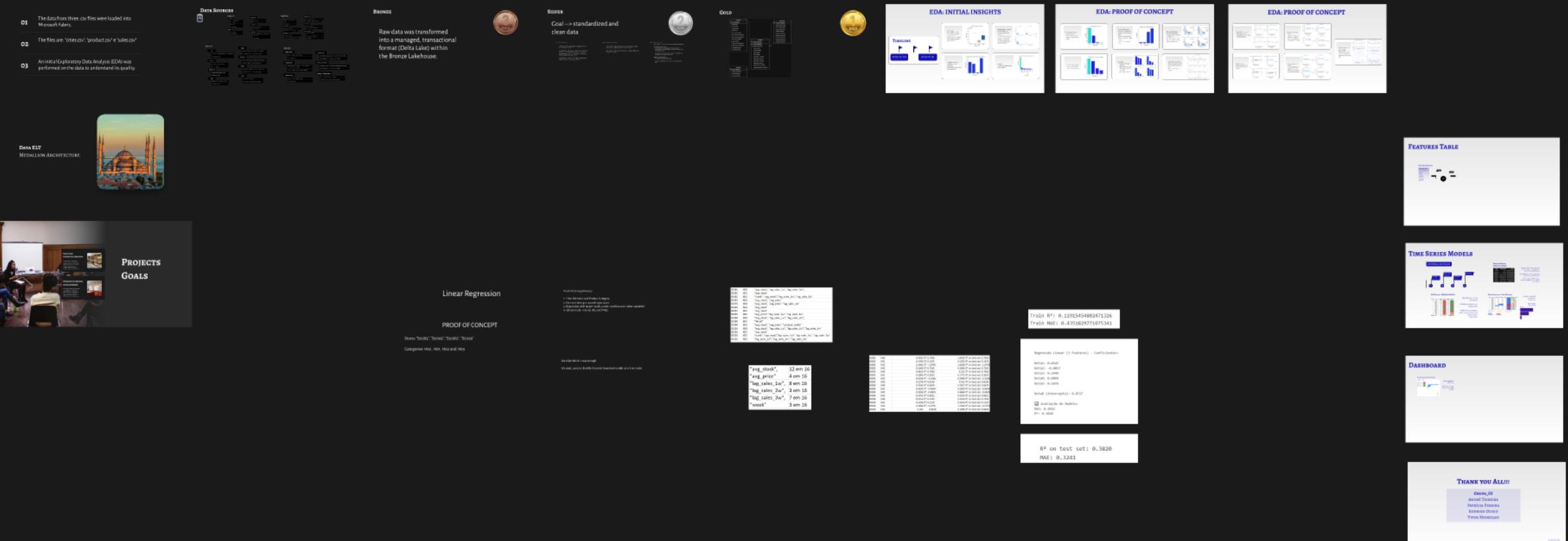


PREDICTIVE MODEL FOR INVENTORY MANAGEMENT

Sales Analysis and Forecasting in Department Stores in Turkey



PROJECTS GOALS



SALES AND INVENTORY ANALYSIS

The project aims for a detailed analysis of sales and inventory in department stores in Turkey. This analysis is essential for understanding consumption patterns and inventory dynamics.



PREDICTIVE MODEL DEVELOPMENT

Based on this analysis, the project aims to develop an effective predictive model that allows for the forecasting of weekly sales, thereby contributing to more efficient inventory management.



SALES AND INVENTORY ANALYSIS

The project aims for a detailed analysis of sales and inventory in department stores in Turkey. This analysis is essential for understanding consumption patterns and inventory dynamics.



PREDICTIVE MODEL DEVELOPMENT

Based on this analysis, the project aims to develop an effective predictive model that allows for the forecasting of weekly sales, thereby contributing to more efficient inventory management.



DATA ELT MEDALLION ARCHITECTURE



O1

The data from three .csv files were loaded into Microsoft Fabric.

O2

The files are: "cities.csv", "product.csv" e "sales.csv"

O3

An initial Exploratory Data Analysis (EDA) was performed on the data to understand its quality.

DATA SOURCES



CITIES.CSV	STORE_SIZE	PRODUCT.CSV	PRODUCT_WIDTH
STORE_ID Store Unique Identifier Code 63 distinct stores No duplicates No missing values	STORE_SIZE Store Size Value No missing values	PRODUCT_ID Product Unique Identifier Code 699 distinct products No missing values	PRODUCT_WIDTH Product Width Value (cm) 16 records with missing values (2.28%)
STORETYPE_ID Store Type Code No missing values	CITY_ID_OLD City Unique Identifier Code No missing values	PRODUCT_LENGTH Product Length Value (cm) 18 records with missing values (2.57%)	CLUSTER_ID Product Cluster Code 50 records with missing values (7.15%)
COUNTRY_ID Country Name (where stores and cities are located) Only "TURKEY" No missing values		PRODUCT_DEPTH Product Depth Value (cm) 16 records with missing values (2.28%)	HIERARCHY_ID (1 à 5) Product Hierarchy Code Ranges from the broadest hierarchy (hierarchy1_id) to the most specific (hierarchy5_id) No missing values
SALES.CSV	SALES	SALES.CSV	PROMO_BIN_2
_CO Column with repeated indexes from the dataset No missing values	SALES Quantity of Product X Sold at Store Y on Date Z 302,296 fields with missing values (3.40%)	PROMO_TYPE_1 Promotion Type 1 Identifier Code No missing values	PROMO_BIN_2 Column with Unique Values 'verylow', 'high' and 'veryhigh' 8,873,337 fields with missing values (99.85%)
STORE_ID Store Unique Identifier Code No missing values	REVENUE Revenue (€) of Product X Sold at Store Y on Date Z 302,296 fields with missing values (3.40%)	PROMO_BIN_1 Column with Unique Values 'verylow', 'moderate', 'low', 'high' and 'veryhigh' 7,653,515 fields with missing values (86.13%)	PROMO_DISCOUNT_2 Column with Values that Appear to Be Discount Percentages 8,873,337 fields with missing values (99.85%)
PRODUCT_ID Product Unique Identifier Code No missing values	STOCK Stock Quantity of Product X at Store Y on Date Z 302,296 fields with missing values (3.40%)	PROMO_TYPE_2 Promotion Type 2 Identifier Code No missing values	PROMO_DISCOUNT_TYPE_2 Discount Code for Promotion Type 2 8,873,337 fields with missing values (99.85%)
DATE Sale Date (for product X at store Y) No missing values	PRICE Unit Price of Product (€) 91,381 fields with missing values (1.03%)		

CITIES.CSV

STORE_ID

Store Unique Identifier Code

63 distinct stores

No duplicates

No missing values

STORETYPE_ID

Store Type Code

No missing values

STORE_SIZE

Store Size Value

No missing values

CITY_ID_OLD

City Unique Identifier Code

No missing values

COUNTRY_ID

Country Name (where stores and cities are located)

Only "TURKEY"

No missing values

PRODUCT.CSV

PRODUCT_ID

Product Unique Identifier Code

699 distinct products

No missing values

PRODUCT_LENGTH

Product Length Value (cm)

18 records with missing values (2.57%)

PRODUCT_WIDTH

Product Width Value (cm)

16 records with missing values (2.28%)

CLUSTER_ID

Product Cluster Code

50 records with missing values (7.15%)

HIERARCHY_ID (1 À 5)

Product Hierarchy Code

Ranges from the broadest hierarchy (hierarchy1_id) to the most specific (hierarchy5_id)

No missing values

PRODUCT_DEPTH

Product Depth Value (cm)

16 records with missing values (2.28%)

SALES.CSV

	SALES	
_CO	Quantity of Product X Sold at Store Y on Date Z	
Column with repeated indexes from the dataset	302,296 fields with missing values (3.40%)	
No missing values		
STORE_ID	Revenue (€) of Product X Sold at Store Y on Date Z	
Store Unique Identifier Code	302,296 fields with missing values (3.40%)	
No missing values		
PRODUCT_ID	Stock Quantity of Product X at Store Y on Date Z	
Product Unique Identifier Code	302,296 fields with missing values (3.40%)	
No missing values		
DATE	Unit Price of Product (€)	
Sale Date (for product X at store Y)	91,381 fields with missing values (1.03%)	
No missing values		

SALES.CSV

PROMO_TYPE_1

Promotion Type 1 Identifier Code

No missing values

PROMO_BIN_2

Column with Unique Values 'verylow', 'high', and 'veryhigh'

8,873,337 fields with missing values (99.85%)

PROMO_BIN_1

Column with Unique Values 'verylow', 'moderate',
'low', 'high', and 'veryhigh'

7,653,515 fields with missing values (86.13%)

PROMO_DISCOUNT_2

Column with Values that Appear to Be Discount Percentages

8,873,337 fields with missing values (99.85%)

PROMO_TYPE_2

Promotion Type 2 Identifier Code

No missing values

PROMO_DISCOUNT_TYPE_2

Discount Code for Promotion Type 2

8,873,337 fields with missing values (99.85%)

BRONZE

Raw data was transformed
into a managed, transactional
format (Delta Lake) within
the Bronze Lakehouse.



SILVER



Goal --> standardized and clean data

cities_bronze -> cities_silver

- City Name Correction: Removed the '?' character from city names. Ex: ?zmir --> Izmir
- Column Removal - city_old_id: This column was removed as it appeared to be an old city ID. Since the granularity of this table is based on store_id, this old city ID was not relevant, as the city name depends on the store_id.
- Column Removal - country_id: This column was removed as all data pertains to the same country (Turkey).

product_bronze -> product_silver

- Null Handling - Product Dimensions: Null values in product_length, product_depth, and product_width were filled with 0.0.
- Null Handling - cluster_id: Null values in cluster_id were filled with the string "unknown_cluster".

sales_bronze -> sales_silver

- Column Removal - _co: The _co column (repeated indexes) was removed.
- Numerical Column Processing:
 - sales, revenue, stock, and promo_discount_2 columns were converted to float (double) data types. Null values in these columns were imputed with 0.0.
 - For the price column, the median was used for null imputation, as a product price of 0.0 would not make sense.
- Categorical Column Processing:
 - trim was applied to categorical columns.
 - Null values in categorical columns were replaced with the string 'NA'.

`cities_bronze` -> `cities_silver`

- City Name Correction: Removed the "?" character from city names. Ex: ?zmir --> Izmir
- Column Removal - `city_old_id`: This column was removed as it appeared to be an old city ID. Since the granularity of this table is based on `store_id`, this old city ID was not relevant, as the city name depends on the `store_id`.
- Column Removal - `country_id`: This column was removed as all data pertains to the same country (Turkey).

