Questa colonna non sarà utilizzata direttamente, in quanto sarebbe complesso lavorare direttamente sulle stringhe, ma può fornire un valido supporto alle analisi delle altre colonne, come vedremo successivamente.

Un punteggio basso dei due score di similarità può indicare un tentativo di phishing, poiché il titolo della pagina non corrisponde al contenuto atteso. Un punteggio alto (vicino a 100) suggerisce che il sito è autentico e coerente con ciò che dichiara di essere.

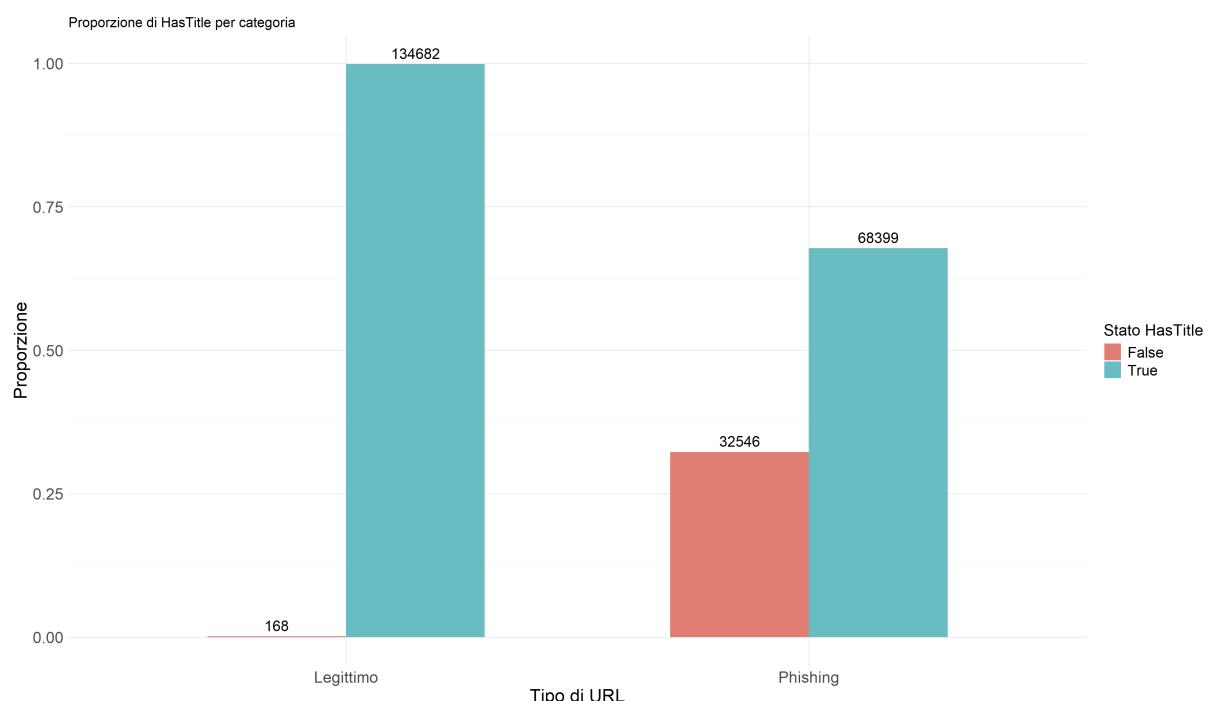


Figura 46: Distribuzione di HasTitle, diviso per URL legittimi e di phishing.

Dal grafico in Figura 46 si può notare come i siti legittimi abbiano nel 99,9% dei casi un titolo: solo 168 pagine non lo hanno. Al contrario, il 32,2% dei siti web di phishing non ha un titolo. La colonna URLTitleMatchScore mostra una distribuzione piuttosto asimmetrica tra URL legittimi e di phishing: questi ultimi, infatti, tendono ad avere un titolo non coerente con l'URL.

Table 16: Riassunto statistico di URLTitleMatchScore per URL legittimi e di phishing

Categoria	Min	Q1	Mediana	Media	Q3	Max	Dev. St.
<b>Legittimi</b>	0,00	77,78	100,00	75,27	100,00	100,00	42,68
<b>Phishing</b>	0,00	0,00	0,00	21,20	1,32	100,00	40,51

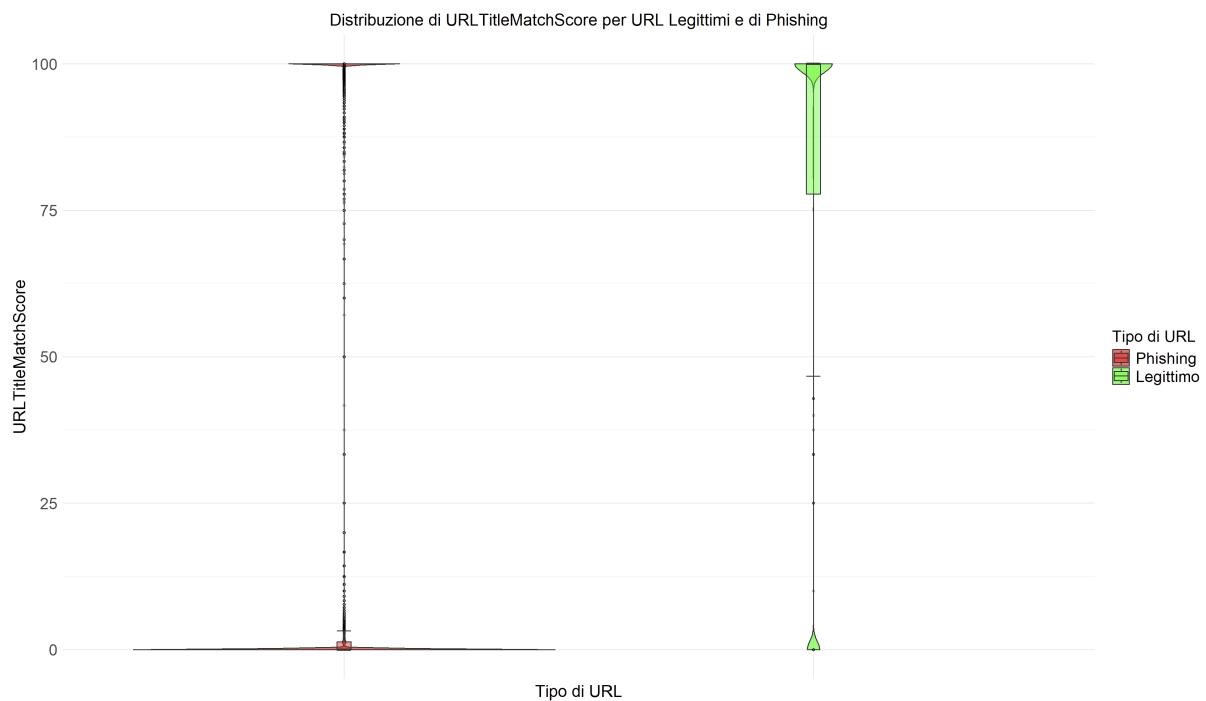


Figura 47: Distribuzione di URLTitleMatchScore, diviso per URL legittimi e di phishing, con quartili e mediana.

Dal grafico in Figura 47 si può notare come, sebbene la distribuzione sia concentrata intorno ai valori agli estremi, 0 e 100, per entrambe le categorie , la mediana e i quartili mostrano un comportamento diverso.

Per quanto riguarda gli URL legittimi, la maggior parte ha un punteggio di similarità estremamente alto, con la mediana al massimo valore (100). Ciò indica che il titolo della pagina e l'URL sono strettamente coerenti nella maggior parte dei casi. Il punteggio di similarità è concentrato nella fascia alta (tra circa 78 e 100), evidenziando che per quasi tutti gli URL legittimi, il contenuto del titolo corrisponde bene all'URL. La distribuzione è molto stretta e concentrata verso l'estremo superiore (100), con una dispersione abbastanza bassa, nonostante un buon numero di URL abbia un punteggio vicino a 0.

Guardando gli URL di phishing, invece, notiamo come la maggior parte abbia un punteggio di similarità pari a 0 e come il 75% dei dati ricada al di sotto di 1,32, il che indica che il titolo della pagina non è coerente con l'URL nella maggior parte dei casi. Anche in questo caso dunque la distribuzione è fortemente sbilanciata verso uno degli estremi (lo 0 in questo caso), ma con una lunga coda che si estende fino a 100, suggerendo che alcuni URL di phishing presentano comunque punteggi più alti, forse per ingannare gli utenti.

Table 17: Statistiche descrittive del DomainTitleMatchScore per URL legittimi e di phishing.

Tipo di URL	Min	1° Quartile	Mediana	Media	3° Quartile	Max	Dev. St.
Legittimo	0.00	77.78	100.00	75.27	100.00	100.00	42.68
Phishing	0.00	0.00	0.00	16.55	0.00	100.00	36.86

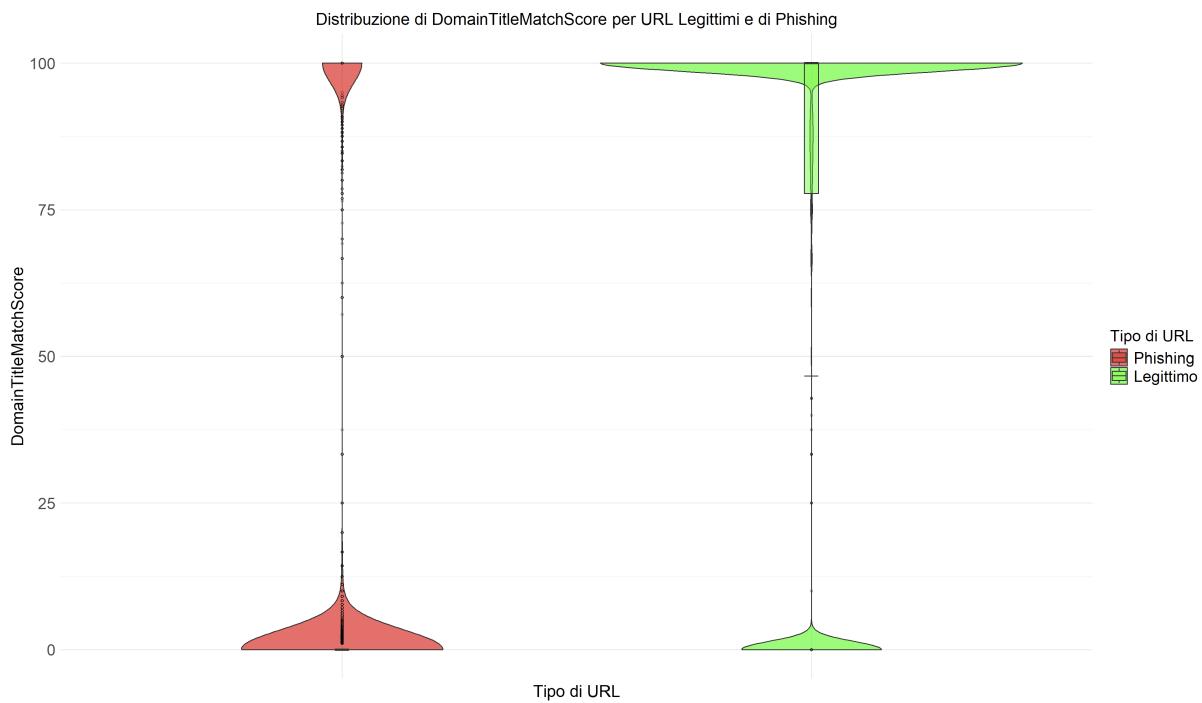


Figura 48: Distribuzione di DomainTitleMatchScore, diviso per URL legittimi e di phishing, con quartili e mediana.

Similmente a quanto accade per URLTitleMatchScore, la distribuzione di DomainTitleMatchScore è concentrata nei valori estremi, con una leggera differenza negli URL di phishing, i cui domini risultano meno coerenti con i titoli, rispetto ai loro URL. Questo potrebbe essere dovuto alla presenza di alcuni parametri dell'URL presenti nel titolo ma non nel dominio, oppure ad errori nel dataset. Solitamente, queste due colonne hanno lo stesso valore. Proviamo dunque ad estrarre tutte le righe che hanno DomainTitleMatchScore diverso da URLTitleMatchScore: otteniamo 7419 URL di phishing e 0 legittimi, che giustifica la differenza nelle statistiche descrittive sopra riportate. Calcolando la differenza tra queste due variabili negli URL di phishing, notiamo come l'DomainTitleMatchScore si discosti particolarmente dall'URLTitleMatchScore, come mostrano la mediana e i quartili.

Min.	Q1	Mediana	Media	Q3	Max.	Dev. st.
0.02	1.75	98.15	63.76	100.00	100.00	46.56

Table 18: Statistiche descrittive della differenza tra URLTitleMatchScore e DomainTitleMatchScore per URL di phishing

Questo significa che gli URL che presentano URLTitleMatchScore e DomainTitleMatchScore diversi, presentano una differenza decisamente marcata, se non completa, nella maggior parte dei casi.

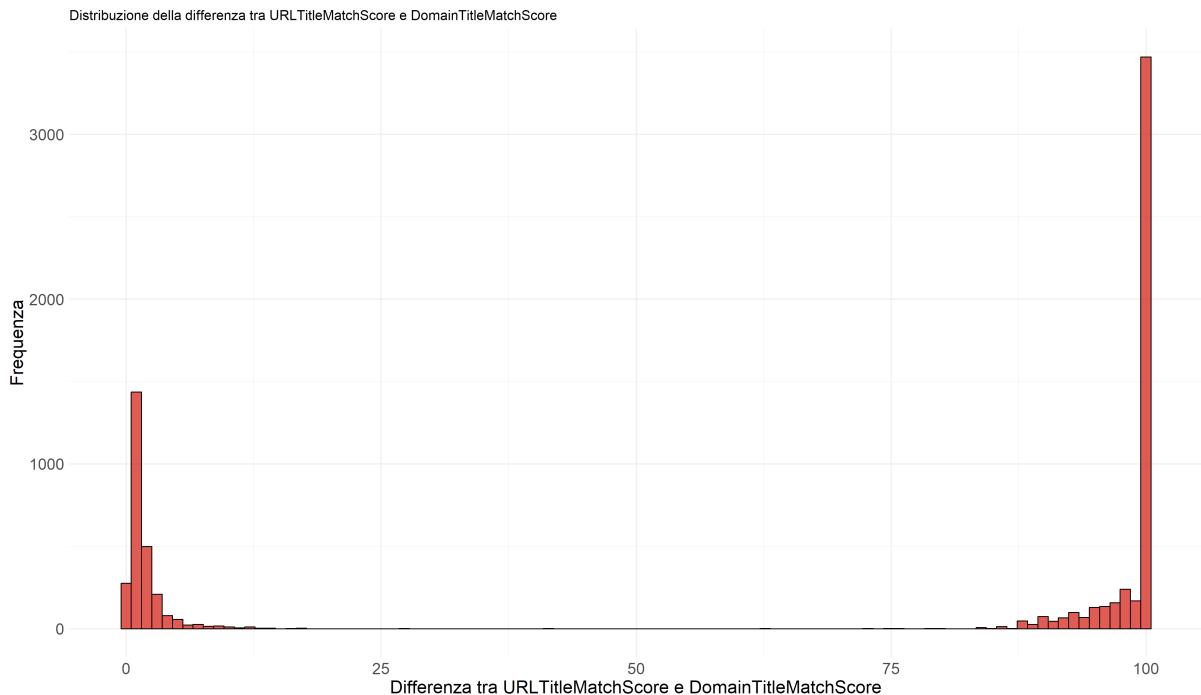


Figura 49: Distribuzione della differenza tra URLTitleMatchScore e DomainTitleMatchScore negli URL di phishing che presentano una differenza tra queste due variabili.

