THỰC HÀNH: PHÂN TÍCH KHÁM PHÁ DỮ LIỆU

A. MỤC TIÊU

- Bài thực hành này nhằm giúp người học nắm vững các kỹ thuật cơ bản trong khám phá dữ liệu để hiểu rõ đặc điểm và cấu trúc của tập dữ liệu. Cụ thể, sinh viên sẽ thực hiện các bước phân tích thống kê mô tả để xác định các đặc trưng chính như giá trị trung bình, trung vị, độ lệch chuẩn và phân bố của dữ liệu.
- Đồng thời, bài thực hành hướng dẫn sử dụng các công cụ trực quan hóa như biểu đồ histogram, boxplot, và scatter plot để phát hiện các mẫu, xu hướng, hoặc bất thường trong dữ liệu. Sinh viên sẽ được làm quen với các thư viện Python như Pandas, Matplotlib, và Seaborn để xử lý và trực quan hóa dữ liệu hiệu quả.
- Ngoài ra, bài thực hành giúp nhận diện các vấn đề như giá trị thiếu, giá trị ngoại lai, hoặc sự không nhất quán trong dữ liệu, từ đó đề xuất các phương pháp tiền xử lý phù hợp. Kết quả cuối cùng là sinh viên có thể đưa ra các nhận định ban đầu về dữ liệu, đặt nền tảng cho các bước phân tích sâu hơn hoặc xây dựng mô hình khai thác dữ liệu trong các ứng dụng thực tiễn như phân tích khách hàng hoặc dự đoán xu hướng.

B. KẾT CẤU THỰC HÀNH

Thực hành bao gồm 3 phần là

- Thống kê mô tả
- Xử lý và trực quan hóa dữ liệu
- Phân tích đơn biến và hai biến

C. NỘI DUNG THỰC HÀNH

1.1. THỐNG KÊ MÔ TẢ

1.1.1. Ôn tập lý thuyết

+ Thống kê mô tả là gì? Nó khác gì với thống kê suy luận (inferential statistics)?

+ Các thước đo thống kê mô tả chính (ví dụ: trung bình, trung vị, phương sai, độ lệch chuẩn) được sử dụng để làm gì? Trong trường hợp nào thì nên dùng trung vị thay vì trung bình?

+ Làm thế nào để xác định phân bố của một tập dữ liệu? Các loại phân bố phổ biến là gì (ví dụ: phân bố chuẩn, lệch trái, lệch phải)?

+ Độ lệch chuẩn và phạm vi (range) có ý nghĩa gì trong việc đánh giá sự phân tán của dữ liệu?

+ Sự khác biệt giữa các thước đo như Q1, Q2, Q3 trong biểu đồ hộp (boxplot) là gì? + Làm thế nào để xử lý giá trị thiếu (missing values) trước khi tính toán các chỉ số thống kê mô tả?

+ Bạn có thể giải thích cách đọc và diễn giải một biểu đồ histogram hoặc boxplot từ dữ liệu thực tế không?

+ Khi gặp một tập dữ liệu có giá trị ngoại lai (outliers), bạn sẽ xử lý chúng như thế nào trước khi thực hiện thống kê mô tả?

1.1.2. Bài làm mẫu

Bài toán 1: Thực hiện các nhiệm vụ trong bài toán 1 để làm quen với các thao tác cần làm để khám phá dữ liệu

Nhiệm vụ 1: Khám phá dữ liệu COVID lấy tại *https://ourworldindata.org /coronavirus*

1. Tính mean, median, mode, variance, standard deviation, range, percentile, quartile, interquartile range (IQR) sử dụng thư viện numpy và stats trên tập dữ liệu COVID.

```
import numpy as np
import pandas as pd
from scipy import stats
# Load the .csv into a dataframe using read_csv
covid_data = pd.read_csv("covid-data.csv")
covid_data = covid_data[['iso_code','continent',
 'location','date','total_cases','new_cases']] # Take a
quick look at the data
covid_data.head(5)
covid_data.dtypes
covid_data.shape
# Get the mean of the data
data_mean = np.mean(covid_data["new_cases"])
# Get the median of the data
data_median = np.median(covid_data["new_cases"])
# Get the mode of the data
data_mode = stats.mode(covid_data["new_cases"])
# Obtain the variance of the data
data_variance = np.var(covid_data["new_cases"])
# Obtain the standard deviation of the data
data_sd = np.std(covid_data["new_cases"])
# Compute the maximum and minimum values of the data
data_max = np.max(covid_data["new_cases"])
data_min = np.min(covid_data["new_cases"])
# Obtain the 60th percentile of the data
data_percentile =
np.percentile(covid_data["new_cases"],60) # Obtain the
quartiles of the data
data_quartile =
np.quantile(covid_data["new_cases"],0.75) # Get the IQR
of the data
data_IQR = stats.iqr(covid_data["new_cases"])
```

Nhiệm vụ 2: Khám phá và xử lý dữ liệu Marketing Campaign lấy tại https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis

1. Import thư viện và nạp dữ liệu vào notebook

```
import pandas as pd
marketing_data = pd.read_csv("data/marketing_campaign.csv")
marketing_data = marketing_data[['ID','Year_Birth', 'Education',
'Marital_Status','Income','Kidhome', 'Teenhome',   'Dt_ Customer',
'Recency','NumStorePurchases',  'NumWebVisitsMonth']]
```

