RCMVis: A Visual Analytics System for Route Choice Modeling

DongHwa Shin, Jaemin Jo, Bohyoung Kim, Hyunjoo Song, Shin-Hyung Cho, and Jinwook Seo

Abstract—We present RCMVis, a visual analytics system to support interactive Route Choice Modeling analysis. It aims to model which characteristics of routes, such as distance and the number of traffic lights, affect travelers' route choice behaviors and how much they affect the choice during their trips. Through close collaboration with domain experts, we designed a visual analytics framework for Route Choice Modeling. The framework supports three interactive analysis stages: exploration, modeling, and reasoning. In the exploration stage, we help analysts interactively explore trip data from multiple origin-destination (OD) pairs and choose a subset of data they want to focus on. To this end, we provide coordinated multiple OD views with different foci that allow analysts to inspect, rank, and compare OD pairs in terms of their multidimensional attributes. In the modeling stage, we integrate a k-medoids clustering method and a path-size logit model into our system to enable analysts to model route choice behaviors from trips with support for feature selection, hyperparameter tuning, and model comparison. Finally, in the reasoning stage, we help analysts rationalize and refine the model by selectively inspecting the trips that strongly support the modeling result. For evaluation, we conducted a case study and interviews with domain experts. The domain experts discovered unexpected insights from numerous modeling results, allowing them to explore the hyperparameter space more effectively to gain better results. In addition, they gained OD- and road-level insights into which data mainly supported the modeling result, enabling further discussion of the model.

Index Terms—Route choice modeling, urban planning, trajectory data, origin-destination, visual analytics

1 Introduction

In transportation engineering, Route Choice Modeling (RCM) is an analysis method used to understand travelers' perceptions of road characteristics and to predict traffic conditions on routes with given characteristics [1]. By developing a quantitative model based on travelers' route choice behaviors, researchers can gain insights into how and why people take a specific route. RCM decides which road characteristics should be given higher priority for road network design projects; for example, if it is found that bicycle riders prefer a route with a gentle slope to a short route, civil engineers can use this finding in designing bicycle lanes. Furthermore, RCM analysis also allows researchers to quantitatively evaluate the effectiveness of the bike lane pavement in advance.

However, we found that RCM researchers resort to an ad hoc or improvised solution to conduct route choice modeling by combining general-purpose systems; for example, they first obtain an overview of route choice behaviors using general geographic information systems (GIS), such as ArcGIS [2] and QGIS [3], and then use Python or R scripts to clean the data and build a model. However, such an ad hoc combination of multiple general-purpose tools does not support RCM analysis effectively, especially when researchers form hypotheses to test in an early stage of analysis. In such analysis, they need to repeatedly select a subset of data with a certain filtering condition (e.g., temporal or spatial) and slightly edit the scripts to find meaningful patterns, which requires tremendous time and effort.

To resolve these issues, we present RCMVis, an interactive visual analytics system to streamline RCM analysis with a three-stage analysis framework. To identify the challenges researchers confront every day and inform the design process, we collaborated with three domain experts in the urban planning field for six months: one postdoctoral researcher (P1) and two graduate researchers (P2 and P3). After domain situation analysis and task abstraction, we suggest an interactive analysis pipeline consisting of three stages: exploration, modeling, and reasoning. In the exploration stage, we enable users to explore and filter movement data to decide targets for RCM. Then, in the modeling stage, users conduct modeling on the targets with multiple hyperparameter sets

- *DongHwa Shin and Jinwook Seo are with the Department of Computer Science and Engineering, Seoul National University, Seoul 08826, South Korea. E-mail: dhshin@hcil.snu.ac.kr, jseo@snu.ac.kr.*
- *Jaemin Jo is with the College of Computing and Informatics, Sungkyunkwan University, Suwon, Gyeonggi-do 16419, South Korea. E-mail: jmjo@skku.edu.*
- *Bohyoung Kim is with the Division of Biomedical Engineering, Hankuk University of Foreign Studies, Seoul 02450, South Korea. E-mail: bkim@hufs.ac.kr.*
- *Hyunjoo Song is with the School of Computer Science and Engineering, Soongsil University, Seoul 06978, South Korea. E-mail: hsong@ssu.ac.kr.*
- *Shin-Hyung Cho is with the School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA. E-mail: scho370@gatech.edu.*

Manuscript received 21 Mar. 2021; revised 13 Sept. 2021; accepted 17 Nov. 2021. Date of publication 1 Dec. 2021; date of current version 31 Jan. 2023.
This work was supported in part by the National Research Foundation of Korea (NRF) grants funded by the Korea government (MSIT) under Grants NRF-2019R1A2C2089062 and NRF-2019R1A2C1088900, and in part by the Hankuk University of Foreign Studies Research Fund.
(Corresponding authors: Jaemin Jo and Jinwook Seo.)
Recommended for acceptance by A. Endert.
Digital Object Identifier no. 10.1109/TVCG.2021.3131824

and identify patterns by comparing them. In the reasoning stage, users perform a data-level analysis of the selected model by exploring movement records that explain the modeling result well. To this end, we design novel visualizations and interactions to effectively support each stage and streamline the whole analysis process. We evaluate RCMVis through a case study with two domain experts using a large bicycle travel dataset from a public bicycle-sharing system of the Seoul Metropolitan Region.

The contributions of this paper are:

1) Design and development of RCMVis, a visual analytics system with a three-stage interactive modeling framework for effective route choice modeling,
2) Identification and abstraction of the domain situation of route choice modeling analysis, and
3) Evaluation of RCMVis through a case study of a real-world bicycle travel dataset.

2 Related Work

2.1 Route Choice Modeling

RCM aims to explain and predict a route choice probability among a choice set with two or more routes. For example, RCM is appropriate for answering the following question: *when driving from LA to Seattle, which route is preferred and why?* A literature survey of RCM [1] divides it into two parts: choice set generation and model estimation.

Choice set generation is a step for generating a discrete choice set for decision-makers. For example, *what are the routes, and how many are there from LA to Seattle?* Traditional approaches derive a choice set based on a road network structure. These include k-shortest paths [4], labeling approach [5], link elimination [4]. Since these methods solely consider properties of road networks, false positive (e.g., generating a non-realistic route) or false negative (e.g., omitting a probable route) errors may be likely to occur [6].

As a remedy to these limitations, recent studies adopt a data-driven method using observed routes when generating a choice set [6], [7]. The advance of global positioning system (GPS) technology made it easy to collect actual traveling routes of individuals and opened an opportunity to utilize these revealed preference (RP) for choice set generation. In RCMVis, we adopt on RP-based k-medoids clustering method, which is actively studied by our collaborators, to generate k alternative routes from observed routes. With our visual analytics approach, we explored various characteristics of the generation techniques based on the k-medoids method.

In discrete choice modeling, a general framework to which RCM belongs, logit-based models such as multinomial logit (MNL) or nested logit (NL) are commonly adopted when estimating the model parameters. However, when it comes to RCM, both MNL and NL are not appropriate because of the model's independence of irrelevant alternatives (IIA) property [8]. In other words, MNL and NL require that candidate items should not be correlated with each other. However, in most cases, routes on a road network may overlap with each other to some extent. In this specific context, path-size logit (PSL) [8] is widely used to deal with the similarities between candidate routes using a term called path size. In that sense, we adopt PSL for estimating a model.

There are well-known tools that can perform route choice modeling. NLOGIT [9] is a commercial software program for choice modeling, which supports a GUI interface and can conduct an analysis with multiple OD pairs. Although NLOGIT widely supports a variety of choice models and their variations, it only provides basic charts for showing the modeling results and does not support map-based visualization; hence, users cannot explore spatial distributions of travel data. The transport planning software called Emme [10] visualizes a spatial overview of travel data with map interfaces and provides a choice modeling component. However, Emme does not provide a means to comprehend the modeling result other than showing the statistics of the model; thus, users might find it challenging to gain deeper insight into the modeling result.

2.2 Trajectory Visual Analytics for Urban Planning

A trajectory is a common form of movement description consisting of location coordinate information recorded at a specific time interval [11]. There are already many existing studies analyzing trajectory data with visual analytics, and researchers can get a sense of the research history and future directions through survey papers [11], [12], [13], [14], [15], [16].

There have been many attempts to solve urban traffic problems, such as traffic surveillance [17], [18], [19], microscopic pattern discovery [20], [21], [22], [23], optimal pattern finding [24], [25], accessibility modeling [26], [27] and route choice behavior modeling [28] with a trajectory visual analytics. Lee *et al.* [19] visualize traffic congestion with a novel visualization called Volume-Speed Rivers, with congestion forecasting results from the Long Short-Term Memory (LSTM) model. Wang *et al.* [18] utilize taxi GPS trajectories to show traffic jam conditions over time and propagation graphs to understand how traffic jams are propagated in a road network. T-Watcher [17] provides visualizations of trajectories at three different levels, including a region, a road, and individual vehicles, to effectively monitor traffic conditions. Like traffic congestion analysis, our route choice model can predict an amount of traffic for given OD pairs and routes. However, RCM differs in that it determines the probability that travelers will choose a specific route under the clear condition that an origin, destination, and route choice set are defined. Therefore, RCM mainly focuses on understanding a traveler's perception of route characteristics rather than conducting macro-level traffic analysis across road networks.

For microscopic pattern discovery, TripVista [20] focuses on the traffic pattern of a single road intersection. They provide ring-style sliders to select data and show a ThemeRiver-style [29] visualization to help users explore microscopic movements of vehicles. Liu *et al.* [21] facilitate the exploration of route diversity between origin and destination in terms of the spatial and temporal information of routes. Zeng *et al.* [22] suggest an interchange circos diagram to visualize traffic patterns on each junction of a road network. Wang *et al.* [23] provide a sketch-based interface to support road-level trajectory querying with multiple coordinated views to understand multiple aspects of traffic. Like the aforementioned works, RCMVis enables conducting a street-level analysis by spatially filtering overall travel data into small regional data to figure out route choice behaviors within a small, specific region.

A study of visual analytics for RCM was reported by Lu *et al.* [28], and they support a visual analytics pipeline for route choice modeling with trajectory filtering. However, they assumed that only a single OD pair could be of interest at once. In practice, regional movement data often comprise multiple OD pairs, and RCM researchers have to consider all of them to model the route choice behavior of the area. Further, researchers make many attempts to find models that better explain the route choice behavior using a variety of algorithms or tuning hyperparameters. After finding the best-fit modeling result, researchers seek to determine the result's implications at the data level by identifying which movements primarily support this result. By reflecting on the aforementioned analysis scenarios, we present a novel three-stage analysis framework to support more realistic analytic tasks than existing RCM tools.

3 Background

During our design study process, we had weekly meetings with three domain experts for six months. Through this tight collaboration with them, we were able to gain a deep understanding of the domain situation and RCM analysis in general. We identified their existing analysis procedures and challenges in processing the data with their tools and interpreting and reasoning the modeling results. This section elaborates on the background of our work in terms of the domain situation, data abstraction, and task abstraction, in accordance with the visualization design framework presented by Brehmer and Munzner [30], [31].

3.1 Domain Situation Analysis

We recognized that the analysis process of the experts could be divided into three conceptual stages. Although the experts did not explicitly mention this division, they strongly agreed with it when we introduced our three-stage framework. We summarize the domain situation using an illustrative example where Jenny, an urban planning researcher, performs route choice modeling. The goal of Jenny's analysis is to identify which factors are primarily considered by bicycle riders when choosing their travel route, which is a common analysis scenario for RCM analysts.

Exploration Stage. She loads the data to visualize it on a map through a GIS. She first explores the geographical distribution of bicycle traffic and discovers some prominent areas with heavy traffic. Then, she wonders what these patterns will be like during the peak hour. However, since the GIS does not support interactive filtering, she runs R scripts to keep only the travels of her interest in the data and loads the filtered data again to visually inspect patterns.

Modeling Stage. After exploring the characteristics of the data, Jenny decides to conduct RCM with this data to determine riders' route choice behaviors during the peak hour. She tunes a set of hyperparameters of the algorithms for choice set generation and model estimation, which are the two key parts of route choice modeling. Since the quality of a model is greatly affected by its hyperparameters, she experiments with various sets of hyperparameters and inspects the results to compare them. She eventually finds a set of hyperparameters that results in a high goodness of fit.

Reasoning Stage. She decides to interpret the model estimated using the aforementioned set of hyperparameters. To understand the route choice behavior of bicycle riders, she inspects the statistical significance of each route attribute in the model. A positive coefficient for an attribute (e.g., distance) means that the routes with higher values on that attribute are more preferred by the riders. She combines the modeling result (i.e., significance), her domain knowledge, and geographical information to gain higher-level knowledge.

Limitations of the Previous Approaches. The foremost limitation in Jenny's data exploration with existing tools is that the entire exploratory analysis was fragmented, so she needed to go back and forth between the GIS and data manipulation scripts. Furthermore, although it was necessary to explore a hyperparameter space to obtain a good model in the modeling stage, this task was tedious and inefficient, as it was done manually without the support of interactive interfaces. Finally, in the reasoning stage, she needed to combine the findings from various sources to elicit knowledge, but this task would be cognitively overwhelming if done without the aid of external representations.

3.2 Data Preprocessing and Abstraction

We used a real-world bicycle trip dataset from the Seoul bike-sharing system [33]. The dataset included information on 210 K trips that took place in March 2018. Each trip consisted of GPS-tracked path records (recorded every minute), origin and destination stations, rental and return times, travel distance, and duration.

3.2.1 Terminology

For a clear understanding of RCM analysis, we define important terms as follows:

- *Station:* A physical facility where riders can rent or return a bicycle.
- *Route:* A path between an origin station and and a destination station (OD) pair. There can be multiple routes between the same OD pair.
- *Trip:* A movement record of an individual rider. It consists of an OD pair and a route taken.
- *Trip Set:* A set of trips. Multiple riders can move between the different origins and destinations, and their trips constitute a trip set.
- *Station Attributes:* The attributes that a single station can have. All the station attributes are listed in Table 1.
- *OD Attributes:* The attributes that a single OD pair can have. All the OD attributes are listed in Table 2.
- *Route Attributes:* The attributes that a single route can have. All the route attributes are listed in Table 3.
- *Model Instance:* A result of modeling the trip set. It mainly refers to model statistics and estimated coefficients of the route attributes. Detailed information about the modeling process and its result is provided in Section 4.

