LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-CONF-202657

Feature Selection in Scientific Applications

E. Cantu-Paz, S. Newsam, C. Kamath

March 2, 2004

International Conference on Knowledge Discovery and Data Mining
Seattle, WA, United States
August 22, 2004 through August 25, 2004

Feature Selection in Scientific Applications

Erick Cantú-Paz
Lawrence Livermore Natl. Lab.
7000 East Avenue, L-561
Livermore, CA 94550
cantupaz@llnl.gov

Shawn Newsam
Lawrence Livermore Natl. Lab.
7000 East Avenue, L-551
Livermore, CA 94550
newsam1@llnl.gov

Chandrika Kamath
Lawrence Livermore Natl. Lab.
7000 East Avenue, L-561
Livermore, CA 94550
kamath2@llnl.gov

ABSTRACT

Numerous applications of data mining to scientific data involve the induction of a classification model. In many cases, the collection of data is not performed with this task in mind, and therefore, the data might contain irrelevant or redundant features that affect negatively the accuracy of the induction algorithms. The size and dimensionality of typical scientific data make it difficult to use any available domain information to identify features that discriminate between the classes of interest. Similarly, exploratory data analysis techniques have limitations on the amount and dimensionality of the data that can be effectively processed. In this paper, we describe applications of efficient feature selection methods to data sets from astronomy, plasma physics, and remote sensing. We use variations of recently proposed filter methods as well as traditional wrapper approaches where practical. We discuss the importance of these applications, the general challenges of feature selection in scientific datasets, the strategies for success that were common among our diverse applications, and the lessons learned in solving these problems.

1. INTRODUCTION

Scientific data sets generated by computer simulations, observations, or experiments present challenges that are not usually present in commercial data mining. For instance, many scientific applications of data mining require the extraction of features from low-level data, such as images or mesh data from computer simulations. The data can be noisy, especially data coming from experiments or sensors, and removing the noise without affecting the signal is difficult. Also in contrast with commercial data, assigning labels to scientific data usually requires a domain scientist to identify examples of the objects of interest. Besides being tedious, this subjective process is prone to errors and experts often disagree on the labeling. Another difficulty is that scientific data is sometimes obtained from different sources and is captured at different resolutions or with different instruments, so data fusion becomes necessary to incorporate all the data into the analyses.

In this paper we are concerned with the problem of feature selection. This problem has its origins in some of the difficulties mentioned above. In particular, there are many methods to extract features from low-level image or simulation data. One could use simple statistics of the variables of interest or more sophisticated features that describe shapes or textures. It is likely that the collection of data was not performed with a particular analysis in mind. Therefore, the data may contain irrelevant or redundant features that affect the analysis negatively. Domain information is very helpful in pruning the data and to identify candidate variables, but in many cases the size and dimensionality of typical scientific data make it difficult to use any available domain information to identify features that discriminate between the classes of interest.

Regardless of these difficulties, data mining is gaining acceptance in many scientific fields. This paper describes applications of feature selection in three very different scientific data sets. The number of features in these applications vary from a few tens to a few hundreds. We describe the challenges common to these applications, the strategies we followed to face these challenges, and the general lessons we learned in solving these problems.

The first application is a classical classification problem where the goal is to build a predictor that will identify galaxies of a particular type. The second application is to identify variables that might explain the presence of a desirable harmonic oscillation on the edge of the plasma in fusion experiments. The third problem we present is to identify human settlements in satellite imagery. We explain the problems in more detail in later sections.

The next section describes the feature selection algorithms that we use. Section 3 describes the astronomical data, how it was generated with input from the astronomers, why we assumed that feature selection was necessary, and finally we present the results using different automatic methods as well as further reductions performed manually. Section 4 presents the problem with the fusion data, the approaches we followed and the results. Section 5 discusses the data used to detect human settlements in satellite imagery. These data contain the highest number of features of the three problems considered in the paper, and an effective feature selection could save considerable computing resources used

in creating and storing these features. Finally, section 6 summarizes the approaches common to the diverse applications, the lessons learned in applying the feature selection methods, and the conclusions of this paper.

2. FEATURE SELECTION ALGORITHMS

The feature selection problem has received considerable attention from machine learning and statistics and numerous feature selection algorithms have been proposed. Kohavi and John [10] classify feature selection algorithms as wrapper and filter methods. Wrappers treat an induction algorithm as a black box that is used to evaluate each candidate feature subset. While usually giving good results in terms of the accuracy of the final classifier, wrapper approaches are computationally expensive and are impractical in many scientific applications. Filter methods are independent of the classifier and select features based on properties that good feature sets are presumed to have, such as class separability or high correlation with the target. Filter methods are computationally efficient, but may produce disappointing results, because they ignore completely the induction algorithm.

We use four variable ranking filters, two classical wrapper methods, and one hybrid filter-wrapper method. We use the filters and the filter-wrapper hybrid to rank the features. We evaluate the rankings by training a naive Bayes classifier on increasingly larger subsets of the ranked features and report the 10-fold crossvalidation estimate of the prediction error. We also tried a decision tree classifier, but only report results when they are better than the naive Bayes classifier.

The data may contain features that are irrelevant to the classification. To detect these features in the rankings, we introduce into each dataset a "sentinel" random noise feature. This feature is uniformly distributed in the interval [0,1], and serves as in indicator to discard those variables that ranked lower than the sentinel.

