Data Validation In A Data Centric Era

Validation and testing techniques through the different phases of a Data Science projects

By Aviram Berg

Data Quality

Characterizes how reliable the information is to serve some intended purpose.

Data Cleansing

Is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data.

Data Validation

The practice of checking the integrity, accuracy and structure of data before it is used for a business operation

Data Verification

Is the process of checking a copy of data to make sure that it is exactly equal to the original copy of the data.

Why Data Quality Is Important?

Impact on organisational decisions

- missing or incorrect data can result in wrong decision making

Legal obligations in certain business scenarios

- Medtech, consumer facing products, etc'...

Impact on machine learning models

- Cleaner data can greatly improve model performance

Potential for causing biased decisions in ML-based systems

- Not well understood, area of active research

Operational stability: missing and inconsistent data can cause havoc in production systems

- Crashes (e.g., due to "NullPointerExceptions" for missing attributes)
- Wrong predictions (e.g., change of scale in attributes)

Data: Academia vs the Real-World

Academic datasets

- Static
- Often down-sampled, cleaned and aggregated before publication
- Attributes typically well understood
- Most of time: size convenient for processing on desktop machines
- Example: UCI ML datasets

Real-world data

- Constantly changing
- Often hundreds of attributes
- Data originates from multiple sources / people / teams / systems
- Several potentially inconsistent copies
- Often too large to conveniently handle on a desktop machine
- Often difficult to access (e.g., data compressed and partitioned in a distributed filesystem)

Sources of Error in Data

Data entry errors

- Typos in forms
- Different spellings for the same real-world entity (e.g., addresses, names)

Measurement errors

- Outside interference in measurement process
- Placement of sensors

Distillation errors

- Editorial bias in data summaries
- Domain-specific statistical analyses not understood by database manager

Data integration errors

- Resolution of inconsistencies w.r.t. duplicate entries
- Unification of units, measurement periods

