

HIỂU ĐÚNG DỮ LIỆU: BẮT ĐẦU TỪ HYPOTHESIS TESTING (PHẦN 1)

Một ngày nọ, trong phòng họp sau khi thử nghiệm mô hình bán hàng bằng machine learning, sếp hỏi: "Làm sao biết chắc việc dùng ML thực sự giúp tăng hiệu quả?"

Cả phòng im lặng. Có kết quả. Có chênh lệch. Nhưng liệu nó có thật? Hay chỉ là... may mắn?

Đây là lúc Hypothesis Testing lên tiếng.

Bạn thấy conversion rate tăng từ 12% lên 15%. Có vẻ hứa hẹn, đúng không?

Nhưng nếu:

- Mùa đó vốn bán chạy hơn?
- Khách hàng nhóm thử nghiệm "dễ chốt" hơn?
- Mẫu thử quá nhỏ để tin được?

→ Nếu chỉ dựa vào sự khác biệt bề ngoài, bạn rất dễ ra quyết định sai từ một kết quả tưởng như "tốt hơn".

Vậy Hypothesis Testing là gì?

Hãy hình dung đây là một phiên tòa dữ liệu:

- H_0 (Null Hypothesis): Không có sự khác biệt / không có tác động
- H_1 (Alternative Hypothesis): Có sự khác biệt / có tác động
- Bằng chứng: Là dữ liệu bạn thu thập được
- Kết luận: Có bác bỏ H_0 hay không?

Một số khái niệm then chốt:

- P-value: Xác suất thấy được kết quả như vậy nếu H_0 đúng
- Alpha (α): Ngưỡng ý nghĩa thống kê (thường dùng 0.05)
- Reject H_0 : Khi $p\text{-value} < \alpha$
- Type I Error: Báo động nhầm (nói có khác biệt khi không có)
- Type II Error: Bỏ sót tín hiệu (nghĩ là không khác biệt khi có)

Nhưng thống kê thôi là chưa đủ. Trong môi trường kinh doanh, còn 3 lớp ý nghĩa cần đánh giá:

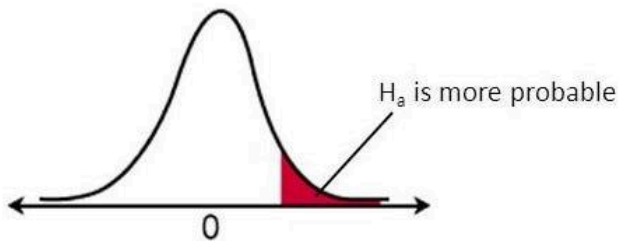
- Ý nghĩa thống kê (Statistical significance): Kết quả có đáng tin không?
- Ý nghĩa thực tiễn (Practical significance): Mức chênh lệch có đáng quan tâm?
- Ý nghĩa kinh tế (Economic significance): Tác động có đủ lớn để đầu tư, thay đổi?

Kiểm định giả thuyết không chỉ giúp xác nhận dữ liệu đủ tin cậy, mà còn giúp bạn ra quyết định đúng lúc, đúng cách.

Hypothesis Testing là kỹ năng nền tảng với người làm Data – vì:

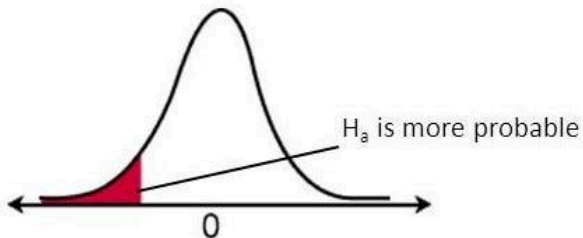
- Mọi mô hình AI đều bắt đầu bằng giả định. Kiểm định giúp bạn biết điều gì đáng tin, điều gì chỉ là nhiễu.
- Đây là công cụ cốt lõi để làm A/B Testing, đánh giá chiến dịch, kiểm chứng insight.
- Biết kiểm định giúp bạn tránh sai lầm phổ biến nhất trong phân tích dữ liệu: kết luận hấp tấp từ kết quả “trông có vẻ khác biệt”.

Một Data Scientist giỏi không chỉ viết được model, mà còn biết bằng chứng nào thực sự đủ mạnh để đưa ra quyết định.



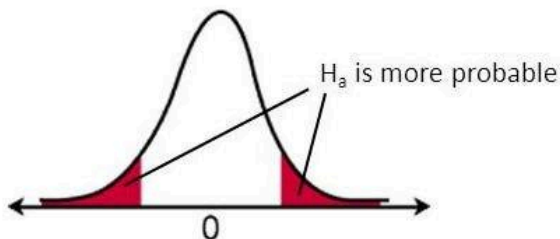
Right-tail test

$$H_a: \mu > \text{value}$$



Left-tail test

$$H_a: \mu < \text{value}$$



Two-tail test

$$H_a: \mu \neq \text{value}$$

HIỂU ĐÚNG DỮ LIỆU (PHẦN 2)

3 kiểm định phổ biến nhất – nếu bạn làm việc với dữ liệu, bạn sẽ gặp ít nhất một cái mỗi tuần

Trong phần trước, bạn đã biết:

Không phải mọi con số đều có ý nghĩa.

