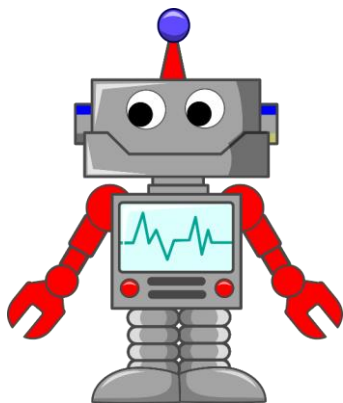




ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

CS116 – LẬP TRÌNH PYTHON CHO MÁY HỌC

Introduction Feature Engineering & Selection



TS. Nguyễn Vinh Tiệp

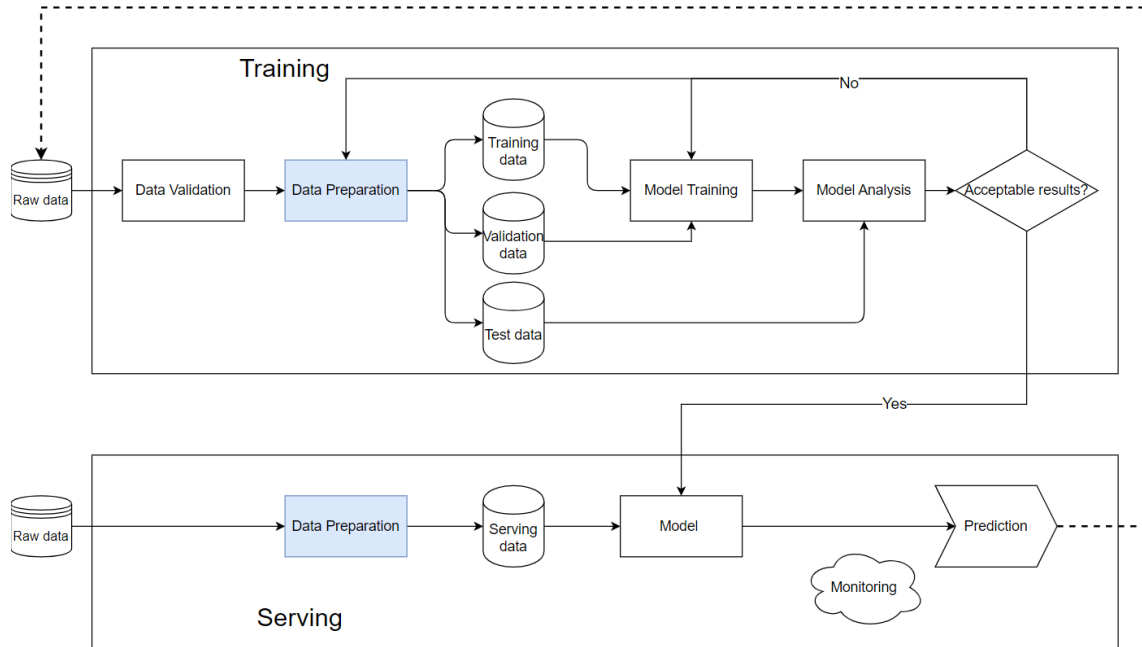


NỘI DUNG

- EDA
- Feature Engineering & Data Transformation
- Preprocessing Data



ML Pipeline



https://machinelearningcoban.com/tabml_book/ch_intro/pipeline.html



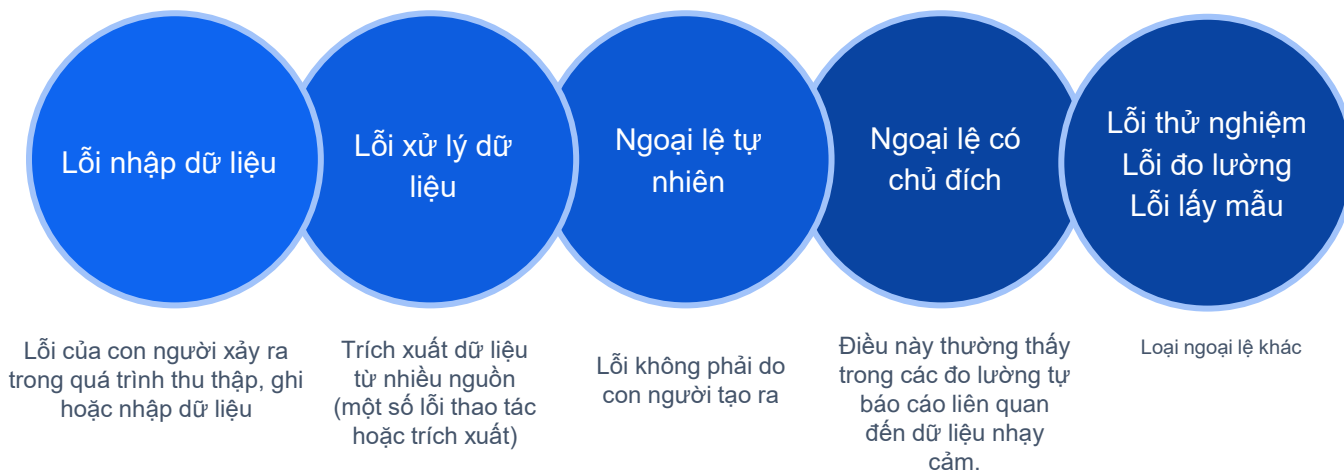
Data Validation

1. Phát hiện các ngoại lệ
2. Xử lý các ngoại lệ (ngoại bỏ, thay thế giá trị)





Ngoại lệ: Các kiểu ngoại lệ khác nhau





Làm sạch dữ liệu - Ngoại lệ

- ❑ Có hai loại ngoại lệ:
 - ❑ Các ngoại lệ đơn biến: các điểm dữ liệu **có giá trị nằm ngoài phạm vi giá trị dự kiến**
 - ❑ Các ngoại lệ đa biến: có các ngoại lệ **phụ thuộc** vào **mối tương quan giữa hai biến**



Làm sạch dữ liệu: Hướng tiếp cận đơn giản

- ❑ Một số bước có thể được thực hiện để làm sạch dữ liệu:
 - ❑ Xóa các giá trị bị thiếu, trùng lặp, ngoại lệ và các hàng/cột không cần thiết
 - ❑ Lập lại chỉ mục và định dạng lại dữ liệu

```
# Drop rows with missing value
data.dropna(inplace = True)

# Remove duplicates
data.drop_duplicates()

# Drop unnecessary columns
data.drop(columns = [list cols], axis = 1)

# Drop/Filter unnecessary rows
data.drop([0, 1], inplace = True)
data[data['column_filter'] == 'abc']
```

```
# Re indexing
data.set_index('column', inplace = True)
data.reset_index(drop = True)

# Re-formatting
data['column'] = data['column'].astype(int)

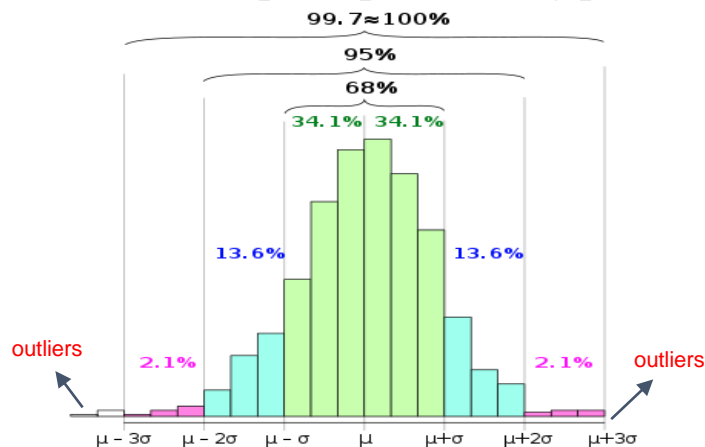
# Correcting inconsistent data
data['column'].replace(old_value, new_value, inplace = True)
```



