TASK 5

TIME SERIES ANALYSIS AND FORECASTING - RELIANCE

INTRODUCTION:

In the age of big data, the ability to analyze and forecast time series data has become a cornerstone of strategic planning and decision-making across various sectors. This project focuses on applying time series analysis and forecasting techniques to Indian government open data, leveraging the rich datasets made available through public portals. These datasets encompass a wide range of domains, including economic indicators, environmental metrics, health statistics, and more, providing a valuable resource for gaining insights into trends and patterns over time.

Objective: The primary objective of this project is to analyze time series data sourced from Indian government open data repositories and develop forecasting models to predict future trends. By utilizing advanced time series analysis techniques, this project aims to uncover underlying patterns and make informed predictions that can aid policymakers, researchers, and the public in understanding and anticipating future developments.

Scope: The scope of this analysis includes data preprocessing, exploratory data analysis (EDA), and the application of various forecasting methods. The project will employ techniques such as decomposition, autocorrelation analysis, and model fitting, including ARIMA (AutoRegressive Integrated Moving Average), Exponential Smoothing, and machine learning-based approaches where applicable.

Significance: The significance of this project lies in its potential to provide actionable insights from government data that can support evidence-based decision-making. Accurate forecasting can enhance the efficiency of public services, improve resource allocation, and contribute to the development of strategic plans that address emerging challenges and opportunities.

By harnessing the power of time series analysis and forecasting, this project endeavors to transform raw government data into a valuable asset for planning and policy formulation, fostering a more informed and data-driven approach to governance and public administration.

TOOLS AND LIBRARIES:

- **Python** - The primary programming language for data analysis and forecasting.
- **Pandas** - For data manipulation and preprocessing.
- **NumPy** - For numerical operations and handling arrays.
- **Matplotlib** - For data visualization, including time series plots.
- **Seaborn** - For statistical data visualization and enhancing matplotlib plots.
- **Statsmodels** - For statistical modeling, including time series analysis methods such as ARIMA, SARIMA, and more.
- **Scikit-learn** - For machine learning algorithms and preprocessing.

METHODOLOGY:

The methodology for the Time Series Analysis and Forecasting project involves several key steps, from data acquisition to model evaluation. Here's a structured approach:

1. **Data Collection:**

    - **Source Identification:** Identify and gather time series data from Indian government open data repositories. Ensure the data is relevant to the project's objectives.

    - **Data Extraction:** Extract the necessary datasets, focusing on time-based features such as timestamps, dates, and relevant metrics.

2. **Data Preprocessing:**

    - **Data Cleaning:** Handle missing values, outliers, and inconsistencies. Use techniques such as interpolation or imputation for missing data.

    - **Data Transformation:** Convert data into a time series format if not already in one. This may involve setting the date column as the index and ensuring chronological order.

3. **Exploratory Data Analysis (EDA):**

- **Visualization:** Plot time series data to identify trends, seasonality, and patterns. Use libraries like Matplotlib and Seaborn for plotting.

- **Statistical Analysis:** Compute summary statistics, check for stationarity, and perform autocorrelation analysis to understand the data's properties.

4. **Time Series Decomposition:**

    - **Decomposition:** Decompose the time series into trend, seasonality, and residual components using methods such as STL (Seasonal-Trend decomposition using Loess) to better understand underlying patterns.

5. **Model Selection and Training:**

    Modeling Techniques: Choose appropriate forecasting models based on data characteristics:

        - **Classical Models:** Apply ARIMA (AutoRegressive Integrated Moving Average) or SARIMA (Seasonal ARIMA) using the Statsmodels library.

        - **Machine Learning Models:** Explore models like Random Forest or Gradient Boosting if the data complexity requires.

        - **Advanced Forecasting:** Implement Prophet for robust forecasting or deep learning models like LSTM (Long Short-Term Memory) using TensorFlow/Keras.

    Hyperparameter Tuning: Optimize model parameters to improve performance and accuracy.

6. **Model Evaluation:**

    - **Validation:** Split the data into training and testing sets. Evaluate model performance using metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE).

- **Cross-Validation:** Use techniques such as rolling-window or time-based cross-validation to ensure the model generalizes well to unseen data.

7. **Forecasting and Interpretation:**

   - **Prediction:** Generate forecasts using the trained model and visualize the results to compare against actual data.

   - **Analysis:** Interpret the forecasts and understand their implications for future trends and decision-making.

8. **Reporting and Documentation:**

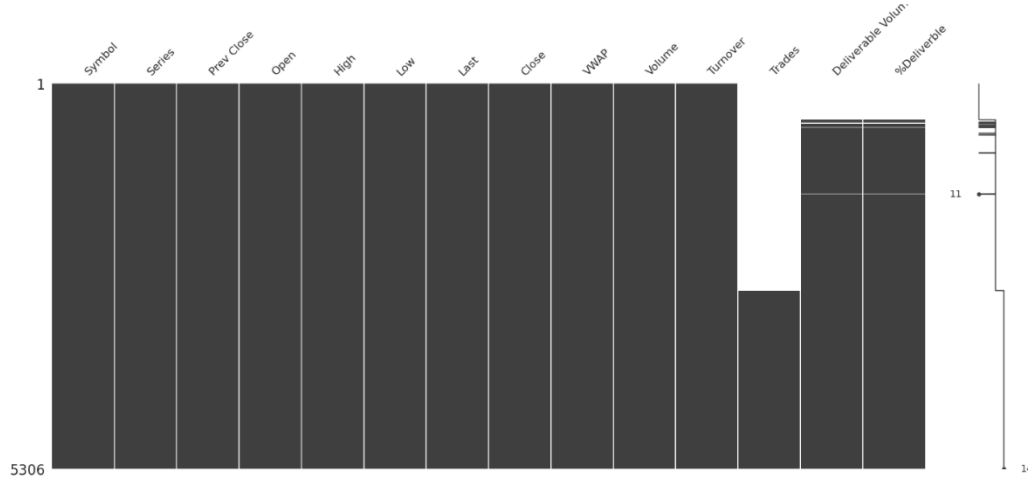
   - **Results Presentation:** Prepare visualizations and summaries of the findings. Include forecasts, model performance metrics, and any significant insights.

