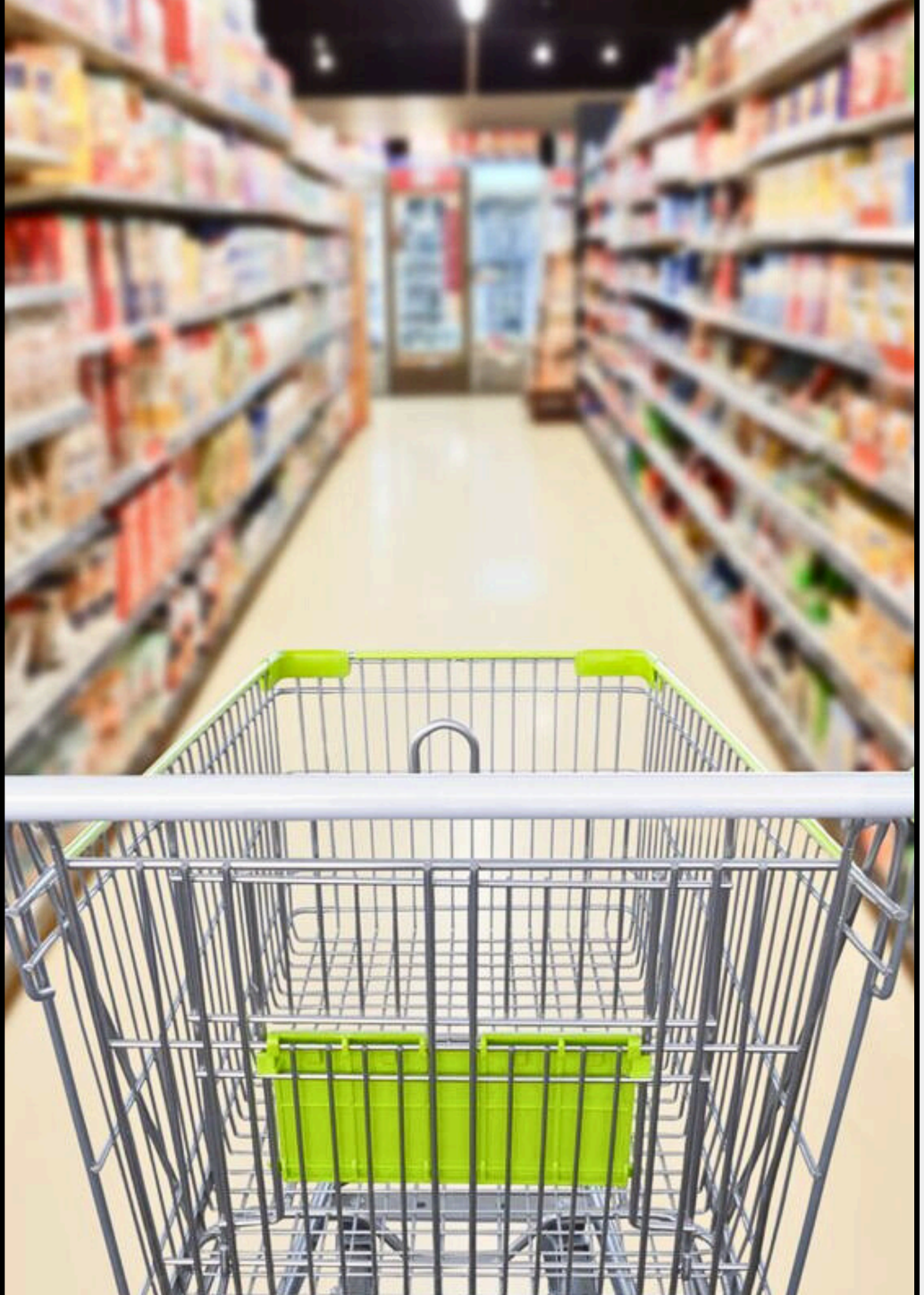
MARKET PRICE PREDICTION

A TIME SERIES ANALYSIS

MAHAMMAD BASHEER K S
MENTORNESS



Introduction

In market analysis and forecasting, understanding time series data is crucial for making informed decisions. With machine learning, we can analyze historical market data to predict future trends accurately. This project aims to develop a robust time series forecasting model to predict commodity quantities and prices for future months. Utilizing advanced algorithms, we seek to empower stakeholders to make proactive decisions regarding production, procurement, pricing strategies, and resource allocation.

Objective

- Develop a robust time series machine learning model.
- Accurately forecast market trends using historical data.
- Leverage advanced algorithms for prediction.
- Predict future quantities and prices of commodities.
- Empower stakeholders with actionable forecasts.
- Enhance decision-making for production and procurement.
- Optimize pricing strategies and resource allocation.

Data Description

Dataset Columns:

- Market
- Month
- Year
- Quantity
- PriceMin
- PriceMax
- Price Mod
- State
- City
- Date

```
In [2]: # Loading the dataset
        data=pd.read_csv('MarketPricePrediction.csv')

In [3]: data.head(10)
```

Out[3]:

	market	month	year	quantity	priceMin	priceMax	priceMod	state	city	date
0	ABOHAR(PB)	January	2005	2350	404	493	446	PB	ABOHAR	January-2005
1	ABOHAR(PB)	January	2006	900	487	638	563	PB	ABOHAR	January-2006
2	ABOHAR(PB)	January	2010	790	1283	1592	1460	PB	ABOHAR	January-2010
3	ABOHAR(PB)	January	2011	245	3067	3750	3433	PB	ABOHAR	January-2011
4	ABOHAR(PB)	January	2012	1035	523	686	605	PB	ABOHAR	January-2012
5	ABOHAR(PB)	January	2013	675	1327	1900	1605	PB	ABOHAR	January-2013
6	ABOHAR(PB)	January	2014	440	1025	1481	1256	PB	ABOHAR	January-2014
7	ABOHAR(PB)	January	2015	1305	1309	1858	1613	PB	ABOHAR	January-2015
8	ABOHAR(PB)	February	2005	1400	286	365	324	PB	ABOHAR	February-2005
9	ABOHAR(PB)	February	2006	1800	343	411	380	PB	ABOHAR	February-2006

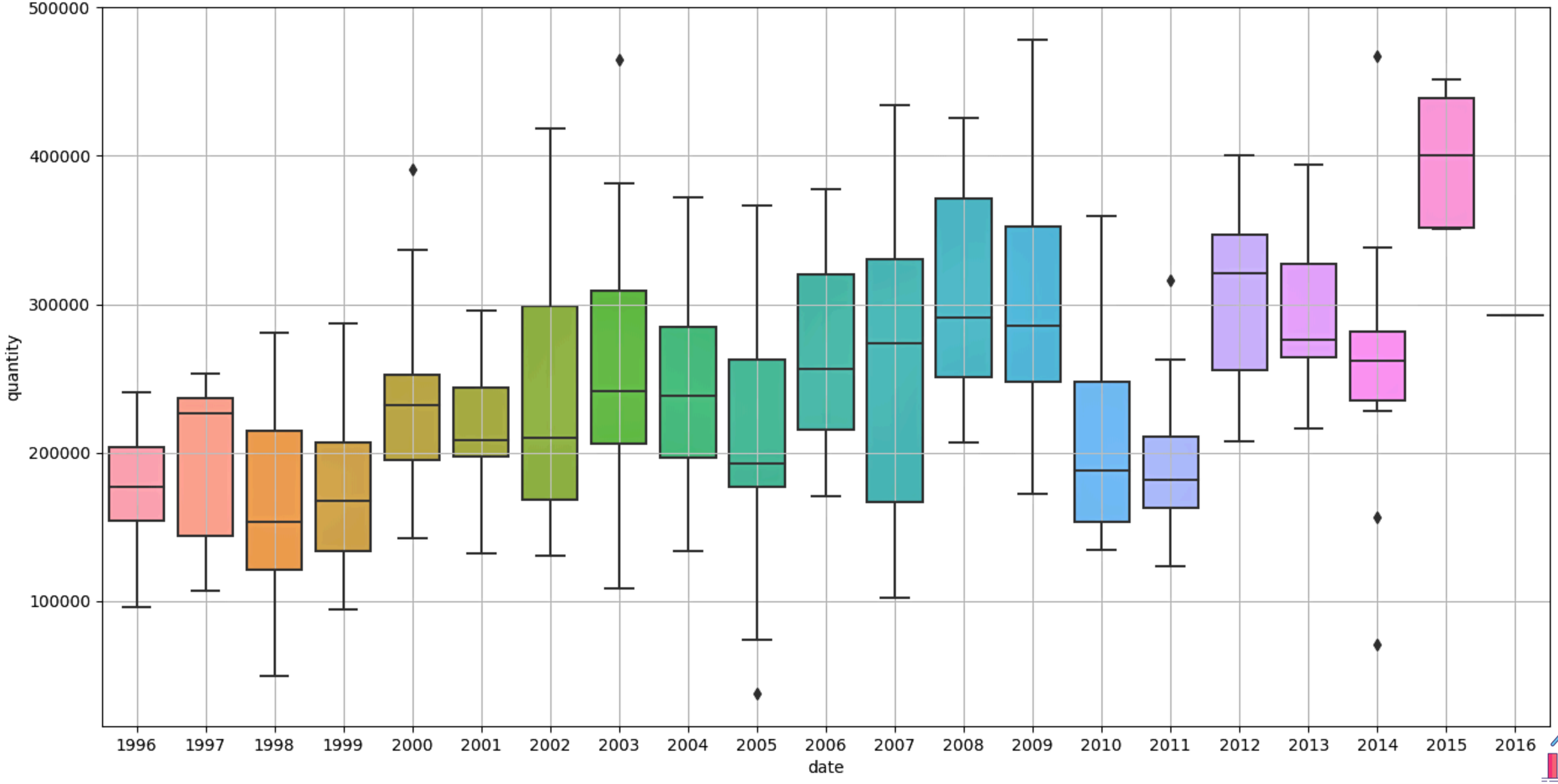
```
In [4]: data.shape
Out[4]: (10227, 10)
```

