Retail Sales Prediction and Item Recommendations Using Customer Demographics at Store Level

Michael Giering
M.Giering@gmail.com

ABSTRACT

This paper outlines a retail sales prediction and product recommendation system that was implemented for a chain of retail stores. The relative importance of consumer demographic characteristics for accurately modeling the sales of each customer type are derived and implemented in the model. Data consisted of daily sales information for 600 products at the store level, broken out over a set of non-overlapping customer types. A recommender system was built based on a fast online thin Singular Value Decomposition. It is shown that modeling data at a finer level of detail by clustering across customer types and demographics yields improved performance compared to a single aggregate model built for the entire dataset. Details of the system implementation are described and practical issues that arise in such real-world applications are discussed. Preliminary results from test stores over a one-year period indicate that the system resulted in significantly increased sales and improved efficiencies. A brief overview of how the primary methods discussed here were extended to a much larger data set is given to confirm and illustrate the scalability of this approach.

Keywords

Recommender system, SVD, PLSA, IISVD, clustering, retail data mining

1. INTRODUCTION

A major challenge for large retailers is to address the needs of the consumers more effectively on a local level, while maintaining the efficiencies of central distribution. As the demand for mass customization by consumers grows, methods focused on store level optimization increase in value. In retail data mining the ability to accurately predict expected sales translates directly into several high impact and implementable actions. Common applications resting on accurate item sales modeling include product assortment optimization, sales anomaly detection, customer segment targeting, new product distribution and new store stocking [6,7].

Supplemental data allows one to partition the data set in multiple ways and build separate models for each partition. A straight forward method to deal with this multiplicity of models is to build a rolled-up model that is a linear combination of all the sub-models. The problem then is to determine the mixing weights of the sub-models, where each sub-model corresponds to a different partition. The mixing proportions can be interpreted as a proxy for the importance of customer characteristics and business setting contexts in accurately modeling sales at store level.

There were major constraints on the methods chosen for this work. We consciously decided that the value of methods that are more easily communicated to the client at any stage of the analysis outweighed marginal methodological gains.

We designed our approach with the expectation, that in the future, the data we received would be continuously updated. Given the updating requirements and the size of the data sets involved, it was decided to utilize fast online methods. These methods underlie the more processing intensive facets of our approach.

The specific applications of the sales prediction models that we focus on in this paper are a store level product recommender, an outlier detection model and a new store stocking tool. The paper is organized as follows: In section 2 we describe our data and the primary goals of the project. Section 3 contains our mathematical formulation and description of the baseline naïve approach. Section 4 expands on the details of the methodology. Section 5 describes both experimental results and some realized business impacts. Section 6 presents a discussion of the results and practical technical issues that occurred in the course of the project. The scalability of this approach is illustrated in a brief discussion of subsequent work completed on significantly larger data sets. Finally, Section 7 provides conclusions and avenues for future work.

2. Retail Case Study – Data and Goals

In this paper we present a methodology that was implemented for a retailer in a test set of $|S| = 80$ stores in early 2007. For each store we have 18 months of daily sales figures for $|U| = 600$ items. The sales figures are broken out across a set of N non-overlapping customer types. Let us denote this set by $\mathbf{T} = \{T_i \mid i \in 1\ldots N\}$. In addition we have descriptive customer information along various demographic descriptors (such as age, wealth and income) from both the retailer and publicly available data sets. We denote this set of descriptors by $\mathbf{F} = \{F_j \mid j \in 1\ldots L\}$. For every demographic descriptor F_j and customer type T_i, we have the customer base broken out as percentages across all the bins of the demographic descriptor (e.g. age split into 10 year wide bins) for every store. Let us denote this $|S|$-by-nb_j matrix as $\mathbf{D}_{ji}$, where nb_j is the number of bins along demographic descriptor F_j.[1] Figure 1 shows a schematic description of the data.

