**Detailed Report on the Grading System Implementation**

**1. Overview** The Grading System is a comprehensive software application designed to assist instructors in managing student grades effectively. It supports both absolute and relative grading methodologies, enabling instructors to analyze and adjust grade distributions seamlessly. The primary goal is to ensure fairness and statistical validity in grade adjustments while providing instructors with detailed analytics, including descriptive statistics and visualizations. This project adheres to the guidelines set by the **Higher Education Commission (HEC)** for grade distributions.

This report outlines the system's key features, functionalities, and implementation details, ensuring clarity and usability for end-users.

**2. Key Features**

**2.1 Input Module**

The input module allows instructors to input, manage, and manipulate student grades efficiently. The features include:

**a. File Management**

* **Open Existing Files**: Instructors can load student data from existing CSV files.

A computer screen with white text

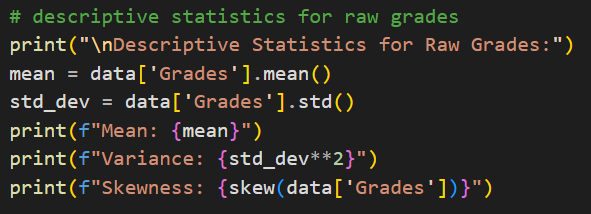
Description automatically generated

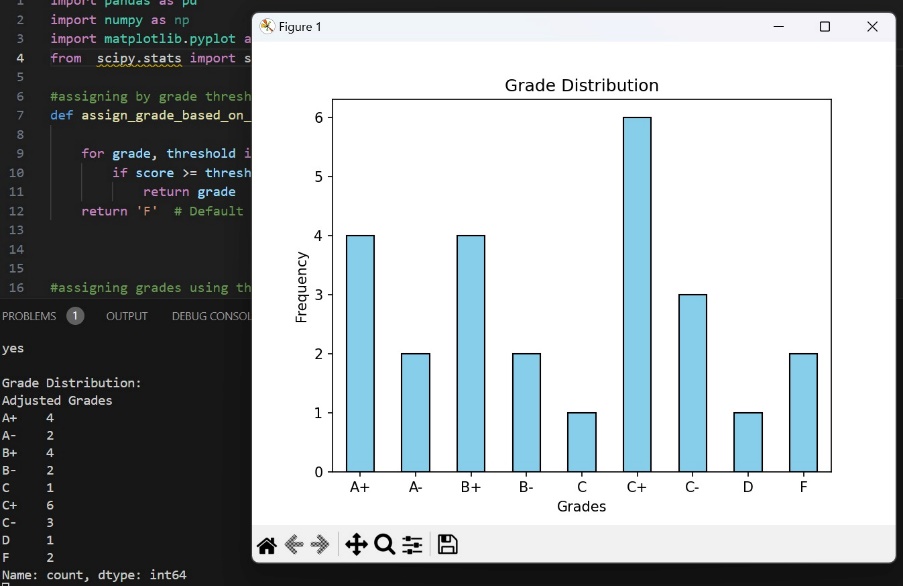
**b. Grading Options**

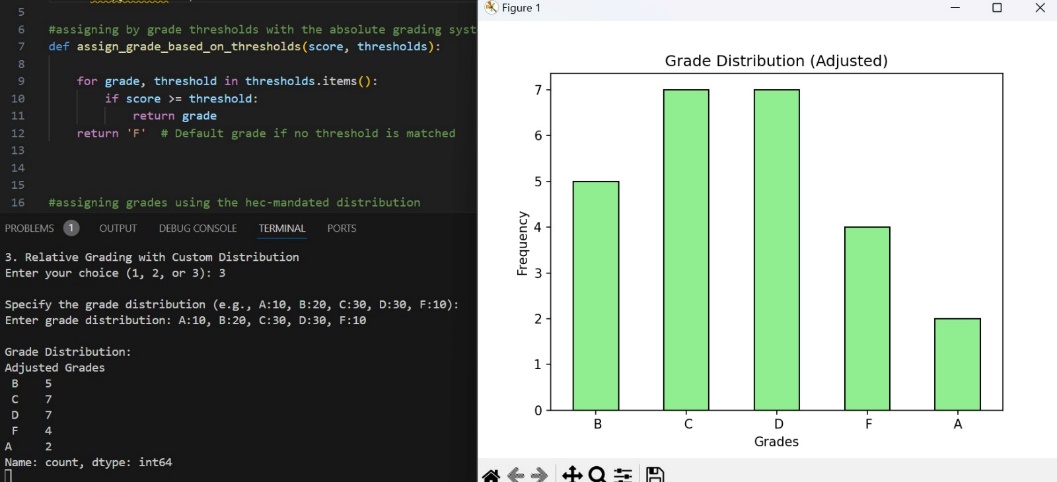
* **Absolute Grading**: Provides the ability to assign fixed grade thresholds (e.g., A for ≥90%).
* **Relative Grading**: Offers flexibility to fit grades into a predefined distribution, such as a normal curve, with optional customization of grade percentages (e.g., 10% A, 20% B).

**2.2 Statistical Analysis** The system incorporates robust statistical analysis tools for both grading methodologies:

* **Descriptive Statistics**: Automatically calculates mean, variance, and skewness of the input grades.



* **Graphical Representations**:
  + **Histograms**: Visualize the distribution of grades.

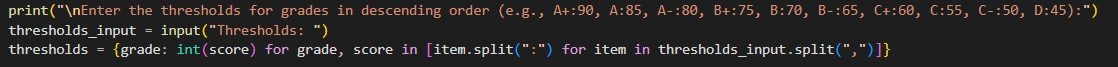


* + **Normal Distribution Curve**: Plots a bell curve overlay for relative grading adjustments.

**2.3 Grade Adjustment**

**a. Absolute Grading**

* Allows the professor/lecturer to define thresholds to assign grades based on student performance.



* Ensures grades align with fixed boundaries (e.g., >=90% = A).

A computer screen with text

