Restaurant Favouritism Interactions in Gig Economy Food Couriers

Student ID: 21197822
Word Count: <1568 (excluding tables, graphs, headers)

Research Question

Within flexibility of the gig economy, many factors are known to influence the behaviours of individual self-employed drivers. Christie & Ward (2019) interviewed and surveyed gig drivers, revealing some of their on-the-job decision-making processes: foremost, gig drivers experience intrinsic self-employment pressure regarding piece rate earnings; this may lead them to working longer hours and working at night and during inclement weather and holidays when there are pay incentives. However, modelling the collective behaviour gig drivers under various pressures may reveal bigger picture insights. Shaheen (2018) employed agent-based models to examine the financial gain of ride-share drivers with different movement behaviours, and Segui-Gasco (2019) simulated autonomous vehicle ride-share services from a traveller's, an operator's, and a city's perspectives.

This study will explore an agent-based model for a gig economy food courier service with a focus on drivers' financial gains. With the idea of gig self-employment pressures in mind, the model will explore the influences of variables including hot spot-memory, driver competition and order frequency.

ODD Description

PURPOSE AND PATTERNS

The purpose of this agent-based model is to predict the dynamics of gig economy food couriers influenced by hot spot-memory, driver competition, and order frequency. The model will illustrate and help understand the impact of the interactions of these variables on the total earnings of drivers and the time segments of delivery trips.

The main pattern in this model is the interaction between favoritism and number of drivers; as these two variables increase, mean earnings of drivers decrease more quickly than if influenced by either of these variables alone.

ENTITIES, STATE VARIABLES, AND SCALES

Entities in this model include three breeds of agents: drivers, orders, and customers. Their state variables trace their wealth and relationships with other agents over time. There are no dynamic observer state variables or dynamic patches state variables.

Table 1: Drivers State Variables

Drivers are agents that find food orders, collect them from restaurants, and bring them to customers. All drivers are the same in their vehicular speed and decision making.

Variable name	Variable type and units	Meaning
dest	Text-string, dynamic	The destination used in navigation, set to: myorder, mycustomer, favorite-restaurant, or nobody
myorder	Text-string, dynamic	Driver's paired order agent; reset to nobody when driver reaches the order
mycustomer	Text-string, dynamic	Driver's paired customer agent; reset to nobody when driver reaches customer
favorite-restaurant	Text-string, dynamic	Remembers the patch with the best-order.
best-order	Integers > 0, dynamic, unit of fictional currency	Remembers the highest money value of any order the driver has received.
money	Integers > 0, dynamic, unit of fictional currency	The driver's bank account; it accumulates payments from orders.
x-stop?	Boolean, dynamic,	Becomes true to signal a driver's arrival near the East-West coordinate of a restaurant or customer-home; immediately

		reset to false upon arrival.
y-stop?	Boolean, dynamic,	Becomes true to signal a driver's arrival near the North-South coordinate of a restaurant or customer-home; immediately reset to false upon arrival.
hasfood?	Boolean, dynamic,	True if a driver has picked up an order from a restaurant and is driving towards the customer.

Table 2: Orders State Variables

Orders are agents that represent a food order transaction that pairs with a driver. Each order comes with money payout to the driver upon delivery to customer.

Variable name	Variable type and units	Meaning
mydriver	Text-string, dynamic	Order's paired driver
Money	Numeric, dynamic, unit of fictional currency, integers > 0	Payment to the driver upon completion of the delivery to the customer. Pseudorandom value dependent on restaurant caliber.

Customers are the agents to whom drivers deliver food orders from restaurants. Customers have no state variables. Improvements to this model may setup customers as patches instead of agents.

Spatial scale

The model uses a geographic space representing a generalized grid-like urban area; it does not represent a real location. Each city-block is eight units square, and roads are one unit across. Drivers may only travel on roads. The roads form a grid of seven-by-seven city-blocks, and the outside of this grid is a continuous city-block two units in thickness bordering the model world. The world does not wrap, which would be geographically unrealistic.

