An Information Foraging Approach to Determining the Number of Relevant Features

Brian Connolly
Cincinnati Children's
Hospital Medical Center
Burnet Ave
Cincinnati, OH 45229
brian.connolly
@cchmc.org

Benjamin Glass
Cincinnati Children's
Hospital Medical Center
Burnet Ave
Cincinnati, OH 45229
benjaminglass
@gmail.com

John P. Pestian
Cincinnati Children's
Hospital Medical Center
3333 Burnet Ave
Cincinnati, OH 45229
john.pestian
@cchmc.org

Abstract

For many types of high-dimensional data, such as natural language corpora, the vast majority of extracted variables or features are essentially noise. Culling such features can not only reveal important patterns, but also improve the performance of supervised and unsupervised machine algorithms. Most research on feature selection has focused on the statistical measures used to rank features. Meanwhile, little work has been done developing techniques for identifying the optimal subset of features without repeatedly training models. However, developing such techniques is important, as they can significantly decrease computation time while providing a way to determine the features that characterize the classes within a data set, independent of how the data may be classified in the future. Here we introduce a novel method based on information foraging that works in conjunction with existing feature ranking methods to automatically determine a subset of important features. The method is demonstrated on simulated and linguistic data from psychiatric interviews. We show that the method is able to accurately determine the features that characterize the classes within both data sets. The method is fast, simple, and independent of any method of classifying the data, and can be extended to any high-dimensional data set.

1 Background

For many types of high-dimensional data, such as natural language corpora, gene microarrays, and images, the vast majority of extracted variables or features are essentially noise (Yu and Liu, 2004). Culling such features can not only reveal important patterns, but also improve the performance of supervised and unsupervised machine algorithms (Guyon and Elisseeff, 2003; Saeys et al., 2007). For example, Pestian et al. (Pestian et al., 2016) have recently used natural language processing (NLP) and supervised machine learning methods to automatically distinguish suicidal from non-suicidal patients using words and phrases from psychiatric interviews (Pestian et al., 2016). In that work, identifying which types of words and phrases were most discriminative not only improved classification performance, but also provided important insights into the language of those at risk of suicide.

Feature selection is usually done in the context of optimizing machine learning models, and so feature selection techniques are divided into three categories by how they relate to the search over such models: filter, wrapper, and embedded methods (Blum and Langley, 1997; Saeys et al., 2007). Filter methods rank features using a statistical measure of relevance (Forman, 2003; Yang and Pedersen, 1997). Typically, lower-ranked features are removed prior to training a machine learning model. By contrast, in wrapper methods, the optimal feature subset is identified by repeatedly training a model on multiple feature subsets and evaluating its performance (Kohavi and John, 1997). The search for an optimal model is "wrapped" in the feature subset search. Finally, in embedded methods the feature search is performed in conjunction with the model search. For example, the number of parameters can be incorporated as a regularization term to be minimized in the objective function (Weston et al., 2003).

Most research on feature selection has focused on the statistical measures used to rank features (Forman, 2003; Yang and Pedersen, 1997). Mean-

Proceedings of the 15th Workshop on Biomedical Natural Language Processing, pages 175–180,
Berlin, Germany, August 12, 2016. ©2016 Association for Computational Linguistics

while, little work has been done developing techniques for identifying the optimal subset of features without repeatedly training models (Koller and Sahami, 1996; Ding and Peng, 2005). However, developing such techniques is important, as they can significantly decrease computation time while providing a way to determine the features that characterize classes within a data set, independent of any classification method.

Here we introduce a novel method based on information foraging that works in conjunction with existing feature ranking methods to automatically determine a subset of important features. Information foraging is a behavioral model for maximizing the rate of attaining valuable information (Pirolli and Card, 1999). It assumes that useful information exists in a patchy structure, where the diminishing return of a continued search in a patch must be balanced with the time cost of moving to a new patch.

