The Electronic Primaries: Predicting the U.S. Presidency Using Feature Selection with Safe Data Reduction

Pablo Moscato **Luke Mathieson** **Alexandre Mendes** **Regina Berretta**

Newcastle Bioinformatics Initiative
School of Electrical Engineering and Computer Science
Faculty of Engineering and the Built Environment
University of Newcastle
Callaghan NSW 2308

Abstract

The data mining inspired problem of finding the critical, and most useful features to be used to classify a data set, and construct rules to predict the class of future examples is an interesting and important problem. It is also one of the most useful problems with applications in many areas such as microarray analysis, genomics, proteomics, pattern recognition, data compression and knowledge discovery. Expressed as k-Feature Set it is also a formally hard problem. In this paper we present a method for coping with this hardness using the combinatorial optimisation and parameterized complexity inspired technique of sound reduction rules. We apply our method to an interesting data set which is used to predict the winner of the popular vote in the U.S. presidential elections. We demonstrate the power and flexibility of the reductions, especially when used in the context of the $(\alpha, \beta)k$-Feature Set variant problem.

1 Introduction

The prediction of the next U.S. president is an important pastime of political pundits in the U.S. and the rest of the world. One may consider that such a complex, large problem, with many variables would be insoluble to methodical systems. Despite this apparent difficulty, Lichtman and Keilis-Borok determined in 1981 (Lichtman & Keilis-Borok 1981) that a system of only twelve questions was needed to predict the swing of the popular vote in the United States.

This study uses this problem to demonstrate the generalisability of a system developed around the k-Feature Set problem and the determination of lesion pathologies in breast cancer cases (Mathieson et al. 2004). The prediction problem presented here gives a good 'toy'[1] problem to work with, as it allows transparent demonstration of the principles of the reduction technique, and further clarification of the importance and usefulness of the confidence measures inherent in the system.

[1] Note that 'toy' refers here to the size of the problem, which in terms of a computational problem is tiny.

2 k-Feature Set and its Variants

The k-Feature Set problem as considered here, derives from the field of data mining and knowledge discovery. It asks whether, in a set of m examples, each with n features, there is a k size subset of the features that explains some dichotomy within the examples. Formally:

k-Feature Set
Instance: A boolean $m \times n$ matrix $\mathcal{M}$, a boolean $m \times 1$ target vector $\mathcal{T}$, and a positive integer k.
Parameter: k
Question: $\exists S \subseteq [1, \ldots, n]$, $|S| \leq k$ such that $\forall i, j \in [1, \ldots, m]$ where $\mathcal{T}_i \neq \mathcal{T}_j$ $\exists s \in S$ such that $\mathcal{M}_{i,s} \neq \mathcal{M}_{j,s}$?

Clearly this problem can be generalized quite easily to deal with non-boolean, discrete entries in both $\mathcal{M}$ and $\mathcal{T}$. Unfortunately not only is this problem NP-Complete (Davies & Russell 1994), but also W[2]-Complete (Cotta & Moscato 2003). This intractability, in both the classical and parameterized sense, suggests that the problem must be attacked through heuristic means. It is interesting to note however, that although this problem is inherently hard (in the formal sense), there are still reduction rules that seem to deal with the problem very effectively.

In this paper we are concerned with a generalisation of k-Feature Set called $(\alpha, \beta)k$-Feature Set. This variant takes into consideration the possibility of choosing a set of features that maximizes the number of differences between two examples with different target values, and maximizes the similarities between examples with the same target value. Formally:

$(\alpha, \beta)k$-Feature Set
Instance: A discrete valued $m \times n$ matrix $\mathcal{M}$, a discrete valued $m \times 1$ target vector $\mathcal{T}$, and positive integers α, β and k.
Parameter: k
Question: $\exists S \subseteq [1, \ldots, n]$, $|S| \leq k$ such that $\forall i, j \in [1, \ldots, m]$ and
- if $\mathcal{T}_i \neq \mathcal{T}_j$, $\exists S' \subseteq S$ where $|S'| \geq \alpha$ and $\forall s \in S'$ $\mathcal{M}_{i,s} \neq \mathcal{M}_{j,s}$
- if $\mathcal{T}_i = \mathcal{T}_j$, $\exists S' \subseteq S$ where $|S'| \geq \beta$ and $\forall s \in S'$ $\mathcal{M}_{i,s} = \mathcal{M}_{j,s}$?

Note that if we choose $\alpha = 1$ and $\beta = 0$ then we return immediately to k-Feature Set. Thus this more general version is also intractable.

2.1 $(\alpha, \beta)k$-Feature Set as a Graph Problem

The $(\alpha, \beta)k$-Feature Set problem can be easily represented in the form of a bi-partite graph, where we have a vertex for each feature, and a vertex for each pair of examples. Edges are inserted between a pair vertex and a feature vertex when the examples differ in that feature, if they are from different classes[2], or if they are the same in that feature, if they are from the same class. Now it is easy to see that the k-Feature Set problem is equivalent to the Red/Blue-Dominating Set problem (Downey & Fellows 1997) where we must choose at most k of the feature vertices to dominate the pair vertices. For the $(\alpha, \beta)k$-Feature Set variant that we deal with, we must choose at most k vertices from the set of feature vertices that dominate pairs representing examples from different classes α times, and dominate pairs representing examples from the same class β times. Precisely:
Given an instance of the $(\alpha, \beta)k$-Feature Set problem, construct a bi-partite graph $G = (A \cup B, F, E)$, such that $\forall i, j \in [1, \ldots, m], i \neq j$ if $\mathcal{T}_i \neq \mathcal{T}_j$ then $\exists v_{ij} \in A$, and $\exists v_{ij} \in B$ otherwise. Further $\forall s \in [1, \ldots, n]$ $\exists f_s \in F$. Thus $\forall i, j \in [1, \ldots, m], i \neq j, \forall s \in [1, \ldots, n]$ if $v_{ij} \in A$ and $\mathcal{M}_{i,s} \neq \mathcal{M}_{j,s}$ or if $v_{ij} \in B$ and $\mathcal{M}_{i,s} = \mathcal{M}_{j,s}$, then $\exists (v_{ij}, f_s) \in E$.

