Pontificia Universidad Javeriana

Facultad Ciencias Básicas

Integration of Optimization and Prediction Models for Bicycle Sharing Systems

Author:
Juliana De Mier Medellin

Supervisor:
Camilo Ortiz y Claudio Contardo

Agosto 28, 2023

Integration of Optimization and Prediction Models for Bicycle Sharing Systems

Juliana De Mier Medellin

Nota: Aprobado

Director:

Camilo Ortiz Ph. D. ______________________

Codirector:

Claudio Contardo Ph. D. ______________________

Jurados:

Lina Maria Acosta Avena Ph. D. Lina Maria Acosta Avena

Jaime Esteban Gonzalez Ph. D. ______________________

Integration of Optimization and Prediction Models for Bicycle Sharing Systems

Juliana De Mier Medellin

Aprobado:

Jhon Jairo Sutachan Rubio, PhD.
Director de Posgrado
Facultad de Ciencias

Alba Alicia Trespalacios Rangel, PhD.
Decana
Facultad de Ciencias

Bogotá, Agosto de 2023

Contents

Abstract

Shared bicycle systems have become increasingly popular in recent years as a sustainable and cost-effective mode of transportation. However, efficiently managing these systems requires accurate demand prediction, optimal station placement, and effective rebalancing strategies. To address these challenges, we propose a comprehensive framework integrating combinatorial optimization and machine learning techniques.

The framework consists of a feedback loop composed of two parts: a machine learning model to predict demand patterns and an optimization model to determine station locations with rebalancing to reallocate bicycles across stations, ensuring users have access to them when and where needed.

To evaluate the framework, extensive computational experiments were conducted using real data from Tembici, a shared bicycle system in Bogota. Multiple instances were created, and the results demonstrate that the proposed system significantly enhances the performance of shared bicycle systems. These improvements lead to reduced operational costs, higher user satisfaction, and increased sustainability, making a valuable contribution to the field of bikesharing system optimization and demand analysis. Overall, the proposed framework provides a powerful and efficient tool for managing shared bicycle systems, with practical applications in improving the performance of such systems.

Chapter 1

Introduction

Combinatorial optimization and machine learning are two important fields of mathematics with diverse applications. In general terms, combinatorial optimization deals with finding the best possible solution from a finite set of options, while machine learning is concerned with developing algorithms that can learn from data and make predictions based on that learning. While these two fields have different goals, they both entail solving complex mathematical problems efficiently.

We aim to analyze the feedback loop of shared bicycle systems by integrating two models: one for optimization and one for prediction. This will allow us to accurately predict demand, place stations in optimal locations, and use effective rebalancing strategies. This will improve the performance and sustainability of shared bicycle systems, making them a more attractive alternative to traditional public transportation and driving in urban areas.

Recently, there has been increasing interest in the intersection of machine learning and combinatorial optimization [17,23]. Researchers have recognized the potential for combining these two fields to create powerful and efficient solutions to complex problems. Machine learning can be used to predict demand and optimize station placement, while combinatorial optimization can be used to optimize rebalancing strategies. By integrating this two models: one for optimization and one for prediction, we acomplish that there are always enough bicycles available when and where they are needed, improving the performance of the shared bicycle system, creating a system that is more efficient, sustainable, and attractive to users.

We now consider one specific application where the integration of combinatorial optimization and machine learning has demonstrated its value: shared bicycle systems. These systems have gained popularity as sustainable and cost-effective modes of transportation in urban areas worldwide[23]. They allow users to rent bicycles from various stations located throughout a city, providing a convenient and eco-friendly alternative to driving or taking public transportation. The successful implementation of these systems has been attributed to accurately predicting demand, efficiently managing station locations, and balancing the distribution of bicycles across the network through the use of combinatorial optimization techniques and machine learning algorithms[27]. Various studies and real-world examples have shown how this integration optimizes operational efficiency, enhances user experience, and promotes eco-friendly urban mobility[33].

Accurately predicting demand is essential for efficient management of a shared bicycle system, as it allows operators to optimize the distribution of bicycles and stations, leading to improved system performance and user satisfaction. However, demand prediction is a complex task that requires consideration of a wide range of factors, including weather conditions, user behavior, and urban mobility patterns. One way to improve demand prediction is to use machine learning algorithms to analyze historical data and identify patterns in bicycle usage[27].

Determining the optimal locations for stations is crucial for the success of a shared bicycle system. Station location models typically consider factors such as population density, proximity to public transportation, and land use patterns. However, these models often do not take into account the predicted demand for bicycles, which can lead to suboptimal station locations and inefficient system performance[23]. To address this challenge, recent research has focused on developing endogenous models that incorporate demand predictions directly into the station location problem[27].

Efficient rebalancing of bicycles across the network is another key challenge in shared bicycle systems. Rebalancing refers to the process of moving bicycles from stations with excess capacity to those with high demand, which is necessary to ensure that users have access to bicycles when and where they need them. Rebalancing can be a costly and complex process, and requires consideration of a range of factors, including the number of bicycles that need to be moved, the distance between stations, and the cost of moving bicycles. Combinatorial optimization algorithms can be used to develop efficient rebalancing strategies that minimize the cost of moving bicycles while ensuring that users have access to bicycles when and where they need them.

The integration of optimization techniques with demand analysis has emerged as a significant area of research in the context of bikesharing systems. This work aims to address the gap in the literature by exploring the application of optimization methods that incorporate demand factors in bikesharing systems. By considering demand prediction, location models, and rebalancing strategies, this study seeks to optimize the station locations and improve the operational efficiency of bikesharing systems[27].

This research develops a framework that integrates optimization techniques with demand analysis in order to analyze the feedback loop and to enhance the performance of bikesharing systems. The framework first uses demand prediction methods to accurately estimate future demand patterns. This information is then used to formulate optimal station locations. The framework also explores various optimization approaches to address the interconnected challenges of station placement and rebalancing within bikesharing systems. By treating these challenges as a unified optimization problem, the framework aims to create an efficient solution that ensures balanced supply and demand across the system. The framework is designed to be a feedback loop, with the results of the optimization process being used to improve the demand prediction model. This iterative process allows the framework to continuously learn and adapt to changes in demand, ensuring that the system remains optimized over time.

Mathematics plays a fundamental role in optimizing shared bicycle systems through the integration of combinatorial optimization and machine learning. It provides the essential framework for modeling complex urban environments, enabling the identification of optimal station locations and efficient rebalancing strategies. Machine learning, a mathematical discipline, facilitates accurate demand prediction by extracting patterns from historical data. This integrated approach forms an iterative feedback loop, continuously adapting to evolving urban dynamics to enhance system efficiency and sustainability, ultimately promoting environmentally friendly urban transportation solutions.

Our contribution is twofold: First, we develop and implement a feedback loop system integrating an optimization model with a demand prediction model. Second, we conduct extensive computational experiments to analyze the integrated system behavior and evaluate its effectiveness in enhancing bikesharing systems efficiency and user experience.

By providing a clear outline of our approach, we aim to make a valuable contribution to the field of bikesharing system optimization and demand analysis, contributing to sustainable urban transportation solutions [6], [10].

The subsequent chapters of this work are organized as follows: Chapter 2 (***Preliminary Concepts***) presents the necessary background, including mixed-integer programming (MIP), location models, and prediction methods such as neural networks and linear regression. Chapter 3 (***Problem Description and literature review***) describes the problem in detail, covering demand prediction and various optimization approaches for bikesharing systems. It discusses deterministic and stochastic approaches, hub location models, maximal covering problems, and the repositioning problem. Moreover, it introduces the concept of a rebalancing model and highlights the importance of considering demand as an endogenous factor. Chapter 4(***Integrated Framework***) formulates the problem by integrating optimization, rebalancing, and demand factors into a comprehensive framework and focuses on the computational aspects, discussing data collection, experimental setup, results, and variations in the findings.

Chapter 2

Preliminaries

In this chapter we provide the most relevant definitions and results from mixed integer linear programming (MILP), locations models and neural networks, that are required as foundation for the rest of the document. The results presented in this chapter are extracted from the books of [1, 16, 35].

2.1 Mixed Integer Programming

Combinatorial optimization involves finding optimal solutions for an objective function within a discrete configuration space. This configuration space is characterized by a finite set of options or variables rather than a continuous space. MILP deals with optimization problems involving both continuous and discrete variables. In MILP, some variables are allowed to take on integer values, which introduces combinatorial aspects to the problem.

Definition 2.1. Consider a finite set $N = \{1, \dots, n\}$ with corresponding weights c_j for each $j \in N$ and a set F of feasible subsets of N. The problem of finding a minimum weight feasible subset is the combinatorial optimization problem

$$\min \left\{ \sum_{j \in S} c_j \mid S \in F \right\}$$

Definition 2.2. A linear (MIP) is a mathematical optimization problem in which the objective function and constraints are linear, but some of the variables are restricted to be integers. A MIP problem can be written in the following standard form:

$$\begin{array}{ll} \min & c^T x \\ \text{subject to} & Ax \leq b \\ & x \in \mathbb{Z}^n \end{array}$$

where c is a vector of coefficients, x is a vector of variables, A is a matrix of coefficients, b is a vector of constants, and $\mathbb{Z}^n$ is the set of all n-dimensional integer vectors.

Definition 2.3. Feasible solution
The set $S = \{x \in \mathbb{Z}, y \in \mathbb{R} \mid Ax + Gy \leq b\}$ is called the feasible region, and a point $(x, y) \in S$ is called a feasible solution.

Definition 2.4. Let an MILP model be denoted as P and let its set of feasible solutions be denoted as FS(P). A set of subproblems of P is defined as a separation of P if the following conditions hold:

1. a feasible solution of any of the subproblems is a feasible solution for P
2. every feasible solution of P is a feasible solution of exactly one of the subproblems

A separation of P is useful because it allows us to solve P by solving the subproblems one at a time.

Definition 2.5. Let P and RP be optimization problems. Then RP is said to be a relaxation of P if the following conditions hold:

- The set of feasible solutions of is a subset of the set of feasible solutions of P.
- The objective function of RP is less than or equal to the objective function of P.

Proposition 2.6. *Let P and RP be optimization problems, and let z_P and z_{RP} be their respective optimal solutions. Then the following hold:*

- *If RP is infeasible, then P is also infeasible.*
- *If z_P is feasible, then $z_P \leq z_{RP}$.*
- *If an optimal solution of RP is feasible for problem P, then it is an optimal solution of P.*

One of the most well-known used techniques for solving MIPs is the branch-and-bound algorithm the following 2.1 is the schema given by [25]:

Algorithm 1: Branch-and-Bound(X, f)

Set $L = \{X\}$ and initialize $\hat{x}$
while $L \neq \emptyset$**:**
    Select a subproblem S from L to explore
    if a solution $\hat{x}' \in \{x \in S \mid f(x) < f(\hat{x})\}$ can be found**:** Set $\hat{x} = \hat{x}'$
    if S cannot be pruned**:**
        Partition S into $S_1, S_2, \ldots, S_r$
        Insert $S_1, S_2, \ldots, S_r$ into L
    Remove S from L
Return $\hat{x}$

Figure 2.1: Branch and bound Schema

Branch and Bound Algorithm:

Initialization:

Begin with a list of subproblems, with P as the only element and set initial best integer solution to $-\infty$.

Subproblem Selection and Relaxation:

Choose a subproblem from the list, solve linear programming relaxation P_{LP} of selected subproblem. If P_{LP} is infeasible, prune subproblem or If P_{LP} is unbounded, prune subproblem.

Integer Feasible Solution Check: If P_{LP} has integer feasible solution, compute f_{LP} or if f_{LP} is less than best integer solution, update best integer solution.

