Online Learning of Visualization Preferences through Dueling Bandits for Enhancing Visualization Recommendations

J.-F. Kassel[1,2] and M. Rohs[1]

[1]Human-Computer Interaction Group, Leibniz University Hannover (LUH), Germany
[2] Volkswagen Group, Germany

Abstract

A visualization recommender supports the user through automatic visualization generation. While previous contributions primarily concentrated on integrating visualization design knowledge either explicitly or implicitly, they mostly do not consider the user's individual preferences. In order to close this gap we explore online learning of visualization preferences through dueling bandits. Additionally, we consider this challenge from a usability perspective. Through a user study ($N = 15$), we empirically evaluate not only the bandit's performance in terms of both effectively learning preferences and properly predicting visualizations (satisfaction regarding the last prediction: $\mu = 85\%$), but also the participants' effort with respect to the learning procedure (e.g., NASA-TLX = 24.26). While our findings affirm the applicability of dueling bandits, they further provide insights on both the needed training time in order to achieve a usability-aligned procedure and the generalizability of the learned preferences. Finally, we point out a potential integration into a recommender system.

CCS Concepts

• ***Human-centered computing*** → ***Visualization systems and tools;*** • ***Computing methodologies*** → *Reinforcement learning;*

1. Introduction

The objectives of visualization recommenders not only include the acceleration of the visual exploration process, but also the lowering of barriers for novice users. In order to achieve these objectives, visualization recommenders automatically generate visualizations based on either explicit knowledge of visualization designs given by effectiveness studies [LQT*18, MHS07, WQM*17, MWN*19] or implicit knowledge through end-to-end machine learning [DD18, HBL*18]. Especially, effectiveness studies provide substantial empirical knowledge about the recommended use of visualization designs in general (e.g., [Mac86]), or task-dependent (e.g., [HYFC14, KH16, KH18]).

The effectiveness of visualization designs is further influenced by the user's characteristics [ZOC*12]. Green and Fisher [GF10] show an effect of the user's personality on the task completion time in visual analytics. Furthermore, the user's preferences on the visual mappings also has an effect on the performance in high and low-level tasks [CCH*14]. Since preferences are generally challenging to quantify [BHM18], Moritz et al. [MWN*19] suggest to use pairwise comparisons in the visualization domain.

A recent promising methodology for learning preferences through pairwise comparisons is the dueling bandit approach [YBKJ12], e.g., in human-robot interaction [SK17], or massive open online courses [CZK16]. A dueling bandit is a special case of the multi-armed bandit which aims to maximize a numerical criterion within a sequential decision process by choosing from a set of actions (or items) at each time step [BHM18]. The bandit predominantly selects the item which is likely to be preferred by the user. Additionally, it needs to explore potentially better items, although the user's preferences are uncertain in the moment of the decision [BHM18]. In our case, the dueling bandit shows two visualizations to the user each in accordance with its exploration strategy. Based on the user's decisions for preferred visualizations, the bandit learns online the user's individual preferences.

As previous approaches use either offline learning [MWN*19] or need previously collected data [MVT16], we attack this problem from a new angle by using the reinforcement learning method of dueling bandits. In order to properly learn the user's preferences but concurrently keep the user's effort low, we propose an approach (see Figure 1) to learn preferences within each visualization feature, e.g., whether the visual mapping `(x:C,y:Q1,color:Q2)` is preferred to `(x:Q1,y:Q2,color:C)`. Through a user study ($N = 15$), we show the promising performance both over time (see Figure 2) and in comparison to a rule-based approach (in 91% of all cases, the visualizations given by the bandit were preferred). Additionally, the results indicate a manageable effort for the participants to train the bandit in terms of time and workload (NASA Task Load Index [HS88] of 24.26). Finally, the insights address the dueling bandit's effectiveness for learning visualization preferences.

