ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN - TIN HỌC

BÁO CÁO CUỐI KỲ MÔN XỬ LÝ ĐA CHIỀU

TOPIC FACTOR ANALYSIS

Giảng viên hướng dẫn: TS. Kha Tuấn Minh
Sinh viên thực hiện: Trần Ngọc Dễ
Mã số sinh viên: 21110057

Tp. Hồ Chí Minh, Tháng 05/2024

Mục lục

1 Giới thiệu **3**

1.1 Tổng quan . . . 3

1.2 Sự khác biệt cơ bản giữa FA và PCA . . . 4

1.3 Ví dụ . . . 5

1.4 Mô hình nhân tố . . . 7

2 Các phương pháp phân tích nhân tố **8**

2.1 Phương pháp thành phần chính . . . 9

2.1.1 Lý thuyết . . . 9

2.1.2 Ví dụ . . . 10

2.2 Phương pháp ước lượng hợp lý cực đại . . . 15

2.2.1 Giả định . . . 15

2.2.2 Ví dụ . . . 16

3 Xoay nhân tố **20**

3.1 Xoay vuông góc . . . 21

3.2 Xoay không vuông góc . . . 24

4 Các loại phân tích nhân tố **25**

4.1 Phân tích nhân tố khám phá (Exploratory Factor Analysis) . . . 25

4.1.1 Mục tiêu . . . 25

4.1.2 Quy trình phân tích nhân tố khám phá . . . 25

4.1.3 Ưu và nhược điểm . . . 26

4.2 Phân tích nhân tố khẳng định (Confirmatory Factor Analysis) . . . 27

4.2.1 Mục tiêu . . . 27

4.2.2 Quy trình phân tích nhân tố khẳng định . . . 27

4.2.3 Ưu và nhược điểm . . . 28

5 Ưu và nhược điểm của phân tích nhân tố **28**

5.1 Ưu điểm . . . 29

5.2 Nhược điểm . . . 29

6 Ứng dụng **30**

6.1 Ứng dụng về phân tích nhân tố khám phá 30

6.1.1 Mục tiêu 30

6.1.2 Mô tả dữ liệu 30

6.1.3 Ý tưởng chính 30

6.1.4 Nhận xét - Đánh giá chung 37

6.1.5 Hướng mở rộng 37

6.2 Ví dụ khác 37

7 Tài liệu tham khảo **44**

1 Giới thiệu

1.1 Tổng quan

Factor analysis là một phương pháp cho việc mô hình hóa các biến quan sát được và cấu trúc hiệp phương sai của chúng dưới dạng một số lượng nhỏ hơn các "yếu tố tìm ẩn" không quan sát được (latent). Những yếu tố này thường được xem là các khái niệm hoặc ý tưởng rộng rãi có thể mô tả một hiện tượng quan sát được. Hoặc nói một cách dễ hiểu hơn là nó dùng để rút gọn một tập gồm nhiều biến quan sát phụ thuộc lẫn nhau thành một tập biến (gọi là các nhân tố) ít hơn nhưng vẫn chứa đựng hầu hết các nội dung thông tin của tập biến ban đầu.

Phương pháp này tương tự như thành phần chính (principal components). Ở một khía cạnh nào đó, phân tích nhân tố là sự nghịch đảo của thành phần chính. Trong phân tích nhân tố, chúng ta mô hình hóa các biến quan sát dưới dạng hàm tuyến tính của “các yếu tố”. Trong thành phần chính, chúng ta tạo ra các biến mới là sự kết hợp tuyến tính của các biến quan sát được. Cả trong PCA và FA, kích thước của dữ liệu được giảm đi. Trong PCA, việc diễn giải các thành phần chính thường không rõ ràng. Đôi khi, một biến cụ thể có thể đóng góp đáng kể vào nhiều thành phần. Lý tưởng, chúng ta muốn mỗi biến đóng góp một cách đáng kể vào chỉ một thành phần. Một kỹ thuật gọi là xoay nhân tố (factor rotation) được sử dụng để đạt được mục tiêu đó. Các lĩnh vực mà phân tích nhân tố tham gia bao gồm sinh lý học, sức khỏe, trí tuệ, xã hội học và đôi khi cả sinh thái học.

Dưới đây là một số trường hợp khi ta nên sử dụng phân tích nhân tố:

1. Giảm chiều dữ liệu: Khi có một tập hợp lớn các biến quan sát, và ta muốn giảm số lượng các biến này xuống mà vẫn giữ được thông tin cơ bản, phân tích nhân tố có thể được sử dụng để tìm ra những nhân tố chính giải thích phần lớn sự biến thiên trong dữ liệu.

2. Khám phá cấu trúc tiềm ẩn: Trong nghiên cứu khoa học xã hội hoặc tâm lý học, phân tích nhân tố được sử dụng để xác định các cấu trúc tiềm ẩn hay các khía cạnh chưa được nhìn nhận trực tiếp thông qua các câu hỏi khảo sát hoặc bài test.

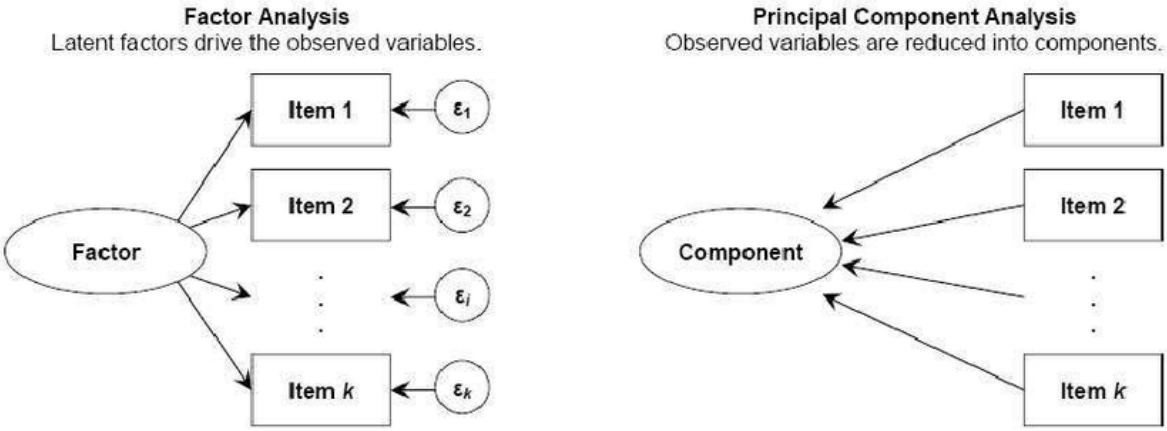
3. Nghiên cứu tính đo lường: Khi phát triển một công cụ đo lường mới, như một bảng câu hỏi, phân tích nhân tố có thể giúp xác định số lượng và bản chất của các yếu tố (nhân tố) mà công cụ đo lường đang cố gắng để đánh giá.

4. Tối ưu hóa các biến cho các mô hình phức tạp hơn: Trong mô hình hóa thống kê hoặc máy học, phân tích nhân tố có thể giúp loại bỏ đa cộng tuyến giữa các biến, làm cho các mô hình dễ dàng quản lý và ổn định hơn.

5. Phân tích tương quan: Nếu các biến quan sát có mức độ tương quan cao với nhau, phân tích nhân tố có thể giúp hiểu rõ hơn về cấu trúc tương quan đó bằng cách nhóm các biến tương quan thành các nhân tố chung.

1.2 Sự khác biệt cơ bản giữa FA và PCA

Tương tự như PCA, phân tích nhân tố (FA) cũng không có biến phụ thuộc.



Hình 1: *Sự khác biệt giữa FA và PCA*

PCA:

- Mục tiêu của PCA là tìm ra các thành phần chính của dữ liệu, các hướng mà dữ liệu biến thiên nhiều nhất. PCA không giả định về sự tương quan giữa các biến, mà chỉ tập trung vào việc tối ưu hóa việc giữ lại phương sai.
- Trong PCA, chúng ta chọn một số thành phần giải thích được sao cho càng nhiều phương sai tổng càng được giải thích càng tốt.
- Trong PCA, mỗi thành phần chính (PC) được biểu diễn dưới dạng tổ hợp tuyến tính của các biến quan sát.
- Kết quả của PCA thường được sử dụng để giảm chiều dữ liệu hoặc để làm nền tảng cho các phân tích và mô hình hóa khác.

FA:

- Mục tiêu của FA là tìm ra các biến tiềm ẩn (nhân tố) mà dẫn đến sự tương quan giữa các biến quan sát. FA giả định rằng mỗi biến được quan sát là một hàm tuyến tính của các nhân tố ẩn cộng với một lỗi đo.
- Trong FA, các yếu tố được lựa chọn chủ yếu để giải thích mối tương quan giữa các biến ban đầu. Lý tưởng nhất là số lượng các yếu tố dự kiến được biết trước.
- Sự chú trọng chính đặt vào việc thu được các yếu tố dễ hiểu mà truyền đạt thông tin cần thiết chứa đựng trong tập hợp ban đầu của biến.
- Trong FA, mỗi biến quan sát (X) được biểu thị dưới dạng kết hợp tuyến tính của các yếu tố.
- Kết quả của FA thường được sử dụng để hiểu cấu trúc tiềm ẩn của dữ liệu và để phát triển các mô hình lý thuyết.

1.3 Ví dụ

Phân tích nhân tố được giải thích tốt nhất trong ngữ cảnh của một ví dụ đơn giản. Sinh viên tham gia một chương trình MBA cụ thể phải học ba khóa bắt buộc về tài chính, tiếp thị và chính sách kinh doanh. Gọi Y_1, Y_2 và Y_3 lần lượt là điểm số của sinh viên trong các khóa học này. Dữ liệu có sẵn bao gồm điểm của năm sinh viên (trong một thang điểm số từ 0 đến 10), như được thể hiện trong Bảng 1.

Bảng 1: *Student grades*

Student	**Grades in**		
	Finance, Y_1	**Marketing,** Y_2	**Policy,** Y_3
1	3	6	5
2	7	3	3
3	10	9	8
4	3	9	7
5	10	6	5

Có ý kiến cho rằng các điểm số này là các hàm của hai nhân tố cơ bản, F_1 và F_2, được mô tả tạm thời và một cách mơ hồ là khả năng định lượng và khả năng nói. Giả định rằng mỗi biến Y có mối quan hệ tuyến tính với nhân tố, như sau:

$$\begin{aligned} Y_1 &= \beta_{10} + \beta_{11}F_1 + \beta_{12}F_2 + e_1 \\ Y_2 &= \beta_{20} + \beta_{21}F_1 + \beta_{22}F_2 + e_2 \\ Y_3 &= \beta_{30} + \beta_{31}F_1 + \beta_{32}F_2 + e_3 \end{aligned} \tag{1}$$

Các sai số e_1, e_2, và e_3 dùng để chỉ ra rằng các mối quan hệ được giả định là không chính xác.

Trong từ vựng đặc biệt của phân tích nhân tố, các tham số β_{ij} được gọi là trọng số (loadings). Ví dụ, β_{12} được gọi là trọng số của biến Y_1 đối với nhân tố F_2.

Trong chương trình MBA này, Finance có tính định lượng cao, trong khi Marketing và Policy có xu hướng định hướng định tính mạnh mẽ. Kỹ năng định lượng sẽ giúp ích cho sinh viên trong lĩnh vực finance chứ không phải trong makerting hoặc policy. Kỹ năng nói sẽ hữu ích trong tiếp thị hoặc chính sách nhưng không hữu ích trong tài chính. Nói cách khác, người ta mong đợi rằng các tải trọng có cấu trúc đại khái như sau:

Biến, Y_i	F_1, β_{i1}	F_2, β_{i2}
Y_1	+	0
Y_2	0	+
Y_3	0	+

Điểm số trong khóa học finance được dự kiến sẽ có mối quan hệ tích cực với khả năng định lượng nhưng không liên quan đến khả năng nói; mặc khác các điểm số trong marketing và polyci được kỳ vọng sẽ có mối quan hệ tích cực với khả năng nói nhưng không liên quan đến khả năng định lượng. Tất nhiên, các giá trị 0 trong bảng trên không được dự kiến sẽ chính xác bằng không. Bằng "0" có nghĩa là gần bằng không và bằng "+" một số dương khác biệt với 0.

Có vẻ như các trọng số có thể được ước lượng và các kỳ vọng được kiểm tra bằng cách hồi quy mỗi Y theo hai nhân tố. Tuy nhiên, cách tiếp cận như vậy không khả thi vì không thể quan sát được các nhân tố, ta cần một phương pháp hoàn toàn mới.

Hãy xem xét quá trình tạo ra các quan sát trên Y_1, Y_2 và Y_3 theo (1). Mô hình phân tích nhân tố đơn giản nhất dựa trên hai giả định liên quan đến các mối quan hệ (1). Đầu tiên, chúng ta sẽ mô tả những giả định này và sau đó xem xét ý nghĩa của chúng.

1. **A1**: Các sai số e_i độc lập với nhau và sao cho $\mathrm{E}(e_i) = 0$ và $\mathrm{Var}(e_i) = \sigma_i^2$. Có thể coi mỗi e_i là kết quả của một phép chọn ngẫu nhiên có thay thế từ một tập hợp các giá trị e_i có trung bình 0 và một phương sai cụ thể là σ_i^2.
2. **A2**: Các nhân tố không quan sát được F_j độc lập với nhau và với các sai số, và sao cho $\mathrm{E}(F_j) = 0$ và $\mathrm{Var}(F_j) = 1$. Trong ngữ cảnh của ví dụ hiện tại, điều này có nghĩa là không có mối quan hệ nào giữa khả năng định lượng và khả năng nói.

Bây giờ hãy xem xét một số hệ quả của những giả định này. Mỗi biến quan sát được là một hàm tuyến tính của các nhân tố độc lập và các sai số, có thể được viết dưới dạng:

$$Y_i = \beta_{i0} + \beta_{i1}F_1 + \beta_{i2}F_2 + (1)e_i$$

Phương sai của Y_i có thể được tính bằng:

$$\mathrm{Var}(Y_i) = \beta_{i1}^2\mathrm{Var}(F_1) + \beta_{i2}^2\mathrm{Var}(F_2) + (1)^2\mathrm{Var}(e_i) = \beta_{i1}^2 + \beta_{i2}^2 + \sigma_i^2$$

Chúng ta thấy rằng phương sai của Y_i bao gồm hai phần:

$$\mathrm{Var}(Y_i) = \beta_{i1}^2 + \beta_{i2}^2 + \sigma_i^2$$

Trong đó, $\beta_{i1}^2 + \beta_{i2}^2$ được gọi là communality, và σ_i^2 được gọi là phương sai đặc thù (specific variance).

