ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

——————oOo——————

ĐỒ ÁN MÔN HỌC

PHÂN TÍCH KHÁM PHÁ DỮ LIỆU

Học viên thực hiện:

BÙI TẤT HIỆP 22C01007

Ngành: Khoa học dữ liệu - K32

TP. HỒ CHÍ MINH - 2023
ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

——————oOo——————

ĐỒ ÁN MÔN HỌC

PHÂN TÍCH KHÁM PHÁ DỮ LIỆU

Học viên thực hiện:

BÙI TẤT HIỆP 22C01007

Ngành Khoa học dữ liệu - K32

TP. HỒ CHÍ MINH - 2023
ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

Mục lục

Phần mở đầu

Có một câu ngạn ngữ rằng: "Một bức tranh đáng giá hơn ngàn câu nói". Thực vậy, để có thể hiểu được một đoạn văn, một cuốn sách thì con người cần đọc hiểu và nắm được ý chính của chúng. Trong lĩnh vực phân tích dữ liệu, khoa học dữ liệu hay phân tích thống kê, để có một cái nhìn tổng thể về vấn đề đang xử lý, hay trước khi thực hiện các bước xây dựng mô hình nhằm phân tích, dự báo chuyên sâu, trực quan hóa dữ liệu (visualization) là một bước thực hiện có vai trò quan trọng - vẽ nên bức tranh tổng thể của vấn đề; chúng được xem là một cách hiệu quả để trình bày về dữ liệu và thông tin, bằng cách sử dụng các yếu tố hình ảnh như biểu đồ, hình vẽ, bản đồ,...

Giống như những bức tranh vẽ, trước khi bức tranh được hoàn thiện chi tiết từng chi tiết nhỏ thì cần có một bố cục tổng thể; một vấn đề trước khi bước vào phân tích thì cần có một cái nhìn tổng quát. Vì vậy, đồ án được thực hiện với mục đích mô phỏng lại quá trình EDA cho một tập dữ liệu khá phổ biến - Titanic. Đồ án được thực hiện dựa trên sự tìm hiểu và hiểu biết của học viên trong quá trình học, vì vậy không thể tránh khỏi việc sai sót, học viên mong người đọc có thể chỉ ra sai sót để có thể hiểu hơn về kiến thức. Qua đó, học viên xin chân thành gửi lời cám ơn các Quý thầy đã hỗ trợ và truyền dạy kiến thức trong môn học này.

Xin chân thành cám ơn!

Chương 1

Kiến thức chuẩn bị

1.1 Cases, variables và các loại thang đo

Một bộ dữ liệu (dataset) hoàn chỉnh là gồm có 2 thành phần Case và Variable, trong đó Case là một đại lượng chỉ đến từng cá thể trong bộ dữ liệu (dataset), case cũng có thể được gọi là observation (quan sát), là việc quan sát và ghi nhận cho một sự vật, sự việc hay một người nào đó; trong khi đó, variable (biến) hướng đến tính chất, tính năng của sự vật, hiện tượng hay con người đó. Nếu biến là một đại lượng không thay đổi thì nó được gọi là hằng số (constant).
Biến được chia thành 02 loại, cụ thể là: Biến định tính (Qualitative variable) và biến định lượng (Quantitative variable). Biến định tính là các biến dùng để chỉ các danh mục (theo tên gọi hoặc được dán nhãn), chúng được phân loại mà không có ý nghĩa về mặt thứ tự. Biến định lượng là các đại lượng biểu hiện dưới dạng số học, chúng có ý nghĩa về mặt thứ tự, có giá trị thực; trong đó biến định lượng có thể tiếp tục phân thành 02 loại: biến rời rạc (discrete) và biến liên tục (continuoues).
Thang đo (level of measurement): Trong thống kê, chúng ta phân loại các dạng dữ liệu trên thành 04 loại thang đo. Đây là các loại thang đo dùng để phân loại các phân tích thống kê, ảnh hưởng và tác động đến dữ liệu, vì vậy việc hiểu rõ và vận dụng tốt sẽ hỗ trợ cho việc phân tích thống kê mô tả. Cụ thể gồm:

- Thang đo định danh (Nominal Scale);
- Thang đo thứ bậc (Ordinal Scale);
- Thang đo khoảng (Interval Scale);
- Thang đo tỉ lệ (Ratio Scale).

1.2 Các đại lượng trong thống kê

1.2.1 Đo lường xu hướng trung tâm (Measures of central tendency)

Yếu vị (Mode)

Mode là giá trị số lần xuất hiện nhiều nhất trong tệp dữ liệu. Giá trị mode thường được sử dụng như một đại lượng đo lường trung tâm với thang đo định danh và thang đo thứ bậc. Một biến có thể có nhiều Mode.

Số trung vị (Median)

Số trung vị là giá trị chính giữa của các quan sát của dữ liệu khi chúng ta sắp xếp chúng theo giá trị tăng dần hoặc giảm dần.

Khi có n số phần tử lẻ:

$$Median = \left(\frac{n+1}{2}\right)^{th} obs. \tag{1.1}$$

Khi có n số phần tử chẵn:

$$Median = \frac{\frac{n^{th}}{2} obs. + \left(\frac{n}{2}+1\right)^{th} obs.}{2} \tag{1.2}$$

Số trung bình (Mean)

Số trung bình được tính bằng cách lấy tổng giá trị chia cho số phần tử quan sát. Công thức được tính như sau:

$$\overline{X} = \frac{1}{n}\sum_{i}^{n} X_i \tag{1.3}$$

Ba đại lượng trên được sử dụng thông dụng giúp ta hiểu được độ tập trung của dữ liệu. Mode có thể sử dụng khi đó là biến định tính, trong khi Mean và Median có thể sử dụng với biến định lượng. Thông thường, số trung bình và số trung vị có khuynh hướng gần xấp xỉ bằng nhau. Tuy nhiên, trong trường hợp tệp dữ liệu có xuất hiện dữ liệu ngoại lai (outlier) hoặc dữ liệu phân bố không đều, lúc này phân phối dữ liệu sẽ bị lệch đi (lệch trái hoặc phải); giá trị trung bình và trung vị sẽ không còn xấp xỉ nhau nữa. Lúc này ta nên chọn giá trị trung vị để tìm độ tập trung dữ liệu.

