



UNICUSANO

Università degli Studi Niccolò Cusano - Telematica Roma

Master in Data Analyst

ACQUISIZIONE, TRASFORMAZIONE E VISUALIZZAZIONE DI DATI WEB DA GOOGLE ANALYTICS

Candidato
Domenico Rossi

Relatore
prof. Daniele di Giorgio

ANNO ACCADEMICO 2020/2021

Ringraziamenti

Vorrei ringraziare il professore Daniele Di Giorgio che ha accettato di seguirmi durante la stesura di questo lavoro di tesi.

Un enorme grazie va a Maria Rosaria, Alex e a tutti gli amici di **WeLabo** per avermi fornito informazioni senza le quali questo lavoro sarebbe rimasto incompleto.

Infine grazie a mia moglie, che crede sempre in me.

Indice

Introduzione	1
1 Raccolta, organizzazione e visualizzazione dei dati	5
1.1 Raccolta dei dati: Google Analytics	5
1.1.1 Perché raccogliere i dati del traffico verso un sito web?	7
1.1.2 Quali informazioni si ottengono da GA?	8
1.1.3 Le metriche di GA	11
1.1.4 Le conversioni	11
1.1.5 Come utilizzare i dati acquisiti da GA?	12
1.2 Organizzazione dei dati: Microsoft Excel	13
1.2.1 Scopo e generalità di utilizzo	13
1.2.2 Perché importare i dati in Excel?	15
1.3 Visualizzazione dei dati: Python	16
1.3.1 Storia e caratteristiche	16
1.3.2 Principali editor di testo per Python	17
1.3.3 I pacchetti <i>pandas</i> e <i>seaborn</i>	19
2 I software utilizzati	22
2.1 Google Analytics	22
2.1.1 Premessa all'utilizzo di GA: i cookies	22
2.1.2 Creazione e setup iniziale dell'Account Google Analytics	23
2.1.3 I filtri di Google Analytics	25
2.1.4 Il Menu di Google Analytics	26
2.2 Microsoft Excel	35
2.2.1 La struttura a celle di Excel	35
2.2.2 Importare i dati	37

2.2.3	Le tabelle	39
2.2.4	L'ordine dei dati e i filtri	40
2.2.5	Le funzioni	41
2.2.6	Creazione di grafici con Excel	46
2.3	Python	47
2.3.1	caricare i pacchetti in Python	47
2.3.2	Il pacchetto seaborn	54
3	Un caso di studio	61
3.1	Descrizione del sito web	61
3.2	Quali informazioni ricavare dal sito web?	63
3.3	Raccolta dei dati da GA	64
3.4	Import dei dati su Excel	67
3.5	Costruzione e gestione dei dataset	68
3.5.1	Dataset "utenti per giorno"	70
3.5.2	Dataset "utenti attivi"	71
3.5.3	Dataset "esplorazione utenti"	73
3.5.4	Dataset "tutte le pagine"	75
3.6	Costruzione dei grafici	78
3.6.1	Quanti utenti visitano il sito? Lineplot e scatterplot	78
3.6.2	Gli utenti attivi: lineplot multiplo	80
3.6.3	Provenienza degli utenti: grafico a ciambella e grafico a bolle	80
3.6.4	Caratterizzazione del traffico per sezioni: barplot	83
3.6.5	Le sottosezioni di "Illuminazioni per interni"	83
3.7	Analisi delle visualizzazioni: istogramma e scatterplot dimensionale	85
3.7.1	Stripplot e pairlot: le visualizzazioni suddivise per decadi	87
3.7.2	Visualizzazioni delle pagine suddivisi per decadi	88
3.7.3	Le sottosezioni di "Esterno"	96
3.7.4	Velocità del sito web	97
3.7.5	Sorgente del traffico dati: barplot	106
4	Conclusioni	111

Introduzione

L'analisi dei dati rappresenta, oggigiorno, un'azione imprescindibile per chiunque svolga attività commerciali, e particolarmente per chi opera in rete [Pla11]. Ogni attività di e-commerce, infatti, necessita di gestire in maniera ottimale il proprio sito web per renderlo il più possibile attrattivo nei confronti dei clienti, siano essi potenziali o già fidelizzati [OMS11]. Se di strada ne è stata fatta da quando un giovane commesso scoprì come massimizzare le vendite del negozio in cui lavorava esponendo in scaffali attigui casse di birra e pannolini, il concetto di base resta oggi lo stesso: un'analisi attenta dei resoconti degli acquisti può contribuire ad apportare significativi miglioramenti all'esperienza degli utenti e, di conseguenza, aumentare il tasso di vendite¹. Va da sé che la cura di un sito di e-commerce presenta criticità aggiuntive rispetto a quelle di uno store fisico, quale che sia il già citato market di quartiere o un più moderno negozio di articoli tecnologici. È fondamentale oggi raggiungere nuovi potenziali clienti pescandoli dalla vastità della rete, ed è questa un'operazione molto complessa, che viene effettuata tramite molteplici canali (banner pubblicitari, indicizzazione del sito web, inserzioni nei social network) e il cui approfondimento richiederebbe un lavoro dedicato, ma la cui corretta impostazione determina il successo di uno store in rete.

¹Questa storia viene spesso riportata come il primo esempio di analisi dei dati ai fini economici. Come accade spesso per gli eventi riportati da molteplici fonti, però, i suoi dettagli reali si mescolano ad altri di fantasia. Il riferimento al commesso riguarda una colorita versione dell'aneddoto, che descrive come un giovane commesso di un minimarket americano avesse scoperto, analizzando i registri di cassa, una correlazione tra le vendite di birra e di pannolini, acquistati insieme da giovani padri spediti dalle mogli a fare la spesa il venerdì sera, e avesse deciso di sistemare i due prodotti in scaffali attigui, per massimizzare la loro vendita. Riporto anche una versione più verosimile, proposta da Verhoefh, Kooge e Walk nel loro libro "Creating Value with Big Data Analytics - making smarter marketing decisions" [VKW16] : *"By digging in the data, one might gain very interesting insights, which can guide marketing decisions. The most famous example in this respect is the UK-based retailer Tesco: when analyzing data of their loyalty card, they discovered that consumers buying diapers also frequently buy beer and chips (Humby, Hunt 'e' Phillips, 2008)."*

Gli utenti necessitano di essere attratti dal sito web, di poterlo trovare facilmente, e richiedono un'esperienza di navigazione confortevole e chiara. Al fine di assicurare una navigazione apprezzabile la cura del design del sito stesso è fondamentale, così come il suo continuo aggiornamento, sia in termini di contenuti sia in termini di organizzazione grafica [UM02]. Il cliente che compra dal sito web di una determinata compagnia vorrà, sperabilmente, acquistare un nuovo prodotto in futuro, e lo farà solamente se avrà avuto ricordo di un'esperienza favorevole. La gestione di tutte queste attenzioni richiede un monitoraggio continuo, e solamente la corretta interpretazione dei dati provenienti da tale monitoraggio può indirizzare verso una migliore e più profittevole organizzazione del proprio sito di e-commerce. L'acquisizione dei dati di vendita e di utilizzo del sito web è, tuttavia, solamente il primo passo da affrontare per poter gestire al meglio il proprio spazio commerciale online. Le informazioni raccolte, infatti, vanno organizzate e filtrate, rese idonee alla consultazione e ad una rapida e chiara visualizzazione. Solo al termine di questa catena di processi i dati in sé saranno davvero utili ai fini commerciali, e la loro utilità sarà tanto maggiore quanto migliore sarà stato il loro processing [AKMZ01]. I software utilizzati per una completa ed efficiente strategia di analisi dei dati sono molteplici [WAS05], e ognuno di essi si caratterizza per molti aspetti:

- versatilità.
- fruizione gratuita / a pagamento.
- capacità di gestire diverse tipologie di dati.
- facilità di utilizzo.
- possibilità di esportare / importare dati da e verso altri software.

In particolare la capacità di gestire i dati ed esportarli per l'utilizzo da parte di altri applicativi è fondamentale, perché permette una più versatile trattazione dei dati stessi e anche perché consente ai dati di passare da persone diverse qualora la gestione non sia ad appannaggio di un solo operatore. In linea generale quando si raccolgono dei dati per un'analisi del traffico web di un sito, è importante agire con ordine e metodo. Per aiutarsi l'analista può porsi alcune domande guida:

- "*Cosa devo scoprire?*".

- "*Di quali informazioni ho bisogno per trovare ciò che devo scoprire?*".
- "*Quali informazioni NON mi occorrono per trovare ciò che devo scoprire?*".

Queste domande qui generalizzate troveranno esplicitazione nel terzo capitolo, dove verranno applicate all'analisi del caso studio in oggetto a questo lavoro di tesi. In questo lavoro di tesi, infatti, analizzerò in dettaglio il processo di raccolta, organizzazione e visualizzazione dei dati di traffico diretto verso un sito di e-commerce italiano dal nome "Luce & Luci"². Utilizzerò la suite di *Google Analytics* per la raccolta dei dati di navigazione, *Microsoft Excel* per l'organizzazione tabellare dei dati raccolti. Infine, per il plotting dei dati, utilizzerò il linguaggio di programmazione *Python*, in particolare i pacchetti *pandas* e *seaborn*. Il processo è stato idealmente suddiviso in tre parti: **acquisizione**, **organizzazione** e **visualizzazione** dei dati, che sono state dettagliatamente trattate nei tre capitoli di questo lavoro. Nella realtà, come è facile immaginare, i tre passaggi appena citati non sono distinti e separati, è anzi necessario pensarli fin dal principio come una unica operazione interconnessa. È molto importante infatti impostare correttamente, fin da subito, le modalità con cui i dati saranno acquisiti e poi trasferiti al secondo software per essere analizzati, filtrati, selezionati: in una parola organizzati. D'altro canto non si può prescindere da un'accurata selezione dei dati necessari, per evitare di ottenere un report finale confuso e inadeguato.

²<https://luceeluci.com/>

Capitolo 1

Raccolta, organizzazione e visualizzazione dei dati

Nel primo capitolo di questo lavoro di tesi descriverò il flusso di operazioni effettuate per l'analisi dei dati presentata in questo lavoro.

1.1 Raccolta dei dati: Google Analytics

Il primo software utilizzato è *Google Analytics* (*GA*)¹.

GA è un'applicativo offerto dalla compagnia Google. Nato nel 2005 [LTT11], esso permette di effettuare analisi sulla modalità di navigazione degli utenti di un determinato sito web. La tipologia di dati acquisibili da un utilizzo accurato di GA è esponenzialmente cresciuta negli anni, al punto da renderlo il tool più utilizzato per l'analisi dei flussi dei dati relativi a un sito web [use], sebbene non sia l'unico esistente. I dati raccolti possono essere analizzati e visualizzati graficamente all'interno dello stesso applicativo, o esportati in diversi formati per essere gestiti da altri software. GA è disponibile in una versione gratuita e in una versione Premium, usufruibile dietro pagamento: in questo secondo caso la gamma di servizi messi a disposizione è ancora più ampia.

I dati raccolti da GA sono molteplici. Essi variano molto in mole e tipologia, e la loro acquisizione totale è operazione non immediata, tutt'altro che semplice e, in fin dei conti, neppure necessaria. In particolare la tipologia di informazioni da recepire

¹da questo momento in poi ci riferiremo a Google Analytics utilizzando l'acronimo GA. La stessa azienda produttrice del software fa riferimento ad esso con la medesima sigla.

1.1. Raccolta dei dati: Google Analytics

è sempre dipendente dalla domanda che l'utilizzatore finale si pone (o che gli viene posta). In altre parole: GA offre, con dovizia di dettagli, tutti i dati possibili sul traffico di dati di un sito web, e sta a noi scegliere quali recuperare.

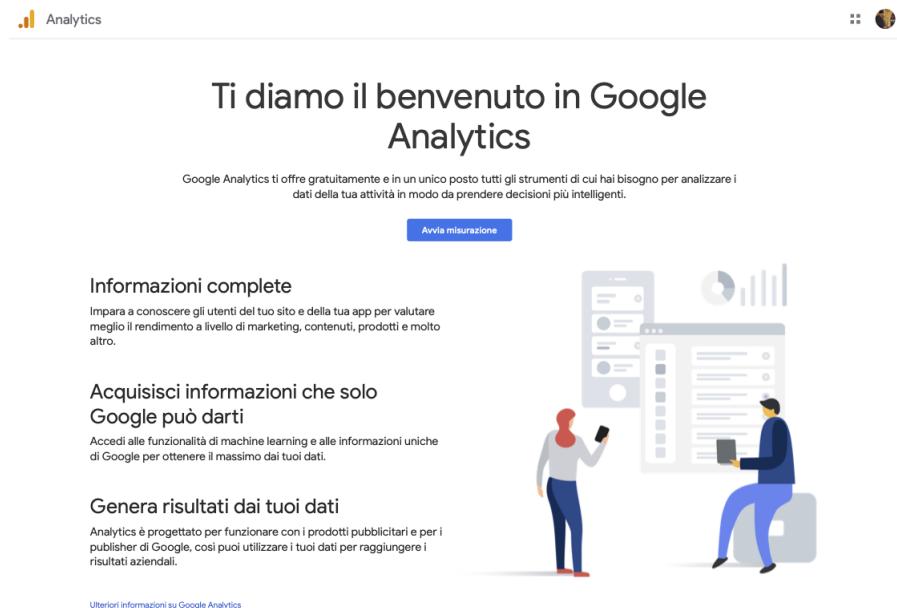


Figura 1.1 – Pagina web di GA.

A titolo esemplificativo: GA offre la possibilità di caratterizzare gli utenti in funzione, tra le altre cose, del loro genere (maschile o femminile) e della tecnologia utilizzata per navigare in rete. Queste informazioni, di per se stesse non mutualmente esclusive, possono avere pesi estremamente differenti nell'analisi del traffico di due siti di e-commerce differenti. Uno store di prodotti tecnologici (portatili, smartphone) sarà estremamente interessato a conoscere il sistema operativo dell'utente che sta navigando il proprio sito. Potrà infatti sapere, in questo modo, quanti sono gli utenti che stanno valutando l'acquisto di un prodotto possedendone uno della medesima azienda e anche, al contrario, quanti sono gli utenti potenzialmente interessati ad acquistare un prodotto senza già possederne uno (e quindi di fatto abbandonare un competitor). Certamente meno interesse desterà il sesso dei potenziali acquirenti, essendo un portatile uguale per uomini e donne. All'opposto si può immaginare come un e-commerce di abiti abbia notevole vantaggio nel conoscere il genere predominante della sua clientela, in modo da orientare le proprie offerte verso capi femminili o maschili, e si comprende anche facilmente come la scelta di un pantalone o di una gonna difficilmente siano influenzati dal fatto che il loro acquirente stia navigando utilizzando un portatile Windows o uno smartphone della Apple.

Con questi semplici esempi teorici si vuole, quindi, ribadire un concetto fondamentale: il dato è valido solo se la sua raccolta si appoggia ad una precisa richiesta di base, e solamente se è in grado di aderire ad essa quanto più fedelmente possibile [SS14]. Una mole eccessiva di dati porta con sé notevoli problematiche. È molto complicato, ad esempio, gestire una dashboard di riepilogo finale dovendo inserire un numero elevato di grafici. Quando essi sono troppi l'attenzione degli ascoltatori è messa a dura prova, e l'impatto finale genera confusione. In ottica di esportazione si può incorrere in altri problemi: files molto grandi, numero eccessivo di interconnessioni tra varie tabelle, rallentamento delle operazioni di data cleaning. Non bisogna mai dimenticare, in altre parole, che tra le "V" che caratterizzano i dati c'è, senza dubbio, la "Velocità", ma anche il "Valore" e la "Veridicità" (figura 1.2²).

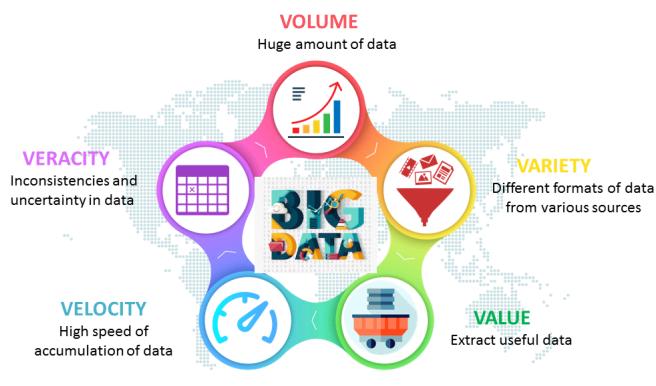


Figura 1.2 – Le 5 "V" dei Big Data.

1.1.1 Perché raccogliere i dati del traffico verso un sito web?

Monitorare il modo in cui viene visitato un sito web ha lo scopo primario di comprendere l'esperienza degli utenti nel navigare su quel sito, che si riflette sull'esperienza provata nel rapporto con l'azienda. Nell'era digitale, ogni aspetto reale della nostra vita ha una sua controparte virtuale: la comunicazione, l'informazione e, chiaramente, gli acquisti. La visita ad uno store fisico per valutare, ed eventualmente concretizzare l'acquisto di un prodotto si è trasformata in una visita virtuale ad un sito web, che rappresenta un'enorme vetrina. Ad essere precisi questo aspetto non è una semplice sostituzione, visto che l'esperienza all'interno di uno store fisico resta una realtà, che

²<https://bigdatopath.wordpress.com/2021/01/27/5-vs-of-big-data/>

però non è più l'unica a disposizione del cliente/utente³. La facilità disarmante con cui oggi è possibile concludere una transazione online spinge sempre di più ad acquistare stando seduti di fronte allo schermo del proprio computer [sol]. Esempio estremo della facilità di acquisto online sono i "dash button" di Amazon [FR17, RFK19]. Il meccanismo è semplice: sono dei piccoli device elettronici che presentano un unico tasto. Premendo su di esso si effettua automaticamente un ordine, senza dover completare i passaggi solitamente richiesti (scelta, trasferimento nel carrello, aggiunta dell'indirizzo di spedizione, scelta della modalità di pagamento, conferma finale). Tale semplificazione delle operazioni è pensata per gli acquisti ricorrenti, e infatti i dash button sono associati ai prodotti di uso più frequente: pannolini, creme, dentifricio. Vale la pena precisare come la stessa Amazon abbia interrotto la produzione fisica dei dash button a partire dal 1 Marzo 2019, sostenendo che essi non siano più necessari potendo ora effettuare ordini ancora più velocemente tramite dispositivi di riconoscimento vocale ("Alexa")⁴.

È naturale comprendere come chiunque offra un servizio di e-commerce si occupi e si preoccupi di organizzare il proprio sito web al meglio per offrire l'esperienza di navigazione e di acquisto migliore possibile, sperando che la clientela possa tornare per il prossimo acquisto. È fondamentale quindi conoscere l'opinione dei clienti sulla loro esperienza, e i dati raccolti attraverso GA non lasciano dubbi al proposito.

1.1.2 Quali informazioni si ottengono da GA?

Il parallelismo tra esperienza di acquisto in uno store fisico e quella in una vetrina virtuale perde di fattibilità quando si tengono in considerazione alcuni aspetti:

- gli acquisti online sono anonimi.
- gli acquisti online non seguono le stesse dinamiche temporali di quelli fisici (si dice infatti che gli "utenti online comprano a stadi" [Ber20]).
- il sito web viene tendenzialmente ricercato con un motore di ricerca, ma il risultato non indirizza quasi mai alla pagina principale, bensì direttamente alla pagina del sito web che contiene il prodotto per cui abbiamo effettuato la ricerca.

³"Nel pieno delle trasformazioni imposte dall'avvento dell'era digitale, le aziende sono chiamate, allo stesso tempo, a ripensare l'idea di store fisico, trasformandola per offrire ai clienti un luogo di relazione di valore" [neg19].

⁴oggi i Dash Button esistono, ma sono solamente "virtuali".

- il sito web può essere raggiunto immediatamente dopo aver visualizzato una pubblicità.
- gli acquisti online possono essere effettuati da qualsiasi parte del mondo.

Su questi aspetti si fonda l'analisi del traffico diretto verso un sito web.

Gli acquisti online sono anonimi nel senso che chiunque può acquistare un bene o un prodotto senza interagire fisicamente con un addetto alla vendita. Questo aspetto rende critica l'identificazione di una "clientela tipo" verso la cui ricezione possono essere indirizzati gli sforzi di gestione ed eventuale trasformazione delle proprie azioni commerciali. GA ci permette di recuperare quante più informazioni possibili riguardo ad ogni singolo utente, in modo da colmare questo gap. GA infatti identifica gli utenti inquadrandoli in particolari *segmenti*: genere (uomo - donna), fascia d'età (18-24, 25-34, 35-44, 45-54, 55-64, 65+), e una più o meno lunga serie di altri segmenti che identificano gli *interessi* dell'utente stesso. In questo modo si cerca di comprendere che tipo di persona naviga più frequentemente il sito e che, quindi, mostra più interesse per i nostri prodotti. L'identificazione di queste informazioni avviene tramite l'utilizzo dei *cookie* [PKM19]⁵. In questa maniera, in definitiva, GA registra la nostra clientela e la classifica, ottenendo informazioni riguardanti preferenze e abitudini della stessa.

Gli acquisti online non seguono le stesse dinamiche temporali di quelli fisici perché il coinvolgimento fisico è fondato su basi estremamente diverse [RRHV21]. L'approdo ad uno store fisico prevede alcune decisioni premeditate: trovare del tempo, stabilire l'ora di arrivo e quella (approssimativa) di uscita dallo store, avere un'idea più o meno precisa dell'acquisto che si intende operare, preventivare un'esperienza sensoriale con l'oggetto che intendiamo acquistare (provare un abito, testare le caratteristiche

⁵i *cookie* hanno un particolare rilievo nell'identificazione del soggetto che utilizza la rete internet. Essi sono spezzoni di codice utilizzati da ogni sito web nell'istante in cui un utente approda su di esso. Raccolgono prevalentemente informazioni riguardanti le ricerche dell'utente stesso (la cronologia di navigazione), e permettono al sito che li ha lanciati di seguire, per periodi più o meno lunghi, le sue abitudini. Nati negli anni '90, da lungo tempo sono oggetto di attenzione legislativa per evitare che essi vengano utilizzati per operazioni di violazione della privacy. La loro trattazione approfondita meriterebbe un lavoro a parte, in questa sede mi limito ad elencare i loro principali utilizzi: permettono di ricordare le credenziali di ingresso ad una pagina (evitando di doverle inserire nuovamente ad ogni visita), mantengono gli acquisti nel carrello di un qualsiasi sito di e-commerce, personalizzano l'esperienza pubblicitaria di ogni utente, personalizzano le home page dei social network.

1.1. Raccolta dei dati: Google Analytics

di un portatile, annusare il tester di un profumo). Al contrario un acquisto in rete è dilazionato nel tempo, non consentendo le operazioni appena descritte, e richiede una più lunga osservazione e valutazione dei prodotti. Un acquisto online viene fatto generalmente a *step*, nel senso che è possibile selezionare il prodotto da acquistare e decidere di concretizzare l'acquisto in un momento successivo. Il tempo che intercorre tra queste due fasi può essere anche molto lungo, soprattutto se l'utente nutre ancora dubbi sull'acquisto in sé. GA raccoglie le tipologie di informazioni che possono descrivere queste dinamiche: quanto tempo impiega un utente a finalizzare un acquisto? Quanto tempo ha trascorso sulla stessa pagina del sito? Quante volte ha abbandonato il sito e quante volte è tornato su di esso? Quanti utenti abbandonano il sito subito dopo essere approdati alla prima pagina [Kam20]?

Il risultato di una ricerca indirizza direttamente alla pagina del sito web che contiene il prodotto per cui abbiamo effettuato la ricerca. Vale a dire che un utente raramente atterrerà alla pagina di presentazione del sito, ma a una sua sezione specifica. Proseguendo il parallelo con lo store fisico, è come se si potesse accedere direttamente ad un reparto che contiene una selezione specifica di oggetti, ossia quelli per cui stiamo effettuando una ricerca. Ne possono risultare situazioni in cui determinate sezioni del sito siano molto visitate mentre altre risultano completamente ignote. È fondamentale quindi conoscere le parole chiave della ricerca che hanno indirizzato a quella pagina, e GA raccoglie queste informazioni.

Il sito web può essere **raggiunto immediatamente dopo aver visualizzato una pubblicità**. È altresì fondamentale sapere come gli utenti siano venuti a conoscenza della pagina web. Ogni banner pubblicitario, o molto più spesso l'inserzione su di un social network, permette in un clic - o in un tap - di raggiungere il sito di cui abbiamo notato l'inserzione. GA raccoglie le informazioni di provenienza del traffico, caratterizzando l'impatto di una campagna pubblicitaria⁶.

Siccome **gli acquisti online possono essere effettuati da qualsiasi parte del mondo**, è fondamentale raccogliere informazioni circa la provenienza di ogni utente.

⁶se la campagna pubblicitaria è costruita con gli stessi strumenti di Google, come ad esempio Google Ads, GA correla molto più efficacemente le visite ricevute e gli avvisi pubblicitari cliccati dagli utenti.

GA raccoglie e cataloga la Nazione di origine del traffico dati e, volendo, può identificare lo stato, la regione e persino la città da cui l'utente naviga.

1.1.3 Le metriche di GA

Per *metriche* si intendono le *tipologie di dati raccolti da GA*, e il loro significato. Da ognuna di esse si possono recuperare informazioni preziose sul comportamento degli utenti durante le loro sessioni di navigazione. Vedremo nel dettaglio tutte le informazioni che è possibile ricavare nel capitolo successivo, qui elenchiamo quelle principali esplicitando il loro significato:

- *Sessione*: rappresenta il tempo trascorso da un utente sulle pagine web del sito⁷, e contemporaneamente anche tutte le sue azioni su quelle pagine.
- *Traffico Utenti*: rappresenta il numero dei visitatori unici (ossia identificati da un solo ID) che hanno navigato il sito (in unità di tempo).
- *Nuovo utente*: rappresenta il visitatore che genera traffico (si connette) verso il sito in analisi per la prima volta.
- *Utente di ritorno*: rappresenta il visitatore che genera traffico (si connette) dopo averlo già visitato in passato.
- *Acquisizione di novo utente*: rappresenta il canale (mail, motore di ricerca, approdo diretto) attraverso cui un nuovo utente raggiunge il sito.
- *Sessione per dispositivo*: è l'indicazione delle tipologie di dispositivi utilizzate dagli utenti che navigano il sito web.

1.1.4 Le conversioni

Una *conversione* è il processo attraverso il quale un nuovo utente si trasforma da potenziale cliente a cliente effettivo. Questa definizione non è sempre così semplice e lineare, e occorre specificare che la conversione **può rispondere a domande diverse**. Per una qualsiasi azienda il goal finale è sempre e comunque quello del profitto, e questo

⁷occorre ricordare che sebbene sia possibile misurare il tempo durante il quale l'utente risulta collegato ad una pagina del sito web, non è possibile conoscere se effettivamente egli sia attivamente collegato (o abbia, ad esempio, abbandonato la pagina senza chiuderla).

obiettivo si identifica immediatamente col numero di acquisti effettuati da un cliente. L'esplicitazione più semplice dell'idea di conversione, quindi, è: "quanti nuovi utenti hanno acquistato un prodotto dal nostro sito web?". A seconda degli obiettivi aziendali, però, è possibile identificare la conversione in altri modi:

- un utente che ha visualizzato una determinata pagina o una determinata sezione del sito.
- un utente che ha visualizzato un determinato banner pubblicitario.
- un utente che ha scaricato un determinato materiale dal sito.
- un utente che si è iscritto alla mailing list del sito.

L'idea di conversione, chiaramente, può variare nel tempo a seconda del particolare periodo storico attraversato dall'azienda stessa.

1.1.5 Come utilizzare i dati acquisiti da GA?

Abbiamo visto, e vedremo ancora di più nel dettaglio nel prossimo capitolo, come GA possa raccogliere molte tipologie di dati. Aggiungiamo qui che questi dati possono essere raccolti per un periodo di tempo indefinito⁸, e persino in tempo reale. Diventa chiaro che la mole di dati acquisita ben presto può diventare notevolmente ampia. È necessario quindi avere la possibilità di gestire questi dati in maniera coerente, e trarre da essi le informazioni nella maniera più chiara possibile. GA offre la possibilità di visualizzare graficamente i dati che si acquisiscono, attraverso plot a barre, a torta o a linea. Con questi grafici, aggiornabili in tempo reale, possiamo avere una grafica di overview e percepire l'andamento di un determinato dato nel tempo. Quando questi dati però devono essere presentati più organicamente, verosimilmente inseriti all'interno di un contesto più ampio di presentazione, i suddetti grafici possono dover essere implementati⁹. Esiste allora la possibilità di **esportare** i dati in file tabellari (csv).

⁸non si confonda il lasso di tempo per cui GA *conserva* i dati acquisiti (variabile a seconda delle impostazioni) prima di eliminarli con il fatto che essi possono essere acquisiti per qualsivoglia periodo di tempo, premunendosi di esportarli per non perderli.

⁹GA offre la possibilità di creare delle dashboard personalizzabili [VDK⁺13], in cui plottare le metriche analizzate con le più comuni tipologie di grafici. Sebbene la possibilità di personalizzazione sia maggiore rispetto ai grafici aggiornabili in tempo reale, non è comunque completa e rende difficoltosa la selezione di sole porzioni di dati.

1.2 Organizzazione dei dati: Microsoft Excel

Il secondo software utilizzato è *Microsoft Excel*.

1.2.1 Scopo e generalità di utilizzo

Lo scopo principale di Excel è quello di **organizzare** serie di dati, offrendo una visuale chiara e personalizzabile e correlarli tra di loro in maniera rapida e semplice [BC07]. Excel è un *foglio di calcolo* ideato da Microsoft nel 1985 (versione 1.0), organizzato in *celle*, all'interno delle quali si possono inserire valori di varia natura: numeri, stringhe di testo, date e formule. Le celle in Excel sono identificate univocamente da una serie di coordinate: un numero (visibile in alto, e che quindi identifica le **colonne**) e una lettera in maiuscolo, visibile a sinistra (e che quindi identifica le **righe**). Ogni volta che si vuole fare riferimento ad una cella, quindi, essa si indica in maniera univoca con una lettera e un numero. Ogni cella contiene un valore, e le celle possono essere collegate tra di loro per effettuare operazioni tra i loro contenuti attraverso, appunto, le formule. Il parco di formule messe a disposizione da Excel è cresciuto negli anni: dalle semplici operazioni matematiche a quelle finanziarie [May20], test statistici avanzati [LMMB12], gestione di stringhe¹⁰. Ogni inserimento di un valore all'interno di una cella di Microsoft Excel comporta l'aggiornamento in tempo reale di tutte le celle ad essa collegate. Excel organizza ulteriormente lo spazio di lavoro tramite le "*Sheet*", ossia i *fogli*: serie di dati possono essere inseriti in fogli diversi, selezionabili cliccando sul loro nome in basso a sinistra o con una semplice scorciatoia da tastiera (figura 1.3). L'idea dei fogli è quella di separare dati appartenenti a diverse categorie ma che necessitano comunque di venir raccolti nello stesso file: è possibile infatti creare funzioni che richiamino celle appartenenti a fogli diversi.

Erroneamente, Excel viene spesso identificato come una versione "user-friendly" di un DBMS¹¹: sebbene sia ottimizzato per una gestione fluida di grosse quantità di dati, però, esso non può essere sostituito ad un buon DBMS [FM04], e questo perché la fruizione di formule che interconnettono i dati diventa molto pesante quando i dati sono in numero notevole, portando a significativi rallentamenti e anche alla possibilità di errori. Non esiste un limite dichiarato alla quantità di dati per cui Excel si comporta

¹⁰è bene chiarire: Excel può operare su brevi stringhe di testo, ma non è pensato per la gestione ottimale di testi veri e propri.

¹¹sigla che sta per "*DataBase Management System*", ossia Sistema di Gestione di un Database.

1.2. Organizzazione dei dati: Microsoft Excel

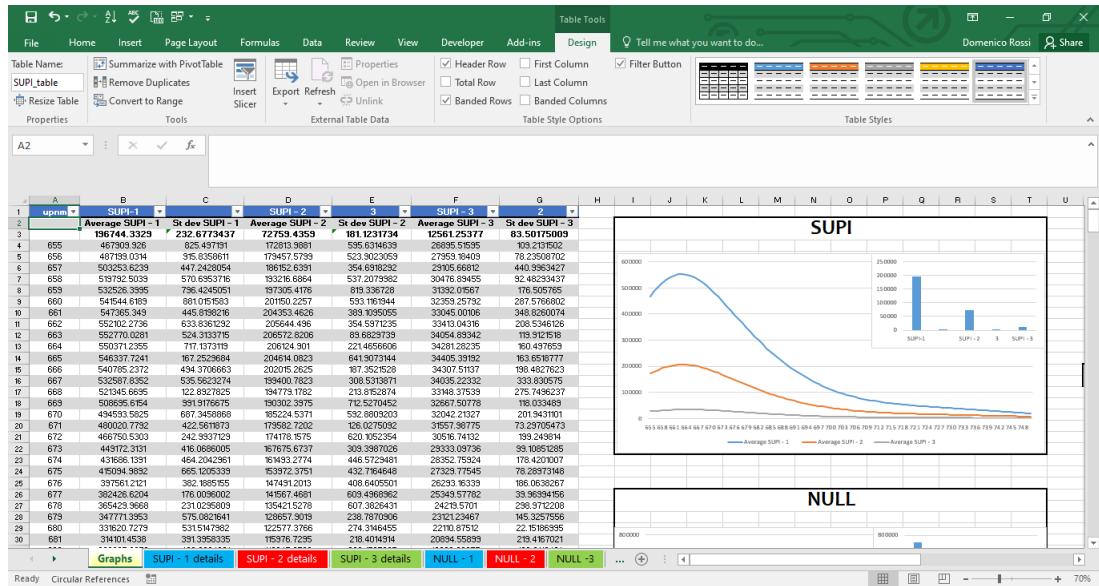


Figura 1.3 – File Excel con una tabella, alcuni grafici riepilogativi e vari fogli utilizzati (evidenziati da colori diversi).

in maniera ottimale e al di là del quale andrebbe preferito un'altra metodologia di gestione. Per rispondere alla domanda se preferire o meno Excel per la gestione dei propri dati si possono elencare i punti deboli e i punti di forza di questo programma.

• Punti di forza di Excel:

offre una buona visualizzazione dei dati: si possono facilmente vedere *tutti* i dati semplicemente scrollando la pagina, si può personalizzare ogni singola cella, si può effettuare una ricerca per risalire a un dato di cui non si ricorda la posizione (comandi simili a quelli di un foglio di testo).

permette di visualizzare velocemente le funzioni di ogni determinata cella, evidenziando le altre celle a cui essa fa riferimento.

offre una dettagliata caratterizzazione di ogni funzione, in modo da aiutare la sua composizione qualora non se ne ricordi la sintassi.

permette di copiare il contenuto di ogni funzione e incollarlo in una nuova: il programma aggiorerà automaticamente i riferimenti posizionali della funzione.

permette di filtrare velocemente i dati e ottenere solo parte di essi (previa costituzione di tabelle).

permette di creare grafici rapidamente, selezionando le celle di dati interessate.

- **Punti di debolezza di Excel:**

non ha la stessa stabilità di un database.

la visualizzazione di grandi moli di dati contemporaneamente dà l'illusione di avere gli stessi sotto controllo, ma genera confusione e aumenta la possibilità di errori da parte dell'utente.

l'automatizzazione delle operazioni (macro) può generare file grandi e a volte instabili.

per sua natura Excel decentralizza i dati: essi sono presenti su solo un file, e non sono a disposizione di più utenti contemporaneamente.

non tiene traccia delle modifiche apportate. Se esse inseriscono un errore quel-l'errore rimane e può non essere recuperabile.

1.2.2 Perché importare i dati in Excel?

Dopo l'acquisizione dei dati da parte di GA, li ho esportati per avere una maggiore possibilità di organizzazione e gestione. Quali sono i motivi per cui conviene effettuare questo passaggio, e perché con Microsoft Excel? La scelta del foglio di calcolo per l'organizzazione dei dati di questo lavoro risponde ad alcune necessità.

In primo luogo la necessità di avere una **visione organica dei dati**. Dopo aver importato i dati in Excel, infatti, essi appaiono ordinatamente visualizzabili nella loro totalità. Come già specificato, in linea generale questo può essere uno svantaggio, perché una quantità ingente di dati visualizzata a schermo può confondere l'utente, e appesantire inutilmente il foglio di calcolo stesso. Nel caso in analisi, tuttavia, i dati raccolti non raggiungono una quantità così elevata da comprometterne la gestione. Inoltre, i pacchetti di acquisizione dati di Python si connettono molto facilmente con Excel, e avere un file preordinato da cui attingere è sicuramente un vantaggio, come si vedrà nel prossimo capitolo.

Un'ulteriore necessità è quella di avere un'**idea dei dati preventiva** alla creazione dei grafici, per capire quale come organizzare il plot dei dati stessi. Una delle più interessanti funzioni di Excel, infatti, è quella della creazione di *tabelle*. Quando una serie di dati vengono inseriti nelle celle di un foglio Excel, essi possono essere facilmente convertiti in una **tabella**, e questa operazione presenta molti vantaggi. Il più interessante è quello di poter avere a disposizione un immediato resoconto dei valori

inseriti: alla fine della tabella infatti si genera una riga di report in cui viene mostrato il valore medio, quello minimo e quello massimo, e con pochi clic si può ottenere anche l’andamento di una particolare colonna o riga. Questa rapida visualizzazione permette di capire se ci sono dati che superano di molto la media e che vanno considerati quindi *outsider*¹², quale sia in generale il range su cui attenersi per una corretta grafica¹³, quali tipologie di grafici è bene utilizzare.