[1] We also have firmographic information available that provides contextual business information about each store. An example firmographic factoid would be, "there exist 3 shopping centers and 10 gas stations within a 5 mile radius." For the sake of our discussion, we can consider this firmographic matrix as part of the demographic data.

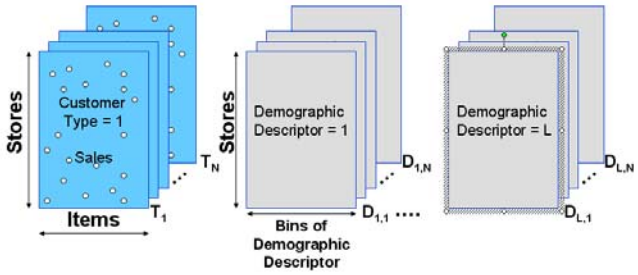

Figure 1. A schematic representation of the retail data and demographic data sets.

The product selection in each store varied significantly and hence there were missing values in our data set that corresponded to items that were not available in all stores.. This worked to our advantage in forecasting sales of items not currently carried in a particular store.

The primary goal of the work was to build a store level modeling method that could be utilized in two ways: (1) as a product recommender for optimizing product selection over product universe U of the retailer. (2) as a screen for anomalous sales patterns at store level.

A product recommender is an implementation of a recommender system similar to those used by movie rating sites like Netflix, but for retail items. The purpose is to recommend items that will optimize store level sales. Mathematically this can be viewed as predicting sales for items for which data was missing.

Anomaly detection is an effective screening tool to identify statistically improbable sales occurrences. Flagging these occurrences can help analyze performance issues with individual stores or items.

Due to the proprietary nature of the data, we are limited in the degree of detail to which we can discuss data specific issues.

2.1 Statistical Overview of the Data Set

For all of the challenges faced with large sales data sets, there are two distinct advantages. The distributions of the data are well defined. The sheer volume of data provides enough statistical support for modeling even when partitioned into many subsets. Our sales data for individual items, item subcategories and store level sales were all surprisingly good fits to log-normal distributions. Examples are shown in Figure 2 below.

The SVD methodology's underlying assumption of normally distributed data is reasonably well suited for analysis and prediction of the logarithm of the sales data. In addition, sales outliers which by definition are in the tails of the distribution are frequently removed in the data preparation phase of the work.

For the sake of simplicity we discuss the aggregate months of data and reserving the initial 6 months of data for out of sample verification. The only limitations on the width of the data time window are that enough sales data is collected to maintain necessary statistical support for the models. Smaller time windows would facilitate more seasonal aspects of sales.

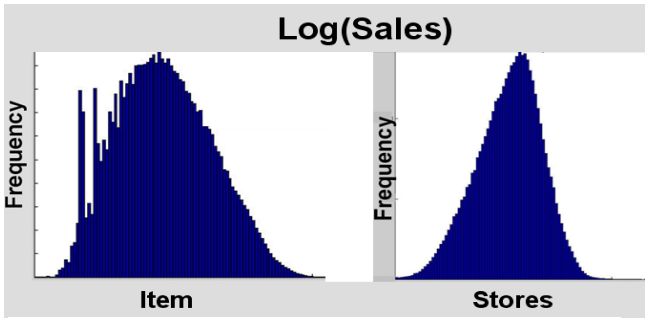

Figure 2. Examples of log normal nature of daily item level sales for a particular store [left panel] and daily aggregate sales across items by store [Right panel].

3. Mathematical Formulation

The mathematical approach underlying both the product recommender and the outlier analysis tools is a high speed reduced rank SVD [1,2]. Reduced rank SVD imputation is a well known and often used technique in data mining. I refer the reader to [1,3,4] for more details.

Given a data matrix X the SVD provides a bilinear factoring of X,

$$X_{m\,x\,n} = U_{m\,x\,r}\; diag(S_{r\,x\,1})\; V^{T}_{r\,x\,n} \quad | \quad r < \min(m,n) \qquad (1)$$

where **U** and **V** are unitary orthogonal matrices whose columns are a linear basis for the columns and rows of X respectively. When X represents sales data with missing values, **U**, **S** and **V** are calculated using only the known values within X. These matrices are used to construct a rank-r reconstruction of the data. The entries of this rank-r reconstruction are used as estimates of values that were missing from the original data set.

