Presently Surprised:
A Gift Recommender Model

Capstone by Samuel Ang

Who

I am a Data Scientist with Presently Surprised Co. Ltd.

An online retailer of gifts

Data source: “Online Retail Data Set” from the UCI Machine Learning Repository

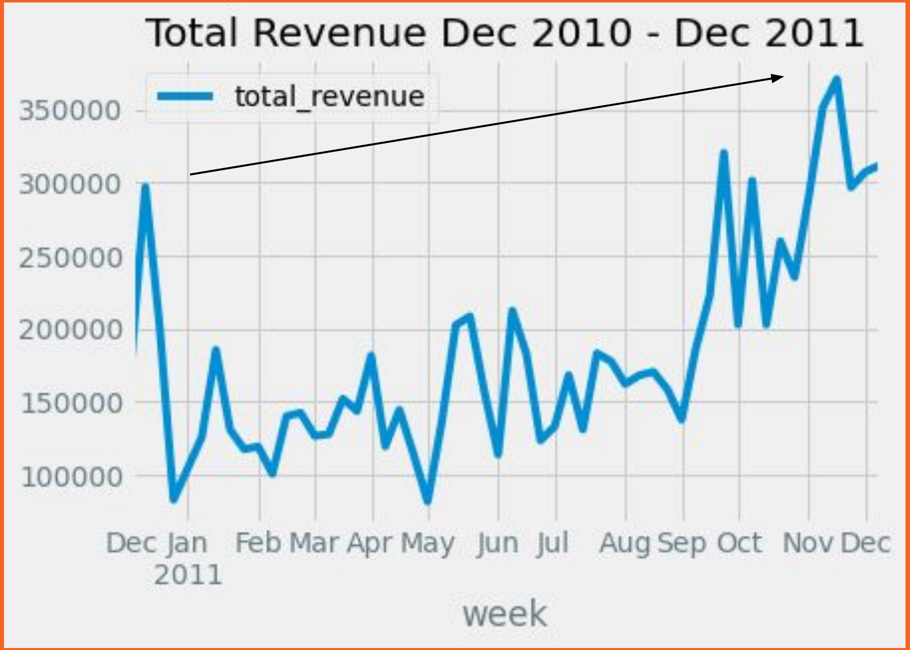
1. Introduction

- **Current performance**
  We did well this year
- **Business Challenges**
  Customer retention issues
- **Data Challenges**
  Cleaning of product codes, aggregation of invoices, lack of features
- **How We Can Help**
  Recommender model to improve our online retail experience

We did well this year

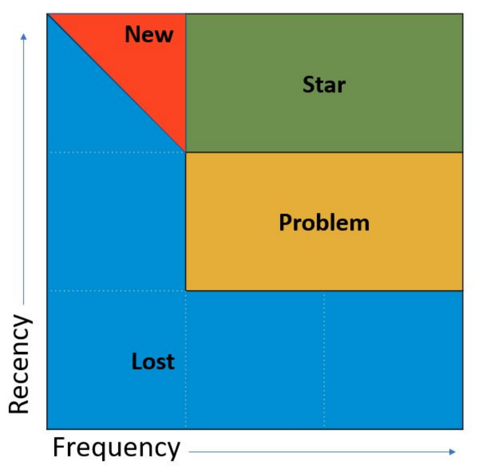
Sales grew by 20% (December peak to peak)

Note: Dataset is limited to this time period, for narrative purposes only



But we lost some customers

- Problem - a significant proportion (~20%) of customers have been lagging behind in purchases
- Star - Top Customers
- New - Customers that only recently started purchasing

