Methodology, Modelling and Consulting Skills – Final Project

Group 2 consultancy report

Group members:

Tarun Cox, Saghar Rezanavaz Gheshlagh, Deep Upadhyay, Swyam Salvi

Contents

1. Introduction

In October 2018, a bike-sharing scheme was rolled out in Edinburgh, only to be discontinued in September 2021. With plans to relaunch the scheme, this report aims to advise the Edinburgh city council in finding better locations to build bike-docking stations to make sure users are both engaged and satisfied, and to ensure the project remains within budget. In this report, we will propose a model for determining the locations for new docking stations, advise as to how local businesses should invest in the scheme, and give insight into how this may change over time.

Our model employs a p-centre approach to find the optimal positioning of bike-docking stations around the city. This will provide city-wide coverage of docking stations and ensure that the vast majority of residences, workplaces, and points of interest (POI) in Edinburgh are catered for. We believe that making our bike-sharing scheme visible and accessible to the public will both increase ridership and give people the freedom and flexibility to use the system in a way that suits them.

The original cycle scheme was not a success. The vast majority of bike docking stations in the old system were located in the city centre, with a few stations dotted at key places around the edge of the city. Whilst this might seem logical at first glance, this strategy contains a major flaw: ignoring the suburbs. The majority of residential areas had no bike docks close by, and so the majority of residents would default to other modes of public transport, like buses or trams. Commuters have been shown in previous research to be the primary users of bike-shares [1], and this is also reflected in the data of our old scheme (with the peak of journeys occurring on weekdays around 4-5 pm 'rush hour'). A lot of these commuters will live and work outside the city centre. In cities of over 200,000 people, full city coverage is essential [2]; even if their demand is less, their importance remains high.

Bike-sharing schemes have been rolled out in cities across the world and have, in many cases, been a roaring success. Edinburgh, despite the hills and the weather, definitely has the potential to embrace a cycle-sharing scheme, and we believe that our new system can succeed where its predecessor failed.

2. Demand Forecasting

First, we combined the available trip counts data from 2018 to 2021. After performing basic tasks such as type and range validation, missing value handling, and duplicate removal, we created origin-destination matrices for each hour of the day. As shown in figure 2, trip counts can vary throughout the day, and demands might change each hour.
Therefore, it is important to plan for different levels of demand. To that end, we have used Agglomerative Clustering with connectivity constraints to group similar hours that immediately follow each other. The idea is to derive time blocks that have similar trip patterns. We have extracted four hour-clusters: cluster 0 includes hours 11 to 15, cluster 1 contains hours 16 and 17, cluster 2 contains hours 18 to 23, and cluster 3 contains hours 0 to 10.

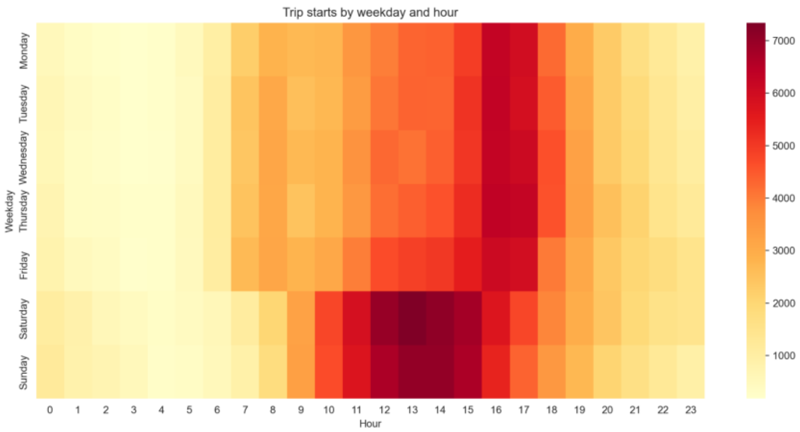

Figure 1: heatmap of demand per hour, for each day of the week

We applied doubly-constrained gravity modeling to predict the number of bike trips during each time block. The gravity model, inspired by Newton's law of gravitation, predicts trip flows between stations based on their production and attraction characteristics while accounting for the impedance of distance. The modeling incorporated an exponential distance decay function (as a negative exponential is common as a deterrence function for cycling), where the parameter beta was optimized through systematic grid search. For each of the four time clusters, we computed station-to-station trip matrices that capture the average daily demand patterns specific to those hours. To establish consistent scaling across our multi-year dataset, we normalized all trip counts by the total number of calendar days and then hours, converting the predicted counts to average hourly values.
Because the existing 85 stations do not fully capture all key locations and potential demand across the city, we needed to generate additional candidate stations based on Points of Interest (POIs). The raw dataset contained 33,669 POIs, of which over 32,000 were residential. Using all of them directly would have made the computations extremely slow and inefficient. To address this, we grouped together POIs of the same category that were located within 100 meters of each other, applying the DBSCAN clustering method. This process reduced the number of POIs to 885, allowing us to work with a much more manageable and representative set of candidate locations.