Tuttavia, osservando le entry del dataset, si notano alcuni dati non coerenti. Innanzitutto, alle pagine senza titolo è stato assegnato il valore 0, ed è stato comunque calcolato il valore di URLTitleMatchScore e DomainTitleMatchScore, che risultano spesso essere diversi. Probabilmente, i creatori del dataset hanno comunque calcolato questi due valori utilizzando come titolo "0" e, se nell'URL o nel dominio appare il carattere 0, il punteggio relativo aumenta. Estraendo tutte le righe senza titolo ma con differenza tra queste due variabili, ne otteniamo 2667, una buona parte delle entry inizialmente isolate. Scartando dai calcoli queste righe, otteniamo una distribuzione molto diversa ma più coerente con i dati realmente presenti nel dataset, senza distorsioni dovute a pagine senza titolo. Data l'assenza di un titolo, infatti, non ha senso controllare quanto questo sia simile ad URL e al dominio.

Min.	Q1	Mediana	Media	Q3	Max.	Dev. st.
2,53	98,82	100,00	98,54	100,00	100,00	4,16

Table 19: Statistiche descrittive della differenza tra URLTitleMatchScore e DomainTitleMatchScore per URL con titolo valido

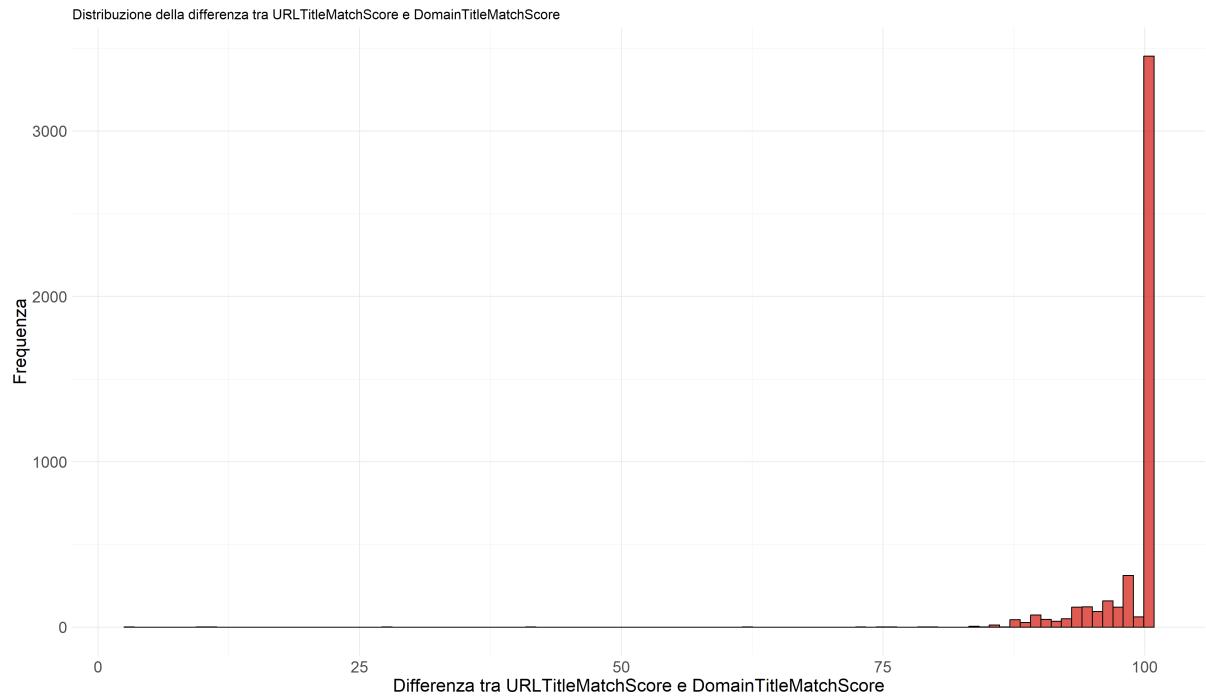


Figura 50: Distribuzione della differenza tra URLTitleMatchScore e DomainTitleMatchScore, dopo aver escluso le entry senza titolo.

Notiamo una marcata differenza nelle statistiche descrittive e nella dispersione dei dati. Dai quartili e dalla mediana notiamo subito come i valori si concentrino quasi totalmente intorno a 100; inoltre, la deviazione standard pari a 4,16 ci conferma che la dispersione è molto bassa, a differenza di quanto succedeva precedentemente.

Ricalcolando adesso tutte le statistiche, notiamo come siano cambiate decisamente le concentrazioni, poiché sono state eliminate molte righe con punteggi pari a 0. In particolare URLTitleMatchScore ha adesso il terzo quartile pari a 100 per gli URL di phishing, rispetto a quello precedente di 0.

<b>Tipo di URL</b>	<b>Min</b>	<b>1° Quartile</b>	<b>Mediana</b>	<b>Media</b>	<b>3° Quartile</b>	<b>Max</b>	<b>Dev. St.</b>
Legittimo	0.00	80.00	100.00	75.36	100.00	100.00	42.63
Phishing	0.00	0.00	0.00	31.01	100.00	100.00	46.06

Table 20: Statistiche descrittive di URLTitleMatchScore per URL legittimi e di phishing dopo pulizia dei dati senza titolo.

<b>Tipo di URL</b>	<b>Min</b>	<b>1° Quartile</b>	<b>Mediana</b>	<b>Media</b>	<b>3° Quartile</b>	<b>Max</b>	<b>Dev. St.</b>
Legittimo	0.00	80.00	100.00	75.36	100.00	100.00	42.63
Phishing	0.00	0.00	0.00	24.16	0.00	100.00	42.70

Table 21: Statistiche descrittive di DomainTitleMatchScore per URL legittimi e di phishing dopo pulizia dei dati senza titolo.

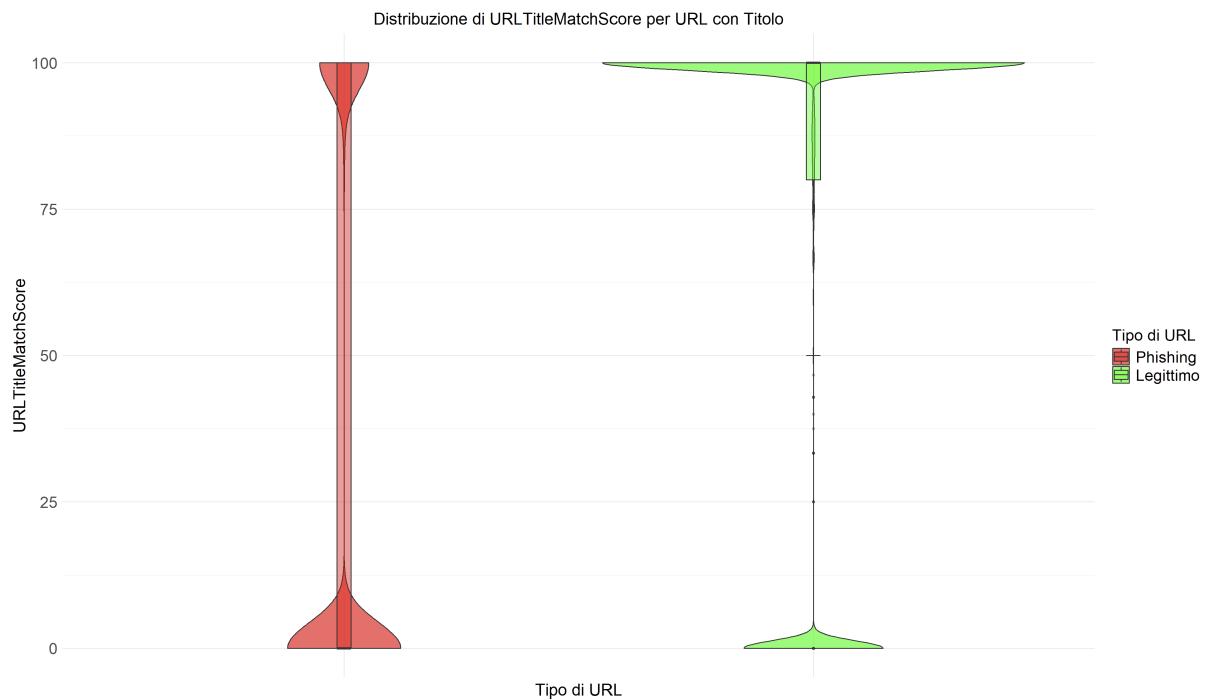


Figura 51: Distribuzione di URLTitleMatchScore, diviso per URL legittimi e di phishing, dopo pulizia dei dati senza titolo.

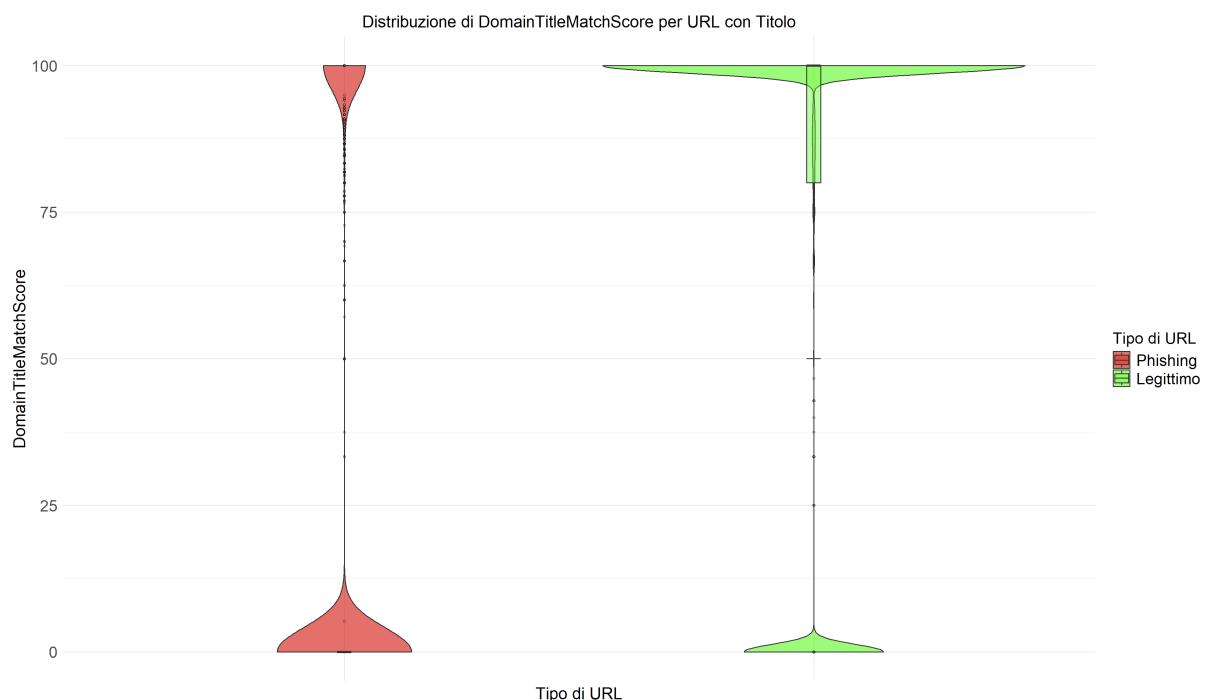


Figura 52: Distribuzione di DomainTitleMatchScore, diviso per URL legittimi e di phishing, dopo pulizia dei dati senza titolo.

Le differenze sono decisamente più accentuate negli URL di phishing, in quanto sono questi principalmente a non avere un titolo. In particolare sono diminuiti i valori pari o prossimi allo 0, oltre ad alcuni valori che già avevano minore frequenza, compresi tra 0 e 100.

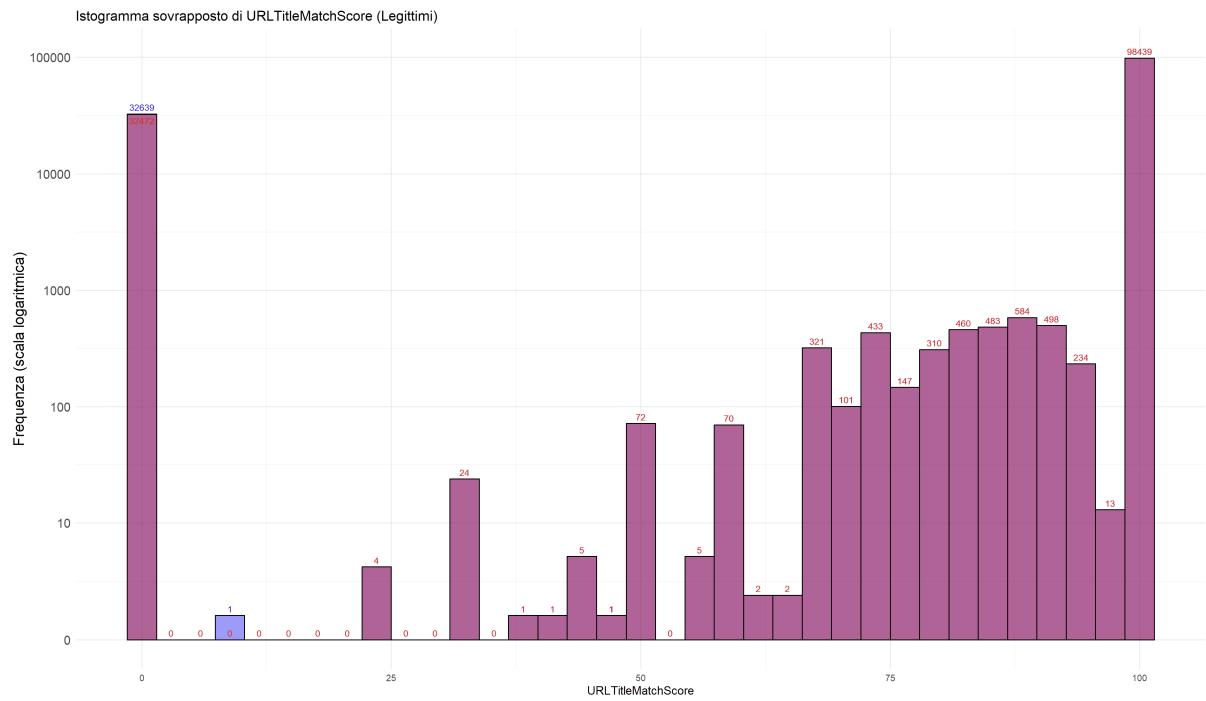


Figura 53: Confronto della distribuzione di URLTitleMatchScore, dopo pulizia dei dati senza titolo, per URL legittimi.

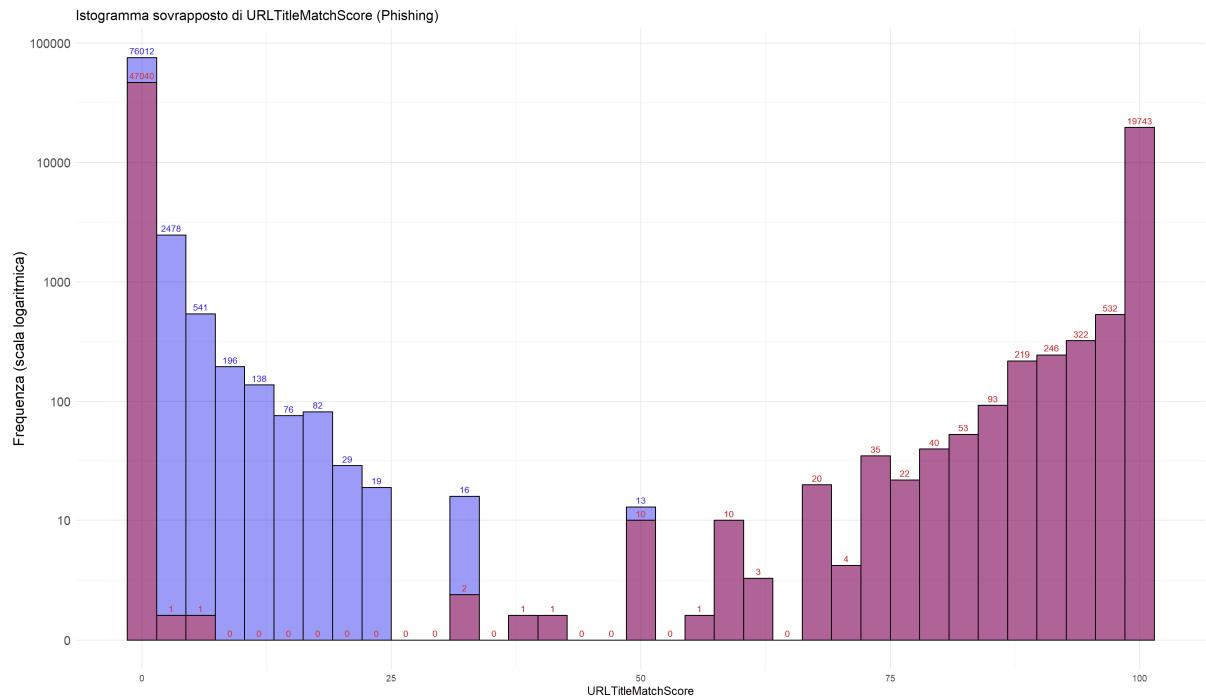


Figura 54: Confronto della distribuzione di URLTitleMatchScore, dopo pulizia dei dati senza titolo, per URL di phishing.

