Approches basées sur la distribution pour l'estimation de la taille du skyline

Nicolas Hanusse, Patrick Kamnang Wanko, Sofian Maabout

LaBRI. Université de Bordeaux. CNRS. France
{hanusse,pkamnang,maabout}@labri.fr

RÉSUMÉ. *Soit T un ensemble de points dont chacun est défini sur les mêmes attributs. Soit $p \in T$, p est un point skyline de T ssi il n'existe dans T aucun point meilleur que p. Une requête skyline retourne l'ensemble des points skyline. Le nombre de tuples dans le résultat n'est pas toujours proportionnel à la taille (nombre de tuple) des données en entrée. C'est pourquoi estimer la taille du skyline est une question cruciale à laquelle plusieurs travaux se sont consacrés ces dernières années. Ce problème est particulièrement important dans l'optique de l'intégration de l'opérateur skyline au sein de systèmes de gestion de bases de données. S'appuyant sur la distribution des valeurs, nous proposons deux estimateurs présentant de bonnes propriétés : le premier est sans biais et nécessite un parcours de l'ensemble des données tandis que le second est convergent en loi. Ces estimateurs sont simples à mettre en œuvre et montrent leur précision et leur efficience à travers les résultats des expérimentations qui ont été faites.*

ABSTRACT. *Let T be a set of points each of which is described by the same set of attributes. Let $p \in T$. Then p is a skyline point of T iff there is no other point* better *than p. A skyline query returns the set of all skyline points. The number of tuples in the result is not always proportional to the size of the source data. Hence, estimate skyline cardinality is a challenging problem which attracted a great deal of work in recent years. This problem is particularly important in case of integration of the skyline operator in database management systems. We propose techniques for estimating skyline cardinality. We first provide an* unbiased *estimator of the skyline cardinality which requires a pass on data. On another hand, we provide a* convergent *estimator which does not require any data scanning. It estimates skyline cardinality* expectation *for those data sets respecting data distribution given as input. The advantages of these solutions are the ease of implementation and, by contrast to other proposals, no costly subskyline queries are required. Our solutions are implemented and some experiments are reported showing both the accuracy of the estimations and the efficiency by which they are obtained.*

MOTS-CLÉS : *Skyline, optimisation, dépendances fonctionnelles*

KEYWORDS: *Skyline cardinality, expectation, sampling, estimation, algorithms*

DOI:10.3166/ISI.21.3.119-140

1. Introduction

Déterminer le skyline d'un ensemble d'objets est une requête courante survenant dans divers domaines. Ce concept se présente à la littérature sous différentes appelations : requête de préférence, vecteur maximal, front de Pareto. Le terme *skyline* a été employé pour la première fois dans Börzsönyi *et al*. (2001) où l'auteur propose un nouvel opérateur permettant d'étendre les tâches des systèmes de gestion de bases de données.

Dans ce papier, nous analysons le problème de l'estimation de la taille du skyline. Ce problème est de grande importance dans l'éventualité de l'intégration de l'opérateur skyline au sein des systèmes de gestion de bases de données. En effet, l'optimisation de requêtes requière la connaissance d'une idée (estimation) de la taille des résultats intermédiaires afin de proposer les meilleurs plans d'exécution. Par exemple, estimer le nombre de tuples retournés par une requête de sélection peut être possible si on connait le nombre de valeurs distinctes. Cette estimation peut être plus précise si nous connaissons également la distribution de ces valeurs (l'histogramme des valeurs). De la même façon, nous prenons en consideration la connaissance de la distribution des données dont on veut estimer la taille du skyline.

Au regard des solutions proposées (état de l'art) dans la littérature, pour l'estimation de la taille du skyline, nous apportons deux contributions majeures. D'une part, nous améliorons la complexité des calculs dans ce sens que le calcul ne dépend pas de la taille réelle du skyline comme c'est le cas pour certains travaux dont nous parlerons dans la partie 2. D'autre part, nos propositions ne reposent pas sur des calculs complexes tels que les plus proches voisins, les méthodes à noyaux, le calcul de skylines d'échantillons ou encore le calcul de multiples intégrales ; ce qui en facilite l'implémentation. Par exemple, lorsque les données sont parcourues pour estimer la taille du skyline, notre proposition consiste en le calcul d'une simple somme de produits de termes.

Nous commençons par présenter les principaux concepts et notations qui seront utilisés le long de cet article.

Définition des concepts et notations

Soit $T(Id, D_1, \ldots, D_d)$ une table relationnelle. L'ensemble $\mathcal{D} = \{D_1, \ldots, D_d\}$ est l'ensemble des dimensions de T. Il s'agit de l'*espace multidimensionnel*. Pour tout D_i, nous supposons qu'il existe un ordre total $<_i$ sur les valeurs de D_i. Soient $v, v' \in D_i$, alors v est *préféré* à v' ssi $v <_i v'$. Soient $t, t' \in T$. Alors t est *dominé* par t' ssi $t'[D_i] \leq_i t[D_i]$ pour tout D_i et il existe j tel que $t'[D_j] <_j t[D_j]$. Le skyline de T est l'ensemble des tuples qui ne sont pas dominés dans T. Soient n le nombre de tuples appartenant à T (la taille de T) et k_j le nombre de valeurs distinctes de la dimension D_j, i.e., $k_j = |\pi_{D_j}(T)|$. k_j est aussi appelé la cardinalité de la dimension D_j. Le tableau 1 résume les différentes notations utilisées tout le long de cet article.

Tableau 1. Les notations

Notation	Definition
T	instance de relation
$\mathcal{D}$	l'ensemble d'attributs (dimensions)
d	nombre de dimensions ($d = \|\mathcal{D}\|$)
D_j	j^{ieme} dimension de T
k_j	cardinalité de la j^{ieme} dimension
$f_j(x)$	la fonction de distribution de la j^{ieme} dimension
$F_j(x)$	la fonction de distribution cumulative de la j^{ieme} dimension
$\mathcal{T}$	l'ensemble de tous les tuples possibles
n, m	le nombre de tuples
t, t', t_i, t_j	les tuples de T
$t_i[j]$	la projection d'un tuple t_i sur la j^{ieme} dimension
q_{t_ℓ}	la probabilité que le tuple t_ℓ appartienne à T
p_{t_ℓ}	la probabilité que le tuple $t_\ell \in T$ appartienne au skyline $Sky(T)$
$t' \prec t$	t' domine t
$Sky(T)$	le skyline de T
$\|Sky(T)\|$	la taille (cardinalité) du skyline de T

EXEMPLE 1. — Le tableau 2 présente le jeu de données que nous prendrons comme exemple dans les différentes parties. Dans ces données, l'utilisateur cherche les meilleurs hôtels au regard des différents critères apparaissant dans le tableau. Ces critères sont le prix de la chambre d'hôtel, la distance jusqu'à l'arrêt bus, la surface de la chambre et le classement de l'hôtel en termes de nombre d'étoiles.