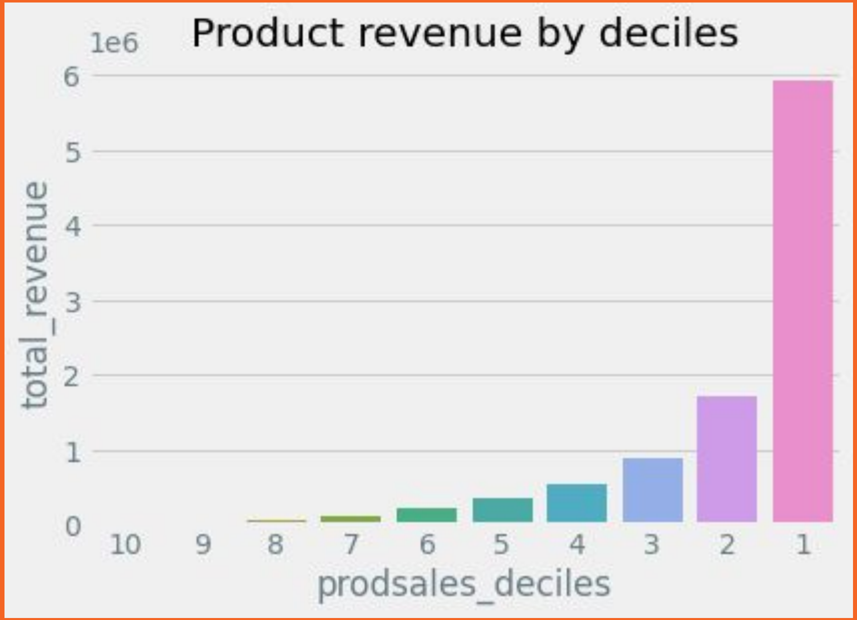


Product distribution

There are over 5000 products in our portfolio and top 40% products account for 92% of sales