Temporal scale

Drivers travel one spatial unit per time step. Although time steps do not correspond to a specific real length of time, driver trips imitate real-life delivery trips. Simulations run for two-thousand-time steps, an arbitrary time span long enough for responses to stabilise. Similarly, units of money also do not correspond to any real currency. Rather, these units are analysed based on their relative value. Chance-of-order (%) considered in relative terms serves as analogy to real-life slow-times and fast-times, such as holidays and meal-time rushes. Favouritism relates to a driver's knowledge of their delivery pickup territory; if high favouritism is default behaviour for experienced drives, low favouritism may signify the presence of incentives to pick up orders from neglected restaurants.

PROCESS OVERVIEW AND SCHEDULING

Process

For each tick, there are two actions: the observer generates orders, then drivers will drive. There are three condition-based driving modes: roam, pickup, and drop-off. Each corresponds to a leg of a delivery trip.

When a driver does not have an order, it will roam around randomly until it receives an order from a nearby restaurant; it may also choose to roam towards its favorite restaurant, if it has one, in hopes of receiving an order from the favorite restaurant. Once the driver receives an order, it travels to the restaurant with the order to pick-up food. It leaves the restaurant to deliver the food to the customer.

Scheduling

During each tick, there is a chance that one restaurant will generate one order. Upon generation, the order will come with money but no assigned driver. During every tick, any orders without drivers, including newly generated orders, will claim a driver if any are within range.

A driver drives forward only one unit per tick, following only one drive mode (i.e. it will not roam and also drop-off). The drive mode depends on the driver's state variables, and drivers go through the decision-flow every tick. When roaming, drivers with favorite restaurants may decide to move one unit towards their favorite restaurants based on a favoritism probability. Stochasticity in this decision may reflect drivers' time pressure to make decisions, traffic, etc.

When the driver is on a pickup tick, it arrives at the restaurant in the same tick if its final coordinates satisfy the proximity* requirements (y-stop? = true and x-stop? = true) to the restaurant. At the restaurant it trades having an order to having food and changes its color to orange. Its proximity markers reset (to false) and its new destination is its newly received customer location.

Likewise, when the driver is on a drop-off, it may arrive at the customer's location when its coordinates are proximate*. There, it trades food for money, resets proximity markers, and resets its destination to nobody; thence it may resume roaming.

*the model currently requires proximity in either x or y coordinates, (y-stop? = true or x-stop? = true), instead of both, to bypass an unsolved error. As a result, drivers will arrive only partway to their destinations. However, realistically drivers may experience deviations, such as difficulties finding parking and mobile app distractions leading to mis-navigation (Christie & Ward, 2019).

DESIGN CONCEPTS

Basic Principles

This model reflects basic principles of variable interactions and competition (supply and demand) amongst agents.

Emergence

The model output is too simple for significant emergent properties. Visually, drivers congregate more intensely around certain buildings with higher pay-out orders when favoritism increases.

Adaptation

Drivers may roam to restaurants with potentially high pay-outs. This decision depends on stochasticity and memory.

Objectives

Drivers change their favorite restaurants based on their previous order pay-outs. This depends on previous restaurants-visits, i.e. past experiences.

Learning

Drivers learn which restaurants produce the orders with the highest pay-outs by storing past restaurant and order pricing information in their long-term memory. Drivers also hold the location of orders and customers in their short-term memory.

Prediction

Drivers predict from which restaurants the high pay-out orders will emerge when they roam.

Sensing
Orders sense nearby drivers within a certain radius and reach out to pair with them. Drivers sense when they have reached a destination.

Interaction
Interaction occurs when drivers and orders pair based on their proximity. Indirect interaction occurs amongst drivers, who compete for number of orders and high pay-out orders, and amongst orders, who compete for quick pairings with drivers.

Stochasticity
Initialisation of driver locations, restaurant locations, restaurant expensiveness, as well as driver roaming, customer generation, and order generation involve stochasticity. Order money pay-out is a stochastic function of restaurant expensiveness.

Collectives
The model has no collectives.

