Homework 1-1

生工所 李世耀 F08622011

1.

問題分析

以 2021 波士頓市長選舉第一輪投票結果，假設進入第二輪投票的兩位候選人支持率：

虛無假設 H_0：吳弭支持度 $p = 0.598$，A.E. George 支持度 $q = 0.402$。

替代假設 H_1：吳弭支持度 $p = 0.598 \pm 0.03$，A.E. George 支持度 $q = 0.402 \pm 0.03$。

第 A 小題

以民調隨機抽樣 150 人進行檢定，在雙尾檢定型一錯誤發生機率$\alpha/2 \leq 0.025$的條件下，計算拒絕 H_0 的的人數範圍，以及真實支持度為 $p = 0.598 \pm 0.03$ 發生型二錯誤的機率β。

民調抽樣 150 人可視為進行 150 次成功機率為 p (即吳弭支持度)的獨立伯努力試驗，

$$X_i \sim Bernoulli(p)$$

令 S 為 150 次試驗成功次數(即支持吳弭人數)，其服從二項式分布，當抽樣人數 n 足夠大時，依中央極限定理，S 的分布近似於平均值為 np、標準差為 $np(1-p)$的常態分布，則

$$S = \sum_{i=1}^{150} X_i \sim B(n,p) \overset{CLT}{\Longrightarrow} S \sim N(\mu = np, \sigma = npq)$$

吳弭支持度估計值$\hat{p}$為：

$$\hat{p} = \frac{S}{n} \sim N\left(\mu = p, \sigma = \sqrt{\frac{pq}{n}}\right)$$

第 B 小題

在真實支持率和假設的差異在$\pm 3\%$，且符合雙尾檢定型一錯誤發生機率$\alpha/2 \leq 0.025$、型二錯誤發生機率$\beta \leq 0.05$要求下，民調抽樣人數應為多少人，以及其拒絕虛無假設的人數範圍。

由第 A 小題的分析可知，當抽樣人數增加時，則支持度估計值$\hat{p}$服從的常態分布標準差變小，替代假設 $p = 0.598 \pm 0.03$ 對應的分布與虛無假設接受域重疊部分減少(下圖橘色與綠色區域的面積)，即型二錯誤發生機率(下圖中β_1與β_2)下降。

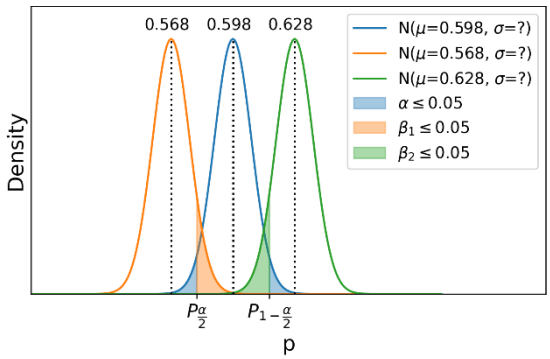

圖 1 型一錯誤與型二錯誤示意圖

第 C 小題

利用民意調查新論的近似法計算比較。

《民意調查新論》中提到當樣本數量大時，抽樣誤差為$\pm 2\sigma/\sqrt{n}$，若估計方式是採百分比，則樣本百分比的標準差會隨百分比變動但不超過$0.5/\sqrt{n}$，因此樣本百分比抽樣誤差可表示為：

$$\pm \frac{1}{\sqrt{n}} = \text{抽樣誤差}$$

分析方法

第 A 小題

虛無假設吳弭支持度 $p = 0.598$，抽樣人數 $n = 150$ 人，支持度估計值分布近似於：

$$N\left(\mu = p, \sigma = \sqrt{\frac{p(1-p)}{n}}\right)$$

其對應$\alpha/2$百分比與$1-\alpha/2$百分比的數值為：

$$p_{\alpha/2} = p + z_{\alpha/2}\sqrt{\frac{p(1-p)}{n}} \qquad p_{1-\alpha/2} = p + z_{1-\alpha/2}\sqrt{\frac{p(1-p)}{n}}$$

拒絕虛無假設 H_0 的人數範圍即可由抽樣人數乘上$p_{\alpha/2}$與$p_{1-\alpha/2}$獲得。或使用非參數化檢定，假設 150 次獨立伯努力試驗成功($\boldsymbol{A} = \{X = 1\}$)的機率為 p，統計值 T 為樣本中成功的次數(即支持吳弭的人數)，統計值 T 不通過檢定的上下界為：

$$T_L = np + z_{\alpha/2}\sqrt{np(1-p)}$$

$$T_U = np + z_{1-\alpha/2}\sqrt{np(1-p)}$$

而型二錯誤發生機率β即計算當真實支持率為 $p = 0.598 \pm 0.03$ 時，支持吳弭的人數 s 落在接受域範圍的機率，即：

$$P(T_L \le s \le T_U \mid p = 0.568) = \sum_{i=T_L}^{T_U} \binom{n}{i} 0.568^i (1-0.568)^{n-i}$$

$$P(T_L \le s \le T_U \mid p = 0.628) = \sum_{i=T_L}^{T_U} \binom{n}{i} 0.628^i (1-0.628)^{n-i}$$

