A Study on the Credit Rating Analysis Based on Apartment Big-Data

Seung-Yeon Hwang[a], Jeong-Joon Kim[b]
[a] Dept of Computer Engineering, University of Anyang, Anyang-si, Gyeonggi-do, Republic of Korea
[b] Corresponding Author, Dept of Software, University of Anyang, Anyang-si, Gyeonggi-do, Republic of Korea
email: [a]syhwang@ayum.anyang.ac.kr, [b]jjkim@anyang.ac.kr

Abstract: With the rapid progress of the fourth industrial revolution recently, related technologies such as IoT, AI and Big-Data have developed innovatively and various kinds of big data are being generated in various fields. In particular, various studies using such big data are underway in the financial sector. Domestic and foreign fintech companies and financial institutions have applied big data and related technologies to credit rating and are making various efforts to expand and apply from traditional to new credit rating methods. Therefore, in this paper, we want to present guide information on the development of new credit rating model, such as carrying out credit rating analysis using apartment big data and visualizing the results so that we can understand them intuitively.

Keywords: Big Data Analysis, Credit rating, Visualization etc

1. Introduction

With the recent rapid progress of the Fourth Industrial Revolution, related technologies such as IoT, AI, and Big-Data have evolved innovatively, generating various kinds of big data in different fields [1]. This increase in big data is becoming a new opportunity to solve fundamental problems for complex phenomena, especially in the field of finance [2]. According to Gartner, big data has become a major driver of growth in the information and communication sector, and according to a survey of big data demand by industry, region and company, big data demand in three areas: finance, service and manufacturing is the highest [3].

In addition, traditional credit evaluation methods have recently been changing, with domestic and foreign fintech companies applying technologies such as machine learning and blockchain to non-traditional data such as SNS and call log data to use for credit evaluation. Currently, non-traditional data is being used by credit rating agencies, Internet banks, and P2P lenders in Korea, but active use is limited due to related regulations such as credit information laws and data accumulation. Therefore, it is necessary to expand domestic credit evaluation techniques through continuous inspection and utilization of new credit evaluation methods by domestic and foreign fintech companies and financial institutions. Therefore, in this paper, creditworthiness analysis based on apartment big data using the number of members, type of residence, length of residence, and market price data of each household of apartments was carried out and the results were visualized and expressed. In this work, we used the Jupiter Notebook and Python programming languages in a Windows 10 operating system environment for creditworthiness analysis.

This paper describes the relevant research in Chapter 2, and in Chapter 3 describes the data preprocessing methods and results. Chapter 4 describes the results of the analysis using pre-processed data and concludes in Chapter 5.

2.Related Work

2.1. Big Data

Big data refers to data with a vast amount that cannot be addressed in traditional computing environments, and the advent of the Fourth Industrial Revolution has resulted in a variety of data as multiple technologies evolve [4]. Big data includes semiformal data, such as xml and html documents, as well as unstructured data, which are text and multimedia data, and features of big data include volume, variety, and velocity of data [5]. Big data techniques are divided into the process of collection-storage-processing-analysis and visualization, which aims to derive meaningful information and discover new values [6].

2.2. Big Data Visualization

While big data is important for analysis itself, how to represent the value of analysis results to users is more important. Results expressed in simple numerical information or text are difficult for people to intuitively understand their meaning, so visualizing the analyzed results with charts, images, etc. is effective. Big data visualization is a technique that visually expresses information for intuitive understanding, which can be divided into information visualization, scientific visualization, infographics, etc. [7].

2.3. Regression Analysis

Regression analysis is an analytical technique used to verify the relationship between dependent and independent variables to determine how independent variables affect dependent variables, or to predict changes in dependent variables with changes in independent variables [8]. In this work, we will use OLS (Ordinary Least Squares) regression analysis to find out how each variable in apartment big data affects creditworthiness.

	Number_of_household_members	Residence_Type	Duration_of_residence	Supply_Area	Market_price	Vehicle_Registration_Count	management_cost	management_cost_payment_method	unpayment_Control_Rating	Number_of_unpayments	Unpaid_frequency	Credit_rating
0	4.0	Own house	133	123.31	17.3	1.0	244757	credit card	paid	0	0	100.0
1	1.0	Own house	9	108.53	15.0	1.0	268708	credit card	paid	0	0	100.0
2	5.0	Own house	3	123.31	17.3	1.0	326591	Bank Transfer	NaN	1	33	0.0
3	3.0	Own house	229	108.53	15.0	2.0	268491	credit card	NaN	18	8	60.0
4	3.0	Own house	36	108.53	15.0	2.0	283468	Bank Transfer	NaN	10	28	10.0
...	...	...	...	...	...	...	...	...	...	...	...	...
60294	1.0	Own house	96	60.53	3.1	1.0	131362	cash	NaN	7	7	60.0
60295	3.0	Own house	34	60.07	3.1	1.0	116818	credit card	Long-term unpaid	3	9	60.0
60296	1.0	Own house	2	60.07	3.1	1.0	104523	cash	paid	0	0	100.0
60298	1.0	Own house	72	60.53	3.1	1.0	152261	credit card	NaN	17	24	10.0
60308	1.0	Own house	478	0.00	0.0	4.0	45617	cash	NaN	5	1	100.0

