Amazon Sales

Nicholas Morris

Machine Learning

Wrangling → Feature Engineering → Modeling

Text Manipulation

Extracting the brand name from product name.
Simplifying the category.
Removing currency symbols from price.
Removing the percent symbol from discount.

Transformations

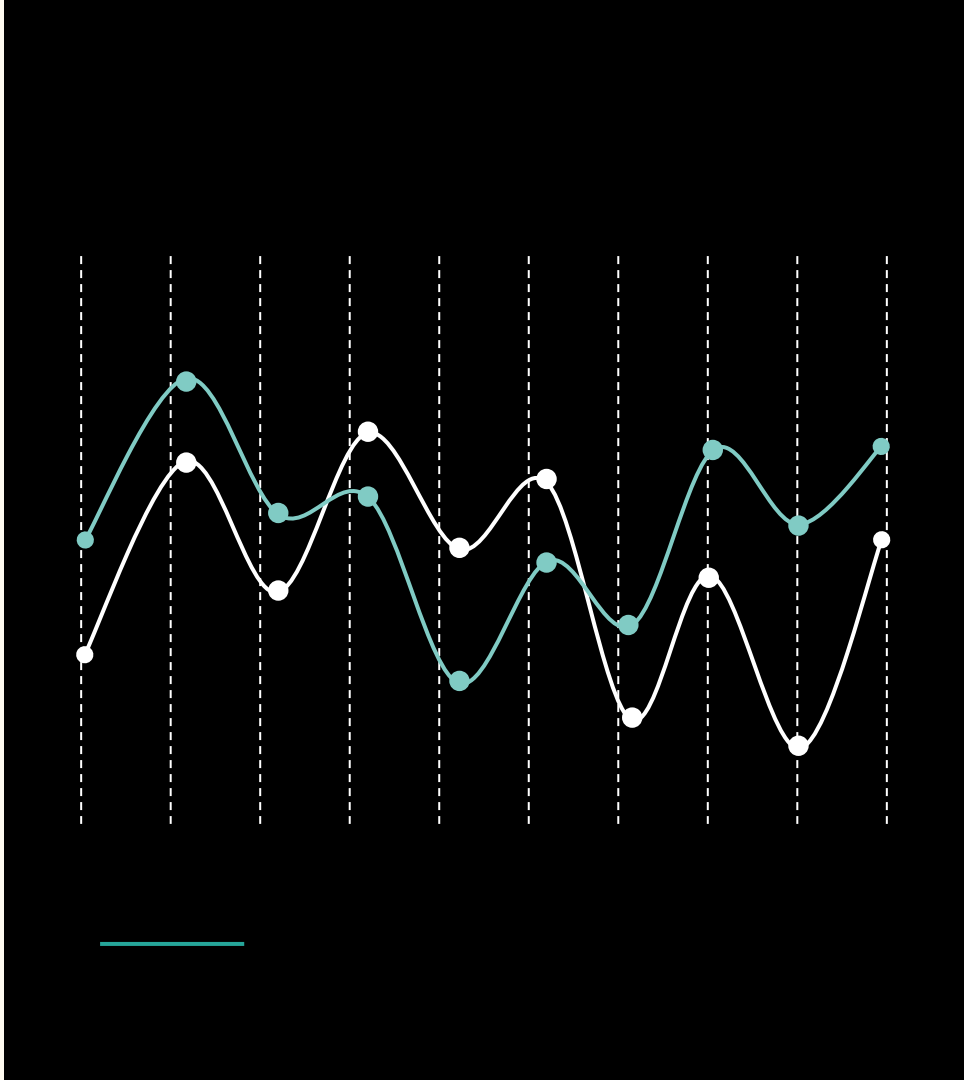
Extracting the individual words from reviews.
Computing the positivity of reviews.

Attempting feature transformations.

Predictions

Training linear regression, XGBoost, and deep learning neural network models.
Evaluating performance.
Computing feature drift to signal retraining.

Wrangling



Dataset

Below is the first review in the data. There are 1,465 total reviews and 16 columns. The target we are predicting is actual_price. [Link to the dataset and code]

product_id	product_name	category	discounted_price	actual_price	discount_percentage	rating	rating_count	about_product	user_id	user_name	review_id	review_title	review_content	img_link	product_link
B002PD61Y4	D-Link...	Computers&Accessories...	₹507	₹1,208	58%	4.1	8,131	Connects your...	AGA2P...	nilesh,EAGLE,...	R2EJIN...	good tool...	good quality tool...	https://m.media-amazon.com...	https://www.amazon.in...

Removing Unnecessary Columns

Discounted_price, user_id, user_name, review_id, img_link, and product_link were all removed from the data.

Product Name

The first word in product_name was extracted.

Category

The first section in category was extracted.

Actual Price

The ₹ symbol and commas were removed from actual_price.

Discount Percentage

The percent symbol was removed from discount_percentage.

