Project: Analyzing a Market Test

Overview

This project aims to perform an AB test experiment to generate sufficient data in order to assist a new management of a company to decide if it should roll out a new launch menu to its stores.

The software application used for the entire analytical process and model building is Alteryx.

The following steps were taken to get the desired results.

Step 1: Plan Your Analysis

Since profit is represented in the given dataset as the gross_margin variable, the performance metric I intend to use is WEEKLY GROSS MARGIN (at least 18% increase in profit growth compared to the comparative period – the ensuing jump in marketing budget – while compared to the control stores; otherwise known as incremental lift). The predicted impact to profitability should be high enough to justify the increased marketing budget.

If the 10 treatment stores (units) in the 2 cities (Denver and Chicago) experienced such an increase, then the new management can make an informed decision to apply the menu changes to all stores in the company's chain, otherwise it would not.

The comparison/baseline period (used to compare or measure performance metric in comparison with the actual test/experimental period of the A/B experiment) is from 29th April, 2015 to 21st July, 2015 (12 weeks), while the actual duration the test was run was from 29th April, 2016 to 21st July, 2016 – also a period of 12 weeks.

The data will be aggregated at the weekly level since such businesses usually consider patronage at the weekly level, assuming that the average customer visits a store once every week. More so, aggregating at the weekly level helps to remove missing values which will cause the A/B Trends and Control tools to error.

Step 2: Clean Up Your Data

The following three raw data files were given to us:

- RoundRoastersTransactions file: Contains transaction data for all stores from 2015-January-21 to 2016-August-18

- round-roaster-stores file: Contains a listing of all Round Roasters stores with all information about them
- treatment-stores file: Contains a listing of the 10 stores (5 in each market) that were used as test markets.

Using these, we will create three data files/tables, namely:

1. <u>Weekly store traffic</u> data for A/B Trends Tool, which produces our seasonality and trend indices to help us match our treatment and control stores. This will contain the field **number of invoices per week** at each store. We will use this data with the A/B Trends tool to create measures of trend and seasonal patterns that can be used in helping to match treatment to control units. For our analysis, we will use the **week** and **week_start** fields to calculate the **week_end** date field (which is the field we need for analysis)

   To begin, I used an INPUT DATA tool to introduce the RoundRoastersTransactions file. Next, I connected a SELECT tool to change appropriate fields to the correct data type (being a csv file).

   The trend tool is used to create trend and seasonality variables to use as control variables. To do this, we will need at least 52 weeks of data, plus the number of weeks we normally select in the tool to calculate trend, before the beginning of the test start date. For the project, we are asked to use 12 weeks to calculate trend, so we'll need 64 (52 + 12) weeks of data prior to the test start date.
   And since the test lasts for 12 weeks, this means we'll need a total of 64 + 12 = 76 weeks of data.

   So, I used a FILTER tool to filter all stores to remove data values where the **Region** field is blank and to have 76 weeks of data. Counting 76 weeks backwards from the end of the test period, 2016-July-21, we arrive at 2015-02-06. Thus, every other date records outside this range were filtered out using the following expression in the FILTER TOOL:
   !IsEmpty([Invoice Date]) AND [Invoice Date]>= '2015-02-06' AND [Invoice Date]<='2016-07-21'. I matched on **Region** in the FILTER tool.

   Next, to assist in summarizing transaction data to a store and week level, I used the FORMULA TOOL and some complex formulas in alteryx to create a **Week** number field, a **Week_Start** date field and a **Week_End** date field and also a **Product_flag** field that will be used to identify transactions that included the new menu launch.

   To avoid skewing our data, we only want stores that have 76 weeks of data. So I used a SUMMARIZE tool to count the number of weeks for each store for checking purposes. Using the summarize tool, I grouped by **Region** and then counted the distinct number of weeks related to each store.

   Then, I used the FILTER TOOL in alteryx to keep only those stores that have 76 weeks of data.

Then using a JOIN TOOL, I added back the data (while removing fields not needed), but only for those stores that have 76 weeks of data. This resulted in having a dataset that contains only stores with 76 weeks of data.

Now, we are only interested in the *sales* and *gross margin* fields for the new menu since we are testing the change that occurs with an introduction of the new menu in the business chain of stores. However, I still used total *sales* when calculating the stores' sales transactions used to match control and treatment stores since total sales is a representative of foot traffic for a store.
I used unique sales and invoices as a proxy for store traffic.

Next, using the SUMMARIZE tool, I aggregated the data to achieve one invoice per row. We want the total *gross margin* and *sales* amount per invoice whether the invoice contains a new menu or not. So I grouped by the *Region, store ID, invoice number* and *invoice date* fields, and then I calculated the sum of the *gross margin* and *sales* fields, and then I used the CountDistinctNonNull for the *Product_flag* field (such that the null values will be converted to a 0 for invoices without a new menu involvement and 1 for invoices involving a new menu launch).

