ĐẠI HỌC ĐÀ NẴNG

TRƯỜNG ĐẠI HỌC BÁCH KHOA

CHƯƠNG 5: XỬ LÝ DỮ LIỆU

5.2 Chuẩn hoá dữ liệu

Khoa Công nghệ thông tin

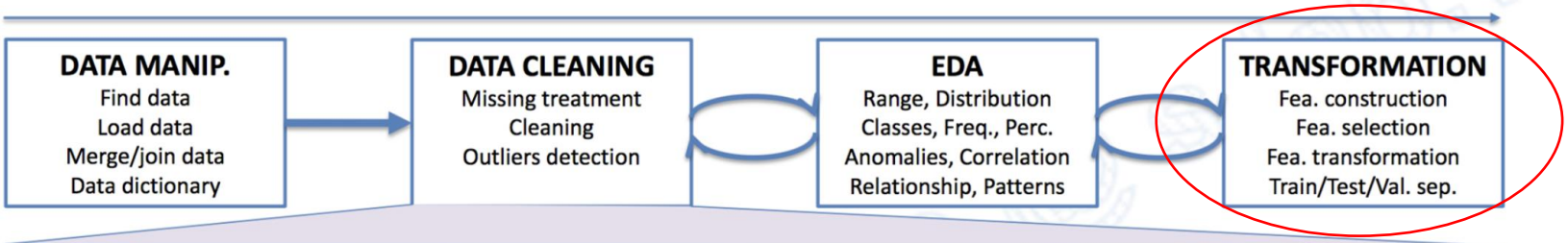
Tài liệu tham khảo

- Các khoá học về KHDL và học máy trên internet

Nội dung

5.2. Chuẩn hoá dữ liệu (hay Biến đổi đặc trưng - Feature transformation)

DATA CLEANING



Missing : Data is not available	
Type of Missing	**Solution**
Small % of observation	- Leave missing (treat as a category) - Delete missing - Single value or Model-based imputation
High % of observation	- Leave missing (treat as a category) or Consider remove fea.
Small % at multiple fea.	- Delete missing and contact data source people

Cleaning: Data is available but looks strange (Use EDA to detect)	
Type of problems:	**Solution**
String: - Typing error: 123a,... - Wrongly data manip. - Non-unique value: Hanoi or Ha noi or Ha_noi or HaNoi - Abbrev.: Chicago or IL	- Correct manually (using script) and/or contact data source people - Check previous step (data manip.)
Format/type: date, temperature, numeric vs string	- Convert manually using script

Outliers: Data is available but looks irregular (Use EDA to detect)	
Outlier detection	**Reason and treatment**
Unreasonable value: Negative age, extremely high/low value	- Input errors -> treat as missing and/or contact data source people Scaling error: check unit
Zero value (income, date)	- Probably missing value which is converted in to numeric
Irregular pattern	- Could have valuable information
Others	- Consider Transformation

Easy Data Science @ s-lab.vn - khoahocdulieudedang@gmail.com – Follow us on Facebook **Học Machine Learning và Deep Learning**

Tại sao phải chuẩn hoá dữ liệu?

- Tập dữ liệu chứa các đặc trưng (biến) khác nhau về giá trị, đơn vị, dải giá trị
- Mỗi quan sát thường được biểu diễn thành 1 điểm trong không gian vec-tơ nhiều chiều với số chiều bằng số đặc trưng
- Hầu hết các thuật toán học máy (để mô hình hoá dữ liệu) đều dùng khoảng cách Euclid giữa 2 điểm để tính toán

→ các đặc trưng sẽ có mức độ đóng góp vào hàm khoảng cách này khác nhau phụ thuộc vào giá trị (magnitude) của chúng

- Điều này không công bằng đối với đặc trưng có giá trị nhỏ vì một sự biến thiên nhỏ của đặc trưng này có thể có ý nghĩa tương đương với một sự biến thiên lớn của một đặc trưng khác có giá trị lớn

→ Cần phải làm cho các đặc trưng có giá trị tương đương nhau

- KHDL gọi là **chuẩn hoá dữ liệu**, học máy gọi là **biến đổi đặc trưng**

Các phương pháp chuẩn hoá dữ liệu

1. Chuẩn hoá theo z-score
2. Chuẩn hoá Min-Max
3. Chuẩn hoá mạnh với ngoại lệ (robust to outliers)
4. Các kỹ thuật biến đổi dữ liệu khác

Chuẩn hoá theo z-score

- Thay thế giá trị của đặc trưng bằng z-score của nó:

$$x' = \frac{x - \bar{x}}{\sigma}$$

PP này làm cho đặc trưng có mean = 0 và std = 1 sau khi chuẩn hoá.

- Dùng hàm **fit_transform()** của lớp **StandardScaler** của thư viện **sklearn.preprocessing**

Chuẩn hoá theo z-score (tt)

Ví dụ trên dataset titanic.csv:

```
import pandas as pd
df=pd.read_csv('titanic.csv', usecols=['Pclass','Age','Fare','Survived'])
df.head()
```

	Survived	Pclass	Age	Fare
0	0	3	22.0	7.2500
1	1	1	38.0	71.2833
2	1	3	26.0	7.9250
3	1	1	35.0	53.1000
4	0	3	35.0	8.0500

Chuẩn hoá theo z-score (tt)

```
# replace missing values of Age
df['Age'].fillna(df.Age.median(),inplace=True)
df.isnull().sum()
```

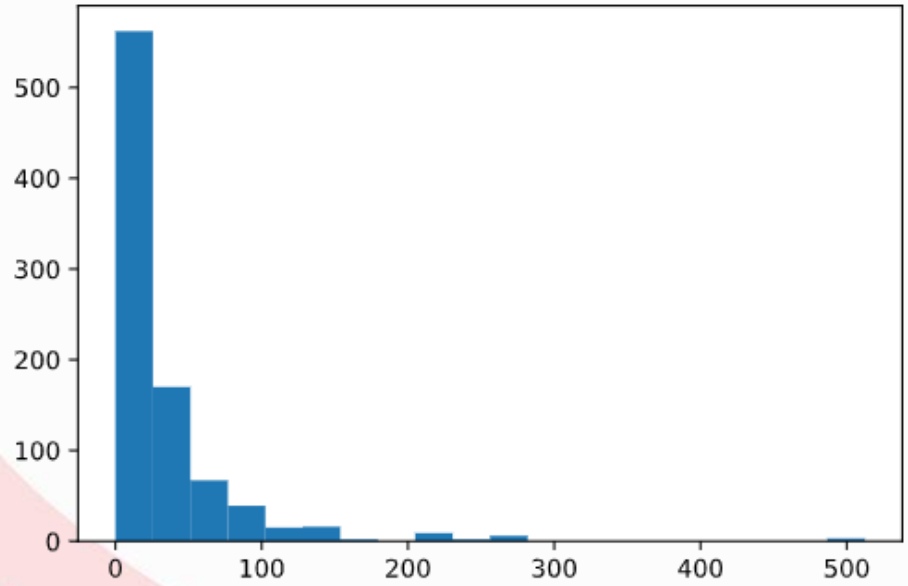
```
Survived    0
Pclass      0
Age         0
Fare        0
dtype: int64
```

```
#### We use the Standardscaler from sklearn library
from sklearn.preprocessing import StandardScaler
Scaler = StandardScaler()
df_scaled = scaler.fit_transform(df)
df_scaled
```