1.2.2 Độ phân tán dữ liệu (Measures of dispersion)

Khoảng biến thiên (Range)

Khoảng biến thiên là một độ đo đơn giản thể hiện sự khác biệt giữa các giá trị trong tệp dữ liệu, được tính toán bằng cách lấy giá trị lớn nhất trừ đi giá trị nhỏ nhất (khi đã loại bỏ các dữ liệu ngoại lai).

$$Range(X) = Max(X) - Min(X) \tag{1.4}$$

Độ trải giữa (Interquartile range)

Độ trải giữa hay còn gọi là khoảng tứ phân vị, là một độ đo nhằm đo lường mức độ phân tán của tập dữ liệu. Chúng chia tệp dữ liệu trên thành các khoảng bằng nhau, có 4 ngưỡng và chúng được gọi là các phân vị, như vậy có 4 phân vị được sắp xếp từ nhỏ tới lớn và được ký hiệu là: Q1, Q2, Q3, Q4; trong đó Q2 chính là giá trị trung vị (median) của tệp dữ liệu. Độ trải giữa được tính bằng cách lấy giá trị tứ phân vị thứ 3 trừ đi tứ phân vị thứ 1 (sau khi đã được sắp xếp theo thứ tự tăng dần).

Phương sai (Variance) và Độ lệch chuẩn (Standard Deviation)

Phương sai là một độ đo nhằm đo lường sự thay đổi, phương sai được tính bằng công thức sau (đây là công thức áp dụng cho việc tính toán trên một mẫu).

$$s^2 = \frac{\sum (x_i - \overline{x})^2}{n-1} \tag{1.5}$$

Độ lệch chuẩn cho biết giá trị dữ liệu đã sai lệch đi bao nhiêu so với giá trị trung bình (mean). Nếu giá trị độ lệch chuẩn càng nhỏ, dữ liệu càng có xu hướng tập trung quanh giá trị trung bình, và ngược lại. Độ lệch chuẩn là căn bậc hai của phương sai. Độ lệch chuẩn của 1 mẫu được tính như sau:

$$s = \sqrt{\frac{\sum (x_i - \overline{x})^2}{n-1}} \tag{1.6}$$

Hệ số biến thiên (Coefficient of variation)

Hệ số biến thiên là tỉ lệ của độ lệch chuẩn so sánh với giá trị trung bình cộng:

$$CV = \frac{s}{\overline{x}} \tag{1.7}$$

Dữ liệu ngoại lai (outlier)

Dữ liệu ngoại lai là các quan sát có giá trị nằm ở xa một cách bất thường với các dữ liệu khác trong tệp dữ liệu. Các dữ liệu được xem là dữ liệu ngoại lai khi:

- Nhỏ hơn ngưỡng $Q1 - 1.5 \times (IQR)$
- Lớn hơn ngưỡng $Q3 + 1.5 \times (IQR)$

1.2.3 Hiệp phương sai (Covariance)

Hiệp phương sai là của 2 biến ngẫu nhiên X và Y nhằm xác định mỗi quan hệ biến thiên giữa chúng. Nếu $Cov > 0$, hai biến có xu hướng thay đổi đồng biến và ngược lại, nghịch biến nếu $Cov < 0$. Trường hợp $Cov = 0$ thì X và Y độc lập. Hiệp phương sai có thể được biểu diễn như sau:

$$Cov[X, Y] \triangleq \mathrm{E}[(X - \mathrm{E}[X])(Y - \mathrm{E}[Y])] = \mathrm{E}[XY] - \mathrm{E}[X]\,\mathrm{E}[Y] \tag{1.8}$$

1.2.4 Hệ số tương quan (Correlation)

Hệ số tương quan nhằm giải thích tác động đối với biến Y khi giá trị đại lượng X thay đổi. Trong đó hệ số tương quan Pearson (Pearson's correlation coefficient) được định nghĩa như sau:

$$\rho \triangleq corr\,[X,Y] \triangleq \frac{Cov\,[X,Y]}{s_X s_Y} \tag{1.9}$$

Hệ số tương quan Pearson (ρ) có giá trị dao động trong khoảng $[-1,1]$ với ý nghĩa như sau:

- $\rho = 0$: Hai biến không có tương quan tuyến tính.
- $\rho = -1; \rho = 1$: Hai biến có mối tương quan tuyến tính tuyệt đối.
- $\rho < 0$: Tương quan âm, khi X tăng thì Y giảm và ngược lại.
- $\rho > 0$: Tương quan dương, khi X tăng thì Y cùng tăng và ngược lại.
- Nếu $\rho \in [-1,-0.5]$ hoặc $\rho \in [0.5,1]$: Hai biến tương quan mạnh.
- Nếu $\rho \in [-0.49,-0.3]$ hoặc $\rho \in [0.3,0.49]$: Hai biến tương quan bình thường.
- Nếu $\rho \in [-0.29,0.29]$: Hai biến tương quan yếu hoặc không tương quan (khi $\rho = 0$).

Lưu ý: Việc hai biến tương quan với nhau không có nghĩa việc xảy ra ở biến này là nguyên nhân khiến biến kia thay đổi.

Chương 2

Phân tích khám phá dữ liệu

2.1 Phân tích khám phá dữ liệu - Exploratory Data Analysis (EDA)

2.1.1 Khái niệm về EDA

Phân tích khám phá dữ liệu (EDA) được sử dụng nhằm phân tích và điều tra về tệp dữ liệu, EDA giúp tổng hợp thông tin từ các đặc điểm của dữ liệu, được trình bày bằng cách kết hợp các bảng thông tin và trực quan hóa dữ liệu.
EDA giúp ta có cái nhìn về dữ liệu trước khi đưa ra bất kỳ giả thuyết nào, xác định các lỗi, các khuôn mẫu, nhận dạng các dữ liệu ngoại lai, các điểm dữ liệu bất thường và tìm ra mối quan hệ giữa cấc biến. EDA giúp ta xác định và định hình các kỹ thuật thống kê mà ta cần sử dụng để phân tích hay xây dựng mô hình phù hợp ở các bước chuyên sâu hơn.

