# Market Basket Recommendations for the HP SMB Store

Pramod Singh
*Hewlett-Packard Company*
*14231 Tandem Blvd*
*Austin, TX  - 78727*
*(512) 432 8794*
pramod.singh@hp.com

A Charles Thomas
*Hewlett-Packard Company*
*20555 Tomball Parkway*
*Houston, TX - 77070*
*(281) 514 1804*
cthomas@hp.com

Ariel Sepulveda
*Hewlett-Packard Company*
*PO Box 4048*
*Aguadilla, PR - 00605*
*(787) 819.6057*
ariel.sepulveda@hp.com

## Abstract

*The Analytics team at Hewlett-Packard recently executed a manually-driven cross-sell/up-sell pilot in the Small and Medium Business online store and call center. The pilot, for which management dictated a 1 month development timeframe, utilized sales transaction, product configuration, and product availability data. Leveraging Market Basket analysis techniques among a small subset of available product SKUs, the pilot yielded a ROI of more than $300K/month and more importantly, gave birth to greater opportunities to further showcase the power of analytics and data driven decision-making at HP.*

## 1.  Introduction

For the purpose of enhancing direct sales transaction revenue and profit, the Analytics team at HP was asked to execute a cross-sell/up-sell project for the Small and Medium Business (SMB) store's online site and call center.  Cross-selling includes, among others, adding a monitor, docking station, or a digital camera to a notebook purchase.  An up-sell is loosely defined as "inside the box." For example, up-selling would include adding anything that enhances value of a PC, such as upgraded memory, hard drive, or a DVD drive.

The pilot's overall goal was to increase the revenue and margin of the store by increasing average order value (AOV) and attach rate per product by implementing an analytic solution.  This pilot served as a proof of concept, which may translate into future investment in a more automated alternative.   To create the manual execution, a process was built where existing sales, product information, and internal marketing information were integrated and analyzed to identify potential cross-sell and up-sell recommendations.   Note that the SMB store is public.  In other words, the team could

not link visitors to their historical purchase behavior. Thus, they leveraged techniques more geared towards aggregate-level analyses.

The pilot was originally designed to generate learning that would lead to a staged regional implementation and eventual world-wide program which enables segment marketing and telesales teams to increase attach, margin and revenue in the direct business by providing the foundation, tools and capabilities to enable the generation and delivery of analytically driven offers.

Recommendations for more than 25 models of desktops, notebooks, printers, servers, storage and workstations product categories, representing more than 100 SKUs, were generated during the pilot. Maximization of revenue coverage and product availability were key criteria in the SKU selection process.   The recommended offers were rendered at the product configurator (web page where customers can customize their machine) and the check-out pages of the selected products.

The 5 person team (3 from Analytics and 2 from IT) assigned to the project planned two incremental releases.  Release 1 leveraged historical purchase data to generate recommendations based upon what has traditionally been purchased together in the past. The objective of this phase was to maximize expected cross-sell and up-sell revenue.  Release 2 added product profitability data to historical purchase data to yield recommendations based upon traditional product affinities and those products most profitable to the company.  In this phase, the team planned to maximize a weighted function of revenue and profitability, weights being provided by the business owners.  Release I of the pilot ran from Nov 04 – March 05 and Release II of the pilot ran from April 05 to July 05.

Overall, the project required data preparation, statistical analysis, business review and final recommendation generation, recommendation

execution, periodic reporting, and post-pilot performance evaluation.

## 2. Data Preparation

For meeting the objectives of using margin and revenue in the scoring function to rank available recommendations, and also to provide lead time as a potential criterion, different HP data sources, all new to the analytics and supporting IT team were identified, evaluated and integrated. Sources included sales transactional data, product configuration, product hierarchy, and product availability data among others. Data challenges included a high percentage of missing values in some key fields and difficulty associating the appropriate price for a SKU given frequent price changes. In cases where these values were missing, the analysts developed a hierarchical imputation process in which missing margins for SKUs would be estimated based on similar SKUs in the product hierarchy.