2. Loại bỏ dữ liệu trùng lặp

```
marketing_data.head()
# Remove duplicates across the columns in our dataset:
marketing_data_duplicate =
marketing_data.drop_duplicates() # Delete a specified row
at index value 1:
marketing_data.drop(labels=[1], axis=0)
# Delete a single column
marketing_data.drop(labels=['Year_Birth'], axis=1)
```

3. Thay thế dữ liệu và thay đổi định dạng của dữ liệu

```
# Replace the values in Teenhome with has teen and has no
teen marketing_data['Teenhome_replaced'] =
 marketing_data['Teenhome'].
 replace([0,1,2],['has no teen','has teen','has teen'])

# Fill NAs in the Income column
marketing_data['Income']   =   marketing_data['Income'].fillna(0)   #
Change  the  data  type  of  the  Income  column  from  float  to  int
marketing_data['Income_changed']                                    =
marketing_data['Income'].astype(int)
```

4. Xử lý dữ liệu thiếu

```
# Check for missing values using the isnull and sum
methods marketing_data.isnull().sum()
# Drop missing values using the dropna method
marketing_data_withoutna = marketing_data.dropna(how =
'any') marketing_data_withoutna.shape
```

1.1.3. Bài tập thực hành 1

Thực hiện thống kê mô tả trên tập dữ liệu về phân loại chất lượng rượu đỏ.

Dữ liệu lấy tại https://www.kaggle.com/code/eisgandar/red-wine-quality-eda-classification

1.1.4. Bài tập thực hành 2

Thực hiện thống kê mô tả trên tập dữ liệu về bệnh tiểu đường. Dữ liệu lấy tại

https://www.kaggle.com/code/vincentlugat/pima-indians-diabetes-eda-prediction-0-906

1.2. XỬ LÝ VÀ TRỰC QUAN HÓA DỮ LIỆU

1.2.1. Ôn tập lý thuyết

+ Trực quan hóa dữ liệu có vai trò gì trong phân tích dữ liệu? Tại sao nó quan trọng trong khám phá dữ

liệu (EDA)?

+ Các loại biểu đồ phổ biến (như histogram, scatter plot, boxplot, bar chart) được sử dụng trong các trường hợp nào?

+ Làm thế nào để chọn loại biểu đồ phù hợp với đặc điểm của dữ liệu (ví dụ: dữ liệu phân loại, dữ liệu số, dữ liệu thời gian)?

+ Sự khác biệt giữa các thư viện trực quan hóa trong Python như Matplotlib, Seaborn và Plotly là gì?

+ Những nguyên tắc thiết kế nào cần tuân thủ để tạo ra một biểu đồ trực quan hóa dễ hiểu và hiệu quả?

+ Làm thế nào để tạo một biểu đồ đơn giản như histogram hoặc bar chart bằng Matplotlib? Bạn có thể chia sẻ đoạn code mẫu không?

+ Làm thế nào để xuất biểu đồ từ Python ra các định dạng như PNG, PDF hoặc HTML để sử dụng trong báo cáo?

1.2.2. Bài làm mẫu

Bài toán 1: Thực hiện các nhiệm vụ trong bài toán để làm quen với các công cụ trực quan hóa dữ liệu. Dữ liệu thực hiện là dữ liệu về giá nhà lấy từ https://www.kaggle.com/datasets/thomasnibb/amsterdam house-price-prediction

Nhiệm vụ 1:

1. Chuẩn bị dữ liệu cho trực quan hóa dữ liệu

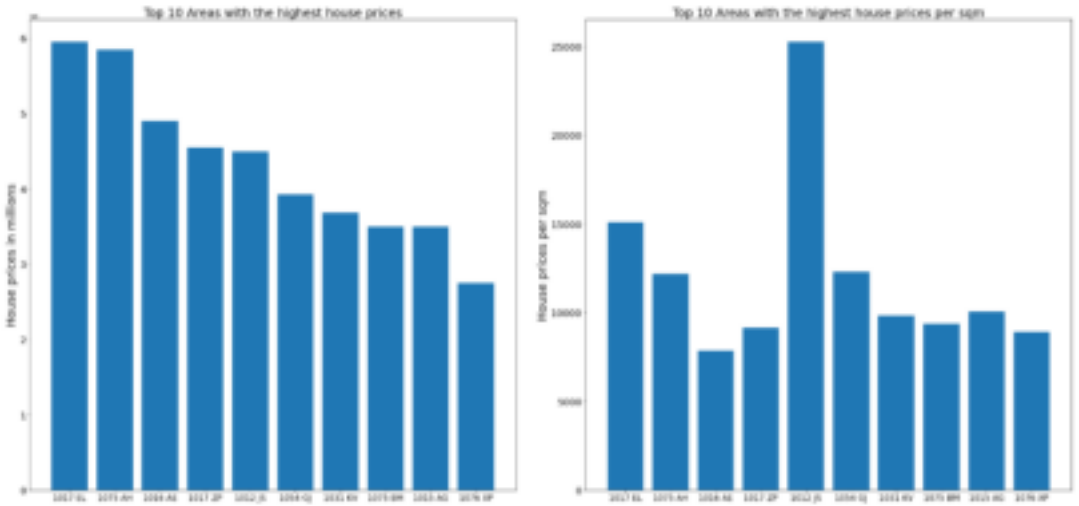
```
import pandas as pd
houseprices_data = pd.read_csv("data/HousingPricesData.csv")
houseprices_data = houseprices_data[['Zip', 'Price', 'Area',
'Room']] # Create a PriceperSqm variable based on the Price and Area
variables: houseprices_data['PriceperSqm'] =
houseprices_data['Price']/  houseprices_data['Area']
```

