Information Sciences 361–362 (2016) 14–28

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Optimizing skyline queries over incomplete data

Jongwuk Lee[a], Hyeonseung Im[b], Gae-won You[c,*]

[a] Division of Computer and Electronic Systems Engineering, Hankuk University of Foreign Studies, Republic of Korea
[b] Department of Computer Science, Kangwon National University, Republic of Korea
[c] Samsung Advanced Institute of Technology, Samsung Electronics, Republic of Korea

ARTICLE INFO

Article history:
Received 19 August 2015
Revised 25 April 2016
Accepted 29 April 2016
Available online 6 May 2016

Keywords:
Skyline queries
Incomplete data
Dominance
Incomparability
Intransitivity
Cyclicity

ABSTRACT

Skyline queries have been widely used as an attractive operator in multi-criteria decision making applications. Because of the intuitive notion of skyline queries, many skyline algorithms have been developed in various data settings. However, most of the skyline algorithms rely on the assumption of *completeness*, *i.e.*, all values of points are known. In many cases, because this assumption does not hold, conventional skyline algorithms cannot be applied. To handle *incomplete* data, existing work redefines the *dominance* notion by using the *common subspace* between points. However, it can incur too many pairwise comparisons over incomplete data. To address this problem, we first propose a new sorting-based bucket skyline algorithm using two optimization techniques: *bucket-* and *point-level orders*. In case that too few or no skyline points exist over incomplete data, we develop a novel skyline ranking method that adjusts two user-specific parameters for retrieving meaningful skyline points. Lastly, we empirically evaluate the efficiency and effectiveness of our proposed algorithms over both synthetic and real-life datasets.

1. Introduction

Skyline queries have been widely used for supporting multi-criteria decision making applications such as e-business, supply chain management [33], and power management [35]. Given two points p and q on d-dimensional space $\mathcal{D}$, it is said that p *dominates* q, denoted by $p \succ q$, if p is better than q in at least one dimension and is no worse than q in any other dimensions. Given a set $\mathcal{S}$ of n points, a skyline query finds a set of points (or a *skyline*) that are not dominated by any other points in $\mathcal{S}$ [2,7,12,15,20,23,26,30,34,40]. Based on the property of the dominance notion, the skyline always contains all top-1 points for any monotone ranking functions.

Because the dominance notion is intuitive and easy to understand, the user can easily formulate skyline queries. For example, let us consider a classical hotel-finding scenario. A user may formulate a skyline query to retrieve hotels that are cheap and close to the conference venue. Fig. 1 depicts two-dimensional points for hotels, where D_1 and D_2 indicate the price and the distance to the conference venue, respectively. The user prefers a hotel with a lower price and shorter distance. Given a dataset $\mathcal{S}$ with 10 points, we compare all pairwise points, and remove dominated ones. That is, because $b \succ \{a, c, f, i\}$, $g \succ j$, and $h \succ k$, the skyline becomes $\{b, e, g, h\}$ (Fig. 1). Given any monotone ranking function, the skyline is enough to identify a top-1 point without examining the whole dataset.

* Corresponding author. Tel.: +821062461755.
E-mail addresses: julee@hufs.ac.kr (J. Lee), hsim@kangwon.ac.kr (H. Im), gaewonyou@gmail.com (G.-w. You).

http://dx.doi.org/10.1016/j.ins.2016.04.048
0020-0255/

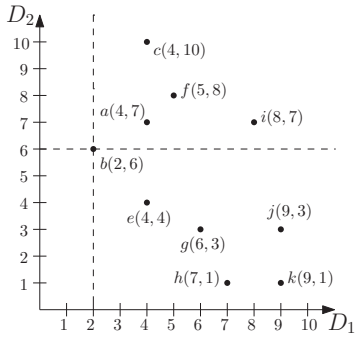

Fig. 1. Skyline queries in two-dimensional space.

Owing to the useful property of skyline queries, many research efforts have been devoted to various data settings such as stream [27], distributed [1,37], and multicore [16,19,36] environments. Most of the algorithms rely on the assumption of *completeness*, *i.e.*, all values of points are *known*. However, the completeness assumption does not hold in many real-life datasets.

Let us consider a movie-rating application with hundreds of movies and thousands of users, *e.g.*, MovieLens[1]. Each user is highly likely to rate some movies and ratings of many movies remain incomplete. In this case, because the existing dominance notion does not work for incomplete data, conventional skyline algorithms cannot be applied. To address this problem, one simple solution is to replace all missing values with minimum (or maximum) values, and then to apply conventional skyline algorithms. However, it is unclear whether the skyline shows a meaningful and satisfactory result. Although Bartolini et al. [3] addressed the skyline computation over incomplete data, they defined the dominance notion with a probabilistic value. In this sense, it is indispensable to refine the dominance notion between points to handle *incomplete* data.

We go beyond this completeness assumption by refining the dominance notion over incomplete data [17]. Given two points p and q, the dominance notion between p and q is defined on a *common subspace* $\mathcal{U} \subseteq \mathcal{D}$, where the values of p and q are both known. It is intuitive that only the known values on the common subspace are used. However, two issues arise from the refined dominance notion:

- **Intransitivity**: Given $p, q, r \in \mathcal{S}$, if $p \succ q$ and $q \succ r$ hold, $p \succ r$ also holds for complete data. However, $p \succ r$ may not hold for incomplete data.
- **Cyclicity**: Given $p, q, r \in \mathcal{S}$, $p \succ q$, $q \succ r$, and $r \succ p$ may hold over incomplete data.

Owing to the two issues, conventional skyline algorithms cannot be applied. A simple solution is to perform all exhaustive pairwise comparisons for n incomplete points, incurring a quadratic cost $\binom{n}{2}$.

In this paper, we first aim to optimize the efficiency of skyline computation over incomplete data. Specifically, we partition a dataset $\mathcal{S}$ into disjoint sets of points with the same known dimensions, called *buckets* [17]. Because the points within the same bucket lie on the same subspace, we can easily prune out non-skyline points by using traditional skyline algorithms, and reduce unnecessary pairwise comparisons. However, it is difficult to dispense with unnecessary comparisons between points across different buckets over incomplete data.

We observe that unnecessary comparisons across buckets can be significantly removed by adjusting the order of accessing points. Specifically, we propose the following optimization techniques.

- **Bucket-level optimization**: It is more effective to compare points across buckets with smaller common subspaces to remove unnecessary comparisons. After points are partitioned into buckets, we assess points by sorting buckets in ascending order of the size of their known dimensions.
- **Point-level optimization**: Within the bucket, we sort points by descending order of the sum of known values of points as in [12]. The reordering of points is more effective for pruning non-skyline points early on.

We next discuss a novel skyline ranking scheme to retrieve more meaningful skyline points over incomplete data. Whereas the number of skyline points for complete data increases exponentially with the dimensionality [5], an opposite phenomenon happens when incomplete data have a lot of intransitive and cyclic dominance relationships. In particular, as the size of the common subspace between points is smaller, more cyclic relationships can happen. In this case, too few or no skyline points can be returned. To alleviate this problem, we develop a skyline ranking method with two parameters, which control the size of the common subspace and relax the number of dominating points. This method is implemented on top of our proposed skyline algorithm. Despite its simplicity, it is effective for retrieving meaningful skyline points over incomplete data.

To summarize, this paper makes the following contributions.

[1] https://grouplens.org/datasets/movielens/