BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**



**TIỂU LUẬN**

**CƠ SỞ TOÁN CHO HỌC MÁY**

**Tên tiểu luận: Chương 8 – Ước lượng**

.....

Giảng viên HD: **TS. Nguyễn Văn Hậu**

Học viên thực hiện: **Đỗ Quốc Hưng**

Lớp: **H01222**

*Hưng Yên, 6/2023*

Mã lệnh cho chương này ở file estimation.py. Để biết cách tải về và làm việc với mã lệnh,

**8.1  Trò chơi ước lượng**

Ta hãy cùng tham gia một trò chơi. Tôi sẽ nghĩ trong đầu một dạng phân bố, còn bạn phải đoán xem đó là phân bố gì. Tôi cho bạn hai gợi ý; đó là một phân bố chuẩn, và sau đây là một mẫu ngẫu nhiên được rút từ nó:

[-0.441, 1.774, -0.101, -1.138, 2.975, -2.138]

Bạn thử đoán xem tham số trị trung bình, μ, của phân bố này bằng bao nhiêu?

Một cách lựa chọn là dùng trị trung bình mẫu để ước tính μ. Ở ví dụ này, \bar{x} bằng 0.155, vì vậy sẽ có lý khi đoán µ = 0.155. Quá trình này được gọi là **ước lượng**, và đặc trưng thống kê mà ta xét đến (trị trung bình mẫu) được gọi là **tham số ước lượng**.

Việc dùng trị trung bình mẫu để ước lượng μ là hiển nhiên đến nỗi thật khó tưởng tượng nổi một cách làm khác. Nhưng giả dụ rằng chúng ta thay đổi trò chơi bằng cách đưa vào các điểm biệt lập.

*Tôi đang nghĩ về một phân bố.* Đó là một phân bố chuẩn, và sau đây là một mẫu thu thập được bởi người điều tra viên không đáng tin cậy, đôi khi lại mắc lỗi đặt nhầm chỗ dấu phẩy của phần thập phân.

[-0.441, 1.774, -0.101, -1.138, 2.975, -213.8]

Bây giờ ước đoán của bạn về μ bằng bao nhiêu? Nếu bạn dùng trị trung bình mẫu thì ước đoán sẽ bằng –35,12. Liệu đó có phải là lựa chọn tốt nhất không? Còn có những cách nào khác?

Có một cách khác là nhận biết và loại bỏ các điểm biệt lập, rồi tính trị trung bình mẫu của những số còn lại. Một cách nữa là dùng số trung vị để ước lượng.

Ước lượng nào là tốt nhất còn phụ thuộc vào từng trường hợp cụ thể (chẳng hạn, liệu có điểm biệt lập hay không) và vào mục tiêu của ước lượng. Bạn nhằm mục đích giảm thiểu sai số, hay nâng cao tối đa khả năng tìm được đáp số đúng?

Nếu như không có điểm biệt lập nào thì trị trung bình mẫu sẽ làm giảm thiểu **sai số quân phương** (mean squared error, MSE). Điều đó nghĩa là: nếu bạn chơi trò đoán số này nhiều lần, và mỗi lần tính sai số \bar{x} − µ, thì trị trung bình mẫu sẽ làm giảm thiểu đại lượng

MSE = \frac{1}{m} \sum (\bar{x} - \mu)^2 

trong đó *m* là số lần đoán (khác với *n*, là kích thước mẫu dùng để tính \bar{x}.

Sau đây là một hàm để mô phỏng trò chơi đoán số và tính sai số căn quân phương (RMSE), vốn là căn bậc hai của MSE:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | def Estimate1(n=7, m=1000):      mu = 0      sigma = 1        means = []      medians = []      for \_ in range(m):          xs = [random.gauss(mu, sigma) for i in range(n)]          xbar = np.mean(xs)          median = np.median(xs)          means.append(xbar)          medians.append(median)        print('rmse xbar', RMSE(means, mu))      print('rmse median', RMSE(medians, mu)) |

Ở đây, n lại là kích cỡ mẫu còn m là số lần ta đoán. means là danh sách các ước lượng dụa trên \bar{x}. medians là danh sách các trung vị.