2.1.2 Công cụ hỗ trợ thực hiện quá trình EDA

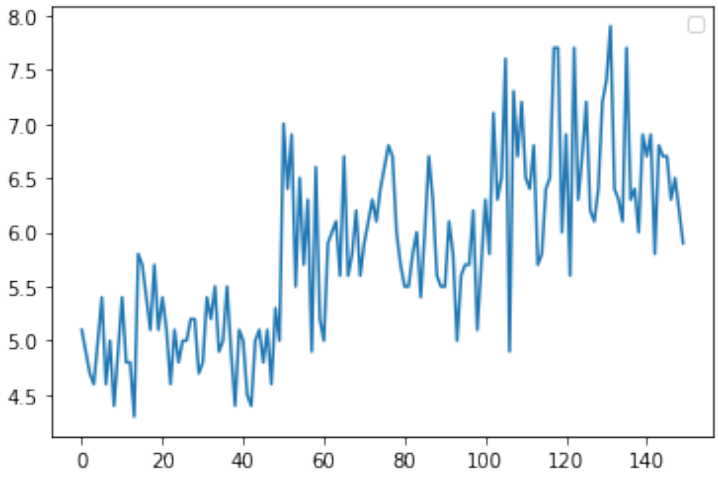
Python: là một ngôn ngữ lập trình được sử dụng rộng rãi trong khoa học dữ liệu và máy học (machine learning). Các nhà phát triển sử dụng Python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau. Phần mềm Python được tải xuống miễn phí, tích hợp tốt với nhiều loại hệ thống và có tốc độ phát triển nhanh.
R: là một phần phềm mã nguồn mở, miễn phí và được sử dụng rộng rãi cho các nhà thống kê, các nhà khoa học, y tế,... Ngôn ngữ R hiện nay được sử dụng rộng rãi trong cộng đồng khoa học học dữ liệu, thống kê.
Các ngôn ngữ và phần mềm khác: ngoài Python, R còn có nhiều công cụ và phần mềm khác như: Julia, JavaScript, C, Java... Các phần mềm thống kê như STATA, SPSS,... Phụ thuộc vào hiểu biết và văn hóa làm việc mà người dùng có thể tự do chọn các công cụ hỗ trợ cho việc phân tích này.

2.2 Các biểu đồ cơ bản sử dụng trong quá trình EDA

Trong phần này, học viên sử dụng tệp dữ liệu (dataset) hoa Iris để mô tả các biểu đồ.

2.2.1 Biểu đồ đường - Line Chart

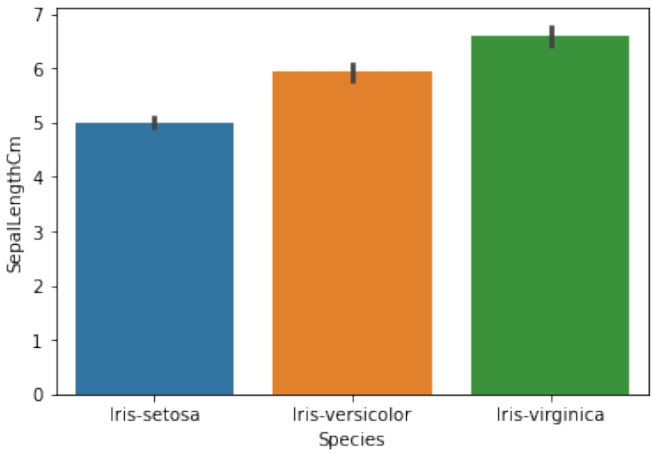
Biểu đồ đường là dạng biểu đồ thường xuyên được sử dụng nhất đối với dạng dữ liệu liên tục. Chúng kết nối các điểm dữ liệu (data point), tạo một chuỗi liên tục cho một hoặc nhiều chuỗi dữ liệu. Biểu đồ đường còn có thể được sử dụng như một đường xu hướng tuyến tính (linear) hoặc đa thức (polynomial trendlines) để trực quan hóa các mẫu hoặc dự báo các giai đoạn trong tương lai. Trong quá trình phân tích thống kê mô tả, biểu đồ đường cũng thường được vẽ cho đường trung bình nếu dữ liệu có biên động dao động cao.



Hình 2.1: Chiều dài đài hoa trong tệp dữ liệu hoa Iris

2.2.2 Biểu đồ cột - Bar Chart

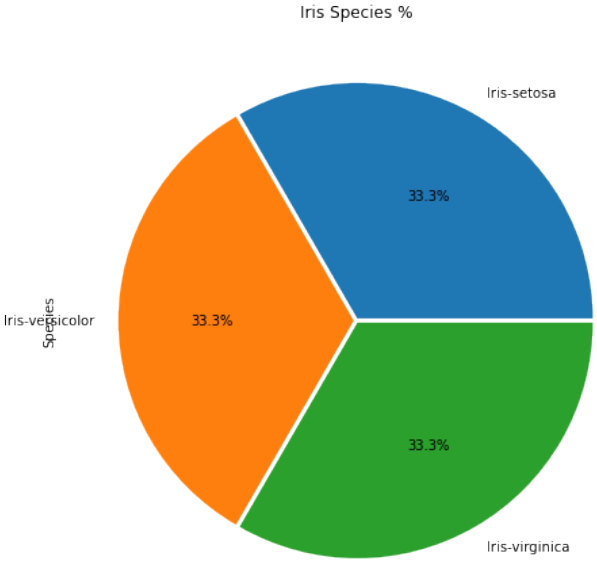
Biểu đồ cột là một trong những hình thức biểu đồ thông dụng nhất trong quá trình phân tích dữ liệu, chúng có rất nhiều biến thể và được sử dụng để giúp người xem nhanh chóng so sánh dữ liệu giữa các biến phân loại. Với biểu đồ cột, chúng ta hoàn toàn có thể tùy chỉnh các thanh/cột xếp chồng lên nhau hoặc theo cụm để nhóm theo danh mục con hoặc so sánh nhiều chỉ số.



Hình 2.2: Chiều dài đài hoa khi phân loại theo loài

2.2.3 Biểu đồ tròn - Pie Chart

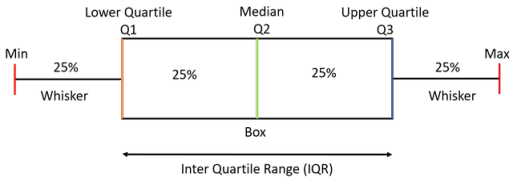
Biểu đồ tròn là một hình tròn được chia thành nhiều khu vực, trong đó mỗi khu vực đại diện cho tỷ lệ của tổng thể và được so sánh tỷ lệ theo phần trăm (tổng thể là 100%). Khi sử dụng biểu đồ tròn, lưu ý nên giữ số lượng các mục < 6 để tối đa hóa việc đọc và phân tích dữ liệu. Trong trường hợp nhãn dữ liệu có nhiều hơn thế, ta có thể xem xét sử dụng biểu đồ cột ngang; một lưu ý khác là không nên sử dụng định dạng 3D vì chúng dễ gây đánh lừa thị giác.