50134 rows × 12 columns

Fig 1: Original apartment data for credit rating analysis

3.Data Processing

This chapter describes data preprocessing methods necessary for analyzing the creditworthiness based on apartment big data. Figure 1 is the original apartment big data used in this study, and personal information is de-identified.

3.1. Preprocess Raw Data

```
    apt_data.rename(columns={apt_data.columns[5]         :
'Number_of_household_members'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[8]         :
'Residence_Type'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[11]        :
'Duration_of_residence'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[14]        :
'Supply_Area'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[15]        :
'Market_price'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[16]        :
'Vehicle_Registration_Count'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[27]        :
'management_cost'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[36]        :
'management_cost_payment_method'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[38]        :
'unpayment_Control_Rating'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[40]        :
```

```
'Number_of_unpayments'}, inplace=True)

    apt_data.rename(columns={apt_data.columns[41]          :
'Unpaid_frequency'}, inplace=True)
```

Fig 2: Field name change source code

We leverage the source code in Figure 2 to simplify field names for improved data utilization.

```
    apt_data = apt_data.loc[:, ['Number_of_household_members', 'Residence_Type', 'Duration_of_residence',
'Supply_Area',          'Market_price',          'Vehicle_Registration_Count',          'management_cost',
'management_cost_payment_method','unpayment_Control_Rating','Number_of_unpayments','Unpaid_frequency']
]
```

Fig 3: Unnecessary field removal source code

The source code in Figure 3 removes unused fields from the original data for the entire analysis scenario. We then check the datatype of each field and change the inappropriate datatype to the appropriate datatype via the astype() function, and in the case of missing fields, remove it using the fillna() function or replace it with a specific value.

```
    apt_data.loc[(apt_data['Unpaid_frequency']    >=    0)    &
(apt_data['Unpaid_frequency'] < 5), 'Credit_rating'] = 100

    apt_data.loc[(apt_data['Unpaid_frequency']    >=    5)    &
(apt_data['Unpaid_frequency'] < 10), 'Credit_rating'] =  60

    apt_data.loc[(apt_data['Unpaid_frequency']    >=    10)    &
(apt_data['Unpaid_frequency'] < 20), 'Credit_rating'] = 30

    apt_data.loc[(apt_data['Unpaid_frequency']    >=    20)    &
(apt_data['Unpaid_frequency'] < 30), 'Credit_rating'] = 10

    apt_data.loc[apt_data['Unpaid_frequency']         >=         30,
'Credit_rating'] = 0
```

Fig 4: Credit rating field addition and allocation source code

Table 1: Standard of credit rating based on frequency of non-payment

Unpaid frequency	Credit rating
0~5	100 (A)
5~10	60 (B)
10~20	30 (C)
20~30	10 (D)
More than 30	0 (E)

Through the source code in Figure 4, we perform first-order preprocessing, such as adding credit rating fields to the original data and allocating credit ratings by dividing them into five grades (A–E) according to their outstanding frequency. Table 1 refers to credit rating criteria based on frequency of outstanding payments.

3.2 Data Pre-processing for Analysis Scenario 1

```
    s1_apt_data                                =                                s1_apt_data.loc[:,
['Number_of_household_members','Duration_of_residence','Supply_Area','Market_price','Vehicle_Registration_C
ount','management_cost','Number_of_unpayments','Credit_rating']]
```

Fig 5: Required Field Extraction Source Code 1

From the data pre-processed through the source code in Figure 5, ' Number_of_household_members ', ' Duration_of_residence ', ' Supply_Area ', ' Market_price ', ' Vehicle_Registration_Count ', ' management_cost ', ' Number_of_unpayments ', and 'Credit_rating' fields are extracted.

```
    Group_A                                                          =
s1_apt_data[(s1_apt_data['Duration_of_residence']  >=  1)  &
```

```
(s1_apt_data['Duration_of_residence'] <=121)]
Group_B                                                  =
s1_apt_data[(s1_apt_data['Duration_of_residence'] >= 122) &
(s1_apt_data['Duration_of_residence'] <=242)]
Group_C                                                  =
s1_apt_data[(s1_apt_data['Duration_of_residence'] >= 243) &
(s1_apt_data['Duration_of_residence'] <=363)]
Group_D                                                  =
s1_apt_data[(s1_apt_data['Duration_of_residence'] >= 364)]
```

Fig 6: Source Code for Grouping according to Duration of Residence

Table 2: Criteria for grouping according to length of residence

Group	Duration of Residence(month)
A	1~121
B	122~242
C	243~363
D	More than 364

Through the source code in Figure 6, we group them into four groups (A–D) according to their length of residence and save them as csv files, respectively. Table 2 refers to the criteria for grouping by length of residence.