Bottom 60% only occurs in less than 10% of purchases

Customer pain point: takes a long time to navigate product list



Data Issues

Products unclassified, varying descriptions per product code

Product code upper and lowercase with same description

Repeated product codes in each invoice

Multiple invoices from same customer with exact same DateTime

Data Cleaning

Select longest description for each product code

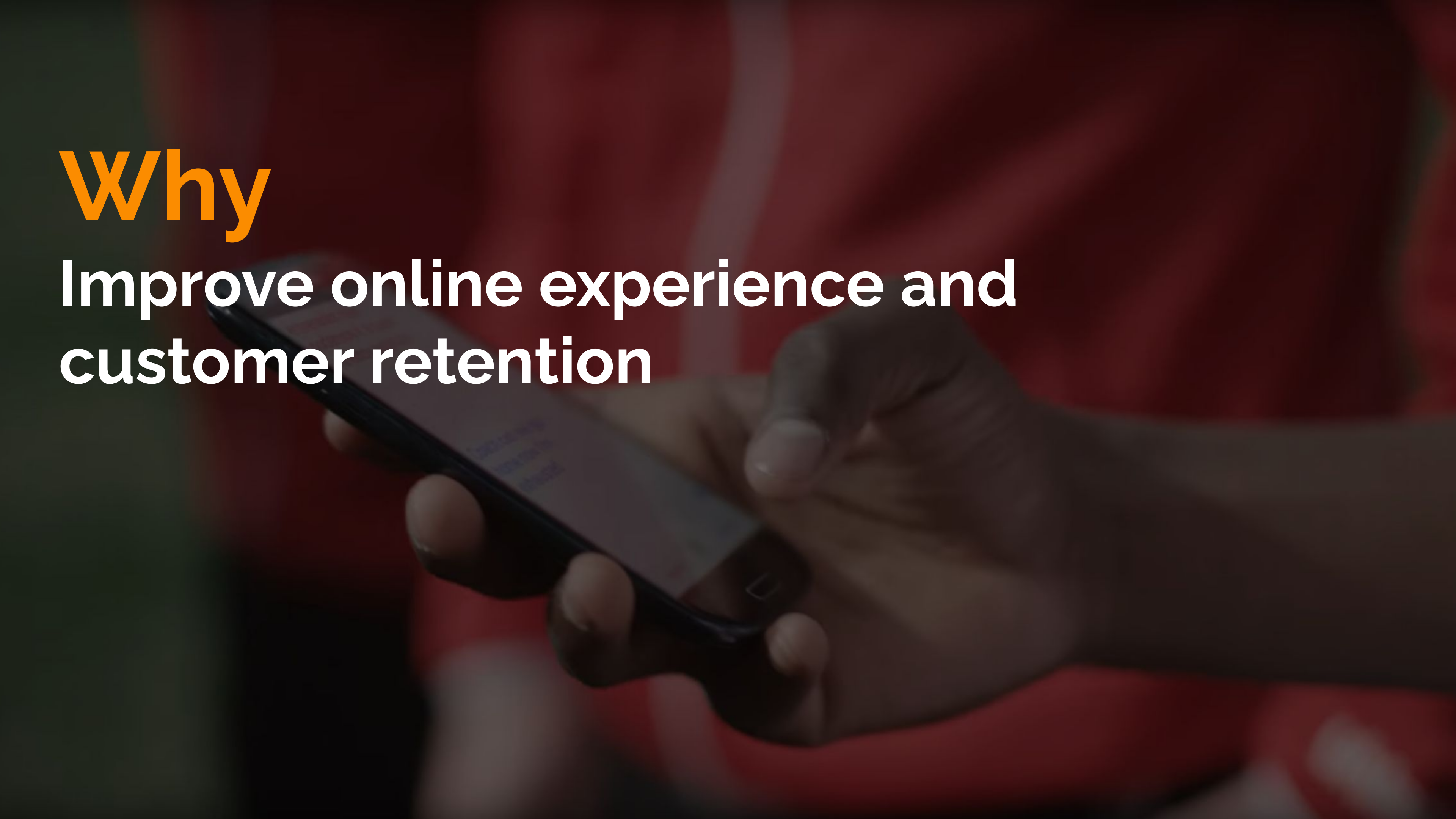
Standardize capitalization

Aggregate sales of repeated product codes in each invoice together

Aggregate invoices with exact DateTime

Why

Improve online experience and customer retention



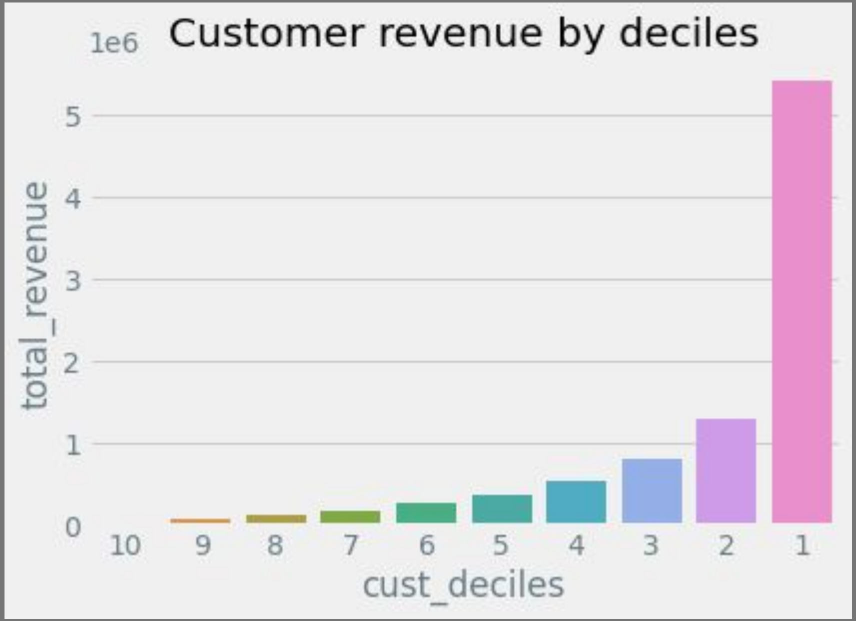
How

To build a **recommender system** to predict the **top 10 products** a customer would buy.

Evaluated by the **hit rate** and **predicted revenue** against the last actual invoice of each customer's purchase in the dataset.

2. Analysis

- **Customer sales distribution**
  Customer sales is skewed by top customers
- **Sales frequency distribution**
  Most customers purchase at least once in 1-2 months
- **Product seasonality**
  Products are the most different between the months of June and December

