

Wally-Mart's Improvement Strategies for Sales and Operations

By Anna H

Coursera BI Analyst Capstone Project



Table of Contents

- Introduction
- Methodology
- Results
- Discussion
- Conclusion
- Appendix





Introduction



Introduction

Wally-Mart is a leading global retail chain renowned for its diverse product offerings and commitment to customer satisfaction.

Wally-Mart wants to keep its competitive edge and optimize business performance.

We will examine the company's existing sales data and provide insights that will enhance Wally-Mart's sales strategies and its operational efficiency.





Methodology

Data Details

Historical Daily Sales Data

- 30,000-50,000 transactional data points
- January 1st, 2017, to February 16th, 2017
- 6 stores
- 5 cities
 - Edinburgh
 - Helsinki
 - London
 - Saint Petersburg
 - Vienna
- over 700 products in 10 product clusters
- the original data was kept in 6 comma delimited files (.csv)

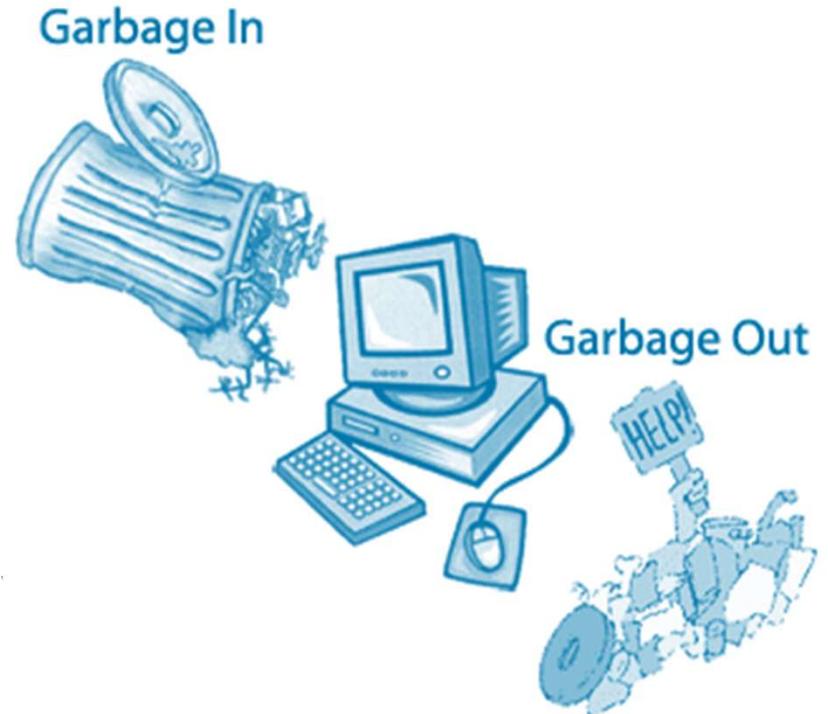


Data Cleaning Importance

Clean data leads to overall data quality improvements.

- better accuracy
- boosted performance
- improved models
- enhanced operational efficiency
- better, more reliable insights
- superior decision making
- satisfied customers
- higher profitability

Best done early to get the improvements early!





Module 1: Data Cleaning and Preparation Using Excel

- sales.csv — daily transaction data
 - 50,000 rows and 13 columns
 - 710 rows (1.42% of the data) were missing important information and were removed
 - data formats were standardized
 - string → text
 - number → number
 - date DD/MM/YYYY → YYYY-MM-DD

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
product_id	store_id	date	sales	revenue	stock	price	promo_type	promo_bin	promo_type	promo_bin	promo_discount	promo_discount_type	type_2	category
1	P0001	S0002	2/1/2017	0	0	8	6.25	PR14	PR03					
2	P0001	S0012	2/1/2017	1	5.3	0	6.25	PR14	PR03					
3	P0001	S0013	2/1/2017	2	10.59	0	6.25	PR14	PR03					
4	P0001	S0023	2/1/2017	0	0	6	6.25	PR14	PR03					
5	P0001	S0025	2/1/2017	0	0	1	6.25	PR14	PR03					
6	P0001	S0028	2/1/2017	0	0	7	6.25	PR14	PR03					
7	P0001	S0027	2/1/2017	0	0	19	6.25	PR14	PR03					
8	P0001	S0040	2/1/2017	0	0	8	6.25	PR14	PR03					
9	P0001	S0049	2/1/2017	0	0	6	6.25	PR14	PR03					
10	P0001	S0050	2/1/2017	0	0	5	6.25	PR14	PR03					
11	P0001	S0051	2/1/2017	0	0	6	6.25	PR14	PR03					
12	P0001	S0055	2/1/2017	0	0	6	6.25	PR14	PR03					
13	P0001	S0056	2/1/2017	1	5.30	6	6.25	PR14	PR03					
14	P0001	S0062	2/1/2017	0	0	2	6.25	PR14	PR03					
15	P0001	S0063	2/1/2017	0	0	7	6.25	PR14	PR03					
16	P0001	S0066	2/1/2017	0	0	2	15.9	PR14	PR03					
17	P0001	S0067	2/1/2017	0	0	7	15.9	PR14	PR03					
18	P0001	S0073	2/1/2017	0	0	4	15.9	PR14	PR03					
19	P0001	S0074	2/1/2017	0	0	6	15.9	PR14	PR03					
20	P0001	S0077	2/1/2017	0	0	4	15.9	PR14	PR03					
21	P0001	S0078	2/1/2017	0	0	6	15.9	PR14	PR03					
22	P0001	S0082	2/1/2017	0	0	4	15.9	PR14	PR03					
23	P0001	S0083	2/1/2017	0	0	10	15.9	PR14	PR03					
24	P0001	S0086	2/1/2017	0	0	10	6.25	PR14	PR03					
25	P0001	S0102	2/1/2017	0	0	7	6.25	PR14	PR03					
26	P0001	S0104	2/1/2017	0	0	22	6.25	PR14	PR03					
27	P0001	S0105	2/1/2017	0	0	6	6.25	PR14	PR03					
28	P0001	S0106	2/1/2017	1	5.30	3	6.25	PR14	PR03					
29	P0001	S0107	2/1/2017	0	0	10	6.25	PR14	PR03					
30	P0001	S0108	2/1/2017	0	0	10	6.25	PR14	PR03					
31	P0001	S0109	2/1/2017	0	0	10	6.25	PR14	PR03					
32	P0001	S0110	2/1/2017	0	0	10	6.25	PR14	PR03					
33	P0001	S0111	2/1/2017	0	0	10	6.25	PR14	PR03					
34	P0001	S0112	2/1/2017	0	0	10	6.25	PR14	PR03					
35	P0001	S0113	2/1/2017	0	0	10	6.25	PR14	PR03					
36	P0001	S0114	2/1/2017	0	0	10	6.25	PR14	PR03					
37	P0001	S0115	2/1/2017	0	0	10	6.25	PR14	PR03					
38	P0001	S0116	2/1/2017	0	0	10	6.25	PR14	PR03					
39	P0001	S0117	2/1/2017	0	0	10	6.25	PR14	PR03					
40	P0001	S0118	2/1/2017	0	0	10	6.25	PR14	PR03					
41	P0001	S0119	2/1/2017	0	0	10	6.25	PR14	PR03					
42	P0001	S0120	2/1/2017	0	0	10	6.25	PR14	PR03					
43	P0001	S0121	2/1/2017	0	0	10	6.25	PR14	PR03					
44	P0001	S0122	2/1/2017	0	0	10	6.25	PR14	PR03					
45	P0001	S0123	2/1/2017	0	0	10	6.25	PR14	PR03					
46	P0001	S0124	2/1/2017	0	0	10	6.25	PR14	PR03					
47	P0001	S0125	2/1/2017	0	0	10	6.25	PR14	PR03					
48	P0001	S0126	2/1/2017	0	0	10	6.25	PR14	PR03					
49	P0001	S0127	2/1/2017	0	0	10	6.25	PR14	PR03					
50	P0001	S0128	2/1/2017	0	0	10	6.25	PR14	PR03					
51	P0001	S0129	2/1/2017	0	0	10	6.25	PR14	PR03					
52	P0001	S0130	2/1/2017	0	0	10	6.25	PR14	PR03					
53	P0001	S0131	2/1/2017	0	0	10	6.25	PR14	PR03					
54	P0001	S0132	2/1/2017	0	0	10	6.25	PR14	PR03					
55	P0001	S0133	2/1/2017	0	0	10	6.25	PR14	PR03					
56	P0001	S0134	2/1/2017	0	0	10	6.25	PR14	PR03					
57	P0001	S0135	2/1/2017	0	0	10	6.25	PR14	PR03					
58	P0001	S0136	2/1/2017	0	0	10	6.25	PR14	PR03					
59	P0001	S0137	2/1/2017	0	0	10	6.25	PR14	PR03					
60	P0001	S0138	2/1/2017	0	0	10	6.25	PR14	PR03					
61	P0001	S0139	2/1/2017	0	0	10	6.25	PR14	PR03					
62	P0001	S0140	2/1/2017	0	0	10	6.25	PR14	PR03					
63	P0001	S0141	2/1/2017	0	0	10	6.25	PR14	PR03					
64	P0001	S0142	2/1/2017	0	0	10	6.25	PR14	PR03					
65	P0001	S0143	2/1/2017	0	0	10	6.25	PR14	PR03					
66	P0001	S0144	2/1/2017	0	0	10	6.25	PR14	PR03					
67	P0001	S0145	2/1/2017	0	0	10	6.25	PR14	PR03					
68	P0001	S0146	2/1/2017	0	0	10	6.25	PR14	PR03					
69	P0001	S0147	2/1/2017	0	0	10	6.25	PR14	PR03					
70	P0001	S0148	2/1/2017	0	0	10	6.25	PR14	PR03					
71	P0001	S0149	2/1/2017	0	0	10	6.25	PR14	PR03					
72	P0001	S0150	2/1/2017	0	0	10	6.25	PR14	PR03					
73	P0001	S0151	2/1/2017	0	0	10	6.25	PR14	PR03					
74	P0001	S0152	2/1/2017	0	0	10	6.25	PR14	PR03					
75	P0001	S0153	2/1/2017	0	0	10	6.25	PR14	PR03					
76	P0001	S0154	2/1/2017	0	0	10	6.25	PR14	PR03					
77	P0001	S0155	2/1/2017	0	0	10	6.25	PR14	PR03					
78	P0001	S0156	2/1/2017	0	0	10	6.25	PR14	PR03					
79	P0001	S0157	2/1/2017	0	0	10	6.25	PR14	PR03					
80	P0001	S0158	2/1/2017	0	0	10	6.25	PR14	PR03					
81	P0001	S0159	2/1/2017	0	0	10	6.25	PR14	PR03					
82	P0001	S0160	2/1/2017	0	0	10	6.25	PR14	PR03					
83	P0001	S0161	2/1/2017	0	0	10	6.25	PR14	PR03					
84	P0001	S0162	2/1/2017	0	0	10	6.25	PR14	PR03					
85	P0001	S0163	2/1/2017	0	0	10	6.25	PR14	PR03					
86	P0001	S0164	2/1/2017	0	0	10	6.25	PR14	PR03					
87	P0001	S0165	2/1/2017	0	0	10	6.25	PR14	PR03					
88	P0001	S0166	2/1/2017	0	0	10	6.25	PR14	PR03					
89	P0001	S0167	2/1/2017	0	0	10	6.25	PR14	PR03					
90	P0001	S0168	2/1/2017	0	0	10	6.25	PR14	PR03					
91	P0001	S0169	2/1/2017	0	0	10	6.25	PR14	PR03					
92	P0001	S0170	2/1/2017	0	0	10	6.25	PR14	PR03					
93	P0001	S0171	2/1/2017	0	0	10	6.25	PR14	PR03					
94	P0001	S0172	2/1/2017	0	0	10	6.25	PR14	PR03					
95	P0001	S0173	2/1/2017	0	0	10	6.25	PR14	PR03					
96	P0001	S0174	2/1/2017	0	0	10	6.25	PR14	PR03					
97	P0001	S0175	2/1/2017	0	0	10	6.25	PR14	PR03					
98	P0001	S0176	2/1/2017	0	0	10	6.25	PR14	PR03					
99	P0001	S0177	2/1/2017	0	0	10	6.25	PR14	PR03					
100	P0001	S0178	2/1/2017	0	0	10</								



Module 1: Data Cleaning and Preparation Using Excel

- product_hierarchy.csv — product detail data
 - 700 rows and 10 columns
 - removed missing data
 - > 2% of missing data related to product sizes
 - > 7% of missing data related to product clusters (categories)
 - data formats were standardized
 - string → text
 - number → number