2.1 KL Class Separability Filter

The first filter that we consider in this paper uses a natural measure of how well a feature separates the data into different classes. The filter calculates the class separability of each feature using the Kullback-Leibler (KL) distance between histograms of feature values. For each feature, there is one histogram for each class. Numeric features are discretized using $\sqrt{|D|}/2$ equally-spaced bins, where $|D|$ is the size of the training data. The histograms are normalized dividing each bin count by the total number of elements to estimate the probability that the j-th feature takes a value in the i-th bin of the histogram given a class n, $p_j(d = i|c = n)$. For each feature j, we calculate the class separability as

$$\Delta_j = \sum_{m=1}^{c} \sum_{n=1}^{c} \delta_j(m, n), \tag{1}$$

c is the number of classes and $\delta_j(m, n)$ is the KL distance between histograms corresponding to classes m and n:

$$\delta_j(m, n) = \sum_{i=1}^{b} p_j(d = i|c = m) \log \left(\frac{p_j(d = i|c = m)}{p_j(d = i|c = n)} \right), \tag{2}$$

where b is the number of bins in the histograms. Of course, other metrics such as the Bhattacharya distance could be used instead of KL distance.

The features are ranked simply by sorting them in descending order of the distances Δ_j (larger distances mean better separability).

2.2 Chi-Square Filter

This filter computes Chi-square statistics from contingency tables for every feature. The contingency tables have one row for every class label and the columns correspond to possible values of the feature (see table 1, adapted from [9]). Numeric features are represented by histograms, so the columns of the contingency table are the histogram bins.

Table 1: A 2×3 contingency table of a fictitious attribute A1 with observed and expected frequencies. Expected frequencies are in parenthesis.

Class	A1=1	A1=2	A1=3	Total
0	31 (22.5)	20 (21)	11 (18.5)	62
1	14 (22.5)	22 (21)	26 (18.5)	62
Total	45	42	37	124

The Chi-square statistic for feature j is

$$\chi_j^2 = \sum_i \frac{(o_i - e_i)^2}{e_i},$$

where the sum is over all the cells in the $r \times c$ contingency table, where r is the number of rows and c is the number of columns; o_i stands for the observed value (the count of the items corresponding to the cell i in the contingency table); and e_i is the expected frequency of items calculated as:

$$e_i = \frac{\text{(column total)} \times \text{(row total)}}{\text{grand total}}$$

The variables are ranked by sorting them in descending order of their χ^2 statistics.

2.3 Stump Filter

A stump is a decision tree that makes exactly one decision (i.e., it is a simple if-then-else rule on one variable). Decision trees split the data by examining each feature at a time and finding the split that optimizes an impurity measure. To search for the optimal split of a numeric feature x, the feature is sorted ($x_1 < x_2 < ... < x_n$) and all intermediate values $(x_i + x_{i+1})/2$ are evaluated as possible splits using a given impurity measure. The optimal impurity of each feature is recorded, and the features are ranked according to their optimal impurities.

Many measures of the impurity of a split have been proposed, such as the information gain [15], the Gini index, or the twoing rule [2]. In this paper we use the Gini index, which is defined as:

$$\sum_b \frac{n_b}{n} \left(1 - \sum_c \left(\frac{n_{bc}}{n_b} \right)^2 \right)$$

where n is the total number of instances, n_b is the number of instances in branch b, and n_{bc} is the number of instances of class c in branch b.

Essentially this filter performs one step in the construction of a decision tree. Besides being efficient, it provides information on the optimal thresholds on the values of the features, which may be of interest to the scientists.

2.4 PCA Filter

Principal component analysis (PCA) produces mutually orthogonal linear combinations of the variables in the original data, such that the direction of the first principal component (PC) corresponds to the direction of maximum variance in the data, the direction of the second PC corresponds to the direction of second largest variance in the data, and so on. The data are standardized to have mean zero and variance one before computing the PCs to avoid the dominance of variables with large variances. The principal components are the eigenvectors of the data covariance matrix, with the first PC being the eigenvector corresponding to the largest eigenvalue. In many cases, the first few PCs explain most of the variability in the data and therefore provide a compact representation of the important features in the data.

We adopted a method suggested by Mardia et al. [13] to use the PCs to eliminate unimportant variables. Starting with the eigenvector that corresponds to the smallest eigenvalue of the covariance matrix, we discarded the variable with the largest coefficient (in absolute value) in that vector. This variable is considered the least important. We then proceed to the eigenvector that corresponds to the second largest eigenvalue and discarded the variable with the largest coefficient, among the variables not discarded earlier. We continued with this process until we had ranked all the variables.

2.5 Sequential Forward Selection and Backward Elimination

Sequential forward selection (SFS) and sequential backward elimination (SBE) are two classic greedy wrappers. Forward selection starts with an empty set of features. In the first iteration, the algorithm considers all feature subsets with only one feature. The feature subset with the highest accuracy is used as the basis for the next iteration. In each iteration, the algorithm tentatively adds to the basis each feature not previously selected and retains the feature subset that results in the highest estimated performance. The search terminates after the accuracy of the current subset cannot be improved by adding any other feature.

Backward elimination works in an analogous way, starting from the full set of features and tentatively deleting each feature not deleted previously.

In these algorithms, each feature subset is evaluated by estimating the accuracy of a classification algorithm using the candidate subset of features. In this paper, we use 10-fold crossvalidation to estimate the accuracy.

2.6 Boosting Filter-Wrapper Hybrid

This algorithm is a generalization of Das' filter-wrapper hybrid algorithm [4]. The algorithm starts with an empty feature set and in each iteration greedily adds one feature, so it is similar to SFS. The differences are in the way the feature is selected and in how the algorithm terminates.

