MIND (Multimedia Artificial Intelligent Networking Database) Journal

MIND Journal | ISSN (p): 2528-0015 | ISSN (e): 2528-0902 | Vol. 9 | No. 2 | Halaman 220 - 234
DOI : https://doi.org/10.26760/mindjournal.v9i2.220-234 Desember 2024

Influence of Data Scaling and Train/Test Split Ratios on LightGBM Efficacy for Obesity Rate Prediction

NUR FITRIANTI FAHRUDIN, KURNIA RAMADHAN PUTRA, SOFIA UMAROH, GAMAS BLOORY LAUTAN

Information System, Institut Teknologi Nasional Bandung, Indonesia
Email: nurfitrianti@itenas.ac.id

Received 11 Oktober 2024 **|** *Revised* 2 Desember 2024 **|** *Accepted* 24 Desember 2024

ABSTRAK

Normalisasi adalah proses yang tidak dapat dilewatkan dalam data mining yang membantu menyesuaikan nilai atribut data ke skala yang sama. Dalam konteks data mining, perbedaan skala antar atribut dapat menyebabkan kesalahan dalam pemodelan atau interpretasi hasil. Penggunaan normalisasi dalam pra-pemrosesan masih diperdebatkan, terutama ketika menggunakan algoritma dari kelompok pohon keputusan. Penelitian ini membandingkan model dengan data yang dinormalisasi dan tidak dinormalisasi dengan menggunakan metode normalisasi, MinMaxScaler, MaxAbsScaler, dan RobustScaler. Hasil penelitian menunjukkan bahwa model LightGBM tanpa normalisasi memiliki tingkat akurasi sebesar 96,6 dalam mengklasifikasikan tingkat obesitas pada data saat ini. Tidak hanya normalisasi yang mempengaruhi hasil klasifikasi, tetapi juga jumlah rasio antara data pelatihan dan pengujian. Penelitian menunjukkan bahwa semakin besar persentase data yang digunakan untuk pelatihan, semakin tinggi tingkat akurasinya. Pada dataset obesitas, rasio 80:20 memiliki akurasi hingga 97%.

***Kata kunci**: Decision Tree, LightGBM, Obesitas, Data Mining, Klasifikasi*

ABSTRACT

Normalization is an essential process in data mining that helps adjust the values of data attributes to the same scale. In data mining, differences in attribute scales can lead to errors in modeling or interpreting results. Normalization in preprocessing is still debated, particularly when using algorithms from the decision tree family. This study compares models with normalized and non-normalized data using normalization methods such as MinMaxScaler, MaxAbsScaler, and RobustScaler. The results show that the LightGBM model without normalization achieved an accuracy rate of 96.6% in classifying obesity levels in the current dataset. Not only does normalization affect classification results, but the ratio between training and testing data also plays a role. The study indicates that the larger the percentage of data used for training, the higher the accuracy rate. In the obesity dataset, an 80:20 ratio resulted in an accuracy rate of up to 97%.

***Keywords**: Decision Tree, LightGBM, Obesity, Data Mining, Classification*

1. INTRODUCTION

The evolution of computer technology and machine learning algorithms influences the rapid development of technology in many aspects, including medical research **(Dwivedi, Srivastava, Dhar, & Singh, 2019) (Kumar & Singh, 2019) (Palanisamy & Thirunavukarasu, 2019)**. Machine learning techniques need computers to execute a learning process from data to make predictions. Highly accurate predictions make it easier for researchers to evaluate experiments quickly and accurately **(Saura, Herraez, & Reyes-Menendez, 2019).** One of the most commonly used machine learning techniques is decision trees. Decision trees can extract information from datasets into perceptive and understandable knowledge **(Yamada, Suzuki, Yokoi, & Takabayashi, 2003)**. The advantages of decision tree over other algorithms are noise immunity, low computational rate to generate models, and the capability to manage redundant functions **(Machado, Karray, & De Sousa, 2019)**.

However, decision trees have limitations in the availability of data with weak predictors, which can be overcome by using ensemble techniques **(Patel & Prajapati, 2018)**. This method is a learning algorithm built from multiple classifier models or predictive models. Ensembles can also be used to solve common machine-learning problems **(Nugraha, n.d.)**. In this case, the optimal value that the algorithm can achieve is within a certain finite range of values. Classification and prediction using well-treated ensemble algorithms can generally achieve higher accuracy and stability than using algorithms alone. The technique ordinarily used in the ensemble method is boosting and bagging **(Dogan & Birant, 2021)**. The ensemble method uses boosting by training groups of models one by one and combining all the models to make predictions. One of the most commonly used boosting methods is gradient boosting **(Sun, Wang, & Sun, 2020)**. This method uses a gradient descent boosting approach. The gradient descent mechanism is carried out to evaluate and create the next model. This algorithm has been further developed and a form of implementation is LightGBM or Light Gradient Boosting Machine **(A. Mohammed, Kadhem, Maisa, & Ali, 2021)**. The LightGBM algorithm is based on the decision tree calculation and could be a framework for machine learning created by Microsoft that's utilized to classify existing information **(Dwivedi et al., 2019)**. The LightGBM algorithm employs an interesting procedure called Gradient-Based One-Side Testing (GOSS) to channel the test information and discover the finest separator esteem. LightGBM speeds up processing times 20 times over conventional Gradient Boosting Decision Tree (GDBT) phases with the same accuracy **(Jin, Lu, Qin, Cheng, & Mao, 2020)**.

