Submodular sketches of single-cell RNA-seq measurements

Wei Yang[1], Jacob Schreiber[1], Jeffrey Bilmes[2,3], and William Stafford Noble*[1,3]

[1]Department of Genome Sciences, University of Washington
[2]Department of Electrical and Computer Engineering, University of Washington
[3]Paul G. Allen School of Computer Science and Engineering, University of Washington

Abstract

Analyzing and sharing massive single-cell RNA-seq data sets can be facilitated by creating a "sketch" of the data—a selected subset of cells that accurately represent the full data set. Using an existing benchmark, we demonstrate the utility of submodular optimization in efficiently creating high quality sketches of scRNA-seq data.

By capturing variation of gene expression within a population of cells, single-cell RNA-seq (scRNA-seq) measurements add an additional dimension to already large gene expression datasets. As a result, scRNA-seq datasets can be massive. For example, a recent scRNA-seq analysis of 61 staged mouse embryos yielded measurements of >2 million cells.[1] Clearly, meta-analyses that aim to aggregate scRNA-seq data from multiple such studies will be challenging.

One common strategy to facilitate analysis of very large datasets is to identify and remove redundant examples. In scRNA-seq, this strategy can be used to select a subset of the cells in an experiment that show different patterns of gene expression. The selected subset, sometimes referred to as a "sketch" of the full dataset, can then be analyzed using clustering or cell type assignment methods.[3] A recently described method for selecting sketches of scRNA-seq data, Geosketch, is based on minimizing a particular distance function—the Hausdorff distance—between the full dataset (the "ground set") and the sketch.[8]

Here, we propose *submodular optimization* as a theoretically-grounded and powerful framework for selecting a sketch of scRNA-seq data. Loosely speaking, submodular optimization can be considered a discrete analog of convex optimization, where the goal is to identify a set of discrete elements, rather than a collection of continuous values, that optimize an objective function.

Submodular functions are set functions that satisfy the property of diminishing returns: if we think of a function $f(A)$ as measuring the value of a sketch A that is a subset of a larger set of data items $A \subseteq V$, then the submodular property means that the incremental "value" of adding a data item s to a sketch A decreases as the size of A grows (e.g., $f(A+s)-f(A) \geq f(B+s)-f(B)$ whenever $A \subseteq B$ and $s \notin B$). Unfortunately, searching for a sketch of maximal quality, as measured by $f(A)$, is computationally infeasible for an arbitrary set function; however, when the set function is submodular, then the quality can be approximately maximized (i.e., the quality of the identified solution is a least constant factor times optimal) in low-order polynomial time[6;17;19]. Moreover, the approximation ratio achieved by these optimization algorithms is provably the best achievable in polynomial time, assuming $P \neq NP$. For these reasons, submodular optimization has a long history in economics,[2;27] game theory,[24;25] combinatorial optimization,[5;16;23] electrical networks,[18] and operations research.[4] Furthermore, submodular optimization has recently been used with great success for selecting sketches of text documents,[12–14] recorded speech,[15;29;30] image compendia,[26] sets of protein sequences,[11] sets of genomics assays,[28] and sets of genomic loci.[7]

The Hausdorff distance that Geosketch minimizes is not submodular but can be viewed as a robust form of a commonly used submodular function (Supplementary Note 1). Hence, we hypothesized that switching to a submodular optimization framework, with its fast algorithms and performance guarantees, would yield better (or at least as good) sketches much more quickly than Geosketch.