Using a (user-defined) filter, the algorithm ranks the features that have not been selected so far and adds the highest-ranking feature to the feature subset. Then, a classifier is trained using the current subset and it is used to classify the instances in the training set. The weights of the instances are updated using the normal Ada Boost procedure (giving more weight to instances misclassified by the classifier) and the algorithm iterates.

The classifier used to re-weight the instances can be trained on all the features selected so far, or only on the newly selected feature. Training on only one feature means that only simple uni-variate classifiers can be used. Das presents versions of this algorithm that re-weight the instances in the training set using boosted decision stumps as well as decision trees trained on the unweighted training set using all the features selected so far.

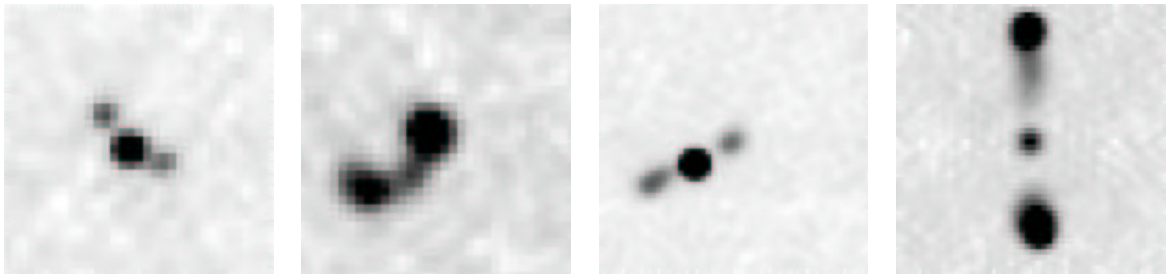
The algorithm can be stopped after an arbitrary number of iterations or when the performance of a classifier trained with all the selected features does not improve from the previous iteration. Das argued that using the accuracy on the training set was adequate for stopping the algorithm. We performed experiments using crossvalidation estimates of the accuracy, but confirmed that the results were not different. Note that the classifier used to stop the algorithm is not necessarily of the same type as the one used to re-weight the training set.

The key idea of this algorithm is that the filter that ranks the features in each iteration is using the boosted weights. In this way, the filter is asked to identify the feature that best discriminates the instances that are hard to classify using the features selected previously.

For the experiments in this paper, we re-weight the training set using a naive Bayes trained with the unweighted training set using all the features selected so far. We also use a naive Bayes to stop the algorithm. Preliminary tests did not show large differences in the error rates of the final classifiers when stumps were used to re-weight instances and trees were used to stop the algorithm.

3. FIRST ASTRONOMICAL SURVEY

The first data set that we examine in this paper comes from the Faint Images of the Radio Sky at Twenty-cm (FIRST) survey [1]. This survey started in 1993 with the goal of producing the radio equivalent of the Palomar Observatory Sky Survey. Using the Very Large Array at the National Radio Astronomy Observatory, FIRST is scheduled to cover more than 10,000 square degrees of the northern and southern galactic caps. At present, FIRST has covered about 8,000 square degrees, producing more than 32,000 two-million pixel images. At a threshold of 1 mJy, there are approximately 90 radio-emitting galaxies, or radio sources, in a typical square degree.



(a) (b) (c) (d)

Figure 1: Example radio sources: (a)-(b) Bent-doubles, (c)-(d) Non-bent doubles.

Radio sources exhibit a wide range of morphological types that provide clues to the source's class, emission mechanism, and properties of the surrounding medium. Sources with a bent-double morphology are of particular interest as they indicate the presence of clusters of galaxies, a key project within the FIRST survey. FIRST scientists currently identify the bent-double galaxies by visual inspection, which—besides being subjective, prone to error and tedious—is becoming increasingly infeasible as the survey grows.

Figure 1 has several examples of radio sources from the FIRST survey. Note the similarity between the bent-double in example (a) and the non-bent-double in example (c).

Data from FIRST are available on the FIRST web site (sundog.stsci.edu). There are two forms of data available: image maps and a catalog. The images in figure 1 are close-ups of galaxies. The catalog [17] was obtained by the astronomers by fitting two-dimensional Gaussians to each radio source on an image map. Each entry in the catalog corresponds to a single Gaussian. The catalog entries include information such as the right ascension (RA, analogous to longitude) and declination (Dec, analogous to latitude) for the center of the Gaussian, the major and minor axes of the ellipse, the peak flux, and the position angle of the major axis (degrees counterclockwise from North). Note that we differentiate between catalog entries and radio sources, with a radio source being composed of one or more catalog entries. In this paper, we present results using radio sources composed of three catalog entries based on the 2000 version of the catalog.

We decided that, initially, we would identify the radio sources and extract the features using only the catalog. The astronomers expected that the catalog was a good approximation to all but the most complex of radio sources, and several of the features they thought were important in identifying bent-doubles were easily calculated from the catalog.

We identified candidate features for the bent-double problem through extensive conversations with FIRST astronomers. When they justified their decisions of identifying a radio source as a bent-double, they placed great importance on spatial features such as distances and angles. Frequently, the astronomers would characterize a bent-double as a radio-emitting "core" with one or more additional components at various angles, which were usually wakes left by the core as it moved relative to the Earth. We focused on features that were scale, rotation and translation invariant. In total, we extracted 99 non-housekeeping features. A full list is described elsewhere [5].

