Electricity Price **Explanation**

Github Link:
https://github.com/peter-b-k/ensemble-learning-qrt

Peter KESZTHELYI
Qihang PU
Runjia JIANG
Vennela SEELAM

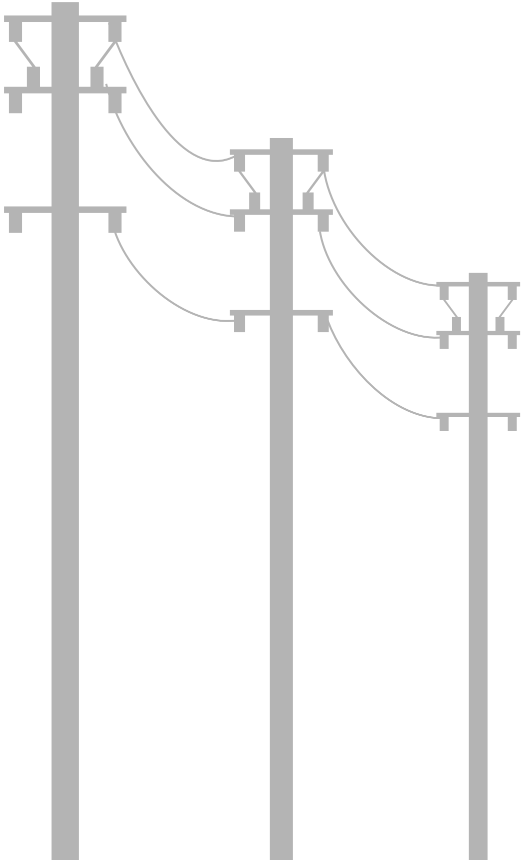
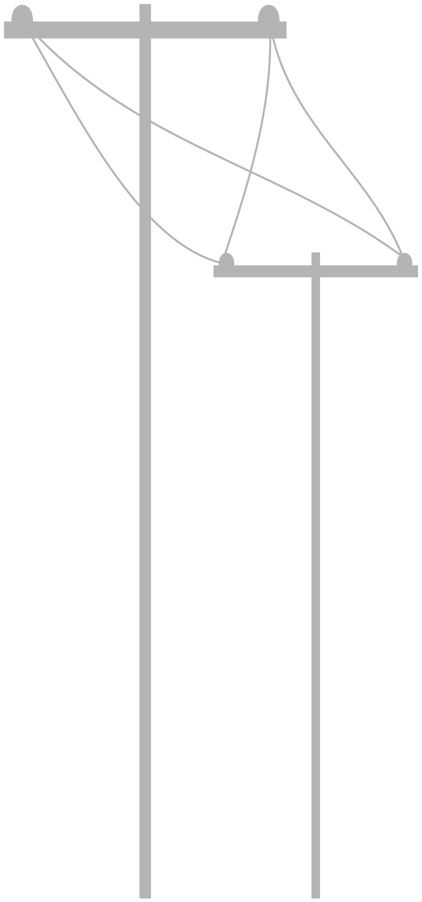


Table of contents

01

Context & Objectives



Context

Objective

AIM

Aims to **model** the **electricity price** from weather, energy and commercial data for two European countries- France and Germany

GOAL

- Applying approaches like
  - Decision Trees,
  - Bagging,
  - Randoms Forests,
  - Gradient Boosting,
  - AdaBoost.
- Comparing the performances using MSE, MAE.

Challenge Overview

The challenge is to learn **a model that outputs** from the explanatory variables a good estimation for the daily price variation of electricity futures contracts in France and Germany.

Explanatory Variables

- Daily commodity price variations
- Weather measures
- Electricity production measures
- Electricity use measures

