Analyzing the Shifts in Users Data Focus in Exploratory Visual Analysis

Sanad Saha*
Computer Science
Oregon State University
Corvallis, Oregon, USA
sahasa@oregonstate.edu

Nischal Aryal*
Computer Science
Oregon State University
Corvallis, Oregon, USA
aryaln@oregonstate.edu

Leilani Battle
Paul G. Allen School of Computer Science & Engineering
University of Washington
Seattle, Washington, USA
leibatt@cs.washington.edu

Arash Termehchy
Computer Science
Oregon State University
Corvallis, Oregon, USA
termehca@oregonstate.edu

Abstract

Users often begin exploratory visual analysis (EVA) without clear analysis goals but iteratively refine them as they learn more about their data. As an essential step in data science, researchers want to aid EVA by developing responsive and personalized visualization tools. For this, accurate models of users' exploration behavior are becoming increasingly vital. However, many computational models assume that the human exploration behavior is *static*, which goes against the *dynamic* nature of EVA. In this benchmark study, we investigate how users dynamically shift their data focus in EVA and seek to find the best online learning methods for modeling users' data focus shifts. Through empirical analyses, we find reinforcement learning algorithms are better in this regard than existing approaches from visualization research. Furthermore, we discuss our findings and their impact on the future of user modeling for visualization system design.

CCS Concepts

• **Human-centered computing** → *User models.*

Keywords

Human-centered Computing, Visual Analytics, User Modeling, Reinforcement Learning

ACM Reference Format:
Sanad Saha, Nischal Aryal, Leilani Battle, and Arash Termehchy. 2025. Analyzing the Shifts in Users Data Focus in Exploratory Visual Analysis. In *30th International Conference on Intelligent User Interfaces (IUI '25), March 24–27, 2025, Cagliari, Italy*. ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3708359.3712154

*Both authors contributed equally to this research.

IUI '25, Cagliari, Italy

ACM ISBN 979-8-4007-1306-4/25/03
https://doi.org/10.1145/3708359.3712154

1 Introduction

Data analysts interactively query **visual exploration systems (VESs)** such as Tableau [2] and PowerBI [1] to explore large datasets and discover insights [11]. This iterative and complex process is known as **exploratory visual analysis (EVA)** [36]. EVA is particularly challenging as analysts often explore new datasets with unknown structures and content. Complexity increases as analysts may start EVA with vague analysis goals [10], like finding interesting insights [36, 71].

Initially, a user may not know what information is interesting or where to find them. In each interaction, the user focuses on a specific data area to find insights or learn something that might help in future interactions. In this work, we use the term ***data focus*** *to refer to the specific data area the user is focusing on at any given time.* However, the scope of data focus can vary. For example, in tabular data, data focus can be a specific data point (in row x, column y) or an entire data column y. In this research, *data focus is defined based on the exploration task, dataset, and VES.* For instance, if a VES recommends visualizations based on data columns (attributes) chosen by the user, it is more effective to analyze the user's data focus in terms of attributes. Likewise, if the VES emphasizes specific row filters or data points [32], then data focus should be modeled accordingly. Consider the following scenario *where a user's data focus (attributes) changes as EVA progresses.*

Example 1. *Alice is exploring the Birdstrikes dataset [67] containing records of wildlife strike incidents with aircrafts. Her task is to find patterns to improve aviation safety and airline revenue (Figure 1). However, Alice is unfamiliar with this dataset. So she starts by randomly examining visualizations to learn more. After some interactions, she notices that the* `number_of_incidents` *has been decreasing since the year 2000. Motivated to find other trends using attribute* `flight_date`*, she explores visualization showing* `repair_costs` *over the years. Without finding conclusive trends, she decides to shift to other attributes. She observes a pattern involving* `wildlife_size`*: small birds cause more collisions. Consequently, Alice explores visualizations involving* `wildlife_size` *and discovers that repair costs are significantly higher for collisions with large animals due to the substantial damage they cause. Noting this insight, she continues her interaction to complete the task.*

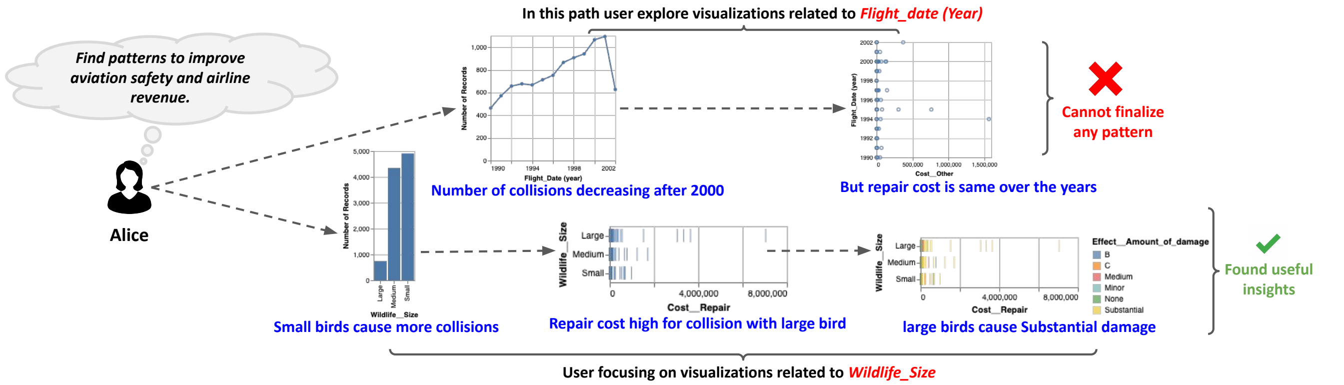


Figure 1: A user exploring Birdstrikes dataset [67].

Alice's strategy for shifting her data focus changes based on what she learns from the data, her information needs, etc. Over time, her shifts become more precise, especially when she identifies an interesting attribute. Even after an unsuccessful exploration of `flight_date`, her data focus shifts are less random because she understands the dataset better. Eventually, she decides to shift data focus to `wildlife_size`, an attribute she encountered earlier.

By recommending useful visualizations, we can improve EVA experience for users like Alice. For that we need to understand users exploration strategies and data interests [12, 69]. visualization researchers have studied modeling users' data focus shifts to predict their future actions and data point interests [7, 42, 46]. Besides recommending visualizations, these models can be used to prefetch data areas to increase interactivity [7] and suggest relevant data regions for insights [27]. These approaches learn data focus shifts from offline interactions using machine learning [7], rule-based pattern matching [27], or online learning algorithms that update parameters after each interaction [42, 46].

However, as users learn more about the data, their strategies for shifting data focus may evolve to complete EVA tasks more efficiently and effectively [10, 36, 54, 55]. Thus, knowing Alice will interact with `wildlife_size` in the next step may not be enough. When Alice is exploring randomly, VES should recommend a diverse set of visualizations, providing an overview of the dataset. But in later stages of exploration, when Alice is more selective, it might be better to recommend visualizations showing `wildlife_size`'s relationship with other attributes. Therefore, analyzing and modeling the dynamic shifts in users' data focus complements current models for predicting future actions or data points [32, 42, 46]. Additionally, there is a lack of empirical analysis on how well current algorithms adapt to evolving data focus shifting strategies.

Moreover, existing approaches often overlook online learning algorithms, such as reinforcement learning (RL). Many online RL algorithms have shown promising results in modeling decision-making strategies in cognitive psychology, neuroscience, etc. [17, 44, 45]. In EVA, users may change their data focus strategies in response to what they learn from the data. They may choose suboptimal strategies, like exploring areas that do not contain any insight. However, they help to understand the dataset better and maximize long-term rewards. *Such similarities with online RL algorithms raise an important question: are RL algorithms better suited for modeling these dynamic shifts in data focus?*

The main goal of our paper is to present a benchmark study using a variety of offline and online learning algorithms, which answers: What are the best methods for modeling shifts in users' data focus during EVA? We mirror Feng et al.'s [24], Gathani et al.'s [25], and He et al.'s [32] approach of leveraging prior user studies to address our research questions. Rather than designing a single study covering a single tool, our analysis spans three influential studies [7, 10, 36] from the visualization community that applies distinct approaches to capture exploration behavior. We perform statistical tests to determine *whether users' strategies for shifting data focus evolve* in our selected EVA datasets, serving as a sanity check for comprehensive benchmark analysis.

Our analysis is breadth rather than depth-focused, and exploratory in nature. Through this work, we make the following contributions:

- To evaluate how well different algorithms model users' dynamic shifts in data focus, we analyze their ability to predict users' future data focus shifting strategies. We include a comprehensive list of algorithms and user modeling techniques from existing visualization papers, such as Hidden Markov Models [32, 42, 46]. Our analysis shows *RL algorithms significantly outperform existing methods*, suggesting that current EVA models can be improved.
- We present common findings and challenges from our empirical investigations and suggest future research directions for developing more accurate adaptive data focus models for VESs.

2 Related Work

Understanding the user during Exploratory Visual Analysis (EVA):. We see analyses of users' acquisition of knowledge, specifically in the form of insights [10, 30, 33, 36]. However, these approaches focus on proposing VESs that increase accuracy insights or how fast users can generate insights [10] rather than analyzing the evolution in users' exploration behavior. Some researchers have proposed visual analysis taxonomies that map users' activities in EVA with users' high-level reasoning process [10, 25]. Several works present users' cognitive frameworks or mental models for analyzing datasets through VESs [29, 38, 48]. However, they do not address how users mental models may evolve as they learn about the data.

More recent studies acknowledge the dynamic nature of users' exploration behavior. For instance, Ottley et al. [46] proposed a hidden Markov model (HMM) approach to maintain a belief over users' evolving *hidden* attention and actions in a *point-based* visualization setting. Utilizing user clicks, they update the model and use particle filtering to infer a set of *top-k* data points for the next click. Monadjemi et al. [42] proposed a Bayesian learning approach to also predict *top-k* data points. Instead of predicting exact data points, we use user interactions to learn how they shift their data focus and predict the next shifting strategy. Because predicting data points of interest may not scale to large datasets [42]. Typically, visualization papers propose systems focusing on unique EVA scenarios and user models tailored for such instances [70]. In contrast, we adopt breath-focused analysis similar to [32] with the goal of analyzing and modeling data focus shifts in various EVA scenarios.

Researchers model users' low-level interactions into high-level exploration phases or analysis patterns to improve how systems support EVA [7, 27, 28]. These offline models aim to infer users' exploration goals, prefetch corresponding data regions [7], or recommend tailored visualizations [28]. We argue that such static models may ignore users' evolving information needs and produce suboptimal recommendations.

Researchers have also studied users' eye movements to understand user's analysis strategies. For instance, Netzel et al. conducted empirical analyses during users' exploration of metro maps [43]. Similarly, Chen et al. examined users' gaze behavior while exploring wrapped graph visualizations [19]. In another study, Borkin et al. analyzed participants' eye movements to identify focal points and assess their ability to recognize and recall information from visualizations [51]. In contrast to eye movements – data passively generated by the user [69], our analysis is focused on users' active interactions with data attributes during visual analysis.

