# ECO394D Probability and Statistics Homework I

Mingwei Li*(ID:ml49942)

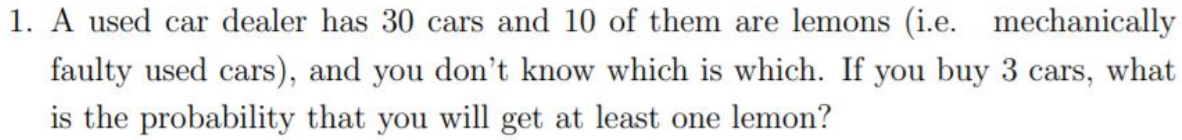
**Abstract**

The contents of this essay is the week-one homework of ECO394D Probability and Statistics.

*Mingwei Li is a master student of economics, The University of Texas at Austin

# Contents

# 1 Mathematical Problems



1. A used car dealer has 30 cars and 10 of them are lemons (i.e. mechanically faulty used cars), and you don't know which is which. If you buy 3 cars, what is the probability that you will get at least one lemon?

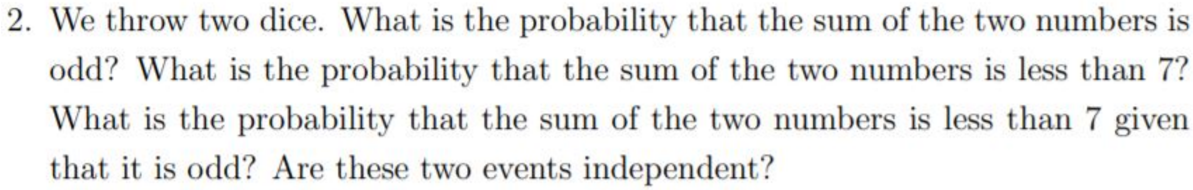
**Proof:**

P(no lemon)=1-P(at least one lemon)

total_num=$C_{30}^{3}$=$\frac{30!}{3!27!}$=$\frac{30\times29\times28}{6}$, no_lemon_num=$C_{20}^{3}$=$\frac{20!}{3!17!}$=$\frac{20\times19\times18}{6}$

P(no lemon)= no_lemon_num/total_num=$\frac{20\times19\times18}{30\times29\times28}$=0.2807

$\therefore$ P(at least one lemon)=1-P(no lemon)=0.7192



2. We throw two dice. What is the probability that the sum of the two numbers is odd? What is the probability that the sum of the two numbers is less than 7? What is the probability that the sum of the two numbers is less than 7 given that it is odd? Are these two events independent?

**Proof:**

total_num=6×6=36

Notice the fact that 1,3,5 are odd, and 2,4,6 are even.

P(The sum of 2 numbers is odd)=P(First is odd and Second is even $\cup$ First is even and Second is odd)

odd_num=2×3×3=18

$\therefore$ **P(The sum of 2 numbers is odd)**=odd_num/total_num=$\frac{18}{36}$=0.5

There are 15 circumstances under which the sum of 2 numbers is less than 7

(1,1),(1,2),(1,3),(1,4),(1,5),(2,1),(2,2),(2,3),(2,4),(3,1),(3,2),(3,3),(4,1),(4,2),(5,1)

$\therefore$ **P(The sum of 2 numbers is less than 7)**=$\frac{15}{36}$=0.4167

P(The sum of 2 numbers is less than 7|The sum of 2 numbers is odd)=P(The sum of 2 numbers is less than 7$\cap$The sum of 2 numbers is odd)/P(The sum of 2 numbers is odd)

