The problem of glass composition analysis based on multiple regression and neural networks

summaries

This paper mainly focuses on the changes of chemical substances and the examination of the composition of high-potassium glass and lead-barium glass after weathering, collects relevant data and facts in websites and journal papers, establishes a series of mathematical analyses and prediction models by using multiple linear regression, cluster analysis, principal component analysis, neural network classification algorithms, and applies R language iteration, MATLAB, t-test, correlation coefficient matrix and other Mathematical methods to find the most reasonable results of each problem, with strong practical work guidance.

To address the first problem, quantitative analysis of various indicators of glass artifacts, in the process of exploring the relationship between multiple variables through the establishment of a **logistic model of multiple regression**, the use of **the R language** for the fitting of the final relationship between the variables to obtain a more reliable law. Secondly, on the basis of washing data, the mean value of each group of data and the distribution characteristics were obtained by using excel. Finally, by introducing the variable of weathering or not, a more accurate inference result was obtained by comparing the difference between the elemental content corresponding to the weathering area and the product of weathering condition and weathering coefficient.

For problem two, **cluster analysis was** performed based on the chemical composition data of the attached glass. Its elemental content was reduced to the state when it was not weathered, and a **neural network classification** algorithm was utilized on the basis of treating the sum of the proportions of the constituent elements as 100%. Subclassification was then performed using **principal component analysis** to assign scores to each feature quantity. After the principal components are tested for normality, the subclasses can be divided according to the mean value based on the characteristics of normal distribution.

For Problem 3, to identify the type of glass belonging to the premise of known glass composition, can use the classification model test of Problem 1, in the process of data processing, it will be transformed into the data before weathering, so as to identify the type of its belonging to more accurately, due to space reasons, will not be repeated here.

For the fourth problem, in order to avoid the interference of weathering on the data, the data were converted into weathering before for further processing of the data. The **correlation coefficients** and **P-values of the** different chemical compositions of the two types of glasses were solved using the R language. In order to compare the differences in the correlation relationship of chemical compositions between different categories, the **t-test was** used to obtain the differences in the correlation coefficients and P-value relationships of the same chemical compositions in different categories of glasses.

Keywords: glass identification, multiple linear regression, cluster analysis, covariance matrix

I. Restatement of the problem

1.1 Background to the issue

Ancient glass in the process of burial for a long time, very susceptible to the influence of the surrounding environment weathering, darkening, reduced transparency, halo color or the formation of weathering products on the outside of the crust cited. In the case of severely weathered glass, the surface is completely covered by weathering materials, and its original appearance is almost unrecognizable. From the perspective of chemical changes, in the process of weathering, internal elements and environmental elements will be exchanged, resulting in changes in the proportion of its composition, and the chemical composition of the weathered glass there is a certain degree of statistical regularity between the internal, and this statistical regularity is affected by such as the type of glass, weathering and other exogenous variables, and objectively reflected in the differences in color, decoration. This has caused great disturbance to the identification of glass types by archaeologists. There is a batch of relevant data of ancient glass products in China, and archaeologists have classified them into two types of high-potassium glass and lead-barium glass based on the chemical composition of these cultural relics samples and other detection means. According to the classification information of these artifacts and its corresponding proportion of the main components for analytical modeling, so as to solve the identification and deduction related to the type of glass and the content of the components.

Disconnected issues.

1.2 Formulation of the problem

Based on the above background, a model was developed to solve the following problem:

1. The weathering of these glass artifacts was analyzed in relation to their glass type, decoration and color and combined with their type to explore the statistical pattern of their chemical composition content. On this basis, the chemical composition content before weathering is predicted by using the detection data of weathering points.

2. Based on the attached data, we analyze the classification rules of high-potassium glass and lead-barium glass and select appropriate chemical compositions for each category to classify them into subclasses, and give specific classification methods and results. On this basis, the rationality and sensitivity of the classification results are analyzed.

3. The chemical composition of the glass artifacts of the unknown category in Form 3 of the Annex was analyzed to identify the type to which they belonged, and the sensitivity of the classification results was analyzed.

4. For different categories of glass artifact samples, the correlations between their chemical compositions were analyzed, and the differences in the correlations of chemical compositions between different categories were compared.

II. Analysis of issues

2.1 Analysis of question one

Before constructing the model, the data were first preprocessed to assign values to variables such as glass type, grain, and color. Based on the quantification of the indicators, a multivariate regression model with **0-1 variables** as dependent variables was used for analysis. Weathering was used as the dependent variable in the modeling process, and glass type, grain, and color were used as independent variables to fit the **multivariate logistics regression**. It is worth noting that this process requires the use of the R language to iterate the phenomenon. Since it is observed that the relationship between the dependent and independent variables is not linear, we chose to create an **S-curve** of logistic shape to transform the problem into a linear one, and then use **the least squares method to reduce the** loss. For the prediction problem of composition, we arrived at a more accurate inference by comparing the difference between the elemental content corresponding to the weathering area and the product of the weathering condition and the weathering coefficient.

2.2 Analysis of question two

Cluster analysis was used to investigate the classification rules of high-potassium glass and lead-barium glass. Before the operation, the data were first pre-processed, i.e., the elemental content was reduced to the state when it was not weathered, and the sum of the proportions of the constituent elements was summed up with excel, and processed to be 100% under the standard condition, and finally the classification model was constructed by using the classification algorithm of neural network. Then subclassification was carried out on the basis of clustering, and **principal component analysis** was used to retain the chemical components with larger standard deviation and discard the ones with smaller standard deviation, and principal components were obtained by using principal component analysis. The contribution of the first principal component point was obtained through MATLAB to be sufficiently high, so we used this as a criterion. On this basis, points were assigned to each variable to obtain the first principal component score. Due to the universality and widespread existence of the normal distribution, it is reasonable to assume that the vested data is normally distributed in order to simplify the analysis, and therefore the main components of the sample can be judged according to the level of the mean, thus enabling subclassification.

2.3 Analysis of question three

Problem three is similar to problem one and can be tested using the classification model from problem one, applying the transformation to pre-weathering treatment data to identify the type to which it belongs, which will not be repeated here for space reasons.

2.4 Analysis of question four

In order to avoid the interference of weathering on the data, the data were converted to be used before weathering, and the **correlation coefficients** and **P-values** of different chemical compositions were solved for the two categories of glass using the R language, respectively. For comparing the differences in the correlation relationship of chemical compositions between different categories, the **t-test was** utilized to test the differences in the correlation coefficients and P-value relationships of the same chemical compositions in different categories of glasses.