3.1 Naïve Baseline Methodology

We define our baseline naïve model as the application of our reduced rank SVD method on the total sales data across all customer types for each store and item. In this model, no demographic or customer type information is utilized. Our metric of accuracy is the r-squared value of the known data values vs. the predicted sales values.

Our expectation is that when we model all of the sales data at once, the reduced rank will be too general, causing over and under fitting of some portions of the data set.

3.2 Customer Type Models

There are three levels at which we can model the sales data. As mentioned in the previous section, we can build a single model for the entire sales data that has been rolled up across all customer types.

At a finer grain we can model the sales data for each customer type separately. In this case, for each customer type T_j, we have an |S| x |U| table of sales per item for each store (see Figure 1). Since this data is non-overlapping, separate models M_j constructed for each customer type T_j can simply be combined to give the total sales model M,

$$M = \sum_{j=1..N} M_j \,. \quad (2)$$

Note that no demographic data is utilized in the construction of any M_j.

In the final level, we make use of the demographic data that is available for each customer type. For a given pairing $Q_{jk} = \{ T_j , F_k \}$ of customer type T_j and demographic descriptor F_k, the demographic data can be represented as a matrix $\mathbf{D}_{jk}$. We can use $\mathbf{D}_{jk}$ to obtain a non-overlapping partitioning of the stores. The sales data for each partition can be modeled separately. Let M_{jk}^{P} denote the model for sales in the p^{th} partitioning of stores where the sales data corresponds to sales for customer type T_j and where the store partitioning is based on demographic descriptor F_k. To obtain a model for all the stores we roll up models for each partition together as,

$$M_{jk} = \bigcup_{p} M_{jk}^{p} \,. \quad (3)$$

For a given T_j , notice that all of the models M_{j1} , … , M_{jL} are modeling the same set of sales data. There is no reason *a priori* to expect the partitioning of any F_k to improve or hurt the performance of our total sales model. The problem therefore reduces to finding the optimal mixing weights w_{jk} for each Q_{jk}. The model for customer type T_j can then be written as

$$M_j = \sum_{k=1..N} w_{jk} M_{jk} \,. \quad (4)$$

4. Methodology

The primary steps for constructing the model for a particular customer type can be summarized as:

- Partitioning of the stores based on demographic descriptors.
    - For a given demographic descriptor, modeling sales in each of the partitions.
- Calculation of the demographic mixing weights w_{jk}
- Creation of store level sales model.

We now walk through each of these steps in detail.

4.1 Partitioning

For each Q_{jk} we can segment the stores into a small number of subsets. For each demographic descriptor and a particular store, there exists a vector of information. For example, the age demographic information contains age-binned counts of store customers and of the general population within a given distance of the store. The age bins are further split out by gender. These vectors ranged from 10 - 115 dimensions. Similar vectors existed for each demographic descriptor and store.

The store clustering (partitioning) was carried out by a combination of three methods. The reason was to minimize the bias that may occur in any single method. After clustering with all three methods, a simple voting scheme was used to assign a cluster to each store.

The clustering methods were:

1. K-means clustering. This is a common and robust method of clustering, though not completely deterministic.
2. Correlation clustering. PCA was performed on the (|S| x |S|) covariance matrix of the demographic vectors across stores. The PCA coefficients were clustered via K-means.
3. PLSA clustering. Probabilistic Latent Semantic Analysis was used to cluster by generating a latent feature for each of our p-clusters desired. PLSA generated an |S| dimension feature for each latent variable. Each store was assigned to the cluster for which it had the highest probability.

4.2 Imputation of Retail Sales

Here we describe how we use the SVD imputation procedure to build the model M_{jk}^{P} .

