

CDLM DATA SCIENCE, UNIVERSITÀ DEGLI STUDI DI MILANO BICOCCA,
CORSO DI DATA SCIENCE LAB, ANNO ACCADEMICO 2021-2022

Campaign Click

Progetto realizzato da:

Emanuela	Elli	(892901)
Armando	Epifani	(826153)
Gloria	Giorgetti	(826226)
Francesco	Gregori	(889206)
Federica	Madon	(825628)





Indice

1	Introduzione	2
2	Data Preparation	2
2.1	Analisi descrittiva dataset	3
2.2	Analisi descrittiva <i>suspicious</i>	6
3	Modello	8
3.1	Valutazione dei risultati	9
4	Conclusioni	10
	Sitografia e Bibliografia	11



1 Introduzione

In questo report viene presentata un'analisi di un dataset contenente dati provenienti da una campagna pubblicitaria online, **Campaign (click)**, al fine di predire il click dell'utente e di individuare a quali utenti inviare la pubblicità in futuro. Tale dataset è composto dalle variabili:

- **ad_form_id** → identificativo univoco per i cookie, si suppone che ogni cookie fa riferimento ad un utente diverso;
- **suspicious** → tale variabile assume valore 1 per indicare che il click sia un sospetto di truffa (ovvero che non sia una persona ad aver cliccato ma probabilmente un *bot*)
- **clicks** → variabile da prevedere, fa riferimento al numero di click;
- **impressions** → numero di pagine viste;
- **buy** → acquisti effettuati;
- **categories** → serie di variabili che fanno riferimento alle categorie semantiche ovvero la tipologia di pagina a cui è interessato l'utente, suddivise in tre livelli di aggregazione;
- **admans** → categorie semantiche ad hoc (l'algoritmo identifica delle categorie semantiche indipendenti dalle altre);
- **time** → identificano il momento in cui l'utente ha cliccato, suddivise in due gruppi, il primo distingue i giorni infrasettimanali da quelli feriali mentre il secondo considera l'intera settimana;
- **navigation** → percentuale di navigazione;
- **L** → serie di variabili che fanno riferimento alla lunghezza del testo della pagina (ad esempio L050 identifica le pagine con un testo lungo da 0 a 50 parole);
- **feelings** → identificano il “sentimento” della pagina (amore, odio, etc);
- **device_type** → identifica l'hardware, ovvero permette di riconoscere il device, tale variabile è importante perché ad esempio in un device piccolo è più facile che l'utente possa sbagliare e dunque cliccare;
- **os** → identificano il sistema operativo dell'utente;
- **browser** → identificano il browser utilizzato dall'utente.

2 Data Preparation

La fase iniziale di questo lavoro è stata dedicata alla preparazione dei dati. Il dataframe contiene 1416 colonne e 82564 righe. Data la dimensione del dataset si è deciso di ridurre il numero di colonne eliminando, innanzitutto, quelle che presentavano tutti i valori pari a 0 e quelle con informazioni ripetitive o ridondanti al fine di prevedere, come anticipato,



il click dell'utente. Un ulteriore controllo è stato effettuato anche sulle righe, eliminando quelle con tutti valori pari a 0 o nulli.

Sono, quindi, state mantenute le variabili relative alle **categories** al primo livello di aggregazione, quelle temporali che considerano la divisione tra giorni lavorativi e feriali e **device_type**.

Le variabili relative al sistema operativo e al browser sono invece state accorpate in due variabili:

os_type	browser_type	code
os_android	browser_android	0
os_bsd	browser_chrome	1
os_linux	browser_edge	2
os_osx	browser_firefox	3
os_windows	browser_ie	4
os_other	browser_opera	5
	browser_safari	6
	browser_other	7
	browser_unknown	8

Le colonne relative alla lunghezza invece sono state unite siccome presentavano molti valori pari a 0:

- L00_50 e L51_100 sono rimaste come le originali;
- L101_250 e L251_500 sono state unite nella colonna denominata L101_500;
- L501_1000, L1001_2500, L2501_5000, L5001_10000 e L10001_more sono state unite nella colonna denominata L501_more.

Si è poi deciso di creare una nuova colonna denominata **click_status** che tenga traccia, attraverso delle stringhe (“yes” o “no”), di quali utenti hanno effettuato il click oppure no (e di conseguenza è stata eliminata la variabile **clicks**).