2. Trực quan hóa dữ liệu với thư viện **Matplotlib**

```
houseprices_sorted = houseprices_data.sort_values('Price', ascending
=  False)
houseprices_sorted.head()
# case 1: basic
plt.figure(figsize= (12,6))
x = houseprices_sorted['Zip'][0:10]
y = houseprices_sorted['Price'][0:10]
```

```
plt.bar(x,y)
plt.show()
# case 2: advanced 1
plt.figure(figsize= (12,6))
plt.bar(x,y)
plt.title('Top 10 Areas with the highest house prices',
fontsize=15) plt.xlabel('Zip code', fontsize = 12)
plt.xticks(fontsize=10)
plt.ylabel('House prices in millions', fontsize=12)
plt.yticks(fontsize=10)
plt.show()
# case 3: advanced 2
fig, ax = plt.subplots(figsize=(40,18))
x = houseprices_sorted['Zip'][0:10]
y = houseprices_sorted['Price'][0:10]
y1 = houseprices_sorted['PriceperSqm'][0:10]
plt.subplot(1,2,1)
plt.bar(x,y)
plt.xticks(fontsize=17)
plt.ylabel('House prices in millions', fontsize=25)
plt.yticks(fontsize=20)
plt.title('Top 10 Areas with the highest house prices',
fontsize=25)
plt.subplot(1,2,2)
plt.bar(x,y1)
plt.xticks(fontsize=17)
plt.ylabel('House prices per sqm', fontsize=25)
plt.yticks(fontsize=20)
plt.title('Top 10 Areas with the highest house prices per
sqm', fontsize=25)
plt.show()
```


Kết quả thực hiện case 3

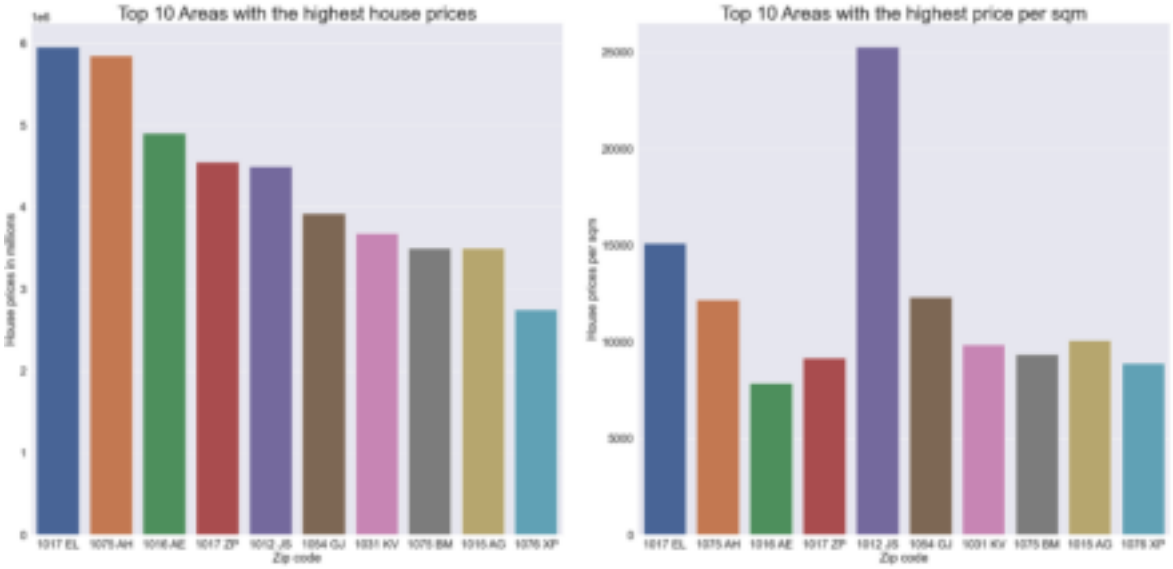


Hình 1.1 - Ảnh biểu đồ của thư viện Matplotlib

3. Trực quan hóa dữ liệu với thư viện **Seaborn**

```
import matplotlib.pyplot as plt
import seaborn as sns
# case 1: basic
plt.figure(figsize= (12,6))
data = houseprices_sorted[0:10]
sns.barplot(data= data, x= 'Zip',y = 'Price')
# case 2: advanced 1
plt.figure(figsize= (12,6))
data = houseprices_sorted[0:10]
ax = sns.barplot(data= data, x= 'Zip',y = 'Price')
ax.set_xlabel('Zip code',fontsize = 15)
ax.set_ylabel('House prices in millions', fontsize = 15)
ax.set_title('Top 10 Areas with the highest house prices', fontsize=
20) # case 3: view multiple perspectives at once
fig, ax = plt.subplots(1, 2,figsize=(40,18))
data = houseprices_sorted[0:10]
sns.set(font_scale = 3)
ax1 = sns.barplot(data= data, x= 'Zip',y = 'Price', ax =
ax[0]) ax1.set_xlabel('Zip code')
ax1.set_ylabel('House prices in millions')
ax1.set_title('Top 10 Areas with the highest house
prices') ax2 = sns.barplot(data= data, x= 'Zip',y =
'PriceperSqm', ax=ax[1])
ax2.set_xlabel('Zip code')
ax2.set_ylabel('House prices per sqm')
ax2.set_title('Top 10 Areas with the highest price per sqm')
```