Muốn biết sự khác biệt trong dữ liệu là thật – hay chỉ là nhiều – bạn phải biết kiểm định đúng.


Vậy:


- Tỷ lệ mở email tăng 4% – có đáng mừng không?
- Một nhóm khách hàng dùng app lâu hơn – có nên đổ ngân sách vào?
- Gen Z có thực sự thích mobile banking hơn?

Câu trả lời không nằm trong cảm tính. Nó nằm trong kiểm định giả thuyết.

Khi làm việc với dữ liệu doanh nghiệp, bạn sẽ thường xuyên gặp 3 loại câu hỏi tương ứng với 3 kỹ thuật kiểm định phổ biến dưới đây:


1. Two-sample proportion test – So sánh tỷ lệ giữa hai nhóm

 Mục tiêu: Two-sample proportion test là kiểm định thống kê giúp so sánh tỷ lệ/phần trăm của một hiện tượng (nhị phân: xảy ra/không xảy ra) giữa hai nhóm độc lập. Mục tiêu là xác định xem sự khác biệt quan sát được giữa hai tỷ lệ có ý nghĩa thống kê hay chỉ là do ngẫu nhiên.

 Ví dụ: Một công ty muốn kiểm tra xem tỷ lệ khách hàng phản hồi sau khi nhận email tiếp thị có khác nhau giữa nhóm A nhận email cá nhân hóa và nhóm B nhận email thông thường.

- Nhóm A: 120/1000 khách hàng phản hồi (12%)
- Nhóm B: 80/1000 khách hàng phản hồi (8%)

Two-sample proportion test sẽ giúp xác định liệu chênh lệch 4% này có đáng tin cậy hay chỉ là do ngẫu nhiên.

 Ứng dụng thực tế: Trong môi trường ngân hàng, kiểm định này rất hữu ích để:


- So sánh tỷ lệ chuyển đổi giữa hai quy trình bán hàng
- Kiểm tra hiệu quả của các chiến dịch email marketing, telesales
- Đánh giá tác động của một thay đổi nhỏ (ví dụ: thiết kế landing page, cách gọi điện thoại) đến hành vi khách hàng
- So sánh tỷ lệ chấp nhận sản phẩm mới giữa hai nhóm khách thử nghiệm (A/B testing)


 Điều kiện áp dụng:


- Biến phân tích là nhị phân (ví dụ: có mở tài khoản/không, có phản hồi/không)
- Hai nhóm độc lập

- Kích thước mẫu đủ lớn
- Mỗi quan sát là độc lập (một khách hàng không xuất hiện ở cả hai nhóm)

2. One-way ANOVA – So sánh giá trị trung bình giữa nhiều nhóm

 Mục tiêu: So sánh giá trị trung bình (mean) giữa 3+ nhóm để xem có ít nhất một nhóm khác biệt có ý nghĩa thống kê hay không.


 Ví dụ: Khi muốn so sánh điểm trung bình của học sinh từ ba lớp học khác nhau: A, B, C. ANOVA sẽ giúp trả lời câu hỏi: “Có ít nhất một lớp có điểm trung bình khác biệt rõ rệt so với các lớp còn lại không?”


 Ứng dụng thực tế: So sánh giá trị giao dịch trung bình giữa các kênh bán hàng (chi nhánh, online, mobile app, tổng đài). Nếu có sự khác biệt, ta có thể điều chỉnh chiến lược theo từng kênh.


 Điều kiện áp dụng:

- Các nhóm độc lập
- Dữ liệu gần phân phối chuẩn (với mẫu lớn có thể dùng xấp xỉ)
- Phương sai giữa các nhóm xấp xỉ nhau

3. Chi-square test – Kiểm tra mối liên hệ giữa các biến phân loại


 Mục tiêu: Khi muốn kiểm tra xem hai biến phân loại (categorical) có mối liên hệ với nhau hay không.

 Ví dụ: Thực hiện khảo sát 100 người về giới tính (nam/nữ) và sở thích đồ uống (cà phê/trà). Chi-square sẽ giúp trả lời câu hỏi: “Giới tính có ảnh hưởng đến sở thích đồ uống không?”


 Ứng dụng thực tế: Phân tích hành vi khách hàng theo nhóm tuổi, giới tính, vùng miền; kiểm chứng giả định marketing.

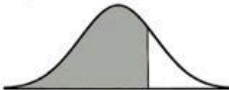
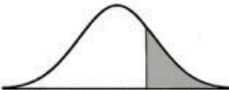
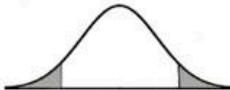
 Điều kiện áp dụng:

- Biến phân loại (categorical)
- Dữ liệu là dạng đếm (số lượng)
- Các quan sát độc lập

 Đây là lý do người học Data Science không thể bỏ qua 3 kiểm định này:

- Không hiểu kiểm định, bạn có thể đề xuất thay đổi dựa trên nhiều.
- Bạn có thể đánh rơi tín hiệu quý giá chỉ vì không kiểm chứng giả thuyết.
- Và tệ hơn, bạn sẽ báo cáo kết quả đẹp mà... vô nghĩa.

