Simulation

Group Project - 1

Due March 6th, 2026 at 16:00

1 Problem Description

As you graduate from our MSc programme, you begin working as a simulation analyst at a consulting firm in Edinburgh. You and your teammate are assigned to a project for a ride-sharing company, BoxCar, operating in Squareshire, a county named for its unusual square-shaped boundaries, measuring 20 miles on each side. A rather bland map of Squareshire is provided in Figure 1.

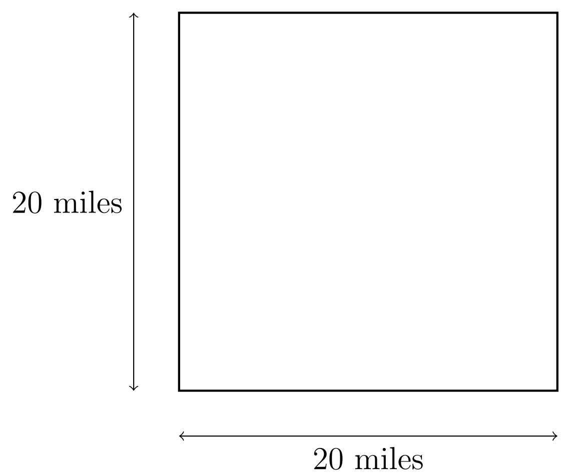

Figure 1: A bland map of Squareshire

BoxCar operates using an app developed by the company to match available taxis (drivers) with the customers (riders) arriving at random times. Through extensive discussions with BoxCar executives, you found out that the company believes that the ride-sharing system operates as follows:

1. **Taxis/Drivers:**
    - All BoxCar taxis are standard and equally preferable, i.e., there is no luxury, XLarge, etc. distinction between taxis.
    - The drivers make themselves (and their taxis) available in the app at random times with an inter-arrival of exponential(3/hour).
    - Once a driver becomes available, the driver stays available for a random time which is uniformly distributed between 5 and 8 hours and then goes offline. The length of availability is known to the driver, but BoxCar does not know about it

and hence, cannot make assignments using this. If the driver is serving a rider at the time s/he was planning to get offline, then the driver waits until s/he drops the rider at the destination and then goes offline.

- The location where the driver becomes available is equally likely to be anywhere in Squareshire, i.e., follows a uniform distribution. The driver stays at that location until being assigned to a passenger. Similarly, when a driver drops a rider after reaching the destination, s/he stays there (at the last destination) until being assigned to another rider.

2. **Riders (Customers):**

- The riders arrive at random times to demand a taxi ride with an inter-arrival time of exponential(30/hour).
- The origin (the point where the rider appears to demand the ride) and the destination of the trip are independent of each other and is equally likely to be anywhere in Squareshire.
- Each arriving customer has an exponential(5/hour) patience times and if it is not matched with a driver within this patience time, s/he cancels the request and leaves the system. Note that, it is not necessary for the ride to start within this patience time and it is enough for the rider to be informed that a taxi is driving to the rider for the pick-up. Once the rider is assigned a driver, the rider does not leave the system until arrival at the requested destination.

3. **Rider-Driver Matching and the Trip:**

- When a rider demands a ride from the app, the app checks the available/idle drivers, i.e., drivers who are not currently serving a customer. If there exists an idle driver, the app matches the closest driver with the rider and the driver starts driving for picking up the customer from the rider's origin location.
- Similarly, when a driver drops a rider at the rider's destination, the app checks if there are any riders waiting to be matched with a driver. If there are unmatched riders waiting, then the app assigns the driver to the closest unmatched rider and the driver starts driving for picking up the customer from the rider's origin location.
- Once a driver and a rider is matched as described above, the driver starts driving towards to rider to pick her/him up. When the driver picks up the rider, then s/he starts to drive towards the rider's destination for drop-off.
- The length of each trip (trip from location of the driver to origin for pick-up and trip from origin to destination for drop-off) depends on the Euclidean distance between points (denote as d_{OD}). It is assumed that the average speed is approximately 20 miles per hour and the expected trip time is $\mu_t = d_{OD}/20$ and the actual trip time is uniformly distributed between $0.8\mu_t$ and $1.2\mu_t$.

