Group Project:

A Scoring Model Based on Analytic Hierarchy Process (AHP) and Multinomial Logistic Regression Testing

Group Members: Xingyu Yao, Xinnan Zhou, Zhiyuan Ding

Mckelvey School of Engineering, WashU

FL2023.T81.INFO.574: Foundations of Analytics

Prof. Angelique Zeringue

December 14, 2023

Contents

1. Problem Statement 1

2. Background 2

2.1 Research Inspiration and Significance 2

2.1 Literature Review 2

3. Data Sources and Description 2

3.1 Data Sources 3

3.2 Data Description 3

3.2.1 Data Statistical Description 3

3.2.2 Sample Structure Analysis 4

4. Methods 5

4.1 Data Preprocessing 5

4.1.1 Credit Rank Segmentation 5

4.1.2 Data Merging 5

4.1.3 Feature Selection 6

4.1.4 f Feature Engineering 7

4.1.4.1 Standardization 7

4.1.4.2 Resample 7

4.1.5 Collinearity Diagnostics 8

4.2 Analytic Hierarchy Process (AHP) 9

4.2.1 Build Hierarchy Evaluation Model 9

4.2.2 Build Pairwise Judgement Matrix 10

4.2.3 Single Hierarchical Sorting and Consistency Check 11

4.2.3.1 Single Hierarchical Sorting 11

4.2.3.2. Solve for the Maximum Eigenvalue and C.I. Value 11

4.2.3.3. Consistency Check 12

4.2.4 Use Weight Vector to Calculate the Credit Score 12

4.3 Model Fitting for AHP Validation 13

4.3.1 Multinomial Logistic Regression 13

4.3.2 Neural Network 13

4.3.3 AdaBoost 14

4.3.4 Potential Problems of Dataset 14

5. Results 15

5.1 Credit Score Obtained from AHP 15

5.2 Comparison of Performance by Different Models 16

5.2.1 Performance of Multinomial Logistic Regression 16

5.2.2 Performance of Neural Network 17

5.2.3 Performance of AdaBoost 17

5.3 Credit Score of Final Validation Model 18

5.4 Test AHP Model by Root Mean Square Error (RMSE) 19

6. Discussion 19

Reference 22

Appendix 23

A Scoring Model Based on Analytic Hierarchy Process (AHP) and Multinomial Logistic Regression Testing

Group members: Xingyu Yao, Xinnan Zhou, Zhiyuan Ding

Date: December 14, 2023

1. Problem Statement

In this paper, we are going to combine the credit card application dataset and income dataset from Kaggle and build a scorecard model, followed by a test to prove its feasibility. The dataset contains various features of clients and their credit record which can separate customers into “Good”, “Fair” and “Bad” respectively. Based on the different features, the Analytic Hierarchy Process (“AHP” hereafter) will generate a score for each client, which can be verified by the logits of the result after fitting the multinomial logistic regression model.

2. Background

2.1. Research Inspiration and Significance

Interest rate margin and commission are the main sources of profit for credit card business. The degree of use of customer revolving credit determines the profit. The use of appropriate risk measurement methods and tools to make correct credit risk level judgments on applicants to determine reasonable card issuance and credit limits, has always been a worthwhile topic to explore. Therefore, we want to establish a reasonable customer segmentation system based on relevant databases and propose a credit scorecard model. The significance of this study lies in improving the credit scorecard model to help banks decide whether to issue credit cards based on the credit rating results of their customers, achieving the goal of effectively preventing credit risks.

2.2. Literature Review

In terms of the application of AHP, it was originally proposed in the 1970s by Thomas L. Saaty, a renowned American operations researcher and professor at the University of Pittsburgh. Saaty (2008) pointed out that AHP is the application of simple tools, combined with operations research ideas, to decompose complex problems into various constituent factors, and form a hierarchical structure by grouping them according to dominant relationships. By integrating the mutual influence relationships between various factors and their roles in the system, the relative importance of each factor is determined.

According to other researchers' studies, AHP has already been widely used in credit evaluation. Kokangül et al. (2017) conducted a risk assessment study on large manufacturing enterprises, determined the degree of harm based on experience, classified statistical records from the past 10 years, and applied AHP to rank each category. The relationship between risk level assessment and AHP points was studied, and the risk level interval of AHP was determined. Unutmaz Durmuşoğlu (2018) used AHP to analyze the factors that should be used in the evaluation of technology entrepreneurship projects. The model was tested and ranked based on real data containing the attributes and outcomes (success/failure) of 10 technology entrepreneurship projects. The results showed that three projects that had already failed were at the bottom of the list. Therefore, the proposed AHP model has been validated. The AHP framework proposed in this study is expected to be helpful to other fields as well. Oliver Gottfried et al. (2018) used the SWOT-AHP-tows analysis method to identify the strengths, weaknesses, opportunities, and threats (SWOT) of stakeholders, and then used AHP to determine priorities.

3. Data Sources and Description

3.1. Data Sources

The sample data used in this paper are all from the Kaggle database. One of our original datasets is Credit Card Approval Prediction ("CCAP" hereafter) with two sub-tables, one is application record ("CCAP-AR" hereafter), and the other is credit record ("CCAP-CR" hereafter). The other dataset is Income Dataset ("INC" hereafter).