 Một Data Scientist không chỉ biết mô hình nào chạy tốt – Mà còn biết khi nào nên tin vào dữ liệu, và khi nào phải hoài nghi.

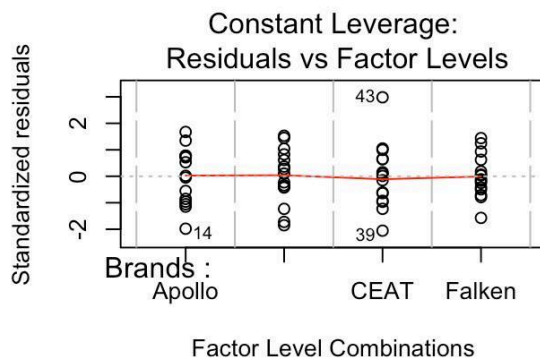
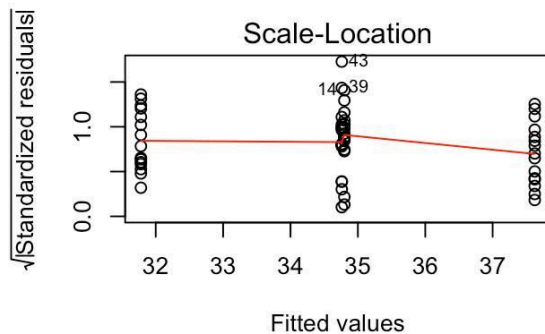
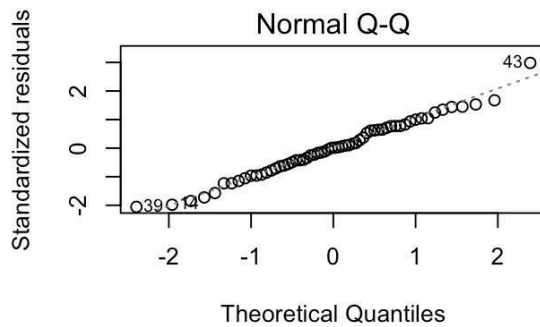
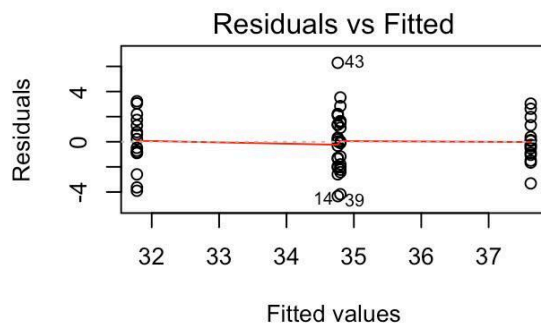
State Hypothesis		
$H_0: p_1 - p_2 = 0$ $H_a: p_1 - p_2 < 0$	$H_0: p_1 - p_2 = 0$ $H_a: p_1 - p_2 > 0$	$H_0: p_1 - p_2 = 0$ $H_a: p_1 - p_2 \neq 0$
Conditions		
<ul style="list-style-type: none"> Simple random sampling independent $p_1 n_1 \geq 10$ $p_1 n_1 \geq 10$ $p_1 n_1 \geq 10$ $p_1 n_1 \geq 10$ Population₁ $\geq 20n_1$ Population₂ $\geq 20n_2$ 		
Find Standardized Test Statistic		
$z = \frac{\hat{p}_1 - \hat{p}_2}{\sqrt{\frac{p_c(1-p_c)}{n_1} + \frac{p_c(1-p_c)}{n_2}}}$ <p>Where $p_c = \frac{p_1 n_1 + p_2 n_2}{n_1 + n_2}$</p>		
Find the p-value		
		
$H_a: p_1 - p_2 < 0$	$H_a: p_1 - p_2 > 0$	$H_a: p_1 - p_2 \neq 0$
On z-table $P(x < z)$	On z-table $P(x > z)$	On z-table $2 \times P(x > z)$
On calculator $NormCDF(-99, z, 0, 1)$	On calculator $NormCDF(z, 99, 0, 1)$	On Calculator $2 \times [NormCDF(z , 99, 0, 1)]$
Conclusion		
p-value $< \alpha$	Reject H_0	
p-value $> \alpha$	Fail to Reject H_0	