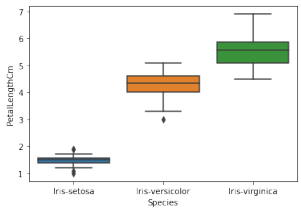
Hình 2.3: Tỉ lệ phân bổ các loài hoa trong tệp dữ liệu Iris

2.2.4 Biểu đồ hộp - Box Plot

Biểu đồ hộp là một dạng biểu đồ giúp người xem nhanh chóng biết được độ phân tán của dữ liệu, các khoảng phân vị, giá trị lớn nhất, nhỏ nhất hoặc các giá trị ngoại lai (outlier), hình 2.4 thể hiện rõ các giá trị của một biểu đồ hộp thể hiện. Biểu đồ hộp cũng có thể được dùng so sánh các giá trị trên với các biến phân loại khác, ví dụ như hình 2.5.



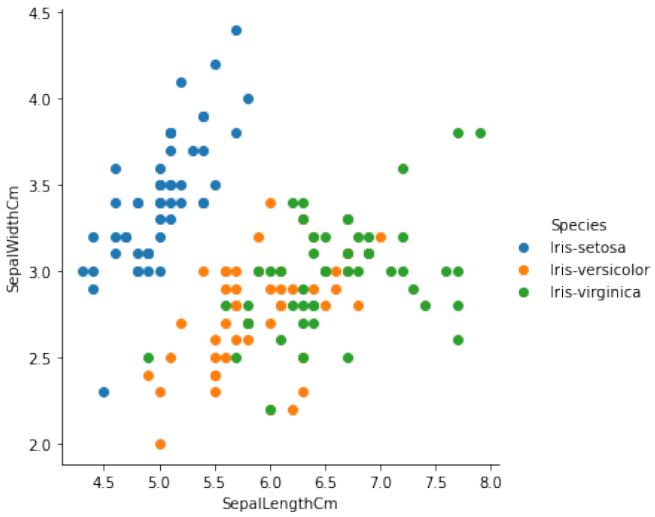
Hình 2.4: Các giá trị được biểu diễn trong biểu đồ hộp



Hình 2.5: Biểu đồ hộp về chiều dài cánh hoa giữa các loài

2.2.5 Biểu đồ phân tán - Scatter Plot

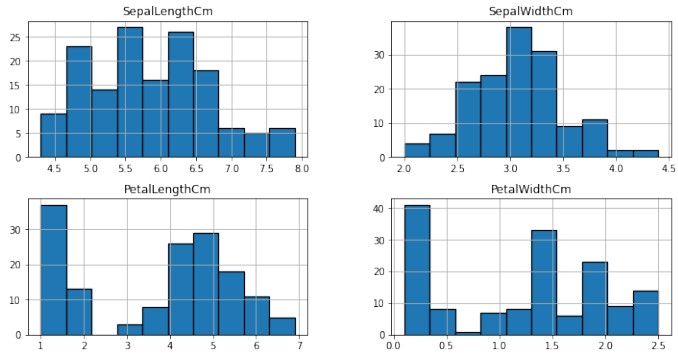
Scatter plot là biểu đồ thể hiện mối quan hệ giữa 2 biến số (numerical variables). Mỗi điểm dữ liệu được biểu diễn là một chấm tròn. Một ưu điểm là khi sử dụng biểu đồ phân tán là cúng ta phối hợp thêm một biểu đồ đường (đại diện cho một đường hồi quy tuyến tính) khi xây dựng mô hình cho dữ liệu.



Hình 2.6: Biểu đồ phân tán giữa chiều dài đài hoa và chiều rộng đài hoa giữa các loài

2.2.6 Biểu đồ Histogram

Histrogram là một dạng đồ thị thể hiện sự phân phối của số liệu theo một hoặc nhiều nhóm dữ liệu. Giá trị được chia thành nhiều đơn vị bins, ta có thể để tùy chỉnh kích thước của từng bin bằng cách nhóm các giá trị được phân nhóm (binning). Khi thể hiện trên biểu đồ, mỗi bin được biểu hiện dưới dạng một thanh trong biểu đồ cột, mỗi bin có giá trị bằng nhau.

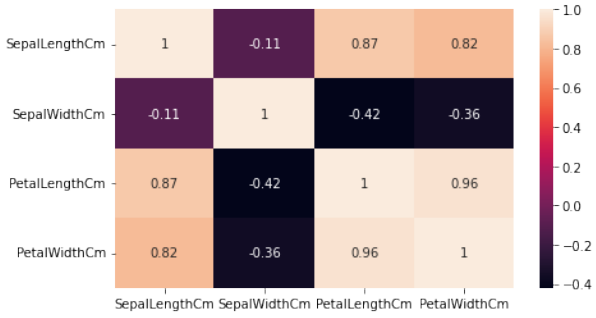


Hình 2.7: Histogram theo từng biến độc lập: chiều dài/chiều rộng đài hoa/cánh hoa

2.2.7 Biểu đồ nhiệt - Heatmap

Biểu đồ nhiệt là một hình thức nhằm trực quan hóa dữ liệu dưới hình thức là một ma trận bảng dữ liệu, chúng trực quan đến người xem bằng cách sử dụng thang màu. Ngoài việc cung cấp dữ liệu, biểu đồ sẽ nâng dần cường độ màu sắc khi giá trị dữ liệu càng cao, thúc đẩy việc kích thích thị giác chú ý ngay đến con số đó.

Biểu đồ nhiệt được sử dụng nhiều nhất khi cần tính chỉ số tương quan (correlation) hoặc confusion matrix. Tuy nhiên, một lưu ý rằng chỉ nên sử dụng heatmap khi có khoảng 6-8 giá trị cần quan sát. Nếu nhiều hơn, người đọc sẽ rất dễ bị rối loạn thông tin.



