**DEMOGRAPHIC SNAPSHOT OF SCHOOLS IN NYC**

**4 Key features**

**NAMES: Akshitha Yarlagadda**

**Gyaneshwar Rao**

**Sathvika Muthyala**

**Sudheer Kumar Suda**

**Data source:** we collected the data from the <https://opendata.cityofnewyork.us/>

Demographis snapshots of schools in NYC contain information about multiple schools over several academic years, including various performance metrics and statistics. Each row in the dataset seems to represent a specific school in a given academic year, with detailed information such as the school code, name, academic year, and various performance indicators. Here are the three datasets from years 2013-2021

2013\_-\_2018\_Demographic\_Snapshot\_School\_20231116

2019-20\_Demographic\_Snapshot\_-\_School\_20231116

2020-2021\_Demographic\_Snapshot\_School\_20231116

**Objective of this Project:**

Analyse how poverty levels vary across different years.

Investigating how international enrollments in schools change over time.

Investigate the gender distribution in schools.

**Version Control (Git)**

Step 1: Install Git

If you haven't already, install Git on your machine. You can download it from the official website: [Git Downloads](https://git-scm.com/downloads)

Step 2: Create a New Repository

Open a terminal or command prompt and navigate to the directory where you want to create your project. Then run:

git init

Step 3: Create a `.gitignore` File

Create a `.gitignore` file in the root of your project to specify files and directories that should be ignored by Git. This can include compiled binaries, temporary files, and other artifacts.

Step 4: Make Your First Commit

Add your files to the staging area and make your first commit:

git add .

git commit -m "Initial commit"

Step 5: Create a Branch

Create a new branch for the project. This allows you to work on features or fixes without affecting the main branch:

git branch feature-branch

git checkout feature-branch

or in one step: git checkout -b feature-branch

Step 6: Make Changes in the New Branch

Make changes to your project in the feature branch.

Make changes to your code

git add .

git commit -m "Implemented feature X"

Step 7: Switch Back to Main Branch

Before merging changes, switch back to the main branch:

git checkout main

Step 8: Merge Changes

Merge the changes from the feature branch into the main branch:

git merge feature-branch

Step 9: Resolve Conflicts (if any)

If there are conflicts during the merge, resolve them manually. Git will mark the conflicted areas in your files.

Step 10: Push Changes to a Remote Repository

If you're collaborating with a team member, you'll likely want to use a remote repository. GitHub is a common choice, but there are others like GitLab and Bitbucket.

1. Create a repository on the remote platform.

2. Add the remote URL to your local repository:

git remote add origin <remote-repository-url>

3. Push your changes:

git push -u origin main

**Data Preprocessing:**

1. Handling Missing Values: Rows with any missing values were dropped from each dataframe.
2. Checking for Missing Values After Cleaning: Printed the summary of missing values for each cleaned dataframe.
3. Checking for Duplicates: Checked for and removed duplicate rows in each dataframe.

Print the number of duplicate rows found in each cleaned dataframe.

1. Checking the Cleaned Dataframe: Print the information about each cleaned dataframe, including data types, non-null counts, and memory usage.

Printing the descriptive statistics of each cleaned dataframe.

Repeated the same process for all 3 datasets.

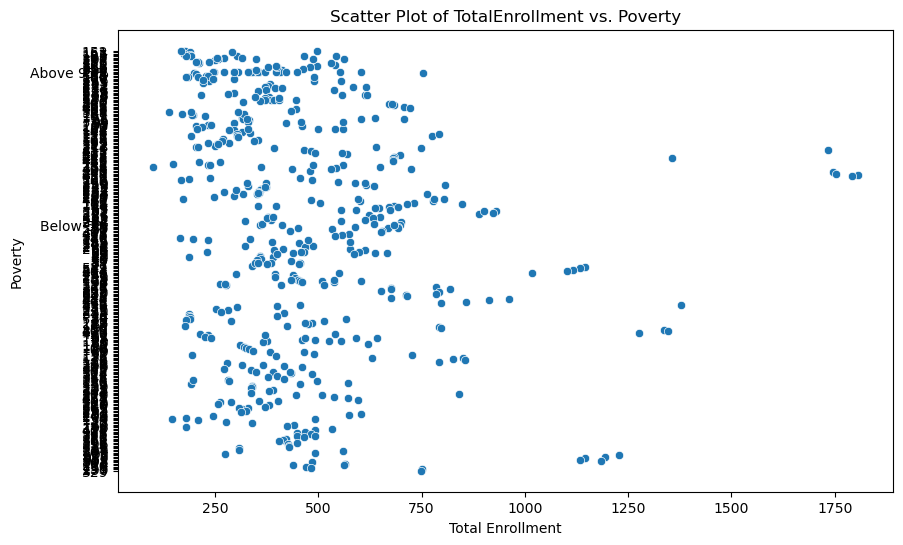
**Handling outliers and Inconsistencies**

Dataset 1:

A subset of the first 500 rows from the 'df1' dataframe, named 'subset\_df,' was selected for visualization.