Chi-square test

$$\chi^2 = \frac{\sigma s^2}{\sigma p^2} (n-1)$$

$$\chi^2 = \sum \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Degree of Freedom	Probability of Exceeding the Critical Value								
	0.99	0.95	0.90	0.75	0.50	0.25	0.10	0.05	0.01
1	0.000	0.004	0.016	0.102	0.455	1.32	2.71	3.84	6.63
2	0.020	0.103	0.211	0.575	1.386	2.77	4.61	5.99	9.21
3	0.115	0.352	0.584	1.212	2.366	4.11	6.25	7.81	11.34
4	0.297	0.711	1.064	1.923	3.357	5.39	7.78	9.49	13.28
5	0.554	1.145	1.610	2.675	4.351	6.63	9.24	11.07	15.09
6	0.872	1.635	2.204	3.455	5.348	7.84	10.64	12.59	16.81
7	1.239	2.167	2.833	4.255	6.346	9.04	12.02	14.07	18.48
8	1.647	2.733	3.490	5.071	7.344	10.22	13.36	15.51	20.09
9	2.088	3.325	4.168	5.899	8.343	11.39	14.68	16.92	21.67
10	2.558	3.940	4.865	6.737	9.342	12.55	15.99	18.31	23.21
11	3.053	4.575	5.578	7.584	10.341	13.70	17.28	19.68	24.72
12	3.571	5.226	6.304	8.438	11.340	14.85	18.55	21.03	26.22
13	4.107	5.892	7.042	9.299	12.340	15.98	19.81	22.36	27.69
14	4.660	6.571	7.790	10.165	13.339	17.12	21.06	23.68	29.14
15	5.229	7.261	8.547	11.037	14.339	18.25	22.31	25.00	30.58
16	5.812	7.962	9.312	11.912	15.338	19.37	23.54	26.30	32.00
17	6.408	8.672	10.085	12.792	16.338	20.49	24.77	27.59	33.41
18	7.015	9.390	10.865	13.675	17.338	21.60	25.99	28.87	34.80
19	7.633	10.117	11.651	14.562	18.338	22.72	27.20	30.14	36.19
20	8.260	10.851	12.443	15.452	19.337	23.83	28.41	31.41	37.57
22	9.542	12.338	14.041	17.240	21.337	26.04	30.81	33.92	40.29
24	10.856	13.848	15.659	19.037	23.337	28.24	33.20	36.42	42.95
26	12.198	15.379	17.292	20.843	25.336	30.43	35.56	38.89	45.64
28	13.565	16.928	18.939	22.657	27.336	32.62	37.92	41.34	48.28
30	14.953	18.493	20.599	24.478	29.336	34.80	40.26	43.77	50.89
40	22.164	26.509	29.051	33.660	39.335	45.62	51.80	55.76	63.69
50	27.707	34.764	37.689	42.942	49.335	56.33	63.17	67.50	76.15
60	37.485	43.188	46.459	52.294	59.335	66.98	74.40	79.08	88.38
Not Significant								Significant	



HIỂU ĐÚNG DỮ LIỆU – PHẦN 3

Từ lý thuyết đến thực chiến: Ứng dụng Hypothesis Testing trong tình huống thật

Điều nguy hiểm nhất khi làm phân tích dữ liệu là gì?

Không phải sai số.

Không phải model yếu.

Mà là... kết luận hấp tấp từ một con số “trông có vẻ đẹp”.

Một chi nhánh vừa thử nghiệm quy trình bán hàng mới – ứng dụng công cụ số để tăng hiệu suất. Sau 3 tháng thử nghiệm, conversion rate tăng từ 15.5% lên 17.6%.

Liệu có nên triển khai đại trà? Đây là lúc cần Hypothesis Testing.

Tình huống : Kiểm tra hiệu quả quy trình bán hàng mới

Business Context: Chi nhánh A áp dụng quy trình mới có sử dụng công cụ số; chi nhánh B vẫn sử dụng quy trình truyền thống. Sau 3 tháng thử nghiệm, chúng ta muốn đánh giá hiệu quả để xem xét có triển khai quy trình mới trên toàn bộ các chi nhánh hay không.

 Bước 1 – Đặt câu hỏi đúng:

- H_0 : Tỷ lệ chuyển đổi giữa 2 chi nhánh bằng nhau

- H_1 : Tỷ lệ chuyển đổi khác nhau


 Bước 2 – Chọn phương pháp:

Two-sample proportion test, vì cần so sánh 2 conversion rate giữa chi nhánh A và B

 Bước 3 – Dữ liệu mô phỏng + Tính toán (ảnh minh họa):

- Conversion rate: 17.6% (mới) vs 15.5% (cũ)

- P-value = 0.0037

 Bước 4 – Kết luận:

Sau khi tiến hành kiểm định mẫu, thu được p-value = 0.0037 (< 0.05)


→ Bác bỏ H_0 : Có đủ bằng chứng thống kê để kết luận rằng quy trình bán hàng mới tạo ra tỷ lệ chuyển đổi khác biệt so với quy trình cũ.

Từ góc nhìn kinh doanh: Với mức chênh lệch conversion rate giữa 2 nhóm là ~2% (17.6% vs 15.5%) và kết quả kiểm định có ý nghĩa thống kê, quy trình mới cho thấy tiềm năng cải thiện tỷ lệ chuyển đổi (conversion rate).

Hành động khuyến nghị: Cân nhắc mở rộng thử nghiệm sang các chi nhánh khác hoặc bắt đầu triển khai dần, kết hợp theo dõi hiệu quả thực tế sau rollout.

Vì vậy, điều quan trọng không nằm ở con số đẹp, mà ở chỗ bạn biết:

 Khi nào có thể tin vào dữ liệu

 Khi nào cần nghi ngờ và kiểm định lại

Nếu bạn học Data Science mà chưa hiểu Hypothesis Testing – thì bạn chỉ đang bắt trend, chưa thật sự phân tích.

