HASE: A Hybrid Approach to Selectivity Estimation for Conjunctive Predicates

Xiaohui Yu[1], Nick Koudas[1], and Calisto Zuzarte[2]

[1] Department of Computer Science
University of Toronto
Toronto, ON, M5S 3G4, Canada
xhyu, koudas@cs.toronto.edu
[2] IBM Toronto Lab, 8200 Warden Avenue
Markham, ON, L6G 1C7, Canada
calisto@ca.ibm.com

Abstract. Current methods for selectivity estimation fall into two broad categories, *synopsis-based* and *sampling-based*. Synopsis-based methods, such as histograms, incur minimal overhead at query optimization time and thus are widely used in commercial database systems. Sampling-based methods are more suited for ad-hoc queries, but often involve high I/O cost because of random access to the underlying data. Though both methods serve the same purpose of selectivity estimation, their interaction in the case of selectivity estimation for conjuncts of predicates on multiple attributes is largely unexplored. Our work aims at taking the best of both worlds, by making consistent use of synopses and sample information when they are both present. To achieve this goal, we propose HASE, a novel estimation scheme based on a powerful mechanism called *generalized raking*. We formalize selectivity estimation in the presence of single attribute synopses and sample information as a constrained optimization problem. By solving this problem, we obtain a new set of weights associated with the sampled tuples, which has the nice property of reproducing the known selectivities when applied to individual predicates. We discuss different variants of the optimization problem and provide algorithms for solving it. We also provide asymptotic error bounds on the estimate. Extensive experiments are performed on both synthetic and real data, and the results show that HASE significantly outperforms both synopsis-based and sampling-based methods.

1 Introduction

Query optimizers in most relational database systems rely on cost estimation of various candidate query execution plans to select a good one. Accurate plan costing can help avoid intolerably slow plans. A key ingredient in cost estimation is to estimate the selectivity of various predicates. In this paper, we are mainly concerned with selectivity estimation for conjunctive predicates of the form $Q = P_1 \wedge P_2 \ldots P_m$ where each component P_i is a simple predicate on a single attribute, taking the form of (*attribute* `op` *constant*) with `op` being one of the comparison operators $<, \leq, =, \neq, \geq$, or $>$ (e.g., $R.a = 100$ or $R.a \leq 200$).

In terms of methodology, existing work on selectivity estimation takes two fundamentally different approaches: one is based on synopsis data structures and the other is based on sampling. Synopsis-based approaches seek to pre-compute summary data structures which capture statistics on the data (attribute value distributions). Such synopses are stored in the database catalogs, and subsequently used for estimation when required. A prominent example in this class of approaches is histograms, which have received heavy attention; numerous types of histograms [1, 2] have been proposed in recent years aiming to improve the accuracy of histogram-based selectivity estimation. Almost all major commercial database management systems (e.g., IBM® DB2® Universal Database™ product(DB2 UDB), Oracle, SQL Server) keep some form of histograms in their catalogs and use them for selectivity estimation.

Sampling-based approaches are more query-driven in nature, in the sense that data is not accessed until optimization time. Given a query, a sample is derived from the database, and selectivities are estimated based on this sample. There exists an extensive literature on sampling-based methods for selectivity estimation; see [3] for a comprehensive survey. In recent years, all of the major commercial database system vendors have incorporated sampling capabilities into their engines [4].

Both approaches have their advantages and disadvantages. Synopsis structures, such as histograms, only need to be computed once and can be used many times while incurring minimal overhead at selectivity estimation time. However, it is difficult to capture all useful information in the limited space. For example, the one-dimensional histograms commonly used in the commercial DBMS's do not provide correlation information between attributes. Although it is possible to compute multi-dimensional histograms for some attribute combinations, it is generally not feasible to compute and store the multi-dimensional histograms for all attribute combinations, because the number of combinations is exponential in the number of attributes [5]. Without knowing of the query workload, deciding which combinations of attributes to choose in order to construct multi-dimensional histograms can be very difficult. Sampling approaches, on the other hand, are able to provide such crucial information through a representative sample of the data. The downside, however, is that sampling at selectivity estimation time incurs non-trivial cost, because in order to obtain a fairly accurate estimate, sometimes a significant portion of the data might have to be accessed. Since sampling requires random access, which is much slower than sequential access, it is possible that the cost of sampling exceeds that of a sequential scan of the data when the sample size is relatively large. (Haas et al. [4] show that under certain assumptions, the cost of sampling is greater than that of sequential scan when the sample rate is greater than 2% and tuple-level sampling is used.)

To the best of our knowledge, there is no previous work exploring the interaction of these two approaches in order to make consistent use of both sources of information. This paper represents a first step in this direction. In particular, we propose HASE (A Hybrid Approach to Selectivity Estimation), a novel method based on the powerful generalized raking procedure originally deployed in the context of survey sampling. Sampling-based methods usually associate with each sampled tuple a *sampling weight* reflecting its inclusion probability (i.e., the probability of being selected to the sample), which is used to produce

a selectivity estimate. Given selectivities of individual predicates P_i (which can be easily obtained from attribute synopses) in addition to the sample, we aim to obtain better estimates by adjusting sampling weights, in a way that is consistent with the information on individual selectivities obtained from the synopses. In particular, we adjust the weights of the tuples in the sample, while maintaining the new weights as close as possible to the original weights. We formalize this problem as a constrained optimization problem. Its solution derives the new weights that can then be used to obtain improved selectivity estimates.

We present a general numerical solution to this optimization problem, as well as an iterative solution based on the intrinsic structure of the problem. We consider two different measures of "closeness" between the new weights and the original weights, namely the linear distance function and the multiplicative distance function, and compare them in terms of computational efficiency and interpretability. We also provide asymptotic bounds on the estimation errors.

The rest of this paper is organized as follows. In Section 2, we formally define the problem of selectivity estimation for conjunctive predicates, and describe how selectivity estimates are obtained in existing approaches. Section 3 presents HASE, our proposed approach based on generalized raking. Experimental results on both synthetic and real data sets are presented in Section 4. We briefly review existing approaches to selectivity estimation in Section 5. Section 6 concludes this paper and discusses directions for future work.

2 Background

In this section, we formally define the problem of selectivity estimation for conjunctive predicates and discuss two existing ways of conducting the estimation, one based on synopses and one on sampling.