*Corresponding author: william-noble@uw.edu

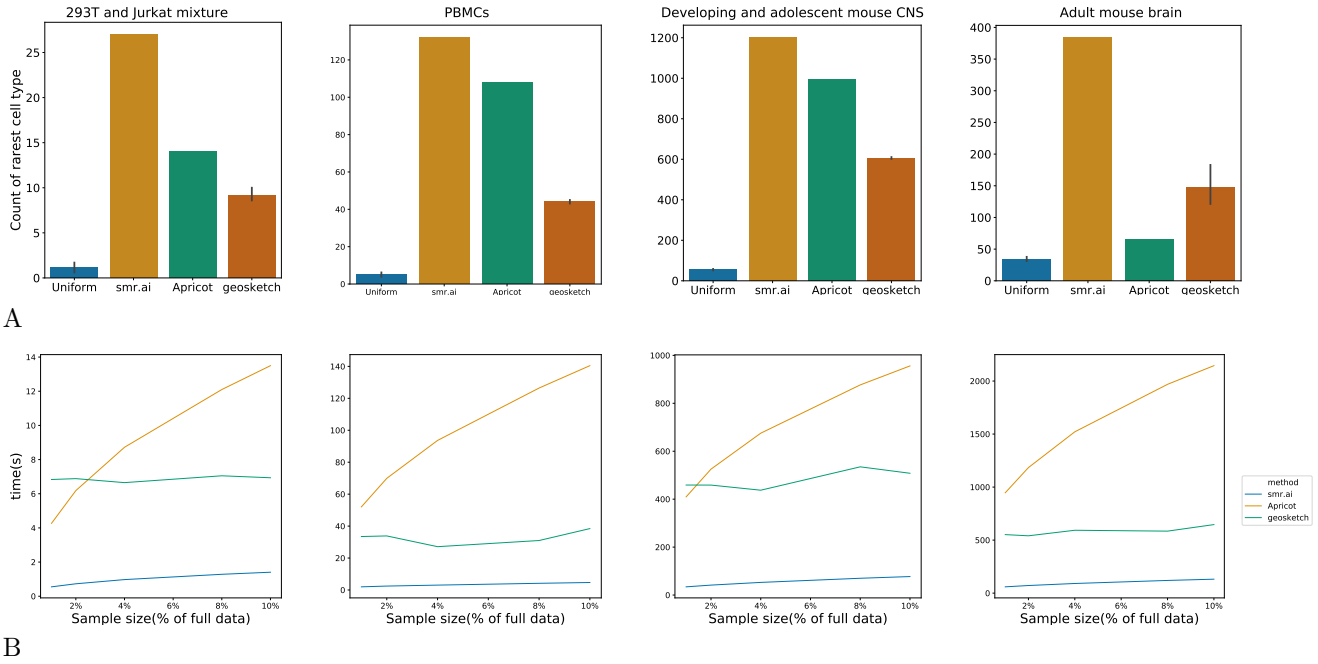

Figure 1: (A) Each panel plots, for a specified benchmark dataset, the number of cells of the rarest type that were selected by each of the four sketching methods. (B) Each panel compares the wall clock times required by Geosketch, apricot and `smr.ai` to produce sketches of varying sizes.

To test this hypothesis, we applied two submodular optimization toolkits—an open source Python package called "apricot" [22] and a commercial tool provided by Summary Analytics Inc. (`smr.ai`)—to the four benchmark datasets that were analyzed in the Geosketch paper. To evaluate the methods, we used the same performance measure employed by Hie *et al.*, namely, the count of the number of cells of the rarest cell type that are included in the sketch. Here, the intuition is that a good sketch is one that undersamples common cells that presumably occupy dense regions of the transcriptional space. We observe that, in all four datasets, the sketches produced by the submodular approaches outperform Geosketch, often by a large margin (Figure 1A). Both submodular approaches use roughly the same objective function, though the commercial tool includes proprietary forms of the user-specified objective. These modifications explain the overall better performance of smr.ai relative to apricot on these evaluation tasks. The submodular approaches also compare favorably to Geosketch in running time (Figure 1B). The apricot software, which is implemented in Python using both numpy [20] and numba [10] to accelerate computation, performs comparably to Geosketch for smaller selected sets but runs more slowly as the size of the selected set gets larger. On the other hand, the smr.ai tool, which is implemented in C++, is generally an order of magnitude faster than Geosketch.

Overall, our empirical results suggest that submodular optimization provides a powerful and efficient way to summarize large-scale scRNA-seq datasets. Of particular note, the results we report here have not been optimized with respect to parameter selection—they represent the very first objective function that we tried. The space of submodular functions is large and very diverse, and particular functions can in principle be constructed to obtain particular types of sketches. Thus, users of these tools may wish to experiment with varying the form of the submodular objective to obtain sketches that are, for example, particularly enriched in rare cell types or that pay particular attention to capturing outliers.