User Models Facilitate Exploratory Analysis: Researchers have used user models to analyze and improve the EVA experience. Reda et al. proposed a Markov chain-based framework to evaluate changes in user behavior influenced by an ensemble of high-resolution interfaces, equipped with a query-by-example tool and large screen displays [52]. Todi et al. applied model-based RL to customize user interfaces according to individual preferences [62]. Zhou et al. modeled users' cognitive attention, i.e., analytic focus, to recommend relevant articles that support insights derived from visualizations and suggest future opportunities. As users' analytic focus shifts during EVA, the proposed model dynamically updates the importance of explored contexts after each interaction [72]. Taramigkou et al. used HMMs to model and infer users' exploration patterns to improve exploratory search by recommending useful actions. The authors discovered that incorporating the interactional context and personality traits helps outperform HMMs trained on global user data [61]. Shrinivasan et al. proposed a retrieval algorithm to recommend relevant notes, views, and concepts from users' past analyses to help users connect past observations to current investigation [58]. Oulasvirta et al. investigated how users adapt their interactions to optimize expected utility within the constraints of their preferences, cognitive limits, and environment—a concept they call "computational rationality" [47]. They proposed a unified framework that employs RL and partially observable MDP to model and explain user behavior. In this benchmark study, we focus on modeling users' shifts in data focus, as VESs can generate useful visualizations by understanding users' data interests and helping users in EVA.

Online Learning Algorithms to Model User's Evolving Exploration Strategies: Recent research indicates that users learn and modify their keyword queries to express focused intents while interacting with a system [40]. This evolving behavior can be modeled using online learning algorithms. Cen et al. modeled users' evolving information-search strategies from scholarly databases using RL [18]. Luo et al. have modeled users' exploration-exploitation policies in formulating keyword queries for document retrieval [39]. Unlike data querying, users often lack a predefined and concrete intent during EVA. Consequently, EVA presents a significantly larger action space requiring users to make more complex decisions. We posit that the evolutionary nature of EVA provides a more natural setting for the users to learn and shift their data focus.

Use of Reinforcement Learning (RL) and Statistical Metrics in EVA:. Some VESs statistically identify and suggest the underlying data patterns to users. These systems use statistical measures like *diversity, interestingness, and coherency* [4, 5, 21] in their models to determine *how interesting a visualization is from users' perspectives.* However, when utilizing them, they assume these factors have the same contribution, wherein the demand for visualizations based on these elements evolves. Prior studies underscore the need for a framework that utilizes user interaction history for personalized recommendations [23, 50]. Additionally, we see Deep-RL models trained by expert user demonstrations to aid future analysts in similar exploration tasks by automatically generating exploration sessions and relevant recommendations [5, 56]. However, relying solely on static statistical metrics, e.g., KL-divergence, and features from expert demonstrations, to identify and suggest visualizations can be problematic. These approaches may not accurately capture/adapt to users' dynamic shifts in their data focus strategies or information needs to gain more insights.

Exploration as a Markov Decision Process (MDP): Because of the popularity of RL algorithms, it has become common to model problems using the MDP framework. MDP-based RL algorithms have been used to explain the workings of the human brain [26] and decision making processes [20, 44, 45]. MDP has also become popular in modeling data exploration [39] and visual analytics problems [5, 21, 41, 57]. In these works, the RL algorithms try to find the optimal policy that replicates users' analytical processes and use that policy for visualization recommendations.

3 Considerations for EVA-Focused User Studies

Users' shifts in data focus depend significantly on the characteristics of EVA tasks and VESs. Therefore, our benchmark study encompasses diverse EVA tasks and characteristics that capture users' different information needs (introduced in subsection 3.1). However, instead of conducting a new user study, we *reuse* well-known studies by Liu et al. on the **imMens** system [36], Battle et al. on

Characteristics	**Tableau user study[10]**		**imMens user study[36]**	**Voyager user study[70]**	
	Task [T1, T2, T3]	**Task [T4]**		**Task [T1, T2]**	**Task [T3, T4]**
Open-endedness	Focused	Open-ended	Open-ended	Focused	Open-ended
Task Complexity	Analyzing visualizations generated by Tableau based on users' query on dataset		Interactive querying using imMens actions on summarized plots	Specifying data attributes and interacting with visualization recommendations	
Prior experience	15 minutes with a different dataset		15 minutes with datasets	10 minutes demo with a different dataset	

Table 1: Selected user studies with different EVA task characteristics

Tableau [10], and Zeng et al. on a simulation of **Voyager** [70] to attain EVA diversity (see Table 1).

We reached this decision after consulting with experts from the visualization community, including the authors of the selected studies. At the start of the project, we contacted the authors of the Voyager user study [70]—one of the largest EVA user studies to date—for guidance on participant selection and study procedures. Their study involved 72 professional analysts and knowledgeable students and took over 12 months to complete. Still, the Voyager user study has some limitations, such as using a concise version of Voyager as VES and small datasets. Whereas the Tableau user study [10] uses more realistically sized datasets and facilitates users to be more involved in selecting data areas and visual encodings, thereby adding complexity to the EVA process.

Furthermore, we learned that conducting a single large-scale study to cover these diverse tools, datasets, and characteristics of EVA tasks with experienced analysts is impractical for our benchmarking approach, if not infeasible. Not to mention, EVA studies with a much smaller scope are considered significant contributions, deserving a separate paper, as seen in the works of Battle and Heer [10], Liu and Heer [36], Zgraggen et al. [71], and Zeng et al. [70].

3.1 Characteristics of EVA User Study Tasks

Task Open-endedness: Researchers categorize exploration tasks into two groups based on how clear users' objectives are [10, 42, 70]. In **open-ended tasks** analysts do not have a clear intent or hypothesis. For example, in the imMens user study (Table 1), participants are asked to find *interesting* information from the given datasets. Since "interesting information" is a vague term, users are uncertain about what and where to search. As a result, they may explore different data areas to understand the data and report insights that seem interesting to them.

In **focused tasks**, analysts have a precisely defined goal and exploration path that often contributes to a broader objective. For instance, in the Voyager user study task T1 (Table 7), users are asked to find the maximum number of movies for the genre "creative" and the source "book/small story.". Users know what information to retrieve and which data area to explore as the question includes corresponding column names. Nevertheless, with the large data size, users may need additional interactions to find the exact information.

Task Complexity: How users operate a VES to generate insights and the amount of information displayed in the interface increases task complexity, thereby impacting users' exploration strategies in EVA [3, 31, 35]. Voyager aims to recommend the best possible visualizations, whereas Tableau [2] is a more complex system. Tableau has a large action space, allowing users to select, filter, and visualize data in a highly customizable manner. In contrast, imMens summarizes the data into pre-defined visualizations, requiring users to perform only four actions to query these visualizations and identify patterns.

Prior Experience: Users' prior knowledge about the dataset and familiarity with the exploration interface may influence their shifts in data focus, thereby their exploration strategy. While users have a limited time to get familiar with the dataset in imMens, Voyager and Tableau analysis tasks require users to explore an unseen dataset [49].

3.2 Analysis Methodology

The visualization literature has precedents for benchmarking models using existing experimental data [24, 25], including comparisons across multiple studies [32, 42]. We believe our analysis aligns with established research practices. Unlike typical ML studies that showcase the superiority of a single algorithm by the highest accuracy, we adopt a compare-and-contrast approach. The idea is, 'Whatever learning algorithm is best for modeling should have a consistently high ranking across EVA conditions'. The task open-endedness and other EVA characteristics (section 3) are well-established as key factors influencing user behavior, thereby affecting algorithm performance [8, 9]. We incorporate these factors into our modeling by adopting different techniques and narrowing the scope of data focus to address the same research questions. The analysis steps are discussed below:

3.2.1 Scope of Data Focus: Defining the scope of data focus for each user study is crucial, as data manipulation in VES can vary widely—from selecting a data column to filtering parts of it. Tableau and Voyager user studies have emphasized the importance of analyzing attribute selection for characterizing user behavior in EVA and recommending visualizations. For these studies, we analyze shifts in user data focus via the attributes selected during each interaction.

However, the EVA process differs for imMens, designed to explore datasets with millions of data points. imMens displays five visualizations showing relationships among the traveler check-in attributes—User, Date, Time, Latitude, and Longitude—which facilitates interactive querying and enables users to aggregate or filter data to understand groups of data points. Therefore, imMens operations—such as pan, brush, range-select, and zoom—are indicators of shifts in users' data focus.

3.2.2 Formalizing the Exploration Problem. In this step, we formalize how users shift their data focus to generate insights. Specifically, though not limited to, the following EVA steps: 1. What data area is the user currently focusing on? 2. Based on her current knowledge, what action will the user choose? 3. Consequently, the VES suggests a new visualization or changes the interface. What does the user learn from that? 4. How does the user update her current knowledge to gather insights?

These steps guide the development of the learning algorithms discussed in the next section. For instance, in Step 1, we define how to encode the information. In Step 3, we establish a process for quantifying the information learned.

3.2.3 Statistically Analyzing Shifts in Data Focus: We first validate whether users' data focus shifting strategies change across the selected studies to benchmark the models accurately.

We use mixed-effects models, which are well-regarded in the visualization community for analyzing users' exploration strategies [10, 36]. We divide each user's exploration session into two halves, ensuring each contains sufficient interactions to effectively capture the strategies while maintaining consistency across users' tasks and datasets, allowing a uniform approach to significance testing. While users may shift their data focus strategies at different stages of an EVA session, this division also represents the most fundamental level at which we would expect to observe changes in strategy. We investigate: *Do shifts in users' exploration strategies significantly change during EVA (initial half vs. later half)?* When sufficient data is available, a more granular sampling of the exploration session yields consistent results.

We analyze three main factors: (a) *exploration half* (initial vs. later); (b) *task type* (open-ended vs. focused); and (c) *analysis scenario* (participants and datasets). In our mixed effects model design, fixed effects are factors consistent across all groups, such as the exploration half and task type. Random effects vary across different groups and help account for data variability that fixed effects alone cannot explain.

To determine the importance of the exploration half, we use likelihood-ratio tests (lr-tests) [66]. We build two models: a full model that *includes* all fixed effects and a null model that *excludes* the exploration half. By comparing these models via lr-tests, we obtain p-values that indicate *whether users' exploration strategies for shifting data focus change between the two exploration halves.* We use the lme4 R library [6] for our analysis.

3.2.4 Modeling Shifts in Data Focus: Some visualization researchers may learn **individual** user models, facilitating adaptation to personal preferences. These models are tuned online, using past interactions of that specific user for predicting future interactions, detecting exploration bias, and recommending visualizations [42, 46]. While this approach offers tailored adaptation, it assumes that we have sufficient if not any data for a specific user.

Another approach is to learn a single model from multiple users' interaction sessions with the assumption that training on a diverse **population** helps generalize to a new user. This is specifically beneficial for machine learning-based models that require large amounts of training data [7, 21, 57]. In our benchmark study, we test both modeling assumptions. *We aim to investigate how well the online learning algorithms trained on multiple users (population) adapt to individual preferences compared to individual models.*

The models are empirically evaluated based on their accuracy in predicting actions, highlighting how users will change their data focus. Here the interaction lengths between predictions and ground truth are identical. Our preferred metric, i.e., **accuracy**, is consistent with prior user model evaluations [16, 32, 42], including the visualization baselines used in our paper [42, 46], which also utilize accuracy and accuracy@k.

4 Learning Algorithms for User Model

In this section, we present the learning algorithms used in the benchmark study.

4.1 Reinforcement Learning (RL)

To employ RL for modeling users in EVA, we formulate *users' interactions with the VES to discover insights* as a Markov Decision Problem (MDP).

