CSI 4142 - Project Phase 2: Physical Design and Data Staging

uOttawa

Elli Traboulsi #300175764

Tamara Micic #300163312

Teodora Vukojevic #300199584

Group #4

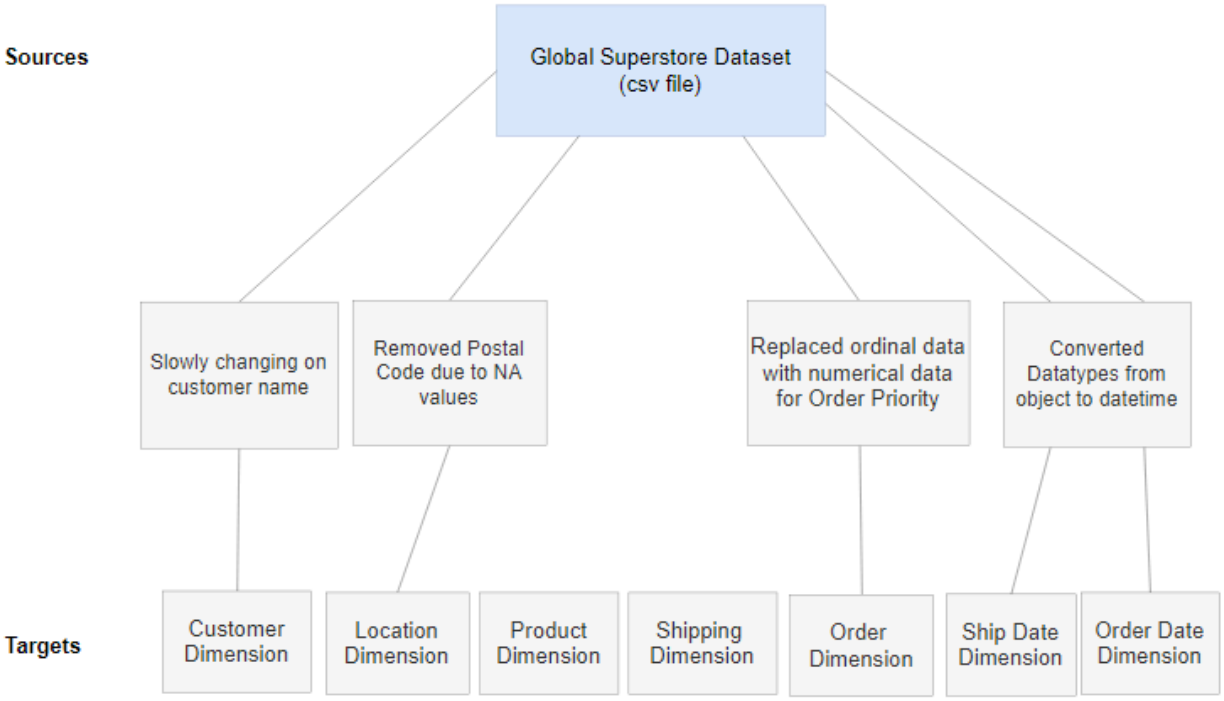
Prof. Yazan Otoum

TA Lansu Dai

March 27, 2024

Table of Contents

A. High-Level Data Staging Plan



B. Additional Details

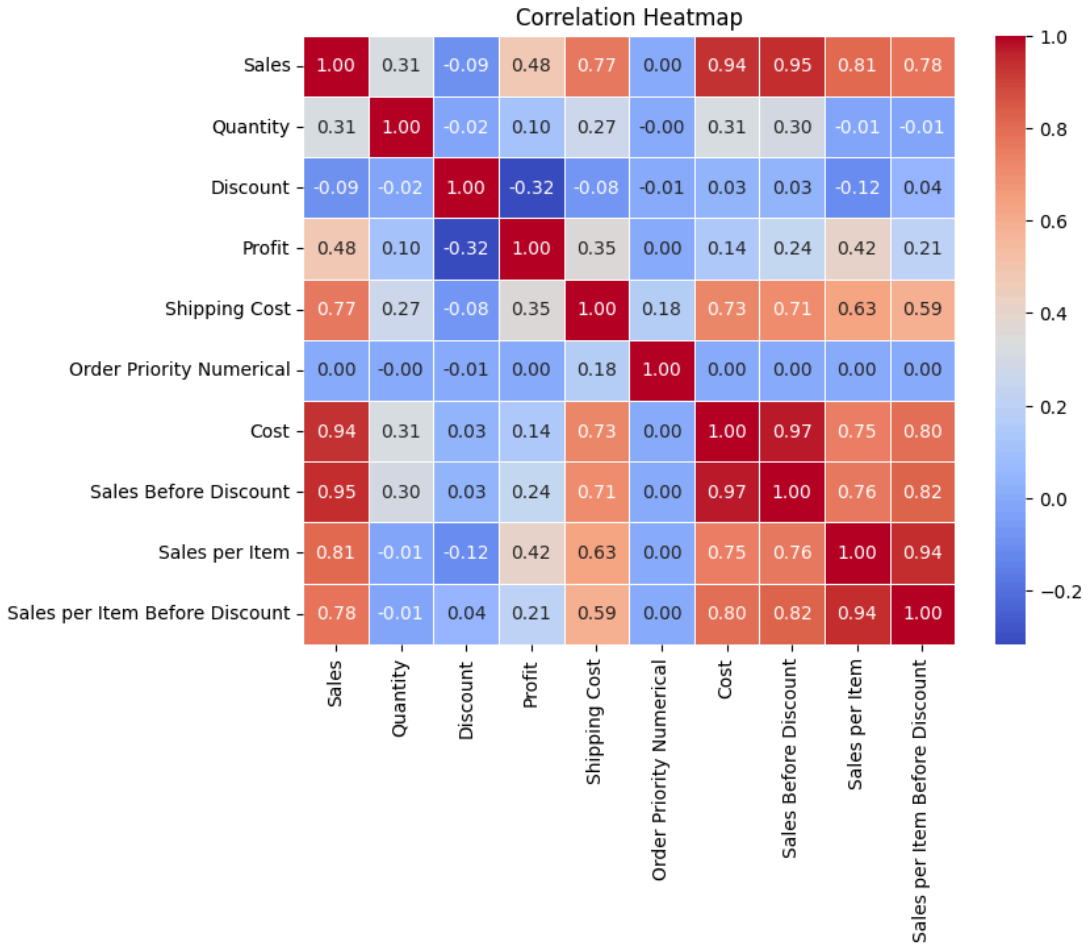
Google Colab and GitHub Links

All of our code is in this Google Colab file: https://colab.research.google.com/drive/1enToH8d5EMoeghI8Bwt5zm2PIPIF6Y2R?usp=sharing

All of our code and files are in this GitHub repository: ellitraboulsi/CSI4142-Project (github.com)

Correlation Heatmap

We created a heat map that shows the correlations between all the numerical attributes:



For the attributes that were present before feature engineering, we see in the heat map that Sales and Shipping Cost are highly correlated (higher shipping with higher sales), as well as Sales and Profit (higher profit with higher sales), and there is a notable correlation between Shipping Cost and Quantity as well (higher shipping cost with higher quantity). There is also a

negative correlation between Profit and Discount (lower profit with higher discount). None of the other variables have notable correlations.

After feature engineering (refer to Section C under Feature Engineering to see which features were created by us), we get some strong correlations between attributes (i.e. features). This makes sense since the new attributes are derived directly from the original ones. For example, there is a very high correlation of 0.95 between Sales and Sales Before Discount.

