TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**MÔ HÌNH ĐỒ THỊ CÓ XÁC SUẤT**

**TÌM HIỂU VỀ THƯ VIỆN PGMPY VÀ ỨNG DỤNG**

*Người hướng dẫn*: **TS NGUYỄN CHÍ THIỆN**

*Người thực hiện*: **ĐÀO VIỆT DUY – MSSV: 51603070**

**NGUYỄN DUY HÀN LÂM – MSSV: 196005004**

Khoá  **: 2019**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**MÔ HÌNH ĐỒ THỊ CÓ XÁC SUẤT**

**TÌM HIỂU VỀ THƯ VIỆN PGMPY VÀ ỨNG DỤNG**

*Người hướng dẫn*: **TS NGUYỄN CHÍ THIỆN**

*Người thực hiện*: **ĐÀO VIỆT DUY – MSSV: 51603070**

**NGUYỄN DUY HÀN LÂM – MSSV: 196005004**

Khoá  **: 2019**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020**

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của TS Nguyễn Chí Thiện;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Đào Việt Duy*

*Nguyễn Duy Hàn Lâm*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

Tài liệu này xin trình bày về thư viện pgmpy của Python trong nghiên cứu về lĩnh vực mô hình đồ thị có xác suất và xây dựng ứng dụng dựa trên thư viện này.

MỤC LỤC

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN ii](#_Toc57995834)

[TÓM TẮT iii](#_Toc57995835)

[MỤC LỤC 1](#_Toc57995836)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 3](#_Toc57995837)

[CHƯƠNG 1: MÔ HÌNH ĐỒ THỊ XÁC SUẤT 6](#_Toc57995838)

[1.1 Tổng quan về mô hình đồ thị có xác suất 6](#_Toc57995839)

[1.1.1 Đồ thị xác suất 6](#_Toc57995840)

[1.1.2 Mô hình đồ thị vô hướng 9](#_Toc57995841)

[1.1.3 Một số giải thuật học 10](#_Toc57995842)

[1.1.3.1 Maximum likelihood estimator 10](#_Toc57995843)

[1.1.3.2 Bayesian estimator 11](#_Toc57995844)

[1.1.4 Suy luận (Inference) 12](#_Toc57995845)

[1.1.4.1 Suy luận biên (Marginal inference) 13](#_Toc57995846)

[1.1.4.2 Suy luận Maximum A Posteriori (MAP inference) 16](#_Toc57995847)

[1.1.4.3 Suy luận Bayes (Bayesian inference) 17](#_Toc57995848)

[1.2 Tạo và định nghĩa mô hình 17](#_Toc57995849)

[1.2.1 Xây dựng mô hình khi biết cấu trúc 18](#_Toc57995850)

[1.2.2 Xây dựng mô hình khi chưa biết cấu trúc 21](#_Toc57995851)

[1.3 Suy luận thống kê bằng pgmpy 22](#_Toc57995852)

[1.4 Thực hiện các giải thuật học bằng pgmpy 25](#_Toc57995853)

[1.4.1 Parameter 25](#_Toc57995854)

[1.4.2 Structure 28](#_Toc57995855)

[1.5 Tóm tắt chương 1 31](#_Toc57995856)

[CHƯƠNG 2: ỨNG DỤNG PHÂN TÍCH NHIỄM BỆNH 32](#_Toc57995857)

[2.1 Bayesian 32](#_Toc57995858)

[2.1.1 Giới thiệu 32](#_Toc57995859)

[2.1.2 Học 34](#_Toc57995860)

[2.1.3 suy luận xác suất 35](#_Toc57995861)

[2.2 Phân tích đề tài 38](#_Toc57995862)

[2.2 Xây dựng chương trình 39](#_Toc57995863)

[2.4 Kết quả học mô hình 43](#_Toc57995864)

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH**

[Hình 1.1. Mô hình đồ thị biểu thị mối tương quan giữa R, W1, W2 6](#_Toc57995870)

[Hình 1.2. Ví dụ về mô hình đồ thị 7](#_Toc57995871)

[Hình 1.3. Ví dụ về mô hình đồ thị 8](#_Toc57995872)

[Bảng 1.1. Bảng biểu thị xác suất giữa W1 và R 8](#_Toc57995873)

[Bảng 1.2. Bảng biểu thị xác suất giữa W2 và R 8](#_Toc57995874)

[Hình 1.4. Ví dụ về đồ thị vô hướng 9](#_Toc57995875)

[Hình 1.1. Mô hình ví dụ cho variable elimination [0] 15](#_Toc57995876)

[Hình 1.5. Mô hình đồ thị biểu diễn các mối tương quan của mô hình cancer [1] 19](#_Toc57995877)

[Hình 1.6. Tạo mô hình cancer bằng pgmpy [1] 19](#_Toc57995878)

[Hình 1.7. Khởi tạo các giá trị xác suất của mô hình 1 [1] 20](#_Toc57995879)

[Hình 1.8. Thêm các giá trị xác suất vừa khởi tạo vào mô hình 1 [1] 20](#_Toc57995880)

[Hình 1.9. Xuất các mối quan hệ độc lập của mô hình 1 [1] 21](#_Toc57995881)

[Hình 1.10. Mô hình “asia” [2] 22](#_Toc57995882)

[Hình 1.11. Thực hiện lấy mô hình “asia” [3] 23](#_Toc57995883)

[Hình 1.12. In các giá trị đỉnh, cạnh và xác suất quan hệ của mô hình “asia” [3] 23](#_Toc57995884)

[Hình 1.13. Thực hiện loại bỏ các biến của mô hình “asia” [3] 23](#_Toc57995885)

[Hình 1.14. Tiến hành in các giá trị xác suất quan hệ mong muốn [3] 24](#_Toc57995886)

[Hình 1.15. Các bảng thống kê các giá trị xác suất quan hệ mong muốn [3] 24](#_Toc57995887)

[Hình 1.16. Kiểm tra xem các thuộc tính có bị ảnh hưởng bởi các thuộc tính khác hay không [3] 25](#_Toc57995888)

[Hình 1.17. Kết quả sau khi chạy câu lệnh hình 1.16 [3] 25](#_Toc57995889)

[Hình 1.18. Mô hình “alarm” [4] 26](#_Toc57995890)

[Hình 1.19. Lấy và tạo mô hình “alarm” [5] 26](#_Toc57995891)

[Hình 1.20. Tạo mô hình “alarm” bằng các cặp cạnh đã được định nghĩa [5] 27](#_Toc57995892)

[Hình 1.21. Tính toán các giá trị xác suất cho mô hình bằng MaximumLikelihoodEstimator [5] 27](#_Toc57995893)

[Hình 1.22. Tính toán các giá trị xác suất cho mô hình bằng BayesianEstimator [5] 28](#_Toc57995894)

[Hình 1.23. Chuẩn bị các thư viện cho việc học cấu trúc [6] 29](#_Toc57995895)

[Hình 1.24. Lấy dữ liệu mô hình alarm [6] 29](#_Toc57995896)

[Hình 1.25. Xây dựng mô hình bằng thuật toán PC biến thể stable [6] 29](#_Toc57995897)

[Hình 1.26. Xây dựng mô hình bằng thuật toán PC biến thể orig [6] 29](#_Toc57995898)

[Hình 1.27. Xây dựng hàm score để đánh giá thuật toán [6] 30](#_Toc57995899)

[Hình 1.28. Kết quả đánh giá thuật toán PC biến thể “stable” [6] 30](#_Toc57995900)

[Hình 1.29. Kết quả đánh giá thuật toán PC biến thể “orig” [6] 30](#_Toc57995901)

[Hình 1.30. Tiến hành xây dựng học cấu trúc cho mô hình bằng Hill-climb [6] 31](#_Toc57995902)

[Hình 2.1. Minh họa mô hình mạng Bayesian 33](#_Toc57995903)

[Hình 2.2. Minh họa inference suy luận xác suất Bayesian 36](#_Toc57995904)

[Hình 2.3. Minh họa inference suy luận xác suất Bayesian 37](#_Toc57995905)

[Hình 2.4. Minh họa inference suy luận xác suất Bayesian 38](#_Toc57995906)

[Hình 2.5. Chuẩn bị thư viện cho chương trình 39](#_Toc57995907)

[Hình 2.6. Tiến hành nạp dữ liệu vào chương trình 40](#_Toc57995908)

[Hình 2.7. Lọc ra các giá trị riêng biệt cho từng loại bệnh, symptom 40](#_Toc57995909)

[Hình 2.8. Xây dựng hàm tạo các cặp cạnh cho mô hình 41](#_Toc57995910)

[Hình 2.9. Tiến hành xây dựng mô hình 41](#_Toc57995911)

[Hình 2.10. Tiến hành thống kê symptom và disease 41](#_Toc57995912)

[Hình 2.11. Xây dựng hàm chuyển đổi sang dataframe 42](#_Toc57995913)

[Hình 2.12. Tính xác suất cho Symptom\_1 bằng MaximumLikelihooddEstimator 42](#_Toc57995914)

[Hình 2.13. Tính xác suất cho Disease bằng MaximumLikelihooddEstimator 42](#_Toc57995915)

[Hình 2.14. Tính xác suất cho Disease bằng BayesianEstimator 42](#_Toc57995916)

[Hình 2.15. Xuất kết quả tính xác suất của Disease bằng MaximumLikelihooddEstimator ra file CSV 42](#_Toc57995917)

[Hình 2.16. Xuất kết quả tính xác suất của Disease bằng BayesianEstimator ra file CSV 43](#_Toc57995918)

[Hình 2.17. Thống kê các triệu chứng cho các bệnh khác nhau 43](#_Toc57995919)

[Hình 2.18. Thống kê số lần xuất hiện các triệu chứng của Symptom\_1 44](#_Toc57995920)

[Hình 2.19. Bảng thống kê xác suất giữa symptom và disease 45](#_Toc57995921)

CHƯƠNG 1: MÔ HÌNH ĐỒ THỊ XÁC SUẤT

Chương này, tài liệu sẽ giới thiệu các lý thuyết về mô hình đồ thị có xác suất và đồng thời hướng dẫn ta xây dựng các mô hình này bằng phương pháp lập trình dựa trên thư viện pgmpy của python.

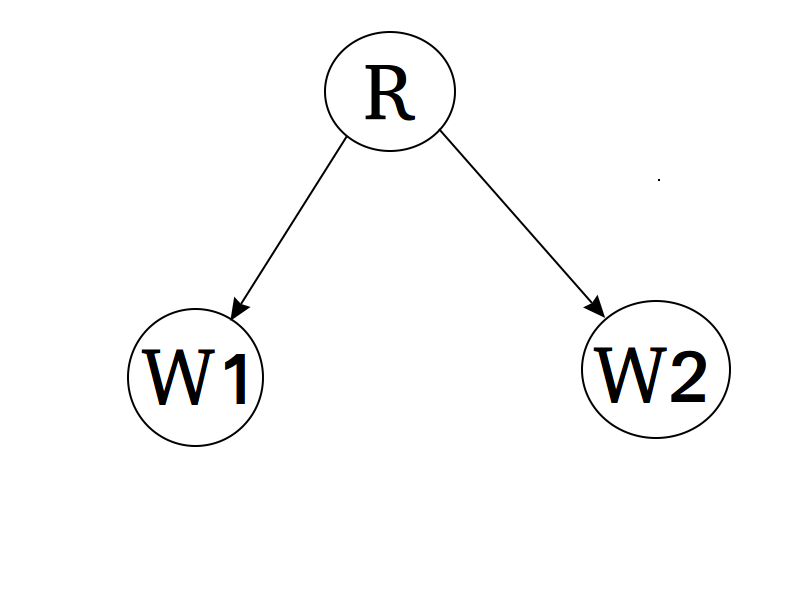
* 1. Tổng quan về mô hình đồ thị có xác suất

Khi đưa vào huấn luyện mạng Neural cho các mô hình lớn, bộ dữ liệu đầu vào thường rất lớn và quan hệ giữa các biến trong bộ dữ liệu rất phức tạp. Đến một lúc nào đó, dữ liệu không thể chỉ đơn thuần xuất ra bảng để phân tích vì có quá nhiều loại. Với mô hình đồ thị xác suất, ta có thể hiển thị thông tin dưới dạng gói gọn, xúc tích hơn trong kiểu đồ thị, thể hiện thông tin quan hệ các biến ngẫu nhiên phụ thuộc vào các biến khác.

1.1.1 Đồ thị xác suất

Mọi mô hình đồ thị xác suất (Probabilistic Graphical Model – PGM) bao gồm hai thành phần: các nút (node), hay gọi là biến ngẫu nhiên và các cạnh phụ thuộc thống kê giữa các nút với nhau. Đôi khi tất cả các biến đó có thể có sẵn để quan sát, trong khi đôi khi chỉ có tập hợp con của chúng.

