Detecting Change in Categorical Data: Mining Contrast Sets

Stephen D. Bay and Michael J. Pazzani
Department of Information and Computer Science
University of California, Irvine
Irvine, CA 92697, USA
{sbay,pazzani}@ics.uci.edu

Abstract

A fundamental task in data analysis is understanding the differences between several contrasting groups. These groups can represent different classes of objects, such as male or female students, or the same group over time, e.g. freshman students in 1993 versus 1998. We present the problem of mining contrast-sets: conjunctions of attributes and values that differ meaningfully in their distribution across groups. We provide an algorithm for mining contrast-sets as well as several pruning rules to reduce the computational complexity. Once the deviations are found, we post-process the results to present a subset that are surprising to the user given what we have already shown. We explicitly control the probability of Type I error (false positives) and guarantee a maximum error rate for the entire analysis by using Bonferroni corrections.

1 Introduction

A common question in exploratory research is: "How do several contrasting groups differ?" Learning about group differences is a central problem in many domains. For example, the US Census Bureau prepares many statistical briefs that compare groups such as the publication, "The Earnings Ladder: Who's at the Bottom? Who's at the Top?" which contrasts high and low income earners over the years 1979 to 1992. They report such facts as: "About 4 in 10 year-round, full-time workers aged 18 to 24 had low earnings in 1992, up 19 percentage points since 1979."

Our goal is to detect differences like these between contrasting groups automatically from data. We seek conjunctions of attributes and values that have different levels of support in different groups. For example, if groups are based on education, we might find that P(occupation=sales | PhD) = 2.7%, while P(occupation=sales | Bachelor) = 15.8%.

Association rule programs [1] learn relations between variables within a dataset, so one might try to find differences by augmenting the data with an additional group variable and let an association rule learner run on this representation. This will not, however, return group differences, and the results will be difficult to interpret. For example, we ran an association rule program[1] on census data and obtained the results in Figure 1 (1% min-support, 80% confidence).

Examining these rules, it is extremely difficult to tell what is different between the two groups. First, there are too many rules to compare. Second, the results are difficult to interpret because the rule learner does not enforce *consistent contrast* [6] (i.e., using the same attributes to separate the groups). Clearly there are at least $26796 - 1674 = 25122$ rules that have no match. Finally, even with matched rules, we still need a proper statistical comparison to see if differences in support and confidence are significant.

Most association rule programs find all large itemsets, so another approach is to mine the large itemsets for each group separately and then compare them. This is not a good approach for three reasons. First, there is no information on itemsets that are "small" and do not make the minimum support cutoffs. If the itemset is not large in all groups, then we will need to make an additional pass over the database to count the small itemsets so we can statistically compare the results. Second, we lose opportunities to prune when the sets are mined separately, and we show in Section 4 that these pruning opportunities can improve efficiency. Finally, it is difficult to present intermediate results as the mining progresses in an anytime or interactive manner.

We will begin by defining the problem of detecting group differences, and then we will present and evaluate a mining algorithm for finding these differences.

2 Problem Definition

In association rules, we typically deal with market basket data where the database $\mathcal{D}$ is a set of transactions with each transaction $T \subseteq I = \{i_1, i_2, \ldots, i_m\}$. Each member of I is a literal called an *item*, and any set of these literals is called an *itemset*.

In this paper we generalize the data model to grouped categorical data. The data is a set of k-dimensional vectors where each component can take on a finite number of discrete values. The vectors are organized

[1] We used C. Borgelt's implementation of Apriori version 2.1.

Bachelors ← (93.1, 93.1)
CapitalLoss=0 ← Bachelors (86.9, 93.4)
Bachelors ← CapitalLoss=0 (86.9, 93.4)
United-States ← Bachelors (83.4, 89.5)
Bachelors ← United-States (83.4, 93.8)
CapitalGain=0 ← Bachelors (82.3, 88.4)
Bachelors ← CapitalGain=0 (82.3, 93.7)
White ← Bachelors (81.6, 87.7)
Bachelors ← White (81.6, 93.0)
Bachelors ← Male (64.4, 92.0)