Figure 9 shows the results of Group A data preprocessing in analytical scenario 1, and the forms of Group B, C, and D data are shown in Group A.

3.3 Data Pre-processing for Analysis Scenario 2

```
s2_apt_data                              =                        s2_apt_data.loc[:,
['Number_of_household_members','Residence_Type','Supply_Area','Market_price','Vehicle_Registration_Count','
management_cost','Number_of_unpayments','Credit_rating']]
```

Fig 7: Required Field Extraction Source Code 2

From the data pretreated through the source code in Figure 7, 'Number_of_household_members', 'Residence_Type', 'Supply_Area', 'Market_price', 'Vehicle_Registration_Count', 'management_cost', 'Number_of_unpayments', 'Credit_rating' fields required to analyze the number of unpaid times according to the type of residence are extracted.

```
Group_self_data                              =
s2_apt_data[s2_apt_data['Residence_Type'] == 'Own house']

Group_lease_data                             =
s2_apt_data[s2_apt_data['Residence_Type'] == 'lease']
```

Fig 8: Source Code of Grouping according to Residential Type

Through the source code in Figure 8, we group into two groups according to the type of residence (own house, lease) and save each as a csv file.

	Number_of_household_members	Duration_of_residence	Supply_Area	Market_price	Vehicle_Registration_Count	management_cost	Number_of_unpayments	Credit_rating
0	1.0	9	108.53	15.0	1.0	268708	0	100.0
1	5.0	3	123.31	17.3	1.0	326591	1	0.0
2	3.0	36	108.53	15.0	2.0	283468	10	10.0
3	5.0	7	123.31	17.3	1.0	232726	2	10.0
4	5.0	121	123.31	17.3	2.0	320502	8	60.0
...	...	...	...	...	...	...	...	...
33936	2.0	98	60.53	3.1	1.0	128561	20	10.0
33937	1.0	96	60.53	3.1	1.0	131362	7	60.0
33938	3.0	34	60.07	3.1	1.0	116818	3	60.0
33939	1.0	2	60.07	3.1	1.0	104523	0	100.0
33940	1.0	72	60.53	3.1	1.0	152261	17	10.0

Fig 9: Group A Data Preprocessing Results

	Number_of_household_members	Residence_Type	Supply_Area	Market_price	Vehicle_Registration_Count	management_cost	Number_of_unpayments	Credit_rating
0	4.0	Own house	123.31	17.3	1.0	244757	0	100.0
1	1.0	Own house	108.53	15.0	1.0	268708	0	100.0
2	5.0	Own house	123.31	17.3	1.0	326591	1	0.0
3	3.0	Own house	108.53	15.0	2.0	268491	18	60.0
4	3.0	Own house	108.53	15.0	2.0	283468	10	10.0
...	...	...	...	...	...	...	...	...
49680	1.0	Own house	60.53	3.1	1.0	131362	7	60.0
49681	3.0	Own house	60.07	3.1	1.0	116818	3	60.0
49682	1.0	Own house	60.07	3.1	1.0	104523	0	100.0
49683	1.0	Own house	60.53	3.1	1.0	152261	17	10.0
49684	1.0	Own house	0.00	0.0	4.0	45617	5	100.0

Fig 10: Data preprocessing results for groups whose residence type is their own home

Figure 10 shows the data preprocessing results of groups whose residence type is 'Own house' and has the same data form as groups whose residence type is 'lease'.

3.4 Data Pre-processing for Analysis Scenario 3

```
s4_apt_data = s4_apt_data.loc[:,
['Number_of_household_members',
'Supply_Area','Market_price','Vehicle_Registration_Count',
'management_cost','Number_of_unpayments','Credit_rating']]
```

Fig 11: Required Field Extraction Source Code 3

The source code in Figure 11 extracts 'Number_of_household_members','Supply_Area','Market_price','Vehicle_Registration_Count','management_cost', 'Number_of_unpayments','Credit_rating' fields needed to analyze the impact of each variable on credit rating (Scenario 3).

```
s4_apt_data = s4_apt_data[s4_apt_data['Market_price'] != 0]
```

Fig 12: Source code for removing fields with zero quotations

The source code in Figure 12 removes a row with zero value from the quoted field and saves it as a csv file.

