Leverage Sparse Information in Predictive Modeling *

Liang Xie
Countrywide Home Loans, Countrywide Bank, FSB

October 27, 2008

Abstract

This paper examines an innovative method to leverage information from a collection of sparse activities. By sparse, it means recorded activities among monitored subjects are rare, and a typical example is recorded customers' behavior on tagged web page. A customer may browse certain web pages during a long time interval, and each tagged web page gets hits only from a small portion of marketing eligible customers. The final Customer-WebHits matrix will be a sparse matrix.

Traditional statistical methods are not very effective in dealing with this kind of sparse data. Even though we can resort to techniques such as Principle Component Analysis (PCA) or variable clustering (VARCLUS) to reduce the number of variables, these unsupervised classification methods don't fully explore the relationship between all these variables and dependent variable, and more often than not their predictive power is not strong enough.

In this paper, we explore a variation of the Term Vector Space Model (TVSM) from Information Retrieval (IR) and demonstrate how it can provide maxmized predictive power in a binary outcome predictive modeling case. A simulated data having similar distribution of real data is generated and this variant algorithm is applied. We compare the lift chart from TVSM to those from PCA and VARCLUS. The result shows TVSM certainly holds its advantage.

1 Introduction

In one Response Modeling project, analyst wanted to utilize customers' behaviors, such as web browsing pattern and servicing call logs, to enchance predictive models. These signals are considered, by business intuition, to be strong indicator for customers' refinancing need in the near future. For example, if we found a customer repeatedly check the rate section in his online account, we know there is high likelihood this customer

*Draft version

is search for refinancing opportunities. If we can capture and analyze these signals and response to customers' need promptly, we are able to win the business among fierce competition.

These behavioral data are from two sources. The first source is certain web pages our IT department tagged in the online account section. When a customer log on to his online account, our IT system is able to record which pages he visited at what time. The second source is our servicing department, which record when a customer calls in and what the call is about. Usually a servicing call will be identified up to 4 standardized different purposes and recorded in our system.

Considering we observe 60 days history, and these behaviors happened at minutes level, the potential analytical work is heavy. Therefore, the analyst pre-processed the data in order to manage the analytical work load. If we call any of these recorded behavior as an signal, and group each signal in one week interval into one category, we will form a Customer-Category matrix, which will become our subject to be analyzed.

There are two challenges the analyst faced.

First, it is very difficult to use simple business rules to identify all qualified customers. For example, if a customer calls in last week and explciitly ask for payoff, then there is such an obvious signal we can pick up and response. But among over 60 different categories, only few of them have this significant effect, while most of them are not indicative at all by business intuition or priliminary data analysis. The analyst hopes to find out an identifiable pattern from some combination of all observed signals that is usable.

The second challenge from these data is that they are very sparse both in frequency and categories. Usually only a small portion of our 9million eligible portfolio customers will visit the tagged web pages in a 60 day time period, and when they do visit, only limited number of these pages will be actually clicked. For the same reason, not many customers will call in frequently regarding their mortgage servicing or telling our CSR they are looking for payoff or refinancing opportunities. Besides, the timing of any of these behaviors happened at irregular intervals over continuous time, hence all behaviors within one week are grouped into one category to simplify the analysis.

The resulting Customer-Category matrix is extremely sparse, as can be seen from Figure **??**, where empty cells are in white. Since this data is sorted by pre-defined classification groups, the area under the dashed lines are all observations with an Event, while those above the dashed lines are all non-Event observations. It is obvious that no immediately detectable differences by examining the graph.

2 Analysis

The data we are going to conduct analysis is a high dimensional sparse matrix, and we are trying to do a supervised classification based on some measurements derived from this matrix. There are several options.