Kết quả thực hiện case 3



Hình 1.2 - Ảnh biểu đồ của thư viện Seaborn

1.2.2. Bài tập thực hành 1

+ Thực hiện trực quan hóa dữ liệu trên tập dữ liệu về phân loại chất lượng rượu đỏ. Dữ liệu lấy tại https://www.kaggle.com/code/eisgandar/red-wine-quality-eda-classification

1.2.3. Bài tập thực hành 2

+ Thực hiện trực quan hóa dữ liệu trên tập dữ liệu về bệnh tiểu đường. Dữ liệu lấy tại https://www.kaggle.com/code/vincentlugat/pima-indians-diabetes-eda-prediction-0-906

+ Thực hiện EDA trên tập dữ liệu mua sắm tại siêu thị. Tập dữ liệu lấy từ https://www.kaggle.com/code/rajatkumar30/eda-online-retail

1.3. Phân tích đơn biến và hai biến

1.3.1. Ôn lý thuyết

+ Phân tích đơn biến (univariate analysis) là gì? Nó khác gì với phân tích hai biến (bivariate analysis) trong khám phá dữ liệu?

+ Các thước đo thống kê nào thường được sử dụng trong phân tích đơn biến (ví dụ: trung bình, trung vị, mode, độ lệch chuẩn)?

+ Trong phân tích hai biến, làm thế nào để xác định mối quan hệ giữa hai biến (ví dụ: tương quan, nhân quả)?

+ Sự khác biệt giữa tương quan (correlation) và hiệp biến (covariance) trong phân tích hai biến là gì? + Khi nào nên sử dụng biểu đồ trực quan hóa trong phân tích đơn biến so với phân tích hai biến? + Đoạn code mẫu để tạo biểu đồ scatter plot hoặc heatmap để phân tích mối quan hệ giữa hai biến?

+ Làm thế nào để trực quan hóa mối quan hệ giữa một biến số và một biến phân loại bằng biểu đồ boxplot hoặc violin plot trong Python?

1.3.2. Bài làm mẫu

Bài toán 1: Thực hiện các nhiệm vụ trong bài toán 1 để làm quen với các hàm và thư viện hỗ trợ phân tích dữ liệu đơn biến. Bài toán này được thực hiện trên 2 tập dữ liệu là tập dữ liệu về chim cánh cụt và tập dữ liệu giá nhà.

Nhiệm vụ 1: phân tích dữ liệu đơn biến trên dữ liệu về chim cánh cụt lấy tại https://www.kaggle.com/datasets/parulpandey/palmer-archipelago-antarctica-penguin-data

1. Import thư viện và nạp dữ liệu

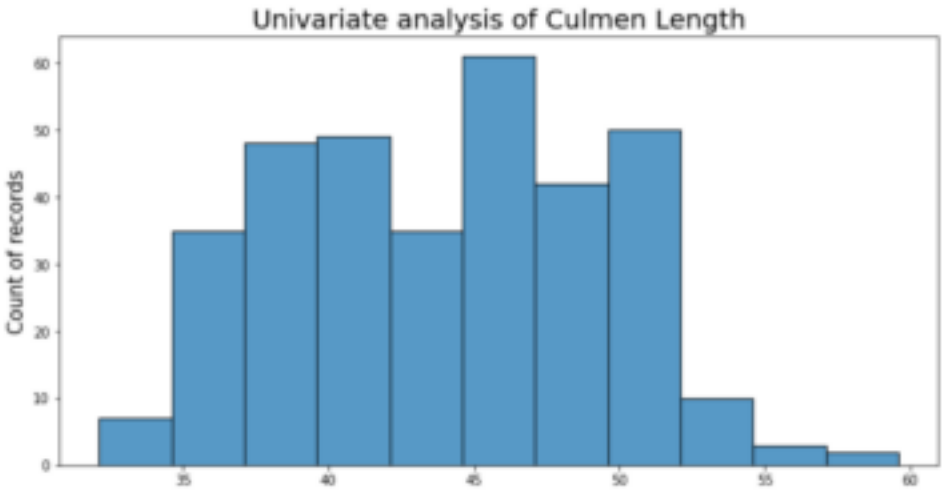
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

penguins_data = pd.read_csv("data/penguins_size.csv")
penguins_data = penguins_data[['species','culmen_length_mm']]
```

2. Phân tích đơn biến bằng Histogram

```
# case 1: basic
sns.histplot( data = penguins_data, x= penguins_
data["culmen_length_mm"])
# case 2: advanced
plt.figure(figsize= (12,6))
ax = sns.histplot( data = penguins_data, x=
 penguins_data["culmen_length_mm"])
```

