**Happiness Prediction**

**Classification Tree, Random Forest, bagging, boosting**

Kumbham Nuthan Manideep

National Institute of Technology, Silchar

Data Mining Methods

Dr. Ripon Patigiri

**Abstract**

The World Happiness Report 2015 dataset contains data on the happiness levels of people in 159 countries around the world. It includes information on a variety of factors that are thought to contribute to happiness, such as economic factors (GDP per capita), social factors (family, health, and freedom), and trust in government. The data also includes a happiness score and a happiness rank for each country. The happiness score and happiness rank are based on a poll in which people were asked to rate their overall happiness on a scale of 0 to 10. The economy, family, health, freedom, trust, and generosity columns contain various measures of those factors. Overall, the World Happiness Report 2015 dataset provides valuable insights into the factors that contribute to happiness and well-being at the national level, and can be used by researchers, policymakers, and others to better understand and promote happiness around the world.

**Overview**

The aim of this project is to use classification tree, random forest, bagging, and boosting algorithms to predict the level of happiness of individuals based on a given dataset. The dataset, obtained from Kaggle, contains information on happiness levels in 159 countries in 2015, including factors such as GDP per capita, family, health, freedom, trust in government, and generosity. The first phase involved cleaning and pre-processing the data to ensure its suitability for analysis, and then splitting it into training and testing sets for model evaluation. A classification tree is a graphical representation of a series of decisions or rules that are made based on the values of the predictor variables. The basic idea of random forest is to train multiple decision trees on different random subsets of the training data, and then average their predictions to obtain the final prediction. This reduces the risk of overfitting and increases the generalization performance of the model. Bagging, short for bootstrap aggregating, is ensemble learning method that combines multiple models to improve the accuracy and stability of the predictions. The basic idea is to generate multiple bootstrap samples of the training data, and then train a separate model on each sample. Boosting is ensemble learning method that combines multiple models to improve the accuracy of the predictions. The basic idea is to train multiple weak models sequentially, and each model is trained on the residuals of the previous model.

**Data**

The dataset can be found on Kaggle, and it contains data on happiness levels of people in 159 countries around the world. The data is collected in the year 2015 and it includes information on a variety of factors that are thought to contribute to happiness, such as:

|  |  |
| --- | --- |
| **Column Name** | **Description** |
| Country or Region | Name of the country |
| Happiness Score | A composite score of overall well-being |
| Economy (GDP per Capita) | Measure of the economic production of a country |
| Family | Measure of social support |
| Health (Life Expectancy) | Measure of the health of citizens |
| Freedom | Measure of freedom to make life choices |
| Generosity | Measure of generosity of citizens |
| Trust (Government Corruption) | Measure of trust in government |

In this dataset, the response variable is 'Happiness Score', and predictor variables are 'Economy (GDP per Capita)', 'Family', 'Health (Life Expectancy)', 'Freedom', 'Generosity', and 'Trust (Government Corruption)'.

**Research Questions**

* How accurately can a neural network model predict happiness levels based on factors such as GDP per capita, family, health, freedom, trust in government, and generosity?
* Which of the factors, including GDP per capita, family, health, freedom, trust in government, and generosity, have the most significant impact on predicting happiness using a neural network model?
* Can a neural network model effectively identify nonlinear relationships between factors such as GDP per capita, family, health, freedom, trust in government, and generosity, and happiness levels?
* How does the inclusion of different combinations of factors, such as GDP per capita, family, health, freedom, trust in government, and generosity, affect the accuracy of the neural network model in predicting happiness levels?
* What is the predictive power of a neural network model in forecasting happiness levels using factors such as GDP per capita, family, health, freedom, trust in government, and generosity, and how does it compare to other traditional statistical methods for happiness prediction?

