International Journal of Internet, Broadcasting and Communication Vol. 16 No.4 442-459 (2024)
http://dx.doi.org/10.7236/IJIBC.2024.16.4.442

IJIBC 24-4-49

A Study on the Influence of Changing Data Classification Criteria Depending on the Correlation of Variables

Seung-Jae Kim

Assistant Professor, Department of Convergence Honam University
ksj0909@honam.ac.kr

Abstract

Currently, many industrial fields are pursuing research and development toward a hyper-connected society. However, as we become a hyper-connected society that perceives virtual reality as if it were reality, accurate classification of data to recognize objects, emotions and facial expressions must be accompanied. In other words, only when data meaning objects, emotions, and facial expressions are accurately classified will reliability of cognition and recognition be obtained not only in the physical world but also in a hyper-connected society. In addition, errors in perception and recognition of objects, emotions, and facial expressions can be reduced through big data analysis, and it will be protected from secondary incidents and damages. Therefore, in this study, we try to find out whether the classification of data is well done in the stage where AI with automatic cognition ability recognizes and recognizes objects, emotions, and facial expressions, and whether the data classified according to characteristics is a reliable classification result. In the experiment, when classifying data using a decision tree, we plan to conduct a study to find out whether the classification criteria of the data affect the classification criteria according to the degree of correlation between variables.

Keywords: *Data Classification, Correlation of Variable, Decision Tree, Python Program, R Program, Data Analysis*

1. Introduction

The 4th industry aims to create a hyper-connected society with the goal of building an optimal management system for each field's characteristics by connecting all fields of society centered on AI [1,2]. The fields of research and development aimed at this hyper-connected society include national R&D, large corporations, government offices, universities and research institutes, and the core technologies can be said to be all algorithms utilizing ICT technologies such as AI, Big-data, IOT, and Metaverse. These technologies come together to realize a hyper-connected society, and within it, a convergence world without boundaries between ‘virtual to reality’ and ‘reality to virtual’ is realized. Attempts to connect various fields into a hyperconnected society using these various AI implementation technologies are continuing, and the trend is to expand around countries that use ICT technology worldwide. As a step-by-step process to realize a complete hyper-connected society, the current era discovers new models by using big data analysis based on data collected in specific fields [3-5]. There are increasing attempts to present future visions by inferring and predicting new values with new models [6,7]. In addition, by using these technological capabilities,

Manuscript Received: October. 24, 2024 / Revised: October. 31, 2024 / Accepted: November. 5, 2024
Corresponding Author: ksj0909@honam.ac.kr
Assistant Professor, Department of Convergence Honam University

individual small business owners, commercial people who want to open a business, and people preparing for employment are trying to maximize their income, profits, and self-empowerment by using the core technologies of the 4th industry.

However, the more hyper-connected society recognizes virtuality like reality, the informativity of data for recognizing objects, emotions, and expressions must also be accompanied by accurate data classification [8-10]. In other words, only when data meaning objects, emotions, and expressions are accurately classified, errors in recognition and recognition of objects, emotions, and expressions can be reduced through big data analysis and protected from secondary events and damages [11]. In order to realize a reliable hyper-connected society, when decisions must be made by recognizing and recognizing tangible and intangible objects, reliable decisions must be made by learning and training based on a large amount of data. In other words, reliable decisions must be made through big data analysis. In order to obtain the reliability and sophistication of statistics as a result of big data analysis, it is necessary to analyze each variable in consideration of the meaning of each variable, the correlation between the variables, and multicollinearity. If the data is classified differently from the hypothesis test from the beginning, unreliable results will be obtained even if the analysis is done well. In other words, before analyzing big data, the data must be properly classified to suit the purpose of analysis.

In this study, we aim to find out whether AI with automatic recognition ability recognizes objects, emotions, and facial expressions and classifies data well at the recognition stage, and whether the data classified according to characteristics is a reliable classification result. The assumption of the experiment is to conduct research to determine whether the data classification criteria affect the classification criteria depending on the degree of correlation between variables when classifying data using a decision tree. Decision Tree (DT) [12] and Correlation Analysis (CA) [13] are one of the Machine Learning (ML) [14] techniques among AI implementation technologies. DT is a technique that sets data classification criteria based on the characteristics of the data and classifies data according to the characteristics. CA is an analysis technique that creates indicators to determine the degree of correlation between variables involved in the decision-making stage when making decisions through big data analysis. In this experiment, we want to find out whether the data classified by DT have a correlation with each other. Therefore, this experiment could be a criterion for examining the accuracy and reliability of data classification to recognize and recognize objects, emotions, and expressions in the realization of a hyperconnected society based on AI.

2. Data Classification and Correlation

Data classification analysis and correlation analysis are one of the machine learning (ML) techniques, and classification analysis can classify data by grouping the collected data with different characteristics. Correlation analysis is an analysis technique that allows you to determine whether there is a correlation between data. Today, there are many statistical tools such as SPSS and SAS for DT analysis and correlation analysis, but computer languages are used a lot from the perspective of implementing AI technology. Computer languages used include R Program [15] and Python Program [16].

2.1 Decision Tree(DT) Definition

Decision trees (DTs) have rules that are relatively quick, simple, and easy to understand compared to other classification analysis techniques. DT is a technique that can classify collected data into several groups and classify decision rules that appear between variables using a tree structure. For elaborate data classification, a step-by-step analysis process is performed, and the analysis process consists of five steps.

DT uses the concept of impurity to select branching criteria in a decision tree, which refers to the complexity of the data. In other words, it means to what extent different data are mixed within one category.

When setting the branching standard, the impurity of the child node must be set to decrease compared to the impurity of the current node, and this difference is called information gain. The impurity function as shown Eq(1). p is the proportion of classes belonging to each group.

$$G(S) = 1 - \sum_{i=1}^{c} p_i^2 \qquad p_i(i = 1, 2, \ldots, k) \tag{1}$$

DT divides the entire data into a train set and a test set through internal operations, learns each, and then calculates the classification rate. Additionally, in order to classify data using DT analysis, the type of data used must be data with continuous information. Let's define these data using R and Python programs.

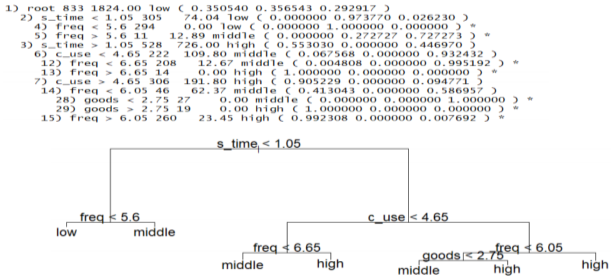
Table 1. shows the five-step analysis process of the decision tree

Analysis stage	Step-by-step analysis details	
Step 1	Analysis process	Creation of decision tree
	Depending on the purpose of analysis, appropriate separation criteria and stopping rules are established.	
Step 2	Analysis process	pruning
	Branches that have the potential to increase classification error or have inappropriate induction rules are removed.	
Step 3	Analysis process	feasibility assessment
	Cross-validation by means of a Gain chart, a risk chart, or verification data is analyzed.	
Step 4	Analysis process	Interpretation and prediction
	Interpret decision trees and set up prediction models.	
Step 5	Analysis process	Decision tree formation
	In the above process, different decision trees are formed depending on how separation criteria, stopping rules, and evaluation criteria are applied.	

2.1.1 Decision making by R program

The R program is a computer language that has been used for a long time as a free program for the public, and its use has increased rapidly since the importance of big data analysis was mentioned. DT analysis using the R program divides the entire data into a training set and a test set, learns it, and then calculates the classification rate.

The R program uses the internal prune() function to classify data into branch types, which is called a pruning function. In DT analysis, no matter what analysis tool is used, the classification criteria for pruning are determined by entropy and information gain from the parent note to the child node, and the calculation is performed by ‘installPackage(tree)’. The pruning after the pruning standard has been set by the prune() function of the R program as shown in (Figure 1).



```
1) root 833 1824.00 low ( 0.350540 0.356543 0.292917 )
  2) s_time < 1.05 305   74.04 low ( 0.000000 0.973770 0.026230 )
    4) freq < 5.6 294    0.00 low ( 0.000000 1.000000 0.000000 ) *
    5) freq > 5.6 11   12.89 middle ( 0.000000 0.272727 0.727273 ) *
  3) s_time > 1.05 528  726.00 high ( 0.553030 0.000000 0.446970 )
    6) c_use < 4.65 222  109.80 middle ( 0.067568 0.000000 0.932432 )
     12) freq < 6.65 208   12.67 middle ( 0.004808 0.000000 0.995192 ) *
     13) freq > 6.65 14    0.00 high ( 1.000000 0.000000 0.000000 ) *
    7) c_use > 4.65 306  191.80 high ( 0.905229 0.000000 0.094771 )
     14) freq < 6.05 46   62.37 middle ( 0.413043 0.000000 0.586957 )
       28) goods < 2.75 27    0.00 middle ( 0.000000 0.000000 1.000000 ) *
       29) goods > 2.75 19    0.00 high ( 1.000000 0.000000 0.000000 ) *
     15) freq > 6.05 260   23.45 high ( 0.992308 0.000000 0.007692 ) *
```

Figure 1. Pruning using R program

2.1.2 Decision making by Python program

The Python program is a computer program that has been used for a long time as a free program for the public, and its usage has increased rapidly since the importance of big data analysis was mentioned. Additionally, in order to implement AI functions today, a complex system must be built using Python and various libraries. When performing DT analysis using this widely used Python program, the entire data is divided into a training set and a test set, studied, and then the classification rate is calculated.