```
ax.set_xlabel('Culmen Length in mm',fontsize = 15)
ax.set_ylabel('Count of records', fontsize = 15)
ax.set_title('Univariate analysis of Culmen Length',fontsize= 20)
```

Kết quả thực hiện case 2

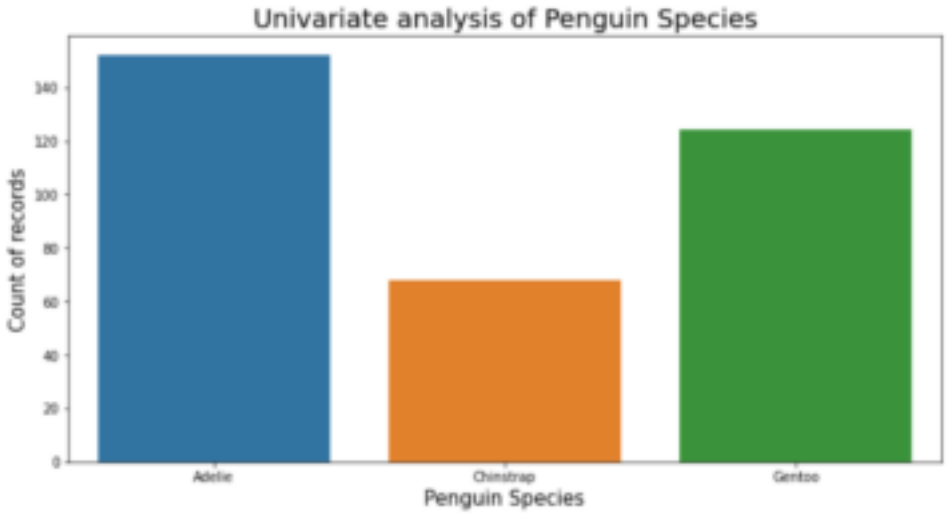


Hình 1.3 - Biểu đồ histogram

3. Phân tích đơn biến bằng bar chart

```
# case 1: basic
sns.countplot(data = penguins_data, x= penguins_data['species'])

# case 2: advanced
plt.figure(figsize= (12,6))
ax = sns.countplot(data = penguins_data, x=
penguins_data['species']) ax.set_xlabel('Penguin Species',fontsize
= 15)
ax.set_ylabel('Count of records',fontsize = 15)
ax.set_title('Univariate analysis of Penguin Species', fontsize=
20) ax.set_title('Univariate analysis of Culmen Length', fontsize=
20)
```


Kết quả thực hiện case 2



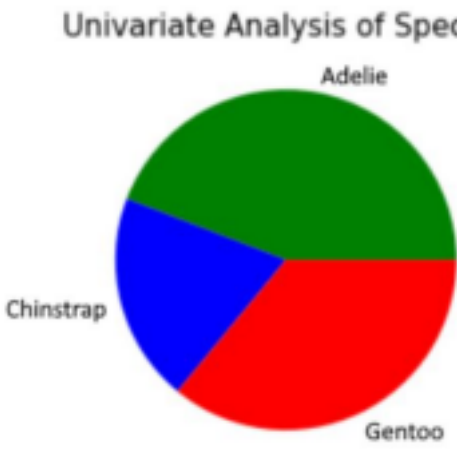
Hình 1.4 - Biểu đồ thanh (bar chart)

4. Phân tích đơn biến bằng biểu đồ tròn (Pie-chart)

```
penguins_group = penguins_data.groupby('species').count()
penguins_group = penguins_group.reset_index()
# case 1: basic
plt.pie(penguins_group["culmen_length_mm"], labels =
 penguins_group['species'])
plt.show()
# case 2: advanced
cols = ['g', 'b', 'r']
plt.pie(penguins_group["culmen_length_mm"], labels =
 penguins_group['species'],colors = cols) plt.title('Univariate
Analysis of Species', fontsize=15)
plt.show()
```


Hình 1.5 - Biểu đồ tròn (pie-chart)

Kết quả thực hiện case 2



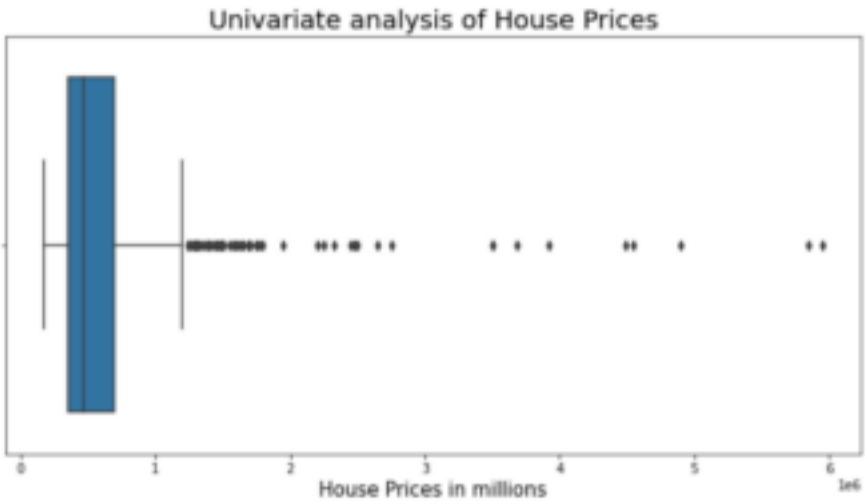
Nhiệm vụ 2: Phân tích dữ liệu đơn biến trên dữ liệu giá nhà lấy từ

https://www.kaggle.com/datasets/thomasnibb/amsterdam-house-price-prediction

1. Import thư viện, nạp dữ liệu giá nhà và phân tích đơn biến dựa vào boxplot

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
houseprices_data = pd.read_csv("Data/HousingPricesData.csv")
houseprices_data =
houseprices_data[['Zip','Price','Area','Room']] # case 1: Create
a boxplot using the boxplot method
sns.boxplot(data = houseprices_data, x= houseprices_data["Price"])

# case 2: advanced
plt.figure(figsize= (12,6))
ax = sns.boxplot(data = houseprices_data, x=
houseprices_data["Price"]) ax.set_xlabel('House Prices in
millions',fontsize = 15)
ax.set_title('Univariate analysis of House Prices', fontsize=
20) plt.ticklabel_format(style='plain', axis='x')
```