Ví dụ ta có ba biến nhị phân (có/không) biểu thị sự hiện diện của mưa (R), nước trên cửa sổ (W1) và nước trên đường (W2). Nếu không có bất kỳ thông tin bổ sung nào, chúng ta biết rằng có mối quan hệ nhân quả giữa mưa và hai biến số khác:



1. Mô hình đồ thị biểu thị mối tương quan giữa R, W1, W2

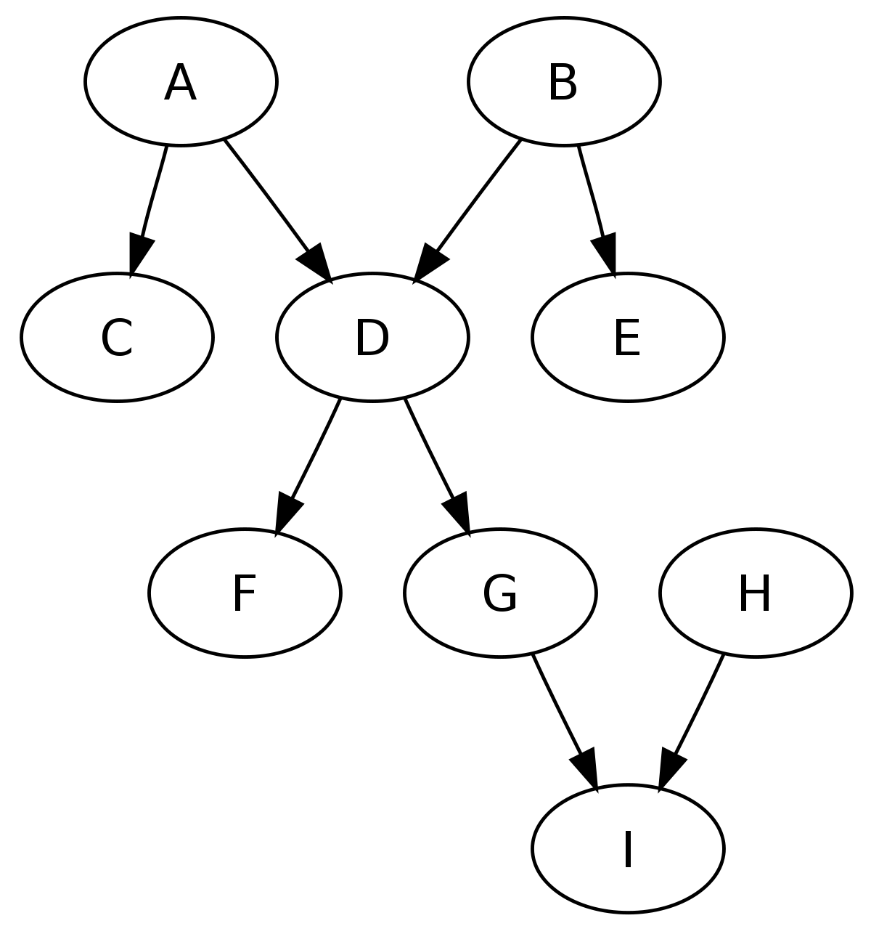
Chỉ có mưa mới có thể làm ướt cửa sổ và đường xá, nhưng ngược lại thì không. Ngoài ra, không có chu kỳ. Đây là mô hình đồ thị có hướng không có chu kỳ (Directed acyclic graph – DAG), loại PGM phổ biến nhất.

Nếu chúng ta quan sát thấy rằng khi R = Yes, chúng ta biết rằng xác suất W1 = Yes và W2 = Yes cao. Khi R = No, xác suất W1 = Yes và W2 = Yes sẽ rất thấp.

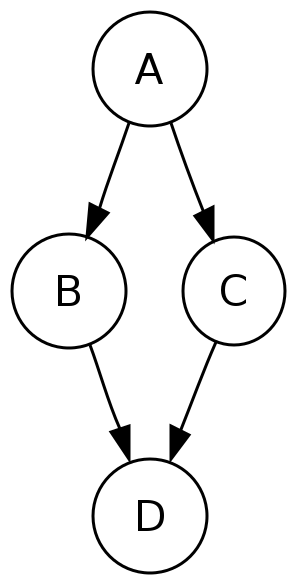
Khi chúng ta biết giá trị của R, xác suất cửa sổ ướt và đường ướt trở nên độc lập. Chúng có thể cao hoặc thấp, nhưng những sự kiện này không ảnh hưởng lẫn nhau nữa.

Ngoài ra, đây là Singly Connected DAG (SC-DAG), nghĩa là chỉ có một đường đi duy nhất, vô hướng giữa bất kỳ cặp nút nào.

Một số ví dụ khác:



1. Ví dụ về mô hình đồ thị



1. Ví dụ về mô hình đồ thị

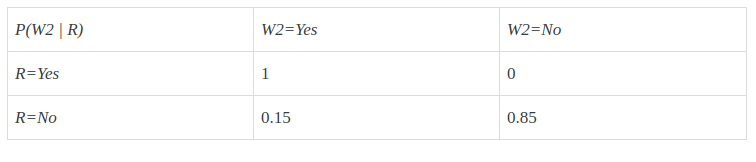
Ta có thể biểu diễn mô hình (hình 1.1) dưới dạng bảng xác suất có điều kiện (CPT), trong đó mỗi bảng biểu thị xác suất có điều kiện của việc quan sát một biến cho trước một biến khác.

Xác suất cửa sổ bị ướt khi mưa:

|  |  |  |
| --- | --- | --- |
| P(W1| R) | W1 = Yes | W1 = No |
| R = Yes | 0.9 | 0.1 |
| R = No | 0.03 | 0.97 |

1. Bảng biểu thị xác suất giữa W1 và R

Xác suất đường bị ướt khi mưa:



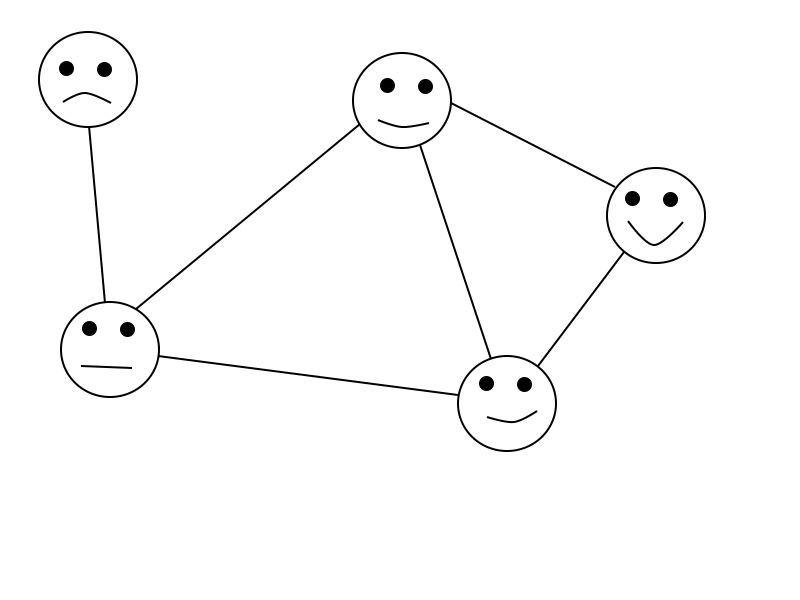
1. Bảng biểu thị xác suất giữa W2 và R

Bayes hay Bayesian network (BN) là những trường hợp đặc biệt của DAG. Chúng có thể được kết nối đơn lẻ hoặc nhiều lần và sử dụng logic Bayes để giải thích các xác suất. Nói tóm lại, điều đó có nghĩa là họ bắt đầu với một số giả định trước về sự phụ thuộc và sử dụng quy tắc Bayes để cập nhật niềm tin của họ về dữ liệu. Tương tự, hầu hết các mô hình DAG thực sự là Bayesian.

Mô hình Markov ẩn (Hidden Markov Model – HMM) và Mô hình không gian-trạng thái Gaussian tuyến tính (Gaussian State Space – GSS) cũng là những loại SC-DAG đặc biệt.

1.1.2 Mô hình đồ thị vô hướng

Còn được gọi là Trường ngẫu nhiên Markov (Markov Random Field – MRF). Giống như các mô hình có hướng, chúng thể hiện sự phụ thuộc có điều kiện giữa các biến ngẫu nhiên, nhưng không bao hàm quan hệ nhân quả có thứ tự. Ví dụ một mô hình đơn giản: Điểm hạnh phúc giữa mọi người.



1. Ví dụ về đồ thị vô hướng

Mức độ hạnh phúc của mỗi người trong ví dụ này ảnh hưởng đến điểm số của những người khác. Khi chúng ta quan sát một trong số họ, nó cũng cung cấp một số thông tin về hạnh phúc của những người còn lại, vì tất cả chúng ta đều được kết nối với nhau.

Trong số các loại phổ biến khác của mô hình đồ thị vô hướng là Trường ngẫu nhiên có điều kiện (Conditional Random Fields) và Restricted Boltzmann Machines.

Nhân tiện, DAG cung cấp quy trình suy luận đơn giản hơn và dễ giải thích hơn, trong khi các mô hình vô hướng có nhiều tiềm năng hơn để mô tả các mối quan hệ phức tạp.

Factor Graph cũng rất giống với MRF. Hơn nữa, các mô hình nhân tố truyền thống có thể được mở rộng để bao gồm cả mô hình Được hướng và Không có hướng. Trên thực tế, có rất nhiều nghiên cứu tương tự được thực hiện với mục tiêu tìm ra các kết nối sâu hơn giữa PGM, NN và các công cụ, thuật toán tính xác suất khác.

1.1.3 Một số giải thuật học

1.1.3.1 Maximum likelihood estimator

Ta muốn dự đoán một giá trị xác suất P\* dựa trên một giá trị xác suất P cho trước, ta sẽ sử dụng hàm phân kỳ Kullback-Leibler:

Ta có entropy được tính bằng:

Giá trị kì vọng được tính bằng:

Thế (2) và (3) vào (1) ta được:

Ta muốn dự đoán từ P nên ta cần hội tụ hàm này từ P về . Mà KL hội tụ khi nó đạt giá trị nhỏ nhất.

Do entropy không phụ thuộc vào P và ta thấy KL và trái dấu nên ta được:

Giả định rằng kỳ vọng được tính bằng phương pháp Monte-Carlo, ta được:

Thế (5) vào (4) ta được:

Hàm lỗi của giải thuật học sẽ là:

1.1.3.2 Bayesian estimator

Giải thuật học Bayesian estimator chính là quá trình ta sẽ làm giảm thiểu giá trị mất mát của hậu xác suất giữa tham số và dữ liệu x trong quá trình suy luận Bayes (được đề cập cụ thể hơn ở mục 1.1.4.3):

Nếu các giá trị dự đoán là a thì ta có là hàm mất mát khi dự đoán các giá trị a.

Ta có hàm tính kỳ vọng của các giá trị mất mát hậu nghiệm (posterior) là:

Quá trình học Bayesian estimation chính là quá trình làm tối thiểu hóa mất mát tính được từ .

1.1.4 Suy luận (Inference)

Nếu đã quan sát một số biến, phân phối nào sẽ có các biến khác? Phương pháp phổ biến là sử dụng quy tắc Bayes, P (A cho trước B = b) = P (A và B = b) / P (B = b). Ý tưởng là: để có được xác suất của sự kiện A cho rằng B đã nhận giá trị b, bạn cần phải lấy xác suất cả A và B = b đã xảy ra và mở rộng quy mô nó bằng xác suất B = b, vì bạn đã biết rằng nó đã xảy ra.

Tương tự như vậy, bằng cách tính toán các xác suất chung như A và B và các phân phối biên như B = b từ dữ liệu đào tạo, bạn có thể suy ra CPT. Khi một số biến bị ẩn, bạn có thể ước tính phân phối của chúng từ dữ liệu quan sát bằng phương pháp Tối đa hóa kỳ vọng (EM) hoặc Markov Chain Monte Carlo (MCMC).

Hơn nữa, các phép tính suy luận trong PGM có thể được thực hiện cục bộ. Thông tin về ảnh hưởng mà các nút gửi cho nhau thường được gọi là thông điệp. Chúng thường được chia theo loại Pi (mang xác suất trước) và Lambda (mang xác suất có thể xảy ra). Thuật toán phổ biến để trao đổi các thông điệp đó là Truyền bá niềm tin (BP), đây cũng là sự tổng quát của thuật toán Chuyển tiếp-Quay lại cho HMM và thuật toán Làm mịn Kalman cho GSS. Trong trường hợp đồ thị có chu kỳ hoặc vô hướng, nó được gọi là Loopy BP, vì những thông báo đó có thể đi theo các vòng gần nhau.

Phần lớn tài liệu trong các mô hình đồ thị có xác suất tập trung vào suy luận. Lý do là vì:

Suy luận xác suất có thể đưa ra dự đoán từ những gì chúng ta đã biết.

Suy luận khó về mặt tính toán. Trong một số loại đồ thị cụ thể, chúng ta có thể thực hiện suy luận khá hiệu quả, nhưng trên đồ thị chung, nó không thể thực hiện được. Vì vậy, chúng ta cần sử dụng các thuật toán gần đúng đánh đổi độ chính xác để lấy hiệu quả.

Có một số câu hỏi mà chúng ta có thể trả lời bằng suy luận:

Suy luận biên (marginal inference) được sử dụng để tìm phân phối xác suất của một biến (một thuộc tính) cụ thể mà ta đang quan tâm hay mong muốn.

Suy luận hậu nghiệm (bayesian inference): Cho một số biến quan sát E (E làm bằng chứng) nhận giá trị e, tìm phân phối hậu nghiệm p(H|E = e) cho một số biến ẩn giả thuyết (hypothesis) H.

Suy luận Maximum-A-Posteriori (MAP): Cho một số biến quan sát E nhận giá trị e, tiến hành tối đa hóa hậu xác suất của H.

Ta sử dụng các suy luận này để trả lời các câu hỏi mà ta mong muốn hay ta quan tâm hoặc thực hiện các nhiệm vụ mà ta được giao. Ví dụ như nếu ta quan tâm đến một thuộc tính nào đó trong đồ thị ta sẽ sử dụng suy luận biên, nếu ta muốn cập nhật các giá trị xác suất xảy ra trước các giá trị quan sát được ta sẽ dùng bayesian inference,…

1.1.4.1 Suy luận biên (Marginal inference)

Như ta đã biết xác suất Bayesian của một mô hình được định nghĩa như sau:

Để biết xác suất của một thuộc tính hay một biến nào đó trong mô hình, ta sẽ sử dụng xác suất biên để tính:

Triển khai (6) ta được:

Ở biểu thức (7) ta có thể thấy rằng ở mỗi phép tổng của x, ta sẽ phải tính k lần giá trị. Tổng quát hóa trên toàn bài toán ta có n lần phép tính biên k vì thế độ phức tạp của bài toán sẽ là . Độ phức tạp này là quá lớn vì thế ta cần phải tối ưu lại cách giải quyết cho bài toán.