The utility of our approach is best demonstrated with an example of a typical feature selection approach for text classification. Suppose a large data set of text documents is divided into multiple classes. We want to classify documents into the correct categories using word frequencies. Typically, a text data set may contain many thousands of unique words, most of which have no discriminative power (Scott and Matwin, 1999). Feature selection is used to determine the features that best discriminate between the classes, thereby optimizing classifier performance. A univariate filter method, such as information gain (Fano and Wintringham, 1961) for discrete data, or Analysis of Variance (ANOVA) (Michel et al., 2008) for continuous variables, may be applied to rank the features by their discriminative power. A subset of top-ranked features are then chosen based on some ad-hoc threshold, or by using a wrapper method, where classifiers are built using various sets of top ranked features. The classifier with the best performance then determines the best feature subset. Classifier performance is evaluated using some flavor of bootstrapping, potentially making this method computationally expensive.

In this scenario, the optimal number of features is defined by both the method of ranking features and the classifier; there is no 'objective' determination of which features characterize the classes.

From a computational perspective, no matter how efficient the subset search strategy, a wrapper or embedded method which entails training models will be more costly than a univariate filter subset selection which runs in O(N) time. Other work on filter-only methods for subset selection has been primarily multivariate, identifying correlations between variables and eliminating redundant ones. Hall and Smith (Hall and Smith, 1997) used Pearson's correlation for forward selection filtering, with good results on fairly low-dimensional data. Others (Koller and Sahami, 1996; Yu and Liu, 2004; Ding and Peng, 2005) have used Markov blanket filtering to iteratively remove redundant features via backward elimination. These generally have a complexity of O(N^2).

In this work, we show the proposed foraging-based feature selection leads to performance gains comparable to wrapper methods on a text classification task, while running in linear time. In addition, the algorithm is useful simply for the objective identification of a relevant feature subset, since it is deterministic and entirely independent of the choice of learning algorithm. Further, the method is not tied to a particular feature ranking method, but rather it simply provides a method of determining the optimal number of features *given* a ranking method.

2 Theory

The method of selecting the number of features is based on the Holling's Disk equation (Holling, 1959), which has been used to explain the foraging behavior of both animals (Stephens, 1990; Stephens and Krebs, 1986) and humans (Winterhalder and Smith, 1992). It has also been useful in understanding information foraging (e.g., in web searches (Pirolli, 2007)). The equation is dependent on three variables: the time spent gathering energy from a certain food type i (t_{Wi}), the amount of time it takes to travel to that food type $1/\lambda_i$, and the energy gained from that food type ($g_i(t_{Wi})$). The overall rate of gain for k food sources is then

$$R(k) = \frac{\sum_{i=1}^{k} \lambda_i g_i(t_{Wi})}{1 + \sum_{i=1}^{S} \lambda_i t_{Wi}}. \tag{1}$$

Given S food types, the optimal diet is then found through an algorithm suggested by (Stephens and Krebs, 1986). In this algorithm, the profitability of the food type, given by $g_i(t_{Wi})/t_{Wi}$, is ranked so that $g_1(t_{W1})/t_{W1} > g_2(t_{W2})/t_{W2} > ... > g_S(t_{WS})/t_{WS}$. Food types are added until the rate of gain for a type of top k food types is greater than the $k+1$ food type; that is, until

$$R(k) > g_{k+1}/t_{Wk+1}. \quad (2)$$

For our purposes, feature subset selection is modeled as a diet optimization task, where features are represented by food types, and a diet is a subset of features. Each feature or food type added to the diet may add gain in terms of the informativeness of the feature, but entails cost in terms of sparseness.

