A Comparison of Selectivity Estimators for Range Queries on Metric Attributes[1]

Björn Blohsfeld Dieter Korus Bernhard Seeger

Fachbereich Mathematik und Informatik
University of Marburg
{blohsfel,korus,seeger}@mathematik.uni-marburg.de

Abstract:

In this paper, we present a comparison of nonparametric estimation methods for computing approximations of the selectivities of queries, in particular range queries. In contrast to previous studies, the focus of our comparison is on metric attributes with large domains which occur for example in spatial and temporal databases. We also assume that only small sample sets of the required relations are available for estimating the selectivity. In addition to the popular histogram estimators, our comparison includes so-called kernel estimation methods. Although these methods have been proven to be among the most accurate estimators known in statistics, they have not been considered for selectivity estimation of database queries, so far. We first show how to generate kernel estimators that deliver accurate approximate selectivities of queries. Thereafter, we reveal that two parameters, the number of samples and the so-called smoothing parameter, are important for the accuracy of both kernel estimators and histogram estimators. For histogram estimators, the smoothing parameter determines the number of bins (histogram classes). We first present the optimal smoothing parameter as a function of the number of samples and show how to compute approximations of the optimal parameter. Moreover, we propose a new selectivity estimator that can be viewed as an hybrid of histogram and kernel estimators. Experimental results show the performance of different estimators in practice. We found in our experiments that kernel estimators are most efficient for continuously distributed data sets, whereas for our real data sets the hybrid technique is most promising.

1. Introduction

The efficient support of computing approximate answers of aggregate queries has been an important subject in the database community for more than two decades. In case of query optimization, the sizes of intermediate results of a query are estimated to evaluate execution plans. First methods for estimating the size of the intermediate results were developed in System R [12]. More recently, the subject of approximate computation of aggregate queries has received the attention of researchers in the area of data warehousing. Since the underlying databases are very large, a precise computation of aggregate queries is generally too expensive. Moreover, a user would also appreciate approximate answers, when they are sufficiently precise and when they are delivered in considerably less time than the exact answers [6].

In this paper, we address the problem of selectivity estimation of a query, i. e., we are interested in a precise and inexpensive estimation of the query result size. There are many different methods for estimating the selectivity of a query ([2], [3], [4], [7], [9]). These methods are based on statistical or numerical methods for approximating either the density of a distribution or the frequency distribution. In accordance with [9], we first present a classification of these methods w.r.t. two dimensions. In the first dimension we distinguish between parametric estimation methods and non-parametric ones. In the second dimension, we use properties of the domain of the corresponding attribute as a criterion for classification. Let us first discuss the difference between parametric and nonparametric methods.

For a *parametric method* an estimate of the distribution function is computed by using a so-called model function. A model function can be a predefined distribution function with a certain number of free parameters or a polynomial function of a certain degree. For example, the uniform distribution was used as a model function in System R [12]. Although inexpensive to compute, these methods provide only accurate estimations if the real distribution is closely related to the model function. *Nonparametric methods* do not assume that the real distribution function belongs to a certain family. Examples for such methods are histogram and kernel estimation methods. Both methods require a sample set of the underlying database. *Histogram estimates* first partition the complete attribute domain into disjoint subsets where each of them represents a histogram class, also called *bin*. For each bin, the number of samples is stored that belong to the corresponding set. In order to reduce the cost of building a histogram, the sample set should be small. It is also shown in [8] that a small sample set is sufficient for computing an accurate histogram.

In the current statistical literatur, see [15] for example, the main interest is however on *kernel estimators,* but their great potential have not been recognized in the database community, so far. Kernel estimators can be viewed as a generalization of sampling where a sampling point distributes its mass among its neighborhood. The *bandwidth* of the kernel estimator controls the size of the impact ranges of its samples and a *kernel function* is responsible for the distribution of the

SIGMOD '99 Philadelphia PA

[1]This work has been supported by grant no. SE 553/2-1 from DFG.

mass among the impact range of a sample. The density function of the distribution is then approximated by superimposing the different occurrences of the kernel function. An example is shown in figure 1, where the lower curves corresponds to the kernel functions of 5 samples and the upper curve depicts the estimation of the underlying density function. Kernel estimation methods offer the advantage of sampling methods that there is no need to keep further statistics about the data. In comparison to pure sampling, however, the approximation of kernel estimation methods converges on a much faster rate to the underlying distribution function under mild assumptions [11]. A user therefore can either compute an estimation of a certain precision with less samples or a higher precision of the estimation is achieved with the same number of samples.

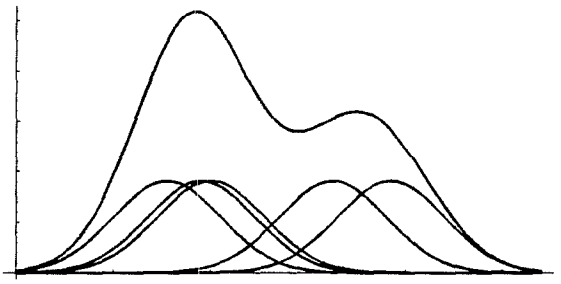

Fig. 1: An example of kernel density estimation

In the second dimension of our classification the estimation methods are distinguished according to the properties of the domain of the attribute. For a *categorical domain*, estimation methods are only able to estimate the probability that a record will be in one of the categories. In general, there is no natural ordering on the data of a categorical domain and therefore, range queries are only supported when an artificial ordering is introduced. A *metric domain*, also termed numerical or cardinal, provides an ordering on the data and therefore, range queries can be performed on the corresponding data sets. Examples of such domains are found in spatial and temporal databases. This class can be refined into *discrete* and *continuous* domains [9]. For a continuous domain a value appears at most once, whereas duplicates are allowed for a discrete domain. In the broad class of histogram estimation methods there are methods which are most suitable either for categorical domain or for metric domain. For example, a serial histogram [2] is a method for a categorical domain, whereas equi-width histograms and equi-depth histograms [3] generally require a metric domain.

The main contributions of the paper are as follows. In this paper, we introduce *kernel selectivity estimation* which is a nonparametric method suitable for estimating the selectivity of selection queries on metric attributes, in particular selection queries with a range predicate. Recently, the technique of kernel density estimation has been received much attention in statistics ([13],[15]), but so far it has not been applied to estimating the selectivity of database queries. Kernel estimators can produce high errors for range queries that are close to the left (or right) boundary of the domain. The reason is that the lack of data items left (or right) beside of the boundary results in high estimation errors. Two techniques are introduced to reduce the boundary problem of kernel estimation methods.

Most nonparametric estimation methods require an appropriate setting of a smoothing parameter. For kernel estimation method, the smoothing parameter controls the bandwidth of the kernel function. The choice of the bandwidth is crucial to the accuracy of the kernel estimation methods. For histograms, the smoothing parameter determines the number of bins. Similar important as the choice of the bandwidth of a kernel estimation method is an appropriate setting of the number of bins for a histogram. We present for kernel estimation methods and equi-width histograms the optimal bandwidth and optimal number of bins such that the mean integrated square error is asymptotically minimized, respectively. Moreover, we give simple rules for computing approximations of both the optimal bandwidth and the number of bins.

Extensive experiments were conducted with both synthetic and real data sets. The results of our experiments are interesting as they do not always confirm the results of other experiments ([3], [8]). For example, we found in our experiments that equi-width histograms provide slightly better results than equi-depth histograms and considerably better results than max-diff histograms. One reason is that our experiments were performed on metric attributes where the cardinality of the domains was large and a value appeared only a few times in the database, in general.