For example, taking the lower left corner of Squareshire as the coordinate (0,0), suppose a rider wishes to travel from (1,1) to (10, 13). Then, the distance from origin to the destination is $d_{OD} = \sqrt{(10-1)^2 + (13-1)^2} = 15$ and the expected trip time is $\mu_t = 15/20$ hours $= 45$ minutes. So the actual trip time is uniformly distributed between $(0.8\mu_t, 1.2\mu_t) = (36, 44)$ minutes.

- Once the trip finishes, the rider pays the driver an initial charge of £3 and £2 per each mile travelled from origin to the destination. Note that the distance for the payment excludes the distance between the driver's location to the origin (pick-up location).
- The driver incurs a petrol cost of £0.20 per mile.

2 Goals of the Project

The goal of this project is to evaluate the system's performance in terms of rider and driver satisfaction and to propose improvements.

Rider satisfaction can be measured based on the following factors:

- Rider abandonments due to driver unavailability
- Rider waiting time for pick-ups

On the other hand, driver satisfaction is influenced by:

- Average earnings per hour
- Fairness among drivers
- Sufficient resting time

BoxCar expects you, as a consultant, to assess the system's current performance and recommend ways they can implement to improve it.

3 Data

The description above includes almost all the input parameters necessary to model the system. However, these parameters rely on the beliefs of the BoxCar team and may not necessarily reflect the reality. The BoxCar team provided a small dataset covering the first few days of December 2025 and will be happy if you can provide further insights. The data provided consist of an excel file for rider data and another for driver data. The rider data includes

- The pick-up and drop-off locations as coordinates. The southwest corner (lower left corner) of Squareshire is considered as (0,0).

- Final status of the request, indicating whether the rider was successfully dropped off at the destination or abandoned the trip by canceling the request.
- The times when the rider request was received, the rider was picked up and the rider was dropped off. The pick up and drop off times are missing if the customer abandoned without being picked up.

The driver data is similar and includes

- The location where the driver became available in the app.
- The time when the driver became available.
- The time when the driver became offline.

3.1 Generation of Random Variates

For this project, you are welcome to use the available random variate generation packages in the programming language you use. For Python random package the following functions can be handy:

```python
import random

# Uniform distribution between a and b
random.uniform(a, b)

# Exponential distribution with mean 1/lambda
random.expovariate(lambd)  # lambd is the rate parameter

# Normal distribution with mean mu and standard deviation sigma
random.gauss(mu, sigma)
```

4 Marking Scheme

You will carry out the project as groups of two and need to sign up for a group on Learn by going to the Groups tab. At this point, we have not covered how the statistical methods we learned can be appropriately used to analyze input and output. Hence, the main emphasis of this project will be on modelling and ideas. The following criteria will play a role in marking:

1. Modelling of the basic structure and assumptions provided by BoxCar using your programming language. You are free to use either the procedural approach or the object-oriented programming template provided.
2. Modification of the models using insights and corrections based on the driver and rider datasets.

3. Analysis of the results and creativity in improvement suggestions.

4. Presentation quality. Do not forget that you are submitting this report to your client/manager to read so it should look professional. Your client does not necessarily try to go through the data and the codes, so ideally everything should be clear from the report without consulting any auxiliary files. A sloppily written report will make you lose your client. You can assume that the reader can understand how discrete-event simulation works and hence, will be able to understand details in your problem description.

4.1 Deliverables

You are expected to submit:

1. A report not exceeding 20 pages which details the model and the insights.

2. Codes for implementation

3. Other auxiliary files (e.g. excel files) can be submitted but not necessary. Remember that ideally everything is contained in the report.