



Università degli Studi di Milano Bicocca

Scuola di Scienze

Dipartimento di Informatica, Sistemistica e Comunicazione

Corso di laurea magistrale in Data Science

Simulazione e analisi di un sistema di bike sharing

Settembre 2022

Merlo Riccardo - 829805 - r.merlo2@campus.unimib.it

Riva Leonardo - 830647 - l.riva37@campus.unimib.it

Stoffa Giacomo - 830159 - g.stoffa1@campus.unimib.it

Abstract

Il progetto è costituito da una prima fase di simulazione del sistema di bike sharing nella città di San Jose (California, USA), utilizzando i dati Lyft degli anni 2018-2019. La simulazione è stata costruita da un punto di vista statistico, andando a contestualizzare i percorsi, in entrata e in uscita da ogni stazione, in base al periodo dell'anno / settimana / giornata. Una volta costruito il modello, è stata affrontata la tematica dello shortage, ovvero lo studio della mancanza di biciclette e di parcheggi nelle stazioni, andando a dimostrare che, aumentando sia il numero di parcheggi che il numero di bici disponibili, molti di questi problemi possono essere mitigati. Infine, sono stati analizzati due diversi scenari what-if: il primo riguarda la variazione di biciclette totali nel sistema, mentre il secondo riguarda la rimozione di una stazione.

Indice

1. Introduzione	4
2. Stato dell'arte	6
3. Download e preprocessing dei dati	9
3.1. Struttura dati	9
3.2. Pulizia dati	9
4. Esplorazione	10
5. Simulazione	13
5.1. Esecuzione	14
5.2. Validità simulazione	16
6. Analisi degli shortage	18
6.1. Soluzione 1: incrementare il numero di parcheggi	18
6.2. Soluzione 2: incrementare il numero di biciclette totali	20
6.3. Soluzione 3: unione delle soluzioni precedenti	21
6.4. Shortage nelle stazioni	23
7. Scenari what-if	29
7.1. Se ci fossero più / meno biciclette?	29
7.2. Se ci fossero più / meno stazioni?	32
8. Conclusioni e Sviluppi Futuri	37
9. Bibliografia	39

1. Introduzione

Dagli e-scooter al bike sharing, le soluzioni di **micromobilità** sono il futuro dei trasporti nelle città di tutto il mondo. Aiutano a ridurre le emissioni dei veicoli, migliorare la salute, ridurre la congestione e il consumo di carburante e fare risparmiare denaro. I numeri della mobilità condivisa, inoltre, sono in continua crescita [1].

Il **bike sharing** si basa su un sistema di stazioni bici self-service. Gli utenti in genere controllano una bicicletta utilizzando un abbonamento o una carta di credito o debito. Possono quindi raggiungere la loro destinazione e parcheggiare la bicicletta in una vicina docking station.

In pochi anni, sono nate molte aziende concorrenti nello sviluppo di sistemi di Bike Sharing. Non tutte finiscono per funzionare a dovere, scontrandosi con diversi **problemi**; uno tra questi è il sovraffollamento di biciclette rispetto ai potenziali utenti, che comportano l'inutilizzo del servizio. Con il fallimento di start-up di questo tipo, le bici devono poi essere smaltite con un costo che dovrebbe essere sostenuto dall'azienda (ma che poi non succede mai).

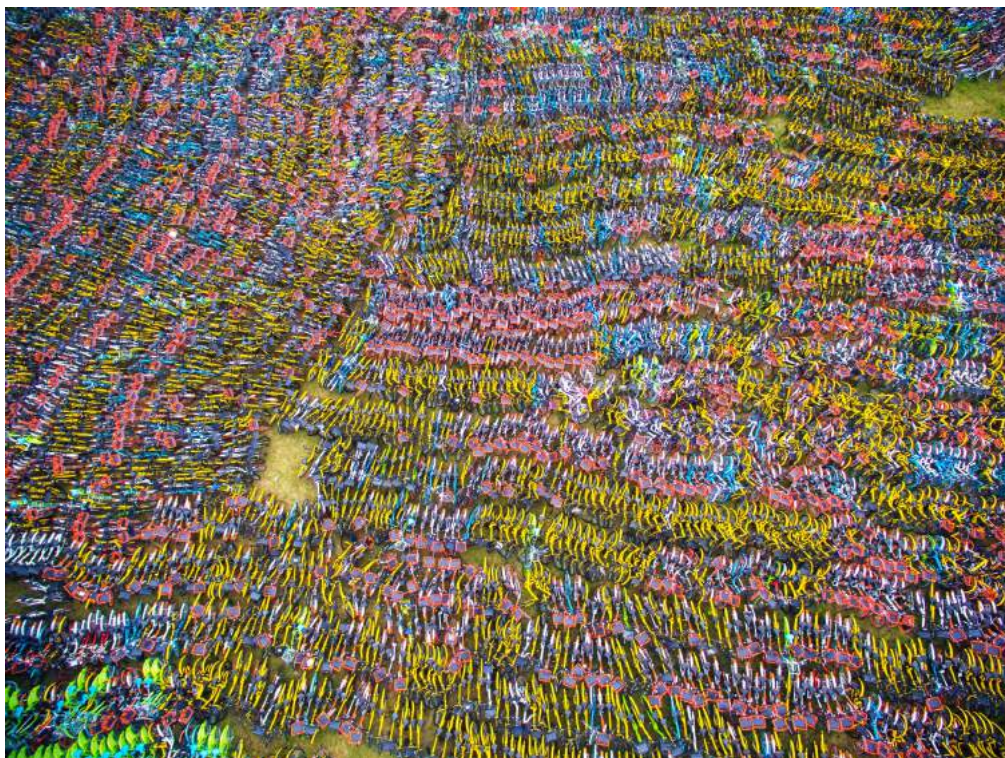


Figura 1.1. Cimitero di biciclette ad Hangzhou, Cina [2].

Ciò che può portare un'azienda a imporsi su altre può essere uno studio simulativo sul funzionamento generale del sistema, in modo da massimizzare i profitti e ridurre i problemi che gli utenti devono affrontare.

In quest'ottica di spostamenti tra biciclette indipendenti che effettuano movimenti interconnessi tra differenti stazioni risulta immediato il collegamento a **sistema complesso**, ovvero un sistema dinamico multi-agente, composto da diversi soggetti che interagiscono tra loro, che possono subire continue modifiche localmente o singolarmente prevedibili. In questo tipo di sistema tipicamente non è facile prevedere il suo comportamento futuro nell'insieme e, per essere studiato, necessita di essere simulato. In questo caso, tramite la simulazione è possibile innanzitutto replicare cosa succede nel mondo reale; inoltre, è possibile andare a studiare la fattibilità / comportamento di possibili scenari ipotetici, come delle modifiche alle biciclette o alle stazioni.

In questo progetto, verranno analizzati i dati esistenti del sistema di Bike Sharing di una data città. In particolare, è stato preso come caso di studio la città californiana di **San Jose**, che rappresenta un chiaro esempio di Bike Sharing funzionante ed utilizzato quotidianamente. Infatti, l'azienda produttrice Lyft [3], ha lavorato con il dipartimento dei trasporti della città, in modo tale da garantire che il sistema rimanesse accessibile a tutti e incoraggasse un buon comportamento degli utenti nella fase di parcheggio, portando ad una migliore disponibilità delle biciclette quando e dove i cittadini ne avessero bisogno.

Il codice Python è interamente disponibile su GitHub [4]

2. Stato dell'arte

Di seguito vengono riportati alcuni lavori in letteratura in questo settore, descrivendo poi come si costruirà il nostro lavoro sulla base di essi.

Saltzman [5] propone una simulazione basata su eventi discreti, utilizzando un dataset di un servizio di bike sharing di San Francisco. Partendo dai dati, vengono stimate le dimensioni iniziali delle stazioni, in termini di numero di biciclette presenti. Nella simulazione, le stazioni di destinazione dei viaggi vengono scelte osservando le distribuzioni di probabilità calcolate sui dati storici. Viene analizzato il fenomeno della mancanza di biciclette nelle stazioni, dovute al fatto che i viaggi sono per la maggior parte unidirezionali. Tramite delle simulazioni, ha definito che, aumentando solamente del 3-4% il numero di postazioni nelle stazioni e di biciclette totali nel sistema, si riesce a ridurre gli shortage del 30%, andando a tamponare i costi del rebalancing manuale da parte dell'azienda.

Soriguera [6] ha costruito una simulazione agent-based al fine di valutare il processo di decision-making nella fase iniziale di implementazione e pianificazione. Nella simulazione, le stazioni di destinazione vengono calcolate campionando una distribuzione di probabilità tra tutte le stazioni. Inoltre, viene fissata una soglia di distanza massima che l'utente è disposto a percorrere nel caso in cui la sua stazione di riferimento fosse limitata. Vengono sollevate le cause che portano il sistema al collasso, ovvero la presenza di stazioni sovraccariche o deserte, spostando il focus sulle motivazioni che rendono il ribilanciamento una necessità piuttosto che sulle soluzioni ad esso.

Hebenstreit [7] modella una simulazione di Bike Sharing focalizzata su l'instradamento delle biciclette, pesato da diversi parametri, tra stazioni. Il sistema viene descritto attraverso un grafo, dove i nodi rappresentano le stazioni e gli archi il costo per viaggiare tra stazioni. Viene utilizzato l'algoritmo di routing di dijkstra per calcolare il costo totale dei percorsi, al fine di individuare il percorso migliore tra stazione A e B.

Lammel [8], nella sua simulazione, si concentra invece dal punto di vista del flusso di traffico e delle strategie di routing dei percorsi. Importante è il concetto della modifica del percorso quando non risulta più vantaggioso (anche se inizialmente un percorso risulta il migliore, consigliandolo a tutti si rallenta).

Fernandez [9] ha sviluppato una simulazione, che permette di analizzare diverse strategie di rebalancing, costruita con un design modulare che ne permette l'ampliamento e che permette di ricavare i risultati facilmente per successive visualizzazioni o analisi. Tra le caratteristiche della simulazione, spicca la logica del booking. Questa permette agli utenti di prenotare una bicicletta, in modo da poter aiutare il sistema in fase di rebalancing, avendo quindi a disposizione delle stime di utilizzo in anticipo.

Mukku [10] presenta una simulazione modulare basata su agenti, che dimostra il comportamento di diversi elementi quali: utenti, bici, stazioni, aree operative, instradamento,

ribilanciamento, etc. Il focus è spostato verso l'implementazione del comportamento dell'utente, con instradamento personalizzato e rebalance autonomo tramite bici con guida automatica. La strategia proposta permette all'utente di prenotare una bicicletta e riceverla nel luogo richiesto entro un tempo limite. Al termine del viaggio la bici torna automaticamente in stazione o si avvicina al prossimo utente.

Salah [11] Paper concettuale che tratta il lavoro di Mukku, nel quale viene presentato il modello logico della simulazione con un focus sugli utenti che ricoprono il ruolo di agenti. Inoltre, si cerca di rispondere alla domanda di ricerca: La riduzione degli sforzi per il ribilanciamento e del numero di biciclette necessarie per utente compensano il maggior investimento per bicicletta? (*Do reduced efforts for rebalancing and fewer bikes per user needed compensate for the higher investment per bike?*).

Jian [12] propone un metodo per ottimizzare la posizione iniziale delle biciclette nelle varie stazioni all'inizio della giornata. L'approccio si basa su una simulazione ad eventi discreti (una generazione è un minuto). Gli utenti vengono generati randomicamente nelle stazioni, i viaggi sono costruiti seguendo una distribuzione di probabilità così come la loro durata. Se un utente non trova un mezzo nella stazione allora abbandona il sistema. Invece, se un utente arriva a destinazione ma non trova parcheggio, ritenta al massimo tre volte nelle stazioni vicine.

Nel complesso, la maggior parte dei simulatori di Bike Sharing sono stati realizzati più per cercare di risolvere lo specifico problema del rebalancing, piuttosto che per costruire un sistema generale, utilizzabile per analizzare vari scenari, come quello proposto in questo progetto. Inoltre, quasi nessun tool presentato utilizza un'interfaccia grafica, la quale è stata ritenuta non particolarmente rilevante anche per questo lavoro. È importante però disporre di un modo per modellare e conservare i dati in output, necessari ad ogni successiva analisi.

Questa simulazione, come tutte, sfrutta la distribuzione probabilistica di Poisson per generare i dati, ovvero per le biciclette in uscita dalle stazioni. Inoltre, diversamente dalla letteratura, sono stati omessi gli utenti come agenti (personificati dalle biciclette) in quanto non interessanti ai fini della ricerca e al funzionamento della simulazione stessa.

Un'altra decisione che è stata presa riguarda il comportamento delle biciclette all'arrivo e alla partenza, qualora la stazione fosse senza parcheggi o senza biciclette. Nel primo caso, procediamo con il trovare una stazione alternativa dove parcheggiare, poiché riteniamo ovvio il fatto che, una volta partita, la bicicletta debba obbligatoriamente fermarsi. Al contrario, nel secondo caso riguardo la partenza, qualora la stazione non fosse disposta di biciclette abbiamo ipotizzato il fatto che l'utente decida di abbandonare l'idea di noleggiarla e quindi viri verso un'altra tipologia di mezzo di trasporto. In generale, viene ripreso il concetto di modifica del percorso di Lammel, dovuto ad un fattore esterno da parte di altri agenti.

