

## Introduzione a great\_expectations e data profiling



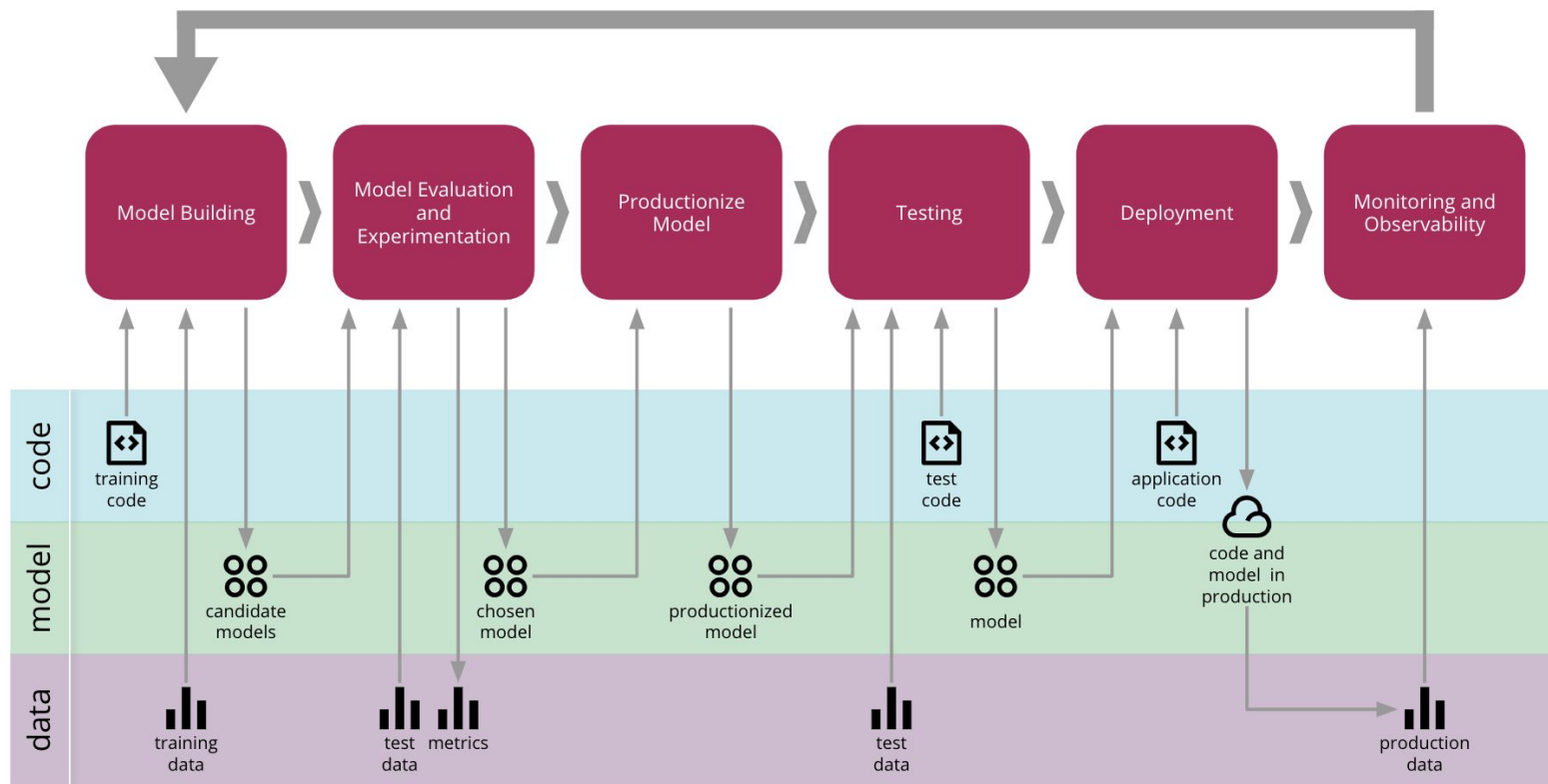
## Programma di oggi

- 1) Introduzione a **great\_expectations** e al concetti di **data testing**
- 2) Costruzione di data tests tramite **data profiling**
- 3) Esempi pratici su notebook

# Perche' testare dati?



## Esempio: Machine learning in produzione



# Il cuore del DevOps: CI/CD

## Continuous Integration/ Continuous Delivery

Facilitare l'interazione tra sviluppo e produzione dando la possibilità di migliorare continuamente codice tramite piccoli interventi che vengono automaticamente messi in produzione.

