

實驗物理學（二）
實驗報告

Fundamental Python
Chi-square fitting - 3

Group 2

洪 瑜 B125090009

黃巧涵 B122030003

洪懌平 B102030019

2025/05/13

摘要

本次實驗著重於了解卡方擬合 (Chi-square fitting) 與統計檢定中 p-value 的應用與意義。透過兩個練習 (Practice 1 與 Practice 2)，我們分別以自行設定的二次函數產生含隨機誤差的模擬數據，進行擬合分析，計算對應的卡方值與 p-value，並探討其與模型合理性的關係。我們自行撰寫 p-value 計算器，並比較實驗所得的 p-value 分布與理論預測。進一步，我們利用顯著水準 (significance level) 判斷擬合是否合理，並對 p-value 與模型參數準確性之關聯進行實證討論。結果顯示，p-value 雖可反映擬合結果是否顯著偏離理論模型，但不應作為評估參數準確性的唯一指標。本實驗強調統計方法與視覺化工具在模型擬合與驗證中的重要性，對於日後進行更複雜的資料分析與科學建模具有實質助益。

1 前言

在本周的課程內容，我們會從practice 1練習到p-value的意義和計算，以及其和 χ^2 的關係；而在practice 2，則會著重在利用p-value去探討其與擬合結果好壞的關聯，並認識甚麼是significance level。

1.1 P-value

在了解 P -value 前，需要先知道一些先備知識。

- 虛無假設(null hypothesis) H_0 ：對某件事情預設為某種結果或狀態。
- 對立假說(alternative hypothesis) H_1 ：為虛無假設的對立面。

	H_0 為真	H_1 為真 (H_0 錯誤)
拒絕 H_0	錯誤判斷 (第I型錯誤) 發生機率 = α	正確判斷 發生機率 = $1 - \beta$
接受 H_0	正確判斷 發生機率 = $1 - \alpha$	錯誤判斷 (第II型錯誤) 發生機率 = β

Table 1: 假設檢定中的判斷與錯誤類型

接著來介紹甚麼是p-value，並且要怎麼透過觀察這個數值來判斷擬合的結果是否合理。

- **p-value之定義**：在虛無假設的情況下，發生此統計結果、或更極端的結果之機率；所以p-value越小表示 H_0 為真時越不可能發生，即可推論 H_0 假設錯誤。
 1. P-value的P是Probability的P，也就是描述事件發生的可能性。
 2. 計算p-value的前提是虛無假設為真。

我們使用 *desmos* 繪圖作為範例，用圖像化的方式更清楚的解釋 p -value：

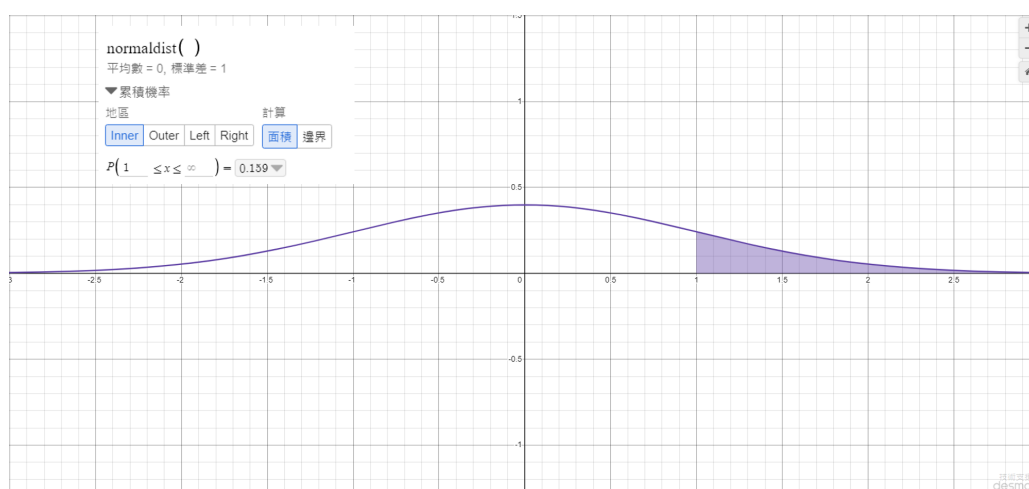


Figure 1: 利用常態分佈解釋 p -value

Fig.1 為一平均 $\mu = 0$ 、標準差 $\sigma = 1$ 之常態分佈的圖形，我們標示出在 $1 \leq x \leq \infty$ 這個範圍下，常態分佈下的面積，而其代表的意義是「觀察值大於等於 1 的機率」，在圖的左上角，可看見 $P = 0.159$ ，說明在虛無假設成立的情況下，發生此事件的機率為 0.159。