(a) First 10 of 26796 Association Rules for Bachelor holders

CapitalLoss=0 ← PhD (6.1, 88.6)
United-States ← PhD (5.5, 80.5)
CapitalGain=0 ← PhD (5.5, 80.3)
White ← PhD (6.1, 88.6)
Male ← PhD (5.6, 81.0)
CapitalLoss=0 ← United-States PhD (4.9, 87.7)
CapitalLoss=0 ← CapitalGain=0 PhD (4.7, 85.7)
White ← CapitalLoss=0 PhD (5.4, 88.2)
CapitalLoss=0 ← White PhD (5.4, 88.2)
Male ← CapitalLoss=0 PhD (5.0, 81.7)

(b) First 10 of 1674 Association Rules for Ph.D. holders

Figure 1: Association rules for Bachelor and Ph.D. degree holders. Rules are in the form $Y \leftarrow X$ (support,confidence).

into n mutually exclusive groups. The concept of an itemset can be extended to a contrast-set as follows:

Definition 1. Let $A_1, A_2, \ldots, A_k$ be a set of k variables called attributes. Each A_i can take on values from the set $\{V_{i1}, V_{i2}, \ldots, V_{im}\}$. Then a **contrast-set** is a conjunction of attribute-value pairs defined on groups $G_1, G_2, \ldots, G_n$.

Example. $(\mathsf{sex} = \mathsf{male}) \wedge (\mathsf{occupation} = \mathsf{manager})$.

We define the support of a contrast-set with respect to a group G as follows:

Definition 2. The **support** of a contrast-set with respect to a group G is the percentage of examples in G where the contrast-set is true.

Our goal is to find all contrast-sets whose support differs meaningfully across groups. Formally, we want to find those contrast-sets (cset) where:

$$\exists ij \;\; P(\mathsf{cset} = \mathsf{True} \mid G_i) \neq P(\mathsf{cset} = \mathsf{True} \mid G_j) \quad (1)$$

$$\max_{ij} |\mathsf{support}(\mathsf{cset}, G_i) - \mathsf{support}(\mathsf{cset}, G_j)| \geq \mathsf{mindev} \quad (2)$$

and mindev is a user defined threshold. We call contrast-sets where Equation 1 is statistically valid *significant*, and contrast-sets where Equation 2 is met *large*. If both requirements are met, then we call it a *deviation*.

3 STUCCO: A Mining Algorithm

We treat the problem of mining contrast-sets as a tree search problem. The root node is an empty contrast-set, and we generate children of a node by specializing the set by adding one more term. We use a canonical ordering of attributes to avoid visiting the same node twice [9]. Children are formed by appending terms that follow all existing terms in a given ordering.

For example, consider an artificial domain with two attributes, $A_1 = \{V_{11}, V_{12}\}$ and $A_2 = \{V_{21}, V_{22}\}$, each with two possible values. Figure 2 shows the resulting search tree and enumerates every possible subset of values for A_1 and A_2. Nodes 3 and 4 have no children because A_2 comes after A_1 in our ordering.

We search this tree in a breadth-first, levelwise manner. Given all nodes at a level, we scan the database and count their support for each group and then examine each node to determine if it is significant

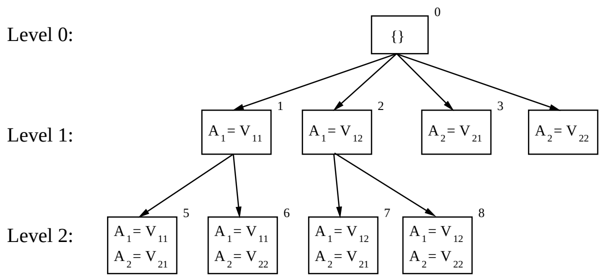


Figure 2: Example of the search tree for two Attributes $A_1 = \{V_{11}, V_{12}\}$ and $A_2 = \{V_{21}, V_{22}\}$.

and large, if it should be pruned, and if children should be generated. Figure 3 outlines the STUCCO (Search and Testing for Understandable Consistent Contrasts) algorithm. Section 3.1 explains the significance testing used. Section 3.2 describes pruning.