The remaining paper is structured as follows. The next section introduces our terminology, the most important requirements on estimators and the error metrics. In section we review histograms for selectivity estimation and propose the kernel estimator for selectivity estimation. In section 4, we present different rules for choosing the smoothing parameter which is applicable to both histogram methods and kernel methods. Section 5 presents some of our experimental results. Section 6 concludes the paper.

2. Preliminaries

In this section, we first present our most important notations. Then, we introduce the error measures used throughout the paper.

Let R be a relation with an attribute A. The domain of A is assumed to be the real line $\Re$. Let F be the underlying distribution of attribute A. F is a continuous distribution, if there is a real function f, the so-called probability density function (PDF), with the property that $F(x) = \int_{-\infty}^{x} f(t)dt$ holds for all $x \in \Re$.

For $a, b \in \Re$ and $a < b$, a range query $Q = Q(a,b)$ retrieves all records r from R with $a \le r.A \le b$. In order to quantify the query result size we introduce the terms instance selectivity and distribution selectivity. The *instance selectivity* of a query is given by the number of results in the actual instance of R divided by the total number of records in the actual instance, whereas the *distribution selectivity* $\sigma(a, b)$

is the probability that a record is in the range $[a, b]$. The distribution selectivity is independent from an arbitrary instance of a relation, but the instance selectivity can be estimated by N times the distribution selectivity where N denotes the number of records in the actual instance. In the following, selectivity always refers to distribution selectivity. For a continuous distribution F, the selectivity is simply given by

$$\sigma(a,b) = F(b) - F(a) = \int_a^b f(t)dt. \quad (1)$$

Let $X = \{X_1, ..., X_n\}$ be a set of n samples of the actual instance of R. Our goal is to compute an approximation of the selectivity only by using X. A simple approach is first to compute an estimator $\hat{f} = f(x;X)$ of the PDF f and then, to substitute f by $\hat{f}$ in (1). We obtain then the following selectivity estimation of σ:

$$\hat{\sigma}(a,b) = \int_a^b \hat{f}(t)dt \quad (2)$$

An estimator $\hat{\sigma}$ of σ is *consistent*, if for all $\varepsilon > 0$ the following holds:

$$\lim_{n \to \infty} P(|\sigma - \hat{\sigma}| < \varepsilon) = 1$$

where P denotes the probability of an event. Consistency is obviously an important requirement for an estimator. It gives a guarantee (with high probability) that the estimator approaches to the desired value when the number of samples is sufficiently large. Pure sampling can directly be used as a method for selectivity estimation. The method is an example for a consistent estimator, but the convergence rate is only $O(n^{-1/2})$. The construction of estimators with higher convergence rate therefore has been an important research subject in statistics. For example, an equi-width histogram provides an estimator with a higher convergence rate than pure sampling (up to $O(n^{-2/3})$) when the number of bins varies with the number of samples in a certain way [11]. However, when the number of bins is kept constant, an equi-width histograms is not consistent.

There are many different error metrics used to evaluate the quality of an estimator. In order to achieve the theoretical fundament of our approach, we first introduce the *mean integrated squared error (MISE)*. The MISE is commonly used to express the global accuracy of an estimator. Let $\hat{f}$ be an estimator of the PDF f. The MISE is then defined by

$$MISE(\hat{f}) = E\left(\int_{-\infty}^{\infty} (\hat{f}(x) - f(x))^2 dx\right). \quad (3)$$

Note that the MISE is not directly related to selectivity estimation, but it is defined on the PDF f and its estimator $\hat{f}$. The convergence of the MISE of $\hat{f}$ is however a sufficient criterion for $\hat{\sigma}$ being consistent. This may explain that the MISE is commonly used in statistics. Moreover, the MISE can also be simplified to

$$MISE(\hat{f}) = \int_{-\infty}^{\infty} Var(\hat{f}(x))dx + \int_{-\infty}^{\infty} Bias(\hat{f}(x))^2 dx, \quad \text{where}$$

$$Var(\hat{f}) = E(\hat{f}^2) - E(\hat{f})^2 \quad \text{and} \quad Bias^2(\hat{f}) = (E(\hat{f}(x)) - f(x))^2$$

N	**number of tuples in the database**
n	sample size
Q(a,b)	range query from a to b
$\sigma = \sigma(Q) = \sigma(a,b)$	(distribution) selectivity of range query $Q(a,b)$
F	distribution function
f	probability density function (PDF)
$\hat{\sigma}, \hat{f}$	estimation of σ and f
(A)MISE	(approximated) mean integrated squared error
K	kernel function
h	bandwidth of the kernel function

Table 1: Notation

In most cases, however, the MISE cannot be computed explicitly and therefore, an asymptotic approximation (AMISE) is used instead. The AMISE corresponds to the tailor expansion of the MISE up to a certain degree where the error term is left out.

The MISE and AMISE provide powerful methods to achieve theoretical results of the overall accuracy of an estimator. However, the AMISE is still too difficult to compute in practice because it requires knowledge of the function that should be estimated. Thus, an other error metric is required that can be used in a more easy way for practical purposes. For a query $Q(a,b)$, we therefore consider the *absolute error* and *the relative error.*

3. Selectivity Estimation with Density Estimators

In this section, we present estimators suitable for range queries on continuously distributed data sets. We first discuss histograms and then kernel estimators.

3.1 Histogram Selectivity Estimation

For metric attributes, histograms are in general only applicable to estimate the selectivity of queries if the value set of a bin is an interval of the data space. The i-th bin is then described by an interval $(c_i, c_{i+1}]$ with width $h_i = c_{i+1} - c_i$ and an integer value n_i which is the number of records in the interval $(c_i, c_{i+1}]$. For a set of n samples, the *histogram density estimator* f_H is then computed by

$$\hat{f}_H(x) = \frac{1}{n} \cdot \sum_{i=0}^{k-1} \frac{n_i}{h_i} \cdot I_{(c_i, c_{i+1}]}(x) \quad \text{and} \quad I_S(x) = \begin{cases} 1 \text{ if } x \in S \\ 0 \text{ else} \end{cases}$$

where $I_S(x)$ is also called the *indicator function*. The histogram density estimator is based on the assumption that the records are uniformly distributed in a bin. Let N be the num-

ber of records in the database and let us consider a query $Q(a, b)$. The *histogram selectivity estimator* $\hat{\sigma}_H$ for the query $Q(a,b)$ is given by inserting $\hat{f}_H$ into (2). We then obtain the following estimator

$$\hat{\sigma}_H(a, b) = \int_a^b \hat{f}_H(t)dt = \frac{1}{n} \cdot \sum_{i=0}^{k-1} \frac{n_i}{h_i} \cdot \int_a^b I_{(c_i, c_{i+1}]}(t)dt$$

$$= \frac{1}{n} \cdot \sum_{i=0}^{k-1} \frac{n_i}{h_i} \cdot \psi_i(a, b)\text{, where} \qquad (4)$$

$$\psi_i(a, b) = \begin{cases} 0 & \text{if } [a, b] \cap [c_i, c_{i+1}] = \varnothing \\ min(b, c_{i+1}) - max(a, c_i) & \text{else} \end{cases}$$

is the fraction of the intersection between the i-th bin and the query range. There are many histograms which differ in their policy on computing the boundaries of their bins. The most popular histograms are the *equi-width* histogram (where all intervals of the bins have the same size $h = h_i$), and the *equi-depth* histogram [3] (where all intervals contain the same amount of data). For the equi-width histogram and equi-depth histogram, formula (4) can be simplified to $\frac{1}{nh} \cdot \sum_{i=0}^{k-1} n_i \cdot \psi_i(a, b)$ and $\sum_{i=0}^{k-1} \psi_i(a, b)/h_i$, respectively. A more recently proposed policy is max-diff [8]. For the max-diff histogram with k bins, the k-1 adjacent pairs with maximum distance are computed and a boundary is set between each of the k-1 pairs. Experiments performed on data sets with small cardinalities [8] have shown the following results: First, the max-diff histogram provides more accurate selectivity estimations for range queries than equi-width histograms and equi-depth histograms. Second, equi-depth histograms are more efficient than equi-width histograms. Since the experiments are only related to very small domains it is still an open question how these methods perform on data sets that are from a continuous domain or from a large domain where the values occur with low frequencies.

