Comparing the Results of Support Vector Machines with Traditional Data Mining Algorithms

Scott Pion and Lutz Hamel

Abstract— **This paper presents the results of a series of analyses performed on direct mail data. A total of 577 features were used to identify individuals who were most likely to respond to a life insurance mailing. The tools used for these analyses were support vector machines (SVMs), logistic regression, and a series of algorithms employed automatically using the integrated "Affinium Model" © data mining software package produced by the Unica Corporation. The data mining software included linear regression, logistic regression, classification and regression trees (CART) and neural networks. Results indicated that SVMs performed slightly better than logistic regression, and logistic regression performed slightly better than the data mining tool. The results indicated that SVMs can be used with very unbalanced and noisy data, and that SVMs can be used to rank observations based on likelihood.**

I. Introduction

THE purpose of the current study was to compare the results of support vector machines (SVMs) with logistic regression and a data mining software package. The goal of the analysis was to predict who would be most likely to respond to a mailing solicitation for life insurance. SVMs were developed as a machine learning algorithm for classifying objects into classes [1], [2]. A review from a recent conference on SVMs [3] indicated that SVMs are often used for tasks that are almost perfectly determined by their predictive variables. For example, classifying faces as male or female is almost completely determined by the colors of the pixels of the photographs. In contrast, in the current study the task was to determine who would be most likely respond to a direct mail solicitation based on demographic variables. The demographic variables only explain a small amount of the variance in who does or does not respond and this makes it more difficult to construct a predictive model. In direct mail models, it is not typical to classify individuals as responders or non-responders, because everyone is unlikely to respond. Therefore probabilities of responding, rather than classifications as responders or non-responders, are typically the output of direct mail predictive models.

This work was supported in part by the Amica Life Insurance Company.

Scott Pion is with the Department of Computer Science and Statistics, University of Rhode Island, Kingston, RI 02881 USA (phone: 401-874-2701; e-mail: pions@cs.uri.edu).

Lutz Hamel is with the Department of Computer Science and Statistics, University of Rhode Island, Kingston, RI 02881 USA (phone: 401-874-2701; e-mail: lutz@inductive-reasoning.com).

Instead of building models for an exact classification, we are interested in building models that predict the probability of belonging to a certain category. This is more in line with common practice in direct mail in which instead of modeling customers as responders or non-responders, customers are modeled according to their propensity for responding to a mailing. The SVM tool LIBSVM [4] was used for the analyses. At the time this analysis was performed, October of 2003, LIBSVM did not output probabilities. By dropping the sign function from the decision function for binary classification SVMs, we can convert the decision function from a function that only returns 1 or -1 to a function that returns a positive or negative real number. This real number, which represents the distance and orientation of an observation from the discriminating hyperplane, can loosely be interpreted as the likelihood of belonging to a particular class [5], [6]. Intuitively this is clear because the closer a point is to the hyperplane, the less certain the classification. One of the goals of this study is to determine if these values can be used to rank individuals from least likely to most likely to respond to a mailing. A common technique [5], [6] is to use the values of the SVM decision function to rank observations and then scale them to convert them into probabilities. Because lift is used to evaluate models in the current study, we do not need exact probabilities. We only need to rank individuals from least to most likely to respond to a mailing. Therefore, scaling of the decision function values in order to map them to probabilities is not needed.

II. Methodology

A. Dataset

The training dataset consisted of 577 features and 250,994 observations. The data is from the Amica Life Insurance Company©. It is an extremely unbalanced dataset in which 99.15% of the individuals mailed did not respond and 0.85% of the individuals did respond. The challenge of the current analysis is to build a model on an unbalanced dataset that does not default to a useless model that simply classifies all observations as non-responders.

Another challenge in the dataset is its sheer size in terms of number of observations. If it were possible to cut down on the training dataset without sacrificing predictive power of the model it would greatly reduce both computer memory demands and computing time. In the current study we consider whether eliminating some of the 248,869 non-responders impacts the performance of the SVM models. If

the impact is minimal then time could be saved at every step of the analysis. We consider whether a small subset of the non-responders can adequately represent all of the non-responders. Another question we address is whether it is possible to reduce the total number of *features* in the dataset. One approach implemented in the current analysis is to use principal components analysis to reduce the number of features. Another approach is to use logistic regression to select features and then use those features in an SVM.