Once all the data were available, a flat table was created with dimensions of about 3,000,000×20 for more than a year of data. Some of the most significant fields in that table were Order ID, Date, SKU, SKU Description (descriptions at different levels of the product hierarchy), Quantity, Price, Cost, and Lead Time (an indicator for product availability). The winning Cross-sell recommendations were selected from thousands of competing SKUs. A sample of the data is shown in Table 1. In this table, the bold faced and shaded rows describe base units or machine sold with upgrades described in the remaining bold face rows.

The next section describes in more detail how the team analyzed the data to yield the recommendations.

## 3. Data Analysis

Before merely jumping in and performing the analysis, the team decided to operationalize the dependent variables: cross-sell and up-sell. Cross-sell refers to "outside-the-box" purchases that customers add to a base unit. One simple example is the purchase of a printer cable, print cartridges, or paper along with a printer. On the other hand, up-sell refers to upgrades to systems that are sold as a box. For example, a desktop bundle including monitor, software, and all the internal features like DVD writer or memory is considered to be a "box." Therefore, upgrading from a 256 MB memory to 512 MB memory in such a desktop package is considered to be an "up-sell." However, in cases where a customer does not select a package, but rather chooses to purchase a desktop alone, selling a 21 inch monitor in many cases may also be considered as a cross-sell because the CPU might not include a monitor as part of the main product or box.

Due to the different options for configuring a desktop, up-sell becomes an important characteristic of customer behavior. For configuration purposes, up-sells are offered right in the configuration page (where customers customize the box to meet their needs), whereas cross-sells can be offered on the configurator as well as in the cart page (see Figure 1 below as implemented on the www.smb.compaq.com website).

Up-sells are also challenging in the sense that compatibility must be taken into account. For example, the DVD drive for a certain notebook might not be a valid up-sell for a desktop. Thus, it was important to create a solution capable of differentiating and evaluating cross-sells and up-sells right from the data as presented in Table 1, and of taking into account all compatibility constraints.

The nature of the SMB store drove the type of analysis the team was able to use. The store is considered public, in other words, we are unfamiliar with the customer as they come in and browse or speak to a representative. Since no login exists, and therefore the system cannot link this customer to their individual historical purchase behavior, we are unable to leverage techniques that yield individual product/offer probabilities (such as via logistic regression). Furthermore, time did not permit enhancements to the site that would call for real time data collection on the customer (i.e. technographics, clickstream behavior, or pop-up survey) for the purpose of individual-level modeling and scoring. Therefore, the team chose to adopt market-basket analyses that take advantage of the data available, despite its individual-level predictive power limitations.

**Table 1:  Sample of fields for order 849608**

| tie_group_id | parent_comp_id | comp_id | osku | description | qty |
|---|---|---|---|---|---|
| 1 | 6706 | 6706 | 301897-B22 | HP StorageWorks MSL5030 Tape Library, 1 LTO Ultri | 1 |
| 2 | 0 | 22814 | C7971A | HP Ultrium 200 GB Data Cartridge | 10 |
| 3 | 0 | 22961 | C7978A | HP Ultrium Universal Cleaning Cartridge | 1 |
| **4** | **7167** | **7167** | **292887-001** | **Intel Xeon Processor 2.40GHz/512KB** | **2** |
| **4** | **7167** | **26919** | **1GBDDR-1BK** | **1GB Base Memory (2x512)** | **2** |
| **4** | **7167** | **20481** | **286714-B22** | **72.8 GB  Pluggable Ultra320 SCSI 10,000 rpm Unive** | **2** |
| **4** | **7167** | **20478** | **286713-B22** | **36.4GB  Pluggable Ultra320 SCSI 10,000 rpm Unive** | **2** |
| **4** | **7167** | **24578** | **326057-B21** | **Windows® Server 2003 Standard Edition + 5 CALs** | **2** |
| 5 | 7167 | 7167 | 292887-001 | Intel® Xeon Processor 2.40GHz/512KB | 4 |
| 5 | 7167 | 20478 | 286713-B22 | 36.4GB  Pluggable Ultra320 SCSI 10,000 rpm Univers | 4 |
| 5 | 7167 | 20478 | 286713-B22 | 36.4GB  Pluggable Ultra320 SCSI 10,000 rpm Univers | 4 |
| 5 | 7167 | 24578 | 326057-B21 | Windows® Server 2003 Standard Edition + 5 CALs | 4 |
| 6 | 7167 | 7167 | 292887-001 | Intel® Xeon Processor 2.40GHz/512KB | 1 |
| 6 | 7167 | 26919 | 1GBDDR-1BK | 1GB Base Memory (2x512) | 1 |
| 6 | 7167 | 20484 | 286716-B22 | 146.8 GB  Pluggable Ultra320 SCSI 10,000 rpm Unive | 1 |
| 6 | 7167 | 20484 | 286716-B22 | 146.8 GB  Pluggable Ultra320 SCSI 10,000 rpm Unive | 1 |
| 6 | 7167 | 24578 | 326057-B21 | Windows® Server 2003 Standard Edition + 5 CALs | 1 |
| 7 | 8947 | 8947 | FA107A#8ZQ | iPAQ h5550 Pocket PC | 2 |
| 8 | 0 | 0 | FA136A#AC3 | hp 256MB SD memory | 2 |
| 9 | 0 | 0 | FA121A#AC3 | hp iPAQ Compact Flash Expansion Pack Plus | 2 |
| 10 | 0 | 0 | 271383-B21 | Compaq 320MB flash memory card | 1 |