3.2. Data Description

3.2.1. Data Statistical Description

For CCAP-AR, there are 438557 observations in total, it contains customers' different types of information, including client number, gender, whether own a car, whether own a property, number of children, etc. (see Feature description of "Credit Card Approval Prediction-Application Record" in Appendix). Also, there are 1048575 observations in total in CCAP-CR, which records the customers' credit card repayment status (see Feature description of "Credit Card Approval Prediction-Credit Record" in Appendix). For INC, there are also mostly demographic information of individuals, including work class, education, marital status, race, etc. (see Feature description of "Income Dataset" in Appendix)

For numerical data in the selected dataset, we generated statistical summary of INC (Table 1.1) and CCAP-AR (Table 1.2). The mean age of samples in INC is 38.61, with minimum of 17 and maximum of 90, which means the sample population covers adolescents to the elderly. Similarly, it can be inferred from the educational-num that the sample's educational level covers from primary school to doctoral degree. Also, from the hours-per-week, the sample on average works 40 hours per week.

Statistical Summary of Income Dataset

	age	fnlwgt	educational-num	capital-gain	capital-loss	hours-per-week
count	44856	44856.00	44856	44856	44856	44856
mean	38.616328	189762.70	10.076355	1086.251583	88.721152	40.421995
std	13.721436	105804.70	2.573193	7519.761299	405.401776	12.400502
min	17	13492.00	1	0	0	1
25%	28	117528.80	9	0	0	40
50%	37	178211.00	10	0	0	40
75%	48	237754.00	12	0	0	45
max	90	1490400.00	16	99999	4356	99

For statistical summary of CCAP-AR, the sample on average earns $187,524.3 per year and with high standard deviation of 110086.9, indicating the income of the sample fluctuates greatly.

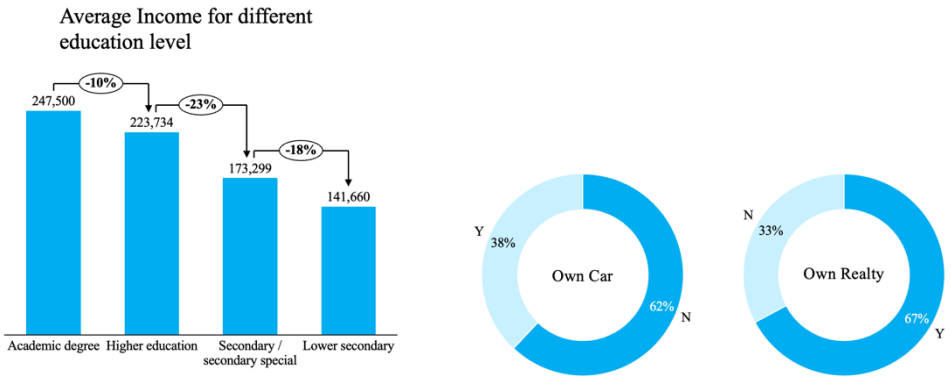
Statistical Summary of Credit Card Approval Prediction-Application Record

	CNT_CHILDREN	AMT_INCOME_TOTAL	DAYS_BIRTH	DAYS_EMPLOYED	FLAG_MOBIL	FLAG_WORK_PHONE	FLAG_PHONE	FLAG_EMAIL	CNT_FAM_MEMBERS
count	438557	438557.00	438557	438557	438557	438557	438557	438557	438557
mean	0.42739	187524.30	-15997.90465	60563.67533	1	0.206133	0.287771	0.108207	2.194465
std	0.724882	110086.90	4185.030007	138767.7996	0	0.404527	0.452724	0.310642	0.897207
min	0	26100.00	-25201	-17531	1	0	0	0	1
25%	0	121500.00	-19483	-3103	1	0	0	0	2
50%	0	160780.50	-15630	-1467	1	0	0	0	2
75%	1	225000.00	-12514	-371	1	0	1	0	3
max	19	6750000.00	-7489	365243	1	1	1	1	20

3.2.2. Sample Structure Analysis

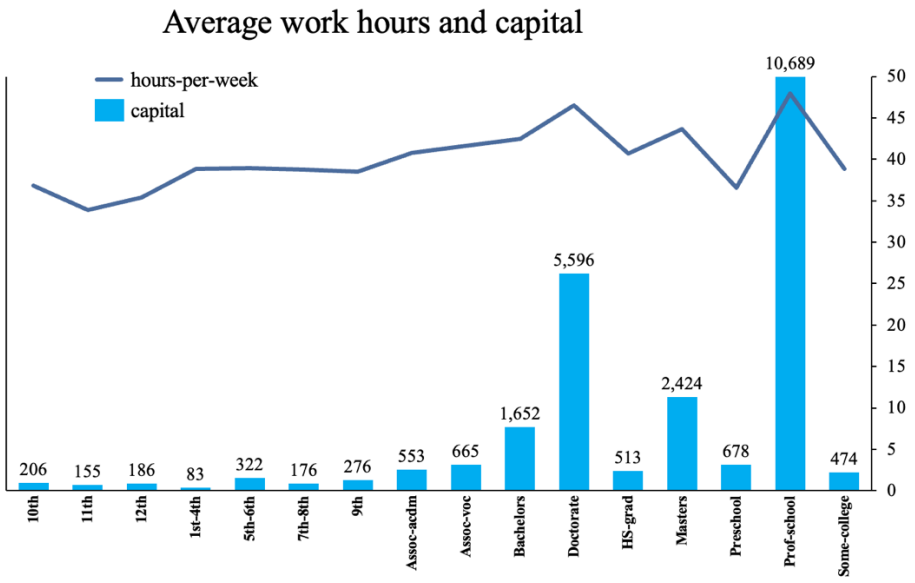
Since we use CCAP as our base dataset, our sample structure will focus on CCAP.