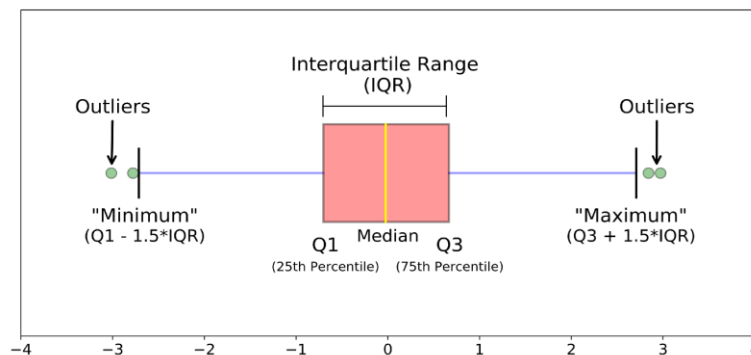
Dự đoán ngoại lệ như thế nào

❑ Phương pháp thống kê:

- ❑ Phương pháp tính độ lệch chuẩn và trung bình: giới hạn để xác định các giá trị ngoại lệ (Gaussian hoặc Gaussian-like)
- ❑ Phương pháp Interquartile Range (IQR): một thống kê tốt để tóm tắt mẫu dữ liệu phân phối không phải Gaussian



https://en.wikipedia.org/wiki/68%E2%80%9395%E2%80%9399.7_rule



<https://github.com/NaysanSaran/stats101/blob/master/>



Tự động phát hiện ngoại lệ

- ❑ Tự động phát hiện ngoại lệ:
 - ❑ Yếu tố ngoại lệ cục bộ ([Local Outlier Factor](#)): Xác định các ngoại lệ là xác định vị trí của các mẫu ở xa mẫu khác
 - ❑ Rừng cách ly ([Isolation Forest](#)) : Thuật toán phát hiện bất thường dựa trên cây
 - ❑ Xác định hiệp phương sai tối thiểu ([EllipticEnvelope](#)): Tập dữ liệu theo phân phối chuẩn.
 - ❑ [One-class SVM](#) : Phát hiện ngoại lệ không được giám sát



Yếu tố ngoại lệ cục bộ - Local Outlier Factor (LOF)

- Thuật toán yếu tố ngoại lệ cục bộ có thể được chia thành bốn phần



K-Distance and K-Neighbors

Nếu k nhỏ thì thuật toán trở nên nhạy cảm với nhiễu và nếu k lớn, nó có thể không nhận ra được các dị thường cục bộ.



Reachability Distance

Biểu thị khoảng cách lớn nhất của hai điểm và khoảng cách- k của điểm thứ hai



Local Reachability Density

Đề cập đến việc chúng ta cần đi bao xa từ điểm hiện tại để đến điểm hoặc tập hợp điểm tiếp theo

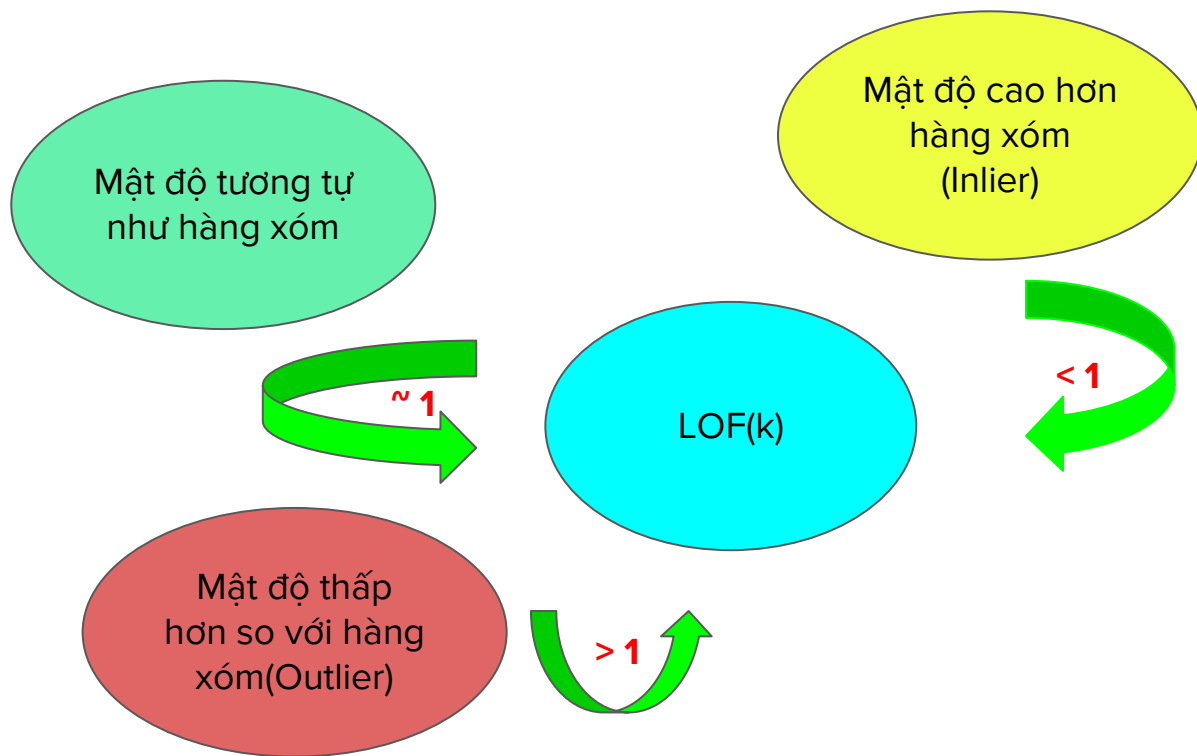


Local Outlier Factor Calculation

Mật độ khả năng tiếp cận cục bộ được tìm thấy được so sánh với mật độ khả năng tiếp cận cục bộ của k hàng xóm gần nhất



LOF: Làm thế nào để phát hiện các ngoại lệ?





Ví dụ

- ❑ Example
- ❑ LOF with sklearn





Cleanlab: Công cụ AI tập trung vào dữ liệu tiêu chuẩn

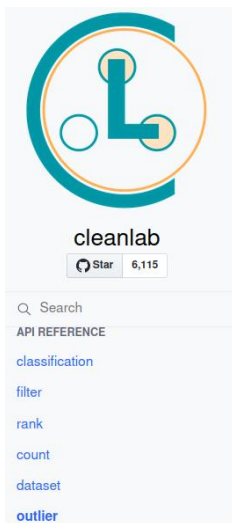
 [github](#)





Cleanlab: Tìm mẫu OOD (Out-Of-Distribution)

□ [Source code](#)



outlier

Methods for finding out-of-distribution examples in a dataset via scores that quantify how atypical each example is compared to the others.

