Detecting Seasonal Trends and Cluster Motion Visualization for Very High Dimensional Transactional Data

Gunjan K Gupta and Joydeep Ghosh

1 Introduction

Real life transactional data often poses challenges such as very large size, high dimensionality, skewed distribution, sparsity, seasonal variations and market-drift or migration [1, 2]. Most studies have taken a static view of the data while making predictions about a customer's buying behavior, market segmentation, etc. [3, 4]. A notable exception is recent work on temporal association rule mining, dealing with incremental characteristics and change, for example, see [5, 6]. This paper focusses on the problem of segmenting customers visiting a rapidly growing e-tailer. The segments are dynamic and seasonal, so preprocessing and trend characterization is key. We use a real-life data belonging to an e-commerce business and referred to as **Horizon data** in this paper, provided by KD1[1] (since then acquired by Net Perceptions) to illustrate the issues. In Section 2, the Horizon data is summarized. Section 3 quantifies market migration for choosing the appropriate period of data. Based on seasonal variations in purchasing behavior, a novel seasonality detection and partitioning scheme is described. Some of the market migration and oscillation results on Horizon data are also presented. Section 4 describes a new concept called Cluster Space for converting this high dimensional ($> 10,000$) data into a continuous low dimensional space using a graph based clustering called VBACC [7] on the seasonally partitioned data. Motion detection and visualization schemes are introduced, and some interesting trends found in the Horizon data are described.

[1] http://www.kd1.com

A note on Market vs. Customer Migration: For our discussion we define market migration as a non-periodic change in the product purchase distribution for all the customers. Customer migration is another trend in which the purchase profile of a customer changes with time and may or may not be periodic over long periods. It is important to note that although a customer might migrate to a new set of products with time, new customers might replace him. Thus, it is possible to have substantial customer migration without corresponding market migration. A model is meaningful only for the period for which the market profile is reasonably stable, i.e the market migration is not substantial. In such a period it is useful to look at customer migration since the customer migration often happens faster than market migration.

2 Data Description

For the rest of this paper, we represent the Horizon data $\mathbf{T}$ in the following form; Let $\mathbf{C}$ represent the set of all customers, $\mathbf{P}$ represent the set of all products and $\mathbf{J}$ = $[J_{min}, J_{max}]$ be the range of Julian days for the transactions. Each row of table $\mathbf{T}$ represents a transaction $\mathbf{T_i} = (C_i\ P_i\ J_i\ R_i)$, where $C_i \in \mathbf{C}$, $P_i \in \mathbf{P}$, $J_i \in \mathbf{J}$ and R_i is the amount in dollars spent by customer C_i on product P_i on day J_i.

For the Horizon data $|\mathbf{T}|$ = 394,917, $|\mathbf{P}|$ = 10,842, $|\mathbf{C}|$ = 98,914, $\mathbf{R}(\mathbf{T}) = \sum_{i=1}^{|T|} R_i = 5.435$ million dollars, ,$J_{min} = 2450161$ (March 18, 1996), $J_{max} = 2451438$ (September 16, 1999). Thus, the transactional history is over a period of J_{range} = J_{max} - J_{min} + 1 = 1277 days i.e. approximately 3.5 years. The Horizon data, like any typical large transactional data-set, has a large number of products and limited history, and is thus very sparse. The average number of distinct products purchased by a customer is only about 4 out of a choice of 10,842 products. Also, a customer makes an average of only 1.32 visits over the 1277 day period. The majority of customers (84 percent) visit the store only once. This is largely because a majority of customers made their first purchase late into the time period of data acquisition, due to rapid customer growth (Figure 1, bottom plot), and is quite typical of many young e-commerce businesses. Clearly, these customers cannot be used for motion/trend estimation in the Cluster Space. We found empirically that customers with two visits also do not provide reliable information for inferences on trends. Therefore, a subset of the data $\mathbf{T_v}$ containing transactions of only those customers with 3 or more visits was selected for further study.

3 Seasonality Detection and Partitioning

Most transactional data will show some seasonal trends. For a data-set having 10,000 or more products, it is not easy to deal with each product separately for seasonal or periodic trend analysis, but it is possible to detect if there are any relatively global seasonal trends for groups of products. One example of such sales would be gardening or holiday products in cold regions. The following sections describe a technique for detecting "seasons" such that all transactions within a season can be considered as being generated by one model, and significantly different

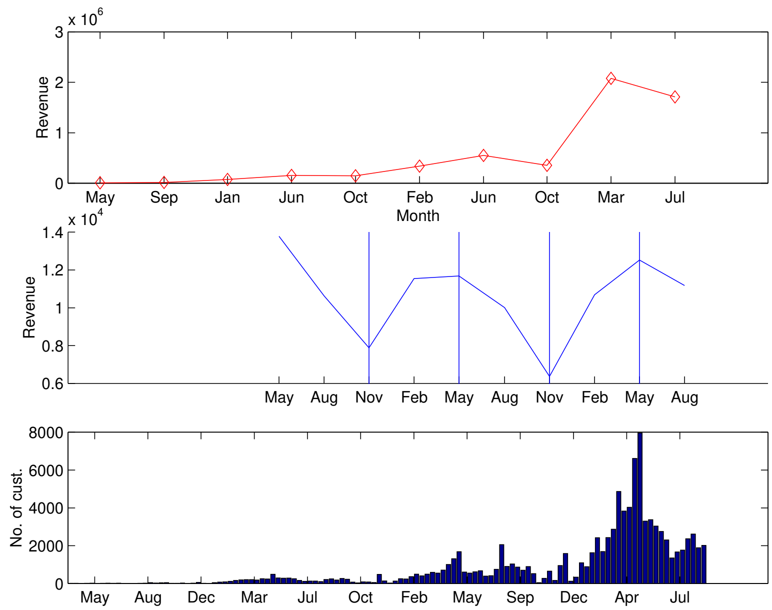


Figure 1. *Aligned Plots for original revenue trend (top), detrended revenue with detected seasons (middle) and joining date of customers (bottom) showing clear correlation among the three.*

models are needed for different seasons.

