

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені ТАРАСА ШЕВЧЕНКА

Факультет інформаційних технологій

Кафедра прикладних інформаційних систем



Звіт

до виконання лабораторної роботи №1

з дисципліни « **Data Science та Big Data** »

на тему:

« Агрегація, обробка пропусків та візуалізація даних пакетами
Python »

Виконано:

студ. групи ПП- 41, підгрупа 2

Шкандюк Анною Леонідівною

Перевірено:

к.т.н., доц. Білий Р.О.

Київ – 2023

1. Мета роботи:

Метою лабораторної роботи є отримання практичних навичок у роботі з raw data, використовуючи пакети jupyter, pandas, seaborn.

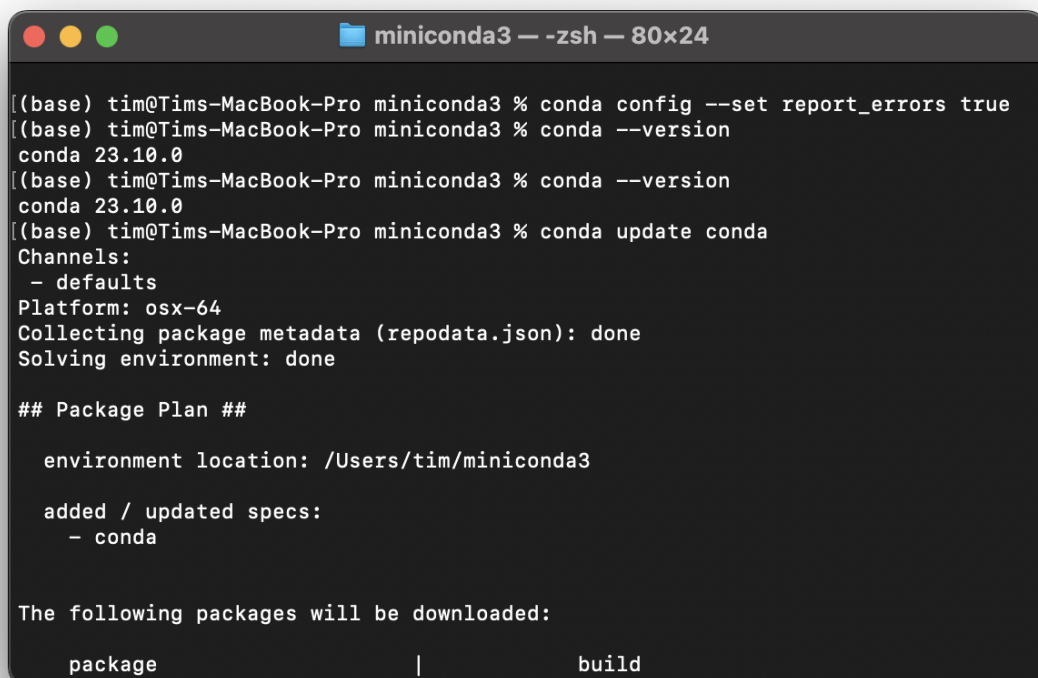
2. Завдання:

У дата сеті знаходяться 31 набір даних з іменами nyt1.csv, nyt2.csv, ..., nyt31.csv. Кожен із них демонструє один (симульований) день показів оголошень та переходів по них, записаних на головній сторінці газети The New York Times у травні 2012 року. Кожен рядок представляє одного користувача. Існує п'ять стовпців: вік, стать (0 = жінка, 1 = чоловік), кількість показів, кількість переходів та статус авторизації.

На базі цього виконати наступні завдання:

3. Хід виконання:

Підготовка необхідного обладнання та середовища для виконання:



```
miniconda3 — zsh — 80x24

[(base) tim@Tims-MacBook-Pro miniconda3 % conda config --set report_errors true ]
[(base) tim@Tims-MacBook-Pro miniconda3 % conda --version ]
conda 23.10.0
[(base) tim@Tims-MacBook-Pro miniconda3 % conda --version ]
conda 23.10.0
[(base) tim@Tims-MacBook-Pro miniconda3 % conda update conda ]
Channels:
- defaults
Platform: osx-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

environment location: /Users/tim/miniconda3

added / updated specs:
- conda

The following packages will be downloaded:

package | build
```

```

miniconda3 — -zsh — 80x24

## Package Plan ##

environment location: /Users/tim/miniconda3

added / updated specs:
- conda

The following packages will be downloaded:

package | build
-----|-----
certifi-2023.11.17 | py311hecd8cb5_0 160 KB
-----|-----
Total: 160 KB

The following packages will be UPDATED:

certifi 2023.7.22-py311hecd8cb5_0 --> 2023.11.17-py311
hecd8cb5_0

Proceed ([y]/n)? y

```

```

miniconda3 — -zsh — 80x24

((base) tim@Tims-MacBook-Pro miniconda3 % conda create --name bigdata jupyter pan
das seaborn
Channels:
- defaults
Platform: osx-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

environment location: /Users/tim/miniconda3/envs/bigdata

added / updated specs:
- jupyter
- pandas
- seaborn

The following packages will be downloaded:

package | build
-----|-----
anyio-3.5.0 | py311hecd8cb5_0 212 KB
appnope-0.1.2 | py311hecd8cb5_1001 11 KB
-----|-----

```

```

miniconda3 — -zsh — 80x24

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate bigdata
#
# To deactivate an active environment, use
#
#     $ conda deactivate

((base) tim@Tims-MacBook-Pro miniconda3 % conda activate bigdata
(bigdata) tim@Tims-MacBook-Pro miniconda3 %

```

1. Завантажити файли з даними у папку проекту з посилання:

https://github.com/oreillymedia/doing_data_science



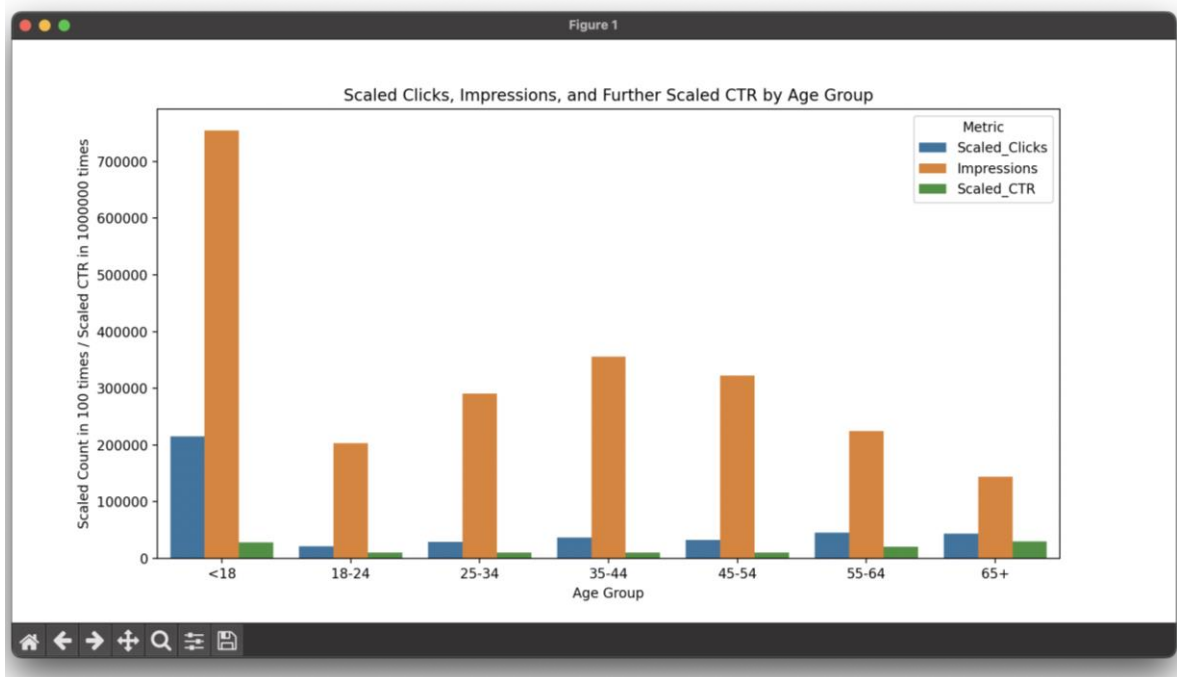
The screenshot shows the JupyterLab interface. At the top, there's a 'Jupyter' logo and 'Quit'/'Logout' buttons. Below is a tab bar with 'Files', 'Running', and 'Clusters'. A message says 'Select items to perform actions on them.' with 'Upload' and 'New' buttons. The main area shows a file browser for 'Untitled Folder 1'. It contains a table of files:

	Name	Last Modified	File size
	..	декілька секунд тому	
<input type="checkbox"/>	lab1.py	40 хвилин тому	4.17 kB
<input type="checkbox"/>	nyt1.csv	35 хвилин тому	4.92 MB
<input type="checkbox"/>	nyt10.csv	35 хвилин тому	4.86 MB
<input type="checkbox"/>	nyt11.csv	35 хвилин тому	5.13 MB
<input type="checkbox"/>	nyt12.csv	35 хвилин тому	4.25 MB
<input type="checkbox"/>	nyt13.csv	35 хвилин тому	8.43 MB
<input type="checkbox"/>	nyt14.csv	35 хвилин тому	4.73 MB
<input type="checkbox"/>	nyt15.csv	35 хвилин тому	4.63 MB
<input type="checkbox"/>	nyt16.csv	35 хвилин тому	4.75 MB
<input type="checkbox"/>	nyt17.csv	35 хвилин тому	4.71 MB
<input type="checkbox"/>	nyt18.csv	35 хвилин тому	4.78 MB
<input type="checkbox"/>	nyt19.csv	35 хвилин тому	4.47 MB
<input type="checkbox"/>	nyt2.csv	35 хвилин тому	4.83 MB
<input type="checkbox"/>	nyt20.csv	35 хвилин тому	7.70 MB

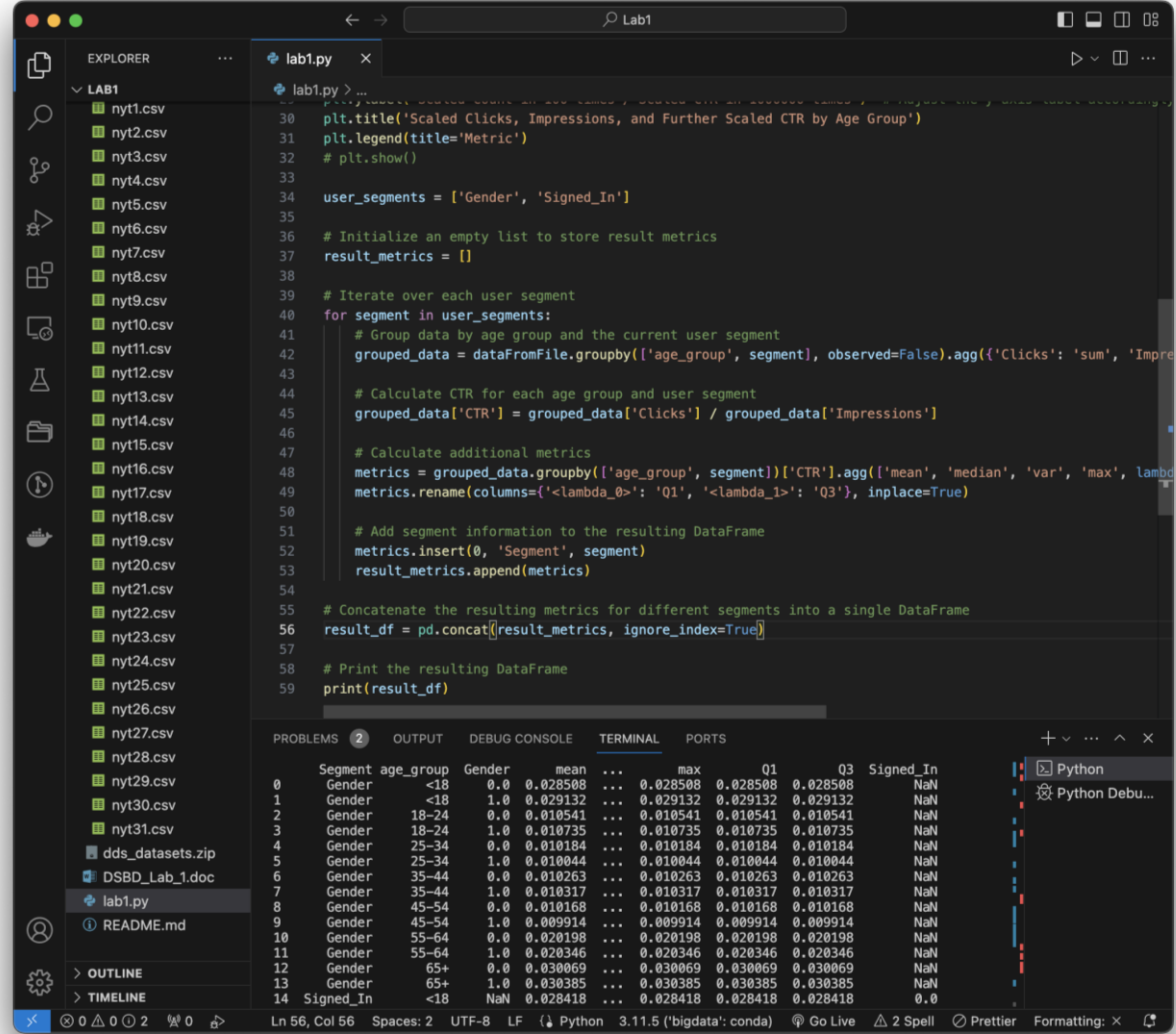
2. Створіть нову змінну **age_group**, яка агрегує користувачів як <18, 18–24, 25–34, 35–44, 45–54, 55–64 та 65+.
3. Зафіксуйте на діаграмі кількість показів та показник переходів ($CTR = \frac{\#clicks}{\#impressions}$) для цих шести вікових категорій.