Sau đây là hàm để tính RMSE:

|  |  |
| --- | --- |
| 1  2  3  4 | def RMSE(estimates, actual):      e2 = [(estimate-actual)\*\*2 for estimate in estimates]      mse = np.mean(e2)      return math.sqrt(mse) |

estimates là một danh sách các ước lượng; actual là giá trị thực cần được ước lượng. Dĩ nhiên, trên thực tế, ta không biết actual; nếu biết rồi thì ta đã chẳng cần ước lượng làm gì. Mục đích của thí nghiệm này là để so sánh hiệu năng của hai ước lượng.

Khi tôi chạy mã lệnh này, sai số RMSE của trị trung bình mẫu bằng 0.41; điều đó có nghĩa là nếu ta dùng \bar{x} để ước lượng trị trung bình của phân bố này, dựa vào một mẫu có kích cỡ *n*= 7, thì ta sẽ phải lường trước rằng kết quả bị lệch đi 0.41 (xét ở mức độ trung bình). Khi sử dụng trung vị để ước lượng trung bình sẽ cho ra RMSE 0.53, điều này khẳng định rằng \bar{x} cho ta sai số RMSE thấp hơn, ít nhất là ở ví dụ này.

Giảm thiểu MSE là một điều tốt, nhưng nó không phải luôn là chiến lược tối ưu. Chẳng hạn, giả sử ta cần ước tính dạng phân bố tốc độ gió tại một công trường. Nếu ta đoán gió quá mạnh thì có thể sẽ phải xây dựng với chi phí quá đắt. Nhưng nếu ta đoán gió quá yếu thì công trình có thể sẽ sụp đổ. Bởi vì chi phí xây dựng phụ thuộc vào sai số theo một dạng bất đối xứng nên việc giảm thiểu MSE không phải là cách tốt nhất.

Lấy một ví dụ khác, chẳng hạn khi tôi gieo xúc sắc 3 lần và đề nghị bạn ước tính tổng số chấm. Nếu đoán đúng, bạn sẽ nhận một phần thưởng; còn nếu không thì sẽ chẳng được gì. Trong trường hợp này giá trị làm giảm thiểu MSE là 10,5; nhưng nó sẽ là một dự đoán hoàn toàn sai. Trong trò chơi này, bạn muốn có một ước đoán có khả năng đúng nhiều nhất, tức là một **ước lượng hợp lý cực đại** (Maximum Likelihood Estimator, MLE). Nếu bạn chọn 10 hoặc 11, bạn sẽ có khả năng đoán trúng bằng 1 phần 8, và đó là khả năng tốt nhất bạn có thể đạt được.

**8.2  Dự đoán phương sai**

*Tôi đang nghĩ về một phân bố.* Đó là phân bố chuẩn, và sau đây là một mẫu (quen thuộc):

[-0.441, 1.774, -0.101, -1.138, 2.975, -2.138]

Theo bạn thì phương sai, σ2, của phân bố này bằng bao nhiêu? Một lần nữa, lựa chọn hiển nhiên nhất là lấy phương sai mẫu, *S*2, làm giá trị ước lượng.

S^2 = \frac{1}{n} \sum (x_i - \bar{x})^2 

Với những mẫu lớn, *S*2 là một ước lượng thích hợp, nhưng với mẫu bé thì giá trị nó thiên nhỏ. Vì đặc điểm không may này mà nó được gọi là một ước lượng **chệch**. Một ước lượng được gọi là **không chệch** nếu như sai số tổng cộng (hoặc trung bình) được trông đợi sau nhiều lượt chơi bằng 0.

Thật may là còn có một đại lượng thống kê đơn giản khác làm ước lượng không chệch cho σ2:

S_{n-1}^2 = \frac{1}{n - 1} \sum (x_i - \bar{x})^2 

Để hiểu được tại sao *S*2 là ước lượng chệch, và chứng minh rằng *Sn*-12 không chệch

Vấn đề nan giải nhất đối với ước lượng này là cách dùng tên gọi và kí hiệu của nó không thống nhất. Cái tên “phương sai mẫu” có thể được dùng cho cả *S*2 hay *Sn*−12, còn kí hiệu *S*2 được dùng để chỉ cả hai đại lượng này.

