

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA CÔNG NGHỆ THÔNG TIN 1

----------

BÁO CÁO

NHẬP MÔN KHOA HỌC DỮ LIỆU

ĐỀ TÀI: DỰ ĐOÁN TIÊM PHÒNG VACCINE COVID-19

CỦA CÁC QUỐC GIA

|  |  |
| --- | --- |
| Giảng viên: | Vũ Hoài Nam |
| Nhóm học phần: | 03 |
| Nhóm bài tập: | 06 |

|  |  |
| --- | --- |
| Thành viên: |  |
| Trịnh Việt Anh | B20DCCN076 |
| Nguyễn Thị Thêu | B20DCCN665 |
| Phạm Văn Huy | B20DCCN323 |

Hà Nội, 2023

MỤC LỤC

[I. TỔNG QUAN DỰ ÁN 4](#_Toc151597985)

[1. Phát biểu bài toán 4](#_Toc151597986)

[II. PHÂN TÍCH CHI TIẾT 4](#_Toc151597987)

[1. Tiền xử lí dữ liệu 4](#_Toc151597988)

[III. TRỰC QUAN HÓA DỮ LIỆU 4](#_Toc151597989)

[1. Biểu đồ thể hiện mối quan hệ giữa các quốc gia và các loại vaccine 4](#_Toc151597990)

[2. Biểu đồ thể hiện số lượng tiêm chung ở các quốc gia 5](#_Toc151597991)

[3. Biếu đồ mối quan hệ giữa quốc gia, số lượng tiêm chủng và chương trình vaccine 5](#_Toc151597992)

[4. Biểu đồ thể hiện tiêm chủng hàng ngày của từng quốc gia 6](#_Toc151597993)

[5. Biểu đồ tổng số người được tiêm chủng theo quốc gia 6](#_Toc151597994)

[6. Biểu đồ tổng số mũi tiêm của từng chương trình vaccine 6](#_Toc151597995)

[7. Biểu đồ tỷ lệ tiêm chủng của từng quốc gia 7](#_Toc151597996)

[8. Số lượng tiêm chủng hàng ngày của từng quốc gia tính theo chương trình vaccine 7](#_Toc151597997)

[9. Tổng diễn biến phần trăm tiêm chủng của các quốc gia trên thang đo log 8](#_Toc151597998)

[10. Số lượng người được tiêm chủng trên một trăm ở mỗi quốc gia 8](#_Toc151597999)

[11. Diễn biến tiêm chủng hàng ngày của các quốc gia trên thang đo log 9](#_Toc151598000)

[12. Tỷ lệ phần trăm số người được tiêm chủng đầy đủ của các quốc gia trên thang đo log 9](#_Toc151598001)

[IV. MÔ HÌNH VÀ THUẬT TOÁN 10](#_Toc151598002)

[1. Mô hình Ramdom Forest 10](#_Toc151598003)

[1.1. Phương pháp 10](#_Toc151598004)

[- Thuật toán Random Forest là một phương pháp máy học dựa trên việc xây dựng nhiều cây quyết định và kết hợp kết quả của chúng để có được dự đoán cuối cùng. Dưới đây là mô tả cơ bản về cách Random Forest hoạt động: 10](#_Toc151598005)

[1.2. Thuật toán 11](#_Toc151598006)

[2. Mô hình Decision Tree 12](#_Toc151598007)

[2.1. Phương pháp 12](#_Toc151598008)

[- Thuật toán Cây Quyết Định là một phương pháp máy học dựa trên việc xây dựng một cấu trúc cây quyết định để đưa ra quyết định. Dưới đây là mô tả cơ bản về cách Cây Quyết Định hoạt động: 12](#_Toc151598009)

[2.2. Thuật toán 13](#_Toc151598010)

[V. KẾT QUẢ CỦA MÔ HÌNH 15](#_Toc151598011)

[1. Tại tệp kiểm tra là 20%: 15](#_Toc151598012)

[1.1. Mô hình Random Forest 15](#_Toc151598013)

[1.2. Mô hình Decision Tree 15](#_Toc151598014)

[2. Tại tệp kiểm tra là 30%: 16](#_Toc151598015)

[2.1. Mô hình Random Forest 16](#_Toc151598016)

[2.2. Mô hình Decision Tree 16](#_Toc151598017)

[VI. ĐÁNH GIÁ MÔ HÌNH 16](#_Toc151598018)

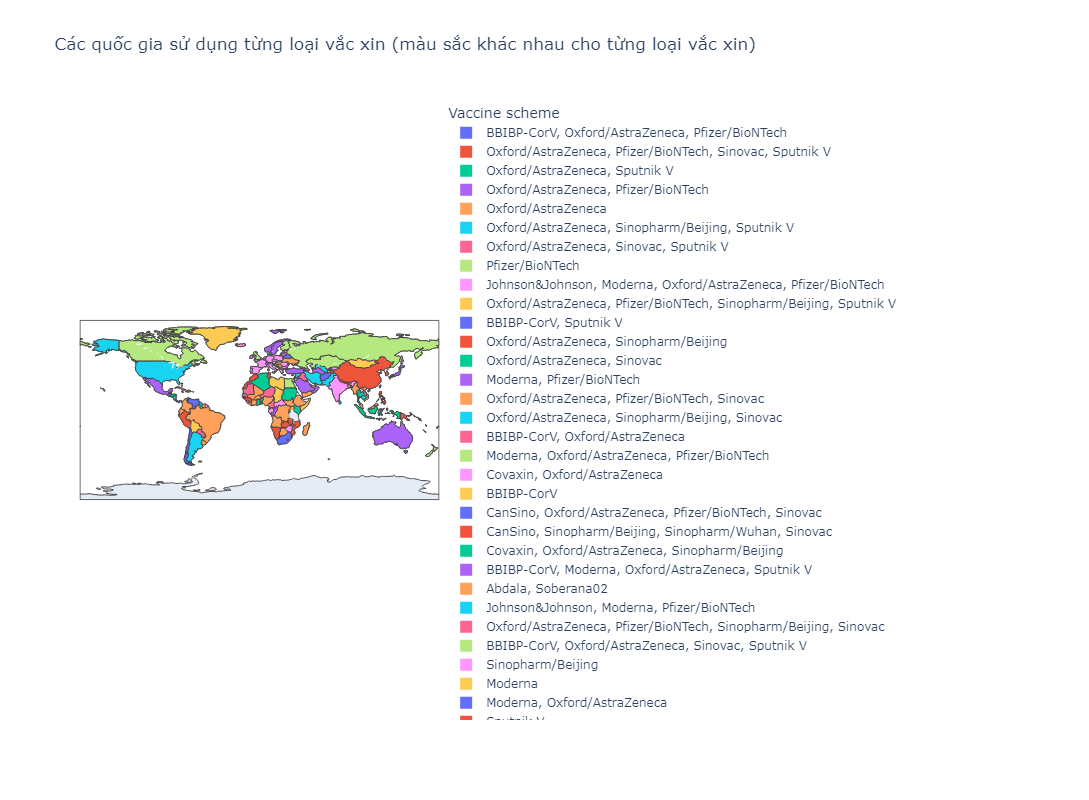
1. TỔNG QUAN DỰ ÁN
2. Phát biểu bài toán