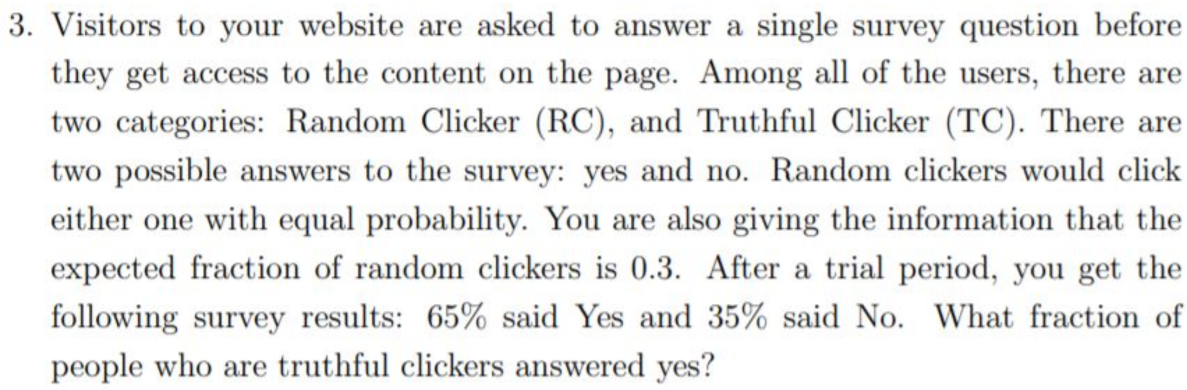
(1,2),(1,4),(2,1),(2,3),(3,2),(4,1) 6 items whose sum is both odd and less than 7.

$\therefore$ **P(The sum of 2 numbers is less than 7|The sum of 2 numbers is odd)**=$\frac{6}{18}$=0.3333

P(The sum of 2 numbers is less than 7$\cap$The sum of 2 numbers is odd)=$\frac{6}{36}$=$\frac{1}{6}$

**P(The sum of 2 numbers is less than 7)**×**P(The sum of 2 numbers is odd)**=$\frac{15}{72}$=$\frac{5}{24}$

$\because \frac{5}{24} \neq \frac{1}{6}$ $\therefore$ These 2 events aren't independent.



3. Visitors to your website are asked to answer a single survey question before they get access to the content on the page. Among all of the users, there are two categories: Random Clicker (RC), and Truthful Clicker (TC). There are two possible answers to the survey: yes and no. Random clickers would click either one with equal probability. You are also giving the information that the expected fraction of random clickers is 0.3. After a trial period, you get the following survey results: 65% said Yes and 35% said No. What fraction of people who are truthful clickers answered yes?

**Proof:**

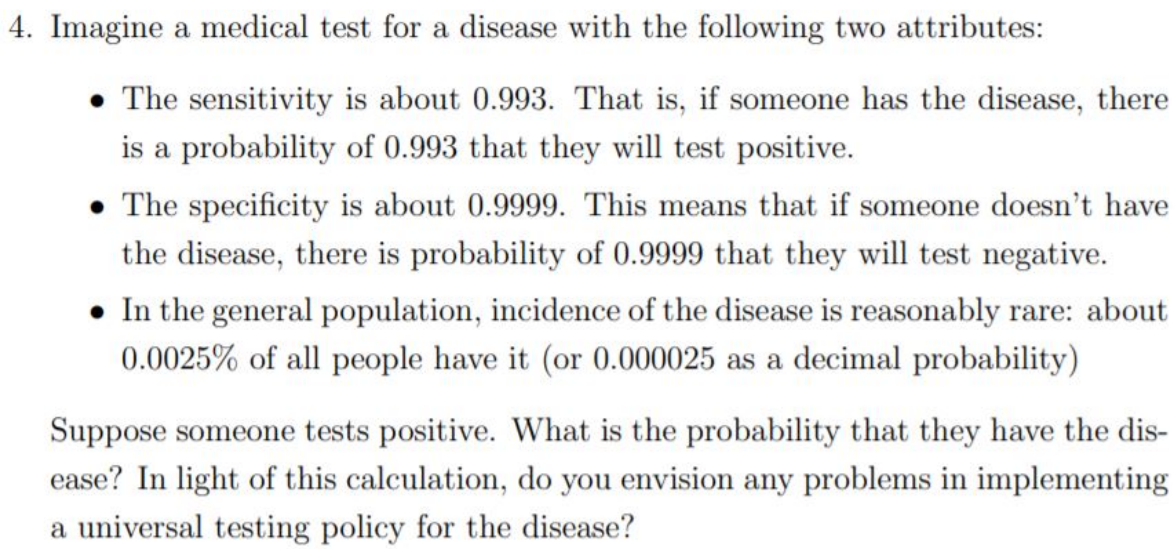
Given the info we have: random clickers occupies a fraction of 0.3, while truthful clickers takes 0.7. We assume that $y$ of truthful clickers said yes and the remaining $1 - y$ of truthful clickers said no. Therefore, we get following equations.

$$0.3 \times 0.5 + 0.7 \times y = 0.65$$

$$0.3 \times 0.5 + 0.7 \times (1 - y) = 0.35$$

It's clear that $y$ takes $\frac{5}{7}$.

