Written report

Analysis of Superstore shop's profits based on sales

Goda Beinortaite
2023-02-08

Table of contents

1.Execution summary

A superstore operating within the US has given us the opportunity to identify what works best for them in terms of sales and making profit.
Given the insights gained from this analysis, the Superstore can choose to remove non-profitable products or invest in marketing efforts for products, segments and geographical areas that are driving their profit.

- **Technology** and **Home Office** have on average a high profit margin, while **Furniture** are being sold at low margin or even at a loss for some its sub-categories.
- **Technology** is the only category that contains sub-categories with only positive profit margin values.
- The **Consume**r segment is driving most of the Superstore's sales and across all segments Technology is contributing the most to revenue.
- Over 50% of their profit is being made in **California and New York**. This makes this store quite geographically dependent on keeping up their sales here.
- Their sales in **Texas** (which is their 3rd highest state in sales) is on aggregate making a loss.
- **Tables, Appliances and Binders** are product categories being sold at a loss with lowest profit margin values out of all sub-categories.
- Some sub-categories, namely **machines and tables**, were the main drivers of loss in profits in worst profit states like **Texas and Ohio**, as almost all records of these products netted negative profits.
- **Discounts** really affect profit. Bigger discount gives minus profit to the store. Too many discounts are offered for some specific categories and in some states like Texas and Ohio.

2.Introduction

2.1.Finding data

The dataset for this project was found independently by myself. The main criteria was to have similar data as seen during the course in order to be able to apply different Data Analytics techniques. Also, the chosen dataset has lots of information, is interesting and easy to understand.

2.2.Data source

I used public e-commerce dataset of Superstore from this source:
https://www.kaggle.com/datasets/vivek468/superstore-dataset-final
https://community.tableau.com/s/question/0D54T00000CWeX8SAL/sample-superstore-sales-excelxls

It contains ~10,000 transaction records from January 2014 to January 2018, meaning time span of 4 years. The data was collected from one e-commerce merchant as a part of educational purpose.

The data from this particular store spans over 1457 days with sales totaling at $2,297,201 and a profit of $286,397.
There are 1862 unique items in their inventory which are divided into 3 categories, all of which are being sold across 49 states in the US.

Geography: US
Currency: Dollar
Time period: 01/2014 – 01/2018
Unit of analysis: E-Commerce orders of purchases
Sensitive data: Customer names

2.3. Targeted audience and key stakeholders

This Superstore company’s product and sales analysis targets sales executive team.

2.4. Metadata

Row ID => Unique ID for each row.
Order ID => Unique Order ID for each Customer.
Order Date => Order Date of the product.
Ship Date => Shipping Date of the Product.
Ship Mode=> Shipping Mode specified by the Customer.
Customer ID => Unique ID to identify each Customer.
Customer Name => Name of the Customer.
Segment => The segment where the Customer belongs.
Country => Country of residence of the Customer.
City => City of residence of the Customer.
State => State of residence of the Customer.
Postal Code => Postal Code of every Customer.
Region => Region where the Customer belongs.
Product ID => Unique ID of the Product.
Category => Category of the product ordered.
Sub-Category => Sub-Category of the product ordered.
Product Name => Name of the Product
Sales => Sales of the Product.
Quantity => Quantity of the Product.
Discount => Discount provided.
Profit => Profit/Loss incurred.

2.5. Purpose of analysis

The objective of this analysis is to provide data insights to the Superstore sales team who are planning to grow the sales revenue and increase company's profits based on sales. To do this, they should get a better understanding of customer behaviour by analyzing the historical sales data from 2014 to 2018.
Our goal is to gain insights into the performance of the store and identify opportunities for improvement.

2.6. Approach

A structured approach was followed by listing the six steps – ask, prepare, process, analyse, share and act.
I used BigQuery and SQL to write queries and extract data. Also, the notebook includes visualizations and analysis of data from an online superstore. I used data visualization tools like LookerStudio and Excel to explore the data and draw meaningful conclusions.

Utilization of techniques learnt in the course:
Customer segmentation, RFM;
Application of Pareto principle;
Cohorts, retention, CLV;
Regression model to predict Profit;
Data visualization by using LookerStudio/Excel;

3. Analysis

PHASE 1: ASK

Business task can be formulated by these questions:
- What kind of customers do we have the most?
- What is correlation between sales and profit?
- What are the most and least profitable product categories?
- What are the most and least profitable states?

• How do the discounts contribute to the profit?

PHASE 2: PREPARE

For this analysis I am going to focus mostly on the last columns of dataset – sales revenue and profit. Secondary important columns will be product name and category, quantity, state, discount and my manually created column - profit margin (profit divided by sales).

PHASE 3: PROCESS

Dealing with missing values:
The given dataset contains only six missing values "NA" for a product ID = TEC-AC-10004659.
I decided to remove them, since in this case it is less than 1% of the entire dataset.
Looks like this product was registered in the system but no amounts recorded for that.

Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer N	Segment	Country	City	State	Postal Code	Region	Product ID	Category	Sub-Catego	Product Nai	Sales	Quantity	Discount	Profit
182	CA-2014-1661	2014-12-05	2014-12-09	Second Clas	DK-13150	David Kend	Corporate	United Stat	Decatur	Illinois	62521	Central	TEC-AC-100	Technology	Accessories	Imation Sec	0	0	0	0
431	US-2016-1237	2016-04-15	2016-04-21	Standard Cl	RB-19795	Ross Baird	Home Offic	United Stat	Gastonia	North Carol	28052	South	TEC-AC-100	Technology	Accessories	Imation Sec	0	0	0	0
432	US-2016-1237	2016-04-15	2016-04-21	Standard Cl	RB-19795	Ross Baird	Home Offic	United Stat	Gastonia	North Carol	28052	South	TEC-AC-100	Technology	Accessories	Imation Sec	0	0	0	0
1407	US-2014-1184	2014-04-06	2014-04-08	First Class	SD-20485	Shirley Dan	Home Offic	United Stat	Philadelphi	Pennsylvani	19143	East	TEC-AC-100	Technology	Accessories	Imation Sec	0	0	0	0
1970	CA-2017-1174	2017-09-23	2017-09-29	Standard Cl	BD-11320	Bill Donatel	Consumer	United Stat	Tulsa	Oklahoma	74133	Central	TEC-AC-100	Technology	Accessories	Imation Sec	0	0	0	0
1972	CA-2017-1402	2017-05-06	2017-05-11	Standard Cl	ML-17755	Max Ludwig	Home Offic	United Stat	Chicago	Illinois	60623	Central	TEC-AC-100	Technology	Accessories	Imation Sec	0	0	0	0

Dealing with the duplicates:
There is one duplicated entry in the dataset.
Even though at the first sight it looks like there are no duplicates, this is not true - when we look more carefully we can see that after dropping row_id (first column) not all rows are distinct and one row repeats.

The query below checks whether there are any duplicate rows.
As the total row number is higher than the distinct row number, we know that this dataset contains duplicates:

```
SELECT
(SELECT COUN(1) FROM (SELECT DISTINCT * FROM `my-capstone-project-376621.Superstore_db.Superstore_main`)) AS distinct_rows,
(SELECT COUNT(1) FROM `my-capstone-project-376621.Superstore_db.Superstore_main`) AS total_rows
```

Row	distinct_rows	total_rowsselect
1	9993	9994

```
CREATE OR REPLACE TABLE `my-capstone-project-376621.Superstore_db.Superstore_main_cleaned`
AS
SELECT
DISTINCT *
FROM `my-capstone-project-376621.Superstore_db.Superstore_main`
```

The above query will first look for the existence of the table and then replace it with the result if it exists, or creates the table if it doesn't with only the distinct rows.
Given the structure of the dataset I will assume that this was a faulty entry which needs to be removed.

PHASE 4: ANALYSE

Customer segmentation, RFM

I decided to have 8 categories of clients: Best customers, Loyal customers, Potential loyalists, Promising, Big spenders, Almost lost, Lost customers, At risk.

Definitions of categories:

Best customers
Who Bought most recently, most often, and are heavy spenders.
How They can become early adopters for new products and will help promote your brand.

Loyal customers
Who Buys most frequently (high F).
How Offer membership or loyalty programs or recommend related products to upsell them.

Potential loyalists
Who Recent customers with average frequency and who spent a good amount.
How Offer membership or loyalty programs or recommend related products to upsell them.

Big spenders
Who Spend the most (high M).
How Suggest discounts if spend more, market most expensive products.