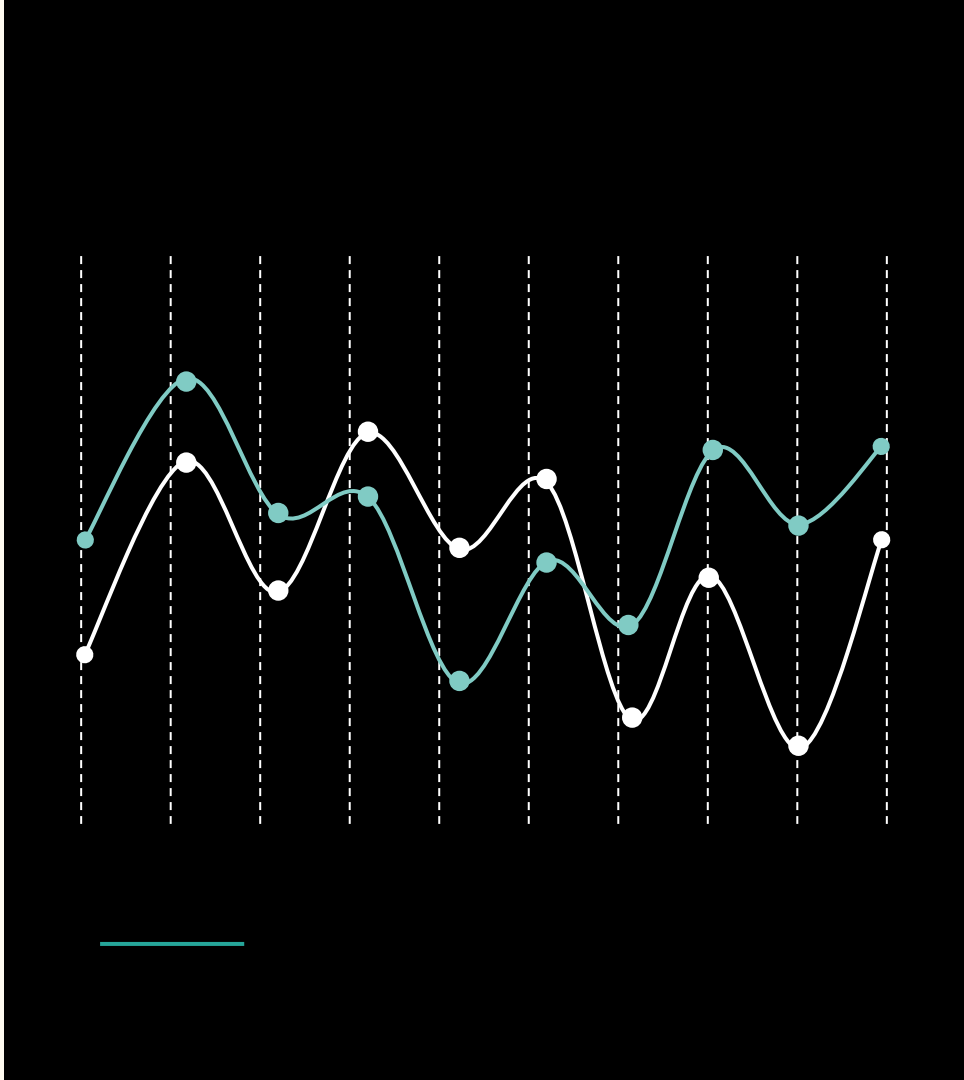
Rating Count

Commas were removed from rating_count.

Aggregation

The data was aggregated by product_id so each product_id appears once.

Feature Engineering



Text Data

The reviews were transformed so that each word in the reviews got their own binary column. The positivity of the reviews were also computed.

Binary Data

product_name and category were converted to binary data points.

Atwood Numbers

An Atwood Number is a calculation that shows the relative change between two variables. The formula for two variables x and y is:
$(x - y) / (x + y)$

This calculation was done on all pairs of non-binary variables; but did not improve model performance, so, it was left out of the final model.

Binning

Binning is when a non-binary variable is grouped into histogram bins, and represented as binary variables.

Binning did not improve model performance, so, it was left out of the final model.

Reciprocals

A reciprocal is when a non-binary variable x is calculated as 1 / x.

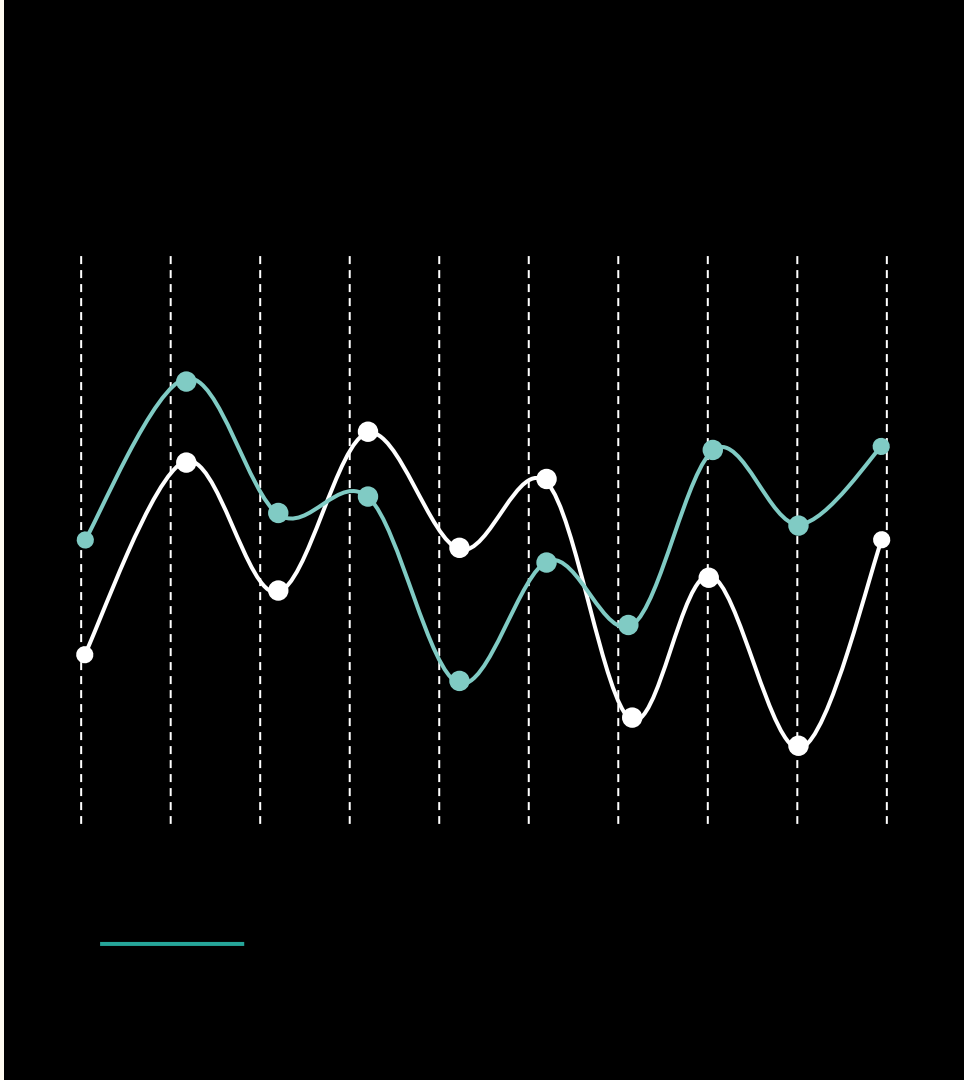
Reciprocals did not improve model performance, so, it was left out of the final model.