Đầu tiên, communality là phần được giải thích bởi các nhân tố chung F_1 và F_2. Thứ hai, specific variance là phần phương sai của Y_i không được tính bởi các nhân tố chung. Nếu hai nhân tố là dự đoán hoàn hảo về điểm số thì luôn luôn có $e_1 = e_2 = e_3 = 0$ và $\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = 0$.

Để tính hiệp phương sai của bất kỳ hai biến quan sát nào, Y_i và Y_j, chúng ta có thể viết

$$\begin{aligned} Y_i &= \beta_{i0} + \beta_{i1}F_1 + \beta_{i2}F_2 + (1)e_i + (0)e_j; \\ Y_j &= \beta_{j0} + \beta_{j1}F_1 + \beta_{j2}F_2 + (0)e_i + (1)e_j; \end{aligned}$$

và áp dụng các công thức để tính được:

$$\mathrm{Cov}(Y_i, Y_j) = \beta_{i1}\beta_{j1}\mathrm{Var}(F_1) + \beta_{i2}\beta_{j2}\mathrm{Var}(F_2) + (1)(0)\mathrm{Var}(e_i) + (0)(1)\mathrm{Var}(e_j) = \beta_{i1}\beta_{j1} + \beta_{i2}\beta_{j2}.$$

Chúng ta có thể sắp xếp tất cả các phương sai và hiệp phương sai theo dạng bảng sau:

	Y_1	Y_2	Y_3
Y_1	$\beta_{11}^2 + \beta_{12}^2 + \sigma_1^2$	$\beta_{21}\beta_{11} + \beta_{22}\beta_{12}$	$\beta_{31}\beta_{11} + \beta_{32}\beta_{12}$
Y_2	$\beta_{11}\beta_{21} + \beta_{12}\beta_{22}$	$\beta_{21}^2 + \beta_{22}^2 + \sigma_2^2$	$\beta_{21}\beta_{31} + \beta_{22}\beta_{32}$
Y_3	$\beta_{11}\beta_{31} + \beta_{12}\beta_{32}$	$\beta_{21}\beta_{31} + \beta_{22}\beta_{32}$	$\beta_{31}^2 + \beta_{32}^2 + \sigma_3^2$

Chúng ta đã đặt các phương sai của các biến Y vào các ô chéo chính của bảng và các hiệp phương sai ở ngoài đường chéo chính. Đây là các phương sai và hiệp phương sai được gợi ý bởi các giả định của mô hình. Chúng ta sẽ gọi bảng này là ma trận phương sai hiệp phương sai lý thuyết (theoretical variance covariance matrix). Ma trận trên là đối xứng.

Bây giờ ta sẽ chuyển sang dữ liệu có sẵn. Dựa vào các quan sát trên các biến Y_1, Y_2 và Y_3, ta có thể tính toán các phương sai và hiệp phương sai quan sát của các biến rồi xếp chúng vào ma trận phương sai hiệp phương sai quan sát được:

	Y_1	Y_2	Y_3
Y_1	S_1^2	S_{12}	S_{13}
Y_2	S_{21}	S_2^2	S_{23}
Y_3	S_{31}	S_{32}	S_3^2

Do đó, S_1^2 là phương sai quan sát của Y_1, S_{12} là hiệp phương sai quan sát của Y_1 và Y_2, và cứ thế. Trong đó $S_{12} = S_{21}$, $S_{13} = S_{31}$, v.v.;, do ma trận đối xứng. Có thể dễ dàng xác nhận rằng ma trận phương sai hiệp phương sai quan sát được cho dữ liệu của Bảng 1.1 là như sau:

$$\begin{bmatrix} 9.84 & -0.36 & 0.44 \\ -0.36 & 5.04 & 3.84 \\ 0.44 & 3.84 & 3.04 \end{bmatrix}$$

Một mặt, chúng ta có các phương sai và hiệp phương sai quan sát được của các biến; mặt khác, các phương sai và hiệp phương sai được gợi ý bởi mô hình nhân tố. Nếu các giả định của mô hình là đúng, chúng ta sẽ có thể ước lượng các loading β_{ij} sao cho các ước lượng thu được của các phương sai và hiệp phương sai lý thuyết gần với các giá trị quan sát được.

1.4 Mô hình nhân tố

Từ ví dụ trên, ta có các định nghĩa sau. Mô hình nhân tố có thể được coi như một chuỗi các phương trình hồi quy đa biến, dự đoán mỗi biến quan sát được từ các giá trị của các nhân tố chung không quan sát được:

$$Y_1 = \mu_1 + \beta_{11} f_1 + \beta_{12} f_2 + \cdots + \beta_{1m} f_m + \epsilon_1 \tag{2}$$
$$Y_2 = \mu_2 + \beta_{21} f_1 + \beta_{22} f_2 + \cdots + \beta_{2m} f_m + \epsilon_2 \tag{3}$$
$$\vdots \tag{4}$$
$$Y_p = \mu_p + \beta_{p1} f_1 + \beta_{p2} f_2 + \cdots + \beta_{pm} f_m + \epsilon_p \tag{5}$$

Ở đây, giá trị trung bình của biến là μ_1 bởi vì μ_p có thể được coi là các intercept terms cho các mô hình hồi quy đa biến. m là số yếu tố chung, thường $m << p$. Đôi khi, m được biết trước.

Trong đó:

- $Y_i = \sum \beta_{ij} f_j + \epsilon_i$
- f_i là các nhân tố chung hoặc nhân tố tố ẩn. Chúng không tương quan với nhau và mỗi yếu tố có giá trị trung bình bằng 0 và phương sai bằng 1.
- β_{ij} là các hệ số của các nhân tố chung = factor loadings.
- ϵ_i là nhân tố đặc trưng (unique factors) liên quan đến một trong các biến ban đầu. Các ϵ_i và f_j không tương quan với nhau.

Các hệ số hồi quy (β_{ij}) cho tất cả các phương trình hồi quy đa biến này được gọi là hệ số tải nhân tố (factor loading) . Ở đây, β_{ij} là loading của biến thứ i trên yếu tố thứ j. Các loading này được thu thập thành một ma trận được thể hiện dưới đây:

$$\mathbf{B} = \begin{pmatrix} \beta_{11} & \beta_{12} & \dots & \beta_{1m} \\ \beta_{21} & \beta_{22} & \dots & \beta_{2m} \\ \vdots & \vdots & & \vdots \\ \beta_{p1} & \beta_{p2} & \dots & \beta_{pm} \end{pmatrix} = \text{matrix of factor loadings}$$

Và cuối cùng, các sai số (ϵ_i) được gọi là các nhân tố chuyên biệt (specific factor). Ở đây, ϵ_i là specific factor cho biến i. Các specific factor này cũng được thu thập vào một vector:

$$\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_p \end{pmatrix} = \text{vector of specific factors}$$

Tóm lại, mô hình cơ bản giống như một mô hình hồi quy. Mỗi biến phản ứng Y được dự đoán như là một hàm tuyến tính của các nhân tố chung không quan sát được $f_1, f_2, \dots, f_m$. Do đó, các biến giải thích là $f_1, f_2, \dots, f_m$. Ta có m yếu tố không quan sát được điều khiển sự biến thiên trong dữ liệu.

Nói chung, giảm bớt các tập hợp trên thành biểu thức ma trận được thể hiện ở dạng dưới đây:

$$Y = \mu + \beta F + \epsilon$$

2 Các phương pháp phân tích nhân tố

Có nhiều phương pháp khác nhau để ước tính các tham số của mô hình nhân tố, ở bài báo cáo này em chỉ đề cập đến Phương pháp thành phần chính (Principal Component Method) và Ước lượng hợp lý cực đại (Maximum Likelihood Estimation)

2.1 Phương pháp thành phần chính

Phương pháp thành phần chính là một kỹ thuật thống kê được sử dụng để giảm số lượng biến trong một tập dữ liệu lớn mà vẫn giữ lại phần lớn thông tin ban đầu. Phương pháp này thường được sử dụng trong phân tích nhân tố để xác định các yếu tố chính ảnh hưởng đến các biến quan sát.

Mục tiêu của phương pháp này là làm giảm chiều dữ liệu bằng cách tìm các thành phần chính (principal components) mà tối đa hóa phương sai của dữ liệu.

2.1.1 Lý thuyết

Cho $\mathbf{X_i}$ là vector các quan sát cho i đối tượng:

$$\mathbf{X_i} = \begin{pmatrix} X_{i1} \\ X_{i2} \\ \vdots \\ X_{ip} \end{pmatrix}$$

trong đó X_{ip} là quan sát thứ p của đối tượng thứ i.

Ma trận hiệp phương sai - phương sai mẫu được biểu diễn là:

$$\mathbf{S} = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{X_i} - \bar{\mathbf{x}})(\mathbf{X_i} - \bar{\mathbf{x}})'$$

Chúng ta có p trị riêng cho ma trận phương sai- hiệp phương sai này cũng như các vector riêng tương ứng cho ma trận này.

Trị riêng của $\mathbf{S}$:

$$\hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_p$$

Vector riêng của $\mathbf{S}$:

$$\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \ldots, \hat{\mathbf{e}}_p$$

Ta có thể biểu diễn lại được ma trận phương sai- hiệp phương sai dưới dạng sau dựa trên các giá trị riêng và các vector riêng tương ứng.

Phân rã phổ của $\sum$:

$$\Sigma = \sum_{i=1}^{p} \lambda_i \mathbf{e_i e_i'} \cong \sum_{i=1}^{m} \lambda_i \mathbf{e_i e_i'} = \begin{pmatrix} \sqrt{\lambda_1}\mathbf{e_1} & \sqrt{\lambda_2}\mathbf{e_2} & \dots & \sqrt{\lambda_m}\mathbf{e_m} \end{pmatrix} \begin{pmatrix} \sqrt{\lambda_1}\mathbf{e_1'} \\ \sqrt{\lambda_2}\mathbf{e_2'} \\ \vdots \\ \sqrt{\lambda_m}\mathbf{e_m'} \end{pmatrix} = \mathbf{BB'}$$

Ý tưởng đằng sau phương pháp thành phần chính là xấp xỉ biểu thức này. Thay vì tổng từ 1 đến p, ta giờ chỉ cần tổng từ 1 đến m, bỏ qua thành phần cuối cùng $p - m$ trong tổng, và thu được biểu thức thứ ba. Ta có thể viết lại điều này như trong biểu thức thứ tư, được sử dụng để định nghĩa ma trận các hệ số tải nhân tố, $\mathbf{B}$, dẫn đến biểu thức cuối cùng trong matrix notation.

Lưu ý: Nếu sử dụng các phép đo đã chuẩn hóa, chúng ta thay $\mathbf{S}$ bằng ma trận tương quan mẫu $\mathbf{R}$.

Điều này dẫn đến ước lượng sau cho các hệ số tải nhân tố:

$$\hat{\beta}_{ij} = \hat{e}_{ji}\sqrt{\hat{\lambda}_j}$$

Điều này tạo thành ma trận $\mathbf{B}$ của các hệ số tải nhân tố trong phân tích nhân tố. Điều này đươc theo sau bởi ma trận chuyển vị của $\mathbf{B}$. Để ước lượng phương sai cụ thể, nhớ rằng mô hình nhân tố cho ma trận phương sai-hiệp phương sai của chúng ta là:

$$\mathbf{\Sigma} = \mathbf{BB'} + \mathbf{\Psi}$$

trong ký hiệu ma trận. $\mathbf{\Psi}$ sẽ bằng ma trận phương sai-hiệp phương sai trừ đi $\mathbf{BB'}$.

$$\mathbf{\Psi} = \mathbf{\Sigma} - \mathbf{BB'}$$

Điều này gợi ý rằng các phương sai đặc thù (specific variances), các phần tử trên đường chéo của $\mathbf{\Psi}$, được ước lượng bằng biểu thức sau:

$$\hat{\Psi}_i = s_i^2 - \sum_{j=1}^{m} \lambda_j \hat{e}_{ji}^2$$

Chúng ta lấy phương sai mẫu cho biến thứ i và trừ đi tổng bình phương các hệ số tải nhân tố (tức là commonality).

2.1.2 Ví dụ

Ta xét ví dụ sau trên Tập data Places Rated sử dụng Phương pháp thành phần chính.

Places Rated Almanac (Boyer and Savageau) đánh giá 329 cộng đồng theo chín tiêu chí:

1. Climate and Terrain
2. Housing
3. Health Care and Environment
4. Crime
5. Transportation
6. Education
7. The Arts
8. Recreation
9. Economic

Trừ housing và crime, điểm càng cao càng tốt. Đối với housing và crime, điểm càng thấp càng tốt.

Mục tiêu của chúng ta ở đây là mô tả mối quan hệ giữa các biến.

Trước khi thực hiện phân tích nhân tố, chúng ta cần xác định m. Nên cần bao nhiêu nhân tố chung được bao gồm trong mô hình? Điều này đòi hỏi phải xác định có bao nhiêu thông số sẽ được liên quan.

Với p = 9, ma trận phương sai-hiệp phương sai $\sum$ chứa

$$\frac{p(p+1)}{2} = \frac{9 \times 10}{2} = 45$$

unique elements hoặc entries. Đối với phân tích nhân tố với m nhân tố, số lượng tham số trong mô hình nhân tố bằng

$$p(m+1) = 9(m+1)$$

Lấy m = 4, chúng ta có 45 tham số trong mô hình nhân tố, điều này bằng số lượng tham số ban đầu. Điều này sẽ dẫn đến không có việc giảm chiều trong trường hợp này. Do đó, trong trường hợp này, chúng ta sẽ chọn $m = 3$, và tạo ra 36 tham số trong mô hình nhân tố và do đó có sự giảm chiều trong phân tích của chúng ta.

Chúng ta cần chọn m sao cho một lượng đủ lớn của sự biến thiên trong dữ liệu được giải thích. Mức đủ là tất nhiên là tương đối và phụ thuộc vào ví dụ cụ thể.