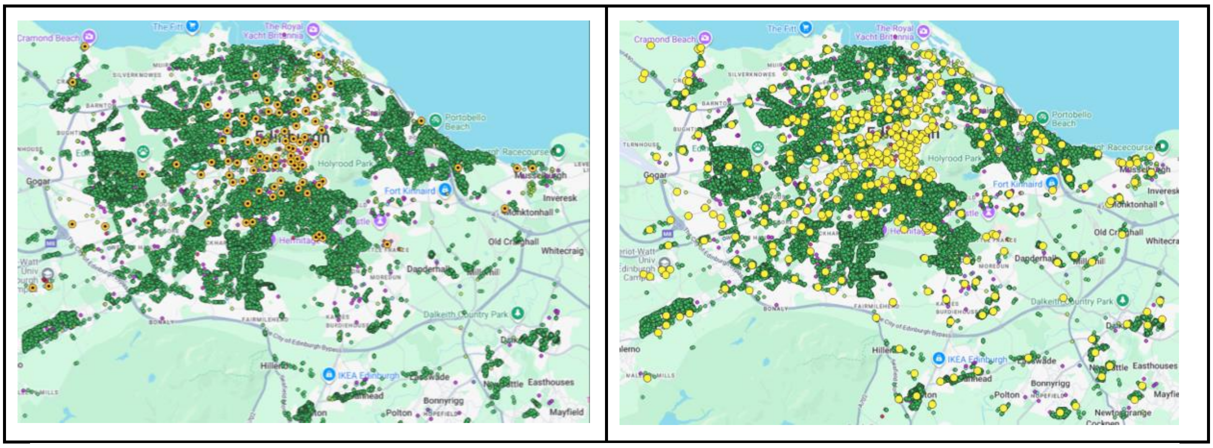

Figure 2: Old stations in orange (left), Candidate stations with predicted weighting in yellow (right), with POIs

However, we did not know the demand at each POI. We came up with a system to allocate the predicted demand at each station to the 3 nearby POIs (within 500 meters). The weight given to each POI is proportional to the importance of the POI category and the inverse of the distance of that POI to the station. By running this code, we could see that no matter how much we increase the proximity radius, we can only assign demand to a maximum of 238 of POIs, which showcases how more than 70% of our POIs have been neglected by the current station locations (with neighbouring number of POIs (k)=3). Even by increasing k from 3 to 10, we end up covering less than half of the POIs. This was also emphasized when we ran the basic pure P-centre model using only the existing stations, and the minimum of maximum distance turned out to be more than 7 kilometers!

We therefore decided to create a baseline demand for POIs that have been neglected by the current station system. The baseline demand is proportional to 10% of the average OD count, the importance of the POI category, and the inverse of distance. In the end, we derived a POI-POI OD matrix for each hour cluster and each category weighting strategy. We used three different weighting strategies:

1. Uniform Weights

All POI categories receive weight = 1. This scenario assumes each POI contributes equally to trip generation and attraction.

2. Data-Driven Predicted Weights

Weights were derived from National Travel Survey (UK) [3] patterns plus forecasting for the year 2026 using an Exponential Triple Smoothing method. The NTS reports trip purposes using categories that do not directly match our POI categories. These percentages represent how people travel overall, but not where they travel. To convert these into POI category weights, we had to make logical assumptions about which types of locations each trip purpose corresponds to.

3. Commercial-Emphasis Weights

Weights heavily favour commercial POIs here. This weighting system allocates 70% of demand to commercial POIs, making our candidate stations more biased towards them and later impacts our docking decisions. We created this category as the cost associated with opening stations near commercial POIs is less due to our sponsor subsidies. We would like to know if opening stations near employers would decrease the costs related to opening and docking stations.

The next stage involves generating a set of potential facility locations. The objective is to produce candidate points that represent meaningful spatial summaries of the underlying demand distribution. For this purpose, weighted K-Means clustering was applied to the full set of POIs, creating clusters whose centroids serve as candidate facility locations, and each POI is weighted by its sum of inflowing and outflowing trips. We went with the highest number of K (close to 400) as this would provide us with more options for stations. Since it would cost less to reopen existing stations, we replaced any candidate station within 40 meters of an existing station with that existing station. In the end, we added all remaining existing stations to our list of candidate stations.

To run our optimisation code, we would also need to have the demand at each POI to complete the dock constraints. The function processes per-hour demand by calculating the sum of inflow and outflow for each POI, providing an estimate of activity levels. For each temporal cluster (0-3) and weighting type (uniform, pred, commercial), it reads the corresponding flow data and computes demand metrics $D_{i,s}$.

3. Model

Our model is based upon a p-centre model: we are minimising the maximum distance between points of interest and their closest bike docking station. Whilst we ultimately want to minimise distance, there are two main goals we want to consider: demand and budget.

We have estimated demands for each POI, and we want to ensure these demands are met. Station capacity is determined by the number of docks and the dock turnover rate, which represents how many times a dock is used (i.e., a bike is taken and later returned) within a day. By multiplying the number of docks by this turnover rate, we approximate the total number of trips a station can support.

Next, we want the project to be within budget. We determine the cost of our proposed project by assigning a cost to building each bike station, depending on whether we are reopening an existing station from the previous scheme (which we assume will be cheaper to reopen, and already have some docks available) or building a new station from scratch. We have assumed that the cost of opening a new station is £90,000, and local businesses will subsidize a certain percentage of the cost for stations. We then add £1,000 per dock in each station. The total cost of all the stations we decide to open must be below our proposed budget. Notice that there is a tradeoff between budget and city coverage: the more stations we build, the more accessible and ubiquitous the scheme is, but the more the project costs.