На діаграмі синім кольором це: кліки (Scaled_Clicks), помаранчевим - кількість показів (Impressions), а зеленим - показник переходів (Scaled CTR). Для того, щоб все ці значення могли розміститися на одному графіку я збільшила Scaled_Clicks у 100 разів, а Scaled_CTR - у $1000000=10^6$ разів. На графіку зображенні данні за один день.

```
lab1.py
1 import pandas as pd
2 import seaborn as sns
3 import matplotlib.pyplot as plt
4
5 dataFromFile = pd.read_csv("maybeDataForLab1/nyt1.csv")
6
7 age_bins = [0, 18, 25, 35, 45, 55, 65, float('inf')]
8 age_labels = ['<18', '18-24', '25-34', '35-44', '45-54', '55-64', '65+']
9
10 dataFromFile['age_group'] = pd.cut(dataFromFile['Age'], bins=age_bins, labels=age_labels, right=False)
11
12 agg_data = dataFromFile.groupby('age_group', observed=False)[['Clicks', 'Impressions']].sum().reset_index()
13
14 agg_data['CTR'] = agg_data['Clicks'] / agg_data['Impressions']
15
16 # Scale 'Clicks' and CTR for better visibility
17 agg_data['Scaled_Clicks'] = agg_data['Clicks'] * 10 # Adjust the scaling factor as needed
18 agg_data['Scaled_CTR'] = agg_data['CTR'] * 1000000 # Adjust the scaling factor as needed
19
20 # Melt the DataFrame to combine 'Scaled_Clicks', 'Impressions', and 'Scaled_CTR' into a single column
21 melted_data = pd.melt(agg_data, id_vars='age_group', value_vars=['Scaled_Clicks', 'Impressions', 'Scaled_CTR'])
22
23 # Create a bar plot with 'Metric' as hue
24 plt.figure(figsize=(12, 6))
25 sns.barplot(x='age_group', y='value', hue='Metric', data=melted_data)
26
27 # Add labels and legend
28 plt.xlabel('Age Group')
29 plt.ylabel('Scaled Count in 100 times / Scaled CTR in 1000000 times') # Adjust the y-axis label accordingly
30 plt.title('Scaled Clicks, Impressions, and Further Scaled CTR by Age Group')
31 plt.legend(title='Metric')
32 plt.show()
```