Data Description

3 CSV File Datasets

- Training inputs X_train
- Training outputs Y_train
- Test inputs X_test

X Input Features

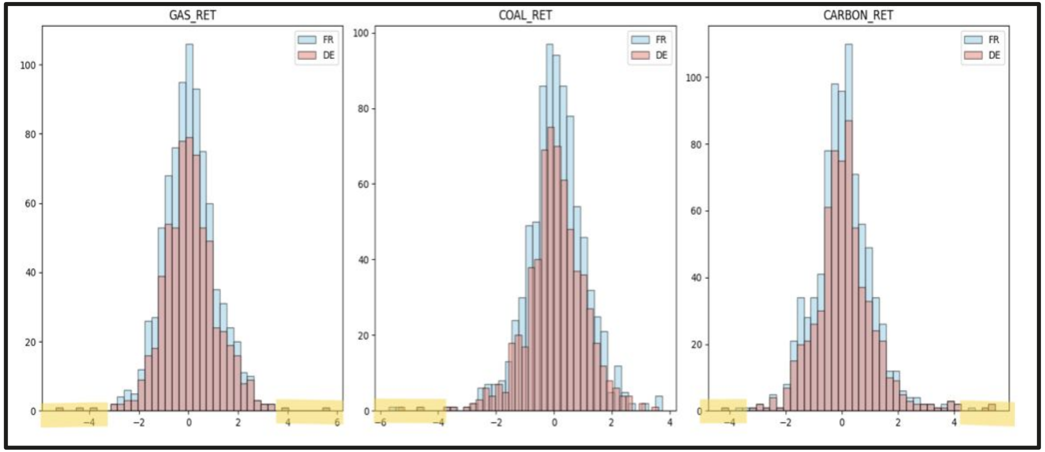
Columns	The columns in X_train and X_test represent the explanatory variables.
Time Periods	Both X_train and X_test have columns representing the same explanatory variables, but over different time periods.
Unique ID	Each row in X_train corresponds to a unique ID associated with a day and a country.
Features	The features include DE_CONSUMPTION, FR_CONSUMPTION,DE_FR_EXCHANGE, FR_DE_EXCHANGE
Missing Values	Some columns in X_train and X_test have missing values that need to be addressed during preprocessing.

Y Target Variable

Column	The column named “TARGET” in Y_train represents the target variable.
Definition	The target variable corresponds to the price change for daily futures contracts of 24H electricity baseload.
Unique ID	Similar to X_train, each in Y_train is associated with unique ID linked to a day and a country.

Exploratory Data Analysis

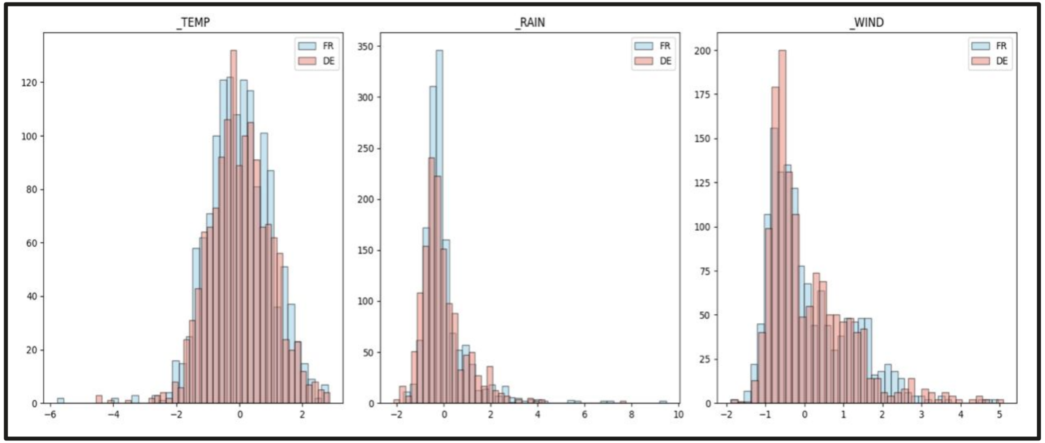
Daily Commodity Price Variation



Price distribution for Europe market, don’t have difference between DE and FR, but contain

