Optimizing Inventory of Perishable Goods by Forecasting Daily Sales

Team 2

111078505 Hsuan-Jung Lin (Tiffany)
111078513 Yuan-Jie Fong (Francis)
111078518 Kai-Yi Chu (Kyle)
111078519 Chieh-Jui Ho (Jerry)

Summary

Business Problem

Retailers often grapple with the challenge of balancing inventory levels, especially for perishable goods. Overstocking leads to waste and financial loss, while understocking can result in missed sales opportunities. Our focus is on providing forecast suggestions for effective inventory management that is crucial for financial success, sustainability, and maintaining a positive brand image among eco-conscious consumers.

The project aims to optimize inventory management at a Maryland store of a multinational retail chain by forecasting daily sales of the three top-selling perishable goods sold in-store. The forecasts are to be generated on a weekly basis to guide inventory decisions.

Data

We have utilized daily sales data from a leading multinational retail chain in the USA, provided by Nuqleouse, a retail technology solutions firm. The analysis concentrates on the three top-selling perishable items sold through the in-store channel at the Maryland location.

Forecasting Solution

Our approach involves advanced analytics to forecast daily sales on a weekly basis. This predictive model allows for more accurate inventory planning, minimizing the risk of overstocking and the associated costs and waste. The forecasting is sensitive to over-forecasting risks, which is crucial for perishable goods due to their limited shelf life and the financial implications of unsold stock.

Recommendations

To maximize the effectiveness of our advanced analytics forecasting model for perishable goods inventory management, it is crucial for the store to maintain a robust and consistent data collection process to prevent inaccuracies due to missing data. Equally important is the implementation of a digital inventory record system that is updated in real-time, ensuring seamless integration with our weekly forecasting updates. By adopting this forecasting solution, the store can expect to see a reduction in waste, improved financial performance, and a boost in its reputation among eco-conscious consumers.

Problem description

Business goal

Retail store inventory managers have a significant opportunity to improve their inventory management processes, especially for perishable goods. By focusing on efficient inventory management, they can reduce waste, align with sustainable business practices, and enhance their brand's reputation among environmentally conscious consumers. The primary objective is to enhance inventory turnover rates, which will help alleviate financial constraints and prevent issues related to overstocking. To achieve this, store inventory managers should consider using daily sales forecasts. These forecasts can inform inventory management decisions, ensuring that the stock levels are optimized to meet consumer demand without excess, thereby promoting a more efficient and environmentally friendly approach to inventory management.

We chose three products from the in-store channel of the Maryland store because it has the highest sales volume. By focusing on the top three products in their most popular consumer channel, we aim to identify specific trends using a substantial amount of data. Once we can ascertain the trends in these high-sales-volume data, theoretically, there's a possibility to apply these findings to the in-store performance of these three products at other stores.

Forecasting goal

Forecast the daily sales for the three highest-sales perish goods through the in-store channel at a Maryland store, over a period of one week.

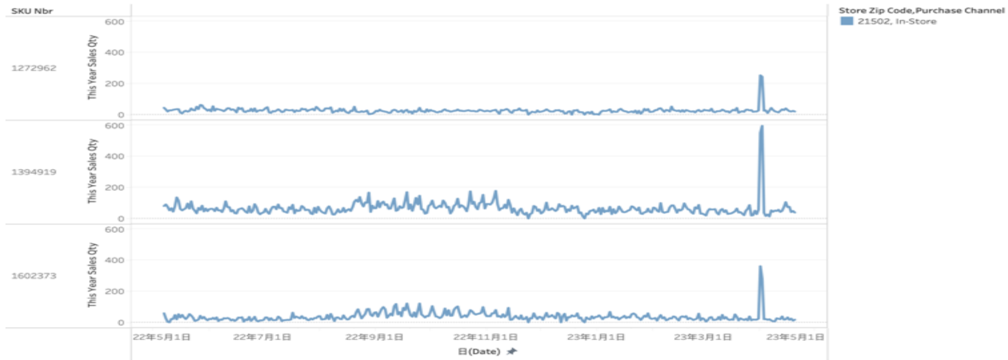
Data description

We have received the data from Nuqleous, a company specializing in retail technology solutions. Their expertise lies in enhancing agility for prominent brands and offering scalable tools for smaller companies. Our analysis concentrates on the daily sales data from a Maryland store, covering the period from January 1, 2022, to April 20, 2023. The key metric in our study is the Sales Quantity, specifically from the in-store channel. This data provides us with valuable insights into consumer behavior and sales trends, enabling us to refine our retail strategies effectively.