Tableau 2. Les hôtels: données brutes

Hôtels	Prix	Dist.	Surf.	Clas.
$h1$	150	100	10	****
$h2$	150	1000	10	***
$h3$	150	100	20	****
$h4$	40	100	40	***
$h5$	300	100	10	*****
$h6$	150	5000	40	*****
$h7$	40	100	10	***
$h8$	300	1000	10	***
$h9$	150	100	20	*****
$h10$	40	100	20	*****
$h11$	150	1000	20	****
$h12$	150	5000	20	***

L'hôtel h_{12} ne fait pas partie du skyline parce qu'il est dominé par l'hôtel h_6. En effet, l'hôtel h_6 est meilleur que l'hôtel h_{12} en termes de surface (40 pour h_6 contre 20 pour h_{12}) et de classement (l'hôtel h_6 est un cinq-étoiles tandis que l'hôtel h_{12} est un trois-étoiles) ; ces deux hôtels sont néanmoins équivalents au regard des autres attributs (à savoir le prix et la distance). Dans ce cas, k_j vaut 3 pour tous les attribut D_j, j allant de 1 à 4. L'ensemble skyline des hôtels au regard de toutes les dimensions est constitué de h_4, h_6 et h_{10}. Donc la taille exacte du skyline est de 3 ($|Sky(T)| = 3$). □

Notons que l'utilisateur cherche à minimiser le prix tandis qu'il veut maximiser la surface. Maximiser la surface équivaut à minimiser la différence entre la surface maximale et la surface de la chambre d'hôtel courante. L'ensemble peut être transformé en un problème de minimisation de chacun des critères en remplaçant chaque valeur par son *rang* dans le tri obtenu en utilisant la relation d'ordre $<_j$ comme critère de tri.

EXEMPLE 2. — De notre exemple courant, le rang de la valeur 40 dans la dimension surface est 1 (préférence pour les valeurs élevées). La valeur 40 a également le rang 1 pour l'attribut prix mais cette fois parce qu'il s'agit de la plus petite valeur (préférence pour les valeurs faibles pour le prix). Le fait de remplacer les valeurs par leur rang selon l'ordre naturel pour chacune des dimensions ne change pas l'ensemble skyline. En guise d'illustration, le tableau 3 présente la normalisation (transformation) du tableau 2 suivant ce principe.

Tableau 3. Les hôtels: Données normalisées

T	D_1	D_2	D_3	D_4
$h1$	2	1	3	2
$h2$	2	2	3	3
$h3$	2	1	2	2
$h4$	1	1	1	3
$h5$	3	1	3	1
$h6$	2	3	1	1
$h7$	1	1	3	3
$h8$	3	2	3	3
$h9$	2	1	2	1
$h10$	1	1	2	1
$h11$	2	2	2	2
$h12$	2	3	2	3

□

Donc, sans nuire à la généralité et par simplicité de présentation, nous considèrerons désormais la préférence pour les valeurs faibles.

Nous rappelons la définition de quelques notions statistiques sur lesquelles repose la compréhension de ce document.

DÉFINITION 3 (Indépendance). — *Soient X et Y deux variables aléatoires. X et Y sont dites indépendantes ssi le processus de génération de X est indépendant de celui de Y.*

DÉFINITION 4 (Espérance). — *On suppose que X peut prendre la valeur x_1 avec la probabilité p_1, la valeur x_2 avec la probabilité p_2, et ainsi de suite jusqu'à x_k avec la probabilité p_k. Alors, l'espérance de cette variable aléatoire X, notée $E(X)$ s'écrit*

$$E(X) = x_1 \times p_1 + x_2 \times p_2 + \cdots + x_k \times p_k$$

DÉFINITION 5 (Estimateur d'Horvitz-Thompson). — *Horvitz et Thompson (1952) Formellement, soit $\mathcal{T} = \{t_i, i = 1, 2, \ldots, N\}$ la population constituée de N individus distincts, soit $Y(t_i)$ la valeur observée du tuple t_i pour la variable aléatoire Y. Soit $\mathcal{S}$ l'échantillon indépendant tiré avec remise de $\mathcal{T}$ tel que $|\mathcal{S}| = n$. On suppose en plus que π_i est la probabilité qu'un tuple $t_i \in \mathcal{T}$ appartienne à l'échantillon $\mathcal{S}$, $\pi_i = Prob(t_i \in \mathcal{S})$. L'estimateur de Horvitz-Thompson $\hat{Y}_{HT}$ du total $\sum_{t_i \in \mathcal{T}} Y(t_i)$ est donné par*

$$\hat{Y}_{HT} = \sum_{t_i \in S} \pi_i^{-1} Y(t_i)$$

DÉFINITION 6 (Estimateur sans biais). — *Soit s un paramètre d'une population et soit $\hat{s}$ un estimateur de s. $\hat{s}$ est qualifié d'estimateur sans biais de s si la différence entre l'espérance de l'estimateur et la valeur exacte du paramètre estimé est égale à zero ; formellement, $E(\hat{s}) - s = 0$*

DÉFINITION 7 (Théorème Central Limite). — *Soit $\{U_1, \ldots, U_d\}$ un échantillon aléatoire de taille d. C'est une séquence de variable aléatoire indépendantes et identiquement distribuées suivant une distribution d'espérance μ_U et de variance finie notée σ_U^2. Lorsque $d \to \infty$, alors la distribution de $Y = \sum_{j=1}^{d} U_j$ converge vers la distribution normale $\mathcal{N}(d \times \mu_U, d \times \sigma_U^2)$*

Organisation de l'article

Dans la section suivante, nous résumons quelques travaux précédents relatifs à l'estimation de la cardinalité du skyline. Ensuite, nous présentons nos propositions reposant sur la connaissance mise à disposition des données. Enfin, sont présentées les évaluations expérimentales montrant aussi bien la précision de nos propositions que leur rapidité d'exécution.

2. Travaux relatifs

Plusieurs algorithmes ont été proposés pour calculer les requêtes skyline, en guise d'exemple, Börzsönyi *et al*. (2001) ; Lee et Hwang (2010) ; Morse *et al*. (2007) ; Chomicki *et al*. (2003) ; Bartolini *et al*. (2008). Bien que ces algorithmes reposent sur différents principes, ils possèdent tous une complexité dans le pire des cas de l'ordre

de $\mathcal{O}(n^2)$. Donc, afin de présenter un intérêt pratique, tout algorithme dédié à l'estimation de la taille du skyline devrait avoir une complexité moindre. Nous évoquons alors quelques précédents travaux à ce sujet. Bentley *et al.* (1978) considère le cas où les dimensions sont indépendantes et les valeurs apparaissant dans chacune d'elles disctinctes. Sous ces hypothèses, les auteurs montrent que l'espérance de la taille du skyline est de l'ordre de $\mathcal{O}\left((\ln n)^{d-1}\right)$. Dans Buchta (1989), la formule précédente a été améliorée en montrant que le nombre de vecteurs maximum est de l'ordre de $\Theta\left(\frac{(\ln n)^{d-1}}{(d-1)!}\right)$. L'avantage de ce résultat est de donner une idée de la taille du skyline mais ce résultat reste imprécis. Sous les mêmes hypothèses, Godfrey *et al.* (2004) montre que la taille du skyline peut être estimée par le $(d+1)^{iem}$ ordre de la série harmonique en n, $H_{d+1,n} = \sum_{i=1}^{n} \frac{H_{d,i}}{i}$ où $H_{0,n} = 1$ et d est le nombre de dimensions. Même si cette formule est plus précise , elle reste sans intérêt d'un point de vue pratique car elle requière une grande quantité de mémoire, de l'ordre de $\mathcal{O}(n^2)$ ce qui la rend aussi difficilement applicable que le calcul du skyline. Chaudhuri *et al.* (2006) présente des formules théoriques donnant l'espérance exacte de la taille du skyline sans aucune contrainte sur la nature des données (dimensions n'étant pas forcement indépendantes et présence éventuelle de repétitions de valeurs par attribut). Même si cela semble attrayant, ces formules contiennent d intégrales simples ou une intégrale sur un espace à d dimensions ce qui les rend difficiles à implémenter. Plus précisement, la probabilité qu'un tuple appartienne au skyline est donnée par [1] $p = \int_{[0,1]^d} f(x).(1 - F(x))^{n-1} dx$ où $f(x)$ est la fonction de densité jointe des valeurs des tuples et $F(x)$ est la fonction de densité cumulative correspondant à f. La taille du skyline est alors estimée par $n \times p$. Zhang *et al.* (2009) propose une approche basée sur l'estimation par noyau de la fonction de densité des tuples. L'association de cette méthode au calcul exacte de la taille du skyline d'un échantillon des données permet d'obtenir une estimation de la probabilité p qu'un tuple appartienne au skyline et de ce fait d'estimer tout comme Chaudhuri *et al.* (2006) la taille du skyline. En dépit de ces avantages, c'est-à-dire aucune hypothèse faite sur la nature des données, cette méthode souffre de deux problèmes : (i) elle requière le calcul du skyline d'un échantillon, ce qui peut être couteux ; (ii) intuitivement, les méthodes à noyau font un usage intensif du calcul de distances afin de trouver les plus proches voisins. Lorsque le nombre de dimensions augmente, il est connu, en raison du fléau de la dimensionnalité, que les tuples ont tendance à être difficilement distinguable les uns des autres du point de vue de la distance. Ce qui nécessitera alors des échantillons de grande taille pour garantir une moindre précision des résultats. Comme la méthode basée sur le noyau, Luo *et al.* (2012) propose une méthode non paramétrique. Leur approche consiste à calculer le skyline d'un échantillon de données que l'on appelera le skyline intermédiaire. Ensuite, les données d'origine sont parcourues dans le but d'éliminer du skyline intermédiaire les tuples qui sont dominés par un tuple de l'ensemble des données. Si m est le nombre de tuples restants et s la taille de l'échan-