Observation
Mean earnings of drivers plotted by favoritism as a function of number of drivers reveals an interaction effect between favoritism and number of drivers.

INITIALISATION
The world, 61 square units with a centred origin patch, contains an urban street grid. The urban street grid is initialized by defining *roads* patches. At setup, drivers are generated randomly on the roads, and restaurants are generated randomly anywhere else, i.e. on city blocks. Restaurants generate with a random number of Michelin stars (1-5) at setup. The number of restaurants is initialized at 65. *Driver-range* is set as 15, a radius less than two city block lengths.

All drivers start out roaming without any orders or customers. This means their *myorder* and *favorite-restaurant* are initialized as nobody and their *best-order* and *money* variables are initialized as 0 because drivers have not earned payment by completing any orders yet. Without having visited any restaurants, *hasfood?* is initialized false. Finally, the proximity markers *y-stop?* and *x-stop?* are set as false, as no drivers are en-route to destinations.

INPUT DATA
The model uses no input data.

SUBMODELS

Table 3: Parameters

Variable name	Variable type and units	Meaning
num-drivers	Integer input, static	Number of drivers
favoritism	0-100% slider, static	Favoritism – likelihood of roaming towards favorite restaurant
chance-of-order	0-100% slider, static	Probability an order will spawn per time step.

Table 4: Initialisation and Processes Submodels

Variable name	Variable type	Meaning
roads	Initialisation	Number of drivers
roam	Process	Favoritism – likelihood of roaming towards favorite

		restaurant
nav	Process	Probability an order will spawn per time step.

Methodology

This experiment varies three parameters in Behaviorscape: number of drivers, favoritism, and chance-of-order. This includes eight settings for number of drivers, ranging from 5 to 65 in steps of 10. This parameter is a global input variable in the Netlogo model, while both favoritism and chance-of-order are global slider variables from 1-100%, in increments of 1. In Behaviorscape, the experiment varies favoritism from 0 to 100 in steps of 20, providing eleven settings; with six settings, chance-of-order steps from 10 to 60 in strides of 10. The experiment takes five repetitions of measurements at the end of runs.

Coded as a reporter, *mean-earnings* is the only measured model behaviour. It is equal to the mean of the money of all drivers at time limit 2000, chosen for when the model behaviour measure stabilises.

Results

Figure 1 Mean Earnings vs. Number of Drivers

The major finding** is that mean earnings of drivers decrease linearly as the number of drivers increases. It decreases at a faster rate with higher percent probabilities of favouritism. This interaction suggests that the more strongly that all drivers seek restaurants that produce orders with better payouts, the less money all drivers earn on average. ** (Unfortunately on a final run the fan-shape pointing to the bottom right disappeared, invalidating these conclusions)

