The dataset record.txt contains running records obtained from athletes from different countries in various types of athletics events (sprints and middle-distance). We have data about 55 countries (observations) and 6 records (variables): 100 meters, 200 meters, 400 meters, 800 meters, 1500 meters and 3000 meters.

Load the dataset record.txt in R, using the function read.table

In [86]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [100…
```
# sep는 데이터 분류 기준을 지정하는 파라미터, \s+는 스페이스바를 기준으로 한다는 의미이며 디폴트 값은 \t+ 탭이다.
data = pd.read_table('/Users/jimin/Desktop/지민/ewha/2023-2/Regression/week1/record.txt', sep="\s+", header=0)
data.head(10)
```

Out[100…

	m100	m200	m400	m800	m1500	m3000
argentin	11.61	22.94	54.50	129.0	265.8	587.4
australi	11.20	22.35	51.08	118.8	247.8	544.8
austria	11.43	23.09	50.62	119.4	253.2	560.4
belgium	11.41	23.04	52.00	120.0	248.4	532.8
bermuda	11.46	23.05	53.30	129.6	274.8	588.6
brazil	11.31	23.17	52.80	126.0	269.4	586.2
burma	12.14	24.47	55.00	130.8	267.0	570.6
canada	11.00	22.25	50.06	120.0	243.6	528.6
chile	12.00	24.52	54.90	123.0	253.8	562.2
china	11.95	24.41	54.97	124.8	259.8	558.6

Produce summaries of the variable m800, including

Numerical summaries: average, standard deviation, median and quartiles, maximum and minimum, interquartile difference

In [33]:
```
m800_info = data['m800'].describe() # data.describle() : data의 describtic info를 반환해줌
m800_info # type : Series
```

Out[33]:
```
count     55.000000
mean     124.581818
std        6.493447
min      113.400000
25%      120.000000
50%      123.000000
75%      129.000000
max      139.800000
Name: m800, dtype: float64
```

In [37]:
```
IQR = m800_info['75%'] - m800_info['25%']
m800_info['IQR'] = IQR # IQR을 추가적으로 계산해 더해줌
m800_info
```

Out[37]:
```
count     55.000000
mean     124.581818
std        6.493447
min      113.400000
25%      120.000000
50%      123.000000
75%      129.000000
max      139.800000
IQR        9.000000
Name: m800, dtype: float64
```

Graphical summaries: histogram and boxplot

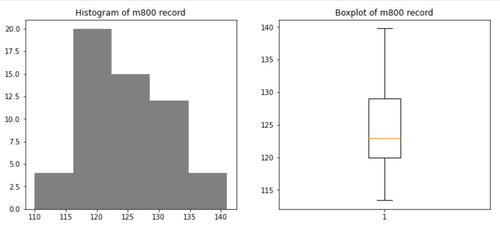
What can you observe about the variable distribution?

In [75]:
```
fig, (axs1, axs2) = plt.subplots(ncols=2, figsize=(12, 5))

axs1.hist(data['m800'], bins=5, range=(110, 141), color='gray')
axs1.set_title('Histogram of m800 record')

axs2.boxplot(data['m800'])
axs2.set_title('Boxplot of m800 record')

plt.show()
```

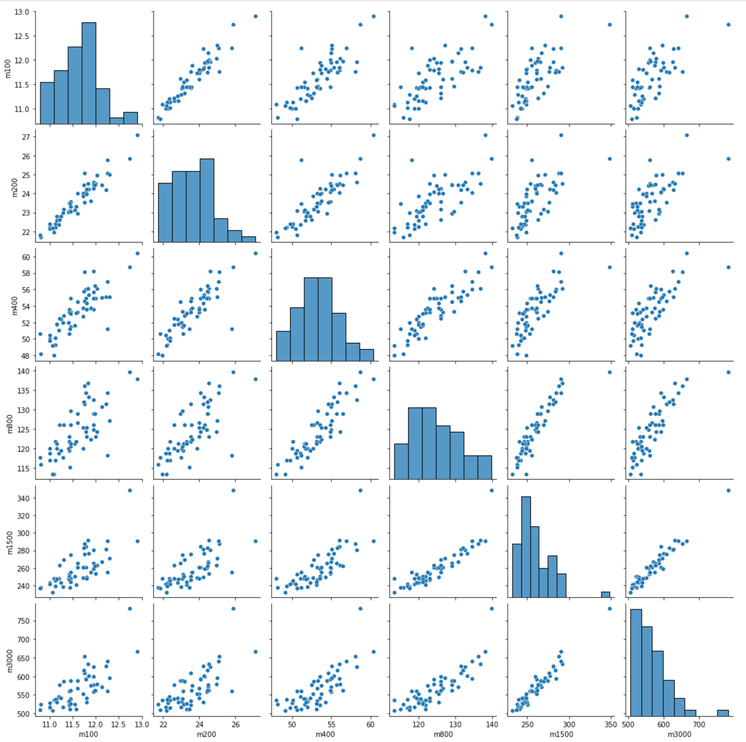


A. the data is right-skewed.

Produce scatter plot between all the variables(m100,m200,m400,m800,m1500,m3000).

What can you observe from the scatter plot? Are they correlated?

In [81]:
```
sns.pairplot(data)
plt.show()
```

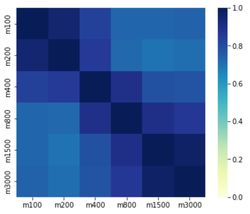


In [83]:
```
corr_df = data.corr() # correlation 분석
corr_df
```

Out[83]:

	m100	m200	m400	m800	m1500	m3000
m100	1.000000	0.952791	0.834692	0.727689	0.728371	0.741699
m200	0.952791	1.000000	0.856962	0.724060	0.698364	0.709871
m400	0.834692	0.856962	1.000000	0.898405	0.787842	0.777637
m800	0.727689	0.724060	0.898405	1.000000	0.901614	0.863565
m1500	0.728371	0.698364	0.787842	0.901614	1.000000	0.969169
m3000	0.741699	0.709871	0.777637	0.863565	0.969169	1.000000

In [99]:
```
plt.figure(figsize=(6.5,5))
sns.heatmap(corr_df, vmin=0.0, cmap='YlGnBu', square=True) # corrletaion 시각화, vmin은 범위의 최솟값을 지정
plt.show()
```



When x increases, also y increase. positive correlation.