1. Sans nuire à la généralité, les valeurs des données sont considérées appartenant à l'intervalle $[0, 1]$.

tillon, alors $p = \frac{m}{s}$ est une estimation de la probabilité qu'un tuple appartienne au skyline. Une fois encore, la taille du skyline global est estimée par $n \times p$. Le principal inconvénient de cette approche est le fait que bien qu'il s'agisse d'une approche basée sur l'échantillonnage, l'ensemble des données est requis et il est possible que chaque point soit comparé à tous les autres ce qui dans ce cas est plus coûteux que de calculer directement le skyline et obtenir sa taille.

En outre, certains travaux reposent sur l'estimation de la taille du skyline. Par exemple, Xia *et al*. (2012) traite de l'optimisation des requêtes skyline multidimensionnelles dans le cadre du skycube. Les auteurs proposent une solution pour la selection du meilleur ensemble de *skycuboids* à matérialiser pour optimiser les autres requêtes skyline. Cette solution suppose que les tailles des 2^d skyline sont connues c'est-à-dire estimées avec précision.

3. Estimation de la taille du skyline

Dans cette section, nous présentons nos principales contributions. Nous commençons par la méthode qui requière l'accès aux données, que ce soit le parcours de l'ensemble des données ou juste celui d'un échantillon pour ce faire.

3.1. Parcours des données

3.1.1. Accéder à toutes les données

Nous montrons ici comment après un parcours des données, des statistiques sont conservées ensuite utilisées afin de fournir une estimation sans biais de la taille du skyline. Il est nécessaire de mentionner que cette procédure possède une complexité de l'ordre de $\mathcal{O}(n)$ puisque la première étape tout comme la deuxième étape s'exécute en $\mathcal{O}(n)$. Nous présentons tout d'abord quelques notations.

Considérons les tuples de T comme étant des lignes d'une matrice, i.e., le i^{iem} tuple t_i de T est égal à $\langle T[i,1], T[i,2], \ldots, T[i,d] \rangle$. Soient t_j et t_i tels que $t_j \prec t_i$. Ceci signifie que $t_i[\ell] \leq t_j[\ell]$ pour $1 \leq \ell \leq d$. Étant donné que nous considérons les données normalisées, le domaine de chaque dimension D_ℓ, noté Dom_ℓ, est égal à $\{1, 2, \ldots, k_\ell\}$ où k_ℓ est le nombre de valeurs distinctes de la dimension D_ℓ. Soit $c \in Dom_\ell$ et soit t un tuple aléatoire de T. Notons par $f_\ell(c) = Prob(t[\ell] = c)$ et $F_\ell(c) = Prob(t[\ell] \leq c)$ respectivement la probabilité que $t[\ell] = c$ et la probabilité que $t[\ell] \leq c$. $f_\ell(x)$ est la fonction de distribution des valeurs de D_ℓ et $F_\ell(x)$ est la fonction de distribution cumulée associée à f_ℓ.

EXEMPLE 8. — De notre exemple courant, les fonctions de distribution cumulées des différentes valeurs des attributs sont présentées dans le tableau 4.

□

Soit $v \in \mathcal{T}$ un tuple fixé et soit $P^{\prec}(v)$ la notation représentant la probabilité que v soit dominé par un tuple aléatoire suivant la loi décrite par les distributions F_ℓ. Alors $P^{\prec}(v)$ est donné par le résultat suivant.

Tableau 4. Les fonctions de densité cumulée

x	$F_1(x)$	$F_2(x)$	$F_3(x)$	$F_4(x)$
1	1/4	7/12	1/6	1/3
2	5/6	5/6	7/12	7/12
3	1	1	1	1

LEMME 9. — *Soit $v \in \mathcal{T}$ et soit $t \in \mathcal{T}$ un tuple choisi de façon aléatoire et suivant la loi décrite par les distributions F_ℓ. Alors,*

$$Prob(t \prec v) = P^{\prec}(v) = \prod_{\ell=1}^{d} F_\ell(v[\ell]) - \prod_{\ell=1}^{d} f_\ell(v[\ell])$$

.

PREUVE. — $t \prec v \Leftrightarrow t[\ell] \leq v[\ell]$ pour tout $1 \leq \ell \leq d$. Puisque $Prob(t[\ell] \leq v[\ell]) = F_\ell(v[\ell])$, et les dimensions sont indépendantes, alors $Prob(t \prec v \vee t = v) = \prod_{\ell=1}^{d} F_\ell(v[\ell])$. D'autre part, $Prob(t = v) = \prod_{\ell=1}^{d} f_\ell(v[\ell])$, donc

$$\begin{aligned} Prob(t \prec v) &= Prob(t \prec v \vee t = v) - Prob(t = v) \\ &= \prod_{\ell=1}^{d} F_\ell(v[\ell]) - \prod_{\ell=1}^{d} f_\ell(v[\ell]) \end{aligned}$$

■

LEMME 10. — *Soit $v \in \mathcal{T}$ un tuple et soit $\mathcal{U} = \{u_1, u_2, \ldots, u_m\}$ un ensemble de m tuples indépendants et choisis aléatoirement dans $\mathcal{T}$ suivant les distributions F_ℓ, alors, la probabilité $Prob(v \not\prec u_i, \forall u_i \in \mathcal{U})$ que v ne soit dominé par aucun tuple appartenant à $\mathcal{U}$ est donnée par $(1 - P^{\prec}(v))^m$.*

PREUVE. — À partir du lemme précédent, $Prob(u_i \prec v) = P^{\prec}(v)$, $\forall u_i \in \mathcal{U}$, alors $Prob(u_i \not\prec v) = 1 - P^{\prec}(v)$. En raison de l'indépendance des u_i, la probabilité que v ne soit dominé par aucun $u_i \in \mathcal{U}$ peut être écrite comme suit

$$\begin{aligned} Prob(v \not\prec u_i, \forall u_i \in \mathcal{U}) &= Prob(v \prec u_1, \wedge v \not\prec u_2 \cdots \wedge v \not\prec u_m) \\ &= \prod_{u_i \in \mathcal{U}} Prob(v \not\prec u_i) \\ &= \prod_{u_i \in \mathcal{U}} (1 - P^{\prec}(v)) \\ &= \left(1 - P^{\prec}(v)\right)^m \end{aligned}$$

■

Du lemme précédent, nous déduisons la probabilité qu'un tuple de T appartienne à son skyline.

