JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 13 No. 3S1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v13i3S1.8173

# PERBANDINGAN MODEL SARIMA, *EXPONENTIAL SMOOTHING*, DAN XGBOOST UNTUK PREDIKSI PENJUALAN SUPER STORE

**Ketut Rega Arunika[1] , Luh Joni Erawati Dewi[2]**

[1,2]Program Studi Ilmu Komputer, Universitas Pendidikan Ganesha; 67 Jalan Ahmad Yani 81116

**Keywords:**
Prediksi Penjualan, SARIMA, *Exponential smoothing*, XGBoost, Deret Waktu.

**Corespondent Email:**
regawp55@gmail.com

***Abstrak****:* Prediksi penjualan merupakan aspek penting dalam mendukung pengambilan keputusan strategis pada industri ritel, terutama dalam perencanaan stok dan manajemen rantai pasok. Seiring meningkatnya kompleksitas pola pembelian konsumen, diperlukan model prediksi yang mampu menangkap pola tren dan musiman secara akurat. Penelitian ini bertujuan untuk membandingkan efektivitas tiga metode prediksi, yaitu SARIMA, *Exponential smoothing*, dan **X**GBoost, dalam memprediksi jumlah produk yang terjual pada dataset *Super Store* periode 2014–2017. Data harian dikonversi menjadi data bulanan, kemudian melalui proses *preprocessing* seperti pembersihan data, pengecekan duplikasi, pemilihan atribut dan pembagian *train–test*. Model SARIMA dibangun dengan optimasi parameter melalui *grid search*, *Exponential smoothing* menggunakan konfigurasi tren dan musiman aditif, sedangkan XGBoost menerapkan *feature engineering* berbasis *lag* dan musiman. Evaluasi dilakukan dengan metrik MAE, MSE, RMSE, dan $R^2$. Hasil menunjukkan bahwa SARIMA memberikan performa terbaik dengan $R^2 = 0{,}932$, diikuti oleh *Exponential smoothing* dan XGBoost. Temuan ini menunjukkan bahwa metode *time series* tradisional lebih sesuai untuk data berpola musiman stabil dibandingkan pendekatan *machine learning*.



***Abstract****: Sales forecasting is a crucial aspect of supporting strategic decision-making in the retail industry, particularly for inventory planning and supply chain management. As consumer purchasing patterns become increasingly complex, predictive models capable of accurately capturing trend and seasonal patterns are required. This study aims to compare the effectiveness of three forecasting methods: SARIMA, Exponential smoothing, and XGBoost, in predicting the number of products sold in the Super Store dataset for the period 2014–2017. Daily data were aggregated into monthly data and underwent preprocessing, including data cleaning, duplication checks, attribute selection and splitting into training and testing sets. The SARIMA model was developed with parameter optimization using grid search, Exponential smoothing employed additive trend and seasonal configurations, while XGBoost applied feature engineering based on lag and seasonality. Evaluation was conducted using MAE, MSE, RMSE, and $R^2$ metrics. The results indicate that SARIMA achieved the best performance with $R^2 = 0.932$, followed by Exponential smoothing and XGBoost. These findings suggest that traditional time-series methods are more suitable for data with stable seasonal patterns compared to machine learning approaches.*

Vol. 13 No. 3S1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v13i3S1.8173

## 1. PENDAHULUAN

Prediksi penjualan merupakan aspek penting dalam pengambilan keputusan strategis pada industri ritel. Prediksi yang sangat penting karena dapat membantu menjaga kelancaran distribusi serta mendukung perencanaan kebutuhan sumber daya secara menyeluruh [1]. Selain itu, prediksi yang tepat juga mampu membantu perusahaan dalam mengelola stok, mengatur rantai pasok, menentukan strategi pemasaran, serta meminimalisasi kerugian akibat ketidakseimbangan permintaan dan persediaan. Seiring dengan meningkatnya kompleksitas pola pembelian konsumen, diperlukan model prediksi yang dapat menangkap karakteristik data penjualan yang bersifat musiman, memiliki tren, dan sering kali dipengaruhi faktor *non-linear*. Dengan perkembangan model *time series* dan *artificial intelligence* terutama *machine learning. Machine learning* dapat dimanfaatkan tidak hanya untuk klasifikasi, tetapi juga dapat digunakan untuk prediksi [2]. *Machine learning* dapat mempermudah prediksi yang akurat, deteksi anomali, dan pengenalan pola yang kompleks dalam data yang bergantung pada waktu, melampaui kemampuan metode statistik tradisional [3].

Beberapa penelitian sebelumnya, metode *time series* tradisional seperti *Seasonal Autoregressive Integrated Moving Average* (SARIMA) yang ditulis oleh Falatouri *et al*., (2022), terbukti efektif dalam mengatasi pola musiman dan tren pada data penjualan ritel [4]. Metode lain seperti *Exponential smoothing* yang ditulis oleh İnce (2024), khususnya *Holt-Winters*, banyak digunakan untuk meramalkan data penjualan yang bersifat stabil dengan tren linier sederhana. Namun, perkembangan *machine learning* menghadirkan alternatif baru [5]. Model berbasis *gradient boosting* seperti *XGBoost* yang ditulis oleh Mustapha dan Sithole (2025) mampu menangkap hubungan *non-linear* dan kompleks antar variabel sehingga memberikan akurasi lebih tinggi pada kasus tertentu [6]. Dari beberapa penelitian sebelumnya menunjukkan bahwa setiap pendekatan memiliki keunggulan tersendiri, dimana metode *time series* tradisional unggul dalam memodelkan tren dan musiman, sedangkan *machine learning* lebih efektif menangani interaksi *non-linear* [7].

Kebaruan penelitian ini terletak pada pendekatan komparatif atau perbandingan yang mengintegrasikan metode *time series* tradisional dan *machine learning* untuk mengevaluasi efektivitasnya pada kasus prediksi penjualan Super Store. Pendekatan integratif dan komparatif ini penting karena pada studi *benchmark* yang sistematis masih terbatas, terutama yang menilai keseimbangan antara kemudahan model tradisional dan akurasi tinggi model *machine learning* dalam konteks bisnis ritel yang dinamis [8]. Kontribusi penelitian ini tidak hanya bersifat akademis dengan memperluas literatur mengenai perbandingan model prediksi, tetapi juga praktis dengan memberikan wawasan bagi pelaku industri ritel dalam memilih metode prediksi yang tepat sesuai karakteristik data penjualan mereka.