La terza e ultima necessità è quella di avere un **backup ordinato** dei dati. GA conserva i dati, ma essi risiedono all’interno di un database che può non essere agevole da interrogare. Spesso i dati hanno bisogno di essere riconsiderati più volte, sia per correggere eventuali errori, sia per poter effettuare analisi diverse. Accade comune-mente infatti che alcuni set di dati vengano nuovamente interrogati a distanza di molto tempo per poter rispondere ad una domanda diversa da quella che ne aveva generato la raccolta. Avere a disposizione una raccolta di dati già ordinata, ed eventualmente com-mentata, offre un grande vantaggio in questo senso, perché ci permette di recuperarli già ordinati e velocizzare le nuove operazioni.

1.3 Visualizzazione dei dati: Python

La terza parte dell’analisi dei dati, concernente la **visualizzazione** degli stessi, è stata effettuata utilizzando i pacchetti di gestione dati e creazione di grafici di Python.

1.3.1 Storia e caratteristiche

Python è un linguaggio di programmazione di alto livello, ideato negli anni ’90 da Guido Van Rossum [pyta]. A oggi è uno dei linguaggi di programmazione più utilizzati, e sicuramente uno dei più conosciuti. Si caratterizza per una sintassi molto sempli-ce e lineare e una costruzione che lo rende molto comprensibile anche ai non esperti del settore (si dice che Python sia il linguaggio perfetto per chi vuole *imparare a pro-grammare* [pytb]). Sicuramente il punto di forza più importante è rappresentato dal costante aggiornamento a cui questo linguaggio va incontro. Negli anni si è costituita

¹²sebbene sia più comune assistere a questo tipo di dati in ambito scientifico: un dato *outsider* è un dato che si discosta notevolmente dal valore medio di tutti i dati, ed è quindi verosimilmente errato.

¹³l’utilizzo delle scale di riferimento è fondamentale quando si costruisce un set di grafici: una scala di riferimento coerente (stessi valori di minimo e massimo, stessa suddivisione, degli intervalli) li rende più immediatamente confrontabili.

una comunità di appassionati che, successivamente sotto la direzione dello stesso Guido van Rossum, ha acquisito la connotazione di ufficialità [pytc]. Questa comunità si occupa della diffusione di questo linguaggio di programmazione, della filosofia su cui esso è basato¹⁴ e del suo continuo aggiornamento. Python non ha in sé tutte le funzioni necessarie ad un funzionamento completo, e per poter effettuare tutte le operazioni possibili è necessario importare di volta in volta dei *pacchetti* specifici. Così facendo si possono importare esclusivamente i pacchetti di nostro interesse, e l'editor non elabora informazioni non necessarie, che appesantirebbero il codice. Fin dall'inizio l'implementazione di molti di questi pacchetti ha riguardato aspetti concernenti l'analisi dei dati (gestione e modifica di database, creazione di grafici). Un altro aspetto molto interessante di questa organizzazione a "pacchetti" è che accanto a quelli "ufficiali" ne sono nati molti sviluppati da terze parti¹⁵. Il loro costante utilizzo da parte degli utenti, tuttavia, li ha resi sempre migliori e alcuni di essi, alla fine, sono finiti sotto l'egida della stessa comunità ufficiale, che li sviluppa e li aggiorna continuamente.

Python non è, chiaramente, privo di difetti, e quello principale è che, essendo costruito come un linguaggio ad alto livello, risulta essere lento rispetto ad altri linguaggi di programmazione di basso livello. Inoltre non è pensato per lavorare molto bene in multi-thread, sfruttando, cioè, tutti i core del processore. Ciononostante, esso gode di ottima popolarità ed è costantemente utilizzato da numerose compagnie hi-tech: Wikipedia, Google, Yahoo!, CERN, NASA, Facebook, Amazon, Instagram, Spotify [pyt21].

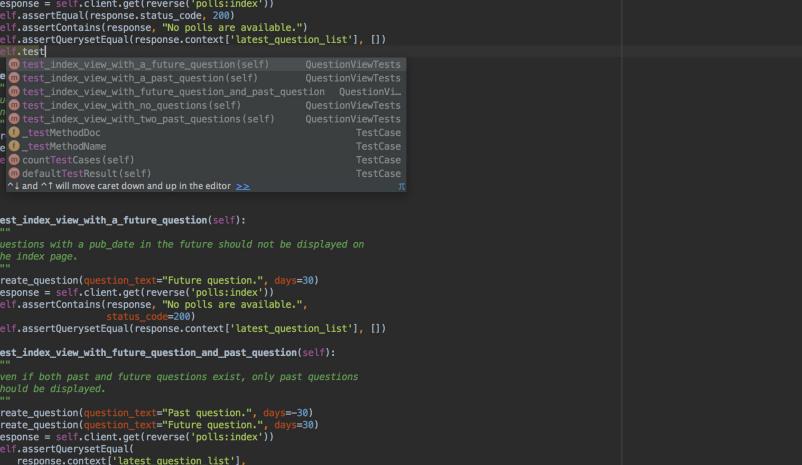
1.3.2 Principali editor di testo per Python

Python è un linguaggio di programmazione multipiattaforma, può funzionare cioè in ambiente Windows, OS e Linux. Per poter operare una serie di istruzioni scritte in Python esistono diverse possibilità, e la più immediata è quella di utilizzare una shell, ossia l'interfaccia a riga di comando. Esistono però degli editor più performanti e più

¹⁴Scrivendo the "The Zen of Python" si ottengono 20 aforismi che sono alla base della filosofia del linguaggio di programmazione. È possibile leggerli a questo link [pytd]

¹⁵*Python is developed under an OSI-approved open source license, making it freely usable and distributable, even for commercial use. Python's license is administered by the Python Software Foundation. The Python Package Index (PyPI) hosts thousands of third-party modules for Python. Both Python's standard library and the community-contributed modules allow for endless possibilities.* Queste due frasi campeggiano sulla pagina principale del sito ufficiale di Python [pyte]

funzionali, soprattutto dal punto di vista dell'esperienza utente: ne citerò due. Il primo editor è *Pycharm*, sviluppato dalla società JetBrains (figura 1.4). Si presenta con un tipico sfondo scuro, e si caratterizza per una notevole attenzione all'utente, nel senso che offre il completamento automatico del codice, segnala errori di sintassi, evidenzia automaticamente le parti della sintassi in colori diversi per una più facile identificazione del testo.



```
20     """
21     response = self.client.get(reverse('polls:index'))
22     self.assertEqual(response.status_code, 200)
23     self.assertContains(response, "No polls are available.")
24     self.assertQuerysetEqual(response.context['latest_question_list'], [])
25     self.assertTrue(response.context['has_questions'])
26
27     def test_index_view_with_a_future_question(self):
28         """Index view for past question should not be displayed on the index page.
29         """
30         question = create_question(question_text="Future question.", days=30)
31         response = self.client.get(reverse('polls:index'))
32         self.assertContains(response, "No polls are available.", status_code=200)
33         self.assertQuerysetEqual(response.context['latest_question_list'], [])
34
35     def test_index_view_with_a_past_question(self):
36         """Index view for past question should be displayed on the index page.
37         """
38
39     def test_index_view_with_a_future_question(self):
40         """
41             Questions with a pub_date in the future should not be displayed on
42             the index page.
43         """
44         question = create_question(question_text="Future question.", days=30)
45         response = self.client.get(reverse('polls:index'))
46         self.assertContains(response, "No polls are available.", status_code=200)
47         self.assertQuerysetEqual(response.context['latest_question_list'], [])
48
49     def test_index_view_with_future_question_and_past_question(self):
50         """
51             Even if both past and future questions exist, only past questions
52             should be displayed.
53         """
54
55         question = create_question(question_text="Past question.", days=-30)
56         question = create_question(question_text="Future question.", days=30)
57         response = self.client.get(reverse('polls:index'))
58         self.assertQuerysetEqual(
59             response.context['latest_question_list'],
60             ['<Question: Past question.>']
61         )
62
63     def test_index_view_with_no_questions(self):
64         """
65             If no questions exist, an appropriate message should be displayed.
```

Figura 1.4 – Esempio di codice Python scritto in ambiente PyCharm (immagine presa dal sito web di Pycharm [pyc]).

Il secondo editor è *Jupyter*. Jupyter è un programma "open source [...]" per la programmazione con numerosi linguaggi" [jup]. La sua caratteristica principale è che opera utilizzando una pagina del proprio browser predefinito. Si presenta con uno sfondo bianco e rispetto al precedente è più scarno su aspetti quali autocompilazione ed evidenziazione del testo¹⁶. Al contrario di Pycharm presenta un grosso vantaggio: organizza il testo in blocchi (figura 1.5). In questo modo si può scrivere una parte di codice in un blocco e valutarne il funzionamento, senza la necessità di completare tutto il restante codice. La suddivisione a blocchi è molto utile durante i processi di debug. Il lavoro di tesi è stato svolto utilizzando quest'ultimo editor.

¹⁶che pure è presente per alcuni comandi ed espressioni di base.

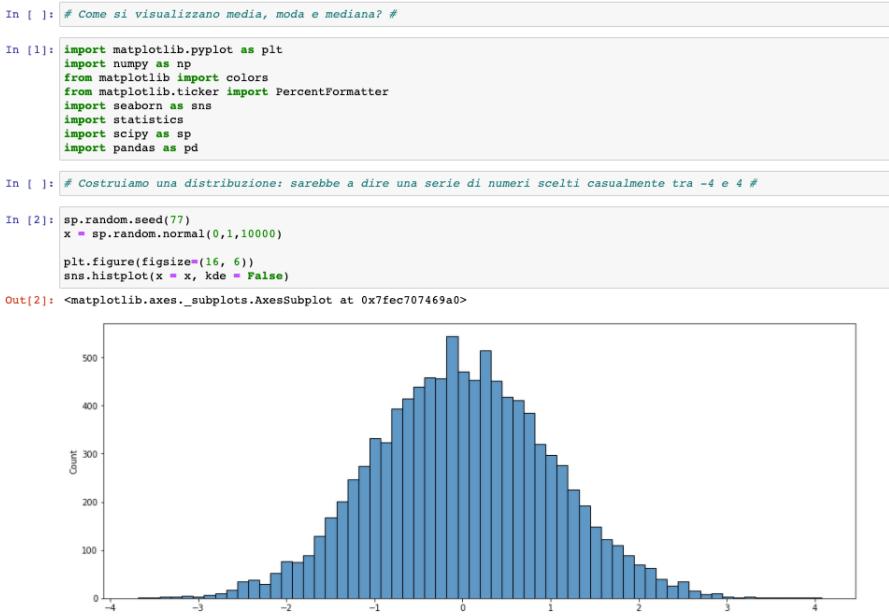


Figura 1.5 – Esempio di codice Python scritto in ambiente Jupyter.

1.3.3 I pacchetti *pandas* e *seaborn*

I due pacchetti maggiormente utilizzati per questo lavoro di organizzazione e visualizzazione dei dati sono *pandas* e *seaborn*.

pandas è utilissimo per la gestione di dati, creazione e modifica di database, ricerca e visualizzazione parziale o integrale degli stessi. Come indicato dagli stessi sviluppatori sulla pagina di descrizione del sito web [pana], *pandas* è molto versatile perché è in grado di operare con diverse tipologie di dati, da quelli strutturati a quelli non strutturati, matrici e serie temporali. Ha un'impostazione molto vicina a quella di SQL, il che lo rende molto intuitivo per chi vi si approccia per la prima volta conoscendo però già i fondamenti del linguaggio per il management di database. Riportando ancora (e integrando) la descrizione fatta sul sito web, seguono in elenco alcune caratteristiche di *pandas*:

- offre una *gestione semplice dei dati mancanti* (definiti *NAN*), che possono facilmente essere rimpiazzati da qualsiasi tipo di valore.
- permette di *inserire e cancellare intere colonne* da un set di dati e riorganizzare la sua struttura con molta facilità.
- ampia *versatilità nell'unione verticale e orizzontale (joining)* di set di dati.
- possibilità di *importare dati di molteplici tipologie*: CSV, Excel, HDF5.

- specifica funzionalità per le *serie temporali di dati*.

seaborn è un pacchetto per la costruzione di grafici. La sua architettura è costruita su un altro pacchetto (matplotlib) di cui seaborn migliora notevolmente l'aspetto, conservandone i codici di base per il plot. La pagina web di descrizione [sea] lo definisce come un pacchetto per la *visualizzazione di dati statistici*. Inoltre è stato creato per garantire la sua massima potenzialità quando si utilizzano dati organizzati con pandas. seaborn offre numerose tipologie di grafici:

- *plot di distribuzione*: tipici plot statitici, che mostrano la quantità di dati presenti in un set ordinandoli secondo valori crescenti. Sono possibili moltissime personalizzazioni, prevalentemente di tipo grafico (colori, sovrapposizione di *kde*¹⁷).
- *plot a barre*: i plot a barre sono costituiti da barre (verticali o orizzontali) affiancate, ed eventualmente raggruppate. La sua funzione è quella di mostrare lo stesso valore riportato in categorie diverse, o per paragonare categorie affiancandole tra loro. Alla sommità delle barre è possibile aggiungere l'indicazione della deviazione standard¹⁸ della serie di dati.
- *plot a linea*: il plot a linea visualizza l'andamento di un valore al variare del tempo. Permette un'immediata visualizzazione di trend di crescita, stabilità o decrescita. Con *seaborn* è possibile aggiungere le "error bands", ossia indicare, in trasparenza, l'ampiezza di tutti i dati che costituiscono la serie e di cui la linea centrale rappresenta la media.
- *scatterplot*: lo scatterplot è un plot che correla due variabili. Ogni valore, identificato da un punto, è quindi caratterizzato da una coppia di valori. *seaborn* permette di apportare interessanti implementazioni allo scatterplot di base. Ad esempio è possibile ingrandire o diminuire la circonferenza dei punti qualora si voglia correlare una terza variabile (generalmente un'intensità). In questo caso il plot prende il nome di *relplot*. Altra opportunità è quella di implementare il

¹⁷*kde* è la sigla di *kernel density estimation*, un metodo statistico utilizzato per il riconoscimento di pattern all'interno di una serie di valori. In sostanza utilizza i dati per costruire un grafico sovrapposto che possa identificare meglio i valori utilizzati e trovare, qualora ce ne siano, pattern ricorrenti, code di valori oltre la media, doppie o triple distribuzioni, etc... [ZR13].

¹⁸o un qualsiasi altro parametro che indichi l'ampiezza della distribuzione.

1.3. Visualizzazione dei dati: Python

quadrante dello scatter con due plot di distribuzione, che mostrano l'andamento delle due variabili tenute in oggetto.

Capitolo 2

I software utilizzati

Dopo aver genericamente esposto il flusso di operazioni con cui i dati di questo lavoro sono stati trattati, descriverò nel dettaglio i software utilizzati, esplicitando le funzionalità che sono state utili al lavoro stesso di tesi. È necessario precisare, benché ovvio, che le funzionalità analizzate non rappresentano la totalità di quelle possibili per ognuno dei software.

2.1 Google Analytics

In questo capitolo analizzerò nel dettaglio l'utilizzo della suite di GA.

2.1.1 Premessa all'utilizzo di GA: i cookies

Ho già accennato, nel capitolo precedente, ai *cookies*: cosa sono, il loro funzionamento e la loro importanza nel traffico web. In tempi recenti la normativa al riguardo ha notevolmente caratterizzato l'utilizzo di questo strumento. Ogni sito web ha, oggi, l'obbligo di chiarire all'utente, tramite l'utilizzo di una finestra pop up, sulle modalità in cui i cookies influiranno sulla sua navigazione. In figura 2.1 è mostrata una tipica finestra pop up, dove possiamo notare alcune caratteristiche fondamentali (e obbligatorie ai termini di legge)¹:

¹sebbene la normativa sia molto chiara al riguardo è ancora possibile riscontrare siti web il cui l'utilizzo della schermata popup per l'informativa sui cookies non sia conforme alle norme: assenza dell'opzione *personalizza*, assenza dell'opzione *rifiuta*. Un'altra esperienza comune è quella di notare come la finestra popup sparisca non appena scrolliamo la pagina o clicchiamo su qualche contenuto del sito stesso, senza averci permesso di selezionare alcuna scelta. <https://www.altalex.com/documents/news/2021/07/20/cookie-nuove-linee-guida-garante>

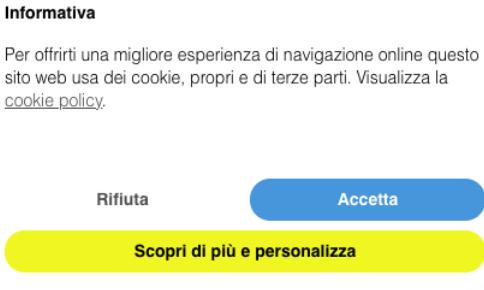


Figura 2.1 – Finestra popup informativa dell'utilizzo dei *cookies*.

- il pulsante *accetta*: cliccando su di esso l'utente accetta l'utilizzo, da parte del sito, dei cookies in maniera preimpostata.
- il pulsante *rifiuta*: cliccando su di esso l'utente rifiuta l'utilizzo dei cookies.
- il pulsante *personalizza*: cliccando su di esso l'utente può scegliere le tipologie di cookies da accettare e quelle da rifiutare.

È fondamentale ricordare, per poter comprendere le dinamiche di GA, che l'azione di tracciare il traffico web è possibile esclusivamente su quegli utenti che accettano l'utilizzo dei cookies.

Nello specifico, è sufficiente che l'utente accetti i *cookies di profilazione*, che si possono abilitare accettando tutti i cookies o selezionandoli dopo aver cliccato sul tasto "personalizza" (figura 2.2).

2.1.2 Creazione e setup iniziale dell'Account Google Analytics

"GA è una piattaforma che raccogliere dati e li organizza in report."². Essa è pensata per offrire una visione di insieme di tutto ciò che accade all'interno di un sito web, raccogliendo dati in tempo reale e/o in un determinato periodo di tempo. Tutto il traffico di dati analizzato da GA può essere **filtrato** per poter corrispondere al meglio alle esigenze della propria ricerca, e raccolto in report che possono essere esportati. Il primo passo per l'utilizzo di un account di GA è la creazione di un account base di

privacy-6-mesi-per-adeguarsi

²Google offre un corso a vari livelli per acquisire le conoscenze necessarie all'utilizzo di GA, organizzato in video ed accessibile a chiunque abbia un account Google. Questa frase apre il primo video del corso base, e possiamo considerarla come la definizione più semplice e basilare di GA stesso.

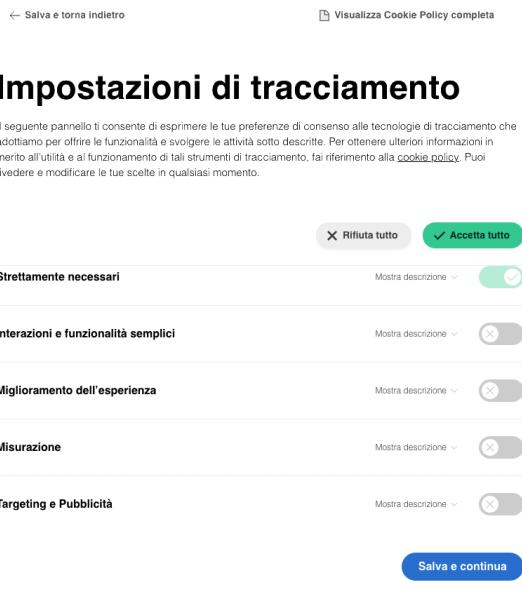


Figura 2.2 – Schermata di personalizzazione dell'utilizzo dei cookies.

Google, all'interno del quale è possibile accedere ai servizi Analytics. Una volta effettuata l'iscrizione è necessario poi settare l'account di GA, inserendo il proprio nome: sarà questo il nome con cui riconosceremo le nostre attività.

A seguire ci verranno richieste altre informazioni addizionali, come la nostra regione geografica di appartenenza, il relativo fuso orario e altre informazioni riguardanti la nostra attività. Google utilizza queste informazioni per preconfigurare la nostra esperienza di utilizzo - va da sé che anche queste informazioni sono dati che vengono raccolti e analizzati da Google per i propri fini commerciali. Una volta configurato l'account di GA si è virtualmente pronti per iniziare il monitoring dei dati. In realtà è necessario un ulteriore step per poter fattivamente ottenere dati, ed è quello di inserire un tag nel sito web da monitorare.

Per poter monitorare i dati di un sito web, Google Analytics richiede l'aggiunta di un tag sul codice sorgente del sito web stesso. Questo tag va aggiunto su ogni pagina del sito, immediatamente dopo la stringa

`</head>`

Dal momento in cui il tag viene aggiunto al codice sorgente del sito web, Google Analytics comincerà a raccogliere i dati provenienti da tutti i singoli utenti che visiteranno quel sito web. I dati raccolti sono molteplici, e di diverso tipo:

- numero di volte in cui il sito viene visitato.
- durata delle visite.
- lingua in cui è impostato il browser del visitatore.
- la tipologia del browser utilizzato del visitatore (Chrome, Safari, etc...).
- device utilizzato dal visitatore.
- sistema operativo del device utilizzato dal visitatore.
- sorgente di traffico (come il visitatore è giunto sul sito: motore di ricerca, banner pubblicitario, etc...).
- pagina del sito web visitata.

I dati vengono raccolti e conservati in un database da cui sono accessibili per visualizzazioni in-site o per l'esportazione. Google ci mette immediatamente in guardia su un aspetto molto importante di questi dati:

Una volta che i dati vengono raccolti essi non possono essere modificati.

Per questo motivo è particolarmente utile impostare una raccolta adeguata, sfruttando soprattutto l'utilizzo dei *filtri*.

2.1.3 I filtri di Google Analytics

L'utilizzo dei filtri è uno dei primi e più importanti passi per una raccolta adeguata, al fine di non acquisire moli di dati disordinati e successivamente difficili - se non impossibili - da utilizzare. Il setup dei filtri permette di *escludere* i dati provenienti da alcune fonti o, al contrario, includere solo questi. Una delle strategie più comuni, ad esempio, è quella che prevede l'esclusione del traffico "interno", ovverosia quello proveniente dagli utenti di GA stesso, che possono dover visitare il sito e monitorarlo allo stesso tempo. In questo modo ci si assicura la sola raccolta di dati esterni, ossia di visitatori reali. È possibile filtrare il traffico in funzione degli indirizzi IP o del dominio di rete in cui ci si trova. È importante ricordare che una volta applicato un filtro, da quel momento GA smetterà di acquisire dati provenienti dai visitatori esclusi da quel filtro. Per tutto il lasso di tempo in cui il filtro sarà attivo, quindi, quei dati non

saranno raccolti. Risulta evidente quindi che una accurata scelta preventiva dei filtri è fondamentale per evitare di perdere informazioni sul traffico che non potrebbero in alcun altro modo essere recuperate.

2.1.4 Il Menu di Google Analytics

Da questo momento in poi descriverò alcuni dettagli del Menù di GA, con le relative funzionalità.

Una volta terminati i settaggi - tra cui quello di generare il codice per il tracking da inserire nel sito - possiamo iniziare la visualizzazione dei dati raccolti da GA. Come descritto nel precedente capitolo, i dati raccolti da GA sono molti, e la loro raccolta totale non è quasi mai necessaria. La scelta dei dati da visualizzare, quindi, è sempre dipendente dal tipo di analisi che si vuole portare avanti, e va ponderata in base alle proprie necessità.

La figura 2.3 mostra le opzioni del menù di GA. Esse sono:

- Home page
- Personalizzazione
- In tempo reale
- Pubblico
- Acquisizione
- Comportamento
- Conversioni

Le prime due sono voci a sé stanti, mentre le successive cinque fanno parte della sezione "*Report*". Nei prossimi paragrafi analizzeremo in dettaglio ognuna di queste voci, concentrandoci in particolare su alcune di esse.

Home Page

Nella sezione *Home page* ci viene mostrata una overview generale delle attività monitorate.

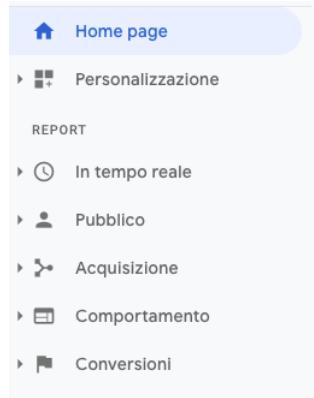


Figura 2.3 – Menù di GA.

Un **primo riquadro** (figura 2.4) mostra gli **utenti** del nostro sito web, caratterizzandoli con il numero di sessioni, la frequenza di rimbalzo (la percentuale di utenti che abbandona il sito dopo essere giunto sulla prima pagina) e la durata media delle sessioni, che indica il tempo di permanenza sul sito.



Figura 2.4 – Riquadro "Utenti"

della schermata Home di GA.

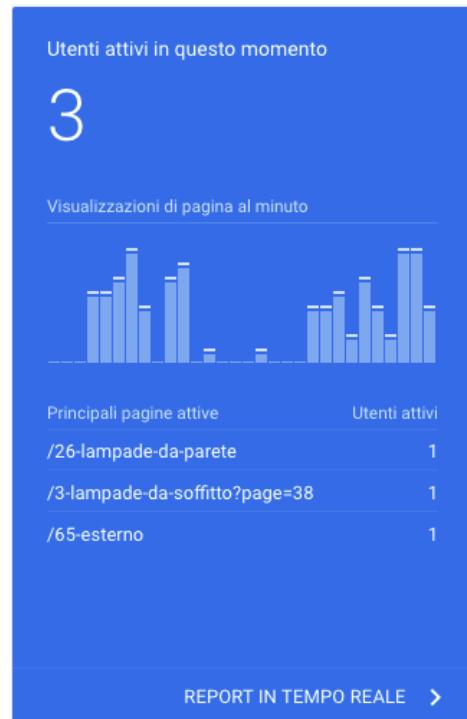


Figura 2.5 – Riquadro "Utenti attivi" della schermata Home di GA.

Un **secondo riquadro** (figura 2.5) mostra gli **utenti attivi** al momento in cui stiamo visualizzando il report di GA.

Un **terzo riquadro** (figura 2.6) mostra il processo di **acquisizione di nuovi utenti**, indicando se essi provengono da un accesso diretto, se mediato da banner

pubblicitari o inserzioni sui social network.

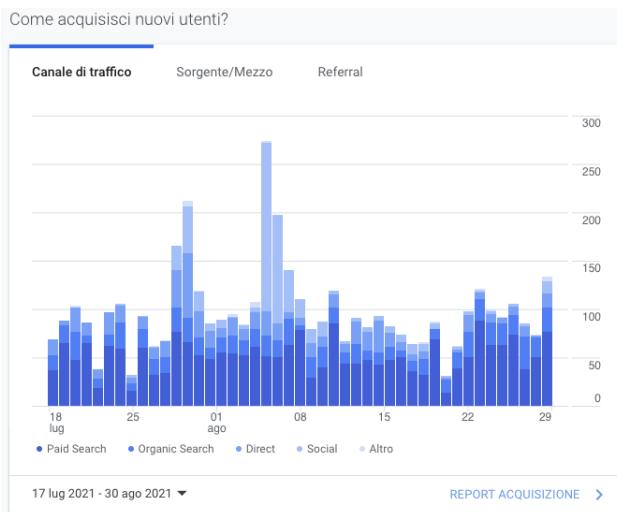


Figura 2.6 – Riquadro della provenienza dei nuovi utenti.

Un **quarto riquadro** (figura 2.7) mostra le sessioni suddivise per **area geografica**, con la percentuale di sessioni per paese.

Un **quinto riquadro** (figura 2.8) mostra la **l'orario di accesso al sito**.

Un **sesto riquadro** (figura 2.9) mostra la **le pagine maggiormente visitate del sito**.

Un **settimo riquadro** (figura 2.10) ci mostra le sessioni suddivise per tipologia di dispositivo utilizzato.

I riquadri sono espandibili per visualizzare i dettagli, relativi al singolo utente, delle informazioni genericamente visualizzate.

Personalizzazione

Nella sezione personalizzazione è possibile creare dashboard e report personalizzati a seconda delle proprie necessità.

In tempo reale

La sezione in tempo reale permette di ricevere la situazione del traffico diretto verso il nostro sito web in tempo reale, monitorando tutti gli aspetti secondo per secondo. È suddiviso in sei sottosezioni, ognuna delle quali approfondisce un particolare aspetto.

La **prima** sottosezione è definita **panoramica**. Cliccando su di essa si possono osservare gli utenti attivi in tempo reale, le visualizzazioni al minuto e al secondo, i referral principali, il traffico proveniente dai social network, le principali pagine attive

2.1. Google Analytics



Figura 2.7 – Riquadro della posizione geografica degli utenti.

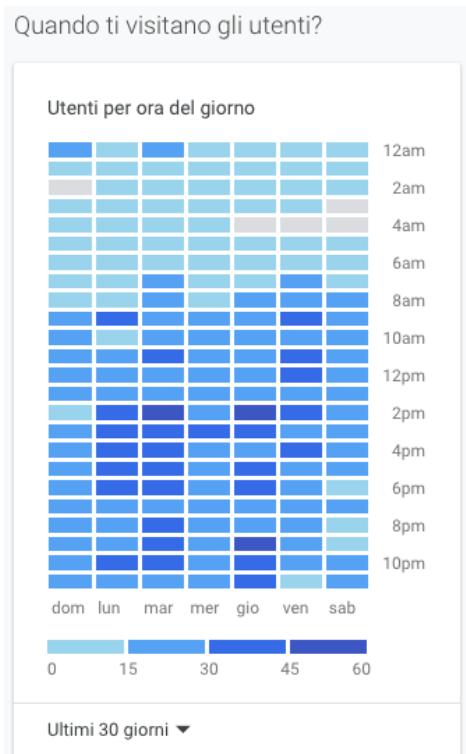


Figura 2.8 – Riquadro dell'ora in cui gli utenti navigano il sito.



Figura 2.9 – Riquadro delle pagine del sito web maggiormente visitate.

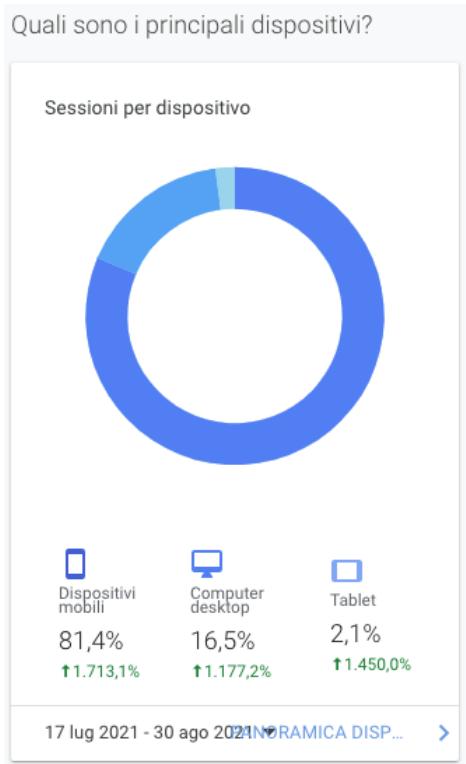


Figura 2.10 – Riquadro dei dispositivi utilizzati per la navigazione sul sito web

2.1. Google Analytics

e le parole chiave principali. È inoltre visibile la provenienza degli utenti che stanno visitando il sito (figura 2.11)

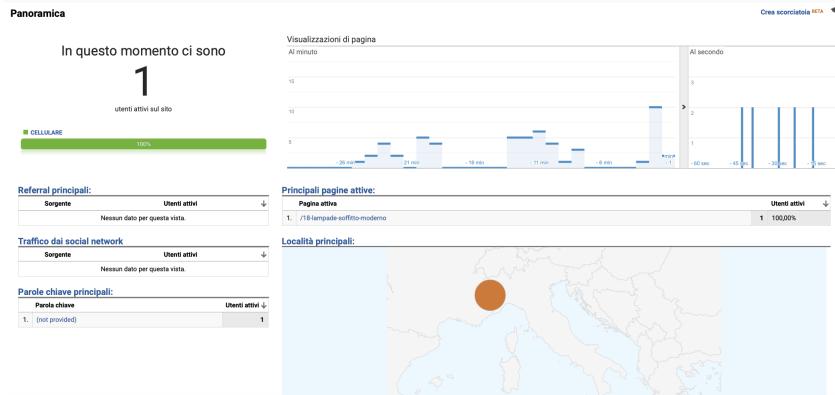


Figura 2.11 – Eventi in tempo reale.

La **seconda** sottosezione è definita **località**, e come suggerisce il nome approfondisce la natura della provenienza geografica dei singoli utenti che stanno visitando il sito al momento.

La **terza** sottosezione è definita **sorgenti di traffico**, e specifica da quale sorgente provengano gli utenti attivi (figura 2.12).

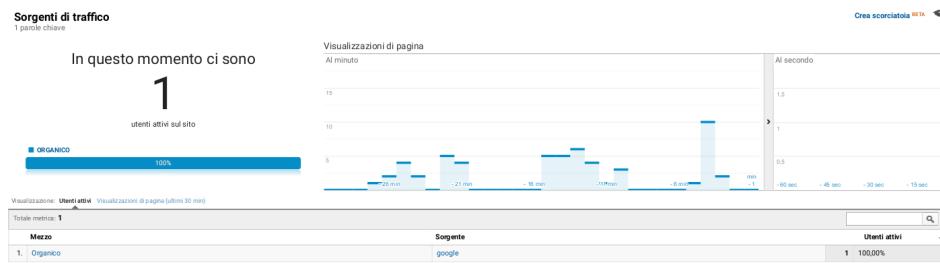


Figura 2.12 – Sorgenti di traffico in tempo reale.

La **quarta** sottosezione è definita **contenuti**. Ci mostra in dettaglio quali pagine del sito sono attualmente visitate e quali sono state visitate nei minuti immediatamente precedenti.

La **quinta** e la **sesta** sottosezione sono rispettivamente **eventi** e **conversioni**.

Pubblico

La sezione **Pubblico** offre a possibilità di espandere le caratteristiche degli utenti che navigano il nostro sito, declinandone molti dettagli. È una sezione molto ampia, che contiene 10 sottosezioni, quattro delle quali a loro volta espandibili e contenenti

ulteriori sottosezioni. Ci soffermeremo sui dettagli di alcune di esse, descrivendo invece in maniera generale le altre³.

La prima sottosezione offre una **panoramica** sull'andamento degli utenti (figura 2.13). In primo piano campeggia un grafico a linea spezzata che mostra la durata media della sessione, frequenza di rimbalzo, numero di sessioni per utente, nuovi utenti, pagine per sessione, sessioni, utenti, visualizzazioni di pagina. Il grafico ottenuto si riferisce al giorno in cui viene visualizzato, ma è possibile allargare o restringere il tempo di visualizzazione fino a un mese. Immediatamente sotto sono presenti dei box in cui si aggiornano i valori numerici relativi alle stesse tipologie di dati che è possibile visualizzare in grafico. In questo modo GA offre una visione d'insieme tramite valori numerici, e la contestuale possibilità di approfondire una visualizzazione grafica degli stessi. Più in basso ancora vi è una tabella che mostra alcuni dati degli utenti: la lingua, il paese e la città di provenienza del traffico di dati; il sistema operativo utilizzato, il Browser e il fornitore di servizi di rete in caso di navigazione da un computer, il sistema operativo, il fornitore di servizi e la risoluzione dello schermo in caso di navigazione da un dispositivo mobile.

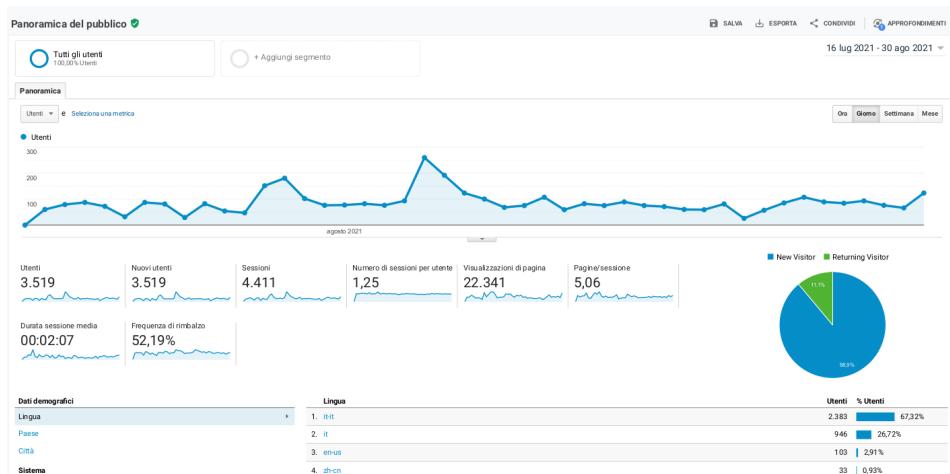


Figura 2.13 – Riquadro della panoramica del comportamento del pubblico.

Una seconda sottosezione permette di osservare il numero degli **utenti attivi**, fino a 28 giorni precedenti quello della visualizzazione. È senz'altro una sezione molto importante, essendo gli utenti attivi quelli maggiormente interessanti dal punto di vista commerciale. Essi infatti sono quelli che con più probabilità porteranno a termine un

³da questo elenco sono state escluse alcune sottosezioni. Per la loro caratterizzazione si rimanda a testi più specifici.