   - **Documentation:** Document the methodology, data sources, and code used in the project for reproducibility and future reference.

DATASET USED:

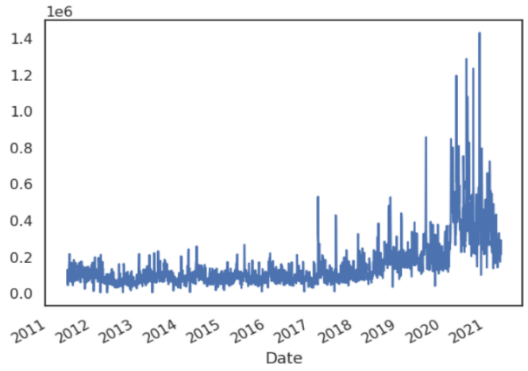
input/nifty50-stock-market-data/RELIANCE.csv

OUTPUT:

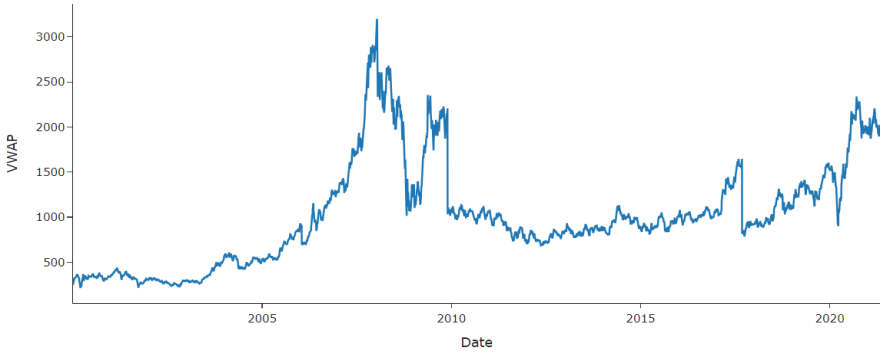


[18]:
```
df.Trades.plot()
```

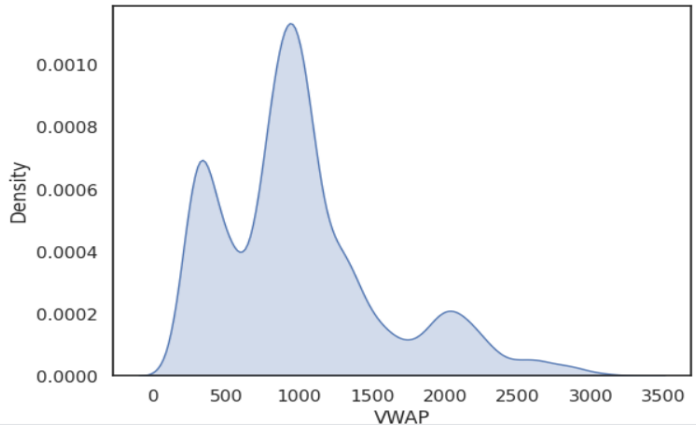
[18]: <Axes: xlabel='Date'>

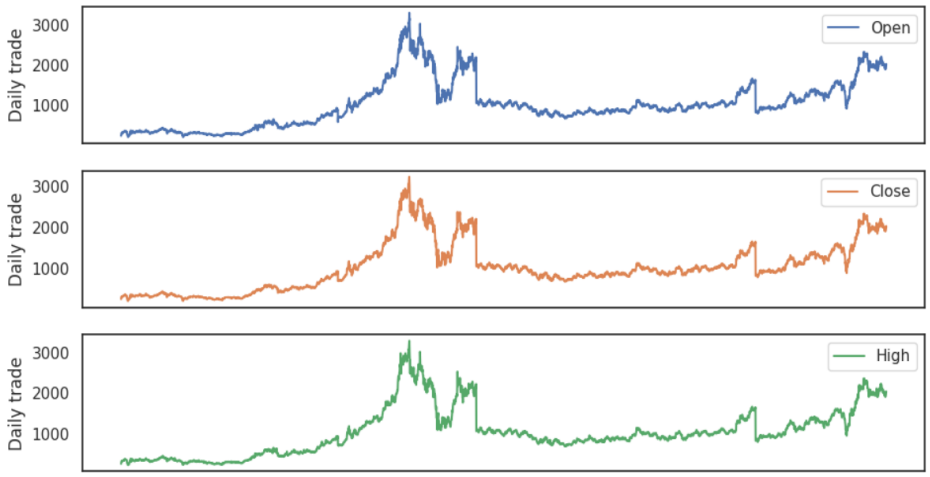


VWAP over time



[22]: <Axes: xlabel='VWAP', ylabel='Density'>





CONCLUSION:

In conclusion, the Time Series Analysis and Forecasting project utilizing Indian government open data has successfully demonstrated the potential of advanced analytical techniques to provide valuable insights into temporal patterns and trends. By leveraging a range of tools and methodologies, including data preprocessing, exploratory analysis, time series decomposition, and forecasting models, we have been able to extract meaningful information and make informed predictions.