Beyond Hit-or-Miss: A Comparative Study of Synopses for Similarity Searching

Marcos V. N. Bedo[1], Daniel de Oliveira[2], Agma J. M. Traina[3], Caetano Traina Jr.[3]

[1] Fluminense Federal University, St. A. Pádua, Rio de Janeiro, Brazil
marcosbedo@id.uff.br
[2] Fluminense Federal University, Niterói, Rio de Janeiro, Brazil
danielcmo@ic.uff.br
[3] University of São Paulo, São Carlos, São Paulo, Brazil
{agma, caetano}@icmc.usp.br

Abstract. A DBMS optimizer module takes its decisions by modeling the query costs upon the distribution of the data space. Cost modeling of similarity queries, however, requires the representation of distances' rather than data distributions. Therefore, the finding of a suitable representation (or *synopsis*) for the distance distribution has a major impact in the optimization of similarity searches. In this study, we evaluate the quality of estimates drawn from five synopses of distinct paradigms regarding two common query criteria. Moreover, we embed the synopses into a new parametric cost model, called `Stockpile`, for the cost estimation of similarity queries on metric trees. The model uses the synopses estimation for calculating the probability of traversing a metric tree node, which defines the expected number of both disk accesses (I/O costs) and distance calculations (CPU costs). We performed an extensive set of experiments on real-world data sources regarding the estimates of each synopsis (and its parametric variations) by using paired ranking tests. In global terms, three synopses have outperformed their competitors regarding selectivity estimation, whereas two of them have also surpassed the others in the prediction of both I/O and CPU costs with respect to `Stockpile` model predictions. Additionally, results also indicate the choice of the most suitable synopsis may depend on characteristics of the distance distribution.

Categories and Subject Descriptors: H.2.1 [**Database Management**]: Logical Design; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords: Similarity searching, cost model, synopsis, distance distribution

1. INTRODUCTION

Similarity searching is a core paradigm for many modern computer applications, such as content-based retrieval, classification, clustering, and data visualization [Zezula et al. 2010]. Commonly, two of the most requested similarity operations are the range and neighborhood searches. An example of range query is **(Q1)** "List the bottles in the wine cellar whose combination of fixed and volatile acidities differs at most 4 mg/L to this Italian wine", while a neighborhood query example is **(Q2)** "Find the 3 closest cabs to this restaurant". Both range and neighborhood queries can be modeled upon a *metric space*, where the elements (bottles and cabs, in the aforementioned examples) are represented as points and the (dis)similarity between each pair of points is evaluated by a distance function.

Formally, a metric space is a pair $\mathcal{M} = \langle \mathbb{S}, \delta \rangle$, where $\mathbb{S}$ is the domain of the points and δ is a distance function that complies with the properties of symmetry, non-negativity, and triangular inequality [Hetland 2009]. Accordingly, given a data source $\mathcal{S} \subseteq \mathbb{S}$, a query element $s_q \in \mathcal{S}$ and a threshold $\xi \in \mathbb{R}_+$, a range query Rq retrieves all elements in $\mathcal{S}$ within the closed ball centered at s_q

with radius ξ such that $Rq(\mathcal{S}, s_q, \xi) = \{s_i \in S \mid \delta(s_i, s_q) \leqslant \xi\}$. Range queries can be seen as a subset of relational-based queries in which range is constrained by multiple attributes. On the other hand, a neighborhood (k-NN) query returns a quantity $k \in \mathbb{N}$ of elements whose distance to the query element s_q are the smallest. In an equivalent way, a k-NN query can be seen as a variation of the range query, *i.e.*, a range search with a defined radius ξ such that $|Rq| = k$ [Tasan and Ozsoyoglu 2004].

Although sequential search can be used as the standard access method for solving both range and neighborhood queries, several indexing strategies (in the format of metric access methods) have been proposed to speed up similarity searching [Chen et al. 2017]. In particular, *tree-based* methods stand out as the most suitable strategies for the indexing of very large data sources as they organize the search space into a hierarchical and balanced fashion. Tree-based structures, such as M-Tree and Slim-Tree [Zezula et al. 2010], focus on minimizing both distance calculations (through the clustering of the elements) and disk accesses (by using paging principles). In practice, the performance of these indexes also depends on the query conditions. For instance, Slim-Tree can perform better than M-Tree for a given query element in a k-NN search, whereas M-Tree can outperform Slim-Tree regarding other query elements for the same search criterion. Therefore, given a similarity query and one or more indexes, a database query optimizer must decide which access method will be employed to execute the search. Such a query optimizer' decision is made upon a representation (or *synopsis*) of the data source distance distribution that meets the query conditions [Ioannidis 2003; Zezula et al. 2010].

Many challenges arise from handling distance distributions as they are related to both data sources and distance functions. For instance, several distributions can be built for the same data source if it is compared by different distance functions [Cha 2007]. Moreover, distance distributions can be gathered regarding two distinct semantics, *i.e.*, as the (pairwise) distances between every pair of elements within the data source [Pestov 2012] or as the distances of every element to a given viewpoint (pivot) [Ciaccia et al. 1999]. Synopses are short representations of such distributions and, therefore, fall under the categories of *pairwise synopses* or *pivot-based synopses*. A manifold of synopses designed for specific problems of similarity searching optimization is found in the literature, more concrete goals being selectivity, radius, I/O costs, and CPU costs estimation [Aly et al. 2015]. Although all these problems certainly depend on the handling of distance distributions, to the best of our knowledge no comparison between the synopses predictions has been conducted so far [Cormode et al. 2012; Chen et al. 2015].

In fact, most of the research effort has addressed selectivity estimation for range queries [Tao et al. 2004; Aly et al. 2015] regarding multidimensional spaces. Such approaches are mainly based on *biasing* the distance distribution, such as the uniform assumption, the fractal behavior, or the standard distribution assumption [Clauset et al. 2009]. Another group of studies has tackled the problem of optimizing the execution of k-NN algorithms by using synopses estimates [Tasan and Ozsoyoglu 2004; Vieira et al. 2007]. The idea is to reduce k-NN to range queries by using estimated coverage radii so that the k-NN queries can be efficiently solved by a branch-and-bound algorithm. The synopses designed for optimizing k-NN queries are mainly *histograms* and most of them also follow a *biased assumption*, in which query elements are more likely posed in high-density areas of the search space [Zezula et al. 2010]. A few models have been proposed for the estimation of both I/O and CPU costs regarding similarity searching. The landmark model in Ciaccia et al. [1998] proposes using histograms on pairwise distance distributions for the evaluation of the probability of traversing a metric-tree node, whereas the models in Tao et al. [2004] use pairwise histograms for cost estimation on spatial indexes. The main drawback of such models is they disregard the 'locality' of each query, *i.e.*, they rely on a single synopsis (derived from the pairwise distribution) without properly considering the query element. The study in Tasan and Ozsoyoglu [2004] discusses I/O and CPU costs from a broader perspective by defining a parametric probabilistic model for emulating the behavior of metric indexes, in which the probability density function (*p.d.f.*) can be biased towards either a known distribution assumption or multiple histogram-based representations.

In this study, we aim at evaluating the quality of estimates drawn from synopses with distinct biases regarding both range and k-NN queries. To do so, we extended our previous approach in Bedo et al. [2017] by designing a parametric version of `Stockpile` model, which complies with both theoretical and experimental indications of reviewed models in such a way any synopsis-based p.d.f. can be coupled to the new approach. `Stockpile` distinguishes itself from related models as it enables the cost estimation of range and k-NN queries by using multiple pivot-based synopses. The overall idea is a small set of pivot-based synopses are enough for the representation of density around a given query element. Our model also supports the use of synopses from pairwise distance distribution as probability density functions, but it is unable to estimate the local densities with such a setup. Last, but not least, `Stockpile` cost prediction of k-NN queries is conducted by using radii estimates, which are parameterized towards the synopsis-based probability density function. We experiment on real-world data sources and compare five synopses of distinct paradigms in terms of their precision for selectivity and radii estimation, as well their quality for I/O and CPU costs prediction by using paired ranking tests. Accordingly, the contributions of this manuscript are as follows:

(1) We revisited discrete-valued and continuous synopses for pairwise/pivot-based distance distributions and discuss their implementation,

(2) Discrete-valued synopses outperformed continuous representation in the task of providing both selectivity and radii estimation regarding the `Stockpile` cost model estimation rules, and

(3) Three synopses (`CDH-PAIR`, `CDH-LINEAR`, and `V-OPT HIST`) showed the best performance for I/O cost estimation, whereas two synopses (`CDH-PAIR` and `CDH-LINEAR`) presented the best results for CPU cost estimation.

The remainder of the article is organized as follows. Section 2 summarizes related work. Section 3 introduces Stockpile and its parameters. Sections 4, 4.1, 4.2, and 4.3 show the results of experimental evaluations, while Section 5 concludes the article.