a. Outliers

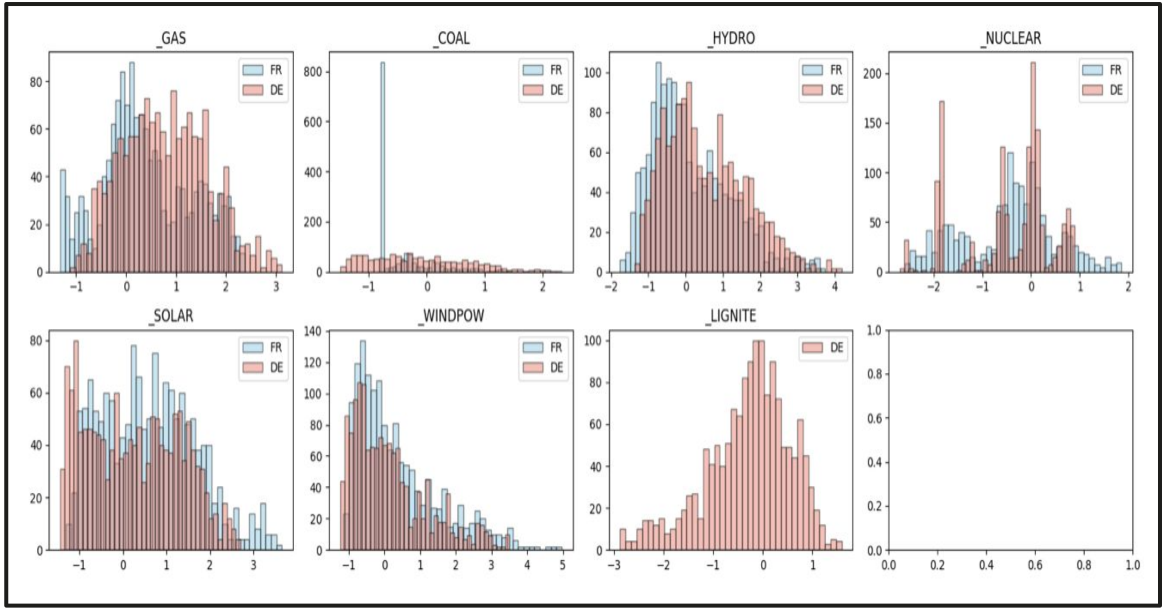
Weather Measures



Because DE and FR are close geographically, so the weather data are similar, but the distribution have

a. Outliers
b. Skewness

Energy Production Measures

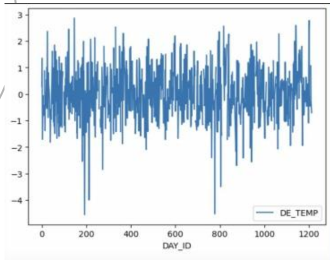


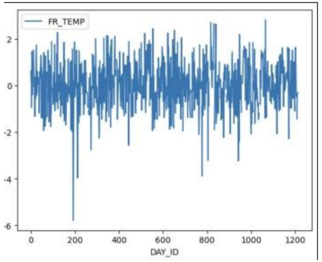
Here, DE & FR have a different energy produce structure:

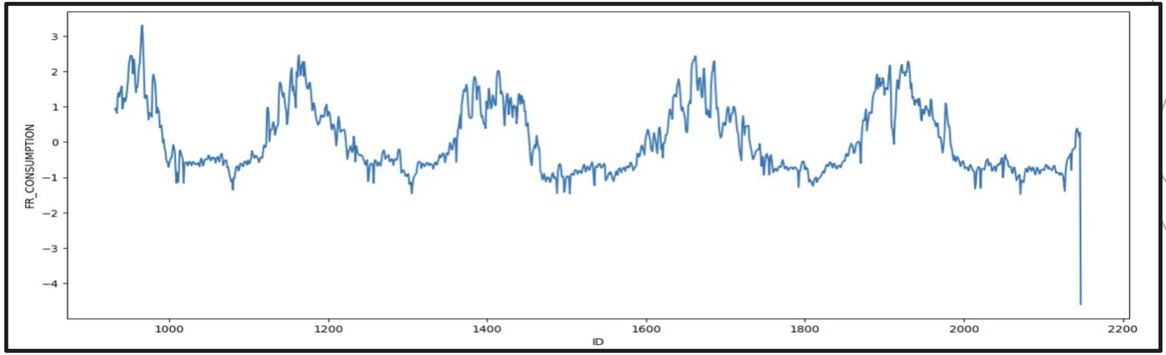
a. DE: relies more on Gas, Lignite
b. FR: Nuclear is one essential part

