École SUPERIEURE PRIVEE D’IngénIERIE ET DE TECHnologIE

DATA EngIneeRIng PROjecT RePORT

Edusight

Students :
Hamza Zighni
Jihed Bakalti
Zeineb Moujehed
Meryem Bennani
Melek Amimi
Med Malek Manai

Professors
Mr Ridha Berrahal
Mme Jihene Jebri

2025 – 2026

List Of Contents

List of Figures

List of Tables

Introduction

Educational systems generate large volumes of heterogeneous data related to institutions, students, and academic outcomes. When properly consolidated and analyzed, this data can support performance monitoring, comparison across regions or establishment types and more informed decision-making. However, educational data is often dispersed across multiple sources, stored in different formats and not directly accessible through unified analytical tools which limits its practical value for stakeholders.

This project proposes **EduSight**, an end-to-end data-driven solution that transforms raw educational data into actionable insights through an integrated workflow. The project combines **Business Intelligence** and **Machine Learning** components: data is first collected, cleaned, and integrated through an ETL process, then structured into a data warehouse with a multidimensional model to enable efficient querying and reporting. On top of this foundation, interactive dashboards and analytical models are developed to support exploration, monitoring, and predictive use cases.

The overall objective of EduSight is to provide a unified platform that makes educational indicators easier to analyze and interpret. By centralizing reliable data, enabling intuitive visualization and supporting advanced analytics, the solution contributes to a more transparent and data-driven approach to understanding educational performance and supporting continuous improvement.

Chapter 1 : Project Context

1.1 Introduction

This chapter introduces the overall context of the EduSight project. It presents the business background in which the project is situated, the key challenges motivating its development and the proposed data-driven solution. In addition, this chapter clarifies the main concepts, definitions and tools used throughout the project to ensure a common understanding before addressing data processing, modeling, and analytical results in the subsequent chapters.

1.2 Business Understanding

The following section describes the business context of the project, outlines the main challenges encountered by educational stakeholders and introduces the proposed solution.

1.2.1 Business Context

Educational institutions generate large volumes of data related to academic performance, institutional characteristics, student enrollment, and learning outcomes. These data originate from multiple administrative systems and public educational platforms and are continuously enriched over time. When properly exploited, such data can provide valuable insights into institutional performance, student success, and systemic inequalities across regions and establishment types.

However, in many educational environments, available data remains scattered across heterogeneous sources, difficult to interpret, and insufficiently leveraged for operational and strategic decision-making. Stakeholders such as students, teachers, and academic administrators often lack unified tools that allow them to explore, analyze, and monitor educational indicators in a clear and actionable manner. As a result, decision-making processes remain largely reactive and descriptive rather than proactive and analytical.

This context highlights the growing need for integrated analytical solutions capable of consolidating educational data and transforming it into meaningful information that supports monitoring, comparison, and performance evaluation across educational institutions.

1.2.2 Problem Statement

Despite the availability of extensive educational data, several critical challenges persist:

- **Limited access to meaningful insights for students:** Students often have little visibility into their academic performance beyond raw results, making it difficult to assess progress, identify weaknesses, or compare outcomes over time.
- **Insufficient analytical support for teachers:** Teachers face difficulties in analyzing class-level and institutional trends due to the absence of advanced visualization and analytical tools that support early detection of learning difficulties.
- **Fragmented and inefficient decision-making for administrators:** Academic administrators rely on disconnected data sources and static reports, limiting their ability to monitor institutional performance, identify at-risk populations, and support strategic planning.

These challenges reduce the effectiveness of academic monitoring and hinder proactive intervention. There is therefore a clear need for a unified, data-driven solution that can consolidate educational data, improve accessibility, and support informed decision-making at all levels of the educational system.

1.2.3 Proposed Solution

The EduSight Project addresses these challenges by proposing an integrated analytical platform designed to centralize educational data and provide actionable insights for all stakeholders. The solution is built around the following key components:

- **Centralized data consolidation:** Aggregation of educational data from multiple sources into a unified and structured repository.
- **Interactive dashboards and indicators:** Visual representations and Key Performance Indicators (KPIs) that enable users to monitor academic performance, enrollment trends, and institutional outcomes.
- **Analytical and predictive capabilities:** Data-driven tools that support performance analysis, risk identification, and trend monitoring to facilitate proactive academic management.

By transforming raw educational data into structured insights, EduSight enables more transparent, efficient, and informed decision-making. The platform supports continuous monitoring and evaluation, contributing to improved learning outcomes and enhanced institutional performance.

Figure 2.1: EduSights logo.

1.3 Concept and Definitions

1.3.1 Core Definitions

1.3.1.1 Project Methodology: CRISP-DM

A standard model for data science projects, structured into six key phases : understanding objectives, exploring data, preparing data, modeling, evaluating results, and deployment.

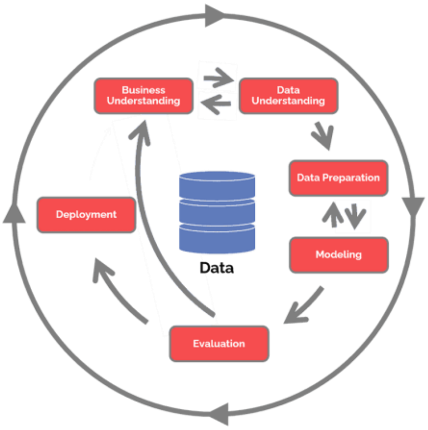


figure 4 – The CRISP-DM Framework

1.3.1.2 Machine Learning

Machine Learning (ML) is a branch of Artificial Intelligence that enables systems to learn patterns from data rather than being explicitly programmed. By training on historical observations, ML models generalize relationships between input features and target outcomes, allowing them to produce predictions or classifications on new, unseen data.

1.3.1.3 Extraction, Transformation, and Loading (ETL) Process

ETL (Extract, Transform, Load) is a core data engineering process used to convert raw and heterogeneous data into structured, analysis-ready datasets stored in a data warehouse. It ensures that data is consistent, validated and aligned with the target analytical model.

The ETL workflow typically includes:

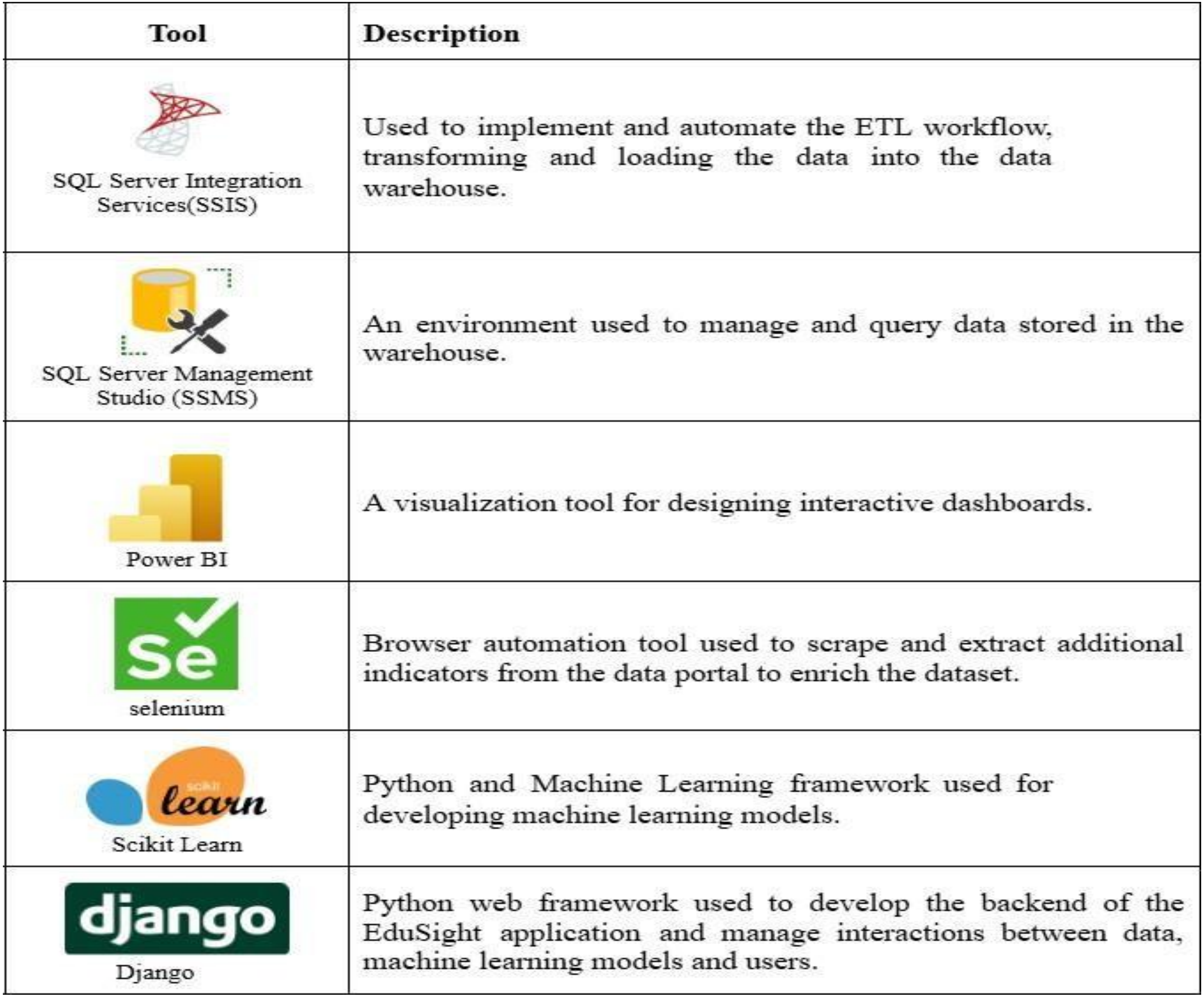
1.3.1.3.1 **Data Cleaning:** handling missing values, removing duplicates and correcting inconsistencies.

1.3.1.3.2 **Transformation:** standardizing formats, encoding categorical variables when needed, reshaping structures and deriving new fields based on predefined rules.

1.3.1.3.3 **Loading:** inserting the curated outputs into the target storage layer (e.g., a data warehouse) to support reporting and advanced analytics.

1.3.2 Tools Overview

Figure 2.2: Main tools used across the project workflow and their roles.



Tool	Description
SQL Server Integration Services(SSIS)	Used to implement and automate the ETL workflow, transforming and loading the data into the data warehouse.
SQL Server Management Studio (SSMS)	An environment used to manage and query data stored in the warehouse.
Power BI	A visualization tool for designing interactive dashboards.
selenium	Browser automation tool used to scrape and extract additional indicators from the data portal to enrich the dataset.
Scikit Learn	Python and Machine Learning framework used for developing machine learning models.
Django	Python web framework used to develop the backend of the EduSight application and manage interactions between data, machine learning models and users.

Chapter 2 : Data Understanding and Preparation

2.1 Introduction

This chapter presents the data-related foundations of the project, from source identification to the construction of a structured analytical model. It first introduces the data sources and acquisition methods, combining official open data with enriched indicators collected through web scraping. The chapter then focuses on data understanding and quality assessment, providing insights into the structure, completeness, consistency, and reliability of the collected information.

Based on these observations, a comprehensive data preparation and integration process was implemented using ETL workflows to clean, transform, and standardize the data. Finally, the chapter concludes with the multidimensional modeling adopted for the business intelligence layer, highlighting the star schema design used to support efficient analysis, reporting, and downstream machine learning tasks.

2.2 Data Sources and Acquisition

This project relies on two complementary data assets extracted from the same official French open-data ecosystem.

The primary dataset is a CSV file corresponding to the fr-en-annuaire-education dataset. It contains structured administrative and descriptive attributes for educational institutions.

To enrich this baseline for BI analysis, four additional indicators were collected directly from the portal’s online explorer through automated web scraping: *number of students*, *success rate*, *mention rate*, and *Indice de Position Sociale (IPS)*. The scraping process was implemented using **Selenium WebDriver** to automate browser interactions and access dynamically rendered content, enabling reliable extraction from JavaScript-driven interfaces [3, 4].

2.3 Data Understanding and Quality Assessment

The consolidated dataset contains **252,669 rows** and **11 columns**, organized as a fact-like table where each record corresponds to an institution observation described through foreign keys (*service, geography, establishment, date, section, track/voie, institution type*) and quantitative indicators (*number of students, success rate, mention rate, IPS*).

A data quality profiling phase was conducted to better understand the characteristics and limitations of the dataset prior to integration. The analysis focused on **completeness, uniqueness, consistency, and numerical validity** of the available attributes.

Overall, missing values were minimal for core identifiers, with no missing data observed for Geographie_FK and Etablissement_FK. Limited missingness was identified for Service_FK (0.47%), Date_FK (2.84%), and TypeEtab_FK (0.97%). Higher missing rates were observed for Section_FK (22.87%) and Voie_FK (25.49%), which is consistent with the fact that these attributes are not applicable to all types of establishments. Similarly, enriched performance indicators exhibited moderate missingness, notably for Taux_de_reussite (19.18%), Taux_de_mentions (19.18%), and IPS (7.90%).

Uniqueness analysis revealed a substantial number of exact duplicate records (**140,979 rows**, approximately **55.8%**), indicating repeated identical entries in the consolidated dataset. Consistency checks also highlighted formatting constraints, including the use of semicolon delimiters and French decimal notation, which resulted in some numerical fields initially being stored as text.

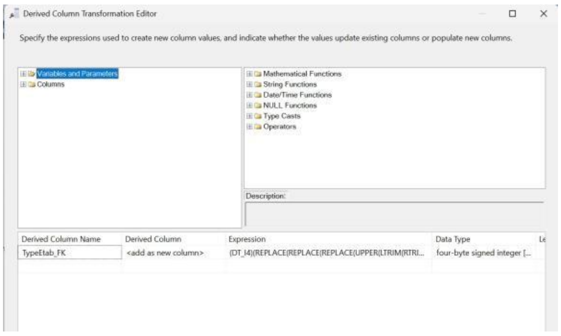
Finally, an exploratory review of indicator ranges confirmed plausible value distributions. The number of students (Nb_eleves) ranged from 2 to 2403, success rates (Taux_de_reussite) from 16.6 to 100, mention rates (Taux_de_mentions) from 0 to 100, and IPS values from 43.3 to 158.9, with no negative values or out-of-bound rates observed.