III. Description of symbols

Symbols/assignments	representativeness
$m = 1$	High Potassium Metals
$m = 0$	Lead barium metal
$c = 1$	Class C Tattoos
$c = 2$	Class B Tattoos
$c = 3$	Class A Tattoos
$y = 1$	light green
$y = 2$	greener
$y = 3$	pale blue
$y = 4$	blue-green
$y = 5$	dark green
$y = 6$	Deep Blue, chess-playing computer, first defeat reigning world champion, developed by IBM (1985-1997)
$y = 7$	violet (color)
$y = 8$	ferrous
$s = 0$	Not weathered
$s = 1$	weathering
$s1 = 0$	No weathering of artifact surfaces
$s1 = 1$	Unweathered surface of artifacts

$s1 = 2$	public morals
$s1 = 3$	severe weathering
$b0$	Coefficient 1
$b1$	Coefficient 2
$b2$	Coefficient 3

IV. Modeling and solving

4.1 Issue 1

4.1.1 Processing of data

Before constructing the model, the data were first preprocessed by assigning values to variables such as glass type $m = 1$, grain, and color $m = 0$ according to their different types. For high potassium metal $c = 2$ and for lead-barium metal $c = 1$; and for different decorations, the value $c = 3$ is assigned to the A type of decoration, to the B type of decoration , and to the C type of decoration ; and for different colors of

Metal, light green assignment $y = 1$, green assignment $y = 2$, light blue assignment $y = 3$, blue-green assignment $y = 4$, dark green assignment

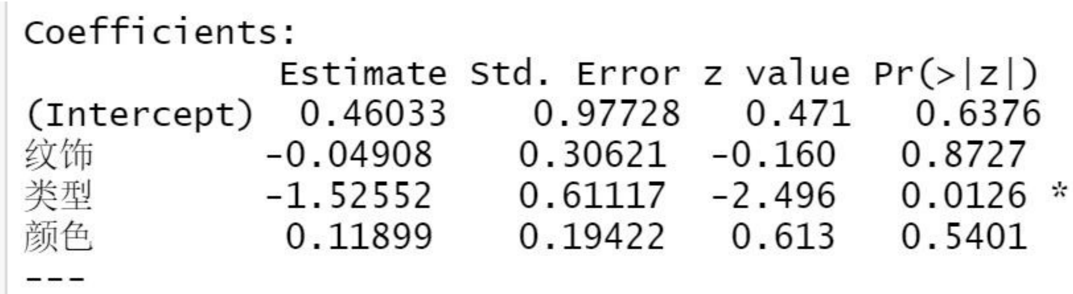
$y = 5$, dark blue is assigned to $y = 6$, purple is assigned to $y = 7$, and blac $s = 1$ igned to $y = 8$; and for weathering conditions, the

indicates that weathering occurred, and $s = 0$ indicates that weathering did not occur. As for the data with some missing attributes, the group did not delete them directly. Rather, considering the limited amount of data, the loss of other attribute values of this type of data will reduce the fitting effect of the model, so take the corresponding attribute of the assignment of the mean value to give the assignment, in order to reduce the impact on the accuracy of the condition, to ensure the number of samples.

At the same time, in order to combine the type of glass, to carry out the study of the statistical law of the surface of cultural relics with or without weathering chemical composition content. First of all, according to the requirements of the topic, we remove the chemical composition content and the data that are not in the range of 85%~100%, i.e., cultural relics sampling points 15 and 17. On the basis of the original weathering condition, we re-express the state of unweathered area on the surface of weathered cultural relics, as well as the state of the weathered layer on the surface of weathered cultural relics, and introduce a new weathering state variable. That is, no wind、$s1 = 0$, no wind、$s1 = 1$,wind、$s1 = 2$, severe wind、$s1 = 3$.

4.1.2 Modeling

For the first question in question one, i.e., the relationship between whether the surface is weathered or not and other factors, since the dependent variable weathering status is a 0-1 variable, a logistic model is used here for multiple regression analysis. Based on the R language, through programming, we obtained the relationship between weathering status and the type of ornamentation, metal color type, and metal type:



```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)  
(Intercept)  0.46033    0.97728   0.471   0.6376  
纹饰        -0.04908    0.30621  -0.160   0.8727  
类型        -1.52552    0.61117  -2.496   0.0126 *
颜色         0.11899    0.19422   0.613   0.5401  
---
```

To wit:

$$s =- 0.04908c - 1.52552m + 0.11899y + 0.46033$$

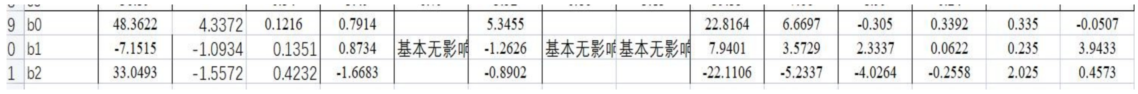
The equations for linear regression analysis with the content of different metals as the dependent variable were derived using weathering state and glass type as independent variables respectively. The solution is based on MATLAB software, meanwhile, in the process of solving the equation, we adopt the method of removing residual outliers for metals with sufficient amount of data, while for metals with less amount of data, we do not do this treatment. At the same time, there are some metals for which the test of the F-statistic was found to be problematic when solving the linear regression equation, i.e., it can be assumed that basically the content of the metal is not related to whether it is weathered or not, and therefore it is not included in the scope of our regression statistics. The rest of the metals passed our R-square test and F-statistic test.

muc

A linear regression model was developed as follows:

$$muc = b0 + b1s1 + b2m$$

Solving yields the coefficient matrix as follows:



9	b0	48.3622	4.3372	0.1216	0.7914		5.3455			22.8164	6.6697	-0.305	0.3392	0.335	-0.0507
0	b1	-7.1515	-1.0934	0.1351	0.8734	基本无影响	-1.2626	基本无影响	基本无影响	7.9401	3.5729	2.3337	0.0622	0.235	3.9433
1	b2	33.0493	-1.5572	0.4232	-1.6683		-0.8902			-22.1106	-5.2337	-4.0264	-0.2558	2.025	0.4573

Finally, for each point weathering before the prediction, that is, on the basis of each data now the content of each element to exclude the weathering of the

disruptions

$$muc^{'} = muc - s1 * b1$$

The elemental content was obtained and the results are shown in the "Before weathering" page of the fujian3.csv file in the Supporting Materials.

4.2 Issue 2

4.2.1 Processing of data

Since the chemical composition of the glass will be affected by whether or not it is weathered, the linear regression equation obtained in Problem 1 was first used to exclude the effect of weathering before converting all its contents into the sum of the percentages of each element to 100% to reduce the effect of measurement accuracy on the clustering results.

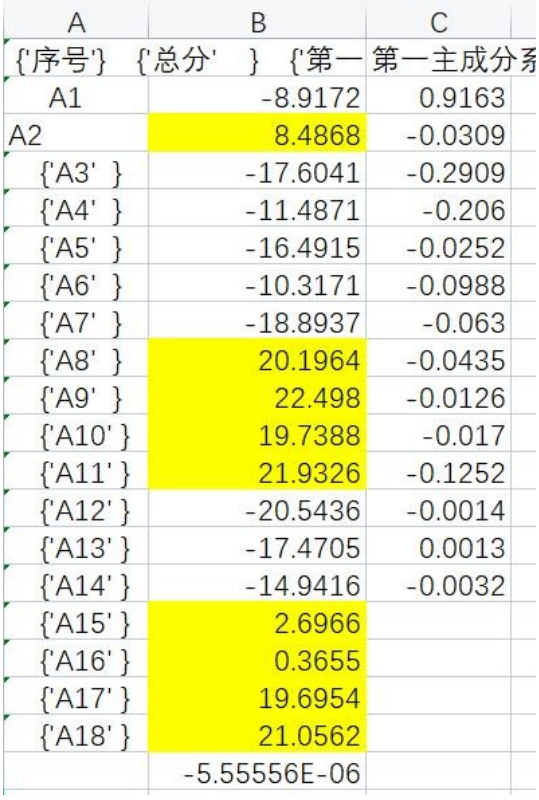
4.2.2 Modeling

Here, a more mature neural network model is used to cluster analyze the classification of glass, and several iterations are performed to obtain the classification criteria, and this is used to deal with problem three.