第 B 小題

如圖 1，分別計算符合雙尾檢定型一錯誤發生機率$\alpha/2 \le 0.025$、型二錯誤發生機率$\beta_1, \beta_2 \le 0.05$對應的閾值：

$p = 0.598$，Type I error $\alpha/2 \le 0.025$

下界：$p_{\alpha/2} = p + z_{\alpha/2}\sqrt{\frac{p(1-p)}{n}}$

上界：$p_{1-\alpha/2} = p + z_{1-\alpha/2}\sqrt{\frac{p(1-p)}{n}}$

$p = 0.568$，Type II error $\beta_1 \le 0.05$：$p_{\beta_1} = p + z_{1-\beta_1}\sqrt{\frac{p(1-p)}{n}}$

$p = 0.628$，Type II error $\beta_2 \le 0.05$：$p_{\beta_2} = p + z_{\beta_2}\sqrt{\frac{p(1-p)}{n}}$

分別解$p_{\alpha/2} = p_{\beta_1}$、$p_{1-\alpha/2} = p_{\beta_2}$獲得對應抽樣人數，拒絕 H_0 的人數範圍則利用 A 小題方法。

第 C 小題

抽樣誤差設定為正負 3 個百分點，利用民調《民意調查新論》方法計算抽樣人數：

$$\pm\frac{1}{\sqrt{n}} = \pm 0.03$$

結果分析

第 A 小題

$$p_{0.025} = 0.598 - 1.96\sqrt{\frac{0.598\times0.402}{150}} = 0.5196 \quad n \times p_{0.025} = 77.94$$

$$p_{0.975} = 0.598 + 1.96\sqrt{\frac{0.598\times0.402}{150}} = 0.6764 \quad n \times p_{0.975} = 101.46$$

因此，拒絕 H_0 的人數範圍為 0~77 (<78)或 102~150 (>101)，利用非參數檢定方法可以獲得相同的結果：

$$T_L = 150 \times 0.598 - 1.96\sqrt{150 \times 0.598 \times 0.402} = 77.93$$
$$T_U = 150 \times 0.598 + 1.96\sqrt{150 \times 0.598 \times 0.402} = 101.47$$

而型二錯誤發生機率β分別為：

$$P(78 \le s \le 101 \mid p = 0.568) = \sum_{i=78}^{101} \binom{n}{i} 0.568^i (1-0.568)^{n-i} = 0.8942$$

$$P(78 \le s \le 101 \mid p = 0.628) = \sum_{i=78}^{101} \binom{n}{i} 0.628^i (1-0.628)^{n-i} = 0.8894$$

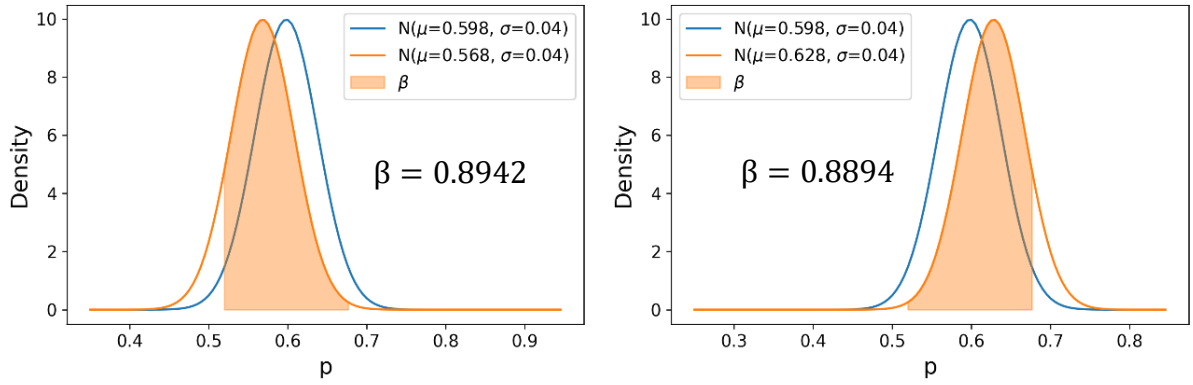

圖 2 真實支持度 $p = 0.568$ 與 $p = 0.628$ 第二型錯誤發生機率

第 B 小題

$$p_{\alpha/2} = p_{\beta_1} \Rightarrow 0.598 - 1.96\sqrt{\frac{0.598\times0.402}{n}} = 0.568 + 1.645\sqrt{\frac{0.568\times0.432}{n}} \Rightarrow n = 3505$$

$$p_{1-\alpha/2} = p_{\beta_2} \Rightarrow 0.598 + 1.96\sqrt{\frac{0.598\times0.402}{n}} = 0.628 - 1.645\sqrt{\frac{0.628\times0.372}{n}} \Rightarrow n = 3427$$

拒絕 H_0 的人數範圍分別為 0~2039 或 2153~3505($p = 0.568$)與 0~1993 或 2106~3427($p = 0.628$)。

第 C 小題

以民調《民意調查新論》方法計算抽樣人數，應抽樣人數為$n = 1112[(1/0.03)^2 = 1111.1]$，此時拒絕 H_0 的人數範圍為 0~632 或 698~1112，而型二錯誤發生機率為：

$$P(633 \le s \le 697 \mid p = 0.568) = \sum_{i=633}^{697} \binom{n}{i} 0.568^i (1-0.568)^{n-i} = 0.4792$$

$$P(633 \le s \le 697 \mid p = 0.628) = \sum_{i=633}^{697} \binom{n}{i} 0.628^i (1-0.628)^{n-i} = 0.4782$$

2.