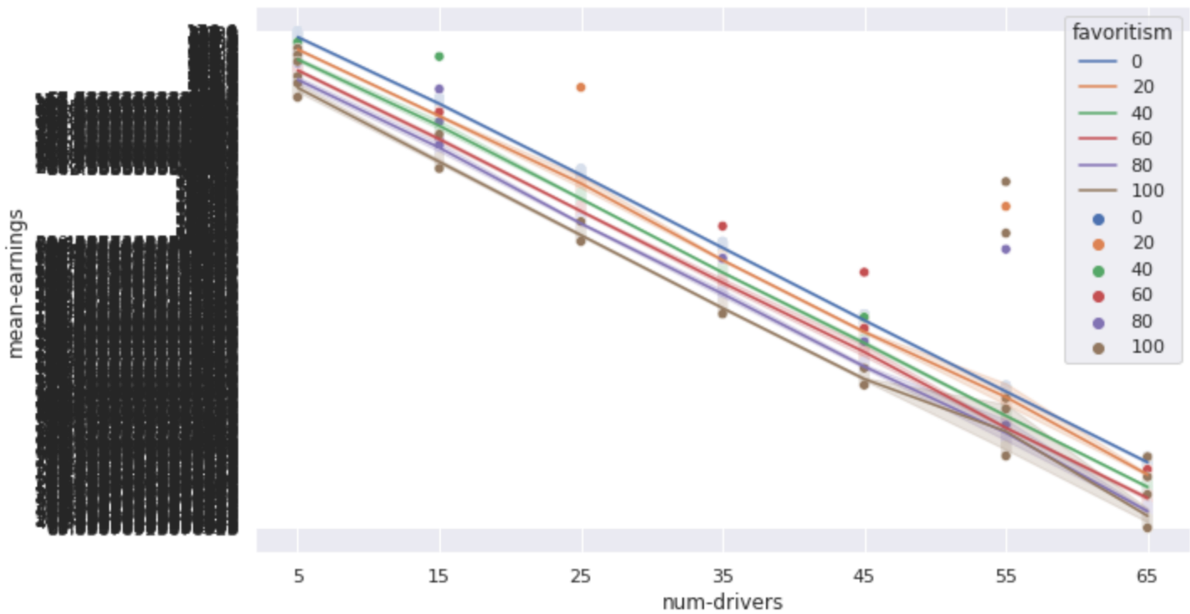


Figure 2 Mean Earnings vs. Chance-of-orders

This figure shows minimal interactions.

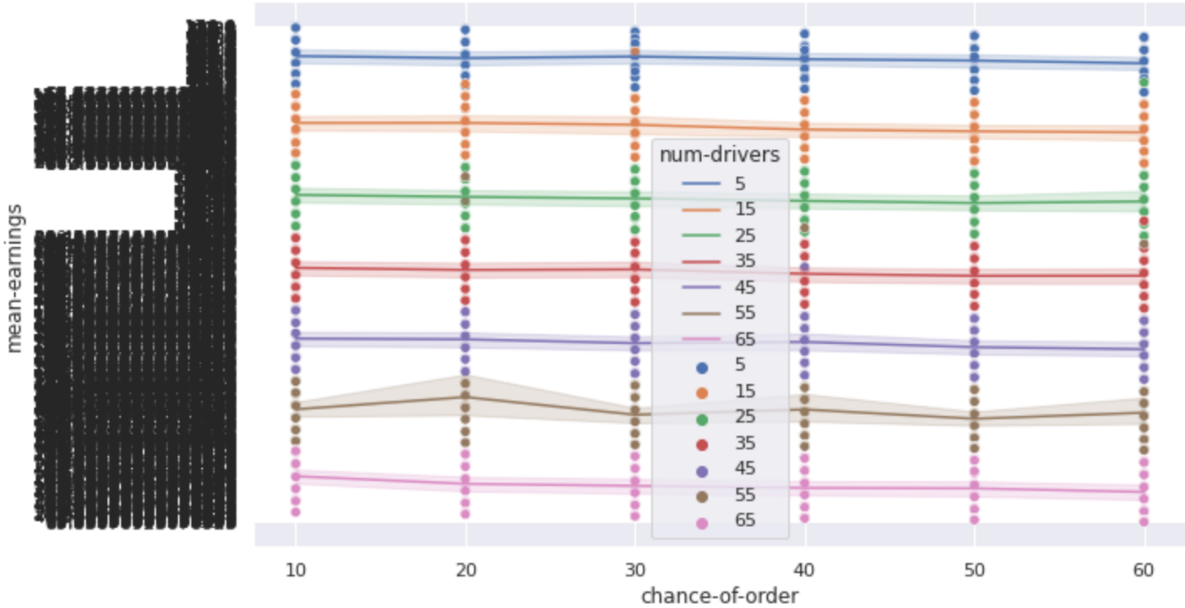
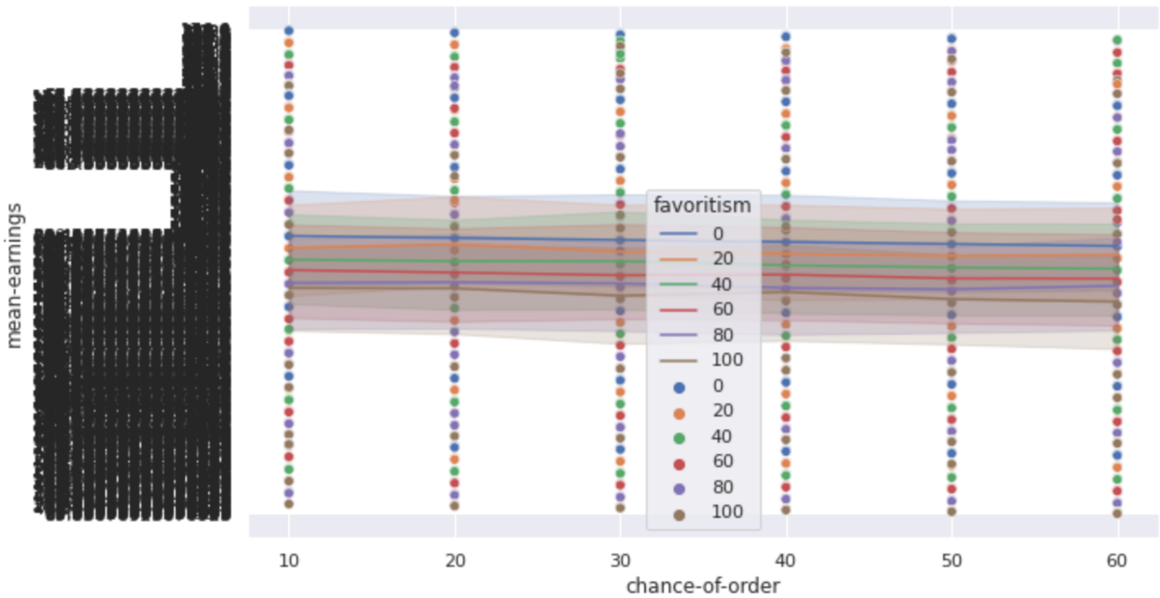