• p -value 與 χ^2 的關係

在這裡，一樣使用 *desmos* 圖像化解釋：

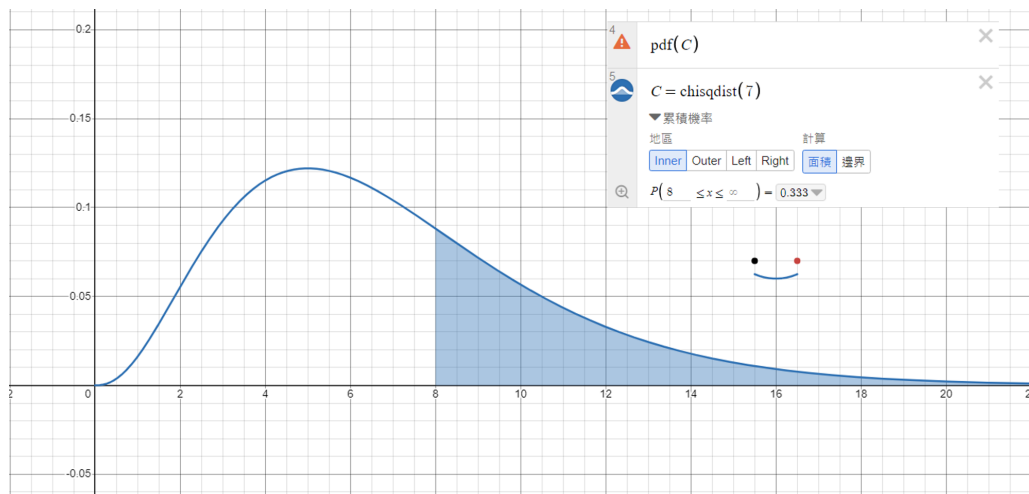


Figure 2: 利用 $\text{PDF}(dof = 7)$ 解釋 p -value 與 χ^2 的關係

Fig.2 模擬一個 $dof = 7$ 時的 PDF，而 x 軸為 χ^2 值。假設我們實驗所得之 $\chi^2 = 8$ ，那發生此事件的機率即為 $8 \leq \chi^2$ 這個區間在 PDF 下的面積（也就是圖上所標示的藍色區塊），而這個機率就是 p -value（假設 H_0 為真），由圖的右上角可知： p -value = 0.333。

note: p -value 不是 H_0 為真的機率，而是「 H_0 為真的前提下」，發生目前的統計結果或更極端的情況之機率。

1.2 significance level

significance level是拒絕 H_0 、但其實 H_0 為真的機率；即為Tab.1上寫的「第I型錯誤」，發生機率為 α 。可以把significance level想像成p-value的門檻，如果 $p < \alpha$ 就拒絕 H_0 （放在本實驗的情況就是不接受這個擬合結果）。另一個相關的參數是信賴水準 $\gamma = 1 - \alpha$ ，指的是我們對 H_0 為真時，建立的統計模型有多大的信心，例如假設現在信賴水準是95%，表示如果重複做很多次實驗，有95%的機率算出來的 χ^2 會落在此範圍內；從另一個角度來看就是有5%的機率我們會拒絕這個虛無假設，所以p-value會小於0.05。

在本次的實驗中，我們設定significance level為上下各5%；分別排除欠擬合（ χ^2 過大）與過擬合（ χ^2 過小）這兩種情況，因此得到的 χ^2 要落在這兩個區間內，這個模型才是可被接受的。

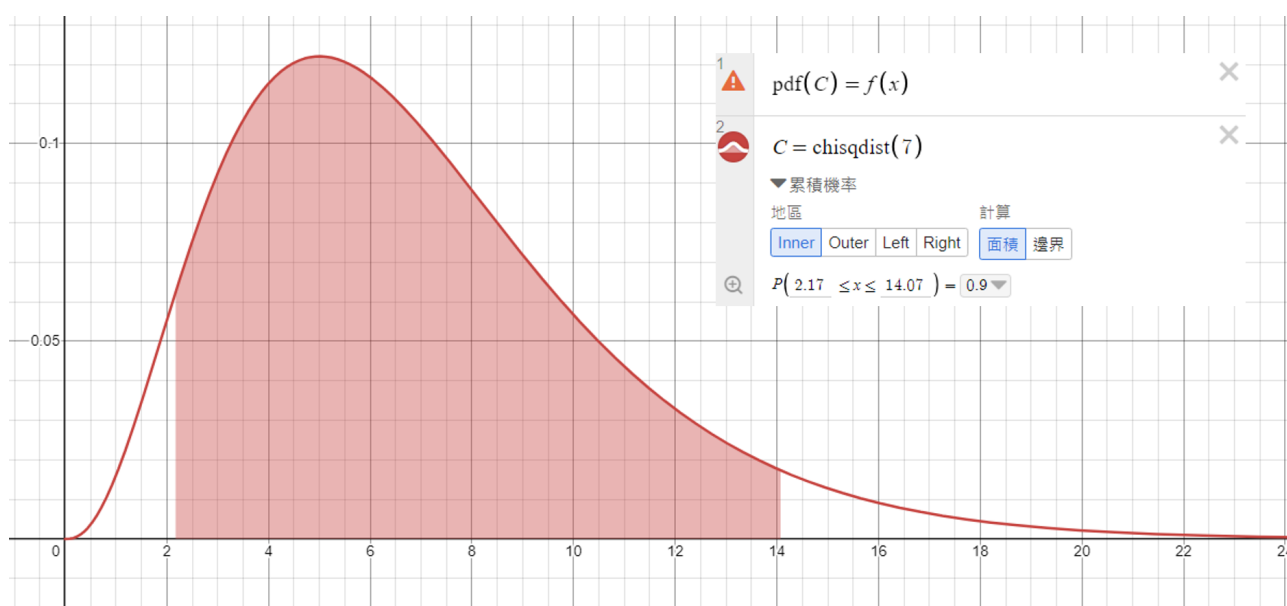


Figure 3: 紅色面積標示出 $5\% < p - \text{value} < 95\%$ 的範圍，實驗所得之 χ^2 需落在此範圍方可接受

2 實驗步驟

2.1 Practice 1

1. By using previous knowledge, choose your own function to generate data points.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from scipy.stats import chi2
```

✓ 0.0s Python

```
np.random.seed(42)
```

✓ 0.0s Python

2. Generate 1000 sets of data points with the same function with normally distributed random noise.