When pruning Python programs, the classification criteria are determined by entropy and information gain. The algorithms include ‘ID3, C4.5, C5.0’ based on machine learning and ‘CART, CHAID’ based on statistics. Among them, C5.0 is a supervised learning algorithm that improves the previous two types [20].

C5.0 is based on the concepts of entropy and information gain, and if the data of the initial target variables (explanatory variables) are mixed, impurity increases and entropy becomes large. In the process of classifying the data of each input variable (dependent variable), the data of the target variable is grouped by similar characteristics, lowering the entropy, and at this time, information gain occurs. The upper class of value keepers is determined by information gain, and the variable that creates the greatest information gain is selected. In C5.0, classification prediction uses the clf.predict() function, and the entropy model based on the target variable uses the tree. DecisionTree() function. The entropy and tree structure by the Python program as shown in (Figrue 2).

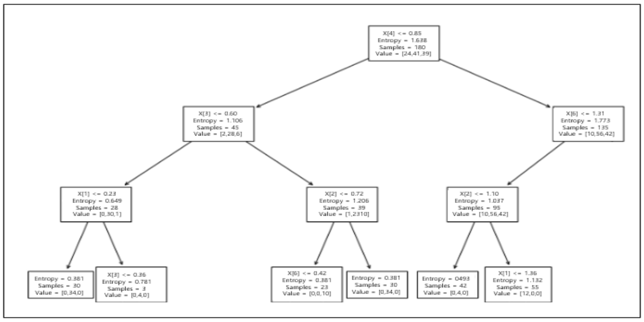
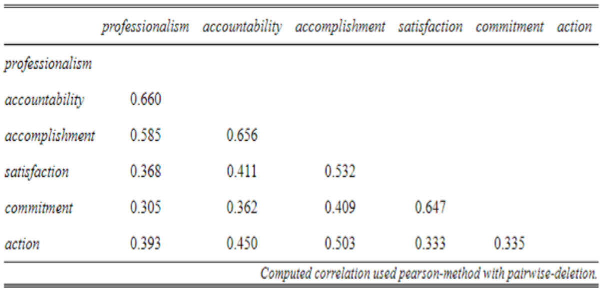


Figure 2. Tree structure by Python program

In DT analysis using Python and R programs, when the entropy becomes 0, the impurity concentration becomes 0%, so it can be said to be classified with only one piece of information.

2.2 Correlation Analysis(CA) Definition

Correlation analysis is an analysis technique that allows you to know the degree of relationship between variables using specific indicators. Specific indicators refer to correlation coefficients. Correlation refers to the relationship between variables. An indicator that shows the extent to which a change in one variable affects other variables is called correlation coefficients (CC). Correlation analysis is used when both the independent and dependent variables are ordinal scale or higher. When the scale of a variable is greater than the interval scale, the degree of correlation is confirmed by calculating Pearson's correlation coefficient, a continuous correlation analysis method. The correlation coefficient between variables extracted from Pearson's correlation analysis process as shown in (Figure 3).



	professionalism	accountability	accomplishment	satisfaction	commitment	action
professionalism						
accountability	0.660					
accomplishment	0.585	0.656				
satisfaction	0.368	0.411	0.532			
commitment	0.305	0.362	0.409	0.647		
action	0.393	0.450	0.503	0.333	0.335	

Computed correlation used pearson-method with pairwise-deletion.

Figure 3. Pearson's correlation coefficient

The range of the correlation coefficient can be obtained using a calculation formula, but it can only tell whether there is a linear relationship. The correlation coefficient between the two random variables X and Y is as shown Eq(2).

$$\rho_{X\cdot Y} = Corr(X, Y) = \frac{Cov(X, Y)}{\sqrt{V(X)\cdot V(Y)}},$$
$$-1 \le \rho_{X\cdot Y} \le 1 \qquad (2)$$

The sample correlation coefficient is obtained by the calculation formula in as shown Eq(3).

$$r = \frac{s_{xy}}{s_{xx}\cdot s_{yy}} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2\sum_{j=1}^{n}(y_j-\bar{y})^2}},$$
$$-1 \le r \le 1 \qquad (3)$$

The correlation coefficient has values from –1 to +1, and its meaning is as shown in [Table 2]. Looking at (Table 2), the correlation coefficient () has a value from -1 to +1 (), and the interpretation of the analysis results varies depending on whether it is close to -1 or +1. If the correlation coefficient is 0, it means that there is no correlation between variables.

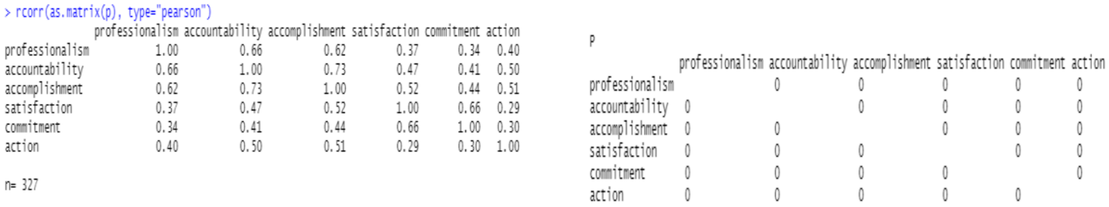
Table 2. Meaning of Pearson correlation coefficient

Correlation Coefficient Range	Mean
if, -1 < r < -0.7	Strong negative linear relationship
if, -0.7 < r < -0.3	A clear negative linear relationship
if, -0.3 < r < -0.1	Weak negative linear relationship

if, -0.1 < r < +0.1	Negligible linear relationship
If, +0.1 < r < +0.3	Weak positive linear relationship
if, +0.3 < r < +0.7	A clear positive linear relationship
if, +0.7 < r < +1	Strong positive linear relationship

Finding out the degree of correlation between variables is being discussed and studied in various fields today when trying to implement AI functions, and is becoming a field of much greater interest than causal inference. This correlation analysis can also be analyzed using R and Python programs, and the basic concept of correlation analysis is the same. It is briefly defined using an R program as an example.

When performing correlation analysis using an R program, you can roughly check the distribution and characteristics of the data by checking the structure of the data using R code commands. Additionally, the degree of relationship between variables can be determined by extracting the correlation coefficient using the cor() function. The correlation coefficient obtained by the cor() function as shown in (Figure 4).



```
> rcorr(as.matrix(p), type="pearson")
                professionalism accountability accomplishment satisfaction commitment action
professionalism            1.00           0.66           0.62         0.37       0.34   0.40
accountability             0.66           1.00           0.73         0.47       0.41   0.50
accomplishment             0.62           0.73           1.00         0.52       0.44   0.51
satisfaction               0.37           0.47           0.52         1.00       0.66   0.29
commitment                 0.34           0.41           0.44         0.66       1.00   0.30
action                     0.40           0.50           0.51         0.29       0.30   1.00

n= 327
```

```
P
                professionalism accountability accomplishment satisfaction commitment action
professionalism                 0              0              0            0          0
accountability  0                              0              0            0          0
accomplishment  0               0                             0            0          0
satisfaction    0               0              0                           0          0
commitment      0               0              0              0                       0
action          0               0              0              0            0
```

Figure 4. Extracting correlation coefficient of cor() function

Using the cor.test() function, you can find out whether there is a statistically significant correlation between variables through the significance probability value. By visualizing it through ‘install.packages(car), install.packages(corplot)’, you can more easily observe the relationships between variables. The visualization information using the R program as shown in (Figure 5).