Our training set is relatively small, containing 195 examples for the galaxies described by three entries in the catalog. Since the bent- and non-bent-doubles must be manually labeled by FIRST scientists, putting together an adequate training set is non-trivial. Moreover, scientists are usually subjective in their labeling of galaxies, and the astronomers often disagree in the hard-to-classify cases. There is also no ground truth we can use to verify our results. These issues imply that the training set itself is not very accurate, and there is a limit to the accuracy we can obtain.

We have analyzed these data with different techniques. Fodor and Kamath [6] used exploratory data analysis and the PCA filter explained in the previous section to select relevant features. Before the exploratory data analysis, Fodor and Kamath pruned the feature set by eliminating features that depend on the scale or were sensitive to small changes in the data. However, even with extensive domain knowledge, reducing the number of features in this problem was problematic. For example, in consultation with the astronomers, we generated three different measures of symmetry and three measures of "bentness." These measures are clearly correlated, but it is not obvious which one(s) should be preferred to induce a classifier. After eliminating as many features as possible, box-plots were used to identify features that discriminate between bent and non-bent double galaxies.

In a different study, Cantú-Paz and Kamath [3] applied several combinations of evolutionary algorithms and neural networks. All the combinations they tried resulted in classifiers with similar accuracy, except when they used an evolutionary algorithm for feature selection. This case resulted in significantly higher accuracies.

Figure 2 presents 10-fold crossvalidation estimates of the error rates of a naive Bayes using increasingly large feature subsets, as explained in the previous section. The figure shows that the PCA filter and the stump filter find small feature subsets that result in the lowest accuracy (both reach 12.1%). Interestingly, as we test larger feature subsets identified by the PCA ranking, the error of the classifier becomes the worst. The error of naive Bayes using all the features

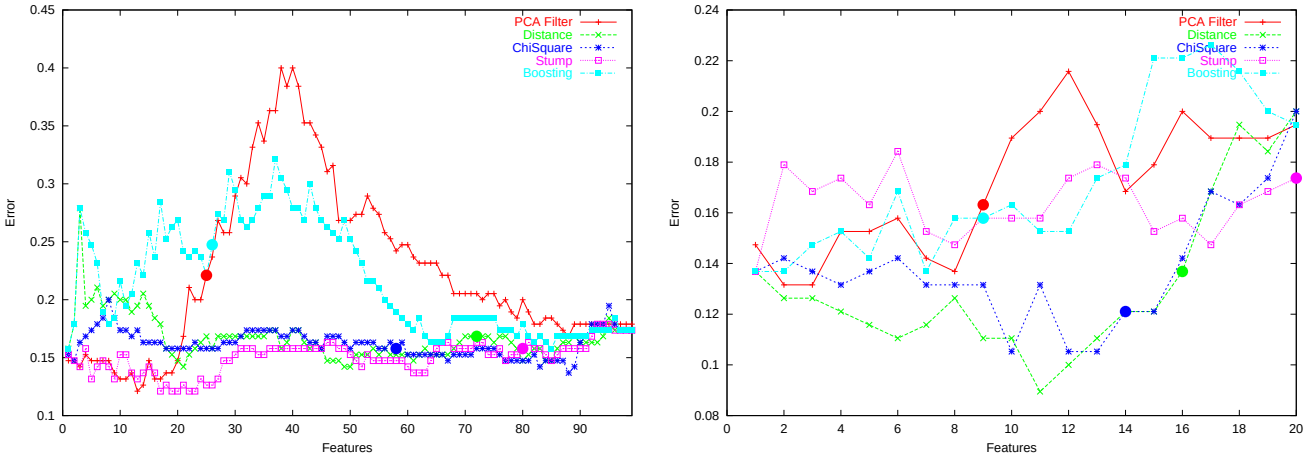


Figure 2: Error rates varying the number of features using FIRST data. The left graph shows results considering all 99 features and the right graph shows results considering only the 20 triplet features. The large dots represent the ranking of the random noise feature.

is 17.3%, so the observed improvements are significant (according to a two-sided t-test at 0.05 level of significance).

This data is sufficiently small that SFS and SBE wrappers are practical. SFS found a subset with four features that resulted in an error rate of 12.63%, not significantly different from the best result of the ranking methods. On the other hand, SBE did not manage to eliminate any variables from this data. Of the four features that SFS found, one is the total area of the three Gaussians that represent the galaxy. This feature is irrelevant to the identification of bent-double galaxies (because the area depends on the proximity of the galaxy to the Earth, not of any intrinsic property of the galaxy) and can be eliminated.

Our previous experience with this data suggested that the best accuracies are usually achieved using features extracted considering triplets of catalog entries (as opposed to pairs or single entries). There are only 20 of these features, and the results are presented in the right panel of figure 2. In this case the Chi-square and the KL distance filter found subsets of 10 and 11 features that resulted in the lowest errors (8.9 and 10.5%, respectively). These errors are significantly different from the best results obtained using all the features. SBE found a subset of three features (but again including the total area), that resulted in an error of 10%. SBE failed again to eliminate any feature.

Although the goal of this project was to produce a predictor to classify galaxies as accurately as possible, it is important to examine the composition of the feature subsets selected. As we have noted above, SBE produces impressively small feature subsets that always include one obviously irrelevant feature. This is possible since the naive Bayes is insensitive to truly irrelevant features, but stochastic errors of the cross-validation estimates may make an irrelevant feature appear as giving a small advantage.