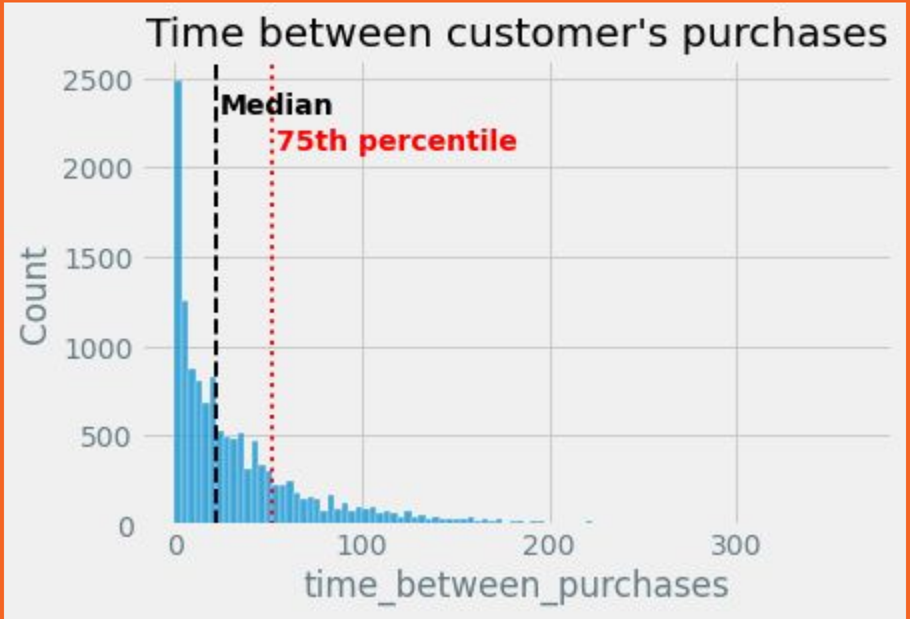


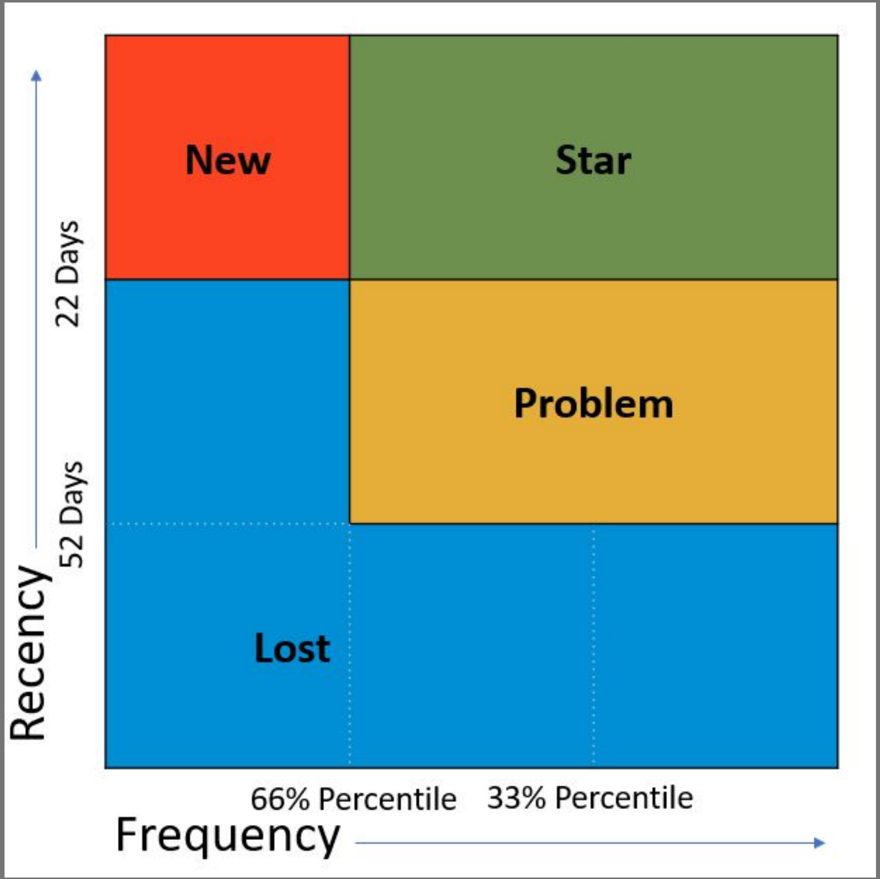
Customer distribution

Top 30% customers account for 83% of sales

Typical time between purchases

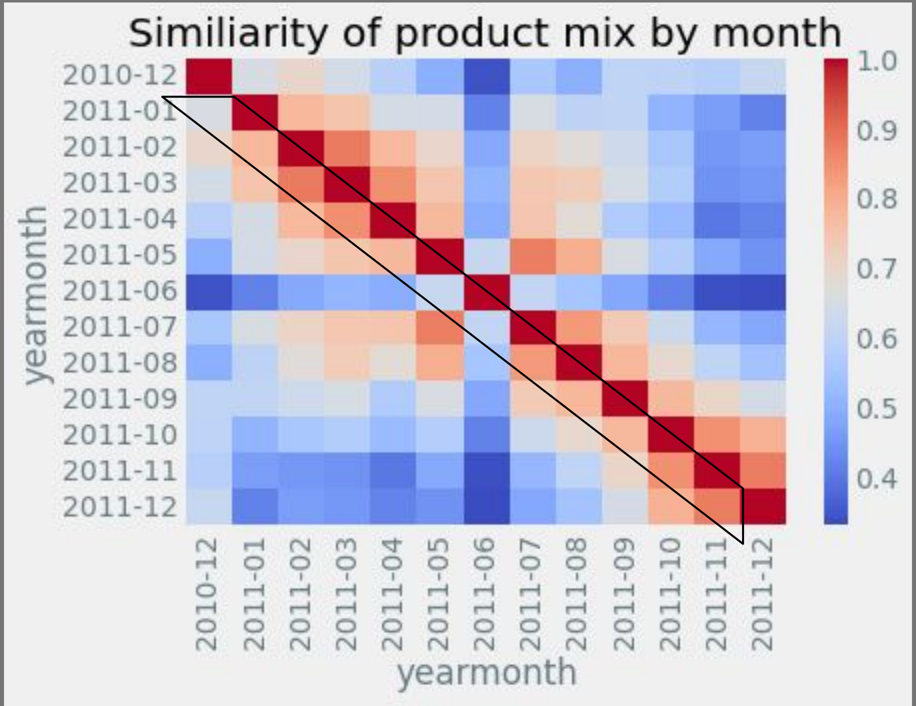
Median of 22 days and 75 Percentile of 51 days





Customer segments revisited - Link to Model

- Customers older than 52 days and frequency
- New: customers that have purchase within 22 days, and have no purchases older than 22 days
- Exclude from model: Lost customers and bottom 33% percentile in Monetary total sales



Seasonal sales

Sales generally are quite similar month to month but the difference gets the largest between June and December

3. Recommender

- ➔ **Model 1**
  Top products from all sales
