Demand Forecasting to Maximize Sales

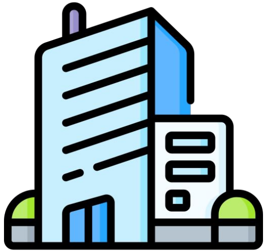
Bekzod Normatov, Cathy Choo, Charlize Tan,
Daniel Zhou, Joonghyun Eo, Marc Herrera

Table of contents

1. Explain the context
2. Patterns we noticed
3. Your naive model
4. Our awesome model
5. Assumptions & limitations
6. Next steps

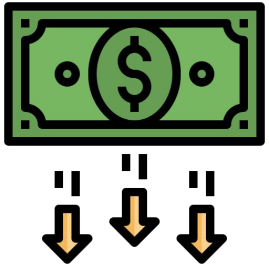
NEXT

This is you!



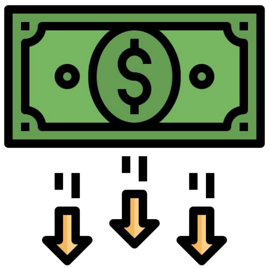
You are the owner of a medium-size grocery

You notice that sometimes you order too much stock, and sometimes too little



You’re losing money and customers are unhappy

Your current forecasting model is causing a loss



Your current model is costing you $10.8M in losses annualized*

Your customers are unhappy and leaving, and you are not maximizing your sales potential

You’re wasting a lot of products and you’re doing a lot of bad for the environment

*Based on naive model

You try our forecasting model and notice that:

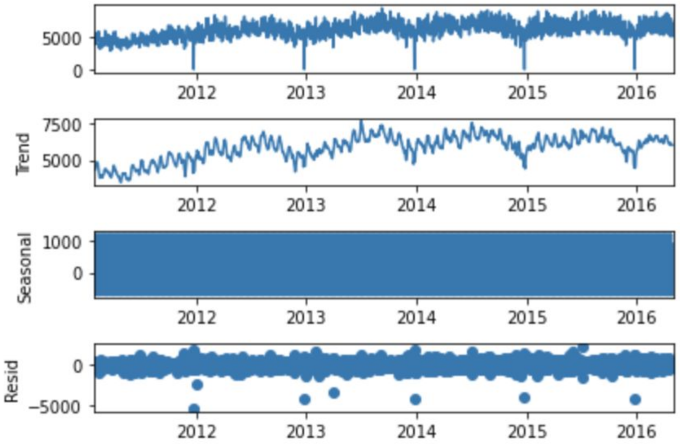
1. **Customer satisfaction has increased.**
   When customers come into the store, the product they are looking for is available.

2. **You don't overstock as often**
   Some items, like food, are perishable, so minimizing the chance of overstocking reduces cost and waste.

3. **You can plan better financially.**
   More accurate revenue and profit projections.

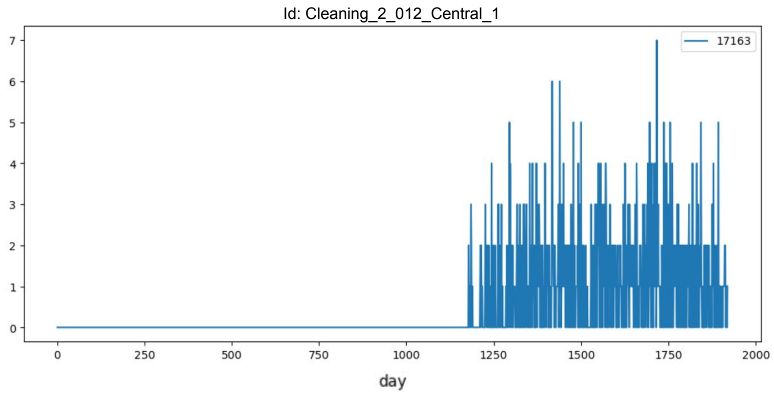
What did we notice?