Hình 2.8: Biểu đồ nhiệt hệ số tương quan trong tập Iris

Chương 3

Thực nghiệm EDA trên tệp dữ liệu

3.1 Giới thiệu về tệp dữ liệu

Tệp dữ liệu mà đồ án thực hiện là về sự việc đắm tàu Titanic, sự kiện Titanic đã gây ra cái chết cho 1502/2224 hành khách. Hằng năm, **Kaggle** tổ chức cuộc thi "Titanic - Machine Learning from Disaster" nhằm áp dụng các kỹ thuật trí tuệ nhân tạo, máy học để dự đoán việc ai sẽ có nhiều khả năng sống sót hơn trong sự kiện trên. Trong giới hạn của đồ án, tệp dữ liệu được sử dụng là tập huấn luyện (train.csv). Phần thực hành code được lưu tại **link thư mục**.
Trong trường hợp các link trên không sử dụng được, học viên xin đính kèm link truy cập ở mã QR sau.
Một số công trình tiêu biểu có sử dụng tệp dữ liệu Titanic gồm [1,2,3] và nhiều

Hình 3.1: Link truy cập bằng mã QR

công trình khác. Các biến dữ liệu trong tập dữ liệu huấn luyện gồm:

Biến (Variable)	Định nghĩa	Giải thích
PassengerId	Số thứ tự hành khách	
Survived	Sống sót	0 = Không; 1 = Có
Pclass	Hạng vé	1 = Hạng 1; 2 = Hạng 2; 3 = Hạng 3
Name	Tên hành khách	
Sex	Giới tính	
Age	Tuổi	
Sibsp	Số lượng anh/chị/em trên tàu	
Parch	Số lượng cha mẹ/con cái trên tàu	
Ticket	Mã số vé tàu	
Fare	Giá vé	
Cabin	Số cabin ở	
Embarked	Cảng xuất hành	C: Cherbourg; Q: Queenstown; S: Southampton

3.2 EDA với Python

3.2.1 Chuẩn bị

Trước khi bắt đầu thực hiện EDA, ta gọi các thư viện cần thiết và load dữ liệu, gọi dữ liệu ta load vào với tên biến là data. Một số các thư viện được sử dụng tại bộ dữ liệu này là:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import missingno as msno
import warnings
from collections import Counter
from sklearn.preprocessing import LabelEncoder
```

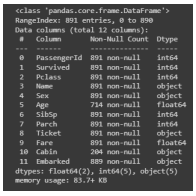
3.2.2 Bước 1: Hiểu về dữ liệu - Data Understanding

Các việc ta cần lưu ý thực hiện ở bước này bao gồm:

- Tìm hiểu về Dataframe shape;
- Sử dụng lệnh head/tail để có cái nhìn lướt qua về dữ liệu;
- Kiểm tra định dạng dữ liệu ở các cột với dtypes;
- Kiểm tra nhanh các đại lượng thống kê bằng lệnh describe.

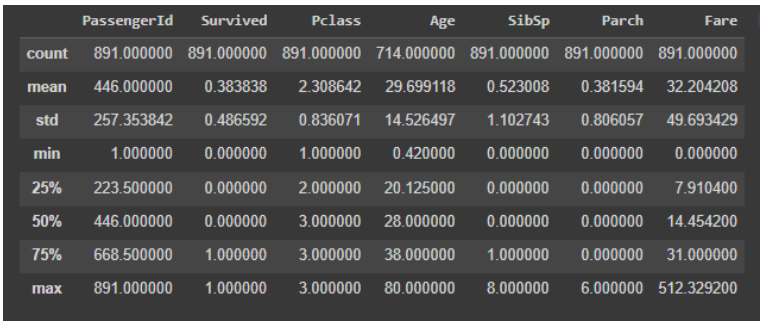
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Hình 3.2: lệnh data.head()



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    int64
 3   Name         891 non-null    object
 4   Sex          891 non-null    object
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64
 7   Parch        891 non-null    int64
 8   Ticket       891 non-null    object
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

Hình 3.3: lệnh data.info()



	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

Hình 3.4: data.describe()

Một số hình ảnh kết quả:
Bộ data gốc có 891 dòng quan sát với 12 biến (11 biến độc lập và 1 biến phụ thuộc Survived). Nhìn qua lệnh *data.info* 3.3, ngoài nhận thấy các định dạng dữ liệu, ta cần lưu ý có dữ liệu NA. Và dựa vào lệnh *data.describe* 3.4 ta thấy được cần lưu ý tới các biến Sex (giới tính), Embarked (Cảng xuất hành) cần được xử lý.

3.2.3 Bước 2: Chuẩn bị dữ liệu - Data Preparation

Các việc ta cần lưu ý thực hiện ở bước này bao gồm:

- Loại bỏ các cột không liên quan/không ảnh hưởng đến kết quả cần phân tíhc. Ở đây học viên quyết định loại bỏ các biến: PassengerId, Name, Cabin, Ticket;

- Mã hóa các biến Sex, Embarked thành thang đo định danh;
- Lấp đầy/xóa dữ liệu khuyết/NA;
- Kiểm tra nhanh việc trùng lắp thông tin (duplicate);
- Đổi tên cột (nếu cần thiết)

```
data.isna().sum()

Survived      0
Pclass        0
Sex           0
Age         177
SibSp         0
Parch         0
Fare          0
Embarked      2
dtype: int64
```

```
[15] data.loc[data.duplicated()]
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
47	1	3	female	NaN	0	0	7.7500	Q
76	0	3	male	NaN	0	0	7.8958	S
77	0	3	male	NaN	0	0	8.0500	S
87	0	3	male	NaN	0	0	8.0500	S
95	0	3	male	NaN	0	0	8.0500	S
...	...	...	...	...	...	...	...	...
870	0	3	male	26.0	0	0	7.8958	S
877	0	3	male	19.0	0	0	7.8958	S
878	0	3	male	NaN	0	0	7.8958	S
884	0	3	male	25.0	0	0	7.0500	S
886	0	2	male	27.0	0	0	13.0000	S

111 rows × 8 columns

Hình 3.5: Kiểm tra dữ liệu khuyết và trùng lặp

```
[17] data.loc[detect_outliers(data,["Age","SibSp","Fare", "Parch"])]
```