The code utilizes Seaborn and Matplotlib to create a scatter plot comparing 'TotalEnrollment' on the x-axis and 'Poverty' on the y-axis, with a specified figure size of 10x6 inches.

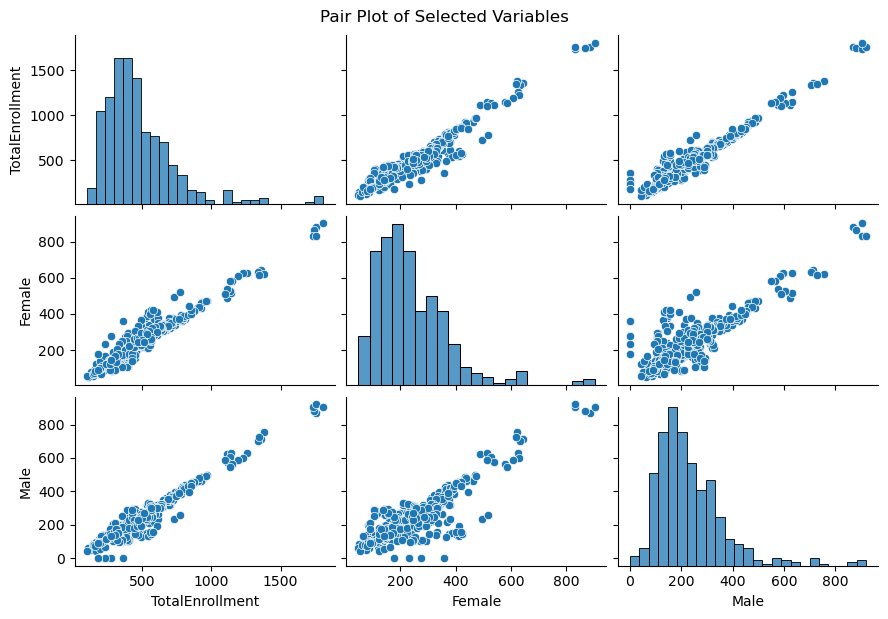
The resulting plot, titled 'Scatter Plot of TotalEnrollment vs. Poverty,' visually represents the relationship between total enrollment and poverty levels in the selected subset.1



Dataset 2:

A subset of the first 500 rows from the 'df2' dataframe, named 'subset\_df,' was chosen for pair plot visualization.

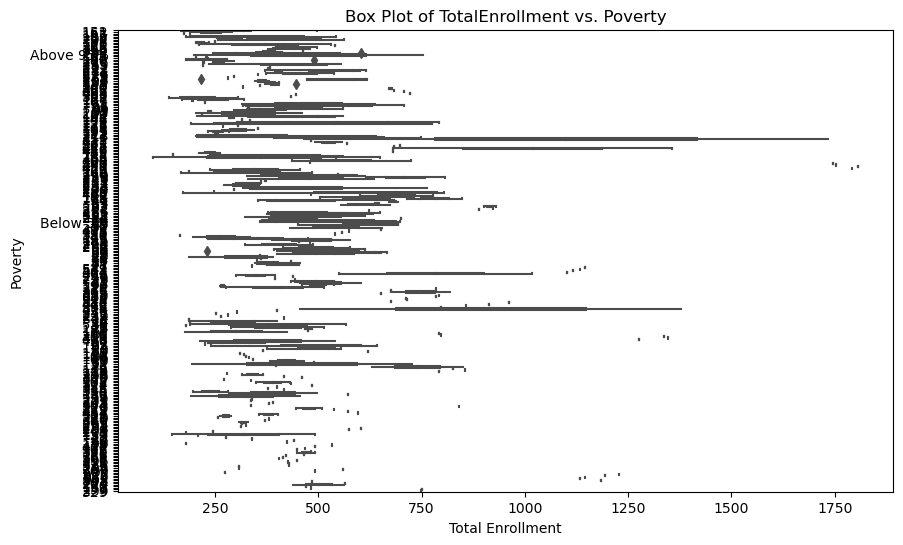
The pair plot was created using Seaborn to explore relationships between selected variables ('TotalEnrollment,' 'Poverty,' 'Female,' and 'Male') with a specified height and aspect ratio, and the title 'Pair Plot of Selected Variables' was added for clarity.