Promising
Who Recent shoppers but haven't spent much, low frequency so far.
How Offer free trials.

At risk
Who Purchased often and spent big amounts, but haven't purchased recently.
How Send them personalized reactivation campaigns to reconnect, and offer renewals and helpful products to encourage another purchase.

Almost lost
Who Haven't purchased for some time, purchase frequency and monetary value below average. Will lose if not reactivated.
How Win them back via renewals or new products.

Lost
Who Last purchase long ago, purchased few and spent less.
How Don't spend too much trying to re-acquire.

Steps completed in query:
1. Compute for recency, frequency, and monetary values per customer

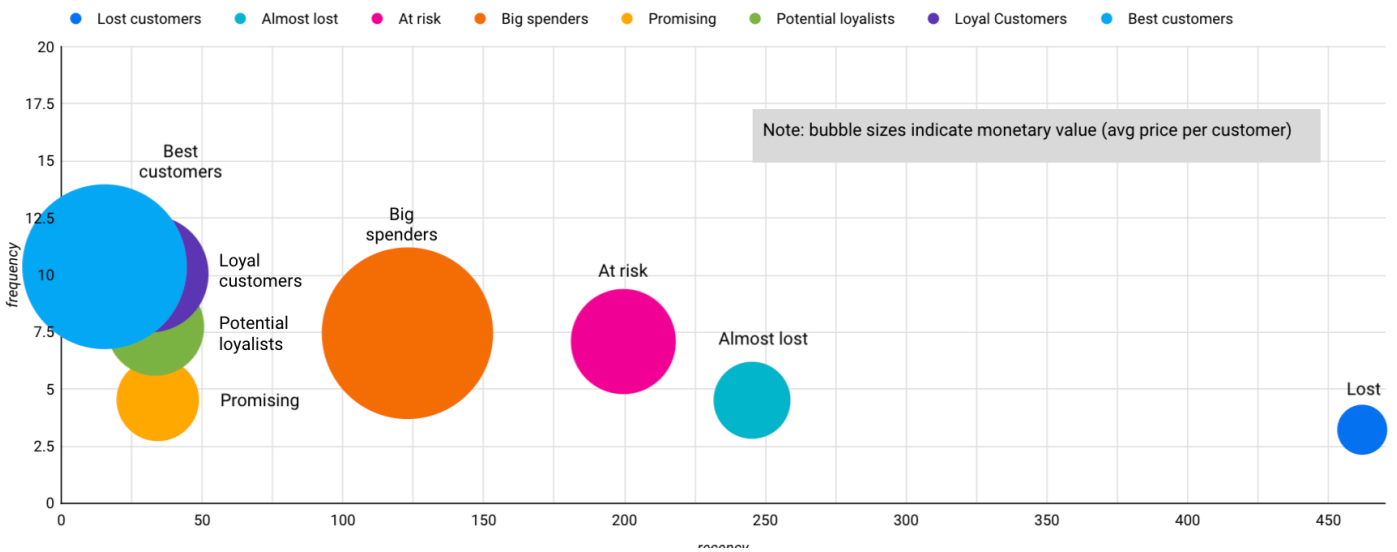
2. Determine quintiles for each RFM metric

3. Assign scores for each RFM metric

4. Define the RFM segments using the scores in step 3

Please see Appendix 1 for query.

Screenshots from LookerStudio visualisations:



RFM segment	Unique customers	% Unique customers	Recency (avg days ago)	Frequency (avg)	Monetary (avg EUR per cutomer)
Big spenders	168	21%	123	7	6,447
Promising	144	18%	34	5	1,494
Almost lost	126	16%	245	5	1,302
At risk	125	16%	200	7	2,426
Potential loyalists	93	12%	33	8	2,093
Lost customers	67	8%	462	3	549
Loyal Customers	40	5%	32	10	2,995
Best customers	30	4%	15	10	5,948
Grand total	**793**	**100%**	**148**	**6**	**2,894**

Main insights:
- Biggest percentage of customers belong to "Big spenders" which means that we should offer them more discounts and market the most expensive products to them. In such case we could increase recency and frequency.

- Second biggest group is "Promising". Recent shoppers but haven't spent much, low frequency so far. Offer free trials, new onboarding of the products.

- Big percentage of customers belong to "Almost lost" and "At risk". Haven't purchased for some time, purchase frequency and monetary value below average. Will lose if not reactivated. Win them back via renewals or new products, use aggressive price incentives.

- We have small groups of "Potential loyalists", "Loyal customers" and "Best customers", therefore we should focus more on these fragile categories and try to increase the loyal customer base. Introduce reward system, ask for advice and listen to it, consistently engage them.

Application of Pareto principle

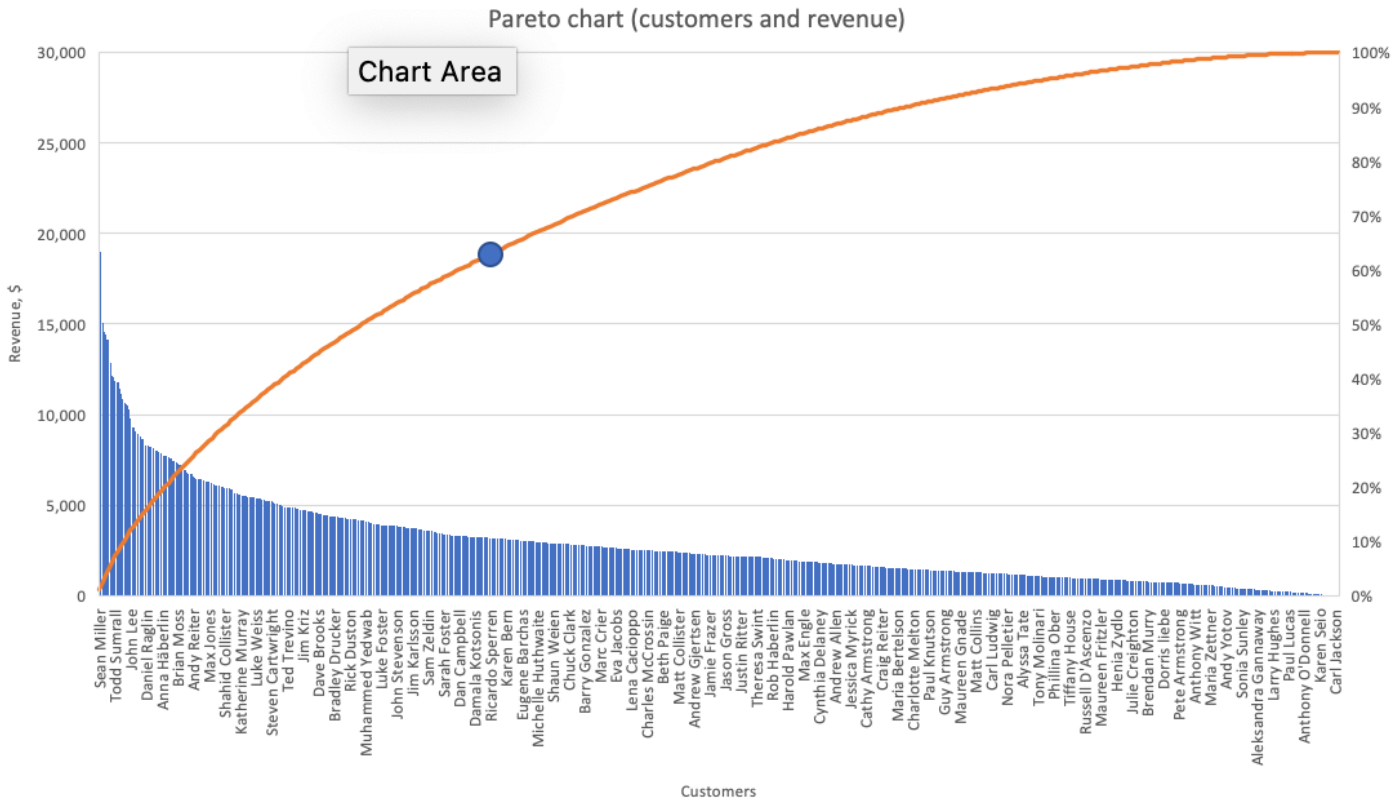
Pareto principle for customers and sales

Let's prove that top 20% customers generate 80% sales in order to be able to focus the team efforts on them and maybe drive sales revenue even further.

We pull the sales numbers by customer, order them in descending order, and count cumulative sales next to it, as well as total sales for all customers.
We use this data to calculate the percentage of cumulative sales to be able to identify where it reaches 80% of total sales.