TABLE 1
Station Attributes

Station Attribute	Description
ID	unique ID of a station
Type	user-designated type of a station
In-flow OD Pairs	set of OD pairs that have this station as a destination (i.e., incoming OD pairs)
In-flow Traffic	sum of all the in-flow OD pairs' traffic
Out-flow OD Pairs	set of OD pairs that have this station as an origin (i.e., outgoing OD pairs)
Out-flow Traffic	sum of all the out-flow OD pairs' traffic
Total Traffic (i.e., *Traffic*)	sum of the in- and out-flow traffic (i.e., the number of all trips associated with this station)

3.2.2 Data Cleaning

We found that the raw data had erroneous records, such as trips with missing fields. To clean the data, we referred to Wang *et al.* [18] and modified their cleaning criteria. We filtered out the trips that met one of the following conditions:

- *Missing Fields:* Trips with a missing field.
- *Out of Bounds:* Trips that contain GPS records outside the boundaries of Seoul [127.1861E, 126.7686E] x [37.4213N, 37.6929N].
- *Same O/D:* Trips whose origin and destination are the same, not being of interest in RCM analysis.
- *High Speed:* Trips that have GPS records of riding farther than 0.5 km in a minute.
- *Long Distance from Origin:* Trips whose distance between the origin and the first GPS record is over 0.5 km.
- *Long Distance to Destination:* Trips whose distance between the last GPS record and the destination is over 0.5 km.

3.2.3 Map Matching

After cleaning the trip dataset, we matched the path records with the street network of Seoul to reduce possible noise in GPS records. We used a well-known map matching algorithm, ST-Matching [34], to convert the raw path records to road network-bounded routes. For the matching process, we used the OpenStreetMap (OSM) [35] road network dataset. The OSM road network is mainly comprised of nodes and segments. A node is a single point in space defined by its latitude and longitude. A segment is a straight line between exactly two nodes. With nodes and segments, we can represent and deal with all roads in the road network. We used this road information for map matching. Among seven principal types of roads in OSM, we chose to use only the primary, secondary, tertiary, and residential types of roads after consultation with our domain experts.

TABLE 2
OD Attributes

OD Attribute	Description
OD Type Pair	pair of origin and destination station type
Number of Trips (i.e., *Traffic*)	number of trips in an OD pair
Number of Routes	number of routes in an OD pair
OD Distance	straight distance between origin and destination stations
Nonparametric Skew	skewness measure for route attribute distributions within an OD pair defined for each route attribute [32]
Mean Silhouette Score	mean of all trips' silhouette scores in an OD pair; details are described in Section 4.1
Mean Estimation Contribution Score	mean of all trips' estimation contribution scores (*ECS*) defined for each route attribute; details are described in Section 4.4

TABLE 3
Route Attributes

Route Attribute	Description
Route Distance	distance of a route in kilometer
Number of Intersections	average number of intersections per kilometer on a route
Number of Traffic Lights	average number of traffic lights per kilometer on a route
Primary Road Ratio	ratio of primary road on a route
Secondary Road Ratio	ratio of secondary road on a route
Tertiary Road Ratio	ratio of tertiary road on a route
Residential Road Ratio	ratio of residential road on a route
Bike Lane Ratio	ratio of bicycle lane on a route (independent to road type)
Maximum Upslope	maximum gradient of upslopes (%)
Average Upslope	average gradient of upslopes (%)
Maximum Downslope	maximum gradient of downslopes (%)
Average Downslope	average gradient of downslopes (%)
Average Slope	average gradient of all slopes (%)
Path Size	correction term of PSL model; the formula can be found in supplementary materials, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TVCG.2021.3131824

3.2.4 Collection of Route Attributes

To include routes in RCM analysis, the characteristics of routes must be identified. Table 3 summarizes the route attributes we collected and used in the RCM analysis. The route attributes are those that our domain experts have been interested in and actively studied. The data source and detailed processing procedures can be found in the supplementary materials, available online.

3.3 Task Analysis and Abstraction

From the current practice of domain experts, we have established the following important tasks in the RCM analysis. The tasks were iteratively revised through the iterative design process with our domain experts. We used the visualization design framework of Brehmer and Munzner [30], [31] to describe our tasks; each task is described in the form of [*Action* → *Target*].

Exploration Stage (E)

- *E1: Summarize Trip Set*. Users analyze data with specific conditions, such as trips that took place during weekend or peak time, rather than the entire data. Thus, they apply filters to *summarize* the trip set [*Summarize* → *Trip Set*].
- *E2: Explore Geographical Distribution of Trips*. Users explore how riders' trips are geographically distributed, especially areas, flows, or roads with heavy traffic [*Explore* → *Feature*].

- *E3: Identify Attribute Distribution of Chosen Routes*. Users inspect distributions of chosen routes' attribute values in each OD pair [*Identify* → *Distribution*]. Thus, they can obtain an overview of riders' perceptions of the route attributes and how biased the chosen attribute values are before modeling.

Modeling Stage (M)

- *M1: Perform Modeling with Different Sets of Hyperparameters*. Users perform choice set generation and model estimation with various sets of hyperparameters to find a meaningful model instance with a high goodness of fit [*Derive* → *Model Instance*].
- *M2: Obtain an Overview of Model Instances*. Users obtain an overview of many different model instances to identify their common or different patterns [*Summarize* → *Model Instance*].
- *M3: Compare Model Instances*. Users compare statistics and estimates between model instances to choose a model instance for explaining route choice behaviors [*Compare* → *Model Instance*].

Reasoning Stage (R)

- *R1: Discover Route Choices Contributing to an Estimation Result*. Users discover trips and OD pairs that follow the model's estimated coefficients well. For example, when a model instance has a negative coefficient for *Route Distance*, trips with a relatively short travel distance within their OD pair are deemed to contribute to the estimation result. As users investigate such route choices, they aim to gain a deeper understanding of the model and better explain the route choice behaviors [*Summarize* → *Trip Set*].
- *R2: Re-estimate to Obtain Better Fitting Model Instances*. An essential premise of RCM is that all individual route choices have rationality. Therefore, if users encounter route choices that seem irrational in reasoning, they remove these trips or OD pairs and re-estimate the model. Their goal is to get a refined model instance that is well fitted to the data and better reflects riders' perceptions [*Derive* → *Model Instance*].

4 Route Choice Model

The general process of route choice modeling is twofold: choice set generation and model estimation. In this section, we will briefly describe the concept of each step and introduce the methods used in our study.

4.1 Choice Set Generation

In the context of RCM, a choice set is a set of route options a rider can choose when traveling from origin to destination. To generate a choice set, we adopt a newly emerging approach that utilizes the routes actually chosen by riders (i.e., observed routes). However, the number of observed routes can be too large for modeling. Our domain experts mentioned that riders tend to consider just several routes with distinct features rather than considering the entire space of possible routes. Based on the discussions with the experts, we decided to use the k-medoids clustering algorithm [36] that they actively use to group similar routes. The illustrative example of a choice set generation process is shown in Fig. 1.

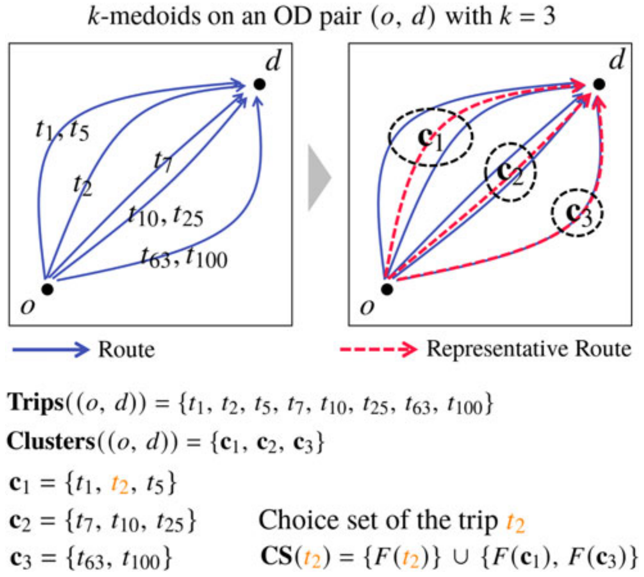

k-medoids on an OD pair (o, d) with $k = 3$
d
t_1, t_5
t_2
t_7
t_{10}, t_{25}
t_{63}, t_{100}
o
d
c_1
c_2
c_3
o
Route
Representative Route
$\mathbf{Trips}((o, d)) = \{t_1, t_2, t_5, t_7, t_{10}, t_{25}, t_{63}, t_{100}\}$
$\mathbf{Clusters}((o, d)) = \{\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3\}$
$\mathbf{c}_1 = \{t_1, t_2, t_5\}$
$\mathbf{c}_2 = \{t_7, t_{10}, t_{25}\}$
$\mathbf{c}_3 = \{t_{63}, t_{100}\}$
Choice set of the trip t_2
$\mathbf{CS}(t_2) = \{F(t_2)\} \cup \{F(\mathbf{c}_1), F(\mathbf{c}_3)\}$

Fig. 1. Illustrative example of a choice set generation process for the trip t_2 traveling an OD pair (o, d).

4.1.1 Clustering

We define $\mathbf{T} = \{t_i \,|\, i = 1, 2, \ldots, n\}$ as a set of trips, where n is the number of trips, and t_i refers to the trip with index i. Let $\mathbf{P} = \{p \,|\, p = (\, o(t_i),\ d(t_i)\,),\ i = 1, \ldots, n\}$ be a set of OD pairs, where $o(t_i)$ is an origin station of t_i, $d(t_i)$ is a destination station of t_i, and p is a pair of origin and destination stations (i.e., OD pair). Hereafter, we will use p to denote an arbitrary OD pair, $(o,\ d) \in \mathbf{P}$. Note that the size of $\mathbf{P}$ (i.e., $|\mathbf{P}|$) is not always equal to n since trips with the same OD pair may exist. We perform k-medoids only on trips having the same OD pair. Thus, we define $\mathbf{Trips}(p) = \mathbf{Trips}((o,\ d)) = \{t_i \,|\, t_i \in \mathbf{T},\ o(t_i) = o,\ d(t_i) = d\}$, which indicates a set of trips having the same OD pair $p = (o,\ d) \in \mathbf{P}$. Accordingly, we need to perform k-medoids on $\mathbf{Trips}(p)$ for each OD pair $p \in \mathbf{P}$.

To quantify the distance between trips' routes, we use two types of distance: *overlap distance* takes the overlapping segments of the two trips' routes into account, and *attribute distance* only considers the route attributes (Table 3) of the two routes. There are four overlap distances: *Overlapping Distance*, *Overlapping Intersection*, *Overlapping Traffic Light*, and *Overlapping Bike Lane Ratio*. The values of these four overlap distances are ratios; for example, *Overlapping Traffic Light* is the ratio of the number of traffic lights on overlapping segments to the number of traffic lights on the route having the shorter *Route Distance* among the two routes.

Meanwhile, the attribute distance is computed by the euclidean distance between a certain route attribute of two trips. There are 13 attribute distances for the route attributes shown in Table 3 excluding *Path Size*, as it is derived using a generated choice set and is only used for the model estimation step. In summary, the 17 distances mentioned above can be used selectively, and the sum of the chosen distances is used as a distance measure for k-medoids.

Fixing the number of clusters k to a specific number equally for all OD pairs may not be effective because it is likely that the number of representative routes could be different for each OD pair. Therefore, we provide two types of k: the fixed (k) and bounded ($k{*}$) types. The bounded type automatically finds an optimal value of k that best describes the routes between an OD pair. The use of such an optimization is indicated as a star ($*$); for example, $k = 5*$ means that the clustering algorithm will test different k values, ranging from 2 to 5, to cluster the trips in $\mathbf{Trips}(p)$ and choose the clustering result with the best quality. As the clustering quality measure we adopt the *silhouette score* [37]. To compare the results, we use the mean silhouette scores of the trips in $\mathbf{Trips}(p)$ since the silhouette score is obtained for each trip. The same goes for evaluating the overall clustering quality of the set $\mathbf{T}$.

After k-medoids on $\mathbf{Trips}(p)$ for every OD pair $p \in \mathbf{P}$ is done, we obtain k clusters of trips. We define the clustering result for $\mathbf{Trips}(p)$ as $\mathbf{Clusters}(p) = \{\mathbf{c}_j \mid j = 1, 2, \ldots, k\}$, where $\mathbf{c}_j$ is one of the k clusters.

4.1.2 Choice Set

In this section, we introduce how we define a choice set $\mathbf{CS}(t_i)$ for each trip t_i traveling an OD pair p using the results of k-medoids $\mathbf{Clusters}(p)$.

A choice set contains each route in the form of a feature vector. We define a feature vector $F(t_i) \in \mathbb{R}^{|\boldsymbol{\sigma}|}$ representing route attribute values of the route taken by the trip t_i. $\boldsymbol{\sigma}$ is a user-designated subset of the route attributes (Table 3). Our interface supports users to interactively choose the set $\boldsymbol{\sigma}$. One dimension of $F(t_i)$ corresponds to the value of a route attribute in $\boldsymbol{\sigma}$ for t_i.

To extend the concept of a feature vector for a trip to a cluster $\mathbf{c}$ (from k-medoids clustering), we define a cluster feature vector $F(\mathbf{c}) \in \mathbb{R}^{|\boldsymbol{\sigma}|}$ as follows:

$$F(\mathbf{c}) = \frac{\sum_{t \in \mathbf{c}} F(t)}{|\mathbf{c}|}, \tag{1}$$

which is the mean of route attribute values of all trips t in the cluster $\mathbf{c}$. We call an hypothetical route having $F(\mathbf{c})$ as its feature vector a representative route of a cluster $\mathbf{c}$ (Fig. 1).

A choice set is defined for each trip $t_i \in \mathbf{T}$. We specify the choice set of the trip t_i (i.e., $\mathbf{CS}(t_i)$) as the set of $F(\mathbf{c})$ for all k trip clusters $\mathbf{c} \in \mathbf{Clusters}(\,(o(t_i),\ d(t_i))\,)$, where $(\,o(t_i),\ d(t_i)\,)$ is the OD pair of the trip t_i. However, we replace $F(\mathbf{c})$ with $F(t_i)$ only for the $\mathbf{c}$ containing the trip t_i. This is because we already know that the rider traveled the trip t_i's route among all the routes of the trip cluster $\mathbf{c}$. We define the choice set $\mathbf{CS}(t_i)$ as follows:

$$\mathbf{CS}(t_i) = \{F(t_i)\} \cup \{F(\mathbf{c}) \mid \mathbf{c} \in \mathbf{Clusters}(\,(o(t_i),\ d(t_i))\,),\ t_i \notin \mathbf{c}\}. \tag{2}$$

That is to say, $\mathbf{CS}(t_i) \in \mathbb{R}^{k \times |\boldsymbol{\sigma}|}$ contains the trip t_i's feature vector $F(t_i)$, and $(k-1)$ cluster feature vectors $F(\mathbf{c})$ for all clusters $\mathbf{c}$ in the results of k-medoids on t_i's OD pair except for the one containing t_i.

4.2 Model Estimation

The objective of the model estimation step is to estimate a coefficient for each route attribute. Once the set of route attributes to be estimated $\boldsymbol{\sigma}$ is decided, and choice sets $\mathbf{CS}(t_i)$ for all trips $t_i \in \mathbf{T}$ are generated, the probability of choosing a specific route, called the *route choice probability*, can be computed based on the utility value that riders can obtain when choosing the route. The *utility value* of a feature vector $F \in \mathbb{R}^{|\boldsymbol{\sigma}|}$ is defined as follows:

$$U(F,\ \theta) = F \cdot \theta, \tag{3}$$

where F can be either $F(t_i)$ or $F(\mathbf{c})$, $\theta \in \mathbb{R}^{|\boldsymbol{\sigma}|}$ is the vector of coefficients for each of the route attributes in $\boldsymbol{\sigma}$, and $\cdot$ indicates the dot product. When modeling route choices, it is assumed that a route with a higher utility value is more likely to be chosen. Therefore, the coefficient of each route attribute directly affects the route choice probability. For example, a positive coefficient for the *Primary Road Ratio* indicates that riders are more likely to take routes with a higher ratio of the primary road; however, we do not know the exact coefficient values, so we want to estimate them.