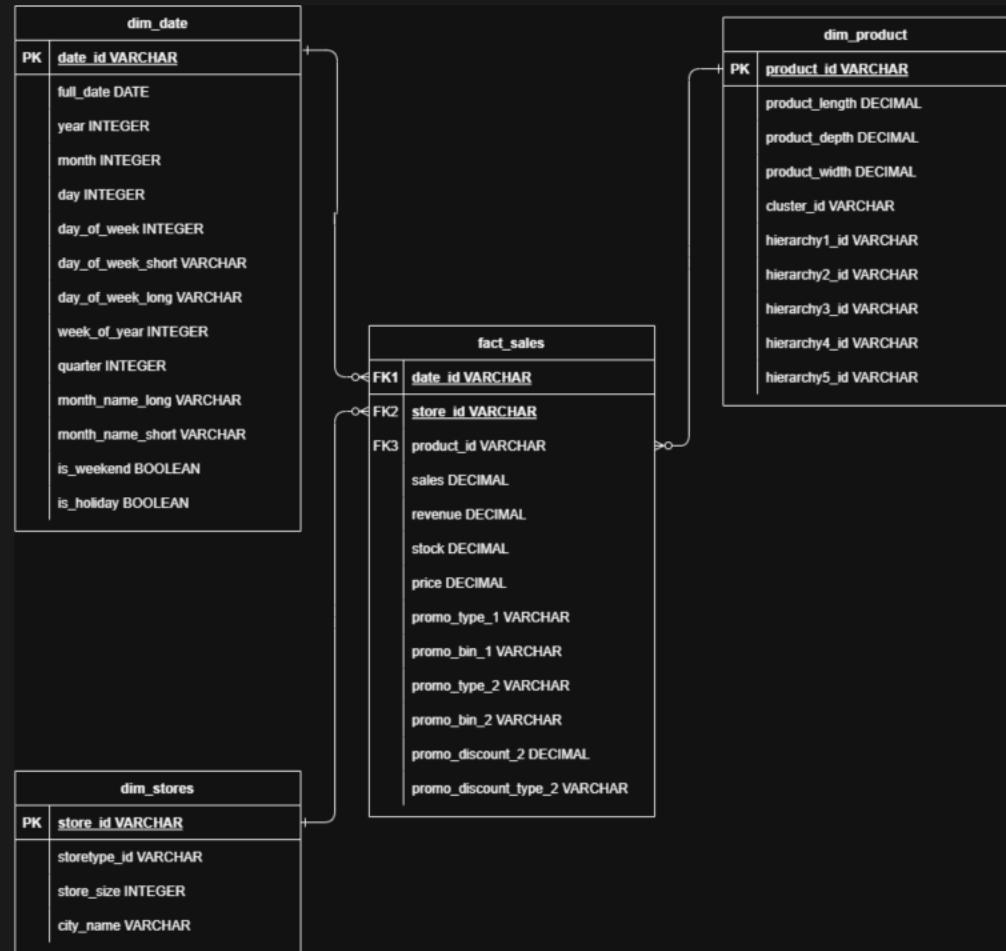
product_bronze -> product_silver

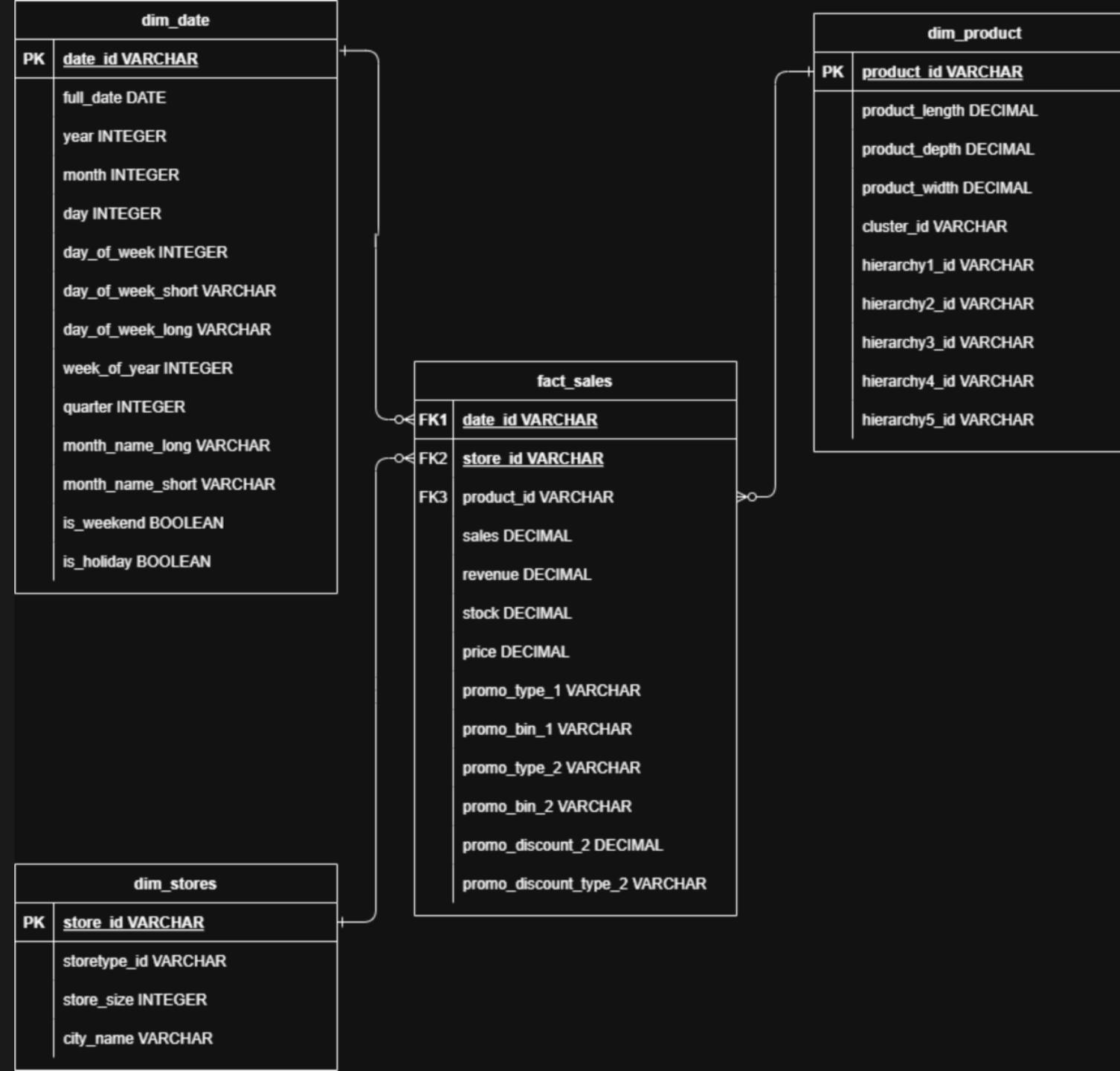
- Null Handling - Product Dimensions: Null values in `product_length`, `product_depth`, and `product_width` were filled with 0.0.
- Null Handling - `cluster_id`: Null values in `cluster_id` were filled with the string "unknown_cluster".

`sales_bronze -> sales_silver`

- Column Removal - `_co`: The `_co` column (repeated indexes) was removed.
- Numerical Column Processing:
 - `sales`, `revenue`, `stock`, and `promo_discount_2` columns were converted to float (double) data types. Null values in these columns were imputed with 0.0.
 - For the `price` column, the median was used for null imputation, as a product price of 0.0 would not make sense.
- Categorical Column Processing:
 - `trim` was applied to categorical columns.
 - Null values in categorical columns were replaced with the string "NA".

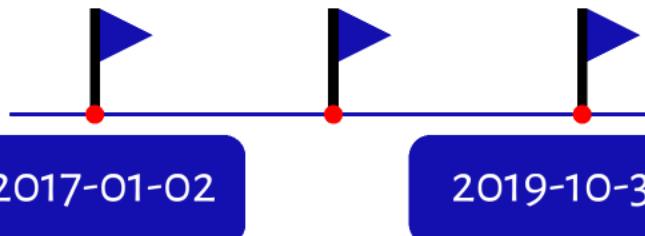
GOLD



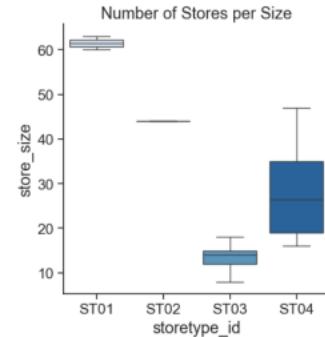


EDA: INITIAL INSIGHTS

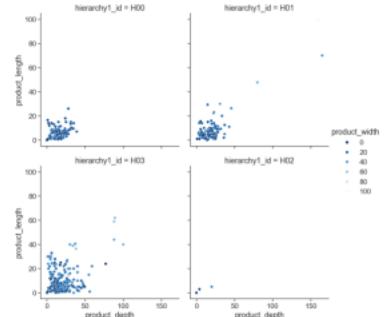
• TIMELINE



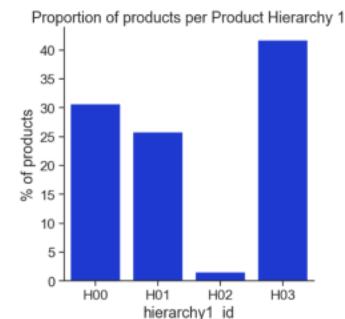
- 4 store types.
- ST01 and ST02: 2 and 1 stores, respectively, but larger store sizes.
- ST03 and ST04: 20 and 40 stores, respectively, but smaller in size.



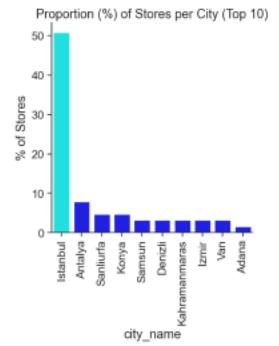
- Small products in general.
- No clear difference between categories in Hierarchy 1.
- Values of zero are due to imputation regarding missing values (NA) or when one of the feature is zero (fabric peace).



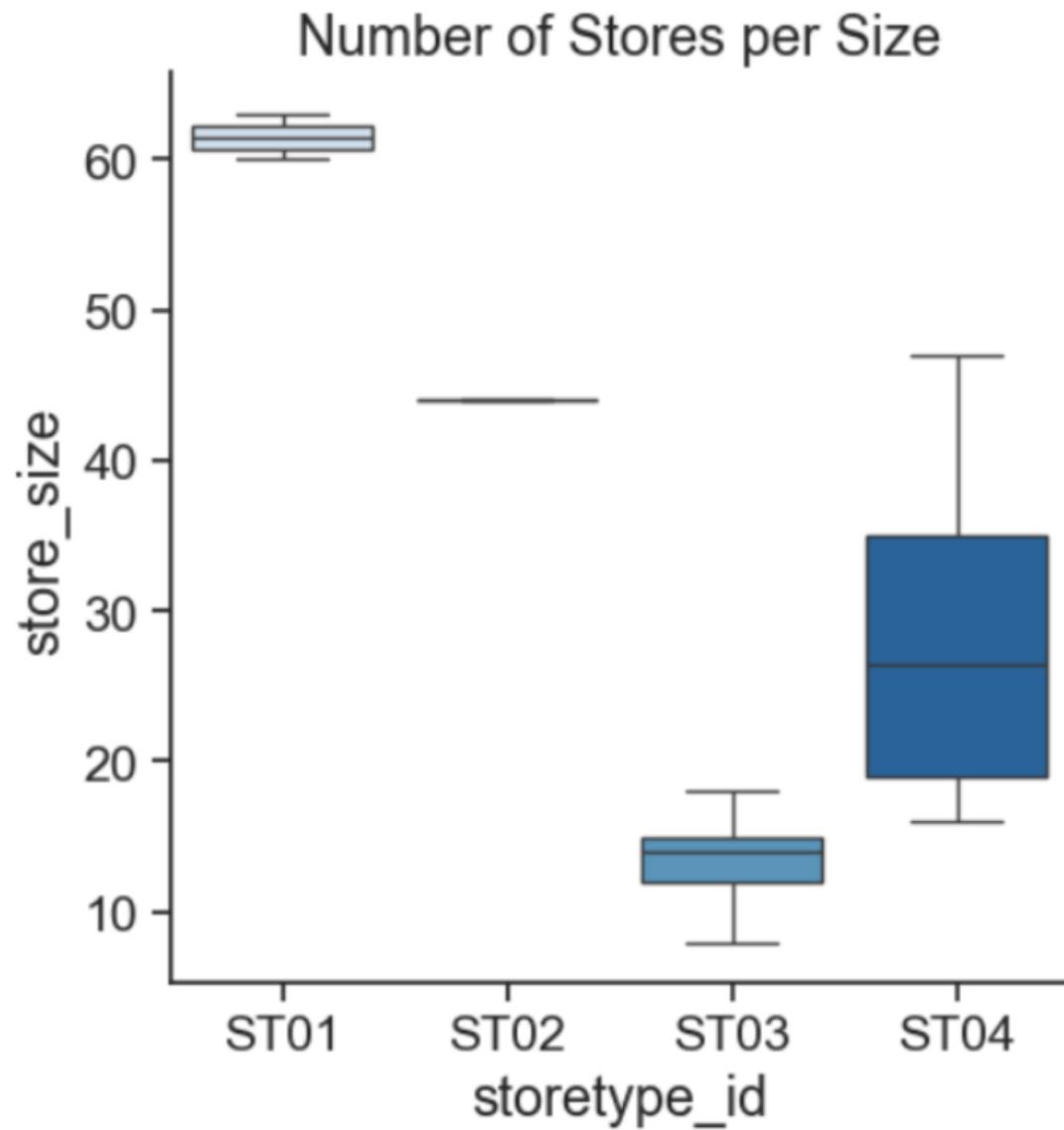
- 5 products hierarchy.
- Hierarchy1 has 4 categories.
- H02 represents only 1.6% of total products.



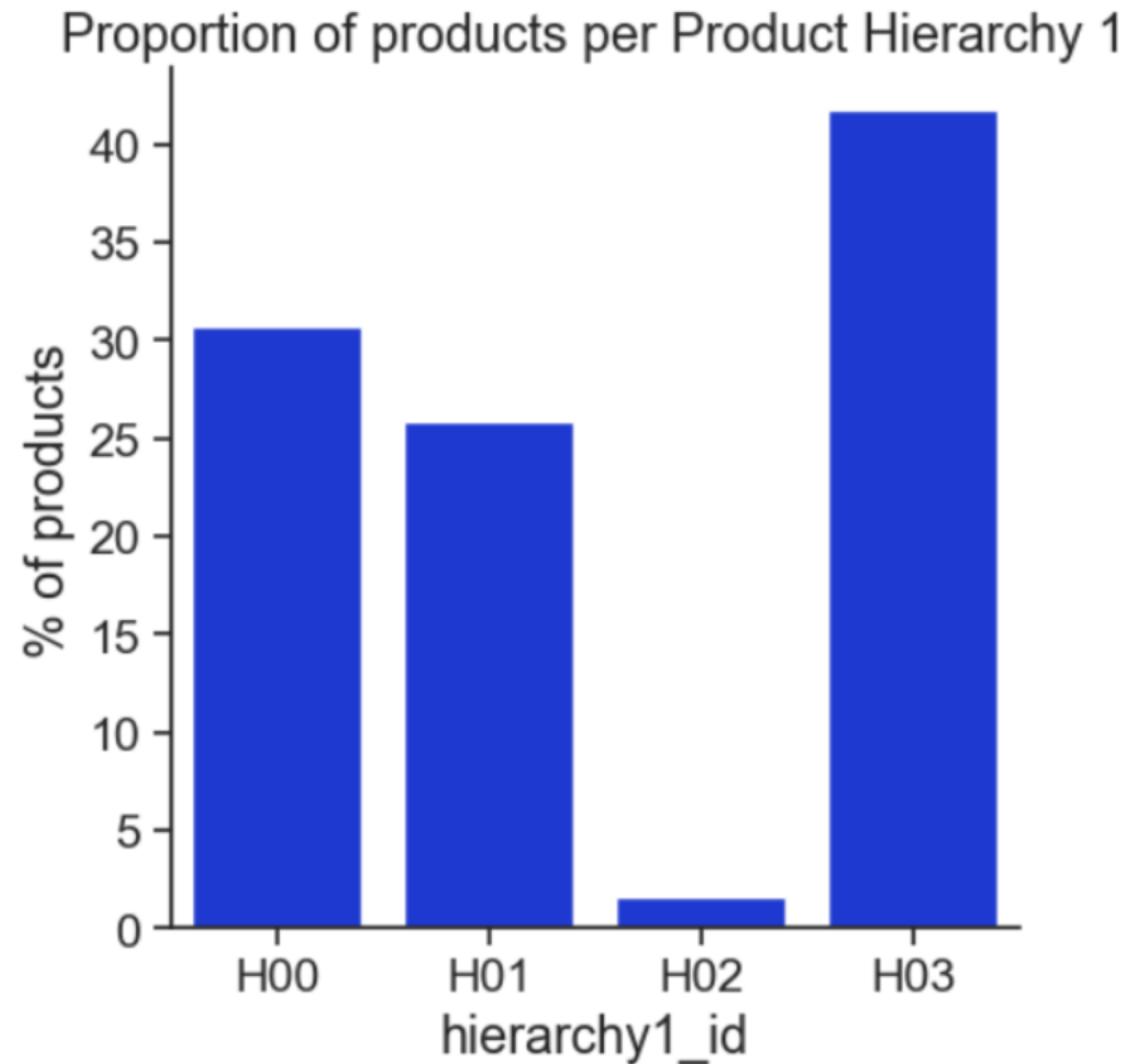
- 19 cities.
- Around 51% of the stores are in Istanbul.