Tujuan dari penelitian ini adalah melakukan analisis dan perbandingan kinerja model SARIMA, *Exponential smoothing*, dan XGBoost dalam meramalkan penjualan Super Store yang memiliki pola musiman, tren, serta variasi *non-linear*. Penelitian ini diharapkan mampu memberikan gambaran menyeluruh mengenai keunggulan dan keterbatasan masing-masing metode dalam memodelkan data penjualan, sehingga dapat membantu pelaku industri dalam memilih pendekatan prediksi yang paling sesuai.

## 2. TINJAUAN PUSTAKA

### 2.1 Machine Learning

*Machine Learning* merupakan bagian dari *artificial intelligence* yang membantu komputer atau mesin pengajaran belajar dari semua data sebelumnya dan membuat keputusan yang cerdas [9]. Machine learning terbagi menjadi tiga kategori utama yaitu *supervised learning*, *unsupervised learning*, *semi-supervised learning*, dan *reinforcement learning*, masing-masing cocok berdasarkan jenis data dan tugas yang dihadapi [10]. Penerapan machine learning sudah sangat luas di beberapa bidang misalnya kesehatan, pemeliharaan prediktif, dan analisis data survei, di mana kemampuan *machine learnin* mengolah data dalam jumlah banyak secara otomatis dan memberikan wawasan yang sulit dicapai melalui metode manual [11].

### 2.2 XGBoost

XGBoost (*Extreme Gradient Boosting*) merupakan salah satu algoritma *machine learning* berbasis *ensemble* yang mengadopsi metode *gradient boosting framework*. Algoritma *gradient boosting* yang membangun model prediksi secara bertahap melalui kombinasi pohon keputusan, dimana setiap pohon baru dibuat untuk memperbaiki kesalahan *residual* dari pohon-sebelumnya, dan keseluruhan tujuan (*objective*) dipilih untuk meminimalkan *loss function*. Kemampuan ini ditambah dengan regularisasi dan optimasi *loss* khusus seperti GHM dalam beberapa penelitian [12]. Dibandingkan dengan implementasi *gradient boosting* tradisional, XGBoost dirancang lebih cepat, efisien, dan akurat karena memiliki berbagai optimasi, seperti dukungan regularisasi L1 dan L2 untuk mengurangi risiko *overfitting*, kemampuan paralelisasi dalam proses *train*ing, serta penanganan nilai hilang secara otomatis.

Algoritma XGBoost terbukti efektif dan memberikan akurasi tinggi dalam melakukan prediksi baik pada permasalahan regresi maupun klasifikasi [13]. Keunggulan utamanya terletak pada kemampuan dalam menggabungkan banyak pohon keputusan untuk memperbaiki kesalahan model sebelumnya, sehingga menghasilkan prediksi yang lebih stabil.

Meskipun demikian, masih terdapat potensi untuk meningkatkan akurasi model XGBoost dengan memanfaatkan berbagai teknik *tuning parameter* [13]. Beberapa parameter penting yang dapat disesuaikan antara lain jumlah pohon (*n_estimators*), kedalaman pohon (*max_depth*), laju pembelajaran (*learning rate*), serta tingkat regularisasi.

### 2.3 SARIMA

Model *Seasonal Autoregressive Integrated Moving Average* (SARIMA) merupakan pengembangan dari ARIMA yang dirancang untuk menganalisis data *time series* dengan mempertimbangkan komponen musiman [14]. SARIMA merupakan model statistik yang kuat untuk peramalan data deret waktu yang menunjukkan pola musiman yang jelas, seperti data bulanan, kuartalan, atau tahunan. Model ini tidak hanya mengatasi tren dan non-stasioneritas (seperti ARIMA), tetapi juga secara eksplisit menyertakan elemen untuk menangkap ketergantungan musiman (periodik) dalam data.

Berbeda dengan ARIMA yang hanya mengakomodasi komponen non-musiman, SARIMA menambahkan parameter musiman (P, D, Q, s) untuk menangkap pola yang berulang pada periode tertentu, misalnya bulanan atau tahunan. Model ini mengombinasikan proses *autoregressive* (AR), differencing (I), dan moving *average* (MA) dengan komponen musiman yang setara, sehingga mampu merepresentasikan tren jangka panjang sekaligus fluktuasi musiman. Tahapan dalam membangun SARIMA mencakup pengujian stasioneritas, identifikasi parameter melalui analisis ACF dan PACF, serta evaluasi model dengan kriteria seperti AIC dan BIC [15]. Meskipun SARIMA efektif dalam menangani pola musiman pada data *time series*, model ini memiliki keterbatasan karena hanya mampu memproses data univariat, sehingga kurang optimal dalam menangkap pengaruh variabel eksternal yang turut memengaruhi hasil prediksi [16].

### 2.4 *Exponential smoothing*

*Exponential smoothing* merupakan pengembangan dari metode *single moving average* yang memberikan bobot lebih besar pada data terbaru dibandingkan data lama [17]. Dengan cara ini, model menjadi lebih responsif terhadap perubahan pola data terkini tanpa sepenuhnya mengabaikan informasi historis. Metode ini memanfaatkan faktor pelicin (*smoothing factor*, $\alpha$) yang bernilai antara 0 hingga 1 untuk mengatur tingkat kepentingan data terbaru terhadap data sebelumnya.

Dalam perkembangannya, *Exponential smoothing* memiliki beberapa varian yang disesuaikan dengan karakteristik data. *Single Exponential smoothing* (SES) digunakan pada data yang relatif stabil tanpa tren maupun musiman, sedangkan *Double Exponential smoothing* (Holt's Method) lebih sesuai untuk data yang menunjukkan adanya tren. Sementara itu, *Triple Exponential smoothing* (Holt-Winters Method) mampu menangkap pola tren sekaligus musiman, baik dalam bentuk aditif maupun multiplikatif. Metode Holt-Winters sendiri menggunakan tiga parameter pelicin, yaitu $\alpha$, $\beta$, dan $\gamma$, yang menentukan sensitivitas

model terhadap data baru, tren, dan pola musiman [18].