```
def parabolic(x, a, b, c):
    return a * x**2 + b * x + c
```

✓ 0.0s Python

```
# Generate synthetic data
true_a = 2.0
true_b = 1.0
true_c = 0.5

sigma0 = 3.0

# Generate data
n_data = 10
n_sets = 1000

x = np.linspace(-5, 5, n_data)
```

✓ 0.0s Python

3. Fit these data points with the function you used to generate the data points.
4. Calculate the χ^2 values of each fitting result.

```

all_data = []
fit_results = []
chi2_values = []

✓ 0.0s Python

for i in range(n_sets):
    # Generate synthetic data with noise
    y = parabolic(x, true_a, true_b, true_c) + np.random.normal(0, sigma0, size=x.shape)
    all_data.append(y)
    # Fit the data
    popt, pcov = curve_fit(parabolic, x, y)

    # Calculate chi-squared
    residuals = y - parabolic(x, *popt)
    chi2_value = np.sum((residuals / sigma0)**2)

    fit_results.append(popt)
    chi2_values.append(chi2_value)

✓ 0.1s Python

```

5. Create your own **p-value** calculator to calculate the **p-value** of each χ^2 instead of using functions in `scipy.stats`.

```

from math import gamma

def chi2_p_value(chi2_value, dof):
    """
    Calculate the p-value for a given chi-square value and degrees of freedom (dof).
    """
    # Compute the incomplete gamma function (upper tail)
    incomplete_gamma = gamma(dof / 2) - sum((chi2_value / 2)**k / gamma(k + 1) for k in range(dof // 2))

    # Compute the p-value
    p_value = incomplete_gamma / gamma(dof / 2)
    return p_value

# Example usage
dof = 7 # degrees of freedom
p_values = [chi2_p_value(chi2, dof) for chi2 in chi2_values]
print(p_values[:10]) # Print the first 10 p-values

✓ 0.0s Python

```

6. Plot the histogram of χ^2 .

```

# Display the results
plt.figure(figsize=(10, 6))

x_plot = np.linspace(-8, 8, 500)
plt.plot(x_plot, parabolic(x_plot, true_a, true_b, true_c), color='C4', label='True model',
         linewidth=4, linestyle='-')

sampled_sets = np.random.choice(n_sets, 200, replace=False)
for i in sampled_sets:
    plt.scatter(x, all_data[i, :], color='C1', alpha=0.3)
    plt.plot(x_plot, parabolic(x_plot, *fit_results[i]), color='plum', alpha=0.1,
            linewidth=1.5, linestyle=':')

plt.scatter([], [], color='C1', alpha=0.3, label='Sample points')
plt.plot([], [], color='plum', alpha=0.5, linewidth=1.5, label='Fitted lines', linestyle=':')

plt.title('Sample scatter plot', fontsize=16)
plt.xlim(-5.5, 5.5)
plt.ylim(-10, 70)
plt.xlabel('x', fontsize=14)
plt.xticks(fontsize=12)
plt.ylabel('y', fontsize=14)
plt.yticks(fontsize=12)
plt.legend(fontsize=14, loc='upper left', framealpha=1)
plt.savefig("output1_1.pdf", transparent=True)
plt.show()

```

✓ 1.0s

Python

```

#chi square直方圖
plt.figure(figsize=(10, 6))

plt.hist(chi2_values, bins=30, density=True, alpha=0.7, label='χ²', color='C6',
         histtype='stepfilled')

x_chi = np.linspace(0, 50, 500)
plt.plot(x_chi, chi2.pdf(x_chi, df=7), linestyle='-', color='maroon', label='PDF (dof=7)',
         linewidth=3.5)
plt.plot([np.mean(chi2_values), np.mean(chi2_values)], [0, 0.2], linestyle='--',
         color="purple", lw=2, label='__no_legend__')
plt.text(np.mean(chi2_values)+0.5, 0.12, f'$\mu$={np.mean(chi2_values):.3f}',
         fontsize=14, color='purple')

plt.title('Chi-square Distribution (dof=8)', fontsize=16)
plt.xlabel('Chi-square value', fontsize=14)
plt.ylabel('Probability density', fontsize=14)
plt.xlim(left=0, right=30)
plt.ylim(bottom=0, top=0.13)
plt.legend(fontsize=14, framealpha=1)
plt.savefig("output1_2.pdf", transparent=True)
plt.show()

```

✓ 0.1s

Python

7. Mark a certain χ^2 value with its corresponding **p-value** on the histogram.

