2021年美国数学建模竞赛电子科技大学模拟赛
承诺书

我们仔细阅读了数学建模竞赛的竞赛规则.

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与队外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛规则的，如果引用别人的成果或其他公开的资料(包括网上查到的资料)，必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺，严格遵守竞赛规则，以保证竞赛的公正、公平性。如有违反竞赛规则的行为，我们将受到严肃处理。

我们的题目编号是（填写：A 或者 B）： A

我们报名**队伍**号是： H166

参赛队员姓名学号（**中文填写打印并签名**）：

1. 廖桢琰 2019011206022
2. 崔冬辰 2019091601026
3. 胡江波 2019011206010

导教师或指导教师组负责人（有的话填写）：

是否愿意参加2021年美国赛（是，否）： 是

日期：2021年11月22日

Finding the best summer job based on AHP

Summary

The main task of Question A is to analyze the factors that undergraduates will consider when looking for a summer job, and build a mathematical model based on their considerations to help them choose the most suitable job. Then, showcase our solution.

Based on the actual situation, our group determined the dimensions of college students' preference for summer vacation jobs. In order to solve the problem, our team set all optional summer jobs first. Second, we quantitatively describe college students' preference for summer vacation work and create a model to find the most suitable job. Third, we showcase our work. Last but not least, we analyze the shortcomings of the model and propose the improvement direction.

To set all jobs, we abstracted a map of the distribution of summer jobs. We divide the jobs into online and offline. Based on the actual situation, we collected relevant data of job opportunities in different developed-level area and put them into a region according to certain rules.

To quantitatively describe user's preference properly, we use analytic hierarchy process (AHP). And we check the consistency of inputs given by user.

We create a modified AHP model to score each job for the user's preference dimension, because the original AHP model is not so suitable for us to do analysis. With these scores, we combined the scores and the quantitative description of preferences to output a most suitable job for user.

Last but not least, we showcase our work in form of website, so as to reflect the practical significance of the work.

Keywords: Summer job　Analytic hierarchy process　Judgement matrix

Contents

1 Introduction

1.1 Background

For the summer vacation, every undergraduate has a unique plan. Although there is still some time before the summer vacation, we have already begun to be prepared for the summer job. Opening the job site, the various positions dazzled us. Among so many choices, we hope to find the "best" choice for us.

Some of these jobs enable us to carry out remote working, and some are far away from our home. Some jobs do not require our ability, such as some physical activities. Some jobs require us to have certain abilities, such as a salesperson. Some jobs require us to have certain specialized knowledge or skills, such as a tutor.

It is certainly delightful to make some money, but some of us may not like to take up too much time, and some may also want to do some challenging work, Of course, there are also some who are eager to make as much money as possible.

1.2 Restatement of the problem

We are under a variety of jobs to choose from and our personal preferences for work vary. We believe that the “best” job equals to the most suitable job.

Based on the above background, we need to establish a mathematical model to solve the following problems:

1) Analyze the factors we should consider when looking for a summer job.

2) How to find the most satisfying job based on everyone's preference for work?

3) Showcase our work.

2 Overall Analysis

To solve the above problems, first of all, we need to consider two major aspects. One is how to represent the work presented to undergraduates, and the other is how to represent and make use of their preferences. Furthermore, we need to develop a mathematical model that can choose the most satisfying job based on each person's preferences.

Our work will be divided into the following steps:

1) As answer to the question we look up relative information on Internet and sum up the factors undergraduates will consider mostly when looking for a Summer job.

For a summer job, we usually care about the location, the wage (dollars per hour), the number of hours per week we are required to work, and the difficulty index of the job.

2) Determine the attributes of the job.
   Usually on the recruitment website, we can see the hourly salary, working hours and job threshold of each job.
   Considering that college students may want to work remotely. For offline work, commuting distance will be considered. It is also essential to consider that the actual job opportunities in each region are different. Therefore, we will introduce a "map" to show the distribution of all jobs. This makes each position in addition to the attributes of hourly wage, working hours and work difficulty index, but also a coordinate.

3) Find the best job according to the preferences entered by college students.
   In order to fully start from the preferences of each undergraduate, we will use the analytic hierarchy process to find a quantitative description of preferences. Based on this description, we will find the most suitable job for the user.

3 Assumptions

To simplify and clarify our problems, we make the following basic assumptions.

1) Assuming that the home of a college student looking for a summer job is at any point in a definite area.
2) Assuming that every college student is competent for the job he chooses.
3) Assuming that all job opportunities are public. Every college student can browse all jobs.
4) Assuming that each job is adjusted on a weekly basis.
5) Assuming that different jobs are divided into five categories according to difficulty index.
6) Assuming that different regions can provide the same types of jobs, but their proportions are different.
7) Assuming that the jobs available in each region are randomly distributed.

3 Notation

Abbreviation	Description	Unit
(x_q,y_q)	Location of a job.	
(x_p,y_p)	Location of an undergraduate.	
d	Distance between (x_q,y_q) and (x_p,y_p)	km
D	Maximum acceptable commuting distance.	
w	Wage of a job.	dollars/h
t	Required working hours.	h/week
q	Difficulty index of a job.	
Q	Difficulty index of the job the student wants to do.	
M,N	Attributes of jobs may be compared with each other.	
I_{MN}	Importance scale of M compare to N.	
CI	Consistency index.	
RI	Random consistency index.	
CR	Consistency ratio.	
A	Judgement matrix of user's preference.	
B	Judgement matrix of attributes of jobs.	
ω	User preference weight	
φ	Score of jobs.	

4 Analysis and Modeling

4.1 Jobs Distribution Model

4.1.1 Jobs attribute setting

Based on the factors considered by college students when choosing a job (distance, wage, working hours, difficulty index of work), we also set the job as five attributes: type of job, location coordinate, commuting distance, hourly salary, weekly work duration.

The specific settings are as follows:

1) Job Category:

We set the job type as a constant q between 1 and 5. That is, according to the difficulty index of the job and the required skill level, it is divided into:

Low difficulty index work $q = 1$. Such as courier delivery person, construction site builder.

Medium to low difficulty index jobs $q = 2$. For example, restaurant dishwashing worker, handing out flyers.

Medium difficulty index jobs $q = 3$. Such as salesmen, receptionists.

Medium to high difficulty index jobs $q = 4$. Such as ordinary employees, interns in small companies.

Difficult jobs $q = 5$. Such as tutors, teaching assistants, interns in large enterprises.

2) Work location

For offline work, we set the work location to a point in the plane coordinate system in the area

For remote work, we believe that the work location is the home of the job seeker, and there is no need to set the coordinates of the work location.

3) Commuting distance

We set the home of all undergraduates to any point in an area. So, the commuting distance is set at the straight-line distance between home coordinates and work location coordinates.

If the job is remote, then we consider the commuting distance to be zero.

4) Wage

For each job, we consider that wage is a constant amount. Its unit is dollars per hour.

In different regions, the hourly wages provided by the same type of work may be different, which will be reflected in the subsequent Jobs generation and distribution steps.

5) Working hours

For each job, we consider that working hour is a constant amount. Its unit is hours per week.

4.1.2 Jobs generation and distribution

We first built a model, more specifically, a map, showing where all undergraduate job-seekers live and where they work offline.

We set this map as a country, including the developed area and less developed area. This map will affect the distribution of offline jobs.

In order to reasonably simplify the model, we refer to the local economic development of the United States. As showed in the figure below, we divide the central half into less developed area, the eastern $1/3$ and the western $1/6$ into developed area. The north-south span is $1000km$ and the east-west span is $1000km$. Even for demanding users, we can find good results.