As we adopt PSL model, the probability of choosing the route of t_i given the coefficients θ is specified as follows [8]:

$$f(t_i \mid \theta) = \frac{e^{U(F(t_i),\, \theta)}}{\sum_{F \in \mathbf{CS}(t_i)} e^{U(F,\, \theta)}}, \tag{4}$$

where e is the base of the natural logarithm. Then, the probability of observing all trips of the set $\mathbf{T}$ given the coefficient vector θ is as follows:

$$L = f(t_1,\ t_2,\ \ldots,\ t_n \mid \theta), \tag{5}$$

which is called likelihood. Because PSL model relaxes IIA property by including the term *Path Size* [8], we can assume that the route choices of all trips are independent. Thus, we can express the likelihood as follows:

$$L = f(t_1 \mid \theta) \cdot f(t_2 \mid \theta) \cdot \ldots \cdot f(t_n \mid \theta). \tag{6}$$

The goal is to estimate θ that maximizes L. To this end, we use an optimization method, *maximum likelihood estimation* (MLE) [38]. For ease of computation, MLE maximizes the following logarithm of L:

$$LL = \ln(\,f(t_1 \mid \theta)\,) + \ln(\,f(t_2 \mid \theta)\,) + \cdots + \ln(\,f(t_n \mid \theta)\,), \tag{7}$$

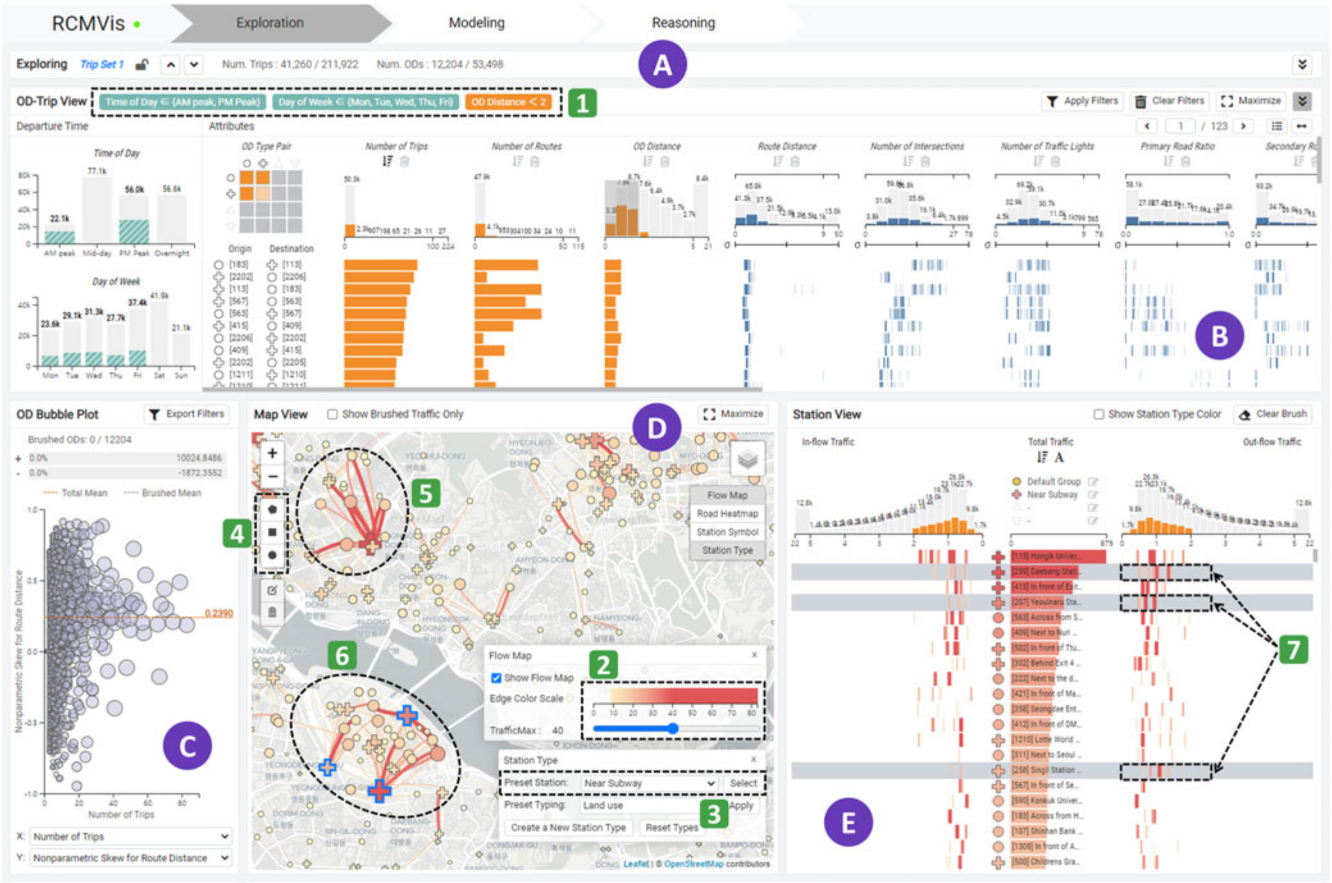
which is called log-likelihood. As a result of MLE, we can get the vector of estimated coefficients $\hat{\theta} \in \mathbb{R}^{|\boldsymbol{\sigma}|}$ that maximizes LL.

Note that $\hat{\theta}$ could be estimated differently depending on which route attributes are included in the model (i.e., elements of the set $\boldsymbol{\sigma}$). Regarding this, our collaborators mentioned that they usually perform many estimation trials by including or excluding specific attributes and compare the results to obtain insight into the route attributes.

4.3 Goodness of Fit

When comparing the quality between models, our domain experts mainly use a measure of goodness of fit. Goodness of fit is an indicator of how well a model fits the data. The $\bar{\rho}^2$ (*rho-squared-bar*), a measure of goodness of fit widely used in route choice modeling, is specified as follows [39], [40]:

$$\bar{\rho}^2 = 1 - \frac{LL_{final} - |\boldsymbol{\sigma}|}{LL_{init}}, \tag{8}$$



Exploring Trip Set 1 Num. Trips : 41,260 / 211,922 Num. ODs : 12,204 / 53,498

OD-Trip View: Time of Day ∈ {AM peak, PM Peak} Day of Week ∈ {Mon, Tue, Wed, Thu, Fri} OD Distance < 2

Brushed ODs: 0 / 12204

X: Number of Trips

Y: Nonparametric Skew for Route Distance

Flow Map ☑ Show Flow Map TrafficMax : 40

Station Type: Preset Station: Near Subway; Preset Typing: Land use

Fig. 2. Interface for the exploration stage. Users deal with a *trip set*, a set of bicycle riders' trips, throughout the entire analysis process. In the exploration stage, users explore the geographical distribution of a trip set and apply filtering conditions to a trip set for preparing the modeling stage. (A) The header of the trip set list shows information about the currently activated trip set, which is called an *active trip set*. (B) The OD-Trip view shows distributions of time, OD, and route attributes and allows users to interactively apply filtering conditions to the active trip set. (C) The OD bubble plot represents an OD pair as a bubble and supports brushing to help users selectively see the OD pairs of their interest. (D) The map view shows both OD-level and road-level geographical distributions of the active trip set in a flow map and a road heatmap, respectively. (E) The station view allows users to rearrange and compare stations and their attributes.

$$LL_{final} = \ln(\, f(t_1 \,|\, \hat{\theta})\,) + \ln(\, f(t_2 \,|\, \hat{\theta})\,) + \cdots + \ln(\, f(t_n \,|\, \hat{\theta})\,)\,, \tag{9}$$

$$LL_{init} = \ln(\, f(t_1 \,|\, \theta_0)\,) + \ln(\, f(t_2 \,|\, \theta_0)\,) + \cdots + \ln(\, f(t_n \,|\, \theta_0)\,)\,, \tag{10}$$

where $|\boldsymbol{\sigma}|$ is the number of elements in $\boldsymbol{\sigma}$, and $\theta_0 \in \mathbb{R}^{|\boldsymbol{\sigma}|}$ is the zero vector indicating that all the route attributes in $\boldsymbol{\sigma}$ have no effect on choosing routes. Since θ_0 makes the utility value U (Equation (3)) to 0, $f(t_i \,|\, \theta_0)$ (Equation (4)) equals to $1/k$ when the k is the fixed type. This makes LL_{init} the constant value, $-n\ln(k)$. Note that LL_{final} (i.e., final log-likelihood) and LL_{init} (i.e., initial log-likelihood) are all negative, so maximizing LL_{final} (i.e., closer to 0) brings $\bar{\rho}^2$ closer to 1. In other words, we can think of better estimation as maximizing the gap between LL_{final} - LL_{init}. $|\boldsymbol{\sigma}|$ is a penalty term that makes $\bar{\rho}^2$ smaller as the number of the route attributes to be estimated increases.

4.4 Estimation Contribution Score

To measure how much an arbitrary trip contributes to yielding the estimated coefficients $\hat{\theta}$, we define the estimation contribution score (ECS). If the route choice probability of the trip t_i (Equation (4)) significantly increases with $\hat{\theta}$ after model estimation, we can say that $\hat{\theta}$ explains the route choice behavior of the trip t_i well. In that sense, we define the ECS of t_i as follows:

$$ECS(t_i) = \ln(\, f(t_i \,|\, \hat{\theta})\,) - \ln(\, f(t_i \,|\, \theta_0)\,). \tag{11}$$

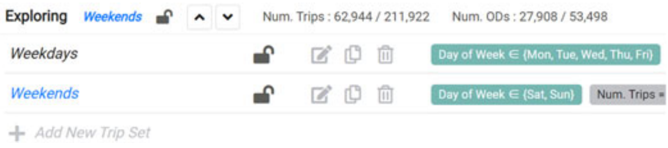
We can think of trips with a larger ECS make greater contributions to estimating θ as $\hat{\theta}$ since those trips contribute to make LL_{final} - LL_{init} larger. The ECS for the specific route attribute $a \in \boldsymbol{\sigma}$ can be defined as follows with the same logic as Equation (11):

$$ECS_a(t_i) = \ln(\, f(t_i \,|\, \hat{\theta})\,) - \ln(\, f(t_i \,|\, \hat{\theta}_{a=0})\,)\,, \tag{12}$$

where $\hat{\theta}_{a=0} \in \mathbb{R}^{|\boldsymbol{\sigma}|}$ is the vector identical to $\hat{\theta}$ except that its coefficient of the route attribute a is zero (i.e., the effect of a for modeling route choices removed). Not only a trip, but we can also measure the ECS of an arbitrary OD pair p. To do this, we take the mean ECS of all trips $t \in \textbf{Trips}(p)$.

5 The RCMVis Design

The RCMVis design is guided by the three analytic stages found during the domain situation analysis: these stages are presented as separate tabs on the header of the interface (Fig. 2), and users can switch between the stages by clicking on the corresponding tab.



Exploring *Weekends* ^ ˅ Num. Trips : 62,944 / 211,922 Num. ODs : 27,908 / 53,498

Weekdays	Day of Week ∈ {Mon, Tue, Wed, Thu, Fri}
Weekends	Day of Week ∈ {Sat, Sun} Num. Trips

+ *Add New Trip Set*

Fig. 3. The trip set list shows a list of trip sets created by users. The name of the active trip set is shown in blue. Each trip set row contains the icons for showing lock status, renaming, copying, and deleting the trip set. To the right of the icons, filtering conditions on the trip set are displayed in badges.

5.1 Exploration Stage

Our domain experts visually explore trips as the first step of RCM analysis. The main goal of the exploration stage is to explore and prepare trip sets for the next stage (i.e., modeling). A trip set is a subset of trips that satisfy specific filtering conditions of interest, such as trips that took place during the weekend or trips where the *Route Distance* is shorter than 2 km.

Users can manage trip sets in a *trip set list* (Figs. 2A and 3). They can activate a trip set by clicking on its name in the trip set list, and hereafter we call an activated trip set an *active trip set*. Activating a trip set is crucial, as the subsequent visualizations and interactions happen on the active trip set; this reflects the practice where domain experts work on one trip set at a time. Trip sets are added initially without filtering conditions (thus, including all trips in the data) but can be adjusted in the OD-Trip view.

In addition to the trip set list, the exploration interface provides the OD-Trip view, an OD bubble plot, a map view, a station view, and a route view. The OD-Trip view (Fig. 2B) allows users to modify filtering conditions applied on the active trip set (Task E1). The other four views were designed for understanding the characteristics (Tasks E2 and E3) of the active trip set, such as OD attributes (OD bubble plot, Fig. 2C), geographical distribution (map view, Fig. 2D), station statistics (station view, Fig. 2E), and individual routes (route view, Fig. 5C).

5.1.1 *OD-Trip View*

The OD-Trip view (Fig. 2B) supports the interactive modification of filtering conditions applied to the active trip set (Task E1). The filtering conditions are represented as badges in the view header (Fig. 2 (1)).

The conditions can be divided into two types: by *departure time* and by *attributes*. The two time bar charts (i.e., two bar charts on the left of the OD-Trip view) summarize the number of trips aggregated by departure time, such as time of day (*AM peak* (from 07:00 to 10:00), *Mid-day* (between *AM* and *PM peak*), *PM peak* (from 17:00 to 20:00), and *Overnight* (between *PM* and *AM peak*)), and day of the week. All these time spans were determined, reflecting domain experts' exploration practice identified during the domain situation analysis.

The attributes panel on the right visualizes the active trip set's OD pairs and associated trips with their attributes. In this panel, a column represents either an OD or route attribute of OD pairs. A column header shows the distribution of the corresponding attribute as a matrix or a histogram. Below the column headers, each row (Fig. 4 (1)) represents an OD pair and its attribute values.

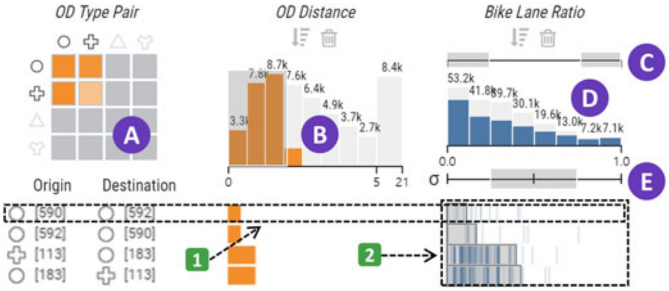


Fig. 4. The attribute panel of the OD-Trip view shows each OD or route attribute as a column. The column header shows (A) the OD type matrix and (B, D) the distribution histograms of an OD (in orange) and route attribute (in blue). Below the header, (1) each row represents an OD pair and it shows the details for the corresponding columns.