Sample of 10 rows per series

Date <chr>	fc_avg.1272962 <dbl>	fc_avg.1394919 <dbl>	fc_avg.1602373 <dbl>
2022-01-01	5	93	18
2022-01-02	12	76	21
2022-01-03	13	94	30
2022-01-04	0	103	62
2022-01-05	23	129	59
2022-01-06	18	45	23
2022-01-07	39	121	91
2022-01-08	2	89	61
2022-01-09	27	99	22
2022-01-10	36	68	20

Time plot



Data preparation

Missing data: The missing data is predicted by using a moving average, which is calculated by averaging the data from the same day of the week over four weeks, encompassing the month before and after the missing data point.

Outliers: For re-estimating the outliers on specific dates (2023/4/1,4/2 & 2022/4/1,4/2), a moving average method is utilized. This approach involves recalculating the values for these outliers based on the sales data from the same days in the previous year.

Time plot after data preprocessing

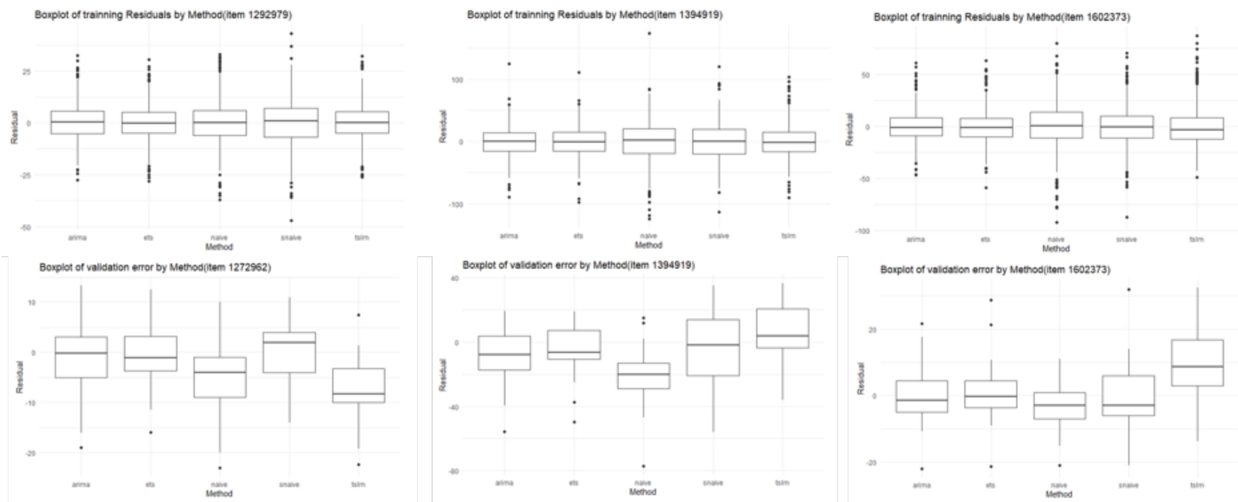
Forecasting solution

We employ five forecasting methods to analyze three products. Among these, the naive and snaive methods serve as benchmarks. The remaining three methods – ETS, TSML, and ARIMA – are utilized to determine the most effective model for these products. Each model incorporates various parameter combinations. The optimal parameters for each of these methods (ETS, TSML, and ARIMA) in relation to the products are detailed below.

To prioritize concerns about over-forecasting, we utilize box plots in both training and validation phases to determine the most suitable model for each product. For product 1272962, while TSML exhibits the lowest over-forecasting, it predominantly under-forecasts significantly, and its median error is not close to zero. In contrast, ETS demonstrates a more balanced error distribution, with errors mostly ranging between plus and minus zero, and importantly, it avoids severe over-forecasting.

