

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



CƠ SỞ TOÁN CHO KHOA HỌC MÁY TÍNH (CO5263)

Đề tài:

Xác suất và Thống kê

GVHD: TS. Nguyễn An Khương
TS. Trần Tuấn Anh
Nhóm: 7
Học viên: Trần Đăng Hùng - 2470750
Trần Hoài Tâm - 2470743
Ngô Minh Đại - 2470722
Vương Minh Toàn - 2491057
Nguyễn Đình Nhật Minh - 2370736

TP. Hồ Chí Minh, 06/2025

Mục lục

Mở đầu	5
1 Các khái niệm cơ bản về xác suất	6
1.1 Không gian mẫu, biến cố, các phép toán trên biến cố	6
1.1.1 Không gian mẫu, biến cố	6
1.2 Tổng quan về xác suất	7
1.2.1 Định nghĩa về xác suất theo quan điểm tần suất	8
1.2.2 Ba tiên đề của xác suất	8
1.3 Mối quan hệ biến cố và Quy luật xác suất	10
1.4 Các biến cố độc lập	11
1.5 Xác suất có điều kiện và công thức nhân xác suất	11
1.5.1 Xác suất có điều kiện	11
1.5.2 Công thức nhân xác suất tổng quát	11
1.5.3 Công thức nhân xác suất cho các biến cố độc lập	12
1.6 Công thức xác suất đầy đủ và công thức Bayes	12
1.6.1 Công thức xác suất đầy đủ (Law of total probability)	12
1.6.2 Công thức Bayes	13
2 Biến ngẫu nhiên và Phân phối Xác suất	13
2.1 Biến ngẫu nhiên	13
2.2 Biến ngẫu nhiên rời rạc	15
2.2.1 Định nghĩa	15
2.2.2 Tính chất phân phối xác suất	15
2.2.3 Các tham số đặc trưng	16
2.2.4 Một số phân phối xác suất phổ biến	16
2.3 Biến ngẫu nhiên liên tục	19
2.3.1 Khái niệm	19
2.3.2 Hàm phân phối tích lũy	19
2.3.3 Tính chất phân phối xác suất	20
2.3.4 Các tham số đặc trưng	21
2.3.5 Một số phân phối xác suất phổ biến	21
2.4 Các định lý về giới hạn	23

2.4.1	Bất đẳng thức Markov's	23
2.4.2	Bất đẳng thức Chebyshev's	23
2.4.3	Luật số lớn yếu, the weak law of large numbers	23
2.4.4	Định lý giới hạn trung tâm, the central limit theorem	24
2.4.5	Luật số lớn mạnh, the strong law of large numbers	24
2.5	Phân phối xác suất đa biến	25
2.5.1	Xác suất đồng thời (Joint Probability)	25
2.5.2	Xác suất có điều kiện (Conditional Probability)	31
2.5.3	Xác suất toàn phần (Total Probability)	35
2.5.4	Phân phối xác suất của hai biến ngẫu nhiên	36
2.5.5	Hiệp phương sai và hệ số tương quan	38
3	Ứng dụng trong Khoa học Máy tính	39
4	Bài toán thực tế	40
5	Hàm mật độ xác suất, Hàm phân phối tích lũy	42
5.1	Hàm Mật độ Xác suất	42
5.2	Hàm Phân phối Tích lũy	43
6	Kỳ vọng, phương sai, độ lệch chuẩn	44
6.1	Kỳ Vọng (Expectations)	44
6.2	Phương Sai	45
6.3	Độ lệch chuẩn	47
6.4	Hiệp Phương Sai	47
6.5	Thảo luận	49
7	Bài tập	51
8	Bài toán nâng cao	97
9	Kết luận	117



Danh sách bảng

1	Xác suất có điều kiện của $P(D_1 H)$	40
2	Xác suất có điều kiện của $P(D_2 H)$	41
3	Bảng xác suất đồng thời	83
4	Bảng xác suất đồng thời	84

Danh sách hình vẽ

1	Sự hội tụ xác suất sau 500 lần tung	8
2	Kết quả demo của ví dụ 1.3	10
3	Kết quả demo cho ví dụ 3.1	28
4	Kết quả demo ví dụ 3.2 cho biên của A	30
5	Kết quả demo ví dụ 3.2 cho biên của B	30
6	Kết quả demo cho ví dụ 3.4(1)	33
7	Kết quả demo cho ví dụ 3.4(2)	33
8	Kết quả demo ví dụ 3.6	35
9	Trực quan hóa mối quan hệ giữa kích thước tập dữ liệu quan sát và entropy	55
10	Monte Carlo Averaging với 10.000 lần thử	57
11	So sánh nhiều giá trị p trong phân phối Bernoulli	59
12	Mô phỏng entropy cho biến rời rạc nhiều giá trị	60
13	KL-divergence giữa phân phối ước lượng và thật	61
14	So sánh KL-divergence giữa nhiều phân phối thật khác nhau	63
15	Mô phỏng ví dụ về giới hạn không chắc chắn quan sát được.	67
16	Phương sai giảm khi n tăng	71
17	Kiểm nghiệm bất đẳng thức Chebyshev	72
18	Kiểm nghiệm Định lý CLT	73
19	Kiểm nghiệm Định lý CLT với các giá trị khác nhau của p	74
20	Kết quả bài tập 7(Python)	87
21	Biểu đồ trực quan hóa hai xác suất hậu nghiệm	87
22	Danh mục đầu tư tối ưu	96
23	Mô phỏng trường hợp xảy ra dấu bằng tại giới hạn trên của $P(A \cup B \cup C)$	102
24	Mô phỏng trường hợp xảy ra dấu bằng tại giới hạn dưới của $P(A \cup B \cup C)$	103
25	Mô phỏng trường hợp xảy ra dấu bằng tại giới hạn trên của $P(A \cap B \cap C)$	104
26	Mô phỏng trường hợp xảy ra dấu bằng tại giới hạn dưới của $P(A \cap B \cap C)$	104
27	Kết quả thực hiện demo tính toán	107
28	Mô phỏng, trực quan hóa cây quyết định	113

Mở đầu

Học máy vốn luôn gắn liền với sự bất định: chúng ta dùng những dữ liệu đã biết (features) để dự đoán điều chưa biết (target), đồng thời muốn hiểu rõ mức độ tin cậy của dự đoán đó. Chẳng hạn, trong học có giám sát, nếu đưa vào các chỉ số huyết áp, mỡ máu và tuổi tác của một bệnh nhân, ta không chỉ muốn dự đoán xem họ có khả năng bị đau tim hay không, mà còn muốn biết xác suất cụ thể – ví dụ 10% hay 30% – để bác sĩ có hướng can thiệp phù hợp. Tùy mục tiêu, ta có thể tối ưu hóa độ chính xác (chọn giá trị có xác suất cao nhất) hoặc giảm thiểu sai số trung bình so với giá trị thực.

Trong học không giám sát, bất định giúp chúng ta phát hiện “sai lệch” (anomaly). Ví dụ, một thiết bị đo nhiệt độ môi trường liên tục gửi về dãy số, việc biết được giá trị 45°C liệu có hiếm gặp trong quá khứ hay không sẽ quyết định xem ta có gắn cờ cảnh báo hay không. Còn trong học sâu, một robot dọn nhà sẽ phải cân nhắc: nếu quét sàn (hành động A) tốn 5 phút nhưng mang lại 10 điểm sạch sẽ, trong khi lau bụi (hành động B) tốn 3 phút nhưng chỉ được 4 điểm, robot sẽ chọn chuỗi hành động tối ưu sao cho tổng “phần thưởng” là lớn nhất.

Từ những ví dụ trên, có thể thấy rõ: để hiểu và áp dụng hiệu quả các mô hình học máy, chúng ta cần nắm vững nền tảng xác suất. Tài liệu này được thiết kế nhằm cung cấp cái nhìn hệ thống và dễ tiếp cận về các khái niệm cơ bản trong xác suất học – bao gồm không gian mẫu, biến cố, biến ngẫu nhiên rời rạc và liên tục, hàm phân phối xác suất, kỳ vọng, phương sai, và những định lý quan trọng như định lý Bayes hay độc lập có điều kiện.

Tài liệu này giới thiệu các khái niệm cơ bản trong xác suất, bao gồm không gian mẫu, biến cố, biến ngẫu nhiên, hàm phân phối, kỳ vọng, phương sai, và các định lý quan trọng như Bayes. Những kiến thức này không chỉ là lý thuyết khô khan mà còn là nền tảng để xây dựng các mô hình trong học máy, ra quyết định dưới rủi ro, và hiểu rõ bản chất ngẫu nhiên trong thế giới thực.

Giờ hãy bắt đầu với **Phần 2: Khái niệm cơ bản**, nơi chúng ta sẽ khám phá nền tảng của xác suất và thống kê.

1 Các khái niệm cơ bản về xác suất

1.1 Không gian mẫu, biến cố, các phép toán trên biến cố

1.1.1 Không gian mẫu, biến cố

Định nghĩa

Trong lý thuyết xác suất, một **thí nghiệm** (*experiment*) là một quy trình hay một phép đo thu được một quan sát hoặc thu được một kết quả đầu ra (*outcome*) không được dự đoán chắc chắn (*certainty*).

Ví dụ 1.1 - Phép thử: Ghi lại điểm kiểm tra; Đo lượng mưa hàng ngày; Phỏng vấn người dân để lấy ý kiến về một sắc lệnh; Kiểm tra sản phẩm được sản xuất ra để xác định xem đó có phải là sản phẩm lỗi hay sản phẩm chấp nhận được; Tung đồng xu và quan sát mặt xuất hiện.

Khi thực hiện một thí nghiệm, điều chúng ta quan sát được là một kết quả được gọi là **sự kiện đơn giản** (*simple event*), ký hiệu là E_i .

Ví dụ 1.2 - Sự kiện đơn giản: Gọi E_i là sự kiện đơn giản xuất hiện mặt thứ i khi tung hột xí ngẫu gồm 6 mặt, $i = (1, \dots, 6)$. Ta có: E_1 là sự kiện xuất hiện 1 mặt có 1 chấm, ..., E_6 là sự kiện xuất hiện 1 mặt có 6 chấm.

Biến cố (*event*) là một tập hợp các sự kiện đơn giản.

Ví dụ 1.3 - Biến cố: $A = \{E_1, E_3, E_5\}$ là biến cố quan sát có một số lẻ. $B = \{E_1, E_2, E_3\}$ là biến cố quan sát có một số nhỏ hơn 4. $C = \{E_2, E_4, E_6\}$ là biến cố quan sát có một số chẵn.

Hai biến cố được gọi là **biến cố loại trừ lẫn nhau** (*mutually exclusive events*) nếu cả hai không thể xảy ra cùng lúc.

Ví dụ 1.4 - Biến cố loại trừ: Biến cố A và C loại trừ lẫn nhau.

Không gian mẫu (*sample space*) là tập hợp tất cả các sự kiện đơn giản (*simple event*), hay tập hợp tất cả các kết quả có thể xảy ra đã được biết từ một thí nghiệm, ký hiệu là S . Biến cố là một tập hợp con của không gian mẫu.

Ví dụ 1.5 - Không gian mẫu: $S = \{E_1, E_2, E_3, E_4, E_5, E_6\}$.

1.2 Tổng quan về xác suất

Xác suất của một sự kiện (hay một biến cố, tình huống giả định) là khả năng xảy ra của sự kiện đó, được đánh giá dưới dạng một số thực nằm giữa 0 và 1.

Khi một sự kiện không thể xảy ra, thì xác suất của nó bằng 0. Ví dụ, xác suất của sự kiện “ném một đồng xu lên mà rơi lơ lửng giữa không trung mãi mãi” là 0.

Khi một sự kiện chắc chắn xảy ra, thì xác suất của nó bằng 1 (hay còn viết là 100%). Ví dụ, xác suất của sự kiện “mặt trời mọc ở hướng đông vào sáng mai” là 1.

Khi một sự kiện có thể xảy ra hoặc không xảy ra, và chúng ta không biết chắc chắn kết quả, thì xác suất của nó nằm trong khoảng từ lớn hơn 0 đến nhỏ hơn 1. Một sự kiện càng dễ xảy ra thì xác suất của nó càng gần 1; ngược lại, nếu càng khó xảy ra thì xác suất càng gần 0.

Ví dụ, giả sử tôi tham gia một trò chơi quay số may mắn mà chỉ có 1 người thắng trong số 500 người chơi. Khi đó, xác suất tôi là người thắng là

$$\frac{1}{500} = 0.002 = 0.2\%.$$

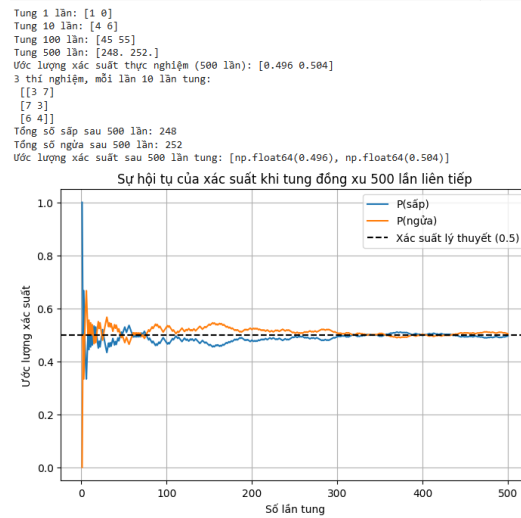
Không chỉ các sự kiện trong tương lai, mà cả các sự kiện trong quá khứ – nếu ta không có đủ thông tin để biết chắc chắn chúng đã xảy ra hay chưa – cũng có thể được gán một xác suất nào đó, thể hiện mức độ tin tưởng của chúng ta. Ví dụ, có một giả thuyết lịch sử cho rằng danh họa Van Gogh đã tự cắt tai mình vì khủng hoảng tâm lý. Dù không ai có thể biết chắc chắn, nhưng nhiều nhà nghiên cứu đánh giá giả thuyết này có xác suất cao dựa trên bằng chứng lịch sử và thư từ để lại.

Giả sử ta có một đồng xu cân đối – nghĩa là xác suất xuất hiện mặt sấp (Heads) và ngửa (Tails) đều bằng nhau. Khi tung đồng xu một lần, xác suất thấy mặt sấp là 0.5 và xác suất thấy mặt ngửa cũng là 0.5.

Tuy nhiên, nếu đồng xu có thể bị lệch (ví dụ: do khối lượng phân bố không đều), ta cần một cách kiểm tra để xác định xem nó có cân đối hay không. Cách đơn giản nhất là tung đồng xu nhiều lần và ghi lại kết quả để xem mặt nào xuất hiện thường xuyên hơn. Từ đó, ta có thể ước lượng xác suất thực nghiệm cho mỗi mặt bằng cách lấy số lần xuất hiện chia cho tổng số lần tung.

Luật số lớn (*Law of Large Numbers*) trong xác suất học cho ta biết rằng khi số lần tung đủ lớn, xác suất thực nghiệm sẽ gần với xác suất lý thuyết. Trước khi đi sâu hơn vào lý thuyết, chúng ta sẽ viết một đoạn mã để mô phỏng quá trình tung đồng xu này và quan sát sự hội tụ của xác suất.

Ta sẽ demo thí nghiệm sau (xem code demo trên notebook)



Hình 1: Sự hội tụ xác suất sau 500 lần tung

Sau 500 lần tung, tần suất thực nghiệm cho thấy gần với xác suất lý thuyết 50–50. Sự chênh lệch nhỏ này là hệ quả bình thường của tính ngẫu nhiên; khi số lần tung đủ lớn, tần suất sẽ tiến càng gần giá trị lý thuyết. Kết quả cho thấy đồng xu về cơ bản là cân đối, và nếu tiếp tục tăng số lần tung, sai số giữa thực nghiệm và lý thuyết thường sẽ giảm đi.

1.2.1 Định nghĩa về xác suất theo quan điểm tần suất

Giả định rằng một thí nghiệm được lặp đi lặp lại nhiều lần dưới điều kiện giống hệt nhau và tần suất tương đối (relative frequency) của một biến cố là tỷ lệ số lần biến cố đó xảy ra.

1.2.2 Ba tiên đề của xác suất

- Mỗi xác suất của một sự kiện đơn giản nằm giữa 0 và 1 (bao gồm 0 và 1).

$$0 \leq P(E) \leq 1$$

- Tổng xác suất của tất cả các sự kiện đơn giản trong không gian mẫu bằng 1.

$$P(S) = 1$$

- Với bất kỳ dãy biến cố loại trừ E_1, E_2, \dots sao cho $E_i \cap E_j = \emptyset$ với $i \neq j$, ta được:

$$P\left(\bigcup_{i=1}^{\infty} E_i\right) = \sum_{i=1}^{\infty} P(E_i) \quad (1)$$

Ví dụ 1.3:

Giả sử ta có một hộp chứa các quả bóng được đánh số từ 1 đến 100. Ta rút ngẫu nhiên 1 quả bóng.

Gọi các sự kiện:

- A_1 : rút được số chia hết cho 2 nhưng không chia hết cho 4
- A_2 : rút được số chia hết cho 4 nhưng không chia hết cho 8
- A_3 : rút được số chia hết cho 8 nhưng không chia hết cho 16
- A_4 : rút được số chia hết cho 16

Ta sẽ kiểm tra:

$$P\left(\bigcup_{i=1}^4 A_i\right) = \sum_{i=1}^4 P(A_i)$$

Kiểm tra tính xung khắc từng đôi một:

Các sự kiện A_1, A_2, A_3, A_4 không giao nhau từng đôi một, vì:

- Không thể có một số chia hết đồng thời cho 2 mà không chia hết cho 4 (A_1), lại vừa chia hết cho 4 mà không chia hết cho 8 (A_2), v.v.
- Do đó: $A_i \cap A_j = \emptyset$ với mọi $i \neq j$.

Tính xác suất từng sự kiện:

- Có 25 số chia hết cho 2 nhưng không chia hết cho 4 (ví dụ: 2, 6, 10, ..., 98) $\Rightarrow P(A_1) = \frac{25}{100} = 0.25$
- Có 13 số chia hết cho 4 nhưng không chia hết cho 8 $\Rightarrow P(A_2) = 0.13$
- Có 6 số chia hết cho 8 nhưng không chia hết cho 16 $\Rightarrow P(A_3) = 0.06$
- Có 6 số chia hết cho 16 $\Rightarrow P(A_4) = 0.06$

Tổng các xác suất riêng lẻ:

$$P(A_1) + P(A_2) + P(A_3) + P(A_4) = 0.25 + 0.13 + 0.06 + 0.06 = 0.5$$

Tính xác suất của hợp các sự kiện:

Tổng số phần tử trong $A_1 \cup A_2 \cup A_3 \cup A_4$ là 50 số $\Rightarrow P(A_1 \cup A_2 \cup A_3 \cup A_4) = \frac{50}{100} = 0.50$

Kết luận:

$$P\left(\bigcup_{i=1}^4 A_i\right) = \sum_{i=1}^4 P(A_i)$$

Điều này xác nhận **tiên đề cộng tính đếm được** (*countable additivity*) trong xác suất khi các sự kiện là **xung khắc đôi một**.

Ta sẽ demo ví dụ bằng python (xem code demo trên notebook)

```
Số phần tử A1 (chẵn ko chia hết 4): 25
Số phần tử A2 (chia hết 4 ko chia hết 8): 13
Số phần tử A3 (chia hết 8 ko chia hết 16): 6
Số phần tử A4 (chia hết 16): 6

Kiểm tra giao nhau giữa các sự kiện:
A1 n A2 = 0
A1 n A3 = 0
A1 n A4 = 0
A2 n A3 = 0
A2 n A4 = 0
A3 n A4 = 0

Xác suất từng sự kiện:
P(A1) = 0.250
P(A2) = 0.130
P(A3) = 0.060
P(A4) = 0.060

Tổng xác suất các sự kiện riêng lẻ (cộng tính): 0.500
Xác suất hợp A1 U A2 U A3 U A4: 0.500
```

Hình 2: Kết quả demo của ví dụ 1.3

1.3 Mỗi quan hệ biến cố và Quy luật xác suất

- Hợp (union) của hai biến cố A và B, ký hiệu là $A \cup B$, là biến cố A hoặc B hoặc cả hai xảy ra. Xác suất (công thức cộng):

$$P(A \cup B) = P(A) + P(B) - P(A \cap B) \quad (2)$$

Đặc biệt, nếu A và B loại trừ nhau, mutually exclusive hay disjoint, xác suất:

$$P(A \cup B) = P(A) + P(B) \quad \text{vì} \quad P(A \cap B) = 0 \quad (3)$$

- Phần bù (complement) của biến cố A, ký hiệu là A^c , là biến cố A không xảy ra.

$$P(A^c) = 1 - P(A) \quad (4)$$

- Giao (intersection) của hai biến cố A và B, ký hiệu là $A \cap B$, là cả biến cố A và B xảy ra.

1.4 Các biến cố độc lập

Hai biến cố A và B được coi là "độc lập" (independent events), nếu việc xảy ra của biến cố A không ảnh hưởng đến xác suất xảy ra của biến cố B. Nếu hai biến cố không độc lập, chúng được gọi là "phụ thuộc" (dependent). Khái niệm biến cố độc lập có liên quan chặt chẽ với xác suất có điều kiện.

1.5 Xác suất có điều kiện và công thức nhân xác suất

1.5.1 Xác suất có điều kiện

Xác suất biến cố A tìm được khi biến cố B xảy ra được gọi là xác suất có điều kiện (conditional probability) của A, ký hiệu $P(A|B)$ với điều kiện $P(B) > 0$.

Khái niệm xác suất có điều kiện rất quan trọng khi chúng ta quan tâm đến việc tính toán xác suất khi có sẵn một số thông tin bộ phận liên quan đến kết quả của một thí nghiệm. Ngay cả khi không có thông tin bộ phận nào, xác suất có điều kiện vẫn có thể giúp tính toán các xác suất mong muốn dễ dàng hơn.

1.5.2 Công thức nhân xác suất tổng quát

Công thức nhân xác suất (Multiplication rule) liên quan đến xác suất đồng thời của hai hoặc nhiều biến cố và thường được sử dụng cùng với xác suất có điều kiện và được xác định như sau:

$$P(A \cap B) = P(A)P(B|A) = P(B)P(A|B) \quad (5)$$

Từ công thức nhân xác suất tổng quát, ta suy ra công thức tính xác suất có điều kiện:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad \text{với } P(B) > 0 \quad (6)$$

hay

$$P(B|A) = \frac{P(A \cap B)}{P(A)} \quad \text{với } P(A) > 0$$

1.5.3 Công thức nhân xác suất cho các biến cố độc lập

Nếu biến cố A và B độc lập thì:

$$P(B|A) = P(B) \quad \text{hay} \quad P(A|B) = P(A) \quad (7)$$

$$P(A \cap B) = P(A)P(B) \quad (8)$$

Nếu biến cố A, B, C là các biến cố loại trừ lẫn nhau (hay độc lập từng đôi một) thì:

$$P(A \cap B \cap C) = P(A)P(B)P(C) \quad (9)$$

1.6 Công thức xác suất đầy đủ và công thức Bayes

1.6.1 Công thức xác suất đầy đủ (Law of total probability)

Cho tập các sự kiện giản đơn S_1, S_2, \dots, S_k loại trừ lẫn nhau và đầy đủ, xác suất của một biến cố A được xác định như sau:

$$P(A) = P(S_1)P(A|S_1) + P(S_2)P(A|S_2) + \dots + P(S_k)P(A|S_k) \quad (10)$$

1.6.2 Công thức Bayes

Công thức Bayes (Bayes's rule hoặc Bayes's formula) là một công thức quan trọng trong xác suất. Công thức này cho phép tính toán xác suất hậu nghiệm (posterior probabilities) dựa trên xác suất tiên nghiệm (prior probabilities) và xác suất có điều kiện. Cụ thể, với k tập sự kiện đơn giản S_1, S_2, \dots, S_k loại trừ lẫn nhau và đầy đủ với xác suất tiên nghiệm $P(S_1), P(S_2), \dots, P(S_k)$, nếu biến cố A xảy ra, xác suất hậu nghiệm có điều kiện của S_i được xác định như sau:

$$P(S_i|A) = \frac{P(S_i)P(A|S_i)}{\sum_{j=1}^k P(S_j)P(A|S_j)} \quad \text{với } i = 1, 2, \dots, k \quad (11)$$

2 Biến ngẫu nhiên và Phân phối Xác suất

2.1 Biến ngẫu nhiên

Biến ngẫu nhiên (Random variable) x có giá trị được giả định là thu được tương ứng với kết quả của một phép thử, là một cơ hội hoặc sự kiện ngẫu nhiên. Ví dụ như số sản phẩm lỗi trên số sản phẩm được chọn ngẫu nhiên... Biến ngẫu nhiên là một khái niệm trung tâm trong lý thuyết xác suất. Các biến ngẫu nhiên có thể là rời rạc hoặc liên tục (random continuous variable).

Giả sử ta xét một biến ngẫu nhiên X . Ta giả định rằng tồn tại nhiều tình huống (hay kịch bản) khác nhau có thể xảy ra, và trong mỗi tình huống đó, X sẽ nhận một giá trị cụ thể. Do đó, ta có thể mô hình hóa biến ngẫu nhiên này như một hàm số ánh xạ từ không gian các tình huống có thể xảy ra sang tập số thực:

$$X : \Omega \rightarrow \mathbb{R}$$

Trong đó, Ω là tập hợp đại diện cho toàn bộ các tình huống (hay còn gọi là không gian mẫu). Các tình huống riêng lẻ hoặc nhóm tình huống (tập con của Ω) được gọi là **sự kiện**.

Giả sử mỗi sự kiện đều được gán một xác suất để biểu thị khả năng xảy ra của nó. Khi đó, Ω kết hợp với một độ đo xác suất P sẽ tạo thành **không gian xác suất**, ký hiệu là (Ω, P) . Chúng ta cũng giả định rằng, với mọi cặp số thực $a < b$, xác suất $P(a < X < b)$ luôn tồn tại. Nói cách khác, tập hợp

$$\{\omega \in \Omega \mid a < X(\omega) < b\}$$

phải là một tập đo được trong không gian xác suất (Ω, P) . Khi điều kiện này được thỏa mãn,

hàm X được gọi là **hàm đo được**.

Từ đó, ta có thể phát biểu hai định nghĩa toán học như sau:

Định nghĩa 2.1 [6]: Một biến ngẫu nhiên (random variable) với giá trị thực là một hàm số đo được trên một không gian xác suất:

$$X : (\Omega, P) \longrightarrow \mathbb{R} \quad (2.1)$$

Định nghĩa 2.2 [6]: Nếu ta có hai biến ngẫu nhiên X, Y (với cùng một mô hình không gian xác suất), thì ta sẽ nói rằng $X = Y$ *theo nghĩa xác suất*, hay $X = Y$ *hầu khắp mọi nơi*, nếu như sự kiện “ $X = Y$ ” có xác suất bằng 1 (tức là tập hợp các trường hợp mà ở đó $X \neq Y$ có xác suất bằng 0, có thể bỏ qua).

Ví dụ 2.1: Một trò chơi gồm 4 lần tung một đồng xu không đối xứng, trong đó xác suất xuất hiện mặt Sấp là 0.6 và mặt Ngửa là 0.4. Mỗi lần tung được xem là một phép thử độc lập. Gọi Ω là không gian mẫu gồm tất cả các chuỗi độ dài 4 chỉ gồm ký tự ‘S’ (sấp) và ‘N’ (ngửa). Khi đó, Ω có $2^4 = 16$ phần tử. Ví dụ, chuỗi ‘SSNN’ là một phần tử trong Ω .

Xác suất của một chuỗi cụ thể được tính bằng cách nhân các xác suất thành phần. Ví dụ:

$$P(\text{SSNN}) = 0.6 \times 0.6 \times 0.4 \times 0.4 = 0.0576$$

Ta định nghĩa một biến ngẫu nhiên

$$X : \Omega \longrightarrow \{0, 1, 2, 3, 4\},$$

trong đó $X(\omega)$ là số lần xuất hiện mặt Sấp trong chuỗi ω . Đây là một biến ngẫu nhiên rời rạc, và có thể mô tả phân phối xác suất của nó dựa theo phân phối nhị thức có trọng số.