* Dự đoán tiêm phòng vaccine covid-19 các quốc gia trên thế giới được thu thập từ nhiều nguồn khác nhau trong khoảng thời gian từ 01/02/2021 đến 22/06/2021 thực hiện với tập dữ liệu gồm 25863 hàng gồm có các thông tin sau:
  + Country: Quốc gia cung cấp thông tin tiêm chủng
  + ISO\_Code: Mã ISO của quốc gia
  + Date: Ngày tiêm
  + Total\_vaccinations: Tổng số mũi tiêm
  + People\_vaccinated: Tổng số người đã được tiêm chủng
  + People\_fully\_vaccinated: Tổng số người được tiêm phỏng đầy đủ
  + Daily\_vaccinations\_raw: Tiêm chủng hàng ngày (thô)
  + Daily\_vaccinations: Tiêm chủng hàng ngày
  + Total\_vaccinations\_per\_hundred: Tỷ lệ (tính bằng phần trăm) giữa số lượng tiêm chủng và tổng dân số tính đến thời điểm hiện tại trong cả nước
  + People\_vaccinated\_per\_hundred: tỷ lệ (tính bằng phần trăm) giữa dân số được tiêm chủng và tổng dân số tính đến thời điểm hiện tại của cả nước
  + People\_fully\_vaccinated\_per\_hundred: tỷ lệ (tính bằng phần trăm) giữa dân số được tiêm chủng đầy đủ và tổng dân số tính đến thời điểm hiện tại trong nước
  + Daily\_vaccinations\_per\_million: tỷ lệ tiêm chủng hàng ngày trên một triệu người
  + Vaccines: Loại vaccine
  + Source name: nguồn thông tin (cơ quan quốc gia, tổ chức quốc tế, tổ chức địa phương, v.v.)
  + Source website: trang web chứa nguồn thông tin

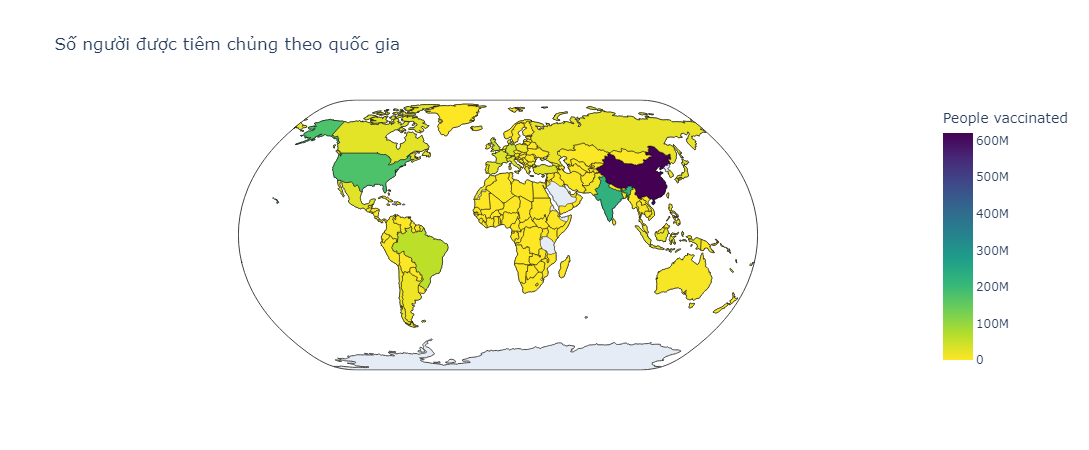
1. PHÂN TÍCH CHI TIẾT
2. Tiền xử lí dữ liệu

* Do thông tin từ trang web nguồn không có ảnh hưởng quan trọng đến mô hình dự đoán số liều tiêm hàng ngày nên giảm chiều dữ liệu bằng cách loại bỏ cột “source\_website”
* Dữ liệu sau khi xử lí vẫn còn 25863 dòng dữ liệu

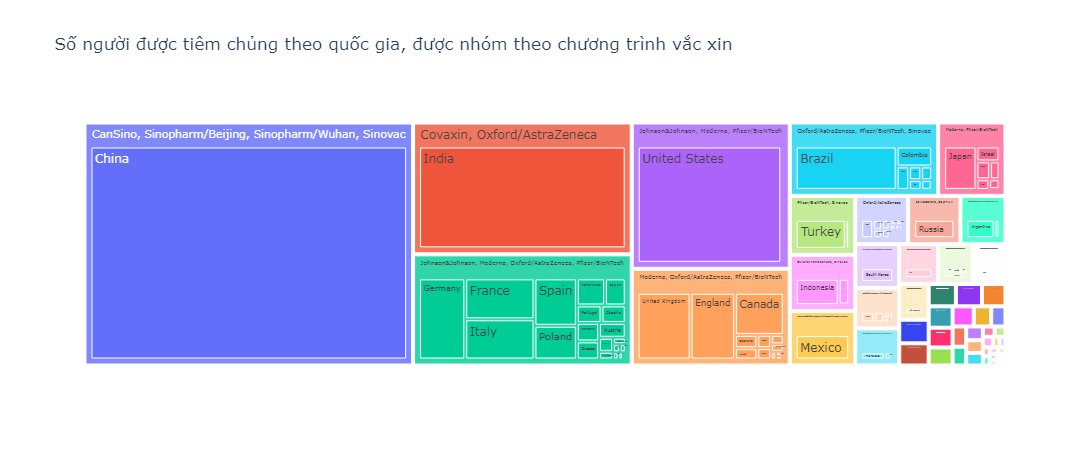
1. TRỰC QUAN HÓA DỮ LIỆU
2. Biểu đồ thể hiện mối quan hệ giữa các quốc gia và các loại vaccine