3.1 Detrending

Detrending is the process of removing long-term trends from a signal or a series [8, 9]. As shown in the top plot in Figure 1, the gross revenue, averaged over 128 day intervals, shows a rapid growth but with some local variations. This is expected for any retail company with a high growth rate. Also the bottom plot in Figure 1 shows that the number of new customers increases over time more than linearly. The last month shows a fall in the number of new customers because of an artifact: the history for the last month is not complete, because it was a future month representing customers who made advance purchases. In such a rapid growth scenario, it makes more sense to look at only the older customers' purchases for revenue modeling. This leads to an automatic detrending that would otherwise be difficult to estimate. Based on this idea a novel detrending technique is proposed as follows -

1. Sort all customers in set $\mathbf{C}$ by their start date $\mathbf{J_S}$.

2. Take the top p percentile of customers and form a subset $\mathbf{C_{top}}$ such that, $\mathbf{C_{top}} = \{C_j \in \mathbf{C} | J_s(C_j) <$ (largest value of J_ϖ such that $|\mathbf{C_{top}}|/|\mathbf{C}| \leq p$) $\}$.

3. Take the subset $\mathbf{T_{top}}$ from original data $\mathbf{T}$ such that

$$\mathbf{T_{top}} = \{T_i \in \mathbf{T} | C_i \in \mathbf{C_{top}} : J_i \geq J_\varpi \}.$$

This subset of data contains only transactions after time J_ϖ, of customers who have made their first purchase before time J_ϖ. Since no new customers are added during this period in this subset, the exponential revenue growth component in the data gets removed. For a reasonable choice of p, a substantial number of customers with substantial history are still left for further analysis.

3.2 Season detection

After obtaining the detrended subset of data as described in Section 3.1, a Temporal Windowing [2] is performed on this subset $\mathbf{T_{top}}$ to obtain a detrended temporal revenue vector R_ϖ. The number of parts into which the data is partitioned for windowing determines the resolution of the temporal vector.

Any seasonal variation in revenue per customer will be clearly visible in the detrended temporal revenue vector (DTRV), R_ϖ. The middle plot in Figure 1 shows such a seasonal trend in the detrended Horizon data that was not so obvious in the original data in the top plot in Figure 1. The correlation between the three plots in Figure 1 are clearly visible, although the seasonal boundaries are clear only in the detrended data.

The periodicity in DTRV can be found by searching for a local minima and maxima after suitable smoothing has been performed to avoid detecting false periods. One way of verifying whether the periodicity detected is reliable or not is by estimating the variation in the width of half-period between a peak and a trough, as indicated by the standard deviation. A good detection of the seasons will minimize standard deviation σ between local minima and maxima for a season, and depends on appropriate selection of percentile p of oldest customers chosen for detrending, number of partitions N_ϖ and the low-pass filtering threshold f_ϖ. Therefore, in order to find the best solution, an exhaustive search was performed using values of f_ϖ and N_ϖ in the valid range. First, a reasonable value of p, the percentile of customers, is chosen so that sufficient time interval is left for temporal vector computation. Then, the frequency f_ϖ is varied between the range $0 < f_\varpi \leq 1$. The range in which N_ϖ is varied is computed as follows:

To detect a frequency in a signal, the sampling rate should be at least twice the frequency (Nyquist frequency). Thus for discovering a period P, $U_\varpi \leq \frac{P}{2}$, where $U_\varpi = \lfloor \frac{J^\varpi_{range}}{N_\varpi} \rfloor$. Thus the lower bound on N_ϖ is given by $N_\varpi \geq 2J_{\varpi range}/P$. Under-sampling can cause detection of false periods as shown in right bottom plot in Figure 2. The upper bound on N_ϖ is determined by the time granularity of the date itself. For the Horizon data this is one day, so N_ϖ cannot be more than J^ϖ_{range}. Thus the range in which N_ϖ can vary is given by:

$$\frac{2J_{\varpi range}}{P} \leq N_\varpi \leq J^\varpi_{range}. \tag{1}$$

[2]We define *Temporal Windowing* as partitioning of any transactional data T into N parts based on the time of the transaction, such that each part covers an equal interval of time, and then aggregating the revenue in each interval. The result is a revenue vector of size N.