In generale, il lavoro che proponiamo è una simulazione modulare, personalizzabile tramite diversi iperparametri (numero di biciclette, dimensione stazioni, numero parcheggi, ...) al fine di permettere l'analisi di svariati scenari e le rispettive conseguenze, senza entrare nei dettagli del rebalancing (come nella letteratura a cui abbiamo fatto riferimento), ma portando avanti dei confronti più olistici riguardo le relazioni tra i diversi agenti.

3. Download e preprocessing dei dati

3.1. Struttura dati

I dati [13] consistono in una serie di csv zippati, relativi a 3 diverse città della Bay Area in California (San Jose, San Francisco, Oakland), negli anni 2018-2019 (di base dal 2018 ad oggi, ma abbiamo preferito fermarci al 2019 compreso per evitare di costruire una simulazione basata su un andamento di dati anomalo causato dalla pandemia).

Ogni viaggio è anonimo e include:

- Durata del viaggio (in secondi)
- Data e ora di partenza
- Data e ora di arrivo
- ID, nome e coordinate della stazione di partenza
- ID, nome e coordinate della stazione di arrivo
- ID della bicicletta
- Tipo di utente: “subscriber” (cioè un membro del servizio) o “customer” (cioè un utente casuale)

3.2. Pulizia dati

Una volta scaricati, i dati sono stati uniti in un unico csv. Successivamente è stata necessaria una pulizia, che consiste in:

- **fix delle coordinate:** fissata una stazione, le coordinate non sempre coincidono, ma sparse nei dintorni di qualche metro; ne è stata fatta una media di latitudine e longitudine. Inoltre, sono state eliminate le osservazioni con errori (cioè con coordinate non plausibili, come [0,0]).
- **aggiunta della feature “city”,** determinata a partire dalle coordinate, calcolando le distanze dai centroidi delle tre città.
- **conversione della feature “durata viaggio” da secondi a minuti,** per rendere più comprensibili i risultati.
- **rimozione dei viaggi con durata inferiore al minuto e superiore alle 12 ore,** con l’assunzione che siano errori o comunque non siano significativi ai fini dello studio)
- **rimozione dei viaggi tra città diverse** (considerati improbabili)
- **rimozione osservazioni con stazioni non corrette** (cioè non presenti nelle tre città o con valore NaN)

4. Esplorazione

	San Francisco	San Jose	Oakland	Totale
Stazioni	224	85	130	439
Viaggi totali	3 154 671	198 333	902 779	4 255 783

Tabella 4.1. Analisi quantitativa dei viaggi presenti nei dati.

Vista la grossa mole di dati, sono stati presi in considerazione solo i dati relativi alla città di **San José**.

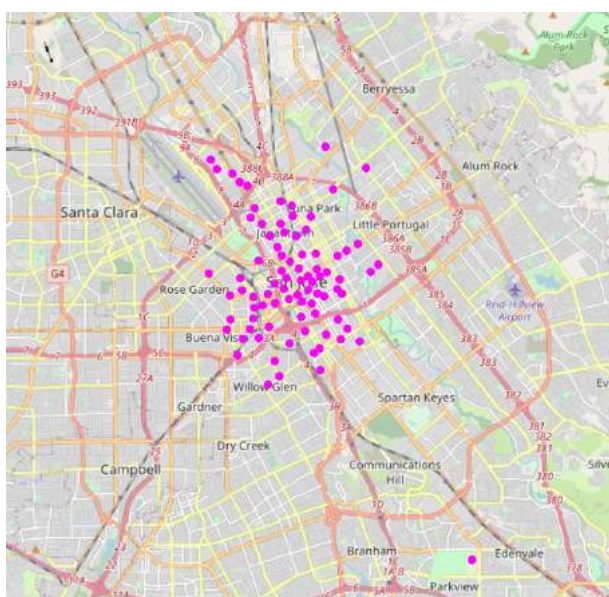


Figura 4.1. Distribuzione delle stazioni a San Jose.

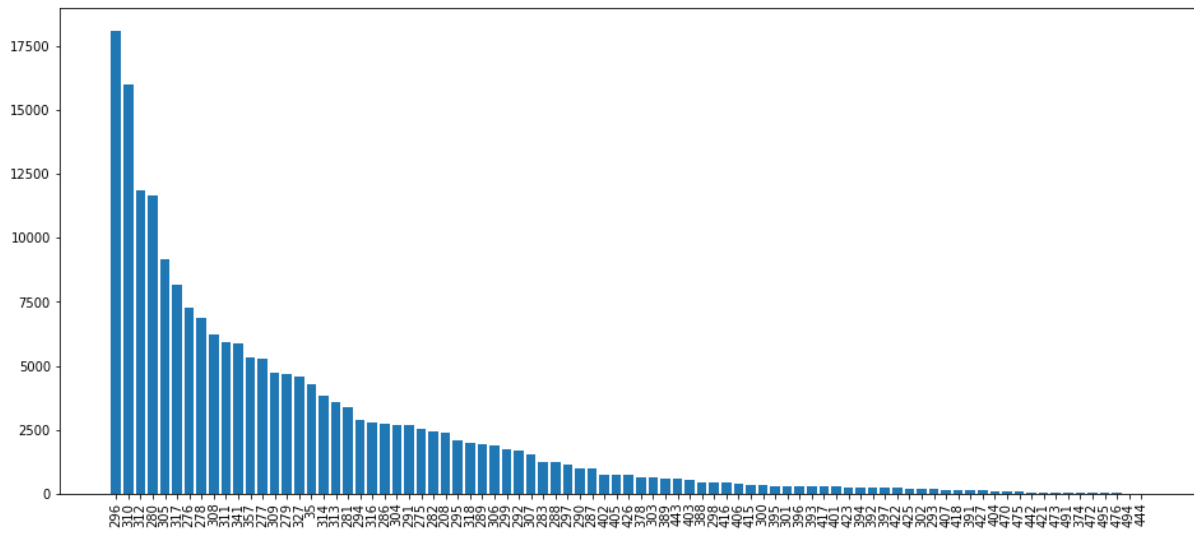


Figura 4.2. Utilizzo delle stazioni nei due anni.

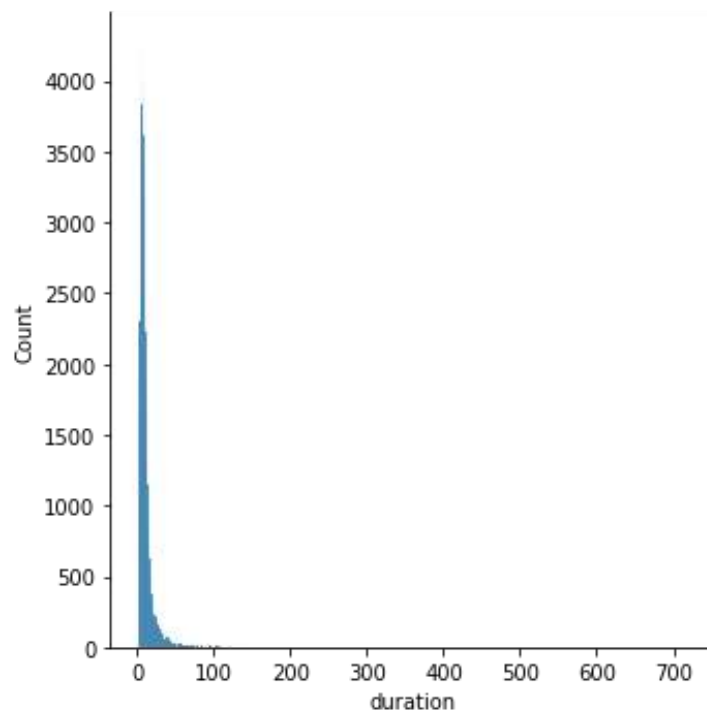
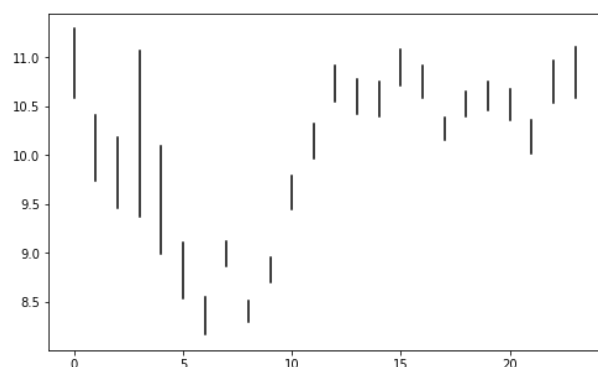
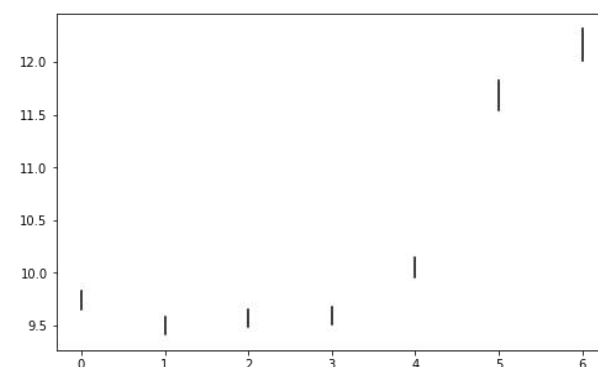


Figura 4.3. distribuzione della durata dei viaggi, in minuti.

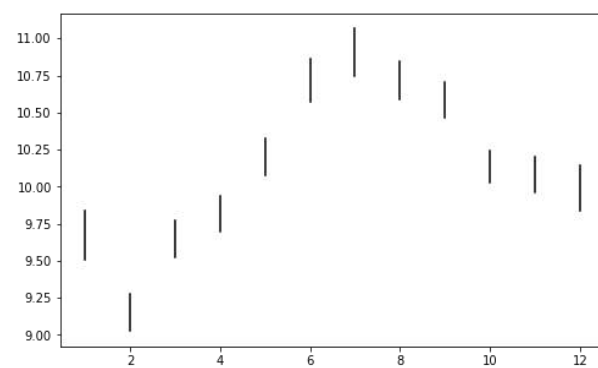
Ore



Giorni della settimana



Mesi



Anni

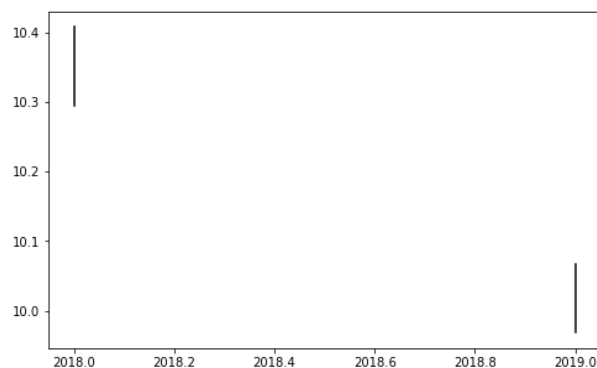


Figura 4.4. Variazione della durata media dei viaggi (intervalli di confidenza al 95%), con un campionamento orario, giornaliero, mensile e annuale.

5. Simulazione

La simulazione è interamente di natura statistica*, rispetto ai viaggi presenti nei dati.

Questo significa che in ogni momento della simulazione si osserverà cosa è successo nei dati reali in quella specifica ora / giorno della settimana / mese, in modo da replicare, da un punto di vista probabilistico, le modalità dei viaggi. Usare una suddivisione di questo tipo può essere in realtà troppo specifico; per questo, si è optato per un raggruppamento, osservando i risultati in fig. 4.2, come segue:

- ora: raggruppando a intervalli di 3 ore
- giorno della settimana: differenziando tra weekday e weekend
- mese: considerando la stagione invece del mese
- anno: è stato ignorato, essendoci solo due campioni

La simulazione si divide in due fasi: preparazione ed esecuzione.

Nella fase di **preparazione** avviene la generazione di:

- biciclette. Vengono contrassegnate da identificativi costruiti in base alla stazione di partenza (dove erano parcheggiate quando la simulazione è stata avviata) e un numero incrementale. Esempio: la bicicletta con id `312_4` è la quarta bici che è stata generata nella stazione 312.
- stazioni. Identificate da “id” reali (in modo da poter essere confrontate con i dati di Lyft), hanno il compito di salvare o rilasciare biciclette. Per semplicità, non hanno limiti di storage (possono ipoteticamente venire parcheggiati un numero infinito di mezzi) ma viene comunque tenuta traccia di eventuali sforamenti.

È necessario che venga definito il numero totale di biciclette nel sistema, le quali vengono successivamente distribuite nelle varie stazioni in modo proporzionale all’ipotetica dimensione della loro “dock”.

* **Disclaimer:** *l'utilizzo di un seed al fine di rendere il codice riproducibile si scontra con le assunzioni stocastiche intrinseche del sistema che sfrutta la conoscenza statistica inferita dai dati per costruire delle distribuzioni di probabilità dalle quali poi estrarre informazione. Fissare un seed modificherebbe quindi le sorti imprevedibili necessarie per selezionare i dati con un criterio probabilistico senza seguire la distribuzione. Evitando la ripetizione di situazioni uguali nel tempo.*

Al fine di ottenere questi valori, è stata effettuata una stima equivalente al numero massimo di biciclette parcheggiate in contemporanea, nell'arco dei due anni, per ogni stazione.