After finding all significant contrast-sets in the data, we then process the results and select a subset to show to the user. We display the low order results first, which are simpler, and then show only the higher order results that are surprising and significantly different. This is described in Section 3.3.

```
Algorithm STUCCO
Input: data D
Output: D_surprising
Begin
Set of Candidates C ← {}
Set of Deviations D ← {}
Set of Pruned Candidates P ← {}
Let prune(c) return true if c should be pruned
while C is not empty
    scan data and count support ∀c ∈ C
    for each c ∈ C
        if significant(c) ∧ large(c) then D ← D ∪ c
        if prune(c) is true then P ← P ∪ c
            else C_new ← C_new ∪ GenChildren(c, P)
    C ← C_new
D_surprising ← FindSurprising(D)
```

Figure 3: STUCCO: Search and Testing for Understandable Consistent Contrasts

3.1 Finding Significant Contrast Sets

We can check if a contrast-set is significant by testing the null hypothesis that *contrast-set support is equal* *across all groups* or, alternatively, *contrast-set support is independent of group membership.*

The support counts from each group is a form of frequency data which can be analyzed in contingency tables. We form a $2 \times c$ contingency table where the row variable represents the truth of the contrast-set, and the column variable indicates the group membership.

For example, consider the top admitted students at UCI as measured by SAT Verbal scores (SATV > 700) and their school of admission (Arts, Biology, Engineering, Information and Computer Science, and Social Ecology):

	Arts	Bio.	Eng.	ICS	SocEc
$SATV > 700$	45	142	85	60	11
$\neg(SATV > 700)$	583	2465	1523	502	414

If SATV and UCI School are independent variables, then we would expect the proportion of students with high SATV scores to be roughly equal across all groups. Clearly, the proportions are not equal and vary from a high of 10.7% for ICS to a low of 2.6% for Social Ecology. We need to determine if the differences in proportions represent a true relation between the variables or if it can be attributed to random causes.

The standard test for independence of variables in contingency tables is the chi-square test. It works by computing the statistic χ^2:

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \tag{3}$$

where O_{ij} is the observed frequency count in cell ij, and E_{ij} is the expected frequency count in cell ij given independence of the row and column variables and is calculated as follows: $E_{ij} = \sum_j O_{ij} \sum_i O_{ij}/N$ with N being the total number of observations. We then compare the result to the distribution of χ^2 when the null hypothesis is true.

To determine if the differences in proportions are significant, we first pick a test α level. The choice of α sets the maximum probability of rejecting the null hypothesis when it is true. For a single test, α is commonly set to 0.05. We then calculate that $\chi^2 = 35.4$ with 4 degrees of freedom and has a p-value of 3.8e-7. Since the p-value is less than the 0.05 cutoff, we can infer that the null hypothesis is likely false.

3.1.1 Controlling Type I Error

With a single test, α sets the maximum probability of falsely rejecting the null hypothesis. However, with multiple tests, the probability of false rejection can be highly inflated. This is especially true in data mining, where often thousands, or millions, of hypotheses are tested. For example, if the null hypothesis is always true and we made 1000 tests each at $\alpha = 0.05$, we would obtain on average 50 "significant" differences. Falsely rejecting the null hypothesis, i.e., concluding that there is a difference when none exists, is known as a Type I error or false positive.

Type I error can be controlled for a family of tests by using a more stringent α cutoff for the individual tests. We can relate the α_i levels used for each individual test to a global α (the expected error rate) by using the Bonferroni inequality: given any set of events $e_1, e_2, \ldots, e_n$, the probability of their union ($e_1 \vee e_2 \vee \ldots \vee e_n$) is less than or equal to the sum of the individual probabilities. Applied to hypothesis testing, we let e_i be the rejection of the ith hypothesis h_i. Then we reject h_i if $p_i \leq \alpha_i$ where $\sum_i \alpha_i \leq \alpha$. Usually $\alpha_i = \alpha/n$, where n is the total number of tests.

This method controls the *error rate per family* (PFE), which is the expected number of false rejections ($PFE \leq \alpha$), for any combination of true or false hypotheses and holds even with dependent tests [10].

There are two problems with applying this: (1) If we are reporting results incrementally after we mine each level, we do not know how many tests we will make in total. Thus, n is unknown. (2) We use the same cutoff for testing a conjunction of size 1 as size 10. This is undesirable because as α_i gets smaller, we lose power and are less able to detect a difference if it exists. This is an unavoidable tradeoff, as power is related to Type I error. Since lower order conjuncts are more general, we would like more power on those tests.