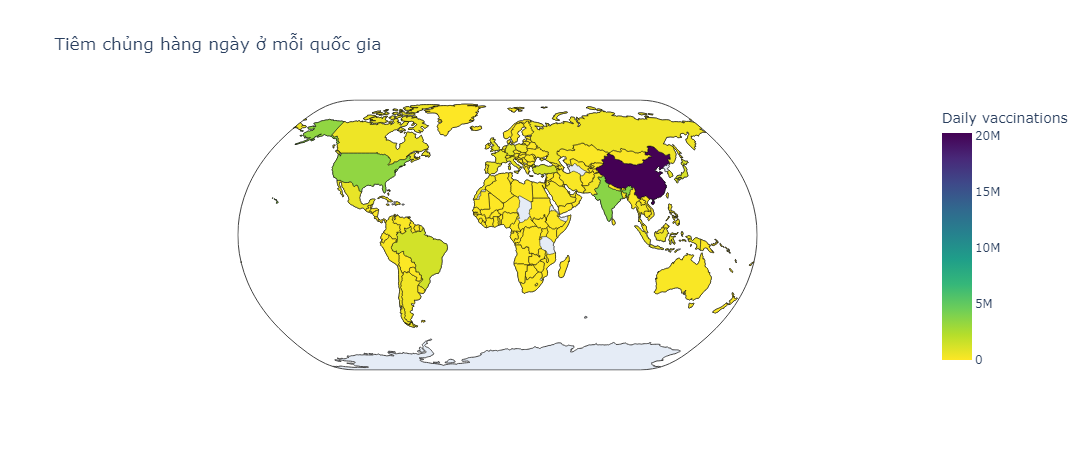
1. Biểu đồ thể hiện số lượng tiêm chung ở các quốc gia



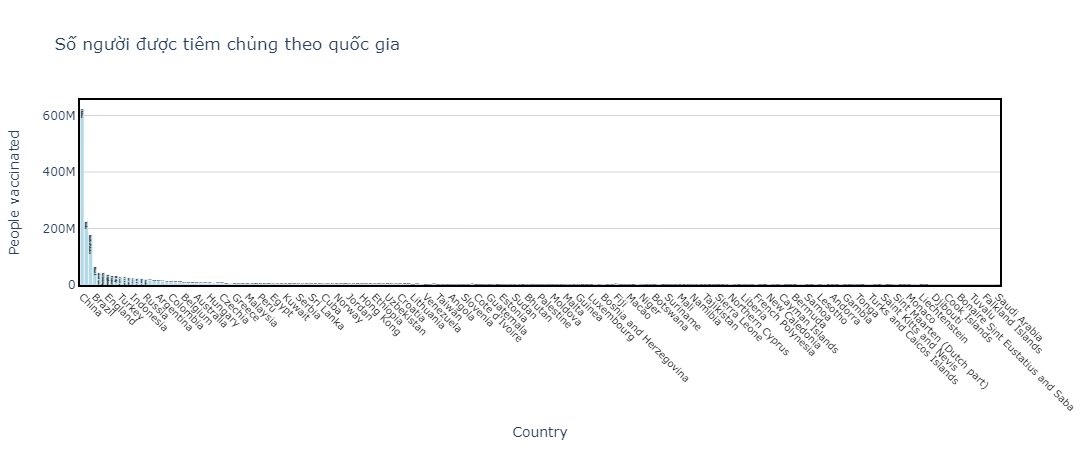
1. Biếu đồ mối quan hệ giữa quốc gia, số lượng tiêm chủng và chương trình vaccine



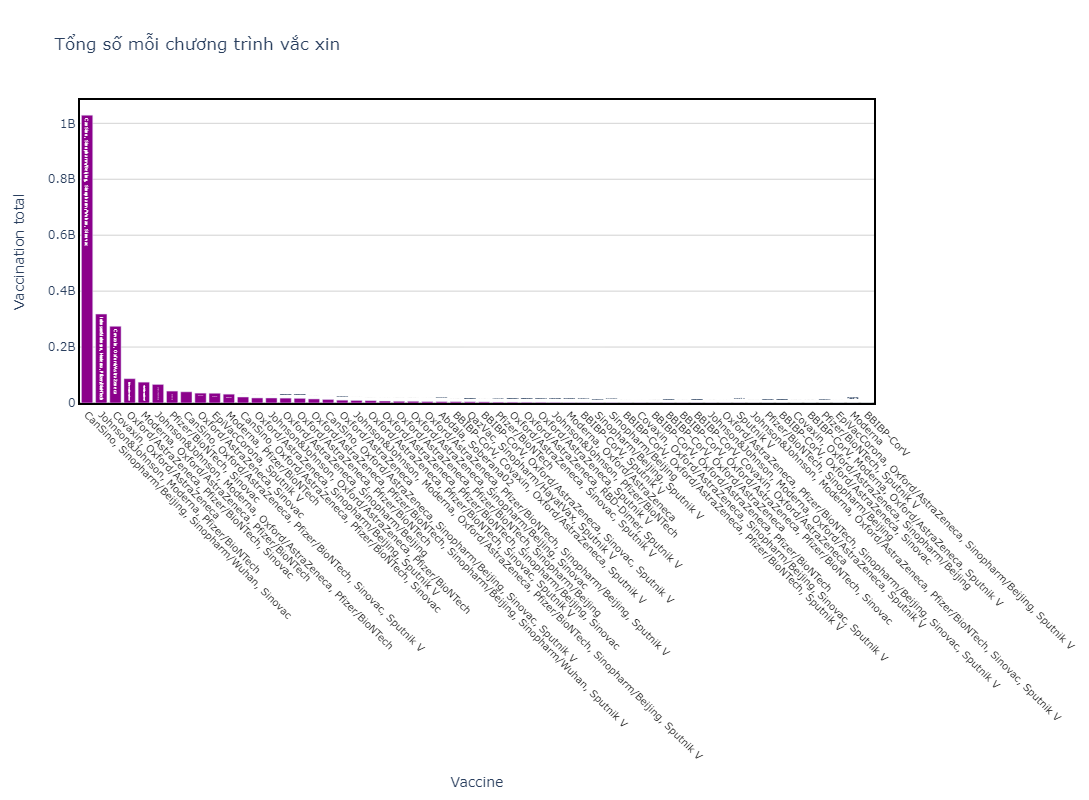
1. Biểu đồ thể hiện tiêm chủng hàng ngày của từng quốc gia