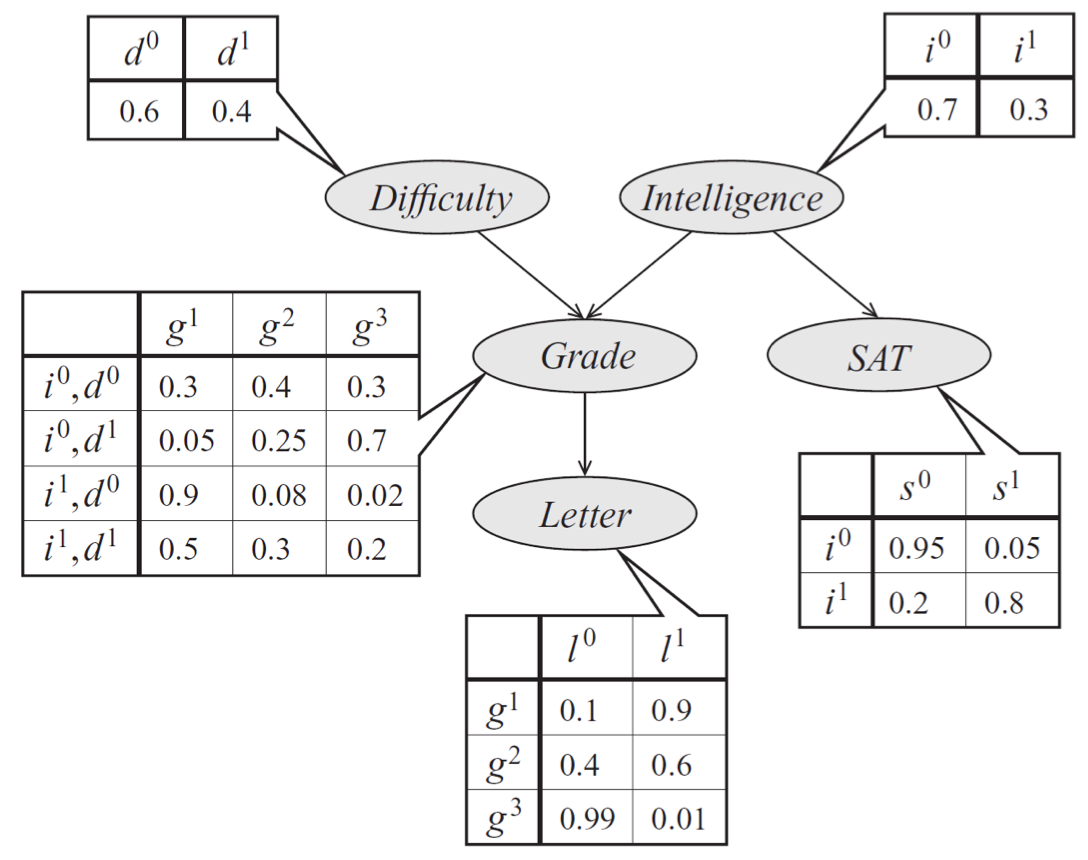
Ta có thể tối ưu bằng cách sau mỗi lần tính tổng ta sẽ đặt kết quả của phép tổng bằng . Ví dụ ta sẽ đặt:

Thế (8) vào (7) ta được:

Ta có thể thấy ở mỗi bước ta chỉ giá trị biên được tính bằng giá trị biên của bước trước đó với giá trị biến ở bước hiện tại. Ở (8) ta có thể thấy với bước hiện tại là làm ví dụ thì ta chỉ cần tính k lần của và k lần của vì thế độ phức tạp ở bước này sẽ là . Như vậy ở mỗi bước độ phức tạp của ta sẽ là . Tổng quát cho n lần phép tính biên ta được độ phức tạp của bài toán sẽ là . Ta thấy nhỏ hơn nhiều so với , độ phức tạp của bài toán đã được tối ưu hơn.

Kỹ thuật variable elimination dựa trên marginal inference để tìm ra xác suất biến mà ta quan tâm hay mong muốn. Variable elimination thực hiện tính giá trị xác suất của các biến mà ta không quan tâm và cuối cùng sẽ nhân các giá trị được tính này lại với nhau.

Ví dụ ta có mô hình đồ thị sau:



1. Mô hình ví dụ cho variable elimination [0]

Ta cần quan tâm xem xác suất p(“Letter”) = p(l) là bao nhiêu.

Vì ta chỉ quan tâm đến “Letter” nên ta cần phải loại bỏ các thuộc tính còn lại. Muốn loại bỏ ta cần phải tính được các giá trị xác suất của các thuộc tính này là bao nhiêu.

Đầu tiên ta sẽ tiến hành loại bỏ thuộc tính “Difficulty” (d) dựa trên “Grade” (g) và “Intelligence” (i).

Tiếp theo ta sẽ tiến hành loại bỏ “Intelligence” (i) bằng cách tính xác suất của nó.

Ta loại bỏ “SAT” (s) bằng cách:

Cuối cùng ta loại bỏ “Grade” (g) bằng cách:

Ta sẽ tìm được p(“Letter”) bằng cách:

Bằng cách tính và loại dần dần các thuộc tính ta đã tìm được xác suất “Letter” mà ta cần tìm bằng kĩ thuật variable elimination.

1.1.4.2 Suy luận Maximum A Posteriori (MAP inference)

Mục đích của MAP inference chính là tối đa hóa giá trị xác suất của dữ liệu x dựa trên các tham số .

MAP dùng để ước tính các đại lượng chưa biết hay các chưa biết, bằng cách tính hậu xác suất . MAP có thể được sử dụng để ước tính giá trị xác suất của một đại lượng không được quan sát trên cơ sở dữ liệu thực nghiệm.

Giả sử ta sử dụng maximum-likelihood estimator (MLE) làm giải thuật học tham số:

Với là hàm mật độ phân phối.

Do luôn dương và không phụ thuộc nên ta rút gọn thành:

1.1.4.3 Suy luận Bayes (Bayesian inference)

Suy luận Bayes là một phương pháp suy luận thống kê trong đó định lý Bayes được sử dụng để cập nhật xác suất cho một giả thuyết khi có thêm bằng chứng hoặc thông tin. Suy luận Bayes là một kỹ thuật quan trọng trong thống kê, và đặc biệt là trong thống kê toán học. Cập nhật Bayes đặc biệt quan trọng trong phân tích động của một chuỗi dữ liệu.

Suy luận Bayes tính hậu xác suất dựa trên 2 phần chính đó là: tiền xác suất và giải thuật học của mô hình thống kê cho dữ liệu quan sát. Suy luận Bayes tính hậu xác suất theo định lý Bayes:

Trong đó:

* H là giả thuyết (hypothesis) đang quan tâm.
* E là bằng chứng (evidence) quan sát được.
* là tiền xác suất của hypothesis trước khi E xảy ra.
* là xác suất của E tính được.
* là xác suất E dựa trên H.
* là xác suất H dựa trên E là giá trị đang quan tâm.

1.2 Tạo và định nghĩa mô hình

Thư viện pgmpy là một thư viện được sử dụng trên python để xây dựng và thực hiện các bài toán liên quan đến các mô hình đồ thị có xác suất.

Trong thư viện pgmpy, cấu trúc mô hình không phụ thuộc tham số hóa mà sẽ dựa trên các giá trị phân phối xác suất có điều kiện (Conditional Probability Distribution – CPD). Vì thế, ta có thể định hình mô hình trước, sau đó định hình tham số đầu vào. Như vậy, tham số có thể thay đổi tùy chỉnh tùy ý mà không ảnh hưởng đến mô hình đã được định hình ban đầu.

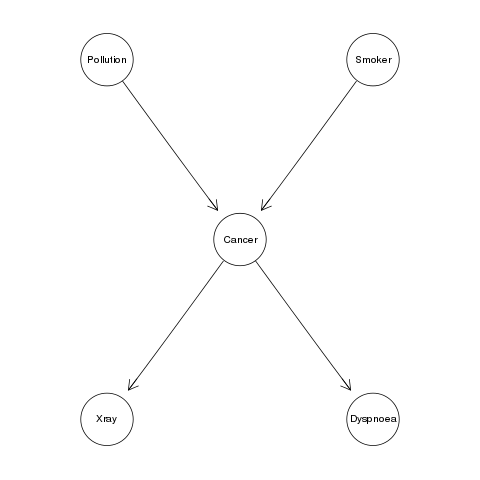
Ta cần biết một số thông tin trước khi tạo và áp dụng mô hình:

1. Dựa vào dữ liệu ta đã biết được cấu trúc của mô hình hay chưa. Nếu dựa vào dữ liệu mà ta biết được cấu trúc của mô hình thì ta sẽ tiến hành xây dựng cấu trúc mô hình dựa vào sự hiểu biết này. Nếu từ dữ liệu mà ta vẫn chưa biết được cấu trúc của mô hình thì ta sẽ tiến hành các giải thuật học cấu trúc để xây dựng cấu trúc cho mô hình.
2. Dữ liệu mà ta có chứa các giá trị rời rạc hay liên tục. Điều này là quan trọng bởi vì hiện nay thư viện pgmpy chưa hỗ trợ cho dữ liệu có tính liên tục, tác giả của thư viện đã nói điều này trên github: <https://github.com/pgmpy/pgmpy/issues/772>.
3. Ngoài ra ta cần hiểu về dữ liệu để từ đó ta có thể chọn ra các giải thuật học sao cho phù hợp.

1.2.1 Xây dựng mô hình khi biết cấu trúc

Để tiện cho việc giải thích, ta sẽ gói gọn lại trong việc thông qua ví dụ để giải thích cách sử dụng thư viện để định nghĩa mô hình gồm 5 trạng thái: Pollution, Smoker, Cancer, Xray, Dysponea (ta gọi tắt là mô hình cancer). Ta có thể tìm hiểu thêm về mô hình cancer tại [https://www.bnlearn.com/bnrepository/discrete-small.html#cancer](https://www.bnlearn.com/bnrepository/discrete-small.html%23cancer).

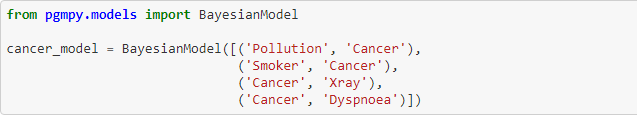
Mô hình cancer là mô hình ta đã biết cấu trúc đồ thị của nó từ trước nên ta sẽ tiến hành xây dựng mô hình từ cấu trúc này.



1. Mô hình đồ thị biểu diễn các mối tương quan của mô hình cancer [1]

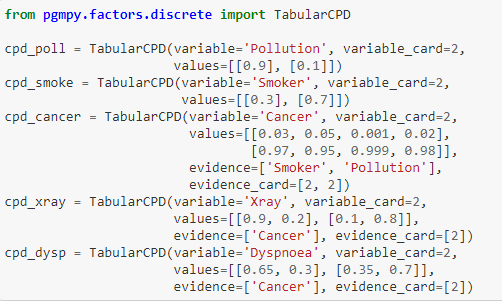
Giả định tạo một mô hình có hướng Bayesian từ đầu như trong ảnh trên.

Mô hình Bayesian được pgmpy cung cấp, BayesianModel, có thể được tạo bằng cách nhập vào danh sách các cạnh.



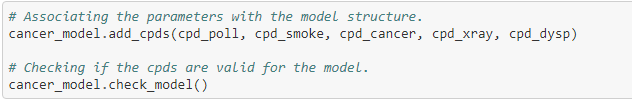
1. Tạo mô hình cancer bằng pgmpy [1]

Mỗi node trong mô hình đều có CPD tương ứng, trong trường hợp ảnh trên có 5 node Pollution, Smoke, Cancer, Xray, Dypnoea. Pgmpy cũng cung cấp lớp định nghĩa CPD là TabularCPD.



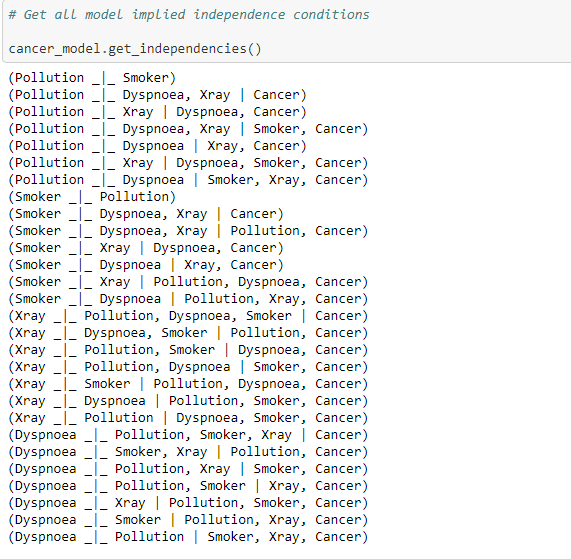
1. Khởi tạo các giá trị xác suất của mô hình 1 [1]

Sau khi đã định nghĩa mô hình và tham số đầu vào, giờ ta có thể them tham số vào mô hình thông qua hàm add\_cpd. Có thể dùng hàm check\_method để kiểm tra CPD có được định nghĩa phù hợp cho mô hình không.



1. Thêm các giá trị xác suất vừa khởi tạo vào mô hình 1 [1]

Sau khi định nghĩa mô hình, ta có thể kiểm định các quan hệ giữa các biến với nhau.