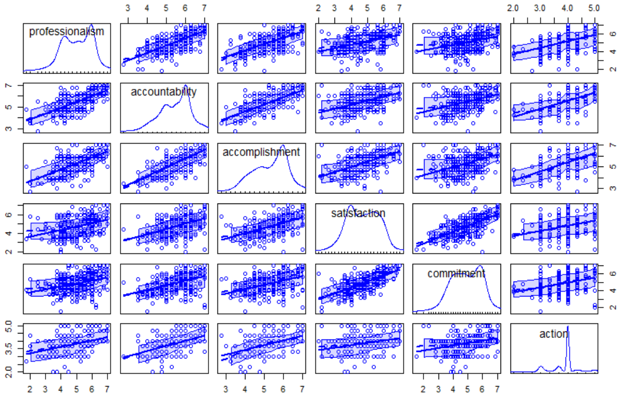


Figure 5. Visualization of correlation coefficient in R program

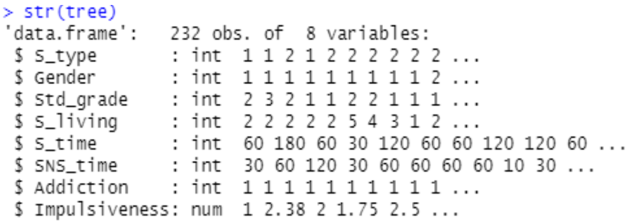
3. DT and CA experiments and results

In this study, a total of 8 variables were used as an experiment to find out what variables determine the addictiveness of smartphone use, and the number of data was 232. In this experiment, in order to evaluate whether smartphones are addictive, the values of each variable will be used to sequentially classify data that has a significant impact on smartphone addiction through DT analysis. By classifying data that affects addiction, we find out which variables have an influence. At this time, it is checked whether the information classified by DT analysis is a reliable classification result. In other words, by examining whether the classification results by DT analysis are affected by the degree of correlation between data, we examine whether the pruning criteria by DT analysis can be determined by the degree of correlation. Therefore, in this experiment, based on the prepared data, first, using DT analysis, the data is classified step by step according to whether the smartphone is Middle Eastern or not, and the criteria for classification are confirmed. Second, using the same data, we conduct correlation analysis to check how the correlation between variables is formed. Third, we compare and analyze the data classification criteria by DT analysis and the degree of correlation by correlation analysis to see what kind of relationship they have between them.

3.1 Decision Tree experiment

3.1.1 DT experiment using R code

In order to find out which independent variables affect the addictiveness of smartphones, the data is classified through DT analysis based on the prepared data. The data used in the analysis consists of a total of 232 data with 8 variables, and the extension is a CSV file. The information that summarizes the data in the actual CSV file using R code commands as shown in (Figure 6).

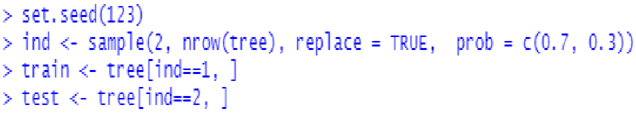


```
> str(tree)
'data.frame':   232 obs. of  8 variables:
 $ S_type       : int  1 1 2 1 2 2 2 2 2 2 ...
 $ Gender       : int  1 1 1 1 1 1 1 1 1 2 ...
 $ Std_grade    : int  2 3 2 1 1 2 2 1 1 1 ...
 $ S_living     : int  2 2 2 2 2 5 4 3 1 2 ...
 $ S_time       : int  60 180 60 30 120 60 60 120 120 60 ...
 $ SNS_time     : int  30 60 120 30 60 60 60 60 10 30 ...
 $ Addiction    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ Impulsiveness: num  1 2.38 2 1.75 2.5 ...
```

Figure 6. Data summary information using R code

Looking at the summary information using the str(tree) function in (Figure 6), it shows that it consists of 232 data and 8 variables, and the data structure is a DataFrame structure. There are 8 variables used, and each variable name is ‘S_type, Genger, Std_grade, S_living, S_time, SNS_time, addiction, Impulsiveness’. Additionally, you can see that the data type of each variable is 7 integer types, and the number type is 1.

For DT analysis, training and learning must be done by separating training data (train set) and test data (test set) based on a total of 232 data. In this experiment, the training data is divided into 70% and the test data is divided into 30% based on the total data. The process of dividing the actual data into 70% and 30%, respectively, using R code commands as shown in (Figure 7).



```
> set.seed(123)
> ind <- sample(2, nrow(tree), replace = TRUE,  prob = c(0.7, 0.3))
> train <- tree[ind==1, ]
> test <- tree[ind==2, ]
```

Figure 7. Data division process using the seed() function

In (Figure 7), the number of data corresponding to the divided 70% is 169, and the number of data corresponding to 30% is 63, which were replaced with train and test object variables, respectively.

The tree() function was used to create a tree structure based on the training data using R code. In order to use the tree() function to find and classify variables that affect addiction, entropy is calculated through comparative analysis of the addictive variable, which is the dependent variable, with 7 other independent variables, resulting in the independent variable with the fewest entries and the greatest information gain. Pruning is done based on variables. The original text of the function used at this time is tree(train$Addiction., data=train), and pruning according to data classification appears as shown in (Figure 8).

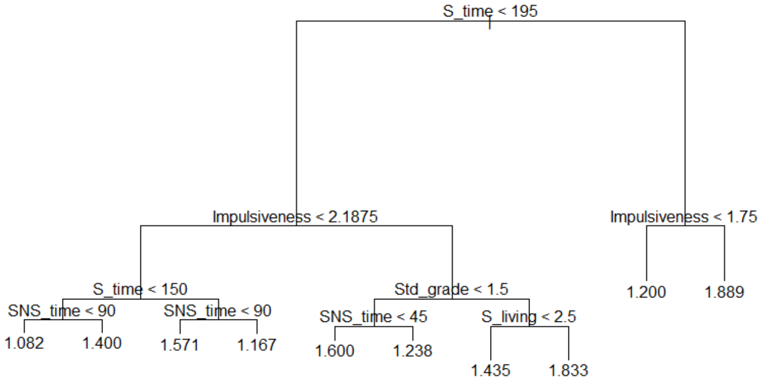


Figure 8. Pruning according to data classification

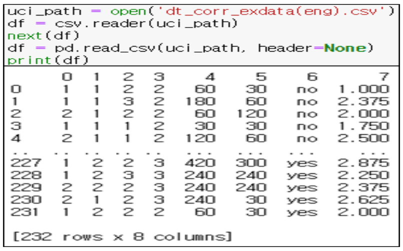
As can be seen in (Figure 8), pruning by DT analysis selected the S_time variable as the highest node, and the meaning of the S_time variable means ‘smartphone usage time’. In other words, the decision was made that the variable that has the greatest influence on smartphone addiction is that the risk of addiction is greatest when the smartphone itself has been used for a long time. In other words, the variable with the least etropy and the greatest information gain is S_time. The second largest information gain is the Impulsiveness variable, which refers to the ‘impulsiveness’ of smartphone use. Next, pruning was done with Std_grade, SNS_time, S_living, etc. In this pruning situation, the S_time and Impulsiveness variables that have the greatest variance among variables can be said to have the greatest impact on smartphone addiction.

3.1.2 DT experiment using Python code

So far, we have used R code to find out the pruning criteria according to data classification on smartphone addiction. If so, use Python code to check once again whether the data is classified based on the same criteria and whether pruning is done based on the same criteria.

Using a Python program, we will check which variables have an influence on smartphone addiction. First, the data structure for 8 variables of 232 data that will determine the presence or absence of smartphone addiction is confirmed in matrix and table structures. The Python code to show the data structure with summarized information based on the entire data and the data summary information output by the corresponding command as shown in (Figure 9).

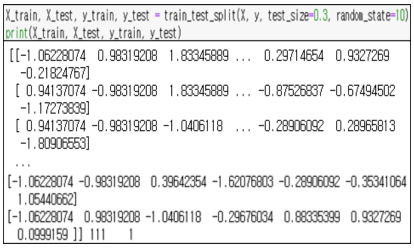
Second, separate the training set and test set based on the entire data. The training set to be separated is set to 70% (162 items) of the total, and the test set is set to 30% (70 items) to create the same criteria as the separation criteria in the R code. The refers to a command that sets the separation criteria to 70% of the training set and 30% of the test using Python code, and calculates the classification criteria for training and test data corresponding to each variable based on the entire data as shown in (Figure 10).



```
uci_path = open('dt_corr_exdata(eng).csv')
df = csv.reader(uci_path)
next(df)
df = pd.read_csv(uci_path, header=None)
print(df)
```

```
     0  1  2  3    4    5    6      7
0    1  1  2  2   60   30   no  1.000
1    1  1  3  2  180   60   no  2.375
2    2  1  2  2   60  120   no  2.000
3    1  1  1  2   30   30   no  1.750
4    2  1  1  2  120   60   no  2.500
..  .. .. .. ..  ...  ...  ...    ...
227  1  2  2  3  420  300  yes  2.875
228  1  2  3  3  240  240  yes  2.250
229  2  2  2  3  240  240  yes  2.375
230  2  1  2  3  240   30  yes  2.625
231  1  2  2  2   60   30  yes  2.000

[232 rows x 8 columns]
```