4.Analysis Results

In this section, we discuss analysis scenarios to produce meaningful results through creditworthiness analysis using apartment big data.

4.1. (Scenario 1) Analysis of the number of unpaid payments according to the period of residence

```
plt.figure(figsize=(20, 15))

heatmap_A = sns.heatmap(data = Group_A_data.corr(), annot=True, annot_kws={'size': 20}, fmt = '.2f',
linewidths=.5, cmap='Greens')
```

```
heatmap_A.set_ylim(8, 0)

heatmap_A.set_yticklabels(['Number_of_household_members','Duration_of_residence','Supply_Area','Market
_price','Vehicle_Registration_Count', 'management_cost','Number_of_unpayments','Credit_rating'], va='center')

plt.title("Group_A Correlation among variables")

plt.show()
```

Fig 13: Correlation Analysis and Visualization Source Code 1

The source code in Figure 13 analyzes and visualises correlations (positive, negative, unrelated) between 'Number_of_household_members', 'Duration_of_residence', 'Supply_Area', 'Market_price', 'Vehicle_Registration_Count', 'management_cost','Number_of_unpayments','Credit_rating'.

Figure 14 illustrates the group A visualization of the correlation between each variable as a heatmap. The correlation between each variable in Group A shows that the remaining variables, except for 'Number_of_unpayments', do not seem to correlate with the 'Credit_rating', and 'Credit_rating' decreases as 'Number_of_unpayments' increases.

Figure 15 illustrates the group B visualization of the correlation between each variable as a heatmap. The correlation between each variable in Group B shows that the longer 'Duration_of_residence', the more 'Credit_rating' seems to improve slightly, and the lower 'Credit_rating' as the 'Number_of_unpayments' increases.

Figure 16 illustrates the group C visualization of the correlation between each variable as a heatmap. As a result of analyzing the correlation between each variable in Group C, the 'Credit_rating' decreases slightly as the 'Number_of_household_members' increases, and the 'Credit_rating' decreases as the 'Duration_of_residence' increases. Also, the 'Credit_rating' is lowered as the 'Number_of_unpayments' increases. On the other hand, the more expensive the 'Market_price' is, the better the 'Credit_rating' is.

Figure 17 illustrates the group D visualization of the correlation between each variable as a heatmap. Group D correlation between each variable resulted in a slight improvement in 'Credit_rating' as the 'Number_of_household_members' and 'Vehicle_Registration_Count' increased, and the longer the 'Duration_of_residence', the larger the 'Supply_Area', the more expensive the 'Market_price', the better the 'Credit_rating'. And as the 'Number_of_unpayments' increases, the 'Credit_rating' decreases.
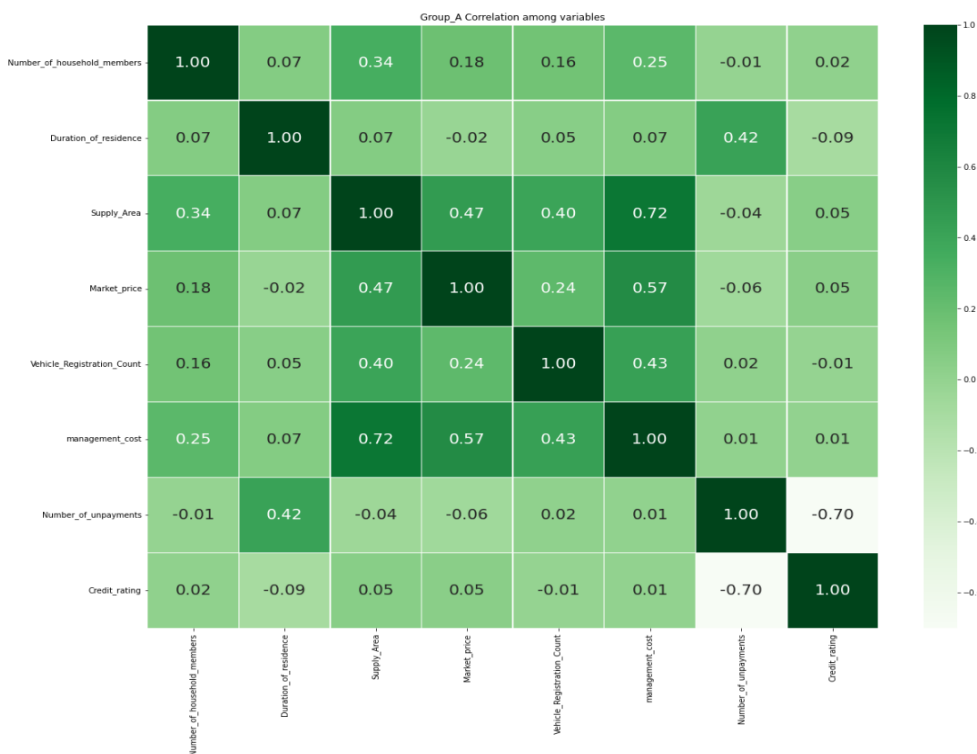