For asset possession status, 38% of the sample own a car and 62% don't, 67% of the sample own a realty while 33% don't. For education level, the average annual income varies by level of education, as the level of education rises, so does the average income.



As for the income dataset, we can obviously see that different education level has

different average values of work hours per week and the capital gain.



4. Methods

4.1. Data Preprocessing

4.1.1. Credit Rank Segmentation

Since we need to create labels for logistic regression, we labeled each applicator with different credit ranks. Based on CCAR-CR, we have established following credit rating standards:

Credit Rank	**Segmentation Standard**
Good	Pay bill in time over 70% of the time (Over 70% “C” in the credit record)
Fair	No usage record (All credit records of one person are “X”)
Bad	The rest

4.1.2. Data Merging

We merged CCAP and INC based on education because in practical situations educational level affects other personal information (People with the same level of

education are likely to share the same characteristics). For CCAP, there are 5 categories in education type, while there are 16 categories in INC, so we first unified categories of education based on the following corresponding rule:

Unification Rule of Merging by Education

Categories after unification	**Original categories**
Lower secondary	1st-4th, 5th-6th, 7th-8th, Preschool
Secondary/secondary special	9th, 10th, 11th, 12th, HS-grad, Incomplete higher
Higher education	Some-college, Assoc-voc, Assoc-acdm
Academic degree	Bachelors, Masters, Doctorate, Prof-school

Then we started to merge the two datasets. Initially, we followed Professor Angelique's instruction to take majority (for categorical variables) or average (for continuous variables) of each feature in INC and fill the CCAP with the majority or average results based on education categories. However, by doing so, incredibly high collinearity exists. As a result, we decided to refill the occupation feature by randomly generating data points following the occupation categories' distribution of each education level. After the combination of two datasets, we got the new dataset.

4.1.3. Feature Selection

After observing the combined dataset, we detected features that only have one category due to data merging, and features which we considered would either bring bias or have little impact on credit card approval consideration. Firstly, we noticed that there's only one category of "private" under the feature of "workclass", so we dropped this column. Secondly, we went through each feature and dropped those biased ones, such as "race" because taking applicants' race into consideration when deciding whether to approve the application is racism. Also, we dropped features which we think have little relationship with credit card application, such as CNT_CHILDREN. Also, we dropped overlapped features in INC, and keep the column in CCAP. After feature selection, our final dataset contains the following features: ID,

NAME_EDUCATION_TYPE, CODE_GENDER, FLAG_OWN_CAR, FLAG_OWN_REALTY, AMT_INCOME_TOTAL, NAME_INCOME_TYPE, NAME_FAMILY_STATUS, age, occupation, capital, hours-per-week and credit_decision.

4.1.4. Feature Engineering

4.1.4.1. Standardization

In the final dataset, we found that different variables have different scales, which may affect the results of data analysis. To eliminate the scale influence between variables, data standardization is needed to achieve the comparability between variables. For numerical variables, we applied StandardScaler() in python to normalize them to make sure all numerical variables are in the same order of magnitude, making it suitable for comprehensive comparative evaluation. For categorical variables, we applied One-Hot-Encoding, which uses an N-bit status register to encode N states, each with its own independent register bit, and at any time, only one bit is valid (by get_dummies() in python) to standardize them, and we drop one of the columns in the dummy variable to use as the reference category. The main benefits of doing so are that it solves the problem of classifiers not being able to handle categorical data well, and to a certain extent, it also expands features.

4.1.4.2. Resample

Because there is a category imbalance in the dataset, i.e., the number of people with a credit status of “Bad” is much larger than the number of people with a credit status of “Good” or a credit status of “Fair”. If we directly fit the model with the original data set, it will cause the model to be unable to obtain minority class prediction results, so we have to resample the data.

The first way is Random Under-sampling that is randomly removing sample points

from the majority class to reduce the number of samples from the majority class to a same number as the minimum class. However, this needs to be done carefully as under-sampling may result in loss of information. The second way is Random Over-Sampling that is randomized simple replication of minority class sample points to expand the number of minority class samples to the same number as the class with the largest sample size. However, since the method only simply replicates existing minority class samples to expand the number, it is likely to lead to overfitting. The third way is Synthetic Minority Over-sampling Technique ("SMOTE" hereafter) which is a more sophisticated over-sampling technique. Instead of simply replicating the minority class samples, it generates new, synthetic minority class samples. This is accomplished by interpolating between the minority class samples, which helps to increase the diversity of the samples and reduce the risk of overfitting. The final way we tried is Adaptive Synthetic Sampling ("ADASYN" hereafter) which is very similar to SMOTE, but the characteristic of ADASYN is that it adaptively generates more synthetic samples according to the learning difficulty of each minority class sample.

Despite using the four methods mentioned above, the final model fitting results is unsatisfactory. These four methods all improved the data set to a similar degree, and the model's final precision for each class was not improved very well.