Completeness

Uniqueness

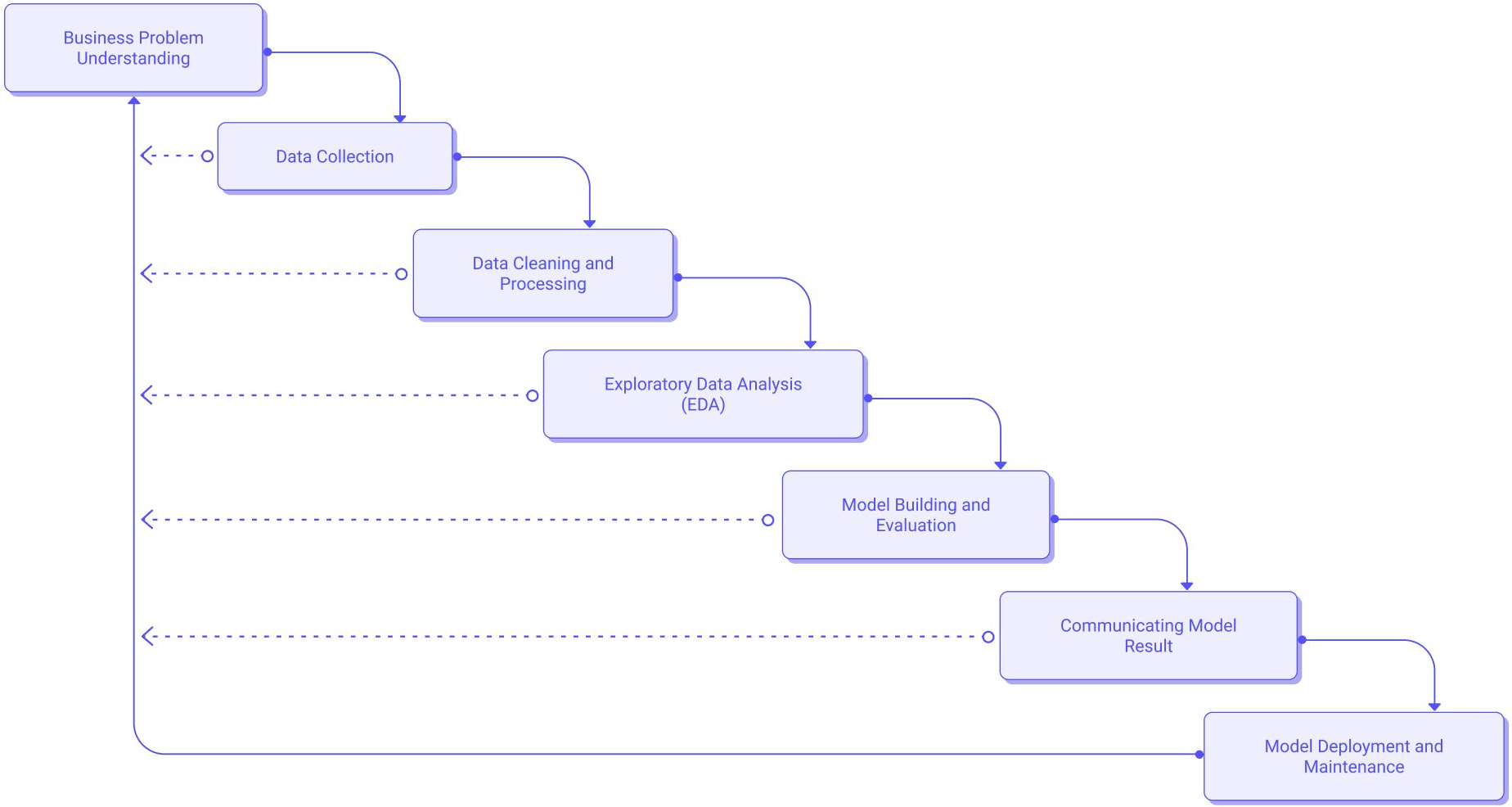
Consistency

Data Quality & Data Integrirty

Timeliness

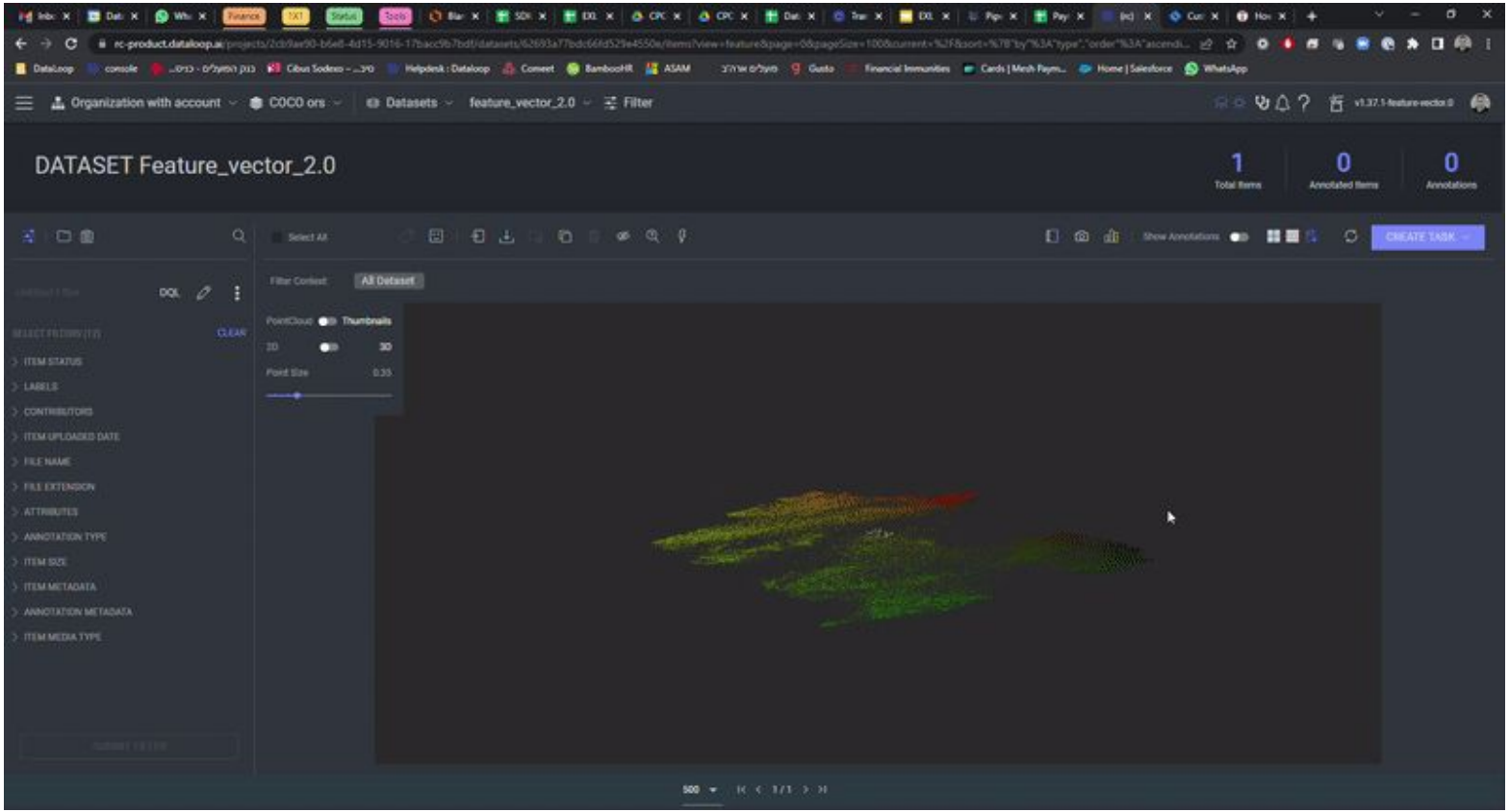
Accuracy

Validity

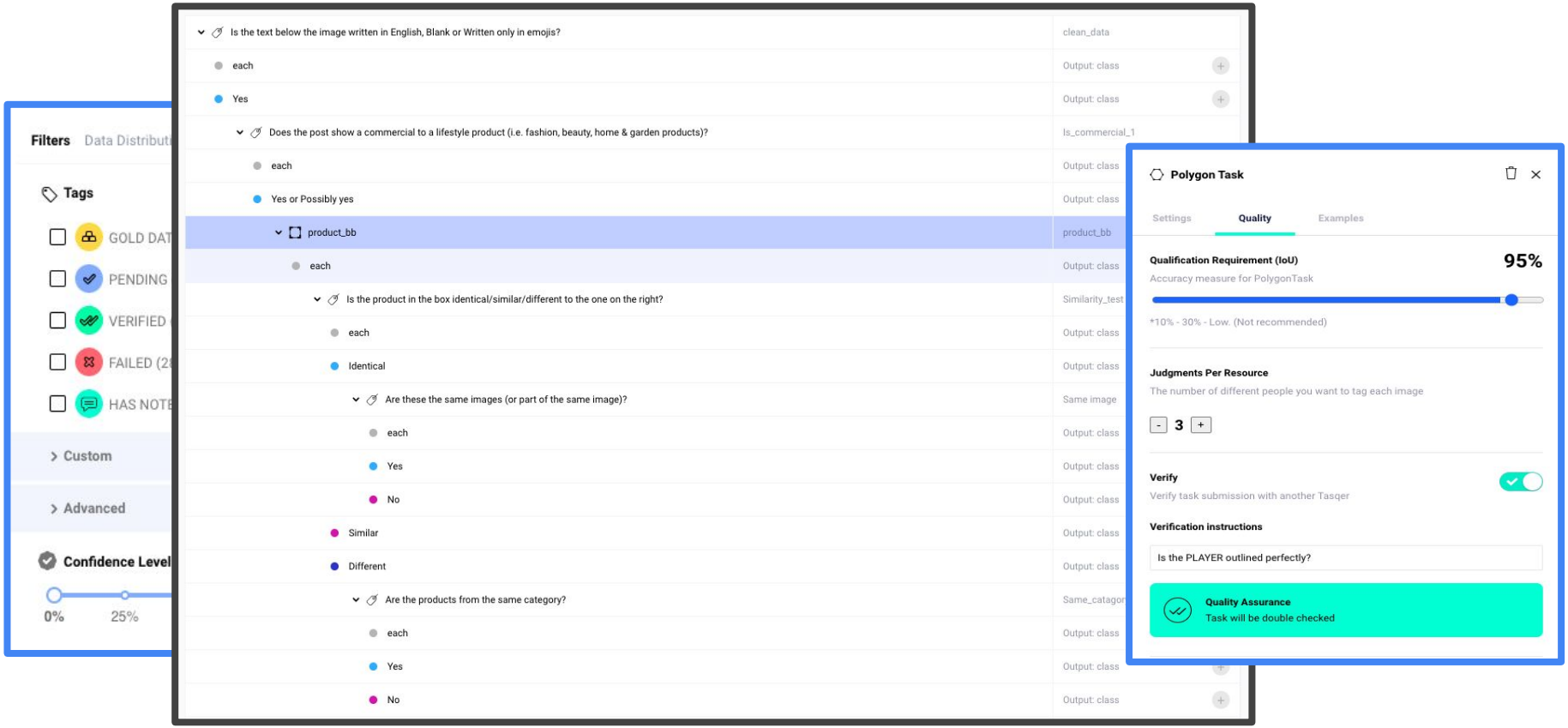

Business Problem Understanding
Data Collection
Data Cleaning and Processing
Exploratory Data Analysis (EDA)
Model Building and Evaluation
Communicating Model Result
Model Deployment and Maintenance