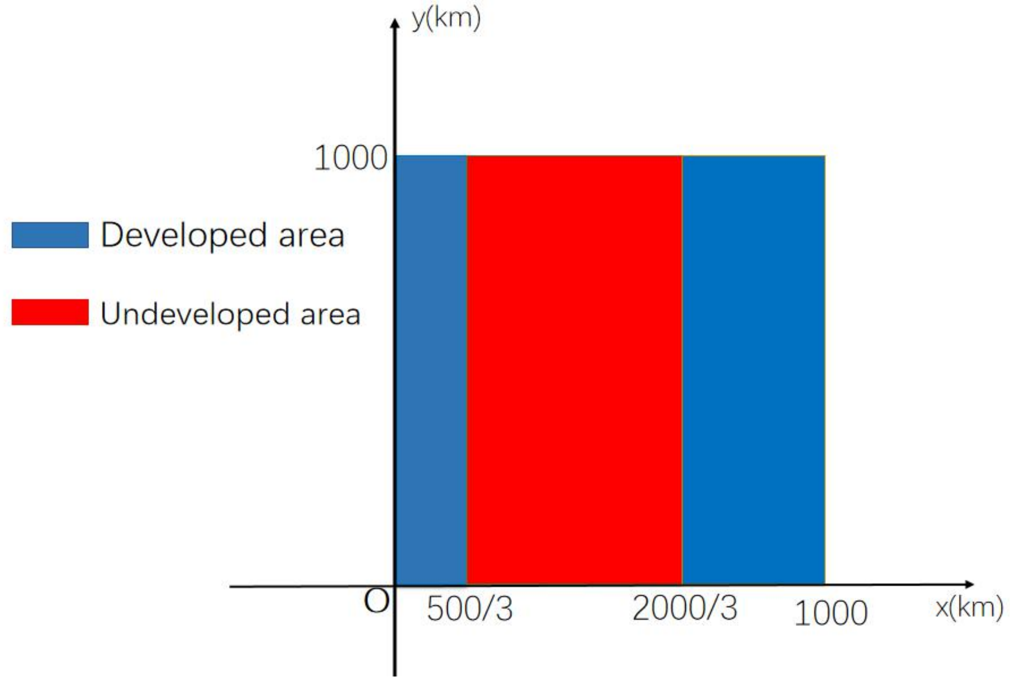


Figure 1 Division of developed and less developed area.

In order to generate and put different kinds of jobs into the map, we inquired about famous job search websites and sorted out the information about summer jobs on the website.

Requirement	Wage	Hours per week	Proportion
Low $q = 1$	\$3-\$5 /h	20-40 h/w	0.0400
Mid-Low $q = 2$	\$3-\$10 /h	20-30 h/w	0.1000
Medium $q = 3$	\$10-\$15 /h	20-50 h/w	0.2600
Mid-High $q = 4$	\$20-\$30 /h	30-50 h/w	0.3600
High $q = 5$	\$30-\$50 /h	40-50 h/w	0.2400

Table 1 Distribution of jobs in developed area [1]

Requirement	Wage	Hours per week	Proportion
Low $q = 1$	\$1-\$3 /h	20-40 h/w	0.1535
Mid-Low $q = 2$	\$3-\$8 /h	30-40 h/w	0.4386
Medium $q = 3$	\$8-\$10 /h	30-50 h/w	0.3289
Mid-High $q = 4$	\$10-\$20 /h	30-50 h/w	0.0768
High $q = 5$	\$20-\$25 /h	40-50 h/w	0.0022

Table 2 Distribution of jobs in less developed area [2]

Requirement	Wage	Hours/week	Proportion
Medium q=3	\$15-\$20/h	10-50h/w	0.34
High q=5	\$25-\$50/h	20-40h/w	0.66

Table 3 Distribution of remote jobs.[3]

For remote job, commuting distance is no need to consider. So, we randomly generated 1,000 remote job.

We suppose that the developed area provides as many offline jobs as the less developed area. There are 20,000 offline jobs nationwide for our undergraduates to select from.

We will randomly generate 10,000 different jobs in the developed area and 10,000 jobs in the less developed area in proportion. Then randomly put them into the map.

Among them, for each range value in Table 1 and Table 2, we use the random number generation function to randomly select the value. For example, for a High requirement job (that is $q = 5$) in Table 1, we may generate a $w = \$40/h$, $t = 45h/week$ job.

Then, the two groups of 50,000 positions were evenly allocated to the developed area and the less developed area in proportion.

In this way, we get all the summer jobs available for undergraduate students.

4.2 Preference describing using AHP

4.2.1 Judgement matrix constructing

We believe that each student's preference refers to how important the student judges the attributes of a job to him or her.

In order to obtain more accurate analysis results, we will first ask the user to determine the maximum of commuting distance, and then input the difficulty index of the work q.

We chose a constant between 1 and 9 as the scale. If work attribute M is compared with the work attribute N, then enter numbers as following rules:

Scale	Description
1	M and N are of equal importance.
3	M is a bit more important than N.
5	M is more important than N.
7	M is mightily more important than N.
9	M is extremely more important than N.
2,4,6,8	The median of the two adjacent judgments above
$\{\frac{1}{k} \| k \in 1,2,3\ldots,9\}$	Measurement of N compare to M.

Table 4 Rules of scale used in judgement matrix

For example, if I_{wt} represents Important scale of wage compare to working hours, then I_{tw} represents important scale of working hours compare to wage and $I_{tw} = 1/I_{wt}$. Then, for remote work and offline work, we can set a judgment matrix A for undergraduates' work attributes.

A for remote work is as follows:

$$\begin{bmatrix} 1 & I_{wt} \\ I_{tw} & 1 \end{bmatrix} \#(1)$$

A for offline work is as follows:

$$\begin{bmatrix} 1 & I_{wt} & I_{wd} \\ I_{tw} & 1 & I_{td} \\ I_{dt} & I_{dt} & 1 \end{bmatrix} \#(2)$$

The judgment matrix A is constructed entirely based on the preferences given by the user.

4.2.2 Quantitative description for preference

For remote work, only hourly pay and working hours are compared. Users' evaluation of these two attributes is accurate. Therefore, we think judgment matrix for remote work is rational.

However, for offline work, this judgment matrix is not rigorous enough. Because the user's evaluation of the three attributes may be unclear. For example, the user may think that hourly wages are more critical than working hours, and working hours are more important than distance. distance is more important than time, which constitutes a self-contradictory situation. Therefore, we need to check the consistency of the judgment matrix.

Solve the characteristic root of the judgment matrix $A(2)$, find its maximum characteristic roots λ_{max}, the corresponding characteristic vector, then the consistency index (n is the rank of the matrix):

$$CI = \frac{\lambda_{max} - n}{n - 1} = \frac{\lambda_{max} - 3}{2} \#(3)$$

By looking up the table, when the matrix rank is three, $RI = 0.58$

$$CR = \frac{CI}{RI} = \frac{\lambda_{max} - 3}{1.16} \#(4)$$

It is generally believed that if the consistency ratio $CR < 0.1$, the inconsistency of the judgment matrix A is considered to be within the allowable range, with a satisfactory consistency. If the calculated $CR > 0.1$, the inconsistency is considered out of allowable range. We will ask users to rethink their preferences.