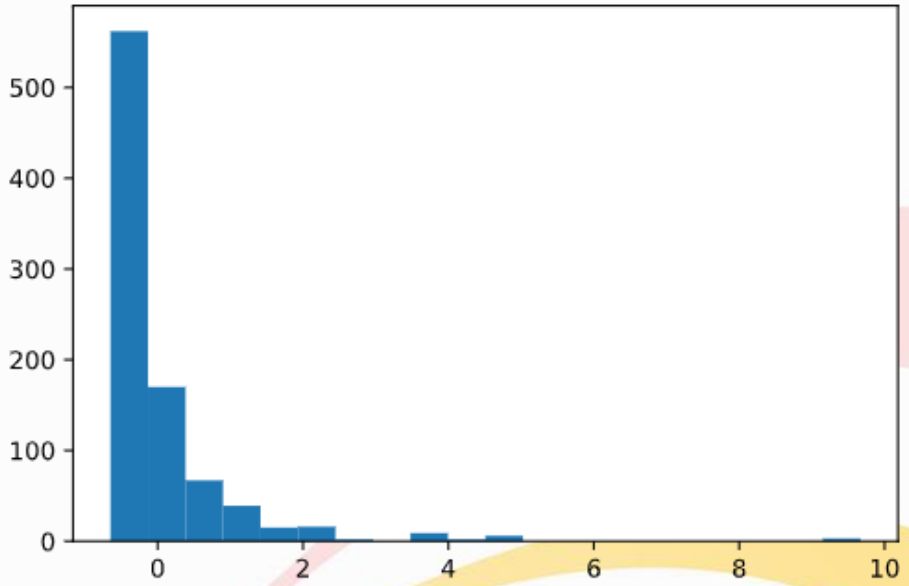
```
array([[-0.78927234,  0.82737724, -0.56573646, -0.50244517],
       [ 1.2669898 , -1.56610693,  0.66386103,  0.78684529],
       [ 1.2669898 ,  0.82737724, -0.25833709, -0.48885426],
       ...,
       [-0.78927234,  0.82737724, -0.1046374 , -0.17626324],
       [ 1.2669898 , -1.56610693, -0.25833709, -0.04438104],
       [-0.78927234,  0.82737724,  0.20276197, -0.49237783]])
```

Chuẩn hoá theo z-score (tt)

```
import matplotlib.pyplot as plt
plt.hist(df['Fare'],bins=20)
plt.hist(df_scaled[:,3],bins=20)
```



Histogram của biến Fare trước chuẩn hoá



Histogram của biến Fare sau chuẩn hoá

Chuẩn hoá Min-Max

- Thay thế giá trị của đặc trưng bằng 1 giá trị thuộc đoạn [0;1]:

$$x' = \frac{x - x_{min}}{x_{\max} - x_{min}}$$

- Dùng hàm **fit_transform()** của lớp **MinMaxScaler** của thư viện **sklearn.preprocessing**

Chuẩn hoá Min-Max (tt)

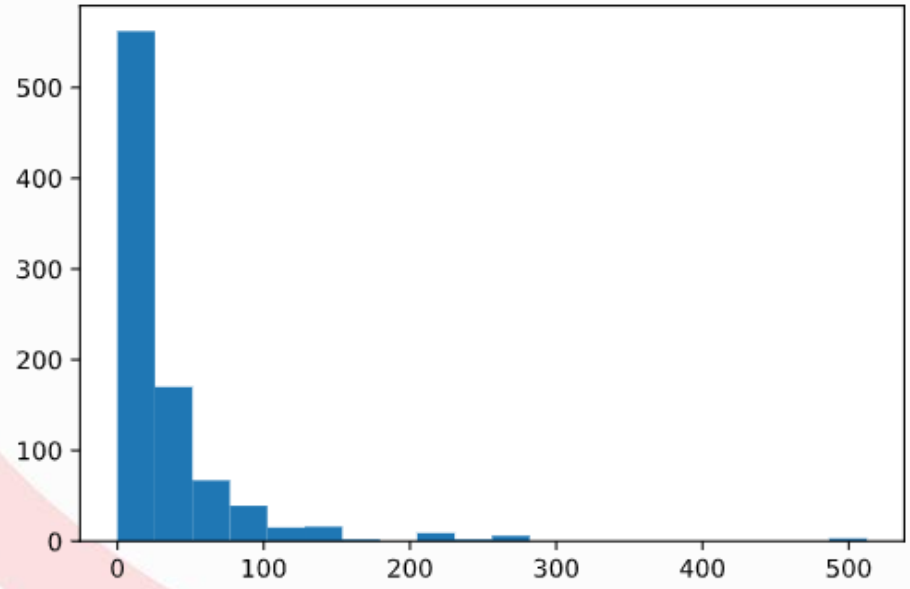
```
from sklearn.preprocessing import MinMaxScaler
min_max=MinMaxScaler()
df_minmax=pd.DataFrame(min_max.fit_transform(df),columns=df.columns)
df_minmax.head()
```

Kết quả sau khi chuẩn hoá:

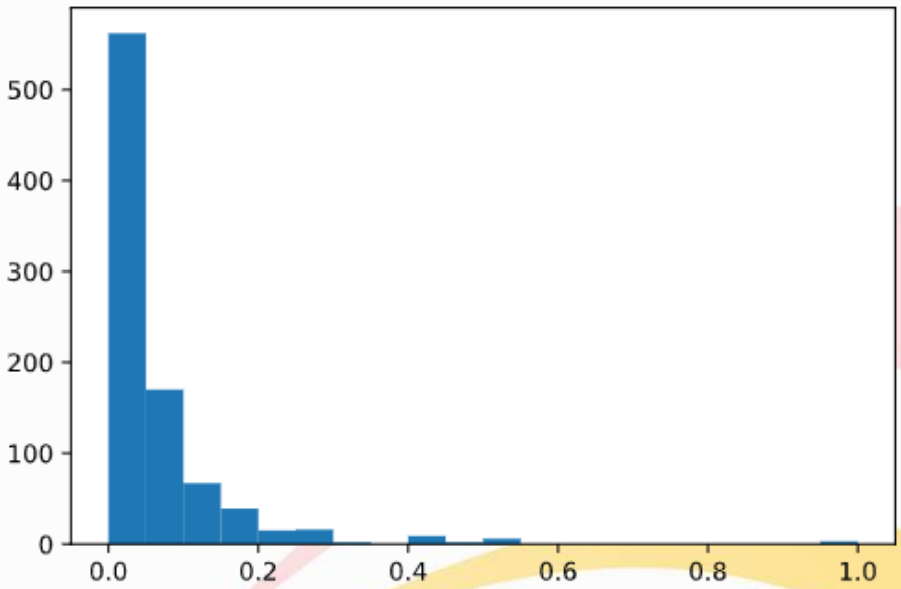
	Survived	Pclass	Age	Fare
0	0.0	1.0	0.271174	0.014151
1	1.0	0.0	0.472229	0.139136
2	1.0	1.0	0.321438	0.015469
3	1.0	0.0	0.434531	0.103644
4	0.0	1.0	0.434531	0.015713

Chuẩn hoá Min-Max (tt)

```
plt.hist(df['Fare'],bins=20)
plt.hist(df_minmax['Fare'],bins=20)
```



Histogram của biến Fare trước chuẩn hoá



Histogram của biến Fare sau chuẩn hoá

Chuẩn hoá mạnh với ngoại lệ

- Dùng các thống kê mẫu ít bị ảnh hưởng bởi các giá trị ngoại lệ là trung vị (median) và IQR (interquartile range) để chuẩn hoá

$$x' = \frac{x - x_{median}}{IQR} = \frac{x - x_{median}}{x_{Q3} - x_{Q1}}$$

với x_{Q1}: 1st quartile của x
x_{Q3}: 3rd quartile của x

- Dữ liệu sau chuẩn hoá có trung vị = 0 và được scale theo IQR

- Dùng hàm **fit_transform()** của lớp **RobustScaler** của thư viện **sklearn.preprocessing**

Chuẩn hoá mạnh với ngoại lệ (tt)