That is, we construct the graph out of three sets of vertices, A, which we will call the 'alpha' vertices, B, the 'beta' vertices, and F, the 'feature' vertices. Each 'alpha' vertex represent a pair of examples from different classes (i.e. that have different entries in the target vector $\mathcal{T}$). Each 'beta' vertex represents a pair of examples from the same class. Each feature in the original matrix has its own vertex in F. Edges exist only from A to F and B to F[3]. If, in the original data matrix, two examples i, j have different classes, and differ in feature f, then there is an edge from vertex $v_{i,j} \in A$ to vertex v_f. If they are in the same class, and are the same in feature f, then we place an edge between $v_{i,j} \in B$ and v_f.

2.2 Data Reduction

With many intractable problems, the technique of data reduction, or reduction to a 'problem kernel' in the parameterized setting, is an important method for creating practical tools to use with the problems as it often allows algorithms that are efficient in the size of the instance, with the non-polynomial component confined to the parameter, which is fixed for a given instance. The fundamental idea of data reduction is to pre-process the instance with a set of rules that reduce the size of the data set without losing optimal solutions. This sort of technique has long been used in Operations Research and related fields, but has received very little attention outside of these areas. With the formalisation of the concept of data reduction and the development of accompanying analysis and complexity tools by Downey and Fellows (Downey & Fellows 1997), and the demonstration that many of these long standing reduction techniques are not in fact heuristic, this approach for dealing with large, complex data sets is slowly gaining momentum.

The reductions that we apply to the $(\alpha, \beta)k$-Feature Set problem come from Weihe (Weihe 1998), as applied to the Red/Blue Dominating Set problem[4], with appropriate modification dealing with α and β domination greater than 1 and 0 respectively (Cotta, Sloper & Moscato 2004). First define d_v to be the number of times it is necessary to further dominate the vertex v, initially d_v will be either α or β, if v is in A or B respectively. Further let $N(u)$ denote the (open) neighbourhood of u, that is, all vertices attached to u, not including u itself. The rules are:

1. If there exists a pair vertex v and $(deg(v) = d_v)$, then
   - add $N(v)$ to the dominating set
   - for each $f \in N(v)$ decrease d_u for every $u \in N(f)$
   - remove v and $N(v)$ from the graph
2. If there exists two feature vertices f_1 and f_2 such that $N(f_1) \subset N(f_2)$ where for every $u \in N(f_1)$ where $deg(u) - d_u > 0$, then
   - remove f_1 from the graph
3. If there exists two pair vertices v_1 and v_2 such that $N(v_1) \subset N(v_2)$ and $d_{v_1} \geq d_{v_2}$ then
   - remove v_2 from the graph

In other words:

1. If there is a pair vertex that needs to be dominated x more times to reach the requisite domination number (α or β as appropriate), and it is only connected to x features, all these features must be in the feature set. Thus for each of these x features we can mark its neighbours as being dominated by that feature, and remove that feature from the graph. Further, as the original pair vertex is now sufficiently dominated, we can remove it from the graph too.
2. If we have two features, one whose neighbourhood is a subset of the other's, and for every pair vertex attached to the smaller feature we do not need both to reach the appropriate domination number, then the smaller feature would never be chosen over the larger feature, as the larger feature does all the work of the smaller feature, and possibly more, thus we can safely remove the smaller feature[5].
3. If we have two pair vertices, such that the neighbourhood of one is a subset of the neighbourhood of the other, and the smaller needs to be dominated at least as many times as the larger, then we know that we are going to have to choose sufficient of the smaller's neighbours to also dominate the larger, thus we need not consider the larger, as it will be automatically dealt with if the smaller is.[6]

[2] The examples i, j are in different classes if $\mathcal{T}_i \neq \mathcal{T}_j$.

[3] The edges are undirected, the use of 'to' is not strict.

[4] These rules do have earlier genesis in at least combinatorial optimisation, Weihe however provides an eminently relevant formulation and application.

[5] Note here that the rule is specifically stated for strict subsets. It can be reformulated such that if the two features have precisely the same neighbourhood, then they are merged, rather than one being deleted, as they would be equivalent in their use. We have chosen not to use this as it complicates the final procedure of producing decision trees and rules, and we currently have no method for dealing with this complication.

[6] Again we can potentially merge pair vertices here if the neighbourhoods were equal. However as the pair vertices are merely representatives, and are not produced in the solution, merging is an unnecessary complication.

What remains after these reductions are applied completely, and no further reduction can be applied, is called the 'kernel', or 'problem kernel'.

These reduction rules allow pre-processing of the data to indicate features that must be in the feature set (Rule 1), features that can be discarded (Rule 2), pairs that provide no extra information about the solution (Rule 3), and reduces the size of the graph in general. Often this reduction is quite significant (*q.v.* (Mathieson et al. 2004) for further examples), and allows the kernel of the problem to be solved quickly by a heuristic method, or if the reduction is sufficient, by complete enumeration.

The real importance of these reductions is that they not only significantly reduce the data, but that in doing so they preserve the optimality of the solution. This is an important distinction from many other data reduction techniques that cull or condense data, but do not guarantee that the optimal solution is still present in the reduced data set. This distinction is especially important when dealing with instances that derive from areas such as oncology and radiology, as destruction of the optimal solution could be lead to incorrect conclusions.

3 $(\alpha, \beta)k$-Feature Set as a Tool

A solution to the $(\alpha, \beta)k$-Feature Set problem for a given set of data gives us a set of features that are the minimal set needed to correctly classify all of the examples in the data set. Thus the ability to solve $(\alpha, \beta)k$-Feature Set for a given set of data gives us a powerful tool for revealing, in a large data set, what the underlying controlling factors are. This knowledge then allows us to both explore the forces in action in the system represented by the data, and to guide our predictions about new, unclassified examples.

