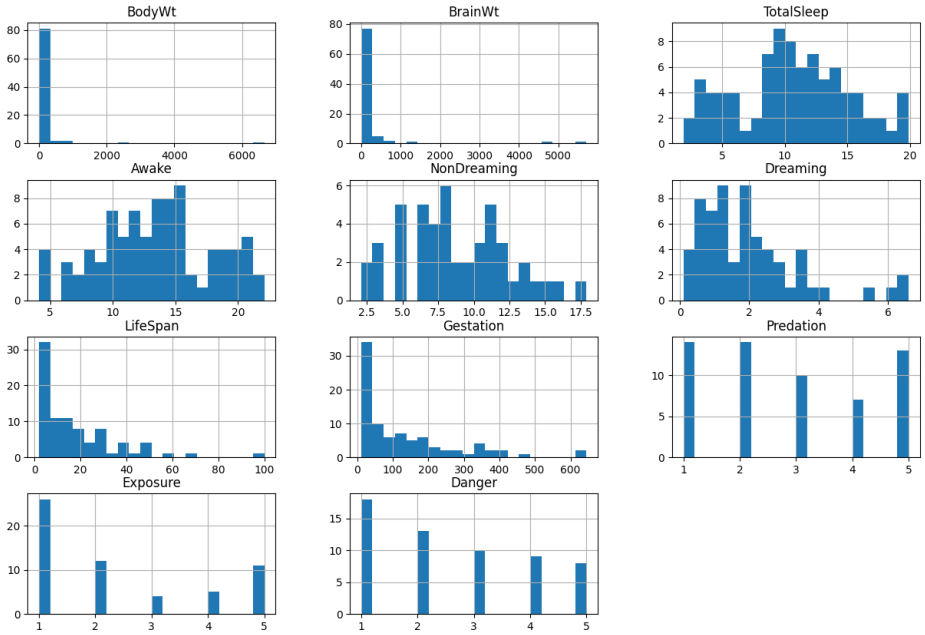
Project Report: Predicting Total Sleep and Dreaming Duration Using Regression Models

This report details the development and evaluation of regression models aimed at predicting two distinct variables: "TotalSleep" and "Dreaming" duration, extracted from the sleep dataset loaded from "sleep_dataset.xlsx". The project adheres to the outlined requirements, encompassing data analysis, feature selection, model training, evaluation, and reporting, ensuring reproducibility.

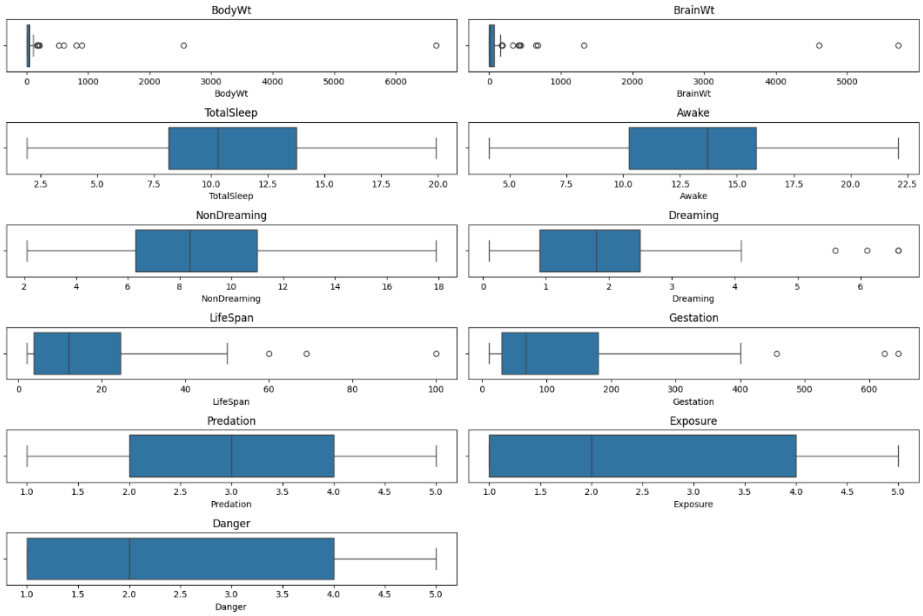
Data Analysis and Exploration:

The initial phase involved a thorough examination of the data using various techniques implemented within the code.