問題分析

以氣象局臺北站 1961~2009 年 7 月日最高溫資料分析「氣候變遷」現象是否統計顯著。

虛無假設 H_0：49 年共 1519 筆資料來自<u>**相同**</u>分布，即「氣候變遷」現象<u>**無**</u>統計顯著。

替代假設 H_1：49 年共 1519 筆資料來自<u>**不同**</u>分布，即「氣候變遷」現象<u>**有**</u>統計顯著。

<u>第 A 小題</u>

以卡方檢定，顯著水準 $\alpha = 0.05$，判斷 1519 筆 7 月日最高溫資料是否通過常態分布的虛無假設。

卡方適合度檢定(Goodness of fit test)虛無假設為樣本來自於假設的機率分布，其檢定統計值為：

$$T = \sum_{i=1}^{k} \frac{(n_i - e_i)^2}{e_i}$$

其中 k 為一次隨機試驗可能結果的互斥類別(事件)個數，n_i 和 e_i 分別為在 n 次試驗中第 i 類內觀察到的次數與假設分布的期望次數，當 n 趨於無限大時，統計值 T 分布趨向自由度為 $k-1$ 的卡方分布 χ^2_{k-1} (若假設分布的參數需由樣本估計，則自由度為 $k-m-1$，m 為參數數量)。當統計檢定值 $T > \chi^2_{1-\alpha,k-1}$，則拒絕虛無假設。

<u>第 B 小題</u>

排序並找出 1961~2000 年 7 月日最高溫低門檻值與高門檻值(前後 2.5%)，計算 2001~2009 年 7 月日最高溫小於低門檻值與大於高門檻值的百分比，是否能判斷 7 月日最高溫紀錄存在氣候變遷。

若前 30 年與後 9 年資料來自相同分布(虛無假設)，則兩者落在低門檻值以下和高門檻值以上的資料數量百分比會接近 2.5%。若存在氣候變遷(以溫度升高說明)，則後 9 年資料小於低門檻的百分比會下降(< 2.5%)而大於高門檻的百分比會上升(> 2.5%)，即後 9 年資料往高溫偏移，前 30 年與後 9 年資料來自不同分布。兩個百分比只能說明可能存在氣候變遷，然而是否具有統計顯著則需要進一步討論。

<u>第 C 小題</u>

利用蒙地卡羅法模擬 10,000 組、每組 1519 筆彼此互相獨立的標準常態分布資料，依循第 B 小題計算 10,000 筆小於低門檻值與大於高門檻值的百分比。

為檢定第 B 小題獲得的百分比是否顯著偏離 2.5%，假設資料符合常態分布，利用蒙地卡羅法產生 10,000 組、每組 1519 筆來自相同分布(虛無假設)的資料，將每組資料分為前 1240 筆和後 279 筆，找出前 1240 筆低門檻值與高門檻值並計算後 279 筆小於低門檻值和大於高門檻值的百分比，兩個百分比具有隨機性為隨機變數，透過上述的方法可以產生兩個百分比的 10,000 筆樣本值，兩個百分比樣本平均值應會接近 2.5%。

<u>第 D 小題</u>

利用第 B 小題與第 C 小題的結果判斷，能不能說明「7 月日最高溫紀錄是否存在氣候變遷」。

第 C 小題產生的 10,000 筆百分比樣本可以分別找出「小於低門檻值的百分比」對應 $\alpha/2 = 0.025$ 的百分比門檻值($LB_{L,M}$, $UB_{L,M}$)與「大於高門檻值的百分比」對應 $\alpha/2 = 0.025$ 的百分比門檻值($LB_{U,M}$, $UB_{U,M}$)，和第 B 小題得到的百分比(LB_O, UB_O)進行比較：

<u>虛無假設 H_0：1519 筆資料來自相同分布(P)且呈常態分布(Q)，即氣候變遷現象無統計顯著。</u>

1. 若 $LB_{L,M} \le LB_O \le UB_{L,M}$ 且 $LB_{U,M} \le UB_O \le UB_{U,M}$，則不拒絕虛無假設，即說明「氣候變遷」現象無統計顯著。
2. 若 $LB_{L,M} > LB_O$ 或 $UB_{L,M} < LB_O$ 或 $LB_{U,M} > UB_O$ 或 $UB_{U,M} < UB_O$，則拒絕虛無假設，但因第 A 小題拒絕常態分布的虛無假設，$P \cap Q \rightarrow R$, $\sim R \rightarrow \sim P \cup \sim Q$，故也無法說明氣候變遷現象是否統計顯著筆資料來自不同分布(~P)。

