MCDA5580 Assignment 2

Team Member

* Hemalatha Srinivasan A00452621
* Ajay Jain A00455849
* Kin Wa, Chan A00467755

**Table of Content**

[Executive Summary 2](#_Toc128062925)

[Objectives 2](#_Toc128062926)

[Data Analysis 2](#_Toc128062927)

[Design/ Methodology/ Approach 2](#_Toc128062928)

[Overview 2](#_Toc128062929)

[Decision tree 2](#_Toc128062930)

[Random Forest 2](#_Toc128062931)

[Caret (Classification and Regression Training) 6](#_Toc128062932)

[Conclusion 7](#_Toc128062933)

[Appendix 7](#_Toc128062934)

[Reference/ citation 7](#_Toc128062935)

[R Script 7](#_Toc128062936)

# Executive Summary

# Objectives

# The main Objective of this process is to create a data model to classify the datasets into different classes using the results obtained and group together as different sets for easy understanding. This will create a model which will have high accuracy and less error. The obtained result will give clarity in determining the best and fewer performing categories which may help to understand the user preference. This will also provide an overview of where the growth of the organization is being affected. Overall, the aim of the model is to give a quality result for identifying where the improvement should be made to increase the future result.

# Data Analysis

# Design/ Methodology/ Approach

## Overview

## Decision tree

Decision Tree Algorithm is one of the classification algorithms for understanding the dataset and predicting the future results by recursively partitioning the input dataset into smaller sets. Since the decision trees visualization is effective which will make the user view and understand how the predictions are obtained by the algorithm. They provide results fast that will make them easy to train. They are good in handling both numerical and categorical datasets.

## Random Forest

(Basic description … advantage/ disadvantage)

**Methodology**

* Data handling

Data will be split to training (80 %) and testing (20%) set.

* Hyperparameter tuning
  + “mtry” parameter

It controls the number of randomly selected features that are used to determine the best split at each node. i.e. the level of fitting of the algorithm.

tuneRF() is used in tuning the parameter:

> model <- tuneRF(train[,1:6], train[,7] , mtryStart = 2)

mtry = 2 OOB error = 5.56%

Searching left ...

mtry = 1 OOB error = 26.45%

-3.753247 0.05

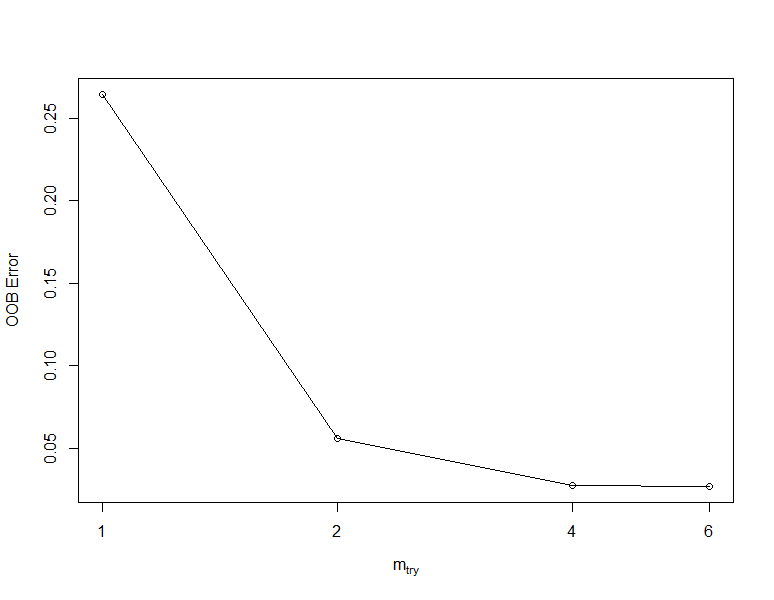
Searching right ...

mtry = 4 OOB error = 2.75%

0.5064935 0.05

mtry = 6 OOB error = 2.67%

0.02631579 0.05



We see mtry = 6 have the lowest OOB Error, so it is the optimal value.

* + “nodesize” parameter

This parameter specifies the minimum of terminal nodes in the tree. Smaller “nodesize” value results higher predictive accuracy, but may cause overfitting.

Random Forest models with different setting are built and tried to predict data in the test data set. The table above contains the confusion matrices and accuracy of these models. We can see node size =1 have the best accuracy, so it will become the optimal parameter.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Node size = 1 | Node size = 5 | Node size = 10 |
| ntree = 50 |  |  |  |
| ntree = 100 |  |  |  |

**Detail description of the optimal model**

The hyerparameter tuning yield mtry = 6 and node size = 1 the optimal value for the random forest model

## Caret (Classification and Regression Training)

# Conclusion

# Appendix

## Reference/ citation

Random Forest

https://cran.r-project.org/web/packages/randomForest/randomForest.pdf

<https://afit-r.github.io/random_forests> (tuning)

## R Script