

# A/B test 的執行流程與計算



簡報閱讀



範例與作業



問題討論



學習心得(完成)



## 重要知識點



- 假設檢定的操作流程。
- 運用 python 操作假設檢定，進行 A/B test。

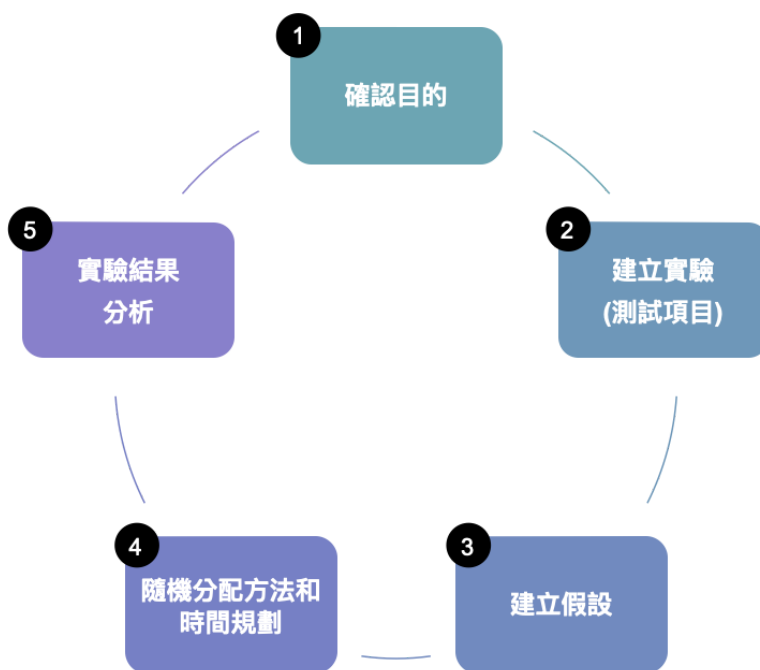
重要知識點

A/B test 的 5 步驟

情境

1 確認目的

## 2. 選擇實驗的測試項目 >



## 情境

為了更準確且更有效率的了解使用者習慣，A/B 測試成為數位行銷中不可或缺的檢測方式。今天我們以優化 Email 行銷為例，進行A/B 測試。

### 1. 確認目的

訂閱服務越來愈火紅，然而每個服務都想獲取人的注意力，每人每天都會收到很多電子郵件，如何在眾多郵件中，讓收件人能點開你的信件內容，看到你想提供的資訊是很關鍵的環節。

以你的目標族群，是否有一種客製化信件書寫格式，能有效的提高 email 行銷的觸及率，而開信率是量度觸及率的指標之一。

今天的案例的目的為「提高 email 的開信率」。

- 影響開信率的因素以信件主旨、寄件人姓名和寄送時間是比較容易操作的 A/B 測試項目。
- 不同的測試項目，建議是分開進行測試，每一項測試可以是 A/B 選擇，或 A/B/C/.. 的選擇，今天課程內容以 A/B 兩種選擇為主。

測試項目	A 選擇	B 選擇
信件主旨	收件人 + 主旨	主旨
寄信時間	平日	周末

現在的實驗，是想挑選添加何種個人化風格，讓收件者感覺到你是特別寄給他的，實驗中設計了兩種情境，左邊是偏向罐頭訊息，用統一的内容，右邊是個人化訊息，透過比較親親切的語言，運用個人化的打招呼的方式，是否能提高開信率？

大家早安~

mora早安~

那這樣的目標下要怎麼做轉換變成假設檢定的問題？

### 3. 建立假設

大家早安~

mora早安~

每一個人，隨機分配到兩種可能，每一種可能都可以分別計算開信的機率，假設左邊公版的訊息，開信的機率  $p_a$ ，右邊客製化的訊息，開信的機率為

使用背景，原本都使用公版的訊息，希望實驗有足夠的證據，證明右邊客製化的訊息較好，所以我們的假設應該這樣設計。

假設：

$$H_0: p_a \geq p_b$$

$$H_1: p_a < p_b$$

## 4. 隨機分配方法

隨機分配的方法，會根據資料特性不同，而採用不同的方式，簡單隨機分配是最常見的一種，還有很多不同的變形，常見是簡單隨機分配，這個案例中採用簡單隨機分配。

各種隨機分配的方法	定義
Randomization：簡單隨機分派	根據亂數表，隨機分配。
Stratified randomization：分層隨機	依照特定資訊 (Ex：不同縣市) 先進行分層，再依照各分層組內進行隨機分派，之後再將各組分派後的結果整併為治療組與控制組
Cluster randomization：分群隨機	執行時須先將樣本區分成數個組內個體相似性高的群組(cluster)，接著再以這些 cluster 而非個體，來進行隨機分派。