2.1. Google Analytics

acquisto o, comunque, un processo di conversione in base agli obiettivi che l'azienda si è proposta.

Una terza sottosezione chiamata **Dati demografici** è a sua volta composta da: *panoramica*, *età* e *sesso*, e ripropone lo stesso schema visto nella sottosezione *Panoramica*, dettagliando gli utenti in funzione, appunto, di età e sesso.

Una quarta sottosezione chiamata **Dati geografici** riprende lo schema mostrando il paese di provenienza degli utenti e la lingua impostata sul loro device di navigazione (figura 2.14).

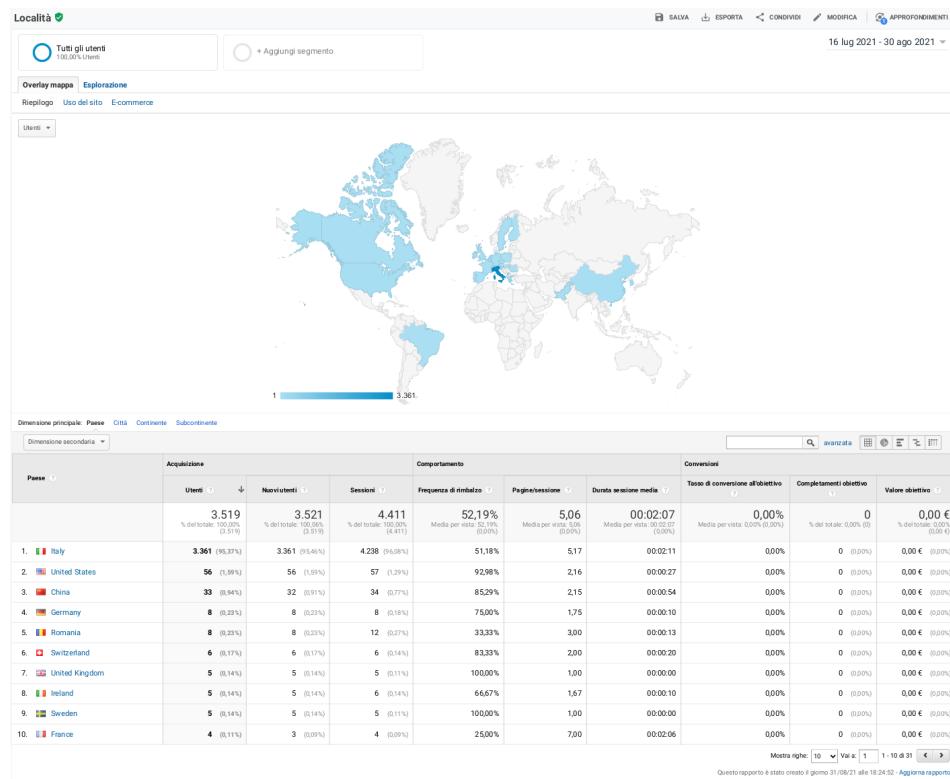


Figura 2.14 – Dati geografici degli utenti che navigano il sito.

Una quinta sottosezione chiamata **Tecnologia** identifica il tipo di browser e sistema operativo utilizzati, oltre che la rete internet con la quale stanno navigando.

Una sesta sottosezione chiamata **Dispositivo mobile** offre il dettaglio delle navigazioni che vengono effettuate tramite dispositivi mobili (figura 2.15).

Acquisizione

Nella sottosezione *Acquisizione* vengono mostrati i dati relativi all'acquisizione di nuovi clienti. È questo un aspetto molto importante dell'analisi di un sito di e-commerce, essendo di fatto l'acquisizione di nuova clientela il primo obiettivo che un'azienda si

2.1. Google Analytics

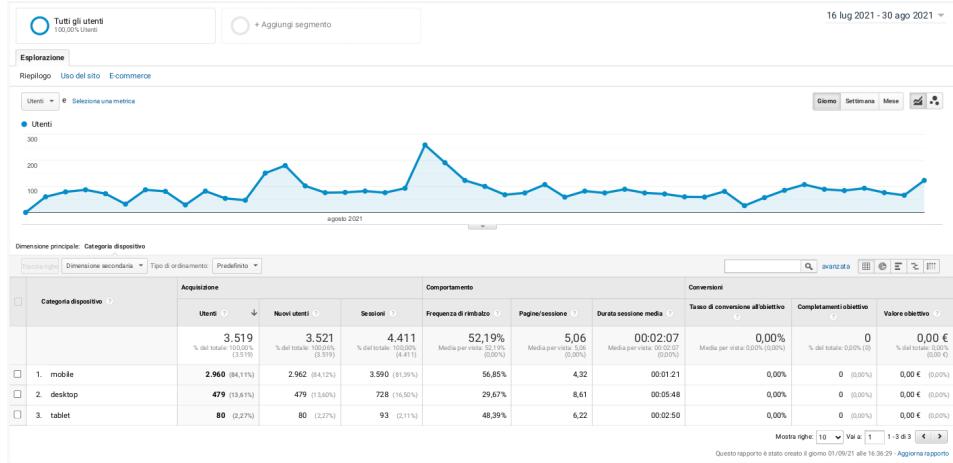


Figura 2.15 – Dettaglio dei dispositivi utilizzati per la navigazione.

pone quando vende online. La gestione di un sito è infatti interamente votata a cercare di offrire la migliore esperienza possibile ai clienti, attirarne di nuovi e, in ultima analisi, fidelizzarli. È composto, anch’esso, da diverse sezioni ulteriori, che sono: *Panoramica*, *Tutto il traffico*, *Google Ads*, *Search console*, *Social*, *Campagne*. Vediamole nel dettaglio.

Nel blocco *Panoramica* si osservano due grafici adiacenti che mettono in relazione il numero di utenti per unità di tempo e il tasso di conversioni effettuate. Una tabella sottostante specifica alcuni dati già reperibili in altre sezioni, come il numero di sessioni, la loro durata media o la frequenza di rimbalzo.

Il blocco *Tutto il traffico* caratterizza l’utilizzo del sito da parte degli utenti tramite le combinazioni di sorgente e mezzo che hanno indirizzato li traffico, con un grafico a torta che suddivide l’apporto delle varie sorgenti all’arrivo dei nuovi utenti.

Il blocco *Google Ads* permette di valutare l’efficienza delle proprie campagne di advertising costruite tramite, appunto, Google Ads. Permette di valutare le Query di ricerca e le parole chiave utilizzate, suddivise per le ore del giorno a cui esse sono state effettuate.

Il blocco *social* ci mostra quante sessioni sono state effettuate dopo che un cliente è stato indirizzato alla pagine attraverso un link su di una pagina social network. La presenza di inserzioni pubblicitarie è oggi una delle strategie più utilizzate per la pubblicizzazione dei propri prodotti, ed è quindi fondamentale conoscere il tasso di tale reindirizzamento. Per poter valutare l’impatto dei social network sul nostro sito è necessario impostare degli **obiettivi**.

Infine il blocco *Campagne* fa mette in risalto l’effetto delle nostre campagne pub-

blicitarie, l'efficacia delle parole chiave e l'analisi dei costi associati ad esse.

Comportamento

Nella sezione *Comportamento* del menù di GA si possono valutare alcuni aspetti del comportamento degli utenti, vale a dire di come essi interagiscono con il nostro sito web. Particolarmente interessanti risultano le parti *Contenuti del sito*, *Velocità del sito* e *Ricerca sul sito*⁴.

Nei *Contenuti del sito* (figura 2.16) è possibile osservare nel dettaglio le pagine del sito web maggiormente visualizzate, quelle su cui più frequentemente gli utenti atterrano⁵ e quelle dalle quali lasciano il sito.

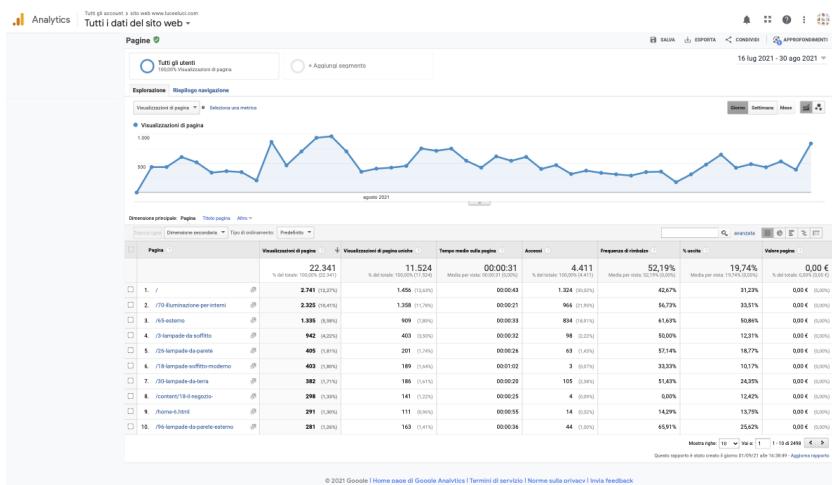


Figura 2.16 – Dati sulle pagine del sito e la frequenza con cui sono visitate.

In *Velocità del sito* (figura 2.17) GA offre un'interessante serie di valori (e la loro visualizzazione grafica): il tempo di caricamento medio di una pagina, il tempo di rein-dirizzamento medio, il tempo medio di ricerca dominio, il tempo medio di connessione al server, il tempo medio di risposta del server e il tempo di download della pagina. È sempre più evidente che la qualità dell'esperienza di navigazione passa per la velocità con cui il sito è disponibile per gli utenti, che non hanno piacere ad attendere un caricamento estremamente lento per poter visualizzare un contenuto. Il monitoraggio

⁴anche in questa sezione si sono volutamente tralasciati alcuni aspetti specifici, per i quali si rimanda ad ulteriori approfondimenti personali.

⁵La pagina di "atterraggio" o di "landing" è, in gergo tecnico, la pagina su cui un utente viene indirizzato da un link, sia esso pubblicitario o sia esso il risultato ottenuto dopo aver interrogato un motore di ricerca. È inusuale, infatti, che un utente raggiunga un sito partendo dalla sua home page e navigandopoi alla ricerca dei contenuti.

di questi tempi è quasi essenziale, e fornisce informazioni fondamentali su come poter migliorare l'esperienza dell'utente. Questa sezione risulta utile, infine, per non cadere nel paradosso di profondere notevoli sforzi nel tentativo di migliorare l'estetica o i contenuti di un sito web senza rendersi conto che questi non sono facilmente accessibili né godibili semplicemente perché impiegano troppo tempo ad essere visualizzabili dall'utente.

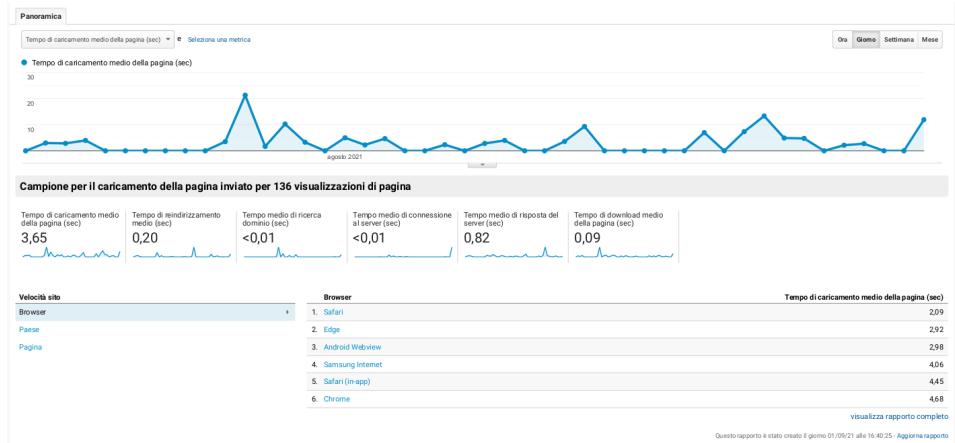


Figura 2.17 – Riquadro della velocità del sito.

2.2 Microsoft Excel

In questo capitolo analizzerò nel dettaglio l'utilizzo del programma Microsoft Excel⁶.

2.2.1 La struttura a celle di Excel

Excel presenta una struttura a celle, identificate da una coppia di valori: un numero posto sulla sinistra del foglio di lavoro identifica le righe, una lettera posta in alto identifica le colonne (figura 2.18). Ogni cella può contenere un valore numerico, un valore testuale o una funzione. La struttura a celle, e il fatto che in ognuna di esse sia possibile inserire un determinato valore permette di raccogliere i dati in maniera simile a quanto è possibile fare con un database. Come già detto, però, Excel non è pensato per funzionare efficientemente come un database (cfr. 1.2.1). La struttura a celle, invece, è più idonea per costruire *relazioni* tra i contenuti di ognuna di esse. Per inserire un contenuto in una cella è sufficiente cliccare su di essa col puntatore e digitare

⁶Le descrizioni di questo paragrafo e dei relativi sottoparagrafi fanno riferimento alla versione di Excel in lingua italiana.

il dato, oppure incollarlo (se lo si è precedentemente copiato). La cella acquisirà, oltre a un valore, anche un formato. Per visualizzare i formati disponibili, o per modificare il formato di una cella, è sufficiente cliccare su "*Numeri*", pannello **Home**.

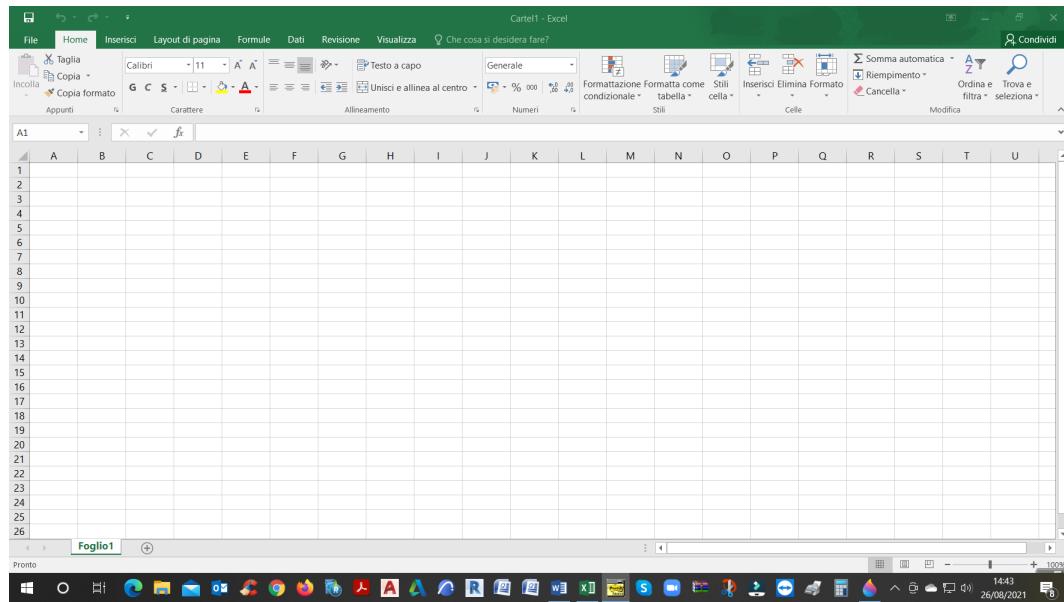


Figura 2.18 – Foglio Excel, con la tipica struttura a celle.

I formati delle celle di Excel

I formati possibili per ogni cella sono i seguenti:

- *Generale* - la cella tratta il suo contenuto senza classificarlo in alcun tipo particolare, sia che si inserisca un testo sia che si inserisca un valore numerico.
- *Numero* - la cella identifica il suo contenuto come un valore numerico. In questo caso, ad esempio, non è possibile inserire un numero che inizi con la cifra 0.
- *Valuta* - la cella considera i valori numerici come ammontare di denaro, nella valuta impostata.
- *Contabilità* - la cella considera i valori numerici come ammontare di denaro, nella valuta impostata, come nel caso della "Valuta". La differenza tra le due tipologie è di formattazione (nella Valuta il numero è allineato alla destra della cella, mentre nella Contabilità si posiziona al centro). Un'altra differenza è che quando impostiamo le celle come "Valuta", esse colorano automaticamente di rosso i valori negativi, risultando molto utili per identificare immediatamente uscite di denaro.

- *Data in cifre* e *Data estesa* - Excel è in grado di gestire le date, che possono essere inserite in numerosi formati, sia estesi che brevi.
- *Ora* - stesso discorso vale per l'ora, in formato esteso o breve, 12 o 24h.
- *Percentuale* - tratta i valori inseriti nella cella come valori percentuali.
- *Frazione* - la cella visualizza i valori decimali come una frazione (esempio: "1,5" viene visualizzato come "1 1\2").
- *Scientifico* - i valori inseriti in questa cella vengono visualizzati in notazione scientifica.
- *Testo* - la cella identifica il contenuto come una stringa di testo.

2.2.2 Importare i dati

I dati possono essere manualmente inseriti in un foglio di calcolo, cella per cella, o anche incollati da altre fonti. Una terza via per l'acquisizione di dati in un foglio Excel è la possibilità di importarli, tramite un apposito comando. Questa opzione è vantaggiosa quando si hanno già a disposizione dati organizzati, presumibilmente provenienti da altri software, che si vogliono gestire in Excel. Esistono numerosi formati di file che Excel permette di importare: *csv*, *XML*, *Microsoft Access*, direttamente dal *web*, da un *testo*, da un database *SQL*, da *Facebook*. Ogni volta che si importano file in Excel il programma ci offre una serie di personalizzazioni da poter effettuare. In questo lavoro descriverò l'acquisizione da csv e le sue personalizzazioni (che sono simili per tutte le tipologie di file).

Importare i dati da csv

CSV è una sigla che sta per **C**omma **S**eparated **V**alues. È il formato che identifica un file di testo in cui i valori in esso presenti sono separati, appunto, da una virgola. È un formato molto comune ed è utilizzato da numerosi software. Un file csv può essere aperto alla stessa maniera da tutti i lettori di testo - è quindi multiplattforma - ed è generalmente molto leggero, anche quando contiene numerosi record. Per importare in Excel dei dati da csv bisogna cliccare sul menù **Dati**, poi su *Nuova richiesta, da file*, e infine *da csv*. La finestra di dialogo che si apre permette di scegliere il file da importare. Una volta scelto il file si apre una nuova finestra di anteprima, che

ci mostra come i dati verranno importati sul foglio Excel (figura 2.19). Qualora non siamo soddisfatti della modalità con cui Excel ha scelto di importare il file, possiamo apportare delle modifiche. La prima modifica che è possibile apportare è l'operatore per la separazione. Normalmente i CSV files contengono i record separati da virgola, ma si possono verificare casi in cui essi siano separati da altri operatori (punto, punto e virgola). In questo modo possiamo scegliere l'operatore più idoneo. Andando oltre con le possibilità di modifica, è possibile scegliere alcune colonne e scartarne altre, ordinare i record in base a criteri scelti da noi⁷, gestire i dati mancanti, aggiungere un indice, scegliere il formato delle celle di destinazione, scegliere la cella in cui verrà importato il primo valore in alto a sinistra. Quando siamo soddisfatti delle nostre impostazioni, clicchiamo su "importa" e i dati verranno riversati sul foglio Excel, pronti per essere osservati e trattati.

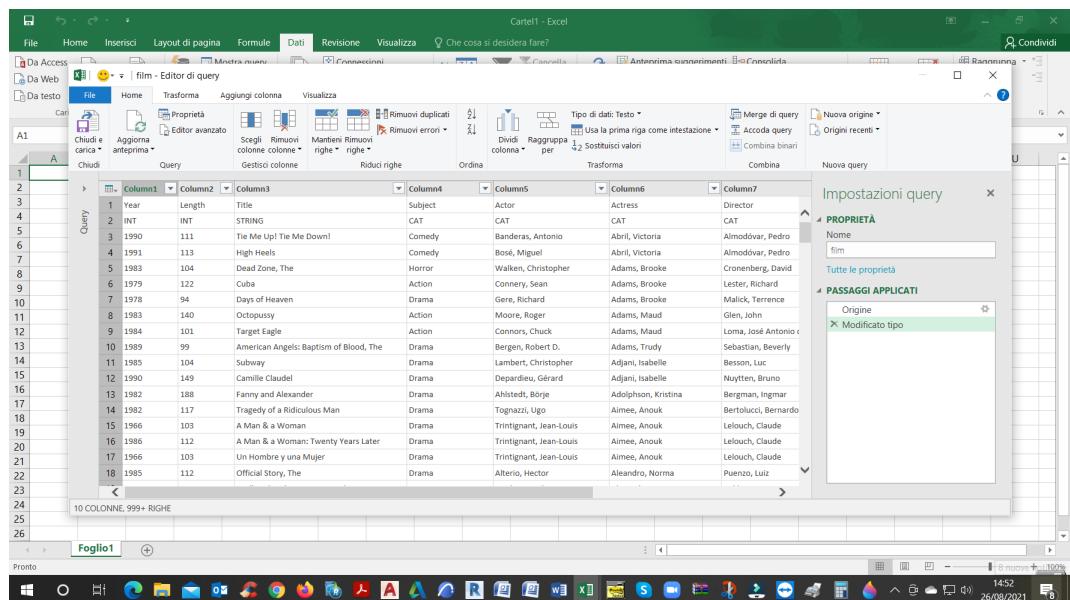


Figura 2.19 – Finestra di dialogo per l'import di dati da un file csv.

La funzione import è di fondamentale importanza perché facilita l'integrazione di dati provenienti da altri software. Più in generale, considerando che il formato csv è da considerarsi come uno dei formati più utilizzati per esportare database, si può dire che Excel facilita molto il lavoro con qualsiasi tipo di dati. Questo aspetto acquisisce molta importanza quando i dati da organizzare provengono da fonti esterne, ossia quando vengono forniti da altri operatori. Il modo in cui si salva un set di dati non solo è importante per facilitare la loro importazione, ma è anche fondamentale per evitare grossolani errori che influenzerebbero negativamente il risultato finale.

⁷i filtri e l'ordinamento delle colonne sono funzioni basilari di Excel, descritte nei prossimi paragrafi.

2.2.3 Le tabelle

Quando si inseriscono o si importano dei dati in Excel, è possibile organizzarli in *tabelle*. L'organizzazione in tabelle prevede alcuni notevoli vantaggi. Una tabella è considerata come un unico corpo comprendente tutti i dati, le intestazioni e, soprattutto, ogni dato che viene aggiunto in seguito. Nel caso di dati in costruzione ad esempio, ogni riga che viene implementata in un secondo momento entra a far parte automaticamente del corpo tabella. Altrettanto automaticamente si aggiorneranno le funzioni di riepilogo. Per creare una tabella è necessario cliccare su uno qualsiasi dei valori del set di dati che è stato appena importato e utilizzare la shortcut *Ctrl + T*. Excel evidenzierà tutti i dati che sono in collegamento con il dato da cui abbiamo iniziato⁸, e li considererà come facenti parte della tabella, chiedendoci conferma. È possibile effettuare una selezione diversa dei dati qualora sia necessario, e soprattutto è possibile definire se la nostra tabella presenta o meno delle intestazioni, o "headers". Le intestazioni si identificano nella prima riga in alto, e indicano il tipo di dato che è presente in ogni colonna. Utilizzare le headers permette chiaramente di identificare a prima vista il tipo di dato che si sta manipolando. Una volta creata una tabella si apre un nuovo pannello che permette di effettuare operazioni su di essa. Riporto alcune delle più interessanti e utili per questa tesi:

- è possibile **dare un nome** alla tabella. In questo modo la si può richiamare più facilmente nell'applicazione nelle successive funzioni, senza dover selezionare ogni volta tutto il set di dati.
- è possibile ricercare e rimuovere elementi duplicati all'interno di una o più colonne. La rimozione dei duplicati è un aspetto importantissimo per qualsiasi tipo di dato si stia analizzando, ed è uno dei primi controlli che si effettuano a monte di qualsiasi analisi.
- si possono creare delle tabelle Pivot⁹.

⁸semplicemente evidenziando tutti i dati che si trovano in continuità, ossia che non siano separati da nessuna cella vuota.

⁹Le tabelle Pivot sono delle tabelle secondarie che si creano a partire dalla prima. Esse hanno il compito di mostrare i dati da un'angolazione diversa, spesso raggruppando gli stessi in gruppi. Servono a dare uno sguardo generalizzato (mostrando valori medi e numero di occorrenze ad esempio), e si aggiornano automaticamente all'aggiornarsi della tabella principale da cui provengono. Le tabelle Pivot sono uno strumento molto utilizzato e molto potente per l'analisi dei dati. La loro trattazione

- si può attivare o disattivare la *Total Row*. Essa rappresenta un riepilogo visivo dei dati in tabella, automaticamente costruito sulla prima riga libera della tabella (quindi l'ultima verso il basso). Quando la total Row viene attivata essa mostra automaticamente la somma di tutti i valori della tabella. Con un semplice clic sul menù a tendina, però, è possibile cambiare informazione. Tra quelle disponibili ci sono: *valore medio*, *conta*, *conta numero*, *Max*, *Min*, *Deviazione Standard*¹⁰.

In ultimo, le tabelle diventano immediatamente riconoscibili perché lo sfondo delle celle assume una colorazione differente rispetto al bianco usuale, ed è possibile cambiare il colore nel medesimo menu Tabella (figura 2.20).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	name	mfr	type	calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups	rating	
2	100% Bran	N	C	70	4	1	130	10	5	6	280	25	3	0.33	68.402979		
3	100% Natural Bran	Q	C	120	3	5	15	2	8	135	0	3	1	1	33.983679		
4	All-Bran	K	C	70	4	1	260	9	7	5	320	25	3	1	0.33	59.425505	
5	All-Bran with Extra Fiber	K	C	50	4	0	140	14	8	0	330	25	3	1	0.5	93.704912	
6	Almond Delight	R	C	110	2	2	200	1	14	8	-1	25	3	1	0.75	34.384843	
7	Apple Cinnamon Cheerios	G	C	110	2	2	180	1.5	10.5	10	70	25	1	1	0.75	29.509541	
8	Apple Jacks	K	C	110	2	0	125	1	11	14	30	25	2	1	1	33.174094	
9	Basic 4	G	C	130	3	2	210	2	18	8	100	25	3	1.33	0.75	37.038562	
10	Bran Chex	R	C	90	2	1	200	4	15	6	125	25	1	1	0.67	49.120253	
11	Bran Flakes	P	C	90	3	0	210	5	13	5	190	25	3	1	0.67	53.313813	
12	Cap'n Crunch	Q	C	120	1	2	220	0	12	12	35	25	2	1	0.75	18.042851	
13	Cheerios	G	C	110	6	2	290	2	17	1	105	25	1	1	1.25	50.764999	
14	Cinnamon Toast Crunch	G	C	120	1	3	230	0	15	9	45	25	2	1	0.75	19.823573	
15	Custers	G	C	110	3	2	140	2	13	7	105	25	3	1	0.5	40.400208	
16	Crispy O's	G	C	110	1	1	180	0	12	13	55	25	2	1	1	23.773146	
17	Corn Chex	R	C	110	2	0	280	0	22	5	25	25	1	1	1	41.445019	
18	Corn Flakes	K	C	100	2	0	290	1	21	2	35	25	1	1	1	45.863324	
19	Corn Pops	K	C	110	1	0	90	1	13	12	20	25	2	1	1	35.782791	
20	Count Chocula	G	C	110	1	1	180	0	12	13	65	25	2	1	1	22.396513	
21	Cracklin' Oat Bran	K	C	110	3	3	140	4	10	7	160	25	3	1	0.5	40.448772	
22	Cream of Wheat (Quik)	N	H	100	3	0	80	1	21	0	-1	0	2	1	1	64.533816	
23	Crispix	K	C	110	2	0	220	1	21	3	30	25	3	1	1	46.895644	
24	Crispy Wheat & Raisins	G	C	100	2	1	140	2	11	10	120	25	3	1	0.75	36.176196	
25	Double Chex	R	C	100	2	0	190	1	18	5	80	25	3	1	0.75	44.330856	
26	Froot Loops	K	C	110	2	1	125	1	11	13	30	25	2	1	1	32.207582	
27	Frosted Flakes	K	C	110	1	0	200	1	14	11	25	25	1	1	0.75	31.435973	
28	Frosted Mini-Wheats	K	C	100	3	0	0	3	14	7	100	25	2	1	0.8	58.345141	

Figura 2.20 – Tabella di dati in Microsoft Excel.

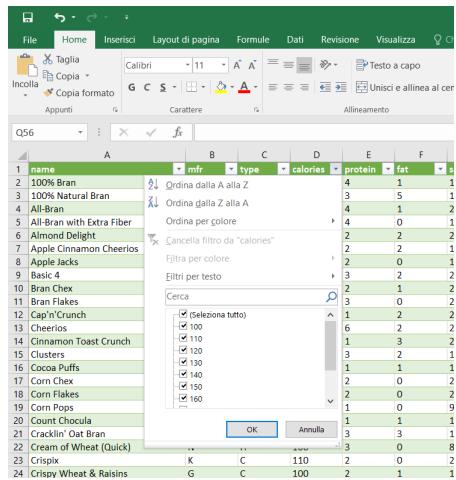
2.2.4 L'ordine dei dati e i filtri

In ogni tabella le celle delle intestazioni - o headers - si trasformano, acquisendo un menu a tendina, identificato da un triangolo rivolto verso il basso. Se si clicca su questo triangolo si apre un menù che ci permette di *ordinare* e *filtrare* i dati della colonna scelta. Si possono ordinare i dati in ordine alfabetico - qualora si tratti di testo - o in ordine crescente/decrescente - qualora si tratti di valori. Quando si ordina una colonna, Excel ordina automaticamente tutti i valori della tabella. Nello stesso menù si possono filtrare i dati, mostrando solamente quelli che ci interessano, semplicemente spuntando

tuttavia richiederebbe tempo e spazio, ed esula dallo scopo di questa tesi.

¹⁰descriverò dettagliatamente in seguito tutte queste funzioni.

la casella accanto al loro identificativo. Queste due opzioni offrono un'arma potente ed interessante: la possibilità di selezionare parte dei dati da una grande mole, per poterli analizzare e scovare più facilmente eventuali criticità (figura 2.21).



The screenshot shows a Microsoft Excel spreadsheet titled 'Cereals.xlsx'. The data consists of 24 rows of cereal information across columns A through F. The columns are labeled: name, mfr, type, calories, protein, and fat. A filter dialog box is open over the data, specifically for the 'calories' column. The dialog lists several filter options: 'Ordina dalla A alla Z', 'Ordina dalla Z alla A', 'Ordina per colore', 'Cancella filtro da "calories"', 'Filtra per colore', 'Filtri per testo', and 'Cerca'. Below these, there is a list of numerical values from 100 to 160, each preceded by a checkmark. At the bottom of the dialog are 'OK' and 'Annulla' buttons.

Figura 2.21 – Selezionare e filtrare dati in Excel.

2.2.5 Le funzioni

Se da un lato abbiamo visto che all'interno delle celle possono essere inseriti dati di vario tipo, in esse possono essere presenti anche delle *funzioni*. Una funzione è un'operazione che restituisce all'interno della cella in cui è inserita, un valore che proviene dall'interazione di due o più celle. Per inserire una funzione all'interno di una cella è necessario farla precedere dal segno "`=`". Ogni funzione si compone di un nome - univoco - seguito da una serie di parentesi tonde che contengono uno o più *argomenti*, generalmente separati da un punto e virgola. Perché una funzione sia valida è necessario che la sua sintassi sia correttamente riportata, ma non è sempre facile ricordarla, soprattutto con funzioni complesse o funzioni annidate¹¹. Per questo Excel offre un aiuto sottoforma di completamento automatico nel mentre si digita la funzione stessa. In alternativa è possibile ricercare la funzione dall'elenco di quelle disponibili: si aprirà una finestra di dialogo che permetterà l'inserimento passo passo degli argomenti. Per facilitarci ulteriormente, Excel raggruppa tutte le funzioni per tipologia, allo scopo di renderle più facilmente identificabili in caso di ricerca. Cliccando sul menu **Formule**, infatti, troviamo:

- *Usate di recente*: sono le ultime funzioni utilizzate.

¹¹le funzioni *annidate*, o *nested*, sono funzioni che hanno come argomento un'altra funzione.

- *Finanza*: sono le funzioni per calcoli finanziari e di gestione di somme di denaro. Alcune funzioni sono: *PMT* per il calcolo del pagamento di un mutuo, *PV* per il calcolo del valore presente e futuro di un investimento.
- *Logica*: comprende l'utilizzo degli operatori logici (E, O, NON, VERO, FALSO), ma anche la funzione SE.
- *Testo*: raggruppa le funzioni per trattare stringhe di testo. Alcuni esempi: *SOSTITUisci* che permette di selezionare parti di testo in una o più celle e sostituirle con un altro testo, *CONCATENA*, che unisce una o più stringhe, *IDENTICO*, che confronta due stringhe per valutare se sono uguali.
- *Data e ora*: sono le funzioni per trattare le celle in cui sia inserita una data o un'indicazione temporale in genere. Esempi: *ORA* restituisce la data e l'ora esatti in cui viene inserita, *NUM.SETTIMANA* restituisce il numero della settimana corrente dell'anno, *GIORNI* restituisce il numero di giorni che intercorrono tra due date.
- *Ricerca e riferimento*: raggruppa funzioni interessanti per creare indici all'interno di una tabella e per formare cross reference all'interno di tabelle.
- *Matematica e trigonometria*: contiene funzioni matematiche (*MCM* e *MCD* per il calcolo di minimo comune multiplo e massimo comun divisore, *ARROTONDA* per arrotondare un numero decimale) e trigonometriche (funzione *SEN*, funzione *COS*).
- *Statistiche* che contiene numerose funzioni statistiche come *MEDIA*, *MEDIANA* e *MODA*, *DEV.ST.P standard*, *TESTT*.

Quando si scrive la sintassi di una funzione, in essa si possono inserire valori numerici, che resteranno fissi, oppure riferimenti ad altre celle. In quest'ultimo caso, se cambia il valore interno della cella inserita in una funzione, automaticamente Excel aggiornerà il risultato finale di quella stessa funzione. Per comprendere meglio questa differenza ci riferiamo a queste due sintassi:

"=15 + 25"

"=B1 + B2"

come si può notare in figura 2.22, la prima funzione, inserita nella cella E1, (colorata di rosso) e la seconda funzione, inserita nella cella E2 (colorata di giallo), restituiscono lo stesso valore. Ma se cambiamo il valore della cella A1, soltanto la funzione che ha tra gli argomenti i riferimenti alle celle - e non i valori assoluti - vedrà aggiornarsi il proprio valore finale:

	A	B	C	D	E	F
1		15			40	
2		25			40	
3						
4						

Figura 2.22 – Le funzioni in Excel.

Questo semplice concetto è alla base del funzionamento logico di Excel: un foglio di calcolo è in costante aggiornamento ogni volta vengono aggiunti, rimossi o modificati dei dati.

È possibile copiare una funzione da una cella all'altra, ma occorre prestare attenzione ai riferimenti relativi della funzione stessa. Quando si copia¹² una funzione da una cella all'altra, infatti, Excel mantiene i riferimenti alle celle in essa contenuta in termini di distanza. I riferimenti della formula nella nuova posizione cioè, vengono ricercati misurando la stessa distanza, in ordine di celle, rispetto alla posizione originale. Ritornando all'esempio della funzione precedente: la funzione che contiene in essa i riferimenti alle celle B1 e B2 è posta nella cella E2. Se copiamo la medesima funzione nella cella E3, essa non farà più riferimento alle celle B1 e B2, ma alle celle C1 e C2, che sono "una cella in basso" rispetto alle precedenti. Quando vogliamo spostare la posizione di una funzione, mantenendo però fissi i riferimenti alle celle in essa contenute, è possibile inserire accanto ai riferimenti delle celle il simbolo \$: in questo modo i riferimenti interni alla funzione non si sposteranno. La figura 2.23 esplicita questo concetto.

Per descrivere come inserire correttamente una funzione riporto di seguito la dettagliata descrizione di alcune delle più comuni tra queste funzioni.

¹²o si sposta.

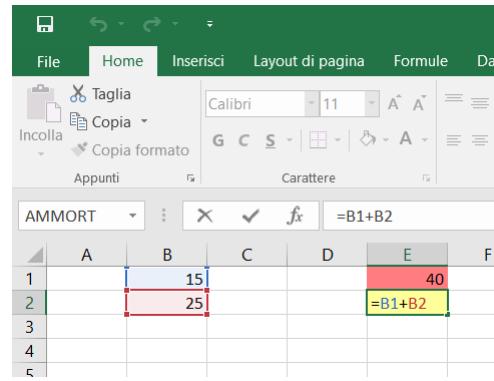


Figura 2.23 – Funzione in Excel e i riferimenti alle celle.

La funzione SOMMA

La funzione SOMMA permette di restituire, nella cella in cui essa è inserita, il valore che proviene dalla somma di tutti i suoi argomenti. Gli argomenti possono essere inseriti come singoli valori provenienti da singole celle, o come intervalli. Per inserire la funzione SOMMA occorre digitare questa sintassi:

"=SOMMA(primo valore;secondo valore;terzo valore; ... ;ultimo valore)"

quando i dati da sommare sono presenti su celle adiacenti (sulla stessa linea o sulla stessa colonna), è possibile selezionare un intervallo, e la formula si modifica in questo modo:

"=SOMMA(primo valore:ultimo valore)"

Nella figura (2.24) mostro un caso reale di utilizzo della funzione SOMMA.