2. PRELIMINARIES AND RELATED WORK

Distance distributions can be obtained from pairwise or pivot-based measurements. The pairwise distance distribution (Definition 2.1) represents the frequencies of the distances between every pair of elements from the data source. On the other hand, a pivot-based distance distribution (Definition 2.2) represents the frequencies of the distances from all elements within the data source to a given pivot element. Next, we present synopses for both distributions and discuss selectivity and radii estimation strategies. Table I summarizes the notation employed in the remaining of the manuscript.

Definition 2.1 *Pairwise Distance Distribution* – $\mathcal{T}$. Given a data source $\mathcal{S}$ and a distance function δ, $\mathcal{T}$ captures the distances from every $s_i, s_h \in \mathcal{S}$. Distance value set contains the distinct and sorted values of distances $\delta(s_i, s_h)$ between pairs of elements s_i, s_h and is given by $\mathcal{V} = \{v(j) : 1 \leqslant j \leqslant m\}$, where $m \leqslant |\mathcal{S}|^2$. Frequency $f_{req}(j)$ of $v(j)$ is the number of distances in which $\delta(s_i, s_h) = v(j)$ and joint frequency $c(j)$ of $v(j)$ is the number of distances $\delta(s_i, s_h) \leqslant v(j)$. Therefore, $\mathcal{T} = \{\langle v(1), f_{req}(1)\rangle, \ldots, \langle v(m), f_{req}(m)\rangle\}$, where $v(m) \in \mathcal{V}$ is the largest distance between any pair of elements $s_i, s_h \in S$. The joint pairwise distribution is given by $\mathcal{T}^{\mathcal{C}} = \{\langle v(1), c(1)\rangle, \ldots, \langle v(m), c(m)\rangle\}$. $\mathcal{T}^{+}$ is the $\mathcal{T}$ extension by setting 0 as the frequency to any $v(j) \in \mathbb{R}_{+} \backslash \mathcal{V}$.

Definition 2.2 *Pivot-based distance distribution* – $\mathcal{T}_p$. Given a data source $\mathcal{S}$, a distance function δ, and a pivot $p \in \mathcal{P}$, $\mathcal{T}_p$ captures the distance from each $s_i \in \mathcal{S}$ to p. Distance value set $\mathcal{V}_p$ contains the distinct and sorted values of $\delta(s_i,\ p)$, *i.e.*, $\mathcal{V}_p = \{v_p(j) : 1 \leqslant j < m_p,\ m_p \leqslant |\mathcal{S}|\}$, where $v_p(m_p)$ is the largest distance between any s_i to p. Frequency $f_{req_p}(j)$ is the number of elements of $\mathcal{S}$ whose distance $\delta(s_i,\ p) = v_p(j)$. Accordingly, $\mathcal{T}_p = \{\langle v_p(1), f_{req_p}(1)\rangle, \ldots, \langle v_p(m_p), f_{req_p}(m_p)\rangle\}$. $\mathcal{T}_p^{+}$ is the $\mathcal{T}_p$ extension by setting 0 as the frequency to any $v_p(j) \in \mathbb{R}_{+} \backslash \mathcal{V}_p$.

Table I. Summary of symbols.

Symbol	Definition	Symbol	Definition
$\mathbb{S}$	Data domain	$\mathcal{S} \subseteq \mathbb{S}$	A given data source
k	Number of retrieved neighbors	ξ	Radius of a range query
$\mathcal{P} \subset \mathbb{S}$	A set of pivots	$p \in \mathcal{P}$	A selected pivot
$\mathcal{T}_p$	A pivot-based distribution	$\mathcal{T}$	The pairwise distribution
$\phi_p(x)$	Linear function of a CDH	CDH	A Compact-Distance Histogram
$f(x)$	Any synopsis p.d.f.	$\mathcal{D}$	Distance exponent
$\bar{F}(x)$	A normalized $F(x)$	$F(x)$	A cumulative p.d.f.
$\lvert\mathtt{N}_j\rvert$	Number of entries in a node	$\mathtt{M}$	A metric tree

2.1 Representation of distance distributions

Techniques for representing distance distributions can be grouped into two classes, namely continuous-valued and discrete-valued synopses.

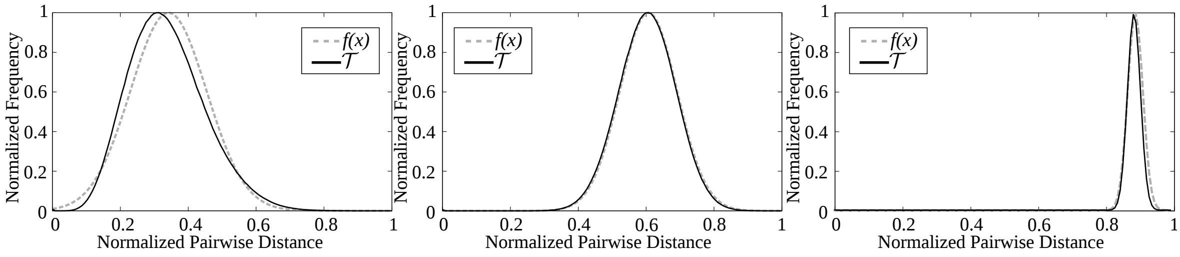
(1) Continuous-valued synopses – Most of the reviewed approaches rely on known distributions, as Uniform, Fractal, or Standard distributions [Cormode et al. 2012]. Such functions are particularly suitable when certain distance conditions within the data source are met.

(2) Discrete-valued synopses – These synopses are usually built as histograms [Ioannidis 2003]. Histograms provide more details of the distribution in comparison to continuous-valued synopses, as they enable the partitioning of the values and/or frequencies according to a number of parameters. Such parameters can be set towards an available budget of memory.

Continuous-valued synopses

A Normal synopsis $N(\mu(\mathcal{T}), \sigma(\mathcal{T}))$ provides a very short representation of the original distance distribution by requiring the storage of only two parameters: mean $\mu(\mathcal{T})$ and standard deviation $\sigma(\mathcal{T})$. Normality hypothesis tests, such as the Kolmogorov–Smirnov test [Clauset et al. 2009], can be used to determine whether or not $\mathcal{T}$ is well approximated by $N(\mu(\mathcal{T}), \sigma(\mathcal{T}))$. Such a synopsis is appropriate for data sources embedded in high dimensional spaces, where distances tend to concentrate by the ratio $\sigma(\mathcal{T})/\mu(\mathcal{T})$ whenever L_p functions are used for measuring distances [Pestov 2012]. Figure 1 presents the pairwise distribution and Normal approximation for a synthetic data source of $1,000$ elements and increasing number of independent and identically distributed (i.i.d.) dimensions in the $[0, 1]$ interval. Equation 1 presents the p.d.f. for the Normal synopsis for any distance x to be evaluated.

$$f(x) = \frac{e^{-\frac{(x-\mu(\mathcal{T}))^2}{2\sigma(\mathcal{T})^2}}}{\sqrt{2\pi}\sigma(\mathcal{T})}, \tag{1}$$



(a) d = 2 (b) d = 32 (c) d = 1500

Fig. 1. Pairwise distance distributions of a synthetic data source with i.i.d. attributes and their approximation by normal standard distributions for increasing number of dimensions.

On the other hand, the fractal assumption represents pairwise distributions within data sources with *self-similarity* properties, *i.e.*, the distribution within small portions are similar to the distribution of the whole data according to an exponent scaling. Although several approaches can be used to determine the fractal dimension [Clauset et al. 2009], a relevant approach is the approximation of the joint pairwise distribution by the distance exponent [Vieira et al. 2007]. Accordingly, if the joint pairwise distribution of a self-similar data source is scaled to a log-log plot, then its joint frequencies can be approximated by a linear p.d.f. $f(x)$. In other words, given a data source $\mathcal{S}$ and its log-log representation of $\mathcal{T}$, $f(x)$ is obtained by a linear method, e.g., least-squares[1], as in Equation 2.

$$f(x) = c' x^{\mathcal{D}}, \tag{2}$$

where c' is a constant of proportionality and slope $\mathcal{D}$ is the distance exponent that approximates the distribution's fractal dimension. Figure 2(a) shows both log-log plot and $f(x)$ for the joint pairwise distance distribution of a synthetic data source with $1,000$ elements that exhibit the self-similarity behavior. A useful synopsis variation is presented in Figure 2(b), in which $f(x)$ is shifted for the interpolation of Point 0 and provide radii estimation (see Section 2.2).