In order to further clarify the user's preference for each attribute, we also need to calculate the user's preference weight ω for each attribute.

$$\omega_i = \frac{1}{n}\sum_{j=1}^{n} \frac{A_{ij}}{\sum_{k=1}^{n} A_{kj}} \#(5)$$

If $D = 0$, then $n = 2$, $i = 1,2$

If $D > 0$, then $n = 3$, $i = 1,2,3$

From the above analysis, we have obtained a quantitative description of user preference.

4.3 Job choosing

4.3.1 Judgement matrix constructing

We first evaluate the 10 jobs and look for the best.

When $D = 0$, that is, the user only wants to do remote work, the judgment matrix B_w, B_t is constructed.

When $D > 0$, that is, when the user wants to do offline work, construct judgment matrix B_w, B_t and B_d.

Take $D > 0$ as an example to illustrate. After the user enters Q, it begins to analyze all the work that meets the requirements. As for the attributes $w, t\ and\ d$ of work, they are not fixed as constants from one to nine, so they cannot be used as the scale of judgment. We nondimensionalize these tasks and reduce them to scales that form judgment matrices.

Select n jobs wage $\{w_k|k = 1,2,3...n\}$ forms the following matrix.

$$B_w = \begin{bmatrix} 1 & w_{1,2} & \cdots & w_{1,n} \\ w_{2,1} & 1 & \cdots & w_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ w_{n,1} & w_{n,2} & \cdots & 1 \end{bmatrix} \#(6)$$

If $w_i > w_j (i,j = 1,2,...n \text{ and } i \neq j)$

$$w_{i,j} = 1 + [\frac{9|w_i - w_j|}{w_{k_{max}} - w_{k_{min}}}]$$

$$w_{j,i} = \frac{1}{1 + [\frac{9|w_i - w_j|}{w_{k_{max}} - w_{k_{min}}}]}$$

Select the working hours of n jobs $\{t_k|k = 1,2,...n\}$ constitutes the following matrix:

$$B_t = \begin{bmatrix} 1 & t_{1,2} & \cdots & t_{1,n} \\ t_{2,1} & 1 & \dots & t_{2,n} \\ \vdots & \vdots & \dots & \vdots \\ t_{n,1} & t_{n,2} & \dots & 1 \end{bmatrix} \#(7)$$

If $t_i < t_j$,$(i,j = 1,2,...n$ and $i \neq j)$

$$t_{i,j} = \frac{1}{1 + [\frac{9|t_i - t_j|}{t_{k_{max}} - t_{k_{min}}}]}$$

$$t_{j,i} = 1 + [\frac{9|t_i - t_j|}{t_{k_{max}} - t_{k_{min}}}]$$

Note that the constructors of B_t and B_d are different from those of B_w.

Referring to the consistency checking method in 4.2.2, we use the same method for consistency detection.

RI value query is shown in the following table when consistency check is carried out:

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49

Table 5 Random consistency index

4.3.2 Best job choosing base on user's preference

After constructing three judgment matrices, the score of each job from these three dimensions can be obtained.

In terms of wage, each job is scored as:

$$\varphi_{w_i} = \frac{1}{n}\sum_{j=1}^{n}\frac{B_{w_{ij}}}{\sum_{k=1}^{n} B_{w_{kj}}},\ (i = 1,2,3 \ldots n)$$

Similarly, in terms of working hours and commuting distance, each job are scored as $\varphi_{t_i}, \varphi_{d_i}$. To find the job that meets the user's preference best, we calculate the final score for each of the n jobs for the user.

$$\varphi_i = \varphi_{w_i}\omega_1 + \varphi_{t_i}\omega_2 + \varphi_{d_i}\omega_3,\ (i = 1,2,3, \ldots n)$$

The job with the highest final score is the one that best meets the user's requirements among the n jobs.

In the application of AHP, n is usually no greater than ten. So there are far more jobs to choose from than we can analyze at one time. In order to analyze the total number of jobs available, we reserve the job with the highest final score each time and use it as an input parameter for the next analysis process, along with the rests of the jobs below that have not yet been analyzed. In this way, we can get the choice that best meets the user's requirements among all the jobs.

5 Model verification

According to the question, we will test with ten virtual characters whose Settings have been deliberately considered.

Fictional character 1：Bob

Bob is a freshman living in the Midland. He wants to experience social life and earn some money during this summer, so he uses our model to look for a job . He decides to do an offline job, and because of a lack of skill, he hopes the job to be less difficult. Moreover, he doesn’t care about going to a far place to work at all.

```
>> A=[1,3,9;1/3,1,3;1/9,1/3,1];   %Preference of this college student
Q=[1,2];                          %Difficulty
Distance=50;                      %Maximum acceptable distance(km)
X=500;Y=500;                      %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);        %The most suitable job for this college student
A most suitable job:
    position    :(476,489)
      wage      :    8
     hours      :    31
difficulty index:    2
    distance    :    27
```

Figure 2 Fictional character1

Analysis: Since he doesn’t care about distance, the model calculates a job that is a bit far from him. Compared to jobs of the same difficulty index, this job offers a higher hourly wage and appropriate working hours. It can be considered that Bob's searching result fits his own situation.

Fictional character 2:Tim

Tim is a senior in college, living in the western area. The graduation is coming, he will take this summer to do an internship in his related field. He is a capable college student who wants to do a difficult job, and at the same time, he also wants to make some money for the future, he doesn't mind traveling long to do the job.

```
>> A=[1,3,8;1/3,1,3;1/8,1/3,1];   %Preference of this college student
Q=[4,5];                          %Difficulty
Distance=40;                      %Maximum acceptable distance(km)
X=10;Y=100;                       %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);        %The most suitable job for this college student
A most suitable job:
    position    :(14,113)
      wage      :    49
     hours      :    42
difficulty index:    5
    distance    :    14
```

Figure 3 Fictional character 2

Analysis: Although he doesn't mind traveling long , the model finds a job close to him——This meets our assumption. Compared to job of the same difficulty index , it pays very well and has appropriate working hours. It can be considered that Tim's searching result fits his own situation.

Fictional character 3: Peter

Peter is a junior , living in the Midland. He is a capable student who wants to try a challenging online job this year and has a lot of time to do this job.

```
>> A=[1,1/4;4,1];                 %Preference of this college student
Q=[5];                            %Difficulty
Distance=0;                       %Maximum acceptable distance(km)
X=200;Y=700;                      %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);        %The most suitable job for this college student
A most suitable job:
    position    :(344,956)
      wage      :    48
     hours      :    20
difficulty index:    5
    distance    :    294
```

Figure 4 Fictional character 3

Analysis: Although this job is far away from him, this factor can be ignored because he works online. The hourly wage is quite great and although the working hours are more than jobs of the same difficulty index, it meets Peter's requirement. Peter's search results can be considered to fit his own situation.

Fictional character 4: Pat

Pat is a freshman and lives in the eastern area. She is interested in online jobs and would like to try a job which is not too difficulty , and she has enough time to do it.

```
>> A=[1,4;1/4,1];                     %Preference of this college student
Q=[3];                                %Difficulty
Distance=0;                           %Maximum acceptable distance(km)
X=420;Y=310;                          %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);            %The most suitable job for this college student
A most suitable job:
    position     :(353,890)
      wage       :    20
     hours       :    22
difficulty index:    3
    distance     :    584
```

Figure 5 Fictional character 4

Analysis: Although this job is far away from her, the factor can be ignored because she works online. The hourly wage of this job is quite great and the working hours are appropriate compared to jobs of the same difficulty index. Pat's search results can be considered to fit her own situation.