Loading Process (Surrogate Key Generation)

From the original dataset, we created 7 attribute dimensions and 1 fact table. For example, This is the Customer dimension:

	Customer Key	Customer ID	Customer Name
0	100000	RH-19495	Rick Hansen
1	100001	JR-16210	Justin Ritter
2	100002	CR-12730	Craig Reiter
3	100003	KM-16375	Katherine Murray
4	100004	RH-9495	Rick Hansen
...	...	...	...
1585	101585	SC-10800	Stuart Calhoun
1586	101586	BD-1500	Bradley Drucker
1587	101587	RC-9825	Roy Collins
1588	101588	MG-7890	Michael Granlund
1589	101589	ZC-11910	Zuschuss Carroll

1590 rows × 3 columns

This is the fact table:

	Customer Key	Location Key	Shipping Key	Product Key	Order Key	Order Date Key	Ship Date Key	Quantity Sold	Total Price
0	100000	200000	300000	400000	500000	600000	700000	7	2309.650
1	100001	200001	300001	400001	500001	600001	700001	9	3709.395
2	100002	200002	300002	400002	500002	600002	700002	9	5175.171
3	100003	200003	300003	400003	500003	600003	700003	5	2892.510
4	100004	200004	300004	400004	500004	600004	700004	8	2832.960
...	...	...	...	...	...	...	...	...	...
51285	100665	200861	317618	405301	551275	600856	700315	5	65.100
51286	100044	200130	317619	410681	551276	600270	701005	1	0.444
51287	100023	201860	317618	409458	551277	600061	700088	3	22.920
51288	100255	201350	317620	407196	551278	601077	701149	2	13.440
51289	100657	201394	317621	404624	551279	600106	700108	3	61.380

