PARAS: Interactive Parameter Space Exploration for Association Rule Mining[*]

Abhishek Mukherji, Xika Lin, Christopher R. Botaish, Jason Whitehouse,
Elke A. Rundensteiner, Matthew O. Ward and Carolina Ruiz
Computer Science Department, Worcester Polytechnic Institute
100 Institute Road, Worcester MA, USA.
mukherab|xika|cbotaish|jwhitehouse|rundenst|matt|ruiz@wpi.edu

ABSTRACT

We demonstrate our PARAS technology for supporting interactive association mining at near real-time speeds. Key technical innovations of PARAS, in particular, *stable region abstractions* and *rule redundancy management* supporting novel parameter space-centric *exploratory queries* will be showcased. The audience will be able to interactively explore the parameter space view of rules. They will experience near real-time speeds achieved by PARAS for operations, such as comparing rulesets mined using different parameter values, that would otherwise take hours of computation and much manual investigation. Overall, we will demonstrate that the PARAS system provides a rich experience to data analysts while significantly reducing the trial-and-error interactions.

Keywords

Association Rules, Interestingness Parameters, Visual Mining

1. INTRODUCTION

1.1 Motivation

Mining of associations and correlations from huge data sets is critical for applications ranging from market basket analysis [2] and bioinformatics [9] to intrusion detection and web usage mining [8]. Data analysts often need to perform numerous successive trial-and-error interactions and compare mining results with varying parameter values in order to find interesting rules. Further, analysts may need to interactively separate spurious rules from genuine ones. Existing rule mining algorithms [2, 6] are compute-intensive, rendering even their fast implementations, such as in [4], unfit for interactive analysis. Mining systems with delayed response times risk losing a user's attention and, more importantly, are often unacceptable in mission critical applications.

Besides having high response times, the parameterized nature of these mining algorithms poses another challenge. Poor selection of parameter values may lead to failure in discovering true associations. Often the algorithms may report spurious associations that do not really exist, or greatly overestimate the significance of the reported associations. The tasks of distinguishing spurious rules from genuine ones, as part of sense-making of the mined rules, require much manual effort with little or no help from existing systems [5, 10]. The role of parameter selection is significant, yet appropriate parameter values are difficult to determine apriori as they tend to differ for different datasets, contexts and analysts' objectives. Therefore, an interactive data mining system, capable of not only answering mining queries but also providing parameter tuning recommendations at near real-time speeds, is important for decision making applications as motivated by the following example:

Motivating Example: The study of protein-DNA bindings between transcription factors (TFs) and transcription factor binding sites (TFBSs) is an important bioinformatics topic [9]. A transcription factor (TF) is a special type of protein that binds to a region of DNA called a transcription factor binding site (TFBS) to regulate (activate and control) the expression of a gene. Traditionally, high-resolution (length < 10) TF-TFBS binding cores are discovered by expensive and time-consuming 3D structure experiments.

Recent association rule mining approaches on low-resolution binding sequences (TF length > 490) are shown to be promising in identifying accurate binding cores without the use of costly 3D structure experiments [9]. An example rule in the MIPS Yeast Genome dataset looks like R1 = $[\{0.\text{transcription}\} \Rightarrow \{0.\text{nucleus}\}]$ with support = 0.7% and confidence = 69.59%, where $\{0.\text{transcription}\}$ is a TF k-mer and $\{0.\text{nucleus}\}$ is a TFBS k-mer [9]. Thus, biologists interactively mine associations over experimental data to find candidate protein-DNA rules for subsequent binding experiments. The choice of appropriate parameters, to discover genuine candidate rules, is important yet difficult to know apriori. Therefore, strong motivation exists for interactive rule mining support to accelerate such experiments. First, real-time responsiveness is important as the biologist can seldom wait hours for the mining process to finish. Further, some users may give more importance to one parameter than the other. Consider another rule R2 = $[\{0.\text{nucleus}\} \Rightarrow \{1.\text{nucleus}\}]$ with support = 5.74% confidence = 29.02%. While R1 has a much higher confidence than R2, R2 has higher support than R1. Then, the user may miss one of these rules unless lower values of both parameters are chosen. We will demonstrate that a comprehensive parameter space view[1], as in our work, can provide useful insights to the biologists about the distribution of rules in the entire parameter space.