In the first column, there is an *OD type matrix* (Fig. 4A). Users can define their own station type and assign it to the stations they want in the map view described in the later section. A station type is represented as a symbol throughout the system, and the system supports up to four types. The matrix row and column represent origin and destination types, respectively. The color saturation of a matrix cell represents the number of trips of all the OD pairs having the corresponding OD type pair. Below the column header, station type symbols and IDs of origin and destination are shown in each row (Fig. 4 (1)).

All the remaining columns represent numerical attributes, and each of them shows an attribute distribution histogram (Figs. 4B and 4D). Users can distinguish between time, OD, and route attribute by color: time as cyan, OD as orange, and route as blue. We apply the identical color scheme to the filter badges in the view header. In each row below the column headers, we visualize an OD attribute value as a horizontal orange bar, but route attribute values are represented as a barcode plot with blue bars (Fig. 4 (2)) since there can be multiple trips in a single OD pair.

The OD-Trip view supports two types of filtering with different targets: an OD, and a trip filtering. The OD filtering inspects all OD pairs in a trip set and filters them out that do not meet the given conditions. Whereas, the trip filtering inspects all trips, and filters out trips. Then, OD pairs with no trips left also get filtered out. The supported filtering conditions are summarized in Table 4.

5.1.2 *OD Bubble Plot*

The OD bubble plot (Fig. 2C) represents each OD pair as a bubble, encoding the number of trips of the OD pair to the area of the bubble. Users can designate two OD attributes, which are mapped to the x- and y-axes using the two dropdown lists at the bottom . Note that OD attributes also include derived statistics of the underlying trips, such as the nonparametric skew (Table 2) of trips' *Bike Lane Ratio* (Fig. 5A), and identifying the distribution of such statistics can give a preliminary view of how a route attribute affects route choice behavior. In addition, the OD bubble plot supports brushing and linking; users can brush on particularly interesting bubbles (Fig. 5 (1)) so that only the corresponding OD pairs remain visible in the map view and the station view.

TABLE 4
Filtering Conditions

Name	Target	Filter By	Visualization	Interaction	Example
OD-Type	OD	OD type pair	OD type matrix (Figure 4(A))	Click on multiple cells (OR)	Leave only OD pairs with *Commercial* type origin and *Residential* type destination
OD-Range		OD attribute value	OD attribute histograms (Figure 4(B))	Brush on a single range	Leave only OD pairs with a number of trips above 20
OD-TripRange		Route attribute value	A top axis of route attribute histograms (Figure 4(C))	Brush on multiple ranges (AND)	Leave only OD pairs that contain both trips with a *Bike Lane Ratio* less than 0.2 and trips with a *Bike Lane Ratio* greater than 0.8 (Figure 4(C))
Trip-Range	Trip	Route attribute value	Route attribute histograms (Figure 4(D))	Brush on a single range	Leave only trips with a *Route Distance* less than 2.0
Trip-StdevRange		Standard deviation of route attribute value	A bottom axis of route attribute histograms (Figure 4(E))	Brush on a single range	Leave only trips fall within the range from -2σ to $+2\sigma$ of a *Bike Lane Ratio* in their own OD pair (Figures 4(E) and 4(2))
Trip-Time		Trip departure time	The two time bar charts (Figure 2(B))	Click on multiple bars (OR)	Leave only trips that occurred during the weekdays, *AM peak*, and *PM peak* hours

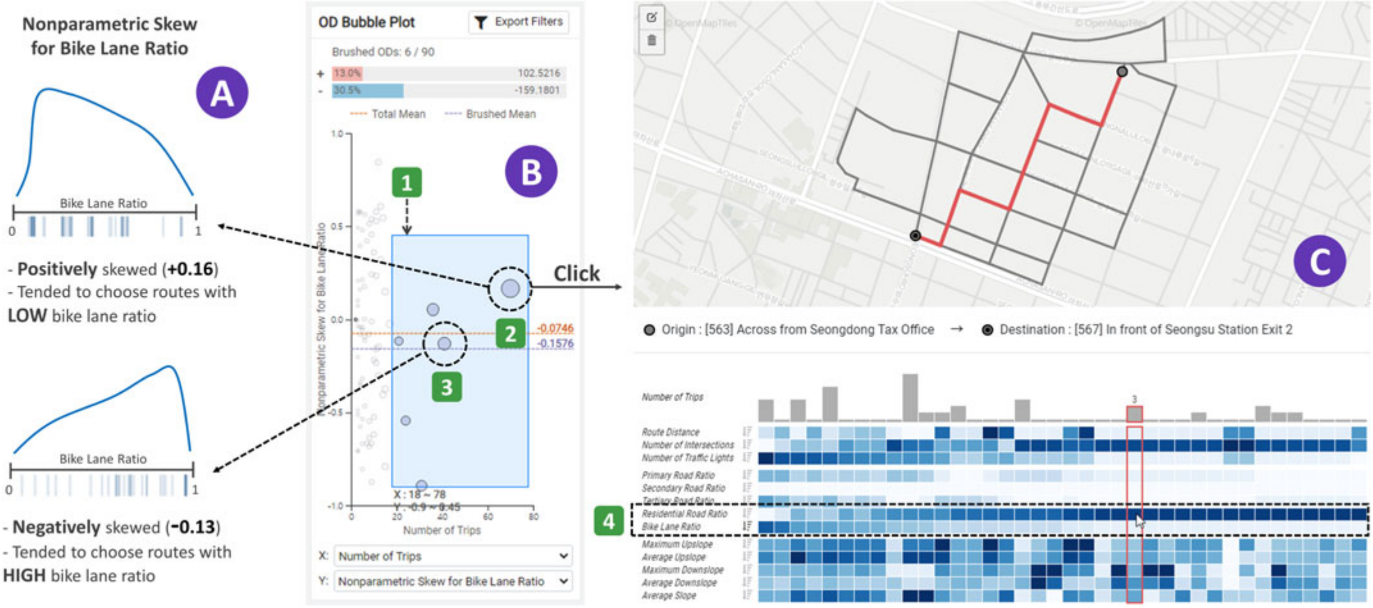


Fig. 5. (A) Illustrative example of the nonparametric skew for *Bike Lane Ratio* of an OD pair. (B) The OD bubble plot represents an OD pair as a bubble. Users can brush on bubbles with the rectangular area (1) by mouse dragging. If users want to figure out the routes between a specific OD pair, they can click the bubble to open the route view. (C) The route view shows the geographical information and the route attributes of a specific OD pair, via the map and the route heatmap, respectively.

5.1.3 Map View

The map view (Fig. 2D) allows users to grasp the geographical distribution of the trips. To this end, the map view shows visual elements of the following targets: station, OD pairs, and road segments (introduced in Section 3.2.3). Every visual element has its own traffic, although the definition of traffic is slightly different for each target. We describe the exact definition of traffic for each target later in this section.

Instead of encoding the traffic directly, we convert each element's traffic into the following weight according to Wood *et al.* [41]:

$$w_{elem} = \left(\frac{Traffic_{elem}}{TrafficMax_{elem}}\right)^{1.5}, \tag{13}$$

where $Traffic_{elem}$ is the traffic of the element, and $TrafficMax_{elem}$ is the maximum traffic among the elements of the current target (i.e., station, OD pair, or road segment). We use the weight w_{elem} because it increases exponentially as the traffic increases; thus, it makes elements with relatively large traffic more prominent than other elements with little traffic. The power 1.5, derived from the empirical experiments, is known to provide the right balance between dominant and less frequent elements [41].

The map view represents each station as a glyph whose size and color redundantly encode the traffic. The traffic of a station indicates the *total traffic* (Table 1), which is the sum of incoming and outgoing (i.e., in- and out-flow) traffic. The shape of a glyph represents its station type as in the OD type matrix in the OD-Trip view. To represent the other two targets (i.e., OD pair and road segment), we overlay two visualizations on the map view: *a flow map* (Fig. 2D) and *a road heatmap* (Fig. 9B). These allow users to explore the geographical distribution of different targets (Task E2); in the flow map, trips are aggregated and shown as *flows* between OD pairs, while in the road heatmap, the traffic on individual roads is color-encoded.

The flow map (Fig. 2D) shows the number of trips between an OD pair as a curved edge. We adopt the edge rendering technique presented by Wood *et al.* [41], as it is computationally cheap enough to render a large number of flows responsively. The color and thickness of each edge is proportional to w_{elem}. The thickness of an edge is set to $5w_{elem}$ pixels. To show the direction of an edge, we use a Bezier curve, which was originally proposed by Fekete *et al.* [42]. To make both ends of an edge distinguishable, we draw a curve straighter at the origin and sharper at the destination.

The road heatmap (Fig. 9B) encodes the number of trips passed down each road segment to the color of a line. Therefore, road segments with higher traffic are represented in a more reddish and saturated color. In addition, the road heatmap panel (Fig. 9B) allows users to selectively see only the road types (i.e., primary, secondary, tertiary, and residential) they want. Independent of the road types, bike lanes can be installed on any of the road types. By clicking on the "Bike Lane" checkbox, bike lanes are overlaid in green segments on road segments with a thinner line (Fig. 9B).

The maximum traffic for each target, $TrafficMax_{elem}$, plays an important role in determining the density of the visual elements in the map view since their sizes depend on it, as shown in Equation (13). For example, an outlying OD pair with very high traffic will suppress other OD pairs, making the elements for the OD pairs too small to see. Whereas, if $TrafficMax_{elem}$ is too small, even relatively insignificant elements will be over-plotted. To alleviate this, we parameterize $TrafficMax_{elem}$ for each target to allow users to interactively adjust it from the minimum value (1) to the actual maximum traffic through the slider control (Fig. 2 (2)). Depending on $TrafficMax_{elem}$ that users set, w_{elem} can exceed 1, making elements too large on the map. So, we clamped w_{elem} to be in a range $[0, 1]$. To further alleviate visual clutter, we hide edges that are thinner than 0.5 pixels.

The map view supports brushing on stations, and this is especially useful when understanding the traffic in a specific area (e.g., 500-m neighborhood from a specific subway station). For brushing, three drawing shapes are provided: polygon-, rectangle-, and circle-shaped (Fig. 2 (4)). The brushed stations' borders are thicker and rendered in blue (Fig. 2 (6)).

Once a certain set of stations is brushed, users can assign them to a new station type in the station type panel (Fig. 2 D). Since the assigned station type is represented as a distinct symbol, this can help users gain further insight by taking the semantics of the station type into account in the analysis. We tried applying Bubble Sets [43] to highlight the membership of stations of the same type. However, we eventually decided not to use it since it often obscured other visual elements, and the domain experts did not find insights through it.

It is possible to brush on the predefined station sets and manage them as an independent station type. In the station type panel, there is a station preset drop-down list. After selecting the desired list item, users need to click on the "Select" button at the right side of the list. Then, the preset stations are brushed on the map. By doing so, users can brush on and label stations such as stations close to a subway or stations in a commercial area.

5.1.4 Station View

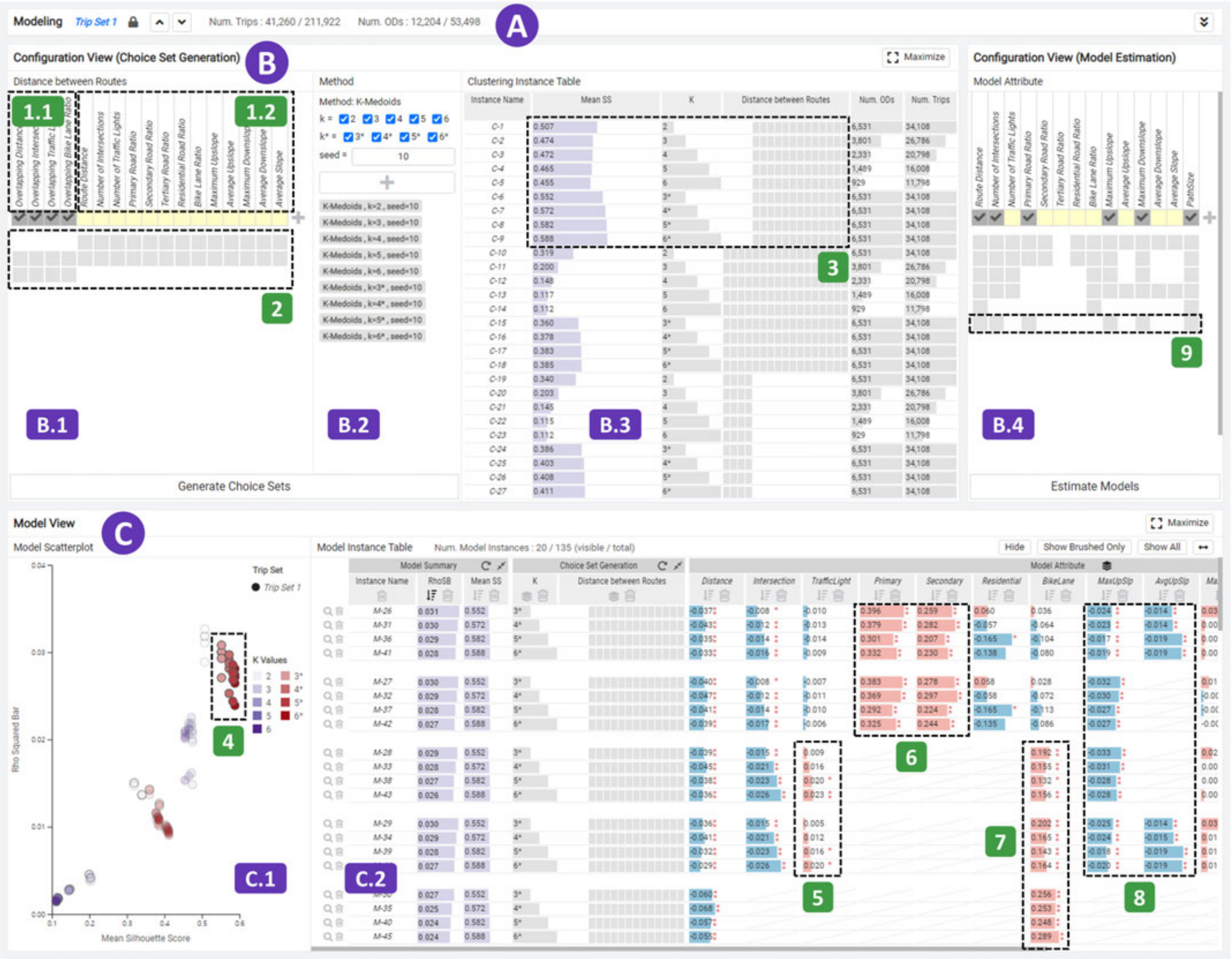
The station view (Fig. 2E) visualizes each station as a row in the table-based interface. Therefore, this view shows stations and their attributes without interference from other visual elements. Additionally, users can sort the stations by traffic and compare them by their attributes.