Fig 14: Group A - Correlation Analysis Result Heat Map

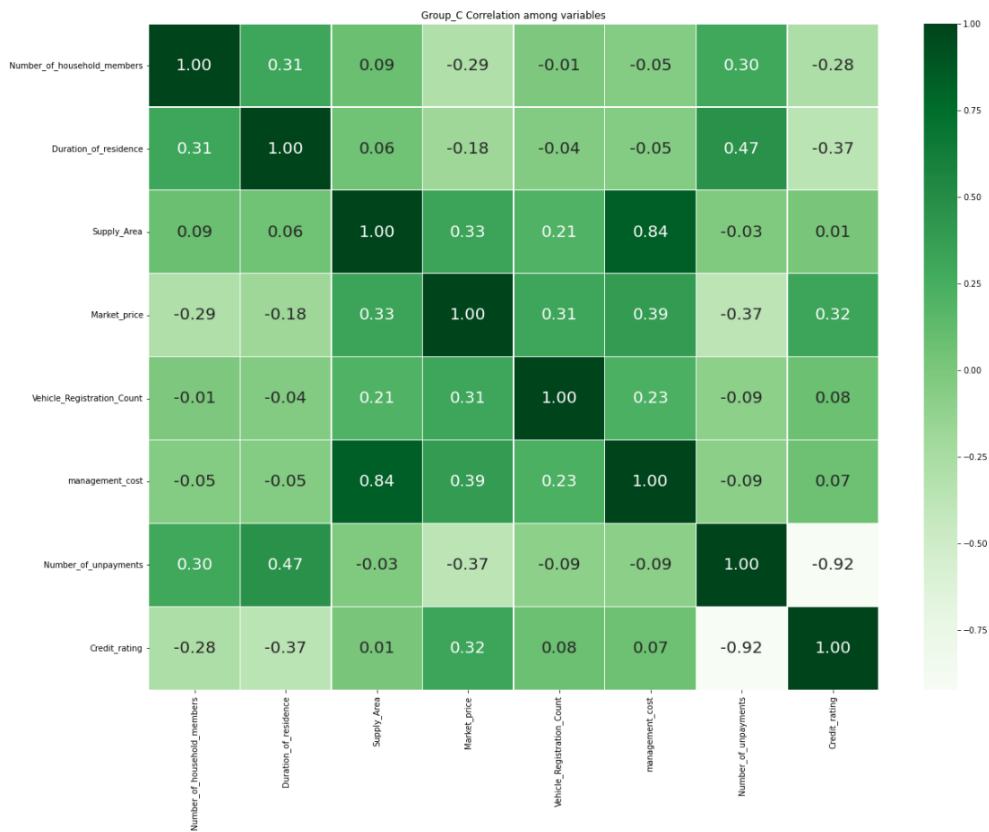**Fig 15:** Group B - Correlation Analysis Result Heat Map

