Problem Statement

For grocery retailers, driving strong unit movement is essential because shelf space is fixed, limited, and constantly under pressure. Every aisle represents valuable real estate, and products must turn quickly enough to justify the space they occupy.

Given this, the key question for the UK Grocery Retailer (UGR) is: **Can they increase unit sales in brick-and-mortar stores (excluding e-commerce) by adjusting the pricing and distribution of top-selling items?**

Key consideration:
Any lift from a price decrease must offset the reduction in retail unit price. (For example, a 10% price decrease must produce a unit sales lift greater than 10%.)

Data Wrangling

I began by importing and validating the raw sales data and item-level metadata. Initial steps included:

- Ensuring all numerical features were stored as numeric data types
- Confirming there were **no missing values**, **no duplicates**, and **no negative entries**

After merging the item metadata into the sales table and removing irrelevant inventory fields, I filtered out e-commerce transactions to focus strictly on brick-and-mortar retail sales (as required by the project scope).

A key step was identifying the items that drive the majority of the business. By ranking products by total revenue and examining cumulative contribution, we observed a strong **Pareto pattern**—a small number of items accounted for most of the sales. I isolated the **top 80% revenue-contributing items**.

The result was a fully cleaned, merged, and segmented dataset ready for exploratory analysis and modeling.

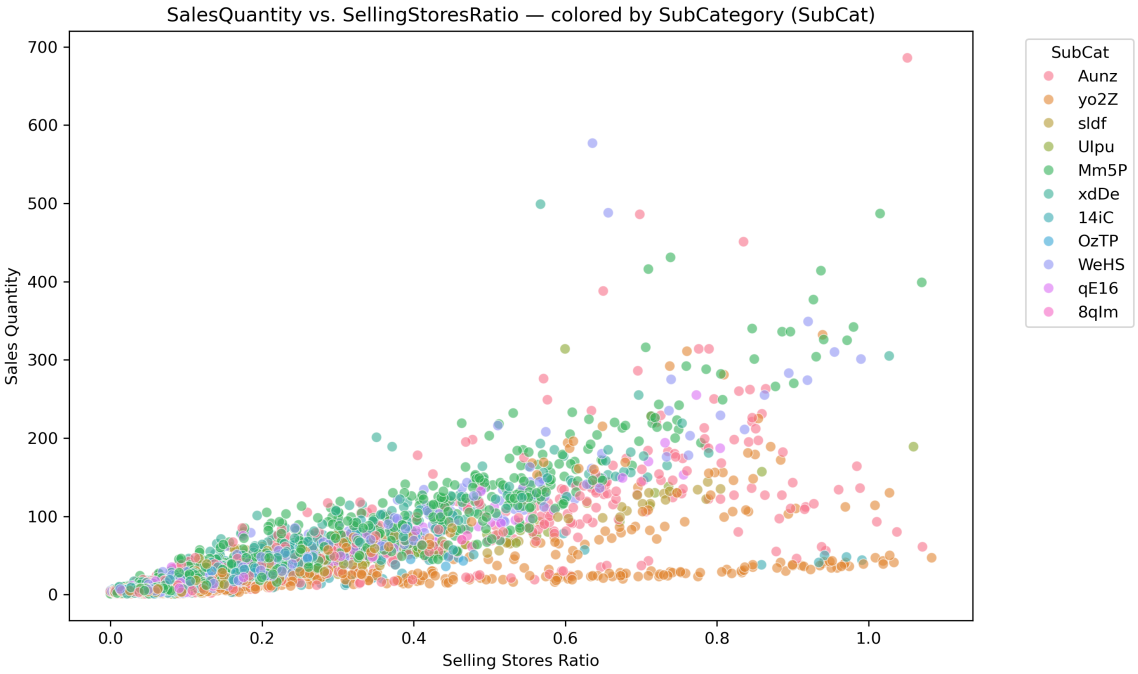
Exploratory Data Analysis (EDA)

The exploratory analysis focused on understanding distributional patterns, feature relationships, and structural properties of the cleaned dataset.

Summary statistics confirmed the dataset was complete, consistent, and free of missing values or duplicates.