The first top 270 customers (out of 793) comprise roughly 65% of total revenue. That is ~35% of all customers, so essentially we can conclude that not strong Pareto principle exists in this case.
This cumulative distribution of sales revenue in terms of individual customers results in an unbalanced distribution, where 35% customers control 65% of the sales.

Please see Appendix 2 for query.



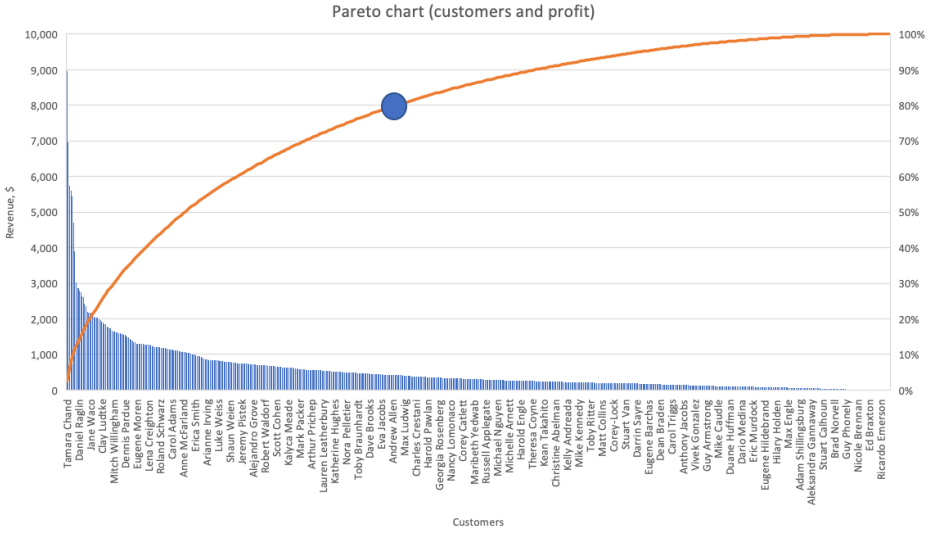
Pareto principle for customers and profit

Let's prove that top 20% customers generate 80% profit in order to be able to focus the team efforts on them and maybe drive profit even further.

We pull the profit numbers by customer, order them in descending order, and count cumulative profit next to it, as well as total profit for all customers.
We use this data to calculate the percentage of cumulative profit to be able to identify where it reaches 80% of total profit.

The first top 153 customers (out of 793) comprise roughly 80% of total profit. That is ~20% of all customers, so essentially we can conclude that Pareto principle exists in this case.
This cumulative distribution of profit in terms of individual customers results in an unbalanced distribution, where 20% customers control 80% of the profit.

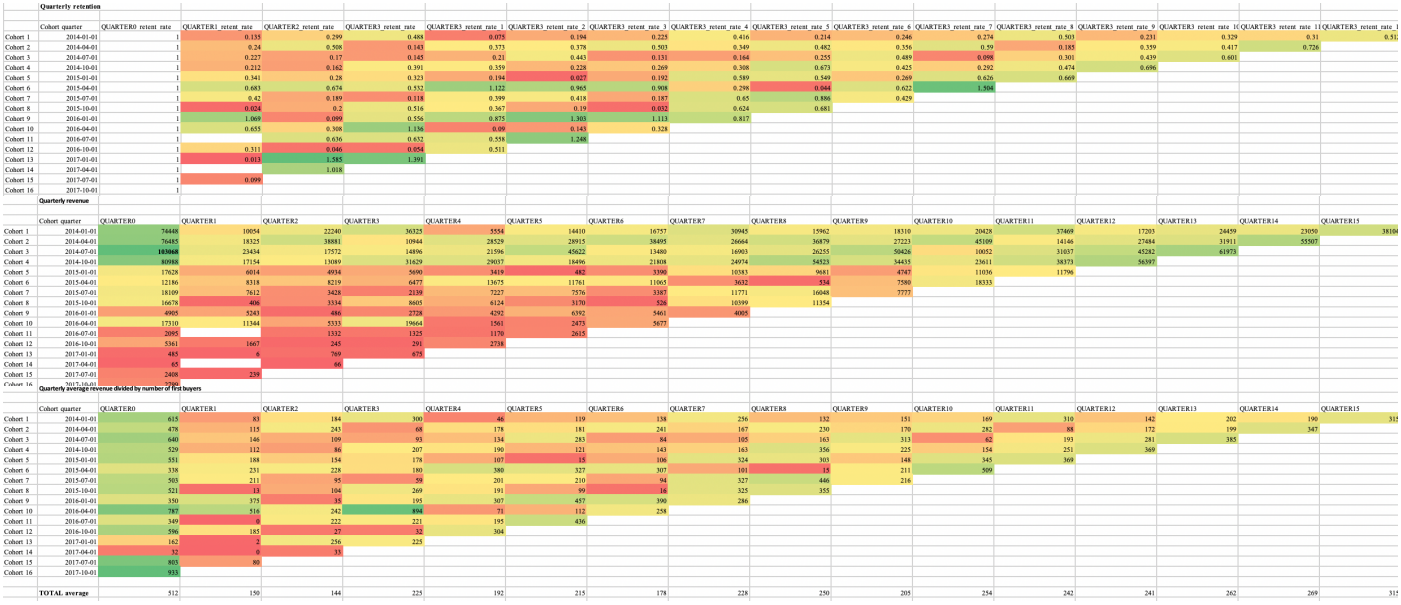
Please see Appendix 3 for query.

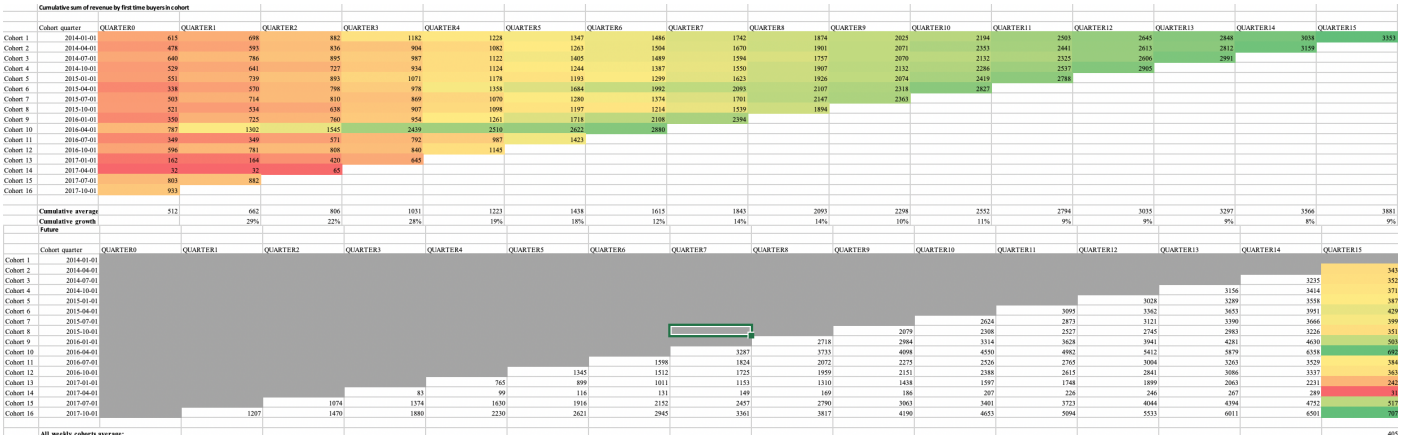


Cohorts, retention, CLV

Please see Appendix 4 for query.