Histograms provide a simple and easy computable estimator. For a given set of samples, the quality of a histogram estimator depends on the number of bins and on the starting point, see [10], [15] and [11]. The method itself suffers for continuous data distributions under the assumption that data is uniformly distributed in a bin. Moreover, discontinuous jump points can be observed in the boundary of two adjacent bins for $\hat{f}_H$. A first method to reduce these deficiencies is the so-called average shifted histogram. The *average shifted histogram* is not a histogram w.r.t. our definition, but a sequence of equi-width histograms with the same number of bins and different starting points. The estimation of the selectivity is simply computed by taking the average among the estimations of all equi-width histograms. The problem of jump points however still exists (however in a more diminished form).

3.2 Kernel Selectivity Estimation

The method of kernel selectivity estimation avoids the problem of discontinuous jump points completely. Moreover, the method does not require a starting point. We will first introduce a very general approach in this section and refine it to the kernel selectivity estimation. A very generalized approach to approximating the true PDF is to use the average of a weight function w that occurs once for each sample. In order to ensure that the estimate

$\hat{f}_A(x) = \frac{1}{n} \cdot \sum_{i=1}^{n} w(x, X_i)$ fulfills the properties of a density function it is sufficient to require $\int_{-\infty}^{\infty} w(x, y)dy = 1$ and $w(x, y) \geq 0$ for all x and y. This approach is called a *general weight function estimate* [11]. The basic idea of this approach is that a continuous estimation requires that the mass of a sample is distributed among its neighborhood. Note that histograms also follow the paradigm of this approach, but, as we have shown, they did not result in an estimator without discontinuous jump points. The question therefore arise what kind of functions are suitable for being a weight function of a continuous estimator?

A class of general weight functions very easy to compute is that of the kernel function estimators. This class has been well studied in statistics recently. The basic idea of this approach is to choose a weight function such that the sample is the center. Moreover, a new parameter $h \in \Re$, $h > 0$, is introduced which controls the distribution of the mass among the neighborhood of the sample. In its most general form, such a weight function can then be defined as $w(x, y) = (1/h)K((x-y)/h)$, where K is a real function which integrates to one. The *kernel density estimator* of the true PDF f with *kernel function* K and *bandwidth* h is then defined by

$$\hat{f}_K(x) = \frac{1}{nh} \cdot \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \qquad (5)$$

The kernel density estimator can be viewed as a set of superimposed "bumps" which are positioned at the samples X_i. The bandwidth h determines the impact range of these bumps, and the kernel K determines their shape. The kernel density estimator is now constructed by adding up this bumps, see figure 1.

The bandwidth h is also called smoothing parameter. On the one hand, if h becomes too large all details of the true PDF will be obscured. This effect is caused by oversmoothing the estimator with its parameter h. On the other hand, if h is chosen too small spurious structures become visible from the sample set. The right tuning of h is important to obtain an accurate estimator. This problem is addressed in all details in section 4.2.

The selection of the kernel function K is not as important as the selection of the smoothing parameter h. It has been shown [13] that varying the kernel function K causes only small effects on the accuracy of the estimator in comparison to varying h. Therefore, we are only interested in a suitable kernel function K that is inexpensive to compute. Among the rich source of kernel functions, the *Epanechnikov kernel* is used in our approach. The Epanechnikov kernel is defined

as

$$K(t) = \begin{cases} \frac{3}{4}(1-t^2) \text{ if } |t| \le 1 \\ 0 \quad \text{else} \end{cases}.$$

We obtain the kernel estimator by first inserting the kernel function into (5). Then, the density estimator in equation (2) has to be substituted by (5). Overall, we obtain

$$\hat{\sigma}_K(a,b) = \int_a^b \frac{1}{nh} \cdot \sum_{i=1}^{n} K\left(\frac{x-X_i}{h}\right) dx = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h} \cdot \int_a^b K\left(\frac{x-X_i}{h}\right) dx$$

With the substitution $t = (x-X_i)/h$ in the integral we obtain

$$\hat{\sigma}_K(a,b) = \frac{1}{n} \sum_{i=1}^{n} \int_{(a-X_i)/h}^{(b-X_i)/h} K(t)dt \,. \qquad (6)$$

In (6) the integral will very often return one or zero. The different cases are illustrated in figure 2 where the grey area shows the portion of the kernel functions that contribute to the selectivity estimation of query $Q(a,b)$. The result of the integral expression will be zero, if $[(a-X_i)/h, (b-X_i)/h]$ does not have a common intersection with $[-1, 1]$, see the kernel function at sample X_1 in figure 2. If $[(a-X_i)/h, (b-X_i)/h]$ completely overlaps $[-1, 1]$, the result of the integral is one (see the kernel function at sample X_3). Only for those cases where $(a-X_i)/h \in [-1, 1]$ or $(b-X_i)/h \in [-1, 1]$ (see the kernel function at X_2) we need an explicit computation of the integral.

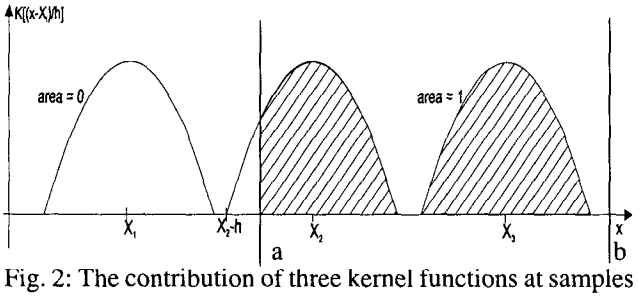

Fig. 2: The contribution of three kernel functions at samples X_1, X_2 and X_3 on the selectivity estimation of $Q(a,b)$.