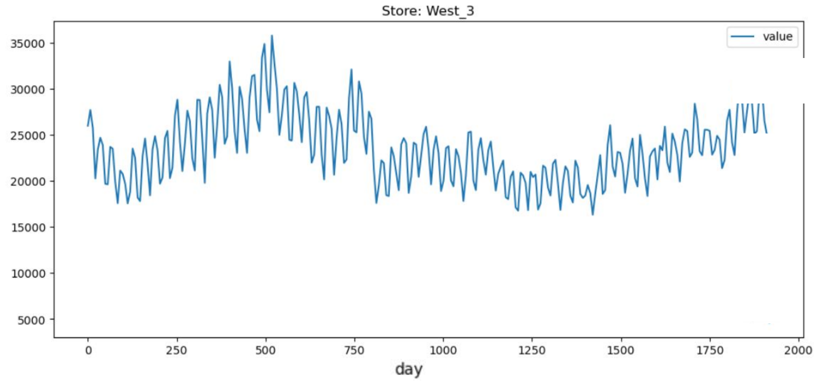
- Positive trend overtime
- High correlation between a day and the previous day's sales
- Seasonality every 7 days across stores and subcategories

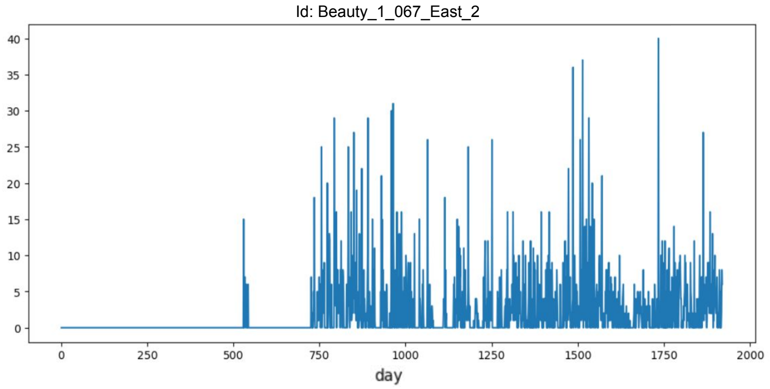


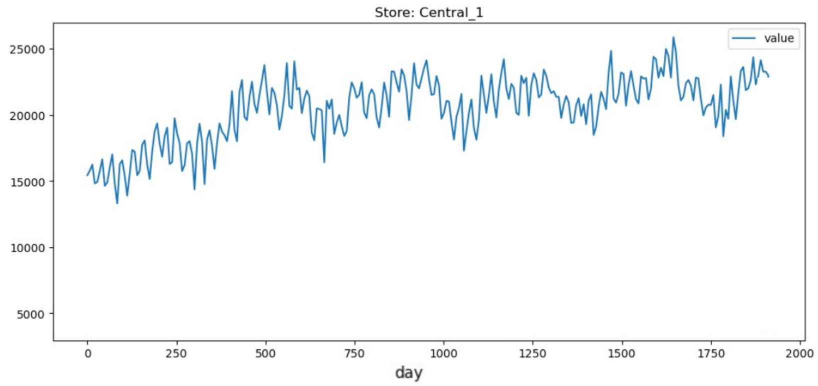
Decomposition for store_id: "East_3" with lag 210