	A	B	C	D	E	F	G	H	I	J	K
1	product_id	product_k	product_d	product_v	cluster_id	hierarchy1	hierarchy2	hierarchy3	hierarchy4	hierarchy5	hierarchy6
2	P0000		5	20	12	H004	H0004	H0004010	H000401040	H00040104054	
3	P0001		13.5	22	20	cluster_5	H01	H0105	H010501	H0105010	H0105010006
4	P0002		22	40	22	cluster_0	H03	H0315	H031508	H03150800	H0315080028
5	P0004		2	13	4	cluster_3	H03	H0314	H031405	H03140500	H0314050003
6	P0005		16	30	16	cluster_9	H03	H0312	H031211	H0312110	H0312110917
7	P0006		8.5	15	15	cluster_0	H03	H0316	H031608	H0316081	H0316081708
8	P0007		2	22	9.5	cluster_4	H03	H0313	H031305	H0313051	H0313051904
9	P0008		5	16	5	cluster_0	H00	H0000	H0000040	H00000400	H0000040017
10	P0009		5	18	14	cluster_6	H00	H0002	H000201	H0002010	H0002010012
11	P0010		2	22	3	cluster_0	H01	H0108	H010801	H0108010	H0108010917
12	P0011		9	22	15	cluster_0	H03	H0315	H031506	H0315061	H0315061202
13	P0012		10	20	12.3	cluster_0	H01	H0107	H010702	H0107021	H0107021001
14	P0013		7.1	27.5	13.7		H01	H0105	H0105030	H0105030	H0105030906
15	P0014		32.5	6	33	cluster_0	H03	H0313	H031302	H0313020	H0313020501
16	P0015		10	33	10	cluster_1	H00	H0000	H000003	H0000030	H0000030901
17	P0016		2	18	8	cluster_0	H00	H0004	H0004010	H0004010	H0004010512
18	P0017		8	28	8	cluster_6	H00	H0000	H000003	H0000031	H0000031001
19	P0018		1	14	11	cluster_4	H00	H0003	H000316	H0003160	H0003160922
20	P0019		1.5	42	32	cluster_0	H03	H0312	H031201	H0312010	H0312010004
21	P0020		5	28	16	cluster_0	H03	H0311	H031109	H0311090	H03110901
22	P0021		8	26	23	cluster_0	H03	H0313	H031302	H0313021	H0313021002
23	P0022		5.7	18.6	10.1	cluster_0	H01	H0108	H010800	H0108000	H0108000912



A	B	C	D	E	F	G	H	I	J
product_id	product_length	product_depth	product_width	cluster_id	hierarchy1_id	hierarchy2_id	hierarchy3_id	hierarchy4_id	hierarchy5_id
P0001	13.50	22.00	20.00	cluster_5	H01	H0105	H010501	H01050100	H0105010006
P0002	22.00	40.00	22.00	cluster_0	H03	H0315	H031508	H03150800	H0315080028
P0004	2.00	13.00	4.00	cluster_3	H03	H0314	H031405	H03140500	H0314050003
P0005	16.00	30.00	16.00	cluster_9	H03	H0312	H031211	H03121109	H0312110917
P0006	8.50	15.00	15.00	cluster_0	H03	H0316	H031608	H03160817	H0316081708
P0007	2.00	22.00	9.50	cluster_4	H03	H0313	H031305	H03130519	H0313051904
P0008	5.00	16.00	5.00	cluster_0	H00	H0000	H0000004	H000000400	H00000040017
P0009	5.00	18.00	14.00	cluster_6	H00	H0002	H000201	H00020100	H0002010012
P0010	2.00	22.00	3.00	cluster_0	H01	H0108	H010801	H01080109	H0108010917
P0011	9.00	22.00	15.00	cluster_0	H03	H0315	H031506	H03150612	H0315061202
P0012	10.00	20.00	12.30	cluster_0	H01	H0107	H010702	H01070210	H0107021001
P0014	32.50	6.00	33.00	cluster_0	H03	H0313	H031302	H03130205	H0313020501
P0015	10.00	33.00	10.00	cluster_1	H00	H0000	H000003	H00000309	H0000030901
P0016	2.00	18.00	8.00	cluster_0	H00	H0004	H000401	H00040105	H0004010512
P0017	8.00	28.00	8.00	cluster_6	H00	H0000	H00000310	H0000031001	H0000031001
P0018	1.00	14.00	11.00	cluster_4	H00	H0003	H000316	H00031609	H0003160922
P0019	1.50	42.00	32.00	cluster_0	H03	H0312	H031201	H03120100	H0312010004
P0020	5.00	28.00	16.00	cluster_0	H03	H0311	H031109	H03110901	H0311090101
P0021	8.00	26.00	23.00	cluster_0	H03	H0313	H031302	H03130210	H0313021002
P0022	5.70	18.60	10.10	cluster_0	H01	H0108	H010809	H01080900	H0108090012
P0024	1.00	22.00	16.00	cluster_0	H02	H0214	H021404	H02140407	H0214040701



Module 1: Data Cleaning and Preparation Using Excel

- store_cities.csv — store detail data
 - 145 rows and 4 columns
 - no missing data – no removals needed
 - data formats were standardized
 - string → text
 - number → number



The diagram illustrates the process of data cleaning and preparation. On the left, a raw CSV file named "store_cities" is shown with four columns: A (store_id), B (storetype_store_size), C (city_id), and D (city_name). The data contains various store IDs, store types, sizes, and city codes. On the right, the same data is shown in a cleaned and prepared format. The columns are renamed and reorganized: A (store_id), B (storetype_id), C (store_size), and D (city_id). The data is now presented in a more structured and readable format, with numerical values correctly formatted as numbers. The original file name "store_cities" is also present at the bottom right of the cleaned table.

	A	B	C	D
1	store_id	storetype_store_size	city_id	
2	S0091	ST04	19	C013
3	S0012	ST04	28	C005
4	S0045	ST04	17	C008
5	S0032	ST03	14	C019
6	S0027	ST04	24	C022
7	S0088	ST04	20	C009
8	S0095	ST02	44	C014
9	S0055	ST04	24	C014
10	S0099	ST03	14	C014
11	S0078	ST04	19	C036
12	S0006	ST03	8	C024
13	S0135	ST03	16	C035
14	S0066	ST04	47	C033
15	S0106	ST04	21	C031
16	S0112	ST04	46	C031
17	S0059	ST03	15	C014
18	S0129	ST04	18	C021
19	S0116	ST03	15	C031
20	S0098	ST03	15	C022
21	S0049	ST04	25	C031
22	S0144	ST04	21	C022
23	S0097	ST04	45	C014
24	S0130	ST03	17	C037
25	S0041	ST03	8	C013
26	S0142	ST04	31	C006
27	S0123	ST04	20	C021
28	S0007	ST03	16	C014
29	S0107	ST04	19	C014
30	S0036	ST04	21	C001
31	S0044	ST03	6	C022
32	S0052	ST04	17	C014
33	S0101	ST04	30	C026

	A	B	C	D
1	store_id	storetype_id	store_size	city_id
2	S0091	ST04	19.00	C013
3	S0012	ST04	28.00	C005
4	S0045	ST04	17.00	C008
5	S0032	ST03	14.00	C019
6	S0027	ST04	24.00	C022
7	S0088	ST04	20.00	C009
8	S0095	ST02	44.00	C014
9	S0055	ST04	24.00	C014
10	S0099	ST03	14.00	C014
11	S0078	ST04	19.00	C036
12	S0006	ST03	8.00	C024
13	S0135	ST03	16.00	C035
14	S0066	ST04	47.00	C033
15	S0106	ST04	21.00	C031
16	S0112	ST04	46.00	C031
17	S0059	ST03	15.00	C014
18	S0129	ST04	18.00	C021
19	S0116	ST03	15.00	C031
20	S0098	ST03	15.00	C022
21	S0049	ST04	25.00	C031
22	S0144	ST04	21.00	C022
23	S0097	ST04	45.00	C014
24	S0130	ST03	17.00	C037
25	S0041	ST03	8.00	C013
26	S0142	ST04	31.00	C006
27	S0123	ST04	20.00	C021

Module 1: Data Cleaning and Preparation Using Excel

- product_names.csv — full product names

- 701 rows and 2 columns
- trimmed leading spaces
- data formats were standardized
 - string → text

A	B	C	D	E
1	product_id	product_name		
2	P0000	Bottom Freezer Refrigerator		
3	P0001	Side-by-Side Refrigerator		
4	P0002	Mini Fridge		
5	P0004	Beverage Center		
6	P0005	Wine Cooler		
7	P0006	Chest Freezer		
8	P0007	Upright Freezer		
9	P0008	Compact Freezer		
10	P0009	Gas Range		
11	P0010	Electric Range		
12	P0011	Induction Cooktop		
13	P0012	Wall Oven		
14	P0013	Countertop Oven		
15	P0014	Microwave Oven		
16	P0015	Convection Oven		
17	P0016	Steam Oven		
18	P0017	Pizza Oven		
19	P0018	Toaster Oven		
20	P0019	Built-In Dishwasher		
21	P0020	Portable Dishwasher		
22	P0021	Top-Load Washing Machine		
23	P0022	Front-Load Washing Machine		
24	P0023	High-Efficiency Washing Machine		
25	P0024	Stackable Washer and Dryer		
26	P0025	Ventless Dryer		
27	P0026	Steam Dryer		
28	P0010	Built-In Dishwasher		

→

A	B	C
1	product_id	product_name
2	P0000	Bottom Freezer Refrigerator
3	P0001	Side-by-Side Refrigerator
4	P0002	Mini Fridge
5	P0004	Beverage Center
6	P0005	Wine Cooler
7	P0006	Chest Freezer
8	P0007	Upright Freezer
9	P0008	Compact Freezer
10	P0009	Gas Range
11	P0010	Electric Range
12	P0011	Induction Cooktop
13	P0012	Wall Oven
14	P0013	Countertop Oven
15	P0014	Microwave Oven
16	P0015	Convection Oven
17	P0016	Steam Oven
18	P0017	Pizza Oven
19	P0018	Toaster Oven
20	P0019	Built-In Dishwasher
21	P0020	Portable Dishwasher
22	P0021	Top-Load Washing Machine
23	P0022	Front-Load Washing Machine
24	P0023	High-Efficiency Washing Machine
25	P0024	Stackable Washer and Dryer
26	P0025	Ventless Dryer
27	P0026	Steam Dryer

< > ... sales | product_hierarchy | store_cities | **product_names** | store_names | city



Module 1: Data Cleaning and Preparation Using Excel

- store_names.csv — full store names
 - 146 rows and 2 columns
 - trimmed leading spaces
 - store names standardized
 - spacing
 - capitalization
 - consistent naming
 - data formats were standardized
 - string → text

A B

	store_id	store_name
1	S0005	MediaMarkt(National Chain)
2	S0036	Saturn
3	S0104	Euronics
4	S0068	FNAC
5	S0086	Darty
6	S0038	Currys
7	S0012	El Corte Inglz
8	S0092	Worten
9	S0109	Conrad Electronic
10	S0136	Boulanger
11	S0141	Expert
12	S0030	MediaWorld
13	S0094	M.Video
14	S0142	Dustin
15	S0002	DIGI
16	S0023	Dixons
17	S0045	Comet
18	S0122	Electro World
19	S0088	Cosmos Sport
20	S0083	MediaMarkt(National Chain)
21	S0132	Expert MegaDom
22	S0131	Unieuro
23	S0041	Cecchini
24	S0091	Fnac
25	S0003	Darty
26	S0007	Mediamarkt Saturn Technik
27	S0010	Coolblue
28	S0014	Mister Appliance
29	S0015	El Gigante
30	S0016	Jet
31	S0020	Block
32	S0021	Richer Sounds
33	S0022	Expert (National Chain)
34	S0023	Dixons
35	S0024	Casa Tech
36	S0025	Appliance Direct
37	S0026	Mediagalaxy
38	S0027	Centro dell'Elettronica
39	S0028	Elettrodomestici Toscani
40	S0029	Smart Home Firenze
41	S0030	MediaWorld (National Chain)
42	S0031	Technik Frankfurt
43	S0032	SmartLiving.dk
44	S0033	Milar Electrodomésticos
45	S0034	Technik
46	S0035	Elektrohaus Müller
47	S0036	Saturn (National Chain)
48	S0037	SmartLiving.dk
49	S0038	Currys (National Chain)
50	S0039	Lefevre
51	S0040	Elektronikhuset
52	S0041	Cecchini
53	S0042	Euronics Dublin
54	S0043	Euronics Madrid (National Chain)



< > ⋮ sales | product_hierarchy | store

Module 1: Data Cleaning and Preparation Using Excel

- city_names.csv — city names
 - 146 rows and 2 columns
 - removed 107 blank rows
 - trimmed leading spaces
 - data formats were standardized
 - string → text