**Figure 1 – Example of Up-sell and Cross-sell recommendations**
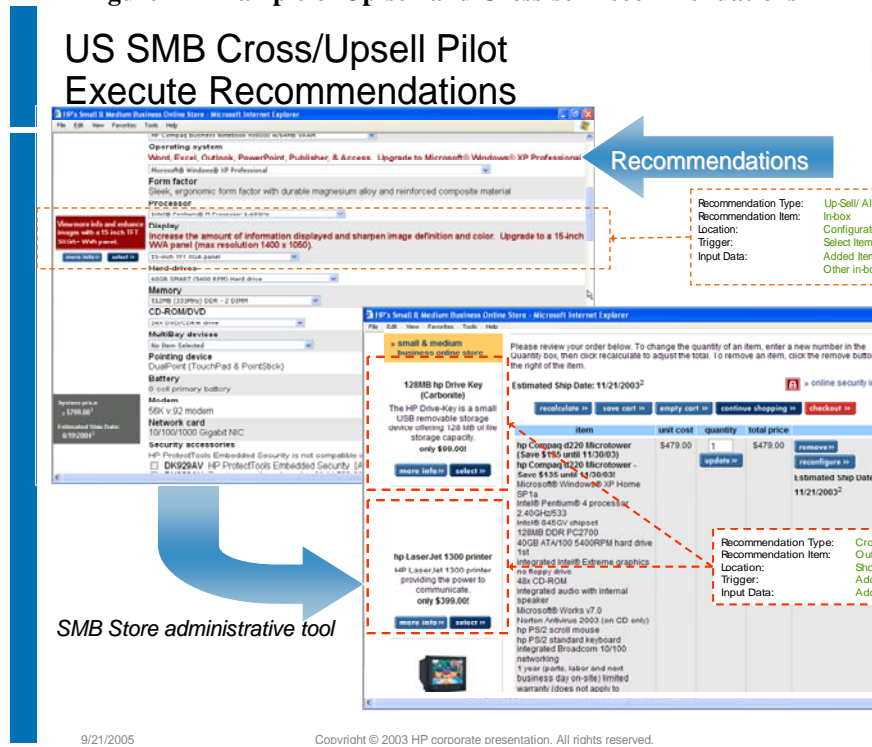


Figures 2 and 3 respectively illustrate cross-sells and up-sells associated with the data in Table 1 taking into account the quantities for each SKU in the order (refer to Table 1 for a description of the SKUs). This type of graphical representation of the market basket analysis is used to simplify the interpretation of results.  In the context of the example order represented in Table 1, the nodes of the graph in Figure 2 represent the SKUs and the lines along with the corresponding label represent the frequency of the paired relationship.  For example, the pair C7971A: 292887-001 occurred seven times  in the example order.  Note that the arc's thickness is proportional to its associated frequency; hence the important relationships are represented by the thicker lines. One simple characteristic of our graphical representation is that it represents quantities using the color of the origin node, this helps distinguish the quantities associated to each arc.