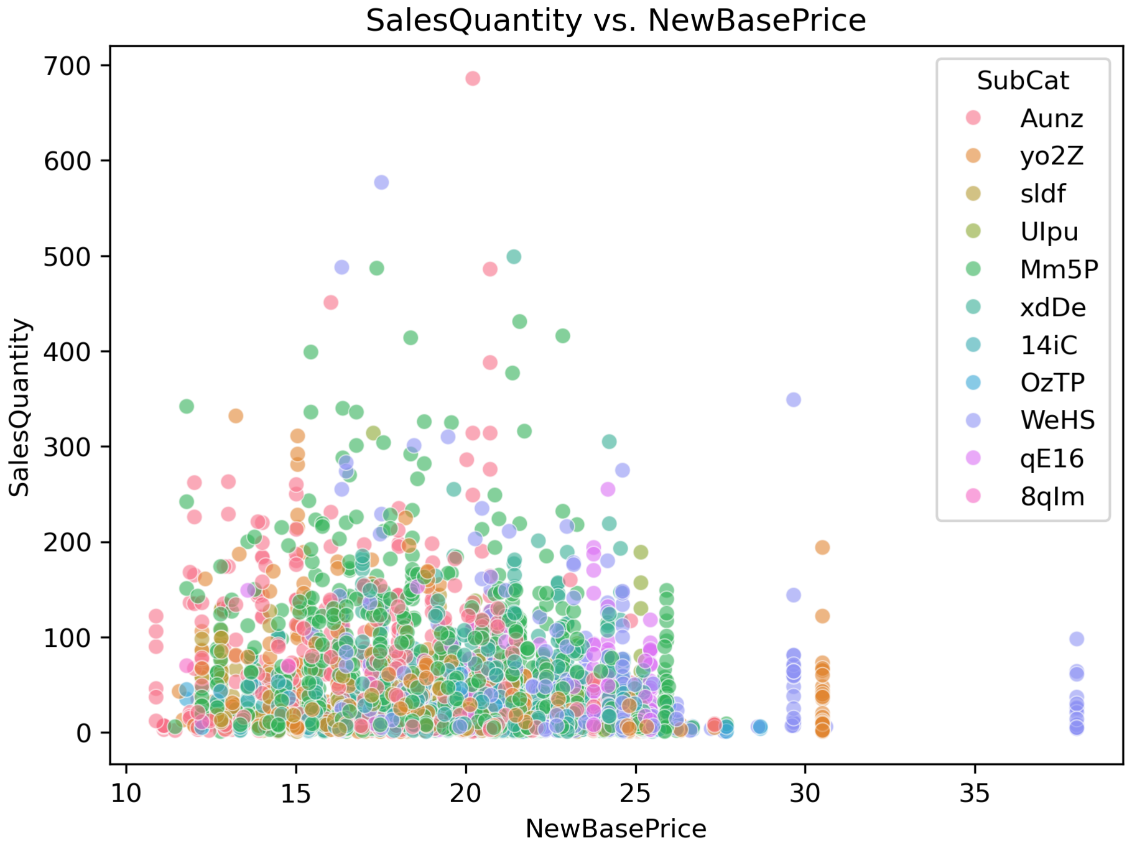
SalesQuantity and SellingStoresRatio (Distribution)