A	B	C
1	city_id	city_name
2	C001	Amsterdam
3	C002	Berlin
4	C003	Barcelona
5	C004	Budapest
6	C005	Copenhagen
7	C006	Dublin
8	C007	Edinburgh
9	C008	Florence
10	C009	Frankfurt
11	C010	Geneva
12	C011	Istanbul
13	C012	Krakow
14	C013	Lisbon
15	C014	London
16	C015	Madrid
17	C016	Milan
18	C017	Munich
19	C018	Oslo
20	C019	Paris
21	C020	Prague
22	C021	Rome
23	C022	Saint Petersburg
24	C023	Stockholm
25	C024	Vienna
26	C025	Venice
27	C026	Warsaw
28	C027	Zurich
29	C028	Athens
30	C029	Belgrade
31	C030	Brussels

A		B
1	city_id	city_name
2	C001	Amsterdam
3	C002	Berlin
4	C003	Barcelona
5	C004	Budapest
6	C005	Copenhagen
7	C006	Dublin
8	C007	Edinburgh
9	C008	Florence
10	C009	Frankfurt
11	C010	Geneva
12	C011	Istanbul
13	C012	Krakow
14	C013	Lisbon
15	C014	London
16	C015	Madrid
17	C016	Milan
18	C017	Munich
19	C018	Oslo
20	C019	Paris
21	C020	Prague
22	C021	Rome
23	C022	Saint Petersburg
24	C023	Stockholm
25	C024	Vienna
26	C025	Venice
27	C026	Warsaw





Module 1: Pivot Table Analysis of Data Using Excel

- Clean data was summarized and analyzed by using pivot tables
 - to aggregate data for use, XLOOKUP was utilized
 - primary and foreign key were identification for each type of data
 - calculation of
 - Total Sales
 - Available Stock
 - Generated Revenue
- the following analyses were performed with the help of pivot tables
 - [Sales Data Analysis](#)
 - [Sales by City Analysis](#)
 - [Product Performance Analysis](#)



more details are available in the [Results](#) section

Data Type	Primary key	Foreign key(s)
sales		product_id store_id
product hierarchy	product_id	cluster_id hierarchy_id
store cities	store_id	storetype_id city_id
product names	product_id	
store names	store_id	
city names	city_id	



Module 2: Database Setup Help for Future Analysis

- Databases provide a structured and organized way to store, manage and retrieve data.
- This makes analysis easier and faster and can automate certain processes reducing the need to redo calculations.
- Using SQL with databases allows us
 - to quickly calculate aggregates
 - to perform multidimensional analysis
 - to summarize and analyze data along hierarchies
 - to identify trends by combining different kinds of data (ex. time and geographical data)
- Better analysis leads to deeper and more useful actionable insights that will be used to optimize sales strategies and stock management



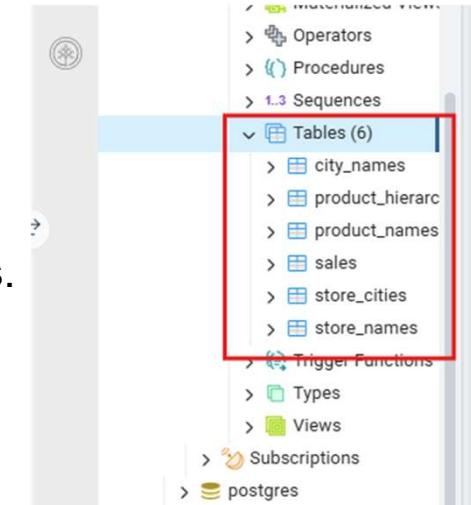
PostgreSQL



Module 2: Data Querying and Analysis Using PostgreSQL

- We created a PostgreSQL database – BICapstone and imported our data.

```
theia@theiadocker-malyshka: ~ 
postgres=# CREATE DATABASE BICapstone;
CREATE DATABASE
postgres=# \connect BICapstone
connection to server at "172.21.225.0", port 5432 failed: FATAL:  database "BICapstone" does not exist
Previous connection kept
postgres=# \connect bicapstone
psql (14.17 (Ubuntu 14.17-0ubuntu0.22.04.1), server 13.2)
You are now connected to database "bicapstone" as user "postgres".
bicapstone=# \i /home/theia/BI-dump.sql
SET
SET
SET
SET
SET
SET
set_config
-----
```



- After the data was loaded, we checked the names of the imported tables.



PostgreSQL



Module 2: Data Querying and Analysis Using PostgreSQL

- We also ran queries to confirm that the data was imported correctly.

The screenshot shows the pgAdmin 4 interface. In the left sidebar, under the 'Tables' section, the 'sales' table is selected. In the main area, the SQL tab contains the following query:

```
1 SELECT * FROM sales LIMIT 5;
```

The Data Output tab displays the results of the query:

product_id	store_id	date	sales	revenue	stock	price	promo_type_1	promo_bin_1	promo_type_2	prom
P0001	S0002	2017-02-01	0	0	8	6.25	PR14	[null]	PR03	[null]
2	S0012	2017-02-01	1	5.3	0	6.25	PR14	[null]	PR03	[null]
3	S0013	2017-02-01	2	10.59	0	6.25	PR14	[null]	PR03	[null]
4	S0023	2017-02-01	0	0	6	6.25	PR14	[null]	PR03	[null]
5	S0025	2017-02-01	0	0	1	6.25	PR14	[null]	PR03	[null]

Total rows: 5 of 5 Query complete 00:00:00.262 Ln 1, Col 20



- Various analyses were performed on the data that can be seen in the [Results](#) section.



Module 3: Data Visualization and Statistical Analysis

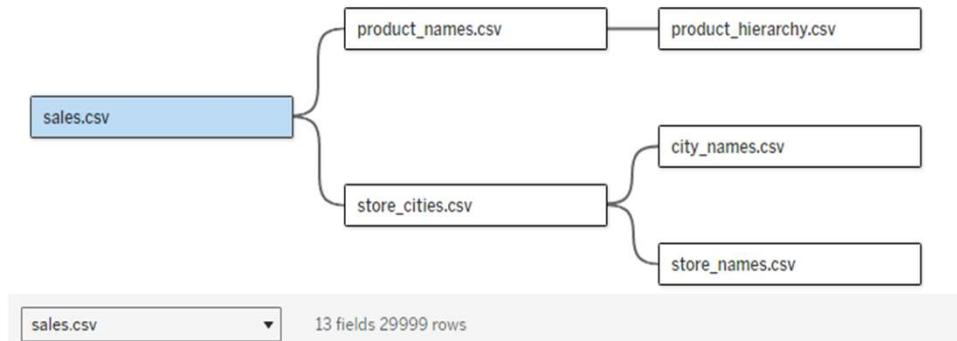
We then loaded the data back into Excel and combined it into a single spreadsheet to make it easier to visualize the data and to use the Analysis ToolPak to do the statistical analysis.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	
1	product_id	product_name	store_id	store_name	store_city	date	sales	revenue	stock	price	promo	promo	promo	promo_bin_2	promo
2	P0005	Wine Cooler	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	7	\$ 33.90	PR14		PR03		
3	P0011	Induction Cooktop	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	10	\$ 49.90	PR14		PR03		
4	P0015	Convection Oven	S0001	Electro World (National Chain)	Helsinki	2017-01-02	1	2.41	20	\$ 2.60	PR14		PR03		
5	P0017	Pizza Oven	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	13	\$ 1.49	PR14		PR03		
6	P0018	Toaster Oven	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	49	\$ 1.95	PR14		PR03		
7	P0024	Stackable Washer and Dryer	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	23	\$ 1.95	PR14		PR03		
8	P0035	Immersion Blender	S0001	Electro World (National Chain)	Helsinki	2017-01-02	2	4.54	14	\$ 2.45	PR14		PR03		
9	P0046	Smart Doorbells (with video)	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	4	\$ 34.50	PR14		PR03		
10	P0051	Smart Speakers (with voice assistants)	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	27	\$ 0.70	PR14		PR03		
11	P0054	Robot Vacuums	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	6	\$ 3.95	PR14		PR03		
12	P0055	Smart TVs	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	14	\$ 3.50	PR05	verylow	PR03		
13	P0057	Upright Vacuums	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	6	\$ 12.90	PR14		PR03		
14	P0060	Steam Mops	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	1	\$ 15.90	PR14		PR03		
15	P0062	Hardwood Floor Cleaners	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	1	\$ 19.90	PR14		PR03		
16	P0066	Electric Grills	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	43	\$ 2.50	PR14		PR03		
17	P0067	Smokers	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	4	\$ 15.90	PR14		PR03		
18	P0068	Patio Heaters	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	6	\$ 5.10	PR10	verylow	PR03		
19	P0070	Lawn Mowers (gas, electric)	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	1	\$ 6.25	PR14		PR03		
20	P0071	String Trimmers	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	1	\$ 9.45	PR14		PR03		
21	P0072	Leaf Blowers	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	13	\$ 3.90	PR14		PR03		
22	P0079	Phone Chargers	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	46	\$ 2.25	PR14		PR03		
23	P0080	Portable Speakers	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	10	\$ 7.90	PR06	veryhigh	PR03		
24	P0083	Security Cameras	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	2	\$ 6.90	PR05	moderate	PR03		
25	P0089	Portable Blenders	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	2	\$ 5.99	PR14		PR03		
26	P0090	Handheld Vacuum Cleaner	S0001	Electro World (National Chain)	Helsinki	2017-01-02	0	0.00	6	\$ 2.90	PR14		PR03		

Module 4: Data Visualization and Dashboards Using Tableau

- Lastly, we moved the data to Tableau to create easy to understand visualizations.
- We had 6 different files, so we set up a relationship model to define their connections.

Sales Relationship Model



- Addressing cardinality issues was an important step because our data was normalized.
 - The relationship between **store_cities.csv** and **city_names.csv** and the relationship between **store_cities.csv** and **store_names.csv** required a Many to One connection





Module 4: Data Visualization and Dashboards Using Tableau

To create effective visualizations, we had to create several calculated fields.

- First, we had to convert the date in the sales.csv file to its proper format – YYYY-MM-DD
- Then, we created 2 more fields to aid in our analysis:
 - Sales Growth Percentage
 - Average Stock Levels

The screenshot shows the Tableau interface with two calculated field dialog boxes open and a data source pane on the right.

Sales Growth Percentage:
Calculation: $(\text{SUM}([\text{Sales}]) - \text{LOOKUP}(\text{SUM}([\text{Sales}]), -1)) / \text{LOOKUP}(\text{SUM}([\text{Sales}]), -1)$
Message: The calculation is valid.
Buttons: Default Table Calculation, Apply, OK.

Average Stock Levels:
Calculation: $\text{AVG}([\text{Stock}])$
Message: The calculation is valid.
Buttons: Apply, OK.

Data Source Pane:
store_names.csv (Count)
Measure Names
= # Sales Growth Percentage (highlighted)
Average Stock Levels
Latitude (generated)
Longitude (generated)
Data Source, Sheet 1, etc.

- We also created a hierarchy for the cities, stores and products, to facilitate filtering of data.
- Once this was done, we were able to create our visualizations.



Results



Module 1, Lesson 1: Data Cleaning and Preparation

As mentioned in the [Methodology section](#), data cleaning improves the quality of the data and the results of the analysis. After completing the cleaning (as [described earlier](#)), we had a set of data that could be used for further analysis.