# Il cuore del DevOps: CI/CD

## Continuous Integration/ Continuous Delivery

Facilitare l'interazione tra sviluppo e produzione dando la possibilità di migliorare continuamente codice tramite piccoli interventi che vengono automaticamente messi in produzione.

**Piccoli interventi** → Affidabili, riproducibili e utilizzabili in produzione in qualunque momento

# Il cuore del DevOps: CI/CD

## Continuous Integration/ Continuous Delivery

**Obiettivo:** mantenere alta la qualità del codice

Pratiche più importanti:

1. Version control
2. Testing

## Il cuore del DevOps: CI/CD

### Continuous Integration/ Continuous Delivery

**Obiettivo:** mantenere alta la qualità del codice

Sappiamo come effettuare version control e testing per il software...

Possiamo applicare gli stessi concetti ai dati?



## Data Version Control

DVC permette di versionare dati o modelli (serializzati) in maniera completamente integrata a Git.

Files possono essere salvati localmente, su storage dedicato o su cloud.

Alternativa: **Git Large File Storage** (richiede server dedicato)



# Testing dei dati

## Qualità del dato

**Testing** = assicurarci di mantenere alta la qualità del dato nel tempo.



# Qualità del dato

Controllo della qualità del dato può avvenire su due livelli.

1. **Metadata** = tutto ciò che riguarda aspetti che definiscono il ruolo del dato all'interno di pipelines come dimensioni, orari di raccoglimento e processamento, etc.



## Qualità del dato

Controllo della qualità del dato puo' avvenire su due livelli.

1. **Metadata** = tutto ciò che riguarda aspetti che definiscono il ruolo del dato all'interno di pipelines come dimensioni, orari di raccoglimento e processamento, etc.
2. **Semantica** = qualità del contenuto del dato stesso come valori mancanti, duplicati, anomalie e outliers.



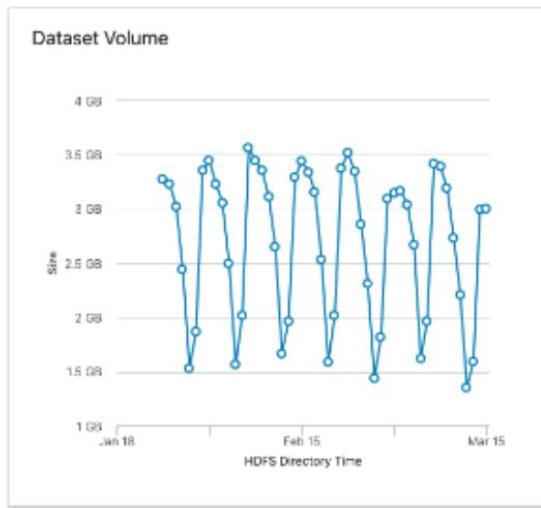
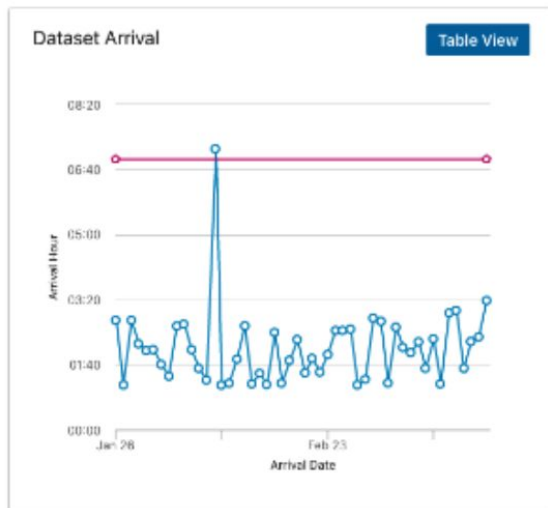
## Qualità del dato

Controllo della qualità del dato puo' avvenire su due livelli.