分析方法

第 A 小題

進行卡方檢定前需將資料分類為互斥事件，分類方式分可為等間距方法或等機率方法，為避免部分類別中樣本數過少，因此採取等機率分類方法(Romanowski et al., 1978)[1]。類別個數 k 則由下式決定(D'Agostino, 1986)[2]，詳細程式碼與類別內樣本數請參見附錄。

$$k = 2n^{2/5}$$

因為常態分布的參數未知，卡方檢定自由度為 $k-m-1$，m＝2 (平均值、標準差)。

第 B 小題

依題意利用 `numpy.quantile()`找到 1961~2000 年 1240 筆資料的低門檻值與高門檻值，並計算 2001~2009 年 279 筆資料小於低門檻值或大於高門檻值的百分比。

第 C 小題

利用 `scipy.stat.norm.rvs()`產生樣本大小為(10000, 1519)彼此互相獨立的標準常態分布資料，依題意計算 10,000 筆「小於低門檻值的百分比」與「大於高門檻值的百分比」的樣本值，且找到兩個百分比各自的低門檻值與高門檻值(前後 2.5%，一樣利用 `numpy.quantile()`)。

第 D 小題

利用第 B 小題和第 C 小題結果依問題分析說明進行比較。

第 E 小題

將常態分布虛無假設改為皮爾森第三型分布假設，重複第 A 小題至第 D 小題。

結果分析

第 A 小題

原始資料經前處理取出歷年 7 月日最高溫資料，資料基本統計值如下**圖 3**。資料依等機率方法分為 37 組繪製直方圖如圖 4。

```
No. of obs:  1519
 Min & Max:  (24.1, 38.0)
      Mean: 33.6221
  Variance:  3.7008
  Skewness: -1.0385
  Kurtosis:  5.0056
```

圖 3 七月日最高溫資料(1519 筆)統計值

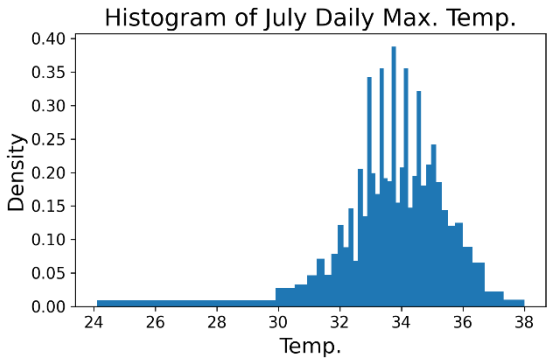

圖 4 七月日最高溫資料直方圖

卡方檢定測試統計值T為 290.079，而自由度 $k-m-1=34$ 的卡方檢定門檻值$\chi^2_{0.95,34}$為 48.602，因 $T > \chi^2_{0.95,34}$，拒絕常態分布的虛無假設。

---

[1] Romanowski, M., McConnell, R. K., &Halpenny, J. F. (1978). The equiprobable interval method of sample classification and the validity of the χ2 test. Metrologia, 14(4), 185–187.

[2] D'Agostino, R. B. (1986). Goodness-of-fit-techniques (Vol. 68). CRC press.

第 B 小題

1961~2000 年 7 月日最高溫低門檻值與高門檻值分別為 **28.4℃**與 **36.6℃**，2001~2009 年 279 筆資料小於低門檻值與大於高門檻值的百分比分別為 **0.717%**、**3.584%**，由這兩個百分比(<2.5%與>2.5%)乍看之下 7 月日最高溫可能存在氣候變遷，但是否有統計顯著並無法說明，需透過第 C 小題更進一步討論。

第 C 小題

10,000 筆「小於低門檻值的百分比」與「大於高門檻值的百分比」的直方圖如圖 5，兩者平均值都接近 2.5%，高低門檻值皆為[0.717%, 4.659%]。

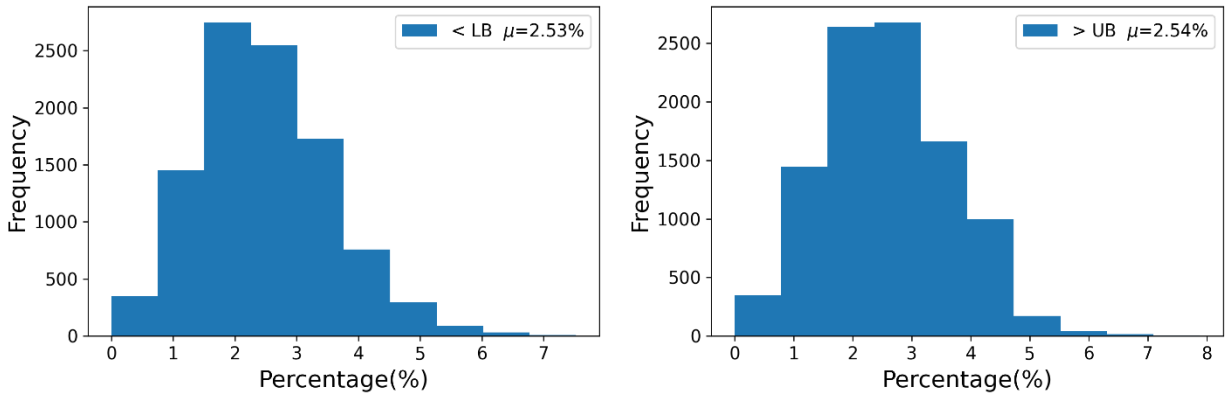

圖 5 「<低門檻值百分比」與「>高門檻值百分比」樣本直方圖

第 D 小題

由第 B 小題與第 C 小題結果：

$$LB_{L,M} \le LB_O(0.717\%) \le UB_{L,M} 且 LB_{U,M} \le UB_O(3.584\%) \le UB_{U,M}$$