A	B	C	D	E	F	G	H	I	J
product_id	store_id	date	sales	revenue	stock	price	promo_type_1	promo_bin_1	promo_type_2
P0001	S0002	2017-02-01	0	0.00	8	6.25	PR14		PR03
P0001	S0012	2017-02-01	1	5.30	0	6.25	PR14		PR03
P0001	S0013	2017-02-01	2	10.59	0	6.25	PR14		PR03
P0001	S0023	2017-02-01	0	0.00	6	6.25	PR14		PR03
P0001	S0025	2017-02-01	0	0.00	1	6.25	PR14		
P0001	S0027	2017-02-01	0	0.00	7	6.25	PR14		
P0001	S0040	2017-02-01	0	0.00	19	6.25	PR14		
P0001	S0049	2017-02-01	0	0.00	8	6.25	PR14		
P0001	S0050	2017-02-01	0	0.00	5	6.25	PR14		
P0001	S0051	2017-02-01	0	0.00	6	6.25	PR14		
P0001	S0055	2017-02-01	0	0.00	6	6.25	PR14		
P0001	S0056	2017-02-01	1	5.30	6	6.25	PR14		
P0001	S0062	2017-02-01	0	0.00	2	6.25	PR14		
P0001	S0063	2017-02-01	0	0.00	7	6.25	PR17		
P0001	S0066	2017-02-01	0	0.00	1	6.25	PR14		
P0001	S0078	2017-02-01	0	0.00	6	6.25	PR14		
P0001	S0082	2017-02-01	0	0.00	4	6.25	PR14		
P0001	S0083	2017-02-01	0	0.00	10	6.25	PR14		
P0001	S0087	2017-02-01	0	0.00	10	6.25	PR14		
P0001	S0094	2017-02-01	0	0.00	6	6.25	PR14		
P0001	S0096	2017-02-01	0	0.00	2	6.25	PR14		
P0001	S0102	2017-02-01	0	0.00	7	6.25	PR14		
P0001	S0103	2017-02-01	1	5.30	10	6.25	PR14		
P0001	S0104	2017-02-01	0	0.00	22	6.25	PR14		
P0001	S0105	2017-02-01	0	0.00	6	6.25	PR14		
P0001	S0106	2017-02-01	1	5.30	3	6.25	PR14		

Ready Accessibility: Investigate

A	B	A	B	A	B	C	D	A	B	C	D	E	F	G	H	I	J
city_id	city_name	store_id	store_name	store_id	storetype_id	store_size	city_id	product_id	product_name	product_id	product_name	product_id	product_name	product_id	product_name	product_id	product_name
1	C001	Amsterdam		1	store_id			1	P0000	Bottom Freezer Refrigerator							
2	C002	Berlin		2	S0001	Electro World (National Chain)		2	P0001	Side-by-Side Refrigerator							
3	C003	Barcelona		3	S0002	DIGI		3	P0002	Mini Fridge							
4	C004	Budapest		4	S0003	Darty (National Chain)		4	P0004	Beverage Center							
5	C005	Copenhagen		5	S0004	Elettrodomestici Rossi		5	P0005	Wine Cooler							
6	C006			6	S0005	MediaMarkt (National Chain)		6	P0006	Chest Freezer							
7	C007			7	S0006	Euronics Lisboa (National Chain)		7	P0007	Upright Freezer							
8	C008			8	S0007	Mediamarkt Saturn Technik		8	P0008	Compact Freezer							
9	C009			9	S0008	Currys (National Chain)		9	P0009	Gas Range							
10	C010			10	S0009	Appliance Centre		10	P0010	Electric Range							
11	C011			11	S0010	Coolblue		11	P0011	Induction Cooktop							
12	C012			12	S0011	Datart (National Chain)		12	P0012	Wall Oven							
13	C013			13	S0012	El Corte Inglés (National Chain)		13	P0013	Countertop Oven							
14	C014			14	S0013	Unileuro (National Chain)		14	P0014	Microwave Oven							
15	C015			15	S0014	Mister Appliance		15	P0015	Convection Oven							
16	C016			16	S0015	El Gigante		16	P0016	Steam Oven							
17	C017			17	S0016	Jet		17	P0017	Pizza Oven							
18	C018			18	S0017	Komplette		18	P0018	Toaster Oven							
19	C019			19	S0018	Komplett		19	P0019	Built-In Dishwasher							
20	C020			20	S0019	Leefree		20	P0020	Portable Dishwasher							
21	C021			21	S0020	Mediagalaxy		21	P0021	Top-Load Washing Machine							
22	C022			22	S0021	Centro dell' Elettronica		22	P0022	Front-Load Washing Machine							
23	C023			23	S0022	Richer Sounds		23	P0023	High-Efficiency Washing Machine							
24	C024			24	S0023	Expert (National Chain)		24	P0024	Stackable Washer and Dryer							
25	C025			25	S0024	Dixons		25	P0025	Ventless Dryer							
26	C026			26	S0025	Dixons		26	P0026	Steam Dryer							
27	C027			27	S0026	Appliance Direct		27	P0027	Front-Load Washing Machine							
28	C028			28	S0027	MediaWorld		28	P0028	High-Efficiency Washing Machine							
29	C029			29	S0028	Smart Home Firenze		29	P0029	Stackable Washer and Dryer							
30	C030			30	S0029	Technik Frankfurt		30	P0030	Ventless Dryer							
31	C031			31	S0030	SmartLiving.dk		31	P0031	Steam Dryer							
32	C032			32	S0031	SmartLiving.dk		32	P0032	Front-Load Washing Machine							
33	C033			33	S0032	Milar Electrodomésticos		33	P0033	High-Efficiency Washing Machine							
34	C034			34	S0033	Technik		34	P0034	Stackable Washer and Dryer							
35	C035			35	S0034	Elektrohaus Müller		35	P0035	Ventless Dryer							
36	C036			36	S0035	Saturn (National Chain)		36	P0036	Front-Load Washing Machine							
37	C037			37	S0036	SmartLiving.dk		37	P0037	High-Efficiency Washing Machine							
38	C038			38	S0037	Lefevre		38	P0038	Stackable Washer and Dryer							
39	C039			39	S0038	Currys (National Chain)		39	P0039	Ventless Dryer							
40	C040			40	S0039	Euromarket Dublin		40	P0040	Front-Load Washing Machine							
41	C041			41	S0040	Elektronikhuset		41	P0041	High-Efficiency Washing Machine							
42	C042			42	S0041	Cocchini		42	P0042	Stackable Washer and Dryer							
43	C043			43	S0042	Euronics Dublin		43	P0043	Ventless Dryer							
44	C044			44	S0043	Euronics Madrid (National Chain)		44	P0044	Front-Load Washing Machine							

Ready Accessibility: Investigate



Module 1, Lesson 2: Sales Data Analysis

- After determining the primary and foreign keys for each set of data, we used Pivot tables in Excel to analyze the data.
- First, we performed an overall Sales Data Analysis, which is used to help companies understand its past performance through various sales metrics such as:
 - Total Sales – 13,148
 - Available Stock – 466,304
 - Generated Revenue – \$47,367.13
- The calculated metrics set a baseline and help evaluate the success of any changes implemented in the future.
- Breaking the data down by months or quarters, to observe temporal trends in sales is useful.
- Unfortunately, there was not enough data to do so, and that line of inquiry was not pursued.

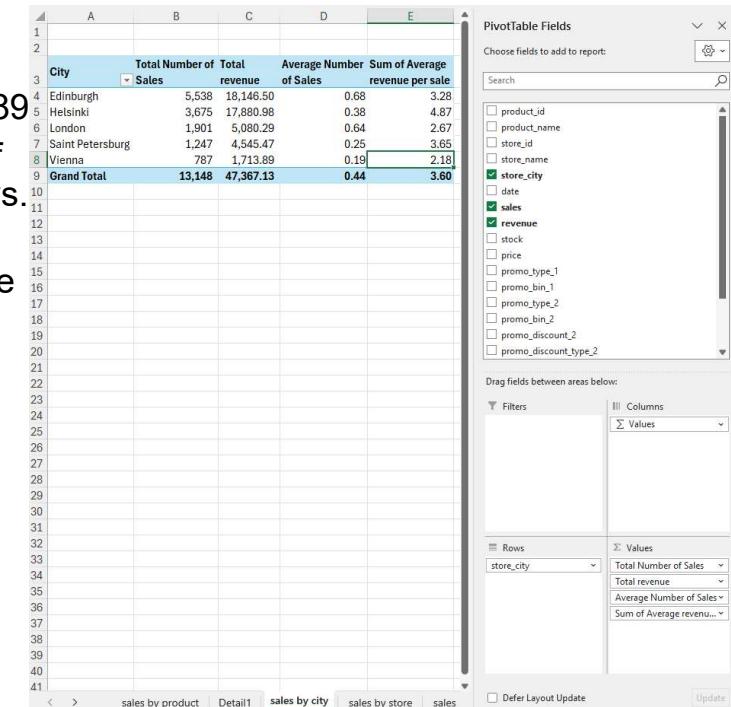
Screenshot of an Excel PivotTable analysis showing sales data by store. The PivotTable Fields pane on the right shows fields like store_name, sales, revenue, stock, etc., with store_name selected as a Row field. The PivotTable itself displays data for various stores, including Currys, Darty, DIGI, Electro World, Elettrodomestici Rossi, and Euronics Lisboa, along with a Grand Total row.

Store	Total		Average	
	Sales	Number of Sales	Total stock	Total revenue
Currys (National Chain)	389	31,473	1,055.21	0.15
Darty (National Chain)	1,901	45,375	5,080.29	0.64
DIGI	5,538	133,536	18,146.50	0.68
Electro World (National Chain)	3,675	153,521	17,880.98	0.38
Elettrodomestici Rossi	1,247	71,470	4,545.47	0.25
Euronics Lisboa (National Chain)	398	30,929	658.68	0.25
Grand Total	13,148	466,304	47,367.13	0.44



Module 1, Lesson 2: Sales by City Analysis

- Next, the sales data was analyzed by location to identify the cities that contribute the most to the sales volume.
- It was determined that sales occurred in 5 cities
 - Edinburgh had the highest number of sales – 5,538 and revenue – \$18,146.50
 - Vienna had the lowest number of sales – 787 and revenue – \$1,713.89
 - Helsinki was an interesting exception with a lower average number of sales – 0.38, but a much higher revenue than London – \$17,880.98 vs. \$5,080.29
 - This led us to create another field that calculated the average revenue per sale. Helsinki had the highest result – \$4.87 per sale, whereas Edinburgh only had an average sale revenue of \$3.28
- This analysis helped us understand the market penetration and regional preferences.
- It also pointed out that Helsinki brings in the most revenue per sale.





Module 1, Lesson 2: Product Performance Analysis

- Finally, we looked at how well the products being sold performed by comparing stock levels against sales figures.
 - This helped us gauge supply and demand dynamics and identify fast- and slow-moving products and help us gain insights into which products should be pulled off the shelves.
 - NanoGrillflow had the highest sales volume (1,649 total sales) and average number of sales (6.84), but only brought in \$0.23 per sale
 - Portable Heater brought in the most total revenue – \$14,378.98 and a higher average revenue per sale (\$22.57 over 637 total sales)
 - Electric Can Opener brought the most revenue per sale (\$84.66), but had only 3 sales
 - There were 20 products that had no sales at all.

	A	B	C	D	E
	Product	Total Number of Sales	Total revenue	Average Number of Sales	Average revenue per sale
3	NanoGrillflow	1,649	381.68	6.84	0.23
4	Handheld Milk Frother	1,218	2,988.52	6.09	2.45
5	PowerDryerflow	867	966.95	6.83	1.12
6	Smart Speakers (with voice assistants)	700	440.13	2.83	0.63
7	Portable Heater	637	14,378.98	3.28	22.57
8	CyberHeaterhub	499	231.03	2.07	0.46
9	TechTVlux	341	315.92	1.43	0.93
10	SmartFridgewave	338	800.06	1.40	2.37
11	SolarGrillpulse	276	1,123.66	1.15	4.07
12	SolarBlenderlux	273	846.66	2.03	3.10
13	SmartFridgedrive	246	170.70	1.02	0.69
14	EcoVacuumcast	244	417.94	1.01	1.71
15	SmartVacuumcast	197	227.97	0.82	1.16
16	TechFridgematic	184	511.26	0.93	2.78
17	TechTVhub	160	146.86	0.70	0.92
18	Pizza Oven	160	147.35	1.03	0.92
19	CyberMixeromatic	157	108.91	0.80	0.69
20	Immersion Blender	151	397.54	1.03	2.63
21	MaxDryerwave	150	736.60	0.75	4.91
22	Smart TV	141	104.96	0.60	0.74
23	Smart Air Conditioner with Voice Control	136	102.00	0.69	0.75
24	LED TV	129	53.81	0.64	0.42
25	SmartBlenderflow	128	242.22	0.66	1.89
26	SolarOvenwave	117	352.14	0.61	3.01



Module 2, Lesson 1: Data Querying Using PostgreSQL

- A sales performance analysis was conducted in the database after importing and verifying the data.

The screenshot shows the pgAdmin interface with the following details:

- Object Explorer:** Shows the database schema with tables: city_names, product_hierarchy, product_names, and sales. The sales table is selected, revealing its 13 columns: product_id, store_id, date, sales, revenue, stock, price, promo_type_1, promo_bin_1, promo_type_2, promo_bin_2, promo_discount_2, and promo_discount_type_2.
- Query Editor:** Displays a SQL query for sales performance analysis. The query uses joins to link the sales table with the product_hierarchy and store_cities tables, and then groups the results by product_id, store_id, and city_id.
- Data Output:** Shows the results of the query in a tabular format. The columns are product_id, store_id, city_id, total_sales, total_revenue, and average_price. The data consists of 8 rows for product P0001 across different stores and cities.

product_id	store_id	city_id	total_sales	total_revenue	average_price
P0001	S0002	C007	0	0	6.25
P0001	S0012	C005	1	5.3	6.25
P0001	S0013	C026	2	10.59	6.25
P0001	S0023	C008	0	0	6.25
P0001	S0025	C024	0	0	6.25
P0001	S0027	C022	0	0	6.25
P0001	S0040	C017	0	0	6.25
P0001	S0049	C031	0	0	6.25



Module 2: Data Querying and Analysis Using PostgreSQL

- Next, we analyzed sales trends over time for each store and accumulate grand totals by grouping the data with ROLLUPS

The screenshot shows the pgAdmin interface with the following details:

- Object Explorer:** Shows the database structure with Schemas (1) and Tables (6). The `sales` table is selected.
- Query Editor:** Displays the SQL query used for the analysis:

```
1  SELECT store_id, TO_CHAR(date, 'YYYY-MM') AS "Date",
2    SUM(sales) AS "Sales"
3  FROM sales
4  GROUP BY ROLLUP (store_id, "Date");
```
- Data Output:** Shows the results of the query, which include 352 rows of sales data grouped by store ID and date. The columns are `store_id`, `Date`, and `Sales`.
- Messages:** Shows the completion message: "Query complete 00:00:00.146 Ln 4, Col 12".

	store_id	Date	Sales
1	[null]	[null]	9856.80999999998
2	S0013	2017-03	27
3	S0052	2017-02	23.675
4	S0120	2017-02	19
5	S0096	2017-03	36
6	S0135	2017-02	32.305
7	S0018	2017-02	38
8	S0024	2017-02	85
9	S0044	2017-02	22
10	S0127	2017-03	9
11	S0107	2017-03	27
12	S0072	2017-02	80
13	S0102	2017-03	23
14	S0082	2017-03	42
15	S0015	2017-03	24



Module 2: Data Querying and Analysis Using PostgreSQL

- We used a combination of the product hierarchy data and the sales data to get a summary of sales data across product hierarchy levels.