1. **Metadata** = tutto ciò che riguarda aspetti che definiscono il ruolo del dato all'interno di pipelines come dimensioni, orari di raccoglimento e processamento, etc.
2. **Semantica** = qualità del contenuto del dato stesso come valori mancanti, duplicati, anomalie e outliers.



## Gestione metadati

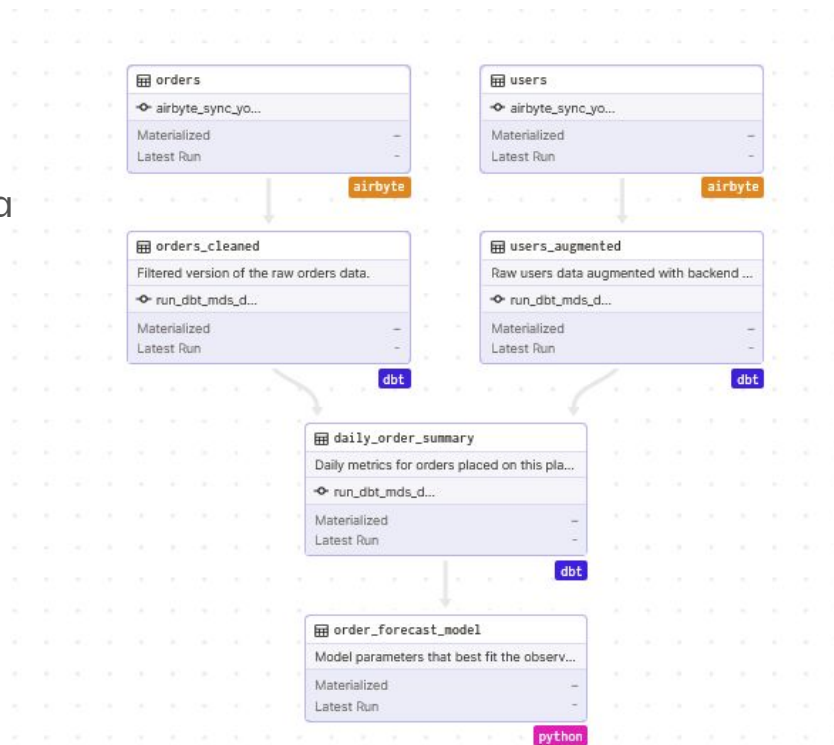


<https://engineering.linkedin.com/blog/2022/towards-data-quality-management-at-linkedin>

## Data orchestration

Controllo qualità del dato dal punto di vista dei metadati effettuata a livello di orchestrazione.

**Airflow**, **Dagster.io** sono strumenti molto popolari per la gestione di pipeline di orchestrazione dati.



<https://dagster.io/>



## Qualità del dato

Controllo della qualità del dato puo' avvenire su due livelli.

1. **Metadata** = tutto ciò che riguarda aspetti che definiscono il ruolo del dato all'interno di pipelines come dimensioni, orari di raccoglimento e processamento, etc.
2. **Semantica** = qualità del contenuto del dato stesso come valori mancanti, duplicati, anomalie e outliers.



## Qualità del dato

Controllo della qualità del dato può avvenire su due livelli.

1. **Metadata** = tutto ciò che riguarda aspetti che definiscono il ruolo del dato all'interno di pipelines come dimensioni, orari di raccoglimento e processamento, etc.
2. **Semantica** = qualità del contenuto del dato stesso come valori mancanti, duplicati, anomalie e outliers.



# testare i dati con great\_expectations



## great\_expectations

Libreria open-source che permette di sviluppare delle suites di test per i dati utilizzando un mini linguaggio Python espressivo.