4.1.5. Collinearity Diagnostics

After resampling the data, we use VIF to measure the collinearity of variables. When we merge data tables, each feature is aggregated according to education level. Therefore, each feature variable must have strong collinearity with the education level. So, when calculating VIF, we choose to eliminate the dummy variables generated by the education feature. Initially, we used the majority occupation in each education level to represent the occupations of everyone in that level, but this would lead to strong collinearity among the variables. For example, all the VIF of occupation_Prof-specialty, age, capital, hours-per-week, occupation_Other-service, occupation_Craft-repair are

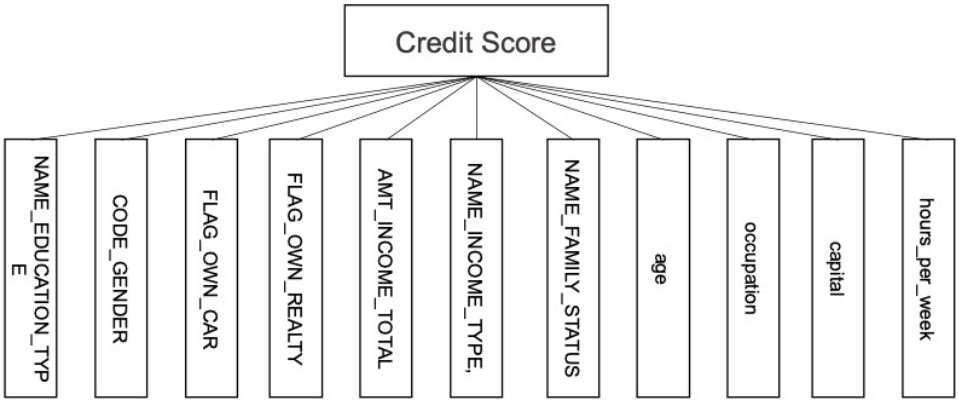
infinity. And there are another two variables DAYS_EMPLOYED 281.86, NAME_INCOME_TYPE_Pensioner 281.66 also have very high VIF.

Then, we chose to refill the occupation feature by randomly generating data points following the occupation categories' distribution of each education level. This method significantly reduces collinearity among variables. In the end, there were only three variables (VIF>5) with significant collinearity, namely NAME_INCOME_TYPE_Pensioner 87.74, DAYS_EMPLOYED 77.70, NAME_FAMILY_STATUS_Married 8.06. We chose to drop these columns which have high VIF value.

4.2. Analytic Hierarchy Process (AHP)

This section methodically delineates the Analytic Hierarchy Process, a pivotal decision-making tool developed by Thomas L. Saaty. AHP initially deconstructs phenomena or problems into relevant factors based on their characteristics, forming a multi-level structural model categorized by the relationships among these factors. Subsequently, through empirical analysis or expert judgment, the relative significance of lower-level factors to higher-level ones is evaluated and measured. This process yields the weight of features according to the relative significance, thereby facilitating quantitative analysis and comparison.

4.2.1. Build Hierarchy Evaluation Model



The top layer is the target layer, since we aim to derive credit scores using a linear weighted model based on the feature weights obtained from the AHP, our target layer is the credit score.

For the second layer, which is the criterion layer, we firstly applied expert diagnostic method to select features (We contacted one of our relatives working in bank to get advice.) and secondly, we tested for collinearity to finally keep the above 11 features. These features will also be used in the following validation model.

4.2.2. Build Pairwise Judgement Matrix

Constructing the judgment matrix involves pairwise comparisons between each feature and determining the weights of the criterion layer relative to the goal layer. Simply put, it's about conducting pairwise assessments of the criteria layer's indicators, typically employing Saaty's 1-9 scale method to provide evaluations.

Scale	Definition
1	Two elements have equal importance compared to each other
3	The former element is slightly more important than the latter
5	The former is clearly more important than the latter
7	The former is strongly more important than the latter
9	The former is extremely important than the latter
2，4，6，8	The intermediate value of the above adjacent judgments
The reciprocal of 1-9	The importance of the corresponding two factors when order is exchanged

For the 11 features we selected, we can build a 11x11 judgement matrix. The diagonal line is the judgment of each feature on itself, so they are all 1. For example, for NAME_EDUCATION_TYPE and NAME_EDUCATION_TYPE, the importance must be 1. And the NAME_EDUCATION_TYPE is clearly more important than the CODE_GENDER.

	NAME_EDUCATION_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	AMT_INCOME_TOTAL	NAME_INCOME_TYPE	NAME_FAMILY_STATUS	age	occupation	capital	hours-per-week
AME_EDUCATION_TYPE	1.00	0.20	0.33	3.00	5.00	1.00	0.20	3.00	1.00	3.00	0.33
CODE_GENDER	5.00	1.00	5.00	5.00	9.00	5.00	2.00	3.00	3.00	9.00	5.00
FLAG_OWN_CAR	3.00	0.20	1.00	5.00	3.00	2.00	0.33	0.20	1.00	3.00	0.33
FLAG_OWN_REALTY	0.33	0.20	0.20	1.00	0.50	0.20	0.14	0.14	0.20	0.33	0.20
AMT_INCOME_TOTAL	0.20	0.11	0.33	2.00	1.00	0.20	0.14	0.20	0.33	0.33	0.20
NAME_INCOME_TYPE	1.00	0.20	0.50	5.00	5.00	1.00	0.33	1.00	2.00	5.00	1.00
NAME_FAMILY_STATUS	5.00	0.50	3.00	7.00	7.00	3.00	1.00	1.00	5.00	5.00	3.00
age	0.33	0.33	5.00	7.00	5.00	1.00	1.00	1.00	3.00	5.00	1.00
occupation	1.00	0.33	1.00	5.00	3.00	0.50	0.20	0.33	1.00	3.00	0.33
capital	0.33	0.11	0.33	3.00	3.00	0.20	0.20	0.20	0.33	1.00	0.33
hours-per-week	3.00	0.20	3.00	5.00	5.00	1.00	0.33	1.00	3.00	3.00	1.00

4.2.3. Single Hierarchical Sorting and Consistency Check

4.2.3.1. Single Hierarchical Sorting

Hierarchical single sorting refers to the pairwise comparison of all elements within a layer relative to an element in the preceding layer, followed by hierarchical ordering to establish a sequence of importance. The specific calculations can be conducted based on the judgment matrix A, ensuring compliance with the eigenvalue and eigenvector condition represented by AW=$\lambda maxW$. This process involves an analytical approach to derive a structured priority ranking, crucial for decision-making in multi-criteria environments.