1. Xuất các mối quan hệ độc lập của mô hình 1 [1]

1.2.2 Xây dựng mô hình khi chưa biết cấu trúc

Nếu từ dữ liệu mà ta vẫn chưa biết được cấu trúc của đồ thị, ta sẽ phải tiến hành thực hiện giải thuật học cấu trúc để đi tìm cấu trúc cho đồ thị. Về cách tiến hành giải thuật học cấu trúc cho dữ liệu bằng pgmpy sẽ được hướng dẫn cụ thể tại mục 1.4.2 của tài liệu.

1.3 Suy luận thống kê bằng pgmpy

Trong tiểu mục này ta sẽ thực hiện các ví dụ về xuất các giá trị suy luận thống kê bằng pgmpy.

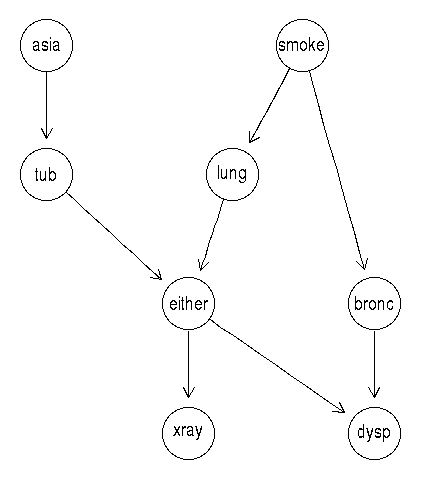
Suy diễn thống kê là quá trình sử dụng dữ liệu phân tích để suy ra các thuộc tính của phân phối xác suất cơ bản. Phân tích thống kê suy diễn đưa ra một tập hợp các đặc tính cần quan tâm.

Trong học máy, một tập các giá trị suy diễn chỉ có thể được xuất từ mô hình khi một tập dữ liệu lớn khác đã đào tạo và huấn luyện cho mô hình.

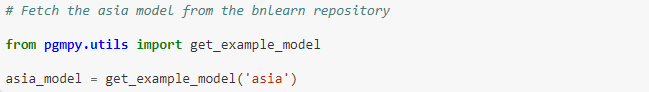
Khác với suy diễn thống kê, suy diễn thống kê sẽ sử dụng chính dữ liệu hiện tại để sinh ra tập các giá trị cần được quan sát, quan tâm.

Phần này ta sẽ thực hiện trên mô hình mẫu có tên là “asia”.

Mô hình “asia” có dạng như sau:

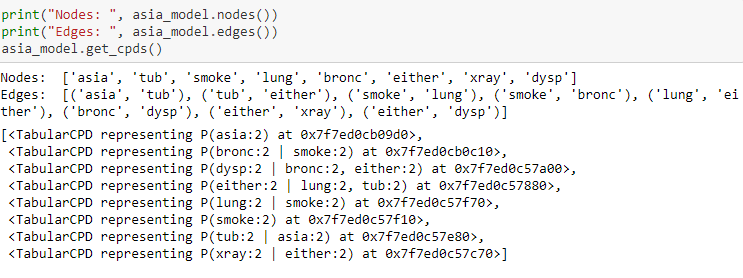


1. Mô hình “asia” [2]



1. Thực hiện lấy mô hình “asia” [3]

Tiếp theo ta sẽ thực hiện việc in các đỉnh (node), cạnh (edge) và các xác suất quan hệ của mô hình “asia”.



1. In các giá trị đỉnh, cạnh và xác suất quan hệ của mô hình “asia” [3]

Tiếp theo ta sẽ thực hiện các suy luận trên mô hình.

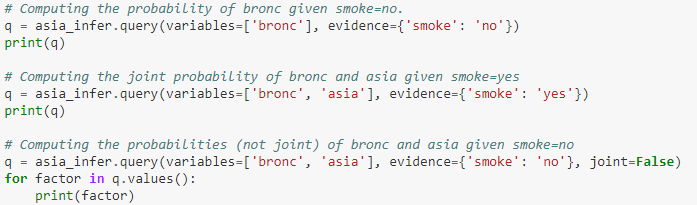
Đầu tiên ta sẽ tìm hiểu về suy luận biên.

Tiếp theo ta sẽ tiến hành trả về đồ thị quy nạp được hình thành bằng cách loại bỏ các biến của mô hình “asia”.



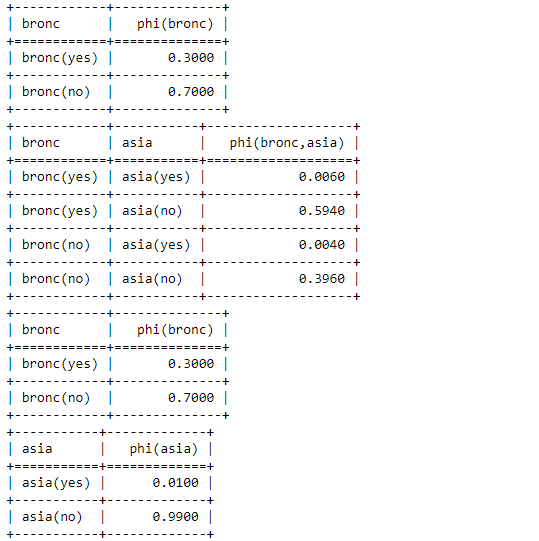
1. Thực hiện loại bỏ các biến của mô hình “asia” [3]

Tiếp theo ta sẽ tiến hành tính toán và in các giá trị xác suất quan hệ mà ta quan tâm.



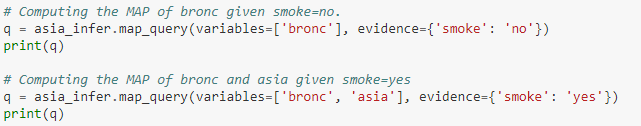
1. Tiến hành in các giá trị xác suất quan hệ mong muốn [3]

Sau khi chạy các câu lệnh ở hình 1.14, ta thu được kết quả:



1. Các bảng thống kê các giá trị xác suất quan hệ mong muốn [3]

Để xem 1 thuộc tính nào đó có bị ảnh hưởng bởi 1 thuộc tính khác không, ta sử dụng hàm map\_query.



1. Kiểm tra xem các thuộc tính có bị ảnh hưởng bởi các thuộc tính khác hay không [3]

Sau khi chạy câu lệnh ở hình 1.16, ta thu được kết quả:



1. Kết quả sau khi chạy câu lệnh hình 1.16 [3]

Như vậy ở hình 1.17 ta thu được kết quả đó là thuộc tính “bronc” có bị ảnh hưởng bởi thuộc tính “smoke” và thuộc tính “asia” không bị ảnh hưởng bởi thuộc tính “smoke”.

1.4 Thực hiện các giải thuật học bằng pgmpy

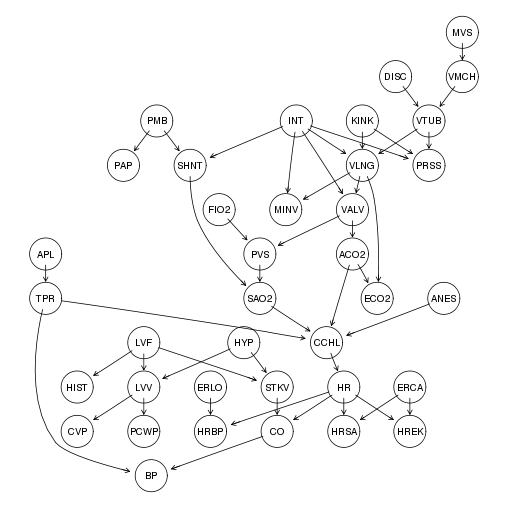
1.4.1 Parameter

Pgmpy có hai cách chính để học tham số:

* Ước tính MaximumLikelihood (Maximum-likelihood estimator).
* Ước tính Bayesian (Bayesian estimator).

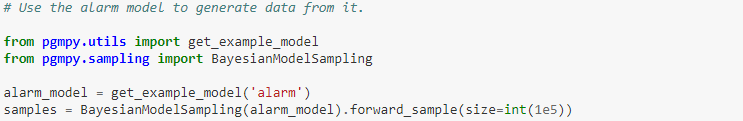
Để minh họa cho tiểu mục này, ta sẽ sử dụng mô hình và tham số có sẵn là alarm và sẽ học tham số mô hình từ dữ liệu đã có sẵn.

Mô hình “alarm” có dạng như sau:



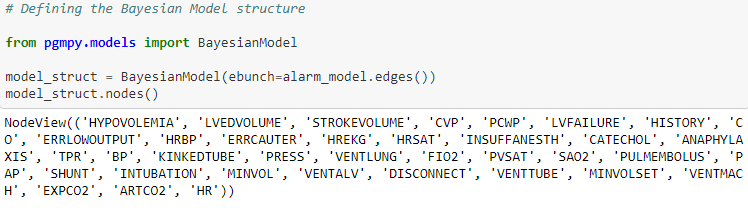
1. Mô hình “alarm” [4]

Bước đầu, ta lấy dữ liệu được pgmpy cung cấp sẵn.



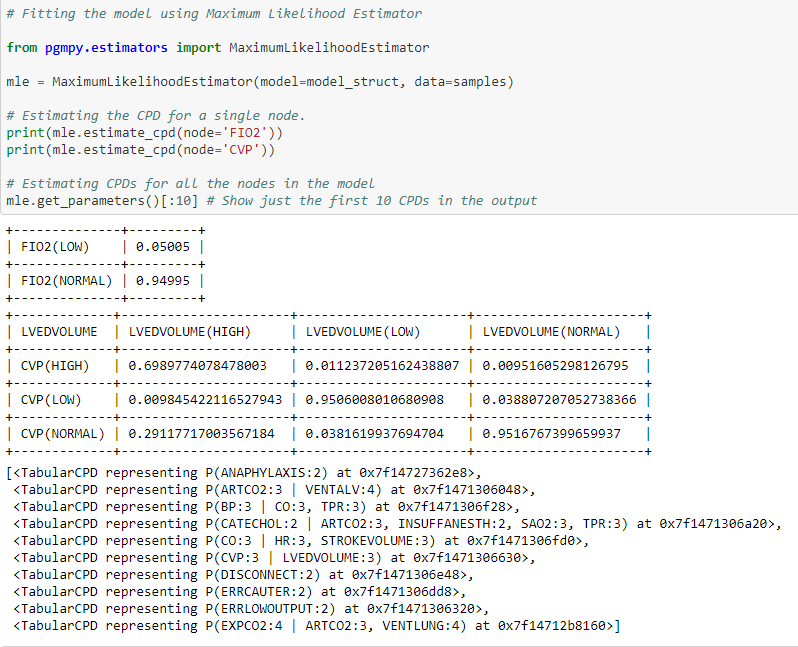
1. Lấy và tạo mô hình “alarm” [5]

Tiếp đến, ta có thể tạo mô hình “alarm” các dữ liệu các cặp cạnh đã được định nghĩa sẵn.

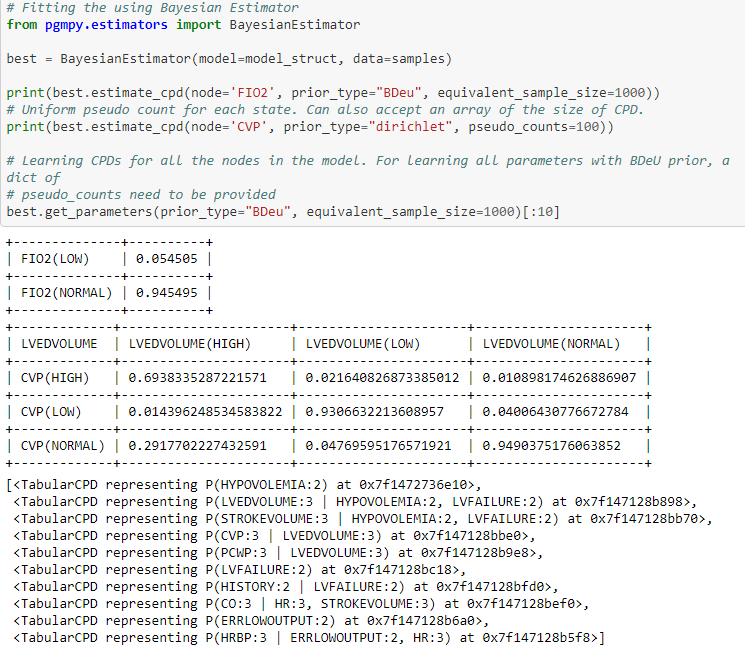


1. Tạo mô hình “alarm” bằng các cặp cạnh đã được định nghĩa [5]

Với mô hình và dữ liệu có được, ta có thể tiến hành học các tham số bằng MaximumLikelihoodEstimator hay BayesianEstimator.



1. Tính toán các giá trị xác suất cho mô hình bằng MaximumLikelihoodEstimator [5]



1. Tính toán các giá trị xác suất cho mô hình bằng BayesianEstimator [5]

1.4.2 Structure

Ở thời điểm bài bài báo cáo này, pgmpy đã có sẵn 3 thuật toán học cấu trúc:

* Thuật toán dựa trên ràng buộc (Prototypical Constraint-based – PC) với các biến thể ổn định và song song.
* Thuật toán tìm kiếm leo đồi (Hill-Climb Search).
* Thuật toán tìm kiếm vét cạn (Exhaustive search).

Với thuật toán PC các phép thử có điều kiện có thể thực hiện trên pgmpy gồm:

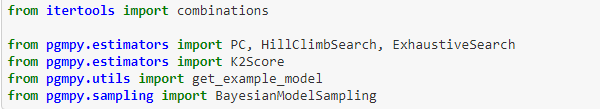
* Kiểm định Pearsonr (Pearsonr test).
* Kiểm thử Chi bình phương (Chi-square test).

