Inventory Management Forecasting

Edwin I. Quijano

Wednesday, December 10, 2025

Table of contents

Chapter I

Introduction

This project focuses on understanding and forecasting warehouse space utilization driven by daily stock transfer activity across multiple product categories. Using transactional inventory data from 2023, the analysis converts product-level information such as quantities and physical dimensions, into a consistent measure of space used in cubic meters. This approach moves beyond traditional unit-based inventory analysis and instead centers on physical capacity, which is a critical constraint in warehouse and distribution operations. By aggregating space usage over time and by product category, the project provides visibility into how operational flows translate into real space requirements and how different product groups contribute to overall capacity demand.

To support proactive planning and decision-making, the project applies time series forecasting methods tailored to both aggregate and category-level behavior. Seasonal ARIMA (SARIMA) models are used to capture trends and weekly seasonality in total warehouse space usage, while Vector Autoregression (VAR) models are employed to analyze and forecast the interdependent dynamics among product categories. Together, these models enable short- to medium-term forecasts that reflect both system-wide patterns and category-specific interactions. The resulting framework offers a data-driven foundation for improving warehouse capacity planning, labor scheduling, and inventory flow management, while also establishing a scalable methodology that can be extended with additional data or more advanced forecasting techniques in the future.

Global Product Inventory Dataset 2025

Published on Kaggle by Keyush Nisar.

Dataset link: https://www.kaggle.com/datasets/keyushnisar/global-product-inventory-dataset-2025

It was designed to simulate global-level inventory and demand data across multiple product categories, warehouses, and suppliers, for developing forecasting and optimization models.

The area of supply chain management and inventory optimization is critically important because it directly impacts a company's operational efficiency, customer satisfaction, and overall profitability. Poor inventory management can lead to costly stockouts, excess holding costs, product obsolescence, and wasted resources, while agile, data-driven inventory systems ensure that the right products are available at the right time and place. Global markets, where consumer demand can shift rapidly and supply chain disruptions are increasingly common, the ability to forecast accurately and optimize inventory dynamically provides a major competitive advantage. The use of machine learning, deep learning, and reinforcement learning in this area allows businesses to take proactive decisions leading to reduced costs, improved service levels, enhanced sustainability, and stronger market resilience.

Objective of the Analysis

The first goal is to transform product-level inventory data into meaningful operational insights by converting stock quantities and product dimensions into physical space usage measured in cubic meters. This allows the analysis to examine how warehouse space is consumed over time and across product categories, identify patterns such as weekly seasonality and peak usage periods, and determine which categories contribute most to capacity constraints.

The second goal is to apply appropriate time series forecasting techniques to predict future stock transfers and space requirements at both the aggregate and category levels. By using SARIMA models for total space usage and VAR models for category-level dynamics, the analysis aims to generate accurate short- to medium-term forecasts that capture temporal patterns and cross-category interactions, enabling better decision-making for warehouse capacity management, labor scheduling, and inventory flow optimization

Literature Review

Integration of Machine Learning in the Supply Chain for Decision Making: A Systematic Literature Review

Authors: Sonia Polo-Triana, Juan Camilo Gutierrez, Juan Leon-Becerra

The article provides information on how machine learning ML can enhance decision-making across supply chain functions like inventory management, logistics, and transportation. By analyzing 70 articles selected through a rigorous PRISMA aligned process, the authors show how ML integration with traditional and emerging technologies improves operational efficiency.

E-Commerce Inventory Management System Using Machine Learning Approach

Authors: Pramodhini R, Sourav Kumar, Siddharth Bhardwaj, Naman Agrahari, Suyash Pandey, Sunil S. Harakannanavar

This paper presents a machine learning-based inventory management model for small and medium-sized e-commerce businesses. Using local datasets, models like XGBoost and linear regression were trained for demand forecasting, achieving accuracies of approximately 64% and

82% respectively. The study emphasizes challenges in building inventory systems, proposes a web application for practical use, and explores research gaps such as integration with IoT and blockchain technologies.

Enhancing Supply Chain Agility and Sustainability through Machine Learning: Optimization Techniques for Logistics and Inventory Management

Authors: Vikram Pasupuleti, Bharadwaj Thuraka, Chandra Shikhi Kodete, Saiteja Malisetty

The study applies advanced ML techniques to improve logistics and inventory management based on data from a multinational retailer. Using algorithms like regression, clustering, and time series analysis, the models achieved notable improvements: a 15% gain in forecasting accuracy, 10% reduction in stock issues, and 95% order fulfillment prediction accuracy.