Ví dụ 2.2:

Giả sử B là sự kiện “xuất hiện ít nhất 3 lần mặt Sấp” trong trò chơi ở Ví dụ 2.1. Khi đó, hàm chỉ báo χ_B của sự kiện B được định nghĩa như sau:

$$\chi_B(\omega) = \begin{cases} 1, & \text{nếu } \omega \in B, \\ 0, & \text{nếu } \omega \notin B, \end{cases} \quad (2.2)$$

Hàm χ_B là một biến ngẫu nhiên nhận hai giá trị 0 và 1, trong đó:

- $\chi_B(\omega) = 1$ nếu chuỗi ω có ít nhất 3 ký tự là 'S';
- $\chi_B(\omega) = 0$ nếu không.

Ngược lại, nếu ta có một biến ngẫu nhiên

$$Y : \Omega \longrightarrow \{0, 1\},$$

thì tồn tại một sự kiện $A \subset \Omega$ sao cho $Y = \chi_A$, với

$$A = \{ \omega \in \Omega \mid Y(\omega) = 1 \}.$$

2.2 Biến ngẫu nhiên rời rạc

2.2.1 Định nghĩa

Một biến ngẫu nhiên rời rạc (Random discrete variable) chỉ có thể nhận một giá trị (hay không gian của chính nó) là hữu hạn hoặc đếm được.

2.2.2 Tính chất phân phối xác suất

Phân phối xác suất (probability distribution) đối với một biến ngẫu nhiên rời rạc có thể là hàm xác suất (probability mass function - PMF), bảng phân phối xác suất (probability distribution table - PMT) hay đồ thị biểu diễn hoặc đưa ra các giá trị có thể của x và xác suất tương ứng $p(x)$.

Tính chất của phân phối xác suất cho biến ngẫu nhiên rời rạc gồm có:

- Xác suất cho mỗi giá trị nằm giữa 0 và 1:

$$0 \leq p(x) \leq 1$$

- Tổng xác suất của tất cả các giá trị có thể bằng 1:

$$\sum p(x) = 1$$

2.2.3 Các tham số đặc trưng

- Giá trị kỳ vọng (expected value) hay trung bình (mean) của x được xác định:

$$\mu = E(x) = \sum xp(x) \quad (12)$$

- Phương sai (variance, Var) của x được xác định:

$$\sigma^2 = E[(x - \mu)^2] = \sum (x - \mu)^2 p(x) \quad (13)$$

- Độ lệch chuẩn (standard deviation, SD) của x là σ .

2.2.4 Một số phân phối xác suất phổ biến

- **Phân phối Bernoulli:** Mô hình kết quả của một thử nghiệm hay phép thử duy nhất chỉ có hai kết quả (thành công hoặc thất bại). Biến ngẫu nhiên $X = 1$ khi kết quả thành công và $X = 0$ khi kết quả thất bại. Ta có hàm phân phối xác suất (PMF):

$$p(0) = P\{X = 0\} = 1 - p = q, \quad p(1) = P\{X = 1\} = p \quad (14)$$

với $0 \leq p \leq 1$ là xác suất phép thử thành công.

Biến ngẫu nhiên X được gọi là biến ngẫu nhiên Bernoulli nếu PMF được cho bởi phương trình trên với $p \in (0, 1)$.

- **Nhị thức Binomial:** Giả định rằng có n phép thử độc lập mà kết quả thành công có xác suất là p , thất bại là $1 - p$. Nếu X đại diện cho số lần thành công xuất hiện trong n phép thử, khi đó X là biến ngẫu nhiên nhị thức Binomial với các tham số (n, p) .

Trường hợp đặc biệt, biến ngẫu nhiên Bernoulli có tham số là $(1, p)$.

Hàm xác suất của biến ngẫu nhiên nhị thức $X \sim (n, p)$ được xác định như sau:

$$p(i) = C_i^n \cdot p^i \cdot (1 - p)^{n-i}, \quad \text{với } i = 0, 1, \dots, n \quad (15)$$

Kỳ vọng (Mean):

$$E[X] = np \quad (16)$$

Phương sai (Var):

$$\text{Var}(X) = np(1-p) \quad (17)$$

Tính toán hàm phân phối nhị thức:

$$P\{X \leq i\} = \sum_{k=0}^i C_k^n \cdot p^k \cdot (1-p)^{n-k}, \quad \text{với } i = 0, 1, \dots, n \quad (18)$$

Mối quan hệ giữa $P\{X = k+1\}$ và $P\{X = k\}$ là:

$$P\{X = k+1\} = \left(\frac{p}{1-p}\right) \left(\frac{n-k}{k+1}\right) P\{X = k\} \quad (19)$$

- **Phân phối Poisson:** Cho biến ngẫu nhiên $X = \{0, 1, 2, \dots\}$ được gọi là biến ngẫu nhiên Poisson với tham số $\lambda > 0$ sao cho xác suất:

$$P\{X = i\} = \frac{e^{-\lambda} \lambda^i}{i!} \quad (19)$$

Phương trình trên được xác định dựa vào phương trình sau:

$$\sum_{i=0}^{\infty} P\{X = i\} = e^{-\lambda} \sum_{i=0}^{\infty} \frac{\lambda^i}{i!} = e^{-\lambda} e^{\lambda} = 1 \quad (20)$$

Biến ngẫu nhiên Poisson có thể được sử dụng như một phép biến đổi gần đúng cho biến ngẫu nhiên nhị thức có tham số (n, p) khi n lớn và p đủ nhỏ để np có kích thước vừa phải. Cụ thể, giả sử X là biến ngẫu nhiên nhị thức có tham số (n, p) và đặt $\lambda = np$ ta chứng minh được công thức gần đúng:

$$P\{X = i\} \approx \frac{e^{-\lambda} \lambda^i}{i!} \quad (21)$$

Kỳ vọng (Mean):

$$\mathbb{E}[X] = \lambda \quad (22)$$

Phương sai (Variance):

$$\text{Var}(X) = \mathbb{E}[X^2] - (\mathbb{E}[X])^2 = \lambda, \quad \text{với } \mathbb{E}[X^2] = \lambda(\lambda + 1) \quad (23)$$

Tính toán hàm phân phối Poisson:

Ứng dụng công thức gần đúng khi n lớn và các biến cố không độc lập, gọi E_i là biến cố thành công trong lần thử i với:

$$P\{E_i\} = \frac{1}{n}, \quad P\{E_i \mid E_j\} = \frac{1}{n-1}, \quad i \neq j \quad (24)$$

Mô hình Poisson: Giả định n biến cố, p_i là xác suất biến cố i xảy ra ($i = 1, \dots, n$), nếu tất cả p_i nhỏ và các phép thử độc lập hoặc phần lớn phụ thuộc yếu, thì số lượng biến cố xảy ra này xấp xỉ phân phối Poisson với kỳ vọng (trung bình) là:

$$\sum_{i=1}^n p_i$$

Mối quan hệ giữa xác suất $P\{X = i + 1\}$ và $P\{X = i\}$ được xác định bởi công thức:

$$P\{X = i + 1\} = \frac{\lambda}{i + 1} P\{X = i\} \quad (25)$$

- **Phân phối hình học (Geometric):** Gọi X là số phép thử thành công đầu tiên trong các phép thử Bernoulli với xác suất thành công $p \in (0, 1)$, ta có:

$$P\{X = n\} = p(1 - p)^{n-1}, \quad n = 1, 2, \dots \quad (26)$$

Kỳ vọng:

$$\mathbb{E}[X] = \frac{1}{p} \quad (27)$$

Phương sai:

$$\text{Var}(X) = \frac{1 - p}{p^2} \quad (28)$$

- **Phân phối nhị thức nghịch (Negative Binomial):** Gọi X là số phép thử cần thiết để thu được r lần thành công đầu tiên với xác suất thành công p , ta có:

$$P\{X = n\} = \binom{n-1}{r-1} p^r (1 - p)^{n-r}, \quad n = r, r + 1, \dots \quad (29)$$

Kỳ vọng:

$$\mathbb{E}[X] = \frac{r}{p} \quad (30)$$

Phương sai:

$$\text{Var}(X) = \frac{r(1-p)}{p^2}, \quad \mathbb{E}[X^2] = \frac{r}{p} \left(\frac{r+1}{p} - 1 \right) \quad (31)$$

- **Phân phối siêu bội (Hypergeometric):** Mô tả số lần thành công X trong mẫu kích thước n chọn từ quần thể kích thước N có M phần tử thành công (và $N - M$ thất bại), xác suất:

$$P\{X = i\} = \frac{\binom{M}{i} \binom{N-M}{n-i}}{\binom{N}{n}}, \quad i = 0, 1, \dots, n \quad (32)$$

Kỳ vọng:

$$\mathbb{E}[X] = np, \quad \text{với } p = \frac{M}{N} \quad (33)$$

Phương sai (xấp xỉ):

$$\text{Var}(X) \approx np(1-p) \quad (34)$$

- **Phân phối Zeta (Zipf):** Hàm xác suất:

$$P\{X = k\} = \frac{C}{k^{\alpha+1}}, \quad k = 1, 2, \dots \quad (35)$$

với hằng số chuẩn hoá:

$$C = \left[\sum_{k=1}^{\infty} \frac{1}{k^{\alpha+1}} \right]^{-1}$$

2.3 Biến ngẫu nhiên liên tục

2.3.1 Khái niệm

Một biến liên tục có thể nhận vô số giá trị tương ứng với các điểm trên một khoảng tuyến tính.

2.3.2 Hàm phân phối tích lũy

Hàm phân phối tích lũy, cumulative distribution function (CDF), của một biến ngẫu nhiên liên tục, ký hiệu là $F_X(x)$, là một hàm liên tục $\forall x \in \mathbb{R}$. Ta có các định lý sau:

Định lý 1: Các tính chất của hàm phân phối tích lũy $F(x)$:

- Hàm $F_X(x)$ là hàm không giảm: Nếu $a < b$ thì $F(a) \leq F(b)$
- Giới hạn dưới (bên trái) của $F_X(x)$ là 0: $\lim_{x \rightarrow -\infty} F_X(x) = 0$
- Giới hạn trên (bên phải) của $F_X(x)$ là 1: $\lim_{x \rightarrow +\infty} F_X(x) = 1$
- $F_X(x)$ liên tục bên phải: $\lim_{x \rightarrow x_0^+} F_X(x) = F_X(x_0)$

Định lý 2: Nếu $a < b$ thì

$$P[a < X \leq b] = F_X(b) - F_X(a)$$

Định lý 3: Với $\forall x \in \mathbb{R}$ và $F_X(x^-) = \lim_{z \rightarrow x^-} F_X(z)$ thì

$$P[X = x] = F_X(x) - F_X(x^-)$$

2.3.3 Tính chất phân phối xác suất

X là biến ngẫu nhiên liên tục, nếu tồn tại một hàm $f(x)$ không âm, được xác định với mọi số thực $x \in (-\infty, +\infty)$ và có tính chất đối với bất kỳ tập hợp các số thực B như sau:

$$P\{X \in B\} = \int_B f(x) dx \quad (36)$$

Trong đó $f(x)$ là hàm mật độ xác suất, probability density function (PDF).

Từ công thức (36) ta xét các trường hợp phân phối xác suất của biến ngẫu nhiên liên tục như sau:

Trường hợp $B = [a, b]$ ta được:

$$P\{a \leq X \leq b\} = \int_a^b f(x) dx \quad (37)$$

Có thể ứng dụng công thức trên để tính xác suất gần đúng của biến ngẫu nhiên gần một giá trị xác định cho trước a . Chẳng hạn, với ε nhỏ và hàm f liên tục tại $x = a$, ta tính được $P(X)$ như sau:

$$P\left\{a - \frac{\varepsilon}{2} \leq X \leq a + \frac{\varepsilon}{2}\right\} = \int_{a-\varepsilon/2}^{a+\varepsilon/2} f(x) dx \approx \varepsilon f(a) \quad (38)$$

Trường hợp $B = [a, a]$ ta được:

$$P\{X = a\} = \int_a^a f(x) dx = 0 \quad (39)$$

Trường hợp $B = (-\infty, a]$ ta được:

$$P\{X < a\} = P\{X \leq a\} = P\{X \in (-\infty, a]\} = F_X(a) = \int_{-\infty}^a f(x) dx \quad (40)$$

Trường hợp $B = (-\infty, +\infty)$ ta được:

$$P\{X \in (-\infty, +\infty)\} = \int_{-\infty}^{+\infty} f(x) dx = 1 \quad (41)$$

2.3.4 Các tham số đặc trưng

Kỳ vọng (trung bình) (Mean):

$$E[X] = \int_{-\infty}^{+\infty} x f(x) dx \quad (42)$$

2.3.5 Một số phân phối xác suất phổ biến

- **Phân phối đều (Uniform):**

Biến ngẫu nhiên có phân phối đều trên khoảng $(0, 1)$ có hàm mật độ (PDF) như sau:

$$f(x) = \begin{cases} 1, & 0 < x < 1 \\ 0, & \text{trường hợp khác} \end{cases} \quad (43)$$

Trường hợp tổng quát $x \in (\alpha, \beta)$ thì hàm mật độ có dạng:

$$f(x) = \begin{cases} \frac{1}{\beta - \alpha}, & \alpha < x < \beta \\ 0, & \text{trường hợp khác} \end{cases} \quad (44)$$

Kỳ vọng (trung bình) (Mean):

$$E[X] = \frac{\beta + \alpha}{2} \quad (45)$$

Phương sai (Var):

$$\text{Var}(X) = \frac{(\beta - \alpha)^2}{12} \quad (46)$$

- **Phân phối chuẩn (Normal) hay Gaussian:**

Biến ngẫu nhiên X là phân phối chuẩn với tham số kỳ vọng μ và phương sai σ^2 có hàm

mật độ xác suất là:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad -\infty < x < +\infty \quad (47)$$

Phân phối chuẩn là một trong những phân phối quan trọng nhất, thường mô tả các biến ngẫu nhiên là kết quả của nhiều yếu tố ngẫu nhiên độc lập. Hàm mật độ có dạng hình chuông đặc trưng bởi hai tham số: kỳ vọng μ và phương sai σ^2 .

- **Phân phối Chuẩn tắc (Standard Normal):**

Biến ngẫu nhiên x được chuẩn hoá bởi giá trị của độ lệch chuẩn nằm bên trái hoặc bên phải của giá trị trung bình μ . Ta có thể tính xác suất cho biến ngẫu nhiên chuẩn bằng cách chuẩn hóa thành biến chuẩn tắc:

$$z = \frac{x - \mu}{\sigma} \quad \text{hay} \quad x = \mu + z\sigma$$

Tính chất của biến chuẩn tắc:

- $x < \mu \implies z < 0$
- $x > \mu \implies z > 0$
- $x = \mu \implies z = 0$

- **Phân phối Chuẩn gần đúng với phân phối xác suất nhị thức:**

Định lý giới hạn DeMoivre–Laplace:

Nếu X_n biểu thị số lần thành công xảy ra trong n lần thử độc lập được thực hiện, mỗi lần thử thành công có xác suất p , thì với bất kỳ $a < b$, ta được:

$$P\left\{a \leq \frac{X_n - np}{\sqrt{np(1-p)}} \leq b\right\} \rightarrow \Phi(b) - \Phi(a), \quad n \rightarrow \infty \quad (48)$$

Phân phối xác suất của X gần đúng với đường cong phân phối chuẩn:

$$\mu = np \quad \text{và} \quad \sigma = \sqrt{np(1-p)}$$

- **Phân phối mũ (Exponential):**

Một biến ngẫu nhiên có phân phối mũ với hàm mật độ xác suất (PDF) như sau:

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad \text{với tham số } \lambda > 0 \quad (49)$$

Hàm phân phối tích lũy (CDF) $F(a)$ có dạng:

$$F(a) = P\{X \leq a\} = 1 - e^{-\lambda a}, \quad a \geq 0 \quad (50)$$

Kỳ vọng (trung bình) (Mean):

$$E[X^n] = \int_0^\infty x^n \lambda e^{-\lambda x} dx = \frac{n}{\lambda} E[X^{n-1}] \quad (51)$$

Phương sai (Var):

$$\text{Var}(X) = \frac{1}{\lambda^2} \quad (52)$$

2.4 Các định lý về giới hạn

2.4.1 Bất đẳng thức Markov's

Nếu X là một biến ngẫu nhiên có giá trị không âm thì ta có công thức xác suất:

$$P\{X \geq a\} \leq \frac{E[X]}{a} \quad \text{với } a > 0 \quad (53)$$

2.4.2 Bất đẳng thức Chebyshev's

Nếu X là một biến ngẫu nhiên có giá trị trung bình μ và phương sai σ^2 là hữu hạn thì với mọi $k > 0$ ta có công thức xác suất:

$$P\{|X - \mu| \geq k\} \leq \frac{\sigma^2}{k^2} \quad (54)$$

Hệ quả: nếu $\text{Var}(X) = 0$ thì $P\{X = E[X]\} = 1$

2.4.3 Luật số lớn yếu, the weak law of large numbers

Cho X_1, X_2, \dots là một dãy các biến ngẫu nhiên độc lập và phân phối giống hệt nhau, mỗi biến có trung bình hữu hạn $E[X_i] = \mu$. Khi đó, với bất kỳ $\varepsilon > 0$, ta được:

$$P\left\{\left|\frac{X_1 + \cdots + X_n}{n} - \mu\right| \geq \varepsilon\right\} \rightarrow 0 \quad (n \rightarrow \infty) \quad (55)$$

2.4.4 Định lý giới hạn trung tâm, the central limit theorem

Cho X_1, X_2, \dots là một dãy các biến ngẫu nhiên độc lập và có phân phối giống hệt nhau, mỗi biến có trung bình μ và phương sai σ^2 . Khi đó phân phối của

$$\frac{X_1 + \cdots + X_n - n\mu}{\sigma\sqrt{n}}$$

sẽ hướng đến phân phối chuẩn tắc khi $n \rightarrow \infty$. Nghĩa là, với $-\infty < a < +\infty$ thì:

$$P\left\{\left|\frac{X_1 + \cdots + X_n - n\mu}{\sigma\sqrt{n}}\right| \leq a\right\} \rightarrow \frac{1}{\sqrt{2\pi}} \int_{-\infty}^a e^{-x^2/2} dx \quad (n \rightarrow \infty) \quad (56)$$

Trường hợp X_i giới hạn đều và tồn tại M sao cho $P\{|X_i| < M\} = 1 \quad \forall i$, và $\sum_{i=1}^{\infty} \sigma_i^2 = \infty$, khi đó:

$$P\left\{\frac{\sum_{i=1}^n (X_i - \mu)}{\sqrt{\sum_{i=1}^n \sigma_i^2}} \leq a\right\} \rightarrow \Phi(a) \quad (n \rightarrow \infty) \quad (57)$$

2.4.5 Luật số lớn mạnh, the strong law of large numbers

Cho X_1, X_2, \dots là một dãy các biến ngẫu nhiên độc lập và phân phối giống hệt nhau, mỗi biến có trung bình hữu hạn $E[X_i] = \mu$. Khi đó, với xác suất là 1, ta được:

$$\frac{X_1 + X_2 + \cdots + X_n}{n} \rightarrow \mu \quad (n \rightarrow \infty)$$

Áp dụng luật số lớn mạnh, giả sử E là một biến cố cố định của thí nghiệm, $P(E)$ là xác suất E xảy ra trong lần thử nghiệm bất kỳ. Đặt:

$$X_i = \begin{cases} 1 & E \text{ xảy ra tại } i \text{ lần thử} \\ 0 & E \text{ không xảy ra tại } i \text{ lần thử} \end{cases}$$

Theo Luật số lớn mạnh thì với xác suất là 1, ta được:

$$\frac{X_1 + \cdots + X_n}{n} \rightarrow E[X] = P(E) \quad (58)$$

2.5 Phân phối xác suất đa biến

2.5.1 Xác suất đồng thời (Joint Probability)

Cho hai biến ngẫu nhiên A và B trên cùng một không gian xác suất (Ω, \mathbb{P}) . Xác suất đồng thời cho biết xác suất để A và B cùng đạt hai giá trị cụ thể đồng thời.

a/ Trường hợp rời rạc

Giả sử A chỉ nhận giá trị trong tập đếm được \mathcal{A} , và B chỉ nhận giá trị trong tập đếm được \mathcal{B} . Khi đó, hàm khối xác suất chung (joint probability mass function) của (A, B) được định nghĩa

$$p_{A,B}(a, b) = \mathbb{P}(A = a, B = b), \quad a \in \mathcal{A}, b \in \mathcal{B}.$$

Hàm khối xác suất chung phải thỏa mãn không âm và tổng bằng 1:

$$p_{A,B}(a, b) \geq 0 \quad \text{với mọi } a, b, \quad \sum_{a \in \mathcal{A}} \sum_{b \in \mathcal{B}} p_{A,B}(a, b) = 1.$$

Từ hàm khối xác suất chung, ta có thể rút ra hàm khối xác suất biên của A bằng cách tổng qua b :

$$p_A(a) = \sum_{b \in \mathcal{B}} p_{A,B}(a, b), \quad a \in \mathcal{A}.$$

Tương tự, hàm khối xác suất biên của B là

$$p_B(b) = \sum_{a \in \mathcal{A}} p_{A,B}(a, b), \quad b \in \mathcal{B}.$$

Vì $\{A = a, B = b\} \subseteq \{A = a\}$ nên luôn có

$$p_{A,B}(a, b) = \mathbb{P}(A = a, B = b) \leq \mathbb{P}(A = a) = p_A(a).$$

Tương tự, $p_{A,B}(a, b) \leq p_B(b)$.

Ví dụ 3.1:

Giả sử ta có hai biến ngẫu nhiên rời rạc A và B , mỗi biến chỉ nhận các giá trị trong tập đếm được. Cụ thể:

- A là số mặt xuất hiện khi ta tung một con xúc xắc (giá trị thuộc $\{1, 2, 3, 4, 5, 6\}$).

- B là biến chỉ ra “tung được số chẵn hay lẻ”:

$$B = \begin{cases} 0, & \text{nếu } A \text{ là số lẻ,} \\ 1, & \text{nếu } A \text{ là số chẵn.} \end{cases}$$

Hàm khối xác suất chung

Ta xây dựng

$$p_{A,B}(a, b) = P(A = a, B = b)$$

như sau:

1. Nếu $a \in \{1, \dots, 6\}$ và b không khớp với tính chẵn/lẻ của a , thì

$$p_{A,B}(a, b) = 0.$$

2. Nếu $a \in \{1, 3, 5\}$ (lẻ), thì phải có $B = 0$, và vì cân đối:

$$p_{A,B}(1, 0) = p_{A,B}(3, 0) = p_{A,B}(5, 0) = \frac{1}{6}, \quad p_{A,B}(1, 1) = p_{A,B}(3, 1) = p_{A,B}(5, 1) = 0.$$

3. Nếu $a \in \{2, 4, 6\}$ (chẵn), thì phải có $B = 1$:

$$p_{A,B}(2, 1) = p_{A,B}(4, 1) = p_{A,B}(6, 1) = \frac{1}{6}, \quad p_{A,B}(2, 0) = p_{A,B}(4, 0) = p_{A,B}(6, 0) = 0.$$

Kiểm tra tính chất

- $p_{A,B}(a, b) \geq 0$ với mọi (a, b) .
- Tổng xác suất:

$$\sum_{a=1}^6 \sum_{b=0}^1 p_{A,B}(a, b) = 3 \cdot \frac{1}{6} + 3 \cdot \frac{1}{6} = 1.$$

Hàm khối xác suất biên

- Biên A

$$p_A(a) = \sum_{b=0}^1 p_{A,B}(a, b).$$

Tính cho từng giá trị:

$$\begin{aligned} p_A(1) &= \frac{1}{6}, & p_A(2) &= \frac{1}{6}, & p_A(3) &= \frac{1}{6}, \\ p_A(4) &= \frac{1}{6}, & p_A(5) &= \frac{1}{6}, & p_A(6) &= \frac{1}{6}. \end{aligned}$$

Do đó, với mọi $a \in \{1, 2, 3, 4, 5, 6\}$,

$$p_A(a) = \frac{1}{6}.$$

• **Biên B :**

$$p_B(b) = \sum_{a=1}^6 p_{A,B}(a, b),$$

trong đó

$$p_B(0) = 3 \times \frac{1}{6} = 0.5, \quad p_B(1) = 3 \times \frac{1}{6} = 0.5.$$

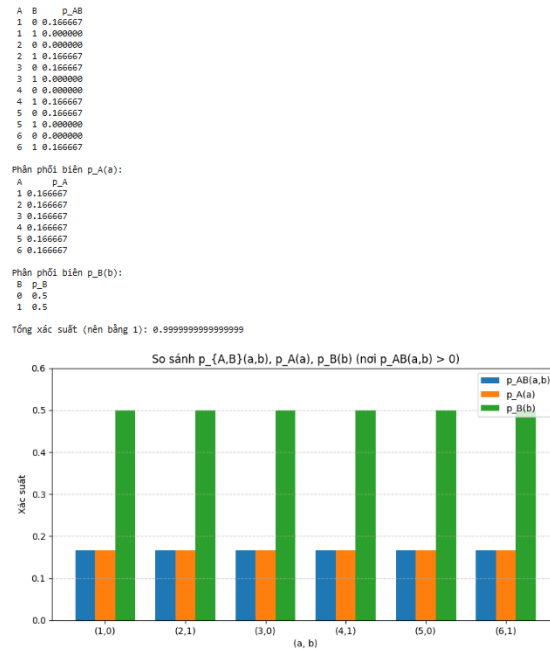
Bất đẳng thức biên

Với mọi (a, b) sao cho $p_{A,B}(a, b) > 0$, luôn có

$$p_{A,B}(a, b) \leq p_A(a), \quad p_{A,B}(a, b) \leq p_B(b).$$

Ví dụ với $a = 2, b = 1$: $p_{A,B}(2, 1) = \frac{1}{6}$, $p_A(2) = \frac{1}{6}$, $p_B(1) = 0.5$.

Ta sẽ demo ví dụ này bằng python (xem code demo cụ thể trên notebook)



Hình 3: Kết quả demo cho ví dụ 3.1

b/ Trường hợp liên tục

Giả sử A và B là hai biến ngẫu nhiên liên tục (định nghĩa trên \mathbb{R}). Khi đó, tồn tại hàm mật độ chung (joint probability density function) $f_{A,B}(a, b)$ sao cho với mọi miền $D \subset \mathbb{R}^2$,

$$\mathbb{P}((A, B) \in D) = \iint_D f_{A,B}(a, b) da db.$$

Hàm mật độ này phải không âm và tích phân trên toàn \mathbb{R}^2 bằng 1:

$$f_{A,B}(a, b) \geq 0 \quad \text{với mọi } (a, b) \in \mathbb{R}^2, \quad \iint_{\mathbb{R}^2} f_{A,B}(a, b) da db = 1.$$

Từ hàm mật độ chung, ta lấy mật độ biên của A bằng cách tích phân qua biến b :

$$f_A(a) = \int_{-\infty}^{+\infty} f_{A,B}(a, b) db, \quad a \in \mathbb{R},$$

và mật độ biên của B bằng cách tích phân qua biến a :

$$f_B(b) = \int_{-\infty}^{+\infty} f_{A,B}(a, b) da, \quad b \in \mathbb{R}.$$

Mặc dù $f_{A,B}(a, b)$ không phải là xác suất trực tiếp mà là mật độ, ta vẫn có quan hệ cận trên:

vì tích phân qua một biến phải bằng mật độ biên của biến kia, nên [6]

$$\int_{-\infty}^{+\infty} f_{A,B}(a,b) db = f_A(a) \implies f_{A,B}(a,b) \leq f_A(a), \quad f_{A,B}(a,b) \leq f_B(b).$$

Ví dụ 3.2: Xét hai biến ngẫu nhiên liên tục A và B đại diện cho tọa độ của một điểm được chọn ngẫu nhiên trên mặt phẳng.