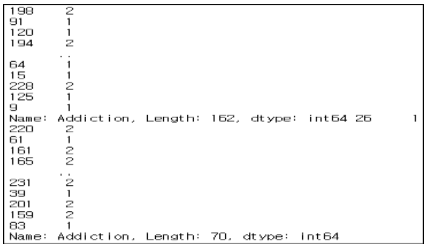
Figure 9. summary



```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=10)
print(X_train, X_test, y_train, y_test)
```

```
[[-1.06228074  0.98319208  1.83345889 ...  0.29714654  0.9327269
  -0.21824767]
 [ 0.94137074 -0.98319208  1.83345889 ... -0.87526837 -0.67494502
  -1.17273839]
 [ 0.94137074 -0.98319208 -1.0406118  ... -0.28906092  0.28965813
  -1.80906553]
 ...
[-1.06228074 -0.98319208  0.39642354 -1.62076803 -0.28906092 -0.35341064
  1.05440662]
[-1.06228074  0.98319208 -1.0406118  -0.29676034  0.88335399  0.9327269
  0.0999159 ]] 111    1
```

Figure 10. standard value

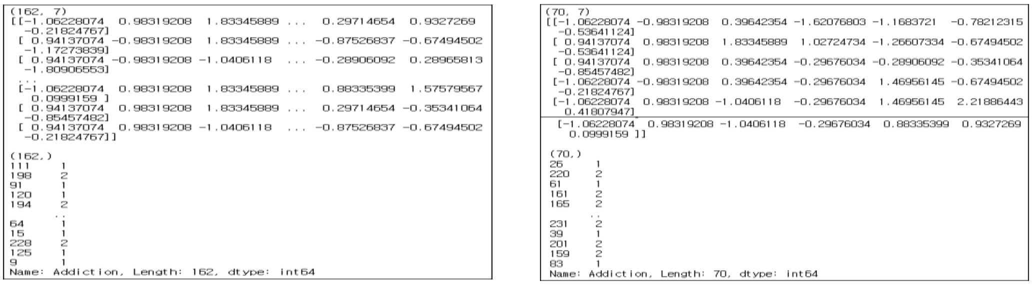


```
198    2
91     1
120    1
194    2
      ..
64     1
15     1
228    2
125    1
9      1
Name: Addiction, Length: 162, dtype: int64 26      1
220    2
61     1
161    2
165    2
      ..
231    2
39     1
201    2
159    2
83     1
Name: Addiction, Length: 70, dtype: int64
```

Figure 11. Train(70), test(30)

(Figure 11) shows the results separated into training and testing purposes according to the data classification standard value as shown in (Figure 10). Looking at (Figure 11), you can see that the instruction set is divided into 162 pieces out of a total of 232 data, and the test set is divided into 70 pieces out of a total of 232 data. Compared to the R code, there are 7 differences in data separation between the R codes, 70% (169) and 30% (63), but it can be seen that the data is separated at the same level. Looking at the additional information in (Figure 11), there is a numerical expression indicated by 1 and 2, which contains information on whether or not the smartphone is addicted. 1 means not addicted, 2 means addicted.

Third, check the input and target values for the training set and test set. Each value can be checked and output using train.shape and test.shape. The input and target values of the training set (left) and test set (right), respectively as shown in (Figure 12).



```
(162, 7)
[[-1.06228074  0.98319208  1.83345889 ...  0.29714654  0.9327269
  -0.21824767]
 [ 0.94137074 -0.98319208  1.83345889 ... -0.87526837 -0.67494502
  -1.17273839]
 [ 0.94137074 -0.98319208 -1.0406118  ... -0.28906092  0.28965813
  -1.80906553]
 ...
 [-1.06228074  0.98319208  1.83345889 ...  0.88335399  1.57579567
   0.0999159 ]
 [ 0.94137074  0.98319208  1.83345889 ...  0.29714654 -0.35341064
  -0.85457482]
 [ 0.94137074  0.98319208 -1.0406118  ... -0.87526837 -0.67494502
  -0.21824767]]

(162,)
111    1
198    2
91     1
120    1
194    2
      ..
64     1
15     1
228    2
125    1
9      1
Name: Addiction, Length: 162, dtype: int64
```

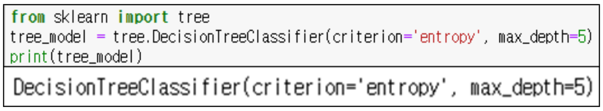
(a)Train set

```
(70, 7)
[[-1.06228074 -0.98319208  0.39642354 -1.62076803 -1.1683721  -0.78212315
  -0.53641124]
 [ 0.94137074  0.98319208  1.83345889  1.02724734 -1.26607334 -0.67494502
  -0.53641124]
 [ 0.94137074  0.98319208  0.39642354 -0.29676034 -0.28906092 -0.35341064
  -0.85457482]
 [-1.06228074 -0.98319208  0.39642354 -0.29676034  1.46956145 -0.67494502
  -0.21824767]
 [-1.06228074  0.98319208 -1.0406118  -0.29676034  1.46956145  2.21886443
   0.41807947]
  [-1.06228074  0.98319208 -1.0406118  -0.29676034  0.88335399  0.9327269
    0.0999159 ]]

 (70,)
 26     1
 220    2
 61     1
 161    2
 165    2
       ..
 231    2
 39     1
 201    2
 159    2
 83     1
 Name: Addiction, Length: 70, dtype: int64
```

(B)Test set

Figure 12. Input and target values of training set

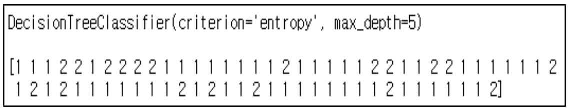
Fourth, set the entropy for the evaluation index of the target variable and proceed with modeling. Modeling is the step of finding a value with less entropy because the information gain increases as the entropy decreases, and the section where the range of change in the entropy value increases becomes the depth of pruning. The code that performs modeling based on the value of entropy and the results of the code as shown in (Figure 13). When executed by DecisionTreeClassifier(), entropy is calculated internally, and the maximum depth of pruning is limited to 5 levels.



```
from sklearn import tree
tree_model = tree.DecisionTreeClassifier(criterion='entropy', max_depth=5)
print(tree_model)
```

```
DecisionTreeClassifier(criterion='entropy', max_depth=5)
```

Figure 13. Entropy calculation



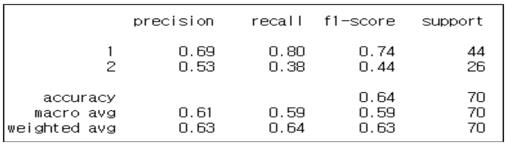
```
DecisionTreeClassifier(criterion='entropy', max_depth=5)

[1 1 1 2 2 1 2 2 2 2 1 1 1 1 1 1 1 2 1 1 1 1 2 2 1 1 2 2 1 1 1 1 1 1 2
 1 2 1 2 1 1 1 1 1 1 1 2 1 2 1 1 2 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 2]
```

Figure 14. Predict pruning model

Fifth, learn the training data as input and target fatigue to predict the model and determine the depth of pruning. Since the entropy calculation and pruning depth are set in (Figure 13), the pruning model set based on the addiction variable is predicted here. The steps for predicting the model based on the addiction variable according to the calculated pruning criteria as shown in (Figure 14).

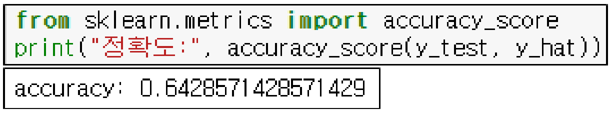
Sixth, predict the presence or absence of smartphone addiction by outputting a confusion matrix for the test set. In a confusion matrix, rows represent actual values and columns represent predicted values. Then, the result of the confusion in this experiment is [[35 9][16 10]], so the positions of each value are 35:[0][0], 9:[0][1], 16:[It becomes 1][0], 10:[1][1], and is the same as the matrix structure. In other words, if the model is predicted based on the entire 70 data in the test set, the rows are actual values, so there are 35 cases where no was selected when smartphone addiction is not present, and 9 cases where yes was incorrectly selected. Also, when it comes to smartphone addiction, 16 incorrectly selected no, and 10 incorrectly chose yes.

Seventh, the data classification results created by the matrix structure are confirmed with statistics. The table based on the data classification results contains various information about smartphone addiction. Among them, since the analysis was done using addictive information (1,2), accuracy (0.64) and other statistics and test sets, the support information is displayed as 70. The data classification results in a table based on the test set as shown in (Figure 15).



```
              precision    recall  f1-score   support

           1       0.69      0.80      0.74        44
           2       0.53      0.38      0.44        26

    accuracy                           0.64        70
   macro avg       0.61      0.59      0.59        70
weighted avg       0.63      0.64      0.63        70
```

Figure 15. Data classification result



```
from sklearn.metrics import accuracy_score
print("정확도:", accuracy_score(y_test, y_hat))
```

```
accuracy: 0.6428571428571429
```

Figure 16. Python code

Eighth, accuracy is calculated and displayed based on the results of data classification. Accuracy calculation in Python code is done with the command as shown in (Figure 16).