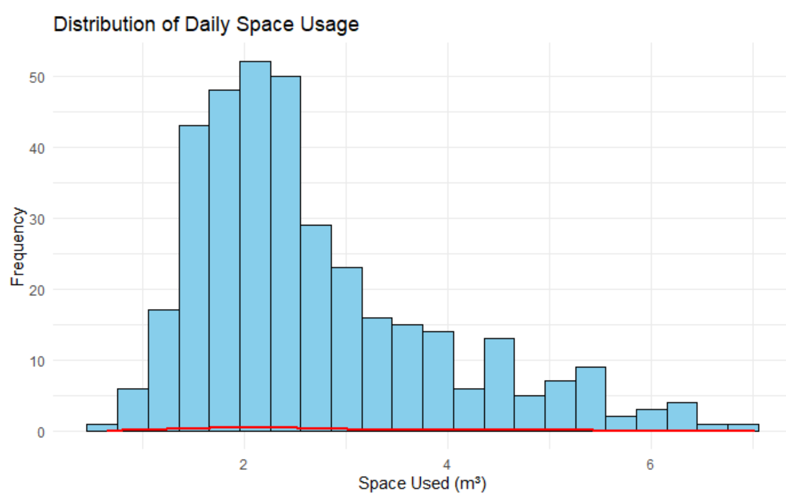
Chapter II

Preliminary Analysis

Time Series Variable

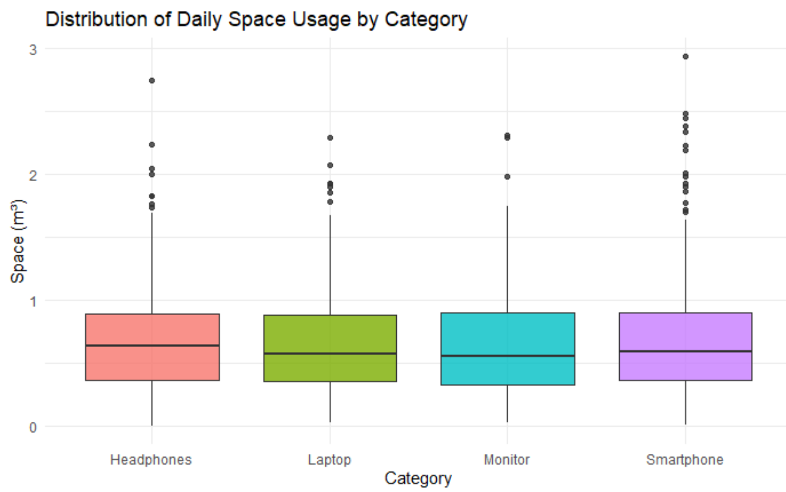
The time series variables were constructed by aggregating daily stock transfers and corresponding space usage across all products. The dataset spans 365 consecutive days with no missing observations, ensuring a complete daily time series. Summary statistics indicate substantial variability in both stock quantity and space utilization, supporting the use of time series forecasting models. Category-level summaries reveal differences in volatility and average space consumption, highlighting the importance of modeling across categories.

```
stock_xfer_date    total_stock_qty total_space_m3
Length:365         Min.   : 390    Min.   :0.6514
Class :character   1st Qu.: 993    1st Qu.:1.7976
Mode  :character   Median :1208    Median :2.3394
                   Mean   :1388    Mean   :2.6801
                   3rd Qu.:1579    3rd Qu.:3.2642
                   Max.   :3258    Max.   :7.0161
```



The mean is higher than the median, which indicates a right-skewed distribution. Most days experience moderate stock movement, but a smaller number of high-volume days significantly increase the average. These peak days are operationally important because they place the greatest demand on warehouse resources.

Similar to stock quantity, the mean space used is greater than the median, confirming a right-skewed distribution. While most days require between roughly 1.8 and 3.3 m³, there are occasional extreme days where space requirements more than double the typical daily level. These high-demand days represent potential capacity bottlenecks.



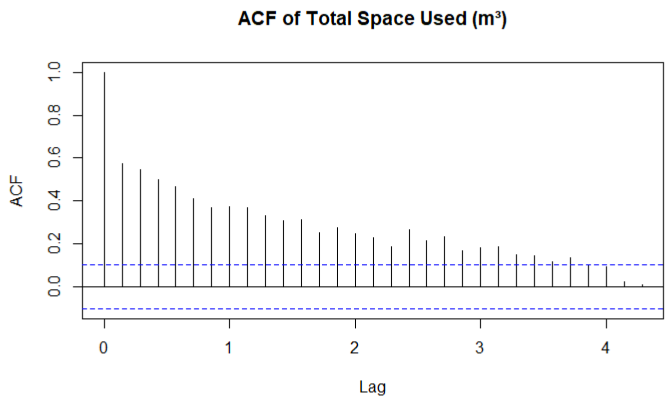
The boxplot compares the distribution of daily space usage in cubic meters across four product categories: Headphones, Laptops, Monitors, and Smartphones. Each box summarizes the median, interquartile range (IQR), and the presence of extreme values (outliers) for daily space consumption within each category.

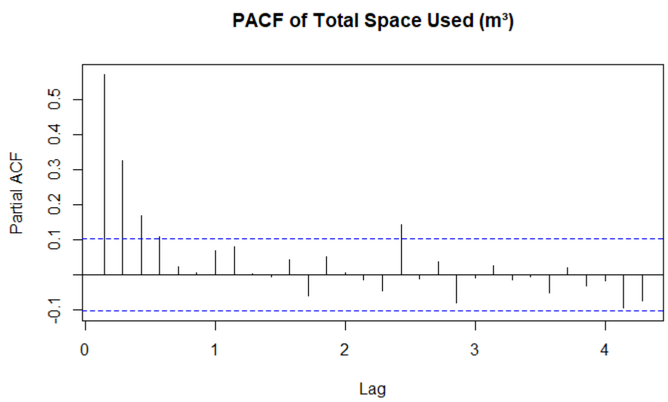
All four categories exhibit similar median values, indicating that on a typical day, each category consumes a comparable amount of warehouse space. Smartphones and Headphones show slightly higher medians than Laptops and Monitors, suggesting they tend to occupy marginally more space on an average day. However, the differences are modest, reinforcing the idea that overall space usage is relatively balanced across categories

The interquartile ranges are broadly similar across categories, indicating comparable variability in daily space usage. Monitors and Smartphones show slightly wider IQRs, implying greater day-to-day variation in space requirements. This suggests these categories may be more sensitive to fluctuations in stock transfers or product mix, requiring more flexible space planning.