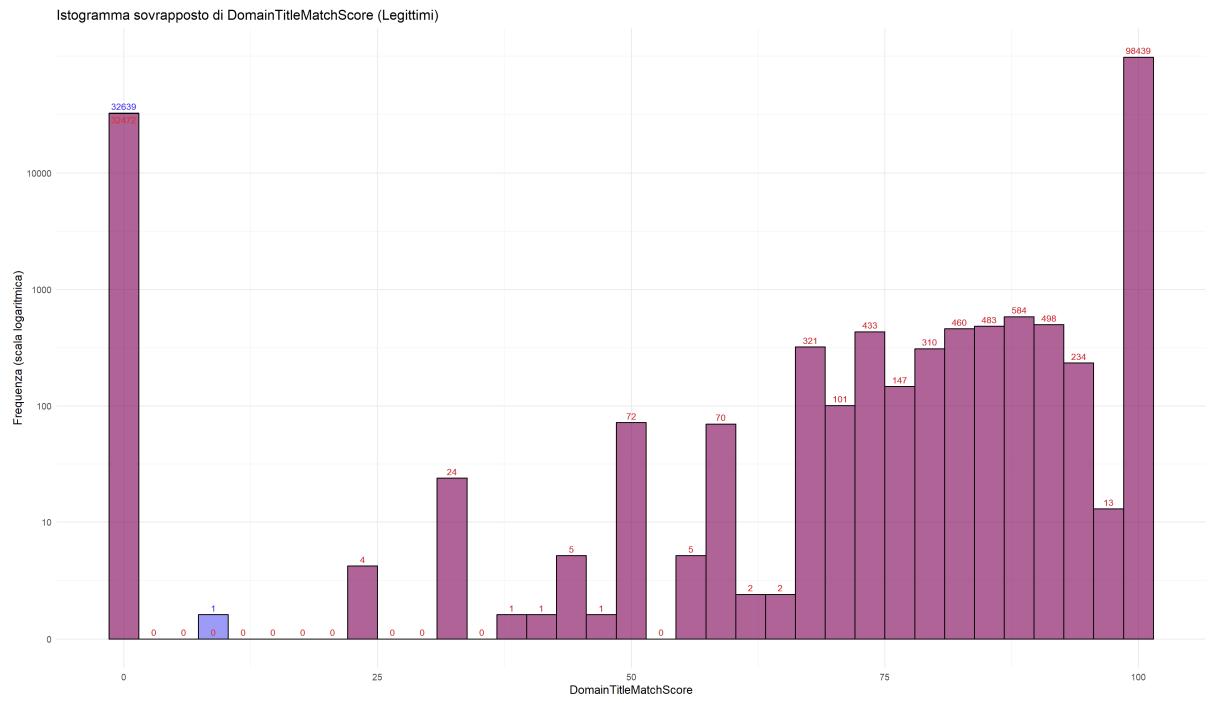


Figura 55: Confronto della distribuzione di DomainTitleMatchScore, dopo pulizia dei dati senza titolo, per URL legittimi.

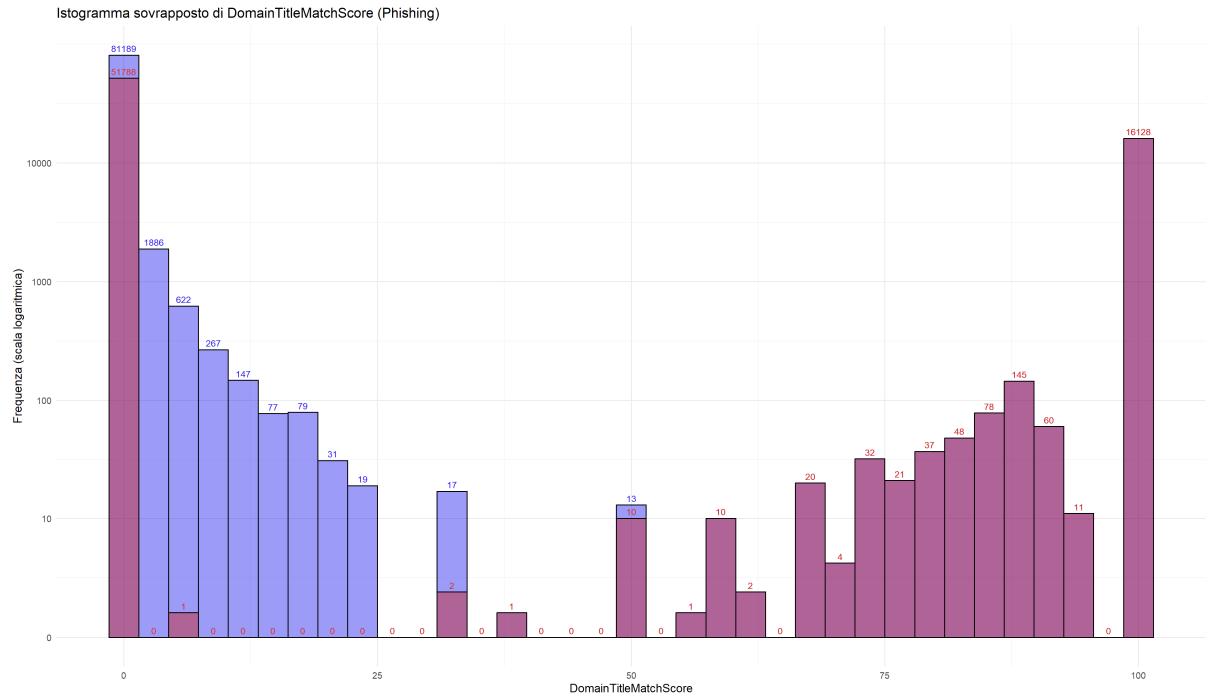


Figura 56: Confronto della distribuzione di DomainTitleMatchScore, dopo pulizia dei dati senza titolo, per URL di phishing.

Sebbene dopo questa pulizia dei dati otteniamo risultati più coerenti, queste due variabili non sono comunque del tutto accurate. I creatori del dataset, infatti, hanno ottenuto questi punteggi tokenizzando URL, dominio e titolo di ogni entry. Tuttavia, la tokenizzazione si basa su dei caratteri di interruzione delle stringhe, che possono essere simboli o spazi, ma non lettere.

Questo porta spesso a valutare un dominio diverso dal titolo solo per la presenza di lettere aggiuntive, che creano un token diverso, nonostante sia molto somigliante con il token preso in considerazione. Ad esempio, l'URL "<https://carpediemsocial.com/optus>" con "carpediemsocial.com" come dominio e "carpediemsocialoptu" come titolo. Sebbene la somiglianza sia piuttosto elevata, sia per l'URL che per il dominio l'URLTitleMatchScore è 95 mentre il DomainTitleMatchScore risulta invece 0.

Estraendo tutte le entry con un titolo, che abbiano DomainTitleMatchScore uguale a 0 e URLTitleMatchScore maggiore o uguale ad 80, otteniamo 4741 entry, molte delle quali presentano il problema sopra citato.

Non è ben chiaro perché il dominio non riesca ad essere rilevato come simile al titolo, a differenza dell'URL; tuttavia, data la complessità di creare un sistema che possa effettivamente correggere questo comportamento, si potrebbe preferire l'utilizzo di URLTitleMatchScore per eventuali analisi.

### 2.8.3 Caratteristiche estetiche ed informazioni

Il dataset contiene diverse colonne che si riferiscono ad altre caratteristiche HTML e ad elementi delle pagine web. Questi possono essere informazioni o caratteristiche di tipo estetico, come favicon, responsività, descrizione, informazioni sul copyright; oppure possono riguardare elementi riferiti a dati bancari o di pagamento o ancora a codice JavaScript o CSS.

Le prime colonne che analizzeremo sono: HasFavicon, Robots, IsResponsive, HasDescription, HasSocialNet, HasCopyrightInfo. Non è ben chiaro a cosa si riferisca la colonna Robots, in quanto è completamente assente nella documentazione del dataset. Potremmo ipotizzare che si riferisca alla presenza (valore 1) o all'assenza (valore 0) del file *robots.txt*, utilizzato per implementare il protocollo di esclusione dei robot, uno standard utilizzato dai siti Web per indicare ai web crawler e ad altri robot Web in visita quali parti del sito Web sono autorizzati a visitare.

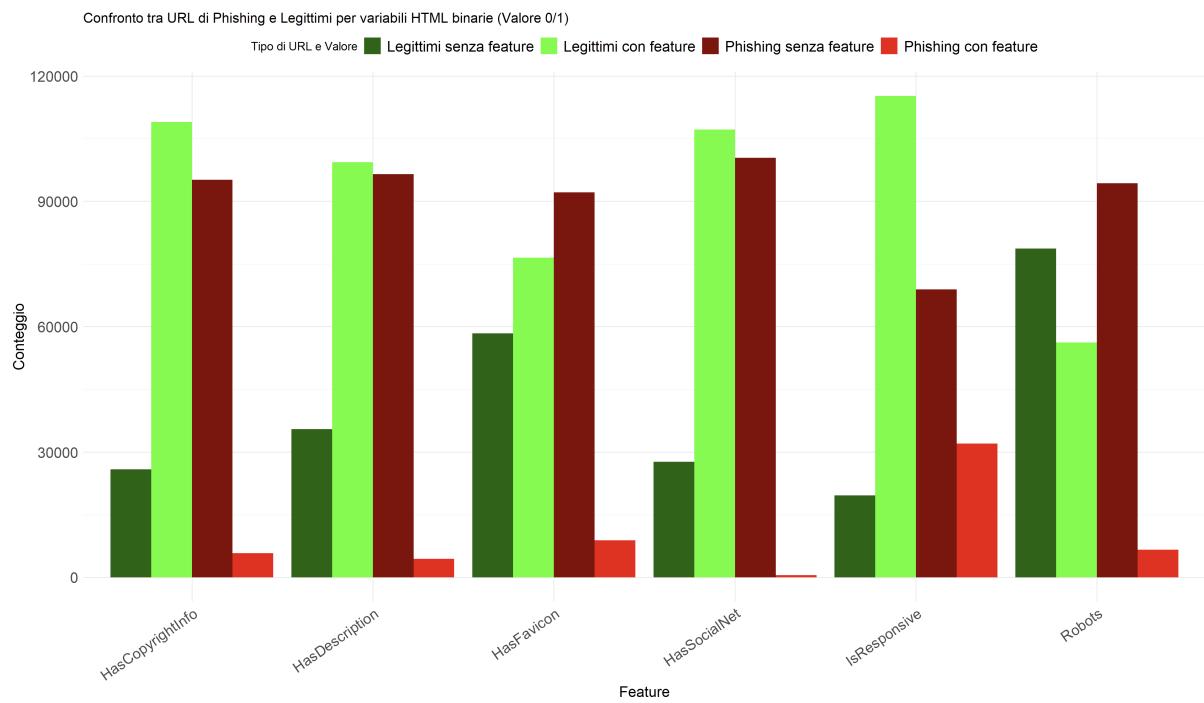


Figura 57: Confronto della distribuzione di HasFavicon, Robots, IsResponsive, HasDescription, HasSocialNet, HasCopyrightInfo, per URL legittimi e di phishing.

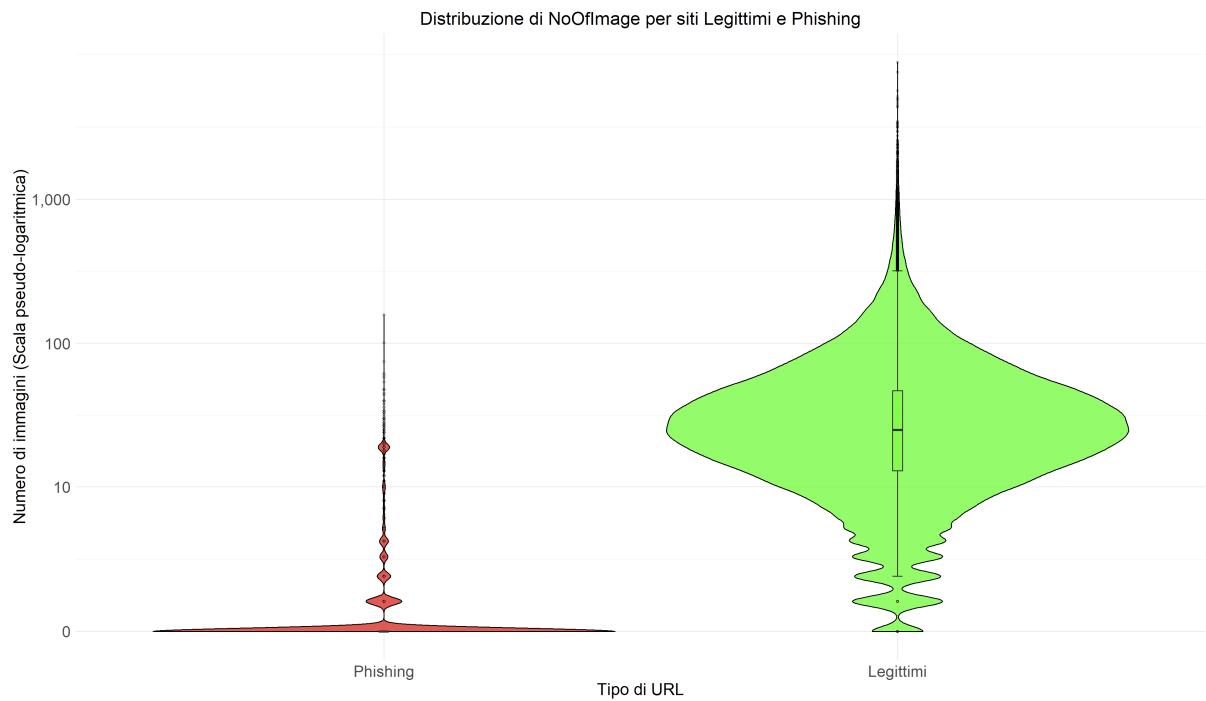


Figura 58: Confronto della distribuzione di NoOfImage per URL legittimi e di phishing.

Dal grafico in figura 57 e 58 è possibile osservare che le pagine di phishing tendono ad essere meno curate dal punto di vista delle feature HTML, delle informazioni e delle immagini, a differenza degli URL legittimi, che, sebbene con variabilità, tendono ad applicare le buone pratiche per una pagina web ben curata, oltre che contenuti multimediali utili per i loro servizi. Le feature meno applicate dagli URL legittimi sono Robots e HasFavicon, mentre per gli URL di phishing sono HasSocialNet e HasDescription. Sorprendentemente, gli URL di phishing tendono ad applicare la responsività nelle loro pagine in proporzione maggiore rispetto alle altre feature, forse per imitare maggiormente i siti web legittimi.

