The Journal of Open Source Software

JavaPermutationTools: A Java Library of Permutation Distance Metrics

Vincent A. Cicirello[1]

1 Computer Science, School of Business, Stockton University, Galloway, NJ 08205

DOI: 10.21105/joss.00950

Software

- Review
- Repository
- Archive

Submitted: 08 August 2018
Published: 06 November 2018

Summary

Permutations can represent a wide variety of ordered data. For example, a permutation may represent an individual's preferences (or ranking) of a collection of products such as books or music. Or perhaps a permutation may represent a route for delivering a set of packages. Permutations can also represent one-to-one mappings between sets (e.g., instructors to courses at a fixed time). There are applications where measuring the distance between a pair of permutations is necessary. For example, a recommender system may assess the similarity of two individuals' preferences for music to make song recommendations. Depending upon the application, the permutation features most important to distance calculation may be the absolute positions of the elements (e.g., one-to-one mappings), the adjacency of elements (e.g., the routing example), or general precedence of pairs of elements (e.g., music preferences). Thus, it is no surprise that there are many permutation metrics in the research literature. Knuth's seminal books on algorithms (Knuth, 1997, 1998a, 1998b) cover permutation related algorithms more generally such as mixed radix representation, permutation inverse computation, etc.

The motivation and origin of this library is our research on fitness landscape analysis for permutation optimization (Cicirello, 2014, 2016, 2018a; Cicirello & Cernera, 2013). In a permutation optimization problem, solutions are represented by permutations of some set, and the objective is to maximize or minimize some function. For example, a solution to a traveling salesperson problem is the permutation of the set of cities that corresponds to the minimal cost tour. During our research, we developed a Java library of permutation distance metrics. Most of the distance metrics in the literature are described mathematically with no source code available. Thus, our library offers convenient access to efficient implementations of a variety of metrics with a common programmatic interface. The library also provides metrics on sequences (strings and arrays of various types); where unlike a permutation, a sequence may contain multiple copies of the same element.

The source repository (https://github.com/cicirello/JavaPermutationTools) contains source code of the library, programs that provide example usage of key functionality, as well as programs that reproduce results from papers that have used the library. API documentation is hosted on the web (https://jpt.cicirello.org/).

Statement of Need

The target audience of this library are those conducting computational research where the similarity of permutations or sequences must be assessed, or for which other computation on permutations is required (e.g., includes functionality for generating and mutating permutations in various ways). Permutation distance is important to those developing

recommender systems, and also important to those applying evolutionary computation to the solution of permutation optimization problems.

Evolutionary computation, such as genetic algorithms, solve problems through simulated evolution (Mitchell, 1998). They maintain a population of solutions to the problem, and this population evolves over many generations using operators such as mutation and crossover. Just as in natural evolution, a diverse gene pool is important. In later generations, if variation within the population declines, then search can stagnate. Population management (Sevaux & Sörensen, 2005), such as in scatter search (Campos, Laguna, & Martí, 2005), attempts to maintain population diversity, requiring a measure of distance.

In search landscape analysis, one must often compute the distance between points on the landscape. A fitness (or search) landscape (Mitchell, 1998) is the space of possible solutions to an optimization problem spatially organized on a landscape with similar solutions as neighbors, and where elevation corresponds to fitness (or solution quality). Peaks (maximization problems) and valleys (minimization problems) correspond to locally optimal solutions. The problem is to find an optimal point on that landscape. Search landscape analysis deals with the theoretical and practical techniques for studying what characteristics of a problem make it hard, how different search operators affect fitness landscape topology, among others. There has been much work on fitness landscape analysis, including for permutation landscapes (Cicirello, 2014, 2016, 2018a; Cicirello & Cernera, 2013; Hernando, Mendiburu, & Lozano, 2016; Schiavinotto & Stützle, 2007; Sörensen, 2007; Tayarani-N & Prugel-Bennett, 2014). Fitness landscape analysis techniques, such as fitness distance correlation (FDC) (Jones & Forrest, 1995) and search landscape calculus (Cicirello, 2016) require distance metrics for the type of structure you are optimizing.

The Metrics of the Library

The following table summarizes the permutation distances in the library, their runtimes (n is permutation length), and whether they satisfy the metric requirements.

Distance	Runtime	Metric?	Citations
acyclic edge distance	$O(n)$	pseudo	(Ronald, 1995, 1997)
cyclic edge distance	$O(n)$	pseudo	(Ronald, 1995, 1997)
cyclic r-type distance	$O(n)$	pseudo	(Cicirello, 2016)
deviation distance	$O(n)$	yes	(Campos et al., 2005; Cicirello, 2016)
deviation distance normalized	$O(n)$	yes	(Ronald, 1998; Sörensen, 2007)
edit distance	$O(n^2)$	yes	(Sörensen, 2007; Wagner & Fischer, 1974)
exact match distance	$O(n)$	yes	(Ronald, 1998)
interchange distance	$O(n)$	yes	(Cicirello & Cernera, 2013)
Kendall tau distance	$O(n \lg n)$	yes	(Fagin, Kumar, & Sivakumar, 2003; Kendall, 1938; Meilă & Bao, 2010)
Lee distance	$O(n)$	yes	(Lee, 1958)
r-type distance	$O(n)$	yes	(Campos et al., 2005; Martí, Laguna, & Campos, 2005)
reinsertion distance	$O(n \lg n)$	yes	(Cicirello, 2016; Cicirello & Cernera, 2013)
reversal distance	Init: $O(n!n^3)$ Calc: $O(n^2)$	yes	(Caprara, 1997; Cicirello, 2016)
squared deviation distance	$O(n)$	yes	(Sevaux & Sörensen, 2005)

The next table summarizes the metrics on sequences ($n \leq m$ are the lengths of the compared sequences).

