Data Mining Coursework

Registration number: 210132797

Word Count: 2300

**Excluding front page content, abstract, content table, tables, figure captions, bibliography, and appendix*

Contents

Abstract

This report assesses Random Forest's (RF) and Multilayer Perceptron (MLP) algorithm performance. The result shows that RF outperforms MLP based on accuracy, MCC, and training runtime. The report is structured as follows. The introduction section describes the aims and the data used for the experiment. Then, data mining theory is discussed in the second section. The third section discusses the exploratory data analysis and missing value handling methods. The fourth section discusses feature engineering and experimental setup. Results and discussion is discussed in the fifth section. Lastly, the last section discussed the conclusion and reflections of the report.

Introduction

Machine learning can outperform basic classification methods/algorithms(Montebruno et al., 2020). However, compared to other data mining algorithms, Random Forest and Neural Networks are known for their performance in dealing with a large amount of data and high computational costs (Al-Jarrah et al., 2015; Bahel et al., 2020; Shichkin et al., 2018; N. Zheng, 2019). However, both data mining has their own workflow and fundamental mathematics, which they rely on, making them unique in terms of statistical performance and computational cost. This report aims to compare the performance of the two algorithms.

The research will be conducted on the Airline Satisfaction Dataset, which has been rearranged in accordance with the INF6028 Coursework. The dataset contains 23 variables consisting of numeric, ordinal, and categorical data. The target variable is *Satisfaction*, which indicates passengers' Satisfaction (satisfied – neutral or dissatisfied). The complete description of the data is mentioned in *Table 1.*

Data Mining Theory

Multi-Layer Perceptron

Multi-Layer Perceptron (MLP, Neural Network) is an algorithm which is based on the concepts of biological neural networks and the human brain (Han et al., 2018; Hinton, 1989). This model mimics the functioning of neurons by using a graph of connected processing units. Generally, there are three layers of NN:n input layer, several or one hidden layers, and an output layer (Wang, 2003). Each layer consists of neurons that connect the layers. Each connection between neurons is called a weight. The weights are assigned a value during training (Kavzoglu & Mather, 2003). The objective of the training process is to minimise the error between values predicted by MLP and actual values. (Kavzoglu & Mather, 2003). NN/MLP has shown successful results in recent years (Abiodun et al., 2019). Furthermore, there are various examples of the usage of this algorithm for binary classification (Fanelli et al., 1993; Faris et al., 2016; N. Zheng, 2019). Although there are several types and improvements of MLP (e.g., Convolutional, Recurrent, LSTM, Boltzman Machine) (N. Zheng, 2019), we only use the default MLP provided by KNIME. KNIME only allow us to set the number of iterations (epochs), the number of layers (depth), and the number of neurons (nodes) per layer.

Random Forest

RF models combine ensembles and tree-based learning to help develop predictive models. It combines a number of different classification trees and averages their predictive accuracy, thereby improving the model's overall efficiency (Twala & Mekuria, 2013). Svetnik et al. (2003) also asserted that RF could reduce overfitting by combining individual decision tree prediction. In addition, RF has excellent performance for binary classification problems (Sobreiro et al., 2021) as in our data.

Evaluation Metrics

We use several metrics for evaluating the performance of each algorithm. A confusion matrix is used as it can generally describe the performance of our algorithm (Canbek et al., 2017). Although AUC and ROC are pretty popular and consistent (Jin Huang & Ling, 2005), recent research asserted that AUC is potentially misleading and sensitive to the class distribution (Lobo et al., 2008; Tharwat, 2021). MCC is used as it scored as the most robust classification method based on several categories (i.e., Base measure correlations, Imbalance uncorrelation, Distinctness, Output smoothness, Monotonicity, Consistency, Universal Discriminancy) (Canbek et al., 2021). In addition to MCC, Accuracy is also used due to its suitability for balanced-class binary classification (Sokolova & Lapalme, 2009; Tharwat, 2021).

