Predictive Modeling of OTT Consumption Behaviour using Machine Learning

Vu Hoang Anh (Jennie) Nguyen

Department of Computing Science, Thompson Rivers University

COMP 4980: Special Topics - Machine Learning

Dr. Piper Jackson

December 4, 2023

Table of Contents

Abstract

This report provides a thorough evaluation of various machine learning algorithms applied to a multiclass classification task within the context of Over-the-Top (OTT) media services. The algorithms under examination include Multiclass Logistic Regression, Gaussian Naive Bayes, Support Vector Machines (SVM), XGBoost, AdaBoost, Gradient Boosting, Random Forests, and Multi-layer Perceptron (MLP). Each algorithm's performance was assessed using key metrics such as accuracy, precision, recall, and F1-score, with consistency evaluated through a 5-fold cross-validation. A confusion matrix was also employed to offer a detailed view of each model's performance. The findings reveal that Random Forests and XGBoost not only deliver the highest performance metrics but also demonstrate more consistent results across different runs. The report concludes with a discussion on the implications of these findings for network usage of OTT services and offers recommendations for future research and applications.

Keywords: machine learning algorithms, predictive modeling, over-the-top (OTT) media, user consumption behavior, network resource management

Introduction

In an era where the Internet has become an integral part of our daily lives, the volume of data being transmitted across networks is increasing at an unprecedented rate. This surge in data traffic necessitates effective network monitoring and analysis of consumption behavior. Such analysis is vital for network operators as it provides crucial insights into consumption trends, enabling them to devise new data plans tailored to specific user needs and gain a comprehensive understanding of the network's status. Over-the-top (OTT) media and communication services are significantly transforming Internet consumption patterns. These services, which deliver audio, video, and other media over the Internet without the involvement of traditional network operators in content control or distribution, are known for their high consumption of network resources. As a result, understanding and classifying OTT consumption behavior has become increasingly important. Classifying OTT consumption behavior allows network operators to understand user behavior patterns, manage network resources more effectively, and improve the quality of service. Furthermore, it can help in identifying potential network issues, predicting future network needs, and developing strategies for network expansion and optimization.

This paper aims to explore various machine learning algorithms that can serve as a guide for classifying users' OTT consumption behavior. The effectiveness of several machine learning algorithms - including Multiclass Logistic Regression, Gaussian Naive Bayes, Support Vector Machines, XGBoost, AdaBoost, Gradient Boosting, Random Forests, and Multi-layer Perceptron - will be evaluated on a data set containing real-world users' OTT application consumption behavior. The goal is to develop a robust and accurate model that can effectively generalize to unseen data and provide valuable insights into OTT consumption behavior. This could ultimately lead to improved network management and better service for users.

Data Description

The chosen data set, titled “User OTT Consumption Profile 2019,” provides a comprehensive snapshot of user behavior within the Universidad del Cauca network, captured over the course of April, May, and June 2019. Users are grouped into Low, Medium, and High Consumption categories based on their usage of various Over-the-top (OTT) media and communication services and applications. This data set is both large and detailed, with 1249 instances, each with 113 attributes. These attributes give us a clear picture of users’ consumption profiles, summarizing their behavior related to 56 identified OTT applications. Each user’s interaction with these applications is measured in terms of time spent (seconds) and data used (Bytes). The data set is available through OpenML in the ARFF format, which ensures uniformity and provides rich metadata, making automated processing easier. The OpenML Python API further enhances its usability by allowing for easy loading of the data. The data set is reliable, with no missing values or errors, ensuring the integrity of the data.

The “User OTT Consumption Profile 2019” data set is an excellent choice for several reasons. Firstly, the data set is rich in information, providing a solid foundation for a machine learning model to learn from. Secondly, the data set includes a clear target variable - the user consumption category (Low, Medium, High), making it suitable for supervised learning tasks. Thirdly, the data set mirrors real-world user behavior in the context of OTT services, a rapidly growing sector. This real-world relevance enhances the applicability and value of models trained on this data set. The classification of users into different consumption categories also can inform network planning and management strategies by providing network operators with a clearer understanding of consumption patterns within their user base. Moreover, user classification can enable more targeted marketing and customer service. For instance, ‘High Consumption’ users might be the ideal target for premium service offerings, while ‘Low Consumption’ users might benefit more from data-saving tips or lower-cost plans. Thus, the classification of users based on their consumption patterns can lead to more personalized and effective strategies, ultimately benefiting both the service providers and the users.

In an effort to streamline the analysis and optimize computational resources, a correlation matrix analysis was employed. This matrix, calculated using the Phi_K correlation coefficient, allowed for the identification of variables that exhibited a moderate to strong association with the target variable ‘cluster’. The Phi_K correlation coefficient is a versatile measure that can handle both numerical and categorical variables, making it particularly suitable for this mixed data set. It provides the strength of the association between variables, with values ranging from 0 (indicating no association) to 1 (indicating a strong association) (Baak et al., 2020). Variables that did not exhibit a sufficient degree of correlation (moderate to strong) with the ‘cluster’ were judiciously removed. This strategic simplification resulted in a more focused and manageable data set for subsequent analyses. This approach not only optimizes computational resources but also sharpens the analytical focus on the most relevant variables. As a result of this data refinement process, the data set now comprises 1249 instances, 14 salient features, and 1 target variable.

Data Analysis

In this section, a thorough examination of the data set is conducted. The exploration begins with an analysis of the descriptive statistics, which provides essential insights into the central tendency, dispersion, and shape of the data set’s distribution. A correlation matrix is then constructed to reveal the relationships between the various variables within the data set. Finally, a pairplot is generated, serving as a powerful visualization tool that enables the observation of pairwise relationships and distributions within the data set. This comprehensive analysis forms the foundation for the predictive modeling that follows.

Descriptive Statistics

Table 1 provides a comprehensive descriptive statistical analysis of the data set, which is crucial for understanding the underlying patterns in the data. The table includes:

- Mean: The average value of each feature, offering a glimpse into the typical behavior of users on each OTT application.
- Standard Deviation (std): This measures the variability or dispersion around the mean in the data set. A high standard deviation indicates a wide range of user behaviors, while a low standard deviation suggests more consistent usage patterns.
- Minimum (min) and Maximum (max): These values provide the range of each feature, indicating the extremes of user behavior on each application.
- 25%, 50%, and 75%: These are the first quartile, median, and third quartile, respectively. They provide a sense of the data distribution and can help identify outliers.