**Obiettivo:** creazione di un open-standard facilmente condivisibile per l'analisi della qualità del dato.

Compatibile con **pandas**, **SQLAlchemy**, **Spark**

```
expect_column_values_to be  
between (  
  column="room_temp",  
  min_value=60,  
  max_value=75,  
  mostly=.95  
)
```



"Values in this column should  
be between 60 and 75, at  
least 95% of the time."

"Warning: more than 5% of  
values fell outside the  
specified range of 60 to 75."



## Tests are docs and docs are tests

Many data teams struggle to maintain up-to-date data documentation. Great Expectations solves this problem by rendering Expectations directly into clean, human-readable documentation.

Since docs are rendered from tests, and tests are run against new data as it arrives, your documentation is guaranteed to never go stale. Additional renderers allow Great Expectations to generate other type of "documentation", including slack notifications, data dictionaries, customized notebooks, etc.

# great\_expectations

gallery

<https://greatexpectations.io/expectations>



# Esempio su notebook



# Data profiling



## Data profiling

Analisi del contenuto semantico di un dataset. Profilazione permette di definire i **test** che il dato deve superare per essere considerato di qualità.

Passaggi importanti nella profilazione del dato.

1. Structure discovery
2. Content Discovery
3. Relationship discovery



## Structure discovery

Capire la corretta rappresentazione e formattazione dei dati e costruire statistiche sulle varie colonne che definiscono il dataset.

Es:

- Colonna A contiene valori numerici rappresentabili come int64, il valore minimo e' 0.
- Colonna B contiene *Personal Indentifiable Information* di tipo *Indirizzo*
- Colonna C contiene stringhe di carattere ordinale

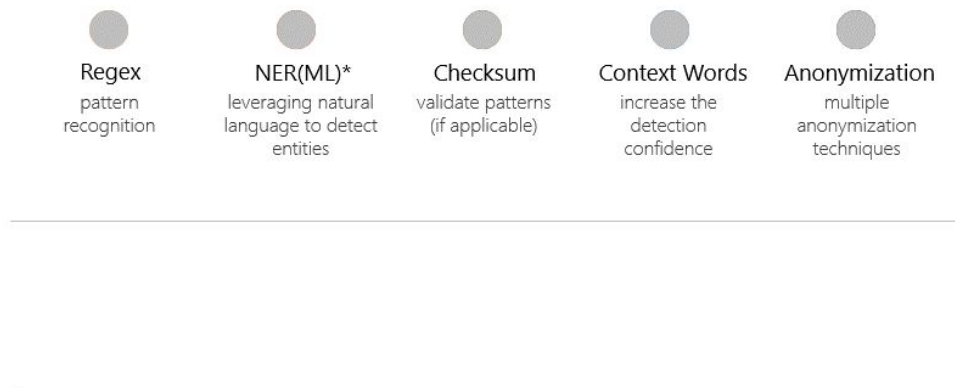


# Structure discovery

Entity recognition

**Esempio structure discovery:** libreria open source di Microsoft per automatizzare i processi di entity recognition.

## Presidio Detection Flow



\*NER – Named Entity Recognition

<https://microsoft.github.io/presidio/>



## Content discovery

Analisi puntuale sui contenuti di ciascuna colonna al fine di valutare la qualità dal punto di vista della presenza valori mancanti o di inconsistenze.

**Esempio:**

Etichette simili →

```
The column VFCTYPE contains ambiguous labels:  
[nan 'Régulier' 'régulier' 'Pouls régulier.' 'Rég' 'Reg' 'Irrég'  
'lent, régulier' 'irrégulier' 'rapide, régulier' 'Rég.'  
'Rapide, irrégulier' 'Régulier, rapide' 'avec de rares ES'  
'Rapide, régulier' 'Rapide régulier']
```



## Content discovery

Analisi puntuale sui contenuti di ciascuna colonna al fine di valutare la qualità dal punto di vista della presenza valori mancanti o di inconsistenze.