Also by selecting different α and β values, we can explore results of varying confidence levels. The most notable approach here is to compare the results from choosing positive α, and either 0 or positive β. The first option gives a feature set that purely considers the smallest explanation for the existence of the separate classes within the data. The second gives a set of features that also explains something about why examples in the same class are actually in that class, not just why they are not in another class. Increasing the α and β values also gives robustness against error. These increased values force the redundant explanation of either why the two examples in the pair are in different classes or the same. Thus we can expect an explanation that offers multiple points on which to make a decision, so if a small number of these were in error, it would still be possible to categorise the example at hand on a majority basis. This is not infallible of course, a large number of errors in the data can still lead to incorrect classification, but this is true of all systems.

The reduction rules in themselves also provide an invaluable tool for analysing the data. As they only go so far as to indicate which features are definitely needed, and those that never will be (and those that are equivalent), the remaining kernel is open to solution by a technique of choice. This allows not only the comparison of different solution techniques, but due to the generally small size of the kernel, application of methods that guarantee optimality, such as complete enumeration, is often feasible. Thus we can often not only find one optimal solution, but all of them.

Another aspect that can be examined is the dominating power of a given feature set. For example, we may have several feature sets of size x, but a given feature set f may dominate more example pairs than the others (that is, the sum of the degrees of the features in the feature set is higher, naturally two features may thus dominate the same pair vertex). We may expect from this that perhaps f is a better feature set in some fashion, as it has features which are in some sense more applicable.

Thus we exploit the solution to the $(\alpha, \beta)k$-Feature Set problem in many ways. Most simply we undertake the procedure described in 2.1 and 2.2, and then apply some heuristic to the kernel to produce a feature set which we can then proceed to examine. We can also vary the α and β values to create smaller feature sets with lower robustness. The small kernel size also allows us to produce all the feature sets of a given size (minimal or otherwise), and compare the similarities.

We also use the WEKA software package[7]. WEKA offers several algorithms and heuristics for building rule sets and decision trees such as C4.5 and ID3. We have employed several techniques to allow examination of the stability and reliability of our results under different heuristics. We used the ID3, J48, PART and PRISM heuristics, but found that ID3 and J48 continually produced the same decision trees.

4 The Election Question

The problem that we apply our system to, presented in (Lichtman & Keilis-Borok 1981), is to use the answers to a set of yes/no questions to classify and subsequently predict the outcome of the popular vote in U.S. presidential elections. Lichtman and Keilis-Borok present twelve questions[8], using which they are able to 'predict' the outcome of the popular vote in all U.S. presidential elections up to 1980.

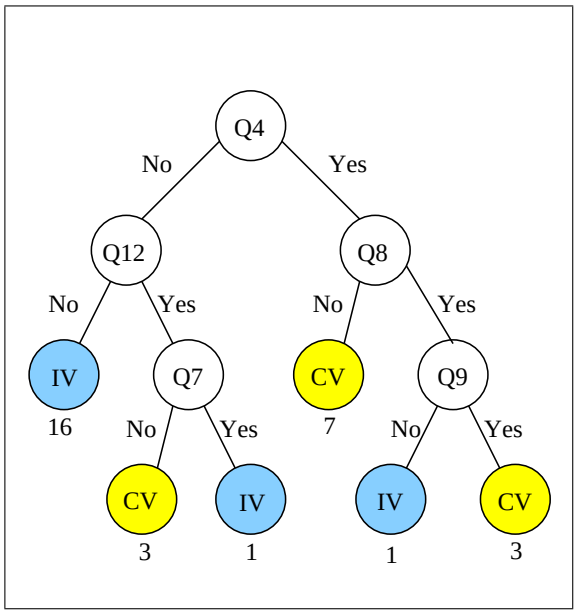

Figure 1: Decision tree for $\alpha = \beta = 2$. 'IV' indicates an incumbent victory, 'CV' a challenger victory. The numbers beneath each leaf indicate how many of the examples in the data set that they classify.

[7] http://www.cs.waikato.ac.nz/ml/weka/

[8] Reproduced here in Table 1, with answers in Table 2.

#	Question	
1	Has the incumbent party been in office more than a single term?	(no)
2	Did the incumbent party gain more then 50% of the vote cast in the previous election?	(yes)
3	Was there major third party activity during the election year?	(no)
4	Was there a serious contest for the nomination of the incumbent party candidate?	(no)
5	Was the incumbent party candidate the sitting president?	(yes)
6	Was the election year a time of recession or depression?	(no)
7	Was the yearly mean per capita rate of growth in real gross national product during the incumbent administration equal to or greater than the mean rate in the previous 8 years and equal or greater than 1%?	(yes)
8	Did the incumbent president initiate major changes in national policy?	(yes)
9	Was there major social unrest in the nation during the incumbent administration?	(no)
10	Was the incumbent administration tainted by a major scandal?	(no)
11	Is the incumbent party candidate charismatic or a national hero?	(yes)
12	Is the challenging party candidate charismatic or a national hero?	(no)

Table 1: The 12 questions presented by Lichtman and Keilis-Borok. The answers in parenthesis favor the incumbent party.

Feature	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Pairs Dominated
	0	1	1	1	1	1	1	1	0	0	1	1	2359
	0	1	1	1	1	0	1	1	1	0	1	1	2371
	0	1	1	1	0	1	1	1	0	1	1	1	2347
	0	1	1	1	0	1	1	1	1	0	1	1	2359
	1	1	1	1	0	1	1	1	0	0	1	1	2329
	1	1	1	1	0	0	1	1	1	0	1	1	2341
	1	0	1	1	0	1	1	1	1	0	1	1	2359
	0	0	1	1	0	1	1	1	1	1	1	1	2377
	1	0	1	1	0	0	1	1	1	1	1	1	2359
Appearances	4	6	9	9	2	6	9	9	6	3	9	9	
Popularity	0.44	0.67	1	1	0.22	0.67	1	1	0.67	0.33	1	1	
Weight	237	237	255	357	267	255	245	245	267	255	231	267	

Figure 2: The feature sets for $\alpha = \beta = 2$. The feature sets are laid out in rows, with a '1' indicating that the feature represented by that column is present, a '0' not present. 'Appearances' indicates the number of times a feature appears in total across all feature sets, 'Popularity' gives this as a ratio of the total number of feature sets, and 'Weight' indicates how many pairs the feature can dominate if it is in the feature set. The final column gives the number of pairs each feature set dominates, with the largest highlighted.