The underlying algorithms are described in [this paper](#).

Classes:

`OutOfDistribution` ([params])

Provides scores to detect Out Of Distribution (OOD) examples that are outliers in a dataset.

class `cleanlab.outlier.OutOfDistribution`(params=None)

[\[source\]](#)

Bases: `object`

Provides scores to detect Out Of Distribution (OOD) examples that are outliers in a dataset.

Each example's OOD score lies in $[0,1]$ with smaller values indicating examples that are less typical under the data distribution. OOD scores may be estimated from either: numeric feature embeddings or predicted probabilities from a trained classifier.

To get indices of examples that are the most severe outliers, call `find_top_issues` function on the returned OOD scores.



Cleanlab: Tìm vấn đề về nhãn

- ❑ Phát hiện các vấn đề về dữ liệu: ngoại lệ, trùng lặp, lỗi nhãn,...

```
KNN = NearestNeighbors(metric='euclidean')
KNN.fit(X_processed.values)

knn_graph = KNN.kneighbors_graph(mode="distance")
```

```
data = {"X": X_processed.values, "y": labels}

lab = Datalab(data, label_name="y")
lab.find_issues(pred_probs=pred_probs, knn_graph=knn_graph)
```

```
Finding label issues ...
Finding outlier issues ...
Finding near_duplicate issues ...
Audit complete. 357 issues found in the dataset.
```

```
lab.report()
```

Here is a summary of the different kinds of issues found in the data:

issue_type	num_issues
label	294
outlier	46
near_duplicate	17

Dataset Information: num_examples: 941, num_classes: 5

----- label issues -----

About this issue:

Examples whose given label is estimated to be potentially incorrect (e.g. due to annotation error) are flagged as having label issues.



Data preparation

- Data fusion
- Data cleaning
- Data augmentation
- Data visualization
- Data splitting
- ...



Làm sạch dữ liệu – thiếu giá trị

- ❑ Bộ dữ liệu chứa các **giá trị bị thiếu**, thường được mã hóa dưới dạng trống, NaN,...
- ❑ Phương pháp đơn giản:
 - ❑ Bỏ các cột có tỷ lệ thiếu giá trị cao (chẳng hạn như 80%)
 - ❑ Tự động điền giá trị còn thiếu
- ❑ Phương pháp khác



EDA

- Phân tích các đặc trưng chính của dữ liệu
- Mô tả bằng số liệu, biểu đồ thống kê hoặc trực quan hóa dữ liệu





EDA: Ví dụ về dữ liệu

Dataset link

The screenshot shows the Kaggle competition page for 'Spaceship Titanic'. The header includes the title 'Spaceship Titanic' and the subtitle 'Predict which passengers are transported to an alternate dimension'. It also shows the Kaggle logo, the number of teams (2,333), and the status (Ongoing). The navigation bar includes links for Overview, Data, Code, Discussion, Leaderboard, Rules, Team, Submissions, and a Submit Predictions button. The main content area is titled 'Overview' and contains a 'Description' section. The description includes a 'Recommended Competition' section that suggests the 'Titanic - Machine Learning from Disaster' competition for beginners. The main text of the description welcomes participants to the year 2912 and describes the mission of the Spaceship Titanic, which was launched a month ago and is currently on its maiden voyage to three newly habitable exoplanets. It also mentions a collision with a spacetime anomaly near Alpha Centauri, which resulted in the transport of passengers to an alternate dimension.

Getting Started Prediction Competition

Spaceship Titanic

Predict which passengers are transported to an alternate dimension

Kaggle · 2,333 teams · Ongoing

Overview Data Code Discussion Leaderboard Rules Team Submissions [Submit Predictions](#) ...

Overview

Description

Evaluation

Frequently Asked Questions

Recommended Competition

We highly recommend [Titanic - Machine Learning from Disaster](#) to get familiar with the basics of machine learning and Kaggle competitions.

Welcome to the year 2912, where your data science skills are needed to solve a cosmic mystery. We've received a transmission from four lightyears away and things aren't looking good.

The *Spaceship Titanic* was an interstellar passenger liner launched a month ago. With almost 13,000 passengers on board, the vessel set out on its maiden voyage transporting emigrants from our solar system to three newly habitable exoplanets orbiting nearby stars.

While rounding Alpha Centauri en route to its first destination—the torrid 55 Cancri E—the unwary *Spaceship Titanic* collided with a spacetime anomaly hidden within a dust cloud. Sadly, it met a similar fate as its namesake from 1000 years before. Though the ship stayed intact, almost half of the passengers were transported to an alternate dimension!



Data loading

Tải dữ liệu và in ra màn hình các dữ liệu đầu tiên

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



Các loại dữ liệu

Có 2 loại dữ liệu chính:

- **Numerical** data: dữ liệu dạng số
- **Categorical** data: dữ liệu dạng phân loại

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



Kiểm tra các đặc trưng thống kê

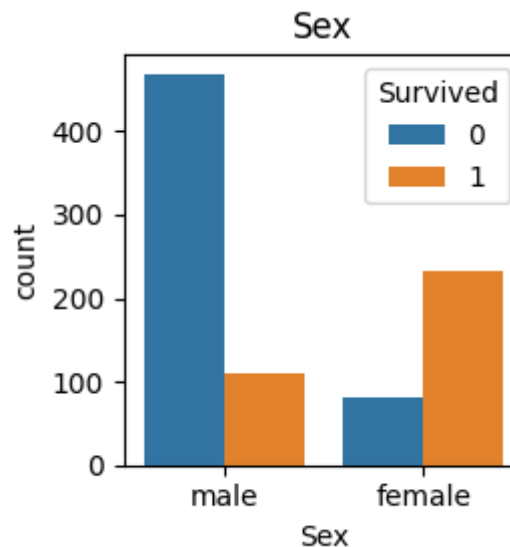
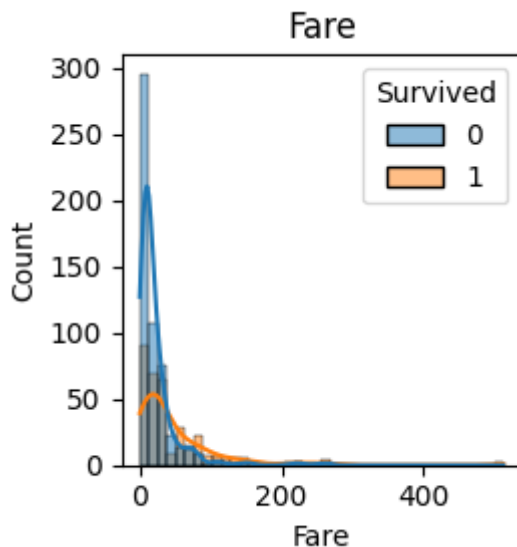
```
1 df_train.describe()
```

	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



Đồ thị thống kê

Sử dụng thư viện **matplotlib** hoặc **seaborn** để vẽ đồ thị





Các kiểu phân tích dữ liệu bằng cách thăm dò

- ❑ Có ba loại EDA chính
 - ❑ Phân tích đơn biến
 - ❑ Phân tích hai biến
 - ❑ Phân tích đa biến