A	B	A	B
10		1	10
15		2	15
17		3	17
=SOMMA(A1;A2;A3)		4	42
5		5	
6		6	
7		7	
8		8	
9		9	
10		10	

Figura 2.24 – La funzione SOMMA.

Le funzioni MEDIA, MINIMO e MASSIMO

Queste tre funzioni hanno tutte la medesima struttura¹³. La funzione MEDIA restituisce il valore medio di un intervallo di valori, le funzioni MINIMO e MASSIMO restituiscono il valore più basso e il valore più alto all'interno di un intervallo di dati.

Le loro sintassi sono, rispettivamente:

`"=MEDIA(primo valore:ultimo valore)"`

`"=MINIMO(primo valore:ultimo valore)"`

`"=MASSIMO(primo valore:ultimo valore)"`

In figura 2.25 mostro un caso reale di utilizzo delle funzioni.

The screenshot shows a Microsoft Excel spreadsheet with the following data in rows 1 through 10:

	A	B	C	D	E	F
1	3		3		3	
2	7		7		7	
3	4		4		4	
4	5		5		5	
5	5		5		5	
6	6		6		6	
7	7		7		7	
8						
9	MEDIA		MIN		MAX	
10	5,285714		3		7	
11						

Figura 2.25 – Le funzioni MEDIA, MINIMO e MASSIMO.

La funzione TESTT

La funzione TESTT è una funzione statistica che esegue il *test di Student* su due popolazioni, per valutare se i loro valori medi siano differenti e che questa differenza non sia dovuta al caso. È un test molto utilizzato quando occorre confrontare due popolazioni di dati. Questa funzione si compone di quattro argomenti, alcuni dei quali sono obbligati, vale a dire che possono assumere solamente alcuni valori, che Excel stesso suggerisce durante la compilazione. La sintassi per la funzione TESTT è la seguente:

¹³si formano cioè inserendo gli argomenti nel medesimo ordine.

$"=TESTT(primo\ array; secondo\ array; tipo\ di\ coda\ della\ distribuzione; tipo\ di\ varianza)"$

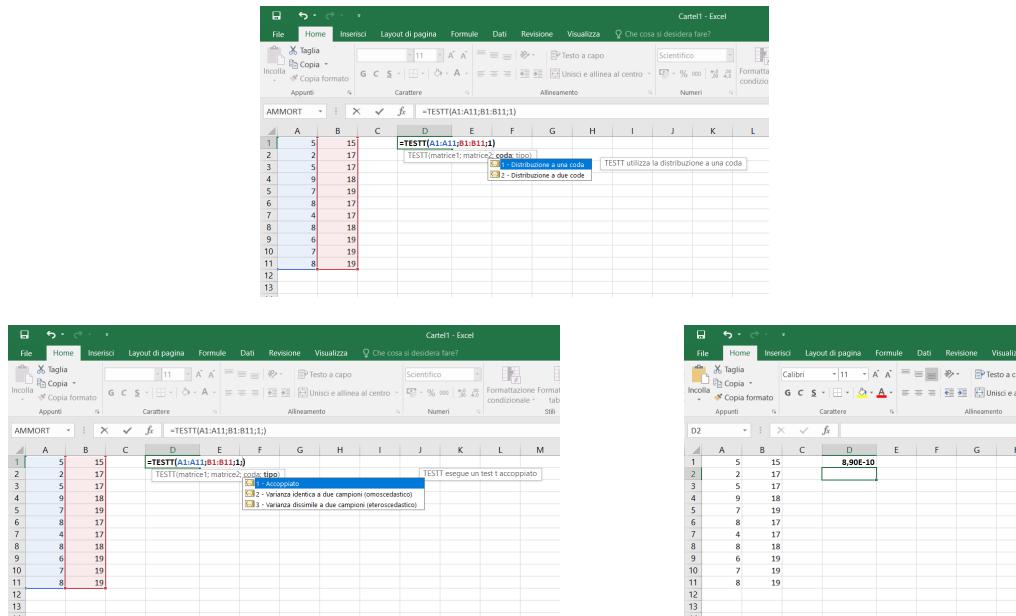


Figura 2.26 – Funzione di Excel per effettuare il Test di Student su due popolazioni di dati.

Come è possibile notare, anche grazie all'esempio in figura 2.26, il terzo e il quarto argomento possono assumere rispettivamente valori di 1 - "Distribuzione a una coda" o 2 - "Distribuzione a due code" e di 1 - "Accoppiato", 2 - "Varianza identica a due campioni (omoscedastico) o 3 - "Varianza dissimile a due campioni (eteroscedastico)". Se l'utente prova ad inserire un valore diverso da quelli previsti Excel non riconosce la sintassi della formula e restituisce un avviso di errore.

2.2.6 Creazione di grafici con Excel

In questo paragrafo descrivo brevemente le possibilità offerte da Excel per la creazione di grafici¹⁴. La creazione di grafici è perfettamente integrata con i dati presenti all'interno del foglio di calcolo. Per creare un grafico è sufficiente cliccare sull'icona

¹⁴Excel offre molte soluzioni per la creazione di grafici, che integra nella pagina come oggetti. Il livello di personalizzazione è abbastanza approfondito, così come le tipologie di grafici possibili. Tuttavia le soluzioni offerte non raggiungono livelli di gradevolezza estetica paragonabile ad altri software, e i grafici sono pensati per appartenere alla pagina su cui risiedono i dati, e non per essere esportati altrove. Per questi motivi ho scelto di non avvalermi della funzione di creazione grafici offerta da Excel, pur non potendo esimermi dal citarla.

corrispondente e seguire le istruzioni della finestra di dialogo che si apre, che ci chiedono di selezionare i dati da plottare. Alternativamente è possibile selezionare i dati e successivamente cliccare sull'icona del grafico corrispondente. Le tipologie di grafici possibili sono numerose, tra cui:

- *grafici a barre* (o *colonne*) sia 2D che 3D, sia orizzontali che verticali.
- *grafici a linea* con o senza punti, sia 2D che 3D.
- *grafici a torta*, sia 2D che 3D.
- istogrammi.
- grafici a dispersione.

Una volta creato il grafico è possibile personalizzarlo cambiando, ad esempio, il nome delle serie di dati selezionate, il colore del grafico stesso, il titolo e/o la leggenda, la scala di riferimento. I grafici si aggiornano ogni qual volta i dati di partenza vengono modificati, eliminati o implementati inserendone dei nuovi.

2.3 Python

In questo capitolo analizzerò nel dettaglio l'utilizzo dei pacchetti *pandas* e *seaborn* di Python. Le operazioni sono state tutte sviluppate in ambiente Jupyter (cfr. 1.3.2).

2.3.1 caricare i pacchetti in Python

Per poter iniziare a lavorare con Python è necessario importare i pacchetti di interesse. La figura 2.27 rappresenta le prime righe di un ambiente di lavoro di Python, dove vengono importati i pacchetti necessari per lavorare, ed è proprio l'inizio dell'ambiente di lavoro con cui verranno preparati i grafici per la rappresentazione dei dati provenienti da GA.

Per importare un pacchetto è sufficiente la seguente sintassi:

```
import nome del pacchetto as alias del pacchetto
```

l'alias è un nome, più breve del nome reale, che attribuiamo al pacchetto per richiamarlo più facilmente nei comandi successivi. Il linguaggio Python, infatti, prevede

che per richiamare un metodo è necessario farlo precedere dal nome del pacchetto cui essa appartiene. Quando il nome del pacchetto è troppo lungo, la scrittura di ripetute operazioni può rendere disordinato il testo. Per questo motivo è largamente diffuso l'utilizzo di alias, sebbene non sia strettamente necessario¹⁵.

```
In [1]: import os
import numpy as np
import matplotlib.pyplot as plt
import time
import pandas as pd
import seaborn as sns
```

Figura 2.27 – Import dei pacchetti.

Rifacendoci alla figura 2.27, leggiamo e descriviamo tutti i pacchetti importati, definendo le loro caratteristiche principali. In questo elenco mancano, volutamente, *pandas* e *seaborn*, che sono descritti a fondo nei due paragrafi seguenti.

- **import os:** il pacchetto *os* è un pacchetto di gestione delle cartelle di sistema. Viene importato per poter "navigare" attraverso lo spazio fisico del nostro hard disk. Risulta fondamentale ogni volta che vogliamo recuperare dei dati provenienti da un file che si trovi in una determinata posizione della memoria fisica del computer, oltre che per scegliere in quale cartella esportare i grafici.
- **import numpy as np:** il pacchetto *numpy* supporta la gestione di grandi matrici di dati, offrendo la possibilità di operare su di esse. Inoltre offre numerose funzioni matematiche, qualora l'operatore necessiti di effettuare calcoli sui dati importati.
- **import matplotlib.pyplot as plt:** come già accennato il pacchetto *matplotlib* è il primo e più completo pacchetto per la creazione di grafici in Python. È bene importare *matplotlib* perché rappresenta la base su cui *seaborn* è stato implementato e costruito, e conserva alcune funzionalità specifiche che non sono riscontrabili in *seaborn*. Inoltre, seppure in numero molto limitato, ci sono grafici che *seaborn* non è in grado di offrire (i grafici a torta).

¹⁵è interessante notare come, tecnicamente, l'alias può essere qualsiasi sequenza di lettere ma, di fatto, si utilizzano sempre le stesse. Il motivo fondamentale è che, utilizzando gli alias comunemente noti, ognuno può leggere e comprendere un codice scritto da altri. Così *pandas* è sempre abbreviato in *pd*, *matplotlib.pyplot* diventa *plt*, *seaborn* è *sns*, semplicemente perché, negli anni, è diventato comune definirli in questo modo.

- `import statistics`: come suggerisce il nome, il pacchetto statistic offre numerose soluzioni per effettuare test statistici di alto livello che possono rendersi necessari per lo studio dei dati a nostra disposizione, come ad esempio il *test di student* che abbiamo visto in Excel.
- `import time`: è un pacchetto che permette di valutare il tempo impiegato da Python per far operare una serie di istruzioni.

Ogni volta che vogliamo operare un comando di un determinato pacchetto richiamiamo il pacchetto stesso - anche tramite il suo alias - e lo facciamo seguire da un punto, a sua volta seguito dal nome del comando, una serie di parentesi tonde e gli eventuali valori interni che il comando deve trattare.

Il pacchetto pandas

Il pacchetto *pandas* permette di importare e gestire dati da numerose fonti. L'unità fondamentale su cui lavora il pacchetto *pandas* è il **dataset**, ossia l'insieme ordinato di dati visualizzabile in maniera tabellare ed interrogabile secondo i comandi interni al pacchetto¹⁶. Iniziamo ad utilizzare *pandas* per importare i dati provenienti da un foglio Excel, utilizzando il comando *read_excel*:

```
Dataset_GA = pd.read_excel('dati_GA.xlsx', sheet_name = 'Sheet_1',
                           usecols = 'A:D', skiprows = 3, nrows = 49)
```

In dettaglio:

1. `Dataset_GA` è il nome scelto per la variabile, ovverosia il nome da dare al nuovo Dataset che conterrà i dati provenienti da Excel.
2. `pd.read_excel` è il comando per importare i dati dal file Excel.

¹⁶il pacchetto pandas utilizza il termine *Dataframe* che però tecnicamente indica la forma con cui i dati sono organizzati - il termine *frame* significa "cornice". L'utilizzo dei due termini è spesso interscambiabile, ma ho preferito essere più rigoroso. Il pacchetto pandas, d'altro canto, opera organizzando, relazionando e selezionando i dati che gli vengono forniti, modificando di volta in volta l'aspetto - (la cornice) - con cui questi dati vengono mostrati. In altre parole, a stretto rigore di logica le operazioni del pacchetto non sono quelle di **costruire** un set di dati, ma di **organizzare dati già esistenti** organizzandoli, appunto, in un *frame*. Questione puramente semantica ma, a mio avviso, importante.

3. 'dati_GA.xlsx' è il nome del file Excel da cui importare i dati. È tra apici per identificarlo come stringa di testo¹⁷.
4. `sheet_name = 'Sheet_1'` identifica il foglio a cui va fatto riferimento per recuperare i file.
5. `usecols = 'A:D'` indica le colonne di Excel da cui importare i dati. È utile quando sono presenti dati non necessari, o quando si vuole analizzare solamente una parte di quelli disponibili. Le lettere maiuscole ('A:D') fanno riferimento alle colonne di Excel, sottoforma di intervallo, nella notazione tipica del foglio di calcolo.
6. `skiprows = 3`: con questo comando indichiamo quante righe occorre saltare prima di cominciare a importare i dati. Questa impostazione risulta necessaria quando nel foglio Excel sono presenti intestazioni che occupano, generalmente, le prime righe del foglio.
7. `nrows = 49`: con questo comando si indica il numero di righe da importare.

Il risultato è un dataset ordinato, che possiamo richiamare semplicemente scrivendo il nome che gli abbiamo assegnato. Così facendo il dataset ci viene proposto a schermo. Python ci restituisce dieci valori, dividendoli tra quelli iniziali e quelli finali se il dataset contiene più di dieci occorrenze, indicando in basso a sinistra il suo shape¹⁸. La prima colonna a sinistra, in grassetto, rappresenta la colonna *indice* del dataset, ossia la numerazione ordinale per identificare il numero di ogni occorrenza, (ma è possibile assegnare anche degli identificativi testuali). La prima riga, parimenti in grassetto, caratterizza il tipo di dato presente in ogni colonna.

Principali operazioni sul dataset

Per gli esempi di questo paragrafo mi avvalgo di un dataset già presente nelle librerie di Python: il dataset "*iris*", che contiene i dati di caratterizzazione di alcune specie di fiori (figura 2.28).

¹⁷la stringa di testo rappresenta, in realtà, il *percorso* da seguire per indirizzare al file Excel. Se è presente solo il nome, come in questo caso, il file Excel si trova nella stessa cartella in cui è presente il file dello script.

¹⁸ossia la forma. Una matrice si indica con due valori, secondo questa sintassi: *numero di righe x numero di colonne*.

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

150 rows x 5 columns

Figura 2.28 – Il dataset "iris".

Importare una serie di dati implica quasi necessariamente la necessità di modificarli, per i più svariati motivi. Quando, come nel nostro caso, i dati vengono importati dopo un passaggio in Excel, è possibile che essi siano già stati sufficientemente organizzati da non richiedere alcune ulteriori modifiche, ma è necessario conoscere le operazioni per interrogare e manipolare un dataset. Elenco di seguito alcune delle più utilizzate.

Interrogare un dataset Con il comando `dataset.head(n)` (cfr. figura 2.29) è possibile visualizzare le prime n righe di un dataset. Si può voler visualizzare un numero determinato di occorrenze nel dataset, a seconda della loro posizione o a seconda di alcune loro caratteristiche.

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa
10	5.4	3.7	1.5	0.2	setosa
11	4.8	3.4	1.6	0.2	setosa
12	4.8	3.0	1.4	0.1	setosa
13	4.3	3.0	1.1	0.1	setosa
14	5.8	4.0	1.2	0.2	setosa

	sepal_length	sepal_width	petal_length	petal_width	species
Out[5]:	5.1	3.5	1.4	0.2	setosa

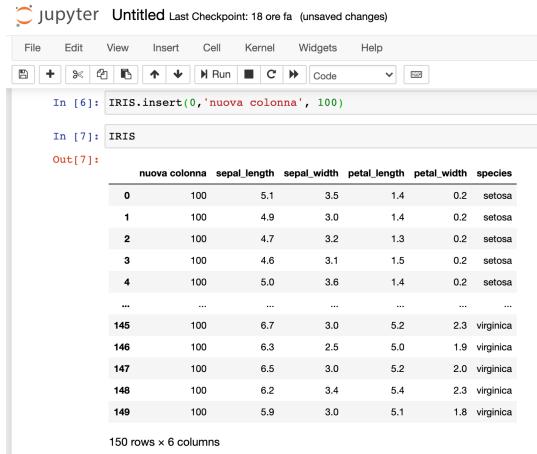
Name: 0, dtype: object

Figura 2.30 – Il comando `dataset.iloc[]`.**Figura 2.29** – Il comando `dataset.head(n)`.

Il comando `dataset.iloc[]`, invece, permette di interrogare il dataset in modo che ci restituisca una riga scelta in funzione della sua posizione ordinale (figura 2.30)¹⁹.

¹⁹la posizione 0 indicata nell'esempio rappresenta la *prima* posizione del dataset. Python, come diversi altri linguaggi di programmazione, inizia a contare infatti proprio da 0.

Modificare un dataset Con la sintassi `dataset.insert(0, 'nuova colonna', [valore])` si aggiunge una colonna al dataset, scegliendone la posizione e i valori (figura 2.31).

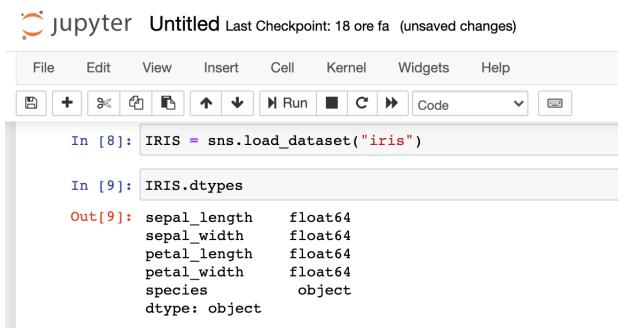


```
jupyter Untitled Last Checkpoint: 18 ore fa (unsaved changes)
File Edit View Insert Cell Kernel Widgets Help
In [6]: IRIS.insert(0,'nuova colonna', 100)
In [7]: IRIS
Out[7]:
   nuova colonna  sepal_length  sepal_width  petal_length  petal_width  species
0             100         5.1          3.5         1.4          0.2    setosa
1             100         4.9          3.0         1.4          0.2    setosa
2             100         4.7          3.2         1.3          0.2    setosa
3             100         4.6          3.1         1.5          0.2    setosa
4             100         5.0          3.6         1.4          0.2    setosa
...
145            100         6.7          3.0         5.2          2.3  virginica
146            100         6.3          2.5         5.0          1.9  virginica
147            100         6.5          3.0         5.2          2.0  virginica
148            100         6.2          3.4         5.4          2.3  virginica
149            100         5.9          3.0         5.1          1.8  virginica
150 rows x 6 columns
```

Figura 2.31 – Il comando `dataset.insert`.

`dataset.dropna()` rimuove tutte le righe che hanno un dato mancante, mentre il comando `dataset.fillna(valore)` sostituisce tutti i dati mancanti con valore scelto dall'operatore. `dataset.drop_duplicates()`, infine, elimina le occorrenze che esistono in duplicato.

Ottenere informazioni sul dataset `dataset.dtypes` ci mostra le tipologie di dati in ogni colonna, ossia se sono numeri, numeri decimali, stringhe, percentuali, etc... (figura 2.32).



```
jupyter Untitled Last Checkpoint: 18 ore fa (unsaved changes)
File Edit View Insert Cell Kernel Widgets Help
In [8]: IRIS = sns.load_dataset("iris")
In [9]: IRIS.dtypes
Out[9]:
sepal_length    float64
sepal_width     float64
petal_length    float64
petal_width     float64
species        object
dtype: object
```

Figura 2.32 – Il comando `dataset.dtypes`.

`dataset.max()` e `dataset.min()` restituiscono il valore più alto e il valore più basso di ogni colonna del dataset.

`dataset.sum()` somma tutti i valori del dataset colonna per colonna e restituisce i risultati. `dataset.std()` ci mostra il valore della deviazione standard dei dati, colonna per colonna²⁰ (figura 2.33).

In [11]:	<code>IRIS.min()</code>	In [13]:	<code>IRIS.sum()</code>
Out[11]:	<pre>sepal_length 4.3 sepal_width 2.0 petal_length 1.0 petal_width 0.1 dtype: float64</pre>	Out[13]:	<pre>sepal_length 876.5 sepal_width 458.6 petal_length 563.7 petal_width 179.9 dtype: float64</pre>
In [12]:	<code>IRIS.max()</code>	In [14]:	<code>IRIS.std()</code>
Out[12]:	<pre>sepal_length 7.9 sepal_width 4.4 petal_length 6.9 petal_width 2.5 dtype: float64</pre>	Out[14]:	<pre>sepal_length 0.828066 sepal_width 0.435866 petal_length 1.765298 petal_width 0.762238 dtype: float64</pre>

Figura 2.33 – I comandi `min()`, `max()`, `sum()` e `std()`.

Operazioni con più dataset Per descrivere queste operazioni utilizzerò un dataset appositamente creato.

`dataset.append(dataset2)`: questo comando permettere di aggiungere le occorrenze del secondo dataset alla coda del precedente (figura 2.34).

In [33]:	<code>dataset1</code>	In [35]:	<code>newdataset = dataset1.append(dataset2)</code>																														
Out[33]:	<table border="1"> <thead> <tr> <th></th> <th>Title</th> <th>seconds</th> <th>Year</th> <th>Album</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Love Me Do</td> <td>139</td> <td>1962</td> <td>Juvenilia</td> </tr> <tr> <td>1</td> <td>P. S. I Love You</td> <td>121</td> <td>1962</td> <td>Juvenilia</td> </tr> <tr> <td>2</td> <td>Please Please Me</td> <td>117</td> <td>1962</td> <td>Juvenilia</td> </tr> <tr> <td>3</td> <td>Ask Me Why</td> <td>144</td> <td>1962</td> <td>Juvenilia</td> </tr> <tr> <td>4</td> <td>There's A Place</td> <td>108</td> <td>1963</td> <td>Juvenilia</td> </tr> </tbody> </table>		Title	seconds	Year	Album	0	Love Me Do	139	1962	Juvenilia	1	P. S. I Love You	121	1962	Juvenilia	2	Please Please Me	117	1962	Juvenilia	3	Ask Me Why	144	1962	Juvenilia	4	There's A Place	108	1963	Juvenilia	In [36]:	<code>newdataset</code>
	Title	seconds	Year	Album																													
0	Love Me Do	139	1962	Juvenilia																													
1	P. S. I Love You	121	1962	Juvenilia																													
2	Please Please Me	117	1962	Juvenilia																													
3	Ask Me Why	144	1962	Juvenilia																													
4	There's A Place	108	1963	Juvenilia																													
In [34]:	<code>dataset2</code>	Out[36]:	<table border="1"> <thead> <tr> <th></th> <th>Title</th> <th>seconds</th> <th>Year</th> <th>Album</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>I Saw Her Standing There</td> <td>171</td> <td>1963</td> <td>Juvenilia</td> </tr> <tr> <td>6</td> <td>A Taste of Honey</td> <td>120</td> <td>1963</td> <td>Juvenilia</td> </tr> </tbody> </table>		Title	seconds	Year	Album	5	I Saw Her Standing There	171	1963	Juvenilia	6	A Taste of Honey	120	1963	Juvenilia															
	Title	seconds	Year	Album																													
5	I Saw Her Standing There	171	1963	Juvenilia																													
6	A Taste of Honey	120	1963	Juvenilia																													
Out[34]:	<table border="1"> <thead> <tr> <th></th> <th>Title</th> <th>seconds</th> <th>Year</th> <th>Album</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>I Saw Her Standing There</td> <td>171</td> <td>1963</td> <td>Juvenilia</td> </tr> <tr> <td>6</td> <td>A Taste of Honey</td> <td>120</td> <td>1963</td> <td>Juvenilia</td> </tr> </tbody> </table>		Title	seconds	Year	Album	5	I Saw Her Standing There	171	1963	Juvenilia	6	A Taste of Honey	120	1963	Juvenilia																	
	Title	seconds	Year	Album																													
5	I Saw Her Standing There	171	1963	Juvenilia																													
6	A Taste of Honey	120	1963	Juvenilia																													

Figura 2.34 – Il comando `dataset.append()`.

`dataset.join(dataset2)`: questo comando permettere di unire i due dataset, ossia di inserire le informazioni (colonne) esistenti sul secondo dataset al primo, collegandole ai risultati in base al loro indice (figura 2.35).

²⁰per queste ultime quattro operazioni ho eliminato dal dataset l'ultima colonna, che contiene stringhe di testo, dove non è ovviamente possibile effettuare le operazioni matematiche descritte.

In [89]:	Dataset1	In [93]:	Dataset2	In [94]:	Dataset1.join(Dataset2)
Out[89]:		Out[93]:		Out[94]:	
	Title seconds Year Album		stars		Title seconds Year Album stars
	0 Love Me Do 139 1962 Juvenilia	0 5 of 5		0	Love Me Do 139 1962 Juvenilia 5 of 5
	1 P. S. I Love You 121 1962 Juvenilia	1 4 of 5		1	P. S. I Love You 121 1962 Juvenilia 4 of 5
	2 Please Please Me 117 1962 Juvenilia	2 3 of 5		2	Please Please Me 117 1962 Juvenilia 3 of 5
	3 Ask Me Why 144 1962 Juvenilia	3 3 of 5		3	Ask Me Why 144 1962 Juvenilia 3 of 5
	4 There's A Place 108 1963 Juvenilia	4 3 of 5		4	There's A Place 108 1963 Juvenilia 3 of 5
	5 I Saw Her Standing There 171 1963 Juvenilia	5 4 of 5		5	I Saw Her Standing There 171 1963 Juvenilia 4 of 5
	6 A Taste of Honey 120 1963 Juvenilia	6 4 of 5		6	A Taste of Honey 120 1963 Juvenilia 4 of 5

Figura 2.35 – Il comando dataset.join().

Esportare un dataset Le due sintassi che seguono permettono di esportare il dataset in formato csv o Excel, rispettivamente.

```
dataset.to_csv(), dataset.to_excel()
```

Questi comandi sono solamente una parte di quelli possibili, la cui estesa documentazione è presente sul sito web del pacchetto *pandas* [panb].

2.3.2 Il pacchetto seaborn

seaborn suddivide i plot in categorie: *relational plots*, *distribution plots*, *categorical plots*, *regression plot* e *matrix plot*. Di seguito le specifiche per ognuna di queste categorie, con uno o più esempi per ognuna di esse. Le immagini sono prese dal sito web di *seaborn*²¹, che propone dettagliati esempi utilizzando dei dataset già esistenti.

relational plots

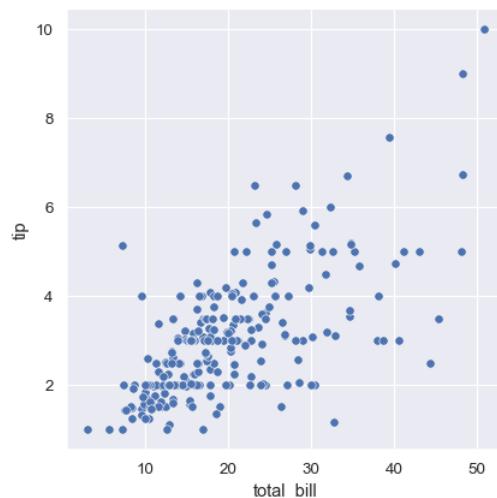
I plot relazionali sono plot in cui due gruppi valori vengono messi in relazione tra di loro, e ogni coppia rappresenta un punto del grafico. Sono utili per confrontare delle variabili o per gestire delle linee temporali, ossia quando i due gruppi di valori da plottare sono rappresentati da una sequenza temporale (ore, giorni, mesi o anni) e da un valore numerico, rispettivamente. Mostro in dettaglio un grafico a punti (scatter) e un grafico a linea.

scatterplot Il grafico relazionale a punti, o *scatterplot*, permette di visualizzare i singoli punti come formati da due coppie di valori, appartenenti a due colonne diverse del dataset. In questo esempio i dati sono presi dal dataset *tips*, che raccoglie i valori delle mance offerte dagli avventori di un ristorante in relazione a quanto speso (figure 2.36 e 2.37).

²¹<https://seaborn.pydata.org/examples/index.html>

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4
...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

244 rows × 7 columns

Figura 2.36 – Il Dataset "tips".**Figura 2.37** – Scatterplot di seaborn.

il comando per ottenerla è il seguente:

```
sns.relplot(x="total_bill", y="tip", data=tips)
```

dove `sns` è l'alias per *seaborn*, `relplot` è il comando per costruire questo tipo di grafico, `x="total_bill"` e `y="tip"` sono i comandi con cui si stabilisce che i valori dell'asse x debbano essere presi dalla colonna del dataset chiamata *total_bill*, e quelli dell'asse y dalla colonna *tip*. Infine, `data=tips` indica il dataset da cui reperire le informazioni. Le possibilità di personalizzazione sono molteplici: si possono cambiare i punti per forma e/o colore, si può cambiare lo sfondo del grafico, si può aggiungere un titolo, si possono cambiare le scale di riferimento²².

lineplot Il grafico a linea viene generalmente utilizzato per mostrare l'andamento di un valore (asse delle y) nel tempo (asse delle x) (figura 2.38).

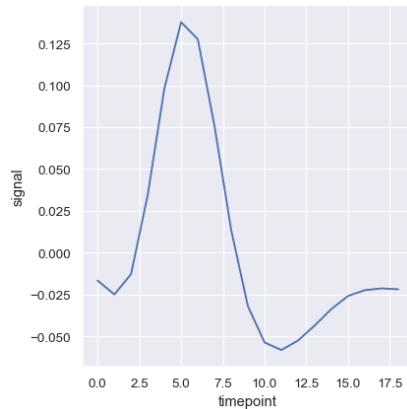


Figura 2.38 – Lineplot di seaborn.

```
sns.relplot(x="timepoint", y="signal", ci=None, kind="line", data=fmri)
```

la sintassi è analoga al grafico precedente, con l'aggiunta del parametro `kind="line"` che indica proprio la tipologia a linea. Possiamo notare come sull'asse delle x sia presente una linea temporale, che proviene dalla colonna "*timepoint*" del dataset chiamato "*fmri*", mentre sull'asse delle y è presente un valore numerico. Tra le varie personalizzazioni di questo tipo di grafico è possibile plottare un intervallo di confidenza che si sovrappone alla linea centrale, a indicare tutti i valori della colonna in esame, evidenziando quello medio.

²²per questo grafico e per tutti i seguenti di questo capitolo mostrerò le applicazioni di base. Nell'ultimo capitolo, invece, costruirò dei grafici con un più elevato grado di personalizzazione.

distribution plots

I plot di distribuzione mostrano, appunto, la distribuzione di una serie di eventi, che sono tipicamente recuperati da una colonna di un dataset.

displot Il grafico *displot* crea una distribuzione a barre, la più semplice possibile (figura 2.39):

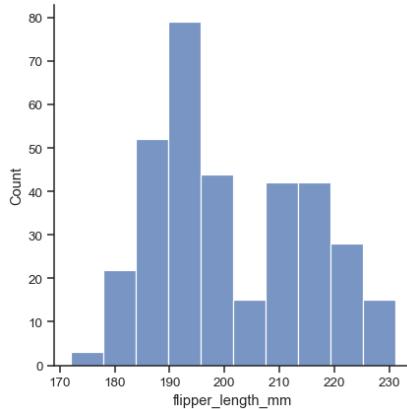


Figura 2.39 – Displot di seaborn.

la sintassi è la seguente:

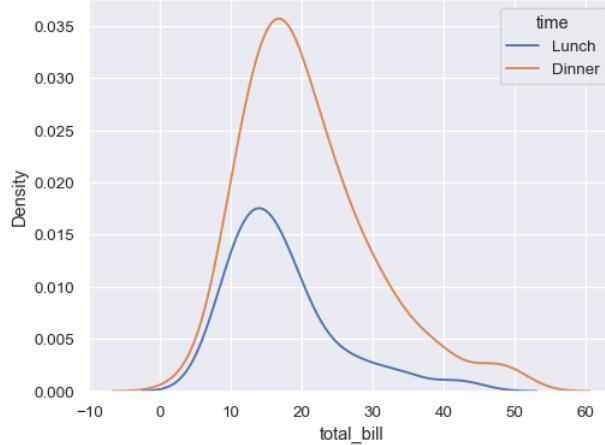
```
sns.displot(data=penguins, x="flipper_length_mm")
```

il dataset in oggetto è nominato "*penguins*", e raccoglie dati biometrici di diverse popolazioni di pinguini (grandezza delle pinne, altezza, colore della pelliccia). I valori sono presi dalla colonna "*flipper_length_mm*" che misura, appunto, la lunghezza delle pinne. In questo tipo di grafico è possibile cambiare colore delle barre, la loro larghezza (binning), lo sfondo e gli assi.

kdeplot Il grafico *kdeplot* plotta la distribuzione utilizzando una *kde*²³ (figura 2.40) con una sintassi ormai nota. È possibile perfezionare questo tipo di plot cambiando il colore della curva, riempiendo l'area sottesa alla curva, cambiando lo sfondo del grafico.

```
sns.kdeplot(data=tips, x="total_bill")
```

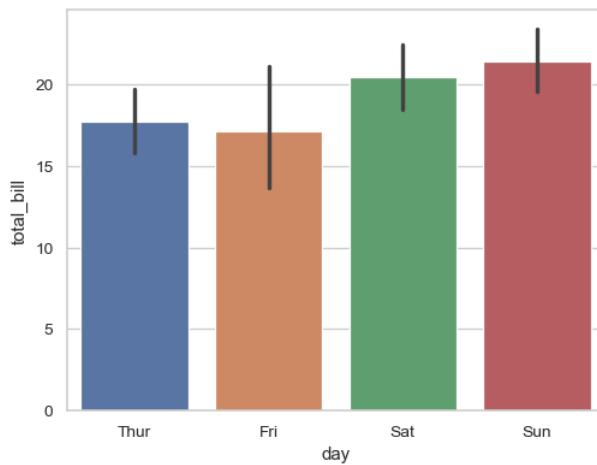
²³kernel density distribution cfr. 1.3.3

**Figura 2.40** – kde plot di seaborn.

categorical plots

I plot "categorici" raggruppano i dati in categorie, mostrando valori cumulativi (somma) o medi. Sono riconoscibili perché sono generalmente presenti le barre di errore, che indicano l'ampiezza della distribuzione dei dati appartenenti ad ogni singola categoria.

barplot Uno dei plot categorici più comuni è la rappresentazione a barre, o *bar plot*. Generalmente verticale - ma può essere anche orizzontale - mostra i dati provenienti da una colonna di un dataset e li suddivide in più barre a seconda dell'appartenenza a una categoria.

**Figura 2.41** – Barplot di seaborn.

```
sns.barplot(x="day", y="total_bill", data=tips)
```

L'asse delle x ha per valori le categorie esistenti nella colonna "*day*" del dataset. Python raggruppa i valori per ogni giorno della settimana e crea altrettante barre nel

grafico, di colore diverso (figura 2.41). È possibile personalizzare questo grafico cambiando i colori delle barre, modificando il valore della barra di errore²⁴, modificando lo sfondo del grafico.

regression plots

I plot di regressione sono dei plot che, identificato uno scatter di punti con delle coppie di valori, sovrappone a questi una retta che si definisce *di regressione*. La retta di regressione rappresenta il tentativo di definire una relazione matematica tra due variabili, una dipendente e una indipendente.

regplot È il tipo di grafico più semplice e immediato per riconoscere eventuali correlazioni tra due set di dati.

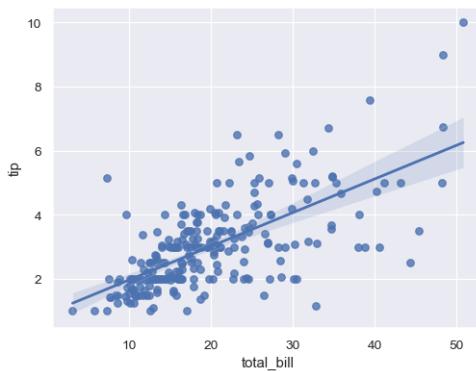


Figura 2.42 – regplot di seaborn.

```
sns.regplot(x="total_bill", y="tip", data=tips)
```

è chiaramente visibile la linea di regressione, anche detta di tendenza, che rappresenta la correlazione matematica più idonea a rappresentare la relazione esistente tra i dati dell'asse x e i dati dell'asse y (figura 2.42).

matrix plot

I plot a matrice mostrano, appunto, una matrice di dati di forma variabile, colorando ognuno a seconda del valore, preso all'interno di una scala di riferimento. La scala è tarata dal valore minimo al valore massimo presente nella matrice, oppure secondo criteri stabiliti dall'utente (ad esempio da zero a uno, a prescindere che il valore zero e il valore uno siano presenti).

²⁴scegliendo i vari metodi statistici per calcolarla.

heatmap Il nome *heatmap* sta per *mappa di calore* e richiama l'idea delle immagini utilizzate dai metereologi per indicare le temperature nelle varie regioni. Cambiando i colori si ottengono i codici di riconoscimento per le altezze dei rilievi montuosi (scala dal giallo al verde) o le profondità dei mari (scala dal ciano al nero). In realtà la *heatmap* è molto utilizzata anche in altri ambiti, soprattutto scientifici, dovunque si debbano rappresentare contemporaneamente molti valori di intensità.

```
sns.heatmap(uniform_data)
```

la sintassi è molto semplice, ma per chiarezza è bene osservare, oltre al grafico in sé (figura 2.43) anche la matrice di dati che ha generato il grafico (figura 2.44):

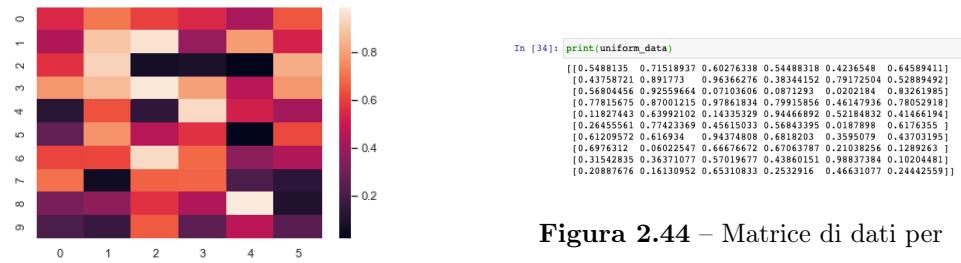


Figura 2.44 – Matrice di dati per il grafico heatmap.

