Analytics Vidhya

Exploratory data analysis (EDA) with ydata-profiling

Instructor

Fabiana Clemente

Chief Data Officer at YData

A bit about myself…

Fabiana Clemente, Chief Data Officer at YData

Professional experience

Applied Maths & Data Science

From big enterprises to startups

Data Science & Architecture

Co-Founder @YData

Interests

Data Science

Time-Series

Generative Models

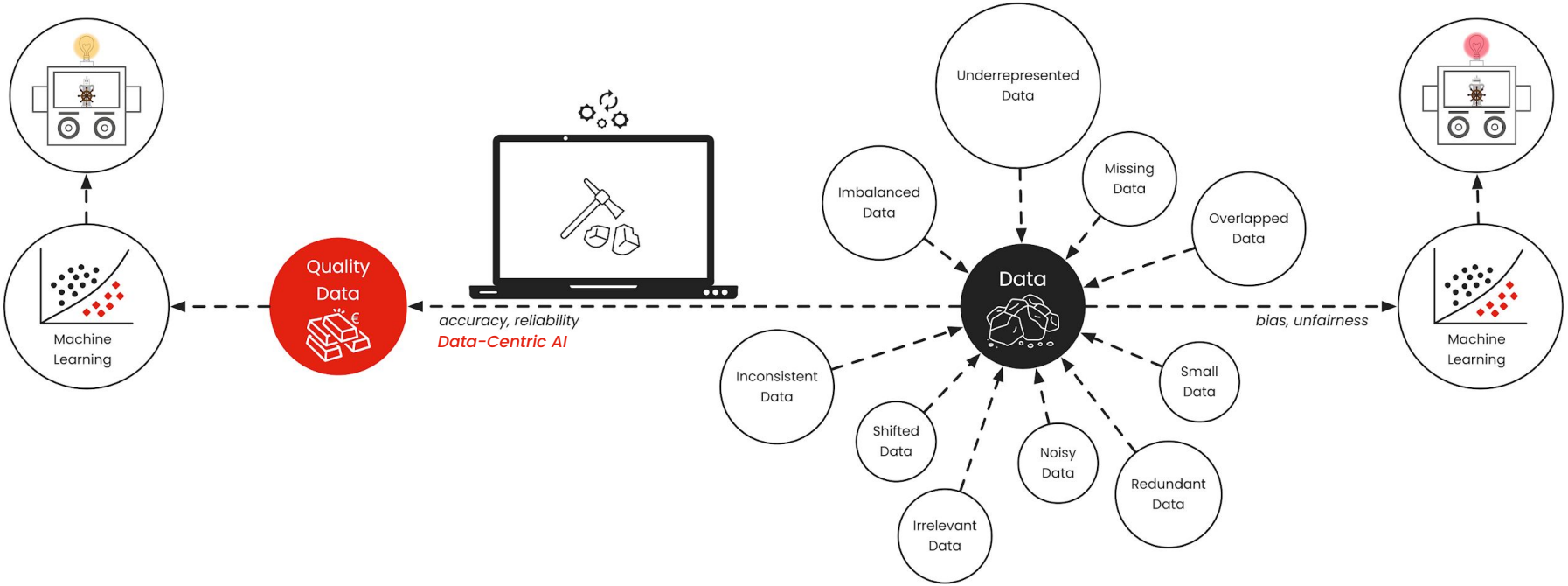
Linked in

Analytics Vidhya

What you can expect to learn today

- Data Quality & Data-Centric AI
- Common Data Quality Issues
- Intro to ydata-profiling
- Hands-on tutorial:
  - Create a virtual environment with conda
  - Explore the HCC dataset