**Esempio:**

Valori mancanti →

shippedDate	status	comments
2003-01-10	Shipped	NaN
2003-01-10	Shipped	NaN
2003-01-10	Shipped	NaN
2003-01-10	Shipped	NaN
2003-01-11	Shipped	Check on availability.
...	...	...
NaN	In Process	NaN
NaN	In Process	NaN

## Relationship discovery

Comprendere le relazioni fra colonne di una determinata tabella o tra più tabelle diverse di un database.

Es:

- Colonna C = Colonna A + Colonna B
- Colonna E e' sempre maggiore di Colonna D
- Colonna F contiene date che devono essere più recenti di colonna G



## Esempio:

	orderNumber	productCode	quantityOrdered	priceEach	orderLineNumber	orderDate	requiredDate	shippedDate	status	comments
0	10100	S18_1749	30	136.00	3	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
1	10100	S18_2248	50	55.09	2	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
2	10100	S18_4409	22	75.46	4	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
3	10100	S24_3969	49	35.29	1	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
4	10101	S18_2325	25	108.06	4	2003-01-09	2003-01-18	2003-01-11	Shipped	Check on availability.
...	...	...	...	...	...	...	...	...	...	...
2991	10425	S24_2300	49	127.79	9	2005-05-31	2005-06-07	NaN	In Process	NaN
2992	10425	S24_2840	31	31.82	5	2005-05-31	2005-06-07	NaN	In Process	NaN
2993	10425	S32_1268	41	83.79	11	2005-05-31	2005-06-07	NaN	In Process	NaN
2994	10425	S32_2509	11	50.32	6	2005-05-31	2005-06-07	NaN	In Process	NaN
2995	10425	S50_1392	18	94.92	2	2005-05-31	2005-06-07	NaN	In Process	NaN

## Esempio:

	orderNumber	productCode	quantityOrdered	priceEach	orderLineNumber	orderDate	requiredDate	shippedDate	status	comments
0	10100	S18_1749	30	136.00	3	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
1	10100	S18_2248	50	55.09	2	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
2	10100	S18_4409	22	75.46	4	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
3	10100	S24_3969	49	35.29	1	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
4	10101	S18_2325	25	108.06	4	2003-01-09	2003-01-18	2003-01-11	Shipped	Check on availability.
...	...	...	...	...	...	...	...	...	...	...
2991	10425	S24_2300	49	127.79	9	2005-05-31	2005-06-07	NaN	In Process	NaN
2992	10425	S24_2840	31	31.82	5	2005-05-31	2005-06-07	NaN	In Process	NaN
2993	10425	S32_1268	41	83.79	11	2005-05-31	2005-06-07	NaN	In Process	NaN
2994	10425	S32_2509	11	50.32	6	2005-05-31	2005-06-07	NaN	In Process	NaN
2995	10425	S50_1392	18	94.92	2	2005-05-31	2005-06-07	NaN	In Process	NaN



## Esempio:


	orderNumber	productCode	quantityOrdered	priceEach	orderLineNumber	orderDate	requiredDate	shippedDate	status	comments
0	10100	S18_1749	30	136.00	3	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
1	10100	S18_2248	50	55.09	2	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
2	10100	S18_4409	22	75.46	4	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
3	10100	S24_3969	49	35.29	1	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
4	10101	S18_2325	25	108.06	4	2003-01-09	2003-01-18	2003-01-11	Shipped	Check on availability.
...	...	...	...	...	...	...	...	...	...	...
2991	10425	S24_2300	49	127.79	9	2005-05-31	2005-06-07	NaN	In Process	NaN
2992	10425	S24_2840	31	31.82	5	2005-05-31	2005-06-07	NaN	In Process	NaN
2993	10425	S32_1268	41	83.79	11	2005-05-31	2005-06-07	NaN	In Process	NaN
2994	10425	S32_2509	11	50.32	6	2005-05-31	2005-06-07	NaN	In Process	NaN
2995	10425	S50_1392	18	94.92	2	2005-05-31	2005-06-07	NaN	In Process	NaN