- ➔ **Model 2**
  Personalized product ranking
- ➔ **Model 3**
  Adaptive nearest neighbour
- ➔ **Model 4**
  Purchase behavior clustering
- ➔ **Model 5**
  Trending machine

Model 1

Top 10 products based on total sales

2 versions:

- Revenue
- Revenue with 1 month decay (half life reduction)

Select **all** prior invoices to the test invoice

Rank revenue by product in descending order and select top 10

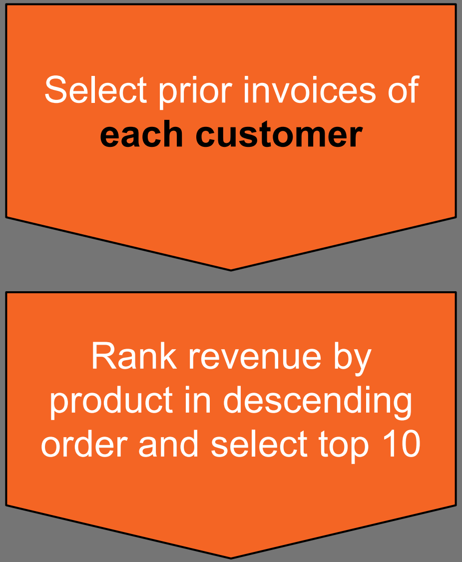
Model 2

Personalized ranking: Top 10 products based on each customer's prior sales

2 versions:

- Revenue
- Revenue with 1 month decay (penalty adjustment based on invoice age)

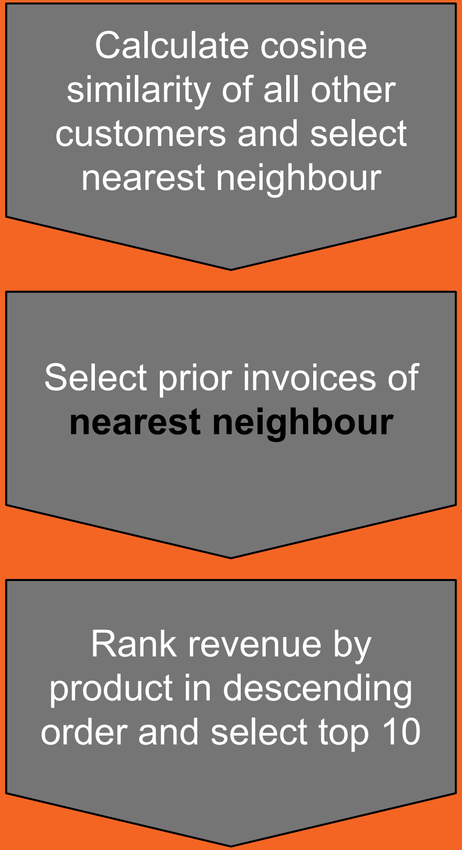
For new customers (no prior sales), Model 1 is used for them