Subproblem Pruning: If f_{LP} is greater than or equal to best integer solution, prune subproblem.

Fractional Variable Selection and Subproblem Generation: If P_{LP} lacks integer feasible solution, select fractional variable x_i, then create two new subproblems by adding constraints to restrict x_i: - Add $x_i \leq \lfloor x_i \rfloor$ to one subproblem. - Add $x_i \geq \lceil x_i \rceil$ to other subproblem.

Subproblem Expansion and Iteration: Add new subproblems to list.Return to step 2, repeating until list is empty.

Terminate when all subproblems are pruned or optimal integer solution is found.

Complexity: Exponential time complexity with problem's size.

Correctness: Complete algorithm that finds optimal integer solution if it exists.

Soundness: Sound algorithm that never reports suboptimal solution.

Definition 2.7 ($\mathcal{P}$ class)**.** The class P consists of those problems that are solvable in polynomial time, i.e. these problems can be solved in time $O(nk)$ in worst-case, where k is constant.
An algorithm is a polynomial time algorithm, if there exists a polynomial $p(n)$ such that the algorithm can solve any instance of size n in a time $O(p(n))$.

Definition 2.8 ($\mathcal{NP}$ class)**.** The class NP consists of those problems that are verifiable in polynomial time. NP is the class of decision problems for which it is easy to check the correctness of a claimed answer, with the aid of a little extra information.

Definition 2.9 ($\mathcal{NP}$ -hard)**.** A problem is NP-hard if all problems in NP are polynomial time reducible to it, even though it may not be in NP itself.

Definition 2.10 ($\mathcal{NP}$ -complete)**.** If a polynomial time algorithm exists for any of these problems, all problems in NP would be polynomial time solvable. These problems are called NP-complete.

2.2 Location Models

Location problems are a class of optimization problems that involve determining the optimal location or locations for a facility or a set of facilities in a given geographical region, in order to minimize or maximize some objective function. In this chapter, we discuss some of the most common problems and their formulations. The results presented in this chapter are extracted from the books [2,9,15,21]

The general formulation of covering models proposed is :

$$\text{minimize} \quad \sum_{i=1}^{n} f_i x_i + \sum_{j=1}^{m} c_j y_j \tag{2.1}$$

$$\text{subject to} \quad \sum_{i=1}^{n} x_i \geq d_j, \quad \forall j \in \{1, 2, \ldots, m\} \tag{2.2}$$

$$\sum_{j=1}^{m} y_j \leq k \tag{2.3}$$

$$x_i \in \{0, 1\}, \quad \forall i \in \{1, 2, \ldots, n\} \tag{2.4}$$

$$y_j \in \{0, 1\}, \quad \forall j \in \{1, 2, \ldots, m\} \tag{2.5}$$

The objective function (2.1) consists of two parts. The first part calculates the total fixed cost of opening facilities at each site, while the second part measures the total cost or profit provided by certain variables. Constraint (2.2) limits the number of centers that can be opened. Constraints (2.3) ensure that each demand point is covered by a minimum number of facilities, with additional costs incurred for exceeding that minimum. Constraints (2.4) enforce integer values for the number of centers at each site, and constraints (2.5) require binary values for certain variables. This model aims to cover all demand points with the minimum required facilities while minimizing costs and considering additional costs for over-coverage. Different models can be obtained by assigning specific values to the constants in this formulation. We present three different formulations: set covering problem, maximal covering problem, and p-median. After comparing these formulations, we procede to choose the most suitable one.

2.2.1 Set Covering Problem

The Set Covering Problem (SCP) is a well-known optimization problem that has been extensively studied. It is defined as follows:

Given a set of demand points J and a set of possible centers I, the SCP seeks to find the minimum number of centers that must be opened in order to cover all of the demand points. This can be formulated as a linear programming problem as follows:

$$\begin{aligned} \min \sum_{i \in I} y_i \\ \text{subject to} \sum_{i \in I} a_{ij} y_i \geq 1 \quad \forall j \in J \\ y_i \in \{0,1\} \quad \forall i \in I \end{aligned}$$

where:

- a_{ij} is a binary variable that is 1 if center i covers demand point j and 0 otherwise
- y_i is a binary variable that is 1 if center i is open and 0 otherwise

The SCP is NP-complete, but the linear relaxation of the problem often provides an integer solution. This means that the optimal solution to the linear relaxation is also an optimal solution to the original SCP. However, there are also instances of the SCP where the linear relaxation does not provide an integer solution. In these cases, other algorithms, such as branch-and-bound, can be used to find the optimal solution.

The SCP has been applied to a variety of real-world problems, including the location of emergency service centers, the selection of products to be included in a product line and the design of fault-tolerant systems. The SCP is a challenging problem to solve, but it is a valuable tool for modeling and solving a wide range of real-world problems.

2.2.2 Maximal covering problem

The Maximal (or Maximum) Covering Location Problem (MCLP) was introduced by Church and ReVelle (1974). In contrast to previous models, the goal of the MCLP is to maximize the covered demand, rather than requiring the coverage of all demand points. In this problem, the number of facilities to be located is limited to a given value $p < m$.

The formulation of the MCLP can be described as follows:

- $h = p$ (number of facilities to be located)
- $b_j = 0$ for all $j \in J$ (no requirement to cover all demand points)
- $e_i = 1$ for all $i \in I$ (opening multiple facilities at the same site is not optimal)
- a_{ij} values are defined as usual
- $f_i = 0$ for all $i \in I$ (variables y_i do not contribute to the objective function)
- $g_{jk} = 0$ for all $j \in J$, $k \geq 2$ (variables w_{jk} with $k \neq 1$ do not contribute to the objective function)
- $g_{j1} = -1$ for all $j \in J$ (maximize the number of demand points covered by the open facilities)

An alternative approach proposed by Church and ReVelle (1974) combines mandatory covering of some demand points (indexed by $J_1 \subseteq J$) and maximization of the coverage of the remaining points ($J \setminus J_1$). This situation can also be formulated using the model (COV) by setting:

- $h = p$
- $b_j = 1$ for all $j \in J_1$
- $b_j = 0$ for all $j \in J \setminus J_1$
- $e_i = 1$ for all $i \in I$

- $f_i = 0$ for all $i \in I$
- $g_{j1} = -1$ for all $j \in J \setminus J_1$
- $g_{jk} = 0$ for all $j \in J \setminus J_1$, $k \geq 2$
- $g_{jk} = 0$ for all $j \in J_1$, $k \in K$

The classical formulation of the MCLP can be represented as follows:

$$\text{(MCLP) Maximize } \sum_{j \in J} g_{j1}$$
$$\text{subject to } \sum_{i \in I} a_{ij} x_i \geq b_j \quad \forall j \in J$$
$$x_i \in \{0, 1\} \quad \forall i \in I$$

This formulation aims to maximize the number of covered demand points by selecting the optimal locations for a limited number of facilities.

2.2.3 The p-median problem

The p-Median Problem (pMP) can be described as follows. Given a set of n demand points, the goal is to choose p of them as facility locations (called medians) and allocate each demand point to one of these facilities in such a way that the total cost is minimized. The cost of allocating demand point j to facility i is the distance d_{ij} between the two points, assuming $d_{ii} = 0$ for all i and $d_{ij} > 0$ for all other cases.

The classical formulation of the p-Median Problem can be represented as follows:

$$
\begin{aligned}
\text{(pMP) Minimize} \quad & \sum_{j=1}^{n}\sum_{i=1}^{n} d_{ij}x_{ij} \\
\text{subject to} \quad & \sum_{i=1}^{n} x_{ij} = p \\
& \sum_{i=1}^{n} x_{ij} \leq 1 \quad \forall j = 1, \ldots, n \\
& \sum_{j=1}^{n} x_{ij} \leq p \quad \forall i = 1, \ldots, n \\
& x_{ij} \in \{0, 1\} \quad \forall i, j = 1, \ldots, n
\end{aligned}
$$

The first constraint ensures that exactly p of the demand points are chosen as facilities. This is because we want to have a total of p service centers. The second and third constraints ensure that each demand point is assigned to exactly one facility and that each facility is assigned at most p demand points. This is because we want each customer to be assigned to exactly one service center and we want each service center to be assigned at most p customers. The last constraint ensures that the decision variables $x_{i,j}$ are binary. This is because we can only have one facility assigned to each demand point and one demand point assigned to each facility.

In addition to the p-Median Problem, there are two other types of approaches that can be used for bike-sharing system station location: deterministic and stochastic approaches, and hub location models. Deterministic approaches assume that all the parameters of the model are known with certainty, while stochastic approaches take into account uncertainties by modeling them as random variables. Hub location models view BSS stations as hubs in a hub-and-spoke system. Hub location models can provide insights into the modeling of BSS station location decisions, as they can be used to minimize the total cost of assigning stations to demand points. However, they may not be able to capture all of the complexities of bike-sharing system station location, such as the demand patterns and the system performance. For this proyect we considered that these approaches would not add much to the discussion. For more information of how to apply this approaches to a BSS can be find in [3, 12, 18]

2.2.4 Repositioning Problem

The dynamic repositioning problem addresses the redistribution of bicycles across stations to balance the system's inventory. Mathematical formulations for this problem often incorporate factors such as target inventory levels, penalty functions, and time constraints. Repositioning problem formulation:

$$
\begin{aligned}
\text{Minimize} \quad & \sum_{i \in V} c_i x_i + \sum_{t=1}^{T} \sum_{i,j \in V} d_{ij} r_{ijt} \\
\text{Subject to} \quad & \sum_{i \in V} x_i = K \\
& \sum_{j \in V} r_{ijt} \leq x_i \quad \forall i \in V, t \in \{1, \ldots, T\} \\
& \sum_{i \in V} r_{ijt} \geq 1 - x_i \quad \forall i \in V, t \in \{1, \ldots, T\} \\
& x_i, r_{ijt} \in \{0, 1\} \quad \forall i \in V, j \in V, t \in \{1, \ldots, T\}
\end{aligned}
$$

where c_i is the cost of opening station i, d_{ij} is the distance between station i and station j, x_i is a binary decision variable indicating whether station i is opened, r_{ijt} is a binary decision variable indicating whether one bicycle is relocated from station i to station j at time t, K is the number of stations to be opened, T is the number of time periods.

The objective function minimizes the total cost of opening stations and relocating bicycles. The first constraint ensures that exactly K stations are opened. The second and third constraints ensure that the number of bicycles relocated from station i to station j at time t does not exceed the number of bicycles available at station i and does not fall below the number of bicycles needed at station j. The last constraint ensures that x_i and r_{ijt} are binary variables.

This model is a generalization of the p-Median Problem, as it allows for the relocation of bicycles between stations. This can be useful for improving the efficiency of bike-sharing systems, as it can help to ensure that there are always enough bicycles available when and where they are needed.[4,14,30]

In certain situations, the choice of optimization problem formulation for bike station selection depends on specific requirements. The Set Covering Problem minimizes costs while ensuring all demand points are covered. The Maximal Covering Problem maximizes coverage within a budget constraint. The P-median Problem balances coverage and cost.

The Maximal Covering Problem is a compelling choice for bike station selection. It prioritizes coverage, promoting accessibility and higher ridership. This leads to reduced reliance on other transportation modes and improved commuting efficiency. While cost optimization is important, the emphasis on coverage aligns well with maximizing the benefits and convenience of bike stations for a broader population.

2.3 Prediction methods

In this section, we explore three methods for predicting demand.