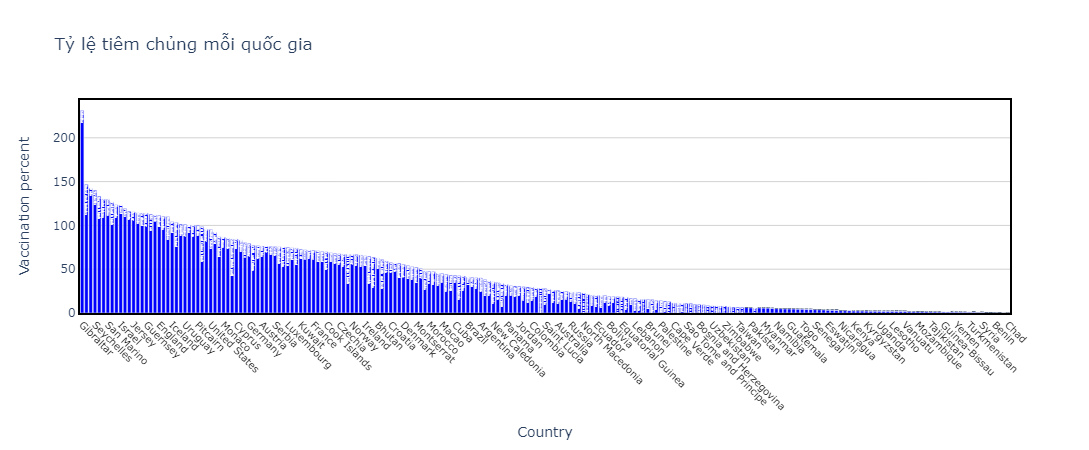
1. Biểu đồ tổng số người được tiêm chủng theo quốc gia



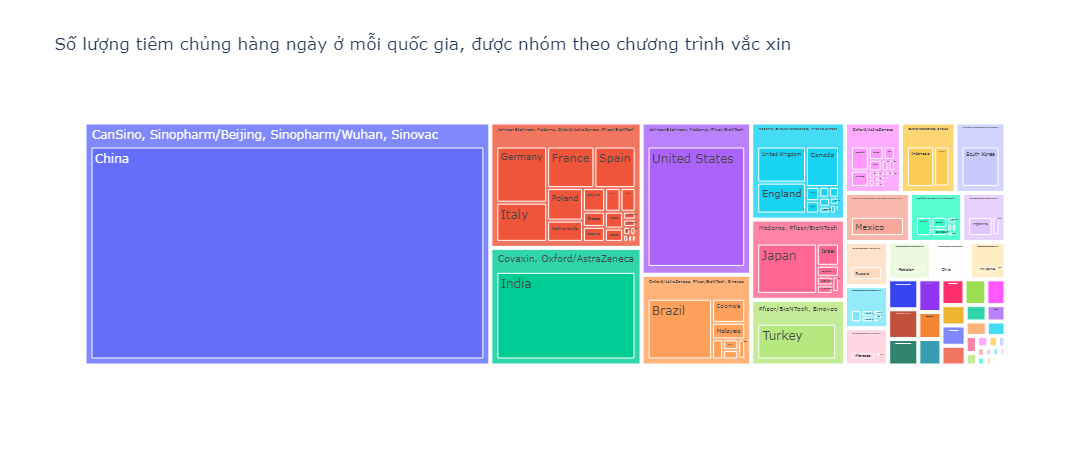
1. Biểu đồ tổng số mũi tiêm của từng chương trình vaccine