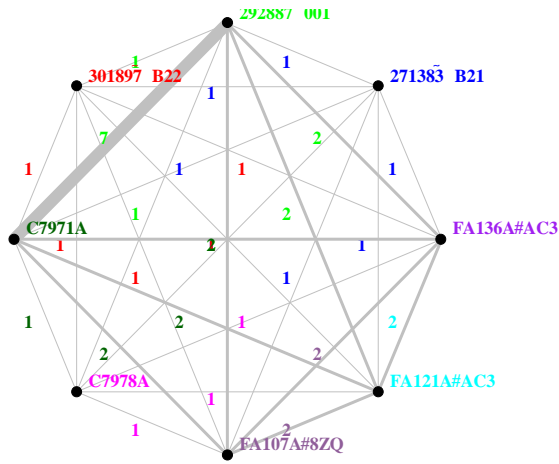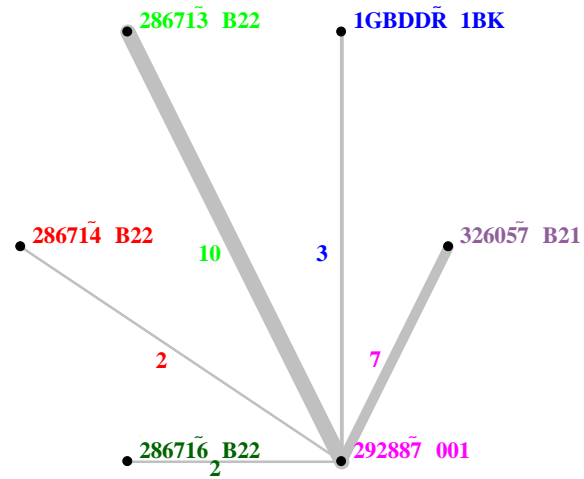**Figure 2 - Cross-sell market basket analysis for order 849608**



**Figure 3 - Up-sell market basket analysis for order 849608**



The analysis presented in Figures 2 and 3 was made for all orders of interest and results were aggregated representing the affinities found in the time span of interest. In summary, for any particular product and its associated SKUs, the solution reads from the database for all the data matching the set of selected SKUs, analyzes for up-sell and cross-sell, ranks recommendations by creating a ranking function that takes into account margin and revenue, and provides up to three alternate recommendations. Because of user navigation and page size restrictions, the business limited the number of recommendations per product to seven for up-sells and five for cross-sells.

Overall, the analysts first generated up to seven up-sell recommendations (unique to the target product) and likewise, up to five unique cross-sell recommendations. Then, an additional two alternate recommendations for each of the above generated recommendations were created to give business owners the option to overrule the originally suggested offers.

## Recommendation Generation

To accomplish the formerly mentioned requirements, one program with depends on several parameters was created to generate all recommendations for all products in the study. For example, one particular function takes product info and time-frame as an input and creates a three-tabbed spreadsheet-level recommendation file for the selected product. The three tabs are provided for the eventual need of substituting any of the winning recommendations (due to reasons such as new information regarding product availability, conflict with other promotions, or marketing's desire to feature a new product). Tab 1 includes all possible recommendations, the second tab includes only those that have a probability of acceptance above a certain acceptable limit, and the third one only includes the final recommendations.

Table 2 represents an example of the output generated by the program for a particular family of desktops. Note that the function created is completely automated in the sense that it generates an optimized list of all up-sell and cross-sell recommendations for the product in question. The program was optimized such that it usually takes less than 5 minutes to read all the data from Database, generate all recommendations for a product, and create the corresponding spreadsheet recommendation file.