Chương trình python:

```
from factor_analyzer import FactorAnalyzer
import numpy as np
import pandas as pd

# Đọc dữ liệu
```

```
data_path = '/content/places_tf.csv'
data = pd.read_csv(data_path)

# Log transformation trên các cột (trừ cột cuối cùng)
transformed_columns = data.columns[:-1]
data_transformed[transformed_columns] =
    data[transformed_columns].applymap(lambda x: np.log10(x))

# Xác định các biến để thực hiện phân tích nhân tố
variable = data.columns[:-1]
X = data[variable]

# Số factor
m = 3

# Tạo và fit model
fa = FactorAnalyzer(n_factors=m, method='principal', rotation = None )
fa.fit(X)

# Kiểm tra eigenvalues
eigenvalues, _ = fa.get_eigenvalues()
print("Eigenvalues:", eigenvalues)

# Lấy factor loadings
factor_loadings = fa.loadings_

# Hiển thị kết quả dưới dạng bảng
print("Factor Loadings:")
print(pd.DataFrame(factor_loadings, index=variable,
        columns=[f"Factor {i+1}" for i in range(m)]))

communalities = np.sum(factor_loadings**2, axis=1)
total_communality = np.sum(communalities)

print("Communalities:", communalities)
print("Total Communality:", total_communality)
```

Kết quả sau khi chạy chương trình:

```
Eigenvalues: [3.20788406 1.21888092 1.10461342 0.92433502 0.86004479
    0.57763963 0.48214885 0.33014925 0.29430406]
Factor Loadings:
          Factor 1  Factor 2  Factor 3
climate   0.264538  0.104883  0.858217
housing   0.699044  0.144250  0.050406
health    0.709545 -0.436887  0.003515
```

```
crime     0.465943  0.532585  0.168383
trans     0.686249 -0.162201 -0.137764
educate   0.489905 -0.499309 -0.193239
arts      0.839382 -0.103141  0.006206
recreate  0.647737  0.308219  0.004496
econ      0.305964  0.575587 -0.529898
Communalities: [0.81751791 0.51201084 0.6943371  0.52910188 0.51622613
    0.5266573 0.71523884 0.51458278 0.70570563]
Total Communality: 5.53137839855572
```

Việc diễn giải các hệ số tải nhân tố tương tự như việc diễn giải các hệ số trong phân tích thành phần chính. Chúng ta muốn xác định một số tiêu chí để đưa vào, mà trong nhiều trường hợp có thể hơi tùy ý. Trong bảng trên, sử dụng ngưỡng khoảng 0.5, các giá trị mà ta coi là lớn sẽ lớn hơn ngưỡng này. Các phát biểu sau đây dựa trên tiêu chí này:

1. Nhân tố 1 tương quan mạnh mẽ với arts (0.840) và cũng tương quan với health, housing, recreate và ở mức độ thấp hơn là với crime và educate. Có thể nói rằng nhân tố 1 chủ yếu là thước đo của các biến này.
2. Tương tự, nhân tố 2 tương quan chặt chẽ với crime, educate và econ. Có thể nói rằng nhân tố 2 chủ yếu là thước đo của các biến này.
3. Tương tự, nhân tố 3 tương quan mạnh với climate và econ. Có thể nói rằng nhân tố 3 chủ yếu là thước đo của các biến này.

Các giá trị cộng đồng (communalities) cho biến thứ i được tính bằng cách lấy tổng của các bình phương các hệ số tải nhân tố của biến đó. Điều này được thể hiện qua công thức:

$$\hat{h}_i^2 = \sum_{j=1}^{m} \hat{\beta}_{ij}^2$$

Ví dụ, để tính giá trị communalities cho Climate, biến đầu tiên. Ta bình phương các hệ số tải nhân tố của Climate sau đó cộng các kết quả lại:

$$\hat{h}_1^2 = 0.264538^2 + 0.104883^2 + 0.858217^2 = 0.8175$$

Các giá trị communalities của 9 biến có thể được lấy từ kết quả chương trình phía trên như sau:

Ta có thể xem các giá trị này như giá trị R^2 trong các mô hình hồi quy dự đoán các biến quan tâm từ 3 nhân tố. Giá trị communality cho một biến cụ thể có thể được hiểu như tỷ lệ biến thiên trong biến đó được giải thích bởi ba nhân tố. Nói cách khác, nếu ta thực hiện hồi quy bội biến của Climate theo ba nhân tố chung, ta sẽ thu được $R^2 = 0.818$, chỉ ra rằng khoảng 81% biến thiên trong Climate được giải thích bởi mô hình nhân tố. Kết quả cho thấy phân tích nhân tố thực hiện tốt nhất trong việc giải thích các biến thiên về Climate, Arts, Economics và Health.

Variable	Communality
Climate	0.818
Housing	0.512
Health	0.694
Crime	0.529
Transportation	0.516
Education	0.527
Arts	0.715
Recreation	0.515
Economics	0.706
Total	5.532

Một cách đánh giá mức độ hiệu quả của mô hình này có thể được thực hiện thông qua các giá trị communalities. Điều ta muốn thấy là các giá trị này gần bằng một. Điều này cho thấy rằng mô hình giải thích hầu hết sự biến thiên của các biến đó. Trong trường hợp này, mô hình hoạt động tốt hơn đối với một số biến so với các biến khác. Mô hình giải thích tốt nhất đối với biến Climate và khá tốt đối với các biến như Economics, Health và Arts. Tuy nhiên, đối với các biến khác như Crime, Recreation, Transportation và Housing, mô hình không hoạt động tốt, chỉ giải thích được khoảng một nửa sự biến thiên.

Tổng của tất cả các giá trị communality là total communality:

$$\sum_{i=1}^{p} \hat{h}_i^2 = \sum_{i=1}^{m} \hat{\lambda}_i$$

Ở đây, total communality là 5.532. Tỷ lệ của tổng biến thiên được giải thích bởi ba yếu tố là

$$\frac{5.532}{9} = 0.614$$

Đây là tỷ lệ phần trăm giải thích của biến trong mô hình của trên. Điều này có thể được coi là một đánh giá tổng quan về hiệu quả hoạt động của mô hình. Tuy nhiên, tỷ lệ phần trăm này cũng giống như tỷ lệ biến thiên được giải thích bởi ba giá trị riêng đầu tiên thu được trước đó. Các giá trị communalities riêng lẻ cho biết mô hình hoạt động tốt như thế nào đối với các biến riêng lẻ và total communality sẽ đưa ra đánh giá tổng thể về hiệu suất. Đây là hai đánh giá khác nhau.

Vì dữ liệu được chuẩn hóa nên phương sai của dữ liệu được chuẩn hóa bằng một. Các phương sai đặc thù được sẽ được tính bằng cách trừ đi giá trị communality khỏi phương sai như công thức dưới đây:

$$\hat{\Psi}_i = 1 - \hat{h}_i^2$$

Ví dụ: Phương sai cụ thể cho biến Climate được tính như sau:

$$\hat{\Psi}_1 = 1 - 0.818 = 0.182$$

Các phương sai cụ thể được tìm thấy trong kết quả của chương trình python dưới dạng các phần tử đường chéo trong bảng như bên dưới.

Residual Correlation Matrix with Uniqueness on the Diagonal:

	climate	housing	health	crime	trans	educate	arts	recreate	econ
climate	0.182482	0.008871	-0.013247	-0.089372	-0.034654	0.167566	-0.073148	-0.101656	0.214087
housing	0.008871	0.487989	-0.021826	-0.271851	-0.128048	-0.059192	-0.083189	-0.035730	0.027856
health	-0.013247	-0.021826	0.305663	0.063124	-0.156701	-0.118082	-0.029332	-0.096288	0.081195
crime	-0.089372	-0.271851	0.063124	0.470898	0.061539	0.122669	-0.009968	-0.183870	-0.082962
trans	-0.034654	-0.128048	-0.156701	0.061539	0.483774	-0.135154	-0.053082	-0.001850	-0.126913
educate	0.167566	-0.059192	-0.118082	0.122669	-0.135154	0.473343	-0.144864	-0.068097	0.164203
arts	-0.073148	-0.083189	-0.029332	-0.009968	-0.053082	-0.144864	0.284761	-0.015954	-0.059918
recreate	-0.101656	-0.035730	-0.096288	-0.183870	-0.001850	-0.068097	-0.015954	0.485417	-0.197861
econ	0.214087	0.027856	0.081195	-0.082962	-0.126913	0.164203	-0.059918	-0.197861	0.294294

Ví dụ: Phương sai cụ thể của biến housing là 0,488. Mô hình này cung cấp một giá trị gần đúng cho ma trận tương quan. Ta có thể đánh giá tính phù hợp của mô hình với phần dư thu được từ phép tính sau:

$$s_{ij} - \sum_{k=1}^{m} \beta_{ik}\beta_{jk}; i \neq j = 1, 2, \ldots, p$$

Về cơ bản, đây là sự khác biệt giữa $\mathbf{R}$ và $\mathbf{BB}'$ hoặc mối tương quan giữa biến i và j trừ đi giá trị kỳ vọng theo mô hình. Nói chung, những phần dư này phải càng gần bằng 0 càng tốt. Ví dụ: phần dư giữa Housing và Climate là 0.008871, gần bằng 0. Tuy nhiên có một số giá trị chưa được tốt lắm. Phần dư giữa Climate và Economy là 0.214087. Các giá trị này cho biết mức độ phù hợp của mô hình nhân tố với dữ liệu.

Một nhược điểm của phương pháp thành phần chính là nó không cung cấp phép kiểm tra sự không phù hợp. Ta có thể kiểm tra những con số này và xác định xem chúng có nhỏ hoặc gần bằng 0 không, nhưng ta thực sự không có một phép kiểm tra nào cho điều này. Và một phép kiểm tra như vậy có sẵn trong phương pháp hợp lý cực đại (Maximun Likelihood Medthod).

2.2 Phương pháp ước lượng hợp lý cực đại

Phương pháp ước lượng hợp lý cực đại (Maximum Likelihood Estimation - MLE) là một kỹ thuật thống kê được sử dụng để ước lượng các tham số của mô hình nhân tố. MLE tìm các giá trị tham số sao cho khả năng quan sát dữ liệu thực tế được tối đa hóa.

Mục tiêu của phương pháp này là tối đa hóa hàm khả dĩ (likelihood function) để tìm các giá trị tham số tối ưu. Đảm bảo rằng các tham số ước lượng được là tốt nhất theo nghĩa là chúng làm cho dữ liệu quan sát được trở nên khả dĩ nhất.

2.2.1 Giả định

Ước lượng hợp lý cực đại yêu cầu dữ liệu được lấy mẫu từ một phân phối chuẩn đa biến. Đây là một nhược điểm của phương pháp này. Dữ liệu thường được thu thập trên thang

đo Likert, đặc biệt là trong lĩnh vực khoa học xã hội. Vì thang đo Likert là rời rạc và bị chặn nên các dữ liệu này không thể tuân theo phân phối chuẩn.

Sử dụng phương pháp MLE, chúng ta phải giả định rằng dữ liệu được lấy mẫu độc lập từ một phân phối chuẩn đa biến với vector trung bình μ và ma trận phương sai-hiệp phương sai có dạng:

$$\mathbf{\Sigma} = \mathbf{B}\mathbf{B}' + \mathbf{\Psi}$$

trong đó $\mathbf{B}$ là ma trận của hệ số tải nhân tố và Ψ là ma trận chéo của các phương sai đặc thù.

Ta định nghĩa thêm ký hiệu: Như thường lệ, các vector dữ liệu cho n đối tượng được biểu diễn như sau:

$$\mathbf{X_1}, \mathbf{X_2}, \ldots, \mathbf{X_n}$$

Ước lượng hợp lý cực đại bao gồm việc ước lượng giá trị trung bình, ma trận của hệ số tải nhân tố và các phương sai đặc thù. Ước lượng hợp lý tối đa cho vector trung bình μ, các hệ số tải nhân tố $\mathbf{B}$ và các phương sai đặc thù Ψ được thu được bằng cách tìm $\hat{\mu}$, $\hat{\mathbf{B}}$ và $\hat{\mathbf{\Psi}}$ sao cho tối đa hóa log-likelihood được cho bởi biểu thức sau:

$$l(\mu, \mathbf{B}, \mathbf{\Psi}) = -\frac{np}{2}\log 2\pi - \frac{n}{2}\log|\mathbf{B}\mathbf{B}' + \mathbf{\Psi}| - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{X_i} - \mu)'(\mathbf{B}\mathbf{B}' + \mathbf{\Psi})^{-1}(\mathbf{X_i} - \mu)$$

Log của phân phối xác suất chung của dữ liệu được cực đại hóa. Ta muốn tìm các giá trị của các tham số (μ, $\mathbf{B}$ và Ψ) tương thích nhất với những gì ta thấy trong dữ liệu. Như đã đề cập trước đó, các nghiệm cho các mô hình nhân tố này không phải là duy nhất. Các mô hình tương đương có thể thu được bằng cách quay (rotation). Nếu như $\mathbf{B}'\mathbf{\Psi}^{-1}\mathbf{B}$ là một ma trận chéo, thì chúng ta có thể thu được một nghiệm duy nhất.

Về mặt tính toán, quá trình này rất phức tạp. Nói chung, không có nghiệm dạng đóng cho bài toán cực đại hóa này, vì vậy các phương pháp lặp được áp dụng. Việc triển khai các phương pháp lặp có thể gặp vấn đề như chúng ta sẽ thấy sau.

2.2.2 Ví dụ

Ta xét ví dụ sau trên Tập data Places Rated như ở mục 2.1.2 bằng phương pháp trích xuất MLE.

Trước khi tiến hành, ta muốn xác định xem mô hình có phù hợp với dữ liệu hay không. Mức độ phù hợp của fit test (goodness-of-fit test) trong trường hợp này so sánh ma trận phương sai-hiệp phương sai theo một mô hình tiết kiệm (parsimonious model) với ma trận phương sai-hiệp phương sai không có bất kỳ ràng buộc nào, tức là theo giả định rằng các phương sai và hiệp phương sai có thể nhận bất kỳ giá trị nào. Ma trận phương sai-hiệp phương sai theo mô hình giả định có thể được biểu diễn như sau:

$$\mathbf{\Sigma} = \mathbf{B}\mathbf{B}' + \mathbf{\Psi}$$

$\mathbf{B}$ là ma trận của hệ số tải nhân tố, và các phần tử trên đường chéo của Ψ bằng với các phương sai đặc thù. Đây là một cấu trúc rất cụ thể cho ma trận phương sai-hiệp phương sai. Một cấu trúc tổng quát hơn sẽ cho phép các phần tử đó nhận bất kỳ giá trị nào.