For the classification of subclasses, we adopted the principal component analysis approach, i.e., different weights were assigned to different metal contents to obtain principal components to maximize the variance of the random variable for different artifacts, i.e., to maximize the degree of differentiation, which was implemented using MATLAB.

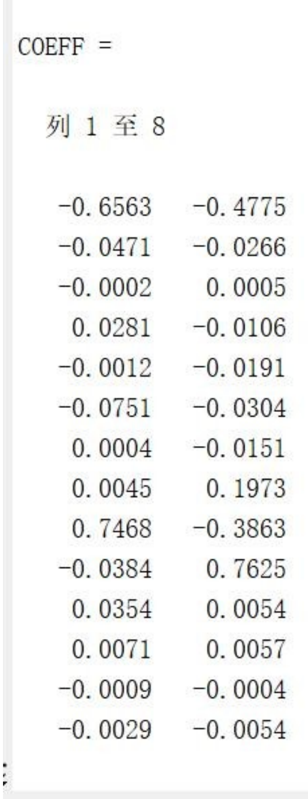
The corresponding number of principal components, coefficients, and corresponding principal component scores were obtained, respectively.

For high potassium glass, the contribution of the first principal component is then more than 85%, i.e. this is used as a criterion for classification. It can be assumed that the values of the first principal component for the high potassium glass artifacts are taken to be normally distributed, and therefore, their expectation can be used as a criterion for classification. Those lower than the expectation are one subcategory, and those higher than the expectation are another subcategory. The final classification obtained is shown in the following figure, where the only subcategory is labeled with yellow, and those not labeled with yellow are another subcategory:



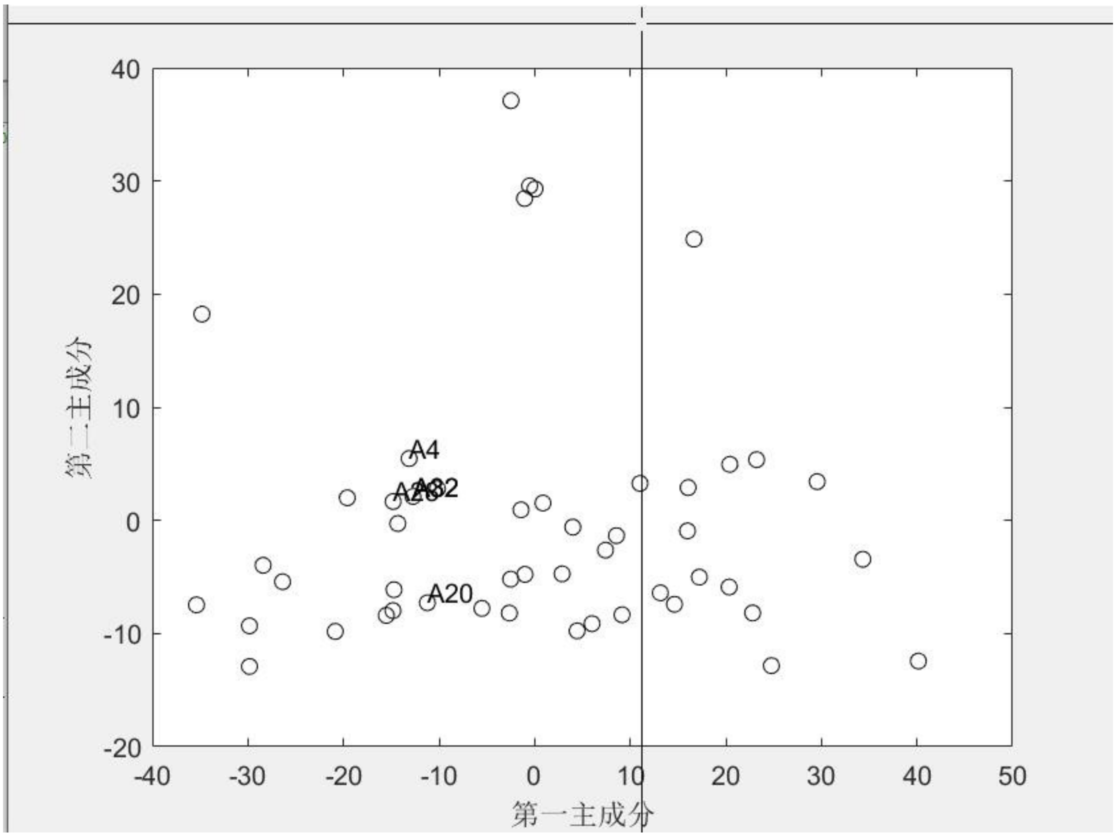
A	B	C
{'序号'} {'总分' } {'第一		第一主成分系
A1	-8.9172	0.9163
A2	8.4868	-0.0309
{'A3' }	-17.6041	-0.2909
{'A4' }	-11.4871	-0.206
{'A5' }	-16.4915	-0.0252
{'A6' }	-10.3171	-0.0988
{'A7' }	-18.8937	-0.063
{'A8' }	20.1964	-0.0435
{'A9' }	22.498	-0.0126
{'A10' }	19.7388	-0.017
{'A11' }	21.9326	-0.1252
{'A12' }	-20.5436	-0.0014
{'A13' }	-17.4705	0.0013
{'A14' }	-14.9416	-0.0032
{'A15' }	2.6966	
{'A16' }	0.3655	
{'A17' }	19.6954	
{'A18' }	21.0562	
	-5.55556E-06	

For lead-barium glass, there are two principal components with a contribution rate of more than 85%, and the corresponding principal component coefficients are as follows:



```
COEFF =

  列 1 至 8

   -0.6563   -0.4775
   -0.0471   -0.0266
   -0.0002    0.0005
    0.0281   -0.0106
   -0.0012   -0.0191
   -0.0751   -0.0304
    0.0004   -0.0151
    0.0045    0.1973
    0.7468   -0.3863
   -0.0384    0.7625
    0.0354    0.0054
    0.0071    0.0057
   -0.0009   -0.0004
   -0.0029   -0.0054
```

In turn, based on the scores of these two principal components, a scatterplot can be plotted as follows:



It is easy to see that the principal component scores are basically distributed on both sides of the straight $y = x$ and can be used as a basis for categorization by observing that the scores located in the

pro re a ds

The charge above the line The point below the line gives the corresponding subclass classification, see the supporting material image format file for the specific graphic.

Since we take the approach of principal component analysis, small changes in individual variables will not affect the classification results, with better sensitivity and model applicability.