Figura 2.43 – heatmap di seaborn.

dal raffronto delle due figure è possibile paragonare la forma della matrice (6 righe x 10 colonne) con quella dei rettangoli costituenti il grafico, e osservare come i valori più alti siano in corrispondenza del colore più scuro: la scala di riferimento alla destra è essenziale per comprendere questo tipo di rappresentazione. È possibile personalizzare questo tipo di grafico cambiando i colori, invertendo la scala di riferimento a destra (dal valore più alto a quello più basso), e indicando in ogni singolo rettangolo i valori numerici corrispondenti.

Capitolo 3

Un caso di studio

In questo capitolo mostrerò dettagliatamente un caso di studio, applicando il flusso di operazioni finora descritte ad un sito di e-commerce italiano.

3.1 Descrizione del sito web

Il sito web utilizzato per questo capitolo appartiene a un negozio di articoli per l'illuminazione: "Luce & Luci", che ha una sede fisica nella città di Roma. Il sito web è costituito da *cinque* parti:

- **Home Page**
- **Categorie**
- **Blog**
- **Il negozio**
- **Contatti**

Home è la home page del sito, ossia la pagina principale. Su di essa vengono indicate genericamente le principali categorie di prodotti venduti, sono presenti immagini di installazioni e ci sono varie altre informazioni: una descrizione dei servizi offerti, una mappa per aiutare i clienti a raggiungere lo store fisico, i link ai profili social dell'attività, un form da compilare per l'iscrizione alla newsletter e uno per prendere contatti con l'azienda.

Sezioni	Sottosezioni
Illuminazione per interni	<i>Lampade da soffitto, Lampade da parete, Lampade d'appoggio, Lampade da terra e Lampade d'emergenza</i>
Ufficio	<i>Lampade da soffitto, Lampade d'appoggio, Lampade da terra, Lampade da parete e Lampade led</i>
Strisce led	<i>Sottopensili e Strip led</i>
Incassi	<i>Incassi, Incassi gesso, Incassi doccia e Cielo stellato</i>
Esterno	<i>Lampade da soffitto, Lampade da parete, Lampade da terra, Lampade d'appoggio, Calpestabili e segnaposto e Proiettori</i>
Fibra ottica	<i>Cielo Stellato</i>
Sistema Binario	<i>Teste di binario, Binari e Accessori</i>
Ventilatori	

Tabella 3.2 – Elenco delle pagine del sito e delle relative sottosezioni

Categorie è la parte del sito dove sono mostrati i prodotti venduti. Da questa pagina si aprono *otto* pagine - che definiamo da questo momento in poi *sezioni*¹, e da ognuna di queste sezioni è possibile accedere ad ulteriori sottosezioni, contenenti i relativi prodotti in vendita. Nella tabella 3.2 sono descritte tutte le sezioni del sito, con le relative sotto-sezioni.

La parte *Blog* offre contenuti di carattere divulgativo. Attualmente in essa sono contenuti quattro articoli.

La parte *Il negozio* è una pagina contenente una breve storia dell'attività ed alcune informazioni di servizio.

Infine la parte *Contatti* contiene le informazioni per contattare i titolari dell'attività, un form per inviare mail al servizio clienti e una cartina di Google Maps per visualizzare la posizione dello store fisico.

¹più una nona, che offre la possibilità di acquistare una "carta regalo" per i prodotti del sito. Tralascio volutamente questa pagina.

Pubblico	Acquisizione	Comportamento
Utenti attivi	Sorgente/mezzo	Contenuti del sito
Comportamento	Social	Velocità del sito
Dati geografici		

Tabella 3.4 – Sezioni di GA da cui ho esportato i dati

3.2 Quali informazioni ricavare dal sito web?

Come precedentemente accennato, la riflessione che un Data Analyst deve compiere prima di iniziare il suo lavoro di raccolta dei dati è: "*Cosa devo scoprire?*". Deve avere ben presente, in altri termini, a quale domanda deve rispondere. In secondo luogo egli si porrà la domanda che naturalmente scaturisce da questa prima: "*Di quali informazioni ho bisogno per trovare ciò che devo scoprire?*". In ultimo, il Data Analyst deve porsi una terza domanda: "*Quali informazioni NON mi occorrono per trovare ciò che devo scoprire?*". Rispondiamo a queste domande esplicitandole nel caso specifico descritto in questo terzo capitolo.

"Cosa devo scoprire?" - il sito web è stato commissionato per aumentare la visibilità dell'azienda e per favorire i contatti con la clientela. L'analisi del sito tramite GA, quindi, è stata contestualmente richiesta per conoscere il volume di clientela che si interessa al sito, la percentuale di persone per cui esso risulta di scarso interesse e il dettaglio sulle sezioni più visualizzate.

"Di quali informazioni ho bisogno per trovare ciò che devo scoprire?" - Le informazioni utili per fornire risposte alle domande in esame sono nelle sezioni *Pubblico*, *Acquisizione* e *Comportamento* di GA. In dettaglio, ho raccolto i *dati geografici*, *dati di comportamento* e i *dati degli utenti attivi* dal menù *Pubblico*. Dal menù *Acquisizione* invece ho preso in analisi i dati delle sottosezioni *Sorgente/mezzo* e *Social*. Infine, sempre dalla sezione *Comportamento*, raccoglierò i dati di *Contenuti del sito*, e *Velocità del sito* (cfr. tabella 3.4).

"Quali informazioni NON mi occorrono per trovare ciò che devo scoprire?" - in questo lavoro non mi interesserò di *Conversioni*, né dei risultati della campagne pubblicitarie.

3.3 Raccolta dei dati da GA

I dati raccolti da GA per questo lavoro di tesi coprono il traffico diretto verso il sito web dal giorno 17 Luglio 2021 al giorno 30 Agosto 2021². Per ottenere questi dati è sufficiente selezionare la pagina della metrica interessata, impostare l'intervallo di tempo desiderato e cliccare sull'icona *esporta*. Tra le possibilità di esportazione, ho scelto il formato *csv* (figura 3.1).

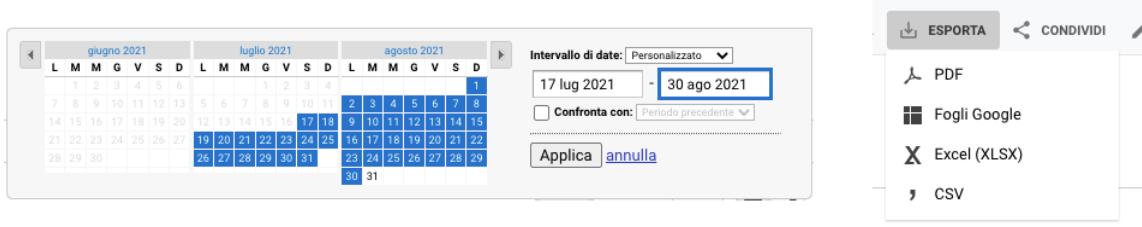


Figura 3.1 – Selezione dell’intervallo di date in cui acquisire dati di GA e salvataggio in formato csv.

Ho già avuto modo di precisare (cfr paragrafo 1.1) come la trattazione aprioristica di tutti i dati possibili da GA sia una procedura non utile perché potenzialmente confusionaria e sicuramente non necessaria. Non tutti i dati sono necessari infatti, e non tutti sono utili per rispondere alle domande del paragrafo precedente. Ho anche detto, tuttavia, come la raccolta di dati deve essere un’operazione funzionale all’analisi richiesta e anche a tutte quelle potenzialmente future. Per questo motivo ho esportato i dati di tutte le sezioni di GA, selezionando successivamente in Microsoft Excel solamente quelli necessari per l’analisi. Il file Excel risultante servirà da base per la creazione e la manipolazione dei dataset da operare in Python e su cui costruire i grafici riepilogativi e, contemporaneamente, come solido backup per ulteriori potenziali analisi future. Esso contiene ventinove Worksheet, le cui specifiche sono visibili nella tabella 3.3. Solamente alcuni di questi worksheet saranno utilizzati in questo lavoro.

²la concezione alla base di GA è che i dati sono tanto più veritieri quanto maggiore è il volume di traffico analizzato. Genericamente tale analisi copre un periodo di tempo ampio, almeno di 12 mesi, e il motivo principale è che un’attività commerciale attraversa fasi alterne in un anno solare, con periodi di maggiore attività e periodi di più scarsa clientela. In questo lavoro mostro, per ragioni di opportunità, i dati raccolti in un periodo di tempo molto più limitato. Anche se non è possibile considerare questi dati come indicativi dell’andamento globale dell’attività commerciale, il concetto di fondo dell’analisi resta valido.

3.3. Raccolta dei dati da GA

Tabella 3.5 – Elenco dei 29 worksheet di Excel ricavati dall’acquisizione dei dati csv di GA con la descrizione del loro contenuti.

Nome Worksheet	Descrizione contenuto
<i>Utenti_per_giorno</i>	Numero degli utenti che hanno visitato il sito giorno per giorno durante il periodo selezionato.
<i>Utenti_attivi</i>	Numero di utenti unici che hanno iniziato sessioni sul sito ³ .
<i>Esplorazione_utenti</i>	Caratteristiche della navigazione di ogni singolo utente, identificato per indirizzo IP ⁴ .
<i>Lingua</i>	Lingua impostata sul sistema operativo degli utenti che navigano sul sito.
<i>Località</i>	Posizione geografica degli utenti che navigano sul sito.
<i>Nuovi_vs Ritorno</i>	Numero dei nuovi utenti e di quelli che visitano il sito per una seconda volta.
<i>Frequenza_e_recency</i>	Tasso di frequenza delle visualizzazioni da parte dei singoli utenti e tempo intercorso tra una visita e quella successiva.
<i>Coinvolgimento</i>	Durata delle interazioni degli utenti col nostro sito.
<i>Browser</i>	Tipologia del browser utilizzato dagli utenti per navigare sul sito.
<i>Sistema_operativo</i>	Sistema operativo utilizzato dagli utenti che visitano il sito.
<i>Risoluzione_schermo</i>	Risoluzione dello schermo del dispositivo utilizzato dagli utenti che visitano il sito.
<i>Colori_schermo</i>	Impostazione dei colori dello schermo del dispositivo utilizzato dagli utenti che visitano il sito.
<i>Versione_flash</i>	Versione di Adobe Flash Player utilizzata dagli utenti.

3.3. Raccolta dei dati da GA

<i>JavaScript</i>	Indica se il device dell'utente che naviga supporta o meno il linguaggio Java.
<i>Fornitore_servizi</i>	Fornitore del servizio telefonico utilizzato per navigare dall'utente.
<i>Tipo_di_cellulare</i>	Modello dello smartphone utilizzato per navigare sul sito.
<i>Brand_cellulare</i>	Brand dello smartphone utilizzato per navigare sul sito.
<i>Acquisizione_panoramica</i>	Panoramica dell'acquisizione degli utenti (ricerca diretta, social, direct, etc...).
<i>Canali</i>	Dettaglio dell'acquisizione degli utenti.
<i>Sorgente_mezzo</i>	Sorgente e mezzo di provenienza della visita.
<i>Sorgente</i>	Dettaglio sulla sorgente di provenienza della visita.
<i>Mezzo</i>	Dettaglio sul mezzo di provenienza della visita.
<i>Referral_social_network</i>	Dettaglio sul referral dei social Network.
<i>Pagina_destinazione_social</i>	Pagine a cui rimandano i link sui social Network.
<i>Tutte_le_pagine</i>	Dettagli delle visualizzazioni di tutte le singole pagine del sito.
<i>Dettaglio_contenuto</i>	Metrica molto simile alla precedente, ma strutturata in funzione della gerarchia delle pagine del sito web.
<i>Pagine_di_destinazione</i>	Dettaglio delle pagine a cui gli utenti sono stati indirizzati dopo una ricerca.
<i>Pagine_di_uscita</i>	Dettaglio delle pagine da cui gli utenti hanno scelto di abbandonare il sito.
<i>Tempi_di_pagine</i>	Tutte le pagine del sito visualizzate in funzione del tempo di navigazione.

3.4 Import dei dati su Excel

Ho già descritto in 3.4 la procedura per importare i file csv in un foglio Excel. Esaminerò ora il processo per i file csv provenienti da GA, i successivi aggiustamenti ai vari fogli e la creazione del documento definitivo con tutti i Sheet contenenti le varie metriche. Per importare i dati delle metriche di GA utilizzo la procedura già descritta nel paragrafo 3.4. Purtroppo Excel non permette l'import di più files csv contemporaneamente, è necessario quindi svolgere le medesime operazioni per ogni singolo file. Una volta selezionato il file si apre la schermata di personalizzazione, in cui effettuo alcuni aggiustamenti preliminari. Per prima cosa imposto il numero di righe da saltare o, volendo, il numero della prima riga che si considera essere valida e costituente il dataset. Evito di inserire righe iniziali di descrizione, in modo da avere i dati del futuro dataset a partire dalla cella A1 e agevolarmi nei successivi passaggi in Python. Valido a questo punto l'import, e la struttura a celle del foglio Excel si riempie dei dati (figura 3.2).

A questo punto trasformo le serie di dati in una tabella, selezionando una cella qualsiasi e utilizzando la combinazione di tasti *Ctrl + T*. Imposto, nella piccola finestra che si apre, la presenza degli "Headers" e valido il comando: da questo momento i dati sono organizzati in una tabella, che si riconosce per la colorazione a righe alterne e per la presenza dei menu di selezione accanto alle celle contenenti gli Headers. Ogni volta che si seleziona una cella della tabella si rendono disponibili due nuovi menu, *Configurazione* e *Tabella*, dai quali posso scegliere numerose impostazioni per la tabella appena creata. Siccome la creazione del dataset finale sarà effettuata con Python, non ho apportato alcuna modifica ai dati tramite Excel, ma mi sono limitato a organizzare ogni singola tabella in uno Sheet diverso del foglio Excel.

Il processo di import di un file csv in particolare ci mostra alcune possibilità di personalizzazione del processo, ed è il file csv relativo alla metrica "*Utenti attivi*". Esso contiene la stessa metrica valutata per 1, 7, 14 e 28 giorni, e i relativi dati sono in

³Questa metrica è ulteriormente suddivisa da GA in: utenti attivi in 1 giorno: numero di utenti unici che hanno iniziato sessioni sul tuo sito l'ultimo giorno dell'intervallo di date. Utenti attivi in 7 giorni: numero di utenti unici che hanno iniziato sessioni sul tuo sito gli ultimi 7 giorni dell'intervallo di date. Utenti attivi in 14 giorni: numero di utenti unici che hanno iniziato sessioni sul tuo sito gli ultimi 14 giorni dell'intervallo di date. Utenti attivi in 28 giorni: numero di utenti unici che hanno iniziato sessioni sul tuo sito gli ultimi 28 giorni dell'intervallo di date.

⁴che, essendo ovviamente un'informazione riservata, viene tradotto in un *ID cliente*.

Figura 3.2 – Import dei dati di GA sul foglio Excel e finestra di dialogo per la personalizzazione del processo di import.

un'unica colonna. Per questo motivo ho dovuto selezionare i dati che mi interessano, spezzando la colonna in quattro parti. Per fare ciò ho selezionato il numero di righe che mi occorrevano utilizzando il comando "*Rimuovi righe*" e poi "*Rimuovi ultime righe*". Per importare i dati successivi, ho eliminato sia le righe iniziali che quelle finali, e in più ho scelto una cella specifica dove incollarli, modificando quella di default (cella A1). In questo modo ho ottenuto su una sola pagina i dati degli utenti attivi, ma su quattro tabelle diverse adiacenti tra loro (figura 3.3).

3.5 Costruzione e gestione dei dataset

In questo capitolo mostrerò come importare da Excel i dati, trasformarli in un dataset definitivo e come costruire i grafici. Per prima cosa ho aperto un nuovo file di Jupyter per il nostro codice, e ho importato i pacchetti necessari per iniziare il lavoro (cfr. figura 2.27). Una volta caricati i pacchetti ho caricato anche il file Excel da cui leggere le informazioni per la creazione dei dataset (figura 3.4).

In questo caso l'accezione "caricare" è differente da quello che si intende per i pacchetti. Con il comando in figura 3.4, infatti, assegniamo alla variabile *xtl* il percorso del file Excel, tutti i successivi comandi che conterranno la variabile *xtl*, quindi, si riferiranno allo stesso file nel medesimo percorso.

3.5. Costruzione e gestione dei dataset

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Giorno	Utenti attivi in un giorno	Giorno	Utenti attivi in 7 giorni	Giorno	Utenti attivi in 14 giorni	Giorno	Utenti attivi in 28 giorni						
2	17/07/21	60	17/07/21	60	17/07/21	60	17/07/21	60						
3	18/07/21	79	18/07/21	135	18/07/21	135	18/07/21	135						
4	19/07/21	87	19/07/21	218	19/07/21	218	19/07/21	218						
5	20/07/21	72	20/07/21	278	20/07/21	278	20/07/21	278						
6	21/07/21	32	21/07/21	301	21/07/21	301	21/07/21	301						
7	22/07/21	87	22/07/21	379	22/07/21	379	22/07/21	379						
8	23/07/21	81	23/07/21	446	23/07/21	446	23/07/21	446						
9	24/07/21	29	24/07/21	414	24/07/21	467	24/07/21	467						
10	25/07/21	82	25/07/21	419	25/07/21	542	25/07/21	542						
11	26/07/21	54	26/07/21	393	26/07/21	589	26/07/21	589						
12	27/07/21	47	27/07/21	368	27/07/21	627	27/07/21	627						
13	28/07/21	151	28/07/21	485	28/07/21	769	28/07/21	769						
14	29/07/21	180	29/07/21	576	29/07/21	931	29/07/21	931						
15	30/07/21	102	30/07/21	590	30/07/21	1.015	30/07/21	1.015						
16	31/07/21	76	31/07/21	632	31/07/21	1.029	31/07/21	1.082						
17	01/08/21	77	01/08/21	627	01/08/21	1.031	01/08/21	1.151						
18	02/08/21	82	02/08/21	657	02/08/21	1.027	02/08/21	1.221						
19	03/08/21	76	03/08/21	686	03/08/21	1.028	03/08/21	1.285						
20	04/08/21	93	04/08/21	627	04/08/21	1.088	04/08/21	1.365						
21	05/08/21	259	05/08/21	712	05/08/21	1.261	05/08/21	1.613						
22	06/08/21	191	06/08/21	796	06/08/21	1.368	06/08/21	1.790						
23	07/08/21	123	07/08/21	849	07/08/21	1.461	07/08/21	1.905						
24	08/08/21	100	08/08/21	874	08/08/21	1.482	08/08/21	2.001						
25	09/08/21	68	09/08/21	865	09/08/21	1.498	09/08/21	2.057						

Figura 3.3 – Import del file csv relativo alla metrica Utenti attivi".

```
In [2]: # import del file Excel

# Individuazione della directory corrente con getcwd()
ptw = os.getcwd()

# Assegnazione del file Excel alla variabile xl
xsl = pd.ExcelFile(ptw + '/GA.xlsx')
```

Figura 3.4 – Assegnazione del percorso del file Excel alla variabile *xtl*.

Da questo momento in poi è possibile costruire i dataset, selezionando di volta in volta dal foglio Excel quelli di nostro interesse.

3.5.1 Dataset "utenti per giorno"

In questo dataset carico i dati relativi al conteggio degli utenti che hanno navigato il sito web durante l'intervallo di tempo prescelto, per ogni singolo giorno. Per caricare il dataset dalla pagina Excel ho utilizzato il comando in figura 3.5:

```
In [4]: # Dataset Utenti_per_giorno
u_p_g = pd.read_excel(xtl, 'Utenti_per_giorno')
u_p_g.head(15)
```

Out[4]:

	Giorno	Utenti
0	17/07/21	60
1	18/07/21	79
2	19/07/21	87
3	20/07/21	72
4	21/07/21	32
5	22/07/21	87
6	23/07/21	81
7	24/07/21	29
8	25/07/21	82
9	26/07/21	54
10	27/07/21	47
11	28/07/21	151
12	29/07/21	180
13	30/07/21	102
14	31/07/21	76

Figura 3.5 – Caricamento del dataset del numero di utenti che hanno visitato il sito ogni giorno, proveniente dal foglio Excel.

Analizzando il codice, ci accorgiamo che `pd.read_excel` è il comando del pacchetto *pandas* - di cui `pd` è l'alias - con cui si caricano informazioni che risiedono su di un foglio Excel. `xtl`, che è il primo argomento, è il nome del file Excel da cui prelevare le informazioni, e 'Utenti per giorno' è il nome del Worksheet da cui cercare i dati.

Questa sintassi è uguale per tutti i dataset, ma per alcuni occorrono ulteriori aggiustamenti.

3.5.2 Dataset "utenti attivi"

La procedura per caricare il dataset "Utenti attivi", ad esempio, presenta alcune criticità (figura 3.6).

```
In [5]: # Dataset Utenti_attivi
u_a_1_gg = pd.read_excel(xtl,'Utenti_attivi', usecols = 'A:B', thousands = '.')
u_a_7_gg = pd.read_excel(xtl,'Utenti_attivi', usecols = 'D:E',
                         thousands = '.').rename(columns = {"Giorno.1" : "Giorno"})
u_a_14_gg = pd.read_excel(xtl,'Utenti_attivi', usecols = 'G:H',
                          thousands = '.').rename(columns = {"Giorno.2" : "Giorno"})
u_a_28_gg = pd.read_excel(xtl,'Utenti_attivi', usecols = 'J:K',
                          thousands = '.').rename(columns = {"Giorno.3" : "Giorno"})
```

Figura 3.6 – Caricamento del dataset "utenti attivi".

In primo luogo sulla pagina Excel sono presenti più tabelle, poste una accanto all'altra, quindi occorre indicare le colonne da cui si vogliono prelevare i dati anziché lasciare che Python carichi l'intera pagina. Analizziamo allora la sintassi utilizzata per questo scopo: `u_a_1_gg = pd.read_excel(xtl,'Utenti_attivi', usecols = 'A:B', thousands = '.')`. L'argomento `usecols` permette di selezionare le colonne A e B, dove risiedono i dati che mi interessano. La seconda criticità sta nel fatto che i dati numerici vengono esportati da GA separando la cifra delle migliaia con un punto mentre Python, al contrario, utilizza il punto per separare le unità dalle cifre decimali. Importati così come sono, questi valori risulterebbero inutilizzabili per le manipolazioni del dataset, e quindi ho passato l'argomento `thousands`, indicando il punto come separatore di migliaia. Con questo comando Python ha provveduto a trasformare la virgola in punto. L'ultima criticità è rappresentata dal fatto che quando sono presenti intestazioni di colonna ripetute nel foglio Excel, l'import di Python le rinomina automaticamente, aggiungendo un punto e un numero ordinale. Ho quindi modificato le intestazioni delle colonne che erano state rinominate, utilizzando il comando `.rename(columns = "Giorno.1" : "Giorno")`.

In figura 3.7 sono mostrati i dataset importati con e senza gli aggiustamenti descritti. Si nota come il primo (a sinistra nella figura) sia inutilizzabile, fornendo dati errati e non veritieri.

Giorno	Utenti attivi in un giorno	Unnamed: 2	Giorno.1	Utenti attivi in 7 giorni	Unnamed: 5	Giorno.2	Utenti attivi in 14 giorni	Unnamed: 8	Giorno.3	Utenti attivi in 28 giorni	Giorno.2 Utenti attivi in 14 giorni	
											0	1
0	17/07/21	60	NaN	17/07/21	60	NaN	17/07/21	60.000	NaN	17/07/21	60.000	0
1	18/07/21	79	NaN	18/07/21	135	NaN	18/07/21	135.000	NaN	18/07/21	135.000	1
2	19/07/21	87	NaN	19/07/21	218	NaN	19/07/21	218.000	NaN	19/07/21	218.000	2
3	20/07/21	72	NaN	20/07/21	278	NaN	20/07/21	278.000	NaN	20/07/21	278.000	3
4	21/07/21	32	NaN	21/07/21	301	NaN	21/07/21	301.000	NaN	21/07/21	301.000	4
5	22/07/21	87	NaN	22/07/21	379	NaN	22/07/21	379.000	NaN	22/07/21	379.000	5
6	23/07/21	81	NaN	23/07/21	446	NaN	23/07/21	446.000	NaN	23/07/21	446.000	6
7	24/07/21	29	NaN	24/07/21	414	NaN	24/07/21	467.000	NaN	24/07/21	467.000	7
8	25/07/21	82	NaN	25/07/21	419	NaN	25/07/21	542.000	NaN	25/07/21	542.000	8
9	26/07/21	54	NaN	26/07/21	393	NaN	26/07/21	589.000	NaN	26/07/21	589.000	9
10	27/07/21	47	NaN	27/07/21	368	NaN	27/07/21	627.000	NaN	27/07/21	627.000	10
11	28/07/21	151	NaN	28/07/21	485	NaN	28/07/21	769.000	NaN	28/07/21	769.000	11
12	29/07/21	180	NaN	29/07/21	576	NaN	29/07/21	931.000	NaN	29/07/21	931.000	12
13	30/07/21	102	NaN	30/07/21	590	NaN	30/07/21	1.015	NaN	30/07/21	1.015	13
14	31/07/21	76	NaN	31/07/21	632	NaN	31/07/21	1.029	NaN	31/07/21	1.082	14

Figura 3.7 – Raffronto dei dataset ottenuti dal worksheet "Utenti attivi" con e senza i comandi aggiuntivi

3.5.3 Dataset "esplorazione utenti"

Il dataset "Esplorazione utenti" presenta notevoli difficoltà di esportazione, e ha richiesto una procedura più complessa. La prima difficoltà è rappresentata dalla colonna "ID cliente", che è costituita da due serie di dieci numeri suddivisi da un punto. Non essendo tecnicamente né numeri ordinali né valori, la strategia migliore è stata trattarli come una stringa di testo. Per fare ciò ho forzato pandas a considerare la colonna come contenente appunto un testo, altrimenti, notando dei valori numerici, essa l'avrebbe trattata, appunto, come un numero, con conseguenti difficoltà⁵. Ho utilizzato il comando *astype* in questo modo: `e_u_1 = pd.read_excel(xtl, 'Esplorazione_utenti', usecols = "A").astype("ID cliente": str)`, dove "*str*" indica che la colonna contiene delle stringhe. Dalla sintassi si nota come ho applicato il comando *astype* esclusivamente alla prima colonna, selezionandola con il già noto comando *usecols*. Il motivo risiede nel fatto che, se avessi applicato il comando a tutto il dataset, esso avrebbe trasformato anche le altre colonne come contenenti stringhe di testo, con il risultato di rendere inutilizzabili le colonne dei valori. La seconda sintassi ci mostra un nuovo comando utilizzato, che è il comando *drop*: `.drop(["Entrate", "Transazioni", "Tasso di conversione all'obiettivo"], axis = 1)`. Esso serve ad eliminare alcune colonne che non sono utili, al fine di alleggerire il dataset. In questo caso le colonne che ho eliminato sono quelle relative alle conversioni, alle transazioni e ai guadagni che, come detto, non sono metriche richieste ai fini di questa analisi. La terza sintassi mostra il comando *join*: `e_u = e_u_1.join(e_u_2)`. Questo comando serve ad unire orizzontalmente due dataset, ancorandoli tramite una colonna, più comunemente quella dell'indice. In pratica alla colonna del primo dataset vengono aggiunte tutte quelle del secondo, e l'ordine di associazione si affida alla numerazione dell'indice stesso: alla riga **0** del primo dataset viene associata la riga **0** del secondo, alla riga **1** del primo viene associata la riga **1** del secondo e così via (cfr figura 3.8).

In ultimo, questo dataset ci consente di riflettere sul primo caso di dato *outsider* (cfr paragrafo 1.2), che è presente nella colonna della *frequenza di rimbalzo*. La frequenza di rimbalzo è la percentuale di utenti che, una volta "atterrati" su una pagina del sito (generalmente la home page), la abbandonano senza alcuna ulteriore interazione. La sua valutazione, come parametro a sé stante, è molto utile per conoscere l'interesse generale del sito web, ma esso stesso può diventare un dato fuorviante quando invece

⁵sarebbe difficile selezionare un utente in base al suo "nome" ad esempio.

	e_u_1.head(15)	e_u_2.head(15)	e_u_.head(15)
ID cliente			
	Sessoni	Durata sessione media	Frequenza di rimbalzo
① 1043577767.1621507	① 66	00:03:48	0,00%
② 923184641.1626166	② 42	00:12:39	0,00%
③ 1956949618.161736	③ 34	00:13:21	0,00%
④ 1101014214.1592498	④ 32	00:29:40	0,00%
⑤ 1691856457.1627925	⑤ 21	00:07:08	47,62%
⑥ 751358216.1626209	⑥ 12	00:00:00	100,00%
⑦ 513718547.1626168	⑦ 12	00:01:18	0,00%
⑧ 123269003.16288295	⑧ 11	00:00:07	63,64%
⑨ 167279939.16294804	⑨ 11	00:00:09	0,00%
⑩ 75733372.1627493	⑩ 11	00:03:22	0,00%
⑪ 1774762826.1624198	⑪ 9	00:00:20	66,67%
⑫ 1567739311.1627762	⑫ 8	00:10:50	0,00%
⑬ 1334069083.1628084	⑬ 7	00:00:51	0,00%
⑭ 100653675.16273876	⑭ 6	00:00:58	16,67%

Figura 3.8 – Risultato del processo di *join* tra i due dataset "esplorazione utenti"

intendiamo analizzare ed osservare l'interazione specifica che i singoli utenti hanno col sito web stesso, valutando magari una sola pagina o gruppi di pagine. Tutti i dati delle visite con frequenza di rimbalzo del 100%, infatti, aumentano notevolmente la media dei valori, falsando così il risultato finale. Ecco quindi che può essere utile costruire una versione del dataset da cui sono stati eliminati tutti i valori relativi, appunto, a una frequenza di rimbalzo del 100%. Per fare questo utilizziamo questa sintassi: `e_u_rimb = e_u[(e_u['Frequenza di rimbalzo'] != '100,00%')]`, che ci dice che dal dataset vengono selezionate le occorrenze il cui valore nella colonna 'Frequenza di rimbalzo' è diverso (simbolo '`!=`') dal 100%. Il risultato è un dataset identico nella forma, ma con un numero di righe inferiore rispetto al precedente. Dalla figura 3.9 è possibile notare le differenze: nel dataset di sinistra, ad esempio, è presente la riga numero 6, che è assente nel dataset di destra, che è quello ottenuto dopo aver eliminato le occorrenze con la frequenza di rimbalzo pari al 100%.

3.5.4 Dataset "tutte le pagine"

Il processo per la creazione del dataset "tutte le pagine", infine, mostra altri tre comandi molto utili. Per prima cosa, utilizzando la sintassi già nota, ho caricato il dataset con i dati presenti nel worksheet di Excel (figura 3.10). A partire da questo dataset ne ho costruiti altri otto, corrispondenti alle otto sezioni di vendita descritte nel paragrafo 3.1, al fine di ottenere informazioni più dettagliate sul traffico di ognuna delle sezioni. L'immagine 3.11 mostra il codice per tutte e otto le sezioni.

La sintassi è la seguente (prendendo ad esempio la prima sezione): `ufficio = t_1_p[(t_1_p['Pagina'] == "/72-ufficio")]` che sta a significare che viene creato un nuovo dataset con il nome di "ufficio" a partire dal dataset "tutte le pagine", contenente solamente le occorrenze in cui la pagina di interesse è quella identificata con il nome "/72-ufficio". In questo modo ho selezionato il traffico che si è diretto verso le pagine delle singole sezioni. Essendo la pagina una sola, il dataset corrispondente sarà costituito da una sola riga (figura 3.12).

Una volta creati tutti e otto i dataset per le rispettive sezioni, li ho uniti per formare un unico dataset, utilizzando il comando *append*. La sintassi è molto semplice ed è stata già descritta nel paragrafo 3.6. La figura 3.13 ci mostra come sia possibile agire associando più dataset con una sola riga di comando.

	In [20]:	e_u.head(15)	Out[20]:		In [21]:	e_u_rimb.head(15)	Out[21]:		
	ID cliente	Sessoni	Durata sessione media	Frequenza di rimbalzo		ID cliente	Sessoni	Durata sessione media	Frequenza di rimbalzo
0	1043577767.1621507	66	00:03:48	0,00%	0	1043577767.1621507	66	00:03:48	0,00%
1	923184641.1626166	42	00:12:39	0,00%	1	923184641.1626166	42	00:12:39	0,00%
2	1956949618.161736	34	00:13:21	0,00%	2	1956949618.161736	34	00:13:21	0,00%
3	1101014214.1592498	32	00:29:40	0,00%	3	1101014214.1592498	32	00:29:40	0,00%
4	1691856457.16275	21	00:07:08	47,62%	4	1691856457.16275	21	00:07:08	47,62%
5	751358216.1626209	18	00:00:25	0,00%	5	751358216.1626209	18	00:00:25	0,00%
6	1745572995.1627922	12	00:00:00	100,00%	6	1745572995.1627922	12	00:01:18	0,00%
7	513718547.1626168	12	00:01:18	0,00%	7	513718547.1626168	12	00:01:18	0,00%
8	123269003.16288295	11	00:00:07	63,64%	8	123269003.16288295	11	00:00:07	63,64%
9	167279939.16294804	11	00:00:09	0,00%	9	167279939.16294804	11	00:00:09	0,00%
10	757333372.1627493	11	00:03:22	0,00%	10	757333372.1627493	11	00:03:22	0,00%
11	1774762826.1624198	9	00:00:20	66,67%	11	1774762826.1624198	9	00:00:20	66,67%
12	1567739311.1627762	8	00:10:50	0,00%	12	1567739311.1627762	8	00:10:50	0,00%
13	1334069083.1628084	7	00:00:51	0,00%	13	1334069083.1628084	7	00:00:51	0,00%
14	100653675.16273876	6	00:00:58	16,67%	14	100653675.16273876	6	00:00:58	16,67%
15	267017044.16173616	6			15	267017044.16173616	6	00:14:15	0,00%

Figura 3.9 – dataset "utenti attivi" prima e dopo aver eliminato le occorrenze con frequenza di rimbalzo pari al massimo.

3.5. Costruzione e gestione dei dataset

In [26]: # Dataset Tutte le pagine							
Out[26]:							
	Pagina	Visualizzazioni di pagina	Visualizzazioni di pagina uniche	Tempo medio sulla pagina	Accessi	Frequenza di rimbalzo	% uscita
0	/	2741	1456	00:00:43	1324	42,67%	31,23%
1	/70-illuminazione-per-interni	2325	1358	00:00:21	966	56,73%	33,51%
2	/65-esterno	1335	909	00:00:33	834	61,63%	50,86%
3	/3-lampade-da-soffitto	942	403	00:00:32	98	50,00%	12,31%
4	/26-lampade-da-parete	405	201	00:00:26	63	57,14%	18,77%
5	/18-lampade-soffitto-moderno	403	189	00:01:02	3	33,33%	10,17%
6	/30-lampade-da-terra	382	186	00:00:20	105	51,43%	24,35%
7	/content/18-il-negozi-	298	141	00:00:25	4	0,00%	12,42%
8	/home-6.html	291	111	00:00:55	14	14,29%	13,75%
9	/96-lampade-da-parete-esterno	281	163	00:00:36	44	65,91%	25,62%
10	/4-lampade-d-appoggio	221	125	00:00:22	49	65,31%	24,89%
11	/contattaci	188	96	00:00:23	9	55,56%	20,74%
12	/cerca?s=ice	141	60	00:00:19	1	0,00%	9,93%
13	/73-strisce-led	139	50	00:00:12	22	36,36%	16,55%
14	/61-incassi	128	66	00:00:21	50	46,00%	28,91%

Figura 3.10 – Caricamento del dataset "Tutte le pagine".

In [27]: # Dataset pagine per le categorie vendute (heatmap)							
ufficio = t_l_p[(t_l_p['Pagina'] == "/72-ufficio")]							
ill_int = t_l_p[(t_l_p['Pagina'] == "/70-illuminazione-per-interni")]							
led = t_l_p[(t_l_p['Pagina'] == "/73-strisce-led")]							
incassi = t_l_p[(t_l_p['Pagina'] == "/61-incassi")]							
esterno = t_l_p[(t_l_p['Pagina'] == "/65-esterno")]							
fibra_ottica = t_l_p[(t_l_p['Pagina'] == "/74-fibra-ottica")]							
sistema_binario = t_l_p[(t_l_p['Pagina'] == "/71-sistema-binario")]							
ventilatori = t_l_p[(t_l_p['Pagina'] == "/118-ventilatori")]							

Figura 3.11 – Creazione dei dataset per ogni singola sezione del sito.

In [28]: ufficio							
Out[28]:							
	Pagina	Visualizzazioni di pagina	Visualizzazioni di pagina uniche	Tempo medio sulla pagina	Accessi	Frequenza di rimbalzo	% uscita
51	/72-ufficio	34	20	00:00:22	2	0,00%	2,94%

Figura 3.12 – Dataset della sezione "ufficio".

