**Model Optimization and Tuning Phase Template**

| Date | July 2024 |
| --- | --- |
| Team ID | 739678 |
| Project Title | Zombie detection using the machine learing |
| Maximum Marks | 10 Marks |

**Model Optimization and Tuning Phase**

The Model Optimization and Tuning Phase involves refining neural network models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

### Hyperparameter Tuning Documentation (8 Marks):

| **Model** | **Tuned Hyperparameters** |
| --- | --- |
| Logistic Regression | #importing the library for grid search  from sklearn.model\_selection import GridSearchCV  The ‘lr\_param\_grid’ specifies different values for regularization strength (C), solvers (solver), and penalty types (penalty). GridSearchCV (lr\_cv) is employed with 5-fold cross-validation (cv=5), evaluating model performance based on accuracy (scoring="accuracy"). The process uses all available CPU cores (n\_jobs=-1) for parallel processing and provides verbose output (verbose=True) to track progress. |
| Random Forest | The parameter grid (rfc\_param\_grid) for hyperparameter tuning. It specifies different values for the number of trees (n\_estimators), splitting criterion (criterion), maximum depth of trees (max\_depth), and maximum number of features considered for splitting (max\_features). GridSearchCV (rfc\_cv) is employed with 3-fold cross-validation (cv=3), evaluating model performance based on accuracy (scoring="accuracy"). |
| XGBoost | The (params) define a grid for hyperparameter tuning of the XGBoost Classifier (XGBClassifier), including min\_child\_weight, gamma, colsample\_bytree, and max\_depth. The XGBClassifier is configured with a learning rate of 0.5, 100 estimators, using a binary logistic regression objective, and utilizing 3 threads for processing. GridSearchCV (xg\_cv) is used with 5-fold cross-validation (cv=5), refitting the best model (refit=True), evaluating based on accuracy (scoring="accuracy") |
| Decision Tree | The parameters (params) define a grid for hyperparameter tuning of the Decision Tree Classifier (DecisionTreeClassifier), including max\_depth, min\_samples\_leaf, and criterion ('gini' or 'entropy'). GridSearchCV (dec\_cv) is used with 5-fold cross-validation (cv=5), evaluating model performance based on accuracy (scoring="accuracy") |
| Ridge Classifier | The parameters (params) define a grid for hyperparameter tuning of the Decision Tree Classifier (DecisionTreeClassifier), including max\_depth, min\_samples\_leaf, and criterion ('gini' or 'entropy'). GridSearchCV (dec\_cv) is used with 5-fold cross-validation (cv=5), evaluating model performance based on accuracy (scoring="accuracy") |
| K- Nearest Neighbors | The parameters (params) define a grid for hyperparameter tuning of the K-Nearest Neighbors Classifier (KNeighborsClassifier), including n\_neighbors, weights ('uniform' or 'distance'), and metric ('minkowski', 'euclidean', or 'manhattan'). GridSearchCV (knn\_cv) is used with 5-fold cross-validation (cv=5), evaluating model performance based on accuracy (scoring="accuracy") |

### 

### 

### 

### 

### Final Model Selection Justification (2 Marks):

| **Final Model** | **Reasoning** |
| --- | --- |
| **Random Forest** | Random Forest model is chosen for its robustness in handling complex datasets and its ability to mitigate overfitting while providing high predictive accuracy.    Above all the models Random Forest model have the highest accuracy among all the models. |