lsh, Nearest neighbor search in high dimensions

Edwin de Jonge[1,⋆]

1. Statistics Netherlands (CBS)
⋆Contact author: e.dejonge@cbs.nl

Keywords: Machine learning, Locality Sensitive Hashing, high dimensional nearest neighbor

Data sets with many variables and rows are very common nowadays. The number of dimensions p can run from ten to thousands of variables. The number of observations n typically runs from thousands to millions. Both a large n and p are challenging for traditional nearest neighbor techniques.

Calculating distance pairs is $O(n^2)$ in memory and time and finding the nearest neighbor is $O(n)$ in time. Tree indexing techniques like kd-tree [2] were developed to cope with large n, however their performance quickly breaks down for $p > 3$ [3]. Locality sensitive hashing (LSH) [3] is a technique for generating hash numbers from high dimensional data, such that nearby points have identical hashes. This enables efficient nearest neighbor search for (very) high dimensional data sets. It has been successfully applied to several problems including text similarity search [5].

R package **lsh** [4] (in development) is an implementation of locality sensitive hashing in *R*. We will describe the implemented locality sensitive hashing technique, the several distance functions and the functionality of **lsh**. Suggestions for further tuning performance will be provided.

References

[1] Andoni, A. and P. Indyk (2006). Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*, pp. 459–468. IEEE.

[2] Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM 18*(9), 509–517.

[3] Datar, M., N. Immorlica, P. Indyk, and V. S. Mirrokni (2004). Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry*, pp. 253–262. ACM.

[4] de Jonge, E. (2014). lsh, locality sensitive hashing in r. `http://github.com/edwindj/lsh`.

[5] Slaney, M. and M. Casey (2008). Locality-sensitive hashing for finding nearest neighbors [lecture notes]. *Signal Processing Magazine, IEEE 25*(2), 128–131.