Model 3

Adaptive Nearest Neighbour

- For each customer: finds nearest neighbour based on cosine similarity
- Return top 10 brands
- Revenue with 1 month decay



Model 4

Customer clustering using SVD & Kmeiods

- Number of clusters optimized with silhouette score n=2

```
n_cluster:2 silhouette score:0.06651118490612762
n_cluster:3 silhouette score:-0.1201863411438585
n_cluster:4 silhouette score:-0.43188417219191616
n_cluster:5 silhouette score:-0.5275650029850792
n_cluster:6 silhouette score:-0.6331486289892376
n_cluster:7 silhouette score:-0.8765888269100155
n_cluster:8 silhouette score:-0.833074235329224
```

Identify customers based on product mix in sales before Oct

Select prior invoices of **each cluster** before each test invoice

Rank revenue by product in descending order and select top 10

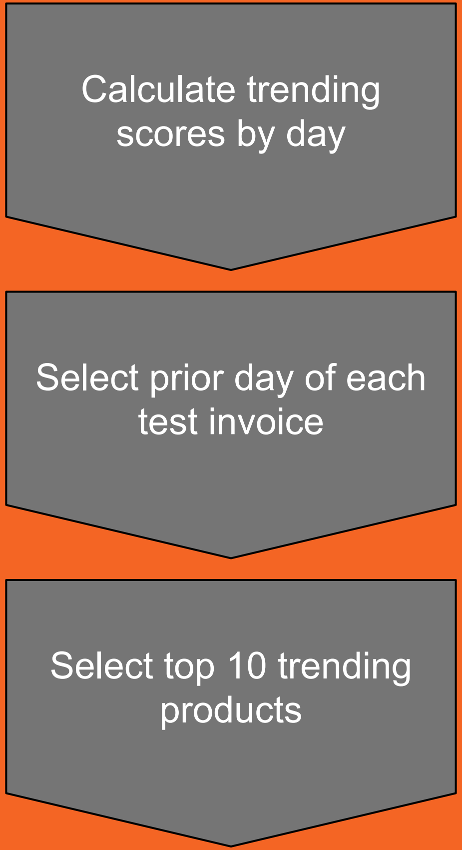
Model 5

Trending machine

- Change in sales by day
  - X-axis= product
  - Y-axis= test invoice age
- Smoothed using Hamming window of 14 days

2 versions:

- All test data
- Only top quartile brands



Evaluation

Nearest Neighbour model is best at revenue prediction, although the hit rate is slightly lower than the ranking model

	hit_rate	predicted_revenue
model		
top10	0.09	102,709.34
top10_1m	0.11	120,728.90
cluster	0.11	125,274.45
ranking	0.14	128,063.58
ranking1m	0.15	142,207.14
nearestneighbour	0.14	149,792.06
trending	0.05	47,825.78
trending1d	0.06	49,723.74
testdata	1.00	3,265,507.54

Evaluation

Adding 1 month decay works: Recency is important

model	hit_rate	predicted_revenue
top10	0.09	102,709.34
top10_1m	0.11	120,728.90
cluster	0.11	125,274.45
ranking	0.14	128,063.58
ranking1m	0.15	142,207.14
nearestneighbour	0.14	149,792.06
trending	0.05	47,825.78
trending1d	0.06	49,723.74
testdata	1.00	3,265,507.54

Evaluation: Aggregative models

Only useful for products in the top decile

Clustering does improve results, even with two just two categories

prodsales_deciles model	1	2	3	4	All
top10	102,709	0	0	0	102,709
top10_1m	120,729	0	0	0	120,729
cluster	125,274	0	0	0	125,274
ranking	110,696	10,521	4,551	2,296	128,064
ranking1m	124,845	10,513	4,553	2,296	142,207
nearestneighbour	143,276	5,093	861	562	149,792
trending	46,912	751	115	48	47,826
trending1d	49,724	0	0	0	49,724
testdata	2,167,185	585,442	316,117	196,764	3,265,508

Evaluation : Individual Models

Improvement in nearest neighbour model comes mainly from customers with purchase intervals between 2-4 weeks.