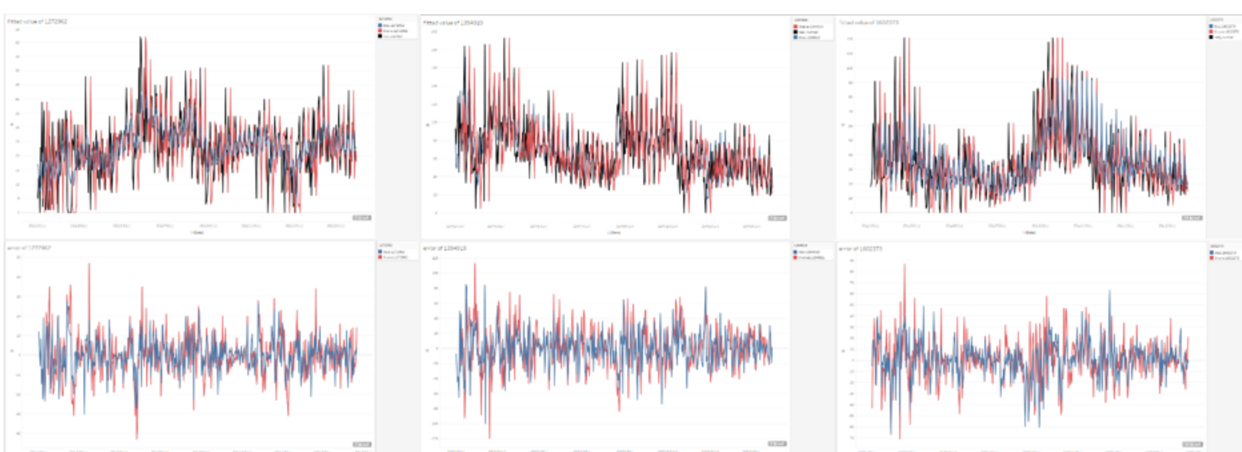
For products 1394919 and 1602373, ETS consistently shows the best performance. The errors for these products are closely clustered around zero, indicating accurate forecasting with no serious instances of over-forecasting.

Comparative Box Plots: Training vs. Validation Error Across Different Products



The provided picture illustrates a performance line chart for all products, showcasing the outcomes using the best-selected model alongside the benchmark model (snaive), complemented by their corresponding error line charts. From this visualization, it is evident that the ETS model excels in capturing the actual trends of the commodities. Notably, in certain segments, ETS even surpasses the performance of the snaive benchmark, demonstrating its effectiveness in accurate trend prediction.

Comparative Line Chart: Fit and Error Across Different Products



Conclusions

Advantages: The adoption of our advanced analytics forecasting model offers significant benefits for managing the inventory of perishable goods. It enhances inventory precision, thereby reducing overstocking and waste, which is in line with sustainable practices and can boost the brand's image among eco-aware consumers. The model's ability to forecast sales weekly enables the store to adjust more effectively to demand changes, ensuring a supply that matches consumer buying trends. This not only cuts down on financial losses from unsold items but also heightens customer satisfaction by lowering the incidence of out-of-stock situations.

Limitations: Despite these benefits, the model has certain limitations that must be considered. The forecasting accuracy is highly dependent on the quality and completeness of the input data, and inaccuracies or omissions can lead to suboptimal forecasting outcomes. This is particularly true for products that may have a higher sensitivity to data quality issues. Furthermore, the model does not currently take into account variations in demand due to holidays or special events, which can significantly impact consumer purchasing behavior. Another limitation identified is the strong day-of-week effect, as evidenced by the performance matrix in the Appendix 5. Specifically, the model shows larger errors for Saturdays and Sundays, indicating that the forecast performance for weekends is less reliable than for weekdays. For a more tailored set of recommendations, further information would be beneficial, such as the specific nature of the perishable goods, seasonal demand variations, or details on the current inventory management processes.