Data Quality & Data-Centric AI



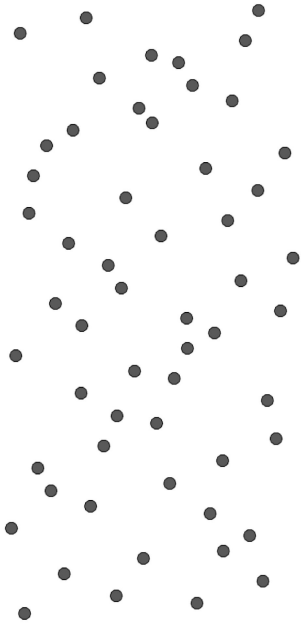
Real world data is complex

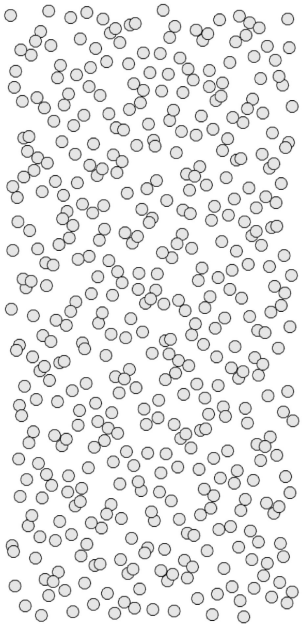
- Large amounts of data (?) sometimes produced within milliseconds
- Handled by different people within organizations
- Collected from multiple sources (heterogeneous data, unstructured data)
- Recorded at different frequencies and with different formats: audio, video, image, text, sensor
- Stored in decentralised databases
- **Data Quality Issues: True *Imperfections* versus *Data Intrinsic Characteristics*?**
- Data *Acquisition, Transmission, and Storage* versus *Nature of Domains*!
- **Special Concerns:** Interpretability/Explainability, high-stakes domains (mistakes cost lives), privacy concerns and data availability, fairness and ethics concerns!
- **Great opportunity for Data-Centric AI to prove truly transformative!**

Common data quality issues

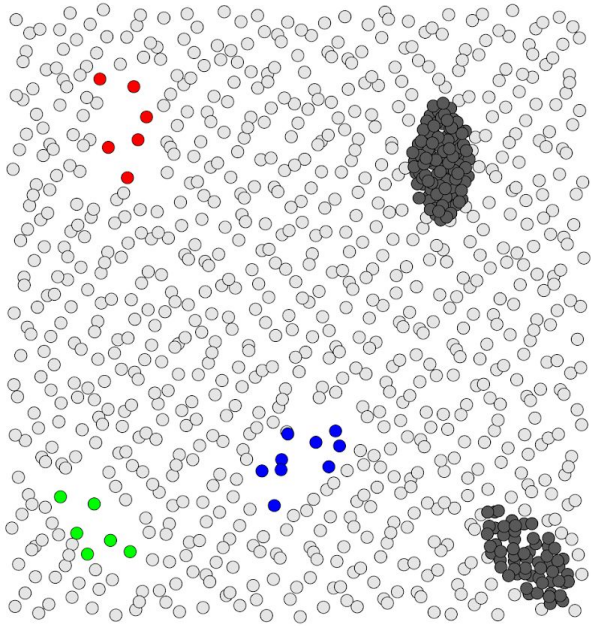
Understanding your data is crucial to a performant and fair AI development

Imbalanced Data

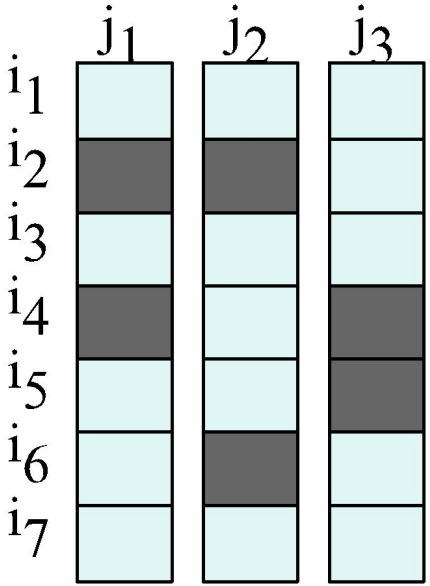




Underrepresented Data



Missing Data



Several others worth exploring: Dataset Shift, Noisy Data, Lack of Data, Irrelevant or Redundant Data, Inconsistent Data, Class Overlap