With high dimensional matrix, statisticians usually turn to Singular Value Decomposition (SVD) or Principle Component Analysis (PCA) for

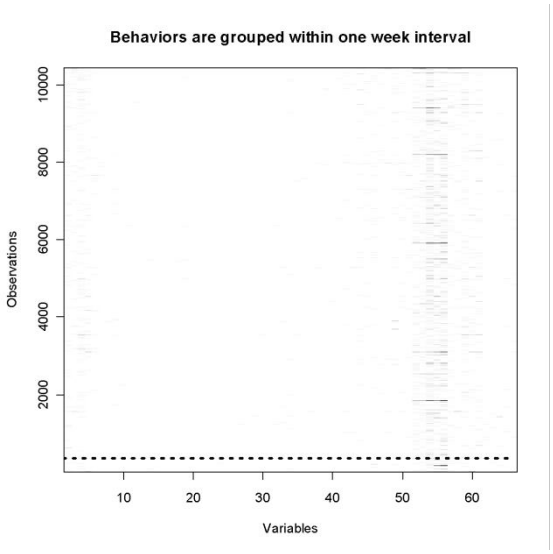

Figure 1.1: Customer-Category Matrix is Sparse

(1) Dimension Reduction, and (2) Feature Extraction. PCA can be seen as an SVD from the covariance matrix of original data, so that they will be discussed together. For reasons explained in the next section, we didn't choose SVD, but rather, in observation the fact that a high dimensional sparse Customer-Category matrix is just similar to a transposed Term-Document matrix in Text Mining field, we decided to adopt well developed methods from there.

While there are many models have been proposed, two most popular methods are classic Term Vector Space Model (cTVSM), which is based on norm vector space distance measurement, and Latent Semantic Index (LSI), which applied SVD on the Term-Document matrix. We will discuss cTVSM only.

2.1 Classic TVSM

Classic TVSM was developed in the 80's and is still an effective mehtod in Information Retrieval (IR) such as search engine, where cTVSM was used to match query, which is regarded as a "Document", to other documents stored in the system and the match is ranked by similarity coefficient calculated according to cTVSM algorithm. The Algorithm follows steps below:

Let Query "D_q" consists of terms $\{w_{qi}\}$, where $i = 1 \cdots N$ and denote the documents in our system as a collection of "D_k" $k = 1 \cdots N$, consists of terms $\{w_{ki}\}$. In observe the sparse nature of this matix, for many "D_k", many of w_{ki} are NULL value.

The we calculate a weight system called TFIDF based on probability

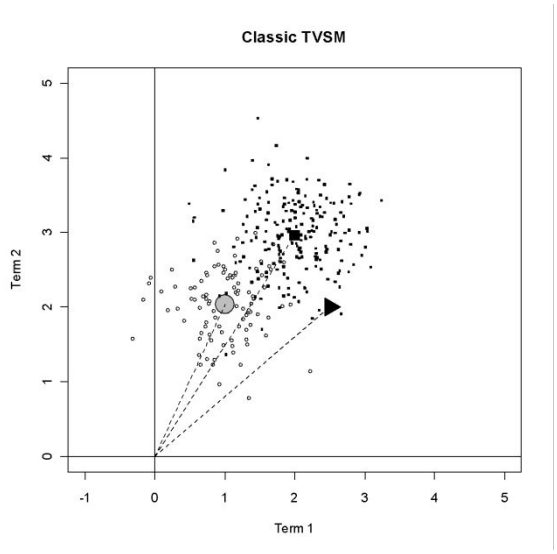

Figure 2.1: Illustration of Nearest Neighboring Applied on cTVSM

model:

$$\delta_{ji} = TF_{ji} \cdot IDF_i$$

where

$$\begin{aligned} TF_{ji} &= \text{Term Frequency} = \text{Number of Term i in Document j} \quad (1) \\ IDF_i &= \text{Inverse Document Frequncy} \\ &= \text{logit of probability a Document does not have term i} \\ &= \log((N-n)/n) \end{aligned}$$

The the similarity coefficient between Query and Document k can be obtained as:

$$\begin{aligned} SIMI_{qk} &= \frac{|D_q \cdot D_k|}{|D_q| \cdot |D_k|} \quad (2) \\ &= \frac{\sqrt{\sum_i \delta_{qi} \cdot \delta_{ki}}}{\sqrt{\sum_i \delta_{qi}^2 \cdot \sum_i \delta_{ki}^2}} \\ &= \cos(D_q, D_k) \end{aligned}$$

Note how close the similarity coefficient mimic correlation coefficient.