Operational Recommendations:

1. **Data Integrity:** Implement rigorous data collection processes to supply the forecasting model with accurate data. Regular audits should be conducted to identify and correct any data issues.
2. **Technology Integration:** Invest in digital inventory systems that can integrate with the forecasting model and support real-time updates to reflect sales and inventory levels precisely.
3. **Continuous Improvement:** Set up a routine for ongoing review and enhancement of the forecasting model, including an analysis of forecast-to-actual sales variances to refine its predictive accuracy.
4. **Contingency Planning:** Formulate backup plans for instances where the model might underperform, such as unexpected events or peak demand periods.
5. **Special Event Consideration:** Modify the forecasting model to include variables for holidays and special events, which can be critical in predicting surges or dips in demand.

By incorporating these operational measures, the store can maximize the benefits of the forecasting model, striking a balance between efficient inventory management and the goals of financial success and environmental responsibility.

Appendix

Appendix 1: Models used for Product ID 1272962

```
### 2-1-1 Naive

```{r}
naive.1272962 <- train.1272962 |>
  model(NAIVE(fc_avg.1272962))
fc.naive.1272962 <- naive.1272962 |> forecast(valid.1272962)
```

2-1-2 SNAIVE

```{r}
snaive.1272962 <- train.1272962 |>
  model(SNAIVE(fc_avg.1272962))
fc.snaive.1272962 <- snaive.1272962 |> forecast(valid.1272962)
```

2-1-3 ETS

```{r}
ets.1272962 <- train.1272962 |>
  model(ETS(fc_avg.1272962 ~ error('A') + trend('A') + season('A')))
fc.ets.1272962 <- ets.1272962 |> forecast(valid.1272962)
```

2-1-4 TSLM

```{r}
tslm.1272962 <- train.1272962 |>
  model(TSLM(fc_avg.1272962 ~   trend() + I(trend()^2) + fourier(K=4, period=56) +
             fourier(K=3, period=7)))
fc.tslm.1272962 <- tslm.1272962 |> forecast(valid.1272962)
```

2-1-5 ARIMA

```{r}
arima.1272962 <- train.1272962 |>
  model(ARIMA(fc_avg.1272962, stepwise = FALSE))
fc.arima.1272962 <- arima.1272962 |> forecast(valid.1272962)
```
```

Appendix 2: Models used for Product ID 1394919

```
### 2-2-1 Naive

```{r}
naive.1394919 <- train.1394919 |>
  model(NAIVE(fc_avg.1394919))
fc.naive.1394919 <- naive.1394919 |> forecast(valid.1394919)
```

2-2-2 SNAIVE

```{r}
snaive.1394919 <- train.1394919 |>
  model(SNAIVE(fc_avg.1394919))
fc.snaive.1394919 <- snaive.1394919 |> forecast(valid.1394919)
```

2-2-3 ETS

```{r}
ets.1394919 <- train.1394919 |>
  model(ETS(fc_avg.1394919 ~ error('A') + trend('N') + season('A')))
fc.ets.1394919 <- ets.1394919 |> forecast(valid.1394919)
```

2-2-4 TSLM

```{r}
tslm.1394919 <- train.1394919 |>
  model(TSLM(fc_avg.1394919 ~   trend() + I(trend()^2) + fourier(K=4, period=56) +
             fourier(K=3, period=7)))
fc.tslm.1394919 <- tslm.1394919 |> forecast(valid.1394919)
```

2-2-5 ARIMA

```{r}
arima.1394919 <- train.1394919 |>
  model(ARIMA(fc_avg.1394919, stepwise = FALSE))
fc.arima.1394919 <- arima.1394919 |> forecast(valid.1394919)
```
```

Appendix 3: Models used for Product ID 1602373

````
### 2-3-1 Naive

```{r}
naive.1602373 <- train.1602373 |>
  model(NAIVE(fc_avg.1602373))
fc.naive.1602373 <- naive.1602373 |> forecast(valid.1602373)
```

### 2-3-2 SNAIVE

```{r}
snaive.1602373 <- train.1602373 |>
  model(SNAIVE(fc_avg.1602373))
fc.snaive.1602373 <- snaive.1602373 |> forecast(valid.1602373)
```

### 2-3-3 ETS

```{r}
ets.1602373 <- train.1602373 |>
  model(ETS(fc_avg.1602373 ~ error('A') + trend('N') + season('A')))
fc.ets.1602373 <- ets.1602373 |> forecast(valid.1602373)
```

### 2-3-4 TSLM