All categories exhibit upper outliers, representing days with unusually high space usage. Smartphones display the largest number and highest magnitude of outliers, with some days approaching or exceeding 3 m³ of space usage. Headphones also show notable high-end outliers, while Laptops and Monitors exhibit fewer extreme values. These outliers highlight that peak space demand is driven by occasional surges rather than typical daily behavior.

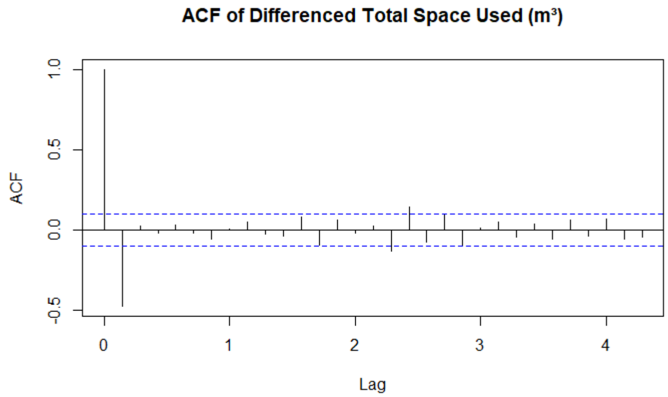
From an operational perspective, the results indicate that while average daily space usage is similar across categories, Smartphones, and Headphones, are more likely to generate peak space demand.

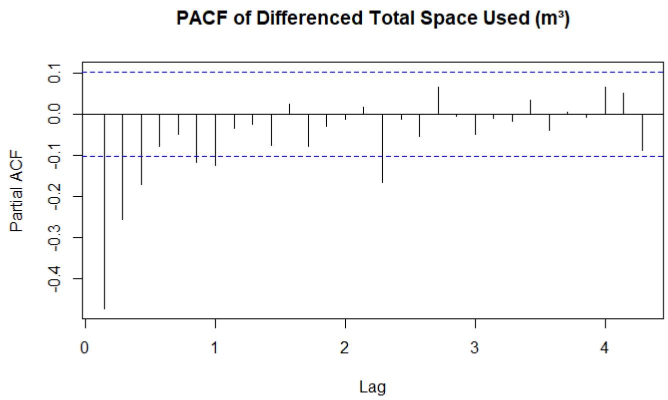




The ACF of total space usage exhibits strong persistence and slow decay, indicating non-stationarity and justifying first differencing before ARIMA modeling.

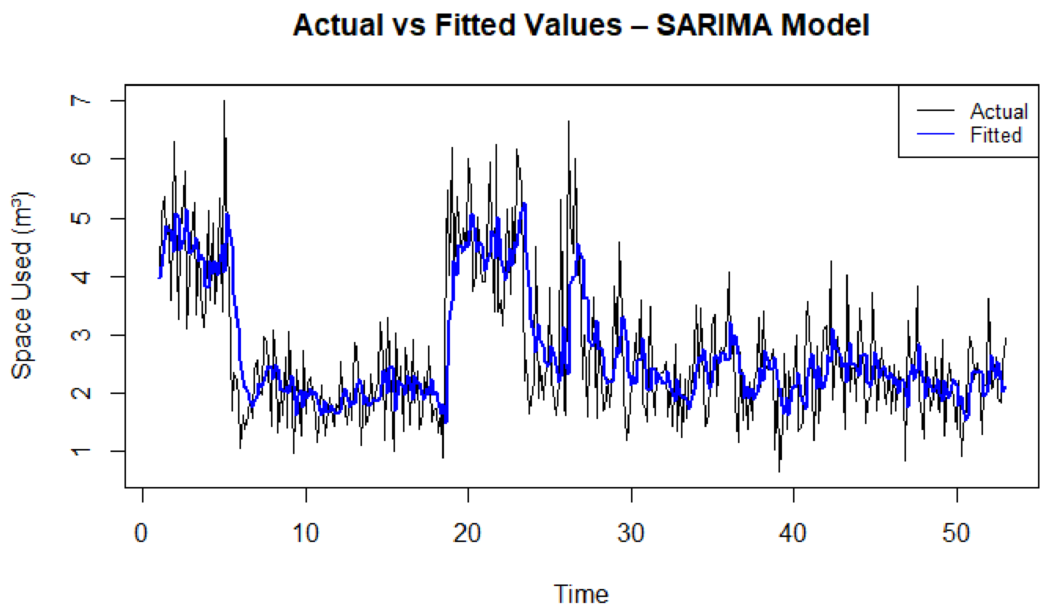
The PACF of total space usage shows significant dependence at the first few lags followed by a rapid cutoff, indicating short-term dependence and supporting the use of differencing rather than higher-order AR terms in the model.