Sau đây là một hàm để mô phỏng trò chơi ước lượng và thử kiểm tra hiệu năng của *S*2 và *Sn*−12:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | def Estimate2(n=7, m=1000):      mu = 0      sigma = 1        estimates1 = []      estimates2 = []      for \_ in range(m):          xs = [random.gauss(mu, sigma) for i in range(n)]          biased = np.var(xs)          unbiased = np.var(xs, ddof=1)          estimates1.append(biased)          estimates2.append(unbiased)        print('mean error biased', MeanError(estimates1, sigma\*\*2))      print('mean error unbiased', MeanError(estimates2, sigma\*\*2)) |

Ở đây, n lại là kích cỡ mẫu còn m là số lần ta đoán. np.var mặc định sẽ tính ra *S*2; còn nếu muốn tính *Sn*−12 bạn phải cung cấp đối số ddof=1, tên của nó viết tắt từ “delta degrees of freedom (bậc tự do)”. Tôi sẽ không giải thích khái niệm này, MeanError tính ra hiệu số giữa ước lượng và giá trị thực:

|  |  |
| --- | --- |
| 1  2  3 | def MeanError(estimates, actual):      errors = [estimate-actual for estimate in estimates]      return np.mean(errors) |

Khi tôi chạy đoạn mã này, sai số trung bình của *S*2 là -0.13. Như đã dự kiến, ước lượng chệch này có xu hướng quá thấp. Với *Sn*−12, sai số trung bình là 0.014, nhỏ hơn khoảng 10 lần. Khi m tăng lên, ta liệu rằng sai số trung bình của *Sn*−12 sẽ tiến tới 0.

Những thuộc tính như sai số MSE và độ chệch là những kì vọng dài hạn dựa theo nhiều lần lặp trong trò chơi đoán số. Bằng cách chạy những mô phỏng như đã thực hiện trong chương này, ta có thể so sánh các ước lượng và kiểm tra xem chúng có những thuộc tính ta mong muốn không.

Nhưng khi bạn áp dụng một ước lượng vào số liệu thực, bạn chỉ thu được một ước lượng thôi. Sẽ không nghĩa lý gì khi nói rằng ước lượng này không chệch; vì sự không chệch là một đặc tính của đại lượng ước lượng, chứ không phải của trị số ước lượng.

Sau khi bạn chọn (đại lượng) ước lượng với các thuộc tính phù hợp, và dùng nó để phát sinh ra một trị số ước lượng, thì bước tiếp theo là cần đặc trưng mức độ bất định của ước lượng, đây là chủ đề của mục tiếp theo.

**8.3  Phân bố lấy mẫu**

Giả sử bạn là nhà khoa học nghiên cứu loài khỉ đột trong một khu bảo tồn động vật hoang dã. Bạn muốn biết cân nặng trung bình của khỉ đột cái trưởng thành trong khu bảo tồn này. Để cân chúng, bạn cần đánh thuốc mê; đây là việc làm nguy hiểm, chi phí đắt, và có thể gây hại tới con khỉ đột. Nhưng nếu thông tin này rất quan trọng thì có thể chấp nhận được việc cân một mẫu 9 con khỉ đột. Ta hãy coi như là đã biết rõ tổng thể khỉ đột trong khu bảo tồn, để ta có thể chọn được một mẫu đại diện cho các khỉ cái trưởng thành. Ta có thể dùng trị trung bình mẫu, \bar{x}, để ước tính trị trung bình tổng thể chưa biết, µ.

Sau khi cân xong 9 con khỉ cái, bạn có thể thu được \bar{x} = 90 kg và độ lệch chuẩn mẫu *S*= 7.5 kg. Trị trung bình mẫu là ước lượng không chệch của µ, và về lâu dài nó làm tối thiểu hóa sai số MSE. Vì vậy, nếu bạn phải báo cáo một con số ước lượng để tóm tắt kết quả, hãy báo cáo 90 kg.