Due to the importance of memory usage and runtime in evaluating ML methods (Al-Jarrah et al., 2015; Gómez-Carmona et al., 2019), memory usage and the node's runtime for each algorithm also will be presented and discussed in the later sections.

Data Exploration and Preparation

Exploratory Data Analysis

The target variable is *Satisfaction*, a binary categorical variable which indicates whether a passenger is satisfied or neutral/dissatisfied with their flight journey. As the ratio between satisfied and dissatisfied or neutral category is 44%: 56%, the target data is considered balanced; therefore, no specific treatment is needed (Weiss, 2001). There are no missing values in the target variable.

Table 1. Dataset Overview

Column	Attribute type	No. Missing	Outliers /Fault	No. Missing + Fault	Missing Percentage	Imputation Method
ID	No Use					
Age	Numeric Integer	0	999 is faulty value	580	11%	Median
Online check-in	Categorical Binary	1531		1531	29%	Exclusion
Flight Distance	Numeric Integer	747		747	14%	Median
Departure/Arrival time convenient	Ordinal	0	0 is faulty value	277	5%	Litwise Deletion
Ease of Online booking	Ordinal	0	0 is faulty value	229	4%	Litwise Deletion
Gate location	Ordinal	0		0	0%	
Food and drink	Ordinal	0	0 is faulty value	5	0%	Litwise Deletion
Seat comfort	Ordinal	0		0	0%	
Inflight entertainment	Ordinal	0		0	0%	
On-board service	Ordinal	0		0	0%	
Leg room service	Ordinal	0	0 is faulty value	28	1%	Litwise Deletion
Baggage handling	Ordinal	0		0	0%	
Checkin service	Ordinal	0		0	0%	
Inflight service	Ordinal	0		0	0%	
Cleanliness	Ordinal	0		0	0%	
Departure Delay in Minutes	Numeric Integer	747		747	14%	Linear Regression
Arrival Delay in Minutes	Numeric Integer	15		15	0%	Linear Regression
Gender	Categorical Binary	0		0	0%	
Customer Loyalty	Categorical Binary	0		0	0%	
Type of Travel	Categorical Binary	0		0	0%	
Class	Categorical	0		0	0%	

From the table above, we can see that there are missing values from the *Online check-in*, *Flight Distance, Departure delay in minutes,* and *Arrival delay in minutes* variables. Although most of the variables only missing less than 15% of the total data, the online check-in variable missed almost 30% of the entire data. This missing value percentage is considered high; hence, excluding the variable from the further step is essential to preserve the balance in the data.

In addition to this issue, there is also inconsistency in the data. The *Age* variable consists of 999, which makes it rather impossible for today's homo sapiens to live that long (Manton et al., 1991). This might suggest that the 999 values are faulty; thus, they can be considered as missing values or outliers. This statement is also strengthened by the output of the outliers' node, which changes the value of 999 to a missing value. Inconsistency also appears in the several ordinal variables (e.g., *Ease of Online booking, Food and Drink, Leg room service)*. Several ordinal variables start with 1, and others start with 0.

Each variable with missing values is treated differently. Missing values from *Ease of Online booking, Food and Drink, Leg room service* variables dealt with replacing them by the *Litwise Deletion,* which will ignore the entire row of the missing value and will only retain complete data. Although this method sometimes results in biased estimation, the performance is considered good (Makaba & Dogo, 2019). In addition to this, most used imputation methods (i.e., Mean Method, Multiple Imputation) are not the best method to treat ordinal data (Quintero & LeBoulluec, 2018). Due to the high correlation between the *Departure delay in minutes,* and *Arrival delay in minutes* variable, a linear regression to replace missing values using another variable can be applied (Musil et al., 2002; Templ et al., 2011). Missing values in the *Travel Distance* and *Age* variable will be handled using the *median* value, which is considered better for handling numeric data types (Jadhav et al., 2019).

Table 2. Correlation value between target variable and other variable.