4.1 Our Results

Several experiments were conducted with the data from (Lichtman & Keilis-Borok 1981). Initial inspection of the graph indicated that the maximum possible α and β were 2 and 2. The graph, with beta vertices, consisted of 12 feature vertices corresponding to the 12 questions, and 465 pair vertices, 234 'alpha' vertices and 231 'beta' vertices. When $\beta = 0$ was considered, the graph contained only the 234 'alpha' vertices, as the 'beta' vertices can be immediately discarded.

When $\alpha = 2$ and $\beta = 2$, the reduction rules added 5 features to the feature set (Q3, Q7, Q8, Q11 and Q12), discarded no features, and reduced the number of pair vertices from 465 to only 13. The minimal feature set size for $\alpha = \beta = 2$ was nine features, with nine different feature sets possible[9] (see Figure 2). Notably, if $\alpha = \beta = 2$, Q3, Q7, Q8, Q11 and Q12 must be in the feature set, as they all are attached to pair vertices of degree 2 (and thus these pair vertices require those features to be dominated the requisite number of times). Interestingly Q4 was required, if we want to achieve the minimally sized feature set, but no reduction indicated this. Out of these nine feature sets, the one consisting of the six common features, plus features Q6, Q9 and Q10, dominated the greatest number of pair vertices (including overlaps), 2377 (see Figure 2). The next nearest feature set in these terms dominated 2371 pairs, and consisted of the six common features plus features Q2, Q5 and Q9. The decision tree for the first feature set (that which dominates 2377 pair vertices) can be seen in Figure 1 (developed with the J48 and ID3 heuristics, both giving the same answer). It is noted however, that at this point we do not have any algorithms or heuristics able to build decision trees or rules sets that take advantage of the redundancy available with $\alpha > 1$ and $\beta > 0$, thus the tree created for this feature set does not use all the features available to it. The sets of rules created by using the PART and PRISM heuristics are presented in Tables 3 and 4.

The feature sets for $\alpha = \beta = 1$ were also generated. In this case the reduction rules did not indicate that any features had to be in the kernel (unsurprisingly, as there were no degree 1 pair vertices), and did not indicate that any of the features were irrelevant. It did however reduce the number of pair vertices from 465 to 43, 30 'alpha' and 13 'beta'. From this kernel we determined that there are two minimal feature sets for $\alpha = \beta = 1$ for this data, (Q4, Q5, Q7, Q9, Q12) and (Q2, Q3, Q4, Q7, Q8). Of the two, the first dominated the most pair vertices, 1403 compared to 1339. The decision trees for these two feature sets are shown in Figures 4 and 5. Note that the decision tree for the feature set that dominates more pair vertices is more compact, suggesting perhaps that this extra domination indicates greater discriminatory power.

The classification rules generated by the PRISM heuristic are shown in Table 5, and the rules from the PART heuristic in Table 6.

Feature sets were also generated for $\alpha = 1$, $\beta = 0$. Of these there was 23, all of size 5, which between them used all of the features. The results of this can

[9] These feature sets were confirmed as the only 9 by complete enumeration.

Year	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Target
1864	0	0	0	0	1	0	0	1	1	0	0	0	1
1868	1	1	0	0	0	0	1	1	1	0	1	0	1
1872	1	1	0	0	1	0	1	0	0	0	1	0	1
1880	1	0	0	1	0	0	1	1	0	0	0	0	1
1888	0	0	0	0	1	0	0	0	0	0	0	0	1
1900	0	1	0	0	1	0	1	0	0	0	0	1	1
1904	1	1	0	0	1	0	1	0	0	0	1	0	1
1908	1	1	0	0	0	1	0	1	0	0	0	0	1
1916	0	0	0	0	1	0	0	1	0	0	0	0	1
1924	0	1	1	0	1	0	1	1	0	1	0	0	1
1928	1	1	0	0	0	0	1	0	0	0	0	0	1
1936	0	1	0	0	1	1	1	1	0	0	1	0	1
1940	1	1	0	0	1	1	1	1	0	0	1	0	1
1944	1	1	0	0	1	0	1	1	0	0	1	0	1
1948	1	1	1	0	1	0	0	1	0	0	0	0	1
1956	0	1	0	0	1	0	1	0	0	0	1	0	1
1964	0	0	0	0	1	0	1	0	0	0	0	0	1
1972	0	0	0	0	1	0	0	1	1	0	0	0	1
1860	1	0	1	1	0	0	1	0	1	0	0	0	0
1876	1	1	0	1	0	1	0	0	0	1	0	0	0
1884	1	0	0	1	0	0	1	0	1	0	1	0	0
1892	0	0	1	0	1	0	0	1	1	0	0	1	0
1896	0	0	0	1	0	1	0	1	1	0	1	0	0
1912	1	1	1	1	1	0	1	0	0	0	0	0	0
1920	1	0	0	1	0	1	0	1	1	0	0	0	0
1932	1	1	0	0	1	1	0	0	1	0	0	1	0
1952	1	0	0	1	0	0	0	0	0	1	0	1	0
1960	1	1	0	0	0	1	0	0	0	0	0	1	0
1968	1	1	1	1	0	0	1	1	1	0	0	0	0
1976	1	1	0	1	1	0	0	0	0	1	0	0	0
1980	0	0	1	1	1	1	1	0	0	1	0	1	0

Table 2: The data set presented by Lichtman & Keilis-Borok (1=yes, 0=no). The target column represents the winner of the popular vote (1=incumbent, 0=challenger).

be seen in Figure 3. This figure also includes some additional information about the feature sets as well. The decision tree for the feature set with the greatest dominating power (Q4,Q5,Q8,Q9,Q12), Figure 6, was generated, as were the rules from the PART (Table 8) and PRISM (Table 7) heuristics. In this case the reduction rules removed no features, and added none to the feature set, as in the case of $\alpha = \beta = 1$, but reduced the number of pair vertices from 234 to only 41.