4. Вивчіть дані та проведіть візуальні та кількісні порівняння між сегментами користувачів/демографічними групами (наприклад, чоловіки старше 18 років у порівнянні з жінками старше 18 років або авторизовані та неавторизовані користувачі).



```
30 plt.title('Scaled Clicks, Impressions, and Further Scaled CTR by Age Group')
31 plt.legend(title='Metric')
32 # plt.show()
33
34 user_segments = ['Gender', 'Signed_In']
35
36 # Initialize an empty list to store result metrics
37 result_metrics = []
38
39 # Iterate over each user segment
40 for segment in user_segments:
41     # Group data by age group and the current user segment
42     grouped_data = dataFromFile.groupby(['age_group', segment], observed=False).agg({'Clicks': 'sum', 'Impressions': 'sum'})
43
44     # Calculate CTR for each age group and user segment
45     grouped_data['CTR'] = grouped_data['Clicks'] / grouped_data['Impressions']
46
47     # Calculate additional metrics
48     metrics = grouped_data.groupby(['age_group', segment])['CTR'].agg(['mean', 'median', 'var', 'max', 'min'])
49     metrics.rename(columns={'<lambda_0>': 'Q1', '<lambda_1>': 'Q3', inplace=True)
50
51     # Add segment information to the resulting DataFrame
52     metrics.insert(0, 'Segment', segment)
53     result_metrics.append(metrics)
54
55 # Concatenate the resulting metrics for different segments into a single DataFrame
56 result_df = pd.concat(result_metrics, ignore_index=True)
57
58 # Print the resulting DataFrame
59 print(result_df)
```

	Segment	age_group	Gender	mean	max	Q1	Q3	Signed_In
0	Gender	<18	0.0	0.028508	0.028508	0.028508	0.028508	NaN
1	Gender	<18	1.0	0.029132	0.029132	0.029132	0.029132	NaN
2	Gender	18-24	0.0	0.010541	0.010541	0.010541	0.010541	NaN
3	Gender	18-24	1.0	0.010735	0.010735	0.010735	0.010735	NaN
4	Gender	25-34	0.0	0.010184	0.010184	0.010184	0.010184	NaN
5	Gender	25-34	1.0	0.010044	0.010044	0.010044	0.010044	NaN
6	Gender	35-44	0.0	0.010263	0.010263	0.010263	0.010263	NaN
7	Gender	35-44	1.0	0.010317	0.010317	0.010317	0.010317	NaN
8	Gender	45-54	0.0	0.010168	0.010168	0.010168	0.010168	NaN
9	Gender	45-54	1.0	0.009914	0.009914	0.009914	0.009914	NaN
10	Gender	55-64	0.0	0.020198	0.020198	0.020198	0.020198	NaN
11	Gender	55-64	1.0	0.020346	0.020346	0.020346	0.020346	NaN
12	Gender	65+	0.0	0.030069	0.030069	0.030069	0.030069	NaN
13	Gender	65+	1.0	0.030385	0.030385	0.030385	0.030385	NaN
14	Signed_In	<18	NaN	0.028418	0.028418	0.028418	0.028418	0.0

5. Створіть метрики/вимірювання/статистику, які підсумовують дані. Приклади можливих метрик включають CTR, квантил, середнє значення, медіану, дисперсію та максимальне значення. Ці показники потрібно розрахувати за різними сегментами користувачів. Подумайте про елементи, які важливо відстежувати з часом - що стискає дані, але, як і раніше, захоплює поведінку користувача.