2.4 Data Preparation and Integration

In this phase of the project, the focus was on preparing, transforming, and structuring the data to ensure its readiness for analysis and model development. Given the complex nature of the raw data, an efficient ETL process was implemented. After completing the extraction phase, the transformed data was systematically processed and loaded into the data warehouse.

Derived Column Tool

Used to generate standardized fields and ETL-ready keys from raw attributes. In our case, it was applied to derive fields such as TypeEtab_FK by cleaning text (trim/upper/replace) and enforcing a consistent representation before loading.



Derived Column Transformation Editor

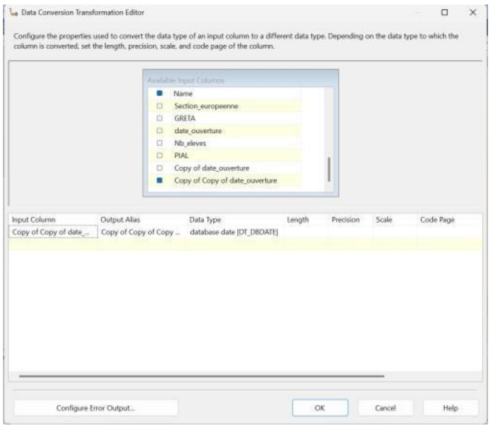
Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Le
TypeEtab_FK	<add as new column>	(DT_I4)(REPLACE(REPLACE(REPLACE(UPPER(LTRIM(RTRI...	four-byte signed integer [...	

Figure 3.1: Derived Column transformation used to standardize TypeEtab_FK.

Data Conversion Tool

Ensures schema consistency by converting raw columns into warehouse-compatible data types. In our flow, fields such as dates (e.g., date_ouverture) and numeric indicators were converted to proper SSIS/SQL types to avoid downstream load errors.



Data Conversion Transformation Editor

Configure the properties used to convert the data type of an input column to a different data type. Depending on the data type to which the column is converted, set the length, precision, scale, and code page of the column.

Available Input Columns

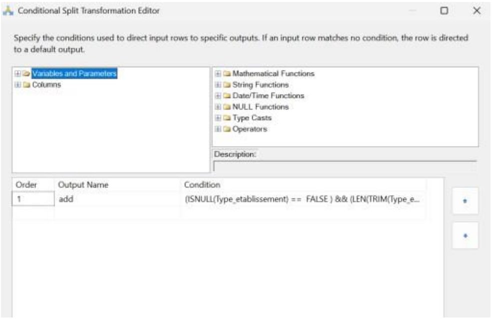
- ■ Name
- □ Section_europeenne
- □ GRETA
- □ date_ouverture
- □ Nb_eleves
- □ PIAL
- □ Copy of date_ouverture
- ■ Copy of Copy of date_ouverture

Input Column	Output Alias	Data Type	Length	Precision	Scale	Code Page
Copy of Copy of date_...	Copy of Copy of Copy ...	database date [DT_DBDATE]				

Figure 3.2: Data Conversion step to enforce warehouse-compatible data types.

Conditional Split Tool

Implements data-quality routing by directing records to different outputs based on validation rules. Here, it was used to keep rows meeting required conditions (e.g., non-null/non-empty Type_etablissement) while sending non-conforming rows to a default/reject path.


Conditional Split Transformation Editor

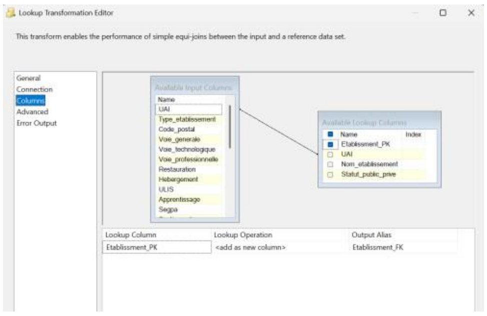
Specify the conditions used to direct input rows to specific outputs. If an input row matches no condition, the row is directed to a default output.

Order	Output Name	Condition
1	add	(ISNULL(Type_etablissement) == FALSE) && (LEN(TRIM(Type_e...

Figure 3.3: Conditional Split used to route valid vs. non-conforming records.

Lookup Tool

Performs key mapping against reference tables to enforce referential integrity. In our case, it was used to retrieve surrogate keys (e.g., Etablissement_PK) and populate the corresponding foreign key (Etablissement_FK) in the target model.


Lookup Transformation Editor

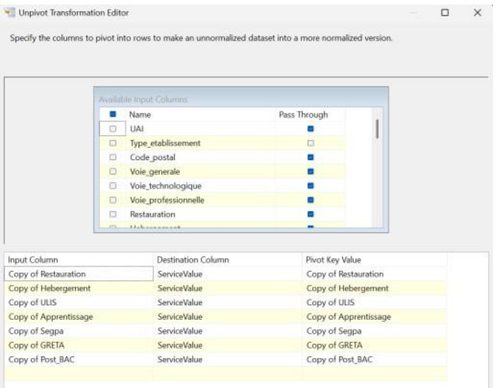
This transform enables the performance of simple equi-joins between the input and a reference data set.

Lookup Column	Lookup Operation	Output Alias
Etablissment_PK	<add as new column>	Etablissment_FK

Figure 3.4: Lookup step for surrogate key retrieval and foreign key mapping.

Unpivot Tool

Converts a wide structure into a normalized long format, which is more suitable for warehousing and analysis. Here, multiple service-related columns were unpivoted into a single attribute/value representation (e.g., service name → ServiceValue).



Specify the columns to pivot into rows to make an unnormalized dataset into a more normalized version.

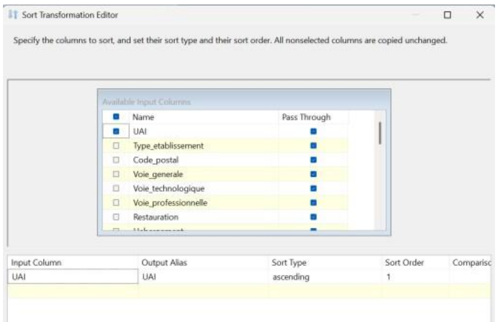
■	Name	Pass Through
□	UAI	■
□	Type_etablissement	□
□	Code_postal	■
□	Voie_generale	■
□	Voie_technologique	■
□	Voie_professionnelle	■
□	Restauration	■

Input Column	Destination Column	Pivot Key Value
Copy of Restauration	ServiceValue	Copy of Restauration
Copy of Hebergement	ServiceValue	Copy of Hebergement
Copy of ULIS	ServiceValue	Copy of ULIS
Copy of Apprentissage	ServiceValue	Copy of Apprentissage
Copy of Segpa	ServiceValue	Copy of Segpa
Copy of GRETA	ServiceValue	Copy of GRETA
Copy of Post_BAC	ServiceValue	Copy of Post_BAC

Figure 3.5: Unpivot transformation converting wide attributes into a normalized long format.

Sort Tool

Prepares ordered datasets required by specific SSIS transformations and ensures stable join behavior. In our pipeline, sorting (e.g., by UAI) was applied as a prerequisite step before performing merge-based joins.



Specify the columns to sort, and set their sort type and their sort order. All nonselected columns are copied unchanged.

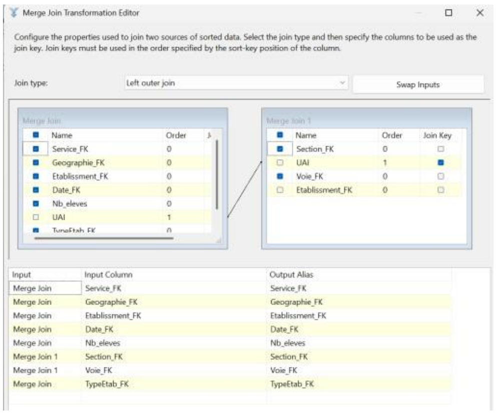
■	Name	Pass Through
■	UAI	■
□	Type_etablissement	■
□	Code_postal	■
□	Voie_generale	■
□	Voie_technologique	■
□	Voie_professionnelle	■
□	Restauration	■

Input Column	Output Alias	Sort Type	Sort Order	Comparisc
UAI	UAI	ascending	1	

Figure 3.6: Sort step applied as a prerequisite for merge-based joins.

Merge Join Tool

Integrates multiple sorted streams using a join condition to build a consolidated dataset. In our case, a left outer join was used to combine complementary attributes (e.g., Section_FK, Voie_FK) with the main flow while preserving all base records.



Configure the properties used to join two sources of sorted data. Select the join type and then specify the columns to be used as the join key. Join keys must be used in the order specified by the sort-key position of the column.

Join type: Left outer join

Merge Join

■	Name	Order
■	Service_FK	0
■	Geographie_FK	0
■	Etablissment_FK	0
■	Date_FK	0
■	Nb_eleves	0
□	UAI	1

Merge Join 1

■	Name	Order	Join Key
■	Section_FK	0	□
□	UAI	1	■
■	Voie_FK	0	□
□	Etablissment_FK	0	□

Input	Input Column	Output Alias
Merge Join	Service_FK	Service_FK
Merge Join	Geographie_FK	Geographie_FK
Merge Join	Etablissment_FK	Etablissment_FK
Merge Join	Date_FK	Date_FK
Merge Join	Nb_eleves	Nb_eleves
Merge Join 1	Section_FK	Section_FK
Merge Join 1	Voie_FK	Voie_FK
Merge Join	TypeEtab_FK	TypeEtab_FK

Figure 3.7: Merge Join used to integrate sorted streams while preserving base records.

2.5 Multidimensional Modeling

This section presents the multidimensional modeling implemented for the BI layer, including the SSIS loading workflow and the resulting star schema design that organizes the data into a central fact table linked to descriptive dimensions for efficient querying and reporting.

Figure 3.8 summarizes the SSIS data flow used to standardize data types, reshape attributes, map surrogate keys via lookups, and load the cleaned outputs into the warehouse tables.

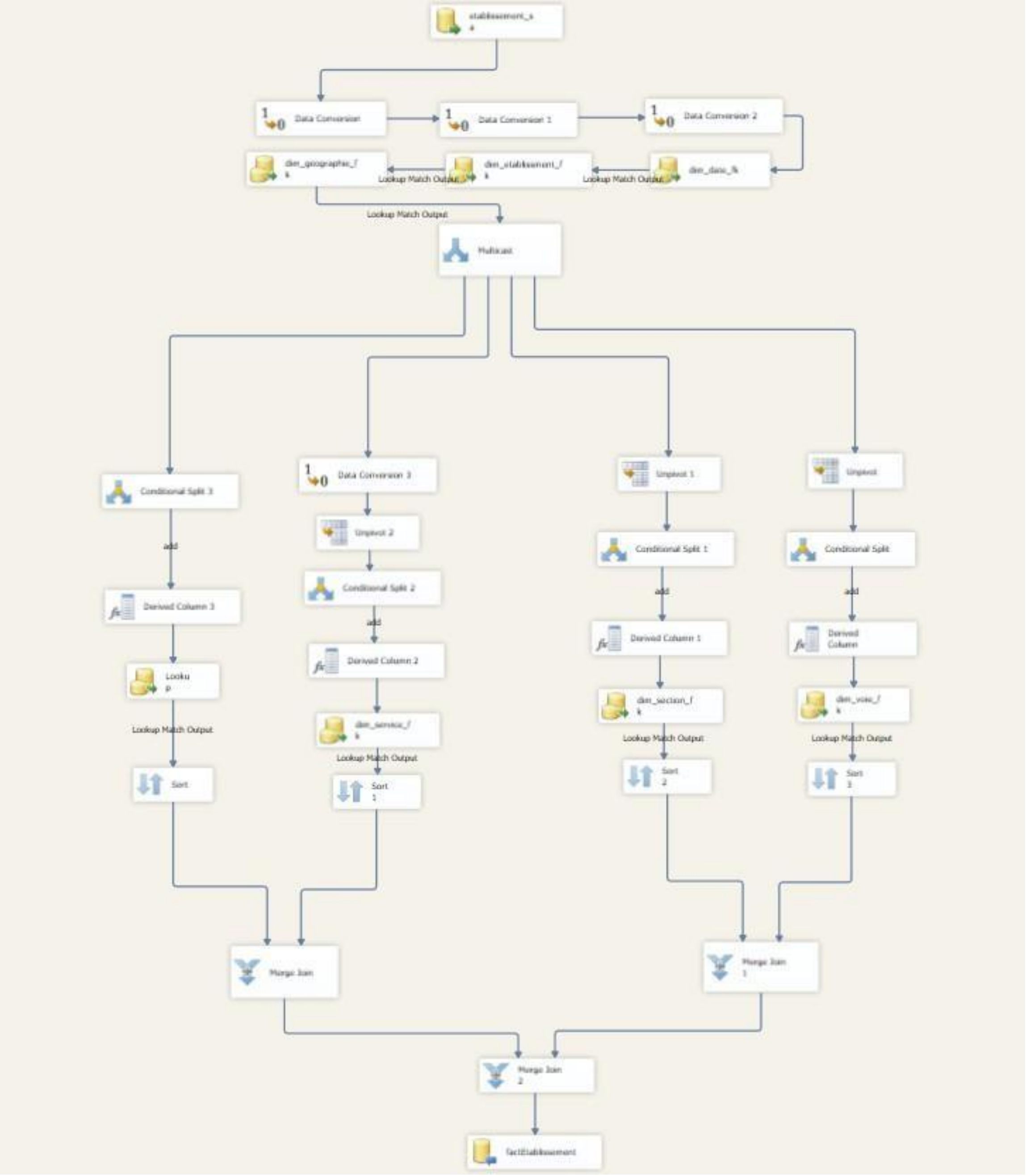


Figure 3.8: SSIS data flow for preparation and warehouse loading.

Figure 3.9 presents the star schema where factEtablissement2 centralizes the quantitative indicators and links to the descriptive dimensions (dimEtablissement, dimGeographie, dimService, dimSection, dimVoie, and dimTypeEtablissement) through foreign keys.