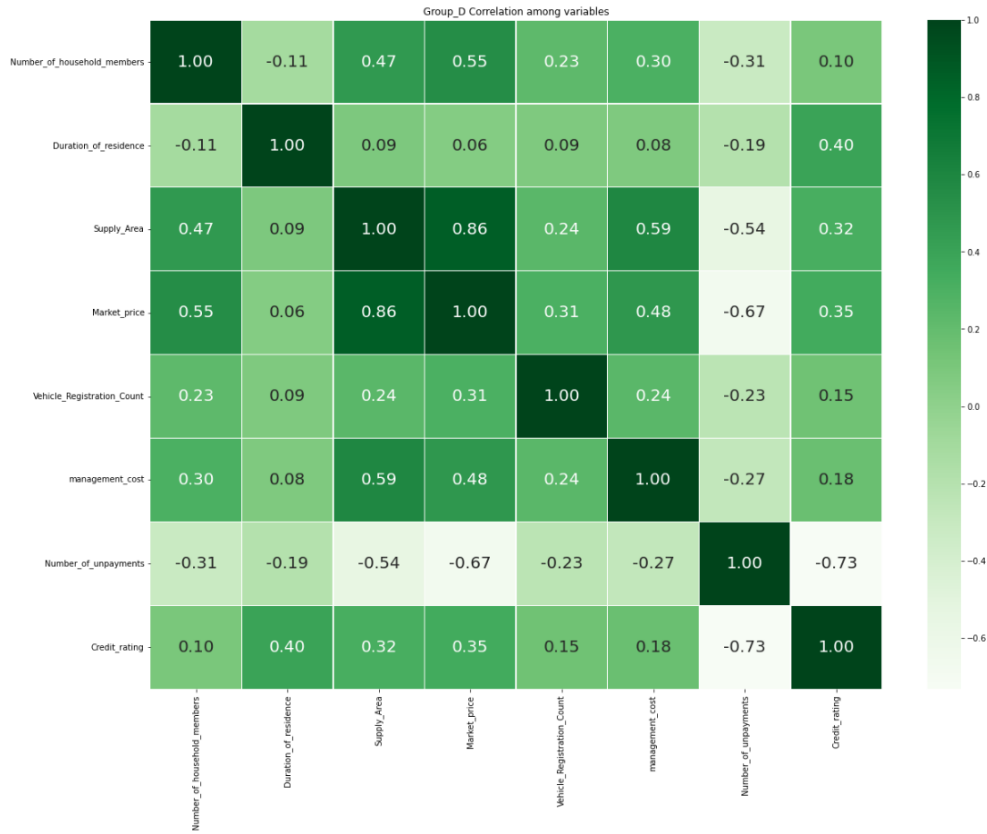**Fig 16:** Group C - Correlation Analysis Result Heat Map

Fig 17: Group D - Correlation Analysis Result Heat Map

4.2 (Scenario 2) Analysis of unpaid times according to type of residence

```
plt.figure(figsize=(20, 15))

heatmap_self = sns.heatmap(data = Group_self_data.corr(), annot=True, annot_kws={'size': 20}, fmt = '.2f',
linewidths=.5, cmap='Reds')

heatmap_self.set_xlim(0, 7)

heatmap_self.set_xticklabels(['Number_of_household_members','Supply_Area','Market_price','Vehicle_Regist
ration_Count', 'management_cost','Number_of_unpayments','Credit_rating'], va='center')


heatmap_self.set_ylim(7, 0)

heatmap_self.set_yticklabels(['Number_of_household_members','Supply_Area','Market_price','Vehicle_Regist
ration_Count', 'management_cost','Number_of_unpayments','Credit_rating'], va='center')

plt.title("Group_Own house Correlation among variables")

plt.show()
```

Fig 18: Group D - Correlation Analysis Result Heat Map

The source code in Figure 18 analyzes and visualizes the correlation (positive, negative, unrelated) between 'Number_of_household_members', 'Supply_Area', 'Market_price', 'Vehicle_Registration_Count', 'management_cost', 'Number_of_unpayments', 'Credit_rating' by group using data grouped by type of residence.

Figure 19 illustrates a heatmap visualization of the correlation between variables in a group whose type of residence is 'Own house'. An analysis of the correlation between variables in the group where the type of residence is 'Own house' shows that the rest of the variables except the 'Number_of_unpayments' do not correlate with the 'Credit_rating', and that the 'Credit_rating' decreases as the 'Number_of_unpayments' increases.

Figure 20 illustrates a heatmap visualization of the correlation between variables in a group where the residence type is 'lease'. An analysis of the correlation between variables in a group where the type of residence is

'lease' shows that the 'Credit_rating' improves slightly as the 'management_cost' increases, while the 'Credit_rating' decreases as the 'Number_of_unpayments' increases.