Phân tích đơn biến

- ❑ Phân tích dữ liệu của chỉ một biến (đặc trưng/cột)
- ❑ Phân tích đơn biến không dùng biểu đồ
 - ❑ Trung tâm dữ liệu (Central Tendency): đề cập đến giá trị nằm ở vị trí trung tâm hoặc khu vực giữa của dữ liệu (các tham số ước lượng: trung bình, trung vị, mode).
 - ❑ Phạm vi: Sự **khác biệt** giữa giá trị **tối đa** và **tối thiểu** trong dữ liệu
 - ❑ Phương sai và độ lệch chuẩn



Phân tích đơn biến không dùng biểu đồ

- Tính trung bình, Độ lệch chuẩn, giá trị nhỏ nhất, lớn nhất và phân vị thứ nhất (first quartile), phân vị thứ ba (third quartile), trung vị của các cột số trong tập dữ liệu.

```
1 df_train[['Survived', 'Pclass', 'Age', 'SibSp', 'Fare']].describe()
```

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

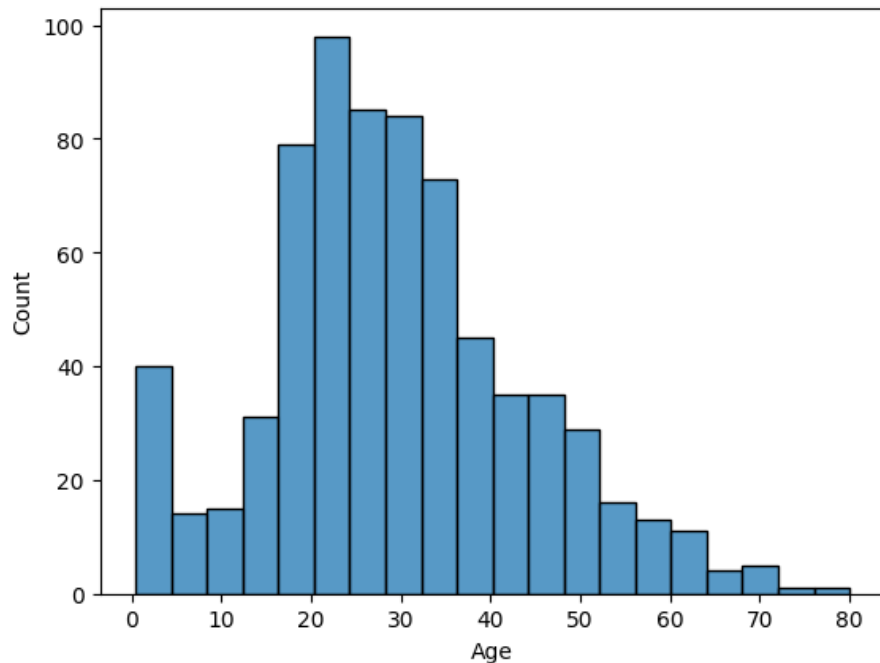


Phân tích đơn biến

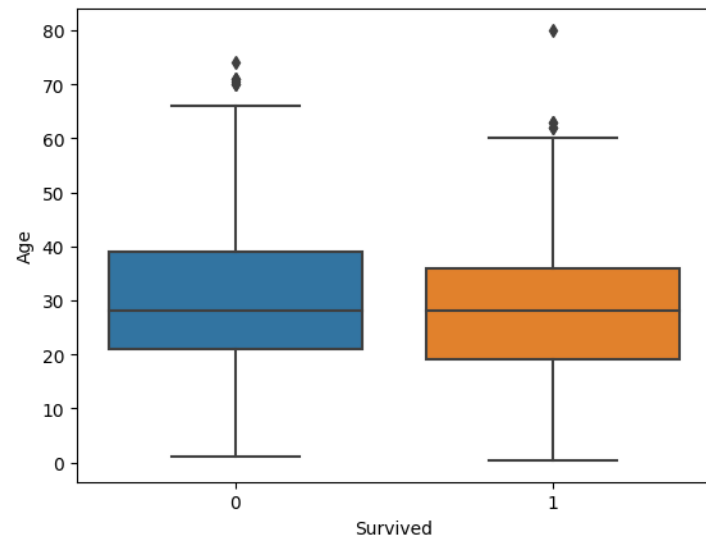
- ❑ Phân tích đơn biến sử dụng biểu đồ
 - ❑ Biểu đồ tần suất (Histogram): Biểu đồ dạng thanh trong đó **tần số của dữ liệu** được biểu thị bằng các thanh hình chữ nhật
 - ❑ Biểu đồ mật độ: giống như một phiên bản **mượt mà hơn** của **biểu đồ tần suất**
 - ❑ Box-plot: Ở đây thông tin được thể hiện dưới dạng các hộp (**giá trị nhỏ nhất, phân vị thứ nhất (first quartile), trung vị (median), phân vị thứ ba (third quartile), giá trị lớn nhất**)



Histogram & Box plot



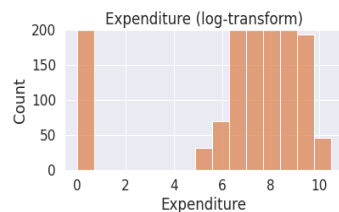
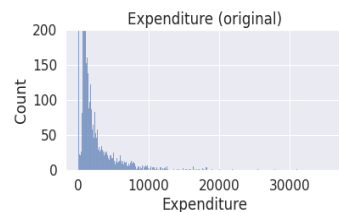
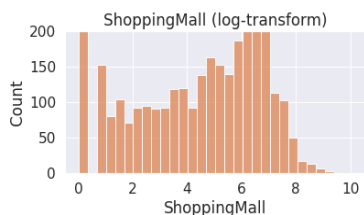
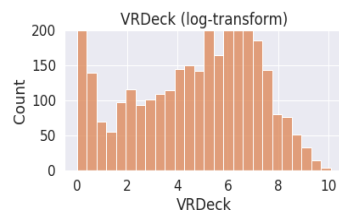
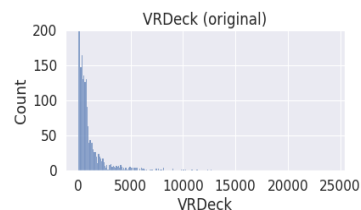
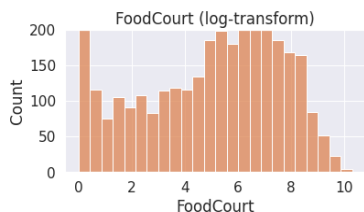
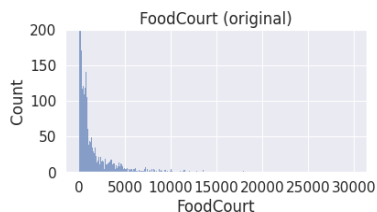
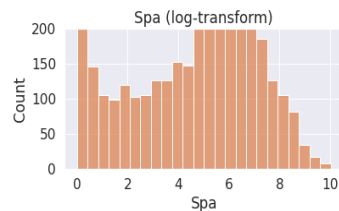
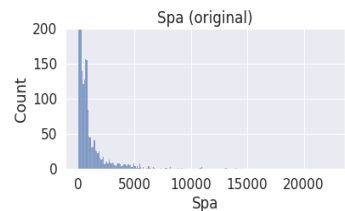
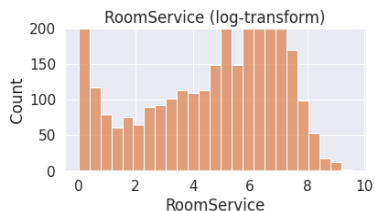
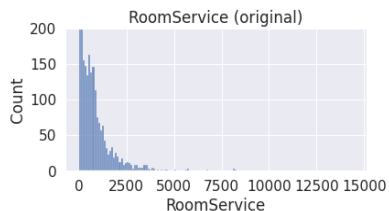
Biểu đồ tần suất về tuổi