As a result, $\frac{5}{7} \times 0.7$=0.5 of people are truthful clickers answered yes.



4. Imagine a medical test for a disease with the following two attributes:

   - The sensitivity is about 0.993. That is, if someone has the disease, there is a probability of 0.993 that they will test positive.
   - The specificity is about 0.9999. This means that if someone doesn't have the disease, there is probability of 0.9999 that they will test negative.
   - In the general population, incidence of the disease is reasonably rare: about 0.0025% of all people have it (or 0.000025 as a decimal probability)

   Suppose someone tests positive. What is the probability that they have the disease? In light of this calculation, do you envision any problems in implementing a universal testing policy for the disease?

**Proof:**

Our goal is to calculate P(Have the disease|Test positive)

What we know is:

P(Has the disease)=0.000025, P(Test negative|Doesn't have the disease)=0.9999

P(Test positive|Has the disease)=0.993

According to the info we know:

P(Has the disease∩Test positive)=P(Test positive|Has the disease)×P(Has the disease)=0.993×0.000025=2.4825e-05

∵ P(Test negative|Doesn't have the disease)=0.9999 ∴ P(Test positive|Doesn't have the disease)=0.0001

∴ P(Test positive)=P(Test positive|Doesn't have the disease)×P(Doesn't have the disease)+P(Test positive|Has the disease)×P(Has the disease)=0.0001×(1-0.000025)+0.993×0.000025=0.0001248225

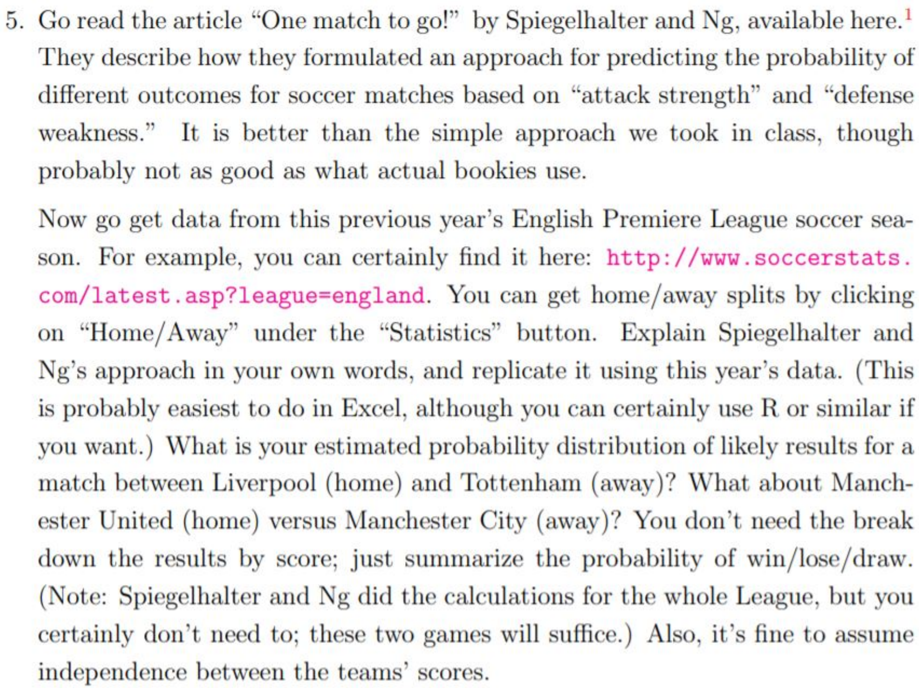
∴ P(Has the disease|Test positive)=P(Has the disease∩Test positive)/P(Test positive)=2.4825e-05/0.0001248225=0.1989

**Problem:**A huge number of positive results would be false if we implement a universal testing. This is a huge waste of money!