In this demonstration, we present our **PARAmeter Space Model for Interactive Association Mining** (**PARAS**)[2] system that over-

[*] This work was partly supported by the National Science Foundation under grants IS-0812027, CCF-0811510 and IIS-1117139.

SIGMOD'13, June 22–27, 2013, New York, New York, USA.
Copyright 2013 ACM 978-1-4503-2037-5/13/06 ...$10.00.

[1] We use the popular rule interestingness measures, namely, support and confidence as the parameter space dimensions.

[2] Hindi name for a legendary philosopher's stone said to be capable of turning base metals (lead, for example) into gold.

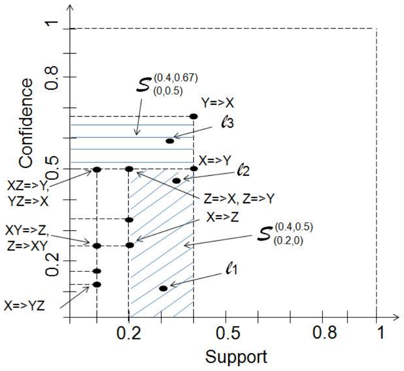


Figure 1: Parameter Space

S. Regions	Neighbors	Unique Rules
$\mathcal{S}^{(0.4,0.67)}_{(0,0.5)}$	$\varnothing$	{(Y⇒X)}
$\mathcal{S}^{(0.4,0.5)}_{(0.2,0)}$	$\mathcal{S}^{(0.4,0.67)}_{(0,0.5)}$	{(X⇒Y)}
$\mathcal{S}^{(0.2,0.5)}_{(0.1,0.33)}$	$\mathcal{S}^{(0.4,0.5)}_{(0.2,0)}$	{(Z⇒X),(Z⇒Y)}
$\mathcal{S}^{(0.2,0.33)}_{(0.1,0.25)}$	$\mathcal{S}^{(0.2,0.5)}_{(0.1,0.33)}$	{(Y⇒Z)}
$\mathcal{S}^{(0.2,0.25)}_{(0.1,0)}$	$\mathcal{S}^{(0.2,0.33)}_{(0.1,0.25)}$	{(X⇒Z)}
$\mathcal{S}^{(0.1,0.5)}_{(0,0.33)}$	$\mathcal{S}^{(0.2,0.5)}_{(0.1,0.33)}$	{(XZ⇒Y),(YZ⇒X)}
$\mathcal{S}^{(0.1,0.33)}_{(0,0.25)}$	$\mathcal{S}^{(0.1,0.5)}_{(0,0.33)} + \mathcal{S}^{(0.2,0.33)}_{(0.1,0.25)}$	{$\varnothing$}
$\mathcal{S}^{(0.1,0.25)}_{(0,0.16)}$	$\mathcal{S}^{(0.1,0.33)}_{(0,0.25)} + \mathcal{S}^{(0.2,0.25)}_{(0.1,0)}$	{(XY⇒Z),(Z⇒XY)}
$\mathcal{S}^{(0.1,0.16)}_{(0,0.125)}$	$\mathcal{S}^{(0.1,0.25)}_{(0,0.16)}$	{(Y⇒XZ)}
$\mathcal{S}^{(0.1,0.125)}_{(0,0)}$	$\mathcal{S}^{(0.1,0.16)}_{(0,0.125)}$	{(X⇒YZ)}

Figure 2: Stable Regions

comes the above challenges by offering real-time responsiveness and enhanced sense-making of mined rules. Over the past 15 years the XMDV team at WPI, composed of visualization, HCI and database experts, supported by a series of six NSF grants, has developed a freeware visual tool suite XmdvTool [10] to facilitate interactive data exploration. We currently focus on extending XmdvTool to support interative parameter space exploration for mining of rules.