Throughout our work, there is one subjective SVD parameter to be defined. In a reduced rank SVD, a parameter defining the maximum percentage of the total data variance that the model must capture from the training set, is chosen. Knowing that the noise level in retail data easily exceeds 15%, we set this parameter to 85%. This constrains the maximum possible rank in all models.

The mean-squared errors for each rank were estimated using a cross validation method and stepping through the possible range of modeling ranks,. The plot below shows how this produces a bias variance curve and the optimal rank chosen.

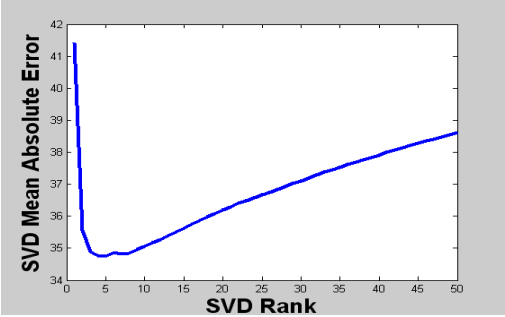

Figure 3. The optimal rank to model data at is determined empirically through cross validation.

In our implementation rank optimization was defined as the minimum number of singular vectors necessary to minimize the absolute error and capture less than 85% of the variance in the data set.

Using the optimal rank derived above, imputation of missing data or existing data is carried out by the reduced rank SVD model [1,2].

Having partitioned and chosen the appropriate modeling rank for each demographic descriptor and store partition, we are now in a position to roll up the partition models to obtain M_{jk}.

4.3 Demographic Mixing Weights

A priori we have no reason to favor the weighting of any single demographic descriptor. Given customer type T_j, our final model will be a weighted sum across demographic descriptors. The set of mixture weights w_{jk} varies for each customer type.

The mixture weights were calculated using a straightforward multi-linear regression method that is equivalent to solving equation (4) for w_{jk} with fixed j to minimize mean squared error.

There are advantages and disadvantages to this multi-linear approach that we should address. A clear disadvantage of this approach is that the solution is path dependent and not necessarily unique. The advantage of this approach is that the results were easy to convey to our client. From a practical perspective, the uniqueness of the solution is a non-issue for prediction. Two sets of weights producing identical results are equally valid. In practice two highly correlated demographic variables are easy to identify and non-uniqueness concerns could be explored in detail. This was not an issue for us.

For the purpose of effectively conveying the information, we express each demographic descriptor's regression coefficient as a percentage of its influence on the model. Though not very useful mathematically, it does enable us to display the demographic drivers of modeling accuracy for each customer type.

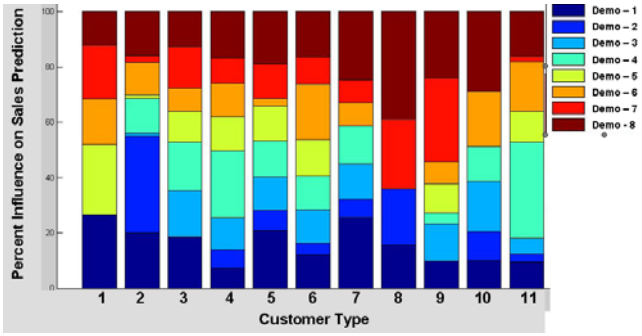

Figure 4. The relative influence of each demographic descriptor is displayed for each customer type.

4.4 Product Recommender

The store level product recommender serves several functions. It provides a means of comparison between the sales of items currently carried in a store to the expected sales of other items in the retailer's product universe U.

When looking at total store sales, this information is primarily used for optimization of product selection. For a given store, the actual sales or an imputed sales number for all items in U is constructed. Sorting this list by total sales and displaying the item descriptions allows an easy comparison of items. A simple color coding allows people to quickly see items that should be brought into the product mix at store level. For small numbers of items and stores this can even be implemented in an excel spreadsheet where columns correspond to stores and rows to items.