Bước 3: Tạo sample data và tính toán các chỉ số thống kê:

```
import pandas as pd
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt
```

```
# Tạo sample data
np.random.seed(42)
new_process = np.random.binomial(1, 0.18, 5550)
old_process = np.random.binomial(1, 0.16, 5300)
```

```
# Tạo DataFrame
data = pd.DataFrame({
    'process': ['new']*5550 + ['old']*5300,
    'converted': np.concatenate([new_process, old_process])
})
```

```
# Tính conversion rates
conversion_summary = data.groupby('process')['converted'].agg(['count', 'sum', 'mean'])
conversion_summary['conversion_rate'] = conversion_summary['mean']
print("Conversion Summary:")
print(conversion_summary)
```

```
Conversion Summary:
      count  sum      mean  conversion_rate
process
new      5550  977  0.176036           0.176036
old      5300  823  0.155283           0.155283
```

```
# Hypothesis testing
from statsmodels.stats.proportion import proportions_ztest
```


```
# Prepare data for test
successes = conversion_summary['sum'].values
nobs = conversion_summary['count'].values
```

```
# Two-sample proportion test
z_stat, p_value = proportions_ztest(successes, nobs)
```

```
print(f"\nTest Results:")
print(f"Z-statistic: {z_stat:.4f}")
print(f"P-value: {p_value:.4f}")
print(f"Significance level: 0.05")
print(f"Reject H0: {p_value < 0.05}")
```

```
Test Results:
Z-statistic: 2.9048
P-value: 0.0037
Significance level: 0.05
Reject H0: True
```


HIỂU ĐÚNG DỮ LIỆU – PHẦN 4

 Mobile App có phải là “con gà đẻ trứng vàng”?
Hay chỉ là... ảo giác dữ liệu?

Bạn đang làm phân tích dữ liệu cho ngân hàng. Một câu hỏi quen thuộc được đặt ra từ ban lãnh đạo:

“Giá trị giao dịch trung bình giữa các kênh có khác biệt không?

Có nên đầu tư nhiều hơn vào Mobile App hay vẫn giữ trọng tâm ở Chi nhánh?”

Đây không chỉ là câu hỏi của ngân hàng.

Đây là dạng câu hỏi bạn sẽ gặp mỗi tuần nếu làm trong Data hoặc Business Intelligence.

Tình huống: So sánh hiệu suất giữa các kênh bán hàng

Business Context: Ngân hàng đang vận hành nhiều kênh bán hàng song song để phục vụ khách hàng, bao gồm:

- Giao dịch trực tiếp tại chi nhánh (Branch)
- Giao dịch trực tuyến qua website (Online)
- Giao dịch trên ứng dụng di động (Mobile App)
- Tư vấn và bán hàng qua tổng đài (Phone Banking)

Ban lãnh đạo muốn đánh giá xem giá trị trung bình mỗi giao dịch (deal size) có sự khác biệt giữa các kênh hay không, nhằm tối ưu hóa nguồn lực và định hướng chiến lược phân phối sản phẩm.

Câu hỏi đặt ra là: “Liệu có kênh nào đang mang lại giá trị giao dịch cao hơn hẳn các kênh còn lại không? Và sự khác biệt đó có ý nghĩa thống kê không?”

Việc trả lời câu hỏi này sẽ giúp ngân hàng:

- Xác định đâu là kênh nên ưu tiên phân phối các sản phẩm có giá trị cao
- Điều chỉnh chính sách bán hàng và khuyến mãi theo từng kênh
- Tối ưu chi phí vận hành và nguồn lực nhân sự

1 Bước 1: Định nghĩa Hypothesis:

- H_0 : Trung bình giá trị giao dịch giữa tất cả các kênh là như nhau
- H_1 : Có ít nhất một kênh có giá trị trung bình giao dịch khác biệt so với các kênh còn lại

2 Bước 2: Lựa chọn phương pháp kiểm định:

One-way ANOVA, vì so sánh trung bình giá trị giao dịch của nhiều nhóm.

3 Bước 3: Tạo sample data và tính toán các chỉ số:

(Chi tiết sample data xem ở hình ảnh minh họa)

4 Bước 4: Kết luận:

Kết quả kiểm định ANOVA cho thấy có sự khác biệt có ý nghĩa thống kê giữa giá trị giao dịch trung bình ở các kênh bán hàng ($p\text{-value} \approx 0$).

Để xác định cụ thể những cặp kênh nào khác biệt với nhau, chúng ta thực hiện các kiểm định hậu kiểm. Kết quả cho thấy: Tất cả 6 cặp so sánh đều có sự khác biệt có ý nghĩa thống kê cao ($p\text{-value} < 0.05$).

Thứ hạng giá trị giao dịch trung bình:

- Mobile App (cao nhất)
- Online (cao hơn 17.3tr so với Phone, cao hơn 1.1tr so với Branch)
- Branch (cao hơn 10.4tr so với Phone)
- Phone Banking (thấp nhất so với các kênh khác)

Từ góc nhìn kinh doanh: Mobile App là kênh mang lại giá trị cao nhất. Các kênh số (Mobile + Online) vượt trội hơn kênh truyền thống (Branch + Phone). Phone Banking có deal size thấp nhất.