```{r}
tslm.1602373 <- train.1602373 |>
  model(TSLM(fc_avg.1602373 ~   trend() + I(trend()^2) + fourier(K=4, period=56) +
              fourier(K=3, period=7)))
fc.tslm.1602373 <- tslm.1602373 |> forecast(valid.1602373)
```

### 2-3-5 ARIMA

```{r}
arima.1602373 <- train.1602373 |>
  model(ARIMA(fc_avg.1602373, stepwise = FALSE))
fc.arima.1602373 <- arima.1602373 |> forecast(valid.1602373)
```
````

Appendix 4: Models Used in Each Products

.model <chr>	ME <dbl>	RMSE <dbl>
NAIVE(fc_avg.1272962)	5.2857143	9.172942
SNAIVE(fc_avg.1272962)	0.0000000	6.928203
ETS(fc_avg.1272962 ~ error("A") + trend("A") + season("A"))	0.7709845	6.606024
TSLM(fc_avg.1272962 ~ trend() + I(trend()^2) + fourier(K = 4, \n period = 56) + fourier(K = 3, period = 7))	7.4040432	10.205142
ARIMA(fc_avg.1272962, stepwise = FALSE)	1.3185688	7.574939

.model <chr>	ME <dbl>	RMSE <dbl>
NAIVE(fc_avg.1394919)	22.238095	30.09430
SNAIVE(fc_avg.1394919)	5.095238	24.22022
ETS(fc_avg.1394919 ~ error("A") + trend("N") + season("A"))	6.359690	18.25262
TSLM(fc_avg.1394919 ~ trend() + I(trend()^2) + fourier(K = 4, \n period = 56) + fourier(K = 3, period = 7))	-5.657362	19.72141
ARIMA(fc_avg.1394919, stepwise = FALSE)	9.088007	19.89973

.model <chr>	ME <dbl>	RMSE <dbl>
NAIVE(fc_avg.1602373)	-2.207506e-03	22.07178
SNAIVE(fc_avg.1602373)	-3.713647e-01	20.17219
ETS(fc_avg.1602373 ~ error("A") + trend("N") + season("A"))	-4.441112e-01	15.84132
TSLM(fc_avg.1602373 ~ trend() + I(trend()^2) + fourier(K = 4, \n period = 56) + fourier(K = 3, period = 7))	4.248762e-15	19.76299
ARIMA(fc_avg.1602373, stepwise = FALSE)	5.423311e-03	15.83355

This figure presents the optimal parameters specific to each method for every product. It also includes the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) for each method-product combination. However, it's important to note that despite displaying these error metrics, we ultimately do not use MAE and RMSE as the primary criteria for evaluating performance.