Nhưng bạn nên tin cậy kết quả trên đến mức nào? Nếu bạn chỉ cân *n* = 9 con khỉ trong số một tổng thể lớn hơn nhiều, bạn có thể không may chọn phải 9 con khỉ nặng nhất (hoặc 9 con nhẹ nhất) hoàn toàn do ngẫu nhiên. Sự biến động giá trị ước lượng do chọn ngẫu nhiên được gọi là **sai số lấy mẫu**.

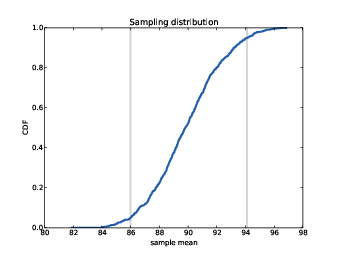
Để lượng hóa sai số lấy mẫu, ta có thể mô phỏng quá trình lấy mẫu với các trị số giả tưởng của µ và σ, rồi xem \bar{x} biến động bao nhiêu.

Vì không biết các giá trị thực sự của µ và σ trong tổng thể, ta sẽ dùng các trị số ước lượng \bar{x} và *S*. Do vậy câu hỏi cần trả lời là: “Nếu các giá trị thực sự của µ và σ là 90 kg và 7.5 kg, mà ta đã thực hiện cùng một phép thử nhiều lần, thì trị trung bình ước lượng, \bar{x}, sẽ biến động bao nhiêu?”

Hàm sau đây giúp ta trả lời câu hỏi đó:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | def SimulateSample(mu=90, sigma=7.5, n=9, m=1000):      means = []      for j in range(m):          xs = np.random.normal(mu, sigma, n)          xbar = np.mean(xs)          means.append(xbar)        cdf = thinkstats2.Cdf(means)      ci = cdf.Percentile(5), cdf.Percentile(95)      stderr = RMSE(means, mu) |

mu và sigma là những giá trị *giả định* của các tham số thống kê. n là kích cỡ mẫu, số con khỉ đột đem cân. m là số lần ta thực hiện phép thử.



|  |
| --- |
| Hình 8.1: Phân bố lấy mẫu của \bar{x}, với khoảng tin cậy. |

Qua mỗi lần lặp, ta chọn ra n giá trị từ một phân bố chuẩn với các tham số đã cho, rồi tính trị trung bình mẫu, xbar. Ta thực hiện 1000 mô phỏng rồi tính phân bố, cdf, của trị số ước lượng này. Kết quả được vẽ trên Hình [8.1](https://greenteapress.com/thinkstats2/html/thinkstats2009.html#estimation1). Đường phân bố này được gọi là **phân bố lấy mẫu** của ước lượng. Nó cho thấy trị số ước lượng sẽ biến động bao nhiêu nếu ta lặp đi lặp lại phép thử.

Trị trung bình của phân bố lấy mẫu khá sát với giá trị giả định µ, điều này nghĩa rằng phép thử cho ra kết quả đúng, xét trung bình. Sau 1000 phép thử, kết quả thấp nhất là 82 kg, và cao nhất là 98 kg. Khoảng giá trị này cho thấy trị số ước tính có thể lệch tới 8 kg.

Thường có hai cách để tóm tắt dạng phân bố lấy mẫu:

* **Sai số tiêu chuẩn** (standard error, SE) là khoảng độ chệch của trị số ước lượng, mà ta trông đợi (nếu xét trung bình). Với mỗi phép thử được mô phỏng, ta tính sai số, \bar{x} − µ, rồi tính sai số căn quân phương (RMSE). Ở ví dụ này, sai số khoảng 2.5 kg.
* Một **khoảng tin cậy** (confidence interval, CI) là khoảng giá trị bao lấy một phần cho trước của phân bố mẫu. Chẳng hạn, khoảng tin cậy 90% là khoảng từ số phần trăm thứ 5 đến số phần trăm thứ 95. Ở ví dụ này, 90% CI là (86, 94) kg.