### 2.5 Prediksi *Time-Series*

Prediksi merupakan proses memperkirakan peristiwa yang akan terjadi di masa mendatang dengan memanfaatkan informasi historis yang relevan, menggunakan pendekatan yang bersifat ilmiah [19]. Proses prediksi melibatkan analisis data masa lalu untuk menemukan pola, tren, dan hubungan antarvariabel yang dapat digunakan untuk memperkirakan nilai atau kejadian di masa depan. Tujuan utama prediksi adalah memberikan dasar keputusan yang lebih akurat dan efektif, sehingga risiko ketidakpastian dapat diminimalkan dalam perencanaan maupun pengambilan kebijakan. Salah satu pendekatan yang dapat diterapkan dalam prediksi adalah *time series* [20]. Prediksi *time series* bermanfaat dalam berbagai bidang, seperti peramalan cuaca dan iklim, ekonomi, teknik, kesehatan, keuangan, ritel dan bisnis, lingkungan, studi sosial, serta banyak bidang lainnya [21].

### 2.6 Evaluasi Model Prediksi

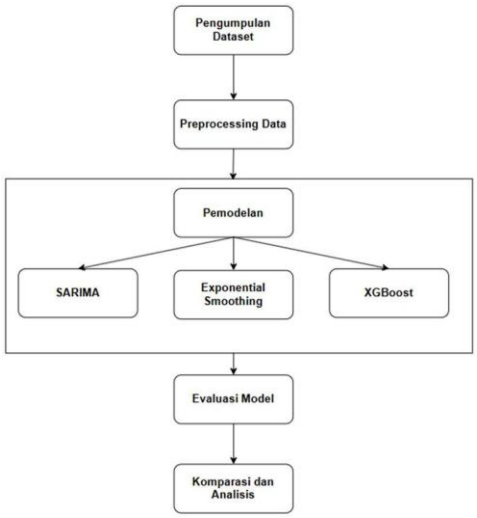
Evaluasi model prediksi sangat penting dilakukan, Evaluasi model memberikan pemahaman tentang seberapa akurat dan dapat diandalkan model tersebut dalam menghasilkan prediksi [22]. Adapun beberapa parameter evaluasi model prediksi yaitu MAE, MSE, MAPE, dan R2-Score.

1. MAE (Mean Absolute Error)
   MAE (Error Absolut Rata-Rata) mengukur besarnya error rata-rata dari prediksi Anda. Metrik ini menghitung selisih absolut (nilai mutlak) antara nilai prediksi dan nilai aktual, lalu merata-ratakannya. Karena menggunakan nilai absolut, MAE tidak mempermasalahkan arah error, apakah prediksi lebih tinggi atau lebih rendah dari nilai sebenarnya. Dengan formula:
   $$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$
   Dimana:
   n adalah jumlah data
   $y_i$ adalah nilai aktual
   $y\hat{}_i$ adalah nilai prediksi
2. MSE (Mean Squared Error)
   MSE (Error Kuadrat Rata-Rata) mirip dengan MAE, MSE menggunakan kuadrat dari selisih antara nilai prediksi dan aktual. Dengan formula:
   $$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$
3. RMSE (Root Mean Squared Error)
   *Root Mean Squared Error* (RMSE) merupakan metrik evaluasi model prediksi yang dihitung sebagai akar kuadrat dari Mean Squared Error (MSE). RMSE mengukur seberapa jauh nilai prediksi menyimpang dari nilai aktual dalam satuan data asli. Dengan formula:
   $$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$
4. $R^2$-Score (R-Squared)
   $R^2$-Score mengukur seberapa baik model regresi yang dibuat cocok dengan data. Secara spesifik, $R^2$ mengukur proporsi varians dari variabel dependen yang dapat dijelaskan oleh variabel independen dalam model. Dengan formula:
   $$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}i)^2}{\sum i = 1^n(y_i - \bar{y})^2}$$
   Nilai $R^2$ berkisaran antara 0 dan 1, dimana ketika $R^2 = 1$ menandakan bahwa model dapat menjelaskan semua variabilitas data, jika $R^2 = 0$ menandakan bahwa model sama sekali tidak dapat menjelaskan variabilitas data, dan jika $R^2 = -$ (negatif) menandakan kinerja model lebih buruk daripada model rata-rata.

## 3. METODE PENELITIAN

Penelitian ini menggunakan metode komparatif, yang bertujuan untuk membandingkan kinerja tiga model prediksi, yaitu SARIMA, *Exponential smoothing*, dan XGBoost, dalam memprediksi penjualan Super Store. Pendekatan ini bersifat kuantitatif, karena mengandalkan pengolahan data numerik dan evaluasi model menggunakan metrik statistik seperti MAE, MAPE, MSE, dan $R^2$-*Score*. Terdapat beberapa tahapan dalam penelitian ini. Tahapan dimulai dengan

pengumpulan dataset, *preprocessing* data, pemodelan, evaluasi model, dan terakhir komparasi dan analisis. Tahapan penelitian dapat dilihat sebagai berikut:




Gambar 1. Tahapan Penelitian

### 3.1 Pengumpulan Dataset

Dataset yang digunakan dalam penelitian ini merupakan dataset penjualan Super Store yang diperoleh dari kaggle. Dataset terdiri dari 21 atribut dimana dalam penelitian ini, penulis hanya berfokus untuk melakukan prediksi terhadap *quantity* produk yang terjual. Dataset yang digunakan memiliki jumlah total 9.994 baris data. Dimana data Super Store yang digunakan merupakan penjualan harian dari tahun 2014 – 2017.