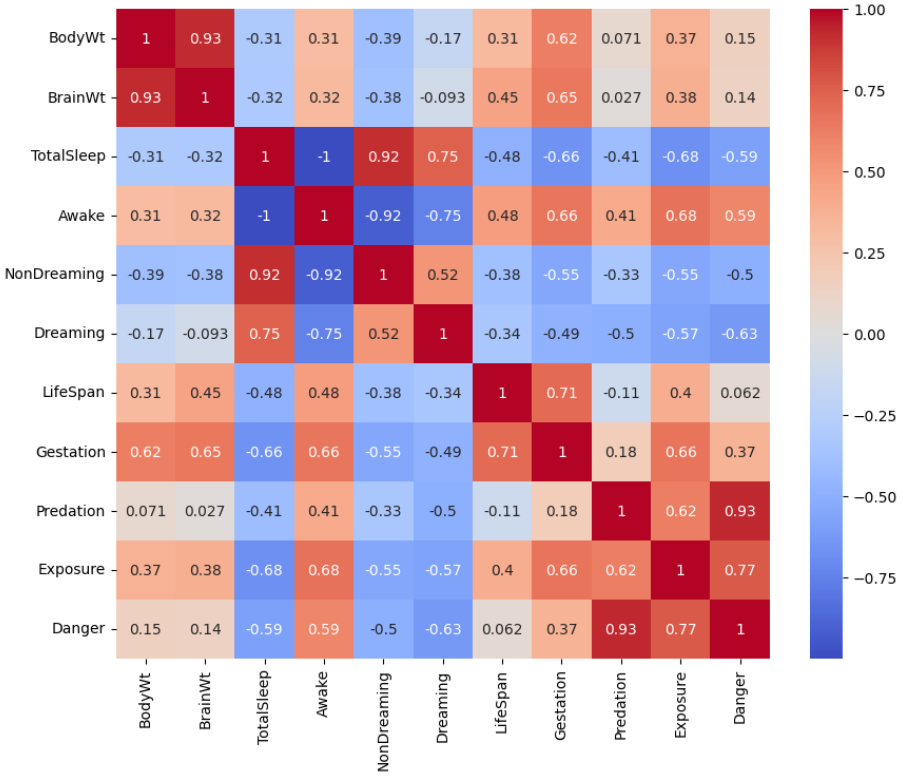
- **Data processing:** The pandas library was used to load the data from the Excel file (pd.read_excel).
- **Data cleaning:** Basic information about the data, including data types and missing values, was obtained using df.info().
- **Exploratory analysis:**
    - Histograms were generated using matplotlib.pyplot (plt.hist) to visualize the distributions of numerical attributes, revealing potential skewness or outliers.



    - Boxplots were constructed using seaborn (sns.boxplot) to identify individual outliers deviating significantly from the central tendency within each numerical column.