Giả thiết:

- Điểm (A, B) được chọn ngẫu nhiên và đều (uniform) trên hình chữ nhật D trong mặt phẳng:

$$D = \{(a, b) \in \mathbb{R}^2 \mid 0 \leq a \leq 2, 1 \leq b \leq 3\}.$$

- Nói cách khác, xác suất điểm rơi vào một vùng con bất kỳ trong D tỉ lệ với diện tích của vùng đó.

Xét hai biến ngẫu nhiên liên tục:

- A : tọa độ x của một điểm ngẫu nhiên, chỉ nhận giá trị trong khoảng $[0, 2]$.
- B : tọa độ y của cùng điểm đó, chỉ nhận giá trị trong khoảng $[1, 3]$.

Hai biến (A, B) phân bố đều (uniform) trên miền hình chữ nhật

$$D = \{(a, b) \in \mathbb{R}^2 \mid 0 \leq a \leq 2, 1 \leq b \leq 3\}.$$

Điều này có nghĩa là xác suất rơi vào bất cứ vùng con nào của D tỉ lệ với diện tích của vùng con đó.

1. Hàm mật độ chung

Với $(a, b) \in D$:

$$f_{A,B}(a,b) = \frac{1}{\text{area}(D)} = \frac{1}{(2-0) \times (3-1)} = \frac{1}{4}.$$

Với $(a, b) \notin D$:

$$f_{A,B}(a,b) = 0.$$

2. Mật độ biên

- **Biên của A** Lấy tích phân theo b :

$$f_A(a) = \int_{-\infty}^{\infty} f_{A,B}(a,b) db = \int_1^3 \frac{1}{4} db = 0.5, \quad 0 \leq a \leq 2,$$

và $f_A(a) = 0$ nếu $a \notin [0, 2]$.

- **Biên của B** Lấy tích phân theo a :

$$f_B(b) = \int_{-\infty}^{\infty} f_{A,B}(a,b) da = \int_0^2 \frac{1}{4} da = 0.5, \quad 1 \leq b \leq 3,$$

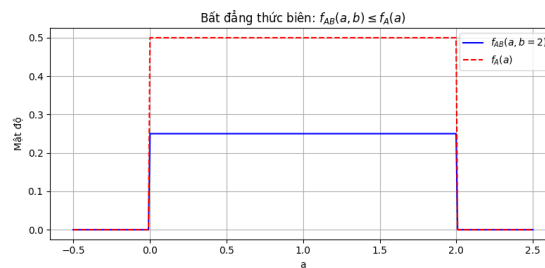
và $f_B(b) = 0$ nếu $b \notin [1, 3]$.

3. Bất đẳng thức biên

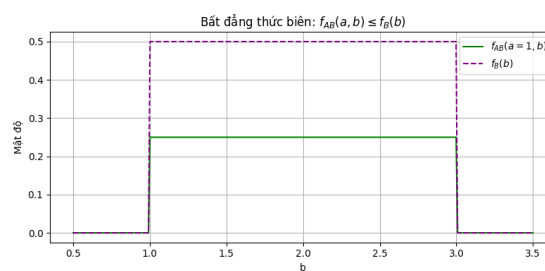
Trên D ta có $f_{A,B}(a,b) = 0.25$ và $f_A(a) = f_B(b) = 0.5$, nên với mọi $(a,b) \in D$:

$$f_{A,B}(a,b) \leq f_A(a), \quad f_{A,B}(a,b) \leq f_B(b).$$

Ta sẽ demo ví dụ này bằng python (xem demo code ví dụ cụ thể trên notebook)



Hình 4: Kết quả demo ví dụ 3.2 cho biên của A



Hình 5: Kết quả demo ví dụ 3.2 cho biên của B

2.5.2 Xác suất có điều kiện (Conditional Probability)

a/ Định nghĩa

Giả sử (trong một không gian xác suất nào đó) điều kiện B có xác suất khác không, $P(B) > 0$. Khi đó, **xác suất của sự kiện A dưới điều kiện B** , ký hiệu là

$$P(A | B),$$

được định nghĩa như sau [6]:

$$P(A | B) = \frac{P(A \cap B)}{P(B)}. \quad (1.18)$$

Một hệ quả trực tiếp của định nghĩa trên là **công thức nhân xác suất** [6]:

$$P(A \cap B) = P(A | B) \cdot P(B). \quad (1.19)$$

Ví dụ 3.3:

Trong một chiếc hộp có 5 viên bi, gồm 2 viên đỏ và 3 viên xanh. Ta thực hiện hai lần rút ngẫu nhiên không trả lại.

- B : “viên bi đầu tiên rút được là đỏ.”
- A : “viên bi thứ hai rút được là đỏ.”

1. Xác suất để lần rút đầu tiên là đỏ:

$$P(B) = \frac{2}{5}.$$

2. Nếu lần đầu đã rút được bi đỏ, trong hộp còn 1 viên đỏ và 3 viên xanh, nên xác suất lần hai rút được bi đỏ (điều kiện B) là

$$P(A | B) = \frac{1}{4}.$$

3. Do đó, xác suất cùng lúc “lần 1 và lần 2 đều rút đỏ” là

$$P(A \cap B) = P(A | B)P(B) = \frac{1}{4} \times \frac{2}{5} = \frac{1}{10}.$$

Nếu tính trực tiếp “cùng lúc lần 1 và lần 2 đều đỏ” theo lối rút nối tiếp cũng cho kết quả $\frac{1}{10}$.

b/ Độc lập và phụ thuộc

Hai sự kiện A và B được gọi là **độc lập** nếu việc biết B có xảy ra hay không không ảnh hưởng đến xác suất xảy ra của A . Ngược lại, nếu việc biết B xảy ra làm thay đổi xác suất của A , thì A và B **phụ thuộc**.

Định nghĩa 1.6 [6]:

Hai sự kiện A và B được gọi là **độc lập** nếu và chỉ nếu

$$P(A | B) = P(A), \quad \text{tương đương với} \quad P(A \cap B) = P(A)P(B). \quad (1.20)$$

Nếu hai sự kiện không thỏa mãn $P(A \cap B) = P(A)P(B)$ tức là không độc lập với nhau thì ta nói chúng **phụ thuộc** nhau [6]. Khi đó:

$$\begin{cases} P(A | B) > P(A) & \text{nếu } B \text{ thuận lợi cho } A, \\ P(A | B) < P(A) & \text{nếu } B \text{ không thuận lợi (bất lợi) cho } A. \end{cases}$$

Vì tính chất đối xứng ta có công thức tương đương:

$$P(A | B) P(B) = P(B | A) P(A) \implies \frac{P(A | B)}{P(A)} = \frac{P(B | A)}{P(B)}. \quad (1.23)$$

Ví dụ 3.4:

1. Hai lần tung đồng xu cân đối

- A : “lần tung thứ nhất ra Ngửa (N).”
- B : “lần tung thứ hai ra Ngửa (N).”

Vì hai lần tung hoàn toàn độc lập, ta có:

$$P(A) = 0.5, \quad P(B) = 0.5, \quad P(A \cap B) = 0.5 \times 0.5 = 0.25,$$

và $P(A)P(B) = 0.25$.

Do đó $P(A \cap B) = P(A)P(B)$, nên A và B độc lập.

Tiến hành demo cho bài toán bằng python (xem code demo chi tiết trên notebook)

```
== Tung đồng xu hai lần ==  
P(A) = 0.5  
P(B) = 0.5  
P(A n B) = 0.25  
P(A) * P(B) = 0.25  
→ A và B độc lập: True
```

Hình 6: Kết quả demo cho ví dụ 3.4(1)

2. Rút hai viên bi không trả từ túi có 2 viên đỏ và 1 viên xanh.

- C : “viên bi thứ nhất rút ra là xanh.”
- D : “viên bi thứ hai rút ra là xanh.”

Xác suất C xảy ra là

$$P(C) = \frac{1}{3}.$$

Nếu lần một đã rút được viên bi xanh thì trong túi không còn viên xanh nào, nên

$$P(D | C) = 0.$$

Khi đó,

$$P(C \cap D) = P(C)P(D | C) = \frac{1}{3} \times 0 = 0,$$

trong khi

$$P(C)P(D) = \frac{1}{3} \times \frac{1}{3} = \frac{1}{9} \neq 0.$$

Vậy $P(C \cap D) \neq P(C)P(D)$, hai sự kiện C và D **phụ thuộc** nhau.

Chạy demo cho ví dụ trên bằng python (Xem code demo chi tiết trên notebook)

```
== Rút bi không hoàn lại từ túi (2 Đ, 1 X) ==  
P(C) = 0.3333333333333333  
P(D) = 0.3333333333333333  
P(C n D) = 0.0  
P(C) * P(D) = 0.1111111111111111  
→ C và D phụ thuộc: True
```

Hình 7: Kết quả demo cho ví dụ 3.4(2)

Ví dụ 3.5:

Giả sử trong một lớp học, ta có:

$$P(A) = 0.1 \quad (10\%), \quad P(B) = 0.2 \quad (20\%),$$

trong đó:

- A : “mang ô”,
- B : “trời mưa”.

Người ta quan sát được xác suất điều kiện:

$$P(A | B) = 0.8 \quad (80\%).$$

Do $P(A | B) > P(A)$, ta kết luận “trời mưa thuận lợi cho việc mang ô”.

Ngược lại, tính $P(B | A)$:

$$P(B | A) = \frac{P(A \cap B)}{P(A)} = \frac{P(A | B) \cdot P(B)}{P(A)} = \frac{0.8 \times 0.2}{0.1} = 1.6 > P(B).$$

Điều này cho thấy “mang ô cũng thuận lợi cho sự kiện trời mưa”.

Từ đó, ta khẳng định công thức (1.23) và tính đối xứng của mối quan hệ phụ thuộc giữa hai sự kiện A và B :

$$\frac{P(A | B)}{P(A)} = \frac{P(B | A)}{P(B)}.$$

Định nghĩa 1.7 [6]: Một họ \mathcal{M} được gọi là **họ các sự kiện độc lập** nếu với mọi số tự nhiên k và mọi k sự kiện khác nhau A_1, A_2, \dots, A_k trong \mathcal{M} , ta có:

$$P\left(\bigcap_{i=1}^k A_i\right) = \prod_{i=1}^k P(A_i). \quad (1.22)$$

Ví dụ 3.6: *Ba lần tung đồng xu cân đối.*

Xét ba sự kiện:

$$A_1 : \text{“lần tung 1 ra Ngửa”}, \quad A_2 : \text{“lần tung 2 ra Ngửa”}, \quad A_3 : \text{“lần tung 3 ra Ngửa”}.$$

Không gian mẫu của ba lần tung đồng xu là $\{0, 1\}^3$, gồm 8 kết quả, mỗi kết quả có xác suất $\frac{1}{8}$.

- $P(A_1) = P(A_2) = P(A_3) = \frac{4}{8} = 0.5$.
- $P(A_1 \cap A_2) = \frac{2}{8} = 0.25 = P(A_1)P(A_2)$.
- $P(A_1 \cap A_3) = \frac{2}{8} = 0.25 = P(A_1)P(A_3)$.
- $P(A_2 \cap A_3) = \frac{2}{8} = 0.25 = P(A_2)P(A_3)$.
- $P(A_1 \cap A_2 \cap A_3) = \frac{1}{8} = 0.125 = P(A_1)P(A_2)P(A_3)$.

Do mọi giao (cặp hoặc ba) có xác suất đúng bằng tích các xác suất riêng lẻ, ta kết luận bộ $\{A_1, A_2, A_3\}$ là một họ ba sự kiện **độc lập toàn phần**.

Ta sẽ demo ví dụ này bằng python (xem code demo cụ thể trên notebook)

```
P(A1) = 0.5
P(A2) = 0.5
P(A3) = 0.5

P(A1 n A2) = 0.25 vs P(A1)P(A2) = 0.25
P(A1 n A3) = 0.25 vs P(A1)P(A3) = 0.25
P(A2 n A3) = 0.25 vs P(A2)P(A3) = 0.25

P(A1 n A2 n A3) = 0.125 vs P(A1)P(A2)P(A3) = 0.125
```

Hình 8: Kết quả demo ví dụ 3.6

2.5.3 Xác suất toàn phần (Total Probability)

Định nghĩa 1.8 [6]:

Một họ các tập con B_1, B_2, \dots, B_n của không gian xác suất Ω được gọi là một *phân hoạch* nếu:

$$B_i \cap B_j = \emptyset \quad \text{với } i \neq j, \quad \bigcup_{i=1}^n B_i = \Omega. \quad (1.24)$$

Công thức xác suất toàn phần [6]: Giả sử A là một biến cố và B_1, \dots, B_n là một phân hoạch của không gian mẫu. Khi đó:

$$P(A) = \sum_{i=1}^n P(A \cap B_i) = \sum_{i=1}^n P(A | B_i) P(B_i). \quad (1.25)$$

Trường hợp đặc biệt với hai thành phần B và \overline{B} , ta có:

$$P(A) = P(A | B) P(B) + P(A | \overline{B}) P(\overline{B}). \quad (1.26)$$

Ví dụ 3.7: Giả sử trong một lớp học:

$$P(A | \overline{B}) = 0.1, \quad P(B) = 0.2, \quad P(A | B) = 0.8,$$

trong đó:

- A : “mang ô”,
- B : “trời mưa”.

Áp dụng công thức (1.26):

$$\begin{aligned} P(A) &= P(A | B) P(B) + P(A | \overline{B}) P(\overline{B}) \\ &= 0.8 \times 0.2 + 0.1 \times (1 - 0.2) \\ &= 0.16 + 0.08 = 0.24. \end{aligned}$$

Vậy xác suất một người bất kỳ trong lớp mang ô là 0.24 (24%).

2.5.4 Phân phối xác suất của hai biến ngẫu nhiên

2.5.1.1 Định nghĩa véc tơ ngẫu nhiên

Cho một phép thử ngẫu nhiên với không gian mẫu S và hai biến ngẫu nhiên X_1, X_2 , với mỗi phần tử $s \in S$ sao cho $X_1(s) = x_1$ và $X_2(s) = x_2$ khi đó (X_1, X_2) là một véc tơ ngẫu nhiên, random vector. Không gian của (X_1, X_2) là một tập:

$$D = \{(x_1, x_2) : x_1 = X_1(s), x_2 = X_2(s), s \in S\}.$$

2.5.1.2 Phân phối xác suất

Với hàm phân phối xác suất tích lũy (CDF) có dạng:

$$F_{X_1, X_2}(x_1, x_2) = P[\{X_1 \leq x_1\} \cap \{X_2 \leq x_2\}] \quad \forall (x_1, x_2) \in \mathbb{R}^2 \quad (59)$$

Với các tập có dạng $(a_1, b_1] \times (a_2, b_2]$ thì CDF trở thành hàm phân phối xác suất tích lũy chung, joint cumulative distribution function, của (X_1, X_2) có dạng:

$$P[a_1 \leq X_1 \leq b_1, a_2 \leq X_2 \leq b_2] = F_{X_1, X_2}(b_1, b_2) - F_{X_1, X_2}(a_1, b_2) - F_{X_1, X_2}(a_2, b_1) + F_{X_1, X_2}(a_1, a_2) \quad (60)$$

Trường hợp (X_1, X_2) là véc tơ ngẫu nhiên rời rạc, discrete random vector, thì X_1, X_2 là biến ngẫu nhiên rời rạc và không gian S là hữu hạn hay đếm được. Và hàm xác suất chung, joint probability mass function, được xác định:

$$p_{X_1, X_2}(x_1, x_2) = P[X_1 = x_1, X_2 = x_2] \quad \forall (x_1, x_2) \in S \quad (61)$$

2.5.1.3 Tham số

Kỳ vọng: giả định rằng $Y = g(X_1, X_2)$ với $g : \mathbb{R}^2 \rightarrow \mathbb{R}$:

Trường hợp (X_1, X_2) là véc tơ ngẫu nhiên rời rạc, luôn tồn tại:

$$E[Y] = \sum_{x_1} \sum_{x_2} g(x_1, x_2) p_{X_1, X_2}(x_1, x_2) \quad (62)$$

Trường hợp (X_1, X_2) là véc tơ ngẫu nhiên liên tục, luôn tồn tại:

$$E[Y] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} g(x_1, x_2) f_{X_1, X_2}(x_1, x_2) dx_1 dx_2 \quad (63)$$

2.5.1.4 Hàm tạo sinh động lượng

Đặt $X = (X_1, X_2)'$ là véc tơ ngẫu nhiên. Nếu tồn tại $E(e^{t_1 X_1 + t_2 X_2})$ với $|t_1| < h_1$ và $|t_2| < h_2$ ($h_1, h_2 > 0$), ta gọi $M_{X_1, X_2}(t_1, t_2)$ là Hàm tạo sinh động lượng, moment generating function, của X . Đặt $t = (t_1, t_2)'$ ta được:

$$M_{X_1, X_2}(t) = E[e^{t'X}] \quad (64)$$

Kỳ vọng:

$$E[X] = \begin{pmatrix} E(X_1) \\ E(X_2) \end{pmatrix} \quad (65)$$

2.5.1.5 Định lý

Với (X_1, X_2) là véc tơ ngẫu nhiên, $Y_1 = g_1(X_1, X_2)$ và $Y_2 = g_2(X_1, X_2)$, $k_1, k_2 \in \mathbb{R}$, ta được:

$$E(k_1 Y_1 + k_2 Y_2) = k_1 E(Y_1) + k_2 E(Y_2) \quad (66)$$

Với (X_1, X_2) là véc tơ ngẫu nhiên và $\text{Var}(X_2)$ là hữu hạn, ta được:

$$E[E(X_2 | X_1)] = E(X_2) \quad (67)$$

$$\text{Var}[E(X_2 | X_1)] \leq \text{Var}(X_2) \quad (68)$$

2.5.5 Hiệp phương sai và hệ số tương quan

Cho hai biến ngẫu nhiên X và Y có hàm phân phối xác suất chung là $f(x, y)$. Nếu $u(x, y)$ là hàm theo x và y thì $E[u(X, Y)]$ được định nghĩa tồn tại.

Giả định các giá trị trung bình của X, Y là μ_1, μ_2 luôn tồn tại, có được từ hàm $u(x, y)$ và phương sai của X, Y là σ_1^2, σ_2^2 tồn tại từ việc đặt $u(x, y)$ bằng $(x - \mu_1)^2$ và $(y - \mu_2)^2$. Ta có kỳ vọng toán:

$$E[(X - \mu_1)(Y - \mu_2)] = E(XY) - \mu_1 \mu_2 \quad (69)$$

Gọi $\text{Corr}(X, Y)$ là hệ số tương quan, Correlation coefficient, và $\text{Cov}(X, Y)$ là hiệp phương sai, Covariance, ta được:

$$\text{Corr}(X, Y) = \frac{E[(X - \mu_1)(Y - \mu_2)]}{\sigma_1 \sigma_2} = \frac{\text{Cov}(X, Y)}{\sigma_1 \sigma_2} \quad (70)$$

Từ phương trình (69), (70) ta được:

$$E(XY) = \mu_1 \mu_2 + \text{Corr}(X, Y) \sigma_1 \sigma_2 = \mu_1 \mu_2 + \text{Cov}(X, Y) \quad (72)$$

Định lý: Nếu $E(Y | X)$ là tuyến tính theo X thì

$$E(Y | X) = \mu_2 + \text{Corr}(X, Y) \frac{\sigma_2}{\sigma_1} (X - \mu_1) \quad (73)$$

$$E[\text{Var}(Y | X)] = \sigma_2^2 [1 - (\text{Corr}(X, Y))^2] \quad (74)$$

3 Ứng dụng trong Khoa học Máy tính

Xác suất là nền tảng cho nhiều kỹ thuật và thuật toán trong computer science:

1. Thuật toán ngẫu nhiên (Randomized Algorithms)

- **Randomized QuickSort:** Chọn phần tử pivot ngẫu nhiên để giảm thiểu xác suất gặp trường hợp xấu.
- **Bloom Filter:** Dùng nhiều hàm băm giả ngẫu nhiên để kiểm tra sự tồn tại phần tử với độ sai lệch có thể tính được xác suất.

2. Machine Learning & Thống kê

- **Naive Bayes:** Dùng định lý Bayes

$$P(C|x) \approx P(x|C)P(C)$$

để phân loại nhãn dựa trên xác suất.

- **Bayesian Networks:** Mô hình biểu diễn quan hệ phụ thuộc giữa các biến ngẫu nhiên bằng đồ thị có hướng.

3. Mô phỏng Monte Carlo

- Tính gần đúng tích phân hoặc thống kê của hệ thống phức tạp (tích phân đa chiều, mô phỏng vật lý hạt).
- Ứng dụng trong đồ họa (path tracing) và tài chính (định giá quyền chọn).

4. Lý thuyết thông tin & mã hóa

- **Entropy:**

$$H(X) = - \sum_x P(x) \log P(x)$$

đo độ không chắc chắn của biến ngẫu nhiên X .

- **Huffman Coding:** Xây cây mã ưu tiên các kết quả có xác suất cao để giảm chiều dài trung bình mã.

Với mọi ứng dụng trên, hiểu rõ biến cố và tính chất của xác suất (như không âm, tổng bằng 1, cộng cho biến cố tách biệt) giúp phân tích, thiết kế và đánh giá hiệu quả các thuật toán và mô hình.

4 Bài toán thực tế

Chúng ta sẽ thử nghiệm một ví dụ với những nội dung đã được đề cập ở trên. Giả sử rằng một bác sĩ phụ trách xét nghiệm AIDS cho một bệnh nhân. Việc xét nghiệm này khá chính xác và nó chỉ thất bại với xác suất 1%, khi nó cho kết quả dương tính dù bệnh nhân khỏe mạnh. Hơn nữa, nó không bao giờ thất bại trong việc phát hiện HIV nếu bệnh nhân thực sự bị nhiễm bệnh. Ta sử dụng D_1 để biểu diễn kết quả chẩn đoán (1 nếu dương tính và 0 nếu âm tính) và H để biểu thị tình trạng nhiễm HIV (1 nếu dương tính và 0 nếu âm tính). Ta có liệt kê xác suất có điều kiện theo bảng dưới đây:

Bảng 1: Xác suất có điều kiện của $P(D_1 | H)$.

Xác suất có điều kiện	$H = 1$	$H = 0$
$P(D_1 = 1 H)$	1	0.01
$P(D_1 = 0 H)$	0	0.99

Ta có thể nhận thấy rằng tổng của từng cột đều bằng 1 (nhưng tổng từng hàng thì không), vì xác suất có điều kiện cần có tổng bằng 1. Hãy cùng tìm xác suất bệnh nhân bị AIDS nếu xét nghiệm trả về kết quả dương tính, tức là

$$P(H = 1 | D = 1)$$

Rõ ràng điều này sẽ phụ thuộc vào mức độ phổ biến của bệnh, bởi vì nó ảnh hưởng đến số lượng dương tính giả. Giả sử rằng dân số khá khỏe mạnh, ví dụ: $P(H = 1) = 0.0015$ Để áp dụng Định lý Bayes, chúng ta cần áp dụng phép biên hóa và quy tắc nhân để xác định

$$\begin{aligned} P(D_1 = 1) &= P(D_1 = 1, H = 0) + P(D_1 = 1, H = 1) \\ &= P(D_1 = 1 | H = 0)P(H = 0) + P(D_1 = 1 | H = 1)P(H = 1) \\ &= 0.01 \times 0.985 + 1 \times 0.015 \\ &= 0.011485. \end{aligned}$$

Do đó, ta có:

$$\begin{aligned}P(H = 1 \mid D_1 = 1) &= \frac{P(D_1 = 1 \mid H = 1)P(H = 1)}{P(D_1 = 1)} \\&= \frac{1 \times 0.015}{0.011485} \\&= 0.1306051\end{aligned}$$

Nói cách khác, chỉ có xấp xỉ 13,06% khả năng bệnh nhân thực sự mắc bệnh AIDS, dù ta dùng một bài kiểm tra rất chính xác. Như ta có thể thấy, xác suất có thể trở nên khá phản trực giác do trước khi xét nghiệm, trực giác chỉ ra có 0.15% khả năng nhiễm bệnh. Một bệnh nhân phải làm gì nếu nhận được tin dữ như vậy? Nhiều khả năng họ sẽ yêu cầu bác sĩ thực hiện một xét nghiệm khác để làm rõ sự việc. Giả sử bài kiểm tra thứ hai có những đặc điểm khác và không tốt bằng bài thứ nhất, như ta có thể thấy như sau:

Bảng 2: Xác suất có điều kiện của $P(D_2 \mid H)$.

Xác suất có điều kiện	$H = 1$	$H = 0$
$P(D_2 = 1 \mid H)$	0.98	0.03
$P(D_2 = 0 \mid H)$	0.02	0.97

Không may thay, bài kiểm tra thứ hai cũng có kết quả dương tính. Hãy cùng tính các xác suất cần thiết để sử dụng định lý Bayes bằng cách giả định tính độc lập có điều kiện:

$$\begin{aligned}P(D_1 = 1, D_2 = 1 \mid H = 0) &= P(D_1 = 1 \mid H = 0)P(D_2 = 1 \mid H = 0) \\&= 0.01 \times 0.03 \\&= 0.0003,\end{aligned}$$

$$\begin{aligned}P(D_1 = 1, D_2 = 1 \mid H = 1) &= P(D_1 = 1 \mid H = 1)P(D_2 = 1 \mid H = 1) \\&= 1 \times 0.98 \\&= 0.98.\end{aligned}$$

Bây giờ chúng ta có thể áp dụng phép biến hóa và quy tắc nhân xác suất:

$$\begin{aligned}P(D_1 = 1, D_2 = 1) &= P(D_1 = 1, D_2 = 1, H = 0) + P(D_1 = 1, D_2 = 1, H = 1) \\&= P(D_1 = 1, D_2 = 1 \mid H = 0)P(H = 0) + P(D_1 = 1, D_2 = 1 \mid H = 1)P(H = 1) \\&= 0.0003 \times 0.985 + 0.98 \times 0.015 \\&= 0.00176955\end{aligned}$$

Cuối cùng xác suất bệnh nhân mắc bệnh AIDS qua hai lần dương tính là

$$\begin{aligned}P(H = 1 \mid D_1 = 1, D_2 = 1) &= \frac{P(D_1 = 1, D_2 = 1 \mid H = 1)P(H = 1)}{P(D_1 = 1, D_2 = 1)} \\&= \frac{0.98 \times 0.015}{0.00176955} \\&\approx 0.8307.\end{aligned}$$

Nhận xét: Xét nghiệm thứ hai đã cho phép chúng ta đạt được mức độ tin cậy cao hơn nhiều rằng có điều gì đó không ổn. Mặc dù xét nghiệm thứ hai kém chính xác hơn đáng kể so với xét nghiệm đầu tiên, nhưng nó vẫn cải thiện đáng kể ước lượng của chúng ta. Giả định rằng hai xét nghiệm độc lập có điều kiện với nhau là yếu tố then chốt giúp chúng ta đưa ra ước lượng chính xác hơn. Hãy xét một trường hợp cực đoan khi chúng ta thực hiện cùng một xét nghiệm hai lần. Trong tình huống này, ta kỳ vọng kết quả sẽ giống nhau ở cả hai lần, do đó không có thông tin mới nào được rút ra từ việc lặp lại cùng một xét nghiệm. Chúng ta có thể nhận ra rằng quá trình chẩn đoán hoạt động giống như một bộ phân loại (classifier) đang "ẩn mình", khả năng của chúng ta trong việc xác định bệnh nhân có khỏe mạnh hay không sẽ tăng lên khi chúng ta thu thập thêm các đặc trưng (kết quả xét nghiệm).