LEMME 11. — *Soit T un ensemble de tuple tel que $|T| = n$ et soit $t \in T$. Alors $Prob(t \in Sky(T)) = (1 - P^{\prec}(t))^{n-1}$*

PREUVE. — Pour que t appartienne au skyline, il est nécessaire qu'il ne soit dominé par aucun des $n-1$ autres tuples. Chacun de ces derniers possède une probabilité $P^{\prec}(t)$ de dominer t, par conséquent une probabilité égale à $1 - P^{\prec}(t)$ de ne pas dominer t. La probabilité que tous les $n-1$ tuples satisfassent cette condition est égale au produit des probabilités que chacun d'eux la satisfasse ; ce qui prouve le lemme. ■

À présent, nous pouvons construire notre premier estimateur de la taille du skyline. Celui-ci dépend de la probabilité pour un tuple d'appartenir au skyline et de la taille de T.

PROPOSITION 12. —

$$\hat{S}_W = \sum_{i=1}^{n} \left(1 - \prod_{\ell=1}^{d} F_\ell(t_i[\ell]) + \prod_{\ell=1}^{d} f_\ell(t_i[\ell]) \right)^{n-1}$$

est un estimateur sans biais de l'espérance de la taille du skyline d'une table contenant n tuples de d attributs dont les valeurs suivent les distributions cumulées F_l.

PREUVE. — Nous devons montrer que $\hat{S}_W$ est égal à l'estimateur de Horvitz-Thompson (Définition 5) qui est connu pour être sans biais. Ici, la population est l'ensemble de tuples possibles $\mathcal{T}$ tandis que l'échantillon de tuples est l'ensemble des tuples $t \in T$. La variable observée est pour un tuple $t \in \mathcal{T}$, la probabilité pour celui-ci d'appartenir au skyline. Pour qu'un tuple $t \in \mathcal{T}$ appartienne au skyline, il doit tout d'abord faire partie des tuples présents au sein de la table T et ensuite faire partie du skyline de T. Soient q_t et p_t représentant respectivement ces probabilités. Alors $Prob(t \in Sky(T))$ est égale à

$$Prob(t \in T) \times Prob(t \in Sky(T) | t \in T) = q_t \times p_t$$

L'espérance de la variable aléatoire $|Sky(T)|$ est égale à $E(|Sky(T)|) = \sum_{t \in \mathcal{T}} q_t \times p_t$. Alors, l'estimateur de Horvitz-Thompson de $E(|Sky(T)|)$ (dans le cas de la somme) est donné par $\hat{E}(|Sky(T)|)_{HT} = \sum_{i=1}^{n} \pi_i^{-1} q_{t_i} \times p_{t_i}$ où π_i est la probilité que le tuple t_i appartienne à l'échantillon. Dans notre cas, l'échantillon est T d'où $\pi_i = q_{t_i}$. Alors,

Algorithme 1 : S_W, estimation de la taille du skyline

Input :

- number of tuples : n
- number of attributes : d
- data table : $T[1 \dots n, 1 \dots d]$

Output : Skyline estimation : S_W

begin
  for ***each*** ℓ ***in*** $1 \dots d$ **do**
    Compute f_ℓ from T
    Compute F_ℓ from f_ℓ
  $S_W \leftarrow 0$
  for ***each*** i ***in*** $1 \dots n$ **do**
    $S_W \leftarrow S_W + \left(1 - \prod_{\ell=1}^{d} F_\ell(T[i,\ell]) + \prod_{\ell=1}^{d} f_\ell(T[i,\ell])\right)^{n-1}$
  return S_W

$$\begin{aligned}
\hat{E}(|Sky(T)|)_{HT} &= \sum_{i=1}^{n} q_{t_i}^{-1} \times q_{t_i} \times p_{t_i} \\
&= \sum_{i=1}^{n} p_{t_i} \\
&= \sum_{i=1}^{n} \left(1 - \prod_{\ell=1}^{d} F_\ell(t_i[\ell]) + \prod_{\ell=1}^{d} f_\ell(t_i[\ell])\right)^{n-1} \\
&= \hat{S}_W
\end{aligned}$$

■

L'algorithme 1 décrit comment implémenter cette estimation.

EXEMPLE 13. — En appliquant cet algorithme à notre exemple courant et en utilisant les fonctions de densité cumulative résumées dans le tableau 4, nous obtenons $\hat{S}_W = 3,67$ tandis que la taille exacte du skyline est de 3. □

Nous concluons cette section en montrant que $\hat{S}_W$ peut être obtenu en $\mathcal{O}(n)$. En effet, calculer les fonctions de distribution cumulative requière un simple parcours des données. Ensuite nous effectuons la somme de n termes. Il est nécessaire de rappeler que le calcul du skyline est de l'ordre de $\mathcal{O}(n^2)$.

3.1.2. Échantillonnage

Lorsque les données sont volumineuses, il est possible que, même une estimation de la taille du skyline en $\mathcal{O}(n)$ soit considéré comme chronophage. De ce fait, il serait intéressant, au lieu de mobiliser l'ensemble des données, que juste un échantillon puisse permettre de fournir une estimation sans biais. Le résultat suivant répond à cette problématique.

PROPOSITION 14. — *Soit $M \subseteq T$ un échantillon aléatoire de tuples tels que $|M| = m$ et $|T| = n$. Alors*

$$\hat{S}_S = \frac{n}{m} \sum_{t_i \in M} \left(1 - \prod_{\ell=1}^{d} F_\ell(t_i[\ell]) + \prod_{\ell=1}^{d} f_\ell(t_i[\ell]) \right)^{n-1}$$

est un estimateur sans biais de la taille du skyline.

PREUVE. — La probabilité (π_i) qu'un tuple $t_i \in \mathcal{T}$ appartienne à l'échantillon est égale au produit des probabilités que t_i appartienne à la table T (probabilité notée q_i) et que t_i appartienne à l'échantillon sachant qu'il appartient à la table $\frac{\binom{n-1}{m-1}}{\binom{n}{m}} = \frac{m}{n}$, où $\binom{n}{m}$ est le nombre total d'échantillons possibles constitués de m tuples tirés de la table T, et $\binom{n-1}{m-1}$ est le nombre de ces échantillons dans lesquels t_i apparaît. Alors, $\pi_i = q_i \times \frac{m}{n}$.

L'estimateur de Horvitz-Thompson donnant la taille du skyline est alors

$$\begin{aligned}
\hat{E}(|Sky(T)|)_{HT} &= \sum_{t_i \in M} (\pi_i)^{-1} \times q_{t_i} \times p_{t_i} \\
&= \sum_{t_i \in M} (q_{t_i} \times \frac{m}{n})^{-1} \times q_{t_i} \times p_{t_i} \\
&= \frac{n}{m} \sum_{t_i \in M} p_{t_i} \\
&= \frac{n}{m} \sum_{t_i \in M} \left(1 - \prod_{\ell=1}^{d} F_\ell(t_i[\ell]) + \prod_{\ell=1}^{d} f_\ell(t_i[\ell]) \right)^{n-1} \\
&= \hat{S}_S
\end{aligned}$$

■

En pratique, f et F peuvent respectivement être estimés par f^M et F^M qui représentent la distribution obtenue au sein de l'échantillon M.

L'algorithme 2 montre comment l'estimation de la taille du skyline est déterminée en ne parcourant qu'un échantillon des données.