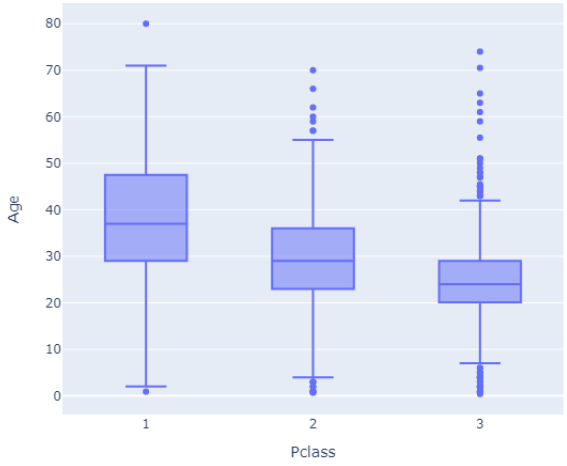
	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
27	0	1	male	19.0	3	2	263.00	S
88	1	1	female	23.0	3	2	263.00	S
159	0	3	male	NaN	8	2	69.55	S
180	0	3	female	NaN	8	2	69.55	S
201	0	3	male	NaN	8	2	69.55	S
324	0	3	male	NaN	8	2	69.55	S
341	1	1	female	24.0	3	2	263.00	S
792	0	3	female	NaN	8	2	69.55	S
846	0	3	male	NaN	8	2	69.55	S
863	0	3	female	NaN	8	2	69.55	S

Hình 3.6: Kiểm tra dữ liệu ngoại lai

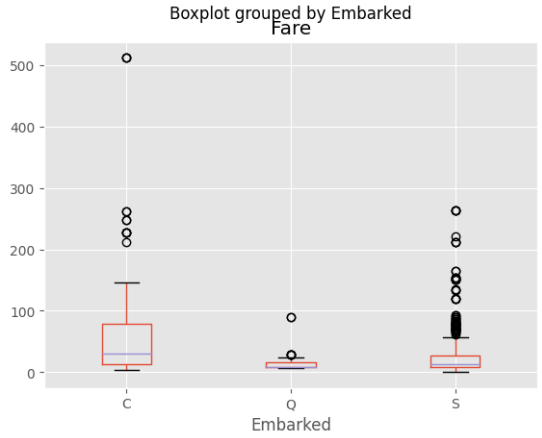
Sau khi đã loại bỏ các biến không cần thiết. Học viên kiểm tra dữ liệu bị khuyết, trong đó có 177 dòng khuyết Tuổi và 2 dòng khuyết Cảng xuất hành. Từ hình 3.5 cho thấy, trên quan điểm chủ quan, dữ liệu trùng lặp không gây ảnh hưởng đến bộ dataset. Tương tự với hình 3.6 về dữ liệu ngoại lai. Vì vậy học viên không loại bỏ các dòng quan sát này.
Ở hình 3.7, sử dụng biểu đồ hộp từ gói thư viện plotly, nhờ vào việc trực quan, học viên thấy được rằng giá trị trung vị về tuổi cho các hạng vé 1, 2, 3 lần lượt là: 37, 29, 24 tuổi. Thay các giá trị tương ứng trên vào các dữ liệu tuổi bị khuyết. Tương tự với hình 3.8, một cách chủ quan có thể quan sát thấy rằng độ trải giữa (IQR) của Cảng xuất hành tại C - Cherbourg có khả năng cao phù hợp với giá vé của 2 quan sát bị khuyết (giá vé mua là $80 đô la).

3.2.4 Bước 3: Quan sát biến và các mối quan hệ của chúng

Các việc ta cần lưu ý thực hiện ở bước này bao gồm:

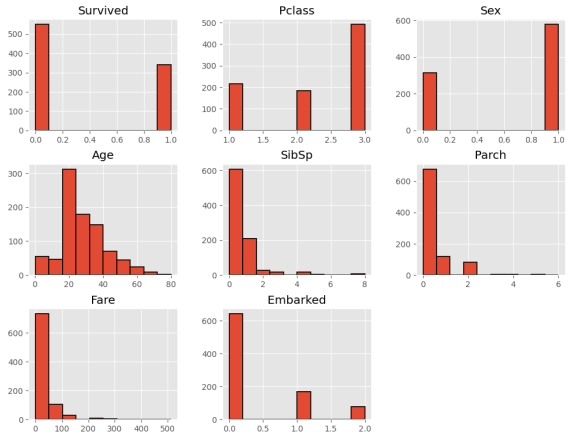


Hình 3.7: Biểu đồ hộp của các hạng vé theo tuổi

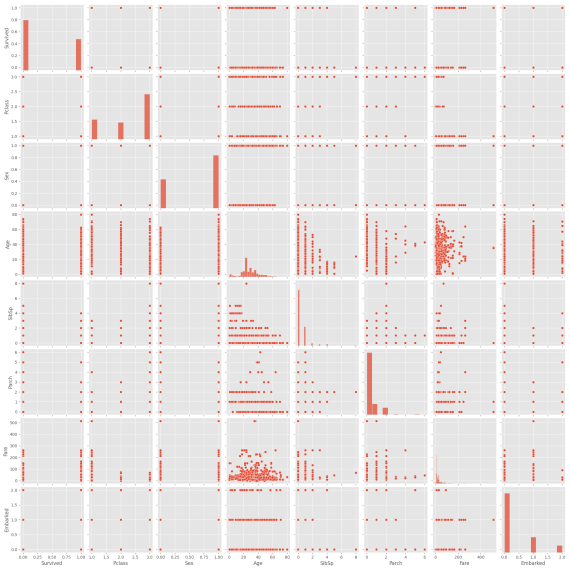


Hình 3.8: Biểu đồ hộp của cảng xuất hành theo giá vé

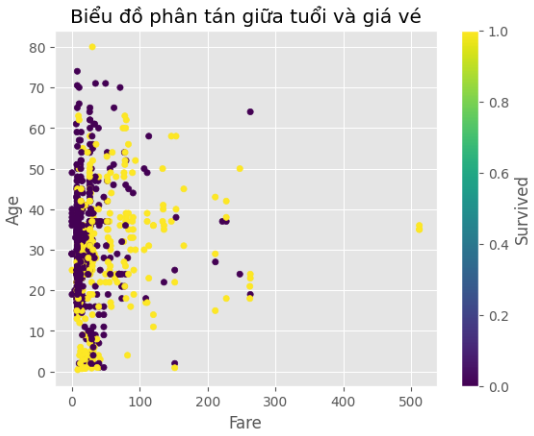
- Quan sát histogram ở các biến;
- Quan sát tương quan ở từng cặp biến và kiểm tra hệ số tương quan;
- So sánh các biến độc lập với biến phụ thuộc và tạm thời đưa ra các giả định.