### 3.2 *Preprocessing* Data

Sebelum melakukan proses pemodelan, penting untuk melakukan pembersihan data dan pembagian data untuk *train* dan *test*. Pemrosesan atau *preprocessing* data dilakukan untuk menjamin kualitas data sebelum digunakan dalam pembangunan model [23]. Pada tahap ini, dataset akan dilakukan pengecekan duplikasi serta pemeriksaan dan penanganan terhadap data yang bernilai kosong. Untuk pengecekan duplikasi, penulis menggunakan fungsi *duplicated* () pada *Python*. Fungsi ini memungkinkan penulis untuk menghitung jumlah baris duplikat sekaligus menampilkan contoh baris yang terduplikasi, sehingga dapat memastikan dataset bersih sebelum digunakan untuk pemodelan. Penanganan nilai kosong menggunakan fungsi *isnull*() pada *Python*. Fungsi ini memungkinkan untuk menghitung jumlah nilai yang hilang pada setiap atribut, sehingga dapat diketahui persentase data yang kosong dan dilakukan penanganan yang sesuai sebelum pemodelan.

Setelah dataset di bersihkan, atribut seperti *order date* akan di pecah menjadi *year* dan *month*. Atribut lain seperti *sales*, *discount*, *profit* dan *quantity* produk yang terjual diambil untuk keperluan prediksi. Terdapat cacatan pada penelitian ini bahwa pada model *time series* tradisional seperti SARIMA dan *Exponential smoothing*, prediksi dilakukan secara *univariat* hanya dengan memanfaatkan data historis *quantity* sebagai input, sedangkan informasi *year*, *month* digunakan sebatas sebagai indeks waktu. Sementara itu, pada XGBoost, fitur tambahan seperti *year*, *month*, *sales*, *discount*, *profit* digunakan sebagai variabel input bersama dengan *quantity* historis. Hal ini dilakukan karena XGBoost bukan model khusus *time series,* sehingga memerlukan *feature engineering* untuk dapat menangkap pola tren maupun musiman.

### 3.3 Pemodelan

Pada tahap pemodelan, dalam penelitian ini digunakan tiga model yaitu SARIMA, *Exponential smoothing*, dan XGBoost. Sebelum pemodelan, dataset dibagi menjadi data latih (*training)* dan data uji (*testing*) agar kinerja setiap model dapat dievaluasi secara objektif. Pemodelan dengan SARIMA dilakukan dengan menentukan parameter orde *autoregresif* (p), *differencing* (d), moving *average* (q), serta komponen musiman (P, D, Q, s) yang sesuai dengan pola data sehingga model dapat menangkap tren maupun pola musiman secara optimal. *Exponential smoothing* diterapkan dengan melakukan estimasi komponen level, tren, dan musiman, di mana setiap pengamatan historis diberikan bobot eksponensial sehingga data terbaru memiliki pengaruh lebih besar dalam prediksi. Sementara itu, XGBoost digunakan sebagai representasi metode *machine learning* berbasis *gradient boosting*, dengan proses pelatihan dilakukan melalui penyesuaian *hyperparameter* untuk menghasilkan model prediktif yang mampu menangani hubungan *non-linear* dan kompleksitas data.

### 3.4 Evaluasi Model

Pada tahap evaluasi, kinerja ketiga model yang telah dilatih dan diuji selanjutnya diukur menggunakan beberapa metrik evaluasi. Evaluasi dilakukan untuk mengukur sejauh mana model yang diterapkan berjalan secara efektif [24]. *Mean Absolute Error* (MAE) digunakan untuk menghitung rata-rata selisih absolut antara nilai prediksi dengan nilai aktual tanpa memperhatikan arah kesalahan. *Mean Squared Error* (MSE**)** mengukur rata-rata kuadrat dari selisih prediksi dan nilai aktual, sehingga kesalahan yang besar akan diberi penalti lebih tinggi. *Root Mean Squared Error* (RMSE**)** merupakan akar kuadrat dari MSE yang memberikan gambaran tingkat kesalahan dalam satuan yang sama dengan data asli, sehingga lebih mudah diinterpretasikan. Sementara itu, *R-Squared* ($R^2$ *Score*) digunakan untuk menilai seberapa baik model mampu menjelaskan variabilitas data aktual, di mana nilai yang lebih mendekati 1 menunjukkan kualitas prediksi yang lebih baik. Hasil dari evaluasi ini menjadi dasar untuk membandingkan performa SARIMA, *Exponential smoothing*, dan XGBoost dalam menghasilkan prediksi yang akurat.

### 3.5 Komparasi dan Analisis

Pada tahap komparasi dan analisis, hasil evaluasi dari ketiga model dibandingkan berdasarkan nilai metrik MAE, MSE, RMSE, dan $R^2$. Proses komparasi ini bertujuan untuk mengetahui model mana yang memberikan performa terbaik dalam melakukan prediksi. Nilai error yang lebih kecil pada MAE, MSE, dan RMSE menunjukkan bahwa model lebih akurat dalam menghasilkan prediksi, sedangkan nilai $R^2$ yang lebih mendekati 1 menunjukkan kemampuan model yang lebih baik dalam menjelaskan variabilitas data aktual. Setelah perbandingan dilakukan, analisis dilanjutkan untuk meninjau keunggulan dan kelemahan masing-masing model. Model *time series* tradisional seperti SARIMA dan *Exponential smoothing* dapat memberikan interpretasi yang jelas terhadap tren dan musiman, sementara XGBoost lebih unggul dalam menangani hubungan *non-linear* dan memanfaatkan variabel eksternal. Hasil analisis ini menjadi dasar dalam menentukan model yang paling sesuai untuk digunakan pada penelitian.

## 4. HASIL DAN PEMBAHASAN

Penelitian ini bertujun untuk memprediksi penjualan Super Store berdasarkan *quantity* produk yang terjual.