#### 2.8.4 Campi per immissione di dati e riferimenti bancari

Tra le varie caratteristiche HTML, sono presenti diverse colonne riferite a dati bancari, di pagamento o a campi per l'inserimento di password e altri dati.

Le colonne analizzate sono: Bank, Crypto, Pay, HasSubmitButton, HasHiddenFields, HasExternalFormSubmit e HasPasswordField.

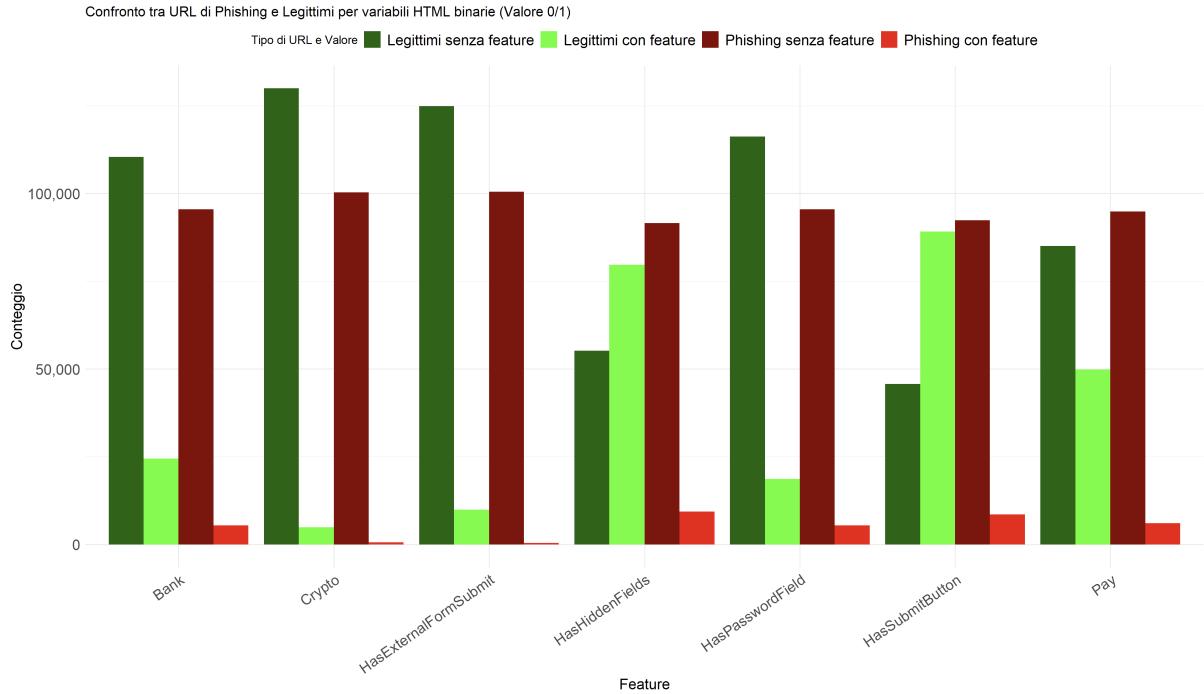


Figura 59: Confronto della distribuzione di Bank, Crypto, Pay, HasSubmitButton, HasHiddenFields, HasExternalFormSubmit e HasPasswordField, per URL legittimi e di phishing.

Sebbene ci si aspetterebbe che gli URL di phishing abbiano un numero significativo di campi relativi a pagamenti o password, poiché uno degli obiettivi principali dei siti di phishing è rubare dati sensibili, ciò non emerge dai dati del dataset. I motivi potrebbero essere dovuti ad un'errata raccolta dei dati, oppure a strategie di phishing diverse, ad esempio tramite il download di malware oppure reindirizzando ad altre pagine dove effettivamente saranno richiesti i dati da rubare.

### 2.8.5 Redirect, riferimenti esterni e popup

Nel dataset sono presenti colonne riferite proprio a reindirizzamenti, sia esterni che interni e a popup. Questi possono essere utilizzati per richiedere dati sensibili e rubarli. In particolare le colonne in questione sono: NoOfPopup, NoOfFrame, NoOfURLRedirect, NoOfSelfRedirect, NoOfSelfRef, NoOfEmptyRef e NoOfExternalRef. Proviamo dunque a verificare l'ipotesi precedentemente avanzata.

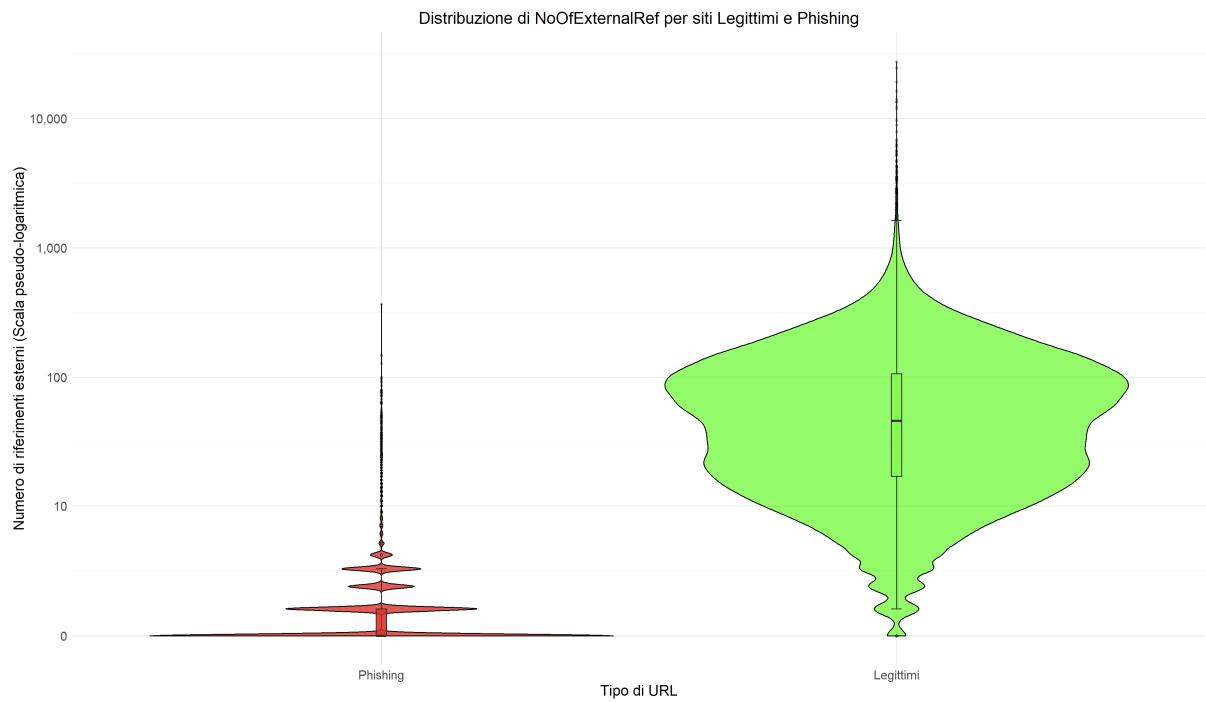


Figura 60: Confronto della distribuzione di NoOfExternalRef per URL legittimi e di phishing.

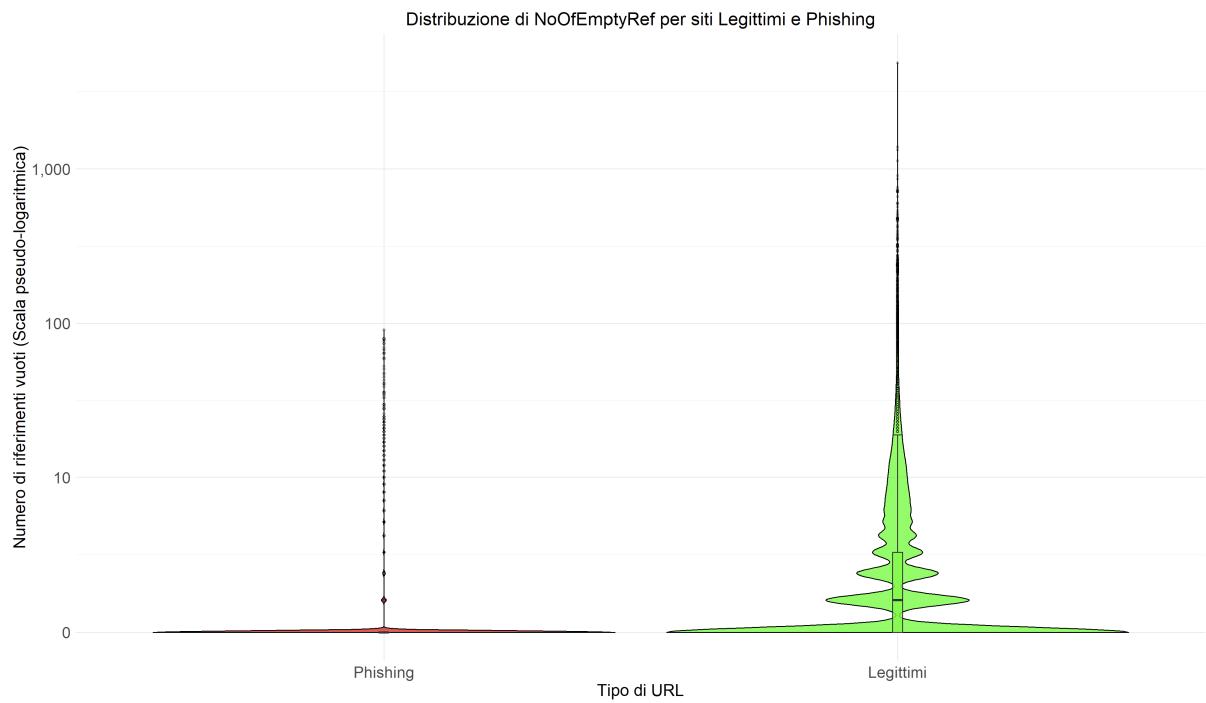


Figura 61: Confronto della distribuzione di NoOfEmptyRef per URL legittimi e di phishing.

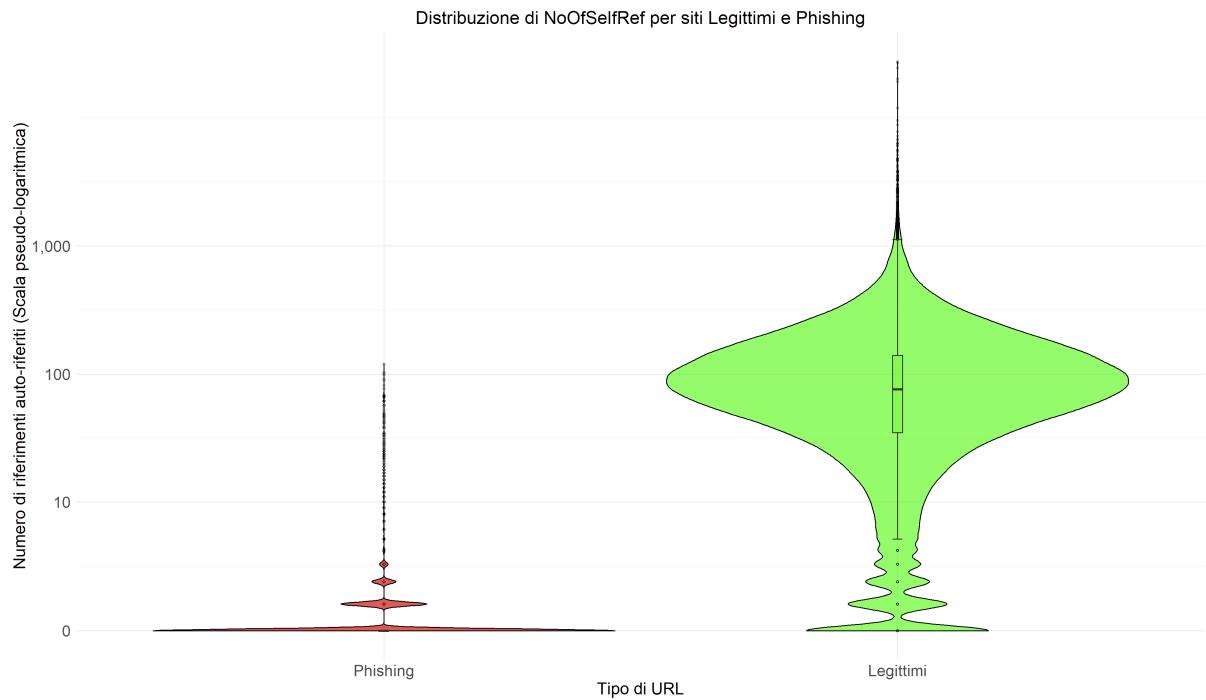


Figura 62: Confronto della distribuzione di NoOfSelfRef per URL legittimi e di phishing.

Le figure 60, 61 e 62 evidenziano che la maggior parte degli URL di phishing non include hyperlink cliccabili, in particolare quelli esterni, vuoti o auto-riferiti. Tuttavia, l'uso di riferimenti esterni è più frequente: infatti, 45.685 URL di phishing (pari al 45,26%) comprendono almeno un riferimento esterno. Questo potrebbe suggerire che l'utente sia invitato a inserire i propri dati nel nuovo sito raggiunto dopo aver cliccato sul link. Inoltre, è legittimo ritenere che, quando una pagina di phishing incorpora link a siti esterni, questi siano limitati a uno o pochi siti, poiché l'intenzione è di dirigere l'utente verso il sito fraudolento.

Gli URL legittimi invece sembrano adottare ampiamente questi riferimenti, probabilmente per una più variegata navigazione tra le pagine del sito e verso l'esterno, in relazione ai servizi offerti.

Le pagine web di phishing che includono campi per far inserire all'utente i propri dati sensibili, potrebbero non utilizzare riferimenti esterni per evitare di far uscire l'utente dalla pagina, cercando di trattenerlo il più possibile per aumentare le probabilità di ottenere quelle informazioni.

In generale i riferimenti vuoti sono meno presenti, sia tra URL legittimi che di phishing, in quanto spesso sono dovuti ad errori, dimenticanze o a codice incompleto in attesa di modifiche. Un altro metodo utilizzato per indirizzare gli utenti verso siti esterni è l'impiego dei reindirizzamenti automatici. Questi sono attivati attraverso script che, una volta eseguiti, rimandano gli utenti a un altro URL. Spesso il reindirizzamento avviene in modo invisibile, ad esempio al momento del caricamento della pagina, senza che l'utente si accorga del passaggio. Nel caso degli URL di phishing, può essere adottata una strategia che prevede una landing page apparentemente innocua per eludere i controlli automatici, seguita da un reindirizzamento verso una pagina fraudolenta progettata per ingannare l'utente.

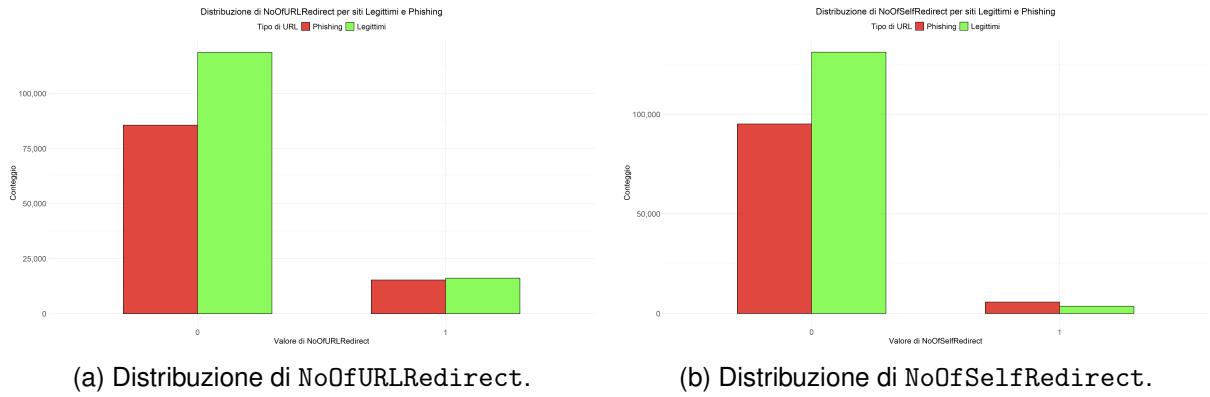


Figura 63: Confronto della distribuzione delle variabili NoOfURLRedirect, NoOfSelfRedirect per URL legittimi e di phishing.

