Title: Time Series Analysis and Forecasting of Stock Prices

Introduction:

This document provides an overview and explanation of the script used for performing Time Series Analysis and Forecasting on stock prices of different companies and indices. The script leverages multiple libraries and tools to download historical stock data, explore data properties, and apply ARMA-GARCH models for analysis and forecasting.

Table of Contents:

1. **Loading Libraries**
2. **Loading Datasets**
3. **Data Understanding**
4. **Data Exploration**
5. **ACF, PACF, ADF Analysis**
6. **AR-GARCH Model Analysis**
7. **Training and Testing the Models**

---

1. Loading Libraries:

- **yfinance**: A library to efficiently download historical market data from Yahoo! Finance.
- **pandas**: A fast, powerful, and easy-to-use library for data analysis and manipulation.
- **numpy**: A library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
- **matplotlib**: A plotting library for creating static, animated, and interactive visualizations.
- **statsmodels**: A library for estimating and interpreting models for statistical analysis.
- **arch**: A library for working with volatility models like GARCH.

```
!pip install yfinance
!pip install arch
!pip install pmdarima

import pandas as pd
import numpy as np
import yfinance as yf
import matplotlib.pyplot as plt
import statsmodels.api as sm
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.stattools import adfuller
from statsmodels.tsa.arima_model import ARIMA
from arch import arch_model
```

2. Loading Datasets:

Download the historical stock data for selected ticker symbols for the specified time period.

```python
# Define the ticker symbols for the stocks you want to download
tickers = ['AAPL', 'GOOG', 'MSFT', 'AMZN', '^GSPC']
...
stock_data = yf.download(tickers, start='2010-01-01', end='2023-02-17')
...
```

3. Data Understanding:

Understanding the structure and content of the data.

```python
stock_data.columns
...
stock_data.info()
...
```

4. Data Exploration:

Visualize the time series data and explore its properties.

```python
...
fig, ax = plt.subplots(figsize=(12, 4))
...
plt.show()
...
```

5. ACF, PACF, ADF Analysis:

Examine the Autocorrelation Function (ACF), Partial Autocorrelation Function (PACF), and perform Augmented Dickey-Fuller (ADF) test to check for stationarity.

```python
...
plot_acf(apple['Adj Close'], zero = False, lags = 2636)
...
plot_pacf(apple['Adj Close'], zero = False)
...
```

6. AR-GARCH Model Analysis:

Implement AR-GARCH models and evaluate their performance for each stock.

```python
...
model = arch_model(train_data['AAPL'], mean='AR', lags=1, vol='GARCH', p=1, q=1)
results = model.fit()
...
print(results.summary())
...
```

7. Training and Testing the Models:

Split the data into training and testing sets, fit the models, and evaluate their performance.

```
...
train_data = data.loc[(data.index >= train_start_date) & (data.index <=
train_end_date)]
test_data = data.loc[(data.index >= test_start_date) & (data.index <= test_end_date)]
...
```

Conclusion:

This script provides a structured approach to analyzing and forecasting stock prices using Time Series Analysis and AR-GARCH models. The analysis includes data loading, exploration, stationarity testing, and model fitting for forecasting.