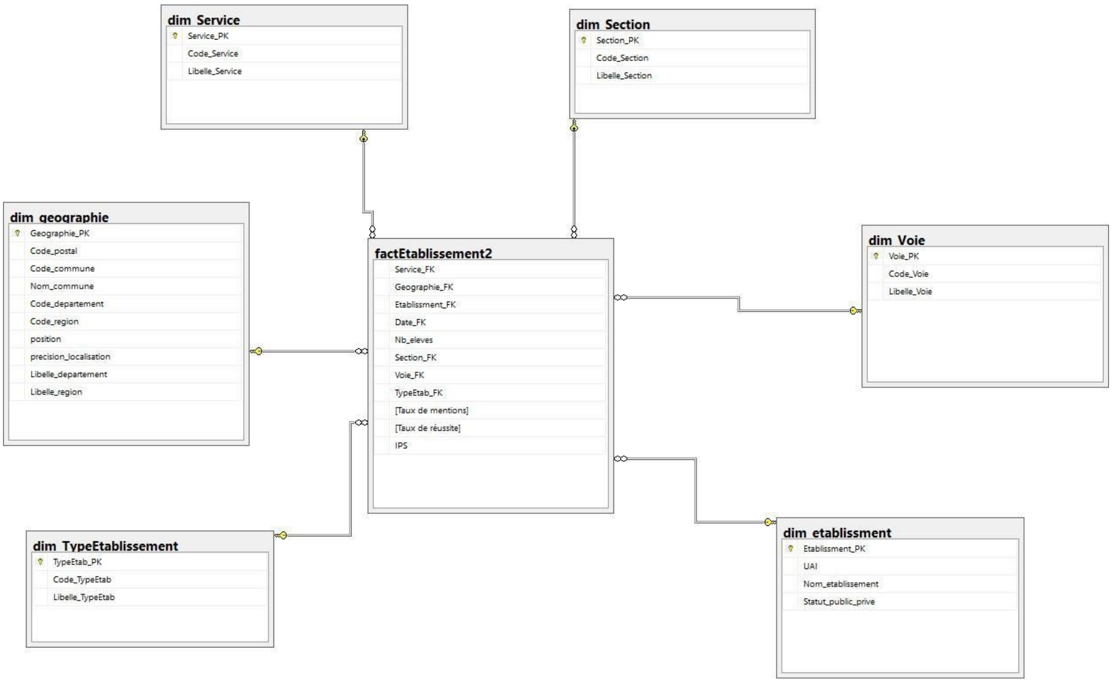


Figure 3.9: Star schema design for the BI layer.

Chapter 3 : Dashboard Development

3.1 Introduction

The dashboard development phase represents a core component of the EduSight project, as it translates processed educational data into meaningful and actionable insights. The primary objective of this phase is to provide a clear and comprehensive analytical view of the educational system through interactive and visually intuitive dashboards.

By leveraging Power BI, EduSight dashboards enable educational stakeholders to explore large volumes of data related to school structure, student enrollment, academic performance, and success indicators. The dashboards are designed to support strategic decision-making by highlighting trends, disparities, and performance gaps across regions, school statuses, and academic tracks.

All dashboards are interconnected and dynamic, allowing users to filter and analyze data from multiple perspectives. This approach ensures transparency, improves data accessibility, and facilitates evidence-based educational planning and evaluation.

3.2 Decision Maker

EduSight is designed to serve a unified decision-making profile, referred to as the **Educational Decision-Maker**. This profile includes:

- **Teachers**, who require insights into academic performance and success trends.
- **School Directors**, who need a global view of school structure, enrollment, and outcomes.

- **Ministry and Educational Authorities**, responsible for strategic planning, policy evaluation, and regional comparisons.

The dashboards aim to support this decision-maker by providing indicators related to:

— school distribution and coverage,

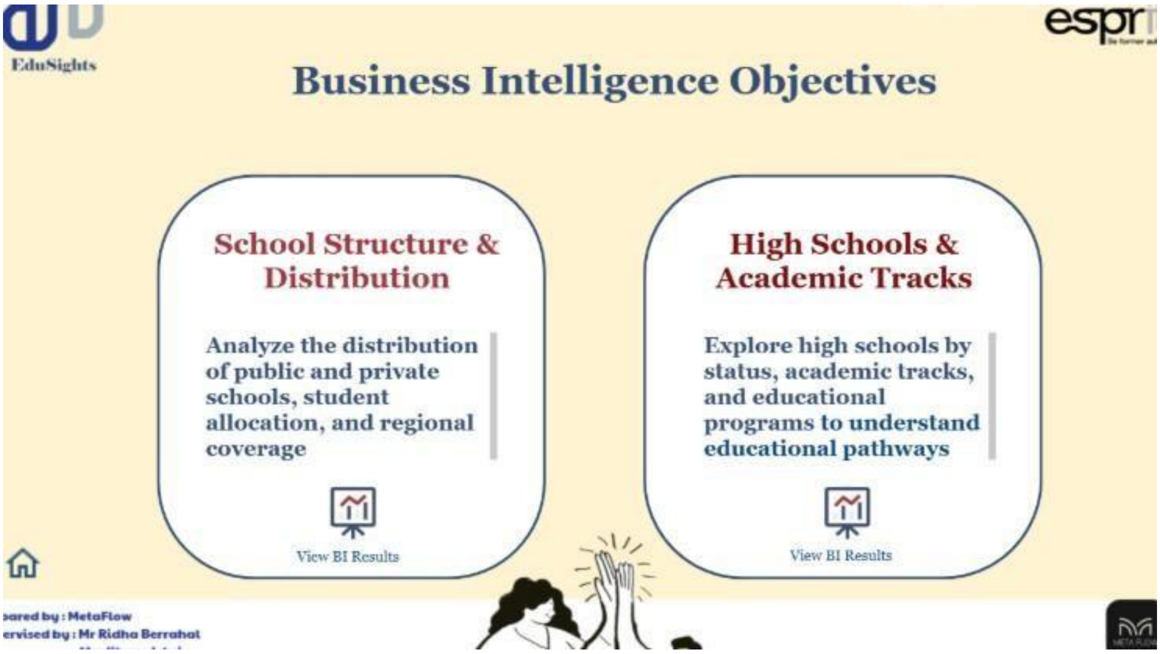
— student population analysis,

— academic success rates,

— socioeconomic impact on performance,

— long-term educational trends.

3.3 Dashboards Design



Business Intelligence Objectives

School Structure & Distribution

Analyze the distribution of public and private schools, student allocation, and regional coverage

High Schools & Academic Tracks

Explore high schools by status, academic tracks, and educational programs to understand educational pathways

Figure 4.1: Dashboard 1 – Business Intelligence Objectives (navigation page).

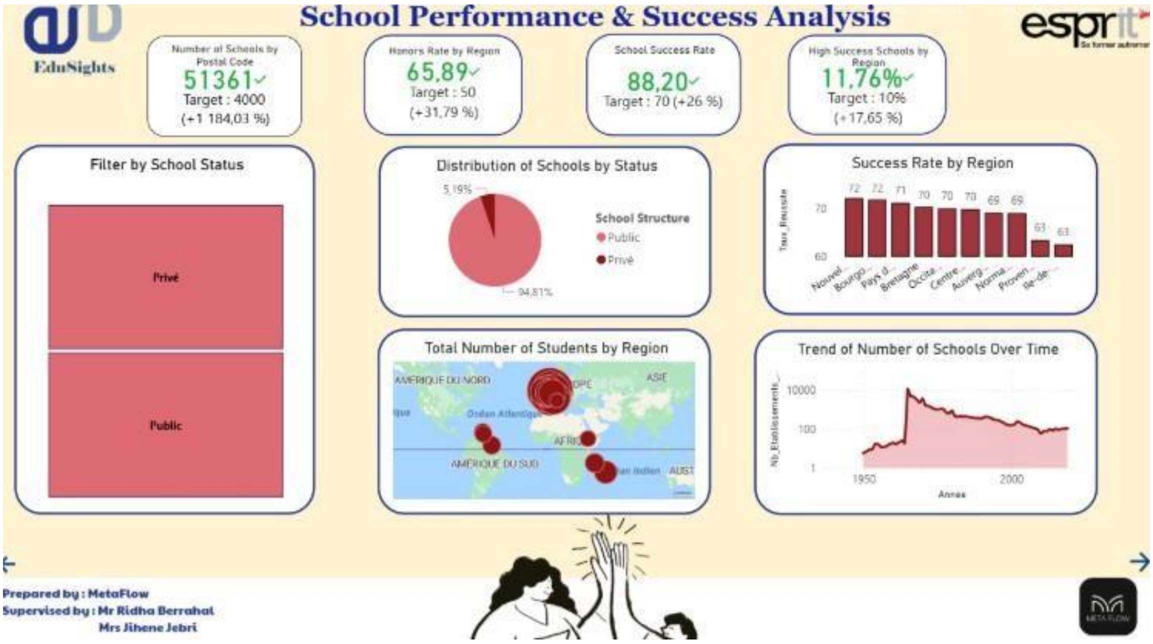


Figure 4.2: Dashboard 2 – School Performance & Success Analysis.

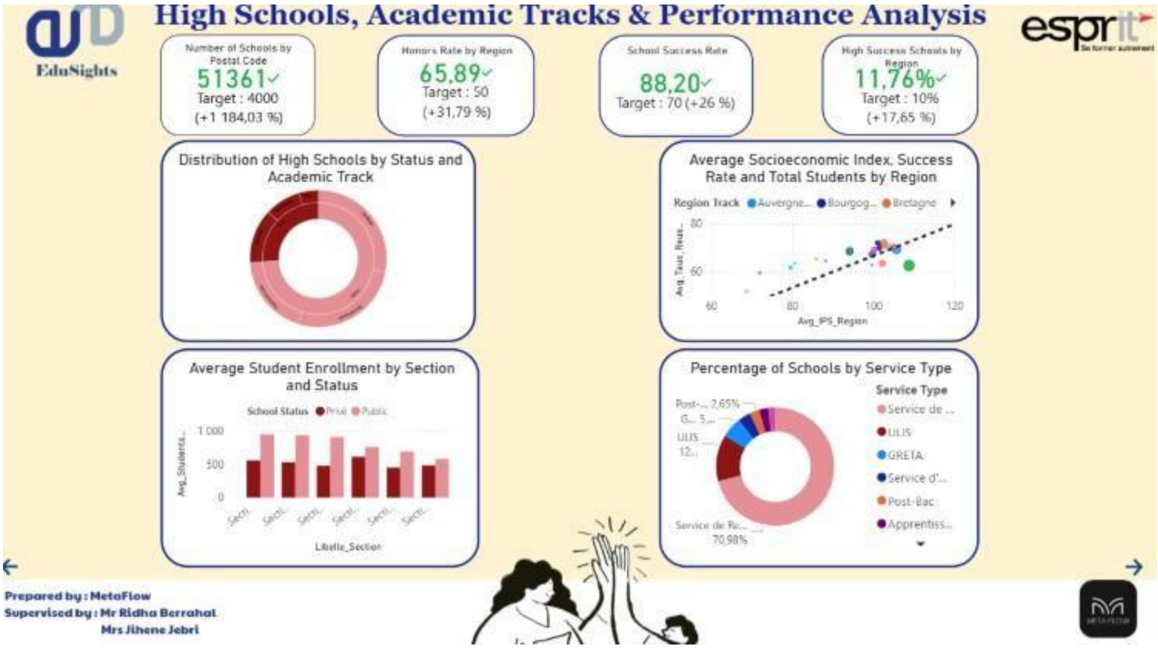


Figure 4.3: Dashboard 3 – High Schools, Academic Tracks & Performance Analysis.

3.3.1 School Structure & Distribution Dashboard

Objective

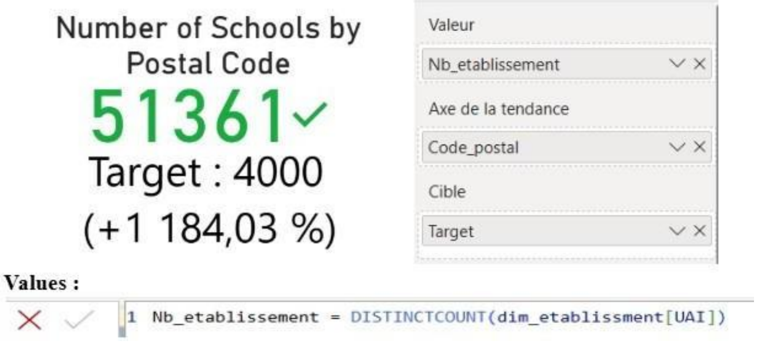
This dashboard focuses on analyzing the structural composition of the educational system. It provides insights into the distribution of public and private schools, regional coverage, and student allocation, allowing decision-makers to assess accessibility and infrastructure balance.

A. Key Performance Indicators (KPIs)

a) KPI 1: Number of Schools by Postal Code

Purpose: Measures the geographical distribution of schools across postal codes. It helps identify regions with high or low educational coverage and supports infrastructure planning and resource allocation.

Values: number of schools **Trend axis:** Code_Postal



Valeur
Nb_etablissement

Axe de la tendance
Code_postal

Cible
Target

Values :

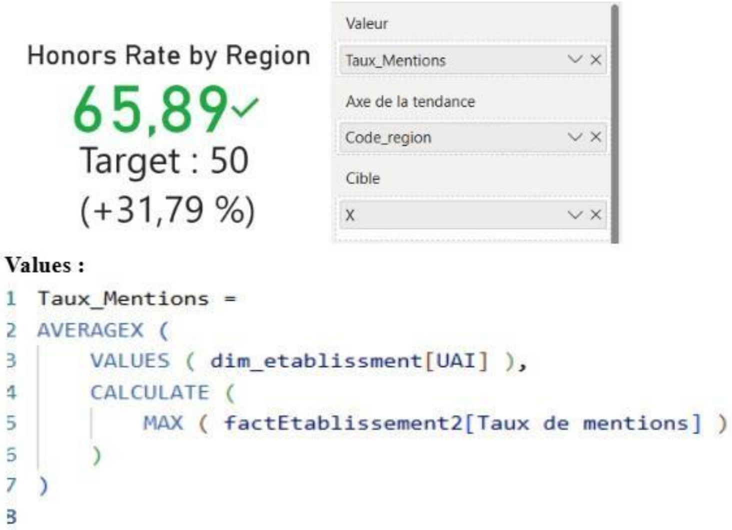
```
Nb_etablissement = DISTINCTCOUNT(dim_etablissment[UAI])
```

Figure 4.4: KPI – Number of Schools by Postal Code.

b) KPI 2: Honors Rate by Region

Purpose: Evaluates the proportion of students achieving honors across regions. It reflects academic excellence and supports performance comparison between regions.