```
# Select a specific  $\chi^2$  value and calculate its p-value
specific_chi2_value = chi2_values[i] # Use the  $\chi^2$  value at index i
specific_p_value = chi2.sf(specific_chi2_value, df=dof) # Calculate p-value using survival function

# Plot the histogram
plt.figure(figsize=(10, 6))
plt.hist(chi2_values, bins=30, density=True, alpha=0.7, label='χ²', color='C6', histtype='stepfilled')

# Plot the  $\chi^2$  PDF
x_chi = np.linspace(0, 50, 500)
plt.plot(x_chi, chi2.pdf(x_chi, df=dof), linestyle='-', color='maroon', label='PDF (dof=7)', linewidth=3.5)

# Mark the specific  $\chi^2$  value
plt.axvline(specific_chi2_value, color='blue', linestyle='--', linewidth=2, label=f'χ²={specific_chi2_value:.2f}')
plt.text(specific_chi2_value + 0.5, 0.1, f'p={specific_p_value:.3f}', fontsize=14, color='blue')

# Add labels, title, and legend
plt.title('Chi-square Distribution (dof=8)', fontsize=16)
plt.xlabel('Chi-square value', fontsize=14)
plt.ylabel('Probability density', fontsize=14)
plt.xlim(left=0, right=30)
plt.ylim(bottom=0, top=0.13)
plt.legend(fontsize=14, framealpha=1)

# Save and show the plot
plt.savefig("output1_3.pdf", transparent=True)
plt.show()
```

✓ 0.1s Python

2.2 Practice 2

1. Using prior knowledge, select an appropriate function with noise to generate your dataset.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from scipy.stats import chi2
```

✓ 0.0s Python

```
np.random.seed(42)
```

✓ 0.0s Python

2. Plot a histogram of the χ^2 values obtained from fitting each dataset.
3. Mark the **p-value** get from the fitting result of mean (y) .
4. You might get a χ^2 of the mean (y) fitting result close to zero and a **p-value** close to 1.
5. Also mark [5% significance level](#), and [95% significance level](#) on the histogram.


```
def parabolic(x, a, b, c):  
    return a * x**2 + b * x + c
```

✓ 0.0s

Python

```
# Generate synthetic data  
true_a = 2.0  
true_b = 1.0  
true_c = 0.5  
  
sigma0 = 3.0  
  
# Generate data  
n_data = 10  
n_sets = 1000  
  
x = np.linspace(-5, 5, n_data)
```

✓ 0.0s

Python

```
# Calculate the 5% and 95% significance levels  
significance_5 = chi2.ppf(0.05, df=7)  
significance_95 = chi2.ppf(0.95, df=7)  
  
# Plot the histogram with significance levels  
plt.figure(figsize=(10, 6))  
  
plt.hist(chi2_values, bins=30, density=True, alpha=0.7, label='χ²', color='C6', histtype='stepfilled')  
  
x_chi = np.linspace(0, 50, 500)  
plt.plot(x_chi, chi2.pdf(x_chi, df=7), linestyle='-', color='maroon', label='PDF (dof=8)', linewidth=3.5)  
  
# Mark the mean chi-square value  
plt.plot([mean_chi2, mean_chi2], [0, 0.2], linestyle='--', color="purple", lw=2, label='Mean χ²')  
plt.text(mean_chi2 + 0.5, 0.11, f'$\mu$={mean_chi2:.3f}\n$p$={p_value:.3f}', fontsize=14, color='purple')  
  
# Mark the 5% and 95% significance levels  
plt.axvline(significance_5, color='blue', linestyle='--', linewidth=2, label='5% Significance Level')  
plt.axvline(significance_95, color='green', linestyle='--', linewidth=2, label='95% Significance Level')  
  
plt.title('Chi-square Distribution (dof=8)', fontsize=16)  
plt.xlabel('Chi-square value', fontsize=14)  
plt.ylabel('Probability density', fontsize=14)  
plt.xlim(left=0, right=30)  
plt.ylim(bottom=0, top=0.13)  
plt.legend(fontsize=14, framealpha=1)  
plt.savefig("output2_4.pdf", transparent=True)  
plt.show()
```