Because the Bonferroni method holds as long as $\sum_i \alpha_i \leq \alpha$, we can use different α_i for tests at different levels of the search tree as follows:

$$\alpha_l = \min(\frac{\alpha}{2^l}/|C_l|, \alpha_{l-1}) \tag{4}$$

where α_l is the cutoff for level l, and $|C_l|$ is the number of candidates at level l. This apportions 1/2 of the total α risk to tests at level 1, 1/4 to tests at level 2, and so on. The minimum requirement ensures that the test α levels always become more stringent.

3.2 Pruning

We prune a node on the search tree when all specializations of that node can never be a significant and large contrast-set. This is similar to *subset-infrequency* pruning used by Apriori [2] and Max-Miner [3]. Nodes are pruned based on: (1) minimum deviation size, (2) expected cell frequencies, and (3) χ^2 bounds.

Minimum Deviation Size: The deviation size of a contrast-set is the maximum difference between the support of any two groups. We require that this difference is greater than the threshold mindev. This can only occur in the children of a node if the support for at least one group is greater than mindev.

Expected Cell Frequencies: The expected cell frequencies in the top row of the contingency table can only decrease as we specialize the contrast-set. This is important because the validity of the chi-square test depends on approximating the distribution of the χ^2 statistic with the chi-square distribution. When the test is invalid, we prune the node because we cannot make valid inferences. The approximation is made under the assumption that the expected cell frequencies are not "too small." Typically, expected values of 5 are considered satisfactory [7].

χ^2 Bounds: We find an upper bound on the χ^2 statistic for any child of a node and use this to prune candidates when it is no longer possible for specializations to meet the χ^2 cutoff implied by α_l.

The χ^2 contribution from each cell is a function of the observed and expected cell counts where $E_{ij} = \sum_i O_{ij} \sum_j O_{ij}/N$. Notice that the column sum and N are fixed; therefore let $f = \sum_j O_{ij}/N$. We can also break the row sum up into two components: O, the observed value in cell ij, and $R = \sum_{ik,k\neq j} O_{ik}$, the sum of the remainder of the row. Thus $E_{ij} = f(O+R)$, and the χ^2 contribution from cell ij is:

$$\chi^2_{ij}(O,R) = \frac{(O - f(O+R))^2}{f(O+R)} \tag{5}$$

Then the following theorem applies to the χ^2 statistic:
Theorem. If O is bounded by $[O_{min}, O_{max}]$ and R by $[R_{min}, R_{max}]$, then the following is an upper bound on the χ^2 statistic obtainable in any specialization:

$$\chi^2_{\max} = \sum_{i=1}^{r}\sum_{j=1}^{c} \max_{\substack{O_{bound}\in\{O_{min},O_{max}\}\\ R_{bound}\in\{R_{min},R_{max}\}}} (\chi^2_{ij}(O_{bound}, R_{bound})) \tag{6}$$

Proof. We find a maximum value for χ^2 by maximizing the contribution from each cell. $\chi^2_{ij}(O,R)$ is a function of two variables where the feasible region is rectangular and in Quadrant I. The partial derivatives $\frac{\partial \chi^2_{ij}(O,R)}{\partial O}$ and $\frac{\partial \chi^2_{ij}(O,R)}{\partial R}$ are never zero in the feasible region (except at the known minimum where the expected value equals the observed count). Also the $\lim_{O\rightarrow 0^+, R\rightarrow 0^+} = 0$. Since χ^2_{ij} is clearly positive, and there are no relative maxima in the feasible region, then the function maximum must occur on a boundary point. Furthermore, the maximum must occur at a corner, since our feasible region is rectangular.□

3.3 Finding Surprising Contrast Sets

As we mine contrast-sets, we only present those sets which are surprising given what we have already shown. For example, consider that we know P(sex=male | PhD) = 0.81 and P(occupation=manager | PhD) = 0.14. Then under independence of sex and occupation we expect P(sex=male $\wedge$ occupation=manager | PhD) = $0.81 \times 0.14 = 0.113$. Similarly, we expect the probability of being a male-manager is 0.173 for Bachelor holders.

The actual proportions for male managers are 0.109 (Ph.D.) and 0.190 (Bachelor), which are very close to our expected results, and thus are not surprising. So although male-managers is a deviation, we do not show it to the user.

This example was simple and only involved two variables which we assumed were independent. However, we can use this general approach for larger and more complicated sets of variables; i.e., we find the maximum likelihood estimates for a conjunction of variables based on its subsets by using *iterative proportional fitting* [7].