Therefore, we split the sum in (6) into two parts. The first part only counts those integrals, which completely overlap the interval $[-1, 1]$, i.e. each of this integrals return one. The second part contains those integrals, which need to be evaluated explicitly. Then,

$$\hat{\sigma}_K(a,b) = \frac{1}{n} \cdot \left(\sum_{i=1}^{n} I_{[a+h,\, b-h]}(X_i) + \sum_{i=1}^{n} \int_{(a-X_i)/h}^{(b-X_i)/h} K(t)dt \cdot I_{[a-h,\, a+h] \cup [b-h,\, b+h]}(X_i) \right)$$

The formula can further be simplified by replacing the integral of the Epanechnikov kernel function by its primitive

$$F_K(t) = \begin{cases} \frac{1}{4}(3t-t^3) \text{ if } |t| \le 1 \\ 0 \quad \text{else} \end{cases}.$$

The total computation of the kernel selectivity estimator is given in algorithm 1. The cost of this algorithm is $\Theta(n)$ where n denotes the number of samples. In general, this can be improved by using an appropriate data structure for the sample set. For example, when a balanced binary search tree is used, the cost of the algorithm is given by $O(\log n + k)$ where k denotes the number of samples that are in the range $[a-h, b+h]$.

```
Given:
  n:      number of samples
  X[i], i=1..n:sample set
  [a,b]:  range of the query
  h:      bandwidth
  F(t):   integrated kernel function

Algorithm
  s = 0.0;
  FOR i=1 TO n DO
  {  IF (X[i] ∈ [a+h,b-h])
        s += 1.0;
     ELSE IF (X[i] ∈ [a-h,a+h] && ∉ [b-h,b+h])
        s += 0.5 - F((a-X[i])/h);
     ELSE IF (X[i] ∈ [b-h,b+h] && ∉ [a-h,a+h])
        s += F((b-X[i])/h) - 0.5;
     ELSE IF (X[i] ∈ [b-h,b+h] ∪ [a-h,a+h])
        s += F((b-X[i])/h) - F((a-X[i])/h);
  }
  RETURN s/n;
```

Alg. 1: Kernel seletivity estimator with boundary treatment

3.2.1 The Boundary Problem

For kernel estimation methods, the results of experiments with different sample sets and different distributions have revealed high estimation errors for range queries which are positioned close to a boundary of the domain. The error at the boundaries was considerably higher than the one of a range query positioned in the center of the domain. This effect is illustrated in figure 3 where the absolute estimation error (with sign) is shown for range queries whose range is 1% of the domain. The data set consists of 100,000 uniformly distributed elements and therefore, the response set of a 1% query consists of 1,000 elements. The curve depicts the estimation error as a function of the position of the query. For example, the error of the center is very low, whereas close to the boundary an absolute error of up to 500 occurs. Moreover, it has also been shown in other experiments that the error at the boundary increases with an increasing smoothing parameter.

The behavior we observed occurs for almost all nonparametric estimation methods based on kernel functions. There are mainly two reasons: First there is no information about the true PDF beyond the boundaries of the domain. Since

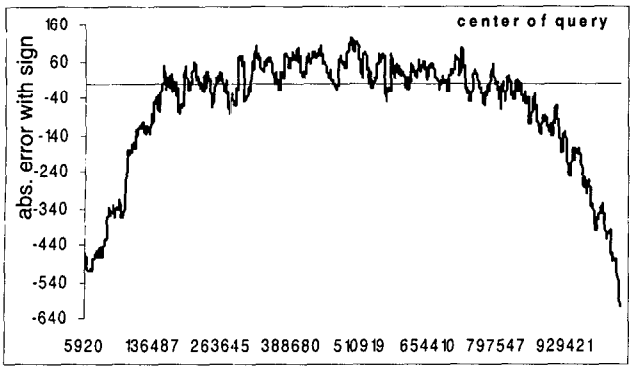

Fig. 3: The absolute estimation error of 1% queries as a function of the query position (uniform data distribution)

the kernel estimator is based on a continuous estimation of the true PDF the estimator will deliver values greater than zero outside of the domain close to the boundaries. This leads to a "loss of weight" in the domain and therefore, the estimation of the density function does not integrate over the domain to one. Second, due to the lack of data the estimator is not consistent for x nearby the boundaries if $x \in [l, l+h) \cup (r-h, r]$ where l and r denotes the left and the right boundary, respectively. The main difficulty in solving these problems is that there are two conflicting goals. In order to fulfill the property of a density function the lost weight of the estimation has to be shifted back into the domain. However, this violates the requirement that the estimator should be consistent. In order to fulfill consistency for points close to the boundaries, the inequality $\int_{\infty}^{-\infty} \hat{f}(x)dx > 1$ is fulfilled with high probability. In the following we present two different methods to avoid the high estimation errors at the boundaries.

The first method provides an estimator that is a density function, but it does not fulfill consistency. The method simply mirrors the samples $X_i \in [l, l+h)$ and $X_i \in (r-h, r]$ at the left and right boundary, respectively. Hence, these samples are considered twice. This method is termed *reflection technique*.

The second method provides a consistent estimator, but it does not fulfill the property of a density function. The basic idea is to use special kernel functions on samples which are close to the boundary. There are only a few proposals for a boundary kernel function, see for example [15]. In our experiments we used the family of boundary kernel functions of [14] because the computation of the primitives was rather simple. For the left boundary, the family of kernels is then defined by

$$K^{(l)}(x, q) = \frac{3 + 3q^2 - 6x^2}{(1+q)^3} \cdot I_{[-1, q]}(x)$$

with $q = (x-l)/h$. A similar family of kernel functions $K^{(r)}(x, q)$ is also available for the right boundary. These kernel functions are used for all $x \in [l, l+h) \cup (r-h, r]$ i.e. they must be considered for all samples X_i with $X_i \in [l, l+2h) \cup (r-2h, r]$. To obtain the primitives for the boundary kernel functions the dependence of their limits on q must be eliminated.

Suppose in the following that the left boundary is zero (l = 0). Then, the family of boundary kernel functions are required for computing an estimation of the selectivity between 0 and h. This means q is a monotone function of x where $q(0) = 0$ and $q(h) = 1$. For each $x \in [0, h]$ the shape of the used boundary kernel function varies with q. In regions without boundary effects the ordinary Epanechnikov kernel function is used. Let $Q(a, b)$ be a range query with $0 \le a \le h$ and $b<h$. Then, the kernel selectivity estimation is computed by the following sum:

$$\hat{\sigma}(a, b) = \frac{1}{nh} \cdot \sum_{i=1}^{n} \left(\int_a^h K^{(l)}\left(\frac{x - X_i}{h}, q\right) dx + \int_h^b K\left(\frac{x - X_i}{h}\right) dx \right)$$

3.3 A Hybrid of Histogram and Kernel Estimator

Kernel estimators are among the most accurate estimators in statistics under the assumption that the underlying PDF is sufficiently smooth. In general, this assumption does not hold in practice. Experiments have revealed high errors for kernel estimators on those points where the true PDF changes considerably. These points are also called *change points* in statistics [16].

The hybrid estimator uses the change points to partition the data space into histogram bins. Adjacent bins are merged into one if the corresponding number of records is not sufficiently large. Inside each of the bins the original kernel estimation method is used. In particular, the bandwidth of the kernel estimator is individually chosen for every bin. The only remaining problem is now to detect change points.

In the following we use the second derivative to detect change points. In our approach, the first change point corrsponds to the point where the maximum of the second derivative occurs. Further change points can be computed similarly in a recursive fashion. There are two reasons for this spproach: First, there are considerable changes of the first derviative of the PDF around a change point. Second, the asymtotic error of kernel estimators depends on the second derivative (see [11] and our discussion in section 4.2). Therefore, eliminating the maxima of the second derivatice can reduce the estimation error.

The performance of the hybrid estimator method depends on the accurancy of the computed set of change points. It is left to future work whether other methods for change point detection are more effective for the hybrid estmator.

4. Selection of the Smoothing Parameter

In this section, we discuss the impact of the number of samples and the smoothing parameter h on the quality of the estimator. In particular, we derive a rule for computing the optimal h as a function of the number of samples (w.r.t. a certain error metric). The discussion is not limited to kernel

estimation, but we also show for equi-width histograms that the number of bins is an important parameter for the error size. For a given number of samples, too less bins as well to many result in high errors of the histogram estimator. Similar to kernel estimation, we derive a rule for the optimal number of bins as a function of the number of samples such that a certain error is minimized.