Data Preprocessing

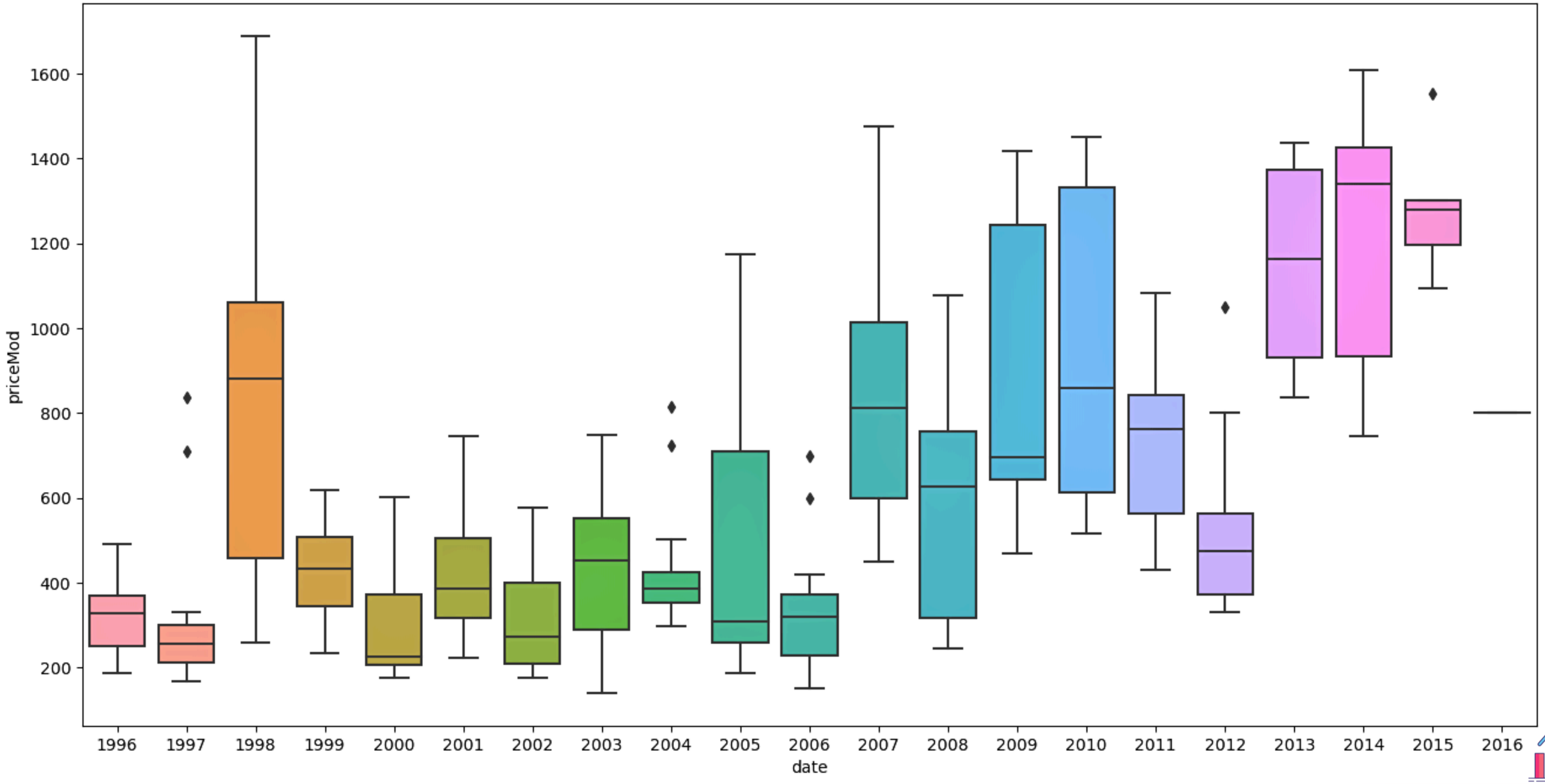
- Ensure data integrity by identifying and handling any null values.
- Transform the Date column's datatype from object to datetime64[ns] for accurate time-based analysis.

```
In [5]: # Checking for missing values in the training dataset
        data.isnull().sum()

Out[5]: market      0
        month       0
        year        0
        quantity    0
        priceMin    0
        priceMax    0
        priceMod    0
        state       0
        city        0
        date        0
        dtype: int64

In [6]: # Displaying information about the training dataset
        data.info()

        <class 'pandas.core.frame.DataFrame'>
        RangeIndex: 10227 entries, 0 to 10226
        Data columns (total 10 columns):
         #   Column    Non-Null Count  Dtype
        ---  ------    --------------  -----
         0   market    10227 non-null  object
         1   month     10227 non-null  object
         2   year      10227 non-null  int64
         3   quantity  10227 non-null  int64
         4   priceMin  10227 non-null  int64
         5   priceMax  10227 non-null  int64
         6   priceMod  10227 non-null  int64
         7   state     10227 non-null  object
         8   city      10227 non-null  object
         9   date      10227 non-null  object
        dtypes: int64(5), object(5)
        memory usage: 799.1+ KB

In [7]: data['date']=pd.to_datetime(data['date'])
```

