Analyzing Quantitative Databases: Image is Everything

Amihood Amir* — Bar-Ilan University and Georgia Tech

Reuven Kashi† — Bar-Ilan University

Nathan S. Netanyahu‡ — Bar-Ilan University and Univ. of Maryland

Abstract

Traditional statistical methods deal with corroborating given hypotheses on a given body of data. However, generating the hypothesis itself is a matter of intuition and ingenuity. It is clearly impossible to test all hypotheses on a database with millions of records and hundreds of fields.

There have been attempts to bridge this gap through data mining. Association generation is a method of creating such statistical hypotheses for binary data. For quantitative databases the situation is still not good. There are a number of known methods. One is a reduction to binary data by creating intervals and then generating associations. This method is computationally expensive. Another suggested method was by generating associations that are statistically interesting. This method also was tried only on small databases and is applicable only for binary relations, e.g., in certain ranges of field X, field Y lies significantly outside its average.

We suggest a method that answers some of the problems with the current techniques. Our idea is based on using visualization techniques and image processing ideas to rank subsets of fields according to the relation between them in the database. This ranking suggests the hypotheses to be statistically investigated.

Our method has the following advantages:

1. It is scalable. Our algorithm is mainly based on analyzing histograms of the data set, thus is more efficient. It is also naturally suitable for sampling.
2. It is generalizable in the size of the set of fields. No current method handles more than a binary relation.
3. It affords comparability between fields over different base sets. This allows a uniform scale for different sets of fields in different databases.

In this paper we present an algorithmic methodology and the results of its application to the census bureau data bases, cpsm93p and nhis93ac.

1 Introduction and Background

Recent years witnessed the mushrooming and evolving of the field of *Knowledge Discovery in Databases* (KDD), also known as *Data Mining*. The main goals of data mining are an automatic hypotheses

*Department of Mathematics and Computer Science, Bar-Ilan University, 52900 Ramat-Gan, Israel, (972-3)531-8770, `amir@cs.biu.ac.il`. Partially supported by NSF grant CCR-96-10170

†Department of Mathematics and Computer Science, Bar-Ilan University, 52900 Ramat-Gan, Israel, (972-3)531-8407, `kashi@cs.biu.ac.il`.

‡Department of Mathematics and Computer Science, Bar-Ilan University, 52900 Ramat-Gan, Israel, (972-3)531-8865, `nathan@cs.biu.ac.il`, and Center for Automation Research, Univ. of Maryland, College Park, MD 20742.

generation about the data and recognizing subsets of the data in which the hypothesis holds and is considered useful for the user, in the appropriate context. More specifically, data mining deals with the search for interesting patterns and structures in large data sets and in the discovery of useful information not explicitly present in the data.

There has been extensive work towards answering the major problem of identifying what should be considered as "interesting" patterns and information within vast amounts of data. Many metrics to measure interestingness in specific contexts where suggested. Notable examples are measuring the most interesting rules [16], and visual feedback measuring the relevance of answers to queries [20]. A major issue is whether such patterns can be defined independently of the data set domain. It is also essential to test the validity of the patterns being discovered by the proposed methods, since in any large volume of data there can be patterns that are reported by the extracting process to be interesting although they are spurious ones and of no real interest. Thus, the usefulness of the discovered patterns and subsequently, the whole mining process, is negligible if meaningful information or interesting knowledge cannot be extracted from it.

Another important issue is the type of attributes in the relation that we investigate. Attributes can be *categorical* or *numeric*, which affects the type of information one is interested in, the measurement of goodness of the extracted information, and the computational complexity of the mining process. Categorical attributes, also referred to as binary or boolean, are attributes whose domain in the relational data set is a single value which either exists within the attribute's field in a data record or does not. For example, market basket data, where each record represents a list of products, can be thought of in terms of categorical data. A field representing a product is assigned 1 if the product was purchased by a customer. Otherwise, it is assigned 0.

In contrast, numeric attributes are attributes whose domain is a numeric value within the possible range of values for an attribute in question, e.g. age, income, weight, etc. This kind of attributes in a relation has a strong effect on the way of information extraction, as well as, on the type of patterns revealed by the mining process, e.g., performing regression tests on numeric attributes or on nominal ones.

Finding an interrelation between paired numeric attributes is a major research area in statistics. Statistical theory deals with hypothesis testing, rather than automatic hypothesis generation, and has developed various tests for many kinds of hypotheses. There are many hypothesis mechanisms to test the appropriate interrelation between numeric attributes, such as those involving covariance and line estimators. Several data exploration and analysis systems include, indeed, statistical data analysis packages.

Some of the weaknesses of these statistical tools are as follows:

1. They are capable of recognizing interrelations in the entire dataset, but cannot extract subsets of data in which a strong interrelation occurs.
2. The attributes to be explored must be specified. Hence, one cannot check automatically all hypotheses on all combinations of attributes and obtain the most important set of attributes for which a strong interrelation holds.

Some heuristics for extracting local interrelations use geometric clustering techniques, but they may not be capable of finding the appropriate semantics of an interrelation between the attributes. In addition, they are time consuming, especially when handling large datasets.

Data mining techniques based on hypothesis generation, rather then hypothesis verification, would contribute significantly to the ability of discovering truly unknown interesting information. On the one hand, it is difficult to formalize the users and their notion of interestingness. On the other hand, if a data mining tool can reveal a subset of data, among huge amount of possibilities, which the user may find interesting and useful, and can help the user to define hypotheses on this data subset, then it may not be necessary to have the hypotheses formally pre-specified.

1.1 Rules

There have been vigorous efforts in trying to quantify and formulate the concept of an interesting data set. This has led to the notion of *association rules* [1, 2] and their variants, e.g., generalized association rules [29, 15], correlation rules [4] and causal structures [28], ratio rules [21], quantitative association rules [30, 3], and optimized association rules [9, 10, 26].

Association rules define a specific type of an hypothesis and the goal of the proposed algorithms is to find instantiations for attributes to derive rules that make the hypothesis more specific and allow the user to confirm or reject them. In its most basic definition a rule is an expression of the form $X \Rightarrow Y$, where X, Y are (subsets of) attributes from the relation and $X \cap Y = \emptyset$. The rule is aimed at defining and revealing a specific type of hypothesis regarding the interrelation or correlation between these sets of attributes. The interestingness or usefulness of the rule is usually measured by some predefined metric function such as confidence and support [2], gain [9], chi-squared value [4], gini [22], entropy gain [23, 22], laplace [6, 32], lift [16], interest [5], strength [8], and conviction [5].