As can be seen in (Figure 16), the accuracy of the measured data classification is 0.642%, which is approximately 64% accuracy. This cannot be considered a high level of accuracy from an accuracy perspective, so the accuracy can be seen as being somewhat low. Low accuracy cannot be directly used to implement AI functions, and additional data refining and collection will need to be done. However, this experiment does not focus on accuracy, but rather studies the influence of correlations between variables in the data classification criteria by DT analysis, so we will not discuss whether the accuracy is high or low.

Ninth, pruning is performed based on the classified results, and a tree structure is drawn based on the value with the largest information benefit as a factor affecting smartphone addiction. In the tree structure, the node at the top is the parent node and the node at the bottom is the child node. The parent node's factor variables can be interpreted in the same way and are composed of variables that can exert significant influence on the child node. The entropy of this parent node has a larger value than the entropy of the child node. In other words, the entropy increases as you go upward, so the information gain decreases. However, on the contrary, as you go downward, the entropy decreases and the information gain increases, making it possible to set a standard value by which data can be classified. Therefore, among all nodes, the parent node is composed of variables that can exert influence on the child nodes. The tree structure created by calculating entropy and information gain at each stage after pruning according to data classification criteria by Python commands as shown in (Figure 17). In the tree structure of (Figure 17), the X[number] sign represents each variable used in smartphone addiction analysis, and the X[number] notation for each variable can be expressed as shown in (Table 3).

Table 3. Display X[number] for each variable

Variable Name	X[number]	Variable Name	X[number]
S_type	X[0]	S_time	X[4]
Genger	X[1]	SNS_time	X[5]
Std_grade	X[2]	Impulsiveness	X[6]
S_living	X[3]		

The tree structure in (Figure 17) must have been created by repeating the process of selecting a parent node by generating the highest information gain at the lowest entropy value. Then, in the tree structure, the factor variable that has the highest influence on smartphone addiction is X[4], which refers to the S_time variable, which means the time of smartphone use. In other words, the biggest influence on addiction is that if you use your smartphone for a long time regardless of time and place, you are most likely to become addicted. Second, X[6] refers to the Impulsiveness variable, which refers to impulsive use of a smartphone. Therefore, the most influential variable is the S_time variable, and the second most influential variable is the Impulsiveness variable. The third most influential variable is the SNS_time variable, and it can be seen that the variable of frequent use of various social networks such as KakaoTalk, Messenger, Fatebok, Twitter, and Instagram is influencing addiction. Next, according to S_type, addiction is relatively higher for high school students than for middle school students, and in terms of gender variable, there is also a difference in addiction between men and women. Additionally, there may be a change in influence between the S_type variable and the Gender variable due to the S_living variable. So far, we have applied DT analysis using R and Python programs to find out what variables have an influence on smartphone addiction. In conclusion, there may be slight numerical differences in both programs, but the data were classified at the same level when selecting factor variables that influence smartphone addiction. In both programs, the S_time variable was selected as the most influential factor variable and selected as the top root node. As a child node, the Impulsiveness variable was set as a child node, and lower child nodes were set to SNS_time, S_type, and S_living variables. The model was estimated appropriately according to the influence of smartphones on addictiveness. However, both programs showed somewhat low accuracy values, but this is not mentioned in this experiment. However, whether the data classification criteria vary depending on the degree of correlation between variables in the classified data, or whether the data classification criteria are related to the correlation between variables. Let's find out if there is any.

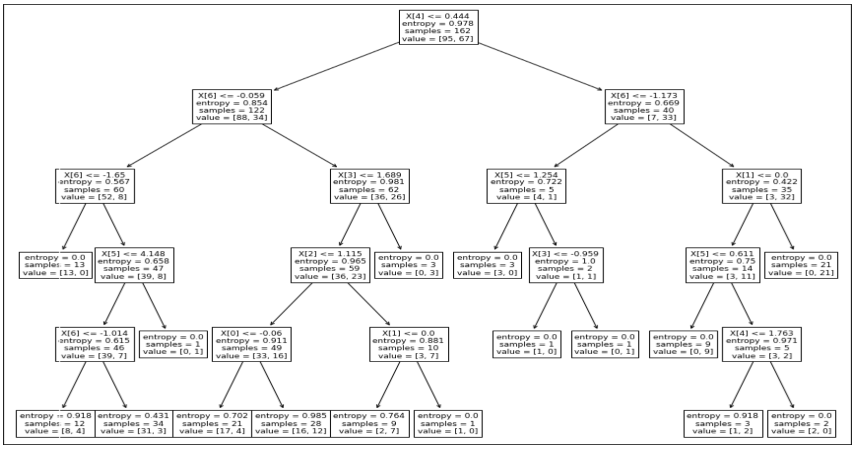


Figure 17. Tree structure creation using entropy and information gain

3.2 Correlation Analysis (CA) experiment

CA is an analysis technique that allows you to know the degree of relationship between variables using specific indicators. Specific indicators refer to correlation coefficients. The results obtained through the CA process include data summary information and correlation coefficients between variables. In addition, visualization based on significance probability values and correlation coefficients for hypothesis testing of correlation can be expressed. There are various expression methods for visualization, but visualization in correlation analysis makes it easy to see the degree of correlation between variables by using a linear relationship using a point distribution diagram and a circular distribution diagram. In addition, by using the correlation coefficient value and displaying it in table form, it is possible to know the degree of correlation that was known through the visualization information through a numerical expression of the correlation coefficient.

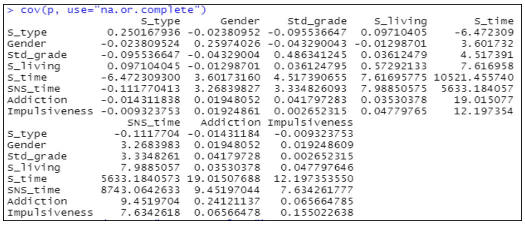
3.2.1 Correlation coefficient extraction

To extract the correlation coefficient, the cov() function and cor() function were used. The cov(p. use="na.or.complete") function is a step in calculating the covariance. The closer the result of the cov(x,y) function is to 0, the smaller the linear correlation. The meaning of the calculation results of the covariance cov(x,y) function as shown in (Table 4).

Table 4. Meaning of covariance results

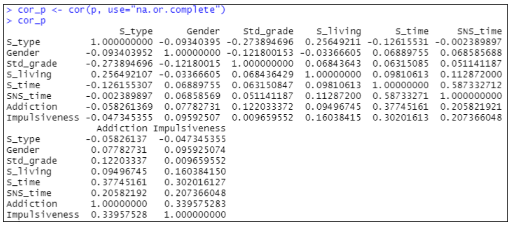
Division	Mean
First	If the Cov(X,Y) value has a positive value, there is a large linear correlation in the positive direction.
Second	If the Cov(X,Y) value has a negative value, there is a large linear correlation in the negative direction.
Third	When Cov(X,Y) is close to 0, linear correlation is small.

The results of the calculation using the cov(p. use="na.or.complete") function of the Python code as a cross table according to each variable as shown in (Figure 18). At this time, the closer the value of each cross information according to the cross table is to 0, the smaller the correlation between the variables, and the closer it is to 0, the greater the correlation between the variables. If you look at the cross table, you can see that the cross information with yourself appears to be the largest value, which means that the correlation with yourself is the greatest.



```
> cov(p, use="na.or.complete")
                    S_type      Gender    Std_grade    S_living       S_time
S_type         0.250167936 -0.02380952 -0.095536647  0.09710405    -6.472309
Gender        -0.023809524  0.25974026 -0.043290043 -0.01298701     3.601732
Std_grade     -0.095536647 -0.04329004  0.486341245  0.03612479     4.517391
S_living       0.097104045 -0.01298701  0.036124795  0.57292133     7.616958
S_time        -6.472309300  3.60173160  4.517390655  7.61695775 10521.455740
SNS_time      -0.111770413  3.26839827  3.334826093  7.98850575  5633.184057
Addiction     -0.014311838  0.01948052  0.041797283  0.03530378    19.015077
Impulsiveness -0.009323753  0.01924861  0.002652315  0.04779765    12.197354
                  SNS_time   Addiction Impulsiveness
S_type          -0.1117704 -0.01431184  -0.009323753
Gender           3.2683983  0.01948052   0.019248609
Std_grade        3.3348261  0.04179728   0.002652315
S_living         7.9885057  0.03530378   0.047797646
S_time        5633.1840573 19.01507688  12.197353550
SNS_time      8743.0642633  9.45197044   7.634261777
Addiction        9.4519704  0.24121137   0.065664785
Impulsiveness    7.6342618  0.06566478   0.155022638
```