Analogamente ai riferimenti, i reindirizzamenti vengono impiegati solo da una parte minoritaria degli URL di phishing. Nonostante ciò, si tratta comunque di un numero consistente, ovvero 15.313 URL, che equivale al 15,17% degli URL di phishing. Inoltre, anche gli URL legittimi tendono a non utilizzarli in modo diffuso. Nel dataset sono inclusi solo URL che possiedono zero o un solo reindirizzamento, poiché dopo un singolo reindirizzamento l'utente non subirebbe ulteriori reindirizzamenti.

I rendirizzamenti auto-riferiti, meno diffusi, sono probabilmente collegati a script che causano il refresh della pagina, causando appunto un reindirizzamento verso la pagina stessa.

Il dataset contiene due colonne, riferite al numero di popup e di iFrame presenti nel sito. La colonna NoOfPopup si riferisce a degli elementi dell'interfaccia grafica, quali finestre o riquadri, che compaiono automaticamente durante l'uso di un'applicazione ed in determinate situazioni, per attirare l'attenzione dell'utente; la colonna NoOfiFrame si riferisce invece a degli elementi HTML composti da frame "ancorati" all'interno della pagina, come se fosse un frame figlio della pagina aperta.

Questi due elementi possono essere utilizzati dai siti di phishing per inserire elementi malevoli in una pagina che sembra legittima, poiché i dati da rubare saranno richiesti all'interno di questi componenti.

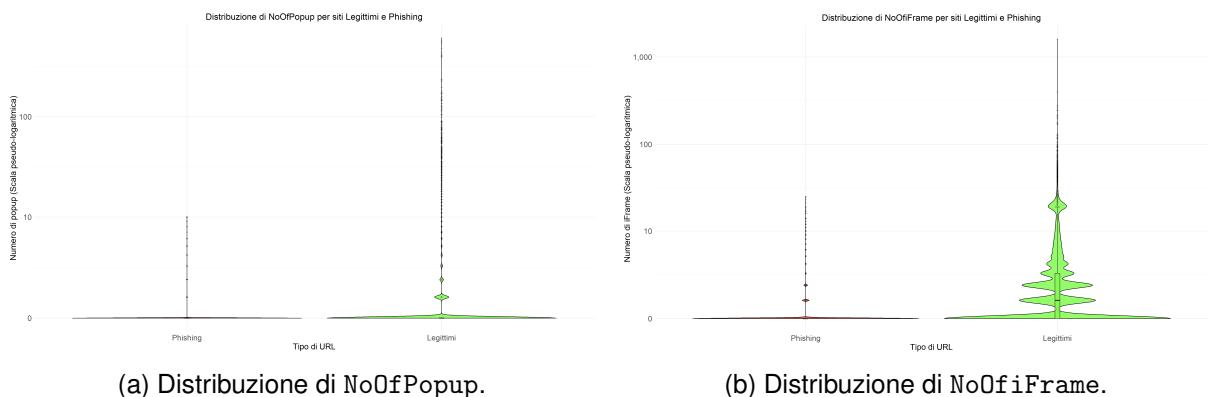


Figura 64: Confronto della distribuzione delle variabili NoOfPopup, NoOfiFrame per URL legittimi e di phishing.

Nonostante le potenzialità per quanto riguarda il phishing, sono pochi i siti fraudolenti che adottano questi due elementi, in particolare i popup. Infatti, sono 4913 le pagine di phishing

che contengono almeno un iFrame, mentre 526 mostrano almeno un popup.

Basandoci sui dati esaminati, non è ancora del tutto chiaro quali tattiche di phishing siano impiegate dalla maggioranza degli URL dannosi, poiché le caratteristiche disponibili che potrebbero chiarire questo aspetto non sono utilizzate come previsto.

### 2.8.6 Codice CSS e JS

Le ultime colonne presenti nel grafico sono quelle riferite al codice CSS e Javascript. Queste sono probabilmente riferite al numero di script presenti per entrambi i tipi.

Il CSS viene impiegato per la formattazione delle pagine web, svolgendo quindi un ruolo estetico e interattivo, e può offrirci indizi sulla qualità e l'attenzione dedicata al sito. Similmente a quanto osservato in precedenza, una maggiore quantità di codice CSS potrebbe suggerire che il sito sia legittimo, in quanto mostrerà un aspetto estetico più dettagliato e sofisticato, caratteristica meno comune nelle pagine fraudolente.

JavaScript è invece un linguaggio di programmazione utilizzato in ambito web, sia alto client che server, per implementare funzioni di script per ottenere effetti dinamici, gestire eventi e in generale sviluppare la logica di business di una pagina web. Tra i vari utilizzi, può essere utilizzato da pagine fraudolente per rubare i dati dell'utente, anche senza richiedere la pressione di un pulsante da parte di quest'ultimo.

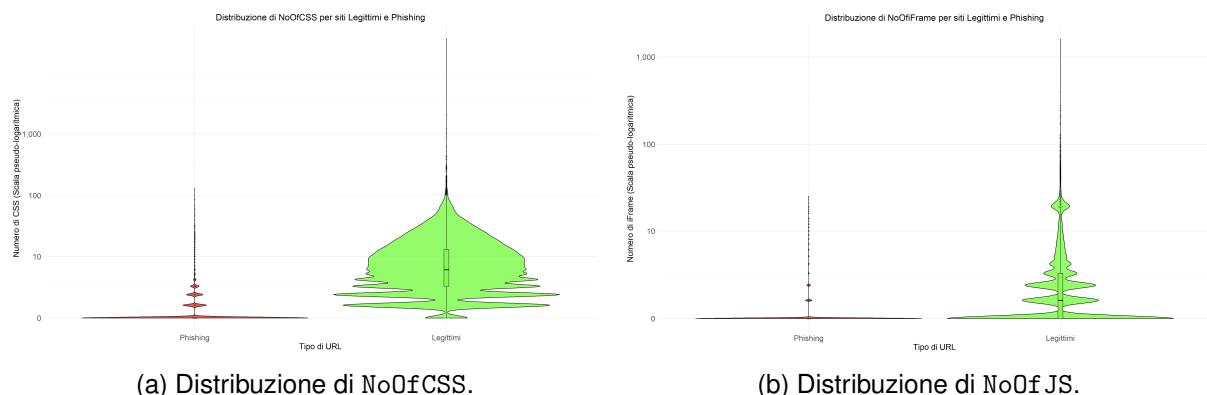


Figura 65: Confronto della distribuzione delle variabili NoOfCSS, NoOf JS per URL legittimi e di phishing.

Gli script sono minori per le pagine di phishing, probabilmente a causa della loro minore complessità e qualità.

I siti legittimi hanno generalmente un numero maggiore di file CSS rispetto ai siti di phishing, inoltre la distribuzione per i siti legittimi è più ampia e presenta una maggiore variabilità. Nei siti di phishing, invece, il numero di CSS è molto limitato, con una concentrazione elevata vicino allo zero.

Anche nel caso degli script JavaScript, i siti legittimi mostrano una distribuzione più ampia del numero di codice JavaScript rispetto ai siti di phishing, e un numero significativamente più alto di script, mentre i siti di phishing tendono ad utilizzarne molti meno.

## 2.8.7 Altre colonne

Terminata l'analisi delle colonne più significative del dataset, ne rimangono alcune che non portano informazioni utili per l'analisi. Queste sono: FILENAME: la documentazione del dataset la riporta con ruolo "*Other*", a differenza delle altre colonne che hanno ruolo "*Feature*"; inoltre, è specificato che la colonna può essere ignorata. Probabilmente si riferisce al titolo del file con cui sono stati salvati i dati di ogni URL prima di essere aggregati nel dataset. IsDomainIp: indica se il dominio è formato da un indirizzo IP. Nonostante questa feature riguardi solo URL di phishing, questi sono solo 638; inoltre, non aggiunge informazioni alle analisi svolte.

### 3 Classificazione degli URL con SVM

Una Support Vector Machine (SVM) è un algoritmo di apprendimento con supervisione, utilizzato in molti problemi di classificazione e regressione.

Le SVM cercano di trovare un iperpiano (o una superficie di separazione) che divida i dati in modo da separare al meglio le classi. L'idea è di massimizzare il margine, cioè la distanza tra l'iperpiano e i punti più vicini di ciascuna classe (chiamati support vectors). Un margine più ampio tende a garantire una maggiore capacità di generalizzazione del modello su dati non visti. I support vectors sono i punti che si trovano più vicini all'iperpiano. Questi punti sono cruciali perché determinano la posizione e l'orientamento dell'iperpiano stesso.

La scelta di usare una SVM è dovuta a diversi motivi: Innanzitutto, il dataset contiene informazioni raccolte da siti web, differenziando quelli legittimi da quelli di phishing. L'obiettivo finale è dunque quello di studiare queste differenze per poter stabilire se è possibile classificare i siti web utilizzando suddette caratteristiche. A questo proposito, la SVM è particolarmente indicata per problemi di classificazione con un numero di feature piuttosto alto, come nel nostro caso. Durante il corso è stato accennato anche il K-Nearest Neighbors che può essere similmente usato per classificare le istanze, calcolandone la distanza rispetto ai vicini. Per classificare un nuovo dato, il metodo calcola la distanza rispetto a tutti i punti del training set e sceglie i k vicini più prossimi per decidere la classe. È stata però preferita la SVM in quanto risulta più adatta con numeri di feature elevati.

#### 3.1 Scelta delle feature e pre-processing dei dati

Il primo passo è scegliere le feature da utilizzare per il modello di classificazione, evitando di inserire nell'addestramento feature ridondanti o irrilevanti. A questo proposito, sono state selezionate le seguenti feature per iniziare, in base alle differenze tra le distribuzioni studiate precedentemente:

- URLLength (discreta),
- NoOfSubDomain (discreta),
- CharContinuationRate (continua),
- URLCharProb (continua),
- NoOfOtherSpecialCharsInURL (discreta),
- LineOfCode (discreta),
- NoOfImage (discreta),
- NoOfSelfRef (discreta),
- LineOfCode (discreta),
- NoOfJS (discreta),
- SpacialCharRatioInURL (discreta),
- HasCopyrightInfo (booleana),
- IsResponsive (booleana),

- HasDescription (booleana),
- HasSocialNet (booleana),
- IsHTTPS (booleana),
- HasTitle (booleana).

Poiché si tratta di feature di tipo misto, tra variabili discrete, continue e binarie, è necessario effettuare una normalizzazione dei dati, in quanto le SVM ne beneficiano in maniera significativa, evitando che una variabile domini le altre a causa della sua scala.

La funzione di normalizzazione è stata dunque applicata alle variabili di tipo continuo e discreto sopra citate, utilizzando il metodo Min-Max Scaling, ottenendo valori compresi tra 0 e 1, combinati successivamente con le variabili binarie.

## 3.2 Creazione del modello

Per utilizzare il modello sono state usate le librerie *caret* e *e1071*<sup>2</sup>, che forniscono strumenti per preparare i dati, dividere i dataframe in training e testing e l'implementazione della SVM, che dunque non necessiterà di scrivere codice. Il dataset è stato dunque suddiviso in 80% training e 20% testing e la SVM è stata prima addestrata e poi testata. La SVM necessita di un kernel che determina come vengono mappati i dati nello spazio. Solitamente si possono usare kernel lineare e radiale, a seconda di come possono essere separati i dati nello spazio. Nel nostro caso, utilizzeremo il kernel lineare, il più semplice e leggero, e, se non dovesse risultare efficace, opteremo per quello radiale.

Dopo la fase di testing, valutiamo il modello, analizzando la matrice di confusione e l'accuracy:

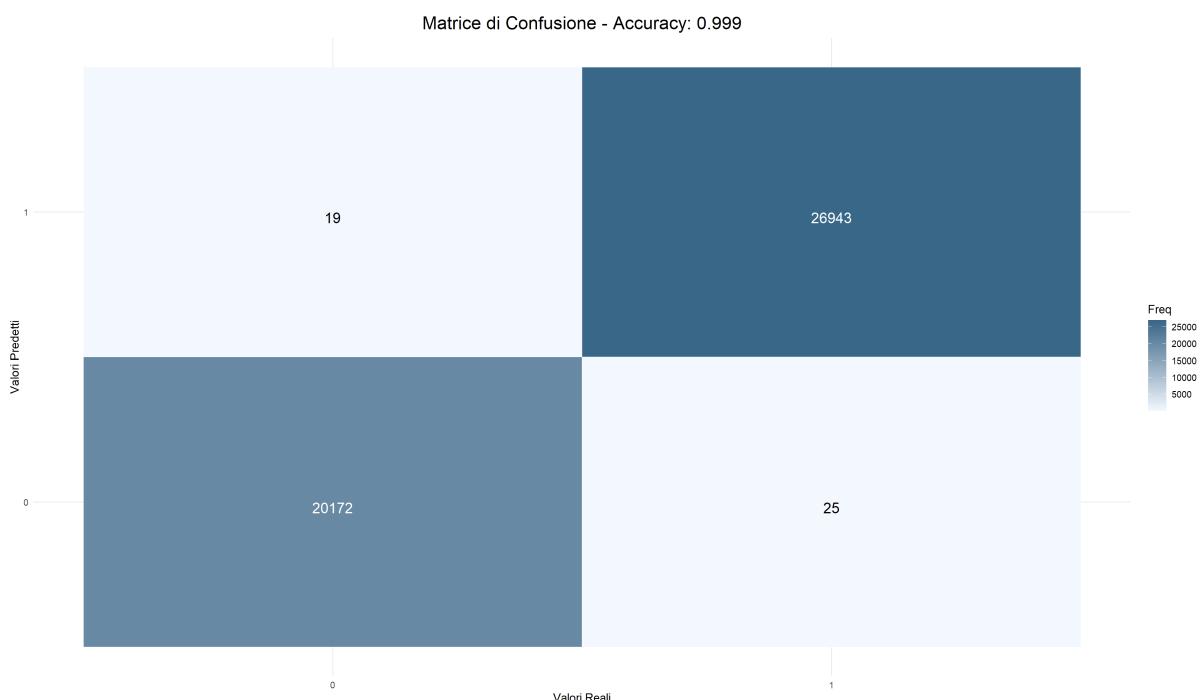


Figura 66: Matrice di confusione e accuracy

<sup>2</sup>[http://www.r-project.it/\\_book/support-vector-machines-svm.html](http://www.r-project.it/_book/support-vector-machines-svm.html)

Come si può notare, l'accuracy è piuttosto alta, quindi il modello riesce a stabilire con precisione se un sito web è legittimo o meno.

### 3.3 Visualizzazione dei dati

Per poter mostrare i dati su un grafico è necessario fare un ulteriore passaggio, ovvero effettuare una riduzione della dimensionalità tramite PCA<sup>3</sup>: in presenza di dataset con molte variabili (o feature), la PCA consente di ridurre il numero di dimensioni mantenendo quanta più informazione possibile. Nel nostro caso è utile per la visualizzazione dei dati, ovvero per proiettare dati ad alta dimensionalità su due componenti principali, consentendo di creare grafici (come scatter plot) per esplorarne i raggruppamenti e la classificazione dei dati. Per prima cosa sono state estratte casualmente 500 entry dal subset di testing, per mostrare in maniera grafica i punti. Su questi dati è stata eseguita la PCA tramite la funzione *prcomp()* che ne restituisce le componenti principali. Quelle che utilizzeremo saranno le prime due, denominate PC1 e PC2 in quanto sono quelle tipicamente più utilizzate per questo tipo di analisi per visualizzare in due dimensioni i dati. Inoltre, queste due componenti catturano la maggior parte della varianza nei dati, offrendo una buona approssimazione della struttura sottostante.

