**Model Development Phase Template**

|  |  |
| --- | --- |
| Date | 3 May 2024 |
| Team ID | 737820 |
| Project Title | EcoForecast: AI-Powered Prediction of Carbon Monoxide Levels |
| Maximum Marks | 6 Marks |

**Model Selection Report**

In the forthcoming Model Selection Report, various models will be outlined, detailing their descriptions, hyperparameters, and performance metrics, including Accuracy or F1 Score. This comprehensive report will provide insights into the chosen models and their effectiveness.

**Model Selection Report:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Description** | **Hyperparameters** | **Performance Metric (e.g., Accuracy, F1 Score)** |
| Decision Tree Regressor | The Decision Tree Regressor was explored because it can capture non-linear relationships between features and the target variable. Decision trees are interpretable and can handle both numerical and categorical data. However, they are prone to overfitting, especially if the tree is allowed to grow too deep. | n\_estimators=100  , max\_depth=3,learning\_rate=0.1, random\_state=0 | Training Accuracy= 0.4111602033098315  Test Accuracy= 0.368581227731857  R-squared: 0.368581227731857 |
| Random Forest Regressor | Random Forest Regressor emerged as a contender due to its ability to mitigate overfitting and handle noisy data. It is an ensemble method that combines multiple decision trees, providing robust predictive accuracy. Random forests are less sensitive to outliers and noise compared to individual decision trees. | n\_estimators=20, random\_state = 0 | R\_squared: 0.36726404580326355  Training Accuracy 0.4102458601234813 Test Accuracy= 0.36726404580326355 |
| XGBoost | XGBoost was considered for its advanced implementation of gradient boosting, which often outperforms other algorithms in terms of predictive accuracy. It is highly customizable and can handle missing values. XGBoost is particularly suitable for modeling complex relationships in the data. | n\_estimators,learning\_rate, max\_depth, random state | R-squared: 0.27036413286966887  Training Accuracy 0.29635162610429255  Test Accuracy 0.27036413286966887 |
| K-Nearest Neighbour | KNN was assessed for its simplicity and intuitive approach. It makes predictions based on the average of the k-nearest neighbors in the feature space, without making strong assumptions about the underlying data distribution. KNN is suitable for capturing complex, non-linear relationships, especially in smaller datasets. | n\_neighbors | 1. squared: 0.32659350142943666   Training Accuracy= 0.3583124866059221  Test Accuracy= 0.32659350142943666 |
| Linear Regression | Linear regression was considered due to its simplicity and interpretability. It assumes a linear relationship between the input features and the target variable, which can be advantageous if the relationship is indeed linear or can be reasonably approximated as such. However, it may not perform well if the relationship is highly non-linear. |  | Training Accuracy= 0.06503968505710755  Test Accuracy= 0.061523854715849424  R\_squared= 0.061523854715849424 |