In [1]:
```
import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.model_selection import KFold
```

In [2]:
```
# Load the dataset
data = pd.read_csv('1M_ahead_dataset.csv')
```

In [3]:
```
# Separate predictors and target; assuming 'Yt.1M' is the target.
X = data.drop(['Yt.1M'], axis=1)
y = data['Yt.1M']
```

In [4]:
```
# Set up 5-Fold cross-validation
kf = KFold(n_splits=5, shuffle=True, random_state=42)
fold_metrics = []
fold_counter = 1
```

In [5]:
```
# Loop over each fold
for train_index, test_index in kf.split(X):
    X_train, X_test = X.iloc[train_index], X.iloc[test_index]
    y_train, y_test = y.iloc[train_index], y.iloc[test_index]

    # Initialize the Random Forest Regressor (you can adjust parameters if needed)
    model = RandomForestRegressor(n_estimators=100, random_state=42)
    model.fit(X_train, y_train)

    # Predict on the test set
    y_pred = model.predict(X_test)

    # Compute evaluation metrics
    mse = mean_squared_error(y_test, y_pred)
    mae = mean_absolute_error(y_test, y_pred)
    rmse = np.sqrt(mse)

    fold_metrics.append({'Fold': fold_counter, 'MSE': mse, 'RMSE': rmse, 'MAE': mae})

    # Print feature importances and metrics for this fold
    print(f"\nFold {fold_counter} Feature Importances:")
    print(model.feature_importances_)
    print(f"Fold {fold_counter} -- MSE: {mse:.4f}, RMSE: {rmse:.4f}, MAE: {mae:.4f}")

    fold_counter += 1
```

```
Fold 1 Feature Importances:
[0.13509883 0.04889066 0.0397044  0.11138954 0.07242384 0.04762999
 0.0807937  0.06448376 0.08452475 0.06029241 0.05509629 0.12047118
 0.07920066]
Fold 1 -- MSE: 0.0159, RMSE: 0.1260, MAE: 0.0758

Fold 2 Feature Importances:
[0.12023042 0.04771412 0.03790176 0.11896093 0.09513735 0.06779251
 0.05484233 0.05283297 0.04447057 0.06549216 0.05721671 0.12615097
 0.1112572 ]
Fold 2 -- MSE: 0.0184, RMSE: 0.1357, MAE: 0.0755

Fold 3 Feature Importances:
[0.11773791 0.04895031 0.04391118 0.10410141 0.11568593 0.06091138
 0.06695169 0.05460215 0.06155982 0.04989205 0.05474247 0.08716689
 0.1337868 ]
Fold 3 -- MSE: 0.0140, RMSE: 0.1182, MAE: 0.0752

Fold 4 Feature Importances:
[0.09528621 0.04475375 0.04524532 0.10368317 0.10129583 0.05781189
 0.11120091 0.0642027  0.08263978 0.04649983 0.04249197 0.10819634
 0.0966923 ]
Fold 4 -- MSE: 0.0133, RMSE: 0.1152, MAE: 0.0758

Fold 5 Feature Importances:
[0.12029715 0.04005138 0.04161956 0.13746674 0.12284139 0.05889433
 0.08350061 0.05623217 0.06060666 0.04487732 0.05308742 0.1160085
 0.06451676]
Fold 5 -- MSE: 0.0190, RMSE: 0.1377, MAE: 0.0767
```

In [6]:
```
# Summarize all results into a DataFrame
results_df = pd.DataFrame(fold_metrics)
print("\nOverall Cross-Validation Results:")
print(results_df)
```

```
Overall Cross-Validation Results:
   Fold       MSE      RMSE       MAE
0     1  0.015876  0.126000  0.075766
1     2  0.018417  0.135709  0.075462
2     3  0.013976  0.118220  0.075187
3     4  0.013261  0.115158  0.075764
4     5  0.018955  0.137677  0.076690
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js