In the present work, the gain is defined by the *informativeness* obtained from feature i, which is broadly defined by any parametrization of the statistical differences between classes. As the class differences for a given feature will be defined in this work as a p-value, we choose two definitions of informativeness which increase with the differences between classes: $1 - p_X$ and $1/p_X$, where p_X is defines as the p-value from either the KS-tests or ANOVA. The time between food types is taken as the mean number of data points between appearances of feature i (Jones, 1987), where each data point equals one time unit. The time spent gathering energy from a food type is arbitrarily set to unity for all i ($t_{Wi} = 1$); λ_i is defined as

$$\lambda_i = \frac{\mathrm{Sum\ of\ Non-ZeroFrequencies\ for\ Feature\ } i}{\mathrm{TotalDataPoints}}. \quad (3)$$

This is the same equation as the reciprocal of the mean time between failures, where “failures” are taken to be non-zero feature frequencies.

3 Experiments

The method is demonstrated on two kinds of data: simulated data sets and a linguistic data set from a clinical trial.

The goal of the simulated experiments is to show that the method is able to accurately identify subsets of features with inter-class statistical differences. In these experiments, the performance of the algorithm is evaluated based on its ability to accurately identify these subsets. The goal of demonstrating the method on clinical trial data is to evaluate the method within a more realistic context of a wrapper method applied to linguistic data. Evaluating the method’s performance on such data also illustrates its behavior on data containing redundant and correlated features.

Each simulated data set is comprised of data points from two classes. (The number of data are kept small to reflect the small sample sizes typically found in clinically annotated NLP data sets (Hutton, 2012).) The data from the first class (class A) are generated from a Gaussian distribution with mean 0 and standard deviation σ. The data from the second class (class B) are generated from two Gaussian distributions; $f \times 100\%$ of the features are generated with mean 1 and standard deviation σ, while the rest of the features are generated in the same fashion as those from class A, with mean 0 and standard deviation σ. In this way, $f \times 100\%$ of the features are generated with inter-class differences.

The performance of the algorithm is then evaluated as a function of the definition of gain, sparsity of the data (s), the total number of features (F), number of features with statistical differences (f), and statistical differences between features (parameterized by σ). The gain is define in four ways: as 1-p-value from the Kolmogorov-Smirnov test (Darling, 1957) ($1 - p_{KS}$), 1-p-value from ANOVA (Fisher, 1992) ($1 - p_{ANOVA}$), and the reciprocal of the KS and ANOVA p-values ($1/p_{KS}$ and $1/p_{ANOVA}$, respectively). The influence of λ_i is also studied by setting it to its empirical value and to unity. When they are not being varied, the default values for F, s, σ and f are: $1,000$, 0.5, 0.2 and 0.5, respectively.

The data from the clinical trial are derived from the Suicide Thought Markers study (Pestian et al., 2016). In this study, three hundred seventy-nine adults and adolescents from Cincinnati Childrens Hospital Medical Center (CCHMC), University of Cincinnati (UC), and Princeton Community Hospital (PCH) were enrolled during the course of the study between October 2013 and March 2015. Participants were evenly divided into three subject groups: suicidal, patients with mental illness, and controls. Suicidal subjects consisted of patients who presented in the Emergency Department (ED) with suicidal ideation or behaviors; the mental illness group was not suicidal, but had a mental health diagnosis; and the control group had no mental illness diagnosis and was not suicidal.

Subjects were then asked five open-ended, ubiquitous questions (UQs) (Pestian, 2010; Pestian et al., 2015): Do you have hope?, “Do you have any fear?”, “Do you have any secrets?”, “Are you angry?”, and “Does it hurt emotionally?”. These questions were intended to stimulate conversation for language sampling, and would later form the basis of the training sample for the machine learning algorithm. The interviews were transcribed and the subjects words were extracted in a systematic way.

For classification purposes, each subject was characterized by (1) their subject group and (2) a vector of word (1-gram) frequencies. Due to the extreme variability of word frequencies and interview lengths, the frequencies were normalized to smooth the frequency distributions and lessen the classifiers sensitivity to interview length. The word frequencies were therefore logarithmically (log(x+1)) transformed to smooth the frequencies, and further L2-normalized at the subject level as to base the classification on relative word frequencies.

