# **README — AI CapScan Dataset**

# **Purpose**

This project processes the **AI CapScan dataset** to produce a **relational, database-ready structure** in **3NF**.  
 It expands the base data with **synthetic records** for scalability and analytical testing, while preserving realistic statistical relationships.

The transformation pipeline:

* Cleans and validates the original CapScan data.
* Generates synthetic data following the same distributions.
* Normalizes the dataset into 3NF (dimension + fact tables).
* Exports UTF-8 CSVs for direct import into relational databases (PostgreSQL, MySQL, SQLite).

## **Input**

**Source file:** dataset\_AI\_capscan - Sheet1.csv

### **Typical content**

The dataset contains:

* **Categorical attributes:** Country, Sector, Technology, UseCase, OrganizationType, etc.
* **Numeric indicators:** AI adoption rates, digital maturity scores, investment levels, employees trained, etc.

### **Synthetic Data**

Synthetic data is generated to scale up the dataset (default SCALE\_FACTOR = 300).  
 The generator:

* Samples real records with replacement.
* Adds Gaussian noise to numeric values (preserving correlations).
* Randomly permutes categorical fields to maintain diversity.  
   Each record includes an **is\_synthetic** flag:

| Value | Meaning |  
 | 0 | Real record (from source file) |  
 | 1 | Synthetic record (auto-generated) |

## **Processing Script — data\_wrangling\_ai\_capscan.py**

### **Requirements**

Python ≥ 3.9

pip install pandas numpy

### **Workflow**

1. **Load & clean:** trims headers, infers types, replaces missing values.
2. **Generate synthetic data:** adds variation to numeric columns and shuffles categorical ones.
3. **Combine & normalize:** merges real and synthetic data; splits categorical dimensions.
4. **Export CSVs:** one fact table and multiple dimension tables in 3NF structure.

## **Output Files (3NF)**

| **File** | **Table** | **Description** |
| --- | --- | --- |
| AI\_CapScan\_3NF\_fact\_table.csv | fact\_capscan | Fact table linking all dimension IDs with numeric indicators |
| AI\_CapScan\_3NF\_<column>.csv | dim\_<column> | Dimension table for each categorical field |

Only the **fact table** contains the is\_synthetic field.

All CSVs are UTF-8 encoded and have headers without index columns.

**Database Schema (3NF)**

### **Example Schema**

CREATE TABLE fact\_capscan (

RecordID TEXT PRIMARY KEY,

is\_synthetic INT,

Country\_id TEXT,

Sector\_id TEXT,

Technology\_id TEXT,

AI\_Adoption FLOAT,

Investment FLOAT,

Employees INT

);

CREATE TABLE dim\_Country (

Country\_id TEXT PRIMARY KEY,

Country TEXT

);

CREATE TABLE dim\_Sector (

Sector\_id TEXT PRIMARY KEY,

Sector TEXT

);

CREATE TABLE dim\_Technology (

Technology\_id TEXT PRIMARY KEY,

Technology TEXT

);

**Relationships**

dim\_\* (1) ───< fact\_capscan

* Every categorical column becomes its own dimension table.
* fact\_capscan holds all foreign keys and numeric measures.
* is\_synthetic appears **only once**, in fact\_capscan.

## **Loading into Database**

### **PostgreSQL**

\copy dim\_Country FROM 'AI\_CapScan\_3NF\_Country.csv' CSV HEADER;

\copy dim\_Sector FROM 'AI\_CapScan\_3NF\_Sector.csv' CSV HEADER;

\copy dim\_Technology FROM 'AI\_CapScan\_3NF\_Technology.csv' CSV HEADER;

\copy fact\_capscan FROM 'AI\_CapScan\_3NF\_fact\_table.csv' CSV HEADER;

### **Add Foreign Keys**

ALTER TABLE fact\_capscan

ADD CONSTRAINT fk\_country FOREIGN KEY (Country\_id) REFERENCES dim\_Country(Country\_id),

ADD CONSTRAINT fk\_sector FOREIGN KEY (Sector\_id) REFERENCES dim\_Sector(Sector\_id),

ADD CONSTRAINT fk\_tech FOREIGN KEY (Technology\_id) REFERENCES dim\_Technology(Technology\_id);

## **Validation Queries**

-- Count synthetic vs real records

SELECT is\_synthetic, COUNT(\*) FROM fact\_capscan GROUP BY is\_synthetic;

-- Check referential integrity

SELECT COUNT(\*) FROM fact\_capscan f

LEFT JOIN dim\_Country c ON f.Country\_id = c.Country\_id

WHERE c.Country\_id IS NULL;

| **Check** | **Result** |
| --- | --- |
| Numeric noise bounded within ±20% of std | Done |
| Missing values standardized | Done |
| is\_synthetic only in fact table | Done |
| UTF-8 encoded output | Done |

## **Developer Notes**

* **Primary keys:** UUIDs (8-char hex strings) for all dimensions.
* **Fact table:** contains only numeric measures + foreign keys.
* **Synthetic expansion:** configurable via SCALE\_FACTOR.
* **Data integrity:** categorical FKs guaranteed valid through merge mapping.
* **Compatibility:** follows the same 3NF logic as Sentiment and Eurostat datasets.
* Designed for integration into the same EUROSTAT\_load\_db schema.