Fictional character 5

White is a sophomore,living in the Central Region.He is looking for a job with moderate difficulty.He hopes it is close to his home and does not require a lot of time.

```
>> A=[1,1/2,1;2,1,2;1,1/2,1];         %Preference of this college student
Q=[3,4];                              %Difficulty
Distance=5;                           %Maximum acceptable distance(km)
X=900;Y=10;                           %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);            %The most suitable job for this college student
A most suitable job:
    position     :(0,10)
      wage       :    22
     hours       :    42
difficulty index:    4
    distance     :    3
```

Figure 6 Fictional character 5

Analysis: The job pays a very high hourly wage compared to jobs of the same difficulty index, has appropriate working hours, and generally meets his preferences. Peter's searching result can be considered to fit his own situation.

Fictional character 6.

Alice is a sophomore living in the Central Region who is looking for offline jobs that are moderately difficult and where the hourly wage, hours, and distance are balanced.

```
>> A=[1,1,1;1,1,1;1,1,1];             %Preference of this college student
Q=[2,3];                              %Difficulty
Distance=30;                           %Maximum acceptable distance(km)
X=650;Y=700;                          %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);            %The most suitable job for this college student
A most suitable job:
    position     :(671,706)
      wage       :    13
     hours       :    22
difficulty index:    3
    distance     :    22
```

Figure 7 Fictional character 6

Analysis: This job has a appropriate hourly wage and appropriate working hours compared to jobs of the same difficulty. Alice's search results can be considered to fit her situation.

Fictional character 7.

Jim is a junior in college, lives out west, is in desperate need of money, and doesn't place much value on the distance of his job from home. He's looking for a less difficult offline job.

```
>> A=[1,3,8;1/3,1,2;1/8,1/2,1];  %Preference of this college student
Q=[1,2,3];                       %Difficulty
Distance=35;                      %Maximum acceptable distance(km)
X=60;Y=800;                      %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);       %The most suitable job for this college student
A most suitable job:
    position     :(73,814)
      wage       :    15
     hours       :    37
difficulty index:    3
    distance     :    20
```

Figure 8 Fictional character 7

Analysis: Compared to jobs of the same difficulty level, this job pays very good hourly wages, has more hours, and can solve his problem of lack of money. Jim's search results can be considered to fit his own situation. He is advised to drive there.

Fictional character 8: John

John is a freshman living in the eastern part of the country who wants to work an offline job that is not too difficult and wants to make money, but also wants to have enough leisure time that he is not particularly concerned about distance.

```
>> A=[1,7,3;1/7,1,1/2;1/3,2,9];  %Preference of this college student
Q=[1,2];                         %Difficulty
Distance=42;                      %Maximum acceptable distance(km)
X=880;Y=880;                     %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);       %The most suitable job for this college student
错误使用 AHP (line 16)
未通过一致性检验，程序退出

出错 Findjob (line 95)
            [W, Lmax, CI, CR] = AHP(A);              %用户输入的A  时薪、时长、距离的权重
```

Figure 9-1 Fictional character 8(1)

The model shows that it does not pass the consistency test, and John is advised to re-enter the preference matrix.

```
>> A=[1,1/2,1/4;2,1,1;4,1,1];    %Preference of this college student
Q=[4,5];                         %Difficulty
Distance=3;                      %Maximum acceptable distance(km)
X=880;Y=880;                     %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);       %The most suitable job for this college studen
错误使用 Findjob (line 57)
There is no work of this type in this scope\n
```

Figure 9-2 Fictional character 8(2)

The model shows that There is no work of this type in this scope, and John is advised to re-enter the Distance.

```
>> A=[1,7,3;1/7,1,1/2;1/3,2,1];   %Preference of this college student
Q=[1,2];                           %Difficulty
Distance=42;                        %Maximum acceptable distance(km)
X=880;Y=880;                       %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);         %The most suitable job for this college student
A most suitable job:
    position      : (900,888)
       wage       :    10
      hours       :    30
difficulty index:    2
    distance      :    22
```

Figure 9-3 Fictional character 8(3)

He searches for the optimal job coordinates (900, 888), which pays $10 an hour, works 30 hours a week, has a difficulty level of 2, and is about 22 km away from him.

Analysis: Compared to jobs of the same difficulty, this job pays very good hourly wages and has many more hours. Jim's search results can be considered to fit his situation. It is recommended that he drive or take the train to get there.

Fictional character 9: Kaki

Kaki is a senior living in the Midlands who is looking for a more difficult offline job. She wants to work close to home and have enough time for breaks.

```
>> A=[1,1/2,3;2,1,1;7,1,1];     %Preference of this college student
Q=[4,5];                           %Difficulty
Distance=5;                        %Maximum acceptable distance(km)
X=880;Y=880;                       %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);         %The most suitable job for this college student
错误使用 AHP (line 16)
未通过一致性检验，程序退出

出错 Findjob (line 145)
            [W, Lmax, CI, CR] = AHP(A);                %用户输入的A
```

Figure 10-1 Fictional character 9(1)

The model shows that it fails the consistency test and suggests that Kaki re-enter the preference matrix.

```
>> A=[1,1/2,1/4;2,1,1;4,1,1];   %Preference of this college student
Q=[4,5];                           %Difficulty
Distance=5;                        %Maximum acceptable distance(km)
X=880;Y=880;                       %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);         %The most suitable job for this college student
A most suitable job:
    position      : (881,882)
       wage       :    48
      hours       :    47
difficulty index:    5
    distance      :    3
```

Figure 10-2 Fictional character 9(2)

Analysis: Compared to jobs of the same difficulty, this job pays very good hourly wages, has fewer hours, and is very close. Kaki's search results can be considered appropriate for her situation. She is advised to walk to it.

Fictional character 10.

Andy is a junior living in the West and is looking for offline jobs that are moderately difficult and where the hourly wage, hours, and distance are balanced.

```
>> A=[1,1,1;1,1,1;1,1,1];          %Preference of this college student
Q=[2,3,4];                          %Difficulty
Distance=35;                         %Maximum acceptable distance(km)
X=150;Y=400;                        %Posiition of this college student(km)
Findjob(A,Q,X,Y,Distance);          %The most suitable job for this college student
A most suitable job:
    position     :(149,393)
      wage       :     29
     hours       :     42
difficulty index:     4
    distance     :     8
```

Figure 11 Fictional character 10

Analysis: Compared to jobs of the same difficulty, this job has a moderate hourly wage, moderate hours, and moderate distance, which basically meets her preferences. Andy's search results can be considered appropriate for her situation. She is advised to drive there.

6 Webpage-based presentation

To show the practicality of our work, we built up a simple website. We will demonstrate the usage.

We get users' location (x_p,y_p) beforehand.

Step 1: We ask for the maximum acceptable commuting distance.