Đối với các giải thuật học tìm kiếm pgmpy có các phương pháp kiểm thử:

* K2 Score
* BDeu Score
* Bic Score

Để minh họa cho học cấu trúc, thay vì phải định hình lại cấu trúc hay tham số đầu vào như tiểu mục tạo và định hình mô hình. Ta sẽ sử dụng mô hình và dữ liệu có sẵn của pgmpy là alarm.

Đầu tiên nhập các thư viện cần có và chuẩn bị mô hình, dữ liệu.



1. Chuẩn bị các thư viện cho việc học cấu trúc [6]

Lấy dữ liệu của mô hình alarm.



1. Lấy dữ liệu mô hình alarm [6]

Sau khi đã chuẩn bị mô hình và tài nguyên. Ta có thể học cấu trúc mô hình bằng thuật toán PC với biến thể ổn định (stable) hay orig.

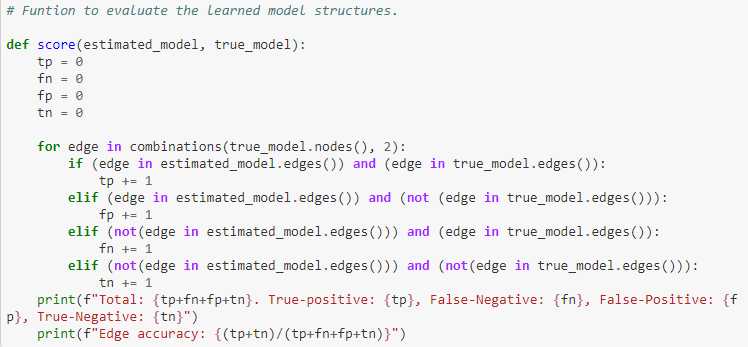


1. Xây dựng mô hình bằng thuật toán PC biến thể stable [6]



1. Xây dựng mô hình bằng thuật toán PC biến thể orig [6]

Tiếp theo ta sẽ xây dựng hàm score để đánh giá mức độ chính xác của quá trình học cấu trúc.



1. Xây dựng hàm score để đánh giá thuật toán [6]

Cuối cùng ta thu được kết quả đánh giá của việc xây dựng mô hình bằng thuật toán PC với biến thể “stable”:



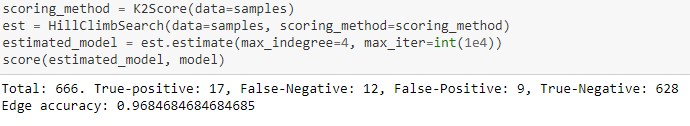
1. Kết quả đánh giá thuật toán PC biến thể “stable” [6]

Và kết quả đánh giá của việc xây dựng mô hình bằng thuật toán PC với biến thể “orig”:



1. Kết quả đánh giá thuật toán PC biến thể “orig” [6]

Tiếp theo ta sẽ thực hiện thử việc xây dựng cấu trúc mô hình bằng thuật toán leo đồi, kiểm thử bằng K2 Score và tiến hành xuất kết quả đánh giá:



1. Tiến hành xây dựng học cấu trúc cho mô hình bằng Hill-climb [6]

1.5 Tóm tắt chương 1

Như vậy, trong chương 1, ta đã nắm một số khái niệm khái quát về mô hình đồ thị xác suất. Bao gồm các yếu tố cơ bản của đồ thị, đồ thị có hướng, đồ thị vô hướng, ta cũng tìm hiểu về các phép tính toán suy luận để tìm ra xác suất giữa quan hệ các biến ngẫu nhiên, và học tham số, học cấu trúc của mô hình. Ta cũng đã tìm hiểu về thư viện PGMPY dùng để tạo mô hình, vẽ đồ thị xác suất. Thông qua các minh họa, ta đã tìm hiểu về cách ứng dụng thư viện pgmpy tạo, định hình mô hình từ đầu, định hình các tham số đầu vào để có thể dễ dàng thay đổi tham số khi cần mà không ảnh hưởng mô hình. Ta cũng tìm hiểu cách suy luận bằng phương pháp loại bỏ biến của thư viện pgmpy. Xây dựng mô hình để học cấu trúc từ dữ liệu đạ có, học tham số trên dữ liệu.

Sau khi đã tìm hiểu về các chức năng cơ bản của thư viện pgmpy cùng lý thuyết cơ bản của mô hình đồ thị xác suất, ta sẽ sang bước tiếp theo trong bài báo cáo này là xây dựng ứng dụng minh họa.

CHƯƠNG 2: ỨNG DỤNG PHÂN TÍCH NHIỄM BỆNH

* 1. Bayesian

2.1.1 Giới thiệu

Mạng Bayes (BN) được xác định bởi:

một cấu trúc mạng, một đồ thị xoay chiều có hướng G, trong đó mỗi nút vi ∈ V tương ứng với một biến ngẫu nhiên Xi;

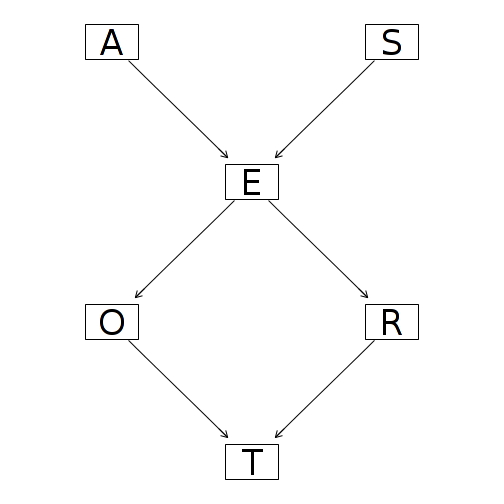
phân phối xác suất toàn cục X (với tham số Θ), có thể được tính theo nhân tử thành các phân phối xác suất cục bộ nhỏ hơn theo các cung có trong biểu đồ.

Vai trò chính của cấu trúc mạng là thể hiện các mối quan hệ độc lập có điều kiện giữa các biến trong mô hình thông qua sự phân tách bằng đồ thị, do đó xác định nhân tử của phân phối toàn cục:

phân hủy thành các phân phối cục bộ

Mỗi phân phối cục bộ có tập tham số riêng ΘXi; và ⋃ ΘXi nhỏ hơn nhiều so với Θ vì nhiều tham số được cố định bởi thực tế là các biến chúng thuộc về độc lập.

Vì vậy, thành phần đầu tiên là một đồ thị xoay chiều có hướng như thế này:



1. Minh họa mô hình mạng Bayesian

Thành phần thứ hai của BN là phân phối xác suất P (X). Sự lựa chọn phải được sao cho BN:

* có thể được học một cách hiệu quả từ dữ liệu;
* linh hoạt (nó có thể mã hóa nhiều loại hiện tượng hợp lý);
* dễ dàng truy vấn để thực hiện suy luận.

Ba lựa chọn phổ biến nhất trong tài liệu (cho đến nay), là:

* BN rời rạc, trong đó X và Xi ∣ ΠXi; và ΘXi là xác suất có điều kiện
* Gaussian BNs (GBN), trong đó X là chuẩn đa biến và Xi ∣ ΠXi là chuẩn đơn biến được xác định bởi mô hình hồi quy tuyến tính
* BN Gaussian tuyến tính có điều kiện (CLGBN), trong đó X là hỗn hợp của các chuẩn tắc đa biến và Xi ∣ ΠXi là đa thức, chuẩn đơn biến hoặc hỗn hợp của các chuẩn.
  + Xi rời rạc chỉ được phép có phần tử cấp trên rời rạc (ký hiệu là ΔXi), được giả định tuân theo phân phối đa thức được tham số hóa với bảng xác suất có điều kiện;
  + Xi liên tục được phép có cả phần tử cấp trên rời rạc và liên tục (ký hiệu là ΓXi, ΔXi ∪ ΓXi = ΠXi, và phân phối cục bộ của chúng là

có thể được viết dưới dạng hỗn hợp các hồi quy tuyến tính

chống lại các phần tử cấp trên liên tục với một thành phần cho mỗi cấu hình δXi của phần tử cấp trên rời rạc. Nếu Xi không có phần tử cấp trên rời rạc, thì hỗn hợp sẽ chuyển thành một hồi quy tuyến tính duy nhất.

2.1.2 Học

Lựa chọn mô hình và ước tính BN được gọi chung là học tập và thường được thực hiện như một quy trình gồm hai bước:

* học cấu trúc, học cấu trúc mạng từ dữ liệu;
* học tham số, học các phân phối cục bộ được ngụ ý bởi cấu trúc đã học ở bước trước.

Quy trình làm việc này là Bayesian; đã cho một tập dữ liệu D và nếu chúng ta biểu thị các tham số của phân phối toàn cục là X với Θ, chúng ta có:

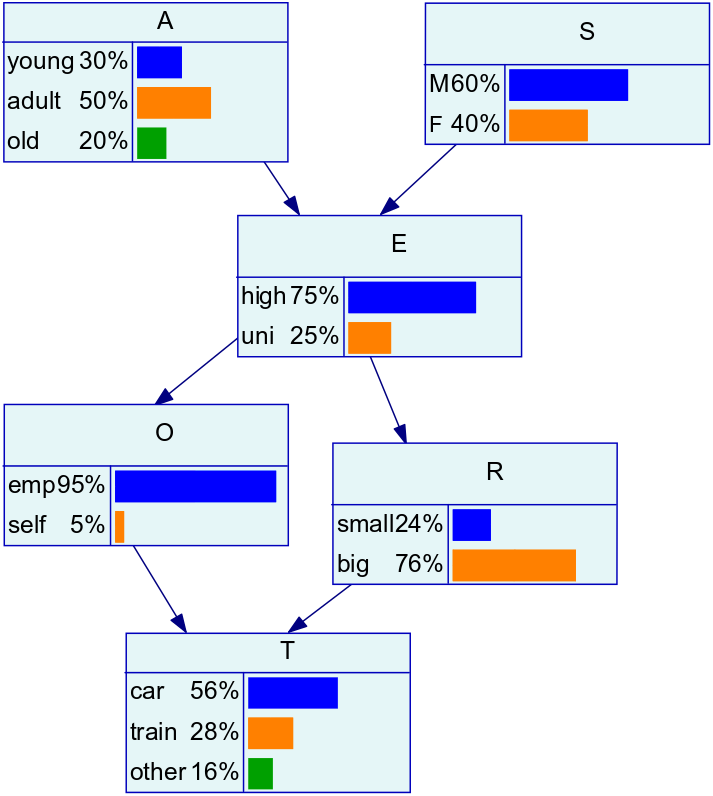
Với là kết quả học được, là kết quả học cấu trúc và là kết quả học tham số.

2.1.3 suy luận xác suất

Suy luận trên BN thường bao gồm các truy vấn xác suất có điều kiện (CP) hoặc tối đa một truy vấn sau thử nghiệm (MAP). Ý tưởng chung của truy vấn CP là:

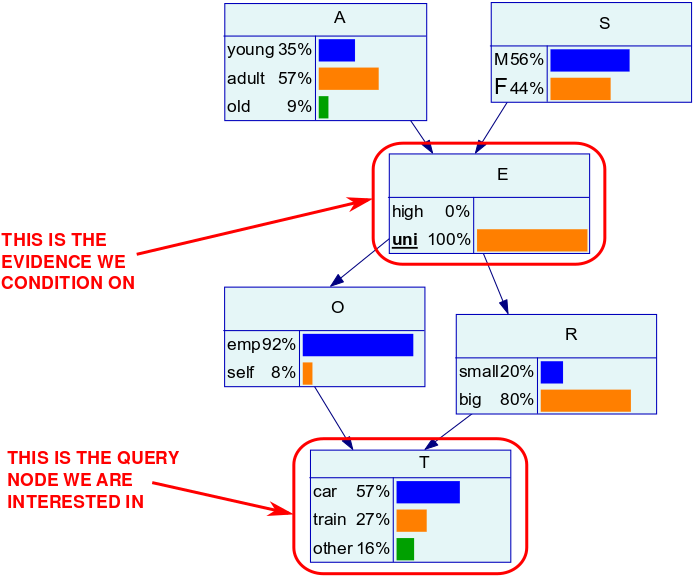
* Ta có một số bằng chứng, đó là chúng ta biết giá trị của một số biến và chúng ta sửa các nút cho phù hợp
* chúng ta muốn xem xét xác suất của một số sự kiện liên quan đến (một tập hợp con của) các biến khác có điều kiện dựa trên bằng chứng chúng ta có.

Giả sử chúng ta bắt đầu từ một BN rời rạc với DAG được sử dụng ở trên.



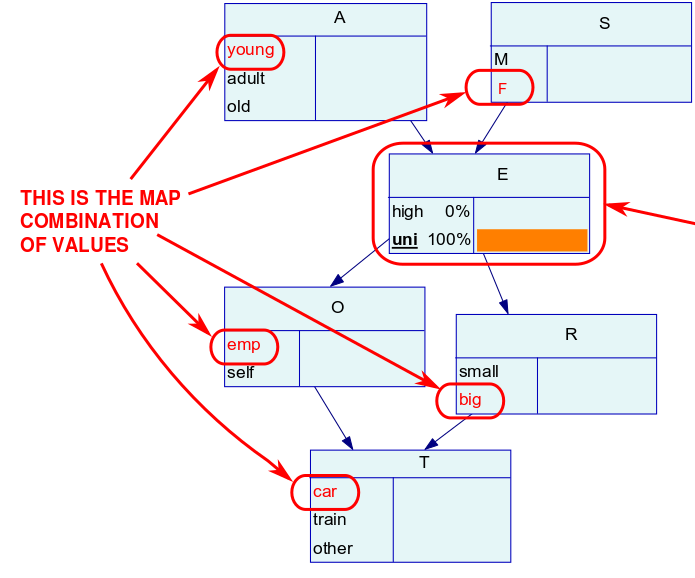
1. Minh họa inference suy luận xác suất Bayesian

Về mặt đồ họa, một truy vấn CP trông như thế này:



1. Minh họa inference suy luận xác suất Bayesian

Mặt khác, mục tiêu của truy vấn MAP là tìm tổ hợp các giá trị cho (một tập hợp con của) các biến trong mạng có xác suất cao nhất được đưa ra một số bằng chứng. Nếu bằng chứng là một cá thể mới được quan sát một phần, thì việc thực hiện truy vấn MAP giống như một bài tập dự đoán cổ điển.