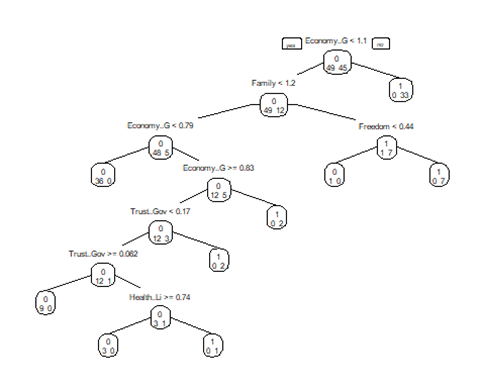
**Analysis**

Firstly, the data from the dataset that is obtained from the sources is loaded, pre-processed and cleaned and the categorial variables are converted into dummy variables. Seed value was then set to a random number (here 666 is taken).

To start, pre-processing the data by taking the target variable and calculating the mean of that variable is done. Then a new binary variable based on this mean is created. The World Happiness Report 2015 dataset was divided into 60% training and 40% validation. The first set, the training set, would be used to fit the Neural Network model and would contain 60% of the total data. The second set, the validation set, would be used to evaluate the performance of the model and would contain 40% of the total data.

**Classification Tree Model**

A classification tree is a graphical representation of a series of decisions or rules that are made based on the values of the predictor variables. The tree was built with all variables. Variables actually used in tree construction are “Economy..GDP.per.Capita.”, “Family”, “Freedom”, “Health..Life.Expectancy.”, “Trust..Government.Corruption”. The classification tree is shown below



The confusion matrix for the training data of classification tree model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 49 | 0 |
| 1 | 0 | 45 |

The accuracy of the model is 1. The sensitivity of the model is 1. The specificity of the model is 1.

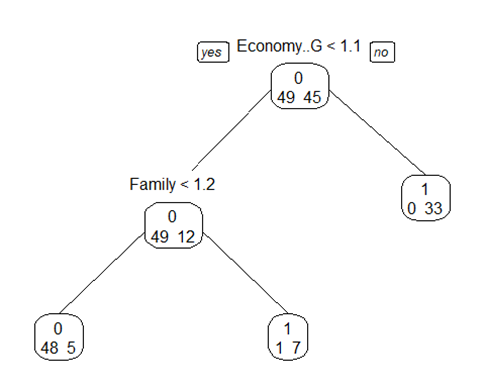
The confusion matrix for the validation data of classification tree model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 29 | 9 |
| 1 | 6 | 20 |

The accuracy of the model is 0.765625. The sensitivity of the model is 0.8286. The specificity of the model is 0.6897.

**Pruning Tree**

Pruning is a technique used to prevent overfitting in decision trees. Pruning can improve the performance of decision trees by reducing overfitting and increasing the generalization ability of the model. The pruning tree is shown below



The confusion matrix for the training data of pruning tree model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 48 | 5 |
| 1 | 1 | 40 |

The accuracy of the model is 0.9361702. The sensitivity of the model is 0.9796. The specificity of the model is 0.8889.

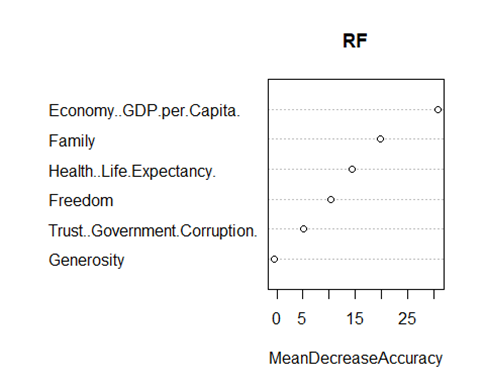
The confusion matrix for the validation data of classification tree model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 30 | 9 |
| 1 | 5 | 20 |

The accuracy of the model is 0.78125. The sensitivity of the model is 0.8571. The specificity of the model is 0.6897.

**Random Forest**

The basic idea of random forest is to train multiple decision trees on different random subsets of the training data, and then average their predictions to obtain the final prediction. This reduces the risk of overfitting and increases the generalization performance of the model. The random forest plot is shown below



The importance of each variable is calculated by measuring the reduction in the impurity of the nodes of the trees in the forest when that variable is used for splitting. The impurity of a node is a measure of how well the node separates the classes of the response variable. The greater the reduction in impurity, the more important the variable.

The plot can be used to identify the most important predictors in the model, which can be useful for understanding the relationships between the predictor variables and the response variable.