5 Hàm mật độ xác suất, Hàm phân phối tích lũy

5.1 Hàm Mật độ Xác suất

Hàm mật độ xác suất (Probability Density Function hay PDF) dùng để biểu diễn một phân bố xác suất theo tích phân. Gọi $p(x)$ là một hàm mật độ xác suất, vì xác suất không bao giờ âm, do đó: $p(x) \geq 0$

Hơn nữa, Ta có:

$$P(X \in \mathbb{R}) = 1$$

và

$$P(X \in \mathbb{R}) = \int_{-\infty}^{\infty} p(x) dx.$$

Vì biến ngẫu nhiên này phải nhận một giá trị nào đó trong tập số thực, do đó ta có thể kết luận rằng với bất kỳ hàm mật độ nào thì:

$$\int_{-\infty}^{\infty} p(x) dx = 1$$

Đi sâu hơn vào phương trình trên, ta thấy rằng với bất kỳ a và b nào thì:

$$P(X \in (a, b]) = \int_a^b p(x) dx. \quad (1.5.1)$$

Công thức này dùng để tính xác suất biến ngẫu nhiên nằm trong một khoảng cụ thể.

5.2 Hàm Phân phối Tích lũy

Trong phần trước, nhóm đã trình bày về hàm mật độ xác suất (PDF). Trong thực tế, đây là một phương pháp thường dùng để thảo luận về các biến ngẫu nhiên liên tục, nhưng nó có một nhược điểm khá lớn: bản thân các giá trị của PDF không phải là các giá trị xác suất, mà ta phải tích phân hàm này để có xác suất. Không có gì sai với một hàm mật độ lớn hơn 10, miễn là nó không lớn hơn 10 trong khoảng có chiều dài lớn hơn 1/10. Điều này có thể hơi phản trực giác, do đó người ta thường dùng hàm phân phối tích lũy - Cumulative Distribution Function hay CDF, mà có giá trị trả về là xác suất. Hàm phân phối tích lũy mô tả đầy đủ phân phối xác suất của một biến ngẫu nhiên giá trị thực X . Với mỗi số thực x , hàm phân phối tích lũy được định nghĩa như sau:

$$F(x) = P(X \leq x).$$

Với một hàm phân phối tích lũy $F(x)$ tương ứng với một hàm mật độ xác suất $p(x)$ được định nghĩa là:

$$F(x) = \int_{-\infty}^x p(x) dx.$$

Với việc sử dụng công thức (1.5.1) ở trên, ta định nghĩa CDF cho một biến ngẫu nhiên X với mật độ $p(x)$ như sau:

$$F(x) = \int_{-\infty}^x p(x) dx = P(X \leq x).$$

Một vài tính chất của hàm phân phối tích lũy:

- $F(x) \rightarrow 0$ khi $x \rightarrow -\infty$.
- $F(x) \rightarrow 1$ khi $x \rightarrow \infty$.
- $F(x)$ không giảm ($y > x \implies F(y) \geq F(x)$).
- $F(x)$ liên tục (không có bước nhảy) nếu X là một biến ngẫu nhiên liên tục.

6 Kỳ vọng, phương sai, độ lệch chuẩn

6.1 Kỳ Vọng (Expectations)

Thông thường, việc ra quyết định không chỉ yêu cầu xem xét các xác suất được gán cho từng sự kiện riêng lẻ mà còn cần tổng hợp chúng thành các đại lượng hữu ích có thể hướng dẫn ta. Ví dụ, khi các biến ngẫu nhiên nhận giá trị liên tục, chúng ta thường quan tâm đến việc biết được giá trị kỳ vọng trung bình là bao nhiêu. Đại lượng này được gọi một cách chính thức là *kỳ vọng* (expectation).

Ví dụ 1.6.1: Giả sử chúng ta đang đầu tư, điều đầu tiên cần quan tâm có thể là lợi nhuận kỳ vọng – trung bình cộng tất cả các kết quả có thể xảy ra (và được cân nhắc theo xác suất tương ứng). Giả sử rằng với 50% xác suất, một khoản đầu tư có thể thất bại hoàn toàn, với 40% xác suất nó có thể mang lại lợi nhuận gấp 2 lần, và với 10% xác suất nó có thể mang lại lợi nhuận gấp 10 lần. Để tính lợi nhuận kỳ vọng, ta cộng tất cả các mức lợi nhuận lại, mỗi mức được nhân với xác suất xảy ra của nó:

$$0.5 \cdot 0 + 0.4 \cdot 2 + 0.1 \cdot 10 = 1.8$$

Vậy nên, lợi nhuận kỳ vọng là 1.8 lần.

Kỳ vọng của biến ngẫu nhiên rời rạc X được định nghĩa là:

$$E[X] = E_{x \sim P}[x] = \sum_x xP(X = x)$$

Giải thích: Đây là giá trị trung bình kỳ vọng của biến ngẫu nhiên rời rạc X . Mỗi giá trị có thể xảy ra của X được nhân với xác suất xảy ra của chính nó. Sau đó, các tích này được cộng lại để tính kỳ vọng. Ví dụ: Nếu một đồng xu có 50% ra sấp (giá trị 0) và 50% ra ngửa (giá trị 1), thì: $E[X] = 0 \cdot 0.5 + 1 \cdot 0.5 = 0.5 \Rightarrow$ Giá trị trung bình kỳ vọng là 0.5. Tương tự, với mật độ, ta có:

$$E[X] = \int x dp(x)$$

Giải thích: Khi biến ngẫu nhiên X có phân phối liên tục, ta không thể dùng tổng như trên. Thay vào đó, ta dùng tích phân của x nhân với hàm mật độ xác suất $p(x)$. Điều này tương tự như tính trung bình có trọng số, với trọng số là xác suất liên tục trên mỗi giá trị x . Khi quan tâm đến kỳ vọng của một hàm số $f(x)$, ta có:

- Công thức tính kỳ vọng theo phân phối rời rạc:

$$E_{x \sim P}[f(x)] = \sum_x f(x)P(x),$$

- Công thức tính kỳ vọng theo phân phối liên tục:

$$E_{x \sim P}[f(x)] = \int f(x)p(x)dx$$

Trở lại ví dụ 1.6.1, nếu độ hài lòng với mặt trắng là -1 , và các độ hài lòng tương ứng với các mức lợi nhuận 1, 2 và 10 lần là 1, 2 và 4 thì lợi nhuận kỳ vọng sẽ là:

$$0.5 \cdot (-1) + 0.4 \cdot 2 + 0.1 \cdot 4 = 0.7$$

Nếu thực sự đây là hàm ích lợi (utility) của bạn, tốt nhất bạn nên giữ tiền trong ngân hàng, không nên đầu tư.

6.2 Phương Sai

Trong các quyết định tài chính, ta không chỉ quan tâm đến kỳ vọng mà còn đến mức độ *dao động* của các kết quả quanh kỳ vọng đó. Lưu ý rằng ta không thể chỉ lấy kỳ vọng của hiệu giữa giá trị thực và giá trị kỳ vọng: $E[X - E[X]]$ Vì:

$$E[X - E[X]] = E[X] - E[E[X]] = 0$$

Tuy nhiên, ta có thể xét kỳ vọng của một hàm không âm bất kỳ của phần chênh lệch này. Phương sai (variance) của một biến ngẫu nhiên được tính bằng cách lấy kỳ vọng của bình phương độ lệch:

$$\text{Var}[X] = E[(X - E[X])^2] = E[X^2] - E[X]^2$$

Từ khai triển bình phương:

$$(X - E[X])^2 = X^2 - 2XE[X] + E[X]^2$$

Chúng minh được rằng:

$$\begin{aligned}\text{Var}(X) &= E[(X - E[X])^2] \\ &= E[X^2 - 2XE[X] + E[X]^2] \\ &= E[X^2] - E[2XE[X]] + E[E[X]^2] \quad , E[X] \text{ là hằng số} \\ &= E[X^2] - 2E[X]E[X] + E[X]^2 \\ &= E[X^2] - 2E[X]^2 + E[X]^2 \\ &= E[X^2] - E[X]^2\end{aligned}$$

Phương sai của một hàm của một biến ngẫu nhiên được định nghĩa tương tự:

$$\text{Var}_{x \sim P}[f(x)] = E_{x \sim P}[f^2(x)] - E_{x \sim P}[f(x)]^2$$

Ví dụ: quay trở lại ví dụ 1.6.1 về đầu tư ở trên, ta có thể tính phương sai khoản đầu tư như sau:

$$0.5 \cdot 0 + 0.4 \cdot 2^2 + 0.1 \cdot 10^2 - 1.8^2 = 8.36$$

Theo quy ước, kỳ vọng và phương sai được ký hiệu lần lượt là μ và σ^2 .

Một vài tính chất của phương sai:

- Với biến ngẫu nhiên X bất kỳ: $\text{Var}(X) \geq 0$, với $\text{Var}(X) = 0$ khi và chỉ khi X là hằng số.
- Với biến ngẫu nhiên X và hai số a, b bất kỳ: $\text{Var}(aX + b) = a^2\text{Var}(X)$.
- Nếu hai biến ngẫu nhiên X và Y là *độc lập*: $\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y)$.

6.3 Độ lệch chuẩn

Độ lệch chuẩn là căn bậc hai của phương sai, giúp diễn giải dễ hơn vì cùng đơn vị với biến gốc.

$$\sigma_X = \sqrt{\text{Var}(X)}$$

Các tính chất của phương sai có thể được áp dụng lại cho độ lệch chuẩn:

- Với biến ngẫu nhiên X bất kỳ: $\sigma_X \geq 0$.
- Với biến ngẫu nhiên X và hằng số a, b bất kỳ: $\sigma_{aX+b} = |a|\sigma_X$
- Nếu hai biến ngẫu nhiên X và Y là độc lập: $\sigma_{X+Y} = \sqrt{\sigma_X^2 + \sigma_Y^2}$

6.4 Hiệp Phương Sai

Khi làm việc với nhiều biến ngẫu nhiên, còn có một thông số thống kê nữa rất có ích: **hiệp phương sai (covariance)**. Thông số này đo mức độ biến thiên cùng nhau của hai biến ngẫu nhiên. Để bắt đầu, giả sử ta có hai biến ngẫu nhiên rời rạc X và Y , xác suất mang giá trị (x_i, y_j) là p_{ij} . Trong trường hợp này, hiệp phương sai được định nghĩa như sau:

$$\sigma_{XY} = \text{Cov}(X, Y) = \sum_{i,j} (x_i - \mu_X)(y_j - \mu_Y)p_{ij} = E[XY] - E[X]E[Y].$$

Để hiểu một cách trực quan về công thức trên, xét cặp biến ngẫu nhiên: X có thể nhận giá trị 1 và 3, và Y có thể nhận giá trị -1 và 3. Giả sử ta có các xác suất sau:

$$\begin{aligned}P(X = 1 \text{ và } Y = -1) &= \frac{p}{2}, \\P(X = 1 \text{ và } Y = 3) &= \frac{1-p}{2}, \\P(X = 3 \text{ và } Y = -1) &= \frac{1-p}{2}, \\P(X = 3 \text{ và } Y = 3) &= \frac{p}{2}.\end{aligned}$$

Trong đó p là tham số tùy ý trong đoạn $[0, 1]$. Nếu $p = 1$ thì X và Y luôn đồng thời mang giá trị lớn nhất hoặc nhỏ nhất của chúng, và nếu $p = 0$ thì một biến mang giá trị lớn nhất trong khi biến còn lại mang giá trị nhỏ nhất. Nếu $p = 1/2$ thì bốn khả năng có xác suất xảy ra như nhau, và không liên quan đến nhau

Hãy cùng tính hiệp phương sai. Đầu tiên, $\mu_X = 2$ và $\mu_Y = 1$, do đó:

$$\begin{aligned}\text{Cov}(X, Y) &= \sum_{i,j} (x_i - \mu_X)(y_j - \mu_Y)p_{ij} \\ &= (1-2)(-1-1)\frac{p}{2} + (1-2)(3-1)\frac{1-p}{2} \\ &\quad + (3-2)(-1-1)\frac{1-p}{2} + (3-2)(3-1)\frac{p}{2} \\ &= 4p - 2.\end{aligned}$$

Khi $p = 1$, hiệp phương sai bằng 2. Khi $p = 0$, hiệp phương sai bằng -2 . Khi $p = \frac{1}{2}$, hiệp phương sai bằng 0.

Với biến ngẫu nhiên liên tục, khái niệm hiệp phương sai không đổi. Khi đó:

$$\sigma_{XY} = \int_{\mathbb{R}^2} (x - \mu_X)(y - \mu_Y)p(x, y) dx dy.$$

Tính chất của hiệp phương sai:

- Với biến ngẫu nhiên X bất kỳ: $\text{Cov}(X, X) = \text{Var}(X)$.
- Với hai biến ngẫu nhiên X, Y và hai số a, b bất kỳ: $\text{Cov}(aX + b, Y) = \text{Cov}(X, aY + b) = a\text{Cov}(X, Y)$.
- Nếu X và Y độc lập: $\text{Cov}(X, Y) = 0$.

Ngoài ra, ta có thể sử dụng hiệp phương sai để mở rộng một hệ thức đã thấy trước đó. Nếu X và Y là hai biến ngẫu nhiên thì:

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y).$$

Với kiến thức về hiệp phương sai, ta có thể khai triển hệ thức này. Thật vậy, sử dụng đại số có thể chứng minh tổng quát rằng:

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y) + 2\text{Cov}(X, Y).$$

Công thức này là dạng tổng quát của quy tắc tính tổng phương sai cho các biến ngẫu nhiên tương quan.

6.5 Thảo luận

Trong học máy, có rất nhiều điều mà chúng ta không chắc chắn. Chúng ta có thể không chắc chắn về giá trị của nhãn được gán cho một đầu vào. Chúng ta có thể không chắc chắn về giá trị ước lượng của một tham số. Chúng ta thậm chí có thể không chắc chắn liệu dữ liệu đến trong quá trình triển khai có đến từ cùng một phân phối như dữ liệu huấn luyện hay không. **Bằng sự bất định aleatoric**, chúng ta hiểu là sự không chắc chắn vốn có trong vấn đề, do tính ngẫu nhiên thật sự mà các biến quan sát được không thể giải thích hết. **Bằng sự bất định epistemic**, chúng ta hiểu là sự không chắc chắn về các tham số của mô hình – loại bất định mà ta có thể hy vọng sẽ giảm đi khi thu thập thêm dữ liệu. Chúng ta có thể có bất định epistemic liên quan đến xác suất đồng xu ra mặt ngửa, nhưng ngay cả khi đã biết xác suất đó, ta vẫn còn bất định aleatoric về kết quả của những lần tung tiếp theo. Dù chúng ta quan sát bao nhiêu, cũng không thể chắc chắn hơn hoặc kém 50% rằng lần tung tới sẽ ra mặt ngửa. Các thuật ngữ này xuất phát từ mô hình cơ học (xem ví dụ Kiureghian và Ditlevsen (2009) về khía cạnh của **định lượng bất định**). Cũng cần lưu ý rằng, về mặt ngôn ngữ triết học, mọi sự bất định đều là epistemic vì nó liên quan đến tri thức. Chúng ta thấy rằng việc lấy mẫu từ một phân phối xác suất không biết có thể giúp ước lượng các tham số của phân phối tạo dữ liệu. Tuy nhiên, tốc độ đạt được điều này có thể rất chậm. Trong ví dụ tung đồng xu, ta không thể làm gì hơn ngoài việc thiết kế các ước lượng hội tụ theo tốc độ:

$$\frac{1}{\sqrt{n}}$$

với n là kích thước mẫu (số lần tung). Nghĩa là khi tăng từ 10 lên 1000 lần quan sát (một việc hoàn toàn khả thi), ta giảm được độ bất định đi 10 lần, còn tăng thêm 1000 lần nữa chỉ giảm thêm được hệ số 1.41. Đây là một đặc điểm cố hữu của học máy: sau những cải thiện dễ dàng ban đầu, các bước tiến tiếp theo đòi hỏi rất nhiều dữ liệu và tính toán. Để thấy rõ điều này trong mô hình ngôn ngữ lớn, xem Revels et al. (2016). Chúng ta cũng làm rõ hơn về ngôn ngữ và công cụ mô hình thống kê. Trong quá trình đó, chúng ta học được về xác suất có điều kiện và một trong những phương trình quan trọng nhất trong thống kê – định lý Bayes. Đây là công cụ hiệu quả để tách biệt thông tin đến từ dữ liệu thông qua phân bố hậu nghiệm:

$$P(B | A)$$

Phân bố này thể hiện mức độ dữ liệu B ủng hộ các tham số A thế nào, cùng với phân bố tiên nghiệm $P(A)$, vốn chi phối mức độ khả thi của A ban đầu. Đặc biệt, ta thấy quy tắc này có thể



được dùng để gán xác suất cho các chẩn đoán dựa trên hiệu quả kiểm tra và độ phổ biến của căn bệnh (tức là phân bố tiên nghiệm).

7 Bài tập

Phần này nhóm trình bài giải cho các bài tập 1, 2, 3, 4, 5, 6, 7, 8 ở mục 2.6.8 và code Python để mô phỏng hoặc thực nghiệm cho các bài tập này.

Bài tập 1. Lấy một ví dụ cho thấy việc quan sát thêm dữ liệu hoặc tăng kích thước tập huấn luyện có thể làm giảm mức độ không chắc chắn (uncertainty) về kết quả, tới mức tùy ý nhỏ.

Bài giải 1. Hiện tượng bất định hay không chắc chắn trong Machine learning và entropy có liên quan nhau "Entropy càng thấp, sự chắc chắn về kết quả càng cao". Hiện tượng bất định liên quan đến khái niệm entropy trong lý thuyết xác suất, entropy dùng để đo lường mức độ không chắc chắn của một biến ngẫu nhiên. Xác suất cao lượng thông tin thu được càng giảm và ngược lại.

Định nghĩa Entropy: ký hiệu $H(X)$ là đại lượng trong lý thuyết thông tin dùng xác suất để tính toán, đo lường mức độ không chắc chắn hoặc lượng thông tin trung bình của một biến ngẫu nhiên X . Entropy càng thấp thì càng dễ đoán trước giá trị của biến ngẫu nhiên đó.

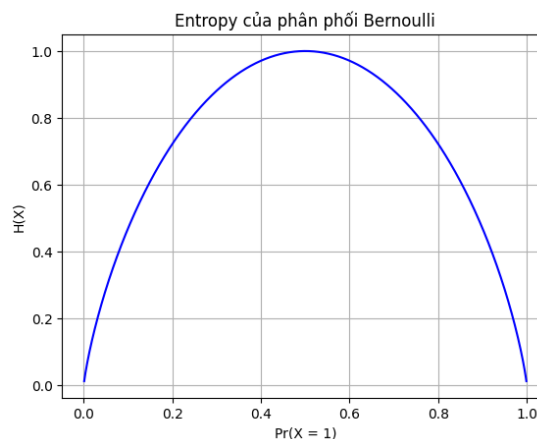
- Công thức tính Entropy (cho biến rời rạc):

$$H(X) = - \sum_{x \in \mathcal{X}} p(x) \log p(x)$$

Trong đó:

- $p(x)$: Xác suất xảy ra sự kiện x .
- \mathcal{X} : Tập hợp tất cả các kết quả có thể.
- Entropy đo lường sự không chắc chắn
 - Khi entropy cao:
 - Xác suất của các kết quả gần bằng nhau (ví dụ: đồng xu cân bằng, $p = 0.5$).
 - Khó dự đoán kết quả nhất vì mọi khả năng đều có khả năng xảy ra tương đương.
 - Ví dụ: Tung một đồng xu cân bằng ($p = 0.5$), entropy là 1 bit nếu dùng log cơ số 2 (không chắc chắn tối đa).
 - Khi entropy thấp (Entropy gần bằng 0):
 - Một kết quả có xác suất rất cao, các kết quả khác rất thấp (ví dụ: đồng xu lệch với $p = 0.99$).

- Dễ dự đoán kết quả hơn vì một sự kiện gần như chắc chắn xảy ra.
- Ví dụ: Đồng xu với $p = 0.99$, entropy $H(X) \approx 0.08$ bit (rất không chắc chắn).
- Trường hợp xấu: Nếu một sự kiện có xác suất $p = 1$ (chắc chắn xảy ra), entropy $H(X) = 0 \Rightarrow$ Không có không chắc chắn nào cả. - Mối quan hệ giữa entropy và dữ liệu
 - Quan sát thêm dữ liệu giúp cải thiện ước lượng xác suất $p(x)$, từ đó giảm entropy.
 - Ví dụ:
 - * Ban đầu: Không biết tỷ lệ mặt ngửa của đồng xu (p có thể là bất kỳ giá trị nào trong $[0, 1]$), entropy cao.
 - * Sau khi tung đồng xu 1000 lần và thấy 950 lần ngửa: Ước lượng $p \approx 0.95$, entropy giảm mạnh (gần 0) \Rightarrow Sự không chắc chắn về kết quả của lần tung thứ 1001 gần như biến mất.
- Giải thích bằng hình ảnh Đồ thị entropy của biến Bernoulli, ví dụ đồng xu với xác suất mặt ngửa p : Code Python để mô phỏng bài tập 1



- Entropy đạt tối đa khi $p = 0.5$ (không chắc chắn nhất).
- Entropy (với log cơ số 2) tiến về 0 khi $p \rightarrow 0$ hoặc $p \rightarrow 1$ (không chắc chắn giảm).
- Ứng dụng trong giải bài tập 1:
 - Ví dụ phù hợp: Ước lượng tham số p của phân phối Bernoulli (như tung đồng xu).
 - Khi số lần quan sát $n \rightarrow \infty$, ước lượng \hat{p} hội tụ về p thực, entropy của biến ngẫu nhiên giảm về 0 (nếu p là 0 hoặc 1) hoặc một giá trị nhỏ (nếu p gần 0/1) \Rightarrow Không chắc chắn gần như mất hẳn” hoặc “Entropy tiến tới giá trị tối thiểu là 0.

- Tóm tắt:

- * Entropy thấp = Phân phối xác suất tập trung vào một vài kết quả \rightarrow Dễ dự đoán \rightarrow Không chắc chắn giảm.
- * Entropy cao = Phân phối đều giữa nhiều kết quả \rightarrow Khó dự đoán \rightarrow Không chắc chắn tăng.
- * Thêm dữ liệu giúp "làm rõ" phân phối thực \rightarrow Giảm entropy.

Chạy mô phỏng bài tập 1

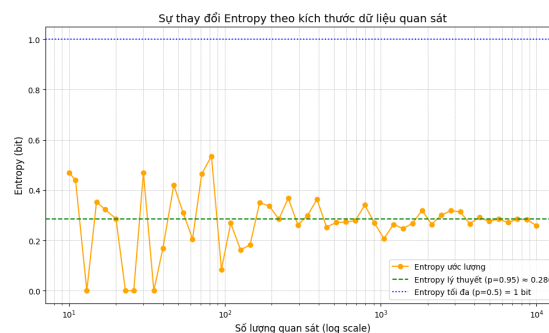
Mô phỏng và trực quan hóa mối quan hệ giữa kích thước tập dữ liệu quan sát và entropy, theo: Đường entropy ước lượng từ dữ liệu; Đường entropy lý thuyết cho $p = 0.95$; Đường entropy tối đa khi $p = 0.5$ để đối chiếu.

```
1  ## 1. Vẽ đồ thị Entropy của phân phối Bernoulli theo xác suất $p$:
2
3  import numpy as np
4  import matplotlib.pyplot as plt
5  def bernoulli_entropy(p):
6      return -p * np.log2(p) - (1 - p) * np.log2(1 - p)
7  p_vals = np.linspace(0.001, 0.999, 1000) # tránh log(0)
8  entropy_vals = bernoulli_entropy(p_vals)
9  plt.plot(p_vals, entropy_vals, color='blue')
10 plt.xlabel("Pr(X = 1)")
11 plt.ylabel("H(X)")
12 plt.title("Entropy của phân phối Bernoulli")
13 plt.grid(True)
14 plt.show()
15
16 ## 2. Mô phỏng và trực quan hóa mối quan hệ giữa kích thước tập dữ liệu quan
17 ↪ sát và entropy, có kèm theo:
18
19 * Đường entropy ước lượng từ dữ liệu.
20 * Đường entropy lý thuyết cho  $p = 0.95$ 
21 * Đường entropy tối đa khi  $p = 0.5$  để đối chiếu.
```

```
21 import numpy as np
22 import matplotlib.pyplot as plt
23 # Hàm tính entropy Bernoulli với log cơ số 2
24 def bernoulli_entropy(p):
25     if p <= 0 or p >= 1:
26         return 0.0
27     return -p * np.log2(p) - (1 - p) * np.log2(1 - p)
28 # Tham số thực tế (p thật của đồng xu)
29 p_true = 0.95
30 entropy_true = bernoulli_entropy(p_true)
31 entropy_max = bernoulli_entropy(0.5)
32 # Các kích thước mẫu (log scale)
33 sample_sizes = np.logspace(1, 4, num=50, dtype=int)
34 estimated_entropies = []
35 # Mô phỏng: với mỗi n, tạo n mẫu Bernoulli với p = 0.95, tính entropy từ p_hat
36 np.random.seed(42) # Đặt seed để tái lập kết quả
37 for n in sample_sizes:
38     samples = np.random.binomial(n=1, p=p_true, size=n)
39     p_hat = np.mean(samples)
40     H_hat = bernoulli_entropy(p_hat)
41     estimated_entropies.append(H_hat)
42 # Vẽ biểu đồ
43 plt.figure(figsize=(10, 6))
44 plt.plot(sample_sizes, estimated_entropies, marker='o', color='orange',
45          ↪ label='Entropy ước lượng')
46 plt.axhline(entropy_true, color='green', linestyle='--', label=f'Entropy lý
47          ↪ thuyết (p={p_true}) ≈ {entropy_true:.3f}')
48 plt.axhline(entropy_max, color='blue', linestyle=':', label='Entropy tối đa
49          ↪ (p=0.5) = 1 bit')
47 plt.xscale('log')
48 plt.xlabel('Số lượng quan sát (log scale)', fontsize=12)
49 plt.ylabel('Entropy (bit)', fontsize=12)
```

```
50 plt.title('Sự thay đổi Entropy theo kích thước dữ liệu quan sát', fontsize=14)
51 plt.grid(True, which="both", linestyle='--', linewidth=0.5)
52 plt.legend()
53 plt.tight_layout()
54 plt.show()
```

Kết quả mô phỏng và trực quan hóa mối quan hệ giữa kích thước tập dữ liệu quan sát và entropy, theo: Đường entropy ước lượng từ dữ liệu; Đường entropy lý thuyết cho $p = 0.95$; Đường entropy tối đa khi $p = 0.5$ để đối chiếu.