1. Minh họa inference suy luận xác suất Bayesian

2.2 Phân tích đề tài

Sau khi đã tìm hiểu về kiến thức mô hình đồ thị xác suất, ta thực hiện một ứng dụng nhỏ để áp dụng thư viện vào thực tiễn. Dùng mô hình Bayesian để thu dự đoán khả năng nhiễm dựa trên các triệu chứng của người bệnh.

Dữ liệu được thực nghiệm lấy từ nguồn: <https://www.kaggle.com/itachi9604/disease-symptom-description-dataset?select=dataset.csv>.

Dữ liệu chứa 41 loại bệnh. Mỗi người bị bệnh sẽ đi kèm với 3 triệu chứng (symptom) liên quan (Symptom\_1, Symptom\_2, Symptom\_3). Symptom\_1 có 34 loại triệu chứng khác nhau. Symptomm\_2 có 48 loại triệu chứng khác nhau. Symptom\_3 có 54 triệu chứng liên quan.

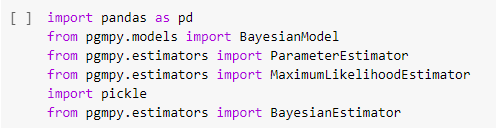
Ta sẽ sử dụng pgmpy và bayesian network để xây dựng bộ giá trị xác suất của mối quan hệ giữa 3 symptom với loại bệnh.

Từ đó ta sẽ sử dụng bộ giá trị này để biết được xác suất người đó sẽ mắc từng loại bệnh là bao nhiêu khi người đó có 3 triệu chứng tương ứng. Từ đó ta sẽ chọn ra giá trị xác suất cao tương ứng với khả năng cao người đó sẽ bị bệnh đó khi mắc 3 triệu chứng tương ứng.

* 1. Xây dựng chương trình

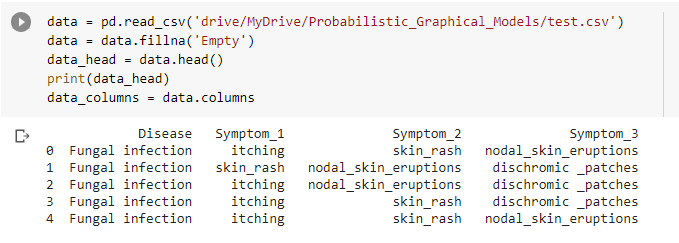
Ta sẽ tiến hành xây dựng chương trình cho ứng dụng.

Đầu tiên ta sẽ chuẩn bị các thư viện.



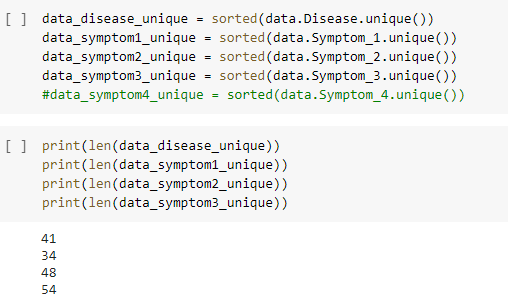
1. Chuẩn bị thư viện cho chương trình

Tiếp theo ta sẽ nạp dữ liệu vào chương trình. Đồng thời ta sẽ thay các giá trị rỗng bằng “Empty”.



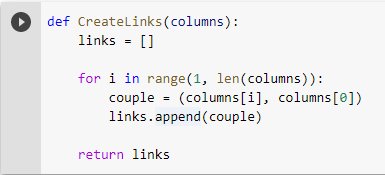
1. Tiến hành nạp dữ liệu vào chương trình

Ta sẽ tiến hành lọc ra các giá trị riêng biệt cho từng loại bệnh, symptom trong dữ liệu.



1. Lọc ra các giá trị riêng biệt cho từng loại bệnh, symptom

Tiếp theo ta sẽ xây dựng hàm tạo các cặp cạnh cho mô hình.



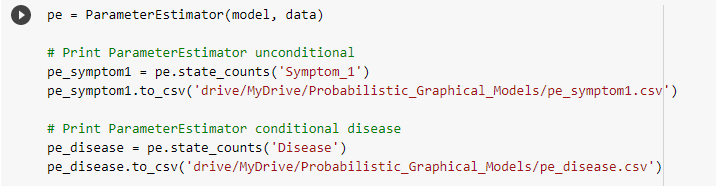
1. Xây dựng hàm tạo các cặp cạnh cho mô hình

Từ các cặp cạnh này ta sẽ tiến hành xây dựng mô hình bayesian network.



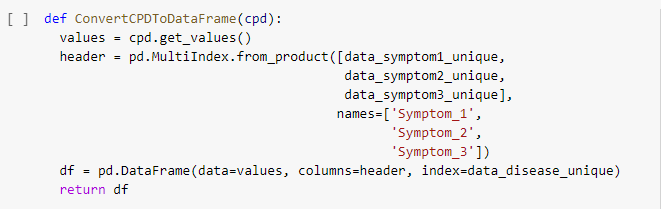
1. Tiến hành xây dựng mô hình

Tiếp theo ta sẽ tiến hành thống kê cho từng loại symptom, và số lần xuất hiện bệnh ứng với các symptom.



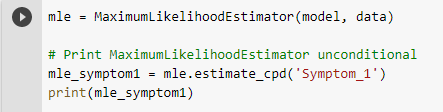
1. Tiến hành thống kê symptom và disease

Do dữ liệu quá lớn và IDE không hỗ trợ việc in kết quả khi dữ liệu lớn nên ta sẽ tiến hành xây dựng hàm để chuyển đổi từ loại của pgmpy sang dataframe của pandas để xuất ra csv để ta có thể xem được.



1. Xây dựng hàm chuyển đổi sang dataframe

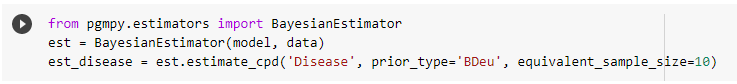
Cuối cùng ta sẽ tiến hành tính toán các giá trị xác suất bằng MaximumLikelihooddEstimator và BayesianEstimator.



1. Tính xác suất cho Symptom\_1 bằng MaximumLikelihooddEstimator

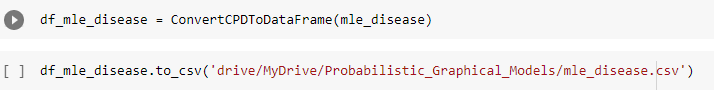


1. Tính xác suất cho Disease bằng MaximumLikelihooddEstimator

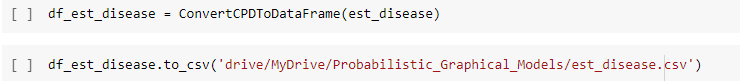


1. Tính xác suất cho Disease bằng BayesianEstimator

Cuối cùng ta sẽ xuất kết quả ra file csv.



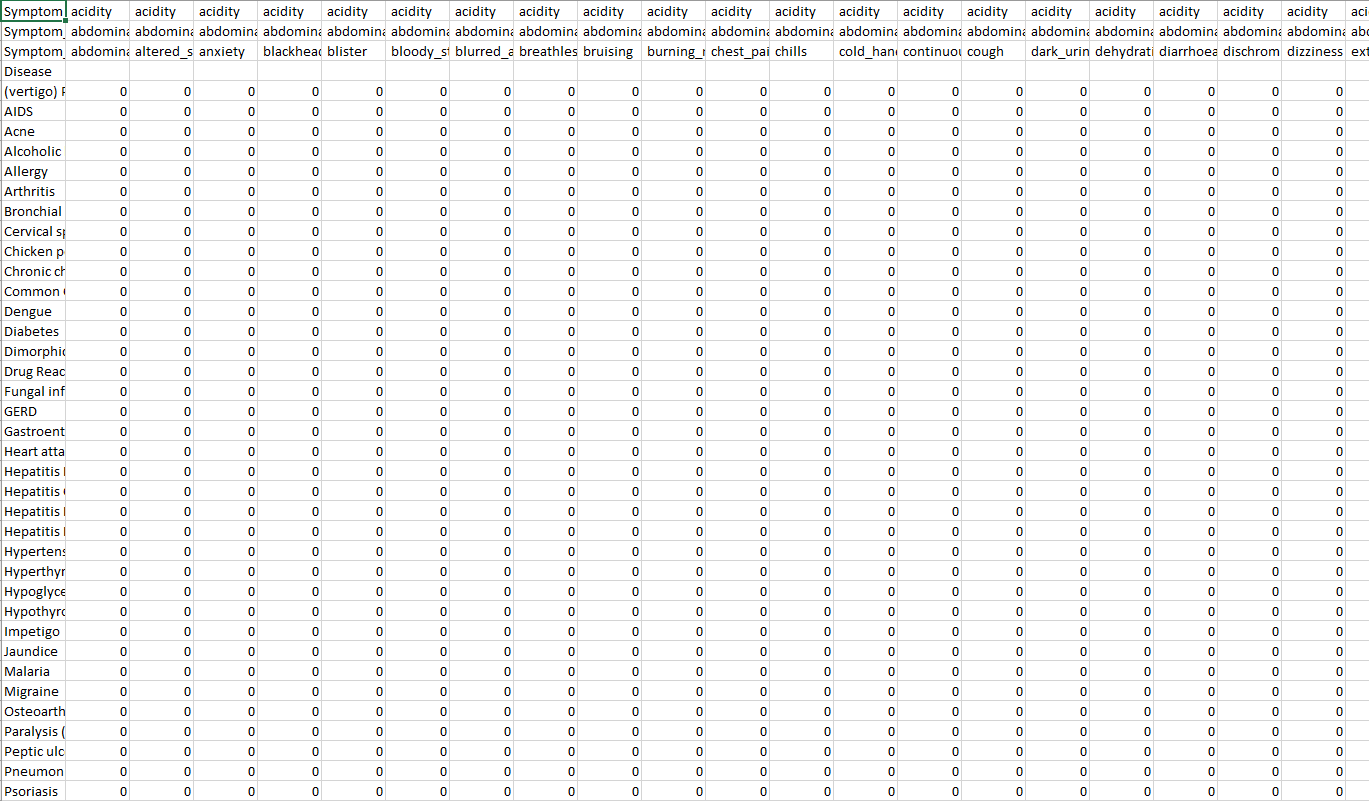
1. Xuất kết quả tính xác suất của Disease bằng MaximumLikelihooddEstimator ra file CSV



1. Xuất kết quả tính xác suất của Disease bằng BayesianEstimator ra file CSV

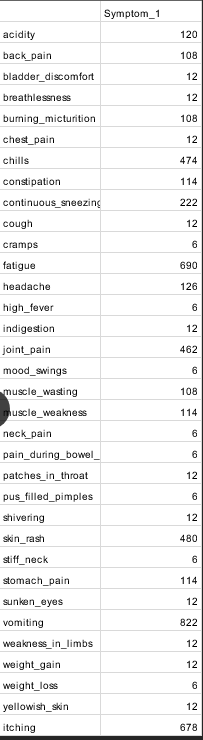
2.4 Kết quả học mô hình

Ta có một số bản khác nhau. Bảng thứ nhất thống kê tất các bệnh khác nhau. Với từng bệnh ta có một nhóm 3 các triệu chứng khác nhau