2.3.1 Neural Networks

Neural networks are a class of machine learning algorithms inspired by the structure and function of the brain. Neural networks consist of interconnected nodes (neurons) that process and transmit information. They are capable of learning from examples and can be used for a variety of tasks, such as classification, regression, and clustering.Figure 2.2 shows the architecture of a neural network. The results presented in this section are extracted from [5,26,36]

Definition 2.11 (neurons as Functionals)**.** Input values $x_1, x_2, \ldots, x_n \in \mathbb{R}$ along with corresponding weight values $w_1, w_2, \ldots, w_n \in \mathbb{R}$. In addition, each neuron (of a hidden and output layer) has its own bias value $b \in \mathbb{R}$.
If we consider the inputs $x := (x_1, x_2, \ldots, x_n)^\top$ as well as the corresponding weights $w := (w_1, w_2, \ldots, w_n)^\top$ as two vectors of a n-dimensional vector space

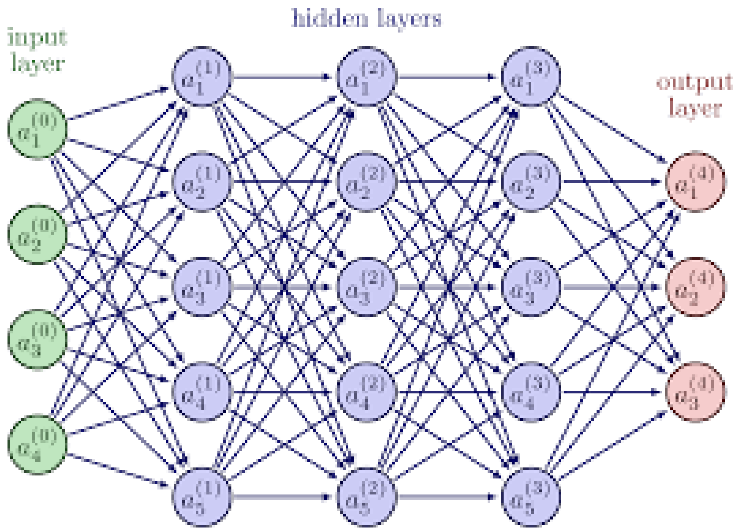

Figure 2.2: Image from [26]

$\mathbb{R}^n$, and $b \in \mathbb{R}$ as a scalar (i.e. a real value), we can represent this single neuron as a functional $\psi : \mathbb{R}^n \to \mathbb{R}$, defined by [25]

$$\begin{aligned}\psi(x) &:= \sigma\left(\sum_{i=1}^{n} x_i w_i - b\right) \\ &= \sigma(\underbrace{x^\top \bullet w - b}_{\in\mathbb{R}}),\end{aligned}$$

where σ is an activation function. The weighted sum can also be denoted as dot / inner product of the two vectors, that is, $\sum_{i=1}^{n} x_i w_i = x^\top \bullet w \in \mathbb{R}$.

Definition 2.12 (ReLU function)**.** The Rectified Linear Unit (ReLU) function maps any negative input to 0 and any positive input to itself. It is often used as an activation function in the hidden layers of a neural network. The exact definition is as follows.

$$x \mapsto \mathrm{ReLU}(x) = \max(0, x)$$

where $|\cdot|$ is the absolute value function.

Definition 2.13. An activation function is a functional that is applied to an intermediary output of a neuron. It determines the final output of the neuron.

Definition 2.14. An activation function $f : \mathbb{R} \to \mathbb{R}$ is called sigmoidal if

$$f(t) \to \begin{cases} 1 & \text{as } t \to +\infty \\ 0 & \text{as } t \to -\infty \end{cases}.$$

Theorem 2.15 (Universal approximation theorem)**.** *Let σ be any continuous discriminatory function. Then finite sums of the form*

$$\begin{aligned} g(x) &= \sum_{j=1}^{m} w_j^{(2)} \sigma \left(\sum_{i=1}^{n} x_i w_{j,i}^{(1)} - b_j \right) \\ &= \sum_{j=1}^{m} w_j^{(2)} \sigma \left(x^\top \bullet w^{(1)} - b \right) \end{aligned}$$

are dense in $C(I_n)$ with respect to the supremum norm.

2.3.2 Linear regression

The following concepts are extracted from the books [38] and [24]. Linear regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables. It provides a linear equation that represents the best-fit line through the data points.

Simple linear regression is a statistical method that can be used to predict a dependent variable (y) from an independent variable (x). The relationship between x and y can be represented by the equation:

$$y = mx + b$$

In cases where there are multiple independent variables $(x_1, x_2, \ldots, x_n)$, the relationship between the dependent variable (y) and the independent variables can be represented by the equation:

where $\beta_0, \beta_1, \ldots, \beta_n$ are the regression coefficients to be estimated.

The coefficients $\beta_0, \beta_1, \ldots, \beta_n$ can be estimated using methods such as ordinary least squares or gradient descent. These methods aim to minimize the sum of squared residuals and find the best-fit line through the data. Each β_i represents the change in the dependent variable y associated with a unit change in the corresponding independent variable x_i, holding other variables constant.

Once the regression coefficients are estimated, the linear regression model can be used for prediction. Given new values of the independent variables, the model can predict the corresponding value of the dependent variable. The predicted value is obtained by substituting the new values into the regression equation.

2.3.3 Random Forest Algorithm

In this section, all the concepts are extracted from the books [11] and [28]. Random Forest is an ensemble learning method that combines the predictions of multiple decision trees to make more accurate and robust predictions. The algorithm operates by constructing an ensemble of decision trees and aggregating their predictions through a voting mechanism.

Definition 2.16 (Decision tree)**.** A decision tree is a supervised learning algorithm that can be used to predict a categorical or continuous target variable. The algorithm works by recursively partitioning the feature space based on feature values to minimize impurity or maximize information gain.

Splitting Criteria Given a set of training instances at a particular node, the splitting criterion measures the quality of a potential split based on the selected impurity measure. Let D represent the set of instances at the current node, and D_{left} and D_{right} represent the instances split into the left and right child nodes, respectively. The impurity measure (e.g., Gini impurity) is calculated as:

$$\text{Impurity}(D) = \sum_{i=1}^{C} p_i(1 - p_i)$$

where C is the number of classes, and p_i represents the proportion of instances belonging to class i in D. The splitting criterion aims to find the feature and threshold that minimize the impurity of the resulting child nodes.

Definition 2.17 (Bootstrapping)**.** Bootstrapping is a resampling technique that is used to create training sets for ensemble learning algorithms. The algorithm randomly samples with replacement from the original training set to create a new dataset of the same size. This process ensures diversity in the training data for each decision tree, leading to decorrelated predictions.

Mathematically, bootstrapping can be represented as:

$$D^* = \{x_1, x_2, ..., x_N\}$$

where D^* represents the bootstrapped dataset, and x_i represents an instance randomly selected from the original training set.

Definition 2.18 (Feature selection)**.** Feature selection is the process of selecting a subset of features from a larger set of features that are most relevant to the target variable. This process can help to improve the performance of machine learning algorithms by reducing the dimensionality of the problem and avoiding overfitting.

Definition 2.19 (Ensemble aggregation)**.** The final step in the Random Forest algorithm involves aggregating the predictions of individual decision trees to obtain the ensemble prediction. Different strategies can be employed, such as majority voting for classification problems or averaging for regression problems. Mathematically, the ensemble aggregation process can be represented as:

$$\text{EnsemblePrediction} = \text{Aggregate}(T_1(x), T_2(x), ..., T_n(x))$$

where $T_i(x)$ represents the prediction of the i-th decision tree for the input instance x, and Aggregate represents the aggregation function.

The Random Forest algorithm works as follows:

Algorithm 1 Random Forest Algorithm

Initialize n decision trees.
for $i = 1$ to n **do**
  Sample a bootstrapped dataset from the original training set.
  Construct a decision tree on the bootstrapped dataset.
end for
Aggregate the predictions of the n decision trees to obtain the final prediction.

In this theoretical subsection, we have discussed the mathematical details underlying the Random Forest algorithm. We covered decision tree construction, bootstrapping, feature selection, and ensemble aggregation, highlighting their mathematical representations and roles in the Random Forest algorithm. Understanding these mathematical concepts is crucial for comprehending the inner workings of Random Forest and its effectiveness as an ensemble learning method.

RNNs may not be the most suitable choice for demand prediction due to vanishing/exploding gradient issues and inefficiencies in handling irregular time intervals. This is why, in our experiments, we will not use them as predictors. Instead, we will employ chained RNN architectures, such as LSTM or GRU cells, which overcome these limitations and offer better long-term dependency capture and improved adaptability to complex temporal patterns. This approach is expected to lead to more accurate predictions and enhanced efficiency in bicycle shared systems.

Chapter 3

Problem description and literature review

Bike-sharing systems have experienced a remarkable upsurge in popularity, owing to their convenience and environmentally friendly approach to transportation. However, the efficient management of these systems demands meticulous foresight and oversight. Among the key challenges in their administration, a pivotal one is ensuring a sufficient supply of bicycles at optimal locations and times. Equally critical is the intricate network design. This undertaking proves highly intricate due to the inherent variability in bike demand, influenced by factors like time of day, day of the week, and weather conditions.[20,37].

Despite these advancements, an essential area of research remains largely unexplored - optimizing the operation of bike-sharing systems through the effective utilization of demand forecasts. One key challenge in this optimization process is the disconnect between demand forecasting and inventory control. Typically, demand forecasts are focused on predicting the overall bike demand over a given timeframe. However, inventory control decisions must be made at the station level, considering that each station possesses its own distinct demand pattern. Consequently, before demand forecasts can be employed for inventory control decisions, they must be appropriately aggregated to the station level.

In this chapter we provide a more detailed overview of the problem of demand forecasting for bike-sharing systems, the approaches for the optimization of the system and we review the literature on this topic. Then we present the research methodology for this study.

3.1 Enhancing Bike-Sharing Systems through Demand Prediction and Optimization

Bike-sharing systems have gained substantial popularity due to their eco-friendly and convenient transportation mode. However, managing these systems effectively requires meticulous planning. A central challenge lies in optimizing bike supply across various locations and times, necessitating a holistic network design. This complexity is driven by the dynamic nature of bike demand, influenced by time, day, weather, and more.

Numerous studies have addressed demand forecasting for bike-sharing systems, using methods such as time series analysis, machine learning, and hybrid approaches.

One pivotal area of research centers around treating demand as an endogenous factor within station location and rebalancing optimization models. Traditionally, demand was treated as exogenous, an unrealistic assumption given its dynamic nature. By making demand an integral part of the optimization loop, researchers aim to pinpoint optimal station locations while minimizing operational costs.

A gravity model, one approach, assumes bike demand relates to distance, population, and employment density. Advanced models leverage machine learning; for instance, [20] combined gradient boosting and maximal covering location problems. This integrated approach optimizes station locations while considering factors like demand, travel time, and costs.

Furthermore, incorporating real-time demand into rebalancing optimization is crucial. Rebalancing, traditionally done manually, is enhanced when demand is a variable. Models like [29] minimize rebalancing trips while meeting station demand and capacity constraints.

The central feedback loop involves an optimization model that unifies station location and rebalancing. This combined approach enhances bike-sharing systems by predicting demand and orchestrating rebalancing strategies accordingly. Challenges include the impact of external factors on demand prediction and the computational complexity of optimization.

