Question 1

Introduction

The Palmer Penguins dataset offers a rich resource for studying the characteristics of three penguin species: Adelie, Chinstrap, and Gentoo, in the Palmer Archipelago, Antarctica. It includes morphological measurements like bill length, bill depth, flipper length, and body mass, along with categorical information such as species, island, sex, and collection year. Our exploration aims to uncover patterns and relationships within the dataset using classification and regression analyses

Data pre-processing & Exploratory data analysis

The code *data.isnull().sum()* gives us a list of the columns in the dataset and tells us the number of null values in each column. After discovering the rows and columns with null values, I decided to replace the missing values.

By imputing missing values in numerical features with the ”median”, I ensured that the imputed values are representative of the central tendency of the data while minimizing the impact of outliers or skewed distributions. This helps maintain the statistical properties of the dataset and ensures that imputed values align with the overall distribution of the feature. Whereas, imputing missing values in categorical features with the ”mode” ensures that the imputed values are consistent with the majority of observations in the dataset. This approach helps maintain the representativeness of the categorical features and ensures that imputed values align with the prevailing categories observed in the dataset.

A positive correlation exists between bill length and bill depth across all penguin species “Fig. 1”. Adelie penguins tend to have the shortest bills “Fig. 2” and flippers, while Gentoo penguins have the longest bills and flippers. Although Torgersen and Biscoe islands have similar flipper lengths, penguins from Biscoe have longer bills “Fig. 2”.

Approaches

Regression model

Linear regression Linear Regression models the relationship between a dependent variable $\boldsymbol{y}$ and one or more independent variables $\boldsymbol{X}$ by fitting a linear equation into the observed data. The model aims to minimize the sum of squared residuals between the observed and predicted values.

Equation: The linear regression model is represented as:

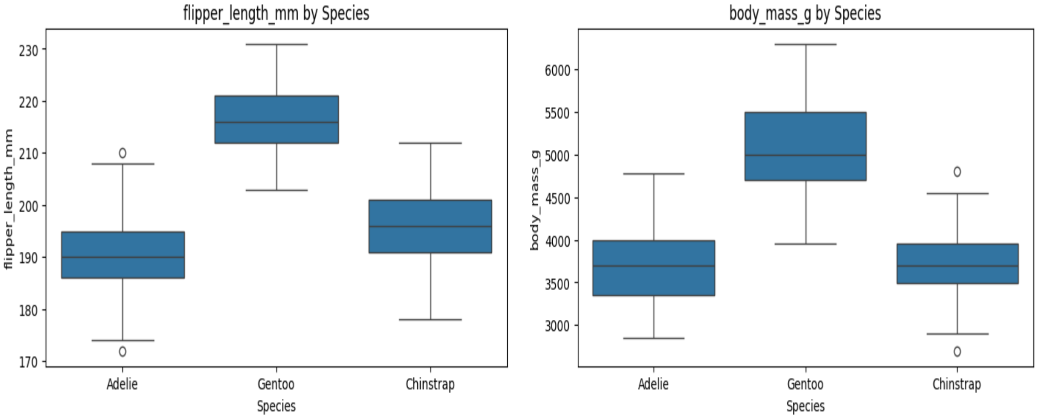

Figure 1: A positive correlation exists between the bill length and bill depth across all three species. This suggests that penguins with longer bills tend to have deeper bills as well.