Electricity Use Metrics

A. Approximate seasonal trends



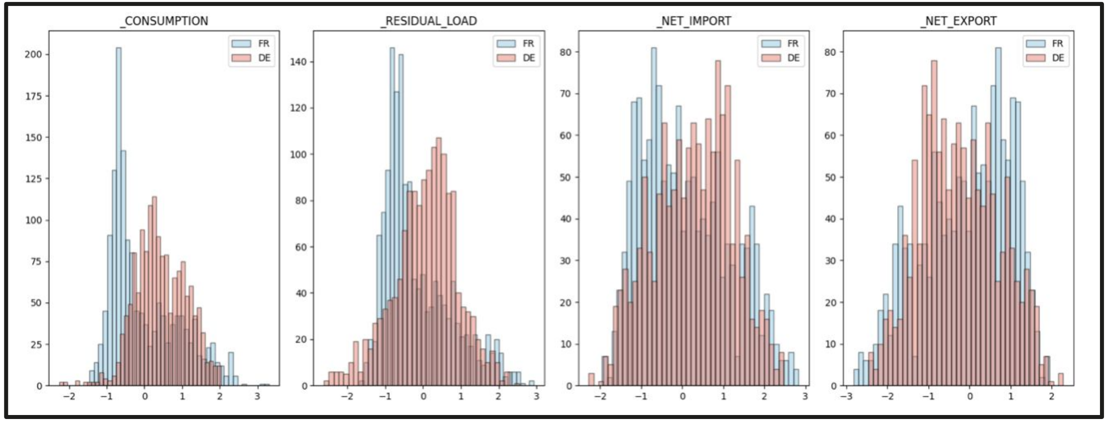




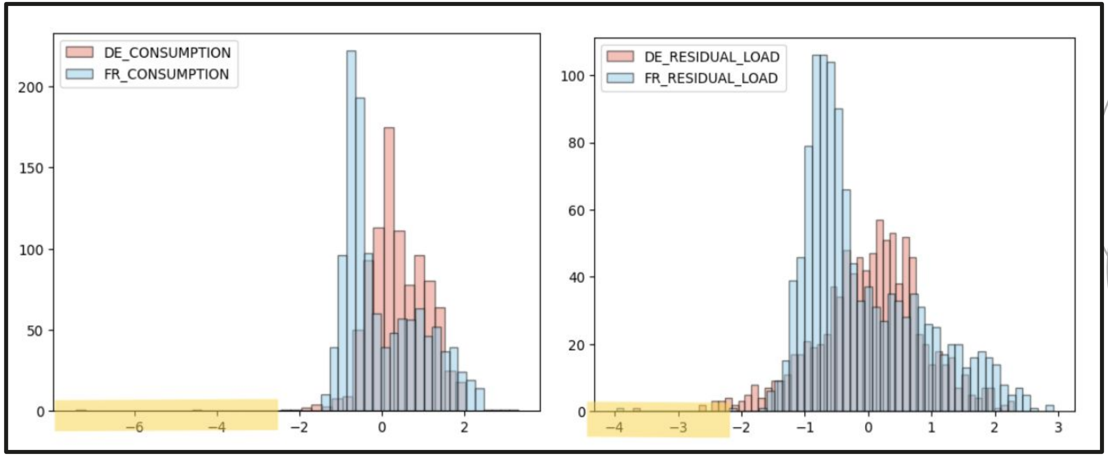
B. CONSUMPTION & RESIDUAL_LOAD:

DE & FR show different distribution

a. If only plot data from *train_x* set



b. If plot data from both train_x set and test_x set **:**



Comparing the distribution of expanded data,

the highlighted part in the X-axis indicates that:

abnormal values exist in test_x set .