In making predictions on data, several steps need to be carried out, such as preprocessing, training, and testing. Preprocessing points to discretize data evacuate outliers and noise from data, coordinate data from different sources, handle fragmented data, and change data into comparable dynamic ranges. Data preprocessing is a critical step in achieving excellent classification execution, sometimes recently assessing information on machine learning calculations. One technique that can be done is data scaling or, commonly called normalization. Normalization is a step that includes changing highlights to the same extent so that the bigger numeric include values cannot rule the littler numeric include values. The normalization technique has been used by many researchers to improve classification performance in different application areas **(Shehadeh, Alshboul, Al Mamlook, & Hamedat, 2021)**. In distance-based algorithms such as k-Nearest Neighbors, data scaling can lead to improved performance and efficiency of the KNN algorithm **(Pagan, Zarlis, & Candra, 2023)**.

The decision tree method itself does not require feature normalization because the model only requires absolute values for branching **(Singh & Singh, 2020)**, so it will not significantly affect the accuracy value of the decision tree model **(Chen & Guestrin, 2016).** However,

research conducted by **(Santisteban Quiroz, 2022)** using the LightGBM method used normalization using MaxAbsScaler to minimize the impact of outliers. To see the implications of using normalization in the LightGBM method, this study compares models created using normalized and non-normalized data sets. In addition, in this study, we also compared the ratio of the data sets used in the training and testing processes.

In summary, this study aims to investigate the impact of data scaling techniques on the performance of the LightGBM algorithm using obesity dataset. The results can provide valuable insights into the applicability and effectiveness of various data scaling techniques and dataset ratios on the LightGBM algorithm.

2. METHODS

The process of determining the classification of obesity levels can be carried out in several stages, as shown in Figure 1. Broadly speaking, the system can be divided into several parts: Getting data set, exploration and pre-processing data, splitting data set into data training and data testing, modeling data, testing and performance measurement.

Figure 1. Research Methodology

2.1. Data Set

In this study, the dataset utilized is an obesity dataset, which has 17 components that can decide whether an individual is obese or not. To gather data, they presented factors as questions through a study that they connected to a gathering of individuals from Colombia, Mexico, and Peru. Within the chosen dataset, it was found that 2.111 individuals aged between 14 and 61 years took part in the study.

Table 1 describes the data used for determining the mathematical model. The data available in the dataset consists of numeric variables and categorical variables; therefore, the data collected in the existing dataset will be labeled encoding so that all the contents of the current dataset variables are turned into a numeric form so that it is easy to process. The methods and techniques utilized in this inquiry about experimentation prepare to refer to the Decision Trees-based method, specifically LightGBM

Table 1. Description of Obesity Dataset

Attribute	Description	Criteria
Gender	Gender	1 = Male 2 = Female
Age	Age	14 - 61
Height	Heigh	145 – 198
Weight	Weight	39 - 173
FHWO	Have a family with a history of being overweight	1 = No 2 = Yes
FAVC	Frequency of consumption of high-calorie foods	1 = No 2 = Yes

Attribute	Description	Criteria
FCVC	Frequency of the food you usually eat where there are vegetables in it	1 = Never 2 = Sometimes 3 = Always
NCP	Number of meals per day	1 = One 2 = Two 3 = Three 4 = More than three
CAEC	Rate of consumption of other foods between regular meals.	1 = No 2 = Sometimes 3 = Frequently
SMOKE	How often do you smoke	1 = No 2 = Yes
CH2O	How much to drink a day in liters	1 = Less than a liter 2 = Between 1 and 2L 3 = More than 2L
SCC	Monitoring how many calories we eat per day	1 = No 2 = Yes
FAF	How often do you do physical activity in a week	0 = I do not have 1 = 1 or 2 days 2 = 2 or 4 days 3 = 4 or 5 days
TUE	Time to use technology tools every day	0 = 0 – 2 hours 1 = 3 – 5 hours 2 = More than 5 hours
CALC	How often do you drink alcoholic beverages	1 = No 2 = Sometimes 3 = Frequently 4 = Always
MTRANS	Type of transportation	1 = Automobile 2 = Motorbike 3 = Bike 4 = Public Transportation 5 = Walking
NObeyesdad	Obesity attribute class	• Insufficient Weight • Normal Weight • Overweight Level 1 • Overweight Level 2 • Obesity Type 1 • Obesity Type 2 • Obesity Type 3

Based on Figure 2, the obesity dataset used based on NObeyesdad attributes shows that more than 350 people fall into the Overweight Level II category, more than 300 people and less than 350 people fall into the Obesity Type III category, 300 people fall into the Obesity Type II category, more than 200 people and less than 300 people who fall into the categories of Normal Weight, Overweight Level I, Overweight Level II, and Insufficient Weight.