The screenshot shows the pgAdmin interface with the following details:

- Object Explorer:** Shows tables like `hierarchy1_id`, `hierarchy2_id`, `hierarchy3_id`, `hierarchy4_id`, `hierarchy5_id`, `product_names`, and `sales`.
- Query Editor:** Displays the following SQL query:

```
1 v SELECT
2 p.hierarchy1_id, p.hierarchy2_id,
3 SUM (s.sales) AS "Sales"
4 FROM
5 sales s
6 JOIN
7 product_hierarchy p
8 ON
9 s.product_id = p.product_id
10 GROUP BY
11 ROLLUP (p.hierarchy1_id, p.hierarchy2_id);
```
- Data Output:** Shows the results of the query in a table format:

	hierarchy1_id	hierarchy2_id	Sales
1	[null]	[null]	9856.880999999998
2	H03	H0317	2
3	H00	H0001	617
4	H03	H0316	0
5	H03	H0314	190
6	H03	H0315	5
7	H01	H0107	374
8	H03	H0313	581
9	H00	H0000	1380
10	H03	H0311	17

Total rows: 23 of 23 Query complete 00:00:00.172 Ln 2, Col 36



Module 2: Data Querying and Analysis Using PostgreSQL

- Finally, we identified sales trends over time and across various regions by combining the sales data and the data identifying store locations.
- The results were complementary to the ones received with the analyses done through Excel.
- The analyses performed with the help of the database allow Wally-Mart to segment the sales by
 - store performance
 - product category strength
 - regional characteristics
- this allows for better management of sales strategy and inventory management, eventually increasing revenue and decreasing expenses

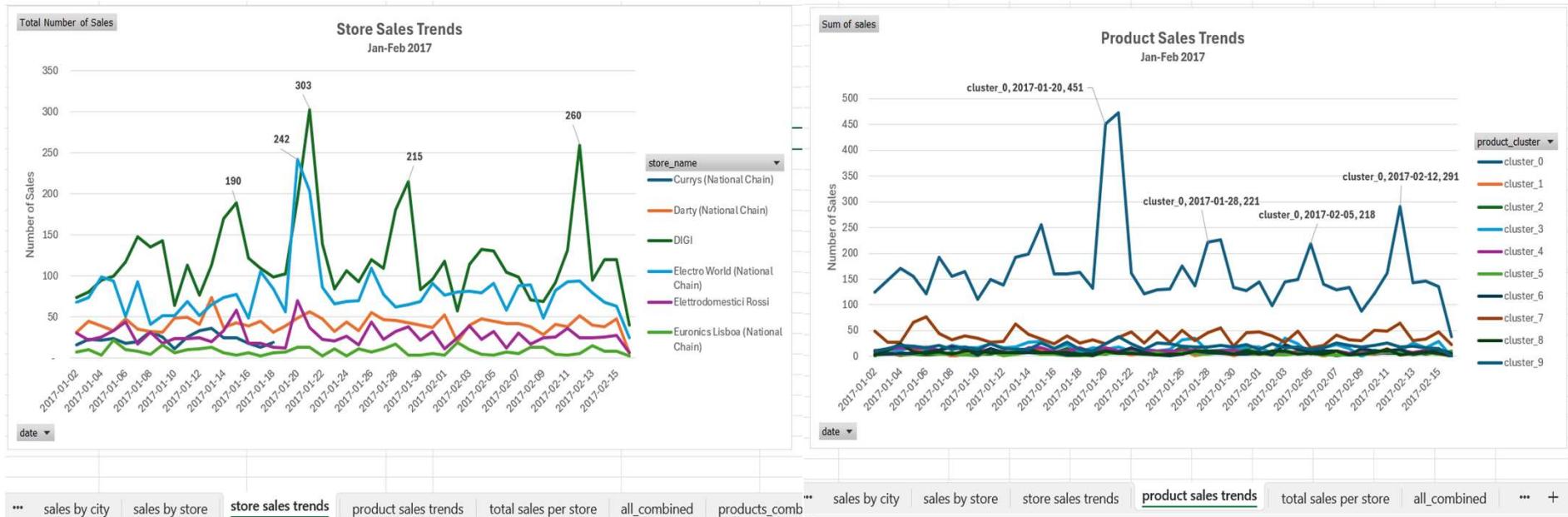
The screenshot shows the pgAdmin interface with the following details:

- Object Explorer:** Shows two tables: "sales" (13 columns) and "store_cities" (4 columns). The "sales" table includes columns like product_id, store_id, date, sales, revenue, stock, price, and various promo-related fields. The "store_cities" table includes columns like store_id, storetype_id, store_size, and city_id.
- Query Editor:** Displays a SQL query that performs a group by operation on the "store_cities" table based on city_id and date, summing up sales for each month.
- Data Output:** Shows the results of the query, which is a table with columns: city_id, Month, and Sum of sales. The data includes rows for various cities and months, with a total of 102 rows.

	city_id	Month	Sum of sales
1	[null]	[null]	9856.880999999998
2	C012	2017-02	39
3	C006	2017-02	124
4	C029	2017-03	92
5	C033	2017-03	33
6	C017	2017-03	44
7	C021	2017-03	22.555
8	C032	2017-02	49
9	C017	2017-02	83
10	C004	2017-03	110.208
11	C018	2017-03	23
12	C007	2017-03	28
13	C011	2017-02	54.97

Module 3, Lesson 1: Data Visualization Using Excel

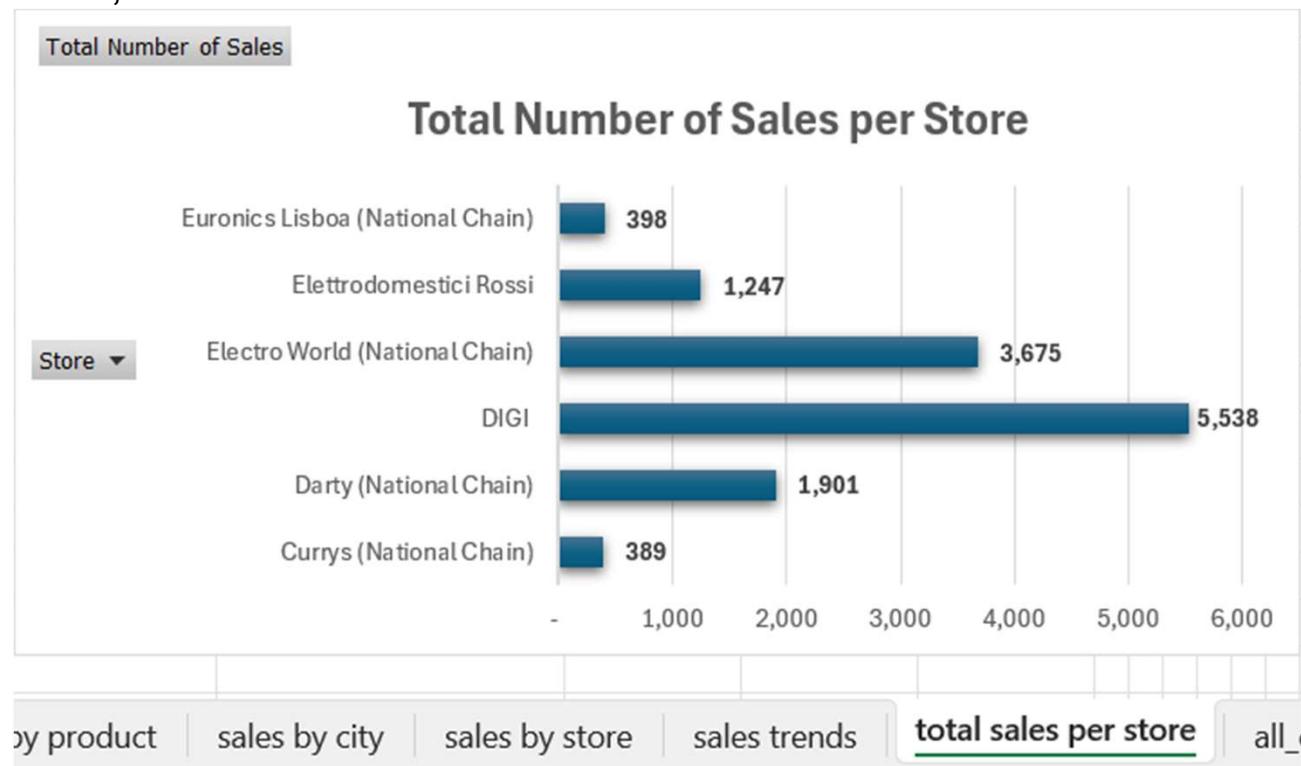
- After combining all the data, we created visualizations of the sales trends per store and per product cluster.
- Highest sales were on January 20, 2017, and second highest on February 12, 2017, for products belonging to product cluster 0.
- Sales were highest in a store named DIGI.





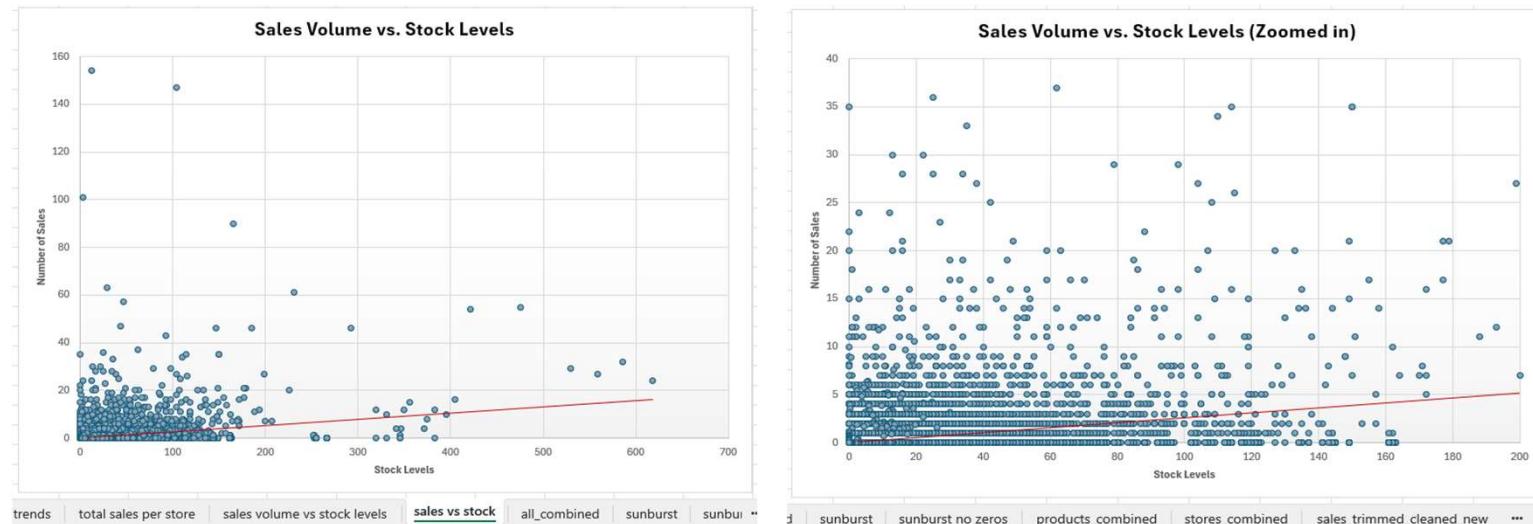
Module 3, Lesson 1: Data Visualization Using Excel

- When we compare the total number of sales for each store, we can confirm that DIGI had the most sales – 5,538.