### 4.1 Dataset

Tahap pertama yang dilakukan adalah pemilihan dataset, yaitu menentukan data yang akan digunakan dalam penelitian ini. Data tersebut kemudian diimpor ke dalam Python menggunakan *library pandas*, sehingga dapat diketahui deskripsi dari data yang diperoleh. Berikut merupakan datas yang diperoleh:

```
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Row ID         9994 non-null   int64
 1   Order ID       9994 non-null   object
 2   Order Date     9994 non-null   object
 3   Ship Date      9994 non-null   object
 4   Ship Mode      9994 non-null   object
 5   Customer ID    9994 non-null   object
 6   Customer Name  9994 non-null   object
 7   Segment        9994 non-null   object
 8   Country        9994 non-null   object
 9   City           9994 non-null   object
 10  State          9994 non-null   object
 11  Postal Code    9994 non-null   int64
 12  Region         9994 non-null   object
 13  Product ID     9994 non-null   object
 14  Category       9994 non-null   object
 15  Sub-Category   9994 non-null   object
 16  Product Name   9994 non-null   object
 17  Sales          9994 non-null   float64
 18  Quantity       9994 non-null   int64
 19  Discount       9994 non-null   float64
 20  Profit         9994 non-null   float64
```

Gambar 2. Dataset

### 4.2 Preprocessing Data

Pada tahap *preprocessing* data, dilakukan pengecekan untuk memastikan tidak terdapat duplikasi maupun *missing value*. Selanjutnya dipilih atribut yang digunakan dalam proses prediksi, yaitu *year*, *month*, dan *quantity*. Penelitian ini berfokus pada prediksi jumlah (*quantity*) produk Super Store yang terjual. Terakhir, dataset dibagi menjadi dua bagian, yaitu data latih dan data uji.

a. Pengecekan Duplikasi

```
total_duplicate_rows = sales_df.duplicated().sum()
duplicate_rows_sample = sales_df[sales_df.duplicated()].head()
```

Gambar 3. Fungsi Mengecek Duplikasi

Kode ini digunakan untuk mendeteksi data duplikat pada *DataFrame sales_df*. Baris

pertama menghitung total baris duplikat, sedangkan baris kedua menampilkan contoh beberapa baris duplikat (default 5 baris pertama) agar bisa diperiksa lebih lanjut.

```
print(f"Total baris duplikat secara keseluruhan: {total_duplicate_rows}\n")
```

```
Total baris duplikat secara keseluruhan: 0
```

Gambar 4. Hasil Pengecekan Duplikasi

Didapatkan bahwa pada dataset tidak terdapat duplikasi data.

b. Pengecekan *Missing Value*

Pengecekan *missing value* penting, karena data yang hilang bisa mempengaruhi akurasi model.

```
missing_values_count = sales_df.isnull().sum()
total_rows = len(sales_df)
missing_values_percentage = (missing_values_count / total_rows) * 100
```

Gambar 5. Fungsi Mengecek *Missing Value*

Kode ini digunakan untuk mengecek data yang hilang (*missing value*) pada *DataFrame sales_df*. Baris pertama menghitung jumlah missing value per kolom, baris kedua mendapatkan total jumlah baris, dan baris ketiga menghitung persentase missing value dari setiap kolom terhadap total data. Hasil pengecekan *missing value* dapat dilihat pada Gambar 6.

| | Missing Count | Missing Percentage |
|---|---|---|
| Row ID | 0 | 0.00% |
| Order ID | 0 | 0.00% |
| Order Date | 0 | 0.00% |
| Ship Date | 0 | 0.00% |
| Ship Mode | 0 | 0.00% |
| Customer ID | 0 | 0.00% |
| Customer Name | 0 | 0.00% |
| Segment | 0 | 0.00% |
| Country | 0 | 0.00% |
| City | 0 | 0.00% |
| State | 0 | 0.00% |
| Postal Code | 0 | 0.00% |
| Region | 0 | 0.00% |
| Product ID | 0 | 0.00% |
| Category | 0 | 0.00% |
| Sub-Category | 0 | 0.00% |
| Product Name | 0 | 0.00% |
| Sales | 0 | 0.00% |
| Quantity | 0 | 0.00% |
| Discount | 0 | 0.00% |
| Profit | 0 | 0.00% |

Gambar 6. Hasil Pengecekan *Missing Value*

c. Pemilihan Atribut Data

Pada tahap pemilihan atribut, atribut yang digunakan adalah *year*, *month*, *sales*, *discount*, *profit* dan *quantity*. Karena atribut *year* dan *month* belum tersedia, kolom *order date* diolah terlebih dahulu untuk mengekstrak informasi *year* dan *month* sebelum data dibagi. Jumlah data yang didapatkan setelah mengekstrak *order date* sebanyak 48 bulan, yang awalnya berjumlah 9.994 data penjualan harian. Dimana data 48 ini digunakan untuk pemodelan.

```
sales_df['Order Date'] = pd.to_datetime(sales_df['Order Date'], errors='coerce')

def get_month(inpDate):
    return inpDate.month

def get_year(inpDate):
    return inpDate.year

sales_df['Month'] = sales_df['Order Date'].apply(get_month)
sales_df['Year'] = sales_df['Order Date'].apply(get_year)
```

Gambar 7. Mengekstrak *Order Date*

Setelah data berhasil di ekstrak, kolom *year, month, sales, discount, profit,* dan *quantity* siap digunakan untuk analisis dan pemodelan. Mengelompokkan penjualan berdasarkan periode, menghitung total *quantity* per bulan.

| | Year | Month | Quantity | Sales | Profit | Discount |
|---|---|---|---|---|---|---|
| 0 | 2014 | 1 | 284 | 14236.895 | 2450.1907 | 0.126582 |
| 1 | 2014 | 2 | 159 | 4519.892 | 862.3084 | 0.176087 |
| 2 | 2014 | 3 | 585 | 55691.009 | 498.7299 | 0.167516 |
| 3 | 2014 | 4 | 536 | 28295.345 | 3488.8352 | 0.110000 |
| 4 | 2014 | 5 | 466 | 23648.287 | 2738.7096 | 0.155328 |

Gambar 8. Contoh Data Setelah Diekstrak

d. Pembagian Dataset

Tahap terakhir dari *preprocessing* data adalah pembagian dataset menjadi dua bagian, yaitu data latih dan data uji. Proporsi pembagian dilakukan sebesar 80% untuk data latih dan 20% untuk data uji. Data latih digunakan untuk melatih model sehingga dapat mempelajari pola dan tren dalam dataset, sedangkan data uji digunakan untuk mengevaluasi performa model pada data yang belum pernah dilihat sebelumnya, sehingga hasil prediksi dapat dinilai secara objektif. Fungsi pembagian dataset dapat dilihat pada gambar 9.

```
train_size = int(len(sales_qt_df) * 0.8)
train = sales_qt_df['Quantity'][:train_size]
test = sales_qt_df['Quantity'][train_size:]
```

Gambar 9. Fungsi Pembagian Dataset

### 4.3 *Train*ing dan *Test*ing

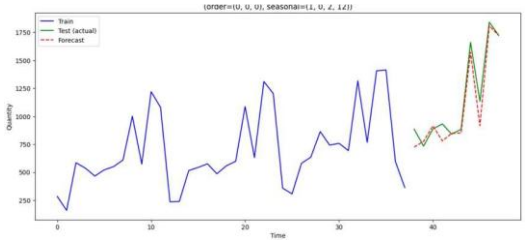
Pada tahap ini, akan dilakukan pembuatan model menggunakan tiga metode yaitu

SARIMA, *Exponential smoothing*, dan XGBoost, untuk membandingkan efektivitas membandingkan moodel dengan metode *time series* tradisional dan model berbasis *machine learning* dalam memprediksi data penjualan berdasarkan *quantity* produk yang terjual.