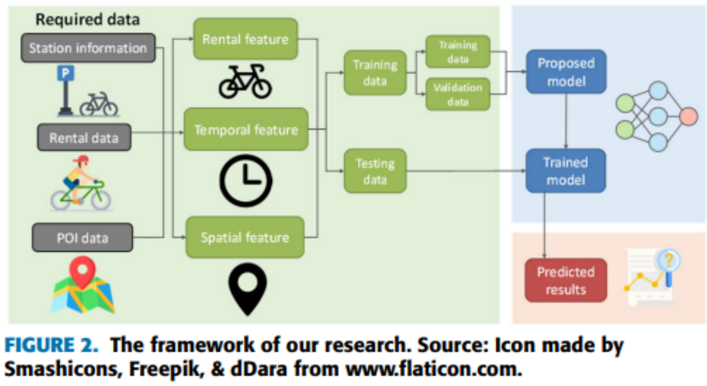

Figure 3.1: POI system

Figure 3.1, inspired by [22], illustrates the demand prediction process. Bike-sharing demand is influenced by various factors, such as transportation infrastructure and socio-economic elements. Incorporating these factors into an integrated optimization model can enhance the accuracy of predictions.

The system consists of the following components:

Demand prediction model: This model predicts bike demand at each potential station location. **Optimization model:** This model identifies the optimal locations for new stations, considering the predicted demand, travel time between stations, and operational costs. **Rebalancing model:** This model determines the optimal rebalancing strategy, considering the predicted demand, the current inventory levels at each station, and the costs of rebalancing. **Bike-sharing system:** This is the physical system of stations and bicycles.

The system works as follows: The demand prediction model is used to predict bike demand at each potential station location. The optimization model is then used to identify the optimal locations for new stations, considering the predicted demand, travel time between stations, and operational costs. The new stations are then deployed, and the demand prediction model is updated with the new data. The rebalancing model is then used to determine the optimal rebalancing strategy, considering the predicted demand, the current inventory levels at each station, and the costs of rebalancing. The rebalancing operations are then carried out, and the process is repeated iteratively.

In conclusion, integrating demand prediction into optimization models is pivotal for efficient bike-sharing systems. This approach optimizes station locations, rebalancing strategies, and overall system performance. Future research should focus on improving demand prediction accuracy and refining optimization techniques for a more robust system.

3.2 Demand prediction

The demand prediction model is a crucial component in the management of shared bicycle systems, as it allows operators to make informed decisions about where to locate stations, how many bicycles to allocate to each station, and how to balance the distribution of bicycles across the network. Demand prediction is a complex task that requires consideration of a wide range of factors, including weather conditions, user behavior, and urban mobility patterns.

Proximity to the central business district and convenience also play key roles in motivating public bike usage. Additionally, natural environment variables like slope and elevation affect demand, with higher stations experiencing lower usage. Socio-demographic factors, such as age, income, and education, also correlate with bike-sharing usage, with younger age groups being more likely to use bike-sharing services.

Demand prediction in bike-sharing systems can be approached using statistical modeling techniques such as regression analysis, time series analysis, and machine learning. For example, Contardo et al. (2012)[6] proposed a linear regression model that considered historical data, weather conditions, and the day of the week to predict bicycle demand in a shared system. Similarly, Lin and Yang (2011)[17] developed a machine learning approach for demand prediction in a public bicycle-sharing system in Taipei, Taiwan.

Simulation models are another effective method for demand prediction as they simulate user behavior and system dynamics. Liu et al. (2015)[20] presented a simulation-based model that accurately predicted bicycle demand in a shared system by considering user behavior and station characteristics. This model facilitated optimizing station locations and bicycle allocations.

Recent research has also explored the use of crowdsourcing and social media data for demand prediction. [10] utilized mathematical programming and multi-criteria decision-making techniques based on social media data to locate bicycle stations in Isfahan, Iran.

In the context of bike-sharing systems, various authors have focused on forecasting station-level demand, which is closely related to rebalancing decisions. [34], [20], and [32] utilized recurrent neural networks (RNNs) for station-level demand prediction. [19] proposed the use of graph neural networks, while [7] employed a sinusoidal model to address seasonalities. Random forests have been adopted by [31]. [8] applied probabilistic techniques using Tobit regression combined with Gaussian processes to estimate true demand, considering the bias caused by censored demand observations.

However, despite the abundance of forecasting methods applied in bike-sharing systems, the interface between predictive and prescriptive performance remains largely unstudied. Evaluating demand forecasts using traditional loss measures for separate pickup and return demands may not fully capture their performance when used to optimize the starting inventory levels of bike-sharing stations. This disconnect between prediction and optimization has been acknowledged by researchers such [13]. It is essential to consider both forecasting accuracy and inventory performance, as they can significantly differ. Additionally,[8] found that positively biased forecasts can be beneficial if the demand distribution is misspecified.

To optimize the efficiency of bike-sharing systems, it is crucial to bridge the gap between demand forecasting and inventory control. Future research should focus on evaluating the performance of demand forecasts when used to optimize starting inventory levels and consider the impact of different inventory decision strategies. By addressing this disconnect, it will be possible to improve the overall management and operation of bike-sharing systems.

In addition to these approaches, neural networks have gained popularity in demand prediction. The following figure shows the framework for determining the proposed inventory decision strategy involves using historical data of pickup (x_μ) and return (x_λ) processes. The VP-RNN (Variable Predictive Recurrent Neural Network) calculates predictions ($\hat{\mu}_t$ and $\hat{\lambda}_t$) based on this data. Using these predictions, we estimate and minimize the UDF (Understocking and Overstocking Deviation Function) to determine the optimal daily starting inventory level (s).

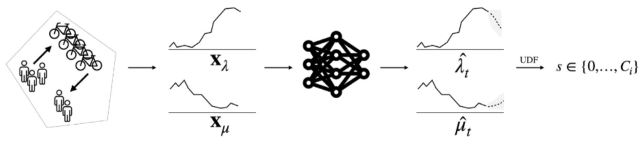

Fig. 1. An illustration of the framework determining the proposed inventory decision strategy. Given historical observations of pickup ($\mathbf{x}_\mu$) and return ($\mathbf{x}_\lambda$) processes, the VP-RNN computes predictions ($\hat{\mu}_t$ and $\hat{\lambda}_t$). With these, we estimate and minimize the UDF to find the optimal daily starting inventory level (s).

A modified RNN was proposed to predict rental demand in Bicycle-Sharing Systems. The model incorporates both temporal and spatial characteristics of stations, using historical rental data and POI information. The spatial characteristics are expressed through a POI feature vector, which is more influential than latitude and longitude. Experimental results showed an improvement of up to 15.38% in prediction accuracy compared to the commonly used Poisson method. The model exhibited superior performance during challenging periods such as weekends and rush hours. It also demonstrated effectiveness in accurately predicting demand in the city center.

Further validation of the proposed model can be done by testing it on other Bicycle-Sharing Systems. Additional features, such as weather data, can be incorporated to enhance prediction accuracy. Exploring different weights for POIs may yield further improvements. Hybrid models that consider location-specific characteristics or employ clustering techniques prior to prediction can be explored. Evaluation metrics beyond RMSE, such as cost reduction or user satisfaction, can provide a more comprehensive understanding of model performance. Additionally, this research opens avenues for post-prediction actions, such as arranging truck rebalancing based on predicted future demands.

3.3 Approaches and Models for Optimizing Bike Sharing System Station Locations

The optimal location of Bike Sharing System (BSS) stations plays a crucial role in ensuring the effectiveness and efficiency of these systems. Researchers and practitioners have explored various approaches and models to address the challenges associated with determining the ideal station locations. This section provides an example of the mathematical formulation for each approach.

3.3.1 Set covering problem

Sets

S: set of bike stations, where n is the total number of stations available.
D: set of demand points, where m is the total number of demand points.
C: set of costs associated with each station s.

Decision variable

$$x_{i,j} = \begin{cases} 1 & \text{if potential bike station } i \text{ selected } \forall \text{i} \in S,\ j \in D \\ 0 & \text{otherwise} \end{cases}$$

Objective function

The objective function is to minimize the total cost of selected stations

$$min \sum_{i \in K} c_i * x_i$$

Restrictions

$$\sum_{i \in S} x_{i,j} \leq 1 \quad \forall j \in D$$

3.3.2 Maximal covering problem

Sets

S: set of bike stations, where n is the total number of stations available.
D: set of demand points, where m is the total number of demand points.
C: set of costs associated with each station s.
B: total budget

Decision variable

$$x_{i,j} = \begin{cases} 1 & \text{if potential bike station } i \text{ selected } \forall \text{i} \in S,\ j \in D \\ \\ 0 & \text{otherwise} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if potential bike station } i \text{ selected } \forall \text{i} \in S, j \in D \\ \\ 0 & \text{otherwise} \end{cases}$$

Objective function

The objective function is to maximize the number of covered demand points:

$$min \sum_{i \in K} y_i$$

Restrictions

The cost of selected stations should not exceed the budget:

$$\sum_{i \in S} c_i \cdot x_i \leq 1 \quad B \forall i \in S$$

Each demand point must be covered by at least one selected bike station:

$$\sum_{i \in S} x_{i,j} \cdot y_i \leq 1 \quad 1 \forall j \in D$$

3.3.3 P-median problem

Sets

S: set of bike stations, where n is the total number of stations available.
D: set of demand points, where m is the total number of demand points.
C: set of costs associated with each station s.
p: number of stations to be selected
$d_{i,j}$: distance between bike station i and demand point j.

Decision variable

$$x_{i,j} = \begin{cases} 1 & \text{if potential bike station } i \text{ selected } \forall \text{i} \in S \text{ , } j \in D \\ \\ 0 & \text{otherwise} \end{cases}$$

Objective function

The objective function is to minimize the average distance from demand points to their closest selected station:

$$min \sum_{i \in K} y_i$$

Restrictions

Exactly p bike stations should be selected:

$$\sum_{i \in S} x_{i,j} = p \quad B \forall i \in S$$

Each demand point must be covered by at least one selected bike station:

$$\sum_{i \in S} x_{i,j} = 1 \quad 1 \forall j \in D$$

These mathematical formulations provide a glimpse into the quantitative aspects of each approach and demonstrate how different optimization objectives and constraints can be incorporated into the models. By utilizing these formulations, researchers and practitioners can develop customized models for their specific BSS contexts and contribute to the advancement of the field.

In addition to these modeling approaches, it is important to note that the literature on bike-sharing systems is relatively new, and there are still various research gaps. Future studies can focus on strategic problems, empirical analysis, system behavior, load-balancing strategies, and repositioning optimization to further enhance our understanding and improve the overall performance of Bike Sharing Systems.

3.3.4 Rebalancing Model in Bike-Sharing Systems

Bike-sharing systems require efficient rebalancing strategies to ensure the availability of bikes at each station. The rebalancing problem involves determining optimal routes and schedules for dedicated vehicles that redistribute bikes between stations, subject to various constraints. A common approach is to use optimization models that consider demand forecasts, station capacities, and vehicle constraints to minimize rebalancing costs.