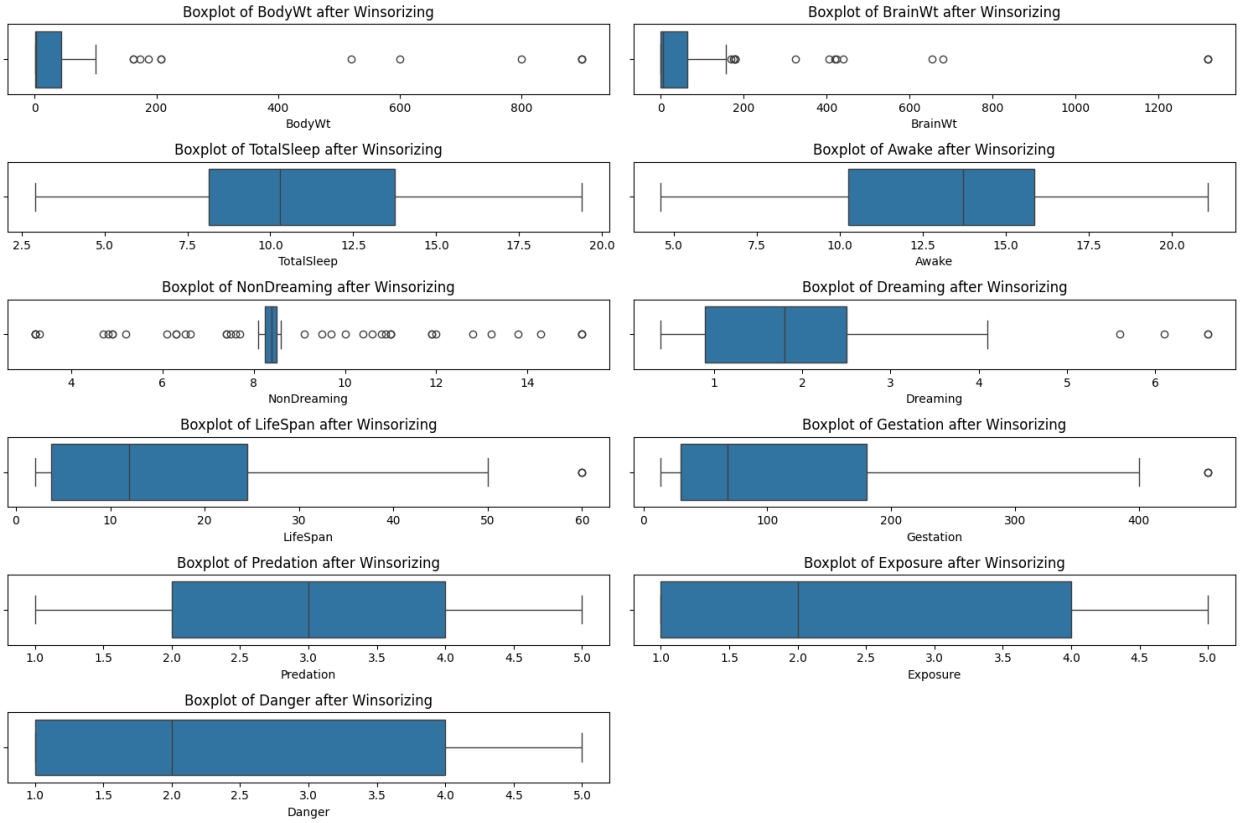
- A correlation heatmap was created using seaborn (sns.heatmap) to assess the relationships between numerical attributes, highlighting potential dependencies.



Addressing Data Issues and Feature Engineering:

Missing values present in the "NonDreaming" column were addressed using a median imputation strategy implemented through SimpleImputer from sklearn.impute. To enrich the dataset and

potentially improve model performance, a new feature, "BrainBodyRatio," was created by calculating the ratio between "BrainWt" and "BodyWt" using basic arithmetic operations within the code. Furthermore, winsorization, a technique for reducing the impact of outliers, was applied to identified outliers in numerical columns using scipy.stats.mstats.winsorize.



Feature Selection and Preprocessing:

The dataset was then divided into features and target variables. Numerical features were separated from categorical features using pandas functionalities. Subsequently, pipelines were established for both numerical and categorical feature transformations.

- **Numerical feature preprocessing:**
    - A pipeline was created using sklearn.pipeline.Pipeline to perform consistent transformations.
    - The pipeline included imputation with the median strategy using SimpleImputer and standardization using StandardScaler from sklearn.preprocessing.
- **Categorical feature preprocessing:**
    - Another pipeline was created using sklearn.pipeline.Pipeline.

- The pipeline employed imputation with the most frequent category using SimpleImputer and one-hot encoding using OneHotEncoder from sklearn.preprocessing to handle categorical features effectively.

Model Training and Evaluation:

The dataset was strategically split into training (80%) and testing (20%) sets using train_test_split from sklearn.model_selection. This split prevents overfitting and ensures unbiased evaluation.

Initial Linear Regression Models:

Initially, linear regression was chosen as the primary model for both "TotalSleep" and "Dreaming" prediction, implemented using LinearRegression from sklearn.linear_model. Separate pipelines, containing feature preprocessing and the model, were created using make_pipeline from sklearn.pipeline. Their performance scores were:

- **TotalSleep Model MSE:** 0.6685
- **Dreaming Model MSE:** 0.2537

Exploring XGBoost for Improved Performance:

To potentially enhance performance, XGBoost, a powerful tree-based ensemble algorithm, was introduced. Separate XGBoost models were trained for both "TotalSleep" and "Dreaming" using the same pipelines as before, replacing the linear regression model with XGBRegressor from the xgboost library. The results were as follows:

- **XGBoost TotalSleep Model MSE:** 0.1338
- **XGBoost Dreaming Model MSE:** 0.4003

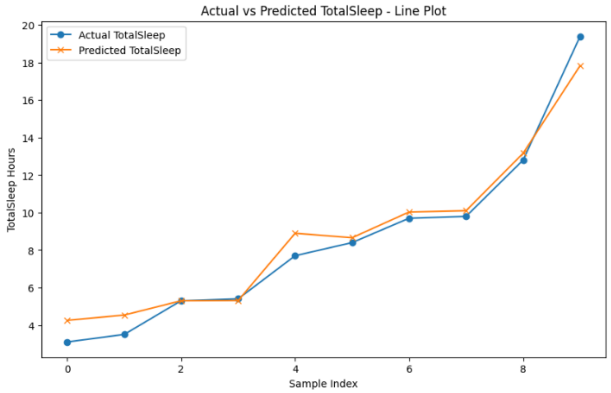
Key Observations:

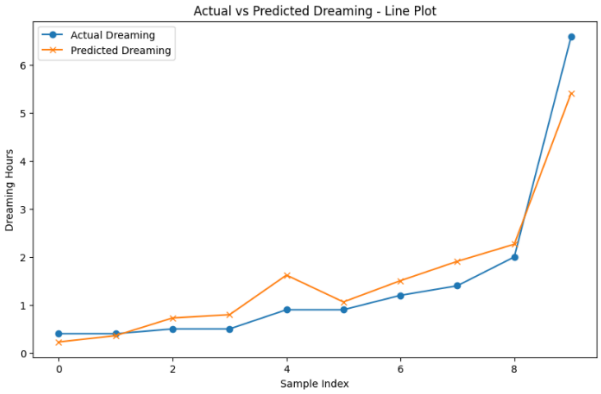
- XGBoost significantly outperformed linear regression in predicting "TotalSleep," suggesting a better ability to capture non-linear relationships within the data.
- For "Dreaming," linear regression outperformed XGBoost, indicating potential overfitting of XGBoost to the training data.

Visualizing Predictions:

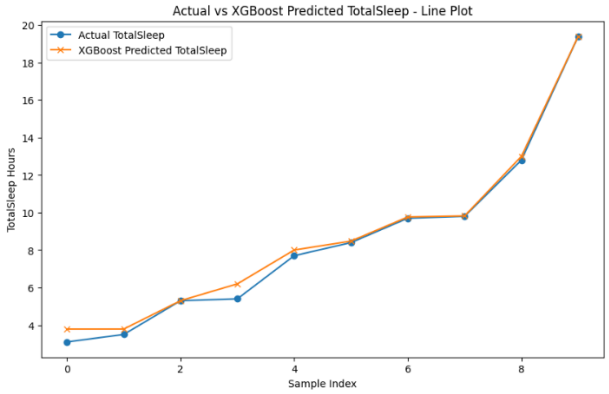
Line plots were created to visualize actual vs. predicted values for both linear regression and XGBoost model, allowing for a comparative assessment of their performance.

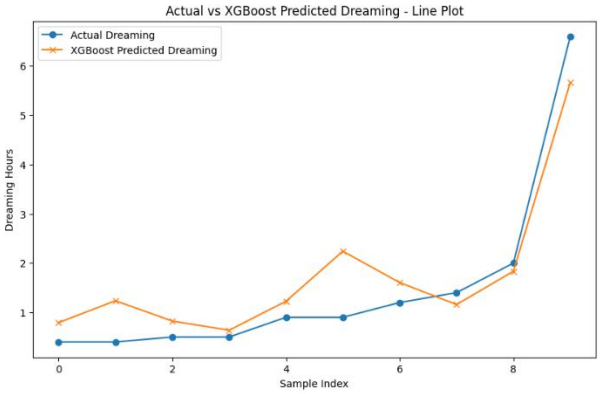
Linear Regression:





XGBoost Model:





These visualizations allow for a visual comparison of the model's predictions against the actual values in the testing set.

Conclusion and Future Directions:

This project successfully established separate regression models for predicting "TotalSleep" and "Dreaming" duration. While the models demonstrate promise, there is further room for improvement. Techniques like hyperparameter tuning and feature selection can be explored to potentially enhance model performance.