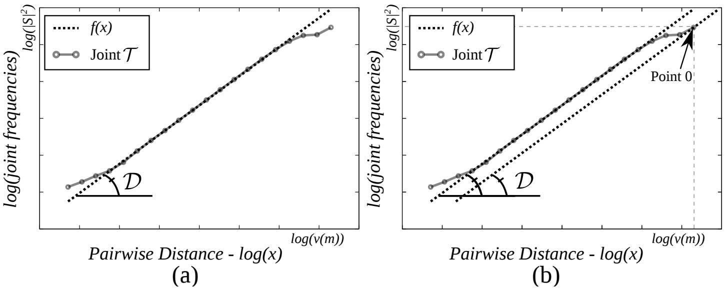


Fig. 2. Joint pairwise distance distribution of a synthetic data source with self-similar behavior approximated by $f(x)$. (a) Least-squares approximation. (b) Shifted $f(x)$ for the interpolation of Point 0.

Discrete-valued synopses

Histograms are discrete alternatives to continuous-valued synopses. The idea is both $\mathcal{T}$ and $\mathcal{T}_p$ can be partitioned according to the maximum number of buckets that satisfy the DBMS optimizer memory budget [Shekelyan et al. 2017]. Essentially, histograms differ from each other by their *partition constraints* that define how buckets are formed. For instance, classic equi-histograms target the summarization of equal parameters, *e.g.*, the constraint on an equal interval of distance values generate Equi-Width histograms, while the restriction on an equal interval of frequencies induces Equi-Depth histograms. Also, objective functions to be optimized can play the role of histogram partitioning constraints. For instance, the minimization of the variance of frequencies for each bucket of a histogram generates V-Optimal histograms, which are still state-of-the-art synopses of minimal error for query optimizers that handle fixed-memory budgets [Ioannidis 2003; Cormode et al. 2012]. Figure 3(b) presents an example of a V-Optimal limited by a fixed-memory budget of six buckets on the pivot-based distance distribution of Figure 3(a).

Aimed at avoiding the uniform distribution of frequencies, a Compact-Distance Histogram [Bedo et al. 2018] represents $f(x)$ as a piecewise linear function. Therefore, the frequency within a bucket b_i is calculated according to a linear function $\phi(x) = \alpha_{b_i} \cdot x + \beta_{b_i}$. The partition constraint of a CDH is defined in such a way the absolute differences between the frequencies of $\phi(x)$ and that of the distance distribution is minimal. As a result, CDH enables the representation of the original distribution as both discrete-valued and continuous-valued functions. Notice, p.d.f.'s can be straightforwardly drawn from

[1] Although the natural logarithm is often employed, the fitting method does not depend on the logarithm base.

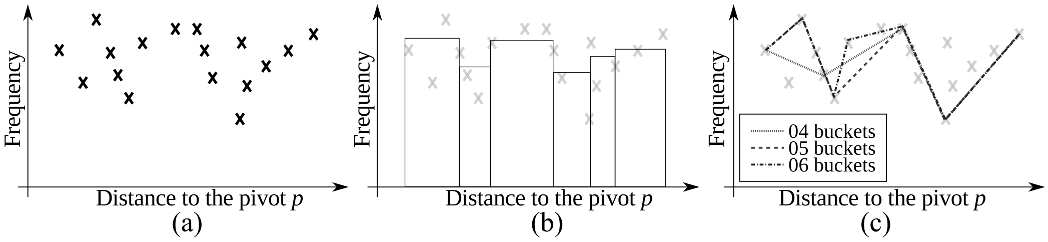


Fig. 3. Histogram partitioning of a pivot-based distance histogram. (a) Original distance distribution. (b) V-Optimal histogram limited by 06 buckets. (c) CDH histograms limited by 04, 05, and 06 buckets.

histograms as the frequency of the bucket that contains distance value x. In particular, $f(x) = \phi(x)$ for the bucket b_i that includes distance x. CDH construction follows the interactive routine of V-Optimal histograms, as the approximation error decreases in terms of the number of buckets. However, the CDH approximation error tends to be smaller than in comparison to V-Optimal for the same number of buckets. Figure 3(c) presents CDH's for the distribution $\mathcal{T}_p$ in Figure 3(a).

2.2 Cost models for metric trees

Metric trees are indexing strategies that rely on balanced data source partitioning [Zezula et al. 2010]. The basic methods of this category, M-Trees, Slim-Trees, and PM-Trees, are dynamic and overlapping structures that hierarchically organize the elements within metric spaces into closed balls, which are stored as *nodes* and noted `N` [Traina et al. 2002]. Basically, M-Trees use two types of nodes, namely directory and leaf nodes. Directory nodes store a set of balls, whereas leaf nodes store the indexed elements themselves. Accordingly, a leaf node includes a set of elements and their distances to the parent of the leaf node. A directory node includes rooted subtrees, the covering radius, and the distance of rooting elements to the parents. Figure 4(a–b) shows examples of M-Tree and Slim-Tree.

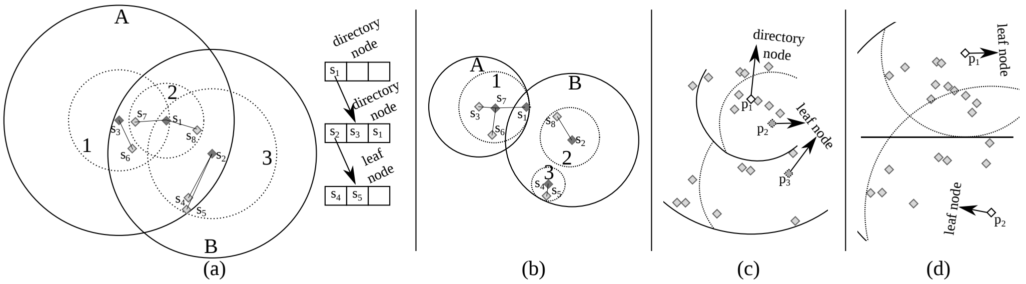


Fig. 4. Examples of metric tree structures by using the L_2 distance. (a) Overlapping M-Tree. (b) Slim-Tree with minimized node overlapping. (c) Two-level binary VP-Tree. (d) Leaves of a binary MDF-Tree.

Analogously, static metric trees, as VP-Trees and MDF-Trees, rely on the balanced and disjoint data source partitioning. *Directory nodes* of static trees include a pivot element, a covering radius, and two pointers to the subtrees. *Leaves* include a pivot, both minimum $\xi^{\downarrow}$ and maximum $\xi^{\uparrow}$ coverage radii, and data elements. For VP-Trees, the minimum radius of *left* leaf nodes is zero, while the maximum is the median of distances between the elements and the pivot. Likewise, the minimum radius of *right* leaf nodes is the median, whereas the maximum is the longest distance between leaf elements and the pivot. In MDF-Trees, the minimum radius is zero, while the maximum is the largest distance between leaf elements and the pivot. Figure 4(c–d) shows VP-Tree and MDF-Tree partitioning examples.

Both dynamic and static metric trees rely on the *ball partitioning* principle, where each node can be seen as a closed ball with a rooting pivot and coverage radii. Therefore, parametric estimations can be drawn from any metric tree just by adjusting the interval of distances to the ball coverage radii.

Cost estimation of range queries

A widely employed baseline cost model for metric trees is the approach in [Ciaccia et al. 1998]. Their proposal assumes the cost of range queries can be estimated by a *biased* model, which implies the distances between the query element and the points in $\mathcal{S}$ are supposed to follow a pairwise distance distribution. The distribution itself is approximated by one Equi-Width histogram that provides the frequency approximation within the buckets as the p.d.f. $f(x)$. Therefore, given a range query $Rq(\mathcal{S}, s_q, \xi)$, the probability of scanning a node rooted by s_i with radius ξ_i of a M-Tree-like method `M` that indexes $\mathcal{S}$ is expressed as in Equation 3. The number of scanned nodes in tree `M` is estimated by the sum of the probability of accessing each node as in Equation 4. Likewise, the number of comparisons between s_q and the elements in `M` is the sum of the weighted probabilities of accessing each node, where the weight is the number of entries in the node, as in Equation 5.

$$\texttt{Prob}\{\delta(s_q, s_i) \leqslant \xi + \xi_i\} \approx \int_{\xi_{s_i}^{\downarrow}=0}^{\xi+\xi_{s_i}^{\uparrow}=\xi+\xi_i} f(x)dx \;/ \int_0^{v(m)} f(x)dx = \bar{F}(\xi + \xi_i), \tag{3}$$

$$\texttt{nodes\_scanned}_{Ciaccia}(\texttt{M}, s_q, \xi) \approx \sum_{\texttt{N}_i \in \texttt{M}} \bar{F}(\xi + \xi_i) \tag{4}$$

$$\texttt{distances\_calculated}_{Ciaccia}(\texttt{M}, s_q, \xi) \approx \sum_{\texttt{N}_i \in \texttt{M}} |\texttt{N}_i| \cdot \bar{F}(\xi + \xi_i) \tag{5}$$

Cost estimation of k-NN queries

Cost modeling of k-NN queries requires one additional step in comparison to range searches: the estimation of the query radius. The baseline model in Ciaccia et al. [1998] proposes the use of $\bar{F}(x)$ as part of a binomial probability function so that an estimated radius ξ for a k-NN query can be drawn from $\mathcal{T}$. The drawback of such approach is the resulting p.d.f. is not analytical and depends on an expensive numeric procedure. Alternatively, the approach in Vieira et al. [2007] proposes the use of the distance exponent for radius estimation. Other models also rely on radii estimation by using synopses on the joint pairwise distribution. For instance, the study in Tao et al. [2004] extends the binomial approach of Ciaccia et al. [1998] for a fast k-NN cost estimation in low dimensional spaces, whereas the proposal of Lu et al. [2014] estimates the query costs by combining $\mathcal{S}$ to other domains.