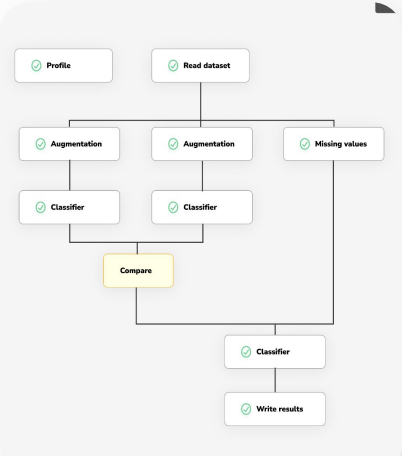
Introducing ydata-profiling

Complete exploratory data analysis in a single line of code

`ydataai/ydata-profiling` **12K stars**

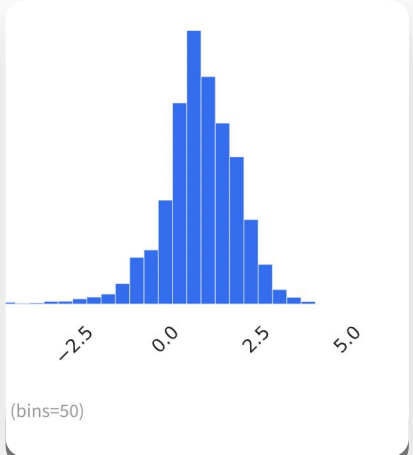
Alerts & warnings

Quickly access a summarized understanding of the challenges and quality issues of your dataset.



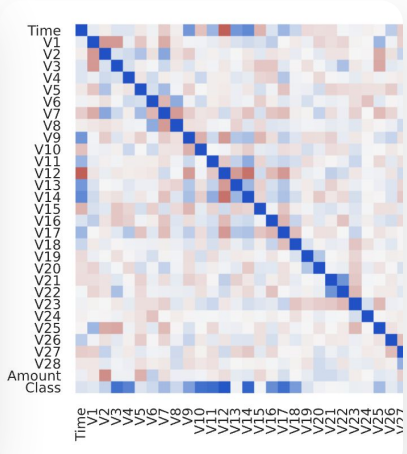
Univariate analysis

Through statistics and algebra, get all the information at once.



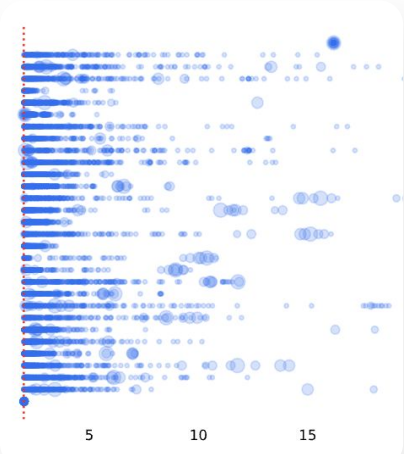
Multivariate analysis

Multivariate profiling in an unsupervised manner for interactions validation and optimized correlations.



Missing data

Missing data analysis details along with outlier detection for further analysis.



Hands-on tutorial

Exploring the HCC dataset

Kaggle link: kaggle.com/datasets/mrsantos/hcc-dataset
Github: github.com/Data-Centric-AI-Community/awesome-python-for-data-science/blob/main/tutorials/workshop-eda-datahour