4 Evaluation

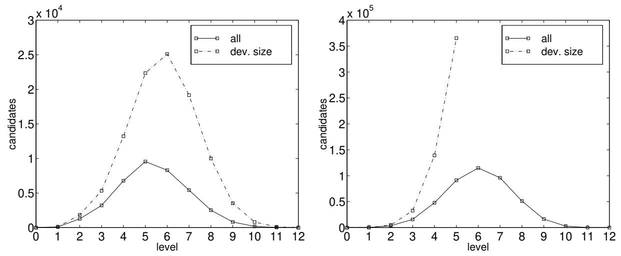
We evaluated STUCCO on two datasets: the Adult Census data from the UCI Repository of Machine Learning Databases [4] (originally from the US Census Bureau), and UCI Admissions Data. We will first present pruning results, and then we will show practical results with examples of mined deviations.

The Adult Census data has 48842 records and 14 variables such as age, working class, education, sex, hours worked, salary, etc. The UCI Admissions data describes applicants to the University of California at Irvine. There are 6 years of data from 1993–98 with about 17000 applicants per year. The data contains 17 variables such as ethnicity, school, sex, home location, first language, GPA, SAT scores, etc. For both databases, continuous attributes were discretized into approximately equal sized intervals.

4.1 Pruning

We compared two pruning strategies: (1) using the minimum deviation size only – this is equivalent to extending Apriori subset infrequency pruning to handle multiple groups, and (2) using all pruning methods: minimum deviation size, expected cell frequencies, and χ^2 bounds. The last two pruning methods can only be used with contrast-sets as they require frequency counts from all groups in the data.

Figure 4 shows the number of candidates counted at each level for two different data sets (mindev = 1%, $\alpha = 0.05$). In both cases the additional pruning methods (expected cell frequencies and χ^2 bounds) significantly reduced the number of candidates that were evaluated. In (b) deviation size pruning ran out of memory.



(a) (b)

Figure 4: Effectiveness of pruning strategies: (a) Bachelor vs. Ph.D. degree recipients, (b) comparison of UCI Schools for 1998.

4.2 Adult Census Data

For the Adult Census data we asked, "What are the differences between people with Ph.D. and Bachelor degrees?" (mindev = 1%, $\alpha = 0.05$). Table 1a summarizes the number of candidates, deviations (significant and large contrast-sets), and surprising sets found at each level. Table 1b shows several mined contrast-sets.

We found over 10000 deviations; however, most were not surprising given their subsets. Thus we reduced the number of returned sets to only 164. In contrast, other mining systems tend to return far more results. Apriori returned over 75000 rules on this dataset.

4.3 UCI Admissions Data

On the UCI Admissions Data we asked, "How has the applicant pool changed from 1993-1998?" (mindev = 5%, $\alpha = 0.05$). Figure 5 shows a subset of the results. The spike in part (a) is probably caused by a change in California state law which (beginning in 1998) barred the UC system from considering ethnicity in admissions.

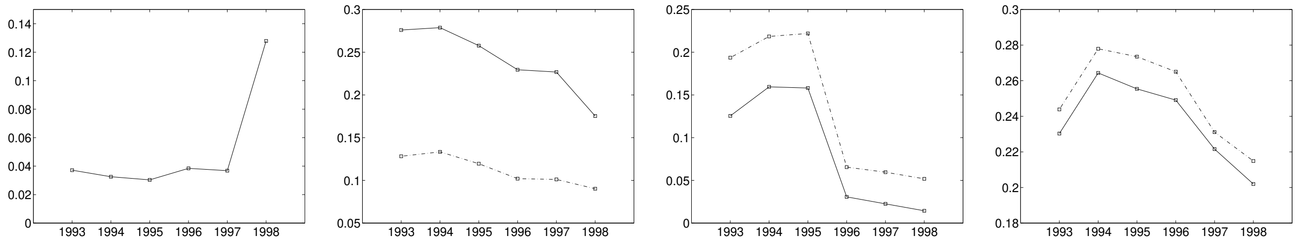
Level	Cand.	Dev.	Surp.
1	108	32	32
2	1273	231	61
3	3207	916	42
4	6777	1946	12
5	9552	2514	3
6	8302	2291	2
7	5419	1550	1
8	2519	790	1
9	811	279	9
10	171	60	1
11	20	6	0
12	1	0	0
total	38160	10615	164