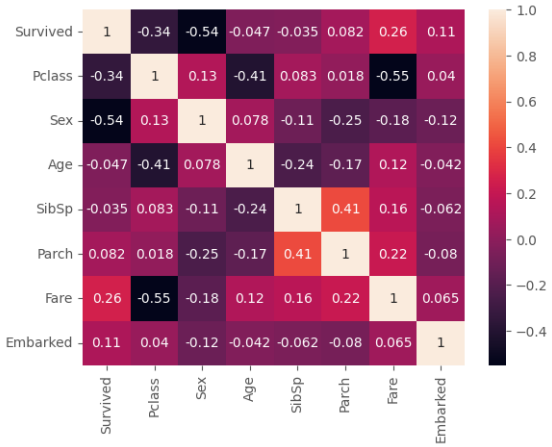
Hình 3.9: Histogram của từng biến



Hình 3.10: Biểu đồ phân tán của từng cặp biến



Hình 3.11: Biểu đồ phân tán cụ thể tại cặp biến tuổi và giá vé



Hình 3.12: Biểu đồ nhiệt về hệ số tương quan

Dựa vào histogram 3.9 có thể thấy được một số kết luận trực quan như: Tỉ lệ sống sót thấp, đa số là khách đi tàu là mua vé hạng 3, về giới tính thì nam nhiều hơn nữ, các hành khách đa số là người còn trẻ tuổi,...
Hình 3.10 là biểu đồ phân tán khi vẽ cho tất cả từng cặp biến, ta quan sát được có vẻ chúng không tạo ra mối quan hệ tương quan nào mạnh cả. Hình 3.11 giúp ta cụ thể hơn biểu đồ phân tán cho biến Tuổi và Giá vé với màu sát được phân loại theo biến Sống sót.
Kiểm tra kỹ tại biểu đồ nhiệt tại hình 3.12, ngoài cặp tương quan Giới tính và Sống sót hay giữa Giá vé và Hạng vé có thể tạm thời được xem là tương quan mạnh thì các biến còn lại có mối quan hệ tương quan trung bình và yếu. Ngoài ra còn có một số kết luận tạm thời có thể được đưa ra gồm:

- Nếu hành khách là nữ giới thì cơ may sống sót cao hơn nam giới;
- Nếu có càng ít anh chị em đi cùng (từ 2 người trở xuống) thì hành khách càng có nhiều cơ may sống sót;
- Nếu số lượng người phụ thuộc (cha mẹ, con cái) không quá nhiều (từ 3 người trở xuống) thì có cơ hội sống sót;
- Nếu khởi hành từ cảng Cherbourg, thì hành khách có nhiều cơ may sống sót.

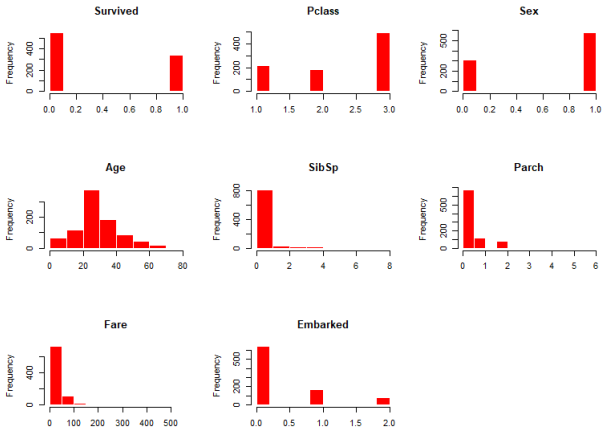
Tuy các giả thuyết trên vẫn cần được kiểm chứng. Nhưng dựa trên các kết luận tạm thời này, người thực hiện phân tích và xây dựng mô hình có thể xác định trước một số mục tiêu để có thể dự đoán kết quả sống sót tốt hơn.

3.3 EDA với R

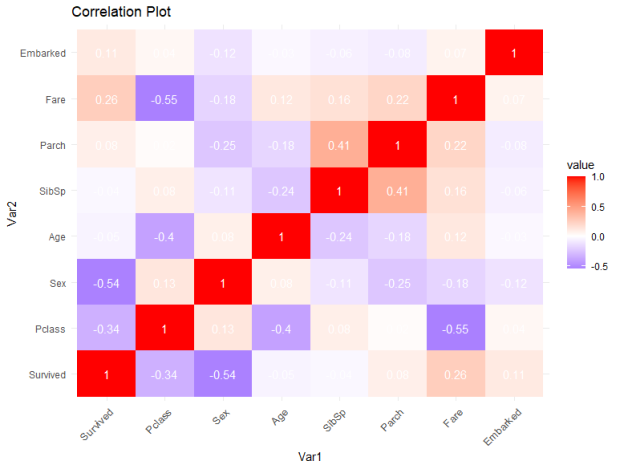
Là một ngôn ngữ chuyên phục cho hoạt động nghiên cứu, thống kê, xây dựng các mô hình máy học mạnh mẽ và không kém phần phổ biến so với Python. Trong mục này, học viên thực hiện lại các bước đã thực hiện tại phần 3.2. Chi tiết phần code có được lưu tại thư mục với tên file **Titanic_EDA_withR.R**. Một số hình ảnh trong quá trình thực hiện EDA với ngôn ngữ R:

```
> head(data) #data.head()
  PassengerId Survived Pclass                                                Name    Sex Age SibSp Parch           Ticket    Fare Cabin Embarked
1           1        0      3                             Braund, Mr. Owen Harris   male  22     1     0        A/5 21171  7.2500              S
2           2        1      1 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0         PC 17599 71.2833   C85        C
3           3        1      3                              Heikkinen, Miss. Laina female  26     0     0 STON/O2. 3101282  7.9250              S
4           4        1      1        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0           113803 53.1000  C123        S
5           5        0      3                            Allen, Mr. William Henry   male  35     0     0           373450  8.0500              S
6           6        0      3                                    Moran, Mr. James   male  NA     0     0           330877  8.4583              Q
> |
```

Hình 3.13: 6 dòng quan sát đầu tiên về tập dữ liệu



Hình 3.14: Histogram với từng biến trong tập dữ liệu



Hình 3.15: Biểu đồ nhiệt cùng hệ số tương quan của từng cặp biến

Kết luận

Quá trình phân tích khám phá dữ liệu (EDA) là quá trình phân tích và khám phá dữ liệu để hiểu rõ hơn về đặc điểm, mối quan hệ và xu hướng của dữ liệu. Từ đó, ta có thể tìm ra các insights và giải thích được những hiện tượng xảy ra trong dữ liệu. Quá trình này bao gồm các bước tiền xử lý dữ liệu, mô tả dữ liệu, phân tích tương quan và trực quan hóa dữ liệu. EDA là một phương pháp quan trọng trong khoa học dữ liệu và đóng vai trò quan trọng trong việc giải quyết các vấn đề thực tế. Tùy vào từng đặc điểm riêng của tệp dữ liệu, của vấn đề cần xử lý mà ta cần có hướng xử lý phù hợp khi thực hiện EDA, không thực hiện một cách máy móc.