Sai số tiêu chuẩn và khoảng tin cậy là nguồn gốc gây nên những nhầm lẫn:

* Người ta thường nhầm giữa sai số tiêu chuẩn và độ lệch chuẩn. Hãy nhớ rằng độ lệch chuẩn mô tả sự biến động trong một đại lượng được đo đạc; trong ví dụ này, độ lệch chuẩn của cân nặng con khỉ là 7.5 kg. Sai số tiêu chuẩn thì mô tả sự biến động trong một giá trị ước lượng. Ở ví dụ này, sai số tiêu chuẩn của trị trung bình dựa trên 9 kết quả đo, là 2.5 kg. Một cách để nhớ khác biệt này, đó là khi kích cỡ mẫu tăng, sai số tiêu chuẩn thì nhỏ đi, còn độ lệch chuẩn thì không.
* Người ta thường nghĩ rằng có 90% xác suất mà giá trị thật của tham số, µ, rơi vào trong khoảng tin cậy 90%. Đáng buồn là không phải vậy. Nếu bạn muốn tuyên bố điều đó, bạn phải dùng các phương pháp Bayes (xem cuốn sách tôi viết, *Think Bayes*). Phân bố mẫu trả lời một điều khác: nó cho ta hình dung xem một giá trị ước lượng tin cậy đến mức nào, bằng cách cho bạn biết rằng ước lượng đó sẽ thay đổi bao nhiêu nếu bạn thực hiện lại phép thử.

Điều quan trọng cần nhớ là khoảng tin cậy và sai số tiêu chuẩn chỉ định lượng hóa sai số lấy mẫu, tức là sai số do đo đạc trên một bộ phận tổng thể, mà thôi. Phân bố lấy mẫu không xét đến những nguồn sai số khác mà đáng kể là sự thiên lệch lấy mẫu và sai số đo đạc, vốn là chủ đề của mục kế tiếp.

**8.4  Sự thiên lệch lấy mẫu**

Giả dụ rằng thay vì cân nặng của khỉ trong khu bảo tồn, bạn cần biết trọng lượng trung bình của phụ nữ trong thành phố bạn đang sống. Chắc bạn sẽ không được phép chọn một mẫu đại diện các phụ nữ và đem cân họ rồi.

Một phương án khác đơn giản là “lấy mẫu qua điện thoại”; nghĩa là bạn chọn các số điện ngẫu nhiên từ cuốn danh bạ, bấm máy gọi và đề nghị nói chuyện với người phụ nữ trưởng thành, rồi hỏi cân nặng của cô ta.

Lấy mẫu qua điện thoại dĩ nhiên là có những hạn chế. Chẳng hạn, mẫu bị giới hạn bởi những người costrong danh sách danh bạ, bởi vậy sẽ loại trừ những người không có số điện thoại được liệt kê (có thể là người nghèo hơn mức trung bình) và những người không lên danh sách (có thể giàu hơn). Ngoài ra, nếu bạn gọi số điện thoại nhà vào thời gian trong ngày, thì bạn ít khả năng lấy mẫu những người có công việc làm. Và nếu bạn chỉ lấy mẫu những người nghe máy, thì bạn ít khả năng lấy mẫu được những người phải dùng chung đường dây.

Nếu những yếu tố như thu nhập, công việc, và cỡ hộ gia đình lại có ảnh hưởng đến số cân nặng—mà điều đó là hợp lý—thì kết quả cuộc khảo sát sẽ bị ảnh hưởng bằng cách này hay cách khác. Vấn đề này được gọi là **sự thiên lệch lấy mẫu** vì nó là một thuộc tính của quá trình lấy mẫu.

Quá trình lấy mẫu còn bị ảnh hưởng bởi việc chọn lại, vốn là một kiểu thiên lệch lấy mẫu. Một số người sẽ từ chối trả lời câu hỏi; và nếu xu hướng từ chối này lại gắn với cân nặng, thì điều đó sẽ ảnh hưởng đến kết quả.

Sau cùng, nếu bạn hỏi người ta cân nặng bao nhiêu, thay vì đem cân đến đo, thì kết quả có thể không chính xác. Ngay cả những người điều tra nhiệt tình cũng có thể làm tròn con số lên hoặc xuống nếu họ không thấy thoải mái với cân nặng thật. Và không phải người nào được tra cũng nhiệt tình hợp tác. Những yếu tố như vậy là các ví dụ **sai số đo đạc**.

