**Technical Report for CS598 Project 2**

**Team Members and Contributions:**

Zilal Eiz Aldin ( [Zelalae2@illinois.edu](mailto:Zelalae2@illinois.edu))- train model, evaluation; Mesay Taye ([mesayst2@illinois.edu](mailto:mesayst2@illinois.edu)) - train model; Jianci Zhai ([jianciz2@illinois.edu](mailto:jianciz2@illinois.edu))- train model and technical report

**Section I: Technical Report**

Data Transformation:

1. We converted relevant data variables from strings to date types and used one-hot encoding to create binary indicators (1/0) for weeks and key holidays.
2. Sales Week Tagging: We identified and tagged the two key sales weeks around Christmas. The Pre-Christmas week includes the week in which December 23 falls, while the Post-Christmas week includes the week containing December 26. This approach accounts for the possibility that Pre-Christmas and Post-Christmas sales weeks may either overlap with or fall in different weeks from Christmas itself.
3. Year Dummy Variables: We created dummy variables for each year to capture intercepts for different years, which will help in analyzing year-over-year trends.

**Exploratory Data Analysis:**

1. **Setting Benchmarks:** before starting modeling, we use naïve approach to predict without utilizing any advanced modeling methods. We use the the sales number for the same store and the same department for the same week from the previous year as prediction, and we got the weighted mean absolute error (WMAE) around 1888.
2. **Seasonality trends:** we observed huge sales increase around Christmas and Thanksgiving, which indicates needs to handle them specially.
3. **Sales Variation by Category and Store**: Large variations in sales across different categories and stores were also observed, suggesting that segmenting by category and store could improve model performance.

**Modeling Process:**

1. First, we stand to create models for each store and category by using package patsy and statsmodels, and we are able to improve the overall WAE to 1,653. After zooming into details, we found the major problems coming from folder 1 and folder 5, both with WAE above 2,300
2. **Variable Selections**: The issue with folder 1 was clear—limited historical data meant that time-related variables were ineffective. To address this, we focused on selecting variables with a strong, direct correlation to sales figures. By filtering out low-correlation variables, we improved folder 1’s performance, reducing the WAE from 2,323 to 1,919. However, applying the same key variable filtering approach to other folders did not yield similar improvements. (See below for performance by folder with varying covariance/correlation thresholds.)A green and yellow chart

   Description automatically generated with medium confidence
3. We also tried to smooth the models with various methods – eventually we are able to improve the models very little by applying SVD among different stores

**Post Processing:**

we took a closer look at the error performance across each folder. This detailed examination revealed two potential strategies to improve performance in folder 5:

* 1. **Adjust Christmas-related Sales**: We observed consistently lower sales predictions from weeks 48 to 51, while week 52 showed significantly higher sales compared to the previous year. Additionally, we noted a shift in the pre-Christmas Eve week, which falls in week 52 in the training data but moves to week 51 in the testing data. Adjusting our model to emphasize week 52 may improve prediction accuracy.
  2. **Adjust Category-Specific Sales Estimates**: We observed a pattern where certain categories consistently outperformed or underperformed relative to our predictions. By adjusting sales estimates for these categories based on observed percentage increases or decreases, we can enhance prediction accuracy.

Ultimately, we decided to proceed with method (b), as it aligns with real-world scenarios where we expect to receive category-level insights from other teams (e.g., the category management team or store teams). Testing confirmed a significant improvement in performance. However, this improvement was only observed in the first six folders. We have also tried to apply a similar adjustment by store level- though but the improvement was not ideal.

**Section II: Performance Metrics**

Microsoft Windows 10 Enterprise; System Model: HP ZBook Firefly 14 inch G8 Mobile Workstation PC

Processor(s): Intel64 Family 6 Model 140 Stepping 1 GenuineIntel ~2611 Mhz

Python Version: Python 3.12.4

Model Performance Summary

A white background with black lines and numbers

Description automatically generated with medium confidence