Values: honors rate **Trend axis:** Region_Code **Target:** 50



Valeur
Taux_Mentions

Axe de la tendance
Code_region

Cible
X

Values :

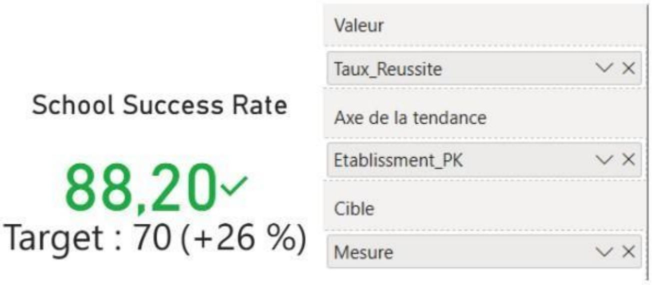
```
Taux_Mentions =
AVERAGEX (
    VALUES ( dim_etablissment[UAI] ),
    CALCULATE (
        MAX ( factEtablissement2[Taux de mentions] )
    )
)
```

Figure 4.5: KPI – Honors Rate by Region.

c) KPI 3: School Success Rate

Purpose: Represents the overall percentage of students successfully completing their academic programs. It is a key indicator for assessing educational system effectiveness.

Values: success rate

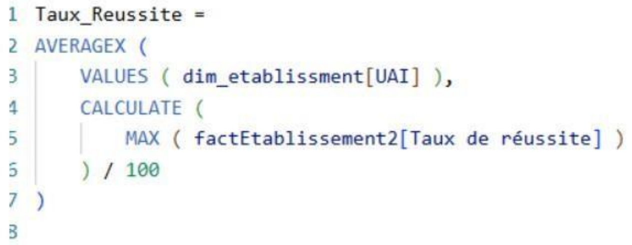


Valeur
Taux_Reussite

Axe de la tendance
Etablissment_PK

Cible
Mesure

Values :



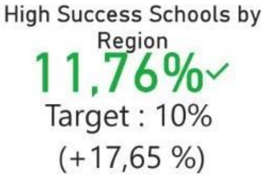
```
Taux_Reussite =
AVERAGEX (
    VALUES ( dim_etablissment[UAI] ),
    CALCULATE (
        MAX ( factEtablissement2[Taux de réussite] )
    ) / 100
)
```

Figure 4.6: KPI – Global School Success Rate.

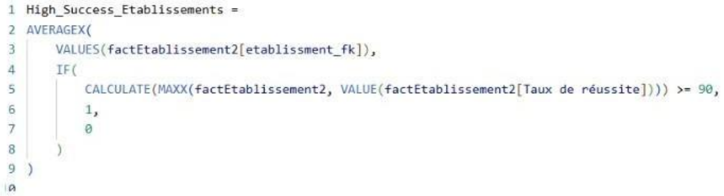
d) KPI 4: High Success Schools by Region

Purpose: Identifies the percentage of schools classified as high-performing within each region. It helps detect areas of excellence and benchmark best-performing regions.

Values: percentage of high success schools **Target:** (set according to business rule)

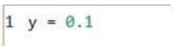


Values :



```
High_Success_Etablissements =
AVERAGEX(
    VALUES(factEtablissement2[etablissment_fk]),
    IF(
        CALCULATE(MAXX(factEtablissement2, VALUE(factEtablissement2[Taux de réussite]))) >= 90,
        1,
        0
    )
)
```

Target:

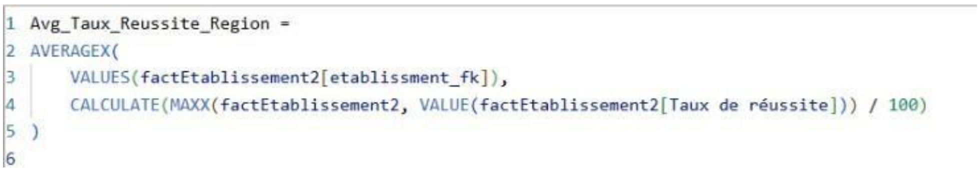


```
y = 0.1
```

Figure 4.7: KPI – High Success Schools by Region.

B. Measures

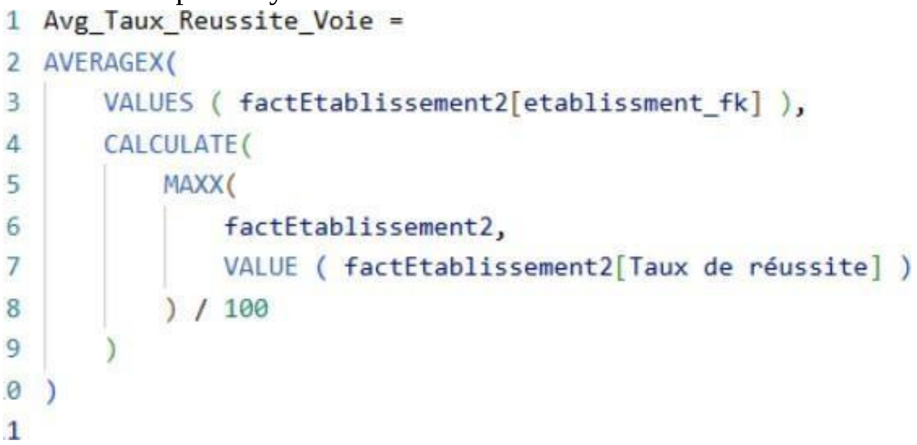
- **Average Success Rate by Region:** Measures academic achievement across regions and enables regional performance comparison.



```
Avg_Taux_Reussite_Region =
AVERAGEX(
    VALUES(factEtablissement2[etablissment_fk]),
    CALCULATE(MAXX(factEtablissement2, VALUE(factEtablissement2[Taux de réussite])) / 100)
)
```

Figure 4.8: Example measure definition – Average Success Rate by Region.

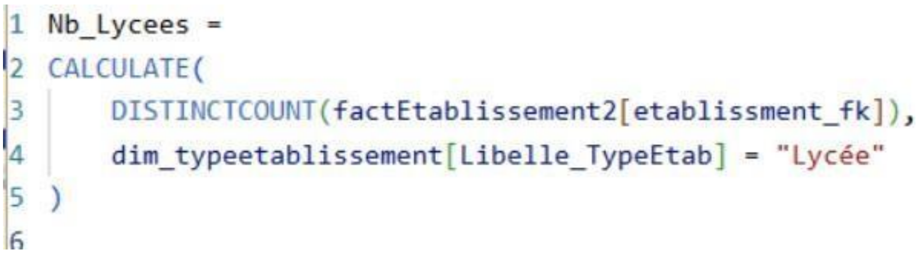
- **Average Success Rate by Academic Track:** Highlights differences in student success among the various educational pathways.



```
Avg_Taux_Reussite_Voie =
AVERAGEX(
    VALUES ( factEtablissement2[etablissment_fk] ),
    CALCULATE(
        MAXX(
            factEtablissement2,
            VALUE ( factEtablissement2[Taux de réussite] )
        ) / 100
    )
)
```

Figure 4.9: Example measure definition – Average Success Rate by Academic Track.

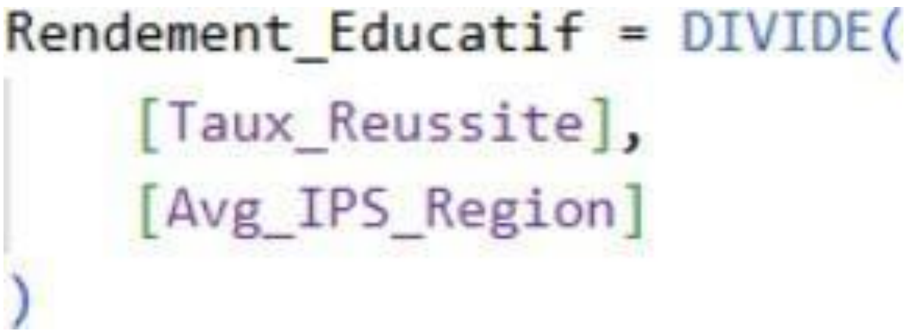
- **Number of High Schools:** Represents the total count of secondary education institutions within the educational system.



```
Nb_Lycees =
CALCULATE(
    DISTINCTCOUNT(factEtablissement2[etablissment_fk]),
    dim_typeetablissement[Libelle_TypeEtab] = "Lycée"
)
```

Figure 4.10: Example measure definition – Number of High Schools.

- **Educational Performance Index:** Evaluates educational efficiency by relating academic success to the regional socioeconomic context.

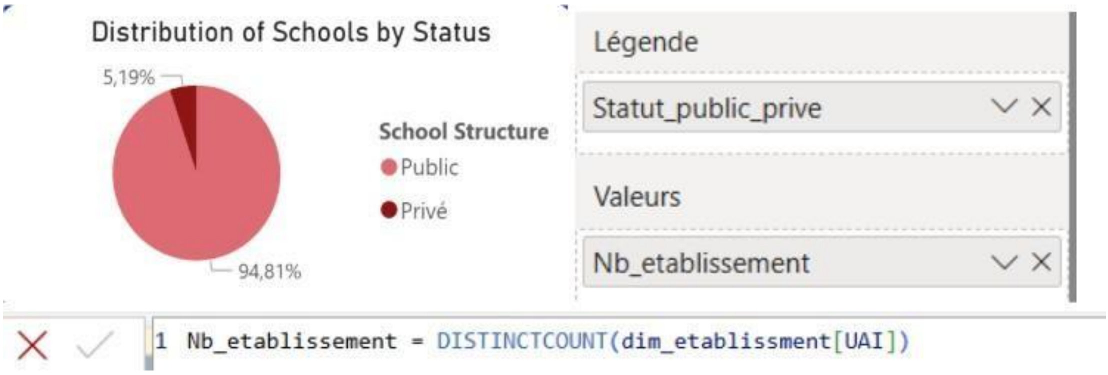


```
Rendement_Educatif = DIVIDE(
    [Taux_Reussite],
    [Avg_IPS_Region]
)
```

Figure 4.11: Example measure definition – Educational Performance Index.

C. Visuals Implemented in Power BI

- **Pie Chart – Distribution of Schools by Status:** Displays the proportion of public and private schools for a clear structural comparison.



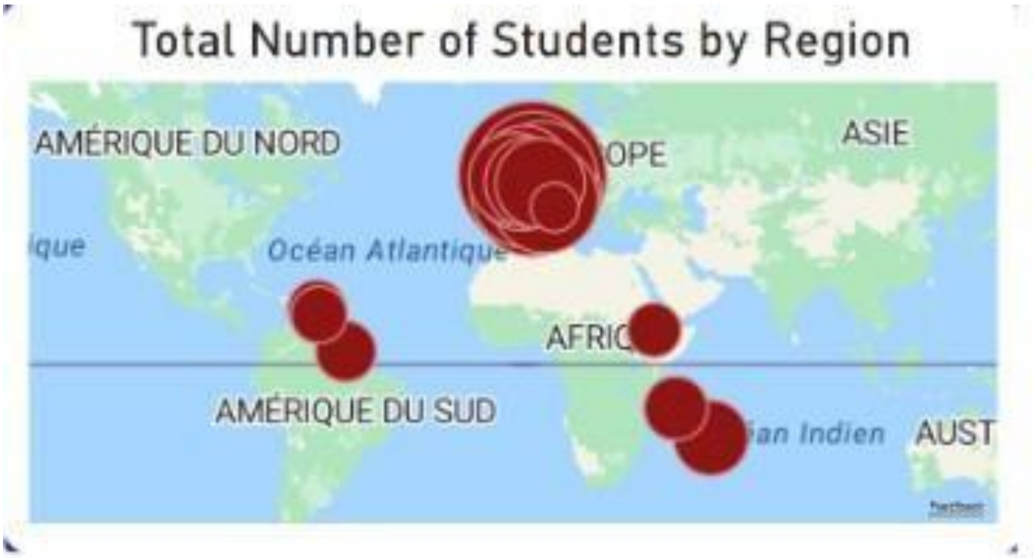
Légende
Statut_public_prive

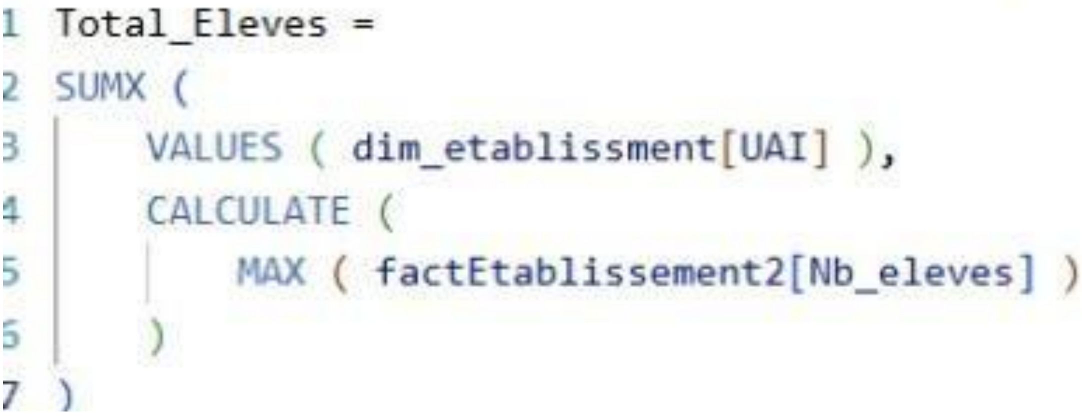
Valeurs
Nb_etablissement

```
Nb_etablissement = DISTINCTCOUNT(dim_etablissment[UAI])
```

Figure 4.12: Pie Chart – Distribution of Schools by Status.

- **Azure Maps – Total Number of Students by Region:** Visualizes student density across regions.