Extracts from analysis in Excel:





Cumulative sum of revenue by first time buyers in cohort

Cohort quarter	QUARTER0	QUARTER1	QUARTER2	QUARTER3	QUARTER4	QUARTER5	QUARTER6	QUARTER7	QUARTER8	QUARTER9	QUARTER10	QUARTER11	QUARTER12	QUARTER13	QUARTER14	QUARTER15
Cohort 1	2014-01-01	615	698	882	1182	1228	1347	1486	1742	1874	2025	2194	2503	2645	2848	3038	3353
Cohort 2	2014-04-01	478	593	836	904	1082	1263	1504	1670	1901	2071	2353	2441	2615	2812	3159	
Cohort 3	2014-07-01	640	786	895	987	1122	1405	1489	1594	1757	2070	2132	2325	2606	2991		
Cohort 4	2014-10-01	529	641	727	934	1124	1244	1387	1550	1907	2132	2286	2537	2905			
Cohort 5	2015-01-01	551	739	893	1071	1178	1193	1299	1623	1926	2074	2419	2788				
Cohort 6	2015-04-01	338	570	798	978	1358	1684	1992	2093	2107	2318	2827					
Cohort 7	2015-07-01	503	714	810	869	1070	1280	1374	1701	2147	2363						
Cohort 8	2015-10-01	521	534	638	907	1098	1197	1214	1539	1894							
Cohort 9	2016-01-01	350	725	760	954	1261	1718	2108	2394								
Cohort 10	2016-04-01	787	1302	1545	2439	2510	2622	2880									
Cohort 11	2016-07-01	349	349	571	792	987	1423										
Cohort 12	2016-10-01	596	781	808	840	1145											
Cohort 13	2017-01-01	162	164	420	645												
Cohort 14	2017-04-01	32	32	65													
Cohort 15	2017-07-01	803	882														
Cohort 16	2017-10-01	933															
Cumulative average		512	662	806	1031	1223	1438	1615	1843	2093	2298	2552	2794	3035	3297	3566	3881
Cumulative growth			29%	22%	28%	19%	18%	12%	14%	14%	10%	11%	9%	9%	9%	8%	9%

Future

Cohort quarter	QUARTER0	QUARTER1	QUARTER2	QUARTER3	QUARTER4	QUARTER5	QUARTER6	QUARTER7	QUARTER8	QUARTER9	QUARTER10	QUARTER11	QUARTER12	QUARTER13	QUARTER14	QUARTER15
Cohort 1	2014-01-01															
Cohort 2	2014-04-01															343
Cohort 3	2014-07-01														3235	352
Cohort 4	2014-10-01													3156	3414	371
Cohort 5	2015-01-01												3028	3289	3558	387
Cohort 6	2015-04-01											3095	3362	3653	3951	429
Cohort 7	2015-07-01										2624	2873	3121	3390	3666	399
Cohort 8	2015-10-01									2079	2308	2527	2745	2983	3226	351
Cohort 9	2016-01-01								2718	2984	3314	3628	3941	4281	4630	503
Cohort 10	2016-04-01							3287	3733	4098	4550	4982	5412	5879	6358	692
Cohort 11	2016-07-01						1598	1824	2072	2275	2526	2765	3004	3263	3529	384
Cohort 12	2016-10-01					1345	1512	1725	1959	2151	2388	2615	2841	3086	3337	363
Cohort 13	2017-01-01				765	899	1011	1153	1310	1438	1597	1748	1899	2063	2231	242
Cohort 14	2017-04-01			83	99	116	131	149	169	186	207	226	246	267	289	31
Cohort 15	2017-07-01		1074	1374	1630	1916	2152	2457	2790	3063	3401	3723	4044	4394	4752	517
Cohort 16	2017-10-01	1207	1470	1880	2230	2621	2945	3361	3817	4190	4653	5094	5533	6011	6501	707

All weekly cohorts average 495

I checked quarterly retention, quarterly revenue by cohort weeks. I also calculated Cumulative sum of revenue by first time buyers in cohort and computed future CLV. By using this technique I did not get any valuable insights related to sales and profit.

Regression model to predict Profit

By using 4 independent and 1 dependent variable I wanted to populate a regression. In the dataset Ship_mode has 4 distinct values, so I decided to use 0 for not-same-day shipping and 1 for same-day shipping. The remaining variables were continuous.

Independent:
Quantity (continuous)
Discount (continuous)
Sales (continuous)
Ship mode (binary/dummy)

Dependent:
Profit (continuous)

Descriptive statistics of variables

Profit		**Quantity**		**Discount**		**Sales**		**Ship Mode**	
Mean	28.6331876	Mean	3.788646376	Mean	0.156196436	Mean	229.8031562	Mean	0.054365238
Standard Error	2.344684027	Standard Error	0.022263124	Standard Error	0.002066297	Standard Error	6.23797322	Standard Error	0.002268845
Median	8.64135	Median	3	Median	0.2	Median	54.376	Median	0
Mode	0	Mode	3	Mode	0	Mode	12.96	Mode	0
Standard Deviation	234.3276795	Standard Deviation	2.224976163	Standard Deviation	0.206505734	Standard Deviation	623.4229313	Standard Deviation	0.226748334
Sample Variance	54909.46137	Sample Variance	4.950518925	Sample Variance	0.042644618	Sample Variance	388656.1512	Sample Variance	0.051414807
Kurtosis	396.9704644	Kurtosis	1.996869701	Kurtosis	2.407333447	Kurtosis	305.1498116	Kurtosis	13.45893482
Skewness	7.559729632	Skewness	1.279927344	Skewness	1.684120715	Skewness	12.96970006	Skewness	3.931442463
Range	14999.954	Range	13	Range	0.8	Range	22638.036	Range	1
Minimum	-6599.978	Minimum	1	Minimum	0	Minimum	0.444	Minimum	0
Maximum	8399.976	Maximum	14	Maximum	0.8	Maximum	22638.48	Maximum	1
Sum	285988.2777	Sum	37841	Sum	1560.09	Sum	2295273.924	Sum	543
Count	9988	Count	9988	Count	9988	Count	9988	Count	9988
Largest(1)	8399.976	Largest(1)	14	Largest(1)	0.8	Largest(1)	22638.48	Largest(1)	1
Smallest(1)	-6599.978	Smallest(1)	1	Smallest(1)	0	Smallest(1)	0.444	Smallest(1)	0
Confidence Level(95%)	460%	Confidence Level(95%)	4%	Confidence Level(95%)	0.004050359	Confidence Level(95%)	12.22768442	Confidence Level(95%)	0.004447393

Correlation with Profit

Profit	Quantity	Discount	Sales	Ship mode
	6.6%	-21.9%	47.9%	0.1%

Correlation accross variables to check possible multicollinearity

	Profit	Quantity	Discount	Sales	Ship Mode
Profit	1				
Quantity	0.066	1			
Discount	-0.219	0.009	1		
Sales	0.479	0.201	-0.028	1	
Ship Mode	0.001	-0.019	-0.004	0.003	1

Multicollinearity is a statistical concept where several independent variables in a model are correlated. Two variables are considered to be perfectly collinear if their correlation coefficient is +/- 1.0.
Multicollinearity among independent variables will result in less reliable statistical inferences.

Here we don't see any perfect multicollinearity.

We get correlation between two variables but we haven't tested these for statistical significance, so at this point we cannot tell if this correlation is significantly different from zero.
We need to run another procedure to check that.

SIMPLE LINEAR REGRESSION

First we can make a model with the highest correlation between dependent and independent variable, i.e. sales and profit.
Because the *p* value is so low ($p < 0.0001$), we can reject the null hypothesis and conclude that sales has a statistically significant effect on profit.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.479054816
R Square	0.229493517
Adjusted R Square	0.229416358
Standard Error	205.6996177
Observations	9988

ANOVA

	df	SS	MS	F	Significance F
Regression	1	125849836.3	125849836.3	2974.306266	0
Residual	9986	422530954.4	42312.3327		
Total	9987	548380790.7			

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95%	Upper 95%
Intercept	-12.74600563	2.193625573	-5.81047458	0.00%	-17.0459538	-8.446057452	-17.0459538	-8.446057452
Sales	0.180063642	0.003301666	54.53720075	0.00%	0.17359171	0.186535573	0.17359171	0.186535573

MULTIPLE LINEAR REGRESSION

We can also make a model including all independent variables.

We need to figure out which of these relationships are statistically significant and reliable.
If p-value is > significance level (e.g. 0,05), then variable is not significant.