Through the simple yet effective PARAS interactive visualizations, the audience will be able to experience the benefits of using the PARAS model [7] over using other existing mining technologies. First, for benchmark datasets such as IBM Quest [2] and webdocs [8], PARAS demonstrates 2 to 5 orders of magnitude improvement in response times over the state-of-the-art techniques [4]. In particular, our experiments [7] confirm that even for the large sized datasets (e.g. webdocs [8] $\sim$ 1.5 GB) PARAS responds in less than a second while state-of-the-art technologies may take several hours to compute the results. Second, PARAS provides a competitive advantage to the analysts by making recommendations for parameter tuning. Thus, they help the analysts to extract desired associations using significantly fewer trial-and-error cycles. Third, PARAS enables the analysts to perform real-time in-depth investigation of the query parameter space via a rich set of novel *exploratory mining queries*. We demonstrate the powerful interactive capabilities of PARAS using two datasets, namely, the MIPS Yeast Genome [9] and the Adult Census [3] datasets.

2. THE PARAS FRAMEWORK

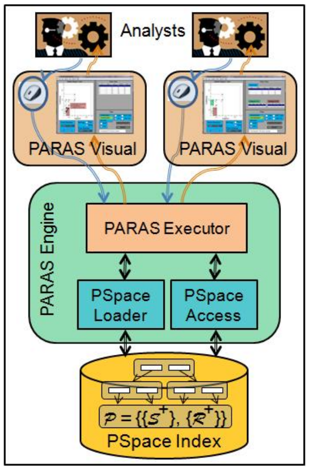


Figure 3: PARAS Framework

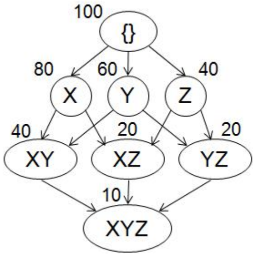


Figure 4: Adjacency Lattice

The PARAS Framework is depicted in Figure 3. The interactive user requests are supported via the PARAS visualizer allowing the analysts to interact with a two-dimensional parameter space, or in short PSpace (Figure 1), to submit mining requests and navigate through responses in a visual manner. The requests are passed to the PARAS Processor for efficient execution. The PSpace Access module offers the API for accessing the PSpace index that had been constructed in an offline step by the PARAS index loader. The index compactly stores stable regions along with their association rules and redundancy abstractions (Figure 2) as explained below.

3. KEY INNOVATIONS OF PARAS SYSTEM

The PARAS system encompasses several innovations that form the foundation for the effective management and exploration of rules within the parameter space. Our key technical contributions, namely, **stable region abstractions** of parameter space, offline and online **redundancy resolution management**, and novel **sense-making query support**, are introduced below.

3.1 PARAS Stable Region Abstractions

An adjacency lattice [1] in Figure 4 denotes items such as X, Y and Z that may represent proteins or DNA in our biological domain. The *support* value of each item (say, X) or itemset (say, XY) indicates the total instances of the item or itemset in the dataset. For example, in a total of 100 records, X occurs in 80 and Y in 60 records. Itemset XY has a support of 40 records. We use the PSpace (Figure 1) as a model for managing and exploring the association rules extracted from a dataset. For simplicity, we henceforth work with a two-dimensional PSpace using *support* and *confidence* as dimensions. A *parametric location* ℓ_1 is a point in the space constructed by support and confidence dimensions, denoted by (ℓ_1.supp, ℓ_1.conf) (Figure 1). Many association rules may map to the same *parametric location*, e.g., (XZ ⇒Y) and (YZ ⇒X) both map to (0.1,0.5). All rules that map to the same parametric location are compactly indexed in our PSpace model.

In Figure 1, we observe that many regions of the parameter space either contain no rules or contain the same set of rules across a large range of parameter settings (e.g., the shaded regions marked $\mathcal{S}^{(0.4,0.5)}_{(0.2,0)}$ and $\mathcal{S}^{(0.4,0.67)}_{(0,0.5)}$). We call them *stable regions*. They form our *coarse granularity abstractions* for storing and managing rules. Thus, the parameter space can be divided into several stable regions such that the ruleset valid for any two parametric locations within a stable region remains unchanged, whereas rulesets valid for two locations not in the same stable region must be different. Also, rule (Y $\Rightarrow$ X) first appears in region $\mathcal{S}^{(0.4,0.67)}_{(0,0.5)}$ and is also valid