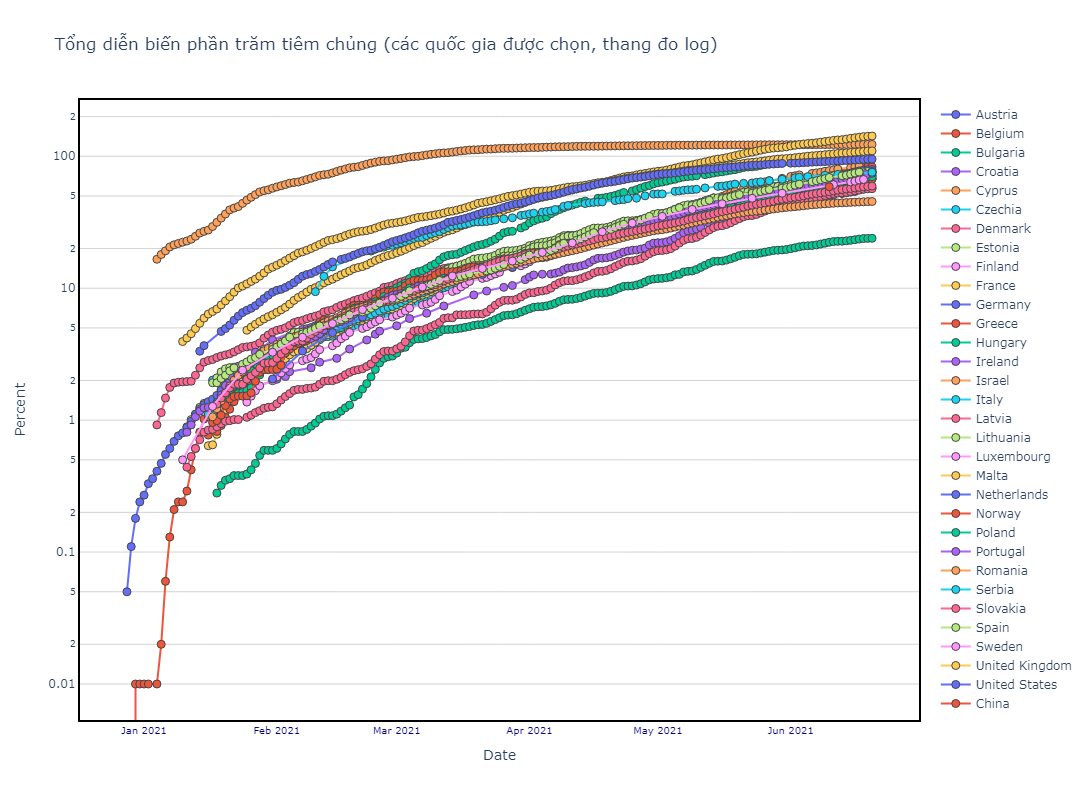
1. Biểu đồ tỷ lệ tiêm chủng của từng quốc gia



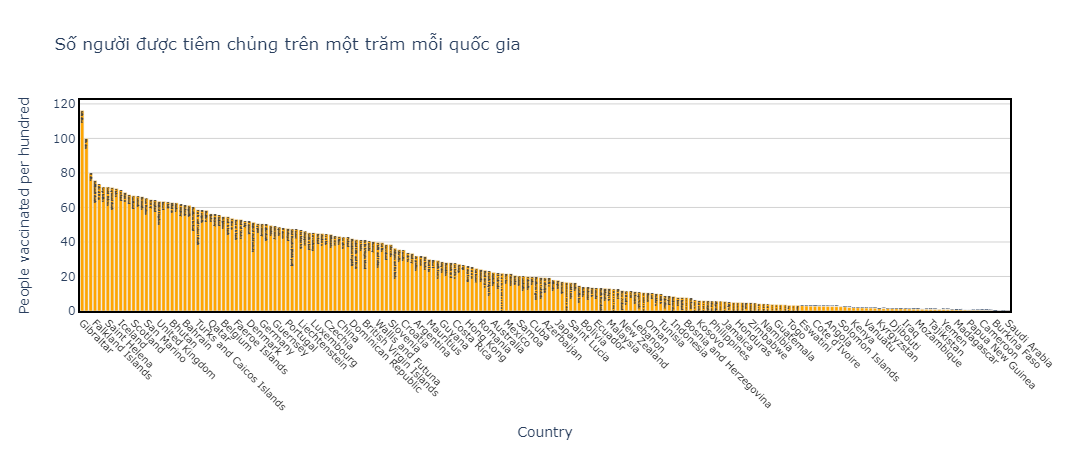
1. Số lượng tiêm chủng hàng ngày của từng quốc gia tính theo chương trình vaccine



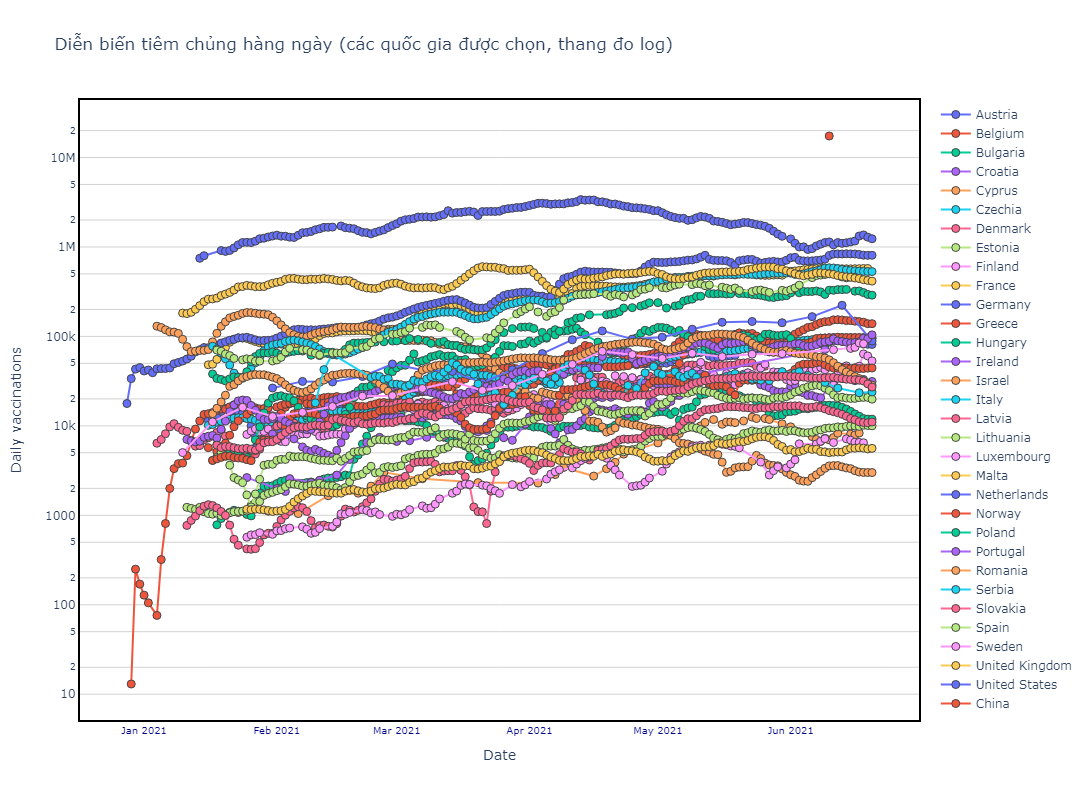
1. Tổng diễn biến phần trăm tiêm chủng của các quốc gia trên thang đo log