Methods

Data

We downloaded four sc-RNA-seq datasets previously used to assess the performance of Geosketch: a 293T and Jurkat mixture dataset with 4,185 cells, [32] a peripheral blood mononuclear cell (PBMC) dataset with

68,579 cells,[32] a developing and adolescent central nervous system (CNS) dataset with 465,281 cells,[31] and an adult mouse brain dataset with 665,858 cells.[21] For each dataset, we focused on the same rarest cell type as Geosketch: 28 293T cells (0.66% of the total number of cells in the dataset) in the 293T and Jurkat mixture dataset, 262 dendritic cells (0.38%) in the PBMC dataset, 2,777 ependymal cells (0.60%) in the mouse CNS dataset, and 1695 macrophages (0.25%) in the adult mouse brain dataset.

Submodular optimization

We ran both apricot and smr.ai using a "feature-based" objective[9] of the form

$$f(A) = \sum_{u=1}^{m} \sqrt{\sum_{a \in A} x_u(a)}, \tag{1}$$

where X is a dataset represented as an $n \times m$ matrix with columns x_u for feature u, u is the index of a single feature in the dataset, m is the number of features in the dataset, and $x_u(a)$ is the value of feature u for example a. The square root function is critical, since its concavity is what provides a diminishing returns property. Note that, in general, this type of feature-based selection requires non-negative feature values (i.e., $x_u(a) \geq 0$). In the case of scRNA-seq data, the read counts are naturally non-negative. For both smr.ai and apricot we subsequently normalized the read counts for each cell to a unit vector, maintaining the non-negativity of each feature. Prior to analysis by apricot, the expression values of each gene were linearly rescaled to the range $[0, 1]$. Feature-based selection was carried out by calling apricot's `FeatureBasedSelection` function with default parameters.

The apricot software is freely available under MIT license at https://github.com/jmschrei/apricot. The smr.ai software can be, on request, made to be used freely by academic researchers via http://smr.ai.

Geosketch

Geosketch was imported as a Python package obtained from http://cb.csail.mit.edu/cb/geosketch. Each data matrix was reduced to a dimensionality of 100 PCs by randomized PCA before the sketch, as implemented in Geosketch. The matrix of 100 PCs and subset size were then passed to the `gs_gap` function to generate a list of indices of selected cells. Note that we consider the randomized PCA part of the Geosketch method. Hence, in contrast to the results reported in Hie *et al.*, the error bars in Figure 1A reflect variation from both the randomized PCA and the Geosketch algorithm, rather than just the Geosketch algorithm.

Timing

To compare running times, we generated sketches of varying sizes on a single thread on a 2.70GHz Intel Xeon CPU E5-2680 with 250 GB of memory. Wall clock times for apricot and Geosketch were recorded using the Python "time" module. For smr.ai, we used the wall clock time reported by the software. As is common in the high-performance computing literature, we repeated each timing test 10 times and reported the minimum wall clock time, to reduce any possible effect of operating system interference.

Author Contributions Data analysis was carried out by WY, JS, and WSN. The manuscript was written by WY and WSN, and was edited by all authors

Competing Interests JB holds a commercial interest in smr.ai. WY, JS and WSN declare no competing interests.

References

[1] J. Cao, M. Spielmann, X. Qiu, X. Huang, D. M. Ibrahim, A. J. Hill, F. Zhang, S. Mundlos, L. Christiansen, F. J. Steemers, C. Trapnell, and J. Shendure. The single-cell transcriptional landscape of mammalian organogenesis. *Nature*, 566:496–502, 2019.

[2] M. Carter. *Foundations of Mathematical Economics*. The MIT Press, 2001.

[3] G. Cormode. Data sketching. *Communications of the ACM*, 60(9):48, 2017.

[4] G. Cornunéjols, G. L. Nemhauser, and L. A. Wolsey. The uncapacitated facility location problem. In P.B. Mirchandani and R.L. Franci, editors, *Discrete Location Theory*, chapter 3, pages 119–171. Wiley/Interscience, New York, 1990.