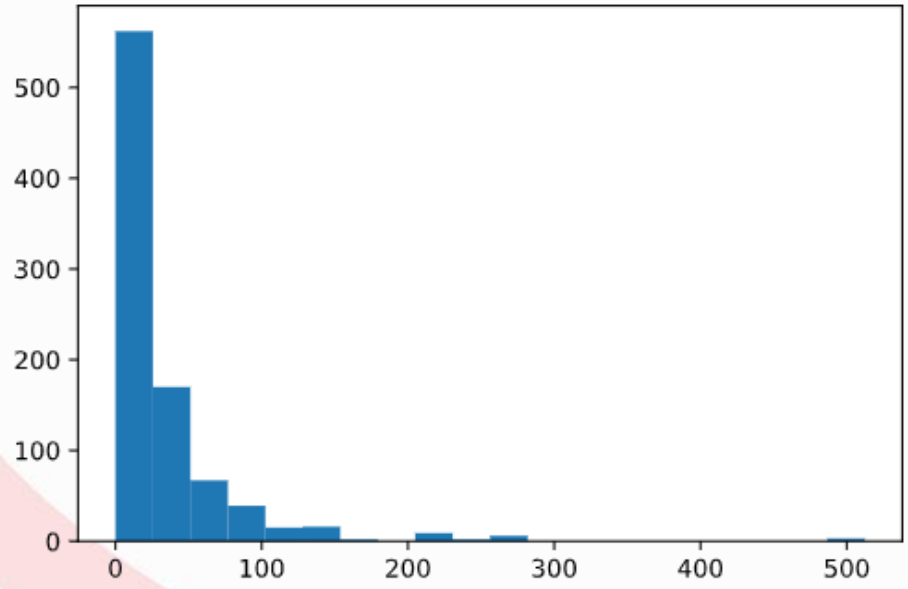
```
from sklearn.preprocessing import RobustScaler
scaler=RobustScaler()
df_robust_scaler=pd.DataFrame(scaler.fit_transform(df),columns=df.columns)
df_robust_scaler.head()
```

Kết quả sau khi chuẩn hoá:

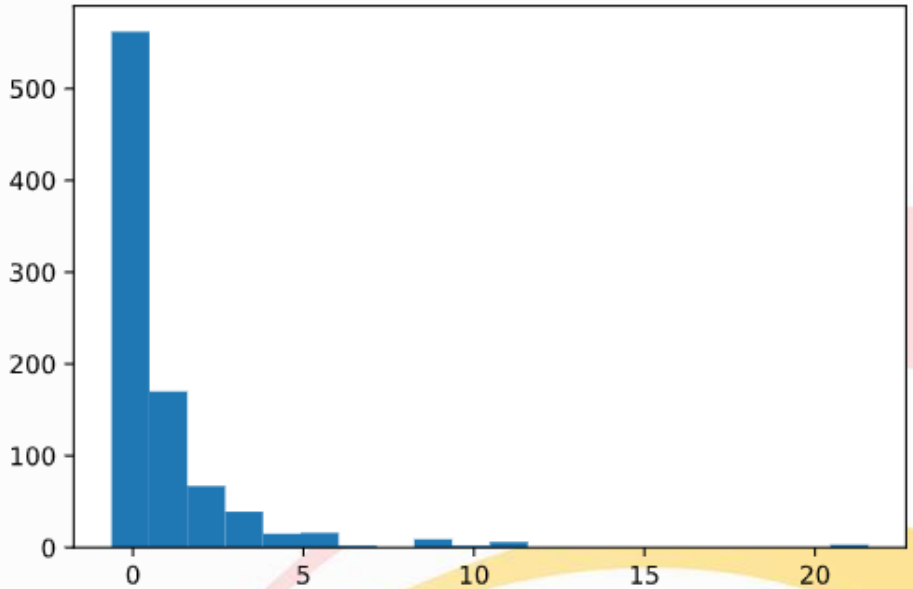
	Survived	Pclass	Age	Fare
0	0.0	0.0	-0.461538	-0.312011
1	1.0	-2.0	0.769231	2.461242
2	1.0	0.0	-0.153846	-0.282777
3	1.0	-2.0	0.538462	1.673732
4	0.0	0.0	0.538462	-0.277363

Chuẩn hoá mạnh với ngoại lệ (tt)

```
plt.hist(df['Fare'],bins=20)
plt.hist(df_robust_scaler['Fare'],bins=20)
```