The station view visualizes three important types of information about a station: *total traffic*, *in-flow traffic*, and *out-flow traffic* (Table 1). At the center of a row, there is a horizontal bar representing the total traffic of a station (i.e., total traffic bar). There are columns representing in-flow and out-flow on the left and right of the total traffic. The x-axis of the two columns encodes the *OD Distance* (Table 2). We represent a station as a collection of associated OD pairs. We visualize each OD pair as a bar (i.e., OD bar) at the corresponding position on the x-axis.

In designing the station view, we mainly considered the consistency and interactivity of the map. The reason for doing so is to allow users to identify the geographical distribution of data represented in the station view to perform Task E2. For example, we make the color of the total traffic bars the same as that of the station symbol in the map view. The shape on the left of the total traffic bar represents a station type and is also the same as the map view. The OD bars of in-flow and out-flow share the same color and thickness as the map view's edge. Moreover, by adjusting the $TrafficMax_{elem}$ value in the panels of the map view, all the visual elements of the station view mentioned above are synchronized accordingly, as in the map view. Since the station view and the map view are closely connected in this way, users may not have any difficulties in using both views in succession. When users want to focus on the station's type information, the unique color for the station type can be encoded in the OD bars and the total traffic bars (Fig. 7).

5.1.5 Route View

The route view (Fig. 5C) shows the details of all routes taken between a particular OD pair, such as matched paths and route attributes. Unlike the aforementioned views, the route view allows users to take a detailed look at geographical distribution or route attribute distribution within a single OD pair (Tasks E2 and E3). Users can open the route view by clicking on any visual element representing an OD pair, such as a bubble in the OD bubble plot or a row in the OD-Trip view. The route view consists of two parts: *a route map* and *a route heatmap. The route map* shows all routes of the target OD pair on a map, while *the route heatmap* shows route attributes (Table 3) as a heatmap, where each row represents a route attribute and each column represents a route. Cells in the heatmap can either show the raw values as text or color-encoded as the *normalized* route attribute (i.e., the attribute



Modeling *Trip Set 1* Num. Trips : 41,260 / 211,922 Num. ODs : 12,204 / 53,498

Configuration View (Choice Set Generation)

Distance between Routes

Method

Method: K-Medoids

k = ☑2 ☑3 ☑4 ☑5 ☑6

k* = ☑3* ☑4* ☑5* ☑6*

seed = 10

K-Medoids , k=2 , seed=10
K-Medoids , k=3 , seed=10
K-Medoids , k=4 , seed=10
K-Medoids , k=5 , seed=10
K-Medoids , k=6 , seed=10
K-Medoids , k=3* , seed=10
K-Medoids , k=4* , seed=10
K-Medoids , k=5* , seed=10
K-Medoids , k=6* , seed=10

Clustering Instance Table

Instance Name	Mean SS	K	Distance between Routes	Num. ODs	Num. Trips
C-1	0.507	2		6,531	34,108
C-2	0.474	3		3,801	26,786
C-3	0.472	4		2,331	20,798
C-4	0.465	5		1,489	16,008
C-5	0.455	6		929	11,798
C-6	0.552	3*		6,531	34,108
C-7	0.572	4*		6,531	34,108
C-8	0.582	5*		6,531	34,108
C-9	0.588	6*		6,531	34,108
C-10	0.319	2		6,531	34,108
C-11	0.200	3		3,801	26,786
C-12	0.148	4		2,331	20,798
C-13	0.117	5		1,489	16,008
C-14	0.112	6		929	11,798
C-15	0.360	3*		6,531	34,108
C-16	0.378	4*		6,531	34,108
C-17	0.383	5*		6,531	34,108
C-18	0.385	6*		6,531	34,108
C-19	0.340	2		6,531	34,108
C-20	0.203	3		3,801	26,786
C-21	0.145	4		2,331	20,798
C-22	0.115	5		1,489	16,008
C-23	0.112	6		929	11,798
C-24	0.386	3*		6,531	34,108
C-25	0.403	4*		6,531	34,108
C-26	0.408	5*		6,531	34,108
C-27	0.411	6*		6,531	34,108

Configuration View (Model Estimation)

Model Attribute

Model View

Model Instance Table Num. Model Instances : 20 / 135 (visible / total)

Model Summary			Choice Set Generation		Model Attribute									
Instance Name	RhoSB	Mean SS	K	Distance between Routes	Distance	Intersection	TrafficLight	Primary	Secondary	Residential	BikeLane	MaxUpSlp	AvgUpSlp	Ma
M-26	0.031	0.552	3*		-0.037	-0.008	-0.010	0.396	0.259	0.060	0.036	-0.024	-0.014	0.03
M-31	0.030	0.572	4*		-0.043	-0.012	-0.013	0.379	0.282	-0.057	-0.064	-0.023	-0.014	0.00
M-36	0.029	0.582	5*		-0.035	-0.014	-0.014	0.301	0.207	-0.165	-0.104	-0.017	-0.019	0.00
M-41	0.028	0.588	6*		-0.033	-0.016	-0.009	0.332	0.230	-0.138	-0.080	-0.019	-0.019	0.00
M-27	0.030	0.552	3*		-0.040	-0.008	-0.007	0.383	0.278	0.058	0.028	-0.032		0.01
M-32	0.029	0.572	4*		-0.047	-0.012	-0.011	0.369	0.297	-0.058	-0.072	-0.030		-0.0
M-37	0.028	0.582	5*		-0.041	-0.014	-0.010	0.292	0.224	-0.165	-0.113	-0.027		-0.0
M-42	0.027	0.588	6*		-0.039	-0.017	-0.006	0.325	0.244	-0.135	-0.086	-0.027		-0.0
M-28	0.029	0.552	3*		-0.039	-0.015	0.009				0.192	-0.033		0.02
M-33	0.028	0.572	4*		-0.045	-0.021	0.016				0.155	-0.031		0.00
M-38	0.027	0.582	5*		-0.038	-0.023	0.020				0.132	-0.028		0.00
M-43	0.026	0.588	6*		-0.036	-0.026	0.023				0.156	-0.028		0.00
M-29	0.030	0.552	3*		-0.036	-0.015	0.005				0.202	-0.025	-0.014	0.03
M-34	0.029	0.572	4*		-0.041	-0.021	0.012				0.165	-0.024	-0.015	0.01
M-39	0.028	0.582	5*		-0.032	-0.023	0.016				0.143	-0.018	-0.019	0.01
[illegible]	0.027	0.588	6*		-0.029	-0.026	0.020				0.164	-0.020	-0.019	0.01
[illegible]	0.027	0.552	3*		-0.060						0.256			
M-35	0.025	0.572	4*		-0.068						0.253			
M-40	0.024	0.582	5*		-0.057						0.248			
M-45	0.024	0.588	6*		-0.055						0.289			

Fig. 6. The interface for the modeling stage. (A) The trip set list shows the information of the active trip set as in the exploration interface. (B) The configuration view enables users to configure sets of hyperparmeters used in the two primary modeling process: choice set generation and model estimation. (C) The model view shows the model estimation results, called *model instances*.

divided by the maximum value on the same attribute). The route map and the route heatmap are linked; a route focused on in one visualization is highlighted in the other.

5.2 Modeling Stage

After users create trip sets in the exploration stage, they can fit a route choice model to a trip set in the modeling stage. The modeling process consists of two procedures: choice set generation and model estimation. *The configuration view* (Fig. 6B) allows users to specify hyperparameter configurations for the two procedures (Task M1), and *the model view* (Fig. 6C) allows users to explore produced model instances based on an Overview+Detail approach (Tasks M2 and M3).

5.2.1 *Configuration View*

The configuration view (Fig. 6B) allows users to produce hyperparameter configurations for choice set generation and model estimation. We chose a data-driven approach to generate choice sets, where we cluster the observed routes (routes that are actually taken). Once the choice sets for all trips are generated, we fit a model that predicts the probability of routes being chosen from their characteristics (i.e., route attributes).

As introduced in Section 4.1.1, we adopted the k-medoids clustering algorithm that our domain experts are actively using. To measure the distance between trips' routes, we provide 17 distances with two types; four of them are the *overlap distances* (Fig. 6 (1.1)), and the rest are the *attribute distances*. Details of the 17 distances are already described in Section 4.1.1. Users can choose a subset of distances that will be included in distance computation, and we will denote such a subset of distances as a vector γ. γ is a 17-dimensional binary vector where each dimension represents whether a certain distance that will or will not be included in calculation of the distances between routes.

We support two types of the number of clusters k: the fixed (k) and bounded ($k*$) types as defined in Section 4.1.1. We denote a hyperparameter configuration for the k-medoids clustering algorithm as $\lambda = (k, \mathit{seed})$, where k can be either a fixed or a bounded type, and *seed* is a seed number for random number generation.

In practice, experts test different hyperparameter combinations based on their knowledge (Task M1) since it is hard

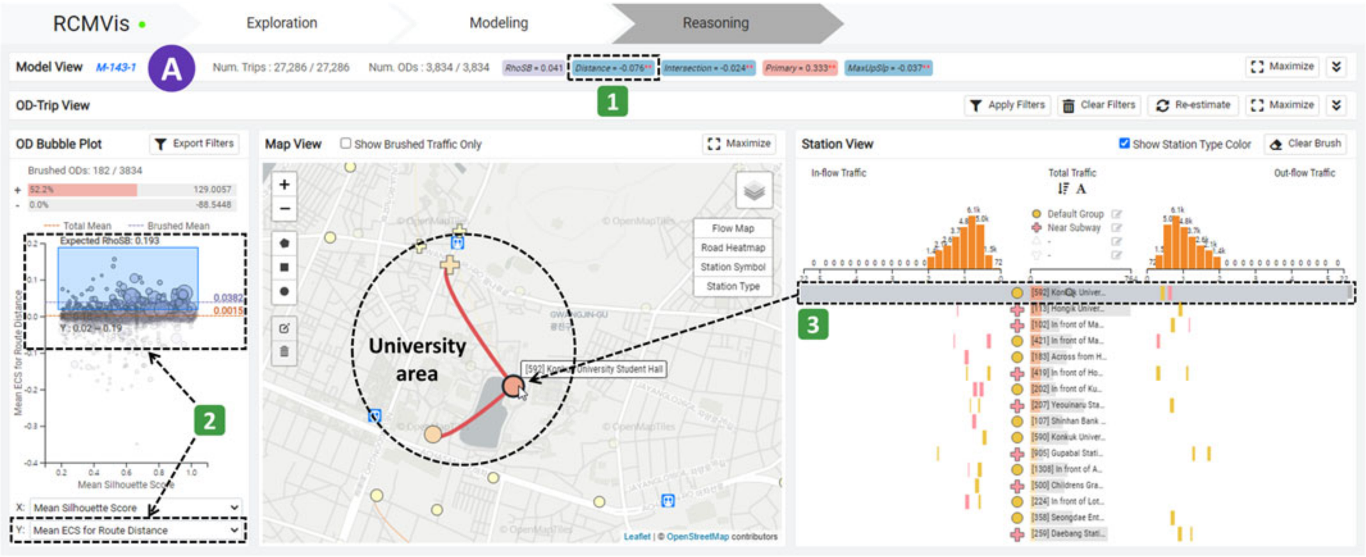


Fig. 7. The interface for the reasoning stage. This interface visualizes a selected model instance called an *active model instance* based on the views used in the exploration interface. (A) The header of the model view shows information about the active model instance, such as the number of trips and OD pairs, $\bar{\rho}^2$ (*rhoSB* in the interface), and statistically significant attributes and their coefficients. By clicking on the rightmost icon of the header, users can open the model view, which allows users to scan and even change the active model instance.

to figure out the best values for the hyperparameters (γ and λ) for choice set generation, for example, in terms of silhouette coefficient. To streamline this process, we allow experts to specify a set of hyperparameters for distance computation, $\boldsymbol{\Gamma} = \{\gamma_1, \gamma_2, \gamma_3, ...\}$, and a set of hyperparameters for clustering, $\boldsymbol{\Lambda} = \{\lambda_1, \lambda_2, \lambda_3, \ldots\}$), and test all possible combinations (γ_i, λ_j) in the Cartesian product of the two, $\boldsymbol{\Gamma} \times \boldsymbol{\Lambda}$.

The distance panel (Fig. 6B.1) allows users to configure $\boldsymbol{\Gamma}$. There are 17 check boxes in the panel, so users can include or exclude a distance for calculation of distances between routes. Clicking on the + symbol on the right will add a new configuration, γ_i, to $\boldsymbol{\Gamma}$ (Fig. 6 (2)). Similarly, *the method panel* (Fig. 6B.2) allows users to configure $\lambda_j = (k, \mathit{seed}) \in \boldsymbol{\Lambda}$.

After configuring the sets of hyperparameters for choice set generation ($\boldsymbol{\Gamma} \times \boldsymbol{\Lambda}$), they click on the “Generate Choice Sets” button to generate *clustering instances*. Each set of hyperparameters will generate one clustering instance; therefore, $|\boldsymbol{\Gamma}| \cdot |\boldsymbol{\Lambda}|$ clustering instances will be generated. An instance appears as a row in *the clustering instance table* (Fig. 6B.3). As a quality measure for a clustering instance, we use the mean silhouette score (*Mean SS* in the table), which is defined by averaging the silhouette scores of all trips in the active trip set.

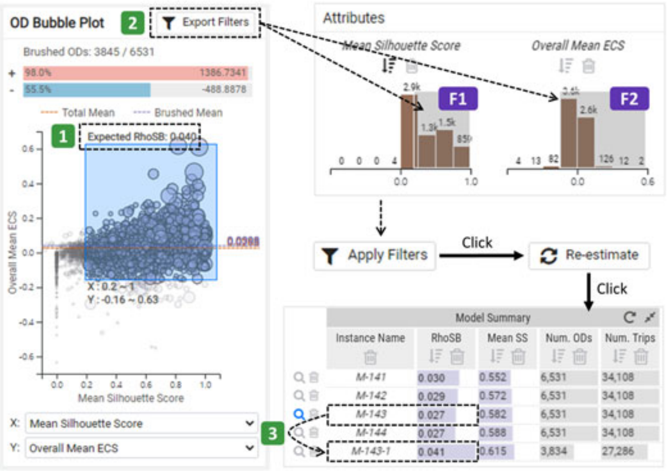
In the model estimation procedure, users fit a PSL model (Equation (4)) to each clustering instance. Similar to choice set generation, users must choose a set of *model attributes*, σ (introduced in Section 4.1.2), which are route attributes used as independent variables in modeling. Similar to specifying $\boldsymbol{\Gamma}$ and $\boldsymbol{\Lambda}$, users specify different combinations of model attributes, $\boldsymbol{\Sigma} = \{\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \boldsymbol{\sigma}_3, \ldots\}$ in *the model attribute panel* (Fig. 6 B-4). Finally, users click on the “Estimate Models” button to fit a model to each of clustering instances using one configuration of model attributes $\boldsymbol{\sigma}_i \in \boldsymbol{\Sigma}$, obtaining $|\boldsymbol{\Gamma}| \cdot |\boldsymbol{\Lambda}| \cdot |\boldsymbol{\Sigma}|$ *model instances* as a result.