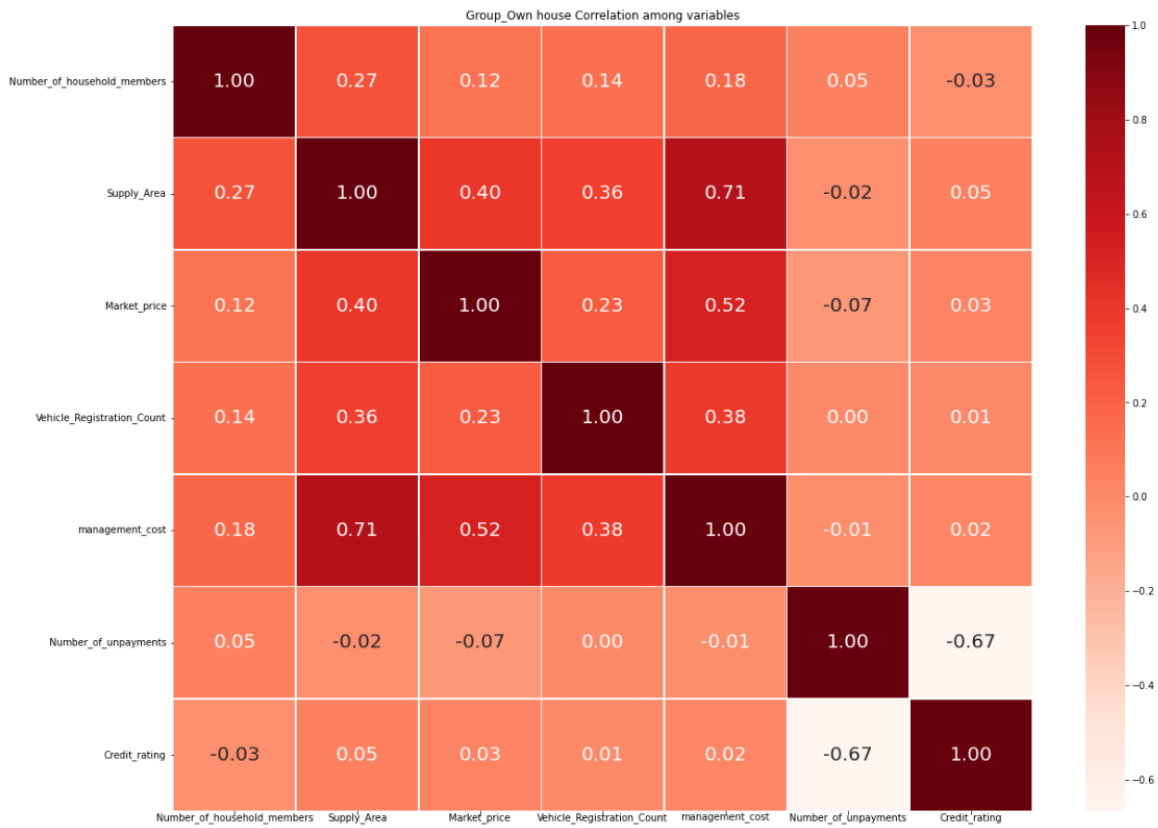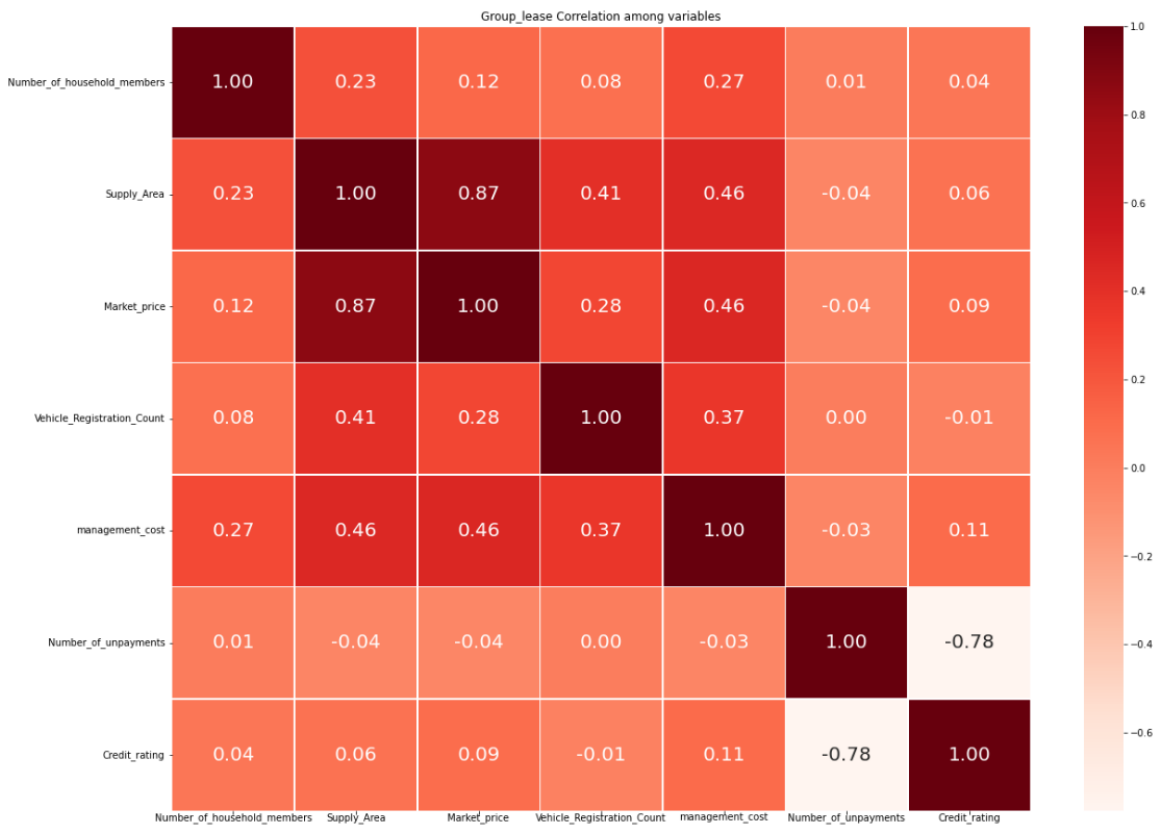**Fig 19:** Group(own home) - Correlation Analysis Result Heat Map