DOI: 10.2312/evs.20191175

2. Related Work

First insights on automatic visualization generation are given by Mackinlay [Mac86] by ranking visual variables based on their effectiveness. *ShowMe* [MHS07] integrated this into Tableau Desktop later. As the intention of visualization recommenders is to accelerate visual analysis, some approaches focus on generating visualizations for automatically discovered insights in the data [VRM*15, DHPP17, SKL*16]. *Voyager* [WQM*17] additionally shows alternative visualizations along with the recommended ones. *VizDeck* [KHPA12] recommends visualizations based on both data characteristics and chart type. It further organizes them in a card deck, which the user can flick through.

In addition, machine learning models have been investigated for visualization recommendations. *Data2Vis* [DD18] generates a set of Vega-lite specifications [SMWH17] from a given data set by using an encoder-decoder neural network combined with long short-term memory units. *VizML* [HBL*18] applies end-to-end machine learning through deep neural networks. It learns design choices based on a corpus of both existing visualizations (Plotly gallery) and the associated data sets. Both *DeepEye* [LQT*18] and *VizRec* [MVT16] use a rule-based approach for generating visualizations as a first step. *DeepEye* then uses a decision tree for classifying "good" visualizations in order to rank them afterwards. *VizRec* instead uses both collaborative filtering based on crowdsourced visualization ratings and content-based filtering based on crowdsourced tags associated with the visualizations. *Draco* [MWN*19] formalizes visualization design knowledge by combining both hard and soft constraints. The soft constraints are represented through weights of a RankSVM trained on the data of two effectiveness studies.

Essentially, the discussed systems, except for *VizRec* [MVT16], produce recommendations without taking into account the user's preferences. Compared to *VizRec*, we use qualitative feedback through pairwise comparisons for learning visualization preferences instead of crowdsourced quantitative ratings. Furthermore, collaborative filtering needs a large set of user histories in order to work properly. Through pairwise comparisons, *Draco* [MWN*19] essentially learns the preferences of the populations of its used effectiveness studies. Hence, it learns overall preferences, but not the preferences of the user of the system. Furthermore, *Draco* takes an offline learning approach, while we focus on online learning. In comparison to the related work, our approach can individually adapt its visualization recommendations to the user without requiring any additional resources, like ratings of other users.

3. Dueling Bandit for Visualization Preferences

A dueling bandit with a set of k different items tries to approximate the user's preference distribution $P^{k\times k} = [p_{ij}] = p(i \succ j)$, where $p(i \succ j)$ indicates the probability that the user prefers item i to j [BHM18]. As a broad variety of different dueling bandit algorithms exists [BHM18], we pick an algorithm that fits our requirements.

The first requirement is the absence of a total order of the visualizations, since we assume that some users likely have multiple equally preferred visualizations. In the dueling bandit terminology, we primarily focus on algorithms with Copeland winner strategy [ZKWdR15] (a set of items can be equally preferred) instead of Condorcet winners (only one item is preferred). The second requirement relates to efficiency in terms of the number of needed comparisons due to usability. As a user eventually has to tell the system her or his preferences, we need an algorithm with a small number of needed comparisons, but which smartly explores the set of potential items in order to avoid local optima.

A suitable algorithm appears the Double Thompson Sampling (D-TS) by Wu and Liu [WL16]. The D-TS is currently one of the most comparison-efficient algorithms with an intelligible model. It tracks the number of wins $B^{k\times k} = [b_{ij}] = \#(i \succ j)$, where $\#(i \succ j)$ indicates the number of duels in which i was preferred to j. However, the basis for choosing two items to compare are samples from the posterior Beta distributions estimated on previous comparisons:

$$\begin{aligned}&\theta_{ij} \sim Beta(b_{ij}+1, b_{ji}+1), \text{ for } i<j\\&\theta_{ji} = 1-\theta_{ij}\end{aligned}$$

At each round, D-TS selects a candidate a by using the Relative Upper Confidence Bound [ZKWdR15] while considering the sampled θs. In order to estimate the opponent of a, D-TS again samples from the Beta distributions but limited to the columns related to a. After the user decides for a preferred item, B is updated. The more comparisons are made for a specific pair, the more stable the expected value becomes for this pair.