Hình 9: Trực quan hóa mối quan hệ giữa kích thước tập dữ liệu quan sát và entropy

Code Python để mô phỏng bài tập 1 (Trung bình 10.000 lần thử (Monte Carlo Averaging))

```
1
2 ## 3. Trung bình 10.000 lần thử (Monte Carlo Averaging)
3 import numpy as np
4 import matplotlib.pyplot as plt
5 # Hàm tính entropy của phân phối Bernoulli
6 def bernoulli_entropy(p):
7     if p <= 0 or p >= 1:
8         return 0.0
9     return -p * np.log2(p) - (1 - p) * np.log2(1 - p)
10 # Xác suất thật của đồng xu
11 p_true = 0.95
12 entropy_true = bernoulli_entropy(p_true)
```



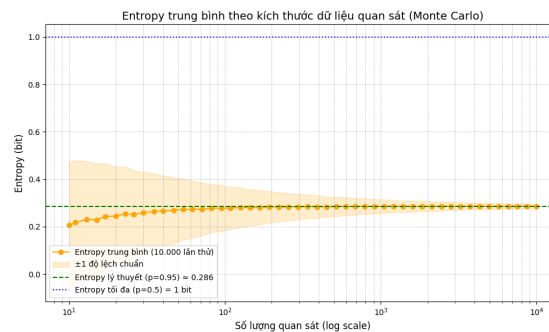
```
13 entropy_max = bernoulli_entropy(0.5)
14 # Các kích thước mẫu (log scale)
15 sample_sizes = np.logspace(1, 4, num=50, dtype=int)
16 # Số lần mô phỏng mỗi kích thước
17 num_trials = 10000
18 # Danh sách lưu kết quả
19 mean_entropies = []
20 std_entropies = []
21 # Đặt seed để tái lập kết quả
22 np.random.seed(42)
23 # Mô phỏng Monte Carlo
24 for n in sample_sizes:
25     entropies = []
26     for _ in range(num_trials):
27         samples = np.random.binomial(n=1, p=p_true, size=n)
28         p_hat = np.mean(samples)
29         H_hat = bernoulli_entropy(p_hat)
30         entropies.append(H_hat)
31     mean_entropies.append(np.mean(entropies))
32     std_entropies.append(np.std(entropies))
33 # Vẽ biểu đồ kết quả
34 plt.figure(figsize=(10, 6))
35 plt.plot(sample_sizes, mean_entropies, marker='o', color='orange',
36         ↪ label='Entropy trung bình (10.000 lần thử)')
37 plt.fill_between(sample_sizes,
38                 np.array(mean_entropies) - np.array(std_entropies),
39                 np.array(mean_entropies) + np.array(std_entropies),
40                 color='orange', alpha=0.2, label='±1 độ lệch chuẩn')
41 plt.axhline(entropy_true, color='green', linestyle='--', label=f'Entropy lý
42         ↪ thuyết (p={p_true}) ≈ {entropy_true:.3f}')
43 plt.axhline(entropy_max, color='blue', linestyle=':', label='Entropy tối đa
44         ↪ (p=0.5) = 1 bit')
```

```

42 plt.xscale('log')
43 plt.xlabel('Số lượng quan sát (log scale)', fontsize=12)
44 plt.ylabel('Entropy (bit)', fontsize=12)
45 plt.title('Entropy trung bình theo kích thước dữ liệu quan sát (Monte Carlo)',
    ↪  fontsize=14)
46 plt.grid(True, which="both", linestyle='--', linewidth=0.5)
47 plt.legend()
48 plt.tight_layout()
49 plt.show()
50

```

Kết quả mô phỏng Monte Carlo Averaging



Hình 10: Monte Carlo Averaging với 10.000 lần thử

Code Python để mô phỏng bài tập 1 (So sánh nhiều giá trị p trong phân phối Bernoulli)

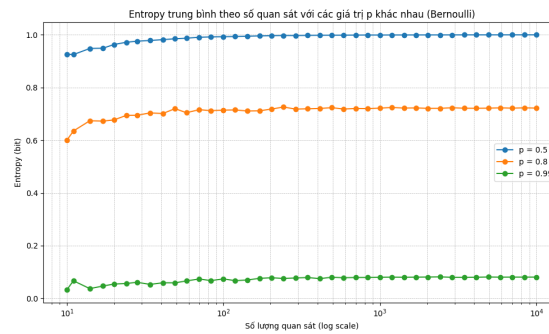
```

1
2 # ===== (4) So sánh nhiều giá trị p trong phân phối Bernoulli =====
3
4 def bernoulli_entropy(p):
5     if p <= 0 or p >= 1:
6         return 0.0
7     return -p * np.log2(p) - (1 - p) * np.log2(1 - p)
8
9 ps = [0.5, 0.8, 0.99]

```

```
10 sample_sizes = np.logspace(1, 4, num=40, dtype=int)
11 num_trials = 300
12
13 plt.figure(figsize=(10, 6))
14 for p_true in ps:
15     mean_entropies = []
16     for n in sample_sizes:
17         entropies = []
18         for _ in range(num_trials):
19             samples = np.random.binomial(n=1, p=p_true, size=n)
20             p_hat = np.mean(samples)
21             H_hat = bernoulli_entropy(p_hat)
22             entropies.append(H_hat)
23             mean_entropies.append(np.mean(entropies))
24         plt.plot(sample_sizes, mean_entropies, marker='o', label=f'p = {p_true}')
25 plt.xscale('log')
26 plt.xlabel('Số lượng quan sát (log scale)')
27 plt.ylabel('Entropy (bit)')
28 plt.title('Entropy trung bình theo số quan sát với các giá trị p khác nhau
29 ↪ (Bernoulli)')
30 plt.grid(True, which="both", linestyle='--', linewidth=0.5)
31 plt.legend()
32 plt.tight_layout()
33 plt.show()
```

Kết quả so sánh nhiều giá trị p trong phân phối Bernoulli



Hình 11: So sánh nhiều giá trị p trong phân phối Bernoulli

Code Python để mô phỏng bài tập 1 (Mô phỏng entropy cho biến rời rạc nhiều giá trị)

```

1
2 # ===== (5) Mô phỏng entropy cho biến rời rạc nhiều giá trị =====
3
4 def discrete_entropy(p_vec):
5     p_vec = np.array(p_vec)
6     p_vec = p_vec[p_vec > 0]
7     return -np.sum(p_vec * np.log2(p_vec))
8
9 # Phân phối thật: 4 giá trị không đều
10 p_true_vec = np.array([0.5, 0.3, 0.15, 0.05])
11 n_categories = len(p_true_vec)
12 mean_entropies = []
13
14 for n in sample_sizes:
15     entropies = []
16     for _ in range(num_trials):
17         samples = np.random.choice(n_categories, size=n, p=p_true_vec)
18         counts = np.bincount(samples, minlength=n_categories)
19         p_hat_vec = counts / np.sum(counts)
20         H_hat = discrete_entropy(p_hat_vec)
21         entropies.append(H_hat)
22     mean_entropies.append(np.mean(entropies))
23
24 entropy_true_discrete = discrete_entropy(p_true_vec)
25 plt.figure(figsize=(10, 6))

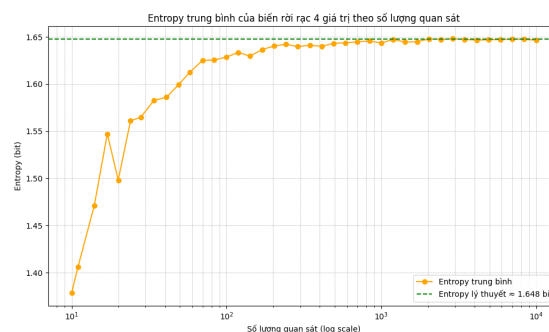
```

```

23 plt.plot(sample_sizes, mean_entropies, marker='o', color='orange',
    ↪ label='Entropy trung bình')
24 plt.axhline(entropy_true_discrete, linestyle='--', color='green',
25             label=f'Entropy lý thuyết ≈ {entropy_true_discrete:.3f} bit')
26 plt.xscale('log')
27 plt.xlabel('Số lượng quan sát (log scale)')
28 plt.ylabel('Entropy (bit)')
29 plt.title('Entropy trung bình của biến rời rạc 4 giá trị theo số lượng quan
    ↪ sát')
30 plt.grid(True, which="both", linestyle='--', linewidth=0.5)
31 plt.legend()
32 plt.tight_layout()
33 plt.show()

```

Kết quả mô phỏng entropy cho biến rời rạc nhiều giá trị



Hình 12: Mô phỏng entropy cho biến rời rạc nhiều giá trị

Code Python để mô phỏng bài tập 1 (KL-divergence giữa phân phối ước lượng và thật)

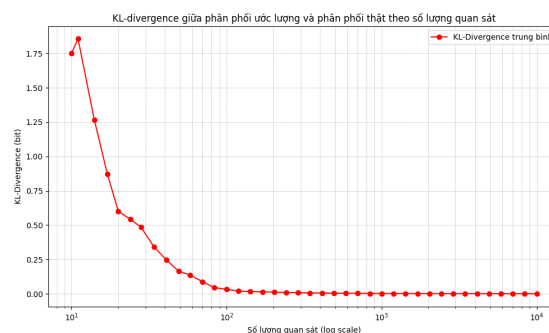
```

1
2 # ===== (6) KL-divergence giữa phân phối ước lượng và thật =====
3
4 kl_means = []
5 for n in sample_sizes:
6     kls = []

```

```
7     for _ in range(num_trials):
8         samples = np.random.choice(n_categories, size=n, p=p_true_vec)
9         counts = np.bincount(samples, minlength=n_categories)
10        p_hat_vec = counts / np.sum(counts)
11        # Tránh log(0)
12        p_hat_vec = np.clip(p_hat_vec, 1e-10, 1)
13        kls.append(kl_divergence(p_true_vec, p_hat_vec, base=2))
14    kl_means.append(np.mean(kls))
15    plt.figure(figsize=(10, 6))
16    plt.plot(sample_sizes, kl_means, marker='o', color='red', label='KL-Divergence
    ↪ trung bình')
17    plt.xscale('log')
18    plt.xlabel('Số lượng quan sát (log scale)')
19    plt.ylabel('KL-Divergence (bit)')
20    plt.title('KL-divergence giữa phân phối ước lượng và phân phối thật theo số
    ↪ lượng quan sát')
21    plt.grid(True, which="both", linestyle='--', linewidth=0.5)
22    plt.legend()
23    plt.tight_layout()
24    plt.show()
```

Kết quả mô phỏng KL-divergence giữa phân phối ước lượng và thật



Hình 13: KL-divergence giữa phân phối ước lượng và thật

Code Python để mô phỏng bài tập 1 (So sánh KL-divergence giữa nhiều phân phối thật khác

nhau.)

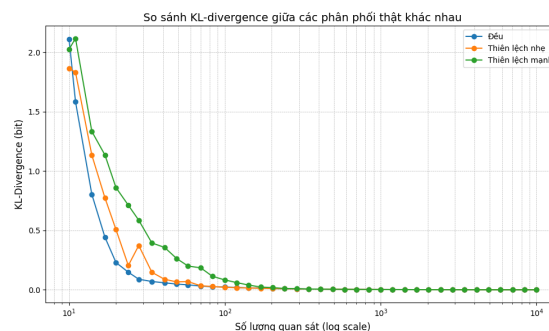
```
1  ## 7. So sánh KL-divergence giữa nhiều phân phối thật khác nhau.
2
3  import numpy as np
4  import matplotlib.pyplot as plt
5  from scipy.stats import entropy as kl_divergence
6  # Hàm: Tính KL-divergence trung bình giữa phân phối thật và ước lượng
7  def compute_kl_curve(p_true_vec, sample_sizes, num_trials=300):
8      kl_means = []
9      n_categories = len(p_true_vec)
10     for n in sample_sizes:
11         kls = []
12         for _ in range(num_trials):
13             samples = np.random.choice(n_categories, size=n, p=p_true_vec)
14             counts = np.bincount(samples, minlength=n_categories)
15             p_hat = counts / np.sum(counts)
16             # Tránh log(0)
17             p_hat = np.clip(p_hat, 1e-10, 1)
18             kls.append(kl_divergence(p_true_vec, p_hat, base=2))
19         kl_means.append(np.mean(kls))
20     return kl_means
21 # Thiết lập các phân phối thật để so sánh
22 true_distributions = {
23     "Đều": np.array([0.25, 0.25, 0.25, 0.25]),
24     "Thiên lệch nhẹ": np.array([0.4, 0.3, 0.2, 0.1]),
25     "Thiên lệch mạnh": np.array([0.7, 0.2, 0.08, 0.02])
26 }
27 # Kích thước mẫu quan sát
28 sample_sizes = np.logspace(1, 4, num=40, dtype=int)
29 # Vẽ biểu đồ
30 plt.figure(figsize=(10, 6))
31 for label, p_true in true_distributions.items():
```

```

32     kl_curve = compute_kl_curve(p_true, sample_sizes)
33     plt.plot(sample_sizes, kl_curve, marker='o', label=label)
34 # Biểu đồ
35 plt.xscale('log')
36 plt.xlabel('Số lượng quan sát (log scale)', fontsize=12)
37 plt.ylabel('KL-Divergence (bit)', fontsize=12)
38 plt.title('So sánh KL-divergence giữa các phân phối thật khác nhau',
    ↪     fontsize=14)
39 plt.grid(True, which="both", linestyle='--', linewidth=0.5)
40 plt.legend()
41 plt.tight_layout()
42 plt.show()

```

Kết quả so sánh KL-divergence giữa nhiều phân phối thật khác nhau



Hình 14: So sánh KL-divergence giữa nhiều phân phối thật khác nhau

Bài tập 2. Đưa ra một ví dụ về việc quan sát thêm dữ liệu sẽ chỉ giảm lượng không chắc chắn đến một điểm nhất định và sau đó không giảm thêm nữa. Giải thích tại sao lại như vậy và bạn mong đợi điểm này sẽ xảy ra ở đâu.

Bài giải 2. Ví dụ minh họa: Đo chiều dài một vật bằng thước đo có độ chính xác hữu hạn.

Giả sử chúng ta muốn đo chiều dài của một thanh kim loại bằng một thước đo có vạch chia đến 1mm. Chúng ta tiến hành đo thanh này 500 lần để lấy trung bình nhằm giảm bất định do đo đạc.

Hiện tượng xảy ra:

- Ban đầu, khi chúng ta đo nhiều lần, sai số ngẫu nhiên (do tay đo, góc nhìn, vị trí đặt thước...) sẽ được triệt tiêu dần qua trung bình cộng.
- Tuy nhiên, do độ phân giải của thước chỉ đến 1mm, nên chúng ta không thể phân biệt chiều dài thật nếu nó nằm giữa hai vạch (ví dụ: 153.3mm hay 153.7mm đều có thể bị làm tròn thành 153mm hoặc 154mm tùy lúc).
- Lúc này, có một ngưỡng sai số hệ thống hoặc giới hạn độ chính xác thiết bị mà chúng ta không thể vượt qua bằng cách tăng số lượng đo.

Giải thích nguyên nhân:

- Tổng độ không chắc chắn có thể chia thành 2 phần:
 - Không chắc chắn ngẫu nhiên: có thể giảm bằng nhiều phép đo.
 - Không chắc chắn hệ thống: không thể giảm bằng quan sát thêm nếu không cải thiện thiết bị hay phương pháp đo.
- Khi số lượng đo tăng lên, phần không chắc chắn ngẫu nhiên giảm, nhưng không thể giảm phần hệ thống.

Kỳ vọng điểm bão hòa xảy ra khi:

- Số phép đo đủ lớn để trung bình hóa toàn bộ sai số ngẫu nhiên.
- Từ đó, độ không chắc chắn còn lại chủ yếu là do sai số hệ thống (ví dụ: do thước không chính xác, mắt người không đọc được vạch chia nhỏ hơn 1mm...).

Tóm lại, việc quan sát thêm dữ liệu chỉ giúp giảm bất định do yếu tố ngẫu nhiên. Khi bất định chủ yếu đến từ giới hạn của hệ thống (như độ phân giải thiết bị), việc thu thêm dữ liệu sẽ không làm giảm bất định nữa. Điểm "bão hòa" xảy ra khi sai số hệ thống chiếm ưu thế.

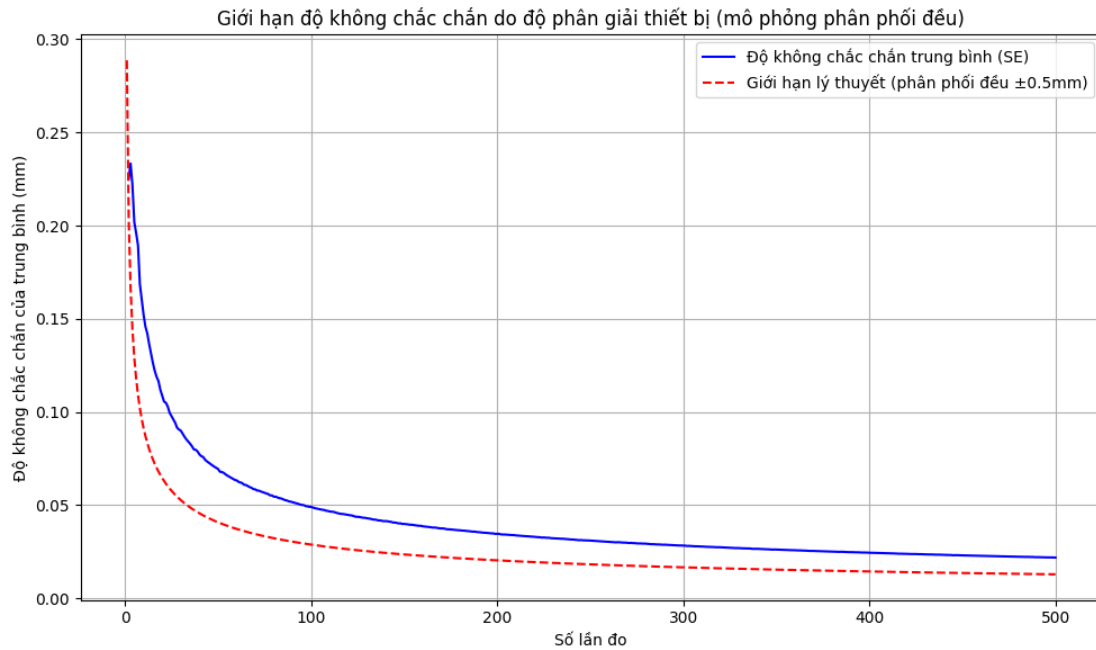
Code Python chạy mô phỏng:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # Chiều dài thật sự của vật thể (mm)
5 true_length = 153.6
```

```
6
7 # Độ phân giải của thước đo (1 mm)
8 measurement_resolution = 1.0
9
10 # Danh sách số lần đo tăng dần từ 1 đến 500
11 num_measurements_list = np.arange(1, 501)
12
13 # Danh sách lưu độ không chắc chắn trung bình qua nhiều lần mô phỏng
14 avg_uncertainties = []
15
16 # Số lần lặp lại để lấy trung bình SE (Monte Carlo)
17 num_trials = 100
18
19 for n in num_measurements_list:
20     se_trials = []
21
22     for _ in range(num_trials):
23         # Sai số do làm tròn theo phân phối đều ±0.5mm (chỉ xét sai số làm
24         ↪ tròn)
25         rounding_error = np.random.uniform(low=-0.5, high=0.5, size=n)
26         raw_measurements = true_length + rounding_error
27
28         # Làm tròn theo độ phân giải của thiết bị
29         rounded_measurements = np.round(raw_measurements /
30         ↪ measurement_resolution) * measurement_resolution
31
32         if n > 1:
33             std_error = np.std(rounded_measurements, ddof=1) / np.sqrt(n)
34         else:
35             std_error = np.nan
36
37     se_trials.append(std_error)
```

```
36
37     # Trung bình độ không chắc chắn qua các lần mô phỏng
38     avg_uncertainties.append(np.nanmean(se_trials))
39
40 # Đường giới hạn lý thuyết của sai số làm tròn phân phối đều ±0.5mm
41 # Std của phân phối đều = (b - a)/sqrt(12)
42 theoretical_limit = (1.0 / np.sqrt(12)) / np.sqrt(num_measurements_list)
43
44 # Vẽ biểu đồ
45 plt.figure(figsize=(10, 6))
46 plt.plot(num_measurements_list, avg_uncertainties, label='Độ không chắc chắn
    ↪ trung bình (SE)', color='blue')
47 plt.plot(num_measurements_list, theoretical_limit, linestyle='--',
    ↪ color='red', label='Giới hạn lý thuyết (phân phối đều ±0.5mm)')
48 plt.xlabel('Số lần đo')
49 plt.ylabel('Độ không chắc chắn của trung bình (mm)')
50 plt.title('Giới hạn độ không chắc chắn do độ phân giải thiết bị (mô phỏng phân
    ↪ phối đều)')
51 plt.legend()
52 plt.grid(True)
53 plt.tight_layout()
54 plt.show()
```

Kết quả xuất ra màn hình:



Hình 15: Mô phỏng ví dụ về giới hạn không chắc chắn quan sát được.

Kết luận từ biểu đồ:

- Ban đầu, việc tăng số lượng đo giúp giảm độ không chắc chắn rõ rệt.
- Nhưng sau một mức nhất định (khoảng 100 lần đo), đường cong tiệm cận về một giới hạn không thể vượt qua – chính là sai số hệ thống do độ phân giải thước đo.
- Quan sát thêm không làm giảm thêm độ không chắc chắn.

Bài tập 3. Kết quả thực nghiệm cho thấy sự hội tụ về giá trị trung bình khi tung đồng xu. Hãy tính phương sai của ước lượng xác suất xuất hiện mặt ngửa sau khi tung đồng xu n lần.

1. Phương sai thay đổi thế nào khi số lần quan sát n tăng lên?
2. Sử dụng bất đẳng thức Chebyshev để chặn độ lệch so với kỳ vọng.
3. Mối liên hệ với định lý giới hạn trung tâm (Central Limit Theorem - CLT)?

Bài giải 3. 1. Tính phương sai và Phương sai thay đổi thế nào khi số lần quan sát n tăng lên?

- Ước lượng xác suất (tần suất):

- Gọi X_1, X_2, \dots, X_n là đại diện cho kết quả n lần tung đồng xu:
- $X_i = 1$ nếu ra mặt ngửa (head).
- $X_i = 0$ nếu ra mặt sấp (tail).
- Xác suất của mặt ngửa là p , tức là

$$P(X_i = 1) = p \quad (1)$$

- Ước lượng \tilde{p}_n của p sau n lần tung là:

$$\tilde{p}_n = \frac{1}{n} \sum_{i=1}^n X_i \quad (2)$$

- Phương sai

- Tính $\text{Var}(X_i)$:

$$\text{Var}(X_i) = E[X_i^2] - (E[X_i])^2 = p - p^2 = p(1 - p) \quad (3)$$

- Do các X_i độc lập và cùng phân phối (mỗi X_i là biến nhị phân Bernoulli) nên:

$$\text{Var}(\tilde{p}_n) = \text{Var}\left(\frac{1}{n} \sum_{i=1}^n X_i\right) = \frac{1}{n^2} \sum_{i=1}^n \text{Var}(X_i) = \frac{1}{n^2} \cdot n \cdot \text{Var}(X_i) = \frac{\text{Var}(X_i)}{n} = \frac{p(1-p)}{n} \quad (4)$$

- Phương sai tỷ lệ nghịch với n . Khi n tăng, phương sai giảm với tốc độ $\frac{1}{n}$.
- Ví dụ: Nếu n tăng gấp 5 lần, phương sai giảm còn $\frac{1}{5}$.

2. Bất đẳng thức Chebyshev

- Áp dụng bất đẳng thức Chebyshev cho biến ngẫu nhiên \tilde{p}_n , với kỳ vọng $E[\tilde{p}_n] = p$:

$$P(|p - \tilde{p}_n| \geq \epsilon) \leq \frac{\text{Var}(\tilde{p}_n)}{\epsilon^2} = \frac{p(1-p)}{n\epsilon^2} \quad (5)$$

- Điều này chứng tỏ xác suất $E(p)$ lệch khỏi p một khoảng ϵ bị chặn bởi $\frac{p(1-p)}{n\epsilon^2}$.
- Ý nghĩa là khi n lớn, xác suất này tiến về 0, $E(p) \approx p$.

3. Liên hệ với định lý giới hạn trung tâm (CLT):

- Áp dụng được CLT vì X_i độc lập và phân phối Bernoulli giống nhau.

- Định lý giới hạn trung tâm phát biểu với n đủ lớn:

$$\sqrt{n}(\tilde{p}_n - p) \xrightarrow{d} N(0, p(1-p)) \quad (6)$$

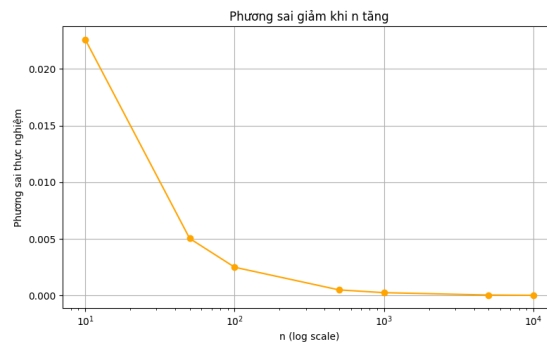
- Tức là \tilde{p}_n có phân phối xấp xỉ chuẩn với kỳ vọng p và phương sai $\frac{p(1-p)}{n}$.
- CLT cho thấy sự hội tụ phân phối của \tilde{p}_n về phân phối chuẩn, mạnh hơn so với chỉ sử dụng phương sai hoặc Chebyshev.

Code Python để chạy mô phỏng, tính toán và trực quan hoá kết quả của bài tập 3 (Phương sai giảm khi n tăng)

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import seaborn as sns
4 from scipy.stats import norm
5 # Thiết lập hạt giống để tái lập kết quả
6 np.random.seed(42)
7 # Thông số chung
8 p = 0.5 # Xác suất mặt ngửa
9 eps = 0.05 # Ngưỡng epsilon cho Chebyshev
10 n_values = [10, 50, 100, 500, 1000, 5000, 10000] # Các giá trị n
11 num_trials = 1000 # Số lần mô phỏng cho mỗi n
12 # Danh sách lưu kết quả
13 variances = []
14 chebyshev_bounds = []
15 chebyshev_empirical = []
16 # Vòng lặp qua các giá trị n
17 for n in n_values:
18     # Sinh num_trials mẫu Bernoulli, mỗi mẫu có n quan sát
19     samples = np.random.binomial(1, p, (num_trials, n))
20     # Tính ước lượng p
21     estimates = np.mean(samples, axis=1)
22     # 1. Tính phương sai thực nghiệm
23     var_empirical = np.var(estimates)
```

```
24     variances.append(var_empirical)
25     # 2. Kiểm nghiệm Chebyshev
26     cheb_bound = p * (1 - p) / (n * eps**2)
27     chebyshev_bounds.append(cheb_bound)
28     cheb_actual = np.mean(np.abs(estimates - p) >= eps)
29     chebyshev_empirical.append(cheb_actual)
30     # 3. Kiểm nghiệm CLT với n lớn
31     n_clt = 1000
32     samples_clt = np.random.binomial(1, p, (num_trials, n_clt))
33     estimates_clt = np.mean(samples_clt, axis=1)
34     z_scores = np.sqrt(n_clt) * (estimates_clt - p)
35
36     # Biểu đồ 1: Giảm phương sai theo n
37     plt.figure(figsize=(8,5))
38     plt.plot(n_values, variances, marker='o', color='orange')
39     plt.xscale("log")
40     plt.xlabel("n (log scale)")
41     plt.ylabel("Phương sai thực nghiệm")
42     plt.title("Giảm phương sai khi n tăng")
43     plt.grid(True)
44     plt.tight_layout()
45     plt.show()
```

Phương sai giảm khi n tăng

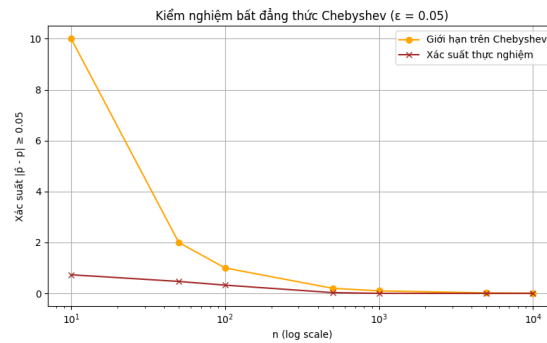


Hình 16: Phương sai giảm khi n tăng

Code Python để chạy mô phỏng, tính toán và trực quan hoá kết quả của bài tập 3 (Kiểm nghiệm bất đẳng thức Chebyshev)

```
1
2 # Biểu đồ 2: Kiểm nghiệm BDT Chebyshev
3 plt.figure(figsize=(8,5))
4 plt.plot(n_values, chebyshev_bounds, label="Giới hạn trên Chebyshev",
5          ↪ marker='o', color='orange')
6 plt.plot(n_values, chebyshev_empirical, label="Xác suất thực nghiệm",
7          ↪ marker='x', color='brown')
8 plt.xscale("log")
9 plt.xlabel("n (log scale)")
10 plt.ylabel(f"Xác suất  $|p - p| \geq \epsilon$ ")
11 plt.title(f"Kiểm nghiệm bất đẳng thức Chebyshev ( $\epsilon$ )")
12 plt.legend()
13 plt.grid(True)
14 plt.tight_layout()
15 plt.show()
```