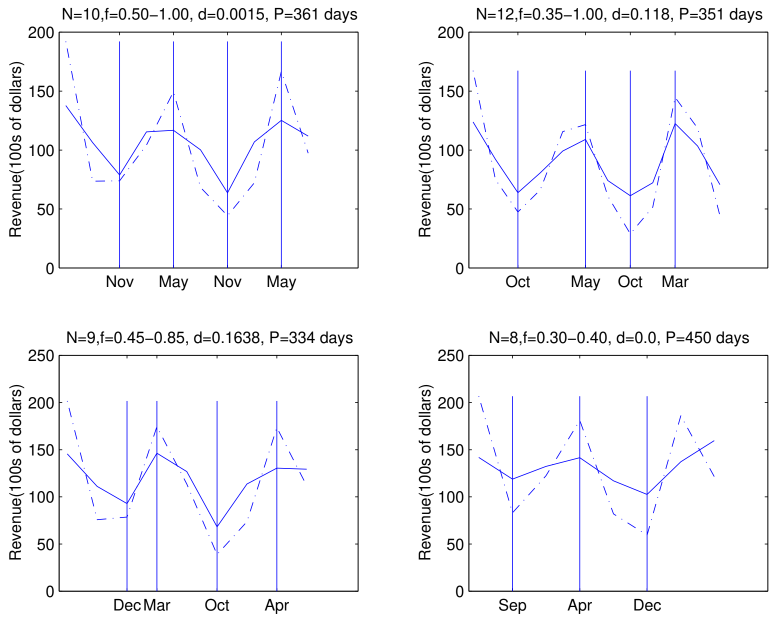


Figure 2. *Season detection for various values of f_ϖ and N_ϖ. The top-left plot corresponds to the best solution found.*

Each possible solution (f_ϖ, N_ϖ) is evaluated in terms of the variance in the intervals and how many times it gives the same periodicity P. While the standard deviation $\sigma(P)$ of the the period across peaks and valleys should be low, the trend should appear on various choices of (f_ϖ, N_ϖ) combinations. Defining reliability $R(P)$ as the number of times the period was found among a small range of (f_ϖ, N_ϖ), the overall cost of a period P is defined as:

$$C(P) = \sigma(P) + \frac{1}{R(P)}. \tag{2}$$

The solution with the minimum cost is found by an exhaustive search with various values of f_ϖ and N_ϖ over their respective valid ranges.

Figure 3 shows the solution (the one marked 'Best') found with minimum cost $C(P)$ among 14x14=196 solutions, corresponding to $N_\varpi = 10$. This point actually represents a set of 14 identical solutions among the 196 solutions. The period P of this solution is 361 days, and is very close to a one year period.

3.3 Seasonal Migration and Drift

A strong correlation in terms of the products purchased among peak season sales and off season sales indicates a seasonal trend in product purchase distribution. To identify such correlations, we first partition the transactional data by using the midpoints of two consecutive seasons as boundaries. Thus if May is the center of

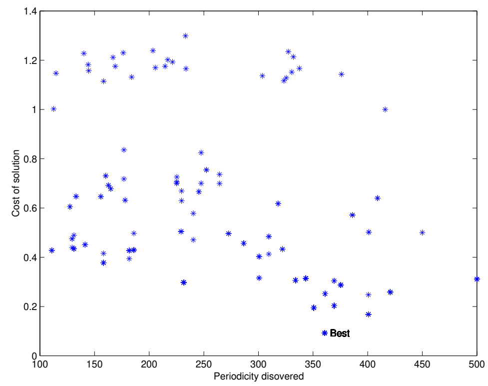


Figure 3. *A plot of the solutions discovered using various values of f_{ϖ} and N_{ϖ}. The X-axis represents the cost C_i of the solution while the Y-axis is the Period P detected. The best solution is the one with minimum Cost C_i near the Period of 361 days.*

	PS 1	*OS 1*	*PS 2*	*OS 2*	*PS 3*
PS 1	1	0.1306	**0.2022**	0.0889	**0.1442**
OS 1	0.1306	1	0.1688	**0.2186**	0.1251
PS 2	**0.2022**	0.1688	1	0.1790	**0.2713**
OS 2	0.0889	**0.2186**	0.1790	1	0.1928
PS 3	**0.1442**	0.1251	**0.2713**	0.1928	1

Table 1. *Product Overlap Matrix showing overlap between seasonal sales. PS stands for Peak Season and OS for Off Season. As can be seen, the three peak-season sales and the two off-season sales are more similar to one another than peak-season vs. off-season sales except for peak-season 3 & off-season 2.*

the off-season and November is the center of the peak-season, then say for year 1999-2000, all sales from February 16 1999 to August 15 2000 can be treated as peak-season sales and all sales from August 16 1999 to February 15 2000 can be treated as off-season sales. Transactions are first partitioned into subsets $\mathbf{T^1_{\varpi}}..\mathbf{T^{N_O}_{\varpi}}$ and $\mathbf{T^1_{\varpi}}..\mathbf{T^{N_P}_{\varpi}}$, corresponding to off-seasons and peak-seasons respectively, where N_O represents the number of off-seasons detected and N_P represents the number of peak-seasons detected. Then a product vector $\mathbf{H}$ is generated for each of the peak and off-season transaction sets where the k^{th} entry of the vector $\mathbf{H_i}$ contains the total revenue for product k in the season i. This vector is normalized by dividing all the entries by the largest vector entry i.e. for all entries k -

$$\mathbf{H_i}(k) = \mathbf{H_i}(k)/max(\mathbf{H_i}) \quad (3)$$

Finally, we compute the Product Overlap Matrix $\mathbf{M_O}$, such that $\mathbf{M_O}(i,j) = Sim_{MMSIM}(\mathbf{H_i}, \mathbf{H_j})$, where Sim is a suitable similarity measure. This paper uses a value based similarity measure called $MMSIM$, first introduced in VBACC [7], since we are using revenue for Trend detection. Let $\mathbf{P}_k$ represent the product vector for a customer k such that P_{ki}, its i^{th} entry, represents the value of the product i purchased by customer k over all transactions. Let $\mathbf{T_{ki}}$ represents the subset of all transactions in which customer k bought product i and W_{ki} represent the total value or money spent by customer k on purchasing product i. Then the MMSIM similarity between two customers k and ℓ is given by -

$$sim_{MMSIM}(k,\ell) = \frac{\sum_{i=1}^{|\mathbf{P}|} \min(W_{ki}, W_{\ell i})}{\sum_{i=1}^{|\mathbf{P}|} max(W_{ki}, W_{ji})} \tag{4}$$