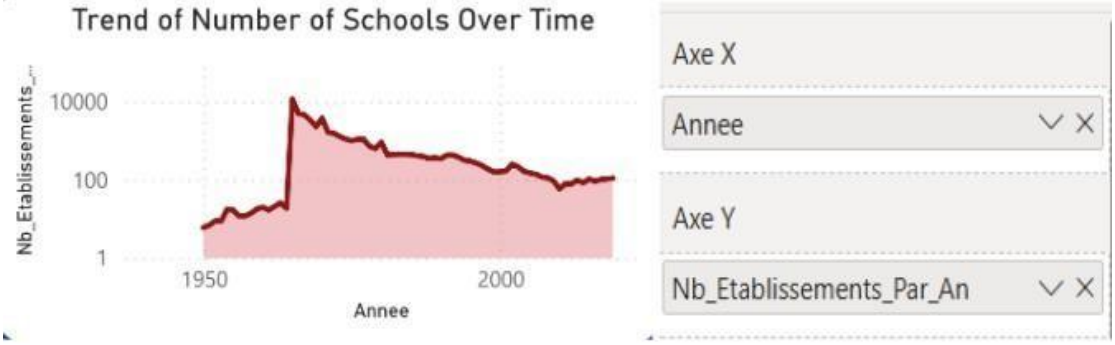




```
Total_Eleves =
SUMX (
    VALUES ( dim_etablissment[UAI] ),
    CALCULATE (
        MAX ( factEtablissement2[Nb_eleves] )
    )
)
```

Figure 4.13: Azure Maps – Total Number of Students by Region.

- **Line Chart – Trend of Number of Schools Over Time:** Illustrates the evolution of the number of schools over the years.

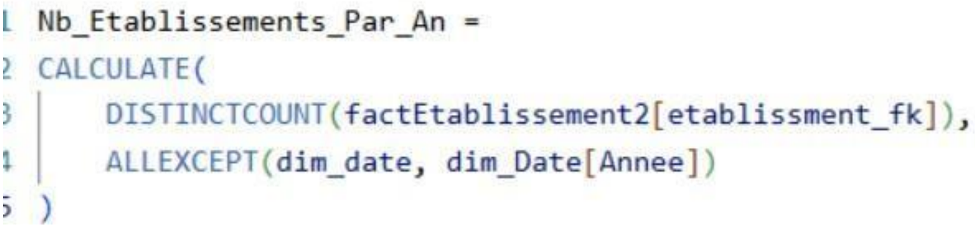


Axe X

Annee

Axe Y

Nb_Etablissements_Par_An



```
Nb_Etablissements_Par_An =
CALCULATE(
    DISTINCTCOUNT(factEtablissement2[etablissment_fk]),
    ALLEXCEPT(dim_date, dim_Date[Annee])
)
```

Figure 4.14: Line Chart – Trend of Number of Schools Over Time.

- **Stacked Histogram – Success Rate by Region:** Compares success rates across regions to identify disparities.

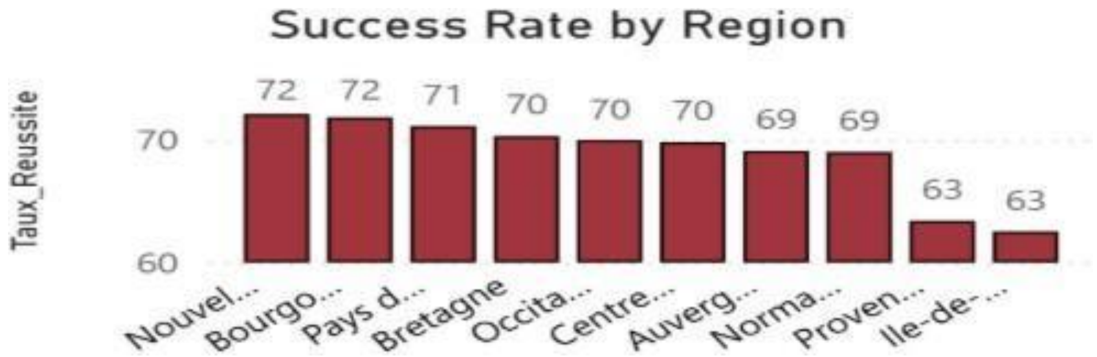


Figure 4.15: Stacked Histogram – Success Rate by Region.

D. Filters

- **Filter by School Status (Public / Private):**

Purpose: This filter allows targeted regional and structural analysis.

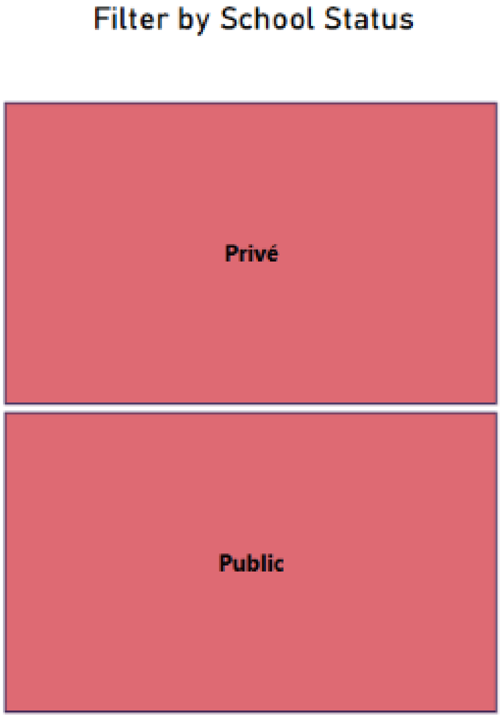


Figure 4.16: Filter – School Status (Public / Private).

E. Challenges

- Ensuring data consistency across regions and time periods.

- Maintaining clarity while visualizing large-scale geographic data.

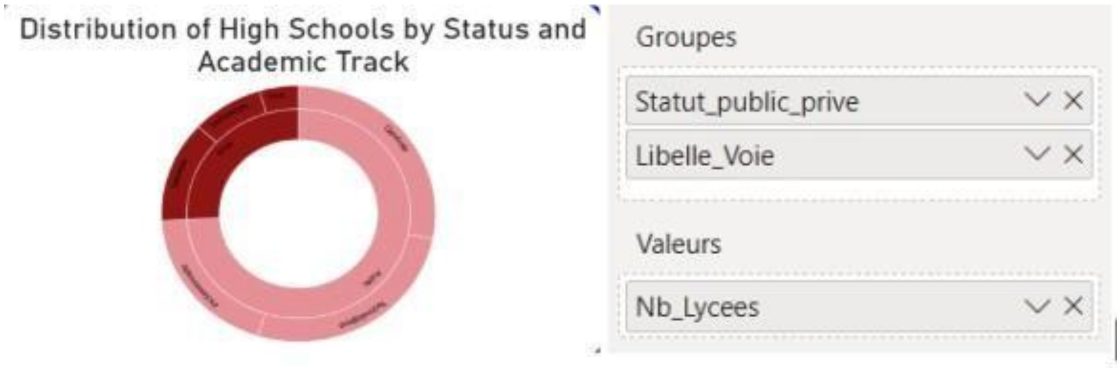
3.3.2 High Schools, Academic Tracks & Performance Dashboard

Objective

This dashboard focuses on high schools and academic pathways, analyzing how school status, academic tracks, and socioeconomic conditions influence educational outcomes.

A. Individual Dashboards in Power BI

- **Sunburst Chart – Distribution of High Schools by Status and Academic Track:** Shows the hierarchical distribution of high schools according to status and academic orientation.



Groupes

Statut_public_prive

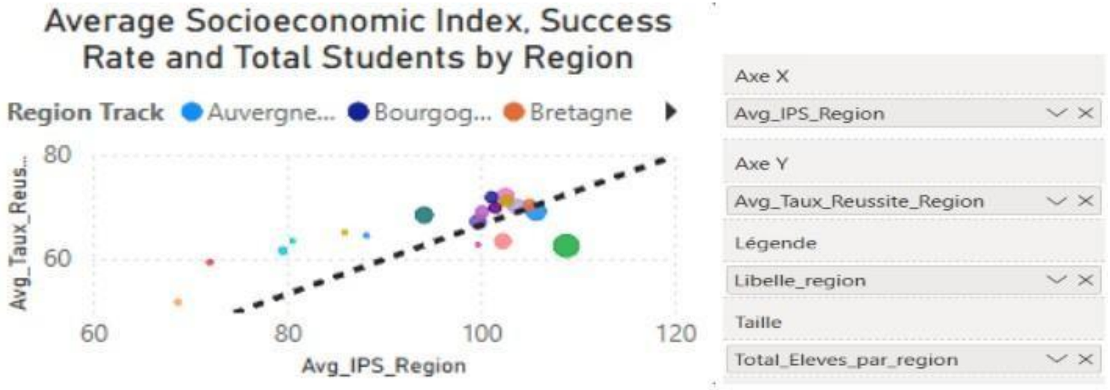
Libelle_Voie

Valeurs

Nb_Lycees

Figure 4.17: Sunburst Chart – High Schools by Status and Academic Track.

- **Scatter Plot – Socioeconomic Index vs Success Rate:** Analyzes the correlation between socioeconomic factors and academic success.



Axe X

Avg_IPS_Region

Axe Y

Avg_Taux_Reussite_Region

Légende

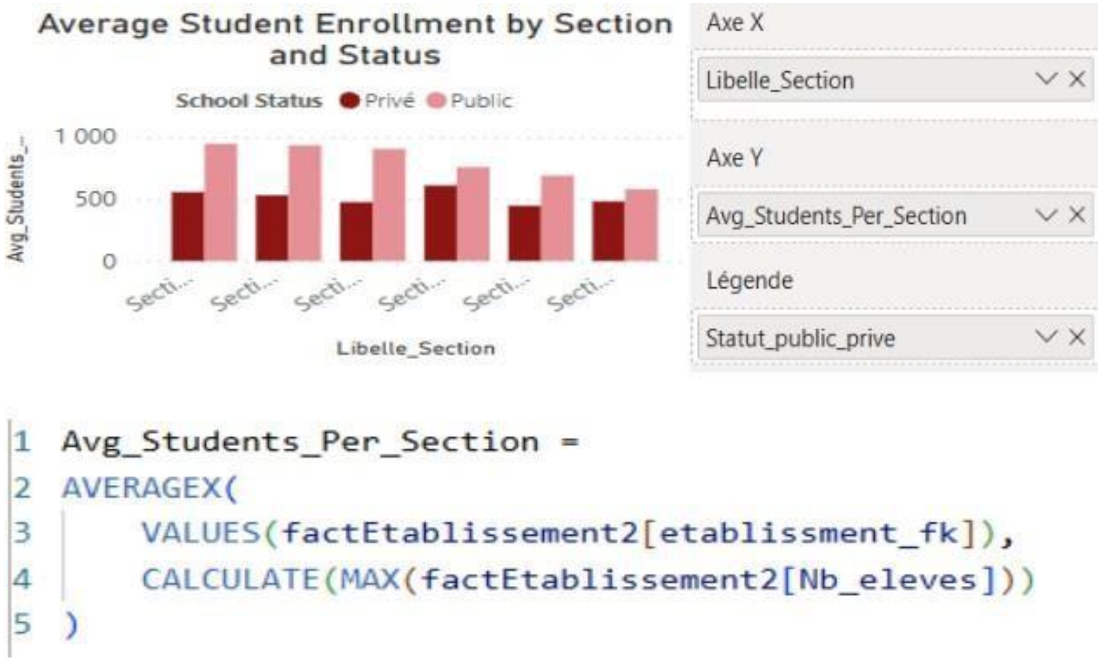
Libelle_region

Taille

Total_Eleves_par_region

Figure 4.18: Scatter Plot – Socioeconomic Index vs Success Rate.

- **Grouped Histogram – Average Student Enrollment by Section and Status:** Compares enrollment patterns between public and private institutions.



Axe X

Libelle_Section

Axe Y

Avg_Students_Per_Section

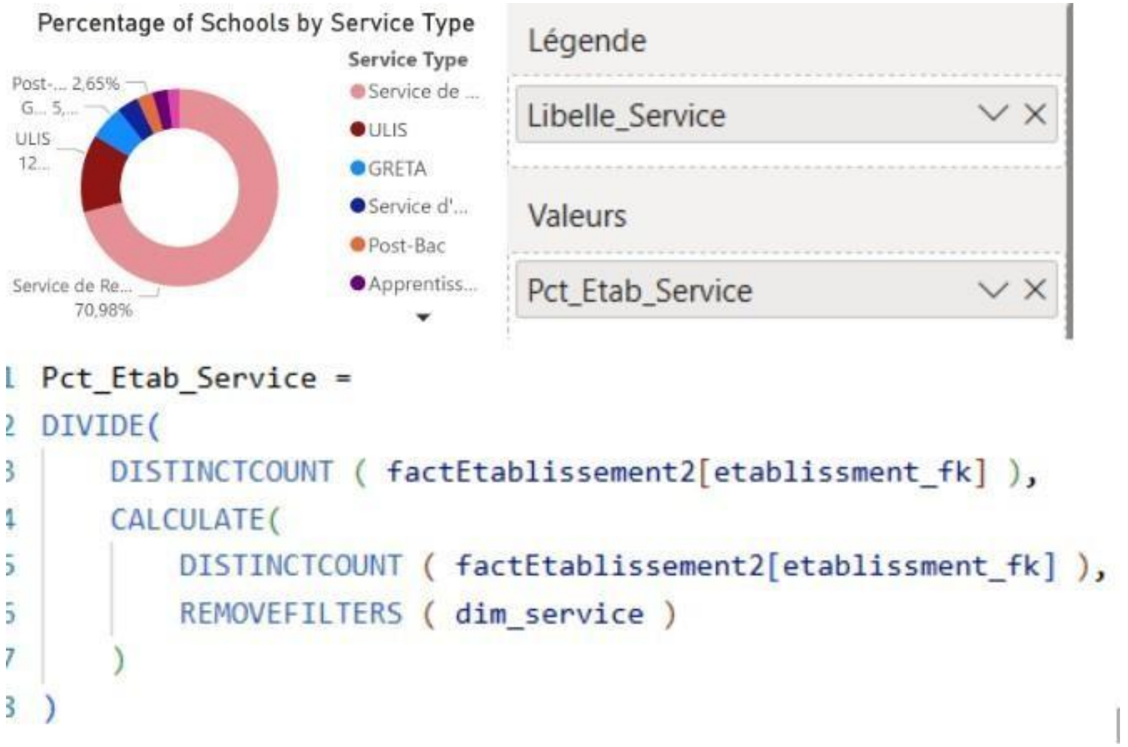
Légende

Statut_public_prive

```
Avg_Students_Per_Section =
AVERAGEX(
    VALUES(factEtablissement2[etablissment_fk]),
    CALCULATE(MAX(factEtablissement2[Nb_eleves]))
)
```

Figure 4.19: Grouped Histogram – Average Student Enrollment by Section and Status.

- **Donut Chart – Percentage of Schools by Service Type:** Displays the distribution of educational services and programs.