In [32]: categ = ufficio.append(esterno).append(led).append(incassi).append(ill_int).append(fibra_ottica).append(sistema_binario)							
Out[32]:							
	Pagina	Visualizzazioni di pagina	Visualizzazioni di pagina uniche	Tempo medio sulla pagina	Accessi	Frequenza di rimbalzo	% uscita
194	/74-fibra-ottica	12	7	00:00:06	1	0,00%	8,33%
51	/72-ufficio	34	20	00:00:22	2	0,00%	2,94%
50	/71-sistema-binario	34	10	00:00:07	1	0,00%	0,00%
28	/118-ventilatori	50	19	00:00:12	0	0,00%	4,00%
14	/61-incassi	128	66	00:00:21	50	46,00%	28,91%
13	/73-strisce-led	139	50	00:00:12	22	36,36%	16,55%
2	/65-esterno	1335	909	00:00:33	834	61,63%	50,86%
1	/70-illuminazione-per-interni	2325	1358	00:00:21	966	56,73%	33,51%

Figura 3.13 – Creazione di un unico dataset contenente i dati delle sezioni del sito.

Dalla figura è anche possibile osservare l'utilizzo del terzo nuovo comando, il comando *sort*, che ordina i valori secondo un criterio stabilito dall'utente. In questo caso la sintassi `categ.sort_values(["Visualizzazioni di pagina"])` orienta la visualizzazione del dataset in base ai valori presenti nella colonna "Visualizzazioni di pagina", in ordine ascendente. Vengono mostrate, cioè, prima le sottopagine con meno visualizzazioni e poi quelle con maggiori visualizzazioni⁶.

3.6 Costruzione dei grafici

Una volta ottenuti tutti i dataset, essi possono essere utilizzati per costruire grafici riepilogativi che permettano di visualizzare, in maniera quanto più possibile chiara, le caratteristiche della navigazione verso il sito Web⁷.

3.6.1 Quanti utenti visitano il sito? Lineplot e scatterplot

Il primo grafico che ho implementato è quello che mostra quanti utenti hanno navigato, giorno per giorno, il sito in oggetto, valutandone il numero e l'andamento generale. Ne scaturisce il grafico a linea in figura 3.14.

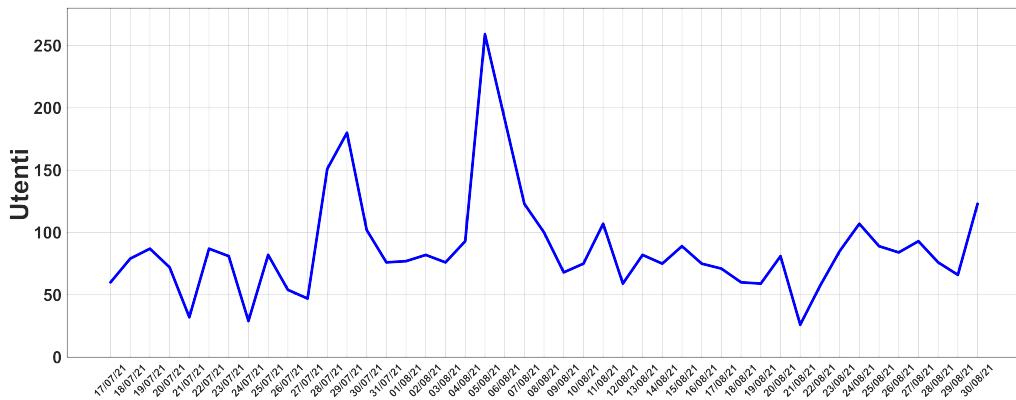


Figura 3.14 – Lineplot con l'andamento degli utenti giorno per giorno.

⁶la possibilità di visualizzare il dataset in maniere alternative con comandi molto semplici è un punto di forza del pacchetto *pandas*. È bene chiarire che i dati del dataset rimangono sempre gli stessi, ma vengono solamente spostati in ordini differenti a seconda delle necessità di chi li osserva.

⁷il dataset non è una struttura fissa e inamovibile. Spesso, nelle ultime fasi dell'analisi dei dati, ossia quella della creazione di grafici di riepilogo, è necessario apportare leggere modifiche o filtri di visualizzazione a questi dataset. È importante quindi pensare ai dati raccolti come una solida base - ordinata e conservata - da cui partire per ulteriori modifiche ed aggiustamenti che si possono rivelare via via necessari.

Sull'asse delle x troviamo una *serie temporale*: il lasso di tempo, cioè, durante il quale ho analizzato gli utenti del sito, suddiviso giornalmente. I valori sull'asse delle y rappresentano invece il numero di utenti. Dal grafico si nota un andamento alquanto stabile con una media di una cinquantina di utenti per giorno, a parte due picchi di visualizzazione in corrispondenza del 28 Luglio e del 5 Agosto. È importante identificare i motivi di questi aumenti, in considerazione del fatto che in quei giorni il numero dei visitatori è aumentato rispettivamente di circa tre volte e di circa cinque volte. Il primo picco è giustificabile con il fatto che il sito web era stato terminato e lanciato una decina di giorni prima: si assiste sempre a un picco di visite nei giorni immediatamente successivi al lancio di un sito ben indicizzato. Il secondo picco può essere in parte giustificato con una manutenzione dello stesso - molte più visite a partire dal webmaster - o con una probabile concomitanza con il lancio di nuovi prodotti. Altro motivo potrebbe essere invece il lancio di una campagna pubblicitaria tramite Google o Social Network⁸.

I grafici di questo tipo offrono una visuale organica dell'*andamento* di un valore, ossia di un trend di crescita, decrescita o stabilità, ma possono essere, a volte, poco chiari. Per aiutarci a chiarire la situazione e ad identificare un eventuale trend si può trasformare il grafico in uno scatterplot, aggiungendo una linea di tendenza. Nella nuova figura (3.15) è più visibile come l'andamento delle visite del sito nei giorni di analisi sia rimasto costante nel tempo.

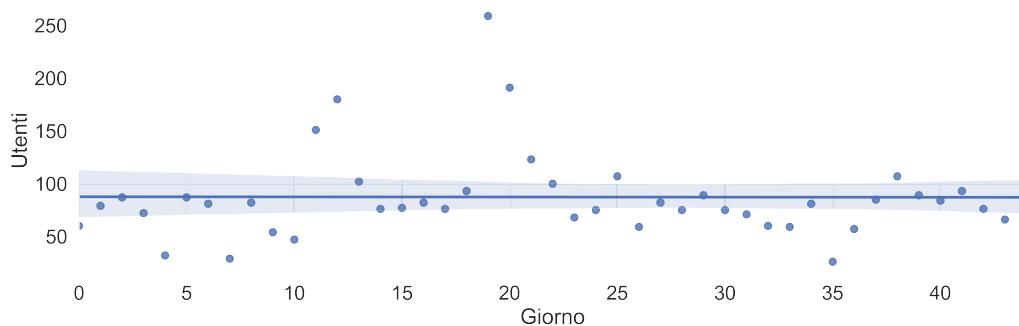


Figura 3.15 – Scatterplot con l'andamento degli utenti giorno per giorno.

⁸compito dell'analista di dati è *riportare* i numeri in maniera ordinata e leggibile, in modo da poter comprendere le dinamiche che li hanno generati. Il dialogo con il committente è passo successivo e fondamentale perché a questi dati venga attribuito un significato. In questo caso, quindi, non siamo tenuti necessariamente a sapere perché in quei giorni il numero di utenti sia notevolmente aumentato, ma possiamo aiutare a comprenderne i motivi.

3.6.2 Gli utenti attivi: lineplot mutiplo

GA raccoglie i dati utenti caratterizzandoli anche come "utenti attivi", ovverosia "utenti unici che hanno iniziato sessioni sul tuo sito". La metrica viene suddivisa in intervalli temporali: utenti attivi in 1 giorno, utenti attivi in 7 giorni, utenti attivi in 14 giorni e utenti attivi a 28 giorni, tenendo in considerazione l'ultimo giorno dell'intervallo di dati scelto. In pratica, questa metrica misura gli utenti con almeno una sessione di interazioni sul sito guardando a ritroso nel tempo (i giorni in questione si contano dall'ultimo dell'intervallo impostato). Questa metrica è importante perché rivela se le persone che si interessano al sito sono in crescita, stazionarie o in decrescita. Per poter visualizzare immediatamente questo trend ho plottato i dati con un altro lineplot, ma stavolta ho inserito nella stessa figura (3.16) i valori di tutti e quattro i database contemporaneamente. Si vede chiaramente che il numero di utenti attivi sale al crescere dell'intervallo di tempo analizzato, e questo andamento ci permette di affermare che, perlomeno nell'ultimo mese, il numero di persone che si sono interessate al sito è sempre maggiore.

Figura 3.16 – Lineplot multiplo per la visualizzazione degli utenti attivi

3.6.3 Provenienza degli utenti: grafico a ciambella e grafico a bolle

Un secondo dato interessante è: da dove provengono gli utenti che navigano il sito? Il dataset costruito a partire dai dati della *Località* dice che la maggioranza di essi proviene dall'Italia (figura 3.17, sinistra). Per ottenere un'informazione più veritiera però è necessario notare che alcune di queste visite presentano una *frequenza di rimbalzo* pari al 100%. La *frequenza di rimbalzo* è il parametro che indica "*il rapporto tra le sessioni di una sola pagina divise per tutte le sessioni o la percentuale di tutte le sessioni sul tuo sito nelle quali gli utenti hanno visualizzato solo una pagina e hanno attivato una sola richiesta al server Analytics*".⁹. Esso indica, quindi, quanti utenti hanno aperto una

⁹<https://support.google.com/analytics/answer/1009409?hl=it>

	Paese	Utenti	Nuovi utenti	Sessioni	Frequenza di rimbalzo	Pagine/sessione	Durata sessione media
0	Italy	3361	3361	4238	51,18%	5,17	00:02:11
1	United States	56	56	57	92,98%	2,16	00:02:27
2	China	33	32	34	85,29%	2,15	00:00:54
3	Germany	8	8	8	75,00%	1,75	00:00:10
4	Romania	8	8	12	33,33%	3,00	00:00:13
5	Switzerland	6	6	6	83,33%	2,00	00:00:20
6	United Kingdom	5	5	5	100,00%	1,00	00:00:00
7	Ireland	5	5	6	66,67%	1,67	00:00:10
8	Sweden	5	5	5	100,00%	1,00	00:00:00
9	France	4	3	4	25,00%	7,00	00:02:06
10	Albania	3	2	3	66,67%	2,00	00:00:10
11	Pakistan	3	3	4	75,00%	1,25	00:00:25
12	Austria	2	2	2	50,00%	2,00	00:00:06
13	Belgium	2	2	2	50,00%	4,00	00:00:48
14	Canada	2	2	2	50,00%	8,00	00:02:24
15	Czechia	2	2	3	66,67%	7,33	00:00:00
16	Malta	2	2	2	100,00%	1,00	00:00:00
17	Netherlands	2	2	2	50,00%	2,00	00:00:15
18	Poland	2	2	3	66,67%	1,33	00:00:00
19	Singapore	2	2	2	50,00%	3,00	00:07:04
20	Brazil	1	1	1	0,00%	5,00	00:00:43
21	Spain	1	1	1	100,00%	1,00	00:00:00
22	Finland	1	1	1	0,00%	4,00	00:00:36
23	Greece	1	1	1	100,00%	1,00	00:00:00
24	Hong Kong	1	1	1	0,00%	4,00	00:00:54
25	Hungary	1	1	1	100,00%	1,00	00:00:00
26	South Korea	1	1	1	0,00%	2,00	00:00:48
27	Luxembourg	1	1	1	0,00%	25,00	00:04:13
28	Portugal	1	1	1	100,00%	1,00	00:00:00
29	Slovenia	1	1	1	0,00%	3,00	00:00:14
30	Kosovo	1	1	1	100,00%	1,00	00:00:00

	Paese	Utenti	Nuovi utenti	Sessioni	Frequenza di rimbalzo	Pagine/sessione	Durata sessione media
0	Italy	3361	3361	4238	51,18%	5,17	00:02:11
1	United States	56	56	57	92,98%	2,16	00:02:27
2	China	33	32	34	85,29%	2,15	00:00:54
3	Germany	8	8	8	75,00%	1,75	00:00:10
4	Romania	8	8	12	33,33%	3,00	00:00:13
5	Switzerland	6	6	6	83,33%	2,00	00:00:20
7	Ireland	5	5	6	66,67%	1,87	00:00:10
9	France	4	3	4	25,00%	7,00	00:02:06
10	Albania	3	2	3	66,67%	2,00	00:00:10
11	Pakistan	3	3	4	75,00%	1,25	00:00:25
12	Austria	2	2	2	50,00%	2,00	00:00:06
13	Belgium	2	2	2	50,00%	4,00	00:00:48
14	Canada	2	2	2	50,00%	8,00	00:02:24
15	Czechia	2	2	3	66,67%	7,33	00:00:00
17	Netherlands	2	2	2	50,00%	2,00	00:00:15
18	Poland	2	2	3	66,67%	1,33	00:00:00
19	Singapore	2	2	2	50,00%	3,00	00:07:04
20	Brazil	1	1	1	0,00%	6,00	00:01:49
22	Finland	1	1	1	0,00%	4,00	00:00:06
24	Hong Kong	1	1	1	0,00%	4,00	00:00:54
26	South Korea	1	1	1	0,00%	2,00	00:00:48
27	Luxembourg	1	1	1	0,00%	25,00	00:04:13
29	Slovenia	1	1	1	0,00%	3,00	00:00:14

Figura 3.17 – Il dataset "Località", che mostra la provenienza del traffico dati. A sinistra il dataset puro, a destra la sua versione da cui sono stati eliminati i dati del traffico con una frequenza di rimbalzo del 100%

pagina del sito e ne sono usciti senza effettuare altre azioni. Rappresenta, in pratica, gli utenti che hanno aperto la pagina per errore e che non avevano interesse a visualizzarla. Questo traffico è, quindi, sostanzialmente nullo, e considerarlo nel computo sarebbe un errore. Per avere una visualizzazione più veritiera bisogna rimuovere questi i dati. Il dataset privo di queste occorrenze è in figura 3.17, destra¹⁰.

La maniera più usuale per visualizzare questo tipo di dati è il grafico a *torta*, che è in sostanza un circolo rappresentante l'insieme dei dati suddiviso in varie sezioni colorate, con ogni sezione corrispondente ad un valore. Se svuotiamo la parte centrale di questo cerchio otteniamo una versione leggermente differente ma nella sostanza uguale: il grafico a *ciambella* (figura 3.18).

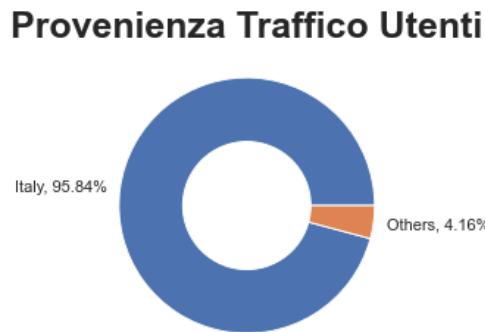


Figura 3.18 – Grafico a ciambella per la visualizzazione della provenienza del traffico utenti.

¹⁰il dataset ottenuto con l'importazione in Excel e la successiva organizzazione con Python non viene modificato, ma solamente visualizzato in maniera differente. I dati della frequenza di rimbalzo pari al 100%, cioè, non vengono eliminati dal dataset originale, ma solamente nascosti.

I proprietari dell'attività commerciale in analisi hanno la volontà di espandere il proprio mercato anche oltre i confini Italiani? Questa domanda è importante per poter leggere al meglio questi dati: se la risposta è "si" allora sono evidentemente necessari aggiustamenti al sito per far sì che venga raggiunta una quota molto maggiore di utenti all'estero. Se così non fosse, invece, il traffico pressoché totalmente proveniente dall'Italia è un dato coerente con le aspettative. Ma il traffico extra Italia da dove proviene? Per non lasciare nulla al caso, e per fornire informazioni quanto più dettagliate possibili, ho aggiunto un grafico, visibile in figura 3.19. Il grafico, cosiddetto "*a bolle*", è costruito a partire dal dataset di figura 3.17, destra (ma privo dei dati riferiti al traffico Italiano), e mostra un cerchio per ogni località, con il nome e la percentuale di traffico. La dimensione dei cerchi - delle bolle - è proporzionale al valore contenuto, fornendo così una visualizzazione rapida ed organica del peso del traffico extra italiano (che rappresenta il 95,84%, come specificato nel titolo del grafico). Questo grafico, però, va letto con molta attenzione. È prassi comune per molti utenti del web, infatti, navigare usufruendo di un protocollo *VPN*, per mascherare la propria provenienza. I motivi sono molteplici e la loro trattazione esula dallo scopo di questo lavoro, ma non si può non tenere in conto di questo aspetto se si vuole dare un'interpretazione corretta dei dati analizzati. Molto probabilmente il traffico extra Italia, quindi, non è altro che un traffico dati Italiano effettuato attraverso un canale *VPN*.

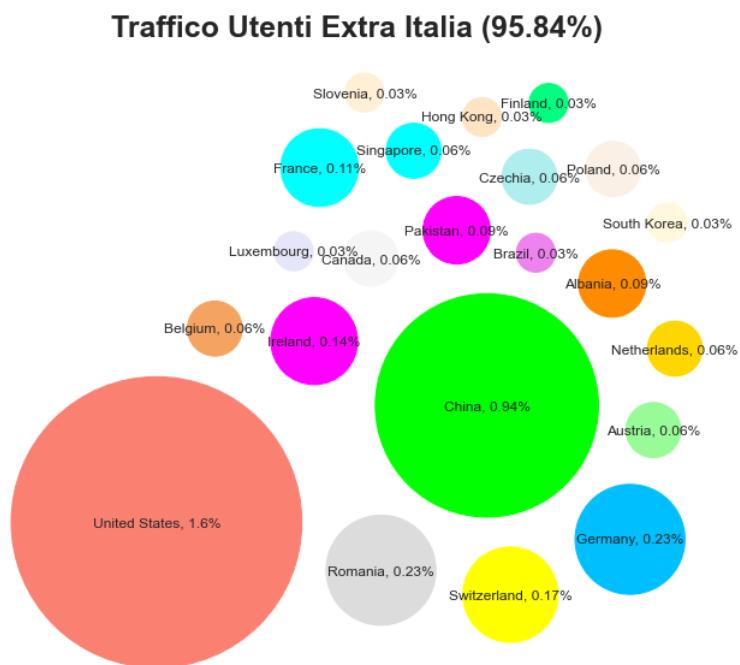


Figura 3.19 – Grafico a bolle per la visualizzazione del traffico extra Italia.

3.6.4 Caratterizzazione del traffico per sezioni: barplot

Siccome lo scopo principale dell'analisi in oggetto è quello di comprendere quale sia l'interesse degli utenti verso gli articoli presentati nel sito, è utile suddividere le informazioni secondo lo schema del sito, lo stesso descritto nel paragrafo 3.1, facendo riferimento in particolare alla tabella 3.2. Partendo dal dataset "tutte le pagine" ho preso quelle delle sezioni, e ho valutato quale fosse il numero di visualizzazioni per ogni sezione (cfr.figura 3.11). Con un semplice grafico a barre - (figura 3.20) - è facile notare che le sezioni di gran lunga più visualizzate sono "Illuminazioni per interni" e "Esterno".

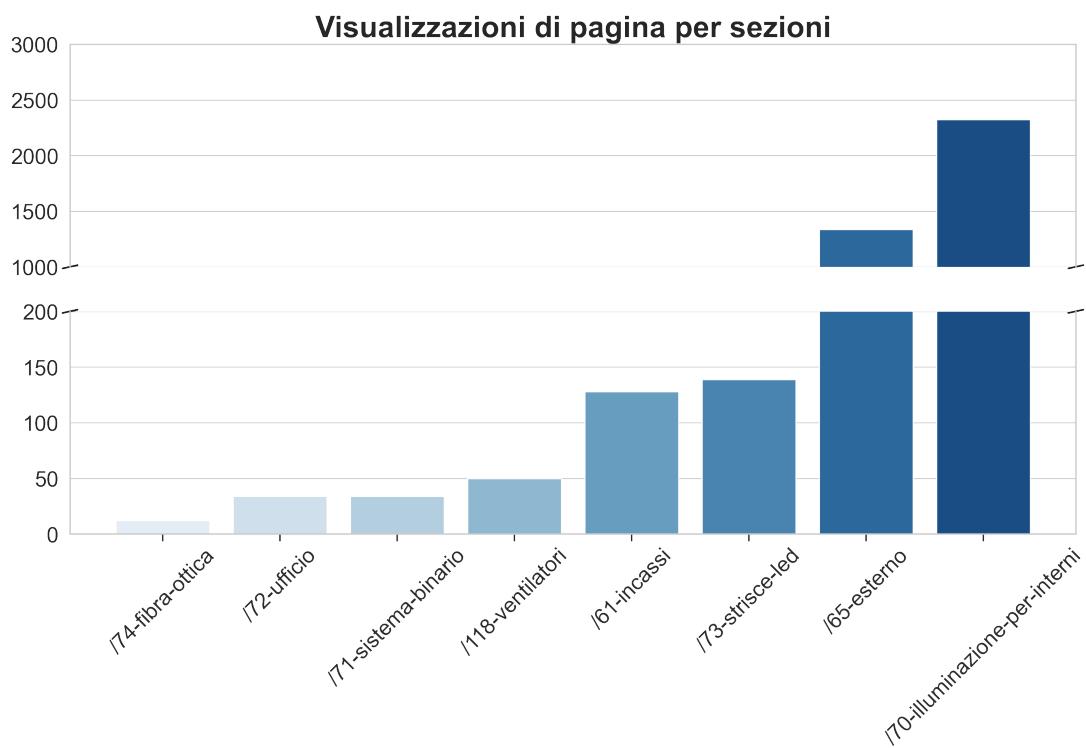


Figura 3.20 – Numero delle visualizzazioni del sito in funzione delle sezioni in cui è suddiviso.

Questo dato dice che gli utenti del sito sono molto più interessati a due tipologie di prodotti rispetto agli altri, e mi ha indirizzato verso un'osservazione più puntuale di queste due sezioni. Nello specifico mi sono chiesto: posso trarre informazioni aggiuntive dall'analisi approfondita di tutti gli articoli di queste due sezioni?

3.6.5 Le sottosezioni di "Illuminazioni per interni"

Per ottenere una caratterizzazione ancora più profonda del traffico dati, quindi, ho costruito un nuovo dataset a partire da quello che mostra il dettaglio di *tutte* le pagine

del sito appartenenti alla sezione "Illuminazioni per interni". Come è visibile in tabella 3.2, la sezione "Illuminazioni per interni" possiede cinque sottosezioni, e ogni sottosezione ha, a sua volta, un numero variabile di sotto-sottosezioni. Per avere un quadro più completo ci rifacciamo alla figura 3.21, che descrive la struttura di tutte le sezioni del sito tramite un semplice albero gerarchico¹¹.



Figura 3.21 – Albero gerarchico delle sezioni in cui è diviso il sito web

Per ottenere questo dataset ho utilizzato *BeautifulSoup*, un pacchetto di Python che permette di effettuare web scraping con notevole facilità¹². Il codice per ottenere l'elenco di tutte le pagine è abbastanza lungo (circa seicento righe), e ne descrivo qui solo i passaggi fondamentali:

1. importo i pacchetti necessari: `request`, `BeautifulSoup` as `bs`.
2. imposto una richiesta: il pacchetto si collega al sito web indicato.

¹¹costruito con il pacchetto *tree*.

¹²il web scraping è la raccolta di informazioni dal web, l'operazione cioè di scaricare e manipolare i contenuti di un sito web. *BeautifulSoup* è l'esempio perfetto di un pacchetto scritto da terze parti ma che è oramai imprescindibile per chiunque utilizzi questo linguaggio di programmazione (cfr. paragrafo 1.3.1).

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

3. trasformo il contenuto della pagina in testo.
4. applico un loop: il pacchetto `bs` cerca tutti gli argomenti della pagina html che sono classificati come "a", ne raccoglie il link corrispondente e lo inserisce in una lista.
5. costruisco un nuovo loop che ripeta l'operazione nelle pagine seguenti della sottosezione.
6. rimuovo tutte le pagine non necessarie (link a siti esterni, header).

3.7 Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

Una volta ottenuto il dataset su cui lavorare, mi sono concentrato sugli accessi dei visitatori. Il primo grafico possibile è un istogramma, in figura 3.22.

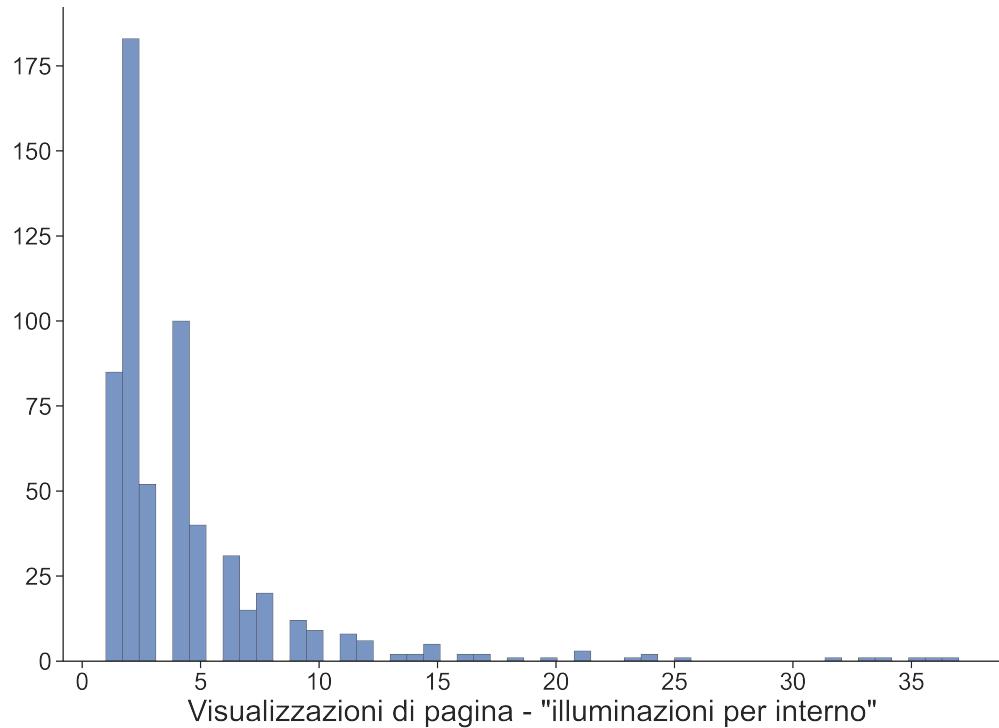


Figura 3.22 – Istogramma delle visualizzazioni di pagina.

Il grafico ci dice che le pagine con un numero di visualizzazioni basse (da 0 a 10) è molto alto, mentre risultano essere molte di meno le pagine con più di dieci visualizzazioni. Ma quante di meno? Per ottenere una visuale più organica si potrebbe

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

spezzare l'asse delle y, ma si può rendere il grafico più leggibile anche con un'altra strategia: impostare l'asse x con una scala logaritmica, come in figura 3.23.

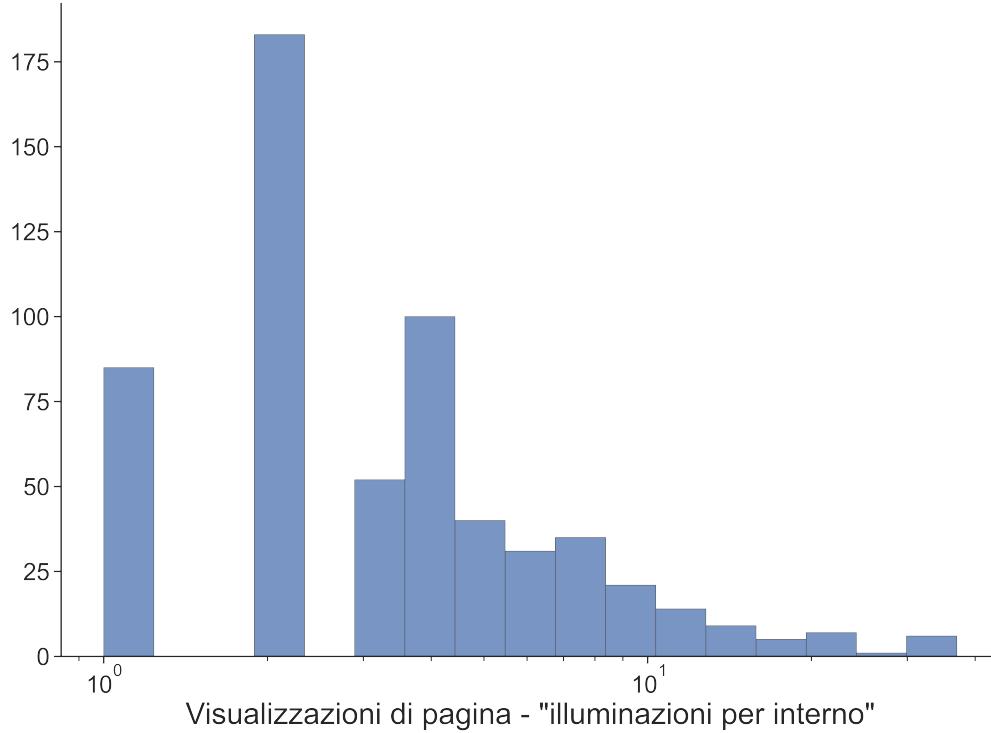


Figura 3.23 – Istogramma delle visualizzazioni di pagina in scala logaritmica.

In questa modalità il grafico risulta sicuramente molto più leggibile, ma non ancora al meglio delle possibilità. Ho trasformato ancora il grafico, allora, cambiandolo in uno scatterplot dimensionale. Questo tipo di grafico correla due valori (asse x e asse y) in un unico punto, e aggiunge un terzo parametro, tipicamente un parametro di intensità, che è determinato dalla grandezza dei punti stessi. Se si osserva in figura 3.24 si nota che le visualizzazioni di pagina sono plottate in relazione alla frequenza di rimbalzo, e i punti sono correlati con il numero delle visualizzazioni. Il risultato è un grafico in cui i punti sono posizionati nello spazio cartesiano a formare tre aree¹³. La prima area si posiziona in basso a sinistra, con valori di accessi molto bassi e frequenza di rimbalzo alta. Una seconda area, invece, si colloca in basso a destra, e rappresenta le pagine con accessi bassi ma frequenza di rimbalzo parimenti bassa. Una terza area di punti è quella che si discosta dall'asse x, e va a collocarsi nello spazio centro/superiore del piano Cartesiano. Questa area rappresenta le pagine che hanno avuto un numero alto di visualizzazioni e una frequenza di rimbalzo ragionevole (entro il 50%). Sono questi i punti che maggiormente ci devono incuriosire, perché essi rappresentano persone che

¹³escludendo alcuni dati outsider.

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

hanno visitato il sito per più volte, dirigendosi verso la medesima pagina, e non l'hanno abbandonata senza prima aver osservato e interagito con la stessa.

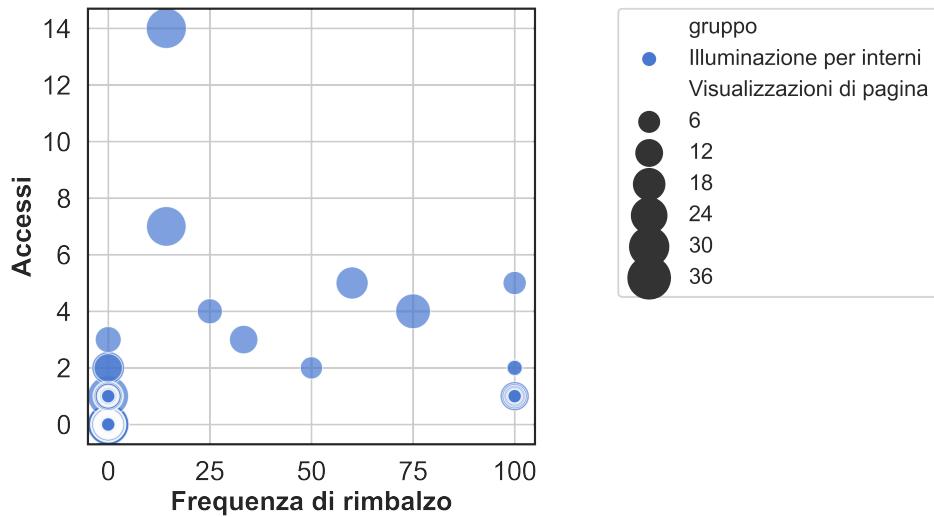


Figura 3.24 – Scatterplot delle visualizzazioni di pagina della sezione "Illuminazioni per interni".

3.7.1 Stripplot e pairlot: le visualizzazioni suddivise per decadi

Il nuovo dataset getta una luce diversa sul traffico più influente diretto verso il sito web. Riguardando la figura 3.23, vediamo che le pagine della sottosezione "Illuminazione per interni" vanno da valori molto bassi - poche unità - fino a più di trenta visualizzazioni. Da quest'osservazione l'idea: probabilmente le caratteristiche di queste sessioni sono differenti se analizzate in funzione del numero stesso di visualizzazioni. Ho diviso quindi il traffico in tre parti, che ho definito "decadi": le pagine che hanno avuto da 0 a 3 visualizzazioni, quelle che hanno avuto da 3 a 30 visualizzazioni e quelle che hanno avuto più di 30 visualizzazioni. Per prima cosa ho implementato un grafico che potesse mostrare la quantità relativa (figura 3.25).

Da questo grafico si nota che sono più rappresentate le pagine che hanno avuto una sola visualizzazione (punti verdi) o una decina di visualizzazioni (punti arancioni), mentre sono in numero molto minore quelle che sono state visitate trenta volte. Molto interessante è anche il medesimo tipo di plot quando sull'asse delle x impostiamo la frequenza di rimbalzo, come in figura 3.26 (in questo caso non è necessario impostare la scala logaritmica). Si nota come le pagine che vengono visualizzate poche volte si dividono esclusivamente in pagine con una frequenza di rimbalzo pari al 100% - quindi pagine in cui non c'è stata alcuna interazione - e pagine con la frequenza di rimbalzo pari

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

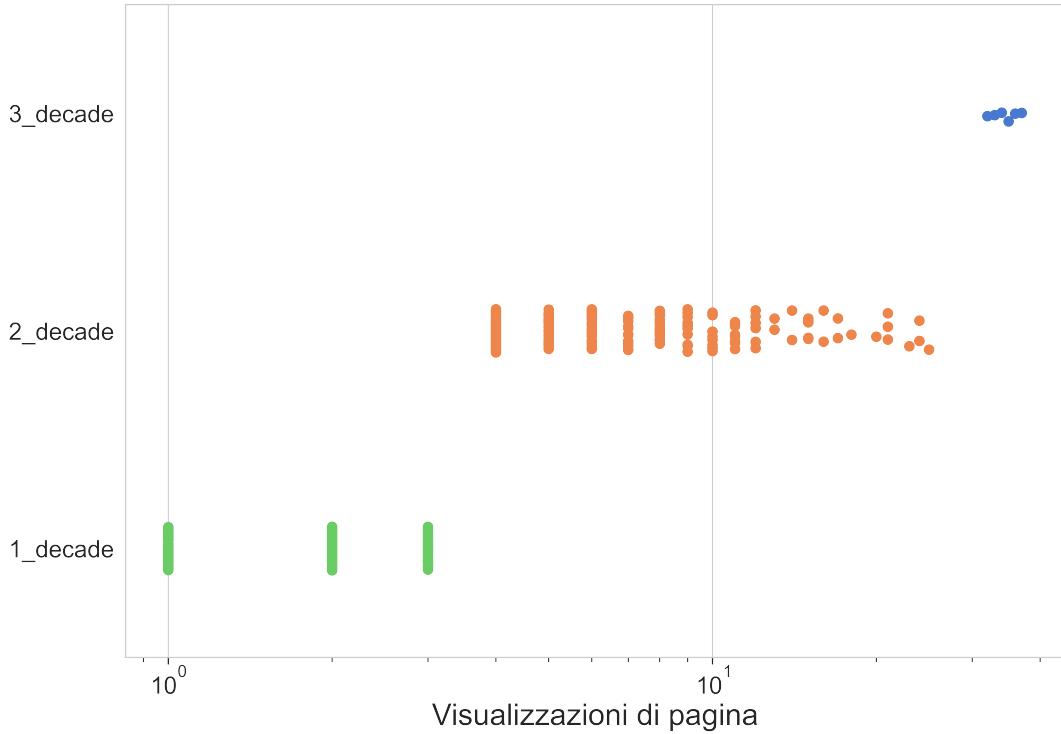


Figura 3.25 – Stripplot delle visualizzazioni di pagina della sottosezione "Illuminazioni per interni" - numero di visualizzazioni per pagina.

a zero, ossia pagine con una e una sola visualizzazione, di tempo variabile. Le pagine che sono state visualizzate tra 3 e 30 volte, invece, mostrano anche un comportamento intermedio. Infine le pagine che hanno più 30 visualizzazioni non presentano frequenza di rimbalzo alta. È sempre più evidente, quindi, che sia necessario visualizzare il traffico verso le pagine in maniera differenziata in base al numero di visualizzazioni.

3.7.2 Visualizzazioni delle pagine suddivisi per decadi

Gli ultimi grafici implementati riguardo alla metrica "visualizzazione" sono rivolti a una più approfondita comprensione delle dinamiche di navigazioni sulle varie pagine del sito web. Essi sono grafici multipli, ossia formati da più grafici in relazione tra di loro. Nello specifico (figura 3.27) esso è formato da sei grafici: tre grafici di distribuzione e tre grafici scatterplot.