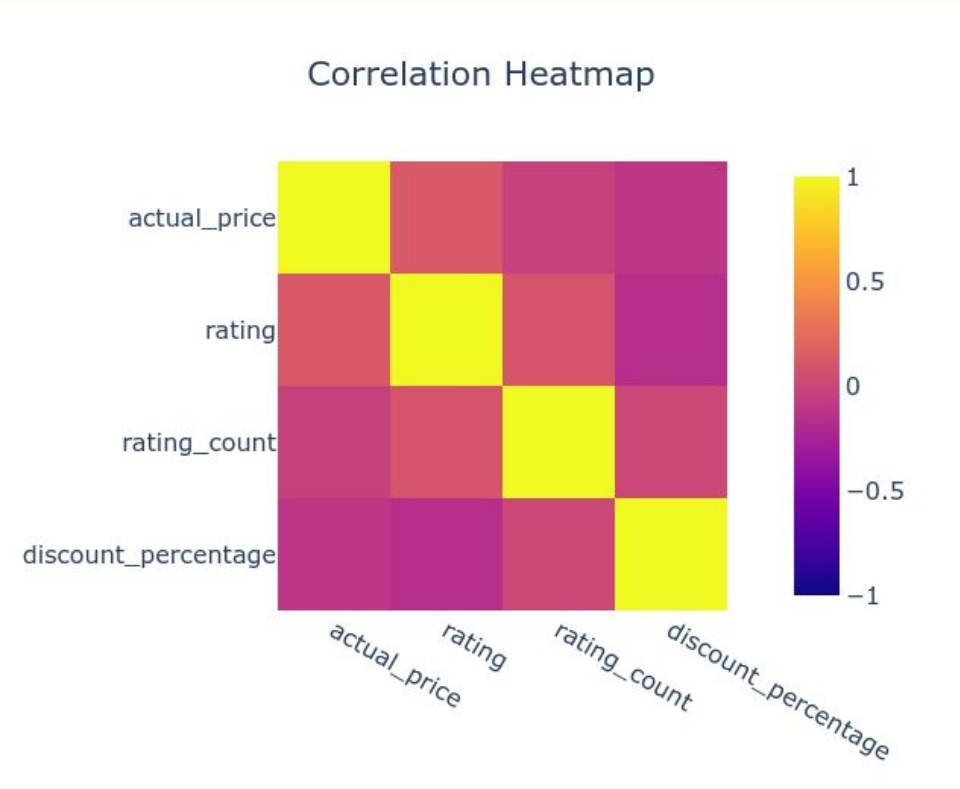
Interactions

An interaction is when two variables x and y are calculated as x * y. Reciprocals were fed into this calculation to generate x / y as well.

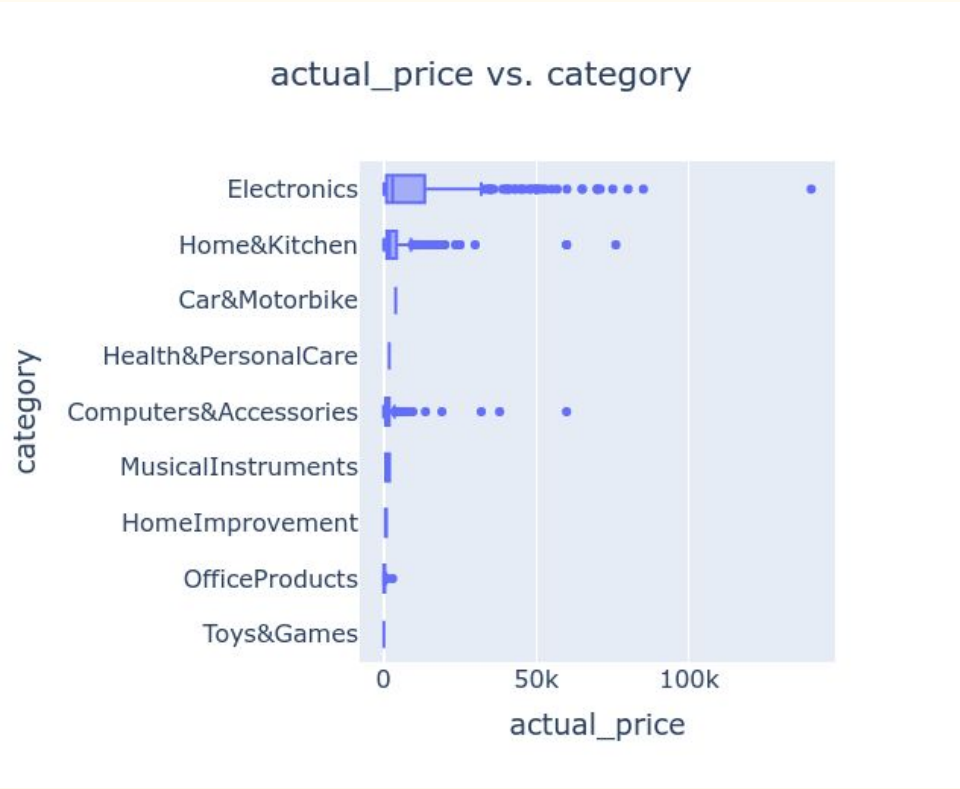
Interactions did not improve model performance, so, it was left out of the final model.