Box-plot của tuổi



Ví dụ về phân tích đơn biến





Phân tích hai biến

- ❑ Phân tích hai biến: xác định xem có tồn tại mối liên hệ thống kê giữa hai biến hay không
- ❑ Có ba loại chính:
 - ❑ Phân tích dạng số-số
 - ❑ Phân tích dạng số-phân loại
 - ❑ Phân tích dạng phân loại-phân loại

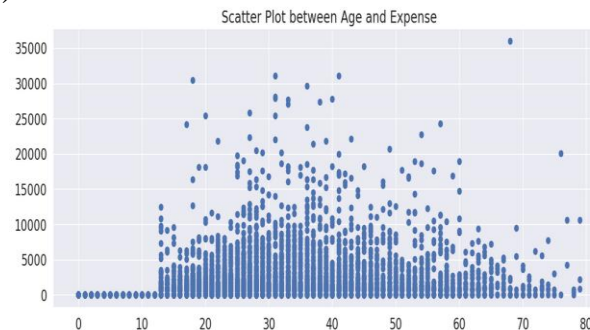


Phân tích dạng số - số

- ❑ Khi cả hai biến được so sánh đều có dữ liệu số
- ❑ Một số phương pháp trực quan có thể được sử dụng:
 - ❑ Biểu đồ phân tán (Scatter plot): được sử dụng để thể hiện mọi điểm dữ liệu trong biểu đồ
 - ❑ Biểu đồ cặp (Pair plot)
 - ❑ Ma trận tương quan (Correlation matrix)



Ma trận tương quan kết hợp bản đồ nhiệt



Biểu đồ phân tán giữa tuổi và chi tiêu



Phân tích dạng số - phân loại

- ❑ Khi một biến có kiểu số và biến khác là biến phân loại
- ❑ Bạn có thể nhóm lại để sắp xếp dữ liệu thành các nhóm tương tự. Các hàng có cùng giá trị trong một cột cụ thể sẽ được sắp xếp thành một nhóm với nhau

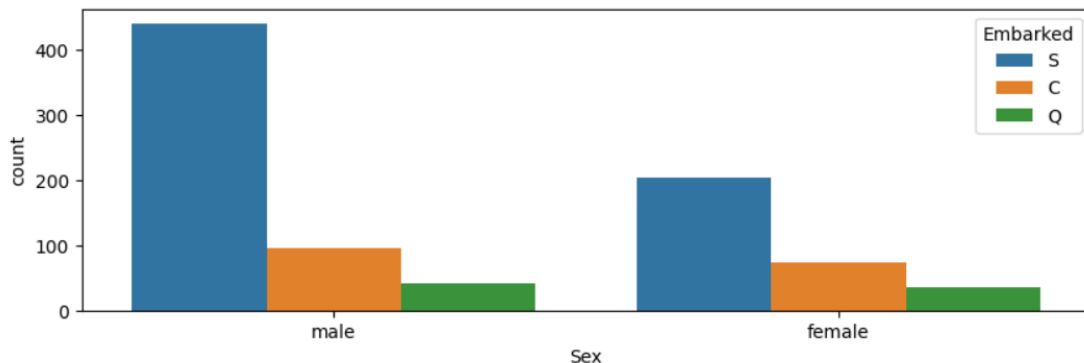
```
1 df_train[['Survived', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare',  
2          | , 'Embarked']].groupby(['Survived', 'Embarked']).mean()
```

		Pclass	Age	SibSp	Parch	Fare
Survived	Embarked					
0	C	2.200000	33.666667	0.253333	0.253333	35.443335
	Q	2.936170	30.325000	0.510638	0.276596	13.335904
	S	2.545667	30.203966	0.611241	0.348946	20.743987
1	C	1.634409	28.973671	0.494624	0.451613	79.720926
	Q	2.866667	22.500000	0.300000	0.000000	13.182227
	S	1.967742	28.113184	0.493088	0.539171	39.547081



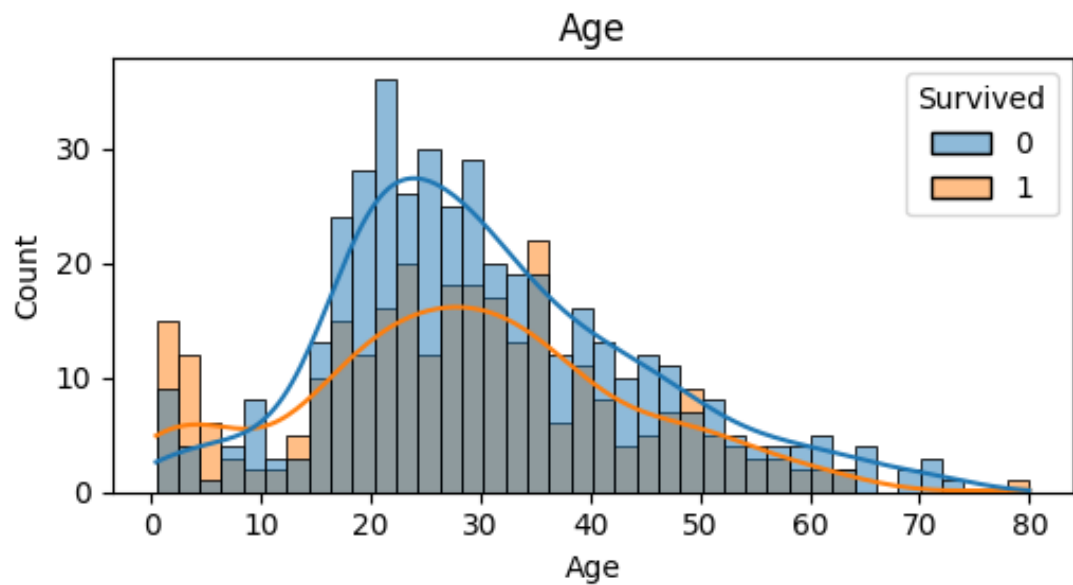
Phân tích dạng phân loại – phân loại

- ❑ Khi cả hai biến đều có tính phân loại
- ❑ Một số phương pháp trực quan có thể được sử dụng:
 - ❑ Biểu đồ cột xếp chồng (Stacked Bar Chart hay Segmented Bar Chart)
 - ❑ Biểu đồ cột tạo cụm (Cluster Bar Chart)