Usually, in IR applications, such as Search Engine, we have only one or two queries with large amount of "D_k" in system, which is not the case in our supervised classification application, where we have a large amount of Queries to be classified into two or more pre-defined classes. As an heuristic method, cosnider figure **??**.

In the figure, we showed a case where we applied Nearest-Neighboring model on top of cTVSM. For simplicity, consider we have only two pre-defined classes, Event and Non-Event. If we apply cTVSM directly, we

are going to compare subjects in our scoring data set to each of subject in the modeling data set, and then we can rank subjects in the scoring data by either:

1. How many "Event" class subjects are ranked in top 100 or top 1000 by similarity coefficient;
2. Weighted sum of $SIMI_{qk}$ for each Subject_q in scoring data and each Subject_k in modeling data;

However, both of these methods have disadvantage that they are computational very intensive. In typical predictive modeling and scoring application in the industry, we have millions of subjects to be scored and in a typical modeling sample, thousands of subjects, resulting in billions of similarity computations.

But by figure 2, we thought that we can turn above computation into a Nearest-Neighboring problem, which is widely used in classification and clustering applications, besides it requires no distributional assumption but some distance measurement which is readily provided by the similarity coefficient. Applying the idea of K-means Nearest Neighboring, we instead compute the similarity coefficient of each subject in the scoring data set to the mean position of each pre-defined class.

There is still another difficulty in comparing the final similarity measurement. For each subject in the scoring data, we will have K similarity coefficient for a K-group classification problem. By examine the formula of similarity coefficient, we found that if a subject has fewer activities recorded, that is more sparse row vector on the Customer-Category matrix, the resulting similarity coefficients for all K groups will be smaller compared to another subject with more dense row vector. In order to adjust this bias, or say to standardize the coefficients across K-groups, we take the the ratio of those coefficients, and the baseline group will be in the denumerator, which makes the final result similar to an odds ratio.

In a binary classification application, the final result is a one dimensional value that is positively correlated with odds of event outcome. This final result can them be put into a regression analysis as independent variable. Our analysis shows this variable has good predictive power and has low correlation with other quantitative independent variables in the model, which are very desirable.

2.2 SVD based Method

SVD has a very nice property that makes it attractive to modeler for two applications: 1. Feature Extraction; 2. Dimension Reduction. Let X be $m \times n$ matrix, SVD is:

$$X = U\Sigma V^T$$

where U is a $m \times n$ orthogonal matrix consists of left singular vectors, and V is a $n \times n$ square orthogonal matrix with rank $r \leq n$ consists of right singular vectors. S is a diagonal matrix with singular values. The left singular vectors can be seen as a profile expression of row vectors of X while the right singular vectors can be seen as the profile expression of the column vectors of X. With profile expression, analysts are able to extract

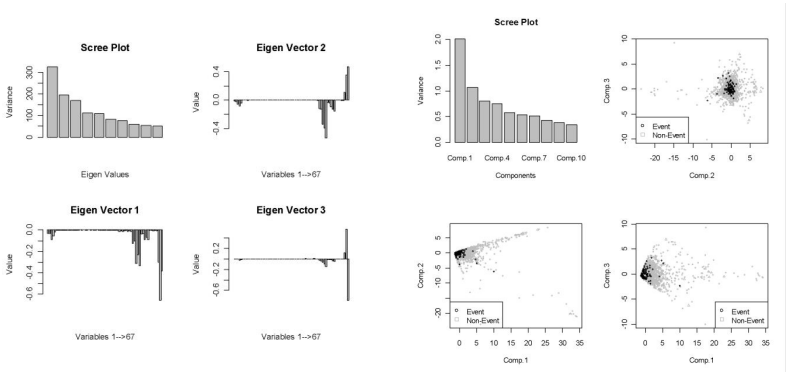

Figure 2.2: Left: Disturbe SVD with hypothetical high variance variables; Right: PCA result on sample data

underlying features of columns, a.k.a variables, of X, as well as underlying groups of rows, a.k.a observations, of X. Therefore by examining the joint distribution of observations along the top profile expression vectors, analysts are able to learn potentially identifiable patterns of variables or subjects.