51290 rows × 9 columns

Aggregations

We did not need aggregations for our fact table however we thought it would be a good idea for future analysis. One of our goals is to maximize profits for this superstore so we created aggregates based on that. We summarized daily sales and profits into monthly and yearly and grouped them by region, country, (sale) category, sub-category and more. We also used the quantity feature to determine which items are the best sellers. Below is an example where we are aggregating sales data by Country and Order Year, summing total sales and profit.

```
[ ] country_order_year_aggregated = global_superstore.groupby(['Country', 'Order Year']).agg({'Sales': 'sum', 'Profit': 'sum'}).reset_index()
    country_order_year_aggregated
```

	Country	Order Year	Sales	Profit
0	Afghanistan	2011	1729.410	293.940
1	Afghanistan	2012	9071.820	1924.140
2	Afghanistan	2013	4242.810	1148.940
3	Afghanistan	2014	6629.280	2093.280
4	Albania	2011	1707.540	267.540

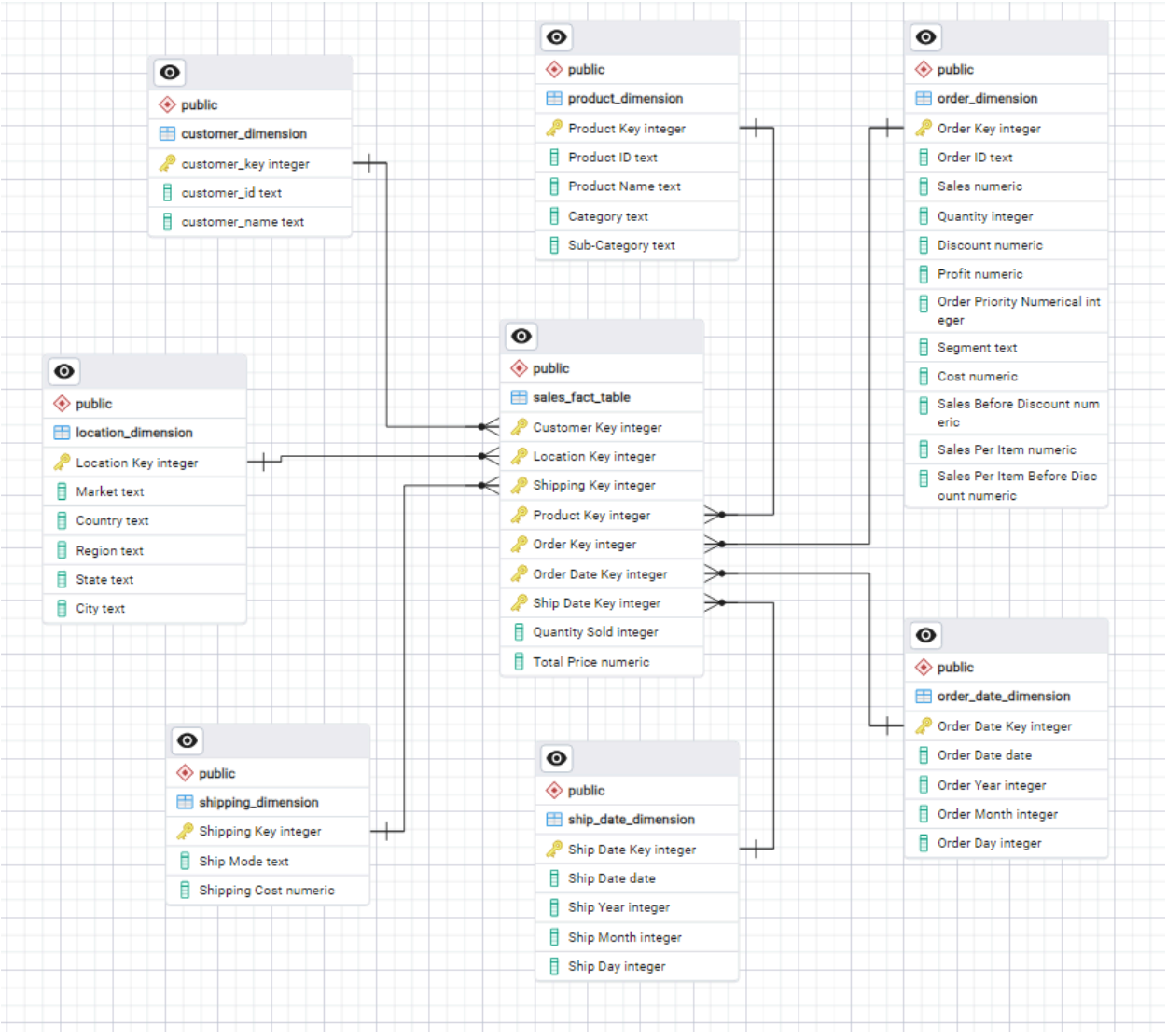
Here is another instance where we are aggregating the product sales by category and Order year, summing the quantity of items sold.

```
[ ] category_order_year_aggregated = global_superstore.groupby(['Category', 'Order Year'])['Quantity'].sum().reset_index()
    category_order_year_aggregated
```