Figure 18. covariance calculation



```
> cor_p <- cor(p, use="na.or.complete")
> cor_p
                   S_type      Gender    Std_grade    S_living      S_time     SNS_time
S_type        1.000000000 -0.09340395 -0.273894696  0.25649211 -0.12615531 -0.002389897
Gender       -0.093403952  1.00000000 -0.121800153 -0.03366605  0.06889755  0.068585688
Std_grade    -0.273894696 -0.12180015  1.000000000  0.06843643  0.06315085  0.051141187
S_living      0.256492107 -0.03366605  0.068436429  1.00000000  0.09810613  0.112872000
S_time       -0.126155307  0.06889755  0.063150847  0.09810613  1.00000000  0.587332712
SNS_time     -0.002389897  0.06858569  0.051141187  0.11287200  0.58733271  1.000000000
Addiction    -0.058261369  0.07782731  0.122033372  0.09496745  0.37745161  0.205821921
Impulsiveness -0.047345355  0.09592507  0.009659552  0.16038415  0.30201613  0.207366048
               Addiction Impulsiveness
S_type       -0.05826137  -0.047345355
Gender        0.07782731   0.095925074
Std_grade     0.12203337   0.009659552
S_living      0.09496745   0.160384150
S_time        0.37745161   0.302016127
SNS_time      0.20582192   0.207366048
Addiction     1.00000000   0.339575283
Impulsiveness  0.33957528   1.000000000
```

Figure 19. Correlation coefficient (0.0-1.0)

The correlation coefficient obtained using the cor(x,y) function during DT analysis, and is an analysis to determine the degree of correlation between each of the eight variables as shown in (Figure 19). The correlation coefficient, an indicator that can determine the correlation between variables, has a value between 0 and 1. In addition, depending on the range of values extracted through the operation of the cor(x,y) function, the meaning of the correlation of the linear relationship changes into ‘weak linear relationship’, ‘clear linear relationship’, and ‘strong linear relationship’.

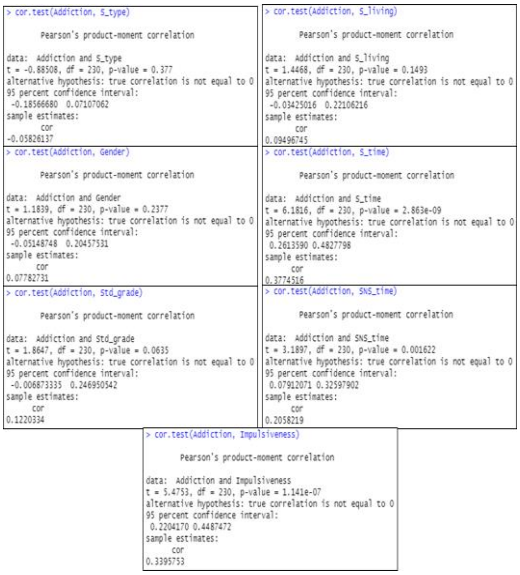
3.2.2 Extract significance probability value

In the DT analysis process, significance probability values are extracted to determine whether eight variables have an influence on smartphone addiction. The hypothesis test for addictiveness follows the following steps as shown in (Table 5).

Table 5. Hypothesis test to verify addictiveness

Division	Mean
First	Null Hypothesis: The addictiveness of smartphone use has no correlation with the variables. Alternative Hypothesis: The addictiveness of smartphone use is related to variables.
Second	Significance level: The confidence interval is 95% and p-value: 0.05.
Third	cor(Addiction~ 7 variables each) Operation, extraction of significance probability value
Fourth	The p-value of the significance level: 0.05 is compared with the significance probability value of the statistic..

The extraction of the significance probability value is compared with the significance level set in the hypothesis testing stage. If the significance probability value extracted by the cor(x,y) function is 0.05 or more based on a p-value of 0.05, the null hypothesis is adopted, thereby reducing the addictiveness of smartphone use. can be seen as having no correlation with the variables. However, if the significance probability value is less than 0.05, the null hypothesis is rejected and the alternative hypothesis is accepted, so it can be considered statistically significant. Therefore, since the significance level, which is the confidence interval, is less than 0.05, it can be said that the seven variables are correlated with the addictiveness of smartphone use. In addition, the greater the significance level below 0.05, the greater the influence on the addictiveness of smartphone use. The result of extracting significance probability values using the cor(x,y) function for each of the seven variables based on the Addiction variable as shown in (Figure 20).



```
> cor.test(Addiction, S_type)

        Pearson's product-moment correlation

data:  Addiction and S_type
t = -0.88508, df = 230, p-value = 0.377
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.18566680  0.07107062
sample estimates:
        cor
-0.05826137
```

```
> cor.test(Addiction, S_living)

        Pearson's product-moment correlation

data:  Addiction and S_living
t = 1.4468, df = 230, p-value = 0.1493
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.03425016  0.22106216
sample estimates:
       cor
0.09496745
```

```
> cor.test(Addiction, Gender)

        Pearson's product-moment correlation

data:  Addiction and Gender
t = 1.1839, df = 230, p-value = 0.2377
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.05148748  0.20457531
sample estimates:
       cor
0.07782731
```

```
> cor.test(Addiction, S_time)

        Pearson's product-moment correlation

data:  Addiction and S_time
t = 6.1816, df = 230, p-value = 2.863e-09
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2613590 0.4827798
sample estimates:
      cor
0.3774516
```

```
> cor.test(Addiction, Std_grade)

        Pearson's product-moment correlation

data:  Addiction and Std_grade
t = 1.8647, df = 230, p-value = 0.0635
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.006873335  0.246950542
sample estimates:
      cor
0.1220334
```

```
> cor.test(Addiction, SNS_time)

        Pearson's product-moment correlation

data:  Addiction and SNS_time
t = 3.1897, df = 230, p-value = 0.001622
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.07912071 0.32597902
sample estimates:
      cor
0.2058219
```

```
> cor.test(Addiction, Impulsiveness)

        Pearson's product-moment correlation

data:  Addiction and Impulsiveness
t = 5.4753, df = 230, p-value = 1.141e-07
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2204170 0.4487472
sample estimates:
      cor
0.3395753
```

Figure 20. Significance p_value(7)

Table 6. Significance p_value(7)

V_name	Significance P_value
S_type	data: Addiction and S_type t = -0.88508, df = 230, p-value = 0.377
Geiger	data: Addiction and Gender t = 1.3124, df = 230, p-value = 0.1907
Std_grade	data: Addiction and Std_grade t = 1.8647, df = 230, p-value = 0.0635
S_living	data: Addiction and S_living t = 1.4468, df = 230, p-value = 0.1493
S_time	data: Addiction and S_time t = 6.1816, df = 230, p-value = 2.863e-09
SNS_time	data: Addiction and SNS_time t = 3.1897, df = 230, p-value = 0.001622
Impulsiveness	data: Addiction and Impulsiveness t = 5.4753, df = 230, p-value = 1.141e-07

A p-value of each significance probability value measured by arranging the addiction variable and the 7 variables as shown in (Table 6). (Table 6) provides information about other variables connected to the Addiction variable, which indicates addiction to smartphones. Additionally, the degree of freedom is expressed through t-value and df along with p-value. Here, t-value can be used as the same standard as p-value, and the standard value is 1.96. In other words, a t-value of 1.96 means a p-value of 0.05. If the measured significance probability value is 0.05 or more, the t-value appears as a value less than 1.96, and if it is less than 0.05, the t-value appears as a value greater than 1.96. Then, when examining (Table 6) with the same standard, all variables are compared at a significance level of 0.05. The explanation follows the following order.

First, the p-value of S_type is 0.37, which is higher than the standard value, so it can be said to have no correlation with addiction by adopting the null hypothesis. Also, since it is 0.37, which is above the standard value, it can be confirmed that the t-value is -0.88, which is less than 1.96.

Second, the p-value of Gender is 0.19, which is more than the standard value, so it can be said to have no correlation with addiction as the null hypothesis is adopted. Additionally, since 0.19 is above the standard value, it can be confirmed that the t-value is 1.31, which is less than 1.96.

Third, the p-value of Std_garde is 0.06, which is higher than the standard value, so it can be said to have no correlation with addiction by adopting the null hypothesis. Also, since it is 0.06, which is above the standard value, it can be confirmed that the t-value is 1.86, which is less than 1.96.

Fourth, the p-value of S_living is 0.14, which is higher than the standard value, so it can be said to have no correlation with addiction by adopting the null hypothesis. Also, since it is 0.14, which is above the standard value, it can be confirmed that the t-value is 1.44, which is less than 1.96.

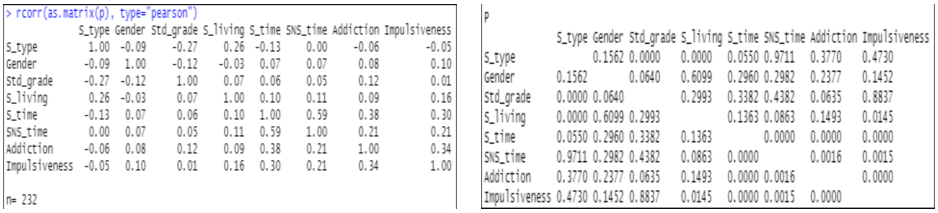
Fifth, the p-value of S_time is 2.863e-09, which is 0.0000000028, so it can be said that there is a correlation that affects addictiveness by rejecting the null hypothesis and adopting the alternative hypothesis as it is less than the standard value. Additionally, it can be confirmed that the t-value is 6.1816, which is much larger than 1.96, which is less than the standard value of 0.0000000028.