## Esempio:

	orderNumber	productCode	quantityOrdered	priceEach	orderLineNumber	orderDate	requiredDate	shippedDate	status	comments
0	10100	S18_1749	30	136.00	3	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
1	10100	S18_2248	50	55.09	2	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
2	10100	S18_4409	22	75.46	4	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
3	10100	S24_3969	49	35.29	1	2003-01-06	2003-01-13	2003-01-10	Shipped	NaN
4	10101	S18_2325	25	108.06	4	2003-01-09	2003-01-18	2003-01-11	Shipped	Check on availability.
...	...	...	...	...	...	...	...	...	...	...
2991	10425	S24_2300	49	127.79	9	2005-05-31	2005-06-07	NaN	In Process	NaN
2992	10425	S24_2840	31	31.82	5	2005-05-31	2005-06-07	NaN	In Process	NaN
2993	10425	S32_1268	41	83.79	11	2005-05-31	2005-06-07	NaN	In Process	NaN
2994	10425	S32_2509	11	50.32	6	2005-05-31	2005-06-07	NaN	In Process	NaN
2995	10425	S50_1392	18	94.92	2	2005-05-31	2005-06-07	NaN	In Process	NaN

# Content discovery

Rilevamento automatico inconsistenze etichette: dirty\_cat



**dirty\_cat**  
Version 0.3.0 ▶ Other versions

Star 618

Usage  
API  
About

## dirty\_cat: machine learning on dirty categories

*dirty\_cat* facilitates machine-learning on non-curated categories: **robust to morphological variants**, such as typos.

### Automatic features from heterogeneous dataframes

**SuperVectorizer**: a transformer automatically turning a pandas dataframe into a numpy array for machine learning – a default encoding pipeline you can tweak.

[An example](#)

### OneHotEncoder but for non-normalized categories

- **GapEncoder**, scalable and interpretable, where each encoding dimension corresponds to a topic that summarizes substrings captured.
- **SimilarityEncoder**, a simple modification of one-hot encoding to capture the strings.
- **MinHashEncoder**, very scalable

**Installing:** `$ pip install --user --upgrade dirty_cat`

[Recent changes](#)  
[Contributing](#)

<https://dirty-cat.github.io/stable/>

## Strumenti per il data profiling

- **Pandas profiling**

Compatibilita' con serie temporali, metadati di files e immagini



- **Data Profiler** (Capital One)

Focus importante su entity recognition e identificazione campi sensibili.