4.3 Issue 3

4.3.1 data processing

The data were first examined to exclude interference from the variable of whether or not they were weathered.

4.3.2 data testing

Building on the foundation of Problem 2, the originally established neural network classification model was tested on Form 3 data with the following results:

Glass Artifact Number	Type of affiliation
A1	High Potassium Glass
A2	Lead and barium glass

A3	Lead and barium glass
A4	Lead and barium glass
A5	High Potassium Glass
A6	High Potassium Glass
A7	High Potassium Glass
A8	Lead and barium glass

4.4 Issue 4

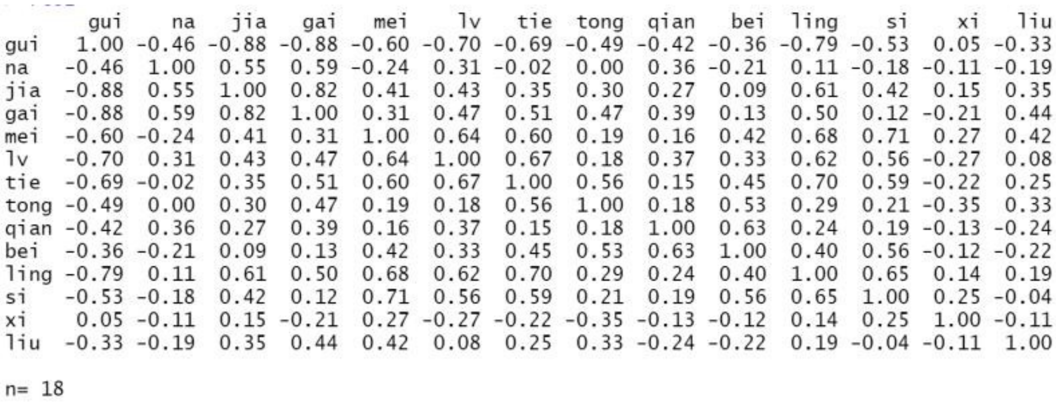
4.4.1 data processing

We analyze the correlation of the content of each chemical component separately for two different types of glass and have previously taken treatments that exclude the effects of weathering.

4.4.2 Modeling and solving

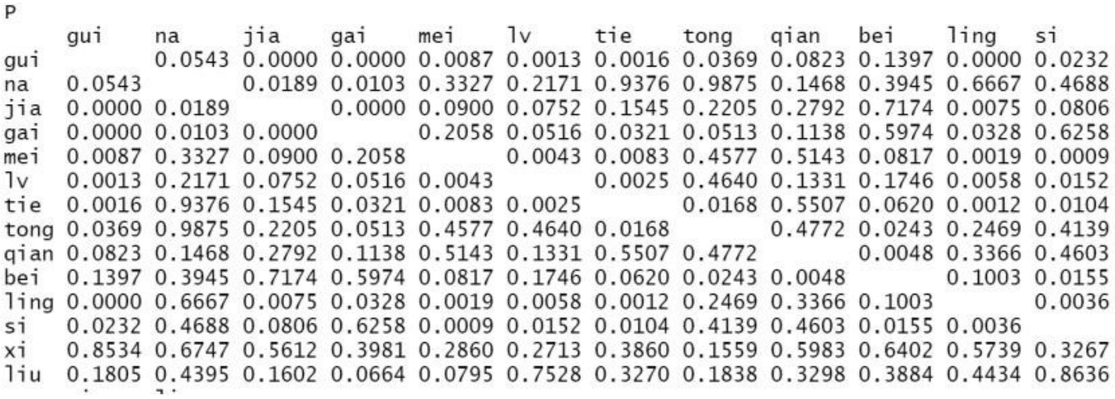
In a statistical sense, the P-value (P-Value, Probability, Pr) when hypothesis testing is an important piece of data and is another basis for making testing decisions. When the original hypothesis is true, the probability of a more extreme result than the sample observations obtained. If the P value is very small, it means that the probability of the occurrence of the original hypothesis is very small, and if it occurs, according to the principle of small probability, we have reason to reject the original hypothesis, the smaller the P value, the more we reject the original hypothesis. Therefore, the smaller the P-value, the more significant the result. However, whether the result of the test is "significant", "moderately significant" or "highly significant" needs to be solved by ourselves according to the size of the P-value and the actual problem.

The P-value obtained from the test of significance is generally $P < 0.05$ for statistically significant difference, $P < 0.01$ for statistically significant difference, and $P < 0.001$ for statistically extremely significant difference. This means that the probability that the difference between the samples is due to sampling error is less than 0.05, 0.01, 0.001. In fact, the P value does not give any significance to the data, but only indicates the probability of an event occurring. The statistic $Pr > F$ can also be written as $Pr(> F)$, $P = P\{ F0.05 > F\}$ or $P = P\{ F0.01 > F\}$. Therefore, here, the correlation coefficient matrix and p-value matrix are obtained using Rstudio, and the correlation coefficient matrix corresponding to the high potassium type is obtained as follows:



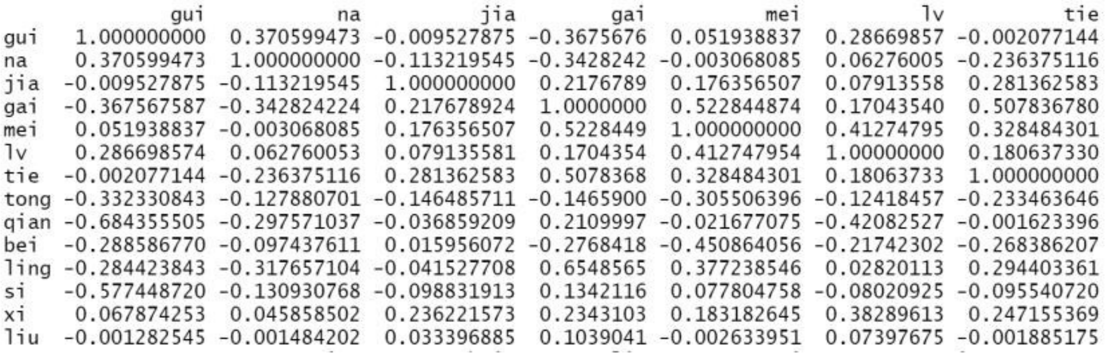
```
      gui    na   jia   gai   mei    lv   tie  tong  qian   bei  ling    si    xi   liu
gui   1.00 -0.46 -0.88 -0.88 -0.60 -0.70 -0.69 -0.49 -0.42 -0.36 -0.79 -0.53  0.05 -0.33
na   -0.46  1.00  0.55  0.59 -0.24  0.31 -0.02  0.00  0.36 -0.21  0.11 -0.18 -0.11 -0.19
jia  -0.88  0.55  1.00  0.82  0.41  0.43  0.35  0.30  0.27  0.09  0.61  0.42  0.15  0.35
gai  -0.88  0.59  0.82  1.00  0.31  0.47  0.51  0.47  0.39  0.13  0.50  0.12 -0.21  0.44
mei  -0.60 -0.24  0.41  0.31  1.00  0.64  0.60  0.19  0.16  0.42  0.68  0.71  0.27  0.42
lv   -0.70  0.31  0.43  0.47  0.64  1.00  0.67  0.18  0.37  0.33  0.62  0.56 -0.27  0.08
tie  -0.69 -0.02  0.35  0.51  0.60  0.67  1.00  0.56  0.15  0.45  0.70  0.59 -0.22  0.25
tong -0.49  0.00  0.30  0.47  0.19  0.18  0.56  1.00  0.18  0.53  0.29  0.21 -0.35  0.33
qian -0.42  0.36  0.27  0.39  0.16  0.37  0.15  0.18  1.00  0.63  0.24  0.19 -0.13 -0.24
bei  -0.36 -0.21  0.09  0.13  0.42  0.33  0.45  0.53  0.63  1.00  0.40  0.56 -0.12 -0.22
ling -0.79  0.11  0.61  0.50  0.68  0.62  0.70  0.29  0.24  0.40  1.00  0.65  0.14  0.19
si   -0.53 -0.18  0.42  0.12  0.71  0.56  0.59  0.21  0.19  0.56  0.65  1.00  0.25 -0.04
xi    0.05 -0.11  0.15 -0.21  0.27 -0.27 -0.22 -0.35 -0.13 -0.12  0.14  0.25  1.00 -0.11
liu  -0.33 -0.19  0.35  0.44  0.42  0.08  0.25  0.33 -0.24 -0.22  0.19 -0.04 -0.11  1.00

n= 18
```