These reviewed models, however, are biased towards the adoption of a single synopsis on the pairwise distribution. The study in Tasan and Ozsoyoglu [2004] comes up with a suggestion for avoiding such a bias in the task of predicting the k-NN radii. Basically, the authors propose using multiple histograms, so that no inference on the type of the distance distribution itself is required. The authors argue if a single synopsis on $\mathcal{T}$ is available, then radius ξ' of a k-NN query can be straightforwardly estimated by the inverse of Equation 6.

$$k = \int_0^{\xi'} f(x)d(x) \tag{6}$$

Nevertheless, the authors also add the *unique* representation of the pairwise distribution can be replaced by synopses on pivot-based distance distributions. Such an observation is similar to the "Homogeneity of Viewpoints" property [Ciaccia et al. 1998; Ciaccia et al. 1999], but authors in Tasan and Ozsoyoglu [2004] suggest keeping the pivot-based distances instead of replacing them by the pairwise distribution. Therefore, given a k-NN query, a set of pivots $\mathcal{P}$, and a set of pivot-based synopses, Equation 6 can be rewritten as Equation 7.

$$k = \sum_{p\ \in\ \mathcal{P}} \texttt{Prob}(p) \cdot \int_0^{\xi'} f_p(x)dx \tag{7}$$

where $\sum_{p\ \in\ \mathcal{P}} \texttt{Prob(p)}$ is the binary probability of pivots p having the same distance distribution of the query element in the neighborhood search, *i.e.*, weighting of individual estimations.

The approach in Tasan and Ozsoyoglu [2004] removes many biases from previous models, but still have practical setbacks. First, predictions drawn from histograms vary according to their *partition constraints* and the best setting for a histogram to represent distance distributions is yet to be found. Second, a suitable p.d.f. must be defined for the calculation of `Prob(p)` as it is not always possible to set $\mathcal{P} = \mathcal{S}$ due to memory constraints. An initial discussion on the finding of a suitable p.d.f. is carried out in the model of Ciaccia et al. [1999], where authors claim expensive empirical assessments can be used for adjusting the parameters of an exponential function to be used as `Prob(p)`.

Alternatives for cost models based on weighted p.d.f.s are the use of pre-computed distance sampling approaches [Cormode et al. 2012]. Although such approaches report good quality cost estimation, they are designed for specific pivot-based indexes so that estimation rules are tightly coupled to either index design or searching algorithms. For instance, the model in Chen et al. [2015] extends the model of Ciaccia et al. [1998] for SPB$^+$-Trees, whereas the method in Aly et al. [2015] assumes neighborhood costs to be *stable*, *i.e.*, costs of executing a k-NN query with larger k is the same of executing a query with a smaller k, provided the incremental searching procedure is employed.

In this manuscript, we pick up from open arguments and indications in Ciaccia et al. [1998] and Tasan and Ozsoyoglu [2004] for the finding of an asymptotic cost model that is isolated and detached of specific indexes' designs. Additionally, we also take into account memory constraints so that continuous and discrete synopses can be seamlessly integrated into our parametric cost model.

3. THE `STOCKPILE` COST MODEL

In this section, we discuss a parametric cost model for predicting selectivity, I/O and CPU costs for any range or k-NN query to be executed by a metric tree. The model itself relies on two previously computed data structures, namely, *(i)* a set of synopses and *(ii)* meta-statistics (*e.g.*, the coverage of nodes) about the metric tree. Such structures are typically kept in main memory as a *pile* of resources (and hence the name `Stockpile` for the model) to be evaluated on-the-fly according to user-posed queries, which requires the model structures to be parameterized by space constraints. Histograms are especially suitable for this scenario, as they enable the constraint on the number of buckets, but continuous-valued synopses can also be used on `Stockpile`, as they scarcely violate space constraints.

If a synopsis on the pairwise distribution is employed for the representation of the p.d.f. $f(x)$, then `Stockpile` estimation disregards the query element and provides global estimates as in the rules of Equations 4, 5, and 6. On the other hand, if pivot-based synopses are employed for representing p.d.f.'s $f_p(x)$, then `Stockpile` also takes into account the query element by assuming *pivots closer to query points are more likely to resemble the distance distributions of the query elements.* Such a premise is particularly fair whenever the density of distances around the pivot is uniformly distributed according to $f_p(x)$. As it is not always the case, `Stockpile` can be set to use a pairwise distance distribution for estimating the relevance of each pivot prediction. Therefore, the `Stockpile` parameters are:

(1) distance distributions, which can set as either pairwise or pivot-based,

(2) synopses on the selected distributions, which must comply with space constraints, and

(3) weighting of pivot-based predictions, which can be obtained as either linear combination or the probability measure drawn from the pairwise distance distribution.

3.1 Weighting of pivot-based predictions

Combining multiple predictions from pivot-based synopsis depends on finding the probability of pivots p having the same distance distribution of the query element of the search. Therefore, if the density is uniformly distributed around a pivot, then the weighting of each pivot contributions can be achieved by using a linear combination of the distances between them to the query element. Formally, let s_q be a query element and $\mathcal{P}$ be a set of pivots, the probability $\texttt{Prob}(p)$ of s_q resembling the distribution $f_p(x)$ is proportional to $\delta(s_q, p)$ so that $\texttt{Prob}(p) = d(s_q, p)/C_2$, where $d(s_q, p) = C_1 - \delta(s_q, p)$. Both values C_1 and C_2 vary according to each query element s_q and are given by $C_1 = \sum_{p \in \mathcal{P}} \delta(s_q, p)$ and $C_2 = \sum_{p \in \mathcal{P}} d(s_q, p)$, respectively. Under this assumption, the joint probability of the query element resembling the pivots in $\mathcal{P}$ is given by Equation 8.

$$\sum_{p \in \mathcal{P}} \texttt{Prob}(p) = \frac{d(s_q, p_1)}{C_2} + \frac{d(s_q, p_2)}{C_2} + \cdots + \frac{d(s_q, p_{|\mathcal{P}|})}{C_2} = \frac{C_1(|\mathcal{P}| - 1)}{C_1(|\mathcal{P}| - 1)} = 1 \tag{8}$$

Removing the uniform assumption on the density distribution requires the use of a second synopsis for determining the relevance of each pivot contribution. A generic strategy for weighting such contributions is using a synopsis on the pairwise distance distribution to represent the distance between the query element and the pivots in terms of a global p.d.f.. Therefore, having a synopsis on the joint pairwise distribution as $F(x)$, the probability $\texttt{Prob}(p)$ of s_q resembling the distribution $f_p(x)$ is proportional to $F(\delta(s_q, p))$ so that $\texttt{Prob}(p) = d(s_q, p)/C_3$, where $C_3 = \sum_{p \in \mathcal{P}} F(d(s_q, p))$. In this case, the joint probability of the query element resembling the pivots in $\mathcal{P}$ is given by Equation 9.

$$\sum_{p \in \mathcal{P}} \texttt{Prob}(p) = \frac{F\left(d(s_q, p_1)\right)}{C_3} + \frac{F\left(d(s_q, p_2)\right)}{C_3} + \cdots + \frac{F\left(d(s_q, p_{|\mathcal{P}|})\right)}{C_3} = \frac{\sum_{p \in \mathcal{P}} F\left(d(s_q, p)\right)}{\sum_{p \in \mathcal{P}} F\left(d(s_q, p)\right)} = 1 \tag{9}$$

3.2 Costs estimation of range queries

Let a range query $Rq(\mathcal{S}, s_q, \xi)$ to be executed in a metric tree $\texttt{M}$. All leaf nodes in $\texttt{M}$ that intercept closed query ball defined by $\langle s_q, \xi \rangle$ must be evaluated because their elements are potentially inside the query ball. Root nodes to these leaf nodes must be evaluated as well. Therefore, the local probability of accessing a node $\texttt{N}_j$ regarding a given pivot p is modeled upon the covering radius of the node (ξ_j) and the range query radius (ξ) as expressed by Equation 10.

$$\begin{aligned} \texttt{Prob}(\text{node is accessed}) &= \texttt{Prob}\{\delta(s_q, p) \leqslant \xi_j + \xi\} \\ \approx \bar{F}_p(\xi_j + \xi) &= \frac{\displaystyle\int_0^{\xi_j + \xi} f_p(x)dx}{\displaystyle\int_0^{v_p(m_p)} f_p(x)dx} \end{aligned} \tag{10}$$