In simple words, hierarchical single sorting is to solve the weight of each feature based on the judgment matrix we formed. We can calculate its weight using the square root method. First, we should calculate the 1/m power of the product of each row to obtain an m-dimensional vector.

$$\overline{W_i} = \sqrt[m]{\prod_{j=1}^{m} a_{ij}}$$

Then we normalize the above vector, we can get the weight vector.

$$w_i = \frac{\overline{W_i}}{\sum_{j=1}^{m} \overline{W_j}}$$

Therefore, we can get the weight vector of each feature: [0.06324513, 0.2670073, 0.06735867, 0.01720726, 0.01998436, 0.08033522, 0.18545356, 0.11098794, 0.05629297, 0.02864629, 0.1034813].

4.2.3.2. Solve for the Maximum Eigenvalue and C.I. Value

After obtaining the weight matrix, we can solve for the maximum eigenvalue λmax. n is the number of dimensions of weight matrix is 11. AW is judgement matrix A * standardized weight W, then we can get our λmax is equal to $((AW)_1/W_1$ +

$(AW)_2/W_2 + (AW)_3/W_3 + \cdots + (AW)_n/W_n)/n = 12.379084394285133$.

$$\lambda_{max} = \frac{1}{n}\sum_{i=1}^{n}\frac{(AW)_i}{W_i}$$

After we get the maximum eigenvalue λmax, we can calculate the C.I. value, which equals to 0.13790843942851333.

$$C.I. = \frac{\lambda_{max} - n}{n - 1}$$

4.2.3.3.Consistency Check

The purpose of the consistency check is to determine whether there are logical problems in the constructed judgment matrix. For example, there are three features A, B, and C. If A is slightly more important than B (3) and C is slightly more important than A (1/3), then the relative importance of C to B should be an integer, otherwise there will be a logic error. And we use the C.R value to determine whether the consistency check is passed.

$$C.R. = \frac{C.I.}{R.I.}$$

The R.I values can be obtained by referring to the below table, which is derived from Satty's simulation of 1000 iterations.

Matrix Order n	1	2	3	4	5	6	7	8	9	10	11	12	13
R.I.	0	0	0.58	0.9	1.12	1.24	1.32	1.41	1.45	1.49	1.51	1.54	1.56

Therefore, we can get our C.R. value is 0.0913300923367638 < 0.1 which indicates that the consistency degree of our judgment matrixA is considered to be within the allowable range, and the weight vector derived from judgment matrixA is available at this time.

4.2.4. Use Weight Vector to Calculate the Credit Score

We use a linear weighted model to calculate credit scores. Continuous variables such as AMT_INCOME_TOTAL, capital, etc. are directly multiplied by their weights.

But for categorical variables such as NAME_EDUCATION_TYPE, FLAG_OWN_CAR, etc., we process them with one-hot encoding. Then, we should equate the original weights to each sub-variable generated by one-hot encoding. For example, if the original weight is 0.3 and the categorical variable has 3 categories, then each one-hot encoded sub-variable can be given a weight of 0.1. After that, we can calculate the credit score generated by categorical variables and add the credit scores obtained by continuous variables.

Finally, we use the Min-Max Scaling method to scale the resulting credit scores to a range from 0 to 100. The formula of this method is:

$$Scaled\ Value = \frac{Original\ Value - Min}{Max - Min} \times (New\ Max - New\ Min) + New\ Min$$

where "New Max" is 100, "New Min" is 0, "Min" and "Max" represent the minimum and maximum values in the original data respectively.

4.3. Model Fitting for AHP Validation

4.3.1. Multinomial Logistic Regression

Since the dataset contains 3 categories in outcome, we applied multinomial logistic regression model to fit the resample data. After that, we used the fitted model to predict the original dataset and got the final prediction result. We used the logistic regression model (LogisticRegression(multi_class='multinomial', solver='lbfgs', max_iter=2000)) from Scikit-learn library in Python and identified the multi class as multinomial. As it is an easy-to-use machine learning package, we can conduct the multinomial logistic regression model easily. In addition, we set the number of iterations to 2000 to make sure the model converges.

4.3.2. Neural Network

In addition to the traditional logistic regression model, our team tried the deep

neural network based on Pytorch. We started with a simple network architecture with only two hidden layers connected with ReLU() function, and the first hidden layer had only 64 neurons. Then we gradually added the Normalization Layer, added two more hidden layers, and increased the neurons of the first hidden layer to 512.

Finally, we set 4 hidden layers of the neural network and connected them with ReLU activation functions. In the output layer, 3 neurons are connected since we have 3 different credit ranks for clients. The Adam algorithm is used to be the optimizer and the cross-entropy loss function will be used to evaluate the performance of the model and calculate the logits. In addition, the early stopping mechanism is also applied to avoid overfitting.

