2.2.2_Pandas_Profiling

June 13, 2025

Creado por:

Isabel Maniega

1 Pandas-Profiling

1.1 ydata-profiling

[1]:
```
# pip install ydata-profiling
```

[2]:
```
from ydata_profiling import ProfileReport
import pandas as pd
```

```
<IPython.core.display.HTML object>
```

[3]:
```
df = pd.read_csv("./files/train.csv")
df.head()
```

[3]:

	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

[4]:
```
profile = ProfileReport(df, title='Titanic')
profile.to_notebook_iframe()
```

```
Summarize dataset:   0%|          | 0/5 [00:00<?, ?it/s]


100%|                      | 12/12 [00:00<00:00,
31203.75it/s]

Generate report structure:   0%|          | 0/1 [00:00<?, ?it/s]

Render HTML:   0%|          | 0/1 [00:00<?, ?it/s]

<IPython.core.display.HTML object>
```

[5]:
```
profile.to_file('profile_titanic.html')
```

```
Export report to file:   0%|          | 0/1 [00:00<?, ?it/s]
```

1.2 sweetviz

[6]:
```
# pip install sweetviz
```

[7]:
```
import sweetviz as sv
import pandas as pd
```

[8]:
```
df = pd.read_csv('./files/train.csv')
report = sv.analyze(df)
report.show_html('Sweetviz_Report.html')
```

```
                                             |      | [  0%]   00:00 -> (? left)

Report Sweetviz_Report.html was generated! NOTEBOOK/COLAB USERS: the web browser
MAY not pop up, regardless, the report IS saved in your notebook/colab files.
```

En la ejecucion anterior falla en un warning, para subsanarlo, vamos a la carpeta: * env/lib/python3.12/site-packages/sweetviz/graph_numeric.py comentamos las lineas: 71, 74

[9]:
```
my_report = sv.compare_intra(df, df["Sex"] == "male", ["Male", "Female"],␣
↪"Survived")
my_report.show_html('Sweetviz_My_Report.html')
```

```
                                             |      | [  0%]   00:00 -> (? left)

Report Sweetviz_My_Report.html was generated! NOTEBOOK/COLAB USERS: the web
browser MAY not pop up, regardless, the report IS saved in your notebook/colab
files.
```

1.3 Itables

[10]:
```
# pip install itables
```

[11]:
```
import itables

itables.init_notebook_mode()
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

[12]:
```
df = itables.sample_dfs.get_countries(html=False)
df
```

[12]:

code	region	country	capital	longitude	latitude
AW	Latin America & Caribbean	Aruba	Oranjestad	-70.0167	12.51670
AF	South Asia	Afghanistan	Kabul	69.1761	34.52280
AO	Sub-Saharan Africa	Angola	Luanda	13.2420	-8.81155
AL	Europe & Central Asia	Albania	Tirane	19.8172	41.33170
AD	Europe & Central Asia	Andorra	Andorra la Vella	1.5218	42.50750
…	…	…	…	…	…
XK	Europe & Central Asia	Kosovo	Pristina	20.9260	42.56500
YE	Middle East & North Africa	Yemen, Rep.	Sana'a	44.2075	15.35200
ZA	Sub-Saharan Africa	South Africa	Pretoria	28.1871	-25.74600
ZM	Sub-Saharan Africa	Zambia	Lusaka	28.2937	-15.39820
ZW	Sub-Saharan Africa	Zimbabwe	Harare	31.0672	-17.83120

[208 rows x 5 columns]

[13]:
```
# agregar botones de formatos de exportacion:
itables.show(df, buttons=["pageLength", "copyHtml5", "csvHtml5", "excelHtml5"])
```

```
<IPython.core.display.HTML object>
```

[14]:
```
# agregar paneles de filtrado:

df = itables.sample_dfs.get_countries(html=False, climate_zone=True)
itables.show(
    df.reset_index(),
    layout={"top1": "searchPanes"},
    searchPanes={"layout": "columns-3", "cascadePanes": True, "columns": [1, 6,␣
 ↪7]},
)
```

```
<IPython.core.display.HTML object>
```

[15]:
```
itables.show(
    df,
    layout={"top1": "searchBuilder"},
    searchBuilder={
        "preDefined": {
            "criteria": [
                {"data": "climate_zone", "condition": "=", "value":␣
 ↪["Sub-tropical"]}
            ]
        }
    },
)
```

```
<IPython.core.display.HTML object>
```

[16]:
```
# añadir el scroll en la ventana:

itables.show(df, scrollY="350px", scrollCollapse=True, paging=False)
```

```
<IPython.core.display.HTML object>
```

Creado por:

Isabel Maniega