More insight into this similarity measure is given in [7, 10]. The seasonal migration is clear from Table 1. Off-Season 1 and Off-Season 2 are very similar to each other while Peak-Season 1 and Peak-Season 2 are close. Similarly, Peak Season 2 sales match Peak Season 3 sales. But as the seasons pass not just the volume of the customers goes up, but the market itself starts drifting. This is visible in Table 1. As we can see product distribution in Peak Season 1 and Peak Season 2 are much more similar to each other than Peak Season 1 and Peak Season 3. Similarly Peak Season 2 & 3 are more similar to each other than Peak Season 1 and Peak Season 3. Thus two consecutive seasons are more similar. In fact Off Season 2 is more similar to Peak Season 3 than Peak Season 1. This can be termed as market drift with time. Thus for Horizon data, in two years (the period between Peak Season 1 and 3) the market model changed a lot. Thus, any model created using past data would be valid for approximately two years for Horizon data. Thus the model will certainly need to be updated after two years.

4 Cluster Space

The trends described in Sections 3 are macroscopic. This section describes a *Cluster Space model*, a framework for visualizing and characterizing *individual* customer dynamics. It involves performing clustering on all of the peak-season data as one set and all the off-season data as another set into two groups of clusters, and then mapping the two sets of clusters obtained into a common *Cluster Space*. Customer motion is then characterized in this continous space. The following subsections describe this technique in more detail.

4.1 Model

After the removal of outliers and clustering on the peak-season and off-season data $\mathbf{T_{vp}}$ and $\mathbf{T_{vo}}$, two corresponding sets of clusters $\mathbf{U_{vp}}$ and $\mathbf{U_{vo}}$ are obtained, representing mutually disjoint and exhaustive set of clusters. The clustering routine used is called VBACC, and is described in detail in [7]. It transforms the transactional data into a graph with edges representing a value based similarity between the two corresponding transactions (vertices). This graph is then partitioned using an efficient algorithm, METIS, to obtain clusters of vertices (transactions).

The set of clusters $\mathbf{U_v} = \mathbf{U_{vp}} \cup \mathbf{U_{vo}}$ is used to define the Cluster Space. The clusters in the combined set $\mathbf{U_v}$ are labeled from 1 to $(|\mathbf{U_{vp}}| + |\mathbf{U_{vo}}|)$ with the clusters from set $\mathbf{U_{vo}}$ labeled from 1 to $|\mathbf{U_{vp}}|$ and clusters from set $\mathbf{U_{vp}}$ labeled from $(|\mathbf{U_{vp}}|+1)$ to $(|\mathbf{U_{vp}}|+|\mathbf{U_{vo}}|)$.

Let $\mathbf{P_v} \subset \mathbf{P}$ represent the set of all the unique products purchased in transactional data $\mathbf{T_v}$. Then, for every cluster $U_m \in \mathbf{U_v}$, a product vector α_m is computed as follows -

1. Find all the transactions T_m for customers $C_m \in U_m$.
2. Generate a product vector $\alpha_{\mathbf{m}}$ for the cluster m where the k^{th} entry of the vector contains the total revenue for product k in T_m.

Let $\beta_{m,k}$ be the fraction of the amount spent by an average customer in cluster U_m on product k -

$$\beta_{m,k} = \frac{\alpha_{m,k}}{\sum_{j=1}^{|P_v|} \alpha_{m,j}} \tag{5}$$

In the product space, both clusters and customers are represented by a vector of products. The similarity between a customer and a cluster is defined in terms of their product distribution similarity. The MMSIM similarity measure defined in Section 3.3 is used to compute these similarities. Let τ_i represent the product vector for a customer C_i, and the product distribution of an average customer in cluster $U_m \in \mathbf{U_v}$ be $\alpha_m/|U_m|$. Then, the similarity between the customer C_i and cluster U_m is given by -

$$sim_{MMSIM}(C_i, U_m) = \frac{\sum_{j=1}^{|\mathbf{P_v}|} \min(\tau_{i,j}, \frac{\alpha_{m,j}}{|U_m|})}{\sum_{j=1}^{|\mathbf{P_v}|} \max(\tau_{i,j}, \frac{\alpha_{m,j}}{|U_m|})} \tag{6}$$

Let $\mathbf{C_v}$ represent the set of all customers in the transactional data $\mathbf{T_v}$. The Cluster Similarity Vector (CSV) ϑ_i can now be computed for each customer $C_i \in \mathbf{C_v}$. The size of this vector is the same as the number of clusters i.e. $|\mathbf{U_v}|$. The m^{th} entry of ϑ_i represents the similarity of customer C_i with cluster U_m and is computed as follows -

$$\vartheta_{i,m} = sim_{MMSIM}(C_i, U_m) \tag{7}$$

Since the MMSIM similarity is always between 0 and 1, Equation 7 represents the mapping of each customer from a very high dimensional (more than 10,000 dimensions for the Horizon data) categorical product space onto a low dimensional ($|\mathbf{U_v}|$ dimensions) continuous space bounded within a unit hypercube. The position of a customer C_i in this space is defined by the value of the CSV vector ϑ_i. This space is referred to as the Cluster Space. Notice that two customers having similar purchase profiles i.e. they are close in the Product Space also lie close together in the Cluster Space. The Cluster Space carries more information than a discrete and static clustering model since each customer is assigned not just a cluster label but a CSV vector defining its similarity with all clusters.