The HCC Dataset

Gender	Age	Alcohol	Hallmark	PS	Encephalopathy	Hemoglobin	HBeAg	MCV	Total_Bil	O2	Dir_Bil	Ferritin	Outcome
Male	67	Yes	AYes	Active	None	13.7	No	106.6	2.1	999	0.5		Alive
Female	62	No	BYes	Active	None		No	103.4		999			Alive
Male	78	Yes	CYes	Ambulatory	None	8.9	No	79.8	0.4	999	0.1	16	Alive
Male	77	Yes	DYes	Active	None	13.4	No	97.1	0.4	999	0.2		Dead
Male	76	Yes	EYes	Active	None	14.3	No	95.1	0.7	999		22	Alive
Male	75	Yes	FYes	Restricted	None	13.4		91.5	3.5	999	1.4	111	Dead
Male	49	No	GYes	Active	None	10.4		102	2.72	999	2.19	1452	Dead
Male	61	Yes	HYes	Selfcare	None	10.8		92	3.2	999	1.3	706	Dead
Male	50	Yes	IYes	Restricted	None	11.9	No	107.5	3.3	999	1.2	982	Alive
Male	43	Yes	JNo	Active	None	11.8	No	87.8	0.5	999	0.7		Alive
Male	41	Yes	KYes	Active	None	13	No	94.2	3	999	1.1		Dead
Male	74	Yes	LYes	Active	None	15.7	No	96.7	1.3	999	0.3	277	Alive
Male	66	Yes	MNo	Active	None	13.3	No	90.1	8.5	999	0.8		Alive
Male	56	No	NYes	Active	None	13.7	No	93.8	1	999			Alive
Male	63	Yes	OYes	Ambulatory	Grade I/II	13.5	No	93	10.5	999	4.5	302	Alive
Female	41	Yes	PYes	Restricted	None	10.2	No	89.6	3.1	999	1.3	60	Alive
Male	72	Yes	QYes	Selfcare	Grade I/II	12.1	No	99.2	9.8	999	2.9	767	Dead
Male	60	Yes	RYes	Ambulatory	None	10.3	No	103.7	0.5	999	3.8	443	Alive
Male	64	Yes	SYes	Restricted	None	14.9	No	94.8	0.9	999	0.9	295	Alive
Male	75	Yes	TYes	Active	None	15.9	No	103.4	3.4	999	1.6	774	Dead
Male	71	Yes	UNo	Active	None	11.7	No	101	1.7	999	0.7	76.9	Alive
Male	73	Yes	VNo	Active	None	16.4	No	90.7	1	999	0.2	84	Alive
Male	66	Yes	WYes	Restricted	None	10.8	No	86.5	1.2	999	0.5	1001	Dead
Male	64	Yes	XYes	Restricted	None	10.7	No	88.1	3.8	999	1.6		Dead
Male	84	Yes	YNo	Disabled	None	13.1		111	1.3	999	0.7		Dead
Male	80	Yes	ZYes	Ambulatory	None	13.7		94.3	1.6	999	0.7	79	Alive
Male	45	Yes	AAYes	Restricted	None	13.6		98.4	1.3	999	0.7		Dead
Male	57	Yes	ABYes	Active	None	15.5	No	88.2	3.2	999	1		Alive
Male	61	Yes	ACYes	Restricted	None	12.2	No	89.5	1.1	999	0.4	70	Dead
Male	20	Yes	ADYes	Restricted	None	9.9		83.4	1.8	999	1.1	369	Alive

Create a virtual environment

Ensure reproducibility of your code and avoid packages incompatibilities

Why create a virtual environment?

- **Isolates Dependencies:** Prevents conflicts between projects.
- **Simplifies Management:** Makes package and version management easier.
- **Enhances Reproducibility:** Ensures code runs consistently across different setups.
- **Facilitates Collaboration:** Helps team members work with the same setup.
- **Improves Security:** Reduces the risk of security vulnerabilities.

```
conda create --name yprofiling python=3.12

conda activate yprofiling

pip install ydata-profiling==4.8.3 jupyter notebook

conda install -c anaconda ipykernel

python -m ipykernel install --user --name=yprofiling
```

Let’s profile our dataset!

1st glimpse of your dataset

Quickly access a summarized understanding of the challenges and quality issues of your dataset.

```python
import pandas as pd
from ydata_profiling import ProfileReport

df = pd.read_csv('hcc.csv')
report = ProfileReport(df, title='HCC Profile Report')

#export report as html file
report.to_file('hcc_report.html')

#preview report in the notebook
report
```

Compare populations

Quickly access a summarized understanding of the challenges and quality issues of your dataset.

```python
import pandas as pd
from ydata_profiling import ProfileReport

df = pd.read_csv('hcc.csv')

#Create the 2 sub-populations from the dataset
df1 = df[df['Gender']==0]
df2 = df[df['Gender']==1]

report1 = ProfileReport(df1, title='Female population profile report')
report2 = ProfileReport(df2, title='Male profile report')

#create the comparison report between both populations
compare = report1.compare(report2)

#Save the report as a html file
compare.to_file("hcc_compare_report.html")
```

How can you leverage ydata-profiling?

Explore more about data quality & data profiling at ydata.ai/register.

- **Quick Exploratory Data Analysis and Visualization:** fully understanding your data assets
- **Debugging and Troubleshooting:** coping with changing data requirements, sources, format, schemas (significantly boosting your ETL)
- **Data Management and Quality Control:** mitigating errors in production and correcting for problems happening in real-time, such as rare events, data drifts, fairness constraints, or misalignments with project goals

Thank You!

Learn more about *data quality, profiling & synthetic* data at
ydata.ai/resources