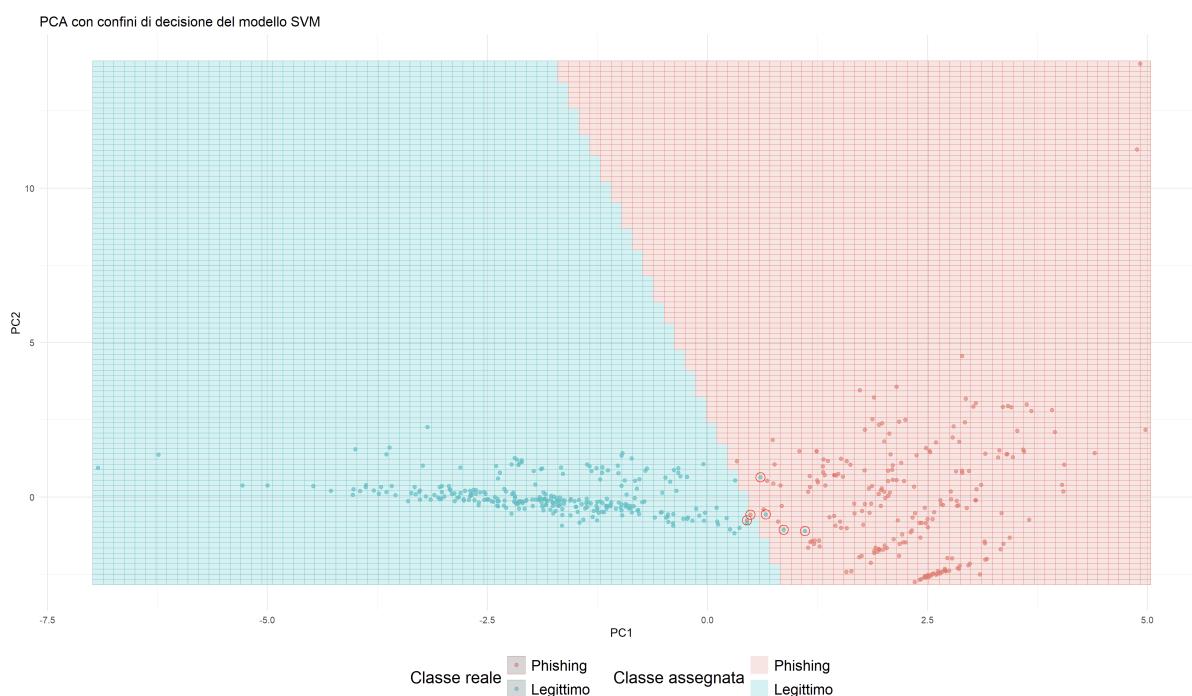


Figura 67: Grafico della SVM su un campione di testing di 500 entry. Le classificazioni sbagliate sono state evidenziate con dei cerchi rossi.

Come si può vedere dal grafico, i dati sono ben differenziati dal margine, con sole 6 entry classificate erroneamente.

Estrapolando questi dati, otteniamo le seguenti entry:

<sup>3</sup>[http://www.r-project.it/\\_book/analisi-delle-componenti-principali-pca-e-analisi-fattoriale-esplorativa-efa.html#pca](http://www.r-project.it/_book/analisi-delle-componenti-principali-pca-e-analisi-fattoriale-esplorativa-efa.html#pca)

<b>Feature</b>	<b>Ph1</b>	<b>Ph2</b>	<b>Legit1</b>	<b>Legit2</b>	<b>Legit3</b>	<b>Legit4</b>
<b>URLLength</b>	34	34	32	29	20	43
<b>NoOfSubDomain</b>	0	1	1	1	1	0
<b>CharContinuationRate</b>	1.00	1.00	1.00	0.43	1.00	1.00
<b>URLCharProb</b>	0.056	0.065	0.064	0.062	0.049	0.059
<b>NoOfOtherSpecialCharsInURL</b>	3	1	1	3	1	2
<b>LineOfCode</b>	233	219	385	165	280	349
<b>NoOfImage</b>	1	19	3	6	18	5
<b>NoOfSelfRef</b>	12	86	1	1	30	0
<b>NoOfJS</b>	13	1	2	1	1	9
<b>HasCopyrightInfo</b>	0	0	0	1	0	0
<b>IsResponsive</b>	1	0	0	0	1	1
<b>HasDescription</b>	0	0	0	1	0	0
<b>HasSocialNet</b>	0	0	0	0	0	0
<b>IsHTTPS</b>	1	1	1	1	1	1
<b>HasTitle</b>	1	1	1	1	1	1

Table 22: Entry contrassegnate in maniera errata. Il nome della entry corrisponde alla classe assegnata dal modello.

Le statistiche generali delle entry legittime e di phishing sono le seguenti:

Table 23: Statistiche descrittive delle principali feature per URL Legittimi e Phishing

Feature	Legittimi	Phishing
<b>URLLength_mean</b>	26.22861	45.72029
<b>URLLength_median</b>	26	34
<b>URLLength_sd</b>	4.815612	61.14552
<b>NoOfSubDomain_mean</b>	1.161661	1.168894
<b>NoOfSubDomain_median</b>	1	1
<b>NoOfSubDomain_sd</b>	0.4040759	0.7908797
<b>CharContinuationRate_mean</b>	0.9331759	0.7283951
<b>CharContinuationRate_median</b>	1	0.7272727
<b>CharContinuationRate_sd</b>	0.1398183	0.243976
<b>URLCharProb_mean</b>	0.0600501	0.04999915
<b>URLCharProb_median</b>	0.06062796	0.05202269
<b>URLCharProb_sd</b>	0.007170559	0.01163442
<b>NoOfOtherSpecialCharsInURL_mean</b>	1.244835	3.803467
<b>NoOfOtherSpecialCharsInURL_median</b>	1	3
<b>NoOfOtherSpecialCharsInURL_sd</b>	0.5039612	4.998448
<b>LineOfCode_mean</b>	1947.492	65.73047
<b>LineOfCode_median</b>	1105	12
<b>LineOfCode_sd</b>	4348.195	195.9181
<b>NoOfImage_mean</b>	44.9466	0.8664917
<b>NoOfImage_median</b>	25	0
<b>NoOfImage_sd</b>	100.9305	3.296497
<b>NoOfSelfRef_mean</b>	113.4102	0.4960325
<b>NoOfSelfRef_median</b>	76	0
<b>NoOfSelfRef_sd</b>	221.6353	3.168894
<b>NoOfJS_mean</b>	17.73253	0.8903363
<b>NoOfJS_median</b>	14	0
<b>NoOfJS_sd</b>	27.21394	3.362975

Confrontando le varie feature si possono notare alcune feature che possono aver confuso il modello, in particolare la lunghezza degli URL, il numero di immagini, i riferimenti o le caratteristiche come la responsività e le informazioni. Queste entry dunque sono probabilmente più difficili da classificare a causa delle caratteristiche in comune con le entry dell'altra classe.

## 4 LLM e dati sintetici

Nel contesto odierno, l'adozione dei Large Language Models (LLM), come quelli di OpenAI o Google, rappresenta una delle innovazioni più significative nell'ambito dell'intelligenza artificiale. Questi modelli, addestrati su vasti insiemi di dati testuali, si sono rapidamente evoluti per supportare una varietà di applicazioni pratiche, tra cui la generazione di testo, la comprensione del linguaggio naturale e, più recentemente, la generazione di dati sintetici. L'utilizzo dei dati sintetici è sempre maggiore in settori come la sicurezza dei dati, la ricerca scientifica e le analisi statistiche avanzate, per addestrare modelli senza utilizzare dati autentici, tutelando la privacy quando si lavora con dati sensibili; o in generale nei casi in cui i dataset reali non sono abbastanza grandi da permettere un buon addestramento, consentendo di superare le limitazioni imposte da dataset reali incompleti o sbilanciati. In questo contesto, gli LLM possono essere utilizzati per simulare dati realistici, mantenendo le proprietà statistiche e strutturali dei dataset originali.

### 4.1 Obiettivo

Nel nostro caso gli LLM possono essere impiegati per la creazione di dataset sintetici che contengono dati di siti web legittimi e di phishing. Partendo dal dataset originale è possibile generare nuovi dati che riproducono fedelmente le distribuzioni e le relazioni presenti nei dati reali.

Innanzitutto, dobbiamo stabilire delle *research question* per le nostre analisi. Dato il tipo di dataset, i nostri obiettivi saranno:

1. **RQ1:** I dati sintetici generati dagli LLM per il dataset di URL mantengono le stesse proprietà statistiche dei dati reali?
2. **RQ2:** Qual è l'impatto delle anomalie presenti nei dati reali sulla qualità dei dati sintetici generati dagli LLM?

### 4.2 LLM utilizzato e prompt

Per questa analisi è stato impiegato il modello *ChatGPT 4o* di *OpenAI*, a cui è stato chiesto di generare un dataset di dati sintetici in formato .csv di 100000 entry a partire da quello originale. Le modalità di prompting sono due:

#### 1) Prompt senza contesto

**Prompt senza contesto:** "Basandoti su questo dataset genera dei dati sintetici restituendoli in un file csv con 100000 entry"

## 2) Prompt con contesto

"Sto analizzando un dataset reale di URL composto da 100945 URL di phishing e 134850 URL legittimi. Ogni record presenta numerose feature che descrivono le proprietà dell'URL e della relativa pagina web. Tra queste feature ci sono:

Lunghezza dell'URL (URLLength),

Numero di domini e sottodomini,

Numero di caratteri e presenza di cifre,

Altre caratteristiche della pagina web (ad esempio, presenza di immagini, codice CSS, ecc.).

Inoltre, il dataset include una colonna 'label' dove il valore 0 indica un URL di phishing e il valore 1 un URL legittimo.

Il tuo compito è generare un dataset sintetico in formato CSV contenente 100000 record che rispecchi fedelmente le proprietà e le distribuzioni del dataset originale. In particolare, il dataset sintetico deve: 1) Contenere tutte le colonne presenti nel dataset originale 2) Riprodurre le distribuzioni delle feature: Ad esempio gli URL di phishing presentano lunghezze più estreme e meno concentrate rispetto a quelli legittimi, oppure fanno uso di più sottodomini o di meno codice css.

3) Preservare la coerenza tra le feature: I valori generati devono essere compatibili con il tipo di dato di ciascuna colonna (numerico, categorico, ecc.) e le relazioni/interazioni tra le feature devono essere realistiche. Inoltre devono essere realistici, non dovrebbero esserci valori negativi per il numero di immagini o per la lunghezza degli URL.

4) Mantenere le differenze tra le classi: Il dataset sintetico deve quindi riflettere le evidenti differenze tra URL di phishing e URL legittimi, così da poter essere utilizzato per addestrare e testare modelli di machine learning in maniera affidabile.

Genera il dataset sintetico in formato CSV, assicurandoti che tutte le entry abbiano valori coerenti e compatibili con le feature originali."

### 4.2.1 Lunghezza dell'URL e del dominio

Analizzando innanzitutto la proporzione tra URL legittimi e di phishing, otteniamo i seguenti risultati:

Dataset	N# URL	Percentuale phishing (%)	Percentuale legittimi (%)
Dataset Originale	235795	42,81	57,19
Synthetic Dataset	100000	42,83	57,17
Synthetic Context Dataset	100000	40,00	60,00

Table 24: Percentuali di URL di phishing e legittimi nei diversi dataset

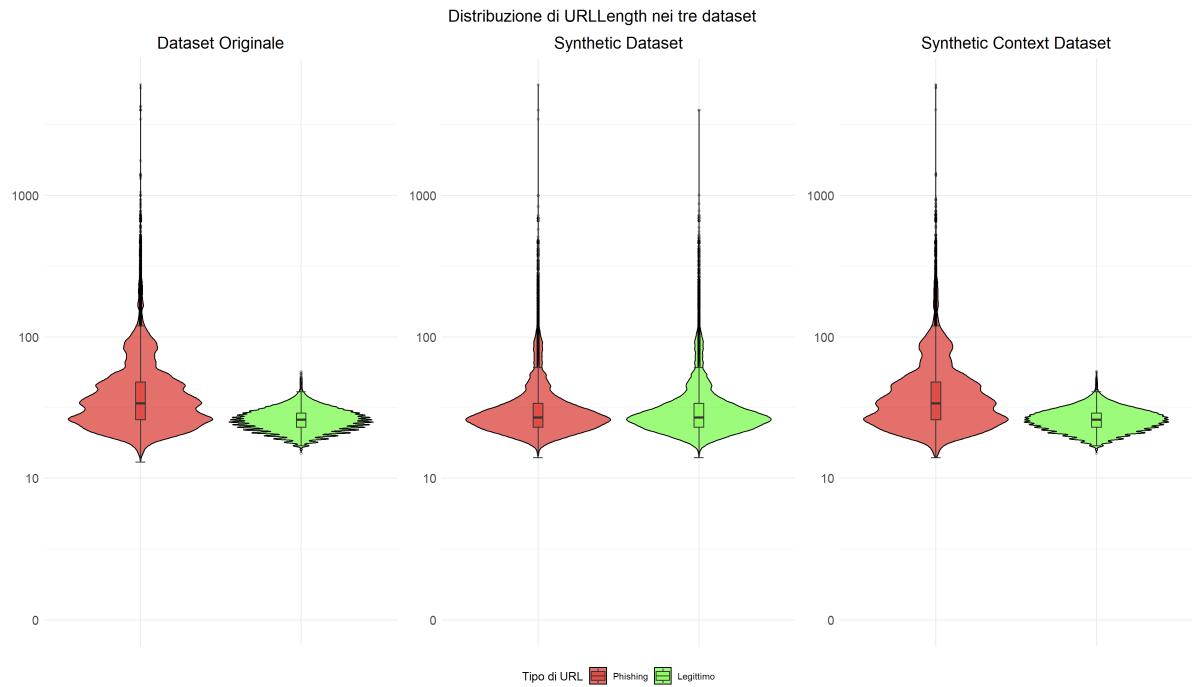


Figura 68: Confronto di URLLength tra il dataset originale e quelli sintetici.

Table 25: Statistiche per URLLength per dataset e categoria

<b>Dataset</b>	<b>Min</b>	<b>1° Quart.</b>	<b>Mediana</b>	<b>Media</b>	<b>3° Quart.</b>	<b>Max</b>	<b>Dev. st.</b>
<b>Legittimi</b>							
Dataset Originale	15	23	26	26,2	29	57	4,82
Synthetic Dataset	13	23	27	34,7	34	4274	43,9
Synthetic Context Dataset	15	23	26	26,2	29	57	4,79
<b>Phishing</b>							
Dataset Originale	13	26	34	45,7	48	6097	61,1
Synthetic Dataset	13	24	28	35,0	34	6097	51,0
Synthetic Context Dataset	14	26	34	45,8	48	6097	65,6

Per quanto riguarda la lunghezza degli URL, questa risulta più uniforme nel primo dataset sintetico (e in parte anche nel secondo), come si può notare dalla forma più affusolata dei violini nei grafici in Figura 68; il modello ha alterato la distribuzione della lunghezza, riducendo quella degli URL di phishing e aumentando significativamente quella degli URL legittimi, che risultano adesso avere massimi che superano i 4000 caratteri, contro i 57 del dataset originale. Inserire un contesto nel prompt ha invece aiutato a mantenere una buona coerenza con i dati originali.