Except for PCA, the filters rank highly features related to symmetries and angles, which are features the astronomers and us expected. PCA selects features that, although unexpected, appear to have good discriminatory power, which we confirmed by a simple exploratory data analysis observing box-plots and histograms.

4. FUSION DATA

Sometimes the goal in a scientific application is not to build a predictor, but to discover a set of features that may provide scientists leads into the problem that interests them. For example, Guyon and Elisseeff [7] mention an application of variable ranking to identify genes from microarray data that discriminate between healthy and sick patients. The goal was not to build a classifier to distinguish between the patients, but to identify genes that code for proteins that may be used to develop drugs. We present an application on fusion physics data that has a similar flavor: We do not intend to build a classifier to identify an "interesting" state of the plasma, we only intend to identify which candidate variables are related with the interesting state.

Fusion is a nuclear reaction where lighter elements combine to form a heavier element. This reaction releases large amounts of energy that, if harnessed and controlled, represent a sustainable and environmentally sound energy source. To achieve nuclear fusion, the particles must be hot enough, in sufficient number and well contained for a sufficiently long time.

The most successful and promising fusion confinement device is known as a tokamak. High-confinement mode (H-mode) is the choice for next generation tokamak devices as it offers superior energy confinement, but it comes at a significant cost due to effects of edge localized modes (ELMs). ELMs cause rapid erosion of some components in tokamaks and giant ELMs can destroy other critical components. Recently, a "quiescent H-mode" of operation has been observed in the DIII-D tokamak operated by General Atomics. Quiescent operation is important because there are no ELMs. The scientists have detected that the presence of an edge harmonic oscillation (EHO) is associated with the QH-mode.

EHOs appear to provide an enhanced particle transport at the edge of the plasma that is rapid enough to provide the needed density control.

Currently, EHOs are identified mostly by visual means using the Fourier spectrum of the data from a magnetic probe. If an experiment seems to contain EHOs, data from other sensors (plasma velocity and the distance between the plasma edge and the tokamak wall) are consulted to verify the existence of the EHO. A program was developed at General Atomics that implements the rules that the scientists have derived from their visual observations to identify EHOs. The program uses a sliding time window to analyze the data and assigns an "EHOness" value to each time window. The program seems to *identify* EHOs satisfactorily, but it does not *explain* the presence of EHOs.

Scientists are interested in knowing which variables are related to the appearance of EHOs. The underlying hypothesis is that there is something in the data that will be useful to formulate a theory to explain EHOs. Our approach to this problem is to identify which of the candidate variables are relevant to create models that predict the EHOness of the experimental data.

During experiments with the tokamak, numerous sensors record vast amounts of data. Each experiment in DIII-D lasts for approximately six seconds and data from numerous sensors is recorded and stored in a database. We have extracted features that describe approximately 700 experiments that have been analyzed (visually) by our collaborator. We are using 37 variables that were identified by a domain scientist as candidates to be involved in the identification of EHOs. Each time window in the data receives a label output from the program that detects EHOs. We restrict the problem to a binary classification using labels that correspond to high/low levels of EHOness.

The data needs some preprocessing before being input to the feature selection algorithms. In our case, one of the major difficulties is that the data from multiple sensors is not sampled at the same rate or may start or end at different points in time. This is a typical problem with scientific data and requires that the data be registered. For a variety of reasons, some sensors may not have been activated for an experiment, and in consultation with our collaborator, we decided to discard the time windows that contain at least one missing value for a candidate variable.

The size and dimensionality of the data still allows for a meaningful exploratory data analysis. Visual examination of box-plots and histograms revealed that the data seemed to contain many outliers. As an attempt to deal with outliers, we use the median value of each variable in each time window. While this eliminated some outliers, many still remained, and we decided to eliminate the time windows that contained at least one variable in the top or bottom percentile of its range.

As we saw in the previous example with the FIRST data, depending on manual labeling of the data means that the training sets available are small. However, the fusion data does not suffer from the typical lack of labeled data, because

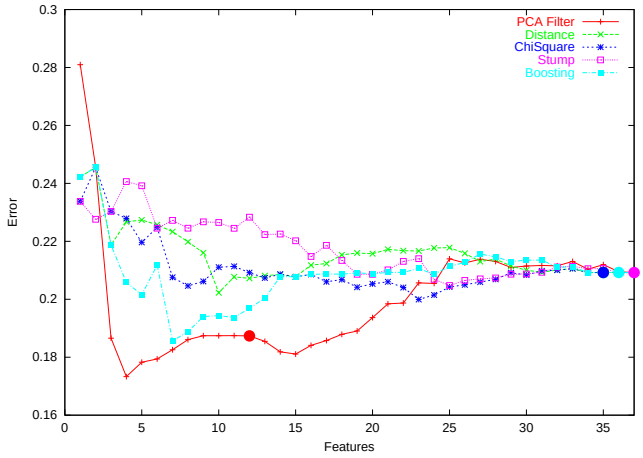


Figure 3: Error rates varying the number of features using the fusion data. The large dots represent the rankings of the random noise feature.

the labeling is performed by the program that implements the heuristics used by the scientists to find EHOs. After the preprocessing, our training set consists of 41818 instances, each described by a numeric vector of the 37 candidate features and some housekeeping features (experiment ID, time ID, etc.) that are ignored in the analyses.