因此不拒絕虛無假設，即說明「氣候變遷」現象無統計顯著。

第 E 小題

假設 1961~2000 年 7 月日最高溫 1240 筆資料呈皮爾森第三型分布進行卡方檢定，一樣使用等機率分類方法將 1240 筆資料分為 37 組(自由度為 $k-m-1=33$)，測試統計值T為 246.674，卡方檢定的門檻值$\chi^2_{0.95,33}$為 47.4，因$T > \chi^2_{0.95,33}$，拒絕皮爾森第三型分布的虛無假設。

利用 `scipy.stat.pearson.rvs()`產生樣本大小為(10000, 1519)彼此互相獨立且參數值與 1240 筆相同的皮爾森第三型分布分佈資料，計算 10,000 筆「小於低門檻值的百分比」與「大於高門檻值的百分比」的直方圖如圖 6，兩者平均值都接近 2.5%，高低門檻值與第 C 小題相同皆為[0.717%, 4.659%]，因此與第 D 小題結論相同。

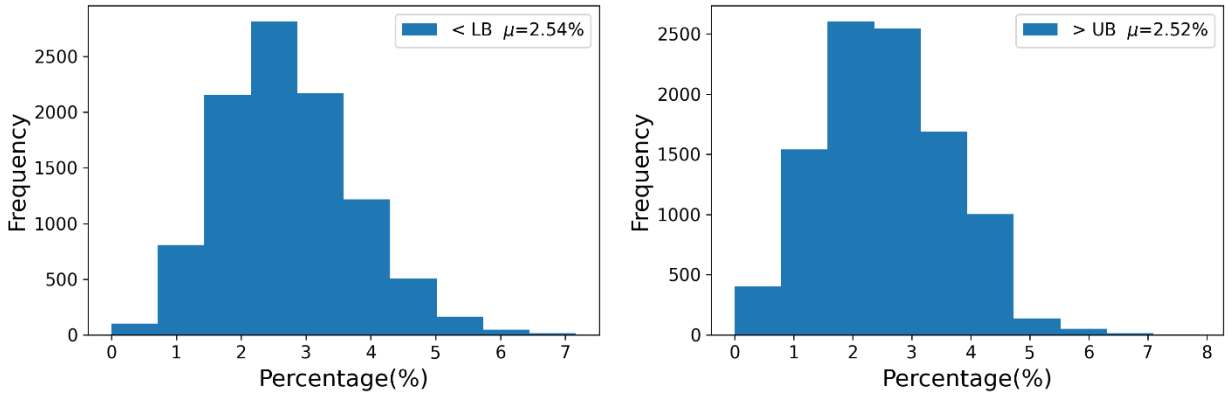

圖 6 「<低門檻值百分比」與「>高門檻值百分比」樣本直方圖 (Pearson Type III)

Homework 1-1 附錄

November 4, 2021

[1]:
```
import numpy as np
import pandas as pd
import scipy.stats as ss
```

1 第一題

1.1 第 A 小題

$$P(78 \leq s \leq 101|p = 0.568) = \sum_{i=78}^{101} \binom{N}{i} 0.568^i (1 - 0.568)^{n-i}$$

$$P(78 \leq s \leq 101|p = 0.628) = \sum_{i=78}^{101} \binom{N}{i} 0.628^i (1 - 0.628)^{n-i}$$

[2]:
```
print(ss.binom.cdf(101, 150, 0.568) - ss.binom.cdf(77, 150, 0.568))
print(ss.binom.cdf(101, 150, 0.628) - ss.binom.cdf(77, 150, 0.628))
```

```
0.8942340977568481
0.8893559102960302
```

1.2 第 C 小題

$$P(633 \leq s \leq 697|p = 0.568) = \sum_{i=633}^{697} \binom{N}{i} 0.568^i (1 - 0.568)^{n-i}$$

$$P(633 \leq s \leq 697|p = 0.628) = \sum_{i=633}^{697} \binom{N}{i} 0.628^i (1 - 0.628)^{n-i}$$

[3]:
```
print(ss.binom.cdf(697, 1112, 0.568) - ss.binom.cdf(632, 1112, 0.568))
print(ss.binom.cdf(697, 1112, 0.628) - ss.binom.cdf(632, 1112, 0.628))
```

```
0.4791774849801923
0.4782405364695516
```