Local probability is set to 1 whenever $\xi_j + \xi > v_p(m_p)$. The overall probability of accessing a node of $\texttt{M}$ is given by each pivot $p \in \mathcal{P}$ and the joining of Equations 8/9 and 10 into Equation 11. Likewise, Stockpile combines Equations 8/9 and 11 into Equation 12 for the estimation of distance calculations of range queries.

$$\texttt{nodes\_scanned}_{\texttt{Stockpile}}(\texttt{M}, s_q, \xi) \approx \sum_{\texttt{N}_j \in \texttt{M}} \sum_{p \in \mathcal{P}} \texttt{Prob}(p) \cdot \bar{F}_p(\xi + \xi_j) \tag{11}$$

$$\texttt{distances\_calculated}_{\texttt{Stockpile}}(\texttt{M}, s_q, \xi) \approx \sum_{\texttt{N}_j \in \texttt{M}} |\texttt{N}_j| \cdot \left(\sum_{p \in \mathcal{P}} \texttt{Prob}(p) \cdot \bar{F}_p(\xi + \xi_j) \right) \tag{12}$$

The intuition in Equation 12 considers the number of distance calculations is proportional to the probability of accessing each node, where $|\texttt{N}_j|$ is either the number of pivots (in the case of directory nodes) or the number of elements (in the case of leaf nodes).

3.3 Cost estimation of k-NN queries

Stockpile models the execution of k-NN searches by setting query radius ξ as the distance between the query element and its k^{th} neighbor, which reduces k-NN to range queries. Formally, given a k-$NN(\mathcal{S}, s_q, k)$ query and a p.d.f. $f_p(x)$ related to pivot p, Stockpile calculates the distance between s_q and its k^{th} neighbor as the threshold ξ'_p according to the inverse of Equation 13.

$$k = \frac{|S|}{\int_0^{v_p(m_p)} f_p(x)dx} \cdot \int_0^{\xi'_p} f_p(x)dx \tag{13}$$

where term $(|S|)/(\int_0^{v_p(m_p)} f_p(x)dx)$ is the uniform distribution of the synopsis approximation error. Thereafter, Stockpile combines the probability of selecting the pivot in Equations 8/9 to Equation 13 in such a way k-NN $(\mathcal{S}, s_q, k)$ is reduced to a range query whose radius depends on $\mathcal{P}$. The number of scanned nodes regarding a k-NN query is given by Equation 14, whereas the number of distance calculations is estimated as in Equation 15.

$$\texttt{nodes\_scanned}_{\texttt{Stockpile}}(\texttt{M}, s_q, k) \approx \sum_{\texttt{N}_j \in \texttt{M}} \sum_{p \in \mathcal{P}} \texttt{Prob}(p) \cdot \left(\frac{k}{|\mathcal{S}|} + \frac{\int_{\xi'_p}^{\xi'_p + \xi_j} f_p(x)dx}{\int_0^{v_p(m_p)} f_p(x)dx} \right) \tag{14}$$

$$\texttt{dist\_calc}_{\texttt{Stockpile}}(\texttt{M}, s_q, k) \approx \sum_{\texttt{N}_j \in \texttt{M}} |\texttt{N}_j| \cdot \left[\sum_{p \in \mathcal{P}} \texttt{Prob}(p) \cdot \left(\frac{k}{|\mathcal{S}|} + \frac{\int_{\xi'_p}^{\xi'_p + \xi_j} f_p(x)dx}{\int_0^{v_p(m_p)} f_p(x)dx} \right) \right] \tag{15}$$

3.4 Selectivity estimation

Stockpile provides a final estimation regarding the selectivity of range queries. The number of retrieved elements in range searches depends on the distances summarized by the p.d.f. $f_p(x)$ for every pivot in $\mathcal{P}$. Formally, given a range query Rq$(\mathcal{S}, s_q, \xi)$, Stockpile predicts the number of retrieved elements according to the Equation 16.

$$|Rq(\mathcal{S}, s_q, \xi)| \approx \left(\sum_{p \in \mathcal{P}} \texttt{Prob}(p) \cdot \bar{F}_p(\xi) \right) \cdot |\mathcal{S}| \tag{16}$$

The intuition in Equation 16 is the number of retrieved elements is related to the proportion of the area of p.d.f. related to each pivot, which is combined according to $\texttt{Prob}(p)$ and scaled by cardinality.

4. EXPERIMENTS

This section reports on the evaluation of the quality of `Stockpile` estimation for similarity searching optimization. Table II describes the group of representative real-world data sources employed in the experiments. We select data sources with varying cardinality (Card.), low-to-medium embedded dimensionality (Dim.), and intrinsic dimensionality ($\lceil \mathcal{D} \rceil$), whose content represent spatial (`CITIES` and `OCCUP`), business (`CARD` and `WINE`), image (`MAGIC` and `NASA`), and biology (`CASP`) patterns and characteristics. Accordingly, we experimented on `Stockpile` for comparing the predictions draw from several types of continuous and discrete-valued synopses on the same cost model. All methods were implemented by using the Arboretum library[2], the g++ compiler and the KUbuntu 17.01 OS on an Intel Core i7 2.67 GHz, 6 GB of RAM and HDD SATA III 7200 RPM.

Aiming at designing a common testbed for the synopses, we bounded `Stockpile` parameters to available memory and equally set budget per histogram to 256 buckets, the number of pivots to 5, and pivot criterion to maximum variance. As a result, we evaluated 5 `Stockpile` variations, as follows:

—**Distance exponent** (`DIST-EXP`), which employs a fractal-based continuous-valued synopsis. Such a `Stockpile` setting coadunates with the revised approach in Vieira et al. [2007];

—**Normal distribution** (`NORMAL`), which uses a Normal approximation of the pairwise distance distribution. This approximation is suitable for pairwise distributions that present the distance concentration phenomenon [Pestov 2012];

—**V-Optimal Histogram** (`V-OPT HIST`), which employs a discrete-valued V-Optimal histogram for the pairwise distribution representation. Such a `Stockpile` setting resembles the baseline model of Ciaccia et al. [1998], but it relies on a minimal error histogram;

—**Linear combination of local histograms** (`CDH-LINEAR`), which uses pivot-based CDH synopses. The rationale for weighting the estimates is the linear combination of dissimilarities between the query element and the pivots. Such a `Stockpile` variation extends the baseline model of Ciaccia et al. [1998] by following the indications of Tasan and Ozsoyoglu [2004], and without being tightly coupled to a specific index as in Aly et al. [2015] and Chen et al. [2015]; and

—**The pairwise weighting of local histograms** (`CDH-PAIR`), which employs pivot-based CDH synopses and one additional pairwise CDH for the weighting of individual estimates. Such a `Stockpile` setting extends the previous linear combination-based of CDH's in Bedo et al. [2017].