4.2 Motion Estimation

Since the position of a customer C_i depends on the product vector τ_i for that customer, it allows for dynamically projecting the customer onto this Cluster Space by recomputing τ_i as new transactions for a customer occur. This is the essence of the dynamic modeling using motion estimation in the Cluster Space.

In Table 1 and Section 3.3, we quantified seasonal market migration and also discovered drift in the market beyond a two years period for the Horizon data. We found that although the market remained relatively unchanged over a period of two years across successive similar seasons, there was a substantial migration between a peak-season and an off-season. By creating a Cluster Space after partitioning the off-season and peak-season data, the dynamics of seasonal variation is captured. The Cluster Space thus models the market drift along with seasonal migration. A customer's position can now be plotted onto this Cluster Space by creating a CSV(*Cluster Space Vector*) vector ϑ across time. For customers with sufficient number of visits, the plot of the position of the customer should show a relatively smooth motion across time as his preferences change with time and maybe even oscillating across seasons. In this section, a set of techniques that can be used for motion detection and estimation in Cluster Space are described.

Position Matrix Computation

The position of a customer in the Cluster Space is determined by his transactional history. As the customer purchases new things, this position shifts with time. The position of a customer C_i in the Cluster Space from the beginning to the end of the transactional history can thus be described by a two dimensional position matrix ϕ_i. The first row of ϕ_i represents the value of the CSV vector ϑ_i, or the position of customer in Cluster Space at the beginning of the transactional history, while the last row represents the value of ϑ_i at the end of the transactional history time. The matrix ϕ_i is computed by windowing across time over the customer's transactions.

Let W represent the width of a time window such that $1 \leq W \leq J_{range}$. Further, let S be a step size such that $1 \leq S \leq W$. Then the t^{th} window $I_t = [\ell_t, r_t]$, where ℓ_t represents the left boundary of the window and r_t represents the right boundary is defined as -

$$\begin{aligned} \ell_t = &\quad J_{min} + (t-1)S \\ r_t = &\quad \ell_t + W \end{aligned} \tag{8}$$

Let $\mathbf{T_i}$ represent the set of all transactions by customer $C_i \in \mathbf{C_v}$. Furthermore, let $\mathbf{T_{i,t}}$ represent all transactions by customer C_i in the window I_t i.e. -

$$\mathbf{T_{i,t}} = \{T_k \in \mathbf{T_i} : \ell_t \leq J_k < r_t\} \tag{9}$$

Let $\tau_{\mathbf{i,t}}$ represent the product vector for customer C_i in time window I_t computed by aggregating the transaction set $\mathbf{T_{i,t}}$ where $\tau_{i,t}(k)$ represents the total revenue for product k in $\mathbf{T_{i,t}}$. Then, from Equation 6 the entry (t, m) of the position matrix ϕ_i in the time window I_t with respect to the cluster U_m is given by -

$$\phi_i(t,m) = \frac{\sum_{j=1}^{|\mathbf{P_v}|} \min(\tau_{i,t}(j), \frac{\alpha_{m,j}}{|U_m|})}{\sum_{j=1}^{|\mathbf{P_v}|} \max(\tau_{i,t}(j), \frac{\alpha_{m,j}}{|U_m|})} \tag{10}$$

The number of time windows N_w depends on both the window size W and the step size S -

$$N_w = 1 + \left\lfloor \frac{J_{range} - W - 1}{S} \right\rfloor \tag{11}$$

Thus the position matrix ϕ is of size $|\mathbf{U_v}|$ (dimensionality of the Cluster Space) columns and N_w rows, each representing position of customer C_i at a different time.

Gradient Estimation and Motion Visualization

For a given customer C_i, $\tau_{i,t}$ and $\tau_{i,t+1}$ are the product vectors in two consecutive time intervals I_t and I_{t+1} respectively, thus representing the incremental change in the purchase behavior of the customer in time. Therefore, the position of the customer in the Cluster Space during the t^{th} time interval, i.e. $\phi_i(t)$ should also be close to $\phi_i(t+1)$ in the $|\tau_{\mathbf{v}}|$ dimensional Cluster Space. One simple way of visualizing the motion is by plotting the matrix ϕ_i as a three-dimensional histogram as shown in Figure 4. In this visualization scheme, similarity (Z-axis) is plotted against N_w time intervals (X-axis) and $|\mathbf{U_v}|$ dimensions of the clusters space (Y-axis).

The rate of change of the similarity of a customer C_i with cluster U_m gives a measure of the instantaneous velocity of the customer in the m^{th} dimension in the Cluster Space. This can be computed from gradient of the m^{th} column vector of matrix ϕ_i. Since there are random perturbations in the velocity components due to the sparsity of the data, a smooth function is fitted to it before estimating customer velocity.

Let ϕ_i^{smooth} denote the smoothed position matrix for customer C_i. The gradient at any time t along the m^{th} dimension of the Cluster Space is computed from ϕ_i^{smooth} for all time $t > 1$:

$$\phi_i'(t, m) = \phi_i^{smooth}(t, m) - \phi_i^{smooth}(t-1, m) \tag{12}$$

where $1 < t \leq NSTEP$. Thus, $\phi_{\mathbf{i}}'$ is a $N_w \times |\mathbf{U_v}|$.