2 第二題

2.1 第 A 小題

2.1.1 資料前處理

[4]:
```
raw = pd.read_csv('466920chkd.txt', sep='\t')
raw.index = pd.date_range('1961-01-01 00', '2009-12-31 23', freq='H')
raw.columns = ['stn', 'Time', 'TX01']

July = raw[raw.index.month==7]['TX01'].values.reshape(49,31*24).T
idx1 = pd.date_range('1961-07-01 00', '1961-07-31 23', freq='D').strftime('%m-%d')
idx2 = pd.date_range('1961-07-01 00', '1961-07-01 23', freq='H').strftime('%H')
midx = pd.MultiIndex.from_product([idx1, idx2], names=['Date', 'Hour'])
col = np.arange(1961, 2010, 1).astype(str)
JulyTemp = pd.DataFrame(July, index=midx, columns=col)
JulyDMax = JulyTemp.groupby(level='Date').max()
```

[5]:
```
print('No. of obs: ', ss.describe(JulyDMax.values.flatten()).nobs)
print(' Min & Max: ', ss.describe(JulyDMax.values.flatten()).minmax)
print('      Mean: {:7.4f}'.format(ss.describe(JulyDMax.values.flatten()).mean))
print('  Variance: {:7.4f}'.format(ss.describe(JulyDMax.values.flatten()).variance))
print('  Skewness: {:7.4f}'.format(ss.describe(JulyDMax.values.flatten()).skewness))
print('  Kurtosis: {:7.4f}'.format(ss.describe(JulyDMax.values.flatten()).kurtosis+3))
```

```
No. of obs:  1519
 Min & Max:  (24.1, 38.0)
      Mean: 33.6221
  Variance:  3.7008
  Skewness: -1.0385
  Kurtosis:  5.0056
```

2.1.2 資料直方圖等機率區間與個數

[6]:
```
mean = ss.describe(JulyDMax.values.flatten()).mean
std = ss.describe(JulyDMax.values.flatten()).variance**0.5
equiprob = []
for i in range(0,38):
    if i == 0:
        equiprob.append(ss.describe(JulyDMax.values.flatten()).minmax[0])
    elif i == 37:
        equiprob.append(ss.describe(JulyDMax.values.flatten()).minmax[1])
    else:
        equiprob.append(ss.norm.ppf(i/37, loc=mean, scale=std))
hist, bin_edges = np.histogram(JulyDMax.values.flatten(), bins=equiprob)
print('{:^10}{:^6}   {:^14}{:^1}'.format('區間','個數','區間','個數'))
for i in range(19):
    if i < 18:
        print('{0:5.2f}~{1:5.2f}: {2:4d}   |   {3:5.2f}~{4:5.2f}: {5:4d}'\
              .format(bin_edges[i],bin_edges[i+1],hist[i],
                      bin_edges[i+19],bin_edges[i+20],hist[i+19]))
    else:
        print('{0:5.2f}~{1:5.2f}: {2:4d}'\
              .format(bin_edges[i],bin_edges[i+1],hist[i]))
```

```
    區間       個數          區間       個數
24.10~29.92:   82   |   33.69~33.82:   77
29.92~30.53:   26   |   33.82~33.95:   31
30.53~30.93:   20   |   33.95~34.08:   42
30.93~31.24:   22   |   34.08~34.22:   73
31.24~31.50:   28   |   34.22~34.36:   31
31.50~31.73:   16   |   34.36~34.50:   42
31.73~31.93:   24   |   34.50~34.65:   72
31.93~32.11:   34   |   34.65~34.80:   42
32.11~32.28:   23   |   34.80~34.96:   52
32.28~32.44:   36   |   34.96~35.13:   63
32.44~32.60:   16   |   35.13~35.32:   52
32.60~32.75:   46   |   35.32~35.52:   44
32.75~32.89:   29   |   35.52~35.74:   41
32.89~33.03:   72   |   35.74~36.00:   49
33.03~33.16:   41   |   36.00~36.31:   42
33.16~33.29:   34   |   36.31~36.71:   40
33.29~33.43:   71   |   36.71~37.33:   21
33.43~33.56:   38   |   37.33~38.00:   10
33.56~33.69:   37
```

2.1.3 常態分布相同區間內預期個數

[7]:
```
nobs = ss.describe(JulyDMax.values.flatten()).nobs
expected = []
for i in range(37):
    lower = bin_edges[i]
    upper = bin_edges[i+1]
    prob = ss.norm.cdf(upper, mean, std) - ss.norm.cdf(lower, mean, std)
    expected.append(np.round(nobs * prob, 1))

print('{:^10}{:^6}   {:^14}{:^1}'.format('區間','個數','區間','個數'))
for i in range(19):
    if i < 18:
        print('{0:5.2f}~{1:5.2f}: {2:4.0f}   |   {3:5.2f}~{4:5.2f}: {5:4.0f}'\
              .format(bin_edges[i],bin_edges[i+1],expected[i],
                      bin_edges[i+19],bin_edges[i+20],expected[i+19]))
    else:
        print('{0:5.2f}~{1:5.2f}: {2:4.0f}'\
              .format(bin_edges[i],bin_edges[i+1],expected[i]))
```

```
    區間       個數          區間       個數
24.10~29.92:   41   |   33.69~33.82:   41
29.92~30.53:   41   |   33.82~33.95:   41
30.53~30.93:   41   |   33.95~34.08:   41
30.93~31.24:   41   |   34.08~34.22:   41
31.24~31.50:   41   |   34.22~34.36:   41
31.50~31.73:   41   |   34.36~34.50:   41
31.73~31.93:   41   |   34.50~34.65:   41
31.93~32.11:   41   |   34.65~34.80:   41
32.11~32.28:   41   |   34.80~34.96:   41
32.28~32.44:   41   |   34.96~35.13:   41
32.44~32.60:   41   |   35.13~35.32:   41
32.60~32.75:   41   |   35.32~35.52:   41
32.75~32.89:   41   |   35.52~35.74:   41
32.89~33.03:   41   |   35.74~36.00:   41
33.03~33.16:   41   |   36.00~36.31:   41
33.16~33.29:   41   |   36.31~36.71:   41
33.29~33.43:   41   |   36.71~37.33:   41
33.43~33.56:   41   |   37.33~38.00:   24
33.56~33.69:   41
```