Inoltre si è deciso di voler inizialmente lavorare solo con i dati relativi ad utenti con **suspicious=0**, ovvero gli utenti che certamente sono persone reali e non *bot*.

Per le variabili **categories**, quelle relative alle lunghezze e **time** è stata eseguita una normalizzazione in modo tale che sommassero a 100.

2.1 Analisi descrittiva dataset

Prima di procedere alla fase di modellizzazione e predizione dei click, si sono effettuate alcune statistiche descrittive. Il dataset ridotto è composto da un totale di 62397 record ognuno dei quali specifica un valore relativo a 42 variabili differenti. Inoltre sono stati verificati che non vi fossero più alcuni valori nulli per nessuna variabile del dataset.



Nell'immagine di seguito (Figura 1) è possibile vedere tutte le variabili presenti nel dataset e la rispettiva tipologia, mentre nella Figura 2 e Figura 3 vengono mostrate le statistiche descrittive di tutte le variabili numeriche presenti nel dataset.

```
Data columns (total 42 columns):
 #   Column           Non-Null Count Dtype  
 ---  -- 
 0   clicks          62397 non-null  int64   
 1   os_type          62397 non-null  category 
 2   browser_type     62397 non-null  category 
 3   device_type      62397 non-null  category 
 4   time1_workday_morning 62397 non-null  float64 
 5   time1_workday_afternoon 62397 non-null  float64 
 6   time1_workday_evening 62397 non-null  float64 
 7   time1_workday_night   62397 non-null  float64 
 8   time1_weekend_morning 62397 non-null  float64 
 9   time1_weekend_afternoon 62397 non-null  float64 
 10  time1_weekend_evening 62397 non-null  float64 
 11  time1_weekend_night   62397 non-null  float64 
 12  L00_50            62397 non-null  float64 
 13  L51_100           62397 non-null  float64 
 14  L101_500          62397 non-null  float64 
 15  L501_more          62397 non-null  float64 
 16  categories1_artandentertainment 62397 non-null  float64 
 17  categories1_automotive        62397 non-null  float64 
 18  categories1_business         62397 non-null  float64 
 19  categories1_careers          62397 non-null  float64 
 20  categories1_education        62397 non-null  float64 
 21  categories1_familyandparenting 62397 non-null  float64 
 22  categories1_finance          62397 non-null  float64 
 23  categories1_foodanddrink     62397 non-null  float64 
 24  categories1_healthandfitness 62397 non-null  float64 
 25  categories1_hobbiesandinterests 62397 non-null  float64 
 26  categories1_homeandgarden    62397 non-null  float64 
 27  categories1_intentions       62397 non-null  float64 
 28  categories1_lawgovtandpolitics 62397 non-null  float64 
 29  categories1_news            62397 non-null  float64 
 30  categories1_pets            62397 non-null  float64 
 31  categories1_realestate       62397 non-null  float64 
 32  categories1_religionandspirituality 62397 non-null  float64 
 33  categories1_science          62397 non-null  float64 
 34  categories1_shopping         62397 non-null  float64 
 35  categories1_society          62397 non-null  float64 
 36  categories1_sports           62397 non-null  float64 
 37  categories1_styleandfashion 62397 non-null  float64 
 38  categories1_technologyandcomputing 62397 non-null  float64 
 39  categories1_travel           62397 non-null  float64 
 40  categories1_uncategorized    62397 non-null  float64 
 41  click_status               62397 non-null  category 

dtypes: category(4), float64(37), int64(1)
```

Figura 1: Output del codice python per identificare la tipologia delle variabili presenti nel dataset.