### 4.3.1 SARIMA

Model SARIMA dibangun menggunakan library *statsmodels*, dengan pencarian parameter terbaik dilakukan secara otomatis melalui pendekatan *grid search* pada kombinasi orde $(p, d, q)(P, D, Q)_s$. Proses pencarian dilakukan dengan mencoba berbagai kombinasi parameter non-musiman (p,d,q) serta musiman (P,D,Q) dengan periode musiman tertentu, kemudian setiap model dilatih menggunakan data latih dan diuji pada data uji. Hasil prediksi dibandingkan dengan data aktual pada data uji, sehingga diperoleh ukuran performa model yang objektif.

Proses pencarian parameter menghasilkan model SARIMA terbaik dengan kombinasi orde $(0,0,0)(1,0,2)_{12}$. Model ini dipilih karena memberikan nilai kesalahan terkecil dibandingkan kombinasi lainnya. Hasil evaluasi menunjukkan nilai MAE sebesar 76,125, MSE sebesar 10.849,44, RMSE sebesar 104,161, serta $R^2$ sebesar 0,932. Nilai $R^2$ yang mendekati 1 mengindikasikan bahwa model mampu menjelaskan sebagian besar variasi pada data penjualan. Visualisasi hasil prediksi model SARIMA dapat dilihat pada Gambar 10.



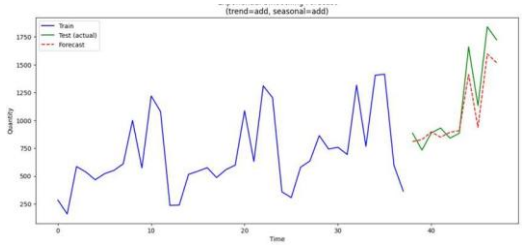

Gambar 10. Hasil Prediksi Model SARIMA

Grafik tersebut menunjukkan hasil pemodelan SARIMA (0,0,0), (1,0,2)12 pada data penjualan. Garis biru merepresentasikan data latih (*train*), garis hijau menunjukkan data uji (*test*/aktual), sedangkan garis merah putus-putus menggambarkan hasil prediksi (*forecast*) dari model.

### 4.3.2 *Exponential smoothing*

Model *Exponential smoothing* (Holt-Winters) dibangun menggunakan library *statsmodels* dengan konfigurasi trend aditif (additive trend) dan musiman aditif (additive seasonal**)** dengan periode musiman sebesar 12. Prediksi yang dihasilkan dari model kemudian dibandingkan dengan data aktual pada periode pengujian untuk menilai kinerja model.

Hasil evaluasi menunjukkan bahwa model *Exponential smoothing* menghasilkan MAE sebesar 122,743, MSE sebesar 22.539,392, RMSE sebesar 150,131, serta $R^2$ sebesar 0,859. Nilai $R^2$ ini menunjukkan bahwa model cukup mampu menjelaskan variasi data penjualan, meskipun tingkat kesalahannya lebih tinggi dibandingkan SARIMA. Hal ini sesuai dengan karakteristik Holt-Winters yang lebih efektif pada data dengan pola musiman sederhana, namun cenderung kurang optimal dalam menangkap variasi data yang lebih kompleks. Visualisasi hasil prediksi model *Exponential smoothing* dapat dilihat pada Gambar 11.




Gambar 11. Hasil Prediksi Model SARIMA

Grafik tersebut menunjukkan hasil pemodelan *Exponential smoothing* **(**trend=add, seasonal=add, seasonal_periods=12) pada data penjualan. Garis biru merepresentasikan data latih (*train*), garis hijau menunjukkan data uji (aktual), sedangkan garis merah putus-putus menggambarkan hasil prediksi (*forecast*) dari model.

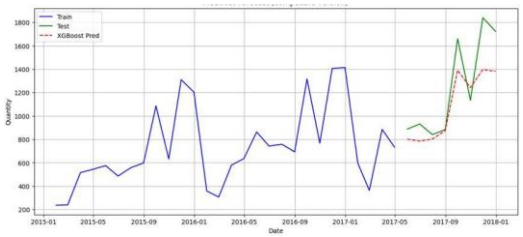
### 4.3.3 XGBoost

Model XGBoost dibangun menggunakan library *xgboost* dengan pendekatan *supervised learning*. Sebelum pemodelan, dilakukan proses *feature engineering* yang mencakup pembuatan variabel *lag* hingga 12 periode sebelumnya, rata-rata dan standar deviasi bergulir, transformasi musiman menggunakan fungsi sinus dan cosinus, serta variabel turunan seperti *profit margin*, *sales per quantity*, dan

interaksi antara diskon dengan penjualan. Variabel-variabel ini digunakan untuk memperkaya informasi sehingga model dapat menangkap pola kompleks dalam data penjualan.

Pemilihat hyperparameter terbaik dilakukan menggunakan *RandomizedSearchCV* pada parameter *n_estimators*, *learning_rate*, *max_depth*, *subsample*, *colsample_bytree*, dan *gamma*. Model terbaik akan digunakan untuk menghasilkan prediksi pada data uji, yang selanjunya akan dibandingkan dengan data aktual.