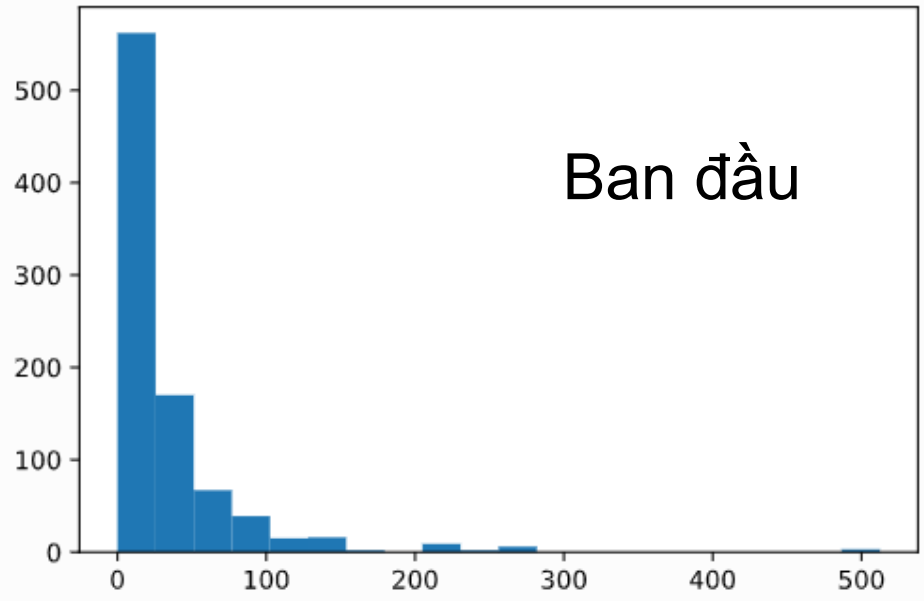


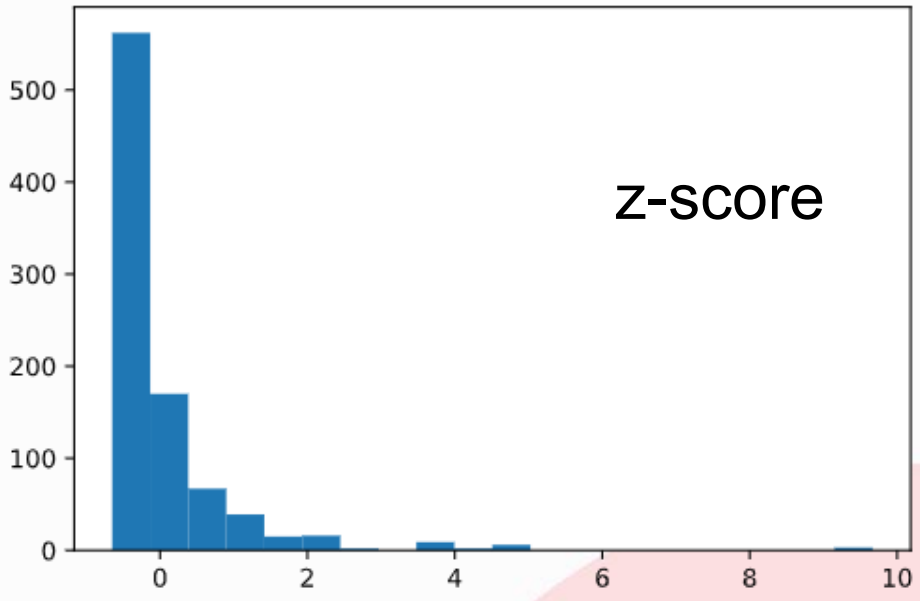
Histogram của biến Fare trước chuẩn hoá

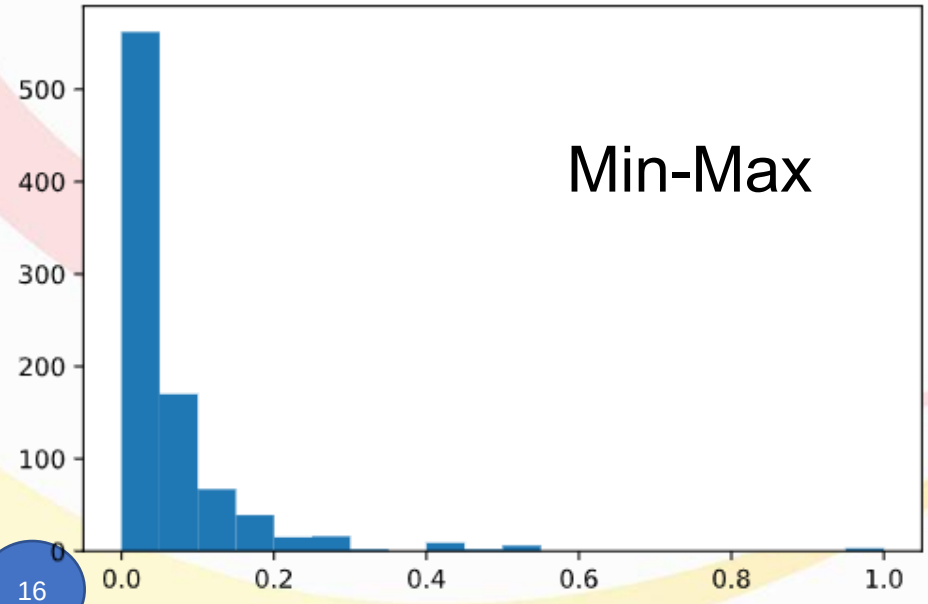


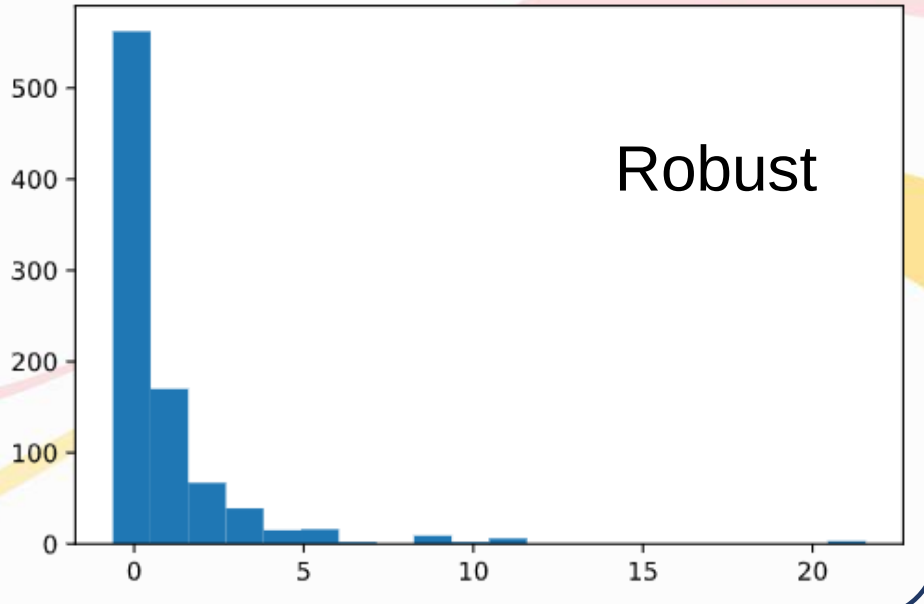
Histogram của biến Fare sau chuẩn hoá

Ví dụ chuẩn hoá với biến Fare

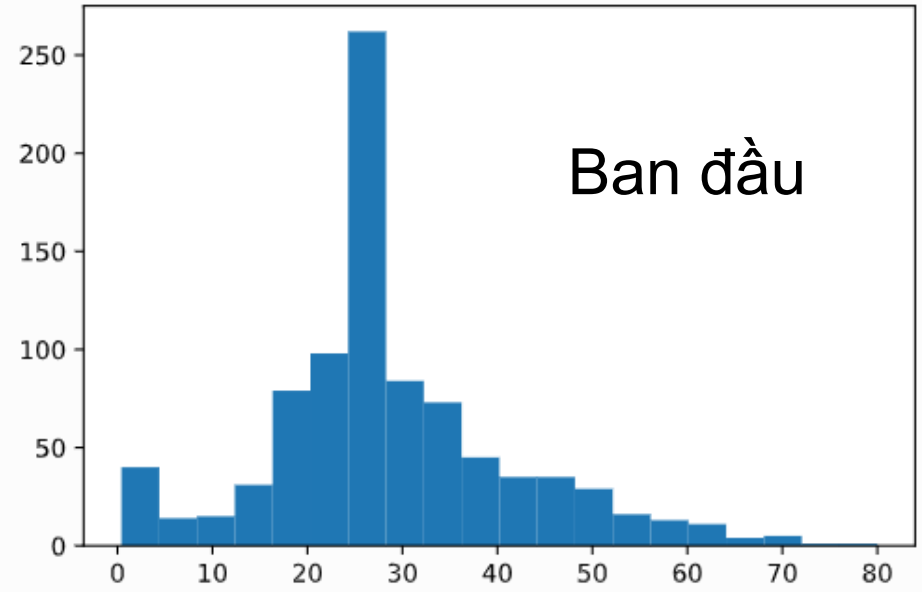


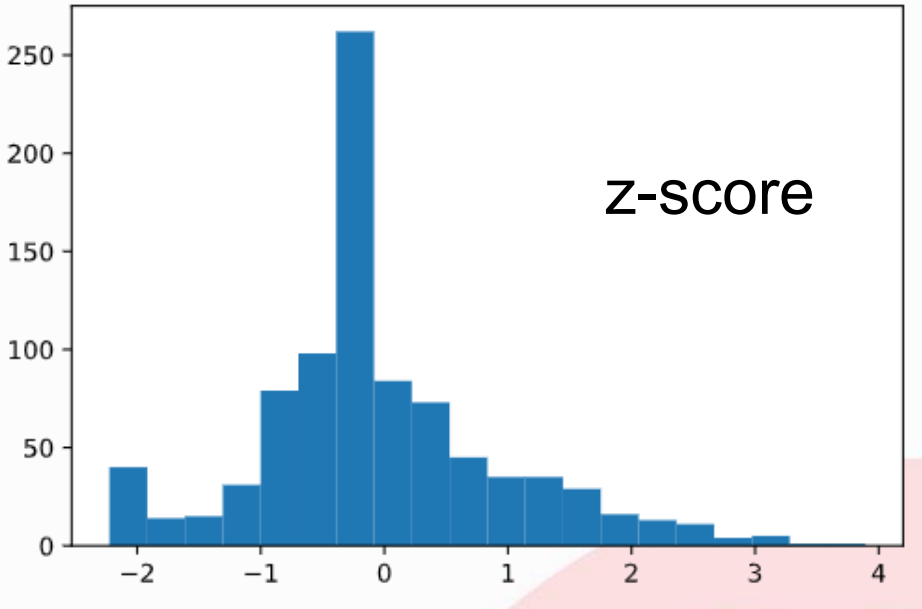


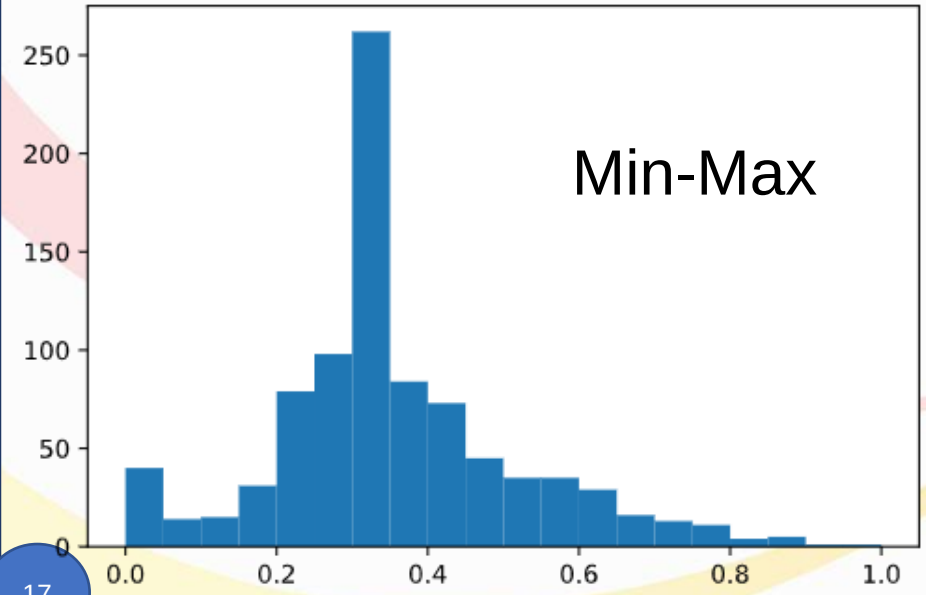


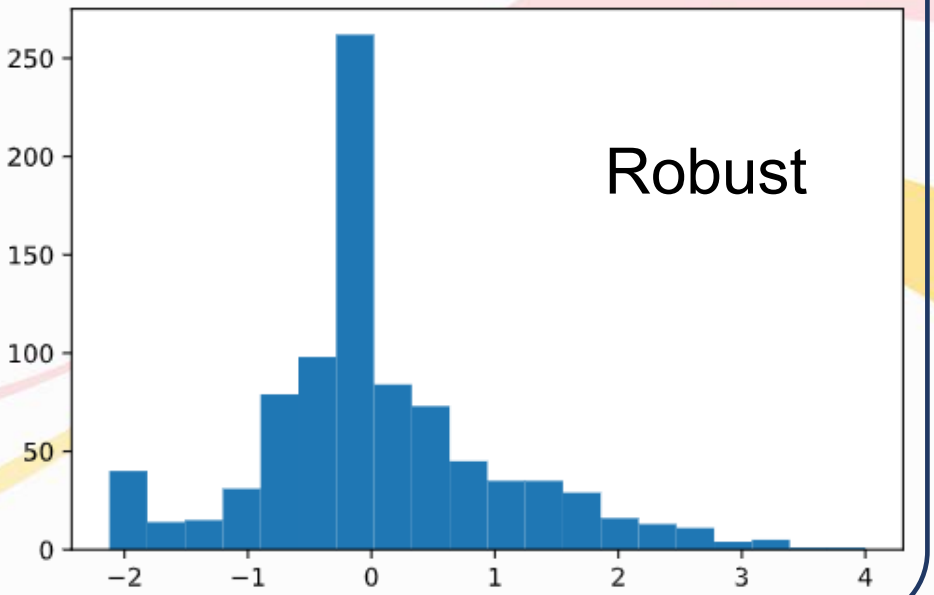


Ví dụ chuẩn hoá với biến Age









Các kỹ thuật biến đổi dữ liệu khác

- Một số thuật toán học máy như hồi quy tuyến tính và hồi quy logistic giả định rằng: **các đặc trưng tuân theo phân bố chuẩn**