✓ 0.1s

Python

3 實驗數據與分析

3.1 Practice 1

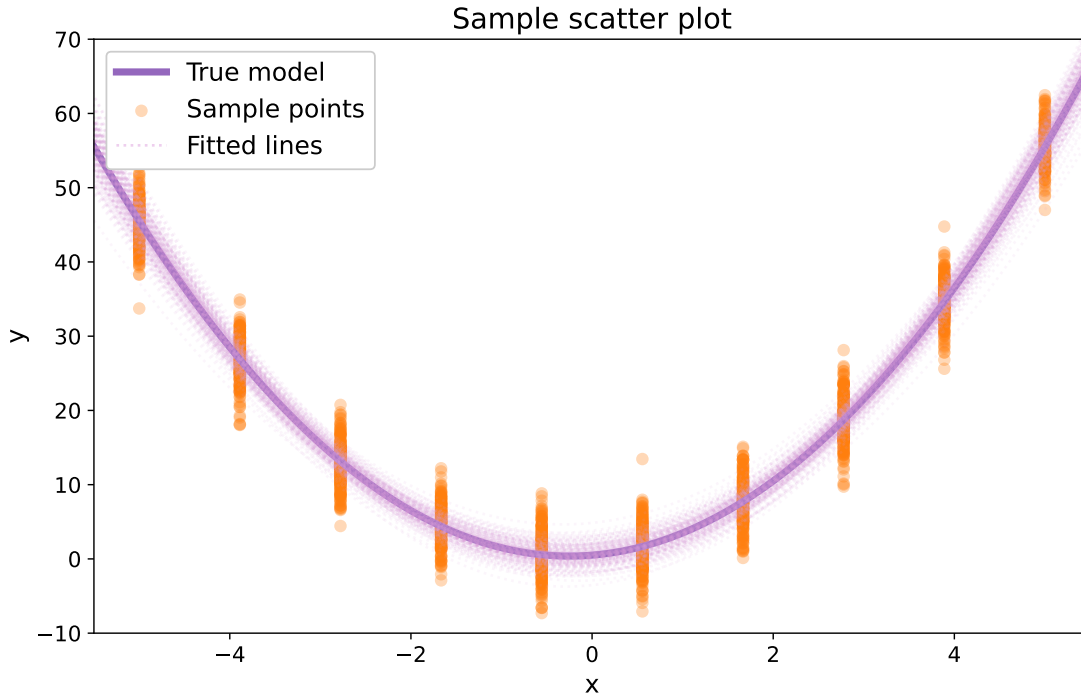


Figure 4: Scatter plot of synthetic data and their fitted parabolas.

Different from the practices of the previous weeks using a linear function to produce experimental data, we tried a different function, i.e., parabola $y = ax^2 + bx + c$, to generate our data points to learn and manipulate different scenarios. Fig. 4 shows not only how the synthetic data are scattered, but also how the data are fitted. The chi-square distribution is plotted in Fig. 5 and the mean χ^2 is marked.

To calculate the p-value of the corresponding chi-square without using `scipy.stats` functions, we referred to the basic definition:

$$p = P(\chi^2 \geq \chi_{obs}^2) = 1 - \text{CDF}_{\chi^2}(\chi_{obs}^2, k) = 1 - \frac{\gamma(k/2, x/2)}{\Gamma(k/2)} \quad (1)$$

where the γ is the lower incomplete gamma function and Γ is the gamma function. With this formula, we chose to calculate the p-value of the mean χ^2 , which is also labeled in the Fig. 5.

Moreover, the cumulative distribution function (CDF) of the experimental chi-square and the theoretical distribution are demonstrated in Fig. 6. As we can see, the empirical CDF (blue solid line) and the theoretical CDF (red dashed line) ideally overlap.

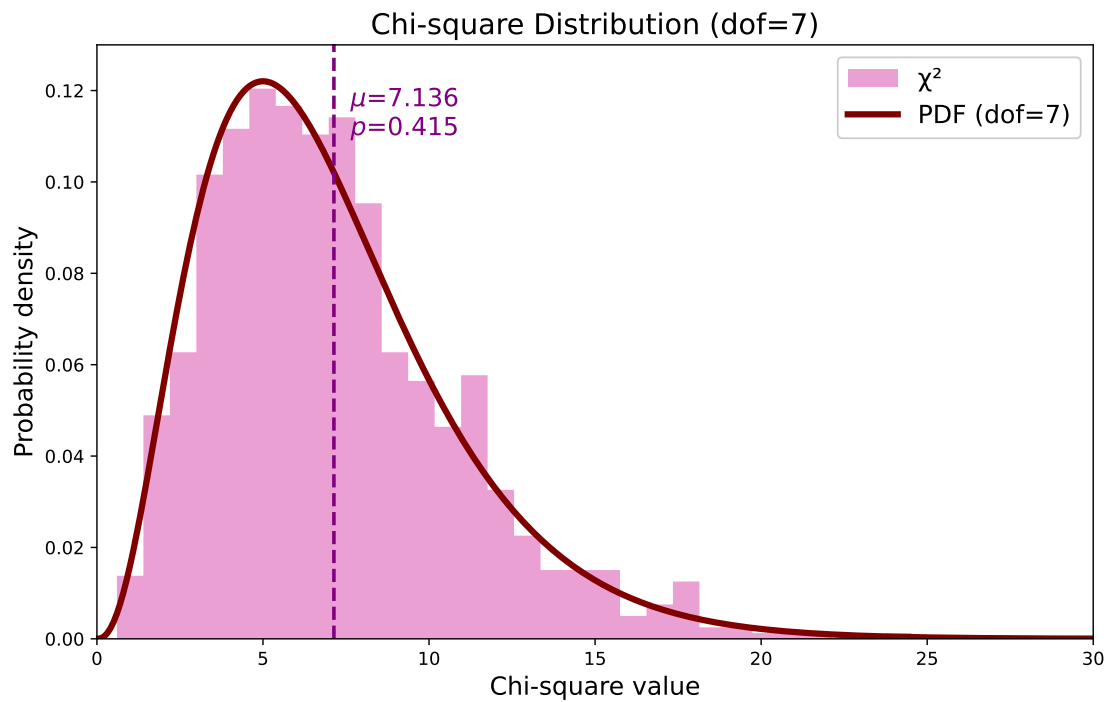


Figure 5: Histogram of the chi-square distribution. The dashed line and text mark the mean χ^2 and its p-value.