5.2.2 Model View

The model view (Fig. 6C) supports an Overview+Detail approach for exploring the $|\boldsymbol{\Gamma}| \cdot |\boldsymbol{\Lambda}| \cdot |\boldsymbol{\Sigma}|$ model instances. From the overview (Fig. 6C.1), users can grasp overall patterns of the model instances (Task M2). From the detail (Fig. 6C.2), users can compare the instances with the help of the interactions, such as sorting, grouping, and hiding unnecessary results (Task M3). If there is an interesting model instance during the analysis, further investigation of the instance can be done in the reasoning interface.

The model scatterplot (Fig. 6C.1) serves as an overview of the model view. It represents each model instance as a single point. Reflecting domain experts’ modeling practice identified during the domain situation analysis, we decided to map the mean silhouette score and $\bar{\rho}^2$ (*rho-squared bar*) to the x- and y-axes, since the two indices are the performance measures for choice set generation and model estimation, respectively. Users can distinguish the trip set of the modeling result through the shape of a point. The hue of a point differentiates the type of k: the fixed type (k) as purple and the bounded type ($k*$) as red. Further, the more saturated the color is, the higher the absolute value of k is. To get details of certain points, the model scatterplot supports brushing and linking; users can brush on points they want to investigate further, and then the corresponding rows of the model instance table are highlighted in a gray background.

The model instance table (Fig. 6C.2) shows the details of each model instance. This table represents each model instance as a row. A row contains information about a model summary, a set of hyperparameters ($\boldsymbol{\Gamma} \times \boldsymbol{\Lambda} \times \boldsymbol{\Sigma}$), and estimated coefficients of model attributes. The table cells represent their value as a horizontal bar to help users compare values within the same column. When interpreting coefficients, the first thing users inspect is the sign of a value since different signs lead to the opposite meaning of the attribute in the modeling context. For instance, a negative *Route Distance* coefficient means that riders tend to avoid routes with longer distances. Therefore, we decided to differentiate the bar color to allow users to recognize it at a glance: positive as red and negative as blue. In addition, the cells show a single asterisk or two when their route



Model Summary

Instance Name	RhoSB	Mean SS	Num. ODs	Num. Trips
M-141	0.030	0.552	6,531	34,108
M-142	0.029	0.572	6,531	34,108
M-143	0.027	0.582	6,531	34,108
M-144	0.027	0.588	6,531	34,108
M-143-1	0.041	0.615	3,834	27,286

Fig. 8. The re-estimation process to obtain a refined model instance with a higher goodness of fit. The OD bubble plot shows (1) the expected $\bar{\rho}^2$ when the user re-estimates the active model instance using only the brushed OD pairs. By clicking on (2), the filtering conditions corresponding to the brush are transmitted to the OD-Trip view. After applying filtering conditions, users re-estimate the model instance with the filtered OD pairs and the set of hyperparameters that the model instance has. Then, the re-estimated model instance (3) appears in the table.

attribute's estimated coefficient is statistically significant ($p < .05$ and $p < .01$, respectively) (Fig. 6 (5)).

For an effective comparison between the model instances (Task M3), the model instance table supports sorting or grouping rows by each column or hiding rows that do not seem important. The columns representing numerical values, such as the $\bar{\rho}^2$ (*rhoSB* in the interface), can be used to sort rows. Other columns related to the set of hyperparameters, such as the k ($\mathbf{\Lambda}$), the set of distances ($\mathbf{\Gamma}$), or the set of model attributes ($\mathbf{\Sigma}$), can be used to group rows. The common analysis scenario using grouping is to investigate the effect of the set composition of model attributes ($\mathbf{\Sigma}$) on model instances; users can group by $\mathbf{\Sigma}$, as in Fig. 6C.2.

If users find a model instance that well describes route choice behaviors, they want to explore it at the data level, such as OD pairs or trips. This can be done by clicking on the magnifying lens icon on the left of the target model instance row. Then, the reasoning interface is activated to allow exploring the instance.

5.3 Reasoning Stage

To better understand the model instance at the data level, users should analyze the instance in the reasoning stage. The analysis target of this stage is the selected model instance in the modeling stage, and we call it an *active model instance*, similar to an active trip set of previous stages. Further, the model view at the top of the reasoning interface (collapsed in Fig. 7A but can be opened) allows users to scan all the model instances and switch the active model instance in the same manner as in the modeling interface.

In the reasoning interface, the views are mostly the same as the exploration interface except for the model view. This is because users need to analyze the data included in a trip set, such as OD pairs, trips, and stations, in the same way as in the exploration stage. However, the statistics derived from the active model instance are provided to help users perform an in-depth analysis of the model instance in the trip set data space. For example, statistics such as the *estimation contribution score* (*ECS*) allow users to selectively explore data that contribute to the estimated coefficients. To this end, users can brush OD pairs having high *ECS* in the OD bubble plot and closely inspect their characteristics in the map view or the station view. By doing so, users can determine which trips or OD pairs mainly contributed to estimating the coefficients (Task R1).

The reasoning interface facilitates re-estimation of the active model instance by applying more filtering conditions (Task R2). The general workflow of the re-estimation process is shown in Fig. 8. To help select the OD pairs used for re-estimation, the OD bubble plot shows the *expected* $\bar{\rho}^2$ (*rho-squared-bar*). This value is obtained by substituting the LL_{final} (Equation (7)) calculated with only the trips contained in the brushed OD pairs for the LL_{final} in $\bar{\rho}^2$ (Equation (8)). The expected $\bar{\rho}^2$ is immediately displayed when users are brushing OD pairs on the bubble plot. Users can refer to this value to determine which OD pairs to keep and re-estimate model coefficients from them.

After brushing the OD pairs, the brushed x and y ranges of the OD bubble plot can be exported to the OD-Trip view's filtering conditions, respectively. As in the exploration stage, the two filtering conditions can be applied to the active model instance by clicking on the "Apply Filters" button. Then, by clicking on the "Re-estimate" button, the re-estimation of the filtered active model instance starts with the same set of hyperparameters that the active model instance used before. The newly estimated instance is displayed as a new row in the model view.

6 Evaluation

In this section, we evaluate the design of RCMVis through a case study and expert interview.

6.1 Case Study

We conducted a case study with two of our domain experts (P1 and P2). They participated in the case study together and had a one-hour tutorial session to learn the features of RCMVis before participating in the case study. We allowed them to use the system for 90 minutes and then interviewed them for 30 minutes. All the processes were done remotely due to the COVID-19 pandemic. The same bicycle trip path dataset in Section 3 was used in the case study. The experts' main goal is to obtain insights on which road factors are considered by bicycle riders when choosing a route and where such behaviors are strongly seen.

6.1.1 *Exploration Stage*

One of the primary purposes of operating public bicycle systems is to provide a means of transportation connected to public transportation. For example, bicycles allow commuters to quickly move from home to a subway station (i.e., the first mile) and from a subway station to an office (i.e., the last mile), even during peak hours. Hence, the experts were first interested in traffic occurring during peak hours. To view such traffic in the exploration view, they first applied three filtering conditions on the OD-Trip view to derive the active trip set comprised of weekdays, AM/PM peaks, and short-distance (0—2km) trips (Task E1; Fig. 2

(1)). Then, they defined a new station type consisting of predefined "near subway" stations (Fig. 2 (3)). They postulated that most of the first or last mile riders had used these "near subway" stations as their origin or destination. After creating the new type, the "near subway" type stations appeared as cross symbols on the map view.

Geographical Distribution of Trips. To understand the geographical distribution of the trips, they attempted to locate heavy traffic regions on the flow map (Task E2). In particular, they wanted to identify the trip distributions for some areas they frequently investigate in their usual analysis and re-confirm from the real-world data that these areas are worth analyzing. Initially, most of OD pairs on the flow map were suppressed due to a few OD pairs with excessive traffic, so the experts could hardly see the overall distribution of the trips. To make the suppressed OD pairs visible, the experts adjusted $TrafficMax_{elem}$ on the flow map (Fig. 2 (2)).

Subsequently, they discovered two prominent areas (Figs. 2 (5) and 2 (6)) with high traffic. Fig. 2 (5) is *Hongdae*, one of the city's most popular downtown areas, with a large floating population. The major subway line also passes through this area. Fig. 2 (6) is *Yeouido*, a central business district of this city, where many office workers commute. Interestingly, both regions were ones they often analyzed, yet they were able to discover unexpected traffic distributions in these areas. In Hongdae, the flows from surrounding areas were highly concentrated on a specific "near subway" station (i.e., "Hongik University Station Exit 2"). Unlike Hongdae, Yeouido had noticeable flows between several "near subway" stations in the outer areas and the center, where many companies were located. The experts speculated that these flows might be the trips of office workers commuting to and from Yeouido.

There were three stations with relatively high traffic among the "near subway" stations in Yeouido (blue-bordered cross symbols in Fig. 2 (6)). To investigate the OD pairs associated with those stations, the experts clicked on them on the flow map to highlight the corresponding rows in the station view. In the station view, they determined that those stations' out-flow traffic was higher than in-flow (Fig. 2 (7)), which implies that riders mainly used bicycles in Yeouido for last-mile riding. They noted that this finding could be useful in rebalancing bicycles in Yeouido during peak hours.

Route Choice Behaviors. Before modeling, the experts attempted to hypothesize about route choice behavior by checking the distortion of the distribution of route attributes (Task E3). In the OD bubble plot, the total mean nonparametric skew for *Route Distance* was about 0.23 (orange dotted line in Fig. 2C), indicating that route choices were biased toward routes with relatively short distances. The experts mentioned that commuters tend to choose a shorter route because they want to reach their destination quickly; the finding was consistent with their background knowledge. Therefore, the experts expected a negative coefficient for *Route Distance*, although it could vary depending on a set of hyperparameters used in the modeling stage.

They also tried to identify the nonparametric skew of attributes they were interested in, such as the *Maximum Upslope* and *Bike Lane Ratio*, which were 0.11 and -0.03, respectively (Task E3). Although the nonparametric skew of *Maximum Upslope* was not as large as that of *Route Distance*, the result was consistent with the general notion that riders do not prefer slopes. For *Bike Lane Ratio*, the experts initially expected that riders would prefer roads with bike lanes. Correspondingly, the sign of the nonparametric skew was negative as expected, but the absolute value was relatively small (-0.03). Regarding this result, the experts mentioned that it might be due to the OD pairs with low diversity of routes. For example, some OD pairs may not have bike lanes on their routes, or in extreme cases, all routes have a high ratio of bike lanes. When riding between such an OD pair, riders have no choice but to choose a route with or without a bike lane. To eliminate the influence of such OD pairs, they applied a filtering condition *OD-TripRange* (Task E1; Table 4). This condition left only OD pairs that contained both trips whose *Bike Lane Ratio* was less than 0.2 and greater than 0.8 (Fig. 4C). The nonparametric skew after filtering changed to -0.07, which was slightly larger than before but not very impressive (Fig. 5B); they decided to remove this condition.

They found an interesting OD pair in the bubble plot. That OD pair was positively skewed, unlike the mean (-0.07), and had high traffic (Fig. 5 (2)). This indicated that bike lanes were not preferred in this OD pair, contrary to the expectation. To learn more about the trips of this OD pair, the experts clicked on the bubble to open the route view.

In the route view (Fig. 5C), the experts found that the region around this OD pair was industrial based on the names and locations of the stations. There was a subway station near the destination ("In front of Seongsu Station Exit 2"). The experts speculated that the traffic of this OD pair had resulted mainly from factories or offices to the subway station to return home. To identify routes with a high *Bike Lane Ratio*, they sorted the routes by *Bike Lane Ratio* in the heatmap. This revealed an appreciable pattern, routes with a low *Bike Lane Ratio* had a high *Residential Road Ratio* (Fig. 5 (4)). Considering that bike lanes are mainly installed on primary or secondary roads in this city, the experts speculated that riders' choice of residential roads over primary or secondary roads also affected bike lane choice. They also mentioned that riders sometimes choose residential roads rather than other roads because residential roads are relatively wider for riding bicycles. This OD pair seems to be a good example of such riding.

6.1.2 Modeling Stage

After exploring the active trip set, the experts started to configure a set of hyperparameters for modeling (Task M1). They wanted to determine the usefulness of the k-medoids clustering algorithm as a method of choice set generation. Their main concerns were to find the optimal set of distances between routes and the hyperparameter k. In particular, they wanted to test the effectiveness of overlap distances (Fig. 6 (1.1)) compared with attribute distances (Fig. 6 (1.2)). Therefore, they created three sets of distances ($|\mathbf{\Gamma}| = 3$): a set with only the overlap distances (γ_O), a set with only the attribute distances (γ_A), and a set with both distances (γ_{O+A}) (Fig. 6 (2)). Concerning the k, they decided to test all possible values ($|\mathbf{\Lambda}| = 9$): five fixed and four bounded k values (Fig. 6B.2) with the same seed.

Overview and Comparison of Model Instances. The clustering instance table (Fig. 6B.3) shows the results of choice set generation (i.e., clustering instances). The mean silhouette scores (*Mean SS* in the interface) were higher when only the attribute distances (γ_A) were used (Fig. 6 (3)). The experts found it interesting that the scores of γ_O and γ_{O+A} were far lower than the score of γ_A. The reason behind these results, they conjectured, is that as the overlap increases the similarity of routes' attributes also increases, but the opposite may not be true. Regarding k, there was a difference in the pattern of scores between using the fixed k and bounded k. In the case of the fixed k, the scores decreased as k increased. Conversely, with the bounded k, the score always increased as $k*$ increased. Since the bounded k selects the k with the highest score within a range, it was more effective in obtaining high silhouette scores.

To estimate models from the clustering instances, the experts chose sets of model attributes $\boldsymbol{\Sigma}$ (Task M1; Fig. 6B.4). They included *Path Size* in all sets because they always used it as a correction term when estimating the route choice model except for some unusual cases. *Tertiary Road Ratio* was excluded since the sum of all the road type ratios on a route is always 1, which can cause multicollinearity. In addition, they added the sets using only the maximum slopes (i.e., *Maximum Upslope* and *Maximum Downslope*) instead of the average slopes. Finally, the set with a small number of attributes was added by including only *Route Distance*, *Bike Lane Ratio*, and *Path Size*. Before estimation, the experts expected a positive correlation between the silhouette scores and the $\bar{\rho}^2$ (*rho-squared bar*) of the model instances; the better a clustering instance is (i.e., with a high silhouette score), the better a model instance fits the clustering instance (i.e., with a high goodness of fit or $\bar{\rho}^2$).

After the model estimation process had been done, the model scatterplot (Fig. 6C.1) showed the overview of the model instances with the mean silhouette score on the x-axis and the $\bar{\rho}^2$ on the y-axis (Task M2). As expected, there seemed to be a positive correlation between the two. The experts looked into the instances with the high $\bar{\rho}^2$ at the upper right (Fig. 6 (4)). According to the reddish colors of the instances, they noticed that most of the high $\bar{\rho}^2$ instances had the bounded k; however, they found that the model instance with the highest $\bar{\rho}^2$ had the fixed k and was represented as a light purple point in the plot.