	Category	Order Year	Quantity
0	Furniture	2011	6309
1	Furniture	2012	7279
2	Furniture	2013	9700
3	Furniture	2014	11666
4	Office Supplies	2011	18976
5	Office Supplies	2012	23135

Creating our Data Mart

We decided to use postgreSQL as our DBMS for our data mart. We created a new database, loaded all of our dimension csv files into tables and added them to an ERD using pgAdmin 4. Then we linked the tables by primary keys, foreign keys and primary foreign keys.



C. Data Quality Issues

Data Quality Issues and Handling

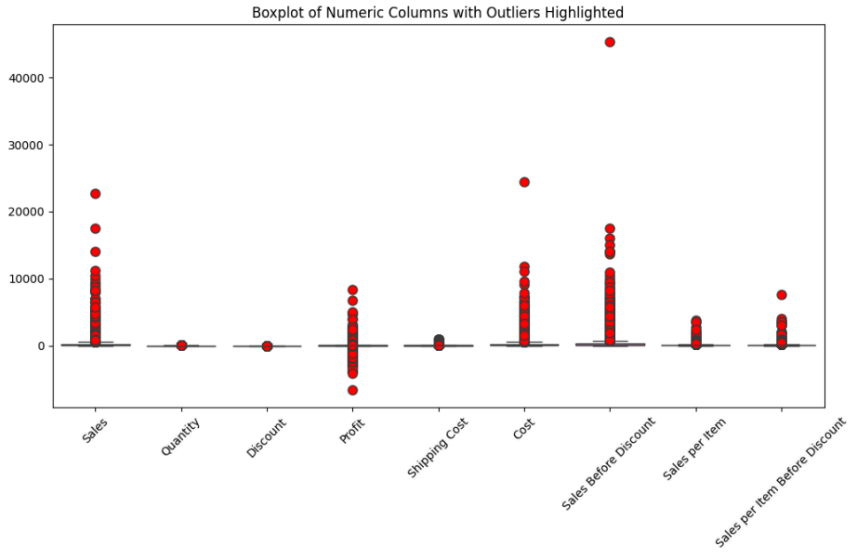
During our data staging, we followed the ETL process. We started by extracting the data from a CSV file using Python's Pandas library. We then proceeded with transforming the data (we walk through this step by step below) and finally generating surrogate keys and loading which in

Data Cleaning

Our data cleaning process included handling missing values, looking for typos, and outliers, removing duplicates and converting data types.

We started by looking for NA values. We noticed that only one column had NA values, Postal Code. About 80% of the values were NA, so we removed the column from our dataset. Although it might have been useful for analysis, there needed to be more data. We also drop Row ID, since it is not useful for our analysis.

Next, we checked for typos and outliers. We found no typos in our dataset. As for outliers, we did not find any either. We did detect some slight anomalies but decided not to remove them when looking at the rest of the data in the row. For example, the highest sale is $22,638.48 and the second highest is $17,499.56. After looking at the item that was sold and its quantity, we decided not to remove it because that purchase does not resemble one of an outlier. See the graph below for more insights.



We also checked to make sure our dataset did not have any duplicates. Luckily there were no duplicates to remove.

The last step of our data-cleaning process was to check the datatypes of each column were appropriate. We confirmed that monetary values were type float, 'Quantity' was already of type int and all categorical variables were of type "object" so we did not have to convert the datatypes of those features. We did however have to change 'Order Date' and 'Ship Date' from type object to type datetime.

Transforming the data into a format that can be used for analysis

We transformed the ‘Order Priority’ column values from ‘Critical’, ‘High’, ‘Medium’, and ‘Low’, to 4, 3, 2, and 1, respectively, so that we can use the numerical value to do analyses (e.g. look at the correlation between order priority and shipping cost). We create a new column, ‘Order Priority Numerical’, and leave the old one, in case we need it later.