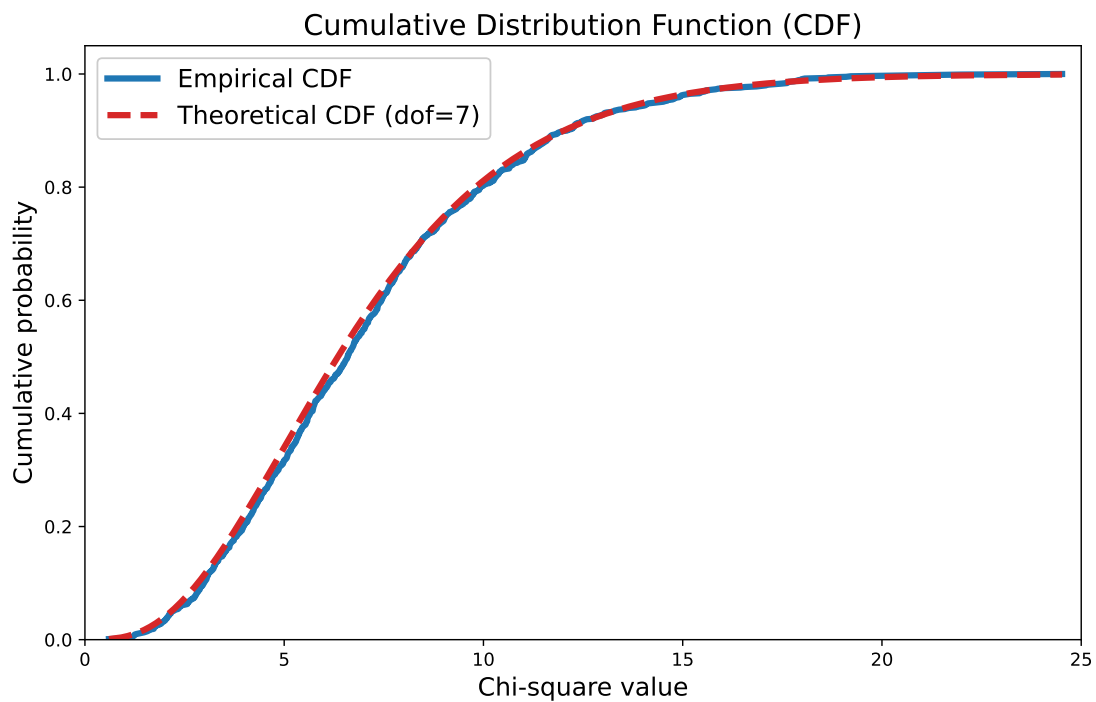


Figure 6: CDFs of the empirical theoretical values

3.2 Practice 2

```
chi-square of mean(y)= 0.004356074591544417
p-value of mean(y)= 0.9999999999586144
```

Figure 7: The chi-square of mean (y) and its p-value

The chi-square of the mean (y), derived from the fitted line, is 0.004376. Accordingly, the p-value of the chi-square value is extremely close to 1 (the actual values are in the Fig. 7). These indeed match our expectation that 'You might get a χ^2 of the mean (y) fitting result close to zero and a **p-value** close to 1.'

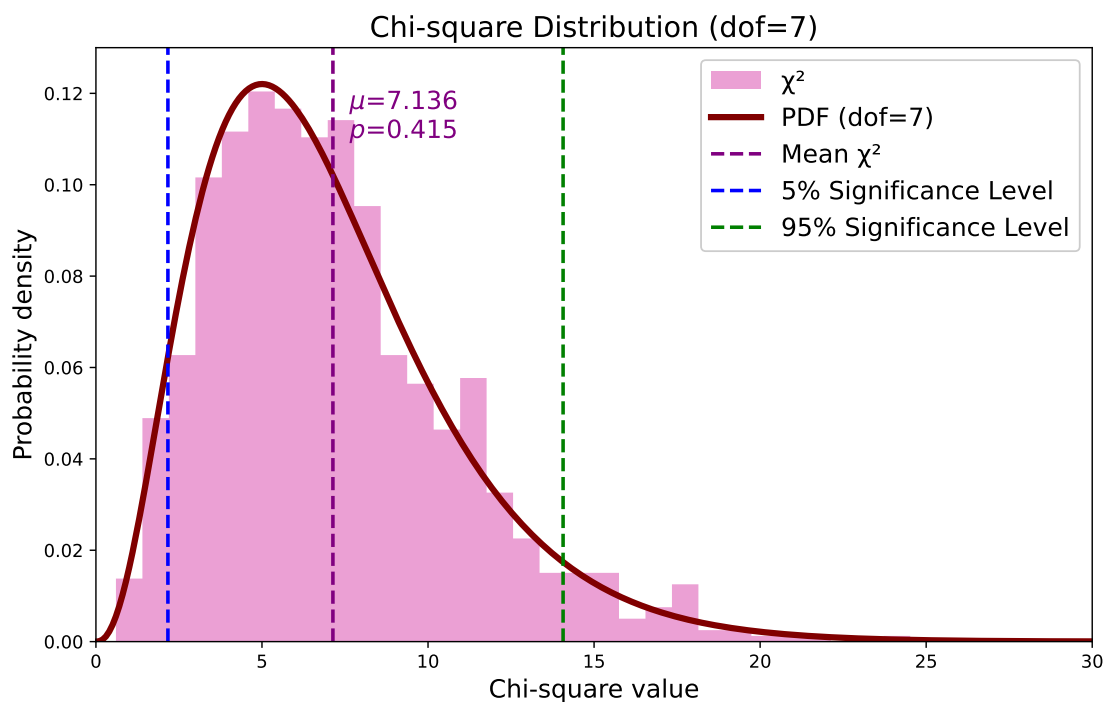


Figure 8: Similar plot as Fig. 5 and the the significant levels of 5% and 95% are labeled.