Để đánh giá mức phù hợp, ta sử dụng kiểm định thống kê Bartlett-Corrected Likelihood Ratio:

$$X^2 = \left(n - 1 - \frac{2p + 4m - 5}{6}\right) \log \frac{|\hat{\mathbf{B}}\hat{\mathbf{B}}' + \hat{\mathbf{\Psi}}|}{|\hat{\mathbf{\Sigma}}|}$$

Kiểm định ở đây là likelihood ratio test tạm dịch là tỉ số khả dĩ, trong đó hai giá trị likehoods được so sánh, dưới một mô hình tiết kiệm và một không có bất kỳ ràng buộc nào. Hằng số trong thống kê được gọi là hiệu chỉnh Bartlett. Log là log tự nhiên. Trong tử số, ta có định thức của mô hình nhân tố đã phù hợp cho ma trận phương sai-hiệp phương sai, và bên dưới mẫu ta có ước lượng mẫu của ma trận phương sai-hiệp phương sai giả định không có cấu trúc trong đó:

$$\hat{\mathbf{\Sigma}} = \frac{n-1}{n}\mathbf{S}$$

và $\mathbf{S}$ là ma trận phương sai-hiệp phương sai mẫu. Đây chỉ là một ước lượng khác của ma trận phương sai-hiệp phương sai, bao gồm một bias nhỏ. Nếu mô hình nhân tố phù hợp tốt thì hai định thức này sẽ gần bằng nhau và ta sẽ nhận một giá trị nhỏ cho X^2. Tuy nhiên, nếu mô hình không phù hợp, thì các định thức sẽ khác nhau và X^2 sẽ lớn.

Theo giả thuyết không (H_0): mô hình nhân tố mô tả đủ các mối quan hệ giữa các biến:

$$\mathbf{X}^2 \sim \chi^2_{\frac{(p-m)^2-p-m}{2}}$$

Và mô hình nhân tố mô tả đầy đủ dữ liệu, thống kê kiểm định này có phân phối chi-bình phương với một tập hợp các bậc tự do như được trình bày ở trên. Số bậc tự do là sự khác biệt về số lượng tham số duy nhất trong hai mô hình. Nếu giá trị X^2 lớn hơn giá trị tới hạn từ bảng chi-bình phương, chúng ta bác bỏ giả thuyết không, nghĩa là mô hình không khớp tốt với dữ liệu.

Chương trình python:

```
from factor_analyzer import FactorAnalyzer
from factor_analyzer import calculate_bartlett_sphericity
import numpy as np
import pandas as pd
from scipy.stats import chi2

# Đọc dữ liệu
data_path = '/content/places_tf.csv'
data = pd.read_csv(data_path)

# Chuyển đổi các cột bằng log10, trừ cột 'id'
transformed_columns = data.columns[:-1]
data[transformed_columns] =
    data[transformed_columns].applymap(lambda x: np.log10(x))

# Chọn các biến để phân tích
variables = data.columns[:-1]
X = data[variables]
```

```
# Số lượng nhân tố cần trích xuất
m = 3

# Tạo và fit model với phương pháp MLE
fa = FactorAnalyzer(n_factors=m, method='ml', rotation=None)
fa.fit(X)

# Kiểm tra eigenvalues
eigenvalues, _ = fa.get_eigenvalues()
print("Eigenvalues:", eigenvalues)

# Lấy factor loadings
factor_loadings = fa.loadings_

# Hiển thị kết quả dưới dạng bảng
factor_loadings_df = pd.DataFrame(factor_loadings, index=variables,
        columns=[f"Factor {i+1}" for i in range(m)])
print("Factor Loadings:")
print(factor_loadings_df)

# Tính communalities
communalities = np.sum(factor_loadings**2, axis=1)
total_communality = np.sum(communalities)

print("Communalities:", communalities)
print("Total Communality:", total_communality)

# Bartlett's test of sphericity
chi_square_value, p_value = calculate_bartlett_sphericity(X)

# Test: No common factors
df_no_factors = X.shape[1] * (X.shape[1] - 1) / 2
chi_square_no_factors = chi_square_value
p_value_no_factors = chi2.sf(chi_square_no_factors, df_no_factors)
print(f"Test: No Common Factors: DF = {df_no_factors},
    Chi-Square = {chi_square_no_factors:.4f},
    p-value = {p_value_no_factors:.4f}")

# Test: 3 Factors are sufficient
correlation_matrix = np.corrcoef(X, rowvar=False)
uniquenesses = np.diag(fa.get_uniquenesses())
residual_matrix = np.dot(factor_loadings, factor_loadings.T)
        + uniquenesses
result = (X.shape[0] - 1 - (2 * X.shape[1] + 4 * m - 5) / 6) *
    np.log(np.linalg.det(residual_matrix)
```

```
    / np.linalg.det(correlation_matrix))
chi_square_3_factors = result

df_3_factors = ((X.shape[1] - m)**2 - X.shape[1] - m)/2

# Tính p-value
p_value_3_factor = chi2.sf(chi_square_3_factors, df_3_factors)
print(f"Test: 3 Factors are sufficient: DF = {df_3_factors},
    Chi-Square = {chi_square_3_factors:.4f}, p-value = {p_value:.4f}")
```

Kết quả sau khi chạy chương trình:

```
Eigenvalues: [3.20788406 1.21888092 1.10461342 0.92433502 0.86004479
    0.57763963 0.48214885 0.33014925 0.29430406]
Factor Loadings:
          Factor 1  Factor 2  Factor 3
climate   0.252908  0.031210 -0.003257
housing   0.997367 -0.015998 -0.002332
health    0.423219  0.716957 -0.296106
crime     0.145312  0.274142  0.368471
trans     0.330388  0.487917  0.249484
educate   0.208555  0.423197 -0.177128
arts      0.501175  0.646240  0.226077
recreate  0.467759  0.234278  0.460929
econ      0.298108 -0.042619  0.159950
Communalities: [0.06494716 0.99500168 0.78082161 0.23203989 0.40946163
    0.25396516 0.71991403 0.48614021 0.11626855]
Total Communality: 4.058559918253862
Test: No Common Factors: DF = 36.0, Chi-Square = 770.2431,
    p-value = 0.0000
Test: 3 Factors are sufficient: DF = 12.0, Chi-Square = 84.0472,
    p-value = 0.0000
```

Ta có bảng so sánh sau:

Test	**DF**	**Chi-Square**	**Pr > ChiSq**
H_0: No common factors	36	770.24301	< 0.0001
H_A: At least one common factor			
H_0: 3 Factors are sufficient	12	84.0472	< 0.0001
H_A: More Factors are needed			

Bảng 2: *Significance Tests based on 329 Observations*

Đối với tập dữ liệu Places Rated, $X^2 = 84.05; d.f = 12; p < 0.0001$. Ta nhận thấy sự thiếu phù hợp đáng kể. Vậy kết luận rằng các mối quan hệ giữa các biến không được mô tả đầy đủ bởi mô hình nhân tố. Điều này cho thấy rằng ta không có mô hình đúng.

Biện pháp duy nhất trong trường hợp này là tăng số lượng nhân tố cho đến khi đạt được sự phù hợp đầy đủ với dữ liệu. Tuy nhiên, cần lưu ý rằng số lượng nhân tố m phải được xác định một cách hợp lý và không quá lớn để tránh việc mô hình hóa quá mức, thỏa mãn

$$p(m+1) \leq \frac{p(p+1)}{2}$$

Trong ví dụ hiện tại, điều này có nghĩa là $m \leq 4$. Quay lại chương trình python và thay đổi giá trị "m" từ 3 thành 4:

Test	**DF**	**Chi-Square**	**Pr > ChiSq**
H_0: No common factors	36	770.24301	< 0.0001
H_A: At least one common factor			
H_0: 3 Factors are sufficient	6	44.9194	< 0.0001
H_A: More Factors are needed			

Bảng 3: *Significance Tests based on 329 Observations*

Ta nhận thấy rằng mô hình nhân tố với $m = 4$ cũng không phù hợp với dữ liệu, $X^2 = 44.92; d.f = 6; p < 0.0001$. Ta không thể fit đúng mô hình nhân tố để mô tả tập dữ liệu này và kết luận rằng mô hình nhân tố không phù hợp với tập dữ liệu này. Có điều gì đó khác đang diễn ra ở đây, có lẽ là một sự phi tuyến tính. Dù sao đi nữa, có vẻ như điều này không mang lại một mô hình nhân tố phù hợp tốt. Bước tiếp theo có thể là loại bỏ các biến khỏi tập dữ liệu để có được một mô hình phù hợp hơn.

3 Xoay nhân tố

Từ kết luận ở mục trên, có vẻ như mô hình nhân tố không hoạt động tốt. Không có gì đảm bảo rằng bất kỳ mô hình nào cũng sẽ phù hợp với dữ liệu.

Động lực đầu tiên của phân tích nhân tố là cố gắng nhận ra một số nhân tố tiềm ẩn mô tả dữ liệu. Phương pháp ước lượng hợp lý cực đại (Maximum Likelihood Method) đã không thành công trong việc tìm ra một mô hình như vậy để mô tả tập dữ liệu Places Rated. Động lực thứ hai vẫn còn giá trị, đó là cố gắng có được một sự giải thích tốt hơn về dữ liệu. Để làm điều này, ta sẽ xem lại các hệ số tải nhân tố đã thu được trước đây từ phương pháp thành phần chính (principal component method).

Phân tích này gặp vấn đề do một số biến được nhấn mạnh trong nhiều hơn một cột. Ví dụ, Education có vẻ có ý nghĩa đối với cả Nhân tố 1 và Nhân tố 2. Điều tương tự cũng xảy ra với Economics trong cả Nhân tố 2 và Nhân tố 3. Điều này không cung cấp một cách giải thích đơn giản và rõ ràng về dữ liệu. Lý tưởng nhất là mỗi biến chỉ xuất hiện là nhân tố đóng góp quan trọng trong một cột duy nhất.

Trên thực tế, bảng trên có thể cho thấy các kết quả mâu thuẫn nhau. Nhìn vào một số quan sát, có thể thấy rằng một quan sát có thể có giá trị cao ở cả Nhân tố 1 và Nhân tố 2. Nếu điều này xảy ra, một giá trị cao cho Nhân tố 1 cho thấy rằng cộng đồng có chất

	Factor 1	Factor 2	Factor 3
climate	0.264538	0.104883	0.858217
housing	0.699044	0.144250	0.050406
health	0.709545	-0.436887	0.003515
crime	0.465943	0.532585	0.168383
trans	0.686249	-0.162201	-0.137764
educate	0.489905	-0.499309	-0.193239
arts	0.839382	-0.103141	0.006206
recreate	0.647737	0.308219	0.004496
econ	0.305964	0.575587	-0.529898

Bảng 4: *Factor Loadings*

lượng giáo dục tốt, trong khi một giá trị cao cho Nhân tố 2 lại cho thấy ngược lại, rằng cộng đồng có chất lượng giáo dục kém.

Xoay nhân tố được thúc đẩy bởi thực tế là các mô hình nhân tố không là duy nhất. Nhớ lại rằng mô hình nhân tố cho vector dữ liệu $\mathbf{X} = \boldsymbol{\mu} + \mathbf{BF} + \boldsymbol{\epsilon}$ là một hàm của giá trị trung bình $\boldsymbol{\mu}$, cộng với ma trận hệ số tải nhân tố nhân với vector các nhân tố chung, cộng với vector các nhân tố chuyên biệt.

Hơn nữa, ta nên lưu ý rằng điều này tương đương với một mô hình nhân tố xoay $\mathbf{X} = \boldsymbol{\mu} + \mathbf{B}^*\mathbf{F}^* + \boldsymbol{\epsilon}$, trong đó ta đặt $\mathbf{L}^* = \mathbf{BT}$ và $\mathbf{f}^* = \mathbf{T}'\mathbf{f}$ đối với một ma trận trực giao $\mathbf{T}$ mà $\mathbf{T}'\mathbf{T} = \mathbf{TT}' = \mathbf{I}$. Và ta có vô số các ma trận trực giao có thể có, mỗi ma trận tương ứng với một phép xoay nhân tố cụ thể.

Ta sẽ tìm một phép xoay thích hợp, được định nghĩa thông qua một ma trận trực giao $\mathbf{T}$, để mang lại các nhân tố dễ giải thích nhất.

Có hai phương pháp xoay nhân tố: vuông góc (Orthogonal Methods) gồm Varimax, Equimax, Quartimax và không vuông góc (Oblique Methods) gồm Promax, Oblimin, Orthoblique.

Trong đó phép xoay Varimax và phép xoay Promax được sử dụng phổ biến nhất nên trong bài báo cáo này em sẽ sử dụng 2 phép xoay này đại diện cho 2 nhóm phép quay vuông góc và không vuông góc.

3.1 Xoay vuông góc

Phép xoay Varimax là phép xoay vuông góc phổ biến nhất. Sau khi quay trục các nhân tố vẫn ở vị trí vuông góc với nhau. Phép quay này giả định rằng các nhân tố không có sự tương quan với nhau. Phép quay vuông góc ứng dụng nhiều ở các đề tài chỉ có hai loại biến độc lập và phụ thuộc.