Data Exploration

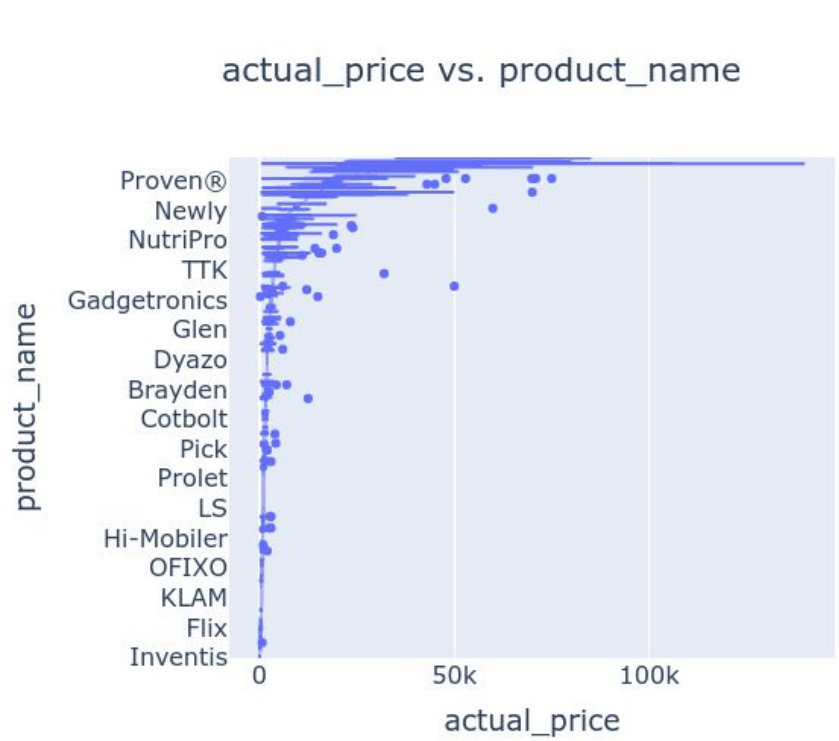




There are no strong correlations in the data.

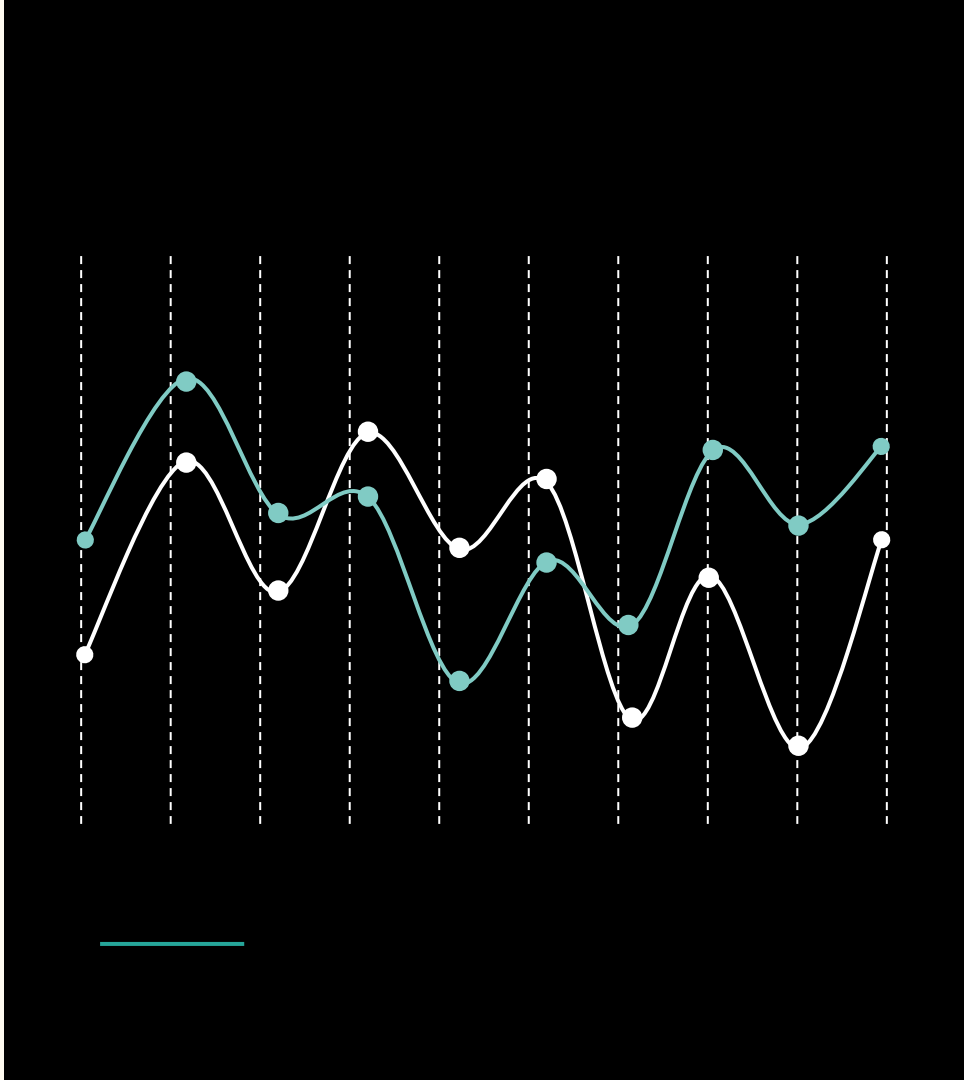


Price varies by category with Electronics having the highest prices and Toys&Games having the lowest prices.