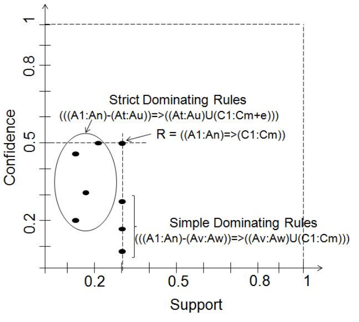


Figure 5: Dominating Rules

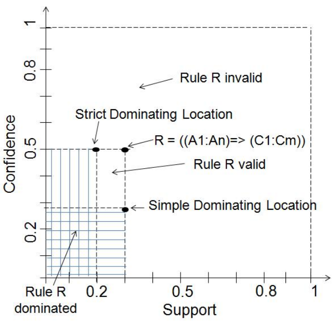


Figure 6: Dominating Locations

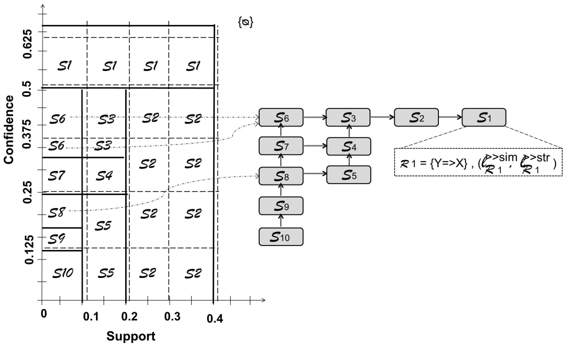


Figure 7: The PSpace Index

for region $\mathcal{S}^{(0.4,0.5)}_{(0.2,0)}$. Therefore, $\mathcal{S}^{(0.4,0.67)}_{(0,0.5)}$ is the *lending neighbor stable region*, or in short *l-neighbor*, for $\mathcal{S}^{(0.4,0.5)}_{(0.2,0)}$.

We partition the *parameter space* $\mathcal{P}$ into a finite number of *non-overlapping stable regions*, denoted by $\{\mathcal{S}\}$. As depicted in Figure 2, for each such *stable region* we maintain (a) the rules that are valid within that region and (b) the links to its *l-neighbors*. This stable region abstraction of the parameter space forms a solid foundation for efficient processing of novel exploratory mining queries as well as for recommending parameter settings based on user interest.

3.2 PARAS Rule Redundancy Management

Rule	Support	Confidence
X⇒YZ	S(X ∪ Y ∪ Z) = 0.1	S(X ∪ Y ∪ Z) / S(X) = 0.125
XY⇒Z	S(X ∪ Y ∪ Z) = 0.1	S(X ∪ Y ∪ Z) / S(X ∪ Y) = 0.25
XZ⇒Y	S(X ∪ Y ∪ Z) = 0.1	S(X ∪ Y ∪ Z) / S(X ∪ Z) = 0.5
X⇒Y	S(X ∪ Y) = 0.4	S(X ∪ Y) / S(X) = 0.5
X⇒Z	S(X ∪ Z) = 0.2	S(X ∪ Z) / S(X) = 0.25

Table 1: Redundancy Among Rules

Redundancy relationships among rules are defined to filter out redundant rules for presenting succinct query results to the user [1]. In particular, two types of redundancies exist among rules, namely, *simple* and *strict*. In Table 1, rule (X ⇒ YZ) *simple dominates* the rules (XY ⇒ Z) and (XZ ⇒ Y) and *strict dominates* rules (X ⇒ Y) and (X ⇒ Z). In general, a rule may be dominated by several *dominating* rules and may in turn dominate several other *dominated* rules. Moreover, we now observe that redundancy is a **query-time phenomenon** and dependent on the user parameter selection. Thus, rules cannot be tagged as redundant and discarded apriori.

We examine how such rule redundancies can be identified in the PSpace model [7]. In Figure 5, rule R = ((A_1:A_n) ⇒ (C_1:C_m)) is *simple dominated* by all rules with template (((A_1:A_n)-(A_v:A_w)) ⇒ ((A_v:A_w) ∪ (C_1:C_m))), whereas rule R is *strict dominated* by all rules with template (((A_1:A_n)-(A_t:A_u)) ⇒ ((A_t:A_u) ∪ (C_1:C_{m+e}))). Maintaining all dominating rules for every rule is memory and compute-intensive. Fortunately, we have discovered surprisingly compact representation of rule redundancies in the context of our PSpace model. In Figure 6, we show that it is sufficient to compare each candidate rule R with only two dominating locations instead of the large number of dominating rules. Thus, for N rules in the output ruleset, while state-of-the-art online redundancy resolution [1] takes $\mathcal{O}(2^N)$ time, our newly proposed online redundancy resolution solution takes $\mathcal{O}$(N) time by performing a $\mathcal{O}(\mathrm{N}^2)$ time offline redundancy abstraction step [7].