**Table 2:  Example of the output generated (some sample fields only)**

| Cross-sell/Upsell | Recommendation type | Category | Offer description | Offer sku | Probability | Price | Revenue Rank | Margin Rank | Rank | Lead time |
|---|---|---|---|---|---|---|---|---|---|---|
| up sell | Primary | Memory | 512MB PC2-3200 (DDR2-400) | PM848AV | 0.51 | $200 | 111 | 111 | 111 | 7 |
| up sell | Secondary | Memory | 1GB PC2-3200 DDR-2 400 (2X512) | PM842AV | 0.25 | $310 | 109 | 110 | 109.5 | 7 |
| up sell | Secondary | Memory | 2GB PC2-3200 DDR2-400 (4x512MB) | PM846AV | 0.05 | $560 | 100 | 104 | 102 | 7 |
| up sell | Primary | Processor | Intel Pentium 4 520 w/HT (2.80GHz, 1M | PM675AV | 0.36 | $228 | 110 | 108 | 109 | 7 |
| up sell | Secondary | Processor | Intel Pentium 4 540 w/HT (3.20GHz, 1M | PM677AV | 0.22 | $298 | 108 | 106 | 107 | 7 |
| up sell | Secondary | Processor | Intel Pentium 4 550 w/HT (3.40GHz, 1M | PM678AV | 0.16 | $373 | 107 | 103 | 105 | 7 |
| cross sell | Primary | Mobility Thin & Light | HP Compaq Business Notebook nx6110 | PT602AA#A | 0.04 | $999 | 103 | 95 | 99 | 21 |
| cross sell | Secondary | Mobility Thin & Light | Configurable- HP Compaq Business Not | PD875AV | 0.04 | $510 | 87 | 77 | 82 | 15 |

The optimization method takes into account several other parameters provided by the users like minimum selection probability which puts a restriction on recommendations such that all recommendations are warranted to have a minimum empirical probability of acceptance in the training data set independent of the revenue or margin associated to it.  Once these low- probability recommendations are eliminated, the margin and the revenue are used to rank the remaining candidates.   The ranking function implemented for this purpose is discussed next.

Note that there are three ranking columns in Table 2; *Revenue Rank* gives the position of the recommendation if these were sorted by ascending revenue such that the largest expected revenue would get the largest rank; likewise for the *Margin Rank*. The third *Rank* is the weighted average of the former two.   The weights for revenue and margin were provided by business owners associated with the corresponding product category (printers, desktops etc).     In   mathematical   form,   the   function implemented to rank each recommendation ($R_R$) is as follows:

$$R_R = W_M \times R_M + W_R \times R_R \mid (W_M + W_R = 1)$$

where,

- $W_M \in [0,1]$ is the weight given to margin
- $R_M$ is the ranking position of recommendation R when sorted ascending by expected margin ($EM_R$)
- $W_R \in [0,1]$ is the weight given to revenue
- $R_R$ is the ranking position of recommendation R when sorted ascending by expected revenue ($ER_R$)

In the former definitions $ER_R = R_R \times p_R$, and $EM_R = M_R \times p_R$ where $p_R$ is the probability of acceptance associated to recommendation R.  In Table 1 $p_R$ is labeled *Probability* and is calculated using the following formula: $p_R = n_R/n_{sku}$ where $n_{SKU}$ is the number of product SKUs sold in the time frame analyzed, and $n_R$ = number of times that the recommendation R was selected in those $n_{SKU}$ products sold.

The reason for selecting this relatively simple ranking formula was to provide business owners with the option of giving relative importance to revenue or margin depending on their organizational goals or needs.  Some managers decided to use a 50%/50% weight split, while others put more emphasis on margin (75%).  The team considered several Ranking functions, but this was selected because it was easy to get business buy-in and allowed their participation in the development of the solutions implemented for their respective business sectors.  Besides, note that the header *Recommendation Type* refers to whether the recommendation for the *Category* (e.g. memory) is the first choice (Primary) as selected by the *Rank*, or an alternate recommendation (Secondary) for the same category.  Thus, in general the team tried to involve product managers as much as possible in the process of selecting the final recommendations for their products.

## 4.   Recommendation Review Process & Final File Generation

Analysts     presented     and     reviewed     the recommendations with business owners and members of North America eBusiness and segment marketing organizations to receive approval and finalize the target offer(s) to be deployed into production web-site and call-centers.   The process also required agreement on the proper website placement (left hand, inline, cart) and text to be used for the offer/callout within the website.

The deliverable of the review process was up to 7 up-sell  and  5  cross-sell  recommendations.    The recommendations   were   then   handed   over   to   the marketing team, which finalized the offers' wording and their respective placements.