MDP: MDP is a mathematical framework for modeling sequential decision-making problems by defining interactions through states, actions, and rewards. RL algorithms learn optimal *policies* within the environment, i.e., deciding which action to take in a given state and updating the policy based on the rewards received. The structure of the MDP will vary depending on the EVA characteristics of our selected user studies. However, we follow the general framework outlined below to maintain consistency:

In EVA, user (RL algorithm) learns to find insights through repeated interactions with the VES (environment). The user (RL algorithm) has an exploration policy (π) that guides her shifts in data focus (action) based on the current data area (state). VES (environment) provides visualizations that contain insights (rewards). The user uses these rewards to update her current policy (π). Eventually, the user learns an optimal policy (π^*), i.e., better decision-making strategies for shifting data focus to find the desired information. Like an RL algorithm, user's main goal is to maximize the amount of insight (reward).

Selection of RL Algorithms: We include RL algorithms, broadly classified into value-based, policy-gradient-based, and Actor-Critic methods.

Value-based RL algorithms learn a *value function (vf)*, which outputs the expected discounted reward of a state or a state-action pair. The algorithm's *policy*, which determines which action to take in a particular state, depends on this *vf* and is updated through trial and error. Examples of such algorithms include Q-learning and SARSA.

Other researchers [14] advocate for algorithms that directly learn a policy without a *vf*. A *vf* may still be used to learn the parameters defining a policy but is not required for action selection [59]. These algorithms are called *value-free or policy-gradient* algorithms, e.g., Reinforce. Bennett et al. [13] demonstrated that combining value-based methods with policy gradients, known as Actor-Critic, also yields promising results in explaining human behavior. Now, let us briefly discuss our selected RL algorithms:

Q-learning (QLearn): iteratively updates a *vf* called the Q-function. QLearn learns the optimal policy through trial and error with a ϵ-greedy policy. That is, choosing either a random action with a small probability, ϵ, or the action with the highest estimated reward with probability, 1 - ϵ. QLearn aims to learn a policy that maximizes the expected reward in an environment [64] based on the Q-function update rule:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)] \quad (1)$$

Where $Q(s_t, a_t)$ is the value of taking action a in state s at time t. α is the learning rate and controls for the degree of Q-value (Q) update, r_t is the reward received at time t, γ is the discount factor to give more weight to r_t than future rewards, and s_{t+1} is the next state. The last hyper-parameter in this algorithm is ϵ for ϵ-greedy.
SARSA: is value-based like QLearn [53]. But unlike Qlearn, which updates its Q using the action that yields **maximum** Q-value in the next state, SARSA updates Q by following the action based on the ϵ-greedy policy:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \quad (2)$$

SARSA has the same hyperparameters as QLearn : γ, α, ϵ
Reinforce is the simplest policy-gradient method [59]. It directly improves the policy based on the observed rewards without vf [65]. Neural-network-represented parameters define the policy that is improved by following the gradient of the expected reward. Reinforce uses γ and α as hyperparameters.
Actor-Critic extends Reinforce by improving the policy through learning a value function in parallel. It combines value-based (critic) methods with a policy-gradient side (actor) [34]. The actor decides which action to perform based on a given state. The critic uses the vf to tell the actor how good the performed action was and how to update the policy parameters. This algorithm has the same hyperparameters as Reinforce.

4.2 Current baselines for user modeling in visualization

User models in visualization have been used for a broad range of applications, such as studying biases [42, 46], predicting user's personality attributes [69], and even prefetching next data tiles for responsive VES [7, 22]. For our analysis, we try to include as many models as possible as long as their assumptions are compatible with our benchmark. For example, a key assumption for the Competing Models approach [42] is that users interact with a set of *point-based visual data*, takes as input a *complete underlying dataset* with d attributes and compute 2^d Bayesian models. While it was not prohibitive for the authors to compute exact probabilities (largest model space: 2^7), as the authors note, doing the same for larger datasets can be infeasible and may require additional assumptions or sampling methods [42]. However, we include a more general algorithm that is independent of the underlying dataset but shares the same Bayesian update framework. Specifically, our evaluation includes the following established approaches for comparison.
Momentum: This model assumes that the user's next action will be the same as her previous action in that state.
Hidden Markov Model (HMM): HMM assumes that the user's behavior evolves according to a Markov process—that is, the current state depends solely on the previous state. While an MDP is defined by states and actions, HMM is characterized by hidden and observable states. To align our user modeling problem and evaluation via predicting users' shifts in data focus (actions), we consider actions as hidden states, and the observed state at time t comprises the data area the user is currently visualizing. We used the nltk.tag.hmm module [15] to train an HMM model on the training data and then utilized it to predict action sequences based on the test sequence of observations.
Bayesian Learning: This model assumes a uniform prior on action probabilities, observes new user interaction data (state-action) data and updates the state-action probabilities in light of new observations. At each step, the model picks the next action by sampling from its updated probability distribution.
Support Vector Machine (SVM) [63]: SVMs have been employed in modeling user behavior in EVA, leveraging offline data to prefetch data tiles [7], predict future mouse movements, and infer cognitive traits[16]. In this work, we use information about users' current data areas (states) as input features and employ SVM to predict corresponding actions. To ensure a fair comparison with online RL algorithms and emphasize the use of online learning algorithms to model users' shifts in data focus we include an online version of SVM. We implement **OnlineSVM** with `sklearn's` partial-fit function on top of OfflineSVM.

4.3 Simple Decision Making Heuristics

We use the Random strategy as our baseline and some heuristics that users often adopt in decision-making scenarios.
Random Strategy: The agent always picks an action *uniformly at random* from the available choices, irrespective of the rewards received.
Greedy Strategies: The user picks an action for immediate success based on previous experience, choosing the action that has yielded the highest reward so far [59].
Win-Stay Lose-Shift (WSLS): Repeats a successful action until it no longer yields rewards, then switches to other actions with equal probabilities. WSLS is a popular heuristic to model human learning in games [60].

5 imMens user study

Unlike Tableau and Voyager systems, imMens does not require users to select attributes to generate visualizations.

Therefore, we first focus on the imMens user study [36], which allows us to analyze users' shifts in data focus in a *restrictive setting*.

5.1 Overview of Exploration Task

5.1.1 Analysis Task. 16 participants in this user study report *interesting findings*, defined as surprising events, data abnormalities, etc. [36]. They explore (a) travelers' check-in data from *Brightkite*, a location-based service, and (b) U.S. flight performance data. The Brightkite dataset includes travelers' check-in dates and locations, while the U.S. flight data covers airline carriers, flight dates, and arrival and departure delays.

5.1.2 Characteristics. The analysis task in this user study is open-ended, as the definition of *interesting findings* is not precise, leading to uncertainty about which data areas to explore. Although the authors provide some examples of interesting findings, the guidance is vague, leaving participants unsure about what specifically to search for. Participants get 15 minutes of *prior experience* with the datasets and interface, which may impact what they learn to a certain degree compared to scenarios where they proceed EVA without familiarity.

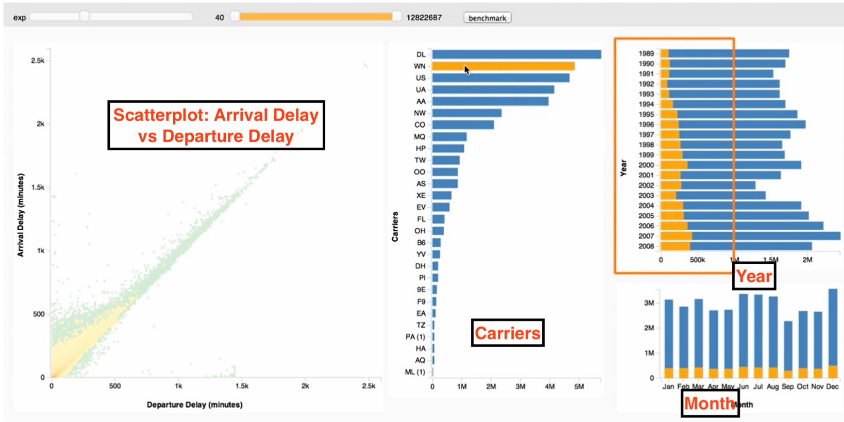


Figure 2: imMens user exploring flight performance data

5.1.3 imMens Interface presents users with four fixed visualizations for exploring the flight performance dataset: (1) Scatter plot: showing the relationship between arrival and departure delays. (2) Carriers: A bar chart for US airline carriers. (3) Year: A bar chart showing flights across years. (4) Month: A histogram with flight over timeframes (Figure 2).

Users interact via *brush & link, pan, zoom, and select* operations [37], with changes in one visualization updating the others. For instance, if a user *selects 'Year = 2003' in the visualization 'Year'* (Figure 2), the data is filtered, and visualizations update to show information specifically for the year 2003.

Details on the visualizations generated for the travelers' check-in dataset can be found in the original paper [36].

5.1.4 imMens Interaction Log contains users' imMens operations and the visualizations they interact with at each time step.

Additionally, it has users' verbal feedback, where they explain their actions, findings, and reasonings, e.g., what type of information they want to find, if they have discovered anything new, etc.

5.2 Formalizing User Exploration Problem

5.2.1 User Activities. At each step, a user selects a visualization (i.e., data area), applies imMens operations, and analyzes the interface for insights. The information obtained from the interface acts as a reward, guiding the user to either continue focusing on the current data area or shift their data focus elsewhere. For example, they might select airlines one by one in the "Carriers" visualization and observe changes in the "Year" and "Scatterplot" views, helping them identify patterns like when an airline started or its delay trends. If the user discovers any *interesting findings*, she reports them before continuing their exploration.

5.2.2 Extracting Features from Verbal Feedback: Liu et al. annotated users' verbal feedback into seven categories [36], revealing user exploration goals, thus offering insights into their data focus-shifting strategies. We leverage user feedback logs and these annotations for feature and reward engineering, enriching our user modeling process.

Out of the seven categories, we select four as features for our model, as they encompass 95% of the feedback. They are, *observation:* Users discover a piece of information about the data originating from a single visualization. Users may stay in the same visualization until they believe no further insights can be gleaned without switching. *Generalization:* Users aggregate information from multiple visualizations and report. *Hypothesis:* Conjecture about the data, made to steer exploration or explain observation/generalization. Thus, indicating possible shifts in data focus. *Question:* Indicates users' desire to explore different data areas. Users' data focus-shifting strategies depend on whether the desired information can be found by revisiting previously explored areas or requires venturing into new ones. More information on this categorization process and examples are available in [36].

5.2.3 Modeling Exploration Using MDP: We use an MDP to model the user's decision-making process while using imMens, aiming to maximize insight discovery. In MDP terms, the *exploration policy* defines how users select the optimal imMens operation to shift their data focus based on the current data area.

However, adhering to a single policy is not optimal. The user may effectively change the policy as she exhausts its utility for insights or learns more about the data, discovering better strategies for finding insights. For example, a user may pan the scatterplot to understand the relationships between arrival and departure delays over time and later decide to identify which airlines experience the most delays. Next, we define the MDP components:

States: The state s_t includes information regarding the user's selected visualization at interaction t and the corresponding exploration phase, introduced in subsubsection 5.2.2. We encode this information in a one-hot vector (Figure 3).