Module 3, Lesson 1: Data Visualization Using Excel

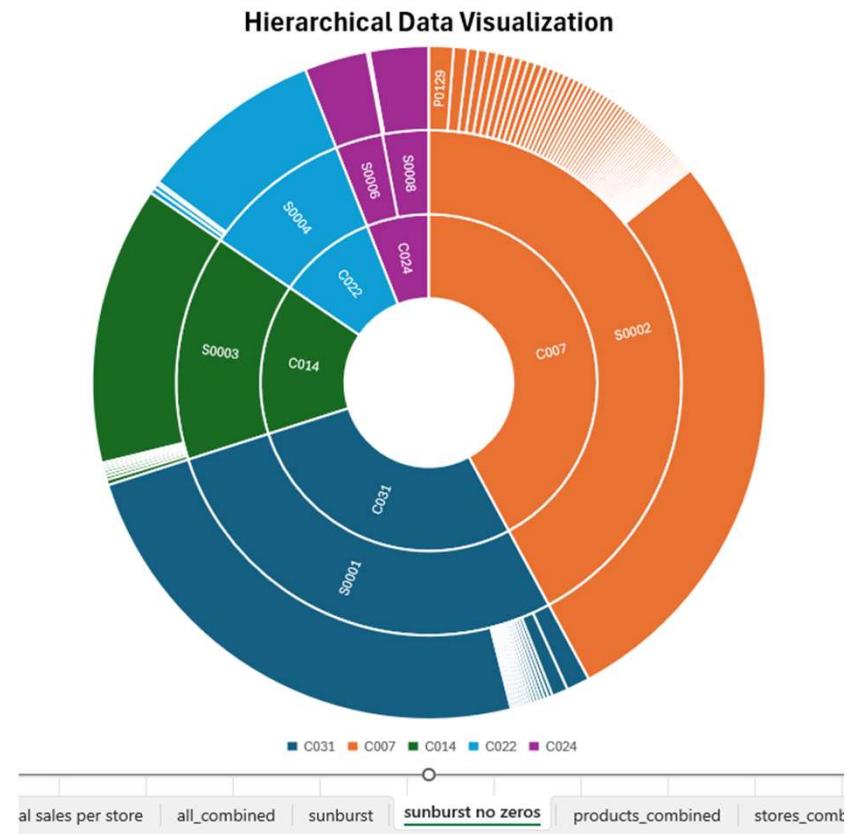
- We wanted to see the relationship between the number of sales and stock levels. The red trendline shows a small upward relationship, suggesting that as the stock levels went up, so did the sales.
- The graph (on the left) showed that there were several outliers, so we created a zoomed in chart (on the right) that focused on the majority of the data, excluding most of the outliers.





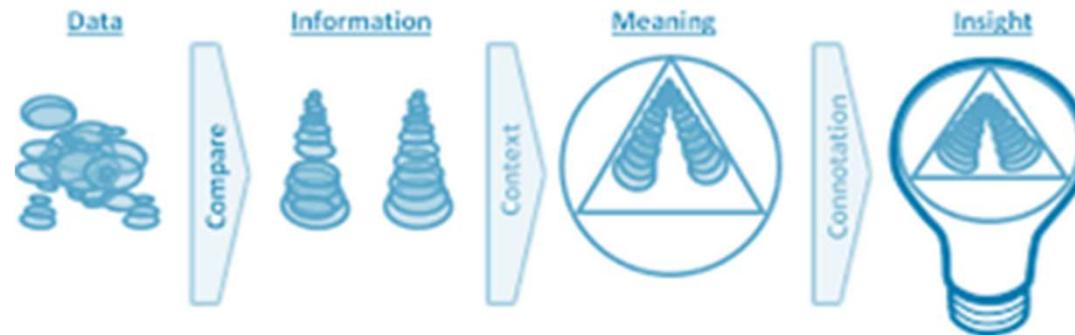
Module 3, Lesson 1: Data Visualization Using Excel

- When we holistically look at our sales data and present it as a sunburst chart (which allows us to look at the data hierarchically), we can see that the city with the largest number of sales is Edinburgh (id C007) with 1 store – DIGI (id S0002) and many different products.
 - The second largest is Helsinki (C031) with Electro World (S0001) but with fewer products being sold.
 - The city that had the least number of total sales is Vienna (C024) with 2 stores – Euronics Lisboa (S0006) and Currys (S0008)



Module 3, Lesson 1: Data Visualization Using Excel

- Using visualizations like the various graphs we presented makes it much easier to:
 - comprehensively understand complex data
 - be able to discern various trends that may be hidden when the view presents only numbers
 - communicate and engage stakeholders
- Faster and better understanding helps identify insights and make better decisions.





Module 3, Lesson 2: Statistical Analysis

- Using the combined worksheet data, we performed a regression analysis to see if the number of sales depended on the date.
- 20,000 rows of data were analyzed
- The results of the analysis suggest that our model:
 - explains only 0.000653% of the variance ($R^2=6.5294E-06$)
 - shows the expected change in the number of sales for each day is 0.000556 (Coefficient for date)
 - is **not** statistically significant (p-value is 0.71784, much greater than 0.05)
- Essentially, this model does not explain or predict anything about the number of sales.

	A	B	C	D	E	F	G	H	I	
1	SUMMARY OUTPUT									
2										
<i>Regression Statistics</i>										
3										
4	Multiple R		0.00255527							
5	R Square		6.52941E-06							
6	Adjusted R Square		-4.34753E-05							
7	Standard Error		2.805644826							
8	Observations		20000							
9										
10	ANOVA									
11		df	SS	MS	F	Significance F				
12	Regression	1	1.027846923	1.027846923	0.130575909	0.71783948				
13	Residual	19998	157417.1145	7.87164289						
14	Total	19999	157418.1424							
15										
16		Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	
17	Intercept	-23.25708307	65.83469919	-0.353264819	0.723893642	-152.2985326	105.7843664	-152.2985326	105.7843664	
18	date	0.000556371	0.001539689	0.361352883	0.717839483	-0.002461547	0.003574289	-0.002461547	0.003574289	
19										
	<	>	Revenue Regression	Sales Regression	sales_trimmed_cleaned_new	+	:	◀	▶	

Module 3, Lesson 2: Statistical Analysis

- Since the last analysis does not help us gain any insights, we ran a second regression analysis that looked at the relationship between the stock levels and the number of sales.
- 20,000 rows of data were analyzed
- The results of this analysis suggest that our model
 - explains 11.05% of the variance ($R^2=0.11047794$)
 - shows that the expected change in the number of sales for 1-unit increase of stock is 0.033669739 (Coefficient for stock)
 - is statistically significant** (p-value is 0.00000, less than 0.05)
- This model, although still not very good, is statistically significant and does explain about 11% of the variation, suggesting that the number of sales will increase by 1 for about every 30 units of stock.
- We also tried removing the rows with 0 sales (resulting in an analysis of 5,067 rows of data).
- That model was able to explain almost 12% of the variance and suggested an increase of 1 sale for every 20 units of stock.

A	B	C	D	E	F	G	H	I
1 SUMMARY OUTPUT	Stock vs. Sales							
2								
3 Regression Statistics								
4 Multiple R	0.33238222							
5 R Square	0.11047794							
6 Adjusted R Square	0.11043346							
7 Standard Error	2.311200069							
8 Observations	20000							
9								
10 ANOVA								
11	df	SS	MS	F	Significance F			
12 Regression	1	13267.2372	13267.2372	2483.735873	0			
13 Residual	19998	106822.2319	5.34164576					
14 Total	19999	120089.4691						
15								
16	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%
17 Intercept	-0.047173585	0.019374922	-2.434775521	0.014909715	-0.085150031	-0.009197138	-0.085150031	-0.009197138
18 stock	0.033669739	0.000675596	49.83709335	0.000000000	0.032345516	0.034993963	0.032345516	0.034993963



Module 4, Lesson 1: Basic Tableau Visualizations

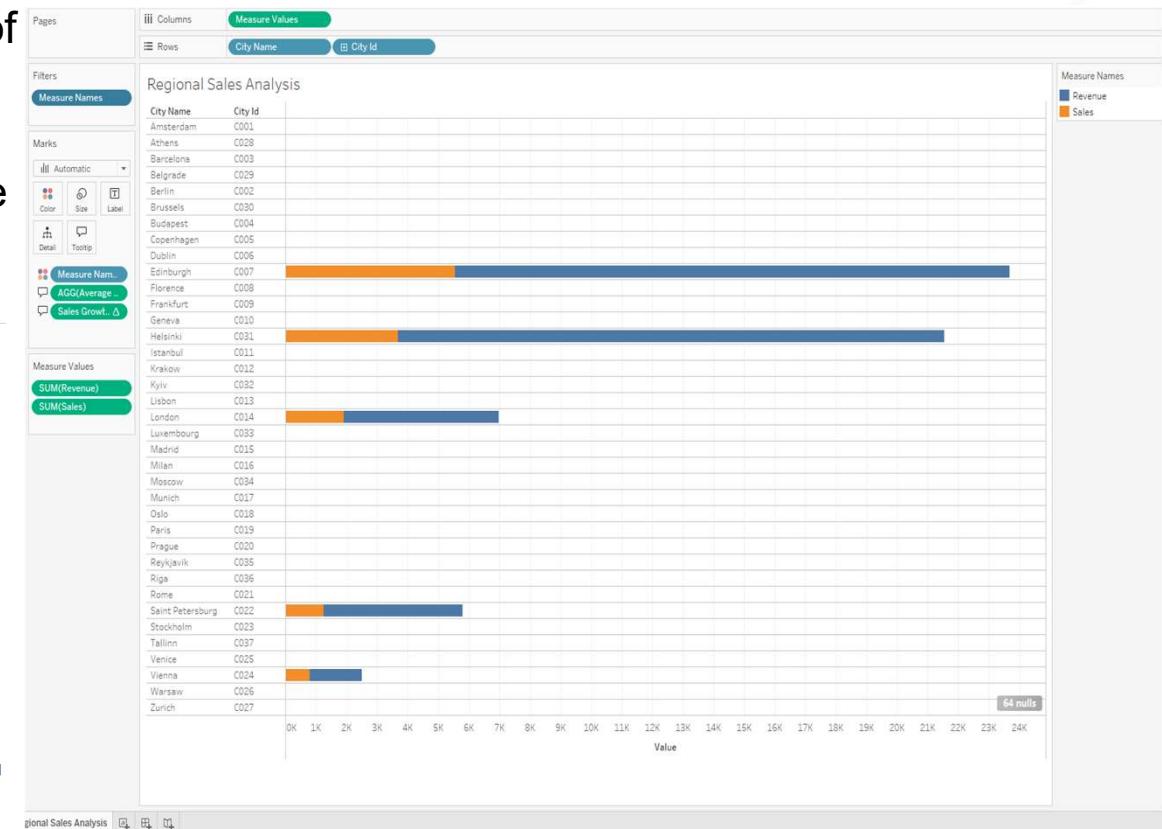
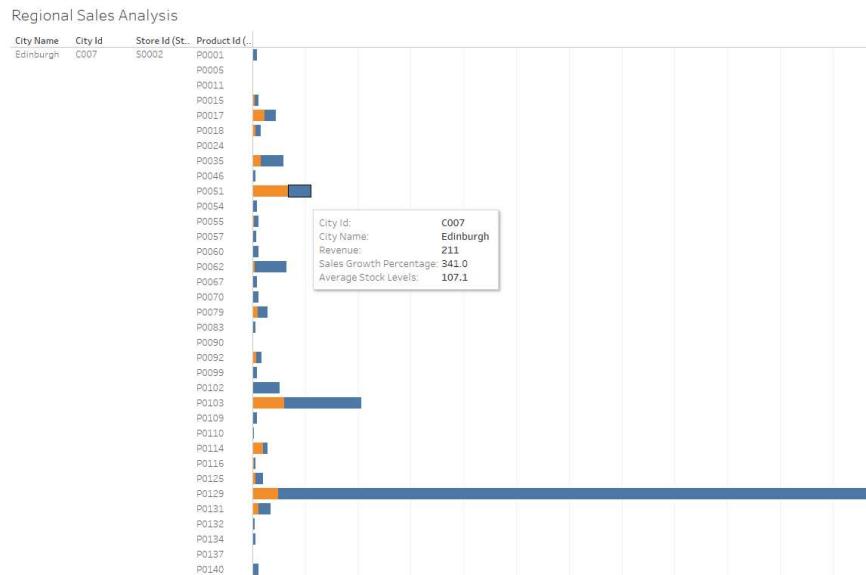
- After setting up our data in Tableau, we created a Sales Performance chart.
- Our first visualization showed the revenue, number of sales and amount of stock per day per city.
- It showed a large jump in revenue around January 21st, 2017, consistent with the Excel chart.
- The number of sales tended to fluctuate, but the fluctuations were kept within the same range per city.
- Stock levels for all cities were higher until around January 17th, 2017, and then all dropped down by about 2,000 units per each store.