Phân bố của giới tính và cảng khởi hành

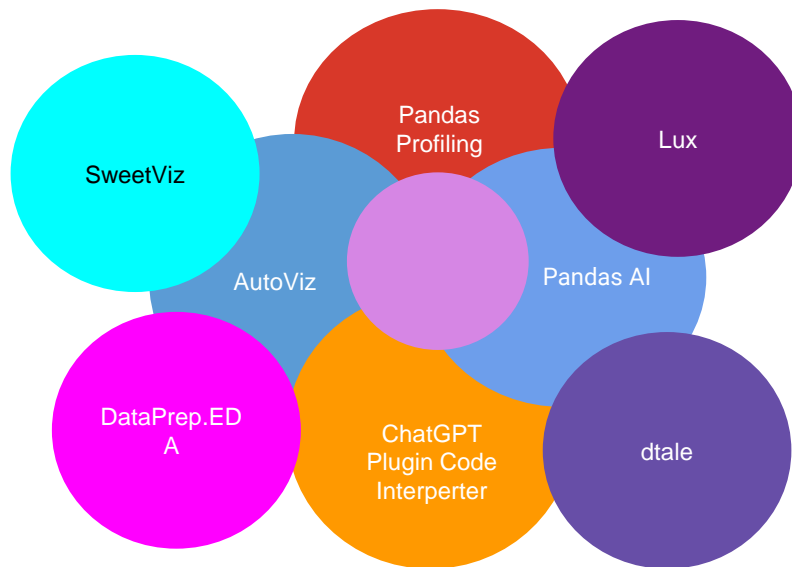
Thực hiện bởi Trường Đại học Công nghệ Thông tin, ĐHQG-HCM





Một số công cụ Automatic EDA

❑ Comparison notebook





Ưu điểm của Automatic EDA tools

01	Hiệu quả	<ul style="list-style-type: none">• Phân tích dữ liệu nhanh chóng và tạo ra cái nhìn sâu sắc• Cho phép Nhà khoa học dữ liệu dành ít thời gian hơn cho việc khám phá sơ bộ
02	Tính toàn diện	<ul style="list-style-type: none">• Quét qua mọi thành phần trong bộ dữ liệu• Góc nhìn toàn diện và đảm bảo rằng không thành phần nào bị bỏ sót.
03	Sự tiện lợi	<ul style="list-style-type: none">• Tóm tắt và trực quan hóa mà không yêu cầu mã hóa rõ ràng cho từng cái
04	Kiểm tra chất lượng dữ liệu	<ul style="list-style-type: none">• Kiểm tra chất lượng dữ liệu: thiếu giá trị, ngoại lệ, không nhất quán
05	Hiểu rõ hơn về dữ liệu	<ul style="list-style-type: none">• Giúp hiểu được cấu trúc cơ bản của dữ liệu, mối tương quan và xu hướng trong dữ liệu.



Pandas Profiling

Overview

Overview

Warnings 14

Reproduction

Dataset statistics

Number of variables	12
Number of observations	891
Missing cells	866
Missing cells (%)	8.1%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	83.7 KiB
Average record size in memory	96.1 B

Variable types

Numeric	5
Categorical	7



Pandas profiling

Tổng quan

Overview

Overview

Warnings 14

Reproduction

Warnings

Name has a high cardinality: 891 distinct values	High cardinality
Ticket has a high cardinality: 681 distinct values	High cardinality
Cabin has a high cardinality: 147 distinct values	High cardinality
Age has 177 (19.9%) missing values	Missing
Cabin has 687 (77.1%) missing values	Missing
PassengerId is uniformly distributed	Uniform
Name is uniformly distributed	Uniform
Ticket is uniformly distributed	Uniform
Cabin is uniformly distributed	Uniform
PassengerId has unique values	Unique
Name has unique values	Unique
SibSp has 608 (68.2%) zeros	Zeros
Parch has 678 (76.1%) zeros	Zeros
Fare has 15 (1.7%) zeros	Zeros



Pandas profiling

Thiếu giá trị

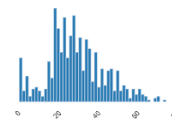
Age

Real number ($\mathbb{R}_{\geq 0}$)

MISSING

Distinct	88
Distinct (%)	12.3%
Missing	177
Missing (%)	19.9%
Infinite	0
Infinite (%)	0.0%

Mean	29.69911765
Minimum	0.42
Maximum	80
Zeros	0
Zeros (%)	0.0%
Memory size	7.1 KiB



Toggle details

Statistics

Histogram

Common values

Extreme values

Quantile statistics

Minimum	0.42
5-th percentile	4
Q1	20.125
median	28
Q3	38
95-th percentile	56
Maximum	80
Range	79.58
Interquartile range (IQR)	17.875

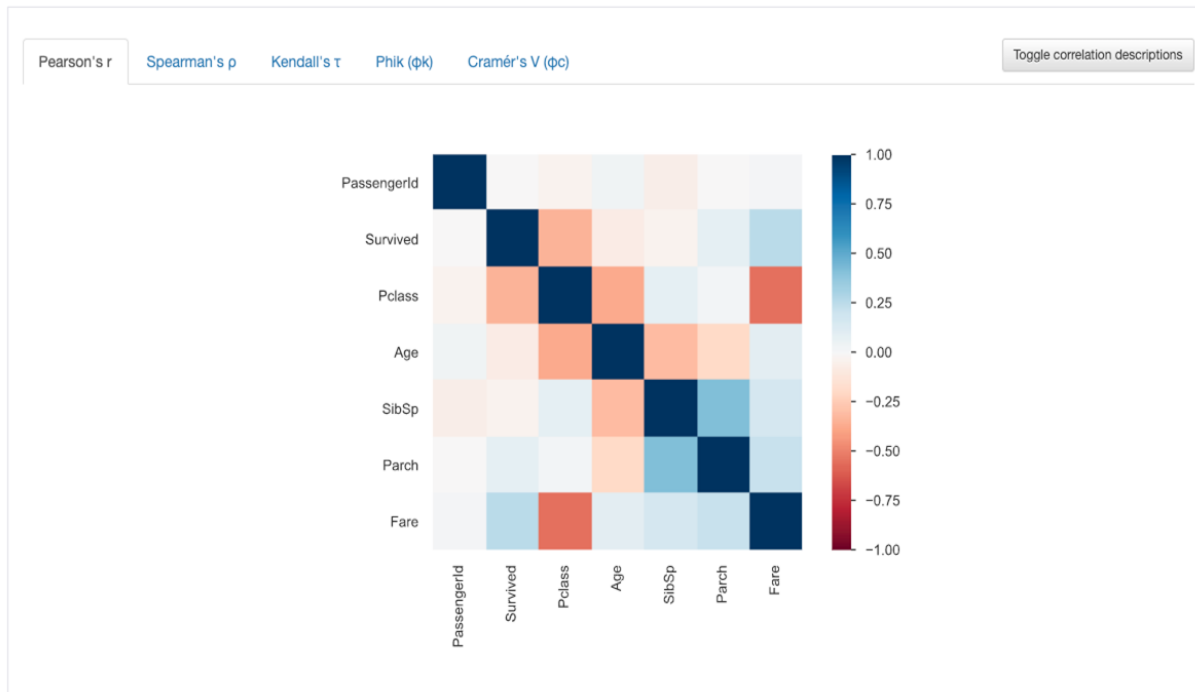
Descriptive statistics

Standard deviation	14.52649733
Coefficient of variation (CV)	0.4891221855
Kurtosis	0.1782741536
Mean	29.69911765
Median Absolute Deviation (MAD)	9
Skewness	0.3891077823
Sum	21205.17
Variance	211.0191247
Monotocity	Not monotonic