Algorithme 2 : S_S, estimation de la taille du skyline

Input :

- number of tuples : n
- size of sample : m
- number of attributes : d
- sample : $T[1 \dots m, 1 \dots d]$

Output : Skyline estimation : S_S

begin
  for ***each*** ℓ ***in*** $1 \dots d$ **do**
    Compute f_ℓ from T
    Compute F_ℓ from f_ℓ
  $S_S \leftarrow 0$
  for ***each*** i ***in*** $1 \dots m$ **do**
    $S_S \leftarrow S_S + \left(1 - \prod_{\ell=1}^{d} F_\ell(T[i,\ell]) + \prod_{\ell=1}^{d} f_\ell(T[i,\ell])\right)^{n-1}$
  $S_S \leftarrow S_S \times n/m$
  return S_S

EXEMPLE 15. — Supposons, à partir de notre exemple courant, que nous sélectionnions un échantillon M de 4 tuples $\{h_5, h_8, h_{11}, h_{12}\}$. Nous calculons la distributions des données au sein de M. À partir de ce résultat intermédiaire et de l'échantillon nous estimons la taille du skyline et obtenons $\hat{S}_S = 3,44$ tandis que la taille exacte du skyline est 3, rappelons qu'en faisant usage de l'ensemble des données nous avions obtenu $\hat{S}_W = 3,67$. □

3.2. Espérance de la taille du skyline

Dans cette partie, nous présentons une manière d'estimer la taille du skyline lorsqu'on ne dispose que de la connaissance de la distribution des valeurs des données. Une fois de plus, nous faisons recours à l'estimation de l'espérance de la taille du skyline. Plus précisement, soit $\mathcal{T}(n, \{F_\ell\}, d)$ l'ensemble de toutes les tables T constituées de tuples dont les valeurs de chaque dimension D_ℓ suivent la loi définie par F_ℓ. Alors la valeur moyenne des tailles de skyline des éléments de $\mathcal{T}(n, \{F_\ell\}, d)$ est donnée par $\frac{1}{|\mathcal{T}(n,\{F_\ell\},d)|} \sum_{T \in \mathcal{T}(n,\{F_\ell\},d)} |Sky(T)|$. Cette information donne une idée de la taille du skyline des tables présentant ce profil.

A présent, soit $Sky_{\mathcal{T}}$ la variable aléatoire correspondant à la taille du skyline des tables $T \in \mathcal{T}(n, \{F_\ell\}, d)$. Alors, l'espérance de $Sky_{\mathcal{T}}$ représente la taille moyenne que nous cherchons. Nous présentons une formule précise pour les cas où d est grand.

Nous nous restreignons au cas où l'ensemble des dimensions présentent la même distribution, c'est-à-dire $\forall i, j, k_i = k_j$ que nous noterons désormais k et $F_i = F_j$ qui sera noté F.

En raisons du fait que $\hat{S}_W$ est un estimateur sans biais de $E(Sky_{\mathcal{T}})$, nous avons $E(\hat{S}_W) = E(Sky_{\mathcal{T}})$. De ce fait, estimer $E(\hat{S}_W)$ est équivalent à estimer $E(Sky_{\mathcal{T}})$; ce que nous developpons par la suite.

Nous considérons la table T comme une matrice où chaque cellule $t_i[j]$ est vu comme la réalisation du variable aléatoire T_{ij} dont la fonction de densité cumulée est F. Soient $X_i = \prod_{j=1}^{d} F(T_{ij})$ et $Z_i = (1 - X_i)^{n-1}$ deux ensembles de variables aléatoires.

Notre principal résultat peut être énoncé comme suit :

PROPOSITION 16. — *Soient X et Y deux variables aléatoires telles que $Y = log(X)$ suit la loi normale $\mathcal{N}(\mu, \sigma^2)$. Alors, lorsque $d \to \infty$, la variable aléatoire Z_i converge en loi vers $(1 - X)^{n-1}$ et alors $E(Sky_{\mathcal{T}})$ peut être estimé par $\hat{S}_E = n \times E\left[(1 - X)^{n-1}\right]$ où*

$$\mu = d\sum_{\ell=1}^{k} f(\ell) log(F(\ell))$$

et $\sigma^2 = d\left[\sum_{\ell=1}^{k} f(\ell)\left(log(F(\ell))\right)^2 - \left(\sum_{\ell=1}^{k} f(\ell) log(F(\ell))\right)^2\right]$

PREUVE. — Soient $U_j = \log(F(T_{ij}))$ des variables aléatoires définies sur $\{\log(F(1)), \log(F(2)), \ldots, \log(F(k))\}$. Puisque les variables U_j sont indépendantes et identiquement distribuées, nous notons μ_U leur espérance et σ_U^2 leur variance. L'espérance de U_j est égale à $\mu_U = E(U_j) = \sum_{\ell=1}^{d} f(\ell) \times \log(F(\ell))$ tandis que la variance de U_j est égale à :

$$\begin{aligned}
\sigma_U^2 &= V(U_j) \\
&= E(U - E(t)) = E(U^2) - E(U)^2 \\
&= \sum_{\ell=1}^{k} f(\ell)\left(log(F(\ell))\right)^2 - \left(\mu_U\right)^2
\end{aligned}$$

Soit $Y_i = \log(X_i)$ une variable aléatoire. Puisque $X_i = \prod_{j=1}^{d} F(T_{ij})$ alors $Y_i = \prod_{j=1}^{d} \log(F(T_{ij})) = \sum_{j=1}^{d} U_j$. Etant donné que les variables aléatoires Y_i sont indépen-

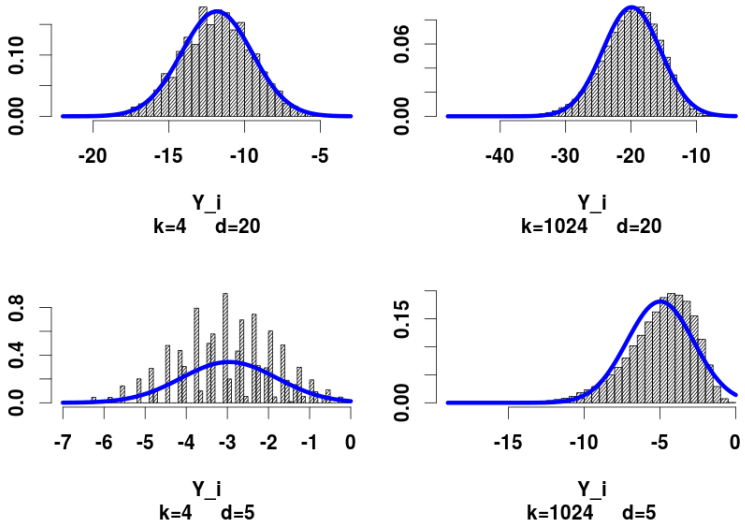

Figure 1. Distribution des $Y_i = \sum_{j=1}^{d} U_j$

dantes et identiquement distribuées, nous notons μ_Y et σ_Y^2 respectivement leur espérance et leur variance. L'espérance de Y_i est égale à $\mu_Y = E(Y_i) = \sum_{j=1}^{d} E(U_j) = d \times \mu_U$ et la variance de Y_i est égale à $\sigma_Y^2 = V(Y_i) = \sum_{j=1}^{d} V(U_j) = d \times \sigma_U^2$.

Étant donné que Y_i représente une somme de variables aléatoires indépendantes et identiquement distribuées, nous lui appliquons le *Théorème Central Limite*. Alors, lorsque d devient assez grand, Y_i suit la loi normale $Y_i \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$. Cette convergence est illustrée dans la figure 1 où, dépendant des valeurs de k et de d, différents cas sont représentés. Pour k et d fixés, l'histogramme de 10000 valeurs de Y_i est tracé, cet histogramme est comparé à la courbe de la densité de la loi normale correspondante. La figure 1 montre qu'au fur et à mesure que k et/ou d augmentent, la convergence se précise.