Only suicidal and control patients are used in the present work. To test the method on various sizes and types of data, the data are split three ways: patients from CCHMC (pediatric patients), patients from PCH and UC (adults patients), and patients from all three hospitals. In the end, 2,471, 4,788, and 5,457 unique words were extracted over 84, 169, and 253 suicidal and control subjects from CCHMC, PCH and UC, and all hospitals, respectively.

The number relevant of features are then evaluated using the method presented in this work, and a wrapper method whereby the performance of Support Vector Machine (SVM) classifiers are evaluated using LOO cross-validation. Note the classifications here are simplified versions of the classifications in (Pestian et al., 2016); for instance, the features here are not partitioned based on the questions.

4 Results

Figure 1 show the F_1 scores for selecting features, varying the total number of features (F), the matrix sparsity (s), σ, and the fraction of features with statistical differences. The method is able to determine the features with significant features of a large parameter space when $1 - p_X$ defines the gain. On the other hand, when the reciprocal p-values are used, the method fails spectacularly, indicating that p_X must be bounded or it must possess a more direct statistical interpretation. This aside, performance is, to a degree, invariant to the type of statistic used; the KS test p-value performs better when the matrix is sparse, while the ANOVA p-value works better when the statistical differences are small. This may be less of a reflection on the method, and more to do with the KS test's ability to detect differences in small data samples, and ANOVA's ability to detect statistical differences when the distributions are Gaussian.

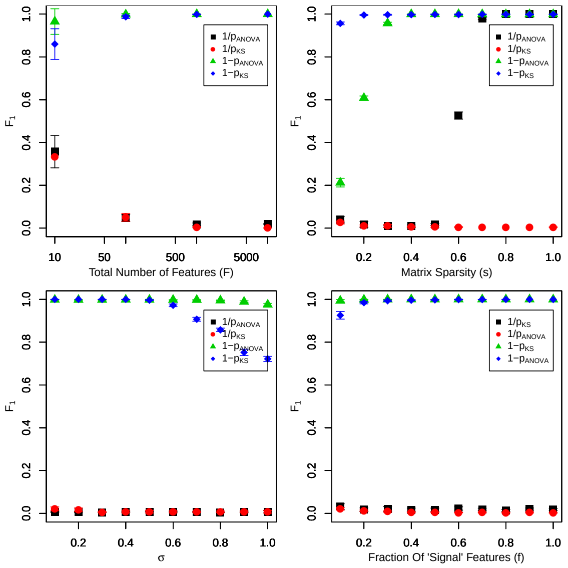

Figure 1: The F_1 score for identifying the features in simulated as a function of F, s, σ, and f. These scores are evaluated using varying definitions of gain: $1/p_{ANOVA}$ (squares), $1/p_{KS}$ (circles), $1 - p_{ANOVA}$ (triangles), and $1 - p_{KS}$ (diamonds).

Figure 2 shows the same plots with the mean time between patches set to unity ($\lambda_i = 1$). The two sets of figures look nearly identical indicating that λ_i does not play a significant role in determining the number of features.

Figure 3 shows the area under the cross-validated receiver operating curve (AROC) of the SVM classifier as a function of the number of top-ranked features. The number of features determined by our method, along with the corresponding AROC, are circled on these plots. In these plots, the relevant number of features are the minimal number of features that optimize classifier performance. When the KS test p-values are used for the gain, the method is unable to predict the optimal features. However, the oscillating performance as the number of features increase indicate the KS test may not be the best choice for feature ranking for this data set. In contrast, the ANOVA p-value is more stable, leading to more monotonic curves, and the method is better able to determine the optimal number of features.