C. Use metrics' heatmap

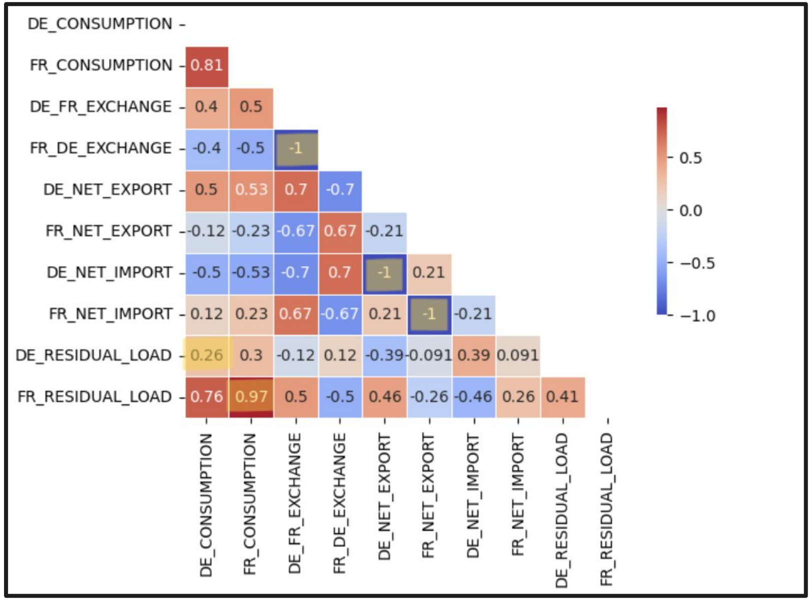
a. Correlation(Consumption, Residual):

DE's 0.26, FR's 0.96

b. EXCHANGE:

FR_DE_EXC = – DE_FR_EXC

c. IMPORT = – EXPORT

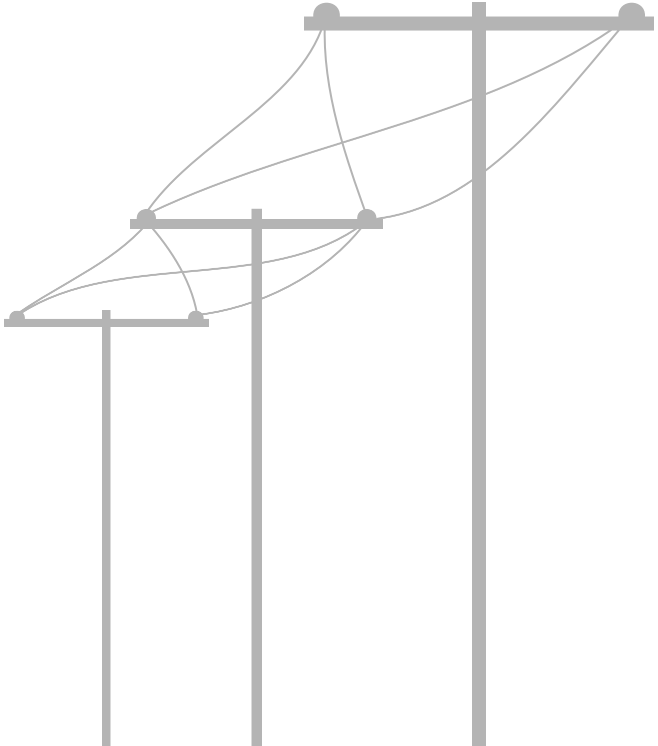


D. Null Value:

Null value only exists in FR and DE has no null value.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 932 entries, 0 to 931
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   DAY_ID            932 non-null    int64
 1   COUNTRY           932 non-null    object
 2   DE_CONSUMPTION    932 non-null    float64
 3   DE_NET_EXPORT     932 non-null    float64
 4   DE_NET_IMPORT     932 non-null    float64
 5   DE_RESIDUAL_LOAD  932 non-null    float64
 6   DE_FR_EXCHANGE    932 non-null    float64
dtypes: float64(5), int64(1), object(1)
memory usage: 58.2+ KB
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1216 entries, 932 to 2147
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   DAY_ID            1216 non-null   int64
 1   COUNTRY           1216 non-null   object
 2   FR_CONSUMPTION    1216 non-null   float64
 3   FR_NET_EXPORT     1122 non-null   float64
 4   FR_NET_IMPORT     1122 non-null   float64
 5   FR_RESIDUAL_LOAD  1216 non-null   float64
 6   FR_DE_EXCHANGE    1182 non-null   float64
dtypes: float64(5), int64(1), object(1)
memory usage: 76.0+ KB
```



02 Data Preprocessing

Data Preprocessing

The goal of data preprocessing is to clean, transform, and prepare the dataset for analysis and modeling.

Benefits of preprocessing

- Enhancing the model performance
- Improving data quality
- Increasing model robustness

Preprocessing Functions

- `trim_tail` function: trim the tail of the data to reduce the influence of extreme values.
- `do_knn_impute` function: perform KNN imputation of missing values.
- `load_preprocess` function: apply the entire preprocessing transformation.

Feature Engineering

Lag Items

- In-week lag features for Germany (DE) and France (FR).
- Includes Consumption, Net Export and Residual Load...
- Comparing lag vs. no-lag data in our models, lag items perform better. **In-week lag items capture temporal dependencies.**

Consumption Related Trends

- **Average Commodity Price Variations**: smoothed via moving averages for gas, coal, carbon, etc.
- **Nuclear Ratio Trend**: Trends in nuclear energy ratio for DE and FR captured.
- **New Energy Transformation Efficiency**: Efficiency measures for hydro and wind energy relative to environmental factors computed.
- **Residual Load Premium Cost**: Cost implications of residual load and net imports estimated based on commodity price variations.

03

Modeling

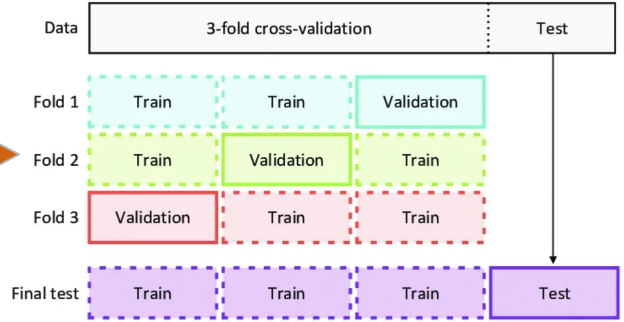
Hyper Parameter Tuning

Cross-Validation on Train Set

Divide the data set into 2 parts → **Train the model on training set** → **Validate the model on test set**

- 80-20 train-test split on the train dataset.
- Use the test set to evaluate our tuned models.

- K-Fold Cross-Validation



Method: GridSearchCV

Predictive Models for Electricity Price Variation

Model Selection

- Decision Tree Regression
- Random Forest
- Bagging
- Ada Boost
- Gradient Boosting
- Extra Tree Regression
- XGBoost

Decision Tree Regression

- The model is tuned separately for France (FR) and Germany (DE).
- Different parameters were identified for each country and emphasizing the need for country-specific training.

.

Decision Tree for France (FR)

- **Best Parameters**: {'criterion': 'absolute_error', 'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 10}