The most important variables in the plot are “GDP”, “Family”, “Life Expectancy”, “Freedom”.

The confusion matrix for the training data of random forest model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 49 | 3 |
| 1 | 0 | 42 |

The accuracy of the model is 0.9787234. The sensitivity of the model is 1. The specificity of the model is 0.9333.

The confusion matrix for the validation data of random forest model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 29 | 9 |
| 1 | 6 | 20 |

The accuracy of the model is 0.765625. The sensitivity of the model is 0.8286. The specificity of the model is 0.6897.

**Bagging**

Bagging, short for bootstrap aggregating, is ensemble learning method that combines multiple models to improve the accuracy and stability of the predictions. The basic idea is to generate multiple bootstrap samples of the training data, and then train a separate model on each sample.

The confusion matrix for the training data of bagging model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 47 | 5 |
| 1 | 2 | 40 |

The accuracy of the model is 0.9255319. The sensitivity of the model is 0.9592. The specificity of the model is 0.8889.

The confusion matrix for the validation data of bagging model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 28 | 9 |
| 1 | 7 | 20 |

The accuracy of the model is 0.765625. The sensitivity of the model is 0.8. The specificity of the model is 0.6897.

**Boosting**

Boosting is ensemble learning method that combines multiple models to improve the accuracy of the predictions. The basic idea is to train multiple weak models sequentially, and each model is trained on the residuals of the previous model.

The confusion matrix for the training data of boosting model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 46 | 4 |
| 1 | 3 | 41 |

The accuracy of the model is 0.9255319. The sensitivity of the model is 0.9388. The specificity of the model is 0.9111.

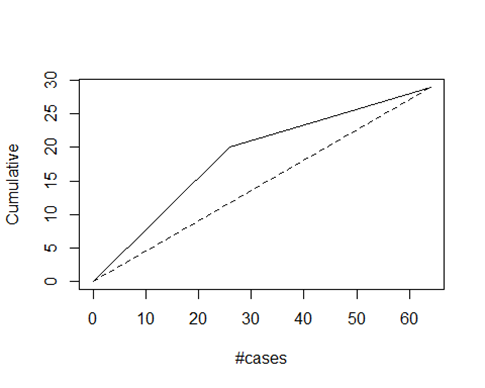
The confusion matrix for the validation data of boosting model is shown below

|  |  |  |  |
| --- | --- | --- | --- |
|  | Actual | | |
| Predicted |  | 0 | 1 |
| 0 | 27 | 9 |
| 1 | 8 | 20 |

The accuracy of the model is 0.734375. The sensitivity of the model is 0.7714. The specificity of the model is 0.6897.

The analysis reveals that among all constructed models, the random forest model stands out as the best performer. When applied to the validation data, this model demonstrates the highest accuracy, with a value of 0.765625

The gains plot for the best model is shown below



The gains plot demonstrates a direct proportionality between the area enclosed by the dotted line (representing random sample selection) and the model curve. A larger area signifies superior model performance. This plot serves as a visual tool to effectively represent the predictive power of the model and enables straightforward comparison across multiple models.

**Conclusion**

After thorough analysis of all the models, it can be concluded that the random forest model performs the best. The model achieves an accuracy of 0.765625, sensitivity 0.8286, specificity of 0.6897 based on the evaluation results. The most important variables in the plot are “GDP”, “Family”, “Life Expectancy”, “Freedom”.

**References**

Alfaro, E., Gámez, M., & García, N. (2013). **adabag**: An*R*Package for Classification with Boosting and Bagging. *Journal of Statistical Software*, *54*(2). <https://doi.org/10.18637/jss.v054.i02>

Hofner, B., Mayr, A., Robinzonov, N., & Schmid, M. (2014). Model-based boosting in R: a hands-on tutorial using the R package mboost. *Computational Statistics*, *29*(1–2), 3–35. <https://doi.org/10.1007/s00180-012-0382-5>

Wright, M. N., & Ziegler, A. (2017). **ranger**: A Fast Implementation of Random Forests for High Dimensional Data in *C++* and *R*. *Journal of Statistical Software*, *77*(1). <https://doi.org/10.18637/jss.v077.i01>