D'après le *Théorème Central Limite*, lorsque d augmente, $Y_i = \log(X_i)$ converge vers la loi normale $\mathcal{N}(\mu_Y, \sigma_Y^2)$.

Soit Y une variable aléatoire telle que Y_i converge en loi vers Y lorsque $d \to \infty$ ce qui se note $Y_i \xrightarrow{\mathcal{L}} Y$. Il est établi que $Y \sim \mathcal{N}(d\mu_U, d\sigma_U^2)$. Soit X une variable

aléatoire telle que $Y = \log(X)$. Alors, lorsque $d \to \infty$, X_i converge vers X. D'où $Z = (1 - X_i)^{n-1}$ converge vers $(1 - X)^{n-1}$.

Partant de la définition de $\hat{S}_W$ (voir Proposition 12), nous avons $\hat{S}_W = \sum_{i=1}^{n} \left(1 - \prod_{\ell=1}^{d} F_\ell(t_i[\ell]) + \prod_{\ell=1}^{d} f_\ell(t_i[\ell])\right)^{n-1}$. Le terme $\prod_{\ell=1}^{d} f_\ell(t_i[\ell])$ correspond à la probabilité pour deux tuples d'être égaux, mais au fur et à mesure que d augmente, cette probabilité tend à s'annuler ; alors $\hat{S}_W = \sum_{i=1}^{n} \left(1 - \prod_{\ell=1}^{d} F_\ell(t_i[\ell])\right)^{n-1} = \sum_{i=1}^{n} Z_i$, cependant, Z_i converge en loi vers $(1 - X)^{n-1}$, de ce fait, $E(\hat{S}_W) = \sum_{i=1}^{n} E((1 - X)^{n-1}) = n \times E\left((1 - X)^{n-1}\right)$.

$E(Sky_{\mathcal{T}}) = E(\hat{S}_W)$ puisque $\hat{S}_W$ est sans biais ; alors, lorsque d devient grand, $\hat{S}_E = n \times E\left[(1 - X)^{n-1}\right]$ est une une bonne estimation de l'espérance de la taille du skyline $E(Sky_{\mathcal{T}})$. ■

Nous n'avons jusque-là uniquement montré que $E(Sky_{\mathcal{T}})$ pourrait, sous certaines conditions, être assimilé à $nE[(1 - X)^{n-1}]$. D'un point de vue pratique, ce n'est pas suffisant pour obtenir une estimation de la taille du skyline. En effet, il est nécessaire d'estimer l'espérance de la variable aléatoire $Z = (1 - X)^{n-1}$. Par définition, cette espérance est égale à $E(Z) = \int_0^1 z f_Z(z) \cdot dz$ où f_Z est la fonction de densité associée à Z.

Nous utilisons la *méthode des rectangles* pour estimer $E(Z)$. Soit N le nombre d'intervalles utilisés pour estimer $E(Z)$. L'estimation de $E(Z)$ est alors

$$\hat{E}(Z) = \sum_{\alpha=1}^{N} \frac{\alpha - \frac{1}{2}}{N} \left[F_Z\left(\frac{\alpha}{N}\right) - F_Z\left(\frac{\alpha - 1}{N}\right)\right]$$

.

où $F_Z(z) = Prob(Z \leq z)$ est la fonction de distribution cumulée relative à Z. Nous devons déterminer F_Z.

$$\begin{aligned} F_Z(z) &= Prob(Z \leq z) = Prob\left[(1 - X)^{n-1} \leq z\right] \\ &= Prob\left[X \geq 1 - \sqrt[n-1]{z}\right] \\ &= 1 - Prob\left[Y < log\left(1 - \sqrt[n-1]{z}\right)\right] \\ &= 1 - F_Y\left[log\left(1 - \sqrt[n-1]{z}\right)\right] \end{aligned}$$

Où F_Y est la fonction de densité cumulée relative à Y. Étant donné que Y suit une loi normale, la fonction F_Y peut être facilement accessible puisqu'elle est largement implémentée en informatique.

L'algorithme 3 montre une façon d'implémenter le calcul de l'estimation de l'espérance de la taille du skyline conformement à la méthode présentée. Cet algorithme a une complexité de l'ordre de $\mathcal{O}(d \times k + N)$, sachant que N est choisi par l'utilisateur.

Algorithme 3 : S_E, estime la taille du skyline

Input :
- number of tuples : n
- number of attributes : d
- number of distinct values per dimension: k
- distribution function : $f(quantile)$
- cumulative distribution functions : $F(quantile)$
- number of intervals : N
- normal cumulative distribution functions : $F_Z(quantile, \mu, \sigma^2)$

Output : Skyline estimation : S_E

begin
- $\mu \leftarrow d \times \sum_{j=1}^{k} f(j) log(F(j))$
- $\sigma^2 \leftarrow d \left[\sum_{j=1}^{k} f(j) \left(log(F(j))\right)^2 - \left(\sum_{j=1}^{k} f(j) log(F(j)) \right)^2 \right]$
- $S_E \leftarrow 0$
- **for** ***each*** α ***in*** *1... N* **do**
  - $S_E \leftarrow S_E + \frac{\alpha - \frac{1}{2}}{N} \left[F_Z\left(\frac{\alpha}{N}, \mu, \sigma^2\right) - F_Z\left(\frac{\alpha - 1}{N}, \mu, \sigma^2\right) \right]$
- $S_E \leftarrow S_E \times n$
- **return** S_E

EXEMPLE 17. — Toujours à partir de notre exemple courant, nous calculons l'estimation de l'espérance de la taille du skyline pour différentes valeurs de N en suivant l'algorithme présenté précédemment. Nous avons également calculé l'estimation de l'espérance de la taille du skyline en réalisant la moyenne des tailles de skyline de 10000 jeux de données générés suivant le même format que celui de notre exemple ($k = 3$, $d = 4$, $n = 12$) ; cette méthode est proche de la méthode *bootstrap* de Efron (1979) utilisée pour l'estimation l'espérance d'un paramétre au sein d'une population. Le résultat est présenté dans le tableau 5.

Tableau 5. Estimation de l'espérance de la taille du skyline

$N = 8$	$N = 16$	$N = 32$	$N = 32768$	10000 jeux de données
$3,00$	$2,89$	$2,85$	$2,82$	$3,59$

□

4. Expérimentations

Dans cette section, nous comparons nos méthodes avec l'algorithme Purely Sampling (PS) de Luo *et al*. (2012). Nous ne considérons pas l'algorithme Kernel Based (KB) proposé par Zhang *et al*. (2009) parce qu'il a été montré dans Luo *et al*. (2012) que KB est plus lent et moins précis que PS. Dans leurs expérimentations, Luo *et al*. (2012) choisissent des échantillons de taille 5 %, 10 % et 15 % de n. Dans nos configurations, nous considérons plutôt des échantillons de taille $\sqrt{n}$ parce que PS possède une complexité dans le pire des cas de l'ordre de $\mathcal{O}(m^2 + s_1 n)$ où m est la taille de l'échantillon et s_1 la taille du skyline de l'échantillon ; tandis que la complexité de l'algorithme $\hat{S}_W$ est de l'ordre de $O(n)$. En prenant $m = \sqrt{n}$, les deux algorithmes deviennent de ce fait comparables du point de vue du temps d'exécution.

Tous les algorithmes comparés ont été implémentés en C++ en utilisant l'outil Code::Blocks pour l'implémentation. Lesdites expérimentations ont été faites sur un ordinateur possédant un processeur Intel Core i7 - 2600 CPU @ 3,4 GHz x 8, une mémoire RAM de 7,8 GB et 500 GB de disque dur ; le tout sous le système d'exploitation Ubuntu 64 bit.