Data Collection -

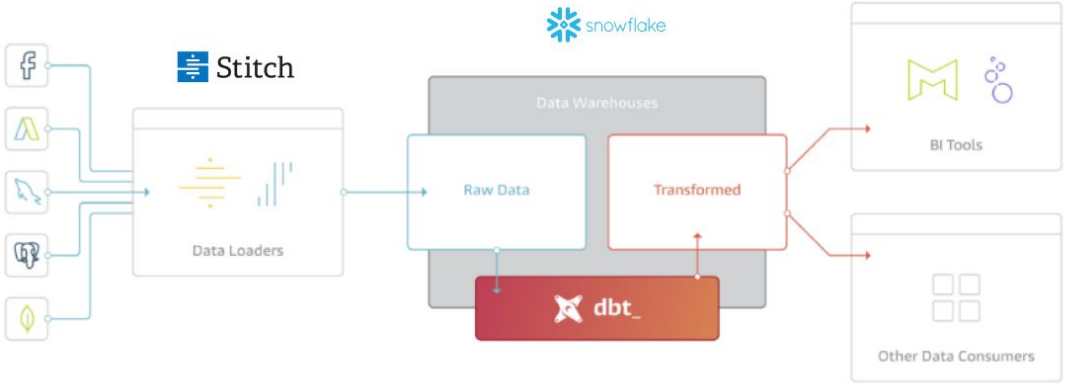
Dataloop



Data Collection -



Data Cleaning & Processing



Dbt's key functions

Testing

- Dbt tests data quality, integration, and code performance
- Create test programs that check for missing/incomplete entries, unique constraints, and accepted values within specific columns
- Manually run scripts that will then run automated tests and deploy changes after passing said tests

Deplyment

- Publish both public and private repositories that can then be referenced by other users

Documentation

- Automatically creates a visual representation of data flows throughout an organization
- Create documentation through schema files, reate documentation through schema files.

Dbt's Tests

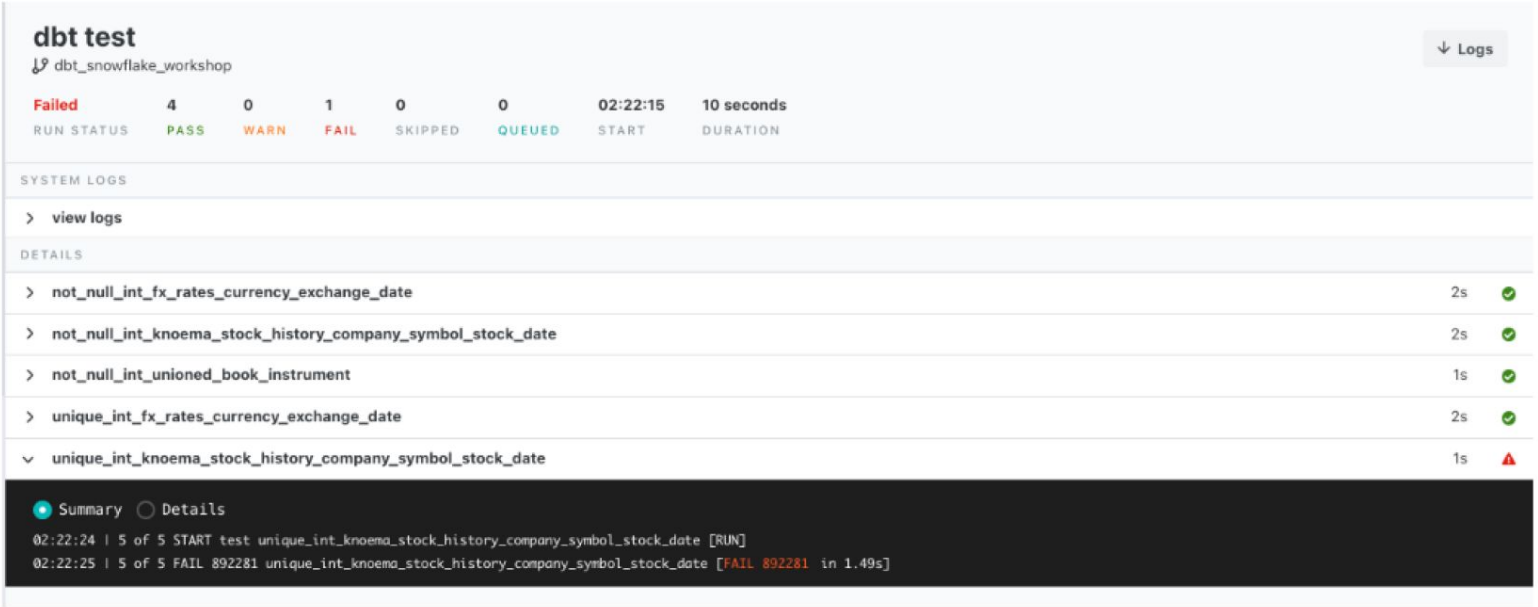
```yaml
version: 2

models:
  - name: orders
    columns:
      - name: order_id
        tests:
          - unique
          - not_null
      - name: status
        tests:
          - accepted_values:
              values: ['placed', 'shipped', 'completed', 'returned']
      - name: customer_id
        tests:
          - relationships:
              to: ref('customers')
              field: id
```