→Cần biến đổi dữ liệu sao cho đặc trưng sau khi bị biến đổi thoả mãn điều kiện trên

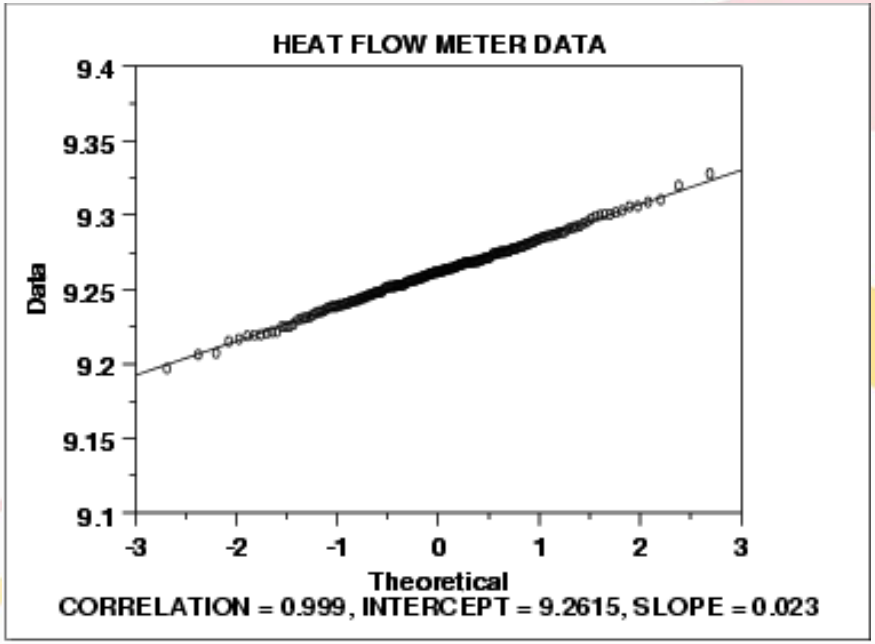
- Làm thế nào để kiểm chứng một đặc trưng có tuân theo phân bố chuẩn hay không?

→ Dùng đồ thị xác suất (probability plot) áp dụng cho phân bố chuẩn

Đồ thị xác suất

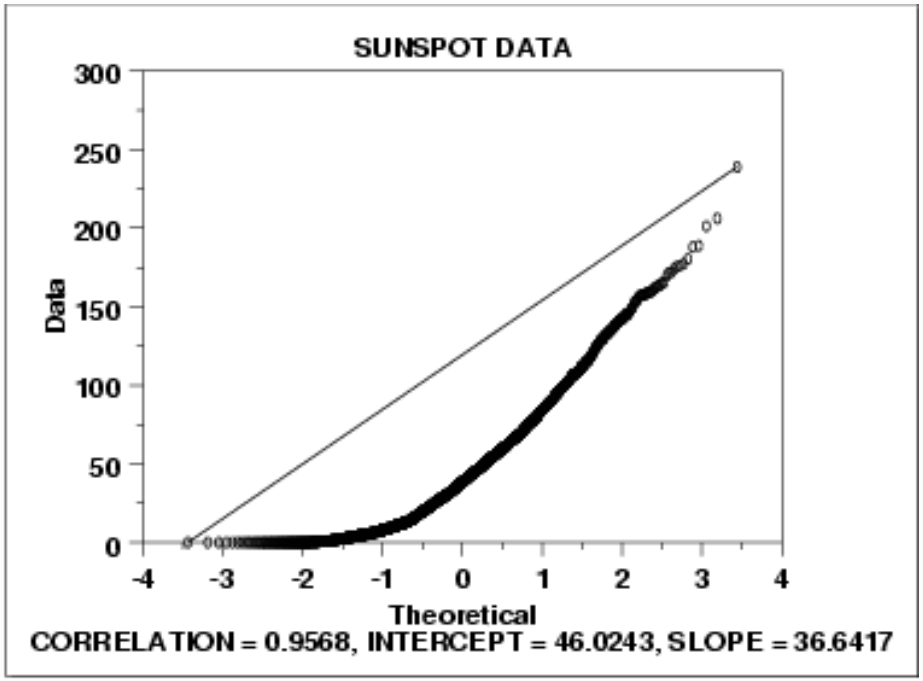
- Là kỹ thuật dùng đồ thị để đánh giá tập dữ liệu có tuân theo một phân bố lý thuyết nào đó (vd: phân bố chuẩn) hay không
- Dữ liệu được vẽ theo phân bố lý thuyết theo cách mà các điểm dữ liệu phải tạo thành một đường (gần như) thẳng
- Độ lệch của các điểm dữ liệu khỏi đường thẳng này tỉ lệ thuận với độ lệch khỏi hàm phân bố lý thuyết đã chỉ định