Variable 1	Variable 2	Correlation	Absolute Correlation	P Value	Hypothesis
satisfaction	Type of Travel	-0.463	0.463	0	Accept H0
satisfaction	Class	-0.453	0.453	0	Accept H0
satisfaction	Inflight entertainment	0.437	0.437	0	Accept H0
satisfaction	Seat comfort	0.390	0.390	0	Accept H0
satisfaction	On-board service	0.360	0.360	0	Accept H0
satisfaction	Cleanliness	0.345	0.345	0	Accept H0
satisfaction	Leg room service	0.322	0.322	0	Accept H0
satisfaction	Baggage handling	0.298	0.298	0	Accept H0
satisfaction	Inflight service	0.284	0.284	0	Accept H0
satisfaction	Check in service	0.270	0.270	0	Accept H0
satisfaction	Food and drink	0.226	0.226	0	Accept H0
satisfaction	Flight Distance	0.209	0.209	0	Accept H0
satisfaction	Ease of Online booking	0.207	0.207	0	Accept H0
satisfaction	Customer Loyalty	-0.192	0.192	0	Accept H0
satisfaction	Age	0.134	0.134	0	Accept H0
satisfaction	Arrival Delay in Minutes	-0.098	0.098	8.60E-12	Accept H0
satisfaction	Departure Delay in Minutes	-0.081	0.081	1.76E-08	Accept H0
satisfaction	Departure/Arrival time convenient	-0.057	0.057	7.60E-05	Accept H0
satisfaction	Gate location	-0.039	0.039	0.006971856	Accept H0
satisfaction	Gender	-0.020	0.020	0.174630349	Reject H0

Departure Delay in Minutes	Arrival Delay in Minutes	0.95	0.95	0.00E+00	Accept H0

The correlation using Pearson's and ranked method analysis shows a considerable correlation score between *Satisfaction* with *type of travel, class,* and *inflight entertainment* with P-Value < 0.05, which indicates H0 is accepted. There is a strong association and correlation between those variables. The significant correlation scores are also shown by *Departure delay in minutes* and *Arrival delay in minutes* (c = 0.96). In addition to that, there is also a causative relationship between the two variables (Z. Zheng et al., 2021). Hence, linear regression is used to predict the missing values from each variable using one another.

Experimental Setup

Two types of experiments are conducted. The first one is to see the effect of iteration and cross-validation/partitioning on MLP. Before the first experiment, the Feature Engineering process consists of One-Hot-Encoding, String to Integer conversion, and normalisation is held. The second experiment is to assess the effect of Feature Engineering on RF. Each experiment measures the MLP's train-test accuracy, train-test accuracy difference, and execution time. Lastly, a comparison between the best performing RF and MLP is conducted.

Feature Engineering

Overall, there are three primary nodes used for this step. The first node is the One-to-Many node as a substitution for the One Hot Encoding process. One Hot Encoding is necessary when a non-binary categorical variable is used, like this class variable containing three non-hierarchical values (Choong & Lee, 2017). The second is the Category to Number node to change the *gender, customer loyalty, class, and type of travel* variables to number variables from string. Lastly, the Normalizer node normalises data numbers (travel distance, age, departure & arrival delay time). As such, this step is critical in order to preserve the significance of each variable. That way, all variables will have the same distance between their highest and lowest point, improving the classification results. (Singh & Singh, 2020). All variables are normalised to values between 0 and 1.

Partitioning

Prior to processing the data with the model, a partitioning process of splitting the data for trains and testing is conducted to evaluate better the model's performance (Steyerberg & Harrell, 2016). K-Fold cross-validation and regular partitioning are used to compare both methods' effects on MLP's overfitting. We use k-fold cross-validation with the number of k = 5 in accordance with Marcot & Hanea (2021). The regular partitioning is conducted using a 4:1 ratio between training and testing. Both methods are conducted using the Stratified Sampling method based on the Satisfaction variable or the target variable. This method ensures that the target variable is equally distributed between the training and testing set, improving the prediction result (Shahrokh Esfahani & Dougherty, 2014). The cross-validation process used the X-Partitioner node which is placed before the learner and predictor nodes, and X-Aggregator put after for K-Fold cross-validation.