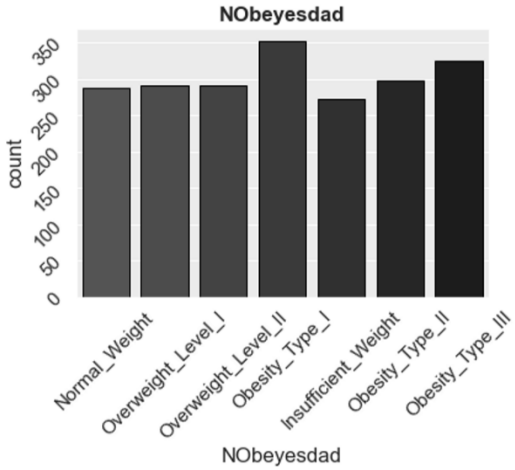

Figure 1. Obesity Dataset Distribution

2.2. Pre-Processing

This research starts with the pre-processing arrangement which is partitioned into two stages, specifically Categorical Encoding and additionally data normalization.

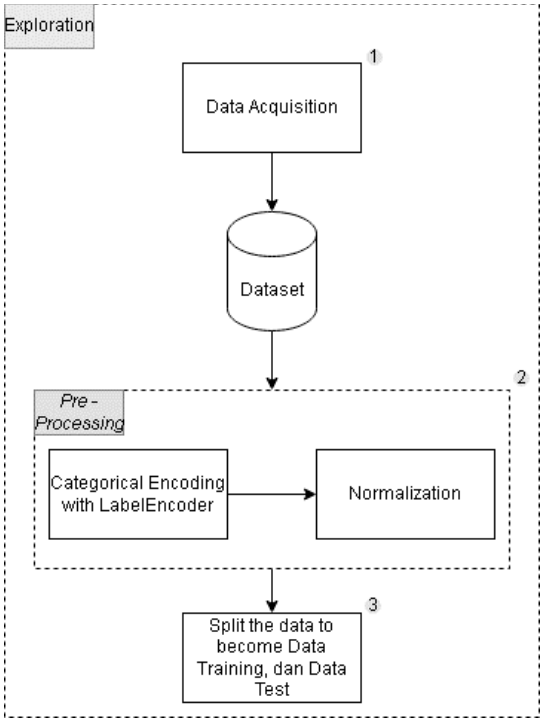

Figure 2. Exploration & Pre - Processing

The categorical encoding utilized is LabelEncoder, which gives name values with a run of values between 0 and N. This alter is used to encode values such as "Yes or No" and so on. At this stage, not all factors go through the categorical encoding preparation. Factors that go through this preparation are factors that have nominal value attribute types. In this obesity dataset, there are 9 properties to be changed to be specific sex, family_history_with_overweight, FAVC, CAEC, SMOKE, SCC, CALC, MTRANS, Nobeyesdad. As Figure 3 shows, feature scaling is not needed in this process because the nature of the decision tree itself is invariant, so it will not

affect the accuracy value of the model itself **(Patel & Prajapati, 2018)**. To discover the truth of this hypothesis, in this study, a comparison was made between datasets that went through the normalization handle previously and those that did not. This study utilized 4 distinctive normalization strategies such as MaxAbsScaler, MinMaxScaler, RobustScaler, and Normalize. After testing the normalized demonstration, we tried the demonstration without utilizing any normalization, and we compared which procedure delivered the finest esteem for the LightGBM strategy on the obesity dataset. An example of the difference between normalized and non-normalized data can be seen in Tables 2 and Table 3.The following is the formula of several normalization methods

$$\text{MinMaxScaler} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

$$\text{MaxAbsScaler} = \frac{x}{\max(x)} \quad (2)$$

$$\text{RobustScaler} = \frac{x - Q_1}{Q_3 - Q_1} \quad (3)$$

$$\text{Normalize} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (4)$$

In addition, we also use data splitting techniques to evaluate or test data that have been previously trained in the model. Data splitting methods split the information into two parts, specifically the training set and the test set. This time the scale used for the distribution of data splits is 80 : 20, 70 : 30, and 60 : 40. There were 3 trials of data splitting with different percentages of division to identify optimal proportions of the LightGBM model and the obesity dataset.