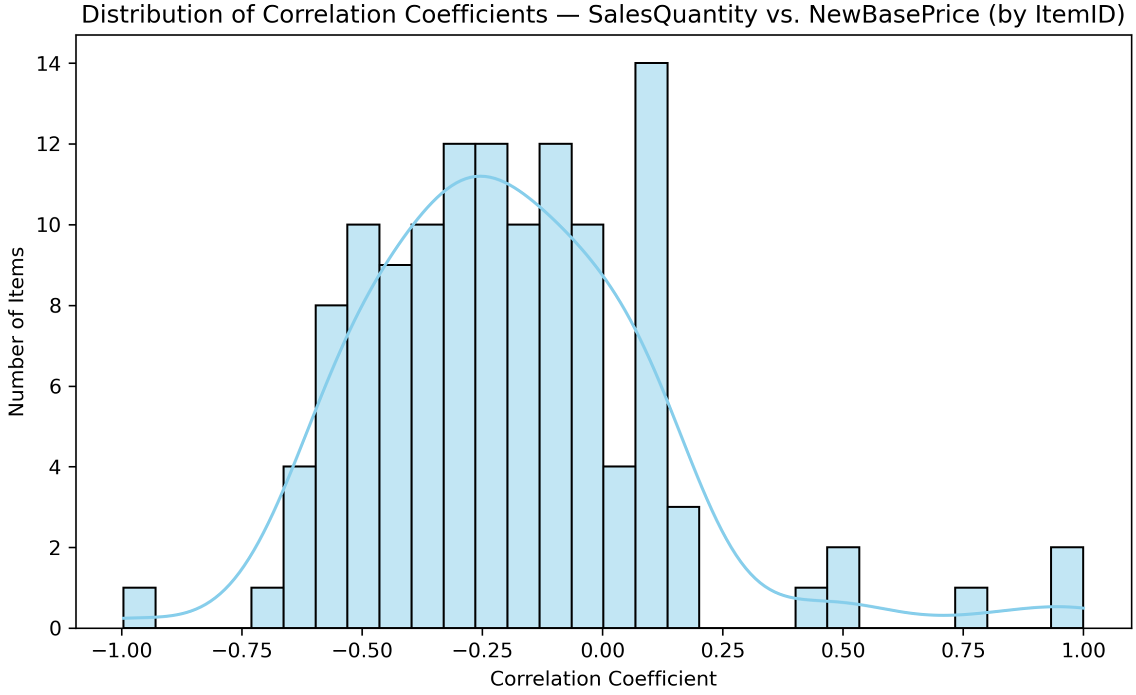
Scatterplots revealed a clear positive relationship between SalesQuantity and distribution breadth. This relationship was stronger for some sub-categories than others.



SalesQuantity and NewBasePrice (Price)

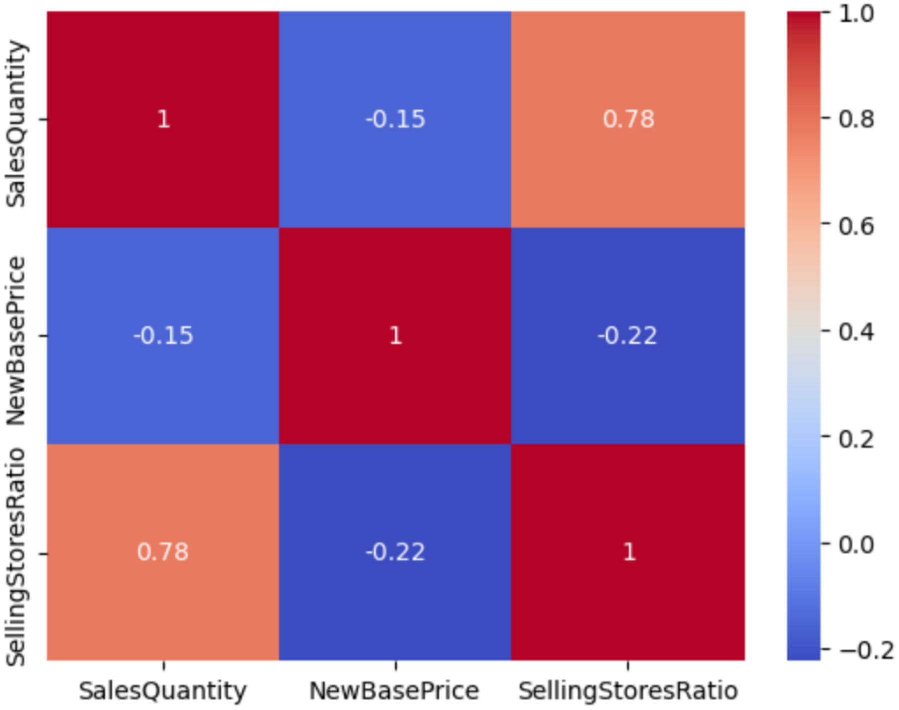
Scatterplots showed **no visible relationship** between price and demand at the aggregate level. Correlation analysis confirmed this—negative correlations between Price and Sales occurred only at the **individual item level**, not at the dataset level.


SalesQuantity vs. NewBasePrice
SalesQuantity
NewBasePrice
0
100
200
300
400
500
600
700
10
15
20
25
30
35
SubCat
Aunz
yo2Z
sldf
Ulpu
Mm5P
xdDe
14iC
OzTP
WeHS
qE16
8qlm


Distribution of Correlation Coefficients — SalesQuantity vs. NewBasePrice (by ItemID)
Number of Items
Correlation Coefficient
0
2
4
6
8
10
12
14
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

Multivariate Relationships

A heatmap of numerical features confirmed the insights from scatterplots. Importantly, **Price and Distribution were not correlated**, indicating **no multicollinearity concerns** in the predictive modeling phase.