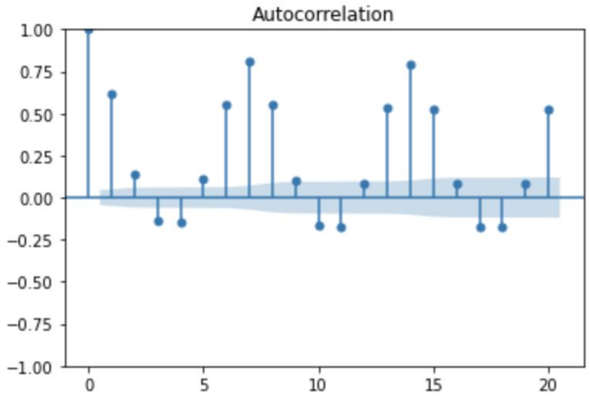
Time series of stores vs. item ids



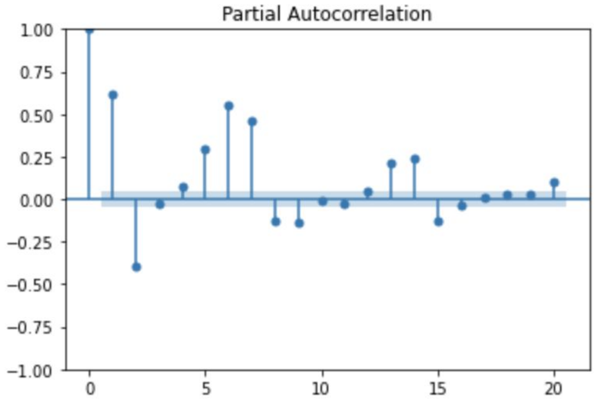




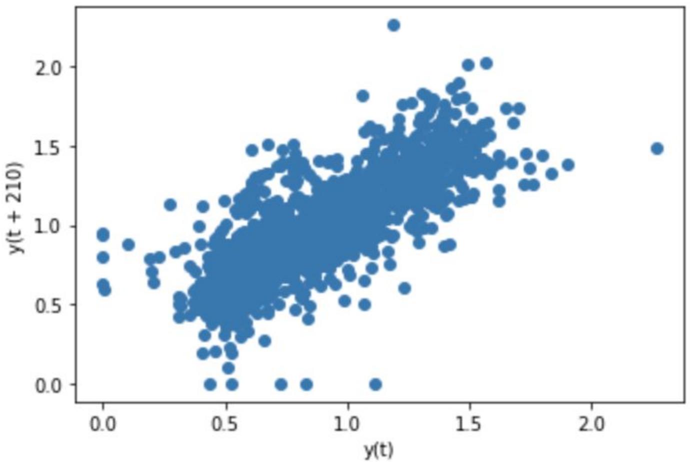




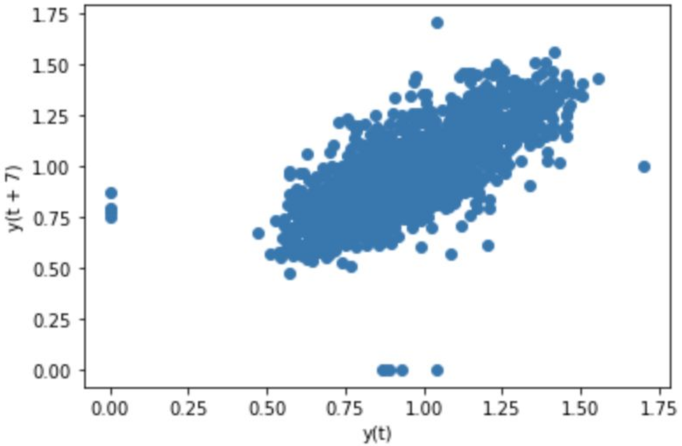
Autocorrelation plot for store_id: “East_1” - notice spike every 7 days



Partial autocorrelation plot for store_id: “East_1” - notice spike every 7 days



Lag plot for store_id: “West_1” with lag 210



Lag plot for store_id: “Central_1” with lag 7

Baseline - Naive Sales Model

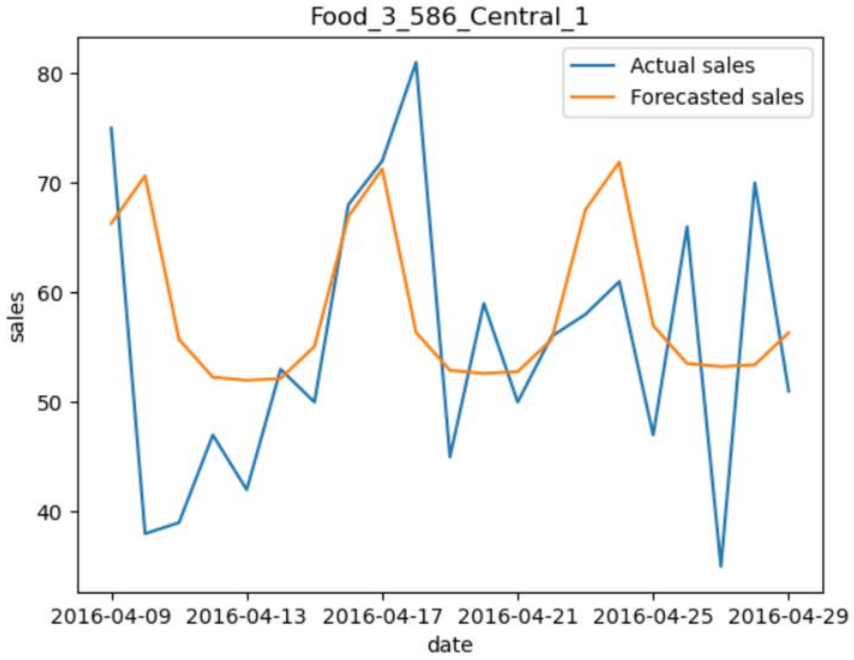
- Let's assume your business currently forecasts sales by taking the average number of goods sold in the past.
- An overprediction of sales results in **extra costs.**
- An underprediction of sales results in **lost potential profit.**

