Comparing Bayesian Networks and Decision Tree Classifiers/Random Forests with Machine Learning Models on the Dinosaur Dataset

Name: Muhammad Hamza Zafar

Student ID: 22022247

Colab Code Link: https://colab.research.google.com/drive/1dkbiGDwN77bPat3Q5nKqQDpPHFAiCrNf?usp=sharing

Introduction:

This dataset contains detailed information about dinosaurs, including names, diet, life span, type, length, taxonomy, species, and information sources. The goal is to use two distinct methods Bayesian Networks and Decision Tree Classifiers/Random Forests to estimate the type of dinosaurs based on different criteria.

Data Mining Techniques:

1. Bayesian Networks:

Bayesian Networks are probabilistic graphical models that use Bayesian inference for reasoning under uncertainty. They represent variables and their dependencies via a directed acyclic graph (DAG). Variables are represented by nodes, and their probabilistic relationships are encoded by edges. According to the assumptions of Bayesian networks, every variable is Conditionally independent of its parents' non-descendants. They are applied in fields including fault identification, risk assessment, and medical diagnostics for probabilistic reasoning, prediction, categorization, and decision-making.

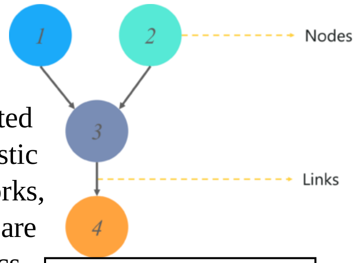

Fig. 1.0 Bayesian Network

2. Classifiers for Decision Trees and Random Forests

2.1. Decision Tree Classifiers:

Applied to both regression and classification applications, Decision Tree Classifiers are non-parametric supervised learning techniques. The dataset was recursively divided into subsets according to the most important feature at each stage. This procedure generates a structure like a tree, with leaf nodes representing class labels or regression values, and continues until a stopping requirement is satisfied. (Brownlee, n.d.) Every leaf node has a class name or a predicted value, and every inside node represents a test on an attribute. Simple to comprehend and analyze, offering perceptions into the process of making decisions. Must be pruned or limited in tree depth due to overfitting in complicated trees.

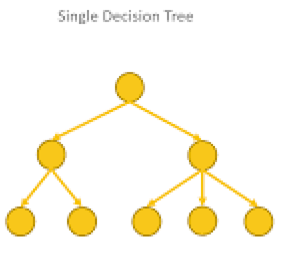

Fig. 2.0 Single Decision Tree

Random Forest

Fig. 2.1 Random Forest Demonstration

2.2. Random Forests:

Based on the idea of combining predictions from several decision trees, Random Forests are ensemble learning techniques. They use random feature selections and different dataset subsets to train an enormous number of decision trees. By averaging or taking the mode of the predictions made by each tree, the final prediction is obtained. (Liberman, n.d.) The ensemble method combines several decision trees to increase the robustness and accuracy of predictions. As opposed to separate decision trees, this helps to reduce overfitting. The ability to withstand data noise and outliers.

Data Preprocessing:

- **Data Loading:** To create a data frame, read the dataset using Pandas' read_csv() method.
- **First exploration:** To learn the structure and contents of the columns, using head() for first few rows
- **handling Missing Values:** Determine missing values with isnull().sum() and use dropna(inplace=True) to remove rows that contain NaN.
- **'Length' to Numeric Conversion:** Take off the' unit and change the values of 'length' to float type.
- **Encoding Labels for Categorical Variables:** Utilize scikit-learn's LabelEncoder to convert categorical columns into the numerical format.
- **Data Splitting:** Use the train_test_split() function from scikit-learn to split the dataset into training and testing sets.

Application and Results:

Bayesian Networks:

Using the pgmpy package, a Bayesian Network model was built to predict **'type_encoded'** based on **'species_encoded'** and **'diet_encoded'**. Attained a test set accuracy of **20.00%,** suggesting limits in capturing intricate interactions between attributes. Considering their probabilistic methodology, the Bayesian Network model had the lowest accuracy, suggesting difficulties in identifying complex links in the dataset.

Classification Trees and Random Forests

Used the scikit-learn Random Forest and Decision Tree classifiers to predict **'type_encoded'** based on a variety of factors. On the test set, the Random Forest Classifier fared better with an accuracy of **59.65%,** while the Decision Tree Classifier achieved **57.89%** accuracy.

Conclusion:

By capturing the complex connections found in the dinosaur dataset, more research and modifications to the Bayesian Network topology may improve its forecasting power. The performance of Decision Tree Classifiers or other ensemble models beyond Random Forest may be improved via hyperparameter tuning or investigating alternative ensemble techniques.

In conclusion, the analysis shows that the Random Forest and Decision Tree Classifier performed better than the Bayesian Network model in classifying dinosaur types based on attributes on the Jurassic Park dinosaur dataset. The most accurate model was the Random Forest Classifier, which was followed by the Decision Tree Classifier.

References:

- A Gentle Introduction to Bayesian Belief Networks by Jason Brownlee. (Brownlee, n.d.)
- Decision Trees and Random Forests by Neil Liberman. (Liberman, n.d.)
- Scikit-learn documentation on Decision Trees and Random Forests: http://scikit-learn.org/stable/modules/tree.html.
- The pgmpy package documentation: https://pgmpy.readthedocs.io/_/downloads/en/latest/pdf/.