Store foot traffic (determined by calculating the number of invoices per store per week) will be used to calculate trends and seasonality for analysis. Thus, I used the SUMMARISE tool to count the number of invoices per store per week.

This gave me a final dataset showing the number of invoices per store per week along with the date information – that is, store traffic dataset. Finally, I used an OUTPUT DATA tool to save the dataset.

2. Store list data for A/B Controls tool, which produces which control stores to match with our treatment stores along with results from the A/B Trends Tool. This table will contain the fields *stores, regions,* and the *group* that the store is in – treatment or control. This data will be used with the A/B Controls tool, using information from the A/B Trends tool to create control/treatment periods for the analysis.

I began by using two INPUT DATA tools to bring the round-roaster-stores and treat-stores files into the canvass. Then I used a SELECT tool to change required fields into the appropriate data types whilst deselecting unnecessary fields. I made sure to retain the exact same fields in both data files to ease matching.

Then I used a FORMULA tool to create a new field called *Test_group* to identify which stores are treatment (for the treatment-stores dataset). I then joined both datasets using a JOIN tool – joining by *storeID.*

Since we are joining the treatments store data with data from all the stores, the JOIN tool would produce two data streams we are interested in. The L-output would be the control unit stores, while the J-output would be the treatment stores. Noticing that the L-output was missing the *Test group* field, I saw the need to create another field.

Thus, since the stores in the round-roaster-stores file are the Control stores, I used another FORMULA tool (connected to the L-output of the JOIN tool) to create a field to identify the control candidates/unit stores for this dataset with same name of **Test_group.**

Next, I combined both tables together (the L- and J- outputs) using a UNION tool (since both tables contain the same fields). This gives us the dataset that we wanted – a listing of stores, with the region and test group associated with each store. Then I saved the dataset using an OUTPUT DATA tool in alteryx.

3. <u>Sales data</u> for A/B Analysis tool, which produces the final results. We will aggregate the sales data to show the **weekly sales** and **gross_margin** per store of all transactions that included the new menu to perform the analysis.

Since we already filtered out the transactions that included the new menu when creating the WEEKLY STORE TRAFFIC dataset, we can reuse that output for the sales data.

I aggregated the data to the store week level by adding a SUMMARISE tool and connecting it to the output of the SUMMARISE tool used to **CountDistinctNonNull.** FILTER tool where we checked for new menu transactions. I grouped by **storeID, week, week_start, week_end,** and then sum by **gross margin** and **sales.** This gave me a dataset that shows the gross margin and sales transactions that included a new menu per week. I connected an OUTPUT DATA tool to save the file for future use.

Using the three data files created, we will then match our treatment and control stores to calculate the sales lift.

Step 3: Match Treatment and Control Units

In this section, we are interested in creating the trend and seasonality variables, and using them along with the other control variable(s) to match two control units to each treatment unit. Treatment stores were matched to control stores in the same region. I also endeavored to c*alculate the number of transactions per store per week and used 12 periods to calculate trend and seasonality.*

Important variables here include
  i.   The stores sales trends: Matching the stores based on foot traffic.
  ii. Seasonality of sales: Matching the stores with similar seasonal traffic patterns and geographic region to control for regional customer preferences.

Looking at the round-roasters-stores and comparing with the treatment-stores files, two variables that appear relevant as potential control variables to be used are **Region, Sq_Ft** and **AvgMonthSales.**

Since in an AB Analysis we use the correlation matrix to find the most correlated variable to the performance metric to include in the AB CONTROLS tool to help find the best matches, I had to check these potential control variables against the performance metric (**gross margin**).

For the categorical variable **Region,** I used a PLOT OF MEANS tool to check its relevance with **gross margin,** and I noticed a very good correlation.

While for the numeric potential control variables, I employed the ASSOCIATION ANALYSIS tool. I noticed that there was some inner correlation of – 0.05 between **AvgMonthSales** and **Sq_Ft** (lurking variables) whilst virtually no correlation exists between **Sq_Ft** and **gross margin.**

Meanwhile, there is a very good correlation of 0.99 between **AvgMonthSales** and **gross margin.**

More so, looking at the R-output of the ASSOCIATION ANALYSIS tool, the Matrix of corresponding p-values between **AvgMonthSales** vs **Sq_Ft** and **Sq_Ft** vs **gross margin** are much higher than 0.05, while that between **AvgMonthSales** vs **gross margin** is much lower than 0.05 – a good thing!

Hence, I decided to use **Region** and **AvgMonthSales** as my control variables.

Now, since numeric measures are needed in order to match treatment stores to control candidates (control stores), two of the best measures to use are Trends and Seasonality.

Thus, I brought in the WEEKLY STORE TRAFFIC dataset into a new canvass using the INPUT DATA tool. Then I connected the AB TRENDS tool to the dataset and configured it properly, and I used 12 as the number of periods – in this project, weeks – to calculate the trend (as required for this project).
The test start date used is 2016-April-29 as required.