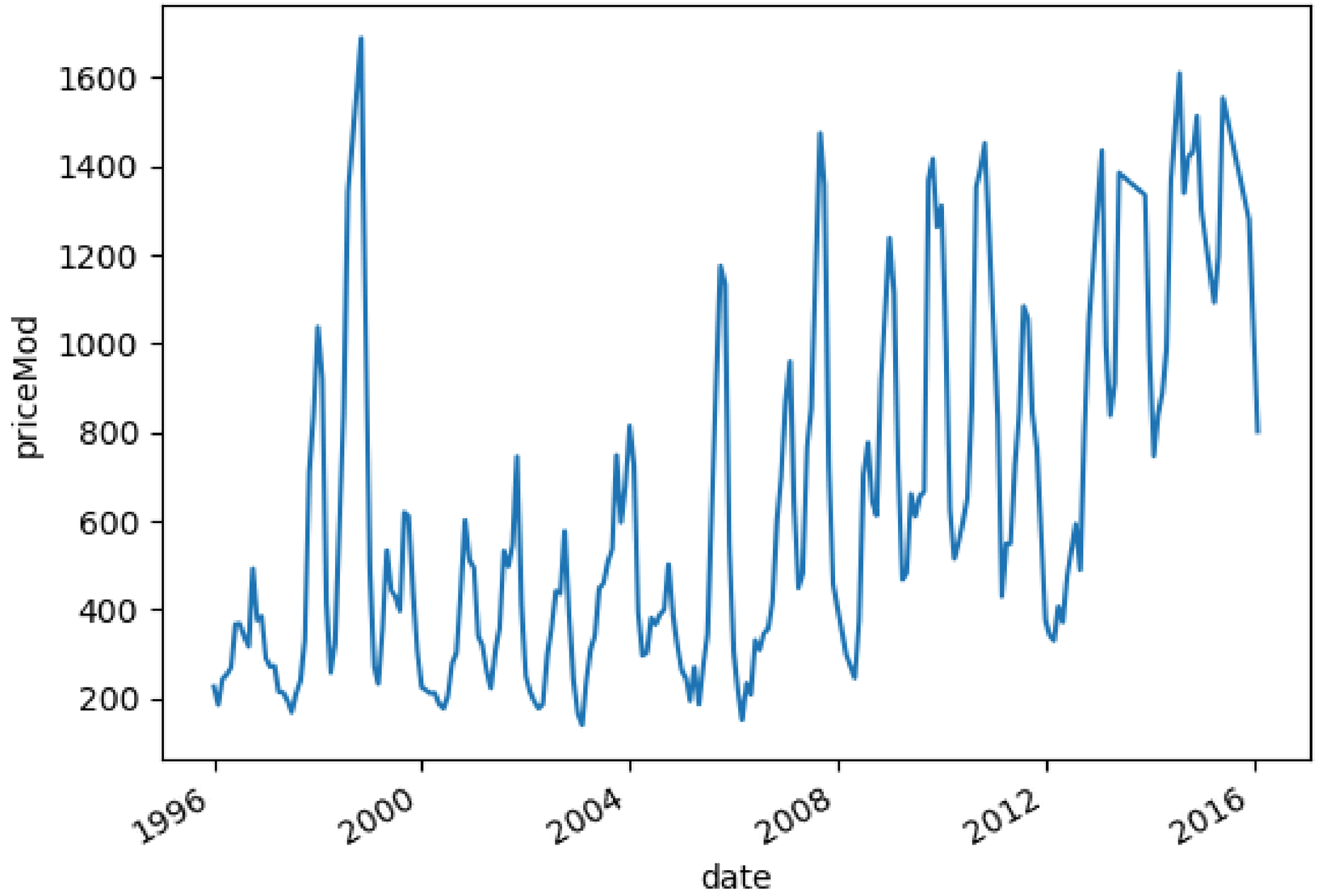
Exploratory Data Analysis (EDA)

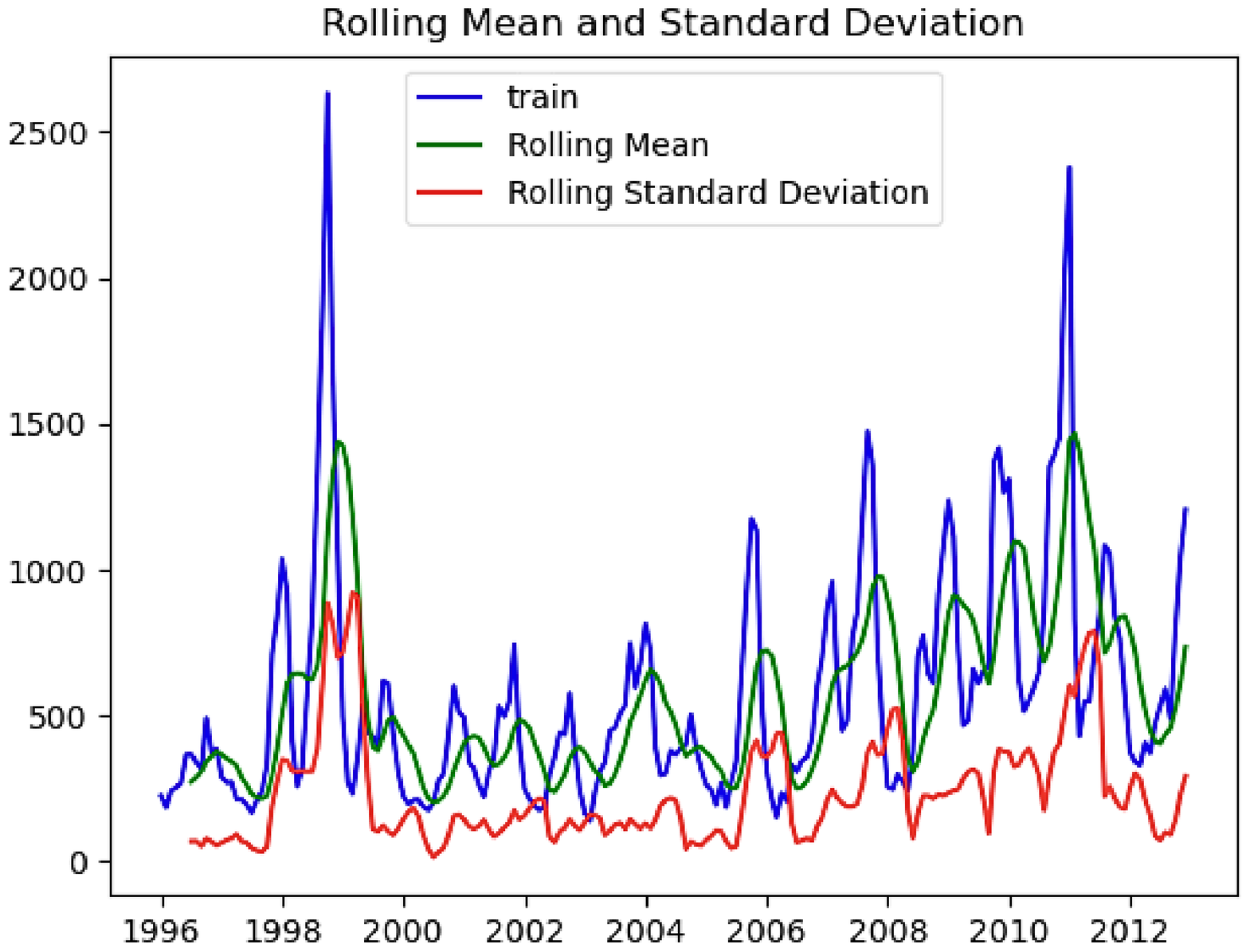


Exploratory Data Analysis (EDA)



Exploratory Data Analysis (EDA)




Rolling Mean and Standard Deviation
train
Rolling Mean
Rolling Standard Deviation
2500
2000
1500
1000
500
0
1996
1998
2000
2002
2004
2006
2008
2010
2012