Nella fase di **esecuzione**, invece, vengono simulati i viaggi tra le varie stazioni, con una frequenza di aggiornamento fissata a 5 minuti, per un trade-off tra realismo ed efficienza di computazione. Devono essere definite le date di inizio e fine simulazione.

Verrà di seguito analizzata più in dettaglio la fase di esecuzione.

5.1. Esecuzione

L'esecuzione si compone di due ulteriori sotto-fasi.

La fase di **update**, che consiste nell'aggiornamento delle bici in circolazione.

1. Per ciascuna bici in circolazione, viene calcolata la probabilità che continuino la corsa oppure si fermino (dato che hanno passato M minuti in circolazione, partendo dalla stazione S, in un certo periodo dell'anno)

es. la bici 312_4 si ferma con prob. 0.8 e continua corsa con prob. 0.2

2. Sulla base di queste probabilità, viene determinato se fermano la corsa, randomicamente (estraendo un numero con distribuzione costante e confrontandolo con la probabilità dell'evento).
3. Se la bici ha deciso di fermarsi, bisogna stabilire la stazione di destinazione. Viene calcolata, per ciascuna stazione, la probabilità di terminare in essa la corsa (sempre sulla base delle statistiche).

es. la bici 312_4 si ferma:

- nella stazione 304 con prob. 0.4
- nella stazione 287 con prob. 0.2
- nella stazione 281 con prob. 0.2
- ...
- nella stazione 401 con prob 0

4. Sulla base di queste probabilità, viene selezionata la stazione randomicamente.

La fase di **add**, che consiste nella partenza di altre bici dalle stazioni.

1. Per ciascuna stazione, viene costruita una distribuzione di Poisson (nota anche come legge degli eventi rari). Il parametro λ è il numero medio di partenze in quel periodo dell'anno.

- Successivamente viene estratto un numero casuale dalla distribuzione, che corrisponderà alle bici uscenti. Se non sono presenti abbastanza biciclette nella stazione, verranno usate tutte quelle disponibili.

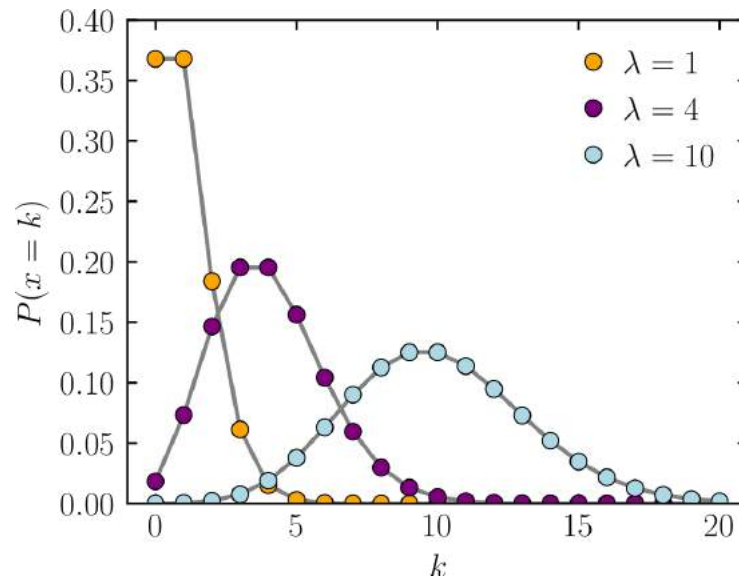


Figura 5.1. Esempi di distribuzione di Poisson [14].

Di seguito viene mostrato uno pseudocodice riepilogativo (fig. 5.2) sul funzionamento generale della simulazione; successivamente un diagramma di flusso prendendo in considerazione le biciclette come degli agenti in un sistema complesso.

```

procedure simulation
  while current_datetime < end_datetime:
    do update
    do add
    increase current_datetime by 5 minutes

procedure update
  for each running_bike:
    determine if it stops
    if stops:
      determine at which station

procedure add
  for each station:
    generate n outgoing_bikes
  
```

Figura 5.2. Pseudocodice relativo all'intera simulazione.

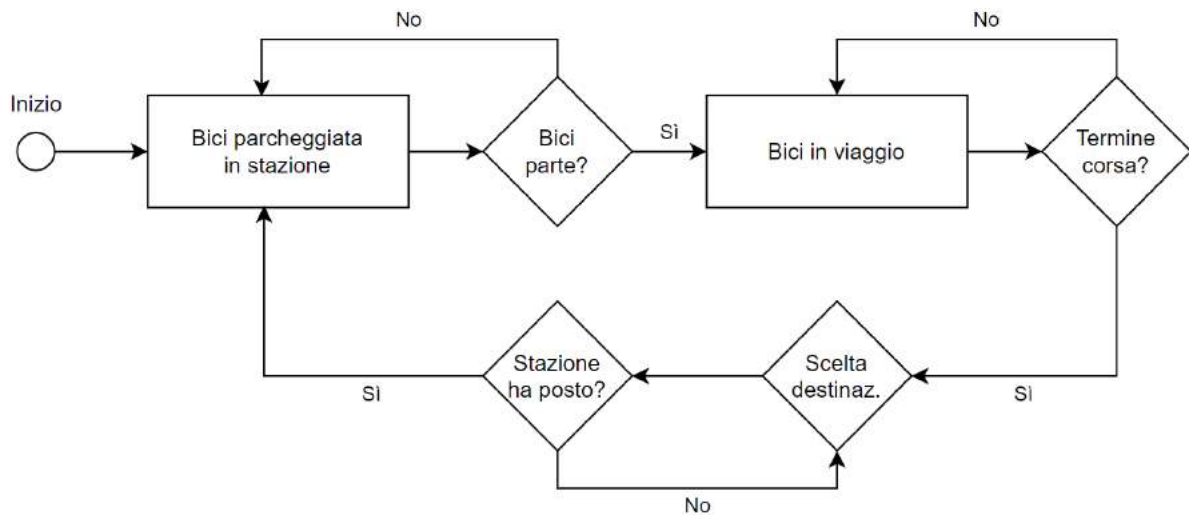


Figura 5.3. Flowchart dal punto di vista dell'agente nel sistema complesso.

5.2. Validità simulazione

Per determinare se effettivamente la simulazione creata sia realistica e affidabile, se ne possono osservare le differenze con i dati reali.

Dal punto di vista del flusso di biciclette che passano per le stazioni nell'intera durata, i risultati della simulazione sembrano replicare quelli reali (fig 5.4 e 5.5).

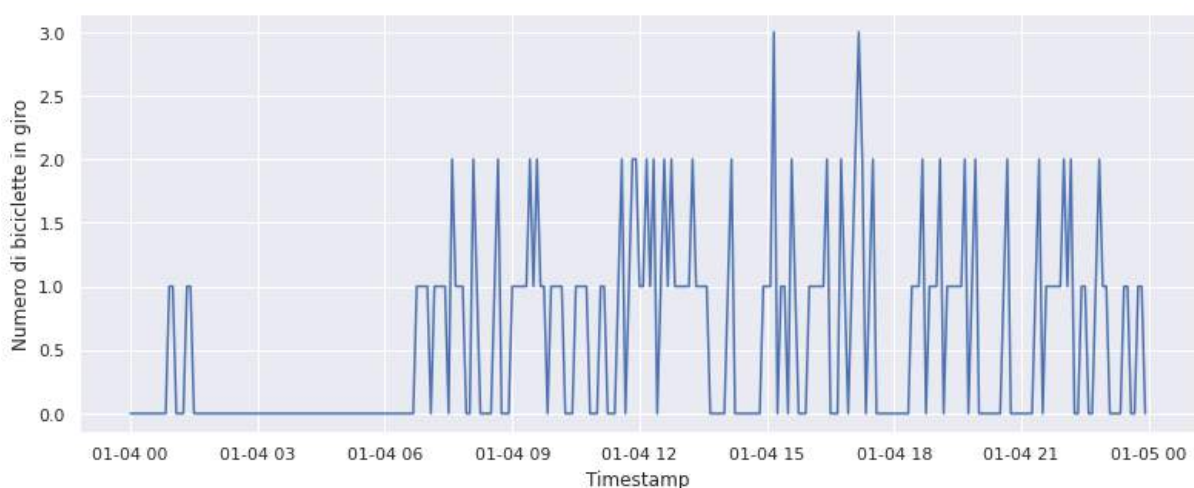


Figura 5.4. Flusso di biciclette il giorno 4 gennaio, nella simulazione.

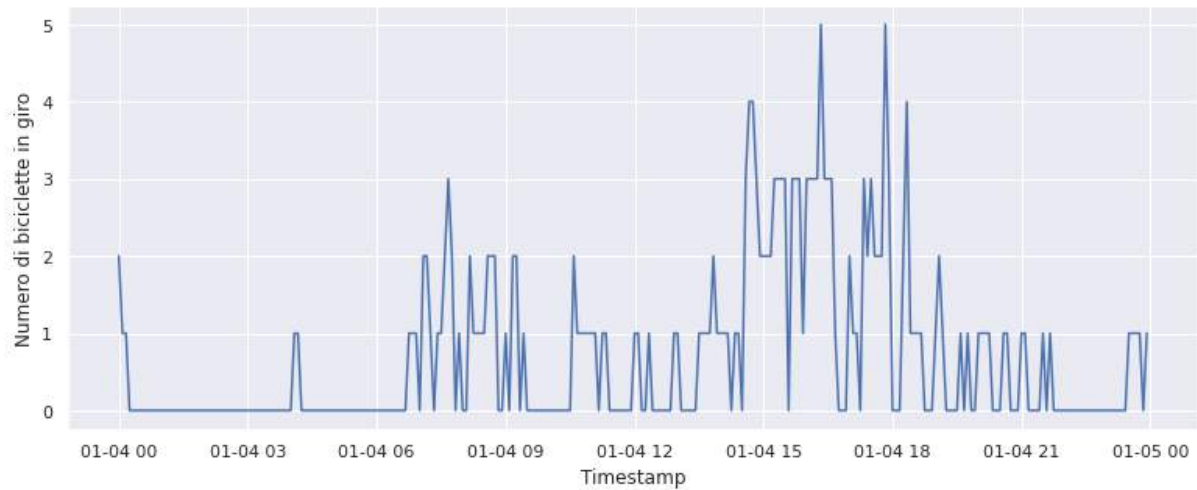


Figura 5.5. Flusso di biciclette il giorno 4 gennaio, nei dati reali.

Al fine di dare un giudizio riassuntivo della bontà della simulazione, rispetto ai dati reali, è stato calcolato il MAE (Errore Medio Assoluto) tra il flusso di biciclette simulato e quello reale.

MAE: 0.785

L'utilizzo di questa metrica, inesistente in letteratura e che da sola è fine a se stessa, viene proposta come misura per confronti futuri nostri o di altri.

6. Analisi degli shortage

Di base, nella simulazione non vengono posti limiti rispetto al numero di bici parcheggiabili in una determinata stazione. Stimare le dock size delle stazioni a partire dai dati reali permette di analizzare l'utilizzo delle stazioni e quindi ad individuare eventuali sovraffollamenti o sotto-utilizzi.

Questa analisi ha quindi lo scopo di identificare i problemi di **shortage**, di due diverse tipologie:

- shortage di **biciclette**: si tratta di stazioni vuote che, a causa del numero finito di biciclette, rimangono senza mezzi a disposizione per gli utenti. Queste mancanze causano i cosiddetti viaggi persi, risolvibili tramite un ribilanciamento manuale da parte dell'azienda. Il conteggio di questi viaggi persi può permetterci di analizzare quali stazioni necessitano di una maggiore affluenza di mezzi e quali invece ne possiedono fin troppi.
- shortage di **parcheggi**: riguarda le stazioni che posseggono un numero troppo elevato di mezzi inutilizzati, i quali vanno ad occupare tutti i posti disponibili. Questo fenomeno impedisce agli utenti di terminare in esse il proprio viaggio, costringendoli a ricalcolare il loro arrivo nella stazione più vicina.

Al fine di risolvere questi problemi, vengono testate diverse possibili soluzioni.

6.1. Soluzione 1: incrementare il numero di parcheggi

La prima proposta è quella di incrementare il numero di parcheggi in tutte le stazioni. Come descritto nel [capitolo 5](#), nella simulazione viene istanziato un numero fisso di biciclette, distribuite in maniera proporzionale rispetto alla grandezza della stazione inferita dai dati statistici. Partendo da questo valore (che chiameremo *bici iniziali*), la dimensione della stazione viene calcolata come segue:

$$\max(\text{parcheggi minimi}, \text{bici iniziali} * k)$$

dove *i parcheggi minimi* è la più piccola dimensione (intera) che devono avere le stazioni, mentre *k* è un numero reale che rappresenta un aumento percentuale rispetto al numero di bici iniziali.

Vengono quindi eseguite diverse simulazioni, tenendo traccia dei due tipi di shortage al variare di questi due parametri. I restanti parametri della simulazione sono le date (da 1 a 14 gennaio 2018) e il numero di bici totali (300, scelto a partire dai dati reali, in cui sono state individuate 282 bici uniche in circolazione).