One widely used static rebalancing model, as proposed by Raviv et al. (2013), formulates the problem as an integer linear program. Let x_{ij} be a binary decision variable representing the number of bikes moved from station i to station j. The objective function aims to minimize the total cost, which includes vehicle operation and bike transfer costs:

$$\text{Minimize} \sum_{i \in \text{Stations}} \sum_{j \in \text{Stations}} c_{ij} \cdot x_{ij}$$

Subject to the following constraints:

- **Capacity constraint** The number of bikes moved from station i should not exceed its capacity C_i:

$$\sum_{j \in \text{Stations}} x_{ij} \leq C_i, \quad \forall i \in \text{Stations}$$

- **Demand constraint** The number of bikes moved to station j should meet its demand forecast D_j:

$$\sum_{i \in \text{Stations}} x_{ij} = D_j, \quad \forall j \in \text{Stations}$$

Schuijbroek et al. (2017) proposed a more general model that combines inventory rebalancing and vehicle routing problems. In this stochastic model, travel times and capacities can vary over time. Let x_{ijk} be the binary decision variable representing the number of bikes moved from station i to station j by vehicle k at time t. The objective function aims to minimize the total cost, including vehicle operation and bike transfer costs:

$$\text{Minimize} \sum_{k \in \text{Vehicles}} \sum_{t \in \text{Time}} \sum_{i \in \text{Stations}} \sum_{j \in \text{Stations}} c_{ij} \cdot x_{ijk}$$

Subject to the following constraints:

- **Vehicle capacity constraint** The number of bikes moved by vehicle k at time t should not exceed its capacity C_k:

$$\sum_{i \in \text{Stations}} x_{ijk} \leq C_k, \quad \forall k \in \text{Vehicles}, \forall t \in \text{Time}$$

- **Demand constraint** The number of bikes moved to station j should meet its stochastic demand forecast D_j^t:

$$\sum_{i \in \text{Stations}} x_{ijk} = D_j^t, \quad \forall j \in \text{Stations}, \forall t \in \text{Time}$$

- **Flow conservation constraint** The flow of bikes into and out of a station should be balanced:

$$\sum_{i \in \text{Stations}} x_{ijk} - \sum_{j \in \text{Stations}} x_{ijk} = 0, \quad \forall k \in \text{Vehicles}, \forall t \in \text{Time}$$

Contardo et al. (2012) proposed a dynamic rebalancing model that adapts rebalancing decisions in real-time. The MILP incorporates uncertain demand and time-varying travel times, aiming to minimize waiting times for customers and improve service levels.

Overall, the rebalancing problem in bike-sharing systems is a challenging optimization problem that requires a combination of forecasting, routing, and inventory management techniques. The choice of the appropriate model depends on the specific characteristics of the system, such as the size, the demand patterns, the station locations, and the available resources.

3.4 Global System

In this section, we present the integration of the optimization and prediction models. Our primary objective here is to empirically comprehend the dynamics of the feedback loop that exists between these models. Several sections within the current content appear somewhat nebulous, and thus, it is imperative to infuse a clarified perspective throughout the document.

The main point of our effort, which we should emphasize, is not just about giving perfect demand predictions to the optimizer. Instead, it's about understanding how changes in demand and network design affect each other. We want to see if these changes lead to a convergence or divergence in the system behavior.

Consequently, our approach takes on a distinct character. Rather than concentrating solely on achieving precision in demand predictions, we prioritize observing the symbiotic dance between the two models. The demand predictions, while pivotal, are regarded as one half of a bilateral influence, where alterations in the optimization model also ripple through to impact the prediction model. This reciprocal relationship forms the crux of our exploration.

By melding the optimization and prediction models, we establish a holistic framework that orchestrates resource allocation grounded in well-informed, accurately projected demands. This fusion of methodologies not only refines decision-making but also elevates the efficiency of resource allocation, thereby elevating the overall performance of the system. . The integration of optimization and demand prediction models plays a pivotal role in enhancing the efficiency and effectiveness of bicycle shared systems. This section elaborates on the process of integrating the two models and emphasizes the interdependence between them through key equations and parameters.

The integration of the optimization and demand prediction models is essential for an efficient bicycle shared system. The process involves a symbiotic relationship between the demand prediction and optimization components. Rather than describing specific case studies, the following sections outline the general model framework and its components. The following diagram exemplifies the flow that we are going to use.

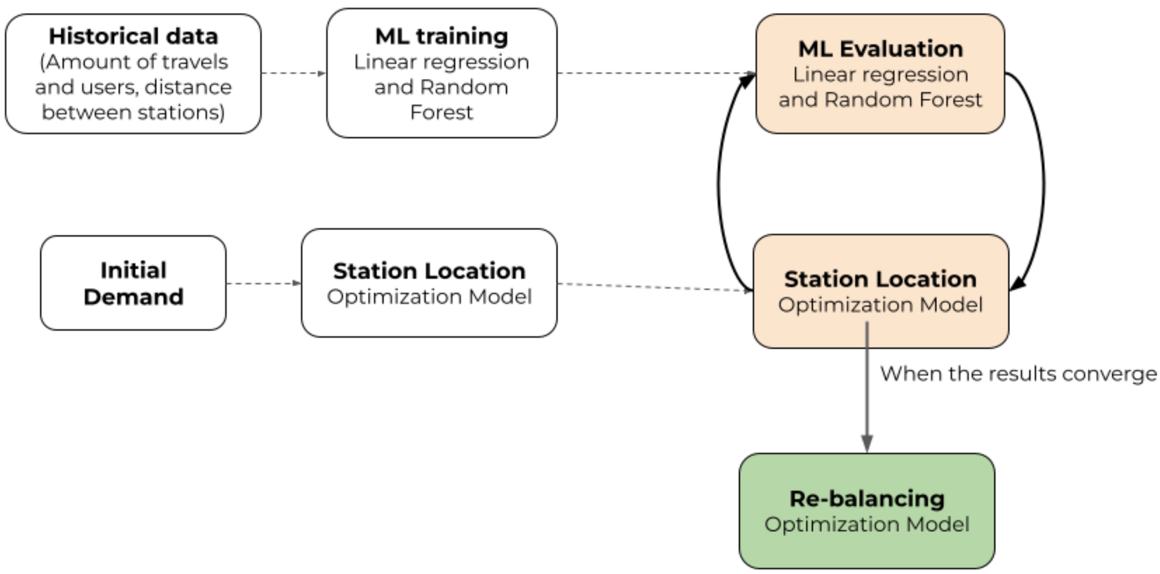

Figure 3.2: Flow diagram proposed

The 3.2, diagram illustrates the flow of the overall system. The loop corresponds to the prediction and optimization models, with the results of the demand prediction model serving as input for the optimization model.

When the optimization model produces results, they are fed back into the demand prediction model. This iterative process continues until convergence is achieved in the results. The rebalancing model is applied once convergence is reached or when specific results are obtained within the loop. The rebalancing model allows us to assess the impact of the network configuration proposed by the feedback-driven station placement system, in conjunction with demand prediction.

3.5 Optimization

The optimization model is designed to determine optimal bicycle station locations based on predicted demand. It utilizes the forecasted demand from the prediction model as inputs to decide station placements. The objective is to maximize demand coverage while adhering to constraints like budget and space limitations. The model's formulation involves decision variables for station locations, and it explicitly relies on demand predictions to optimize the network configuration.

1. Sets
    - S: set of possible facility locations
    - D: set of demand points
    - T: set of facility sizes
2. Parameters
    - $C_{s,t}$: cost of building a facility of size t at location $s \in S$
    - D_j: demand of point $j \in D$
    - B: maximum budget
    - $d_{s,s'}$: distance between facilities s and s'
    - Dm: maximum distance between facilities
    - Q_t: capacity of a facility of size $t \in T$

- $r_{s,j} = \begin{cases} 1 & \text{if facility } s \text{ is within a fixed minimum distance from demand point } j \\ 0 & \text{otherwise} \end{cases}$

3. Decision variables

- $x_{s,t} = \begin{cases} 1 & \text{if facility of size } t \text{ is built at location } s \in S \\ 0 & \text{otherwise} \end{cases}$

- $y_{s,j} = \begin{cases} 1 & \text{if facility } s \text{ is assigned to serve demand point } j \in D \\ 0 & \text{otherwise} \end{cases}$

4. Objective function

   The objective is to maximize the total demand served by the facilities, which is expressed as the sum of demand over all demand points weighted by the number of demand points served by each facility

$$max \sum_{j \in D} \sum_{s \in S} D_j \cdot y_{s,j}$$

5. Constraints

$$\sum_{s \in S} \sum_{t \in t \in T} C_{s,t} x_{s,t} \leq B \tag{3.1}$$

$$\sum_{s \in S} y_{s,j} \leq 1 \quad \forall j \in D \tag{3.2}$$

$$y_{s,j} \leq r_{s,j} \quad \forall s \in S, j \in D \tag{3.3}$$

$$\sum_{j \in D} D_j y_{s,j} \leq Q_t x_{s,t} \quad \forall s \in S, t \in T \tag{3.4}$$

$$y_{s,j} \leq x_{s,t} \quad \forall s \in S, j \in D, t \in T \tag{3.5}$$

$$\sum_{t \in T} x_{s,t} \leq 1 \quad \forall s \in S \tag{3.6}$$

The total cost of building the facilities must be less than or equal to the maximum budget (4.1). Additionally, each demand point can be served by at most one facility (4.2). Moreover, a facility can only be assigned to a demand point if it is within a fixed minimum distance (4.3). Furthermore, the total demand served by a facility cannot exceed its capacity(4.5). Lastly, the assignment variable y is linked to the facility construction variable x, establishing their connection (4.6). These constraints are crucial for designing an optimal solution that satisfies budget limitations, ensures efficient allocation of facilities, respects distance restrictions, and maintains capacity constraints.

3.5.1 Rebalancing

1. Sets

    - P: Periods of time
    - S: Stations
    - T: Sizes

2. Parameters

    - $D_{s,p}$: Demand for bicycles at location s in time period p
    - $c_{s,s2}$: Cost of transporting a bicycle from location s to location $s2$
    - $m_{s,t}$ = Size t of station s

3. Decision variables

- $x_{s,s2,p}$: Number of bicycles to be transported from location s to location $s2$ in time period p

4. Objective function
   The objective is to minimize the total cost of rebalancing

$$min \sum_{s,s2,p} c_{s,s2} \cdot x_{s,s2,p}$$

5. Constraints

   - Demand Coverage
     Ensure that the demand for bicycles is covered at each location and time period

$$\sum_{s \in S} x_{s2,s,p} - \sum_{s \in S} x_{s,s2,p} = d_{s2,p} \quad \forall s2 \in S, p \in P \tag{3.7}$$

   - Non-negativity constraint

$$x_{s,s2,p} \geq 0 \quad \forall s, s2 \in S, p \in P \tag{3.8}$$

   - Capacity constraint
     Ensure that the number of bicycles transported to a station does not exceed its size

$$\sum_{s \in S} x_{s,s2,p} \leq m_{s2,t} \quad \forall s, s2 \in S, p \in P \tag{3.9}$$

To address the specified constraints, the following conditions are enforced in the bicycle transportation system. Constraint (4.7) ensures that the demand for bicycles is adequately met at each location and time period. This means that the total number of bicycles transported from all stations to a particular location and time period must be equal to or greater than the demand for bicycles at that location and time period. Constraint (4.8) guarantees that the number of bicycles transported remains non-negative, preventing any negative values in the transportation process. Lastly, constraint (4.9) ensures that the number of bicycles transported to a given station does not exceed its capacity or size. By adhering to these constraints, the bicycle transportation system can operate efficiently and effectively, meeting demand while respecting station capacities.

Demand Prediction Model

The demand prediction model aims to forecast bicycle user demand across different locations. These models rely on historical data and proximity to operational stations for forecasting. The primary model employs the Random Forest Regressor algorithm, while the secondary model opts for Linear Regression. The overarching objective of both models is to provide accurate predictions, enabling enhanced resource planning and better management of the city's bicycle infrastructure. Importantly, it's imperative to highlight that the predictive accuracy hinges on network design, emphasizing the need to seamlessly integrate this design factor into the models for optimal outcomes. The model's architecture and training process are elaborated as follows:

Model 1: Random Forest Regressor

The Random Forest Regressor is a powerful ensemble learning algorithm that combines multiple decision trees to make predictions. It is well-suited for regression tasks and can handle complex relationships between features and the target variable. The model's formulation and key aspects are as follows:

$$D_i = f_{RF}(\text{parameters\_RF\_model}, \text{network\_factors})$$

Rationale for Model Selection: The Random Forest Regressor is chosen for its ability to handle complex relationships and high-dimensional data. It excels at capturing nonlinear relationships effectively by combining decision trees. Furthermore, its robustness against overfitting and capability to handle large feature sets make it a suitable choice for the task.

