

```

1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 from statsmodels.tsa.arima.model import ARIMA
6 from sklearn.metrics import mean_squared_error
7 import warnings

1 df_bax_m = pd.read_csv(r'/content/drive/MyDrive/PRN23039142546/df_bax_cleaned_to_view_outliers.csv', index_col=0, parse_dates=True)
2 df_bax_m.head()

```



	Price	Open	High	Low	Vol.	Change %
Date						
2010-05-24	1482.42	1491.98	1491.98	1482.42	926980.0	-0.64
2010-05-25	1454.85	1482.42	1482.42	1454.85	1660000.0	-1.86
2010-05-26	1472.29	1456.50	1472.29	1454.85	1500000.0	1.20
2010-05-27	1453.82	1472.29	1478.07	1453.82	2480000.0	-1.25
2010-05-30	1455.16	1453.82	1462.04	1453.72	5910000.0	0.09

```

1 df = df_bax_m.copy()
2 series = df['Price']

1 # Train-test split
2 train_size = int(len(series) * 0.8)
3 train, test = series[:train_size], series[train_size:]

1 # Fit ARIMA model
2 # We'll start with ARIMA(1,1,1) - adjust later
3 model = ARIMA(train, order=(5, 1, 5)) # p=1, d=1, q=1
4 model_fit = model.fit()

```



```

/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but it
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but it
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but it
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/base/model.py:607: ConvergenceWarning: Maximum Likelihood optimization failed to
warnings.warn("Maximum Likelihood optimization failed to ")

```

```

1 # Forecast
2 preds = model_fit.forecast(steps=len(test))

```



```

/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: ValueWarning: No supported index is available. Predic
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: FutureWarning: No supported index is available. In t
return get_prediction_index(

```

```

1 # Evaluate
2 rmse = np.sqrt(mean_squared_error(test, preds))
3 print(f"ARIMA RMSE: {rmse:.2f}")

```



```
ARIMA RMSE: 75.57
```

```

1 # Plot
2 plt.figure(figsize=(10, 4))
3 plt.plot(train.index, train, label='Train')
4 plt.plot(test.index, test, label='Actual')
5 plt.plot(test.index, preds, label='Forecast')
6 plt.title('ARIMA Forecast')
7 plt.legend()
8 plt.grid(True)
9 plt.show()

```



Grid Search for best p,d,q

```
1 # Grid Search over p, d, q
2 import itertools
3 p = d = q = range(0, 3) # You can expand to range(0, 5) later
4 pdq_combinations = list(itertools.product(p, d, q))

1 best_rmse = float("inf")
2 best_order = None

1 print("Trying combinations of ARIMA(p,d,q):\n")
2
3 for order in pdq_combinations:
4     try:
5         model = ARIMA(train, order=order)
6         model_fit = model.fit()
7         preds = model_fit.forecast(steps=len(test))
8         rmse = np.sqrt(mean_squared_error(test, preds))
9         print(f"ARIMA{order} RMSE: {rmse:.2f}")
10
11         if rmse < best_rmse:
12             best_rmse = rmse
13             best_order = order
14
15     except:
16         continue
17
18 print(f"\n✅ Best ARIMA Order: {best_order} with RMSE: {best_rmse:.2f}")
```