Figure 3 presents the error rates of a naive Bayes trained on increasingly large feature subsets. As with the FIRST data, the PCA filter produced a compact feature subset that results in the lowest classification error of 17.3%. Although this error is not notably smaller than the error obtained with all the features (20.9%), the fact that very few features are necessary to explain the presence of EHOs is interesting.

There is significant overlap between the top ten features ranked by the different methods, except for the PCA filter that selects features that the other methods rank lower. Six of the features were ranked in the top ten by four filters, and an additional two were ranked in the top ten by three filters.

The SFS and SBE wrappers found feature subsets with three and four features, respectively, and both subsets resulted in accuracies of 16.2%. There was only one common feature in these subsets, and it was a feature that appeared consistently in the top ten rankings with the filters (except in the PCA).

Interpretation of the physical significance of these results is beyond of our abilities. The goal of this project was to identify variables that the scientists can use as leads to explain the presence of EHOs. While these results might provide useful leads, this is an ongoing project where we are exploring other feature selection methods as well as summarization of the results in novel ways.

5. REMOTE SENSING DATA

The automated production of maps of human settlements from satellite imagery is essential to studies of urbanization,

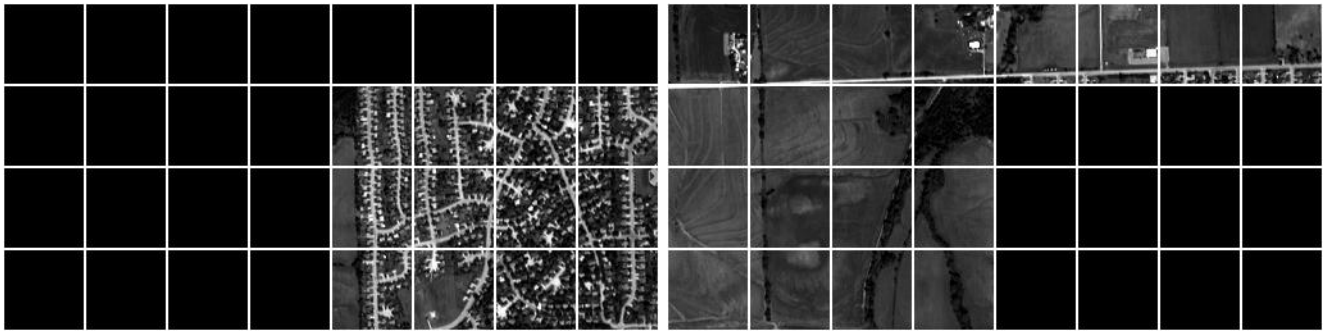


Figure 4: Example of a region in satellite imagery illustrating the ground truth. The image on the left shows the inhabited tiles while the image on the right shows the uninhabited tiles. Original satellite image by Space Imaging.

population movement, etc. While the spatial resolution of such imagery is high enough to make the identification of settlements feasible through visual means, it also implies that the size of the data is so large that such visual identification is infeasible for all but small regions of the world. Further, human populations tend to change over time as cities grow and shrink, making it necessary to update these maps periodically. In this application, we consider the use of data mining techniques to automate the production of these maps of human settlements. We used satellite imagery from IKONOS [16], which is available both as 4-band (near-infrared, red, green, and blue) multi-spectral images at 4m ground sample distance (GSD) and single-band panchromatic images at 1m GSD. Our initial focus is on multi-spectral imagery. Given its lower resolution, we can work with smaller amounts of data, while exploiting the additional information available in the four bands. Our previous work [14] has shown that the use of all four bands results in features that better represent the data than a combination of the bands into a single band.

Our approach to the identification of human settlements is as follows. We considered two satellite images—one from a region in Nebraska and the other from northern Mexico. We first divided each image into non-overlapping tiles, each of size 64×64 pixels. Next, we extracted four different sets of image texture features from each of the four spectral bands in the tiles. We also manually labeled each tile, through visual inspection, as either being inhabited–i.e., having human settlements–or as being uninhabited. Figure 4 shows examples of both types of tiles. A tile was considered inhabited if it was predominantly composed of man-made structures such as buildings. We currently restrict ourselves to a binary classification, though it is possible to handle partially inhabited tiles by considering additional classes. We also removed tiles that were of uniform intensity. The texture features, along with the class assignment, form a training set that is input to a classifier. For our images this resulted in 7419 instances. This classifier is then used to classify unseen tiles as inhabited or uninhabited.

Image texture can be considered as the spatial dependence of the pixel values in digital images. While a number of texture features have been proposed over the last several decades, none has proven superior for all applications, so deciding on the appropriate features remains a challenge. This challenge is compounded in the analysis of multi-band imagery, such as multi-spectral remote sensed images, since it is also not clear which band(s) should be used to compute the texture features.

Table 2: Number of components in the texture feature vectors for single-band and four-band images.

	GLCM	PS	wavelet	Gabor	total
single-band	20	20	24	60	124
four-band	80	80	96	240	496

In this work, we extracted texture features based on 1) gray level co-occurrence matrices (GLCMs), 2) the Fourier power spectrum, 3) wavelets, and 4) Gabor filters. The GLCM texture features summarize the spatial co-occurrence of pixel values at different spatial offsets [8][14]. The following five features were extracted at four offsets: angular second moment, contrast, inverse difference moment, entropy, and correlation. The power spectrum texture features consist of the average of the Fourier power spectrum computed over different regions in the two-dimensional frequency space [14]. The wavelet texture features are the mean and standard deviations of the energy of the frequency bands in a three-level multi-resolution discrete wavelet decomposition of an image [11][14]. The Gabor texture features are the mean and standard deviations of the outputs of a bank of scale and orientation selective Gabor filters [12][14].