The result below shows some of the attributes with a P-value higher than the preferred alpha (5%), which suggests that the attributes have a low statistically significant relationship with the profit.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.522227799
R Square	0.272721874
Adjusted R Square	0.272430468
Standard Error	199.8760894
Observations	9988

ANOVA

	df	SS	MS	F	Significance F
Regression	4	149555437.2	37388859.29	935.8807785	0
Residual	9983	398825353.5	39950.45112		
Total	9987	548380790.7			

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95%	Upper 95%
Intercept	35.1120564	4.256005744	8.250002116	0.00%	26.76942717	43.45468562	26.76942717	43.45468562
Quantity	-2.973602322	0.917871772	-3.239670738	0.12%	-4.772816029	-1.174388616	-4.772816029	-1.174388616
Discount	-233.471761	9.690211289	-24.09356762	5.86%	-252.4665285	-214.4769934	-252.4665285	-214.4769934
Sales	0.180014859	0.00327644	54.94221545	0.00%	0.173592377	0.186437342	0.173592377	0.186437342
Ship Mode	-2.087124905	8.822526402	-0.236567714	81.30%	-19.38105518	15.20680537	-19.38105518	15.20680537

To solve that, we use elimination approach for removing those attributes (yellow rows) with the highest P-value one at a time followed by running the regression again until all attributes have P-values less than 0,05. Also, here we have a bigger R(squared) measure.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.522223896
R Square	0.272717797
Adjusted R Square	0.272499263
Standard Error	199.8666396
Observations	9988

ANOVA

	df	SS	MS	F	Significance F
Regression	3	149553201.4	4985106712.5%	1247.940382	0
Residual	9984	398827589.3	3994667.4%		
Total	9987	548380790.7			

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95%	Upper 95%
Intercept	34.98169776	4.219982714	829.0%	0.00%	26.70968104	43.25371448	26.70968104	43.25371448
Quantity	-2.969225218	0.917641879	-323.6%	0.12%	-4.767988264	-1.170462171	-4.767988264	-1.170462171
Sales	0.180009885	0.003276217	5494.4%	0.00%	0.173587839	0.186431932	0.173587839	0.186431932

Constructing multiple linear regression and estimating predictions

The classification goal was to predict how sales influence profit.
Using the results, the regression model (multiple regression equation) can be written as:

Predicted profit = 34.98 - 2.97 * (quantity) + 0.18 * (sales)

Predicted profit	=	Intercept	+	Quantity	+	Sales
34.982		34.98		-2.97		0.18

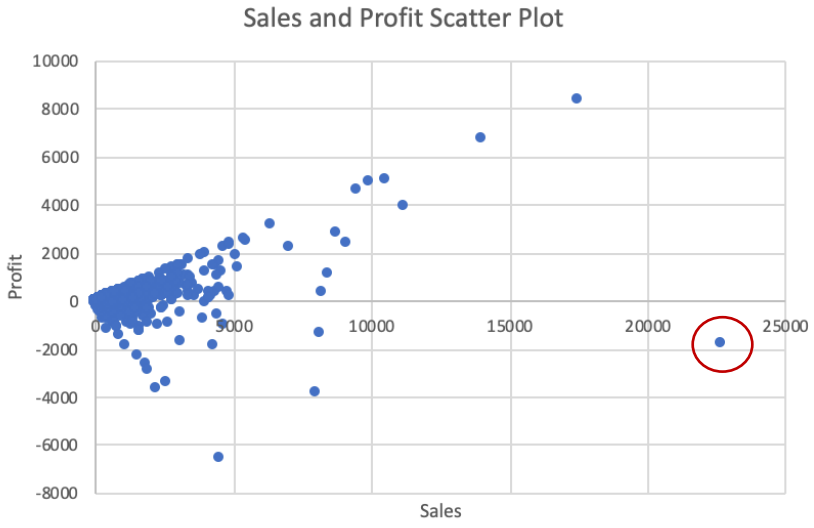
We got the attributes which p-value is less then alpha (5%) :

b1 *quantity* (continuous)
b2 *sales* (continuous)

Interpretation of coefficients

The intercept is β0 = 34.982. It has a positive sign, so having the higher outcome (profit) will be true assuming a value of 0 for all the predictors in the model.
A negative coefficient for quantity suggests that as the independent variable increases, the profit tends to decrease.
A positive coefficient for sales indicates that as the value of the independent variable increases, the mean of the dependent variable (profit) also tends to increase.

Plotted sales and profit in a scatter plot with notable upward tendency:



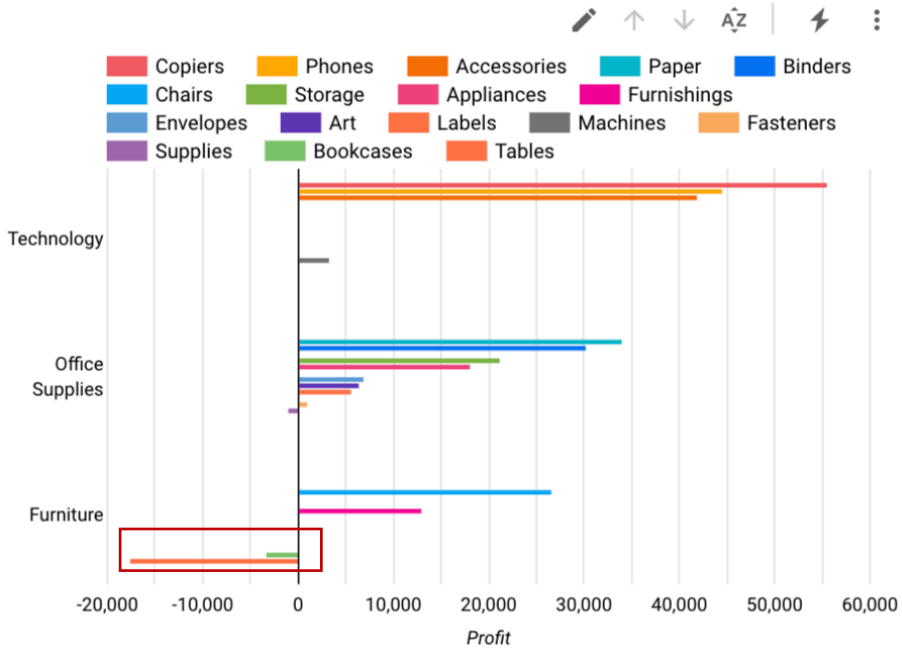
Note that with respect to customers, TOP1 customer (Sean Miller) according to the most sales revenue is in negative profit and company should pay attention to this.

Main insights

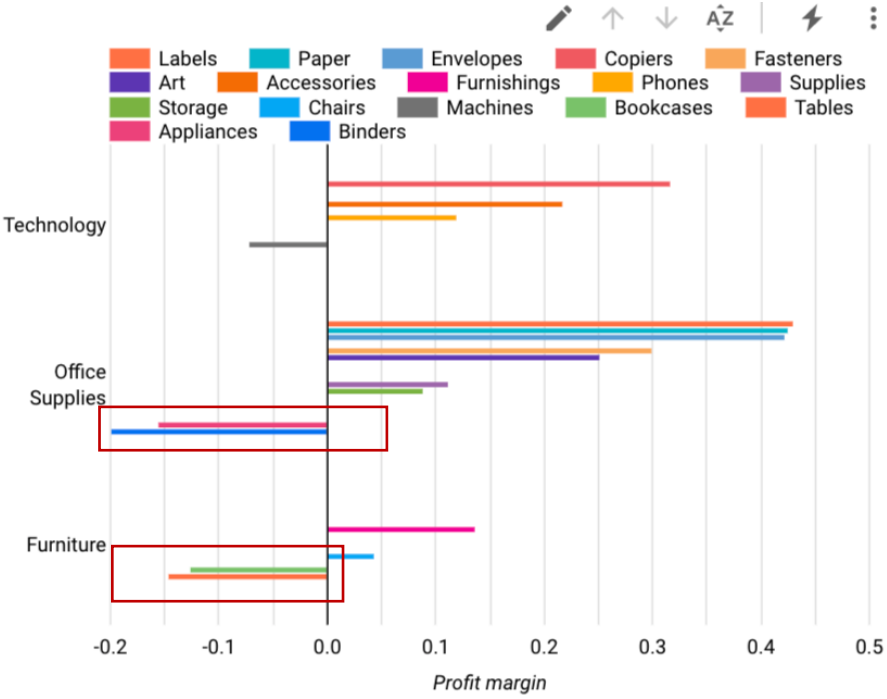
- All attributes selected after the elimination process show P-values lower than 5% and thereby suggesting the attributes that have been selected have significant role in profit prediction.
- R^2 of a model is 0.27, so only approximately 27% of the observed variation can be explained by the model's inputs. We should aim to include more independent variables and test them as well.
- Overall model could be improved with more data.