3.1. Learning Detailed Visualization Preferences

D-TS already has an efficient exploration strategy for learning the preferences within a set of k items, yet, the number of visualizations (k) quickly increases when more design options (e.g., more coloring schemes) are given. Thus, more comparisons are needed. In order to counterbalance this increasing number of needed comparisons, we learn the preferences for each visualization feature (e.g., visual mapping, coloring schema, etc.) separately, instead of learning the preferences between fully specified visualizations. This circumstance considerably reduces the number of comparisons and potentially generates more detailed insights in the preferences. This approach follows the divide-and-conquer paradigm.

Figure 1 illustrates the learning procedure. It starts by selecting a feature F to learn from the set of all visualization features $\mathcal{F}$ based on a round robin scheme, i.e., we play a different feature in each round, following a fixed order. Once F is selected, the corresponding counting matrix is given to the D-TS. Since the D-TS only chooses two values f_i and f_j, we still have to compute two visualizations for the comparison. First, we use the two visualization sets $V_i = \{v \mid v = (\ldots, F = f_i, \ldots)\}$ and $V_j = \{v \mid v = (\ldots, F = f_j, \ldots)\}$ with $V_i \cap V_j = \emptyset$. Based on these sets, we compute a set of visualization pairs which are maximally similar according to the computed Hamming similarity (penalizes inequality): $V' = \{(v_i,\ v_j) \mid \gamma^* = sim(v_i, v_j)\ v_i \in V_i,\ v_j \in V_j\}$ with $\gamma^* = max\{sim(v_i, v_j) \mid v_i \in V_i,\ v_j \in V_j\}$. Finally, we randomly select a pair for the comparison: $(v_i, v_j) \sim U(V')$. Since we know v_i and v_j are similar but differ in F, the counting matrix of F will consequently be updated with respect to the feature values represented by the selected visualization. Continuing with this procedure, the next feature is explored during the next learning step.

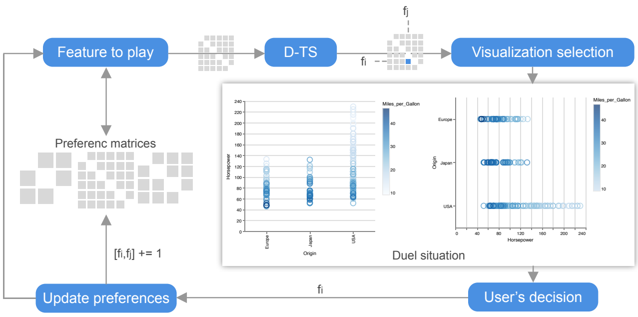


Figure 1: *At each step t, a visualization feature $F \in \mathcal{F}$ is selected. Based on both the corresponding counting matrix B_F and $\lceil \frac{t}{|\mathcal{F}|} \rceil$, D-TS selects two values f_i, $f_j \in F$. Afterwards, two maximally similar visualizations representing f_i and f_j, respectively, are shown to the user. Based on the user's decision, B_F is updated.*

3.2. Predicting Visualizations

For predicting visualizations based on the learned preferences, we compute the best candidate for each visualization feature by applying the routine of D-TS for choosing the first candidate for a duel. However, some visualization features influence the set of applicable values of other features. For example, having a mapping that includes a categorical attribute represented by the color, we cannot apply a diverging coloring scheme. Therefore, we initially rank the features based on their influence on other features. Following this ranking in descending order, we determine a value for each feature while considering potential restrictions imposed by previously selected feature values. Once all feature values are set, we generate the corresponding visualization.

4. Evaluation

To evaluate the presented approach we primarily focus on tri-dimensional visualizations. Furthermore, we limit the visual mappings to the channels *x*, *y*, and *color*. Additionally, we include two coloring schemes for both categorical (*dark2* vs. *set2*) and quantitative (*blues* vs. *greenblues*) as well as three different mark shapes (*circle*, *square*, and *point*). As a result, we get 36 different visualizations and $\binom{36}{2}$ unique pairwise comparisons.