Nó liên quan đến việc chia tỷ lệ hệ số tải bằng cách chia chúng cho cộng đồng tương ứng như dưới đây:

$$\tilde{\beta}^*_{ij} = \hat{\beta}^*_{ij}/\hat{h}_i$$

Phép xoay Varimax tìm kiếm phép xoay làm cực đại hóa phương sai của các hệ số tải

bình phương cho từng nhân tố; mục tiêu là làm cho một số hệ số tải này lớn nhất có thể và phần còn lại càng nhỏ càng tốt về giá trị tuyệt đối. Phương pháp varimax khuyến khích việc phát hiện các nhân tố mà mỗi cái chỉ liên quan đến một vài biến. Nó cũng làm giảm khả năng phát hiện các nhân tố ảnh hưởng đến tất cả các biến. Quy trình Varimax, được định nghĩa dưới đây, chọn phép xoay để tối đa hóa:

$$V = \frac{1}{p}\sum_{j=1}^{m}\left\{\sum_{i=1}^{p}(\tilde{\beta}_{ij}^{*})^4 - \frac{1}{p}\left(\sum_{i=1}^{p}(\tilde{\beta}_{ij}^{*})^2\right)^2\right\}$$

Đây là phương sai mẫu của các hệ số tải được chuẩn hóa cho từng nhân tố, được tổng theo m nhân tố

Chương trình python:

```
# Import các thư viên
....
# Đọc dữ liệu
data_path = '/content/places_tf.csv'
data = pd.read_csv(data_path)

# Chuyển đổi các cột bằng log10, trừ cột 'id'
transformed_columns = data.columns[:-1]
data[transformed_columns] =
    data[transformed_columns].applymap(lambda x: np.log10(x))

# Chọn các biến để phân tích
variables = data.columns[:-1]
X = data[variables]

# Số lượng nhân tố cần trích xuất
m = 3

# Tạo và fit model với phương pháp principal và phép xoay varimax
fa = FactorAnalyzer(n_factors=m, method='principal', rotation='varimax')
fa.fit(X)

# Kiểm tra eigenvalues
eigenvalues, _ = fa.get_eigenvalues()
print("Eigenvalues:", eigenvalues)

# Lấy factor loadings
factor_loadings = fa.loadings_

# Hiển thị kết quả dưới dạng bảng
factor_loadings_df = pd.DataFrame(factor_loadings, index=variables,
    columns=[f"Factor {i+1}" for i in range(m)])
print("Factor Loadings:")
```

```
print(factor_loadings_df)

# Tính communalities
communalities = np.sum(factor_loadings**2, axis=1)
total_communality = np.sum(communalities)
print("Communalities:", communalities)
print("Total Communality:", total_communality)
```

Kết quả sau khi chạy chương trình:

```
Eigenvalues: [3.20788406 1.21888092 1.10461342 0.92433502 0.86004479
    0.57763963 0.48214885 0.33014925 0.29430406]
Factor Loadings:
          Factor 1  Factor 2  Factor 3
climate   0.019992  0.204546  0.880499
housing   0.449567  0.538528  0.141022
health    0.819069  0.087685  0.125595
crime     0.014132  0.697724  0.205143
trans     0.653395  0.297190 -0.031296
educate   0.714001 -0.086628 -0.096725
arts      0.716198  0.430690  0.129633
recreate  0.316055  0.638966  0.080090
econ     -0.037389  0.666500 -0.509986
Communalities: [0.81751791 0.51201084 0.6943371  0.52910188 0.51622613
    0.5266573 0.71523884 0.51458278 0.70570563]
Total Communality: 5.531378398555718
```

Bây giờ ta giải thích dữ liệu dựa trên phép xoay. Ta đưa ra các cách giải thích sau, việc giải thích rõ ràng hơn nhiều so với phân tích ban đầu:

- **Nhân tố 1**: Chủ yếu là thước đo về Health, nhưng cũng tăng cùng với điểm số của Transportation, Education và Arts.
- **Nhân tố 2**: Chủ yếu là thước đo về Crime, Recreation, Economy và Housing.
- **Nhân tố 3**: Chủ yếu là thước đo của Climate.

Đây chỉ là mô hình hiện có trong dữ liệu và không nên đưa ra các suy luận nhân quả nào từ các giải thích này. Nó không cho ta biết tại sao mô hình này tồn tại. Rất có thể có những yếu tố quan trọng khác mà chúng ta chưa được thấy ở đây.

Hãy xem xét lượng biến thiên được giải thích bởi các nhân tố trong mô hình đã xoay và so sánh với mô hình ban đầu. Xem xét phương sai được giải thích bởi mỗi nhân tố trong phân tích ban đầu và các nhân tố đã xoay:

Có thể thấy tổng lượng biến thiên được giải thích bởi 3 nhân tố vẫn không thay đổi. Các phép xoay, trong một số lượng nhân tố cố định, không thay đổi lượng biến thiên mà mô hình giải thích. Độ phù hợp của mô hình không thay đổi bất kể sử dụng phép xoay nào.

Factor	Original	Rotated
1	3.2079	3.2079
2	1.2189	1.2189
3	1.1046	1.2189
Total	5.5313	5.5313

Bảng 5: *Analysis*

Phép xoay giúp làm rõ ràng cách giải thích hơn. Lý tưởng nhất, ta nên tìm thấy các số trong mỗi cột cách xa 0 hoặc là gần bằng 0. Các số gần +1 hoặc -1 hoặc 0 trong mỗi cột sẽ cho cách giải thích lý tưởng hoặc rõ ràng nhất. Tuy nhiên, mục tiêu của ta là giải thích dữ liệu. Sự thành công của phân tích có thể được đánh giá bằng việc nó giúp ta hiểu dữ liệu của mình tốt đến đâu. Nếu kết quả cho ta cái nhìn sâu sắc về mô hình biến thiên trong dữ liệu, ngay cả khi không hoàn hảo, thì phân tích đó đã thành công.

Một số phương pháp xoay vuông góc khác chẳng hạn như phép xoay quartimax, phép xoay này tìm cách cực đại hóa phương sai của các hệ số tải bình phương cho mỗi biến và có xu hướng tạo ra các nhân tố có hệ số tải cao cho tất cả các biến. Phép xoay Equimax được cải tiến từ varimax. Nó điều chỉnh theo số lượng nhân tố được xoay, dẫn đến một tập các nhân tố phân phối đồng đều hơn so với varimax và tạo ra các nhân tố ít tổng quát hơn.

3.2 Xoay không vuông góc

Trong phép xoay không vuông góc, các trục mới có thể tự do lấy bất kỳ vị trí nào trong không gian nhân tố, nhưng mức độ tương quan cho phép giữa các nhân tố thường là nhỏ vì hai nhân tố có tương quan cao thường được hiểu là chỉ có một nhân tố. Do đó, các phép xoay không vuông góc sẽ làm giảm bớt ràng buộc về tính trực giao để đạt được sự đơn giản trong diễn giải, nhưng phép xoay này được sử dụng ít hơn so với các phép xoay vuông góc.

Với các phép xoay không vuông góc, phép xoay promax có ưu điểm là nhanh chóng và đơn giản về mặt khái niệm, sau khi xoay trục các nhân tố sẽ di chuyển đến vị trí phù hợp nhất. Phép xoay này giả định các nhân tố có sự tương quan với nhau. Nó cần hai bước. Bước đầu tiên xác định ma trận mục tiêu, hầu như luôn thu được là kết quả của phép xoay varimax mà các giá trị của nó được nâng lên một số mũ (thường từ 2 đến 4) để "ép" cấu trúc của các hệ số tải trở thành lưỡng cực. Bước thứ hai đạt được bằng cách tính toán sự phù hợp của bình phương tối thiểu từ nghiệm varimax cho ma trận mục tiêu.

Phép xoay Oblimin cố gắng tạo ra cấu trúc ma trận mẫu nhân tố đơn giản bằng cách sử dụng một tham số kiểm soát mức độ tương quan giữa các nhân tố. Nó tìm kiếm cấu trúc tốt nhất trong khi tối thiểu hóa các hệ số tải mạnh và sự tương quan giữa các nhân tố. Dùng phép xoay này ta có thể tự thiết lập mức độ tương quan giữa các nhân tố. Ngoài ra còn có phép xoay Quartimin, nó cung cấp một giải pháp tốt cho các dữ liệu phức tạp nhưng lại tạo ra sự thiên vị đối với các đặc điểm có mối tương quan cao khi tạo ra các nhân tố.

Cả hai phép xoay vuông góc và không vuông góc đều nhằm mục đích làm cho hệ số tải

nhân tố của các biến quan sát sẽ tối đa ở trục nhân tố chúng đo lường và tối thiểu ở các trục nhân tố khác. Và không có sự khác biệt lớn khi sử dụng 2 loại phép xoay này. Chính vì vậy, ta có thể sử dụng bất cứ phép xoay nào phù hợp để có được cấu trúc ma trận xoay tốt nhất.

4 Các loại phân tích nhân tố

4.1 Phân tích nhân tố khám phá (Exploratory Factor Analysis)

4.1.1 Mục tiêu

Các mục tiêu chính của Phân tích nhân tố khám phá (EFA) là xác định:

- Số lượng các nhân tố chung ảnh hưởng đến một tập hợp các đo lường.
- Mức độ tương quan của mối quan hệ giữa mỗi nhân tố và mỗi đo lường quan sát được.

EFA sử dụng trong các trường hợp cơ bản sau:

- Để giảm một số lượng lớn các biến thành phần các nhân tố nhỏ hơn.
- Để chọn một tập hợp nhỏ các biến từ một tập hợp lớn ban đầu.
- Để tạo ra một tập hợp các nhân tố, mà tập các nhân tố này được xem như là các biến không có tương quan với nhau. Đây chính là một cách tiếp cận để xử lý vấn đề đa cộng tuyến (Multicollinerity) trong mô hình hồi quy bội.
- Để xác định tính hợp lệ của thang đo.

4.1.2 Quy trình phân tích nhân tố khám phá

Có sáu bước cơ bản để thực hiện EFA:

1. Xác định các biến đo lường
2. Kiểm tra giả thuyết
3. Xác định ma trận tương quan
4. Trích xuất nhân tố
5. Xoay nhân tố
6. Hiệu chỉnh và diễn giải cấu trúc nhân tố

Ở các bước trên, em đã trình bày về lý thuyết của mỗi bước. Ở đây em chỉ giải thích sơ lược về một số bước. Đầu tiên là phần kiểm tra giả thuyết.

Kiểm định Bartlett: Kiểm định Bartlett là một kiểm định thống kê dùng để kiểm tra giả thuyết rằng ma trận tương quan của một tập dữ liệu là ma trận đơn vị, tức là các biến không có tương quan với nhau.

Giả thuyết của kiểm định Bartlett:

- **Giả thuyết không (H0)**: Hệ số tương quan của các biến trong ma trận tương quan bằng 0 (các biến không tương quan với nhau).
- **Giả thuyết đối (H1)**: Hệ số tương quan của các biến trong ma trận tương quan khác 0 (các biến có tương quan với nhau).

Điều kiện bác bỏ giả thuyết không: Kiểm định Bartlett yêu cầu giá trị p-value < 0.05 (5%) để bác bỏ H_0. Nếu p-value < 0.05, ta bác bỏ giả thuyết không và kết luận rằng các biến có tương quan với nhau một cách có ý nghĩa ở độ tin cậy 95%.

Hệ số KMO (Kaiser-Meyer-Olkin): Là một chỉ tiêu để xem xét sự thích hợp của FA. Nếu giá trị hệ số KMO của thang đo thuộc khoảng (0.00; 0.50) thì phân tích nhân tố EFA là không thích hợp vì tương quan riêng chiếm tỷ trọng phần lớn. Ta còn có hệ số KMO của biến, nếu hệ số này của biến nhỏ hơn 0.50 thì biến đó nên được xem xét loại bỏ. Sau khi loại biến, giá trị KMO có thể sẽ thay đổi theo hướng tích cực.

Ở bước trích xuất nhân tố, mục tiêu ở bước này là xác định các nhân tố và phương pháp hay được sử dụng là trích xuất thành phần chính. Ngoài ra việc quyết định số nhân tố còn phụ thuộc một số tiêu chuẩn chọn lựa nhân tố khác như bên bảng dưới:

Phương pháp	Tiêu chuẩn
Eigenvalue	> 1.00
Biểu đồ Scree	> 1.00
Phần trăm phương sai	$> 50\%$ (hay > 0.50)

Bảng 6: *Tiêu chuẩn cho việc lựa chọn số lượng nhân tố*

Hiệu chỉnh và diễn giải cấu trúc nhân tố: Quá trình phân tích có thể lặp đi lặp lại vài lần trước khi có kết quả cuối cùng. Nếu bất kì biến nào không thỏa mãn các yêu cầu thì được xem xét và loại ra khỏi thang đo trước khi lặp lại quá trình phân tích. Nếu các hệ số và tiêu chuẩn đều đạt yêu cầu thì kết quả được giải thích thông qua các phần chung Communalities, hệ số tương quan, hệ số tải nhân tố trong ma trận xoay nhân tố (Rotated Factor Matrix), hệ số KMO, tổng phương sai trích (Total Variance Explained).

4.1.3 Ưu và nhược điểm

Ưu điểm

- Khám phá cấu trúc tiềm ẩn: Giúp khám phá cấu trúc tiềm ẩn của dữ liệu mà không cần giả thuyết trước.

- Giảm số lượng biến: Giúp giảm số lượng biến quan sát thành một số ít các nhân tố, làm cho dữ liệu dễ quản lý và phân tích hơn.
- Đánh giá tính hợp lệ: Giúp kiểm tra và xác định tính hợp lệ của các thang đo.
- Linh hoạt trong Ứng dụng: EFA được sử dụng rộng rãi trong nhiều lĩnh vực như tâm lý học, y tế, kinh tế và xã hội học để khám phá cấu trúc của các biến đo lường.

Nhược điểm

- Cần nhiều dữ liệu: EFA thường đòi hỏi một lượng lớn dữ liệu để sản xuất kết quả tin cậy và ổn định.
- Phụ thuộc vào mẫu: Kết quả của EFA có thể bị ảnh hưởng mạnh bởi đặc điểm của mẫu dữ liệu.
- Phức tạp: Đòi hỏi kiến thức chuyên sâu về thống kê và phân tích nhân tố.

4.2 Phân tích nhân tố khẳng định (Confirmatory Factor Analysis)

4.2.1 Mục tiêu

Mục tiêu chính của phân tích nhân tố khẳng định (CFA) là kiểm tra xem dữ liệu quan sát có phù hợp với mô hình nhân tố đã giả định trước hay không.

CFA sử dụng trong một số trường hợp phổ biến sau:

- Thiết lập tính hợp lệ của một mô hình nhân tố đơn.
- So sánh khả năng của hai mô hình khác nhau trong việc giải thích cùng một tập hợp dữ liệu.
- Kiểm định ý nghĩa của một hệ số tải nhân tố cụ thể.
- Kiểm tra mối quan hệ giữa hai hoặc nhiều hệ số tải nhân tố.
- Kiểm tra xem một tập hợp các nhân tố có tương quan hay không tương quan.
- CFA giúp đánh giá tính hợp lệ hội tụ (convergent validity) và tính hợp lệ phân biệt (discriminant validity) của các thang đo, cũng như kiểm tra độ tin cậy nội tại (internal consistency) của các nhân tố.

4.2.2 Quy trình phân tích nhân tố khẳng định

Có sáu bước cơ bản để thực hiện CFA:

1. Xác định mô hình nhân tố

2. Thu thập dữ liệu
3. Ước lượng các tham số trong mô hình, bao gồm trọng số của các nhân tố và phương sai (hoặc tương quan) của các biến quan sát.
4. Khớp mô hình với dữ liệu
5. Đánh giá sự phù hợp của mô hình
6. So sánh với các mô hình khác

4.2.3 Ưu và nhược điểm

Ưu điểm

- Kiểm định giả thuyết: CFA cho phép kiểm định các giả thuyết cụ thể về cấu trúc nhân tố dựa trên lý thuyết đã có.
- Đo lường chính xác: Xác định độ tin cậy và tính hợp lệ của các thang đo sử dụng trong nghiên cứu.
- Phân tích sâu: Cho phép hiểu rõ hơn về mối quan hệ giữa các biến quan sát và nhân tố tiềm ẩn.
- CFA cho phép so sánh độ phù hợp của các mô hình khác nhau để chọn ra mô hình tốt nhất.