Hành động khuyến nghị:

- Tăng cường đầu tư vào kênh digital (Mobile + Online).
- Tối ưu hóa Chi nhánh: Chuyển dần các giao dịch đơn giản sang kênh số; Tập trung chi nhánh vào tư vấn sản phẩm phức tạp.
- Tái cấu trúc Phone Banking: Cân nhắc giảm quy mô hoặc chuyển đổi vai trò thành hỗ trợ thay vì bán hàng chính.

Kết luận đẹp không làm nên chiến lược đúng.

Chỉ kiểm định đúng – mới giúp bạn chọn được hướng đi đúng.

Vì vậy, nếu bạn học Data Science, đừng chỉ học vẽ biểu đồ.

Hãy học nghi ngờ dữ liệu đúng chỗ, và đưa ra quyết định đúng lúc.

Bước 3: Tạo sample data và tính toán các chỉ số:

```
import pandas as pd
import numpy as np
from scipy import stats
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# Tạo sample data cho 4 channels
np.random.seed(123)

# Tạo deal sizes (trd)
branch_deals = np.random.normal(48.2, 12.8, 850)
online_deals = np.random.normal(54.7, 11.2, 2200)
mobile_deals = np.random.normal(55.1, 10.9, 1850)
phone_deals = np.random.normal(37.8, 13.1, 500)

# Tạo DataFrame
data = pd.DataFrame({
    'channel': ['branch']*850 + ['online']*2200 + ['mobile']*1850 + ['phone']*500,
    'deal_size': np.concatenate([branch_deals, online_deals, mobile_deals, phone_deals])
})

# Descriptive statistics
channel_summary = data.groupby('channel')['deal_size'].agg(['count', 'mean', 'std'])
print("Channel Performance Summary:")
print(channel_summary.round(2))
```

```
Channel Performance Summary:
      count  mean  std
channel
branch     850  47.89 12.86
mobile    1850  55.87 10.84
online    2200  54.72 10.87
phone      500  37.47 12.84
```

```
# One-way ANOVA
channel_groups = [group['deal_size'].values for name, group in data.groupby('channel')]
f_stat, p_value = stats.f_oneway(*channel_groups)

print(f"\nANOVA Results:")
print(f"F-statistic: {f_stat:.4f}")
print(f"P-value: {p_value:.6f}")
print(f"Reject H0: {p_value < 0.05}")
```

```
ANOVA Results:
F-statistic: 418.8746
P-value: 0.000000
Reject H0: True
```

```

from scipy.stats import tukey_hsd

# Post-hoc analysis using Tukey's HSD
res = tukey_hsd(branch_deals, online_deals, mobile_deals, phone_deals)
print("Tukey's HSD Post-hoc Analysis:")
print(res)

# Alternative: Pairwise t-tests with Bonferroni correction
from scipy.stats import ttest_ind

channels = ['branch', 'online', 'mobile', 'phone']
channel_data = [branch_deals, online_deals, mobile_deals, phone_deals]

print("\nPairwise Comparisons (Bonferroni corrected):")
alpha = 0.05
n_comparisons = 6
bonferroni_alpha = alpha / n_comparisons

for i in range(len(channels)):
    for j in range(i+1, len(channels)):
        t_stat, p_val = ttest_ind(channel_data[i], channel_data[j])
        significant = p_val < bonferroni_alpha
        print(f"{channels[i]} vs {channels[j]}: p = {p_val:.6f}, significant = {significant}")

```

Tukey's HSD Post-hoc Analysis:


Tukey's HSD Pairwise Group Comparisons (95.0% Confidence Interval)

Comparison	Statistic	p-value	Lower CI	Upper CI
(0 - 1)	-6.836	0.000	-8.018	-5.654
(0 - 2)	-7.982	0.000	-9.194	-6.769
(0 - 3)	10.420	0.000	8.771	12.070
(1 - 0)	6.836	0.000	5.654	8.018
(1 - 2)	-1.146	0.008	-2.069	-0.222
(1 - 3)	17.256	0.000	15.807	18.706
(2 - 0)	7.982	0.000	6.769	9.194
(2 - 1)	1.146	0.008	0.222	2.069
(2 - 3)	18.402	0.000	16.927	19.877
(3 - 0)	-10.420	0.000	-12.070	-8.771
(3 - 1)	-17.256	0.000	-18.706	-15.807
(3 - 2)	-18.402	0.000	-19.877	-16.927

Pairwise Comparisons (Bonferroni corrected):

branch vs online: p = 0.000000, significant = True
branch vs mobile: p = 0.000000, significant = True
branch vs phone: p = 0.000000, significant = True
online vs mobile: p = 0.000830, significant = True
online vs phone: p = 0.000000, significant = True
mobile vs phone: p = 0.000000, significant = True


HIỂU ĐÚNG DỮ LIỆU – PHẦN 5

 Gen Z có thật sự “nghiện” mobile banking?
Đừng đoán theo cảm tính. Hãy kiểm định bằng dữ liệu.