4.1 `Stockpile` radii estimation

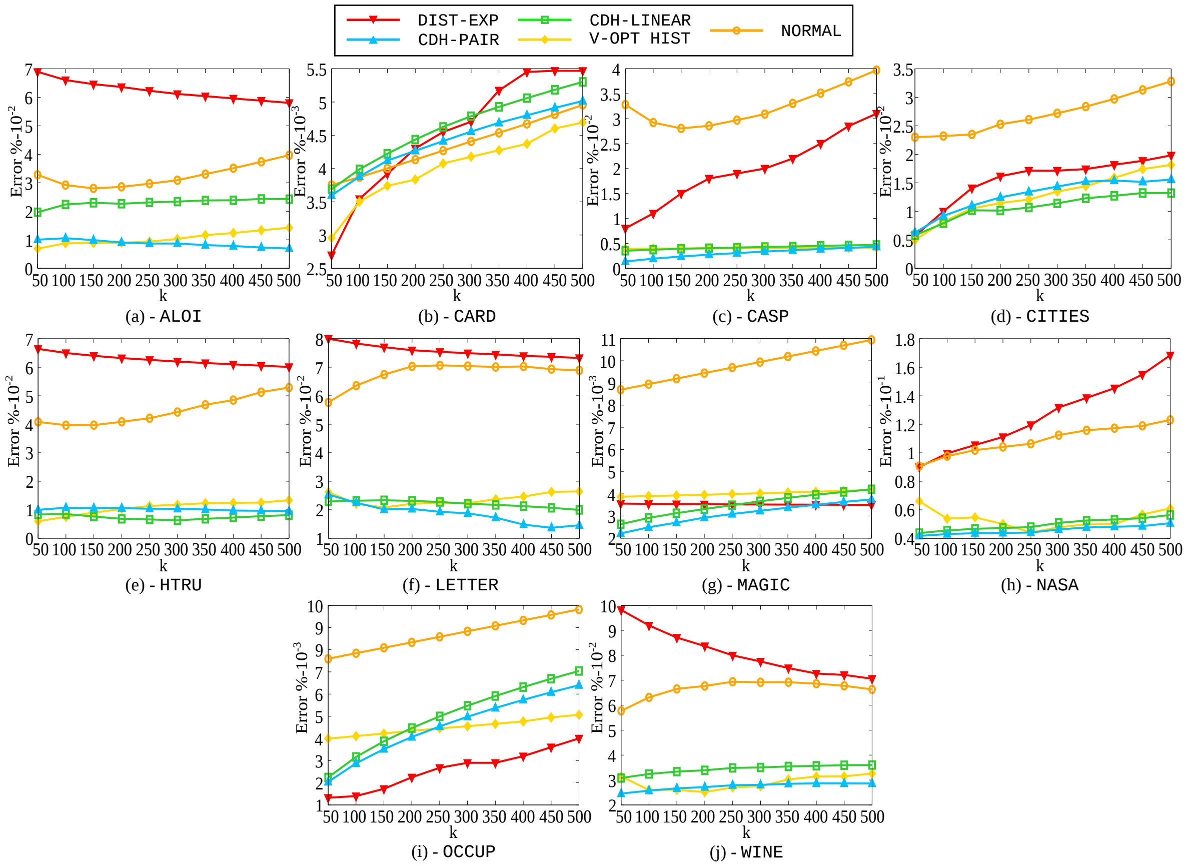
In the first experiment, we employed `Stockpile` parametric radius estimation so that estimates from `DIST-EXP`, `NORMAL`, and `V-OPT HIST` were calculated as in Equation 6, whereas `CDH-LINEAR` and `CDH-PAIR` as in Equation 13. We took 10% of random instances of each data source to emulate query elements and queried the remaining 90% of data. The neighborhood queries were defined for values of k ranging from 50 to 500 in steps of 50. We also calculated the absolute differences of the radii provided by the synopses to the true k-NN query radius (obtained after the k-NN query execution) and normalize that differences by the maximum pairwise distance within the queried elements, *i.e.*, Error %$= \frac{|\text{estimated\_radius} - \text{true\_radius}|}{\text{maximum\_distance}}$. Figure 5 shows the medians of Error % reached by each `Stockpile` parameterization. In overall, discrete synopses, *i.e.*, histograms, outperformed continuous-based synopses distance exponent and Normal distribution for most of data sources and the wider values of k. Both `DIST-EXP` and `NORMAL` settings presented *extreme* behaviors, *i.e.*, they achieved the best predictions for some cases (*e.g.*, `DIST-EXP` and `OCCUP` data sources), but reached the worst estimations in others (*e.g.*, `DIST-EXP` and `WINE` data sources). Such results indicate continuous

[2] Available at: `bitbucket.org/gbdi/arboretum`

Table II. data sources employed in the experiments.

Name	**Card.**	**Dim.**	$[\mathcal{D}]$	δ	**Available at:**
ALOI	110,250	13	8	Canberra	aloi.science.uva.nl
CARD	30,000	23	5	L_1	archive.ics.uci.edu/ml/data sources/default+of+credit+card+clients
CASP	45,730	09	4	Bray-Curtis	archive.ics.uci.edu/ml/data sources/Physicochemical+Properties+of+Protein+Tertiary+Structure
CITIES	5,507	02	2	L_1	www.ibge.gov.br
HTRU	17,898	08	6	Canberra	archive.ics.uci.edu/ml/data sources/HTRU2
LETTER	20,000	16	9	L_2	archive.ics.uci.edu/ml/data sources/letter+recognition
MAGIC	19,020	10	4	L_2	archive.ics.uci.edu/ml/data sources/magic+gamma+telescope
NASA	40,700	20	6	Canberra	dimacs.rutgers.edu/Challenges/Sixth/participants
OCCUP	20,560	05	3	Canberra	archive.ics.uci.edu/ml/data sources/Occupancy+Detection+
WINE	6,497	11	6	Bray-Curtis	archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/

synopses are suitable for *hardened* distributions, whereas discrete synopses are more *flexible* in capturing distribution nuisances. Moreover, histograms reached a *stable* behavior for Error % in every k value in all but CARD and OCCUP scenarios, *i.e.*, similar error ratio regardless of query parameters.



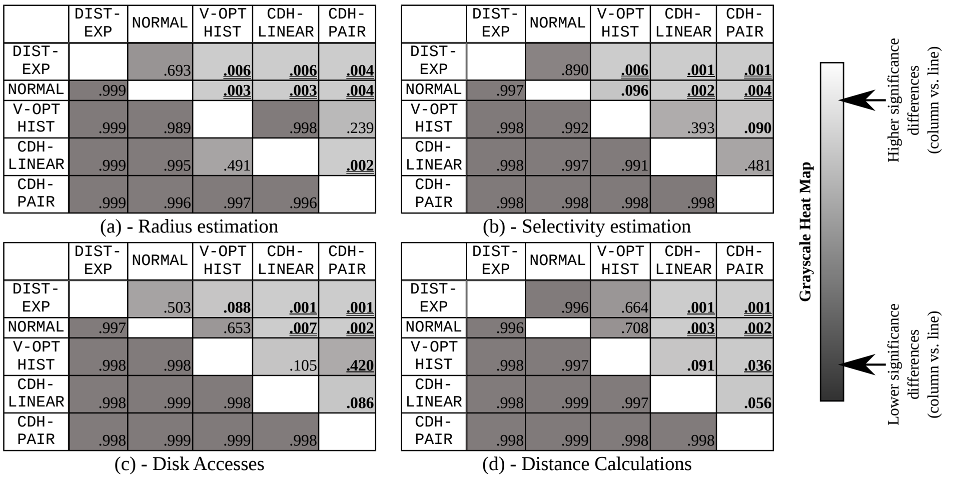
(a) - ALOI (b) - CARD (c) - CASP (d) - CITIES

(e) - HTRU (f) - LETTER (g) - MAGIC (h) - NASA

(i) - OCCUP (j) - WINE

Fig. 5. Differences in the radii estimation for the five evaluated Stockpile parameterizations.

We performed a statistical evaluation for determining if significant differences between the predictions could be found. Accordingly, we applied the Friedman ranking test [Demsar 2006] to check the differences among the predictions of 316,140 searches (31,614 query elements and 10 search parameters). By using a significance level of 0.1, we obtained a p-value of $2.1 \cdot 10^{-15}$ and, consequently, we rejected the hypothesis that differences among the approaches are due to random sampling and concluded at least one of them differs from the others. After the rejection of the Friedman's null hypothesis, we performed a *post-test* for determining the strategies whose predictions are significantly different. We applied the Nemenyi post-test [Garcia and Herrera 2008], which essentially uses a ranking strategy to determine the p-values for each pair of settings. In the post-test, the lower the p-value, the more significant the difference between the pair of compared `Stockpile` parameterizations. Figure 6 shows the grayscale heat map (column *versus* line) based on the Nemenyi p-values regarding the evaluated parameterizations. Table cells in lighter background present lower p-values. Additionally, the heat map also divides the Nemenyi p-values into four classes: *(i)* 99% confidence or higher – highlighted with a double underline, *(ii)* 95% to 99% confidence – single underline, *(iii)* 90% to 95% confidence – bold text, and *(iv)* lower than 90% confidence – non-bold text.



	DIST-EXP	NORMAL	V-OPT HIST	CDH-LINEAR	CDH-PAIR
DIST-EXP		.693	**.006**	**.006**	**.004**
NORMAL	.999		**.003**	**.003**	**.004**
V-OPT HIST	.999	.989		.998	.239
CDH-LINEAR	.999	.995	.491		**.002**
CDH-PAIR	.999	.996	.997	.996	

(a) - Radius estimation

	DIST-EXP	NORMAL	V-OPT HIST	CDH-LINEAR	CDH-PAIR
DIST-EXP		.890	**.006**	**.001**	**.001**
NORMAL	.997		**.096**	**.002**	**.004**
V-OPT HIST	.998	.992		.393	**.090**
CDH-LINEAR	.998	.997	.991		.481
CDH-PAIR	.998	.998	.998	.998	

(b) - Selectivity estimation

	DIST-EXP	NORMAL	V-OPT HIST	CDH-LINEAR	CDH-PAIR
DIST-EXP		.503	**.088**	**.001**	**.001**
NORMAL	.997		.653	**.007**	**.002**
V-OPT HIST	.998	.998		.105	**.420**
CDH-LINEAR	.998	.999	.998		**.086**
CDH-PAIR	.998	.999	.999	.998	

(c) - Disk Accesses

	DIST-EXP	NORMAL	V-OPT HIST	CDH-LINEAR	CDH-PAIR
DIST-EXP		.996	.664	**.001**	**.001**
NORMAL	.996		.708	**.003**	**.002**
V-OPT HIST	.998	.997		**.091**	**.036**
CDH-LINEAR	.998	.999	.997		**.056**
CDH-PAIR	.998	.999	.998	.998	