The visualization of the smoothed position vector using a 3-D histogram is very useful in identifying some of the important trends in a customer's purchase behavior, especially for the most frequently visiting customers since they account for a significant part of the revenue and also have sufficient number of visits for good motion modeling. Prominent and long-lived trends include customer migration of various kinds such as periodicity w.r.t. each cluster, moving from one cluster to another, oscillating between peak and off-season clusters, opposite movement with respect to certain cluster pairs, moving in the same manner with certain clusters signifying that the clusters are closely related for the specific customer, continuously drifting away from a given cluster; although still being closest to it, starting to move the eventually closest cluster quite early, relatively flat similarity from a cluster signifying that the customer always buys some products from the cluster regularly etc. Some of the visualization results for the Horizon data are discussed in Section 5.

Time(day)	*Closest Cluster*	*Type of Cluster*
0 - 426	28	Peak-Season
427 - 684	7	Off-Season
685 - 795	28	Peak-Season
796 - 869	15	Off-Season
870 - 999	7	Off-Season
1000 - 1095	28	Peak-Season

Table 2. *A table showing a summary of the oscillation of customer 159582 between Peak-Season & Off-Season Clusters (also visible in Figure 5).*

5 Experimental Results for Cluster Space

5.1 Cluster Space formation for Horizon Data

After clustering the peak-season and off-season data separately and removing the outliers in the Horizon data, 24 clusters are obtained for each season. The two sets of clusters are then combined to give a Cluster Space of 48 reasonably balanced clusters (the revenue of the smallest cluster was only 5 times smaller than that of the largest cluster). Clusters 1 through 24 represent the off-season clusters, while clusters 25 through 48 are clusters from the peak-season data. There were 46 and 14 outliers in the off-season and peak-season clusters respectively.

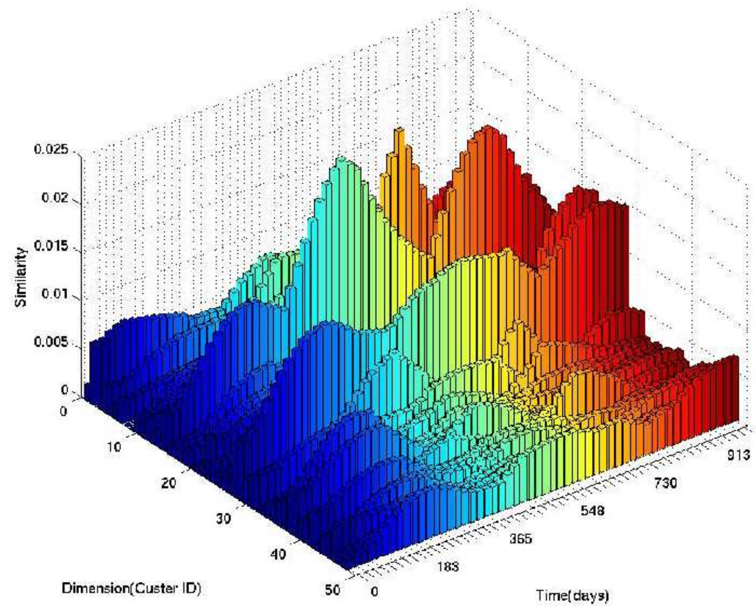
5.2 Some Interesting Detected Motions

Figure 4(a) and 4(b) show the 3-d plots of the position matrix ϕ_i for two frequent customers: Customer 159582 and 306671, having 57 and 53 visits respectively. Some of the interesting trends discovered for these customers using the 3-D visualization are discussed in this section.

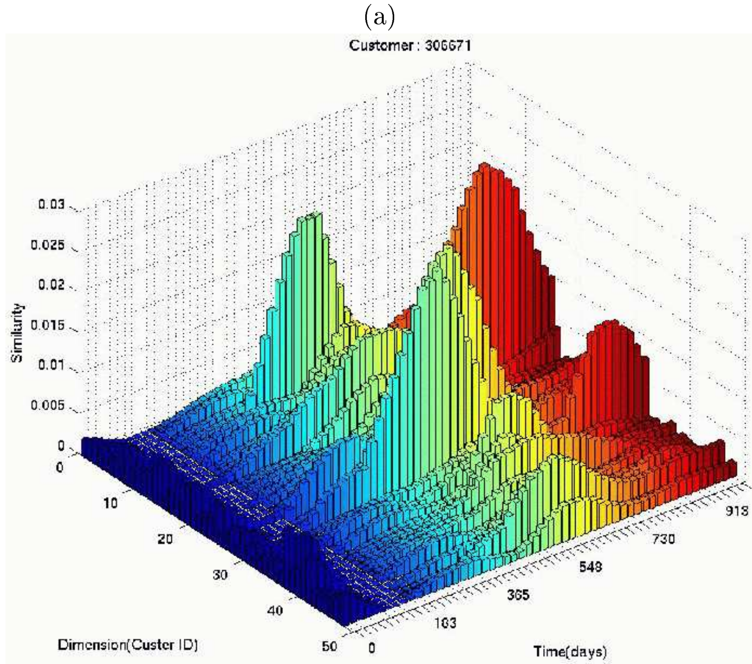
Seasonal Migration

Figure 5 shows a plot of the similarity/position of customer 159582 w.r.t dimensions/clusters 7,15 and 28. These are three clusters to which the customer is closest to at various times and they are also visible (albeit less clearly) in Figure 4(a). It is a classical example of seasonal migration in customers that could not have been detected by static methods, or even without seasonal partitioning of the data. Thus the Cluster Space model, with its combination of two sub-spaces, is able to discover such seasonal migrations.