- **MSE**: 1.1148, **MAE:** 0.5511
- **Spearman correlation**: 21.7%

Decision Tree for Germany(DE)

- **Best Parameters: {**'criterion': 'absolute_error', 'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 10}
- **MSE:** 0.9809, **MAE:** 0.6362
- **Spearman correlation:** 43.5%

Decision Tree Overall

- **Best Parameters**: {'criterion': 'absolute_error', 'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 10}
- **MSE**: 1.4075, **MAE**: 0.6866
- **Spearman correlation**: -1.8%

Random Forest Regressor

- The Random Forest model was tuned separately for France (FR) and Germany (DE).
- Different optimal parameters were identified for each country, highlighting the need for country-specific tuning.
- The overall Random Forest model, combined both countries, showed an intermediate performance with a correlation of 10.5%.
- Tuning parameters led to improvements in model performance,

Random Forest for France (FR)

- **Best Parameters:** {'max_depth': 15, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 100}
- **MSE:** 0.9879, **MAE:** 0.5185
- **Spearman correlation:** 7.0%

Random Forest for Germany (DE)

- **Best Parameters:** {'max_depth': None, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 100}
- **MSE:** 0.5354, **MAE:** 0.4780
- **Spearman correlation:** 57.4%

Random Forest Overall

- **Best Parameters:** {'max_depth': 15, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 100}
- **MSE:** 1.1493, **MAE:** 0.6328
- **Spearman correlation:** 10.5%

Bagging

- Bagging Regressor was tuned separately for France (FR) and Germany (DE), emphasizing the significance of country-specific tuning.
- Different optimal parameters were identified for each country.
- The overall Bagging Regressor model, combining both countries, demonstrated an intermediate performance with a correlation of 14.1%.

Bagging for France (FR)

- **Best Parameters:** {'bootstrap': True, 'bootstrap_features': False, 'max_features': 0.5, 'max_samples': 0.5, 'n_estimators': 200}
- **MSE:** 0.9871, **MAE:** 0.5096
- **Spearman correlation:** 13.1%

Bagging for Germany(DE)

- **Best Parameters:** {'bootstrap': True, 'bootstrap_features': False, 'max_features': 1.0, 'max_samples': 0.5, 'n_estimators': 100}
- **MSE:** 0.5561, **MAE:** 0.4951
- **Spearman correlation:** 54.4%

Bagging Overall

- **Best Parameters:** {'bootstrap': True, 'bootstrap_features': False, 'max_features': 0.5, 'max_samples': 0.5, 'n_estimators': 200}
- **MSE:** 1.1335, **MAE:** 0.6245
- **Spearman correlation:** 14.1%

Ada Boost

- AdaBoost Regressor was tuned separately for France (FR) and Germany (DE), emphasizing country-specific adjustments.
- Optimal parameters were identified for both countries, highlighting robustness across regions.
- The overall AdaBoost Regressor model, combined both countries, demonstrated a moderate performance with a correlation of 11.0%.

Adaboost for France (FR)

- **Best Parameters:** {'learning_rate': 0.1, 'n_estimators': 100}
- **MSE:** 0.9997, **MAE:** 0.4871
- **Spearman correlation:** 1.0%

Adaboost for Germany (DE)

- **Best Parameters:** {'learning_rate': 0.1, 'n_estimators': 100}
- **MSE**: 0.6264, **MAE:** 0.5451
- **Spearman correlation:** 55.0%

Adaboost Overall

- **Best Parameters:** {'learning_rate': 0.01, 'n_estimators': 50}
- **MSE:** 1.1048, **MAE:** 0.6078