2.1 Problem Definition

We are interested in predicates taking the form of $Q = P_1 \wedge P_2 \wedge \cdots \wedge P_m$, where each $P_i (1 \leq i \leq m)$ is a simple predicate of the form (*attribute* `op` *constant*) with `op` being one of the comparison operators $<, \leq, =, \neq, \geq$, or $>$. The selectivity $s_i (\in [0, 1])$ is defined as the fraction of tuples on which predicate P_i evaluates to true, i.e., $s_i = N_i/N$, where N is the number of tuples in the table, and N_i is the number of tuples satisfying P_i. The selectivity of the conjuncts of predicates Q, denoted by $s_Q (\in [0, 1])$, is the fraction of tuples satisfying all the P_i's simultaneously. s_Q is the quantity we would like to estimate. When there is no ambiguity, we use s as a shorthand for s_Q.

We measure the error of an estimate $\hat{s}$ by the *absolute relative error*

$$E(\hat{s}) = \frac{|\hat{s} - s|}{s}. \tag{1}$$

Throughout the paper, we use the following scenario as a running example. Consider a table R with $N = 10,000$ tuples and three attributes $A_i (i = 1, 2, 3)$. Let $P_1 = (A_1 = 1)$, and $P_2 = (A_2 = 1)$. Suppose we need to estimate the selectivity of the following query: $Q = P_1 \wedge P_2$. If there are 500 tuples satisfying Q, then the true selectivity of Q is $s = 500/10000 = 0.05$.

2.2 Synopsis-based estimation

Assume that we have access to synopsis structures for all individual attributes involved such that selectivity estimates $s_i (1 \leq i \leq m)$ can be obtained. Without any information regarding the correlation between attributes, optimizers in current database systems estimate s_Q based on the assumption that the values in distinct attributes are independently distributed. In other words, knowing that a tuple satisfies a predicate on one attribute does not give any information as to whether it satisfies a predicate on another. Therefore, s is estimated by taking a product of the selectivity estimates of individual predicates, i.e., $\hat{s}_{\text{his}} = \prod_{i=1}^{m} s_i$.

In the running example, suppose we have access to single-attribute histograms on A_1 and A_2, and therefore we can derive the selectivities of the two predicates, namely s_1 and s_2, from the histograms. Suppose $s_1 = 0.6$, and $s_2 = 0.3$. If we assume A_1 and A_2 are independent, then the selectivity of Q is estimated to be $\hat{s}_{\text{his}} = s_1 \cdot s_2 = 0.18$, and the error is $E(\hat{s}_{\text{his}}) = |0.18 - 0.05|/0.05 = 260\%$.

This simple estimation scheme gives accurate estimates when the attributes are indeed independent. Real-life data sets, however, almost always demonstrate a certain degree of correlation between attributes; therefore, making the attribute-value independence assumption often leads to erroneous estimates. In the above example, treating the attributes A_1 and A_2 as independent incurs a large error (260%). As another example, suppose we have the following query on a CAR table in a vehicle information database: Q = (MAKE = "BMW")$\wedge$(MODEL = "M3"), and we know through one-dimensional histograms that the selectivity of the predicate (MAKE = "BMW") is 0.1, and that the predicate (MODEL = "M3") has a selectivity of 0.01. The optimizer then would estimate the selectivity of Q as $0.1 \times 0.01 = 0.001$, as per the attribute-value independence assumption. Note, however, that there is strong correlation between the attributes MAKE and MODEL. Because M3 is exclusively made by BMW, all tuples satisfying the predicate MODEL="M3" would also satisfy the predicate MAKE="BMW". Therefore, the selectivity of Q is actually 0.01, 10 times that of the estimated selectivity.

2.3 Sampling-based estimation

Now let us look at how to obtain an estimate of the selectivity based on a sample of the data. Suppose a random sample S of size n is taken from the queried table R of size N, where the *inclusion probability* (the probability of being selected into the sample) of the j-th tuple is π_j. The Horvitz-Thompson (HT) estimator [6] for the selectivity of the query Q, given the sample S, is

$$\hat{s}_{\text{spl}} = \frac{1}{N} \sum_{j \in S} \frac{y_j}{\pi_j} \tag{2}$$

where y_j is an indicator variable such that $y_j = 1$ if tuple j satisfies Q, and $y_j = 0$ otherwise. In the case of simple random sampling (SRS), where the inclusion probabilities are all equal to n/N, Eq. (2) simplifies to $\hat{s}_{\text{spl}} = \frac{1}{n} \sum_{j \in S} y_j$.

In our running example, suppose we take an SRS S of size $n = 100$ from table R. Clearly, the inclusion probabilities for tuples in R are all equal to $100/10000 =$

0.01. If 9 tuples in the sample satisfy Q, then the HT estimator is $\hat{s}_{\text{spl}} = 9/100 = 0.09$, and the error is $E(\hat{s}_{\text{spl}}) = 80\%$.

A major problem with the use of sampling is the I/O overhead incurred. Since sampling requires random access to data, it is often the case that even if a very small sample is taken, the associated I/O cost is comparable to that of a full sequential scan of the data. For example, if each page contains 50 tuples, and the sample rate is higher than 2%, essentially all pages have to be accessed because $50 \times 2\% = 1$ (See [4] and [7] for a detailed analysis of this issue). Recently, there has been work on using page-level sampling in conjunction with tuple-level sampling to reduce the sampling cost [4, 7]. We take a complementary approach to this problem and attempt to decrease the sampling cost by utilizing existing synopsis information on the data. Haas et al. [4] show that the expected fraction f of pages to be accessed for a sample rate of q is given by $f = 1 - (1 - q)^c$, where c is the number of tuples on each page. It is evident that f decreases very fast as the sample rate drops, which means that if we can achieve the same level of accuracy with a lower sample rate, the I/O savings can be significant.

3 HASE

Our objective is to use the sample information in conjunction with the synopses to obtain better estimates. To this end, we develop a hybrid approach, HASE, by applying *generalized raking* [8, 9], a procedure originally utilized in survey sampling, to the problem of selectivity estimation.

3.1 Calibration

Suppose we have obtained a sample of the data, and we also know the selectivities of individual predicates P_i. We begin with an estimator constructed based on the sample only, without reference to any additional information, such as the HT estimator (Eq. (2)). For each tuple j in table R, in addition to the variable of interest y_j, we also associate with it an auxiliary vector $\mathbf{x}_j$ to reflect the results of evaluating P_i on j. Suppose each predicate P_i divides tuples in R into two disjoint subsets, $\mathcal{D}_i$ and $\bar{\mathcal{D}}_i$, according to whether they satisfy the predicate or not. We further define $\mathcal{D}_{m+1} = R$, i.e., $j \in \mathcal{D}_{m+1}$ for all j. Let $\mathbf{x}_j$ be a column vector of length $m+1$: $\mathbf{x}_j^T = (x_{j1}, \ldots, x_{jm}, x_{j,m+1})$, with the i-th ($1 \le i \le m+1$) element being 1 if $j \in \mathcal{D}_i$, and 0 otherwise. For instance, in the running example, $\mathbf{x}_j^T = (1, 0, 1)$ indicates that tuple j satisfies P_1, but not P_2.

