STATS 7022 - Data Science PG

Assignment3 - Question 4: Model Explanation

Possakorn Kittipipatthanapong a1873765

Introduction

In today's business world, data guides our decisions. One popular method used is the Random Forest. It's like using a series of flow charts (called **Decision Trees**) where each step makes a data-driven choice following on **Figure01**. Imagine using not just one but a group of these charts, each offering a different solution based on varied data inputs. That's the idea behind the **Bagging Ensemble Method** - we combine multiple trees' solutions for better accuracy. Together, these methods form the **Random Forest**.

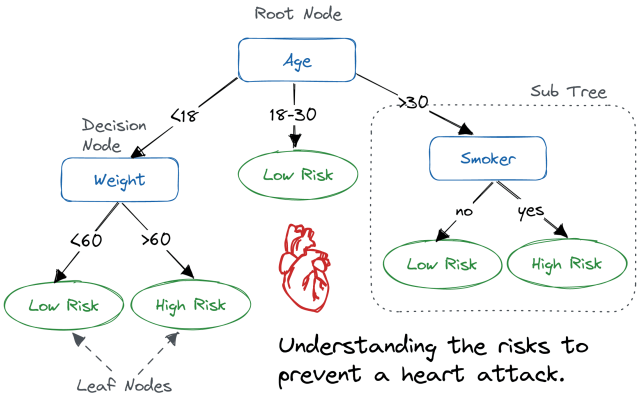

Figure 1: Decision Tree Diagram shown the principle of this algorithm

Type of Data Suitable for Random Forest:

The Random Forest method thrives on versatility, accommodating diverse types of data.
01 Predictors : which related to the input data.
- Quantitative: Numerical attributes, such as sales volumes, temperatures, or weights.
- Qualitative: Categorical attributes like product types, customer classifications, or color categories.

02 Response: which represented the output of the model.
- Regression: When the outcome or the response is continuous, like predicting sales values or temperatures.
- Classification: When the outcome is categorical, for instance, determining whether an email is spam or not, or classifying customers into segments.

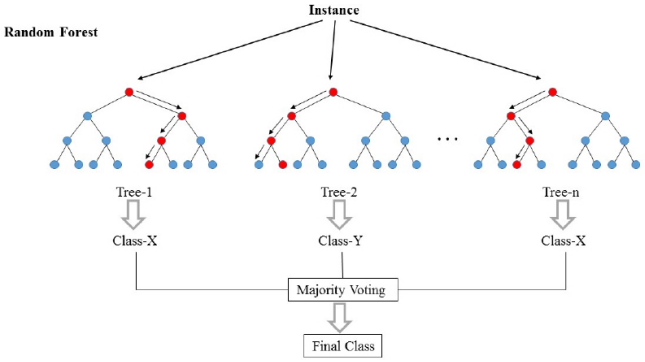

Figure 2: Random Forest represent the exmaple diagram of decisions

Understanding the Methodology:

Random Forest operates by creating a 'forest' of decision trees during training. Each tree is constructed using a random subset of data and features, thereby ensuring diversity. When making a prediction, every tree in this 'forest' provides its results. The model will use each results to make the outcome based on type of response as shown on **Figure02**.
For classification tasks, the mode (most common class) is extracted from the collective outputs of the trees.
For regression tasks, the mean of the tree outputs is utilized, ensuring a comprehensive, integrated final determination.

Model tuning:

Tuning is like adjusting a musical instrument for the best sound. While parameters are learned during the training process, hyperparameters are settings that we can adjust before testing to optimize performance. Random Forest has several such parameters that can be fine-tuned to achieve the best results.

Key Hyperparameters in Random Forest:

01 Number of Trees(n_estimators):
- Description: How many experts (trees) should be in our team? More trees like more experts in the team might give better results, but after a point, the improvement is minimal.
- Tuning guidance: This hyperparameter commonly to start with reasonable number like 100 as default based on Tidymodels in r and tend to be fixed for overview performance or computational constraints.

02 Maximum Depth of Trees(max_depth):
- Description: How detailed should each expert's analysis be? Too detailed, and we might get overly specific results.
- Tuning guidance: suggest to either set a max depth or leave it unspecified, allowing trees to expand until they contain less than a set number of samples.

03 Minimum Samples Split (min_samples_split):
- Description: How many data points should we consider before making a split in our flowchart?
- Tuning guidance: Increase this value to make your model more conservative, reducing the risk of overfitting. A good starting point might be 2, meaning at least two samples are required to make a further split at a node.

04 Maximum Features (m_try):
- Description: How many data features should we consider at each step?
- Tuning guidance: using all features might increase the risk of overfitting. A common practice is to use the square root of the total features for classification and one-third of the total for regression.

Tuning involves either manual experimentation, grid search, or random search to identify optimal hyperparameter values, ensuring a balance between model performance and computational efficiency.

Advantages of Adopting this model:

01 High Accuracy : In comparison to many other models, Random Forest frequently stands out in terms of prediction accuracy.
02 Versatility: It's good at avoiding over-specific results, especially with a lot of data.
03 Variable Importance: Random Forest provides insights into which features in the dataset are most influential, aiding domain experts in understanding which data features are most crucial.
04 User-friendly: It requires minimal preparation, making it user-friendly.

Potential Disadvantages:

01 Computational Intensity: For extremely large datasets, it can be resource-intensive and time-consuming.
02 Risk of Overfitting: Although robust, if not tuned properly, the model can still overfit, meaning that it might give great results on known data but perform poorly on new data.
03 Complexity: Individual decision trees are simple to understand, but a forest of them can be complex, making it harder to explain the model's decisions.

Reference:

Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32. This seminal paper provides deeper insights and foundational concepts underlying the Random Forest method.

Sruth, E R (2021). Understand Random Forest Algorithms With Examples. Retrieved from Analytics Vidhya.
Navlani, A. (2023). Decision Tree Classification in Python Tutorial. Retrieved from datacamp.