Several proposals for mining different types of rules according to different types of pre-specified interest metrics have been suggested in the literature. The suggested techniques are fully automatic but need to have predefined tasks. The ground work of formalizing the concept of a rule, namely *association rules*, on the categorical attributes of a relation, was introduced in [2].

Most real world databases contain some, if not primarily, numerical data. The real need for analyzing numeric data resulted in applying the framework of categorical association rules to the case where attributes are numeric. There are several suggested rule types that use different interest measures to generate rules and optimized rules with numeric attributes.

Srikant et al. [30] followed their basic definition of associations and defined quantitative association rules to include sets of categorical attributes and intervals of numerical attributes. The following is an example of such a rule: Age$\in$[24,30] $\wedge$ Sex=male $\Rightarrow$ Weight$\in$[72,90] (support 30%, confidence 80%). The interval representation of numerical attributes arose from the need to partition the numerical values into ranges so as to get discrete values, allowing one to apply any algorithm in the boolean case.

Another approach for mining quantitative association rules is introduced in [3]. Aumann and Lindell developed a different definition for quantitative association rules based on statistical distribution of the numerical values of a quantitative attribute. Instead of viewing the intervals as discretization of the numeric values of the attributes, they regarded the numeric attributes as continuous. Thus, the left-hand side of a rule in question describes a specific population (defined by a certain range of one numeric attribute), while the right-hand side of the rule characterizes an interesting behavior of another numeric attribute, when restricted to the population on the left-hand side. An example of a rule according to their definition is: Age$\in$[45,65] $\wedge$ Sex=male $\Rightarrow$ Weight: mean = 105kg (overall mean = 80). Their idea is to define an interesting behavior by comparing (e.g., the mean and variance) distributions of the numerical attributes using statistical tests for the mean and variance. This approach is sufficient for generating rules with only one numerical value on each side of the rule, and it is well founded within their definition. Their technique does not generalize, however, to higher dimensions.

The methods described above are part of a constraint-based approach which generates all rules that mitigate the user's predefined minimum support and minimum confidence thresholds. Such methods are likely to run into difficulties, as far as identifying interesting and useful attributes out of the vast amounts of reported rules. This operation is time consuming and does not scale to large data sets. In addition, it poses the need for additional tools for extracting the most interesting patterns.

To alleviate some of the problems that arise when generating all possible rules that satisfy a certain threshold, the class of optimization-based rule miner [10] was proposed. The purpose here is to identify only rules that are optimized according to some measure of interest. This approach is usually preferred when the constraint-based approach generates an excessive amount of rules and results in poor performance.

In [26] the framework of optimized association rules, described in [9, 10], was extended in several ways. However, the rules obtained suffer from two problems. They pre-specify the variable in the hypothesis, and their consequence is a categorical variable, rather than numeric.

1.2 Data Visualization

Visualization of data is a known concept. Many visualization techniques have been developed over the years. In addition, existing techniques have been extended to work for larger data sets and make the displays interactive. Visual data mining and information visualization techniques are aimed at supporting the exploration and analysis of very large amounts of data, so that users can browse large datasets and find patterns, correlations, clusters, gaps, and outliers that reveal opportunities for action.

The common perception is to treat relational databases as multidimensional data sets with the attributes of the database corresponding to the dimensions. Since a human is included in the visual data mining process, it is important to provide techniques that give a good overview of the data, so that the user can navigate the data effectively. Therefore, several techniques for viewing (or visualizing) effectively multidimensional (i.e., multivariate) data were suggested. These include: scatter-plot matrices and coplots, projection matrices, parallel coordinates, geometric projection techniques, icon-based techniques, hierarchical techniques, dynamic techniques, graph-based techniques, pixel-oriented techniques, and combinations thereof. A complete comprehensive survey of these techniques appears in [17, 18], and in [19] a detailed evaluation and comparison of several visual data mining techniques including pixel-oriented, geometric and icon-based techniques is made.

It is important to note that while visualization techniques have the advantage of suggesting hitherto unknown hypotheses, they are slow, imprecise, and confusing in the case of high dimensional data. For example, if a record consists of 30 fields, then just viewing all relationships between triples involves $24,360$ visualizations. This is clearly not feasible for a human user.

1.3 Desiderata

A truly effective tool for generating hypotheses in a data set should have the following characteristics:

1. Integrate categorical and numeric variables.
2. Generalize in the dimensions.

3. Be scalable to large data sets.

4. Generate hypotheses for various statistical tests.

We know of no method that can answer all of the above needs. In this paper we present a novel idea for generating quantitative hypotheses. Our methodology is based on data visualization. Specifically, it uses image processing techniques to automate the decision process. It decides which hypotheses are invalid, and ranks the valid ones in order of "interest". This allows statistical verification of the highest ranking hypotheses, thereby meeting all of the above needs.

2 Main Idea and Proposed Approach

Our method is not restricted in the number of dimensions. However, for ease of exposition and to better convey our idea we will henceforth consider three dimensions. In other words, we will try to find all triples of variables $\langle X, Y, Z\rangle$, where the X and Y variables have an influence over the Z variable. We restrict ourselves to three dimensions since this case best illustrates the intuition behind our idea. (In Section 5 we will point out how the method generalizes to higher dimensions.) At any rate, the three-dimensional illustration is already meaningful since, to our knowledge, there is no currently known algorithm that suggests such a relation.

It should be noted that the main goal of this paper is to suggest the method's viability. We have yet to speed up process. However, point to the directions that can be exploited for scalability. Also, we demonstrate that even the straight-forward implementation currently used gives reasonably fast results.

2.1 The Ideal Case

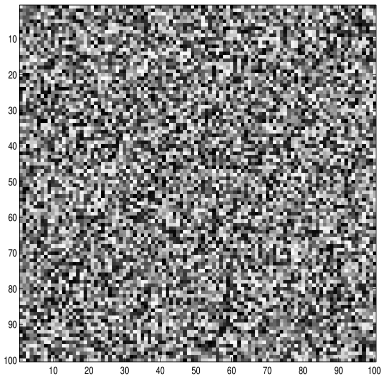
Assume that our data set is composed of records with numerical (quantitative) fields (variables). Fix variables X, Y, and Z. Assume that for every instantiation of X and Y there is a single instantiation of Z.

Example: Suppose X is social security number, Y is age, and Z is weight. For a given social security number 123-45-6789 and age 15, there is a single possible weight. On the other hand if X is weight, Y is age, and Z is social security number, there may be many possible values for a fixed pair $\langle X, Y\rangle$. For example, there may be many 40 year old people whose weight is 180 lbs.