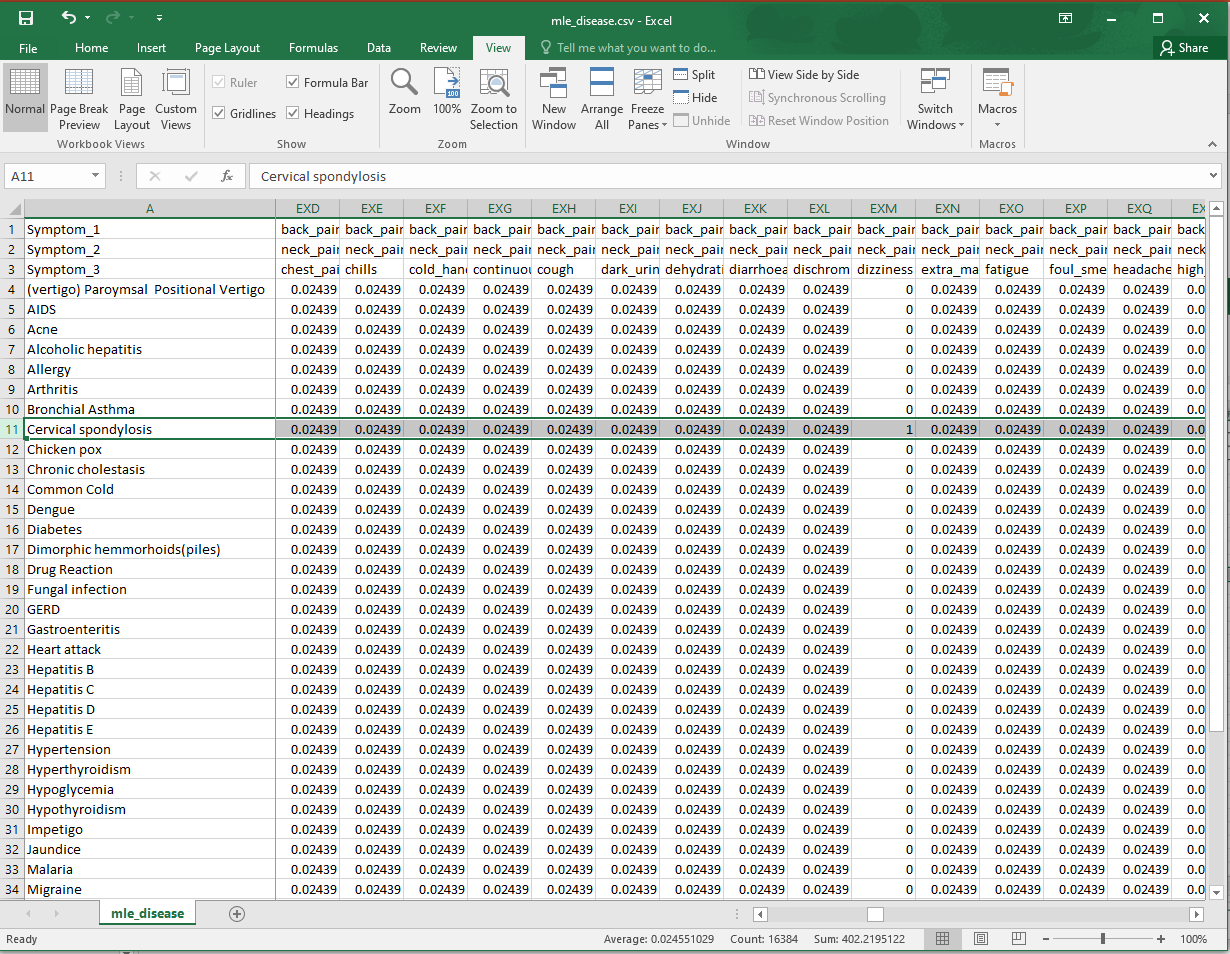
1. Thống kê các triệu chứng cho các bệnh khác nhau

Bảng tiếp theo ghi nhận có bao nhiêu triệu chứng đã được liệt kê trong nhóm các triệu chứng có thể xảy ra trong từng dịch bệnh.



1. Thống kê số lần xuất hiện các triệu chứng của Symptom\_1

Tiếp theo ta sẽ cho chương trình tính các giá trị xác suất quan hệ giữa 3 symptom với từng loại bệnh.



1. Bảng thống kê xác suất giữa symptom và disease

Từ hình 2.7 ta có thể thấy rằng ứng với 1 người có các triệu chứng là “back\_pain”, “neck\_pain”, “dizziness” sẽ có khả năng mắc bệnh “AIDS” là 0. Tuy nhiên khả năng mắc bệnh “Cervical spondylosis” là 1. Từ đó ta có thể suy luận rằng khi 1 người bị có các triệu chứng “back\_pain”, “neck\_pain”, “dizziness” sẽ có khả năng cao mắc bệnh “Cervical spondylosis”.

**BỔ SUNG**

**Chương 1**

**1.5** Kiểm định Chi square Pearsonr

Kiểm tra chi bình phương của Pearson ( ) là kiểm tra thống kê được áp dụng cho các tập hợp dữ liệu phân loại để đánh giá khả năng xảy ra bất kỳ sự khác biệt quan sát nào giữa các tập hợp một cách tình cờ. Nó được sử dụng rộng rãi nhất trong nhiều phép thử chi bình phương (ví dụ: Yates, tỷ lệ khả năng xảy ra, thử nghiệm tổng thể trong chuỗi thời gian, v.v.) - các thủ tục thống kê có kết quả được đánh giá bằng cách tham chiếu đến phân phối chi bình phương. Các đặc tính của nó được Karl Pearson nghiên cứu lần đầu tiên vào năm 1900. Trong bối cảnh điều quan trọng là phải cải thiện sự khác biệt giữa thống kê thử nghiệm và phân phối của nó, các tên tương tự nhth”thống kê” hoặth”thử nghiệm bình phương Pearsođ” được sử dụng. Nó kiểm tra một giả thuyết rỗng nói rằng phân bố tần suất của các sự kiện nhất định được quan sát trong một mẫu phù hợp với một phân bố lý thuyết cụ thể. Các sự kiện được xem xét phải loại trừ lẫn nhau và có tổng xác suất là 1. Một trường hợp phổ biến cho điều này là mỗi sự kiện bao hàm một kết quả của một biến phân loại. Một ví dụ đơn giản là giả thuyết rằng một con súc sắc sáu mặt bình thường là "công bằng" (ví dụ: tất cả sáu kết quả đều có khả năng xảy ra như nhau.)

Phép thử chi bình phương của Pearson được sử dụng để đánh giá ba loại so sánh: mức độ phù hợp, tính đồng nhất và tính độc lập.

* Kiểm tra mức độ phù hợp thiết lập liệu phân bố tần số quan sát có khác với phân bố lý thuyết hay không.
* Một bài kiểm tra tính đồng nhất so sánh sự phân bố số lượng cho hai hoặc nhiều nhóm bằng cách sử dụng cùng một biến phân loại (ví dụ: lựa chọn hoạt động — đại học, quân đội, việc làm, du lịch — của học sinh tốt nghiệp trung học được báo cáo một năm sau khi tốt nghiệp, được sắp xếp theo năm tốt nghiệp, để xem liệu số sinh viên tốt nghiệp lựa chọn một hoạt động nhất định có thay đổi từ lớp này sang lớp khác, hoặc từ thập kỷ này sang thập kỷ khác hay không).
* Kiểm tra tính độc lập đánh giá liệu các quan sát bao gồm các phép đo trên hai biến số, được thể hiện trong bảng dự phòng, có độc lập với nhau hay không (ví dụ: các câu trả lời thăm dò ý kiến ​​từ những người thuộc các quốc tịch khác nhau để xem liệu quốc tịch của một người có liên quan đến phản hồi hay không).

Đối với cả ba bài kiểm tra, quy trình tính toán bao gồm các bước sau:

Tính toán thống kê kiểm tra chi bình phương, χ², giống như một tổng bình phương được chuẩn hóa giữa các tần số quan sát và lý thuyết.

Xác định bậc tự do, df, của thống kê đó.

Đối với kiểm tra mức độ phù hợp, df = Cats - Parms, trong đó Mèo là số loại quan sát được mô hình công nhận và Parms là số thông số trong mô hình được điều chỉnh để làm cho mô hình phù hợp nhất với các quan sát: số loại giảm bởi số lượng các thông số được trang bị trong phân phối.

Để kiểm tra tính đồng nhất, df = (Hàng - 1) × (cột - 1), trong đó Hàng tương ứng với số danh mục (tức là các hàng trong bảng dự phòng được liên kết) và cột tương ứng với số nhóm độc lập (tức là các cột trong bảng dự phòng liên quan).

Để kiểm tra tính độc lập, df = (Hàng - 1) × (cột - 1), trong trường hợp này, Hàng tương ứng với số danh mục trong một biến và Cols tương ứng với số danh mục trong biến thứ hai.

Chọn mức độ tin cậy mong muốn (mức ý nghĩa, giá trị p hoặc mức alpha tương ứng) cho kết quả của phép thử.

So sánh với giá trị tới hạn từ phân phối chi bình phương với df bậc tự do và mức độ tin cậy đã chọn (một phía vì phép thử chỉ là một hướng, tức là giá trị thử nghiệm có lớn hơn giá trị tới hạn không?), trong nhiều trường hợp, giá trị này cho phép phân phối gần đúng của .

Duy trì hoặc bác bỏ giả thuyết rỗng rằng phân phối tần suất quan sát được giống như phân phối lý thuyết dựa trên việc liệu thống kê thử nghiệm có vượt quá giá trị tới hạn của hay không. Nếu thống kê thử nghiệm vượt quá giá trị tới hạn của thì giả thuyết rỗng H 0 = không có sự khác biệt giữa các phân phối ) có thể bị từ chối và giả thuyết thay thế H1 = có sự khác biệt giữa các phân phối) có thể được chấp nhận, cả với mức độ tin cậy đã chọn. Nếu thống kê thử nghiệm giảm xuống dưới giá trị ngưỡng , thì không thể đưa ra kết luận rõ ràng và giả thuyết rỗng được duy trì (chúng tôi không thể bác bỏ giả thuyết rỗng), nhưng không nhất thiết được chấp nhận

**Chương 2**

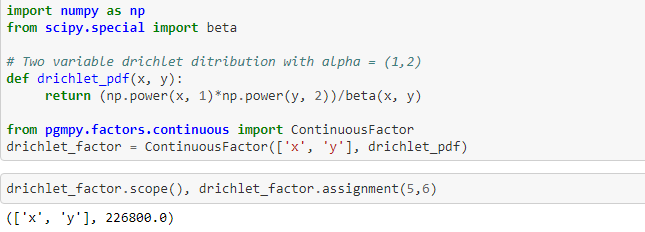
2.3 Dữ liệu liên tục

Trọng tâm của tiểu mục này nhằm giải thích cách xử lý các biến liên tục, biến rời rạc của thư viện pgmpy.

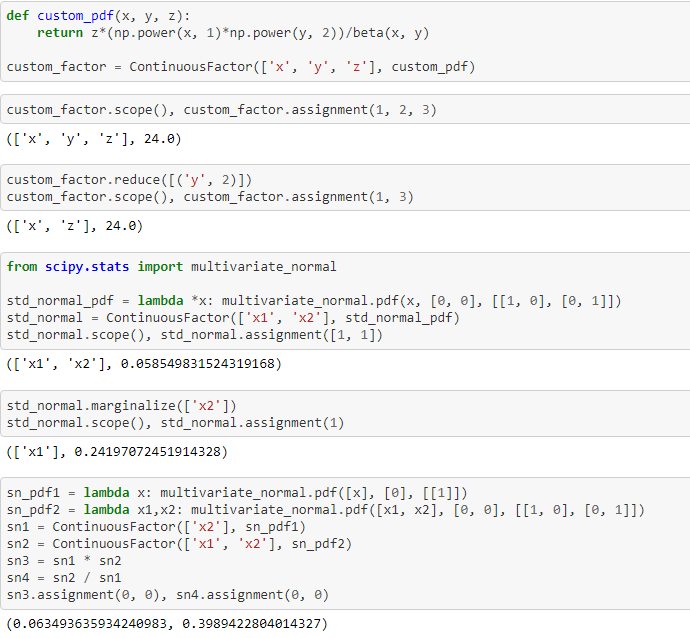
Trong nhiều tình huống, một số biến được mô hình hóa tốt nhất là nhận các giá trị trong một số không gian liên tục. Ví dụ bao gồm các biến như vị trí, vận tốc, nhiệt độ và áp suất. Rõ ràng, chúng ta không thể sử dụng biểu diễn bảng trong trường hợp này. Không có gì trong công thức của mạng Bayes yêu cầu chúng ta hạn chế chú ý đến các biến rời rạc. Yêu cầu duy nhất là CPD, P (X | Y1, Y2, ... Yn) đại diện cho mọi phép gán giá trị y1 ∈ Val (Y1), y2 ∈ Val (Y2), ..... yn ∈ val (Yn), một phân phối trên X. Trong trường hợp này, X có thể là liên tục, trong trường hợp đó, CPD sẽ cần biểu diễn các phân phối trên một chuỗi giá trị liên tục; chúng ta cũng có thể có phần tử cấp trên của X liên tục, do đó CPD cũng sẽ cần biểu thị một chuỗi liên tục của các phân phối xác suất khác nhau. Tồn tại các đại diện ngầm định cho CPD thuộc loại này, cho phép chúng tôi áp dụng tất cả các máy móc mạng cho trường hợp liên tục.

2.3.2.1 Lớp cơ sở cho các yếu tố liên tục

Lớp này sẽ hoạt động như một lớp cơ sở cho các biểu diễn yếu tố liên tục. Tất cả các lớp nhân tố hiện tại và tương lai sẽ được dẫn xuất từ lớp cơ sở này. Chúng ta cần chỉ định tên biến và một hàm pdf để khởi tạo lớp này.



Lớp này hỗ trợ các phương thức như biên, giảm, tích và chia giống như những gì chúng ta có với các lớp rời rạc. Một lưu ý là khi có một số biến liên quan, các phương pháp này chứng tỏ là không hiệu quả và do đó chúng tôi sử dụng một số Gaussian nhất định hoặc một số phép gần đúng khác sẽ được thảo luận sau.

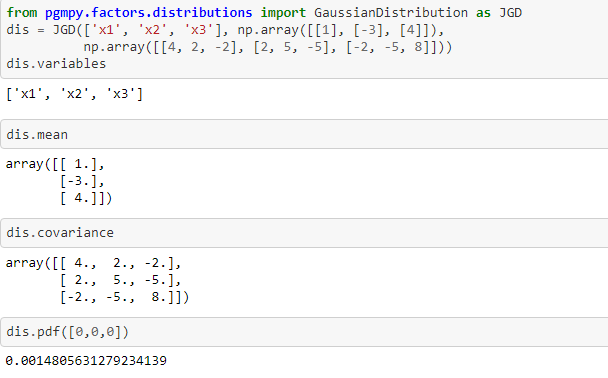


Lớp ContinuousFactor cũng có một phương thức tùy ý lấy một lớp pgmpy Discretizer làm đầu vào. Nó sẽ xuất ra một danh sách các khối lượng xác suất rời rạc hoặc một đối tượng Factor hoặc TabularCPD tùy thuộc vào phương pháp tùy chỉnh được sử dụng. Mặc dù hiện tại, chúng tôi không có các thuật toán tùy biến sẵn có cho các bản phân phối đa biến, người dùng luôn có thể xác định lớp Discretizer của riêng mình bằng cách phân lớp lớp pgmpy.BaseDiscretizer.