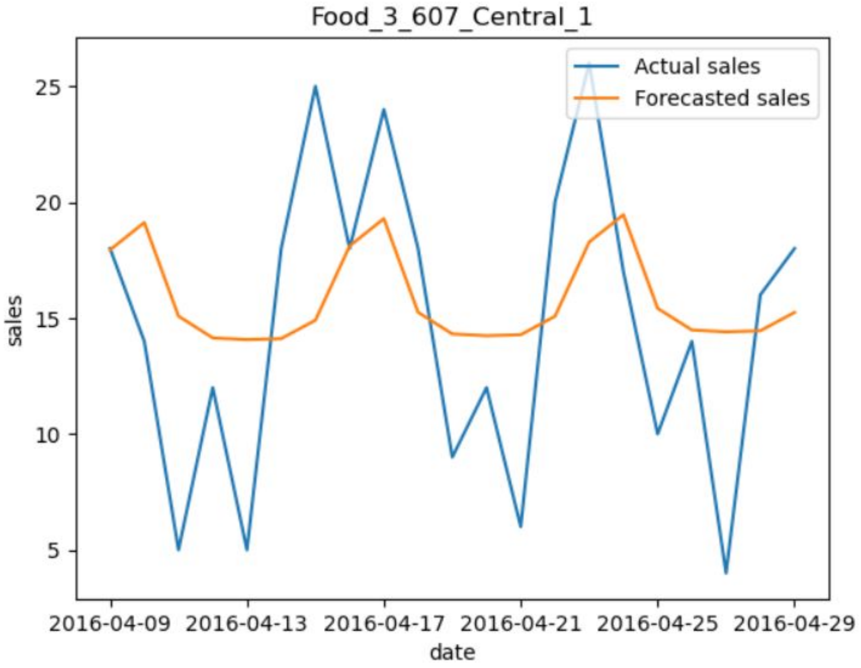
Pros	Cons
• Easy to understand • Easy to implement	• Can't predict spikes

Our model:
Error-Trend-Seasonality

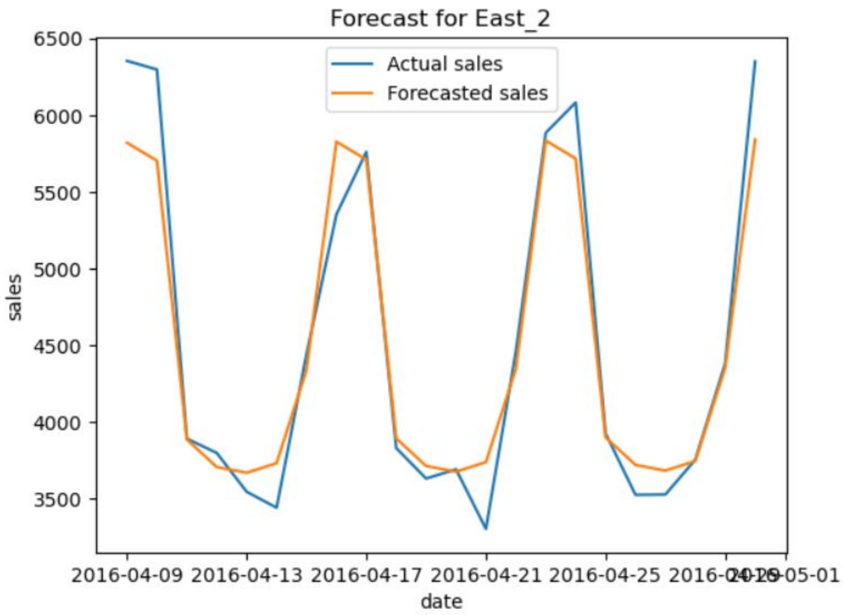
- Considers the trends of the sales of each store
- Takes into account that there is a repeating pattern every 7 days
- Uses the proportion of an item of total sales from the last week in the data
- Results imply ~$6m saved annually compared to baseline