Let $\mathbf{t}_x^T = (t_{x1}, \ldots, t_{xm}, t_{x,m+1}) = \frac{1}{N} \sum_{j \in R} \mathbf{x}_j$. Clearly, $t_{xi} = \frac{1}{N} \sum_{j \in S} x_{ji} = s_i$ ($1 \le i \le m$), the selectivity of predicate P_i, and $t_{x,m+1} = 1$. Therefore,

$$\mathbf{t}_x^T = (s_1, s_2, \ldots, s_m, 1) \tag{3}$$

Suppose s_i can be obtained based on synopsis structures, and $\mathbf{x}_j$ are observed for each tuple $j \in S$. This allows construction of a new estimator (which we call the *calibration estimator*)

$$\hat{s}_{\text{cal}} = \frac{1}{N} \sum_{j \in S} w_j y_j, \tag{4}$$

where the weights w_j are as close to the weights $d_j = 1/\pi_j$ as possible according to some distance metric (recall that π_j is the inclusion probability of j), and where

$$\frac{1}{N}\sum_{j\in S} w_j \mathbf{x}_j = \mathbf{t}_x, \tag{5}$$

meaning that the weighted average of the observed $\mathbf{x}_j$ has to reproduce the known selectivities s_i.

In light of the definition of $\mathbf{x}_j$ and Eq. (3), Eq. (5) can be rewritten as

$$\frac{1}{N}\sum_{j\in S\cap\mathcal{D}_i} w_j = s_i, \quad i = 1, 2, \ldots, m+1. \tag{6}$$

where $s_{m+1} = s$. Now w_j has a natural representation interpretation: it is the number of tuples "represented" by the sampled tuple j.

In our running example, Eq. (6) becomes

$$\frac{1}{10000}\sum_{j\in S\cap\mathcal{D}_1} w_j = 0.6, \frac{1}{10000}\sum_{j\in S\cap\mathcal{D}_2} w_j = 0.3, \text{ and } \frac{1}{10000}\sum_{j\in S} w_j = 1. \tag{7}$$

Although in general, there can be many possible choices for the sets of weights $\{w_j\}$ satisfying the constraints in Eq. (6), our goal is to select a set of new weights that are as close as possible to the original weights $d_i = 1/\pi_i$, which enjoy the desirable property of producing unbiased estimates. By keeping the distance between the new weights and the original weights as small as possible, we expect the new weights to remain nearly unbiased. We formulate this idea as a constrained optimization problem as described below.

3.2 The constrained optimization problem

Let $D(x)$ be a distance function (with $x = w_j/d_j$) that measures the distance between the new weights w_j and the original weights d_j. We assure that $D(x)$ satisfies the following requirements (for reasons that will become clear later): (i) D is positive and strictly convex, (ii) $D(1) = D'(1) = 0$, and (iii) $D''(1) = 1$. The optimization problem we have to solve is:

Minimize

$$\sum_{j\in S} d_j D(w_j/d_j) \tag{8}$$

subject to

$$\frac{1}{N}\sum_{j\in S} w_j \mathbf{x}_j = \mathbf{t}_x. \tag{9}$$

Here, both $\mathbf{x}_j$ and $\mathbf{t}_x$ are defined as in Section 3.1. Since $D(w_j/d_j)$ can have a large response to even a slight change in w_j when d_j is small, we minimize $\sum_{j\in S} d_j D(w_j/d_j)$ instead of $\sum_{j\in S} D(w_j/d_j)$ in order to dampen this effect. Also note that different distance functions can be used to measure the distance between $\{w_j\}$ and $\{d_j\}$, as long as the distance function complies with conditions (i) to (iii). In this paper, we consider the following two distance functions

because of the computational efficiency and interpretability. Both distance functions exhibit properties (i) to (iii). We discuss the choice of distance functions in Section 3.5.

1 The *linear* distance function: $D_{\text{lin}}(w_j/d_j) = \frac{1}{2}(\frac{w_j}{d_j} - 1)^2$, and
2 The *multiplicative* distance function: $D_{\text{mul}}(w_j/d_j) = \frac{w_j}{d_j} \log \frac{w_j}{d_j} - \frac{w_j}{d_j} + 1$

3.3 An algorithm based on Newton's method

We now present algorithms to solve the constrained optimization problem. A classical technique for solving constrained optimization problems is the method of Lagrange multipliers [10]. Note that the optimization problem can be rewritten as follows:

Minimize

$$\sum_{j \in S} d_j D(w_j/d_j) - \lambda^T (\sum_{j \in S} w_j \mathbf{x}_j - N\mathbf{t}_x) \tag{10}$$

with respect to $w_j (j \in S)$,

where $\lambda = (\lambda_1, \ldots, \lambda_m, \lambda_{m+1})$ is a Lagrange multiplier. Differentiating Eq. (10) with respect to w_j, we have

$$D'(w_j/d_j) - \mathbf{x}_j^T \lambda = 0 \tag{11}$$

Then we can solve the system formed by Eq. (11) and (9) for w_j. To do this, we obtain from (11) that

$$w_j = d_j F(\mathbf{x}_j^T \lambda), \tag{12}$$

where $F(x)$ is the inverse function of $D'(x)$. Conditions (i)-(iii) dictate that the inverse function always exists, and $F(0) = F'(0) = 1$. Substituting (12) into Eq. (9), we have the *calibration equations*

$$\sum_{j \in S} d_j F(\mathbf{x}_j^T \lambda) \mathbf{x}_j = N\mathbf{t}_x, \tag{13}$$

which can be solved numerically using Newton's method.

Let $\phi(\lambda) = \sum_{j \in S} d_j F(\mathbf{x}_j^T \lambda) \mathbf{x}_j - N\mathbf{t}_x$. Then

$$\phi'(\lambda) = \partial \phi(\lambda)/\partial \lambda = \sum_{j \in S} d_j F'(\mathbf{x}_j^T \lambda) \mathbf{x}_j \mathbf{x}_j^T.$$

We obtain successive estimates of λ, denoted by λ_k $(k = 0, 1, \ldots)$, through the following iteration:

$$\lambda_{k+1} = \lambda_k + [\phi'(\lambda_k)]^{-1} \phi(\lambda_k) \tag{14}$$

We take $\lambda_0 = \mathbf{0}$. Since we have

$$\phi(\mathbf{0}) = \sum_{j \in S} d_j F(0) \mathbf{x}_j - N\mathbf{t}_x = \sum_{j \in S} d_j \mathbf{x}_j - N\mathbf{t}_x,$$

and

$$\phi'(\mathbf{0}) = \sum_{j \in S} d_j F'(0) \mathbf{x}_j \mathbf{x}_j^T = \sum_{j \in S} d_j \mathbf{x}_j \mathbf{x}_j^T,$$

the first iteration yields $\lambda_1 = (\sum_{j \in S} d_j \mathbf{x}_j \mathbf{x}_j^T)^{-1} (\sum_{j \in S} d_j \mathbf{x}_j - N \mathbf{t}_x)$. The subsequent values of λ_k can be obtained following Eq. (14) until convergence.