Analysis by using visualisations from LookerStudio tool

Categories and sub-categories by profit:



Categories and sub-categories by profit margin:



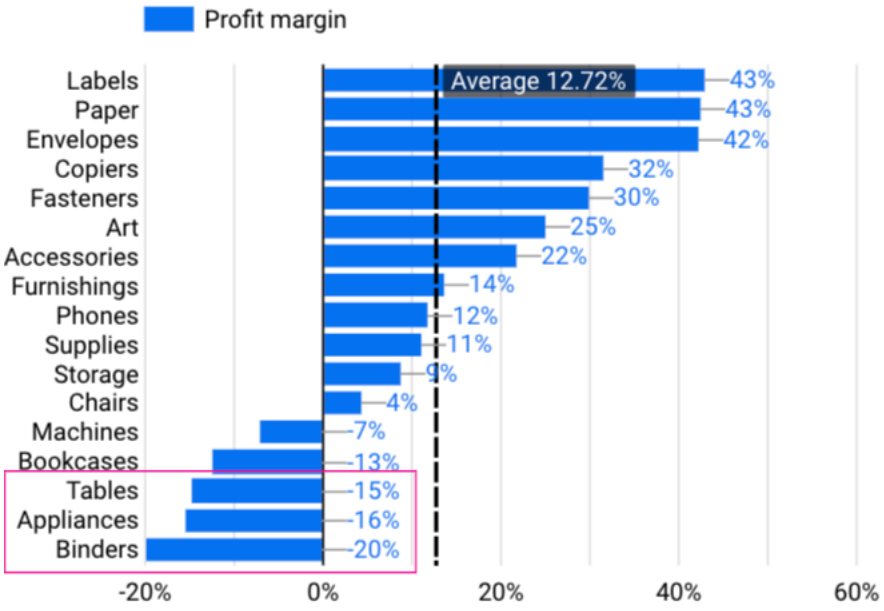
The bar chart above shows that the company has a very high margin (more than double the company's total margin) on a few of the sub-categories. Most of **Technology's** subcategories also have a high margin. It seems Technology has a better track record of being profitable vs. the other categories across these variables.

The chart also confirms that the profit margin on the sub-categories within **Furniture** are low. Half of the sub-categories from Furniture are even being sold at a loss.

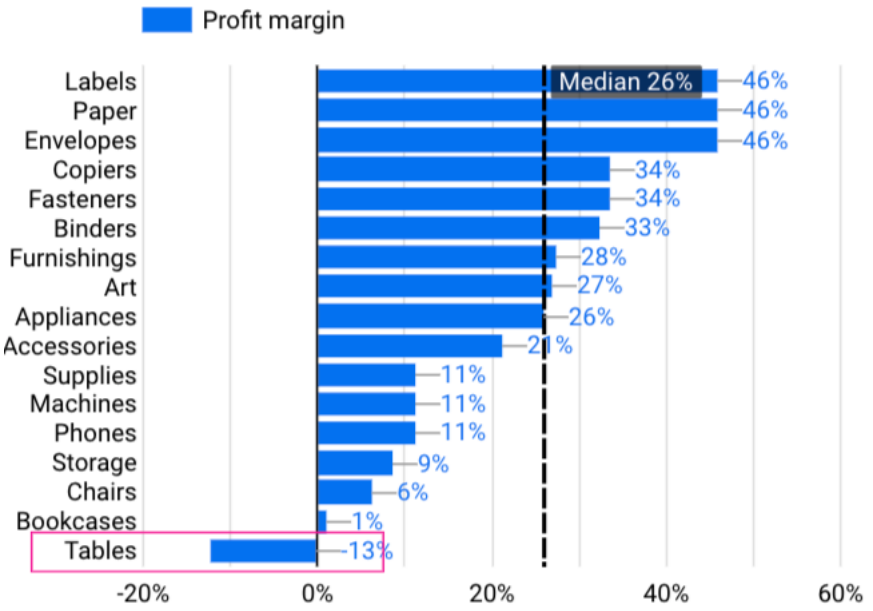
Also, despite **Office supplies** being the most popular category by sales quantity, it demonstrates negative average profit margin for some sub-categories like Appliances and Binders.

Given this, let's take a look at the items within the three sub-categories where the company is losing money.

Average profit margin in sub-categories:

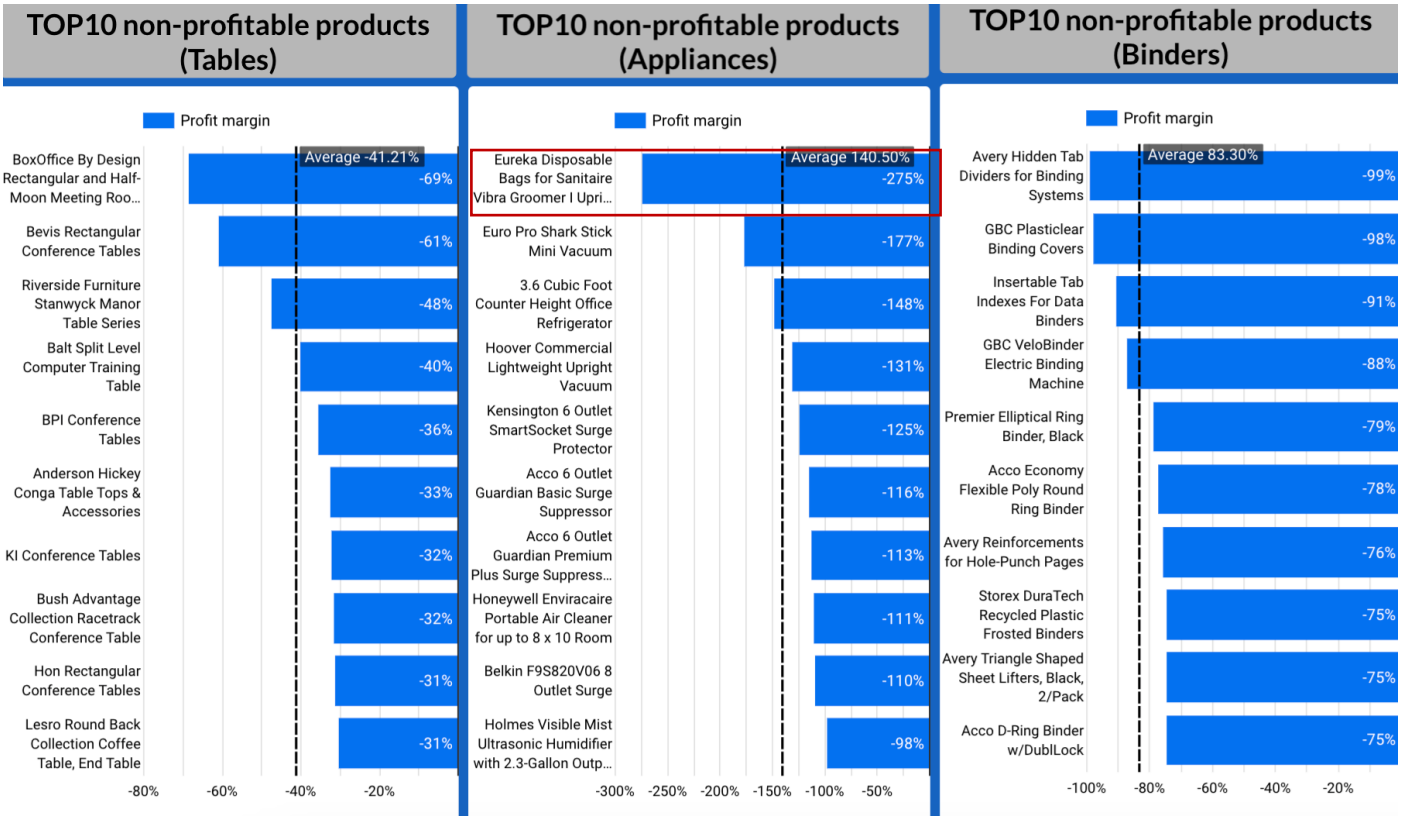


Median profit margin in sub-categories:



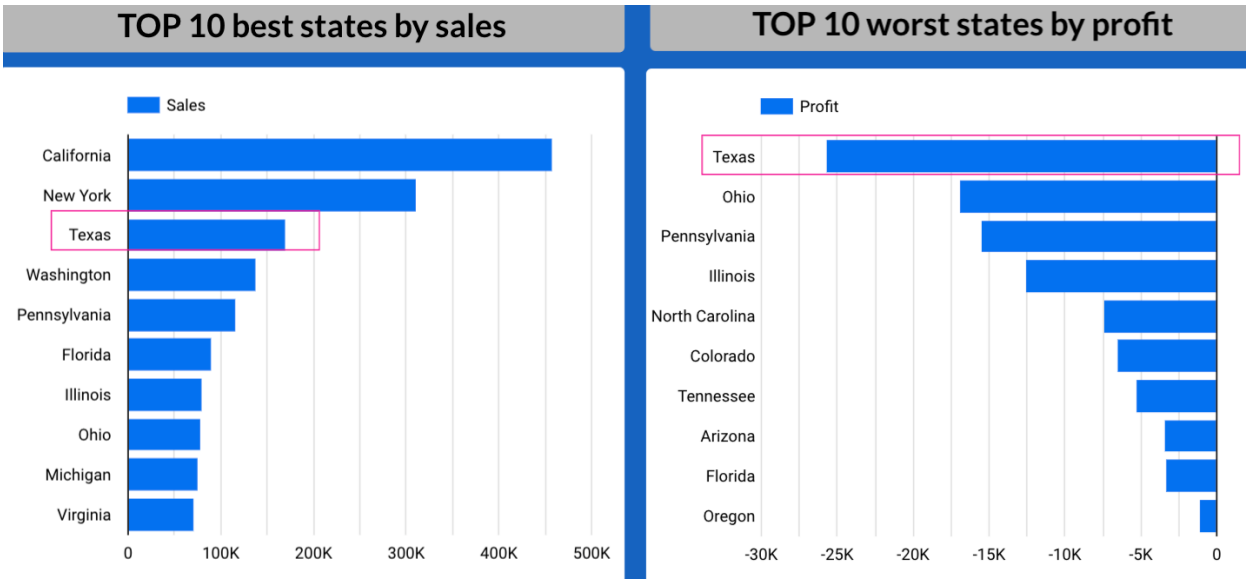
Using the company's average total profit margin (12.72%) as a benchmark I compare these non-profitable sub-categories. From the 3 sub-categories that this Superstore is selling at a loss, **Tables, Appliances and Binders** are clearly categories where this store is struggling.

Let's investigate even further by taking the top 10 worst items with the lowest margins within each respective sub-category.



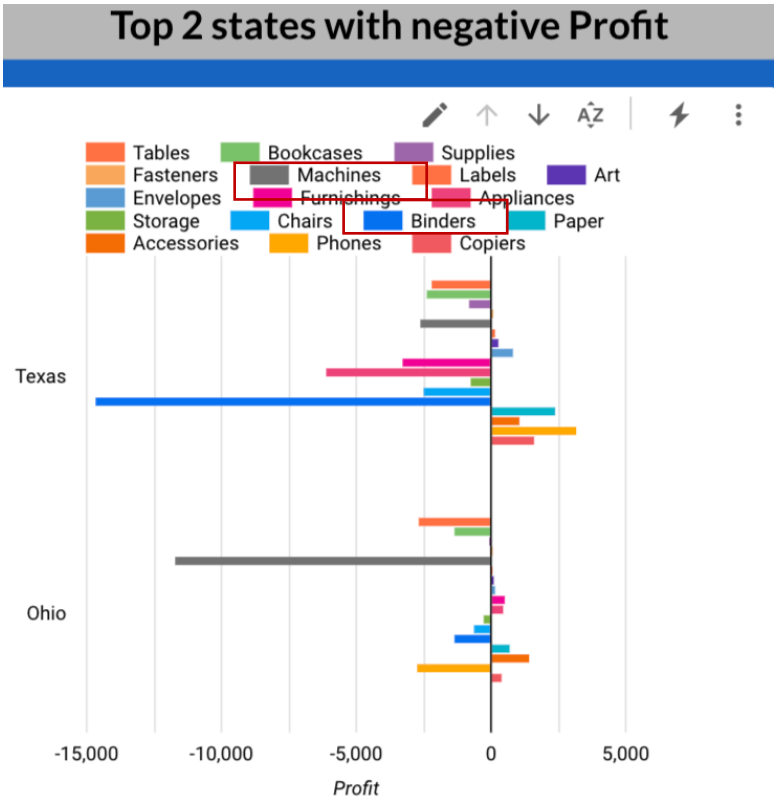
Looks like the biggest negative margins are in Appliances section, with **Eureka disposable bags** being the worst.

Let's explore this more on geographical level. There are top 2 states: California and New York where both sales and profit are the highest.
Unsurprisingly, most of the sales are coming from the heavily populated states like California, New York, and Texas. They are the most popular states among the customers and biggest markets in the US.



An important finding here is that some of the states where this company is making most of their sales is actually bringing down the profit that this company is making.

Texas for instance, which is their 3rd highest segment in terms of sales, is actually contributing to more than 1/4 of the loss that the company is making on its sales. The second worst state is Ohio. Let's explore sub-categories and their profits in these states.



Here you can notice that majority of sub-categories have negative profit. In Texas the top non-profitable sub-category is **Binders** and in Ohio it is **Machines.**
However, we can check what kind of discounts are given for these products.

Minimum discounts and lowest profit

	State	Order_ID	Discount	Sales	Profit
1.	Texas	487	0.2	170,188.05	-25,729.36
2.	Ohio	236	0.2	78,258.14	-16,971.38
3.	Pennsylvania	288	0.2	116,511.91	-15,559.96
4.	Illinois	276	0.2	80,166.1	-12,607.89
5.	North Carolina	136	0.2	55,603.16	-7,490.91
6.	Colorado	79	0.2	32,108.12	-6,527.86
7.	Tennessee	91	0.2	30,661.87	-5,341.69
8.	Arizona	108	0.2	35,282	-3,427.92
9.	Florida	200	0.2	89,473.71	-3,399.3
10.	Oregon	56	0.2	17,431.15	-1,190.47

There are some sub-categories with high number of orders with discount (for example binders, phones, chairs). In some states like Texas or Ohio **all orders are only with discounts** which highly contributes to negative profit margin. Here minimum discount given is 20%.

4. Results

PHASE 5 & 6: SHARE AND ACT

Key findings:
Based on the initial purpose with this analysis we helped the Superstore to identify:

- **Technology** and **Home Office** have on average a high profit margin, while **Furniture** are being sold at low margin or even at a loss for some its sub-categories.
- **Technology** is the only category that contains sub-categories with only positive profit margin values.
- The **Consume**r segment is driving most of the Superstore's sales and across all segments Technology is contributing the most to revenue.
- Over 50% of their profit is being made in **California and New York**. This makes this store quite geographically dependent on keeping up their sales here.
- Their sales in **Texas** (which is their 3rd highest state in sales) is on aggregate making a loss.
- **Tables, Appliances and Binders** are product categories being sold at a loss with lowest profit margin values out of all sub-categories.
- Some sub-categories, namely **machines and tables**, were the main drivers of loss in profits in worst profit states like **Texas and Ohio**, as almost all records of these products netted negative profits.
- **Discounts** really affect profit. Bigger discount gives minus profit to the store. Too many discounts are offered for some specific categories and in some states like Texas and Ohio.

Recommendations:

- Stop selling to certain states where they on aggregate are making big loss (e.g. Texas, Ohio).
- Stop selling non-profitable products or stop selling them in worst profit states (e.g. Eureka disposable bags).
- Consumer segment represented the most sales and they ordered the most quantity of item across all segments. Therefore, Superstore's owners should further target Consumer segment's customers when setting sales and marketing strategy.

Limitations:
I noticed that some customers had several states attributed to them.

Further analysis:

- Ideally a **customer basket analysis** would help to get deeper results. A customer basket analysis would check if low or non-profit items are part of customer baskets that on aggregate are making a profit.
- Also, further analysis is needed to identify items that are not driving profit but contributing to sales of other more profitable items.
- We could also include definition of regular/non-regular customer.