2.1.4 Chi-square Test

[8]:
```
statistic, pvalue = ss.chisquare(hist, expected, ddof=2) # df = k-ddof-1
print('Chi-square Test Statistic: {:5.3f}'.format(statistic))
chi_square = ss.chi2.ppf(0.95, df=37-3)
print('Critical Value: {:5.3f}'.format(chi_square))
```

```
Chi-square Test Statistic: 290.079
Critical Value: 48.602
```

Test Statistic $>$ Critical Value $\Rightarrow$ Reject Null Hypothesis

2.2 第 B 小題

[9]:
```
LB = np.quantile(JulyDMax.loc[:,'1961':'2000'], 0.025, interpolation='midpoint')
UB = np.quantile(JulyDMax.loc[:,'1961':'2000'], 0.975, interpolation='midpoint')
lsLB = np.sum(JulyDMax.loc[:,'2001':'2009'].values.flatten() < LB)
grUB = np.sum(JulyDMax.loc[:,'2001':'2009'].values.flatten() > UB)
print('Lower Bound: {:4.1f}'.format(LB))
print('Upper Bound: {:4.1f}'.format(UB))
print('   Less than Lower Bound: {0:2d} ({1:.3f}%)'.format(lsLB, lsLB/279*100))
print('Greater than Upper Bound: {0:2d} ({1:.3f}%)'.format(grUB, grUB/279*100))
```

```
Lower Bound: 28.4
Upper Bound: 36.6
   Less than Lower Bound:  2 (0.717%)
Greater than Upper Bound: 10 (3.584%)
```

2.3 第 C 小題

[10]:
```
sim = ss.norm.rvs(loc=0, scale=1, size=(10000, 1519))
lsList = []
grList = []
for i in range(10000):
    lb = np.quantile(sim[i, :1240], 0.025, interpolation='midpoint')
    ub = np.quantile(sim[i, :1240], 0.975, interpolation='midpoint')
    lsList.append(np.sum(sim[i, 1240:] < lb)/279*100)
    grList.append(np.sum(sim[i, 1240:] > ub)/279*100)

lsLLB = np.quantile(lsList, 0.025, interpolation='midpoint')
lsLUB = np.quantile(lsList, 0.975, interpolation='midpoint')
grLLB = np.quantile(grList, 0.025, interpolation='midpoint')
grLUB = np.quantile(grList, 0.975, interpolation='midpoint')
print('小於 LB 的百分比門檻值： [{0:.3f}%, {1:.3f}%]'.format(lsLLB, lsLUB))
print('大於 UB 的百分比門檻值： [{0:.3f}%, {1:.3f}%]'.format(grLLB, grLUB))
```

```
小於 LB 的百分比門檻值： [0.717%, 4.659%]
大於 UB 的百分比門檻值： [0.717%, 4.659%]
```

2.4 第 E 小題

[11]:
```
data = JulyDMax.loc[:,'1961':'2000'].values.flatten()
print('No. of obs: ', ss.describe(data).nobs)
print(' Min & Max: ', ss.describe(data).minmax)
print('      Mean: {:7.4f}'.format(ss.describe(data).mean))
print('  Variance: {:7.4f}'.format(ss.describe(data).variance))
print('  Skewness: {:7.4f}'.format(ss.describe(data).skewness))
```

```
No. of obs:  1240
 Min & Max:  (24.1, 37.8)
      Mean: 33.5122
  Variance:  3.7085
  Skewness: -1.0765
```