There are several ways in which the textures from the four different bands can be combined. Based on our prior work, we extracted the texture features separately for each of the four bands, and concatenated them to form a feature vector of length 496, with 124 features contributed by each band. With such a long feature vector, it is important that we use feature selection techniques to keep only the relevant features. Table 2 shows the number of components per texture feature vector for both single-band and four-band images. The complete composite texture feature vector for the satellite image thus contains a total of 496 components.

For the remote sensing data, we performed two sets of exper-

iments. First, we considered the four sets of texture features independently, and then we combined all the features. Our goal was to understand the performance of the feature selection algorithms, identify if any of the features performed better than the rest, and see if combinations of features worked better than each set considered independently. Tables 3 and 4 summarize the minimum error rates for each feature selection method using each of the four sets of texture features in isolation and in combination with the naive Bayes and decision tree classifiers, respectively.

For this problem, we found that decision trees gave better results than the naive Bayes classifier, often by more than 1% error rate. When the combination of the four sets of texture features was used without any feature selection, the naive Bayes classifier had an error rate of 41.8% compared to the decision tree error rate of 25.6%. This is to be expected as the naive Bayes classifier is known not to perform well in the presence of many features and the decision tree can be considered to have in-built feature selection. Thus, the explicit use of feature selection benefits the naive Bayes classifier more than the decision tree classifier. Similarly, we observe that when we consider only the Gabor features, which are more numerous than the other features, the error rate is higher for the naive Bayes classifier (33.2%) than the decision trees (25.8%).

The PCA filter performed very differently from the other techniques. It selected texture features that were rarely selected by other methods (e.g. power spectrum features). It also selected features in the red and blue bands which were rarely selected by other techniques. Further, it ranked the noise feature quite highly (often within the top 10% of the features in the order selected), even though the feature was irrelevant. We believe this poor performance is the result of the PCA filter ignoring the class of each instance.

We also found that there was not much difference in performance among the remainder of the feature selection techniques, though with the naive Bayes classifier, the PCA filter and the Stump filter selected features that performed slightly worse than other techniques.

From a domain standpoint, we also made several interesting observations about the top ten features that were selected. These observations were consistent across the two sets of experiments.

The features corresponding to the green and NIR bands were selected more often than those corresponding to the blue and red bands. This may indicate that we could reduce the computation time as well as storage by focusing on only two of the four bands.

A majority of the top ten features are from the GLCM category, while the wavelet and Gabor features are selected less frequently. Power spectrum features are rarely selected. This agrees with our prior experience with texture features in an information retrieval application [14].

The GLCM features selected in the top ten features are entropy and inverse difference moment. The contrast and correlation features were rarely selected, while the angular second moment was selected occasionally. This indicates that we may be able to reduce the number of GLCM features to two or three.

The wavelet and Gabor features selected correspond to the higher frequencies. While we considered three levels in the wavelet decomposition, only the first two were ever selected in the top ten features. Similarly, for the Gabor features, only the two highest of five scales were selected. Further, for the Gabor and wavelet features, it was mainly the energy feature that was selected in the top ten; the standard deviation was rarely selected. These observations can again be used to reduce substantially both the computation time for calculation of the features as well as the storage requirements.

We also observed that combining all the four sets of texture features did not provide any benefit. The GLCM features performed the best, with the Gabor and wavelet features a close second. In contrast, the power spectrum features typically had an error rate 1% higher than the other methods.

Table 3: Comparison of the minimum error rate using the naive Bayes classifier for the different feature selection methods with the texture features considered independently and in combination.

Method	Pow. Sp	GLCM	Wavelet	Gabor	All
No filter	29.0	27.5	28.8	33.2	41.8
PCA	28.0	27.1	28.2	27.8	28.8
KL	27.1	26.0	26.7	26.1	26.0
χ^2	27.0	26.0	26.5	26.1	26.0
Stump	28.2	26.6	27.8	26.6	26.9

Table 4: Comparison of the minimum error rate using the decision tree classifier for the different feature selection methods with the texture features considered independently and in combination.

Method	Pow. Sp	GLCM	Wavelet	Gabor	All
No filter	26.5	25.1	24.6	25.8	25.6
PCA	25.9	24.3	24.3	25.4	24.8
KL	25.9	24.5	24.6	25.5	25.1
χ^2	25.7	24.7	24.6	25.6	24.9
Stump	25.4	24.3	24.4	25.4	24.8

There are several different ways in which this work can be extended. First, we would like to consider remote sensing imagery from various parts of the world to better understand how well our feature selection techniques perform. Second, we want to expand the ground truth to include tiles which are a mix of inhabited and uninhabited regions; we expect that having more than 2 classes will improve the accuracy of the classifier. Finally, we want to investigate if the observations made about the importance of various features carry over to images from other regions in the world.

6. DISCUSSION

The three diverse examples of scientific applications that we presented in this paper illustrate the difficulties of performing feature selection in scientific data.

One of the problems faced frequently is that labeling of examples to form a training set may contain errors. In FIRST and the human settlements problems the labeling was performed manually. This limits the size of the training set, which together with the subjective nature of the labeling, restricts the accuracy we can expect from classification methods. In the fusion data, the labeling is automated, but it may also contain mistakes, as the program implements heuristics that may not be valid in all cases.