The individuals who were sent life insurance mailings already owned homeowner or automobile insurance. Therefore, the dataset consisted of a variety of demographics and features related to automobiles and homes. Examples included age, number of children, marital status, number of mailings received in the past, and past responses to mailings. Because logistic regression was used primarily in the past, the dataset had been transformed for use in logistic regression. To account for non-linear relationships between continuous features and the target variable, all variables were standardized (to avoid multi-colinearity problems) and then squared in order to allow for the modeling of quadratic relationships. Missing values were replaced with averages and then an additional binary variable was created to indicate whether the value was a missing value. In regression, this process allows the missing values to be treated differently than the non-missing values. These transformations resulted in a total of three variables for each of the original continuous variables: one for the original, one for the missing value indicator, and one squared value. Categorical variables were decomposed into a series of binary variables that indicated membership in each of the categories. Because this data was prepared specifically for logistic regression, an interesting question is how well SVMs will perform on this data. It may be possible that missing values and squared features do not lead to greater SVM performance. The squared data is especially suspect, because Hsu, Chang, and Lin [7] have suggested that the SVM with a radial basis function kernel can account for non-linear relationships.

B. Evaluating the Results

The results of this series of analyses were evaluated using the measure "lift." Lift can be thought of as the gains that are achieved when using the model as opposed to not using the model. The response rate is the number of people that responded divided by the number of people that were mailed. The model results were used to rank individuals from the most likely to respond to the least likely to respond. Then the response rate for the top 50% was computed, and this number was then divided by the overall response rate, resulting in the lift of the model. For example, if the overall response rate is 1%, and we did not employ a model, and simply mailed a random sample of 50% of the total individuals, we would obviously expect a 1% response to our mailing. However, if we ranked people based on a model, and then mailed the top 50%, we would get a response rate larger than 1% if our model had any predictive power at all. The more predictive power the model has, the better our response rate should be. If the top 50% of individuals responded at 2%, and the overall response rate is 1%, then the lift would be 2.0, meaning the top 50% are twice as likely to respond compared to the overall average.

Along with lift, the area under a roc curve (AUC) has also become a common way to evaluate models [8]. However, lift and AUC both measure the same quantity, which is the difference between using a model and not using a model. Therefore, the results from using lift or AUC should be very similar. One study [8] confirmed this, finding lift and AUC correlated at 0.96 when comparing results from 2000 models.

One way that the results could be measured is to look at the lift of the sample that was used for training the model. This, however, is a rather poor measure of results because it is subject to overfitting. Therefore, a more realistic test is to measure the lift of a sample that was not used in modeling, referred to as the test data set. Mailings are sent out 24 times per year, so it is easy to simply train on a training sample and then test on a testing sample that was mailed at a later date. The test dataset that was used in the current analysis consisted of 237,699 observations with a response rate of 0.75%.

C. Analysis Techniques

Our general procedure for performing the SVM analyses follows. The SVM tool LIBSVM [4] was used for the analyses. LIBSVM is appealing because it offers tools that automatically search for optimal parameter settings to be used in SVMs, which simplifies the model construction substantially. The data was scaled using the scaling tool provided by LIBSVM. At the time of this study, the LIBSVM tool did not return the real decision values of the decision function, which were required for this study. Therefore the LIBSVM source code was altered to output decision values. Although the results are tested using the lift concept discussed earlier, actual model selection was performed by using five-fold cross validation.

The general procedure for performing logistic regression follows. First, as suggested by Sheppard [9], step-wise linear regression was used to select features. The model started with all features, and the procedure then removed any feature with a significance level greater than 0.001 at each step. A logistic regression analysis was then performed on the remaining features, and each feature with a significance level greater than 0.05 was removed.

The general procedure for using the data mining software package is as follows. The product that was used was "Affinium Model"® by the company "Unica"® [10]. The algorithms used in the current study included linear regression, logistic regression, CART [11], and neural networks [12]. The software automatically searches through

a number of possible values for the parameters used in the algorithms, while using cross-validation for model selection. The software also performs automatic pre-processing of data.

III. Results

Table I displays the method, features used, sample size, and lift for a number of analyses.

TABLE I
LIFT VALUES

Method	*Features used*	*N*	*Test set lift*
Svm	577 from original data set	4,250	1.332
Logistic regression	577 from original data set	250,994	1.314
Svm	Principal components	4,250	1.306
Svm	Principal components without squared variables	4,250	1.304
Logistic regression	Principal components	250,994	1.302
Data mining tool	577 from original data set	250,994	1.295
Svm	Logistic regression variables	250,994	1.285
Logistic regression	Principal components without squared variables	250,994	1.280
Svm	Logistic regression variables	4,250	1.257

No SVM with all variables and features were run because in these cases LIBSVM either halted unexpectedly or would not produce any indication of progress after 20 hours. The analysis was then attempted with SVMlight [13], but the same problems occurred. The size of the dataset was reduced by including all of the positive responses (n = 2,125) and a random sample of 2,125 negative responses. The opposite problem occurred with the data mining tool and logistic regression. With these two techniques overfitting occurred when the data sets were reduced. Therefore logistic regression and the data mining tool used the entire data set.