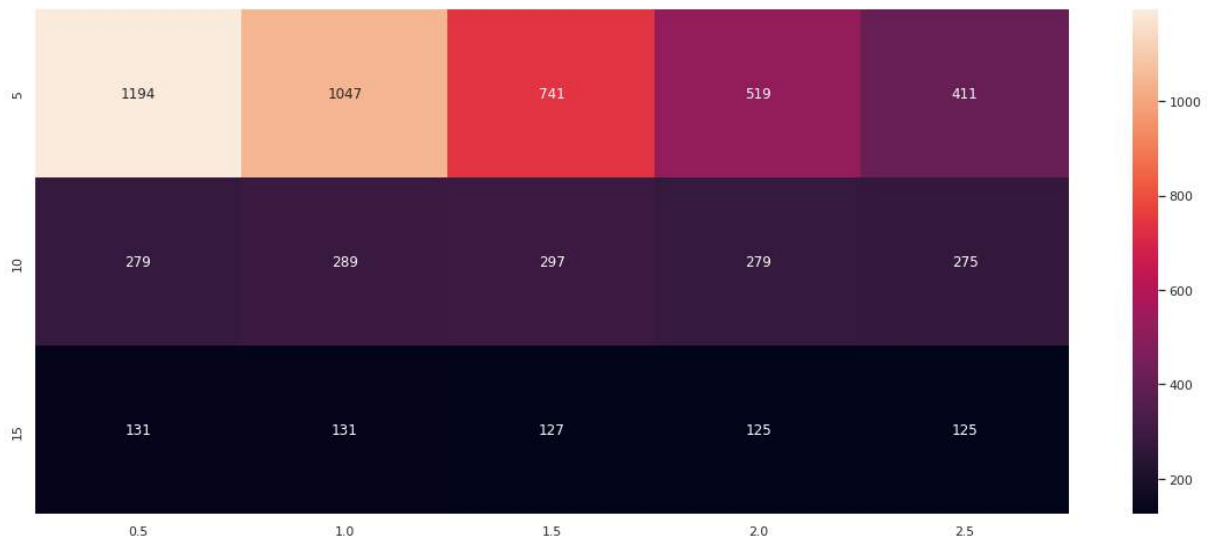


Figura 6.1. Shortage di parcheggi al variare di *parcheggi_minimi* (asse y) e *k* (asse x).



Figura 6.2. Shortage di biciclette al variare di *parcheggi_minimi* (asse y) e *k* (asse x).

Si nota come il problema delle stazioni piene (fig. 6.1) si risolva in maniera piuttosto evidente aumentando il numero di *parcheggi minimi*; nel caso in cui il numero di *parcheggi minimi* è 5, anche il valore *k* risulta essere efficace. La motivazione principale è legata allo “scaling” della funzione: avendo un numero di biciclette totali piuttosto basso e 85 stazioni, i valori scelti come parametro per *parcheggi_minimi* tendono ad avere una influenza maggiore e annichilire l’effetto di *k*.

Il problema delle stazioni vuote (fig. 6.2), invece, non sembra mostrare nessuna relazione con il numero di parcheggi.

In seguito a queste osservazioni, si è deciso di fissare il parametro k ad 1 e osservare i cambiamenti al variare del solo parametro *parcheggi minimi*.

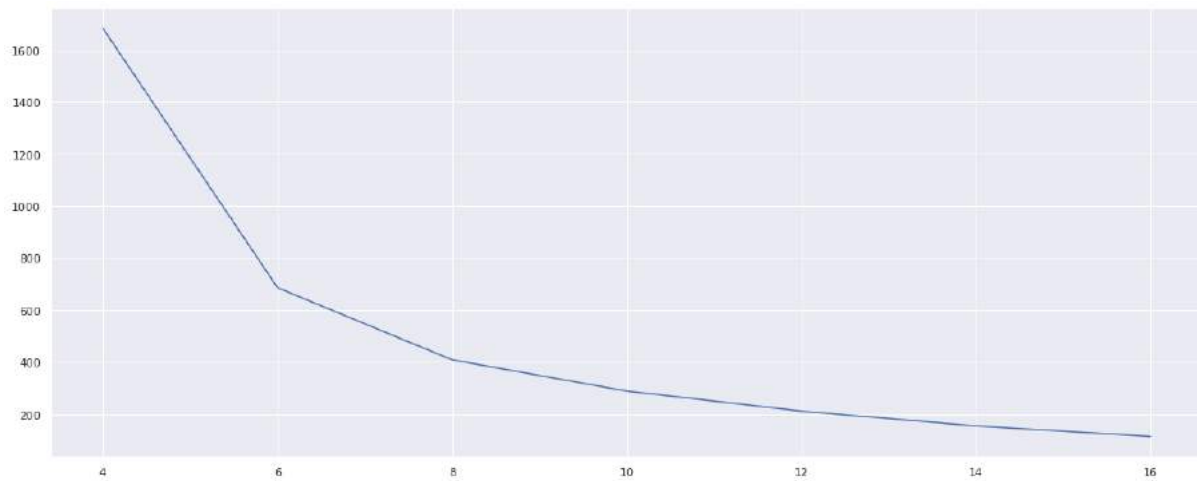


Figura 6.3. Shortage di parcheggi al variare di parcheggi minimi.

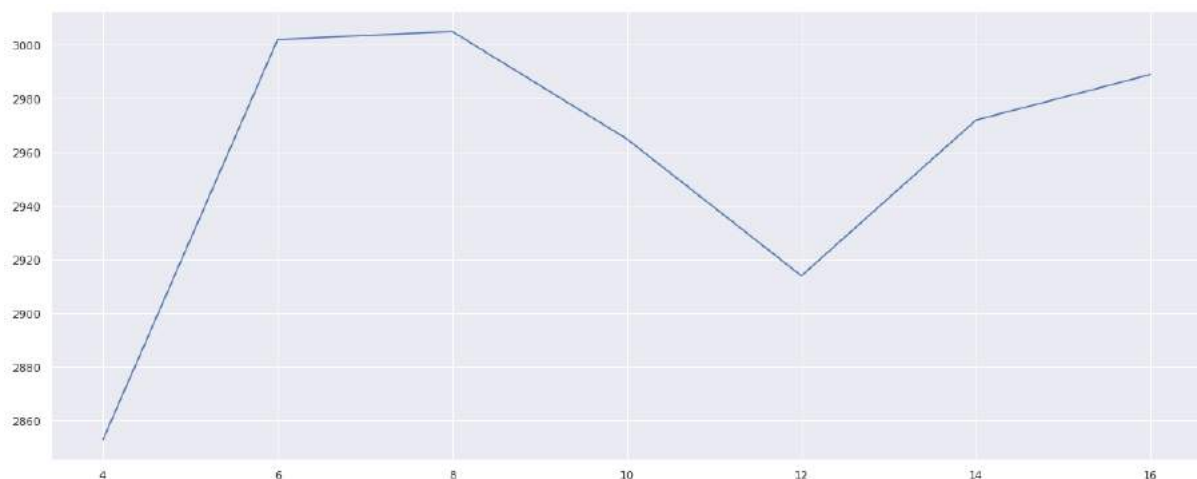


Figura 6.4. Shortage di biciclette al variare di parcheggi minimi.

Si nota come aumentando linearmente il numero di *parcheggi minimi*, diminuisce il numero di volte in cui l'utente è costretto a cambiare stazione di destinazione per mancanza di parcheggi (fig. 6.3); la mancanza di biciclette non sembra mostrare alcuna correlazione (fig. 6.4).

6.2. Soluzione 2: incrementare il numero di biciclette totali

La seconda proposta è quella di incrementare il numero di biciclette totali. In modo simile al caso precedente, sono state eseguite diverse simulazioni al variare del parametro *bici totale*.

Da notare che, aumentando le bici, aumenteranno di conseguenza anche il numero di parcheggi (ma in modo proporzionale, diversamente dalla prima proposta).

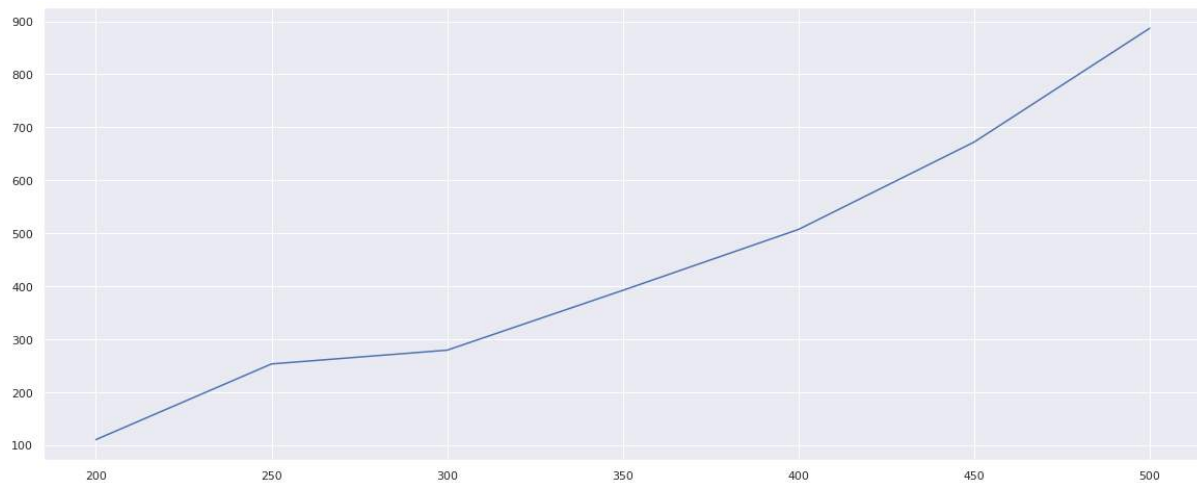


Figura 6.5. Shortage di parcheggi al variare di bici totali.

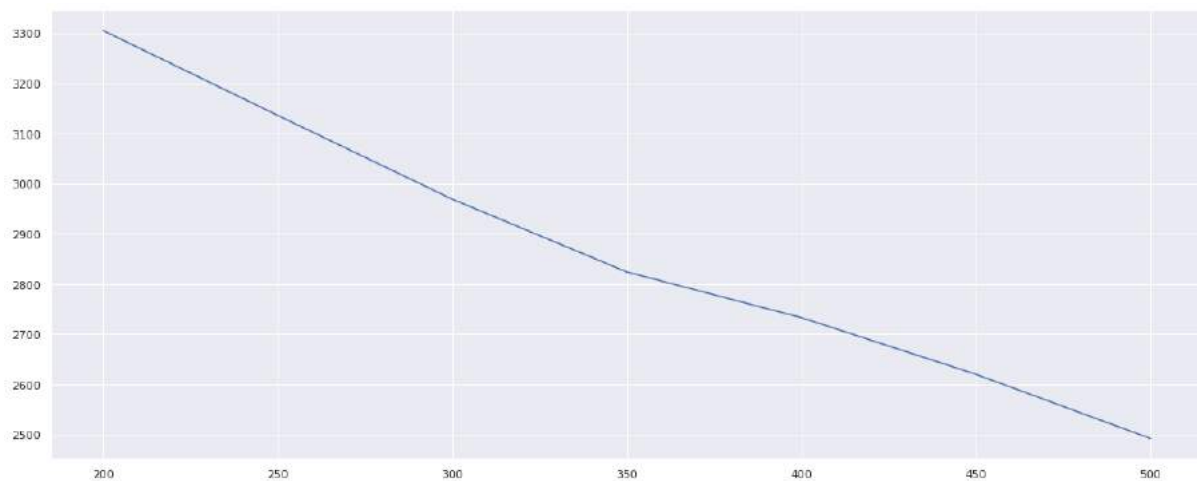


Figura 6.6. Shortage di biciclette al variare di bici totali.

Si nota come aumentando il numero di biciclette disponibili in ogni stazione, il numero di casi in cui l'utente non trova parcheggio nella stazione di arrivo aumenta drasticamente (fig. 6.5); invece, il numero di viaggi persi per mancanza di bici si riduce linearmente (fig. 6.6).

6.3. Soluzione 3: unione delle soluzioni precedenti

La terza proposta unisce le due precedenti ipotesi, combinando l'aumento di parcheggi e biciclette, al fine di analizzare i vari scenari ed individuare quello che riduca maggiormente

entrambi i problemi. Come nella prima analisi viene fissato il parametro k ad 1 e vengono analizzati i risultati al variare dei *parcheggi_minimi* e delle *bici_totali*.



Figura 6.7. Shortage di parcheggi al variare di *parcheggi_minimi* e *bici_totali*.



Figura 6.8. Shortage di biciclette al variare di *parcheggi_minimi* e *bici_totali*.

È lampante la relazione mostrata in fig. 6.7, che mostra come all'aumentare del numero di biciclette, aumenta il numero di casi nel quale l'utente non ha potuto parcheggiare il proprio mezzo per mancanza di parcheggi. Mentre aumentando il numero di parcheggi minimi, il numero di shortages di parcheggi si riduce.

Gli shortage di biciclette (fig. 6.8), invece, tendono a diminuire al crescere del numero di biciclette in circolazione. Se però vengono aumentati il numero di parcheggi minimi, gli shortage aumentano significativamente.

In conclusione, per ottenere i migliori risultati, la configurazione consigliata sarebbe quella con massimo numero di biciclette e parcheggi minimi. Questa soluzione, triviale, è però puramente teorica e non applicabile al mondo reale, nel quale esistono vincoli materiali e non è possibile avere disponibilità illimitate. Rimane comunque di valore nell'indirizzare le scelte dell'azienda.



Figura 6.9. Configurazione ottimale.

In conclusione, nelle analisi che seguiranno verranno mantenuti dei parametri che si rifanno a quelli reali.