	clicks	time1_workday_morning	time1_workday_afternoon	time1_workday_evening	time1_workday_night	time1_weekend_morning	time1_weekend_afternoon	time1_weekend_evening
count	62397	62397	62397	62397	62397	62397	62397	62397
mean	0,003862365	13,7526348	21,83422978	15,22388661				
std	0,094062902	28,39628371	32,65589621	29,23075816				
min	0	0	0	-33,33333333				
25%	0	0	0	0				
50%	0	0	0	0				
75%	0	9,090909091	36,36363636	16,66666667				
max	17	100	133,33333333	100				
	time1_weekend_night	time1_weekend_morning	time1_weekend_afternoon	time1_weekend_evening	time1_weekend_night	time1_weekend_morning	time1_weekend_afternoon	time1_weekend_evening
count	62397	62397	62397	62397	62397	62397	62397	62397
mean	8,540652713	10,50581665	15,65111508	8,05955362				
std	23,08734514	26,60280109	31,83629972	23,55418739				
min	0	0	0	0				
25%	0	0	0	0				
50%	0	0	0	0				
75%	0	0	10	0				
max	100	100	100	100				
	L00_50	L51_100	L101_500	L501_more	categories1_artandentertainment	categories1_automotive	categories1_business	categories1_careers
count	62397	62397	62397	62397	62397	62397	62397	62397
mean	6,432110756	43,74700209	31,47093239	20,44511174				
std	21,86294913	48,15236623	45,08605398	38,52999825				
min	0	0	0	-100				
25%	0	0	0	0				
50%	0	0	0	0				
75%	0	100	100	3,846153846				
max	100	100	200	100				
	categories1_education	categories1_familyandparenting	categories1_finance	categories1_careers	categories1_entertainment	categories1_automotive	categories1_business	categories1_technologyandcomputing
count	62397	62397	62397	62397	62397	62397	62397	62397
mean	1,865439093	0,480062079	0,2600685	0,573084692				
std	12,76561565	5,937320871	2,972489023	6,702740548				
min	0	0	0	0				
25%	0	0	0	0				
50%	0	0	0	0				
75%	0	0	0	0				
max	100	100	100	100				
	categories1_styleandfashion	categories1_travel	categories1_uncategorized	categories1_technologyandcomputing	categories1_entertainment	categories1_automotive	categories1_business	categories1_careers
count	62397	62397	62397	62397	62397	62397	62397	62397
mean	0,4873907	22,64750027	1,352598619	7,489422558				
std	5,289509887	31,91452231	10,32446784	23,07338631				
min	0	0	0	0				
25%	0	0	0	0				
50%	0	0	0	0				
75%	0	40	0	100,0499002				
max	100	100,0499002	100,035968	100,0499002				

Figura 2: Output del codice python per visualizzare le statistiche descrittive delle variabili numeriche presenti nel dataset.



categories1_foodanddrink		categories1_healthandfitness		categories1_hobbiesandinterests	
count	62397		62397		62397
mean	0,611543848		0,200219842		19,30538538
std	6,940070794		2,986992225		31,44345821
min	0		0		0
25%	0		0		0
50%	0		0		0
75%	0		0		60
max	100,0509165		100		100,0408092

categories1_homeandgarden		categories1_intentions		categories1_lawgovtandpolitics	
count	62397		62397		62397
mean	0,181895747		0,110889148		0,328670014
std	2,937332425		1,453861137		4,360599101
min	0		0		0
25%	0		0		0
50%	0		0		0
75%	0		0		0
max	100		100		100

categories1_news		categories1_pets		categories1_realestate	
count	62397		62397		62397
mean	0,528317952		0,579929111		0,048937282
std	6,249984385		4,655200456		1,667515269
min	0		0		0
25%	0		0		0
50%	0		0		0
75%	0		0		0
max	100		100		100

categories1_religionandspirituality		categories1_science		categories1_shopping	
count	62397		62397		62397
mean	0,021309204		2,463694014		0,090912437
std	1,108139305		11,44735172		1,764320386
min	0		0		0
25%	0		0		0
50%	0		0		0
75%	0		0		0
max	100		100		100

categories1_society		categories1_sports	
count	62397		62397
mean	0,312669676		7,032364847
std	4,132773385		21,12768586
min	0		0
25%	0		0
50%	0		0
75%	0		0
max			

Figura 3: Continuazione dell'output precedente.

2.2 Analisi descrittiva suspicious

Si è deciso inoltre, a completamento dell'analisi, di eseguire un'ulteriore indagine riguardo l'affidabilità o meno dell'algoritmo che assegna i valori alla variabile **suspicious**. L'analisi viene effettuata sullo stesso dataset utilizzato per il modello con la sola differenza di aver mantenuto la colonna **suspicious**.

Nel dataset sono presenti 134 record con **suspicious=1**. Sul totale dei record del dataset questo valore rappresenta l'1%.

Inizialmente si è analizzata la distribuzione dei valori di **suspicious** rispetto alle variabili categoriche, ed è emerso quanto riportato nei seguenti istogrammi.