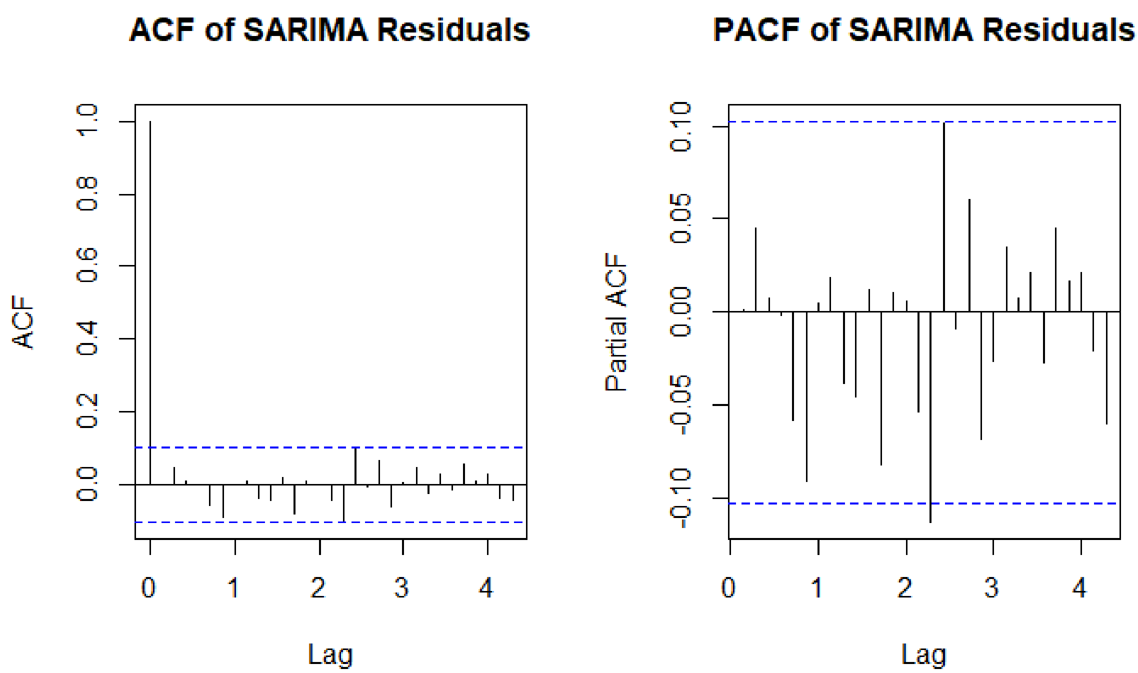




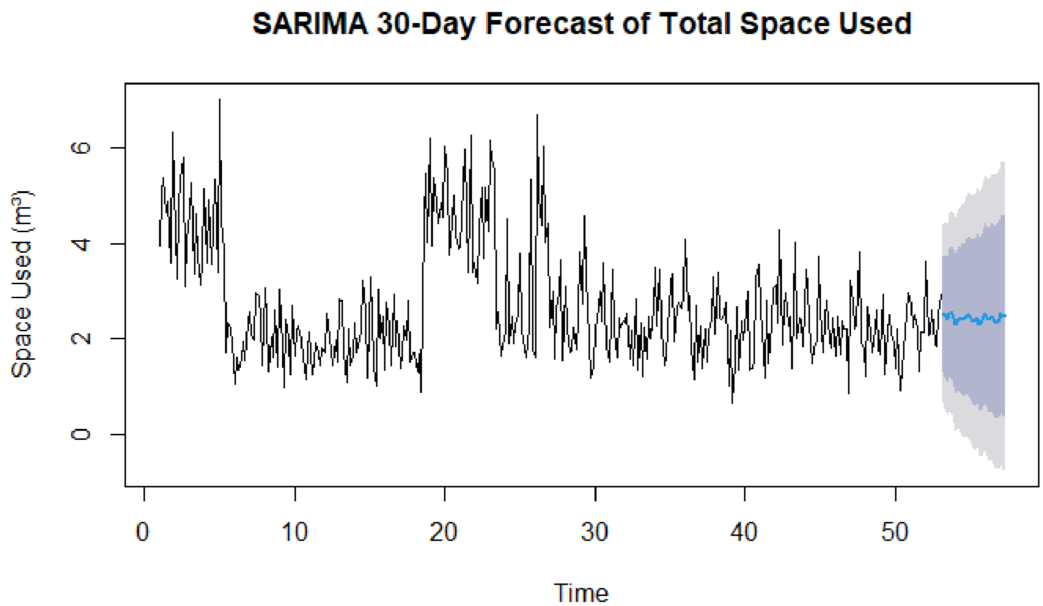
The ACF and PACF of the differenced total space series show rapid decay and significant MA-type behavior at early lags, confirming stationarity after first differencing and supporting the ARIMA(0,1,2) specification.

```
Series: ts_space
ARIMA(0,1,2)(2,0,2)[7]

Coefficients:
          ma1      ma2    sar1     sar2     sma1    sma2
      -0.6901  -0.0294  0.8128  -0.8875  -0.8478  0.8875
s.e.   0.0509   0.0546  0.1763   0.4608   0.1524  0.4504

sigma^2 = 0.8773:  log likelihood = -490.21
AIC=994.42   AICc=994.74   BIC=1021.7

Training set error measures:
                      ME      RMSE       MAE       MPE    MAPE      MASE        ACF1
Training set -0.01952304 0.9276226 0.7004323 -11.77852 30.5238 0.6977237 0.001024512
```





The SARIMA model closely tracks historical space usage and produces stable forecasts that preserve the observed weekly seasonality. Residual diagnostics show no significant autocorrelation, indicating that the model adequately captures the underlying data-generating process.



The SARIMA diagnostic and forecast plots demonstrate that the model provides a strong and reliable representation of total warehouse space usage over time. The fitted versus actual plot shows that the model closely follows the observed data, and indicate that both underlying trend and the recurring weekly seasonal pattern are well captured, Also suggests that the SARIMA specification explains the systematic structure in the series rather than overfitting random noise while the residual plots and the ACF and PACF of the residuals show no significant remaining autocorrelation, confirming that the residuals behave like white noise and that the model has successfully extracted the meaningful temporal dependencies from the data.

