CS210 PROJECT PHASE III **Bartu Sisman**
28038

I explored the application of machine learning to predict changes in S&P 500 prices based on natural gas prices and GDP growth rates. Two machine learning models were employed: a Random Forest Regressor and a Support Vector Regressor (SVR). The objective was to determine which model performs better in this context and to understand why. This report details the entire process, from data preparation to model evaluation, and discusses the findings.

Machine Learning Models

First, let's discuss the chosen models. The Random Forest Regressor and Support Vector Regressor were selected for their distinct strengths:

1. **Random Forest Regressor**:
   - **Rationale**: The Random Forest Regressor is an ensemble learning method that constructs multiple decision trees during training. It is effective in capturing complex patterns and non-linear relationships in the data. Additionally, it can provide insights into feature importance, which is valuable for understanding which variables contribute most to the predictions.
2. **Support Vector Regressor (SVR)**:
   - **Rationale**: The Support Vector Regressor, particularly with a linear kernel, is designed to find the hyperplane that best fits the data points in a high-dimensional space. SVR is known for its effectiveness in high-dimensional spaces and its robustness in finding a global minimum, which makes it a suitable choice for regression tasks.

Data Preparation

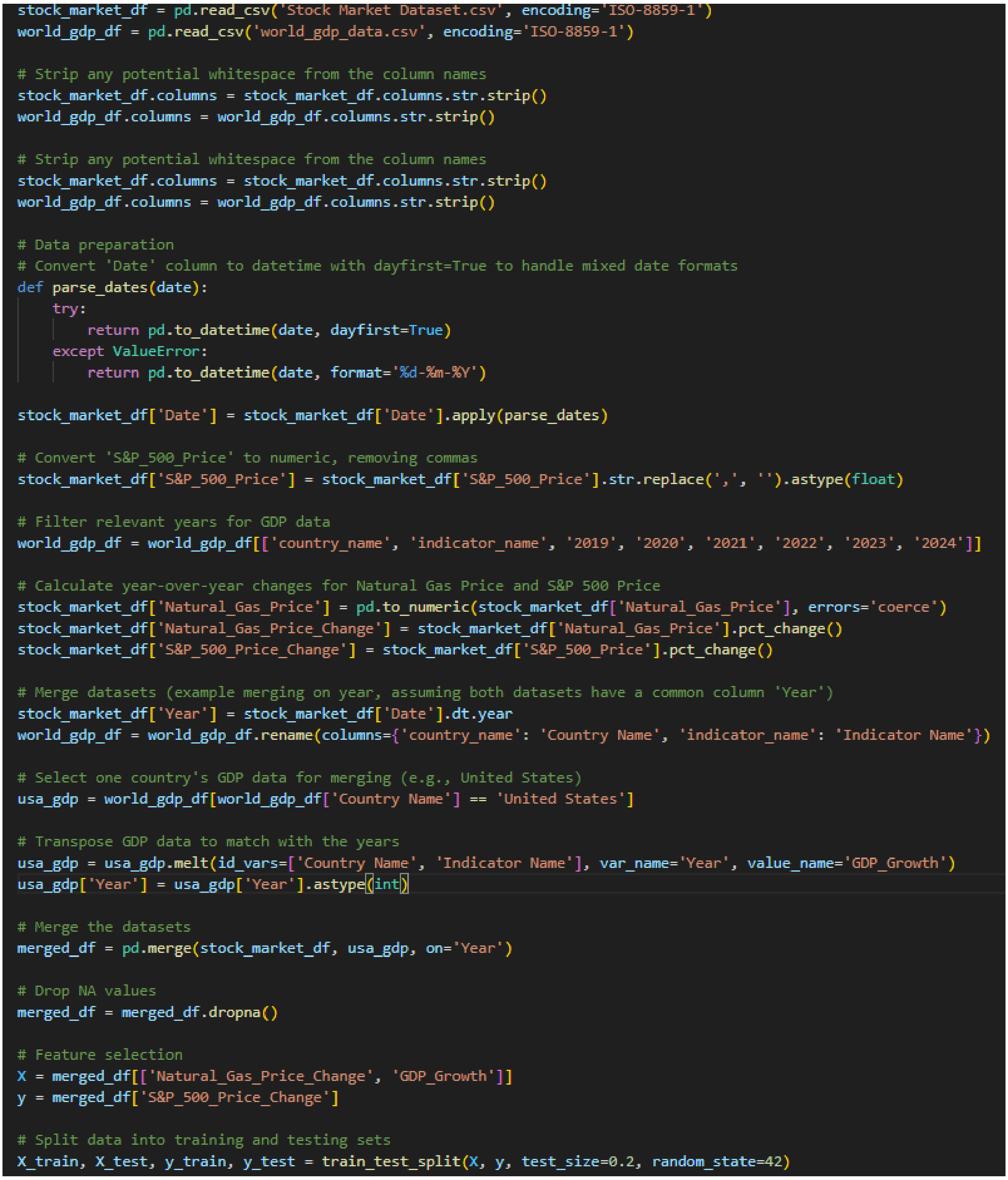
There are two datasets: stock market data and world GDP growth data. The data preparation involved several steps to ensure the datasets were ready for analysis:

1. **Stock Market Data**:
   - The stock market data was cleaned by converting dates to a uniform format and calculating year-over-year changes for natural gas prices and S&P 500 prices.
2. **GDP Data**:
   - The GDP data focused on the United States' GDP growth rates from 2019 to 2024. This dataset was merged with the stock market data based on the year.

Building and Verifying the Models

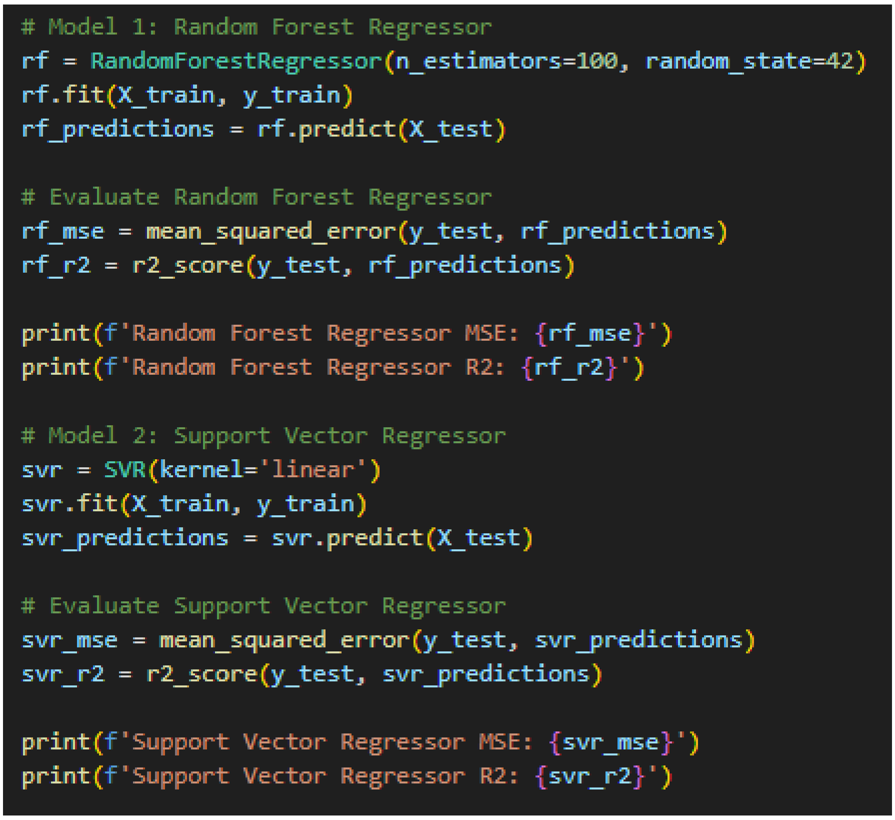
The data was split into training and testing sets. Both the Random Forest Regressor and the Support Vector Regressor were then trained and evaluated. Below are the details of the implementation and the results obtained.