Kết quả thực hiện case 2

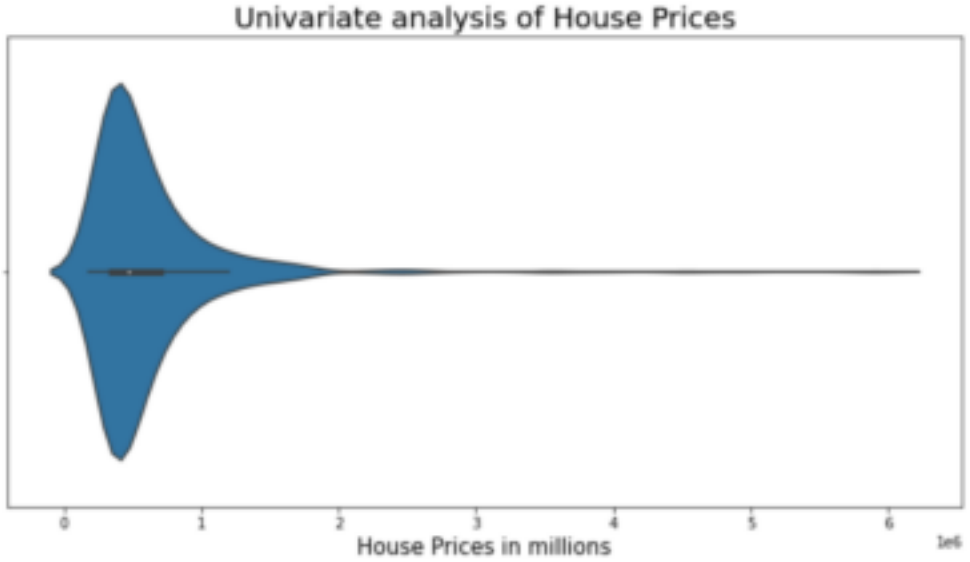


Hình 1.6 - Biểu đồ boxplot (được trình bày phương ngang)

2. Phân tích dữ liệu đơn biến dựa vào violin plot

```
# case 1: basic
sns.violinplot(data = houseprices_data, x=
houseprices_data["Price"]) # case 2: advanced
plt.figure(figsize= (12,6))
ax = sns.violinplot(data = houseprices_data,
x=houseprices_data["Price"]) ax.set_xlabel('House Prices in
millions',fontsize = 15)
ax.set_title('Univariate analysis of House Prices', fontsize=
20) plt.ticklabel_format(style='plain', axis='x')
```

Kết quả thực hiện case 2



Hình 1.7 - Biểu đồ violin

3. Phân tích dữ liệu đơn biến dựa vào bản tóm tắt dữ liệu

```
houseprices_data.describe()
```


Kết quả thực hiện

	Price
count	9.200000e+02
mean	6.220654e+05
std	5.389942e+05
min	1.750000e+05
25%	3.500000e+05
50%	4.670000e+05
75%	7.000000e+05
max	5.950000e+06

Hình 1.8 - Dữ liệu thống kê mô tả

Bài toán 2: Thực hiện các nhiệm vụ trong bài toán 2 để làm quen với việc phân tích hai biến với các hàm trong thư viện scikit-learn.

Nhiệm vụ 1: phân tích dữ liệu hai biến trên dữ liệu về chim cánh cụt. Dữ liệu lấy tại

https://www.kaggle.com/datasets/parulpandey/palmer-archipelago-antarctica-penguin-data

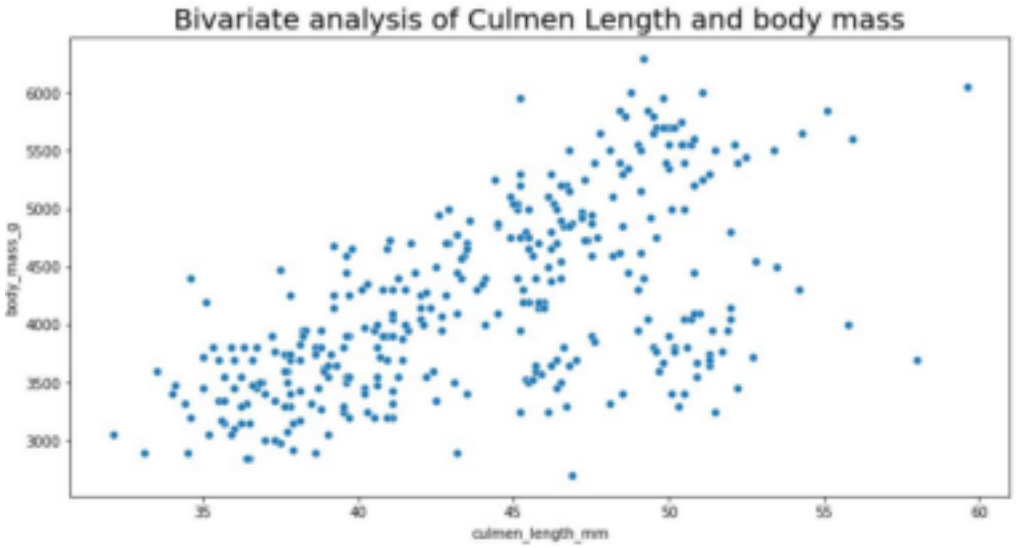
1. Import thư viện và chuẩn bị dữ liệu phân tích

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
penguins_data = pd.read_csv("data/penguins_size.csv")
penguins_data = penguins_data[['species','culmen_length_mm',
'body_mass_g']]
```

2. Phân tích dữ liệu 2 biến dựa vào phương pháp scatterplot