MLP

The MLP used in KNIME is based on the RProp algorithm for multilayer feedforward networks (Riedmiller & Braun, 1993). The MLP use the default learning rate, momentum, and activation function from KNIME. Since there is no established method to determine the optimum value of hyperparameters, we used the trial-and-error process (Goodfellow et al., 2016) and reached the optimum values of hyperparameters with 200 iterations, two hidden layers, and 30 neurons per layer. In addition to this, learning rate, momentum, and bias are automatically configured by KNIME

RF

The Splitting criterion used for the random forest is Information Gain. Compared to Gini Index, Information gain has the upper hand when tested on several well-referred datasets (Jain et al., 2018).

Results and Discussion

Table 3. Experiment Result

Algorithm	One Hot Encoding + Normalisation	Partitioning	Iterations	Training Accuracy	Testing Accuracy	Train-Test Difference	MCC	Time
MLP	Yes	5-Fold Cross Validation	100	90.67%	90.70%	0.03%	0.806	3711
MLP	Yes	5-Fold Cross Validation	200	91.03%	91.10%	0.07%	0.816	7411
MLP	Yes	5-Fold Cross Validation	300	90.36%	90.40%	0.04%	0.804	11216
MLP	Yes	Regular Partitioning	100	92.30%	89.60%	2.70%	0.795	3843
MLP	Yes	Regular Partitioning	200	95.30%	89.90%	5.40%	0.794	7844
MLP	Yes	Regular Partitioning	300	97.10%	90.20%	6.90%	0.800	11005
RF	Yes	Yes	-	91.90%	-	-	0.836	228
RF	No	Yes	-	92.10%	-	-	0.840	225

Based on the first experiment explained in the previous section, cross-validation indeed affects the MLP's performance on the overfitting issue. It is inferred from the result that the experiment with the regular partitioning method has a larger difference between the training and testing accuracy, indicating an overfitting issue. The experiment with the regular partitioning method has a 0.11%, 5.4%, and 6.9% difference, while the 5-Fold Cross validation's difference is below 0.1%. The result is in line with Xu & Goodacre (2018), which asserts that cross-validation tent to minimise overfitting issues by aggregating numbers of different training. In addition to this, 5-Fold validation results a better accuracy and MCC score (90.7%, 91.1%, 90.4% for accuracy, 0.806, 0.816, 0.804 for MCC) compared to regular partitioning (89.6%, 89.9%, 90.2% for accuracy, 0.795, 0.794, 0.8 for MCC). However, the definitive relationship between iteration and train-test difference can only be inferred from the regular partitioning method experiment—the train-test difference increases as the number of iteration increase.

Both RF algorithms have better accuracy and MCC score compared to MLP. In addition to this, the running time of RF is faster than MLP. This makes RF superior to MLP in terms of classification performance and computational cost.

The random forest experiment that conducts feature engineering has lower accuracy than not using feature engineering (91.90% with feature engineering, 92.1% without feature engineering). This might be attributed to two things: the RF algorithm does not need normalisation (Ferreira et al., 2019) and one-hot-encoding. The second is that feature engineering might cause information loss in some cases.

Most contributing factors can be inferred from feature importance, which measures a variable's significance or relevancy in classifying the target (Płoński, 2020). The score is calculated using a formula by Silipo et al. (2014).