3.1 Sets:

$I = Set\ of\ all\ points\ of\ interest\ = \{1, \ldots, m\}$
$J = Set\ of\ all\ candidate\ stations\ = \{1, \ldots, n\}$
$S = Set\ of\ all\ scenarios\ (hour\ clusters)\ = \{0,1,2,3\}$
$C = Set\ of\ all\ POI\ categories\ = \{commercial, residential, library, school, university, hospital\}$

$R = \text{Set of possible states for candidate stations } = \{existing, new\}$
$I_c = \text{Set of all POIs of category } c, \forall i \in I, c \in C$
$J_{r,c} = \text{Set of all candidate stations at state } r \text{ with a nearest POI of category } c \;\; \forall j \in J, c \in C, r \in R$

3.2 Decision Variables:

$x_{ij} \in \{0,1\}$ – If POI i is assigned to station j
$y_j \in \{0,1\}$ – If we choose to open/reopen the candidate station j
Z_j – is a non-negative integer – Number of docks in station j

3.3 Parameters:

Q – Maximum distance from a station to a point of interest (POI), when considering all stations and POIs
L – Maximum number of docks a station can have
f_{ij} – Distance between POI i and station j
$D_{i,s}$ – Hourly demand of POI i at the hour cluster s
B – Budget for the project
$E_{c,r}$ – Cost of opening station of type c and state r
$E_{c,r} = E_r \text{ for } c \in C \backslash \{commercial\}$
$E_{c,r} = \alpha E_r \text{ for } c = commercial$
$\text{where } 0 \le \alpha \le 1$
$\text{and } E_{existing} < E_{new}$

A_j – Number of docks already available at station j
K – Cost of adding one dock to a station
τ – Turnover rate for each dock
w_c – Guaranteed service level for POIs of category $c \in C$
G – minimum number of docks per station

3.4 Objective Functions:

We are implementing a bi-objective linear programming problem, with an objective focused on the distance, and the other one on the costs associated with building new stations and docks. We are going to solve this using the ε-constrained method, by considering the second objective as a constraint, and reducing it iteratively to achieve a Pareto front.

$(1) \min Q$ $\qquad$ $(2) \min = \sum_{r\in R} \sum_{c\in C} \sum_{j\epsilon J_{c,r}} E_{c,r} y_j + K \sum_{j\epsilon J} (Z_j - A_j\, y_j)$

Our first objective is to minimise the maximum distance between any POI and its nearest station. Our second objective is to minimize the budget, which consists of the cost of building stations and the cost per dock of those stations.

3.5 Constraints:

$$\sum_{j\epsilon J} x_{ij} \le 1 \qquad \forall i \epsilon I \tag{3}$$

This constraint allows each POI to be assigned to at most one station, or not at all, which lets us meet realistic service levels rather than forcing full coverage.

$$\sum_{r\in R} \sum_{c\in C} \sum_{j\epsilon J_{c,r}} E_{c,r} y_j + K \sum_{j\epsilon J} (Z_j - A_j\, y_j) \le B \tag{4}$$

Considering costs of building stations to stay within budget. We have assumed we do not pay for using already available docks at existing stations.

$$x_{ij} \leq y_j \qquad \forall i \epsilon I, j \epsilon J \qquad (5)$$

We can only assign POIs to station j if we build station j.

$$\sum_{j \epsilon J} f_{ij} x_{ij} \leq Q \qquad \forall i \epsilon I \qquad (6)$$

Q is an upper bound of the distance between all POIs and their assigned stations.

$$A_j y_j \leq Z_j \leq L y_j \qquad \forall j \epsilon J \qquad (7)$$

Number of docks at the station j must be between A_j and L, where A_j = 0 for new stations.

$$\sum_{i \epsilon I} x_{ij} D_{i,s} \leq \tau Z_j \qquad \forall j \epsilon J \qquad \forall s \epsilon S \qquad (8)$$

We must build enough docks to satisfy the demands that are assigned to the station j in each hour scenario s to satisfy demand in the worst case scenario.

$$\frac{\sum_{i \epsilon I_c} x_{ij} D_{i,s}}{\sum_{i \epsilon I_c} D_{i,s}} \geq w_c \qquad \forall j \epsilon J \qquad \forall s \epsilon S \qquad \forall c \epsilon C \qquad (9)$$

This forces the model to serve at least a fraction w_c of the total demand from every POI category c in each scenario s.

$$G y_j \leq Z_j \qquad \forall j \epsilon J \qquad (10)$$

If station j is opened, it must have at least G docks.