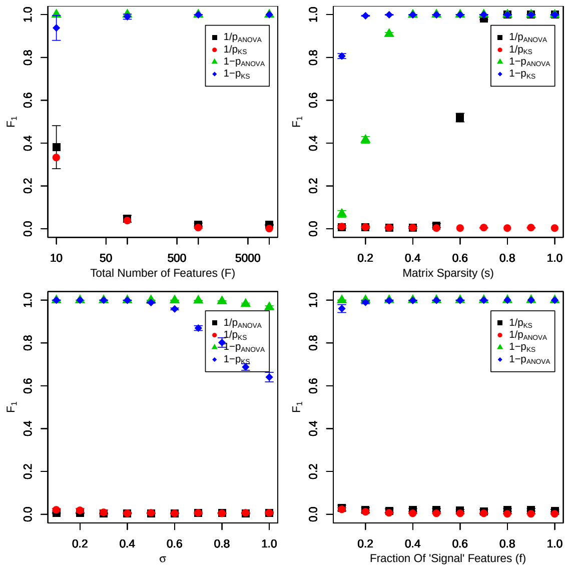

Figure 2: The plots in 1 with the mean time between patches set to unity ($\lambda_i = 1$).

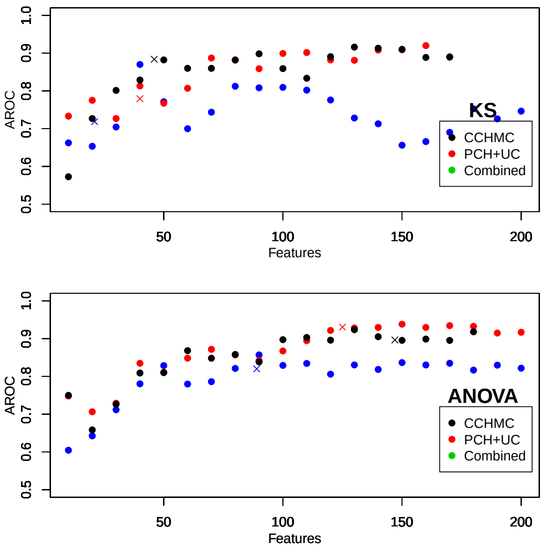

Figure 3: The top plot shows the cross-validated performance of an SVM classifier as a function of the number of top features ranked according to the KS test p-value (top) and ANOVA p-value (bottom) for the CCHMC (diamonds), UC+PCH (circles), and combined data sets (squares). The number of features determined by our method, along with the corresponding AROCs, are circled.

5 Discussion

The results from simulated data indicate there is some flexibility in the definition of informativeness, as long as the statistic gives a proper ranking of features and the statistic is bounded and/or possesses some statistical meaning. The results from real data reflect this conclusion, showing the method performs better when the feature ranking is more accurate. The decrease in classifier performance does not occur until a large number of features are introduced as input to the classifier, which is not shown in the figures. The focus of this study, however, is to determine whether or not the method presented is able to cull superfluous features; the point at which 'gain' in classification performance levels off clearly coincides with the number of features predicted by the method when the ANOVA method is used for feature ranking.

The bad performance of the method when the reciprocal of the p-values are used for the gain, indicates that the gain must be bounded in some way, or that the statistic must have a more direct statistical interpretation. In contrast, the simulated results suggest the method is fairly insensitive to the choice of λ_i, which parametrizes the sparsity of the feature.

Also, although the method is essentially built for univariate data, the performance on real data was good despite the inevitable redundancies and correlations of the features, provided the informativeness measure properly ranked the features.

6 Conclusions

We have presented a simple, fast, and effective method of determining the number of features that characterize classes within a data set where the features are univariate. We have also show it to be useful in determining the features in a linguistic data set, despite the features' inherent redundancies and correlations.

While the method was show to properly identify features that characterize features with inter-class statistical differences, its performance is better when the statistic is able to effectively rank the features in terms of statistical relevance. We have also shown that it performs better when p-values are used, as opposed to their reciprocal, showing the definition of informativeness is important. Whether this is because a p-value is a bounded positive number less than 1 or because it has a direct statistical interpretation merits exploration. For instance, the question remains, could any statistic that effectively rank features be inserted into a softmax function and be used to parameterize gain? Also, the method would doubtlessly perform better if correlations and redundancies were somehow accounted for, possibly by grouping correlated features.