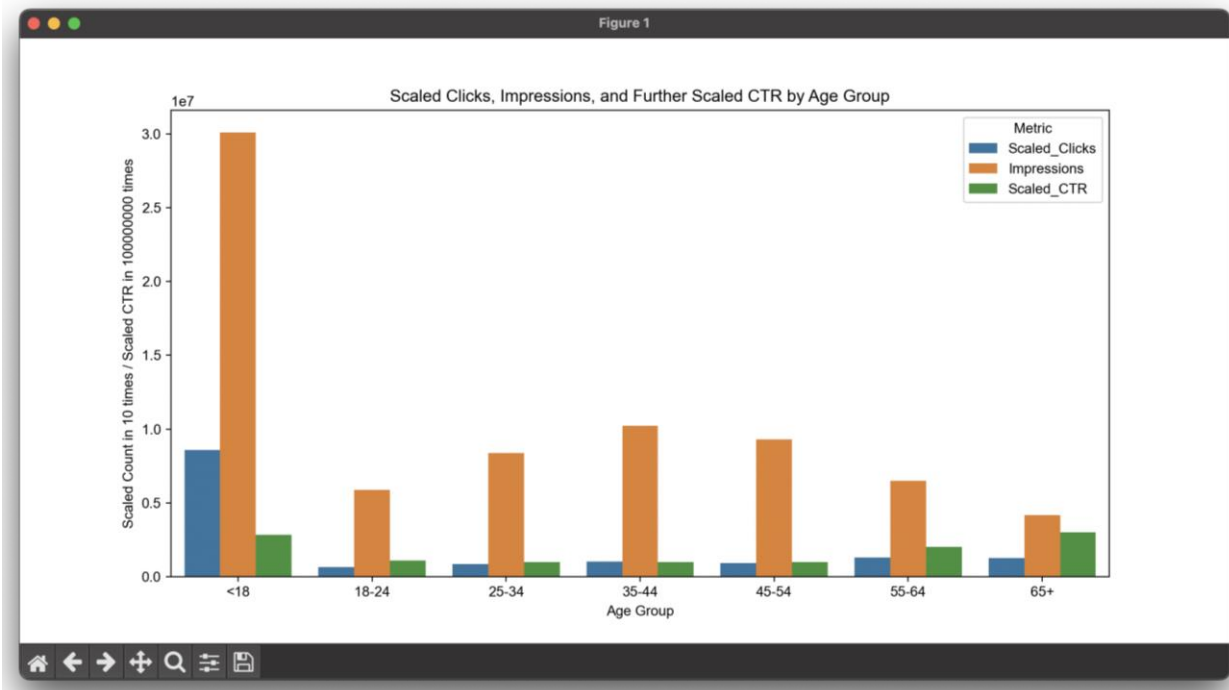
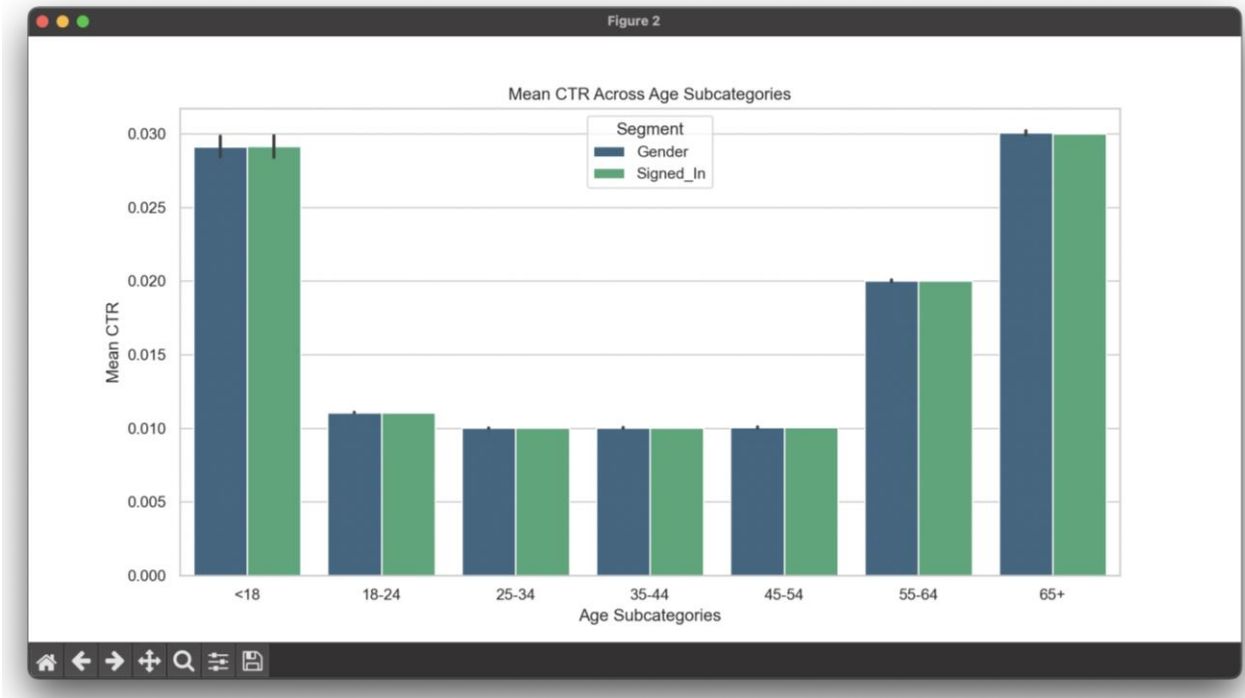

```
lab1.py
30 plt.title('Scaled Clicks, Impressions, and Further Scaled CTR by Age Group')
31 plt.legend(title='Metric')
32 # plt.show()
33
34 user_segments = ['Gender', 'Signed_In']
35
36 # Initialize an empty list to store result metrics
37 result_metrics = []
38
39 # Iterate over each user segment
40 for segment in user_segments:
41     # Group data by age group and the current user segment
42
43     metrics = grouped_data.groupby(['age_group', segment])['CTR'].agg(['mean', 'median', 'var', '
44 max', lambda x: x.quantile(0.25), lambda x: x.quantile(0.75)]).reset_index()
45 /Users/tim/Desktop/projects/Anns projects/4courseUniv/bigData/Lab1/lab1.py:48: FutureWarning: T
46 he default of observed=False is deprecated and will be changed to True in a future version of p
47 andas. Pass observed=False to retain current behavior or observed=True to adopt the future defa
48 ult and silence this warning.
49
50 metrics = grouped_data.groupby(['age_group', segment])['CTR'].agg(['mean', 'median', 'var', '
51 max', lambda x: x.quantile(0.25), lambda x: x.quantile(0.75)]).reset_index()
52
53 Segment age_group Gender mean max Q1 Q3 Signed_In
54 0 Gender <18 0.0 0.028508 ... 0.028508 0.028508 0.028508 NaN
55 1 Gender <18 1.0 0.029132 ... 0.029132 0.029132 0.029132 NaN
56 2 Gender 18-24 0.0 0.010541 ... 0.010541 0.010541 0.010541 NaN
57 3 Gender 18-24 1.0 0.010735 ... 0.010735 0.010735 0.010735 NaN