### Implementation through Content Rendering Tool & Test Design

Once the recommendation file was finalized, it was  again  reviewed  by  analytics  and  segment

marketing teams for accuracy and then scheduled to be implemented on the SMB website at midnight of the agreed-upon date. Figure 1, displayed earlier, illustrates an example of up-sell recommendation in the configurator page and a cross-sell recommendation in the cart page.

Mentioned above, the team was restricted in time and further limited in enhancements that could be made to the site for the purpose of testing. Thus, randomization of customers upon entry to the site was not possible and the team could not have customers sent to test or control cells. Both the analyst and business teams involved understood this limitation and the groups proposed to compare pilot performance to some pre-pilot period.

## 5. Periodic Reporting and Final Pilot Performance Evaluation

The analytics team proposed a process to evaluate the effect of the recommendations. Before the pilot was launched, analytics team members and business owners agreed upon several metrics. Upon recommendation implementation in the web site and call center, the team monitored and evaluated them using several techniques in parallel. A final report was delivered to management regarding the weekly performance of the pilot.

### Financial and Attach Metrics

Performance of each metric was reviewed at the sku level and then rolled up to the product category level (e.g. all notebooks were aggregated to a "notebook" category) and ultimately to the pilot level. Due to the aforementioned challenge regarding the lack of a control cell, a month prior to the launch of the pilot was chosen as the Control period. Some of the metrics and their performance are discussed below:

- **Attach Rate** is the ratio of the Attach orders (orders with either a cross-sell or up-sell or both) to the total number of orders for that sku.

- **Attach revenue by Sku revenue** is the ratio of Attach revenue (sum of the up-sell revenue and cross-sell revenue) to the sku revenue. It reflects for every dollar of sku how many cents of attach revenue is generated.

- **Average Order value** is the ratio of the total order revenue to the total number of orders.

Table 3 below shows the recommendations significantly impacted the metrics discussed above for most of the categories. For example, we saw an 18% increase in *Attach Rate*, a 36% increase in *Attach Revenue by SKU revenue*, and more than a 2% increase in *Average order Value* for Desktops for Phase I. Similarly, we observed significant positive outcomes in Phase II.

### Statistical Significance

Control charts were created to demonstrate the significance of the change (if any) in proportion of acceptance for each recommended sku. Figure 4 shows an example of a graphical analysis made to evaluate the effectiveness of the recommendations. In particular this figure represents cross-sell analysis for Servers DL380 when orders were assisted by sales representatives. In this plot the x axis represents dates (40801=>2004/08/01) and each series describes the cross-sell % over time. Note that each point has a 90% confidence interval around the corresponding proportion.

These confidence intervals help evaluate whether there is a significant change in the proportion from one period to the other. For example, note that for the 2nd processor, the cross sell proportion from 10/16 to 10/30 is significantly lower than that from 11/16 to 11/30. For each product category, similar evaluations were made for combinations of: a) up-sell, cross-sell, and b) assisted, unassisted, or all sales together. All evaluations were programmed such that whenever the program was run these plots would be automatically updated for all products in the pilot. Thus, not only was the recommendation process automated, but also the process of evaluating the effectiveness for all recommendations.

**Table 3 – Sample of Product Performance on Key Metrics (% change from control period)**

| Product Category | Attach Rate | | Attach Revenue by SKU | | Average Order Value | |
|---|---|---|---|---|---|---|
| | Phase I | Phase II | Phase I | Phase II | Phase I | Phase II |
| Desktops | 18.2% | 8.2% | 36.8% | 10.5% | 2.2% | 16.2% |
| Printers | 1.8% | 20.6% | 59.3% | 30.2% | 26.0% | 7.1% |