Fig. 8 plots similar characteristics, including the experimental chi-square distribution, the probability density function (PDF) with 7 degrees of freedom, the mean χ^2 , and its p-value. Also, with `scipy.stats.chi2.ppf` to calculate the significance level, 5% and 95% are marked with blue and green dashed lines, respectively. The area between 5% and 95% significance level is roughly the same as the Fig. 3.

Further discussion and analysis of chi-square, p-value, and significance level are in the following section.

4 問題討論

4.1 Practice 1

4.1.1 Will the fitted parameters be closer to the initial parameters you set to generate the data points with larger p-value? You should trust your results rather than rely on preconceived bias.

我們選出各選出 $p < 0.05$ 、 $0.4 < p < 0.6$ 、 $p > 0.9$ 的其中一組資料，並繪出對應之最佳擬合參數所建立的模型與 true model 比較。

```
#計算每組fitting的p-value(自由度=資料點數-3個參數)
dof = n_data - 3
p_values = 1 - chi2.cdf(chi2_values, dof)

#選出三組範例：每個區間找一組符合條件的
def select_example(p_range_func):
    for i, p in enumerate(p_values):
        if p_range_func(p):
            return i, fit_results[i], all_data[i], chi2_values[i]
    return None

examples = {
    "p < 0.05": select_example(lambda p: p < 0.05),
    "0.4 < p < 0.6": select_example(lambda p: 0.4 < p < 0.6),
    "p > 0.9": select_example(lambda p: p > 0.9)
}

#印出範例的chi2與擬合參數
for label, (idx, params, data, chi2_val) in examples.items():
    print(f"{label}")
    print(f"Index: {idx}")
    print(f"Chi-square: {chi2_val:.4f}")
    print(f"Fitted parameters: a = {params[0]:.4f}, b = {params[1]:.4f}, c = {params[2]:.4f}\n")

#畫圖比較三組擬合曲線與true model
x_dense = np.linspace(min(x), max(x), 300)
true_y = true_a * x_dense**2 + true_b * x_dense + true_c

plt.figure(figsize=(10, 6))
plt.plot(x_dense, true_y, 'k--', label='True Model')

colors = ['red', 'blue', 'green']
for (label, (idx, params, data, _)), color in zip(examples.items(), colors):
    a, b, c = params
    y_fit = a * x_dense**2 + b * x_dense + c
    plt.plot(x_dense, y_fit, color=color, label=f'{label} Fit')
    plt.scatter(x, data, color=color, s=20, alpha=0.5)

plt.title("Fitted Curves vs. True Model by p-value Group")
plt.xlabel("x")
plt.ylabel("y")
plt.legend()
plt.tight_layout()
plt.savefig("output1_5.pdf", transparent=True)
plt.show()
```