The grid search feature of LIBSVM was used to find the ideal values of the C and gamma parameters when using the radial basis function kernel in the SVM model. The C value is the familiar penalty parameter used with SVMs. The gamma parameter is a value that is used to scale the kernel so that the values do not get too large.

It is possible that with 577 features, the SVM analysis cannot separate the data well because the important variables get lost among the many unimportant ones. Therefore for one comparison, principal components analysis was used to reduce the number of features. This was done twice, once including the squared values and once not including the squared values. As seen in Table I, using principal components had a very small effect on the results. Using fewer variables did not improve the performance of the SVM.

Another way to reduce the number of variables is to include only those found to be predictive using logistic regression. It may be expected that if the SVM were only given the most important variables, it may perform better than it will with all of the variables. To test this hypothesis, an SVM analysis was performed with only the features that were selected by logistic regression. Having fewer features created an additional benefit. With fewer features, an SVM was able to be performed on all of the observations, which allowed for a comparison with an SVM with only 4,250 observations. The most important comparison is between the lift for the SVM with the full dataset (1.332) and the lift for the SVM with the variables selected by logistic regression (1.257). Compared to many of the other analyses that have been performed, this is a very large difference. It appears that reducing the amount of features, even if the most important ones remain, has a detrimental effect on SVMs. Next the lift for the set with all of the observations (1.285) was compared to the lift for the set with only 4,250 observations (1.257). Although the SVM with all observations did outperform the SVM with 4,250 observations, the difference was small. This is further evidence that SVMs can handle datasets with few observations, especially when compared to traditional methods like regression.

Although the grid search was used to find the ideal C and gamma parameters, other combinations of C values, gamma values, and kernel functions were also tested. Table II displays the five-fold cross validation accuracies that were achieved using a number of different parameters. The kernels that were tried were the radial basis function (Rbf) and polynomials with degree of 3 and 9. The first column of the table indicates whether the parameters used were from the grid search function of LIBSVM, the default LIBSVM parameters, or a custom combination. The percent correct is the percentage of correctly classified responses using five-fold cross validation. The last column displays the number of support vectors selected for each analysis.

TABLE II
SVM ANALYSES

All variables:				
Type	*Kernel*	*C*	*% Correct*	*Num. SVs*
Grid Search	Rbf	128	58.6	2,952
Default	Rbf	1	58.4	3,087
Custom	Rbf	1000	53.9	2,670
Custom	Rbf	0.1	57.6	3,330
Custom	Sigmoid	1	58.2	3,169
Custom	Poly., d = 3	1	58.5	3,086
Custom	Poly., d = 9	1	58.5	3,158
Principal components:				
Type	*Kernel*	*C*	*% Correct*	*Num. SVs*
Grid Search	Rbf	512	58.4	3,016
Default	Rbf	1	56.7	3,359
Custom	Poly., d = 3	1	49.7	3,389
Custom	Poly., d = 9	1	48.6	3,389
Principal components no squared variables:				
Type	*Kernel*	*C*	*% Correct*	*Num. SVs*
Grid Search	Rbf	512	57.7	3,059
Default	Rbf	1	57.1	3,282
Logistic regression variables:				
Type	*Kernel*	*C*	*% Correct*	*Num. SVs*
Grid Search	Rbf	128	58.6	2,836
Default	Rbf	1	58.0	2,883

For every test, the grid search was the most accurate, although in many cases it was not much more accurate than an analysis with the simple default parameters used by LIBSVM. In most cases the radial basis function led to the highest accuracy. Because the grid search was the most accurate, each SVM was built using the parameters from the grid search.

Overall, the results for the SVM are very impressive. With less data, the SVM did slightly better than the logistic regression and the data mining tool. It should also be noted that the SVM only took about 2 hours of total preparation and processing time, whereas logistic regression took about 4 hours because of all of the preparation time, and the data mining tool took about 33 hours because it created hundreds of models. The actual computer processing time of the SVM algorithm with default parameters and 4,250 observations was only 80 seconds.

One goal of the current study is to test if the decision function values can be loosely treated as probabilities. If they can be treated as probabilities, then we should be able to rank individuals based on their decision function value, and those with the highest values should be most likely to reply to the mailing. Table III displays observations ranked from least to most likely to be a member of the positive classification, grouped into ten deciles. As can be seen in the table, the decision values do an excellent job of ranking individuals. The correlation between the decile and the probability of responding is 0.97.