Appendix 5: Day-of-Week Forecasting Performance for Product ID 1394919

```
$Sun
# A tibble: 1 × 10
  .model        .type     ME  RMSE   MAE   MPE  MAPE  MASE RMSSE   ACF1
  <chr>         <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1 model_formula Test   -27.5  29.0  27.5 -47.3  47.3   NaN   NaN -0.536

$Mon
# A tibble: 1 × 10
  .model        .type     ME  RMSE   MAE   MPE  MAPE  MASE RMSSE    ACF1
  <chr>         <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>   <dbl>
1 model_formula Test    4.86  17.0  13.1 0.952  23.0   NaN   NaN -0.0358

$Tue
# A tibble: 1 × 10
  .model        .type     ME  RMSE   MAE   MPE  MAPE  MASE RMSSE   ACF1
  <chr>         <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1 model_formula Test   -5.22  17.6  14.0 -50.4  66.0   NaN   NaN -0.398

$Wed
# A tibble: 1 × 10
  .model        .type     ME  RMSE   MAE   MPE  MAPE  MASE RMSSE    ACF1
  <chr>         <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>   <dbl>
1 model_formula Test   -1.07  9.88  8.54 -10.5  26.5   NaN   NaN -0.0384

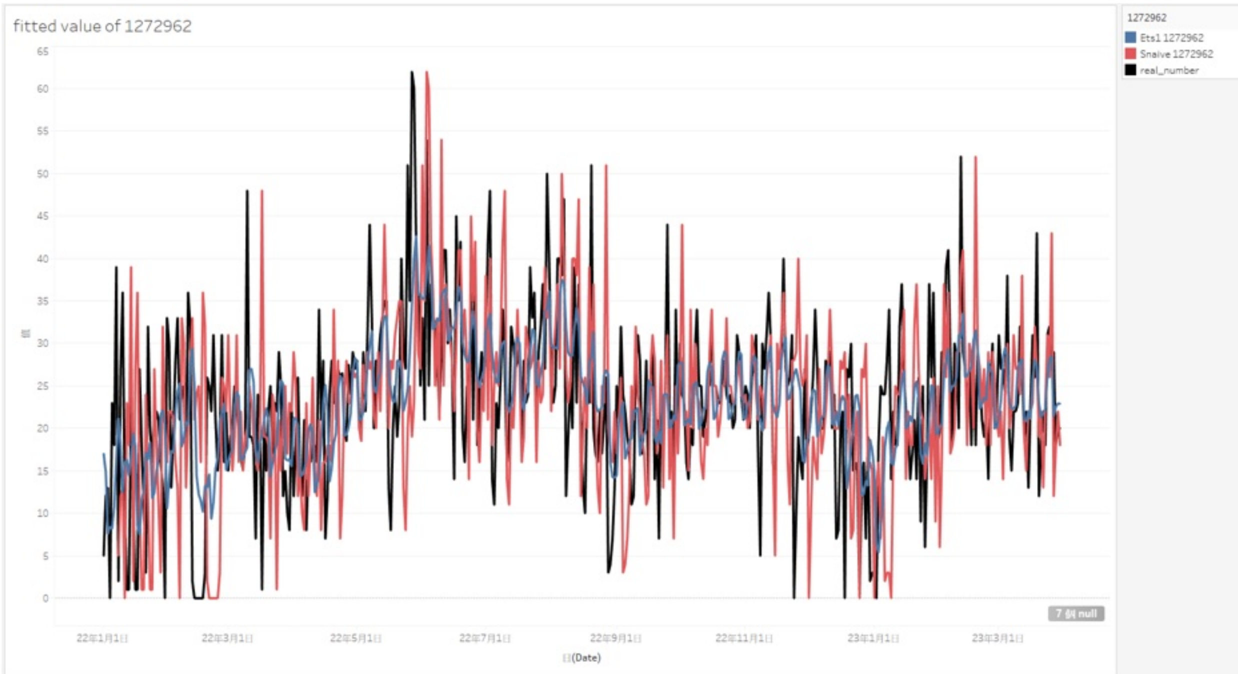
$Thu
# A tibble: 1 × 10
  .model        .type     ME  RMSE   MAE   MPE  MAPE  MASE RMSSE   ACF1
  <chr>         <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1 model_formula Test    2.18  15.7  14.8 -29.7  67.4   NaN   NaN -0.341

$Fri
# A tibble: 1 × 10
  .model        .type     ME  RMSE   MAE   MPE  MAPE  MASE RMSSE   ACF1
  <chr>         <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1 model_formula Test    15.8  18.7  15.8  24.1  24.1   NaN   NaN -0.271

$Sat
# A tibble: 1 × 10
  .model        .type     ME  RMSE   MAE   MPE  MAPE  MASE RMSSE   ACF1
  <chr>         <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
1 model_formula Test    16.8  33.4  21.5  13.0  23.6   NaN   NaN -0.200
```