Tài liệu tham khảo

[1] Kaggle, *Titanic - Machine Learning from Disaster*, 2023.

[2] Ekinci, Ekin and Omurca, S Ilhan and Acun, Neytullah, *A comparative study on machine learning techniques using Titanic dataset*, 2018.

[3] Barhoom, Alaa M., Ahmed J. Khalil, Bassem S. Abu-Nasser, Musleh M. Musleh, and Samy S. Abu Naser, *Predicting Titanic Survivors using Artificial Neural Network*, 2019.

[4] Singh, Aakriti, Shipra Saraswat, and Neetu Faujdar, *Analyzing Titanic disaster using machine learning algorithms*, In 2017 International Conference on Computing, Communication and Automation (ICCCA), pp. 406-411. IEEE, 2017.

[5] Bruce, Peter, Andrew Bruce, and Peter Gedeck, *Practical statistics for data scientists: 50+ essential concepts using R and Python. O'Reilly Media*, 2020.

[6] Matthijs Rooduijn, University of Amsterdam, Coursera, *Basic Statistics*, <https://www.coursera.org/learn/basic-statistics>.

Phụ lục A

Code Python

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import missingno as msno
import warnings
from collections import Counter
from sklearn.preprocessing import LabelEncoder
warnings.filterwarnings("ignore")
plt.style.use('ggplot')

##STEP1
data.shape
data.head()
data.dtypes
data.info()
data.describe()

##STEP2
data.columns
data = data[['Survived', 'Pclass', 'Sex', 'Age',
             'SibSp', 'Parch', 'Fare', 'Embarked']].copy()
data.isna().sum()
data.loc[data.duplicated()]
def detect_outliers(df,features):
    outlier_indices = []

    for c in features:
        # 1st Quartile
        Q1 = np.percentile(df[c],25)
        # 3rd Quartile
        Q3 = np.percentile(df[c],75)
        # IQR
        IQR = Q3 - Q1
        # Outlier Step
        outlier_step = IQR * 1.5
        # Detect Outlier and Their Indicies
        outlier_list_col = df[(df[c]< Q1 - outlier_step)
                        | (df[c] > Q3 + outlier_step)].index
```

```
        # Store Indicies
        outlier_indices.extend(outlier_list_col)
    outlier_indices = Counter(outlier_indices)
    multiple_outliers = list(i for i, v
                        in outlier_indices.items() if v>2)

    return multiple_outliers

data.loc[detect_outliers(data,["Age","SibSp","Fare", "Parch"])]
px.box(data, x="Pclass", y="Age")
def fill_age(cols):
    Age = cols[0]
    Pclass = cols[1]
    if pd.isnull(Age):
        if Pclass == 1:
            return 37

        if Pclass == 2:
            return 29

        if Pclass == 3:
            return 24

    else:

        return Age
data["Age"]= data[["Age","Pclass"]].apply(fill_age,axis =1)
data.loc[data['Embarked'].isna()]
data.boxplot(column="Fare",by = "Embarked")
data['Embarked'] = data['Embarked'].fillna('C')
data.isna().sum()
data['Embarked'] = data['Embarked'].map({'S':0, 'C':1,'Q':2})
data["Sex"]=pd.get_dummies(data["Sex"],drop_first=True)

#STEP3
data.hist(edgecolor='black', linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,9)
plt.show()

sns.pairplot(data,
             vars= ['Survived', 'Pclass', 'Sex', 'Age',
             'SibSp', 'Parch', 'Fare', 'Embarked'])

data.plot(kind = 'scatter',
          x= 'Fare',
          y= 'Age',
          c = 'Survived',
          colormap='viridis',
          title = 'Bieu do phan tan giua tuoi va gia ve')

df_corr = data.corr()
sns.heatmap(df_corr, annot= True)
```

Phụ lục B

Code R

```r
library("reshape2")
library("ggplot2")
library("dplyr")
library("caret")
library("readr")

###STEP1
data <- read.csv(file.choose(), header=T)
dim(data) #data.shape
head(data) #data.head()
str(data) #equivalent to data.dtypes in Python
summary(data) #data.describe()

###STEP2
sapply(data, class)
data <- subset(data, select = c('Survived', 'Pclass', 'Sex',
                                'Age', 'SibSp', 'Parch',
                                'Fare', 'Embarked'))
head(data)
#Check NA values, equivalent data.isna().sum()
colSums(is.na(data))
#return average age.
aggregate(Age ~ Pclass, data = data, FUN = mean)

#
fill_age <- function(Age, Pclass) {
  if (is.na(Age)) {
    if (Pclass == 1) {
      return(38)
    } else if (Pclass == 2) {
      return(30)
    } else {
      return(25)
    }
  } else {
    return(Age)
  }
}
data$Age <- mapply(fill_age, data$Age, data$Pclass)
```

```r
##
data <- data %>%
  mutate(Embarked = case_when(
    Embarked == 'S' ~ 0,
    Embarked == 'C' ~ 1,
    Embarked == 'Q' ~ 2,
    TRUE ~ NA_real_
  ))
colSums(is.na(data))
data[is.na(data$Embarked), ]
#boxplot(Fare ~ Embarked, data = data)
data$Embarked <- ifelse(is.na(data$Embarked), 1, data$Embarked)
colSums(is.na(data))

data <- data %>%
  mutate(Sex = ifelse(Sex == "male", 1, 0))
colSums(is.na(data))

###STEP 3
summary(data)
# Plot histograms for all columns
par(mfrow=c(3,3))
for(i in 1:ncol(data)){
  hist(data[,i], main=names(data)[i], col="red",
       border="white", xlab="")
}

# Calculate the correlation matrix
df_corr <- cor(data)
melted_cor <- melt(df_corr)

ggplot(data = melted_cor, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() +
  geom_text(aes(label = round(value,2)), color = "white") +
  scale_fill_gradient2(low="blue", mid="white",
                       high="red", midpoint=0, space="Lab") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 10, hjust = 1)) +
  labs(title="Correlation Plot")
```