```
# case 1: basic
sns.scatterplot(data = penguins_data, x=
penguins_data["culmen_length_mm"],  y= penguins_data['body_mass_g']) #
case 2: advanced
plt.figure(figsize= (12,6))
ax = sns.scatterplot(data = penguins_data,
 x= penguins_data["culmen_length_mm"],
 y= penguins_data['body_mass_g'])
ax.set_title('Bivariate analysis of Culmen Length and body mass',
fontsize= 20)
```


Kết quả thực hiện case 2



Hình 1.9 - Biểu đồ scatter plot

3. Phân tích 2 biến dựa vào bảng crosstab/two-way

```
pd.crosstab(index= penguins_data['species'], columns=
penguins_data['sex'])
```

Kết quả thực hiện

sex	FEMALE	MALE
species		
Adelie	73	73
Chinstrap	34	34
Gentoo	58	62

4. Phân tích 2 biến sử dụng pivot_table

```
pd.pivot_table(penguins_data, values='culmen_length_mm',
 index='species', aggfunc=np.mean)
```

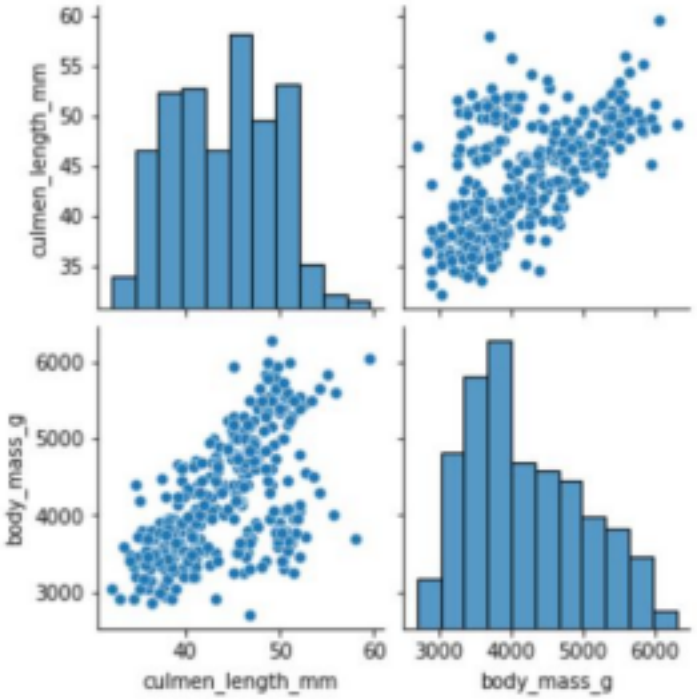
Kết quả thực hiện

	culmen_length_mm
species	
Adelie	38.791391
Chinstrap	48.833824
Gentoo	47.504878

5. Phân tích 2 biến sử dụng pairplot

```
import seaborn as sns
sns.pairplot( data = penguins_data)
```


Kết quả thực hiện



Hình 1.10 - Biểu đồ pair-plot

Bài toán 3: Thực hiện các nhiệm vụ trong bài toán 3 để làm quen với việc sử dụng các công cụ hỗ trợ EDA tự động.

Nhiệm vụ 1: Sử dụng pandas profiling trên dữ liệu Customer Personality Analysis. Dữ liệu lấy tại https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis

1. Cài đặt pandas_profiling sau này đổi tên thành ydata_profiling (xem thông tin chi tiết tại https://pypi.org/project/pandas-profiling/3.1.0)

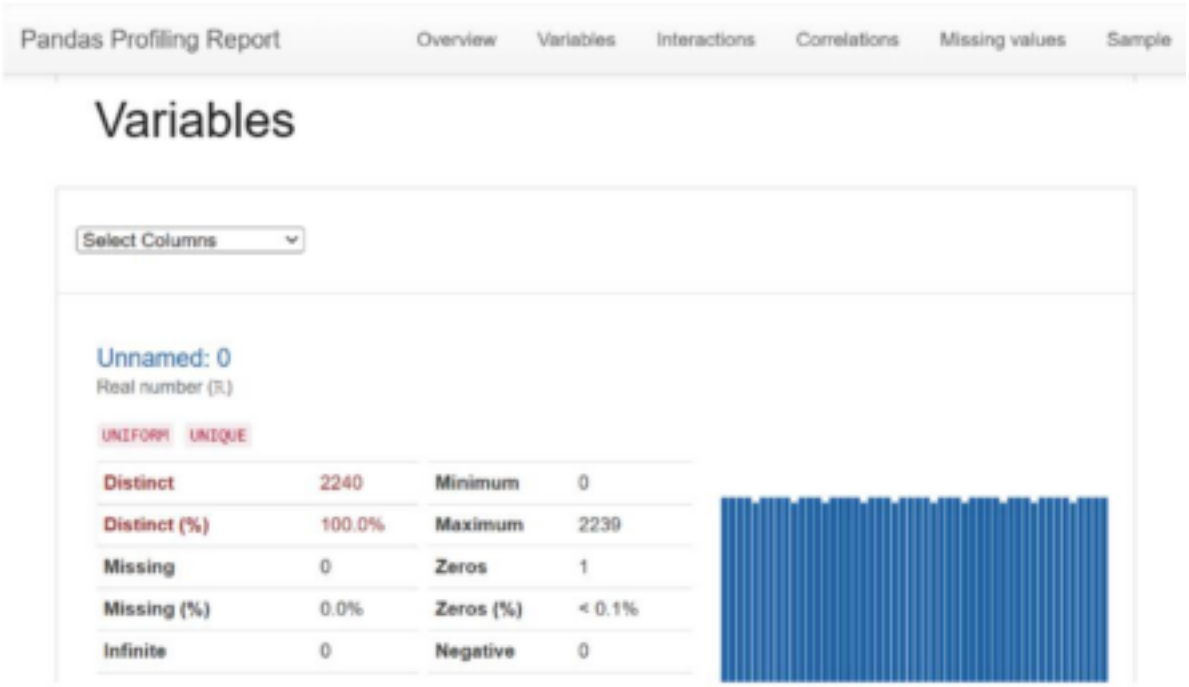
2. Sử dụng công cụ