4. Analysis

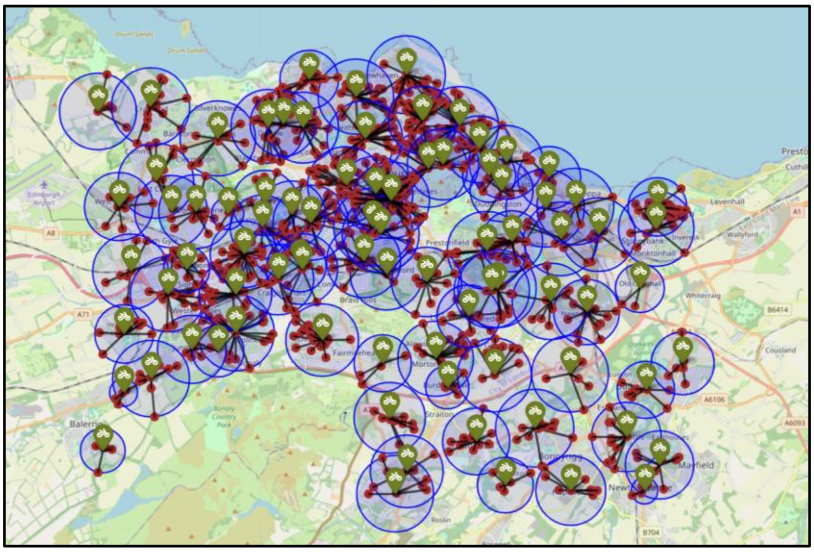

Figure 3: Our proposed locations for docking stations to build, budget = £5,000,000

Figure 3 shows the ideal map of where we would place docking stations around Edinburgh, and what POIs we are assigning to them in order to maximise city-wide coverage with a budget of £5,000,000. The 2018-2021 scheme (shown in Figure 2) had 85 stations, mostly placed in the city centre; this scheme has 91 stations, spread around the city. You will note just how much of the city is covered by the scheme, a stark contrast to the old scheme. Our system fully covers the suburbs in order to give commuters and residents ample access and flexibility to the system.

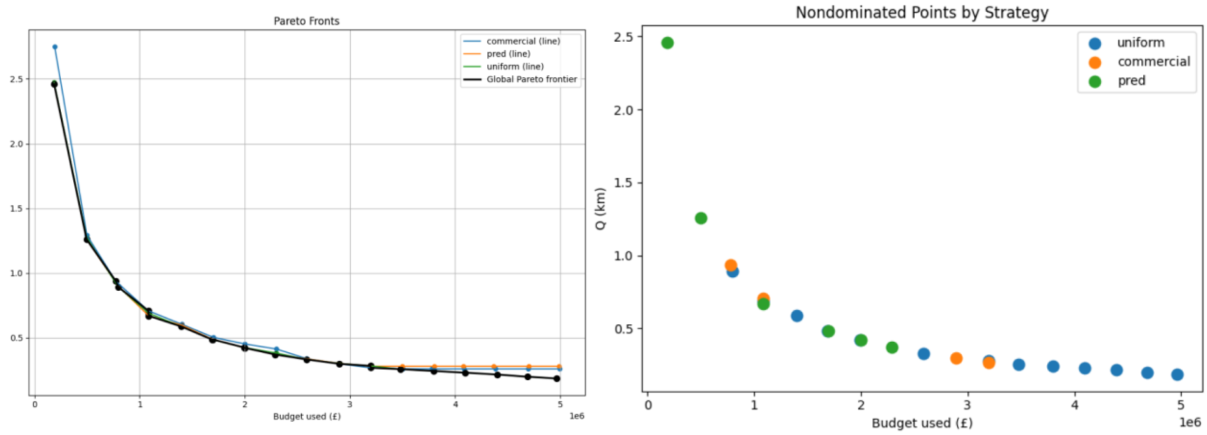

Figure 4: graph of budget against maximum distance (Q), alpha = 0.7

Figure 4 shows the tradeoff between maximum distance (Q) and budget for our 3 weighting strategies, assuming alpha = 0.7, with the pareto optimal frontier shown above.
We can see that the distance decreases rapidly as we begin to increase our budget. It then levels out as we increase our budget further and further. Budget values around £1,500,000 seem reasonable; we get a good tradeoff between budget spent and maximum distance between POIs and stations. If we have more budget, however, It has been shown that distances of over 300m begin to have significant drops in user rates [4], suggesting that a budget in the range of £3-5 million may be a good choice. Ultimately, we can move along the graph until we find a tradeoff between budget and maximum distance that suits our needs.

<u>What happens as we change the level of subsidy of local businesses?</u>
Figure 5 shows how our commercial buy-in level (alpha) affects the optimal distance (Q), and which of the 3 weighting strategies is dominant, for a fixed budget of £1,395,000. For very high values of alpha (where local businesses subsidize docking stations very little), we can see that uniform weighting is optimal. The optimal strategy then becomes predicted weighting for values of alpha between 0.5 and 0.175. Commercial weighting only becomes viable for very high levels of subsidy. This makes sense: the higher the commercial buy-in, the more we “favour” commercial buildings when designing the docking locations. The optimal distance is roughly linear with alpha - once again, this makes sense. The more subsidies we have, the more stations we can build with our fixed budget, and the more of the city we can cover.