ARIMA MODEL

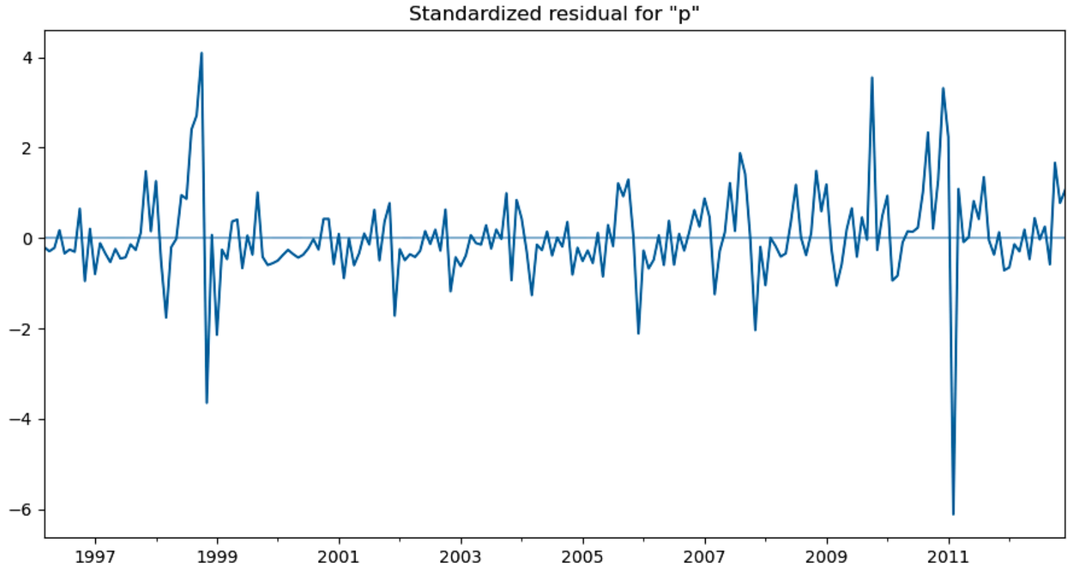
SARIMAX Results

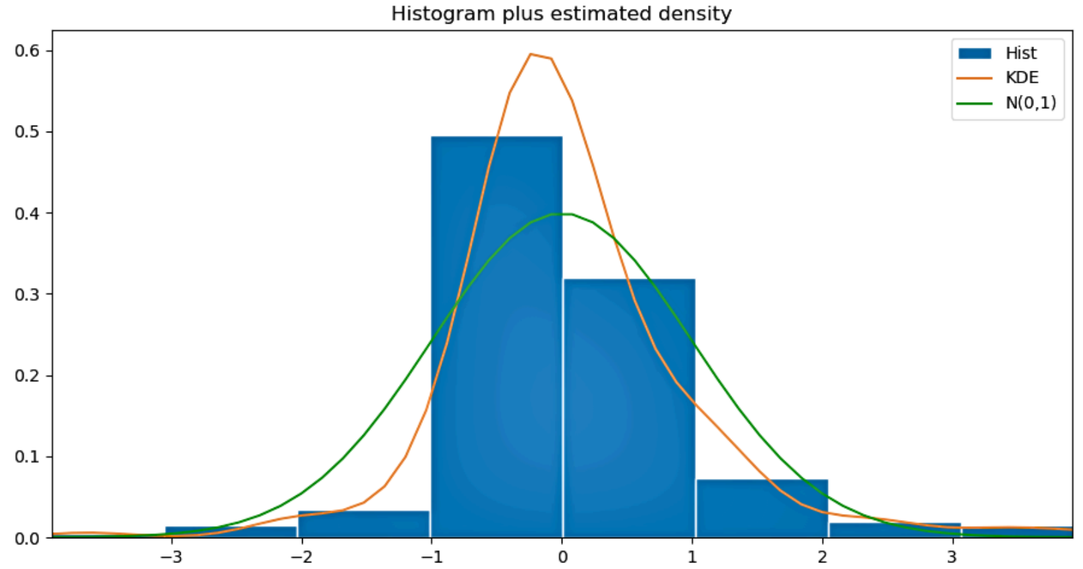
Dep. Variable:	priceMod	No. Observations:	200
Model:	ARIMA(1, 0, 1)	Log Likelihood	-1293.853
Date:	Wed, 29 May 2024	AIC	2595.706
Time:	13:45:18	BIC	2608.859
Sample:	0 - 200	HQIC	2601.030
Covariance Type:	opg		

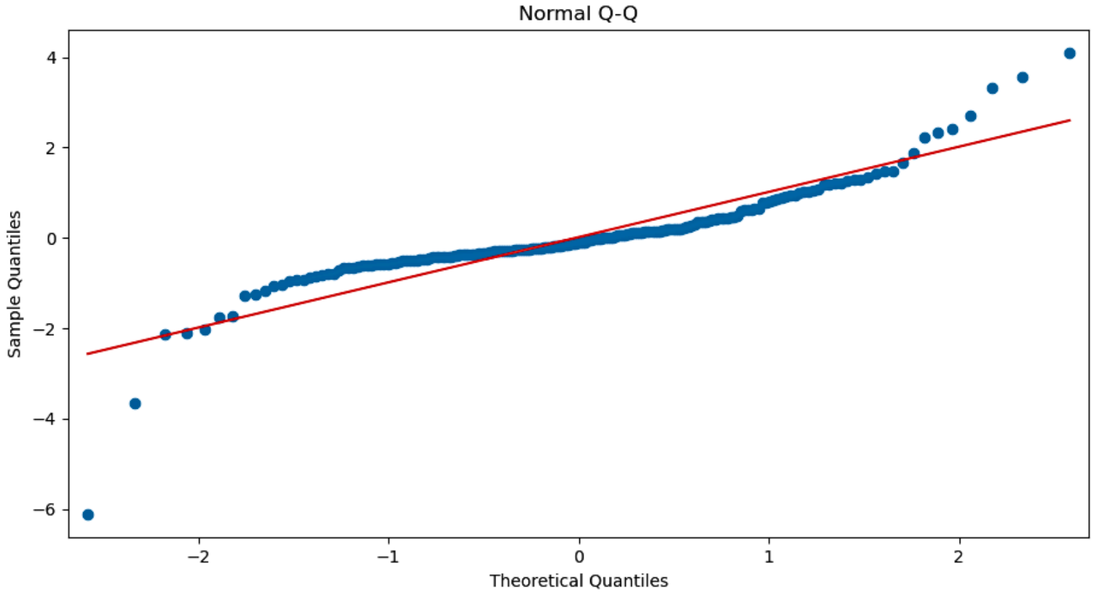
	coef	std err	z	P>|z|	[0.025	0.975]
const	548.5806	91.290	6.009	0.000	369.655	727.507
ar.L1	0.7278	0.053	13.809	0.000	0.625	0.831
ma.L1	0.4567	0.070	6.501	0.000	0.319	0.594
sigma2	2.772e+04	2094.037	13.238	0.000	2.36e+04	3.18e+04

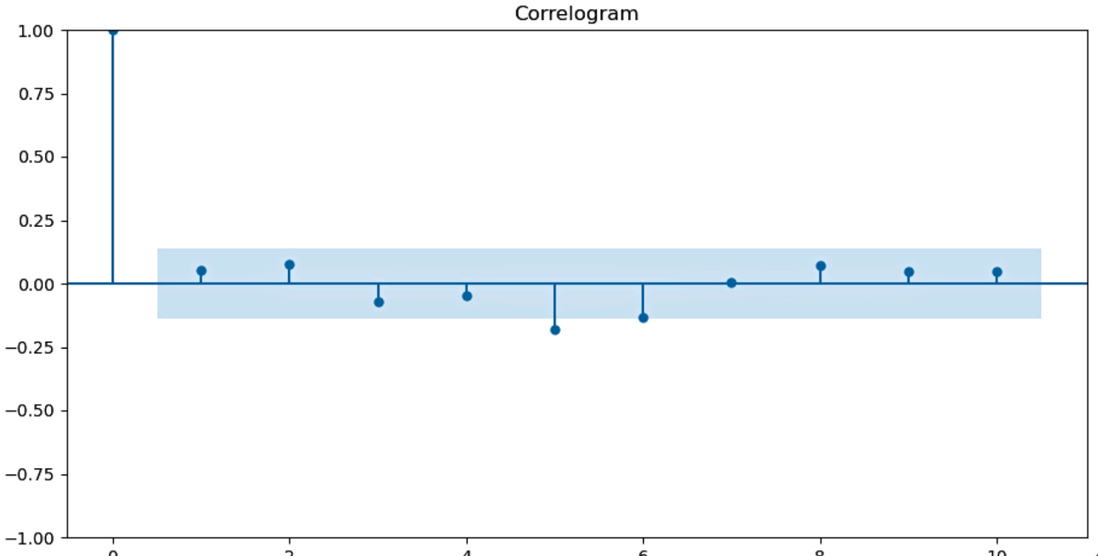
Ljung-Box (L1) (Q):	1.17	Jarque-Bera (JB):	62.04
Prob(Q):	0.28	Prob(JB):	0.00
Heteroskedasticity (H):	2.01	Skew:	0.52
Prob(H) (two-sided):	0.01	Kurtosis:	5.54