Price varies by product name with Proven having the higher prices than Inventis.

Modeling



Model Parameters

Linear Regression

Library: scikit-learn
Length Of Path: 1e-9
Number Of Alphas: 16
Cross Validation Folds: 3
Tolerance: 1e-4
Max Iterations: 500

XGBoost

Library: xgboost
Boosting Rounds: 100
Learning Rate:
0.001, 0.01, 0.1
Max Depth:
5, 7, 10, 14, 18
Min Child Weight: 1
Column Sampling: 0.8
Row Sampling: 0.8
Cross Validation Folds: 3

Neural Network

Library: Tensorflow
Epochs: 500
Learning Rate:
0.0001, 0.001, 0.01
Batch Size: 16
Layers: 10
Nodes Per Layer:
32, 64, 128, 256, 512
Solver: Adam
Cross Validation Folds: 3

Model Comparison

Linear Regression

R2: 0.74
RMSE: 6303
In Control: 95.91%

Model Indicators:
1. about product: 2160
2. about product: 3840x2160
3. about product: x1
4. review content: competitors
5. about product: processor

XGBoost

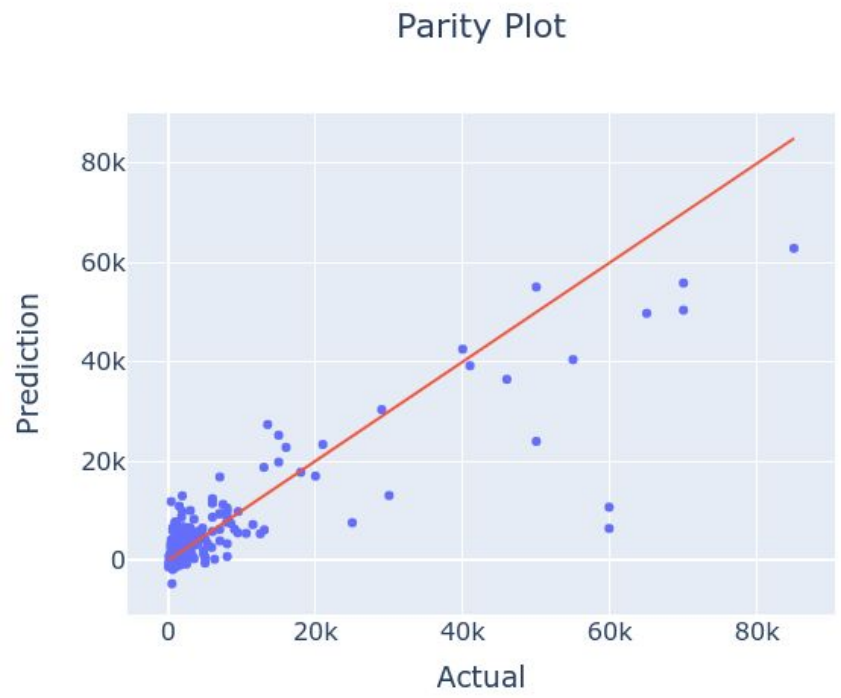
R2: 0.72
RMSE: 6464
In Control: 94.42%

Model Indicators:
1. about product: refresh
2. about product: hertz
3. review content: display
4. review content: absolutely
5. about product: ultra