TABLE III
DECILES BASED ON DECISION FUNCTION VALUES

Decile	*Response rate*	*N*	*Sum of responses*
1	0.38%	23774	90
2	0.43%	23774	102
3	0.58%	23775	138
4	0.48%	23774	115
5	0.64%	23775	151
6	0.74%	23774	177
7	0.94%	23772	223
8	0.94%	23777	223
9	1.18%	23726	279
10	1.20%	23770	286

IV. CONCLUSIONS

Many different algorithms were employed in the current study, including completely automatic neural networks and CARTs, hand tuned logistic regression, and hand tuned SVMs. A number of different datasets, with different numbers of features, and different numbers of observations, were also employed. With all of these differences, one might expect that the results would vary greatly. However, it is quite striking how similar the results are. The lift for the data mining tool and SVM analyses only differed by 0.037, and the logistic regression differed from the SVM by only 0.018.

As can be seen in Table I, SVMs did slightly better than the other two methods in most cases, with three out of the four best performers being SVMs. SVMs were the only method that could handle a reduced data set; they were able to produce the best model with only 1.7% of the total observations. Although SVMs are typically used to classify observations into two classes, the current study indicated that they could be used to rank individuals from least to most likely to respond, indicating that SVMs can produce

values that can be loosely interpreted as probabilities. SVMs also did well with the simple, default parameters offered by the LIBSVM software, therefore creating the SVM models was very simple. Yet another benefit to using SVMs was that including 577 features did not produce any overfitting, even though the results of the logistic regression indicated that most of the features were probably irrelevant. Therefore, the extra step of selecting the most relevant features was not required for the SVM in the current study.

In conclusion, SVMs did an excellent job of ranking individuals for the current dataset. SVMs were able to do just as well as a complex, well-engineered data mining tool and the handcrafted analysis of a domain expert. It seems likely that in the future more data mining packages will begin to include SVMs. Although the results of this study appear interesting, it is important to note that they only apply to the current dataset. Further research could investigate whether the results of the current study replicate with other datasets. Further research could also use different methods to produce exact probabilities from SVMs, and could use the area under the ROC curve as an additional metric to evaluate the models.

REFERENCES

[1] B. Scholkopf, and A. J. Smola, *Learning with Kernels.* The MIT Press, Cambridge, 2002.

[2] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[3] S.W. Lee, and A. Verri, Eds. *SVM 2002*. Berlin, Heidelberg: Springer-Verlag, 2002 .

[4] C. Chang, and C. Lin (November 12, 2003). *LIBSVM : A library for support vector machines.* Software available at http://www.csie.ntu.edu.tw/~cjlin/LIBSVM

[5] J. Drish, (April 12, 2007). *Obtaining Calibrated Probability Estimates from Support Vector Machines.* Available: http://www-cse.ucsd.edu/users/jdrish/svm.pdf

[6] S. Ruping, "A simple method for estimating conditional probabilities in svms," in *LWA 2004 - Lernen - Wissensentdeckung - Adaptivitat.* A. Abecker, S. Bickel, U. Brefeld, I. Drost, N. Henze, O. Herden, M. Minor, T. Scheer, L. Stojanovic, and S. Weibelzahl, ed., Berlin: Humboldt-Universitat, 2004.

[7] C. Hsu, C. Chang, and C. Lin (November 12, 2003). *A Practical Guide to Support Vector Classification.* Available: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

[8] R.Caruana and A. Niculescu-Mizil. "Data mining in metric space: an empirical analysis of supervised learning performance criteria," in *Proceedings of the Tenth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining*. New York: ACM Press, 2004.

[9] D. Sheppard, Statistics and Modeling for Direct Marketers Seminar, New York, New York, 2000.

[10] Unica Corporation, *Unica Affinium Model Customer Segmenter Version 5.0*, 2001.

[11] L. Breiman, J. H. Friedman, R. A. Olshen and C.J. Stone, *Classification and regression trees.* Belmont, CA: Wadsworth International Group, 1984.

[12] C. M. Bishop, *Neural Networks for Pattern Recognition.* Oxford: Oxford University Press, 1995.

[13] T. Joachims, "SVMlight: Making large-Scale SVM Learning Practical," in *Advances in Kernel Methods - Support Vector Learning.* B. Schölkopf, C. Burges and A. Smola ed., Cambridge, MA: MIT-Press, 1999.