As a testbed for the study, we implemented a technical prototype using a client-server architecture. While the frontend renders the visualizations and handles user interactions, the backend computes both the dueling bandit algorithm and the Vega-lite specifications [VGH*18]. Initially, we assume that all visualization are equally preferred by the user ($B_{t=0} \equiv 0^{k \times k}$), i.e., we add no prior knowledge to the bandit. Additionally, we use the setting originally proposed in [ZKWdR15, WL16].

4.1. Study Procedure

The study is designed to last approximately 30 minutes, starting with a standardized introduction to the procedure. For learning the preferences, we use the well-known car data set [VGH*18] restricted to the attributes *horsepower*, *miles per gallon*, and *origin*. After the introduction, each participant completes a sequence of 210 pairwise comparisons. For each pair, the participants decide which visualization they prefer. Although Wu and Liu [WL16] neglect the influence of the items' display order on the decision in a duel, we decide for a random display order to prevent biased data. In order to additionally evaluate the bandit's learning progress during training, a predicted visualization is given by the bandit after every 21 comparisons. This prediction is then rated by the participant on an 11-point scale (very unsatisfying – very satisfying).

After completing the 210 pairwise comparisons as well as rating the overall 10 predicted visualization, a sequence of 4 pairwise comparisons is given to the participant. Each comparison consists a visualization predicted by the bandit and a recommended visualization according to [KH18]. Additionally, each comparison bases on a different data attribute set from a weather data set [VGH*18]. This construct provides insights on the generalizability of the learned preferences. The study ends with a questionnaire on the acceptability of the learning method, and on the participant's demographics as well as experience in information visualization and statistics.

4.2. Participants

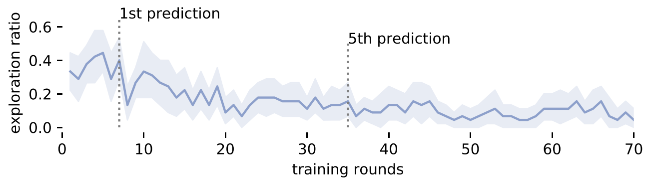
We recruited 15 participants (4 female, 11 male) with an average age of 26.4 ($\sigma = 3.18$) years. The participants' self-reported knowledge ranged from advanced beginners to competent users in both information visualization and statistics. As they stated to design their visualizations either with a dedicated tool (e.g., Tableau) (3), MS Excel (4), or directly in Python or R (8), the majority (13) further asserted to have an overall preferred visualization.

4.3. Results

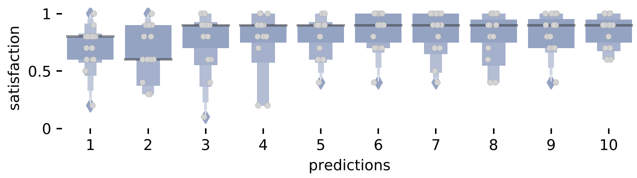
Overall, the participants decided for a preferred visualization in 4.42 sec per comparison on average. In accordance with the participants' feedback, the visualization features have varying importance for a decision on a preferred visualization: mapping was regarded as more important than type, which in turn was regarded as more important than coloring.

Learning: In the beginning, the participants tended to not select the visualizations for which the bandit expected to win. This circumstance continuously decreased during the use (see Figure 2(a)). This indicates that the bandit's certainty about the participant's preferences gradually increased. Although interactively teaching the bandit is a dull procedure, the majority of participants (11) stated that they would also conduct a similar procedure with an entire visualization tool, when they assume to get preference-aligned visualizations afterwards. Additionally, this interactive learning procedure received a mean NASA-TLX score of 24.26 ($\sigma = 9.64$), to which the *mental demand* mostly contributed (score: $\mu = 32.66, \sigma = 12.79$, weight: $\mu = .24, \sigma = .08$).

Predicting: As Figure 2(b) highlights, the bandit's predictions became better over time. Since the preferences are initially unknown, a higher chance for unsatisfying predictions exists in the beginning (satisfaction scores of the first three predictions: $\mu = .72, \sigma = .22$). However, the participant's satisfaction with the predictions significantly improves over time (satisfaction scores of the



(a) Ratio (mean and 95% confidence) of how often the participants did not prefer the visualization assumed by the dueling bandit.