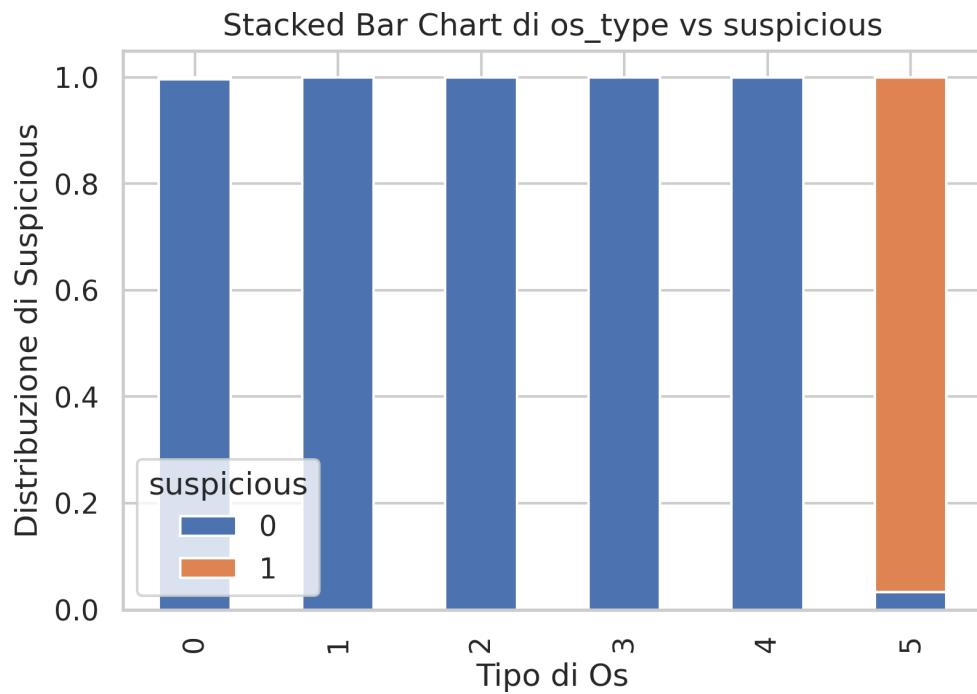


Figura 4: Distribuzione dei valori di **suspicious** rispetto la variabile **os_type**

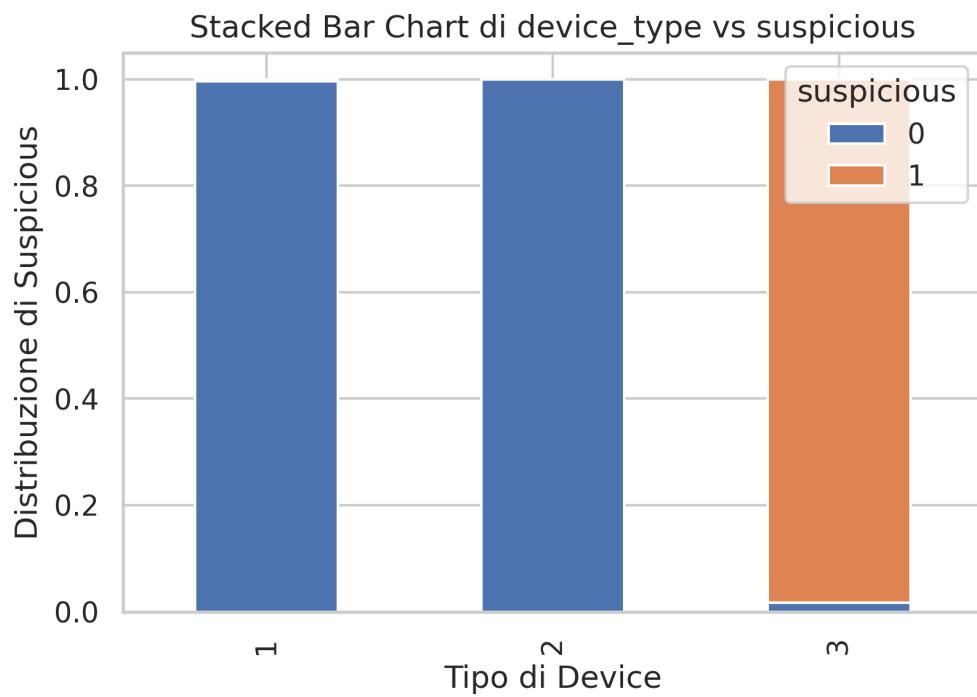


Figura 5: Distribuzione dei valori di **suspicious** rispetto la variabile **device_type**

Dai grafici emerge che i record con **suspicious=1** hanno per la maggior parte **os_type=5** (**os_other**) e **device_type=3** (**Desktop and Laptop**). Dopo questa iniziale analisi si è deciso di lavorare direttamente con i record con **suspicious=1** analizzando quindi un dataset di dimensioni molto inferiori.