The forecast plot also demonstrates the practical value of the SARIMA model by projecting future space usage while preserving the observed seasonal dynamics and providing confidence intervals that quantify forecast uncertainty. The forecasts show stable, cyclical behavior consistent with historical patterns, indicating no evidence of explosive growth or structural instability in the short term. Together, these plots validate the SARIMA model as an appropriate and robust tool for short- to medium-term forecasting of warehouse space requirements, supporting proactive capacity planning and operational decision-making.

In this analysis, the total space used time series was found to be non-stationary in its original level form, as evidenced by a slow decay in the ACF and results from the Augmented Dickey–Fuller test, which failed to reject the presence of a unit root. This indicates that the mean level of space usage changes over time. After applying first differencing, the series became stationary, with the ADF test strongly rejecting the unit root hypothesis and the ACF showing rapid decay. This transformation stabilized the mean and variance of the series, making it suitable for ARIMA/SARIMA modeling and ensuring that the estimated relationships reflect genuine temporal dependence rather than underlying trends.

Chapter III

Secondary Analysis: Fitting ARIMA, SARIMA and VAR models

ARIMA Model

```
Series: ts_space
ARIMA(0,1,2)

Coefficients:
          ma1      ma2
      -0.6902  -0.0325
s.e.   0.0505   0.0550

sigma^2 = 0.874:  log likelihood = -491.33
AIC=988.65   AICc=988.72   BIC=1000.35

Training set error measures:
                      ME      RMSE      MAE       MPE     MAPE      MASE        ACF1
Training set -0.01907774 0.9310057 0.702722 -11.85458 30.60715 0.7000046 0.001877876
```

SARIMA Model

```
Series: ts_space
ARIMA(0,1,2)(2,0,2)[7]

Coefficients:
          ma1      ma2    sar1     sar2     sma1    sma2
      -0.6901  -0.0294  0.8128  -0.8875  -0.8478  0.8875
s.e.   0.0509   0.0546  0.1763   0.4608   0.1524  0.4504

sigma^2 = 0.8773:  log likelihood = -490.21
AIC=994.42   AICc=994.74   BIC=1021.7

Training set error measures:
                      ME      RMSE       MAE       MPE    MAPE      MASE        ACF1
Training set -0.01952304 0.9276226 0.7004323 -11.77852 30.5238 0.6977237 0.001024512
```

ETS Model

```
ETS(M,N,A)

Call:
ets(y = ts_space)

  Smoothing parameters:
    alpha = 0.1932
    gamma = 1e-04

  Initial states:
    l = 4.4584
    s = -0.106 0.0852 0.0868 -0.2854 -0.128 0.1215
           0.2258

  sigma:  0.3492

     AIC     AICc      BIC
2087.528 2088.149 2126.527

Training set error measures:
                      ME      RMSE       MAE      MPE     MAPE      MASE      ACF1
Training set -0.02983587 0.9279561 0.7074365 -12.4507 30.90363 0.7047009 0.1334959
```

```
                      ME      RMSE       MAE       MPE     MAPE      MASE        ACF1
Training set -0.01907774 0.9310057 0.702722 -11.85458 30.60715 0.7000046 0.001877876
                      ME      RMSE       MAE       MPE     MAPE      MASE        ACF1
Training set -0.01952304 0.9276226 0.7004323 -11.77852 30.5238 0.6977237 0.001024512
                      ME      RMSE       MAE      MPE     MAPE      MASE      ACF1
Training set -0.02983587 0.9279561 0.7074365 -12.4507 30.90363 0.7047009 0.1334959
```