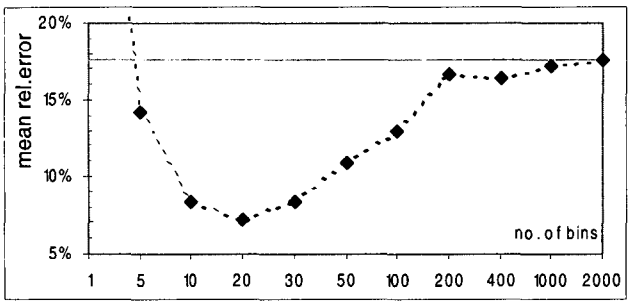

Fig. 4: Dependence of the mean relative error on the number of bins

In order to motivate the discussion, let us first take a closer look on figure 4 where the results of a set of experiments are presented. The curve shows the average relative error of range queries as a function of the number of bins. The database consists of 100,000 Normal-distributed records where 2,000 of them were used as samples for building the histogram. The query size was set to 1% of the domain. The position of the queries followed the underlying data distribution. The dotted curve shows the error for the equi-width histogram, whereas the straight line (17.5%) gives the relative error of pure sampling. The relative error of the equi-width histogram is for a few bins higher than the error of pure sampling. The minimum (relative error of 7%) is achieved for 20 bins. For an increasing number of bins the error increases up to the sampling error.

The problem of computing the optimal number of bins does not only occur for equi-width histograms, but it also arises for other histograms as well (e. g. equi-depth and max-diff). The impact of the number of bins on the estimation error has been mentioned in [9]. Except of [5], however, we are not aware that this serious problem has really been addressed in the context of selectivity estimation.

The problem of computing the optimal number of bins of an equi-width histogram is equivalent to computing the optimal width of a bin. The width of a bin is closely related to the smoothing parameter of a kernel estimation and therefore, similar techniques can be used for computing an optimal estimator. For a given sample set, the optimal estimator is defined as the one which minimizes the MISE, see equation (3) in section 2. Since the computation of the MISE requires detailed knowledge about the real distribution, the MISE is not a practical optimization criterion. Instead, an asymptotic approximation of the MISE (*AMISE*) is used. In order to obtain an approximation we assume that the PDF is sufficiently differentiable. The AMISE then corresponds to the tailor expansion of the MISE up to a certain degree where the error term is left out. The AMISE still requires a few parameters which are determined by the real distribution. These parameters however can be estimated reasonably well. The AMISE only depends on the parameters of the underlying estimator which are the smoothing parameter and the number of samples. For a given sample, the minimum of the AMISE with respect to the smoothing parameter can then be computed by using a standard technique. As a result, we obtain the optimal smoothing parameter as a function of n (the number of samples).

In the following subsections we present the formulas for the smoothing parameter of histograms (*bin width*) and kernel estimators (*band width*), respectively. The proofs for the formulas presented in the following are given for example in [11].

4.1 Bin-width Selection for Histogram Estimation

In this section, we consider an equi-width histogram where h denotes the width of a bin. Let n be the number of samples. The asymptotic approximation of the MISE (AMISE) of the histogram estimator is then

$$AMISE(h) = \frac{1}{nh} + \frac{h^2}{12}\int_{-\infty}^{\infty}\left(\frac{df}{dx}(x)\right)^2 dx$$

By solving $\frac{d}{dh}AMISE(h_{EW}) = 0$ we obtain the asymptotically optimal bin width:

$$h_{EW} = \left(\frac{6}{n\int_{-\infty}^{\infty}\left(\frac{df}{dx}(x)\right)^2 dx}\right)^{1/3} \quad (7)$$

Under the assumption made above it follows that $AMISE(h_{EW}) = O(n^{-2/3})$ and hence, that the convergence rate of the equi-width histogram is higher than the one of pure sampling.

The computation of h_{EW} requires the derivative of f (the true PDF) which is generally not known. There are different techniques to approximate the true PDF in (7). A simple technique called *normal scale rule* uses the Normal distribution for approximating the real distribution. The intuition behind the rule is that due to the theorem of large numbers the density of an arbitrary distribution converges to the Normal distribution. Then,

$$h_{EW} \approx (24\sqrt{\pi})^{1/3} \cdot s \cdot n^{-1/3} \quad (8)$$

follows where s denotes the standard deviation of f. There are two common techniques for estimating the parameter s. First, the standard deviation of the sample set can be used. In practice, it has been observed that this estimator leads to an oversmoothing of the true PDF. Second, the interquartile range (which is the distance between the 0.75 quantile and the 0.25 quantile) of the sample set can also be used to approximate the standard deviation. In our implementation, we decided to estimate s by taking the minimum of both.

4.2 Bandwidth Selection for Kernel Estimation

Let us now consider the problem of computing the optimal bandwidth of a kernel estimation. Similar to computing the optimal size of a bin, an asymptotic approximation is derived from the tailor expansion of the MISE. The analysis is not limited to the Epanechnikov kernel function, but it can be applied to a symmetric kernel function K that fulfills the following conditions:

(a) $\int K(t)dt = 1$

(b) $\int tK(t)dt = 0$

(c) $k_2 = \int t^2 K(t)dt \neq 0$

The Epanechnikov kernel function fulfills these conditions with $k_2 = 1/5$. Under the assumptions that the true PDF f has continuous derivatives of all orders required and that the sample set is given, the following formulas can be derived from the tailor expansion for the *asymptotic integrated bias (AIBias)* and the *asymptotic integrated variance (AIVar)*.

$$\text{(a) } AIBias(h) = \sqrt{\frac{1}{4}h^4 k_2^2 \int \left(\frac{d^2 f}{dx}(x)\right)^2 dx}$$

$$\text{(b) } AIVar(h) = \frac{1}{nh}\int K(t)^2 dt \qquad (9)$$

According to equation (3) the AMISE is then given by $AMISE(h) = AIBias(h)^2 + AIVar(h)$. A fundamental problem of density estimation is the complementary impact of h on bias and variance (9). For a small h, the bias is small and the variance is high, whereas for a large h the bias is high and the variance is small.

By solving $\frac{d}{dh}AMISE(h_K) = 0$ we obtain the asymptotically optimal bandwidth $h_K = \left(\frac{\int K(t)^2 dt}{nk_2^2 \cdot \int \left(\frac{d^2 f}{dx}(x)\right)^2 dx}\right)^{1/5}$.

It follows that $AMISE(h_K) = O(n^{-4/5})$. Hence, the convergence rate of kernel estimation is higher than the one of equi-width histograms.

The formula for the optimal bandwidth still depends on the second derivative of the unknown PDF Again we can use the normal scale rule to approximate the true distribution by the Normal distribution. For the Epanechnikov kernel function ($k_2 = 1/5$), we obtain $h_K \approx 2.345 \cdot s \cdot n^{-1/5}$ for the optimal bandwidth, where s is estimated by using the minimum of the empirical standard deviation of the sample set and the interquartile range divided by 1.348.

4.3 Direct Plug-in Methods

We briefly mention here a technique for improving the estimation techniques presented above. The technique can be used as an alternative to the normal-scale rule. The so-called direct plug-in rule [15] estimates the true density function in an iterative fashion. In an iteration step, the approximation of the density function of the previous iteration is used to compute h_K (h_{EW}). This results again in a new approximation of the density function. In the first iteration, the normal-scale rule can be used to obtain a first estimation of the PDF. The number of iteration steps is a new parameter of the plug-in rule. The influence of the normal scale rule diminishes for an increasing number of iterations. In general, two or three iteration steps are sufficient.