Nhược điểm

- Phức tạp: Đòi hỏi sự hiểu biết sâu về thống kê và mô hình hóa.
- Yêu cầu mẫu lớn: Để có kết quả đáng tin cậy, cần có một mẫu lớn, điều này có thể khó khăn trong một số nghiên cứu.
- Nhạy cảm với vi phạm giả định: Kết quả CFA có thể bị ảnh hưởng bởi các vi phạm giả định thống kê như đa cộng tuyến hoặc không tuân theo phân phối chuẩn.

5 Ưu và nhược điểm của phân tích nhân tố

Phương pháp phân tích nhân tố là một kỹ thuật mạnh mẽ trong thống kê, nhưng nó cũng có những ưu điểm và nhược điểm nhất định.

5.1 Ưu điểm

- Giảm chiều dữ liệu: Phân tích nhân tố giúp giảm số lượng biến trong dữ liệu, làm cho việc phân tích và diễn giải dữ liệu trở nên dễ dàng hơn. Giảm chiều dữ liệu mà vẫn giữ lại phần lớn thông tin quan trọng, giúp đơn giản hóa mô hình.
- Khám phá cấu trúc tiềm ẩn: Giúp xác định các yếu tố tiềm ẩn không quan sát được (latent factors) ảnh hưởng đến các biến quan sát được. Cung cấp cái nhìn sâu sắc về cấu trúc dữ liệu và mối quan hệ giữa các biến.
- Loại bỏ đa cộng tuyến: Giảm đa cộng tuyến giữa các biến quan sát, giúp cải thiện độ tin cậy của các mô hình hồi quy và các phân tích thống kê khác.
- Tối ưu hóa mô hình đo lường: Phân tích nhân tố giúp tối ưu hóa và xác thực các công cụ đo lường như bảng câu hỏi hoặc bài kiểm tra, đảm bảo chúng đo lường chính xác các yếu tố cần thiết.
- Ứng dụng rộng rãi: Được sử dụng trong nhiều lĩnh vực khác nhau như tâm lý học, xã hội học, kinh tế học, tiếp thị, giáo dục, và nhiều lĩnh vực khác

5.2 Nhược điểm

- Giả định về phân phối: Phương pháp phân tích nhân tố thường giả định rằng dữ liệu tuân theo phân phối chuẩn. Điều này có thể không luôn đúng trong thực tế. Việc vi phạm giả định này có thể ảnh hưởng đến tính chính xác của kết quả phân tích.
- Khó khăn trong diễn giải: Diễn giải các nhân tố có thể phức tạp và không luôn rõ ràng, đặc biệt khi các hệ số tải nhân tố không rõ ràng. Các yếu tố tiềm ẩn có thể khó hiểu hoặc không có ý nghĩa thực tiễn.
- Số lượng nhân tố: Quyết định số lượng nhân tố cần giữ lại có thể mang tính chủ quan và dựa vào các tiêu chí khác nhau như eigenvalues lớn hơn 1, biểu đồ Scree plot, hoặc kiến thức lĩnh vực. Việc chọn sai số lượng nhân tố có thể dẫn đến kết quả phân tích không chính xác.
- Yêu cầu mẫu lớn: Phân tích nhân tố yêu cầu kích thước mẫu lớn để đảm bảo tính ổn định và độ tin cậy của kết quả. Với mẫu nhỏ, kết quả phân tích có thể không ổn định và khó tái tạo.
- Tương quan nhân tố: Các phương pháp xoay vuông góc giả định rằng các nhân tố không tương quan với nhau, điều này có thể không thực tế trong nhiều tình huống. Phép xoay không vuông góc cho phép các nhân tố có tương quan, nhưng việc diễn giải các nhân tố này có thể phức tạp hơn.

6 Ứng dụng

6.1 Ứng dụng về phân tích nhân tố khám phá

Phân tích nhân tố khám phá với Factor_analyzer trong Python

6.1.1 Mục tiêu

Mục tiêu của phân tích này là xác định các nhân tố tiềm ẩn (latent factors) tác động đến các biến quan sát được trong tập dữ liệu. Bằng cách sử dụng phân tích nhân tố, ta có thể rút gọn số lượng biến ban đầu thành một số ít các nhân tố có ý nghĩa hơn, từ đó giúp dễ dàng hơn trong việc diễn giải và áp dụng các kết quả vào thực tiễn.

6.1.2 Mô tả dữ liệu

Dữ liệu sử dụng trong phân tích này là Baseball Data, bao gồm thông tin về các chỉ số liên quan đến hiệu suất thi đấu của các đội bóng chày. Cụ thể, tập dữ liệu có 2276 dòng và 17 cột.

6.1.3 Ý tưởng chính

Phân tích nhân tố nhằm mục đích giảm số lượng biến quan sát bằng cách xác định các yếu tố tiềm ẩn ảnh hưởng đến các biến này. Các bước chính bao gồm:

1. Kiểm định KMO và Bartlett để xác định tính phù hợp của dữ liệu cho phân tích nhân tố.
2. Xác định số lượng nhân tố cần trích xuất dựa trên biểu đồ scree và các giá trị riêng (eigenvalues).
3. Thực hiện phân tích nhân tố bằng cách xoay Varimax để tìm ra các nhân tố có ý nghĩa.
4. Diễn giải các nhân tố dựa trên kết quả hệ số tải nhân tố (factor loadings).
5. Kiểm tra communalities để xác định mức độ các biến được giải thích bởi các nhân tố.

Quá trình chi tiết

Đầu tiên, ta phải tải các gói cần thiết rồi nhập dữ liệu.

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

```
from sklearn.decomposition import FactorAnalysis
from sklearn.preprocessing import StandardScaler

from factor_analyzer import FactorAnalyzer, calculate_kmo
from factor_analyzer import calculate_bartlett_sphericity
```

Đọc dữ liệu từ file

```
def read_dataset(path):
    df = pd.read_csv(path)
    display(df.head())
    display(df.describe())
    return df

path = '/content/moneyball-training-data.csv'

df = read_dataset(path)
df.info()
```

Tập data ban đầu có các thông tin sau:

```
RangeIndex: 2276 entries, 0 to 2275
Data columns (total 17 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   INDEX             2276 non-null   int64
 1   TARGET_WINS       2276 non-null   int64
 2   TEAM_BATTING_H    2276 non-null   int64
 3   TEAM_BATTING_2B   2276 non-null   int64
 4   TEAM_BATTING_3B   2276 non-null   int64
 5   TEAM_BATTING_HR   2276 non-null   int64
 6   TEAM_BATTING_BB   2276 non-null   int64
 7   TEAM_BATTING_SO   2174 non-null   float64
 8   TEAM_BASERUN_SB   2145 non-null   float64
 9   TEAM_BASERUN_CS   1504 non-null   float64
 10  TEAM_BATTING_HBP  191 non-null    float64
 11  TEAM_PITCHING_H   2276 non-null   int64
 12  TEAM_PITCHING_HR  2276 non-null   int64
 13  TEAM_PITCHING_BB  2276 non-null   int64
 14  TEAM_PITCHING_SO  2174 non-null   float64
 15  TEAM_FIELDING_E   2276 non-null   int64
 16  TEAM_FIELDING_DP  1990 non-null   float64
dtypes: float64(6), int64(11)
memory usage: 302.4 KB
```

Trong đó:

- **INDEX**: Chỉ số định danh của hàng dữ liệu.
- **TARGET_WINS**: Số trận thắng mục tiêu của đội trong mùa giải.
- **TEAM_BATTING_H**: Số lần đánh bóng trúng của đội.
- **TEAM_BATTING_2B**: Số lần đánh bóng trúng đôi của đội.
- **TEAM_BATTING_3B**: Số lần đánh bóng trúng ba của đội.
- **TEAM_BATTING_HR**: Số lần đánh home runs của đội.
- **TEAM_BATTING_BB**: Số lần "đi bộ" (base on balls) của đội.
- **TEAM_BATTING_SO**: Số lần đánh trượt của đội.
- **TEAM_BASERUN_SB**: Số lần cướp chốt của đội.
- **TEAM_BASERUN_CS**: Số lần cướp chốt thất bại của đội.
- **TEAM_BATTING_HBP**: Số lần bị bóng đánh trúng người của đội.
- **TEAM_PITCHING_H**: Số lần đối thủ đánh bóng trúng.
- **TEAM_PITCHING_HR**: Số lần đối thủ đánh bóng trúng nhà (home runs).
- **TEAM_PITCHING_BB**: Số lần "đi bộ" của đối thủ (base on balls).
- **TEAM_PITCHING_SO**: Số lần đối thủ đánh trượt.
- **TEAM_FIELDING_E**: Số lỗi phòng thủ của đội.
- **TEAM_FIELDING_DP**: Số lần bắt bóng đôi của đội.

```
# Kiểm định Sphericity của Bartlett
chi_square_value,p_value=calculate_bartlett_sphericity(scaled_baseball)
print("Chi-square value:", chi_square_value, "P-value:", p_value)

# Kiểm định KMO
kmo_all,kmo_model=calculate_kmo(scaled_baseball)
print("KMO model:", kmo_model)
```

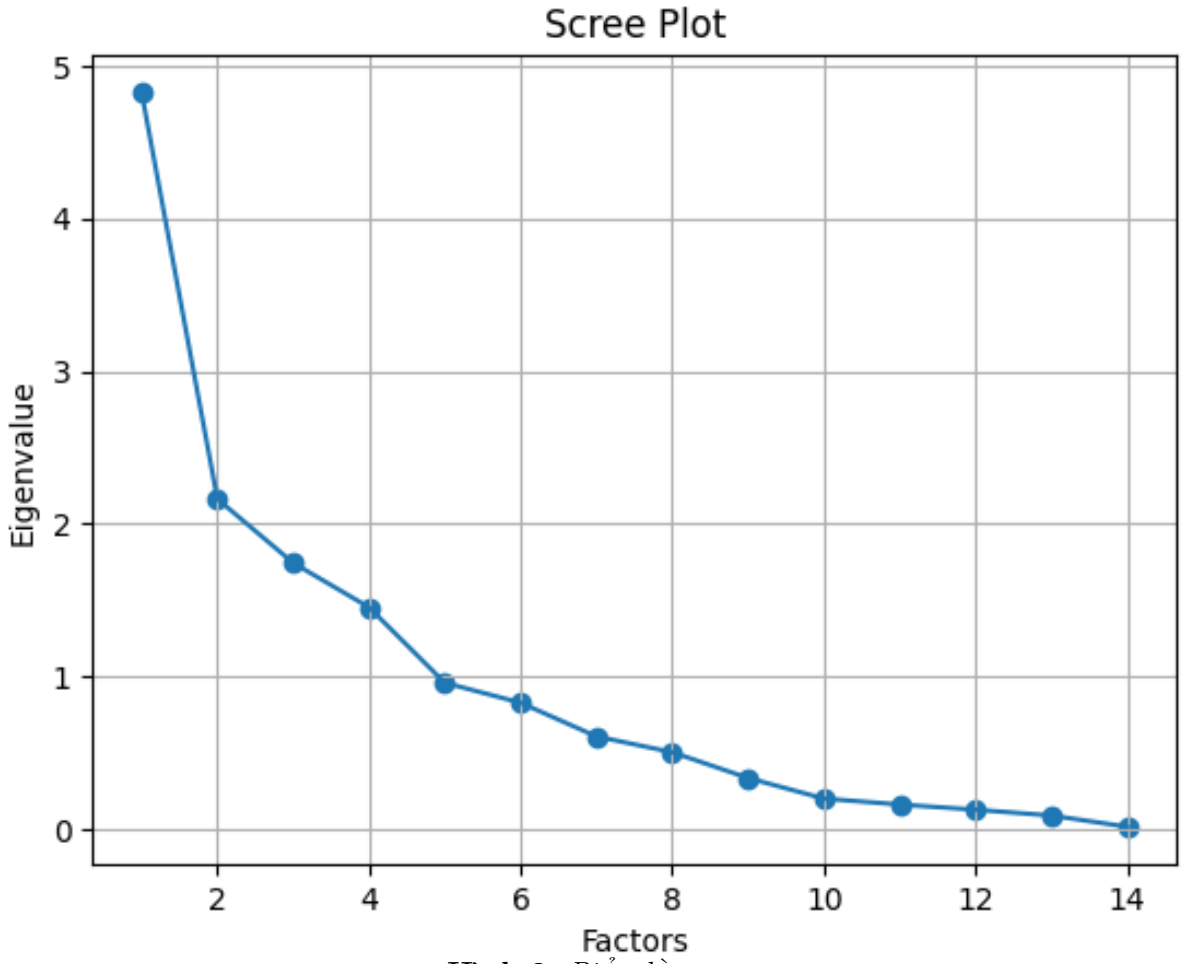
```
Chi-square value: 25914.007256630215 P-value: 0.0
KMO model: 0.6492818754657947
```

Sau khi thực hiện các bước xử lý dữ liệu như sắp xếp, xóa bỏ các cột không liên quan và chuẩn hóa, tiếp theo ta sẽ tiến hành kiểm tra xem phân tích nhân tố có khả thi không? Ta sẽ sử dụng kiểm định Bartlett. Ở đây giá trị p-value bằng 0 cho thấy rằng ma trận tương quan của bộ dữ liệu không phải là ma trận đơn vị và các biến có mối liên quan đáng kể với nhau. Điều này bác bỏ giả thuyết H_0 rằng các biến không tương quan. Vì

vậy, bộ dữ liệu là phù hợp cho phân tích nhân tố, vì có đủ mối liên quan giữa các biến để tiếp tục phân tích này.