Sixth, the p-value of SNS_time is 0.001622, which is less than the standard value. By rejecting the null hypothesis and adopting the alternative hypothesis, it can be said that there is a correlation that affects addiction. Additionally, it can be seen that the t-value is 3.1897, which is much larger than 1.96, which is 0.001622, which is less than the standard value.

Seventh, the p-value of Impulsiveness is 1.141e-07, which is 0.00000011, so it can be said that there is a correlation that affects addictiveness by rejecting the null hypothesis and adopting the alternative hypothesis as it is less than the standard value. Additionally, it can be confirmed that the t-value is 5.4753, which is much larger than 1.96, which is 0.00000011, which is less than the standard value.

In other words, based on the results in (Table 6), it can be said that S_time, SNS_time, and Impulsiveness variables are the factor variables that have a significant influence on the addictiveness of smartphones. Therefore, in the DT analysis using the same data, it can be confirmed that the pruning criteria according to data classification are not incorrect, and it can be seen that the correlation of variables can exert influence when classifying data. Therefore, when implementing AI functions, it has been confirmed that the information carried by the data and the independence of variables can have a significant impact on data classification, and this can be said to be a very large anxiety factor in AI functions due to data misclassification (error classification).

Pearson's correlation coefficient extracted using the rcorr(x,y) function as shown in (Figure 21). By checking the cross information for each variable, you can see which variables have correlations. The larger the cross-information value, the deeper the correlation between variables.



```
> rcorr(as.matrix(p), type="pearson")
              S_type Gender Std_grade S_living S_time SNS_time Addiction Impulsiveness
S_type          1.00  -0.09     -0.27     0.26  -0.13     0.00     -0.06         -0.05
Gender         -0.09   1.00     -0.12    -0.03   0.07     0.07      0.08          0.10
Std_grade      -0.27  -0.12      1.00     0.07   0.06     0.05      0.12          0.01
S_living        0.26  -0.03      0.07     1.00   0.10     0.11      0.09          0.16
S_time         -0.13   0.07      0.06     0.10   1.00     0.59      0.38          0.30
SNS_time        0.00   0.07      0.05     0.11   0.59     1.00      0.21          0.21
Addiction      -0.06   0.08      0.12     0.09   0.38     0.21      1.00          0.34
Impulsiveness  -0.05   0.10      0.01     0.16   0.30     0.21      0.34          1.00

n= 232
```

```
P
              S_type Gender Std_grade S_living S_time SNS_time Addiction Impulsiveness
S_type               0.1562 0.0000    0.0000   0.0550 0.9711   0.3770    0.4730
Gender        0.1562        0.0640    0.6099   0.2960 0.2982   0.2377    0.1452
Std_grade     0.0000 0.0640           0.2993   0.3382 0.4382   0.0635    0.8837
S_living      0.0000 0.6099 0.2993             0.1363 0.0863   0.1493    0.0145
S_time        0.0550 0.2960 0.3382    0.1363          0.0000   0.0000    0.0000
SNS_time      0.9711 0.2982 0.4382    0.0863   0.0000          0.0016    0.0015
Addiction     0.3770 0.2377 0.0635    0.1493   0.0000 0.0016             0.0000
Impulsiveness 0.4730 0.1452 0.8837    0.0145   0.0000 0.0015   0.0000
```

Figure 21. Cross information of Pearson's correlation coefficient

3.2.3 Visualization of correlation coefficient

Visualization based on the correlation coefficient can be expressed as a scatter plot using point distribution and a distribution chart using color density. Among them, the first technique to display visualization information using a scatter plot allows the influence between variables to be known by expressing the degree of correlation between variables as a linear relationship. At this time, the semantic interpretation can be reversed depending on whether the linear relationship between variables is clockwise or counterclockwise. Additionally, depending on each direction, positive characteristics may be strong and negative characteristics may be strong. In the former case, the positive aspect is strong, and in the latter case, the negative aspect is strong. The linear relationship graph between variables with the correlation coefficient expressed as a scatterplot as shown in (Figure 22).

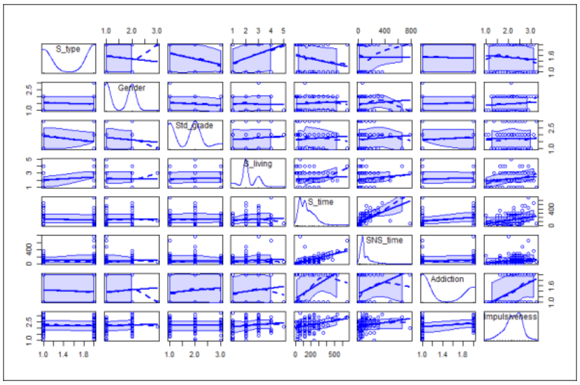


Figure 22. Linear relationship graph

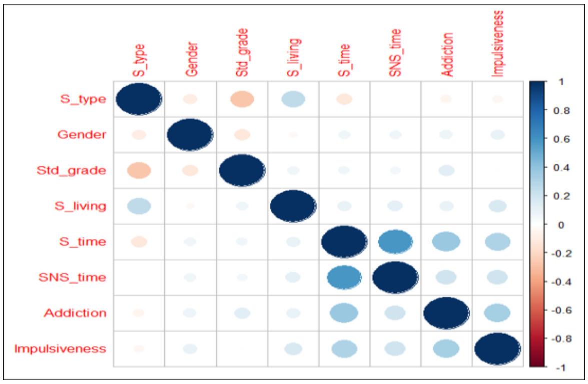


Figure 23. Distribution diagram

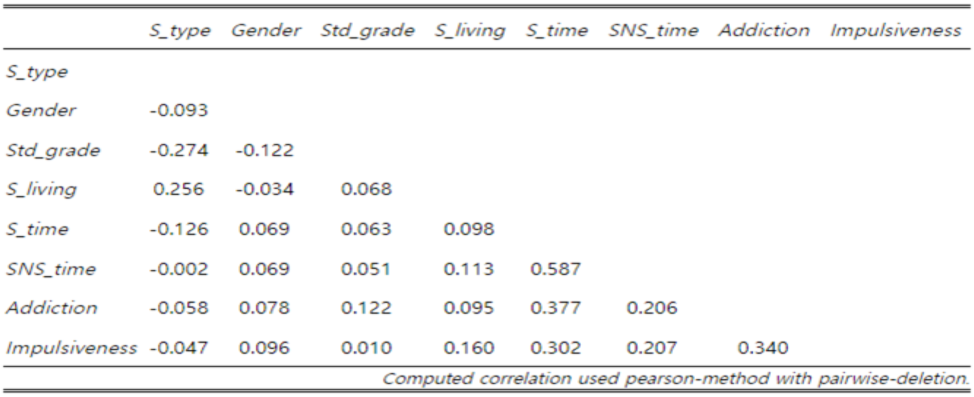
As shown in (Figure 22), the deep linear relationship between variables can be seen through the graph, as the S_time, SNS_time, and Impulsiveness variables in the middle of the matrix structure have a deeper correlation than other variables. In addition, the rate of change of the Gender and S_living variables can be confirmed by the S_time and SNS_time variables, so it can be seen that there is a certain degree of correlation between them.

Second, the degree of correlation is expressed using the original distribution diagram based on the correlation coefficient. When using a circular distribution, the size and color of the circle are expressed differently depending on the degree of correlation with other variables based on one's own cross information. In this way, by looking at the surrounding variables based on one variable, the correlation between them can be confirmed. The original distribution diagram of the correlation coefficient for each variable as shown in (Figure 23).

As can be seen from (Figure 23), it can be said that there is a somewhat clear correlation between the S_type, Gender, Std_grade, and S_living variables. On the other hand, it can be said that S_time and

SNS_time variables have a strong correlation with each other. It can be said that the pulsiveness variable has a clear correlation with the S_time and Addiction variables. In other words, when looking at the original distribution of the variables that affect smartphone addiction, S_time and SNS_time variables can be said to have the strongest influence, and Impulsiveness, S_type, Std_grade, and S_living variables can be said to be the next most influential factor variables. there is. It can be seen that the interpretation of this visualization is consistent with the interpretation by DT analysis.

Expressing data using visualization has the advantage of being able to quickly convey analysis results through visual information and enables quick understanding and interpretation of the analysis results, but it has the disadvantage of not being able to know the exact numbers. If the results of statistical analysis only provide pictorial results, decisions should be made based on rough interpretations. However, since the results of statistical analysis provide extremely numerical statistics, you can use statistics to examine the values corresponding to each item. The reliability and satisfaction with the analysis results can be improved. The correlation coefficients in table form as indicators for eight variables that evaluate the addictiveness of smartphones as shown in (Figure 24).