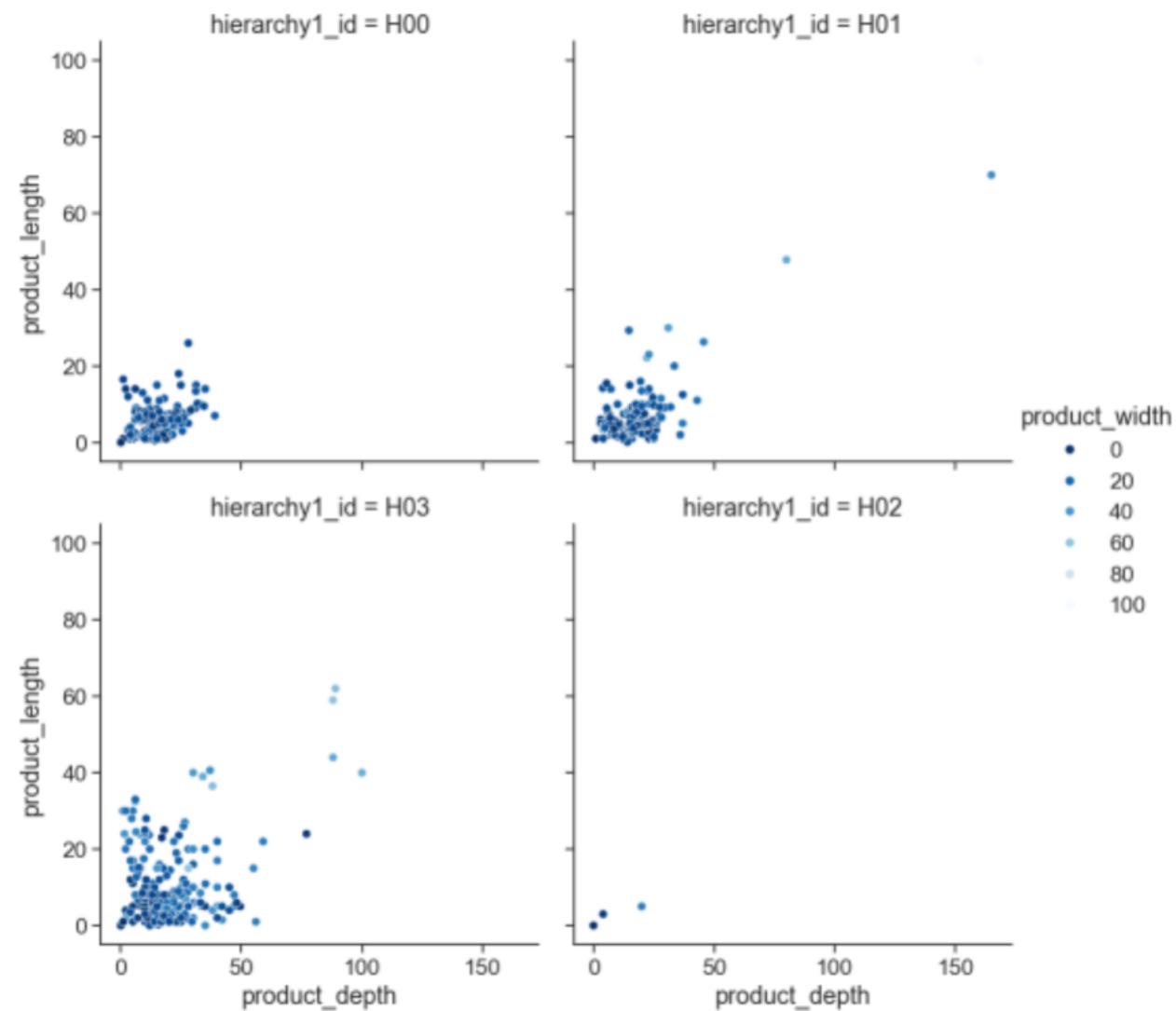
- 4 store types.
- ST01 and ST02: 2 and 1 stores, respectively, but larger store sizes.
- ST03 and ST04: 20 and 40 stores, respectively, but smaller in size.



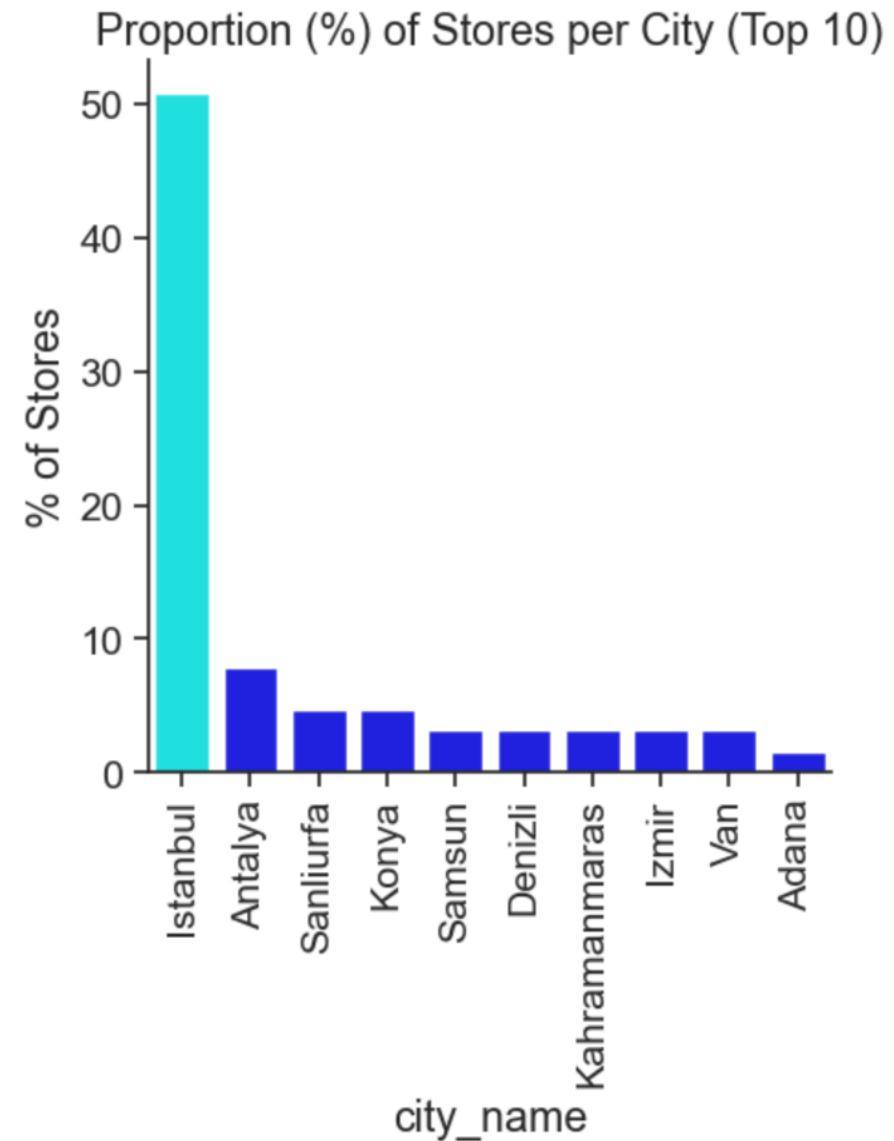
- 5 products hierarchy.
- Hierarchy 1 has 4 categories.
- H02 represents only 1.6% of total products.



- Small products in general.
- No clear difference between categories in Hierarchy 1.
- Values of zero are due to imputation regarding missing values (NA) or when one of the feature is zero (fabric peace).

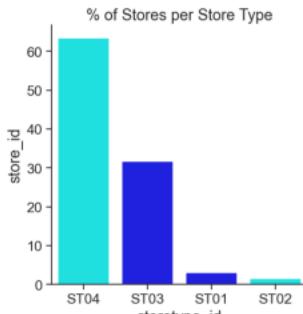


- 19 cities.
- Around 51% of the stores are in Istanbul.

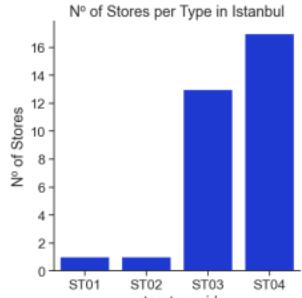


EDA: PROOF OF CONCEPT

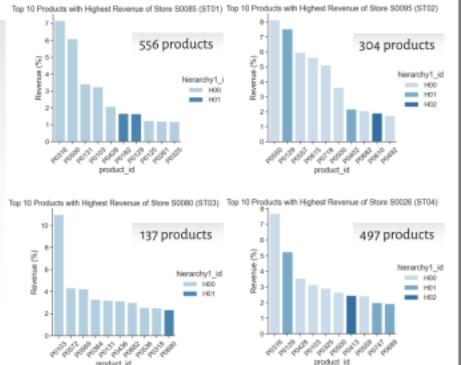
- ST04 has the highest number of stores
- ST02 (and ST01) have a significant low number of stores.



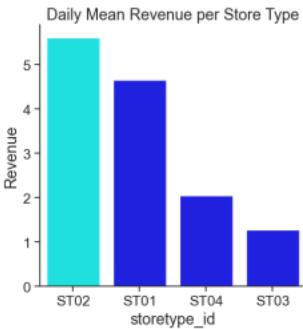
- Bursa has the highest revenue and sales mean.
- However, Bursa has only 1 store.
- Istanbul has the second highest revenue and sales mean.
- Istanbul has all type of stores.



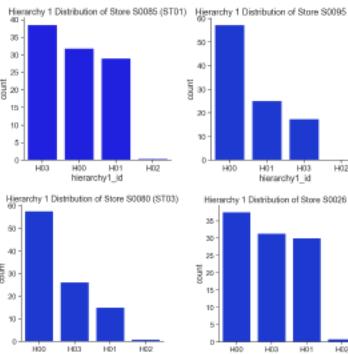
- The Top 10 products in each store represents between 29% and 44% of total Revenue.
- Majority of products are Hoo, a few Ho1 and Ho2, and no Ho3 products.



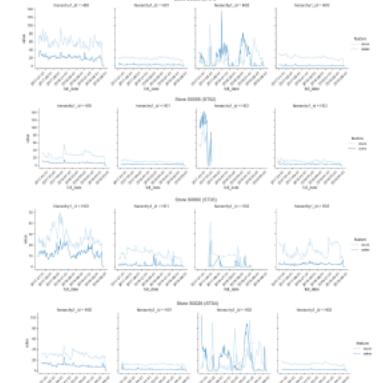
- ST02 (and ST01) stores represent the highest revenue mean per store.



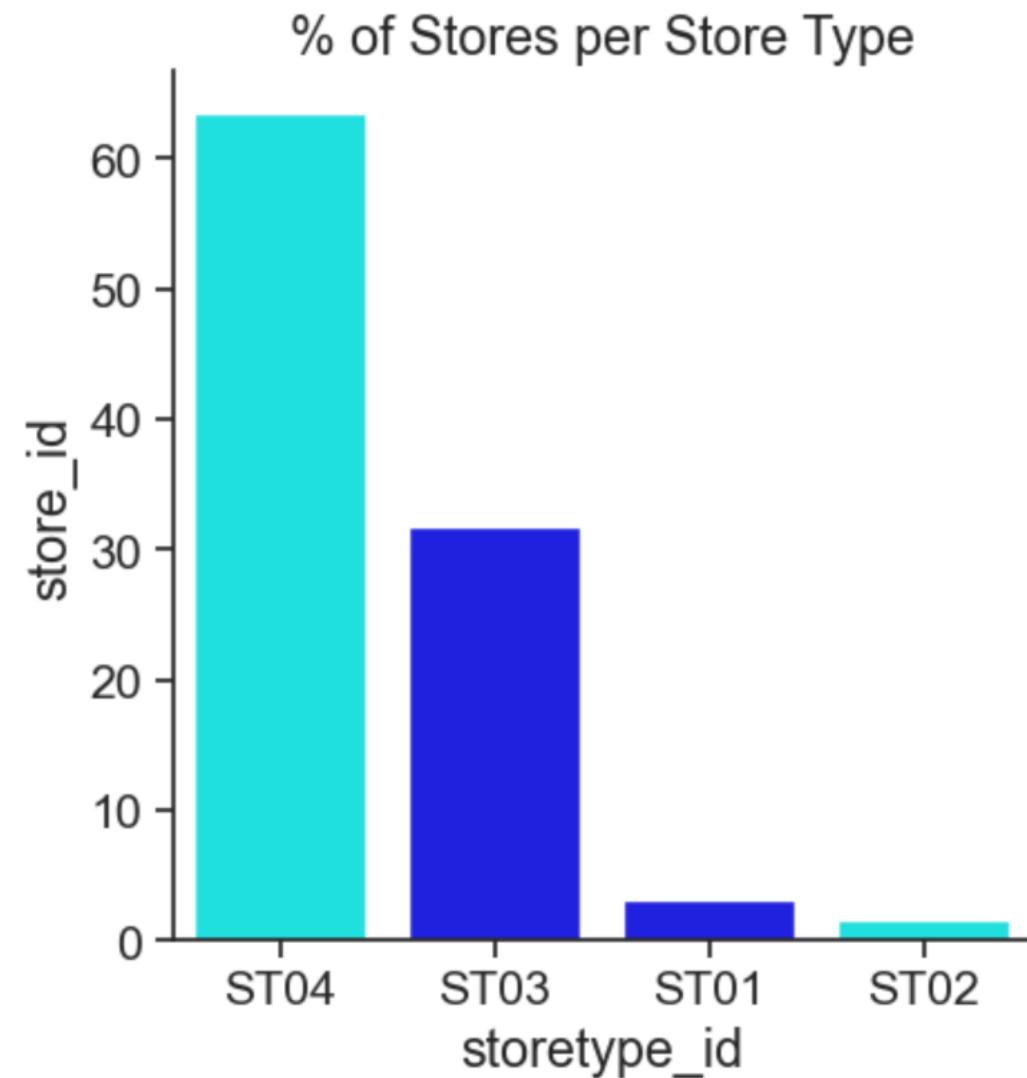
- PoC Scope:
 - 1 store per type in Istanbul.
 - Select the stores with the highest revenue and sales.
 - Hierarchy1 division.