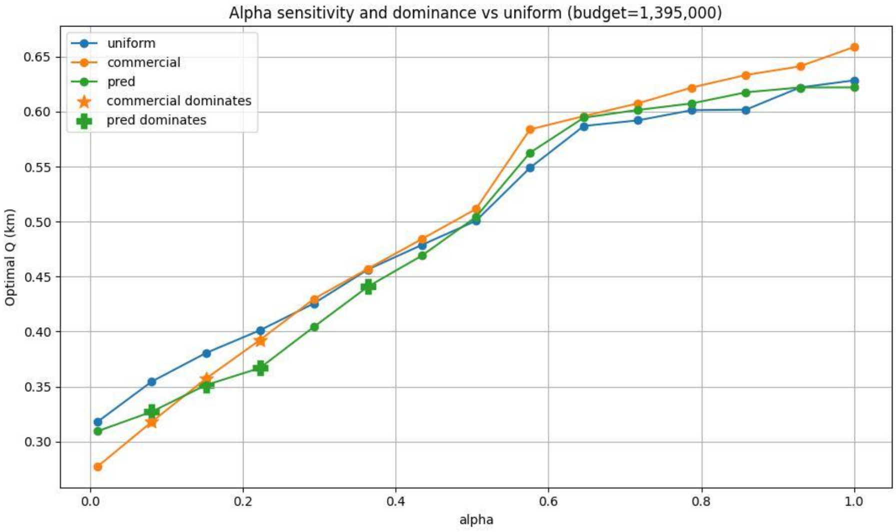

Figure 5: Commercial subsidy against optimal maximum distance (Q), budget = £1,395,000

Note that since our optimal weighting strategy changes, businesses will have incentives to subsidise at certain levels. For example, if the subsidy level rises from 45% to 55%, then we will switch from using uniform weighting to the predicted weighting strategy, and our station locations will begin to favour commercial buildings, which benefits businesses. Therefore, it is reasonable to assume that local businesses will be more inclined to subsidise 55% as opposed to 45%. However, subsidizing above 85% to have the fully commercial weighting strategy might not be worthwhile for businesses. That being said, there are still plenty of incentives for businesses to commit fully to this scheme including: a happier and healthier workforce; meeting environmental targets, investing in the local community and potentially more customer traffic and sales [5]. The exact optimal level of subsidy will depend on the priorities of the local businesses involved and therefore needs to be negotiated.

<u>What happens as we change the turnover rate?</u>
There is no change for values of the turnover rate (tau) in the range of 0.5 - 3.0. The turnover rate is clearly not a variable to focus on.

<u>What happens as we change the service level for non-commercial POIs?</u>
According to figure 6, it can be concluded that by setting non-commercial service levels to a lower value, we can achieve a closer walking distance for commercial POIs (with a fixed commercial service level), without increasing the budget. This means that if the council agrees to a lower service level for non-commercial POIs, it can keep a high service level with a shorter walking distance for commercial POI users, without increasing the budget. Based on the council's priorities, this could also be an option to incentivise businesses to provide a subsidy for our system.

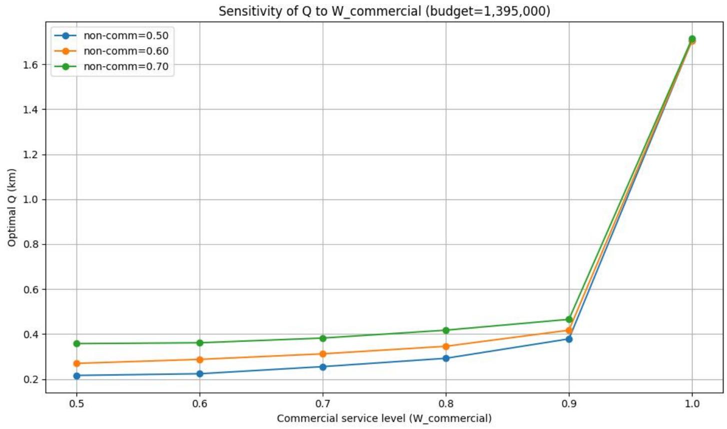

Figure 6: Commercial service level against optimal maximum distance (Q), using different service levels for non-commercial POIs, budget = £1,395,000

<u>How does our system evolve through time?</u>
In the real system, commercial partners contribute only during the deployment stage, when stations are being built, and infrastructure is expanding. Their investment is directed solely toward installing stations, not toward operating or maintaining the system after deployment. Once the target level of service is reached or commercial objectives are satisfied, partner contributions taper off or cease. At that stage, the operator becomes fully responsible for maintenance, operating costs, and long-term system upkeep.

Because the funding structure changes over time, with early investment supporting expansion and later years dominated by maintenance costs, it was necessary to evaluate how the bike-sharing system behaves over the long run. To address this, a 20-year system dynamics model was developed to study the evolution of infrastructure (docks and stations), ridership patterns, profitability and cost sustainability, changes in political support and resulting subsidy levels, and the overall stability or decline of the BSS after initial deployment. The causal loop diagram of this dynamical system can be viewed in Figure 7.