5. Experiments and results

In this section, we report the results of a performance comparison of different methods for estimating the selectivity of range queries. The objective of our experiments was to find out how much of the theoretical results can also be confirmed in practice. We first show the impact of the domain cardinality and of the sample size on the estimation error. In the next set of experiments, we compare the performance of different histogram estimators. In particular, we discuss the problem of computing the number of bins for a given sample set. The next set of experiments deals with kernel estimation. In particular, we investigate the methods for reducing the boundary errors and the rules for computing the smoothing parameter.

5.1 Test Environment

In the following, we first present our test environment in all details (distribution of the data files, distribution of the query files). All the files are freely available [17].

5.1.1 Data Files

In our experiments we used sets of artificial data as well as sets of real data. The artificial data sets follow the Uniform distribution, the standard Normal distribution and the Exponential distribution. The Exponential distribution can be considered as a substitute of the Zipf distribution which is commonly used in experiments. Both are highly skewed distributions with high density at the left boundary of the domain and low density at the right boundary. Each of the files that follow an artificial distribution consists of 100,000 records. The real data sets consist of data derived from the TIGER/Line files from the U.S. Census Bureau [18]. We used the first and second dimension of the endpoints of lines from county Arapahoe and the endpoints of lines from an area around L. A. where the lines represent rail road tracks and rivers. Another set of real data corresponds to the instance weight of a census-income file [19]. The domain of the data files corresponds to integer values in the range from 0 to 2^p-1, where p is considered as a parameter. For the data sets that follow a Normal distribution, we mapped the records to the integer domain such that the mean value is in the center of the domain. We did not consider data records that were outside of the domain. Correspondingly, we also mapped the data from the Exponential distribution to the integer domain. The most important properties of our files are summarized in table 2.

From each of these data sets we have drawn sample sets of 2,000 records by selecting the records from the file in a random fashion without replacement.

5.1.2 *Query Files and Error Metrics*

The query files in our experiments differ from the ones generally used in other experiments (see for example [8]) since the query size is fixed for all queries of a file. The reason for such size-separated query files is that we are interested in the impact of the size of a query on the estimation error.

For each data set D we generated four query files where each of them contains 1,000 range queries of a fixed size. The size of a range query varies between 1%, 2%, 5% and 10% of the size of the underlying domain. We use the notation $F_D(s)$ to refer to a query file with queries of size s. The position of the queries follows the same distribution as the corresponding data records. Query positions which are too close to the boundary of the domain are not accepted in order to avoid queries being partially outside of the domain.

data file	data distribution	p	#records
u(p)	Uniform	{15,20}	100,000
n(p)	Normal	{10,15,20}	100,000
e(p)	Exponential	{15,20}	100,000
arap1	Arapahoe, 1st dim.	21	52,120
arap2	Arapahoe, 2nd dim.	18	52,120
rr1(p)	Rail road & Rivers, 1st dim.	{12,22}	257,942
rr2(p)	Rail road & Rivers, 2nd dim.	{12,22}	257,942
iw	Instance Weight	21	199,523

Table 2: Properties of the data files

In the following, we consider for query files $F_D(s)$ the mean relative error $MRE(D, s)$ defined by

$$MRE(D,s) = \frac{1}{|F_D(s)|} \cdot \sum_{Q(a,b) \in F_D(s)} \frac{||Q(a,b)| - \hat{\sigma}(a,b) \cdot |D||}{Q(a,b)}$$

For example, MRE(rr1(12), 1%) denotes the mean relative error of 1% queries performed on the data file rr1(12). We also considered the mean absolute error in our experiments. The behavior of the absolute error was not much different to the relative error and therefore, we only present the relative error in the following.

5.2 Results

In the following, we discuss the most interesting results we obtained from the experiments. First, we show the impact of different parameters (cardinality, query size, sample size) on the accuracy of the selectivity estimator. Second, we report the results from a comparison of histogram estimators and kernel estimators. We conclude the section with a comparison of the most promising selectivity estimators.

5.2.1 *The impact of the domain cardinality*

In our first set of experiments, we deal with the impact of the domain cardinality on the estimation error. In figure 5 we depict the mean relative error for an equi-width histogram as a function of the number of bins. The three curves refer to the data sets n(10), n(15) and n(20) (Normal distributions). Data sets from a small domain contains more duplicates than sets from a larger domain. As shown in figure 5, the error is considerably higher for large domain cardinalities. Similar results were obtained for the other data files. Because our emphasis is on metric attributes with large domains, we omit in the following the results we obtained from files with high frequencies.

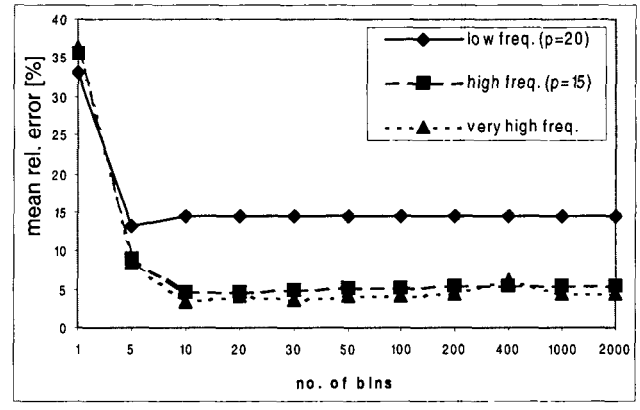

Fig. 5: The MRE as a function of the number of bins for different domain cardinalities

5.2.2 *The impact of the sample size*

One of our most important requirements has been that a selectivity estimator should be consistent, i. e., the estimation error decreases when the sample size increases. From the theory we know that pure sampling and the equi-width histogram (with an adaptive number of bins) are both consistent estimators. This is also confirmed by the results of our experiments. Figure 6 shows the mean relative error MRE(n(20), 1%) for pure sampling, equi-width histogram and kernel estimator as a function of the sample size. For the equi-width histogram, for example, the mean relative error is close to 12% for a sample size of 200, whereas for a sample size of 10000 the MRE is only about 4%. The curves also show that kernel estimators are more accurate than histograms and histograms are more accurate than pure sampling. Hence, these results are in agreement with the theory.

5.2.3 *The impact of the query size*

In the next set of experiments we discuss the influence of the query size on the accuracy of selectivity estimation. We only report the results obtained from experiments of equi-width histograms with normal scale rule. Similar results are achieved for other selectivity estimators. In figure 7, the MRE is depicted for different data files and query files. As expected the error decreases when the query size increases.

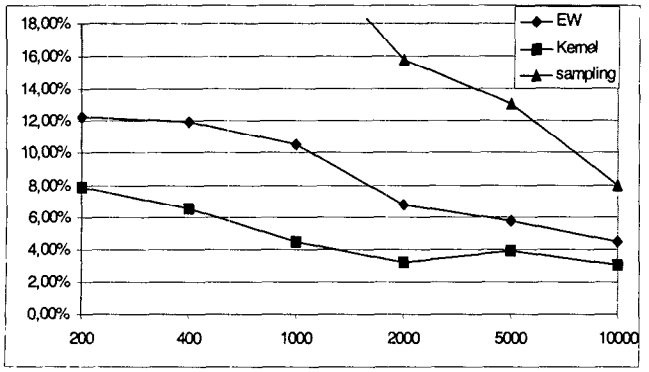

Fig. 6: The MRE(n(20), 1%) as a function of the sample size for sampling, equi-width histograms and kernel estimators

For the data file arap2, for example, the MRE of a 10% query is only 4.5%, whereas the MRE of a 1% query is 17.5%. In the following, we only report the results for small queries.