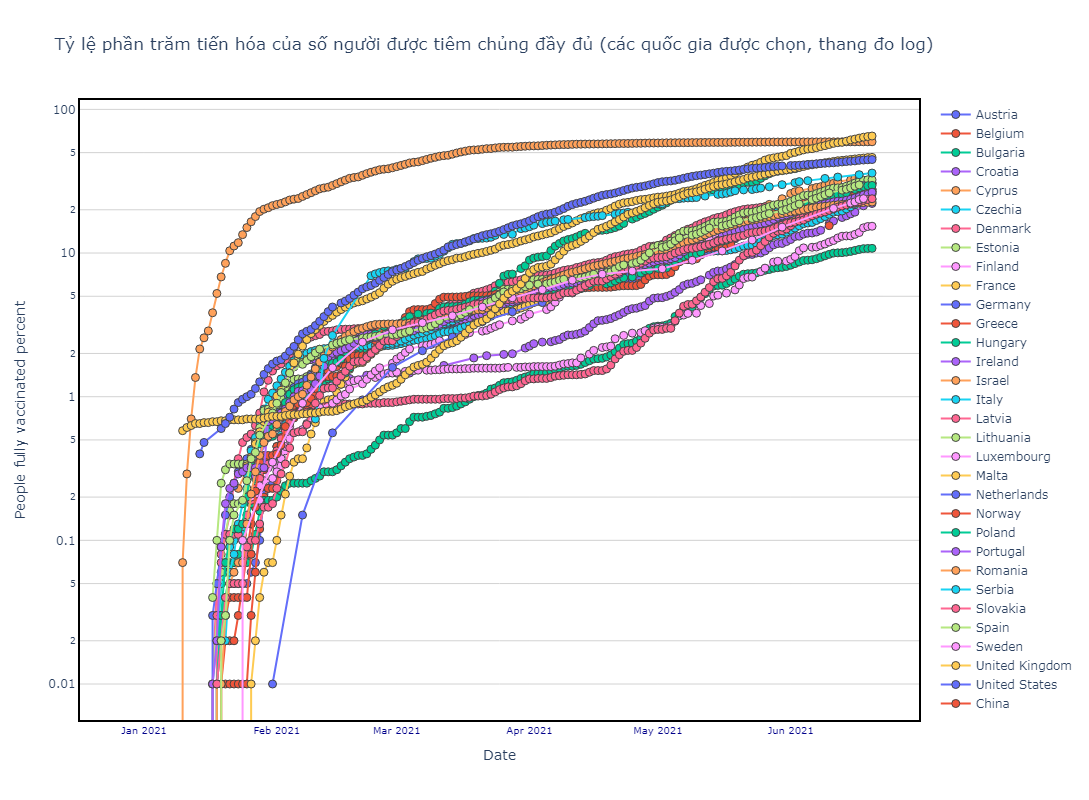
1. Số lượng người được tiêm chủng trên một trăm ở mỗi quốc gia



1. Diễn biến tiêm chủng hàng ngày của các quốc gia trên thang đo log



1. Tỷ lệ phần trăm số người được tiêm chủng đầy đủ của các quốc gia trên thang đo log



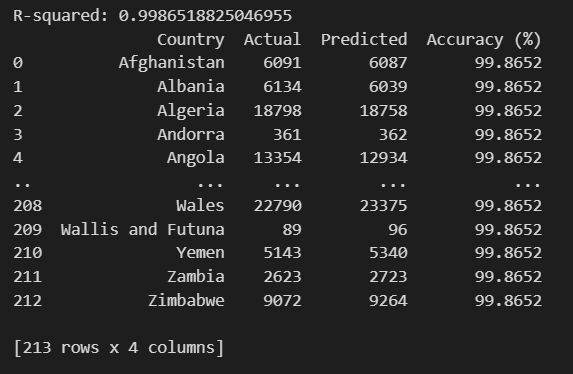
1. MÔ HÌNH VÀ THUẬT TOÁN
2. Mô hình Ramdom Forest
   1. Phương pháp

* Thuật toán Random Forest là một phương pháp máy học dựa trên việc xây dựng nhiều cây quyết định và kết hợp kết quả của chúng để có được dự đoán cuối cùng. Dưới đây là mô tả cơ bản về cách Random Forest hoạt động:
* Lựa chọn dữ liệu ngẫu nhiên: Random Forest sử dụng một mẫu ngẫu nhiên từ tập dữ liệu huấn luyện để xây dựng từng cây quyết định. Điều này được thực hiện bằng cách lựa chọn một phần nhỏ của dữ liệu (có thể có hoặc không có replacement) để huấn luyện mỗi cây.
* Xây dựng nhiều cây quyết định: Mỗi cây quyết định được xây dựng độc lập từ dữ liệu ngẫu nhiên. Quá trình này giúp mỗi cây học một cách đặc trưng khác nhau của dữ liệu và giảm nguy cơ overfitting.
* Chia nhỏ tập dữ liệu: Tại mỗi nút trong mỗi cây, một thuộc tính và giá trị ngưỡng được chọn để chia tập dữ liệu thành hai phần. Quá trình này được thực hiện bằng cách tối ưu hóa một hàm mất mát như Gini impurity (đối với bài toán phân loại) hoặc phương sai (đối với bài toán hồi quy).
* Dự đoán: Khi cây đã được xây dựng, mỗi cây đưa ra dự đoán độc lập. Trong trường hợp phân loại, dự đoán cuối cùng là kết quả của việc bầu chọn dự đoán của tất cả các cây. Trong trường hợp hồi quy, dự đoán cuối cùng là giá trị trung bình của dự đoán từ tất cả các cây.
* Đánh giá và tính chất linh hoạt: Random Forest thường có khả năng tự động đánh giá độ quan trọng của các đặc trưng và có thể xử lý cả các dữ liệu có nhiễu. Nó cũng có thể giảm nguy cơ overfitting so với một cây quyết định đơn lẻ.
  1. Thuật toán