5. Appendices

5.1. Appendix 1

```
WITH
--Compute for F & M  values per customer
t1 AS (
    SELECT
    Customer_ID,
    MAX(order_date) AS last_purchase_date,
    COUNT(distinct(order_id)) AS frequency,
    ROUND(SUM(sales),0) AS monetary
    FROM (
select customer_id, order_date, order_id, sales, product_id
from `my-capstone-project-376621.Superstore_db.Superstore_main`
)
WHERE sales > 0
AND product_ID NOT LIKE "TEC-AC-10004659"   -- N/A values cleaned
    GROUP BY Customer_ID
    ORDER BY last_purchase_date
),
--Compute for R  values per customer
t2 AS (
    SELECT *,
    DATE_DIFF(TIMESTAMP(reference_date), TIMESTAMP(DATE_TRUNC(last_purchase_date,day)),day) AS
 recency
    FROM (
        SELECT *,
        "2017-12-31" AS reference_date          --last order_date is 2017-12-30
        FROM t1
    )
),
--Determine quintiles for each RFM metric
t3 AS (
SELECT
    a.*,
    --All percentiles for MONETARY
    b.percentiles[offset(25)] AS m25,
    b.percentiles[offset(50)] AS m50,
    b.percentiles[offset(75)] AS m75,
    b.percentiles[offset(100)] AS m100,
    --All percentiles for FREQUENCY
    c.percentiles[offset(25)] AS f25,
    c.percentiles[offset(50)] AS f50,
    c.percentiles[offset(75)] AS f75,
    c.percentiles[offset(100)] AS f100,
    --All percentiles for RECENCY
    d.percentiles[offset(25)] AS r25,
    d.percentiles[offset(50)] AS r50,
    d.percentiles[offset(75)] AS r75,
    d.percentiles[offset(100)] AS r100
FROM
    t2 a,
    (SELECT APPROX_QUANTILES(monetary, 100) percentiles FROM
    t2) b,
```

```
    (SELECT APPROX_QUANTILES(frequency, 100) percentiles FROM
    t2) c,
    (SELECT APPROX_QUANTILES(recency, 100) percentiles FROM
    t2) d
),
--Assign scores for each RFM metric
t4 AS (
    SELECT *,
    CAST(ROUND((f_score + m_score) / 2, 0) AS INT64) AS fm_score
    FROM (
        SELECT *,
        CASE WHEN monetary <= m25 THEN 1
            WHEN monetary <= m50 AND monetary > m25 THEN 2
            WHEN monetary <= m75 AND monetary > m50 THEN 3
            WHEN monetary <= m100 AND monetary > m75 THEN 4
        END AS m_score,
        CASE WHEN frequency <= f25 THEN 1
            WHEN frequency <= f50 AND frequency > f25 THEN 2
            WHEN frequency <= f75 AND frequency > f50 THEN 3
            WHEN frequency <= f100 AND frequency > f75 THEN 4
        END AS f_score,
        --Recency scoring is reversed
        CASE WHEN recency <= r25 THEN 4
            WHEN recency <= r50 AND recency > r25 THEN 3
            WHEN recency <= r75 AND recency > r50 THEN 2
            WHEN recency <= r100 AND recency > r75 THEN 1
        END AS r_score,
        FROM t3
        )
),
--Define RFM segments using the scores
t5 AS (
    SELECT
        Customer_ID,
        recency,
        frequency,
        monetary,
        r_score,
        f_score,
        m_score,
        fm_score,
        r_score+f_score+m_score AS rfm_score,
CONCAT(r_score,f_score,m_score) AS rfm_score_combined,
        CASE WHEN (r_score = 4 AND f_score = 4 AND m_score = 4)
        THEN 'Best customers'
        WHEN (r_score BETWEEN 3 AND 4 AND f_score = 4 AND m_score = 3)
        THEN 'Loyal Customers'
WHEN (r_score BETWEEN 3 AND 4 AND f_score BETWEEN 3 AND 4 AND m_score BETWEEN 1 AND 3)
        THEN 'Potential loyalists'
```

```
        WHEN (r_score BETWEEN 1 AND 4 AND f_score BETWEEN 1 AND 4 AND m_score = 4)

        THEN 'Big spenders'
        WHEN (r_score = 1 AND f_score = 1 AND m_score = 1)
 THEN 'Lost customers'
        WHEN (r_score BETWEEN 1 AND 2 AND f_score BETWEEN 1 AND 2 AND m_score BETWEEN 1 AND 2)

        THEN 'Almost lost'
        WHEN (r_score BETWEEN 3 AND 4 AND f_score between 1 AND 2 AND m_score BETWEEN 1 AND 3)

        THEN 'Promising'
        WHEN (r_score BETWEEN 1 AND 2 AND f_score BETWEEN 1 AND 4 AND m_score BETWEEN 1 AND 4)

        THEN 'At risk'



        END AS rfm_segment
    FROM t4
)
SELECT *
FROM t5
ORDER BY rfm_score_combined DESC
```

5.2. Appendix 2

```
WITH
  customers_sales AS (
  SELECT
    DISTINCT(customer_name) AS customer_name,
    SUM(sales) as sales
  FROM
    `my-capstone-project-
376621.Superstore_db.Superstore_main`
  GROUP BY 1
  ORDER BY
    sales DESC )
SELECT
  customer_name,
  sales,
  running_total,
  total,
  running_total / total AS percent_of_total
FROM (
  SELECT
    customer_name,
    sales,
    SUM(sales) OVER (ORDER BY sales DESC) AS running_total,
    SUM(sales) OVER() AS total
  FROM
    customers_sales)
```

```
ORDER BY
  sales DESC
```

5.3. Appendix 3

```
WITH
  customers_profit AS (
  SELECT
    DISTINCT(customer_name) AS customer_name,
    SUM(profit) as profit
  FROM
    `my-capstone-project-
376621.Superstore_db.Superstore_main`
  GROUP BY 1
  ORDER BY
    profit DESC )
SELECT
  customer_name,
  profit,
  running_total,
  total,
  running_total / total AS percent_of_total
FROM (
  SELECT
    customer_name,
    profit,
    SUM(profit) OVER (ORDER BY profit DESC) AS running_total,
    SUM(profit) OVER() AS total
  FROM
    customers_profit)
ORDER BY
  profit DESC
```

5.4. Appendix 4

1)

```
SELECT *,
ROUND(QUARTER0/QUARTER0,3) AS QUARTER0_retent_rate,
ROUND(QUARTER1/QUARTER0,3) AS QUARTER1_retent_rate,
ROUND(QUARTER2/QUARTER0,3) AS QUARTER2_retent_rate,
ROUND(QUARTER3/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER4/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER5/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER6/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER7/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER8/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER9/QUARTER0,3) AS QUARTER3_retent_rate,
```

```
ROUND(QUARTER10/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER11/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER12/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER13/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER14/QUARTER0,3) AS QUARTER3_retent_rate,
ROUND(QUARTER15/QUARTER0,3) AS QUARTER3_retent_rate


FROM

(

-- 1) select user's minimum event when he/she made a purchase
WITH t1 AS
(SELECT customer_id, MIN(order_date) AS most_recent_event_date
FROM `my-capstone-project-376621.Superstore_db.Superstore_main`

GROUP BY customer_id
)

-- 2) select cohorts for users
SELECT
DATE_TRUNC(DATE(most_recent_event_date), QUARTER) AS cohort_QUARTER,

-- 3) select weekly revenue for certain user cohort
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_TRUNC(DATE(most_re
cent_event_date), QUARTER) THEN sales END) AS QUARTER0,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 1 QUARTER) THEN sales END) AS Q
UARTER1,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 2 QUARTER) THEN sales END) AS Q
UARTER2,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 3 QUARTER) THEN sales END) AS Q
UARTER3,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 4 QUARTER) THEN sales END) AS Q
UARTER4,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 5 QUARTER) THEN sales END) AS Q
UARTER5,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 6 QUARTER) THEN sales END) AS Q
UARTER6,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 7 QUARTER) THEN sales END) AS Q
UARTER7,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 8 QUARTER) THEN sales END) AS Q
UARTER8,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 9 QUARTER) THEN sales END) AS Q
UARTER9,
```

```
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 10 QUARTER) THEN sales END) AS
QUARTER10,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 11 QUARTER) THEN sales END) AS
QUARTER11,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 12 QUARTER) THEN sales END) AS
QUARTER12,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 13 QUARTER) THEN sales END) AS
QUARTER13,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 14 QUARTER) THEN sales END) AS
QUARTER14,
SUM(CASE WHEN DATE_TRUNC(DATE(order_date), QUARTER) = DATE_ADD(DATE_TRUNC(DAT
E(most_recent_event_date), QUARTER), INTERVAL 15 QUARTER) THEN sales END) AS
QUARTER15

FROM `my-capstone-project-376621.Superstore_db.Superstore_main` AS events

INNER JOIN
t1 ON events.customer_id=t1.customer_id

GROUP BY cohort_QUARTER
ORDER BY cohort_QUARTER

)
```

2)

```
--first purchase of the unique customer made in this QUARTER
SELECT
DATE_TRUNC(DATE(min_event_date), QUARTER) AS cohort_QUARTER,
COUNT(customer_id) AS purchases

FROM (
SELECT
DISTINCT(customer_id),
MIN(order_date) AS min_event_date,
FROM
`my-capstone-project-
376621.Superstore_db.Superstore_main` AS events
GROUP BY 1
order by 2)

GROUP BY 1
ORDER BY cohort_QUARTER
```