ARIMA MODEL

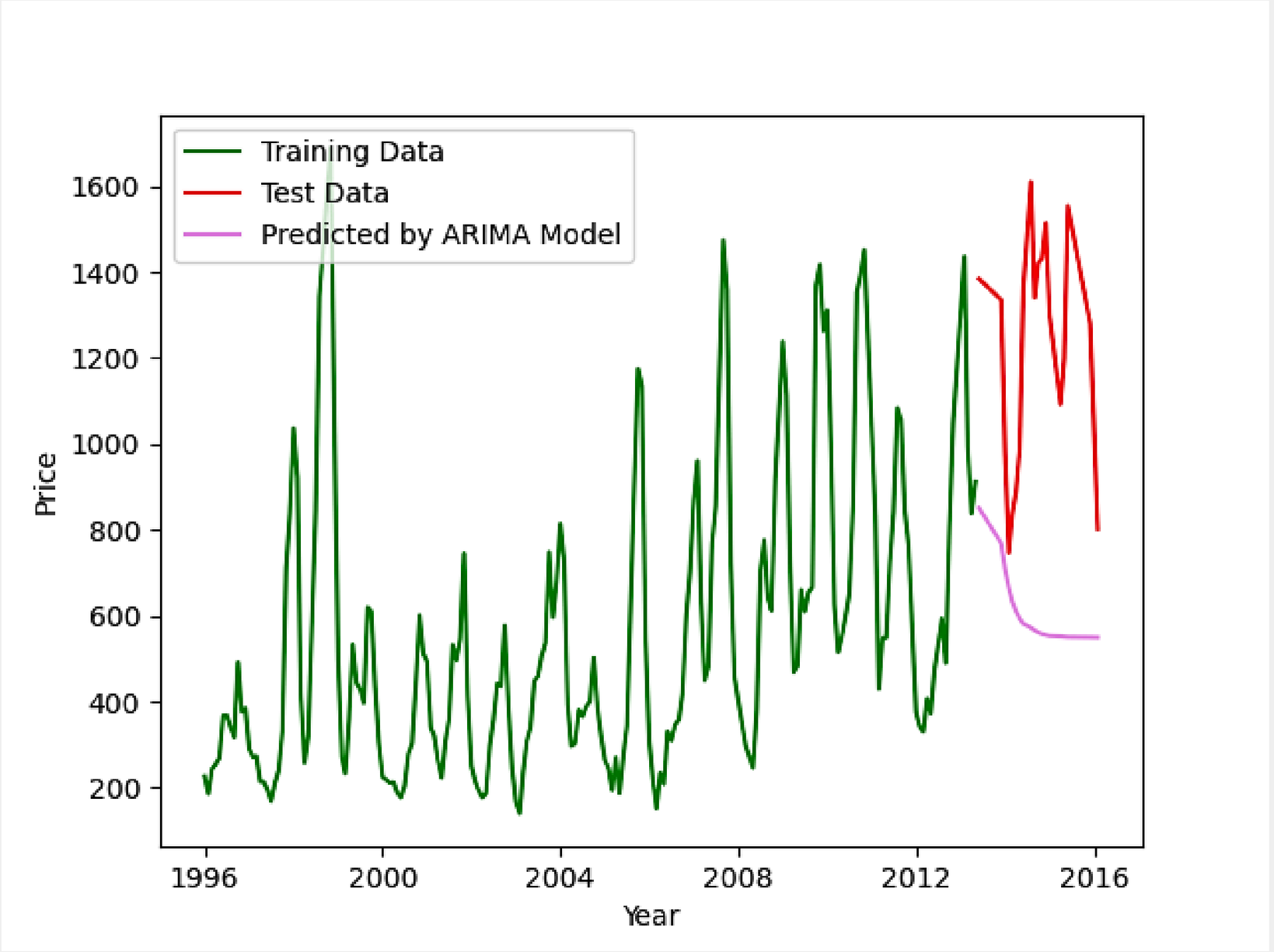

Standardized residual for "p"
4
2
0
−2
−4
−6
1997
1999
2001
2003
2005
2007
2009
2011


Histogram plus estimated density
Hist
KDE
N(0,1)
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−3
−2
−1
0
1
2
3


Normal Q-Q
Sample Quantiles
Theoretical Quantiles
4
2
0
−2
−4
−6
−2
−1
0
1
2


Correlogram
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10

ARIMA MODEL


Training Data
Test Data
Predicted by ARIMA Model
Price
1600
1400
1200
1000
800
600
400
200
1996
2000
2004
2008
2012
2016
Year

SARIMA MODEL

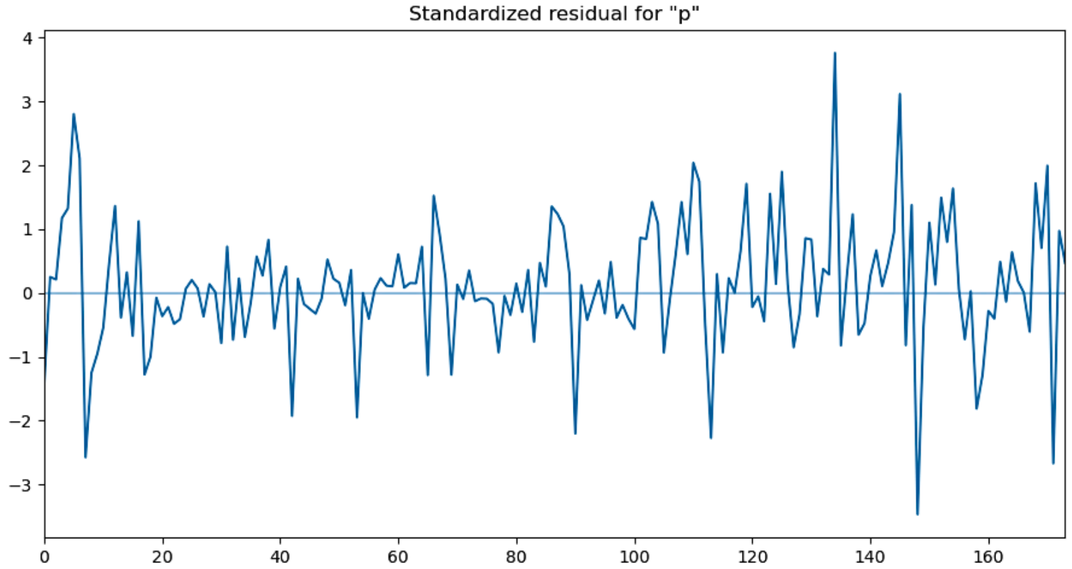
SARIMAX Results

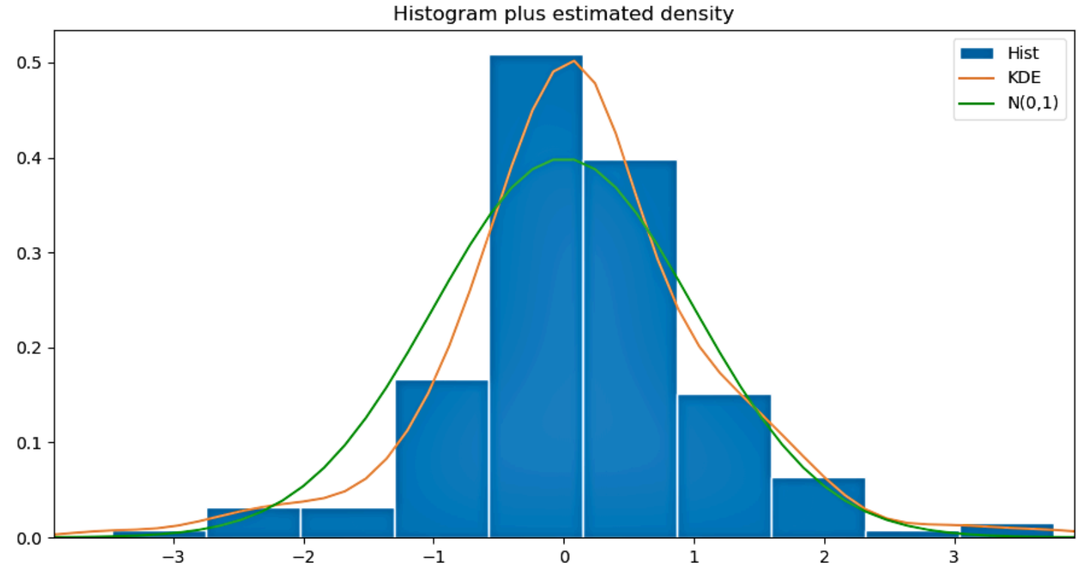
Dep. Variable:	priceMod	No. Observations:	200
Model:	SARIMAX(1, 0, 1)x(0, 1, 1, 12)	Log Likelihood	-1146.728
Date:	Wed, 29 May 2024	AIC	2301.456
Time:	13:47:00	BIC	2314.092
Sample:	0 - 200	HQIC	2306.582
Covariance Type:	opg		