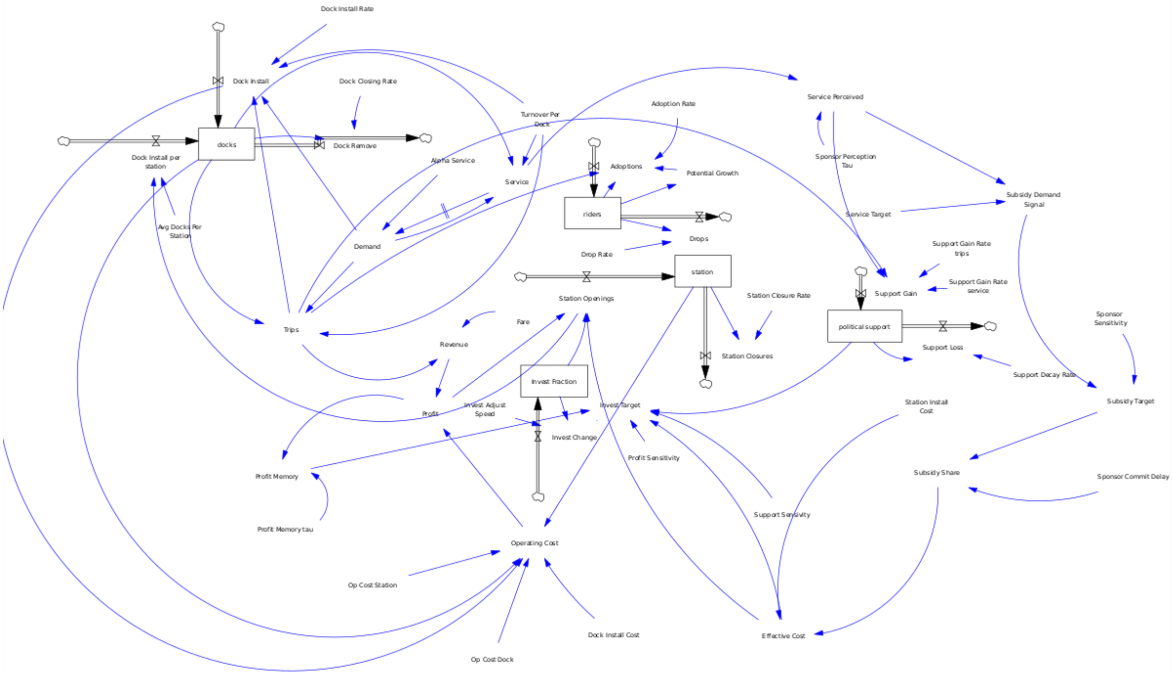

Figure 7: Causal loop diagram of the dynamics of our system

The core stocks include docks, stations, riders, political support, and investment (invest) fraction. They accumulate or decay depending on their inflows and outflows. Each of these flows depends on behavioral, economic, and service-quality variables: for example, infrastructure expands through investment, which itself responds to profitability, political support, and subsidy levels. Profit from completed trips feeds back into investment decisions, forming a long-term economic–operational loop.

A second major feedback loop links political support and commercial subsidies to service performance. Businesses perceive the system's service with delays, respond through adjustments in subsidy targets (to satisfy target service levels), and ultimately influence effective costs through subsidy share. Better service builds political support, which reduces effective costs by increasing investment fraction and encourages expansion; poor service causes decay in support, constraining investment. The model was built this way to reflect real-world delayed perceptions, capacity constraints, economic inertia, and saturation effects. The table below shows a comprehensive list of all stocks, their associated flow variables, and their dependencies.

Stock	Inflow(s)	Outflow(s)	Formula (Stock)	Dependencies (All inputs to inflows/outflows)
riders	Adoptions	Drops	`riders = INTEG(Adoptions - Drops, 2000)`	Potential Growth, Adoption Rate, Trips; Drop Rate

docks	Dock Install, Dock Install per station	Dock Remove	`docks = INTEG(Dock Install + Dock Install per station – Dock Remove, 2200)`	Dock Install Rate, Demand, Trips, Turnover Per Dock; Station Openings, Avg Docks Per Station; Dock Closing Rate
station	Station Openings	Station Closures	`station = INTEG(Station Openings – Station Closures, 85)`	Profit, Invest Fraction, Effective Cost; Station Closure Rate
political support	Support Gain	Support Loss	`political support = INTEG(Support Gain – Support Loss, 0.15)`	Support Gain Rate trips, Trips, Support Gain Rate service, Service Perceived; Support Decay Rate
Invest Fraction	Invest Change	—	`Invest Fraction = INTEG(Invest Change, 0.9)`	Invest Adjust Speed, Invest Target

For the remaining parameters and auxiliaries, refer to the appendix. Running the simulations for a time period of 20 years on Vensim software, we get the results presented in Figure 8.