Another nice feature of SVD is that it provides rank-k $k \leq n$ approximation to the original data as:

$$X^{(k)} = \sum_{i}^{k} s_i \cdot \boldsymbol{u}_i \cdot \boldsymbol{v}_i^t \tag{3}$$

which is used for Dimension Reduction.

Because SVD works on variance and correlation among variables, this imposes a disadvantage when we have sparse data with greater heterogeneity in variance across variables. If some variables have significantly higher variability than the others, the eigen vectors will be dominanted by these variables. This case is illustrated in left panel of Figure **??**. In the following example, we append to our real data 3 random variable with poisson distribution. While the original data is sparse, these three variables are dense. We can see that in first three right singular vectors, which gives profile expression for variables, all these three poisson variables show high value and stand out in the profile, but by design they are not predictive at all for our event. Also note that the variables numbered 1 to 10 and those numbered 50 to 60 all have more dense data other the other variables except for those hypothetical ones, and they, too, show high profile among the right singular vectors.

Our analysis shows scores from PCA or left singular vectors from SVD provide some discrimination power among events and non-events. From right panel of Figure **??**, we see the events are concentrated in certain areas of the space of component 1, component 2 and component 3. But this figure is misleading for predictive modeling. By checking the histogram of events and non-events along side of the first three left singular vectors, we found events and non-events have proportional distribution across the

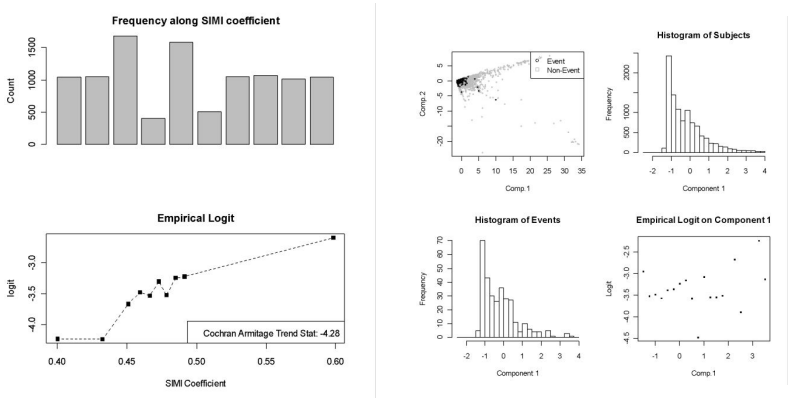

Figure 2.3: Left: Empirical Logit by SIMI coefficient; Right: Distribution and Empirical Logit by Component 1

value range, therefore these score values can't provide good predictive power for modeling. In the right panel of Figure 2.3, we provide the historgram of events and non-events along side of the first left Principle scores. We can see that the proportions of Events in each segment of score value intervals are very not statistically different, which can be seen from the empirical logit.

In contrast, the empirical logit by SIMI coefficient from cTVSM shows promising effect for predictive modeling. Cochran-Armitage Trend Statistics is -4.28 with p-value < 0.00001.

3 Conclusion

In this paper, an Nearest-Neighboring method based on distance measurement from Classic TVSM is applied to calculate a similarity value which in turn is put into the predictive model as an independent variable. This variables shows good predictive power that is not available from directly using those variables or using scores from PCA or SVD, therefore would otherwise been throw away. The reason PCA or SVD doesn't work well in this type of application is discussed.

4 Reference

1. Friedman, Jerome, Hastie, Trevor, Tibshirani, Robert; *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer Series in Statistics, 2001

2. Garcia, E; *A Linear Algebra Approach to the Vector Space Model*, Manuscript, 2006

5 Contact Information

Liang Xie
Countrwyide Home Loans
7105 Corporate Drive
Plano, TX 75024

Work phone: 972-526-4224
E-mail: xie1978@yahoo.com
Web: www.linkedin.com/in/liangxie