[5] J. Edmonds. Matroids, submodular functions, and certain polyhedra. *Combinatorial Structures and Their Applications*, pages 69–87, 1970.

[6] M.L. Fisher, G.L. Nemhauser, and L.A. Wolsey. An analysis of approximations for maximizing submodular set functions—II. *Polyhedral combinatorics*, pages 73–87, 1978.

[7] M. Gasperini, A. J. Hill, J. L. McFaline-Figueroa, B. Martin, S. Kim, D. Jackson, A. Leith, J. Schreiber, W. S. Noble, C. Trapnell, N. Ahituv, and J. Shendure. A genome-wide framework for mapping gene regulation via cellular genetic screens. *Cell*, 176:377–390, 2019.

[8] B. Hie, H. Cho, B. DeMeo, B. Bryson, and B. Berger. Geometric sketching compactly summarizes the single-cell transcriptomic landscape. *Cell Systems*, 8:483–493, 2019.

[9] K. Kirchhoff and J. Bilmes. Submodularity for data selection in machine translation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[10] S. K. Lam, A. Pitrou, and S. Seibert. Numba: A llvm-based python jit compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, LLVM '15, pages 7:1–7:6, New York, NY, USA, 2015. ACM.

[11] M. W. Libbrecht, J. A. Bilmes, and W. S. Noble. Choosing non-redundant representative subsets of protein sequence data sets using submodular optimization. *Proteins*, 86(4):454–466, 2018.

[12] G. Lin, M. K. Chawla, K. Olson, C. A. Barnes, J. F. Guzowski, C. Bjornsson, W. Shain, and B. Roysam. A multi-model approach to simultaneous segmentation and classification of heteregenous populations of cell nuclei in 3d confocal microscope images. *Cytometry A.*, 71(9):724–736, 2007.

[13] H. Lin and J. Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 510–520. Association for Computational Linguistics, 2011.

[14] H. Lin and J. Bilmes. Learning mixtures of submodular shells with application to document summarization. In *Uncertainty in Artificial Intelligence (UAI)*, pages 479–490, Catalina Island, USA, July 2012. AUAI.

[15] Y. Liu, K. Wei, K. Kirchhoff, Y. Song, and J. Bilmes. Submodular feature selection for high-dimensional acoustic score spaces. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 7184–7188. IEEE, 2013.

[16] L. Lovász. Submodular functions and convexity. In M. Grotchel A. Bachem and B. Korte, editors, *Mathematical Programming – The State of the Art*, pages 235–257. Springer-Verlag, Bonn, 1983.

[17] M. Minoux. Accelerated greedy algorithms for maximizing submodular set functions. *Optimization Techniques*, pages 234–243, 1978.

[18] H. Narayanan. Submodular functions and electrical networks. *Annals of Discrete Mathematics*, 54, 1997.

[19] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978.

[20] T. E. Oliphant. *Guide to NumPy*. CreateSpace Independent Publishing Platform, North Charleston, SC, 2006.

[21] A. Saunders, E.Z. Macosko, A. Wysoker, M. Goldman, F.M. Krienen, H. de Rivera, E. Bien, M. Baum, L. Bortolin, S. Wang, A. Goeva, J. Nemesh, N. Kamitaki, S. Brumbaugh, D. Kulp, and S. A. McCarroll. Molecular diversity and specializations among the cells of the adult mouse brain. *Cell*, 174:999–1014, 2018.

[22] J. M. Schreiber, J. Bilmes, and W. S. Noble. apricot: Submodular selection for data summarization in python. *arXiv*, 2019. https://arxiv.org/abs/1906.03543.

[23] A. Schrijver. *Combinatorial Optimization*. Springer, 2004.

[24] L. S. Shapley. Cores of convex games. *International Journal of Game Theory*, 1(1):11–26, 1971.

[25] D. M. Topkis. *Supermodularity and complementarity*. Princeton University Press, 1998.

[26] S. Tschiatschek, R. K. Iyer, H. Wei, and J. A. Bilmes. Learning mixtures of submodular functions for image collection summarization. In *Advances in Neural Information Processing Systems*, pages 1413–1421, 2014.

[27] X. Vives. *Oligopoly pricing: Old ideas and new tools*. The MIT Press, 2001.