Légende

Libelle_Service

Valeurs

Pct_Etab_Service

```
Pct_Etab_Service =
DIVIDE(
    DISTINCTCOUNT ( factEtablissement2[etablissment_fk] ),
    CALCULATE(
        DISTINCTCOUNT ( factEtablissement2[etablissment_fk] ),
        REMOVEFILTERS ( dim_service )
    )
)
```

Figure 4.20: Donut Chart – Percentage of Schools by Service Type.

B. Challenges

- Capturing the complex relationship between socioeconomic context and academic success.
- Avoiding misinterpretation of correlations as causation.

Chapter 4 : Modeling and Evaluation

4.1 Introduction

This chapter presents the machine learning component of the EduSight project. It covers the supervised learning models developed for classification and regression tasks, the unsupervised clustering approach used to identify homogeneous school profiles and the integration of trained models into the Power BI environment to enable predictive and decision-oriented analytics.

4.2 Classification Models

Classification is a supervised machine learning approach that aims to assign each observation to one category among a predefined set of classes. It learns decision rules from labeled historical data in order to predict the class of new, unseen instances based on their input characteristics. Classification problems are commonly divided into *binary classification* (two classes) and *multiclass classification* (more than two classes) as illustrated in Figure 5.1 [5].

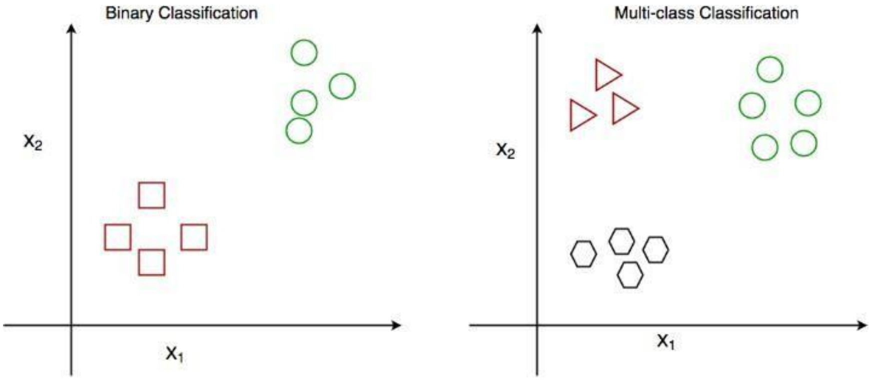


Figure 5.1: Binary vs. multiclass classification illustration.

In our project, classification is used to automatically categorize educational institutions based on structured indicators and administrative attributes. This supports analysis and decision-making by enabling clear segmentation of institutions, facilitating comparisons between groups, and identifying patterns associated with specific categories. In practice, the developed models aim to predict key labels such as an institution's administrative status (e.g., public vs. private), educational level (e.g., primary, middle, or high school), and an enrollment size category (high vs. low) derived from the number of students. Together, these predictions provide actionable insights that can support planning processes and inform strategic policy decisions.

4.2.1 Classification Model 1: Prediction of Educational Institution Status

The objective of this model is to predict the administrative status of an educational institution (Public or Private), formulated as a binary classification task. This supports institutional profiling and enables quantitative analysis of public and private education distribution across geographic and structural contexts.

Target variable (y)

The target variable is Statut_Etablissement. The task is formulated as a binary classification problem with two classes: Public and Private.

Input features (X) and data representation

The input feature set X is built from the data warehouse by joining the fact table with multiple dimensions (e.g., geography, institution, date, section, track/voie, service). The selected variables include:

- **Numerical features:** Nb_eleves, Code_Postal, Annee, Mois, Jour.
- **Categorical features:** region, section codes, voie codes, service codes, and other categorical descriptors retained after removing non-informative fields.

To reduce noise and avoid high-cardinality identifiers, several columns were removed (e.g., institution name, UAI identifier, city name, and descriptive labels).

Feature engineering and preprocessing

To ensure a consistent and reproducible preprocessing workflow for Model 1, we applied a three-step pipeline combining data cleaning, class balancing, and feature transformation, summarized in Figure 5.2.

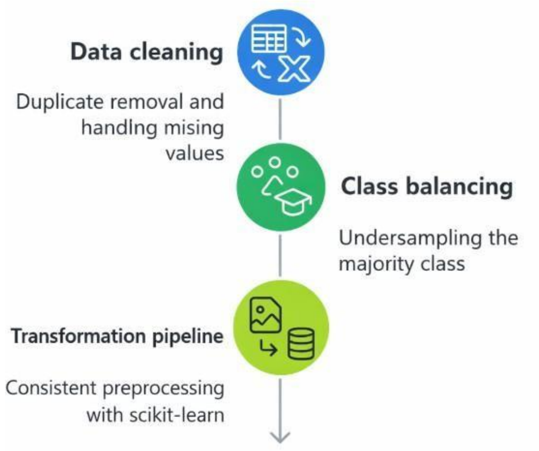

Data cleaning
Duplicate removal and handlng mising values
Class balancing
Undersampling the majority class
Transformation pipeline
Consistent preprocessing with scikit-learn

Figure 5.2: Preprocessing workflow for Model 1 (cleaning, balancing, transformation).

Data cleaning included duplicate removal and missing-value handling by replacing placeholder NULL values; categorical missing values were filled with Unknown, while numerical values were imputed using robust statistics (median for Nb_eleves and mode for Code_Postal). Since the dataset was imbalanced (Public > Private), we performed undersampling of the majority class. Finally, preprocessing was implemented using a scikit-learn pipeline: numerical features were imputed and standardized with StandardScaler, while categorical features were imputed and encoded using OneHotEncoder(handle_unknown='ignore').

Model choice and configuration

A K-Nearest Neighbors (KNN) classifier was retained for this task as an interpretable baseline for binary classification. Given that the input space contains a mix of scaled numerical features and encoded categorical variables, KNN is well suited to capture local similarity patterns between institutions under the assumption that institutions with similar profiles are likely to share the same administrative status.

The final KNN model was configured with $k = 7$ nearest neighbors using distance-weighted voting so that closer neighbors have a stronger influence on the prediction, and Manhattan distance ($p = 1$) as the similarity metric. After class balancing, the dataset was divided into training and test sets using a stratified 80/20 split to preserve class proportions (test_size=0.2, random_state=42, stratify=y).

Evaluation

The classification performance of the KNN model on the held-out test set is summarized in Table 5.1 using accuracy and per-class precision, recall, and F1-score.

Table 5.1: Model 1 (KNN) – Status prediction performance.

Class	Precision	Recall	F1-score
Private	0.91	0.98	0.94
Public	0.98	0.90	0.94
Accuracy: 0.94			

The model achieves an overall accuracy of 0.94 with balanced F1-scores across both classes (0.94). Recall is higher for the Private class (0.98), indicating that most private institutions are correctly identified, while its lower precision (0.91) suggests some public institutions are predicted as private. Conversely, the Public class shows high precision (0.98) but lower recall (0.90), meaning some public institutions are misclassified as private.

4.2.2 Classification Model 2: Prediction of Educational Institution Type/Level

The objective of this model is to predict the institution type/level represented by Code_TypeEtab (e.g., school categories such as primary/middle/high depending on the dataset coding). This task supports institutional segmentation and planning by enabling the automatic categorization of institutions based on measurable indicators and structural characteristics.

Target variable (y)

The target variable is Code_TypeEtab. The task is formulated as a multiclass classification problem where each institution is assigned to one class among multiple possible categories.

Input features (X) and data representation

The input feature set X combines educational performance indicators and institutional attributes including:

— **Numerical indicators:** IPS, Taux_Reussite, Taux_Mentions, Nb_eleves (and other available quantitative measures).

— **Categorical/contextual variables:** region and structural descriptors related to sections, services, and tracks (voie), depending on availability.

To ensure a consistent representation, non-informative identifiers and purely descriptive fields were excluded from the modeling dataset.

Feature engineering and preprocessing

This task required building a consolidated representation per institution. The main preprocessing steps are summarized in Figure 5.3.

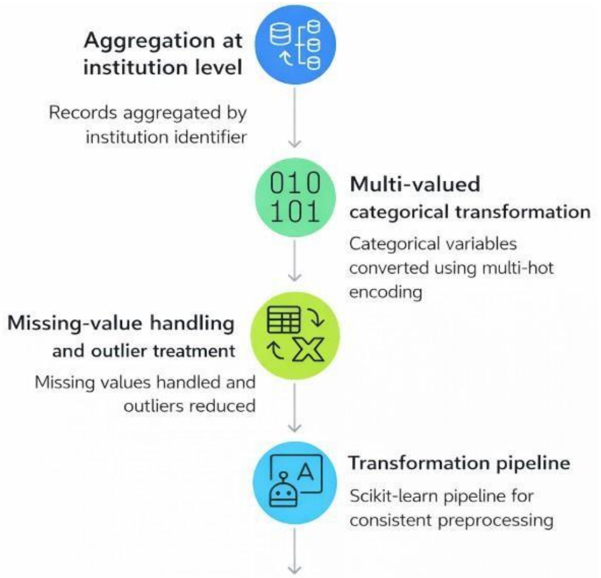

Aggregation at institution level
Records aggregated by institution identifier
010
101
Multi-valued categorical transformation
Categorical variables converted using multi-hot encoding
Missing-value handling and outlier treatment
Missing values handled and outliers reduced
A
Transformation pipeline
Scikit-learn pipeline for consistent preprocessing

Figure 5.3: Preprocessing workflow for Model 2 (aggregation, multi-hot encoding, scaling/encoding).

Records were aggregated at the institution level (e.g., by UAI) to produce one row per establishment using robust summary statistics such as the median for key numerical indicators. Multi-valued categorical attributes (e.g., section/service/track codes) were transformed into a multi-hot representation using MultiLabelBinarizer, generating binary features (e.g., sec_*, srv_*, voie_*). Missing values were handled prior to training and outliers in selected numerical variables were reduced using an IQR-based strategy. Finally, the resulting dataset was processed through a scikit-learn pipeline to ensure consistent preprocessing, including scaling for numerical features and encoding for categorical variables before training the SVC classifier.

Model choice and configuration

We selected a Support Vector Classifier (SVC) for the institution type/level prediction task due to its effectiveness in multiclass classification and its ability to handle high-dimensional feature spaces created by multi-hot/one-hot expansion of categorical attributes. To mitigate potential class imbalance, the model was trained with class_weight='balanced'. The final configuration used a linear kernel with $C = 10$, providing a robust linear decision boundary while remaining computationally efficient. Training was performed using a stratified train/test split to preserve class distributions.

Evaluation

The classification performance of the SVC model on the held-out test set is summarized in Table 5.2 using overall accuracy and per-class precision, recall, and F1-score.

Table 5.2: Model 2 (SVC) – Type/level prediction performance.

Class	Precision	Recall	F1-score
C	0.96	0.97	0.96
E	1.00	1.00	1.00
L	0.94	0.95	0.95
Accuracy: 0.99			

The model achieves an overall accuracy of 0.99 with consistently strong performance across classes. Class E is predicted perfectly (precision/recall/F1 = 1.00), while classes C and L also show high and balanced scores, indicating that the classifier separates institution categories effectively with limited class-specific degradation. The close alignment between precision and recall for each class suggests stable decision boundaries and a low rate of systematic confusion between categories.

4.2.3 Classification Model 3: Prediction of Student Enrollment Demand

Logistic Regression is a supervised classification algorithm, despite its name: it models the probability of belonging to a class by applying a logistic (sigmoid) function to a linear combination of input features [6].

In our project, it is used for a binary classification task to predict the enrollment size category of an establishment. More specifically, it estimates whether a school belongs to the high-enrollment or low-enrollment group based on its academic and contextual profile. This segmentation supports institutional profiling and enables data-driven planning and comparative analysis across establishments.

Target variable (y)

The target variable is Nb_eleves_class, constructed from Nb_eleves using a median-based threshold:

$$\text{Nb\_eleves\_class} = \begin{cases} 1 & \text{if Nb\_eleves} \geq \text{median(Nb\_eleves)} \ \ \text{(high-enrollment)} \\ 0 & \text{otherwise (low-enrollment)} \end{cases}$$

This produces a nearly balanced problem (Class 1: 25,875; Class 0: 25,486).

Input features (X) and data representation

After aggregation to the establishment level, the final representation combines:

- **Numerical features:** IPS, Taux_Reussite, Taux_Mentions, Annee.
- **Categorical features:** region, Code_TypeEtab, Statut_Etablissement.
- **Multi-valued attributes:** Code_Section, Code_Service, Code_Voie, expanded into multi-hot features (sec_*, srv_*, voie_*).

High-cardinality descriptors such as establishment name and city-level identifiers were not retained in the final feature matrix, and UAI was used only for aggregation.

Feature engineering and preprocessing

Feature construction and preprocessing were implemented through a standardized workflow, summarized in Figure 5.4.

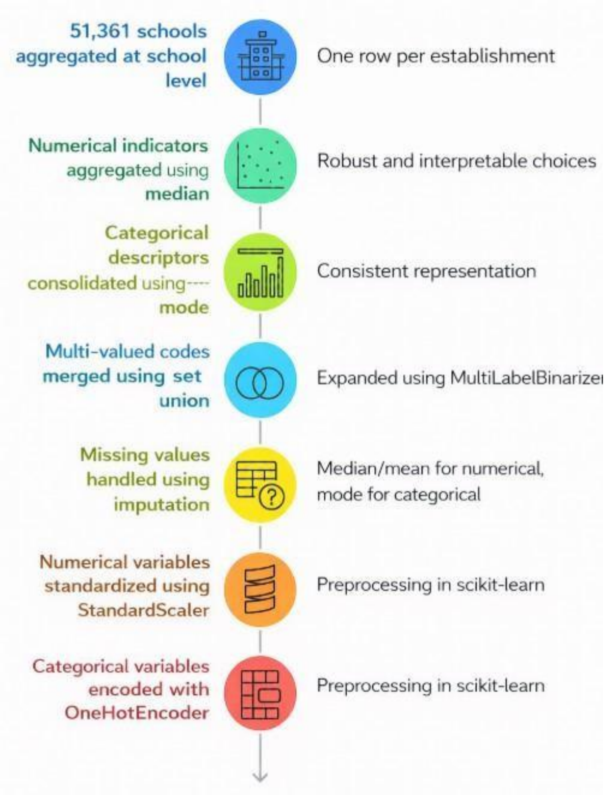

51,361 schools aggregated at school level
One row per establishment
Numerical indicators aggregated using median
Robust and interpretable choices
Categorical descriptors consolidated using---- mode
Consistent representation
Multi-valued codes merged using set union
Expanded using MultiLabelBinarize
Missing values handled using imputation
Median/mean for numerical, mode for categorical
Numerical variables standardized using StandardScaler
Preprocessing in scikit-learn
Categorical variables encoded with OneHotEncoder
Preprocessing in scikit-learn

Figure 5.4: Preprocessing workflow for Model 3 (aggregation, multi-hot encoding, imputation, scaling/encoding).