The p-value matrix is as follows:

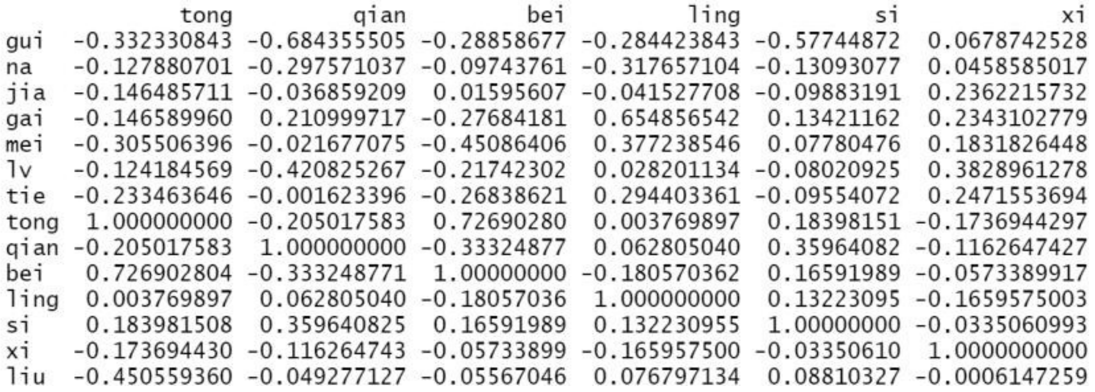


```
P
     gui    na     jia    gai    mei    lv     tie    tong   qian   bei    ling   si    
gui         0.0543 0.0000 0.0000 0.0087 0.0013 0.0016 0.0369 0.0823 0.1397 0.0000 0.0232
na   0.0543        0.0189 0.0103 0.3327 0.2171 0.9376 0.9875 0.1468 0.3945 0.6667 0.4688
jia  0.0000 0.0189        0.0000 0.0900 0.0752 0.1545 0.2205 0.2792 0.7174 0.0075 0.0806
gai  0.0000 0.0103 0.0000        0.2058 0.0516 0.0321 0.0513 0.1138 0.5974 0.0328 0.6258
mei  0.0087 0.3327 0.0900 0.2058        0.0043 0.0083 0.4577 0.5143 0.0817 0.0019 0.0009
lv   0.0013 0.2171 0.0752 0.0516 0.0043        0.0025 0.4640 0.1331 0.1746 0.0058 0.0152
tie  0.0016 0.9376 0.1545 0.0321 0.0083 0.0025        0.0168 0.5507 0.0620 0.0012 0.0104
tong 0.0369 0.9875 0.2205 0.0513 0.4577 0.4640 0.0168        0.4772 0.0243 0.2469 0.4139
qian 0.0823 0.1468 0.2792 0.1138 0.5143 0.1331 0.5507 0.4772        0.0048 0.3366 0.4603
bei  0.1397 0.3945 0.7174 0.5974 0.0817 0.1746 0.0620 0.0243 0.0048        0.1003 0.0155
ling 0.0000 0.6667 0.0075 0.0328 0.0019 0.0058 0.0012 0.2469 0.3366 0.1003        0.0036
si   0.0232 0.4688 0.0806 0.6258 0.0009 0.0152 0.0104 0.4139 0.4603 0.0155 0.0036       
xi   0.8534 0.6747 0.5612 0.3981 0.2860 0.2713 0.3860 0.1559 0.5983 0.6402 0.5739 0.3267
liu  0.1805 0.4395 0.1602 0.0664 0.0795 0.7528 0.3270 0.1838 0.3298 0.3884 0.4434 0.8636
```

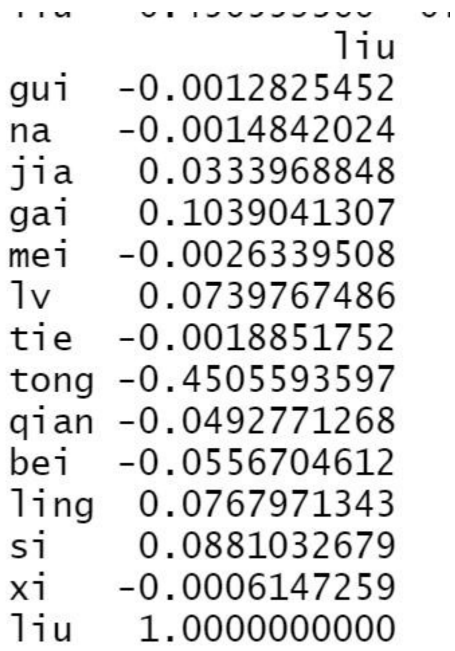
The corresponding correlation coefficient matrix for lead-barium glass is as follows:



```
              gui           na          jia        gai          mei          lv          tie
gui   1.000000000  0.370599473 -0.009527875 -0.3675676  0.051938837  0.28669857 -0.002077144
na    0.370599473  1.000000000 -0.113219545 -0.3428242 -0.003068085  0.06276005 -0.236375116
jia  -0.009527875 -0.113219545  1.000000000  0.2176789  0.176356507  0.07913558  0.281362583
gai  -0.367567587 -0.342824224  0.217678924  1.0000000  0.522844874  0.17043540  0.507836780
mei   0.051938837 -0.003068085  0.176356507  0.5228449  1.000000000  0.41274795  0.328484301
lv    0.286698574  0.062760053  0.079135581  0.1704354  0.412747954  1.00000000  0.180637330
tie  -0.002077144 -0.236375116  0.281362583  0.5078368  0.328484301  0.18063733  1.000000000
tong -0.332330843 -0.127880701 -0.146485711 -0.1465900 -0.305506396 -0.12418457 -0.233463646
qian -0.684355505 -0.297571037 -0.036859209  0.2109997 -0.021677075 -0.42082527 -0.001623396
bei  -0.288586770 -0.097437611  0.015956072 -0.2768418 -0.450864056 -0.21742302 -0.268386207
ling -0.284423843 -0.317657104 -0.041527708  0.6548565  0.377238546  0.02820113  0.294403361
si   -0.577448720 -0.130930768 -0.098831913  0.1342116  0.077804758 -0.08020925 -0.095540720
xi    0.067874253  0.045858502  0.236221573  0.2343103  0.183182645  0.38289613  0.247155369
liu  -0.001282545 -0.001484202  0.033396885  0.1039041 -0.002633951  0.07397675 -0.001885175
```

