Index No: - 210329E

Name: - Laksara K.Y

Final Report - ML Olympiad - Sustainable Urban Living

Evaluation Metrics

Four key evaluation metrics were employed to assess the accuracy of the models developed through each approach:

1. **Mean Absolute Error (MAE):** MAE measures the average absolute differences between predicted and actual habitability scores. It quantifies the average magnitude of errors without considering their direction.
2. **Mean Squared Error (MSE):** MSE calculates the average of the squared differences between predicted and actual habitability scores. It penalizes larger errors more heavily than smaller errors due to the squaring operation**.**
3. **Root Mean Squared Error (RMSE):** RMSE is the square root of the MSE and is the primary metric for evaluation in the competition. It provides a measure of the average magnitude of prediction errors and is directly interpretable in the same units as the original habitability scores.
4. **R-squared (R2):** R2 measures the proportion of the variance in the habitability scores that is predictable from the independent variables. It indicates the goodness of fit of the model to the observed data, with higher values representing better model performance.

Pros of RMSE Evaluation Metrix

- Enhanced Outlier Sensitivity: RMSE's heightened sensitivity to outliers aids in identifying and mitigating extreme deviations in predictions, especially when large errors are undesirable.
- Simplified Gradient Computation: As a loss function, RMSE allows for easier computation of gradients, streamlining optimization processes during model training.

Cons of RMSE Evaluation Metrix

- Comparative Benchmarking Required: RMSE values must be compared with other benchmarks to assess model effectiveness, as RMSE alone does not provide an inherent indication of performance quality.

Key Approaches

In the effort to create a good model for predicting habitability scores, we tried different methods to get the best results. This report highlights the main approaches we used, showing what worked well and what challenges we faced.

1. **Utilization of Cross-Validation Strategy: -**Employing a rigorous cross-validation strategy, various algorithms were evaluated to ascertain the most suitable for prediction purposes. This involved assessing the performance of RandomForest Regressor, XGB Regressor, LGBMRegressor, MLPRegressor (Neural Network), DecisionTreeRegressor, Lasso, GradientBoostingRegressor, and LinearRegression.
2. **Training Model Solely with RandomForest Regressor: -** Training the model exclusively with RandomForest Regressor involved a focused exploration, aiming to harness the inherent strengths of the algorithm for habitability score prediction. This approach included fine-tuning hyperparameters to optimize model performance.
3. **Training the Model Solely with XGBRegressor: -**In a similar manner, I focused on training the model exclusively with XGBRegressor. I carefully adjusted the hyperparameters to improve the model's performance.
4. **Employment of Ensemble Methods (Advanced Stacking Model - StackedRegressor): -**
    - **Method 1**: - stacked regression model using a combination of RandomForest Regressor and XGB Regressor.
    - **Method 2**: - I expanded the ensemble to include LGBMRegressor and MLPRegressor (Neural Network) in addition to RandomForest Regressor and XGB Regressor.

Note: - Additionally, I trained the model solely using the LinearRegressor algorithm. However, I found its accuracy to be slightly lower compared to other algorithms. Therefore, it is not included as a main approach in this report.

Comparison Of Results

The models from each approach underwent rigorous evaluation across separate datasets (training, validation, and test sets) to ensure accuracy. Below are the accuracies obtained from each approach.

- **Utilization of Cross-Validation Strategy: -**

Evaluation Metrics:

	Model	RMSE (CV)	MAE	MSE	RMSE	R^2
0	Linear Regression	9.340663	7.46543	87.09821	9.33264	0.56477
1	Random Forest	6.061253	1.70084	5.07704	2.25323	0.97463
2	XGBoost	6.357820	3.81130	23.83185	4.88179	0.88091
3	Decision Tree	8.382184	0.00000	0.00000	0.00000	1.00000
4	Lasso	10.014476	7.55409	100.24794	10.01239	0.49906
5	Gradient Boosting	7.104601	5.57885	48.75815	6.98270	0.75636
6	LightGBM	6.207474	4.51449	32.88034	5.73414	0.83570
7	MLP	6.357351	4.79884	37.25716	6.10386	0.81383

Source:- my submitted python notebook

Based on the RMSE values obtained above, it appears that the RandomForest Regressor algorithm performed the best. Additionally, LightGBM, XGBoost, and MLP Regression algorithms also showed relatively good performance for model training.