Records were aggregated at the UAI (establishment) level to obtain one row per school using robust aggregation rules (median for numerical indicators and mode for categorical descriptors). Multi-valued attributes were consolidated per school and expanded into multi-hot representations using MultiLabelBinarizer. Missing values were imputed using median/mean for numerical variables and mode for categorical variables. Finally, the modeling matrix was prepared using a scikit-learn pipeline with StandardScaler for numerical features and OneHotEncoder(handle_unknown='ignore') for categorical features.

Model choice and configuration

A Logistic Regression classifier was selected as an interpretable baseline for binary classification. It is well suited to high-dimensional encoded feature spaces resulting from one-hot and multi-hot transformations and provides probabilistic outputs that can be thresholded when required. The model was trained within the preprocessing pipeline using LogisticRegression(max_iter=1000). The dataset was partitioned into training and test sets using a stratified 80/20 split (test_size=0.2, random_state=42, stratify=y) topreserve class proportions.

Evaluation

Performance was evaluated on a held-out test set using accuracy and per-class precision, recall, and F1-score (Table 5.3).

Table 5.3: Model 3 (Logistic Regression) – Enrollment category prediction performance.

Class	Precision	Recall	F1-score
0 (low-enrollment)	0.72	0.85	0.78
1 (high-enrollment)	0.82	0.68	0.74
Accuracy: 0.7647			

The model achieves an overall accuracy of 0.7647 with relatively balanced performance across classes. Recall is higher for the low-enrollment class (0.85), indicating that most low-enrollment establishments are correctly identified, while the high-enrollment class shows higher precision (0.82) but lower recall (0.68), meaning that a portion of high-enrollment establishments are misclassified as low-enrollment.

To complement threshold-dependent metrics, the ROC curve in Figure 5.5 evaluates the model's discriminative ability across all classification thresholds.

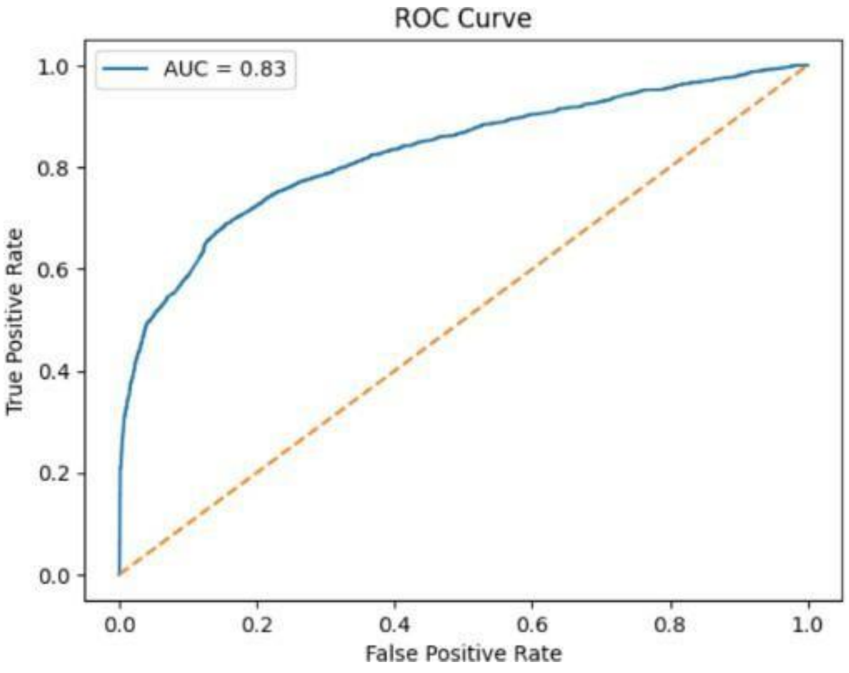


Figure 5.5: ROC curve for Model 3 (Logistic Regression).

As shown in Figure 5.5, the ROC curve remains well above the random baseline with an AUC ≈ 0.83, confirming good separation between the two enrollment categories and consistent classification quality beyond the default decision threshold.

4.3 Regression Models

Regression is a supervised learning technique used to predict continuous numerical values by learning relationships between input variables (features) and an output variable (target). It helps understand how changes in one or more factors influence a measurable outcome and is widely used in forecasting, risk analysis, decision-making, and trend estimation.

In this project, regression is applied to support the objective of improving school quality and performance by predicting continuous academic outcome indicators at the school level. Using a structured set of institutional and contextual features, the model learns to estimate quantitative metrics such as success rates and academic excellence indicators. These predictions provide a data-driven baseline for performance benchmarking, help identify schools with atypical outcomes, and enable targeted improvement actions through scalable, reproducible machine learning pipelines.

4.3.1 Regression Model 1: Prediction of Mention Rate (Taux_Mentions)

This regression model aims to estimate school-level academic excellence by predicting the continuous indicator Taux_Mentions. Rather than describing past performance only, the model learns patterns linking institutional and contextual characteristics to academic distinction outcomes [7].

Data preparation and feature construction

Numerical indicators (e.g., IPS, Taux_Reussite) were aggregated using the median to reduce sensitivity to extreme values, while Nb_eleves was retained as a capacity/context signal. Categorical descriptors (Region, Code_TypeEtab, Statut_Etablissement) were consolidated using the mode and multi-valued attributes (Code_Section, Code_Service, Code_Voie) were merged via set union to preserve the full set of offerings per establishment. The dataset was aggregated at the establishment level using UAI as the unique identifier, producing one row per school aligned with the school-level target. Missing values were handled using median imputation for numerical fields and the most frequent category or Unknown for categorical fields. Outliers were mitigated using an IQR-based filtering rule on selected numerical variables (notably IPS and Nb_eleves). Finally, numerical variables were standardized using StandardScaler, and categorical variables were encoded using OneHotEncoder(handle_unknown='ignore') within a preprocessing pipeline.

Model choice and configuration

This model was implemented using an XGBoost Regressor selected for its strong performance on tabular heterogeneous datasets and its ability to capture nonlinear relationships and feature interactions. Hyperparameters were optimized using cross-validation (via GridSearchCV). The final configuration was: colsample_bytree=0.7, subsample=0.7, learning_rate=0.01, max_depth=5, n_estimators=500, with L1 regularization (reg_alpha=1.0) and L2 regularization (reg_lambda=1.0).

Evaluation

Model performance was evaluated on a held-out test set using standard regression metrics (Table 5.4).

Table 5.4: Regression Model 1 (XGBoost) – Mention rate prediction performance.

Metric	Value
R^2	0.7185
MSE	9.6766
RMSE	3.1107
MAE	0.6481

On the held-out test set, the model achieves $R^2 = 0.7185$, meaning it explains about 72% ofthe variability in Taux_Mentions. Error magnitude remains controlled with RMSE = 3.1107 and MAE = 0.6481, while the gap between RMSE and MAE suggests a small number of cases contribute larger errors.

To complement the numerical evaluation, Figure 5.6 visualizes the relationship between predicted and actual values through a predicted vs. actual scatter plot.

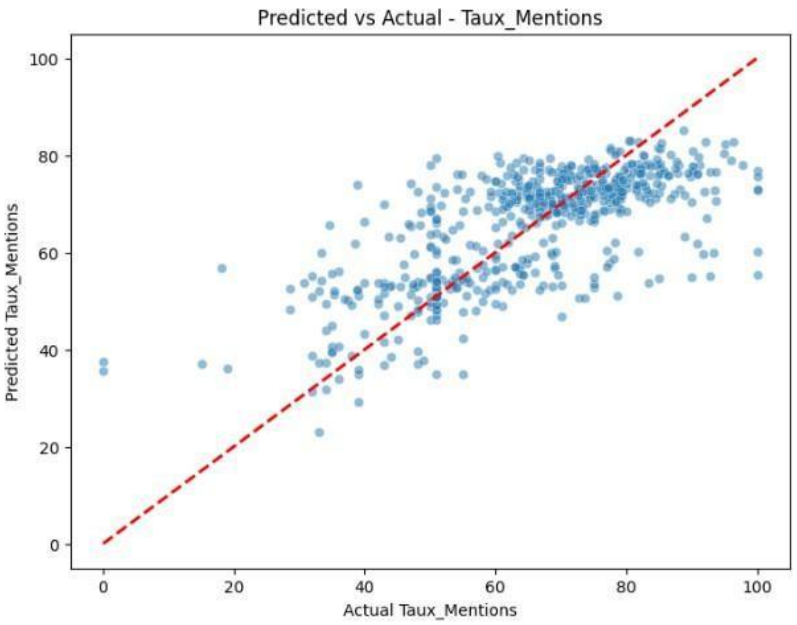


Figure 5.6: Predicted vs. actual values for Taux_Mentions.

Most points lie close to the diagonal ($y = x$), indicating strong agreement between predicted ad actual mention rates for the majority of establishments. The spread increases at more extreme values, suggesting that very low or very high mention rates are slightly harder to predict.

4.3.2 Regression Model 2: Prediction of Success Rate (Taux_Reussite)

This regression model aims to predict the continuous academic outcome Taux_Reussite at the establishment level by learning the relationship between institutional/contextual factors and observed success outcomes.

Data preparation and feature engineering

Missing values were handled using median imputation for numerical variables and by assigning a dedicated Missing category for categorical features. Date-related variables (Annee, Mois, Jour) were transformed into engineered temporal features (day_of_week, day_of_year, quarter). Extremely high-cardinality identifiers (Ville, UAI, Nom_Etablissement) were removed to control dimensionality. After an 80/20 train-test split, key numerical features were standardized using StandardScaler.

Model choice and configuration

An XGBoost Regressor was retained due to its strong performance on structured tabular data and its ability to capture nonlinear patterns and feature interactions across mixed feature types. Hyperparameters were optimized using RandomizedSearchCV with 3-fold cross-validation and RMSE as the optimization objective. The final configuration was: subsample=0.8, colsample_bytree=0.8, learning_rate=0.2, max_depth=5, and n_estimators=400.

Evaluation

Model performance was evaluated on a held-out test set using standard regression metrics (Table 5.5).

Table 5.5: Regression Model 2 (XGBoost) – Success rate prediction performance.

Metric	Value
R^2	0.9987
RMSE	0.1600
MAE	0.0065

The tuned model achieves an R^2 of 0.9987, indicating an excellent fit to the observed Taux_Reussite values on the test set. Prediction errors remain very low (MAE = 0.0065, RMSE = 0.1600), suggesting that predicted success rates closely align with the ground truth for the vast majority of establishments.

To complement the numerical metrics, the residual plot in Figure 5.7 is used to verify that errors are centered around zero and to detect any systematic bias.

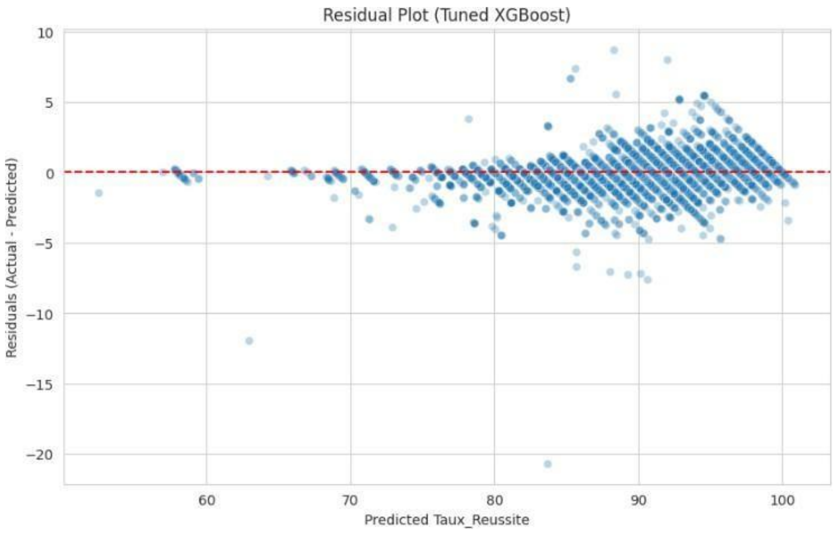


Figure 5.7: Residual plot for Taux_Reussite predictions.

Most errors stay close to 0, indicating no clear tendency to systematically overpredict or underpredict. The error variance increases slightly for very high predicted success rates, indicating less consistent predictions for top-performing schools.

4.4 Clustering Model

Clustering is an unsupervised learning approach that groups observations into clusters based on similarity in their feature space, without relying on labeled targets. The objective is to obtain clusters that are internally coherent (high similarity within a cluster) and well separated from one another, thereby revealing latent structures and patterns in the data [8].

In this project, clustering was introduced to complement the supervised models by providing an unsupervised segmentation of establishments. Rather than predicting an outcome, it groups schools into homogeneous profiles based on contextual and academic indicators (e.g., IPS, success/mention rates, size, administrative status/type, and offered tracks). This segmentation enables meaningful peer benchmarking, helps identify distinct school archetypes, and supports targeted decision-making.

Clustering process

The clustering pipeline was designed to generate a consistent representation per establishment while handling mixed feature types. The dataset was aggregated at the establishment level using UAI, producing one row per school. Numerical indicators (IPS, Taux_Reussite, Taux_Mentions) were aggregated using the median; Nb_eleves was retained as a size signal; categorical descriptors were consolidated using the most frequent value. Multi-valued codes (Code_Section, Code_Service, Code_Voie) were converted into multi-hot vectors via MultiLabelBinarizer. After imputing missing values and mitigating outliers using an IQR-based rule, categorical variables were encoded and all features were standardized using StandardScaler prior to distance-based clustering.

Model choice and validation

K-Means was selected due to its efficiency on tabular data and its ability to produce interpretable centroid-based segments. The number of clusters k was determined empirically using the elbow method, as shown in Figure 5.8.