```
import pandas as pd
from ydata_profiling import ProfileReport
marketing_data = pd.read_csv("data/marketing_campaign.csv")
# Create an Automated EDA report using the ProfileReport class in the ydata_profiling library. Use the to_file method to output the report to an HTML file
profile = ProfileReport(marketing_data)
profile.to_file("Reports/profile_output.html")
```

Kết quả thực hiện

3. Tiến hành EDA trên trang tập tin **profile_output.html**

Kết quả thực hiện



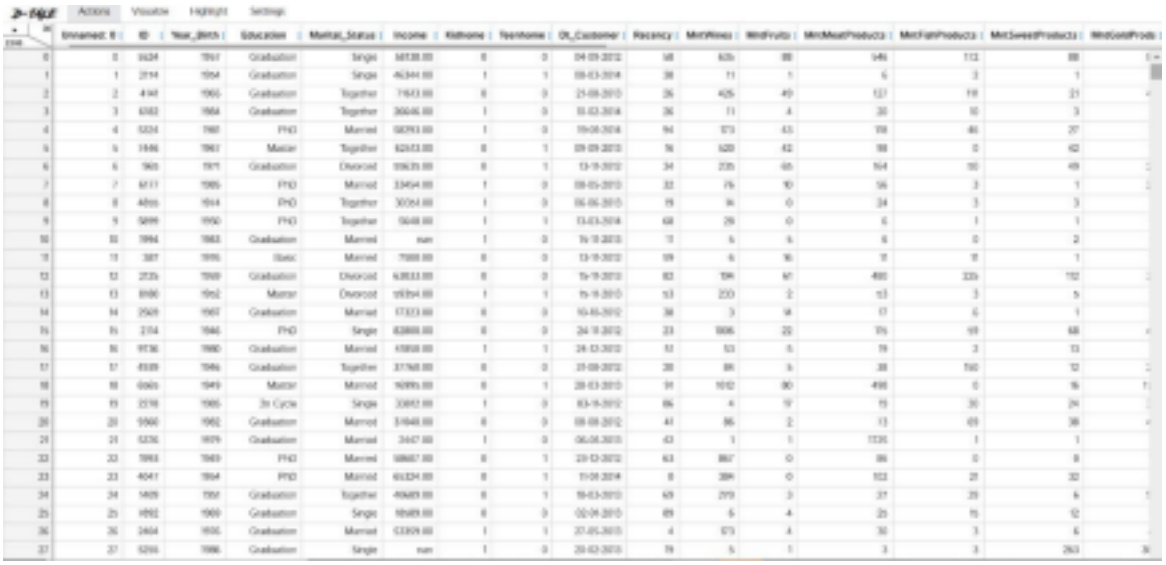
Hình 1.11 - EDA Report của công cụ ydata_profiling

Nhiệm vụ 2: Sử dụng dtale trên dữ liệu Marketing Campaign. Dữ liệu lấy từ https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis

1. Cài đặt dtale (xem thông tin chi tiết tại https://dtale.readthedocs.io/en/latest/)
2. Sử dụng công cụ

```
import pandas as pd
import dtale
marketing_data =
pd.read_csv("data/marketing_campaign.csv")
dtale.show(marketing_data).open_browser()
```


Kết quả thực hiện



Hình 1.12 - EDA Report của công cụ dtale

1.3.3. Bài tập thực hành 1

Tìm hiểu các tính năng và cách sử dụng sản phẩm SweetViz (https://pypi.org/project/sweetviz) áp dụng trên tập dữ liệu Marketing Campaign

1.3.4. Bài tập thực hành 2

Tìm hiểu các tính năng và cách sử dụng sản phẩm AutoViz (https://pypi.org/project/autoviz) áp dụng trên tập dữ liệu Marketing Campaign

D. TÓM TẮT THỰC HÀNH

Khám phá dữ liệu (Exploratory Data Analysis - EDA) là một bước quan trọng trong phân tích dữ liệu và khai thác dữ liệu, nhưng quá trình này không tránh khỏi những khó khăn. Một trong những thách thức lớn nhất là chất lượng dữ liệu không đảm bảo, bao gồm giá trị thiếu, giá trị ngoại lai hoặc dữ liệu không nhất quán, đòi hỏi kỹ năng tiền xử lý phức tạp và tốn thời gian. Bên cạnh đó, việc xử lý khối lượng dữ liệu lớn có thể gây khó khăn trong việc xác định các mẫu hoặc xu hướng có ý nghĩa, đặc biệt khi sử dụng các công cụ không được tối ưu hóa cho dữ liệu lớn. Chương đã trình bày một số kỹ thuật cơ bản khi sử dụng Python và các công cụ phát triển bằng Python giúp thực hiện việc khám phá dữ liệu được hiệu quả hơn.