58 4 Gender 25-34 0.0 0.010184 ... 0.010184 0.010184 0.010184 NaN
59 5 Gender 25-34 1.0 0.010044 ... 0.010044 0.010044 0.010044 NaN
60 6 Gender 35-44 0.0 0.010263 ... 0.010263 0.010263 0.010263 NaN
61 7 Gender 35-44 1.0 0.010317 ... 0.010317 0.010317 0.010317 NaN
62 8 Gender 45-54 0.0 0.010168 ... 0.010168 0.010168 0.010168 NaN
63 9 Gender 45-54 1.0 0.009914 ... 0.009914 0.009914 0.009914 NaN
64 10 Gender 55-64 0.0 0.020198 ... 0.020198 0.020198 0.020198 NaN
65 11 Gender 55-64 1.0 0.020346 ... 0.020346 0.020346 0.020346 NaN
66 12 Gender 65+ 0.0 0.030069 ... 0.030069 0.030069 0.030069 NaN
67 13 Gender 65+ 1.0 0.030385 ... 0.030385 0.030385 0.030385 NaN
68 14 Signed_In <18 NaN 0.028418 ... 0.028418 0.028418 0.028418 0.0
69 15 Signed_In <18 NaN 0.029824 ... 0.029824 0.029824 0.029824 1.0
70 16 Signed_In 18-24 NaN NaN ... NaN NaN NaN NaN 0.0
71 17 Signed_In 18-24 NaN 0.010644 ... 0.010644 0.010644 0.010644 1.0
72 18 Signed_In 25-34 NaN NaN ... NaN NaN NaN NaN 0.0
73 19 Signed_In 25-34 NaN 0.010110 ... 0.010110 0.010110 0.010110 1.0
74 20 Signed_In 35-44 NaN NaN ... NaN NaN NaN NaN 0.0
75 21 Signed_In 35-44 NaN 0.010292 ... 0.010292 0.010292 0.010292 1.0
76 22 Signed_In 45-54 NaN NaN ... NaN NaN NaN NaN 0.0
77 23 Signed_In 45-54 NaN 0.010034 ... 0.010034 0.010034 0.010034 1.0
78 24 Signed_In 55-64 NaN NaN ... NaN NaN NaN NaN 0.0
79 25 Signed_In 55-64 NaN 0.020277 ... 0.020277 0.020277 0.020277 1.0
80 26 Signed_In 65+ NaN NaN ... NaN NaN NaN NaN 0.0
81 27 Signed_In 65+ NaN 0.030183 ... 0.030183 0.030183 0.030183 1.0
```