In summary, the procedure to estimate the selectivity of Q is presented in Algorithm 1.

Algorithm 1 An algorithm for computing the calibration estimator based on Newton's method

INPUT: Q, D, S, N, $N_i (i = 1, \ldots, m)$, $d_j (j \in S)$, stopping threshold ϵ.
OUTPUT: $\hat{s}_{\text{cal}}$
for all $j \in S$ **do**
    Set the values of y_j, $\mathbf{x}_j$ according to the rules in Section 3.1;
end for
/*Solving the calibration equations using Newton's method*/
$\lambda_0 := \mathbf{0}$; $k := 0$;
repeat
    $\lambda_{k+1} := \lambda_k + [\phi'(\lambda_k)]^{-1} \phi(\lambda_k)$;
    $k := k + 1$;
until $||\lambda_k - \lambda_{k-1}|| < \epsilon$
for all $j \in S$ **do**
    $w_j := d_j F(\mathbf{x}_j^T \lambda)$;
end for
/*Obtaining the selectivity estimate based on the new weights*/
$\hat{s}_{\text{cal}} := \frac{1}{N} \sum_{j \in S} w_j y_j$;

Continuing the running example, the true frequencies obtained by evaluating the query Q on table R, and the observed frequency information based on a simple random sample S are given in Fig. 1 (both normalized so that all frequencies sum up to 1). The last row and column in each table correspond to the marginal frequencies.

	$P_2 = true$	$P_2 = false$	–
$P_1 = true$	0.05	0.55	**0.60**
$P_1 = false$	0.25	0.15	**0.40**
–	**0.30**	**0.70**	

(a) True frequencies

	$P_2 = true$	$P_2 = false$	–
$P_1 = true$	0.09	0.56	**0.65**
$P_1 = false$	0.24	0.11	**0.35**
–	**0.33**	**0.67**	

(b) Observed frequencies

Fig. 1. Example: True frequencies and observed frequencies from the sample

From Fig. 1, we know that the true selectivity of Q is 0.05 (the cell corresponding to $P_1 = true \wedge P_2 = true$ in Fig. 1(a)), and the sampling-based selectivity estimate is 0.09 (the cell corresponding to $P_1 = true \wedge P_2 = true$ in Fig. 1(b)). Clearly, the marginal frequencies obtained from the sample do not agree with the true marginal frequencies; therefore, calibration is needed. Applying Algorithm 1 to solve the calibration equations as shown in Eq. (7), we obtain

the following calibrated weights (using the multiplicative distance function):

$$w_j \simeq 60 \text{ for } j \in S \cap \mathcal{D}_1 \cap \mathcal{D}_2, w_j \simeq 102 \text{ for } j \in S \cap \mathcal{D}_1 \cap \bar{\mathcal{D}}_2$$
$$w_j \simeq 97 \text{ for } j \in S \cap \bar{\mathcal{D}}_1 \cap \mathcal{D}_2, w_j \simeq 140 \text{ for } j \in S \cap \bar{\mathcal{D}}_1 \cap \bar{\mathcal{D}}_2.$$

The selectivity estimate can then be computed:

$$\hat{s}_{\text{cal}} = \frac{1}{N} \sum_{j \in S} w_j y_j = \frac{1}{N} \sum_{j \in S \cap \mathcal{D}_1 \cap \mathcal{D}_2} w_j = 60 \times 9/10000 = 0.054.$$

The estimation error is $E(\hat{s}_{\text{cal}}) = |0.054 - 0.05|/0.05 = 8\%$. Compared with the error of the synopsis-based estimate $E(\hat{s}_{\text{his}}) = 260\%$ and the error of the sampling-based estimate $E(\hat{s}_{\text{spl}}) = 80\%$, this represents a significant improvement in the estimation accuracy.

3.4 An alternative algorithm

Although Newton's method works well, it is not the only option to conduct the optimization. Now we present an alternative algorithm for solving the calibration equations, which takes advantage of the intrinsic structure of the equations in (6) and does not require matrix inversion.

Since $w_j = d_j F(\mathbf{x}_j^T \lambda)$, Eq. (6) becomes

$$\frac{1}{N} \sum_{j \in S \cap \mathcal{D}_i} d_j F(\mathbf{x}_j^T \lambda) = s_i, \quad i = 1, \ldots, m+1. \tag{15}$$

Observe that the i-th Eq. ($2 \leq i \leq m$) can be solved for λ_i assuming all other $\lambda_l (l \neq i)$ are known, and the first and last equations can be solved for λ_1 and λ_{m+1} assuming all other $\lambda_l (l \neq 1, l \neq m+1)$ are known. Hence we have the algorithm shown in Algorithm 2. It is well known that such an iterative procedure converges to a proper solution [9], and in the case of multiplicative distance functions, this algorithm yields a variant of the classical iterative proportional fitting algorithm [11].

Replacing lines 6 to 11 in Algorithm 1 with Algorithm 2 results in a complete alternative estimation algorithm.

3.5 Distance measures

We now study the implications of the choice of distance functions D. In general, different distance functions result in different calibration estimators. However, it is well known [8] that regardless of the distance functions used (as long as the functions comply with conditions (i)-(iii)), the estimates obtained using the outcome of our specific optimization problem will converge asymptotically. Therefore, for medium to large sized samples (empirically, with sample size greater than 30), the choice of distance function does not have a heavy impact on the properties of the estimator; one can expect only slight difference in the estimates produced by using different functions. The main difference between the distance functions is thus their computational efficiency as well as interpretability.