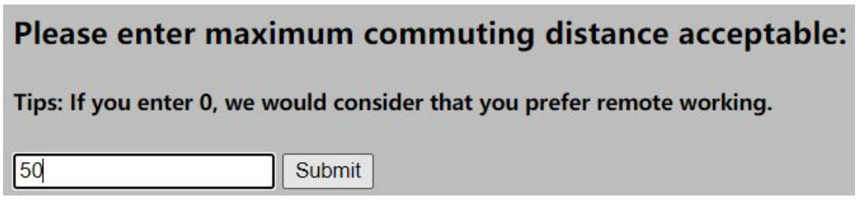
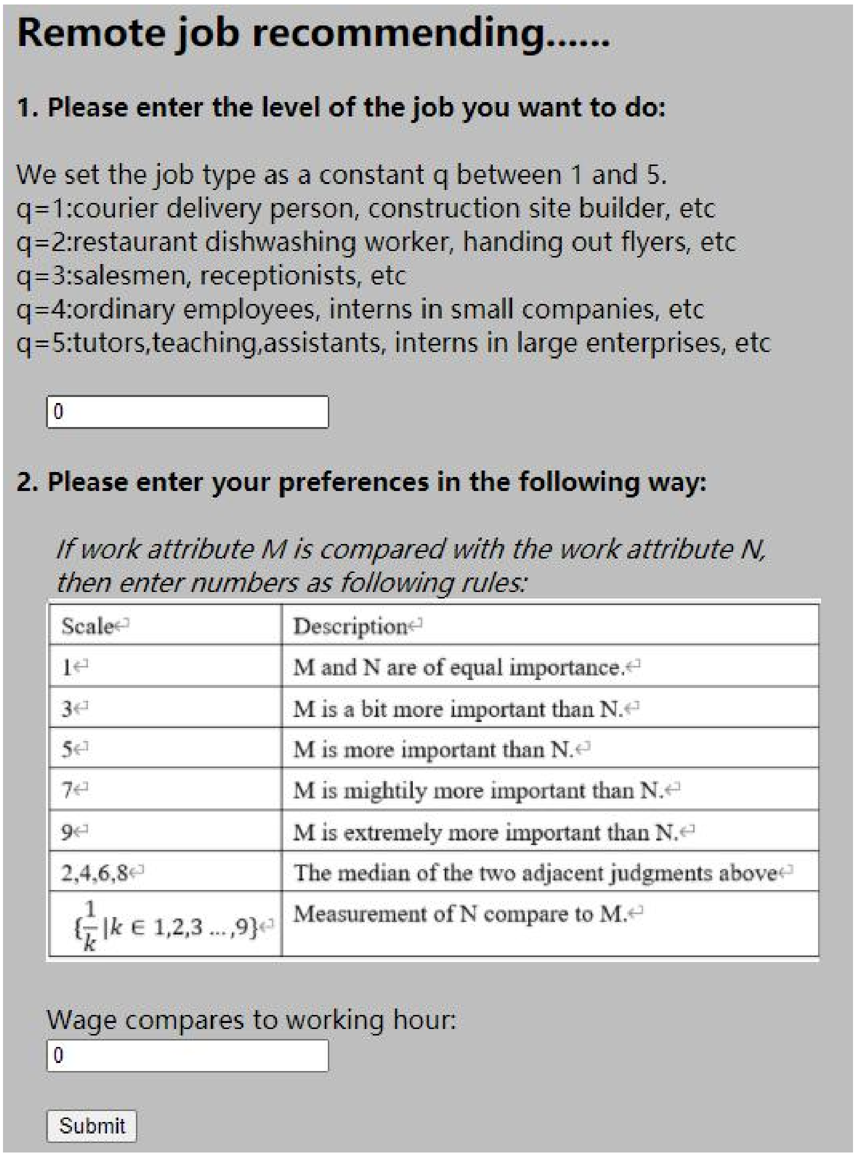


Figure 12 Maximum acceptable commuting distance inputing interface

Step 2A: If zero is entered, the page will jump to remote job search interface.



Remote job recommending......

1. Please enter the level of the job you want to do:

We set the job type as a constant q between 1 and 5.
q=1:courier delivery person, construction site builder, etc
q=2:restaurant dishwashing worker, handing out flyers, etc
q=3:salesmen, receptionists, etc
q=4:ordinary employees, interns in small companies, etc
q=5:tutors,teaching,assistants, interns in large enterprises, etc

0

2. Please enter your preferences in the following way:

If work attribute M is compared with the work attribute N, then enter numbers as following rules:

Scale	Description
1	M and N are of equal importance.
3	M is a bit more important than N.
5	M is more important than N.
7	M is mightily more important than N.
9	M is extremely more important than N.
2,4,6,8	The median of the two adjacent judgments above
$\{\frac{1}{k} \| k \in 1,2,3 \ldots ,9\}$	Measurement of N compare to M.

Wage compares to working hour:

0

Submit

Figure 13 Remote job searching interface

Step 2B: If the user entered a number greater than zero, the page will jump to offline job search interface.

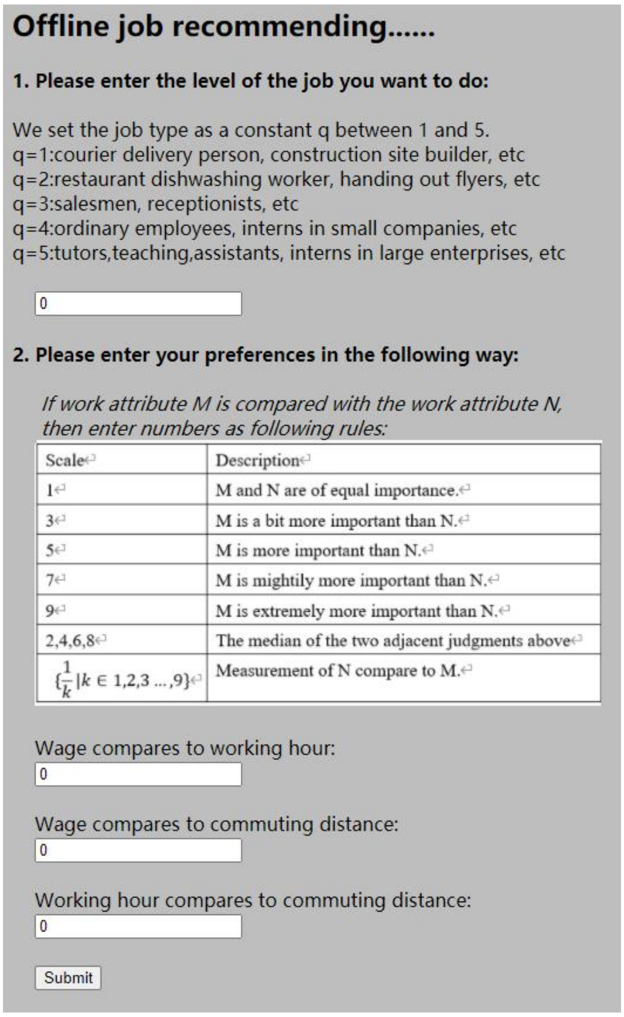


Figure 14 Offline job searching interface

Step 3: According to user's input, it will pop up a window with the results. Take the input of Step 2B as an example, and it pops out the result of location of the job (x_q,y_q) ,wage w , working hours t , commuting distance d and recommend commuting transpertation:

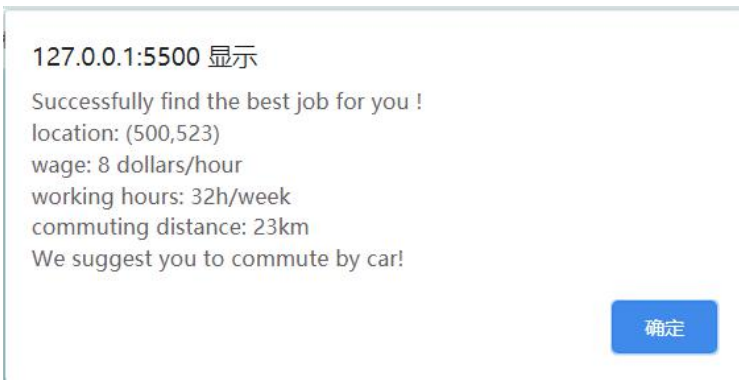


Figure 15 Result-pop-up window

7 Strength and weakness

7.1 Strength

Our model has been adequately tested and proved to be effective and has successfully completed the requirements of the competition. Even meeting demanding users, we can find good results.