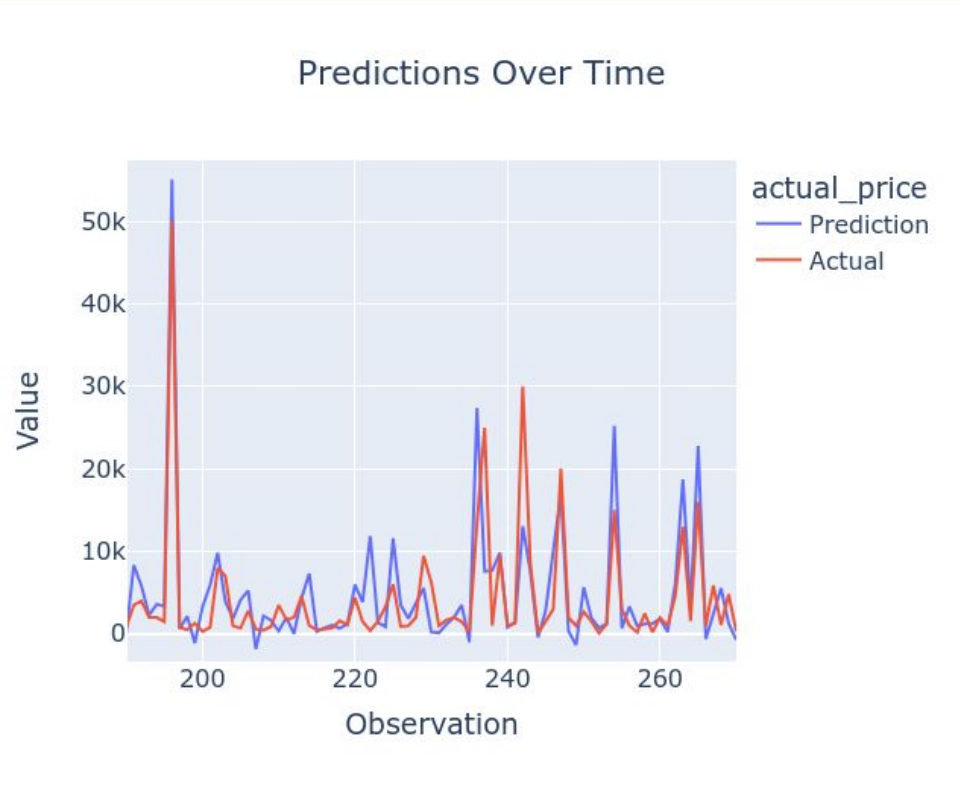
Neural Network

R2: 0.69
RMSE: 6630
In Control: 96.28%

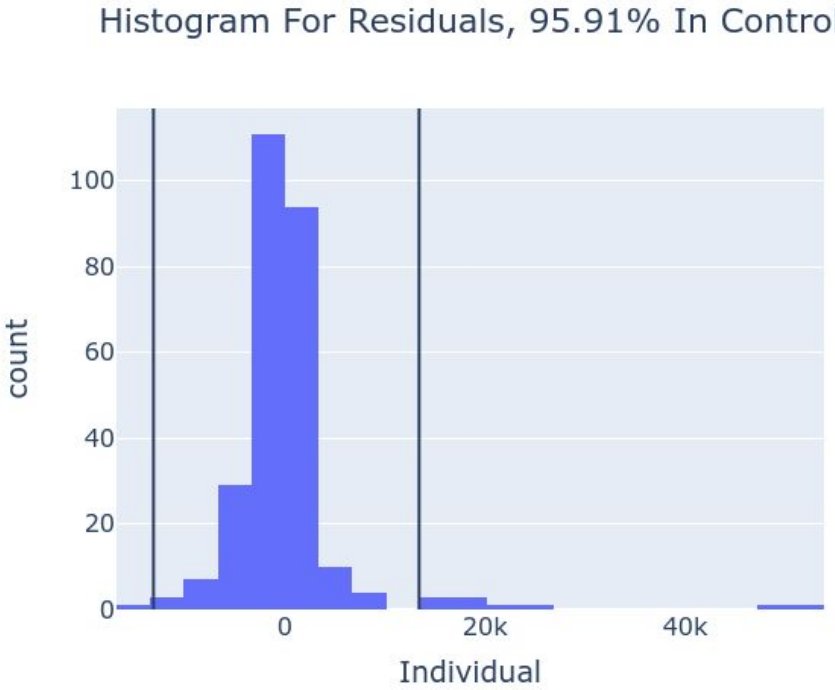
Model Indicators:
DNF



These predictions come from the linear regression model. These predictions are done on 20% of the data that the model did not see during training. The predictions are centered on the red line (perfect predictions). There is a tendency to under-predict larger prices.

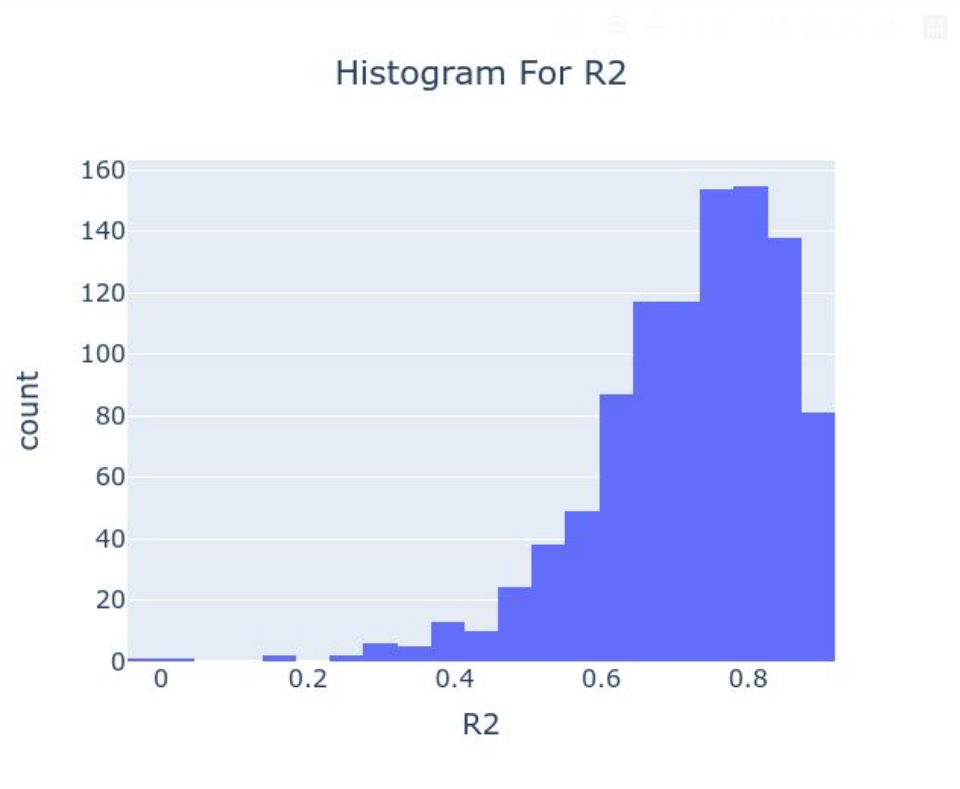


Here's a snapshot of the predictions over time. We can see the the blue predictions follow the actual values well. There are some instances where the predictions over-predict price. We can see there is no seasonality in the data.