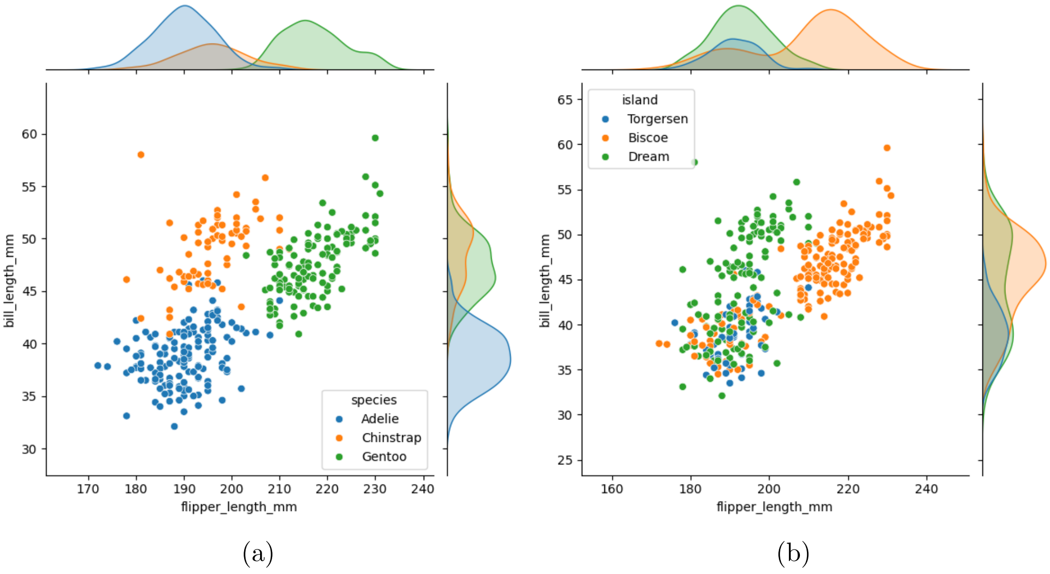

Figure 2: In fig(a): shows the relationship between bill length and flipper length of three penguin species: Adelie, Chinstrap, and Gentoo. , fig(b): Shows the relationship between bill length and flipper length of three islands.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 ..... + \beta_n x_n + \epsilon \tag{1}$$

where β_0 is the intercept, β_i are the coefficients, x_i are the features, and ϵ is the error term.

Classification models

K-Nearest Neighbors (KNN): K-Nearest Neighbors is a non-parametric, lazy learning algorithm used for classification and regression tasks. For classification, the algorithm assigns a class label to a new data point based on the majority class among its k nearest neighbors.

For a given data point $\boldsymbol{x}$, the predicted class Υ is determined by:

$$\Upsilon = model(\Upsilon_i) \tag{2}$$

where $i \ \epsilon$ nearest neighbors of x

The KNN classifier achieved an accuracy of 73.08%. Class 0 (Adelie) has the highest precision and recall, indicating that the model performs well in predicting this class. However, Class 1 (Chinstrap) has lower precision and recall, suggesting that the model struggles to accurately predict the classes.

Decision Trees: Decision Trees partition the feature space into regions and make predictions by traversing the tree from the root to a leaf node. Each internal node tests a feature, and the branches represent the outcome of the test. Leaf nodes represent class labels.

The decision rules at each node are represented as:

$$Node : feature \leq threshold \tag{3}$$

The Decision Tree classifier demonstrated high accuracy of 97.12%. It performed exceptionally well across all classes with high precision, recall, and F1-scores, indicating robust performance in classifying the penguin species.

Table 1: Comparison of KNN and Decision Tree Classification Metrics

Algorithm	KNN			Decision Tree		
	Class 0	Class 1	Class 2	Class 0	Class 1	Class 2
Precision	64%	80%	93%	94%	100%	100%
Recall	96%	20%	74%	100%	90%	97%
F1-score	77%	32%	82%	97%	95%	99%

Choice of Algorithms:

K-Nearest Neighbors (KNN) is a versatile classification algorithm suitable for tasks with complex decision boundaries and where data distribution assumptions are not strict. It excels in capturing local data neighborhoods and is effective for datasets with non-linearly separable classes and ample data for accurate nearest-neighbor identification. On the other hand, Decision Trees offer interpretability and insight into feature importance, handling both numerical and categorical data effectively. They model non-linear relationships and interactions between features, making them beneficial for datasets with mixed feature types and suitable for binary and multi-class classification tasks.

Choice of evaluation for Algorithms

K-Nearest Neighbors (KNN) and Decision Trees are evaluated using metrics such as accuracy, precision, recall, and F1-score. These metrics offer insights into the classifier's performance, particularly in handling imbalanced datasets and different misclassification costs. Additionally, the confusion matrix provides a detailed breakdown of the classifier's performance.

While accuracy measures overall correctness, precision, recall, and F1-score provide a balanced assessment of the classifier's performance. Decision Trees, like KNN, can use these metrics to evaluate performance and are built based on criteria such as information gain or Gini impurity to quantify split effectiveness and assess tree quality. By selecting these algorithms and evaluation metrics, we aim to conduct a comprehensive assessment of classification models on the Palmer Penguins dataset.