Kiểm nghiệm bất đẳng thức Chebyshev (ϵ)

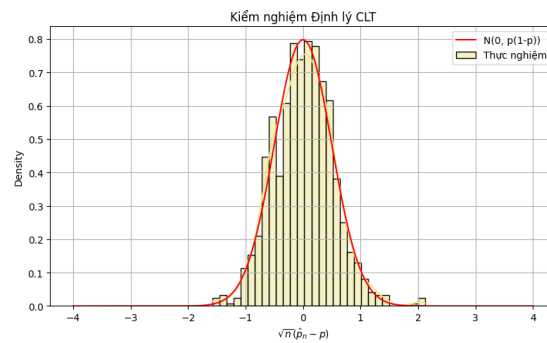


Hình 17: Kiểm nghiệm bất đẳng thức Chebyshev

Code Python để chạy mô phỏng, tính toán và trực quan hoá kết quả của bài tập 3 (Histogram kiểm nghiệm CLT)

```
1
2 # Biểu đồ 3: Histogram kiểm nghiệm CLT
3 plt.figure(figsize=(8,5))
4 sns.histplot(z_scores, bins=30, kde=True, stat="density", label="Thực nghiệm",
5 color='khaki', edgecolor='black')
6 x = np.linspace(-4, 4, 200)
7 plt.plot(x, norm.pdf(x, 0, np.sqrt(p*(1-p))), label="N(0, p(1-p))",
8 color='red')
9 plt.xlabel(r"$\sqrt{n}(\hat{p}_n - p)$")
10 plt.title("Kiểm nghiệm Định lý Giới hạn Trung tâm (CLT)")
11 plt.legend()
12 plt.grid(True)
13 plt.tight_layout()
14 plt.show()
```

Histogram kiểm nghiệm định lý CLT



Hình 18: Kiểm nghiệm Định lý CLT

Code Python để chạy mô phỏng, tính toán và trực quan hoá kết quả của bài tập 3 (Kiểm nghiệm Định lý CLT với các giá trị khác nhau của p)

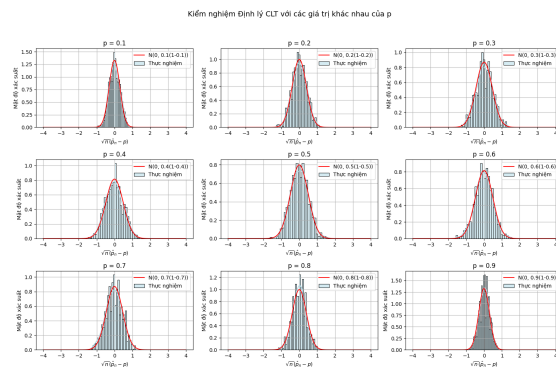
```
1
2 ### Kiểm nghiệm Định lý CLT với các giá trị xác suất khác nhau:
3 p = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]
4 import numpy as np
5 import matplotlib.pyplot as plt
6 import seaborn as sns
7 from scipy.stats import norm
8 # Các giá trị p để kiểm nghiệm
9 p_values = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]
10 n_clt = 1000 # Số lần quan sát
11 num_trials = 1000 # Số lần mô phỏng
12 # Tạo lưới 3x3 để vẽ 9 biểu đồ
13 plt.figure(figsize=(15, 10))
14 for i, p in enumerate(p_values, 1):
15     samples = np.random.binomial(1, p, (num_trials, n_clt))
16     estimates = np.mean(samples, axis=1)
17     z_scores = np.sqrt(n_clt) * (estimates - p)
18     plt.subplot(3, 3, i) # Lưới 3 hàng x 3 cột
19     sns.histplot(z_scores, bins=30, kde=True, stat="density",
20                 color='lightblue', edgecolor='black', label="Thực nghiệm")
```

```

21 x = np.linspace(-4, 4, 200)
22 std_dev = np.sqrt(p * (1 - p))
23 plt.plot(x, norm.pdf(x, 0, std_dev), 'r-', label=f'N(0, {p}(1-{p}))')
24 plt.title(f"p = {p}")
25 plt.xlabel(r"$\sqrt{n}(\hat{p}_n - p)$")
26 plt.ylabel("Mật độ xác suất")
27 plt.legend()
28 plt.grid(True)
29 plt.suptitle("Kiểm nghiệm Định lý Giới hạn Trung tâm với các giá trị khác
30 nhau của p", fontsize=14)
31 plt.tight_layout(rect=[0, 0, 1, 0.95])
32 plt.show()

```

Kiểm nghiệm Định lý CLT với các giá trị khác nhau của p



Hình 19: Kiểm nghiệm Định lý CLT với các giá trị khác nhau của p

Bài tập 4. Assume that we draw m samples x_i from a probability distribution with zero mean and unit variance. Compute the averages $z_m \stackrel{\text{def}}{=} m^{-1} \sum_{i=1}^m x_i$. Can we apply Chebyshev's inequality for every z_m independently? Why not?

Tóm tắt:

- Cho m mẫu x_1, x_2, \dots, x_m được lấy từ một phân phối xác suất có:
 - Kỳ vọng $E[x_i] = 0$ (trung bình bằng 0).
 - Phương sai $\text{Var}(x_i) = 1$ (phương sai đơn vị).

– Trung bình mẫu:

$$z_m = \frac{1}{m} \sum_{i=1}^m x_i$$

- Question: Có thể áp dụng bất đẳng thức Chebyshev cho từng z_m một cách độc lập không?
Tại sao?

Giải bài tập 4:

1. Ta cần tính kỳ vọng và phương sai của z_m .

Vì các x_i có kỳ vọng $E[x_i] = 0$ và phương sai $\text{Var}(x_i) = 1$, và các mẫu là độc lập, ta có:

- Kỳ vọng:

Chúng ta áp dụng tính chất tuyến tính của kỳ vọng, đó là:

$$\mathbb{E}[aX + bY] = a\mathbb{E}[X] + b\mathbb{E}[Y]$$

Cho nên:

$$\mathbb{E}[z_m] = \mathbb{E}\left[\sum_{i=1}^m x_i\right] = \sum_{i=1}^m \mathbb{E}[x_i] = \frac{1}{m} \cdot (0 + 0 + \dots + 0) = 0$$

- Phương sai:

Phương sai tổng của hai biến ngẫu nhiên:

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y) + 2 \cdot \text{Cov}(X, Y)$$

Nếu X và Y là độc lập, thì $\text{Cov}(X, Y) = 0$, và công thức rút gọn thành:

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y)$$

Áp dụng:

$$\text{Nếu } Z = aX, \text{ thì } \text{Var}(Z) = a^2 \cdot \text{Var}(X)$$

$$\text{Var}(z_m) = \text{Var}\left(\frac{1}{m} \sum_{i=1}^m x_i\right) = \left(\frac{1}{m}\right)^2 \cdot \text{Var}\left(\sum_{i=1}^m x_i\right) = \frac{1}{m^2} \cdot m = \frac{1}{m}$$

2. Có thể áp dụng bất đẳng thức Chebyshev cho từng z_m không?

Có vì:

Với mỗi giá trị cụ thể của z_m , ta có thể áp dụng công thức Chebyshev:

$$\mathbb{P}(|z_m| \geq \varepsilon) \leq \frac{1}{m\varepsilon^2}$$

Nếu $m = 100, \varepsilon = 0.1$, thì: $\mathbb{P}(|z_{100}| \geq 0.1) \leq \frac{1}{100 \cdot 0.01} = 1$

Nếu $m = 1000, \varepsilon = 0.1$, thì: $\mathbb{P}(|z_{1000}| \geq 0.1) \leq \frac{1}{1000 \cdot 0.01} = 0.1$

```
1  # Mã nguồn code minh họa
2  import numpy as np
3  def simulate_chebyshev(m_values, epsilon, num_trials=10000):
4      results = {}
5      for m in m_values:
6          # Sinh num_trials mẫu trung bình từ m biến ngẫu nhiên phân phối chuẩn (mean=0, var=1)
7          samples = np.random.normal(loc=0, scale=1, size=(num_trials, m))
8          z_m = samples.mean(axis=1)
9          # Xác suất thực nghiệm: tỷ lệ |z_m| >= epsilon
10         empirical_prob = np.mean(np.abs(z_m) >= epsilon)
11         # Giới hạn từ bất đẳng thức Chebyshev
12         chebyshev_bound = 1 / (m * epsilon**2)
13         results[m] = {
14             "Empirical Probability": empirical_prob,
15             "Chebyshev Bound": chebyshev_bound
16         }
17     return results
18 # Thử nghiệm với các giá trị m khác nhau
19 m_values = [1, 5, 10, 50, 100, 500, 5000]
20 epsilon = 0.5
21 results = simulate_chebyshev(m_values, epsilon)
22 for m, res in results.items():
23     print(f"m = {m}")
24     print(f"  Xác suất thực nghiệm    = {res['Empirical Probability']:.4f}")
25     print(f"  Giới hạn Chebyshev          = {res['Chebyshev Bound']:.4f}")
26     print()
```

```
27 Output:
28 m = 1
29   Xác suất thực nghiệm   = 0.6094
30   Giới hạn Chebyshev     = 4.0000
31 m = 5
32   Xác suất thực nghiệm   = 0.2670
33   Giới hạn Chebyshev     = 0.8000
34 m = 10
35   Xác suất thực nghiệm   = 0.1189
36   Giới hạn Chebyshev     = 0.4000
37 m = 50
38   Xác suất thực nghiệm   = 0.0003
39   Giới hạn Chebyshev     = 0.0800
40 m = 100
41   Xác suất thực nghiệm   = 0.0000
42   Giới hạn Chebyshev     = 0.0400
43 m = 500
44   Xác suất thực nghiệm   = 0.0000
45   Giới hạn Chebyshev     = 0.0080
```

Bài tập 5. Given two events with probability $P(\mathcal{A})$ and $P(\mathcal{B})$, compute upper and lower bounds on $P(\mathcal{A} \cup \mathcal{B})$ and $P(\mathcal{A} \cap \mathcal{B})$. Hint: graph the situation using a Venn diagram.

Tóm tắt:

- Cho hai biến cố A và B với xác suất $P(A)$ và $P(B)$. Hãy tìm cận trên và cận dưới cho:

– $P(A \cup B)$

– $P(A \cap B)$.

Giải bài tập 5:

1. Giải thích bằng sơ đồ Venn

- Vẽ hai hình tròn giao nhau, một đại diện cho A , một cho B . Diện tích mỗi hình tròn tương ứng với xác suất của biến cố đó.
- Phần giao nhau thể hiện $P(A \cap B)$

- Toàn bộ phần nằm trong cả hai hình tròn (không tính phần chồng 2 lần) thể hiện $P(A \cup B)$.

2. Chúng ta sẽ tìm cận trên và cận dưới của $P(A \cup B)$

Ta có công thức tổng quát:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

- Cận dưới:

Để $P(A \cup B)$ nhỏ nhất, thì $P(A \cap B)$ phải lớn nhất có thể, tức là $P(A \cap B) = \min(P(A), P(B))$

Khi đó:

$$P(A \cup B)_{\min} = P(A) + P(B) - \min(P(A), P(B)) = \max(P(A), P(B))$$

- Cận trên:

Để $P(A \cup B)$ lớn nhất, thì $P(A \cap B)$ phải nhỏ nhất có thể, tức là $P(A \cap B) = 0$ (hai biến cố rời nhau)

Khi đó:

$$P(A \cup B)_{\max} = P(A) + P(B)$$

Mà tổng này không được vượt quá 1. Vậy:

$$P(A \cup B)_{\max} = \min(1, P(A) + P(B))$$

3. Chúng ta sẽ tìm cận trên và cận dưới của $P(A \cap B)$

- Cận trên:

Giao của A và B không thể lớn hơn biến cố nhỏ hơn, nên:

$$P(A \cap B)_{\max} = \min(P(A), P(B))$$

- Cận dưới:

Từ công thức ở trên, để $P(A \cap B)$ nhỏ nhất, thì $P(A \cup B)$ phải lớn nhất, tức là:

$$P(A \cap B)_{\min} = P(A) + P(B) - \min(1, P(A) + P(B)) = \max(0, P(A) + P(B) - 1)$$

```
1  # mã nguồn code minh họa
2  def bounds_of_union_and_intersection(p_a, p_b):
3      # Kiểm tra đầu vào hợp lệ
4      if not (0 <= p_a <= 1 and 0 <= p_b <= 1):
5          raise ValueError("Xác suất phải nằm trong khoảng từ 0 đến 1")
6
7      #  $P(A \cup B)$ 
8      union_lower = max(p_a, p_b)
9      union_upper = min(1, p_a + p_b)
10
11     #  $P(A \cap B)$ 
12     intersection_lower = max(0, p_a + p_b - 1)
13     intersection_upper = min(p_a, p_b)
14
15     return {
16         "P( $A \cup B$ )": {
17             "lower_bound": union_lower,
18             "upper_bound": union_upper
19         },
20         "P( $A \cap B$ )": {
21             "lower_bound": intersection_lower,
22             "upper_bound": intersection_upper
23         }
24     }
25
26     # Ví dụ
27     p_a = 0.4
28     p_b = 0.7
29     result = bounds_of_union_and_intersection(p_a, p_b)
30
31     for event, bounds in result.items():
32         print(f"{event}:")
```



```
33 print(f" Cận dưới: {bounds['lower_bound']}")  
34 print(f" Cận trên: {bounds['upper_bound']}")
```

35

36 Output:

37 $P(A \cup B)$:

38 Cận dưới: 0.7

39 Cận trên: 1

40 $P(A \cap B)$:

41 Cận dưới: 0.100000000000000009

42 Cận trên: 0.4

43

Bài tập 6. Assume that we have a sequence of random variables, say A , B , and C , where B only depends on A , and C only depends on B , can you simplify the joint $P(A, B, C)$ probability? Hint: this is a Markov chain.

Tạm dịch: Cho A , B , C là các biến ngẫu nhiên. B chỉ phụ thuộc vào A , C chỉ phụ thuộc vào B . Đơn giản hóa $P(A, B, C)$ thế nào?

Giải bài tập 6:

C chỉ phụ thuộc vào $B \Rightarrow C$ độc lập có điều kiện với A khi đã biết B , ký hiệu là:

$$P(C|A, B) = P(C|B)$$

Theo quy tắc chuỗi (chain rule) trong xác suất:

$$P(A, B, C) = P(A).P(B|A).P(C|A, B)$$

Ta được công thức rút gọn:

$$P(A, B, C) = P(A).P(B|A).P(C|B)$$

```
1 ##### Mã nguồn code minh họa  
2 # Define the probabilities
```

```
3 P_A = {
4     'a1': 0.5,
5     'a2': 0.5,
6 }
7
8 P_B_given_A = {
9     'a1': {'b1': 0.4, 'b2': 0.6},
10    'a2': {'b1': 0.7, 'b2': 0.3},
11 }
12
13 P_C_given_B = {
14     'b1': {'c1': 0.8, 'c2': 0.2},
15     'b2': {'c1': 0.1, 'c2': 0.9},
16 }
17
18 # Calculate joint probability for a specific path: A='a1', B='b1', C='c1'
19 a = 'a1'
20 b = 'b1'
21 c = 'c1'
22
23 joint_prob = P_A[a] * P_B_given_A[a][b] * P_C_given_B[b][c]
24
25 print(f"P(A={a}, B={b}, C={c}) = {joint_prob}")
26 # P(A=a1, B=b1, C=c1) = 0.16000000000000003
```

Bài tập 7. Trong Mục 2.6.5, giả sử rằng kết quả của hai xét nghiệm không độc lập với nhau. Cụ thể, giả sử rằng mỗi xét nghiệm riêng lẻ có tỉ lệ dương tính giả là 10% và tỉ lệ âm tính giả là 1%. Tức là, giả sử rằng:

- $P(D = 1 \mid H = 0) = 0.1$ (xét nghiệm cho kết quả dương tính khi bệnh nhân không bị bệnh)
- $P(D = 0 \mid H = 1) = 0.01$ (xét nghiệm cho kết quả âm tính khi bệnh nhân thật sự bị bệnh)

Hơn nữa, giả sử rằng đối với trường hợp bệnh nhân bị nhiễm bệnh ($H = 1$), kết quả hai xét nghiệm là độc lập có điều kiện, tức là:

$$P(D_1, D_2 | H = 1) = P(D_1 | H = 1) \cdot P(D_2 | H = 1)$$

Nhưng đối với bệnh nhân khỏe mạnh ($H = 0$), kết quả hai xét nghiệm phụ thuộc theo phân phối:

$$P(D_1 = D_2 = 1 | H = 0) = 0.02$$

1. Hãy xây dựng bảng xác suất liên hợp (joint probability table) cho D_1 và D_2 , biết rằng $H = 0$, dựa trên thông tin đã cho ở trên.
2. Suy ra xác suất bệnh nhân bị bệnh ($H = 1$) sau khi một xét nghiệm cho kết quả dương tính. Bạn có thể giả sử xác suất tiên nghiệm (prior) $P(H = 1) = 0.0015$ như trước.
3. Suy ra xác suất bệnh nhân bị bệnh ($H = 1$) sau khi cả hai xét nghiệm đều dương tính.

Tóm tắt bài toán

Biến cố:

- $H = 1$: Bệnh nhân mắc bệnh (nhiễm bệnh).
- $H = 0$: Bệnh nhân khỏe mạnh.
- D_1, D_2 : Kết quả của hai xét nghiệm ($D_i = 1$ nếu dương tính, $D_i = 0$ nếu âm tính).

Giả định:

- Xác suất:
 - Xác suất tiên nghiệm: $P(H = 1) = 0.0015$
 $\Rightarrow P(H = 0) = 0.9985$.
 - Xét nghiệm dương tính giả (False Positive): $P(D_i = 1 | H = 0) = 0.1$.
 \Rightarrow Xét nghiệm âm tính thật (True Positive): $P(D_i = 0 | H = 0) = 0.9$.
 - Xét nghiệm âm tính giả (False Negative): $P(D_i = 0 | H = 1) = 0.01$.
 \Rightarrow Xét nghiệm dương tính thật (True positive): $P(D_i = 1 | H = 1) = 0.99$.
- Tính độc lập có điều kiện:

– Nếu $H = 1$: D_1 và D_2 độc lập, tức:

$$P(D_1, D_2 \mid H = 1) = P(D_1 \mid H = 1)P(D_2 \mid H = 1).$$

– Nếu $H = 0$: D_1 và D_2 không độc lập, với:

$$P(D_1 = D_2 = 1 \mid H = 0) = 0.02.$$

Câu hỏi:

1. Tính bảng xác suất chung cho D_1 và D_2 khi đã biết $H = 0$ dựa trên các thông tin trên.
2. Suy ra xác suất bệnh nhân thực sự bị bệnh ($H = 1$) sau khi một xét nghiệm cho kết quả dương tính. (Giả sử xác suất ban đầu $P(H = 1) = 0.0015$ như trước.)
3. Suy ra xác suất bệnh nhân bị bệnh ($H = 1$) sau khi cả hai xét nghiệm đều cho kết quả dương tính.

Giải bài tập 7:

7.1. Bảng xác suất đồng thời cho $P(D_1, D_2 \mid H = 0)$.

Ta có: Với $H = 0$, hai xét nghiệm không độc lập nhưng có thể mô hình hóa thông qua xác suất đồng thời. Ta cần điền các giá trị còn lại của bảng:

$D_1 \backslash D_2$	$D_2 = 0$	$D_2 = 1$	Tổng
$D_1 = 0$	a	b	0.9
$D_1 = 1$	c	0.02	0.1
Tổng	0.9	0.1	1

Bảng 3: Bảng xác suất đồng thời

Tính toán:

- $c = P(D_1 = 1 \mid H = 0) - P(D_1 = D_2 = 1 \mid H = 0) = 0.1 - 0.02 = 0.08.$
- $b = P(D_2 = 1 \mid H = 0) - P(D_1 = D_2 = 1 \mid H = 0) = 0.1 - 0.02 = 0.08.$
- $a = 0.9 - b = 0.9 - 0.08 = 0.82.$

Kết quả:

$D_1 \backslash D_2$	$D_2 = 0$	$D_2 = 1$	Tổng
$D_1 = 0$	0.82	0.08	0.9
$D_1 = 1$	0.08	0.02	0.1
Tổng	0.9	0.1	1

Bảng 4: Bảng xác suất đồng thời

7.2. Xác suất bệnh sau một xét nghiệm dương tính ($P(H = 1 | D_1 = 1)$) Công thức Bayes:

$$P(H = 1 | D_1 = 1) = \frac{P(D_1 = 1 | H = 1)P(H = 1)}{P(D_1 = 1)}$$

Tính các thành phần:

- $P(D_1 = 1 | H = 1) = 1 - P(D_1 = 0 | H = 1) = 1 - 0.01 = 0.99$.
- $P(H = 1) = 0.0015$ (đề bài cho) $\Rightarrow P(H = 0) = 0.9985$.
- $P(D_1 = 1) = P(D_1 = 1 | H = 1)P(H = 1) + P(D_1 = 1 | H = 0)P(H = 0)$
- $P(D_1 = 1) = 0.99 \times 0.0015 + 0.1 \times 0.9985 \approx 0.101485$.

Kết quả:

$$P(H = 1 | D_1 = 1) = \frac{0.99 \times 0.0015}{0.101485} \approx 0.0146 \quad (1.46\%).$$

7.3. Xác suất bệnh sau hai xét nghiệm dương tính ($P(H = 1 | D_1 = D_2 = 1)$)

Ta có công thức Bayes:

$$P(H = 1 | D_1 = D_2 = 1) = \frac{P(D_1 = D_2 = 1 | H = 1)P(H = 1)}{P(D_1 = D_2 = 1)}.$$

Do D_1, D_2 độc lập khi $H = 1$, ta có:

$$P(D_1 = D_2 = 1 | H = 1) = P(D_1 = 1 | H = 1)^2 = 0.99^2 = 0.9801.$$

- Tử số: $P(D_1 = D_2 = 1 | H = 1)P(H = 1) = 0.9801 \times 0.0015 \approx 0.00147$.
- Mẫu số:

$$P(D_1 = D_2 = 1) = P(D_1 = D_2 = 1 | H = 1)P(H = 1) + P(D_1 = D_2 = 1 | H = 0)P(H = 0)$$

$$P(D_1 = D_2 = 1) = 0.9801 \times 0.0015 + 0.02 \times (1 - 0.0015) \approx 0.02147.$$

- Kết quả:

$$P(H = 1 \mid D_1 = D_2 = 1) = \frac{0.9801 \times 0.0015}{0.9801 \times 0.0015 + 0.02 \times (1 - 0.0015)} \approx 0.0685 \quad (6.85\%).$$

Vậy, sau hai xét nghiệm đều dương tính, xác suất bệnh nhân thực sự bị nhiễm là khoảng 6.85%. Dù xác suất này nhỏ hơn 10%, nhưng vẫn lớn hơn nhiều so với xác suất ban đầu 0.15%.

Code Python cho bài tập 7

```
1 def bayes_single_test(P_H1, false_pos, false_neg):
2     Tính xác suất P(H=1 | D=1) sau một test dương tính.
3     P_H0 = 1 - P_H1
4     P_D_pos_given_H1 = 1 - false_neg
5     P_D_pos_given_H0 = false_pos
6     numerator = P_D_pos_given_H1 * P_H1
7     denominator = numerator + P_D_pos_given_H0 * P_H0
8     posterior = numerator / denominator
9     return posterior
10 def bayes_double_test(P_H1, false_pos, false_neg, P_D1_D2_1_H0):
11     # Tính xác suất P(H=1 | D1=1, D2=1) sau hai test dương tính.
12     P_H0 = 1 - P_H1
13     P_D1_1_given_H1 = 1 - false_neg
14     P_D2_1_given_H1 = 1 - false_neg
15     # Vì D1 và D2 độc lập có điều kiện H=1
16     P_D1_D2_1_H1 = P_D1_1_given_H1 * P_D2_1_given_H1
17     numerator = P_D1_D2_1_H1 * P_H1
18     denominator = numerator + P_D1_D2_1_H0 * P_H0
19     posterior = numerator / denominator
20     return posterior
21 def compute_joint_prob_H0(false_pos, P_D1_eq_D2_1_given_H0):
22     #Tự động suy ra bảng xác suất joint P(D1, D2 | H=0) từ false positive rate
23     #và xác suất đồng thời D1=D2=1
24     p11 = P_D1_eq_D2_1_given_H0
```

```
25     p10 = false_pos - p11
26     p01 = false_pos - p11
27     p00 = 1 - (p11 + p10 + p01)
28     return {
29         (0, 0): p00,
30         (0, 1): p01,
31         (1, 0): p10,
32         (1, 1): p11
33     }
34 # ===== INPUT TỪ ĐỀ BÀI BÀI TOÁN 7 =====
35 P_H1 = 0.0015 # Xác suất bệnh
36 false_positive = 0.1
37 false_negative = 0.01
38 P_D1_eq_D2_eq_1_given_H0 = 0.02
39 # ===== LỜI GIẢI CHO BÀI TOÁN 7 =====
40 # Câu 1: joint probability table
41 joint_table = compute_joint_prob_H0(false_positive, P_D1_eq_D2_eq_1_given_H0)
42 print("Joint probability table for D1, D2 given H=0:")
43 for (d1, d2), prob in joint_table.items():
44     print(f"P(D1={d1}, D2={d2} | H=0) = {prob:.4f}")
45 # Câu 2: P(H=1 | D=1)
46 posterior_1 = bayes_single_test(P_H1, false_positive, false_negative)
47 print(f"\nP(H=1 | D=1) = {posterior_1:.4f} (~{posterior_1 * 100:.2f}%)")
48 # Câu 3: P(H=1 | D1=1, D2=1)
49 posterior_2 = bayes_double_test(P_H1, false_positive, false_negative,
50 P_D1_eq_D2_eq_1_given_H0)
51 print(f"P(H=1 | D1=1, D2=1) = {posterior_2:.4f} (~{posterior_2 * 100:.2f}%)")
```

Kết quả của chạy Code Python để tính toán, trực quan hóa kết quả cho bài tập 7

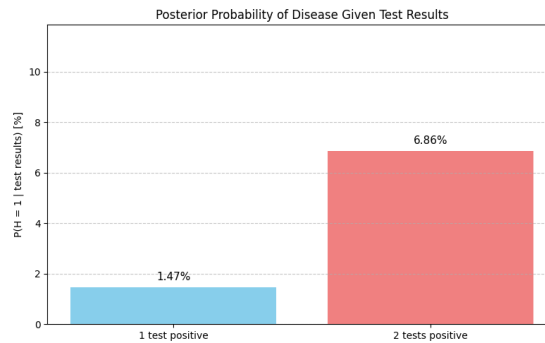
```
Bảng xác suất với H=0:
P(D1=0, D2=0 | H=0) = 0.82000
P(D1=0, D2=1 | H=0) = 0.08000
P(D1=1, D2=0 | H=0) = 0.08000
P(D1=1, D2=1 | H=0) = 0.02000

Xác suất bệnh khi 1 test dương tính:
P(H=1 | D1=1) ≈ 0.01465

Xác suất bệnh khi 2 test đều dương tính:
P(H=1 | D1=1, D2=1) ≈ 0.06857
```

Hình 20: Kết quả bài tập 7(Python)

Trực quan hóa hai xác suất hậu nghiệm $P(H = 1 | D = 1)$ và $P(H = 1 | D_1 = 1, D_2 = 1)$ bằng biểu đồ cột