It should be noted that the imputed values for potential sales of items not currently carried do not account for the splitting of sales between competing items. Introduction of an item, similar to an existing item may simply cannibalize sales. Trying to estimate inter-item cannibalism would introduce an additional source of prediction error. The decision not to include it was conscious and made for ease of comparison and accuracy.

Once we have the mixing weights w_{jk}, we have a model for each customer type expressed in equation (4) .

For each store, one can obtain the likely distribution of items not carried currently across customer types T_j. This information can be used to optimize product mix with additional sensitivity to customer types.

In practice, items with very small distribution and sales do not have enough statistical support to model accurately and are rarely of serious concern to retailers. Core items with consistent and large sales are distributed almost uniformly. The best opportunities for large business impacts lie in new items and mid-tier items with moderate distributions.

4.5 Outlier Analysis

Outlier analysis is the application of the imputation technique to known sales values, by comparing the known sales values to the values estimated by our sales imputation model .

Iterating through the store-item sales values, we zero out each non-zero value and impute the expected value. Because of the normality assumption underlying the SVD methodology, we can express outliers from the imputed SVD model results in probabilistic terms. The z-value for the predicted sales for an individual item at store level can be written as

$$Z = \sum_j \sum_k w_{jk} Z_{jk} \qquad (5)$$

where Z is the z-value, w_{jk} are the mixing weights and $\mathbf{Z}_{jk}$ is the z-value of the predicted sales value for the pairing Q_{jk}. The outlier values can be ranked from most to least likely to occur.

Because the constituent segments of the data were modeled at differing ranks, the aggregated values are more accurate and the outlier probabilities are more reflective of store level implementation issues, rather than methodological issues of over-fitting and under-fitting.

Looking at the outliers by customer type, we gain additional insights into whether outliers are being caused by store level implementation issues or the shopping patterns of customer sub segments.

5. Results

In this section we present both experimental and applied results. We first address the experimental results and the question of measuring the model output vs. the baseline model.

We ran our model on the 6 months of out of sample data. Imputing the known sales values on store by items total sales data, we produced linear regressions of the known vs. expected values.

We repeated this for the naïve model without any customer type or demographic segmentation. For each model we calculated the r-squared error.

The ratio of the r-squared error of our model to the r-squared of the naïve model by store ranged from .3 to .68 as is shown in figure 5. The same comparison at item level ranged from .17 to 1.

For example a ratio of .25 implies that the r-squared of our model is 4 times lower than the naïve model. Of the full product universe roughly 200 can be considered core items of high or moderate distribution. These are the products that benefited the most from our approach.

When the same test was run across differing customer types, the results varied significantly. Customer types with large shopping volumes showed greater improvement. This is not surprising given their high level of statistical support.

A practical issue of the validity occurs because the training and out of sample data were over different time frames. This was a result of the data we were given. We felt it necessary to work on at least one full year of data as a starting point in our work. Looking at prediction error against the time of year showed no significant deviation in results.