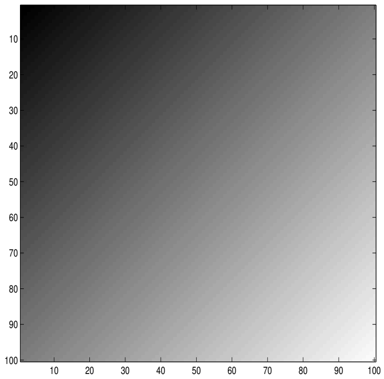
We will discuss in Subsection 2.2 how to handle the latter case. For the moment assume a single Z value per pair $\langle X, Y\rangle$. To provide a more complete description of our ideal environment, we further assume that there is a data set record for every pair $\langle X, Y\rangle$, $X = 1, ..., d_1$, $Y = 1, ..., d_2$. The uniqueness assumption above implies that we can construct a $d_1 \times d_2$ matrix whose value at location $[i, j]$ is the value of the Z field in the record whose X and Y fields are i and j, respectively.

Continue our wishful thinking, and assume even further that the values of variable Z range between 0 and 255. We may then view the value of variable Z in terms of a picture's gray level. In other words, our matrix above will now be a $d_1 \times d_2$ gray level image.

The intuition behind our idea is that, viewing the gray level image above, we can express an opinion on the relatedness of Z to X and Y. If the image is "noisy", as in Figure 1a, then it is unlikely that there is a relation between X, Y, and Z. However, if the image exhibits relative continuity in the level



a. A "noisy" image



b. A continuous image

Figure 1:

of grayness, as in Figure 1b, then it is likely that the values of X and Y are related to the value of Z. Our idea is to utilize image processing techniques to automatically assess the continuity of a given image. Thus, our system will point out to the investigator "interesting" subsets of variables from a given database to be further explored.

We use the *contrast* [12] measure for texture to rank the continuity of an image. In Subsection 3.3 we discuss in detail the formulation used and the motivation for selecting it.

Unfortunately, life is not a rose garden. The ideal situation described above is never encountered. We need to contend with the following pitfalls:

1. Extremely large ranges of X and Y, such that the image processing algorithms become unfeasible due to the huge image size.
2. "Holes" in the image, i.e., a large number of pairs $[i, j]$ for which there is no record in the data set with the values $X = i,\ Y = j$.
3. A large range of values for the Z variable.
4. Pairs $[i, j]$ for which the records with values $X = i,\ Y = j$ have varying different values in the Z variable.

2.2 Creating an Image from "Dirty" Data

2.2.1 Limiting Ranges of X and Y

Theoretically, we can use the raw elements of X and Y. However, the range of X and Y may be huge (e.g., salaries). Constructing and analyzing an image of size $200,000 \times 200,000$, for example, is not computationally feasible. In addition, very large ranges will mean extremely sparse matrices, which also contradicts our intuition of having as contiguous an image as possible. It seems practical to limit the image size to 500×500.

There are two commonly used ways of limiting large variable ranges. One is a linear mapping of subranges to the desired ranges, and the other is a mapping of equal sized subranges to the desired ranges (similar to equi-width).

Examples:

1. Let X range from $-10,000$ to $+10,000$ and assume that the desired range is from 1 to 500. The $20,000$ integer values of X's range are linearly mapped to 1..500 with each 40 values of X being mapped to a single value. For example, $-10,000, ..., -9,961$ are mapped to 1, $-9,960, ..., -9,921$ are mapped to 2, etc.

2. Again, let X range from $-10,000$ to $+10,000$ and assume that the desired range is from 1 to 500. However, suppose that the data set only has 1000 records. Suppose further that for these records the values of X range as follows: $-10,000, ..., -9,501$ and $9,951, ..., 10,000$. We will map every 2 elements of X to a single value, with $-10,000$ and $-9,999$ mapped to 1, $-9,998$ and $-9,997$ mapped to 2, and so on until $-9,502$ and $-9,501$ mapped to 250, $9,951$ and $9,952$ mapped to 251 etc., until $9,999$ and $10,000$ are mapped to 500.

We chose the first option because it is natural, when dealing with large numerical values, to consider subranges (e.g., salaries). Using equal sized buckets may distort the meaning of the data by lumping together elements that are far from each other and separating close elements. We experimented with different granularities of the division and realized that it made negligible difference in the final ranking of the triples (presented later).

2.2.2 Bridging Gaps in the Image

The texture formulas from image processing assume, of course, that image pixels are contiguous. This is not necessarily the case in our data set of generated gray level images. However, by ignoring missing data and using the closest pixel with existing data in the direction of the missing pixel, we managed to overcome this problem. The exact details of choosing neighborhoods for calculating contrast are described in Subsection 3.3.

2.2.3 Handling the Z Variable Range

A linear handling of the Z variable range may cause a granularity that is too rough.

Example: Consider the case where variable Z is the number of doctor's visits a person undergoes a year. This number may range from 0 to 700 (in case the person is in a hospital and is seen by a doctor twice a day). However, it is clear that the vast majority of people visit a doctor between 0 and 2 times a year. Thus, a linear mapping of the Z values to 256 gray levels would produce an almost uniform picture, which is not very interesting.

What is necessary is a way to *trim* the values of the Z variable in a manner that considers only the Z values of the majority of records. We present a new method of *histogram trimming* to achieve this end. Histogram trimming is described in detail in Subsection 3.1.

2.2.4 Multiple Z Value

Perhaps the greatest challenge in trying to construct a gray level image from data is determining the gray level of an element $[i, j]$, where there are many different Z values for records with $X = i$, $Y = j$.

If the values of Z for such an element range over a large set of values, then it is pointless to consider the continuity of Z over X, Y. Even a single element is not coherent.

Example: Assume we are considering the correlation between pulse, blood pressure, and weight. If we find in our data set 50 records of people whose pulse is 75 and whose blood pressure is 110/80, and if the weights of these individuals range from 70 lbs to 400 lbs, then it does not seem likely that there is any correlation between pulse, blood pressure, and weight.

We need to devise a measure for pixel "validity". In addition, it is necessary to be able to discount isolated invalid pixels, or concentrated areas of invalid pixels. We define the notion of a *valid pixel* as one where a large portion of its subpopulation is congregated within a small distance from the median of that subpopulation.

The notion is complicated by the fact that a "small distance" is not a uniform value throughout the data set. There may be subsets of the range of Z, where a large population is densely congregated, versus large sparse range subsets. (One's closest neighbor in the Australian Outback may be 60 miles away, whereas in NYC it is a foot away.) We use the histogram trimming to define the relative closeness to the median. The details of this idea are discussed in Subsection 3.1.