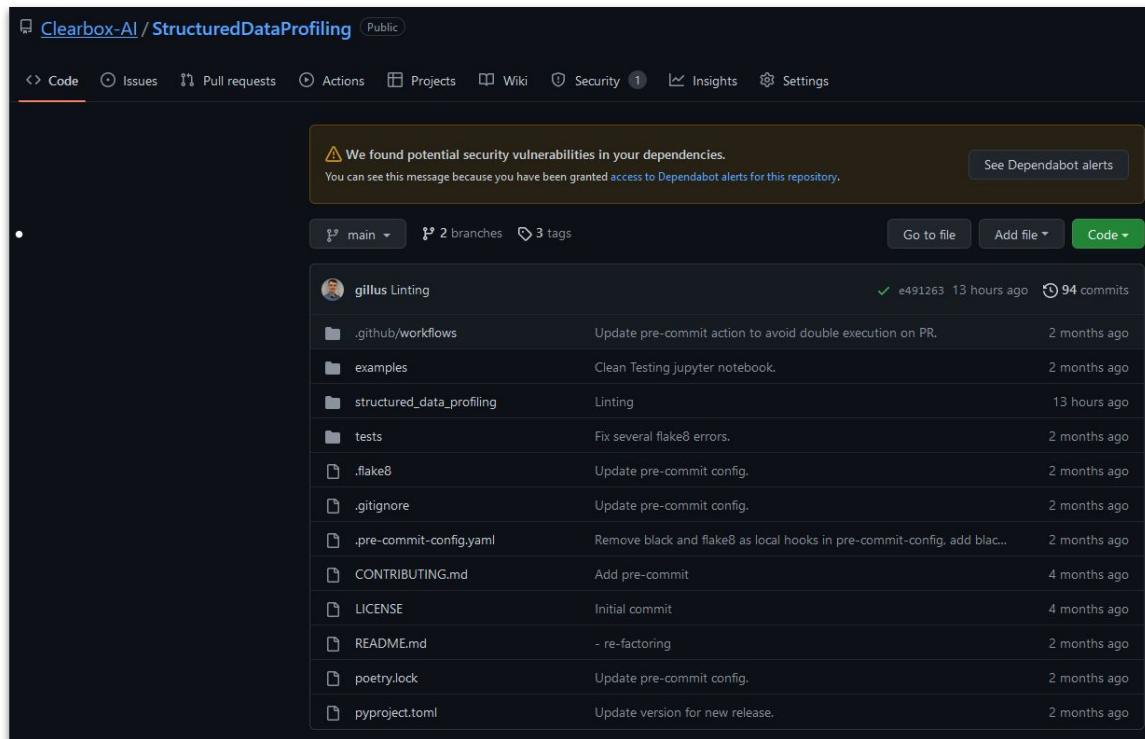
# Data Profiler

Entrambe le librerie forniscono poche soluzioni per quanto riguarda gli aspetti legati alla **Relationship discovery**

# StructuredDataProfiling

Profilazione automatica di un dataset (multi)tabellare.

Progetto open-source nato per offrire una soluzione di data profiling con focus su relationship discovery

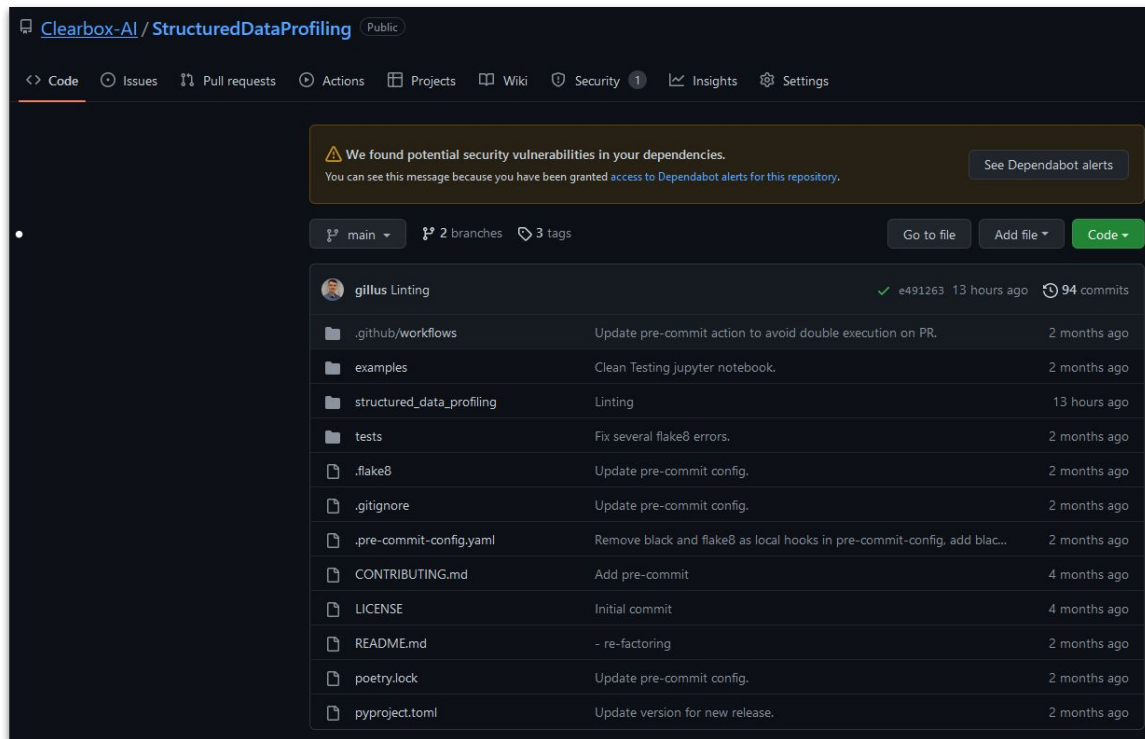


<https://github.com/Clearbox-AI/StructuredDataProfiling>

# StructuredDataProfiling

Profilazione automatica di un dataset (multi)tabellare.