Description automatically generated

**b. Relative Grading**

* Implements algorithms for curving grades:
  + **Z-Score Scaling**:

Z-Score scaling, also known as standard score normalization, is a statistical method used to transform raw data into a standard scale based on the dataset's mean and standard deviation. This process helps in comparing data points across different distributions. Here it standardizes scores based on mean and standard deviation.

* + **Purpose in Grading**

Z-scores are often used in **relative grading** systems to determine how far a student's score deviates from the average performance of the class. Grades are then assigned based on the relative position of Z-scores (e.g., higher Z-scores might receive A+ or A, while lower Z-scores might receive C or F)

* + **Curve Fitting**: **2. Curve Fitting**

Curve fitting is a statistical technique used to model the relationship between data points by fitting them to a mathematical function. In the context of grading, it ensures a smoother distribution of grades that aligns with predefined or natural patterns.

* + **Purpose**

To **adjust raw scores** so they fit a desired grade distribution, like a bell curve (normal distribution). To **normalize grading** in cases where scores are skewed or do not naturally align with expected patterns.

**c. Flexibility and Customization**

* Allows instructors to set grade boundaries (absolute) or desired grade distribution percentages (relative).
* Guarantees the final grading adheres to the instructor’s criteria.

**3. Implementation Details**

**3.1 Programming Framework**

* **Language**: Python
* **Libraries**:

**1. Pandas**

**Purpose**:  
Pandas is used for handling data input, manipulation, and cleaning. It is especially useful for reading CSV/Excel files and performing data analysis on the grade data.

**Usage**:

* **Reading Data**: To load grades from a CSV or Excel file into a DataFrame.
* **Data Cleaning**: To handle missing values and filter out invalid data.

**2. NumPy**

**Purpose**:  
NumPy is used for numerical operations such as calculating the **mean**, **variance**, and **standard deviation** of the grades, as well as performing statistical transformations (e.g., z-score scaling).