Fig 20: Group(lease) - Correlation Analysis Result Heat Map

4.2 (Scenario 3) Analysis of the Impact of Each Variable on Credit Ratings

```
s4_result     =     sm.ols(formula="Credit_rating     ~
Number_of_household_members     +     Supply_Area     +
Market_price     +     management_cost     +
Number_of_unpayments", data = s4_apt_data).fit()
```

Fig 21: OLS Regression Source Code

Using the source code in Figure 21, we analyze and visualize the effect of independent variables on dependent variables through OLS regression by setting the 'Credit_rating' of preprocessed data as dependent variables and setting the 'Number_of_household_members', 'Supply_Area', 'Market_price', 'management_cost', 'Number_of_unpayments' as independent variables.

Figure 22 illustrates the results of first OLS regression. Adj. R-Squared value is 0.45 which shows that the current model does not explain the data well. In addition, the p-value of the 'management_cost' field is 0.758, which is not considered to be a variable that affects the 'Credit_rating', which is a dependent variable, so we performed a regression analysis again except for the variable.

Fig 22: OLS Regression Results (First)

Figure 23 illustrates the results of quadratic OLS regression. Adj. R-Squared value is 0.45 which shows that the current model does not explain the data well. In the model, the p-value value of all variables is 0.05 or less, indicating that it is significant. According to OLS regression, an increase in the 'Number_of_household_members', 'Market_price', and 'Number_of_unpayments' reduces the 'Credit_rating' by 0.2317, 0.1725, and 3.1974, respectively, and an increase in the 'Supply_Area' increases the 'Credit_rating' by 0.0634.

Fig 23: OLS Regression Results (Second)

5.Conclusion

In this paper, we utilize big data from apartments to analyze the number of unpaid payments according to length of residence and type of residence and the impact of each variable on credit ratings, and visualize and express the results intuitively. We have identified a variety of variables that affect creditworthiness depending on the length of residence and type of residence, and it is expected that more significant results related to creditworthiness analysis will be produced if we utilize additional data as well as apartment big data. In addition, it is expected that it will be able to expand and apply from traditional credit rating methods to new credit rating methods if various data required for credit rating analysis are obtained and a model that can explain apartment big data and related data well.

References

1. Soon-Kyu Woo, Sung-In Cho, Soo-Yeon Yoon (2018) A Study on the Use of Big Data-based Personal Information De-indentification Measures in the Financial Industry : Focused on TOE Framework. The Journal of Internet Electronic Commerce Research 18(3):71-90
2. Hyun Joon Shin, Hyunwoo Ra (2015) The Journal of The Korean Operations Research and Management Science Society 32(3):91-103
3. Ji-Ung Kim, Jun Heo, Jang-Il Kim (2013) Use of Big Data for Financial Institutions. The Institute of Electronics and Information Engineers 40(8):49-54
4. Jung-Tae Hwang, Jeong-Joon Kim, Young-Gon Kim (2017) A Study on the relationship between the nightlifes and sexually transmitted infecters by R visualization. The Journal of The Institute of Internet, Broadcasting and Communication 17(6):187-193
5. Seung-Yeon Hwang, Ji-Hun Park, Ha-Young Youn, Kwang-Jin Kwak, Jeong-Min Park, Jeong-Joon Kim (2019) Big Data-based Medical Clinical Results Analysis. The Journal of The Institute of Internet, Broadcasting and Communication 19(1):187-195
6. HwaMin Jeong, SangYun Lee (2019) Analysis of Factors Affecting Big Data Use Intention of Korean Companies : Based on public data availability. The Journal of the Korea Academia-Industrial cooperation Society 20(10):478-485
7. Gwang-Seon Choe, Yeong-Gyeong Ham, Seon-Ho Kim (2013) Big Data Visualization. The Journal of Korean Society of Computer Information 21(1):33-43
8. Seung-Yeon Hwang, Kyung-Min Kwak, Dong-Jin Shin, Kwang-Jin Kwak, Young-J Rho, Kyung-won Park, Jeong-Min Park, Jeong-Joon Kim (2019) Analysis of Defective Causes in Real Time and Prediction of Facility Replacement Cycle based on Big Data. The Journal of The Institute of Internet, Broadcasting and Communication 19(6):203-212