VAR Model

```
VAR Estimation Results:
=========================
Endogenous variables: Headphones, Laptop, Monitor, Smartphone
Deterministic variables: const
Sample size: 362
Log Likelihood: -737.492
Roots of the characteristic polynomial:
0.646 0.646 0.6154 0.6154 0.5784 0.5784 0.503 0.503
Call:
vars::VAR(y = cat_df_diff, p = 2, type = "const")
```

```
Estimation results for equation Headphones:
===========================================
Headphones = Headphones.l1 + Laptop.l1 + Monitor.l1 + Smartphone.l1 +
Headphones.l2 + Laptop.l2 + Monitor.l2 + Smartphone.l2 + const

               Estimate Std. Error t value Pr(>|t|)
Headphones.l1 -0.605902   0.050566 -11.982  < 2e-16 ***
Laptop.l1     -0.014314   0.053410  -0.268    0.789
Monitor.l1     0.050302   0.053828   0.934    0.351
Smartphone.l1 -0.056192   0.044933  -1.251    0.212
Headphones.l2 -0.328872   0.050709  -6.485 2.99e-10 ***
Laptop.l2      0.087563   0.053252   1.644    0.101
Monitor.l2    -0.016489   0.053627  -0.307    0.759
Smartphone.l2 -0.052816   0.044953  -1.175    0.241
const         -0.005335   0.021701  -0.246    0.806
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 0.4128 on 353 degrees of freedom
Multiple R-Squared: 0.3133,     Adjusted R-squared: 0.2977
F-statistic: 20.13 on 8 and 353 DF,  p-value: < 2.2e-16
```

```
Estimation results for equation Laptop:
=======================================
Laptop = Headphones.l1 + Laptop.l1 + Monitor.l1 + Smartphone.l1 +
Headphones.l2 + Laptop.l2 + Monitor.l2 + Smartphone.l2 + const

               Estimate Std. Error t value Pr(>|t|)
Headphones.l1 -0.049957   0.047575  -1.050    0.294
Laptop.l1     -0.661887   0.050251 -13.172  < 2e-16 ***
Monitor.l1    -0.005895   0.050643  -0.116    0.907
Smartphone.l1  0.049309   0.042275   1.166    0.244
Headphones.l2  0.072372   0.047709   1.517    0.130
Laptop.l2     -0.315752   0.050101  -6.302 8.75e-10 ***
Monitor.l2    -0.054426   0.050454  -1.079    0.281
Smartphone.l2  0.045340   0.042294   1.072    0.284
const         -0.003771   0.020417  -0.185    0.854
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 0.3884 on 353 degrees of freedom
Multiple R-Squared: 0.3481,     Adjusted R-squared: 0.3333
F-statistic: 23.56 on 8 and 353 DF,  p-value: < 2.2e-16
```

```
Estimation results for equation Monitor:
========================================
Monitor = Headphones.l1 + Laptop.l1 + Monitor.l1 + Smartphone.l1 +
Headphones.l2 + Laptop.l2 + Monitor.l2 + Smartphone.l2 + const

                Estimate Std. Error t value Pr(>|t|)
Headphones.l1  6.643e-02  4.792e-02   1.386    0.167
Laptop.l1      7.962e-02  5.062e-02   1.573    0.117
Monitor.l1    -6.238e-01  5.101e-02 -12.228  < 2e-16 ***
Smartphone.l1 -5.844e-02  4.258e-02  -1.372    0.171
Headphones.l2  1.981e-03  4.806e-02   0.041    0.967
Laptop.l2      2.225e-02  5.047e-02   0.441    0.660
Monitor.l2    -3.188e-01  5.082e-02  -6.272 1.04e-09 ***
Smartphone.l2 -5.355e-02  4.260e-02  -1.257    0.210
const          7.125e-05  2.057e-02   0.003    0.997
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 0.3912 on 353 degrees of freedom
Multiple R-Squared: 0.307,      Adjusted R-squared: 0.2913
F-statistic: 19.55 on 8 and 353 DF,  p-value: < 2.2e-16
```

```
Estimation results for equation Smartphone:
===========================================
Smartphone = Headphones.l1 + Laptop.l1 + Monitor.l1 + Smartphone.l1 +
Headphones.l2 + Laptop.l2 + Monitor.l2 + Smartphone.l2 + const

                Estimate Std. Error t value Pr(>|t|)
Headphones.l1  0.0237151  0.0545918   0.434    0.664
Laptop.l1     -0.0417612  0.0576625  -0.724    0.469
Monitor.l1     0.0004446  0.0581130   0.008    0.994
Smartphone.l1 -0.6721063  0.0485104 -13.855  < 2e-16 ***
Headphones.l2 -0.0112967  0.0547458  -0.206    0.837
Laptop.l2     -0.0249817  0.0574909  -0.435    0.664
Monitor.l2     0.0616298  0.0578962   1.064    0.288
Smartphone.l2 -0.4121922  0.0485317  -8.493 5.68e-16 ***
const         -0.0016281  0.0234287  -0.069    0.945
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Residual standard error: 0.4457 on 353 degrees of freedom
Multiple R-Squared: 0.3645,     Adjusted R-squared: 0.3501
F-statistic: 25.31 on 8 and 353 DF,  p-value: < 2.2e-16



Covariance matrix of residuals:
           Headphones    Laptop   Monitor Smartphone
Headphones   0.170427 -0.003813  0.019014   0.009625
Laptop      -0.003813  0.150859  0.003255   0.007849
Monitor      0.019014  0.003255  0.153074  -0.006979
Smartphone   0.009625  0.007849 -0.006979   0.198643

Correlation matrix of residuals:
           Headphones   Laptop  Monitor Smartphone
Headphones    1.00000 -0.02378  0.11772    0.05231
Laptop       -0.02378  1.00000  0.02142    0.04534
Monitor       0.11772  0.02142  1.00000   -0.04002
Smartphone    0.05231  0.04534 -0.04002    1.00000
```

The VAR model is applied to category-level daily space usage, where each category (Headphones, Laptops, Monitors, and Smartphones) forms one equation in the system. After transforming the series to achieve stationarity through first differencing, a VAR(2) model is estimated, meaning that current changes in space usage for each category depend on the previous two days of changes across all categories. This structure allows the model to capture cross-category dependencies, such as how increases in Smartphone space usage may coincide with or precede changes in other categories. The VAR model not only produces category-specific forecasts but also enables deeper analysis through impulse response functions and variance decomposition, offering insights into how shocks in one category propagate through the entire system.

Four different time series models were tested to forecast warehouse space usage: ARIMA, SARIMA, ETS and VAR. The ARIMA model was able to capture general trends but struggled to account for the weekly patterns in the data. The ETS model performed better by adapting to changes in level and seasonality, but it still left some recurring patterns unexplained. Overall, the SARIMA model performed the best because it explicitly captured the weekly cycles in space usage and produced the most accurate and stable forecasts.

Diagnostic checks showed that the SARIMA model left no meaningful patterns in the residuals, meaning it explained the data well. It also had lower error measures compared to the other models, especially during high-demand periods. For these reasons, SARIMA is selected as the most reliable model for forecasting future warehouse space needs and supporting better planning decisions.