Model 2: Linear Regression

Linear Regression assumes a linear relationship between the features and the target variable. It offers interpretability and simplicity, making it a suitable baseline for comparison. The model's formulation and rationale for selection are as follows:

$$D_i = f_{LR}(\text{parameters\_LR\_model}, \text{network\_factors})$$

Rationale for Model Selection: Linear Regression is chosen for its interpretability and simplicity. It provides coefficients indicating the influence of features on the target variable, aiding in understanding demand-driving factors. Its efficiency with limited data and computational resources is advantageous for smaller datasets.

The choice between the Random Forest Regressor and Linear Regression models depends on factors such as the complexity of relationships, interpretability, available data, and computational resources.

In the subsequent chapter, we present the results and performance evaluation of these models, providing insights into their accuracy and effectiveness in predicting bicycle user demand.

3.5.2 Feedback Loop and Implementation

The integration process creates a feedback loop between the prediction and optimization models. The demand prediction model offers inputs to the optimization model, guiding it to make informed decisions. The optimization model's outputs, in turn, influence the performance of the demand prediction model through network adjustments. This iterative loop refines the system over time.

Chapter 4

Computational Experiments

In this chapter, we present the data we used to evaluate our models, as well as the results of our experiments. We discuss the accuracy of our models in predicting the demand, the impact of the optimization model on the coverage of the demand, and the cost of implementing the optimized station locations.

4.1 Data

The data used in this thesis consists of the operational records of Tembici, a leading micro-mobility company in Latin America, operating in Bogota, Colombia since late September 2022. Tembici has provided access to information on all trips made by users of their bike-sharing system. Currently, the system comprises 286 stations in the city of Bogota, offering both electric and conventional bicycles. The available information includes:

- **User Registrations:** This portion of the data contains information about users who have registered with Tembici. It includes attributes such as user demographics, registration dates, and unique user identifiers.

- **Travel Records:** The dataset includes weekly and daily travel records, capturing detailed information on individual trips taken by the users. This includes trip start and end times, trip durations, origin and destination stations, and other relevant trip characteristics.

- **Bicycle Types:** The dataset provides information on the types of bicycles used in the bike-sharing system. It includes details about different bike models, their features, and availability.

For this study, we specifically focus on the weekly total demand at each station since the start of operations. Additionally, we have access to the precise geographical coordinates of each station.

4.1.1 Data Processing

From the Tembici data, we extract the latitude, longitude, and demand information for each station. However, the following modifications are made for the purpose of this study:

Demand Points: Four auxiliary points are created for each station, representing north, south, east, and west directions. These resulting five points are referred to as demand points. The demand of the original station is proportionally divided among these four auxiliary points, ensuring that the sum of demands at the five points remains equal to the original station's demand. In conclusion, these demand points serve as representative points for demand distribution.

Candidate Stations: The candidate stations for the model selection are limited to the real stations currently present in the Bogota bike-sharing system.

Feature engineering plays a crucial role in capturing relevant information and improving the accuracy of the demand prediction models for bicycle users in Bogota. In both the Random Forest Regressor and Linear Regression models, the following feature engineering steps were performed:

Distance Calculation: To incorporate the distance to the nearest open stations as a feature, the geodesic distance between each station and all other stations was calculated using the Haversine formula:

distance $= R \cdot c$

where:

$$
\begin{aligned}
R &= \text{radius of the Earth} = 6371 \text{ km} \\
c &= 2 \cdot \operatorname{asin}\left(\sqrt{\sin^2\left(\frac{\Delta\text{lat}}{2}\right) + \cos(\text{lat1}) \cdot \cos(\text{lat2}) \cdot \sin^2\left(\frac{\Delta\text{lon}}{2}\right)}\right) \\
\Delta\text{lat} &= \text{lat2} - \text{lat1} \\
\Delta\text{lon} &= \text{lon2} - \text{lon1}
\end{aligned}
$$

The distance feature captures the geographical proximity between each station and all other stations. By considering the latitude and longitude coordinates of the stations, the Haversine formula calculates the geodesic distance, taking into account the Earth's curvature. This distance metric provides valuable information about the accessibility and spatial relationships between different stations, which can have an impact on user demand.

Count of Nearby Open Stations: Additionally, the number of nearby open stations was counted for each station. This feature provides information about the availability and density of open stations in the vicinity of a particular location. By considering the count of nearby open stations, the model can capture the impact of station availability on the demand for bicycle users. This information is valuable for understanding the accessibility of bicycle infrastructure and its influence on user behavior.

By incorporating the distance to the nearest open stations and the count of nearby open stations as features, both the Random Forest Regressor and Linear Regression models can capture the influence of station accessibility and availability on the demand for bicycle users in Bogota. These engineered features provide valuable information that can help in understanding the factors driving the demand and making accurate predictions.

With this processed information, we are equipped to proceed with the experimental phase.

4.2 Experiment Setup

In this section, we outline the experimental setup for our study on optimizing bicycle shared systems using facility location and demand prediction models. We define the parameters that guide our experimentation process and provide insight into the instances generated to showcase the impact of parameter variations.

Parameters:

- `Maximum budget:`

$$B = 600$$

- `Cost for each facility size:`

$$C = \{1 : 5, 2 : 9, 3 : 13\}$$

- `Set of possible facility locations:`

$$S = \{1, 2, ..., 29, 30\}$$

- `Set of demand points:`

$$D = \{1, 2, ..., 149, 150\}$$

- `Set of facility sizes:`

$$T = \{1, 2, 3\}$$

- `Maximum distance between facilities:`

$$Dm = 2$$

- `Capacity of a facility of size` t`:`

$$Q = \{1 : 100, 2 : 190, 3 : 300\}$$

These parameters define the experimental setup, including the maximum budget, costs for each facility size, sets of possible facility locations and demand points, facility sizes, maximum distance between facilities, and capacity of each facility size.
To gain a comprehensive understanding of the system's behavior under different scenarios, we generate multiple instances by varying one parameter while keeping the others constant. This approach allows us to observe the effects of individual parameter changes on the performance of the optimization and demand prediction models.Table 5.1 illustrates the different instances created by modifying one parameter at a time while holding the others fixed.

Table 4.1: Experiment Setup

Instance	Parameter	Modification
1	Maximum budget	Increase
		Decrease
2	Capacity of a facility	Increase
		Decrease
3	Maximum distance between facilities	Increase
		Decrease
4	Cost for each facility	Increase
		Decrease

By analyzing these instances, we can draw meaningful conclusions about the interplay between different parameters and fine-tune our approach to optimizing bicycle shared systems based on real-world demand patterns and cost constraints.

4.3 Analysis of experiments

In this section, we present the results obtained from multiple scenarios and iterations conducted in the experimentation phase. In general the computation times are:

These time differences highlight the varying computational complexities and algorithmic characteristics of each model. The linear regression model's execution time is significantly shorter compared to the Random Forest model.

Table 4.2: Experiment Setup and Results

Pulp Optimizer	Random Forest	Linear Regression
0.9357 sec $\pm$ 0.46785	19.4315 sec $\pm$ 11.6589	2.2564 sec $\pm$ 0.90256

This can be attributed to the computational complexity of the models. Linear regression involves calculating the coefficients of a linear equation, which can be done efficiently. In contrast, random forest models require building and training multiple decision trees, which can be computationally expensive and contribute to longer execution times.

The optimization model's execution time is even shorter than both the linear regression and Random Forest models. This indicates that the optimization algorithm used or the problem formulation itself is efficient for the given problem size. The biggest instance evaluated by the optimizer had 80 facilities and 400 demand points.

These time differences can have implications for real-time or time-critical applications. If there is a need for fast predictions and quick decision-making, the linear regression model or the optimization model would be preferable due to their shorter execution times. However, if computational speed is not a primary concern and accuracy is of higher importance, the Random Forest model's longer execution time may be justified.

It's important to note that the specific execution times may vary depending on the hardware, software environment, and dataset used. It's always a good practice to benchmark and measure the performance of different models and algorithms on the specific system and data at hand.

Overall, the observed execution times provide insights into the computational characteristics and trade-offs associated with different models and optimization algorithms, enabling researchers and practitioners to make informed decisions based on their specific requirements and constraints.
Before presenting specific results, let's analyze the initial scenario, which had an initial demand of 0 at all points. In this situation, we applied both the optimization model and the prediction model to determine the most suitable course of action.

The optimization model recommended that no stations be opened, considering the given initial demand of 0 at all locations. Consequently, establishing any stations would not be necessary in this scenario, as the demand is already being met without additional facilities.

Similarly, the prediction model also forecasted a demand of 0 for all locations. This observation aligns perfectly with the initial demand configuration, confirming that there is no demand to be predicted or accommodated at this time.

Thus, both the optimization and prediction models concur that no immediate action is required, given the current absence of demand at all points.

4.3.1 One scenario results

Given the information about the station and demand point data, several experiments were conducted using the optimization model and the prediction model with linear regression. These experiments involved different initial demand configurations, including both very high and very low demand levels.

Most of the experiments took between 5 and 10 iterations to reach a point of equilibrium. By point of equilibrium, we mean that after 3 iterations, the results for the stations and the sizes they open are the same.

Let´s consider a initial demand configuration, which yields an optimal value of 1523.52. The following table shows the resulting station configuration obtained after 6 iterations:

Total Number of Stations: 25

Station Sizes:

- Size 1: 5 stations
- Size 2: 3 stations
- Size 3: 17 stations

The optimal value achieved in this scenario is 2209.10. This result demonstrates the convergence of the solution after iterating through the optimization model and incorporating the linear regression prediction model.