(b) Participants' satisfaction with the predicted visualizations

Figure 2: *The visualizations illustrate the preference learning improvements of the dueling bandit. While (a) highlights how often the participants did not selected the visualization assumed by the bandit, (b) shows the satisfaction of the participants with the predicted visualizations.*

last three predictions: $\mu = .82, \sigma = .16$), according to a Wilcoxon Signed-Rank test, $Z = 92, p < .01$. On average, each participant received 3.4 ($\sigma = 0.87$) different visualizations during these prediction steps. Considering the last predicted visualization of each user as an indicator of the participants' preferences, we found an overall Hamming distance between them of $\mu = .57, \sigma = .39$ as well as overall 10 unique visualizations. Furthermore, 14 participants (strongly) agreed that the bandit actually learned their preferences. Considering the comparison with the rule-based approach, the participants preferred the visualizations given by the bandit in 91% of all cases. In 20% of the comparisons both visualizations shared the same visual mapping.

5. Discussion

The results underline not only the effectiveness of the dueling bandit approach for learning the individual visualization preferences, but also address the acceptance of the training procedure.

Training time reduction: We had initially little knowledge about how many training steps the bandit would actually needed in order to properly retrieve preference-aligned visualizations. As the bandit's performance stabilizes veritably after approximately the fifth prediction (cf. Figure 2), the number of needed comparisons can potentially be decreased. This insight helps to better estimate the needed effort for the user.

Factorization for learning vs. feature combination effects: A key choice of our methodology is to base preference learning on individual visualization features, rather than on fully specified visualizations. This choice reduces the needed comparisons for the bandit to learn the user's preferences. However, there are presumably dependencies between these visualization features (e.g., coloring scheme C might be preferred with mark shape A, but not with mark shape B), which potentially influence the decisions. This effect might be even larger when the visualization types are considerably different, e.g., bar chart vs. pie chart. However, the participants were very satisfied with the last predicted visualizations and perceived the training effort as relatively low.

Generalizability of learned preferences: The bandit was clearly preferred to the rule-based approach. Since each comparison based on a unique set of data attributes, it appears that the learned preferences are not necessarily bound to the data on which the bandit was trained on. This implies both the user does not have to retrain the model and the preferences matrices can be used for adjusting current systems' recommendations.

Enhancing visualization recommenders: In case the user's preferences are learned on the same visualization set used by the recommender, the corresponding preference matrix can be conceptually seen as a weighting matrix for adjusting the ranking of the visualizations. This preference matrix can further be persistently stored in the recommender's user model. A persistent preference matrix can be used for predictions even after restarting the system. Additionally, and most importantly, it can serve as prior knowledge for new users, e.g., via the weighted average of known users' preferences.

6. Conclusion and Future Work

In this paper, we explored the challenge of online learning of visualization preferences to enhance visualization recommender systems. We particularly investigated the reinforcement learning methodology of dueling bandits. For usability purposes, we envision a divide-and-conquer approach to accelerate the learning process. The results of the user study provide insights into the needed training time of the bandit, the performance of the approach, and the user acceptance of the resulting recommendations. As knowledge about the effectiveness of visualization designs are commonly integrated into recommender systems, our results contribute to further improve the performance of personalized recommenders.

We see particularly three areas for future work. First, as a variety of other dueling bandit algorithms with different constraints exist, a comparison between good candidates should be investigated. The focus should further not only be on how accurately the preferences are learned, but also on the effort for the user. Second, in our study, we initially considered three different visualization features, but there are more. Investigating the effect of extending the feature space on both the learning procedure and the individual feature importance scores is an interesting avenue for future work. Third, since we provided insights on the generalizability from one data set to another, a potential effect of the task, the domain, or the amount of visualized data attributes on the preferences should be explored.

7. Acknowledgement

Any opinions, findings, and conclusions expressed in this paper do not necessarily reflect the views of the Volkswagen Group.