Distance	Runtime	Metric?	Citations
edit distance	$O(n * m)$	yes	(Wagner & Fischer, 1974)
exact match distance	$O(n)$	yes	(Ronald, 1998)
Kendall tau sequence distance	$O(n \lg n)$	yes	(Cicirello, 2018b; Kendall, 1938)
longest common subsequence distance	$O(n * m)$	yes	(Wagner & Fischer, 1974)

References

Campos, V., Laguna, M., & Martí, R. (2005). Context-independent scatter and tabu search for permutation problems. *INFORMS Journal on Computing*, *17*(1), 111–122. doi:10.1287/ijoc.1030.0057

Caprara, A. (1997). Sorting by reversals is difficult. In *Proceedings of the first annual international conference on computational molecular biology* (pp. 75–83). ACM.

Cicirello, V. A. (2014). On the effects of window-limits on the distance profiles of permutation neighborhood operators. In *Proceedings of the international conference on bioinspired information and communications technologies* (pp. 28–35). doi:10.4108/icst.bict.2014.257872

Cicirello, V. A. (2016). The permutation in a haystack problem and the calculus of search landscapes. *IEEE Transactions on Evolutionary Computation*, *20*(3), 434–446. doi:10.1109/TEVC.2015.2477284

Cicirello, V. A. (2018a). *Classification of permutation distance metrics for fitness landscape analysis* (preprint). Stockton University.

Cicirello, V. A. (2018b). *Kendall tau sequence distance: Extending kendall tau from ranks to sequences* (preprint). Stockton University.

Cicirello, V. A., & Cernera, R. (2013). Profiling the distance characteristics of mutation operators for permutation-based genetic algorithms. In *Proceedings of the 26th international conference of the florida artificial intelligence research society* (pp. 46–51). AAAI Press.

Fagin, R., Kumar, R., & Sivakumar, D. (2003). Comparing top k lists. *SIAM Journal on Discrete Mathematics*, *17*(1), 134–160.

Hernando, L., Mendiburu, A., & Lozano, J. A. (2016). A tunable generator of instances of permutation-based combinatorial optimization problems. *IEEE Transactions on Evolutionary Computation*, *20*(2), 165–179. doi:10.1109/TEVC.2015.2433680

Jones, T., & Forrest, S. (1995). Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In *Proceedings of the 6th international conference on genetic algorithms* (pp. 184–192). Morgan Kaufmann.

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, *30*(1/2), 81–93.

Knuth, D. E. (1997). *The art of computer programming, volume 1, fundamental algorithms* (3rd ed.). Addison Wesley.

Knuth, D. E. (1998a). *The art of computer programming, volume 2, seminumerical algorithms* (3rd ed.). Addison Wesley.

Knuth, D. E. (1998b). *The art of computer programming, volume 3, sorting and searching* (2nd ed.). Addison Wesley.

Lee, C. (1958). Some properties of nonbinary error-correcting codes. *IRE Transactions on Information Theory*, *4*(2), 77–82. doi:10.1109/TIT.1958.1057446

Martí, R., Laguna, M., & Campos, V. (2005). Scatter search vs. Genetic algorithms: An experimental evaluation with permutation problems. In *Metaheuristic optimization via memory and evolution* (pp. 263–282). Springer.

Meilă, M., & Bao, L. (2010). An exponential model for infinite rankings. *Journal of Machine Learning Research*, *11*, 3481–3518.

Mitchell, M. (1998). *An introduction to genetic algorithms*. MIT Press.

Ronald, S. (1995). Finding multiple solutions with an evolutionary algorithm. In *Proceedings of the ieee congress on evolutionary computation* (pp. 641–646). IEEE Press.

Ronald, S. (1997). Distance functions for order-based encodings. In *Proceedings of the ieee congress on evolutionary computation* (pp. 49–54). IEEE Press.

Ronald, S. (1998). More distance functions for order-based encodings. In *Proceedings of the ieee congress on evolutionary computation* (pp. 558–563). IEEE Press.

Schiavinotto, T., & Stützle, T. (2007). A review of metrics on permutations for search landscape analysis. *Computers & Operations Research*, *34*(10), 3143–3153. doi:10.1016/j.cor.2005.11.022

Sevaux, M., & Sörensen, K. (2005). Permutation distance measures for memetic algorithms with population management. In *Proceedings of the metaheuristics international conference (mic2005)* (pp. 832–838).

Sörensen, K. (2007). Distance measures based on the edit distance for permutation-type representations. *Journal of Heuristics*, *13*(1), 35–47. doi:10.1007/s10732-006-9001-3

Tayarani-N, M.-H., & Prugel-Bennett, A. (2014). On the landscape of combinatorial optimization problems. *IEEE Transactions on Evolutionary Computation*, *18*(3), 420–434. doi:10.1109/TEVC.2013.2281502

Wagner, R. A., & Fischer, M. J. (1974). The string-to-string correction problem. *Journal of the ACM*, *21*(1), 168–173.