Algorithm 2 An alternative algorithm for solving the calibration equations

INPUT: D, S, $N_i (i = 1, \ldots, m+1)$, $d_j (j \in S)$, stopping threshold ϵ.
OUTPUT: λ
$\lambda^{(0)} := \mathbf{0}$;
$k := 0$;
repeat
  Solve $\frac{1}{N} \sum_{j \in S \cap \mathcal{D}_1} d_j F(\mathbf{x}_j^T \lambda) = s_1$,
  and $\frac{1}{N} \sum_{j \in S} d_j F(\mathbf{x}_j^T \lambda) = 1$ for $\lambda_1^{(k+1)}$ and $\lambda_{m+1}^{(k+1)}$
  using values of $\lambda_l^{(k)}$ $(l = 2, \ldots, m)$;
  for $i = 2$ to m **do**
    Solve $\sum_{j \in S \cap \mathcal{D}_i} d_j F(\mathbf{x}_j^T \lambda) = s_i$ for $\lambda_i^{(k+1)}$,
    using values of $\lambda_l^{(k)}$ $(l = 1, \ldots, m+1, l \neq i)$;
  end for
  $k := k + 1$;
  $MaxChange := \max\{|\lambda_l^{(k)} - \lambda_l^{(k-1)}|\}$, $l = 1, \ldots, m+1$
until $MaxChange < \epsilon$

For the linear function, D_{lin}, $D'(x) = x - 1$; therefore, the inverse function is $F(z) = z + 1$. In Algorithm 1, it is easy to verify that λ converges at $\lambda_1 = (\sum_{j \in S} d_j \mathbf{x}_j \mathbf{x}_j^T)^{-1} (\sum_{j \in S} d_j \mathbf{x}_j - \mathbf{t}_x)$. Therefore, when the linear function is used, only one iteration is required, which makes the linear method the faster of the two distance functions considered here. A major drawback of this function is that the weights can be negative. This can lead to negative selectivity estimates. For instance, in the running example, we take a sample of size 10 from R, and the observed frequencies are the following: $P_1 = true \cap P_2 = true$: 2; $P_1 = true, P_2 = false$: 5; $P_1 = false \cap P_2 = true$: 3; $P_1 = false \cap P_2 = false$: 0. Solving the calibration equation, we have $w_j = -500$ for $j \in S \cap \mathcal{D}_1 \cap \mathcal{D}_2$. Therefore, the selectivity estimate $\hat{s}_{\text{cal}} = 2 \times (-500)/10000 = -0.1$. Negative weights and selectivity estimates do not have a natural interpretation and thus are undesirable. Note that, however, this usually only occurs for small-sized samples. When the sample size gets large, all estimators with distance functions satisfying conditions (i)-(iii) are asymptotically equivalent and give positive weights and selectivity estimates.

For the multiplicative function, D_{mul}, $D'(x) = \log x$; the inverse function is therefore $F(z) = e^z$. When the multiplicative function is used, it may require more than one iteration, but our experience indicates that it often converges after only a few iterations (typically two in our experiments). An advantage of using this function is that it always leads to positive weights because $w_j = d_j F(\mathbf{x}_j^T \lambda) = d_j \exp\{\mathbf{x}_j^T \lambda\} > 0$. We will contrast the effects of both functions on the estimation accuracy in Section 4.

3.6 Probabilistic bounds on the estimation error

Let π_{jl} be the probability that both j and l are included in the sample, and $\pi_{jj} = \pi_j$. We assume that the sampling scheme is such that the π_{jl}'s are strictly

positive. Let β be a vector satisfying the equation

$$\sum_{j \in R} d_j \mathbf{x}_j (y_j - \mathbf{x}_j^T \beta) = 0$$

and let $\Delta_{jl} = \pi_{jl} - \pi_j \pi_l$, $\epsilon_j = y_j - \mathbf{x}_j^T \beta$. We have the following result on the error bounds of the estimation error.

Theorem 1. *When the sample size is sufficiently large, for a given constant $\alpha \in (0,1)$, the selectivity s_Q is bounded by $(\hat{s}_{cal} - z_{\alpha/2}\sqrt{V(\hat{s}_{cal})}, \hat{s}_{cal} + z_{\alpha/2}\sqrt{V(\hat{s}_{cal})}$ with probability $1-\alpha$, where $z_{\alpha/2}$ is the upper $\alpha/2$ point of the standard normal distribution, and $V(\hat{s}_{cal}) = \sum_{j \in R} \sum_{j \in R} (\Delta_{jl}/\pi_{jl})(w_j \epsilon_j)(w_l \epsilon_l)$.*

Proof Sketch: When the linear distance function is used, $w_j = d_j(1 + \mathbf{x}_j^T \lambda)$. We know from Section 3.5 that the solution of the calibration equation converges at $\lambda = (\sum_{j \in S} d_j \mathbf{x}_j \mathbf{x}_j^T)^{-1} (\sum_{j \in S} d_j \mathbf{x}_j - \mathbf{t}_x)$. Therefore, $w_j = d_j[1 + \mathbf{x}_j^T (\sum_{j \in S} d_j \mathbf{x}_j \mathbf{x}_j^T)^{-1} (\sum_{j \in S} d_j \mathbf{x}_j - \mathbf{t}_x)]$. Let $\hat{\beta}_S$ be the solution to the equation

$$\sum_{j \in S} d_j \mathbf{x}_j (y_j - \mathbf{x}_j^T \hat{\beta}_S) = 0.$$

Then the estimator $\hat{s}_{\text{cal}}$ can be written as

$$\hat{s}_{\text{cal}} = \frac{1}{N} \sum_{j \in S} w_j y_j = \hat{s}_{\text{spl}} + \frac{1}{N} (\mathbf{t}_x - \sum_{j \in S} d_j \mathbf{x}_j)^T \hat{\beta}_S,$$

which takes the form of a generalized regression estimator (GREG) [12]. Applying results on the asymptotic variance of GREG [12], we obtain the asymptotic variance of the estimator $\hat{s}_{\text{cal}}$:

$$V(\hat{s}_{\text{cal}}) = \sum_{j \in R} \sum_{j \in R} (\Delta_{jl}/\pi_{jl})(w_j \epsilon_j)(w_l \epsilon_l).$$

Since it has been shown that all estimators with distance functions satisfying conditions (i)-(iii) are asymptotically equivalent [8], all estimators have the same asymptotic variance $V(\hat{s}_{\text{cal}})$. When the sample S is large enough, the Central Limit Theorem applies. Therefore, for a given constant $\alpha \in (0,1)$, s_Q is bounded by $(\hat{s}_{\text{cal}} - z_{\alpha/2}\sqrt{V(\hat{s}_{\text{cal}})}, \hat{s}_{\text{cal}} + z_{\alpha/2}\sqrt{V(\hat{s}_{\text{cal}})}$ with probability $1-\alpha$. □