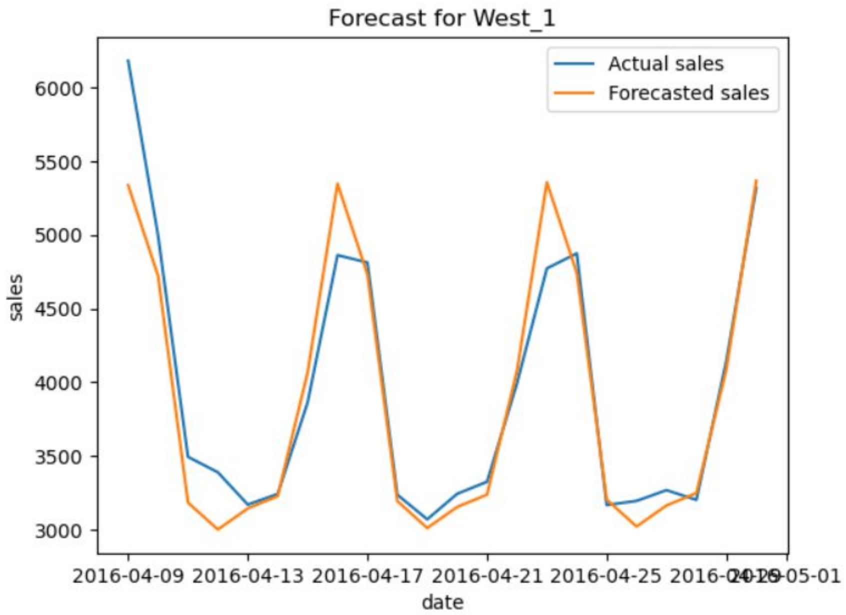
Performance of our model





Performance of our model





There are some limitations

- It might be preferable to be **conservative** or **aggressive** in how much we stock depending on the *kind* of good that we are considering
    - ie. for perishable goods, you may decide to understock rather than overstock (because overstocked goods may expire)
- Inability to account for various levels of errors for different categories
- Inability to handle **seasonality variations**
- Difficulty in forecasting for **long horizons**
- Inability to handle **external factors**
- Inability to account for other potentially **important predictors** such as price of the goods
- Limited ability to capture **sudden changes** in demand

Immediate next steps

- Be more or less conservative with predictions for different types of items
- Incorporate more data that may allow us to build a model that gives insight on what influences sales
    - Regional statistics/features
    - Customer statistics

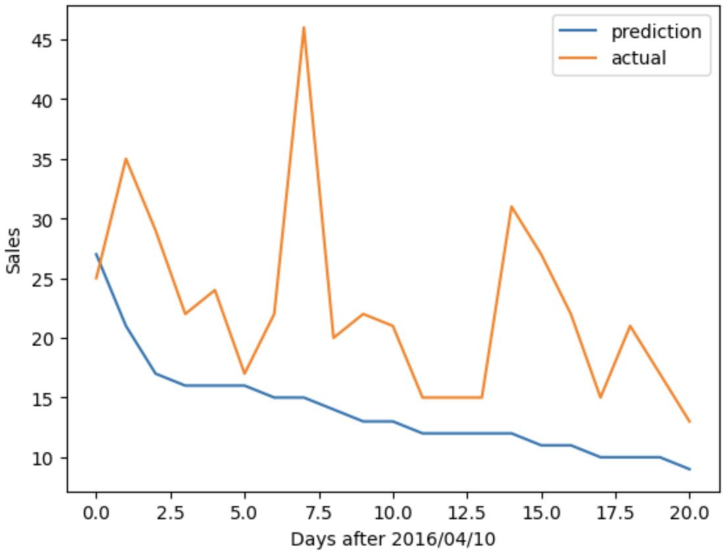
Thank you

Appendix

Performance of our other models

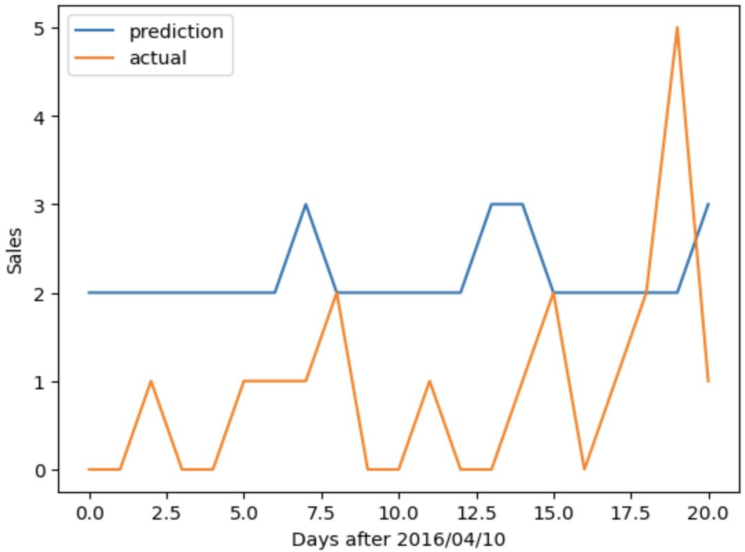
Arima

item: Food_1_122_West_1



XGBoost

item: Food_2_274_East_3



* *predicted sales using Arima and XBoost versus actual predicted sales*