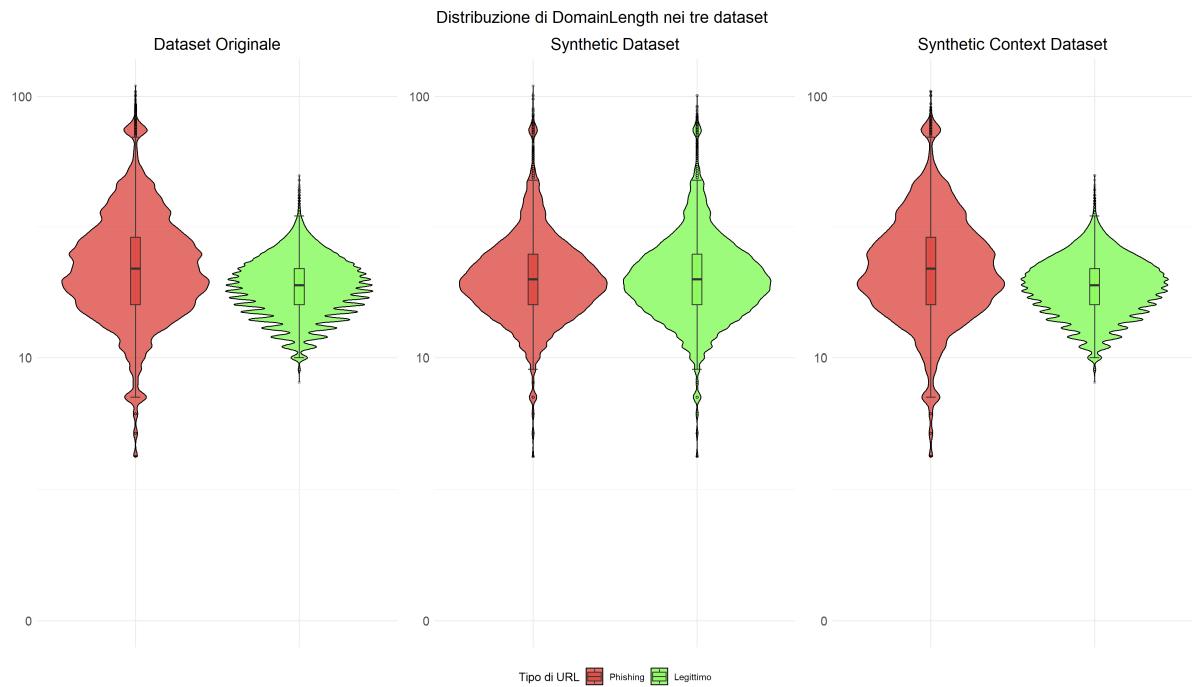


Figura 69: Confronto di DomainLength tra il dataset originale e quelli sintetici.

Table 26: Statistiche per DomainLength per dataset e categoria

<b>Dataset</b>	<b>Min</b>	<b>1 ° Quart.</b>	<b>Mediana</b>	<b>Media</b>	<b>3 ° Quart.</b>	<b>Max</b>	<b>Dev. st.</b>
<b>Legittimi</b>							
Dataset Originale	8	16	19	19.2	22	50	4.82
Synthetic Dataset	4	30	57	56.9	84	110	30.9
Synthetic Context Dataset	8	16	19	19,2	22	50	4.79
<b>Phishing</b>							
Dataset Originale	4	16	22	24.5	29	110	12.2
Synthetic Dataset	4	30	57	57.0	84	110	30.8
Synthetic Context Dataset	4	16	22	24,4	29	105	12,1

Anche per la lunghezza dei domini i dati sintetici creati senza contesto risultano più uniformi e ridotti nella distribuzione, mentre i valori estremi degli URL legittimi sono stati esasperati, cosa che non accade nel terzo dataset.

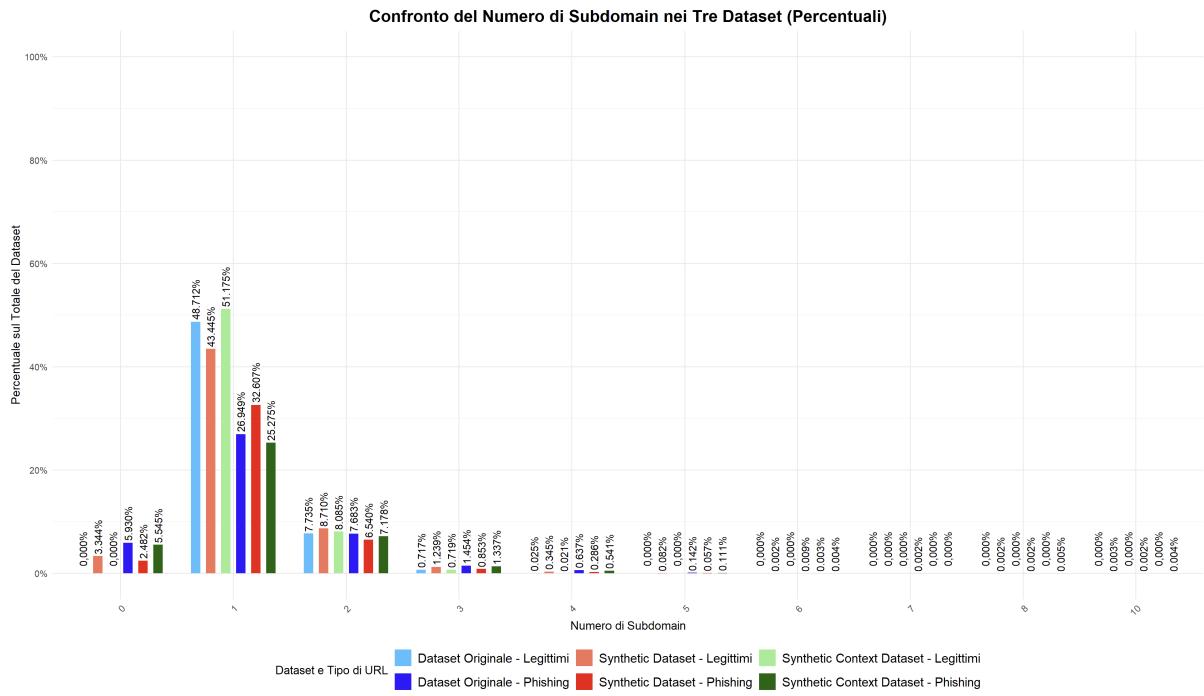


Figura 70: Confronto di NoOfSubDomain tra il dataset originale e quelli sintetici.

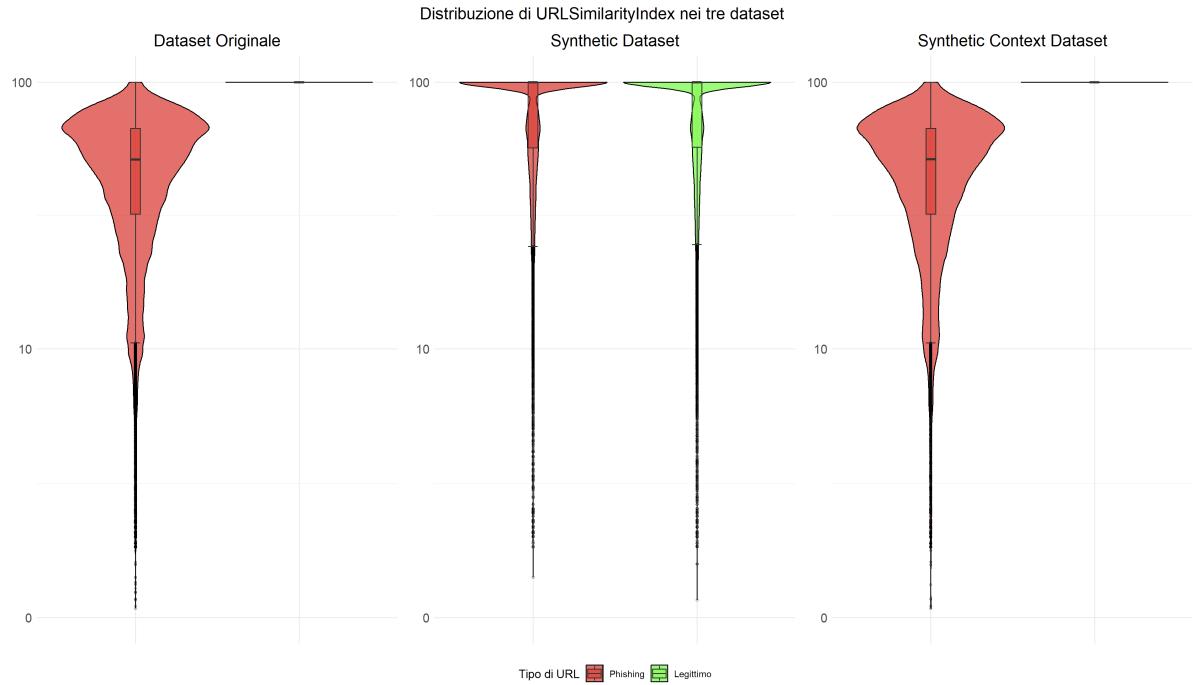


Figura 71: Confronto di URLSimilarityIndex tra il dataset originale e quelli sintetici.

Da queste prime analisi risulta chiaro che, sebbene il modello sia riuscito a generare dei dati più o meno coerenti con quelli originali anche senza contesto, non è riuscito a rispettare le distribuzioni e caratteristiche che differenziano gli URL legittimi da quelli di phishing. Il modello si è infatti limitato ad analizzare le distribuzioni totali del dataset, replicandole. Il risultato è dunque un'equa distribuzione tra le due categorie di URL, creando una simmetria tra i dati. Al contrario, fornendo del contesto al prompt, il modello è riuscito a differenziare le due categorie

di URL.

#### 4.2.2 Confronto dei TLD

Come abbiamo visto, nel dataset originale i TLD presentano due anomalie: una dovuta alla presenza di URL formati da indirizzi IP, che causa la registrazione di TLD composti dall'ultimo blocco di cifre dell'indirizzo IP; mentre la seconda è causata dalla presenza di porte di rete alla fine dell'URL, registrate insieme al TLD come un TLD a parte. Per analizzare al meglio il dataset è stata effettuata un'operazione di data cleaning; in questo caso però, verrà analizzata la presenza di sudette anomalie oltre alla distribuzione dei TLD.

Table 27: Statistiche per i tre dataset senza data cleaning

Numero di TLD	Dataset	IP_Percent	Port_Percent
Dataset Originale	695	0.26	0.01
Synthetic Dataset	695	0.25	2.37
Synthetic Context Dataset	523	0.23	0.01

Applicando le stesse espressioni regolari utilizzate per rilevare la presenza di indirizzi IP e porte di rete nel dataset originale, tali valori risultano effettivamente anche nei dataset sintetici. Il dataset sintetico ottenuto senza contesto nel prompt presenta un valore decisamente più alto di porte di rete nell'URL.

Dataset	1	2	3	4	5
Dataset Originale	com	org	net	app	uk
Synthetic Dataset	golf	pg	br	scot	12
Synthetic Context Dataset	com	org	net	uk	app

Table 28: Top 5 TLD per ciascun dataset (PRE pulizia)

Provando a pulire i dati, i TLD si riducono rispettivamente a 569, 683, 463, ma risultano ancora delle anomalie nel primo dataset sintetico:

Dataset	1	2	3	4	5
Dataset Originale	com	org	net	app	uk
Synthetic Dataset	com	211	cc	240	pl
Synthetic Context Dataset	com	org	net	uk	app

Table 29: Top 5 TLD per ciascun dataset (POST pulizia)

Il motivo sembra chiaro: il modello non ha effettivamente rispettato la correlazione tra la colonna TLD e l'effettivo TLD dell'URL, ma ha inserito dei valori nella colonna TLD sulla sola base delle distribuzioni presenti. Questi TLD riescono a passare i test delle regex, perché la colonna contiene effettivamente il valore numerico ma l'URL è probabilmente normale e le regex filtrano interi indirizzi IP e i numeri dopo il carattere ':'.

Controllando alcuni valori con TLD numerici nei tre dataset, otteniamo quanto segue:

Table 30: Confronto di URL e TLD per ciascun dataset

Dataset	URL	TLD assegnato	TLD in URL
Dataset originale	http://43.156.7.24/servicelogin?passive=1209600&continue=https://accounts.google.com/?&xrealip=107.178.232.255&followup=https://accounts.google.com/?	24	24
Synthetic Dataset	http://bafybeiboyzxtpt5st6yyhmxygbmzhvyoimyieuyjmmxcls4lmzsk5tntc34.ipfs.dweb.link/	24	link
Synthetic Dataset Context	http://43.156.7.24/v3/signin/identifier?dsh=s1661794781%3a1680120069260683&continue=https%3a%2f%2faccounts.google.com%2f%3f&ifkv=aqmjq7she90pcrfea7bxnzg7xfle8nzkgyvwffjdwp9stbsus9glbxk5cukna0fn6ktx9s3lmwa&xrealip=107.21.207.171&flowname=weblitesignin&flowentry=servicelogin	24	24

Come previsto, senza un contesto, il modello ha generato colonne in maniera indipendente, assegnando TLD per distribuzione delle frequenze della colonna e non per l'effettiva presenza nell'URL. Il secondo dataset sembra invece rispettare questa correlazione, nonostante abbia creato un URL con lo stesso dominio di quello originale.

Controlliamo dunque le occorrenze in cui il TLD effettivo è uguale a quello assegnato:

Dataset	Stesso TLD	TLD diverso
Dataset Originale	235795	0
Synthetic Dataset	128	99872
Synthetic Context Dataset	100000	0

Table 31: Confronto tra Domain e TLD per ciascun dataset

Il dataset creato con contesto rispetta effettivamente la regola, a differenza del primo. Tuttavia, analizzando il numero di URL e domini uguali, risultano essere tutti dei duplicati e lo stesso vale per il primo dataset sintetico.

### 4.3 Punti deboli del modello

A questo punto, controllando le intere righe dei dataset, risulta chiaro che il modello abbia duplicato le righe del dataset originale; infatti, eseguendo una inner join, otteniamo 100000

righe in comune tra i due dataset, di cui 18531 duplicate.

Provando a modificare il prompt con contesto per evitare la replica totale del dataset, specificando la necessità di generare dati sintetici nuovi a partire da quelli reali non si ottengono risultati migliori: il prompt è stato modificato in:

Prompt

"Sto analizzando un dataset reale di URL composto da 100945 URL di phishing e 134850 URL legittimi. Ogni record presenta numerose feature che descrivono le proprietà dell'URL e della relativa pagina web. Tra queste feature ci sono:

Lunghezza dell'URL (URLLength) Numero di domini e sottodomini Numero di caratteri e presenza di cifre Altre caratteristiche della pagina web (ad esempio, presenza di immagini, codice CSS, ecc.) Inoltre, il dataset include una colonna 'label' dove il valore 0 indica un URL di phishing e il valore 1 un URL legittimo.

Il tuo compito è generare un dataset sintetico in formato CSV contenente 100000 record che rispecchi fedelmente le proprietà e le distribuzioni del dataset originale. In particolare, il dataset sintetico deve:

- 1) Contenere tutte le colonne presenti nel dataset originale
- 2) Riprodurre le distribuzioni delle feature:

Ad esempio gli URL di phishing presentano lunghezze più estreme e meno concentrate rispetto a quelli legittimi, oppure fanno uso di più sottodomini o di meno codice css. La colonna TLD deve rispecchiare l'effettivo TLD dell'URL inserito nella colonna URL.

3) Preservare la coerenza tra le feature:

I valori generati devono essere compatibili con il tipo di dato di ciascuna colonna (numerico, categorico, ecc.) e le relazioni/interazioni tra le feature devono essere realistiche. Inoltre devono essere realistici, non dovrebbero esserci valori negativi per il numero di immagini o per la lunghezza degli URL.

4) Mantenere le differenze tra le classi:

Il dataset sintetico deve quindi riflettere le evidenti differenze tra URL di phishing e URL legittimi, così da poter essere utilizzato per addestrare e testare modelli di machine learning in maniera affidabile. Per la creazione del dataset sintetico non replicare le righe del dataset originale: non devono esserci duplicati o righe in comune con il dataset originale. I dati sintetici devono rispettare il formato, le distribuzioni e le correlazioni dei dati originali, ma devono essere inferiti senza replicarli.

Genera il dataset sintetico in formato CSV, assicurandoti che tutte le entry abbiano valori coerenti e compatibili con le feature originali."