3.7 Utilizing multi-attribute synopses

In our discussion, we have assumed that we have knowledge of the selectivities s_i of individual predicates P_i based on single-attribute synopsis structures. In fact, the estimation procedure can be easily extended so that multi-attribute synopsis structures can also be utilized when they are present. Suppose that a multi-dimensional synopsis [13, 2] exists on a set of attributes $\mathcal{A}$. It is relatively easy to derive lower-dimensional synopses from higher-dimensional synopses, i.e., synopses on any subset(s) of $\mathcal{A}$ can be obtained from the synopsis on $\mathcal{A}$. Let $\mathcal{A}_Q$

be the set of attributes involved in query Q. If $\mathcal{A} \cap \mathcal{A}_Q \neq \emptyset$, the synopsis on $\mathcal{A}$ can be utilized. Let $\mathcal{U} = \mathcal{A} \cap \mathcal{A}_Q$, and let $P_{\mathcal{U}}$ be the conjuncts of predicates in which attributes in $\mathcal{U}$ are involved. Then the selectivity $s_{\mathcal{U}}$ of $P_{\mathcal{U}}$ can be estimated based on the synopsis on $\mathcal{U}$. We augment the auxiliary vector $\mathbf{x}_j$ by an additional element reflecting whether j satisfies $P_{\mathcal{U}}$. Changes are also made accordingly to $\mathbf{t}_x$, with the addition of an element with value $s_{\mathcal{U}}$. The algorithms for solving the calibration equations presented above can then be applied in order to obtain $\hat{s}_{\text{cal}}$.

4 Experimental evaluation

In this section, we report the results of an experimental evaluation of the proposed estimation procedure.

4.1 Experiment setup

We compare the accuracy of HASE with that of the synopsis-based and sampling-based approaches using synthetic as well as a real data set. The real data set we use is the *Census Income* data obtained from the UCI KDD Archive [14].

- Synthetic data are used to study the properties of the HASE in a controlled manner. We generate a large number of synthetic data sets by varying the following parameters:
  Data skew: The data in each attribute are generated from a Zipfian distribution with parameter z ranging from 0 (uniform distribution) to 3 (highly-skewed distribution). The number of distinct values in each attribute is fixed to 10.
  Correlation: By default, the data are independently generated for each attribute. We introduce correlation between a pair of attributes by transforming the data such that the correlation coefficient between the two attributes is approximately ρ. The parameter ρ ranges from 0 to 1, representing an increasing degree of correlation. In particular, $\rho = 0$ corresponds to the case where there is no correlation between the two attributes; $\rho = 1$ indicates that the two attributes are fully dependent, i.e., knowing the value of one attribute enables one to perfectly predict the value of the other attribute. This is achieved by first independently generating the data for both attributes (say, A_1 and A_2) and then performing the following transformation. For each tuple with $A_1 = a_1$ and $A_2 = a_2$, we replace a_2 by $a_1 \times \rho + a_2 \times \sqrt{1 - \rho^2}$, suitably rounded. For three or more attributes, we create data such that the correlation coefficient between any pair of attributes is approximately ρ.
  The real data set *Census Income* contains weighted census data extracted from the 1994 and 1995 population surveys conducted by the U.S. Census Bureau. It has 199,523 tuples and 40 attributes representing demographic and employment related information. Out of the 40 attributes, 7 are continuous, and 33 are nominal.
- We evaluate HASE on two different query workloads. The first set of queries consist of 100 range queries where each predicate in the query takes the form

of ($attribute <= constant$) with randomly chosen $constant$. The second set of queries consist of 100 equality queries where each predicate takes the form of ($attribute = constant$) where $constant$ is randomly chosen.
- We use simple random sampling as the sampling scheme in our experiments for both the sampling-based approach and HASE. All numbers reported are averages of 30 repetitions.
- We use the exact frequency distributions of individual attributes as the synopses.
- The absolute relative error defined in Eq. (1) is used as the error metric.

4.2 Results on synthetic data

In all experiments, similar trends are observed for both range and equality queries; we only report the results on range queries because of space limitations. We first study the effects of various parameters in the case of two attributes (i.e., only two predicates on two different attributes are involved in the query), and then show the effect of the number of attributes on the estimation accuracy. The individual selectivities are obtained based on the frequencies of values in each attribute. Since our results indicate that the number of tuples T in the table does not have a significant effect on the accuracy of the estimators, only the results for $T = 100,000$ are shown here.

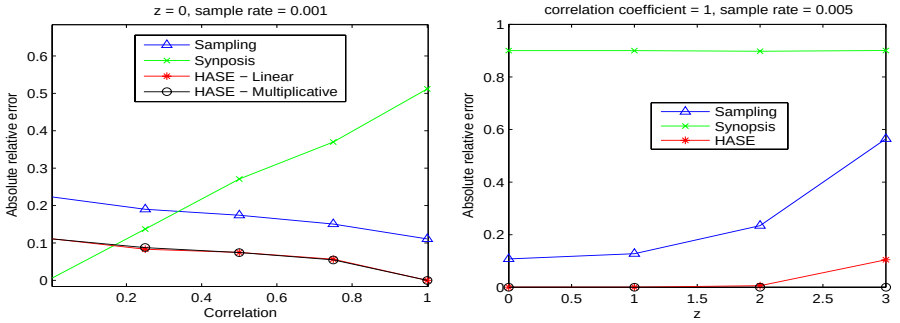
Correlation We study the effect of the correlation between attributes on the estimation accuracy by varying the correlation coefficient ρ from 0 to 1, representing an increasing degree of correlation. Fig. 2(a) presents a typical result.

When the two attributes are totally uncorrelated ($\rho = 0$), the accuracy of the synopsis-based approach is very high, with an error close to zero, better than the other two methods. This is because in such cases, the attribute-value independence assumption holds true, and the selectivity estimate for the query is indeed the product of the individual selectivities of the two predicates. The accuracy of this approach deteriorates when the degree of correlation increases and the actual relationship between the two attributes deviates further from the independence assumption.

The accuracy of the sampling-based approach actually improves when the two attributes become more correlated. The reason is as follows. When the degree of correlation increases, the number of distinct value combinations[3] in the two attributes decreases, as the data become more "concentrated". Therefore, the sample space (containing all distinct value combinations) becomes smaller, and thus sampling becomes more efficient (i.e., for a given sample rate, it is more likely to include in the sample a tuple satisfying the query).

The accuracy of HASE also increases with the degree of correlation. Since HASE utilizes sample information, the preceding argument for the sampling-based approach also applies. Besides, as the degree of correlation increases, the benefit of adjusting the weights in accordance with known single-attribute synopses becomes more evident. In the extreme case where the two attributes are fully dependent ($\rho = 1$), it essentially produces the exact selectivity, provided

[3] (a, b) is considered a value combination if $\exists j \in R$ such that $A_1 = a$ and $A_2 = b$.