Table 1

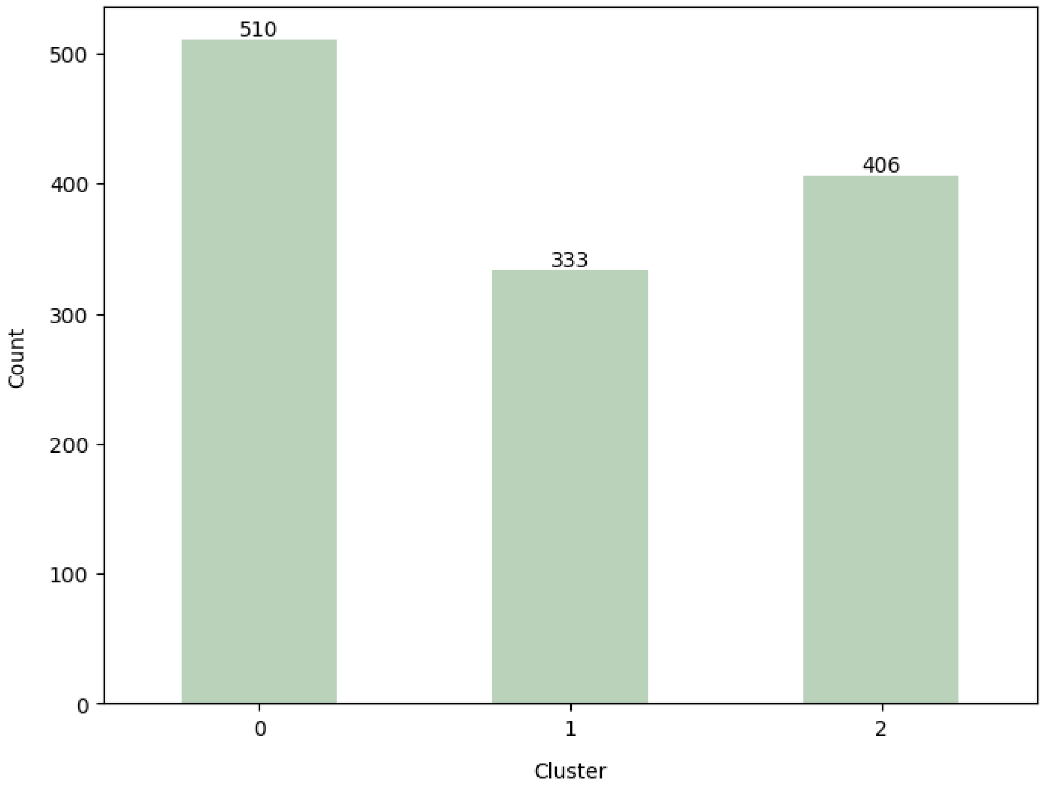
Descriptive statistics of the data set

Variable	mean	std	min	25%	50%	75%	max
Amazon_time_occupation	9890.37	13376.78	0	1241.27	4901.86	13073.46	83694.48
AmazonVideo_time_occupation	770.18	1391.82	0	0	203.41	792.88	7674.31
AppleiCloud_time_occupation	255.52	591.01	0	0	0	189.44	4293.15
GMail_time_occupation	4959.77	6493.04	0	516.85	2347.85	6495.07	30619.55
Google_time_occupation	101346.03	123426.27	0	19828.61	54509.34	130573.54	691934.84
GoogleDrive_time_occupation	840.47	1279.33	0	0.29	324.83	1172.73	8792.22
GoogleServices_time_occupation	12897.82	12748.02	0	3509.54	9922.87	17416.01	77854.65
HTTP_time_occupation	11587.09	15977.90	0	1674.35	5530.86	14991.77	132423.50
HTTP_Proxy_time_occupation	5734.95	18431.18	0	0	0.91	1484.82	184057.09
Skype_time_occupation	2214.55	4205.06	0	4.28	397.17	2296.33	27791.17
YouTube_time_occupation	6388.47	11712.53	0	398.20	2190.19	6499.06	105491.57
AppleiCloud_data_occupation	3755.83	6563.29	0	0	0	7236.98	40080.60
AppleiTunes_data_occupation	3300.78	6570.45	0	0	0	3193.30	65101.71
YouTube_data_occupation	554347.12	791101.96	0	12213.92	221000.12	752711.31	4483892

The ‘cluster’ variable in the data set is a categorical variable that represents the user consumption category. It has three unique values: 0, 1, and 2, which correspond to Low, Medium, and High consumption categories, respectively. Based on Figure 1, there are 510 instances of users falling into the Low consumption category (0). The Medium consumption category (1) comprises 333 instances, and the High consumption category (2) includes 406 instances. While there are slight differences in the number of instances for each category, the distribution is not heavily skewed towards any particular category. Therefore, the data set is relatively balanced.