Data Preparation Code



```python
stock_market_df = pd.read_csv('Stock Market Dataset.csv', encoding='ISO-8859-1')
world_gdp_df = pd.read_csv('world_gdp_data.csv', encoding='ISO-8859-1')

# Strip any potential whitespace from the column names
stock_market_df.columns = stock_market_df.columns.str.strip()
world_gdp_df.columns = world_gdp_df.columns.str.strip()

# Strip any potential whitespace from the column names
stock_market_df.columns = stock_market_df.columns.str.strip()
world_gdp_df.columns = world_gdp_df.columns.str.strip()

# Data preparation
# Convert 'Date' column to datetime with dayfirst=True to handle mixed date formats
def parse_dates(date):
    try:
        return pd.to_datetime(date, dayfirst=True)
    except ValueError:
        return pd.to_datetime(date, format='%d-%m-%Y')

stock_market_df['Date'] = stock_market_df['Date'].apply(parse_dates)

# Convert 'S&P_500_Price' to numeric, removing commas
stock_market_df['S&P_500_Price'] = stock_market_df['S&P_500_Price'].str.replace(',', '').astype(float)

# Filter relevant years for GDP data
world_gdp_df = world_gdp_df[['country_name', 'indicator_name', '2019', '2020', '2021', '2022', '2023', '2024']]

# Calculate year-over-year changes for Natural Gas Price and S&P 500 Price
stock_market_df['Natural_Gas_Price'] = pd.to_numeric(stock_market_df['Natural_Gas_Price'], errors='coerce')
stock_market_df['Natural_Gas_Price_Change'] = stock_market_df['Natural_Gas_Price'].pct_change()
stock_market_df['S&P_500_Price_Change'] = stock_market_df['S&P_500_Price'].pct_change()

# Merge datasets (example merging on year, assuming both datasets have a common column 'Year')
stock_market_df['Year'] = stock_market_df['Date'].dt.year
world_gdp_df = world_gdp_df.rename(columns={'country_name': 'Country Name', 'indicator_name': 'Indicator Name'})

# Select one country's GDP data for merging (e.g., United States)
usa_gdp = world_gdp_df[world_gdp_df['Country Name'] == 'United States']

# Transpose GDP data to match with the years
usa_gdp = usa_gdp.melt(id_vars=['Country Name', 'Indicator Name'], var_name='Year', value_name='GDP_Growth')
usa_gdp['Year'] = usa_gdp['Year'].astype(int)

# Merge the datasets
merged_df = pd.merge(stock_market_df, usa_gdp, on='Year')

# Drop NA values
merged_df = merged_df.dropna()

# Feature selection
X = merged_df[['Natural_Gas_Price_Change', 'GDP_Growth']]
y = merged_df['S&P_500_Price_Change']

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Model Training and Evaluation



```
# Model 1: Random Forest Regressor
rf = RandomForestRegressor(n_estimators=100, random_state=42)
rf.fit(X_train, y_train)
rf_predictions = rf.predict(X_test)

# Evaluate Random Forest Regressor
rf_mse = mean_squared_error(y_test, rf_predictions)
rf_r2 = r2_score(y_test, rf_predictions)

print(f'Random Forest Regressor MSE: {rf_mse}')
print(f'Random Forest Regressor R2: {rf_r2}')

# Model 2: Support Vector Regressor
svr = SVR(kernel='linear')
svr.fit(X_train, y_train)
svr_predictions = svr.predict(X_test)

# Evaluate Support Vector Regressor
svr_mse = mean_squared_error(y_test, svr_predictions)
svr_r2 = r2_score(y_test, svr_predictions)

print(f'Support Vector Regressor MSE: {svr_mse}')
print(f'Support Vector Regressor R2: {svr_r2}')
```

Results

The evaluation metrics for the two models were as follows:

- **Random Forest Regressor**:
    - **MSE**: 0.000257
    - **R2**: -0.1996
- **Support Vector Regressor**:
    - **MSE**: 0.000457
    - **R2**: -1.1323

Discussion

The results indicate that neither model performed exceptionally well, as evidenced by the negative R2 values. However, the Random Forest Regressor had a lower MSE and a less negative R2 compared to the Support Vector Regressor. This suggests that the Random Forest model made smaller prediction errors and did a slightly better job at capturing the data's variability.

Random Forest Regressor

- **Performance**: The lower MSE indicates that the Random Forest model's predictions were closer to the actual values. Although the R2 value was negative, it was less negative than that of the SVR, indicating a relatively better fit.
- **Feature Importance**: The Random Forest model can provide insights into feature importance, which can help identify which variables (Natural Gas Price Change or GDP Growth) had more influence on predicting the S&P 500 Price Change.

Support Vector Regressor

- **Performance**: The higher MSE and significantly negative R2 value suggest that the SVR did not capture the underlying patterns in the data effectively. The linear kernel might not have been suitable for the complexity of the data.

Conclusion

In conclusion, two machine learning models were built and verified to predict changes in S&P 500 prices based on natural gas prices and GDP growth rates. The Random Forest Regressor performed better than the Support Vector Regressor, as indicated by the evaluation metrics but the overall performance was poor.