Kế tiếp kiểm tra hệ số KMO. Giá trị KMO lớn hơn 0.5 (cụ thể là 0.649) cho thấy rằng tỷ lệ của các hệ số tương quan quan sát được so với các hệ số tương quan riêng là đủ cao để phân tích nhân tố là thích hợp. Điều này có nghĩa là bộ dữ liệu của ta có khả năng hiệu quả cao khi phân tích nhân tố, và ta có thể tiếp tục tiến hành phân tích nhân tố mà không lo ngại về độ tin cậy của các nhân tố được trích xuất

Chọn số lượng nhân tố: Trước tiên, ta sẽ chạy một phân tích nhân tố sơ bộ mà không cần xoay các nhân tố. Bước này nhằm hỗ trợ quyết định về số lượng nhân tố sử dụng trong một giải pháp. Trong bước này, ta sẽ lấy các giá trị riêng (eigenvalues) của nghiệm ban đầu và vẽ chúng trên biểu đồ scree. Ta có thể tìm thấy số lượng các nhân tố được tạo ra so với các giá trị riêng. Các giá trị riêng lớn hơn hoặc bằng 1 cần được xem xét khi chọn số lượng nhân tố. Một nhân tố có giá trị riêng bằng 1 giải thích ít nhất phương sai của một đặc trưng. Phương pháp khuỷu tay (elbow method), mặc dù mang tính chủ quan cao nhưng cũng có thể được sử dụng.



Hình 2: *Biểu đồ scree*

Dựa vào biểu đồ scree ta chọn giải pháp có 4 nhân tố. Và ta có thể thiết lập khi sử dụng thư viện Factor_analyzer. Ta có thể bắt đầu xoay các nhân tố này. Vì FA là một phương pháp lặp lại, nên ta có thể thực hiện theo các bước sau:

1. Bắt đầu với quay Varimax
2. Phương pháp này có thể được đặt là pricipal, minres, ml hoặc master. Ở đây ta có thể sử dụng pricipal trong khi thực hiện phép xoay Varimax.
3. Thay đổi phương pháp thành maximum likelihood nhưng vẫn sử dụng xoay Varimax.
4. Có hai lựa chọn hợp lý cho việc sử dụng hệ số tương quan bội bình phương làm dự đoán đầu tiên cho phân tích nhân tố. Luôn luôn bắt đầu với smc (ví dụ: hệ số tương quan bội bình phương) và thử hệ số tương quan tuyệt đối tối đa làm thứ hai. Chúng ta có thể thiết lập điều này bằng cách đặt use_smc=True.
5. So sánh các giải pháp và giữ lại giải pháp hoạt động tốt nhất.
6. Đánh giá các hệ số tải nhân tố và xem xét một số giải pháp nhân tố khác: một cao hơn và một thấp hơn so với nhân tố đã được chọn (trong trường hợp hiện tại là bốn)
7. Nếu ta đã chia dữ liệu, bây giờ chúng ta có thể thử giải pháp trên dữ liệu thử nghiệm.

```
fa = FactorAnalyzer(4, rotation="varimax", method='principal',
    use_smc=True)
fa.fit(scaled_baseball)

fa.loadings_
loadings = pd.DataFrame(fa.loadings_, columns=['Factor 1',
    'Factor 2', 'Factor 3','Factor 4'], index=scaled_baseball.columns)
print('Factor Loadings \n%s' %loadings)
```

```
Factor Loadings
                 Factor 1  Factor 2  Factor 3  Factor 4
TEAM_BATTING_H   -0.302509  0.855875 -0.116007 -0.180324
TEAM_BATTING_2B   0.140487  0.762211  0.089033  0.227272
TEAM_BATTING_3B  -0.386482  0.208646 -0.146040 -0.728201
TEAM_BATTING_HR   0.523187  0.313963  0.179445  0.688820
TEAM_BATTING_BB   0.815919  0.291915  0.274589 -0.029525
TEAM_BATTING_SO   0.573377 -0.242368  0.287742  0.552559
TEAM_BASERUN_SB  -0.057786 -0.071214  0.177390 -0.752188
TEAM_BASERUN_CS   0.133739 -0.104451 -0.027931 -0.539087
TEAM_PITCHING_H  -0.800344  0.173282  0.415141  0.033248
TEAM_PITCHING_HR  0.430821  0.380597  0.244536  0.666450
TEAM_PITCHING_BB  0.106271  0.286256  0.859253 -0.165822
TEAM_PITCHING_SO -0.085084 -0.234543  0.809055  0.256909
TEAM_FIELDING_E  -0.862046 -0.018696  0.048398 -0.283413
TEAM_FIELDING_DP  0.235546  0.444539  0.073292  0.313000
```

Dựa trên kết quả Factor Loadings ta có thể diễn giải mối quan hệ giữa các biến và các nhân tố. Mỗi cột trong kết quả đại diện cho một nhân tố, và các giá trị tương ứng với mức độ mà mỗi biến gốc đóng góp vào các nhân tố đó. Giá trị tải nhân tố càng cao, biến gốc đó càng liên quan mật thiết đến nhân tố đó.

Nhân tố 1

- TEAM_BATTING_BB (0.815919): Số lần đi bộ của đội có tải trọng cao nhất đối với Nhân tố 1, cho thấy nhân tố này liên quan chặt chẽ đến khả năng đi bộ của đội.
- TEAM_PITCHING_H (-0.800344): Số lần đánh trúng mà đội ném bóng cho phép có tải trọng âm cao đối với Nhân tố 1.
- TEAM_FIELDING_E (-0.862046): Số lỗi phòng thủ của đội có tải trọng âm cao đối với Nhân tố 1.

Nhân tố 1 có thể được diễn giải là một nhân tố liên quan đến khả năng tổng thể của đội trong việc tránh lỗi và kiểm soát số lần đánh trúng, phản ánh cả hiệu suất tấn công và phòng thủ.

Nhân tố 2

- TEAM_BATTING_H (0.855875): Số lần đánh trúng của đội có tải trọng cao nhất đối với Nhân tố 2.
- TEAM_BATTING_2B (0.762211): Số lần đánh đôi của đội cũng có tải trọng cao đối với Nhân tố 2.

Nhân tố 2 có thể được diễn giải là một nhân tố liên quan đến hiệu suất đánh bóng của đội, đặc biệt là các cú đánh trúng và đánh đôi.

Nhân tố 3

- TEAM_PITCHING_BB (0.859253): Số lần đi bộ của đội đối thủ có tải trọng cao nhất đối với Nhân tố 3.
- TEAM_PITCHING_SO (0.809055): Số lần strike out mà đội ném bóng thực hiện có tải trọng cao

Nhân tố 3 có thể được diễn giải là một nhân tố liên quan đến hiệu suất ném bóng, đặc biệt là khả năng kiểm soát của người ném bóng qua các cú đi bộ và strike out. Một đội ném bóng mạnh thường có số lần strike out cao và số lần đi bộ thấp, điều này giúp kiểm soát trận đấu tốt hơn và hạn chế cơ hội ghi điểm của đối thủ.

Nhân tố 4

- TEAM_BATTING_HR (0.688820): Số lần đánh home run của đội có tải trọng cao đối với Nhân tố 4.

- TEAM_BATTING_3B (-0.728201): Số lần đánh ba của đội có tải trọng âm cao đối với Nhân tố 4.
- TEAM_BASERUN_SB (-0.752188): Số lần cướp chốt của đội có tải trọng âm cao đối với Nhân tố 4.

Nhân tố 4 có thể được diễn giải là một nhân tố liên quan đến hiệu suất chạy chốt và các cú đánh home run, với mối quan hệ ngược chiều giữa các cú home run và các lần chạy chốt.

```
communalities = fa.get_communalities()
print("Communalities:", communalities)
```

```
Communalities: [0.87000801 0.66028097 0.74450541 0.87897115 0.82720953
    0.77562057 0.60566477 0.32019125 0.84402511 0.83441401
    0.85904883 0.78282252 0.82613874 0.35643786]
              Column  Communality
0     TEAM_BATTING_H     0.870008
1    TEAM_BATTING_2B     0.660281
2    TEAM_BATTING_3B     0.744505
3    TEAM_BATTING_HR     0.878971
4    TEAM_BATTING_BB     0.827210
5    TEAM_BATTING_SO     0.775621
6    TEAM_BASERUN_SB     0.605665
7    TEAM_BASERUN_CS     0.320191
8    TEAM_PITCHING_H     0.844025
9   TEAM_PITCHING_HR     0.834414
10  TEAM_PITCHING_BB     0.859049
11  TEAM_PITCHING_SO     0.782823
12   TEAM_FIELDING_E     0.826139
13  TEAM_FIELDING_DP     0.356438
```

Các giá trị communalities cho biết tỷ lệ phương sai của từng biến gốc được giải thích bởi các nhân tố trích xuất. Đây là một chỉ số quan trọng trong phân tích nhân tố vì nó giúp đánh giá mức độ mà mỗi biến gốc được đại diện bởi các nhân tố.

Các biến có communalities cao (>0.7): Các biến này được giải thích tốt bởi các nhân tố trích xuất. Ví dụ, các biến như TEAM_BATTING_H, TEAM_BATTING_HR, TEAM_BATTING_BB, TEAM_PITCHING_H, và TEAM_PITCHING_HR có communalities rất cao, cho thấy rằng các nhân tố đã trích xuất giải thích tốt các phương sai của những biến này.

Các biến có communalities trung bình (0.5-0.7): Các biến này được giải thích ở mức độ trung bình. Ví dụ, TEAM_BATTING_2B và TEAM_BASERUN_SB.

Các biến có communalities thấp (<0.5): Các biến này không được giải thích tốt bởi các nhân tố trích xuất. Ví dụ, TEAM_BASERUN_CS và TEAM_FIELDING_DP có communalities khá thấp, cho thấy các nhân tố trích xuất không giải thích tốt phương sai của những biến này.

6.1.4 Nhận xét - Đánh giá chung

Kiểm định Bartlett cho thấy giá trị p-value bằng 0, điều này chứng tỏ rằng ma trận tương quan của bộ dữ liệu không phải là ma trận đơn vị và các biến có mối liên quan đáng kể với nhau. Bộ dữ liệu phù hợp cho phân tích nhân tố.

Giá trị KMO là 0.649, cho thấy rằng tỷ lệ của các hệ số tương quan quan sát được so với các hệ số tương quan riêng là đủ cao để phân tích nhân tố là thích hợp.

Số lượng nhân tố được xác định là 4 dựa trên biểu đồ scree và các giá trị riêng.

Các hệ số tải nhân tố cho thấy mối quan hệ giữa các biến và các nhân tố, giúp diễn giải ý nghĩa của từng nhân tố.

Các communalities cho thấy mức độ các biến được giải thích bởi các nhân tố, giúp đánh giá mức độ tin cậy của phân tích.

6.1.5 Hướng mở rộng

1. Nghiên cứu sâu hơn về từng nhân tố: Sau khi xác định được các nhân tố, có thể nghiên cứu sâu hơn về mối quan hệ giữa từng nhân tố và các biến gốc, từ đó tìm ra những yếu tố ảnh hưởng chính đến hiệu suất thi đấu của đội bóng.

2. Áp dụng vào các mô hình dự đoán: Sử dụng các nhân tố trích xuất để xây dựng các mô hình dự báo như mô hình hồi quy, phân cụm hoặc các mô hình dự đoán khác để đánh giá và dự đoán hiệu suất của đội bóng ví dụ như dự đoán số trận thắng của đội dựa trên các yếu tố chính đã được xác định.

3. So sánh với các phương pháp khác: Thử nghiệm và so sánh kết quả của phân tích nhân tố với các phương pháp khác như phân tích thành phần chính (PCA) để xác định phương pháp nào phù hợp nhất với dữ liệu.

4. Thử nghiệm phân tích trên một tập dữ liệu khác

6.2 Ví dụ khác

Ví dụ về cách sử dụng phân tích nhân tố để phân tích tương quan và giải quyết vấn đề đa cộng tuyến

Giả sử ta đang xây dựng một mô hình hồi quy để dự đoán giá nhà dựa trên các đặc điểm của ngôi nhà. Ta có một tập dữ liệu với các biến sau:

1. Diện tích nhà (Square Footage)

2. Số phòng ngủ (Number of Bedrooms)

3. Số phòng tắm (Number of Bathrooms)

4. Diện tích sân vườn (Yard Size)

5. Diện tích nhà để xe (Garage Size)
6. Tổng diện tích đất (Total Lot Size)

Đầu tiên, chúng ta sẽ tạo một tập dữ liệu mẫu để làm việc

```
# Thiết lập seed để kết quả có tính nhất quán
np.random.seed(0)

# Tạo dữ liệu có tương quan
n_samples = 1000
square_footage = np.random.normal(loc=2000, scale=500, size=n_samples)
number_of_bedrooms = square_footage / 1000 +
        np.random.normal(loc=2, scale=0.5, size=n_samples)
number_of_bathrooms = square_footage / 1000 +
        np.random.normal(loc=1, scale=0.5, size=n_samples)
yard_size = np.random.normal(loc=500, scale=200, size=n_samples)
garage_size = yard_size / 500 +
        np.random.normal(loc=2, scale=0.5, size=n_samples)
total_lot_size = yard_size +
        np.random.normal(loc=200, scale=50, size=n_samples)

# Tạo DataFrame
data = pd.DataFrame({
    'Square_Footage': square_footage,
    'Number_of_Bedrooms': number_of_bedrooms,
    'Number_of_Bathrooms': number_of_bathrooms,
    'Yard_Size': yard_size,
    'Garage_Size': garage_size,
    'Total_Lot_Size': total_lot_size
})

# Tạo biến phụ thuộc giá nhà
house_price = square_footage * 0.5 + number_of_bedrooms *
        10000 + number_of_bathrooms * 5000 + \
        yard_size * 0.3 + garage_size * 2000 + total_lot_size *
        0.2 + np.random.normal(loc=0, scale=10000, size=n_samples)
data['House_Price'] = house_price

data.head()
```