Khi bạn báo cáo một kết quả ước lượng, cần thiết phải nêu sai số tiêu chuẩn hoặc khoảng tin cậy, hoặc cả hai, để định lượng hóa sai số lấy mẫu. Nhưng điều quan trọng cũng cần nhớ rằng sai số lấy mẫu chỉ là một nguồn gây sai số mà thôi, và nó thường không phải là nguồn chính gây sai số.

**8.5  Phân bố lũy thừa**

Hãy chơi trò đoán số thêm một lần nữa. *Tôi đang nghĩ về một dạng phân bố.* Đó là phân bố lũy thừa, và sau đây là một mẫu:

[5.384, 4.493, 19.198, 2.790, 6.122, 12.844]

Bạn thử đoán xem tham số, λ, của phân bố này bằng bao nhiêu?

Nói chung, trị trung bình của một phân bố lũy thừa bằng 1/λ, vì vậy tính ngược lại, bạn có thể lấy

|  |
| --- |
| *L* = 1 / \bar{x} |

*L* là một ước lượng của λ. Và không chỉ với một ước lượng bất kì, nó cũng là ước lượng hợp lý cực đại Vì vậy nếu bạn muốn có cơ hội nhiều nhất để đoán trúng λ, thì *L* chính là cách làm.

Nhưng ta biết rằng \bar{x} không còn vững nếu xuất hiện các điểm biệt lập, vì vậy ta đoán rằng *L* cũng gặp vấn đề tương tự.

Ta có thể chọn được một cách tính khác dựa trên số trung vị mẫu. Trung vị của một phân bố lũy thừa bằng log(2) / λ, vì vậy lại tính ngược, ta có thể định nghĩa ước lượng sau

|  |
| --- |
| *Lm* = ln(2) / *m* |

trong đó *m* là số trung vị mẫu.

Để kiểm tra hiệu năng của các ước lượng này, ta có thể mô phỏng quá trình lấy mẫu:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | def Estimate3(n=7, m=1000):      lam = 2        means = []      medians = []      for \_ in range(m):          xs = np.random.exponential(1.0/lam, n)          L = 1 / np.mean(xs)          Lm = math.log(2) / thinkstats2.Median(xs)          means.append(L)          medians.append(Lm)        print('rmse L', RMSE(means, lam))      print('rmse Lm', RMSE(medians, lam))      print('mean error L', MeanError(means, lam))      print('mean error Lm', MeanError(medians, lam)) |

Khi tôi chạy thử nghiệm với λ=2, sai số RMSE của *L* bằng 1.1. Với ước lượng *Lm* dựa trên phương sai, RMSE bằng 1.8. Từ thử nghiệm này, ta chưa thể nói rằng liệu *L* có làm giảm thiểu sai số MSE không, nhưng ít nhất thì có vẻ nó đã tốt hơn *Lm*.

Đáng buồn là, dường như cả hai ước lượng này đều chệch. Với *L* sai số trung bình là 0.33; còn với *Lm* thì là 0.45. Không ước lượng nào hội tụ về 0 khi m tăng lên.

Hóa ra, \bar{x} là một ước lượng không chệch của trị trung bình phân bố, 1 / λ; nhưng *L* không phải là một ước lượng không chệch của λ.

**8.6  Bài tập**

Với các bài tập sau, có thể bạn muốn khởi đầu với phiên bản file estimation.py. Lời giải bài tập này có ở file chap08soln.py

**Bài tập 1**

*Trong chương này ta đã dùng*\bar{x}*và trung vị để ước lượng*µ*, và phát hiện thấy*\bar{x}*cho sai số MSE thấp hơn. Ta cũng đã dùng S*2*và Sn*−12*để ước lượng*σ*, và thấy rằng S*2*chệch còn Sn*−12*không chệch.*