I sei grafici mettono a paragone, in tutte le combinazioni possibili, i tre parametri analizzati: il numero di visualizzazioni, la frequenza di rimbalzo e la percentuale di uscita. Quest'ultimo parametro, che fino ad ora non è stato tenuto in considerazione, rappresenta quante volte gli utenti hanno scelto di abbandonare il sito chiudendo proprio la pagina in esame. Quando in un grafico sono messi a paragone parametri diversi

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

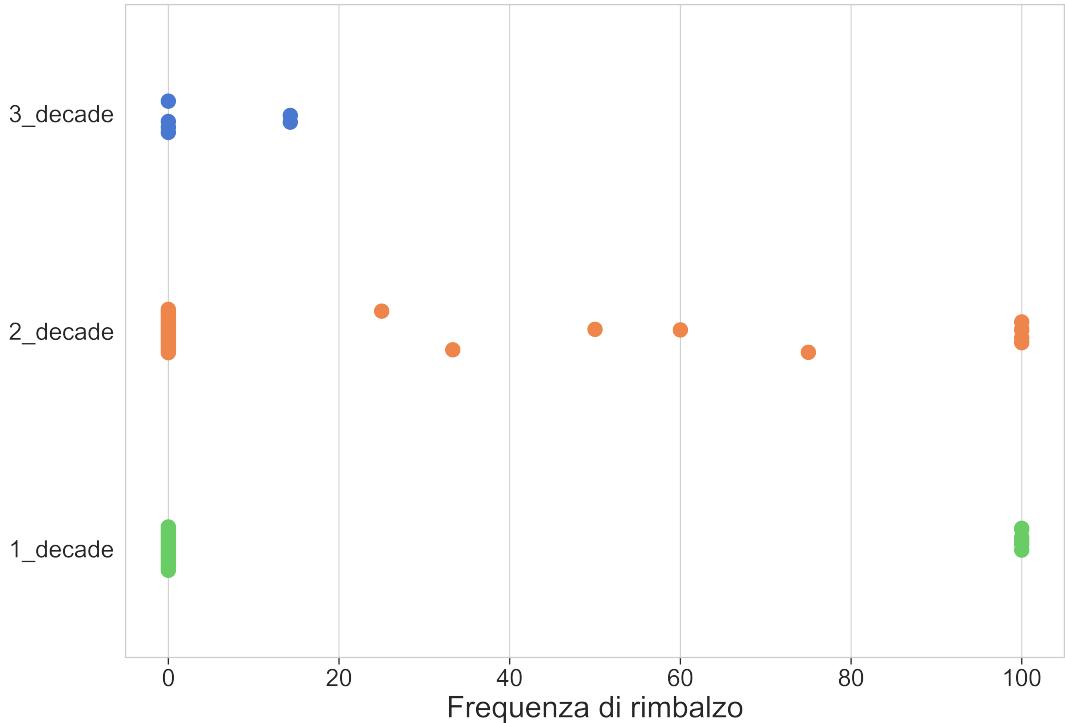


Figura 3.26 – Stripplot delle visualizzazioni di pagina della sottosezione "Illuminazioni per interni" - frequenza di rimbalzo.

esso è un grafico di scatterplot, mentre quando si paragona lo stesso parametro il grafico è un grafico di distribuzione. Analizzando la figura vediamo che i tre grafici di distribuzione mostrano un picco in prossimità di valori vicini allo zero, mentre i scatterplot mostrano i punti schiacciati verso le estremità inferiore dei piani cartesiani. Ho quindi selezionato i dati e li ho ripioltati in tre grafici diversi, seguendo il medesimo schema di separazione per decadi utilizzato per i grafici in figura 3.25 e 3.26. Li vediamo e li analizziamo uno per uno.

Il grafico di figura 3.28 mostra i dati delle pagine che hanno ricevuto un numero di visualizzazioni tra 0 e 3.

L'istogramma del numero di visualizzazioni mostra tre picchi, quasi a significare una preferenza media per il numero di volte in cui si visualizza una pagina. La realtà è che tra 0 e 3 visualizzazioni il range è molto stretto, è questo tipo di differenze non può essere considerata significativa. Più senso ha il grafico di distribuzione della frequenza di rimbalzo, che ci mostra come essa sia praticamente tutta a valore 0. questo dato significa che, in questa specifica sezione, gli utenti che visitano le pagine lo fanno motivati da un reale interesse per i prodotti. Il terzo grafico di distribuzione, che mostra la % di uscita dalla pagina, mostra un andamento simile, a significare che sono

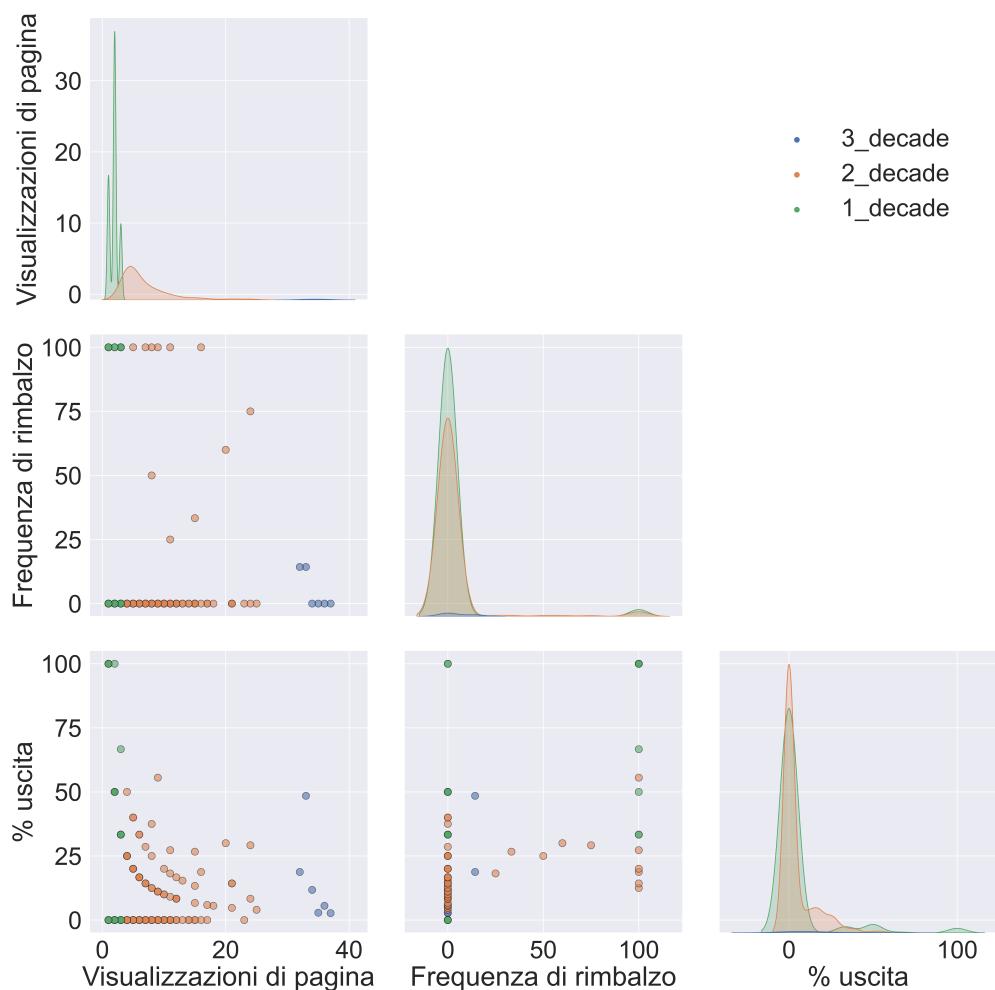


Figura 3.27 – Pairplot delle visualizzazioni di pagina delle sottosezioni di "Illuminazioni per interni"

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

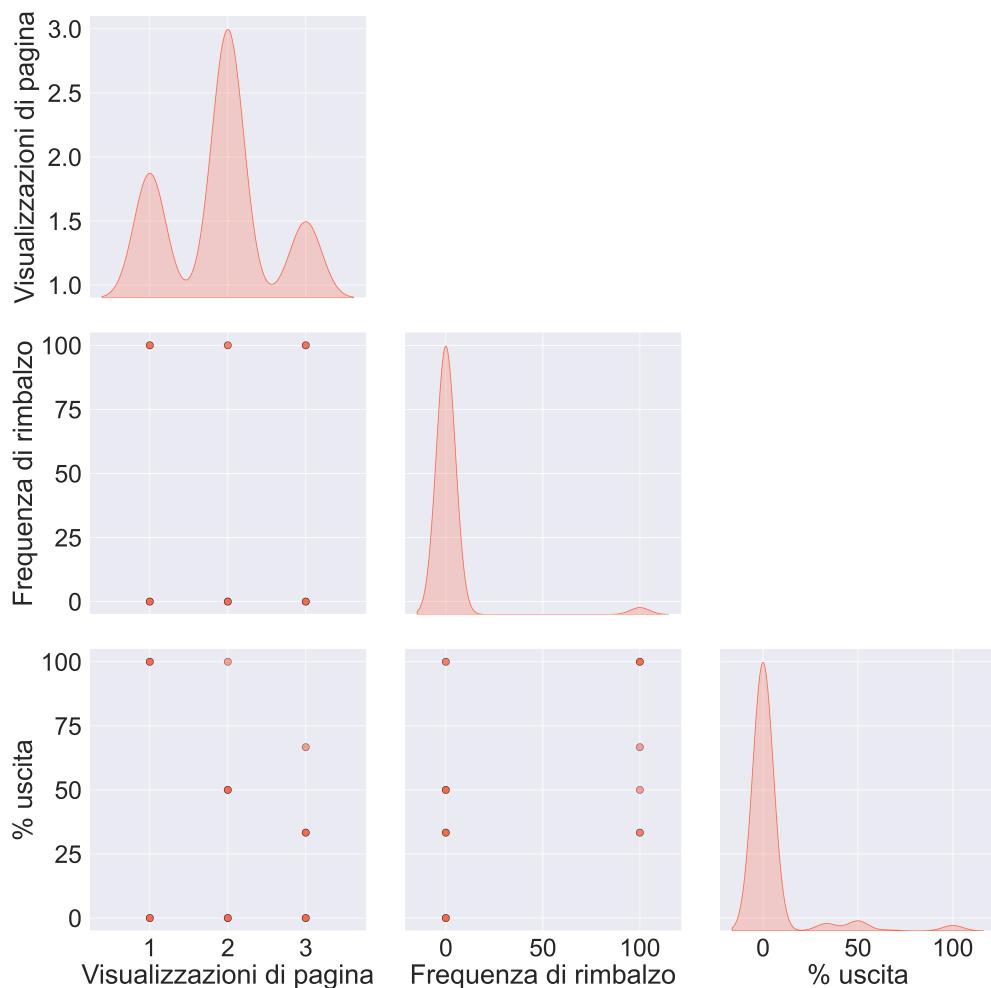


Figura 3.28 – Pairplot delle visualizzazioni di pagina delle sottosezioni di "Illuminazioni per interni" - prima decade

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

poche le esperienze di visita di queste pagine che si sono concluse con l'abbandono del sito. Analizziamo ora i scatterplot. La frequenza di rimbalzo vista in funzione del numero di visualizzazioni mostra tre punti al 100% e tre punti allo 0%. Contrariamente a quanto era immaginabile dall'osservazione della sola distribuzione, quindi, notiamo che anche sulle pagine a basso numero di visualizzazioni esiste una quota significativa di frequenza di rimbalzo. Situazione analoga si manifesta nel rapporto tra frequenza di rimbalzo e % di uscita dalla pagina, con una dicotomia esatta tra 0 e 100%. Il terzo e ultimo scatterplot ci fa notare come correlino la percentuale di uscita con il numero di visualizzazioni di una pagina. In questo caso vediamo una situazione più fluida, con valori intermedi tra lo 0 e il 100.

Il grafico di figura 3.29 mostra i dati delle pagine che hanno ricevuto un numero di visualizzazioni tra 3 e 30.

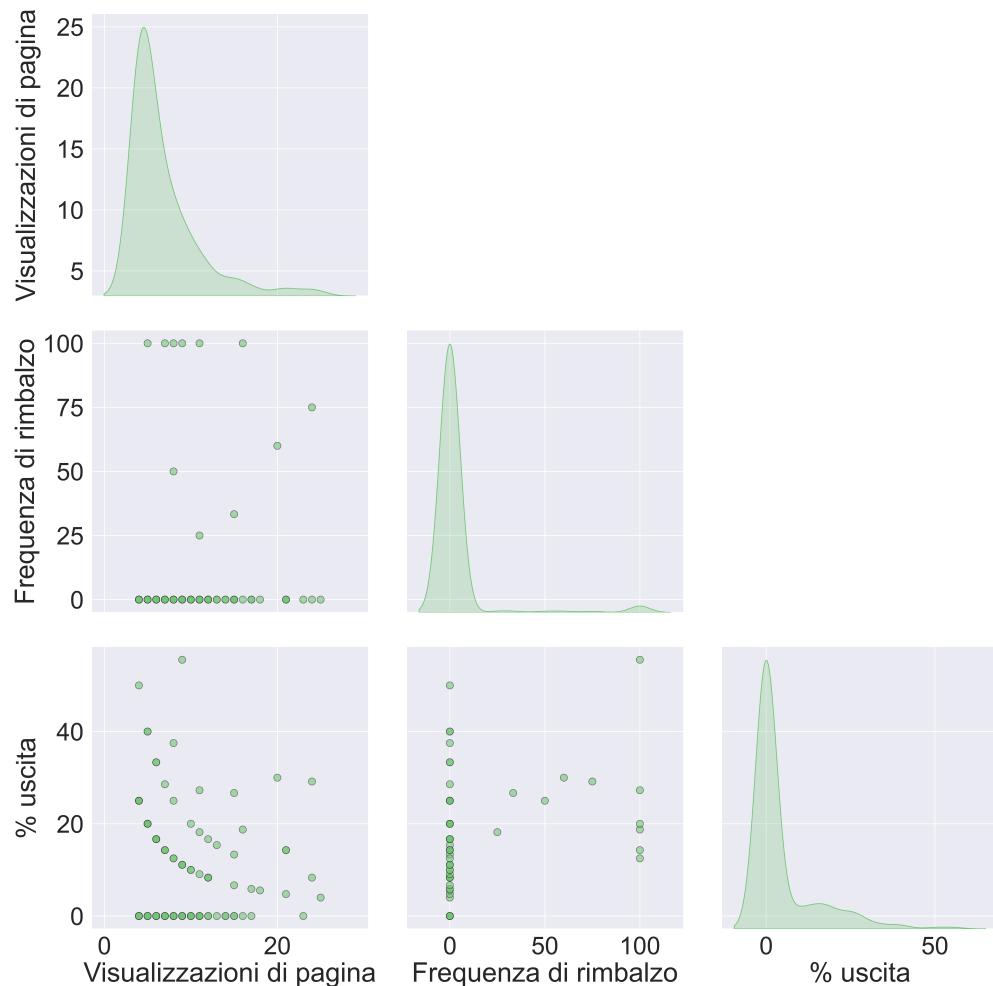


Figura 3.29 – Pairplot delle visualizzazioni di pagina delle sottosezioni di "Illuminazioni per interni" - seconda decade

I tre istogrammi cominciano a differire rispetto ai valori della prima decade. Il

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

numero di visualizzazioni non è più netto, ma si allarga notevolmente, assomigliando a una campana gaussiana, ma ha comunque il suo centro nel valore minimo. L'istogramma della frequenza di rimbalzo si presenta alla stessa maniera, anche se è visibile un picco al valore massimo - sono gli utenti della frequenza di rimbalzo pari al 100%. La percentuale di uscita ha un grafico un po' più sporco, con la gaussiana che presenta una coda molto accentuata. I tre scatterplot ci raccontano tutta una storia diversa: la frequenza di rimbalzo, quando vista nei confronti del numero di visualizzazioni, presenta molti valori pari allo zero, meno valori al 100% e alcuni valori intermedi. Questo significa che gli utenti che hanno visualizzato queste pagine specifiche hanno mostrato un tasso di abbandono immediato molto minore, frutto verosimilmente di un interesse maggiore. Lo scatterplot che correla la frequenza di rimbalzo con la % di uscita mostra un quadro simile: gli utenti con la frequenza di rimbalzo nulla mostrano una % di uscita molto variabile. L'ultimo scatterplot invece, che correla le visualizzazioni di pagina alla % di uscita, mostra una situazione molto più variabile e scatterata.

Il grafico di figura 3.30 mostra i dati delle pagine che hanno ricevuto un numero di visualizzazioni oltre i 30.

I dati della terza decade differiscono ulteriormente: l'istogramma del numero di visualizzazioni di pagine è in pratica molto simile a una gaussiana, pur presentando un piccolo picco secondario spostato a valori più alti. Stessa conformazione ha l'istogramma della frequenza di rimbalzo, ma il secondo picco è più marcato. La gaussiana della % di uscita, infine, ha una forma quasi perfetta. I tre scatterplot ci mostrano punti che quasi mai si posizionano all'estremità superiore del quadro cartesiano, non assumendo quindi mai valori massimi. Nello specifico lo scatterplot che correla la frequenza di rimbalzo e il numero di visualizzazioni vede i punti schiacciati sull'estremità inferiore dell'asse y (a parte due punti). Lo scatterplot che mostra le visualizzazioni di pagina rapportate alla % di uscita ha la medesima impostazione, ma con un pattern meno ordinato. Lo scatterplot che vede la frequenza di rimbalzo rapportata alla % di uscita, infine, ha uno schema molto simile a quello che correla frequenza di rimbalzo e visualizzazioni di pagina, con i punti divisi a metà tra l'estremità inferiore e il centro del quadrante. L'osservazione e l'analisi di questo ultimo grafico richiedono, però, di una considerazione aggiuntiva. Qualsiasi analisi statistica si fonda sul concetto di *popolazione*, che è il nome usato per indicare l'ammontare dei dati analizzati. Perché una statistica - di qualsiasi genere - sia valida, è necessario assicurarsi che la popolazione

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

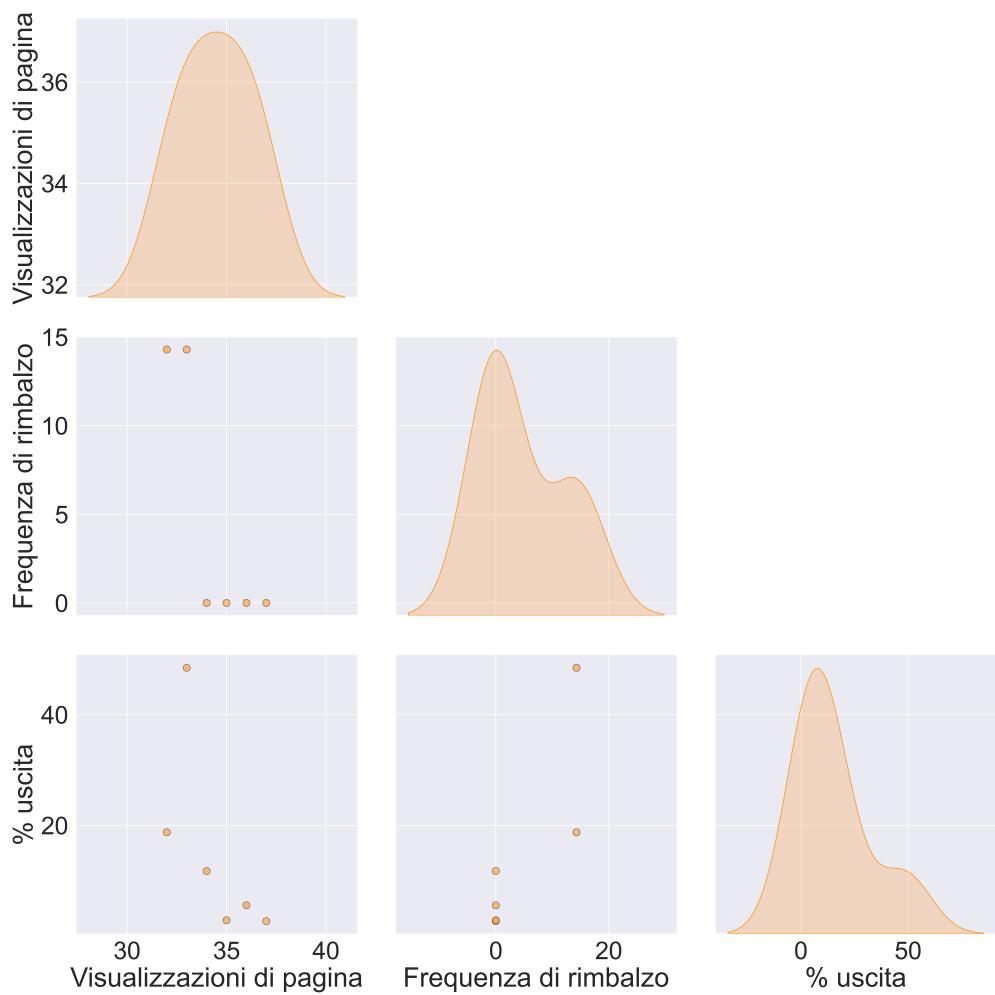


Figura 3.30 – Pairplot delle visualizzazioni di pagina delle sottosezioni di "Illuminazioni per interni" - terza decade

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

rispetti alcuni canoni. Molti sono i test che si possono effettuare per ottenere tale valutazione, e di sicuro il più semplice e immediato è quello che riguarda il numero dei dati. Perché una popolazione sia valutabile, occorre che essa presenti almeno un centinaio di valori. I dati della terza decade hanno un numero molto esiguo (sono 6). Ecco che l'analista di dati si pone una domanda aggiuntiva: ha senso analizzare questi dati a se stanti? Probabilmente no, non essendo un campione rappresentativo. Ho scelto quindi di plottarli solamente per mostrare il paragone con i grafici precedenti, ma sono consapevole che non possono fornire informazioni solide.

In chiusura di questa lunga osservazione sul traffico delle pagine della sezione "Illuminazioni per interni" possiamo precisare che i grafici hanno una doppia funzione: permettono di visualizzare l'andamento generale e particolare dei dati, e contemporaneamente permettono una selezione degli stessi. Se il cammino logico seguito fino ad ora è stato quello "dal dataset al grafico", possiamo anche percorrerlo in senso inverso, selezionando dal grafico alcune occorrenze del dataset che lo ha generato. Se il grafico ci ha mostrato, ad esempio, alcuni punti il cui comportamento ci ha incuriosito, con poche semplici righe possiamo andare a controllare quali pagine siano le responsabili di questo comportamento. Ad esempio, riferendoci alla figura 3.29 (lo scatterplot che correla la frequenza di rimbalzo al numero di visualizzazioni di pagina), notiamo che ci sono pochi punti con una frequenza di rimbalzo molto alta; significa che in un contesto generale di pagine visitate e non immediatamente abbandonate, un numero esiguo di queste ha spinto gli utenti a lasciare immediatamente il sito. Quali sono queste pagine? Basta andare a recuperare dal dataset queste pagine, con una sintassi molto semplice: `df_2_dec[(df_2_dec['Frequenza di rimbalzo'] > 75)]`, che si legge in questo modo: seleziona dal dataset chiamato `df_2_dec` tutte le occorrenze che, nella colonna "Frequenza di rimbalzo", abbiano un valore superiore a 75. La figura 3.31 mostra il risultato. Il dataset ottenuto presenta, ovviamente, sei occorrenze, tanti quanti sono i punti visibili nel grafico. Una volta conosciute le pagine possiamo informare il committente che ragionerà su eventuali azioni da intraprendere nei confronti di questi articoli.

I comandi per selezionare i punti dal grafico sono molto semplici, e *pandas* è molto versatile su queste operazioni, consentendo l'utilizzo di parametri multipli per la selezione (esempio: selezioniamo tutti i punti che hanno frequenza di rimbalzo tra 40 e 50 e, contemporaneamente, % di uscita non inferiore al 46%).

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

In [1804]:	df_2_dec[(df_2_dec['Frequenza di rimbalzo'] > 75)]				
Out[1804]:					
Visualizzazioni di pagina Frequenza di rimbalzo % uscita gruppo_visualizzazione gruppo					
Pagina					
/led/2177-discovery.html	16.0	100.0	18.75	2_decade	Illuminazione per interni
/lampade-soffitto-moderno/1411-martin.html	11.0	100.0	27.27	2_decade	Illuminazione per interni
/lampadari-industriali/1818-ohio.html	9.0	100.0	55.56	2_decade	Illuminazione per interni
/plafoniere-in-gesso/4974-allen.html	8.0	100.0	12.50	2_decade	Illuminazione per interni
/lampade-applique-parete-pitturabili/5021-pula.html	7.0	100.0	14.29	2_decade	Illuminazione per interni
/lampade-parete-modeno/829-wonder.html	5.0	100.0	20.00	2_decade	Illuminazione per interni

Figura 3.31 – Dataset ottenuto dalla selezione dei punti dal grafico.

3.7.3 Le sottosezioni di "Esterno"

Il discorso affrontato per la sezione "Illuminazioni per interni" vale anche per la sezione "Esterno", che rappresenta un'altra grossa fetta di traffico dati diretta verso il sito. Ho costruito quindi lo stesso tipo di dataset contenente tutte le sotto-sottosezioni e gli articoli in vendita per la sezione "Esterno", e ho implementato i medesimi tipi di grafici appena descritti. Andiamo ad analizzarli uno per uno.

In figura 3.32 è mostrato l'istogramma delle visualizzazioni in scala lineare. Essendo il numero di visualizzazioni molto minore rispetto a quelle della sezione "Illuminazioni per interni" il grafico non ha valori molto alti che possono impedire una corretta visualizzazione delle pagine a visualizzazione bassa, e quindi non è necessario ricorrere alla scala logaritmica.

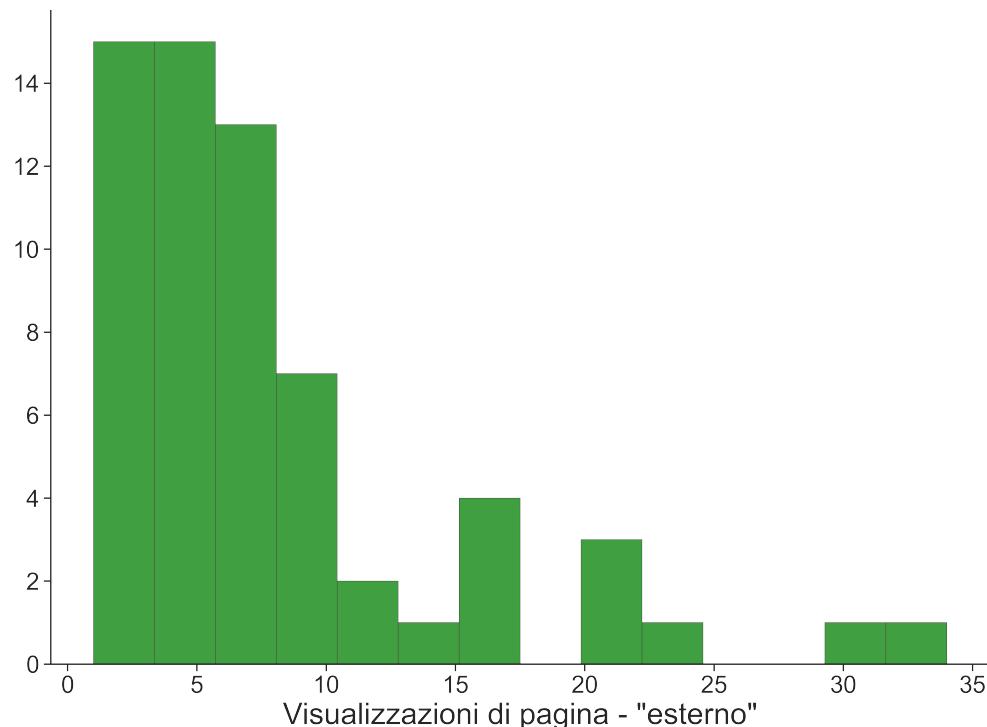


Figura 3.32 – Istogramma delle visualizzazioni di tutte le pagine della sezione "Esterno".

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

Caratterizzando meglio questi dati, in figura 3.33 essi sono rappresentati nello scatterplot dimensionale. Seppur più radi rispetto a quelli della sezione precedente (figura 3.24), essi conservano lo stesso pattern: un gruppo di pagine con bassa frequenza di rimbalzo e basso numero di accessi (in basso a sinistra), un gruppo di pagine con frequenza di rimbalzo maggiore e basso numero di accessi (in basso a destra) e le rimanenti pagine orientate in uno spazio centrale del quadro cartesiano.

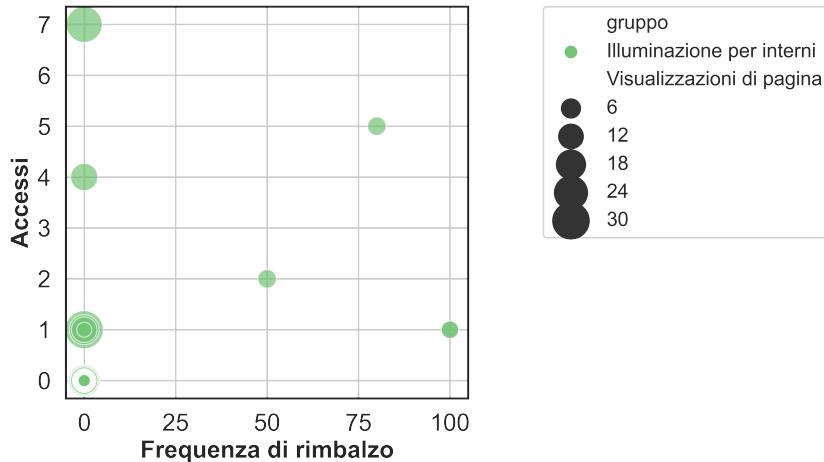


Figura 3.33 – Scatterplot dimensionale delle visualizzazioni di tutte le pagine della sezione "Esterno".

Con i due grafici stripplot (figure 3.34 e 3.35) è possibile valutare la frequenza relativa dei punti, sia per quanto riguarda il numero di visualizzazioni che per la frequenza di rimbalzo. I pattern sono i medesimi della sezione precedente.

Infine, nelle figure 3.36, 3.37 e 3.38 sono presenti i pairplot rispettivamente per tutte le pagine, per le pagine con un numero di visualizzazioni tra 0 e 3, per quelle con un numero di visualizzazioni da 3 a 30. Come già accaduto per la sezione "Illuminazioni per interni", il numero di pagine visualizzate più di 30 volte è così esiguo da non poter assicurare una statistica adeguata. Stavolta non è possibile neppure plottare il grafico, essendoci di fatto una sola pagina nel dataset.

3.7.4 Velocità del sito web

Un'altra metrica interessante è quella della velocità del sito. Questa metrica misura i tempi di connessione degli utenti, ossia quanto il sito impiega a rispondere e a rendersi accessibile. È importante infatti che un utente non debba aspettare per poter visitare una pagina di un prodotto: il tempo di attesa è proporzionale alla probabilità che l'utente lasci il sito, stanco di attendere il caricamento di una pagina che avrebbe dovuto

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

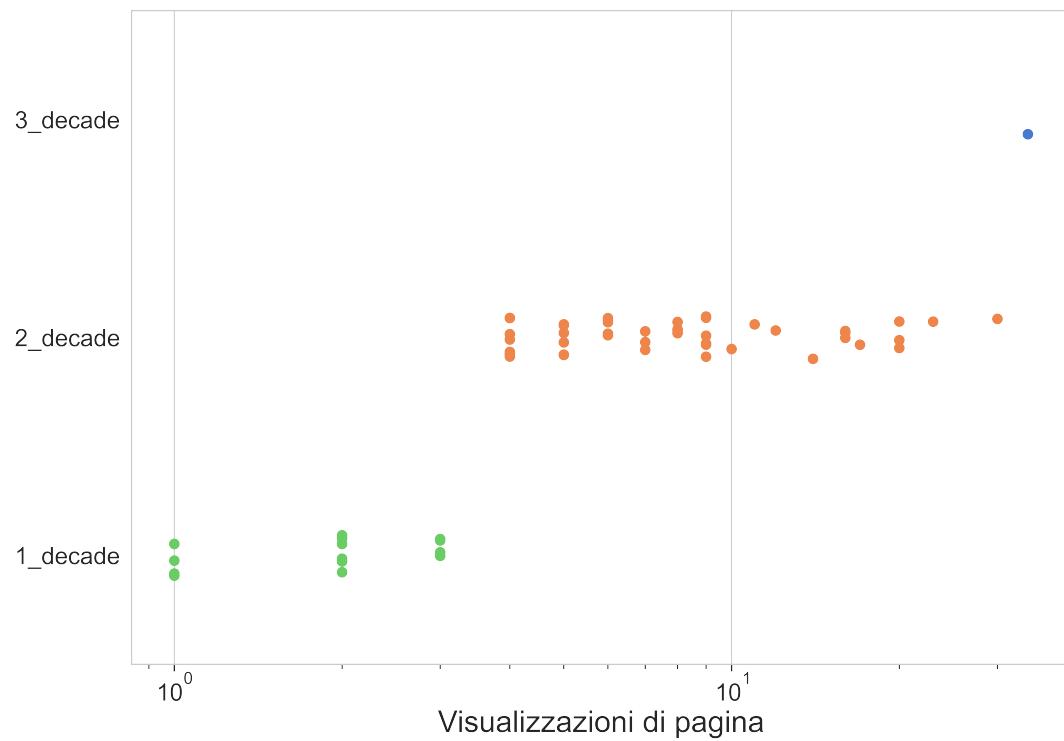


Figura 3.34 – Scatterplot dimensionale delle visualizzazioni di tutte le pagine della sezione "Esterno".

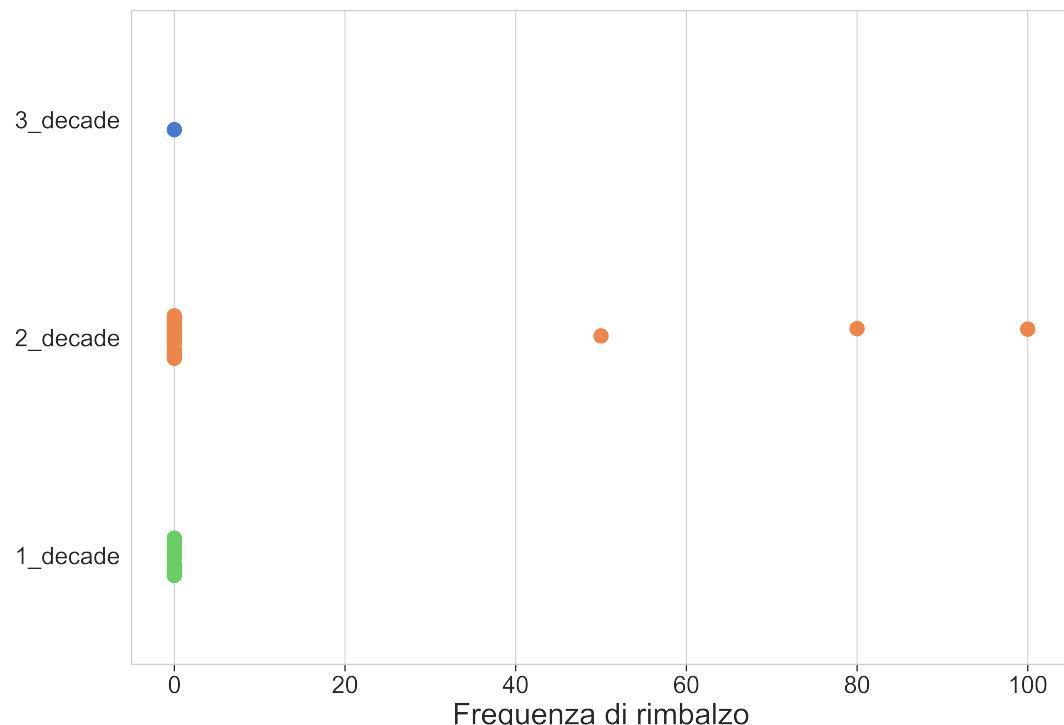


Figura 3.35 – Scatterplot dimensionale delle visualizzazioni di tutte le pagine della sezione "Esterno".