Marketing thường có linh cảm đúng — nhưng nếu bạn làm trong Data, bạn không được phép... chỉ tin linh cảm.

Dữ liệu sẽ nói cho bạn biết: linh cảm đó là insight hay chỉ là định kiến.

 Tình huống : Kiểm định giả định về hành vi khách hàng Gen Z


 Business Context: Marketing team cho rằng Gen Z (18–25 tuổi) có xu hướng ưu tiên sử dụng mobile banking hơn so với các thế hệ khác. Nhóm này dự kiến sẽ là đối tượng chính trong các chiến dịch phát triển sản phẩm số sắp tới.

Để kiểm chứng giả định, ngân hàng khảo sát 2.000 khách hàng thuộc 3 nhóm tuổi (Gen Z, Gen Y, Gen X) và ghi nhận kênh giao dịch chính họ thường sử dụng:


- Chi nhánh (Branch)
- Ứng dụng di động (Mobile App)
- Giao dịch trực tuyến (Online)

Câu hỏi đặt ra: "Có mối liên hệ thống kê giữa độ tuổi và kênh giao dịch ưa thích không?"


Kết quả sẽ giúp ngân hàng xác định liệu có nên đầu tư mạnh hơn vào mobile banking cho Gen Z.


 Bước 1: Định nghĩa Hypothesis:

- H_0 : Không có mối liên hệ giữa độ tuổi và kênh giao dịch ưa thích
- H_1 : Có mối liên hệ giữa độ tuổi và kênh giao dịch ưa thích

 Bước 2: Lựa chọn phương pháp kiểm định:

Chi-square, vì cần kiểm tra xem hai biến phân loại (categorical) – nhóm tuổi vs kênh giao dịch có mối liên hệ với nhau hay không.

 Bước 3: Tạo sample data và tính toán các chỉ số:

 Bước 4: Kết luận:

Kết quả kiểm định Chi-square:

- Chi-square statistic = 177.15,
- p-value ≈ 0.000000 (< 0.05) \rightarrow Bác bỏ H_0 : Có sự liên hệ có ý nghĩa thống kê giữa nhóm tuổi và kênh giao dịch ưa thích

Diễn giải chi tiết từ standardized residuals:

- Gen Z:
 - Dùng mobile banking nhiều hơn kỳ vọng (residual = +6.47)
 - Dùng branch thấp hơn kỳ vọng (residual = -7.23)

- Gen X:

- Dùng branch nhiều hơn kỳ vọng (residual = +6.86)
- Dùng mobile thấp hơn kỳ vọng (residual = -5.94)

- Gen Y: Không có khác biệt đáng kể (các residual đều < 2)

Từ góc nhìn kinh doanh:

- Giả định của Marketing được xác nhận: Gen Z thực sự ưa chuộng mobile banking hơn các thế hệ khác.

- Gen Z ít dùng chi nhánh truyền thống, trong khi Gen X vẫn còn phụ thuộc nhiều vào branch.

- Gen Y có hành vi giao dịch khá cân bằng giữa các kênh.

Hành động khuyến nghị:

- Đầu tư mạnh hơn vào mobile app với thiết kế, trải nghiệm và tính năng phù hợp cho Gen Z.

- Giảm đầu tư vào chi nhánh tại các khu vực tập trung Gen Z, tối ưu hóa nguồn lực.

- Tạo các chiến dịch cá nhân hóa theo thế hệ, ví dụ: app banking dành riêng cho Gen Z với giao diện năng động, social features v.v.

Insight tốt đến từ dữ liệu tốt – và cách bạn kiểm định nó. Nếu không kiểm định, mọi quyết định đều có thể là... “vừa đoán vừa đi”.

Bạn học Data Science để hiểu số liệu? Vậy đừng dừng lại ở biểu đồ đẹp.

👉 Học cách đặt câu hỏi đúng – kiểm định đúng – ra quyết định đúng.


```

import pandas as pd
import numpy as np
from scipy.stats import chi2_contingency
import seaborn as sns
import matplotlib.pyplot as plt

# Tạo sample data
np.random.seed(456)

# Create data
channels = ['branch', 'mobile', 'online']
age_groups = ['gen_z', 'gen_y', 'gen_x']

# Observed data
observed_data = {
    'gen_z': [120, 380, 150],      # branch, mobile, online
    'gen_y': [280, 320, 180],
    'gen_x': [340, 180, 145]
}

# Create contingency table
contingency_table = pd.DataFrame(observed_data, index=channels).T
print("Contingency Table:")
print(contingency_table)
print(f"\nTotal sample size: {contingency_table.sum().sum()}")

# Chi-square test of independence
chi2_stat, p_value, dof, expected = chi2_contingency(contingency_table)

print(f"\nChi-square Test Results:")
print(f"Chi-square statistic: {chi2_stat:.4f}")
print(f"P-value: {p_value:.6f}")
print(f"Degrees of freedom: {dof}")
print(f"Reject H0: {p_value < 0.05}")

# Expected frequencies
expected_df = pd.DataFrame(expected,
                             index=contingency_table.index,
                             columns=contingency_table.columns)
print(f"\nExpected Frequencies:")
print(expected_df.round(2))