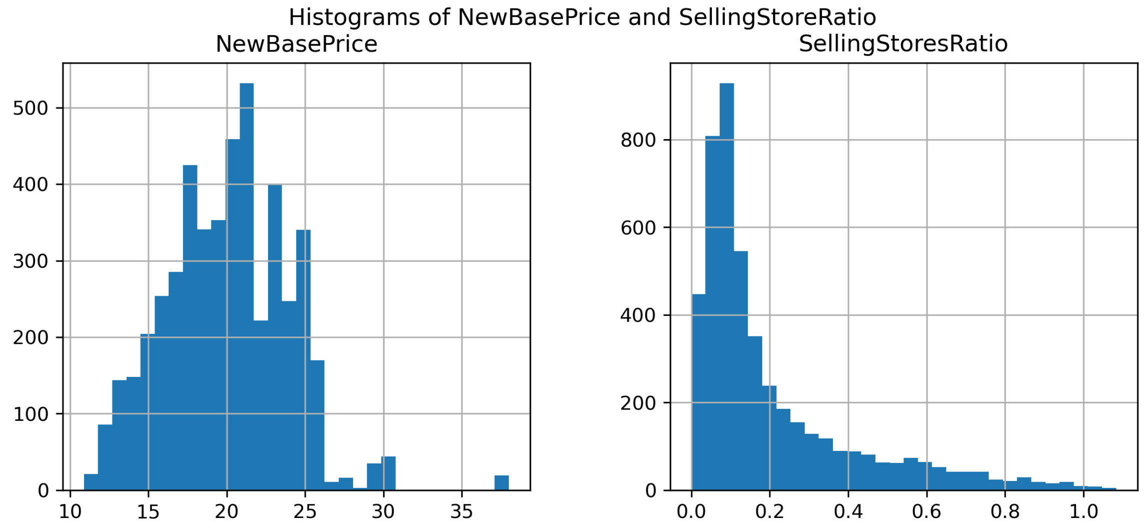
Pre-Processing

The pre-processing phase prepared the dataset for modeling through transformation, scaling, and encoding.

I defined **SalesQuantity** as the target variable and selected a focused predictor set, removing identifiers and fields that could introduce information leakage.

Numerical and categorical features were separated to allow appropriate transformations:

- Several numerical variables (including SalesQuantity and SellingStoresRatio) were strongly right-skewed, so **log transformations** were applied.
- All numerical features were **standardized (z-scaled)** to support regression-based and distance-based algorithms.
- Categorical variables (Category, SubCategory) were **one-hot encoded**.



A unified **scikit-learn preprocessing pipeline** was built to ensure consistency, reproducibility, and protection against leakage.
The final output was a fully transformed training and test set (test size 20%, random state = 42), ready for model fitting.

Modeling

The modeling phase evaluated several algorithms to predict **SalesQuantity** using variables such as price, distribution, and product characteristics.

A baseline **Linear Regression** model provided modest performance and revealed non-linear relationships.
Regularized models (**Ridge**, **Lasso**) improved model stability but did not materially improve accuracy.

Tree-based methods performed substantially better:

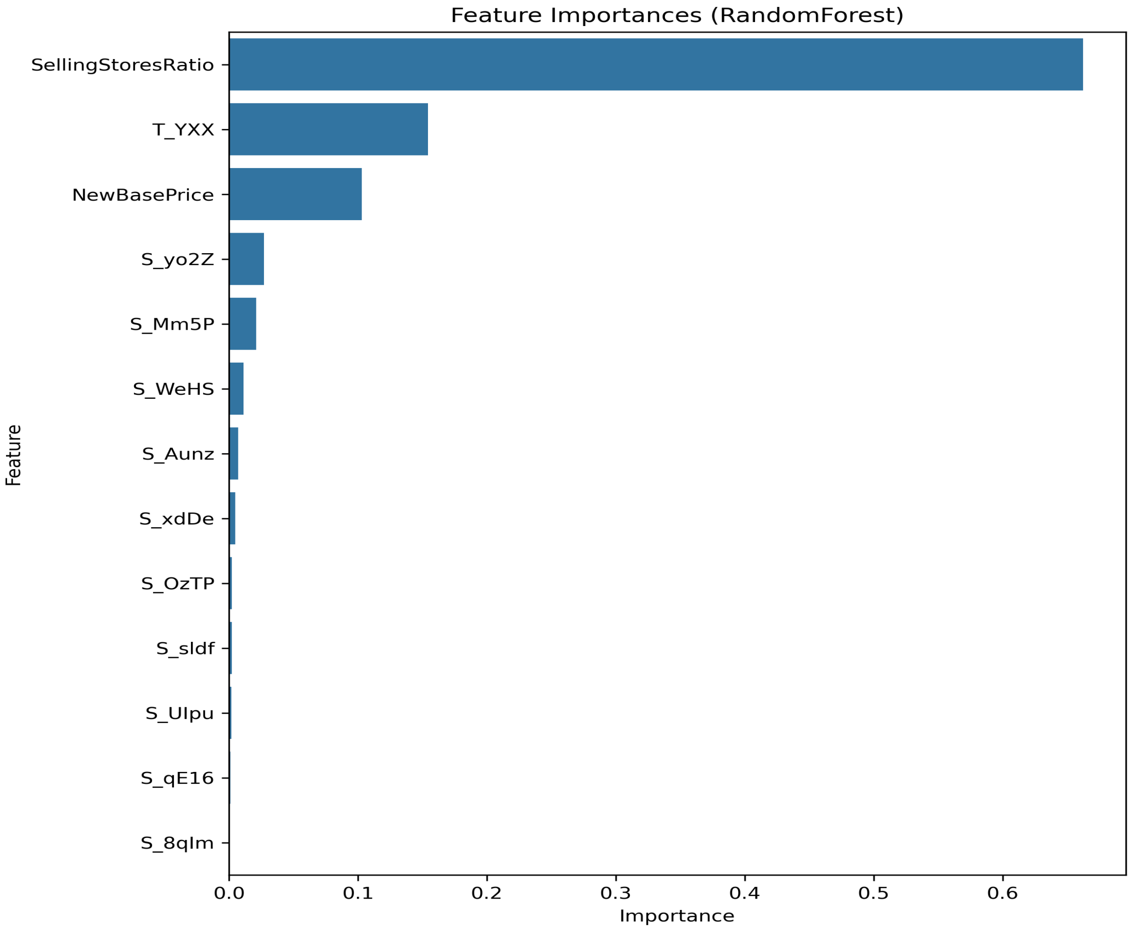
- **Random Forest Regressor** delivered the best accuracy, lowest error, and most stable residuals.
- It effectively captured complex interactions and non-linearities that linear models could not.

Model	R2 (test)	RMSE (test)	MAE (test)
RandomForest	0.731266	29.189361	10.775594
KNNRegressor	0.717366	29.934731	11.255149
GradientBoosting	0.693138	31.191400	12.247399
Lasso	0.428736	42.558063	23.639246
Ridge	0.428717	42.558779	23.642985
Linear Regression	0.428701	42.559347	23.653237

Best Hyperparameters (per tuned model)

- Ridge → alpha = 3.1623
- Lasso → alpha = 0.0121
- RandomForest → max_depth = 20, n_estimators = 200, max_features = "sqrt", min_samples_leaf = 2
- GradientBoosting → learning_rate = 0.05, n_estimators = 200, max_depth = 3
- KNN → n_neighbors = 15, weights = "distance"

Random Forest feature importance showed that **distribution (SellingStoresRatio)** is the single biggest driver of unit sales. The next strongest predictor was the territory variable, suggesting one territory represents a larger market. Price mattered but to a far lesser degree.



Hyperparameter Tuning

Random Forest tuning produced only nominal improvements, so the base model was used for predictions.

