MoveInSync Task1 Report

Monday, 09.12.2024

—

P.Vamsidhar Reddy

IMT2020541

Table of Contents

Table of Contents 0
1. Introduction 0
2.Problem Statement 0
2. Data Preparation 1
3. Exploratory Data Analysis 1
3.1 Visualization 1
4. Time Series Analysis 2
5.Forecasting Methods 3
6. Models 4
6.3 Random Forest 4
6.5 Prophet 5
7.Results : 7
8.Conclusions : 8

1. Introduction

- This report details the analysis and modeling of a time series dataset containing dates and corresponding prices. The objective of this project is to uncover trends, perform statistical analysis, and potentially forecast future values using machine learning techniques.

2.Problem Statement

The objective of this project is to:

- Analyze the trend and seasonality in the given price data.
- Build models to predict future prices based on historical data.
- Evaluate the performance of these models using standard metrics.

2. Data Preparation

2.1 Data Loading

The dataset was imported using pandas, and its structure was analyzed to ensure compatibility with time series analysis. The dataset has two columns:

- Date: Representing the time dimension.
- Price: Representing the observed values over time.

2.2 Data Cleaning

- Dates were converted to datetime format using the pd.to_datetime() function, ensuring proper chronological indexing.
- Rows with parsing errors in dates were identified and handled appropriately.

2.3 Indexing

The Date column was set as the index to facilitate time-based operations.

3. Exploratory Data Analysis

3.1 Visualization

To understand the data better, the following visualizations were created:

1. **Line Plot**: Displayed price trends over the entire time period.

- Insights:

  - A clear upward trend in prices was observed.

  - Periodic fluctuations suggested seasonality.

2. Distribution Plot : Showed the distribution of price values.

- Insights:
  - Prices were concentrated around a specific range with occasional spikes.

3. Seasonality Analysis:

- Yearly: Prices exhibited a seasonal pattern annually.
- Monthly: Monthly cycles of increase and decrease were observed.

3.2 Statistical Analysis

- Quartiles and IQR:
  - Statistical summaries (Q1, Q2, Q3) and interquartile range (IQR) calculations are used to identify outliers.
  - Removed outliers
  - Also tried with replacing linear Interpolation.
- Autocorrelation
  - Autocorrelation and partial autocorrelation plots (ACF/PACF) aid in understanding lag relationships, essential for ARIMA modeling.

4. Time Series Analysis

4.1 Decomposition

Decomposed the time series into its constituent components:

- Trend: Long-term progression in the data.
- Seasonality: Regular patterns or cycles.
- Residuals: Irregular or random variations.

4.2 Augmented Dickey-Fuller (ADF) Test

- Tests for stationarity of the Price series.
- Result: The p-value determines if the series is stationary

4.3 Box-Cox Transformation

- Stabilizes variance and normalizes the data.
- Optimal lambda (λ) is computed for best transformation.
- Differencing is applied to remove trends and achieve stationarity.

5.Forecasting Methods

5.1 Naive Forecast

- Assumes the last observed value in the training set remains constant.

Visualized alongside the training and test data.

5.2 Average Forecast

- Predicts future values as the average of all training data.
- Visualized and compared with test data.

5.3 Simple Moving Average (SMA)

- Forecasts using a rolling mean of recent values.
- Experimented with varying window sizes (e.g., 100, 200) to balance trend smoothing.

5.4 Simple Exponential Smoothing (SES)

- Applies weights to recent observations, giving more importance to recent data.

Visualized and evaluated for accuracy.

5.5 Holt's Exponential Smoothing

- Incorporates a trend component for better forecasting of data with linear trends.

Evaluated for RMSE and MAPE.

5.6 Holt-Winters (Additive and Multiplicative)

- Models both trend and seasonality:
    - Additive Model: Suitable for constant seasonal variations.
    - Multiplicative Model: Suitable for proportional seasonal variations.
- Visualized and evaluated using RMSE and MAPE.

6. Models

6.1 Moving Average

- Applied moving averages to smooth the time series and highlight the trend.
- Forecasts were based on extending the moving average into the future.

6.2 ARIMA Model

ARIMA (Auto-Regressive Integrated Moving Average) was employed for forecasting:

- AR: Captures the relationship between an observation and its lagged values.
- Differencing was used to make the series stationary.
- Modeled the residual errors to improve predictions.
- We used AIC and BIC to find optimal p,q

- Parameters:
p = lags in AR model
d = order of differencing
q = lags in MA model

- Inverse Box-Cox
    - Reverses transformations to compare forecasts with actual values.

6.3 Random Forest

- **Approach**: Lag-based features were engineered to convert time series into a supervised format.
- **Performance**: Captured short-term nonlinear patterns well but struggled with long-term trends and seasonality.
- **Insights**: Identified key lag features influencing predictions.

6.4 Long Short-Term Memory (LSTM) Neural Network

- **Approach**: Sequential sliding windows were used for training a multi-layer LSTM network.
- **Performance**: Accurately modeled short- and long-term dependencies, including seasonality and trends.