Data Discretization

We debated converting some of our continuous values (e.g. Sales, Profit, Shipping Cost) into discrete data by grouping it into bins, but ultimately, we decided that it made more sense to leave these values as continuous since they are monetary and it makes sense to look at them individually.

Feature Engineering

As a part of feature engineering, we created four new attributes:

- ‘Cost’, which is the cost of the sale to the company. It is created by subtracting Profit from Sales.
- ‘Sales Before Discount’. The Sales attribute already accounts for the discount, so we find the value before the discount
- ‘Sales Per Item’, which is the Sales amount per each individual item in the sale
- ‘Sales Per Item Before Discount’, which is the Sales amount per each individual item in the sale, before the discount was applied
- ‘Order Year’. Ex. 2017 from 2012-07-31
- ‘Order Month’. Ex. 7 from 2012-07-31
- ‘Order Day’. Ex. 31 from 2012-07-31
- ‘Ship Year’. Ex. 2013 from 2013-02-07
- ‘Ship Month’. Ex. 2 from 2013-02-07
- ‘Ship Day’. Ex. 7 from 2013-02-07

We could use these attributes for future data analysis if we want to analyze the new attribute with respect to other factors.

	Sales	Quantity	Discount	Profit	Shipping Cost	Cost	Sales Before Discount	Sales per Item	Sales per Item Before Discount
count	51290.000000	51290.000000	51290.000000	51290.000000	51290.000000	51290.000000	51290.000000	51290.000000	51290.000000
mean	246.490581	3.476545	0.142908	28.610982	26.375915	217.879599	292.581209	71.657487	85.121957
std	487.565361	2.278766	0.212280	174.340972	57.296804	430.901539	600.615431	113.581515	136.272759
min	0.444000	1.000000	0.000000	-6599.978000	0.000000	0.554400	0.990000	0.336000	0.990000
25%	30.758625	2.000000	0.000000	0.000000	2.610000	26.880000	38.460000	11.799900	14.190000
50%	85.053000	3.000000	0.000000	9.240000	7.790000	73.635000	100.920000	29.400000	35.060000
75%	251.053200	5.000000	0.200000	36.810000	24.450000	222.965000	296.340000	82.160000	103.590000
max	22638.480000	14.000000	0.850000	8399.976000	933.570000	24449.558400	45276.960000	3773.080000	7546.160000

Here is a summary table that includes aggregates (count, mean, min, max, etc.) of the continuous attributes.

Data integration

We only have one data source, therefore, data integration was not necessary in our case.

D. Team Planning Sheet

CSI4142 - Project W23
Phase 2- Physical design and data staging
Teamwork - breakdown of duties

Deliverable checklist	Responsible team member(s)	Expected completion date	Actual completion date	Estimated time (hours) to complete	Actual time (hours) to complete	Notes (if any)
Create database instance	Elli	March 24th	March 24th	1 hour	1 hour	
Create Customer dimension	Tamara	March 23rd	March 24th	30 minutes	1 hour	
Create Location dimension	Tamara	March 24th	March 24th	30 minutes	1 hour	
Create Product dimension	Tamara	March 24th	March 24th	30 minutes	1 hour	
Create Shipping dimension	Tamara	March 24th	March 24th	30 minutes	30 minutes	
Create Order dimension	Tamara	March 23rd	March 24th	30 minutes	30 minutes	
Create Ship Date dimension	Tamara	March 23rd	March 24th	30 minutes	30 minutes	
Create Order Date dimension	Tamara	March 24th	March 24th	30 minutes	30 minutes	
Staging of dimension Customer	Elli, Tamara, Teodora	March 17th	March 19th	40 minutes	20 minutes	
Staging of dimension Location	Elli, Tamara, Teodora	March 17th	March 19th	40 minutes	30 minutes	
Staging of dimension Product	Elli, Tamara, Teodora	March 17th	March 19th	40 minutes	20 minutes	
Staging of dimension Shipping	Elli, Tamara, Teodora	March 17th	March 19th	40 minutes	20 minutes	
Staging of dimension Order	Elli, Tamara, Teodora	March 17th	March 19th	40 minutes	1 hour	
Staging of dimension Ship Date	Elli, Tamara, Teodora	March 17th	March 19th	40 minutes	20 minutes	
Staging of dimension Order Date	Elli, Tamara, Teodora	March 17th	March 19th	40 minutes	20 minutes	
Surrogate key pipeline	Tamara	March 19th	March 21st	2 hours	4 hours	
Staging of fact table – including FKs and measures	Tamara	March 22nd	March 23rd	30 minutes	45 minutes	
Data quality handling and reporting	Elli, Tamara, Teodora	March 17th	March 19th	3 hours	6 hours	
Others – if any						