5 Discussion

5.1 $(\alpha, \beta)k$-Feature Set in General

Firstly, it is clear that the application of $(\alpha, \beta)k$-Feature Set solvation tools can allow the simplification of the answers to problems of this kind. In this particular case we see that we only need five out of the twelve questions to correctly classify the examples present in the data set ($\alpha = 1$, $\beta = 0$). Of course we also see that there are many different combinations of five features that can achieve this (Figure 3). Even to cover[10] each example twice ($\alpha = \beta = 2$), and thus provide greater robustness against error, we need only nine of the twelve questions. The discovery of these core features and feature sets is greatly facilitated by the restructuring of the $(\alpha, \beta)k$-Feature Set problem as a Red/Blue Dominating Set problem, and the subsequent ability to apply powerful reduction rules. Keep in mind that in general both of these problems are formally hard, even in a parameterized

[10]Note that we use the word 'cover' lightly here. There may be no relation to what are commonly referred to as 'covering' problems.

Rule	Outcome
(Q4 = 1) & (Q8 = 0)	Challenger Victory
(Q4 = 1) & (Q9 = 1)	Challenger Victory
(Q12 = 1) & (Q7 = 0)	Challenger Victory
(Q4 = 0) & (Q12 = 0)	Incumbent Victory
(Q7 = 1) & (Q4 = 0)	Incumbent Victory
(Q8 = 1) & (Q9 = 0)	Incumbent Victory

Table 3: Classification rules generated by the PRISM heuristic for $\alpha = \beta = 2$. Here as well as in the decision tree, the potential robustness given by high α and β values is not exploited, thus not all features from the feature set are used.

Rule	Outcome
(Q4 = 1) & (Q8 = 0)	Challenger Victory (7.0)
(Q9 = 0) & (Q12 = 0)	Incumbent Victory (14.0)
(Q3 = 0) & (Q6 = 0)	Incumbent Victory (4.0)
Otherwise	Challenger Victory (6.0)

Table 4: Classification rules generated by the PART heuristic for $\alpha = \beta = 2$. These rules are used in a cascade fashion beginning with rule 1. The accompanying numbers indicate how many examples each rule classifies out of those left unclassified by the previous rules.

setting. That we are able to apply such simple rules, and yet consistently produce problem kernels that are feasible to completely enumerate, without loss of the optimality of the solution, demonstrates the power of this methodology. Even on such a small data set the reductions in the size of the problem afforded by this approach is appreciable, with the problem reduced by ∼80% or more. This allows the quick and easy application of almost any technique desired to uncover the complete answer, up to and including complete enumeration. Further, as this system provides good answers in a 'short'[11] amount of time, it can also be easily used as a preprocessing step for other techniques such as ANNs and the like.

The use of higher α and β values also seems to have an effect on the results regarding the decision trees and rules. Higher α and β seems to provide more room for strong, but not apparently necessary, features to be included in the minimal feature set, thus leading to more compact classification and prediction tools. This unfortunately is subject to the problem discussed next.

One of the problems with this technique at the moment is that we currently have no formal way of exploiting high α and β values. Any information regarding the robustness of the feature set, or any information about within-class similarity[12] is essentially discarded when we move to the application of techniques such as ID3 or PRISM, etc., for the generation of classification tools.

5.2 The Election Question and Our Results

Examination of our results for this data set shows some interesting characteristics of the data. We see

[11]'Short' of course being a rather relative term when NP-Hard problems are concerned. Here however, the time is a matter of milliseconds at most, and is most heavily influenced by the amount of screen output desired.

[12]This is also potentially an area that can be exploited in regards to automatic class generation.

Feature	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Pairs Dominated
	0	0	1	1	0	1	0	1	0	0	0	1	600
	0	0	1	0	0	1	0	1	1	0	0	1	550
	0	0	1	1	0	1	0	1	1	0	0	0	630
	0	1	1	1	0	1	0	1	0	0	0	0	627
	0	1	0	1	0	1	0	1	0	0	0	1	624
	0	1	0	1	0	1	0	0	1	0	0	1	620
	0	0	0	1	0	1	0	1	1	0	0	1	627
	1	1	0	1	0	0	0	0	1	0	0	1	626
	1	0	0	1	0	0	0	1	1	0	0	1	633
	0	0	0	1	1	0	0	1	1	0	0	1	648
	0	1	0	1	1	0	0	0	1	0	0	1	641
	0	0	0	1	1	0	0	0	1	1	0	1	598
	0	0	0	1	1	1	0	0	1	0	0	1	632
	0	0	0	1	1	0	1	0	1	0	0	1	647
	0	1	1	0	1	0	1	1	0	0	0	0	601
	0	1	1	1	0	0	1	1	0	0	0	0	642
	1	0	1	1	0	0	1	1	0	0	0	0	639
	0	0	1	1	0	0	1	1	0	0	0	1	615
	0	0	1	1	0	0	1	0	0	0	1	1	587
	0	0	1	1	0	0	1	0	1	0	0	1	611
	0	0	0	1	0	0	1	1	1	0	0	1	642
	0	1	0	1	0	0	1	0	1	0	0	1	635
	0	1	0	1	0	0	1	1	0	0	0	1	639
Appearances	3	9	10	21	6	8	10	14	14	1	1	18	
Popularity	0.130435	0.391304	0.434783	0.913043	0.26087	0.347826	0.434783	0.608696	0.608696	0.043478	0.043478	0.782609	
Weight	117	120	96	173	132	111	126	127	123	77	99	93	