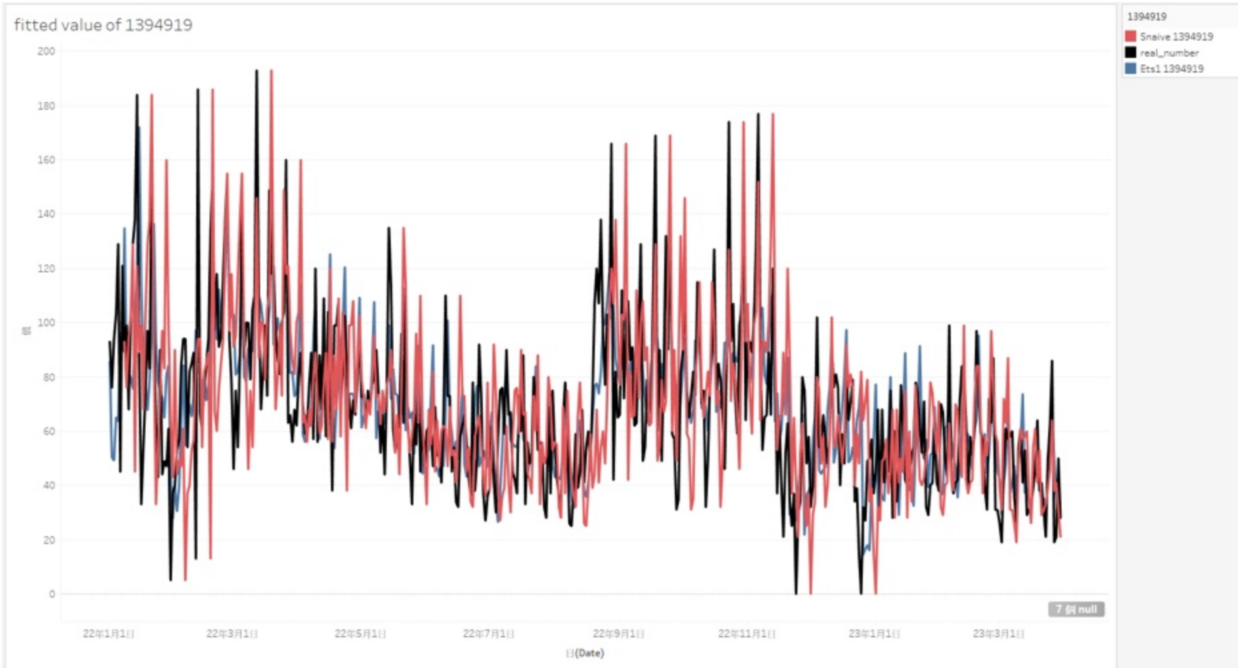
The table presents the performance metrics for the TSLM forecasting model applied to product ID 1394919. The data indicates that the forecasting model's performance varies significantly across different days of the week, with the largest errors observed on the weekends, particularly on Saturdays and Sundays. To address the high errors on weekends, suggestions include incorporating day-specific demand drivers or employing different modeling techniques for weekend data.