6.4. Shortage nelle stazioni

Dopo aver costruito una simulazione che fosse abbastanza fedele alla realtà, l'abbiamo sfruttata per analizzare fenomeni che non erano presenti nei dati. Infatti, non esistono informazioni riguardo i casi di shortage di biciclette o parcheggi nelle stazioni, ma con l'ausilio della simulazione siamo in grado di generare queste informazioni.

Le analisi che seguono riguardano dunque questi problemi, dovuti alla grandezza delle stazioni in generale, cioè all'andamento del numero di parcheggi liberi e di biciclette disponibili in ciascuna stazione. La simulazione che è stata eseguita riprende le due settimane precedentemente analizzate (1-14 gennaio 2018), con 300 bici totali.

Osservando il numero di shortage di biciclette nelle stazioni (fig. 6.10), si nota che alcune stazioni presentano questo problema un elevato numero di volte; ovvero, andrebbero ingrandite o bilanciate (escono più bici di quante ne entrano).

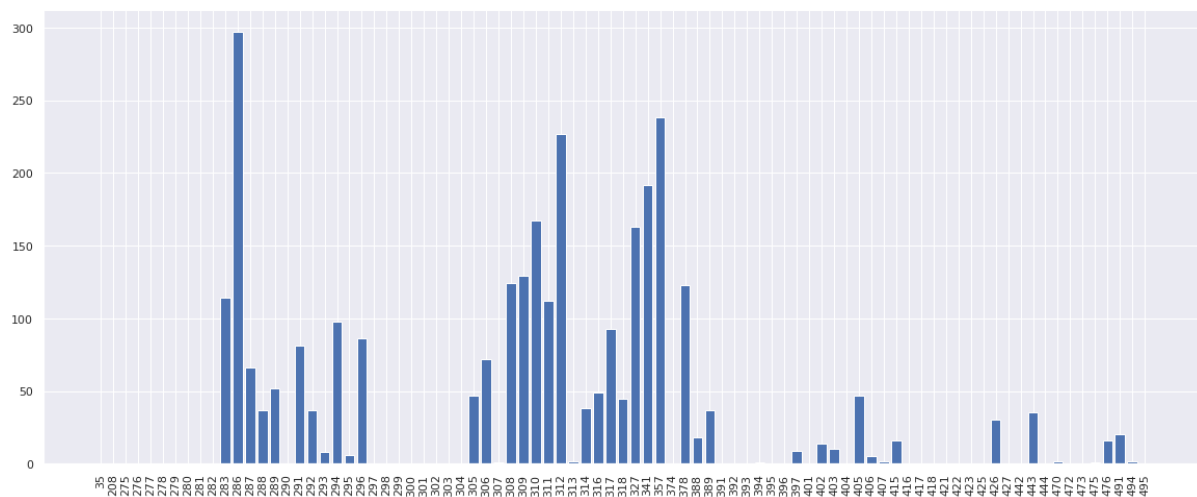


Figura 6.10. Numero di shortage di biciclette in ciascuna stazione.

Un esempio è la stazione 286, che è la stazione che presenta più il maggior numero di casi di shortage, ovvero quasi 300 viaggi persi per mancanza di biciclette. Questo è dovuto al fatto che il numero di biciclette presenti in stazione (inizialmente 3) sono state usate tutte, e non è avvenuto nessun arrivo nella stazione (vedi fig. 6.11. e 6.12).

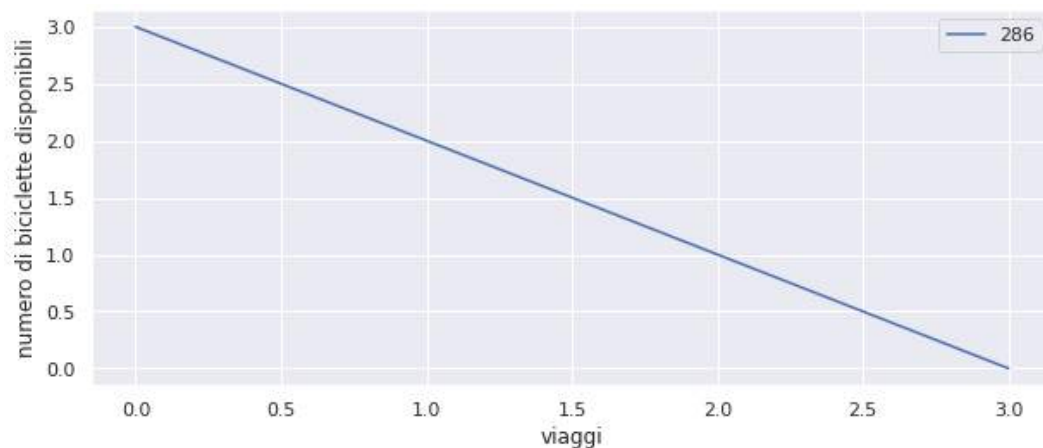


Figura 6.11. Variazione delle biciclette disponibili nel tempo*, nella stazione 286.

* Il grafico mostrerà un aumento di un'unità per ogni bici in entrata, e un decremento per ogni bici in uscita.

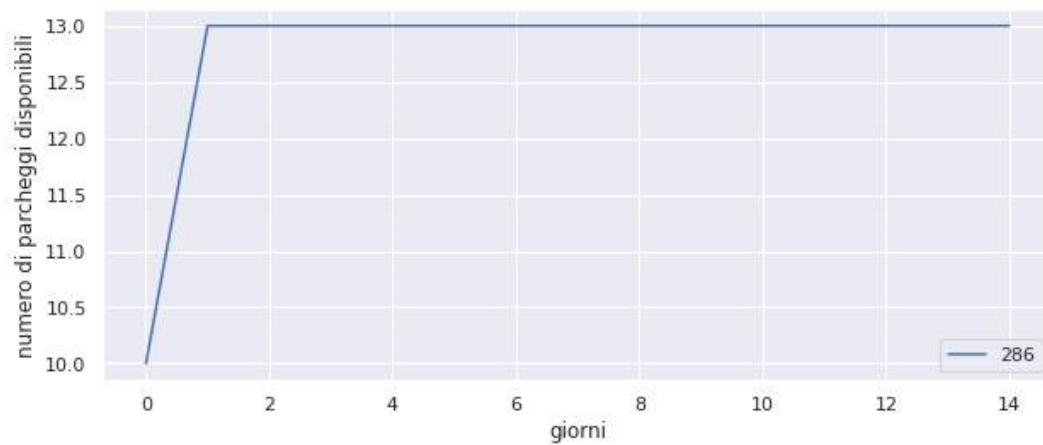


Figura 6.12. Variazione dei parcheggi disponibili nel tempo, nella stazione 286.

Andando invece ad analizzare gli shortage di parcheggi, si vede che il problema si presenta di meno. Come prima, viene mostrato l'andamento di biciclette e parcheggi disponibili per la stazione più "problematica", ovvero la 374, che si comporta esattamente nel modo inverso.

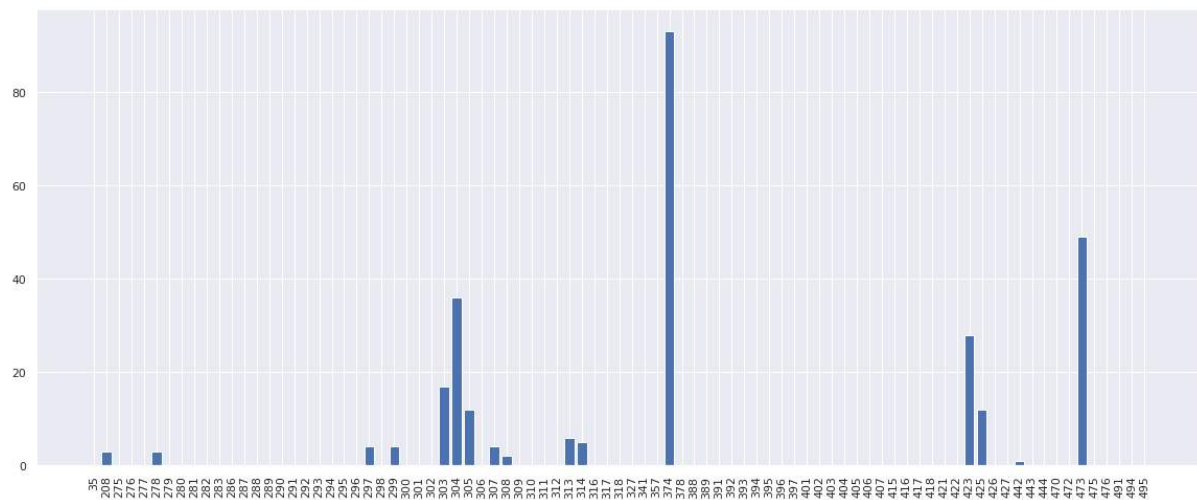


Figura 6.13. Numero di shortage di parcheggi in ciascuna stazione.

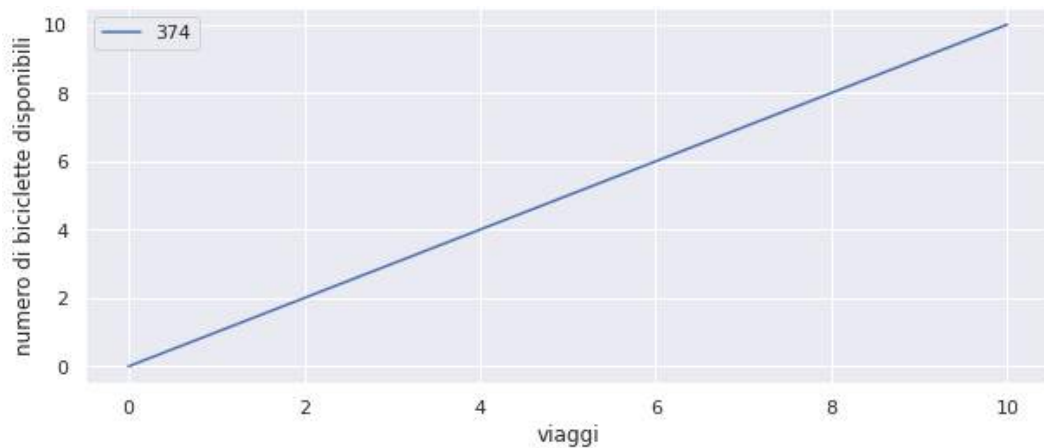


Figura 6.14. Variazione delle biciclette disponibili nel tempo, nella stazione 374.

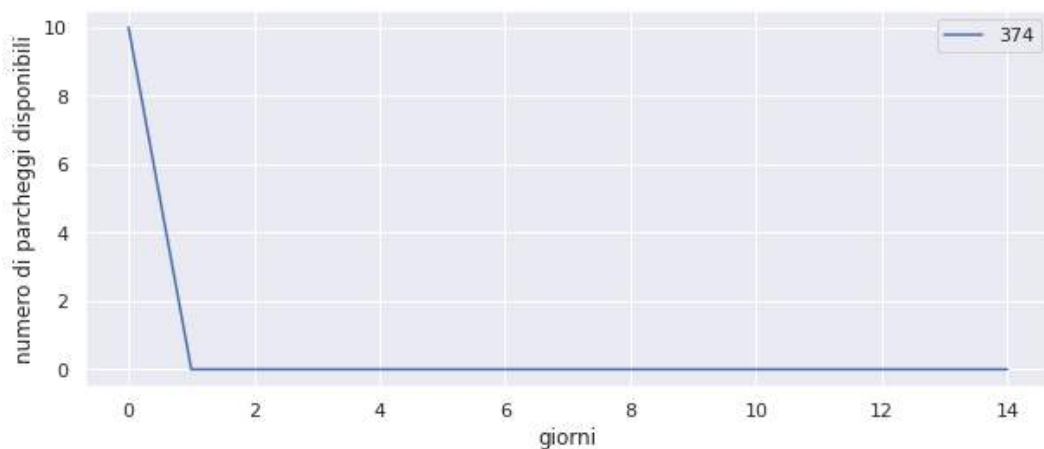


Figura 6.15. Variazione dei parcheggi disponibili nel tempo, nella stazione 374.

Rebalancing

Il problema del rebalancing nel nostro caso è stato analizzato seguendo una metodologia naive, ossia si è cercato di risolvere i problemi di shortage semplicemente aumentando il numero totale di biciclette nel sistema.

Inoltre, sono state effettuate delle analisi in dettaglio per ogni stazione, al fine di valutare gli utilizzi e categorizzare in stazioni funzionanti, sovraffollate o inutilizzate in base al traffico in entrata e uscita, basandosi sull'assunzione che se il flusso è unidirezionale allora è presente un problema. Questa valutazione potrebbe servire in futuro come base sulla quale costruire un criterio funzionante per effettuare il rebalancing delle biciclette.

Come si può notare in fig. 6.16, le stazioni completamente funzionali dal punto di vista del flusso di biciclette sono all'incirca il 20%; il 27% sono sovraffollate mentre il 53% sono inutilizzate.