**Figure 4 – Sequential confidence intervals for testing the effectiveness of recommendations**
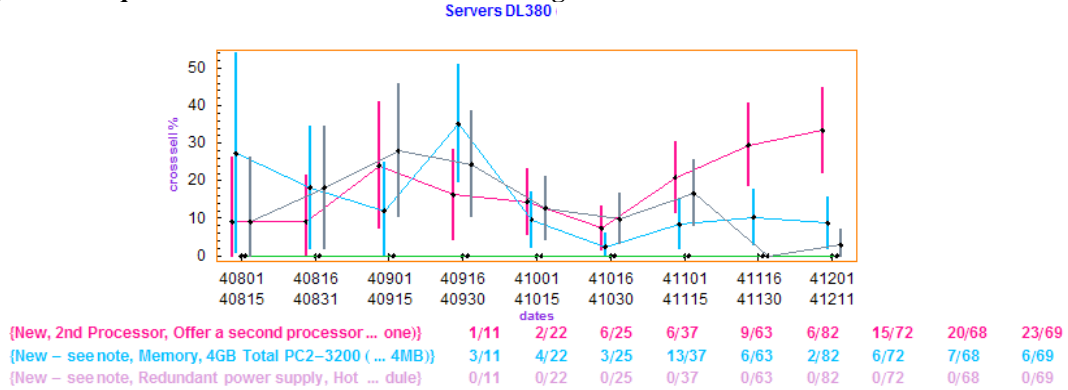


## Incremental Revenue

The team also evaluated the incremental revenue associated with each recommendation for each SKU. If there was a positive, significant change in the percentage of acceptance for a particular recommendation, we proceeded to estimate the incremental revenue associated with each recommendation. The method was as follows: Assuming that for any product SKU (SKUp) there are n recommendations SKUs (SKUp$_R$, R=1, 2…, n), the incremental revenue for each of the SKUp$_R$ is calculated using the following formula:

$$IRp_R = PRp_R \times (1 - CPp_R / PPp_R)$$

where,

- PRp$_R$ = revenue associated to SKU$_R$ when it is sold along with product p in the pilot period,
- CP$_{PR}$ = proportion of acceptance for SKU$_R$ when it is sold along with product p in the control period, and
- PPp$_R$ = proportion of acceptance for SKU$_R$ when it is sold along with product p in the pilot period.

Note that this method is not dependent on the length of the pilot and control periods. This is true because the differences in sample sizes associated with each period are being considered statistically on the test of hypotheses to compare the proportion of acceptances in the pilot and control periods.

To illustrate the idea consider this example that does not necessarily represent real data:
−Configuration sku: DR547AV-DX2
−Offer sku: DR689AV-512MB DDR 333MHz
−Control period data: DR689AV sold in 18 out of 99 orders of DR547AV-DX2 for a 18.2%
− Pilot period data: DR689AV sold in 58 out of 161 orders of DR547AV-DX2 for a 36.0%
−Revenue associated to the DR689AV in pilot period is $53,200
Incremental Revenue associated to DR689AV = 53,200 × (1- 18.2/36) = $26,350

A program to calculate all values of incremental revenue was created such that all of these values would be automatically updated for all products when it was time to evaluate the economical impact of the project.

## 6. Closing Comments

Overall, the pilot generated a ROI of $300K per month, a 3% increase in attach rate, 15% lift in Attach Revenue to SKU Revenue, and greater than 5% improvement in average order value. However, the team found benefits that may be even more important than the financial ones. The relatively new team was able forge strong relationships with the business owners, educate them on the benefits of analytics, and gain their support for future data analytics ventures.

## 7. References

1. R. Agrawal, H. Mannila, R. Srikant, H. Toivonen and A. I. Verkamo, "Fast Discovery of Association Rules", Advances in Knowledge Discovery and Data Mining, Chapter 12, AAAI/MIT Press, 1995

2. C. Aggarwal and P. Yu, "Mining Large Itemsets for Association Rules", Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, vol. 21, no. 1, March 1998.

3. S. Brin, R. Motwani, Craig Silverstein, "Beyond Market Baskets: Generalizing Association Rules to Correlations", Proceedings of 1997 ACM SIGMOD, Montreal, Canada, June 1997.

4. R. Agrawal, T. Imielinski, A. Swami, "Mining Associations between Sets of Items in Massive Databases", Proc. of the ACM SIGMOD Int'l Conference on Management of Data, Washington D.C., May 1993, 207-216.

5. R. Agrawal, R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases", Proceedings of the 20th International Conference on Very Large Data Bases, 1994

6. M. Zaki, M. Ogihara, "Theoretical Foundations of Association Rules", 3rd SIGMOD'98 Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD), Seattle, WA, June 1998.