**8. Combined Models and Regression Results**

Two stacking ensemble regressors were implemented using a meta-learning approach to enhance predictive performance and leverage the strengths of multiple regression models. Both pipelines used Ridge Regression as the meta-learner due to its regularization capabilities and strong performance in combining model outputs.

**8.1. Stacking: RF + GB + Linear Regression**

The base models used in the stacked regressor were Random Forest, Gradient Boosting, and Linear Regression. The ensemble was constructed using a pipeline that applied preprocessing before passing the data to the stacking model. These models were chosen based on their strong individual performances during baseline testing. However, the final ensemble produced only moderate results.

The performance metrics for the ensemble were as follows: RMSE of 0.443, MAE of 0.214, R² of 0.758, and a regression accuracy of 86.8%. While the model explains approximately 76% of the variance, including Linear Regression may have constrained the model's ability to capture the non-linear patterns in the data. Despite Ridge helping to manage overfitting, this limitation likely impacted the overall model performance.

**8.2. Stacking: RF + GB + KNN**

The linear model was replaced with K-Nearest Neighbors (KNN) to improve this, introducing instance-based learning and greater model diversity. KNN is non-parametric and excels at modeling local relationships, which complemented the tree-based learners well.

This revised ensemble significantly improved performance, achieving an RMSE of 0.101, MAE of 0.073, R² of 0.977, and a regression accuracy of 96.6%. The model explained nearly 98% of the variance and outperformed all other regressors, indicating that the fusion of tree-based and distance-based models enabled the stacking ensemble to generalize better on unseen data.

**8.3. Regression Pipeline Architecture**

The regression stacking pipeline was structured to follow a layered architecture comprising a preprocessing step, a set of base learners, and a meta-learner. In the preprocessing stage, the input data was cleaned and transformed to ensure it was suitable for modeling. Following this, three diverse base learners—Random Forest (RF), Gradient Boosting (GB), and K-Nearest Neighbors (KNN)—were employed to capture various patterns within the data. Each algorithm brought a unique perspective: Random Forest captured broad, global trends; Gradient Boosting focused on learning from residual errors through an iterative process; and KNN leveraged local neighborhood information. The predictions from these base models were then passed to a meta-learner, specifically Ridge Regression, which integrated the outputs to produce the final prediction. This ensemble approach enhanced overall model performance by combining the strengths of different learning algorithms.

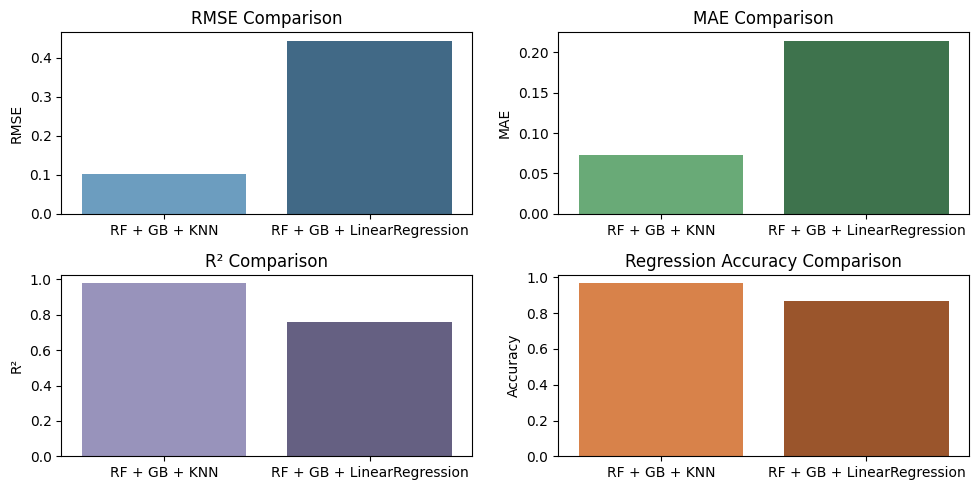
**9. Combined Classifier and Classification Results**

A stacking classifier system integrating Logistic Regression with XGBoost and LightGBM models was used for categorical school performance tier predictions. A model combination was created to manage linear prediction systems and non-linear modeling techniques. LightGBM did not produce the optimal single model results as a component of the ensemble model, but it expanded the ensemble's performance range through model diversity. A Logistic Regression was selected as the meta-learner for the stacking classifier due to its capable interpretation and stability. The model demonstrated 93.7% test accuracy and a 91.9% Macro F1 score, which showed that it could be adequately generalized with high accuracy and balanced predictions between different classes. Such strong performance across all categories becomes essential because the dataset has an existing issue with class imbalance.

**10. Model Comparison/Selection**

**10.1. Regression Comparison**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model Combination | RMSE | MAE | R2 | Regression Accuracy |
| RF + GB + KNN | 0.101 | 0.073 | 0.977 | 0.966 |
| RF+ GB + Linear Regression | 0.443 | 0.214 | 0.758 | 0.868 |



The bar plots indicate that the stacking ensemble consisting of RF + GB + KNN produced both the lowest RMSE and MAE results in addition to acquiring the highest R² and accuracy levels. Including KNN in the ensemble results in superior performance due to its ability to detect intricate local relations.

**10.2. Classification**

|  |  |  |
| --- | --- | --- |
| Model Combination | Accuracy | Macro F1 |
| Stacked Classifier | 0.937 | 0.919 |

The classification stacking model performs very well, most especially the macro F1 score, showing a good balance in classifying not only the majority performance tier but evenly also the minority performance tier accurately.

**11. Final Model Selection**

The Random Forest (RF), Gradient Boosting (GB), and k-Nearest Neighbors (KNN) stacked model is the top-performing ensemble for regression tasks. Combining the strengths of each separate model to achieve high R² values and low error metrics results in a very effective method to extract complex relationships from data. RF's robustness to overfitting, GB's minimization of residuals, and KNN's simplicity and adaptability make the model fairly generalized on any given regression problem.

In classification tasks, the stacked ensemble of Logistic Regression, XGBoost, and LightGBM is chosen from various ensemble techniques for being highly accurate and having balanced F1 scores for different classes. The ensemble method exploits the interpretability of Logistic Regression, the ability of XGBoost to cope with very big data with copious quantities of the features, and the efficiency of LightGBM to work with big data. Combining these models ensures high accuracy and appropriate balance in handling imbalanced datasets.

Combining the strengths of diverse learning algorithms in both ensemble approaches is successful, and the resulting models are more robust and accurate than any individual model can achieve alone. Such a strategy allows the models to capture the underlying patterns in the data better and, therefore, be highly reliable for real-world applications.