# 2 Data Analysis

## 2.1 Introduction

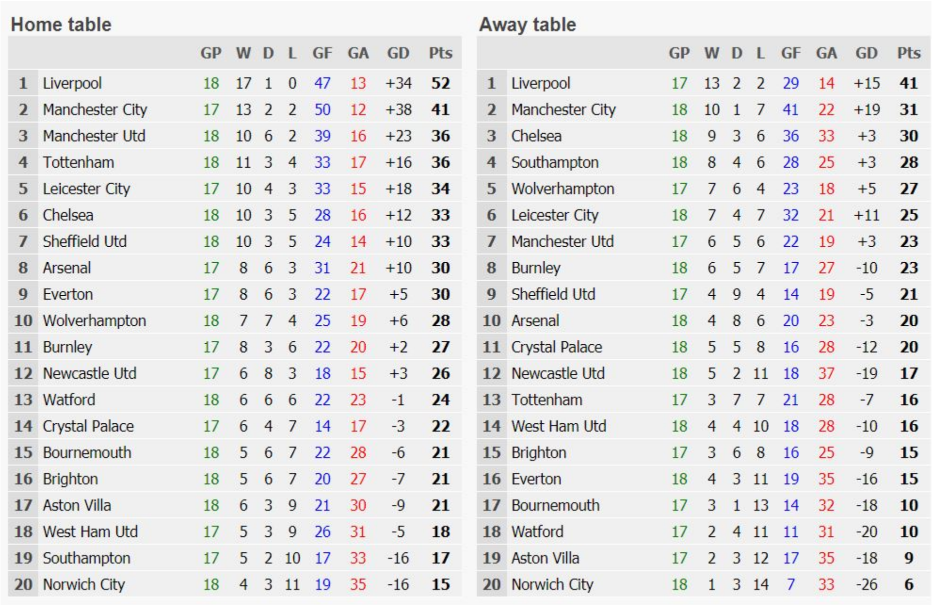
The problem description is as followed.



5. Go read the article “One match to go!” by Spiegelhalter and Ng, available here.[1] They describe how they formulated an approach for predicting the probability of different outcomes for soccer matches based on “attack strength” and “defense weakness.” It is better than the simple approach we took in class, though probably not as good as what actual bookies use.

   Now go get data from this previous year’s English Premiere League soccer season. For example, you can certainly find it here: http://www.soccerstats.com/latest.asp?league=england. You can get home/away splits by clicking on “Home/Away” under the “Statistics” button. Explain Spiegelhalter and Ng’s approach in your own words, and replicate it using this year’s data. (This is probably easiest to do in Excel, although you can certainly use R or similar if you want.) What is your estimated probability distribution of likely results for a match between Liverpool (home) and Tottenham (away)? What about Manchester United (home) versus Manchester City (away)? You don’t need the break down the results by score; just summarize the probability of win/lose/draw. (Note: Spiegelhalter and Ng did the calculations for the whole League, but you certainly don’t need to; these two games will suffice.) Also, it’s fine to assume independence between the teams’ scores.

The data of our analysis problem is as followed. Our data is the performance of different teams of Premier League in 2019-2020 season.