Having identified the valid pixels, we use texture formulas to automatically reject images with predominantly invalid pixels. For the remaining images, the Z value of a valid pixel is taken as the median of the values that the pixel takes on. These images are then ranked by the contrast measure.

3 Data Representation and Processing

3.1 Data Validity

3.1.1 Valid Pixel

The presentation of a data set in terms of an image as described in the previous section may often provide an image containing pixels with missing data. The gray level value assigned to such pixels is 255, i.e., pixels lacking data will appear in white. Also, we would like to exclude from further consideration pixels which contain data that do not seem significant. Such pixels were referred to as *invalid pixels* and they are assigned a gray level of 0, i.e., invalid pixels will appear in black. The rest of the image pixels will be mapped, eventually, onto some gray level value between 0 and 255 as described in Subsection 3.2. Let us treat, for now, all valid pixels as having the same gray level value.

We turn our focus to determining whether or not a pixel containing data is considered valid. As was stated in Subsection 2.2.4, a pixel will be considered valid, provided that a large portion of its subpopulation is congregated within a small distance from the median (of the Z-values) of that subpopulation. Let us elaborate, in more detail, on the various aspects contained in this definition. We choose to consider the median value (as opposed to the mean, for example), because the median is known to be a *robust*, i.e., it provides an estimate that is immune to noisy data and/or outliers. (See, e.g., [27] on the concept of robust estimation.) In the spirit of Rousseeuw's (1-D) *least median of squares* (LMS) estimator (ibid. [27]) — this is the midpoint of the narrowest interval containing (at least) 50% of the data — we will look at a small interval surrounding the median and check whether (at least) 50% of the pixel's data reside in this interval. If so, then the pixel is valid (in the sense that the median was, indeed, representative of the majority of the pixel's data). Otherwise, it is invalid.

As was pointed out in Subsection 2.2.4, we make use of the trimmed histogram to determine relative closeness to the median. We considered some possibilities of specialized histograms (e.g. [25, 24]) but they did not answer all our needs. Assume, for example, that the range of the trimmed histogram (with respect to the Z-values) is r. Then, we could define the surrounding interval as $p \cdot r$, where p is some small fraction (e.g., 0.1). While this may be applicable to, e.g., unimodal histograms, where relatively little information is contained in the tails and trimming is bound to yield a (much) smaller range in the Z-value, in general it is likely to encounter difficulties. For example, consider the case where the distribution of the Z-value is bimodal. Clearly, trimming the "usual" way is likely to have no effect on the range. Thus, the resulting surrounding interval, $p \cdot r$, will be too large, and the pixel in question will be determined valid even though is contains data over a (relatively) large range and should have been considered invalid.

To remedy this phenomenon, we propose a *generalized histogram trimming* procedure where instead of trimming, say, the bottom and top 5% of the Z-values, we trim the least frequent 10% of these values. Suppose that such trimming results in t Z-levels that constitute the remaining 90% of the data. Instead of defining the surrounding interval (around the median of a pixel's subpopulation) as before, we locate the median with respect to the above set of t Z-values and find the narrowest interval around the median that contains, say, 10% of the t values assumed by the trimmed histogram. Having picked an interval in the above described manner, we proceed to determine whether (at least) 50% of the pixel's data reside within the interval, in which case the pixel is valid.

3.1.2 Valid Field Subset

Having designated pixels as invalid does not necessarily mean that the image is not interesting, or that the hypothesis should not be statistically verified. It is possible that there is a very small number of invalid pixels. Alternately, even if there is a larger number of invalid pixel, if they are all congregated together in contiguous areas, it may still be the case that the rest of the image actually has a continuous Z surface.

In Subsection 3.3 we introduce the *texture* measure, adopted for our purposes. The main use we make of it is for ranking the valid images relative to their variance. Incidentally, the same measure was used initially to rank images for validity. We create a tri-color *co-occurrence matrix* (to be defined later), consisting of valid (gray), invalid (black), and non-existing-value (white) pixels. The variance measure described in Subsection 3.3 was used to identify tri-color images having a large number of invalid pixels. Such images were immediately rejected.

An additional validity test is low entropy of the gray level image. This suggests an uninteresting image and causes its rejection. Examples are shown in Subsection 4.

3.2 Mapping Z-Values to Gray Levels

In this subsection we discuss how to map Z-values onto image gray levels. Suppose that the original range of Z-values is smaller than the number of gray levels. (We will assume 256 gray levels as is customary in digital image processing.) Let the range be defined by $z_{\min}$ and $z_{\max}$, and let $\mathrm{glv}_{\min}$, $\mathrm{glv}_{\max}$ denote, respectively, the smallest and largest gray level value. (In general, $\mathrm{glv}_{\min} > 0$ and $\mathrm{glv}_{\max} < 255$, as black and white are reserved for pixels missing data and invalid pixels, respectively.)

A value z ($z_{\min} \leq z \leq z_{\max}$) is mapped to gray level glv, such that,

$$\mathrm{glv} = (\frac{z - z_{\min}}{z_{\max} - z_{\min}}) \cdot (\mathrm{glv}_{\max} - \mathrm{glv}_{\min}) \ + \ \mathrm{glv}_{\min}.$$

In general, however, the range might be very large and we may need to apply generalized histogram trimming, as explained in Subsection 3.1.1. Let t denote the number of Z-values that constitute, say, 90% of the generalized trimmed histogram. Also, let b denote the remaining number of Z values, and let nc denote the number of colors (i.e., gray level values) that are allocated for the above t values. (Typically, if we trim the least frequent 10% of the data, then nc is set to 230, i.e., roughly 90% of the gray level range.) We distinguish between the following two cases:

1. $t \geq \mathrm{nc}$. In this case, we simply assign t/nc Z-values to each color, where smaller values are assigned to smaller gray level values. The distance between gray levels assigned to the t values is computed relative to the b values, i.e., we assign $b/Z - \mathrm{nc}$ values per pixel.
2. $t < \mathrm{nc}$. In this case every one of the t values is assigned a distinct gray level. The distance between gray levels assigned to the t values is computed relative to the b values, precisely as before. Namely, we assign $b/Z - \mathrm{nc}$ values per pixel.

3.3 Data Ranking

At this stage we have a gray level image of valid pixels. The next step is to quantitatively assess the degree to which the image is continuous. The tool we use for this is *texture analysis*.

Texture characteristics have proven very important and have been used widely in the analysis/processing of many types of images. In a nutshell, texture can be characterized by the spatial distribution of gray levels in a neighborhood of a given image. Following the definitions in [14], texture can be defined as having one or more of the properties of uniformity, fineness, smoothness, density, coarseness, roughness, regularity, intensity, and directionality of the gray level primitives and the spatial relationships between them within an image. Numerous definitions for texture, designed for particular applications, have been proposed in the literature.