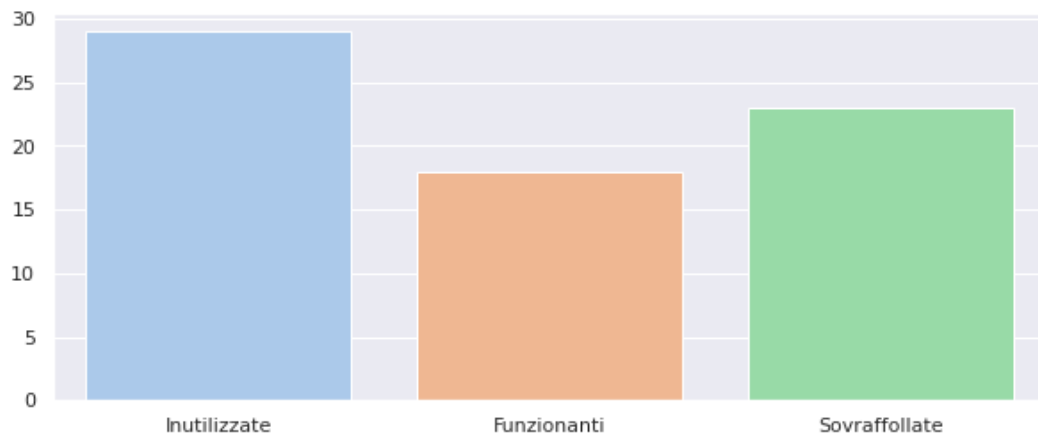


Figura 6.16. Distribuzione delle stazioni funzionanti o meno.

Per classificare le stazioni è stato calcolato il **coefficiente angolare** generale della funzione ottenuta dal numero di biciclette disponibili al variare dei viaggi. Se il coefficiente angolare ha un valore superiore a 0.1 allora la stazione è classificata come sovraffollata, viceversa se inferiore a -0.1 è classificata come inutilizzata. La stazione è ritenuta funzionante se il coefficiente angolare ha un valore compreso tra -0.1 e 0.1 (esclusi). Sono visibili degli esempi in figura 6.17 e 6.18.

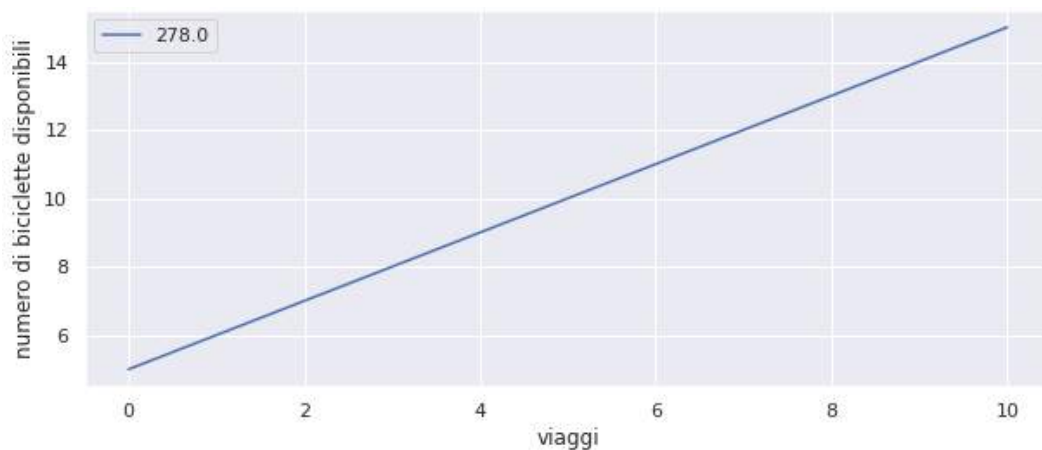


Figura 6.17. Esempio di stazione sovraffollata (coefficiente angolare: 1).

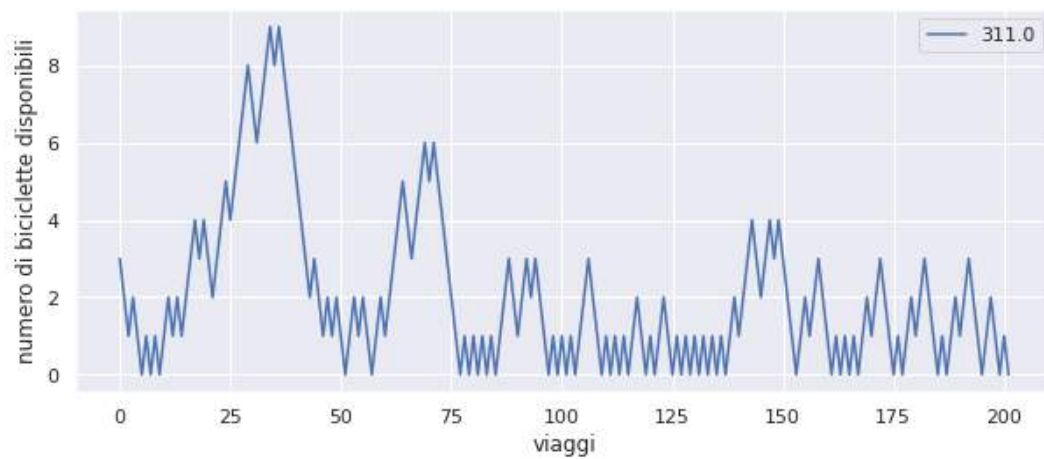


Figura 6.18. Esempio di stazione funzionante (coefficiente angolare: -0.015).

7. Scenari what-if

7.1. Se ci fossero più / meno biciclette?

In questa sezione si analizzeranno le conseguenze legate all'aumento o riduzione delle biciclette nel sistema.

Dato che la simulazione si basa su dati statistici relativi ad un determinato numero di utilizzatori, ci si aspetta che aumentare il numero di biciclette in circolazione non porterebbe a cambiamenti nella simulazione. Al contrario, riducendo il numero di biciclette in circolazione, la simulazione dovrebbe rallentare e diventare statica, riducendo la complessità del problema.

È stata eseguita una simulazione sulle prime due settimane del 2018 (1 gennaio - 14 gennaio). Nei dati reali sono state individuate 282 bici uniche in circolazione e, di conseguenza, è stata testata la seguente combinazione di biciclette: 50, 100, 200, 300, 400, 500, 1000, 2000, 5000, 10000. Inoltre, sono stati fissati i parcheggi, con stazioni grandi almeno 10 posti (o oltre, con il doppio di parcheggi rispetto alle bici iniziali).

Verranno di seguito analizzati tre diversi aspetti della simulazione.

Biciclette

Tra le biciclette generate nella fase iniziale, non tutte vengono effettivamente utilizzate nella simulazione (fig. 7.1 e 7.2). Inoltre, si nota che meno bici ci sono, più queste verranno utilizzate.

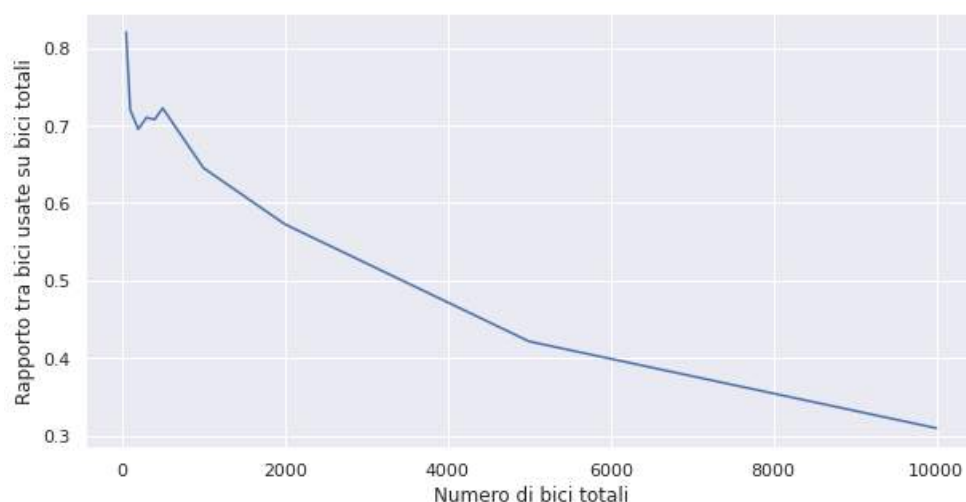


Figura 7.1. Variazione dell'utilizzo delle biciclette al variare del loro volume totale.

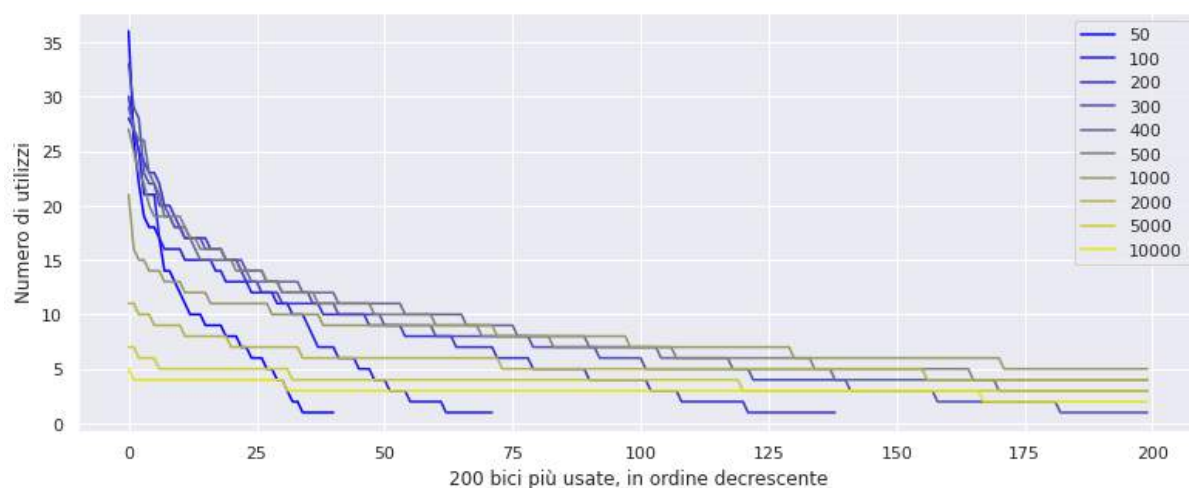


Figura 7.2. Utilizzo delle biciclette al variare del loro volume totale (legenda).

Viaggi

L'osservazione più ovvia riguarda il numero di viaggi (fig. 7.3): aumentano con l'aumentare del numero di bici (i viaggi persi per mancanza di bici si riducono). Si nota inoltre che la crescita non è lineare: continuando ad aumentare il numero di possibili bici, si arriva ad un punto di stallo in cui non cambierà più l'utilizzo. Per quanto riguarda la durata (fig. 7.4 e 7.5) si vede che diminuiscono sia la media che la deviazione standard dei viaggi. Si deduce che si aggiungono molti viaggi brevi, che portano ad una riduzione di queste statistiche.

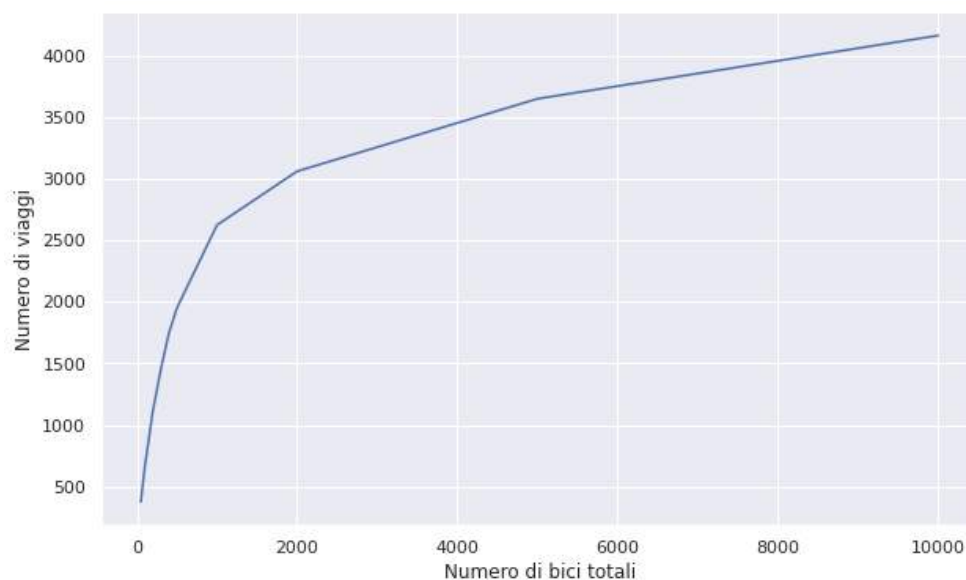


Figura 7.3. Numero di viaggi al variare del numero di biciclette totali.