Module 4, Lesson 1: Basic Tableau Visualizations

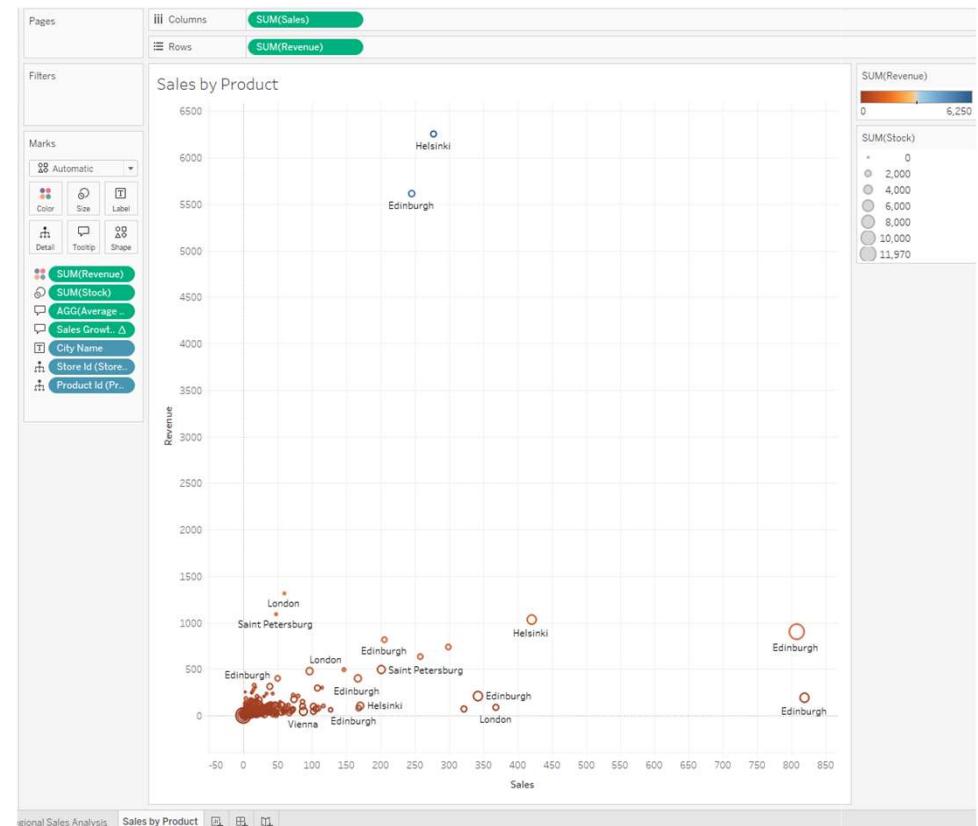
- Next, we analyzed the total number of sales and revenue by region.
- Using our built-in hierarchy, we could drill down through each city and store down to the product level.



Module 4, Lesson 1: Basic Tableau Visualizations



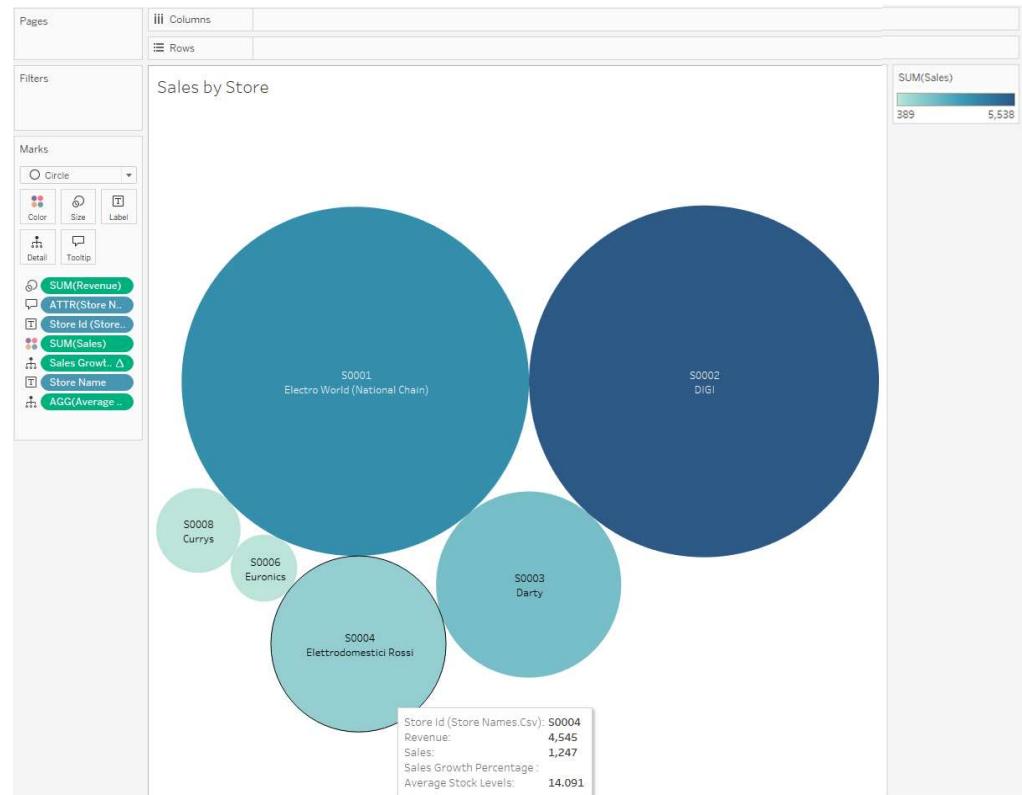
- Then we looked at the total number of sales vs. total revenue for each combination of city/product.
 - We can see the outliers on the bottom right for Edinburgh – showing a high number of sales with low total revenue for NanoGrillflow (P0438) and the PowerDryerflow (P0282)
 - Two more outliers are visible on the top for Edinburgh and Helsinki – showing a smaller number of sales with a high revenue, both for the same product – the Portable Heater (P0129)





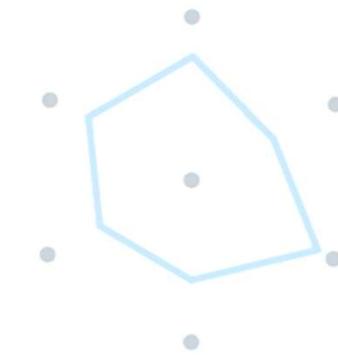
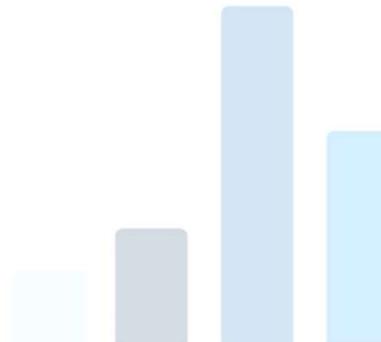
Module 4, Lesson 1: Basic Tableau Visualizations

- Finally, we created a visualization comparing stores based on the number of sales.
- Each bubble's
 - size is based on the number of sales
 - color is based on the amount of revenue
- We got the expected result of DIGI (S0002) having the highest revenue and largest number of sales. Followed closely by Electro World (S0001).
- Currys (S0008), with the fewest sales, had 1.5 times higher revenue than the next store with the fewest sales – Euronics (S0006).



Module 4, Lesson 1: Basic Tableau Visualizations

- Visual representations in Tableau makes it much easier to see multidimensional data at once and to quickly zero in on important points.
- Tooltips allow a lot more information to be displayed when required, but do not clutter the visualization.
- Calculated fields add new, aggregated information that helps understand the data better.
- Dynamic filters help stakeholders see the exact level of detail that each one of them is interested in.



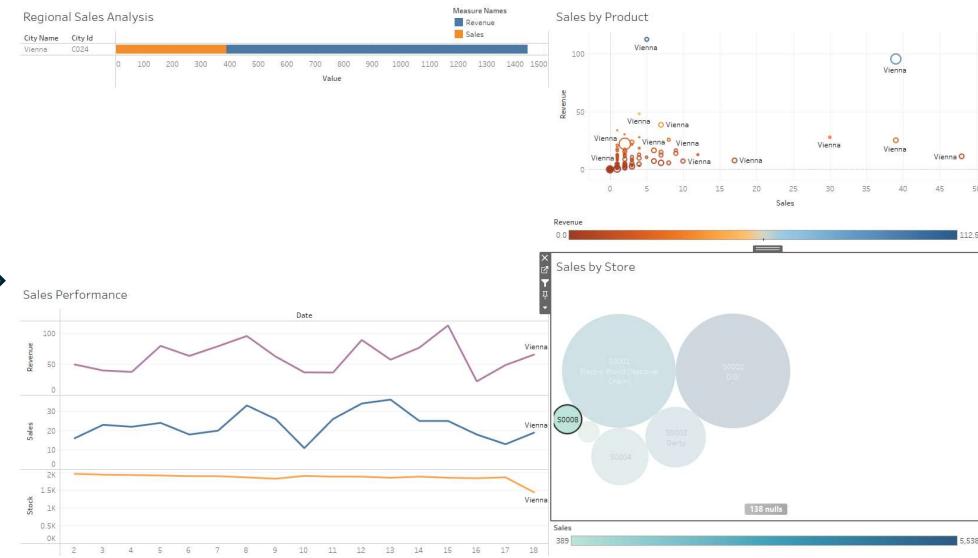
Module 4, Lesson 2: Advanced Visualizations Using Tableau

- The last step in our analysis included putting all the visualizations together.
- An interactive filter was added, allowing drilling down by store from the Sales by Store graph that affected all the other graphs.

Performance Analysis



Performance Analysis



Module 4, Lesson 2: Advanced Visualizations Using Tableau

- Dashboards in Tableau closely support the decision-making processes. They :
 - provide a holistic overview of the data through the consolidation of multiple views in one
 - enable interactive exploration with the help of filters and other interactive elements
 - allow easy and quick exploration of multidimensional data
 - visually engage the stakeholders presenting the data in a more memorable way
 - help spot outliers by visually displaying large amounts of numerical data
 - let each stakeholder see the data that's most relevant to them





Discussion

Insights and Implications

The analysis of the past sales data for Wally-Mart has brought to light that

- 37.7% of all store revenue is generated in Helsinki
- 30.4% of all product revenue is generated by the Portable Heater
- the Electric Can Opener brings in \$84.66 per sale, but has a very small number of sales
- 20 products have no sales and no generated revenue at all, but have over 17 thousand units of stock
- there is no relationship between the date and the number of sales
- 11-12% of the variance in the number of sales can be explained by the levels of stock
- Vienna had the lowest number of sales even though it has 2 stores
- prior sales data only covers 1.5 months, 5 cities and 6 stores and can provide limited temporal and regional trends





Recommendations

Based on our analysis, we recommend that Wally-Mart implement the following improvements:

- Remove the products that have not sold any units from the stores.
- Evaluate a way to sell more Electric Can Openers.
- Assess the necessity of running 2 stores in Vienna.
- Improve data collection and availability to improve analysis and further increase profitability.

Recommendation



Effective Communication

Throughout this presentation we have tried to

- keep a clear structure, making the presentation easy to follow
- include visually interesting charts and graphs
- explain the methodology we used in our analysis
- point out key insights
- provide actionable recommendations
- keep consistent formatting and spacing throughout

We hope that our presentation effectively communicated the results and helped you understand the value of our analysis.





Conclusion



Summary

Wally-Mart is a leading global retail chain renowned for its diverse product offerings and commitment to customer satisfaction.



To keep its competitive edge and optimize business performance we evaluated its existing sales data and recommend that:

- Products with low sales be cut.
- More focus be placed on products with high average revenue per sale.
- Cities with low sales and revenue cut back on the number of operating stores.
- Improve the collection of sales data.