4.1. Parcours des données

Nous avons exécuté tous ces algorithmes sur plusieurs jeux de données synthétiques obtenus en faisant varier trois paramètres : la taille des données (nombre de tuples n), le nombre de dimensions d et le nombre de valeurs distinctes par dimension k_{max}. Une fois que k_{max} est fixé pour un jeu de données, chaque k_j prend une valeur aléatoire comprise entre 4 et k_{max}. En ce qui concerne la distribution de valeurs par dimension, nous choisissons d'attribuer des distributions différents pour les dimensions. Plus précisement, nous adoptons le processus suivant :

– poids uniformes : chaque valeur apparait en moyenne n/k_j fois dans la dimension ;

– poids proportionnels aux valeurs : les petites valeurs apparaissent moins souvent que les grandes valeurs ;

– poids inversement proportionnels aux valeurs : l'opposé de la précédente distribution.

Chaque dimension est générée aléatoirement suivant l'une des procédures précédentes. Nous avons généré 48 jeux de données avec les paramètres suivants :

– $n \in \{1024, 1048576\}$

– $d \in \{10, 12, 14, 16, 18, 20\}$

– $k_{max} \in \{16, 64, 256, 1024\}$

La figure 2 trace les valeurs absolues de tailles de skyline (EXACT) et les estimations ($\hat{S}_W$, $\hat{S}_S$ and PS). Lorsque d augmente, aussi bien que lorsque n augmente, toutes les méthodes semblent être plus précises. De l'ensemble des courbes, PS semble

présenter le plus de fluctuations, suivi de $\hat{S}_S$, de ce fait, $\hat{S}_W$ est la méthode la plus précise d'entre toutes. Nous choisissons de mieux observer cette comparaison à travers le graphique des erreurs.

La figure 3 présente les erreurs relatives des différentes méthodes $\hat{S}_W$, $\hat{S}_S$, $\hat{S}_E$ et PS. L'erreur relative est calculée suivant la formule $\frac{|\widehat{Sky} - EXACT|}{EXACT} \times 100$ où $\widehat{Sky}$ est une estimation. Notons que la façon dont les données ont été générées ne correspond pas exactement aux hypothèses dans lesquelles l'estimateur $\hat{S}_E$ converge. En effet, ici les dimensions présentent différentes cardinalités.

Nous observons qu'en général, $\hat{S}_W$ et $\hat{S}_S$ sont plus précis que PS. La plage de variation de l'erreur de l'estimation PS autour de zéro est plus élevée que celle de $\hat{S}_W$. Lorsque d augmente aussi bien que lorsque n augmente l'erreur relative maximale commise diminue.

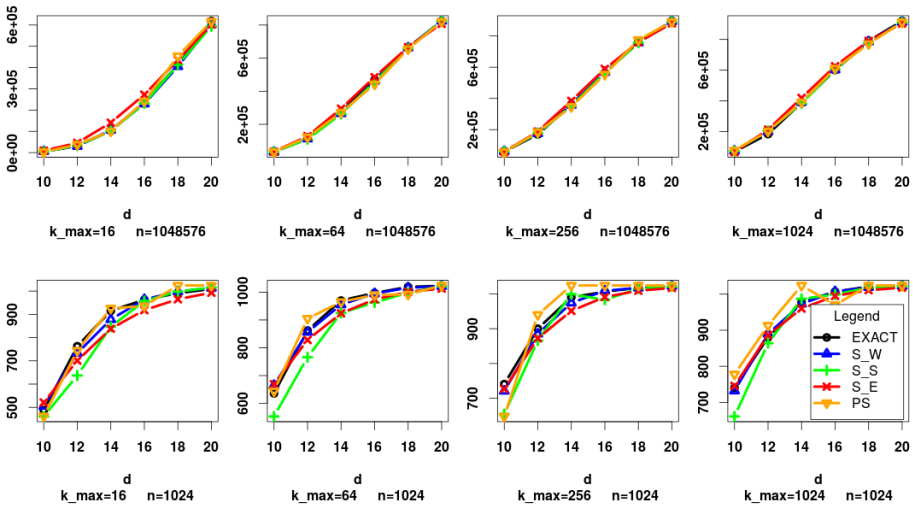

Figure 2. La taille du skyline en fonction du nombre de dimensions

4.2. Espérance de la taille du skyline

Dans la seconde partie des expérimentations, nous comparons notre méthode à la méthode bootstrap de Efron (1979), toutes les deux estimant l'espérance de la taille du skyline. Nous générons les données suivant différents formats :

- $n \in \{2^{10}, 2^{12}, \ldots, 2^{20}\}$
- $d \in \{10, 14\}$

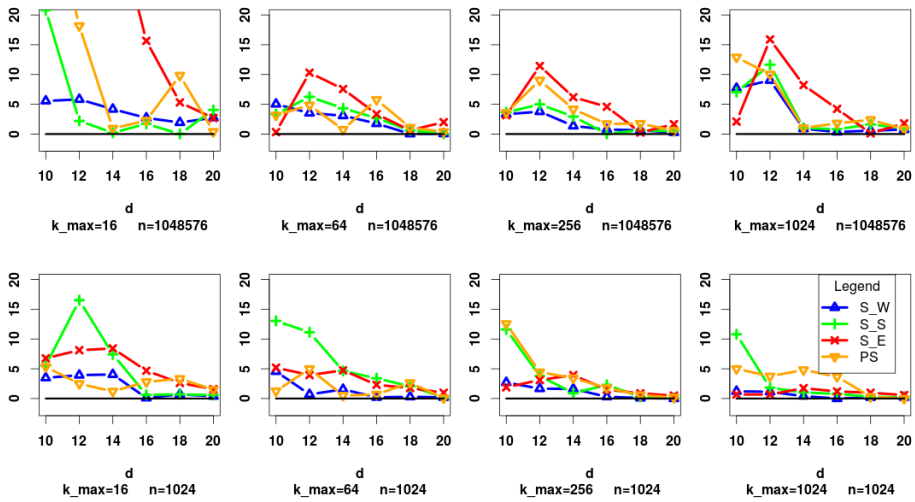

Figure 3. L'erreur relative en fonction du nombre de dimensions

– $k \in \{16, 64, 256, 1024\}$

Les valeurs au sein des dimensions sont uniformement distribuées. pour un triplet (n, d, k) fixé, nous estimons l'espérance de la taille du skyline par la méthode que nous proposons, ensuite pour la méthode bootstrap, nous générons 10 jeux de données dont nous calculons la moyenne des tailles de skyline. Il est nécessaire de souligner que la méthode bootstrap est sans biais, ce qui en fait un estimateur fiable auquel se comparer.

La figure 4 montre les résultats d'approximations, nous présentons également les valeurs minimale et maximale de taille de skyline parmi les jeux de données de même format. Nous constatons que comme on l'aurait espéré, lorsque d croît, l'estimateur s'améliore.