Trying combinations of ARIMA(p,d,q):

```
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: ValueWarning: No supported index is available. Pred
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: FutureWarning: No supported index is available. In
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/sarimax.py:978: UserWarning: Non-invertible starting MA paramet
warn('Non-invertible starting MA parameters found.')
ARIMA(0, 0, 0) RMSE: 614.65
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: ValueWarning: No supported index is available. Pred
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: FutureWarning: No supported index is available. In
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/sarimax.py:978: UserWarning: Non-invertible starting MA paramet
```

```

warn('Non-invertible starting MA parameters found.')
ARIMA(0, 0, 1) RMSE: 614.19
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: ValueWarning: No supported index is available. Pred
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: FutureWarning: No supported index is available. In
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: ValueWarning: No supported index is available. Pred
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: FutureWarning: No supported index is available. In
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but i
self._init_dates(dates, freq)
ARIMA(0, 0, 2) RMSE: 614.05
ARIMA(0, 1, 0) RMSE: 64.48
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: ValueWarning: No supported index is available. Pred
return get_prediction_index(

1 # Refit the best model and plot
2 model = ARIMA(train, order=best_order)
3 model_fit = model.fit()
4 preds = model_fit.forecast(steps=len(test))

/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but it
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but it
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:473: ValueWarning: A date index has been provided, but it
self._init_dates(dates, freq)
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: ValueWarning: No supported index is available. Predic
return get_prediction_index(
/usr/local/lib/python3.11/dist-packages/statsmodels/tsa/base/tsa_model.py:837: FutureWarning: No supported index is available. In th
return get_prediction_index(

1 # Plotting
2 plt.figure(figsize=(10, 4))
3 plt.plot(train.index, train, label='Train')
4 plt.plot(test.index, test, label='Actual')
5 plt.plot(test.index, preds, label='Forecast')
6 plt.title(f"Best ARIMA{best_order} Forecast")
7 plt.legend()
8 plt.grid(True)
9 plt.show()

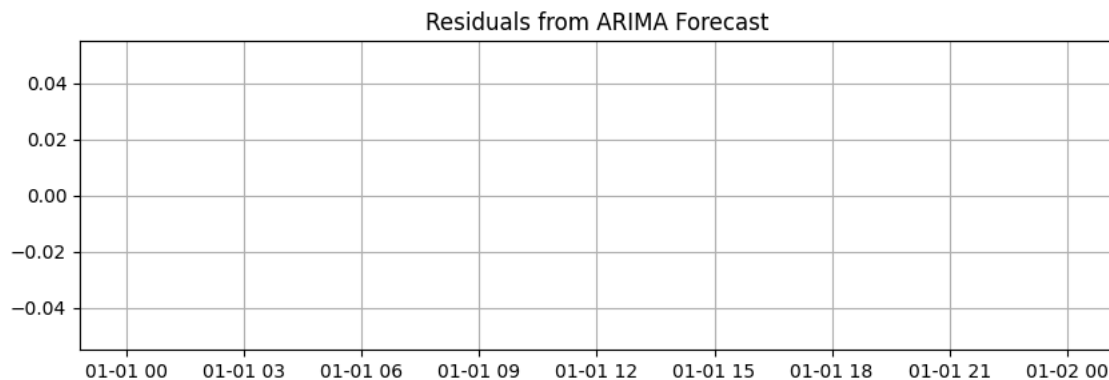
```



```

1 residuals = test - preds
2 plt.figure(figsize=(10, 3))
3 plt.plot(residuals)
4 plt.title("Residuals from ARIMA Forecast")
5 plt.grid(True)
6 plt.show()
7

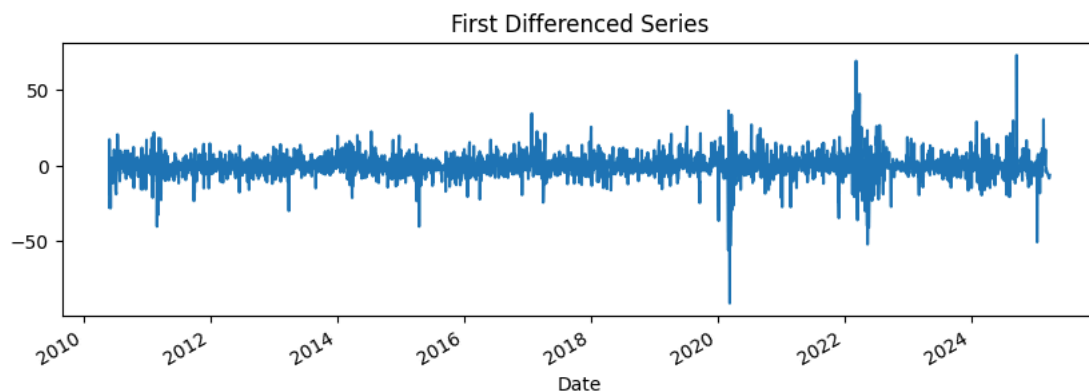
```



```
1 diff_series = series.diff().dropna()
2 diff_series.plot(figsize=(10, 3), title="First Differenced Series")
3
```