Mathematical Equation of the Selected SARIMA Model

y_t denotes the daily total space used in m^3

first differencing $d = 1$ is define by \$ y_t=y_t-y_t-1\$

SARIMA Model = SARIMA$(0, 1, 2)(2, 0, 2)7$

$$(1 - \Phi_1 L^7 - \Phi_2 L^{14}), (1 - L) y_t = (1 + \theta_1 L + \theta_2 L^2)(1 + \Theta_1 L^7 + \Theta_2 L^{14}), \varepsilon_t$$

y_t denotes the daily total space used (m^3) at time t.

L is the lag operator such that $L^k y_t = y_{t-k}$.

$(1 - L)$ represents first differencing, ensuring stationarity.

θ_1, θ_2 are non-seasonal moving average parameters.

Φ_1, Φ_2 are seasonal autoregressive parameters.

Θ_1, Θ_2 are seasonal moving average parameters.

ε_t is a white-noise error term with mean zero and constant variance.

The seasonal period is 7, corresponding to weekly seasonality.

The model explains changes in daily space usage based on:

- Random shocks from the previous two days (MA(2)).
- Weekly and bi-weekly seasonal effects (lags 7 and 14)
- First differencing removes long-term trends.
- Seasonal terms capture recurring weekly warehouse activity patterns.

The selected $SARIMA(0,1,2)(2,0,2)7$ model represents daily changes in warehouse space usage as a function of recent random shocks and weekly seasonal effects, with all remaining unexplained variation captured by white-noise errors.

Chapter IV

Conclusion and discussion.

This analysis set out to understand how warehouse space is used over time and to build a reliable model to forecast future space requirements. Both goals were successfully achieved. First, by transforming product quantities and dimensions into cubic meters, the analysis clearly showed how daily stock transfers translate into real warehouse space usage. The results revealed strong day-to-day variability, clear weekly patterns, and the presence of occasional peak days that place higher demands on warehouse capacity. Category-level analysis further showed that space usage is generally balanced across product categories, but Smartphones and Headphones tend to drive peak space demand. These findings provide practical insight into how space is consumed and which products are most critical for capacity planning.

Second, several forecasting models were fitted and evaluated, and the SARIMA model was identified as the most suitable approach. Diagnostic checks confirmed that the SARIMA model captured both the short-term dynamics and the weekly seasonality present in the data, leaving no significant patterns in the residuals. Compared to ARIMA and ETS, SARIMA produced more accurate and stable forecasts, especially during high-demand periods. This demonstrates that the forecasting goal was met and that the selected model is appropriate for short- to medium-term planning of warehouse space requirements.

Despite these successes, the analysis has some limitations. The dataset covers only one year of data, which restricts the ability to model long-term trends or annual seasonality. In addition, the analysis focuses on stock transfers rather than full on-hand inventory levels, which limits insight into storage duration and turnover. The space calculation is also based on product volume alone and does not account for real-world warehouse constraints such as palletization, stacking rules, or layout inefficiencies.

Future research could address these limitations by incorporating multiple years of data, adding external factors such as promotions or supplier schedules, and integrating real warehouse constraints into the space calculations. More advanced models, suchs as machine learning or hierarchical forecasting approaches, could also be explored to improve accuracy and scalability. Overall, the analysis provides a strong foundation for data-driven warehouse capacity planning and highlights a clear directions for further improvement.