Dataset 3:

A subset of the first 500 rows from the 'df3' dataframe, denoted as 'subset\_df,' was utilized to create a box plot for the variables 'TotalEnrollment' and 'Poverty.'

The Seaborn and Matplotlib libraries were employed to generate the box plot, where 'Total Enrollment' is represented on the x-axis, 'Poverty' on the y-axis, and the plot is visually enhanced with an orange color for better visibility.



The Outliers in all datasets are negligible.

**SQL Queries**

**Query 1:** This query fetches information from the first dataset. It specifically selects the 'SchoolName,' 'TotalEnrollment,' and 'Poverty' columns from the table named `dataset1\_table**`**

**Query 2:** In this query, we're looking at the second dataset. It selects 'SchoolName,' 'TotalEnrollment,' and 'Poverty' columns from the table `dataset2\_table`, but only includes schools where the 'TotalEnrollment' is greater than 1000.

**Query 3:** This query is interesting. It combines data from all three datasets (`dataset1\_table`, `dataset2\_table`, `dataset3\_table`). Then, it calculates the total enrollment and total poverty count for each school across all the datasets.

The total enrollments vary widely across different schools, ranging from relatively small numbers (60-68) to larger ones (21,385). The poverty levels also exhibit a diverse range, with some schools having relatively low poverty rates (43-59) and others with higher rates (12,729.5).

**Query 4:** In this query, we're doing a join operation between `dataset3\_table` and `dataset1\_table` based on the 'SchoolName.' The result includes schools from the third dataset with their 'TotalEnrollment' and 'Poverty,' but only those where the poverty rate is greater than 50**.**

The provided query (Query 4) displays redundant information for the 'P.S. 034 Franklin D. Roosevelt' school, with identical records showing 'TotalEnrollment' as 350 and 'Poverty' as 'Above 95%.' This repetition may indicate a data quality issue or duplication in the dataset, warranting further investigation and potential cleaning to ensure accurate and meaningful analysis.

**Query 5:** Lastly, this query finds the schools with the highest total enrollment across all datasets. It looks at each dataset individually, combines the results, and then identifies the schools with the maximum total enrollment. The final list is limited to the top 5 schools, ordered by the highest total enrollment.

Query 5 presents information about the maximum total enrollment across different schools, indicating that **'Brooklyn Technical High School'** has the highest enrollment with a value of 6040. This query provides a concise summary of the schools with the highest maximum total enrollment, offering valuable insights into the scale of student populations in these educational institutions.

**Collaboration with a Team Member**

Collaborating with a team member involves providing them access to clone the repository, create branches, make changes, and push them back to the remote repository. Then one can pull their changes into their local repository using:

This serves as a fundamental guide, and Git offers numerous additional features for branching, merging, and collaboration. For more in-depth information, team members are encouraged to refer to the [official Git documentation](https://git-scm.com/doc).

Here are the Git commands with brief explanations:

1. Clone Repository:

- `git clone <repository-url>`: Clone the remote repository to your local machine.

2. Create Feature Branch:

- `git checkout -b feature-branch`: Create and switch to a new branch for your feature.

3. Commit Changes:

- `git add .`: Stage changes for commit.

- `git commit -m "Commit message"`: Commit changes with a descriptive message.

4. Push Changes to Remote:

- `git push origin feature-branch`: Push your feature branch to the remote repository.

5. Create Pull Request:

- Open a pull request on the remote repository platform for code review.

6. Review and Merge:

- Team reviews the pull request and merges changes into the main branch.

7. Update Local Repos:

- `git checkout main`: Switch to the main branch.

- `git pull origin main`: Update your local repository with changes from the main branch.

8. Conflict Resolution:

- Resolve conflicts in files marked by Git during pull.

- `git add .`: Stage resolved changes.

- `git commit -m "Resolved conflicts"`: Commit resolved changes.

- `git push origin main`: Push changes to the main branch.

9. Repeat Process:

- Repeat the process for each new feature or task.

These commands form the basic workflow for collaborative development using Git. Adjust branch names and commit messages as needed for your specific project.