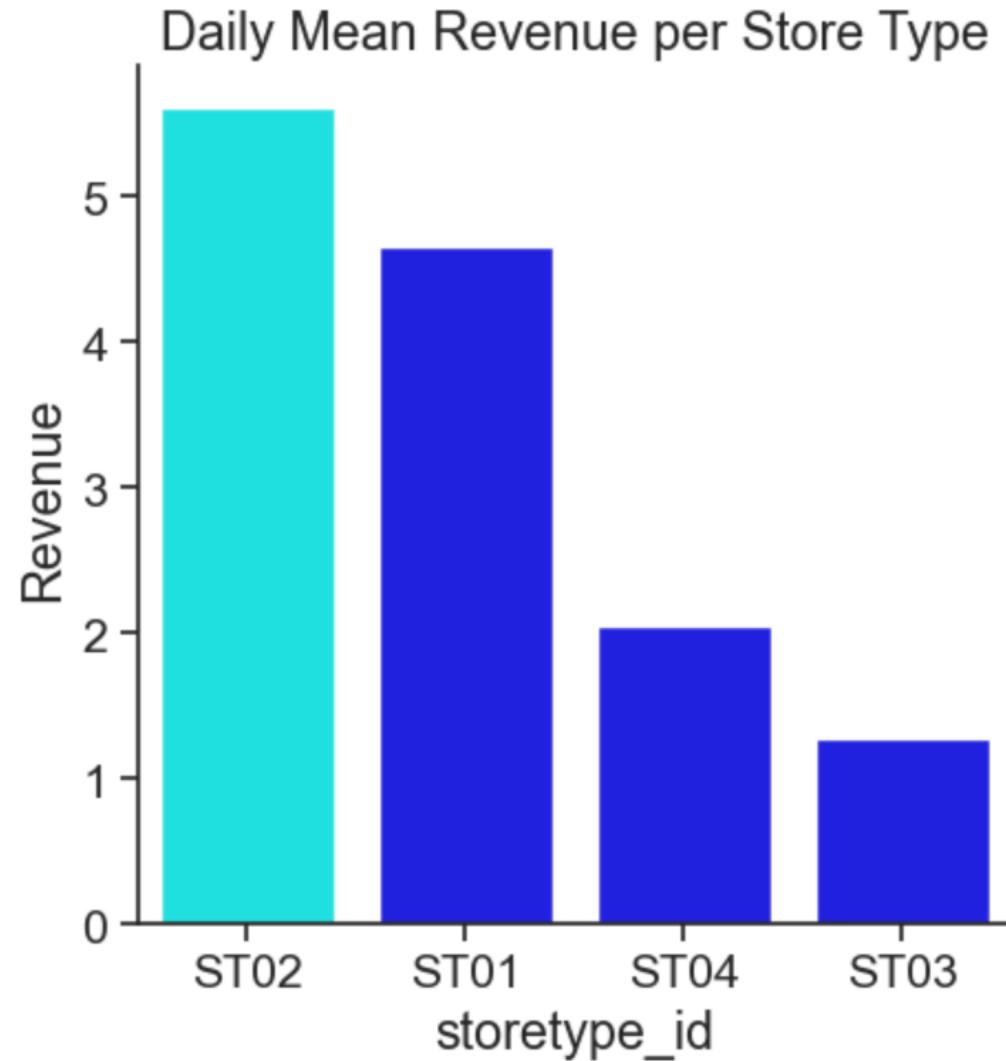
- Clear higher amount of stock than sales in the end of each week, except for Ho2.
- Min-max inventory analysis.



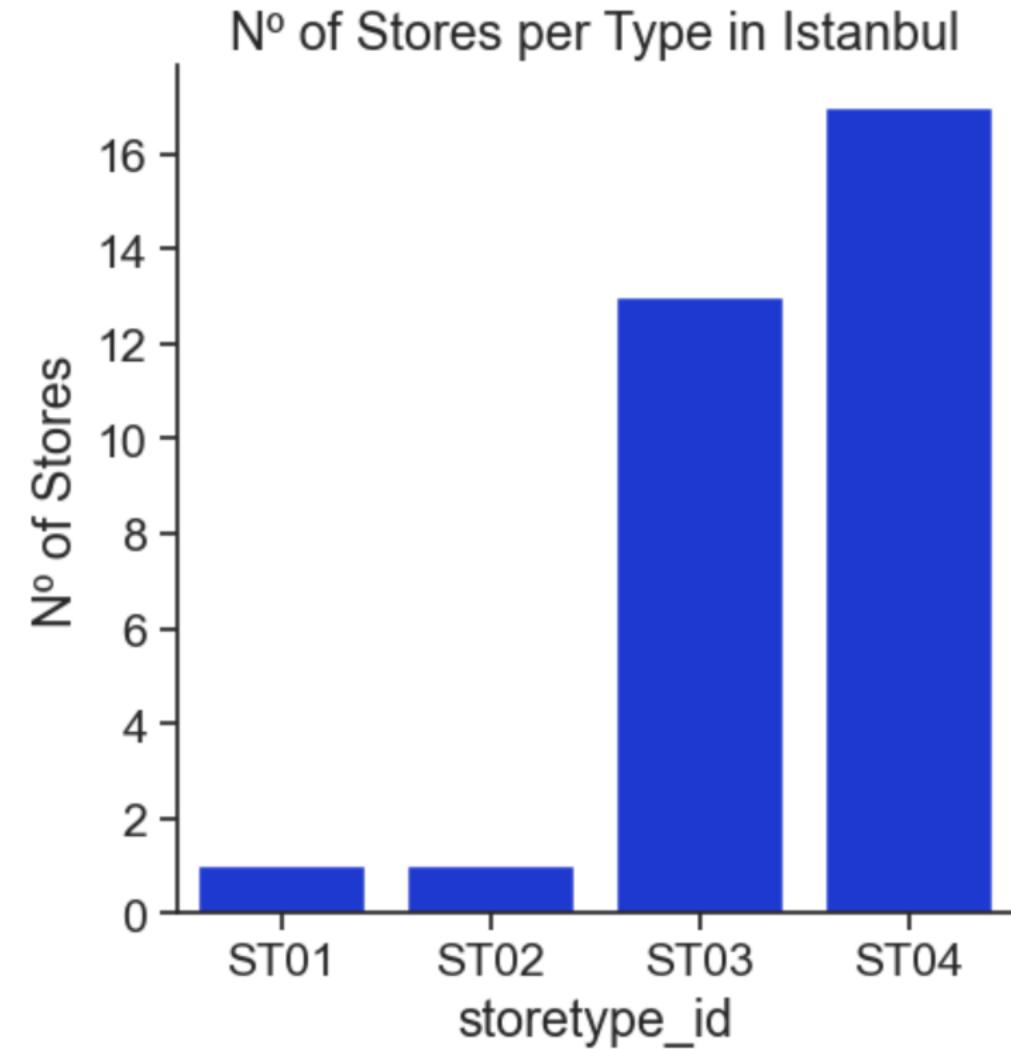
- ST04 has the highest number of stores
- ST02 (and ST01) have a significant low number of stores.



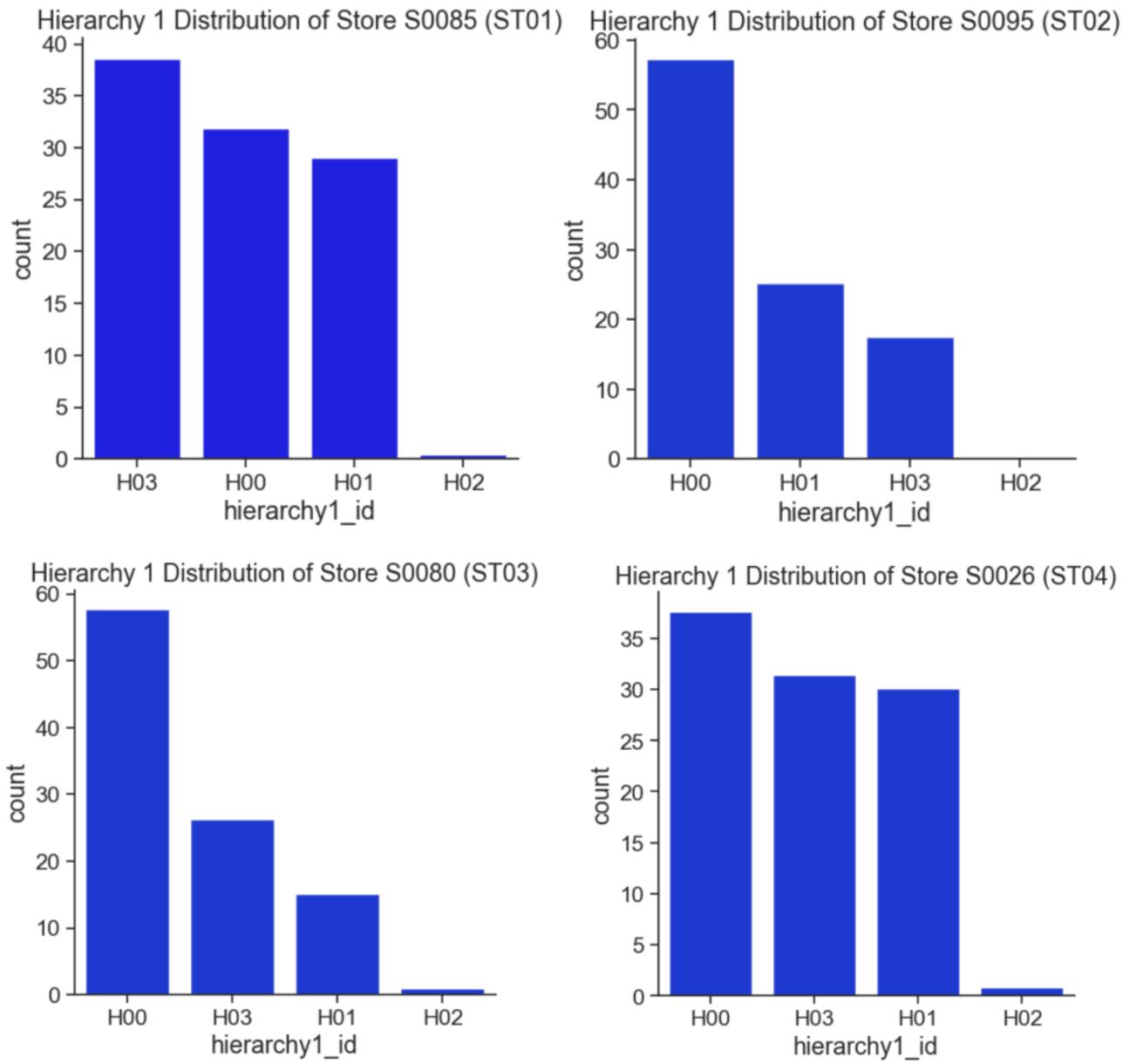
- ST02 (and ST01) stores represent the highest revenue mean per store.



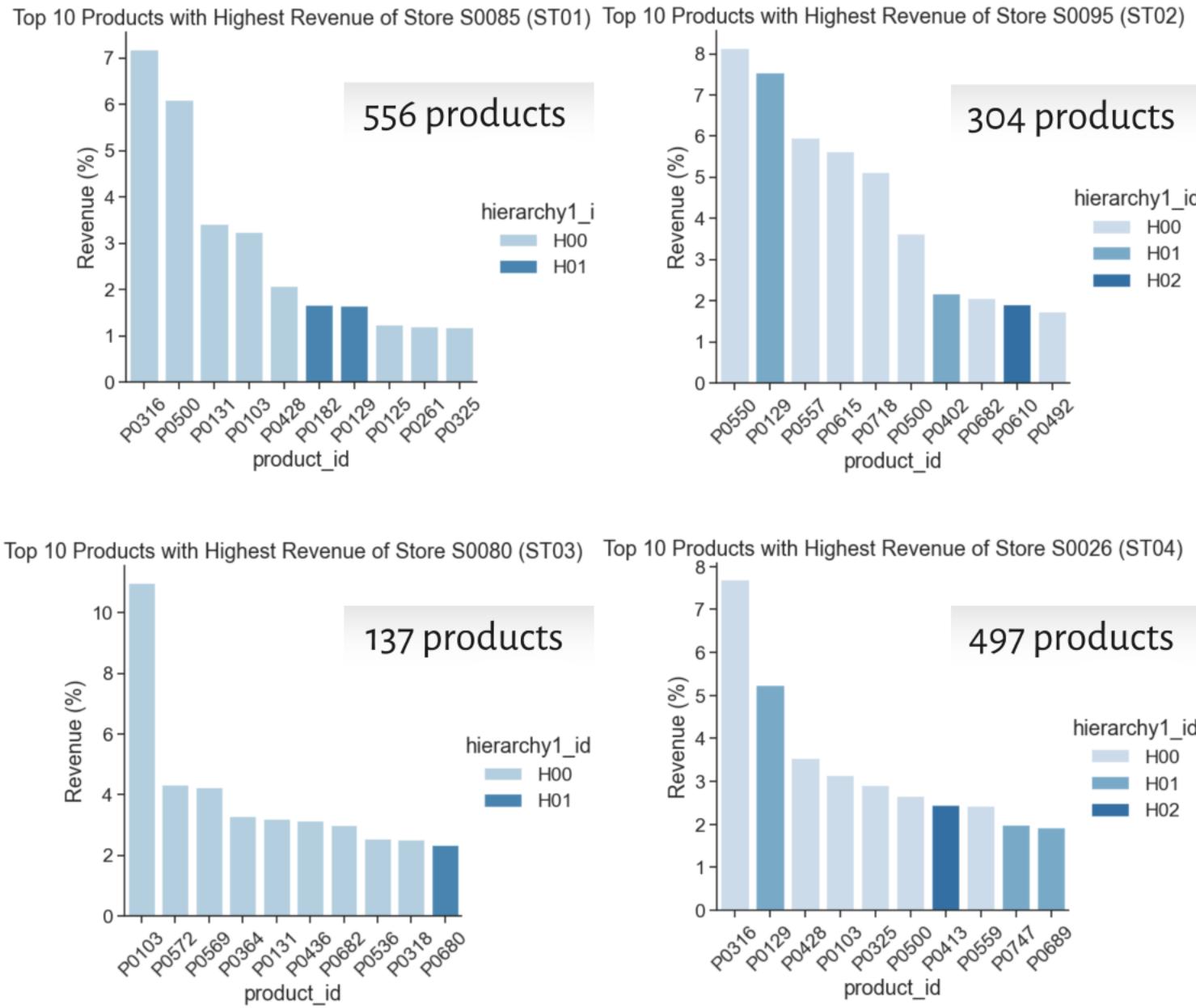
- Bursa has the highest revenue and sales mean.
- However, Bursa has only 1 store.
- Istanbul has the second highest revenue and sales mean.
- Istanbul has all type of stores.