References

[BHM18] BUSA-FEKETE R., HÜLLERMEIER E., MESAOUDI-PAUL A. E.: Preference-based online learning with dueling bandits: A survey. *CoRR abs/1807.11398* (2018). URL: http://arxiv.org/abs/1807.11398, arXiv:1807.11398. 1, 2

[CCH*14] CONATI C., CARENINI G., HOQUE E., STEICHEN B., TOKER D.: Evaluating the impact of user characteristics and different layouts on an interactive visualization for decision making. In *Proceedings of the 16th Eurographics Conference on Visualization* (Aire-la-Ville, Switzerland, Switzerland, 2014), EuroVis '14, Eurographics Association, pp. 371–380. doi:10.1111/cgf.12393. 1

[CZK16] CHAN H. P., ZHAO T., KING I.: Trust-aware peer assessment using multi-armed bandit algorithms. In *Proceedings of the 25th International Conference Companion on World Wide Web* (Republic and Canton of Geneva, Switzerland, 2016), WWW '16 Companion, International World Wide Web Conferences Steering Committee, pp. 899–903. doi:10.1145/2872518.2891080. 1

[DD18] DIBIA V., DEMIRALP Ç.: Data2Vis: Automatic generation of data visualizations using sequence to sequence recurrent neural networks. *CoRR abs/1804.03126* (2018). URL: http://arxiv.org/abs/1804.03126. 1, 2

[DHPP17] DEMIRALP C., HAAS P. J., PARTHASARATHY S., PEDAPATI T.: Foresight: Recommending visual insights. *Proceedings of the VLDB Endowment 10*, 12 (Aug. 2017), 1937–1940. doi:10.14778/3137765.3137813. 2

[GF10] GREEN T. M., FISHER B.: Towards the personal equation of interaction: The impact of personality factors on visual analytics interface interaction. In *2010 IEEE Symposium on Visual Analytics Science and Technology* (Oct 2010), pp. 203–210. doi:10.1109/VAST.2010.5653587. 1

[HBL*18] HU K. Z., BAKKER M. A., LI S., KRASKA T., HIDALGO C. A.: VizML: A machine learning approach to visualization recommendation. *CoRR abs/1808.04819* (2018). URL: http://arxiv.org/abs/1808.04819. 1, 2

[HS88] HART S. G., STAVELAND L. E.: Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Human Mental Workload*, Hancock P. A., Meshkati N., (Eds.), vol. 52 of *Advances in Psychology*. North-Holland, 1988, pp. 139 – 183. doi:10.1016/S0166-4115(08)62386-9. 1

[HYFC14] HARRISON L., YANG F., FRANCONERI S., CHANG R.: Ranking visualizations of correlation using Weber's law. *IEEE Transactions on Visualization and Computer Graphics 20*, 12 (Dec 2014), 1943–1952. doi:10.1109/TVCG.2014.2346979. 1

[KH16] KAY M., HEER J.: Beyond Weber's law: A second look at ranking visualizations of correlation. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (Jan 2016), 469–478. doi:10.1109/TVCG.2015.2467671. 1

[KH18] KIM Y., HEER J.: Assessing Effects of Task and Data Distribution on the Effectiveness of Visual Encodings. *Computer Graphics Forum* (2018). doi:10.1111/cgf.13409. 1, 3

[KHPA12] KEY A., HOWE B., PERRY D., ARAGON C.: VizDeck: Self-organizing dashboards for visual analytics. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 2012), SIGMOD '12, ACM, pp. 681–684. doi:10.1145/2213836.2213931. 2

[LQT*18] LUO Y., QIN X., TANG N., LI G., WANG X.: DeepEye: Creating good data visualizations by keyword search. In *Proceedings of the 2018 International Conference on Management of Data* (New York, NY, USA, 2018), SIGMOD '18, ACM, pp. 1733–1736. doi:10.1145/3183713.3193545. 1, 2

[Mac86] MACKINLAY J.: Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics 5*, 2 (Apr. 1986), 110–141. doi:10.1145/22949.22950. 1, 2

[MHS07] MACKINLAY J., HANRAHAN P., STOLTE C.: Show Me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (Nov 2007), 1137–1144. doi:10.1109/TVCG.2007.70594. 1, 2