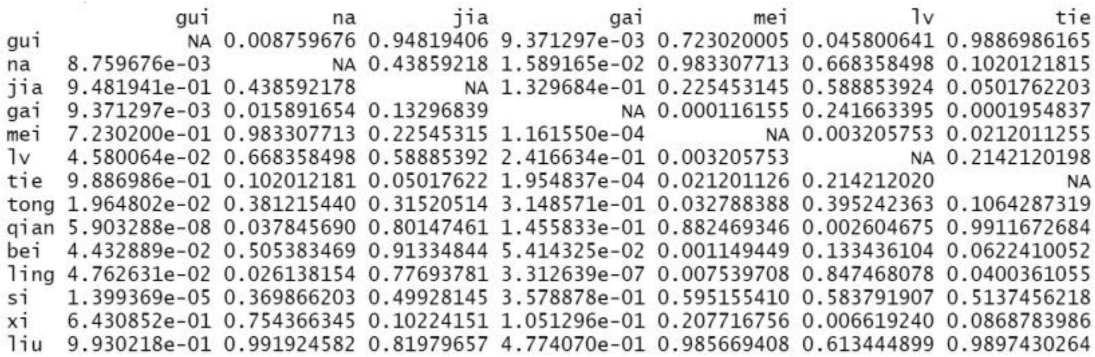


```
             tong         qian         bei         ling          si            xi
gui  -0.332330843 -0.684355505 -0.28858677 -0.284423843 -0.57744872  0.0678742528
na   -0.127880701 -0.297571037 -0.09743761 -0.317657104 -0.13093077  0.0458585017
jia  -0.146485711 -0.036859209  0.01595607 -0.041527708 -0.09883191  0.2362215732
gai  -0.146589960  0.210999717 -0.27684181  0.654856542  0.13421162  0.2343102779
mei  -0.305506396 -0.021677075 -0.45086406  0.377238546  0.07780476  0.1831826448
lv   -0.124184569 -0.420825267 -0.21742302  0.028201134 -0.08020925  0.3828961278
tie  -0.233463646 -0.001623396 -0.26838621  0.294403361 -0.09554072  0.2471553694
tong  1.000000000 -0.205017583  0.72690280  0.003769897  0.18398151 -0.1736944297
qian -0.205017583  1.000000000 -0.33324877  0.062805040  0.35964082 -0.1162647427
bei   0.726902804 -0.333248771  1.00000000 -0.180570362  0.16591989 -0.0573389917
ling  0.003769897  0.062805040 -0.18057036  1.000000000  0.13223095 -0.1659575003
si    0.183981508  0.359640825  0.16591989  0.132230955  1.00000000 -0.0335060993
xi   -0.173694430 -0.116264743 -0.05733899 -0.165957500 -0.03350610  1.0000000000
liu  -0.450559360 -0.049277127 -0.05567046  0.076797134  0.08810327 -0.0006147259
```



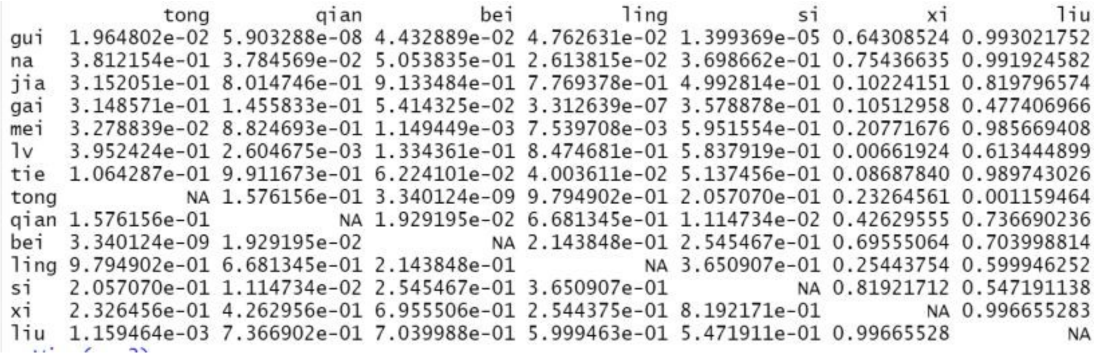
```
               liu
 gui  -0.0012825452
 na   -0.0014842024
 jia   0.0333968848
 gai   0.1039041307
 mei  -0.0026339508
 lv    0.0739767486
 tie  -0.0018851752
 tong -0.4505593597
 qian -0.0492771268
 bei  -0.0556704612
 ling  0.0767971343
 si    0.0881032679
 xi   -0.0006147259
 liu   1.0000000000
```

p-value



```
              gui          na        jia          gai         mei          lv          tie
gui            NA 0.008759676 0.94819406 9.371297e-03 0.723020005 0.045800641 0.9886986165
na   8.759676e-03          NA 0.43859218 1.589165e-02 0.983307713 0.668358498 0.1020121815
jia  9.481941e-01 0.438592178         NA 1.329684e-01 0.225453145 0.588853924 0.0501762203
gai  9.371297e-03 0.015891654 0.13296839           NA 0.000116155 0.241663395 0.0001954837
mei  7.230200e-01 0.983307713 0.22545315 1.161550e-04          NA 0.003205753 0.0212011255
lv   4.580064e-02 0.668358498 0.58885392 2.416634e-01 0.003205753          NA 0.2142120198
tie  9.886986e-01 0.102012181 0.05017622 1.954837e-04 0.021201126 0.214212020           NA
tong 1.964802e-02 0.381215440 0.31520514 3.148571e-01 0.032788388 0.395242363 0.1064287319
qian 5.903288e-08 0.037845690 0.80147461 1.455833e-01 0.882469346 0.002604675 0.9911672684
bei  4.432889e-02 0.505383469 0.91334844 5.414325e-02 0.001149449 0.133436104 0.0622410052
ling 4.762631e-02 0.026138154 0.77693781 3.312639e-07 0.007539708 0.847468078 0.0400361055
si   1.399369e-05 0.369866203 0.49928145 3.578878e-01 0.595155410 0.583791907 0.5137456218
xi   6.430852e-01 0.754366345 0.10224151 1.051296e-01 0.207716756 0.006619240 0.0868783986
liu  9.930218e-01 0.991924582 0.81979657 4.774070e-01 0.985669408 0.613444899 0.9897430264
```