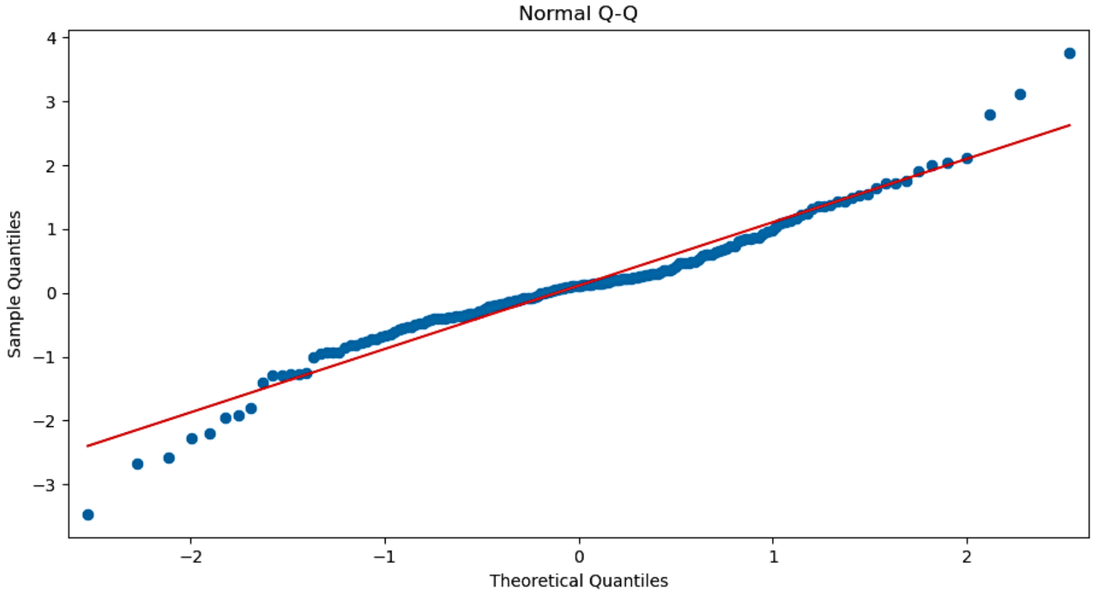
	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.6847	0.047	14.661	0.000	0.593	0.776
ma.L1	0.4924	0.073	6.699	0.000	0.348	0.636
ma.S.L12	-0.8498	0.066	-12.854	0.000	-0.979	-0.720
sigma2	2.952e+04	2282.021	12.935	0.000	2.5e+04	3.4e+04

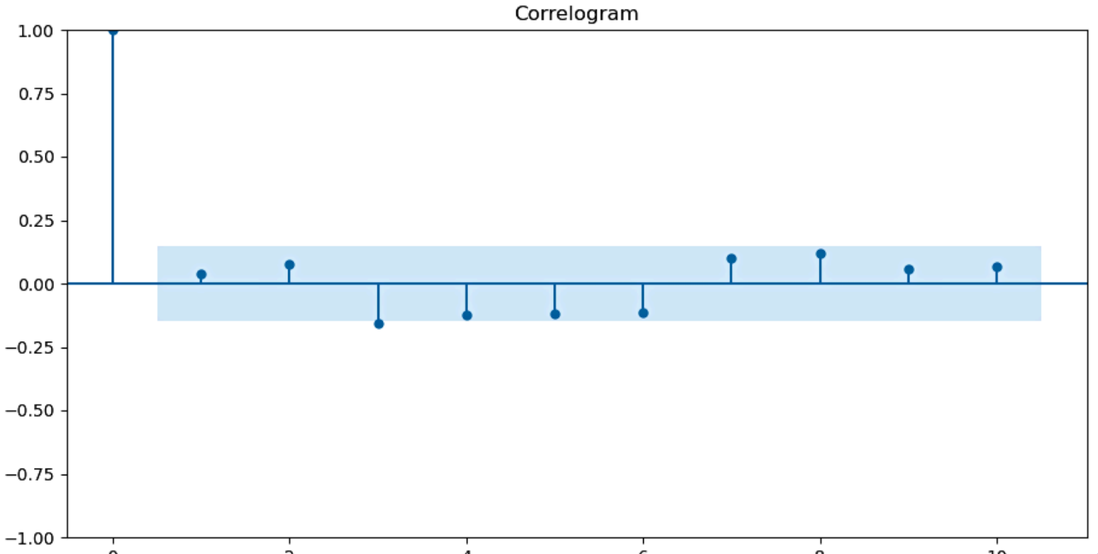
Ljung-Box (L1) (Q):	0.26	Jarque-Bera (JB):	30.21
Prob(Q):	0.61	Prob(JB):	0.00
Heteroskedasticity (H):	1.82	Skew:	0.01
Prob(H) (two-sided):	0.02	Kurtosis:	5.04