Structural approaches to texture analysis use the idea that texture is composed of primitives with different properties appearing in particular spatial arrangements. On the other hand, statistical approaches try to model texture using statistical distributions. Haralick combined these two approaches by defining texture as being specified by the statistical distribution of the properties of the different textural primitives occurring at different spatial relationships (see [12]).

A pixel, with its gray level as its property, comprises the simplest primitive of a digital image. Consequently, the gray level distribution of a pixel can be described in terms of first-order statistics, such as the mean, standard variation, or validity as defined in Subsection 3.1. The gray level distribution can also be described by second-order statistics, such as the probability of two pixels having particular gray levels occurring at particular spatial relationships. This information can be captured in a two-dimensional *co-occurrence matrix*, computed for different distances and orientations. The two-dimensional continuity we seek relies on the above kind of information and can be derived from the corresponding co-occurrence matrix.

Gray level co-occurrence can be specified as a two-dimensional matrix of relative frequencies, $P_{(d,\theta)}[i,j]$, where each entry is equal to the frequency with which two pixels separated by distance d at orientation θ occur in the image, one with gray level i and the other with gray level j.

Example: Consider the following 4×4 gray level image, with gray levels ranging over $\{0, 1, 2, 3, 4\}$

2	3	4	4
1	2	2	4
0	1	2	3
0	0	1	2

Its co-occurrence matrix $P_{(1,0^\circ)}$ is:

$$P_{(1,0^\circ)} = \begin{pmatrix} 2 & 2 & 0 & 0 & 0 \\ 2 & 0 & 3 & 0 & 0 \\ 0 & 3 & 2 & 2 & 1 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 1 & 1 & 2 \end{pmatrix}$$

.

(Note that by definition, a co-occurrence matrix is symmetric.)

For a full local view, one needs the relationship between all neighbors, not just a single fixed angle. Thus, it is customary to consider the following as the co-occurrence matrix:

$$P = \sum_{i=0}^{7} P_{(1,i\pi/4)}.$$

In order to extract useful information from a co-occurrence matrix, Haralick et al. [13] defined 14 statistical measures that measure textural properties like homogeneity, contrast, organized structure, and nature of gray levels transition, as mentioned above. The specific distance and orientation depend on the nature of the examined scene.

In our context, where each pixel represents a lot of information, we need to extract micro-texture features that use second-order statistics (i.e., co-occurrence) of gray levels of pixels in appropriate spatial relationships. We have compared several relevant measures for our purpose and found the variance to be the most adequate and satisfactory with our results. This bodes well with Gotlieb and Kreyszig [11] who claimed that in heuristic tests they had carried out, overall the variance (or contrast) performed best in finding similar textures.

Definition: Given a co-occurrence matrix P, the variance of gray level spatial dependencies, which is the difference moment of P is given by

$$\sum_{i,j} |i - j|^2 p_{ij},$$

where p_{ij} is the probability that gray level i appears next to gray level j (horizontally, vertically or diagonally). In other words,

$$p_{ij} = \frac{P[i,j]}{\sum_{k,l} P[k,l]}.$$

Therefore, the above variance measures contrast in the image. This parameter will have a large value for images that have a large amount of local spatial variation in gray levels and a smaller value for an image with spatially uniform gray level distributions. This observation also explains the reason that the variance is a good measure for the continuity of a gray level image. The smaller the variance, the more continuous the image.

3.3.1 The Number of Colors

There is an additional factor that should be taken into account — the number of "colors" (i.e., gray levels) in the image. Since we seek "interesting" hypotheses, it is not sufficient for the variance to be small. In fact, the variance is 0 when the entire image is composed of a single color, but that is hardly an exciting case. Therefore, we modified the variance measure to be biased toward a larger amount of different colors. In particular, we used the expression below (where c denotes the number of different colors in the picture):

$$\frac{1}{c^2}\sum_{i,j}|i-j|^2 p_{ij}.$$

3.3.2 Disconnected Image

It was mentioned in the overview (Subsection 2.2) that our gray level image is unlikely to be contiguous. Thus, the co-occurrence matrix P may have too few entries. To handle this effect it is possible to fill the image by interpolating into it the missing values. Such interpolation would increase the complexity, and possibly "smooth over" areas that are really not smooth.

We chose, therefore, to divide the neighborhood surrounding each pixel into eight zones (see Figure 2). As far as the co-occurrence computation — our implementation finds, for every pixel, its nearest neighbor pixel in every zone. In the case of a contiguous image, this reduces exactly to the variance.

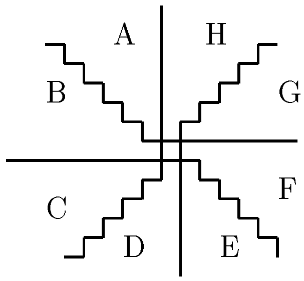


Figure 2: The zones surrounding a pixel

4 Experimental Results

We tested our algorithm on two Census datasets. The experiments were performed on UltraSparc-II 248MHZ running Solaris 2.6, and all algorithms were written in C. In these data sets it is often likely to find known correlations between attributes.

Assuming that the data are sorted on each triple, building the images and performing the ranking by our image processing ideas were highly efficient and took only a few seconds. The output is a set of triple attributes, $\langle X, Y, Z\rangle$ for each data set, that were found to be the most interrelated by examining the continuity of its spatial arrangements of Z-values.

4.1 First Census Data

The first data set, cpsm93p (Current Population Survey of 1993 of personal records), has 3 types of file groups of the March Questionnaire Supplement. We selected the Person Data Files group. The data set is available from the Data Extraction System (DES) on `http://www.census.gov/`. It consists of 413 attributes, both numeric and nominal. Out of these, we selected 6 numeric attributes that we considered to be interesting. These attributes are described below.

Code	Description	Range
AGE	Person's age	0..90
AGI	Adjusted gross income	$-9999..99,999$
CAPGAIN	Amount of capital gain	$0..99,999$
DIVVAL	Amount of dividends from stocks	$0..99,999$
UCVAL	Amount of unemployment benefits	$0..99,999$
WSALVAL	Total wage and salary	$0..199,998$

We took only non-empty personal records which had actual values for our selected attributes. The total number of records was 155,198.