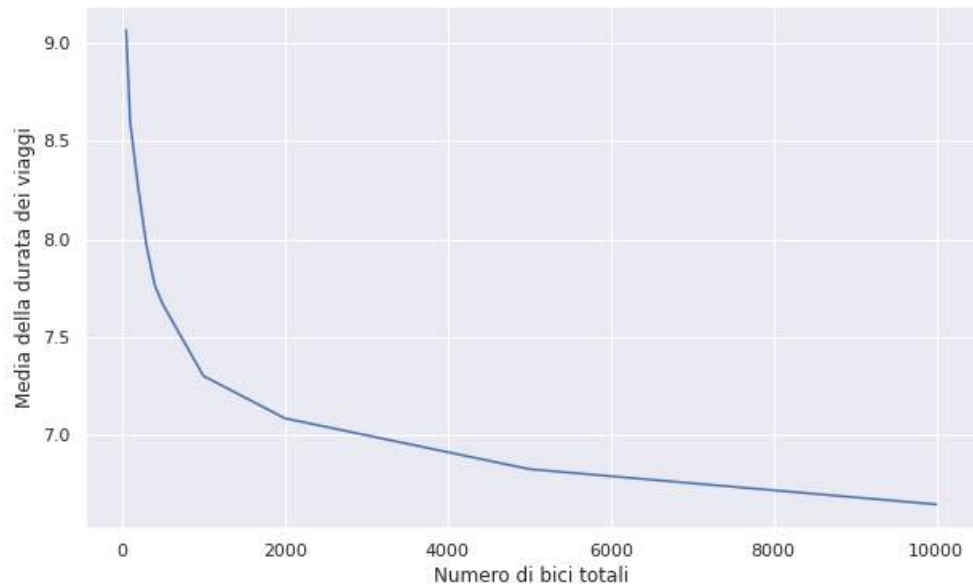


Figura 7.4. Tempo medio dei viaggi al variare del numero di biciclette totali.

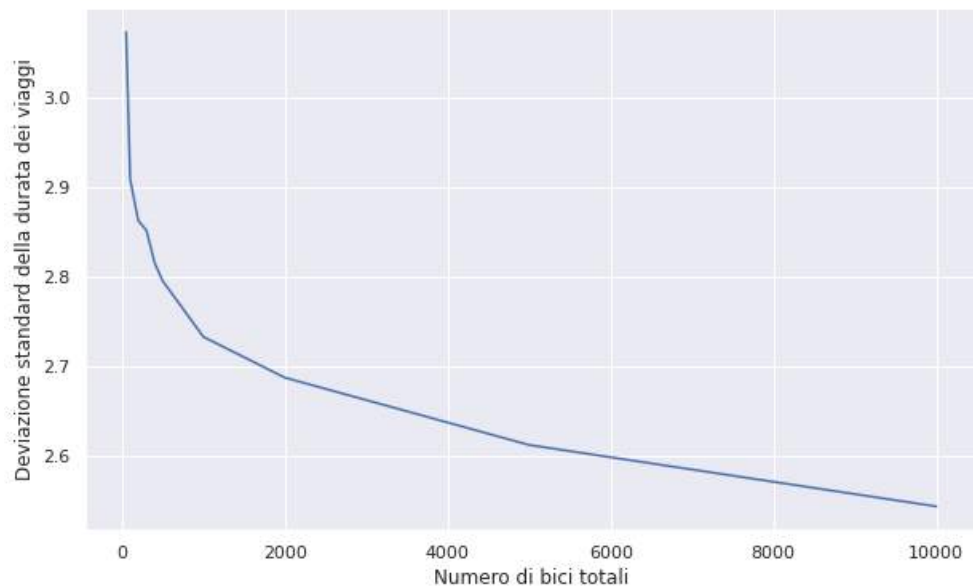


Figura 7.5. Deviazione standard dei viaggi al variare del numero di biciclette totali.

Stazioni

Andando ad osservare l'utilizzo delle stazioni, è evidente l'intensificazione del traffico all'aumentare delle biciclette totali (fig. 7.6). Osservando invece le distribuzioni di utilizzo delle stazioni (fig. 7.7), si desume che queste non variano modificando il numero di bici totali.

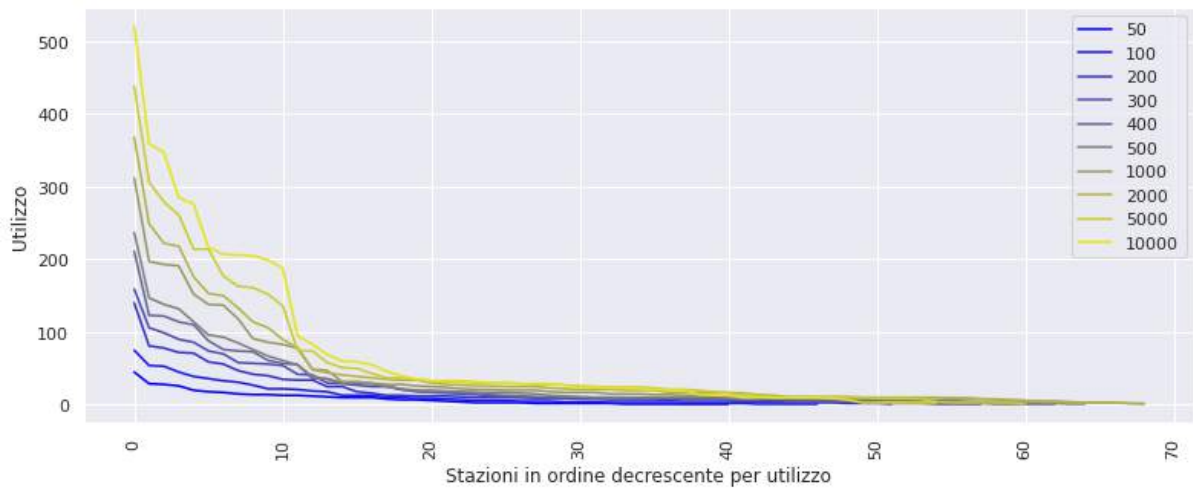


Figura 7.6. Utilizzo delle stazioni al variare del volume totale di biciclette (legenda).

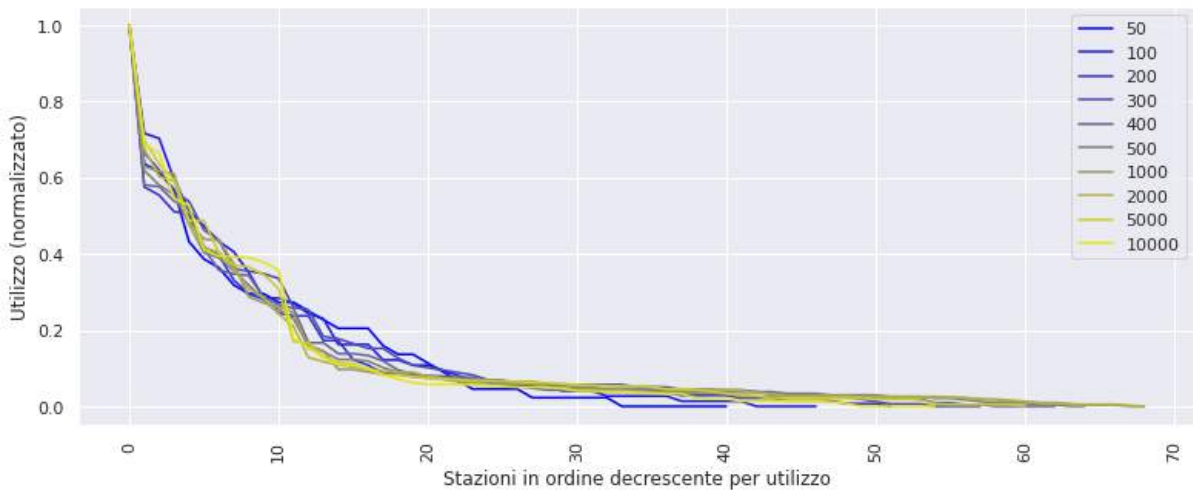


Figura 7.7. Utilizzo (normalizzato) delle stazioni al variare del volume totale di biciclette (legenda).

In conclusione, si può dire che, nell'esempio portato in discussione, un lieve aumento delle biciclette porterebbe a dei benefici in termini di numero di viaggi.

7.2. Se ci fossero più / meno stazioni?

Questo tipo di scenario può essere utile al fine di stabilire se può essere utile aprire nuove stazioni oppure se possono nascere problemi di fronte ad una ipotetica situazione di lavori in corso (o chiusura definitiva) di una stazione.

Purtroppo, diventa subito chiaro che esplorare lo scenario di nuove stazioni con i dati / metodologia attuali è impossibile, in quanto non si riuscirebbe a simulare l'utilizzo di una nuova stazione della quale non si hanno informazioni. Di conseguenza, in questo studio analizzeremo il caso di **una sola stazione chiusa**.

Dal punto di vista della simulazione, tutti quei viaggi che coinvolgerebbero la stazione eliminata vengono reindirizzati sulle stazioni vicine. In particolare:

- nel caso della **partenza**, viene calcolato il numero di bici che uscirebbero ad ogni istante della simulazione. Questi viaggi vengono spostati sulle stazioni vicine, scegliendole in modo casuale con probabilità inversamente proporzionale alla distanza dalla stazione eliminata. Se non esistono stazioni nell'intorno di 500 metri, si ipotizza che l'utente opterà per un mezzo di trasporto differente dalla bicicletta e il viaggio verrà conteggiato tra quelli "persi".
- nel caso dell'**arrivo**, se una bici finirebbe randomicamente nella stazione eliminata, viene scelta una stazione vicina nello stesso modo precedentemente descritto.

Se non ci sono stazioni vicine (entro 500m), terminano la corsa nella stazione più vicina, anche se oltre al limite preimpostato.

È da tenere a mente che, nei viaggi reali, le distanze tra due stazioni fissate possono variare, poiché possono essere intrapresi percorsi differenti. Per avere un valore che possa a grandi linee riprendere le tempistiche reali, verrà calcolata la distanza in linea retta tra le coordinate delle suddette stazioni.

Inoltre, tutte le biciclette che ad inizio simulazione verrebbero normalmente assegnate a quella stazione, vengono invece redistribuite tra tutte le rimanenti stazioni.

Per avere un numero sufficiente di dati per dei test statistici significativi, sono state eseguite 30 simulazioni con stazione 311 (variando il seed) e 30 simulazioni senza quella stazione. La scelta di questa stazione è dettata dal fatto che è molto frequentata, ma non necessariamente la più usata. Come per il what-if 1, la simulazione è stata eseguita per le prime due settimane del 2018, con 300 bici (numero preso dai dati reali) e parcheggi fissati.

Verranno di seguito analizzati tre aspetti della simulazione.

Viaggi

Osservando i viaggi effettuati dalle biciclette, se ne possono analizzare numero e durata (tab. 7.1). Applicando un test di Shapiro-Wilk, si osserva che solo nel secondo caso le variabili si distribuiscono secondo una normale: in quel caso, per confermare o meno se le medie tra i due gruppi sono statisticamente significative, bisognerà eseguire un test T di Student, altrimenti, un test di Mann-Whitney.

	Con stazione 311	Senza stazione 311	Test differenza medie
Numero di viaggi effettuati	avg: 1690.60 std: 81.20	avg: 1733.73 std: 64.91	Mann-Whitney statistic: 568.5 p-value: 0.081
Numero di viaggi persi	avg: 1050.67 std: 87.49	avg: 1031.77 std: 69.28	T di student statistic: 0.912 p-value: 0.366
Durata dei viaggi effettuati	avg: 10.26 std: 3.87	avg: 10.22 std: 3.89	Mann-Whitney statistic: 1311124590.5 p-value: 0.072

Tabella 7.1. Statistiche riguardanti i viaggi con e senza stazione 311.

I test applicati confermano che non esistono differenze statisticamente significative, considerando un livello di confidenza al 95%, in alcuna delle tre categorie, eliminando una stazione piuttosto comune. L'apparente maggiore numero di viaggi è dovuto al fatto che, in questo caso, i nuovi viaggi indirizzati sulle stazioni vicine risultano più brevi, andando a generare un traffico lievemente maggiore.

Stazioni

Osservando l'uso delle stazioni, si nota la stessa distribuzione nei due diversi casi, con l'eccezione che vengono distribuiti i viaggi della stazione 311 alle altre stazioni.

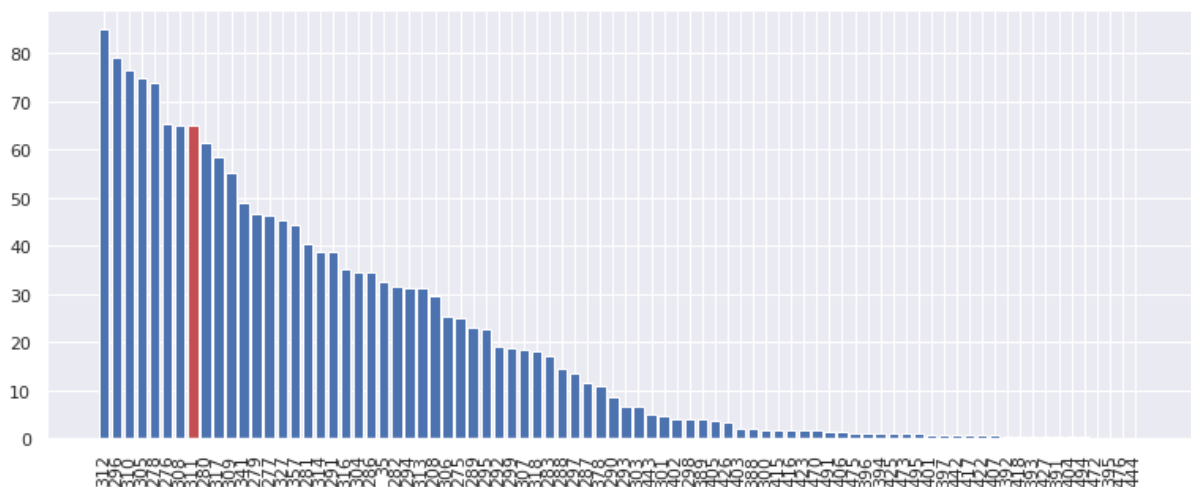


Figura 7.8. Utilizzo medio delle stazioni tra le diverse simulazioni, includendo la 311.