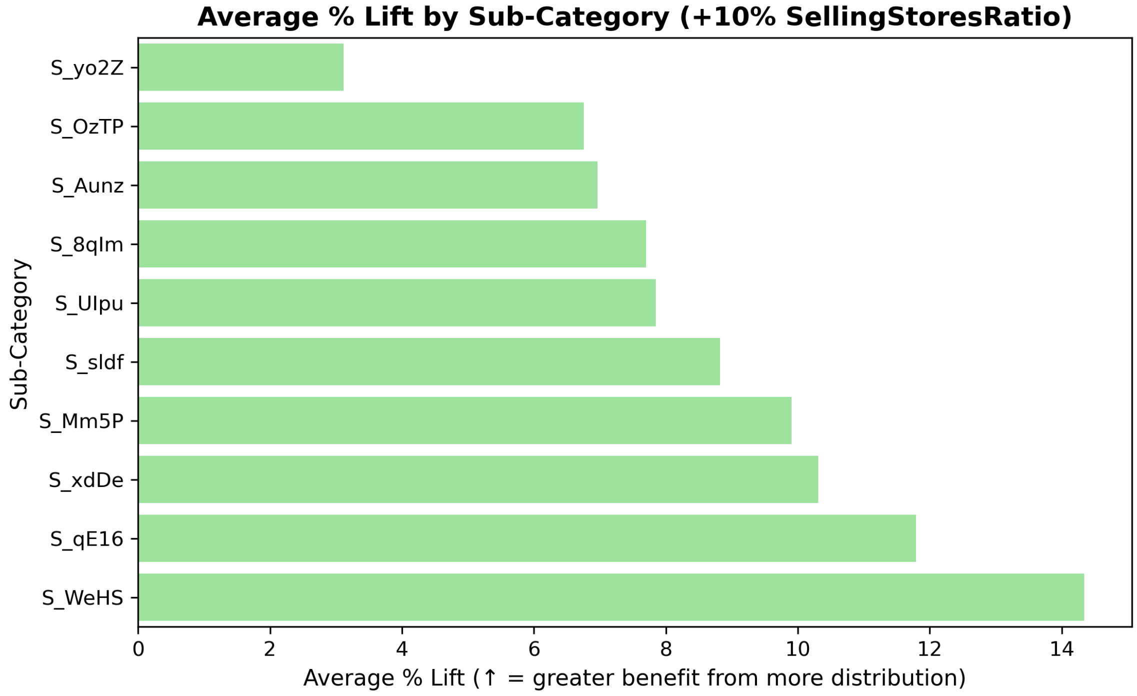
Model	R2 (test)	RMSE (test)	MAE (test)
RandomForest (base)	0.731266	29.189361	10.775594
Randomized SearchCV	0.734607	29.007341	10.62336
GridSearchCV	0.733962	29.042605	10.593079

Predictions

Result:

A 10% increase in distribution yields an **average +8.64% lift in SalesQuantity**.

As expected, some sub-categories respond more strongly than others.



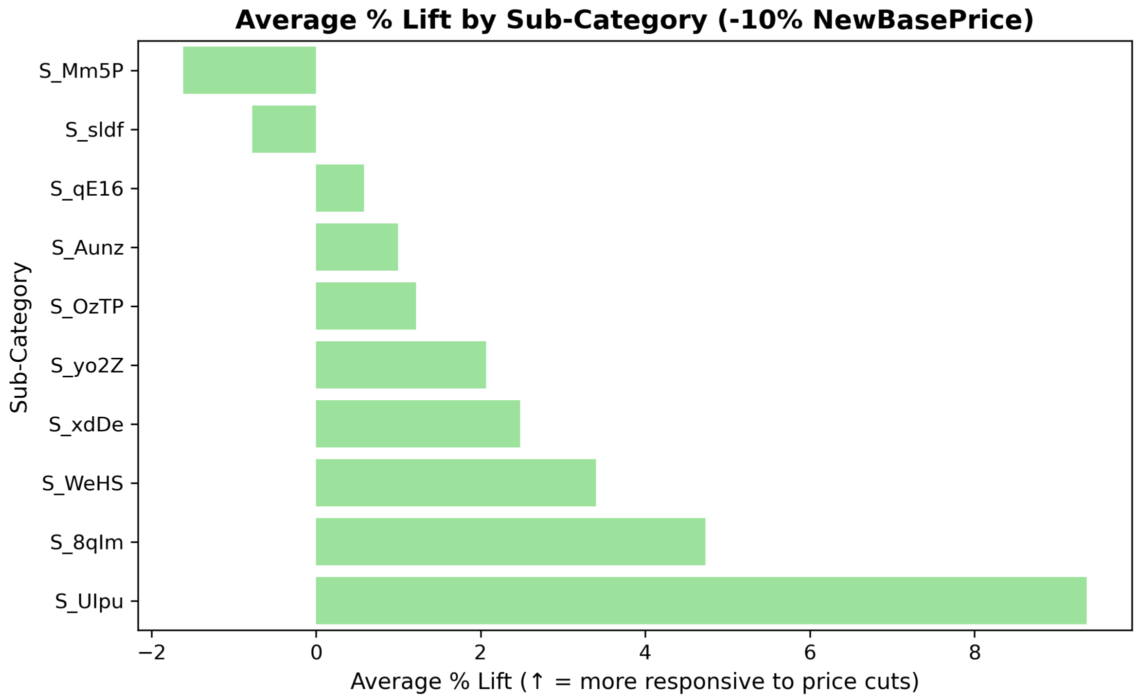
I then assessed the impact of a **–10% price decrease**.