Figure 3: All 23 feature sets for $\alpha = 1, \beta = 0$. The feature sets are laid out in rows, with a '1' indicating that the feature represented by that column is present, a '0' not present. 'Appearances' indicates the number of times a feature appears in total throughout the 23 feature sets, 'Popularity' gives this as a ratio of the total number of feature sets, and 'Weight' indicates how many pairs the feature can dominate if it is in the feature set. The final column gives the number of pairs each feature set dominates, with the largest highlighted.

several features repeatedly being used in most feature sets, indicating that they are probably the most important (and at least have high discriminatory power). The most obvious of these is Q4 (Was there a serious contest for the nomination of the incumbent party candidate?), which occurs in all but two of the feature sets, and in roughly half of the rules generated by the PART and PRISM heuristics. Similarly it appears that Q12 (Is the challenging party candidate charismatic or a national hero?) is a highly important feature, and thus factor in the choice of vote winners. Q7 (The short version being 'Was the economy strong under the incumbent administration?') is also one of the more important features, and appears regularly in the feature sets and decision trees and rules.

Comparing feature sets for fixed values of α and β (Figures 3 & 2), there seems to be no obviously significant trends other than those mentioned above. Some pairs of features seem to be interchangeable once the 'important' features (Q4 and Q12) are present, such as Q8 & Q9 for $\alpha = 1$, $\beta = 0$, where if one is not present, the other almost certainly is, though obviously this is not a rule. Other features seem to have little import at all, especially Q10 (Was the incumbent administration tainted by a major scandal?), which is almost never used[13]. Interestingly for $\alpha = 1$, $\beta = 0$, Q11 (Is the incumbent party candidate charismatic or a national hero?), the complementary feature of Q12, is almost never used. However when we ask for $\alpha = \beta = 2$, it is vital. It seems reasonable to suggest that $\alpha = \beta = 2$ indicates the more subtle interactions present in the data, that only appear when more complex feature sets are considered.

5.3 Our Prediction

Beginning with the decision trees, Figures 1, 4, 5, and 6, we choose the answer to Q4 to be 'no', which we think is a reasonable and obvious answer. If the answer to Q4 were to be 'yes' however, the challenger would be at significant advantage, with 10 out of 11 possible outcomes favouring him in three out of the four trees. There would also be significantly more rules that would be potentially applicable. Based on this it seems that instability in the incumbent party is one of the most significant factors.

From there, we consider the answer to Q12 to be 'yes'[14], although we uncertain of this, being somewhat outside the system. It is interesting to consider that currently (at the time of writing), the Republican election effort seems to be directed towards reversing this opinion (i.e. making the answer to Q12 'no'). If they are successful at this, then three out of four of our decision trees (the fourth doesn't consider Q12) indicate an incumbent victory. Notably if Q12 were 'no', the path in the tree leading to this decision is much shorter, and classifies 16 out of the 31 examples, suggesting that U.S. elections rely largely on a stable incumbent administration, and the discrediting of the challenger. This suggests that the current Republican tactic is a wise and time honoured one, and that perhaps they are aware of this.

We believe the answer to Q7 to also be 'yes'. From our basic research the U.S. economic growth seems to be strong but it appears to have weakened at least in the short term, though the growth rate is still above that specified in Q7. If the answer to Q7 is in fact 'no', then three of the decision trees indicate that the challenging party will win, making the Republican's attack on John Kerry's persona even more relevant, as a change in Q12 would then change the outcome of the vote.

From these three answers we have the result 'Incumbent Victory' for three out of four decision trees. For the last we must also answer Q5 and Q9. Addressing Q9 first, we consider that there has been no major social unrest in the U.S., though again we are far from experts, and find the question to be ambigu-

[13] The implications of this we leave to the reader.

[14] At the time of writing the current challenging candidate was John Kerry, who was decorated 5 times, including 3 Purple Hearts, in the Vietnam conflict, and also seems to be reasonably charismatic.

Rule	Outcome
(Q4 = 1) & (Q8 = 0)	Challenger Victory
(Q4 = 1) & (Q7 = 0)	Challenger Victory
(Q3 = 1) & (Q2 = 0)	Challenger Victory
(Q4 = 1) & (Q2 = 1)	Challenger Victory
(Q7 = 0) & (Q8 = 0) & (Q2 = 1)	Challenger Victory
(Q4 = 0) & (Q7 = 1)	Incumbent Victory
(Q4 = 0) & (Q8 = 1) & (Q3 = 0)	Incumbent Victory
(Q4 = 0) & (Q2 = 0) & (Q3 = 0)	Incumbent Victory
(Q8 = 1) & (Q2 = 1) & (Q4 = 0)	Incumbent Victory
(Q8 = 1) & (Q7 = 1) & (Q2 = 0)	Incumbent Victory

Table 5: Classification rules generated by the PRISM heuristic for $\alpha = \beta = 1$ from the feature set (Q2,Q3,Q4,Q7,Q8).

Rule	Outcome
(Q4 = 1) & (Q8 = 0)	Challenger Victory (7.0)
(Q4 = 0) & (Q7 = 1)	Incumbent Victory (11.0)
(Q4 = 1) & (Q7 = 0)	Challenger Victory (2.0)
(Q2 = 0) & (Q3 = 0)	Incumbent Victory (5.0)
(Q8 = 0)	Challenger Victory (2.0)
(Q2 = 1) & (Q4 = 0)	Incumbent Victory (2.0)
Otherwise	Challenger Victory (2.0)

Table 6: Classification rules generated by the PART heuristic for $\alpha = \beta = 1$. These rules are used in a cascade fashion beginning with rule 1. The accompanying numbers indicate how many examples each rule classifies out of those left unclassified by the previous rules.