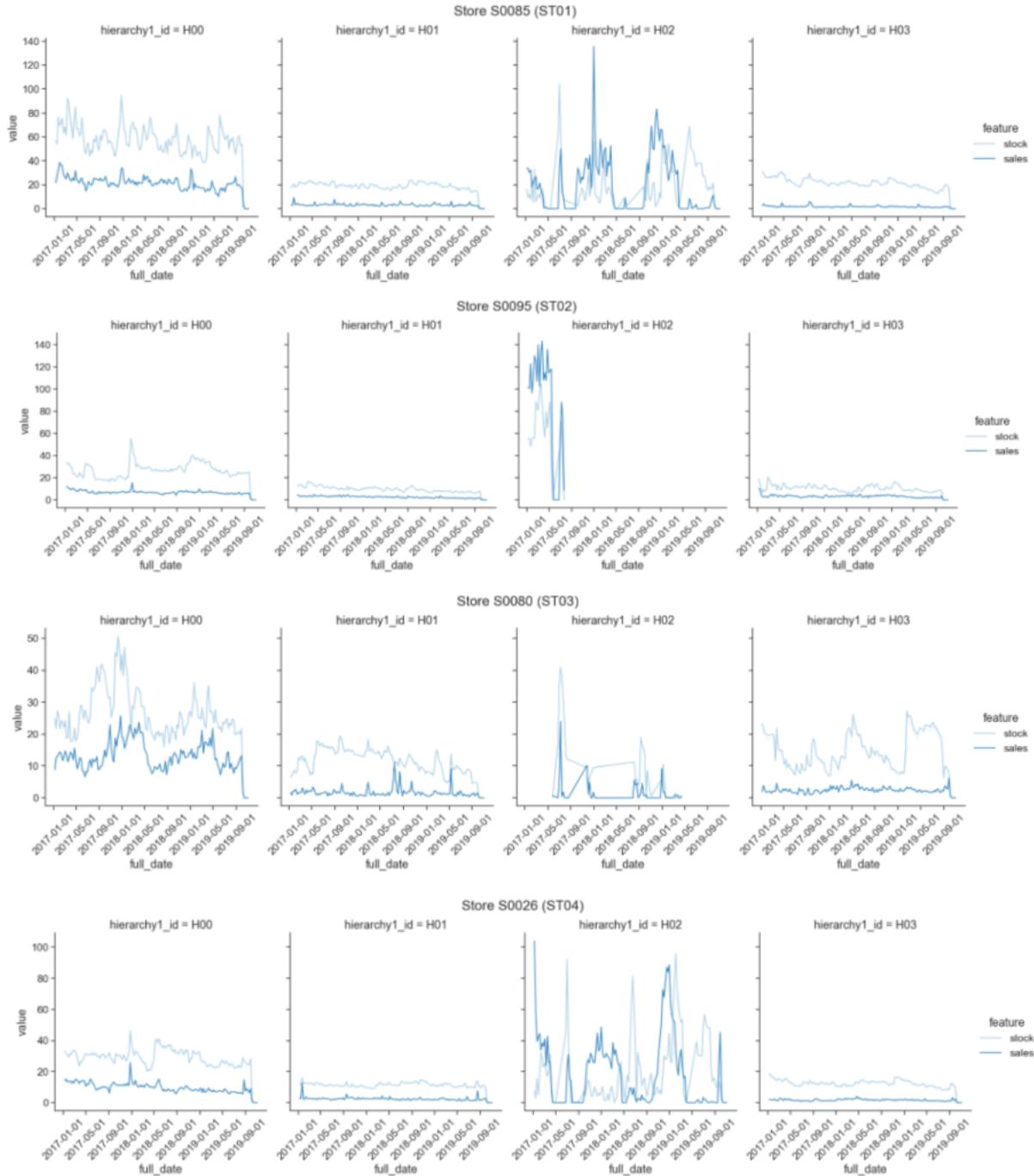
- PoC Scope:
 - 1 store per type in Istanbul.
 - Select the stores with the highest revenue and sales.
 - Hierarchy 1 division.



- The Top 10 products in each store represents between 29% and 44% of total Revenue.
- Majority of products are H00, a few H01 and H02, and no H03 products.

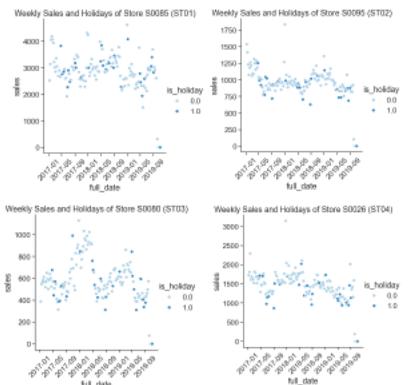


- Clear higher amount of stock than sales in the end of each week, except for H02.
- Min-max inventory analysis.

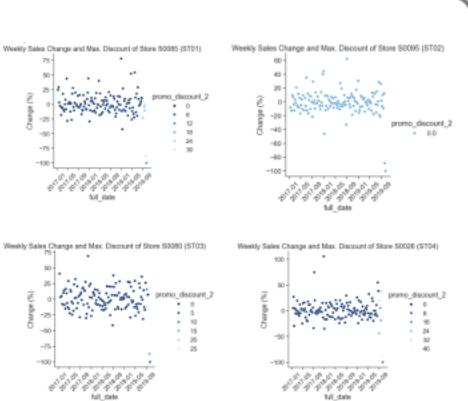


EDA: PROOF OF CONCEPT

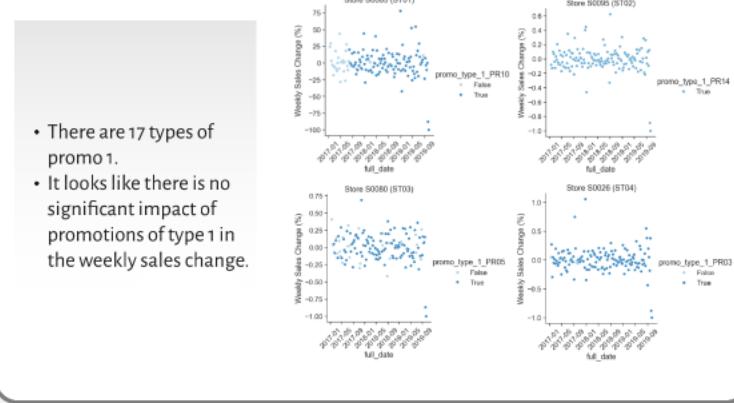
- It looks like there is no difference on sales between weeks with and without holidays.



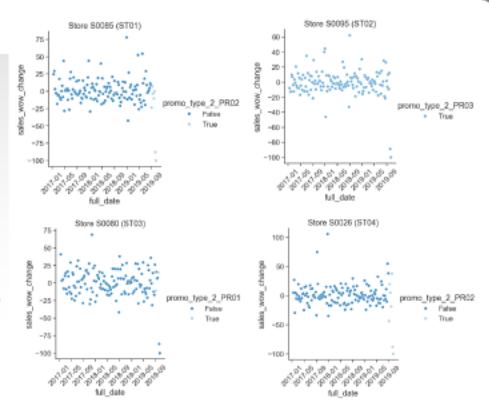
- It looks like the `promo_discount_2` was only applied at the end of 2019.



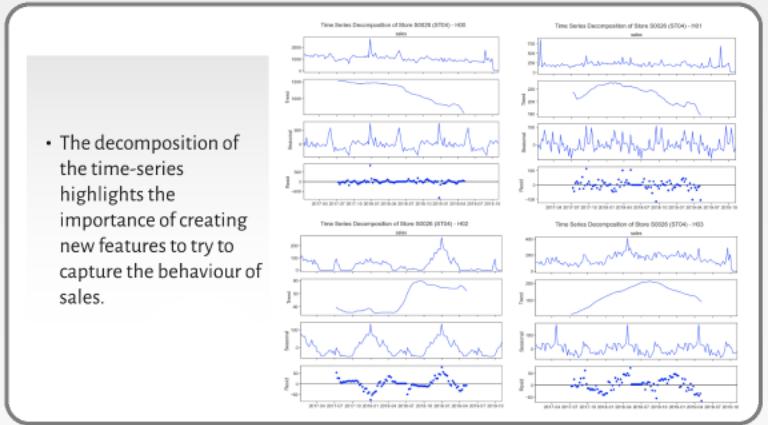
- There are 17 types of promo 1.
- It looks like there is no significant impact of promotions of type 1 in the weekly sales change.



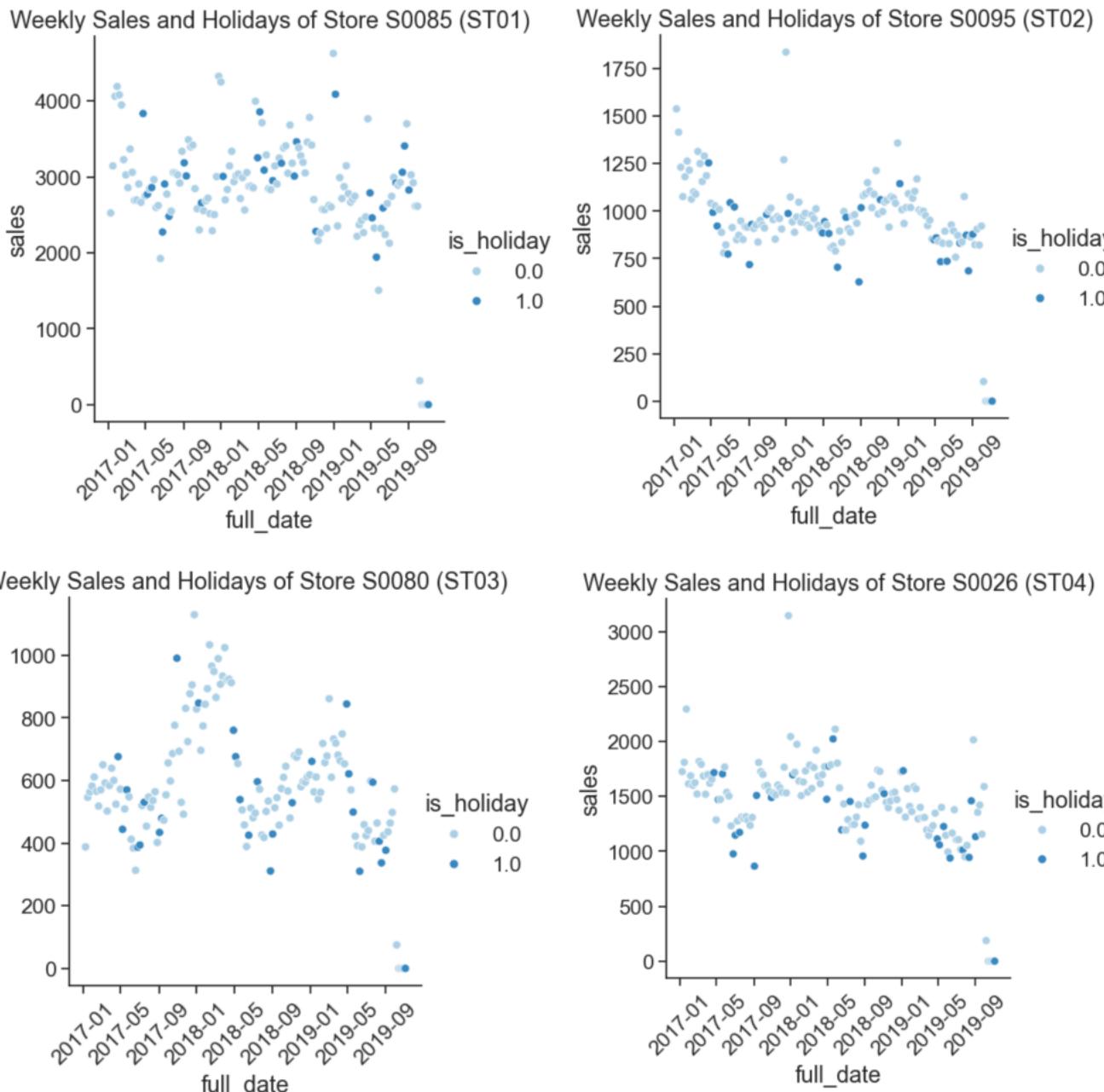
- There are 4 types of promo 2.
- It looks like there is no significant impact of promotions of type 2 in the weekly sales change.



- The decomposition of the time-series highlights the importance of creating new features to try to capture the behaviour of sales.

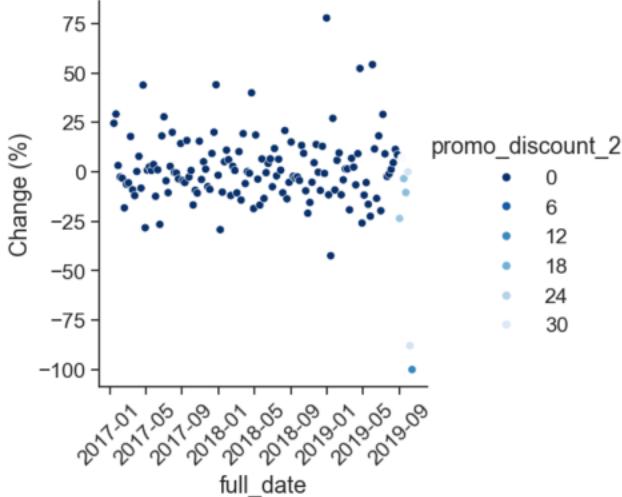


- It looks like there is no difference on sales between weeks with and without holidays.

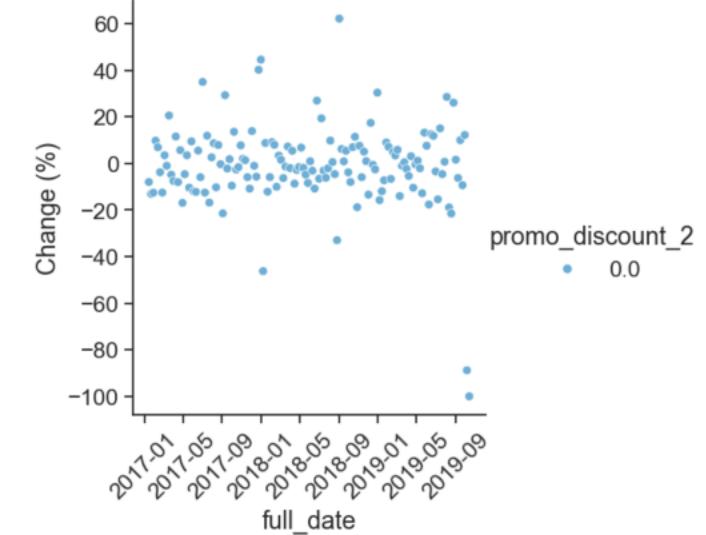


- It looks like the `promo_discount_2` was only applied at the end of 2019.

