# Assignment 2

## Reporting Section

**Try to classify the end use of the energy by the GHG emissions and temperature**:

- The target variable, END_USE, is classified based on features that include GHG-related fuel usage (e.g., Coal, Diesel, Natural_gas, etc.) and temperature (Temp_degC).

**Pick 3 models**:

- The code uses three different models:
  - **Decision Tree Classifier** (DecisionTreeClassifier)
  - **Random Forest Classifier** (RandomForestClassifier)
  - **Support Vector Machine (SVM)** (SVC)

**Try 3 different hyperparameter settings on each**:

- For each of the three models, three different hyperparameter settings are used:
  - **Decision Tree**:
    - max_depth=10
    - max_depth=20
    - criterion='entropy'
  - **Random Forest**:
    - n_estimators=50, max_depth=10
    - n_estimators=100, max_depth=20
    - n_estimators=200, criterion='entropy'
  - **SVM**:
    - kernel='linear'
    - kernel='rbf'
    - kernel='poly', degree=3

**At least one hyperparameter must be a property of the underlying model (like tree size, kernel type, etc.)**:

- The code changes hyperparameters that are specific to the model:
  - **Decision Tree**: max_depth and criterion
  - **Random Forest**: n_estimators, max_depth, and criterion

- **SVM**: kernel type and degree for the polynomial kernel

**Get the overall training and test accuracy for each of the 9 models**:

- The code prints both **training accuracy** and **test accuracy** for each of the 9 models.

**Also get the per-class prediction accuracy for each model (per class accuracy or F1 score for each of the three outcome classes)**:

- The code calculates the **F1 score** for each of the three outcome classes (Process Heating, CHP, and Boiler Use) using the f1_score function with average=None. This is done for all 9 models.

From Here I have described all of the above mentioned things in details:

### Models Used:

1. **Decision Tree Classifier**:
   - **Hyperparameters used**:
     - max_depth=10: Restricts the depth of the tree to 10 levels, balancing complexity and generalization.
     - max_depth=20: Allows for deeper trees, making the model more expressive but also prone to overfitting.
     - criterion='entropy': Uses information gain (entropy) to decide how to split the data at each node, instead of the default "gini" index.
2. **Random Forest Classifier**:
   - **Hyperparameters used**:
     - n_estimators=50: Builds 50 decision trees and averages their predictions to improve generalization.
     - n_estimators=100: Increases the number of trees to 100, aiming to reduce overfitting by aggregating more trees.
     - n_estimators=200, criterion='entropy': Expands to 200 trees and uses entropy as the splitting criterion.
3. **Support Vector Machine (SVM)**:
   - **Hyperparameters used**:
     - kernel='linear': A linear kernel that attempts to classify data with a straight hyperplane.
     - kernel='rbf': A radial basis function (RBF) kernel that allows non-linear classification using Gaussian functions.

- kernel='poly', degree=3: Uses a polynomial kernel with degree 3 to create a non-linear decision boundary by mapping the data into higher dimensions.

### Overall Training and Test Accuracy for the 9 Models:

| Model | Training Accuracy | Test Accuracy |
|---|---|---|
| Decision Tree (max_depth=10) | 87.11% | 85.60% |
| Decision Tree (max_depth=20) | 94.73% | 84.70% |
| Decision Tree (criterion='entropy') | 96.71% | 84.90% |
| Random Forest (n_estimators=50) | 85.79% | 82.65% |
| Random Forest (n_estimators=100) | 95.93% | 85.05% |
| Random Forest (n_estimators=200, entropy) | 96.73% | 84.45% |
| SVM (kernel='linear') | 75.91% | 75.65% |
| SVM (kernel='rbf') | 76.34% | 76.30% |
| SVM (kernel='poly', degree=3) | 76.20% | 75.80% |

### Per-Class Prediction Accuracy (F1 Score) for Each of the 9 Models:

| Model | Process Heating F1 | CHP F1 | Boiler F1 |
|---|---|---|---|
| Decision Tree (max_depth=10) | 0.854 | 0.681 | 0.973 |
| Decision Tree (max_depth=20) | 0.839 | 0.680 | 0.976 |
| Decision Tree (criterion='entropy') | 0.840 | 0.687 | 0.977 |
| Random Forest (n_estimators=50) | 0.833 | 0.575 | 0.967 |
| Random Forest (n_estimators=100) | 0.844 | 0.689 | 0.975 |
| Random Forest (n_estimators=200, entropy) | 0.836 | 0.683 | 0.973 |
| SVM (kernel='linear') | 0.793 | 0.000 | 0.961 |
| SVM (kernel='rbf') | 0.797 | 0.074 | 0.961 |
| SVM (kernel='poly', degree=3) | 0.794 | 0.022 | 0.961 |

## Conclusion Section

### Best Model:

The best-performing model in terms of both accuracy and F1 score was the **Random Forest Classifier** with n_estimators=100 and max_depth=20. This model achieved a **Test Accuracy of 85.05%** and relatively balanced F1 scores across all three classes: Process Heating (0.844), CHP (0.689), and Boiler Use (0.975). The use of more trees in the Random Forest seems to have improved its generalization, leading to better performance across the classes.

### Worst Model:

The **Support Vector Machine (SVM) with kernel='linear'** was the worst-performing model, with a **Test Accuracy of 75.65%** and very poor F1 scores for the CHP class (0.000). This suggests that a linear decision boundary was not sufficient to separate the classes effectively, particularly the CHP class, which had very few correct classifications. The low accuracy and F1 scores across multiple SVM models (linear, RBF, and poly) indicate that the dataset may require more complex, non-linear separations than the SVM models provided.

### Conjecture as to Why Certain Models Did Well or Poorly:

- **Decision Trees**: The deeper decision trees performed better in terms of training accuracy but showed signs of overfitting (higher training accuracy than test accuracy). Shallow trees (e.g., max_depth=10) struggled to capture the complexity in the data, leading to lower test accuracy.
- **Random Forests**: Random Forests performed well due to their ability to reduce overfitting by averaging multiple decision trees. The model with n_estimators=100 struck a good balance between complexity and generalization, explaining its superior performance. The slightly lower performance of the n_estimators=200 model suggests diminishing returns after a certain number of trees.
- **SVMs**: The linear SVM performed poorly, likely because the data is not linearly separable. The RBF and polynomial kernels allowed for more flexibility, but the CHP class consistently had low F1 scores, indicating that the SVM models struggled with this particular class. This might be because of an imbalance or nonlinear complexity in the dataset that was not well-captured by SVM models.

## What I Learned From the Project:

Through this project, I learned that different machine learning models have varying strengths and weaknesses when applied to real-world datasets. Decision Trees and Random Forests were better suited to this dataset because they can capture complex interactions between features. Random Forests, in particular, proved to be the best due to their ensemble approach, which reduces overfitting and improves generalization.

I also learned that SVMs, while powerful in some cases, require careful tuning of hyperparameters (especially the kernel) and may not perform well on certain types of data. The results show that simpler models like linear SVMs are not effective when the data is non-linearly separable.

Overall, this project helped me understand the importance of selecting the right model and hyperparameters based on the dataset and classification task.