Table 2. Obesity Dataset Example before Normalization

Gender	Age	Height	Weight	FHWO	FAVC	FCVC	NCP	CAEC	SMOKE	CH2O	SCC	FAF	TUE	CALC	MTRANS
0	21	1.62	64	1	0	2	3	2	0	2	0	0	1	3	3
0	21	1.52	56	1	0	3	3	2	1	3	1	3	0	2	3
1	23	1.8	77	1	0	2	3	2	0	2	0	2	1	1	3
1	27	1.8	87	0	0	3	3	2	0	2	0	2	0	1	4
1	22	1.78	89.8	0	0	2	1	2	0	2	0	0	0	0	3

Table 3. Obesity Dataset Example after Normalization using Maxabsscaler

Gender	Age	Height	Weight	FHWO	FAVC	FCVC	NCP	CAEC	SMOKE	CH2O	SCC	FAF	TUE	CALC	MTRANS
0	0.34	0.81	0.36	1	0	0.67	0.75	0.67	0	0.67	0	0	0.5	1	0.75
0	0.34	0.76	0.32	1	0	1	0.75	0.67	1	1	1	1	0	0.67	0.75
1	0.37	0.90	0.44	1	0	0.67	0.75	0.67	0	0.67	0	0.67	0.5	0.33	0.75
1	0.44	0.90	0.50	0	0	1	0.75	0.67	0	0.68	0	0.67	0	0.33	1
1	0.36	0.89	0.51	0	0	0.67	0.25	0.67	0	0.69	0	0	0	0	0.75

2.3. Data Training and Testing

In this process, we build a LightGBM model which will later be trained and tested. Previously, data splitting produced two datasets, a training set and a test set, which were used to train and test the LightGBM model. LightGBM is then trained on the training set and evaluated on the validation set so we can view the best values from the model. This process runs 100 times. This is the standard iteration of the LightGBM model.

Separating the training and test set data results in four variables: x_train, y_train, x_test, y_test. The x_train and y_train attributes are used to train the aforementioned LightGBM model using the classifier. After training, predictions or model testing are carried out on the x_test value which will produce the y_pred value. The data split process is shown in Figure 4 below.

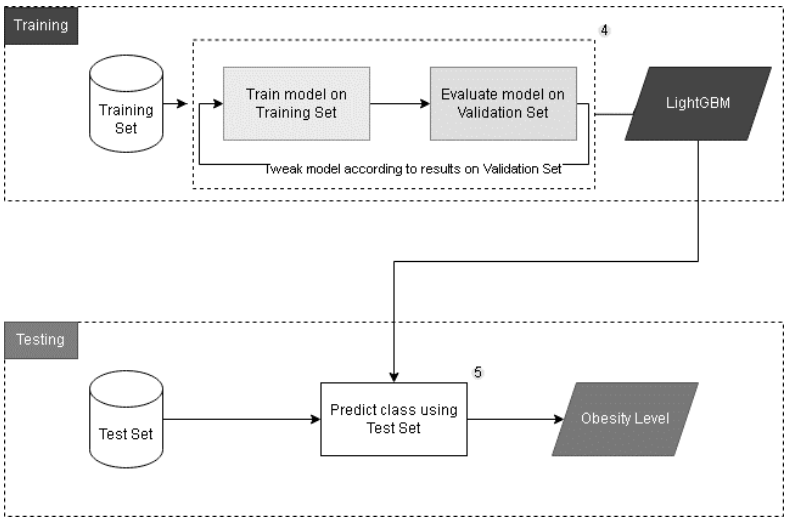

Figure 4. Data Training & Testing

2.4 LightGBM

Microsoft has developed an open source GBDT method called LightGBM. LightGBM is utilized to speed up preparing, diminish memory utilization, and combine advanced organize communication to optimize parallel learning called parallel voting DT calculation. Moreover, LightGBM employments the leaf-wise strategy to develop trees and discover a leaf with the biggest pick up of change to do the part.

By applying the LightGBM method using the Gradient-based One-Side Sampling method, we can find out which features have the most influence on obesity classification from the existing dataset. The LightGBM preparation uses 2 methods to extend the efficiency of the demonstration and additionally decrease memory utilization. These methods are Gradient-Based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). GOSS will test high-gradient tests and low-gradient tests will isolated tests with little angles, and center on tests with huge gradients. GOSS is done to get the best information to gain value. After that, EFB will be carried out to bundle the exclusive features. The following list is a features that affect the classification of obesity, sorted by their level of influence. Figure 5 show the process of LightGBM algorithm.