The performance metric for this tool is the invoice count per week which represent weekly foot traffic, which had been earlier created in the data preparation stage with the help of a SUMMARIZE tool.

After configuring the tool properly, I ran the workflow to ensure I have enough data and periods to calculate trends accurately.

Next, I brought in my STORE LIST dataset. Then using a JOIN tool, I connected both datasets together, joining using the **storeID** field.

Attempting to use **region** as a variable, since it is a categorical variable, the way to approach the data is to match control and treatment separately by each region. For each region. The steps would be the same.

First, I filtered out the region's specific data using a FILTER tool, wherein I used the basic filter option and **Region** = west.

Again, in order to use the AB CONTROLS tool, we need to separate the treatment stores from the control group. Thus, I connected another FILTER tool, wherein I used the basic filter: **Test_group** does not equal control. This splitted the data into two streams – one with the treatment stores (T-output) and another with the control stores (F-output). Now, we can use the AB CONTROLS tool to use the trend and seasonality data calculated by the AB TRENDS tool to match each treatment store to two control stores for each region.

Hence, I dragged the AB CONTROLS tool to the canvass and connected the data with the treatment stores to the T-input while the data with all the stores (treatment and control) was connected to the D-input and configured the tool appropriately, selecting both Trend and Seasonality to match treatment to control units, and I selected 2 control stores for each treatment unit, and repeated these steps for the second region in the dataset – Central (in this project, only two regions exist – West and Central).

Next, I combined all the data from both regions using the UNION tool since both data streams are the same, and usually no configuration of the UNION tool is required. I then ran the workflow and took a look at the output. Looking at my output, I could see that I have the control-treatment pairs. However, we need to add the stores attribute information from the stores information dataset (STORE LIST) to perform the analysis.

Hence, I connected the STORE LIST dataset to the R-input of a JOIN tool, while the output of the UNION tool was connected to the R-input of the JOIN tool, and in the configuration panel of the JOIN tool I selected treatments for the Left and *StoreID* for the Right whilst removing unnecessary fields. Upon running the workflow, I noticed that we now have a dataset that can be used to perform the analysis – a list of control and treatment pairs with their region and associated test group. Thus, I saved the dataset using an OUTPUT DATA tool.

We are now ready to perform the analysis and summarise the results of the test.

Step 4: Analysis and Writeup

We are going to use the two datasets that we previously created. The control-treatments pair data will be used to provide the identification data concerning which 2 pairs of stores will control a treatment unit, and the stores sales analysis dataset will be used to provide performance data associated with the control and treatment stores.

Thus, using two INPUT DATA tools, I brought both datasets into a new canvass in alteryx. Next, I connected an AB ANALYSIS tool. This tool helps to provide reports that detail how the treatment units performed in comparison to the control units. Thus both datasets were connected appropriately to the tool. Clicking on the Test Dates tab in the tool, the dates of the test/experiment (not of our dataset) were entered: Start date: 29th April, 2016 and End date: 21st July, 2016. Next, a BROWSE tool was added to the I-output of the tool (which provides a html report) and another BROWSE tool added to the O-output (which produces a textual report). Then the workflow was run.

Looking at the I-output of the AB ANALYSIS tool, from the Lift Percentage value, it can be seen that there is an overall incremental lift of 34.5% (greater than 18% as required by management), which implies that the average customer patronage per week in the treatment stores increased by 34.5%.
Again, it can also be seen that the average customer spend per week will increase by $646.60.

While the significance level (interpreted as 1 – p-value) of 99.1% shows that the test/experiment was highly significant. That is, it is highly unlikely that the new menu launch did not increase sales.

The Time Comparison Plot compares the sales trend both before and after the new menu was launched. The plot above shows that both the treatment and control stores had similar trend during the historical period, but during the test period, the treatment stores experience an increase or lift in sales.

More so, the Dot Plot (which shows the actual differences between individual pairs of stores, wherein each treatment store is plotted against all the control stores matched to it). It can be seen, as shown in the visualization below, that all the treatment stores out-performed each matching the control pair of stores – a 100% performance!

Now, looking at the O-output Test Summary, it can be seen that the average percentage change in Gross Margin was 39.7% for the treatment units in the test period relative to the comparison period. This same measure was 4.6% for the control units, with the difference between the treatment and control units being 35.1%, which is highly statistically significant.

And just like we observed in the I-output report, a comparison of the treatment-control pairs indicates an average lift in Gross Margin for the treatment units over the control units of 34.5%, which results in an expected impact of 647 on Gross Margin, with 100.0% of the treatment-control pairs exhibiting a positive lift for the treatment units.

Some other similar and relevant information regarding the AB ANALYSIS tool test results with their associated explanations are as shown below.

Hence, from the good feedback we got from the experiment as presented above, I would recommend that the company rolls out the updated menu to all stores, and it will be a good business move made by the new management.