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

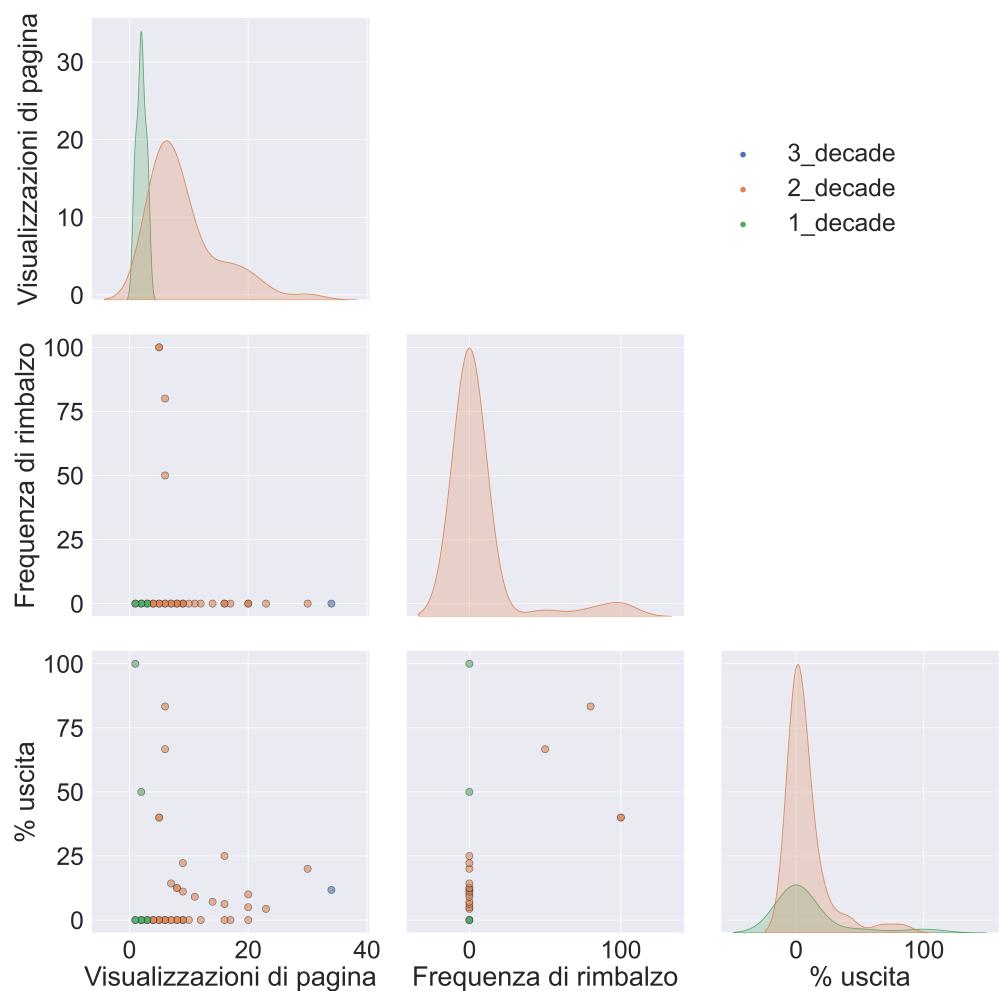


Figura 3.36 – Pairplot delle visualizzazioni di pagina delle sottosezioni di "Esterno"

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

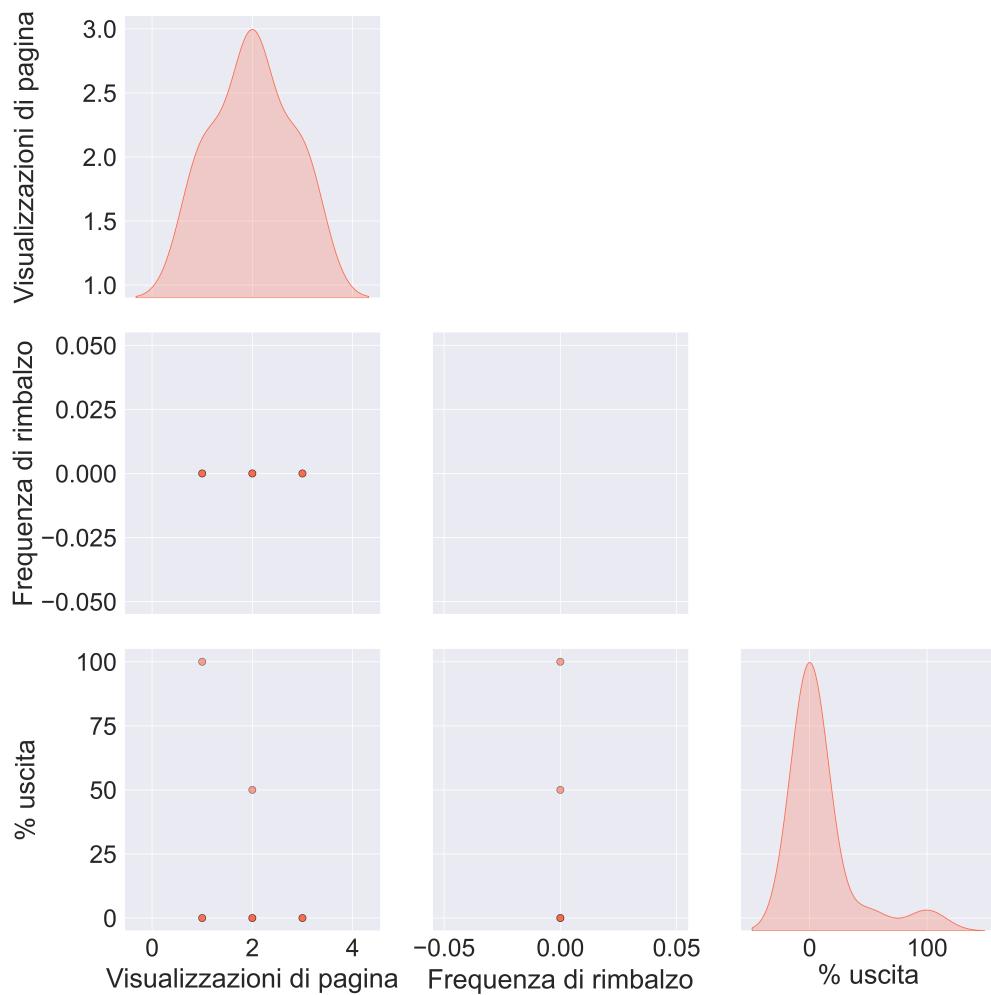


Figura 3.37 – Pairplot delle visualizzazioni di pagina delle sottosezioni di "Esterno" - prima decade

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

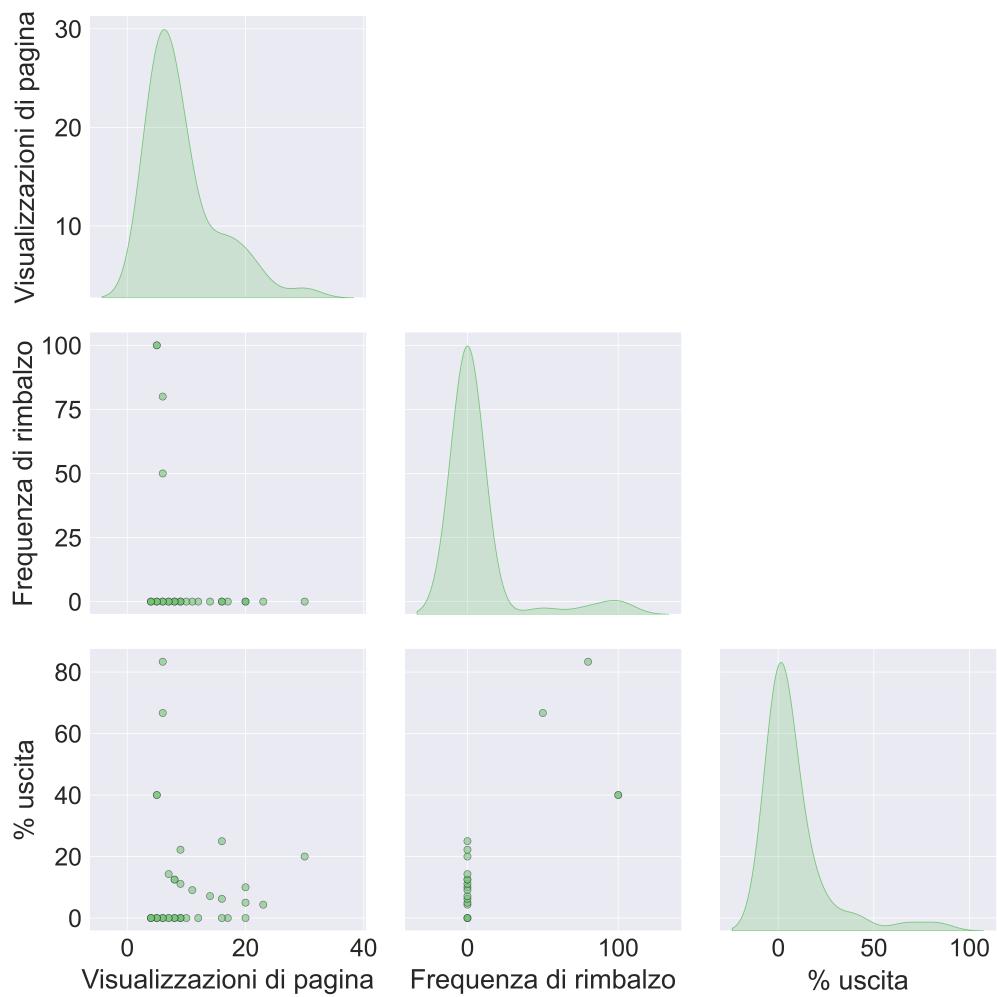


Figura 3.38 – Pairplot delle visualizzazioni di pagina delle sottosezioni di "Esterno" - seconda decade

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

mostraragli un articolo di potenziale interesse. Il dataset che si costruisce a partire dal file csv della suddetta metrica contiene l'elenco di tutte le pagine visualizzate, il tempo medio di caricamento di ciascuna pagina, il numero di visualizzazioni della stessa, la frequenza di rimbalzo e la % di uscita. Siccome non ha senso valutare una metrica di una pagina con cui non c'è stata interazione, per prima cosa ho eliminato le occorrenze con la frequenza di rimbalzo pari al 100%. Dal grafico ottenuto ho selezionato le sezioni del sito e plottato, con un barplot, i tempi medi di caricamento delle pagine delle sezioni (figura 3.39). Con questo approccio ho risposto a una domanda: c'è una sezione con un tempo di caricamento significativamente diverso dalle altre?

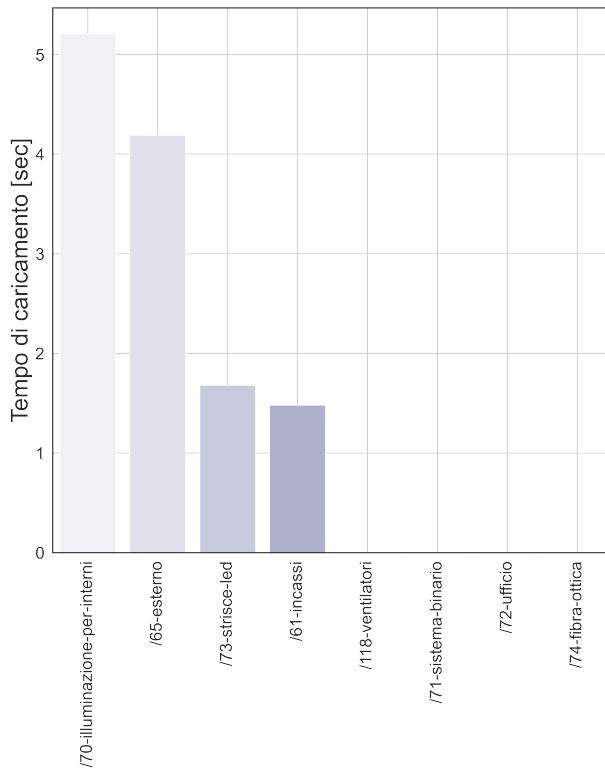


Figura 3.39 – Barplot per la visualizzazione della metrica per la velocità del sito web.

Il grafico ci offre la risposta, e cioè che le sezioni si dividono in due gruppi: un gruppo con il tempo di caricamento medio prossimo allo zero, e un gruppo con tempi di caricamento molto più alti (fino a 5 secondi). Qual è il motivo di questa discrepanza? Le cause possono essere molteplici, una delle più probabili è che le sezioni a tempo maggiore rimandino a molte più pagine da caricare. Possiamo controllare questa ipotesi plottando il numero di articoli presenti in ogni sezione, e sovrapponendolo al grafico precedente¹⁴. Le scale di riferimento sono differenti, ma conservando la dimensione

¹⁴avrei potuto plottare il numero di pagine di ogni sottosezione, ma siccome gli articoli sono presentati in una struttura a griglia, con un numero fisso per pagina, il risultato è il medesimo.

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

generale del grafico si nota un'interessante sovrapposizione (3.40).

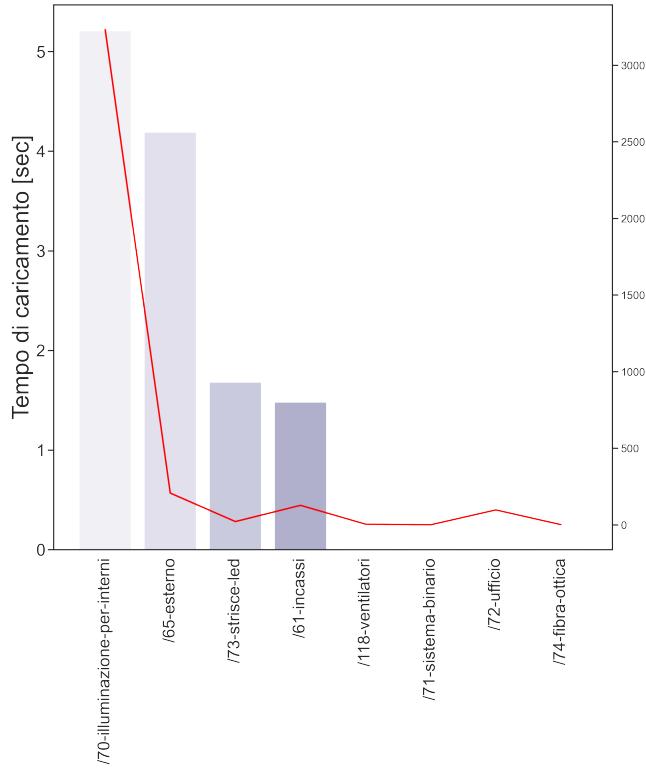


Figura 3.40 – Barplot per la visualizzazione della metrica per la velocità del sito web in relazione al numero di articoli per ogni sezione.

La correlazione quindi sembra avvalorare la tesi sussidata: le sezioni "Illuminazioni per interni", "Esterno", "Strisce Led" e "Incassi" sono significativamente più lente in relazione al numero molto più alto di pagine da aprire rispetto alle altre sezioni. Questo aspetto come influenza la navigazione del sito? Andiamo a controllare il numero di visualizzazioni delle pagine corrispondenti, per valutare se il tempo di attesa maggiore diminuisca il numero di persone disposte a navigare quelle pagine. In figura 3.41 vediamo tre barplot adiacenti: quello già visto del tempo di caricamento (figura 3.39), uno nuovo che mostra il numero di visualizzazioni e uno nuovo che mostra la % di uscita dalla pagina.

La metrica % di uscita assume qui il suo ruolo più importante: quanti utenti abbandonano la pagina perché - presumibilmente - stanchi di attendere troppo il completo caricamento? Dal grafico sembra che "Illuminazione per interni" e soprattutto "Esterno" siano le sezioni più abbandonate, eppure quelle con un numero alto di visualizzazioni. Questi due dati assieme potrebbero insospettire: non sembra esserci scarso interesse nei riguardi dei prodotti, quanto più una probabile problematica di tempistica di caricamento. Indaghiamo ulteriormente allora: quali sono le pagine in assoluto

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

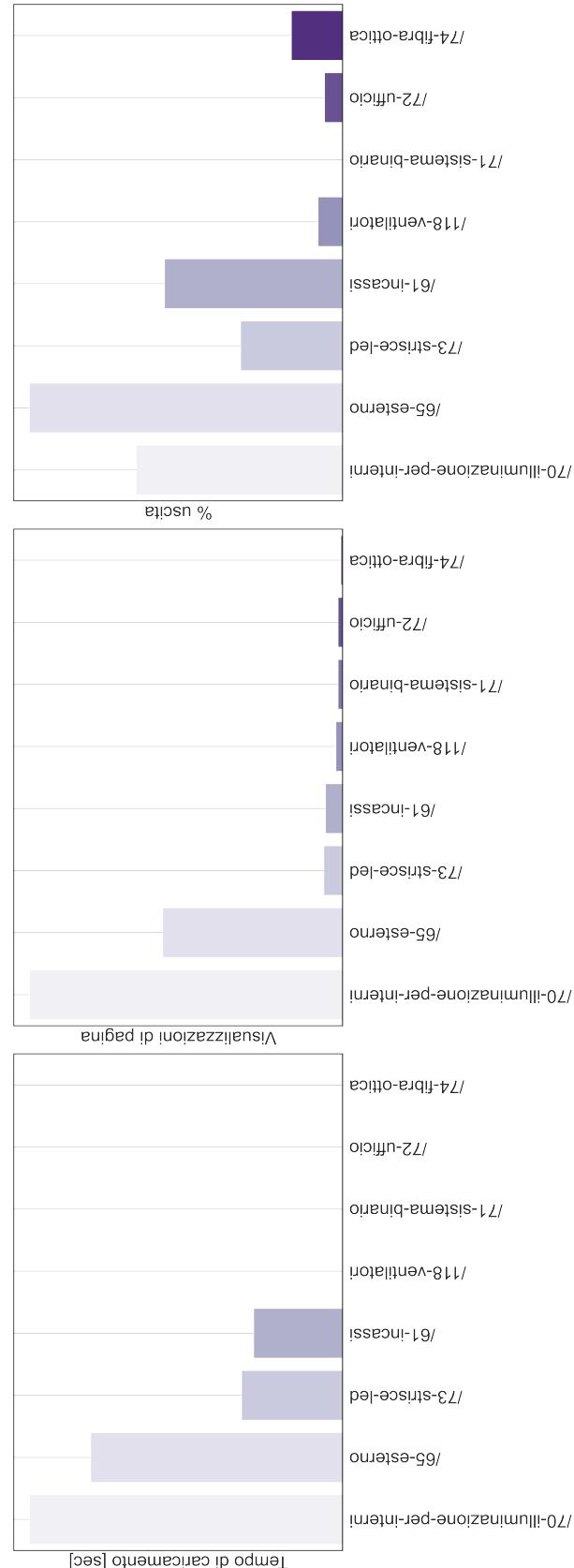


Figura 3.41 – Barplot per la visualizzazione congiunta della velocità di caricamento, il numero di visualizzazioni e la frequenza di uscita delle sezioni del siti.

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

che impiegano più tempo a caricare? Per avere questa informazione è sufficiente un grafico scatterplot che correli i tempi di caricamento con il nome della pagina in questione (figura 3.42: sezione "Illuminazione per interni", figura 3.43: sezione "Esterno"). Vediamo, in entrambi i grafici, che alcuni punti sono significativamente più alti della media¹⁵; per conoscere la natura delle pagine in questione basta selezionare dal dataset i valori con la stessa sintassi che abbiamo visto in figura 3.31.

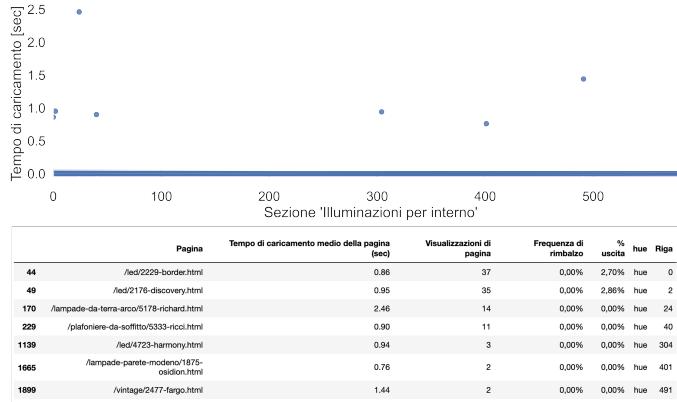


Figura 3.42 – Velocità di caricamento delle pagine - sezione "Illuminazioni per interni" e dataset con i dati outsider.

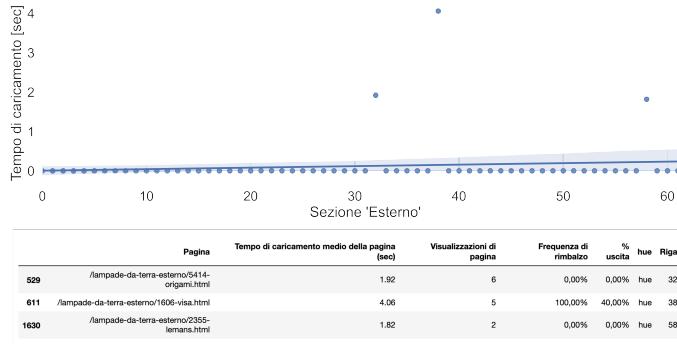


Figura 3.43 – Velocità di caricamento delle pagine - sezione "Esterno" e dataset con i dati outsider.

I grafici di questo paragrafo assicurano una visualizzazione completa sulla metrica velocità, soprattutto in relazione ad altri parametri di interesse quali la frequenza di rimbalzo, il numero di visualizzazioni e la % di uscita dal sito. Esiste un altro tipo di grafico che può raccogliere tutti e quattro i valori, relazionandoli in un unico spazio: lo *spiderplot*. Questo tipo di grafico è sostanzialmente una griglia circolare suddivisa in tante sezioni quante sono i parametri da valutare. Lungo le righe che separano le

¹⁵ecco un altro significativo esempio di dato outsider. In questo caso però non lo ricerchiamo per eliminarlo, ma per approfondirne la natura.

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

sezioni, a partire dal centro, vengono plottate delle righe colorate la cui lunghezza è proporzionale al valore del parametro. Gli estremi di queste linee vengono poi uniti tra di loro a formare un poligono, In figura 3.44 sono visibili gli spiderplot di tutte e otto le sezioni del sito. I cerchi sono suddivisi da due linee perpendicolari, a formare quattro quadranti: in alto il tempo di caricamento, in basso la frequenza di rimbalzo, a sinistra la % di uscita e a destra il numero di visualizzazioni. I poligoni che ne derivano mostrano immediatamente la situazione generale. La sezione "Illuminazioni per interni", ad esempio, nonostante abbia un alto tempo di caricamento ha numerose visualizzazioni e una bassa % di uscita. La sezione "Esterno" invece ha la % di uscita massima anche se il tempo di caricamento non è il più alto in assoluto. Paga però con un numero di visualizzazioni molto minore.

Per poter ottenere questo tipo di grafico è, necessario che tutte le variabili abbiano valori che si trovano nello stesso intervallo. Quando il dataset di partenza non si presenta in questa situazione è necessaria un'operazione preliminare: la sua *normalizzazione*. Importando il pacchetto `sklearn` applichiamo la funzione `MinMaxScaler`, che normalizza i valori ponendoli tutti nell'intervallo tra zero e uno, in modo da renderli paragonabili.

3.7.5 Sorgente del traffico dati: barplot

L'ultima metrica che ho valutato è la *sorgente* del traffico, vale a dire il modo in cui gli utenti hanno conosciuto il sito. GA suddivide le sorgenti in nove categorie (tabella 3.7, adattata dal sito web della guida all'utilizzo di GA¹⁶). Teniamo presente che l'analisi della sorgente di traffico è relativa a tutto il sito, e non a una sezione specifica.

GA genera due tipi di dataset: uno che identifica il *mezzo* (figura 3.45) e un altro che identifica la *sorgente* (figura 3.46).

La differenza tra i due è che il dataset *mezzo* mostra i dati raggruppati per tipologia di traffico, come in tabella 3.7, mentre il grafico *sorgente* esplicita queste tipologie. Cominciamo osservando il dataset *mezzo*: nel nostro caso GA ha identificato cinque delle queste nove sorgenti di traffico: *ppc*, *organic*, *referral*, *(none)* e *Social*. Quando visualizzate in un grafico a barre, impostato in orizzontale, si nota come la maggioranza delle sessioni provenga da click su annunci (*ppc* è l'acronimo di *pay per click*), a segno

¹⁶<https://support.google.com/analytics/answer/1191184?hl=it#zippy=%2Ccontenuti-di-questo-articolo>

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale



Figura 3.44 – Grafico "spider" per ognuna delle otto sezioni del sito.

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

Definizione	Tipologia di traffico identificato
Display	<i>Interazioni con un mezzo "display" o "cpm". Include anche le interazioni Google Ads con la rete di distribuzione degli annunci impostata su "Content".</i>
Paid Search	<i>Traffico dalla rete di ricerca Google Ads o da altri motori di ricerca con mezzo "cpc" o "ppc".</i>
Other Advertising	<i>Sessioni contrassegnate con un mezzo "cpc", "ppc", "cpm", "cpv", "cpa", "cpp", "affiliate" (esclusa ricerca a pagamento).</i>
Organic Search	<i>Incassi, Traffico proveniente dalla ricerca gratuita su qualsiasi motore di ricerca, ad esempio mezzo="organic".</i>
Social Network	<i>Traffico proveniente da uno dei circa 400 social network (non contrassegnati come annunci).</i>
Referral	<i>Traffico proveniente da siti web che non sono social network.</i>
Email	<i>Incassi, Sessioni contrassegnate con un mezzo "email".</i>
Direct	<i>Incassi, Sessioni in cui l'utente ha digitato il nome dell'URL del tuo sito web nel browser o è arrivato al tuo sito tramite un preferito, ad esempio sorgente="(direct)" e mezzo="(not set)" o "(none)".</i>
(unavailable) o (other)	<i>Sessioni che non corrispondono ad alcuna definizione di canale.</i>

Tabella 3.7 – Definizioni delle sorgenti di traffico analizzate da GA.

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

In [2007]:	Mezzo						
Out[2007]:							
	Mezzo	Utenti	Nuovi utenti	Sessoni	Frequenza di rimbalzo	Pagine/sessione	Durata sessione media
0	ppc	1955	1952	2321	58,04%	3,36	00:00:56
1	organic	597	583	793	30,26%	10,19	00:04:10
2	referral	583	581	631	69,41%	3,55	00:01:21
3	(none)	402	402	663	41,63%	6,35	00:04:36
4	Social	3	3	3	33,33%	2,00	00:00:15

Figura 3.45 – Dataset "Mezzo".

In [2009]:	Sorgente.head(15)						
Out[2009]:							
	Sorgente	Utenti	Nuovi utenti	Sessoni	Frequenza di rimbalzo	Pagine/sessione	Durata sessione media
0	google	2505	2500	3075	50,60%	5,09	00:01:44
1	(direct)	402	402	663	41,63%	6,35	00:04:36
2	instagram.com	290	290	296	75,34%	2,38	00:00:23
3	l.instagram.com	246	239	249	70,68%	3,25	00:00:47
4	facebook.com	26	26	26	96,15%	1,23	00:00:02
5	baidu	21	20	21	100,00%	1,00	00:00:00
6	bing	10	10	13	38,46%	16,23	00:10:19
7	m.facebook.com	10	9	10	70,00%	3,50	00:01:08
8	paginegialle.it	4	4	4	25,00%	7,00	00:02:19
9	sogou	4	4	4	100,00%	1,00	00:00:00
10	IGShopping	3	3	3	33,33%	2,00	00:00:15
11	it.search.yahoo.com	3	3	7	0,00%	29,00	00:25:11
12	welabo.it	3	1	22	4,55%	12,59	00:11:36
13	l.facebook.com	2	1	8	0,00%	11,62	00:10:06
14	pinterest.com	2	2	3	33,33%	1,67	00:00:12

Figura 3.46 – Dataset "Sorgente".

che la campagna pubblicitaria messa in piedi dai curatori del sito web occupa un posto di rilievo nel generare il traffico dati del sito. Il grafico, visibile in figura 3.47 (insight), presenta però due criticità: c'è una buona fetta di sorgente non chiaramente identificato ("none"), e sebbene sia presente il campo "Social" la barra corrispondente non è visibile, quasi a voler indicare un valore nullo. Per comprendere meglio ci rifacciamo al secondo dataset, quello della *sorgente*, che rappresentiamo sempre con un grafico a barre, ma questa volta in orientamento classico, ossia verticale (figura 3.47.) Possiamo notare come la maggior parte degli utenti provengano da Instagram, un decimo di questi da Facebook e solo una minoranza da altre fonti. Importante è osservare il numero di accessi, guardando con attenzione la scala dell'asse y. Quasi 600 accessi provengono da Instagram, circa 50 da Facebook. Se ci fossimo fermati ad osservare il primo dei due grafici avremmo potuto pensare che meno di una decina delle nostre visite provenisse da canali social. Indagando invece anche il dataset relativo alla "sorgente" ci accorgiamo che la quasi totalità del traffico identificato come "none" è in realtà proprio traffico social.

3.7. Analisi delle visualizzazioni: istogramma e scatterplot dimensionale

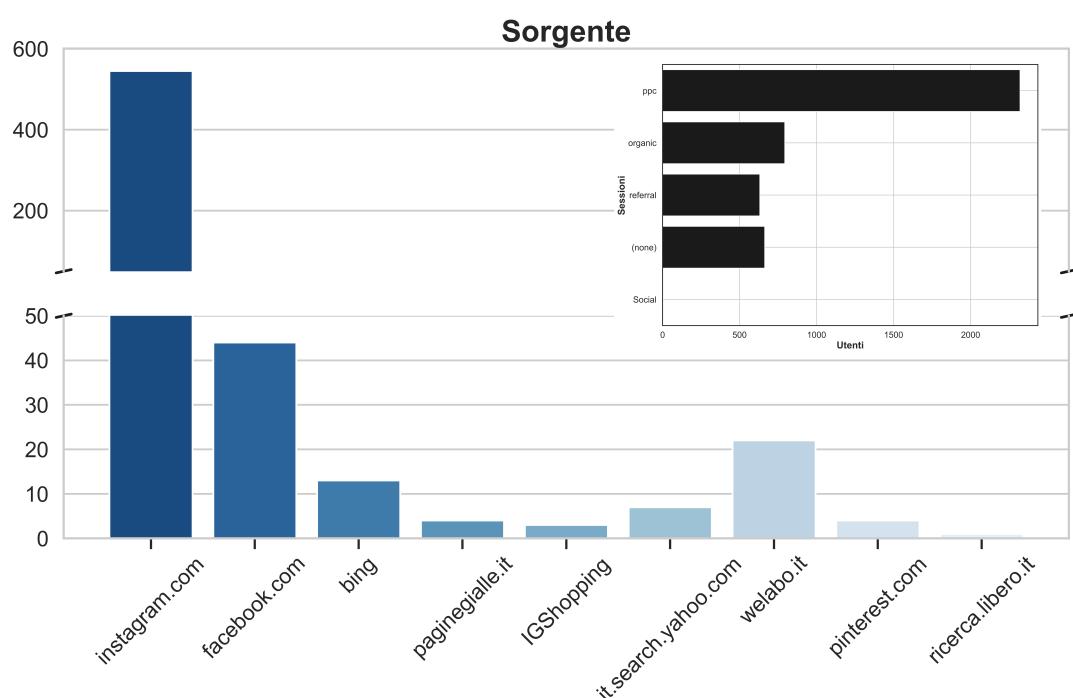


Figura 3.47 – Grafici a barre per l’analisi delle sorgenti del traffico web.

Capitolo 4

Conclusioni

In questo lavoro di tesi ho mostrato un flusso di lavoro per acquisire, selezionare e visualizzare dati di traffico web diretti verso un sito di e-commerce italiano per un periodo di 40 giorni circa.

Il sito è stato concepito e costruito per aumentare la visibilità dell'azienda e raggiungere quanta più clientela possibile. I dati raccolti e analizzati sono stati scelti di conseguenza: *visualizzazione delle pagine, provenienza geografica delle visualizzazioni, canali che hanno indirizzato il traffico verso il sito*. In aggiunta, particolare enfasi è stata data anche all'analisi della metrica *velocità*, intesa come tempo di caricamento delle pagine, in relazione ad altri parametri.

L'analisi dei dati generali non ha fornito informazioni sufficientemente chiare, quindi, per comprendere meglio come gli utenti si siano comportati nelle loro sessioni di navigazione, ho suddiviso i dati in *sezioni*, secondo la struttura con cui è costruito il sito web. Ho scoperto che due sezioni sono significativamente più visualizzate delle altre, e mi sono concentrato sull'analisi del traffico diretto verso queste due sezioni.

Ho costruito diversi tipi di grafici, che mostravano singole variabili o che mettevano in relazione due o tre metriche. In questo modo ho identificato più facilmente le pagine più visitate, quelle con frequenza di rimbalzo maggiore e l'andamento in generale delle sottosezioni. DI Particolare interesse è la visualizzazione delle comparazioni tra velocità di caricamento delle pagine del sito e le altre metriche.

L'analisi attenta delle sorgenti del traffico, infine, ha rivelato che buona parte dei reindirizzamenti al sito proviene da due social network.

Tutte le informazioni raccolte sono state discusse e comunicate con i proprietari del sito web, che valuteranno le azioni da intraprendere per migliorare la struttura del sito

e provare a renderlo sempre più facile da navigare, con lo scopo finale di ottenere una maggiore clientela e una visibilità più estesa.

Bibliografia

- [AKMZ01] Suhail Ansari, Ron Kohavi, Llew Mason, and Zijian Zheng. Integrating e-commerce and data mining: Architecture and challenges. In *Proceedings 2001 IEEE International Conference on Data Mining*, pages 27–34. IEEE, 2001.
- [BC07] Kenneth N Berk and Patrick Carey. *Data Analysis with Microsoft Excel*. Brooks/Cole, Cengage Learning, 2007.
- [Ber20] Barry Berman. Paths to purchase: the seven steps of customer purchase journey mapping. *Rutgers Business Review*, 5(1):84–100, 2020.
- [FM04] Marisa Ferrara and Steven Moran. Review of dbms for linguistic purposes. In *E-MELD Workshop 2004.*, 2004.
- [FR17] Maya F Farah and Zahy B Ramadan. Disruptions versus more disruptions: How the amazon dash button is altering consumer buying patterns. *Journal of Retailing and Consumer Services*, 39:54–61, 2017.
- [jup] <https://jupyter.org/>.
- [Kam20] David Kamerer. Reconsidering bounce rate in web analytics. *Journal of Digital & Social Media Marketing*, 8(1):58–67, 2020.
- [LMMB12] Ronald JW Lambert, Ioannis Mytilinaios, Luke Maitland, and Angus M Brown. Monte carlo simulation of parameter confidence intervals for non-linear regression analysis of biological data using microsoft excel. *Computer methods and programs in biomedicine*, 107(2):155–163, 2012.
- [LTT11] Jerri L Ledford, Joe Teixeira, and Mary E Tyler. *Google analytics*. John Wiley and Sons, 2011.

- [May20] Timothy R. Mayes. *Financial analysis with microsoft excel*. Cengage Learning, 2020.
- [neg19] https://blog.osservatori.net/it_it/negoziodelfuturo-innovazioni, 2019.
- [OMS11] Mohammad Amin Omidvar, Vahid Reza Mirabi, and Najes Shokry. Analyzing the impact of visitors on page views with google analytics. *arXiv preprint arXiv:1102.0735*, 2011.
- [pana] https://pandas.pydata.org/docs/getting_started/overview.html.
- [panb] https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.drop_duplicates.html.
- [PKM19] Panagiotis Papadopoulos, Nicolas Kourtellis, and Evangelos Markatos. Cookie synchronization: Everything you always wanted to know but were afraid to ask. In *The World Wide Web Conference*, pages 1432–1442, 2019.
- [Pla11] Beatriz Plaza. Google analytics for measuring website performance. *Tourism Management*, 32(3):477–481, 2011.
- [pyc] <https://www.jetbrains.com/pycharm/>.
- [pyta] <https://docs.python.org/3/license.html>.
- [pytb] <https://www.python.org/about/gettingstarted/>.
- [pytc] <https://www.python.org/psf-landing/>.
- [pytd] <https://www.python.org/dev/peps/pep-0020/#abstract>.
- [pyte] <https://www.python.org/>.
- [pyt21] [https://en.wikipedia.org/wiki/Python_\(programming_language\)#Design_philosophy_and_features](https://en.wikipedia.org/wiki/Python_(programming_language)#Design_philosophy_and_features), 2021.
- [RFK19] Zahy B Ramadan, Maya F Farah, and Danielle Kassab. Amazon’s approach to consumers’ usage of the dash button and its effect on purchase decision involvement in the us market. *Journal of Retailing and Consumer Services*, 47:133–139, 2019.

- [RRHV21] Alexander Rossolov, Halyna Rossolova, and José Holguín-Veras. Online and in-store purchase behavior: shopping channel choice in a developing economy. *Transportation*, pages 1–37, 2021.
- [sea] <https://seaborn.pydata.org/>.
- [sol] <https://www.ilsole24ore.com/art/la-realta-diventera-virtuale-ma-acquisti-saranno-reali-ADFwMmM>.
- [SS14] Barna Saha and Divesh Srivastava. Data quality: The other face of big data. In *2014 IEEE 30th international conference on data engineering*, pages 1294–1297. IEEE, 2014.
- [UM02] Godwin J Udo and Gerald P Marquis. Factors affecting e-commerce web site effectiveness. *Journal of Computer Information Systems*, 42(2):10–16, 2002.
- [use] https://w3techs.com/technologies/overview/traffic_analysis.
- [VDK⁺13] Katrien Verbert, Erik Duval, Joris Klerkx, Sten Govaerts, and José Luis Santos. Learning analytics dashboard applications. *American Behavioral Scientist*, 57(10):1500–1509, 2013.
- [VWK16] Peter Verhoef, Edwin Kooge, and Natasha Walk. *Creating value with big data analytics: Making smarter marketing decisions*. Routledge, 2016.
- [WAS05] Xiaozhe Wang, Ajith Abraham, and Kate A Smith. Intelligent web traffic mining and analysis. *Journal of Network and Computer Applications*, 28(2):147–165, 2005.
- [ZR13] Adriano Z Zambom and Dias Ronaldo. A review of kernel density estimation with applications to econometrics. *International Econometric Review*, 5(1):20–42, 2013.