SARIMA MODEL

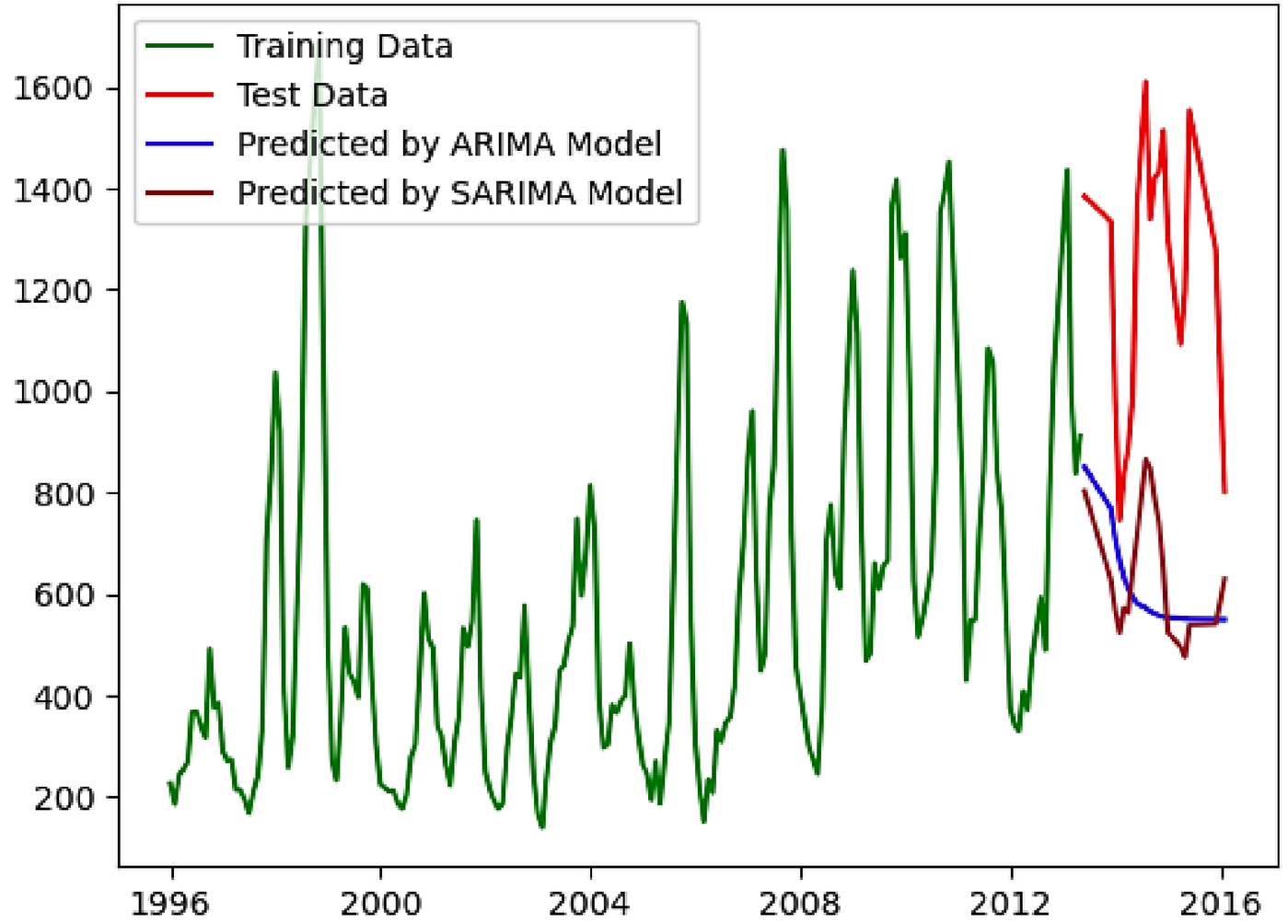




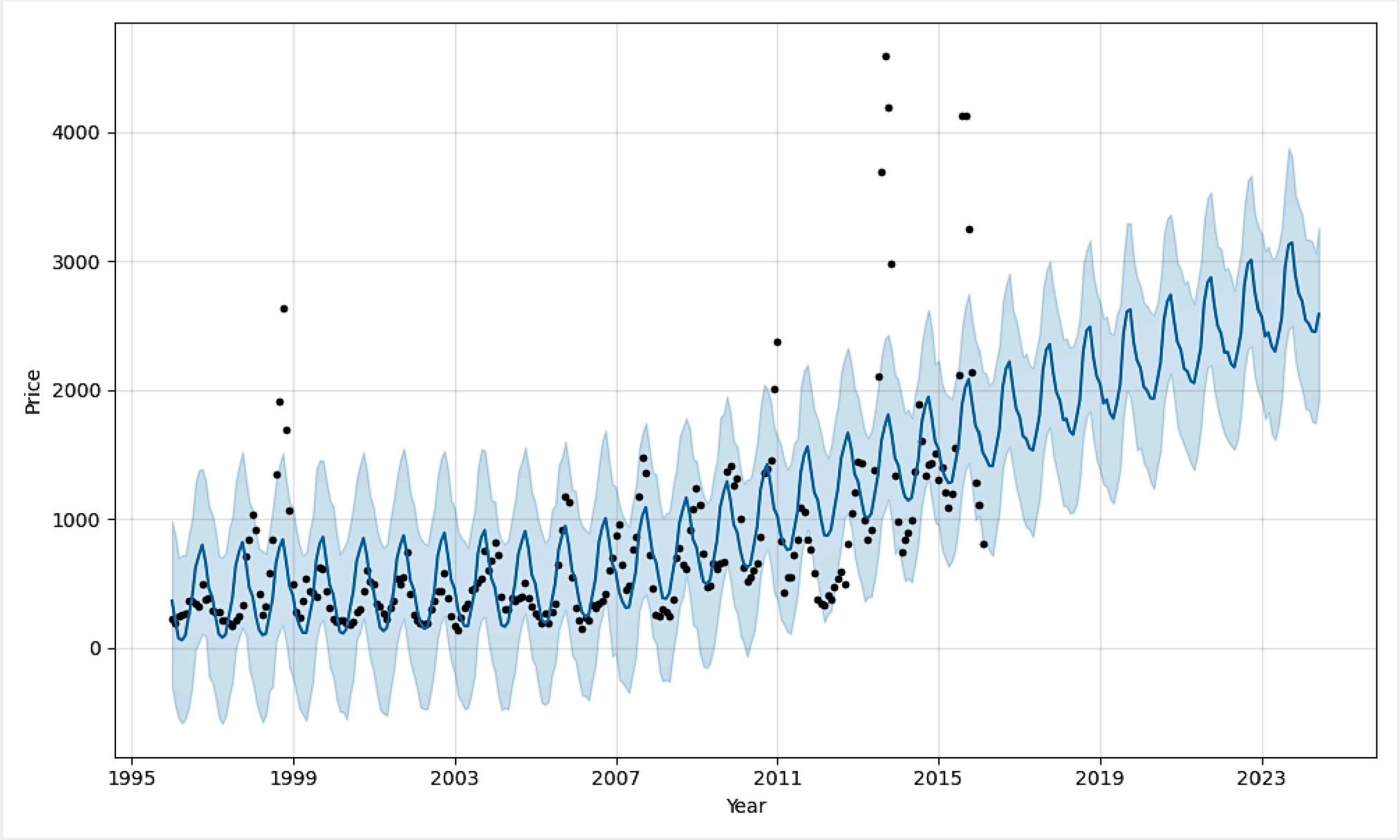




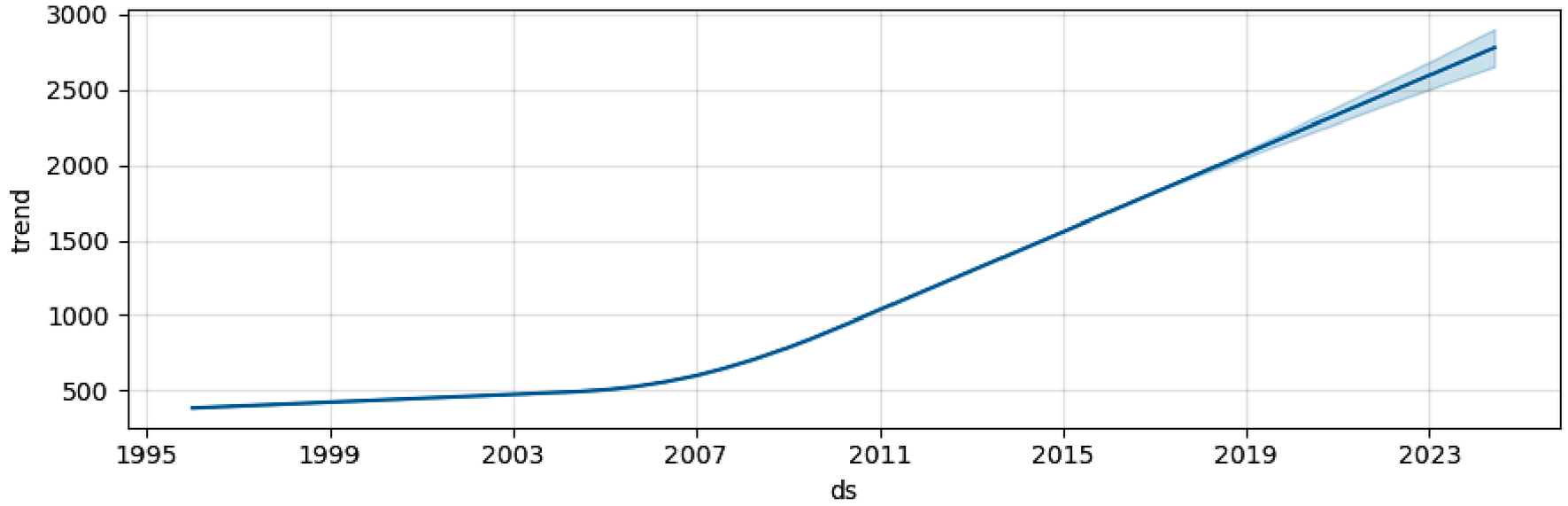
SARIMA MODEL

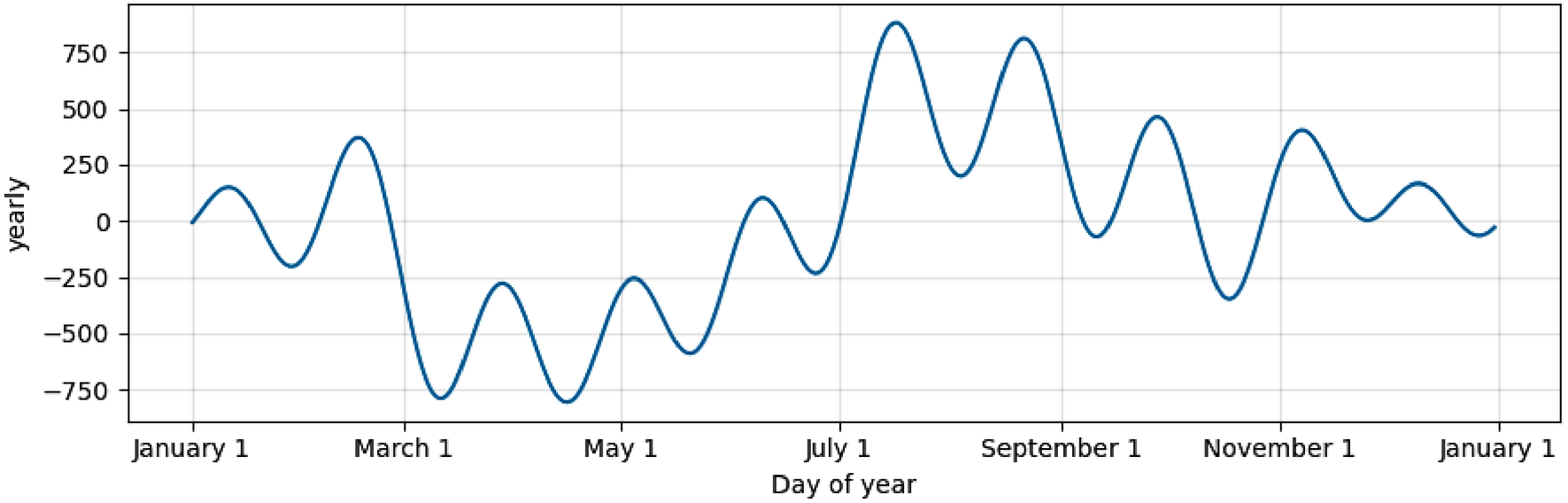


PROPHET MODEL

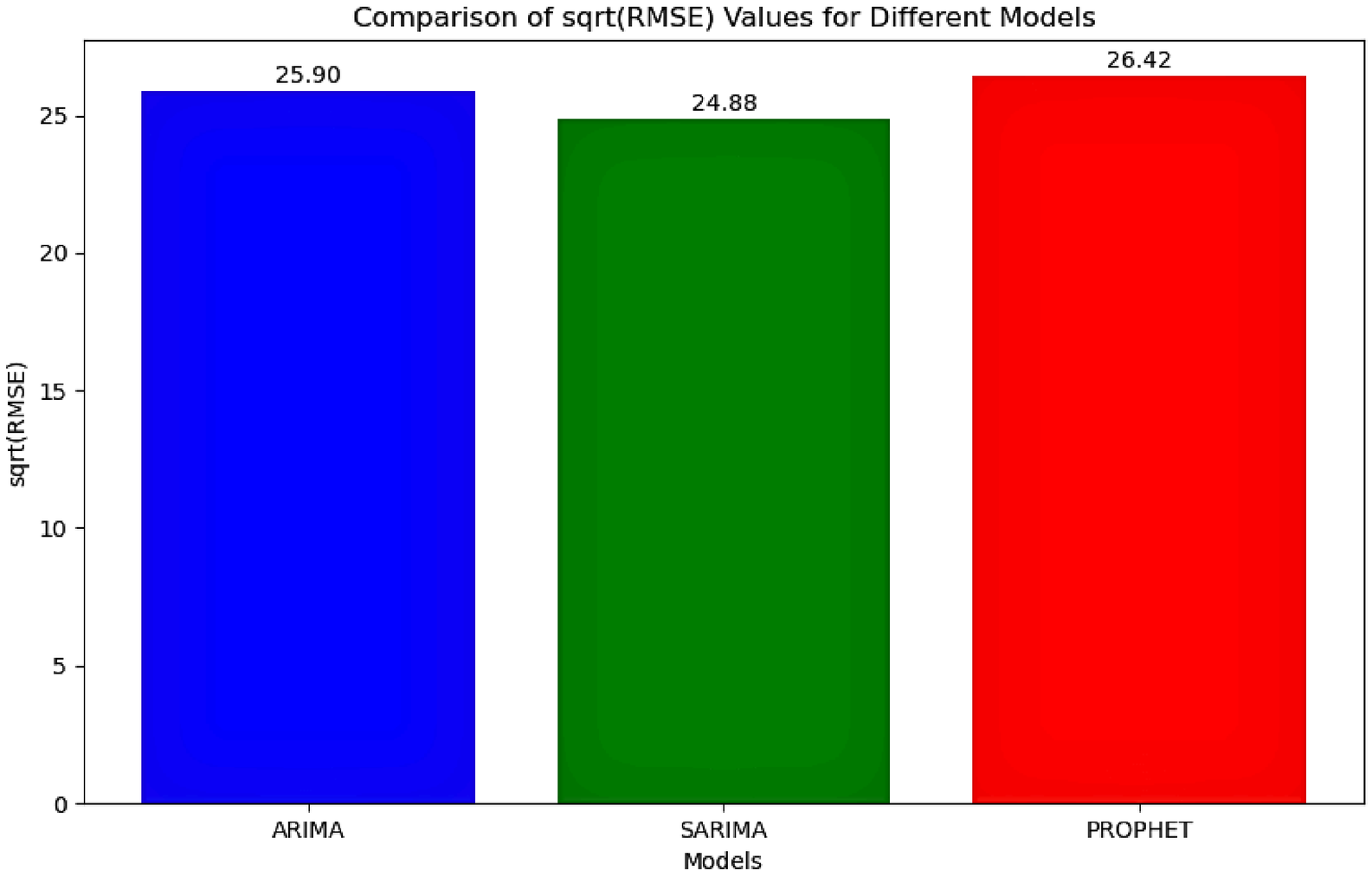


PROPHET MODEL





Comparison of Different Models

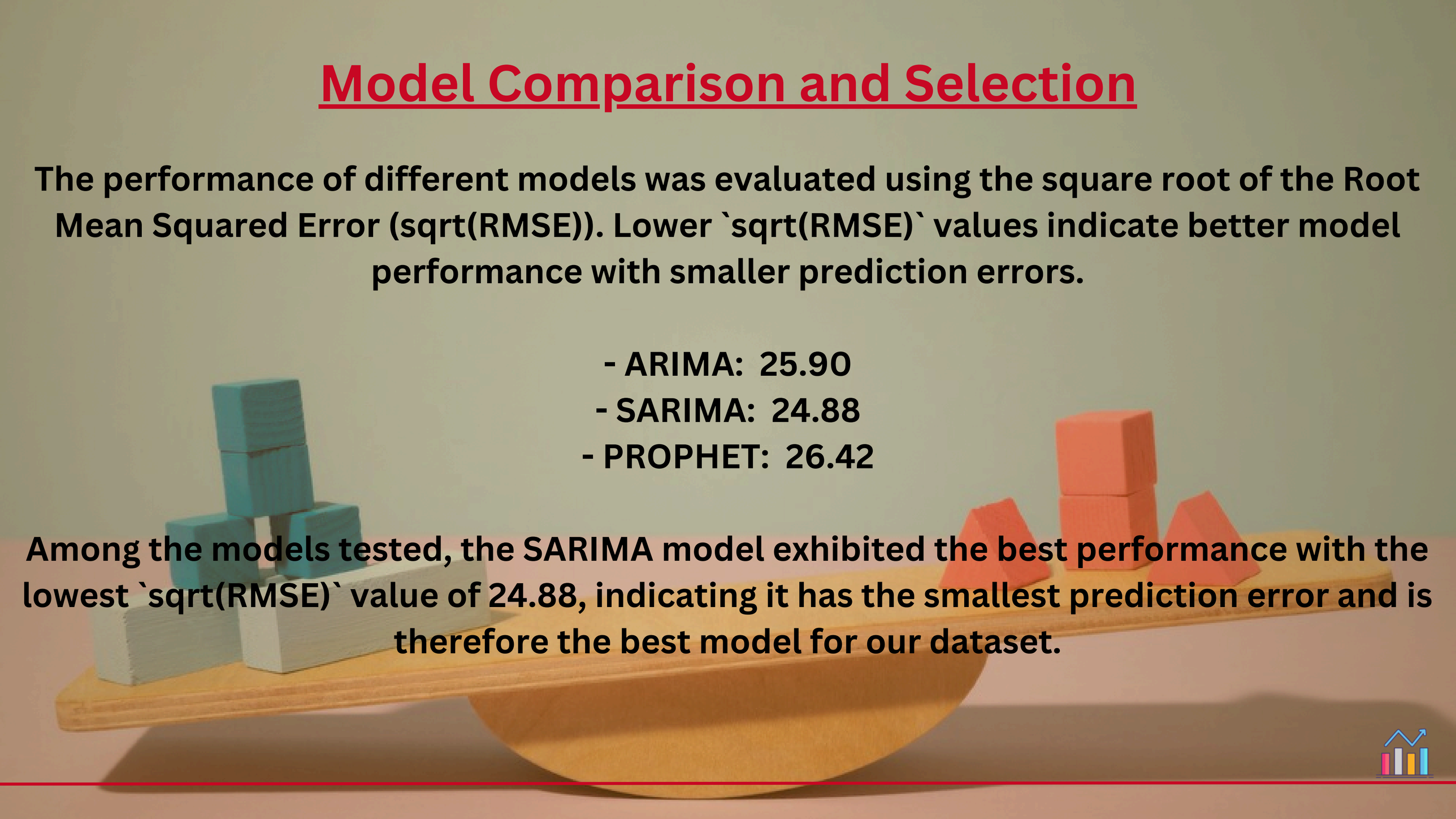


Model Comparison and Selection

The performance of different models was evaluated using the square root of the Root Mean Squared Error (sqrt(RMSE)). Lower `sqrt(RMSE)` values indicate better model performance with smaller prediction errors.

- ARIMA: 25.90
- SARIMA: 24.88
- PROPHET: 26.42

Among the models tested, the SARIMA model exhibited the best performance with the lowest `sqrt(RMSE)` value of 24.88, indicating it has the smallest prediction error and is therefore the best model for our dataset.



Conclusion

- **Models Built:** ARIMA, SARIMA, and PROPHET
- **Performance RMSE Results:**
  - ARIMA: 25.90
  - SARIMA: 24.88
  - PROPHET: 26.42
- **Best Model: SARIMA**
- **Reason:** Lowest sqrt(RMSE) value (24.88), indicating smallest prediction error
- **Outcome:** SARIMA model selected for final implementation to ensure accurate market trend forecasting.

MAHAMMAD BASHEER K S

MENTORNESS