3.3 PARAS PSpace Exploration Queries

Our **PARAS** framework supports a rich variety of analytical queries over the PSpace index (Figure 7). They are broadly classified as (a.) rule mining (RM), (b.) Stable Region (SR) and (c.) Redundancy Resolution (RR) queries. Below, we briefly present sample queries in each category, while the details of respective processing strategies and cost analysis can be found in [7].

Rule Mining (RM) Queries: For a given dataset $\mathcal{D}$, query Q1 (below) finds the set of rules that satisfy query parameters (*minsupp*,*minconf*). The *WITH Redundancy Elimination* clause gives users the option to output only non-redundant rules. In case of an overwhelmingly large number of rules valid for parameter settings (*minsupp*,*minconf*), PARAS offers analysts the choice of viewing only the non-redundant rules for the setting. The RM query takes $\mathcal{O}$(N) time for N rules valid for input (*minsupp*, *minconf*).

```
Q1:  OUTPUT RuleSet $\{\mathcal{R}\}^{(minsupp,minconf)}$
FROM $\mathcal{D}$
HAVING minsupport=minsupp, minconfidence=minconf
WITH Redundancy Elimination = T/F;
```

Stable Region (SR) Queries: This query type (e.g., Query Q2) identifies the stable region $\mathcal{S}^{(usupp,uconf)}_{(lsupp,lconf)}$ containing the user-chosen parameter setting (*minsupp*,*minconf*) through a constant time ($\mathcal{O}(1)$) lookup over the PSpace index. The analyst can further compare the unique rules of a region with the rules borrowed from the l-neighbors. If the rules within the current region are found to be not interesting, instead of random parameter selections, the analysts can quickly jump to the parameter settings of the l-neighbors greatly reducing the number of trial-and-error interactions.

```
Q2:  OUTPUT Stable Region $\mathcal{S}^{(usupp,uconf)}_{(lsupp,lconf)}$
FROM $\mathcal{P}$
HAVING minsupport=minsupp, minconfidence=minconf
```

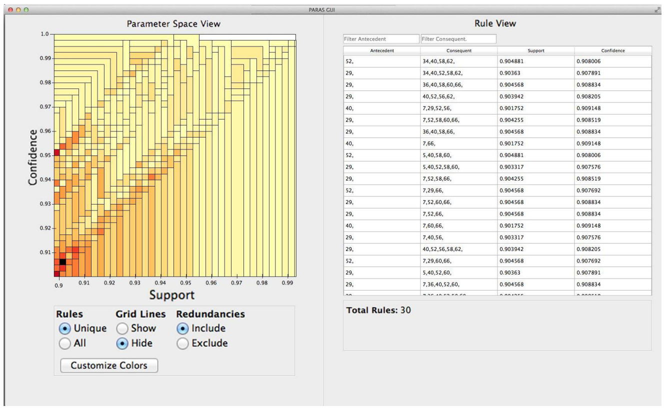
Redundancy Relationship (RR) Queries: For the stable region containing input (*minsupp*,*minconf*), Query Q3 obtains the dominating regions (locations) for its rules. For each rule R, the dominating locations define the area of the parameter space in which R will be dominated (Figure 6). The RR query incurs $\mathcal{O}$(N) time complexity for N rules valid in $\mathcal{S}^{(usupp,uconf)}_{(lsupp,lconf)}$.

```
Q3:  OUTPUT Dominating Regions $\mathcal{S}^{(usupp,uconf)}_{(lsupp,lconf)}.\{\mathcal{S}^{\gg}\}$
FROM $\mathcal{P}$
HAVING minsupport=minsupp, minconfidence=minconf
```

These and other key innovations of the PARAS system [7] form the foundation for the user interactions supported via the PARAS visualizer, as further described in our demo.