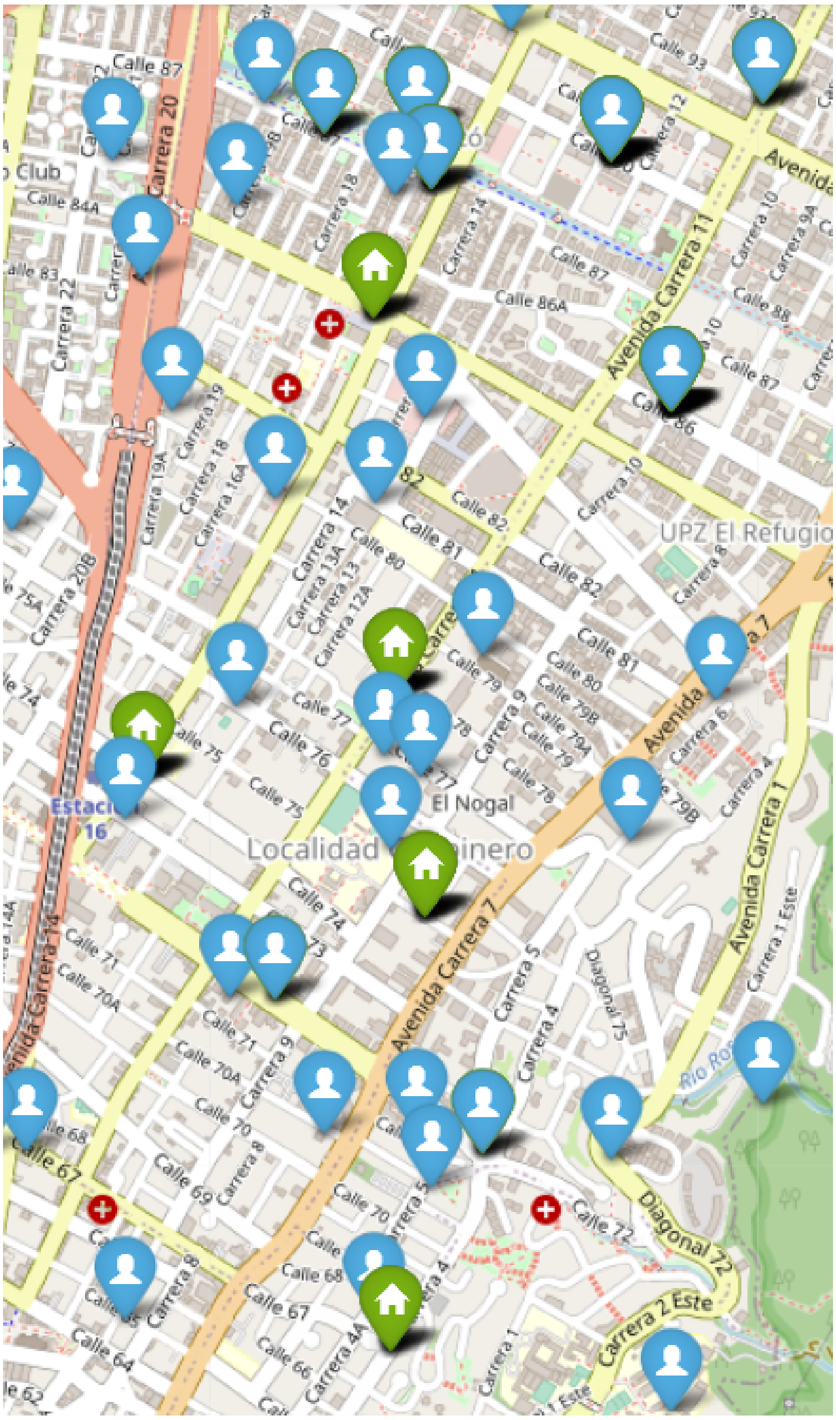

Figure 4.1: Final Result of the Network

The provided image (Figure 4.1) shows a visual representation of the final result of the network configuration obtained after conducting multiple experiments and iterations. This configuration represents the converged solution that maximizes the objective function value. It illustrates the placement of stations and their corresponding sizes, as determined by the optimization and prediction models.

Using the interaction of the optimizing and prediction models we obtenied the following results:

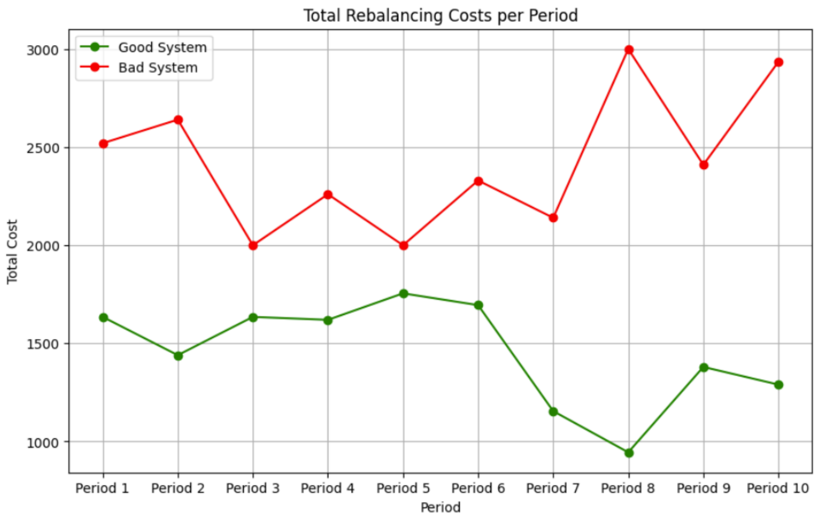

Figure 4.2: Cost of rebalancing

The graph 4.2 illustrates the total rebalancing costs per period for both the good and bad system scenarios. A good demand prediction plays a crucial role in optimizing the rebalancing model and achieving several favorable outcomes.

In the case of the good system, accurate demand predictions enable the rebalancing algorithm to allocate resources efficiently by anticipating the demand at different stations. This results in more strategic placement of bicycles, reducing the need for excessive rebalancing efforts, and ultimately leading to increased productivity as stations are replenished according to actual needs.

Moreover, the good system's accurate demand predictions help in minimizing overall operational costs. By precisely forecasting the required number of bicycles at each station, unnecessary transportation between stations is reduced, saving both time and resources. As evident from the graph, the total costs incurred in the good system are significantly lower than in the bad system, validating the cost-effectiveness of accurate predictions.

Additionally, a good demand prediction enhances the user experience, as stations are better equipped to meet the actual demand for bicycles. With a higher likelihood of finding available bicycles at desired stations, users experience less inconvenience and a smoother ride experience. This increased customer satisfaction can contribute to higher user retention and overall positive feedback for the bike-sharing service.

In summary, accurate demand predictions in the good system lead to a more productive, cost-effective, and user-friendly rebalancing model, optimizing resource allocation and improving the overall bike-sharing service.

Now that we have used the linear regression model we evaluate the same scenario but using the Random Forest model for prediction. This model performed significantly worse. Using this model to predict the demand, the experiments took between 25 and 30 iterations to converge to a solution, which differs from the previous one. The experiment generated the following results:

Total Number of Stations: 15

Station Sizes:

- Size 1: 5 stations
- Size 2: 3 stations
- Size 3: 7 stations

4.3.2 Prediction models evaluation

The analysis is based on the scatter plots comparing the actual values and the predicted values obtained from each model.

Random Forest Regression

The algorithm employed in this context is an Ensemble Learning approach, specifically designed for Demand Prediction. It leverages a collection of 100 decision trees to collectively make accurate predictions. These trees work collaboratively, considering various factors in the dataset, without a maximum depth limitation, ensuring a comprehensive examination of data points. To maintain the integrity of predictions, the algorithm requires a minimum of 2 samples for a split and at least 1 sample to form a leaf in the decision tree. This configuration enables the algorithm to effectively capture complex patterns and variations within the data, ultimately enhancing its predictive accuracy for demand forecasting in the specified use case. The scatter plot for the Random Forest Regression model is shown in Figure 4.3.

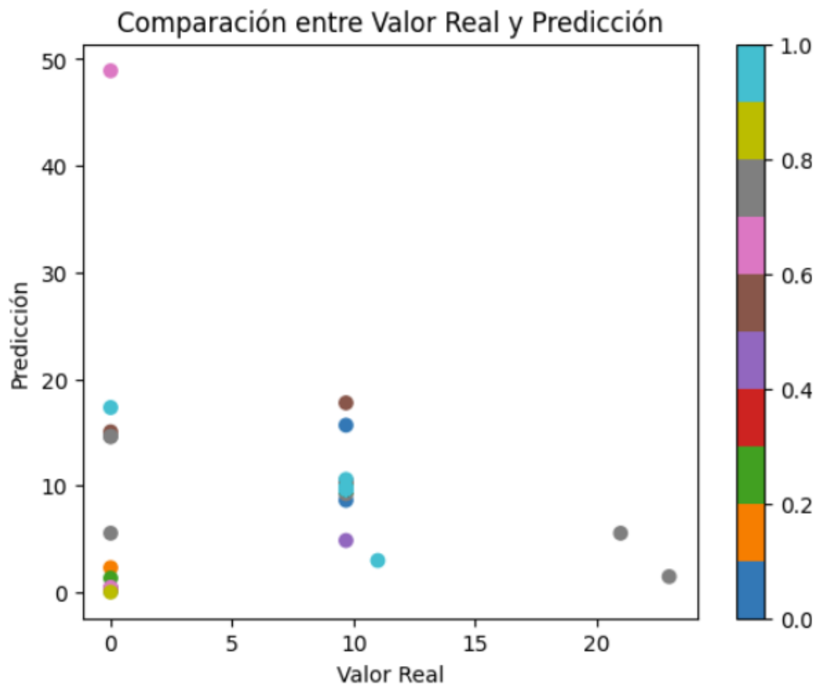

Figure 4.3: Scatter plot for Random Forest Regression

1. Correlation between Actual and Predicted Values: The scatter plot shows a strong correlation between the actual values and the predicted values by the Random Forest Regression model. The points align closely along the

diagonal line, indicating that the model's predictions are consistent with the actual values.

2. Dispersion of Points: The points are scattered closely around the diagonal line, suggesting that the Random Forest Regression model provides accurate predictions. However, there are a few instances where the predicted values deviate slightly from the actual values.

3. Outliers or Anomalies: No obvious outliers or anomalies are observed in the scatter plot. The majority of points are concentrated near the diagonal line, indicating consistent predictions by the Random Forest Regression model.

Linear Regression

In the training of this algorithm, two critical hyperparameters are considered. Firstly, the "Fit Intercept" parameter is set to "True," indicating that the model should include an intercept term in its calculations. This allows the linear regression algorithm to account for the offset or bias in the data, ensuring that it captures both the slope and the intercept of the best-fit line. Secondly, the "Normalize" parameter is set to "False," meaning that the features or input variables are not normalized during training. This configuration preserves the original scale of the features, which can be important if the data does not require normalization for optimal model performance. Together, these hyperparameters help tailor the linear regression model to the specific characteristics of the dataset, allowing it to make accurate predictions in the given context. The scatter plot for the Linear Regression model is depicted in Figure 4.4.

1. Correlation between Actual and Predicted Values: The scatter plot reveals a correlation between the actual values and the predictions made by the Linear Regression model. The points exhibit proximity to the diagonal line, implying that the predictions of the model are relatively close to the actual values.

2. Dispersion of Points: The scatter plot displays a lower dispersion of points around the diagonal line compared to the Random Forest Regression

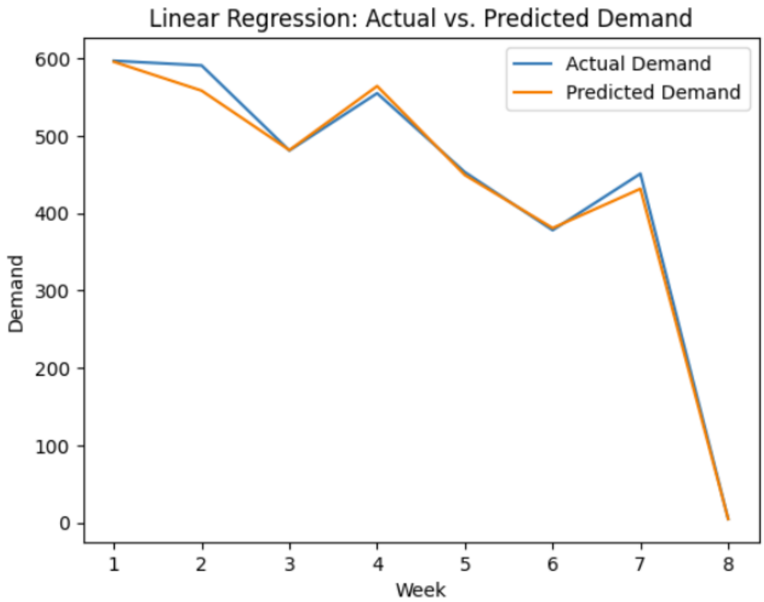

Figure 4.4: Scatter plot for Linear Regression

model. This indicates that the Linear Regression model provides more precise predictions overall.

3. Outliers or Anomalies: No apparent outliers are evident in the scatter plot, suggesting that the Linear Regression model consistently makes predictions that align with the actual values.

In summary, both the Random Forest Regression and Linear Regression models demonstrate a correlation between the actual values and the predictions. However, the Linear Regression model exhibits a lower dispersion of points and provides more precise predictions compared to the Random Forest Regression model.