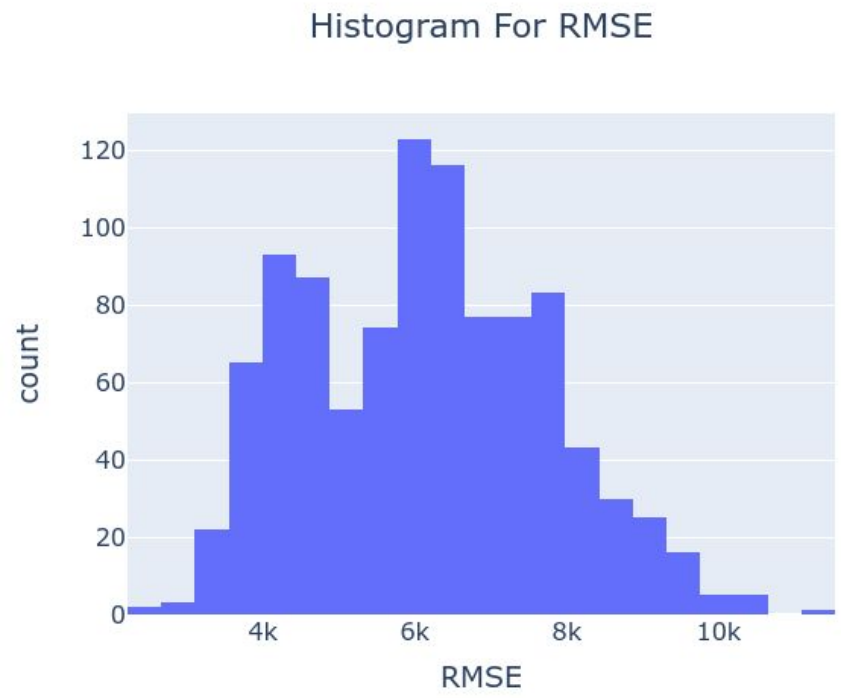
The residuals are prediction error = actual - predicted.

The residuals have a tight bell shape, which is good, and they are centered on zero. Control limits were computed on the residuals and we can see that the prediction error is mostly under control. We can see a skew to the right, which shows the model has a tendency to under-predict the price.



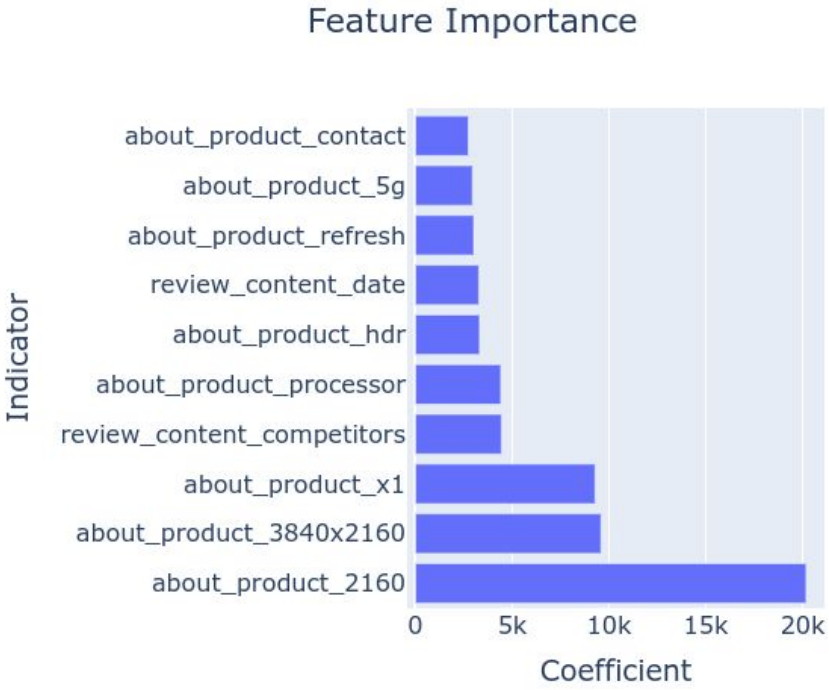
The prediction error was resampled 1000 times at a 50% sampling rate with replacement. Then R2 was computed on each sample to get a distribution.

R2 has a wide range between 0 and 0.8, which is bad. R2 has a bell shape, which is good, and a large skew to the left.



The prediction error was resampled as previously mentioned to get a distribution for RMSE.

On average, the predictions are off by 4 thousand to 10 thousand, which is a wide range. There are two modes in the data around 4 thousand and 6 thousand, which isn't good.