Nonostante le ripetute richieste di correzione, il modello ha prodotto un dataset incompleto, contenente solo alcune delle colonne necessarie. Anche dopo aver raggiunto il numero corretto di colonne, il risultato è rimasto insoddisfacente: i valori non corrispondono a quelli originali. Ad esempio, variabili originariamente binarie o discrete ora assumono valori continui; inoltre, tutti gli URL risultano identici e la colonna TLD è stata associata esclusivamente all'URL anziché al dominio, che a sua volta non è coerente con l'URL. Inoltre, usare chat pulite non migliora la situazione.

L'ultimo tentativo di correzione ha generato un dataset con valori coerenti con quelli originali ma che presentano una distribuzione alterata e nuovamente simmetrica tra URL legittimi e di phishing.

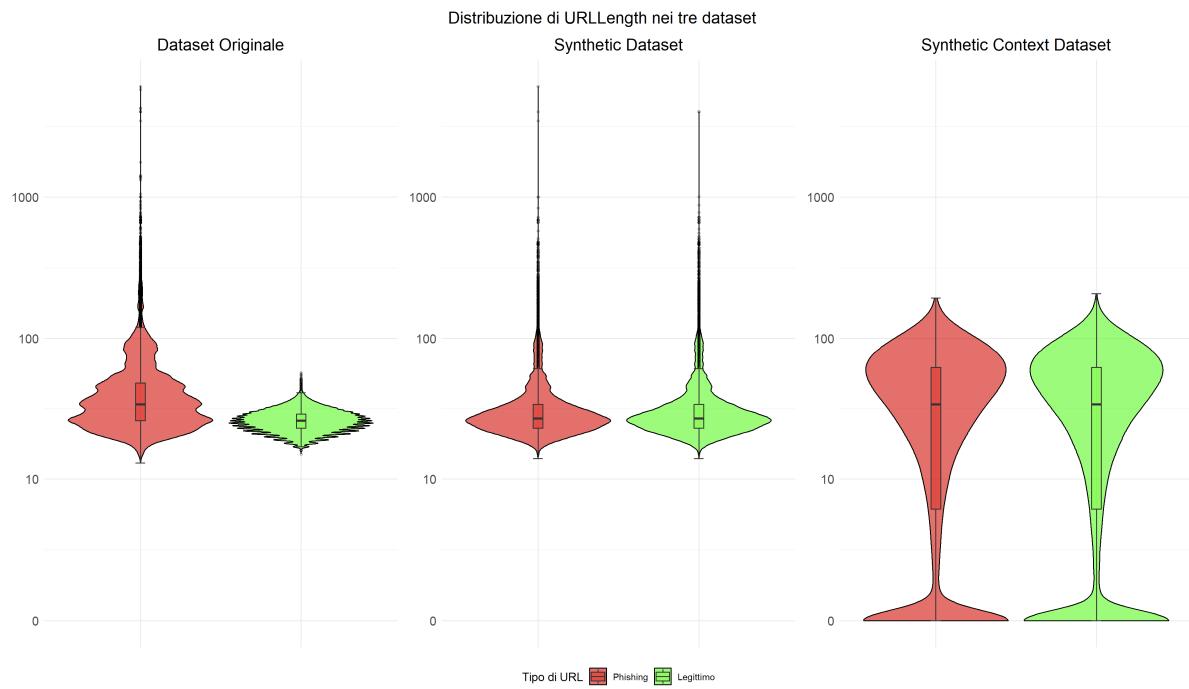


Figura 72: Confronto di URLLength tra il dataset originale e quelli sintetici.

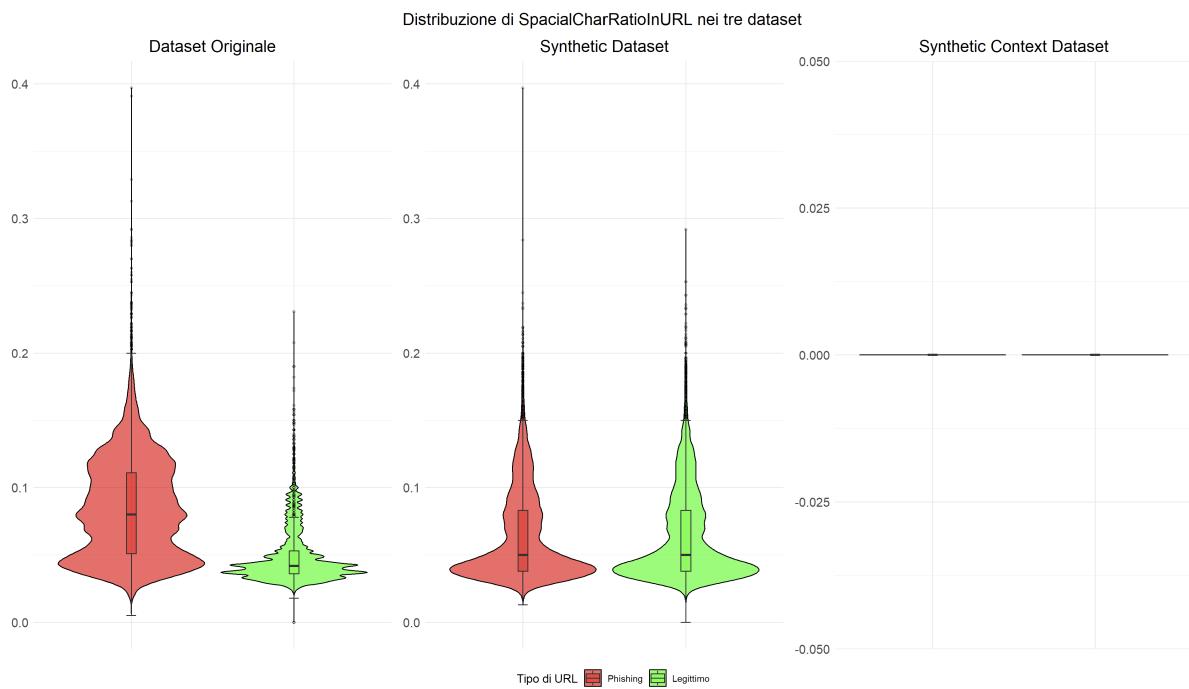


Figura 73: Confronto di SpacialCharRatioInURL tra il dataset originale e quelli sintetici.

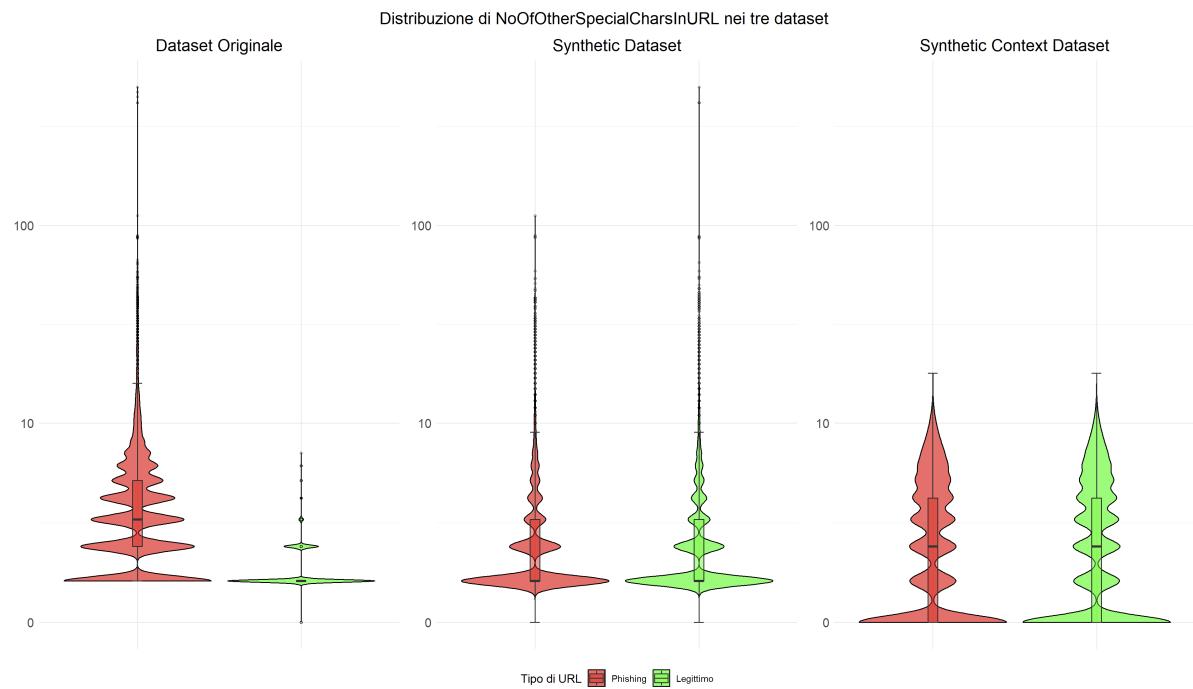


Figura 74: Confronto di NoOfOtherSpecialCharsInURL tra il dataset originale e quelli sintetici.

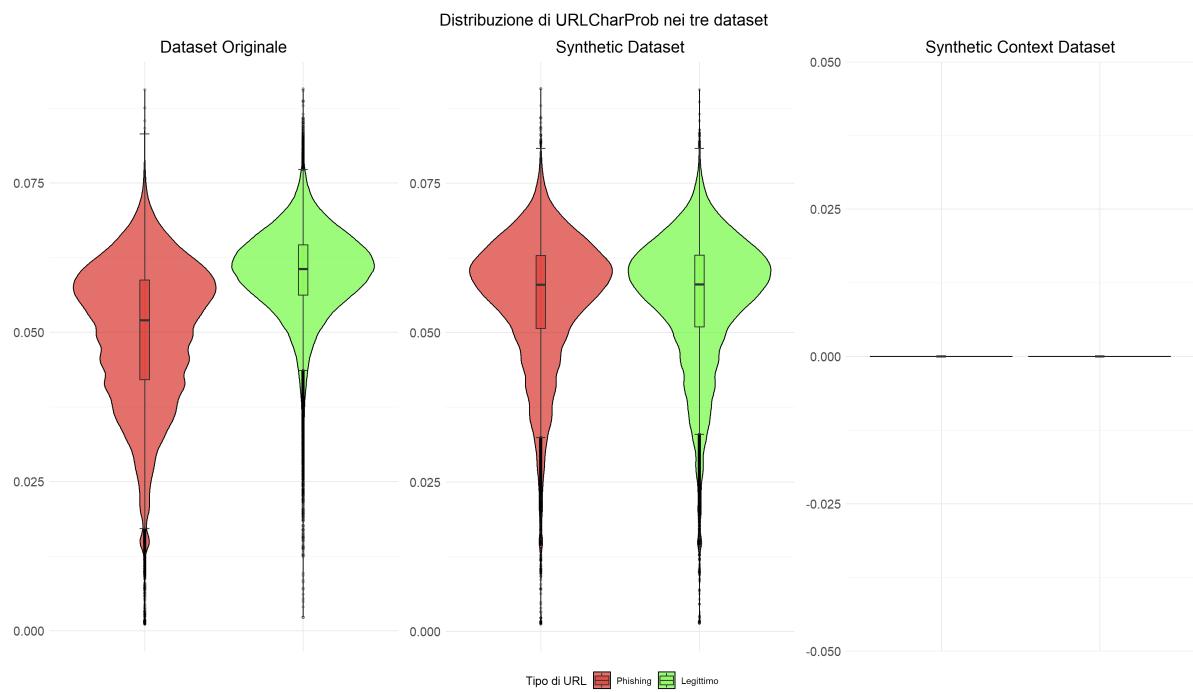


Figura 75: Confronto di URLCharProb tra il dataset originale e quelli sintetici.

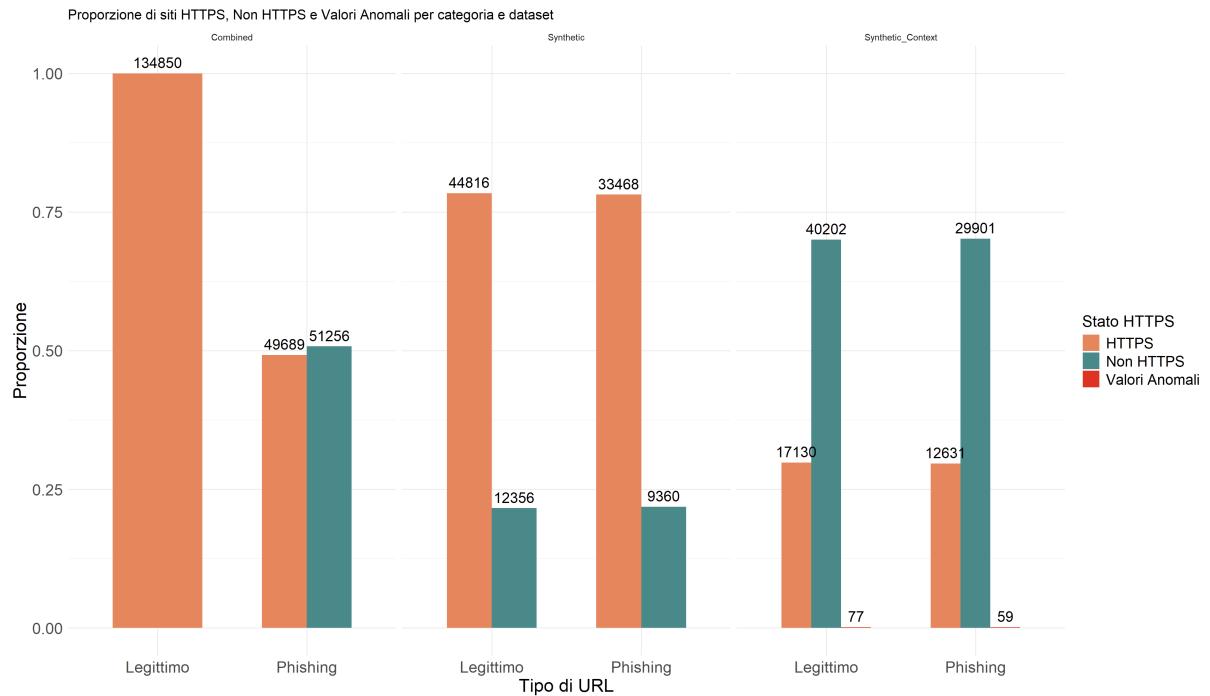


Figura 76: Confronto di IsHttps tra il dataset originale e quelli sintetici.

## 4.4 Conclusioni

Osservando dunque i risultati ottenuti, sembra che il modello *GPT-4o* abbia difficoltà nel generare dataset sintetici rispettando le correlazioni tra le diverse colonne. Per rispondere alla **RQ1**, è possibile generare dati sintetici in maniera indipendente, come dimostrato dal primo dataset sintetico, che mantengano le caratteristiche statistiche della singola colonna; tuttavia, non è altrettanto semplice mantenere le relazioni complesse che esistono tra le diverse feature. Questo implica che, pur generando valori coerenti a livello di singola colonna e rispettando le distribuzioni marginali, il dataset sintetico non riesce a preservare completamente le interazioni tra variabili multiple, come le correlazioni tra la lunghezza dell'URL, il numero di sottodomini e la presenza di caratteri speciali.

In particolare, si osserva che le colonne numeriche vengono generate con valori plausibili, ma senza considerare eventuali pattern o vincoli di dipendenza presenti nel dataset originale. Allo stesso modo, le feature categoriali, come il TLD, l'URL o il dominio, non mostrano la stessa coerenza strutturale che si potrebbe riscontrare nei dati reali. Da ciò si può concludere che la risposta alla **RQ2** è che le anomalie vengono appiattite e rese parte della normale distribuzione delle occorrenze, mescolandosi con i valori normali, alterandone la distribuzione. In alcuni casi, queste possono essere addirittura amplificate e rese parte delle feature, come accade per i TLD formati da parti di indirizzi IP.

Questo limite evidenzia come la generazione di dati sintetici più complessi con LLM richieda probabilmente prompt più efficaci, oltre che modelli più avanzati (si potrebbe ad esempio provare con *GPT-o1*).