Figure 1

Bar chart of 'cluster' distribution



Correlation Matrix

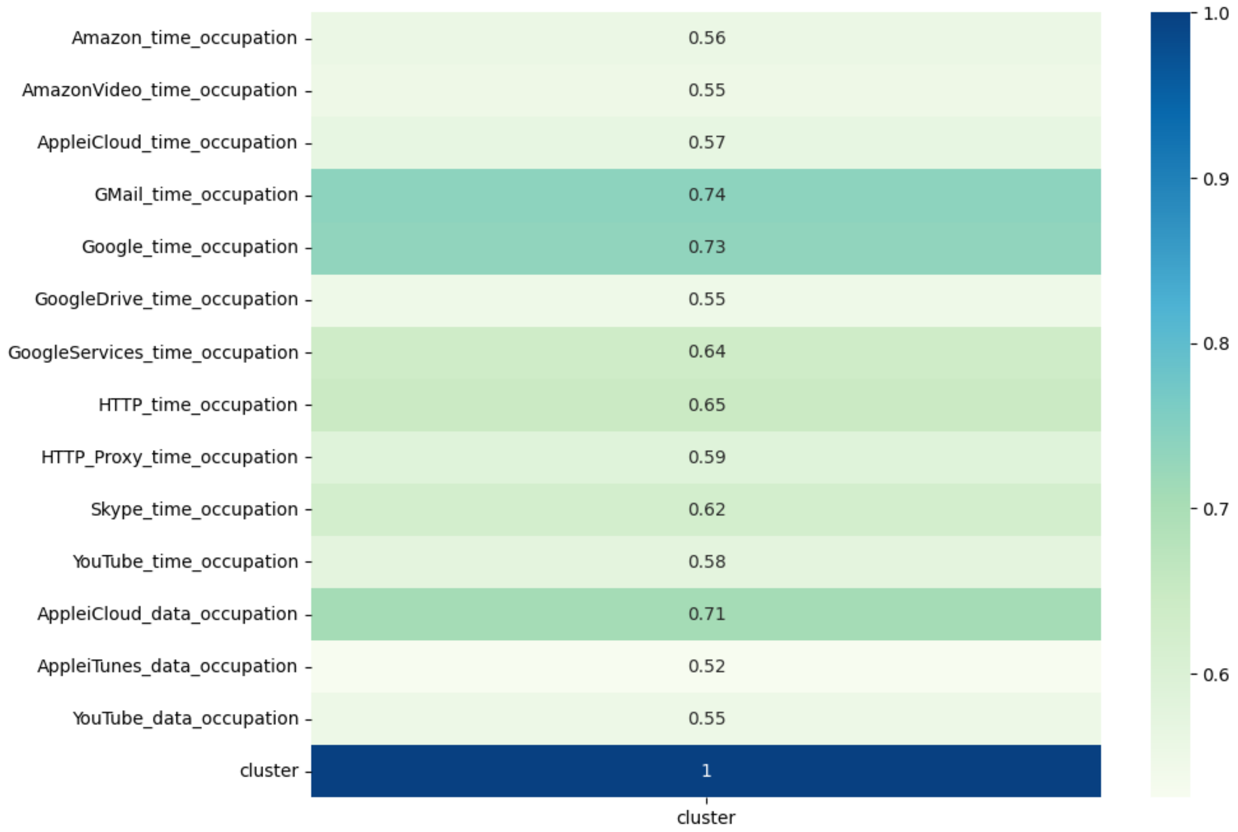
Phi_K Correlation Matrix

The Phi_K correlation coefficient is an innovative and practical metric that builds upon Pearson's test of independence, offering several enhancements. It interprets the contingency test statistic of two variables as originating from a tilted bi-variate normal distribution, where the tilt is understood as Phi_K. Phi_K offers several advantages over traditional coefficients. Firstly, it provides consistent results across categorical, ordinal, and interval variables, making it particularly beneficial when examining the correlation matrix of mixed-type variables. Secondly, Phi_K is capable of capturing non-linear dependencies, a significant improvement over traditional correlation coefficients like Pearson's, which are limited to linear relationships. Lastly, in the case of a bi-variate normal input distribution, Phi_K defaults to the Pearson correlation coefficient. This versatility allows Phi_K to serve as a universal correlation coefficient, capable of accommodating a broad spectrum of data types and distributions (*Phi_K Correlation Analyzer Library*, 2020). The Phi_K correlation coefficient operates within a range of [0; 1], where 0 signifies no association and +1 signifies a complete association. This range is similar to other correlation coefficients such as Pearson’s, making it easy to interpret. A value of 0 implies that there is no relationship between the two variables, while a value of 1 implies a perfect relationship. It’s crucial to note that a high Phi_K value indicates a strong association between variables, but it does not imply causation. (Baak et al., 2020).

Figure 2 presents the results of the Phi_K correlation analysis, which explores the associations between the target variable 'cluster' and the remaining variables in the data set. The analysis reveals that GMail_time_occupation (0.738385) and Google_time_occupation (0.734399) exhibit the highest correlation with the target variable ‘cluster’. This suggests a strong positive association, indicating that these features could be significant predictors in a model aiming to predict or classify ‘cluster’. AppleiCloud_data_occupation also demonstrates a high correlation value of 0.707961, suggesting its potential relevance in predicting ‘cluster’. The remaining features show moderate correlation values from 0.52 to 0.64, indicating a less strong, but still potentially meaningful, association with ‘cluster’.

Figure 2

Phi_K correlation matrix



Pearson's Correlation Matrix

The Pearson correlation coefficient is a statistical metric that is commonly used to quantify the degree and direction of the linear relationship between two quantitative variables. This coefficient is computed as the ratio of the covariance of the two variables to the product of their standard deviations. The value of this coefficient can range from -1 to 1, where:

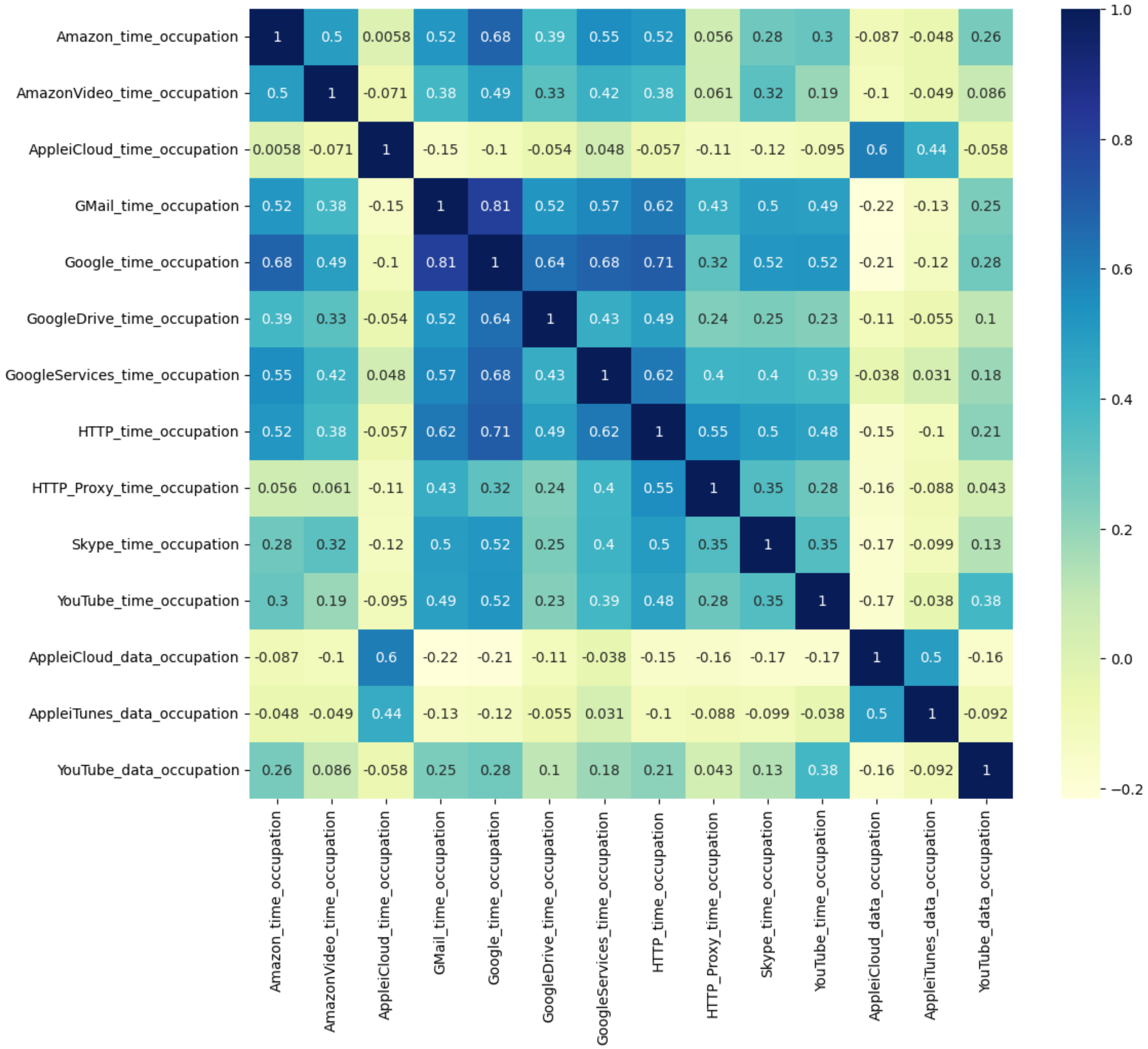
- A value of -1 signifies a perfect negative linear correlation between the two variables.
- A value of 0 signifies the absence of any linear correlation between the two variables.
- A value of 1 signifies a perfect positive linear correlation between the two variables.

The closer the coefficient is to zero, the weaker the relationship between the two variables. However, it’s crucial to remember that the correlation also does not necessarily imply causation (Turney, 2023).

Figure 3 displays the Pearson's correlation matrix results. However, it is important to note that the matrix does not include the target variable 'cluster' since 'cluster' is a categorical variable. The correlation matrix reveals several significant relationships among the variables. Firstly, a strong positive correlation of 0.68 exists between the time spent on Amazon and Google, suggesting that users who spend more time on Amazon are likely to spend a considerable amount of time on Google services as well. Similarly, a strong positive correlation of 0.81 is observed between the time spent on GMail and Google, indicating that heavy GMail users also tend to use various Google services extensively. Another interesting observation is the strong positive correlation of 0.64 between Google and Google Drive, implying that users who frequently engage with Google are likely to use Google Drive for their storage and file management needs. Lastly, a moderate positive correlation of 0.5 is seen between the time spent on Amazon and Amazon Video, suggesting that users who spend more time on Amazon also tend to spend a moderate amount of time streaming on Amazon Video. These correlations underscore the interrelated usage patterns among these variables, offering insights into user behavior and preferences.