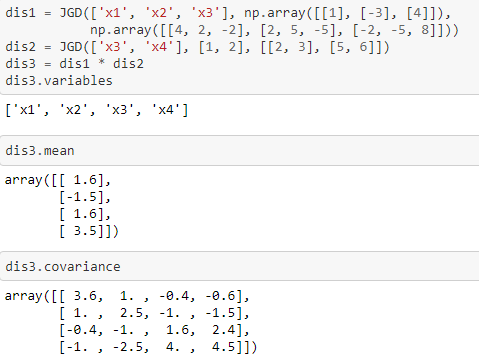
**2.3.2.2** Phân phối Gaussian chung

Trong cách biểu diễn phổ biến nhất của nó, phân phối Gaussian đa biến trên X1 ……… ..Xn được đặc trưng bởi vectơ trung bình n chiều μ và ma trận hiệp phương sai n x n đối xứng Σ. Hàm mật độ được định nghĩa nhiều nhất là –

Lớp pgmpy.JointGaussianDistribution cung cấp đại diện của nó. Điều này có nguồn gốc từ lớp pgmpy.ContinuousFactor. Chúng ta cần chỉ định các tên biến, một vectơ trung bình và một ma trận hiệp phương sai cho sự vô hiệu hóa của nó. Nó sẽ tự động kết nối chức năng pdf với các tham số này.



Lớp này ghi đè các phương thức hoạt động cơ bản (biên, giảm, chuẩn hóa, tích và chia) vì các hoạt động này ở đây hiệu quả hơn các phương thức trong lớp cha của nó. Hầu hết các hoạt động này liên quan đến một nghịch đảo ma trận là O (n ^ 3) đối với số lượng biến.



Các phương pháp khác cũng có thể được sử dụng theo cách tương tự.

2.3.2.3 Các yếu tố chính tắc

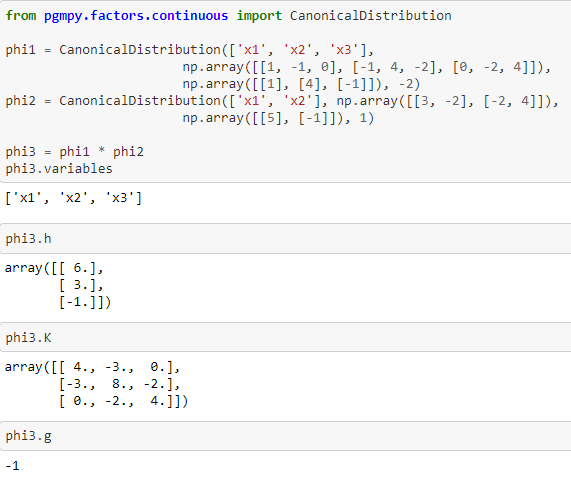
Trong khi biểu diễn Gaussian chung hữu ích cho các thuật toán lấy mẫu nhất định, xem xét kỹ hơn sẽ thấy rằng nó cũng không thể được sử dụng trực tiếp trong các thuật toán tổng tích. Tại sao? Bởi vì các hoạt động như sản phẩm và giảm bớt, như đã đề cập ở trên liên quan đến việc đảo ngược ma trận ở mỗi bước.

Vì vậy, để mô tả ngắn gọn các yếu tố trung gian trong mạng Gaussian mà không cần phải nghịch đảo ma trận tốn kém ở mỗi bước, một biểu diễn tham số đơn giản được sử dụng được gọi là Hệ số chính tắc. Biểu diễn này được đóng theo các phép toán cơ bản được sử dụng trong suy luận: tích nhân tố, phân chia nhân tố, giảm nhân tố và định biên. Do đó, chúng ta có thể xác định một tập hợp các cấu trúc dữ liệu đơn giản cho phép thực hiện quá trình suy luận. Hơn nữa, phép toán tích phân theo yêu cầu của định biên luôn được xác định rõ ràng và nó được đảm bảo tạo ra một tích phân hữu hạn trong các điều kiện nhất định; khi nó được xác định rõ ràng, nó có một giải pháp phân tích đơn giản.

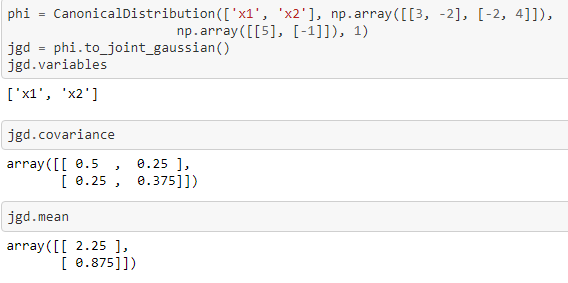
Ta có thể đại diện cho mọi Gaussian như một hình thức chính tắc. Viết lại bản pdf Gaussian chung mà chúng tôi có được,

N (μ; Σ) = C (K, h, g) khi:

Tương tự như lớp JointGaussainDistribution, lớp CanonicalFactor cũng có nguồn gốc từ lớp ContinuousFactor nhưng với các triển khai riêng của các phương thức cần thiết cho các thuật toán tổng tích hiệu quả hơn nhiều so với các phương thức của lớp cha của nó. Hãy để chúng tôi xem xét API của một vài phương thức trong lớp này.



Lớp này cũng có một phương thức, để khớp gaussian để chuyển đổi biểu diễn chính tắc trở lại phân phối gaussian chung.



**2.3.2.4** CPD Gaussian tuyến tính

Phân phối xác suất có điều kiện gaussian tuyến tính được xác định trên một biến liên tục. Tất cả phần tử cấp trên của biến này cũng liên tục. Giá trị trung bình của biến này phụ thuộc tuyến tính vào giá trị trung bình của các biến cha của nó và phương sai là độc lập.

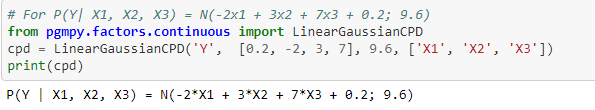
Vd như :

Gọi Y là Gaussian tuyến tính của phần tử cấp trên nó là X1, ..., Xk:

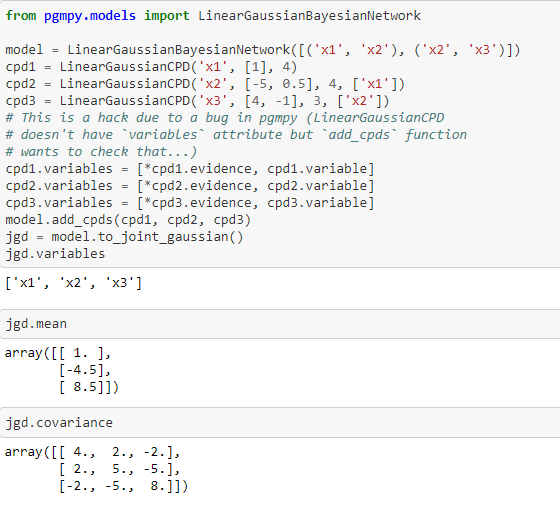
Phân phối của Y là phân phối chuẩn p (Y) trong đó:

Phân phối chung trên {X, Y} là phân phối chuẩn trong đó:

Giả sử rằng X1, ..., Xk cùng là Gaussian với phân phối N (μ; Σ). Sau đó: Đối với biểu diễn của nó, pgmpy có một lớp tên là LinearGaussianCPD trong mô-đun pgmpy.factors.continuous. Để khởi tạo một đối tượng của lớp này, người ta cần cung cấp tên biến, giá trị của số hạng beta\_0, phương sai, danh sách tên biến chính và danh sách các giá trị hệ số của phương trình tuyến tính (beta\_vector), trong đó danh sách tên biến cha và danh sách beta\_vector là tùy chọn và mặc định là Không.



Một Gaussian Bayes được định nghĩa là một mạng mà tất cả các biến của chúng là liên tục và trong đó tất cả các CPD đều là Gaussian tuyến tính. Các mạng này được quan tâm đặc biệt vì đây là một dạng đại diện thay thế của phân phối Gaussian chung. Các mạng này được triển khai dưới dạng lớp LinearGaussianBayesianNetwork trong mô-đun, pgmpy.models.continuous. Lớp này là một lớp con của lớp BayesianModel trong pgmpy.models và sẽ kế thừa hầu hết các phương thức từ nó. Nó sẽ có một phương thức đặc biệt được gọi là to\_joint\_gaussian sẽ trả về một đối tượng JointGuassianDistribution tương đương cho mô hình.



**2.4 Ước lượng**

Nhằm mục đích tìm hiểu khi nào nên dùng ước lượng nào

Giả sử rằng một tập hợp các tham số phân phối xác suất, θ, giải thích tốt nhất cho tập dữ liệu D. Chúng ta có thể ước tính các tham số θ với sự trợ giúp của Quy tắc Bayes:

**2.4.1 Ước lượng Maximum likelihood**

Với MLE, chúng tôi tìm kiếm một giá trị điểm cho θ tối đa hóa khả năng xảy ra, p (D | θ), được chỉ ra trong (các) phương trình trên. Chúng ta có thể ký hiệu giá trị này là θ ^. Trong MLE, θ ^ là một ước lượng điểm, không phải là một biến ngẫu nhiên. Nói cách khác, trong phương trình trên, MLE coi thuật ngữ p (θ) p (D) là một hằng số và không cho phép chúng ta đưa vào giá trị trước đó của mình, p (θ), về các giá trị có thể xảy ra đối với θ trong các phép tính ước lượng .

**2.4.2 Ước lượng Bayesian**

Ngược lại, ước lượng Bayes tính toán đầy đủ (hoặc đôi khi gần đúng) phân phối hậu nghiệm p (θ | D). Suy luận Bayes coi θ là một biến ngẫu nhiên. Trong ước lượng Bayes, chúng tôi đưa vào các hàm mật độ xác suất và lấy ra các hàm mật độ xác suất, thay vì một điểm như trong MLE. Trong tất cả các giá trị θ có thể tạo ra bởi phân phối đầu ra p (θ | D), nhiệm vụ của chúng ta là chọn một giá trị mà chúng ta cho là tốt nhất theo một nghĩa nào đó. Ví dụ, chúng ta có thể chọn giá trị kỳ vọng của θ với giả sử phương sai của nó đủ nhỏ. Phương sai mà chúng ta có thể tính toán cho tham số θ từ phân phối sau của nó cho phép chúng ta thể hiện sự tin tưởng của mình vào bất kỳ giá trị cụ thể nào mà chúng ta có thể sử dụng làm ước tính. Nếu phương sai quá lớn, chúng tôi có thể tuyên bố rằng không tồn tại một ước lượng tốt cho θ. Như một sự cân bằng, ước lượng Bayes được thực hiện phức tạp bởi thực tế là bây giờ chúng ta phải xử lý mẫu số trong quy tắc Bayes, tức là bằng chứng. Ở đây bằng chứng - hoặc xác suất bằng chứng - được biểu thị bằng:

**2.5 Phân tích theo cấu trúc**

Vì cấu trúc dữ liệu dùng cho đề tài là chuỗi dữ liệu dạng số, không có phân loại thời gian hay tuần tự theo chuỗi, mỗi loại bệnh phân hóa thành các triệu chứng với xác suất khác nhau. Do đó dữ liệu thuộc dạng rời rạc.

Với dữ liệu dạng rời rạc ta có sử dụng cả mô hình đồ thị có hướng hay mô hình đồ thị vô hướng. Nhưng trong trường hợp, mục tiêu cần tìm là xác suất của các triệu chứng có liên quan đến dịch tễ cần tìm, do đó ta cần dùng đồ thị có hướng. Trong trường hợp này là mạng Bayesian.

Ở điều kiện thường, pgmpy hỗ trợ cả dữ liệu liên tục và dữ liệu rời rạc.

Với tham số là toàn bộ dữ liệu các xác suất từng triệu chứng cho từng loại bệnh, mô hình Bayesian, sử dụng ước lượng maxium likelihood vì dữ liệu của mô hình thuộc dạng rời rạc nên việc lưu lại các giá trị trước không quan trọng trong bài toán này.

**TÀI LIỆU THAM KHẢO**

**Tiếng Việt**

**Tiếng Anh**

1. <https://ermongroup.github.io/cs228-notes/inference/ve/>
2. <https://github.com/pgmpy/pgmpy/blob/dev/examples/Creating%20a%20Discrete%20Bayesian%20Network.ipynb>
3. [https://www.bnlearn.com/bnrepository/discrete-small.html#asia](https://www.bnlearn.com/bnrepository/discrete-small.html%23asia)
4. <https://github.com/pgmpy/pgmpy/blob/dev/examples/Inference%20in%20Discrete%20Bayesian%20Networks.ipynb>
5. [https://www.bnlearn.com/bnrepository/discrete-medium.html#alarm](https://www.bnlearn.com/bnrepository/discrete-medium.html%23alarm)
6. <https://github.com/pgmpy/pgmpy/blob/dev/examples/Learning%20Parameters%20in%20Discrete%20Bayesian%20Networks.ipynb>
7. <https://github.com/pgmpy/pgmpy/blob/dev/examples/Structure%20Learning%20in%20Bayesian%20Networks.ipynb>
8. <https://medium.com/@Dezhic/understanding-probabilistic-graphical-models-658b6fa40184>
9. <https://medium.com/@neerajsharma_28983/intuitive-guide-to-probability-graphical-models-be81150da7a>
10. <https://blog.statsbot.co/probabilistic-graphical-models-tutorial-and-solutions-e4f1d72af189>
11. <https://blog.statsbot.co/probabilistic-graphical-models-tutorial-d855ba0107d1>
12. <https://stats.stackexchange.com/questions/74082/what-is-the-difference-in-bayesian-estimate-and-maximum-likelihood-estimate>