[12]:
```
mean = ss.describe(data).mean
std  = ss.describe(data).variance**0.5
skew = ss.describe(data).skewness
equiprob = []
for i in range(0,38):
    if i == 0:
        equiprob.append(ss.describe(data).minmax[0])
    elif i == 37:
        equiprob.append(ss.describe(data).minmax[1])
    else:
        equiprob.append(ss.pearson3.ppf((37-i)/37, skew, mean, std))
hist, bin_edges = np.histogram(data, bins=equiprob)
print('{:^10}{:^6}   {:^14}{:^1}'.format('區間','個數','區間','個數'))
for i in range(19):
    if i < 18:
        print('{0:5.2f}~{1:5.2f}: {2:4d}   |   {3:5.2f}~{4:5.2f}: {5:4d}'\
              .format(bin_edges[i],bin_edges[i+1],hist[i],
                      bin_edges[i+19],bin_edges[i+20],hist[i+19]))
    else:
        print('{0:5.2f}~{1:5.2f}: {2:4d}'\
              .format(bin_edges[i],bin_edges[i+1],hist[i]))
```

```
    區間       個數          區間       個數
24.10~28.98:   41   |   33.91~34.03:   38
28.98~29.99:   29   |   34.03~34.15:   24
29.99~30.60:   28   |   34.15~34.27:   31
30.60~31.05:   21   |   34.27~34.38:   23
31.05~31.41:   23   |   34.38~34.50:   36
31.41~31.71:   24   |   34.50~34.61:   60
31.71~31.98:   20   |   34.61~34.72:   31
31.98~32.21:   48   |   34.72~34.84:   27
32.21~32.42:   34   |   34.84~34.95:   16
32.42~32.61:   38   |   34.95~35.07:   25
32.61~32.79:   17   |   35.07~35.20:   20
32.79~32.95:   51   |   35.20~35.33:   39
32.95~33.11:   77   |   35.33~35.46:   19
33.11~33.26:   29   |   35.46~35.61:   40
33.26~33.40:   28   |   35.61~35.77:   13
33.40~33.53:   62   |   35.77~35.96:   23
33.53~33.66:   34   |   35.96~36.21:   23
33.66~33.79:   26   |   36.21~37.80:   62
33.79~33.91:   60
```

[13]:
```
nobs = ss.describe(data).nobs
expected = []
for i in range(37):
    lower = bin_edges[i]
    upper = bin_edges[i+1]
    prob = ss.pearson3.cdf(upper,skew,mean,std) - ss.pearson3.cdf(lower,skew,mean,std)
    expected.append(np.round(nobs * prob, 1))

print('{:^10}{:^6}   {:^14}{:^1}'.format('區間','個數','區間','個數'))
for i in range(19):
    if i < 18:
        print('{0:5.2f}~{1:5.2f}: {2:4.0f}   |   {3:5.2f}~{4:5.2f}: {5:4.0f}'\
              .format(bin_edges[i],bin_edges[i+1],expected[i],
                      bin_edges[i+19],bin_edges[i+20],expected[i+19]))
    else:
        print('{0:5.2f}~{1:5.2f}: {2:4.0f}'\
              .format(bin_edges[i],bin_edges[i+1],expected[i]))
```

```
    區間       個數            區間       個數
24.10~28.98:   33   |   33.91~34.03:   34
28.98~29.99:   34   |   34.03~34.15:   34
29.99~30.60:   34   |   34.15~34.27:   34
30.60~31.05:   34   |   34.27~34.38:   34
31.05~31.41:   34   |   34.38~34.50:   34
31.41~31.71:   34   |   34.50~34.61:   34
31.71~31.98:   34   |   34.61~34.72:   34
31.98~32.21:   34   |   34.72~34.84:   34
32.21~32.42:   34   |   34.84~34.95:   34
32.42~32.61:   34   |   34.95~35.07:   34
32.61~32.79:   34   |   35.07~35.20:   34
32.79~32.95:   34   |   35.20~35.33:   34
32.95~33.11:   34   |   35.33~35.46:   34
33.11~33.26:   34   |   35.46~35.61:   34
33.26~33.40:   34   |   35.61~35.77:   34
33.40~33.53:   34   |   35.77~35.96:   34
33.53~33.66:   34   |   35.96~36.21:   34
33.66~33.79:   34   |   36.21~37.80:   34
33.79~33.91:   34
```

2.4.1 Chi-square Test

[14]:
```
statistic, pvalue = ss.chisquare(hist, expected, ddof=3) # df = k-ddof-1
print('Chi-square Test Statistic: {:5.3f}'.format(statistic))
chi_square = ss.chi2.ppf(0.95, df=37-4)
print('Critical Value: {:5.3f}'.format(chi_square))
```

```
Chi-square Test Statistic: 246.674
Critical Value: 47.400
```

[15]:
```
sim = ss.pearson3.rvs(skew, mean, std, size=(10000, 1519))
lsList = []
grList = []
for i in range(10000):
    lb = np.quantile(sim[i, :1240], 0.025, interpolation='midpoint')
    ub = np.quantile(sim[i, :1240], 0.975, interpolation='midpoint')
    lsList.append(np.sum(sim[i, 1240:] < lb)/279*100)
    grList.append(np.sum(sim[i, 1240:] > ub)/279*100)
```

```
lsLLB = np.quantile(lsList, 0.025, interpolation='midpoint')
lsLUB = np.quantile(lsList, 0.975, interpolation='midpoint')
grLLB = np.quantile(grList, 0.025, interpolation='midpoint')
grLUB = np.quantile(grList, 0.975, interpolation='midpoint')
print('小於 LB 的百分比門檻值： [{0:.3f}%, {1:.3f}%]'.format(lsLLB, lsLUB))
print('大於 UB 的百分比門檻值： [{0:.3f}%, {1:.3f}%]'.format(grLLB, grLUB))
```

```
小於 LB 的百分比門檻值： [0.717%, 4.659%]
大於 UB 的百分比門檻值： [0.717%, 4.659%]
```