Result:

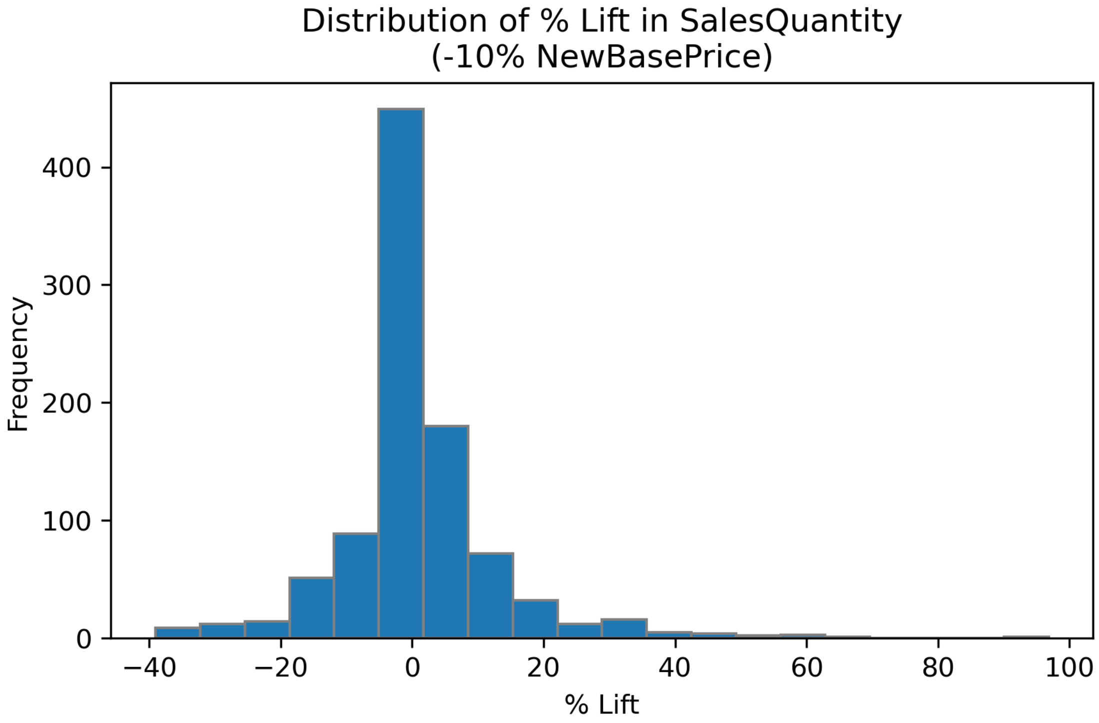
A 10% price decrease yields an **average +1.18% lift** in SalesQuantity.

This aligns with EDA and modeling results showing that **price is not a major driver of demand** in this dataset.

Some sub-categories exhibit higher price sensitivity, but **average lifts remain well below 10%**, meaning the increase in units does **not** offset the revenue loss from lower prices.



Certain items are highly price sensitive and may warrant item-level evaluation



Recommendations

- **Increase distribution coverage** for the items driving 80% of revenue, prioritizing sub-categories that show the highest distribution-driven lift.
- **Avoid broad price reductions** across sub-segments. Most items exhibit low price sensitivity; unit lifts do not offset reduced retail price, resulting in lower revenue.
- Evaluate **item-level price decreases** only for items that show strong, measurable sensitivity.
- Explore price increases as a potential lever ***if the goal is to increase total revenue***. For majority of the sub-categories, the decrease in sales unit will be more than offset by the increase in sales price.

Future research:

- Assortment optimization (of current assortment)
- Research expanded offerings to replace low revenue items