資料來源：[隨機分派\(Randomization in Experimental study\)](#)

## 5. 實驗結果分析

我們總共寄給 100 個人，兩種不同的 email 主旨，50 封是統一式開頭，50 封是個人化開頭。

統一式開頭，開信率為 10%，個人化開頭，開信率為 30%，看到這樣的資料會怎樣下決定？

## 5. 實驗結果分析 - 將抽象情況數值化

### Step 1：將抽象情況數值化

開信服從一個伯努利(Bernoulli) 分配。

統一式服從：

$$X_{a,i} \sim \text{Bernoulli}(p_a), i = 1, \dots, 50$$

平均數為  $p_a$ ，變異數為：

$$\sigma_a^2 = p_a(1 - p_a)$$

客製化服從：

$$X_{b,i} \sim \text{Bernoulli}(p_b), i = 1, \dots, 50$$

平均數為  $p_b$ ，變異數為：

$$\sigma_b^2 = p_b(1 - p_b)$$

## 5. 實驗結果分析 - 建立假設

### Step2：建立假設

根據目標假設為

$$H_1: p_a - p_b < 0$$

根據中央極限定理

$$\overline{X}_a \rightarrow N(p_a, \frac{\sigma_a^2}{n_a}), \overline{X}_b \rightarrow N(p_b, \frac{\sigma_b^2}{n_b})$$

$$\overline{X}_a - \overline{X}_b \rightarrow N(p_a - p_b, \frac{\sigma_a^2}{n_a} + \frac{\sigma_b^2}{n_b}),$$

	統一	客製	合併
開信	5	15	20
沒開信	45	35	80
開信率			

↓

$$\hat{p}_a = 5/45$$

這個有些難度，可以有需要再來翻閱

## 5. 實驗結果分析 - 計算估計值

Step 3：計算估計值

$$\overline{X}_a - \overline{X}_b \rightarrow N(p_a - p_b, \frac{\sigma_a^2}{n_a} + \frac{\sigma_b^2}{n_b}),$$

透過樣本計算參數的估計值，

$$\frac{\sigma_b^2}{n_b} : \text{樣本變異數估計值} = \frac{\hat{p}_b(1-\hat{p}_b)}{n_b-1} \longrightarrow \frac{\hat{p}(1-\hat{p})}{n_b-1}$$

因為在假設下  $p_a = p_b$  ,

所以變異數估計量用  $\hat{p}$  估計。

	統一	客製	合併
開信	5	15	20
沒開信	45	35	80
開信率			

$$\hat{p}_a = 5/45$$

這個有些難度，可搭配[影片](#)交叉學習

## 5. 實驗結果分析 - 計算 p 值

Step 4 : 計算 p 值

$$\hat{p}_a - \hat{p}_b \rightarrow N(p_a - p_b, \frac{\sigma_a^2}{n_a} + \frac{\sigma_b^2}{n_b}),$$

p 值 (p-value) 在  $H_0$  為真下，比我們收集到的資料還要極端的機率

$$\begin{aligned} P\text{值} &= P(\hat{p}_a - \hat{p}_b < 0.1 - 0.3 | H_0) = P(\hat{p}_a - \hat{p}_b - 0.2 | H_0) \\ &= P\left(\frac{(\hat{p}_a - \hat{p}_b) - (p_a - p_b)}{\sqrt{\frac{0.2(0.8)}{50-1} + \frac{0.2(0.8)}{50-1}}} < \frac{(-0.2) - (p_a - p_b)}{\sqrt{\frac{0.2(0.8)}{50-1} + \frac{0.2(0.8)}{50-1}}} \mid p_a - p_b = 0\right) \\ &= P\left(Z < \frac{(-0.2)}{\sqrt{\frac{0.2(0.8)}{50-1} + \frac{0.2(0.8)}{50-1}}}\right) = P(Z < -2.5) = 0.0062 \end{aligned}$$