**Usage**:

* **Statistical Calculations**: Calculating mean, standard deviation, and variance for the relative grading algorithm.
* **Z-Score Calculation**: To adjust grades using z-scores.

**3. Matplotlib**

**Purpose**:  
Matplotlib is used for creating visualizations such as histograms, box plots, and bar charts to compare the original and adjusted grade distributions.

**Usage**:

* **Histograms**: To visualize the grade distribution before and after adjustment.
* **Boxplots**: To identify the spread of grades and outliers.
* **Bar Charts**: To compare the percentage of students in each grade category.

**4. SciPy**

**Purpose**:  
SciPy is used for advanced statistical operations, particularly when implementing **Z-score scaling** or performing **curve fitting** for relative grading. It is also useful for calculating skewness and other statistical measures.

**Usage**:

* **Skewness**: To measure the asymmetry of the grade distribution.
* **Curve Fitting**: To fit the data to a normal distribution or another statistical distribution..

**3.2 Code Structure**

**Authentication**

* Verifies the instructor's credentials using a username-password combination before granting access.

A screen shot of a computer

Description automatically generated

**File Management**

* Supports loading and creating datasets via CSV files.
* Ensures error handling for file operations, such as handling empty files or invalid formats.

**Adding Student Grades**

* Allows manual addition of student names and grades to the dataset.
* Validates input data for correctness (e.g., numeric validation for grades).

**Grading Algorithms**

* **Absolute Grading**:
  + Uses predefined bins to assign grades.
  + Example:
* Semester System
* A 85 and above
* A- 80 - 84
* B+ 75 - 79
* B 71 - 74
* B- 68 - 70
* C+ 64 - 67
* C 61 - 63
* C- 58 - 60
* D+ 54 - 57
* D 50 - 53
* F Below 50
* **Relative Grading**:
  + Computes Z-scores for normalization.
  + Assigns grades based on statistical thresholds (e.g., 1 standard deviation above the mean = A).

**Graphical Visualizations**

* **Histograms**:
  + Displays frequency distribution of student grades.
  + Allows instructors to assess grade trends visually.
* **Normal Distribution Curve**:
  + Provides a smooth bell curve overlay to represent the idealized grade distribution.

**4. User Interface**

**4.1 Design Principles**

* **User-Friendly Navigation**: Intuitive layout with clear labels and instructions.
* **Modular Design**: Segregates functionalities into distinct frames for simplicity.

**4.2 Streamlit elements.**

* **Login Frame**: Authentication portal for instructors.
* **Main Menu**: Options to open/create files, view grading options, and exit the application.
* **Grading Frame**: Interface for inputting grades and applying grading algorithms.
* **Graph Frame**: Displays visual representations of grade distributions.

**5. Benefits and Use Cases**

**a. Efficiency**

* Streamlines grade input and processing.
* Automates statistical analysis and visualization, reducing manual effort.

**b. Flexibility**

* Accommodates diverse grading policies.
* Customizable thresholds and distributions ensure adaptability to institutional standards.

**c. Enhanced Insights**

* Provides detailed statistical insights, enabling instructors to evaluate and refine their grading strategies.

**6. Challenges and Future Enhancements**

**Challenges**

* Handling large datasets efficiently.
* Ensuring the accuracy of relative grading algorithms for varied distributions.

**Future Enhancements**

* Integration with Learning Management Systems (LMS) for automated grade import/export.
* Advanced statistical tools, such as outlier detection and clustering.
* Inclusion of dynamic plots for real-time data visualization.

**7. Conclusion** The Grading System is a robust and versatile tool designed to meet the grading needs of educational institutions. Its dual grading methodologies, statistical analysis, and graphical visualizations ensure a seamless experience for instructors. With future enhancements, the system has the potential to become a comprehensive solution for grade management and academic analytics. In the future, the Student Grading System will integrate with Learning Management Systems (LMS) for automatic grade imports, saving time and reducing errors. Machine learning will predict at-risk students, enabling early interventions.