**Lessons Learned**

**I. “Investigating Customer Buying Process”**

For the most part this project was straightforward. The concept was clear and the tools used to solve the problem at hand was relatively easy to learn. Three lessons learned from this, well, lesson, are:

1. Nuances in new software matter

2. Keep the decision tree levels low

3. Withhold assumptions

First, the biggest challenge for me in this project was learning the new platform. Rapid Miner tries to simplify processes to drag-drop-configure as much as possible. When I first started the project, I found I struggle with little things – little tick boxes being hidden in the UI, or descriptions that were not quite clear to me on how the component worked – things like this. At a high level, Rapid Miner is outstanding. In fact, I was able to immediately apply RapidMiner to a professional context. In doing so, what I learned is doing is important to learn the little things about new platforms. I learned a ton by working more than one project at a time. One of the biggest things I learned about configuring imports was around data type. It is really important to consider the data in each attribute and associated data types in RapidMiner. I ran into configuration errors that took me a little bit to figure out, only to find a data type mismatch occurred during import. RapidMiner provides components to address these inline or at import. So, it’s really important to understand the data you are working with BEFORE you start to import and develop in RapidMiner.

Second, decision trees are pretty straightforward conceptually. They provide a lot of insight very quickly. They can get complicated to interpret though if you have too many attributes or provide for too many levels in the tree. I found it much easier to derive quick high level insight into the data set, and therefore an understanding of Customer Buying Behavior, when I limited the number of levels in the decision tree. This is something that has to be assessed with each project since it’s really a function of the problem and number of attributes used to determine the configuration of the decision tree. The visualization in RapidMiner is a bit challenging to read when there are too many levels as well. So, there is a balance that has to be reached between the depth in the tree and the ability to derive interesting insight. This was a great first project because decision trees are really easy to setup.

Finally, and probably most importantly, it is important to refrain from making assumptions. On two levels, I entered the project making assumptions, that if not validated or debunked by the process, could have led me to incorrect assumptions. I agreed with some of the business level assumptions from the sales VP. This created a bias in my work, not that it inhibited my ability to complete the work, but that I had certain predetermined expectations of the outcome of the insight from the decision tree. Frankly, I think this resulted in me taking longer to confirm my research, because I had to fight my initial biases. My lesson here is, no matter how much business experience I might have, my subject matter expert level of insight is but one input to the process of interpreting data mining results. It’s important to be open minded to possible outcomes that you don’t expect.

Tasks performed in this project consisted of:

* Understanding the source data – review the data in the source documents. Look for anomalies, inconsistencies, missing values, devise approach for handling these issues
* Configuring RapidMiner – develop the flow for importing the data, preparing it for processing, and setting the right configuration attributes of the flow to feed the decision tree (like ID, Label, etc.)
* Using RapidMiner’s statistics and visualization capabilities to profile the data – there are many ways to review the quality of the data, and gain initial insight into the business problem just by looking at different visualizations. There are a number of useful charts. Try different types to answer different questions of the project.
* Using RapidMiner’s decision tree algorithm to visualize decision flow in the data – run the decision tree. Try different parameters for tree generation to see how to best present the data and make it insightful. Fewer nodes results in more aggregation in the leaf nodes. This can be very useful as high level indicators of customer behavior in this project.
* Creating a data driven report removing as much bias as possible and presenting based on the data results to directly, succinctly answer project questions – use the visualizations, statistics, and decision tree values to create an unbiased, data driven insight focused summary of the data answering the questions succinctly.

**II. “Predicting Product Cross-selling Success”**

This project was a little more complex adding in machine learning algorithms and predicted output. This was also my favorite so far. Three lessons learned from this project are:

1. Really research the process for attribute engineering and selection

2. Really research how to measure the quality of the output of each machine learning algorithm

3. Provide enough time on these projects for trial, failure, error and finally success

First, I got really stuck on the feature engineering and selection steps. Correlation and co-linearity are two different concepts. The correlation matrix is an important step in gaining insight into the relationship of each attribute to one another. It’s important to remove highly correlated attributes (one of two where correlation is .90 or higher). Remove the feature with the lower correlation of the pair to the label attribute. Also, make sure to remove attributes that don’t directly add value to the predictive output of the model, like ID’s. Finally, using some subject matter expertise could help in identifying other attributes that might not be required in the model based on the prediction you are trying to achieve. For instance in this project, I could comfortably remove profit margin as an input attribute since my prediction is focusing on customer buying likelihood – and in this case, the customer would have zero insight into profitability, nor would they care. They just want the best product. So, there is definitely science, but also some art in the attribute engineering and selection process.

Second, try multiple algorithms for solving the question at hand. Each algorithm will have different results based on the input data, though similar in nature. In this case, we are looking for the best fit, without over fit. Make sure to differentiate between models for classification and regression. Use the correct type of algorithm for the problem. Also, there are a lot of resources online to help understand the output of these algorithms to guide interpretation and insight. Use similar performance metrics across models to enable like comparison. In this project we used both R squared coefficient (R2) and Root Mean Squared Error (RMSE). It’s important to iterate through running each algorithm changing configuration parameters one at a time to note the impact on R2 or RMSE. When I iterated with enough input values, like Complexity for the Support Vector Machine (SVM) algorithm, I could plot a line that clearly demonstrated the optimal input C to generate the best R2 or RMSE. Rapid Miner makes this process really easy through the UI.

Finally, I expected to be able to quickly build a flow, run it, get the output and report on it. I didn’t provide enough time to account for mistakes, iterations with the configuration details of the algorithms, and general research and understanding of how the algorithms work and how to read the output. Rapid Miner provides a great UI for quickly building flows, processing the data, and reading the output. However, and as a result of the easy of drag and dropping components, it was also easy for me to miss some configuration detail that caused the flows to fail. I spent a lot of time learning how to optimize or configure each component through trial and error. Plan extra time to cleanse the input data, and create the flows and algorithms. One thing that makes RapidMiner excellent is the ability to quickly iterate through runs of the algorithms. As I found mistakes in my approach, I could quickly adjust parameters and run multiple rounds very quickly. Also, it’s really important to understand the overall CRISP-DM methodology. Moving too fast caused me to skip a couple steps that inadvertently resulted in churn and wasted time. Take the time to follow the steps. This is the biggest lesson learned on this project for me.