The three applications demonstrated that preprocessing is crucial for the success of these projects. Preprocessing is also crucial in mining commercial data, but the nature of the preprocessing is different. In scientific data we frequently generate features from low-level data that may be noisy and large (we are approaching petabyte ranges in astronomical surveys and high resolution computer simulations). In the FIRST data the noisy images were processed into a catalog (by the astronomers) creating fairly noiseless processed data that we used to create high-level features; in the fusion data we smoothed the observations using the medians of time windows and, since we had enough data, we dealt with outliers simply by removing them; in the remote sensing data the texture features were processed to ensure that orientation independent.

Feature selection is an important task in scientific applications for different reasons. Removing redundant or irrelevant features is likely to improve the accuracy of classifiers. Optimizing the accuracy was the goal in FIRST and the human settlements applications. Identifying which features are related to the classification can also provide insights into the underlying phenomena, which is of interest to the scientists. In some cases, providing these insights is the goal of the project, as in the fusion data problem.

Generating features is expensive computationally, and using feature selection to identify which features are worth generating saves resources that can be used for analyses or other purposes. In particular, extracting features such as texture from images or simulation data is very expensive. Our results with the remote sensing data, for example, indicate that we can save considerable resources by calculating features from only two out of four bands and wavelet and Gabor features corresponding only to high frequencies.

Domain knowledge and exploratory data analysis (EDA) are alternatives to the methods presented here. However, these alternatives are not always effective. Domain knowledge helps to identify candidate variables as in the FIRST and fusion problems, but pruning the candidates is not straightforward. For example, we found that it is not obvious which variables of a highly correlated group should be kept, and including highly correlated variables into the analysis might be useful (because of the potential discovery that one feature presents an advantage in classification accuracy or is less expensive to generate, or perhaps because both highly correlated variables are needed).

EDA does not work when the data is massively large or when each instance is described by many features. In the case of the remote sensing data, for example, having close to 500 features precludes an effective EDA. Visually examining 500 pairs of box-plots or histograms is too tedious and prone to mistakes as important features might be overlooked.

Wrappers are very effective feature selection methods when they can be applied. However, the evaluation of each feature subset may be computationally expensive and these methods are impractical for large data sets. We experienced this when trying to reduce the entire set of features in the remote sensing data.

On the other hand, filter methods are very efficient. Some filter methods (like the KL distance filter and the Chi-Square filter used in this paper) require only one sequential pass through the data. This efficiency makes it practical to execute several algorithms and identify highly relevant features that are ranked highly by several methods. This approach suggests future algorithms that use ensembles of ranking algorithms.

Our experience suggests that simple methods work well in many cases. While using non-linear classifiers and more sophisticated feature selection methods might result in higher classification accuracies, the results obtained with simpler techniques, such as the ones presented in this paper, are adequate to identify relevant features in many applications.

Acknowledgments

We would like to thank the rest of the members of the Sapphire team at LLNL (www.llnl.gov/casc/sapphire) for useful discussions and computational help. We gratefully acknowledge our FIRST collaborators Robert Becker, Michael Gregg, David Helfand, Sally Laurent-Muehleisen, and Richard White for their technical interest and support of this work. We also thank Keith H. Burrell and Michael Walker for their help and support of the work with the fusion data. We acknowledge Doug Poland for the tool used to generate the ground truth in the remote sensing application. Finally, we thank Bronis de Supinski for his comments on a draft of this manuscript.

UCRL-CONF-202657. This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

7. REFERENCES

[1] R. H. Becker, R. White, and D. Helfand. The FIRST survey: Faint images of the radio sky at twenty-cm. *Astrophysical Journal*, 450:559–599, 1995.

[2] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. CRC Press, 1984.

[3] E. Cantú-Paz and C. Kamath. Evolving neural networks to identify bent-double galaxies in the first survey. *Neural Networks*, 16(3–4):507–517, 2003.

[4] S. Das. Filters, wrappers and a boosting-based hybrid for feature selection. In C. Brodley and A. Danyluk,

editors, *Proceedings of the 18th International Conference on Machine Learning*, pages 74–81, San Francisco, CA, 2001. Morgan Kaufmann Publishers.

[5] I. K. Fodor, E. Cantú-Paz, C. Kamath, and N. Tang. Finding bent-double radio galaxies: A case study in data mining. In *Interface: Computer Science and Statistics*, volume 33, 2000.

[6] I. K. Fodor and C. Kamath. Dimension reduction techniques and the classification of bent double galaxies. *Computational Statistics and Data Analysis*, 41:91–122, 2002.

[7] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[8] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3:610–621, 1973.

[9] S. H. Huang. Dimensionality reduction on automatic knowledge acquisition: a simple greedy search approach. *IEEE Transactions on Knowledge and Data Engineering*, 15(6):1364–1373, 2003.

[10] R. Kohavi and G. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.

[11] S. Mallat. A theory for multi-resolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989.

[12] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.

[13] K. Mardia, J. T. Kent, and J. Bibby. *Multivariate Analysis*. Academic Press, 1995.

[14] S. D. Newsam and C. Kamath. Retrieval using texture features in high resolution multi-spectral satellite imagery. In *SPIE Conference on Data Mining and Knowledge Discovery: Theory, Tools, and Technology VI*, 2004.

[15] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[16] Space Imaging web site: www.spaceimaging.com.

[17] R. L. White, R. Becker, D. Helfand, and M. Gregg. A catalog of 1.4 GHz radio sources from the FIRST survey. *Astrophysical Journal*, 475:479, 1997.