異數估計量用  $\hat{p}$  估計

	統一	客製	合併
開信	5	15	20
沒開信	45	35	80
開信率			

$\hat{p}_a = 5/45$

這個有些難度，可以有需要再來翻閱

## 5. 實驗結果分析 - python code

語法 for 雙樣本比例檢定

- `statsmodels.stats.proportion.proportions_ztest(A, B, alternative='smaller')`

參數說明：

- A：兩組開信的人數
- B：兩組的總人數
- alternative：'two-sided', 'smaller', 'larger' 分別代表雙尾檢定，左尾檢定與右尾檢定

輸出

- Z-test：檢定量
- P-value：這樣檢定量下的 p-value 值



$$H_1: p_a - p_b < 0$$

```
import statsmodels.stats.proportion
A = [5,15] # 有開信的個數
B = [50,50] # 各組實驗總個數
statsmodels.stats.proportion.proportions_ztest(A, B, alternative='smaller')
(-2.4999999999999996, 0.006209665325776138)
```

Z function  
計算的檢定量

pvalue(p 值)

## 5. 實驗結果分析 - 推論

$$H_0: p_a - p_b \geq 0$$

$$H_1: p_a - p_b < 0$$

$$P\text{值} = 0.006 < \alpha = 0.05$$

結果傾向拒絕  $p_a - p_b \geq 0$ ，說明右邊客製化的開信率優於統一式的開信率。

## 知識點回顧

- 假設檢定的操作流程，共包含 5 大步驟。
- 運用 python 的 statsmodels 套件，課程中舉 最常見的 A\B test 案例，進行雙樣本比例檢定。

## 延伸閱讀

比例檢定單樣本語法

網站：[statsmodels.stats.proportion](https://statsmodels.stats.proportion)

#### Examples

```
>>> count = 5
>>> nobs = 83
>>> value = .05
>>> stat, pval = proportions_ztest(count, nobs, value)
>>> print('{0:0.3f}'.format(pval))
0.695
```

```
>>> import numpy as np
>>> from statsmodels.stats.proportion import proportions_ztest
>>> count = np.array([5, 12])
>>> nobs = np.array([83, 99])
>>> stat, pval = proportions_ztest(counts, nobs)
>>> print('{0:0.3f}'.format(pval))
0.159
```

## 雙樣本平均數檢定

網站：[scipy.stats.ttest\\_ind](https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_ind.html)

在今天課程教授的 A/B test 是雙樣本比例檢定，如果雙樣本平均數檢定需求，可以參考此網頁內容，詳細閱讀。

## scipy.stats.ttest\_ind

`scipy.stats.ttest_ind(a, b, axis=0, equal_var=True, nan_policy='propagate')` [\[source\]](#)

Calculate the T-test for the means of two independent samples of scores.

This is a two-sided test for the null hypothesis that 2 independent samples have identical average (expected) values. This test assumes that the populations have identical variances by default.

**Parameters:** `a, b` : *array\_like*

The arrays must have the same shape, except in the dimension corresponding to *axis* (the first, by default).

**axis** : *int or None, optional*

Axis along which to compute test. If None, compute over the whole arrays, *a*, and *b*.

**equal\_var** : *bool, optional*

If True (default), perform a standard independent 2 sample test that assumes equal population variances [1]. If False, perform Welch's t-test, which does not assume equal population variance [2].

*New in version 0.11.0.*

## 相依樣本雙樣本 python 語法

網站：[scipy.stats.ttest\\_rel](https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_rel.html)

一般商業需求，都常為獨立樣本檢定，如果有相依樣本需求，可以參考此網頁的資訊。

```
>>> rvs1 = stats.norm.rvs(loc=5, scale=10, size=500)
>>> rvs2 = (stats.norm.rvs(loc=5, scale=10, size=500) +
...        stats.norm.rvs(scale=0.2, size=500))
>>> stats.ttest_rel(rvs1, rvs2)
(0.24101764965300962, 0.80964043445811562)
>>> rvs3 = (stats.norm.rvs(loc=8, scale=10, size=500) +
...        stats.norm.rvs(scale=0.2, size=500))
>>> stats.ttest_rel(rvs1, rvs3)
(-3.9995108708727933, 7.3082402191726459e-005)
```

[下一步：閱讀範例與完成作業](#)