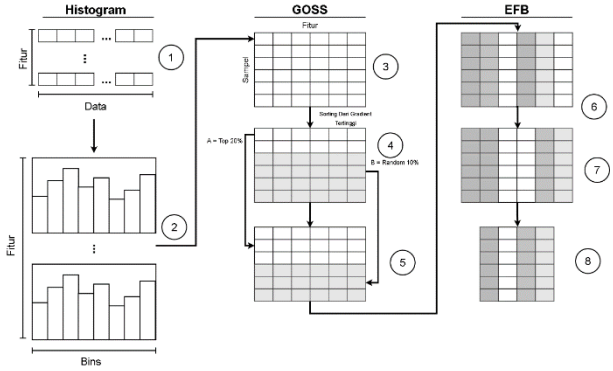

Figure 5. LightGBM Algorithm

LightGBM can be distinguished from other GBDT models by how the variation gains are calculated. Given the same input to the variance gain computation of LightGBM, a split is made considering weak and strong learners. The training instances are sorted in descending order according to the absolute value of the gradient.

$$\widetilde{V}_j(d) = \frac{1}{n}\left(\frac{\left(\sum x_i \in A_l\, g_i + \frac{1-a}{b}\sum x_i \in B_l g_i\right)^2}{n_l^j\,(d)} + \frac{\left(\sum x_i \in A_r\, g_i + \frac{1-a}{b}\sum x_i \in B_r g_i\right)^2}{n_r^j\,(d)}\right) \quad (5)$$

LightGBM provides better classification / predicition than other GBDT models due to the variance gain method including the tree growing method coupled with the acceptance of weak learners within the algorithm.

2.5 Feature Measurement

After going through the training and testing process, the LightGBM model goes through the existing performance measurement process. The tests carried out are accuracy, precision, recall, f1 – score, and ROC – AUC. The result of the assessment are used to measure the performance of the LightGBM method used.

$$\text{Accuracy} = \frac{(\text{TP}+\text{TN})}{(\text{TP}+\text{TN}+\text{FN}+\text{FP})}\ \text{x}\ 100\% \quad (6)$$

$$\text{Precision} = \frac{\text{TP}}{(\text{TP}+\text{FP})}\ \text{x}\ 100\% \quad (7)$$

$$\text{Recall} = \frac{\text{TP}}{(\text{TP}+\text{FN})}\ \text{x}\ 100\% \quad (8)$$

$$\text{F}-\text{Measure} = 2\ \text{x}\ \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (9)$$

3. RESULT

3.1. Feature Important

By applying the LightGBM strategy, which employments the Gradient-based One-Side Sampling method, ready to discover which highlights have the major influence on the

classification of obesity from the existing dataset. The taking after could be a list of highlights that impact the classification of obesity, sorted by level of impact.

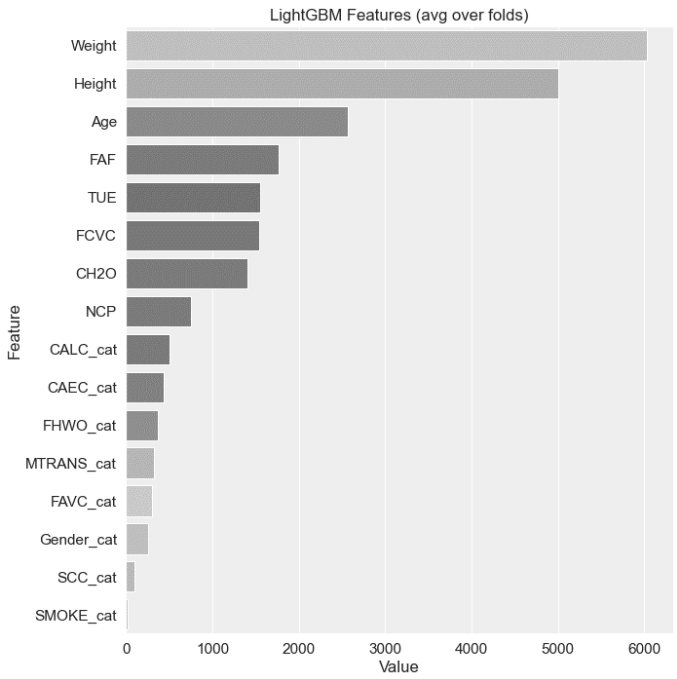

Figure 6. Feature Importance

Based on Figure 6, the Weight feature has the highest level of influence compared to other features, followed by the Height and Age features. This shows that the classification of obesity is strongly influenced by weight, height, and age.

LightGBM also has an algorithm called Exclusive Feature Bundling, which bundles the features in the obesity dataset. In the case of obesity data this time, EFB is run but does not do bundling because the existing features do not require bundling, so the total features used are still 16 attributes.

3.2 Performance Measurement

After the training process and validation were carried out, we utilized a confusion matrix to assess the performance of the LightGBM demonstration on the obesity dataset. A confusion matrix may be an execution measure for machine learning classification issues whose yield can be more than one class. It is exceptionally valuable for measuring-recall, accuracy, precision, f1-measure and most imperatively the ROC – AUC curve. There are 4 terms within the confusion matrix that depicts the classification of the execution measurement comes about, specifically True Negative (TN), False Positive (FP, True Positive (TP), and False Negative (FP). The confusion matrix uses a 7 x 7 confusion matrix because there are 7 target features in the dataset, as Figure 7 shows.