- **Spearman correlation:** 11.0%

Gradient Boosting

- Gradient Boosting Regressor was tuned separately for France (FR) and Germany (DE), emphasizing region-specific adjustments.
- The chosen parameters showed the distinct performances: FR with lower correlation and DE with higher correlation, emphasizing country-specific nuances.
- The overall Gradient Boosting Regressor model demonstrated a moderate performance with a correlation of 17.2%.

Gradient Boosting for France (FR)

- **Best Parameters:** {'learning_rate': 0.01, 'n_estimators': 50}
- **MSE:** 0.9703, **MAE:** 0.4610
- **Spearman correlation:** -2.3%

Gradient Boosting for Germany (DR)

- **Best Parameters:** {'learning_rate': 0.05, 'n_estimators': 50}
- **MSE:** 0.5816, **MAE:** 0.5014
- **Spearman correlation:** 55.3%

Gradient Boosting Overall

- **Best Parameters:** {'learning_rate': 0.01, 'n_estimators': 50}
- **MSE:** 1.1030, **MAE:** 0.5975
- **Spearman correlation:** 17.2%

Extra Tree Regression

- Extra Trees Regressor was tuned separately for France (FR) and Germany (DE), considering country-specific requirements.
- It is observed that FR having lower correlation and DE exhibiting a significantly higher value.
- The overall Extra Trees Regressor model displayed a moderate correlation of 26.8%.

Extra Tree for France (FR)

- **Best Parameters:** {'max_depth': 20, 'n_estimators': 100}
- **MSE:** 1.0299, **MAE:** 0.5582
- **Spearman correlation:** 6.4%

Extra Tree for Germany (DR)

- **Best Parameters:** {'max_depth': None, 'n_estimators': 500}
- **MSE:** 0.5415, **MAE:** 0.4749
- **Spearman correlation:** 59.5%

Overall Extra Tree

- **Best Parameters:** {'max_depth': 10, 'n_estimators': 500}
- **MSE:** 1.0935, **MAE:** 0.6041
- **Spearman correlation:** 26.8%

XGBoost

- Tuning XGBoost Regressor individually for France (FR) and Germany (DE) led to distinctive parameter preferences.
- FR exhibited a negative correlation, indicating potential challenges in capturing trends.
- DE, demonstrated a positive correlation, suggesting a better model fit for the German dataset.
- The overall XGBoost Regressor model presented a moderate correlation of 10.1%, with insights into parameter impact on performance.

XGBoost for France (FR)

- **Best Parameters:** {'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 50}
- **MSE:** 0.9743, **MAE:** 0.4646
- **Spearman correlation:** -5.8%

XGBoost for Germany (DR)

- **Best Parameters:** {'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 150}
- **MSE:** 0.5374, **MAE:** 0.4740
- **Spearman correlation:** 58.4%

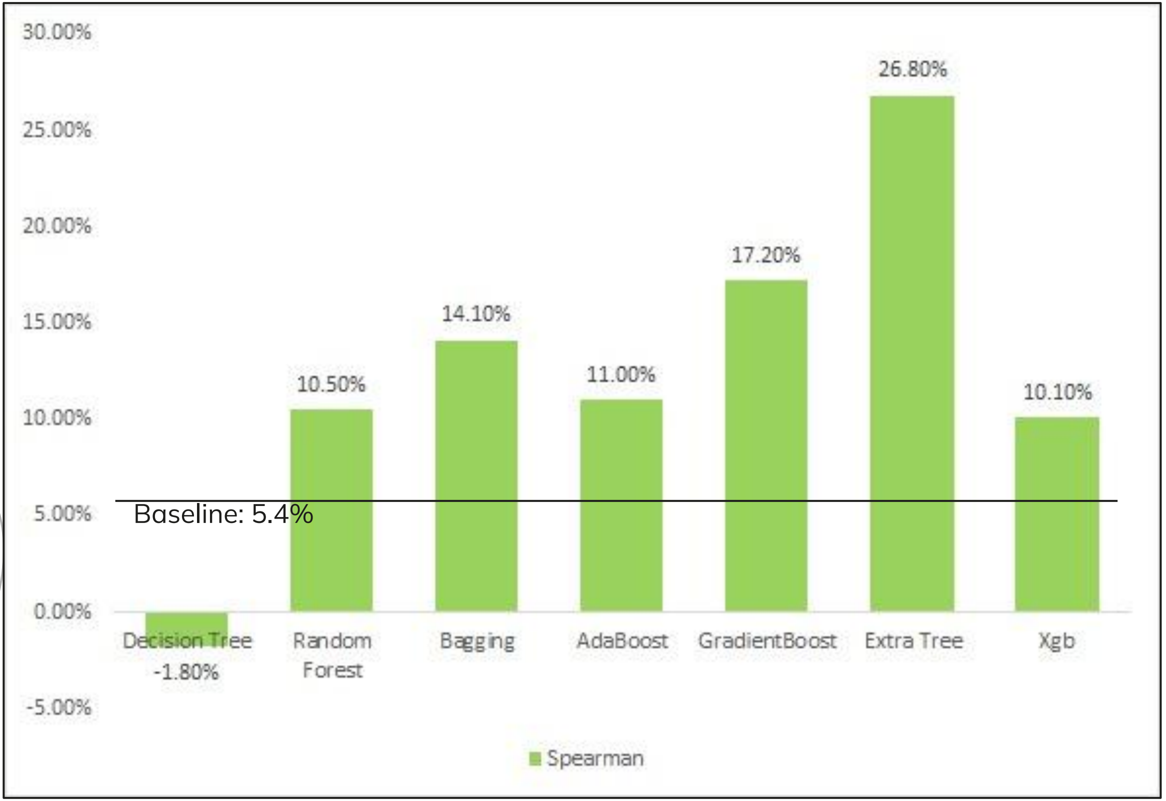
Overall XGBoost

- **Best Parameters:** {'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 50}
- **MSE:** 1.1023, **MAE:** 0.5958
- **Spearman correlation:** 10.1%

Model Performance Comparison

- The Decision Tree model has the highest MSE, indicating less accuracy in the prediction.
- The ExtraTree model has the lowest MSE.

Spearman Correlation Coefficient



- Decision Tree shows a negative Spearman correlation coefficient of -1.80%, indicating a weak and inverse relationship
- Random Forest, Bagging and Xgb models show a moderate positive correlation, with values of 10.50%, 14.10% and 10.1%
- Gradient Boost shows a better correlation at 17.20%
- Extra Tree stands out with the highest Spearman correlation at 26.80%

Thanks!

Q&A