# Data Visualization:

# Poverty in every year

# Picture 4

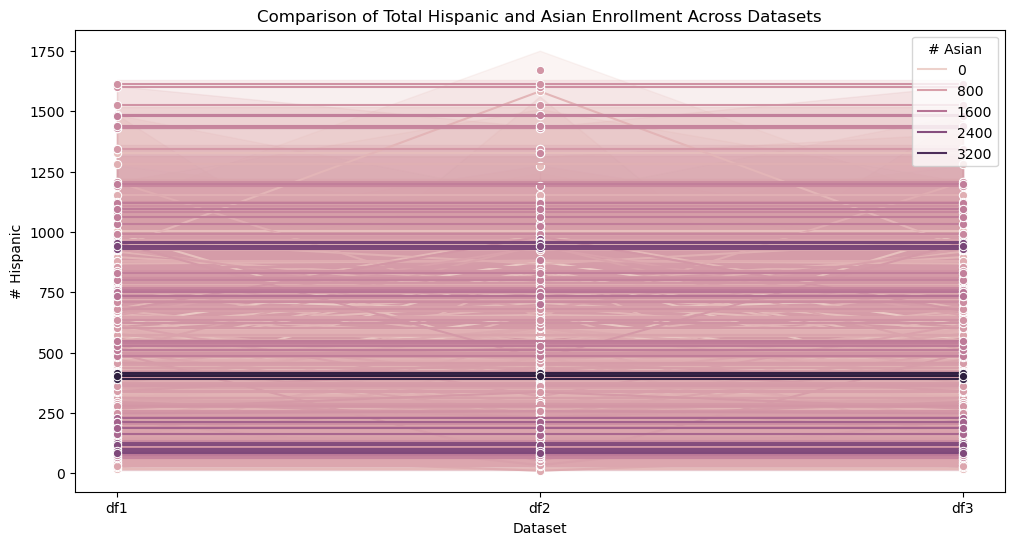
# Relevant columns ('Poverty' and 'Year') are extracted from three different datasets (df1, df2, df3). Datasets are concatenated into a single dataframe (combined\_data\_poverty) with a multi-level index indicating the source dataset. Peaks or shifts in the distribution may indicate changes in the poverty levels over time or differences between datasets.

# Overall Poverty Trends: The mean and median poverty levels show variations across the years, indicating fluctuations in the overall poverty scenario. For example, there is a decrease in both mean and median poverty from 2017-18 to 2020-21.

# Year-to-Year Variability: The standard deviation in poverty levels is relatively high, suggesting substantial variability in poverty rates from one year to another. This variability may be influenced by external factors, economic conditions, or policy changes.

# Poverty Distribution: The summary statistics (min, 25%, 50%, 75%, max) provide a snapshot of the distribution of poverty levels. The minimum and maximum values show the range of poverty rates observed, while quartiles give an idea of the distribution's central tendency.

**2.International enrolments in each year.**



# Hear is the plot of pie charts for each dataset

# Picture 6

# The mean Asian population across the three data frames is relatively consistent, ranging from approximately 95 to 95.41.

# The mean Hispanic population is consistent across the data frames, ranging from approximately 239 to 241.

# The mean total enrollment across the three data frames is similar, with values around 586 to 593.

# The data suggests that the Asian and Hispanic populations in terms of mean sizes are relatively stable across the three data frames.

# The standard deviation indicates the degree of variability in the Asian population, with values around 217 to 218.

# The standard deviation for the Hispanic population is relatively high, suggesting variability in the sizes of Hispanic populations.

# The standard deviation for total enrollment indicates the degree of variability in the sizes of the total student populations.

# The standard deviations for both Asian and Hispanic populations indicate some variability, suggesting that the sizes of these populations can vary.

# This also suggests a strong positive correlation between the number of Hispanic students and the total enrollment. As the number of Hispanic students increases, the total enrollment tends to increase. The positive correlations imply that schools with higher numbers of Asian or Hispanic students tend to have larger total enrollments.

# Gender distribution

# Picture 7

# These statistics provide a numerical summary of the gender distribution in each dataset. The summary statistics for Dataset 3 appear to be identical to those of Dataset 1.

# The mean represents the average number of males and females in each dataset.

# The standard deviation indicates the degree of variability or spread around the mean.

# The minimum and maximum values show the range of counts observed

# The mean values for males and females are similar across datasets, with Dataset 2 having slightly higher means.

# The standard deviations suggest variability in the counts, with some datasets having higher variability.

# The presence of zero as the minimum indicates that there are entries with no reported males or females.

# Challenges and Solutions:

# Challenge: Plotting large datasets was time-consuming.

# Solution: Implemented optimization strategies to enhance plot generation efficiency.

# Gender Distribution Analysis:

# Challenge: Similarities in gender distribution data across datasets hindered conclusive insights.

# Solution: Employed a nuanced approach, blending statistical techniques and domain-specific insights for a comprehensive analysis.