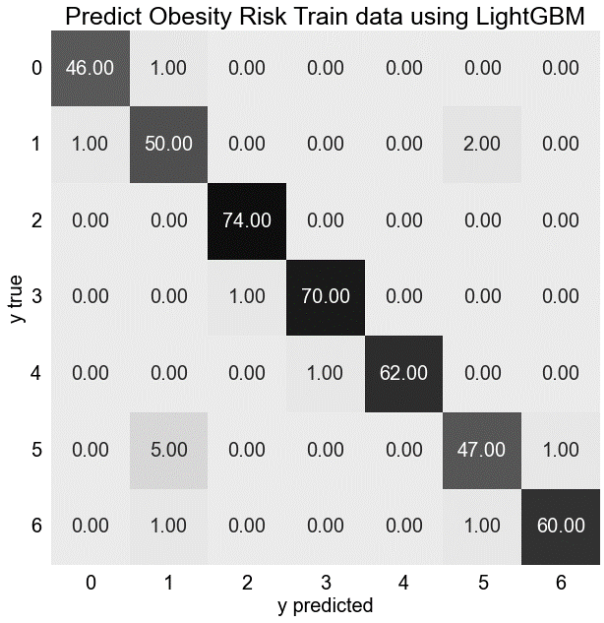

Figure 7. Confusion Matrix

The diagonal components speak to the number of focuses for which the predicted name is equal to the true label, whereas off-diagonal components are those that are mislabeled by the classifier. The higher the confusion matrix's diagonal values, the superior, demonstrating numerous correct forecasts.

The process of measuring the execution of the LightGBM demonstrates the use of the confusion matrix method by deciding the values of accuracy, precision, recall, and f1 – measure and also uses the ROC AUC curve.

Table 4. Normalization with Data Split 80 : 20

Model	Accuracy	Precision	Recall	F1-Measure	ROC - AUC
Without Normalization	96,78%	96,74%	96,71%	96,69%	99,85%
MinMaxScaler	96,73%	96,68%	96,64%	96,62%	99,84%
MaxAbsScaler	96,69%	96,64%	96,59%	96,63%	99,84%
RobustScaler	96,56%	96,50%	96,48%	96,70%	99,83%
Normalize	95,19%	95,09%	95,11%	94,95%	99,67%

Table 4 shows the results of a comparison between data that was not normalized and data that went through the normalization process.

a. Accuracy

Accuracy represents how well the prediction results are correct on the LightGbm model. Table 4 shows the prediction model resulting from data that has not gone through the normalization process has the highest accuracy value of 96.78% for obesity data, while data normalized using the normalize method has a lower accuracy of 95.19%. Based on the accuracy value, there is a difference between data that is not normalized with other methods, such as

MinMaxScaler = 0.058%, MaxAbsScaler = 0.098%, RobustScaler = 0.227%, Normalize = 1.596% lower than non-normalized data.

b. Precision

Precision is how good the model is at predicting a specific category. In Table 4 it can be seen that the best precision values for non-normalized and normalized data. Not normalized data has the highest precision value, namely 96.74%. Based on the precision value, there is a difference between data that is not normalized by other methods, such as MinMaxScaler = 0.068%, MaxAbsScaler = 0.099%, RobustScaler = 0.244%, Normalize = 1.650% lower than the data that is not normalized.

c. Recall

Recall checks how numerous Actual Positives our model captures by labeling it as Positive (True Positive). Based on this understanding, Recall becomes a model metric that can be utilized to choose the most excellent show when there are high costs related to false negatives. Not normalized data has the highest precision value, namely 96.71%. Based on the recall value, there is a difference between data that is not normalized with other methods, such as MinMaxScaler = 0.067%, MaxAbsScaler = 0.116%, RobustScaler = 0.233%, Normalize = 1.603% which is lower than data that is not normalized.

d. F1-Measure

In this obesity data, the distribution in each class is uneven. So, if the accuracy value does not work correctly, the F1 measurement can be used to handle it. F1-measure is an approach to combine the precision and recall measures of a classifier by averaging the harmonics evenly between the two. Based on the F1-Measure value, normalized data using RobustScaler has the highest accuracy, namely 96.70%, while non-normalized data has a value of 96.69%. The difference between normalized data using RobustScaler and non-normalized data is 0.01%. However, the F1-measure value for non-normalized data is higher when compared to other normalization methods, such as MinMaxScaler = 0.063%, MaxAbsScaler = 0.056%, and Normalize = 1.732% higher. Thus, it can be concluded that data without normalization is still superior to those that go through the normalization process.

e. ROC-AUC

After getting the value from the confusion matrix, we utilized the ROC – AUC curve to assess and validate the value produced from the confusion matrix. The ROC – AUC curve in the LightGBM model will produce 7 curves that represent each target feature in the obesity dataset used. Figure 8 shows the ROC – AUC curve on the LightGBM model.