¿From these six attributes there is a total of 60 relevant triples $\langle X, Y, Z \rangle$. For each triple $\langle X, Y, Z \rangle$ we partitioned the X and Y attributes to subranges according to their value ranges. For example, AGE ranges between 0 and 90, so we handled it as is without any partition. On the other hand, WSALVAL ranges between 0 and 199,998, so we it into 500 subintervals by considering each 400 dollars as one subinterval. This has led to a 90×500 image. It is worth noting that practically it is hardly ever necessary to consider images larger than 1000×1000 (500×500 was sufficient in our tests), since a full 1000×1000 image already implies 10^6 records in the database.

We generated the appropriate 60 images and ran our tests. The first step after creating the images was to determine which of them contained valid information and drop those which did not. Recall that for each image pixel we compute whether or not it is valid. Our algorithm first checks the tri-color images and drops all images with high co-occurrence of black-gray pixels, or those having low entropy. Out of 60 possible images 51 were rejected because of the above reasons. The 9 images that were accepted were ranked according to their continuity (i.e., in increasing order of their variance).

In all the images shown, the X attribute is the horizontal one, with values growing from left to right. Y is the vertical axis with values growing from top to bottom. Z is depicted by its gray level, with light being smaller than dark (the reverse of the customary gray level assignment). The black dots indicate invalid pixels. White areas indicate that no record exists for the X and Y values covering these areas. Recall, gray pixels are considered valid at this stage.

Figure 3 was rejected for being invalid. In this image there is a preponderance of black (invalid) pixels, thus it was rejected. The meaning of this rejection is that the adjusted gross income and the total wages and salaries have no relation with a person's age. It is interesting to note that the reason there are no values below the diagonal is that it is practically impossible for the adjusted gross income to be greater than the total wages and salaries.

Figure 4 has a relatively small number of invalid pixels but it was rejected due to low entropy of the gray level image. The vast majority of valid pixels in this image have the same value of 0 (no unemployment was being paid). Thus even though the variance is low, the image was rejected for being unenlightening. It is interesting to note the triangular shape of the data, with the base up. The reason is that it is

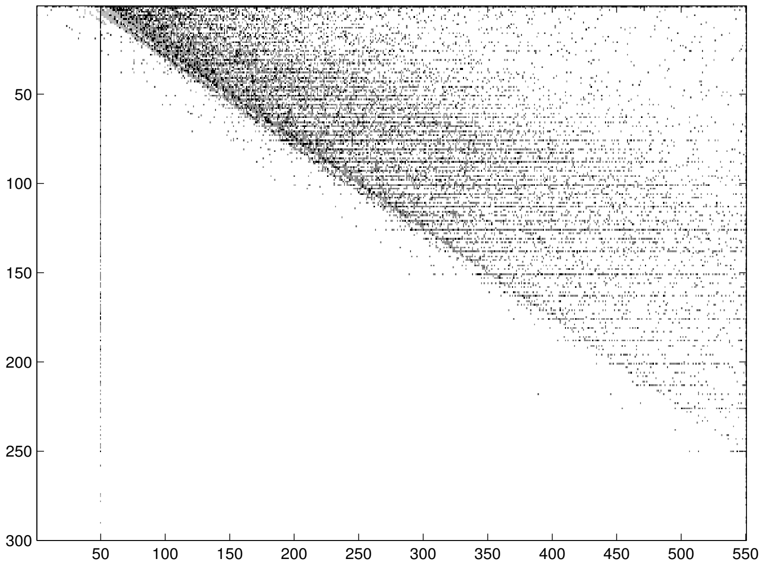


Figure 3: X = total wages and salaries, Y = adjusted gross income, Z = age

unlikely for both the very young and very old to have a high income.

Among the 9 valid images of the data set investigated, Figure 5 represents the image with the highest ranking. (Figure 6 is merely a 3-D plot of the highest ranking image, and conveys, essentially, the same information.) The rising values of Z as a function of X and Y are clearly evidenced. The story told by this data is that as the adjusted gross income rises, so does the amount of capital gain. However, this is most marked around middle age. Youngsters and old people have less income. Another interesting feature is the vertical line appearing most notably at the low end of the capital gain range. It means that as people get close to retirement age (around age 57), they are more likely to have capital gain, even if they are on a lower income scale.

4.2 Second Census Data

The second dataset we tested was the census dataset, nhis93ac (National Health Interview Survey 1993 (NHIS)). The data set is available from `http://ferret.bls.census.gov/` by using the FERRET data system. It has several hundred numeric and nominal attributes and consists of $45,951$ records. We chose 7 numeric attributes for our tests. These attributes are described below.

Code	Description	Range
AGE	Person's age	0..99
BDDAY12	Bed Days in Past 12 Month	0..365
DV12	Doctor Visits in Past 12 Month	0..997
EDUC	Education of Individual – Completed Years	0..18
INCFAMR	Family Income	0..8 (levels)
NCOND	Number of Conditions	0..14
WEIGHT	Weight Without Shoes	(-1 for children under 18) 50..500