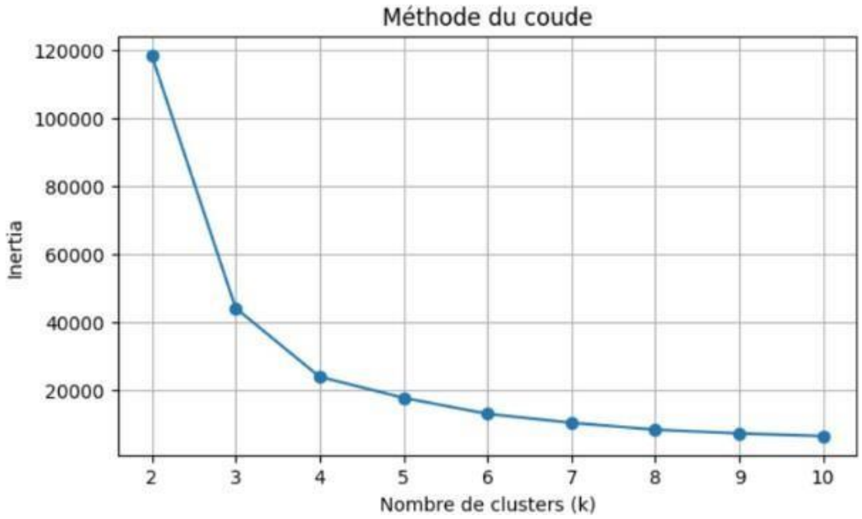


Figure 5.8: Elbow method for selecting the number of clusters k.

Based on this curve, inertia decreases sharply up to $k = 3$, followed by a slower improvement, indicating diminishing returns beyond this point. Therefore, $k = 3$ was retained as the final clustering configuration.

Evaluation

Clustering quality was assessed using the silhouette score. The model achieved a silhouette score of 0.9269, indicating excellent cluster separation and strong within-cluster similarity.

To visually validate this segmentation, Figure 5.9 shows the cluster assignments projected onto the first two PCA components.

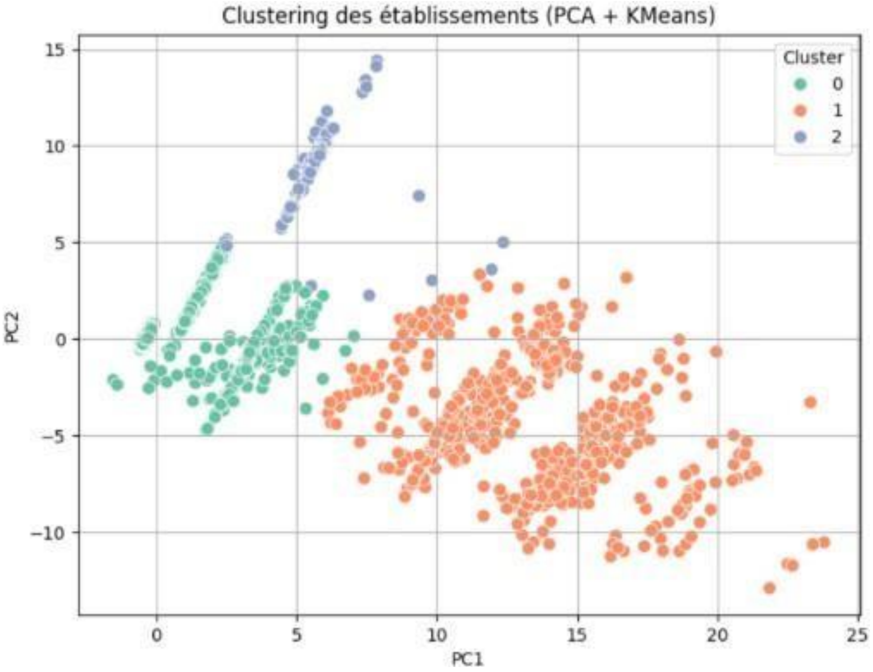


Figure 5.9: Cluster assignments projected onto the first two PCA components.

The three clusters form clearly distinct regions in PCA space with minimal overlap, consistent with the high silhouette score. Cluster 1 exhibits a broader spread along PC1, suggesting higher internal variability, whereas clusters 0 and 2 appear more compact.

4.5 Model Integration

The deployment and monitoring of predictive models are essential to integrating artificial intelligence solutions into the EduSight platform and ensuring their accuracy, reliability, and sustainability over time. This section describes the process adopted to deploy and manage machine learning models within a Power BI environment, enabling advanced educational analytics and real-time insights.

Steps for Deployment in Power BI

— **Model training and saving:** After training, machine learning models are saved in .pkl format, making them reusable and ready for deployment.

— **Power BI setup:** Power BI Desktop is configured to support Python scripting, enabling trained models to be executed directly within the dashboard environment.

— **Python script integration:** Python visuals in Power BI run data preprocessing, feature transformation, and model inference to generate predictions based on pre-trained models.

— **Real-time predictions:** As new educational data is ingested into Power BI, predictions are updated dynamically to monitor performance indicators and detect emerging trends.

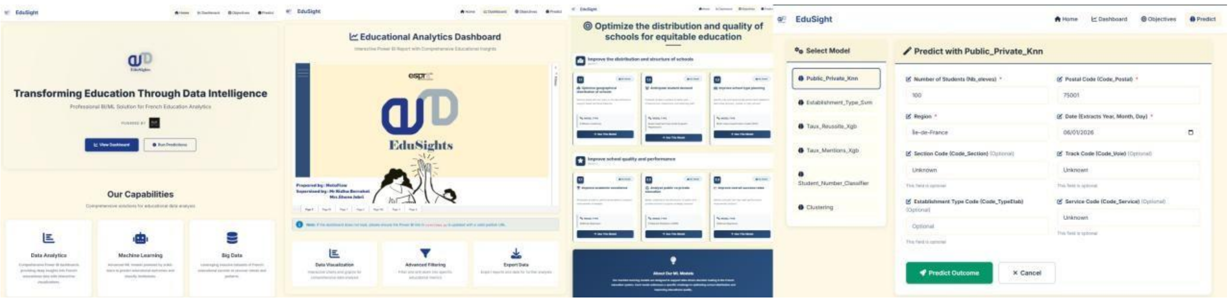
— **Scheduled refresh:** Scheduled refresh ensures dashboards reflect the most recent data and that generated predictions remain up to date.

Chapter 5 : Deployment

The deployment phase constitutes the final stage of the project and represents its most tangible outcome, where the results of all preceding phases are consolidated into an operational solution. Following the completion of data acquisition and preparation, the implementation of the ETL workflow and data warehouse, and the development of analytical assets such as dashboards and predictive models, the proposed solution was deployed in the form of a web-based platform. This approach ensures that the project outputs extend beyond technical artifacts and can be directly accessed and utilized by end users through an interactive and functional interface.

From an architectural standpoint, the deployment layer serves as an integration point that brings together the core components of the system: the curated and structured data produced by the business intelligence pipeline, the reporting layer dedicated to exploratory analysis and monitoring, and the modeling layer responsible for generating predictive insights. By exposing these components within a unified application, the platform delivers a coherent user experience that supports the full analytical workflow, from data exploration to decision support, while preserving traceability to the underlying data and processing mechanisms.

Figure 6.1 provides an overview of the deployed EduSight web interface. It highlights the main user-facing modules, including the landing page that introduces the platform and its objectives, the analytics dashboard dedicated to interactive BI reporting, and the prediction module that allows users to select a model and generate outputs based on institutional input features. As such, the deployment phase represents the culmination of the overall project workflow, translating data engineering and analytical developments into a practical system designed for real-world use.



Number of Students (Nb_eleves) *: 100

Postal Code (Code_Postal) *: 75001

Region *: Île-de-France

Date (Extracts Year, Month, Day) *: 06/01/2026

Section Code (Code_Section) (Optional): Unknown

Track Code (Code_Voie) (Optional): Unknown

Establishment Type Code (Code_TypeEtab) (Optional): Optional

Service Code (Code_Service) (Optional): Unknown

Figure 6.1: Overview of the deployed EduSight web application interface, including the landing page, analytics dashboard, objectives, and prediction module.

Conclusion and Future Work

Conclusion

Bridging Gaps in Educational Data Systems

This project has addressed key challenges within the educational ecosystem by developing a data-driven analytical platform that supports teachers, school directors, and educational authorities in their decision-making processes. By leveraging business intelligence and advanced analytics, **EduSight** transforms fragmented educational data into actionable insights that enhance transparency, performance monitoring, and strategic planning.

Our Innovation

The EduSight platform goes beyond conventional educational dashboards by integrating:

- comprehensive analysis of school distribution and student allocation,
- performance and success indicators across regions and academic tracks,
- socioeconomic context analysis to better understand educational outcomes.

This approach bridges historical educational data with analytical insights, empowering educational decision-makers to identify strengths, detect weaknesses, and design targeted improvement strategies.

Impact and Future Potential

By addressing existing limitations in educational data analysis, this project demonstrates how data-driven approaches can significantly improve educational system management. Teachers gain better visibility into performance trends, school directors can evaluate institutional outcomes more effectively, and educational authorities are equipped with reliable indicators to support policy evaluation and strategic planning.

As a foundation for future advancements, EduSight opens the door to more advanced analytics, deeper performance evaluation, and broader educational insights that can be applied at regional or national levels.

“This project reflects our vision for a smarter, more transparent, and data-driven educational system.”

Next Plans

Future extensions of EduSight can further increase both its analytical depth and its real-world applicability:

- **Expand data sources and coverage:** Enrich the warehouse with additional datasets such as multi-year indicators, demographic variables, and broader socioeconomic context to support more detailed comparisons and deeper interpretations.
- **Improve data quality and automation:** Strengthen validation rules, monitoring, and automated refresh pipelines to support reliable updates and scalable maintenance as the data volume grows.
- **Advanced analytics and model enhancement:** Extend beyond baseline models by integrating more robust predictive approaches, feature engineering strategies, and model monitoring to support forecasting, risk detection, and explainable insights.
- **User experience and visualization improvements:** Refine dashboard design and navigation, improve interactivity, and add role-oriented views to better adapt the platform to different stakeholders (students, teachers, administrators).
- **Operational deployment and stakeholder collaboration:** Move toward real-world adoption through partnerships with educational institutions and public authorities, enabling continuous monitoring and more impactful, data-informed decision-making.

Team METAFLOW : A Journey of Innovation

As Team **METAFLOW**, we are proud of the journey we have undertaken in this project. From its inception to the final implementation, our collaborative efforts and passion for innova- tion have shaped the development of this platform. We believe that this project reflects the essence of teamwork, creativity, and the pursuit of excellence.

Team METAFLOW : Innovation at the Intersection of Data and Education

Summary of Project Phases, Tasks, and Responsible Members

Table 7.1: Summary of project phases, sub-tasks, tools, and responsible members.

Project Phase	Sub-Task	Task Description	Tools Used	Responsible
Business Understanding	Problem Analysis	Identify educational challenges and project scope	Documentation, Brainstorming	Meryem Bennani
Business Understanding	Objective Definition	Define project objectives and success criteria	Documentation	All Members
Data Collection	Data Source Identification	Identify relevant educational datasets	Excel, CSV	Malek Manai
Data Collection	Data Exploration	Explore data structure and quality	Excel, SQL	Jihed Bakalti
Data Preparation	Data Cleaning	Handle missing values and inconsistencies	Python, SQL	Hamza Zighni
Data Preparation	Data Transformation	Normalize and prepare data for analysis	Python, SSIS	Melek Amimi
Data Warehousing	Schema Design	Design dimensional data warehouse schema	SQL Server	All Members
Data Warehousing	Warehouse Implementation	Implement tables and relationships	SQL Server, SSMS	Malek Manai
Dashboard Development	KPI Definition	Define KPIs and analytical indicators	Power BI, DAX	Zeineb Moujehed
Dashboard Development	Dashboard Design	Build interactive dashboards	Power BI	Meryem Bennani
Machine Learning	Model Selection	Select appropriate ML algorithms	Python, Scikit-learn	All Members
Machine Learning	Model Training	Train and evaluate predictive models	Python, Scikit-learn	Jihed Bakalti
Model Integration	Model Deployment	Integrate ML models into Power BI	Power BI, Python	Jihed Bakalti
Model Integration	Monitoring	Monitor model performance	Power BI	Jihed Bakalti
Report Writing	Section Drafting	Write and compile report sections	Word / LaTeX	All Members
Report Writing	Final Review	Review and finalize the report	Word / LaTeX	All Members

Bibliography

[1] DataScience-PM. Crisp-dm is still the most popular framework for executing data science projects. https://www.datascience-pm.com/crisp-dm-still-most-popular/. [Online]. Available: https://www.datascience-pm.com/crisp-dm-still-most-popular/

[2] ——. What is crisp dm? https://www.datascience-pm.com/crisp-dm-2/. [Online]. Available: https://www.datascience-pm.com/crisp-dm-2/

[3] Selenium Project. Webdriver. https://www.selenium.dev/documentation/webdriver/. [Online]. Available: https://www.selenium.dev/documentation/webdriver/

[4] Real Python. Modern web automation with python and selenium. https://realpython.com/modern-web-automation-with-python-and-selenium/. [Online]. Available: https://realpython.com/modern-web-automation-with-python-and-selenium/

[5] GeeksforGeeks. (2025) Getting started with classification. https://www.geeksforgeeks.org/machine-learning/getting-started-with-classification/. Last updated: 8 Nov 2025. [Online]. Available: https://www.geeksforgeeks.org/ machine-learning/getting-started-with-classification/

[6] ——. (2025) Understanding logistic regression. https://www.geeksforgeeks.org/machine-learning/understanding-logistic-regression/. Last updated: 23 Jul 2025. [Online]. Available: https://www.geeksforgeeks.org/machine-learning/ understanding-logistic-regression/

[7] ——. (2025) Regression in machine learning. https://www.geeksforgeeks.org/machine-learning/regression-in-machine-learning/. Last updated: 16 Dec 2025. [Online]. Available: https://www.geeksforgeeks.org/machine-learning/ regression-in-machine-learning/

[8] ——. (2025) Clustering in machine learning. https://www.geeksforgeeks.org/ machine-learning/clustering-in-machine-learning/. [Online]. Available: https://www.geeksforgeeks.org/machine-learning/clustering-in-machine-learning/