Table 4. Feature importance and absolute correlation value

Variables	splits (level 0	splits (level 1	splits (level 2	candidates (level 0)	candidates (level 1)	candidates (level 2)	importance (all levels)	Absolute Correlation Value
Type of Travel	15	34	24	15	50	67	2.038	0.463
Inflight entertainment	12	24	28	14	42	79	1.783	0.437
Class	18	15	23	22	36	65	1.589	0.453
Seat comfort	16	17	25	22	36	92	1.471	0.390
Cleanliness	10	8	20	23	33	72	0.955	0.345
Leg room service	11	9	14	24	33	77	0.913	0.322
Ease of Online booking	1	21	37	24	53	80	0.900	0.207
On-board service	4	11	17	15	39	88	0.742	0.360
Inflight service	4	11	23	20	48	92	0.679	0.284
Customer Loyalty	2	8	18	23	38	80	0.522	0.192
Departure Delay in Minutes	1	6	23	15	40	76	0.519	0.081
Checkin service	3	2	23	18	37	79	0.512	0.270
Arrival Delay in Minutes	1	7	23	26	48	74	0.495	0.098
Flight Distance	0	9	24	24	47	80	0.491	0.209
Baggage handling	1	5	21	13	35	81	0.479	0.298
Food and drink	1	3	18	23	30	71	0.397	0.226
Age	0	5	16	22	36	84	0.329	0.134
Gate location	0	2	10	21	40	73	0.187	0.039
Departure/Arrival time convenient	0	0	5	20	34	83	0.060	0.057
Gender	0	1	1	16	45	91	0.033	0.020

The feature importance score and the correlation score loosely have some correlation. The bigger the feature importance score, the bigger the correlation score. However, the correlation value measures the positive/negative 'change' in one feature as the other is increased or decreased, While feature importance in tree-based models is more likely to identify which features differentiate classes (Perrier, 2015). In addition to this, the availability of feature importance and the visualisation of the Random Forest algorithm makes the model easier to be interpreted and not treated as a 'BlackBox' like MLP.

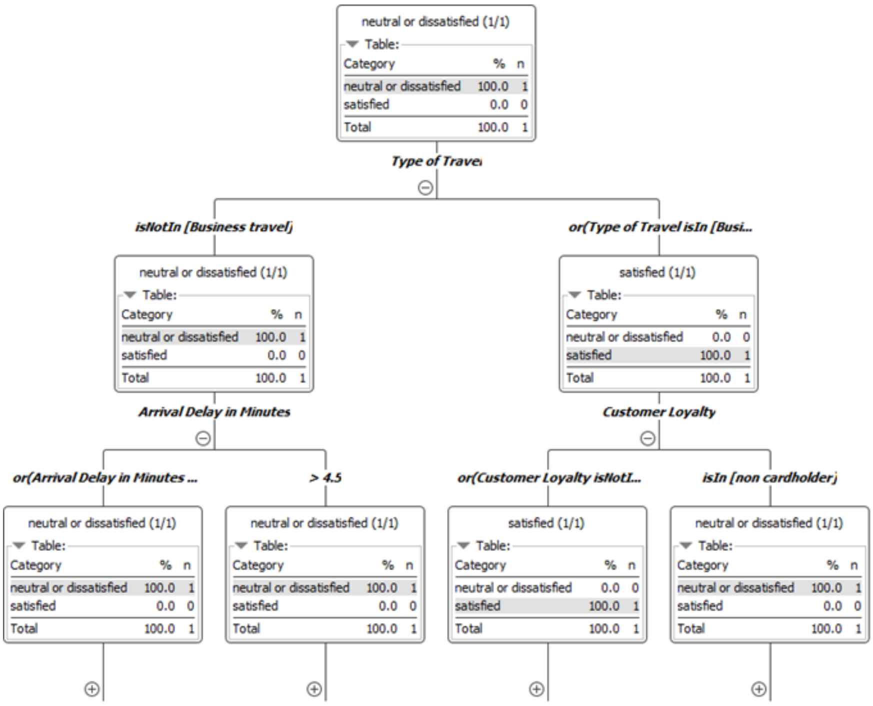

Figure 1 Tree View Example.

The tree view of the Tree-based model can be used to better understand how the model classifies the data based on the variable. In addition to that, threshold information and information gain/entropy information could also strengthen the interpretability of the model.

Conclusion and Reflections

The predictive and computational performance of both RF and MLP was compared in Airline Satisfaction Dataset using KNIME. The result shows that RF performs better than MLP for accuracy, MCC, and training runtime. In addition to that, the RF model requires less process for feature engineering. The RF model that performs without feature engineering is better than its counterpart, which uses feature engineering due to information loss in the process. Moreover, RF is more explainable due to the availability of feature importance information and tree visualisation.