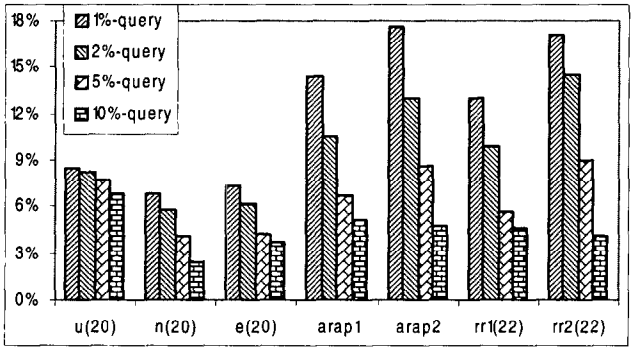

Fig. 7: The MRE of equi-width histograms for different query files

5.2.4 A comparison of histogram estimation methods

Next, we compare the performance of different histogram estimators. In particular, we are interested in the impact of the selection of the bin width on the accuracy of the estimators.

In a first set of experiments we compare the MRE obtained for equi-width histograms (*EWH*), equi-depth histograms (*EDH*), max-diff histograms (*MDH*), pure sampling (*sample*) and the uniform estimator (*uniform*) that corresponds to a histogram with one bin. In figure 8 the MRE is depicted for different data files. For all histogram methods, we used for each query file the optimum number of bins we obereved in our experiments. The results therefore represent the best case of the kernel estimator. The overall loser of our comparison is the uniform estimator (except for uniform data distribution). For the data file *ci*, for example, the MRE of the uniform estimator is 600%, whereas all other methods produce an error of about 5%. In general, the equi-width histogram is the winner. This result is not in common with the results reported in most previous experiments. For example, in [8] it was found that max-diff is considerable superior in comparison to equi-width and equi-depth histo-

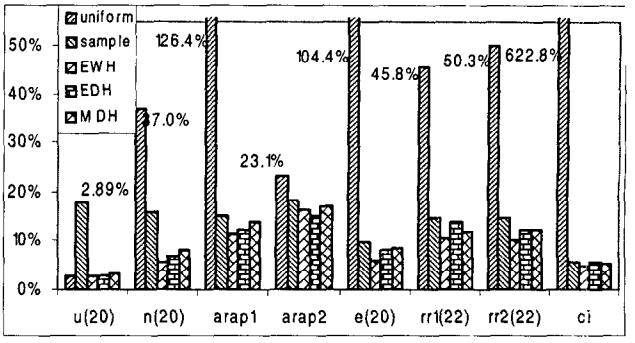

Fig. 8: Average relative error for different histogram estimators compared with sampling and uniform assumption

grams. Pure sampling generally provides estimations with higher errors than histogram estimators for our artificial data sets. For our real data sets, the errors of pure sampling are slightly higher in comparison to histogram estimators.

A second set of experiments was performed to investigate the impact of the number of bins on the estimation error of equi-width histograms. We did not consider other types of histograms because we are not aware of a theory that suggests how to determine the number of bins for equi-depth histograms and max-diff histograms. In general, we observed that the number of bins determined for an equi-width histogram (using for example the normal scale rule) is also reasonable for other histograms. In figure 4, we already have shown that there is a strong relationship between the number of bins and the estimation error. In this section, we are interested in whether the derived rule for computing the number of bins (see equation (8)) is close to the optimal number actually observed in the experiments. In figure 9, the estimation error (MRE) of the equi-width histogram is depicted for different data files. For each data file, there are two columns. The first column shows the MRE of the histogram with an optimal number of bins observed (*h-opt*) and the second column refers to an histogram where the number of columns is computed by using the normal scale rule (*h-NS*). For this rule the estimation error is on the average about 3% higher in comparison to an histogram with the optimal number of bins.

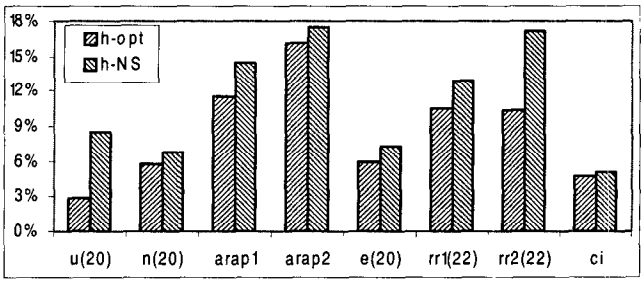

Fig. 9: The MRE of equi-width histograms with different policies for computing the number of bins.

5.2.5 A comparison of kernel estimation methods

Next, we report the results of our experiments for kernel

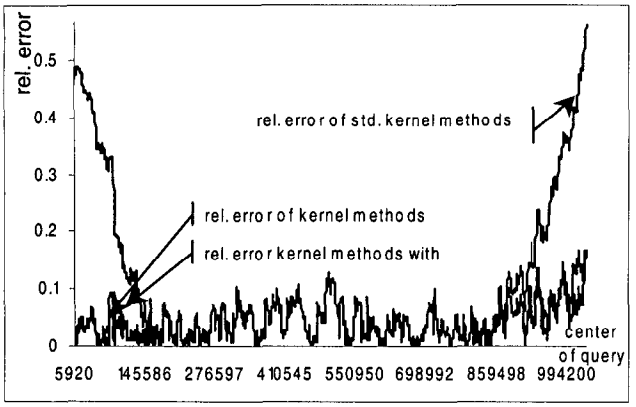

Fig. 10: The relative estimation error of 1% queries as a function of the query position (uniform data distribution).

estimation methods. In particular, we consider the boundary problem as well as the rules for computing the bandwidth.

In our first set of experiments, our focus is on the boundary problem. In figure 10, we depict the relative error of 1% queries as a function of the query position for uniformly distributed data and different kernel estimation methods. The one curve shows the relative error when the boundary problem is not treated. The other curves depict the relative error when the reflection technique and the boundary kernel function is used, respectively. Both approaches lead to a considerable reduction of the estimation errors. In almost all cases the kernel selectivity estimator with boundary kernel functions performs slightly better than the one with the reflection technique.

In our next set of experiments, we compared different techniques for computing the bandwidth of the kernel estimation method. The first technique computes the bandwidth with the lowest MRE. This is not a practical method because it requires that the queries and the sizes of their response sets are known in advance. This method only serves to judge the quality of the other techniques. The other techniques determine the bandwidth by using the normal scale rule and the direct plug-in rule (with 2 iteration steps), respectively. In figure 11, the MRE is shown for different data files and 1% queries. For each data file, the three columns refer to the optimal bandwidth (*h-opt*), the bandwidth of the normal scale rule (*h-NS*) and the bandwidth of the direct plug-in rule (*h-DPI2*). The left column depicts *h-opt*, the column in the middle shows the results of *h-NS* and the right column refers to *h-DPI2*. Each of the kernel estimation methods uses special boundary kernel functions. As shown in figure 11 the normal scale rule results in a low MRE for all synthetic data distributions. In these cases it is slightly superior to the direct plug-in technique. However, the normal scale rule produces high errors for all our real data sets, whereas the direct plug-in rule clearly outperforms the normal scale rule. For real data sets, the MRE of the plug-in technique is however still higher (up to 5%) than the MRE in case of an optimal bandwidth selection.