<Axes: title={'center': 'First Differenced Series'}, xlabel='Date'>



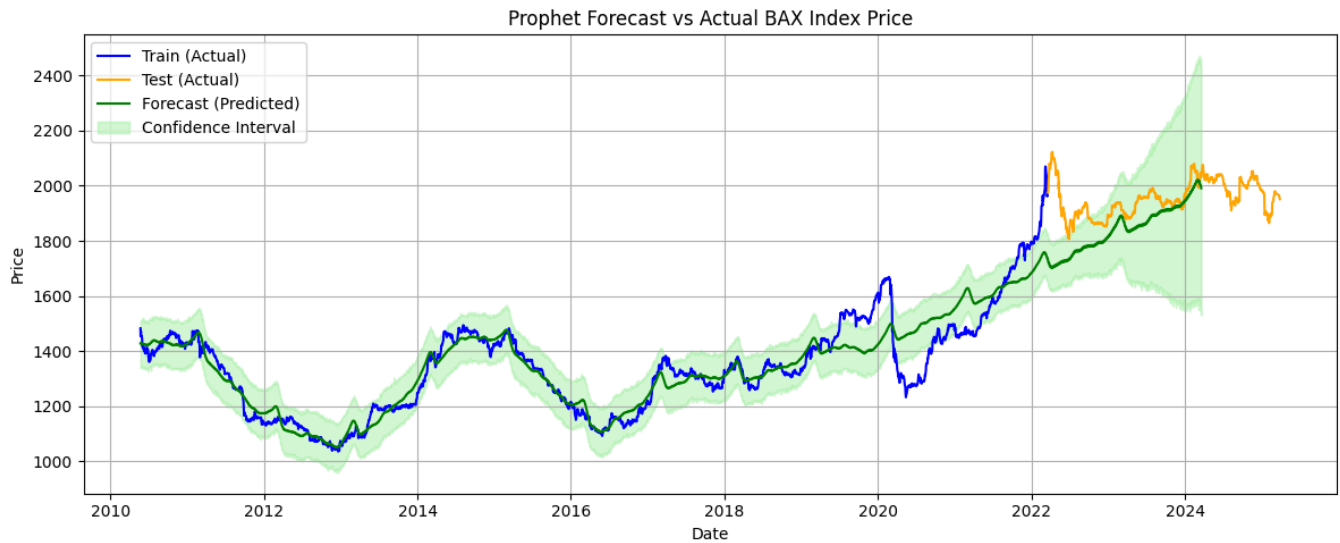
Prophet

```
1 from prophet import Prophet
2
3 df = df.reset_index()
4 df = df[['Date', 'Price']].rename(columns={'Date': 'ds', 'Price': 'y'})
5
6 # Train-test split
7 train_size = int(len(df) * 0.8)
8 train_df = df.iloc[:train_size]
9 test_df = df.iloc[train_size:]
10
11 # Fit Prophet model
12 m = Prophet(daily_seasonality=True)
13 m.fit(train_df)
14
15 # Forecast into the future
16 future = m.make_future_dataframe(periods=len(test_df))
17 forecast = m.predict(future)
18
19 # Plot
20 plt.figure(figsize=(12, 5))
21 plt.plot(train_df['ds'], train_df['y'], label='Train (Actual)', color='blue')
22 plt.plot(test_df['ds'], test_df['y'], label='Test (Actual)', color='orange')
23 plt.plot(forecast['ds'], forecast['yhat'], label='Forecast (Predicted)', color='green')
24 plt.fill_between(forecast['ds'], forecast['yhat_lower'], forecast['yhat_upper'],
25                 color='lightgreen', alpha=0.4, label='Confidence Interval')
26
27 plt.title('Prophet Forecast vs Actual BAX Index Price')
28 plt.xlabel('Date')
29 plt.ylabel('Price')
30 plt.legend()
31 plt.grid(True)
32 plt.tight_layout()
33 plt.show()
```

```

DEBUG:cmdstanpy:input tempfile: /tmp/tmp9gm7gdbd/20k1timx.json
DEBUG:cmdstanpy:input tempfile: /tmp/tmp9gm7gdbd/jpwr1lox.json
DEBUG:cmdstanpy:idx 0
DEBUG:cmdstanpy:running CmdStan, num_threads: None
DEBUG:cmdstanpy:CmdStan args: ['/usr/local/lib/python3.11/dist-packages/prophet/stan_model/prophet_model.bin', 'random', 'seed=8805
13:03:37 - cmdstanpy - INFO - Chain [1] start processing
INFO:cmdstanpy:Chain [1] start processing
13:03:40 - cmdstanpy - INFO - Chain [1] done processing
INFO:cmdstanpy:Chain [1] done processing

```



```

1 # Extract only forecasted values for the test period
2 forecast_test = forecast.iloc[-len(test_df):] # last N predictions
3
4 # Actual and predicted values
5 y_true = test_df['y'].values
6 y_pred = forecast_test['yhat'].values
7
8 # Calculate RMSE
9 rmse = np.sqrt(mean_squared_error(y_true, y_pred))
10 print(f'Prophet RMSE: {rmse:.2f}')

```

```

Prophet RMSE: 140.02

```

```

1 Start coding or generate with AI.

```