Appendix 6: Fit value in best model and snaive for Product ID 1272962

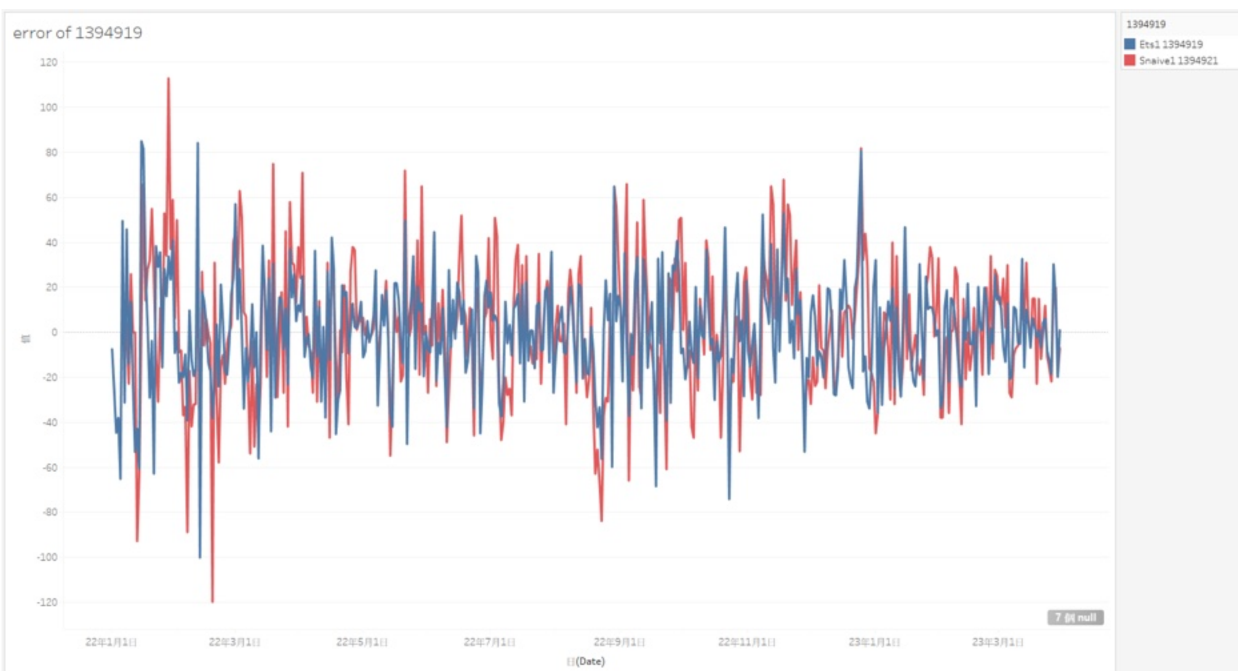


Appendix 7: Error in best model and snaive for Product ID 1272962

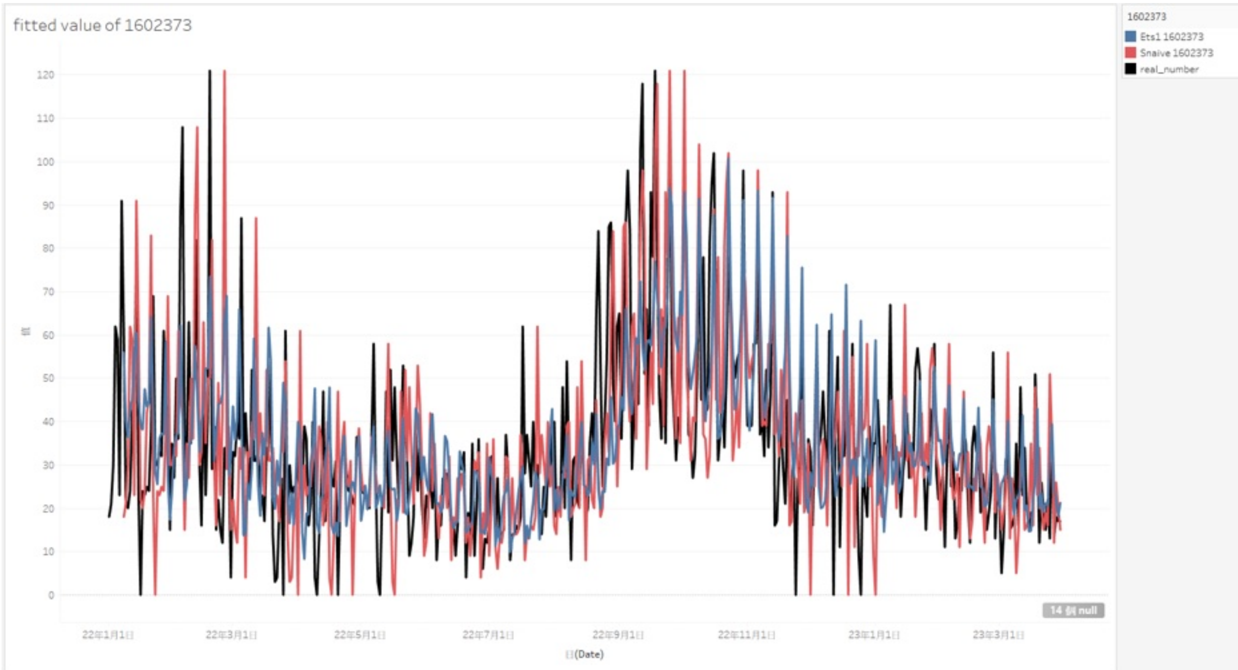
Appendix 8: Fit value in best model and snaive for Product ID 1394919



Appendix 9: Error in best model and snaive for Product ID 1394919



Appendix 10: Fit value in best model and snaive for Product ID 1602373



Appendix 11: Error in best model and snaive for Product ID 1602373