Data Exploration

{codenation}®

Learning Objectives

- To generate exploratory reports

Reporting

Reporting

We explored our data using Pandas' methods like **.describe()**

This helped us work out the data types we were working with, if there were any missing values, any outliers, and gave us our descriptive statistics.

We can use these to get a good overall picture.

Reporting

As a result of your exploring, you might need to change your workflow.

You might need to request more data, more time, or more resources.

You might need to explain why!

Reporting

A **report** is a more readable and accessible way of presenting your findings.

Most people aren't used to reading figures in a dark terminal!

Reporting

Exploring your data and then writing the report can be time consuming.

There are libraries we can use to streamline the process!

As long as we have the data in a dataframe, the ydata-profiling library can report on it for us!

ydata-profiling

ydata-profiling

ydata-profiling is a tool that creates a thorough, visually accessible summary of a data set with a few lines of code.

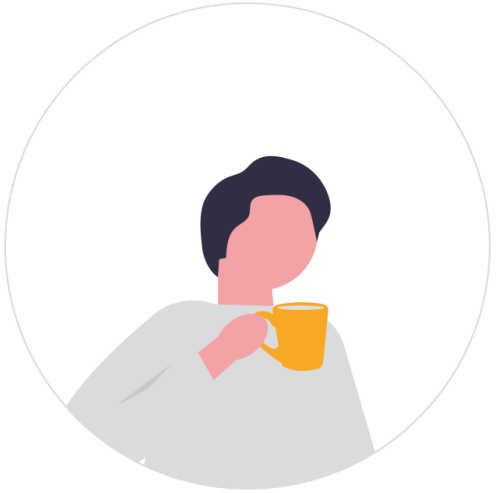
Using a library means your data reports will be consistent, and high quality.

ydata-profiling

```
1 pip install ydata-profiling
```

ydata-profiling is a big library!

It could take a while to install.



ydata-profiling

We need to take three steps with ydata:

1. Import the data as a frame
2. Turn the frame to profile
3. Turn the profile into a web page

To turn our terminal stats into this...

Variables

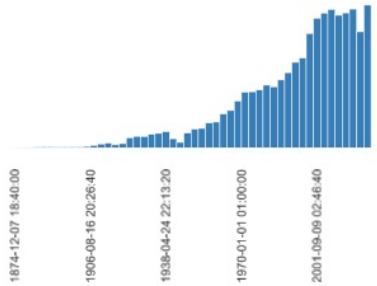
Select Columns

date

Date

Distinct	15649
Distinct (%)	34.5%
Missing	0
Missing (%)	0.0%
Memory size	354.2 KiB

Minimum	1872-11-30 00:00:00
Maximum	2023-11-21 00:00:00



More details

home_team

Text

Distinct	313
Distinct (%)	0.7%
Missing	0
Missing (%)	0.0%
Memory size	354.2 KiB

ydata-profiling

Create a new file called **reporting.py**

Import the necessary libraries and classes.

```
1 import pandas as pd
2 from ydata_profiling import ProfileReport
```

ydata-profiling

Create a dataframe from the results.csv file.

```
1 df = pd.read_csv("results.csv")
```

ydata-profiling

```
1 profile = ProfileReport(df, title = "International Results")
```

Create a new ProfileReport object.

It needs to know what it is building the report from and the title of the report.

ydata-profiling

```
1 profile.to_file('results_report.html')
```

Export your ProfileReport object to an html file!
It can take a while...

Navigating the Report

Navigating the report – overview

The overview section gives us information similar to `df.info()`

It tells us the shape of the frame, how many missing values there are, and how many duplicates.

It warns us lots of our values are 0 – but these figures are acceptable in the context!

Navigating the report – variables

The variables section gives us insight to the columns.

It shows unique values, missing data, and descriptive statistics like the minimum and maximum.

It also turns the data into a visualisation. Histograms, bar charts, and word clouds can put the stats into an accessible format.

Navigating the report – interactions

The interactions section shows us the two score variables and how they might interact to impact a third variable.

It suggests, out of all the results this data set shows, a final score of 1-1 is most likely.

Navigating the report – correlation

The correlation section shows how the strength of the relationship between the variables.

It measures how much the difference between the values can be explained by each other.

There's no relationship between results – a team scoring a goal doesn't suggest the opponent will score a goal.

It also doesn't suggest if a team scores a goal, the other team will score -1 goal.

Navigating the report – missing values

Missing values would show how missing values were distributed across the data set.

It would highlight quickly if certain columns had fewer responses.

This can be useful in surveys to see which questions participants tend to ignore.

Navigating the report – sample

The sample section acts similarly to the .head() and .tail() methods, providing some of the first values in the set, and the same amount at the end of the set.

Review

Review

Which method did you prefer?

Why?

Which method do you think would be more useful to a non-analyst?

Learning Objectives

- To generate exploratory reports

Activity 1

Produce exploratory reports for the other two data sets given.

What information can you see from them?

We will explore visualisations and what they mean more next week!