Ví dụ đồ thị xác suất của một tập dữ liệu tuân theo phân bố chuẩn

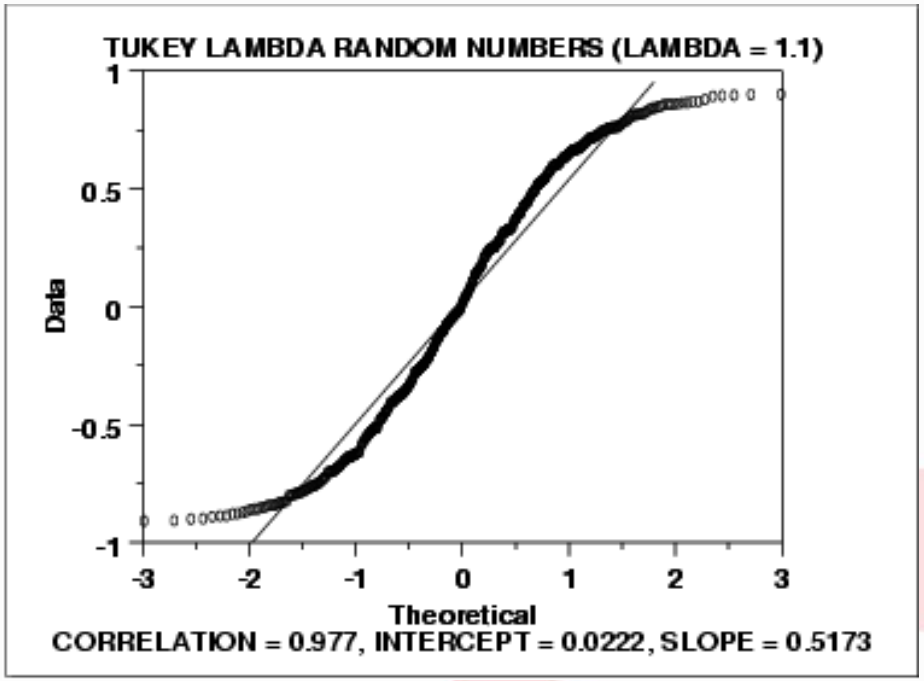


(Nguồn: https://www.itl.nist.gov/div898/handbook/eda/section3/eda33m.htm)

Đồ thị xác suất (tt)



Ví dụ đồ thị xác suất của một tập dữ liệu có phân bố lệch phải (right-skewed)



Ví dụ đồ thị xác suất của một tập dữ liệu có phân bố đuôi ngắn (short tails)

(Nguồn: https://www.itl.nist.gov/div898/handbook/eda/section3/eda33m.htm)

Các kỹ thuật biến đổi dữ liệu khác (tt)

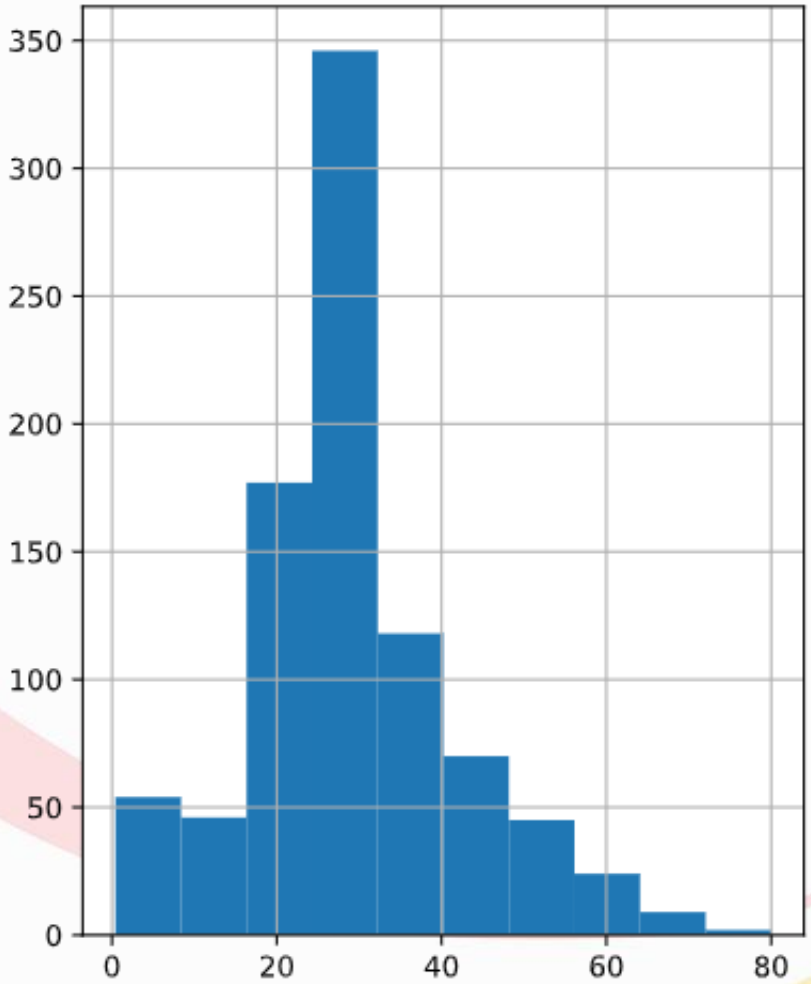
- Nếu đặc trưng không tuân theo phân bố chuẩn, có thể áp dụng 1 trong các kỹ thuật biến đổi dữ liệu sau:
  - Lấy logarit
  - Lấy nghịch đảo
  - Lấy căn bậc 2
  - Lấy luỹ thừa
  - Biến đổi Box-Cox
- Sau đó dùng đồ thị xác suất để kiểm chứng dữ liệu sau khi bị biến đổi có tuân theo phân bố chuẩn không

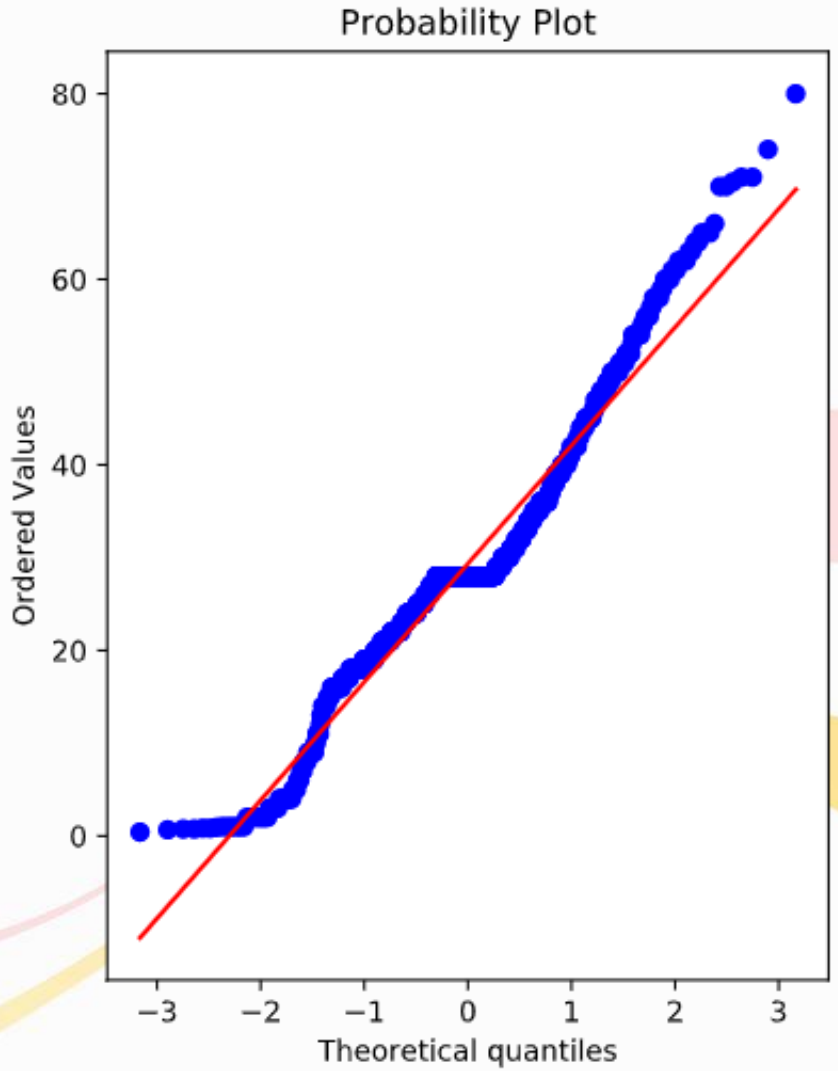
Ví dụ với dataset Titanic

```
import scipy.stats as stat
import matplotlib.pyplot as plt
#### function to check whether feature is normally distributed
def plot_data(df,feature):
    plt.figure(figsize=(10,6))
    plt.subplot(1,2,1)
    df[feature].hist() # histogram
    plt.subplot(1,2,2)
    stat.probplot(df[feature],dist='norm',plot=plt)# prob plot
    plt.show()
```