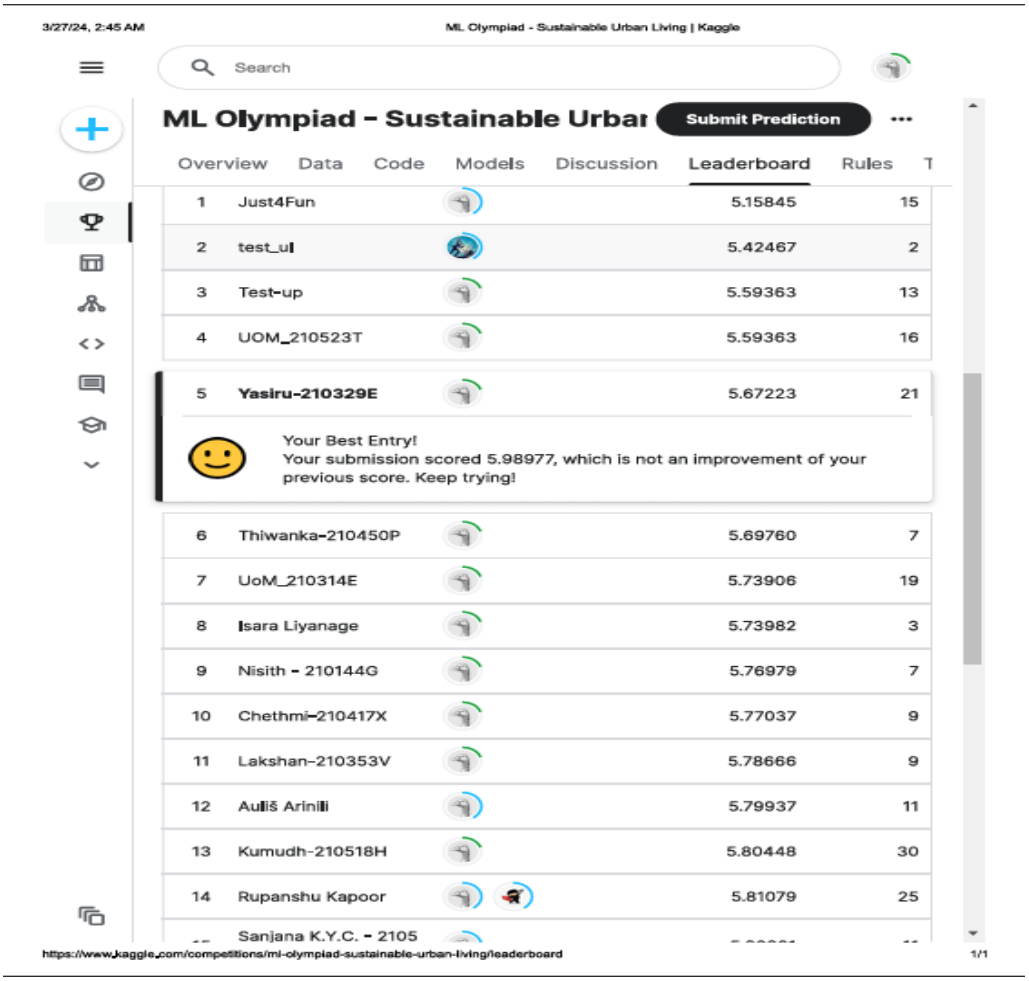
➢ **Other Approaches I listed earlier**

Here are the evaluation matrices obtained by predicting values for the validation dataset. The score is obtained through the competition predicting for the test data set.

Approach	RMSE	MAE	MSE	R^2	Score
2	5.681191910184582	4.399662436329721	32.27594152034674	0.839	5.96980
3	5.716432383055491	4.435198593193972	32.67759919004548	0.837	5.98977
4 - (Method 1)	5.675502770004128	4.400785321840155	32.21133169232453	0.839	5.94858
4 – (Method 2)	5.554273684867802	4.327133883492509	30.849956166414945	0.846	**5.67223**

Discussion

Overall, Approach 4 -Method 2 appears to be the most effective in terms of predictive accuracy, as it consistently outperforms other approaches across multiple evaluation metrics. However, Approach 4 (Method 1) also demonstrates competitive performance, suggesting that both ensemble methods (Stacking) utilized in Approach 4 are viable options for improving model accuracy.



Score & The Rank

Date: 3/27/2024

Time: 2.45 AM

Rank: 5

Score: 5.67223

Note: -
Please refer to the Python notebook provided for detailed information (feature engineering data reprocessing). The content has been structured in a formal manner to ensure clarity and organization.