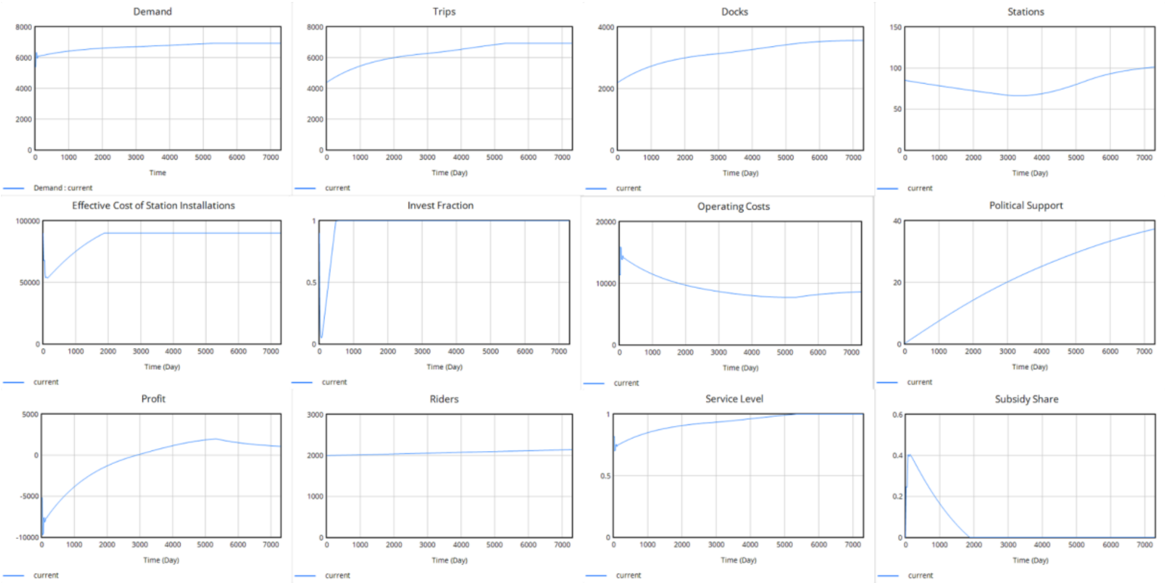

Figure 8: Simulation results of the dynamical system

The simulation reveals a two-phase evolution of the system. During the early deployment period, demand rises faster than supply, pulling service levels below the desired target and triggering commercial partners to increase their subsidy target. This results in a peak subsidy share of 0.4 within the first year, enabling rapid infrastructure deployment when profitability is still negative. However, once the system approaches its service target, businesses immediately begin lowering their subsidy target, causing the actual subsidy share to fall steadily toward zero. After this point, the operator must finance ongoing expansion largely from operating profit. Political support grows continuously through the period as both realised trips and perceived service quality improve, reinforcing the investment environment even as subsidies decline.

As installation costs for new stations are high, the system's expansion strategy naturally shifts after the early phase. While stations remain relatively stable in number, the operator increasingly turns to adding docks to existing stations to satisfy growing demand. This is reflected in the later rise in docks and operating capacity without a corresponding surge in station openings. Profitability increases once early installation costs subside, and trip volumes increase, enabling the operator to maintain high service levels and reinvest efficiently. Overall, the long-run trajectory shows a system that starts subsidy dependent and capacity constrained but transitions into a self-sustaining, high-service network that grows through incremental dock additions.

By performing sensitivity analysis in Vensim, we realised that by increasing the sponsor sensitivity constant by 60%, we can increase the subsidy share peak to 55% of installation costs, which was our proposed solution using the pred weighting strategy. Therefore, by providing enough incentive to our sponsors and based on their desired service level target, we could increase the subsidy share to a favourable level and provide equity for all.

In addition, in the event of a sharp decrease in political support, we could see that the investment fraction (of the system's own profit into growing the system) will very quickly converge to 0, which causes the number of stations and docks to remain constant after the initial increase due to businesses' subsidies. In our simulation and given the parameters, the initial investments by businesses were able to sustain the system at a high service level regardless of the importance of political support, albeit with a near-constant demand and number of riders.

5. Conclusion

In conclusion, the optimal decision depends on your level of investment. We recommend spending anything between £1,000,000 and £5,000,000. The higher the level of investment, the more of the city we will cover.

The reasonable, lowest-budget option of £1,395,000 would, depending on the level of commercial subsidy, yield maximum distance between any point of interest and a docking station would be between 300m (100% commercial subsidy) and 650m (0% commercial subsidy). We would recommend negotiating a subsidy of at least 55%; this is the lowest level of subsidy that will give commercial buildings favourable weighting when choosing where to place docks. This would achieve a maximum distance of around 460m. Higher budget options will provide better city coverage, ensuring that the suburbs are extensively catered for. A budget of £5,000,000 provides exceptional coverage of the entire city, with the maximum distance between any point of interest and its nearest station being under 200m. Therefore, we would not recommend exceeding this budget.

It is also crucial to view this project not as a static installation but as an evolving system. Our system dynamics model demonstrates a clear two-phase lifecycle. During the initial deployment, the system is subsidy-driven and capacity-constrained, relying on commercial investment to build out the initial network rapidly. After the initial build-out, the system transitions to self-sufficiency. As installation costs subside and ridership grows, operating profits fuel further expansion through dock additions to existing stations. Political support grows with service quality, creating a cycle that ensures long-term viability.

In future research, one could introduce subsidy share or commercial service level to the model to directly analyse the impact of each of these variables on our model. Another interesting approach could be to define the service level as the fraction of POIs in each category being served, rather than the fraction of demand served. This approach could theoretically improve equity in the model. Additionally, further demand clusters (considering weekday/weekend or seasonal trends) could be introduced to present a more robust solution.