Iteration and partitioning methods affect the performance of MLP for its overfitting issue, as cross-validation can minimise overfitting, and on the particular condition, the larger number of iterations can also cause overfitting to the trained model.

Although the 92% of accuracy for RF is good, things could continuously be improved. For example, further studies can use GridSearchCV (Grgić et al., 2021) and GARF feature selection to improve RF performance (Paul et al., 2017). In addition, although RF outperforms MLP in this experiment, a more advanced Neural Networks with fully configured hyperparameters could perform better than other machine learning algorithms(Ekman, 2021).

References

Abiodun, O. I., Kiru, M. U., Jantan, A., Omolara, A. E., Dada, K. V., Umar, A. M., Linus, O. U., Arshad, H., Kazaure, A. A., & Gana, U. (2019). Comprehensive Review of Artificial Neural Network Applications to Pattern Recognition. *IEEE Access*, *7*, 158820–158846. https://doi.org/10.1109/ACCESS.2019.2945545

Al-Jarrah, O. Y., Yoo, P. D., Muhaidat, S., Karagiannidis, G. K., & Taha, K. (2015). Efficient Machine Learning for Big Data: A Review. *Big Data Research*, *2*(3), 87–93. http://arxiv.org/abs/1503.05296

Bahel, V., Pillai, S., & Malhotra, M. (2020). A Comparative Study on Various Binary Classification Algorithms and their Improved Variant for Optimal Performance. *2020 IEEE Region 10 Symposium (TENSYMP)*, 495–498. https://doi.org/10.1109/TENSYMP50017.2020.9230877

Canbek, G., Sagiroglu, S., Temizel, T. T., & Baykal, N. (2017). Binary classification performance measures/metrics: A comprehensive visualised roadmap to gain new insights. *2017 International Conference on Computer Science and Engineering (UBMK)*, 821–826. https://doi.org/10.1109/UBMK.2017.8093539

Canbek, G., Taskaya Temizel, T., & Sagiroglu, S. (2021). BenchMetrics: a systematic benchmarking method for binary classification performance metrics. *Neural Computing and Applications*, *33*(21), 14623–14650. https://doi.org/10.1007/s00521-021-06103-6

Choong, A. C. H., & Lee, N. K. (2017). Evaluation of convolutionary neural networks modeling of DNA sequences using ordinal versus one-hot encoding method. *2017 International Conference on Computer and Drone Applications (IConDA)*, 60–65. https://doi.org/10.1109/ICONDA.2017.8270400

Ekman, M. (2021). *Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow* (1st ed.). Addison-Wesley Professional.

Fanelli, S., di Martino, M., & Protasi, M. (1993). An efficient algorithm for the binary classification of patterns using MLP-networks. *IEEE International Conference on Neural Networks*, 936–943. https://doi.org/10.1109/ICNN.1993.298683

Faris, H., Aljarah, I., & Mirjalili, S. (2016). Training feedforward neural networks using multi-verse optimiser for binary classification problems. *Applied Intelligence*, *45*(2), 322–332. https://doi.org/10.1007/s10489-016-0767-1

Ferreira, P., C. Le, D., & Zincir-Heywood, N. (2019). Exploring Feature Normalization and Temporal Information for Machine Learning Based Insider Threat Detection. *2019 15th International Conference on Network and Service Management (CNSM)*, 1–7. https://doi.org/10.23919/CNSM46954.2019.9012708

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning (Adaptive Computation and Machine Learning series)* (Illustrated ed.). The MIT Press.

Gómez-Carmona, O., Casado-Mansilla, D., López-de-Ipiña, D., & García-Zubia, J. (2019). Simplicity is Best. *Proceedings of the 9th International Conference on the Internet of Things*, 1–8. https://doi.org/10.1145/3365871.3365889

Glass, G., V., & Hopkins, K. D. (1995). *Statistical Methods in Education and Psychology (3rd Edition)* (3rd ed.). Allyn & Bacon.