5.2.6 Comparison of the most promising estimation methods

Let us now present a direct comparison of the most promising estimation methods. In addition to the methods previously discussed, this comparison also includes avergage shifted histograms. In figure 12, we present the MRE of 1% queries for the different data files. For each data file, we report the results of the following methods:

- equi-width histogram using the normal scale rule for computing the number of bins (*EWH*),
- kernel estimators using boundary kernel functions and the direct plug-in rule for computing the bandwidth (*Kernel*),
- hybrid estimators using boundary kernel functions (*Hybrid*),
- average shifted histograms using ten shifts (*ASH*).

The results of our synthetic data sets (u(20), n(20), e(20)) show that the kernel estimator produce the most accurate results. The error of the average shifted histogram is only slightly higher than the one of the kernel estimators. For our real data sets from the TIGER/Line database, the methods perform differently. Now, the hybrid estimator gives the most accurate results, whereas kernel estimators and equi-width histograms produce high errors. For the real data file *ci*, there is almost no difference in the performance of the different methods.

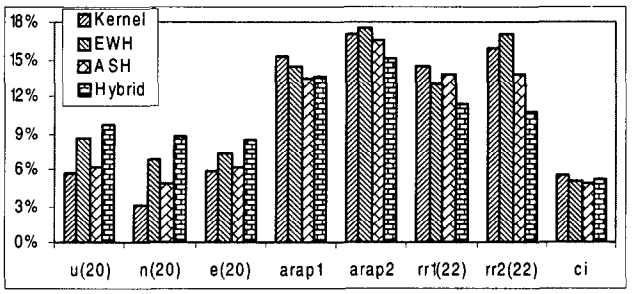

Fig. 12: A compraison of the most promising estimators for 1%queries

6. Conclusions

In this paper, we outlined several nonparametric methods for estimating the selectivity of range queries based on two different statistical approaches (histogram and kernel esti-

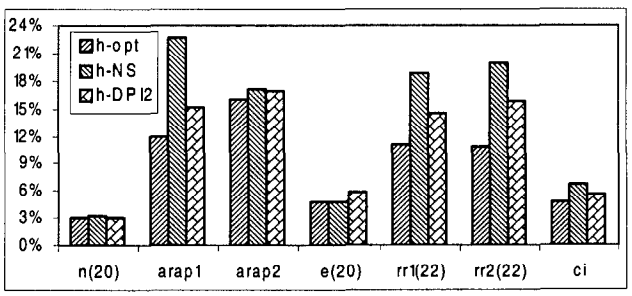

Fig. 11: Average relative error for kernel estimation methods which differ in the bandwidth selection technique.

mators). We considered the accuracy of the selectivity estimators in theory and practice. In particular, we proposed to use kernel estimators which are among the most accurate estimators in statistics. Kernel estimators can be viewed as a generalization of sampling where a sampling point distribute its mass among its neighborhood. The bandwidth of the kernel estimator control the size of the impact ranges of its samples and a kernel function is responsible how the mass of the samples is distributed. We showed that kernel estimators are inexpensive methods that produce fairly accurate results. Kernel estimators produce the most accurate estimations among all estimation methods we considered in the paper under the following conditions: the underlying distribution function is smooth and the cardinality of the data space is large. Data sets in spatial databases are examples where these conditions are (almost) fulfilled. However, even in case that these properties are not completely fulfilled, kernel estimation methods are comparable to other estimation methods currently used in database systems. For highly skewed data distributions, however, kernel estimator suffer under the discontinuity jump points of the density function. We therefore proposed a hybrid estimator that is a combination of histogram and kernel estimator. Experiments confirmed that the hybrid estimator gives more accurate results than the pure kernel estimator and different types of histograms.

In general, only a small sample set is used to create a histogram or a kernel estimator. An important problem is then to determine the so-called smoothing parameter. For histograms, the smoothing parameter corresponds to the number of histogram classes. For example, results of experiments showed high errors when the number of bins is too small or too high. Today commercial database system (e. g. ORACLE) does not provide any help to the user for finding the optimal number of bins. We proposed several rules for computing approximations of the optimal number of bins. The results of experiments generally confirmed that the smoothing parameters obtained from the rules are close to the optimal ones.

In our future work, we are interested in the following problems. First, we will consider multidimensional kernel estimators to estimate the selectivity of multidimensional range queries. Second, we currently investigate how to apply kernel estimators to online processing of aggregate queries [6]. Third, we will include the knowledge of previous queries to improve the quality of kernel estimators [1].

7. Acknowledgment

We would like to thank Professor Volker Mammitzsch (University of Marburg) and Professor Müller (University of California at Davis) for their valuable discussions on nonparametric estimation methods. We also wish to thank Jochen van den Bercken for his comments on a previous version of the paper.

8. References

[1] Chen, C.M. & Roussopoulos, N. "Adaptive Selectivity Estimation Using Query Feedback" Proc. ACM-SIGMOD Intl. Conf. on Management of Data 1994.

[2] Ioannidis, Yannis E. & Christodoulakis, S. "Optimal Histograms for Limiting Worst-Case Error Propagation in the Size of Join Results" TODS 18(4) 1993. 709-748.

[3] Gregory Piatetsky-Shapiro & Charles Connell. "Accurate Estimation of the Number of Tuples Satisfying a Condition" SIGMOD Conference 1984. 256-276.

[4] Yossi Matias, Jeffrey Scott Vitter, Min Wang. "Wavelet-Based Histograms for Selectivity Estimation" SIGMOD Conference 1998. 448-459.

[5] Surajit Chaudhuri, Rajeev Motwani, Vivek R. Narasayya. "Random Sampling for Histogram Construction: How much is enough?" SIGMOD Conference 1998. 436-447.

[6] Joseph M. Hellerstein, Peter J. Haas, Helen Wang. "Online Aggregation" SIGMOD Conference 1997. 171-182.

[7] H. V. Jagadish, Nick Koudas, S. Muthukrishnan, Viswanath Poosala, Kenneth C. Sevcik, Torsten Suel. "Optimal Histograms with Quality Guarantees" VLDB 1998. 275-286.

[8] Viswanath Poosala, Yannis E. Ioannidis, Peter J. Haas, Eugene J. Shekita. "Improved Histograms for Selectivity Estimation of Range Predicates" SIGMOD Conference 1996. 294-305.

[9] Michael V. Mannino, Paicheng Chu, Thomas Sager. "Statistical Profile Estimation in Database Systems" Computing Surveys 20(3) 1988. 191-221.

[10] Gasser, T. & Engel, J. & Seifert, B. „Nonparametric function estimation" in: Rao (Ed.),"Handbook of Statistics Vol. 9", North Holland 1993.

[11] David W. Scott. "Multivariate Density Estimation" Wiley & Sons 1992.

[12] Selinger, P.G. & Astrahan, M.M. & Chamberlin, D.D. & Lorie, R.A. & Price, T.T. „Access path selection in a relational database management system" Proc. ACM SGMOD Conference 1979. 23-34.

[13] Silverman, B.W. „Density Estimation for Statistics and Data Analysis" Chapman & Hall 1986.

[14] Simonoff, J. & Dong, J. "The Construction and Properties of Boundary Kernels for Sparse Multinomials" Journal of Computational and Graphical Statistics 1994.

[15] Wand, M.P. & Jones, M.C. „Kernel Smoothing" Chapman & Hall 1995.

[16] Brodsky, B.E. & Darkhovsky, B.S. "Nonparametric Methods in change-point problems" Kluwer Academic Publishers 1993.

[17] http://www.mathematik.uni-marburg.de/DBS/download/data.html

[18] http://www.tiger.gov/

[19] http://www.kdnuggets.com/datasets.html