Our model

7.2 Weakness

1) We assume that every college student chooses jobs based on wage, working hours, distance (includes whether to accept offline), the difficulty index of the job, ignoring the influence of gender and grade.

2) In the hypothesis part, we assume that students have taken the time and cost of transportation into account in their preference for working distance, so the time and cost of transportation are not involved in the modeling. Therefore, the time and cost of transportation cannot be considered quantitatively.

3) We assume that each college student will be able to perform all of a certain intensity of jobs(example : $Q = 3$ means the student has the ability to finish all of this kind of job ,no matter the job is modeling or essay writing).The same intensity of work may require college students to master different types of skills that we deliberately ignore in the modeling process.

4) In 4.3.1, we adopted a new nondimensionalize method. In terms of the treatment of working hours, we suppose that the fewer the working hours, the better. Rigor of this method remains to be explored.

7.3 Improvement directions

Corresponding to the above description, our model optimizes the direction as follows:

1) Take influence of gender and grade into account.
2) Refine the model's treatment of time-relevant factor.
3) Refine the model’s treatment of job categories.
4) Reach to more appropriate job attribute evaluation index.

8 Reference

[1] College Student Jobs, Employment | Indeed.com 2020.11.20 https://www.indeed.com/jobs?q=College%20Student&remotejob=032b3046-06a3-4876-8dfd-474eb5e7ed11&vjk=10a5ad6470c2fd90

[2] College Student Jobs, Employment | Indeed.com 2020.11.20
https://cn.indeed.com/jobs?q=%E5%A4%A7%E5%AD%A6%E7%94%9F&l=%E4%B8%8A%E6%B5%B7%E5%B8%82&radius=0&vjk=3eb698b794a8c2d6

[3] 58.com 2020.11.20
https://nn.58.com/cantfwy/?PGTID=0d3025a8-0034-dec8-07b1-75fed3aa8308&ClickID=3

[4] Yansong WANG. Research of reasonable buried depth of tunnel in urban underground road based on analytic hierarchy process [C]. Proceedings of 2020 Industrial Architecture Academic Exchange Conference (Vol. 1). Beijing, China: Industrial Building Magazine,2020.228-232.

[5] LI Heng, ZHOU Hongwen , ZHOU Bentao, SONG Hongrui. Study on comprehensive evaluation method of steel box girderbridge quality based on fuzzy analytic hierarchy process [C]. Proceedings of 2020 Industrial Architecture Academic Exchange Conference (Vol. 2). Beijing, China: Industrial Building Magazine,2020.199-205.

9 Appendice

randi.mlx

```
side=1000; %区域边长
numjob=10000; %发达地区线下工作数量
numljob=10000; %欠发达地区线下工作数量
numujob=1000; %全部地区远程工作数量
D1Q11X=randi([floor(side*1/6),floor(side*2/3)],floor(0.1535*numljob),1); %欠发达地区 Low Requirement(q=1)随机数
D1Q11Y=randi([0,side],floor(0.1535*numljob),1);
D1Q11W=randi([1,3],floor(0.1535*numljob),1);
D1Q11H=randi([20,40],floor(0.1535*numljob),1);
D1Q11=[D1Q11X,D1Q11Y,D1Q11W,D1Q11H];
```

```
D1Q21X=randi([floor(side*1/6),floor(side*2/3)],floor(0.4386*numljob),1); %欠发达地区 Mid-Low Requirement(q=2)随机数
D1Q21Y=randi([0,side],floor(0.4386*numljob),1);
D1Q21W=randi([3,8],floor(0.4386*numljob),1);
D1Q21H=randi([30,40],floor(0.4386*numljob),1);
D1Q21=[D1Q21X,D1Q21Y,D1Q21W,D1Q21H];
D1Q31X=randi([floor(side*1/6),floor(side*2/3)],floor(0.3289*numljob),1); %欠发达地区 Medium Requirement(q=3)随机数
D1Q31Y=randi([0,side],floor(0.3289*numljob),1);
D1Q31W=randi([8,10],floor(0.3289*numljob),1);
D1Q31H=randi([30,50],floor(0.3289*numljob),1);
D1Q31=[D1Q31X,D1Q31Y,D1Q31W,D1Q31H];
D1Q41X=randi([floor(side*1/6),floor(side*2/3)],floor(0.0768*numljob),1); %欠发达地区 Mid-High Requirement(q=4)随机数
D1Q41Y=randi([0,side],floor(0.0768*numljob),1);
D1Q41W=randi([10,20],floor(0.0768*numljob),1);
D1Q41H=randi([30,50],floor(0.0768*numljob),1);
D1Q41=[D1Q41X,D1Q41Y,D1Q41W,D1Q41H];
numljobh=numljob-floor(0.1535*numljob)-floor(0.4386*numljob)-floor(0.3289*numljob)-floor(0.0768*numljob); %欠发达地区 High Requirement(q=5)随机数的数量
D1Q51X=randi([floor(side*1/6),floor(side*2/3)],numljobh,1); %欠发达地区 High Requirement(q=5)随机数
D1Q51Y=randi([0,side],numljobh,1);
D1Q51W=randi([20,25],numljobh,1);
D1Q51H=randi([40,50],numljobh,1);
D1Q51=[D1Q51X,D1Q51Y,D1Q51W,D1Q51H];
D1Q12X1=randi([0,floor(side*1/6)],ceil(0.0400*numjob*1/3),1); %发达地区 Low Requirement(q=1)随机数
D1Q12X2=randi([floor(side*2/3),side],floor(0.0400*numjob*2/3),1);
D1Q12X=[D1Q12X1;D1Q12X2];
D1Q12Y=randi([0,side],0.0400*numjob,1);
D1Q12W=randi([3,5],0.0400*numjob,1);
D1Q12H=randi([20,40],0.0400*numjob,1);
D1Q12=[D1Q12X,D1Q12Y,D1Q12W,D1Q12H];
```

```
D1Q22X1=randi([0,floor(side*1/6)],ceil(0.1000*numjob*1/3),1); %发达地区
  Mid-Low Requirement(q=2)随机数
D1Q22X2=randi([floor(side*2/3),side],floor(0.1000*numjob*2/3),1);
D1Q22X=[D1Q22X1;D1Q22X2];
D1Q22Y=randi([0,side],0.1000*numjob,1);
D1Q22W=randi([3,10],0.1000*numjob,1);
D1Q22H=randi([20,30],0.1000*numjob,1);
D1Q22=[D1Q22X,D1Q22Y,D1Q22W,D1Q22H];
D1Q32X1=randi([0,floor(side*1/6)],ceil(0.2600*numjob*1/3),1); %发达地区
  Medium Requirement(q=3)随机数
D1Q32X2=randi([floor(side*2/3),side],floor(0.2600*numjob*2/3),1);
D1Q32X=[D1Q32X1;D1Q32X2];
D1Q32Y=randi([0,side],0.2600*numjob,1);
D1Q32W=randi([10,15],0.2600*numjob,1);
D1Q32H=randi([20,50],0.2600*numjob,1);
D1Q32=[D1Q32X,D1Q32Y,D1Q32W,D1Q32H];
D1Q42X1=randi([0,floor(side*1/6)],ceil(0.3600*numjob*1/3),1); %发达地区
  Mid-High Requirement(q=4)随机数
D1Q42X2=randi([floor(side*2/3),side],floor(0.3600*numjob*2/3),1);
D1Q42X=[D1Q42X1;D1Q42X2];
D1Q42Y=randi([0,side],0.3600*numjob,1);
D1Q42W=randi([20,30],0.3600*numjob,1);
D1Q42H=randi([40,50],0.3600*numjob,1);
D1Q42=[D1Q42X,D1Q42Y,D1Q42W,D1Q42H];
D1Q52X1=randi([0,floor(side*1/6)],ceil(0.2400*numjob*1/3),1); %发达地区 High
  Requirement(q=5)随机数
D1Q52X2=randi([floor(side*2/3),side],floor(0.2400*numjob*2/3),1);
D1Q52X=[D1Q52X1;D1Q52X2];
D1Q52Y=randi([0,side],0.2400*numjob,1);
D1Q52W=randi([30,50],0.2400*numjob,1);
D1Q52H=randi([40,50],0.2400*numjob,1);
D1Q52=[D1Q52X,D1Q52Y,D1Q52W,D1Q52H];
D1Q1=[D1Q11;D1Q12]; %全部地区 Low Requirement(q=1)随机数
D1Q2=[D1Q21;D1Q22]; %全部地区 Mid-Low Requirement(q=2)随机数
```

```
D1Q3=[D1Q31;D1Q32]; %全部地区 Medium Requirement(q=3)随机数
D1Q4=[D1Q41;D1Q42]; %全部地区 Mid-High Requirement(q=4)随机数
D1Q5=[D1Q51;D1Q52]; %全部地区 High Requirement(q=5)随机数
D0Q3XY=randi([0,side],floor(0.34*numujob),2);
D0Q3W=randi([15,20],floor(0.34*numujob),1);
D0Q3H=randi([20,50],floor(0.34*numujob),1);
D0Q3=[D0Q3XY,D0Q3W,D0Q3H]; %全部地区远程工作 Medium Requirement(q=3)
  随机数
D0Q5XY=randi([0,side],numujob-floor(0.34*numujob),2);
D0Q5W=randi([25,50],numujob-floor(0.34*numujob),1);
D0Q5H=randi([20,40],numujob-floor(0.34*numujob),1);
D0Q5=[D0Q5XY,D0Q5W,D0Q5H]; %全部地区远程工作 High Requirement(q=5)随
  机数
D1Q1(:,5)=1; %标记工作难度
D1Q2(:,5)=2;
D1Q3(:,5)=3;
D1Q4(:,5)=4;
D1Q5(:,5)=5;
D0Q3(:,5)=3;
D0Q5(:,5)=5;
```