Hình 21: Biểu đồ trực quan hóa hai xác suất hậu nghiệm

Mô phỏng bài toán bằng thử nghiệm ngẫu nhiên (Monte Carlo)
Cho kết quả tương đồng với lời giải ở trên

Monte Carlo estimate of $P(H = 1 | D_1 = 1, D_2 = 1) = 0.0681 (\approx 6.81\%)$

Python code mô phỏng bài toán bằng thử nghiệm ngẫu nhiên (Monte Carlo)

```
1 import numpy as np
2 def monte_carlo_simulation(n_trials, P_H1, false_pos, false_neg, P_D1_D2_1_H0):
3     # Mô phỏng Monte Carlo để ước lượng P(H=1 | D1=1, D2=1)
4     # Tạo mảng ngẫu nhiên xác định người bệnh (H=1) hay khỏe (H=0)
```



```
5     H = np.random.rand(n_trials) < P_H1
6     # Khởi tạo mảng lưu kết quả test D1 và D2
7     D1 = np.zeros(n_trials, dtype=int)
8     D2 = np.zeros(n_trials, dtype=int)
9     # Với H = 1 (bệnh): D1, D2 độc lập, xác suất đúng dương = 1 - false_negative
10    idx_H1 = np.where(H)[0]
11    D1[idx_H1] = np.random.rand(len(idx_H1)) < (1 - false_neg)
12    D2[idx_H1] = np.random.rand(len(idx_H1)) < (1 - false_neg)
13    # Với H = 0 (khỏe): tạo D1, D2 có phụ thuộc để giữ đúng  $P(D1=1, D2=1 | H=0) = 0.02$ 
14    idx_H0 = np.where(~H)[0]
15    for i in idx_H0:
16        r = np.random.rand()
17        if r < P_D1_D2_1_H0:
18            D1[i] = 1
19            D2[i] = 1
20        elif r < P_D1_D2_1_H0 + (false_pos - P_D1_D2_1_H0): # D1=1, D2=0
21            D1[i] = 1
22            D2[i] = 0
23        elif r < P_D1_D2_1_H0 + 2 * (false_pos - P_D1_D2_1_H0): # D1=0, D2=1
24            D1[i] = 0
25            D2[i] = 1
26        else:
27            D1[i] = 0
28            D2[i] = 0
29    # Đếm số trường hợp D1=1 và D2=1
30    both_positive = (D1 == 1) & (D2 == 1)
31    count_both_positive = np.sum(both_positive)
32    count_H1_given_both_positive = np.sum(H[both_positive])
33    # Tính xác suất hậu nghiệm
34    posterior_estimate = count_H1_given_both_positive / count_both_positive
35    return posterior_estimate
36    # ===== THÔNG SỐ ĐẦU VÀO =====
```

```
37 P_H1 = 0.0015
38 false_pos = 0.1
39 false_neg = 0.01
40 P_D1_D2_1_H0 = 0.02
41 n_simulations = 10**7 # 10 triệu thử nghiệm
42 # ===== CHẠY MÔ PHỎNG VÀ XUẤT KẾT QUẢ =====
43 posterior_mc = monte_carlo_simulation(n_simulations, P_H1, false_pos, false_neg,
44 P_D1_D2_1_H0)
45 print(f"\nMonte Carlo estimate of P(H=1 | D1=1, D2=1): {posterior_mc:.4f}
46 (~{posterior_mc * 100:.2f}%)")
```

Bài tập 8. Giả sử bạn là một quản lý tài sản tại một ngân hàng đầu tư và bạn có nhiều lựa chọn cổ phiếu s_i để đầu tư. Danh mục đầu tư của bạn cần có tổng trọng số bằng 1 ($\sum_{i=1}^n \alpha_i = 1$), với trọng số α_i cho mỗi cổ phiếu. Các cổ phiếu có mức lợi nhuận trung bình

$$\mu = E_{s \sim P}[s]$$

và hiệp phương sai

$$\Sigma = \text{Cov}_{s \sim P}[s].$$

1. Tính toán lợi nhuận kỳ vọng cho một danh mục đầu tư s đã cho.
2. Nếu muốn tối đa hóa lợi nhuận của danh mục đầu tư, nên phân bổ đầu tư như thế nào?
3. Tính toán phương sai của danh mục đầu tư.
4. Xây dựng bài toán tối ưu để tối đa hóa lợi nhuận trong khi giữ phương sai ở một ngưỡng nhất định. Đây chính là danh mục đầu tư Markowitz đã đoạt giải Nobel (Mangram, 2013). Để giải bài toán này, sẽ cần sử dụng một trình giải tối ưu bậc hai (quadratic programming solver), một công cụ vượt xa phạm vi của cuốn sách này.

Giải bài tập 8:

Ta có:

- Lợi nhuận trung bình của cổ phiếu: $\mu = E_{s \sim P}[s]$.
- Ma trận hiệp phương sai: $\Sigma = \text{Cov}_{s \sim P}[s]$.

- Danh mục đầu tư $\mathbf{s} = \{s_1, s_2, \dots, s_n\}$: Gồm các cổ phiếu s_i với trọng số tương ứng α_i , $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$, với ràng buộc:

$$\sum_{i=1}^n \alpha_i = 1$$

- Lợi nhuận trung bình của cổ phiếu: $\mu = E_{s \sim P}[\mathbf{s}]$.

- Ma trận hiệp phương sai: $\Sigma = \text{Cov}_{s \sim P}[\mathbf{s}]$, với vector ngẫu nhiên $\mathbf{s} = [s_1, s_2, \dots, s_n]^T$, ma trận Σ có dạng:

$$\Sigma = \begin{bmatrix} \text{Var}(s_1) & \text{Cov}(s_1, s_2) & \dots & \text{Cov}(s_1, s_n) \\ \text{Cov}(s_2, s_1) & \text{Var}(s_2) & \dots & \text{Cov}(s_2, s_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}(s_n, s_1) & \text{Cov}(s_n, s_2) & \dots & \text{Var}(s_n) \end{bmatrix}$$

8.1. Tính lợi nhuận kỳ vọng của danh mục:

- Lợi nhuận kỳ vọng của danh mục là trung bình trọng số của lợi nhuận các cổ phiếu: - Mỗi cổ phiếu s_i đóng góp tỷ suất sinh lợi trung bình là μ_i , đầu tư trọng số α_i vào nó, nên lợi nhuận kỳ vọng của danh mục là

$$E[\mathbf{s}] = \sum_{i=1}^n \alpha_i \mu_i = \alpha^\top \mu$$

- Ví dụ: Nếu danh mục gồm 2 cổ phiếu với $\alpha = \{0.6, 0.4\}$ và $\mu = \{0.1, 0.05\}$, thì:

$$E[\alpha] = 0.6 \times 0.1 + 0.4 \times 0.05 = 0.08 \quad (8\%).$$

8.2. Tối đa hóa lợi nhuận:

Để tối đa hóa lợi nhuận kỳ vọng của danh mục đầu tư, ta cần phân bổ toàn bộ vốn vào cổ phiếu có lợi nhuận kỳ vọng lớn nhất. Nếu không xét đến rủi ro (phương sai), lời giải đơn giản là:

$$\alpha_j = \begin{cases} 1 & \text{nếu } j = \arg \max_i \mu_i \\ 0 & \text{với } i \neq j \end{cases}$$

Trong đó, μ_i là lợi nhuận kỳ vọng của cổ phiếu s_i .

Tuy nhiên, cần lưu ý rằng ta chưa xét đến đến rủi ro và lời giải này chỉ đúng trong trường hợp:

- Chỉ quan tâm đến lợi nhuận kỳ vọng (không xét phương sai).

- Không có ràng buộc khác ngoài $\sum \alpha_i = 1$ và $\alpha_i \geq 0$ (không bán khống).

Trong thực tế, nhà đầu tư thường cân bằng giữa lợi nhuận và rủi ro, do đó cần xem xét thêm phương sai danh mục (ở phần 8.3) hoặc giải bài toán Markowitz (ở phần 8.4). 8.3. Tính phương sai của danh mục đầu tư: Phương sai danh mục phản ánh độ biến động:

$$\text{Var}(\alpha^T \mathbf{s}) = \alpha^T \Sigma \alpha = \sum_{i,j} \alpha_i \alpha_j \Sigma_{i,j}.$$

Ví dụ: Nếu $\Sigma = \begin{bmatrix} 0.04 & 0.01 \\ 0.01 & 0.02 \end{bmatrix}$ và $\alpha = [0.6, 0.4]^T$:

$$\text{Var}(\alpha^T \mathbf{s}) = 0.6^2 \times 0.04 + 2 \times 0.6 \times 0.4 \times 0.01 + 0.4^2 \times 0.02 = 0.0203.$$

8.4. Bài toán tối ưu danh mục Markovitz:

Tối đa hóa lợi nhuận với ràng buộc rủi ro tối đa σ_{\max}^2 :

$$\begin{aligned} \max_{\alpha} \quad & \alpha^T \mu \\ \text{subject to} \quad & \alpha^T \Sigma \alpha \leq \sigma_{\max}^2, \\ & \sum \alpha_i = 1, \\ & \alpha_i \geq 0 \quad (\text{nếu không cho phép bán khống}). \end{aligned}$$

Để mô phỏng cho bài tập 8: Thực hiện lấy dữ liệu từ nguồn thị trường chính khoán Việt Nam (import vnstock as vn). Đặc tả dữ liệu:

- Chạy thực nghiệm trên 5 mã chứng khoán: VCB, FPT, MWG, VNM, HPG.
- Thời điểm lấy dữ liệu từ ngày 01/01/2023 đến ngày 01/01/2024.
- Điểm dữ liệu: là giá trị trung bình giao dịch trong 1 ngày của tất cả các phiên khớp lệnh trong ngày đó. Có 250 ngày giao dịch \Rightarrow có 250 điểm dữ liệu.

Kết quả chạy thực nghiệm cho bài tập 8:

1. (8.1) Lợi nhuận kỳ vọng hàng ngày và hàng năm của danh mục

- Lợi suất kỳ vọng hàng ngày của danh mục: 0.1898%
- Lợi suất kỳ vọng hàng năm của danh mục: 47.8326%

2. (8.2) Tối đa hóa lợi nhuận danh mục đầu tư

- Tối đa hóa lợi nhuận danh mục:
 - VCB: 0.00%
 - FPT: 100.00%
 - MWG: 0.00%
 - VNM: 0.00%
 - HPG: 0.00%
- Lợi suất kỳ vọng danh mục tối đa lợi nhuận: 0.19%
- Rủi ro tương ứng (độ lệch chuẩn): 1.96%

3. (8.3) Tính toán phương sai của danh mục đầu tư

- Phương sai danh mục Markowitz: 0.000210

4. (8.4) Xây dựng bài toán tối ưu để tối đa hóa lợi nhuận trong khi giữ phương sai ở một ngưỡng nhất định (Tối đa hóa lợi nhuận với ràng buộc phương sai).

- Danh mục tối ưu với ràng buộc phương sai:
 - VCB: 0.00%
 - FPT: 100.00%
 - MWG: 0.00%
 - VNM: 0.00%
 - HPG: 0.00%
- Lợi suất kỳ vọng: 0.19%
- Phương sai danh mục: 0.000383
- Rủi ro tương ứng (độ lệch chuẩn): 1.96%

Code Python cho bài tập 8 (Trực quan hóa danh mục đầu tư tối ưu bằng Bar Chart)

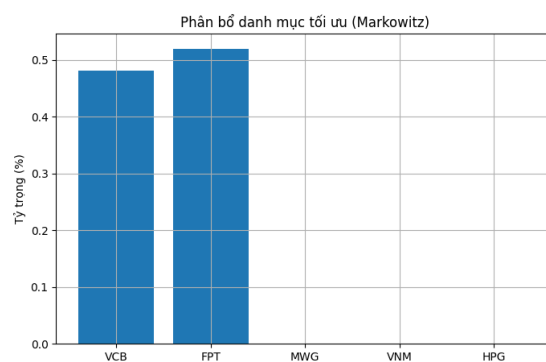
```
1  ## Chọn 5 mã từ danh mục
2  stock = vn.Vnstock().stock(source='TCBS')
3  stock.listing.all_symbols()
4  import vnstock as vn
```

```
5 import pandas as pd
6 import numpy as np
7 import matplotlib.pyplot as plt
8 import cvxpy as cp
9 # Đọc dữ liệu từ file CSV
10 df = pd.read_csv('vietnam_stock_prices_2023_2024_5ma.csv', index_col=0,
11 parse_dates=True)
12 # Tính lợi suất hàng ngày (daily returns)
13 returns = df.pct_change().dropna()
14 # Tính kỳ vọng lợi suất trung bình và ma trận hiệp phương sai
15 mu = returns.mean().values
16 Sigma = returns.cov().values
17 # Số lượng tài sản
18 n = len(mu)
19 # Khởi tạo biến trọng số danh mục
20 w = cp.Variable(n)
21 # Bài toán Markowitz: tối ưu hóa tỷ lệ Sharpe (không có rủi ro)
22 risk_aversion = 1 # hệ số điều chỉnh giữa lợi nhuận và rủi ro
23 # Hàm mục tiêu: cực tiểu hóa rủi ro - tối đa hóa lợi nhuận
24 objective = cp.Maximize(mu @ w - risk_aversion * cp.quad_form(w, Sigma))
25 # Ràng buộc: tổng trọng số = 1 và mỗi trọng số không âm
26 constraints = [cp.sum(w) == 1, w >= 0]
27 # Giải bài toán tối ưu
28 prob = cp.Problem(objective, constraints)
29 prob.solve()
30 # Kết quả
31 optimal_weights = w.value
32 portfolio_return = mu @ optimal_weights
33 portfolio_risk = np.sqrt(optimal_weights.T @ Sigma @ optimal_weights)
34 # --- Tính lợi nhuận kỳ vọng của danh mục ---
35 # Lợi nhuận kỳ vọng hàng ngày đã có:
36 expected_return_daily = np.dot(mu, optimal_weights)
```

```
37 # Giả định có 252 ngày giao dịch/năm
38 expected_return_annual = expected_return_daily * 252
39 print(f"\nLợi suất kỳ vọng hàng ngày của danh mục: {expected_return_daily:.4%}")
40 print(f"Lợi suất kỳ vọng hàng năm của danh mục: {expected_return_annual:.2%}")
41 # --- Tối đa hóa lợi nhuận danh mục đầu tư ---
42 # Khởi tạo biến trọng số mới
43 w_max_return = cp.Variable(n)
44 # Hàm mục tiêu: tối đa hóa lợi nhuận kỳ vọng
45 objective_max_return = cp.Maximize(mu @ w_max_return)
46 # Ràng buộc: tổng trọng số = 1, không bán khống
47 constraints_max_return = [cp.sum(w_max_return) == 1, w_max_return >= 0]
48 # Giải bài toán
49 prob_max_return = cp.Problem(objective_max_return, constraints_max_return)
50 prob_max_return.solve()
51 # Kết quả
52 weights_max_return = w_max_return.value
53 portfolio_return_max = mu @ weights_max_return
54 portfolio_risk_max = np.sqrt(weights_max_return.T @ Sigma @ weights_max_return)
55 print("\n Tối đa hóa lợi nhuận danh mục:")
56 for ticker, weight in zip(df.columns, weights_max_return):
57     print(f"{ticker}: {weight:.2%}")
58 print(f"\n Lợi suất kỳ vọng danh mục tối đa lợi nhuận: {portfolio_return_max:.2%}")
59 print(f" Rủi ro tương ứng (độ lệch chuẩn): {portfolio_risk_max:.2%}")
60 # Tính phương sai danh mục Markowitz
61 portfolio_variance = optimal_weights.T @ Sigma @ optimal_weights
62 print(f"\nPhương sai danh mục Markowitz: {portfolio_variance:.6f}")
63 # Cài đặt ngưỡng phương sai cho phép: 0.0004 tương ứng độ lệch chuẩn  $\approx 2\%$ 
64 target_variance = 0.0004 # Bạn có thể thay đổi giá trị này
65 # Biến trọng số
66 w_risk_constrained = cp.Variable(n)
67 # Hàm mục tiêu: tối đa hóa lợi nhuận kỳ vọng
68 objective_risk_constrained = cp.Maximize(mu @ w_risk_constrained)
```

```
69 # Ràng buộc:
70 constraints_risk_constrained = [
71     cp.sum(w_risk_constrained) == 1,          # Tổng trọng số bằng 1
72     w_risk_constrained >= 0,                  # Không bán khống
73     cp.quad_form(w_risk_constrained, Sigma) <= target_variance
74     # Phương sai không vượt ngưỡng
75 ]
76 # Giải bài toán
77 prob_risk_constrained = cp.Problem(objective_risk_constrained,
78 constraints_risk_constrained) prob_risk_constrained.solve()
79 # Kết quả
80 weights_risk_constrained = w_risk_constrained.value
81 portfolio_return_risk_constrained = mu @ weights_risk_constrained
82 portfolio_variance_risk_constrained = weights_risk_constrained.T @ Sigma @
83 weights_risk_constrained
84 print("\nDanh mục tối ưu với ràng buộc phương sai:")
85 for ticker, weight in zip(df.columns, weights_risk_constrained):
86     print(f"{ticker}: {weight:.2%}")
87 print(f"\nLợi suất kỳ vọng: {portfolio_return_risk_constrained:.2%}")
88 print(f"Phương sai danh mục: {portfolio_variance_risk_constrained:.6f}")
89 print(f"Độ lệch chuẩn: {np.sqrt(portfolio_variance_risk_constrained):.2%}")
90 # Vẽ biểu đồ phân bố danh mục
91 plt.figure(figsize=(8, 5))
92 plt.bar(df.columns, optimal_weights)
93 plt.title("Phân bố danh mục tối ưu (Markowitz)")
94 plt.ylabel("Tỷ trọng (%)")
95 plt.grid(True)
96 plt.show()
```

Trực quan hóa danh mục đầu tư tối ưu bằng Bar Chart



Hình 22: Danh mục đầu tư tối ưu

8 Bài toán nâng cao

Bài tập nâng cao 1. Bài toán nâng cao dựa trên bài tập 5 (phần 2.6.8). Bài toán mô tả như sau:

Cho ba biến cố A, B, C với các xác suất đã biết:

- $P(A) = a$
- $P(B) = b$
- $P(C) = c$

Yêu cầu bài toán: Hãy tìm giới hạn trên và giới hạn dưới có thể có của $P(A \cup B \cup C)$ và $P(A \cap B \cap C)$.

Lời giải bài toán nâng cao 1

1. Giới hạn trên và dưới của $P(A \cup B \cup C)$

Ta có công thức xác suất:

$$P(A \cup B \cup C) = P(A) + P(B) + P(C) - P(A \cap B) - P(A \cap C) - P(B \cap C) + P(A \cap B \cap C)$$

Hợp ba biến cố xảy ra khi có khả năng ít nhất một trong ba biến cố xảy ra.

- (a) Giới hạn trên của $P(A \cup B \cup C)$

- Giá trị lớn nhất xảy ra khi các tập càng ít giao nhau càng tốt – tức là các phần tử trong A, B, C càng khác biệt.
- Trường hợp cực đại:
 - Nếu $A \cap B = \emptyset, A \cap C = \emptyset, B \cap C = \emptyset$, tức ba biến cố rời nhau hoàn toàn, thì:

$$P(A \cup B \cup C) = P(A) + P(B) + P(C) = a + b + c$$

- Tuy nhiên:
 - Tổng xác suất trong không gian mẫu không thể vượt quá 1.
 - Nếu $a + b + c > 1$ thì hợp không thể vượt quá 1.
- Do đó:

$$P(A \cup B \cup C) \leq \min(1, a + b + c)$$

(b) Giới hạn dưới của $P(A \cup B \cup C)$

- Giá trị nhỏ nhất sẽ xảy ra khi:
 - Các phần giao (chồng lấp) giữa các tập là **cực đại** (tức các tập trùng nhau nhiều nhất).
 - Khi đó, phần không bị trừ đi là lớn nhất \implies tổng hợp nhỏ nhất.
- Trường hợp cực đoan:
 - Nếu $A \subseteq B \subseteq C$, thì mọi phần tử trong A cũng thuộc B và C .
 - Lúc này, $A \cup B \cup C$ gần bằng C , tức chỉ bằng xác suất lớn nhất trong ba tập.
 - Do đó, một giới hạn dưới hợp lý là:

$$P(A \cup B \cup C) \geq \max(a, b, c)$$

- Nhưng vẫn chưa chặt chẽ nhất. Ta xét điều kiện cần:
 - Xác suất không thể âm: $P(A \cup B \cup C) \geq 0$
 - Tổng xác suất của ba tập là $a + b + c$, nhưng do các tập chồng nhau, một phần của xác suất được tính nhiều lần và sẽ bị loại trừ.
 - Trong trường hợp tối đa trùng nhau, phần bị trừ đi nhiều nhất là 2.
 - Vì vậy, ta có giới hạn dưới chặt hơn:

$$P(A \cup B \cup C) \geq a + b + c - 2$$

- Nhưng nếu $a + b + c - 2 < \max(a, b, c)$, thì giới hạn dưới hợp lý hơn là:

$$P(A \cup B \cup C) \geq \max(a + b + c - 2, \max(a, b, c))$$

2. Giới hạn trên và dưới của $P(A \cap B \cap C)$

Giao ba biến cố là vùng mà cả A, B, C cùng xảy ra.

(a) Giới hạn trên của $P(A \cap B \cap C)$

- Giá trị lớn nhất xảy ra khi ba tập có phần trùng nhau càng nhiều càng tốt.
- Nếu $A \subseteq B \subseteq C$, thì toàn bộ xác suất của A sẽ nằm trong B và C . Khi đó:

$$P(A \cap B \cap C) = P(A)$$

- Tương tự, nếu $C \subseteq B \subseteq A$, thì:

$$P(A \cap B \cap C) = P(C)$$

- Vậy giá trị lớn nhất của giao là nhỏ nhất trong ba xác suất.
- Do đó:

$$P(A \cap B \cap C) \leq \min(a, b, c)$$

(b) Giới hạn dưới của $P(A \cap B \cap C)$

- Giá trị nhỏ nhất là khi giao ba tập gần như trống rỗng, tức ba tập càng rời nhau càng tốt.
- Trường hợp cực đoan:
 - Nếu ba tập hoàn toàn rời nhau (disjoint), thì $P(A \cap B \cap C) = 0$
 - Đây là giới hạn thấp nhất có thể về mặt xác suất \implies luôn đúng:

$$P(A \cap B \cap C) \geq 0$$

- Nhưng nếu tổng xác suất của ba biến cố vượt quá 2 (tức $a + b + c > 2$), thì:
 - Không thể rời nhau hoàn toàn.
 - Vì toàn bộ không gian mẫu chỉ có tổng xác suất bằng 1, nên phần chồng lặp bắt buộc phải xảy ra.
 - Khi đó, phần giao ba tập phải mang một phần xác suất dương.
- Vậy trong trường hợp tổng lớn hơn 2, thì phần giao ít nhất phải là:

$$P(A \cap B \cap C) \geq a + b + c - 2$$

- Và như mọi xác suất, giá trị này không thể âm:

$$P(A \cap B \cap C) \geq \max(0, a + b + c - 2)$$

Chạy demo bài toán nâng cao 1

Code Python chạy demo cho bài toán

```
1 !pip install matplotlib matplotlib-venn
2 import random
3 import matplotlib.pyplot as plt
4 from matplotlib_venn import venn3
5
6 def simulate_and_plot(P_A, P_B, P_C, case_type, N=100000):
7     A, B, C = [], [], []
8
9     for _ in range(N):
10         r = random.random()
11
12         if case_type == "union_upper_bound": # TH1: rời nhau
13             A.append(r < P_A)
14             B.append(P_A <= r < P_A + P_B)
15             C.append(P_A + P_B <= r < P_A + P_B + P_C)
16
17         # TH2: C là tập con của B, B là tập con của A
18         elif case_type == "union_lower_bound":
19             a = r < P_A
20             b = r < P_B if a else False
21             c = r < P_C if b else False
22             A.append(a)
23             B.append(b)
24             C.append(c)
25
26         # TH3: C là tập con của B, B là tập con của A
27         elif case_type == "inter_upper_bound":
28             c = r < P_C
29             b = True if c else (r < P_B)
30             a = True if b else (r < P_A)
31             A.append(a)
32             B.append(b)
```

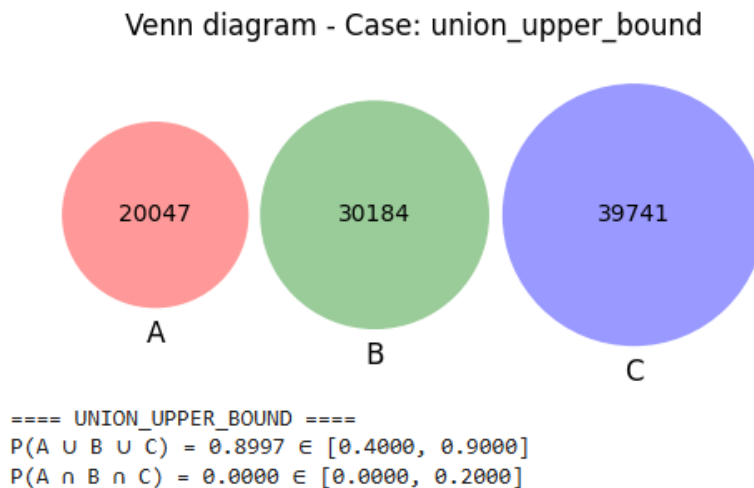
```
33         C.append(c)
34
35         elif case_type == "inter_lower_bound": # TH4: rời nhau
36             A.append(r < P_A)
37             B.append(P_A <= r < P_A + P_B)
38             C.append(P_A + P_B <= r < P_A + P_B + P_C)
39
40         # Đếm từng vùng trong biểu đồ Venn
41         only_A = sum(a and not b and not c for a, b, c in zip(A, B, C))
42         only_B = sum(b and not a and not c for a, b, c in zip(A, B, C))
43         only_C = sum(c and not a and not b for a, b, c in zip(A, B, C))
44         A_B = sum(a and b and not c for a, b, c in zip(A, B, C))
45         A_C = sum(a and c and not b for a, b, c in zip(A, B, C))
46         B_C = sum(b and c and not a for a, b, c in zip(A, B, C))
47         ABC = sum(a and b and c for a, b, c in zip(A, B, C))
48
49         # Biểu đồ Venn
50         plt.figure(figsize=(6,6))
51         venn3(subsets = (only_A, only_B, A_B, only_C, A_C, B_C, ABC),
52               set_labels = ('A', 'B', 'C'))
53         plt.title(f"Venn diagram - Case: {case_type}")
54         plt.show()
55
56         # Tính xác suất thực nghiệm
57         P_union = sum(a or b or c for a, b, c in zip(A, B, C)) / N
58         P_inter = sum(a and b and c for a, b, c in zip(A, B, C)) / N
59
60         # Cận lý thuyết
61         upper_union = min(1, P_A + P_B + P_C)
62         lower_union = max(P_A, P_B, P_C, P_A + P_B + P_C - 2)
63         upper_inter = min(P_A, P_B, P_C)
64         lower_inter = max(0, P_A + P_B + P_C - 2)
```

```

65
66     # In kết quả
67     print(f"==== {case_type.upper()} =====")
68     print(f"P(A ∪ B ∪ C) = {P_union:.4f} ∈ [{lower_union:.4f}, {upper_union:.4f}]")
69     print(f"P(A ∪ B ∩ C) = {P_inter:.4f} ∈ [{lower_inter:.4f}, {upper_inter:.4f}]\n")
70
71     # TH1: dấu bằng tại cận trên của hợp
72     simulate_and_plot(0.2, 0.3, 0.4, "union_upper_bound")
73
74     # TH2: dấu bằng tại cận dưới của hợp
75     simulate_and_plot(0.8, 0.6, 0.4, "union_lower_bound")
76
77     # TH3: dấu bằng tại cận trên của giao
78     simulate_and_plot(0.9, 0.7, 0.5, "inter_upper_bound")
79
80     # TH4: dấu bằng tại cận dưới của giao
81     simulate_and_plot(0.1, 0.3, 0.4, "inter_lower_bound")
82

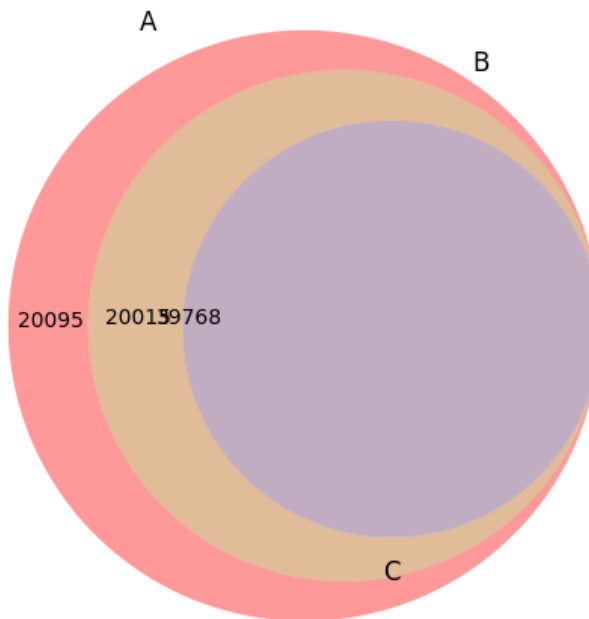
```