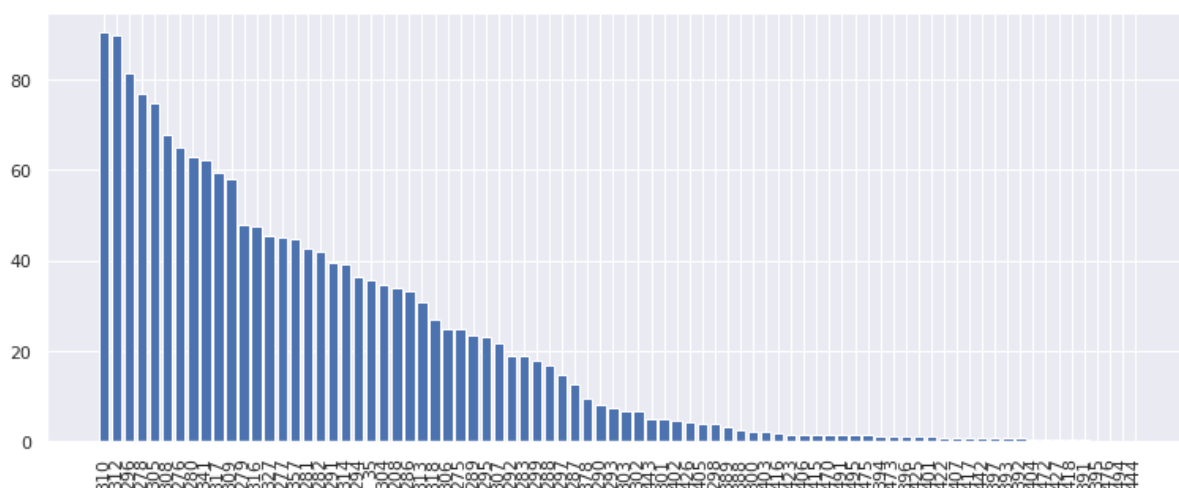


Figura 7.9. Utilizzo medio delle stazioni tra le diverse simulazioni, escludendo la 311.

Biciclette

Di nuovo, nell'analisi delle biciclette (fig. 7.10, 7.11, 7.12) non vengono riscontrate differenze significative. Anche nel caso dei viaggi persi, in cui può sembrare che ci sia una differenza, a livello statistico non si riscontra.

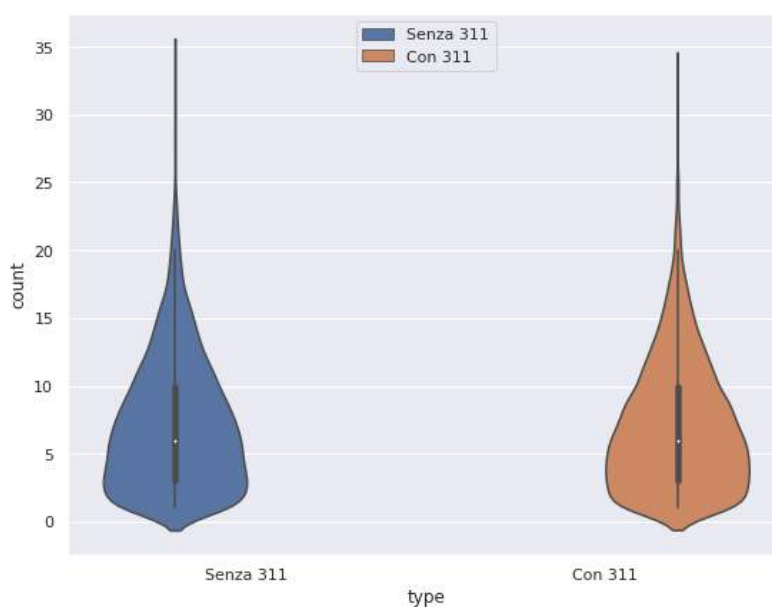


Figura 7.10. Distribuzione del numero di viaggi effettuati dalle biciclette.

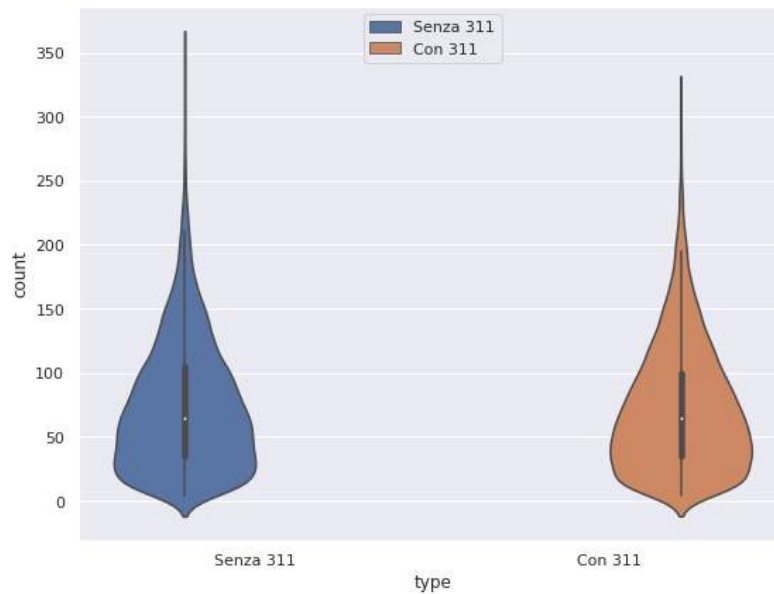


Figura 7.11. Distribuzione della durata dei viaggi effettuati dalle biciclette.

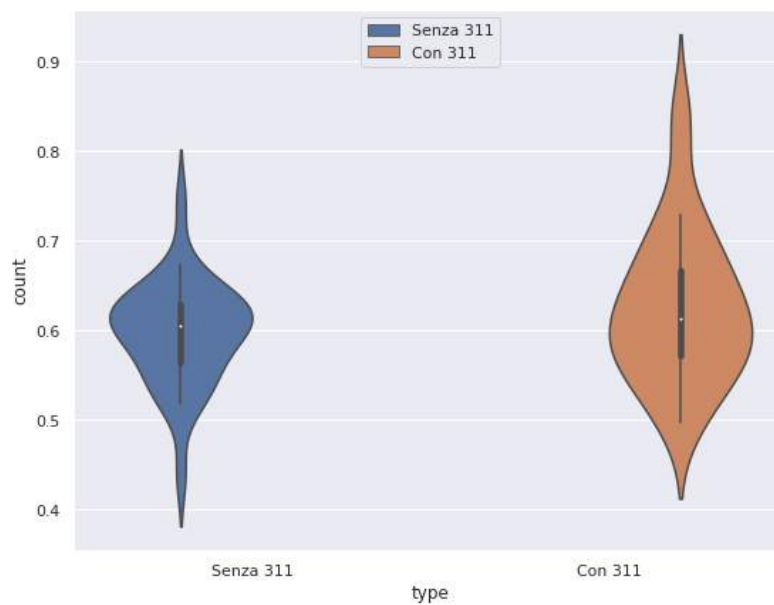


Figura 7.12. Distribuzione del rapporto tra viaggi persi su viaggi effettuati, per simulazione.

In conclusione, in questi dati (questa città in questi due anni), togliere una stazione molto usata non porta a conseguenze rilevanti. Non ha senso provare altre stazioni meno utilizzate, se già questa non dà differenze significative.

8. Conclusioni e Sviluppi Futuri

In questo progetto è stata sviluppata una simulazione per il sistema di bike sharing di San Jose. Il software si dimostra coerente con i dati reali, lasciando comunque spazio a dei miglioramenti dal punto di vista dell'adattamento a cambiamenti annuali, dovuti ad eventi terzi (es pandemia Covid-19).

È stata descritta la tematica dello shortage, mostrando come la situazione possa migliorare con il solo incremento del numero di biciclette totali e di parcheggi minimi nelle stazioni.

Infine, sono stati affrontati due scenari what-if. Nel primo, si è concluso che aumentare il numero di biciclette totali porta a dei benefici, riducendo il numero di viaggi persi. Nel secondo, è stato mostrato che la chiusura di una stazione, anche se molto frequentata, non porterebbe a drastiche variazioni in termini di viaggi persi e nelle redirezioni degli arrivi.

Tra gli sviluppi futuri rientrano:

- il calcolo di stime più accurate nella durata dei viaggi, integrando i dati con i tempi di percorrenza (tra quartieri) forniti da Uber [\[15\]](#).
- provare diversi metodi di inizializzazione delle biciclette nelle stazioni. Ad oggi distribuite osservando le statistiche di utilizzo delle stazioni negli anni 2018-2019 con le quali è stato assegnato un punteggio di importanza e un conseguente numero di biciclette.
- l'esplorazione dello scenario what-if "Se si aprissero una o più nuove stazioni?", attualmente impossibile per via della mancanza di dati, utilizzando delle proiezioni calcolate sulla base dei dati esistenti.
- una modifica alla simulazione, in modo tale da adattarla a modifiche nel tempo, e.g. mostrando la variazione durante la pandemia. Attualmente questi dati sono stati appositamente ignorati per costruire una simulazione più semplice ma solida.
- stima delle perdite dovute ai viaggi non effettuati e i conseguenti costi necessari a risolvere il problema del rebalancing, in modo da ottimizzare ogni quanto effettuare gli spostamenti. Attualmente sono sconosciuti i costi di queste operazioni, che eventualmente potrebbero essere stimati.
- le biciclette istanziate sono uniche e questo ci permette di tenere traccia di dati come i viaggi effettuati e i km percorsi; questi dati, se integrati adeguatamente, possono servire per effettuare analisi più specifiche, come ad esempio un calcolo preventivo sull'applicazione di esercizi di manutenzione ordinaria.

- Gli utenti aspettano se la stazione e' piena, mentre noi ci spostiamo subito su un'altra [da saltzman]

9. Bibliografia

- [1] 136 million trips taken on shared bikes and scooters across the U.S. in 2019. (2020, August 27). Retrieved September 7, 2022, from National Association of City Transportation Officials website: <https://nacto.org/2020/08/27/136-million-trips-taken-on-shared-bikes-and-scooters-across-the-u-s-in-2019/>
- [2] Mallonee, L. (2017, June 30). Photo of the week: A dizzying view of a bicycle graveyard in China. WIRED. Retrieved from <https://www.wired.com/story/photo-of-the-week-a-dizzying-view-of-a-bicycle-graveyard-in-china/>
- [3] Bike shares near you. (2022). Retrieved September 7, 2022, from Lyft Bikes website: <https://www.lyft.com/bikes>
- [4] riccardomerlo. (2022). GitHub - Riccardomerlo/bike_sharing_simulation: Homemade simulator ab-using distributed probabilities over gathered statistical data. Retrieved September 7, 2022, from GitHub website: https://github.com/riccardomerlo/bike_sharing_simulation
- [5] Saltzman, Robert & Bradford, Richard. (2016). Simulating a More Efficient Bike Sharing System. *Journal of Supply Chain and Operations Management*. 36-47.
- [6] Soriguera, F., Casado, V., & Jiménez, E. (2018). A simulation model for public bike-sharing systems. *Transportation Research Procedia*, 33, 139–146. <https://doi.org/10.1016/j.trpro.2018.10.086>
- [7] Hebenstreit, C., Fellendorf, M. (2018). A dynamic bike sharing module for agent-based transport simulation, within multimodal context. *Procedia Computer Science*, 130, 65–72. <https://doi.org/10.1016/j.procs.2018.04.013>
- [8] Lammel, G., Large-scale multi-agent transport simulation in the urban context, Gregor Lämmel | g.laemmel@fz-juelich.de, June 3, 2014
- [9] Fernández, A., Billhardt, H., Ossowski, S., & Sánchez, Ó. (2020). Bike3S: A tool for bike sharing systems simulation. *Journal of Simulation*, 14(4), 278–294. <https://doi.org/10.1080/17477778.2020.1718022>
- [10] Mukku, V. D., Salah, I. H., & Assmann, T. (2021). Simulation Testbed for the next-generation Bike-Sharing System with Self-driving Cargo-Bikes. *IFAC-PapersOnLine*, 54(1), 1098–1103. <https://doi.org/10.1016/j.ifacol.2021.08.129>
- [11] Haj Salah, I., Mukku, V. D., Schmidt, S., & Assmann, T. (2020). A conceptual model for the simulation of the next generation bike-sharing system with self-driving cargo-bikes. In *Advances in Mobility-as-a-Service Systems* (pp. 253–262). Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-61075-3_25

- [12] Jian, N., Freund, D., Wiberg, H. M., & Henderson, S. G. (2016, December). Simulation optimization for a large-scale bike-sharing system. In 2016 Winter Simulation Conference (WSC) (pp. 602-613). IEEE.
- [13] System Data. (2022). Retrieved June, 2022, from Lyft website:
<https://www.lyft.com/bikes/bay-wheels/system-data>
- [14] Contributors to Wikimedia projects. (2022, September 2). Poisson distribution. Wikipedia. https://en.wikipedia.org/wiki/Poisson_distribution
- [15] Uber Movement. (2022). https://movement.uber.com/explore/san_francisco/