Findjob.m

```
function [] = Findjob(A,S,X,Y,D)
randijob;                                                        %随
  机生成全部工作
d=0;
if D>0                                                           %提
  取出符合条件的线下工作
    switch S(1,1)                                                %将
  满足矩阵 S 中第一个点的工作赋给 d
        case 1
            d=D1Q1;
        case 2
            d=D1Q2;
        case 3
```

```
                d=D1Q3;
            case 4
                d=D1Q4;
            case 5
                d=D1Q5;
        end
        while size(S,2)>1                                          %每赋
  值一次，S 列数减一
            S=S(1,2:size(S,2));
                switch S(1,1)                                      %将满
  足矩阵 S 各点的工作放入 d
                    case 1
                        d=[d;D1Q1];
                    case 2
                        d=[d;D1Q2];
                    case 3
                        d=[d;D1Q3];
                    case 4
                        d=[d;D1Q4];
                    case 5
                        d=[d;D1Q5];
                end
        end
else                                                               %提取
  出符合条件的远程工作
        switch S(1,1)
            case 3
                d=D0Q3;
            case 5
                d=D0Q5;
        end
        if size(S,2)>1
            d=[D0Q3;D0Q5];
        end
```

```
end
for i=1:size(d,1)                                           %计算学
  生坐标与每个工作的距离
    B(i,1)=ceil(((X-d(i,1))^2+(Y-d(i,2))^2)^(1/2));
end
d=[d,B];
if(D>0)
    o=1;
    for i=1:size(d,1)                                       %留下在
  大学生可接受距离内的工作
        if d(i,6)<=D
            d(o,:)=d(i,:);
            o=o+1;
        end
    end
end
if o==1
    error('There is no work of this type in this scope\n');
end
d=d(1:(o-1),:);
B1(10,10)=0;
B2(10,10)=0;
B3(10,10)=0;
c=0;
d(1,1)=0;
if(D>0)                                                       %选
  择线下工作的情况下，其工作重要性权重与时薪、时长、距离有关
    while(size(d,1)>1)                                       %循环，
  直到筛选出最适合此学生的工作
        if size(d,1)>10
            for j=1:10
                for k=1:10
                    m=max(d(1:10,3))-min(d(1:10,3));         %十个工
  作中时薪极差
```

```
B1(10,10)=0;
if d(j,3)>d(k,3)
      B1(j,k)=ceil((d(j,3)-d(k,3))*9/m);            %去量纲化
构造成对比较矩阵
elseif d(j,3)<d(k,3)
      B1(j,k)=1/ceil((d(k,3)-d(j,3))*9/m);        %去量纲化
构造成对比较矩阵
else
      B1(j,k)=1;
end
m=max(d(1:10,4))-min(d(1:10,4));                    %十个工
作中时长极差
B2(10,10)=0;
if d(j,4)>d(k,4)
      B2(j,k)=1/ceil((d(j,4)-d(k,4))*9/m);        %去量纲化,
构造成对比较矩阵
elseif d(j,4)<d(k,4)
      B2(j,k)=ceil((d(k,4)-d(j,4))*9/m);            %去量纲化
构造成对比较矩阵
else
        B2(j,k)=1;
end
m=max(d(1:10,6))-min(d(1:10,6));                      %十个
工作中距离极差
B3(10,10)=0;
if d(j,6)>d(k,6)
      B3(j,k)=1/ceil((d(j,6)-d(k,6))*9/m);          %去量纲化
构造成对比较矩阵
elseif d(j,6)<d(k,6)
      B3(j,k)=ceil((d(k,6)-d(j,6))*9/m);              %去量纲化
构造成对比较矩阵
else
      B3(j,k)=1;
end
```

```
            end
        end
        [W, Lmax, CI, CR] = AHP(A);                                %用户
输入的 A   时薪、时长、距离的权重
        [W1, Lmax1, CI1, CR1] = AHP(B1);                           %各工
作时薪权重
        [W2, Lmax2, CI2, CR2] = AHP(B2);                           %各工
作时长权重
        [W3, Lmax3, CI3, CR3] = AHP(B3);                           %各工
作距离权重
        rlts3 =[W1 W2 W3;                                          %合
并结果
            Lmax1 Lmax2 Lmax3;
            CI1 CI2 CI3;
            CR1 CR2 CR3];
        a10 = rlts3(1:10,:)   *   W;
        c=find(a10==max(a10(:)));                                  %找到
权重最大的工作所在的位置
        f=d(c,:);                                                  %f存储
权重最大的工作所在的位置
        d=[f;d(11:size(d,1),:)];                                   %滤除前十
个工作中较不合适的九个矩阵
    else
        n=size(d,1);
        B1=0;B2=0;B3=0;
        B1(n,n)=0;
        B2(n,n)=0;
        B3(n,n)=0;
        for j=1:n
            for k=1:n
                m=max(d(1:n,3))-min(d(1:n,3));                     %最后少
于十个工作中时薪极差
                B1(n,n)=0;
                if d(j,3)>d(k,3)
```

```
                B1(j,k)=ceil((d(j,3)-d(k,3))*9/m);          %去量纲化
构造成对比较矩阵
            elseif d(j,3)<d(k,3)
                B1(j,k)=1/ceil((d(k,3)-d(j,3))*9/m);        %去量纲化 构
造成对比较矩阵
            else
                B1(j,k)=1;
            end
            m=max(d(1:n,4))-min(d(1:n,4));                  %最后少
于十个工作中时长极差
            B2(n,n)=0;
            if d(j,4)>d(k,4)
                B2(j,k)=1/ceil((d(j,4)-d(k,4))*9/m);
            elseif d(j,4)<d(k,4)
                B2(j,k)=ceil((d(k,4)-d(j,4))*9/m);
            else
                B2(j,k)=1;
            end
            m=max(d(1:n,6))-min(d(1:n,6));                  %最后
少于十个工作中距离极差
            B3(n,n)=0;
            if d(j,6)>d(k,6)
                B3(j,k)=1/ceil((d(j,6)-d(k,6))*9/m);        %去量纲化
构造成对比较矩阵
            elseif d(j,6)<d(k,6)
                B3(j,k)=ceil((d(k,6)-d(j,6))*9/m);          %去量纲化
构造成对比较矩阵
            else
                B3(j,k)=1;
            end
        end
    end
    [W, Lmax, CI, CR] = AHP(A);                             %用