Figure 3

Pearson's correlation matrix

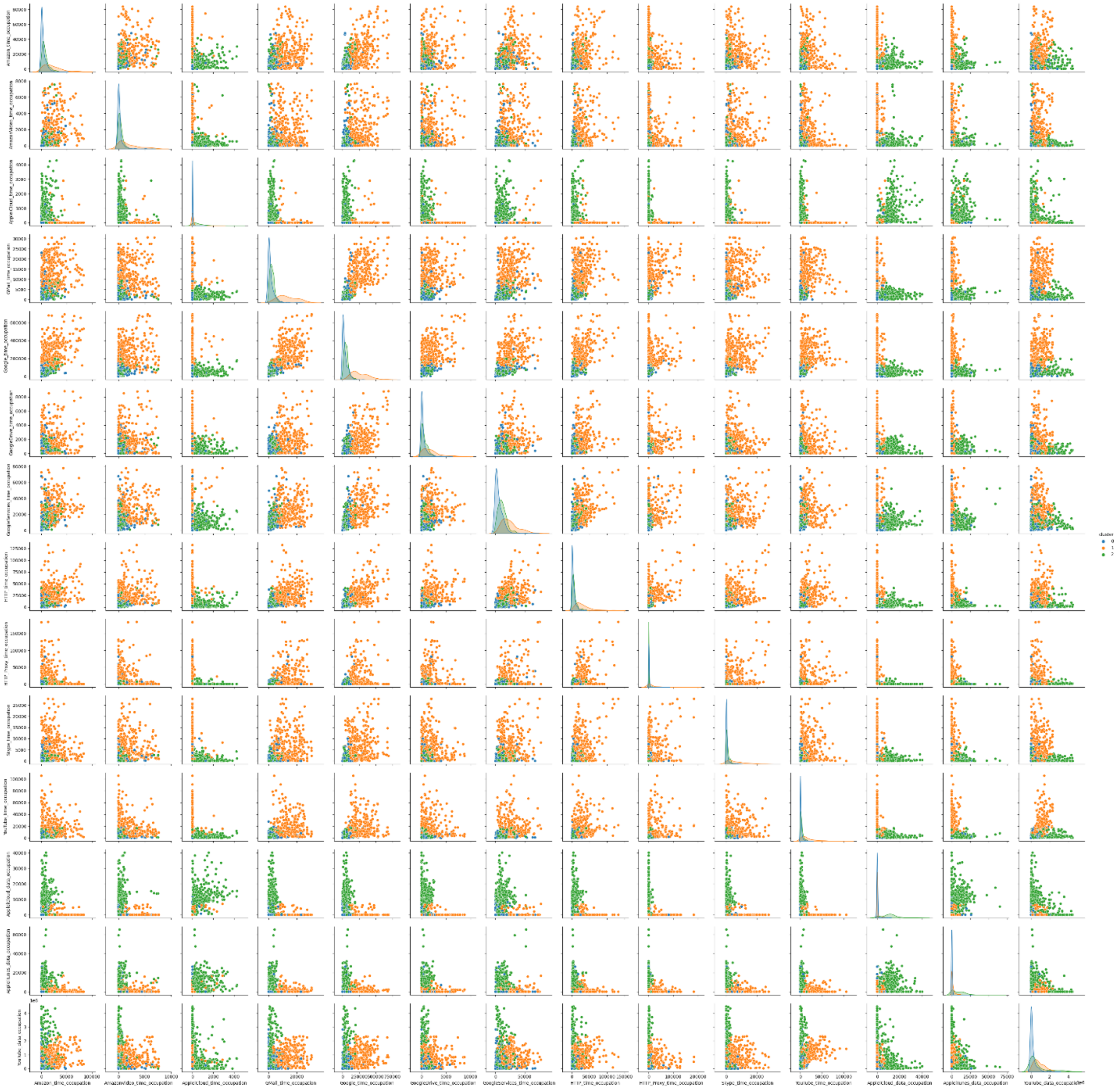


Pairplot

Figure 4 is a pairplot that provides a deep understanding of the relationships between different pairs of variables in the data set. It reveals that the 'cluster' variable, which categorizes user consumption into Low, Medium, and High, can be distinctly separated based on various combinations of variables. This indicates that these variables play a significant role in influencing the 'cluster' variable and are effective in classifying user consumption behavior.

Figure 4

Pairplot of variables



Data Exploration

In this section, a comprehensive exploration of the data set is undertaken using two key techniques: Principal Component Analysis (PCA) and Decision Tree analysis. These techniques serve as guides in uncovering the underlying structure of the data and the interactions between different variables, thereby setting the stage for the predictive modeling process that follows.

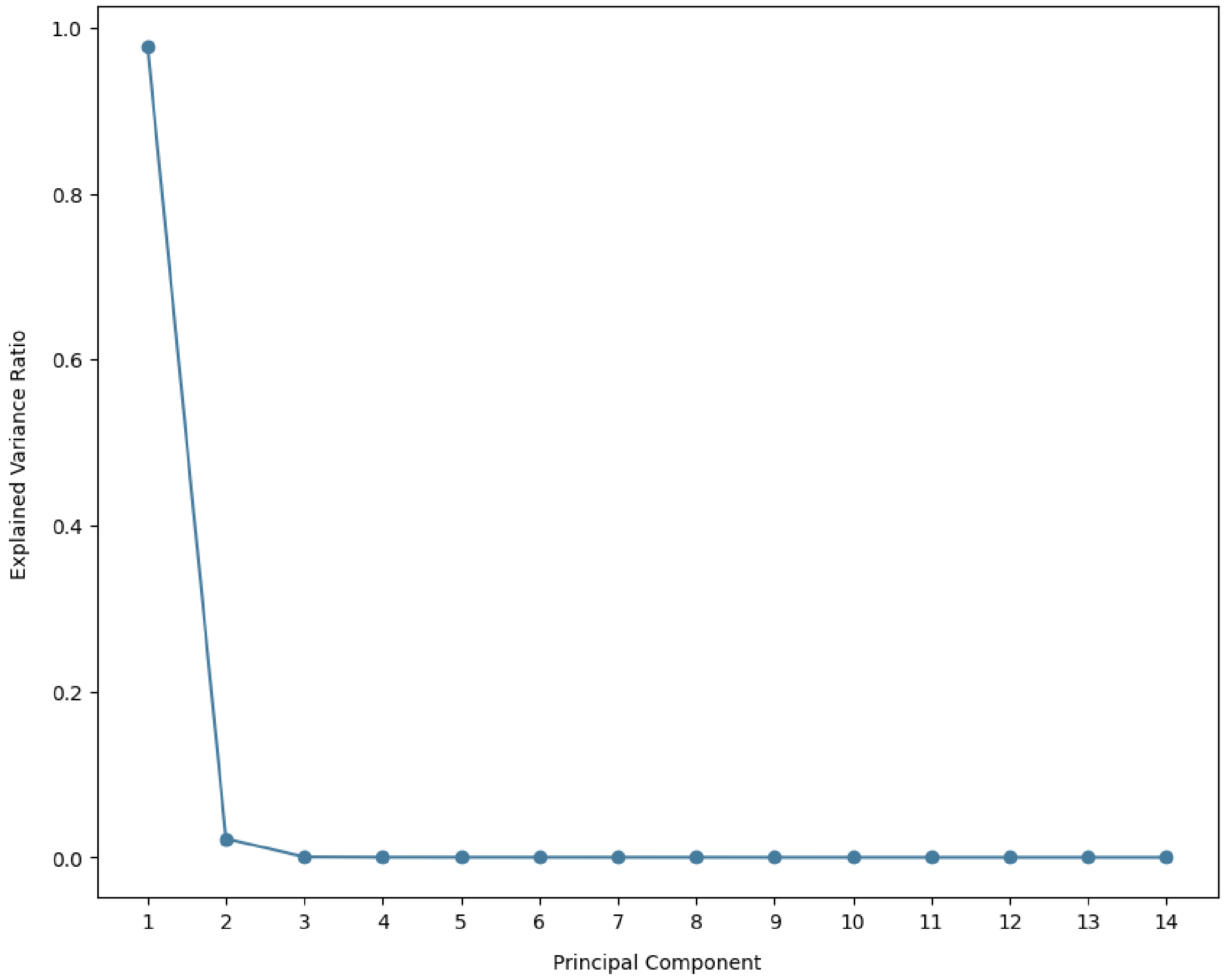
Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a statistical technique frequently employed to decrease the dimensionality of large data sets. By converting a large set of variables into a smaller one, PCA manages to retain most of the information from the original set. This reduction does come with a slight loss of accuracy, but the objective is to sacrifice a bit of precision for the sake of simplicity. PCA operates by pinpointing the directions (known as principal components) where the data exhibits the most variation, and subsequently creating new variables (also referred to as dimensions) that correspond to these directions. The first principal component is responsible for the maximum possible variance, the second principal component accounts for the second largest possible variance, and so on (Jaadi, 2023).