To determine the details of these model instances, the experts started to inspect the instances in the model instance table (Task M3; Fig. 6C.2). They sorted the rows by $\bar{\rho}^2$ to see the instances with a high goodness of fit. Then, they found that the instance with $k = 2$ had the highest $\bar{\rho}^2$. However, they were not sure if $k = 2$ simulates the actual trips effectively because it oversimplifies routes into just two cases; thus, they made these instances hidden from the list. By doing so, the instances that used only the attribute distances for clustering (γ_A) with k from $3*$ to $6*$ became the best instances in terms of the $\bar{\rho}^2$ (Fig. 6 (4)). To further inspect these promising instances, they hid all other instances. Then, they grouped the remaining instances by the set of model attributes to identify the patterns of attribute coefficients.

In most model instances, the attributes *Route Distance* and *Number of Intersections* had negative coefficients and were statistically significant. This was consistent with the common-sense notion that riders do not prefer long routes and many intersections. Moreover, the results indicated that riders preferred primary and secondary roads to residential roads (Fig. 6 (6)) unlike the situation of the one OD pair in Fig. 5 (4). Interestingly, *Bike Lane Ratio* was significant only when *Primary Road Ratio* and *Secondary Road Ratio* were not included in the set of model attributes (Fig. 6 (7)). Since bike lanes are commonly installed on primary or secondary roads in this city, the experts thought that correlations between these road types might be responsible for these results. For the same reason, *Number of Traffic Lights* also showed similar results as *Bike Lane Ratio* (Fig. 6 (5)). Based on these results, the experts decided not to include *Number of Traffic Lights*, *Primary Road Ratio*, and *Bike Lane Ratio* together in the same set of model attributes since they are correlated.

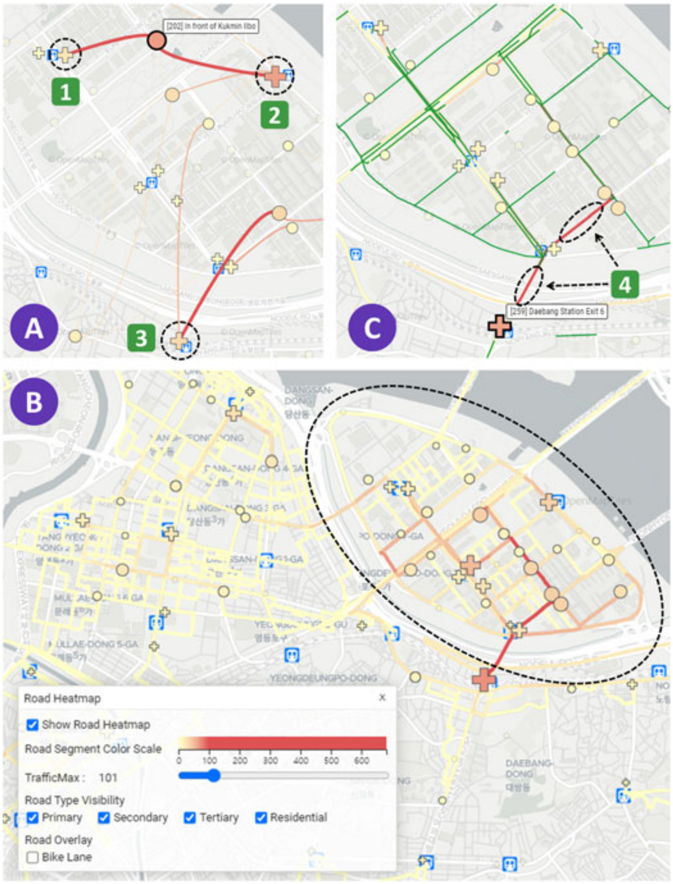
Regarding the attributes about slopes, *Maximum Upslope* and *Average Upslope*, the coefficients were all negative and statistically significant (Fig. 6 (8)). Considering that the two attributes had similar meanings and the coefficients were similar, the experts commented that it would be good to use only one of the two attributes at a time in future analysis.

Based on the findings so far, the experts refined the model estimation procedure with a new set of model attributes: $\boldsymbol{\sigma}_{new}$ ={*Route Distance*, *Number of Intersections*, *Primary Road Ratio*, *Maximum Upslope*, *Maximum Downslope*, *Path Size*} (Fig. 6 (9)). As a result, 27 new model instances ($|\boldsymbol{\Gamma}| \cdot |\boldsymbol{\Lambda}| \cdot |\boldsymbol{\sigma_{new}}| = 3 \cdot 9 \cdot 1$) were created. Similar to before, model instances using only the attribute distances in the calculation of distances between routes and bounded k had a relatively high $\bar{\rho}^2$. Moreover, most of their attributes were statistically significantly estimated. Among them, the experts decided to further inspect the model instance with $k = 5*$ in the reasoning stage.

6.1.3 Reasoning Stage

The experts' main purpose was to selectively explore trips and OD pairs that largely contributed to the estimation result (Task R1) of the active model instance (one with $k = 5*$). To this end, they started with the OD bubble plot. Before inspecting the active model instance (one with $k = 5*$), they wanted to refine the instance first, and thus OD pairs with too low silhouette scores or estimation contribution scores (*ECS*) were excluded. For this purpose, they brushed on OD pairs whose silhouette scores greater than 0.2 in the OD bubble plot (Fig. 8). Since it is difficult to judge *ECS* by its value, they repeatedly brushed and checked the expected $\bar{\rho}^2$ (Fig. 8 (1)). Eventually, they adjusted the range of *ECS* that could make the expected $\bar{\rho}^2$ about 0.04; our experts said that a $\bar{\rho}^2$ of about 0.04 is satisfactory. The blue gauge at the top of the OD bubble plot showed that only about 55 percent of OD pairs with negative *ECS* were brushed. In other words, the other 45 percent of OD pairs negatively contributing to the estimation result were excluded in the selection and were not used for re-estimation. After re-estimating the instance with the two filtering conditions (F1 and F2 in Fig. 8) applied, they obtained a new model instance with a $\bar{\rho}^2$ of 0.041, which was considerably higher than the previous one (Task R2; Fig. 8 (3)).

OD-Level Insights. The newly estimated model instance also had a negative coefficient for *Route Distance* (Fig. 7 (1)). This implied that riders tend to prefer short routes. To understand this result further, the experts tried to find OD pairs that largely contributed to this coefficient (Task R1). They mapped the y-axis of the bubble plot to *mean ECS for*



Road Heatmap ×

☑ Show Road Heatmap

Road Segment Color Scale

TrafficMax : 101

Road Type Visibility

☑ Primary ☑ Secondary ☑ Tertiary ☑ Residential

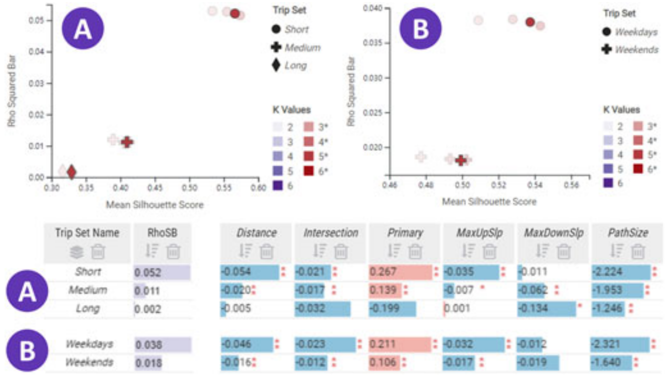
Road Overlay

□ Bike Lane

Fig. 9. (A) There are noticeable last mile flows from near the subway stations ((1), (2), and (3)) to the central commercial area. (B) The road heatmap provides the control panel, and it allows users to adjust the *TrafficMax* to manipulate the color scale of road segments. In addition, it provides checkboxes to hide and show road segments with a certain road type and overlay bike lanes. (C) After overlaying bike lanes (with green segments), road segments with high traffic but no bike lane installed were found.

the Route Distance and brushed on OD pairs having relatively large positive *ECS* (Fig. 7 (2)). Subsequently, they began to explore the map and station views to understand the characteristics of the brushed OD pairs. They soon realized that the station with the largest traffic was located in a university (Task E2; Fig. 7 (3)). Additionally, in the map view, they identified that riders mainly rented bicycles at a station in the university and travelled to nearby subway stations located around the university. Considering these flows and a negative coefficient for *Route Distance* together, the experts concluded that riders at the university (possibly students) strongly prefer short routes and have very purposeful movements. They believed that this example would offer valuable insights for policy-makers to understand public bicycle usage for supporting efficient rides.

After the university case, the experts began to explore Yeouido, which is the place they were originally interested in (Fig. 9A). They found high traffic in the (1), (2), and (3) stations in Fig. 9A, which were very close to subway stations (Task E2). Since these OD flows on the map contributed to the negative coefficient of *Route Distance*, we could infer that riders in this area strongly prefer short routes for the last mile of riding (i.e., going to their office from the subway station).



	Trip Set Name	RhoSB	Distance	Intersection	Primary	MaxUpSlp	MaxDownSlp	PathSize
A	*Short*	0.052	-0.054 ⁑	-0.021 ⁑	0.267 ⁑	-0.035 ⁑	-0.011	-2.224 ⁑
	Medium	0.011	-0.020 ⁑	-0.017 ⁑	0.139 ⁑	-0.007 *	-0.062 ⁑	-1.953 ⁑
	Long	0.002	-0.005	-0.032	-0.199	0.001	-0.134 *	-1.246 ⁑
B	*Weekdays*	0.038	-0.046 ⁑	-0.023 ⁑	0.211 ⁑	-0.032 ⁑	-0.012	-2.321 ⁑
	Weekends	0.018	-0.016 ⁑	-0.012 ⁑	0.106 ⁑	-0.017 ⁑	-0.019	-1.640 ⁑

Fig. 10. The model view with different trip sets. (A) shows the trip sets with *Short*-, *Medium*-, and *Long*-distance. (B) presents the trip sets with *Weekdays* and *Weekends*.

Road-Level Insights. The experts also analyzed riders' preference for primary roads. Unlike *Route Distance*, *Primary Road Ratio* had a positive coefficient. To find OD pairs that contributed to this result, the experts set *mean ECS for the Primary Road Ratio* to the y-axis of the bubble plot and brushed OD pairs having high *ECS* (Task R1). Then, they switched the flow map to the road heatmap to obtain road-level insights. In the road heatmap, they found the traffic in Yeouido was high (Task E2; Fig. 9B), and this consisted of the traffic of riders who preferred primary roads. The experts commented that riders who prefer primary roads could be good potential users of bike lanes, considering that bike lanes are mainly installed on primary roads. To identify the installation status of bike lanes, they made the bike lane overlay visible in the heatmap (Fig. 9C) and identified segments of primary roads without bike lanes despite high traffic (Fig. 9 (4)). The experts noted that these findings could guide the bike lane installation process of policy-makers.

6.1.4 Model Comparison

In the analysis described above, the experts investigated only one trip set at a time. This time, the experts created three trip sets with different distances (*Short*, *Medium*, and *Long*) and two trip sets with different days of the week (*Weekdays* and *Weekends*). Then, they went through choice set generation and model estimation for each trip set with the hyperparameters used above.

Short- versus Medium- versus Long-Distance. The experts created trip sets with three ranges of *OD Distance* based on their domain knowledge: *Short* ($[0, 2)$ km), *Medium* ($[2, 5)$ km), and *Long* ($[5, \infty)$ km) (Fig. 10). The three models differed significantly in terms of $\bar{\rho}^2$, and the *Short* trip set had the highest $\bar{\rho}^2$. The experts noted that trips with *Long* OD distances were likely to be leisure activities. They usually refer these trips whose purpose is not just to travel to a destination as irrational trips, and it is more challenging to model such trips according to the experts. Likewise, the model instance for trips with *Long* distances had a large number of insignificant coefficients.

Weekdays versus Weekends. According to $\bar{\rho}^2$ and attribute coefficients, *Weekday* trips showed more apparent choice behavior than *Weekend* trips. For all the attributes except *Maximum Downslope*, the magnitudes of coefficients were bigger for *Weekday* trips. Like the coefficients, the $\bar{\rho}^2$ of *Weekday* trips was much higher than that of *Weekend* trips. The experts inferred from these results that more riders are riding for leisure on weekends, and these irrational rides might lower the goodness-of-fit of the model for *Weekend* trips. Although the magnitudes were different, the aforementioned attributes' coefficients were equally significantly estimated ($p < .01$), indicating that route choice behaviors of purposeful riders are not much different for weekdays and weekends.

6.2 Domain Expert Interview

We conducted interviews with the domain experts after the case study and summarize their feedback on each analysis process.

Exploration Stage. The experts mentioned that overall they could use the views in the exploration stage without difficulties. P1 mentioned "Previously, I had to use several tools to explore the data alternately, and it was burdensome to manage the various target data with different filtering conditions. However, I could interactive filter data in this system while referring to the distribution of data attributes in the OD view. Then, I could immediately check the filtering results through visualizations such as the map or the station view. Also, it was convenient to manage multiple data with different filtering conditions on the list (i.e., the trip set list)." Both P1 and P2 noted that they could gain insights for modeling by identifying route attribute distributions. Especially, P2 said "It is important to establish a hypothesis about the modeling result through a data exploration process, but we had no specific way to derive a hypothesis other than rule-of-thumb exploration with the existing tools like GIS. One of the strengths of this system is that we could derive clues about how riders perceive a particular route attribute even before modeling by inspecting the bubble plot and the nonparametric skew index. By using them with the other visualizations, we could get various insights."

Modeling Stage. P1 mentioned "During my everyday analysis, I build a lot of models and compare them, but it has been done mostly in a pairwise manner. Therefore, it was hard for me to grasp the overview of multiple model instances. In this system, I could identify which attributes show interesting patterns, since it provides the overview along with the detailed information of multiple instances collectively." P2 commented about the interactions in the model instance table: "Grouping, sorting, and hiding model instances are simple but valuable. These interactions allow me to reveal patterns that were difficult to discover by naively inspecting a bunch of collected model instances."

Reasoning Stage. P1 commented about our reasoning process that "Many analysts question whether the results are sound even after modeling. To explain the model estimation results, analysts often manually find route choices (i.e., trips) that well-follow the estimated coefficients. This system addresses these questions as it allows us to investigate model instances at the data level by delivering *ECS*s of OD pairs and trips." P2 mentioned the model re-estimation process as follows: "We consider $\bar{\rho}^2$ as the most important indicator when evaluating the model. Therefore, to derive a model with a high $\bar{\rho}^2$, we have made efforts such as trying out various hyperparameters in the modeling process or applying various filtering conditions in the exploration process. Instead, in this system, we could exploit *ECS* derived from the modeling result to improve $\bar{\rho}^2$ of the already estimated model. Interactive filtering by *ECS* and re-estimation process make us obtain a better model than before. Also, we could compare the re-estimated models with existing ones as the system seamlessly adds them in the model view."