户输入的 A
```

```
            [W1, Lmax1, CI1, CR1] = AHP(B1);                          %各
工作时薪权重
            [W2, Lmax2, CI2, CR2] = AHP(B2);                          %各
工作时长权重
            [W3, Lmax3, CI3, CR3] = AHP(B3);                          %各
工作距离权重
            rlts3 =[W1 W2 W3;                                          %合
并结果
                Lmax1 Lmax2 Lmax3;
                CI1 CI2 CI3;
                CR1 CR2 CR3];
            a10 = rlts3(1:n,:)   *   W;
            c=find(a10==max(a10(:)));                                %找到
少于 n 个工作中权重最大的矩阵
            d=d(c,:);                                                 %留下
最合适的工作
        end
    end
else                                                                  %选
择远程工作的情况下，其工作重要性权重与时薪、时长有关
        while(size(d,1)>1)                                           %循环，
直到筛选出最适合此学生的工作
            if size(d,1)>10
                for j=1:10
                    for k=1:10

m=max(d(1:10,3))-min(d(1:10,3));                      %十个工作中时薪极差
                        B1(10,10)=0;
                        if d(j,3)>d(k,3)
                            B1(j,k)=ceil((d(j,3)-d(k,3))*9/m);        %去
量纲化   构造成对比较矩阵
                        elseif d(j,3)<d(k,3)
                            B1(j,k)=1/ceil((d(k,3)-d(j,3))*9/m);      %去
量纲化   构造成对比较矩阵
```

```
                else
                    B1(j,k)=1;
                end
                m=max(d(1:10,4))-min(d(1:10,4));                    %十个
工作中时长极差
                B2(10,10)=0;
                if d(j,4)>d(k,4)
                    B2(j,k)=1/ceil((d(j,4)-d(k,4))*9/m);      %去量纲化
构造成对比较矩阵
                elseif d(j,4)<d(k,4)
                    B2(j,k)=ceil((d(k,4)-d(j,4))*9/m);         %去量纲
化  构造成对比较矩阵
                else
                    B2(j,k)=1;
                end
            end
        end
        [W, Lmax, CI, CR] = AHP(A);                                   %用
户输入的 A
        [W1, Lmax1, CI1, CR1] = AHP(B1);                              %各
工作时薪权重
        [W2, Lmax2, CI2, CR2] = AHP(B2);                              %各
工作时长权重
        rlts2 =[W1 W2;                                                 %
合并结果
            Lmax1 Lmax2;
            CI1 CI2;
            CR1 CR2];
        a10 = rlts2(1:10,:)    *    W;
        c=find(a10==max(a10(:)));                                     %找
到权重最大的工作所在的位置
        f=d(c,:);                                                     %存
储此工作
```

```
d=[f;d(11:size(d,1),:)];                                    %滤除前
十个工作中较不合适的九个矩阵
    else
        n=size(d,1);
        B1=0;B2=0;
        B1(n,n)=0;
        B2(n,n)=0;
        for j=1:n
            for k=1:n
                m=max(d(1:n,3))-min(d(1:n,3));              %n 个
工作中时薪极差
                B1(n,n)=0;
                if d(j,3)>d(k,3)
                    B1(j,k)=ceil((d(j,3)-d(k,3))*9/m);      %去量纲
化  构造成对比较矩阵
                elseif d(j,3)<d(k,3)
                    B1(j,k)=1/ceil((d(k,3)-d(j,3))*9/m);    %去量纲化
构造成对比较矩阵
                else
                    B1(j,k)=1;
                end
                m=max(d(1:n,4))-min(d(1:n,4));              %n 个
工作中时长极差
                B2(n,n)=0;
                if d(j,4)>d(k,4)
                    B2(j,k)=1/ceil((d(j,4)-d(k,4))*9/m);    %去量纲化
构造成对比较矩阵
                elseif d(j,4)<d(k,4)
                    B2(j,k)=ceil((d(k,4)-d(j,4))*9/m);      %去量纲
化  构造成对比较矩阵
                else
                    B2(j,k)=1;
                end
            end
```

```
                end
                [W, Lmax, CI, CR] = AHP(A);                    %用
户输入的 A
                [W1, Lmax1, CI1, CR1] = AHP(B1);               %各
工作时薪权重
                [W2, Lmax2, CI2, CR2] = AHP(B2);               %各
工作时长权重
                rlts3 =[W1 W2;                                 %
合并结果
                    Lmax1 Lmax2;
                    CI1 CI2;
                    CR1 CR2];
                a10 = rlts3(1:n,:)  *   W;
                c=find(a10==max(a10(:)));                      %找到
权重最大的工作所在的位置
                d=d(c,:);                                      %留
下最符合条件的工作
            end
        end
end
fprintf('A most suitable job:\n');
fprintf('     position     :(%d,%d)\n',d(1,1),d(1,2));        %最符合条件
工作的坐标
fprintf('       wage       :     %d\n',d(1,3));              %最符合
条件工作的时薪
fprintf('      hours       :     %d\n',d(1,4));              %最符合
条件工作的每周工作时长
fprintf('difficulty index:     %d\n',d(1,5));                %最符合条件
工作的难度指数
fprintf('     distance     :     %d\n',d(1,6));              %最符合条
件工作的距离
```

AHP.m

```
function [W, Lmax, CI, CR] = AHP(A)
% 实现单层次结构的层次分析法
```

```
% 输入: A 为成对比较矩阵
% 输出: W 为权重向量, Lmax 为最大特征值, CI 为一致性指标, CR 为一致性比率

[V,D] = eig(A);
[Lmax,ind] = max(diag(D));            % 求最大特征值及其位置
W = V(:,ind) / sum(V(:,ind));        % 最大特征值对应的特征向量做标准化
Lmax = mean((A * W) ./ W);            % 计算最大特征值
n = size(A, 1);                       % 矩阵行数
CI = (Lmax - n) / (n - 1);           % 计算一致性指标
% Saaty 随机一致性指标值
RI = [0 0 0.58 0.90 1.12 1.24 1.32 1.41 1.45 1.49 1.51];
CR = CI / RI(n);                      % 计算一致性比率
if CR>=0.1
    error('未通过一致性检验，程序退出');
end
```