(a) Accuracy vs. correlation

(b) Accuracy vs. data skew

Fig. 2. The effects of correlation and data skew

that there is at least one tuple in the sample satisfying the query. To see why this is the case, consider the following query $Q = P_1 \cap P_2 = (A_1 = a) \cap (A_2 = b)$. Full dependency dictates that if there is at least one tuple in the table satisfying this query, then for any other value c ($c \neq a$)in A_1 and d ($d \neq b$) in A_2, both $(A_1 = a) \cap (A_2 = d)$ and $(A_1 = c) \cap (A_2 = b)$ evaluate to false. This implies that $s = s_1 = s_2$. Therefore, if in the auxiliary vector $\mathbf{x}_j$ for tuple j, we have $x_{j1} = 1$ (which corresponds to $A_1 = a$), then y_j (the variable indicating whether j satisfies Q) must also be 1, and vice versa. Since we know s_1, we have $\frac{1}{N}\sum_{j \in S} w_j x_{j1} = s_1$ as a constraint in the optimization problem. If we can find a set of w_j that satisfy this constraint, then the calibration estimator $\frac{1}{N}\sum_{j \in S} w_j y_j$ must also yield s_1, which means we have a perfect selectivity estimate. One exception to this analysis is that when there is no tuple $j \in S$ satisfying Q, we may no longer be able to produce the exact estimate. In such cases, all $y_j (j \in S)$ are 0; therefore, regardless of the weights, the calibration estimator $\frac{1}{N}\sum_{j \in S} w_j y_j$ will also be zero, which may be different from the exact selectivity.

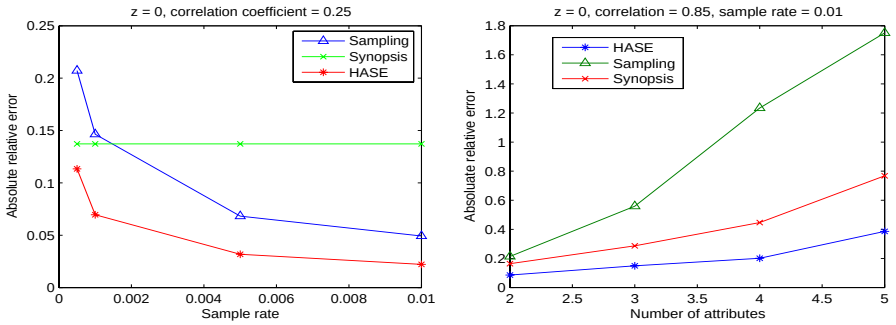
In all cases, HASE produces significantly more accurate estimates than the sampling-based method, with a 50%-100% reduction in error. Both distance functions give very close estimates, verifying the claim that estimators using different distance functions are asymptotically equivalent. In the following discussion, we only show the results for the case of the linear distance function.

Data skew We study the effect of data skew by varying the Zipfian parameter z from 0 (uniform) to 3 (highly-skewed), a typical result is shown in Fig. 2(b). The errors of both HASE and the sampling approach increase as the data becomes increasingly more skewed. The reason is that when the data skew in each attribute increases, the frequencies of some value combinations decrease. As a result, when we query on those value combinations with low occurrence frequencies, it becomes increasingly possible that no sampled tuple can satisfy the query. This gives rise to more errors, because with no sampled tuple satisfying the query, the estimate has to be zero, whereas the actual selectivities are not. Note that this situation is different from the case of increasing correlation as discussed above. The main effect of increasing the skew is a decrease in

the frequencies of some value combinations, not necessarily reducing the number of value combinations present in the table. Increasing correlation, on the other hand, generally results in a reduction in the number of value combinations. Therefore, increasing skew and increasing correlation have different effects on the accuracy of HASE as well as the sampling-based approach.

Another interesting observation from Fig. 2(b) is that the accuracy of the synopsis-based approach remains virtually the same regardless of the data skew. The reason is as follows. Assuming independence between attributes, the synopsis-based approach estimates the selectivity by $\hat{s}_{\text{his}} = s_1 * s_2$. In Fig. 2(b), the two attributes are fully dependent, which implies that the actual selectivity $s = s_1 = s_2$. Thus, $E(\hat{s}_{\text{his}}) = (s - s_1 s_2)/s_1 = 1 - s_1$. The average error over a large number of (uniformly) randomly selected equality queries is therefore $1 - avg(s_1)$. In our case, since there are 10 distinct values in each attribute, $avg(s_1) = 1/10 = 0.1$. the average error of the estimate is thus $1 - 0.1 = 0.9$. Therefore, the accuracy of this approach does not change with data skew in this case.

Sample rate Fig. 3(a) shows a typical result on how the three approaches behave as we increase the sample rate. The number of attributes in the data set is 2. The accuracy of the synopsis-based approach remains unchanged across the range of sample rates, because it does not depend on sampling. The accuracy of both HASE and the sampling-based approach improves with increasing sample rate, as one would expect. For all sample rates, HASE outperforms both the synopsis-based and the sampling-based approaches. It is also worth noting that using HASE, we can achieve the same level of accuracy with a much smaller sample rate than that required by the sampling-based approach. For example, in Fig. 3(a), the sampling-based approach has an error of 0.07 when the sample rate is 0.005. HASE achieves approximately the same level of accuracy with a sample rate of 0.001, resulting in a reduction by a factor of 5. This translates into more significant I/O savings because of the non-linear relationship between the I/O cost and the sample rate as discussed in Section 2.3.



(a) Accuracy vs. sample rate (b) Accuracy vs. number of attributes

Fig. 3. The effects of the sample rate and the number of attributes

Number of attributes We vary the number of attributes involved in the query from 2 to 5 to study the impact of the number of attributes on the estimation accuracy. A typical result is shown in Fig. 3(b). Clearly, the accuracy of all three approaches decreases as the number of attributes increases. This is not surprising, because having more attributes would introduce more sources of errors. A space of higher dimensionality requires a much larger sample to cover a fixed portion of the space, in comparison with a space of lower dimensionality. Note from Fig. 3(b), however, that HASE outperforms the other two approaches for all number of attributes, and has a lower rate of decrease in accuracy.

4.3 Results on real data

Since the *Census Income* data has 40 attributes, there are $40 \times 39 = 1560$ attribute pairs. We randomly choose 100 attribute pairs and record the accuracy of the three approaches as the sample rate increases. The result is shown in Fig. 4. The trends are similar to those for the synthetic data, with HASE significantly outperforming both the synopsis-based and the sampling-based approaches. The error response to the number of attributes is also similar to that for the synthetic data, and is therefore omitted here.