Dall'analisi sul nuovo dataset risulta che:



- 98/134 record hanno valori di `ad_form_id` negativi;
- 131/134 record hanno `clicks=0`;
- 132/134 record con `impressions>clicks`;
- 119/134 record con `os_type=5`;
- 119/134 record con `device_type=3`;
- per quanto riguarda le categorie in media i record hanno un valore di 71.37 della variabile `categories1_technologyandcomputing`;
- per quanto riguarda le colonne relative alle lunghezze i record hanno in media un valore di 81.09 della variabile `L00_50`.

Le altre variabili non presentano nessuna prevalenza di valori.

3 Modello

Per identificare gli utenti a cui inviare la pubblicità, è stato sviluppato un modello di Machine Learning.

Innanzitutto, si è scelto di dare maggiore importanza alla corretta previsione dei “sì”, in modo tale da essere sicuri di riuscire ad individuare correttamente almeno la maggior parte degli utenti che cliccano sulla pubblicità, piuttosto che a quella dei “no”, in quanto è stato ritenuto non dannoso inviare una pubblicità a chi non è disposto a cliccare. Per poter sviluppare un modello che rispondesse a queste caratteristiche, il dataset è stato diviso in train e test set, contenenti rispettivamente il 70% e il 30% dei dati, ed è stato applicato un random over sampling sui dati del train set: ripetendo casualmente più volte gli utenti della classe “sì”, si è fatto in modo che il numero di “sì” fosse identico al numero di “no”. Il train set così aumentato, è stato quindi utilizzato per addestrare una SVM. In precedenza, sono stati valutati degli algoritmi alternativi a quello scelto per risolvere il problema dello sbilanciamento del dataset: in particolare, il random under sampling, che campiona i “no” in modo casuale e in modo tale che diventino in numero uguali ai “sì”, e SMOTE-NC, che genera sinteticamente nuovi dati sulla base dei dati di partenza. I nuovi dataset sono stati utilizzati per addestrare delle SVM e delle Random Forest in modo da poterne confrontare i risultati. I migliori, come anticipato, sono stati ottenuti applicando il random over sampling e utilizzando una SVM, che ha prodotto, sul test set, i risultati riportati nella Figura 6. Visto lo sbilanciamento del test set, si è deciso di non riportare i valori assoluti, ma quelli percentuali. Sempre per questo motivo, al posto dell’accuracy, per verificare che il modello non stesse incorrendo in overfitting, sono stati confrontati questi valori con i seguenti, ottenuti sul train set:

- no classificati come no: 59%
- no classificati come sì: 41%
- sì classificati come no: 30%
- sì classificati come sì: 70%



Si può notare come questi siano simili ai valori ottenuti sul test set, dimostrando come il modello non stia incorrendo in overfitting.

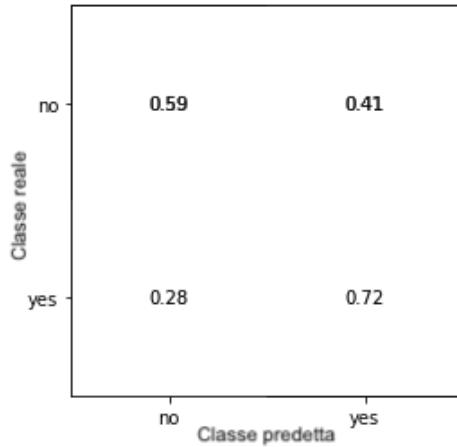


Figura 6: Matrice di confusione sul test set. Le percentuali fanno riferimento rispettivamente alla frazione di “no” classificati come “no” (59%), di “no” classificati come “sì” (41%), di “sì” classificati come “no” (28%) e di “sì” classificati come “sì” (72%).