Figure 5 shows the cumulative sum of the explained variance ratio for each principal component. The graph suggests that a single principal component can account for roughly 96% of the total variance in the data. This implies that a significant portion of the variability in the original data set can be represented by just one component. When such a high percentage of variance can be explained by a single principal component, it indicates the presence of a robust underlying pattern or structure in the data. This pattern could be due to a shared factor or dimension that drives the variation across the variables.

Figure 5

Cumulative explained variance plot obtained from PCA



Decision Tree

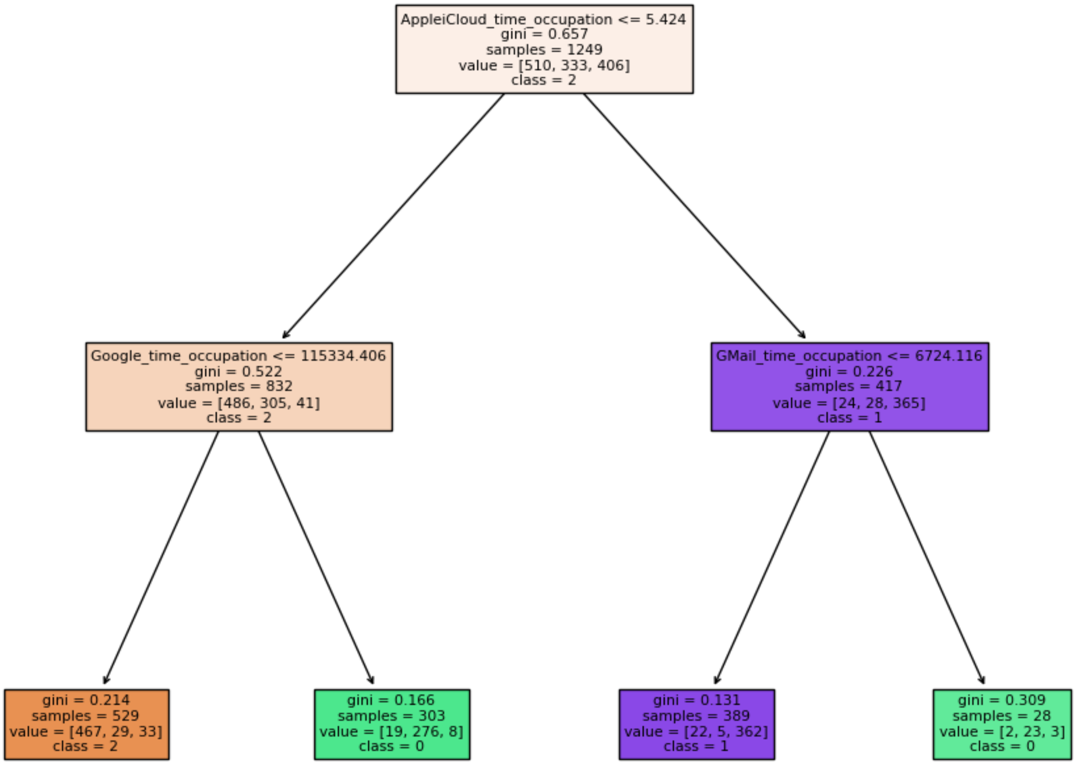
Decision tree analysis is a method in machine learning that is utilized for investigating data and making predictions in various situations. Its simplicity and ease of explanation make it one of the most user-friendly models, and its decision-making process is similar to that of humans (*Classification Using Decision Trees - a Comprehensive Tutorial*, n.d.).

Figure 6 presents a decision tree of depth 2, which serves as a guide for predicting the ‘cluster’ target variable’s class (0, 1, or 2). This prediction is based on three significant features: AppleiCloud_time_occupation, Google_time_occupation, and GMail_time_occupation.

According to the decision tree, these three features play a crucial role in determining the class. Among these, AppleiCloud_time_occupation is the most critical. Depending on its value, the model either directly predicts the class or proceeds to evaluate the next feature.

Figure 6

Decision tree of depth 2



Experimental Methods

The aim of this study is to construct a machine learning model that is capable of predicting the “cluster” variable using a set of 14 chosen input variables. The training of an algorithm, which can accurately classify new data into the correct cluster based on these input variables, is targeted. The ultimate goal to build a robust and precise model that can effectively generalize to unseen data. To achieve this, various machine learning techniques such as Multiclass Logistic Regression, Gaussian Naive Bayes, Support Vector Machines, XGBoost, AdaBoost, Gradient Boosting, Random Forests, and Multi-layer Perceptron are employed for the training and evaluation of the predictive model.

Neither standardization nor Principal Component Analysis (PCA) techniques will be applied. This decision is informed by preliminary experiments that suggest these preprocessing techniques negatively affect the algorithms' accuracy. As such, they will be excluded from the pipeline to optimize results.

The performance of the algorithms will be evaluated using a range of metrics, including accuracy, precision, recall, and the F1-score. The overall correctness of the predictions will be determined by calculating the accuracy. Precision, which is the ratio of correctly predicted positive instances to the total predicted positive instances, will be computed. Recall will be used to assess the model's ability to correctly identify positive instances from all actual positive instances. The F1-score, which is a combination of precision and recall, will be calculated to provide a balanced measure of the algorithms' performance.

A confusion matrix will also be used for a more detailed evaluation of the algorithms' performance. This matrix visually represents the algorithms' performance by highlighting true positives, true negatives, false positives, and false negatives. It will be instrumental in assessing the model's ability to correctly classify instances within each target class.

In addition, to ensure the robustness of the algorithms and to avoid overfitting, a cross-validation with 5 folds will be performed. This means the data set will be split into 5 parts, and the models will be trained and tested 5 times, each time with a different part of the data used as the test set. This process helps to give a more accurate measure of the algorithms' performance.