AUC-ROC curves are performance indicators for classification problems at various threshold settings. ROC may be a likelihood curve and AUC speaks to the degree or degree of distinctness. This curve shows how well a show can differentiate between classes. Table 4 shows the prediction model resulting from data that has not gone through the normalization process has the highest AUC-ROC value of 99.85% for obesity data, while data normalized using the normalize method has a lower accuracy of 99.67%.

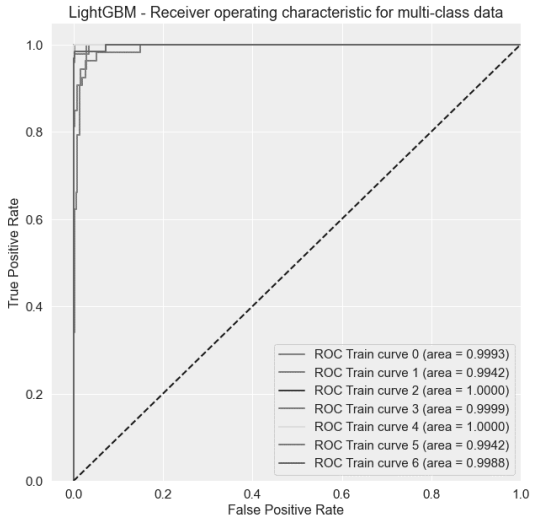

Figure 8. ROC – AUC Curve

Based on the AUC-ROC value, there is a difference between data that is not normalized by other methods, such as MinMaxScaler = 0.012%, MaxAbsScaler = 0.012%, RobustScaler = 0.024%, Normalize = 0.0181%, which is lower than non-normalized data. Even though the performance between normalized and non-normalized data is not too significant the difference, it can still be concluded that adding a normalization step to the LightGBM method is proven to reduce the model's performance in predicting class.

From Table 4, shows that for data splitting 80 : 20, the best method is without normalization which resulted in accuracy 0.967848, precision 0.96740, recall 0.9670864, and so on. Next, we carry out tests to find the best value from the composition of the splitting data. Specifically, the amount of training samples, which can too be deciphered as the training–testing range proportion, has a decisive impact on classification **(Pawluszek-Filipiak & Borkowski, 2020)**.

Based on the results displayed in Table 5, it can be concluded that the leading comes about are accomplished with a preparing information proportion of 80% with an F1 measure value = 96.69%.

Table 5. Comparison of All Data Splitting

Model	Accuracy	Precision	Recall	F1-Measure	ROC - AUC
80:20	96,78%	96,74%	96,71%	96,69%	99,85%
70:30	96,36%	96,30%	96,27%	96,25%	99,81%
60:40	95,81%	95,74%	95,74%	95,70%	99,77%

In Table 5, the accuracy value of the 60:40 splitting data has a difference of 0.98% lower than the 80:20 splitting data, while the 70:30 splitting data has a difference of 0.43%. Based on the accuracy value, it can be concluded that the higher the training data value, the higher the accuracy value. The precision value in Table 5 shows that the 80:20 splitting data has the best value, with a value of 96.74%. When compared with the 60:40 splitting data, the precision value decreased by 1.00%, and with the 70:30 splitting data, it decreased by 0.45%. Based on the precision value, it can be concluded that the higher the percentage of training data,

the higher the accuracy value. Based on the recall value of data splitting 80:20 has the highest value, namely 96.71%. Meanwhile, the 70:30 splitting data decreased by 0.44% and the 60:40 splitting data experienced a 0.97% decrease. Based on the value of F1-Measure data splitting 80:20 also has the highest value of 96.69%. Meanwhile, the 70:30 splitting data experienced a decrease of 0.44% and the splitting data experienced a decrease of 0.98%. ROC and AUC values are still the same as other values, data splitting with a percentage of 80:20 has a value of 99.85%.

It can be concluded based on the values of accuracy, precision, recall, F1-measure and ROC AUC, the greater the composition of the training data, the better the resulting model. This is in accordance with what was conveyed by Anita R, where they recommend using a training/testing separation ratio of 80%/20%, especially for larger data sets **(Rácz, Bajusz, & Héberger, 2021)**.

4. CONCLUSIONS

The result of the obesity classification for the LightGBM model have a probability average value of 0.998509, so it can be stated that the existing LightGBM model can distinguish between positive and negative classes. The best data split technique used for the LightGBM model applied to the obesity dataset is a ratio of 80 : 20. This ratio can also be applied to other datasets, which can be concluded that the more datasets used for training, the better the accuracy. The LightGBM model which is applied to the obesity dataset does not require the normalization method because the value generated from training the LightGBM model without using normalization is higher than a model training with normalization. Actually this is because in obesity datasets that have variables of similar scale the resulting decision tree structure is not affected by the scale of the variable, normalization may not have a significant impact on the performance of the LightGBM method. From these results it can be concluded that the existing LightGBM model can distinguish between the positive and the negative classes that exist well with an average probability value of 0.998509. The model with the highest classification accuracy is then converted into Predictive Model Markup Language (PMML). PMML serves as a bridge for information exchange across platforms, allowing the generated predictive model to be used in other programming languages such as Java. Subsequently, an application will be developed to detect whether someone has the potential for obesity or not.