Pandas profiling

Sự tương quan

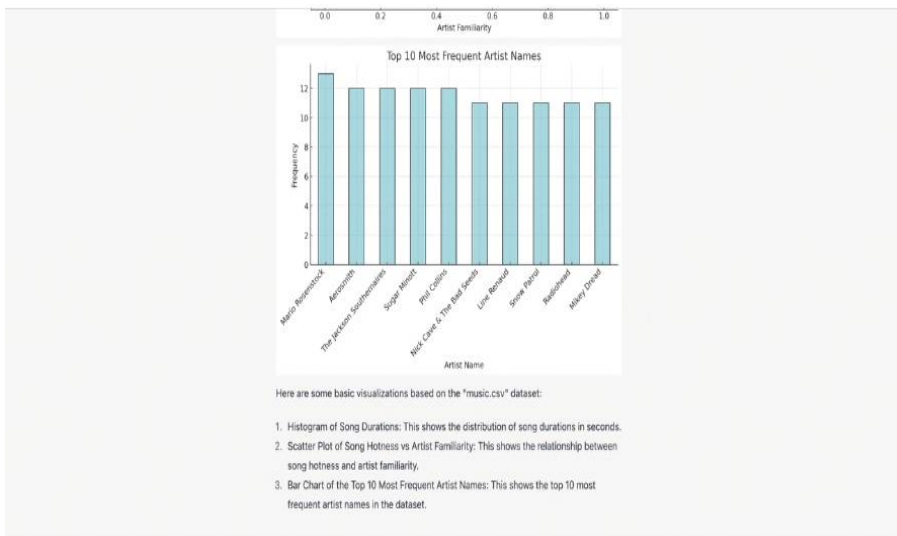




Pandas profiling

Trình thông dịch

- ❑ Thực hiện phân tích và trực quan hóa dữ liệu



Regenerate response

Model
Code Interpreter ALPHA

Code Interpreter ALPHA
An experimental model that can use Python, and handles uploads and downloads

Plugins ALPHA
Default (GPT-3.5)
GPT-4
Legacy (GPT-3.5)
Default (GPT-3.5) with browsing ALPHA
Code Interpreter ALPHA

ChatGPT PLUS



Pandas AI

- ❑ Pandas AI là một thư viện Python bổ sung các khả năng AI tổng quát cho Pandas
- ❑ Làm cho Pandas có thể trò chuyện, cho phép bạn đặt câu hỏi về dữ liệu của mình
- ❑ Github: [link](#)

Installation

```
pip install pandasai
```



Tiền xử lý Dữ liệu

Xử lý các giá trị bị thiếu và ngoại lệ



Các loại giá trị bị thiếu

Dữ liệu bị thiếu về mặt cấu trúc

Những giá trị này bị thiếu vì lẽ ra chúng không tồn tại.

Dữ liệu thiếu hoàn toàn ngẫu nhiên

MCAR (Missing Completely At Random)

Các giá trị bị thiếu xảy ra hoàn toàn ngẫu nhiên.

Dữ liệu thiếu ngẫu nhiên

MAR (Missing At Random)

Giả định ở đây là các giá trị còn thiếu có liên quan phần nào đến các quan sát khác trong dữ liệu.

Dữ liệu thiếu không ngẫu nhiên

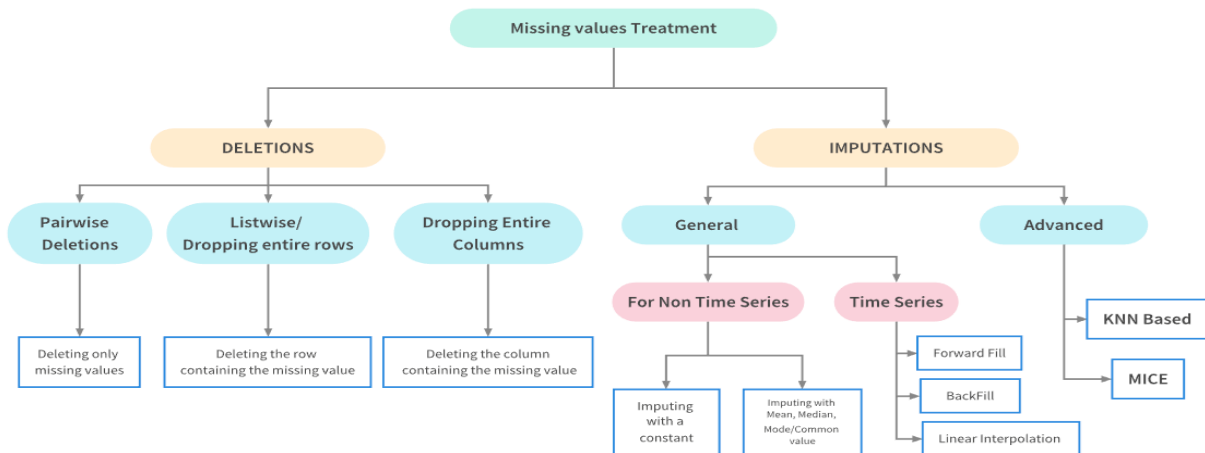
NMAR (Not Missing At Random)

Các giá trị còn thiếu ở đây có nguồn gốc không phải ngẫu nhiên mà có chủ ý



Xử lý các giá trị bị thiếu

- ❑ Các cách tiếp cận chính để xử lý vấn đề thiếu giá trị:
 - ❑ Bằng cách loại bỏ
 - ❑ Bằng cách thay thế: [sklearn-imputation of missing values](#)
 - ❑ Các mô hình (Công cụ ước tính) có thể xử lý giá trị NaN
 - ❑ Cách tiếp cận khác: tạo cột mới chứa thông tin có giá trị bị thiếu





Loại bỏ theo danh sách

- ❑ Loại bỏ theo danh sách
 - ❑ Loại bỏ toàn bộ hàng hoặc cột chứa giá trị còn thiếu
 - ❑ Phương pháp này đơn giản nhưng có thể dẫn đến mất dữ liệu nếu giá trị bị thiếu không phải là MCAR.





Loại bỏ theo cặp

- ❑ Loại bỏ theo cặp
 - ❑ Được sử dụng khi giá trị là MCAR hoặc MAR. Trong quá trình loại bỏ theo cặp, chỉ những giá trị còn thiếu mới bị loại.
 - ❑ Phương pháp này tối đa hóa dữ liệu được sử dụng trong phân tích nhưng có thể dẫn đến kết quả không nhất quán nếu việc thiếu dữ liệu không phải là ngẫu nhiên.





