Project Documentation: Change Point Analysis and Statistical Modeling of Brent Oil Prices

1. Introduction

This project analyzes **Brent oil price fluctuations** over time by detecting key change points and applying statistical modeling. The aim is to understand the impact of major events (e.g., geopolitical crises, OPEC policy changes) on oil prices and provide insights for investors, policymakers, and energy companies.

The **main objectives** are:

- **Task 1.1:** Define the data analysis workflow.
- **Task 1.2:** Understand the model and data.
- **Task 2:** Perform exploratory data analysis (EDA).
- **Task 3:** Detect significant price change points.
- **Task 4:** Model time series and forecast future prices.

---

2. Project Directory Structure

```
Brent_Oil_Analysis/
├── data/
│   ├── BrentOilPrices.csv  # Raw dataset
│   ├── BrentOilPrices_Cleaned.csv  # Processed dataset
├── notebooks/
│   ├── EDA.ipynb  # Exploratory Data Analysis
│   ├── ChangePointAnalysis.ipynb  # Change point detection
│   ├── TimeSeriesModeling.ipynb  # Forecasting Brent oil prices
├── scripts/
│   ├── data_preprocessing.py  # Data loading & preprocessing
│   ├── change_point_detection.py  # Detecting key events affecting prices
│   ├── model_training.py  # ARIMA time series forecasting
├── reports/
│   ├── analysis_report.pdf  # Summary of findings
├── README.md  # Project documentation
```

---

3. Implementation and Contributions

3.1 Data Preprocessing (`data_preprocessing.py`)

- **Loaded** the dataset (`BrentOilPrices.csv`).
- **Converted** the `Date` column to **datetime format** for time-series analysis.
- **Sorted** the data chronologically to maintain consistency.
- **Checked and handled** missing values (if any).
- **Saved** the cleaned dataset (`BrentOilPrices_Cleaned.csv`).

```
import pandas as pd

file_path = "data/BrentOilPrices.csv"
df = pd.read_csv(file_path)

df['Date'] = pd.to_datetime(df['Date'], format='%d-%b-%y')
df = df.sort_values(by='Date')
df = df.dropna()

df.to_csv("data/BrentOilPrices_Cleaned.csv", index=False)
print("Data preprocessing complete.")
```

3.2 Exploratory Data Analysis (`EDA.ipynb`)

- **Plotted** Brent oil prices over time.
- **Performed** an Augmented Dickey-Fuller (ADF) test to check stationarity.

```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.stattools import adfuller

df = pd.read_csv("data/BrentOilPrices_Cleaned.csv", parse_dates=['Date'])

plt.figure(figsize=(12,6))
plt.plot(df['Date'], df['Price'], label="Brent Oil Price", color='blue')
plt.xlabel("Date")
plt.ylabel("Price (USD per barrel)")
plt.title("Brent Oil Prices Over Time")
plt.legend()
```

```
plt.show()

result = adfuller(df['Price'])
print(f"ADF Statistic: {result[0]}")
print(f"p-value: {result[1]}")
```

3.3 Change Point Detection (`change_point_detection.py`)

- **Used the PELT algorithm** from the `ruptures` library to detect **major structural breaks**.
- **Plotted** detected change points on the Brent oil price trend.

```
import pandas as pd
import matplotlib.pyplot as plt
import ruptures as rpt

df = pd.read_csv("data/BrentOilPrices_Cleaned.csv", parse_dates=['Date'])

prices = df['Price'].values
algo = rpt.Pelt(model="rbf").fit(prices)
change_points = algo.predict(pen=10)

plt.figure(figsize=(12,6))
plt.plot(df['Date'], df['Price'], label="Brent Oil Price")
for cp in change_points:
    plt.axvline(df['Date'].iloc[cp-1], color='red', linestyle='--')

plt.xlabel("Date")
plt.ylabel("Price (USD per barrel)")
plt.title("Detected Change Points in Brent Oil Prices")
plt.legend()
plt.show()

print("Detected Change Points:", [df['Date'].iloc[cp-1] for cp in change_points])
```

3.4 Time Series Forecasting (`model_training.py`)

- **Used the ARIMA model** to forecast oil prices.
- **Plotted** historical data with future predictions.

```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.arima.model import ARIMA

df = pd.read_csv("data/BrentOilPrices_Cleaned.csv", parse_dates=['Date'], index_col='Date')

model = ARIMA(df['Price'], order=(5,1,0))
model_fit = model.fit()

forecast = model_fit.forecast(steps=30)

plt.figure(figsize=(12,6))
plt.plot(df.index, df['Price'], label="Historical Prices")
plt.plot(pd.date_range(df.index[-1], periods=30, freq='D'), forecast, label="Forecast", color='red')
plt.xlabel("Date")
plt.ylabel("Price (USD per barrel)")
plt.title("Brent Oil Price Forecast")
plt.legend()
plt.show()

print(model_fit.summary())
```

---

4. Summary of Findings

1. **Change Point Analysis** identified major shifts in Brent oil prices, likely linked to **geopolitical conflicts, OPEC decisions, and economic sanctions**.
2. **Time Series Forecasting (ARIMA)** provided insights into potential future price trends.
3. The project **equipped investors and policymakers** with data-driven insights for strategic planning.

---

5. Future Work

- Implement **GARCH models** to analyze volatility.
- Extend the dataset to include additional **economic indicators (inflation, interest rates, etc.)**.
- Create a **dashboard** for real-time visualization.