```

Contingency Table:

	branch	mobile	online
gen_z	120	380	150
gen_y	280	320	180
gen_x	340	180	145

Total sample size: 2095

Chi-square Test Results:

Chi-square statistic: 177.1527

P-value: 0.000000

Degrees of freedom: 4

Reject H0: True

Expected Frequencies:

	branch	mobile	online
gen_z	229.59	273.03	147.37
gen_y	275.51	327.64	176.85
gen_x	234.89	279.33	150.78


```

# Calculate standardized residuals
residuals = (contingency_table - expected_df) / np.sqrt(expected_df)
print("Standardized Residuals:")
print(residuals.round(2))

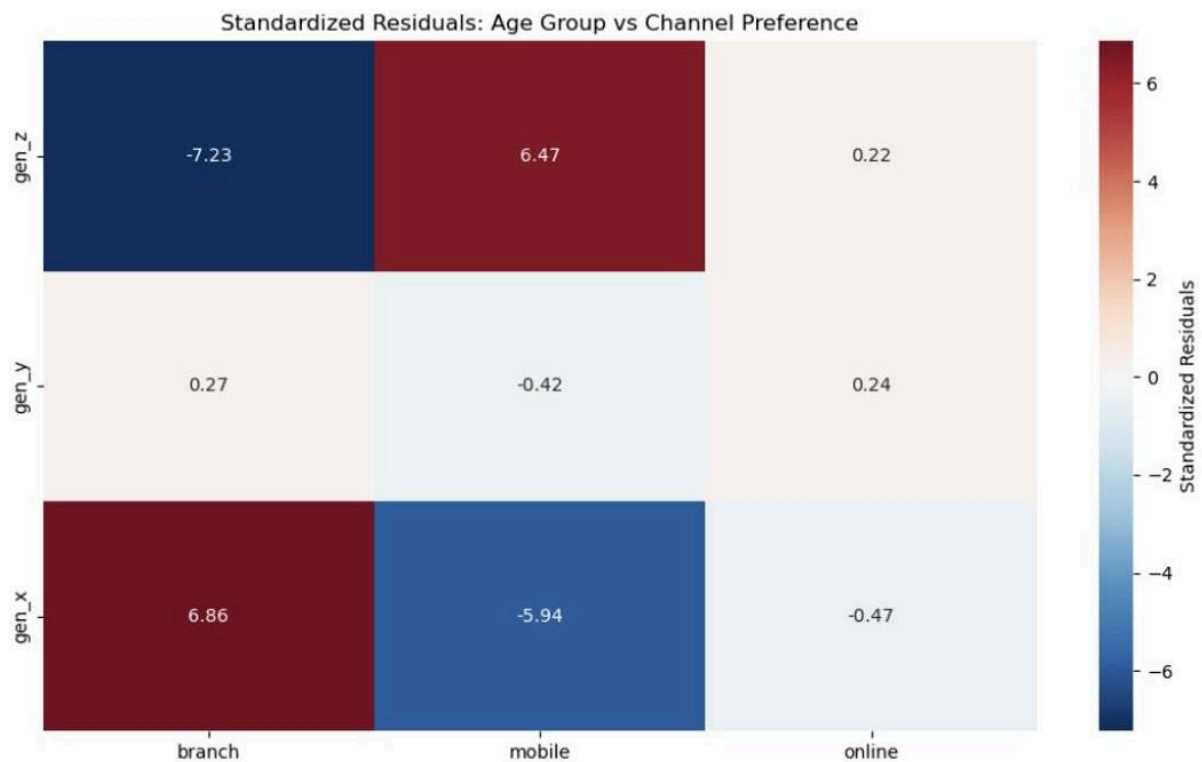
# Visualization
plt.figure(figsize=(10, 6))
sns.heatmap(residuals, annot=True, cmap='RdBu_r', center=0,
            fmt='.2f', cbar_kws={'label': 'Standardized Residuals'})
plt.title('Standardized Residuals: Age Group vs Channel Preference')
plt.tight_layout()
plt.show()

# Interpretation of residuals
print("\nInterpretation (|residual| > 2 indicates significant deviation):")
for age in residuals.index:
    for channel in residuals.columns:
        residual = residuals.loc[age, channel]
        if abs(residual) > 2:
            direction = "higher" if residual > 0 else "lower"
            print(f"{age} - {channel}: {direction} than expected (residual = {residual:.2f})")

```

Standardized Residuals:

	branch	mobile	online
gen_z	-7.23	6.47	0.22
gen_y	0.27	-0.42	0.24
gen_x	6.86	-5.94	-0.47



Interpretation (|residual| > 2 indicates significant deviation):

- gen_z - branch: lower than expected (residual = -7.23)
- gen_z - mobile: higher than expected (residual = 6.47)
- gen_x - branch: higher than expected (residual = 6.86)
- gen_x - mobile: lower than expected (residual = -5.94)