# Outlier Identification:

# Challenge: Identifying outliers in the extensive dataset required robust detection methods.

# Solution: Implemented advanced outlier detection techniques to enhance data integrity.

# Handling Large Datasets:

# Challenge: The sheer volume of data posed challenges in terms of processing and interpretation.

# Solution: Employed strategic data sampling and processing techniques to manage and analyze large datasets effectively.

# Merging Three Datasets:

# Challenge: Working with three distinct datasets presented complexities in data merging and harmonization.

# Solution: Implemented meticulous data merging strategies to ensure coherence and relevance across datasets.

# Future Steps:

# 1. Advanced Predictive Modeling for Enrollment Trends:

# Future Step: Implement advanced predictive modeling techniques to forecast future trends in poverty levels and total enrollments. Utilize machine learning algorithms to analyze historical data and identify potential patterns or cyclical variations in enrollment and poverty. This predictive modeling could offer valuable insights for educational policymakers, enabling them to anticipate changes, allocate resources more effectively, and develop targeted interventions to address emerging challenges.

# 2. Integration of Socioeconomic and Academic Indicators:

# Future Step: Expand the analysis by integrating additional socioeconomic and academic indicators into the dataset. Include variables such as academic performance, graduation rates, and teacher-student ratios. Analyzing the interplay between these factors and enrollment dynamics, poverty levels, and gender distribution could provide a more comprehensive understanding of the factors influencing the educational landscape. This expanded dataset could contribute to the development of holistic strategies for improving educational outcomes and equity.

# These future steps aim to take the project to a more sophisticated level, incorporating predictive modeling, expanding the scope of indicators, enhancing automation, and fostering stakeholder engagement for impactful policy recommendations.

# Conclusion:

**Poverty over the time**

# -Trends in Overall Poverty: Mean and median poverty levels exhibit fluctuations over the years, highlighting variations in the general poverty landscape. Notably, there is a decrease in both mean and median poverty rates from 2017-18 to 2020-21.

# -Yearly Fluctuations: The standard deviation in poverty levels is notably high, indicating significant variability in poverty rates between consecutive years. This variability may be influenced by external factors, economic conditions, or policy changes.

# -Distribution of Poverty: Summary statistics, including minimum, 25th percentile, median (50th percentile), 75th percentile, and maximum values, offer insights into the spread of poverty levels. The minimum and maximum values depict the range of observed poverty rates, while quartiles provide an understanding of the distribution's central tendency.

**International enrollments in each year**

# Consistency in Mean Populations:

# - The mean Asian population remains relatively consistent across the three dataframes, ranging from approximately 95 to 95.41.

# - Similarly, the mean Hispanic population shows consistency, with values ranging from around 239 to 241.

# - Mean total enrollment also exhibits similarity across the dataframes, with values ranging from 586 to 593.

# - These findings suggest stability in the mean sizes of Asian and Hispanic populations across the datasets.

# Variability Indicated by Standard Deviation:

# - The standard deviation for the Asian population hovers around 217 to 218, indicating a moderate degree of variability in Asian population sizes.

# - In contrast, the standard deviation for the Hispanic population is relatively high, suggesting significant variability in the sizes of Hispanic populations.

# - The standard deviation for total enrollment reflects the degree of variability in the sizes of the overall student populations.

# Both Asian and Hispanic populations show some variability, as indicated by their standard deviations, implying that the sizes of these populations can fluctuate.

**Gender distribution**

# Interpretation of Statistical Measures:

# Mean:

# - The mean represents the average number of males and females in each dataset.

# - Mean values for males and females are similar across datasets, with Dataset 2 having slightly higher means.

# Standard Deviation:

# - The standard deviation indicates the degree of variability or spread around the mean.

# - Standard deviations suggest variability in the counts, with some datasets exhibiting higher variability.

# Minimum and Maximum:

# - The minimum and maximum values show the range of counts observed.

# - The presence of zero as the minimum indicates that there are entries with no reported males or females.

# In summary, the mean provides an average count, the standard deviation indicates how much individual counts deviate from the average, and the minimum and maximum values illustrate the range of observed counts. The slight differences in mean values and variability across datasets may suggest variations in the male and female counts. Additionally, the presence of zeros in the minimum values indicates instances where no males or females are reported in certain entries.