Home table

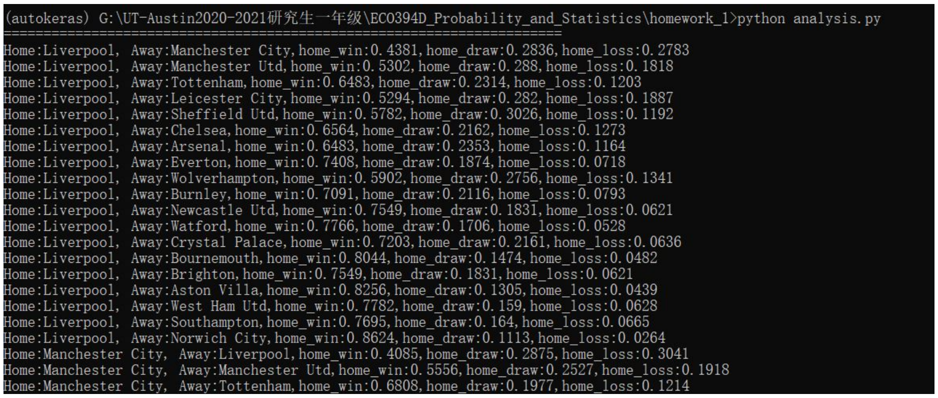
| | | GP | W | D | L | GF | GA | GD | Pts |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Liverpool | 18 | 17 | 1 | 0 | 47 | 13 | +34 | 52 |
| 2 | Manchester City | 17 | 13 | 2 | 2 | 50 | 12 | +38 | 41 |
| 3 | Manchester Utd | 18 | 10 | 6 | 2 | 39 | 16 | +23 | 36 |
| 4 | Tottenham | 18 | 11 | 3 | 4 | 33 | 17 | +16 | 36 |
| 5 | Leicester City | 17 | 10 | 4 | 3 | 33 | 15 | +18 | 34 |
| 6 | Chelsea | 18 | 10 | 3 | 5 | 28 | 16 | +12 | 33 |
| 7 | Sheffield Utd | 18 | 10 | 3 | 5 | 24 | 14 | +10 | 33 |
| 8 | Arsenal | 17 | 8 | 6 | 3 | 31 | 21 | +10 | 30 |
| 9 | Everton | 17 | 8 | 6 | 3 | 22 | 17 | +5 | 30 |
| 10 | Wolverhampton | 18 | 7 | 7 | 4 | 25 | 19 | +6 | 28 |
| 11 | Burnley | 17 | 8 | 3 | 6 | 22 | 20 | +2 | 27 |
| 12 | Newcastle Utd | 17 | 6 | 8 | 3 | 18 | 15 | +3 | 26 |
| 13 | Watford | 18 | 6 | 6 | 6 | 22 | 23 | -1 | 24 |
| 14 | Crystal Palace | 17 | 6 | 4 | 7 | 14 | 17 | -3 | 22 |
| 15 | Bournemouth | 18 | 5 | 6 | 7 | 22 | 28 | -6 | 21 |
| 16 | Brighton | 18 | 5 | 6 | 7 | 20 | 27 | -7 | 21 |
| 17 | Aston Villa | 18 | 6 | 3 | 9 | 21 | 30 | -9 | 21 |
| 18 | West Ham Utd | 17 | 5 | 3 | 9 | 26 | 31 | -5 | 18 |
| 19 | Southampton | 17 | 5 | 2 | 10 | 17 | 33 | -16 | 17 |
| 20 | Norwich City | 18 | 4 | 3 | 11 | 19 | 35 | -16 | 15 |

Away table

| | | GP | W | D | L | GF | GA | GD | Pts |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Liverpool | 17 | 13 | 2 | 2 | 29 | 14 | +15 | 41 |
| 2 | Manchester City | 18 | 10 | 1 | 7 | 41 | 22 | +19 | 31 |
| 3 | Chelsea | 18 | 9 | 3 | 6 | 36 | 33 | +3 | 30 |
| 4 | Southampton | 18 | 8 | 4 | 6 | 28 | 25 | +3 | 28 |
| 5 | Wolverhampton | 17 | 7 | 6 | 4 | 23 | 18 | +5 | 27 |
| 6 | Leicester City | 18 | 7 | 4 | 7 | 32 | 21 | +11 | 25 |
| 7 | Manchester Utd | 17 | 6 | 5 | 6 | 22 | 19 | +3 | 23 |
| 8 | Burnley | 18 | 6 | 5 | 7 | 17 | 27 | -10 | 23 |
| 9 | Sheffield Utd | 17 | 4 | 9 | 4 | 14 | 19 | -5 | 21 |
| 10 | Arsenal | 18 | 4 | 8 | 6 | 20 | 23 | -3 | 20 |
| 11 | Crystal Palace | 18 | 5 | 5 | 8 | 16 | 28 | -12 | 20 |
| 12 | Newcastle Utd | 18 | 5 | 2 | 11 | 18 | 37 | -19 | 17 |
| 13 | Tottenham | 17 | 3 | 7 | 7 | 21 | 28 | -7 | 16 |
| 14 | West Ham Utd | 18 | 4 | 4 | 10 | 18 | 28 | -10 | 16 |
| 15 | Brighton | 17 | 3 | 6 | 8 | 16 | 25 | -9 | 15 |
| 16 | Everton | 18 | 4 | 3 | 11 | 19 | 35 | -16 | 15 |
| 17 | Bournemouth | 17 | 3 | 1 | 13 | 14 | 32 | -18 | 10 |
| 18 | Watford | 17 | 2 | 4 | 11 | 11 | 31 | -20 | 10 |
| 19 | Aston Villa | 17 | 2 | 3 | 12 | 17 | 35 | -18 | 9 |
| 20 | Norwich City | 18 | 1 | 3 | 14 | 7 | 33 | -26 | 6 |

20 teams are involved in the soccer game. Our **ultimate goal** is to calculate the probabilities of different outcomes between any two of these teams.