Multiclass Logistic Regression

Multiclass Logistic Regression is a variant of logistic regression that is designed to handle problems with more than two classes. While standard logistic regression is typically used for binary classification, multiclass logistic regression can handle multiple classes by transforming the problem into multiple binary classification problems (Brownlee, 2020b).

Gaussian Naive Bayes

Gaussian Naive Bayes is a specific type of Naive Bayes classifier that assumes a Gaussian distribution of data. It’s a straightforward yet powerful technique that can handle high-dimensional inputs and can be used to solve complex classification problems (GeeksforGeeks, 2023b).

Support Vector Machines

Support Vector Machines (SVMs) are robust machine learning algorithms that can be used for tasks ranging from linear or nonlinear classification to regression and outlier detection. They are particularly effective in situations where the number of dimensions exceeds the number of samples (GeeksforGeeks, 2023a).

AdaBoost

AdaBoost, or Adaptive Boosting, is a popular boosting algorithm in machine learning that enhances the accuracy of binary classification models. It achieves this by combining multiple weak learning algorithms to create a strong learner that can make precise predictions. It can also be extended to handle multiclass classification problems (Prabhakaran, 2023).

Gradient Boosting

Gradient Boosting is a machine learning technique that creates a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It’s used in various regression and classification tasks and is known for its ability to make few assumptions about the data (Brownlee, 2020a).

XGBoost

XGBoost, or eXtreme Gradient Boosting, is a supervised learning algorithm that is widely used for regression and classification tasks on large data sets. It builds shallow decision trees sequentially to provide accurate results and employs a highly scalable training method that prevents overfitting. (*How to Create a Classification Model Using XGBoost in Python*, 2021).

Random Forests

Random Forests is an ensemble learning method that constructs multiple decision trees during training. For classification tasks, the class selected by the majority of the trees is the output of the random forest (*What Is Random Forest?*, n.d.).

Multi-layer Perceptron (MLP)

MLP is a kind of neural network that consists of interconnected neurons arranged in input, hidden, and output layers. It's capable of learning complex patterns and can perform tasks such as classification and regression by adjusting its parameters during training (GeeksforGeeks, 2023b).

Results Discussion & Future Works

Table 2 presents the performance metrics of different experimental algorithms. Analyzing the table, it can be seen that:

- Random Forests, XGBoost, and Gradient Boosting all have the highest accuracy, precision, recall, and F1-score of 0.97. However, when we consider the standard deviation, Random Forests and XGBoost show a slightly more consistent performance with a standard deviation of +/- 0.02 compared to Gradient Boosting's +/- 0.03.
- Gaussian Naive Bayes and AdaBoost both have an accuracy, precision, and F1-score of 0.93. Gaussian Naive Bayes has a slightly higher Recall of 0.94. However, AdaBoost shows more variability in its performance with a standard deviation of +/- 0.07 compared to Gaussian Naive Bayes' +/- 0.03.

- MLP has an accuracy and F1-score of 0.79, precision of 0.8, and eecall of 0.82. However, it shows the most variability among all models with a standard deviation of +/- 0.09.
- Multiclass Logistic Regression has an accuracy and F1-score of 0.78, precision of 0.8, and recall of 0.81. Its performance is quite consistent with a standard deviation of +/- 0.03.
- SVM has the lowest accuracy and F1-score of 0.7, precision of 0.72, and recall of 0.7. Its performance is quite consistent with a standard deviation of +/- 0.03.

Given these results, **Random Forests** and **XGBoost** could be considered the best methods as they not only have the highest performance metrics but also show more consistent results across different runs. The confusion matrix for these models (Figure 7 and Figure 8) also indicates high performance, with a high number of correctly classified instances for each class and a low number of misclassifications, further confirming their superior performance.

Table 2

Evaluation metrics for different machine learning methods

Method	Accuracy	Precision	Recall	F1-score
Random Forests	0.97 (+/- 0.02)	0.97	0.97	0.97
XGBoost	0.97 (+/- 0.02)	0.97	0.97	0.97
Gradient Boosting	0.97 (+/- 0.03)	0.97	0.97	0.97
Gaussian Naive Bayes	0.93 (+/- 0.03)	0.93	0.94	0.93
AdaBoost	0.93 (+/- 0.07)	0.93	0.93	0.93
MLP	0.79 (+/- 0.09)	0.8	0.82	0.79
Multiclass Logistic Regression	0.78 (+/- 0.03)	0.8	0.81	0.78
SVM	0.70 (+/- 0.03)	0.72	0.7	0.7

