Proceedings of the Twelfth International
Symposium on Combinatorial Search (SoCS 2019)

Algorithms for Average Regret Minimization

Sabine Storandt
University of Konstanz, Germany
sabine.storandt@uni-konstanz.de

Stefan Funke
University of Stuttgart, Germany
funke@fmi.uni-stuttgart.de

Abstract

In this short paper, we summarize the results on regret minimization obtained in (Storandt and Funke 2019). Given a set of objects where each object has multiple attributes, regret is a useful measure to select a subset of bounded size which represents the full set of objects well. We propose a new variant of regret – the so called *average regret* – and design a greedy algorithm which computes a subset with provably small average regret in polynomial time. In addition, we conduct experiments on real-world instances to evaluate the performance of the greedy algorithm in practice.

Motivation

In multicriteria decision making, there is often an abundance of options to choose from. For example, the number of products of a certain type in an online warehouse (as, e.g., laptops) is easily greater than one hundred. Showing all options to the user at once is then either impossible or leads to an incomprehensible result representation. Therefore, the goal is to preselect a small subset of the available options and only present that subset to the user. But as user preferences may differ widely, it is not a priori clear which subset is the best. Some users might, e.g., prefer cheap products, others prefer products with good reviews or with certain other product specific aspects. The regret measure allows to take all these user preferences into account when deciding for the best subset. The idea is that each user preference can be formalized as a maximization function f over the attributes of the objects in the set. Given a set S of objects, $f(S)$ then denotes the utility of the best object in S for the user. Accordingly, the regret of a single user that is presented with a subset S' of the whole set of options S can be measured as $1 - f(S')/f(S)$. The smallest regret value of 0 is achieved when the user is perfectly happy with choosing an option from S' instead of S as the best option has still the same utility for him. The largest regret value of 1 is achieved when all options remaining in S' have zero utility for the user. In general, the regret value can assume any value in $[0, 1]$.

Given not only a single user preference function but a class of such functions, the maximum regret for a subset S' of S is the largest individual regret of any user. Maximum regret as a measure for subset quality was introduced in (Nanongkai et al. 2010). The maximum regret minimization problem then asks for a subset of bounded size with the smallest maximum regret. Two greedy algorithms were proposed in (Chester et al. 2014) and (Nanongkai et al. 2010) to tackle the maximum regret minimization problem. However, no approximation guarantees could be proven for those. There indeed exist approximation algorithms for maximum regret minimization (Agarwal et al. 2017; Cao et al. 2017; Asudeh et al. 2017; Kumar and Sintos 2018), but these algorithms require more complex machinery.

Maximum regret as a measure comes with some disadvantages, though. The most prominent is the so called 'drowning effect'. As the value of the measure is in the end determined by the regret of a single user, the regret of all users other than the most unhappy one do not contribute to said value. This might lead to a strange subset selection in practice. Therefore, we propose the average regret as a viable alternative: Instead of minimizing the regret of the most unhappy user, we minimize the summed regret of all possible users.

Theoretical Results

Given a finite set S of d-dimensional objects and a parameter $q \in \mathbb{N}$, the average regret minimization problem asks for the subset $S' \subseteq S$ of size q with the smallest possible average regret among all subsets of that size.

The objects are given as points in $\mathbb{R}^d_+$. The class of user preference or utility functions we consider here is the set of all possible convex linear functions. This is a standard model for user utility functions also used, e.g., in (Soma and Yoshida 2017).

The main theoretical result obtained is that the average regret function is supermodular while for the maximum regret function we can construct examples where the conditions for supermodularity are violated. Supermodularity is a desirable property as it allows to construct a reverse greedy algorithm which selects a subset of size q of the objects with an average regret value within an instance-based bounded factor of the optimum. Furthermore, we define the average happiness function as 1 minus the regret value. We show that this function is submodular. Therefore, a classical greedy algorithm can be used to obtain a $1 - {}^1\!/_e$ approximation.

Both, the classical and the reverse greedy algorithm require the efficient computation of the average regret of a given object subset S' as a subroutine. We show that this problem can be translated into the problem of computing the hypervolume of the convex hull induced by the points in that subset. In the worst case, this computation takes time $\mathcal{O}(|S'|^{d^2/4})$. Although for any fixed d this yields a polynomial time algorithm, we also devise a faster sampling-based heuristic which estimates the respective volume.