*Hãy chạy những phép thử tương tự để xem liệu*\bar{x}*và trung vị có phải là những ước lượng chệch của*µ*hay không. Cũng kiểm tra xem liệu S*2*hay Sn*−12*có cho ra sai số MSE thấp hơn không.*

**Bài tập 2**

*Giả sử bạn rút ra một mẫu với kích cỡ n*=10*từ một phân bố lũy thừa với*λ=2*. Hãy mô phỏng phép thử này 1000 lần và vẽ biểu đồ phân bố mẫu của ước lượng L. Tính sai số chuẩn của ước lượng này cùng khoảng tin cậy 90%.*

*Hãy lặp lại phép thử với vài giá trị khác nhau của n rồi vẽ biểu đồ sai số chuẩn theo n.*

**Bài tập 3**

*Trong những môn thể thao như khúc côn cầu và bóng đá, khoảng thời gian giữa hai bàn thắng liên tiếp thì gần như theo phân bố lũy thừa. Bởi vậy, bạn có thể ước lượng năng suất ghi bàn của một đội qua việc quan sát số bàn thắng họ ghi được trong một trận đấu. Cách ước lượng này hơi khác với lấy mẫu thời gian giữa các bàn thắng, vậy ta hãy xem cách này hoạt động ra sao.*

*Hãy viết một hàm nhận vào năng suất ghi bàn, lam, tính theo số bàn thắng đội ghi được trong một trận, rồi mô phỏng một trận đấu bằng cách phát sinh thời gian giữa các bàn thắng cho đến khi tổng thời gian vượt quá 1 trận đấu, và trả lại số bàn thắng ghi được.*

*Hãy viết một hàm khác để mô phỏng nhiều trận đấu, lưu lại những ước lượng cho lam, rồi tính sai số trung bình và sai số căn quân phương RMSE của chúng.*

*Có phải cách ước lượng này bị chệch không? Hãy vẽ biểu đồ lấy mẫu các ước lượng và khoảng tin cậy 90%. Sai số tiêu chuẩn bằng bao nhiêu? Điều gì đã xảy ra với sai số lấy mẫu với những giá trị càng tăng của lam?*

**8.7  Glossary**

* **ước lượng (estimation)**: Quá trình suy luận các tham số của một phân bố từ một mẫu.
* **ước lượng (estimator)**: Đặc trưng thống kê được dùng để ước lượng cho một tham số.
* **sai số quân phương (mean squared error, MSE)**: Một độ đo cho sai số của ước lượng.
* **sai số căn quân phương (root mean squared error, RMSE)**: Căn bậc hai của MSE, một cách biểu diễn ý nghĩa hơn cho độ lớn sai số điển hình.
* **ước lượng hợp lý cực đại (maximum likelihood estimator, MLE)**: Ước lượng nhằm tính ra được ước lượng điểm với khả năng đúng đắn là lớn nhất.
* **độ chệch của ước lượng (bias of an estimator)**: Xu hướng của ước lượng nằm trên hoặc dưới giá trị tham số thực tế, sau khi lấy trung bình qua các phép thử được lặp đi lặp lại.
* **sai số lấy mẫu (sampling error)**: Sai số trong một ước lượng gây bởi kích cỡ mẫu bị hạn chế và sự biến động do ngẫu nhiên.
* **độ thiên lệch lấy mẫu (sampling bias)**: Sai số trong một ước lượng gây bởi một quá trình lấy mẫu mà không đại diện cho tổng thể.
* **sai số đo đạc (measurement error)**: Sai số trong một ước lượng gây bởi sự không chính xác trong khâu thu thập và ghi chép số liệu.
* **phân bố mẫu (sampling distribution)**: Phân bố của một đặc trưng thống kê nếu một phép thử được lặp lại nhiều lần.
* **sai số tiêu chuẩn (standard error)**: Sai số RMSE của một ước lượng, vốn lượng hóa mức độ biến động gây ra riêng bởi sai số lấy mẫu (chứ không xét các loại sai số khác).
* **khoảng tin cậy (confidence interval)**: Một khoảng giá trị biểu diễn cho khoảng kì vọng của một ước lượng nếu phép thử được thực hiện lặp đi lặp lại.