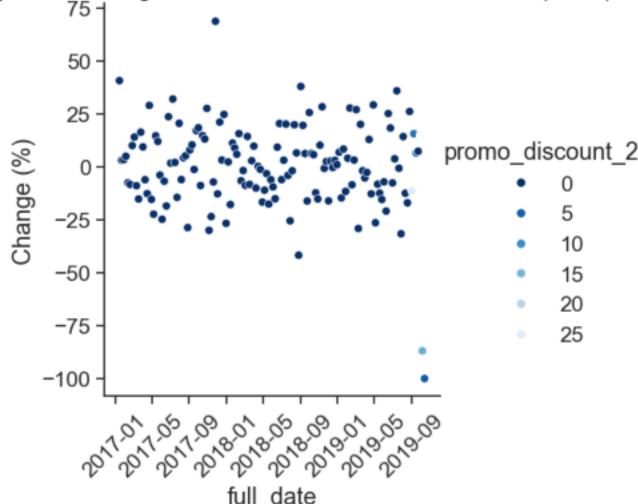
Weekly Sales Change and Max. Discount of Store S0085 (ST01)



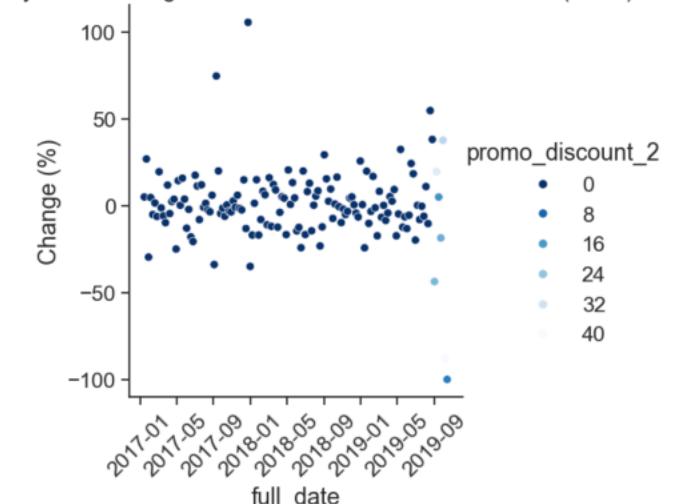
Weekly Sales Change and Max. Discount of Store S0095 (ST02)



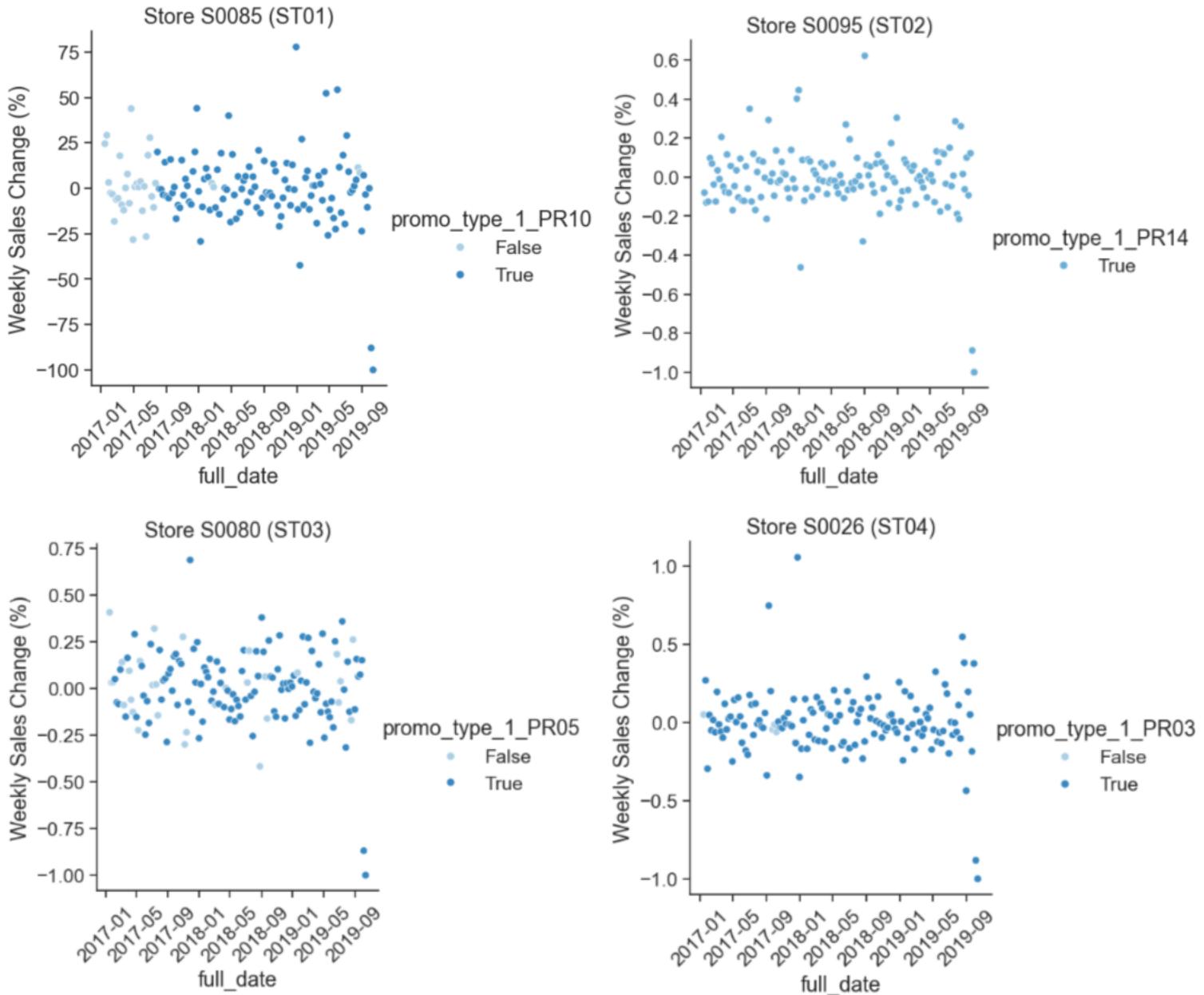
Weekly Sales Change and Max. Discount of Store S0080 (ST03)



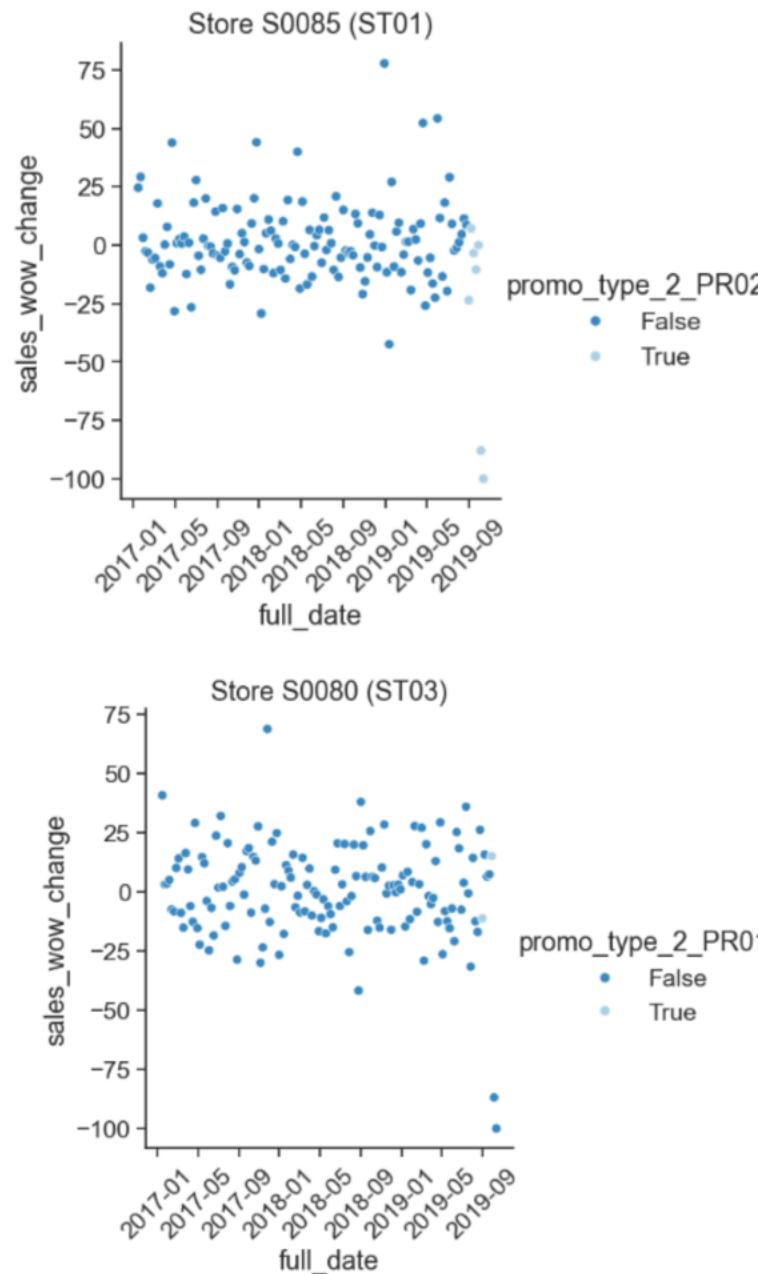
Weekly Sales Change and Max. Discount of Store S0026 (ST04)



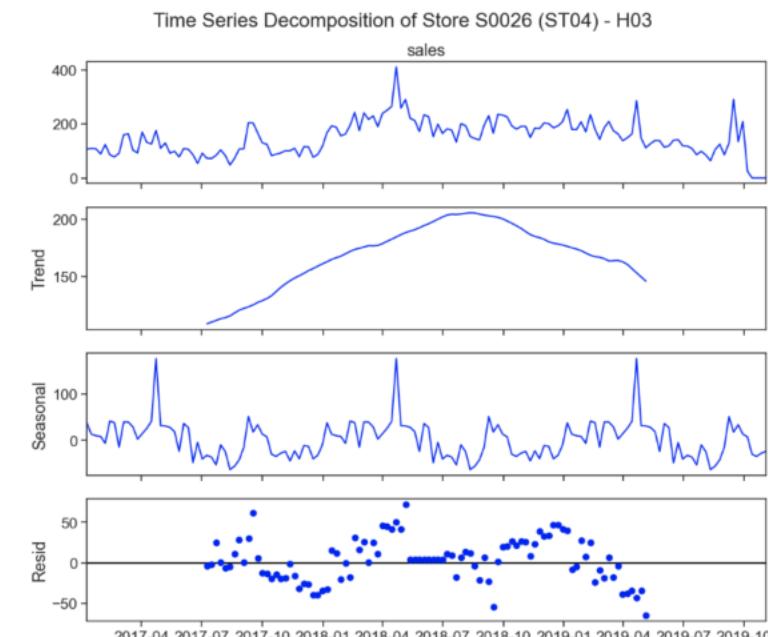
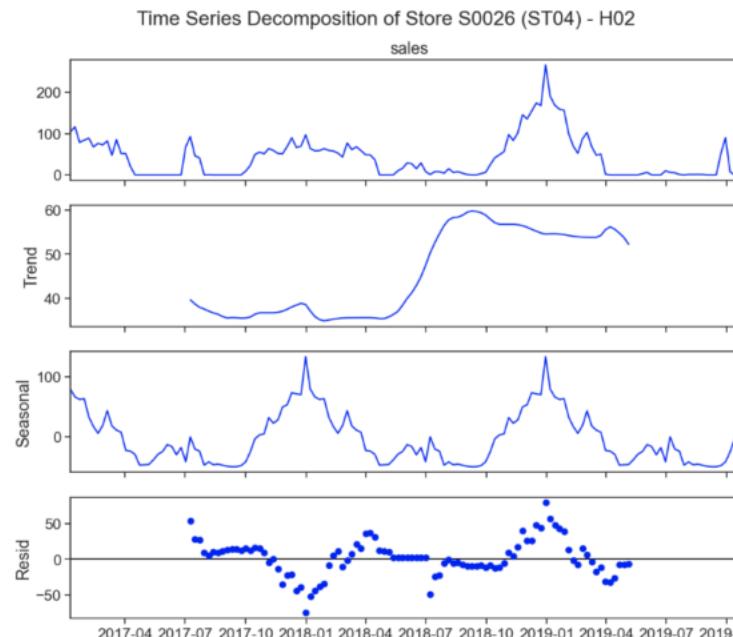
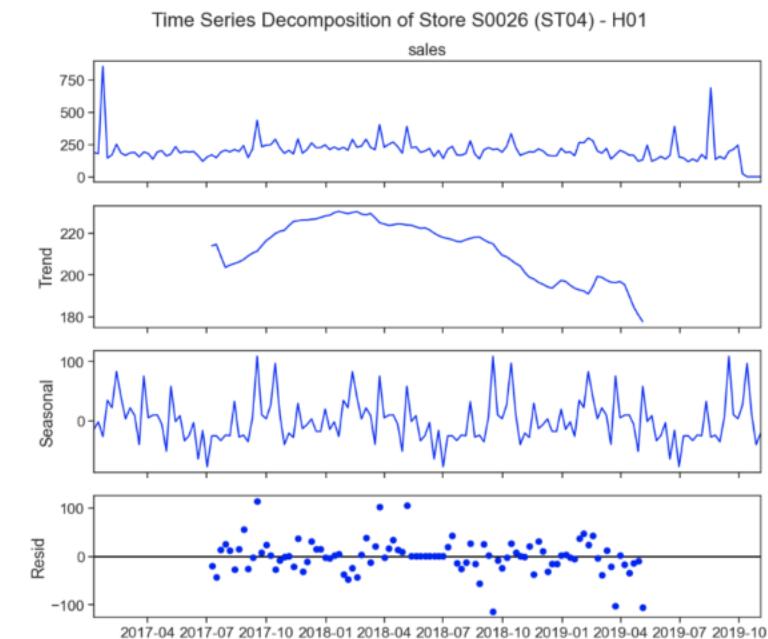
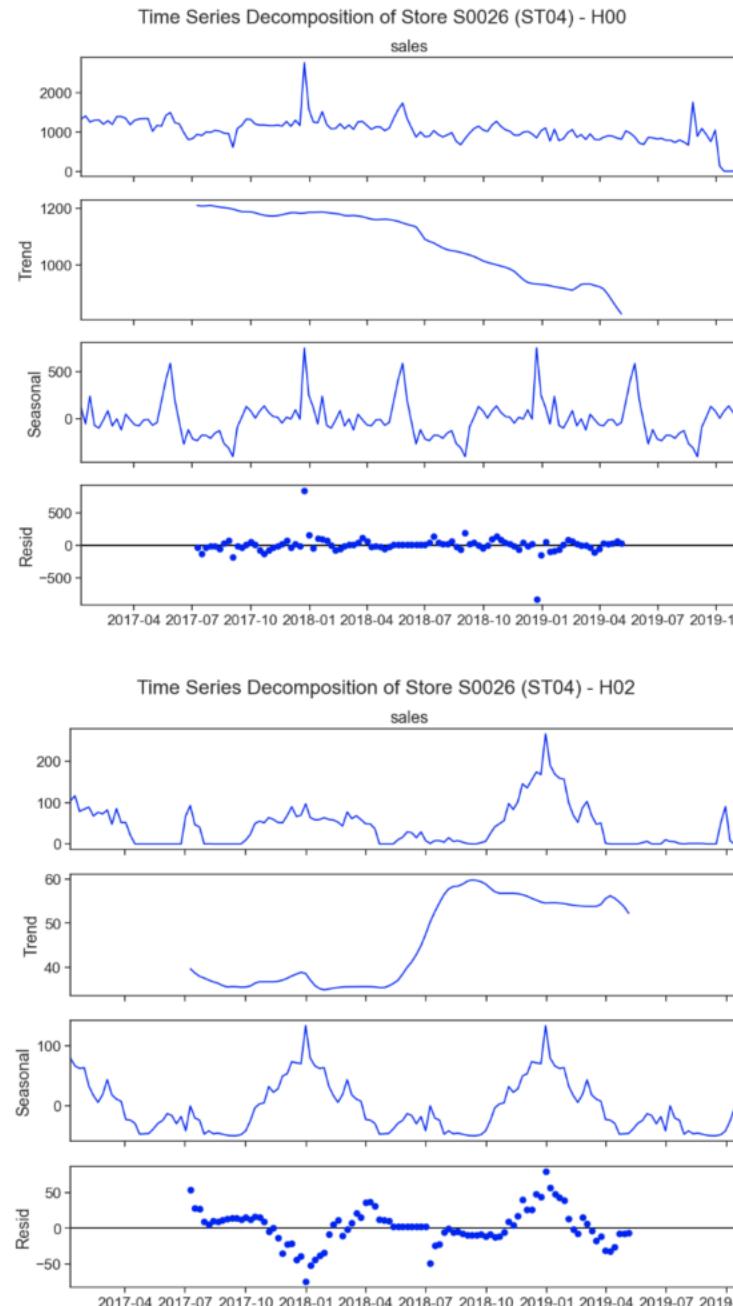
- There are 17 types of promo 1.
- It looks like there is no significant impact of promotions of type 1 in the weekly sales change.