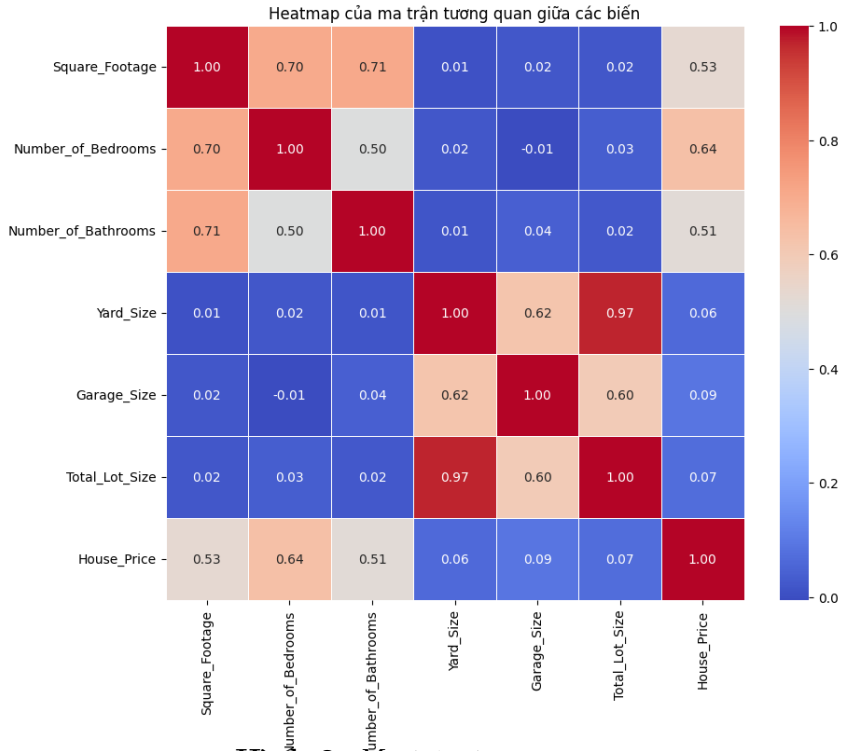
	Square_Footage	Number_of_Bedrooms	Number_of_Bathrooms	Yard_Size
0	2882.026173	5.160008	3.115566	818.654723
1	2200.078604	4.646316	2.344094	613.744480
2	2489.368992	4.278212	3.512437	477.102591
3	3120.446600	5.172804	3.641259	550.326050
4	2933.778995	5.047806	3.893373	257.828872

```
   Garage_Size  Total_Lot_Size   House_Price
0     3.943984     1034.140914  97384.671444
1     4.149339      776.871670  58727.226124
2     3.089751      600.256597  69178.366638
3     3.668876      722.213308  77767.611506
4     1.646492      377.853316  88512.984211
```

Khi kiểm tra ma trận tương quan giữa các biến này, chúng ta nhận thấy một số biến có mức độ tương quan cao, cho thấy sự tồn tại của đa cộng tuyến.

```
# Tính ma trận tương quan
correlation_matrix = data.corr()

# Vẽ heatmap của ma trận tương quan
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f',
    linewidths=.5)
plt.title('Heatmap của ma trận tương quan giữa các biến')
plt.show()
```



Hình 3: *Ma trận tương quan*

Có thể thấy một số điểm quan trọng về mối tương quan giữa các biến:

1. Mối quan hệ mạnh giữa các biến liên quan đến kích thước của ngôi nhà: Square_Footage và Number_of_Bedrooms, Square_Footage và Number_of_Bathrooms.

2. Mối quan hệ mạnh giữa các biến liên quan đến đất: Yard_Size và Garage_Size, Yard_Size và Total_Lot_Size, Garage_Size và Total_Lot_Size.

3. Tương quan giữa giá nhà và các yếu tố khác: House_Price và Square_Footage, House_Price và Number_of_Bedrooms.

Hành động tiếp theo

Sử dụng phân tích nhân tố: Để hiểu rõ hơn về cấu trúc tương quan này, ta tiến hành phân tích nhân tố và những tương quan mạnh này cho thấy rằng sử dụng phân tích nhân tố có thể giúp giảm đa cộng tuyến bằng cách tạo ra các nhân tố độc lập thay vì sử dụng trực tiếp các biến có tương quan cao này.

Đầu tiên ta sẽ kiểm tra độ thích hợp của phân tích nhân tố:

```
# Thực hiện Bartlett's Test
chi_square_value, p_value = calculate_bartlett_sphericity(data)

print("Bartlett's Test of Sphericity:")
print(f"Chi-square Value: {chi_square_value}")
print(f"P-value: {p_value}")

# Kiểm định KMO
kmo_all, kmo_model=calculate_kmo(data)
print("KMO model:", kmo_model)
```

```
Bartlett's Test of Sphericity:
Chi-square Value: 5384.184669730112
P-value: 0.0
KMO model: 0.676357167276583
```

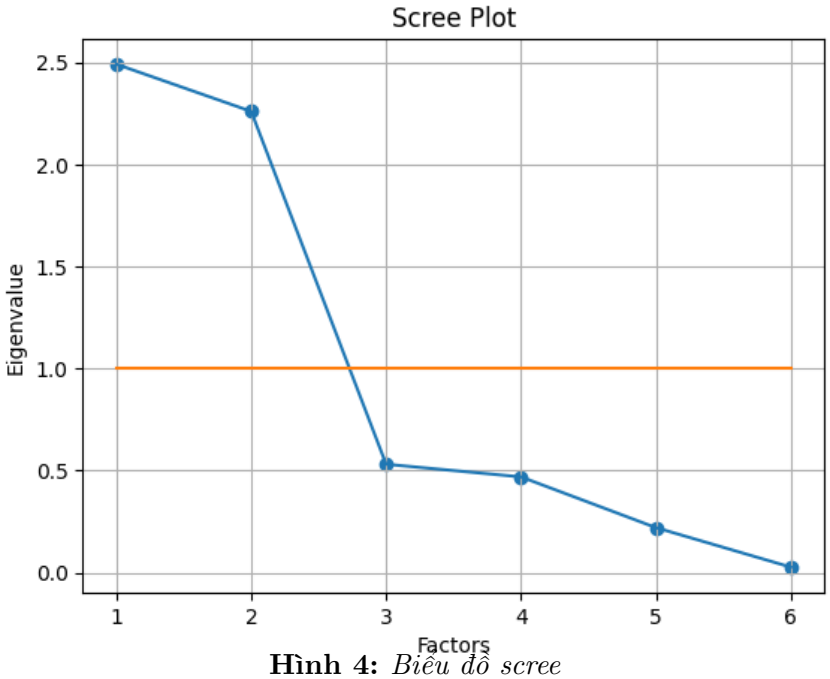
Với KMO Model đạt được là 0.68, điều này cho thấy dữ liệu của có mức độ phù hợp tương đối để thực hiện phân tích nhân tố.

```
# Khởi tạo Phân tích Nhân Tố
fa = FactorAnalyzer(10, rotation=None)
fa.fit(data.drop('House_Price', axis=1))

# Cài đặt của Phân tích Nhân Tố
FactorAnalyzer(bounds=(0.005, 1), impute='median', is_corr_matrix=False,
               method='principal', n_factors=3, rotation=None,
               rotation_kwargs={}, use_smc=True)

# Lấy Giá Trị Riêng
ev = fa.get_eigenvalues()
print("Eigenvalues:", ev)
```

Dựa vào biểu đồ scree và các giá trị riêng ta chọn giải pháp có 2 nhân tố.



Hình 4: *Biểu đồ scree*

```
# Thực hiện phân tích nhân tố
fa = FactorAnalyzer(n_factors=2, rotation='varimax', method='principal')
fa.fit(data.drop('House_Price', axis=1))

# Lấy ma trận tải trọng
loadings = fa.loadings_

# Tạo DataFrame cho ma trận tải trọng
loading_df = pd.DataFrame(loadings, columns=['Factor1', 'Factor2'],
        index=data.drop('House_Price', axis=1).columns)
print("Ma trận hệ số tải nhân tố:")
print(loading_df)
```

```
Ma trận hệ số tải nhân tố:
                     Factor1   Factor2
Square_Footage       0.202717  0.906539
Number_of_Bedrooms   0.185436  0.818596
Number_of_Bathrooms  0.195091  0.818061
Yard_Size            0.943461 -0.201036
Garage_Size          0.778413 -0.157917
Total_Lot_Size       0.938851 -0.184614
```

Phân tích tương quan:

Ta có thể thấy Factor1 có hệ số tải nhân tố cao từ:

- Yard_Size (0.943461)

- Garage_Size (0.778413)
- Total_Lot_Size (0.938851)

Các biến này chủ yếu liên quan đến diện tích đất và các khu vực phụ, vì vậy chúng ta có thể đặt tên cho Factor1 là Land_Aux_Areas_Factor.

Factor2 có tải trọng cao từ:

- Square_Footage (0.906539)
- Number_of_Bedrooms (0.818596)
- Number_of_Bathrooms (0.818061)

Các biến này chủ yếu liên quan đến kích thước ngôi nhà và các phòng, vì vậy chúng ta có thể đặt tên cho Factor2 là House_Size_Factor.

Và ta đã tạo ra hai biến mới từ các nhân tố này:

- Kích thước ngôi nhà và các phòng (House Size Factor)
- Diện tích đất và các khu vực phụ (Land and Auxiliary Areas Factor)

Các biến này sẽ thay thế cho các biến gốc trong mô hình của ta. Điều này giúp loại bỏ vấn đề đa cộng tuyến và làm cho mô hình dễ quản lý và ổn định hơn.

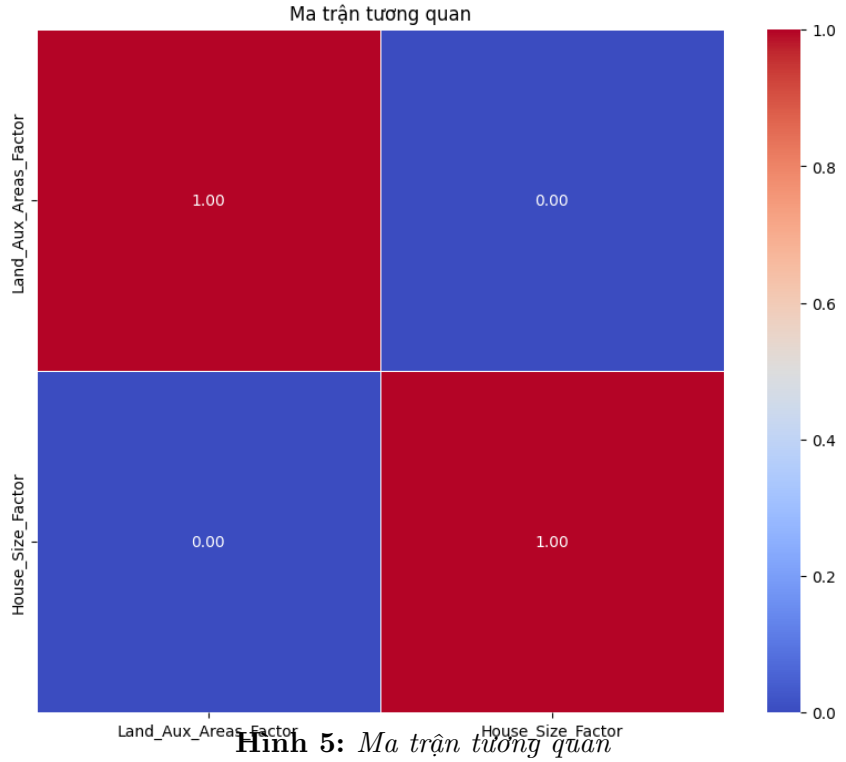
```
# Trích xuất nhân tố
factor_scores = fa.transform(data.drop('House_Price', axis=1))
factor_df = pd.DataFrame(factor_scores, columns=['Land_Aux_Areas_Factor'
        , 'House_Size_Factor'])
print(factor_df)
```

```
     Land_Aux_Areas_Factor  House_Size_Factor
0                 1.961487           1.065385
1                 0.964345          -0.005550
2                 0.013237           0.909982
3                 0.865975           1.821774
4                -1.305182           2.231472
..                     ...                ...
995              -0.385602           1.161613
996               1.875739          -0.435208
997               0.190172           0.094203
998               1.228344          -2.014889
999              -0.747734          -0.512898

[1000 rows x 2 columns]
```

Ta vẽ lại ma trận tương quan của 2 biến mới

```
# Vẽ biểu đồ ma trận tương quan
plt.figure(figsize=(10, 8))
sns.heatmap(factor_df.corr(), annot=True, cmap='coolwarm', fmt='.2f',
        linewidths=.5)
plt.title('Ma trận tương quan')
plt.show()
```



Hình 5: *Ma trận tương quan*

Ta có thể thấy rằng vấn đề đa cộng tuyến trong dữ liệu đã được giải quyết. Tiếp theo ta có thể sử dụng các nhân tố này để xây dựng các mô hình như mô hình hồi quy và đánh giá hiệu quả của mô hình thông qua các chỉ số như R-squared, AIC, BIC và các thống kê khác.

Kết luận

Bằng cách sử dụng phân tích nhân tố, ta có thể nhóm các biến có mức độ tương quan cao thành các nhân tố chung. Điều này giúp ta nhìn rõ hơn mối quan hệ giữa các biến và nhận diện được các nhóm biến có liên quan với nhau.

Bên cạnh đó ta đã giảm số lượng biến trong mô hình từ 6 biến gốc có khả năng đa cộng tuyến xuống còn 2 nhân tố chính. Điều này giúp cải thiện tính ổn định và độ tin cậy của các ước lượng trong mô hình xây dựng, đồng thời giúp cho việc diễn giải kết quả dễ dàng hơn. Phân tích nhân tố đã giúp ta tối ưu hóa các biến cho mô hình hồi quy, loại bỏ sự ảnh hưởng tiêu cực của đa cộng tuyến, và làm cho mô hình trở nên dễ dàng quản lý hơn.

7 Tài liệu tham khảo

1. Tryfos, Peter. *Methods for Business Analysis and Forecasting: Text and Cases*. Product Bundle, 26 Jan. 1998.

2. Johnson, Richard A., and Dean W. Wichern. *Applied Multivariate Statistical Analysis*

3. DeCoster, J. (1998). *Overview of factor analysis*. Retrieved March 22, 2012 from http://www.stat-help.com/notes.html

4. Matsunaga, Masaki. "How to Factor-Analyze Your Data Right: Do's, Don'ts, and How-To's". *International Journal of Psychological Research*, vol. 3, no. 1, 2010, pp. 97-110. Universidad de San Buenaventura, Medellín, Colombia.

5. An Gie Yong and Sean Pearce. "A Beginner's Guide to Factor Analysis: Focusing on Exploratory Factor Analysis". *Tutorials in Quantitative Methods for Psychology*, 2013, Vol. 9(2), pp. 79-94.

6. Schneider, W. Joel. "A Gentle, Non-Technical Introduction to Factor Analysis".*Psychometrics, Statistics, Turorial*, January 13, 2014.

7. Donatello, Robin, và Edward Roualdes. *Applied Statistics*. California State University, Chico.

8. Abdi, Hervé. "Factor Rotations in Factor Analyses". The University of Texas at Dallas.

9. Đỗ Tiến Sỹ, Trần Nguyễn Nhật Nam. (Tháng 9 năm 2019). "Factor analysis (FA) with IBM SPSS". *Ứng dụng tin học hỗ trợ nghiên cứu khoa học và hoạt động xây dựng*. TPHCM.

10. https://online.stat.psu.edu/stat505/lesson/12/12.12