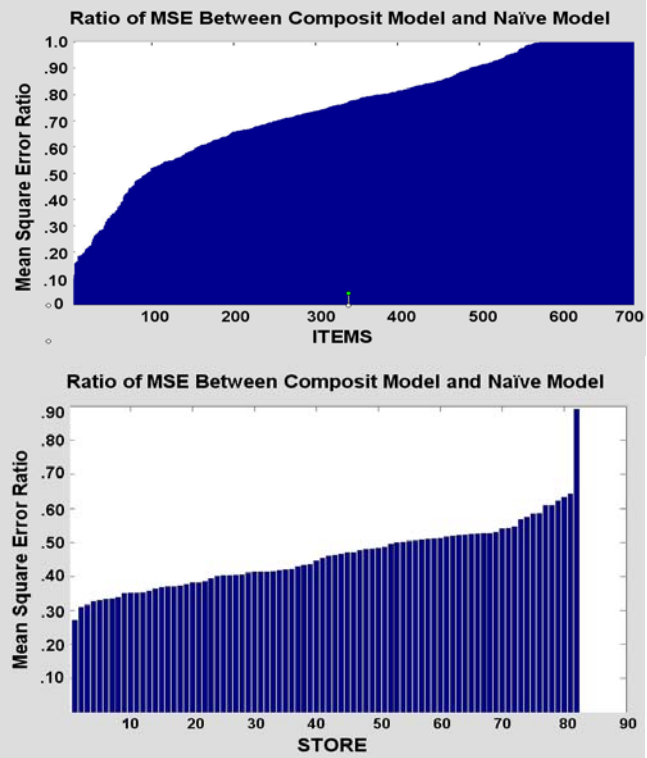

Figure 5. The ratio of the mean squared error of our model vs the naïve model are graphed by store {Top] and by item [Bottom].

5.1 Business Results

Though more difficult to directly quantify, this work has delivered an exceptional return on investment (ROI). A conservative calculation of the ROI of this project places its value between 140% - 185%. Because of the proprietary nature of the work, we are limited in what type of information we can disseminate.

Choosing the two customer types CT = { T_1 , T_2 } that made up the largest proportion of total sales, we identified the stores with strong CT sales. The store level product recommendations for CT were used to emphasize a product mix more in line with these customers. The displays were adjusted by grouping items of interest to CT more closely.

The recommendations for CT were implemented in the test stores. These stores showed gains in sales of several percent in excess of the non-test stores over a 1 year period. There were several times the number of non-test stores than test-stores.

One result of this work is that the methods employed are being widely adopted by our client.

Aside from direct tangible monetary impact, this work has generated a large amount of soft benefits. We are now seen by the retailer as a crucial partner that is actively engaged in growing their sales. An informal survey of experts in the field have conservatively placed the value of this new role between 2-3% of our current market share. This is an annually recurring benefit.

6. Discussion

The primary deliverables in this project were the store level item recommenders and the outlier analysis. If a measure of product margins had been available, the ranked listings of the product recommender could have been improved further. Though it sounds straight forward, true measures of product margins are convoluted by industry practices.

6.1 Scaling to Household Level

This section describes an extension the analytic techniques described thus far to a subsequent project. The purpose is to illustrate how our approach scaled in a significantly larger application.

Our goal was to carry out a similar analysis on 16 months of retail data receipts from 600 stores with 1.2MM items. These 1.7 billion sales transactions were tagged by 2.7MM unique household ID's.

Customer types were derived empirically through a combination of semantic analysis and standard clustering methods on subcategory sales.

In this application the customer desired a measure of unmet spending potential for each household across the more than 150 item subcategories.

Analogous to the earlier work, the sales data being modeled can be envisioned as a |*Household*|-by-|*Subcategory*| matrix. This is an increase in the length of the data set by four orders of magnitude from our previous work while the magnitude of the remaining dimension remained unchanged. Missing values are imputed as discussed in Section 4.2.

For one piece of this analysis household spending across subcategories was normalized to reflect the percentage of spending taking place in each subcategory. This is done to reduce sales differences due to the size of the household.

We then removed 5% of the known values and imputed them. By repeating this process until all known values were imputed we had created a dense data set of predictions to complement the actual data. The assurance of normally distributed errors enabled us to represent the error between known and modeled data as z-scores.

The z-scores are a simultaneous measure of how household shopping behaviors compare to each other as well as how characteristic the spend distribution across subcategories is. This is particularly useful for "hole in the basket" type analysis.

A typical question in such an approach is conditional. How do a subset of customers that purchase items $I \mid I \subset U$, the product universe, behave towards items $K \mid K \subset U$ and $K \cap I = 0$.

Significant differences from expectations are interpreted as those with a higher likelihood of being converted towards the mean behavior. Identifying unusually strong or weak households for more detailed analysis offers a rich variety of analytic directions in itself.

By creating the z-scored sales residues relative to expectations for each subcategory, we can explore these questions with fast sorting algorithms. More generally we can explore the household spending profiles in their entirety and their over/under spending across subcategories.

