Regression Project

1- Read the data file called regression_project_data.csv
2- Explore the Dataset
3- Data Visualization:
    - Distribution of Player Ages
    - Nationality Count Plot
    - Overall Rating Distribution
    - Potential vs. Value Scatter Plot
    - Write your comment about all plots
4- Data Preprocessing:
    - Drop the Columns: ['full_name', 'birth_date', 'nationality', 'value_euro', 'wage_euro', 'preferred_foot', 'release_clause_euro', 'national_team', 'national_rating', 'national_team_position', 'national_jersey_number']
    - Encode the Categorical Columns
    - Split the Data into Features and Label, your target is “overall_rating” Column
    - Split the Dataset using train_test_split
5- Apply All the regression models such as (linear regression, knn, svr, random forest, decision tree, adaboost, xgboost)
6- Evaluate all the models in train and test to check for the best model (champion model) using r2_score, mean_absolute_error, mean_squared_error, root_mean_squared_error
7- **Bonus**:
    - Calculate the residuals for the test set
    - Get the indices of the top 10 most off predictions in the test set
    - Print the top 10 most off predictions in the test set and their corresponding names
    - Calculate the residuals for the training set
    - Get the indices of the top 10 most off predictions in the training set
    - Print the top 10 most off predictions in the training set and their corresponding names

Data Description:

- name: Name of the player.
- full_name: Full name of the player.
- birth_date: Date of birth of the player.
- age: Age of the player.
- height_cm: Player's height in centimeters.
- weight_kgs: Player's weight in kilograms.
- positions: Positions the player can play.
- nationality: Player's nationality.
- overall_rating: Overall rating of the player in FIFA.
- potential: Potential rating of the player in FIFA.
- value_euro: Market value of the player in euros.
- wage_euro: Weekly wage of the player in euros.
- preferred_foot: Player's preferred foot.
- international_reputation(1-5): International reputation rating from 1 to 5.
- weak_foot(1-5): Rating of the player's weaker foot from 1 to 5.
- skill_moves(1-5): Skill moves rating from 1 to 5.
- body_type: Player's body type.
- release_clause_euro: Release clause of the player in euros.
- national_team: National team of the player.
- national_rating: Rating in the national team.
- national_team_position: Position in the national team.
- national_jersey_number: Jersey number in the national team.
- crossing: Rating for crossing ability.
- finishing: Rating for finishing ability.
- heading_accuracy: Rating for heading accuracy.
- short_passing: Rating for short passing ability.
- volleys: Rating for volleys.
- dribbling: Rating for dribbling.
- curve: Rating for curve shots.
- freekick_accuracy: Rating for free kick accuracy.
- long_passing: Rating for long passing.
- ball_control: Rating for ball control.
- acceleration: Rating for acceleration.
- sprint_speed: Rating for sprint speed.
- agility: Rating for agility.
- reactions: Rating for reactions.

- balance: Rating for balance.
- shot_power: Rating for shot power.
- jumping: Rating for jumping.
- stamina: Rating for stamina.
- strength: Rating for strength.
- long_shots: Rating for long shots.
- aggression: Rating for aggression.
- interceptions: Rating for interceptions.
- positioning: Rating for positioning.
- vision: Rating for vision.
- penalties: Rating for penalties.
- composure: Rating for composure.
- marking: Rating for marking.
- standing_tackle: Rating for standing tackle.
- sliding_tackle: Rating for sliding tackle.

Classification Project

Dataset:

Use Pen-Digits datasets (train dataset & test dataset) with provided splits to solve

Questions:

Decision tree

1- Generate a scatterplot matrix to show the relationships between the variables and a heatmap to determine correlated attributes, then write a summary of what you noticed.

2- Ensure data is in the correct format for downstream processes (e.g., remove redundant information, convert categorical to numerical values, address missing values, etc.)

3- Fit a **decision tree** to the training data. Plot the tree, and display accuracy and Confusion Matrix.

4- Try different ways to improve the decision tree algorithm (e.g., use different splitting strategies, prune tree after splitting). Does pruning the tree improves the accuracy?

Bagging

(Bagging is to generate a set of bootstrap datasets, create estimators for each bootstrap dataset, and finally utilize majority voting (soft or hard) to get the final decision.)

1- Apply bagging strategy to classify test set samples by using **SVM** algorithm as base estimator. Display accuracy and Confusion Matrix.

2- Apply **Random Forest algorithm** (the baseline), then fine tune this baseline. For the number of estimators, Try 5 different values within the interval of [10, 200]. Plot accuracy vs. number of estimators.

Boosting

1- Use **AdaBoost** classifier. There are 2 important hyperparameters in **AdaBoost**, (the number of estimators, and learning rate). First, tune number of estimators parameter by trying 4 values in the interval of [10,

200]. Then by using the tuned value for number of estmators, tune the learning rate parameter by trying 4 values within the range of [0.1, 0.9]. Display accuracy and Confusion Matrix separately for the best value of both parameters (Number of estimators and learning rate).

2- Build **XGBoost** classifier with the same parameters that you obtained in the last one.

Provide accuracy and Confusion Matrix.

3- Comment on Bagging and Boosting approaches.