A part of our final results is as followed.



```
(autokeras) G:\UT-Austin2020-2021研究生一年级\ECO394D_Probability_and_Statistics\homework_1>python analysis.py
======================================================================
Home:Liverpool,  Away:Manchester City,home_win:0.4381,home_draw:0.2836,home_loss:0.2783
Home:Liverpool,  Away:Manchester Utd,home_win:0.5302,home_draw:0.288,home_loss:0.1818
Home:Liverpool,  Away:Tottenham,home_win:0.6483,home_draw:0.2314,home_loss:0.1203
Home:Liverpool,  Away:Leicester City,home_win:0.5294,home_draw:0.282,home_loss:0.1887
Home:Liverpool,  Away:Sheffield Utd,home_win:0.5782,home_draw:0.3026,home_loss:0.1192
Home:Liverpool,  Away:Chelsea,home_win:0.6564,home_draw:0.2162,home_loss:0.1273
Home:Liverpool,  Away:Arsenal,home_win:0.6483,home_draw:0.2353,home_loss:0.1164
Home:Liverpool,  Away:Everton,home_win:0.7408,home_draw:0.1874,home_loss:0.0718
Home:Liverpool,  Away:Wolverhampton,home_win:0.5902,home_draw:0.2756,home_loss:0.1341
Home:Liverpool,  Away:Burnley,home_win:0.7091,home_draw:0.2116,home_loss:0.0793
Home:Liverpool,  Away:Newcastle Utd,home_win:0.7549,home_draw:0.1831,home_loss:0.0621
Home:Liverpool,  Away:Watford,home_win:0.7766,home_draw:0.1706,home_loss:0.0528
Home:Liverpool,  Away:Crystal Palace,home_win:0.7203,home_draw:0.2161,home_loss:0.0636
Home:Liverpool,  Away:Bournemouth,home_win:0.8044,home_draw:0.1474,home_loss:0.0482
Home:Liverpool,  Away:Brighton,home_win:0.7549,home_draw:0.1831,home_loss:0.0621
Home:Liverpool,  Away:Aston Villa,home_win:0.8256,home_draw:0.1305,home_loss:0.0439
Home:Liverpool,  Away:West Ham Utd,home_win:0.7782,home_draw:0.159,home_loss:0.0628
Home:Liverpool,  Away:Southampton,home_win:0.7695,home_draw:0.164,home_loss:0.0665
Home:Liverpool,  Away:Norwich City,home_win:0.8624,home_draw:0.1113,home_loss:0.0264
Home:Manchester City,  Away:Liverpool,home_win:0.4085,home_draw:0.2875,home_loss:0.3041
Home:Manchester City,  Away:Manchester Utd,home_win:0.5556,home_draw:0.2527,home_loss:0.1918
Home:Manchester City,  Away:Tottenham,home_win:0.6808,home_draw:0.1977,home_loss:0.1214
```

## 2.2 Analysis

The code of this data analysis process is available at github. This is a simple data manipulation problem which doesn't contain modeling. However, in order to have a clear pipeline of data manipulation, object-oriented programming is highly recommended to deal with such problem.

The main class of this data analysis is as followed:

```python
class assemble:

    def __init__(self,num=2019):
        self.data_cache = parser_data(num)
        home_data,mean_home,mean_away = data_cache
        self.team_list = home_data.index

    def produce_result(self,home,away):
        expected_home_score,expected_away_score = cal_teams(home,away,self.data_cache)
        p_home_win,p_draw,p_home_loss = cal_probability(expected_home_score,expected_away_score)
        print("Home:{}, Away:{},home_win:{},home_draw:{},home_loss:{}".format(home,away,np.round(
                                                      p_home_win,4),np.round(p_draw,4),np.round(
                                                      p_home_loss,4)))

    def produce_whole(self,):
        print("="*70)
        for i in range(len(self.team_list)):
            for j in range(len(self.team_list)):
                if i!=j:
                    self.produce_result(team_list[i],team_list[j])
        print("="*70)
```