fu



```
             tong         qian          bei         ling           si         xi         liu
gui  1.964802e-02 5.903288e-08 4.432889e-02 4.762631e-02 1.399369e-05 0.64308524 0.993021752
na   3.812154e-01 3.784569e-02 5.053835e-01 2.613815e-02 3.698662e-01 0.75436635 0.991924582
jia  3.152051e-01 8.014746e-01 9.133484e-01 7.769378e-01 4.992814e-01 0.10224151 0.819796574
gai  3.148571e-01 1.455833e-01 5.414325e-02 3.312639e-07 3.578878e-01 0.10512958 0.477406966
mei  3.278839e-02 8.824693e-01 1.149449e-03 7.539708e-03 5.951554e-01 0.20771676 0.985669408
lv   3.952424e-01 2.604675e-03 1.334361e-01 8.474681e-01 5.837919e-01 0.00661924 0.613444899
tie  1.064287e-01 9.911673e-01 6.224101e-02 4.003611e-02 5.137456e-01 0.08687840 0.989743026
tong           NA 1.576156e-01 3.340124e-09 9.794902e-01 2.057070e-01 0.23264561 0.001159464
qian 1.576156e-01           NA 1.929195e-02 6.681345e-01 1.114734e-02 0.42629555 0.736690236
bei  3.340124e-09 1.929195e-02           NA 2.143848e-01 2.545467e-01 0.69555064 0.703998814
ling 9.794902e-01 6.681345e-01 2.143848e-01           NA 3.650907e-01 0.25443754 0.599946252
si   2.057070e-01 1.114734e-02 2.545467e-01 3.650907e-01           NA 0.81921712 0.547191138
xi   2.326456e-01 4.262956e-01 6.955506e-01 2.544375e-01 8.192171e-01         NA 0.996655283
liu  1.159464e-03 7.366902e-01 7.039988e-01 5.999463e-01 5.471911e-01 0.99665528          NA
```

According to the sign and size of the correlation coefficient, the variables of lead and barium glass are mainly positively correlated, and their p-values are more

Jr.

V. Evaluation and analysis of the model

5.1 Advantages of the model

1. In the data preprocessing process in the first question, the treatment strategy for missing data is to additionally assign values to them on the basis of retaining the data in order to ensure the adequacy of data utilization as well as the effectiveness of fitting.
2. Assigning values to variables thus facilitates the use of mathematical methods to model and further enable the analysis of real-world problems.
3. For the linear regression model with multiple variables in Problem 1, the nonlinear problem is transformed into a linear problem by creating logistics shaped S-curves, in which the use of least squares greatly reduces the loss.
4. In Problem 2, a neural network classification algorithm is used to build a classification model, which greatly improves the accuracy of the model.
5. In problem 4, to avoid the interference of weathering on the data, so the data will be converted to use before weathering, the two types of glass, respectively, using the R language, to solve the correlation coefficients and P-values of the different chemical compositions.
6. In the principal component analysis method in question 2, chemical components with larger standard deviations were retained and those with smaller standard deviations were discarded to ensure correlation between the data and to avoid overlap in the information reflected in the data.
7. Problem three and problem one use a similar multiple linear regression model, which ensures the completeness of the model and sideways verifies the generalizability of the model.

5.2 Disadvantages of the model

1. In the first question the fitting process in R language, the independent variables are not strictly specified; the multiple regression process may suffer from analytical distortion due to the small amount of data given in the question.
2. During the treatment of Problem 2, the test for normal distribution was not performed because the amount of data was too small, and the data were directly assumed to be normally distributed in nature, which led to further subclassification using the mean.

VI. References

[1] WANG Jie,LI Mo,MA Qinglin,ZHANG Zhiguo,ZHANG Meifang,WANG Julin. Weathering study of an octagonal columnar lead-barium glass vessel from the Warring States period[J]. Glass and Enamel,2014,42(02):6-13.DOI:10.13588/j.cnki.g.e.1000-2871.2014.02.002.

[2] WANG Chengyu,TAO Ying,CHEN Min,HUANG Ming. Weathering of sodium-calcium-aluminum-magnesium silicate glass and alkali lead silicate glass[J]. Silicate Bulletin,1989(06):1-9 .DOI:10.16552/j.cnki.issn1001-1625.1989.06.001.

[3] WANG Chengyu,TAO Ying. Weathering of silicate glass[J]. Journal of Silicates,2003(01):78-85.

VII. Appendix

1 The code to solve for the relationship between surface weathering and glass type, color, and grain in Question 1 is as follows: library(xlsx)

```
read.table(file.choose())
fHandle=file.choose()
my_data=read.xlsx(fHandle,header=TRUE,sheetName="Sheet1")
head(my_data)
## Modeling, incorporating all independent variables
```

```
m1 <- glm(surface weathering ~. ,
family=binomial(link='logit'), data=my_data) ## view model
summary(m1)
anova(m1, test="Chisq") ## Slight change in significance of variables
```

2 One of the codes for solving the multiple linear regression model in Problem 1 is as follows, as are the other elements:

```
clear all
fujian3 = readmatrix("C:\Users\Administrator\Desktop\2022\fujian3.xlsx");
```

```
%% clear
temporary
variables clear
opts
k=0
for i=1:67
    if (fujian3(i,3)<999)&(fujian3(i,16)<9)&(fujian3(i,17)<9);
        k=k+1;
        y(k)=fujian3(i,3);
        x1(k)=fujian3(i,16);
        x2(k)=fujian3(i,17).
    end
end k,

X1=[ones(k,1),x1',x2'];

[b1,bint1,r1,rint1,s1]=regress(y',X1);
b1,bint1,s1
pause

n1=k;
r0=0.
for n=1:n1
    if rint1(n,1)*rint1(n,2)>0 r0=r0+1;
        rr(r0)=n.
    end
end
k=1;
for i=1:n1
    if i==rr(k)
        x1(i)=0; x2(i)=0; y(i)=0.
        k=k+1;
    end
if k>r0
break
end
end
nn1=0.
for i=1:n1
    if x1(i)>0;
        nn1=nn1+1;
        yn1(nn1)=y(i).
        xn1(nn1)=x1(i).
        xn2(nn1)=x2(i).
        end
    end
nn1
XX1=[ones(nn1,1),xn1',xn2'];

[bb1,bbint1,rr1,rrint1,ss1]=regress(yn1',XX1);
bb1,bbint1,ss1
```