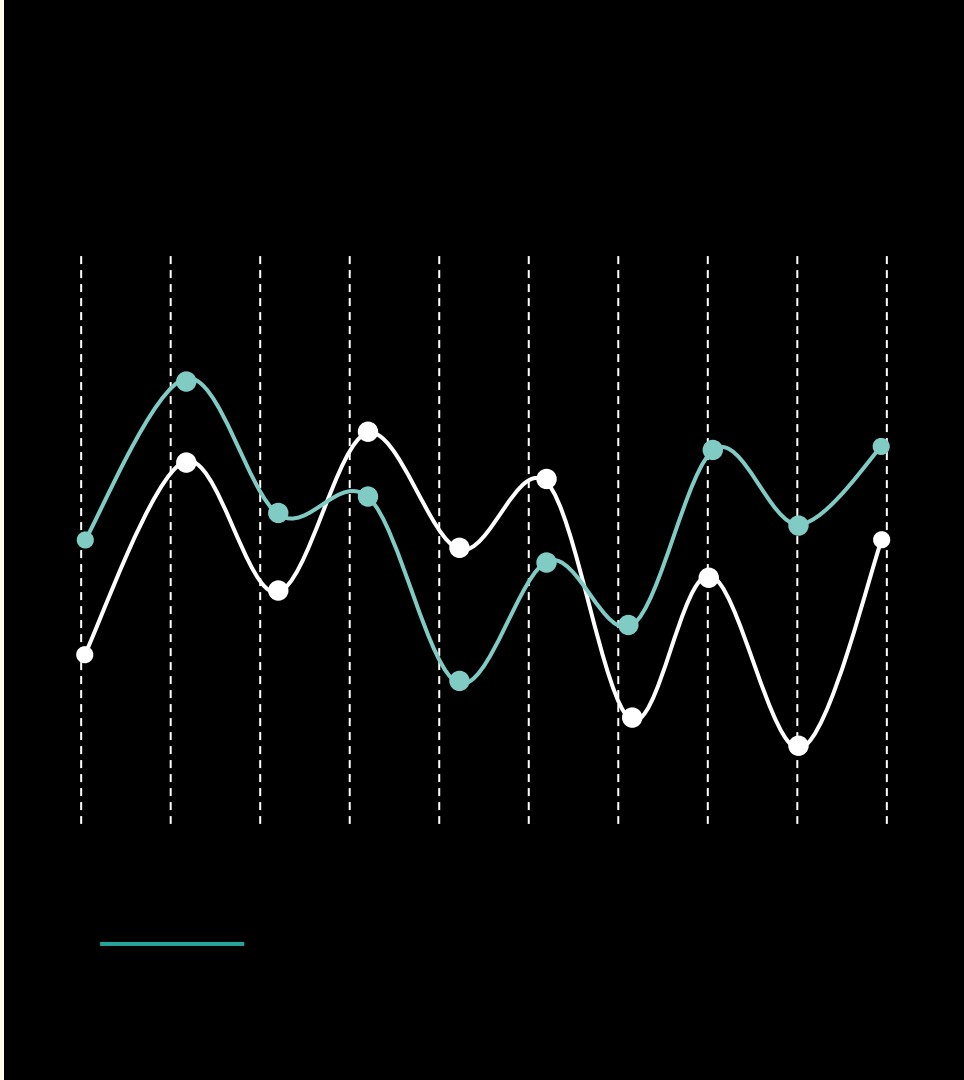


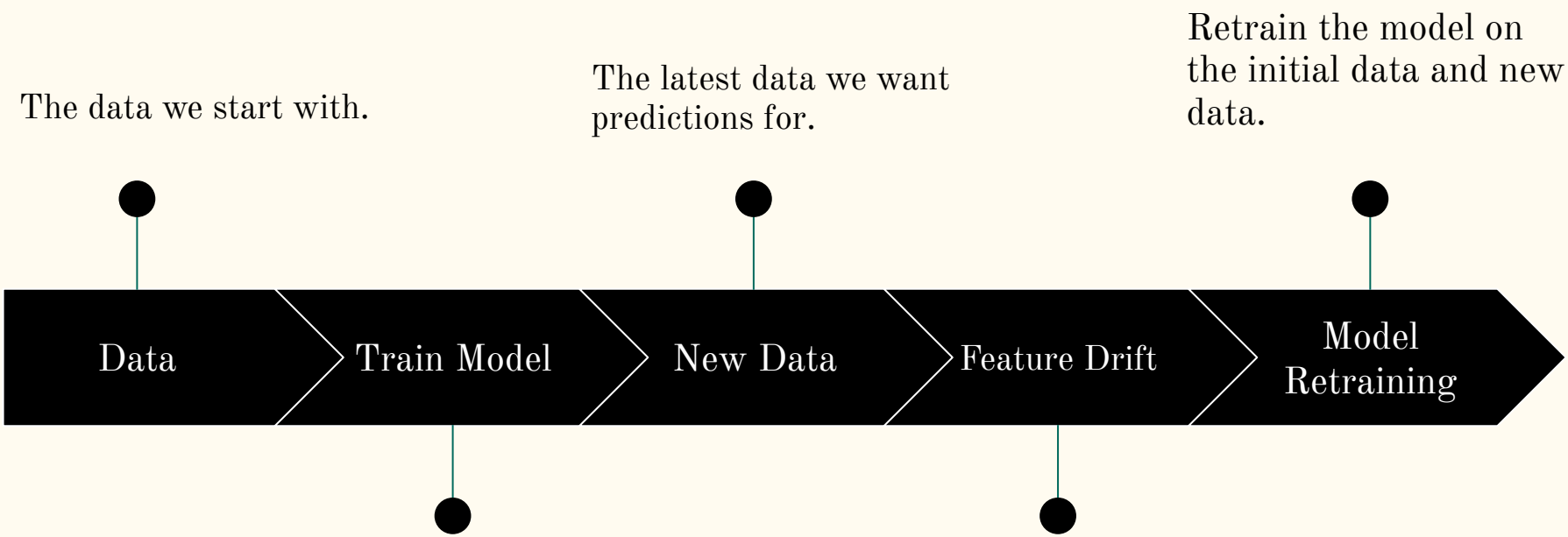
These are the top ten most important indicators of price. When the about product refers to 2160, 3840x2160, or x1 there is an increase in price. The remaining indicators taper off in importance, and also increase the price.

Model Drift

A Kolmogorov-Smirnov test was performed for each column in the data to see if the distribution of the testing data is the same as the training data. If the testing data does not share the same distribution as the training data, then there is a drift, which signals for model retraining. The only column which experienced a drift is actual_price, which is good. The model was retrained to include the test data.

Deployment




The data we start with.
The latest data we want predictions for.
Retrain the model on the initial data and new data.
Data
Train Model
New Data
Feature Drift
Model Retraining
Data wrangling, feature engineering, model training.
See if the distribution of the new data is significantly different than the initial data.

Thank You