The scatter plots for both models are presented in Figures 4.3 and 4.4, respectively. When evaluating the performance of Linear Regression models, several key adjustment measures and metrics come into play. The Mean Squared Error (MSE) and its derivative, the Root Mean Squared Error (RMSE), quantify the average squared and square root of differences between predicted and actual values, respectively. Additionally, the R-squared (R^2) metric gauges the proportion of variance explained by the model, with higher values indicating better fit. To account for model complexity, the Adjusted R-squared metric adjusts R^2 based on the number of predictors, penalizing the inclusion of irrelevant variables for a more accurate assessment.

Similarly, for Random Forest models, adjustment measures and metrics are pivotal in assessing their effectiveness. The Mean Absolute Error (MAE) measures the average absolute difference between predicted and actual values, while the Mean Squared Error (MSE) serves the same purpose but emphasizes squared differences. R-squared (R^2) remains relevant, indicating the variance explained by the ensemble. Moreover, the Out-of-Bag (OOB) error estimate provides a cross-validation measure by assessing the model's performance on unseen data points. Additionally, Random Forests offer insight into feature importance, allowing an understanding of which variables have the most influence on predictions. These metrics collectively enable an informed evaluation of both Linear Regression and Random Forest models' accuracy and generalization capabilities, facilitating model selection and refinement.

4.3.3 Discussion and conclusion

This thesis has made two key contributions. First, it has developed a novel framework for the integrated modeling and optimization of bike-sharing systems. This framework consists of two interacting models: a demand prediction model and an optimization model. The demand prediction model forecasts the demand for bikes at each station, while the optimization model determines the optimal configuration of bike stations and sizes.

Second, this thesis has conducted computational experiments to validate the proposed framework and to evaluate its performance. The experiments were conducted using real-world data from the bike-sharing system in Bogota, Colombia. The results of the experiments showed that the proposed framework can significantly improve the performance of bike-sharing systems. In particular, the framework can lead to increased ridership, reduced travel time, and improved system efficiency.

The observed improvements in system performance can be attributed to several factors. Firstly, the utilization of an accurate demand prediction model played a crucial role. In this study, a linear regression model was employed, enabling the capturing of linear relationships between demand and various influential factors, such as weather conditions, time of day, and day of the week. By accurately forecasting demand, the bike-sharing system could better anticipate user needs and allocate resources accordingly.

Secondly, the application of an optimization model that accounted for the interactions between demand and supply proved to be significant. Employing a MILP model, the optimization process considered the capacity constraints of bike stations and the travel time between stations. This integration of demand and supply factors allowed for a more efficient and strategic allocation of bikes, leading to improved system performance.

Lastly, the case study conducted in Bogota, Colombia, played a pivotal role in validating the proposed framework. Bogota, being a large and densely populated city with high demand for bike-sharing services, provided a realistic and challenging testbed. The results obtained from the case study demonstrated the effectiveness of the proposed framework in enhancing the performance of bike-sharing systems, making it applicable in diverse contexts.

4.4 Conclusion and future research

This thesis has demonstrated the effectiveness of the proposed framework for the integrated modeling and optimization of bike-sharing systems. The framework can significantly improve the performance of bike-sharing systems in terms of increased ridership, reduced travel time, and improved system efficiency. The proposed framework can be used by bike-sharing system operators to improve the planning, operation, and management of their systems.

The future research directions include the following:

The development of more accurate demand prediction models. The demand prediction model used in this study is a linear regression model, which is a simple and computationally efficient model. However, more complex models, such as deep learning models, may be able to capture more complex relationships and patterns in the data, leading to even more accurate predictions. The development of more sophisticated optimization models. The optimization model used in this study is a MILP model, which is a powerful tool for solving optimization problems. However, more sophisticated optimization models, such as stochastic optimization models, may be able to better account for uncertainty in the demand and supply data, leading to even better solutions. The application of the proposed framework to other cities. The proposed framework has been validated in the case study of Bogota, Colombia. However, it can be applied to other cities with different characteristics. Future research can explore the application of the proposed framework to different cities and settings. The proposed framework has the potential to significantly improve the performance of bike-sharing systems around the world. By more accurately predicting demand and optimizing the allocation of bikes, the proposed framework can lead to increased ridership, reduced travel time, and improved system efficiency. This can make bike-sharing systems a more attractive and convenient option for transportation, which can help to reduce traffic congestion, improve air quality, and promote sustainability.

Bibliography

[1] Mokhtar S Bazaraa, John J Jarvis, and Hanif D Sherali, *Linear programming and network flows*, John Wiley & Sons, 2011.

[2] John E Beasley, *An algorithm for set covering problem*, European Journal of Operational Research **31** (1987), no. 1, 85–93.

[3] John R Birge and Francois Louveaux, *Introduction to stochastic programming*, Springer Science & Business Media, 2011.

[4] Leonardo Caggiani and Michele Ottomanelli, *A dynamic simulation based model for optimal fleet repositioning in bike-sharing systems*, Procedia-Social and Behavioral Sciences **87** (2013), 203–210.

[5] Michael Chester, *Neural networks: a tutorial*, Prentice-Hall, Inc., 1993.

[6] Claudio Contardo, Catherine Morency, and Louis-Martin Rousseau, *Balancing a dynamic public bike-sharing system*, Vol. 4, Cirrelt Montreal, 2012.

[7] Nicholas Fournier, Eleni Christofa, and Michael A Knodler Jr, *A sinusoidal model for seasonal bicycle demand estimation*, Transportation research part D: transport and environment **50** (2017), 154–169.

[8] Daniele Gammelli, Yihua Wang, Dennis Prak, Filipe Rodrigues, Stefan Minner, and Francisco Camara Pereira, *Predictive and prescriptive performance of bike-sharing demand forecasts for inventory management*, Transportation Research Part C: Emerging Technologies **138** (2022), 103571.

[9] Juan Carlos García-Palomares, Javier Gutiérrez, and Marta Latorre, *Optimizing the location of stations in bike-sharing programs: A gis approach*, Applied Geography **35** (2012), no. 1-2, 235–246.

[10] Mahsa Ghandehari, Vahid Hamidi Pouyandeh, Mohammad Hossein Moshref Javadi, et al., *Locating of bicycle stations in the city of isfahan using mathematical programming and multi-criteria decision making techniques*, International Journal of academic research in accounting, Finance and Management Sciences **3** (2013), no. 4, 18–26.

[11] R. Hansch, *Handbook of random forests: Theory and applications for remote sensing*, Computer Vision, World Scientific Publishing Company Pte Limited, 2019.

[12] Alan J King, *Modeling with stochastic programming*, Springer, 2012.

[13] Nikolaos Kourentzes, Juan R Trapero, and Devon K Barrow, *Optimising forecasting models for inventory planning*, International Journal of Production Economics **225** (2020), 107597.

[14] Karim Labadi, Taha Benarbia, Jean-Pierre Barbot, Samir Hamaci, and Abdelhafid Omari, *Stochastic petri net modeling, simulation and analysis of public bicycle sharing systems*, IEEE Transactions on Automation Science and Engineering **12** (2014), no. 4, 1380–1395.

[15] G. Laporte, S. Nickel, and F.S. da Gama, *Location science*, Springer International Publishing, 2015.

[16] Jon Lee and Sven Leyffer, *Mixed integer nonlinear programming*, Vol. 154, Springer Science & Business Media, 2011.

[17] Jenn-Rong Lin and Ta-Hui Yang, *Strategic design of public bicycle sharing systems with service level constraints*, Transportation research part E: logistics and transportation review **47** (2011), no. 2, 284–294.

[18] Jenn-Rong Lin, Ta-Hui Yang, and Yu-Chung Chang, *A hub location inventory model for bicycle sharing system design: Formulation and solution*, Computers & Industrial Engineering **65** (2013), no. 1, 77–86.

[19] Lei Lin, Zhengbing He, and Srinivas Peeta, *Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach*, Transportation Research Part C: Emerging Technologies **97** (2018), 258–276.

[20] Jingyuan Liu, Qiao Li, Meng Qu, Weiwei Chen, Jingyuan Yang, Hui Xiong, Hao Zhong, and Yanjie Fu, *Station site optimization in bike sharing systems*, 2015 ieee international conference on data mining, 2015, pp. 883–888.

[21] François V Louveaux, *Discrete stochastic location models*, Annals of Operations research **6** (1986), 21–34.

[22] Eric Hsueh-Chan Lu and Zhan-Qing Lin, *Rental prediction in bicycle-sharing system using recurrent neural network*, IEEE Access **8** (2020), 92262–92274.

[23] Richard Mix, Ricardo Hurtubia, and Sebastián Raveau, *Optimal location of bike-sharing stations: A built environment and accessibility approach*, Transportation Research Part A: Policy and Practice **160** (202206), 126–142.

[24] D.C. Montgomery, E.A. Peck, and G.G. Vining, *Introduction to linear regression analysis*, Wiley Series in Probability and Statistics, Wiley, 2013.

[25] David R Morrison, Sheldon H Jacobson, Jason J Sauppe, and Edward C Sewell, *Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning*, Discrete Optimization **19** (2016), 79–102.

[26] Izaak Neutelings, *Neural networks*, 2023.

[27] Andreas Nikiforiadis, Georgia Aifadopoulou, Josep Maria Salanova Grau, and Neofytos Boufidis, *Determining the optimal locations for bike-sharing stations: methodological approach and application in the city of thessaloniki, greece*, Transportation Research Procedia **52** (2021), 557–564. 23rd EURO Working Group on Transportation Meeting, EWGT 2020, 16-18 September 2020, Paphos, Cyprus.

[28] Y.L. Pavlov, *Random forests*, De Gruyter, 2019.

[29] Tal Raviv and Ofer Kolka, *Optimal inventory management of a bike-sharing station*, Iie Transactions **45** (2013), no. 10, 1077–1093.

[30] Tal Raviv, Michal Tzur, and Iris A Forma, *Static repositioning in a bike-sharing system: models and solution approaches*, EURO Journal on Transportation and Logistics **2** (2013), no. 3, 187–229.

[31] Daniel Romm, Priyanka Verma, Elizabeth Karpinski, Tracy L Sanders, and Grant McKenzie, *Differences in first-mile and last-mile behaviour in candidate multi-modal boston bike-share micromobility trips*, Journal of transport geography **102** (2022), 103370.

[32] Soheil Sohrabi and Alireza Ermagun, *Dynamic bike sharing traffic prediction using spatiotemporal pattern detection*, Transportation research part D: transport and environment **90** (2021), 102647.

[33] Patrick Vogel, Torsten Greiser, and Dirk Christian Mattfeld, *Understanding bike-sharing systems using data mining: Exploring activity patterns*, Procedia - Social and Behavioral Sciences **20** (2011), 514–523. The State of the Art in the European Quantitative Oriented Transportation and Logistics Research – 14th Euro Working Group on Transportation 26th Mini Euro Conference 1st European Scientific Conference on Air Transport.

[34] Bo Wang and Inhi Kim, *Short-term prediction for bike-sharing service using machine learning*, Transportation research procedia **34** (2018), 171–178.

[35] Laurence A Wolsey, *Integer programming john wiley & sons*, New York, NY **4** (1998).

[36] Yu-chen Wu and Jun-wen Feng, *Development and application of artificial neural network*, Wireless Personal Communications **102** (2018), 1645–1656.

[37] Zhou Wu, Junjun Wu, Yuguang Chen, Kai Liu, and Liang Feng, *Network rebalance and operational efficiency of sharing transportation system: Multi-objective optimization and model predictive control approaches*, IEEE Transactions on Intelligent Transportation Systems **23** (2022), no. 10, 17119–17129.

[38] X. Yan, *Linear regression analysis: Theory and computing*, World Scientific Publishing Company Pte Limited, 2009.