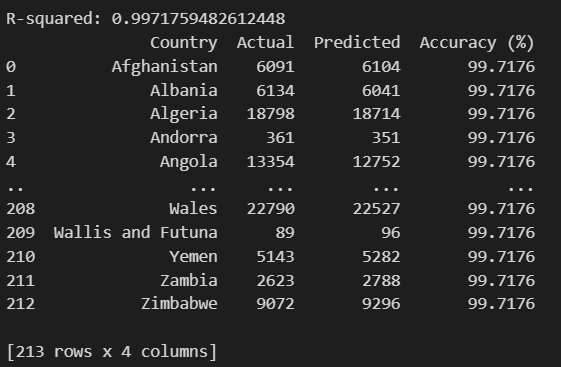
1. #Thuật toán rừng cây
2. import pandas as pd
3. from sklearn.model\_selection import train\_test\_split
4. from sklearn.ensemble import RandomForestRegressor
5. from sklearn.metrics import mean\_squared\_error, r2\_score
6. from sklearn.preprocessing import LabelEncoder
7. from sklearn.compose import ColumnTransformer
8. from sklearn.pipeline import Pipeline
9. from sklearn.impute import SimpleImputer
10. # Đọc dữ liệu từ file CSV
11. data = pd.read\_csv('country\_vaccinations.csv')
12. # Chọn các cột quan trọng để dự đoán
13. selected\_columns = ['date', 'country', 'vaccines', 'daily\_vaccinations']
14. # Lọc dữ liệu
15. data = data[selected\_columns]
16. # Xử lý dữ liệu ngày thành dạng số
17. data['date'] = pd.to\_datetime(data['date'])
18. data['date'] = data['date'].dt.dayofyear
19. # Xử lý dữ liệu văn bản (loại vaccine và quốc gia) thành dạng số
20. le = LabelEncoder()
21. data['vaccines'] = le.fit\_transform(data['vaccines'])
22. data['country'] = le.fit\_transform(data['country'])
23. # Loại bỏ các dòng có giá trị thiếu
24. data = data.dropna()
25. # Tách dữ liệu thành features (X) và target (y)
26. X = data.drop('daily\_vaccinations', axis=1)
27. y = data['daily\_vaccinations']
28. # Chia dữ liệu thành tập huấn luyện và tập kiểm tra
29. X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)
30. # Xây dựng pipeline cho việc xử lý dữ liệu và huấn luyện mô hình
31. pipeline = Pipeline([
32. ('imputer', SimpleImputer(strategy='mean')),
33. ('regressor', RandomForestRegressor(n\_estimators=100, random\_state=42))
34. ])
35. # Huấn luyện mô hình
36. pipeline.fit(X\_train, y\_train)
37. # Dự đoán số liều tiêm hàng ngày cho từng quốc gia
38. predictions = pipeline.predict(X\_test)
39. # Đánh giá độ chính xác bằng hệ số xác định R^2
40. r2 = r2\_score(y\_test, predictions)
41. print(f'R-squared: {r2}')
42. # Tạo DataFrame mới với tên quốc gia và giá trị dự đoán
43. result\_df = pd.DataFrame({'Country': le.inverse\_transform(X\_test['country']), 'Actual': y\_test, 'Predicted': predictions})
44. # Làm tròn giá trị dự đoán và thực tế thành số nguyên
45. result\_df['Predicted'] = result\_df['Predicted'].round().astype(int)
46. result\_df['Actual'] = result\_df['Actual'].round().astype(int)
47. # Tính phần trăm chính xác và làm tròn cột mới về 4 chữ số thập phân
48. accuracy\_percentage = r2 \* 100
49. result\_df['Accuracy (%)'] = round(accuracy\_percentage, 4)
50. # Nhóm theo tên quốc gia và tính giá trị trung bình
51. result\_df\_grouped = result\_df.groupby('Country').mean().reset\_index()
52. # Ghi kết quả vào file CSV
53. result\_df\_grouped.to\_csv('final\_random\_forest.csv', index=False)
54. # Làm tròn giá trị của 'Actual' và 'Predicted' trong DataFrame result\_df\_grouped
55. result\_df\_grouped[['Actual', 'Predicted']] = result\_df\_grouped[['Actual', 'Predicted']].round().astype(int)
56. # Hiển thị DataFrame mới
57. print(result\_df\_grouped)
58. Mô hình Decision Tree
    1. Phương pháp

* Thuật toán Cây Quyết Định là một phương pháp máy học dựa trên việc xây dựng một cấu trúc cây quyết định để đưa ra quyết định. Dưới đây là mô tả cơ bản về cách Cây Quyết Định hoạt động:
* Chia nhỏ tập dữ liệu: Tại mỗi nút trong cây, thuật toán chọn một thuộc tính và giá trị ngưỡng để chia tập dữ liệu thành hai phần. Quá trình này được thực hiện bằng cách tối ưu hóa một hàm mất mát như Gini impurity (đối với bài toán phân loại) hoặc phương sai (đối với bài toán hồi quy).
* Xây dựng cây: Cây được xây dựng bằng cách lặp lại quá trình chia nhỏ tập dữ liệu cho đến khi một điều kiện dừng được đạt đến, chẳng hạn như độ sâu tối đa của cây hoặc số lượng mẫu tối thiểu tại mỗi lá.
* Dự đoán: Khi một mẫu mới được đưa vào cây, nó đi qua các nút từ gốc đến lá dựa trên giá trị của các thuộc tính. Dự đoán cuối cùng là kết quả của lá mà mẫu đó rơi vào.
* Đánh giá và linh hoạt: Cây Quyết Định có khả năng dễ giải thích do cấu trúc cây dễ hiểu. Nó cũng có thể tự động đánh giá độ quan trọng của các đặc trưng và xử lý dữ liệu có nhiễu. Tuy nhiên, cây quyết định có nguy cơ overfitting, và việc sử dụng kỹ thuật như cây ngẫu nhiên (Random Forest) có thể giúp giảm thiểu vấn đề này.
* Điều chỉnh tham số: Cây Quyết Định thường có các tham số như độ sâu tối đa, số lượng mẫu tối thiểu tại lá, và các tham số điều khiển quá trình chia nhỏ. Điều chỉnh các tham số này có thể ảnh hưởng đến hiệu suất của mô hình.
  1. Thuật toán