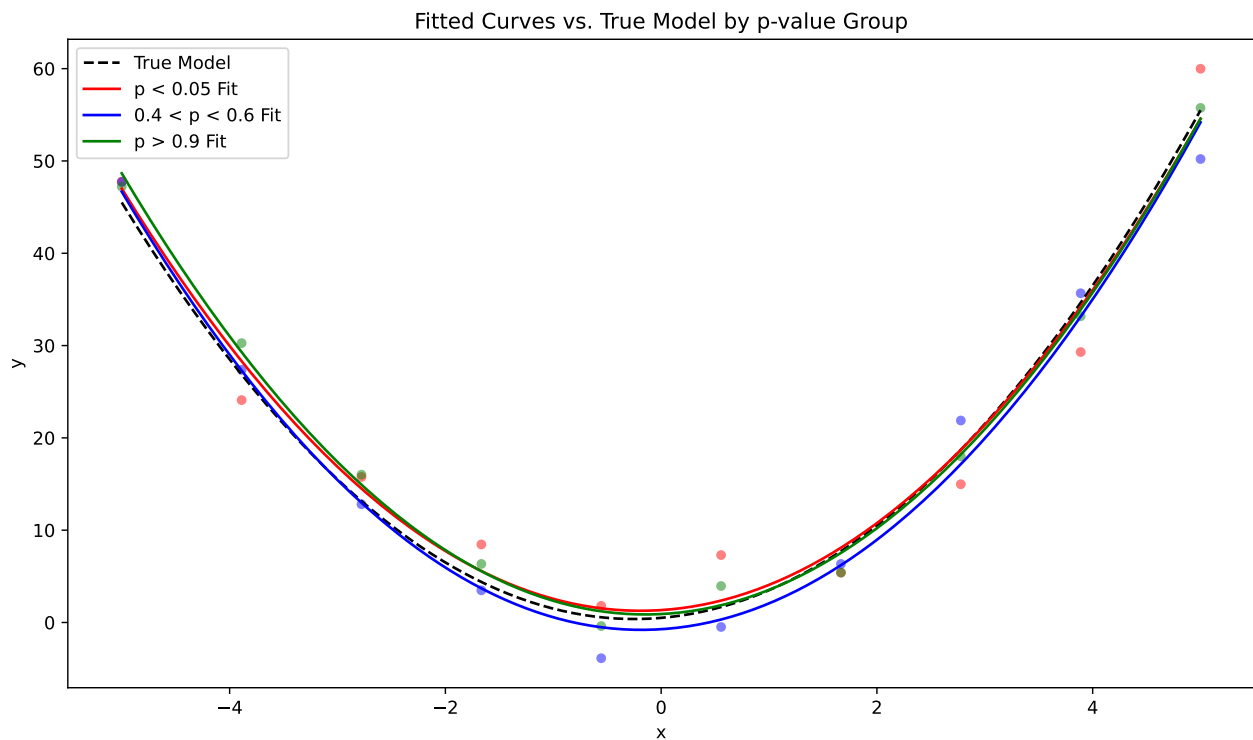


Figure 9: 選三種不同情況的p-value對應之數據，與true model比較

每一組詳細資料：

- $p < 0.05$
 - Index: 60
 - Chi-square: 14.1291
 - Fitted parameters: $a = 1.9778$, $b = 0.7572$, $c = 1.3412$
- $0.4 < p < 0.6$
 - Index: 4
 - Chi-square: 6.4121
 - Fitted parameters: $a = 2.0462$, $b = 0.7519$, $c = -0.7291$
- $p > 0.9$
 - Index: 21
 - Chi-square: 2.0211
 - Fitted parameters: $a = 2.0284$, $b = 0.5925$, $c = 0.9131$

原始數據	$p < 0.05$	$0.4 < p < 0.6$	$p > 0.9$
$a = 2.0$	1.11%	24.3%	160%
$b = 1.0$	2.31%	24.8%	246%
$c = 0.5$	1.42%	40.8%	82.6%

Table 2: 每組擬合參數與原始設定的誤差

雖然統計的理論上，較高的 p -value 通常表示擬合結果會較為精準，所以會預期擬合出來的參數會更接近原始設定的數值。然而，Tab.2 卻反映了不一樣的結果。

- 在 $p > 0.9$ 那組數據中，理應預期他的誤差要最小，結果它的誤差卻是最大的，甚至比其餘二組有非常明顯的差距。
- $p < 0.05$ 預期是擬合結果最差的，結果它反而是最接近 true model 的數據。

這說明了 p -value 雖然可以反映出擬合後與資料的吻合程度，但其實不能保證擬合後的參數會最接近真實情況；當然也可能是因為我們是隨機選擇每一組的數據，或許剛好選到的參數是偏離原始值較多的一組。所以我們不能預設「 p -value 越大擬合參數就會越準」，還是要根據實際的擬合結果判斷，而且也不能單就依賴 p -value 作為唯一依據，應該要搭配更多的方式做更全面的統計分析。

4.2 Practice 2

4.2.1 State the goodness of fitting result by comparing the p-value with the significance level.

p -value 以及 significance level 的定義在本實驗報告前言部分已詳細介紹。在這次實驗中，我們設定的 significance level 為 5% 和 95%，這是我們判斷擬合結果好壞的標準。如果擬合的結果對應到的 p -value 數值在 0.05 到 0.95 之間，就代表此數據的 χ^2 值落在合理的範圍內，擬合結果與理論模型相符，沒有顯著偏差，表示擬合結果良好。如果 p -value 小於 0.05 或大於 0.95，則代表這筆擬合結果比較極端，不符合預期的理論數值： χ^2 過大是欠擬合、 χ^2 過小過擬合。代表這個模型並不適用於此數據擬合。

4.2.2 If the p-value get from the fit is less than 0.05, what does it mean?

初始假設是此模型適用於此次擬合，但是 p -value 小於 0.05，代表在虛無假設為真的情況下，產生目前擬合結果的機率非常低，表示擬合結果與理論模型有很大的差異，因此可以認定這是不合理的結果，並且拒絕原本的模型假設，把此次擬合判定為不合理的結果。

4.2.3 Make a conclusion how to make an accurate statement about the goodness of fitting result.

1. 比較 p -value 與 significance level，若 p -value 落在設定的 significance level 的區間內，就表示該擬合結果統計上合理。
2. 比較 χ^2 值與自由度，若 χ^2 值與自由度接近，即代表擬合品質佳。
3. 利用數據繪製成圖表，可確認趨勢是否符合預期或擬合曲線是否足夠貼合資料點與真實模型。
4. 繪製殘差圖或者與理論 CDF 做比較，確認有無系統性偏差。

5 總結

本次實驗透過兩個練習深入探討了 p-value 在統計檢定中的意涵，並實際操作卡方擬合的過程。從實驗結果得知，p-value 能有效判斷模型擬合結果是否偏離理論預期，當其落在設定的顯著水準（如 5% 到 95%）範圍內，即可視為擬合合理。此外，我們也觀察到 p-value 的大小不一定與模型參數的準確性成正比，說明在評估擬合品質時，應搭配其他統計工具與視覺化圖表進行綜合判斷，例如殘差分析與 CDF 比對。最後，本實驗不僅加深了我們對統計理論的理解，也提升了使用 Python 進行資料處理與圖像化呈現的能力，為後續更進階的實驗與研究奠定良好基礎。

6 分工

- 洪瑜：實驗分析、問題討論
- 黃巧涵：實驗分析、問題討論
- 洪懌平：實驗分析

7 Appendix

7.1 Source code

- https://github.com/hyp0515/exp_phy_ii/tree/main/may13

7.2 References

- <https://www.yongxi-stat.com/hypothesis-stat/>
- https://rpubs.com/Hsuan_/915898
- <https://homepage.ntu.edu.tw/~clhsieh/biostatistic/6/6-7.htm>
- <https://homepage.ntu.edu.tw/~huilin/2008-1/ch11.pdf>