A summary of this motion is shown in Table 2. The customer is closest to Cluster 28, 7, 28, 15, 7 and 28 in that order across time. If we look at the sub-space to which each of these cluster belong, then the we get the sequence : Peak, Off, Peak, Off, Off, Peak clearly showing a seasonal oscillation (since clusters 7 & 15 belong to off-season and 28 to peak-season).



(a)



(b)

Figure 4. *3-D plot of the Position Matrix in the Cluster Space for customer (a) 159582 (b) 306671, showing motion in the 48 dimensions across time. The X-axis is the time in days, Y-axis represents the 48 dimensions and Z-axis is the corresponding similarity with each cluster dimension for a the given time range from 0 to 1095 days.*

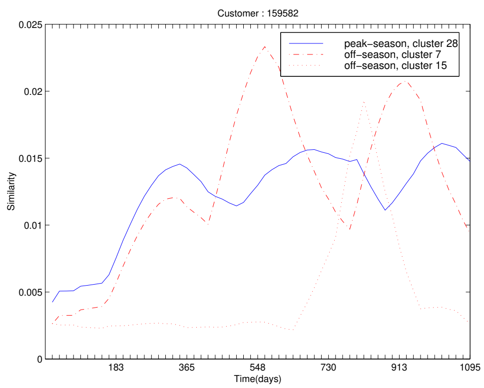


Figure 5. *A plot showing the similarity/position of customer 159582 w.r.t dimensions/clusters 7,15 and 28 showing the customer oscillating between peak-season and off-clusters across time.*

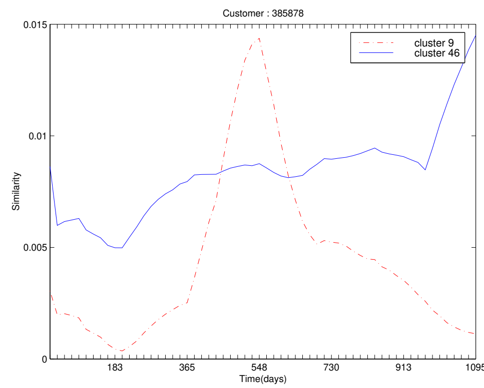


Figure 6. *A comparison of the similarity variation/motion of Customer 385878 across time with clusters 9 and 46.*

Consistent Purchases

Figure 6 shows the similarity of customer 385878 with respect to clusters 9 and 46. Compared to the variation in similarity with cluster 9, the variation with cluster 46 is much smaller for majority of the time. The customer is significantly close to cluster 46 throughout the entire time. It probably means that the customer keeps buying at least some products characteristic of that cluster independent of the season. It is also possible that such products fall into a 'regular use' category for this customer.

Correlated Cluster Motion

Figure 7 show a clear correlation in motion w.r.t. clusters 1 & 3, both of which belong to the off-season sub-space for customer 159582. This means that the two clusters are correlated by products that the customer purchases. Such correlation between these two clusters was also seen across other customers thus signifying that the two groups are correlated. One reason for such seasonal correlation might be that both belong to the off-season sub-space. Figure 8 also shows such a correlation between clusters 27 & 29 both belonging to the peak-season sub-space and a negative correlation with cluster 15 which is an off-season cluster.

Customer Drift

A static model assumes that every customer belongs to one cluster across the time period over which the clustering is performed. Another motivation behind the Cluster Space paradigm was to dynamically assign a cluster label to a customer across the same range and also the immediate future. This is meaningful only if

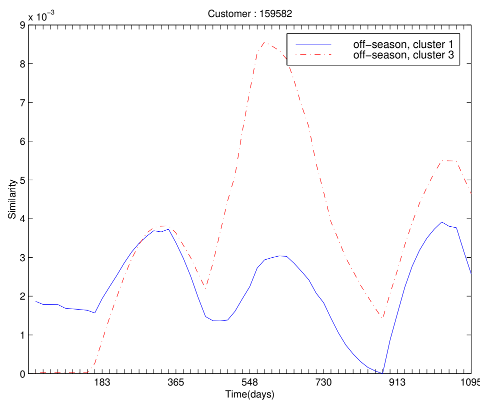


Figure 7. *A plot of similarity between two clusters; 1 & 3 showing a clear seasonal correlation across time for customer 159582.*

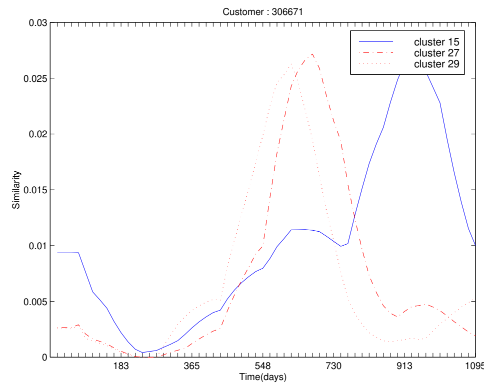


Figure 8. *A plot of similarity between three clusters; 15, 27 & 29 for customer 306671 showing customer drift across time. A clear correlation can also be seen between clusters 27 & 29.*

the customers exhibits a long-term trend across time. For customers with sufficient history, this is possible. Figure 8 shows a classical example of a slow 'drift' or migration of customer 306671 from cluster 27 to 29 to 15. Also, it can be seen that the similarity of the customer with respect to Cluster 15 steadily increases except for a small blip.