Progetto open-source nato per offrire una soluzione di data profiling con focus su relationship discovery



<https://github.com/Clearbox-AI/StructuredDataProfiling>

## Profiling delle probabilita' condizionate

Analisi sistematica delle probabilita' condizionate:

$$P(A \mid \text{colonna } B=x)$$

Profiler effettua calcolo di queste probabilita' condizionate su tutte le possibili combinazioni di colonne, evidenziando le relazioni più significative.



## Riconoscimento regole deterministiche

Insieme di controlli euristici per rilevamento di regole deterministiche.

Colonna x può essere ricostruita utilizzando altre colonne dello stesso dataset?

Esempi:

- Colonna A = 1 se colonna B > 0, altrimenti = 0
- Colonna D = Colonna B + Colonna C

Modello ML fittato utilizzando colonne del dataset come target.

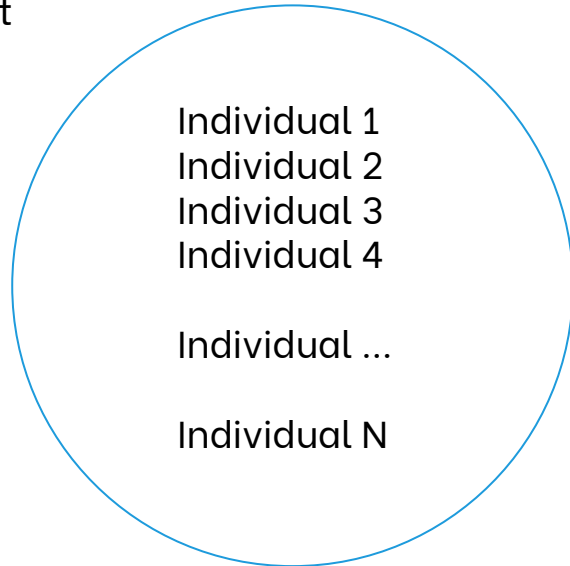




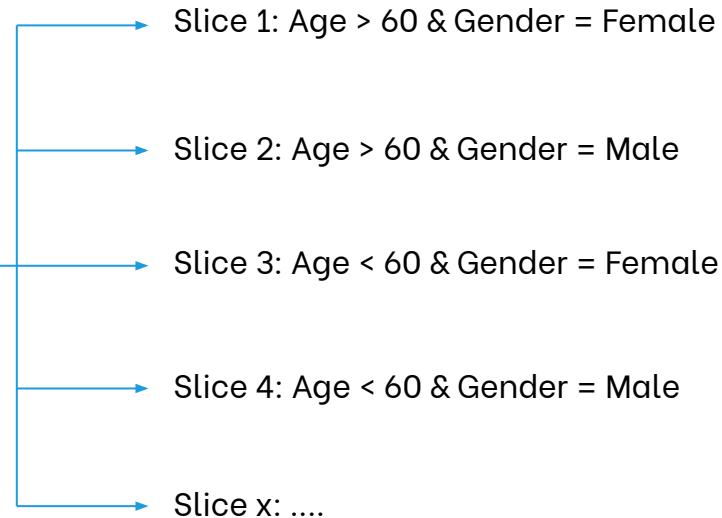
# Data slicing

Suddivisione del dataset in segmenti di dati significativi

Dataset



Slices



# Esempio su notebook





## StructuredDataProfiling

Work-in-progress e ancora alle primissime fasi. Stiamo cercando di spargere la parola e trovare contributors.

Lavoro costantemente focalizzato su:

- Identificazione nuovi controlli convertibili in tests
- Creazione di expectations custom





## Thanks for Reading

Feel free to contact us:



[www.clearbox.ai](http://www.clearbox.ai)



[support@clearbox.ai](mailto:support@clearbox.ai)



[@ClearboxAI](https://twitter.com/ClearboxAI)