Grgić, V., Mušić, D., & Babović, E. (2021). Model for predicting heart failure using Random Forest and Logistic Regression algorithms. *IOP Conference Series: Materials Science and Engineering*, *1208*(1), 012039. https://doi.org/10.1088/1757-899X/1208/1/012039

Han, T., Jiang, D., Zhao, Q., Wang, L., & Yin, K. (2018). Comparison of random forest, artificial neural networks and support vector machine for intelligent diagnosis of rotating machinery. *Transactions of the Institute of Measurement and Control*, *40*(8), 2681–2693. https://doi.org/10.1177/0142331217708242

Hinton, G. E. (1989). Connectionist learning procedures. *Artificial Intelligence*, *40*(1–3), 185–234. https://doi.org/10.1016/0004-3702(89)90049-0

Jadhav, A., Pramod, D., & Ramanathan, K. (2019). Comparison of Performance of Data Imputation Methods for Numeric Dataset. *Applied Artificial Intelligence*, *33*(10), 913–933. https://doi.org/10.1080/08839514.2019.1637138

Jain, V., Phophalia, A., & Bhatt, J. S. (2018). Investigation of a Joint Splitting Criteria for Decision Tree Classifier Use of Information Gain and Gini Index; Investigation of a Joint Splitting Criteria for Decision Tree Classifier Use of Information Gain and Gini Index. In *TENCON 2018 - 2018 IEEE Region 10 Conference*.

Jin Huang, & Ling, C. X. (2005). Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, *17*(3), 299–310. https://doi.org/10.1109/TKDE.2005.50

K. (2019, June 11). *Random Forests Vs. Neural Networks: Which Is Better and When?* JPT. Retrieved 5 May 2022, from https://jpt.spe.org/random-forests-vs-neural-networks-which-better-and-when

Kavzoglu, T., & Mather, P. M. (2003). The use of backpropagating artificial neural networks in land cover classification. *International Journal of Remote Sensing*, *24*(23), 4907–4938. https://doi.org/10.1080/0143116031000114851

Lee, T. B. (2019, December 2). *How neural networks work—and why they've become a big business*. Ars Technica. Retrieved 5 May 2022, from https://arstechnica.com/science/2019/12/how-neural-networks-work-and-why-theyve-become-a-big-business/#:%7E:text=The%202012%20breakthrough%E2%80%94the%20deep,had%20become%20available%20by%202012.

Lobo, J. M., Jiménez-valverde, A., & Real, R. (2008). AUC: A misleading measure of the performance of predictive distribution models. In *Global Ecology and Biogeography* (Vol. 17, Issue 2, pp. 145–151). https://doi.org/10.1111/j.1466-8238.2007.00358.x

Manton, K. G., Stallard, E., & Tolley, H. D. (1991). Limits to Human Life Expectancy: Evidence, Prospects, and Implications. *Population and Development Review*, *17*(4), 603. https://doi.org/10.2307/1973599

Marcot, B. G., & Hanea, A. M. (2021). What is an optimal value of k in k-fold cross-validation in discrete Bayesian network analysis? *Computational Statistics*, *36*(3), 2009–2031. https://doi.org/10.1007/s00180-020-00999-9

Montebruno, P., Bennett, R. J., Smith, H., & Lieshout, C. van. (2020). Machine learning classification of entrepreneurs in British historical census data. *Information Processing & Management*, *57*(3), 102210. https://doi.org/10.1016/j.ipm.2020.102210

Musil, C. M., Warner, C. B., Yobas, P. K., & Jones, S. L. (2002). A Comparison of Imputation Techniques for Handling Missing Data. *Western Journal of Nursing Research*, *24*(7), 815–829. https://doi.org/10.1177/019394502762477004

Paul, D., Su, R., Romain, M., Sébastien, V., Pierre, V., & Isabelle, G. (2017). Feature selection for outcome prediction in oesophageal cancer using genetic algorithm and random forest classifier. *Computerized Medical Imaging and Graphics : The Official Journal of the Computerized Medical Imaging Society*, *60*, 42–49. https://doi.org/10.1016/j.compmedimag.2016.12.002

Perrier, A. (2015). *Feature Importance in Random Forests*. https://alexisperrier.com/datascience/2015/08/27/feature-importance-random-forests-gini-accuracy.html

Płoński, P. (2020). *Random Forest Feature Importance Computed in 3 Ways with Python*. https://mljar.com/blog/feature-importance-in-random-forest/#:%7E:text=June%2029%2C%202020%20by%20Piotr,by%20employing%20the%20feature%20selection.