Thay thế đặc trưng đơn biến

- Các giá trị bị thiếu có thể được thay bằng một giá trị hằng số được cung cấp hoặc sử dụng số liệu thống kê (trung bình, trung vị hoặc giá trị thường xuyên nhất) của mỗi cột chứa các giá trị bị thiếu.

```
# imputing with a constant
```

```
from sklearn.impute import SimpleImputer
train_constant = train.copy()
#setting strategy to 'constant'
mean_imputer = SimpleImputer(strategy='constant') # imputing using constant value
train_constant.iloc[:, :] = mean_imputer.fit_transform(train_constant)
train_constant.isnull().sum()
```

```
from sklearn.impute import SimpleImputer
train_most_frequent = train.copy()
#setting strategy to 'mean' to impute by the mean
mean_imputer = SimpleImputer(strategy='most_frequent') # strategy can also be mean or median
train_most_frequent.iloc[:, :] = mean_imputer.fit_transform(train_most_frequent)
```




Thay thế đặc trưng đa biến

- ❑ Một chiến lược để gán các giá trị bị thiếu bằng cách mô hình hóa từng đặc trưng có giá trị bị thiếu dưới dạng hàm của các đặc trưng khác theo kiểu vòng tròn.
- ❑ Nó thực hiện nhiều phép hồi quy trên mẫu dữ liệu ngẫu nhiên, sau đó lấy trung bình của các giá trị hồi quy và sử dụng giá trị đó để thay thế giá trị còn thiếu.

```
from sklearn.experimental import enable_iterative_imputer
from sklearn.impute import IterativeImputer
train_mice = train.copy(deep=True)

mice_imputer = IterativeImputer()
train_mice['Age'] = mice_imputer.fit_transform(train_mice[['Age']])
```



Thay thế bằng phương pháp K-Nearest Neighbor

- ❑ Thay thế các giá trị còn thiếu bằng cách sử dụng phương pháp K-Nearest Neighbors.
- ❑ Mỗi đặc trưng bị thiếu được gán bằng cách sử dụng các giá trị từ n_neighbors hàng xóm gần nhất có giá trị cho đặc trưng đó.
- ❑ Đặc trưng của các hàng xóm được tính trung bình đồng đều hoặc có trọng số theo khoảng cách đến từng hàng xóm.

```
train_knn = train.copy(deep=True)
```

```
from sklearn.impute import KNNImputer  
train_knn = train.copy(deep=True)  
  
knn_imputer = KNNImputer(n_neighbors=2, weights="uniform")  
train_knn['Age'] = knn_imputer.fit_transform(train_knn[['Age']])
```



Thay thế cho dữ liệu chuỗi thời gian

❑ Các kỹ thuật tính toán cơ bản:

- ❑ Điền theo phía trước: Thay thế NaN s bằng giá trị được quan sát lần cuối
- ❑ Điền theo phía sau: Thay thế NaN bằng giá trị được quan sát tiếp theo
- ❑ Phương pháp nội suy tuyến tính

```
df.fillna(method = 'ffill', inplace = True)  
df.fillna(method = 'bfill', inplace = True)  
df.interpolate(limit_direction = "both", inplace = True)
```

2015-02-24	6.05	2015-02-24	6.05
2015-02-25	0.81	2015-02-25	0.81
2015-02-26	NaN	2015-02-26	0.81
2015-02-27	NaN	2015-02-27	0.81
2015-02-28	NaN	2015-02-28	0.81
2015-03-01	1.32	2015-03-01	1.32
2015-03-02	0.22	2015-03-02	0.22

Forward Fill

2015-01-25	NaN
2015-01-26	NaN
2015-01-27	NaN
2015-01-28	NaN
2015-01-29	209.0
2015-01-30	328.0

Back Fill

2015-01-25	209.0
2015-01-26	209.0
2015-01-27	209.0
2015-01-28	209.0
2015-01-29	209.0
2015-01-30	328.0

2015-02-24	6.05	2015-02-24	6.0500
2015-02-25	0.81	2015-02-25	0.8100
2015-02-26	NaN	2015-02-26	0.9375
2015-02-27	NaN	2015-02-27	1.0650
2015-02-28	NaN	2015-02-28	1.1925
2015-03-01	1.32	2015-03-01	1.3200
2015-03-02	0.22	2015-03-02	0.2200

Linear Interpolation



Phương pháp thay thế: Ưu điểm và nhược điểm

Phương pháp	Ưu điểm và nhược điểm
Thay thế đặc trưng đơn biến (mean/median/mode)	<ul style="list-style-type: none">Thay thế giá trị còn thiếu bằng giá trị trung bình, trung vị hoặc xuất hiện thường xuyên nhất của dữ liệu có sẵn cho biến tương ứng.
Thay thế giá trị hằng số	<ul style="list-style-type: none">Thay thế giá trị còn thiếu bằng một giá trị không đổi, ví dụ: "không xác định" đối với các biến phân loại.
Thay thế bằng phương pháp K-Nearest Neighbors	<ul style="list-style-type: none">Thay thế giá trị bị thiếu bằng giá trị trung bình hoặc thường xuyên nhất của K lân cận gần nhất trong không gian đặc trưng.Phương pháp này có thể cung cấp các phép thay thế chính xác hơn nhưng có thể tốn kém về mặt tính toán đối với các tập dữ liệu lớn.



Phương pháp thay thế: Ưu điểm và nhược điểm

Phương pháp	Ưu điểm và nhược điểm
Phép nội suy tuyến tính	<ul style="list-style-type: none">Thay thế giá trị bị thiếu bằng giá trị được nội suy tuyến tính dựa trên các điểm dữ liệu không bị thiếu lân cận.Giả sử mối quan hệ tuyến tính giữa các điểm dữ liệu và có thể không phù hợp với tất cả các loại dữ liệu
Thay thế bằng phương pháp hồi qui	<ul style="list-style-type: none">Ước tính giá trị còn thiếu bằng cách khớp mô hình hồi quy sử dụng các biến khác làm yếu tố dự đoán.Cung cấp các phép thay thế chính xác hơn nhưng có thể gây ra hiện tượng đa cộng tuyến và quá khớp nếu các giá trị được gán có mối tương quan cao với các yếu tố dự đoán khác.
Thay thế dựa trên mô hình	<ul style="list-style-type: none">Sử dụng mô hình ML để ước tính các giá trị còn thiếu dựa trên dữ liệu được quan sát.Phương pháp này có thể cung cấp các phép thay thế có độ chính xác cao nhưng có thể phức tạp hơn và tốn kém hơn về mặt tính toán.



Giảm dữ liệu

- ❑ Giảm dữ liệu mà không mất thông tin
- ❑ Đường dẫn để thảo luận: [How to Reduce Data Size \(50GB to 6GB\)](#)
- ❑ Memory Trick: [Reduce Memory 8x or 16x](#)
- ❑ Hàm giảm bộ nhớ: [pandas reduce memory function](#)



Giảm dữ liệu

step by step

Kiểu dữ liệu	Hướng tiếp cận
Chuỗi ID (String ID)	(chuỗi có độ dài 64 sử dụng 64 byte mỗi hàng) → chuyển đổi int32 hoặc int64 chỉ sử dụng 4 byte hoặc 8 byte (ví dụ: ID khách hàng, ..)
Thời gian (Datetime)	Giảm 10 byte xuống còn 3 byte: ngày theo thời gian (chuyển đổi bằng cách sử dụng gấu trúc pd.to_datetime())
Phân loại (Categorical)	Giảm 88 byte xuống còn 11 byte: cột phân loại (chuyển đổi thành int8)
Dạng số (Numerical)	Giảm 1416 byte xuống còn 353 byte (float64 thành float32,..)



HỎI ĐÁP