References

Avrim L Blum and Pat Langley. 1997. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1):245–271.

Donald A Darling. 1957. The kolmogorov-smirnov, cramer-von mises tests. *The Annals of Mathematical Statistics*, 28(4):823–838.

Chris Ding and Hanchuan Peng. 2005. Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology*, 3(02):185–205.

Robert M Fano and WT Wintringham. 1961. Transmission of information. *Physics Today*, 14:56.

RA Fisher. 1992. Statistical methods for research workers. In *Breakthroughs in Statistics*, pages 66–70. Springer.

George Forman. 2003. An extensive empirical study of feature selection metrics for text classification. *The Journal of Machine Learning Research*, 3:1289–1305.

Isabelle Guyon and André Elisseeff. 2003. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182.

Mark A Hall and Lloyd A Smith. 1997. Feature subset selection: a correlation based filter approach. In *International Conference on Neural Information Processing and Intelligent Information Systems*, pages 855–858.

Crawford S Holling. 1959. Some characteristics of simple types of predation and parasitism. *The Canadian Entomologist*, 91(07):385–398.

John J Hutton. 2012. *Pediatric Biomedical Informatics: Computer Applications in Pediatric Research*, volume 2. Springer Science & Business Media.

James V Jones. 1987. *Integrated logistics support handbook*. Tab Books.

Ron Kohavi and George H John. 1997. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273–324.

Daphne Koller and Mehran Sahami. 1996. Toward optimal feature selection.

Vincent Michel, Cécilia Damon, and Bertrand Thirion. 2008. Mutual information-based feature selection enhances fmri brain activity classification. In *Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on*, pages 592–595. IEEE.

John P Pestian, Jacqueline Grupp-Phelan, Kevin Bretonnel Cohen, Gabriel Meyers, Linda A Richey, Pawel Matykiewicz, and Michael T Sorter. 2015. A controlled trial using natural language processing to examine the language of suicidal adolescents in the emergency department. *Suicide and life-threatening behavior.*

John P. Pestian, Michael Sorter, Brian Connolly, K. Bretonnel Cohen, Cheryl McCullumsmith, Jeffry T. Gee, Louis-Philippe Morency, Stefan Scherer, and the STM Research Group. 2016. A machine learning approach to identifying the thought markers of suicidal subjects: A prospective multicenter *Suicide and Life-Threatening Behavior*.

John Pestian. 2010. A conversation with edwin shneidman. *Suicide and life-threatening behavior*, 40(5):516–523.

Peter Pirolli and Stuart Card. 1999. Information foraging. *Psychological Review*, 106(4):643.

Peter Pirolli. 2007. *Information foraging theory: Adaptive interaction with information*. Oxford University Press.

Yvan Saeys, Iñaki Inza, and Pedro Larrañaga. 2007. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517.

Sam Scott and Stan Matwin. 1999. Feature engineering for text classification. In *ICML*, volume 99, pages 379–388. Citeseer.

David W Stephens and John R Krebs. 1986. *Foraging theory*. Princeton University Press.

DW Stephens. 1990. Foraging theory: up, down, and sideways. *Studies in avian biology*, 13:444–454.

Jason Weston, André Elisseeff, Bernhard Schölkopf, and Mike Tipping. 2003. Use of the zero norm with linear models and kernel methods. *The Journal of Machine Learning Research*, 3:1439–1461.

Bruce Winterhalder and Eric Alden Smith. 1992. Evolutionary ecology and the social sciences. *Evolutionary ecology and human behavior*, pages 3–23.

Yiming Yang and Jan O Pedersen. 1997. A comparative study on feature selection in text categorization. In *Proceedings of the 14th International Conference on Machine Learning (ICML)*, volume 97, pages 412–420.

Lei Yu and Huan Liu. 2004. Efficient feature selection via analysis of relevance and redundancy. *The Journal of Machine Learning Research*, 5:1205–1224.