4.3.3. AdaBoost

To improve the impact of class imbalance, we also tried the AdaBoost algorithm based on multinomial logistic regression. In the AdaBoost algorithm, several multinomial logistic regression models are trained sequentially, with each model trying to correct the errors of the previous model. During this process, it adjusts the weights of samples to focus on samples that are difficult to classify, making it possible to fix the problem of sample class imbalance.

We used the AdaBoostClassifier(estimator=logreg, n_estimators=500, random_state=42, algorithm='SAMME') from Scikit-learn library in Python and identified the estimator to be multinomial logistic regression models. In addition, we set the maximum number of estimators at which boosting is terminated to be 500 and we chose to use the SAMME discrete boosting algorithm.

4.3.4. Potential Problems of Dataset

Firstly, the dataset presents a concern wherein two clients possessing identical features exhibit disparate credit ranks. This incongruity impacts the precision of

predictive models reliant on features for credit rank prognosis. The divergence in credit ranks, despite identical feature sets, introduces a challenge to the accuracy of the predictive model.

Secondly, during the combination of the dataset, veracity is compromised due to the utilization of an aggregation-based merging technique. The combination process, being post-aggregation, entails a consequential loss of data authenticity. Concurrently, the diminished representation of the education category contributes to a reduction in diversity within the merged dataset. This limitation stems from the inherent scarcity of education categories.

Thirdly, although the CCAP-AR data set has sufficient data volume, the number of samples of CCAP-CR is not enough, resulting in insufficient amount of data available and imbalance of data categories in the end. And under the premise of considering the practical significance, even if different credit rank segmentation methods are formulated, the problem of classification imbalance cannot be well solved.

Last but not least, while resampling data augmentation techniques like SMOTE and ADASYN can help resolve inter-class imbalances, they may not resolve intra-class imbalances. Even if the number of samples is increased in the minority class, these samples may still surround the minority core region without fully exploring the entire feature space. Also, during the resampling process, noisy data may be introduced, especially when similar minority class samples may represent different subcategories.

5. Results

5.1. Credit Score Obtained from AHP

There are 36467 data in total, the first column “ID” represents the ID of customers. The second column refers to the original credit scores we directly got from the AHP model. The third column means the credit score scaled to 0-100.

Credit Score Obtained from AHP

	ID	credit_scores	scaled_credit_scores
0	5008804	0.300667987	10.61151217
1	5008805	0.300667987	10.61151217
2	5008806	0.443697434	13.88903291
3	5008808	0.175293436	7.73855349
4	5008809	0.175293436	7.73855349
...	...	...	...
36452	5149828	0.483454966	14.80007701
36453	5149834	-0.121528283	0.93688181
36454	5149838	-0.11348643	1.121160921
36455	5150049	0.142904492	6.996360611
36456	5150337	0.430114768	13.57778602

5.2. Comparison of Performance by Different Models

We compared the performance of three different models: directly using multinomial logistic regression, deep neural network and AdaBoost algorithm based on multinomial logistic regression.

5.2.1. Performance of Multinomial Logistic Regression

As we trained on the resampled dataset, which was generated by SMOTE, we evaluated the performance of the model on the original dataset and got the accuracy of 36.04%. Besides, the model tends to predict the bad category rather than the other two, which maybe the reason for the low accuracy.

Performance of Multinomial Logistic Regression

	precision	recall	f1-score	support
-1	0.66	0.39	0.49	23913
0	0.10	0.37	0.16	3347
1	0.26	0.28	0.27	9197
accuracy			0.36	36457
macro avg	0.34	0.35	0.31	36457
weighted avg	0.51	0.36	0.40	36457

5.2.2. Performance of Neural Network

From simple to complex neural network architectures, we did not achieve satisfactory results when training models on data resampled using the SMOTE method, as both training loss and validation loss remained consistently high. Although we added the Normalization Layer and increased the number of neurons in the hidden layer, we did not get a good result. This issue may be caused by an imbalance in the dataset's categories, and the dataset contains a certain amount of noise, outliers, or inaccurate data, leading the neural network to learn incorrect patterns and resulting in a decline in performance. In addition, this may also be due to the fact that the dataset itself is relatively simple and is not suitable for the neural network model.

```
Epoch: 1/100    Training Loss: 0.854568         Validation Loss: 0.856278
Epoch: 2/100    Training Loss: 0.844582         Validation Loss: 0.849434
Epoch: 3/100    Training Loss: 0.843140         Validation Loss: 0.848812
Epoch: 4/100    Training Loss: 0.840969         Validation Loss: 0.858959
Epoch: 5/100    Training Loss: 0.839857         Validation Loss: 0.850099
Epoch: 6/100    Training Loss: 0.838593         Validation Loss: 0.852293
Epoch: 7/100    Training Loss: 0.836690         Validation Loss: 0.860777
Epoch: 8/100    Training Loss: 0.835609         Validation Loss: 0.850318
Epoch: 9/100    Training Loss: 0.833826         Validation Loss: 0.852681
Epoch: 10/100   Training Loss: 0.832752         Validation Loss: 0.849652
Epoch: 11/100   Training Loss: 0.831114         Validation Loss: 0.854370
Epoch: 12/100   Training Loss: 0.829167         Validation Loss: 0.852306
Epoch: 13/100   Training Loss: 0.827067         Validation Loss: 0.855265
Epoch: 14/100   Training Loss: 0.824275         Validation Loss: 0.855527
Epoch: 15/100   Training Loss: 0.822634         Validation Loss: 0.849901
Epoch: 16/100   Training Loss: 0.819973         Validation Loss: 0.853149
Epoch: 17/100   Training Loss: 0.818207         Validation Loss: 0.849918
Epoch: 18/100   Training Loss: 0.815650         Validation Loss: 0.856238
Epoch: 19/100   Training Loss: 0.814652         Validation Loss: 0.854443
Epoch: 20/100   Training Loss: 0.811816         Validation Loss: 0.870193
Epoch: 21/100   Training Loss: 0.810605         Validation Loss: 0.856319
Epoch: 22/100   Training Loss: 0.808327         Validation Loss: 0.862251
```