- There are 4 types of promo 2.
- It looks like there is no significant impact of promotions of type 2 in the weekly sales change.

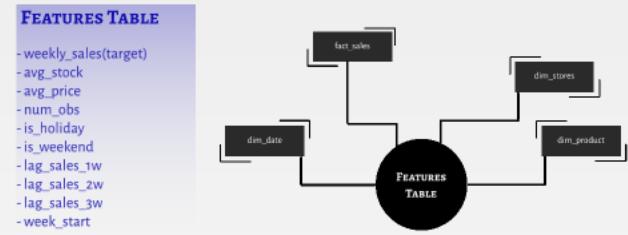


- The decomposition of the time-series highlights the importance of creating new features to try to capture the behaviour of sales.



FEATURES TABLE

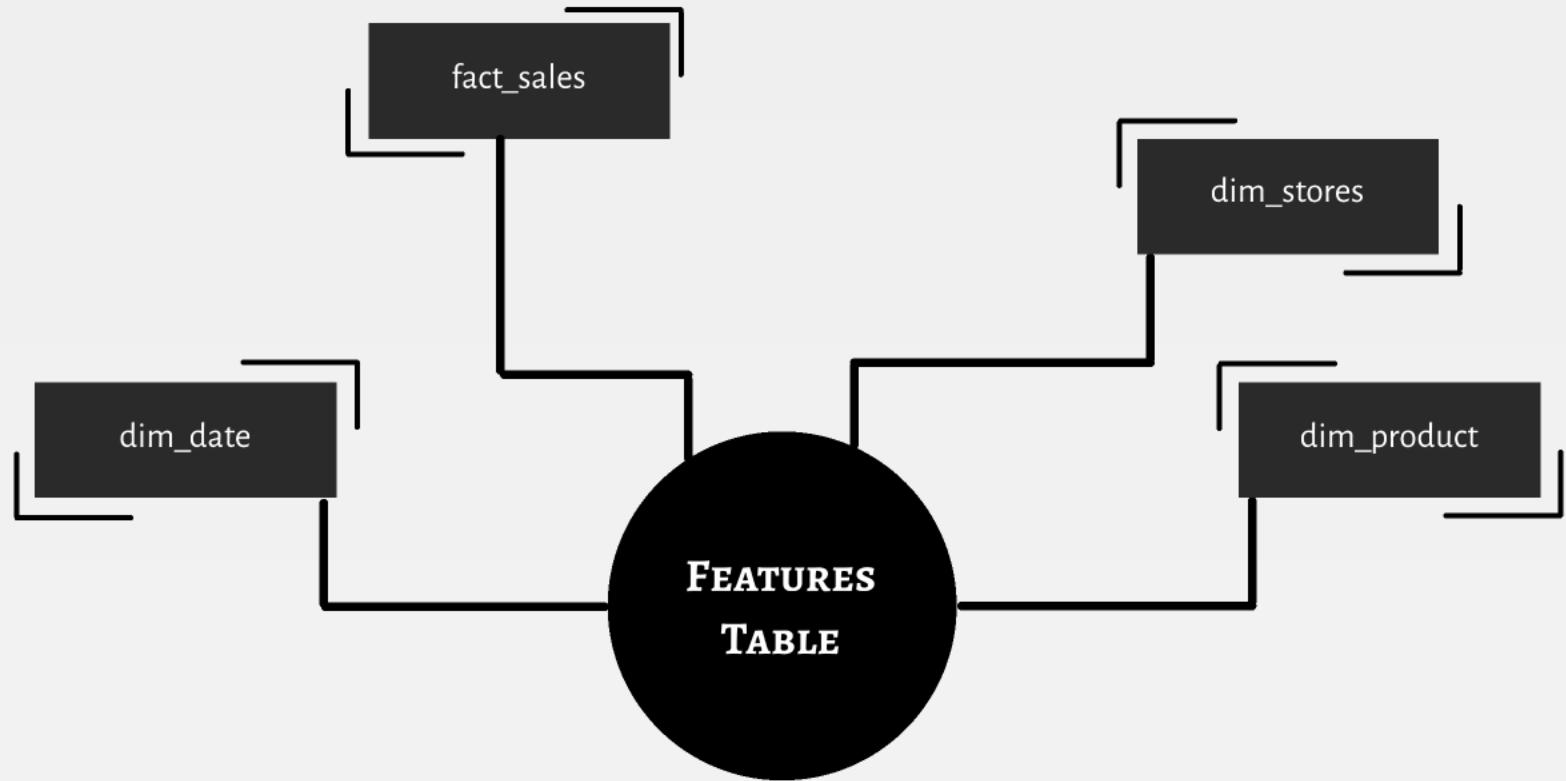
FEATURES CREATION



FEATURES CREATION

FEATURES TABLE

- weekly_sales(target)
- avg_stock
- avg_price
- num_obs
- is_holiday
- is_weekend
- lag_sales_1w
- lag_sales_2w
- lag_sales_3w
- week_start



Linear Regression

PROOF OF CONCEPT

PROOF OF CONCEPT

Stores "Soo85", "Soo95", "Soo80", "Soo26"

Categories H00, H01, H02 and H03

Proof of Concept Process

- 1- Filter the Store and Product Category
- 2- Do train-test split according to dates
- 3- Regression with target "weekly sales" and features "other variables"
- 4- Check model metrics (R² and MAE)

We didn't think it was enough

We used Lasso to identify the most important variables in the model.

S0085	H00	"avg_stock", "lag_sales_1w", "lag_sales_3w",
S0085	H01	"avg_stock"
S0085	H02	"week", "avg_stock", "lag_sales_1w", "lag_sales_3w"
S0085	H03	"avg_stock", "avg_price"
S0095	H00	"avg_stock", "avg_price", "lag_sales_1w"
S0095	H01	"avg_stock"
S0095	H02	"avg_stock"
S0095	H03	"avg_price" "lag_sales_1w", "lag_sales_3w",
S0080	H00	"avg_stock", "lag_sales_1w", "lag_sales_3w",
S0080	H02	"Week"
S0080	H03	"avg_stock", "avg_price", "product_width"
S0026	H00	"avg_stock", "lag_sales_1w", "lag_sales_2w", "lag_sales_3w"
S0026	H01	"avg_stock"
S0026	H02	"week", "avg_stock", "lag_sales_1w", "lag_sales_2w", "lag_sales_3w"
S0026	H03	"lag_sales_1w", "lag_sales_2w", "lag_sales_3w"

"avg_stock",	12 em 16
"avg_price"	4 em 16
"lag_sales_1w",	8 em 16
"lag_sales_2w",	3 em 16
"lag_sales_3w",	7 em 16
"week"	3 em 16

S0085	H00		0.9935	R ² : 0.7956		1.0035	R ² on test set: 0.7956
S0085	H01		0.3704	R ² : 0.1475		0.3293	R ² on test set: 0.1475
S0085	H02		2.2462	R ² : -1.8740		1.8308	R ² on test set: -1.8740
S0085	H03		0.1644	R ² : 0.7182		0.1894	R ² on test set: 0.7182
S0095	H00		0.4019	R ² : 0.7956		0.231	R ² on test set: 0.7956
S0095	H01		0.2804	R ² : 0.2822		0.1772	R ² on test set: 0.2822
S0095	H02		8.6106	R ² : -5.5166		8.2994	R ² on test set: -5.5166
S0095	H03		0.2278	R ² : 0.8196		0.231	R ² on test set: 0.8196
S0080	H00		0.7543	R ² : 0.6879		0.7817	R ² on test set: 0.6879
S0080	H01		0.2433	R ² : -0.4934		0.2659	R ² on test set: -0.4934
S0080	H02		0.2926	R ² : -0.0829		0.4684	R ² on test set: -0.0829
S0080	H03		0.3553	R ² : 0.6812		0.4193	R ² on test set: 0.6812
S0026	H00		0.5114	R ² : 0.1743		0.5154	R ² on test set: 0.1743
S0026	H01		0.3358	R ² : 0.1110		0.2914	R ² on test set: 0.1110
S0026	H02		2.3408	R ² : -0.2775		1.7344	R ² on test set: -0.2775
S0026	H03		0.183	0.8436		0.1896	R ² on test set: 0.8436

Train R²: 0.11915454802471326
Train MAE: 0.4351829771075341

R ² : 0.7956	1.0035 R ² on test set: 0.7956
R ² : 0.1475	0.3293 R ² on test set: 0.1475
R ² : -1.8740	1.8308 R ² on test set: -1.8740
R ² : 0.7182	0.1894 R ² on test set: 0.7182
R ² : 0.7956	0.231 R ² on test set: 0.7956
R ² : 0.2822	0.1772 R ² on test set: 0.2822
R ² : -5.5166	8.2994 R ² on test set: -5.5166
R ² : 0.8196	0.231 R ² on test set: 0.8196
R ² : 0.6879	0.7817 R ² on test set: 0.6879
R ² : -0.4934	0.2659 R ² on test set: -0.4934
R ² : -0.0829	0.4684 R ² on test set: -0.0829
R ² : 0.6812	0.4193 R ² on test set: 0.6812
R ² : 0.1743	0.5154 R ² on test set: 0.1743
R ² : 0.1110	0.2914 R ² on test set: 0.1110
R ² : -0.2775	1.7344 R ² on test set: -0.2775
0.8436	0.1896 R ² on test set: 0.8436

Regressão Linear (3 Features) - Coeficientes:

beta1: 0.0147

beta2: -0.0017

beta3: 0.2449

beta4: 0.0099

beta5: 0.1876

beta0 (intercepto): 0.0727

 Avaliação do Modelo:

MAE: 0.3022

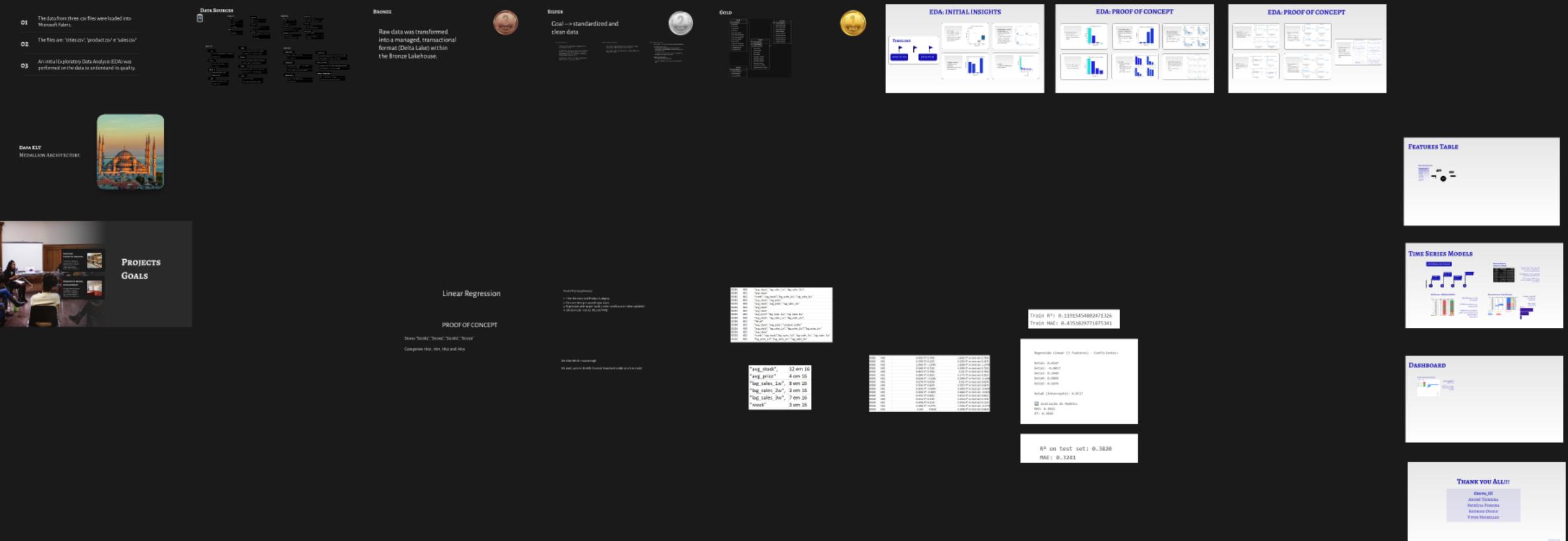
R²: 0.3820

R² on test set: 0.3820

MAE: 0.3241

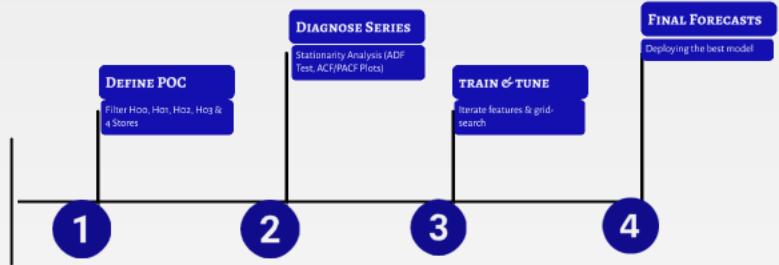
PREDICTIVE MODEL FOR INVENTORY MANAGEMENT

Sales Analysis and Forecasting in Department Stores in Turkey



TIME SERIES MODELS

4-STEP MODELLING PIPELINE



P.O.C RESULTS - SARIMA/ SARIMAX

		MAE VALUES			
Hierarchy	Store	SARIMA Train	SARIMA Test	SARIMAX Train	SARIMAX Test
H00	S0095	14.47	62.3	14.9	12.14
	S0085	53.47	123.76	44.16	34.73
	S0026	20.79	41.01	20.88	9.82
	S0080	12.63	81.46	12.95	70.26
H01	S0095	3.73	6.61	3.65	3.81
	S0085	11.51	32.12	10.52	18.74
	S0026	7.63	29.72	7.83	7.72
	S0080	1.49	1.86	1.54	0.29
H02	S0095	3.79	5.21	3.79	5.21
	S0085	2.91	1.6	2.78	1.12
	S0026	4.55	16.27	68.82	18.27
	S0080	0.55	0.87	0.55	0.87
H03	S0095	4.12	9.97	3.52	5.02
	S0085	9.1	18.11	9.04	3.81
	S0026	7.15	12.72	7.67	8
	S0080	2.57	11.72	2.59	11.29