ACKNOWLEDGEMENT

We thank LPPM Itenas Bandung for providing funding to carry out this research. Access to libraries, laboratories and computing infrastructure significantly contributes to the success of our research.

REFERENCES

A. Mohammed, M., Kadhem, S., Maisa, & Ali, A. (2021). *Insider Attacker Detection Using Light Gradient Boosting Machine*. *1*(February), 48–66.

Dogan, A., & Birant, D. (2021). Machine learning and data mining in manufacturing. *Expert Systems with Applications*, *166*, 114060. https://doi.org/10.1016/j.eswa.2020.114060

Dwivedi, A. D., Srivastava, G., Dhar, S., & Singh, R. (2019). A decentralized privacy-preserving

healthcare blockchain for IoT. *Sensors (Switzerland)*, *19*(2), 1–17. https://doi.org/10.3390/s19020326

Jin, D., Lu, Y., Qin, J., Cheng, Z., & Mao, Z. (2020). SwiftIDS: Real-time intrusion detection system based on LightGBM and parallel intrusion detection mechanism. *Computers and Security*, *97*, 101984. https://doi.org/10.1016/j.cose.2020.101984

Kumar, S., & Singh, M. (2019). Big data analytics for healthcare industry: Impact, applications, and tools. *Big Data Mining and Analytics*, *2*(1), 48–57. https://doi.org/10.26599/BDMA.2018.9020031

Machado, M. R., Karray, S., & De Sousa, I. T. (2019). LightGBM: An effective decision tree gradient boosting method to predict customer loyalty in the finance industry. *14th International Conference on Computer Science and Education, ICCSE 2019*, (Iccse), 1111–1116. https://doi.org/10.1109/ICCSE.2019.8845529

Nugraha, W. (n.d.). *Prediksi penyakit jantung cardiovascular menggunakan model algoritma klasifikasi*.

Pagan, M., Zarlis, M., & Candra, A. (2023). Investigating the impact of data scaling on the k-nearest neighbor algorithm. *Computer Science and Information Technologies*, *4*(2), 135–142. https://doi.org/10.11591/csit.v4i2.pp135-142

Palanisamy, V., & Thirunavukarasu, R. (2019). Implications of big data analytics in developing healthcare frameworks – A review. *Journal of King Saud University - Computer and Information Sciences*, *31*(4), 415–425. https://doi.org/10.1016/j.jksuci.2017.12.007

Patel, H. H., & Prajapati, P. (2018). Study and Analysis of Decision Tree Based Classification Algorithms. *International Journal of Computer Sciences and Engineering*, *6*(10), 74–78. https://doi.org/10.26438/ijcse/v6i10.7478

Pawluszek-Filipiak, K., & Borkowski, A. (2020). On the importance of train-test split ratio of datasets in automatic landslide detection by supervised classification. *Remote Sensing*, *12*(18). https://doi.org/10.3390/rs12183054

Rácz, A., Bajusz, D., & Héberger, K. (2021). Effect of dataset size and train/test split ratios in qsar/qspr multiclass classification. *Molecules*, *26*(4), 1–16. https://doi.org/10.3390/molecules26041111

Santisteban Quiroz, J. P. (2022). Estimation of obesity levels based on dietary habits and condition physical using computational intelligence. *Informatics in Medicine Unlocked*, *29*(July 2021), 100901. https://doi.org/10.1016/j.imu.2022.100901

Saura, J. R., Herraez, B. R., & Reyes-Menendez, A. (2019). Comparing a traditional approach for financial brand communication analysis with a big data analytics technique. *IEEE*

Access, *7*, 37100–37108. https://doi.org/10.1109/ACCESS.2019.2905301

Shehadeh, A., Alshboul, O., Al Mamlook, R. E., & Hamedat, O. (2021). Machine learning models for predicting the residual value of heavy construction equipment: An evaluation of modified decision tree, LightGBM, and XGBoost regression. *Automation in Construction*, *129*(June), 103827. https://doi.org/10.1016/j.autcon.2021.103827

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

Sun, Y., Wang, S., & Sun, X. (2020). Estimating neighbourhood-level prevalence of adult obesity by socio-economic, behavioural and built environment factors in New York City. *Public Health*, *186*, 57–62. https://doi.org/10.1016/j.puhe.2020.05.003

Yamada, Y., Suzuki, E., Yokoi, H., & Takabayashi, K. (2003). Decision-tree Induction from Time-series Data Based on a Standard-example Split Test. *Proceedings, Twentieth International Conference on Machine Learning*, *2*, 840–847.