Practical Results

We implemented the mentioned greedy algorithms and the exact and heuristic volume computation as subroutines. As benchmarks, we used real-world data (oceanographic data, flight data and weather data) as well as synthetic data (random points in a hypercube). The dimension d varies between 2 and 7.

For small desired subset sizes q, our experiments reveal that the sampling-based volume computation produces the same result as the exact volume computation. But for $q = 16$, for example, using exact volume computation leads to smaller regret values (up to a factor of 40). This comes at the price of higher running times: While for 1000 synthetic points with $d = 6$ and $q = 16$ the exact volume computations within the greedy algorithm already take several hours, the sampling-based heuristic leads to running times of less than a second. For our most complex real-world benchmark (about half a million points, $d = 7$), we could only produce results when using the sampling-based approach as exact volume computation took too long.

Furthermore, we compared the classical and the reverse greedy algorithm, which both can be used to tackle the average regret minimization problem. Figure 1 shows the quality of the result sets obtained with either of the two algorithms for various values of q (on the synthetic benchmark with $d = 4$ and $|S| = 1000$). It becomes obvious that the classical greedy algorithm performs way better, especially for small values of q. At the same time, the standard greedy algorithm is also superior with respect to running time. This makes the standard greedy approach the clear winner.

Note that the greedy algorithm can also be used to compute a permutation of the objects in the input set S such that for any prefix of size q of the objects in the resulting order, we have a provable guarantee on the average regret with respect to the optimum solution of size q. This allows to solve the problem not only for a single given value of q but instead for all possible values of q at once.

Conclusions and Future Work

Our theoretical and practical results show that subsets with small average regret can be efficiently computed, at least for moderate dimensions d. To faithfully compute the average regret of object sets with more attributes (e.g., $d \geq 10$), a fast approximation algorithm for the hypervolume computation would be necessary. Furthermore, it would be interesting to study the average regret measure on utility function classes other than the class of convex linear functions.

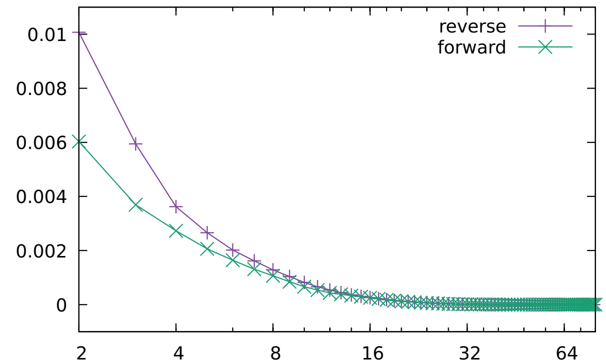

Figure 1: Average regret in dependency of the subset size q for the standard (forward) greedy algorithm as well as for the reverse greedy algorithm (x-axis in logscale).

References

Agarwal, P. K.; Kumar, N.; Sintos, S.; and Suri, S. 2017. Efficient algorithms for k-regret minimizing sets. *arXiv preprint arXiv:1702.01446*.

Asudeh, A.; Nazi, A.; Zhang, N.; and Das, G. 2017. Efficient computation of regret-ratio minimizing set: A compact maxima representative. In *Proceedings of the 2017 ACM International Conference on Management of Data*, 821–834. ACM.

Cao, W.; Li, J.; Wang, H.; Wang, K.; Wang, R.; Chi-Wing Wong, R.; and Zhan, W. 2017. k-regret minimizing set: Efficient algorithms and hardness. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 68. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Chester, S.; Thomo, A.; Venkatesh, S.; and Whitesides, S. 2014. Computing k-regret minimizing sets. *Proceedings of the VLDB Endowment* 7(5):389–400.

Kumar, N., and Sintos, S. 2018. Faster approximation algorithm for the k-regret minimizing set and related problems. In *2018 Proceedings of the Twentieth Workshop on Algorithm Engineering and Experiments (ALENEX)*, 62–74. SIAM.

Nanongkai, D.; Sarma, A. D.; Lall, A.; Lipton, R. J.; and Xu, J. 2010. Regret-minimizing representative databases. *Proceedings of the VLDB Endowment* 3(1-2):1114–1124.

Soma, T., and Yoshida, Y. 2017. Regret ratio minimization in multi-objective submodular function maximization. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, 905–911.

Storandt, S., and Funke, S. 2019. Algorithms for average regret minimization. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, January 27 - February 1, 2019, Honululu, Hawaii, USA.*