Actions: imMens operations *pan, zoom, brush, and range select*

Rewards: In interaction t, the user receives reward r_t based on the information obtained from the interface after performing action a_t. This reward $r_t \in [0, 1]$ reflects the contribution of a_t towards finding insight. When users report new information through observation or generalization, they receive a reward $r_t = 1$. Similarly, when users generate insights that lead to hypotheses or questions, they receive $r_t = 1$ for propelling exploration. For the RL algorithms

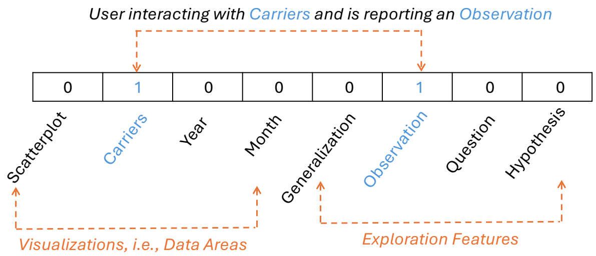


Figure 3: One-hot encoding representation of the state in imMens, capturing the selected visualization and corresponding exploration feature during an interaction.

to better adapt to users' data focus during intermediate interactions, we assign a small reward $r_t = 0.2$.

5.3 Statistically Analyzing Shifts in Data Focus

Following subsubsection 3.2.3, we analyze the imMens actions users use to shift their data focus. We calculate the probability of each action for the two exploration halves for the significance test. The mixed models treat the exploration half as a *fixed effect* and the users and datasets as *random effects*.

The average number of interactions across all users is 447, with a standard deviation of 223.98. The action distribution is as follows: 36.51% for *Brush*, 33.12% for *Pan*, 2.6% for *Zoom*, and 27.77% for *Select*.

The results of the *likelihood-ratio test* between the *full model* and *null model* are shown in Table 2. We find that the exploration half—whether a user is in the initial or later stages of their exploration—significantly influences the probability of selecting an action, indicating shifts in users' focus between data areas (visualizations). For example, initially, users may repeatedly use *brush* to explore the bar chart 'Carriers' and 'Year' (Figure 2) to find out when the airlines start/end their business. Later, they may repeatedly use *pan* and *zoom* to identify patterns between arrival and departure delays.

Actions	χ^2	P-value	Significance
Brush	6.4033	0.01139	*
Pan	8.1423	0.004324	**
Zoom	13.312	0.0002624	***
Select	6.9957	0.00817	**

Table 2: Significance test results for each action. Significance: *** p <0.001; ** p <0.01; * p <0.05;

5.4 Performance Evaluation

5.4.1 Evaluation Procedure: In this study, we assess the performance of different learning algorithms in predicting a user's next action for shifting data focus. As outlined in the analysis methodology (subsubsection 3.2.4), we evaluate these algorithms in two different settings:

Individual Setting: We use an 80-20 train-test split for this setting. The first 80% of each user's interactions are used for hyperparameter tuning (if required, see section 4) and training the learning algorithms, while the remaining 20% are reserved for testing. This helps us evaluate how well the model predicts actions based on strategies from the *same user alone*.

Population Setting: In this case, we use the leave-one-out cross-validation method. The learning algorithms (see section 4) are tuned and trained on the strategies of all other users except the test user. This helps us evaluate how well the model predicts actions given additional data from *other users*.

To ensure consistent comparison between the two settings: (a) we report the test accuracy for the population model over the last 20% of the test user's interactions (b) before making predictions, the population model processes the same initial 80% of the test user's interactions in an online fashion, aligning with the approach used in the individual setting.

5.4.2 Results: To facilitate easier understanding, we aggregate the prediction accuracy of each method *across all EVA tasks and datasets*. Figure 4 presents the experiment results for both settings. To make the plots clearer, we only show the most effective RL model, a simple heuristic, and the current baselines from visualization research. Full results are available in the supplementary materials.

For this study, we find that RL models are slower to adapt to users' shifts in data focus. In the individual experiments, the accuracies are Q-learning (70.81%), SARSA (67.25%), Actor-Critic (67.88%), and Reinforce (64.06%). For the population experiments, the accuracies are Q-learning (80%), SARSA (78%), Actor-Critic (74.94%), and Reinforce (71.88%). We present the best-performing RL algorithm, Q-learning, in Figure 4. However, Momentum and the OnlineSVM outperform all RL algorithms in both settings.

Upon investigation, we attribute this issue to the type of feedback provided to the RL models. Typically, RL models receive feedback on the predicted action and gradually learn the optimal action for each state. Therefore, the RL models predict sequences of interactions one by one, receiving rewards depending on whether their predictions are correct. However, models such as OnlineSVM and Momentum directly use the ground truth action that the user took to adapt their models. RL models lack this direct information, putting them at a disadvantage.

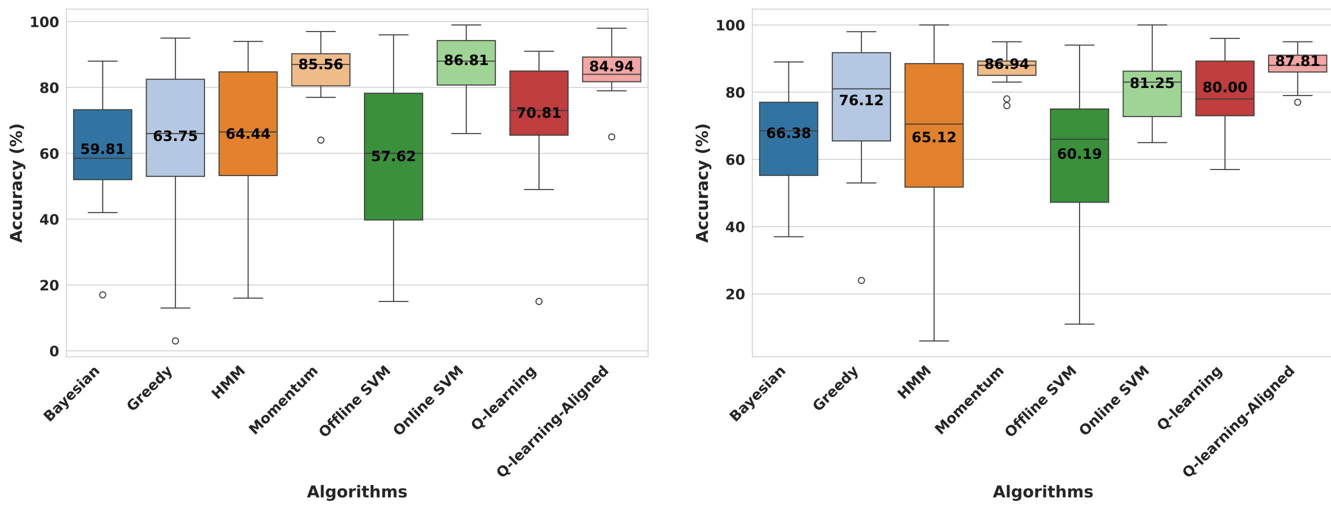
To address this, we modified the best-performing RL model, Q-learning, by assigning a positive reward to the ground truth action, in addition to providing feedback on the predicted action. This extension, named **Q-learning-Aligned (QLearn-Aligned)**, outperforms all other methods on average across both experimental settings.

However, the performance of the Momentum model does provide insights into users' exploration strategy using the *restrictive* imMens interface. When users focus on a particular visualization, they tend to repeatedly use the same operation until their information needs are satisfied.

Finally, these results highlight the importance of online adaptation in modeling users' dynamic data focus shifts. Notably, OnlineSVM shows a 29.19% accuracy improvement over OfflineSVM in the individual model and 21.06% in the population model.

6 Tableau user study

In this study by Battle et al. [10], participants perform a series of EVA tasks (Table 3). These tasks represent a common exploration



(a) Individual Experiments

(b) Population Experiments

Figure 4: Performance results for two experimental settings in the imMens user study.

progression by which analysts maximize the effectiveness of their EVA sessions.

6.1 Overview of Exploration Task

6.1.1 Analysis Tasks. 27 participants use Tableau [2] to complete a series of analysis tasks. Their experience with data analysis and Tableau expertise varied widely. The analysis tasks are from the following dataset: (a) Weather station reports on weather metrics (35 columns, 56.2M rows), (b) U.S. domestic flight performance data (31 columns, 34.5M rows), and (c) Aircraft striking wildlife reports, (94 columns, 173K rows) [10].

6.1.2 Task Characteristics. Let's examine how Battle et al.'s (Table 3) subtask design captures the natural exploration of EVA at a fine granularity. Task T1 captures users' initial exploration to learn the data attributes, facilitating a general understanding of the dataset. Tasks T2 and T3 investigate the statistical relationships or existing patterns between data variables. Finally, task T4 encapsulates more sophisticated explorations, such as prediction and causality analysis. Tasks T1 to T3 are focused, while T4 is open-ended. Users in this study lack prior knowledge of the datasets. To explore datasets using Tableau, users must select data columns (attributes) and search for the most suitable visualization to extract insights, adding overhead to the insight-searching process compared to imMens.

Task	Task Description
T1	Consider the following weather measurements: Heavy Fog[Heavy Fog], Mist [Mist], Drizzle [Drizzle], and Ground Fog [Ground Fog]. Which measurements have more data?
T2	How have maximum temperatures [T Max] and minimum temperatures [T Min] changed over the duration of the dataset (i.e., over the [Date] column)?
T3	How do wind measurements [High Winds] compare for the northeast and southwest regions of the US?
T4	What weather predictions would you make for February 14th 2018 in Seattle and why?

Table 3: Analysis tasks for Weather dataset

6.1.3 Interface and Interaction Log. Users choose which attributes to explore and add them to the Tableau worksheet (Figure 5). Tableau then suggests visualizations with different visual encodings. The interaction log records users' interactions with Tableau, such as selected attributes and visualizations.

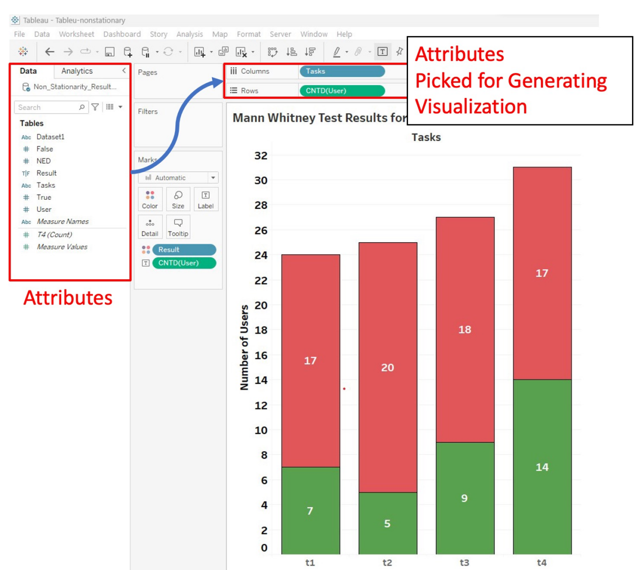


Figure 5: Analyzing attributes using Tableau interface

6.2 Formalizing Exploration Problem

6.2.1 User Activities. To analyze how users shift their data focus in Tableau, we examine how they select and modify attributes during analysis tasks. Users choose attributes to generate visualizations, then either (a) spend more time exploring the current data area or (b) modify the attribute set to shift focus. This process continues until they reach their desired insights. As users explore, they gain deeper understanding of the data, helping them make informed decisions about shifting their data focus.

