An Investigation of Approximate Nearest Neighbor Techniques

Emil Azadian, Romain Camilleri, Andrew Wagenmaker

Overview

In our project we investigate the performance of Locality Sensitive Hashing (LSH) on data drawn from a variety of distributions. As LSH is agnostic to the distribution of the data, we conjecture that it will perform differently on certain distributions. We test this hypothesis on a variety of synthetic and real world experiments and compare against several approaches that do take into account the distribution of the data—attempting to rigorously quantify on the types of distributions and real world problems where LSH performs poorly compared to approaches that take into account the data distribution.

ANN Methods

Euclidean Locality Sensitive Hashing (LSH) [1]

Euclidean LSH functions by randomly generating hyperplanes and then hashing data points $\mathbf{x}$ to buckets by projecting them onto the hyperplanes as:

$$\left\lfloor \frac{\mathbf{a}^\top \mathbf{x} + b}{r} \right\rfloor$$

By varying the number of hyperplanes and how the hash values are grouped together, the accuracy and speed can be traded off.

Spectral Hashing (SH) [2]

Spectral Hashing, first data-dependent approximate nearest neighbor search algorithm, operates by first finding the principal components of the data using PCA and then thresholding the eigenvalues to obtain a binary code.

Product Quantization (PQ) [3]

Product quantization is an efficient and data-dependent approach to approximate nearest neighbors search. If functions by breaking each data vector into M subvectors. It then performs k-means clustering of the training data in each of these M spaces to generate codewords for the data. When a data point is queried, it is encoded based on its proximity to the centers of the k-means step and then the training data is queried using this encoded representation.

Results on Synthetic Data

Our conjecture states that Locality Sensitive Hashing (LSH) does not use the underlying distribution of the dataset, whereas Spectral Hashing (SH) and Product Quantization (PQ) do. We vary the variance of the distribution that generates the datasets on which we benchmark the three methods. Focusing first on LSH, figure 1 illustrates how different the mapping of our data among buckets will be when the distribution of the data is ignored.

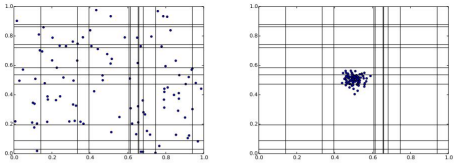
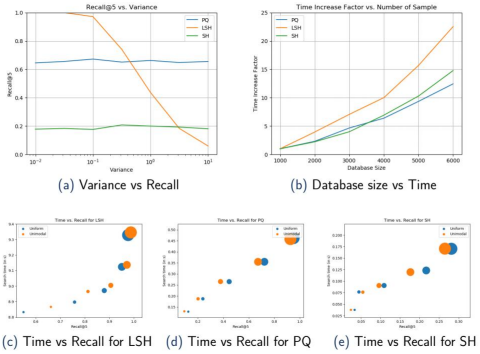


Figure: Illustrations of the hashing process of 2-dimensional datapoints that are drawn from a uniform distribution (left) and a normal distribution (right).

We observe in Figure 2a that an increase in the variance of our data leads to significantly worse results for LSH, while both PQ and SH are very consistent across different variances. Also, referring to figure 2b, LSH pays a lot for maintaining its accuracy: The runtime increases 23-fold from n=1000 to n=6000, whereas PQ and SH both only see an increase by 10-15. Finally, we observe in Figures 2c, 2d and 2e that when the memory allocated to LSH is lower, both its recall and its search time highly depend on the distribution type.



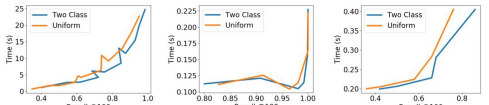
(a) Variance vs Recall (b) Database size vs Time

(c) Time vs Recall for LSH (d) Time vs Recall for PQ (e) Time vs Recall for SH

Figure: Performance of ANN Methods on Synthetic Datasets

Results on Real Data

In order to validate our hypothesis on real world problems, we tested all three methods on a subset of ImageNet and a corpus of articles from Wikipedia. To simulate different distributions, we generate two ImageNet datasets—one of images selected uniformly at random from the dataset, and one of images that are either of cars or berries—and two Wikipedia datasets—one of random articles, and one of articles either about football or computers. We hypothesize that the way we select the data will yield uniform and bimodal distributions, respectively.



(a) Euclidean LSH (b) Product Quantization (c) Spectral Hashing

Figure: Performance of ANN Methods on Wikipedia Data

(a) Euclidean LSH (b) Product Quantization (c) Spectral Hashing

Figure: Performance of ANN Methods on ImageNet Data

These figures indicate that the performance of LSH is impacted by the distribution on certain real world tasks, as we conjectured, and that other approaches are able to take these distributions into account much more effectively.

References

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

[2] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Advances in neural information processing systems*, pp. 1753–1760, 2009.

[3] H. Jegou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 1, pp. 117–128, 2011.