(a) Mining statistics

	Observed %		Expected %			
Contrast-Set	Ph.D.	Bach.	Ph.D.	Bach.	χ^2	p
workclass = State-gov	21.0	5.4			225.1	6.9e-51
occupation = sales	2.7	15.8			74.9	4.8e-18
hour per week > 60	8.4	3.2			43.4	4.4e-11
native country = U.S.	80.5	89.5			45.9	1.3e-11
native country = Canada	1.9	0.5			18.6	1.6e-5
native country = India	1.6	0.5			15.2	9.5e-5
salary > 50K	72.6	41.3			220.2	8.3e-50
sex = male $\wedge$ salary > 50K	61.8	34.8	58.8	28.5	173.6	1.2e-39
occupation = prof-specialty $\wedge$ sex = female $\wedge$ salary > 50K	7.6	2.6	10.7	3.5	48.2	3.8e-12

(b) Mined Contrast Sets

Table 1: Results from Adult Census Data.



(a) Ethnicity = Decline to State

(b) BornInUS = No $\wedge$ First-Language = not English

(c) Admit = Yes $\wedge$ SAT Verbal < 400

(d) HomeLocation = L.A. County $\wedge$ SchoolType = public $\wedge$ Admit = Yes

Figure 5: UCI Applicants 1993-1998. Expected values are shown by the dotted lines.

5 Related Work

Chakrabarti, Sarawagi, and Dom [5] tackle the related problem of finding surprising temporal patterns in market basket data. They use a Minimum Description Length approach where surprising patterns are those with long encoding costs. Our work is fundamentally different. We find differences between two or more probability distributions, whereas they find changes in a single distribution as it varies through time.

Explora [8] searches for subgroups of cases with unusual distributions with respect to a target variable (T) and the parent population: i.e., it finds a subpopulation $G_s \subset G_p$ such that $\mathrm{P}(T \mid G_s) \neq \mathrm{P}(T \mid G_p)$. In contrast, our goal is, given the groups G_1 and G_2, to find conjunctions of variables T_i such that $\mathrm{P}(T_1 \wedge T_2 \wedge \ldots T_n \mid G_1) \neq \mathrm{P}(\ T_1 \wedge T_2 \wedge \ldots T_n \mid G_2)$.

6 Conclusions

We introduced the problem of detecting differences across several contrasting groups as that of finding all contrast-sets, conjunctions of attributes and values, that have meaningfully different support levels. This allows us to answer queries of the form, "How are History and Computer Science students different?" or, "What has changed from 1993 through 1998?"

We combined statistical hypothesis testing with search to develop the STUCCO algorithm for mining contrast-sets. It has (1) admissible pruning rules, (2) guaranteed control over false positives, and (3) compact summarization of results.

Acknowledgments

This research was funded in part by the National Science Foundation grant IRI-9713990. We thank Nira Brand and Wagner Truppel for their comments.

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining associations between sets of items in massive databases. In *Proc. ACM SIGMOD*, pages 207–216, 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. 20th Int. Conf. VLDB*, 1994.

[3] R. J. Bayardo. Efficiently mining long patterns from databases. In *Proc. ACM SIGMOD*, 1998.

[4] C. Blake, E. Keogh, and C. J. Merz. UCI repository of machine learning databases, 1998.

[5] S. Chakrabarti, S. Sarawagi, and B. Dom. Mining surprising patterns using temporal description length. In *Proc. 24th Int. Conf. VLDB*, 1998.

[6] J. Davies and D. Billman. Hierarchical categorization and the effects of contrast inconsistency in an unsupervised learning task. In *Prog. 18th Annual Conf. of the Cognitive Science Society*, page 750, 1996.

[7] B. S. Everitt. *The Analysis of Contingency Tables.* Chapman and Hall, second edition, 1992.

[8] P. Hoschka and W. Klösgen. A support system for interpreting statistical data. In G. Piatetsky-Shapiro and W. J. Frawley, editors, *Knowledge Discovery in Databases*, pages 325–346. AAAI Press, 1991.

[9] P. Riddle, R. Segal, and O. Etzioni. Representation design and brute-force induction in a Boeing manufacturing domain. *Applied Artificial Intelligence*, 8:125–147, 1994.

[10] J. P. Shaffer. Multiple hypothesis testing. *Annual Review Psychology*, 46:561–584, 1995.