[MVT16] MUTLU B., VEAS E., TRATTNER C.: VizRec: Recommending personalized visualizations. *ACM Transactions on Interactive Intelligent Systems 6*, 4 (Nov. 2016), 31:1–31:39. doi:10.1145/2983923. 1, 2

[MWN*19] MORITZ D., WANG C., NELSON G. L., LIN H., SMITH A. M., HOWE B., HEER J.: Formalizing visualization design knowledge as constraints: Actionable and extensible models in Draco. *IEEE Transactions on Visualization and Computer Graphics 25*, 1 (Jan 2019), 438–448. doi:10.1109/TVCG.2018.2865240. 1, 2

[SK17] SCHNEIDER S., KUMMERT F.: Exploring embodiment and dueling bandit learning for preference adaptation in human-robot interaction. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)* (Aug 2017), pp. 1325–1331. doi:10.1109/ROMAN.2017.8172476. 1

[SKL*16] SIDDIQUI T., KIM A., LEE J., KARAHALIOS K., PARAMESWARAN A.: Effortless data exploration with Zenvisage: An expressive and interactive visual analytics system. *Proceedings of the VLDB Endowment 10*, 4 (Nov. 2016), 457–468. doi:10.14778/3025111.3025126. 2

[SMWH17] SATYANARAYAN A., MORITZ D., WONGSUPHASAWAT K., HEER J.: Vega-lite: A grammar of interactive graphics. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (Jan 2017), 341–350. doi:10.1109/TVCG.2016.2599030. 2

[VGH*18] VANDERPLAS J., GRANGER B., HEER J., MORITZ D., WONGSUPHASAWAT K., SATYANARAYAN A., LEES E., TIMOFEEV I., WELSH B., SIEVERT S.: Altair: Interactive statistical visualizations for python. *Journal of Open Source Software* (dec 2018). URL: https://doi.org/10.21105/joss.01057, doi:10.21105/joss.01057. 3

[VRM*15] VARTAK M., RAHMAN S., MADDEN S., PARAMESWARAN A., POLYZOTIS N.: SeeDB: Efficient data-driven visualization recommendations to support visual analytics. *Proceedings of the VLDB Endowment 8*, 13 (Sept. 2015), 2182–2193. doi:10.14778/2831360.2831371. 2

[WL16] WU H., LIU X.: Double Thompson sampling for dueling bandits. In *Advances in Neural Information Processing Systems 29*, Lee D. D., Sugiyama M., Luxburg U. V., Guyon I., Garnett R., (Eds.). Curran Associates, Inc., 2016, pp. 649–657. URL: https://arxiv.org/abs/1604.07101. 2, 3

[WQM*17] WONGSUPHASAWAT K., QU Z., MORITZ D., CHANG R., OUK F., ANAND A., MACKINLAY J., HOWE B., HEER J.: Voyager 2: Augmenting visual analysis with partial view specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2017), CHI '17, ACM, pp. 2648–2659. doi:10.1145/3025453.3025768. 1, 2

[YBKJ12] YUE Y., BRODER J., KLEINBERG R., JOACHIMS T.: The k-armed dueling bandits problem. *Journal of Computer and System Sciences 78*, 5 (Sept. 2012), 1538–1556. doi:10.1016/j.jcss.2011.12.028. 1

[ZKWdR15] ZOGHI M., KARNIN Z. S., WHITESON S., DE RIJKE M.: Copeland dueling bandits. In *Advances in Neural Information Processing Systems 28*, Cortes C., Lawrence N. D., Lee D. D., Sugiyama M., Garnett R., (Eds.). Curran Associates, Inc., 2015, pp. 307–315. URL: http://arxiv.org/abs/1506.00312. 2, 3

[ZOC*12] ZIEMKIEWICZ C., OTTLEY A., CROUSER R. J., CHAUNCEY K., SU S. L., CHANG R.: Understanding visualization by understanding individual users. *IEEE Computer Graphics and Applications 32*, 6 (Nov 2012), 88–94. doi:10.1109/MCG.2012.120. 1