The experts also noted that there was room for improvement. Currently, we represent only traffic or bike lane information on the road segments in the map view, but they suggested that it would be helpful to show information such as slopes on the road segments.

7 Discussion

This section discusses the lessons we learned while designing RCMVis through regular meetings with our domain experts. Hopefully, these lessons can help designers who want to build an interactive visualization tool for route choice modeling.

A Three-Stage Analysis Framework. In the early stages of design, the analysis procedure we originally planned consisted of two steps: data exploration and then modeling. However, we observed that the domain experts did no tend to spend much time on data exploration before modeling. That is, even without data exploration, most of them already had filtering conditions of interest and promising sets of hyperparameters for modeling in mind. Indeed, their primary strategy was to estimate all models with their desired configurations first and choose the model to be further analyzed. Once they selected an impressive model instance, they started to inspect the model's characteristics, OD pairs, and trips. Based on this observation, we decided to add a reasoning stage at the end of the process to support rationalization aligned with the tasks that the experts perform in practice. In addition, the experts also emphasized that the exploration stage is necessary when analyzing unseen data, even though this stage is often skipped when analyzing familiar data.

We designed the interface of each stage as a separate tab. This design intuitively shows what stage users are currently working on and allows users to explicitly move on to the stage they want. In the initial design, users had to perform exploration and reasoning in one integrated interface, as we configured the interface based on the visualization target rather than the analysis stage. The types of data visualized in the exploration and reasoning stages (e.g., OD pairs, trips, and stations) are almost identical. However, the exploration stage explores a trip set, whereas the reasoning stage is for exploring a model instance and the derived model statistics. To support both stages in a single interface, the interface had to become complex. In addition, the domain experts reported that it was somewhat confusing because the stage transition was implicit. For this reason, we decided to organize the interface according to each analysis stage.

View OD Pair as the Context of Route Choices. When analyzing trips, it is necessary to know which OD pair they belong to. An OD pair can serve as an important context for riders when they make route choices. A route choice refers to determining a route among many alternatives in a choice set for a specific OD pair. Since a choice set can be given differently for each OD pair, riders' route choices depend highly on the given OD pair. In other words, even if two riders choose similar routes, their choices could be interpreted differently in the modeling context if their OD pairs are different.

In the initial stage of the design process, we wanted to visualize trips effectively, but we overlooked the need to consider OD pairs together. We collectively visualized all trips using a multi-dimensional visualization (like parallel coordinates) and called it the trip view. However, we received negative feedback from our domain experts when we showed the trip view. They commented that, in RCM analysis, it is necessary to identify meanings of trips within their OD pairs rather than analyzing the individual trip. Additionally, patterns revealed with all trips shown at once do not mean much except for the aggregated departure time or geographic distribution (i.e., the time bar charts and road heatmap). As a result, we decided to revise the trip view and came up with the current design of the OD-Trip view to represent trips within each OD pair.

Limitations and Future Work. In this section, we would like to mention two limitations of our work. First, the exploration stage still depends on users' own exploration strategies rather than providing a more systematic way for exploration. The current exploration stage is designed to help users understand the overall distribution of a movement dataset. Users can discover insights that can be directly helpful in the modeling stage, especially hyperparameter tuning in this exploration process. However, discovering such insights relies on users' own exploration strategy rather than a systematic process. By providing a more systematic data exploration process for route choice modeling, we can help less experienced users in the domain perform better with RCMVis.

The second limitation is related to computational efficiency. The computation time of choice set generation and model estimation is several minutes when the number of OD pairs in a trip set exceeds about 10 K. In such a case, an analysis may be blocked until the computation is done. Currently, RCMVis caches all results of the two processes performed by users. Our domain experts often analyze the same data multiple times, so this caching alone was considered satisfactory by them, but we do not think this is sufficient enough in general. Instead, we may help users make quick judgments before the full result is ready by progressively showing the intermediate results. However, further research is necessary to understand which intermediate results might be useful to users.

8 Conclusion

We present RCMVis, a visual analytics system for interactively supporting route choice modeling. Through close collaboration with the domain experts, we identified the problems they faced in their analysis tasks. Based on such findings, we suggest a three-stage interactive modeling framework to streamline the process of RCM analysis. We also designed an interactive visualization system to effectively support the three-stage modeling framework. Through a case study using a real-world bicycle dataset, the experts could make meaningful discoveries about the data and the models they developed, including geographical distributions of traffic, the hyperparameter space of the models, and data-level insights to help interpret models. Furthermore, through expert interviews, we showed the efficacy of each analysis stage of RCMVis. We believe that our analysis framework and visual designs will not only be helpful to RCM, but can also be extended to other related problems, such as bike rebalancing and bike lane planning.

Acknowledgments

The ICT at Seoul National University provided research facilities for this study. We would like to thank Young-Hyun Seo, Jeongwook Seo, and Christian Kapuku for their valuable feedback. Furthermore, we are grateful to the reviewers of our submission to VAST 2020 and TVCG for their valuable comments and suggestions, which helped us improve our article.

References

[1] C. G. Prato, "Route choice modeling: Past, present and future research directions," *J. Choice Modelling*, vol. 2, no. 1, pp. 65–100, 2009.
[2] ArcGIS, 2020. [Online]. Available: https://www.arcgis.com/
[3] QGIS, 2020. [Online]. Available: https://qgis.org/
[4] R. Bellman and R. Kalaba, "On kth best policies," *J. Soc. Ind. Appl. Math.*, vol. 8, no. 4, pp. 582–588, 1960.
[5] J. Broach, J. Gliebe, and J. Dill, "Calibrated labeling method for generating bicyclist route choice sets incorporating unbiased attribute variation," *Transp. Res. Rec.*, vol. 2197, no. 1, pp. 89–97, 2010.
[6] D. Ton, D. Duives, O. Cats, and S. Hoogendoorn, "Evaluating a data-driven approach for choice set identification using GPS bicycle route choice data from amsterdam," *Travel Behav. Soc.*, vol. 13, pp. 105–117, 2018.
[7] D. Ton, O. Cats, D. Duives, and S. Hoogendoorn, "How do people cycle in Amsterdam, Netherlands?: Estimating cyclists' route choice determinants with GPS data from an urban area," *Transp. Res. Rec.*, vol. 2662, no. 1, pp. 75–82, 2017.
[8] M. Ben-Akiva and M. Bierlaire, "Discrete choice methods and their applications to short term travel decisions," in *Handbook of Transportation Science*. Berlin, Germany: Springer, 1999, pp. 5–33.
[9] NLOGIT, 2020. [Online]. Available: http://www.limdep.com/products/nlogit/
[10] Emme, 2020. [Online]. Available: https://www.inrosoftware.com/en/products/emme/
[11] N. Andrienko and G. Andrienko, "Visual analytics of movement: An overview of methods, tools and procedures," *Inf. Vis.*, vol. 12, no. 1, pp. 3–24, 2013.
[12] G. Andrienko, N. Andrienko, and S. Wrobel, "Visual analytics tools for analysis of movement data," *ACM SIGKDD Explorations Newslett.*, vol. 9, no. 2, pp. 38–46, 2007.
[13] W. Chen, F. Guo, and F.-Y. Wang, "A survey of traffic data visualization," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 2970–2984, Dec. 2015.
[14] Y. Zheng, W. Wu, Y. Chen, H. Qu, and L. M. Ni, "Visual analytics in urban computing: An overview," *IEEE Trans. Big Data*, vol. 2, no. 3, pp. 276–296, Sep. 2016.
[15] G. Andrienko, N. Andrienko, W. Chen, R. Maciejewski, and Y. Zhao, "Visual analytics of mobility and transportation: State of the art and further research directions," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2232–2249, Aug. 2017.
[16] N. Marković, P. Sekuła, Z. Vander Laan, G. Andrienko, and N. Andrienko, "Applications of trajectory data from the perspective of a road transportation agency: Literature review and maryland case study," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1858–1869, May 2019.
[17] J. Pu, S. Liu, Y. Ding, H. Qu, and L. Ni, "T-Watcher: A new visual analytic system for effective traffic surveillance," in *Proc. IEEE 14th Int. Conf. Mobile Data Manage.*, 2013, pp. 127–136.
[18] Z. Wang, M. Lu, X. Yuan, J. Zhang, and H. Van De Wetering, "Visual traffic jam analysis based on trajectory data," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2159–2168, Dec. 2013.
[19] C. Lee *et al.*, "A visual analytics system for exploring, monitoring, and forecasting road traffic congestion," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 11, pp. 3133–3146, Nov. 2020.
[20] H. Guo, Z. Wang, B. Yu, H. Zhao, and X. Yuan, "TripVista: Triple perspective visual trajectory analytics and its application on microscopic traffic data at a road intersection," in *Proc. IEEE Pacific Vis. Symp.*, 2011, pp. 163–170.

[21] H. Liu, Y. Gao, L. Lu, S. Liu, H. Qu, and L. M. Ni, "Visual analysis of route diversity," in *Proc. IEEE Conf. Vis. Analytics Sci. Technol.*, 2011, pp. 171–180.

[22] W. Zeng, C.-W. Fu, S. M. Arisona, and H. Qu, "Visualizing interchange patterns in massive movement data," *Comput. Graph. Forum*, vol. 32, no. 3pt3, pp. 271–280, 2013.

[23] F. Wang *et al.*, "A visual reasoning approach for data-driven transport assessment on urban roads," in *Proc. IEEE Conf. Vis. Analytics Sci. Technol.*, 2014, pp. 103–112.

[24] D. Liu *et al.*, "SmartAdP: Visual analytics of large-scale taxi trajectories for selecting billboard locations," *IEEE Trans. Vis. Comput. Graphics*, vol. 23, no. 1, pp. 1–10, Jan. 2017.

[25] D. Weng *et al.*, "Towards better bus networks: A visual analytics approach," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 2, pp. 817–827, Feb. 2021.

[26] N. Andrienko, G. Andrienko, F. Patterson, and H. Stange, "Visual analysis of place connectedness by public transport," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 8, pp. 3196–3208, Aug. 2020.

[27] F. Kamw *et al.*, "Urban structure accessibility modeling and visualization for joint spatiotemporal constraints," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 104–116, Jan. 2020.

[28] M. Lu, C. Lai, T. Ye, J. Liang, and X. Yuan, "Visual analysis of multiple route choices based on general GPS trajectories," *IEEE Trans. Big Data*, vol. 3, no. 2, pp. 234–247, Jun. 2017.

[29] S. Havre, B. Hetzler, and L. Nowell, "ThemeRiver: Visualizing theme changes over time," in *Proc. IEEE Symp. Inf. Vis.*, 2000, pp. 115–123.

[30] M. Brehmer and T. Munzner, "A multi-level typology of abstract visualization tasks," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2376–2385, Dec. 2013.

[31] T. Munzner, *Visualization Analysis and Design*. Boca Raton, FL, USA: CRC Press, 2014.

[32] B. C. Arnold and R. A. Groeneveld, "Measuring skewness with respect to the mode," *Amer. Statistician*, vol. 49, no. 1, pp. 34–38, 1995.

[33] Seoul Bike, 2020. [Online]. Available: https://www.bikeseoul.com/.

[34] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang, "Map-matching for low-sampling-rate GPS trajectories," in *Proc. 17th ACM SIGSPATIAL Int. Conf. Advances Geographic Inf. Syst.*, 2009, pp. 352–361.

[35] OpenStreetMap contributors, "Planet dump retrieved from Planet OSM," 2017. [Online]. Available: https://www.openstreetmap.org

[36] H.-S. Park and C.-H. Jun, "A simple and fast algorithm for K-medoids clustering," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 3336–3341, 2009.

[37] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, 1987.

[38] R. J. Rossi, *Mathematical Statistics: An Introduction to Likelihood Based Inference*. Hoboken, NJ, USA: Wiley, 2018.

[39] M. Bierlaire, "PandasBiogeme: A short introduction," Transport Mobility Lab., Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, Tech. Rep. TRANSP-OR 181219, 2018.

[40] K. E. Train, *Discrete Choice Methods With Simulation*. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[41] J. Wood, A. Slingsby, and J. Dykes, "Visualizing the dynamics of London's bicycle-hire scheme," *Cartographica: Int. J. Geographic Inf. Geovisualization*, vol. 46, no. 4, pp. 239–251, 2011.

[42] J. Fekete, D. Wang, N. Dang, A. Aris, and C. Plaisant, "Interactive poster: Overlaying graph links on treemaps," in *Proc. IEEE Symp. Inf. Vis. Conf. Compendium*, 2003, pp. 82–83.

[43] C. Collins, G. Penn, and S. Carpendale, "Bubble sets: Revealing set relations with isocontours over existing visualizations," *IEEE Trans. Vis. Comput. Graphics*, vol. 15, no. 6, pp. 1009–1016, Nov./Dec. 2009.

DongHwa Shin received the PhD degree in computer science and engineering from the Seoul National University, Seoul, South Korea, in 2021. He is a postdoctoral researcher with the Institute of Computer Technology, Seoul National University, Korea. His research interests include information visualization, visual analytics, and geospatial data analysis. He is currently focusing on designing a visual analytics system for analyzing transportation data.

Jaemin Jo received the BS and PhD degrees in computer science and engineering from the Seoul National University, Seoul, South Korea, in 2014 and 2020, respectively. He is currently an assistant professor with the College of Computing and Informatics, Sungkyunkwan University, Korea. His research interests include human-computer interaction and large-scale data visualization. He is especially interested in progressive visualization systems that facilitate the responsive exploration of large-scale data.

Bohyoung Kim received the BS and MS degrees in computer science and the PhD degree in computer science and engineering from the Seoul National University, Seoul, South Korea, in 1995, 1997, and 2001, respectively. She is an currently associate professor with the Division of Biomedical Engineering, Hankuk University of Foreign Studies, Korea. Her research interests include computer graphics, volume visualization, medical imaging, and information visualization.

Hyunjoo Song received the BS degree in computer science and engineering from the Seoul National University, Seoul, South Korea, and the MS and PhD degrees in electrical engineering and computer science from the Seoul National University, Seoul, South Korea, in 2016. He is currently an assistant professor with the School of Computer Science and Engineering, Soongsil University, Seoul, South Korea. His research interests include HCI, information visualization, and gaze tracking.

Shin-Hyung Cho received the PhD degree from the Department of Civil and Environmental Engineering, Seoul National University, Seoul, South Korea, in 2018. He is a postdoctoral researcher with the School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, Georgia. His research interests include transportation planning, human travel behavior, smart mobility, and public transport. He is currently focusing on activity-based modeling, which deals with individual daily travel activities.

Jinwook Seo received the PhD degree in computer science from the University of Maryland, College Park, Maryland, in 2005. He is currently a professor with the Department of Computer Science and Engineering, Seoul National University, where he is also the director of the Human-Computer Interaction Laboratory. His research interests include human-computer interaction, information visualization, and biomedical informatics.