Quintero, M., & LeBoulluec, A. (2018). Missing Data Imputation for Ordinal Data. *International Journal of Computer Applications*, *181*(5), 10–16. https://doi.org/10.5120/ijca2018917522

Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: the RPROP algorithm. *IEEE International Conference on Neural Networks*, 586–591. https://doi.org/10.1109/ICNN.1993.298623

Shichkin, A. v., Buevich, A. G., & Sergeev, A. P. (2018). Comparison of artificial neural network, random forest and random perceptron forest for forecasting the spatial impurity distribution. *AIP Conference Proceedings*, *1982*, 020005. https://doi.org/10.1063/1.5045411

Silipo, R., Adae, I., Hart, A., & Berthold, M. (2014). *Seven Techniques for Dimensionality Reduction*.

Singh, D., & Singh, B. (2020). Investigating the impact of data normalisation on classification performance. *Applied Soft Computing*, *97*, 105524. https://doi.org/10.1016/j.asoc.2019.105524

Sobreiro, P., Guedes-Carvalho, P., Santos, A., Pinheiro, P., & Gonçalves, C. (2021). Predicting Fitness Centre Dropout. *International Journal of Environmental Research and Public Health*, *18*(19), 10465. https://doi.org/10.3390/ijerph181910465

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

Steyerberg, E. W., & Harrell, F. E. (2016). Prediction models need appropriate internal, internal–external, and external validation. *Journal of Clinical Epidemiology*, *69*, 245–247. https://doi.org/10.1016/j.jclinepi.2015.04.005

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *Journal of Chemical Information and Computer Sciences*, *43*(6), 1947–1958. https://doi.org/10.1021/ci034160g

Templ, M., Kowarik, A., & Filzmoser, P. (2011). Iterative stepwise regression imputation using standard and robust methods. *Computational Statistics and Data Analysis*, *55*(10), 2793–2806. https://doi.org/10.1016/j.csda.2011.04.012

Tharwat, A. (2021). Classification assessment methods. *Applied Computing and Informatics*, *17*(1), 168–192. https://doi.org/10.1016/j.aci.2018.08.003

Twala, B., & Mekuria, F. (2013). Ensemble multisensor data using state-of-the-art classification methods. *2013 Africon*, 1–6. https://doi.org/10.1109/AFRCON.2013.6757711

Wang, S.-C. (2003). Artificial Neural Network. In *Interdisciplinary Computing in Java Programming* (pp. 81–100). Springer US. https://doi.org/10.1007/978-1-4615-0377-4_5

Weiss, G. M. (2001). *The Effect of Class Distribution on Classifier Learning: An Empirical Study*.

Xu, Y., & Goodacre, R. (2018). On Splitting Training and Validation Set: A Comparative Study of Cross-Validation, Bootstrap and Systematic Sampling for Estimating the Generalisation Performance of Supervised Learning. *Journal of Analysis and Testing*, *2*, 249–262. https://doi.org/10.1007/s41664-018-0068-2

Zheng, N. (2019). Fundamentals and Learning of Artificial Neural Networks. In *Learning in Energy-Efficient Neuromorphic Computing* (pp. 11–60). Wiley. https://doi.org/10.1002/9781119507369.ch2

Zheng, Z., Wei, W., & Hu, M. (2021). A Comparative Analysis of Delay Propagation on Departure and Arrival Flights for a Chinese Case Study. *Aerospace*, *8*(8), 212. https://doi.org/10.3390/aerospace8080212