Legend: Not Overfitted (Green), Light Overfitting (Yellow), Overfitting (Red)

- SARIMAX wins 15 of 16 cases (94%)
- SARIMA over-fits on high-volume series
- SARIMA showed a better performance on stores with lower volume of observations

BASELINE RESULTS ARIMA vs ARIMAX

Model	Exogs Used	MAE
ARIMA	None	2846.303435
ARIMAX	Stock	3026.366468
ARIMAX	Stock + Price	3026.366468
ARIMAX	Top 5 Features	3026.366468
ARIMAX	All Features	2846.303435
LinearReg	Stock + Price	619.562897

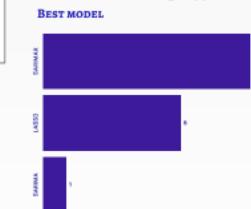
- We start our model baseline tests with Arima/Arimax and Lin.Regession to validate their predictive results
- Main objective was to obtain some overall results about each P.O.C hierarchy
- Major insight - ARIMAX was not capturing the signals from the different exog. features

MODEL SELECTION - FINAL WINNERS

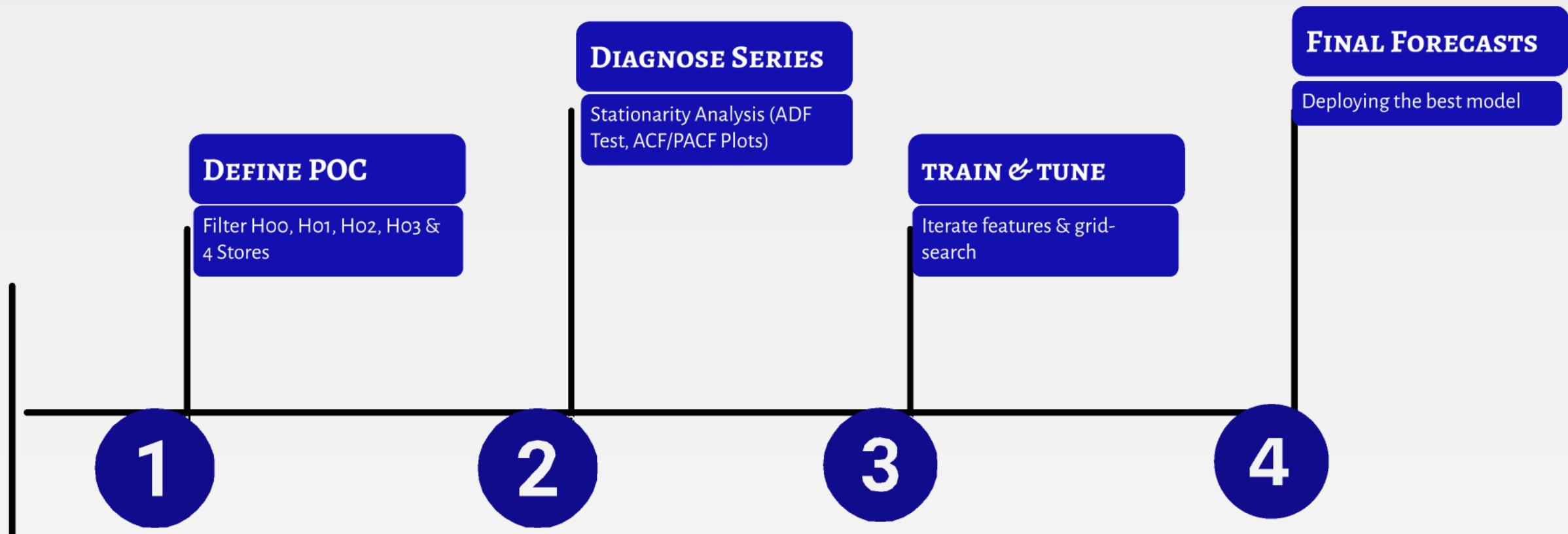
Hierarchy	Store	SARIMA Test	SARIMAX Test	LASSO Train	LASSO Test	LASSO R ² Test	Best Model
H00	S0095	12.14	12.14	12.14	12.14	0.99	SARIMAX
	S0085	133.76	54.73	0.99	1.00	0.80	LASSO
	S0026	41.01	9.82	0.51	0.52	0.17	ARIMAX
	S0080	81.46	70.26	0.75	0.76	0.69	LASSO
H01	S0095	6.65	3.51	0.28	0.18	0.28	SARIMAX
	S0085	32.12	16.74	0.37	0.33	0.15	SARIMAX
	S0026	29.72	7.72	0.34	0.29	0.11	SARIMAX
	S0080	1.86	0.29	0.24	0.27	-0.49	SARIMAX
H02	S0095	5.21	5.21	0.85	0.85	-0.01	SARIMAX
	S0085	1.6	1.6	3.25	1.40	-1.97	SARIMAX
	S0026	16.27	18.27	2.34	1.73	-0.28	SARIMA
	S0080	0.87	0.87	0.29	0.42	-0.08	SARIMAX
H03	S0095	9.37	5.02	0.23	0.20	0.82	LASSO
	S0085	18.11	3.81	0.16	0.16	0.72	LASSO
	S0026	12.72	8	0.18	0.19	0.84	LASSO
	S0080	11.72	11.29	0.36	0.42	0.88	SARIMAX



- Criteria → Positive R² + No over-fit + Lowest MAE.
- Best Model: SARIMAX = 9



4-STEP MODELLING PIPELINE



BASELINE RESULTS

ARIMA vs ARIMAX

Model	Exogs Used	MAE
ARIMA	None	2846.303435
ARIMAX	Stock	3026.366468
ARIMAX	Stock + Price	3026.366468
ARIMAX	Top 5 Features	3026.366468
ARIMAX	All Features	2846.303435
LinearReg	Stock + Price	619.562897

- We start our model baseline tests with Arima/Arimax and Lin.Regession to validate their predictive results
- Main objective was to obtain some overall results about each P.O.C hierarchy
 - Major insight - ARIMAX was not capturing the signals from the different exog. features

P.O.C RESULTS - SARIMA/ SARIMAX

MAE VALUES

Hierarchy	Store	SARIMA Train	SARIMA Test	SARIMAX Train	SARIMAX Test
H00	S0095	14.47	62.3	14.9	12.14
	S0085	53.47	123.76	44.16	34.73
	S0026	20.79	41.01	20.88	9.82
	S0080	12.63	81.46	12.95	70.26
H01	S0095	3.73	6.65	3.65	3.51
	S0085	11.51	32.12	10.52	16.74
	S0026	7.63	29.72	7.83	7.72
	S0080	1.49	1.86	1.54	0.29
H02	S0095	3.79	5.21	3.79	5.21
	S0085	2.91	1.6	2.78	1.12
	S0026	4.55	16.27	68.82	18.27
	S0080	0.55	0.87	0.55	0.87
H03	S0095	4.12	9.97	3.52	5.02
	S0085	9.1	18.11	9.04	3.81
	S0026	7.15	12.72	7.67	8
	S0080	2.57	11.72	2.59	11.29

Not Overfitted

Light Overfitting

Overfitting

- SARIMAX wins 15 of 16 cases (94%)
- SARIMA over-fits on high-volume series
- SARIMA showed a better performance on stores with lower volume of observations

MODEL SELECTION - FINAL WINNERS

Hierarchy	Store	SARIMA Test	SARIMAX Test	LASSO Train	LASSO Test	LASSO R ² Test	Best Model
H00	S0095	62.3	12.14	0.40	0.23	0.80	LASSO
	S0085	123.76	34.73	0.99	1.00	0.80	LASSO
	S0026	41.01	9.82	0.51	0.52	0.17	SARIMAX
	S0080	81.46	70.26	0.75	0.78	0.69	LASSO
H01	S0095	6.65	3.51	0.28	0.18	0.28	SARIMAX
	S0085	32.12	16.74	0.37	0.33	0.15	SARIMAX
	S0026	29.72	7.72	0.34	0.29	0.11	SARIMAX
	S0080	1.86	0.29	0.24	0.27	-0.49	SARIMAX
H02	S0095	5.21	5.21	8.61	8.30	-5.52	SARIMAX
	S0085	1.6	1.12	2.25	1.83	-1.87	SARIMAX
	S0026	16.27	18.27	2.34	1.73	-0.28	SARIMA
	S0080	0.87	0.87	0.29	0.42	-0.08	SARIMAX
H03	S0095	9.97	5.02	0.23	0.23	0.82	LASSO
	S0085	18.11	3.81	0.16	0.19	0.72	LASSO
	S0026	12.72	8	0.18	0.19	0.84	LASSO
	S0080	11.72	11.29	0.36	0.42	0.68	SARIMAX

- R²-SCORE

Good Poor Negative

- MAE VALUES

Good Bad

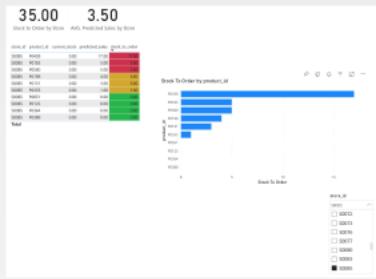
- Criteria → Positive R² + No over-fit + Lowest MAE.
- Best Model: SARIMAX = 9

BEST MODEL



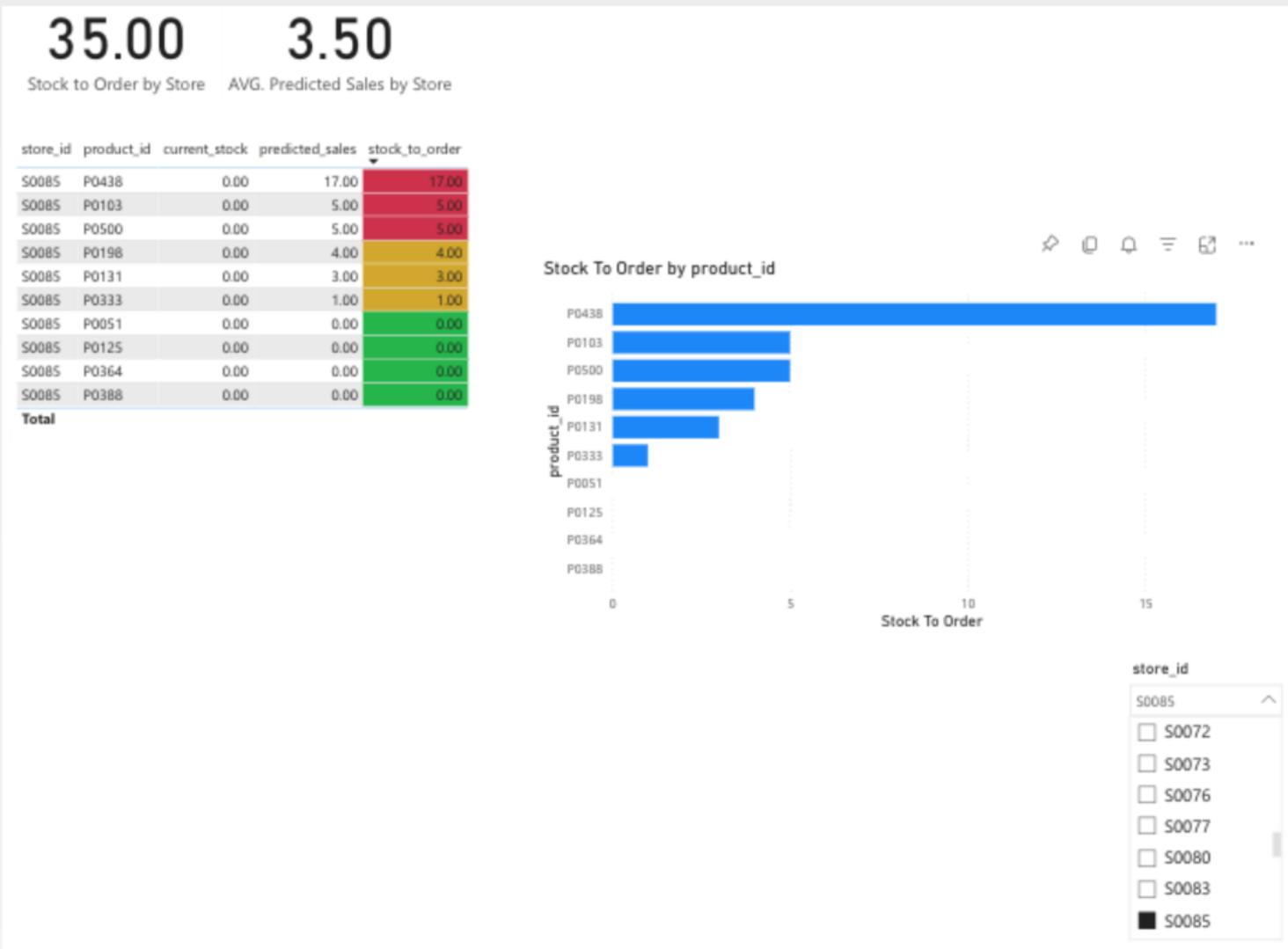
DASHBOARD

LIVE REPLENISHMENT DASHBOARD



- Auto-Refreshed from ML Pipeline
- Stockholders see exact product units to order per store

LIVE REPLENISHMENT DASHBOARD



- Auto-Refreshed from ML Pipeline
- Stockholders see exact product units to order per store

THANK YOU ALL!!!

GRUPO_III

ANDRÉ TEIXEIRA

PATRÍCIA PEREIRA

RODRIGO DIOGO

VITOR MEIRELLES