6. Результати статистичного дослідження подати у вигляді результуючого ДатаФрейма (одного), дивлячись на який можна зрозуміти і порівнювати дані за віковими підкатегоріями.

На цих графіках зображенні дані вже за усі дні.

На першому графіку зображено середні показники переходів (CTR) для вікових сегментів користувачів.

На другому графіку синім кольором позначена кількість кліків (Scaled_Clicks), помаранчевим - кількість показів (Impressions), а зеленим - показник переходів (Scaled CTR). Для того, щоб все ці значення могли розміститися на одному графіку я збільшила Scaled_Clicks у 10 разів, а Scaled_CTR - у $100000000=10^8$ разів.



7. Опишіть та інтерпретуйте будь-які закономірності, які знайдете.

Аналіз результатів вище:

Варіація CTR серед вікових груп та статей:

Значення CTR варіюються в різних вікових групах та статей(ч/ж). Наприклад, у віковій групі '<18' як чоловіки, так і жінки мають відносно високі значення CTR, причому у жінок середнє значення CTR трошки

вище. У вікових групах '55-64' та '65+' чоловіки мають тенденцію до вищих значень CTR порівняно з жінками.

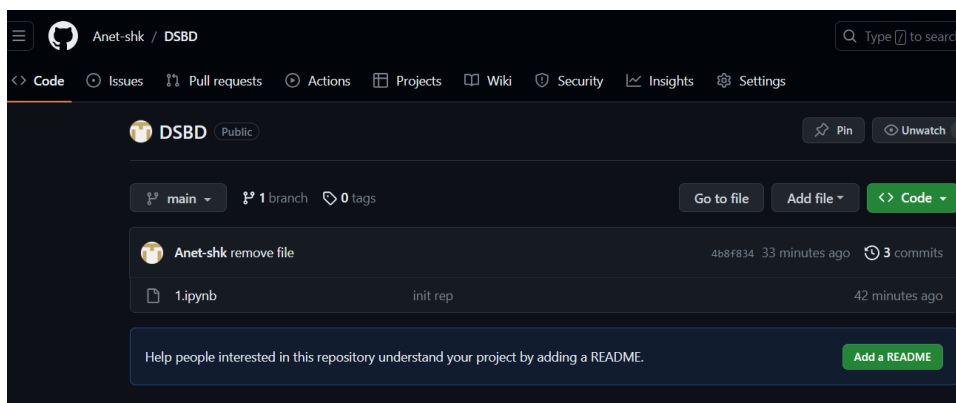
Варіація CTR за віковими групами та статусом входу:

Значення CTR також змінюються в залежності від статусу входу. У віковій групі '<18' користувачі, які увійшли в систему, мають тенденцію мати вищий CTR порівняно з тими, хто не увійшов в систему. Схожі тенденції спостерігаються у вікових групах '18-24' та '35-44'.

Загальні спостереження:

CTR має тенденцію варіюватися більше за статтю (ч/ж), ніж за статусом входу. Є випадки, коли наявність чи відсутність інформації про вхід суттєво впливає на CTR, особливо в деяких вікових групах.

8. Завантажити файл ірпnb з виконаними завданнями на git в окрему папку з відповідною назвою лабораторної роботи.



4. Висновки до роботи

Під час виконання лабораторної роботи 1, я отримала практичні навички у роботі з raw data, використовуючи пакети jupyter, pandas, seaborn.