6.2.2 Modeling Exploration Using MDP: Assuming a user's exploration session has T interactions, we define MDP components for Tableau exploration tasks.

States: State s_t represents a user's data focus in interaction $t \in [1, T]$, *specifically the attributes she has selected to generate a visualization.* The state is represented as a one-hot encoding, capturing information about the selected attributes during an interaction.

Actions: A user's action a_t showcases her strategy for shifting data focus. Our proposed actions $a_t \in A$ provide fine granularity to analyze these shifts. The action *Keep* is used when a user finds the current attributes useful and wants to investigate further. Alternatively, the user can modify the attribute set to explore new information. Actions *Modify-1*, *Modify-2*, *Modify-3*, and *Modify-4+* are used to modify one, two, three, and more than three attributes in a single interaction, respectively. These actions capture how exploratory the user wants to be in their shifts in data focus, Our analysis of the interaction logs reveals that over 99.5% of the time, users modify fewer than four attributes in a single step. Therefore, we do not need to go beyond *Modify-4+*.

Reward: We want our MDP reward to reflect how useful shifting to a specific data area would be for discovering insights, enabling the RL model to learn what drives shifts in users' data focus. Since we are reusing user studies that lack information in this regard, we quantify the usefulness of a set of attributes so that users will find insights when investigating that data area. This reward r_a, calculated using Equation 3, quantifies the importance of attribute a from the dataset D. We exclude the test user's interactions from reward generation before evaluating the learning algorithms for that specific test user.

$$r_a = \frac{\text{number of users that used } a}{\text{total number of users who explored } D} \tag{3}$$

Consequently, if α is the selected attributes in t, users receive reward:

$$R_t = \sum_{a \in \alpha} r_a \tag{4}$$

6.2.3 Exploration Problem in the Context of MDP: Based on the visualization generated from a user's selected attributes (s_t) at interaction $t \in T$, the user decides how to modify these attributes using an action a_t. The strategy by which the user picks a_t in s_t is called the user's exploration policy π. Performing a_t, the user reaches state s'_t and receives reward r_t. The user updates π based on new insights r_t. The goal is to maximize the chances of discovering attributes with desired insights:

$$\max_{\pi} \mathbb{E}\left[\sum_{interaction=t}^{T} \text{reward}(\pi, interaction)\right] \tag{5}$$

To achieve this, the user may need to adapt her policy to $\pi^* : State \rightarrow Action$.

As a simple example, in task T4, from Table 3, a user might continue exploring `Temperature` and `Fog` (state) in detail or shift focus (action) to `Snow` and `Precipitation` to achieve a more accurate weather prediction. Insights gained (reward) from the explored visualizations influence how the user shifts her data focus (policy update).

6.3 Statistically Analyzing Shifts in Data Focus

Using the actions introduced in subsubsection 6.2.2, we determine the probability of using a particular data focus shifting strategy in each exploration half. We treat the exploration half as a fixed effect, with individual users and datasets as random effects. Table 4 presents the likelihood ratio test results.

On average, each user performed 84 interactions, with a standard deviation of 24.24. Action distribution is: *Keep* 52.57%, *Modify-1* 28.98%, *Modify-2* 10.28%, *Modify-3* 2.12%, and *Modify-4+* 6.02%.

The results reveal a significant shift in users' data focus strategies between the exploration halves. For instance, the *Keep* is notably influenced by the *exploration half.* This is because, users initially explore multiple attributes to get a generic idea about the data before performing a drill-down analysis by repeatedly using *Keep* on selected attributes.

Actions	χ^2	P-value	Significance
Keep	5.1005	0.02392	*
Modify-1	5.3292	0.02097	*
Modify-2	6.748	0.009385	**
Modify-3	6.3406	0.0118	*
Modify-4+	6.152	0.01313	*

Table 4: Significance test results for each action. Significance: *** p < 0.001; ** p < 0.01; * p < 0.05;

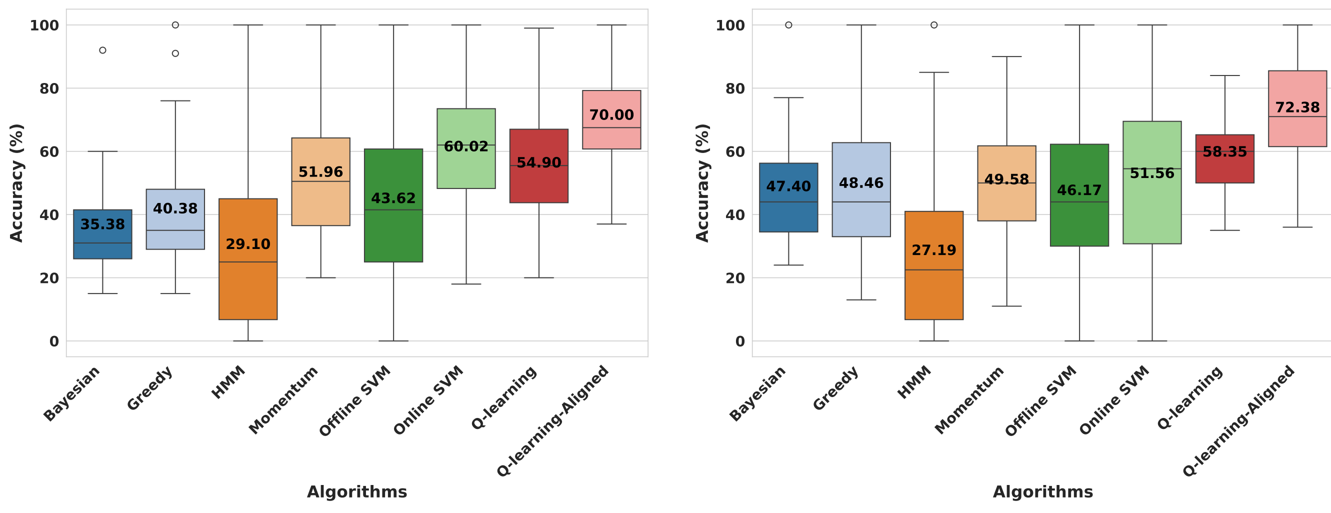
6.4 Performance Evaluation

Following the same evaluation setup, train-test split and summarization procedure described in subsection 5.4. The results of the Tableau experiment are presented in Figure 6.

Results: Similar to subsubsection 5.4.2, in this study, QLearn-Aligned outperforms all other methods. The accuracy of the RL models in the individual experiments are QLearn-Aligned (70%), Q-learning (54.90%), SARSA (53.38%), Actor-Critic (58.79%), and Reinforce (54.65%). For the population experiments, the accuracies are QLearn-Aligned (72.38%), Q-learning (58.35%), SARSA (55.27%), Actor-Critic (55.79%), and Reinforce (55.79%).

QLearn-Aligned explicitly integrates users' attribute interests with reward signals after each interaction and uses hyperparameters to anticipate shifts in their data focus, leading to more accurate predictions. This approach is beneficial because, while predicting a high-reward (insight-yielding) attribute may bring initial success, repeatedly using the same strategy (e.g., using *Keep*) becomes less effective over time, as the user may have already gathered the needed insights and shifted focus to other attributes.

Additionally, we observe that with more data in the population setting, OfflineSVM (+2.55%), Q-learning (+3.45%), SARSA (+1.89%), and QLearn-Aligned (+2.38%) see performance improvements, while



(a) Individual Experiments **(b) Population Experiments**

Figure 6: Performance results for two experimental settings in the Tableau user study.

	Actor-Critic	Q-learning	Q-learning-Aligned	Online SVM	Offline SVM	Greedy
Keep	76.33	87.67	88.00	61.67	50.67	51.67
Modify-1	21.67	3.33	37.67	43.67	31.00	18.33
Modify-2	3.33	16.00	23.00	24.33	21.33	24.67
Modify-3	0.33	0.67	2.00	2.00	0.00	3.33
Modify-4+	1.33	3.33	15.67	13.00	4.67	8.33

Table 5: Individual Setting: Each row shows the accuracy (%) of predicting a specific data focus shifting action (Tableau study)

	Actor-Critic	Q-learning	Q-learning-Aligned	Online SVM	Offline SVM	Greedy
Keep	98.67	93.67	88.67	69.33	54.67	67.67
Modify-1	2.67	13.67	53.00	47.00	30.33	31.67
Modify-2	10.67	21.67	40.00	52.33	48.33	24.33
Modify-3	0.00	9.67	9.67	9.67	11.00	4.67
Modify-4+	0.00	4.00	32.00	12.67	3.67	23.00

Table 6: Population Setting: Each row shows the accuracy (%) of predicting a specific data focus shifting action (Tableau study)

Online SVM (-8.46%), and Actor-Critic (-3.83%) decline. This finding intrigued us, prompting further investigation into what these models are actually predicting.

Each row of Table 6 and Table 5 represents the accuracy of predicting a specific action in individual and population settings, respectively. In the population setting, Actor-Critic and Q-learning overfit to the action *Keep* (Table 6) compared to the individual setting (Table 5). Overall, QLearn-Aligned improves over generic Q-learning on the actions that occur less frequently. Most algorithms struggle to predict *Modify-2*, *Modify-3*, and *Modify-4+*, which appear in less than 15% of interactions. However, leveraging this imbalance and always predicting the most frequent action, *Keep* won't lead to high accuracy. For better performance, models must at least accurately predict smaller shifts like *Modify-1*. Although Momentum did well in the imMens user study, it lacks learning capability and depends on users to shift data focus, making it less suitable for modeling but still informative about the data.

7 Zeng et al. Voyager User Study

The user study by Zeng et al. [70] comprises analysis tasks (e.g., T1–T3 in Table 7) similar to those in the Tableau study subsubsection 6.1.1, and open-ended analysis tasks (e.g., T4 in Table 7) as in subsubsection 5.1.1.

7.1 Overview of Exploration Task

7.1.1 Analysis Task: 72 participants complete four data exploration **tasks** using a visualization recommendation system (Figure 7) [70]. The study includes two **datasets**: (a) *Movies* dataset containing

Task ID	Task	Task Objective
T1	Focused	Which creative type has the max number of movies based on Book/Short Story (Source)?
T2	Focused	Among Disney (Source) movies, what's the running time of the highest-grossing?
T3	Open-Ended	What kinds of movies will be the most successful based on data?
T4	Open-Ended	Explore data for [15 mins]. Use bookmarks to save patterns, trends, or insights.

Table 7: Focused and Open-ended EVA tasks for Movies Dataset

3,101 records and 16 attributes, (b) *Birdstrikes* dataset (Example 1) containing 10,000 records and 14 attributes [67].

7.1.2 Task Characteristics: During the user study participants complete two focused tasks and two open-ended tasks (see Table 7). The **focused** tasks (T1 and T2) provide clear hints on which data area to explore. Whereas, for the **open-ended** tasks (T3 and T4) participants are expected to explore the dataset for interesting insights. Participants received a study overview and a 10-minute demo with a dataset distinct from the ones used in the study. Hence, they have no **prior experience** with the dataset.