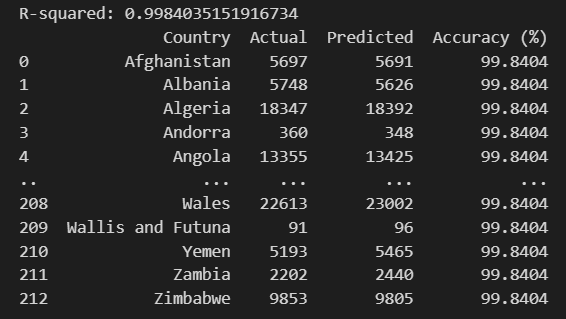
1. #Thuật toán cây quyết định
2. import pandas as pd
3. from sklearn.model\_selection import train\_test\_split
4. from sklearn.tree import DecisionTreeRegressor
5. from sklearn.metrics import mean\_squared\_error, r2\_score
6. from sklearn.preprocessing import LabelEncoder
7. from sklearn.compose import ColumnTransformer
8. from sklearn.pipeline import Pipeline
9. from sklearn.impute import SimpleImputer
10. # Đọc dữ liệu từ file CSV
11. data = pd.read\_csv('country\_vaccinations.csv')
12. # Chọn các cột quan trọng để dự đoán
13. selected\_columns = ['date', 'country', 'vaccines', 'daily\_vaccinations']
14. # Lọc dữ liệu
15. data = data[selected\_columns]
16. # Xử lý dữ liệu ngày thành dạng số
17. data['date'] = pd.to\_datetime(data['date'])
18. data['date'] = data['date'].dt.dayofyear
19. # Xử lý dữ liệu văn bản (loại vaccine và quốc gia) thành dạng số
20. le = LabelEncoder()
21. data['vaccines'] = le.fit\_transform(data['vaccines'])
22. data['country'] = le.fit\_transform(data['country'])
23. # Loại bỏ các dòng có giá trị thiếu
24. data = data.dropna()
25. # Tách dữ liệu thành features (X) và target (y)
26. X = data.drop('daily\_vaccinations', axis=1)
27. y = data['daily\_vaccinations']
28. # Chia dữ liệu thành tập huấn luyện và tập kiểm tra
29. X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)
30. # Xây dựng pipeline cho việc xử lý dữ liệu và huấn luyện mô hình
31. pipeline = Pipeline([
32. ('imputer', SimpleImputer(strategy='mean')),
33. ('regressor', DecisionTreeRegressor())  # Thay đổi thành cây quyết định
34. ])
35. # Huấn luyện mô hình
36. pipeline.fit(X\_train, y\_train)
37. # Dự đoán số liều tiêm hàng ngày cho từng quốc gia
38. predictions = pipeline.predict(X\_test)
39. # Đánh giá độ chính xác bằng hệ số xác định R^2
40. r2 = r2\_score(y\_test, predictions)
41. print(f'R-squared: {r2}')
42. # Tạo DataFrame mới với tên quốc gia và giá trị dự đoán
43. result\_df = pd.DataFrame({'Country': le.inverse\_transform(X\_test['country']), 'Actual': y\_test, 'Predicted': predictions})
44. # Làm tròn giá trị dự đoán và thực tế thành số nguyên
45. result\_df['Predicted'] = result\_df['Predicted'].round().astype(int)
46. result\_df['Actual'] = result\_df['Actual'].round().astype(int)
47. # Tính phần trăm chính xác và làm tròn cột mới về 4 chữ số thập phân
48. accuracy\_percentage = r2 \* 100
49. result\_df['Accuracy (%)'] = round(accuracy\_percentage, 4)
50. # Nhóm theo tên quốc gia và tính giá trị trung bình
51. result\_df\_grouped = result\_df.groupby('Country').mean().reset\_index()
52. # Ghi kết quả vào file CSV
53. result\_df\_grouped.to\_csv('final\_decision\_tree.csv', index=False)
54. # Làm tròn giá trị của 'Actual' và 'Predicted' trong DataFrame result\_df\_grouped
55. result\_df\_grouped[['Actual', 'Predicted']] = result\_df\_grouped[['Actual', 'Predicted']].round().astype(int)
56. # Hiển thị DataFrame mới
57. print(result\_df\_grouped)
58. KẾT QUẢ CỦA MÔ HÌNH
59. Tại tệp kiểm tra là 20%:
    1. Mô hình Random Forest



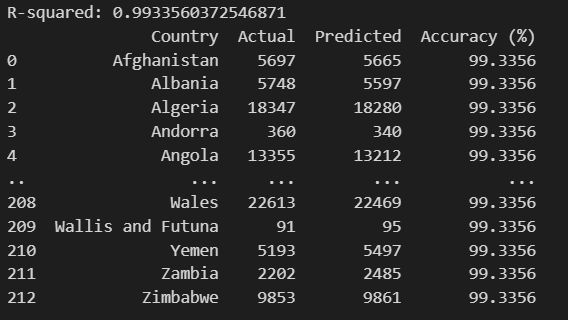
* 1. Mô hình Decision Tree



1. Tại tệp kiểm tra là 30%:
   1. Mô hình Random Forest



* 1. Mô hình Decision Tree



1. ĐÁNH GIÁ MÔ HÌNH

Từ kết quả thu được qua các lần thử nghiệm ta kết luận được như sau:

* Mô hình Random Forest có độ chính xác thuật toán cao hơn Decision Tree, dẫu vậy sự khác biệt là không quá lơn ở cả 2 trường hợp và chỉ số của từng thuật toán thay đổi ở mức nhỏ cho thấy thuật toán có độ chính xác cao.
* Chỉ số đô chính xác của 2 thuật toán trong 2 trường hợp đều có giá trị gần với 1 cho thấy cả 2 thuật toán phù hợp cho tệp dữ liệu đã cho và cũng phù hợp để đánh giá mô hình dự đoán số mũi tiêm vaccine covid-19 hàng ngày.