6 Concluding Remarks

For many transactional datasets, a small number of customers account for most of the visits, revenue and transactions while the vast majority visit very rarely and purchase only a few items out of a large number of products. Looking at only customers with more than three or four visits removes this skewness to a great extent. After this pre-filtering, the process described for seasonality detection in the data was able to find very strong seasonal trends even in limited length of data with strong growth ramp. It could automatically extract both the periodicity and the season boundaries. The Product Overlap Matrix enabled effective partitioning of the data, and both kinds of market migration: Market drift and oscillation, were discovered. The Product Overlap Matrix also makes it possible to automatically detect the duration for which a clustering model remains valid. For the Horizon data, this duration was found to be approximately two years.

Combined with VBACC, the Cluster Space model provides a meaningful conversion of a very high-dimensional categorical product space into a low-dimensional numerical space. At the same time, the Cluster Space overcomes issues in mapping to a metric space with traditional methods such as SVD and PCA, by operating in similarity space. The continuous nature of the position of customers in a Cluster

Space modeling also allows for modeling of the migration of customers. The windowing scheme suggested works well for customers with sufficient number of visits. Combining the off-season and peak-season sub-spaces while forming the Cluster Space also allows for discovering some interesting customer motions such as cyclicity in a customer's purchase preferences. We believe that the cluster space is an effective framework for discovering dynamic trends in customer buying patterns in particular and any high dimensional, categorical sparse data containing sufficient temporal variation in general. More details about this work is available in [10].

7 Acknowledgements

We would like to thank Knowledge Discovery One (*http://www.kd1.com*; now part of Net Perceptions) for providing us with the Horizon dataset, and especially Mark Davis for his valuable suggestions.

Bibliography

[1] M. A. Hernandez and S. J. Stolfo, "Real-world data is dirty: Data cleansing and the merge/purge problem," *Data Mining and Knowledge Discovery* **2**, pp. 9–37, January 1998.

[2] J. Deogun, S. Choubey, V. Raghavan, and H. Sever, "Data mining: Trends and issues," *Journal of ASIS* **49**(5), pp. 397–402, 1998.

[3] A. R. Agrawal, T. Imielinski, "Database mining: A performance perspective," *IEEE Transactions on Knowledge and Data Engineering* **5**, pp. 914–925, December 1993.

[4] M.-S. Chen, J. Han, and P. S. Yu, "Data mining: An overview from a database perspective," *IEEE Transactions on Knowledge and Data Engineering* **8**, pp. 866–883, December 1996.

[5] W. Klosgen, "Deviation and association patterns for subgroup mining in temporal, spatial, and textual data bases," *Proc. First International Conference on Rough Sets and Current Trends in Computing, RSCTC'98* , pp. 1–18, 1995.

[6] X. Chen and I. Petrounias, "Mining temporal features in association rules," *Proc. 3rd European Conference on Principles and Practice of Knowlediscovery in Databases (PKDD'99), Prague, Lecture Notes in Artificial Intelligence* **1704**, pp. 295–300, 1999.

[7] G. K. Gupta and J. Ghosh, "Value balanced agglomerative connectivity clustering," in *SPIE conference on Data Mining and Knowledge Discovery III*, April 2001. Accepted for publication.

[8] T. Masters, "Centering and detrending," in *Neural, Novel & Hybrid Algorithms for Time Series Prediction*, pp. 9–15, April 1995.

[9] A. Harvey and A. Jaeger, "Detrending, stylized facts and the business cycle," *Journal of Applied Econometrics* **8**(3), pp. 231–47., 1993.

[10] G. K. Gupta, "Modeling customer dynamics using motion estimation in a value based cluster space for large retail data-sets," Master's thesis, University of Texas at Austin, August 2000.

[11] G. K. Gupta, A. Strehl, and J. Ghosh, "Distance based clustering of association rules," in *Intelligent Engineering Systems Through Artificial Neural Networks (Proceedings of ANNIE 1999)*, vol. 9, pp. 759–764, ASME Press, November 1999.

[12] D. Pyle, *Data Preparation For Data Mining*, John Wiley & Sons, 1997, 1999.

[13] G. Karypis, R. Aggarwal, V. Kumar, and S. Shekhar, "Multilevel hypergraph partitioning: Applications in VLSI domain," in *Proceedings of the Design and Automation Conference*, 1997.

[14] S. Guha, R. Rastogi, and K. Shim, "Rock: a robust clustering algorithm for categorical attributes," in *Proceedings of the 15th International Conference on Data Engineering*, 1999.

[15] G. Karypis, E.-H. Han, and V. Kumar, "Chameleon: Hierarchical clustering using dynamic modeling," *IEEE Computer* **32**, pp. 68–75, August 1999.

[16] J. A. Hartigan, *Clustering Algorithms*, Wiley, New York, 1975.

[17] B. Everitt, *Cluster Analysis – 3rd ed.*, Halsted Press, New York, 1993.

[18] G. Karypis and V. Kumar, "Multilevel k-way partitioning scheme for irregular graphs," *Journal of Parallel and Distributed Computing* **48**(1), pp. 96–129, 1998.

[19] B. Hendrickson and R. Leland, "An improved spectral graph partitioning algorithm for mapping parallel computations," *SIAM Journal on Scientific Computing* **16**(2), pp. 452–469, 1995.

[20] A. Pothen, H. Simon, and K. Liou, "Partitioning sparse matrices with eigenvectors of graphs," *SIAM Journal of Matrix Analysis and Applications* **11**, pp. 430–452, 1990.

[21] G. Salton, A. Wong, and C. Yang, "A vector space model for automatic indexing," *Communications of the ACM* **18**(11), pp. 613–620, 1975.

[22] M. Berry and G. Linoff, *Data Mining Techniques for Marketing, Sales and Customer Support*, John Wiley & Sons, 1997.