7.1.3 Interface and Interaction Log. The system **interface** (see Figure 7) is inspired by Voyager [67, 68]. Similar to Tableau, users can select attributes to explore from panel (B), and the system recommends visualizations based on these selected attributes in panel (C). Additionally, the system suggests visualizations from related data areas in panel (D). Users can analyze these visualizations and bookmark them for later review, accessible through the bookmark button in panel (A).

The **interaction log** contains information on the user's actions, including selecting attributes, bookmarking charts, mouse hover, and scroll-over charts. The logs also contain metadata about the user's chart bookmarks for open-ended tasks. To remove unintentional noise, we exclude mouse movement logs lasting less than half a second, following the original study [70].

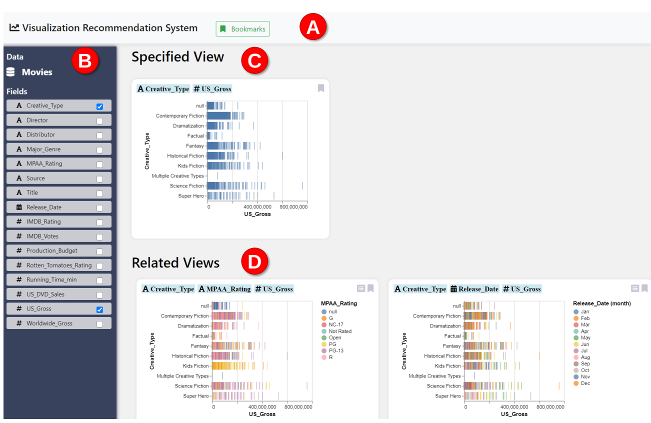


Figure 7: Interface for the Voyager user study.

7.2 Formalizing the Exploration Problem

7.2.1 User Activities. Users' exploration activities in this user study have close similarity to the Tableau study (subsubsection 6.2.1), the exploration is driven by the data attributes users select in each

Actions	χ^2	P-value	Significance
Keep	5.5818	0.01815	*
Modify-1	35.449	7.594×10^{-6}	***
Modify-2	2.9127	0.08789	.
Modify-3	0.8095	0.3683	-

Table 8: Significance test results for each action. Significance: *** p <0.001; ** p <0.01; * p <0.05; . p <0.10; - p >0.10

interaction. The Voyager system generates visualizations relevant to the selected attributes. For the next exploration steps, (1) the user may analyze the specified chart, (2) shift to exploring other attributes in the system-generated related views, or (3) modify her current attribute selections in the data panel.

The visualizations contain rewards that may come in the form of relevant answers to the task questions for focused tasks (T1, T2) or insights/data characteristics that the users may bookmark for open-ended tasks (T3, T4). Rewards help the user decide how to shift her data focus.

7.2.2 Modeling Exploration Using MDP. To maintain consistency, we align the MDP formulation to the Tableau user study (subsubsection 6.2.2). While the state representation (s_t) remains the same, there are minor modifications to the action space (A) due to the Voyager interface.

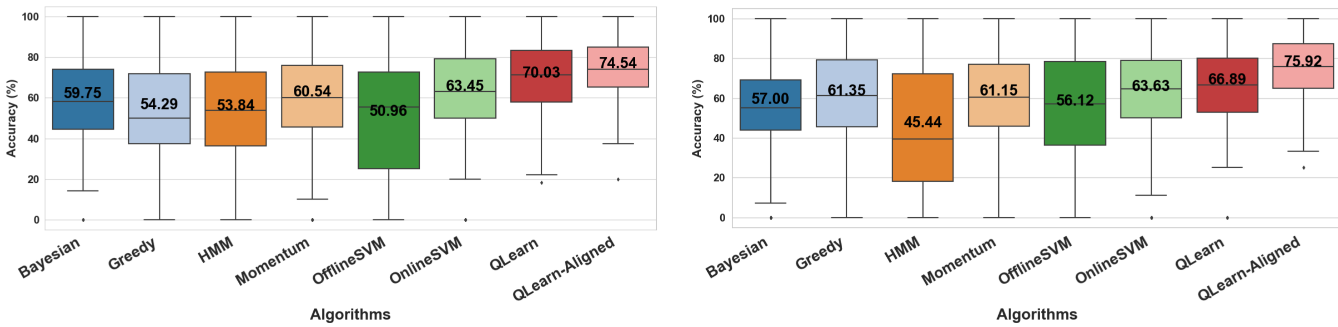
Actions: In contrast to the Tableau interface, the visualization recommendation system used in [70] limits users to select a maximum of three attributes in a single interaction. Therefore, the maximum degree of shift is limited to changing three attributes in a single interaction; thus, we remove the *Modify-4+* action from the Tableau user study action space (subsubsection 6.2.2). The final set of actions are: *Keep*, *Modify-1*, *Modify-2*, and *Modify-3*.

Reward: We utilize the bookmark metadata during open-ended tasks to enhance the reward function defined in Equation 4 and follow the same approach for the reward calculation. Specifically, for each user, we augment R_t by adding a positive scalar value n_a, corresponding to the number of times an attribute (a) appears in the user's bookmarked charts:

$$R_t = \sum_{a \in \alpha} r_a + n_a \quad (6)$$

7.3 Statistically Analyzing Shifts in Data Focus

For this study, the *full model* includes the *exploration phase* and *open-endedness* as fixed effects and the *user* and *dataset* as random effects. Here, we include open-endedness as a fixed effect as there are enough open-ended interactions compared to the Tableau study. As a sanity check, we also tested open-endedness as a random effect, yielding the same significance test results.



(a) Individual Experiments **(b) Population Experiments**

Figure 8: Performance results for two experimental settings in the Voyager user study.

The number of interactions across all exploration tasks has a mean of 106.85 and a standard deviation of 80.84. Action distribution is: 59%, 26%, 14%, 1% for *Keep, Modify-1, Modify-2* and *Modify-3* respectively.

Table 8 shows the likelihood test results, revealing significant effects of the exploration halves on actions *Keep*, *Modify-1*, and *Modify-2*. Overall, the findings align with those from the Tableau user study (subsection 6.3): users tend to favor drill-down analysis of the same attributes (*Keep*) or gradually modify them (*Modify-1, 2*), with these behaviors significantly changing across exploration phases. In contrast, drastic shifts using *Modify-3* account for only 1% of all interactions and are not influenced by the *exploration half*.

7.4 Performance Evaluation

We follow the same evaluation setup, train-test split and summarization procedure described in subsection 5.4. The results of the Voyager study experiments are presented in Figure 8.

Results: Similar to subsections 5.4 and 6.4, RL algorithms benefit from explicitly integrating users' *interesting insights* and *relevant attributes* with the reward signal, thereby outperforming existing methods. For instance, user 31 (Figure 9) initially explores `Title`, `IMDB Rating`, and `Gross Profit`. She then shifts one attribute to `Rotten Tomatoes Rating` before returning to `IMDB Rating` for a detailed drill-down analysis (using *Keep*) during the majority of subsequent interactions. However, HMM and Greedy get stuck in a sub-optimum and associate the shift of one attribute (*Modify-1*) as the best action for future interactions on `Title`, `IMDB Rating` and `Gross Profit`.

QLearn-Aligned accurately predicts rare shifts in data focus, whereas Q-learning tends to remain stuck on previous attributes. For instance, in Figure 9, QLearn-Aligned successfully predicts *Modify-3*. This is further supported by Table 9, which shows QLearn-Aligned outperforming Q-learning in predicting the action *Modify-3*.

Open-endedness: Learning algorithms perform better (+8.48%) in focused tasks compared to open-ended tasks, as the latter entail a larger search space for both models and users [10]. The accuracy of the top-performing QLearn-Aligned model drops by 9.44% in open-ended tasks across both experiment settings. This indicates that while RL models consistently outperform existing models in predicting shifts in data focus, they are not fully robust to the increased exploration noise in open-ended scenarios.

Experiment Setting: Consistent with the Tableau study results, Greedy (+7%) and OfflineSVM (+5.2%) perform significantly better in population experiments, while HMM underperforms (−8.2%). Following subsection 6.4, we investigate models' per-action predictions and find similar results to Tables 6 and 5. Due to space constraints, we only present the population setting results in Table 9. We find that models trained on population data more accurately predict actions that users most frequently perform on average. For example, OfflineSVM's prediction accuracies for actions *Keep*, *Modify-1*, *Modify-2*, and *Modify-3* are 64%, 28%, 17%, and 1.3%, when trained on individual users. However, OfflineSVM trained on the population data identifies *Keep* as the dominant action, increasing its accuracy by 8%.

In contrast, HMM drastically underperforms in the population experiment. For instance, HMM achieves 66% accuracy in focused tasks, whereas its accuracy drops to 28% in open-ended tasks, where users' exploration patterns differ significantly. This highlights the risk of models misinterpreting noise as relevant patterns and failing to adapt to new strategies for shifting data focus.

8 Discussion

This paper presents an in-depth analysis of three prominent EVA studies [10, 36, 70] and compares the performance of popular online learning algorithms with existing EVA modeling techniques. Our initial statistical analyses confirm that analysts' data focus shifts from the first to the second half of their exploration sessions. These results suggest that algorithms that take data focus shifts into account will be well suited to EVA scenarios, which is reinforced by the competitive performance of the RL algorithms over existing methods. In this section, we summarize our key findings, discuss their implications, and suggest new avenues for future research.

8.1 Reinforcement Learning (RL) for user modeling in EVA

A clear takeaway from our analyses is that **RL algorithms, specifically QLearn-Aligned consistently outperform existing techniques for modeling users' shifts in data focus during EVA.** We summarize the accuracy of all tested algorithms in Tables 10

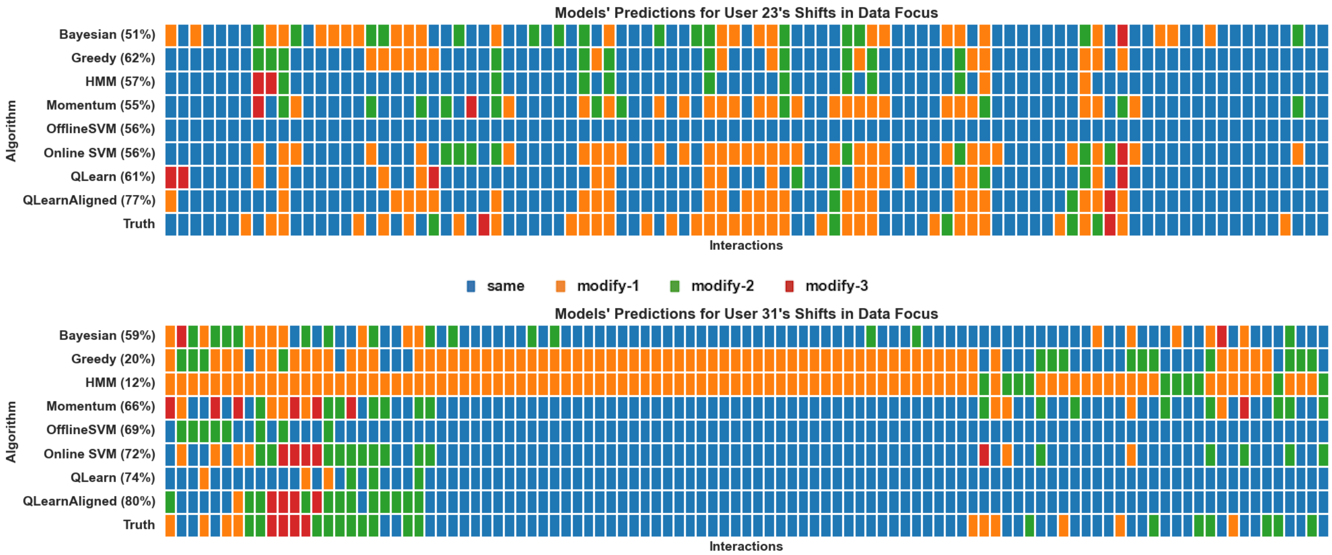


Figure 9: Comparison between model's predictions and ground truth for two users' shifts in data focus. The length of predictions and ground truth is the same for all models.