Appendix



Appendix: List of Software

The following software was used to complete this project:

- Microsoft Excel
- Microsoft PowerPoint
- PostgreSQL (CLI and pgAdmin)
- Tableau Desktop and Tableau Public
- Coursera website – The Business Intelligence (BI) Analyst Capstone Project
- Included images were available in the public domain or created by the author



Appendix # 1

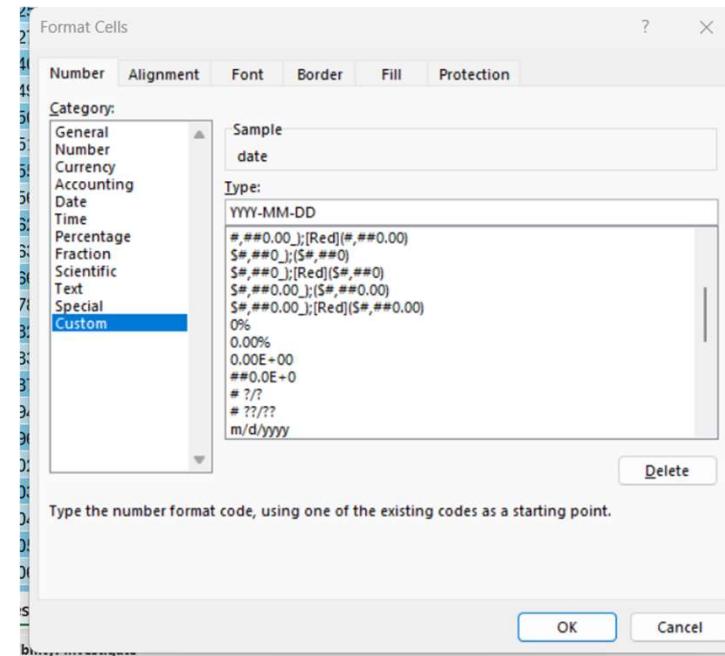
Importing original .csv files into Excel

sales.csv

File Origin		Delimiter			Data Type Detection					
1252: Western European (Windows)		Comma			Based on first 200 rows					
product_id	store_id	date	sales	revenue	stock	price	promo_type_1	promo_bin_1	promo_type_2	promo_bin_2
P0001	S0002	2/1/2017	0	0	8	6.25	PR14			PR03
P0001	S0012	2/1/2017	1	5.3	0	6.25	PR14			PR03
P0001	S0013	2/1/2017	2	10.59	0	6.25	PR14			PR03
P0001	S0023	2/1/2017	0	0	6	6.25	PR14			PR03
P0001	S0025	2/1/2017	0	0	1	6.25	PR14			PR03
P0001	S0027	2/1/2017	0	0	7	6.25	PR14			PR03
P0001	S0040	2/1/2017	0	0	19	6.25	PR14			PR03
P0001	S0049	2/1/2017	0	0	8	6.25	PR14			PR03
P0001	S0050	2/1/2017	0	0	5	6.25	PR14			PR03
P0001	S0051	2/1/2017	0	0	6	6.25	PR14			PR03
P0001	S0055	2/1/2017	0	0	6	6.25	PR14			PR03
P0001	S0056	2/1/2017	1	5.3	6	6.25	PR14			PR03
P0001	S0062	2/1/2017	0	0	2	6.25	PR14			PR03
P0001	S0063	2/1/2017	0	0	7	6.25	PR14			PR03
P0001	S0066	2/1/2017	0	0	1	6.25	PR14			PR03
P0001	S0078	2/1/2017	0	0	6	6.25	PR14			PR03
P0001	S0082	2/1/2017	0	0	4	6.25	PR14			PR03
P0001	S0083	2/1/2017	0	0	10	6.25	PR14			PR03
P0001	S0087	2/1/2017	0	0	10	6.25	PR14			PR03
P0001	S0094	2/1/2017	0	0	6	6.25	PR14			PR03

Appendix # 2

Setup of custom date fields in Excel





Appendix # 3

Confirmation that data from the table city_names imported correctly into PostgreSQL.

The screenshot shows the PgAdmin interface. The Object Explorer on the left lists various database objects: Schemas (1), public, Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (6). The 'city_names' table is selected. The main window displays a SQL query and its results. The SQL query is:

```
1 SELECT * FROM city_names LIMIT 5;
```

The results table shows the following data:

	city_id	city_name
1	C001	Amsterdam
2	C002	Berlin
3	C003	Barcelona
4	C004	Budapest
5	C005	Copenhagen

Total rows: 5 of 5 Query complete 00:00:00.181 Ln 2, Col 1



Appendix # 4

Confirmation that data from the table product_hierarchy imported correctly into PostgreSQL

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database schema with several tables, including 'product_hierarchy'. The main area contains a query window with the following SQL command:

```
SELECT * FROM product_hierarchy LIMIT 5;
```

The results are displayed in a Data Output tab, showing five rows of data from the 'product_hierarchy' table:

	product_id	product_length	product_depth	product_width	cluster_id	hierarchy1_id	hierarchy2_id	hierarchy3_id	hierarchy4_id	hierarchy5_id
	character varying	double precision	double precision	double precision	character varying					
1	P0000		5	20	12	[null]	H00	H0004	H00040105	H0004010534
2	P0001		13.5	22	20	cluster_5	H01	H0105	H01050100	H0105010006
3	P0002		22	40	22	cluster_0	H03	H0315	H03150800	H0315080028
4	P0004		2	13	4	cluster_3	H03	H0314	H03140500	H0314050003
5	P0005		16	30	16	cluster_9	H03	H0312	H031211	H03121109

At the bottom of the interface, a status bar indicates: Total rows: 5 of 5 Query complete 00:00:00.160 Ln 1, Col 32.



Appendix # 5

Confirmation that data from the table
product_names imported correctly into PostgreSQL

Screenshot of PgAdmin 4 showing the results of a SQL query against the product_names table.

The Query window contains:

```
1 SELECT * FROM product_names LIMIT 5;
```

The Data Output window displays the following table:

	product_id	product_name
1	P0000	Bottom Freezer Refrigerator
2	P0001	Side-by-Side Refrigerator
3	P0002	Mini Fridge
4	P0004	Beverage Center
5	P0005	Wine Cooler

Total rows: 5 of 5 Query complete 00:00:00.220 Ln 1, Col 28



Appendix # 6

Confirmation that data from the table store_cities imported correctly into PostgreSQL

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure under the 'Schemas' node, specifically the 'Tables' section which contains six entries: city_names, product_hierarchy, product_names, sales, store_cities, and store_names. The 'sales' table is currently selected. The main pane shows a query editor with the following SQL command:

```
SELECT * FROM store_cities LIMIT 5;
```

The results are displayed in a Data Output tab, showing five rows of data:

	store_id	storetype_id	store_size	city_id
1	S0091	ST04	19	C013
2	S0012	ST04	28	C005
3	S0045	ST04	17	C008
4	S0032	ST03	14	C019
5	S0027	ST04	24	C022

At the bottom of the pgAdmin window, status messages indicate: "Total rows: 5 of 5 Query complete 00:00:00.142 Ln 1, Col 27".



Appendix # 7

Confirmation that data from the table store_names imported correctly into PostgreSQL

The screenshot shows the pgAdmin 4 interface. On the left is the Explorer pane, which lists Schemas (1) under the public schema, and Tables (6) including city_names, product_hierarchy, product_names, sales, store_cities, store_names, Trigger Functions, Types, and Views. The store_names table is selected. The main pane shows a query window with the command:

```
1 SELECT * FROM store_names LIMIT 5;
```

Below the query window is a Data Output pane displaying the results of the query:

store_id	store_name
S0005	MediaMarkt(National Chain)
S0036	Saturn
S0104	Euronics
S0068	FNAC
S0086	Darty

At the bottom of the pgAdmin interface, status messages indicate: Total rows: 5 of 5, Query complete 00:00:00.328, Ln 1, Col 26.



Appendix # 8

Table of data used to create the sunburst chart in Excel

	A	B	C	D	E
1	city_id	store_id	product_id	sales	
2	C031	S0001	P0015	1.00	
3	C031	S0001	P0035	2.00	
4	C031	S0001	P0103	13.00	
5	C031	S0001	P0110	1.00	
6	C031	S0001	P0140	2.00	
7	C031	S0001	P0148	4.00	
8	C031	S0001	P0169	1.00	
9	C031	S0001	P0185	2.00	
10	C031	S0001	P0219	1.00	
11	C031	S0001	P0287	1.00	
12	C031	S0001	P0311	3.00	
13	C031	S0001	P0327	1.00	
14	C031	S0001	P0333	3.00	
15	C031	S0001	P0348	2.00	
16	C031	S0001	P0413	1.70	
17	C031	S0001	P0436	1.00	
18	C031	S0001	P0438	9.00	
19	C031	S0001	P0439	3.00	
20	C031	S0001	P0499	1.00	
21	C031	S0001	P0527	1.00	
22	C031	S0001	P0663	7.00	
23	C031	S0001	P0664	3.00	
24	C031	S0001	P0704	1.00	
25	C031	S0001	P0718	2.00	
26	C031	S0001	P0726	1.00	
27	C031	S0001	P0051	3.00	
28	C031	S0001	P0090	2.00	
29	C031	S0001	P0103	8.00	
30	C031	S0001	P0140	3.00	
31	C031	S0001	P0165	1.00	
32	C031	S0001	P0185	1.00	
33	C031	S0001	P0211	1.00	
34	C031	S0001	P0219	1.00	
35	C031	S0001	P0225	1.00	
36	C031	S0001	P0261	1.00	
37	C031	S0001	P0296	5.00	
38	C031	S0001	P0311	2.00	
39	C031	S0001	P0348	2.00	
40	C031	S0001	P0364	2.00	
41	C031	S0001	P0389	2.00	
42	C031	S0001	P0413	5.31	

Appendix # 9

The following is the comparison of the linear regression tests we have done to find a statistically significant and useful predictor for sales. The tests are filtered to show the results only for statistically significant tests.

Observations	20000 for all regressions															
	Date vs. Sales	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
5	City vs. Sales	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
6	C007	2.440719763	0.008085958	0.47808362	0.0479983	9.96042827	2.5735E-23	0.384002987	0.572164253	-6.30922164	7.265389403	0.808596%	statistically significant	0.47808362		
7	C014	2.440719763	0.008085958	0.447460846	0.059980583	7.46009497	8.9991E-14	0.329893947	0.565027745	-8.034226259	8.92914795	0.808596%	statistically significant	0.447460846		
11	Store vs. Sales	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
14	S0003	2.440709863	0.008143609	0.484457741	0.069104108	7.01054908	2.4495E-12	0.349007979	0.619907503	-9.287117262	10.25603274	0.814361%	statistically significant	0.484457741		
18	Product cluster vs. Sales	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
26	cluster_7	2.434546951	0.013393032	1.539101449	0.135959884	11.3202616	1.282E-29	1.272608837	1.805594061	1.272608837	1.805594061	1.339303%	statistically significant	1.539101449		
29	Date vs. Revenue	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
31	City vs. Revenue	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
32	C007	29.92163218	0.000588187	1.791733556	0.588427842	3.04495034	0.00233021	0.638366364	2.945100748	-81.41620791	84.99967502	0.058819%	statistically significant	1.791733556		
37	Store vs. Revenue	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
44	Product cluster vs. Revenue	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
55	City vs. Stock levels	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
56	C007	24.17680704	0.001288536	0.997671126	0.475452218	2.09836254	0.03588564	0.065745492	1.929596761	-66.23470219	68.23004445	0.128854%	statistically significant	0.997671126		
58	C022	24.17680704	0.001288536	-1.190161672	0.519655928	-2.29028788	0.02201499	-2.20873023	-0.171593114	-74.6732584	72.29293506	0.128854%	statistically significant	-1.190161672		
61	Store vs. Stock levels	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
64	S0003	24.1334415	0.004917834	3.003147416	0.683292993	4.39510934	1.1129E-05	1.663836685	4.342458147	-93.61699464	99.62328947	0.491783%	statistically significant	3.003147416		
65	S0004	24.1334415	0.004917834	1.70764707	0.619856333	2.75490784	0.00587618	0.492677435	2.922616705	-85.94231798	89.35761212	0.491783%	statistically significant	1.70764707		
66	S0006	24.1334415	0.004917834	6.779507936	0.793881426	8.5396984	1.4406E-17	5.22343474	8.335581133	-105.4782451	119.037261	0.491783%	statistically significant	6.779507936		
68	Product cluster vs. Stock levels	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
72	cluster_3	23.8817121	0.025812104	3.792981809	1.179700912	3.2152063	0.00130558	1.480670501	6.105293117	1.480670501	6.105293117	2.581210%	statistically significant	3.792981809		
73	cluster_4	23.8817121	0.025812104	-3.757408126	1.207300698	-3.11223884	0.00185936	-6.123817295	-1.390998957	-6.123817295	-1.390998957	2.581210%	statistically significant	-3.757408126		
74	cluster_5	23.8817121	0.025812104	8.118521418	1.350243312	6.0126359	1.857E-09	5.471932909	10.76510993	5.471932909	10.76510993	2.581210%	statistically significant	8.118521418		
76	cluster_7	23.8817121	0.025812104	12.11907985	1.333699811	9.08681231	1.1126E-19	9.504917974	14.73324173	9.504917974	14.73324173	2.581210%	statistically significant	12.11907985		
78	cluster_9	23.8817121	0.025812104	6.815447962	1.168167723	5.83430601	5.4848E-09	4.525742659	9.105153265	4.525742659	9.105153265	2.581210%	statistically significant	6.815447962		
79	Stock levels vs. sales	Standard Error	R Square	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	% of variance explained by model	significance	expected change for each 1 unit increase		
80	stock	2.311200069	0.11047794	0.033669739	0.000675596	49.8370933	0	0.032345516	0.034993963	0.032345516	0.034993963	11.047794%	statistically significant	0.033669739		