**Supplementary Table 8** XGBoost hyperparameter configuration and description

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value Used** | **Description & Rationale** |
| learning\_rate | 0.05 | Controls the step size at each boosting iteration. A smaller value makes the model more robust to overfitting but requires more iterations. This value was chosen as a balance between training efficiency and model performance. |
| max\_depth | 5 | The maximum depth of a tree. This controls the complexity of the model by limiting the number of splits. A value of 5 was selected to capture non-linear relationships without allowing the trees to become overly complex and fit the noise in the training data. |
| gamma | 0 | The minimum loss reduction required to make a further partition on a leaf node. A value of 0 means no constraint, allowing the model to grow trees freely based on other parameters like max\_depth. |
| subsample | 0.8 | The fraction of instances (rows) to be randomly sampled for building each tree. Using a value less than 1.0 introduces randomness and helps prevent overfitting. |
| colsample\_bytree | 0.8 | The fraction of features (columns) to be randomly sampled for building each tree. Like subsample, this increases model robustness by ensuring trees are built on different feature subsets. |
| min\_child\_weight | 1 | The minimum sum of instance weight (hessian) needed in a child node. In simple terms, it controls the minimum number of data points required in a leaf node. A small value (like 1) allows the model to create leaves with fewer instances, providing flexibility while max\_depth and regularization help control overfitting. |
| reg\_lambda (L2 regularization) | 0 | The L2 regularization term on weights. A value of 0 means no L2 regularization was applied in the final model, as the combination of other parameters (like max\_depth and subsample) was sufficient to prevent overfitting. |