	Actor-Critic	Q-learning	Q-learning-Aligned	Online SVM	Offline SVM	Greedy
Keep	95.42	92.15	92.21	68.62	71.85	80.32
Modify-1	6.04	13.04	35.79	40.81	24.44	32.83
Modify-2	0.32	8.63	28.44	40.99	14.19	11.11
Modify-3	0.00	6.97	20.21	8.57	0.79	4.47

Table 9: Population setting: Each row shows the accuracy (%) of predicting a specific data focus shifting action (Voyager study).

Model Class	Model	imMens	Tableau	Voyager
Reinforcement Learning (RL)	QLearn-Aligned	**84.94%**	**70.00%**	**74.45%**
	QLearn	70.81%	54.90%	70.07%
	SARSA	67.25%	53.38%	69.73%
	Actor-Critic	67.88%	58.79%	65.91%
	Reinforce	64.06%	54.65%	67.36%
Simple Decision Making Heuristics	Greedy	**63.75%**	**40.38%**	**54.29%**
	WSLS	22.81%	24.35%	25.76%
	Random	25.38%	19.92%	25.26%
Current Baselines in Visualization	Momentum	85.56%	51.96%	60.54%
	Bayesian	59.81%	35.38%	59.75%
	HMM	64.44%	29.10%	53.84%
	OnlineSVM*	**86.81%**	**60.02%**	**63.45%**
	OfflineSVM	57.62%	43.62%	50.96%

Table 10: A summary of models' performance for Individual Experiment Settings. *OnlineSVM is a new adaptation to the current SVM baseline in visualization.

Model Class	Model	imMens	Tableau	Voyager
Reinforcement Learning (RL)	QLearn-Aligned	**87.81%**	**72.38%**	**75.92%**
	QLearn	80.00%	58.35%	66.89%
	SARSA	78.38%	55.27%	66.65%
	Actor-Critic	74.94%	54.96%	66.57%
	Reinforce	71.88%	55.79%	63.25%
Simple Decision Making Heuristics	Greedy	**76.12%**	**48.46%**	**61.34%**
	WSLS	23.19%	23.81%	25.96%
	Random	27.56%	25.08%	24.23%
Current Baselines in Visualization	Momentum	**86.94%**	49.58%	61.07%
	Bayesian	66.38%	47.40%	57.00%
	HMM	65.12%	27.19%	45.44%
	OnlineSVM*	81.25%	**51.56%**	**63.63%**
	OfflineSVM	60.19%	46.17%	56.12%

Table 11: A summary of models' performance for Population Experiment Settings.

and 11. RL algorithms mimic users' decision-making strategies for analyzing the next data area based on the current one's usefulness, which we believe helps RL outperform others.

Q-learning models users more effectively than other RL algorithms, such as Actor-Critic and Reinforce (section 4). We believe the effectiveness stems from Q-learning's policy updates, where the chosen action is compared to the best-estimated action in the next state [$\arg\max_a Q(s_{t+1}, a)$]. This resembles how humans evaluate actions by considering the potential benefits of the best option identified so far. Additionally, more complex algorithms typically require larger datasets for effective training. Therefore, QLearn-Aligned, with its relative simplicity, is a stronger candidate for modeling users' dynamic data focus within VESs, where a model may need to learn from limited streaming interaction data and adapt to the user's shifts in data focus in real time.

Overall our results suggest that **popular algorithms within the visualization community, specifically Bayesian and HMM, have limited functionality to capture the nuances of users' dynamic shifts in data focus.** RL algorithms leverage hyperparameters (section 4) to dynamically adjust to the user's exploration rate and integrate users' data interests online. In contrast, HMMs are more popular in unsupervised scenarios with abundant relevant unlabeled data and do not perform as well in noisy and constrained data environments.

8.2 Influence of Exploration Task Characteristics:

EVA task complexity and open-endedness increase difficulty in the modeling process, directly impacting algorithm performance. In the imMens study, users have a limited decision space (subsubsection 5.1.1), exploring fixed visualizations, making it easier to model users' data focus shifts. While in Tableau and Voyager studies, users have a broader decision space with multiple attributes, adjusting their data focus strategies as they explore. They may choose suboptimal strategies, like exploring unknown areas that may not contain any insight. However, it helps in understanding the dataset and maximizing long-term rewards. This benefits RL, which excels in learning decision-making policies in complex scenarios.

Task open-endedness further complicates the modeling process, as supported by our empirical analysis in the Voyager user study, where models achieved an 8.48% higher average accuracy in focused tasks than open-ended tasks (subsection 7.4). These findings allude to a broader hypothesis that current learning models can be further improved to capture users' exploration behavior in open-ended tasks.

We also hypothesize that users with prior experience may require less effort to complete tasks of similar complexity. We discovered it when analyzing the Tableau interaction logs (subsubsection 6.2.1), where the open-ended task (T4) had some commonalities with focused tasks (T1-T3). Therefore, users often reused previously explored attributes to complete T4 instead of exploring new ones. Therefore, if we want to observe how users shift their strategies, we must place them in environments that challenge them to shift.

8.3 Takeaways for Future Experimenters/Model Designers

Considerations for experiment design: Given that the performance of many models are not significantly different across individual (Table 10) and population settings (Table 11), we take care in reporting our observations and hesitate to make broad statements. However, we note some interesting trends from our empirical evaluation.

There is a clear difference between offline and online models when comparing the individual and population experiment settings. For instance, OfflineSVM performs better in the population setting. This increased data allows the model to learn from broader data-focus shifting strategies across users (discussed in subsection 7.4). In contrast, OnlineSVM shows poorer performance in population experiments, likely because the real-time feedback it depends on is drowned out by patterns the model has already internalized. This issue is exacerbated by noisy data-shifting strategies in open-ended tasks, as encountered with HMM in the Voyager study (subsection 7.4). RL model, specifically QLearn-Aligned, performs slightly better in the population setting while already showing strong results in individual experiments. This reliability makes QLearn-Aligned a robust baseline for future systems.

We also find that there may exist an imbalance in actions that users perform (Sections 5.3, 6.3, 7.3). We observe that most algorithms struggle to make accurate predictions on rare actions (Tables 5, 6, 9). Therefore, it is essential to validate the model design by testing on faithful data distributions.

Comparing against all relevant baselines: Testing current approaches for user modeling in visualization, including Momentum, Bayesian, HMM, SVMs, and three additional simple decision-making heuristics, gave us valuable information about how RL algorithms perform. Even though Momentum and Greedy are simple, they outperform other algorithms in a few scenarios (Table 11), when users repeat past actions regardless of their outcomes before reaching a decision. This finding underscores the importance of having relevant baselines to test our modeling assumptions. While these simple heuristics are common in RL and machine learning literature, we observe that *similar natural baselines are often overlooked in the evaluation of sophisticated user models* in visualization [32, 42, 46]. This methodology is even more important for visualization recommendation scenarios, where we know there are already many algorithms available to compare against [70].

References

[1] [n. d.]. Microsoft PowerBI. https://powerbi.microsoft.com.

[2] [n. d.]. Tableau Software. http://www.tableausoftware.com.

[3] Jonathan Back and Charles Oppenheim. 2001. A model of cognitive load for IR: implications for user relevance feedback interaction. *Information Research* 6, 2 (2001), 6–2.

[4] Calvin S Bao, Siyao Li, Sarah G Flores, Michael Correll, and Leilani Battle. 2022. Recommendations for visualization recommendations: Exploring preferences and priorities in public health. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–17.

[5] Ori Bar El, Tova Milo, and Amit Somech. 2020. Automatically generating data exploration sessions using deep reinforcement learning. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. 1527–1537.

[6] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2015. Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software* 67, 1 (Oct. 2015), 1–48. https://doi.org/10.18637/jss.v067.i01

[7] Leilani Battle, Remco Chang, and Michael Stonebraker. 2016. Dynamic prefetching of data tiles for interactive visualization. In *Proceedings of the 2016 International Conference on Management of Data*. ACM, 1363–1375.

[8] Leilani Battle, R Jordan Crouser, Audace Nakeshimana, Ananda Montoly, Remco Chang, and Michael Stonebraker. 2019. The role of latency and task complexity in predicting visual search behavior. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 1246–1255.

[9] Leilani Battle, Philipp Eichmann, Marco Angelini, Tiziana Catarci, Giuseppe Santucci, Yukun Zheng, Carsten Binnig, Jean-Daniel Fekete, and Dominik Moritz. 2020. Database Benchmarking for Supporting Real-Time Interactive Querying of Large Data. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data* (Portland, OR, USA) *(SIGMOD '20)*. Association for Computing Machinery, New York, NY, USA, 1571–1587. https://doi.org/10.1145/

3318464.3389732

[10] Leilani Battle and Jeffrey Heer. 2019. Characterizing exploratory visual analysis: A literature review and evaluation of analytic provenance in tableau. In *Computer graphics forum*, Vol. 38. Wiley Online Library, 145–159.

[11] Leilani Battle and Alvitta Ottley. 2022. A programmatic definition of visualization insights, objectives, and tasks. *arXiv preprint arXiv:2206.04767* (2022).

[12] Leilani Battle and Carlos Scheidegger. 2020. A Structured Review of Data Management Technology for Interactive Visualization and Analysis. *IEEE Transactions on Visualization and Computer Graphics* (2020), 1–1. https://doi.org/10.1109/TVCG.2020.3028891

[13] D. Bennett, G. Davidson, and Y. Niv. 2022. A model of mood as integrated advantage. *Psychological Review* 129, 3 (April 2022), 513–541. https://doi.org/10.1037/rev0000294 Epub 2021 Sep 13.

[14] Daniel Bennett, Yael Niv, and Angela J. Langdon. 2021. Value-free reinforcement learning: policy optimization as a minimal model of operant behavior. *Current opinion in behavioral sciences* 41 (2021).

[15] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.".

[16] Eli T Brown, Alvitta Ottley, Helen Zhao, Quan Lin, Richard Souvenir, Alex Endert, and Remco Chang. 2014. Finding waldo: Learning about users from their interactions. *IEEE Transactions on visualization and computer graphics* 20, 12 (2014), 1663–1672.

[17] Robert R Bush and Frederick Mosteller. 1953. A stochastic model with applications to learning. *The Annals of Mathematical Statistics* (1953), 559–585.

[18] Yuxiang Cen, Lu Gan, and Chuanhe Bai. 2013. Reinforcement learning in information searching. *Information Research* 18, 1 (2013), paper 569. http://InformationR.net/ir/18-1/paper569.html

[19] Kun-Ting Chen, Quynh Quang Ngo, Kuno Kurzhals, Kim Marriott, Tim Dwyer, Michael Sedlmair, and Daniel Weiskopf. 2023. Reading strategies for graph visualizations that wrap around in torus topology. In *Proceedings of the 2023 Symposium on Eye Tracking Research and Applications*. 1–7.