Hasil evaluasi menunjukkan bahwa model XGBoost menghasilkan MAE sebesar 179,928, MSE sebesar 53.383,938, RMSE sebesar 231,050, dan $R^2$ sebesar 0,668. Nilai $R^2$ ini relatif lebih rendah dibandingkan SARIMA dan *Exponential smoothing,* yang menunjukkan bahwa XGBoost kurang optimal dalam kasus ini. Visualisasi hasil prediksi model XGBoost dapat dilihat pada Gambar 12.




Gambar 12. Hasil Prediksi Model XGBoost

Grafik tersebut menunjukkan hasil pemodelan XGBoost pada data penjualan. Garis biru merepresentasikan data latih (*train*), garis hijau menunjukkan data uji aktual (*test*), sedangkan garis merah putus-putus merupakan hasil prediksi model XGBoost.

### 4.4 Komparasi dan Analisis Model

Setelah dilakukan pemodelan dari ketiga metode yaitu SARIMA, *Exponential smoothing,* dan XGBoost dilakukan evaluasi terhadap performa masing-masing model berdasarkan empat metriks yaitu MAE, MSE, RSME, dan $R^2$. Hasil dari ketiga model tersebut dapat dilihat pada Gambar 13.

| Model | MAE | MSE | RSME | $R^2$ |
|---|---|---|---|---|
| SARIMA | 76,125 | 10.849,44 | 104,161 | 0,932 |
| Exponential Smoothing | 122,743 | 22.539,392 | 150,131 | 0,859 |
| XGBoost | 179,928 | 53.383,938 | 231,050 | 0,668 |

Gambar 13. Hasil Evaluasi Ketiga Model

Berdasarkan hasil perbandingan pada tabel di atas, dapat dilihat bahwa model *time series* tradisional menunjukkan kinerja yang lebih baik dibandingkan pendekatan *machine learning* dalam kasus ini. SARIMA menampilkan performa yang paling optimal, diikuti oleh *Exponential smoothing*, sedangkan XGBoost menghasilkan performa paling rendah.

SARIMA menunjukkan keunggulan karena secara alami mampu menangkap autokorelasi dan pola musiman dalam deret waktu tanpa memerlukan banyak fitur tambahan. Sebaliknya, meskipun XGBoost cukup fleksibel, model ini sering kesulitan mengenali pola musiman yang kuat apabila data historis terbatas dan variasi antar periode tidak terlalu kompleks [25]. Sementara itu, *Exponential smoothing*, efektif untuk data dengan tren dan musiman sederhana, namun dibandingkan dengan SARIMA, metode ini kurang sensitif terhadap perubahan musiman yang halus atau anomali tertentu, sehingga kinerjanya biasanya sedikit lebih rendah pada data yang stabil dan memiliki pola musiman yang jelas.

## 5. KESIMPULAN

a. Sistem prediksi penjualan berbasis tiga metode utama, yaitu SARIMA, *Exponential smoothing*, dan XGBoost, telah berhasil diimplementasikan dan diuji menggunakan data penjualan bulanan periode 2014–2017. Hasil pengujian menunjukkan bahwa setiap model memiliki karakteristik performa yang berbeda dalam menangkap pola historis data penjualan. Model SARIMA mampu memberikan hasil paling akurat dengan nilai kesalahan terkecil, diikuti oleh *Exponential smoothing*, sementara XGBoost menunjukkan performa yang lebih rendah. Proses pemodelan ini mencakup tahap *feature engineering*, pemisahan data latih dan uji, serta evaluasi menggunakan metrik MAE,

MSE, RMSE, dan $R^2$ untuk menilai efektivitas model secara objektif.

b. Keunggulan penelitian ini terletak pada keberhasilan untuk menunjukkan setiap metode memiliki keunggulan dan keterbatasan yang berbeda sesuai karakteristik data penjualan. SARIMA terbukti paling unggul karena mampu mengenali pola musiman yang berulang secara konsisten tanpa memerlukan banyak variabel tambahan. Model *Exponential smoothing* memberikan hasil yang cukup baik, meskipun tingkat kesalahannya lebih tinggi karena kesulitannya menyesuaikan fluktuasi tajam pada data. Sementara itu, XGBoost kurang optimal karena model ini cenderung membutuhkan data dalam jumlah besar dan variasi fitur yang lebih kompleks agar dapat belajar dengan efektif.

c. Meskipun demikian, terdapat beberapa keterbatasan dalam penelitian ini. Model SARIMA memerlukan proses pencarian parameter yang relatif kompleks dan waktu pelatihan yang lebih lama. *Exponential smoothing*, walaupun sederhana, memiliki keterbatasan dalam menangkap perubahan pola musiman yang dinamis. Sementara itu, XGBoost membutuhkan feature engineering yang ekstensif dan volume data yang besar agar performanya optimal. Untuk pengembangan selanjutnya, disarankan untuk menggabungkan pendekatan hybrid untuk memanfaatkan keunggulan masing-masing model.

## UCAPAN TERIMA KASIH

Penulis menyampaikan terima kasih kepada semua pihak yang terkait telah memberikan dorongan dan dukungan terhadap penelitian ini.

## DAFTAR PUSTAKA

[1] N. P. N. P. Dewi and R. A. Nugroho, "Optimasi general regression neural network dengan fruit fly optimization algorithm untuk prediksi pemakaian arus listrik pada penyulang," *Komputasi: Jurnal Ilmiah Ilmu Komputer dan Matematika*, vol. 18, no. 1, pp. 1–12, 2021.

[2] D. P. H. Putri, N. P. N. P. Dewi, I. K. Purnamawan, and N. W. Marti, "Perbandingan Performansi Support Vector Machine (Svm) dan Backpropagation untuk Klasifikasi Studi Mahasiswa Undiksha," *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, vol. 9, no. 3, pp. 492–501, 2023.

[3] V. Rao, A. D. Singh, and M. P. Kumar, "Advanced AI and Machine Learning Techniques for Time Series Analysis and Pattern Recognition," *Applied Sciences*, vol. 15, no. 6, 2023.

[4] T. Falatouri, F. Darbanian, P. Brandtner, and C. Udokwu, "Predictive analytics for demand forecasting–a comparison of SARIMA and LSTM in retail SCM," *Procedia Computer Science*, vol. 200, pp. 993–1003, 2022.