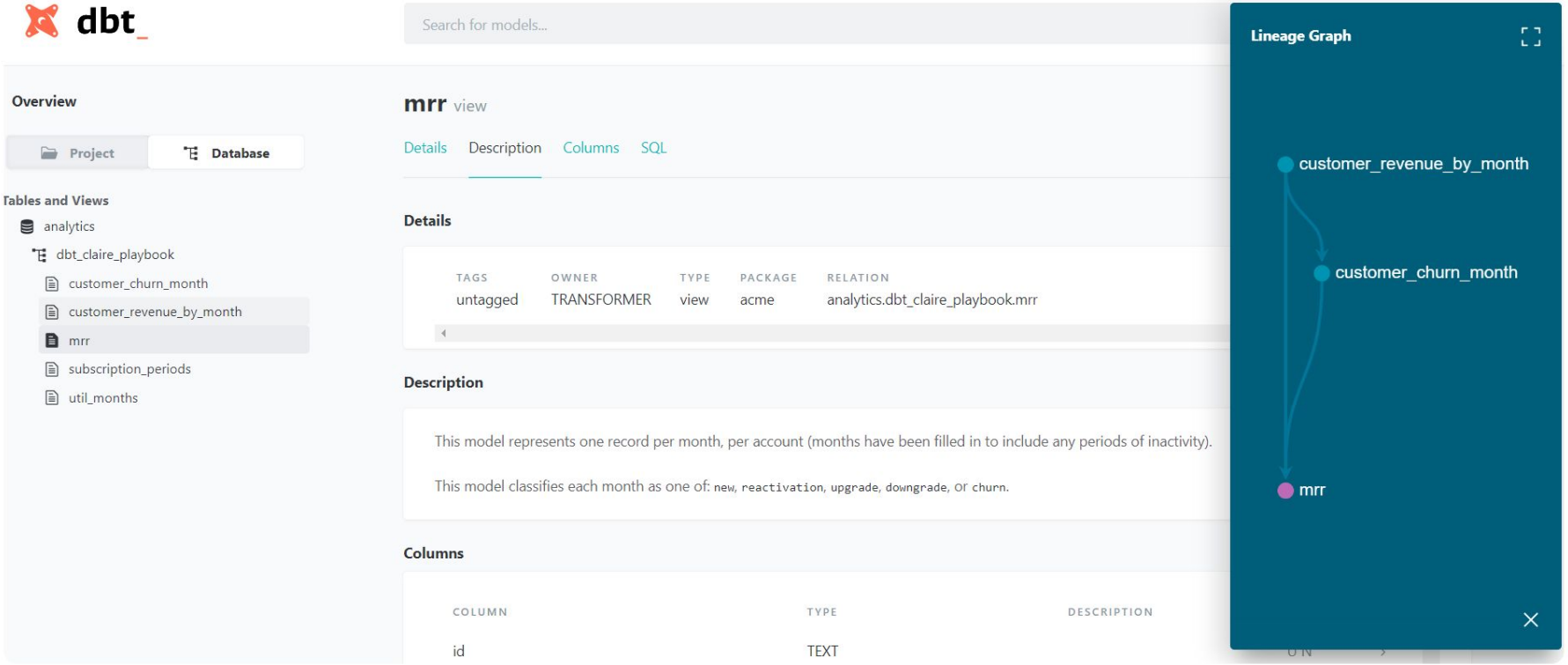
- `unique` : the `order_id` column in the `orders` model should be unique
- `not_null` : the `order_id` column in the `orders` model should not contain null values
- `accepted_values` : the `status` column in the `orders` should be one of `'placed'` , `'shipped'` , `'completed'` , or `'returned'`
- `relationships` : each `customer_id` in the `orders` model exists as an `id` in the `customers` table (also known as referential integrity)