Ví dụ với dataset Titanic (tt)

```
plot_data(df,'Age')
```

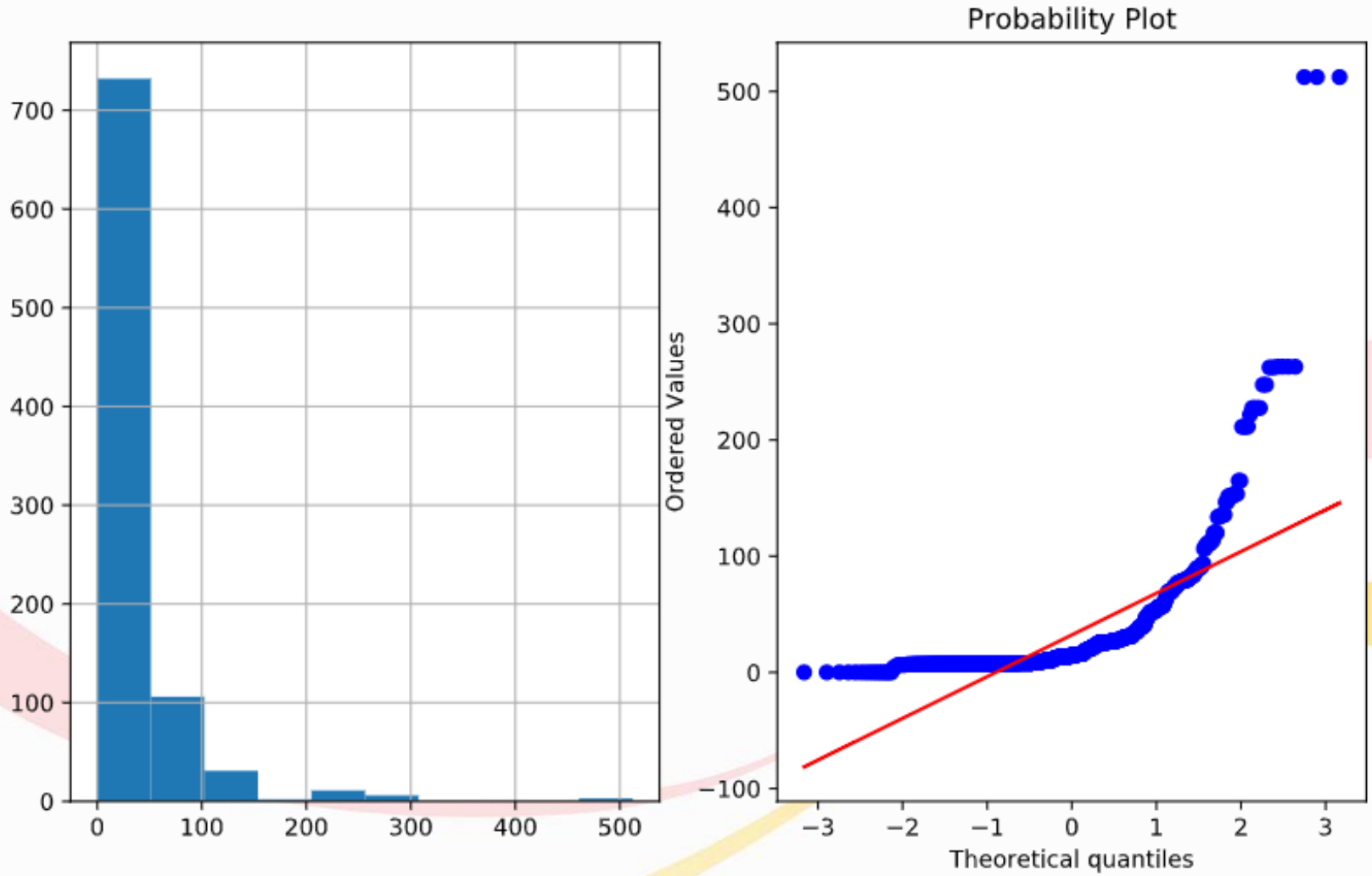




Histogram và đồ thị xác suất của biến Age → đã tương đối theo phân bố chuẩn

Ví dụ với dataset Titanic (tt)