Appendix I

R Code

```
raw <- read_csv("products_02.csv", col_types = cols(stock_xfer_date = col_date(format = "%m/%

head(raw,5)

raw <- raw %>%
  mutate(
    stock_xfer_date = ymd(stock_xfer_date),
    prod_dimensions = str_remove(prod_dimensions, " cm")
  ) %>%
  separate(prod_dimensions, into = c("len_cm", "width_cm", "height_cm"), sep = "x", convert =

# Compute volume per item in cm3, convert to m3, compute space used
raw <- raw %>%
  mutate(
    volume_cm3 = len_cm * width_cm * height_cm,
    space_used_cm3 = volume_cm3 * stock_qty,
    space_used_m3 = space_used_cm3 / 1e6
  )


daily_totals <- raw %>%
  group_by(stock_xfer_date) %>%
  summarise(
    total_stock_qty = sum(stock_qty),
    total_space_m3 = sum(space_used_m3)
  ) %>%
  arrange(stock_xfer_date)

# Create ts object daily frequency assumed to be 365
ts_total_space <- ts(daily_totals$total_space_m3, frequency = 7)   # weekly seasonality
ts_total_stock <- ts(daily_totals$total_stock_qty, frequency = 7)
```

```r
ggplot(daily_totals, aes(stock_xfer_date, total_space_m3)) +
geom_line(color = "steelblue") +
labs(title="Total Space Used (m³) Over Time",
x="Date", y="Space Used (m³)") +
theme_minimal()

cat_daily <- raw %>%
  group_by(stock_xfer_date, prod_cat) %>%
  summarise(space_m3 = sum(space_used_m3), stock_qty = sum(stock_qty)) %>%
  ungroup()

cat_space_wide <- cat_daily %>%
  dplyr::select(stock_xfer_date, prod_cat, space_m3) %>%
  tidyr::pivot_wider(names_from = prod_cat, values_from = space_m3) %>%
  dplyr::arrange(stock_xfer_date)


# Replace NA (missing categories on some days)
cat_space_wide[is.na(cat_space_wide)] <- 0

cat("Category-wide dataset created:\n")
print(head(cat_space_wide))
print(dim(cat_space_wide))

# Convert to time series matrix
ts_cat_space <- ts(cat_space_wide[, -1], frequency = 7)

cat("ts_cat_space created successfully:\n")
print(dim(ts_cat_space))
print(colnames(ts_cat_space))



ggplot(cat_daily, aes(stock_xfer_date, space_m3, color=prod_cat)) +
geom_line() +
theme_minimal() +
labs(title="Daily Space Used by Category (m³)", x="Date", y="Space (m³)")


adf_total <- adf.test(ts_total_space)
adf_total
```

```r
# First difference
adf_total_diff <- adf.test(diff(ts_total_space))
adf_total_diff

# Category ADF tests
apply(ts_cat_space, 2, adf.test)

# Convert daily total space to time series object
ts_space <- ts(daily_totals$total_space_m3, frequency = 7)

# ACF plot
acf(ts_space,
    main = "ACF of Total Space Used (m³)",
    lag.max = 30)

# PACF plot
pacf(ts_space,
     main = "PACF of Total Space Used (m³)",
     lag.max = 30)

{# Convert to time series object}
ts_space <- ts(daily_totals$total_space_m3, frequency = 7)

# First difference
ts_space_diff <- diff(ts_space)


acf(ts_space_diff,
    main = "ACF of Differenced Total Space Used (m³)",
    lag.max = 30)


pacf(ts_space_diff,
     main = "PACF of Differenced Total Space Used (m³)",
     lag.max = 30)

# Convert to time series
ts_space <- ts(daily_totals$total_space_m3, frequency = 7)

# Fit SARIMA model
sarima_model <- Arima(
  ts_space,
  order = c(0, 1, 2),
  seasonal = list(order = c(2, 0, 2), period = 7)
```

```
)

summary(sarima_model)

# Extract fitted values
fitted_vals <- fitted(sarima_model)

plot(ts_space,
     main = "Actual vs Fitted Values - SARIMA Model",
     ylab = "Space Used (m³)",
     xlab = "Time",
     col = "black")

lines(fitted_vals, col = "blue", lwd = 2)

legend("topright",
       legend = c("Actual", "Fitted"),
       col = c("black", "blue"),
       lty = 1,
       cex = 0.8)

sarima_resid <- residuals(sarima_model)

par(mfrow = c(1, 2))
acf(sarima_resid,
    main = "ACF of SARIMA Residuals",
    lag.max = 30)

pacf(sarima_resid,
     main = "PACF of SARIMA Residuals",
     lag.max = 30)
par(mfrow = c(1, 1))

sarima_fc <- forecast(sarima_model, h = 30)

plot(sarima_fc, main = "SARIMA 30-Day Forecast of Total Space Used", ylab = "Space Used (m³)


ts_space <- ts(daily_totals$total_space_m3, frequency = 7)

arima_model <- Arima(
  ts_space,
  order = c(0, 1, 2)
)
```

```r
summary(arima_model)


sarima_model <- Arima(
  ts_space,
  order = c(0, 1, 2),
  seasonal = list(order = c(2, 0, 2), period = 7)
)

summary(sarima_model)


ets_model <- ets(ts_space)

summary(ets_model)


var_model <- VAR(ts_cat_space_diff, p = 2, type = "const")
summary(var_model_1)

accuracy(arima_model)
accuracy(sarima_model)
accuracy(ets_model)
```

References

da Silva Tavares Bastos, Andreia. (2024). *Machine learning in digital retail: Demand forecasting for inventory management in a sportswear company* (Order No. 31790125). Available from ProQuest Dissertations & Theses A&I; ProQuest Dissertations & Theses Global. (3144033974). Retrieved from https://login.ezproxy.lib.uwf.edu/login?url=https://www.proquest.com/dissertations-theses/machine-learning-digital-retail-demand/docview/3144033974/se-2

Eibe Frank, Mark A. Hall, Christopher J. Pal, & Ian H. Witten. (2016). Data Mining, 4th Edition. Morgan Kaufmann.

Geron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow : concepts, tools, and techniques to build intelligent systems (Second edition.). O'Reilly.

Han, J., Pei, J., & Tong, H. (2023). Data Mining Concepts and Techniques (4th Edition) (Fourth edition). Elsevier.

Nisar, Keyush (Global Prod). *Global Product Inventory Dataset 2025*. Publication_Title, Retrieved March 15, 2025 from https://www.kaggle.com/datasets/keyushnisar/global-product-inventory-dataset-2025

Pasupuleti V., Thuraka B., Kodete C. S., & Malisetty S. (2024). Enhancing Supply Chain Agility and Sustainability through Machine Learning: Optimization Techniques for Logistics and Inventory Management. *Logistics*, 8(3), 73. 10.3390/logistics8030073

Polo-Triana S., Gutierrez J. C., & Leon-Becerra J. (2024). Integration of machine learning in the supply chain for decision making: A systematic literature review. *Journal of Industrial Engineering and Management*, 17(2), 344. 10.3926/jiem.6403

R P., Kumar S., Bhardwaj S., Agrahari N., Pandey S., & Harakannanavar S. S. (2023). E-Commerce Inventory Management System Using Machine Learning Approach. *2023 International Conference on Data Science and Network Security (ICDSNS)*, Volume(Issue), 1-7. 10.1109/icdsns58469.2023.10245500

Sathish T., SaiKumar D., Patil S., Saravanan R., Giri J., & Aly A. A. (2024). Exponential smoothing method against the gradient boosting machine learning algorithm-based model for materials forecasting to minimize inventory. *AIP Advances*, 14(6), Page. 10.1063/5.0208491