Bibliography:

[1] Michał Dzięcielski, Alexandros Nikitas, Adam Radzimski, Brian Caulfield,
Understanding the determinants of bike-sharing demand in the context of a medium-sized car-oriented city: The case study of Milton Keynes, UK
Sustainable Cities and Society,
Volume 114,
2024

[2] Juan Carlos García-Palomares, Javier Gutiérrez, Marta Latorre,
Optimizing the location of stations in bike-sharing programs: A GIS approach,
Applied Geography,
Volume 35, Issues 1–2,
2012, Pages 235-246,

[3] Department for Transport. (2025, August 27). *National Travel Survey statistics* (England). GOV.UK.
https://www.gov.uk/government/collections/national-travel-survey-statistics

[4] Kabra, Ashish, Elena Belavina, and Karan Girotra,
Bike-share systems: Accessibility and availability,
Management Science 66.9,
2020: 3803-3824.

[5] Buehler, Ralph, and Andrea Hamre.
Economic benefits of capital bikeshare: A focus on users and businesses
Mid-Atlantic Universities Transportation Center,
No. VT-2013-06,
2014.

Appendix

Variable	Formula	Depends On
Adoption Rate	`0.0001`	—
Adoptions	`MIN(Potential Growth * Adoption Rate , MAX(0, Trips - riders))`	Potential Growth, Adoption Rate, Trips, riders
Alpha Service	`0.5`	—
Avg Docks Per Station	`12`	—

Demand	`DELAY FIXED(5400 * EXP(Alpha Service*(Service - 0.5)), 30, 5400)`	Service, Alpha Service
Dock Closing Rate	`4e-05`	—
Dock Install	`Dock Install Rate * (MAX(0, Demand - Trips) / Turnover Per Dock)`	Dock Install Rate, Demand, Trips, Turnover Per Dock
Dock Install Cost	`10000`	—
Dock Install per station	`Station Openings * Avg Docks Per Station`	Station Openings, Avg Docks Per Station
Dock Install Rate	`0.001`	—
Dock Remove	`Dock Closing Rate * docks`	Dock Closing Rate, docks
Drop Rate	`9e-05`	—
Drops	`Drop Rate * riders`	Drop Rate, riders
Effective Cost	`Station Install Cost * (1 - Subsidy Share)`	Station Install Cost, Subsidy Share
Fare	`1.4`	—
Invest Adjust Speed	`0.1`	—
Invest Change	`Invest Adjust Speed * (Invest Target - Invest Fraction)`	Invest Adjust Speed, Invest Target, Invest Fraction
Invest Target	`MAX(0, MIN(1, political support*Support Sensivity + (Profit Memory/(Effective Cost+1e-09))*Profit Sensitivity))`	political support, Support Sensivity, Profit Memory, Effective Cost, Profit Sensitivity

Op Cost Dock	`1`	—
Op Cost Station	`50`	—
Operating Cost	`Op Cost Dock*docks + Op Cost Station*station + Dock Install*Dock Install Cost`	Op Cost Dock, docks, Op Cost Station, station, Dock Install, Dock Install Cost
Potential Growth	`DELAY FIXED(riders*(1 - riders/600000), 20, 0)`	riders
Profit	`Revenue - Operating Cost`	Revenue, Operating Cost
Profit Memory tau	`20`	—
Profit Memory (in SMOOTHI macro)	`SMOOTHI(Profit, Profit Memory tau, 0)`	Profit, Profit Memory tau
Profit Sensitivity	`1`	—
Revenue	`Fare * Trips`	Fare, Trips
Service	`MIN(1, (docks * Turnover Per Dock) / (Demand + 1e-09))`	docks, Turnover Per Dock, Demand
Service Perceived	`SMOOTHI(Service, Sponsor Perception Tau, Service)`	Service, Sponsor Perception Tau
Service Target	`0.9`	—
Sponsor Commit Delay	`14`	—
Sponsor Perception Tau	`30`	—

Sponsor Sensitivity	`3`	—
Station Closure Rate	`8e-05`	—
Station Closures	`Station Closure Rate * station`	Station Closure Rate, station
Station Install Cost	`90000`	—
Station Openings	`IF THEN ELSE(Profit > 0, Invest Fraction * Profit / (Effective Cost + 1e-09), 0)`	Profit, Invest Fraction, Effective Cost
Subsidy Demand Signal	`(MAX(Service Target - Service Perceived, 0))/(Service Target + 1e-09)`	Service Target, Service Perceived
Subsidy Share	`DELAY FIXED(Subsidy Target, Sponsor Commit Delay, 0)`	Subsidy Target, Sponsor Commit Delay
Subsidy Target	`MAX(0, MIN(1, 1 - EXP(-Sponsor Sensitivity * Subsidy Demand Signal)))`	Sponsor Sensitivity, Subsidy Demand Signal
Support Decay Rate	`0.0002`	—
Support Gain	`Support Gain Rate trips * Trips/(Trips + Trips/2) + Support Gain Rate service * Service Perceived`	Support Gain Rate trips, Trips, Support Gain Rate service, Service Perceived
Support Gain Rate service	`0.01`	—
Support Gain Rate trips	`0.0002`	—
Support Loss	`Support Decay Rate * political support`	Support Decay Rate, political support

Support Sensitivity	`0.3`	—
Trips	`MIN(Demand, docks * Turnover Per Dock)`	Demand, docks, Turnover Per Dock
Turnover Per Dock	`2`	—