4. PARAS DEMONSTRATION

Our demonstration will illustrate how an analyst can explore a dataset using the parameter space-centric interactions of our PARAS



Rules: ● Unique ○ All

Grid Lines: ○ Show ● Hide

Redundancies: ● Include ○ Exclude

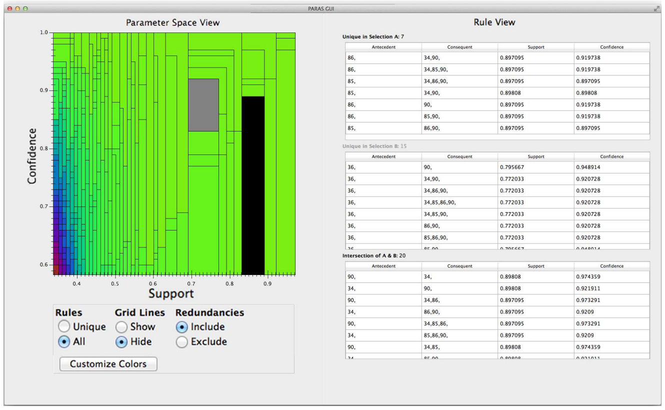
Rule View

Filter Antecedent | Filter Consequent.

Antecedent	Consequent	Support	Confidence
52,	34,40,58,62,	0.904881	0.908006
29,	34,40,52,58,62,	0.90363	0.907891
29,	36,40,58,60,66,	0.904568	0.908834
29,	40,52,56,62,	0.903942	0.908205
40,	7,29,52,56,	0.901752	0.909148
29,	7,52,58,60,66,	0.904255	0.908519
29,	36,40,58,66,	0.904568	0.908834
40,	7,66,	0.901752	0.909148
52,	5,40,58,60,	0.904881	0.908006
29,	5,40,52,58,60,	0.903317	0.907576
29,	7,52,58,66,	0.904255	0.908519
52,	7,29,66,	0.904568	0.907692
29,	7,52,60,66,	0.904568	0.908834
29,	7,52,66,	0.904568	0.908834
40,	7,60,66,	0.901752	0.909148
29,	7,40,56,	0.903317	0.907576
29,	40,52,56,58,62,	0.903942	0.908205
52,	7,29,60,66,	0.904568	0.907692
29,	5,40,52,60,	0.90363	0.907891
29,	7,36,40,52,60,	0.904568	0.908834

Total Rules: 30

Figure 8: PARAS: Single Region Selection



Rules: ○ Unique ● All

Grid Lines: ○ Show ● Hide

Redundancies: ● Include ○ Exclude

Rule View

Unique in Selection A: 7

Antecedent	Consequent	Support	Confidence
86,	34,90,	0.897095	0.919738
86,	34,85,90,	0.897095	0.919738
85,	34,86,90,	0.897095	0.897095
85,	34,90,	0.89808	0.89808
86,	90,	0.897095	0.919738
86,	85,90,	0.897095	0.919738
85,	86,90,	0.897095	0.897095

Unique in Selection B: 15

Antecedent	Consequent	Support	Confidence
36,	90,	0.795667	0.948914
36,	34,90,	0.772033	0.920728
36,	34,86,90,	0.772033	0.920728
36,	34,85,86,90,	0.772033	0.920728
36,	34,85,90,	0.772033	0.920728
36,	86,90,	0.772033	0.920728
36,	85,86,90,	0.772033	0.920728

Intersection of A & B: 20

Antecedent	Consequent	Support	Confidence
90,	34,	0.89808	0.974359
34,	90,	0.89808	0.921911
90,	34,86,	0.897095	0.973291
34,	86,90,	0.897095	0.9209
90,	34,85,86,	0.897095	0.973291
34,	85,86,90,	0.897095	0.9209
90,	34,85,	0.89808	0.974359

Figure 9: PARAS: Comparing Two Regions

visualizer. Datasets from two domains will be used, namely, the MIPS Yeast Genome [9] and the Adult Census [3] datasets. The intuitive interactions provide comprehensive knowledge at two granularities, namely, *Stable Region* and *Rule* Views.

