**ANALYSIS**

The problem presents a challenge where a need to provide two outputs is stablished: Carbon emissions and Energy consumption per country. This situation can be analysed using two possible approaches. I will be presenting them, bearing in mind their respective pros and cons to make the decision on what to use:

|  |  |
| --- | --- |
| **Multi-Output Regression** | **Ensemble Methods** |
| Train a single model that predicts both targets simultaneously. Many regression algorithms in sci-kit-learn, such as Random Forest Regression, Support Vector Regression (SVR), and Gradient Boosting Regression, support multi-output regression directly. You can train the model using the features and both target variables, and the model will learn to predict both targets at the same time. | Instead of training a single model, you can train multiple models, each predicting a single target variable. Then, you can combine the predictions of these models using ensemble techniques such as averaging or stacking. This approach allows you to use different models for each target, potentially improving predictive performance. |
| ***Pros*** | |
| **Simplicity:** Multi-output regression provides a straightforward and easy-to-implement approach to predict multiple targets simultaneously. | **Flexibility*:*** Ensemble methods allow you to train multiple models independently, potentially capturing different aspects of the relationship between features and targets. This flexibility can be beneficial when the relationships between features and targets are complex or heterogeneous. |
| **Model Complexity:** It trains a single model to capture relationships between features and multiple targets, potentially leading to a simpler and more interpretable model compared to ensemble methods. | **Performance:** Ensemble methods can often achieve better predictive performance compared to single models, especially when there are nonlinear relationships or significant variations between targets. |
| **Computational Efficiency:** Training a single model is often computationally more efficient than training multiple models separately. | **Robustness:** Ensemble methods are generally more robust for outliers and noise in the data compared to single models. |
| **Cons** | |
| **Lack of Flexibility:** Multi-output regression assumes that all targets are related in a similar way to the input features, which may not always be the case. It may not capture complex interactions between features and targets as effectively as ensemble methods. | **Complexity:** Ensemble methods involve training and combining multiple models, which can be more complex and require more computational resources compared to multi-output regression. |
| **Performance:** If the relationships between features and targets are highly nonlinear or there are significant differences in the relationships between targets, a single model may not perform as well as ensemble methods. | **Interpretability:** The predictions of ensemble methods may be less interpretable compared to single models, as they involve combining the predictions of multiple models. |

**Recommendation:**

* If the relationships between features and targets are relatively simple and homogeneous, and computational efficiency is important, multi-output regression may be a suitable choice.
* If the relationships between features and targets are complex, nonlinear, or heterogeneous, and achieving the best possible predictive performance is a priority, ensemble methods may be more appropriate.

In practice, it's often beneficial to experiment with both approaches and evaluate their performance using cross-validation or other validation techniques to determine which one works best for your specific problem. Additionally, ensembles of multi-output regression models are also a viable option, combining the advantages of both approaches.

EXAMPLE

<https://samanemami.medium.com/analysis-of-the-performance-of-the-ensemble-model-for-a-multi-output-regression-problem-abcf98cb2afc>

<https://www.kaggle.com/code/gcmadhan/carbon-emission-by-country>

Comparison of models

<https://medium.com/@tubelwj/developing-multi-class-regression-models-with-python-c8beca5dd482>

EDA:

<https://www.kaggle.com/code/inversion/getting-started-eda>

<https://www.kaggle.com/code/inversion/getting-started-modeling>