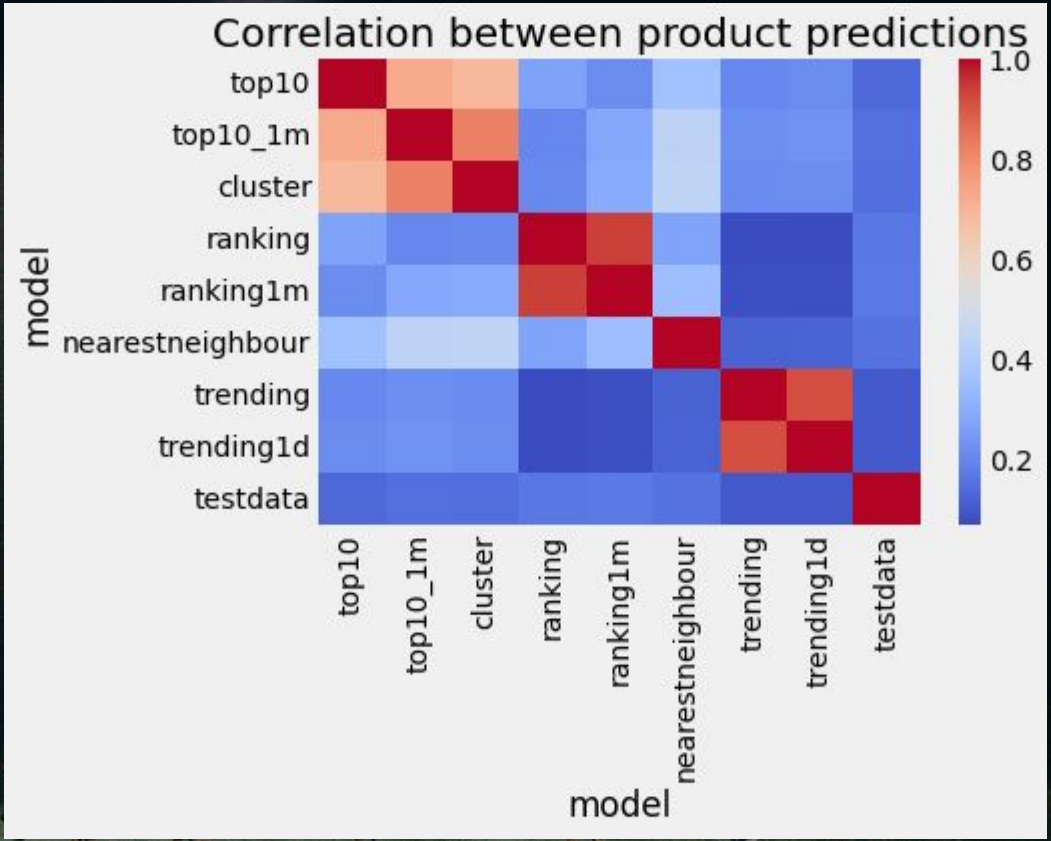
time_intervals	<7days	7-13days	14-20days	21-27days	>28days
model					
top10	14,099	12,049	5,345	3,708	23,061
top10_1m	15,037	12,934	7,037	5,434	21,343
cluster	15,464	13,583	6,477	5,414	25,392
ranking	18,876	12,810	6,794	6,397	38,740
ranking1m	18,780	12,792	6,784	6,397	38,510
nearestneighbour	17,711	18,154	9,428	7,626	37,929
trending	3,754	2,026	1,803	1,611	9,779
trending1d	3,759	2,409	1,870	1,669	9,960
testdata	383,089	273,449	200,262	157,167	862,161

Evaluation

It's clear that recency is important

But somehow ranking is not performing well

Perhaps cluster + trending would work



4. Closing

➔ **Milestones**

We have found a better model - Nearest Neighbour - hit rate improvement by 5 percentage points, or 46% increase in revenue predicted

➔ **Shortcomings**

Hit rate is still quite low. Implies that consecutive purchases by customers are more different than the same

➔ **What’s next?**

Next Steps

Extracting features from product description

Improve clustering of customer by product features

Ensemble method: Product trends by cluster?

Effect of cancellations on sales

Effect of previous purchase on current purchase (penalty coefficient, frequent pattern mining)

Thank you