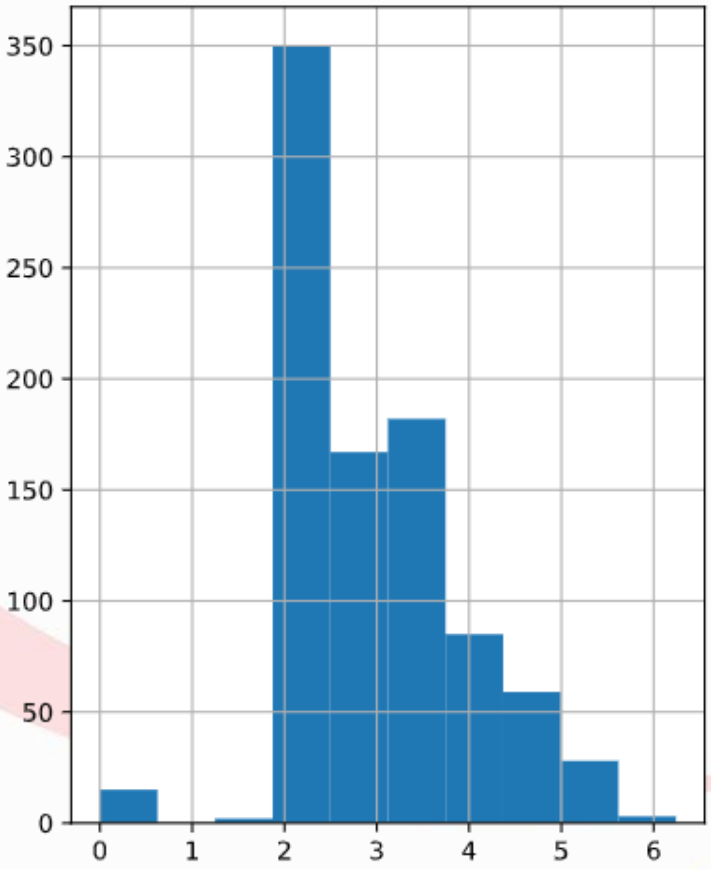
```
plot_data(df,'Fare')
```

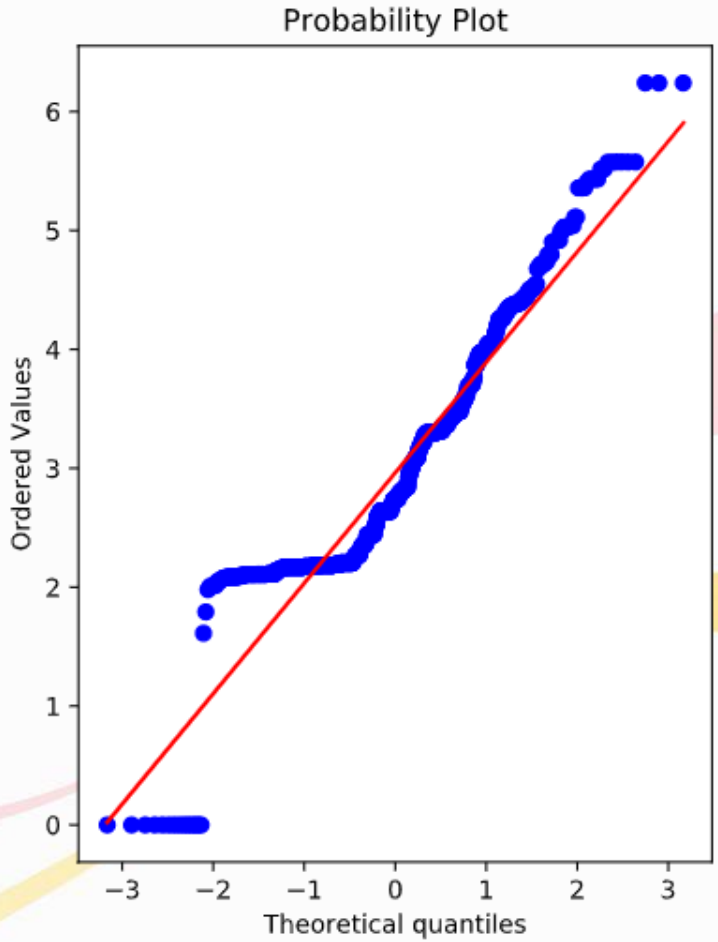


Histogram và đồ thị xác suất của biến Fare → không theo phân bố chuẩn

Ví dụ với dataset Titanic (tt)

```
#### apply Logarithmic transformation on Fare
df['Fare_log']=np.log1p(df['Fare'])
plot_data(df,'Fare_log')
```

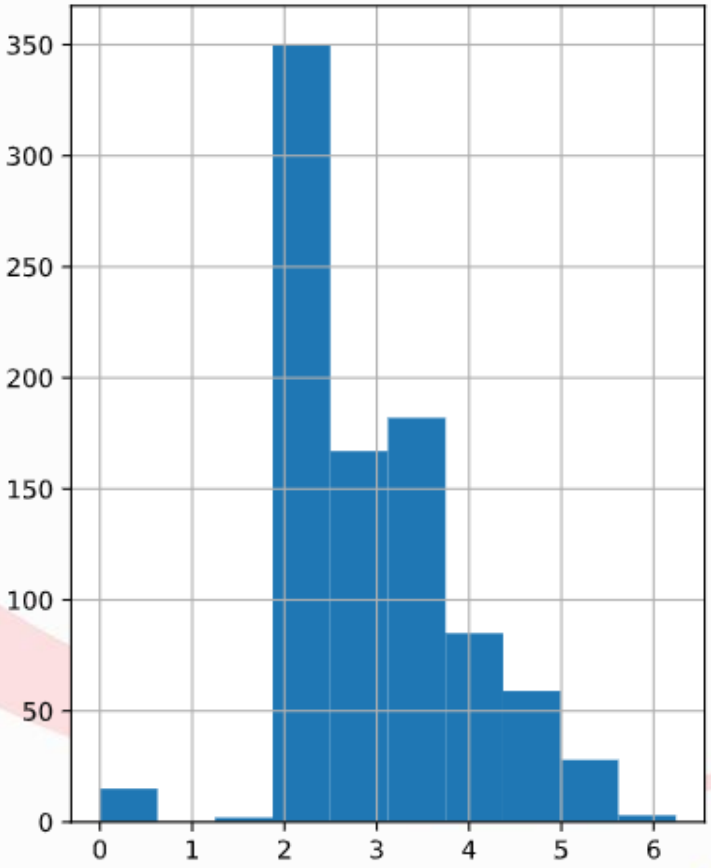


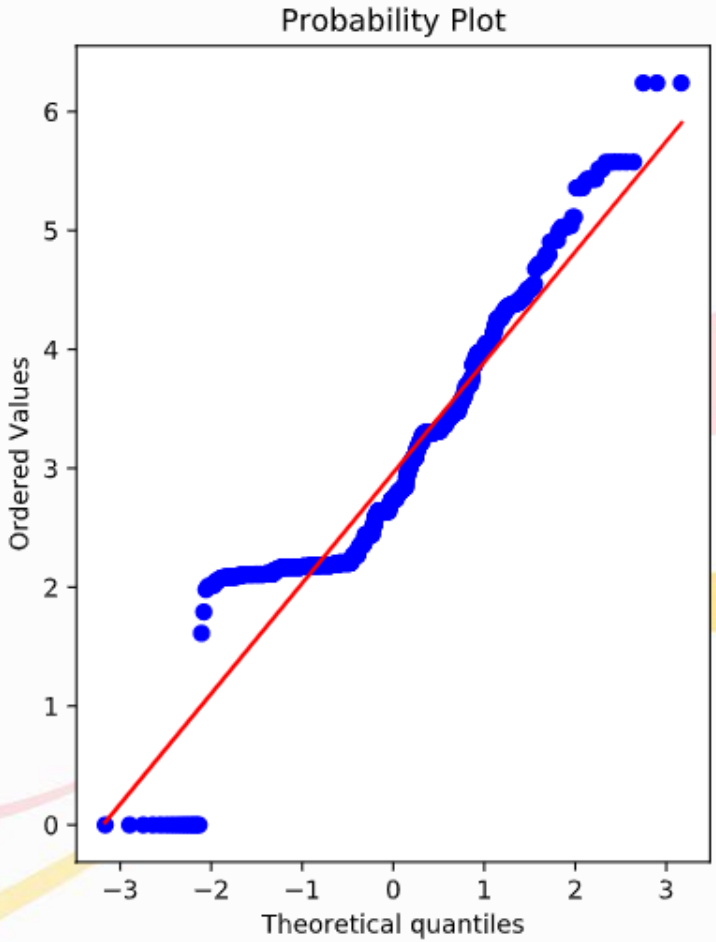


Histogram và đồ thị xác suất của biến Fare_log → gần với phân bố chuẩn hơn

Ví dụ với dataset Titanic (tt)

```
#### apply Box-Cox transformation on Fare
df['Fare_Boxcox'],parameters=stat.boxcox(df['Fare']+1)
plot_data(df,'Fare_Boxcox')
```





Histogram và đồ thị xác suất của biến Fare_Boxcox → gần với phân bố chuẩn hơn

Bài tập (Phần 5.1)

Viết chương trình đánh giá độ chính xác của thuật toán hồi quy logistic trong bài toán dự báo Sự sống/chết (Survived) của hành khách lên tàu Titanic dựa trên Độ tuổi (Age) và Giá vé (Fare) mà họ đã mua trong 14 tổ hợp làm sạch dữ liệu sau:

- 7 kỹ thuật xử lý dữ liệu trống (Median/Mean/Mode imputation coi như 3 kỹ thuật khác nhau, kỹ thuật tạo đặc trưng mới phải sử dụng đồng thời biến Age và biến bổ sung Age_NAN để dự báo)
- 2 trường hợp Không xử lý ngoại lệ & Có xử lý ngoại lệ (đồng thời trên 2 biến Age và Fare)

Lập bảng báo cáo độ chính xác của thuật toán trong 14 trường hợp trên và cho biết độ chính xác cao nhất và thấp nhất xảy ra trong trường hợp nào. Phân tích vì sao độ chính xác cao/thấp như vậy (Train/Test phân theo tỉ lệ 70/30 như chương trình mẫu, mỗi trường hợp cần thử nghiệm 10 lần với random_state=0 đến 9 và lấy trung bình để được độ chính xác trung bình của thuật toán).

Bài tập (Phần 5.2)

- Đề xuất các kỹ thuật chuẩn hoá dữ liệu phù hợp để cải thiện độ chính xác của thuật toán hồi quy logistic trong bài toán dự báo Sự sống/chết (Survived) của hành khách lên tàu Titanic dựa trên Độ tuổi (Age) và Giá vé (Fare) mà họ đã mua.
- Cho biết độ cải thiện (%) về độ chính xác dự báo khi áp dụng kỹ thuật chuẩn hoá dữ liệu đã đề xuất kết hợp với 1 trong 14 tổ hợp kỹ thuật làm sạch dữ liệu của Bài tập Phần 5.1.