(d) - Distance Calculations

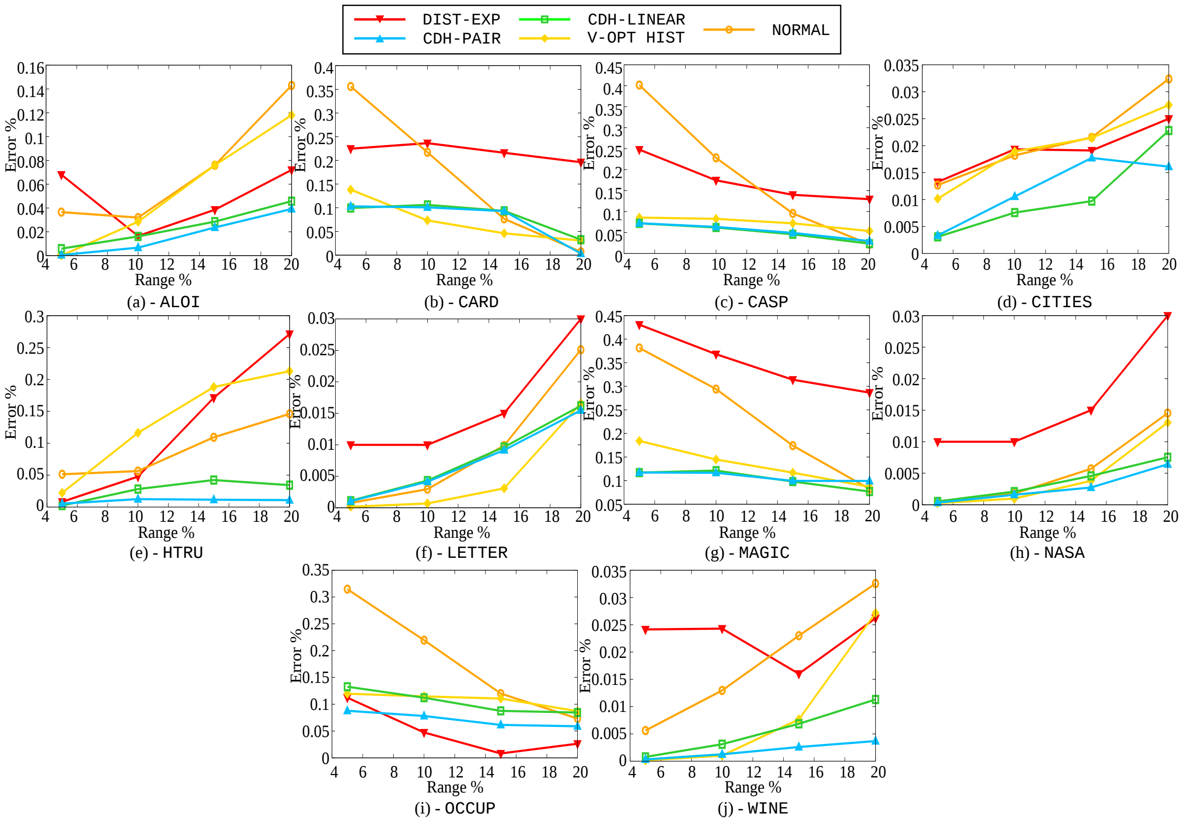
Grayscale Heat Map
Higher significance differences (column vs. line)
Lower significance differences (column vs. line)

Fig. 6. Grayscale heat maps of Nemenyi p-values. Double underline indicates column beats line within more than 99% confidence. Single underline indicates 95% to 99% confidence. Bold text indicates 90% to 95% confidence. Non-bold values indicate no significant difference was found.

According to the heat map in Figure 6(a), discrete synopses `V-OPT HIST`, `CDH-LINEAR` and `CDH-PAIR` were more suitable than continuous-based approaches `DIST-EXP` and `NORMAL` for radii estimation. Moreover, `CDH-PAIR` was also significantly better than `CDH-LINEAR`, which indicates using the pairwise distance distribution for the weighting of contributions of each pivot is more suitable than using the linear combination of their estimates.

4.2 `Stockpile` selectivity estimation

In this experiment, we also took 10% of random elements for selectivity evaluation and queried on the remaining 90% of data. We requested the five `Stockpile` parameterizations for providing selectivity estimates of range query and compare their outputs to the real number of returned elements in terms of absolute differences, *i.e.*, we measured Error $\%= \frac{|\text{estimated\_of\_elements}-\text{number\_of\_returned\_elements}|}{queried\_data\_cardinality}$. Figure 7 shows the medians of Error % calculated for selectivity estimations from `Stockpile` parameterizations regarding queries whose radii vary from 5% to 20% of the data source maximum pairwise distance. Graphs roughly present discrete synopses were more suitable than continuous-based distance representations, whereas overall Error % tends to decrease when radii coverage increases in most cases.



(a) - ALOI

(b) - CARD

(c) - CASP

(d) - CITIES

(e) - HTRU

(f) - LETTER

(g) - MAGIC

(h) - NASA

(i) - OCCUP

(j) - WINE

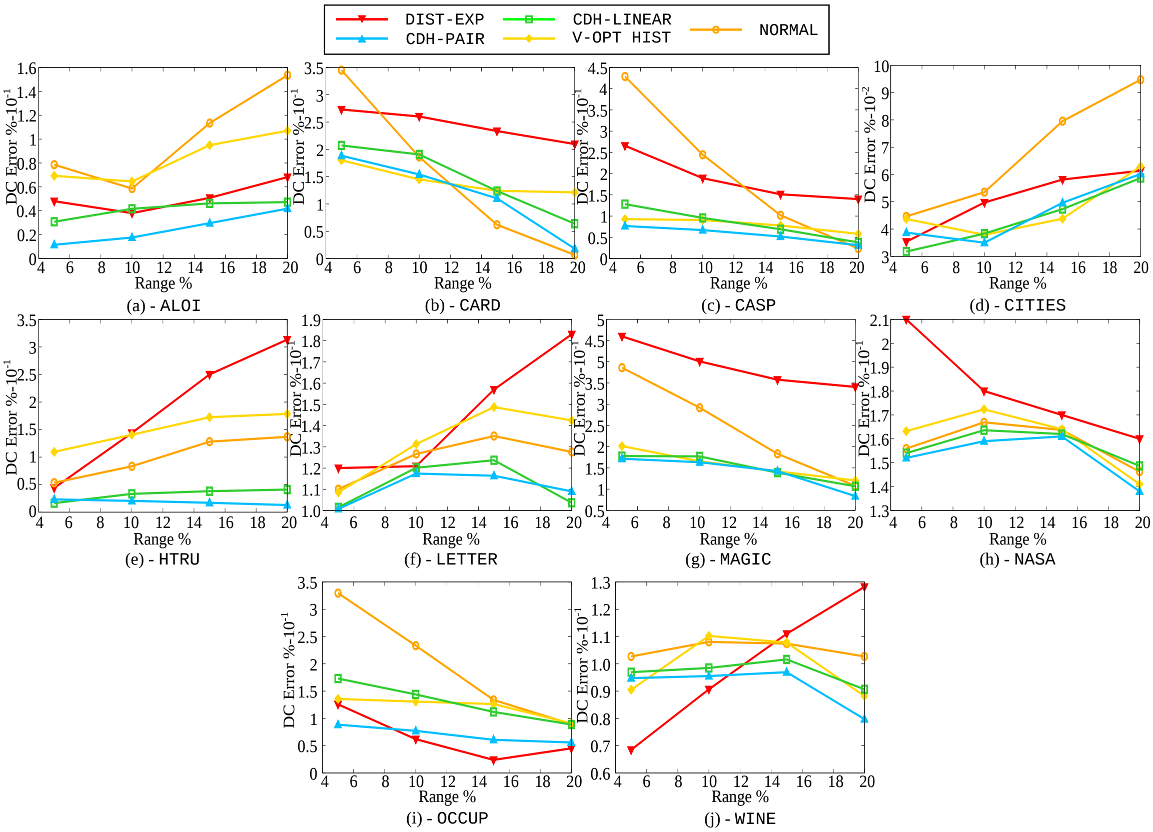
Fig. 7. Differences in the selectivity estimation for the five evaluated Stockpile parameterizations.

We applied the Friedman ranking test by using all predictions and every queried element and radius for the finding of the most suitable Stockpile settings. A p-value of $5.5 \cdot 10^{-9}$ was obtained for a significance level of 0.1 and, consequently, we were able to reject the Friedman's null hypothesis. Next, we resort to the Nemenyi post-test for the comparison of the p-values regarding every pair of Stockpile settings. Figure 6(b) presents the grayscale heat map based on the Nemenyi p-values by using the same rationale of the heat map of Figure 6(b). Predictions drawn from Stockpile with CDH-PAIR outperformed Stockpile with V-OPT HIST estimations for a 0.1 confidence level, whereas synopsis CDH-PAIR outperformed DIST-EXP and NORMAL within 0.05 confidence level. Likewise, Stockpile with both V-OPT HIST and CDH-LINEAR outperformed DIST-EXP and NORMAL by significant levels.