Kết quả xuất ra màn hình:



Hình 23: Mô phỏng trường hợp xảy ra dấu bằng tại giới hạn trên của $P(A \cup B \cup C)$

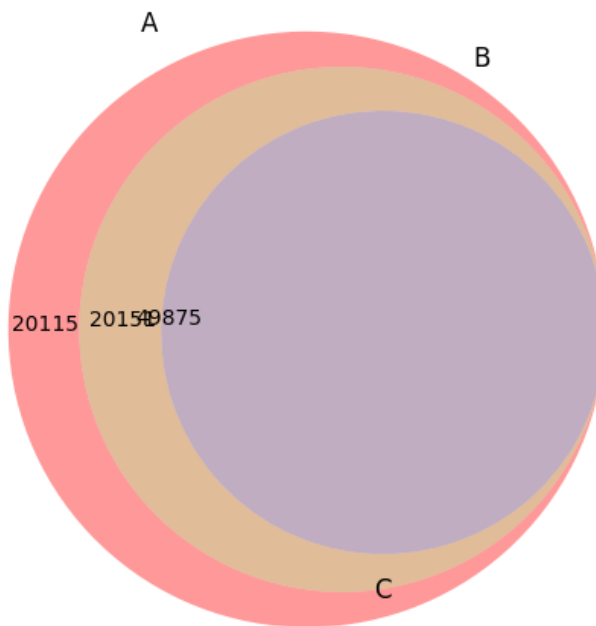
Venn diagram - Case: union_lower_bound



```
==== UNION_LOWER_BOUND ====
P(A ∪ B ∪ C) = 0.7988 ∈ [0.8000, 1.0000]
P(A ∩ B ∩ C) = 0.3977 ∈ [0.0000, 0.4000]
```

Hình 24: Mô phỏng trường hợp xảy ra dấu bằng tại giới hạn dưới của $P(A \cup B \cup C)$

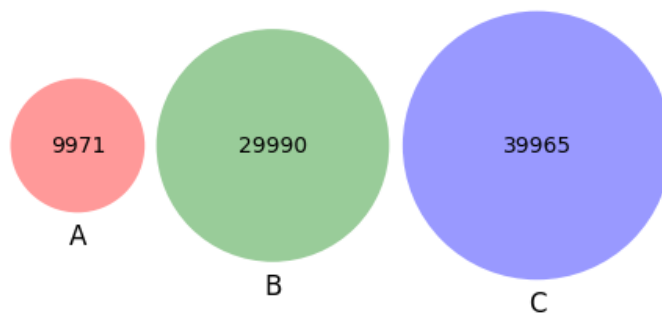
Venn diagram - Case: inter_upper_bound



==== INTER_UPPER_BOUND ====
 $P(A \cup B \cup C) = 0.9014 \in [0.9000, 1.0000]$
 $P(A \cap B \cap C) = 0.4988 \in [0.1000, 0.5000]$

Hình 25: Mô phỏng trường hợp xảy ra dấu bằng tại giới hạn trên của $P(A \cap B \cap C)$

Venn diagram - Case: inter_lower_bound



==== INTER_LOWER_BOUND ====
 $P(A \cup B \cup C) = 0.7993 \in [0.4000, 0.8000]$
 $P(A \cap B \cap C) = 0.0000 \in [0.0000, 0.1000]$

Hình 26: Mô phỏng trường hợp xảy ra dấu bằng tại giới hạn dưới của $P(A \cap B \cap C)$

Bài tập nâng cao 2. Bài toán nâng cao dựa trên bài tập 6 (phần 2.6.8). Bài toán mô tả như sau:

Bài toán: Phát hiện gian lận qua mô hình Bayes

Một công ty tài chính đang sử dụng hệ thống phát hiện gian lận giao dịch dựa trên các biến sau:

- F : Biến nhị phân cho biết giao dịch có gian lận hay không (1 = gian lận, 0 = bình thường).
- L : Biến nhị phân cho biết liệu giao dịch được thực hiện từ vị trí lạ không (1 = vị trí lạ, 0 = vị trí quen thuộc).
- T : Biến nhị phân cho biết thời điểm giao dịch có phải vào giờ bất thường không (1 = bất thường, 0 = bình thường).

Giả sử mô hình thỏa mãn mối quan hệ:

- L và T độc lập có điều kiện khi biết F .
- Sơ đồ phụ thuộc có thể được biểu diễn như:

$$F \rightarrow L$$

$$F \rightarrow T$$

Yêu cầu:

1. Viết biểu thức xác suất đồng thời $P(F, L, T)$ dựa trên cấu trúc phụ thuộc nêu trên.
2. Sử dụng định lý Bayes để viết công thức tính $P(F = 1|L = 1, T = 1)$.
3. Giả sử có các xác suất sau:

- $P(F = 1) = 0.01$
- $P(L = 1|F = 1) = 0.9, P(L = 1|F = 0) = 0.1$
- $P(T = 1|F = 1) = 0.8, P(T = 1|F = 0) = 0.2$

Tính giá trị cụ thể của $P(F = 1|L = 1, T = 1)$.

Lời giải bài toán nâng cao 2

1. Viết biểu thức xác suất đồng thời $P(F, L, T)$

Biểu thức xác suất đồng thời:

$$P(F, L, T) = P(F) \cdot P(L|F) \cdot P(T|F)$$

2. Sử dụng định lý Bayes để tính $P(F = 1|L = 1, T = 1)$

Theo định lý Bayes:

$$P(F = 1|L = 1, T = 1) = \frac{P(F = 1) \cdot P(L = 1|F = 1) \cdot P(T = 1|F = 1)}{P(L = 1, T = 1)}$$

Mẫu số $P(L = 1, T = 1)$ được tính bằng cách tổng trên tất cả giá trị của $F \in \{0, 1\}$:

$$P(L = 1, T = 1) = \sum_{f \in \{0, 1\}} P(F = f) \cdot P(L = 1|F = f) \cdot P(T = 1|F = f)$$

3. Tính giá trị cụ thể của $P(F = 1|L = 1, T = 1)$ với các giá trị xác suất đã cho

Tính mẫu số:

$$P(L = 1, T = 1) = 0.01 \cdot 0.9 \cdot 0.8 + 0.99 \cdot 0.1 \cdot 0.2 = 0.0072 + 0.0198 = 0.027$$

Tính tử số:

$$\text{Tử số} = 0.01 \cdot 0.9 \cdot 0.8 = 0.0072$$

Suy ra:

$$P(F = 1|L = 1, T = 1) = \frac{0.0072}{0.027} \approx 0.2667$$

Ta nhận thấy mặc dù khả năng gian lận gốc chỉ là 1%, nhưng khi thấy **vị trí lạ + thời điểm bất thường**, xác suất gian lận tăng lên $\approx 26.67\%$!

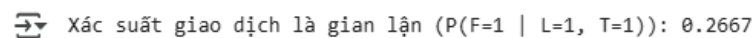
Chạy demo bài toán nâng cao 2

Code Python chạy demo cho bài toán

```
1 # Xác suất đã cho
2 P_F1 = 0.01
3 P_F0 = 1 - P_F1
```

```
4
5 P_L1_given_F1 = 0.9
6 P_L1_given_F0 = 0.1
7
8 P_T1_given_F1 = 0.8
9 P_T1_given_F0 = 0.2
10
11 # Tính tử số:  $P(F=1, L=1, T=1)$ 
12 numerator = P_F1 * P_L1_given_F1 * P_T1_given_F1
13
14 # Tính mẫu số:  $P(L=1, T=1)$ 
15 P_L1_T1 = (
16     P_F1 * P_L1_given_F1 * P_T1_given_F1 +
17     P_F0 * P_L1_given_F0 * P_T1_given_F0
18 )
19
20 # Bayes:  $P(F=1 \mid L=1, T=1)$ 
21 P_F1_given_L1_T1 = numerator / P_L1_T1
22
23 print(f"Xác suất giao dịch là gian lận ( $P(F=1 \mid L=1, T=1)$ ):
24 {P_F1_given_L1_T1:.4f}")
25
```

Kết quả xuất ra màn hình:



Hình 27: Kết quả thực hiện demo tính toán

Bài tập nâng cao 3. Bài nâng cao dựa trên bài toán xác suất xét nghiệm y tế ở EXERCISE 7, kết hợp thêm các yếu tố thực tế như chi phí xét nghiệm, tối ưu quyết định, và mở rộng thành bài toán ra quyết định dựa trên Bayesian decision theory.

Tóm tắt bài toán - INPUT bài toán

1. $H = 1$: Bệnh nhân mắc bệnh (nhiễm bệnh).
2. $H = 0$: Bệnh nhân khỏe mạnh.
3. D_i : Kết quả của xét nghiệm thứ i ($D_i = 1$ kết quả xét nghiệm dương tính, $D_i = 0$ kết quả xét nghiệm âm tính).
4. Xác suất:
 - Xác suất tiên nghiệm: $P(H = 1) = 0.0015$ ($\Rightarrow P(H = 0) = 0.9985$).
 - Xét nghiệm dương tính giả (False Positive): $P(D_i = 1 | H = 0) = 0.1$.
 - \Rightarrow Xét nghiệm âm tính thật (True Positive): $P(D_i = 0 | H = 0) = 0.9$.
 - Xét nghiệm âm tính giả (False Negative): $P(D_i = 0 | H = 1) = 0.01$.
 - \Rightarrow Xét nghiệm dương tính thật (True positive): $P(D_i = 1 | H = 1) = 0.99$.
5. Xét nghiệm độc lập theo H :
 - $P(D_1, D_2, \dots, D_k | H = 1) = P(D_1 | H = 1)P(D_2 | H = 1) \dots P(D_k | H = 1)$
 - $P(D_1, D_2, \dots, D_k | H = 0) = P(D_1 | H = 0)P(D_2 | H = 0) \dots P(D_k | H = 0)$
6. Chi phí:
 - Điều trị đúng thì chi phí: 0\$/người
 - Điều trị nhầm người không mắc bệnh, bệnh viện phải tốn chi phí: 500\$/người
 - Bỏ sót điều trị người mắc bệnh, bệnh viện phải tốn chi phí: 10,000\$/người
 - Một lần xét nghiệm bệnh nhân, bệnh viện phải tốn chi phí: 50\$
7. Ràng buộc bài toán: Được phép xét nghiệm tối đa 3 lần cho mỗi bệnh nhân, với các lần xét nghiệm độc lập theo H . Sau mỗi lần xét nghiệm, có thể quyết định:
 - Hành động (A): Dừng lại (không xét nghiệm thêm 1 lần nào nữa) và điều trị.
 - Hành động (B): Dừng lại (không xét nghiệm thêm 1 lần nào nữa) và không điều trị.

- Hành động (C): Tiếp tục xét nghiệm lần 3 (lần cuối).

Yêu cầu - OUPUT của bài toán:

1. Tính xác suất hậu nghiệm $P(H = 1|D_1, D_2, \dots, D_k)$ sau mỗi bước.
2. Tính chi phí kỳ vọng cho mỗi hành động A, B, C.
3. Mô phỏng thuật toán giúp chọn hành động tối ưu ở mỗi bước xét nghiệm nhằm giảm thiểu chi phí kỳ vọng.
4. Sử dụng mô phỏng Monte Carlo để kiểm nghiệm chiến lược trên với 1 triệu bệnh nhân ngẫu nhiên.

Lời giải bài toán nâng cao 3

Câu 1: Tính xác suất hậu nghiệm $P(H = 1|D_1, D_2, \dots, D_k)$

- Mô hình hóa bài toán:
 - $H = 1$: Mắc bệnh, xác suất ban đầu $P(H = 1) = 0.0015$
 - $D_i \in \{0, 1\}$: Kết quả test thứ i ($1 =$ dương tính, $0 =$ âm tính)
 - Các test độc lập có điều kiện theo H
- Công thức Bayes: Sau k lần xét nghiệm, có s kết quả xét nghiệm dương tính:

$$P(H = 1|D_1, \dots, D_k) = \frac{P(D_1, \dots, D_k|H = 1) \cdot P(H = 1)}{P(D_1, \dots, D_k)} \quad (7)$$

Hay

$$P(H = 1|D_1, \dots, D_k) = \frac{P(D_1 = \dots = D_s = 1|H = 1) \cdot P(D_{s+1} = \dots = D_k = 0|H = 1) \cdot P(H = 1)}{P(D_1, \dots, D_k|H = 1) + P(D_1, \dots, D_k|H = 0)} \quad (8)$$

$$(D_1 = \dots = D_s = 1, D_{s+1} = \dots = D_k = 0)$$

Vì các test độc lập có điều kiện theo H , ta có:

Tử số của (8)

$$\begin{aligned}P(D_1, \dots, D_k | H = 1) &= P(D_1 = \dots = D_s = 1 | H = 1) \cdot P(D_{s+1} = \dots = D_k = 0 | H = 1) \\&= (P(D_1 = 1 | H = 1))^s \cdot (P(D_k = 0 | H = 1))^{k-s} \\&= (0.99)^s \cdot (0.01)^{k-s}\end{aligned}$$

và

$$\begin{aligned}P(D_1, \dots, D_k | H = 0) &= P(D_1 = \dots = D_s = 1 | H = 0) \cdot P(D_{s+1} = \dots = D_k = 0 | H = 0) \\&= (P(D_1 = 1 | H = 0))^s \cdot (P(D_k = 0 | H = 0))^{k-s} \\&= (0.10)^s \cdot (0.90)^{k-s}\end{aligned}$$

Do đó:

$$P(H = 1 | D_1, \dots, D_k) = \frac{(0.99)^s (0.01)^{k-s} \cdot 0.0015}{(0.99)^s (0.01)^{k-s} \cdot 0.0015 + (0.10)^s (0.90)^{k-s} \cdot 0.9985}$$
$$bayes_posterior(s, k) = P(H = 1 | D_1, \dots, D_k) \quad (9)$$

Câu 2: Tính chi phí kỳ vọng của các hành động

1. Giả định của mô hình bài toán:

- Điều trị đúng chi phí: 0\$/người
- Điều trị nhầm người không mắc bệnh, bệnh viện phải tốn chi phí: 500\$/người
- Bỏ sót điều trị người mắc bệnh, bệnh viện phải tốn chi phí: 10,000\$/người
- Một lần xét nghiệm bệnh nhân, bệnh viện phải tốn chi phí: 50\$

2. Tính chi phí kỳ vọng:

- Giả sử tại bước xét nghiệm thứ k , có số lần xét nghiệm dương tính là s , gọi hàm:
 $p = bayes_posterior(s, k)$
 - Nếu hành động (A): Điều trị thì gọi hàm tính chi phí: $cost_A = (1-p)*500 + k*50$;
 - Nếu hành động (B): Không điều trị thì gọi hàm tính chi phí: $cost_B = p*10000 + k*50$;
 - Nếu hành động (C) tiếp tục test \rightarrow xây dựng cây quyết định đệ quy ở Câu 3.

Câu 3: Mô phỏng thuật toán giúp chọn hành động tối ưu ở mỗi bước xét nghiệm

Pseudocode tính chi phí kỳ vọng tối ưu: Sử dụng 'đệ quy' để duyệt theo cây quyết định:

```
1  HÀM EXPECTED_COST(s, k):  
2      NẾU k đạt giới hạn test:  
3          p = BAYES_POSTERIOR(s, k)  
4          cost_A = điều trị ngay  
5          cost_B = bỏ qua  
6          TRẢ VỀ chi phí nhỏ nhất giữa A và B  
7      p = BAYES_POSTERIOR(s, k)  
8      cost_A = điều trị ngay  
9      cost_B = bỏ qua  
10     # Xét thêm 1 test  
11     p_pos = xác suất test dương kế tiếp  
12     p_neg = 1 - p_pos  
13     cost_pos = EXPECTED_COST(s + 1, k + 1)  
14     cost_neg = EXPECTED_COST(s, k + 1)  
15     cost_test_thêm = chi phí test + kỳ vọng tương lai  
16     TRẢ VỀ chi phí nhỏ nhất giữa A, B và test thêm
```

Giải thích

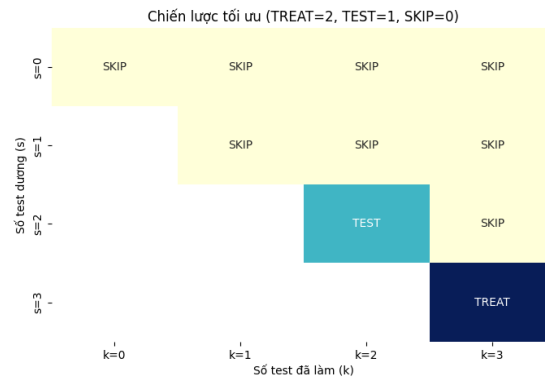
- Hàm `expected_cost(s, k, max_tests=3)`
 - s số test dương tính đã thu được
 - k : số test đã làm
 - max_tests : giới hạn số test tối đa (ở đây là 3)
 - Mục tiêu là trả về chi phí thấp nhất có thể nếu đang ở trạng thái (s, k)
- Khi đã làm đủ test: *if* $k == max_tests$ thì không thể test thêm nữa mà phải ra quyết định lựa chọn giữa:
 - A: điều trị
 - B: không điều trị
- Trường hợp chọn điều trị ngay sau 1 test:

- Điều trị nhầm thì chi phí bệnh viện phải trả: $cost_A = (1 - p) * 500 + k * 50$
- Không điều trị người bệnh thì chi phí bệnh viện phải trả: $cost_B = p * 10000 + k * 50$
- Trường hợp làm thêm 1 test. Vì chưa test đủ, chọn hành động: TEST thêm lần nữa
 - Xác suất test tiếp theo dương hoặc âm
 - * Dương tính $p_next_pos = p * 0.99 + (1 - p) * 0.10$
 - * Âm tính $p_next_neg = 1 - p_next_pos$
 - Chi phí kỳ vọng tương ứng cho hai khả năng
 - * Nếu kết quả dương tính $cost_next_pos = expected_cost(s + 1, k + 1)$
 - * Nếu âm tính: $cost_next_neg = expected_cost(s, k + 1)$
 - Tổng chi phí kỳ vọng khi làm test thêm:
$$cost_C = k * 50 + 50 + (p_next_pos * cost_next_pos + p_next_neg * cost_next_neg)$$
- So sánh và chọn hành động tối ưu (chi phí thấp nhất)

$returnmin(cost_A, cost_B, cost_C)$

Code Python cho bước tính chi phí kỳ vọng (cây quyết định)

```
1 from functools import lru_cache
2 @lru_cache(maxsize=None)
3 def expected_cost(s, k, max_tests=3):
4     if k == max_tests:
5         # Đã test đủ, chọn giữa A và B
6         p = bayes_posterior(s, k)
7         cost_A = (1 - p) * 500 + k * 50
8         cost_B = p * 10000 + k * 50
9         return min(cost_A, cost_B)
10    # Nếu đi điều trị ngay
11    p = bayes_posterior(s, k)
12    cost_A = (1 - p) * 500 + k * 50
```



Hình 28: Mô phỏng, trực quan hóa cây quyết định

```

13 cost_B = p * 10000 + k * 50
14 # Nếu test thêm 1 lần
15 # Kỳ vọng theo kết quả test tiếp theo (dương hoặc âm)
16 p_next_pos = p * 0.99 + (1 - p) * 0.10
17 p_next_neg = 1 - p_next_pos
18 cost_next_pos = expected_cost(s + 1, k + 1)
19 cost_next_neg = expected_cost(s, k + 1)
20 cost_C = k * 50 + 50 + (p_next_pos * cost_next_pos + p_next_neg * cost_next_neg)
21 return min(cost_A, cost_B, cost_C)

```

Mô phỏng, trực quan hóa cây quyết định để tìm chiến lược điều trị tối ưu

Câu 4: Sử dụng mô phỏng Monte Carlo để kiểm nghiệm chiến lược trên với 1 triệu bệnh nhân ngẫu nhiên để tính chi phí trung bình mà bệnh viện phải chi trả.

Ý tưởng

- Sinh $H \sim \text{Bernoulli}(P_{H1})$
- Mỗi bước test sinh ngẫu nhiên dựa trên H
- Áp dụng phương thức `expected_cost()` để ra quyết định sau mỗi test
- Tính tổng chi phí thực tế trên 1 triệu bệnh nhân

Pseudocode cho mô phỏng Monte Carlo:

```
1 INPUT: strategy_matrix[s][k]
2 CONSTANTS: COST_TEST, COST_TREAT_HEALTHY, COST_SKIP_SICK, P_H1, TPR, FPR,
3 NUM_PATIENTS
4 FUNCTION simulate_patient():
5     is_sick ← random() < P_H1
6     s, k, cost ← 0, 0, 0
7     LOOP:
8         action ← strategy_matrix[s][k]
9         IF action == "TREAT":
10             cost += COST_TREAT_HEALTHY if not is_sick
11             RETURN cost + k × COST_TEST
12         IF action == "SKIP":
13             cost += COST_SKIP_SICK if is_sick
14             RETURN cost + k × COST_TEST
15         IF action == "TEST":
16             test_result ← random() < (TPR if is_sick else FPR)
17             k ← k + 1
18             s ← s + 1 if test_result
19             cost += COST_TEST
20 FUNCTION main():
21     total_cost ← 0
22     REPEAT NUM_PATIENTS TIMES:
23         total_cost += simulate_patient()
24     PRINT "Chi phí trung bình mỗi bệnh nhân mà bệnh viện phải chi trả:",
25     total_cost / NUM_PATIENTS, "$"
```

Code Python chạy demo cho mô phỏng Monte Carlo

```
1 import numpy as np
2 # Tham số
3 NUM_PATIENTS = 1_000_000
4 np.random.seed(42) # để kết quả ổn định
5 # Lấy ma trận chiến lược đã tính ở bước trước: strategy_matrix[s, k]
```

```
6  # Đảm bảo bạn đã chạy đoạn mã trước đó để có biến `strategy_matrix`
7  # Hàm mô phỏng chi phí cho một bệnh nhân
8  def simulate_patient(strategy_matrix, max_tests=3):
9      is_sick = np.random.rand() < P_H1
10     k = 0 # số lần test đã làm
11     s = 0 # số test ra dương tính
12     total_cost = 0
13     while True:
14         action = strategy_matrix[s, k]
15         if action == 'TREAT':
16             total_cost += COST_TREAT_HEALTHY if not is_sick else 0
17             total_cost += k * COST_TEST
18             return total_cost
19         elif action == 'SKIP':
20             total_cost += COST_SKIP_SICK if is_sick else 0
21             total_cost += k * COST_TEST
22             return total_cost
23         elif action == 'TEST':
24             test_result = None
25             if is_sick:
26                 test_result = np.random.rand() < TPR
27             else:
28                 test_result = np.random.rand() < FPR
29             k += 1
30             if test_result:
31                 s += 1
32             total_cost += COST_TEST
33         else:
34             raise ValueError(f"Unknown action: {action}")
35 # Mô phỏng nhiều bệnh nhân
36 total_cost = 0
37 for _ in range(NUM_PATIENTS):
```

```
38     total_cost += simulate_patient(strategy_matrix)
39 average_cost = total_cost / NUM_PATIENTS
40 print(f"Chi phí trung bình cho mỗi bệnh nhân mà bệnh viện phải chi trả:
41 {average_cost:.2f}$")
```

Kết quả xuất ra màn hình là:

Chi phí trung bình cho mỗi bệnh nhân mà bệnh viện phải chi trả: 15.07\$

9 Kết luận

Tiểu luận đã trình bày ngắn gọn các kiến thức nền tảng của xác suất thống kê như xác suất, kỳ vọng, phương sai, độ lệch chuẩn, ... và vai trò ứng dụng của xác suất trong học máy nói riêng và trong ngành Khoa học máy tính nói chung. Chẳng hạn, trong các ví dụ và bài tập nhóm đã đề cập đến vấn đề bất định trong Machine Learning. Bất định có 2 dạng là:

1. Bất định aleatoric: Là sự bất định vốn có trong dữ liệu, do tính ngẫu nhiên không thể loại bỏ, ví dụ như kết quả của một lần tung đồng xu.
2. Bất định epistemic: Là sự bất định do thiếu hiểu biết hoặc dữ liệu không đầy đủ, có thể giảm bớt bằng cách thu thập thêm dữ liệu.

Để giảm sự bất định epistemic trong machine learning thì việc thu thập thêm dữ liệu giúp giảm bất định là cần thiết, nhưng tốc độ giảm thường chậm, theo tỉ lệ giảm là $\frac{1}{\sqrt{n}}$, với n là số lượng mẫu. Tức là, nếu tăng gấp đôi dữ liệu thì chỉ giảm bất định một cách tương đối nhỏ. Thông qua thực nghiệm trong phần bài tập 1, 2 nhóm rút ra kết luận rằng sự bất định sẽ giảm ở tỷ lệ nhất định khi dataset tăng lên.

Tiểu luận cũng trình bày định lý Bayes như là một công cụ quan trọng trong thống kê, giúp cập nhật niềm tin dựa trên dữ liệu mới. Xác suất tiên nghiệm và xác suất có điều kiện đóng vai trò quan trọng và hữu ích trong các ứng dụng như chẩn đoán y khoa. Để minh họa cho điều này, nhóm đã chạy thực nghiệm được một số bài toán thực tế như bài toán ra quyết định hành động tiếp theo dựa trên định lý Bayes sau khi thực hiện k lần xét nghiệm HIV.

Cuối cùng, tiểu luận đã ứng dụng của kỳ vọng và xác suất vào trong các bài toán thực tế để xây dựng mô hình hóa xác suất, như mô hình thực nghiệm danh mục đầu tư chứng khoán tối ưu trong bài tập 8. Từ các bài toán thực nghiệm này đã nêu bật tầm quan trọng của kỳ vọng và phương sai trong việc hiểu và mô hình hóa phân phối xác suất, cung cấp cái nhìn sâu sắc về hành vi của các biến ngẫu nhiên.

Tài liệu

- [1] https://d2l.ai/chapter_preliminaries/probability.html
- [2] Probability and Statistics for Computer Science , David Forsyth, (2018).
- [3] Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., ... et al. (2022). Flamingo: a visual language model for few-shot learning. ArXiv:2204.14198.
- [4] Alsallakh, B., Kokhlikyan, N., Miglani, V., Yuan, J., & Reblitz-Richardson, O. (2020). Mind the PAD – CNNs can develop blind spots. ArXiv:2010.02178.
- [5] Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., ... et al. (2023). PaLM 2 Technical Report. ArXiv:2305.10403.
- [6] Nhập Môn Hiện Đại Xác Suất Thống Kê, GS. Nguyễn Tiến Dũng và GS. Đỗ Đức Thái