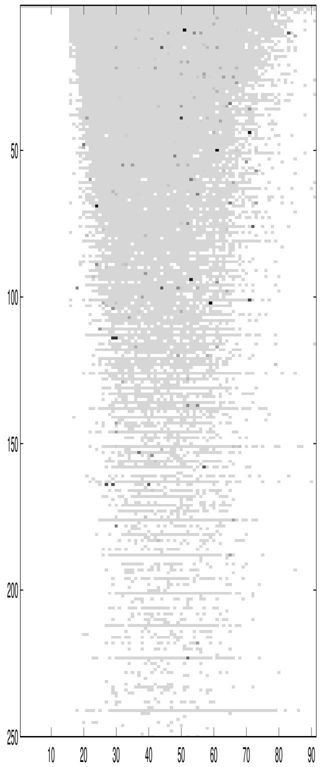


Figure 4: X = age, Y = total wages and salaries, Z = unemployment benefits

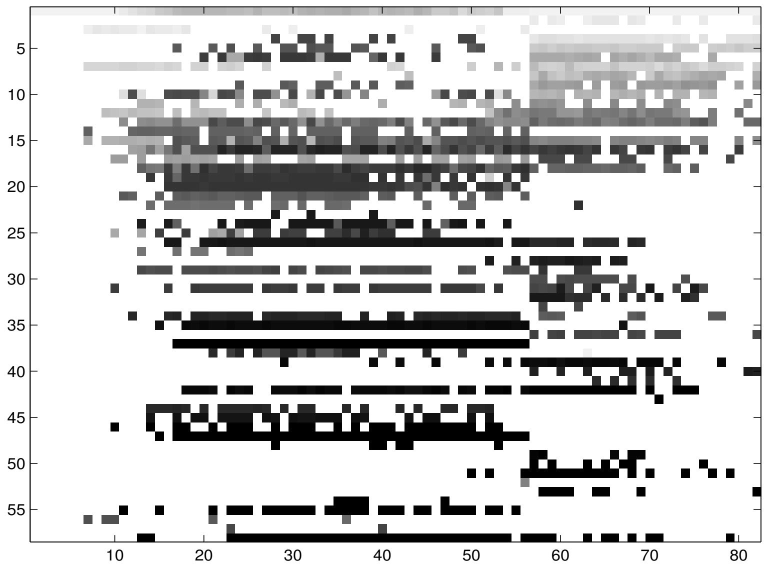


Figure 5: X = age, Y = capital gains, Z = adjusted gross income

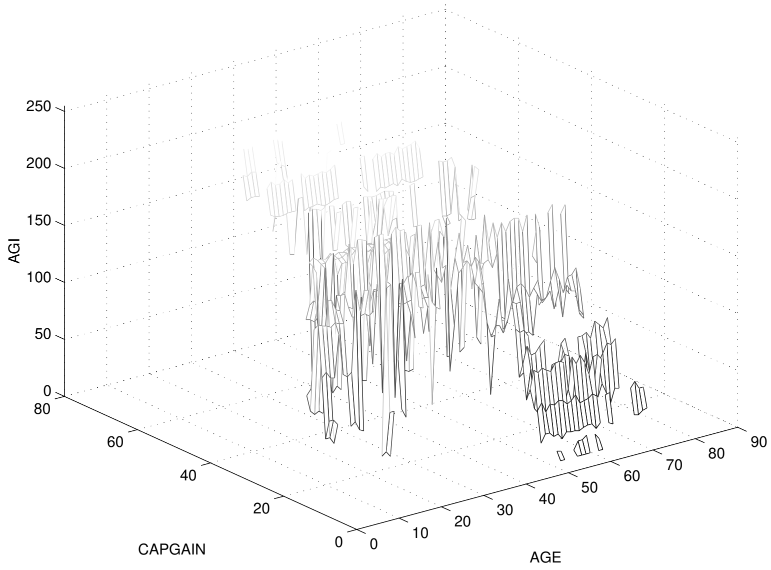


Figure 6: X = age, Y = capital gains, Z = adjusted gross income

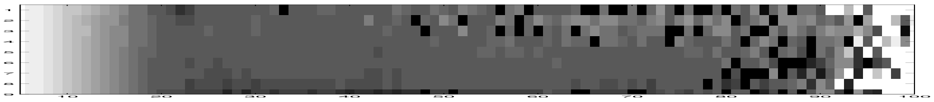


Figure 7: X = age, Y = income, Z = education

For these 7 attributes there are 105 possible triples and thus our algorithm analyzed 105 images. The validity check rejected 86 of the 105 images and accepted 19.

The highest ranking image of this data set makes the case quite strongly that higher income implies higher education. However, there are side tales to be learned from this picture. The darkening of the left-hand side of the image in a uniform manner (or the clearly identified slope in the three-dimensional rendition) means that children until the age of 15 will have virtually the same number of school years, regardless of their family's income. The high number of black pixels on the right means that where very old people are concerned, their number of school years is not well distributed. We also notice that old people have, in general, less education then their middle aged counterparts within the same income class.

5 Discussion and Future Research

Our discussion hitherto relied entirely on mapping the data to an image, thus generating a hypothesis on three variables. This was done primarily for pedagogical reasons. The experimental results provided also dealt with three-dimensional hypotheses, because it is easy to visualize these results.

It is clear, however, that the method proposed is easily generalizable in the number of dimensions. The validity of a pixel can be calculated in the manner described above for any dimension. The co-occurrence matrix will become simply a co-occurrence hypercube in the appropriate dimension and the entire discussion follows through. The dimension will play, though, a role, as far as space complexity is

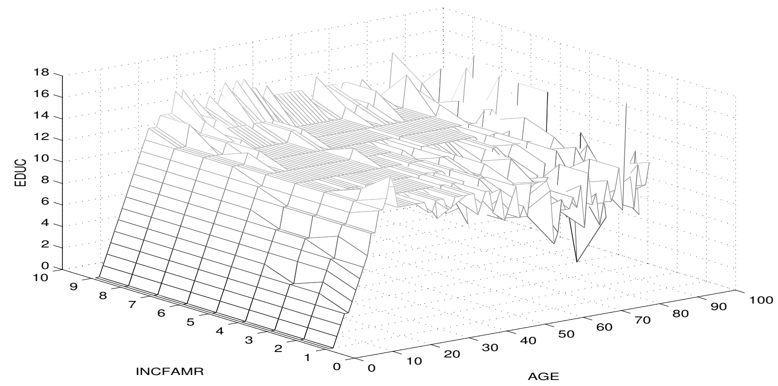


Figure 8: X = age, Y = income, Z = education

concerned. Although, as was previously stated, the main objective of the paper was to introduce the viability of our novel methodology, rather than optimize computational efficiency, we will devote the next few paragraphs to complexity-related issues.

The first issue is the tradeoff between dimensionality and granularity. While theoretically our method is easily generalizable to any dimension, the size of an "image" is multiplied by the number of buckets for every added dimension. We have used 256 for two dimensions, but this number is clearly not feasible for four dimensions and above. This can be compensated by taking smaller histograms.

Note that our treatment of the X and Y values differs from our treatment of the Z value. We computed X and Y linearly, while we used the trimmed histogram idea for the Z value. We are currently considering a uniform treatment of all variables according to the trimmed histogram. In our tests so far, this has no effect on the ranking of our attributes. However, applying this uniform treatment makes the entire process much more efficient, since the complexity now depends solely on the dimension of the relation and on the record size, but *not* on the database size. Thus, the real cost of our algorithm depends on dimensionality and granularity. Once those parameters are fixed, the size of the dataset does not matter (meaning that scalability is not an issue) since the algorithm scans the database exactly once.

Assume that every record has k attributes (fields). Then the number of triples $\langle X, Y, Z\rangle$ the algorithm considers is $\binom{k}{2}$. The number of quadruples would be $\binom{k}{3}$, etc. The total number of relations of all sizes is 2^k. On the other hand, if we construct trimmed histograms of size d (d buckets), then the space requirement for i-dimensional relations is d^i. Thus, our space requirements for computing *simultaneously* all relations of dimension no greater than i is

$$\sum_{\ell=1}^{i} \binom{k}{\ell} d^{\ell}.$$

Hence, if the above value fits in memory, we can compute all relevant relations in time proportional to a *single* scan.