	S_type	Gender	Std_grade	S_living	S_time	SNS_time	Addiction	Impulsiveness
S_type								
Gender	-0.093							
Std_grade	-0.274	-0.122						
S_living	0.256	-0.034	0.068					
S_time	-0.126	0.069	0.063	0.098				
SNS_time	-0.002	0.069	0.051	0.113	0.587			
Addiction	-0.058	0.078	0.122	0.095	0.377	0.206		
Impulsiveness	-0.047	0.096	0.010	0.160	0.302	0.207	0.340	

Computed correlation used pearson-method with pairwise-deletion.

Figure 24. Table of correlation coefficients for variables

Looking at the correlation coefficient between the variables in (Figure 24), the variable with the highest correlation coefficient of Impulsiveness is S_time, which shows that people who use smartphones the most are likely to become addicted, and the second highest is S_time. SNS_time refers to a person who uses social networks a lot. The reason is that using social networks means using the smartphone itself more frequently. Then, if we look at the correlation between the S_time variable and other variables according to the SNS_time variable, we can see that the Std_grade, S_living, and S_type variables have a relative correlation, although there are some differences in strength and weakness. Therefore, S_time and SNS_time have the greatest influence as factor variables that can make smartphone use addictive, followed by S_type, Std_grade, and S_living variables. And there is a weak correlation between the Std_grade variable and the Gender variable.

4. Result

4.1 Discussion

This study aims to find out whether AI with automatic recognition ability recognizes objects, emotions, and facial expressions and classifies data well at the recognition stage, and whether the data classified according to characteristics is a reliable classification result. The assumption of the experiment is to conduct research to determine whether the data classification criteria affect the classification criteria depending on the degree of correlation between variables when classifying data using a decision tree. In this experiment, first, we classified the data using a decision tree (DT) and then checked the pruned tree structure and corresponding variables according to the classification criteria. We checked the pruned tree structure and the corresponding variables. As a result of the DT analysis overshoot, pruning was done between the Impulsiveness variable, which means addiction, and the other 7 variables according to data classification criteria among the 8 variables. As a result of the pruning of the DT analysis, the S_time variable was determined to have the greatest influence on smartphone addiction and was selected as the highest root node. Next, SNS_time, a variable with high social network usage, was selected as the child node of the root node. Next, S_type, Std_grade, and S_living variables were selected as child nodes. Second, we conduct correlation analysis (CA) using the same data and check through correlation coefficients whether the variables used in the analysis are correlated with each other. Since the correlation coefficient was also obtained from the cross information between the eight variables, the degree of correlation between them could be confirmed through the correlation coefficient value. The results of correlation analysis (CA) were also not different from the results of DT analysis.

4.2 Conclusion

It can be concluded that the S_time variable has the greatest influence on smartphone addiction, and next, with the SNS_time variable, people who use social networks a lot are more likely to become addicted. Following this, it can be said that the S_type, Std_grade, and S_living variables have some influence on addiction. Therefore, in DT analysis using the same data, it can be confirmed that the pruning criteria according to data classification are not incorrect, and it can be seen that the correlation of variables can exert influence when classifying data. Therefore, when implementing AI functions, it was confirmed that the information carried by the data and the independence of variables can have a significant impact on data classification. This means that data misclassification (error classification) is a very big anxiety factor in implementing AI functions, and it is necessary to establish and continuously research alternatives to eliminate this anxiety factor. This is because data never stays still and is constantly changing.

We will continue to conduct research to ensure complete data classification in the future, and we will not stop considering methodologies for sophisticated data classification. In order to prevent damage to people and objects in the upcoming AI era, classification and analysis of objects must continue around the world.

References

[1] Suchul Lee and Mihyun Ko, “Exploring the Key Technologies on Next Production Innovation,” Journal of the Korea Convergence Society, Vol. 9, No. 9, pp. 199-207, 2018.
DOI: https://doi.org/10.15207/JKCS.2018.9.9.199

[2] Youngsoon Kim, “Fourth Industrial Revolution(4IR) Hyper-Connected Society and Internet of Things Age,” The Korea contents Association Review, Vol. 17, No. 3, pp. 14-19, Sep 2019.

[3] Se Hoon Jung, Jong Chan Kim, Kim Cheeyong, Kang Soo You and Chun Bo Sim, “A Study on Classification Evaluation Prediction Model by Cluster for Accuracy Measurement of Unsupervised Learning Data,” Journal of Korea Multimedia Society, Vol. 21, No. 7, pp. 779-786, July 2018.
DOI: https://doi.org/10.9717/kmms.2018.21.7.779

[4] Hayoung Eom, Jeonghwan Kim, Seungyun Ji and Heeyoul Choi, “Autonomous Parking Simulator for Reinforcement Learning,” Journal of Digital Contents Society, Vol. 21, No. 2, pp. 381-386, Feb 2020.

DOI: http://dx.doi.org/10.9728/dcs.2020.21.2.381

[5] Hyung-Woo Lee, “Development of Supervised Machine Learning based Catalog Entry Classification and Recommendation System,” Journal of Internet Computing and Services(JICS), Vol. 20, No. 1, pp. 57-65, Feb 2019.
DOI: http://dx.doi.org/10.7472/jksii.2019.20.1.57

[6] Young Jin Kim, Joung Woo Ryu, Won Moon Song and Myung Won Kim, “Fire Probability Prediction Based on Weather Information Using Decision Tree,” Journal of KIISE : Software and Applications, Vol. 40, No. 11, pp. 705-715, Feb 2013.
DOI: http://scholarworks.bwise.kr/ssu/handle/2018.sw.ssu/11916

[7] Jihyun Lee, Jiyoung Woo, Ah Reum Kang, Young-Seob Jeong, Woohyun Jung, Misoon Lee and Sang HyunKim, “Comparative Analysis on Machine Learning and Deep Learning to Predict Post-Induction Hypotension,” Sensors 2020, 20(16), 4575, Aug 2020.
DOI: https://doi.org/10.3390/s20164575

[8] Yi-na Jeong, Yong-bo Sim and Su-rak Son, “DFLM(Deeplearning facial landmark model) for facial emotion classification,” Journal of Internet Computing and Services (JICS), Vol. 24, No. 3, pp. 43-50, June 2023.
DOI: http://dx.doi.org/10.7472/jksii.2023.24.3.43

[9] Jiyoung Lee, Jiho Kim, Euna Lee and Hongchul Lee, “Deep Learning Model Structure for Korean Facial Expression Detection,” Journal of KIIT, Vol. 21, No. 2, pp. 9-17, 2023.
DOI: http://dx.doi.org/10.14801/jkiit.2023.21.2.9

[10] Kyung-Seob Yoon and SangWon Lee, “Music player using emotion classification of facial expressions,” The Korean Society Of Computer And Information, Vol. 27, No. 1, pp. 243-246, Jan 2019.

[11] In-Kyung Byun and Jae-Ho Lee, “The Effect of Cognitive Movement Therapy on Emotional Rehabilitation for Children with Affective and Behavioral Disorder Using Emotional Expression and Facial Image Analysis,” The Journal of the Korea Contents Association(KCA), Vol. 16, No. 12, pp. 327-345, 2016.
DOI: http://dx.doi.org/10.5392/JKCA.2016.16.12.327

[12] Taebok Yoon and Jee-Hyong Lee, “Design of Heuristic Decision Tree (HDT) Using Human Knowledge,” Korean Institute of Intelligent Systems, Vol. 19, No. 4, pp. 525-531, Aug 2009.

[13] Jae-Ho Kim and Jang-Young Kim, “The Analysis of Correlation Between COVID-19 and Seoul Small Business Commercial Districts,” Journal of the Korea Institute of Information and Communication Engineering, Vol. 25, No. 3, pp. 384-388, 2021.
DOI: http://doi.org/10.6109/jkiice.2021.25.3.384

[14] Jae-Wan Yang, Young-Doo Lee and In-Soo Koo, “Sensor Fault Detection Scheme based on Deep Learning and Support Vector Machine,” The Journal of The Institute of Internet, Broadcasting and Communication (IIBC), Vol. 18, No. 2, pp. 185-195, Apr. 30, 2018.
DOI: https://doi.org/10.7236/JIIBC.2018.18.2.185

[15] Kyoungho Choi1 and Jin Ah Yoo, “A reviews on the social network analysis using R,” Journal of the Korean Chemical Society(JKCS), Vol. 6, No. 1, pp. 77-83, 2015.
DOI: http://dx.doi.org/10.15207/JKCS.2015.6.1.077

[16] Youngseok Lee, “Python-based Software Education Model for Non-Computer Majors,” Journal of the Korea Convergence Society, Vol. 9, No. 3, pp. 73-78, 2018.
DOI: https://doi.org/10.15207/JKCS.2018.9.3.073