This work confirmed to us that the methodology we had designed scaled well to over four orders of magnitude on real world data. A more detailed account of this will be the subject of subsequent paper.

6.2 Practical Technology Issues

The fast online implementation of iterative SVD that we used was developed by Matt Brand [1,2]. The online implementation of Iterative Imputative SVD (IISVD) is crucial for speed necessary to carry out the analysis on streaming or periodically changing data.

Though less critical, the high speed implementations of SVD imputation routines were key to completing the extremely large number of iterations performed in this project. We initially tried other implementations of these algorithms but were unable to attain the necessary speed and computational efficiency necessary to do the work.

While the initial database setup, data cleansing and table generation was performed on a dedicated server, the model generation described in this paper was carried out in Matlab on a Dell XPS laptop. The basic technical specifications are: Windows XP Professional, Intel(R) core 2 CPU, T7600 @ 2.33GHz., 2GB of RAM, 110 GB hard drive.

7. Conclusions and Future Work

The main goals in this paper were to deliver a model that took advantage of four distinct types of data to improve sales forecasting for a large retailer.

The data were daily store level sales data broken out by item and retailer-defined customer types. We also utilized retailer provided store level demographic data, publicly available regional demographics and firmographics.

The model we built was implemented both as a product recommender and as an outlier analysis tool. The experimental predictive accuracy was 1.5-5 times greater for the items of interest as measured by r-squared error statistics. From a business perspective we have seen a large and directly measurable impact from this project in economic terms. Lastly we have seen a great deal of "soft" value created as a result of working closely with the retailer.

A straightforward application of this work is in the stocking of new store locations. Given the demographic information of the population surrounding a new store location, we can impute the expected sales for all items in the retailer's product universe. This in turn can be used as an effective new store stocking tool.

As we briefly discussed in Section 6.1, subsequent work that we performed confirmed that this approach was capable of scaling to data sets that were four orders of magnitude larger as we shifted from the imputation of store sales to individual household spending.

When customer types are not pre-defined by the retailer, defining customer categories can be carried out in many ways. The most common way is very time consuming and subjective. People choose a type of customer and then define a shopping behavior that they believe corresponds to it.

An alternative method is to generate semantic features from the data corresponding to various shopping behaviors. This was the primary segmentation method used for the work discussed in Section 6.1. We are currently exploring ways to combine both of these methods. The size of the data sets can be prohibitive for some generative methods and we are attempting to extend current online semantic analysis methods.

8. ACKNOWLEDGMENTS

We would like to acknowledge Gil Jeffer for his database expertise and provocative data exploration that made much of this work possible. Thanks also goes out to Madhu Shashanka for his assistance and helpful discussions during preparation of this paper.

9. REFERENCES

[1] Brand, M. 2002. Incremental Singular Value Decomposition of Uncertain Data with Missing Values. In *Proceedings of the 7th European Conference on Computer Vision-Part I* Springer-Verlag, London, 707-720.

[2] Brand, M. 2003. Fast online svd revisions for lightweight recommender systems. In SIAM Intl conf on DM.

[3] Chandrasekaran, S., Manjunath, B. S., Wang, Y. F., Winkeler, J., and Zhang, H. 1997. An eigenspace update algorithm for image analysis. *Graph. Models Image Process.* 59, 5, 321-332.

[4] Gu, M., and Eisenstat S.C. 1994. A Stable and Fast Algorithm for Updating and Singular Value Decomposition, Technical Report YALE/DCS/RR-996, Yale University.

[5] Hofmann, T. 2001. Unsupervised Learning by Probabilistic Latent Semantic Analysis. *Mach. Learn.* 42, 1-2, 177-196.

[6] Schafer, J. B., Konstan, J. A., and Riedl, J. 2001. E-Commerce Recommendation Applications. *Data Min. Knowl. Discov.* 5, 1-2 (Jan. 2001), 115-153.

[7] Zhang, X., Edwards, J., and Harding, J. 2007. Personalised online sales using web usage data mining. Comput. Ind. 58, 8-9, 772-782