Example: Suppose that every database record has $k = 50$ attributes. For $i = 4$, there are $1,225$ two-dimensional images, $19,600$ three-dimensional images, and $230,300$ four-dimensional images. If we pick $d = 30$, then two-dimensional, three-dimensional and four-dimensional images of interest will be of sizes 900, $27,000$, and $810,000$, pixels, respectively. Thus, the total requirement is around 1GB. This means that given a platform of 1GB memory, we can compute all relations for a database of *any* size in core doing a single database scan. In contrast, it is not feasible to use any statistical method (e.g. [7, 31] for brute-force testing of such magnitude. We provide a method that ranks the hypotheses to be tested. There is currently no known method that ranks relations of dimensions higher than 2.

Finally, it would seem like our algorithm will only detect hypotheses that are uniformly valid. This is not entirely true. If the image is composed of a number of segments, each of whose variance is small, the overall picture will still have a small variance. However, there is one type of case that would be of interest and yet might "slip through the cracks". This is the case of an image containing regions with small variance and other regions that are extremely noisy. The noisy regions will cause the variance to be high, yet we would like to know the range of values where the hypothesis holds. As it stands now, our algorithm will not rank this image high. However, we have been adding a *segmentation* component to our algorithm. The latter should help distinguish an image with large regions having a low variance.

Acknowledgments
The authors are grateful to Leo Mark for his help and suggestions.

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Database mining: a performance perspective. *IEEE Trans. Knowledge and Data Engineering*, 5(6):914–925, 1993.

[2] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proc. of the ACM SIGMOD International Conference on Management of Data*, pages 207–216, Washington, DC, May 1993.

[3] Y. Aumann and Y. Lindell. A statistical theory for quantitative association rules. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, San Diego, CA, USA, August 15-18 1999.

[4] S. Brin, R. Motwani, and C. Silverstein. Beyond market baskets: Generalizing association rules to correlations. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 265–276, Tuscon, AZ, May13–15 1997.

[5] S. Brin, R. Motwani J. D. Ullman, and S. Tsur. Dynamic itemset counting and implication rules for market basket data. In *Proc. of the 1997 ACM SIGMOD Int'l Conf. on Management of Data*, pages 255–264, 1997.

[6] P. Clark and P. Boswell. Rule induction with cn2: Some recent improvments. In *Machine Learning: Proc of the fifth European Conference*, pages 151–163, 1991.

[7] N. A. Cressie. *Statistics for Spatial Data.* John Wiley and Sons, rev. edition edition, 1993.

[8] V. Dhar and A. Tuzhilin. Abstract-driven patterns discovery in databases. *IEEE Transactions on Knowledge and Data Engineering*, 5(6), 1993.

[9] T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama. Data mining using two-dimensional optimized association rules: Scheme, algorithms and visualization. In *Proc. of the ACM SIGMOD Int'l Conference on Management of Data*, pages 13–23, June 1996.

[10] T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama. Mining optimized association rules for numeric attributes. In *Proc. of the 15th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 182–191, June 1996.

[11] C. C. Gotlib and H. E. Kreyszig. Texture descriptors based on coocurrence matrices. *Computer Vision, Graphics and Image Processing 51*, 51(1):70–86, 1990.

[12] R. M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, May 1979.

[13] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-3(6):610–621, November 1973.

[14] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*, volume 2. Addison-Wesley, 1992.

[15] J. Hipp, A. Myka, R. Wirth, and U. Güntzer. A new algorithm for faster mining of generalized association rules. In *Proc. of the 2nd European Symposium on PKDD*, Nantes, France, September 1998.

[16] R. J. Bayardo Jr. and R. Agrawal. Mining the most interesting rules. In *Proc. of the fifth ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pages 145–154, 1999.

[17] D. A. Keim. Visual data mining. *Tutorial, Int. Conference on Very Large Databases (VLDB'97)*, 1997.

[18] D. A. Keim. Visual techniques for exploring databases. *Invited Tutorial, Int. Conference on Knowledge Discovery in Databases (KDD'97)*, 1997.

[19] D. A. Keim and H-. P. Kriegel. Visualization techniques for mining large databases: A comparison. *Transactions on Knowledge and Data Engineering*, 8(6):923–938, Dec. 1996.

[20] D. A. Keim, H-. P. Kriegel, and T. Seidl. Supporting data mining of large databases by visual feedback queries. In *Proc. 10th Data Engineering*, pages 302–303, Housten, TX, 1994.

[21] Flip Korn, Alexandros Labrinidis, Yannis Kotidis, and Christos Faloutsos. "Ratio Rules: A New Paradigm for Fast, Quantifiable Data Mining". In *Proc. of the 24th International Conference on Very Large Data Bases (VLDB)*, pages 582–593, New York, USA, August 1998.

[22] Y. Morimoto, T. Fukuda, H. Matsuzawa, T. Tokuyama, and K. Yoda. Algorithms for mining association rules for binary segmentations of huge categorical databases. In *Proc. of the 24th VLDB conf.*, pages 380–391, 1998.

[23] S. Morishita. On classification and regression. In *Proc. of the First Int'l Conf. on Discovery Science - Lecture Notes in Artificial Intelligence*, pages 40–57, 1998.

[24] V. Poosala. *Histogram-based Estimation Techniques in Databases.* PhD thesis, Univ. of Wisconsin, Madison, 1997.

[25] V. Poosala, Y. Ioannidis, P. Haas, and E. Shekita. Improved histograms for selectivity estimation of range predicates. In *Proc. of ACM SIGMOD Conf.*, pages 294–305, June 1996.

[26] R. Rastogi and K. Shim. Mining optimized association rules with categorical and numeric attributes. In *Proc. of the 14th Int'l Conf. on Data Engineering*, pages 503–512, 1998.

[27] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection.* Wiley Series in Probability and Mathematical Statistics. John Wiley and Sons, 1987.

[28] C. Silverstein, S. Brin, R. Motwani, and J. D. Ullman. Scalable techniques for mining causal structures. In *Proc. of 1998 ACM SIGMOD Int'l Conf. on Management of Data*, Seattle, Washington, USA, June 1998.

[29] R. Srikant and R. Agrawal. Mining generalized association rules. In *Proc. 21st Int'l Conf. on VLDB*, Zurich, Switzerland, Sept 1995.

[30] R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. In *Proceedings of the ACM SIGMOD Conference on Management of Data*, Montreal, Canada, June 1996.

[31] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S-Plus.* Springer, second edition, 1997.

[32] G. I. Webb. Opus: An efficient admissible algorithm for unordered search. *Journal of Artificial Intelligence Research*, pages 431–465, 1995.