Dbt's Tests

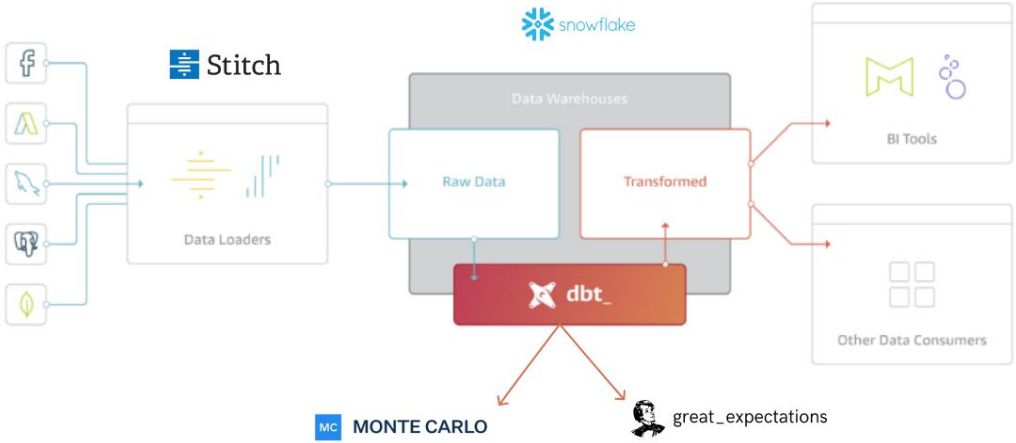
```
dbt test
```



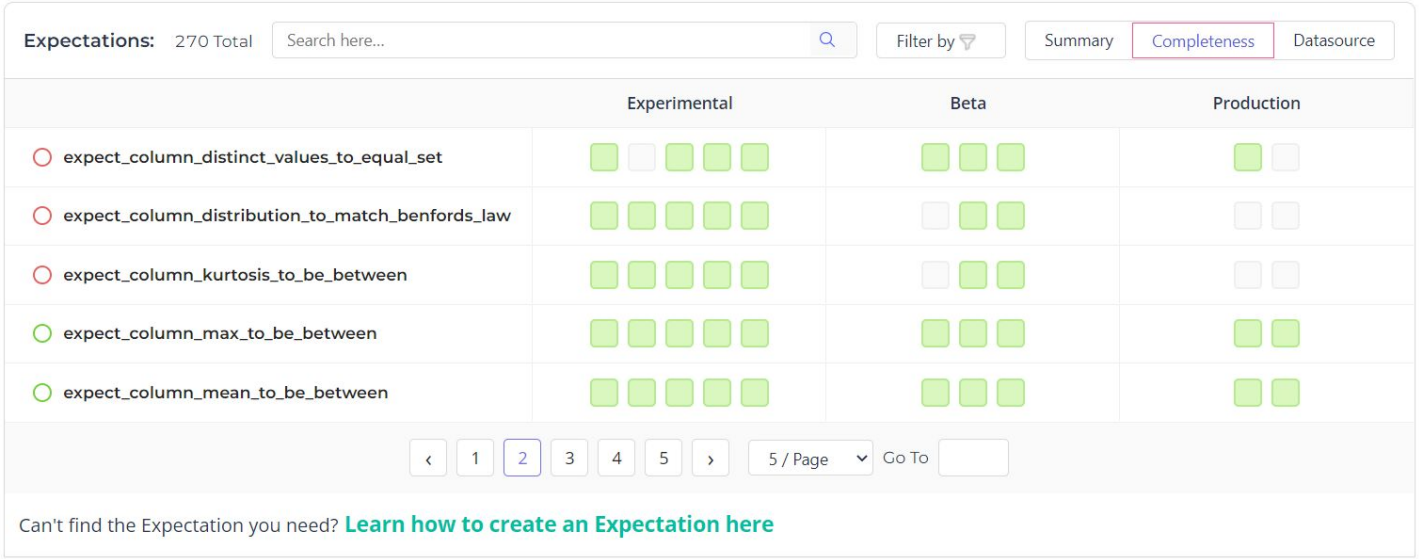
Dbt's Documentation



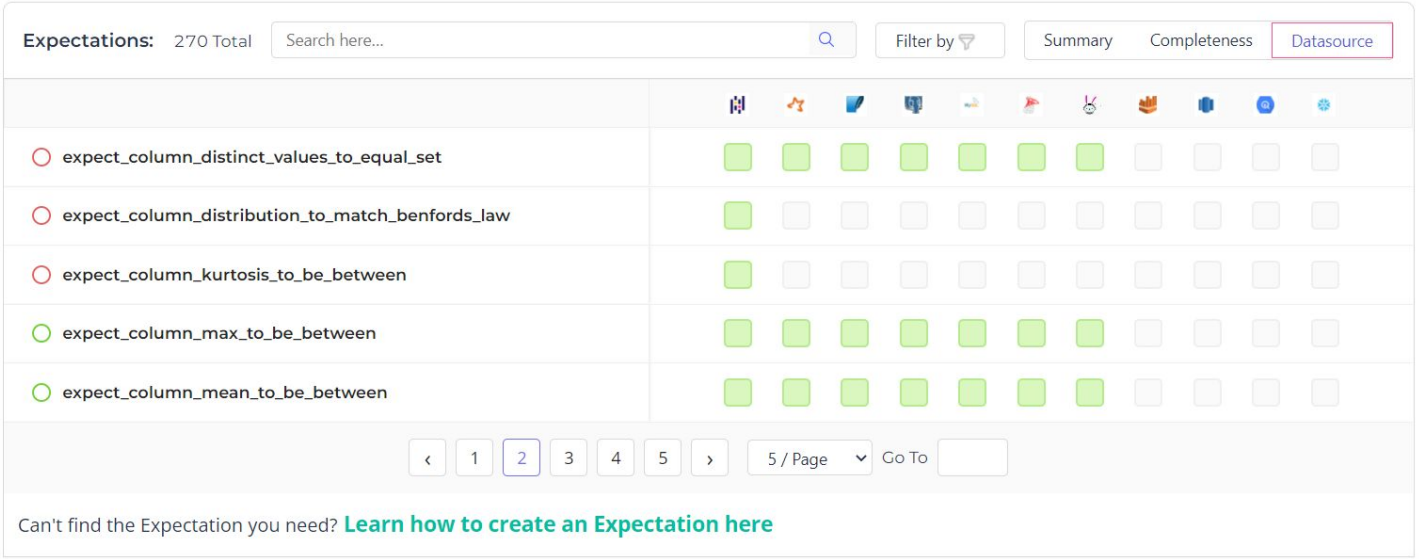
Data Cleaning & Processing



Great Expectations

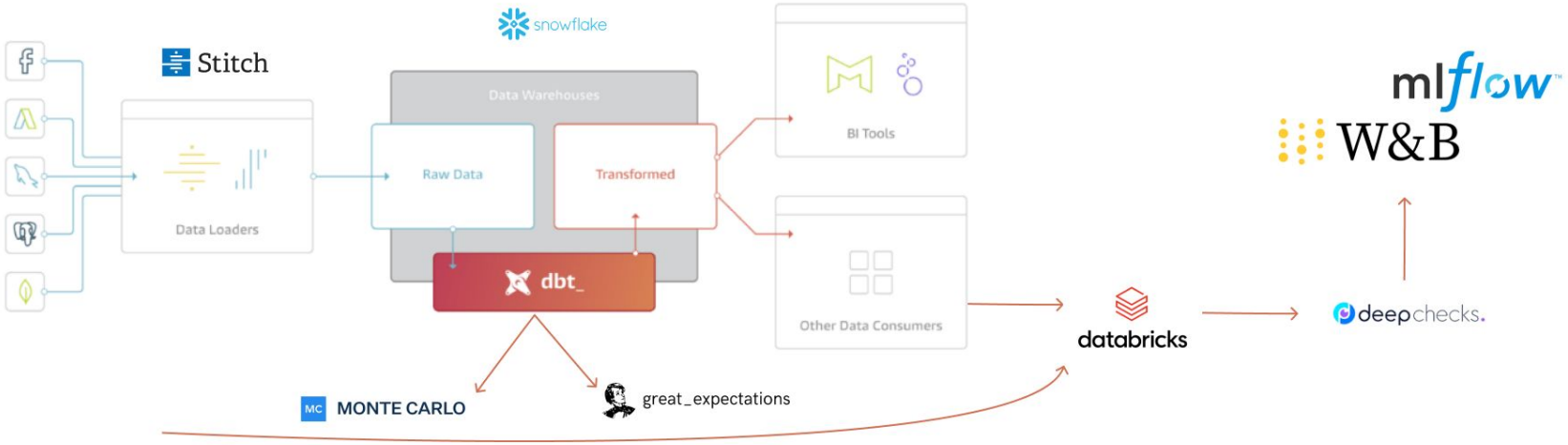


Great Expectations




Completeness
Uniqueness
Consistency
Data Quality &
Data Integrirty
Timeliness
Accuracy
Validity

Unstructured Pipeline



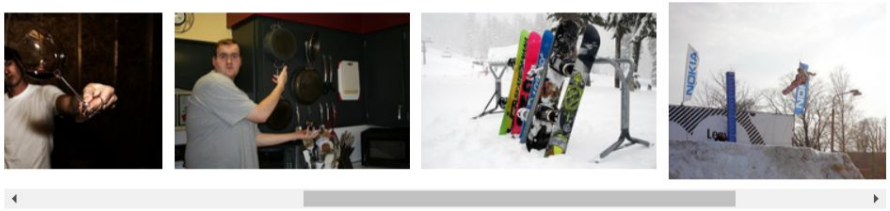
Deepchecks

Property "Brightness"

Total number of outliers: 6

Non-outliers range: 0.24 to 0.69

0.12 0.22 0.71 0.72



Property "RMS Contrast"