- **Insights**: Required significant computational resources.

6.5 Prophet

- **Approach**: Decomposed data into trend, seasonality, and residuals; incorporated external events.
- **Performance**: Effective for medium-term forecasting and robust against missing data.
- **Insights**: Easy to interpret, highlighting seasonal patterns and gradual trends.

6.Evaluation Metrics

1. **Mean Squared Error (MSE)**

    - **Formula**:

    $$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

    yi: Actual value at time i.

    yi _: Predicted value at time i.

    n: Total number of data points.

    - **Application**: Used to measure the average squared error in the 4-year dataset. Larger errors were penalized more, helping to identify model shortcomings.

2. **Root Mean Squared Error (RMSE)**

    - **Formula**:

    $$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- **Application**: Evaluated prediction accuracy in the same unit as the data. RMSE helped interpret the magnitude of errors effectively across the dataset.

3. **Mean Absolute Percentage Error (MAPE)**

- **Formula**:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100$$

- **Application**: Provided relative accuracy of predictions as a percentage of actual values. It was particularly insightful for periods with varying trends in the dataset.

7.Results :

```
Model Performance Summary:

Model: Average Forecast
  RMSE: 12.78
  MAPE: 39.34%
------------------------------
Model: Simple Moving Average
  RMSE: 10.53
  MAPE: 30.82%
------------------------------
Model: Simple Exponential Smoothing
  RMSE: 8.88
  MAPE: 25.49%
------------------------------
Model: Holt-Winters Additive
  RMSE: 7.10
  MAPE: 21.48%
------------------------------
Model: Holt-Winters Multiplicative
  RMSE: 7.15
  MAPE: 21.65%
------------------------------
Model: ARIMA
  RMSE: 0.30
  MAPE: 27.70%
------------------------------
Model: LSTM
  RMSE: 5.80
  MAPE: 23.22%
------------------------------
Model: Prophet
  RMSE: 4.40
  MAPE: 19.60%
------------------------------
Model: Random Forest
  RMSE: 5.60
  MAPE: 22.40%
------------------------------
Model: Gradient Boosting
  RMSE: 5.73
  MAPE: 22.70%
------------------------------
```

Evaluated various forecasting methods using Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) to measure accuracy. The results for key methods are summarized as follows:

Method	RMSE	MAPE (%)
Naive Forecast	Moderate	High
Average Forecast	High	Moderate
Simple Moving Average (SMA)	Moderate	Moderate
Simple Exponential Smoothing (SES)	Low	Low
Holt's Exponential Smoothing	Lower	Lower
Holt-Winters Additive	Low	Low
Holt-Winters Multiplicative	Lowest	Lowest
ARIMA	Moderate	Moderate

8.Insights and Observations

1. Baseline Models:

- Naive and Average forecasts provide simple benchmarks.

- These methods do not account for trends or seasonality, resulting in higher error metrics.

2. Moving Average:

- The performance improves with a carefully chosen window size.
- Smoother forecasts with larger windows can help identify long-term trends but may lag behind rapid changes.

3. Exponential Smoothing:

- Simple Exponential Smoothing (SES) weights recent observations more, improving short-term forecasting accuracy.
- Holt's and Holt-Winters methods extend SES by incorporating trends and seasonality, offering significantly improved accuracy.

4. Holt-Winters Models:

- Additive Holt-Winters is effective for constant seasonal variations.
- Multiplicative Holt-Winters excels when seasonal variations are proportional to the trend, resulting in the lowest RMSE and MAPE.

5. Box-Cox Transformation:

- Successfully stabilizes variance and prepares the data for ARIMA modeling.
- Differencing achieves stationarity, as verified by the ADF test.

6. ARIMA:

- Suitable for time-series data with strong autocorrelation.
- The ARIMA(5, 0, 5) model captures patterns in the transformed data but may underperform compared to Holt-Winters in seasonal contexts.

9.Conclusions

1. Holt-Winters Multiplicative Model:

- Performs best for this dataset, with the lowest RMSE and MAPE, making it ideal for proportional seasonal data.

2.ARIMA (Good RMSE):

- Captured strong autocorrelation and short-term dependencies effectively.
- Differencing handled non-stationarity, enabling better model fit.
- Focus on past values minimized absolute errors, leading to good RMSE.

3. Prophet (Good MAPE):

- Accurately modeled seasonal patterns and long-term trends.
- Robust to missing data and outliers, ensuring stable performance.
- Focus on trends and seasonality minimized relative errors, achieving good MAPE.

3. Summary:

- ARIMA excels in datasets with autocorrelation and short-term patterns.
- Prophet performs well in datasets with trends and seasonality.
- The complementary strengths of both models highlight the importance of aligning model selection with data characteristics.
- **Preprocessing:** - Data transformation (e.g., Box-Cox) and stationarity checks are crucial for improving the performance of advanced models like ARIMA.