Again, synopses based on histograms reached the lowest Error % (as in the case of radii prediction), whereas CDH-PAIR also outperformed V-OPT HIST regarding selectivity estimation. Although histograms were more *stable* than continuous-valued synopses, Error % of Normal-based setting dropped for ranges closer to the mean of the pairwise distribution, *e.g.*, CARD, CASP, MAGIC, and OCCUP. Last, but not least, results show selectivity predictions tend to converge as range increases (CARD, CASP, CITIES, LETTER, MAGIC, and OCCUP), which is a different behavior of that observed for radii estimation regarding increasing values of k and, consequently, larger neighborhood distances.

4.3 Stockpile I/O and CPU cost estimation

In our last experiment, we compared the Stockpile parameterizations in the task of providing I/O and CPU cost estimation. In this comparison, we indexed the queried elements and consider the number of scanned nodes as the I/O costs and the number of distance calculations as the CPU costs. We used 10% random sampling of elements for the querying of the remaining 90% of elements indexed on Slim-Trees by using the default Arboretum parameters. Figure 8 summarizes the comparison



(a) - ALOI (b) - CARD (c) - CASP (d) - CITIES

(e) - HTRU (f) - LETTER (g) - MAGIC (h) - NASA

(i) - OCCUP (j) - WINE

Fig. 8. Differences in the prediction of distance calculation for the five evaluated Stockpile parameterizations.

between the medians of predictions regarding distance calculations of range queries with radii varying from 5% to 20% of the maximum pairwise distance of the indexed elements. Estimates from DIST-EXP, NORMAL, and V-OPT HIST were calculated following Equations 4 and 5, whereas estimates from CDH-LINEAR and CDH-PAIR were calculated as in Equations 11 and 12. Error % was calculated as $\frac{|\text{estimated\_CPU\_cost}-\text{true\_CPU\_cost}|}{queried\_data\_cardinality}$. We omitted the plots of medians for I/O costs as they were similar to that in Figure 8. In overall, results indicate histograms were more suitable than continuous-based synopses. Again, we applied the Friedman's ranking test to determine if the predictions were significantly different. For a significance level of 0.1, we obtained p-values of $5.9{\cdot}10^{-13}$ and $1.9{\cdot}10^{-9}$ regarding I/O and CPU costs. Accordingly, we applied the Nemenyi post-test in both cases. The results for I/O cost estimation, as shown in Figure 8(c), indicate Stockpile with CDH-PAIR outperformed all competitors within significant levels. As for CPU costs, as shown in Figure 8(c), Stockpile with CDH-PAIR also outperformed all competitors, but CDH-LINEAR was more suitable than V-OPT HIST. As in the case of selectivity estimation, Error % ratio of Normal-based setting drops for ranges closer to the mean of the pairwise distributions, whereas Error % of all synopsis predictions seem to converge in most of the scenarios as the distances' range increases.

We highlight, unlike the competitors, estimates drawn from Stockpile with CDH-PAIR were among the most suitable in the four experimented scenarios: radii, selectivity, I/O and CPU costs estimation. For instance, CDH-LINEAR and V-OPT HIST were also suitable for specific tasks, as radii estimation, but they failed in I/O and CPU prediction. Therefore, results show Stockpile with CDH-PAIR provided the best predictions for the estimation of similarity searching costs.

5. CONCLUSIONS

Cost modeling of similarity searches requires a proper handling of distance distributions. In this study, we discussed several distribution representations as probability density functions, called *synopses*, and estimation rules to be drawn from them. We also presented the `Stockpile` cost model that relies on pivot-based synopses for estimating query radii, selectivity, I/O and CPU costs. We performed an extensive set of experiments on real-world data sources and three `Stockpile` settings (`V-OPT HIST`, `CDH-LINEAR`, `CDH-PAIR`) have outperformed their competitors in radii and selectivity estimation, whereas one of them (`CDH-PAIR`) has also surpassed the others in the prediction of I/O and CPU costs. An extension to the isolated and parametric `Stockpile` cost model would be the design of a model for weighting and combining predictions from different cost models tightly coupled to specific metric indexes, which we will investigate as future work.

Acknowledgments. This research was supported by FAPERJ (E. 11/2018 – Sediadas), FAPESP (G. 016/17078-0), and CAPES.

REFERENCES

Aly, A. M., Aref, W. G., and Ouzzani, M. Cost estimation of spatial k-nearest-neighbor operators. In *Int. Conference on Extending Database Technology*. Springer, Berlin, pp. 457–468, 2015.

Bedo, M. V. N., Kaster, D. S., Traina, A. J., and Traina Jr., C. The Merkurion approach for similarity searching optimization in Database Management Systems. *Data & Knowledge Engineering* vol. 113, pp. 18 – 42, 2018.

Bedo, M. V. N., Traina, A. J. M., and Traina Jr., C. A spline-based cost model for metric trees. In *Brazilian Symposium on Databases*. SBC, SBC, Porto Alegre, pp. 28–39, 2017.

Cha, S.-H. Comprehensive survey on distance/similarity measures between probability density functions. *Int. Journal of Mathematical Models and methods in Applied Sciences* 1 (2): 8, 2007.

Chen, L., Gao, Y., Li, X., Jensen, C., and Chen, G. Efficient metric indexing for similarity search. In *International Conference on Data Engineering*. IEEE, IEEE, New York, pp. 591–602, 2015.

Chen, L., Gao, Y., Zheng, B., Jensen, C. S., Yang, H., and Yang, K. Pivot-based metric indexing. *Proceedings of the VLDB Endowment* 10 (10): 1058–1069, 2017.

Ciaccia, P., Nanni, A., and Patella, M. A query-sensitive cost model for similarity queries with m-tree. In *Australasian Database Conference*. Springer, Berlin, pp. 65–76, 1999.

Ciaccia, P., Patella, M., and Zezula, P. A cost model for similarity queries in metric spaces. In *Symposium on Principles of Database Systems*. ACM, New York, pp. 59–68, 1998.

Clauset, A., Shalizi, C. R., and Newman, M. E. Power-law distributions in empirical data. *Society for Industrial and Applied Mathematics Review* 51 (4): 661–703, 2009.

Cormode, G., Garofalakis, M., Haas, P. J., and Jermaine, C. Synopses for massive data: Samples, histograms, wavelets, sketches. *Foundations and Trends in Databases* 4 (1–3): 1–294, 2012.

Demsar, J. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* vol. 7, pp. 1–30, 2006.

Garcia, S. and Herrera, F. An extension on "Statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. *Journal of Machine Learning Research* vol. 9, pp. 2677–2694, 2008.

Hetland, M. L. The basic principles of metric indexing. In *Swarm Intelligence for Multi-objective Problems in Data Mining*. Springer, Berlin, pp. 199–232, 2009.

Ioannidis, Y. The history of histograms (abridged). In *Int. Conference on Very large Data Bases*. VLDB Endowment, New York, pp. 19–30, 2003.

Lu, Y., Lu, J., Cong, G., Wu, W., and Shahabi, C. Efficient algorithms and cost models for reverse spatial-keyword kNN search. *ACM Transactions on Database Systems* 39 (2): 13:1–13:46, 2014.

Pestov, V. Indexability, concentration, and VC theory. *Journal of Discrete Algorithms* vol. 13, pp. 2–18, 2012.

Shekelyan, M., Dignös, A., and Gamper, J. DigitHist: A Histogram-based Data Summary with Tight Error Bounds. *Proceedings of VLDB Endowment* 10 (11): 1514–1525, 2017.

Tao, Y., Zhang, J., P., D., and Mamoulis, N. An efficient model for optimization of kNN in low and medium dimensional spaces. *Transactions on Knowledge and Data Engineering* 16 (10): 1169–1184, 2004.

Tasan, M. and Ozsoyoglu, Z. M. Improvements in distance-based indexing. In *Scientific and Statistical Database Management*. IEEE, New York, pp. 161–170, 2004.

Traina, A. J., Traina Jr., C., Faloutsos, C., and Seeger, B. Fast indexing and visualization of metric data sets using Slim-Trees. *Transactions on Knowledge and Data Engineering* 14 (2): 244–260, 2002.

Vieira, M. R., Traina Jr., C., Traina, A. J., Arantes, A., and Faloutsos, C. Boosting k-nearest neighbor queries estimating suitable query radii. In *Int. Conference on Scientific and Statistical Database Management.* IEEE, New York, pp. 10–10, 2007.

Zezula, P., Amato, G., Dohnal, V., and Batko, M. *Similarity search: the metric space approach.* Vol. 1. Springer, Berlin, 2010.