5.2.3. Performance of AdaBoost

We trained the AdaBoost model on the resampled dataset, which was generated by SMOTE. Then we evaluated the performance of the AdaBoost model on the original dataset and got the accuracy of 35.95% which is very similar to the accuracy of directly using Multinomial Logistic model. This indicates that AdaBoost does not effectively

address the issue of data category imbalance, and the complexity of the AdaBoost algorithm reduces the model's interpretability.

Performance of AdaBoost

	precision	recall	f1-score	support
-1	0.67	0.34	0.45	23913
0	0.11	0.3	0.16	3347
1	0.26	0.42	0.32	9197
accuracy			0.36	36457
macro avg	0.35	0.36	0.31	36457
weighted avg	0.52	0.36	0.39	36457

Taking various factors such as model interpretability, model complexity, and model excellence into account, we finally chose to use the multinomial logistic regression model directly.

5.3. Credit Score of Final Validation Model

The final validation model we choose is Multinomial Logistic Regression model. We first get the model coefficients and model intercepts of three ranks respectively. Then we can solve for total logits by "total_logits = X @ model.coef.T + model.intercept_". After we get the total logits, we can find the largest logits of each data point among the three ranks as its credit score. In the end, we also use the Min-Max Scaling method to scale the resulting credit scores to a range from 0 to 100.

Credit Score from Logistic Regression

	ID	credit_scores	scaled_credit_scores
0	5008804	0.090096492	5.751621871
1	5008805	0.090096492	5.751621871
2	5008806	0.097470594	6.229622593
3	5008808	0.111753491	7.155462089
4	5008809	0.111753491	7.155462089
…			
36452	5149828	0.11167718	7.150515532
36453	5149834	0.215281553	13.8663116
36454	5149838	0.040538145	2.539172952
36455	5150049	0.04562705	2.869043683
36456	5150337	0.087009886	5.551543276

5.4. Test AHP Model by Root Mean Square Error (RMSE)

We can see the summary statistics of the credit scores of AHP model and Multinomial Logitics Regression model below (Left is AHP, Right is Multinomial Logitics Regression). It shows that the means of credit scores and scaled credit scores of both models are very close. And the RMSE between the credit score of AHP model and Logistic model is 6.8176962730222535, being not very high, which means our model is reasonable to some extent.

	ID	credit_scores	scaled_credit_scores
count	3.645700e+04	36457.000000	36457.000000
mean	5.078227e+06	0.182842	7.911534
std	4.187524e+04	0.205181	4.701733
min	5.008804e+06	-0.162413	0.000000
25%	5.042028e+06	0.083289	5.630276
50%	5.074614e+06	0.149912	7.156931
75%	5.115396e+06	0.312427	10.880970
max	5.150487e+06	4.201539	100.000000

	ID	credit_scores	scaled_credit_scores
count	3.645700e+04	36457.000000	36457.000000
mean	5.078227e+06	0.102892	6.581065
std	4.187524e+04	0.080086	5.191307
min	5.008804e+06	0.001366	0.000000
25%	5.042028e+06	0.048865	3.078930
50%	5.074614e+06	0.082338	5.248690
75%	5.115396e+06	0.138461	8.886708
max	5.150487e+06	1.544063	100.000000

6. Discussion

In our analysis, the Root Mean Squared Error (RMSE) attained a value of 6.8176962730222535, which, within the context of our study, can be interpreted as a moderately favorable outcome. This RMSE level indicates that the prediction results of the AHP model have certain reasonableness and credibility when juxtaposed with the potential range of the dataset. It suggests that the AHP model has successfully captured the underlying patterns of this dataset to a significant extent, without being overly precise or overly generalized. In addition, it also shows that the relative importance of the pairwise features we built in the AHP model is relatively reasonable.

However, the accuracy of the multinomial logistic regression model was not particularly high, which may be attributed to various reasons previously mentioned, such as imbalances in data labeling and the presence of data with identical features but differing labels. Consequently, employing this multinomial logistic regression model to

validate our AHP model has certain limitations. It may not effectively verify the rationality of the AHP model we constructed. This scenario underscores the complexity inherent in model validation processes and highlights the necessity of considering a range of factors that might influence the performance and applicability of statistical models in real-world scenarios.

A second potential issue that may arise is the relative subjectivity in the construction of the judgment matrix within our AHP model. Despite the assessment through expert diagnosis, a degree of irrationality may persist. The Consistency Ratio (C.R) value of 0.09, barely below the threshold of 0.1, indicates that the consistency of our constructed judgment matrix is acceptable but not exceptionally high. This could lead to issues concerning the applicability and generalizability of our constructed AHP model in real-world scenarios. It highlights the challenges in ensuring objectivity and reliability in models heavily reliant on subjective evaluations and expert judgments.

