**Sheikh Hasin Ishrak – 202108209**

**Mahodi Hasan Sabab – 202107371**

**Mouadh ben abderrahim – 202004180**

**Aiman Saad Baig – 201708742**

**Zain**

**Documentation and Reporting**

**Project Summary**

**Objectives:** The primary objective of this project was to apply machine learning techniques to accurately classify instances from the mushroom dataset (binary classification). The motivations included exploring the effectiveness of various classification models in handling categorical and numerical data and optimizing these models to achieve the highest possible accuracy.

NOTE: The dataset that we used is directly from this website: https://archive.ics.uci.edu/dataset/848/secondary+mushroom+dataset

**Data Preprocessing**

1. **Data Loading:** The data was loaded using the **ucimlrepo** package, specifically targeting dataset ID 848.
2. **Initial Data Exploration:** Basic explorations such as **data.head()**, **data.info()**, and **data.describe()** were conducted to understand the dataset structure and summary statistics.
3. **Handling Missing Values:** Missing values were imputed using median for numerical columns and mode for categorical columns.
4. **Visualization of Data Distribution:** Histograms and count plots were generated to visualize the distribution of numerical and categorical variables respectively.
5. **Outlier Detection and Handling:** Outliers were detected using Z-scores and handled by capping values at the 1st and 99th percentiles.
6. **Encoding Categorical Variables:** Label encoding, and one-hot encoding were used to transform categorical variables into a format suitable for model ingestion.

**Model Training and Evaluation**

**Feature Selection**

To enhance the performance of the models and ensure they are not overwhelmed by irrelevant or less significant features, two main feature selection techniques were employed:

1. **Recursive Feature Elimination (RFE):** This technique involves iteratively constructing a model (like a decision tree or logistic regression) and removing the least significant features based on the model's coefficients or feature importances. RFE helps in pinpointing the most impactful features that contribute substantially to predicting the target variable.
2. **Mutual Information Classification:** This non-linear method measures the dependency between variables using entropy. A higher mutual information value between a feature and the target variable suggests a stronger relationship, thus aiding in selecting features that have the most predictive power concerning the target.

**Model Building**

Three different types of models were explored to handle the classification task, each chosen for their specific characteristics and strengths in handling different types of data and problem complexities:

1. **Decision Tree Classifier:** Known for its simplicity and interpretability, the Decision Tree builds a model in the form of a tree structure. It splits the dataset into branches to make predictions, making it easy to understand and visualize.
2. **Random Forest Classifier:** An ensemble method that builds multiple decision trees and merges them together to get a more accurate and stable prediction. Random Forests are less likely to overfit than a single decision tree.
3. **Neural Network:** A more complex model that can capture non-linear relationships between features and the target. It's particularly useful for large datasets with complex patterns but requires more computational resources and tuning.

**Model Evaluation**

To assess the efficacy and performance of the models, several metrics were calculated:

* **Accuracy:** Measures the overall correctness of the model, i.e., the ratio of correct predictions to total predictions.
* **Precision:** Indicates the ratio of true positive predictions to all positive predictions (including false positives), which is crucial in scenarios where the cost of a false positive is high.
* **Recall:** Measures the ability of a model to find all the relevant cases (true positives) within a dataset.
* **F1 Score:** Combines precision and recall into a single metric by taking their harmonic mean. It gives a better measure of incorrectly classified cases than the Accuracy Metric.

Additionally, confusion matrices were plotted for each model to visualize the performance in terms of true positives, true negatives, false positives, and false negatives.

**Model Optimization**

**Hyperparameter Tuning**

Optimal model performance often depends on the model parameters chosen. Two methods were used to fine-tune these parameters:

1. **Grid Search:** An exhaustive search method that tests every combination of the provided hyperparameter values to find the combination that performs best. It was applied to both the Decision Tree and Random Forest models to determine optimal settings such as tree depth, minimum samples split, and the number of estimators.
2. **Random Search:** A randomized version of grid search that tests a random selection of parameters. This method can be more efficient than grid search when dealing with a very high number of different parameters and large data, as it does not exhaustively search all combinations.

**Neural Network Tuning**

**Given the complexity and tendency of neural networks to overfit, specific techniques were employed to optimize their performance:**

* **L1 Regularization:** Adds a penalty equivalent to the absolute value of the magnitude of coefficients. This helps in feature selection as some of the coefficient values turn to zero effectively ignoring some features.
* **Dropout:** A technique used to prevent overfitting by randomly dropping units (both hidden and visible) during training, which helps to make the model robust by not relying on any single input.

By applying these techniques, the models were not only tuned to achieve higher performance but were also made more generalized, thus performing better on unseen data.

**Interesting Findings and Conclusions**

1. **High Accuracy Across Models:** All models achieved high accuracy, which may indicate either an effective model or an overly simplistic or biased data set.
2. **Feature Importance:** Analysis showed which features significantly impacted model predictions, guiding future data collection and feature engineering.
3. **Chi-square Tests:** Statistical tests provided insights into the independence of categorical features relative to the target variable.

**Justification of Choices**

1. **Model Selection Rationale:** Each model was chosen based on its strengths in handling the specific characteristics of the dataset—simplicity and interpretability for Decision Trees, accuracy and feature handling for Random Forests, and complex pattern recognition for Neural Networks.
2. **Preprocessing Decisions:** Imputation methods were selected based on the nature of missing data in each feature, while encoding methods were chosen to best represent the categorical data for model consumption.

**Reproducibility and Code Comments**

* The notebook contains well-commented code to ensure that each step of the machine learning process is understandable and reproducible.
* Visualizations are included directly within the notebook to provide insights at each step of the analysis.

**Conclusion**

This project demonstrates the effective application of various machine learning techniques to a real-world classification problem. The comprehensive approach taken from data preprocessing to model evaluation and optimization ensures robustness and reliability of the findings, paving the way for further research and application of these models in similar problems.

Top of Form