Figure 7

Random Forests' confusion matrix

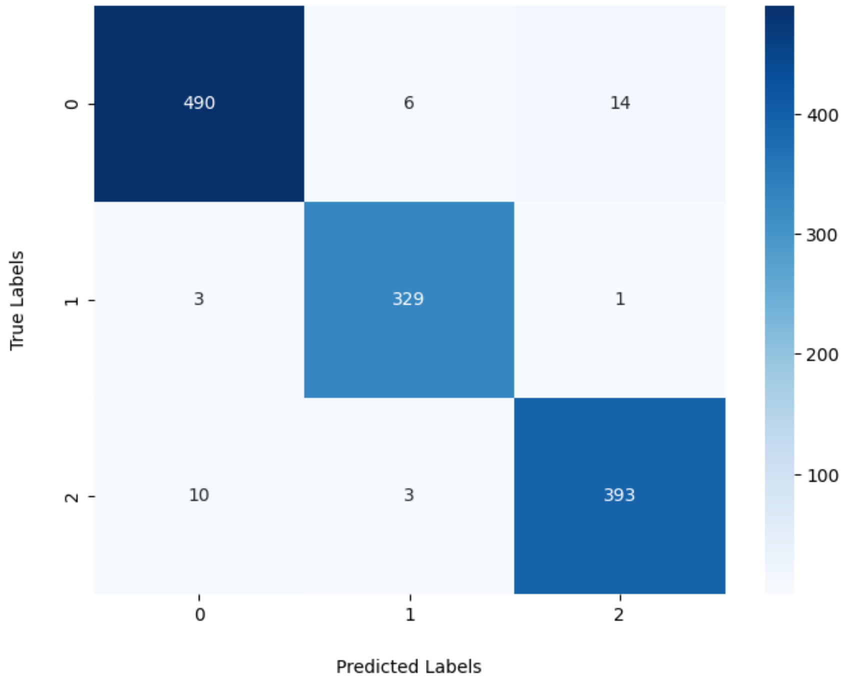
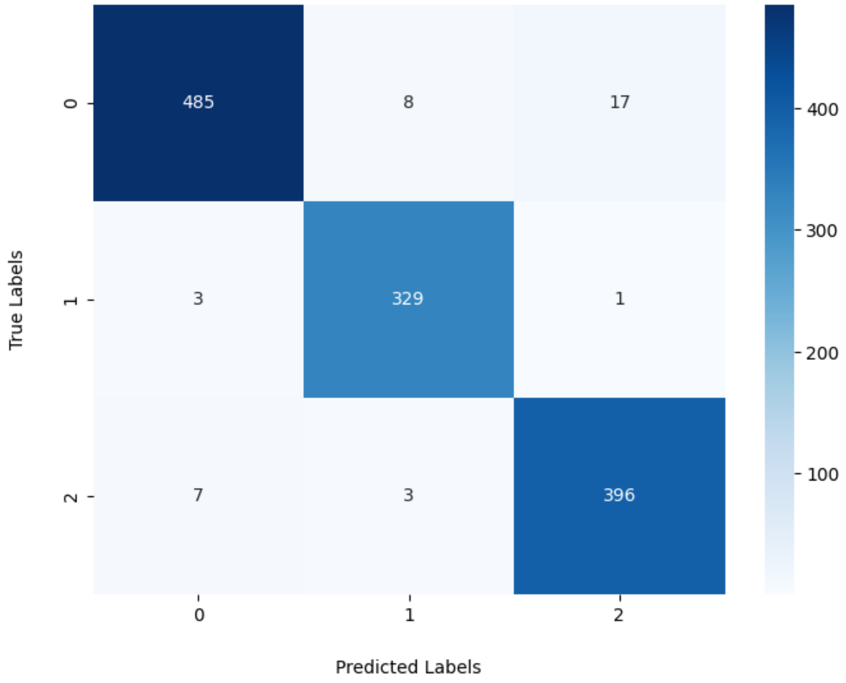


Figure 8

XGBoost's confusion matrix



The findings of this report open up new avenues for employing machine learning to understand and classify OTT consumption behavior. The superior performance of Random Forests and XGBoost models suggests that they could be effectively used for tasks such as personalized content recommendation, user behavior prediction, and targeted advertising. Future research could explore the application of these models in different areas of OTT services, experiment with new and different data sets, perform feature engineering, and tune hyperparameters to improve performance. Furthermore, the use of ensemble methods and improving model interpretability could also be valuable directions for future research. These findings contribute to the ongoing efforts in the field of machine learning to identify effective and reliable models for multiclass classification tasks.

Conclusion

In conclusion, this study conducts a comprehensive exploration of a data set with the aim of predicting the 'cluster' variable, representing user consumption categories, using machine learning techniques. The data set is thoroughly examined using descriptive statistics, a correlation matrix, and a pairplot. Principal Component Analysis (PCA) and Decision Tree analysis are employed in the data exploration phase. A machine learning model is constructed using 14 selected input variables, and various techniques, including Multiclass Logistic Regression, Gaussian Naive Bayes, Support Vector Machines, XGBoost, AdaBoost, Gradient Boosting, Random Forests, and Multi-layer Perceptron, are examined. The results indicate that the Random Forests and XGBoost algorithms outperform the others, achieving an accuracy of 0.97 (+/- 0.02). These insights can guide network operators in managing network resources more effectively, improving the quality of service, and developing new data plans tailored to specific user needs.

References

Baak, M. A., Koopman, R. F., Snoek, H., & Klous, S. (2020). A new correlation coefficient between categorical, ordinal and interval variables with Pearson characteristics. *Computational Statistics & Data Analysis*, *152*, 107043. https://doi.org/10.1016/j.csda.2020.107043

Brownlee, J. (2020a, August 14). *A gentle introduction to the gradient boosting algorithm for machine learning*. MachineLearningMastery.com. https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/

Brownlee, J. (2020b, August 31). *Multinomial logistic regression with Python*. MachineLearningMastery.com. https://machinelearningmastery.com/multinomial-logistic-regression-with-python/

Classification using decision trees - A comprehensive tutorial. (n.d.). Data Science Dojo. https://datasciencedojo.com/blog/classification-decision-trees/

GeeksforGeeks. (2023a, June 10). *Support Vector Machine SVM Algorithm*. https://www.geeksforgeeks.org/support-vector-machine-algorithm/

GeeksforGeeks. (2023b, October 11). *Classification Using Sklearn Multi layer Perceptron*. https://www.geeksforgeeks.org/classification-using-sklearn-multi-layer-perceptron/

GeeksforGeeks. (2023c, November 13). *Gaussian naive bayes*. https://www.geeksforgeeks.org/gaussian-naive-bayes/

How to create a classification model using XGBoost in Python. (2021, May 29). https://practicaldatascience.co.uk/machine-learning/how-to-create-a-classification-model-using-xgboost

Jaadi, Z. (2023, March 29). *A Step-by-Step Explanation of Principal Component Analysis (PCA)*. Built In. https://builtin.com/data-science/step-step-explanation-principal-component-analysis

Phi_K Correlation Analyzer Library. (2020, May). Read the Docs. Retrieved December 5, 2023, from https://phik.readthedocs.io/en/latest/

Prabhakaran, S. (2023, February 21). *AdaBoost – An introduction to AdaBoost*. Machine Learning Plus. https://www.machinelearningplus.com/machine-learning/introduction-to-adaboost/

Turney, S. (2023, June 22). *Pearson Correlation Coefficient (r) | Guide & Examples*. Scribbr. https://www.scribbr.com/statistics/pearson-correlation-coefficient/

What is Random Forest? (n.d.). IBM. https://www.ibm.com/topics/random-forest