We hope to have more time in the future to make some improvements to this project and also hope to find a more reasonable data set to build and verify our model. To address the identified issues in our dataset, which includes data label imbalances, the presence of noise, outliers, or inaccuracies, necessitates a multifaceted approach for further improvement. Firstly, implementing advanced techniques for balancing data, such as synthetic minority over-sampling or targeted under-sampling, could mitigate the effects of label imbalances. Enhancing data preprocessing to rigorously identify and handle noise and outliers is also crucial. Moreover, exploring alternative models (such as Random Forest) that better align with the data features could improve performance. The subjective nature of our model construction calls for a more systematic approach, possibly integrating more objective data-driven methodologies or leveraging ensemble techniques to reduce biases. This holistic strategy aims to refine our model's robustness, ensuring higher accuracy and reliability in diverse application scenarios.

In conclusion, the application of the Analytic Hierarchy Process (AHP) model in the realm of credit scoring signifies a pivotal advancement. Its capability to transform subjective assessments into quantifiable weights allows for more nuanced, data-informed decisions in credit evaluations. The utilization of multinomial logistic

regression to validate the rationality of the AHP model also holds substantial practical significance. This approach offers a methodological juxtaposition, wherein the multinomial logistic regression model, grounded in statistical probabilities, serves as a robust tool to assess the efficacy of the AHP's hierarchical and qualitative framework. This validation strategy not only underscores the importance of empirical testing in model evaluation but also highlights the synergy between qualitative decision-making processes and quantitative validation techniques. Such a methodological interplay is crucial in advancing the reliability and applicability of decision-making models in real-world scenarios, particularly in complex fields like credit scoring.

Reference

Gottfried, O., De Clercq, D., Blair, E., Weng, X., & Wang, C. (2018). SWOT-AHP-TOWS analysis of private investment behavior in the Chinese biogas sector. *Journal of Cleaner Production*, *184*, 632–647. https://doi.org/10.1016/j.jclepro.2018.02.173

Halosec_Wei. (2021, December 22). *用人话讲明白 AHP 层次分析法（非常详细原理+简单工具实现）_ahp 分析-CSDN 博客*. Blog.csdn.net. https://blog.csdn.net/qq_41686130/article/details/122081827

Kokangül, A., Polat, U., & Dağsuyu, C. (2017). A new approximation for risk assessment using the AHP and Fine Kinney methodologies. *Safety Science*, *91*, 24–32. https://doi.org/10.1016/j.ssci.2016.07.015

Saaty, T. L. (2008). Relative measurement and its generalization in decision making why pairwise comparisons are central in mathematics for the measurement of intangible factors the analytic hierarchy/network process. *Revista de La Real Academia de Ciencias Exactas, Fisicas Y Naturales. Serie A. Matematicas*, *102*(2), 251–318. https://doi.org/10.1007/bf03191825

Unutmaz Durmuşoğlu, Z. D. (2018). Assessment of techno-entrepreneurship projects by using Analytical Hierarchy Process (AHP). *Technology in Society*, *54*, 42–67. https://doi.org/10.1016/j.techsoc.2018.02.001

Appendix

Project URL:

https://github.com/Linkkk01/INFO574_Group_Project/tree/main

This includes all our raw data, processed data, and codes.

URL of Credit Card Approval Prediction:

https://www.kaggle.com/datasets/rikdifos/credit-card-approval-prediction

URL of Income Dataset:

https://www.kaggle.com/datasets/mastmustu/income/discussion

Feature description of "Credit Card Approval Prediction-Application Record"

	Feature name	**Explanation**	**Remarks**
1	ID	Client Number	
2	CODE_GENDER	Gender	
3	FLAG_OWN_CAR	Whether owns a car	
4	FLAG_OWN_REALTY	Whether owns a property	
5	CNT_CHILDREN	Number of children	
6	AMT_INCOME_TOTAL	Annual income	
7	NAME_INCOME_TYPE	Income category	
8	NAME_EDUCATION_TYPE	Education level	
9	NAME_FAMILY_STATUS	Marital status	
10	NAME_HOUSING_TYPE	Way of living	
11	DAYS_BIRTH	Birthday	Count backwards from current day (0), -1 means yesterday
12	DAYS_EMPLOYED	Start date of	Count backwards from current day (0). If

		employment	positive, it means the person currently unemployed.
13	FLAG_MOBIL	Is there a mobile phone	
14	FLAG_WORK_PHONE	Is there a work phone	
15	FLAG_PHONE	Is there a phone	
16	FLAG_EMAIL	Is there an email	
17	OCCUPATION_TYPE	Occupation	
18	CNT_FAM_MEMBERS	Family size	
19	MONTHS_BALANCE	Record month	The month of the extracted data is the starting point, backwards, 0 is the current month, -1 is the previous month, and so on
20	STATUS	Status	0: 1-29 days past due 1: 30-59 days past due 2: 60-89 days overdue 3: 90-119 days overdue 4: 120-149 days overdue 5: Overdue or bad debts, write-offs for more than 150 days C: paid off that month X: No loan for the month

Feature description of “Income Dataset”

	Feature name	**Explanation**	**Remarks**
1	age	Age	
2	workclass	Work class	
3	fnlwgt	Final weight	An estimate of the number of individuals in

			the population with the same demographics as this individual.
4	education	Highest education level	
5	educational-num	Number of education in year	
6	marital-status	Marital status	
7	occupation	Career	
8	relationship	Role in a family	
9	race	Race	
10	gender	Gender	