Comparison with baseline models

Comparing baseline models with our selected classification algorithms like most frequent and stratified baseline models, we observe significant differences in performance metrics. The most frequent baseline models exhibit a lower accuracy of 44.23% and the stratified base model hits 33.65% and F1-scores across all classes, indicating poor predictive capability and an inability to generalize effectively. In contrast, our selected models—KNN, Decision Tree, and Random Forest—demonstrate improved performance and a more balanced prediction across classes, suggesting that these advanced algorithms and optimization techniques contribute to enhanced predictive accuracy and robustness.

Description of meta-parameter

The optimization of the K-Nearest Neighbors (KNN) algorithm involved experimenting with various values of the number of neighbors (k), ranging from 3 to 15. Cross-validation was employed to identify the optimal k value that resulted in the best performance. Additionally, we evaluated both Euclidean and Manhattan distances as distance metrics to determine which metric yielded better classification results. Cross-validation played a crucial role in selecting the distance metric that improved the model's accuracy and robustness.

For the Decision Tree algorithm, we assessed the performance using both Gini Impurity and Entropy as splitting criteria. Grid search was utilized to identify the criterion that enhanced the model's predictive capability. Different values for the maximum depth were tested to prevent overfitting, with cross-validation aiding in finding the optimal depth that balanced model complexity and performance. Similarly, various minimum samples split values were experimented with to control the tree's growth and avoid overfitting. Cross-validation helped in selecting the value that optimized the model's generalization ability.

Conmparison and conclusion

In our analysis of the Palmer Penguins dataset, we employed two classification algorithms:

K-Nearest Neighbors (KNN) Classification: KNN identifies 'k' closest training example to a given input in the feature space, making predictions based on the majority class among its neighbors. We experimented with varying 'k' values and distance metrics (Euclidean and Manhattan) for optimization. Ultimately, the KNN classifier achieved an accuracy of approximately 73.08%, showcasing its capability in penguin species classification.

Decision Tree Classification: Decision Trees construct a tree-like structure by recursively splitting the feature space based on maximum information gain or minimum impurity at each node. Criteria such as Gini Impurity and Entropy were considered for splitting, with the maximum tree depth adjusted to prevent overfitting. The Decision Tree classifier exhibited exceptional performance, achieving an accuracy of approximately 97.12In our analysis of the Palmer Penguins dataset, we utilized K-Nearest Neighbors (KNN) and Decision Tree classification algorithms, leveraging the capabilities of scikit-learn in Python. KNN achieved an accuracy of 73.08%, considering various 'k' values and distance metrics. Decision Tree outperformed

KNN, exhibiting exceptional accuracy (97.12%) due to its ability to capture complex relationships. The comparison shows Decision Tree's superior performance in terms of accuracy, precision, recall, and F1-scores. While KNN offers reasonable accuracy, Decision Tree's robust performance and interpretability make it a preferred choice, providing valuable insights into the dataset and penguin species based on provided features.

Question 2

One example that encompasses the challenges of data protection and bias amplification involves the use of facial recognition technology in law enforcement. Facial recognition algorithms, trained on historical datasets, have been found to exhibit biases against certain demographic groups, including people of color and women. This bias can lead to disproportionate targeting and surveillance of these groups by law enforcement agencies, infringing upon their privacy and civil liberties. To address these ethical challenges, transparency and accountability measures can be implemented. Firstly, rigorous evaluation and validation of facial recognition algorithms should be conducted to identify and mitigate biases. Additionally, clear guidelines and regulations should be established to govern the use of facial recognition technology in law enforcement, ensuring that it is deployed ethically and responsibly. Furthermore, individuals should have greater control over their biometric data, with mechanisms in place to obtain consent and enable opt-out options.

Another example pertains to the safety of AI systems and the potential for existential threats from machines. One notable concern is the development of autonomous weapons systems, including drones and robots, equipped with AI capabilities for decision-making and targeting. These systems raise ethical dilemmas regarding their potential for indiscriminate harm and the erosion of human control over military operations. To address these challenges, international regulations and treaties can be established to prohibit the development and deployment of autonomous weapons systems. Additionally, research efforts should prioritize the development of AI technologies that enhance human capabilities and promote safety and security, rather than replacing or superseding human judgment. Ethical frameworks and guidelines should be integrated into the design and deployment of AI systems to ensure that they adhere to principles of accountability, transparency, and human rights.