Rule	Outcome
(Q4 = 1) & (Q8 = 0)	Challenger Victory
(Q4 = 1) & (Q9 = 1)	Challenger Victory
(Q12 = 1) & (Q9 = 1)	Challenger Victory
(Q12 = 1) & (Q5 = 0)	Challenger Victory
(Q4 = 0) & (Q12 = 0)	Incumbent Victory
(Q9 = 0) & (Q8 = 1)	Incumbent Victory
(Q4 = 0) & (Q9 = 0) & (Q5 = 1)	Incumbent Victory

Table 7: Classification rules generated by the PRISM heuristic for $\alpha = 1$, $\beta = 0$ from the feature set (Q4,Q5,Q8,Q9,Q12).

Rule	Outcome
(Q4 = 1) & (Q8 = 0)	Challenger Victory (7.0)
(Q9 = 0) & (Q12 = 0)	Incumbent Victory (14.0)
(Q4 = 0) & (Q12 = 0)	Incumbent Victory (3.0)
(Q9 = 1)	Challenger Victory (5.0)
(Q5 = 0)	Challenger Victory (1.0)
Otherwise	Incumbent Victory (1.0)

Table 8: Classification rules generated by the PART heuristic for the $\alpha = 1$, $\beta = 0$ feature set (Q4,Q5,Q8,Q9,Q12). These rules are used in a cascade fashion beginning with rule 1. The accompanying numbers indicate how many examples each rule classifies out of those left unclassified by the previous rules.

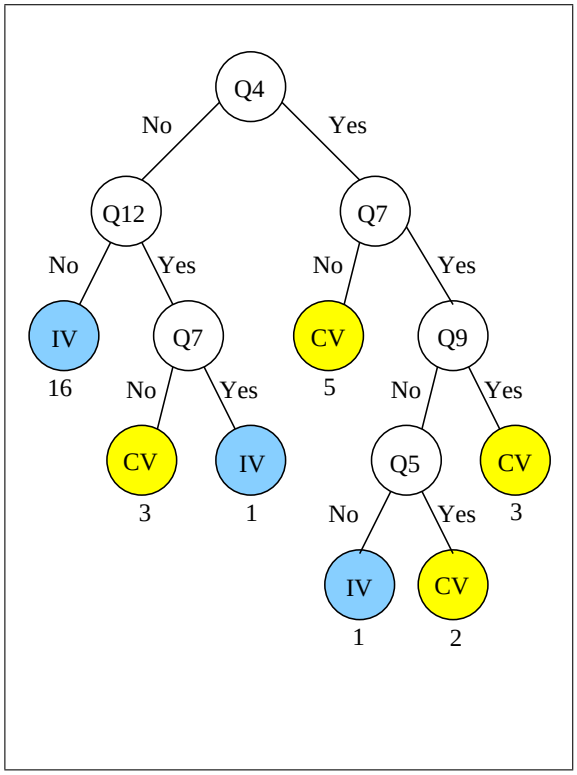

Figure 4: Decision tree for $\alpha = \beta = 1$. 'IV' indicates an incumbent victory, 'CV' a challenger victory. The numbers beneath each leaf indicate how many of the examples in the data set that they classify.

ous. Although the U.S. is currently involved in an overseas conflict, and some protests have occurred, it does not seem to be an unusual position for the nation, and does not constitute *major* unrest *within* the nation. The answer to Q5 is currently 'yes' (George W. Bush), thus indicating a victory for the incumbent party. If the answer to Q9 were 'yes', a 'Challenger Victory' would be the result, which seems a reasonable likelihood if there were significant unrest which would indicate unhappiness with the incumbent administration.

Turning then to the rules generated using the PRISM heuristic, based on the above answers to the questions we get 'Incumbent Victory' from Tables 3 (rule 5) and 5 (rule 6). If we consider the answer to Q8 (Did the incumbent president initiate major changes in national policy?) to be 'yes', which seems reasonable if we consider the USA PATRIOT act[15] and so on, we also get 'Incumbent Victory' from Table 7 (rule 6). The rules cascades generated using the PART heuristic also indicate an 'Incumbent Victory', using the answers to the questions as above except in the case of $\alpha = \beta = 2$, where we also require the answers to Q3 and Q6 (both of which we believe to be 'no'), but still get the same result.

From these results we consider that a Republican victory (in the popular vote) is most likely in this coming election. Unfortunately of course we are no more qualified than any other non historian or political scientist to judge the validity of these answers. However Dr. Lichtman and Dr. Keilis-Borok, authors of the original 1981 paper (Lichtman & Keilis-Borok 1981) that this work is based on, and authors

[15] Available from: http://www.fincen.gov/pa_main.html

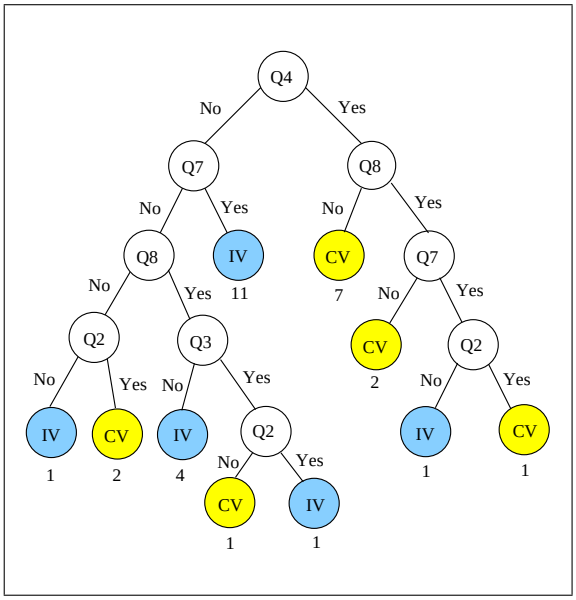

Figure 5: Alternate decision tree for $\alpha = \beta = 1$ based on a different optimal feature set. 'IV' indicates an incumbent victory, 'CV' a challenger victory. The numbers beneath each leaf indicate how many of the examples in the data set that they classify.

