Models

Ranking

'*' = bad

'*****' = excellent

Linear regression *

Mean Squared Error: 13.812306613799512 R-squared: 0.15789654711745998

Cubic regression *

Polynomial Regression Model (Degree 3) Mean Squared Error: 12.10635972000329 R-squared: 0.23955141537046015

Random Forests ***

Cross-validated MSE for 'reborde': 1.160508174787941 Cross-validated MSE for 'inframamaria': 28.070928842629506

Feature importances for 'reborde': volumen_pre: 0.6132 proyeccion: 0.3868

Feature importances for 'inframamaria': volumen_pre: 0.5358 proyeccion: 0.4642

Reborde - Final Mean Squared Error: 0.6035516536688527 Reborde - Final R2 Score: 0.5738156195319692

Inframamaria - Final Mean Squared Error: 17.60542776317 Inframamaria - Final R2 Score: 0.3667867031451294

- Hyperparameter Tuning Random Forest has several hyperparameters you can tune to improve its performance. Key parameters include:

    - n_estimators: This is the number of trees in the forest. More trees generally improve performance, but it increases computational cost.
    - max_depth: This controls how deep each tree can grow. Limiting the depth can prevent overfitting.
    - min_samples_split: The minimum number of samples required to split an internal node. Higher values can reduce overfitting.
    - min_samples_leaf: The minimum number of samples required to be at a leaf node. Increasing this can make the model more conservative.
    - max_features: The number of features to consider when looking for the best split. Adjusting this can impact how well the model generalizes.

    Explanation of Hyperparameters: n_estimators: A higher number of trees (e.g., 500) typically improves accuracy but increases computation time. max_depth: Controlling the depth of the trees helps prevent overfitting (e.g., max_depth=10). min_samples_split and min_samples_leaf: Increasing these values prevents the model from learning too specific rules that may not generalize well. max_features: Controls the number of features considered at each split. auto (default) uses all features, sqrt and log2 are more conservative.

- Feature Importance Analysis Random Forest provides a way to measure the importance of each feature. You can use this to see if some features contribute more than others, and possibly drop less important features to simplify the model If some features have very low importance, you can consider removing them from the model to reduce overfitting and improve generalization.

XGBoosts ***

Cross-validated MSE for 'reborde': 1.1483956576133607 Cross-validated MSE for 'inframamaria': 27.322693419636515

Feature importances for 'reborde': volumen_pre: 0.6089 proyeccion: 0.3911

Feature importances for 'inframamaria': volumen_pre: 0.5660 proyeccion: 0.4340

Reborde - Final Mean Squared Error: 0.8105323822328852 Reborde - Final R² Score: 0.4276608487916962

Inframamaria - Final Mean Squared Error: 18.64400804984342 Inframamaria - Final R² Score: 0.3294321522520962

K-Nearest Neighbors (KNN)

Best parameters for 'reborde': {'algorithm': 'ball_tree', 'n_neighbors': 10, 'weights': 'uniform'} Best parameters for 'inframamaria': {'algorithm': 'brute', 'n_neighbors': 10, 'weights': 'uniform'}

Cross-validated MSE for 'reborde': 1.149363173518897 Cross-validated MSE for 'inframamaria': 28.605657737487228

Reborde - Final Mean Squared Error: 0.8738973922902493 Reborde - Final R² Score: 0.3829170768369844

Inframamaria - Final Mean Squared Error: 22.920611564625855 Inframamaria - Final R² Score: 0.17561582655045282

Interpretation

Mean Squared Error (MSE)

The MSE represents the average squared difference between the actual values and the model's predictions. A **lower** MSE means the model is making more accurate predictions.

An MSE value of 0.1864, which is very low, indicates that the model's predictions are quite close to the actual values. An MSE value of 4.96, which, while not that high, is still relatively low and indicates that the model is performing well but not that much.

R-squared (R²)

The R^2 value indicates how well the model explains the variability in the dependent variables. An R^2 value of 1 means perfect prediction, while a value of 0 means the model explains none of the variability. For a value of 0.8684, means that the model explains 86.8% of the variability in a variable, which is excellent.