Figure 3 Mean Earnings vs. Favoritism

This figure shows minimal interactions.



Conclusion

As the gig economy increases in popularity, especially during COVID, the number of drivers will likely continue to increase (McCain, 2020). Since humans are creatures of memory and favoritism, the experiment in this study suggests that the increase in inter-driver competition and rise in driver experience with time will decrease average driver earnings. Oversaturated gig economy company platforms may lose drivers who seek other companies with less competition and higher payout. In fact, drivers are pressured to use multiple platform apps (Christie & Ward, 2019). Providing location-based incentives to encourage drivers to spread out in their roaming-searches for orders may be a solution. While this model was based on gig economy food couriers, its findings may be applicable for other types of couriers and gig economy ridesharing as well. This model is simplistic compared to real-life, where drivers may have more diverse categories of favoritism, but its findings apply to cases when independent drives are blind to broader spatial trends.

Bibliography

Chen McCain, S.-L. *et al.* (2021) 'An analysis of a third-party food delivery app during the COVID-19 pandemic', *British Food Journal*, ahead-of-print(ahead-of-print). doi:10.1108/BFJ-03-2021-0332.

Christie, N. and Ward, H. (2019a) 'The health and safety risks for people who drive for work in the gig economy', *Journal of Transport & Health*, 13, pp. 115–127. doi:10.1016/j.jth.2019.02.007.

Grimm, V. *et al.* (2020) 'The ODD Protocol for Describing Agent-Based and Other Simulation Models: A Second Update to Improve Clarity, Replication, and Structural Realism', *Journal of Artificial Societies and Social Simulation*, 23(2), p. 7. doi:10.18564/jasss.4259.

Segui-Gasco, P. *et al.* (2019) 'Simulating a rich ride-share mobility service using agent-based models', *Transportation*, 46(6), pp. 2041–2062. doi:10.1007/s11116-019-10012-y.

Shaheen, J.A.E. (2018) 'Simulating the Ridesharing Economy: The Individual Agent Metro-Washington Area Ridesharing Model', *arXiv:1802.07280 [nlin]* [Preprint]. Available at: http://arxiv.org/abs/1802.07280 (Accessed: 9 May 2022).