The Two Views: The audience will be first shown the interaction capabilities over the *Stable Region View* (LHS of Figure 8). The stable region view consists of a two-dimensional plot of the stable regions within a space of support (x-axis) and confidence (y-axis) dimensions. Depending on the distribution of rules within the two-dimensional space, datasets may differ in number, size and density of the stable regions as shown in Figures 8 and 9. The audience may double-click the mouse or trackpad and move up to zoom in and down to zoom out. Each dataset will be preloaded with some default zoom settings. We employ the color mapping schemes given in XmdvTool [10]. The audience will see that each stable region is visually marked by a color or shade of color to denote the count of rules within that region. The choice of color scheme depends on the number of distinct count values. If it is low and sparse such as in the MIPS Genome dataset [9], a single color with intensity denoting the count is sufficient (Figure 8). Here, lighter color will depict low count and darker will depicts high count. But if the number of distinct count values is high and dense such as in the Adult Census dataset [3], a multi-color scheme is shown to work better (Figure 9). The audience can alternate between the *UNIQUE* (Figure 8) and *ALL* (Figure 9) radio button options. With *UNIQUE*, the audience will be able to view only the unique rules within each stable region. *ALL* is the default setting that displays all rules within each region. Notice that for *ALL* the shade becomes cummulatively darker towards lower values of support and confidence as rules valid at higher values are also valid at lower values, while, for the *UNIQUE*, setting the color depends on the count of unique rules within each region. The *Rule View* (RHS of Figure 8) lists the rules in the selected region.

The *UNIQUE* setting is useful in exploring the lower regions of support and confidence for unique rules. Say, rules R1 and R2 in the motivating example (Section 1.1) are found to be spurious by domain experts despite their high parameter values. Further, rule R3 = [{0.transcription} ⇒ {1.nucleus}] with support = 0.29% and confidence = 28.38% is found to be an important discovery by biologists despite lower support and confidence values. In general, biologists find that the important rules are found in regions of low parameter settings with support around 0.2-2 % and confidence around 6-20%. When such low minsupport and minconfidence settings are applied over existing mining systems, spurious rules will be produced ranking higher than the important ones. In such situations, an analyst can choose the *UNIQUE* setting to view only a handful of rules unique to the low parameter settings. Further the audience can use the redundancy radio buttons to optionally view rules with or without redundancy resolution based on Section 3.2.

Single Region Selection: In Figure 8, when the analyst clicks on a single region, it gets highlighted and the actual rules can be viewed in the Rule View list. In cases of overwhelmingly large number of rules, a biologist can selectively view a subset of rules by applying filters on the input boxes for antecedent and consequent available in the Rule View. Further, the audience can sort the rules in the Rule View table by support and confidence.

Two Region Comparison: Figure 9 depicts another walkthrough scenario comparing two stable regions. Here, the audience will select two regions on the *Stable Region View*, one with a left click (region A in black) and the other with shift+click (region B in grey). Through cross links, the Rule View will present a comparative display of common and unique rules among the two chosen regions.

Conclusion: We demonstrate the PARAS system that provides a rich experience to data analysts through parameter recommendations while significantly reducing the trial-and-error interactions.

5. REFERENCES

[1] C. C. Aggarwal and P. S. Yu. A new approach to online generation of association rules. *IEEE Trans. Knowl. Data Eng.*, 13(4):527–540, 2001.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB*, pages 487–499, 1994.

[3] A. Asuncion and D. Newman. UCI machine learning repository. `http://www.ics.uci.edu/~mlearn/MLRepository.html`, November 2007.

[4] C. Borgelt. Efficient implementations of apriori, eclat and fp-growth. `http://www.borgelt.net`, October 2012.

[5] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, 2009.

[6] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *SIGMOD*, pages 1–12, 2000.

[7] X. Lin, A. Mukherji, E. A. Rundensteiner, C. Ruiz, and M. O. Ward. Paras: Parameter space framework for online association mining. In *PVLDB*, volume 6 (3), 2013.

[8] C. Lucchese, S. Orlando, R. Perego, and F. Silvestri. Webdocs: a real-life huge transac. dataset. In *FIMI*, 2004.

[9] P.-Y. Wong, T.-M. Chan, M.-H. Wong, and K.-S. Leung. Predicting approximate protein-dna binding cores using association rule mining. In *IEEE ICDE*, 2012.

[10] Xmdvtool. http://davis.wpi.edu/ xmdv/, March 2013.