3 Problem 2 neural network algorithm (python) # -*- coding: utf-8 -*-

```
"""
Created on Sun Sep 18 19:12:36 2022

@author: Administrator
"""



#import libraries
import os
import pandas as pd
import numpy as np
#Set working directory and load data


def initialize_parameters(n_x, n_h, n_y).

    np.random.seed(2) # we set up a seed so that our output matches ours although the
initialization is random.

    W1 = np.random.randn(n_h, n_x) * 0.01 #weight matrix of shape (n_h, n_x) b1
    = np.zeros(shape=(n_h,))          #biasvector of shape (n_h, 1)
    W2 = np.random.randn(n_y, n_h) * 0.     01#weight matrix of shape (n_y, n_h)
    b2 = np.zeros(shape=(n_y,))       #biasvector of shape (n_y, 1)

    #store parameters into a dictionary
    parameters = {"W1": W1,
                  "b1": b1.
                  "W2": W2.
                  "b2": b2}

    return parameters
def forward_propagation(X, parameters):
#retrieve intialized parameters from dictionary
    W1 = parameters['W1']
    b1 = parameters['b1']
    W2  =  parameters['W2']
    b2 = parameters['b2']


    # Implement Forward Propagation to calculate A2 (probability) Z1
    = np.dot(W1, X) + b1
    A1 =np.tanh(Z1)     #tanhactivation function
    Z2 = np.dot(W2, A1) + b2
    A2 = 1/(1+np.exp(-Z2))      #sigmoid activation function

     cache = {"Z1": Z1,
              "A1": A1.
              "Z2": Z2.
              "A2": A2}

    return A2, cache
def compute_cost(A2, Y, parameters).

    m = Y.shape[1] # number of training examples #

    Retrieve W1 and W2 from parameters
    W1 = parameters['W1']
    W2 = parameters['W2']

    # Compute the cross-entropy cost
    logprobs = np.multiply(np.log(A2), Y) + np.multiply((1 - Y), np.log(1 - A2)) cost
    = - n p .sum(logprobs) / m

    return cost
def backward_propagation(parameters, cache, X, Y): #
Number of training examples
    m = X.shape[1]

    # First, retrieve W1 and W2 from the dictionary "parameters". W1
    = parameters['W1']
    W2 = parameters['W2']
    ### END CODE HERE ###

    # Retrieve A1 and A2 from dictionary "cache". A1
    = cache['A1']
    A2 = cache['A2']

    # Backward propagation: calculate dW1, db1, dW2, db2. dZ2=
    A2 - Y
    dW2 = (1 / m) * np.dot(dZ2, A1.)
    db2 = (1 / m) * np.sum(dZ2, axis=1, keepdims=True)
    dZ1 = np.multiply(np.dot(W2.T, dZ2), 1 - np.power(A1, 2))
    dW1 = (1 / m) * np.dot(dZ1, X.T)
    db1 = (1 / m) * np.sum(dZ1, axis=1, keepdims=True) grads
    = {"dW1": dW1,
             "db1": db1,
             "dW2": dW2.
             "db2": db2}

    return grads
def update_parameters(parameters, grads, learning_rate=1.2):
# Retrieve each parameter from the dictionary "parameters"
    W1 = parameters['W1']
    b1  =  parameters['b1']
    W2 = parameters['W2']
    b2 = parameters['b2']

    # Retrieve each gradient from the dictionary "grads"
    dW1 = grads['dW1']
    db1 = grads['db1']
    dW2 = grads['dW2']
    db2 = grads['db2']

    # Update rule for each parameter
    W1 = W1 - learning_rate * dW1
    b1 = b1 - learning_rate * db1
    W2  =  W2  -  learning_rate  *
    dW2 b2 = b2 - learning_rate *
    db2

    parameters = {"W1": W1,
                  "b1": b1.
                  "W2": W2.
                  "b2": b2}

    return parameters

def layer_sizes(X, Y).
   n_x = X.shape[0]
   n_h = 4
   n_y = Y.shape[0]
   return (n_x, n_h, n_y)

def nn_model(X, Y, n_h, num_iterations=10000, print_cost=False).
```

```
    np.random.seed(3)
    n_x = layer_sizes(X, Y)[0]
    n_y = layer_sizes(X, Y)[2]

    # Initialize parameters, then retrieve W1, b1, W2, b2. Inputs: "n_x, n_h,
n_y". Outputs = "W1, b1, W2, b2, parameters".
    parameters = initialize_parameters(n_x, n_h, n_y)
    W1 = parameters['W1']
    b1  =  parameters['b1']
    W2  =  parameters['W2']
    b2 = parameters['b2']

    # Loop (gradient descent)
    for i in range(0, num_iterations):

        # Forward propagation. Inputs: "X, parameters". Outputs: "A2, cache". A2,
        cache = forward_propagation(X, parameters)

        # Cost function. Inputs: "A2, Y, parameters". Outputs: "cost".
        cost = compute_cost(A2, Y, parameters)

        # Backpropagation. Inputs: "parameters, cache, X, Y". Outputs: "grads".
        grads = backward_propagation(parameters, cache, X, Y)

        # Gradient descent parameter update. Inputs: "parameters, grads".
Outputs: "parameters".
        parameters = update_parameters(parameters, grads)

        ### END CODE HERE ###

        # Print the cost every 1000 iterations if
        print_cost and i % 1000 == 0.
            print ("Cost after iteration %i: %f" % (i, cost)) return
    parameters,n_h

if name____ == " main ": #
    read data
    glasses = pd.read_csv('glasses.csv')

#Create Input and Output columns
    X                                                                  =
glasses[['gui','na','jia','gai','mei','lv','tie','tong','qian','bei','ling','si
' , ' xi','liu',]].values.T
    Y = glasses[['kind']].values.
    Y = Y.astype('uint8')
    parameters = nn_model(X,Y , n_h = 6, num_iterations=10000, print_cost=True)
```

4 Problem 2 Principal Component
Analysis Algorithm (High
Potassium Glass) clc

```
clear all
[CJ,textdata]=xlsread('fujian5.xlsx');
X=CJ(:,1:end);
M=mean(X);
Co=cov(X);        %Calculate
covariance matrix r=corrcoef(X);
                  %Calculate the
correlation coefficient matrix
[COEFF,SCORE,latent,tsquare]=pca(X)      % Principal
Component Analysis percent_explained =
100*latent/sum(latent) figure(2).
pareto(percent_explained)   %plot 2
xlabel('Principal Components')
ylabel('Variance explained ( )')
result(1,:)={'eigenvalue','contribution','cumulative
contribution'}; result(2:15,1)=num2cell(latent)
result(2:15,2:3)=num2cell([percent_explained,cumsum(percent_explained)]); %
Output table 2
stnum=textdata(2:end,1) %extract
the number
sumX=sum(X,2) %calculate the total
score result1=cell(19,4)
result1(1,:)={'ordinal number','total content','first principal
component score y1','second principal component score y2'}
result1(2:end,1)=stnum
result1(2:end,2:end)=num2cell([sumX,SCORE(:,1:2)]) %output table 3
```

5 Algorithm for Solving Problem 4 Correlation Coefficient
Matrix and p-Value Matrix (R Language) (High Potassium Glass)

```
library(xlsx)

read.table(file.choose())
fHandle=file.choose()
my_data=read.xlsx(fHandle,header=TRUE,sheetName="Sheet1")

head(my_data)

library("Hmisc")
res2 <- rcorr(as.matrix(my_data))
res2
# Extract the correlation coefficients
res2$r
# Extract p-values
res2$P
```

VIII. Reference to web resources

Pure Python Neural Network Modeling of the Iris dataset -AliCloud Developer Community (aliyun.com)

Correlation Analysis | R -- Correlation Matrix and Visualization - 简书 (jianshu.com)