4.3. Temps d'exécution

Dans cette section, nous comparons les différentes méthodes du point de vue du temps d'exécution. Ceci présente un intérêt parce que si l'estimation requière un temps comparable à celui du calcul exact alors celle-ci ne sera d'aucune utilité. Les résultats obtenus sont présentés dans la figure 5. Le principal résultat de cette expérimentation est que PS ne passe pas à l'échelle lorsque n et d augmentent, ceci en raison du calcul

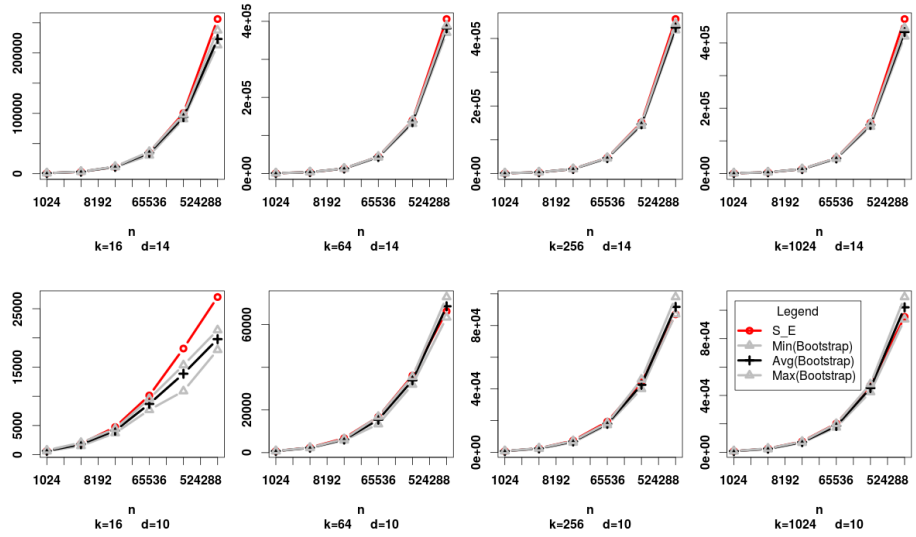

Figure 4. Estimation de l'espérance de la taille du skyline en fonction de la taille des données

d'un skyline dans sa procédure ; bien que nous ayons réduit la taille de l'échantillon (passant de 5 % de n à $\sqrt{n}$).

L'algorithme le plus rapide est $\hat{S}_S$. Par exemple lorsque $k = 1024$, $n = 1048576$, alors $\hat{S}_S$ requière 3 millisecondes tandis que $\hat{S}_W$ et $\hat{S}_E$ requièrent environ 200 millisecondes. Notons dans ce cas que le calcul exact demande 50 minutes.

Indépendamment de k, d et n les méthodes $\hat{S}_W$ et $\hat{S}_E$ semblent être équivalentes en terme de temps d'exécution ceci parce que nous avons choisi le paramètre N, nombre d'intervalles pour l'estimation de l'espérance, égal à n ($N = n$).

5. Conclusion

Ce papier propose des estimateurs sans biais pour le calcul de la taille du skyline basés sur la connaissance de la distribution des données. L'idée maitresse consiste au parcours des données (en totalité ou en partie) pour estimer la probabilité p qu'un tuple appartienne au skyline. Une fois cette probabilité estimée, la taille du skyline se déduit par le produit $p \times n$. Plus encore, connaissant la distribution des données, nous proposons également un estimateur de l'espérance de la taille du skyline pour toutes les tables respectant la distribution en question. Bien que le dernier estimateur

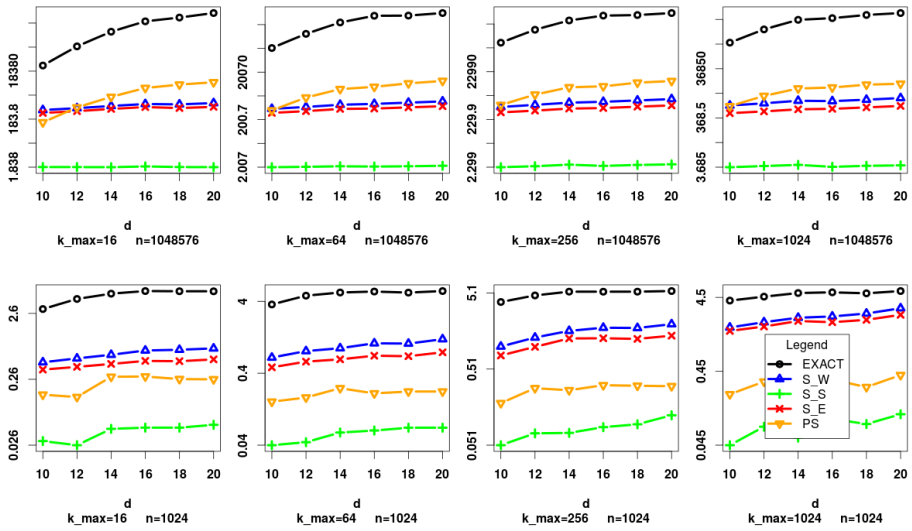

Figure 5. Temps d'exécution des différents algorithmes

soit biaisé, nous avons montré que c'est un bon estimateur de l'espérance de la taille du skyline.

Nous avons comparé nos propositions à la méthode PS de l'état de l'art. Les expérimentations faites montrent que nos solutions offrent de meilleurs résultats des points de vue précision et temps d'exécution. Plus précisement, configurer PS afin qu'il soit plus compétitif en termes de temps d'exécution diminuera la précision de ses résultats.

Nous prévoyons dans les futurs travaux se passer de la contrainte d'indépendance des dimensions. Pour ce faire, nous envisageons l'exploitation des distributions jointes des dimensions ou encore analyser les corrélations entre les dimensions.

Remerciements

Ce travail a été partiellement réalisé dans le cadre des projets PETASKY *et* SPEED DATA *financés respectivement par l'initiative* MASTODONS *du CNRS et le Programme d'investissements d'avenir (PIA).*

Bibliographie

Bartolini I., Ciaccia P., Patella M. (2008). Efficient sort-based skyline evaluation. *ACM Trans. Database Syst.*, vol. 33, n° 4.

Bentley J. L., Kung H. T., Schkolnick M., Thompson C. D. (1978, octobre). On the average number of maxima in a set of vectors and applications. *J. ACM*, vol. 25, n° 4, p. 536–543.

Börzsönyi S., Kossmann D., Stocker K. (2001). The skyline operator. In *Proceedings of the 17th international conference on data engineering*, p. 421–430. Washington, DC, USA, IEEE Computer Society.

Buchta C. (1989, novembre). On the average number of maxima in a set of vectors. *Inf. Process. Lett.*, vol. 33, n° 2, p. 63–65.

Chaudhuri S., Dalvi N., Kaushik R. (2006). Robust cardinality and cost estimation for skyline operator. In *Proceedings of the 22nd international conference on data engineering*, p. 64–. Washington, DC, USA, IEEE Computer Society.

Chomicki J., Godfrey P., Gryz J., Liang D. (2003). Skyline with presorting. In *Proc. of ICDE conference.*

Efron B. (1979, 01). Bootstrap methods: Another look at the jackknife. *Ann. Statist.*, vol. 7, n° 1, p. 1–26.

Godfrey P., Seipel D., Turull-Torres J. (2004). *Skyline cardinality for relational processing: how many vectors are maximal?*. Rapport technique. York Univ., Toronto, Ont., Canada.

Horvitz D. G., Thompson D. J. (1952, December). A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, vol. 47, n° 260, p. 663–685.

Lee J., Hwang S. won. (2010). BSkyTree: scalable skyline computation using a balanced pivot selection. In *Proc. of EDBT conf.*

Luo C., Jiang Z., Hou W.-C., He S., Zhu Q. (2012). A sampling approach for skyline query cardinality estimation. *Knowledge and Information Systems*, vol. 32, n° 2, p. 281-301.

Morse M. D., Patel J. M., Jagadish H. V. (2007). Efficient skyline computation over low-cardinality domains. In *Proceedings of VLDB conf.*

Xia T., Zhang D., Fang Z., Chen C. X., Wang J. (2012). Online subspace skyline query processing using the compressed skycube. *ACM TODS*, vol. 37, n° 2.

Zhang Z., Yang Y., Cai R., Papadias D., Tung A. (2009). Kernel-based skyline cardinality estimation. In *Proceedings of the 2009 acm sigmod international conference on management of data*, p. 509–522. New York, NY, USA, ACM.