Total number of outliers: 3

Non-outliers range: 0.1 to 0.37

Contrast 0.07 0.38 0.4

Image

Deepchecks

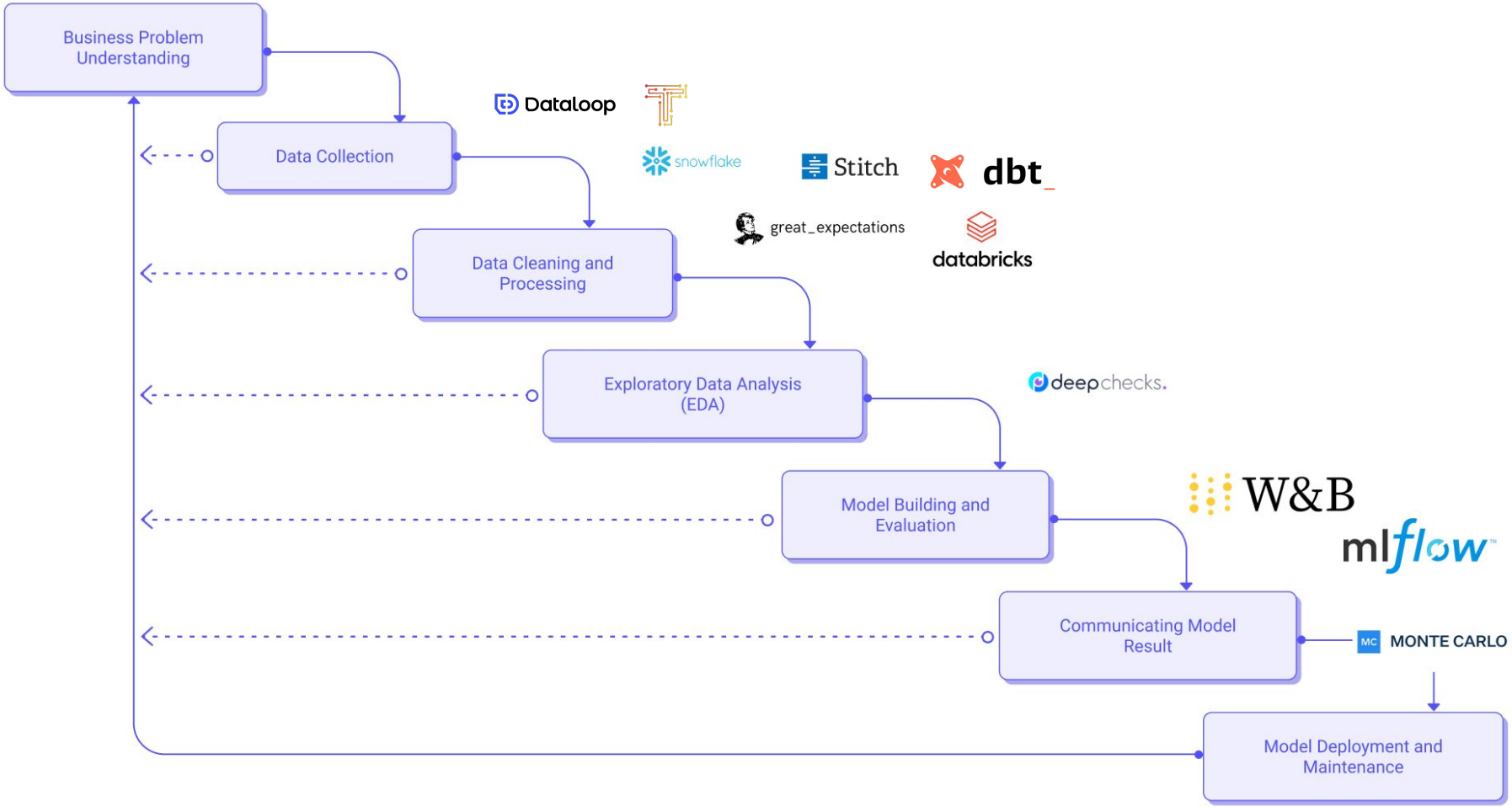
Similar Images

Total number of test samples with similar images in train: 5

Samples

Train			
Test			

```
check = SimilarImageLeakage().add_condition_similar_images_less_or_equal(3)
result = check.run(train_dataset=train_ds, test_dataset=test_ds_modified)
result.show(show_additional_outputs=False)
```


Business Problem Understanding
Data Collection
Data Cleaning and Processing
Exploratory Data Analysis (EDA)
Model Building and Evaluation
Communicating Model Result
Model Deployment and Maintenance
Dataloop
snowflake
Stitch
dbt_
great_expectations
databricks
deepchecks.
W&B
mlflow
MONTE CARLO