[28] K. Wei, M. W. Libbrecht, J. A. Bilmes, and W. S. Noble. Choosing panels of genomics assays using submodular optimization. *Genome Biology*, 17(1):229, 2016.

[29] K. Wei, Y. Liu, K. Kirchhoff, C. Bartels, and J. Bilmes. Submodular subset selection for large-scale speech training data. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 3311–3315. IEEE, 2014.

[30] K. Wei, Y. Liu, K. Kirchhoff, and J. Bilmes. Using document summarization techniques for speech data subset selection. In *HLT-NAACL*, pages 721–726, 2013.

[31] A. Zeisel, H. Hochgerner, P. Lönnerberg, A. Johnsson, F. Memic, J. van der Zwan, M. Häring, E. Braun, L.E. Borm, G. La Manno, S. Codeluppi, A. Furlan, K. Lee, N. Skene, K. D. Harris, J. Hjerling-Leffler, E. Arenas, P. Ernfors, U. Marklund, and S. Linnarsson. Molecular architecture of the mouse nervous system. *Cell*, 174:999–1014, 2018.

[32] G. X. Y. Zheng, J. M. Terry, P. Belgrader, P. Ryvkin, Z. W. Bent, R. Wilson, S. B. Ziraldo, T. D. Wheeler, G. P. McDermott, J. Zhu, M. T. Gregory, J. Shuga, L. Montesclaros, J. G. Underwood, D. A. Masquelier, S. Y. Nishimura, M. Schnall-Levin, P. W. Wyatt, C. M. Hindson, R. Bharadwaj, A. Wong, K. D. Ness, L. W. Beppu, H. J. Deeg, C. McFarland, K. R. Loeb, W. J. Valente, N. G. Ericson, E. A. Stevens, J. P. Radich, T. S. Mikkelsen, B. J. Hindson, and J. H. Biela. Massively parallel digital transcriptional profiling of single cells. *Nature Communications*, 8:14049, 2017.

A Supplementary Note 1

The goal of Hie *et al.*[8] is to minimize the Hausdorff distance between the fixed ground set (they call it $\mathcal{X}$, we call it V) and the subset (they call it S, but we use $A \subseteq V$ above). Hence, V is fixed and only $S \subseteq V$ is variable, and their goal is to perform the following optimization:

$$\min_{S \subseteq V : |S| = k} d_H(V, S), \tag{2}$$

where $d_H(T, S) = \max_{t \in T} \min_{s \in S} d(t, s)$ for arbitrary subsets $T, S \subseteq V$. This expression is parameterized by element-pair distances, i.e., $d(t, s)$ for $t, s \in V$. This optimization in Equation (2) is equivalent to the problem

$$\max_{S \subseteq V : |S| \leq k} \min_{v \in V} \max_{s \in S} a(v, s)$$

where $a(v, s) = \alpha - d(v, s)$ is an affinity (or similarity) between pair v, s and where α is a positive constant. If we define

$$f_r(S) = \min_{v \in V} \max_{s \in S} a(v, s),$$

then this objective is a minimization over a set of submodular functions, since the function $g_v(S) = \max_{s \in S} a(v, s)$ is submodular in S for all $v \in V$. A well-known standard submodular function known as the facility location function has the form

$$f(S) = \sum_{v \in V} \max_{s \in S} a(v, s).$$

When we compare $f_r(S)$ with $f(S)$ we see that $f(S)$ is a sum over the set of submodular functions $\{f_v(S)\}$ while $f_r(S)$ is the minimization over the same set of submodular functions. Hence, Geosketch, which maximizes f_r, maximizes the worst case over the set $\{f_v(S)\}$ of submodular functions. In contrast, when using the facility location function, we maximize the average case over the same set of functions. Unfortunately, the minimization over a set of submodular functions does not preserve submodularity so we cannot use the same fast algorithms while being afforded the same mathematical guarantee. Hence, at least until the space of submodular functions for the problem of finding sketches of scRNA-seq measurements has been fully investigated, it seems that attempting to maximize the worst case may be premature. Our results above, showing good results using a simple feature-based objective, support this suggestion.