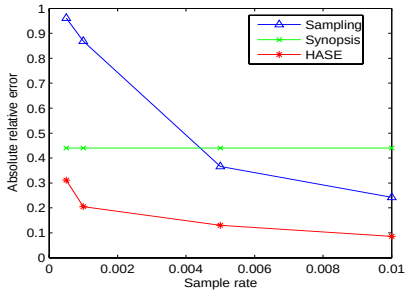


Fig. 4. Accuracy vs. sample rate on the *Census Income* data

5 Related Work

The issue of selectivity estimation has been extensively studied in the literature and a large variety of methods have been proposed [15, 1, 16, 17].

Histograms are probably the most widely used form of synopses in commercial database systems (e.g., DB2 UDB, Oracle, SQL Server, etc.). See [18] for an excellent survey on this topic. Aside from histograms, other types of synopses have also been proposed in the literature, such as wavelet-based synopses (e.g., [16]) and parametric synopses (e.g., [19]).

Markl et al. [20] propose a method to consistently utilize various multi-dimensional synopsis structures for selectivity estimation of conjunctive predicates. This work is close in spirit to ours in that both of them address the issue of consistent utilization of various sources of information for selectivity estimation. However, their focus is on reconciling the estimates obtained from different

synopsis structures, whereas we attack the problem of utilizing both synopses and sample information.

Olken [3] provides a survey of techniques on sampling from databases. Lipton et al. [15] propose an *adaptive sampling* (a.k.a. *sequential sampling*) approach to selectivity estimation. Haas and Swami [21] improve the sequential sampling approach by establishing tighter termination conditions. There has also been work on estimating the number of distinct values via sampling [22–24]. Recently, Haas et al. [4] and Chaudhuri et al. [7] address the efficiency of sampling and propose techniques to utilize page-level sampling in conjunction with tuple-level sampling. Techniques have also been developed to use sampling to construct synopsis structures [25, 24, 7]. Note that sampling is used here only for fast construction of data synopses, which are then used for selectivity estimation; they do not consider the issue of direct utilization of *both* sampling and synopses for selectivity estimation.

6 Conclusions and Future Work

Existing work on selectivity estimation can be classified as either synopsis-based or sampling-based, depending on whether the basis for estimation is the synopsis structures stored in the database or sample information. The presence of both sources of information presents a unique challenge, as it is nontrivial to make consistent use of them in order to obtain better estimation. To the best of our knowledge, we are the first to tackle this challenge. We proposed HASE, a new estimation procedure based on generalized raking, and the problem is formulated as a constrained optimization problem. We then presented two algorithms to solve it. We also discussed the implications of different distance functions, and provided asymptotic error bounds on the selectivity estimate thus obtained. The experiments demonstrated the effectiveness of the proposed approach.

For future work, we would like to consider extending HASE to handle the selectivity estimation of more complex queries, such as joins and aggregations. We also plan to extend HASE to handle the case where multi-attribute synopses (e.g., multi-dimensional histograms) are available. It would also be interesting to study in our framework how to best divide the efforts between constructing histograms and sampling for a given query workload.

Acknowledgment

The authors would like to thank our friend and mentor Prof. Kenneth C. Sevcik for his comments and encouragement during the course of this work. We miss him dearly.

References

1. Poosala, V., Ioannidis, Y.E., Haas, P.J., Shekita, E.J.: Improved histograms for selectivity estimation of range predicates. In: SIGMOD. (1996) 294–305
2. Poosala, V., Ioannidis, Y.E.: Selectivity estimation without the attribute value independence assumption. In: VLDB. (1997) 486–495
3. Olken, F.: Random sampling from databases. PhD thesis, University of California, Berkeley, CA (1993)
4. Haas, P.J., König, C.: A bi-level Bernoulli scheme for database sampling. In: SIGMOD Conference. (2004) 275–286
5. Deshpande, A., Garofalakis, M.N., Rastogi, R.: Independence is good: Dependency-based histogram synopses for high-dimensional data. In: SIGMOD Conference. (2001)
6. Horvitz, D.G., Thompson, D.J.: A generalization of sampling without replacement from a finite universe. Journal of the American Statistical Association **47** (1952) 663–685
7. Chaudhuri, S., Das, G., Srivastava, U.: Effective use of block-level sampling in statistics estimation. In: SIGMOD Conference. (2004) 287–298
8. Deville, J.C., Särndal, C.E.: Calibration estimators in survey sampling. Journal of the American Statistical Association **87** (1992) 376–382
9. Deville, J.C., Särndal, C.E., Sautory, O.: Generalized raking procedures in survey sampling. Journal of the American Statistical Association **88** (1993) 1013–1020
10. Bertsekas, D.P.: Constrained Optimization and Lagrange Multiplier Methods. Athena Scientific (1996)
11. Deming, W.E., Stephan, F.F.: On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. Annals of Mathematical Statistics **11** (1940) 427–444
12. Särndal, C.E., Swensson, B., Wretman, J.: Model Assisted Survey Sampling. Springer-Verlag, New York (1992)
13. Muralikrishna, M., DeWitt, D.J.: Equi-depth histograms for estimating selectivity factors for multi-dimensional queries. In: SIGMOD. (1988) 28–36
14. Hettich, S., Bay, S.D.: The UCI KDD Archive. Irvine, CA: University of California, Department of Information and Computer Science. (1999)
15. Lipton, R.J., Naughton, J.F.: Query size estimation by adaptive sampling. In: PODS. (1990) 40–46
16. Matias, Y., Vitter, J.S., Wang, M.: Wavelet-based histograms for selectivity estimation. In: SIGMOD. (1998) 448–459
17. Aboulnaga, A., Chaudhuri, S.: Self-tuning histograms: building histograms without looking at data. In: SIGMOD, ACM Press (1999) 181–192
18. Ioannidis, Y.E.: The history of histograms (abridged). In: VLDB. (2003) 19–30
19. Fedorowicz, J.: Database evaluation using multiple regression techniques. In: SIGMOD. (1984) 70–76
20. Markl, V., Megiddo, N., Kutsch, M., Tran, T.M., Haas, P.J., Srivastava, U.: Consistently estimating the selectivity of conjuncts of predicates. In: VLDB. (2005) 373–384
21. Haas, P.J., Swami, A.N.: Sequential sampling procedures for query size estimation. In: SIGMOD. (1992) 341–350
22. Naughton, J.F., Seshadri, S.: On estimating the size of projections. In: ICDT. Volume 470. (1990) 499–513
23. Haas, P.J., Naughton, J.F., Seshadri, S., Stokes, L.: Sampling-based estimation of the number of distinct values of an attribute. In: VLDB. (1995) 311–322
24. Chaudhuri, S., Motwani, R., Narasayya, V.R.: Random sampling for histogram construction: How much is enough? In: SIGMOD. (1998) 436–447
25. Gibbons, P.B., Matias, Y., Poosala, V.: Fast incremental maintenance of approximate histograms. In: VLDB. (1997) 466–475