3.1 Valutazione dei risultati

Per meglio valutare le performance del modello, si è deciso di utilizzare la decision function, determinata direttamente dall’algoritmo, che rappresenta la distanza di ogni predizione dall’iperpiano che separa la regione degli utenti classificati come “no”, da quella degli utenti classificati come “sì”. La decision function rappresenta quindi una misura della confidenza di ogni predizione. L’algoritmo, in particolare, calcola i valori di decision function associati alla classe “sì”: tali valori sono compresi tra -1 e 1 e un valore positivo indica che la classe predetta per la particolare istanza è “sì”, mentre un valore negativo indica che la classe predetta è “no”. Per poterli trattare più semplicemente, i valori sono stati riscalati tra 0 e 1 definendo la funzione $\frac{1}{1+e^{-c}}$, dove c rappresenta il particolare valore di decision function. Così facendo, la previsione della classe “sì” è associata a un valore di questa nuova funzione maggiore di 0.5, mentre la previsione della classe “no” è associata a un valore minore di 0.5. È importante notare come i valori ottenuti non rappresentino una probabilità, ma indicano quanto si è confidenti, su una scala da 0 a 1, che l’utente venga classificato come “sì”.

Una volta ottenuti questi valori è stato possibile effettuare la valutazione della performance. Gli utenti sono stati suddivisi prima in due, poi in quattro e infine in otto gruppi, sulla base rispettivamente della mediana, dei quartili e degli ottili. Dalle tabelle 1, 2 e 3 è possibile notare come la maggior parte degli utenti con classe vera "sì" cada nei gruppi a cui è associata una confidenza più alta, ad eccezione del gruppo costruito a partire dagli ottili $q_{0.125}$ e q_0 , dimostrando la bontà del modello. In queste tabelle è riportato anche il numero di utenti con classe vera "no" in ogni gruppo: si può notare come questi siano suddivisi più o meno equamente in ogni gruppo.



Limite inferiore	Limite superiore	Numero di "sì"	Numero di "no"
0.48	1	42	9315
0	0.48	11	9346

Tabella 1: Numero di utenti con classe vera “sì” e “no” nei gruppi costruiti a partire dalla mediana della confidenza (0.48).

Limite inferiore	Limite superiore	Numero di "sì"	Numero di "no"
0.54	1	27	4656
0.48	0.54	15	4659
0.42	0.48	4	4674
0	0.42	7	4672

Tabella 2: Numero di utenti con classe vera “sì” e “no” nei gruppi costruiti a partire dai quartili della confidenza ($q_{0.75} = 0.54$, $q_{0.5} = 0.48$, $q_{0.25} = 0.41$).

Limite inferiore	Limite superiore	Numero di "sì"	Numero di "no"
0.57	1	14	2427
0.54	0.57	13	2329
0.51	0.54	8	2329
0.48	0.51	7	2330
0.44	0.48	2	2361
0.42	0.44	2	2313
0.35	0.42	2	2337
0	0.35	5	2335

Tabella 3: Numero di utenti con classe vera “sì” e “no” nei gruppi costruiti a partire dagli ottimi della confidenza ($q_{0.875} = 0.57$, $q_{0.75} = 0.54$, $q_{0.625} = 0.51$, $q_{0.5} = 0.48$, $q_{0.375} = 0.44$, $q_{0.25} = 0.41$, $q_{0.125} = 0.35$).

4 Conclusioni

Lo scopo del lavoro è predire il click di un utente di una campagna pubblicitaria. Dopo una prima fase di data preparation, che ha portato ad una riduzione della dimensionalità del dataset, si è effettuata un’analisi descrittiva.

Per predire il click si è utilizzato un random over sampling, per risolvere lo sbilanciamento presente nei dati, e una SVM per l’effettiva classificazione, dando maggiore importanza alla corretta previsione dei “sì”, in modo tale da essere sicuri di riuscire ad individuare correttamente almeno la maggior parte degli utenti che cliccano sulla pubblicità, piuttosto che a quella dei “no”, in quanto è stato ritenuto non dannoso inviare una pubblicità a chi non è disposto a cliccare.

Con i dati a disposizione riguardo gli utenti che cliccano, per scegliere a chi inviare la pubblicità si possono considerare le soglie individuate nella valutazione dei risultati, te-



nendo conto che inviare una pubblicità agli utenti con alta probabilità di effettuare un click comporta l'invio anche ad utenti che non lo effettueranno.

Sitografia e Bibliografia

- [1] Seaborn
- [2] Oversampling and under sampling methods for imbalanced classification
- [3] Sklearn
- [4] Marco Fattore, Metodi di riduzione della dimensionalità