[5] M. N. İnce and Ç. Taşdemir, "Forecasting retail *sales* for furniture and furnishing items through the employment of multiple linear regression and holt–winters models," *Systems*, vol. 12, no. 6, pp. 219, 2024.

[6] O. O. Mustapha and T. Sithole, "Forecasting Retail *Sales* using Machine Learning Models," *American Journal of Statistics and Actuarial Sciences*, vol. 6, no. 1, pp. 35–67, 2025.

[7] N. Sunendar, H. P. Putro, and R. Hesananda, "Prediksi Penjualan Aerosol Menggunakan Algoritma ARIMA, LSTM Dan GRU," *INSOLOGI: Jurnal Sains dan Teknologi*, vol. 4, no. 1, pp. 113–126, 2025.

[8] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Forecasting with statistical and machine learning methods: The M4 and M5 competitions and how to improve forecasting accuracy," *International Journal of Forecasting*, vol. 38, no. 1, pp. 26–39, 2022.

[9] R. Diana, H. Warni, and T. Sutabri, "Penggunaan teknologi machine learning untuk pelayanan monitoring kegiatan belajar mengajar pada SMK Bina Sriwijaya Palembang," *Jurnal Teknik Informatika (JUTEKIN)*, vol. 11, no. 1, 2023.

[10] M. J. Rahaman, "A comprehensive review to understand the definitions, advantages, disadvantages and applications of machine learning

algorithms," *Int J Comput Appl*, vol. 186, no. 31, pp. 43–47, 2024.

[11] Q. Lu, S. Taimourzadeh, P. J. Fitzgerald, A. Harzand, J. McCaney, and J. B. Prillinger, "Machine learning methods application: generating clinically meaningful insights from healthcare survey data," *Journal of Medical Artificial Intelligence*, vol. 8, 2025.

[12] B. Liao, T. Zhou, Y. Liu, M. Li, and T. Zhang, "Tackling the Wildfire Prediction Challenge: An Explainable Artificial Intelligence (XAI) Model Combining Extreme Gradient Boosting (XGBoost) with SHapley Additive exPlanations (SHAP) for Enhanced Interpretability and Accuracy," *Forests*, vol. 16, no. 4, pp. 689, 2025.

[13] H. Wijaya, D. P. Hostiadi, and E. Triandini, "Meningkatkan prediksi penjualan retail XYZ dengan teknik optimasi radom search pada model XGBoost", *Seminar Hasil Penelitian Informatika dan Komputer* (SPINTER), pp 829-833. 2024.

[14] H. Kuswanto, P. E. P. Utomo, U. Khaira, and A. Waladi, "Prediksi Nilai Ekspor Migas Indonesia Menggunakan Metode SARIMA dan LSTM," *SATESI: Jurnal Sains Teknologi dan Sistem Informasi*, vol. 5, no. 1, pp. 69–79, 2025.

[15] A. Ermawati, A. Amrullah, K. Huda, and M. A. Haris, "Implementasi Metode Seasonal Autoregressive Integrated Moving Average (SARIMA) untuk Memprediksi Curah Hujan di Kota Semarang," *Jurnal Statistika dan Komputasi*, vol. 3, no. 2, pp. 62–71, 2024.

[16] N. P. N. P. Dewi, Y. Leu, K. Mustofa, and M. Riasetiawan, "Enhancing Diesel Backup Power Forecasting With LSTM, GRU, and Autoencoder-based Input Encoding," *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, vol. 14, no. 1, 2025.

[17] A. Aliniy, Y. P. Pasrun, and A. T. Sumpala, "Prediksi Jumlah Mahasiswa Baru Fti Usn Kolaka Menggunakan Metode Single *Exponential smoothing*," *SATESI: Jurnal Sains Teknologi dan Sistem Informasi*, vol. 3, no. 1, pp. 20–25, 2023.

[18] I. P. S. Handika and I. K. S. Satwika, "Enhancing *Sales* Forecasting Accuracy Through Optimized Holt-Winters *Exponential smoothing* with Modified Improved Particle Swarm Optimization," *JANAPATI: Jurnal Nasional Pendidikan Teknik Informatika*, vol. 12, no. 2, 2024.

[19] A. Damayanti, F. D. Marleny, and A. A. Ningrum, "Implementasi Regresi Linier Berganda Untuk Prediksi Penjualan Pada Pt Trimandiri Sarana Propetindo Banjarmasin," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 3, 2025.

[20] M. W. Aditya, I. N. Sukajaya, and I. G. A. Gunadi, "Forecasting Jumlah Pasien DBD di BRSUD Kabupaten Tabanan Menggunakan Metode Regresi Linier," *Bali Medika Jurnal*, vol. 10, no. 1, pp. 1–12, 2023.

[21] N. P. N. P. Dewi, N. K. Kertiasih, and N. L. D. Sintiari, "Modifikasi Fruit Fly Optimiziation Algorithm untuk Optimasi General Regression Neural Network pada Kasus Prediksi Time-Series," *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, vol. 11, no. 3, pp. 192–204, 2022.

[22] A. Yulianto, R. Adiperkasa, and Y. Farida, "Analisis Prediksi Harga Smartphone Tahun 2023 Menggunakan Model Random Forest Regression Berdasarkan Fitur," *Jurnal Komputer dan Teknologi dan Sistem Informasi (KOMTEKS)*, vol. 3, no. 1, pp. 58–69, 2024.

[23] I. K. N. Ananda, N. P. N. P. Dewi, N. W. Marti, and L. J. E. Dewi, "Klasifikasi Multilabel pada Gaya Belajar Siswa Sekolah Dasar menggunakan Algoritma Machine Learning," *Journal of Applied Computer Science and Technology*, vol. 5, no. 2, pp. 144–154, 2024.

[24] N. W. Y. Wiani, I. M. A. Wirawan, and K. Y. E. Aryanto, "Klasifikasi Gerakan Tangan Berbasis Sinyal sEMG Menggunakan Deep Learning," *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, vol. 11, no. 1, pp. 121–128, 2025.

[25] O. Ozdemir and C. Yozgatligil, "Forecasting performance of machine learning, time series, and hybrid methods for low-and high- frequency time series," *Statistica Neerlandica*, vol. 78, no. 2, pp. 441–474, 2024.