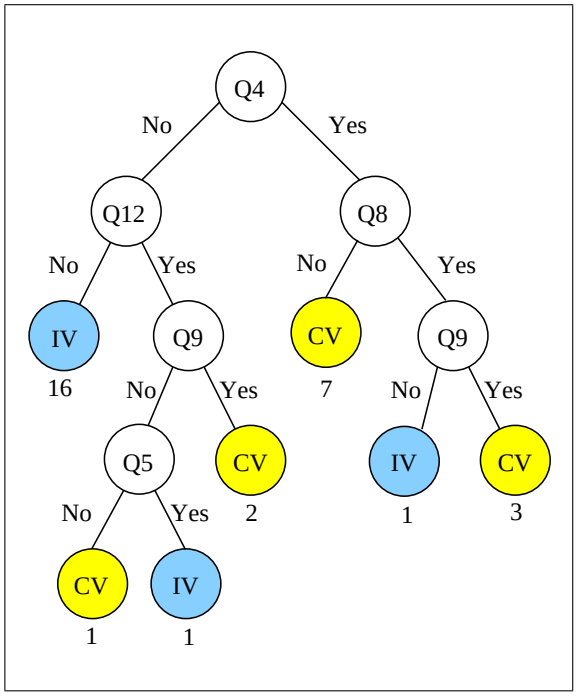

Figure 6: Decision Tree for the feature set (Q4,Q5,Q8,Q9,Q12), with $\alpha = 1$ and $\beta = 0$. The numbers below each leaf indicate how many examples the leaf classifies.'IV' indicates an incumbent victory, 'CV' a challenger victory.

of several subsequent related articles and developments, have also predicted that this coming election will see an incumbent victory[16]. Dr. Fair at Yale has produced his own method of predicting the share of the votes the incumbent administration will receive (Fair 1978), and concurs with our and Dr. Lichtman & Dr. Keilis-Borok's prediction of a Republican victory. His model is based however on entirely economic factors, which seems to be a popular and traditional prediction method. Forbes magazine suggests however that the state of the economy is not as strong a factor as traditionally believed (Ackman & Hazlin 2004), with only 64% of elections being predictably by economic factors alone, and even then Forbes uses a significantly more complex economic model than is usually proposed, with seven interdependent variables.

6 Conclusion

We presented in this paper a deterministic and optimality preserving method of reductions to allow the solutions to a series of problems related to the k-Feature Set problem to be found. We also presented a small test data set and the application of this system and other techniques for the use in analysis, classification and prediction with regards to the system the data represents. We believe this method is both flexible and powerful, and has clear applications in all field where data mining techniques are used.

Further research may include the development or modification of current methods for generating decision trees and classifying rule sets to allow the exploitation of the extra power and information offered by the $(\alpha, \beta)k$-Feature Set variant of the problem. The use of these ideas on unclassified data is also an interesting area of potential research, using the maximisation of the α and β values to indicate which classes example should belong to. At least for the extra information provided by increased α values we can envision a form of decision that allows multiple features at each decision point. For example, if we (manually) construct the decision tree for $\alpha = 2$, we arrive at a tree with (Q4∨Q12) at the root, with the 'no' branch corresponding to Q4 and Q12 as 'no', and classifying 16 of the 'Incumbent Victory' examples (essentially a contraction of the left hand branch of the exhibited trees). From the 'yes' branch we can then insert a decision vertex labelled (¬Q3∨¬Q7∨Q9), which on the 'no' branch classifies the final two 'Incumbent Victories', leaving all the 'Challenger Victories' down the 'yes' branch of this second decision. This kind of tree is obviously more useful for solutions to $(\alpha, \beta)k$-Feature Set with $\alpha > 1$. It remains to generalise and automate this process. It also remains to incorporate the information giving by increased β values.

With regard to the actual data used, we predict that it is most probable that George W. Bush will serve a second term as President of the United States of America, if there are no dramatic changes in the candidates or the knowledge of the public.

6.1 Acknowledgements

P. Moscato would like to thank Dr. Keilis-Borok for a discussion they had in December 1991 in Trieste while both were at the International Centre for Theoretical Physics.

References

D. Ackman & M. Hazlin, It's not the economy, stupid, *Forbes*, http://moneycentral.msn.com/content/invest/forbes/P92882.asp?GT1=4529, (2004)

[16] http://www.counterpunch.org/lichtman07292004.html
http://www.signonsandiego.com/uniontrib/20040509/news_1n9predict.html
http://hnn.us/articles/6599.html

C. Cotta & P. Moscato, The k-Feature Set Problem is W[2]-Complete, *Journal of Computer and System Sciences*, **67**(4), (2003), pages 686-690

C. Cotta, C. Sloper & P. Moscato, Evolutionary Search of Thresholds for Robust Feature Set Selection: Application to the Analysis of Microarray Data, *Proceedings of EvoBio2004 - 2nd European Workshop on Evolutionary Computation and Bioinformatics*, Coimbra, Portugal, April, (2004)

R. Downey & M. Fellows, Parameterized Complexity, *Springer*, (1997)

S. Davies & S. Russell, NP-Completeness of Searches for Smallest Possible Feature Sets, *Proceedings of the AAAI Symposium on Relevence*, (1994), pages 41-43

R. Fair, The Effect of Economic Events on Votes for President, *The Review of Economics and Statistics*, **May**, (1978), pages 159-173
`http://fairmodel.econ.yale.edu/vote2004/index2.htm`

A. J. Lichtman and V. I. Keilis-Borok, Pattern Recognition Applied to Presidential Elections in the United States, 1860-1980: Role of Integral Social, Economic, and Political Traits, *Proceedings of the National Academy of Sciences of the United States of America*, **78**(11), (1981), pages 7230-7234

L. Mathieson, A. Mendes, J. Marsden, J. Pond and P. Moscato, Computer Aided Breast Cancer Diagnosis with Optimal Feature Sets: Reduction Rules and Optimization Techniques, *Manuscript in Preparation*, (2004)

K.Weihe, Covering Trains by Stations or the Power of Data Reduction, *OnLine Proceedings of ALEX'98 - 1st Workshop on Algorithms and Experiments*, `http://rtm.science.unitn.it/alex98/proceedings.html`, (1998)