[20] Nathaniel D. Daw, Samuel J. Gershman, Ben Seymour, Peter Dayan, and Raymond J. Dolan. 2011. Model-Based Influences on Humans' Choices and Striatal Prediction Errors. *Neuron* 69, 6 (March 2011), 1204–1215.

[21] Dazhen Deng, Aoyu Wu, Huamin Qu, and Yingcai Wu. 2022. DashBot: Insight-Driven Dashboard Generation Based on Deep Reinforcement Learning. *IEEE Transactions on Visualization and Computer Graphics* (2022), 1–11. https://doi.org/10.1109/tvcg.2022.3209468

[22] Punit Doshi, Elke Rundensteiner, and Matthew Ward. 2003. Prefetching for Visual Data Exploratio. 195– 202. https://doi.org/10.1109/DASFAA.2003.1192383

[23] Will Epperson, Doris Jung-Lin Lee, Leijie Wang, Kunal Agarwal, Aditya G Parameswaran, Dominik Moritz, and Adam Perer. 2022. Leveraging analysis history for improved in situ visualization recommendation. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 145–155.

[24] Mi Feng, Evan Peck, and Lane Harrison. 2018. Patterns and pace: Quantifying diverse exploration behavior with visualizations on the web. *IEEE transactions on visualization and computer graphics* 25, 1 (2018), 501–511.

[25] Sneha Gathani, Shayan Monadjemi, Alvitta Ottley, and Leilani Battle. 2022. A Grammar-Based Approach for Applying Visualization Taxonomies to Interaction Logs. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 489–500.

[26] Paul W. Glimcher. 2011. Understanding dopamine and reinforcement learning: The dopamine reward prediction error hypothesis. *Proceedings of the National Academy of Sciences* 108, supplement_3 (2011), 15647–15654. https://doi.org/10.1073/pnas.1014269108

[27] David Gotz and Zhen Wen. 2009. Behavior-driven visualization recommendation. In *Intelligent User Interfaces*.

[28] David Gotz and Michelle X Zhou. 2009. Characterizing users' visual analytic activity for insight provenance. *Information Visualization* 8, 1 (2009), 42–55.

[29] Tera Marie Green, William Ribarsky, and Brian Fisher. 2009. Building and Applying a Human Cognition Model for Visual Analytics. *Information Visualization* 8, 1 (2009), 1–13. https://doi.org/10.1057/ivs.2008.28

[30] Hua Guo, Steven R. Gomez, Caroline Ziemkiewicz, and David H. Laidlaw. 2016. A Case Study Using Visualization Interaction Logs and Insight Metrics to Understand How Analysts Arrive at Insights. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (2016), 51–60. https://doi.org/10.1109/TVCG.2015.2467613

[31] Jacek Gwizdka. 2010. Distribution of cognitive load in web search. *Journal of the American Society for Information Science and Technology* 61, 11 (2010), 2167–2187.

[32] Sunwoo Ha, Shayan Monadjemi, Roman Garnett, and Alvitta Ottley. 2022. A unified comparison of user modeling techniques for predicting data interaction and detecting exploration bias. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (2022), 483–492.

[33] Chen He, Luana Micallef, Liye He, Gopal Peddinti, Tero Aittokallio, and Giulio Jacucci. 2021. Characterizing the Quality of Insight by Interactions: A Case Study. *IEEE Transactions on Visualization and Computer Graphics* 27, 8 (2021), 3410–3424. https://doi.org/10.1109/TVCG.2020.2977634

[34] Vijay R Konda and John N Tsitsiklis. 2003. Actor-critic algorithms. *SIAM journal on control and optimization* 42, 4 (2003), 1143–1166.

[35] Heidi Lam. 2008. A framework of interaction costs in information visualization. *IEEE transactions on visualization and computer graphics* 14, 6 (2008), 1149–1156.

[36] Zhicheng Liu and Jeffrey Heer. 2014. The effects of interactive latency on exploratory visual analysis. *IEEE transactions on visualization and computer graphics* 20, 12 (2014), 2122–2131.

[37] Zhicheng Liu, Biye Jiang, and Jeffrey Heer. 2013. imMens: Real-time visual querying of big data. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library, 421–430.

[38] Zhicheng Liu and John Stasko. 2010. Mental Models, Visual Reasoning and Interaction in Information Visualization: A Top-down Perspective. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 999–1008. https://doi.org/10.1109/TVCG.2010.177

[39] Jiyun Luo, Sicong Zhang, and Hui Yang. 2014. Win-win search: Dual-agent stochastic game in session search. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 587–596.

[40] Ben McCamish, Vahid Ghadakchi, Arash Termehchy, Behrouz Touri, and Liang Huang. 2018. The data interaction game. In *Proceedings of the 2018 International Conference on Management of Data*. 83–98.

[41] Tova Milo and Amit Somech. 2018. Next-step suggestions for modern interactive data analysis platforms. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 576–585.

[42] Shayan Monadjemi, Roman Garnett, and Alvitta Ottley. 2020. Competing models: Inferring exploration patterns and information relevance via Bayesian model selection. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 412–421.

[43] Rudolf Netzel, Bettina Ohlhausen, Kuno Kurzhals, Robin Woods, Michael Burch, and Daniel Weiskopf. 2017. User performance and reading strategies for metro maps: An eye tracking study. *Spatial Cognition & Computation* 17, 1-2 (2017), 39–64.

[44] Yael Niv. 2009. Reinforcement learning in the brain. *Journal of Mathematical Psychology* 53, 3 (2009), 139–154.

[45] Yael Niv, Jeffrey A. Edlund, Peter Dayan, and John P. O'Doherty. 2012. Neural Prediction Errors Reveal a Risk-Sensitive Reinforcement-Learning Process in the Human Brain. 32, 2 (2012), 551–562. https://doi.org/10.1523/JNEUROSCI.5498-10.2012

[46] Alvitta Ottley, Roman Garnett, and Ran Wan. 2019. Follow the clicks: Learning and anticipating mouse interactions during exploratory data analysis. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 41–52.

[47] Antti Oulasvirta, Jussi PP Jokinen, and Andrew Howes. 2022. Computational rationality as a theory of interaction. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–14.

[48] Robert E Patterson, Leslie M Blaha, Georges G Grinstein, Kristen K Liggett, David E Kaveney, Kathleen C Sheldon, Paul R Havig, and Jason A Moore. 2014. A human cognition framework for information visualization. *Computers & Graphics* 42 (2014), 42–58.

[49] Adam Perer and Ben Shneiderman. 2008. Systematic yet flexible discovery: guiding domain experts through exploratory data analysis. In *Proceedings of the 13th international conference on Intelligent user interfaces*. 109–118.

[50] Xin Qian, Ryan A. Rossi, Fan Du, Sungchul Kim, Eunyee Koh, Sana Malik, Tak Yeon Lee, and Nesreen K. Ahmed. 2022. Personalized Visualization Recommendation. *ACM Trans. Web* 16, 3, Article 11 (sep 2022), 47 pages. https://doi.org/10.1145/3538703

[51] ENCODING RECOGNITION RECALL. 2016. Beyond Memorability: Visualization Recognition and Recall. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS* 22, 1 (2016).

[52] Khairi Reda, Andrew E Johnson, Michael E Papka, and Jason Leigh. 2016. Modeling and evaluating user behavior in exploratory visual analysis. *Information Visualization* 15, 4 (2016), 325–339.

[53] Gavin A Rummery and Mahesan Niranjan. 1994. On-line q-learning using connectionist systems. In *Proceedings of the 1994 connectionist models summer school*. Citeseer, 1–11.

[54] Sanad Saha, Nischal Aryal, Leilani Battle, and Arash Termehchy. 2024. ShiftScope: Adapting Visualization Recommendations to Users' Dynamic Data Focus *(SIGMOD/PODS '24)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3626246.3654753

[55] Sanad Saha, Nischal Aryal, Leilani Battle, and Arash Termehchy. 2024. User Learning In Interactive Data Exploration. , 5660-5661 pages. https://doi.org/10.1109/ICDE60146.2024.00460

[56] Mariia Seleznova, Behrooz Omidvar-Tehrani, Sihem Amer-Yahia, and Eric Simon. 2020. Guided exploration of user groups. *Proceedings of the VLDB Endowment (PVLDB)* 13, 9 (2020), 1469–1482.

[57] Danqing Shi, Yang Shi, Xinyue Xu, Nan Chen, Siwei Fu, Hongjin Wu, and Nan Cao. 2019. Task-oriented optimal sequencing of visualization charts. In *2019 IEEE Visualization in Data Science (VDS)*. IEEE, 58–66.

[58] Yedendra B Shrinivasan, David Gotz, and Jie Lu. 2009. Connecting the dots in visual analysis. In *2009 IEEE symposium on visual analytics science and technology*.

IEEE, 123–130.

[59] Richard S Sutton and Andrew G Barto. 1998. *Reinforcement learning: An introduction.* MIT press.

[60] Kohei Tamura and Naoki Masuda. 2015. Win-stay lose-shift strategy in formation changes in football. *EPJ Data Science* 4 (2015), 1–19.

[61] Maria Taramigkou, Dimitris Apostolou, and Gregoris Mentzas. 2018. Leveraging exploratory search with personality traits and interactional context. *Information Processing & Management* 54, 4 (2018), 609–629.

[62] Kashyap Todi, Gilles Bailly, Luis Leiva, and Antti Oulasvirta. 2021. Adapting user interfaces with model-based reinforcement learning. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–13.

[63] Vladimir Vapnik, Steven Golowich, and Alex Smola. 1996. Support vector method for function approximation, regression estimation and signal processing. *Advances in neural information processing systems* 9 (1996).

[64] Christopher JCH Watkins and Peter Dayan. 1992. Q-learning. *Machine learning* 8, 3-4 (1992), 279–292.

[65] Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* 8, 3-4 (1992), 229–256.

[66] Bodo Winter. 2013. Linear models and linear mixed effects models in R with linguistic applications. *arXiv preprint arXiv:1308.5499* (2013).

[67] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2015. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 649–658.

[68] Kanit Wongsuphasawat, Zening Qu, Dominik Moritz, Riley Chang, Felix Ouk, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2017. Voyager 2: Augmenting Visual Analysis with Partial View Specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 2648–2659. https://doi.org/10.1145/3025453.3025768

[69] Kai Xu, Alvitta Ottley, Conny Walchshofer, Marc Streit, Remco Chang, and John Wenskovitch. 2020. Survey on the analysis of user interactions and visualization provenance. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 757–783.

[70] Zehua Zeng, Phoebe Moh, Fan Du, Jane Hoffswell, Tak Yeon Lee, Sana Malik, Eunyee Koh, and Leilani Battle. 2021. An evaluation-focused framework for visualization recommendation algorithms. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2021), 346–356.

[71] Emanuel Zgraggen, Zheguang Zhao, Robert Zeleznik, and Tim Kraska. 2018. Investigating the effect of the multiple comparisons problem in visual analysis. In *Proceedings of the 2018 chi conference on human factors in computing systems*. 1–12.

[72] Zhilan Zhou, Ximing Wen, Yue Wang, and David Gotz. 2021. Modeling and leveraging analytic focus during exploratory visual analysis. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.