Student Performance Analytics and Marks prediction

Explanatory file by

Group members:

Rao Faris Ali

Manahil Asghar

Noor Ul Huda

Importing Libraries (All Members)

- import pandas as pd
- import numpy as np
- import seaborn as sns
- import matplotlib
- from matplotlib import pyplot as plt
- pd.options.mode.chained_assignment = None

Explanation:

- pandas – for loading CSV, dataframes, manipulation
- numpy – numerical operations
- seaborn – statistical visualizations
- matplotlib / pyplot – plotting graphs
- Disabled “SettingWithCopyWarning” for cleaner output

Loading the Dataset (Faris)

- sp=pd.read_csv('StudentsPerformance.csv')
- sp.head(5)
- sp.describe()

Explanation:

- Reads the dataset from CSV
- .head() shows first 5 rows
- .describe() shows summary statistics

Data Cleaning (Faris)

- Missing values
- sp.isnull().sum()
- Counts how many empty values each column has.

- Duplicate rows
- sp.duplicated().sum()
- Checks repeated rows.

Categorical Frequency Bar Plots (Faris)

- Creates bar charts of gender, lunch, test prep, race/ethnicity, and parental education:
- Example:
- sp['gender'].value_counts().plot.bar(...)

Explanation:

Shows how many students fall into each category.

Pie Charts (Faris)

- Used plt.subplot() to draw 5 pie charts in one row.
- Example:
- plt.pie(size, labels=labels, autopct='%1.2f%%')

Explanation:

Shows percentage share of each category.

Score Distribution (Manahil)

- Histograms
- sns.histplot(sp['math score'], kde=True)
- Shows distribution + density curve.

- Reading and Writing comparison
- sp['reading score'].value_counts().plot(kind='bar')
- Violin Plots
- sns.violinplot(y='math score', data=sp)
- Shows distribution + outliers.

Gender-wise and Category-wise Barplots (Manahil)

- sns.barplot(x="gender", y="math score", data=sp)
- sns.barplot(hue="gender", x="lunch", y="math score", data=sp)

Explanation:

Shows how scores differ based on:

- gender
- test prep course
- lunch type

Calculating Total Score & Percentage (Manahil)

- sp['total_score'] = sp['math score'] + sp['reading score'] + sp['writing score']
- sp['percentage'] = sp['total_score']/3

Explanation:

- Total = sum of three subjects
- Percentage = average of three subjects

Grade Calculation Function (Manahil)

- def calcgrade(percentage,result):
- if result=='Fail': return 'E'
- if percentage>=90: return 'A'
- if percentage>=75: return 'B'
- if percentage>=50: return 'C'
- if percentage>=33: return 'D'
- else: return 'E'

Explanation:

- Assigns grade based on percentage.
- Fail case always maps to grade ‘E’.
- Applied using:
- sp['grade'] = sp.apply(...)

Pairplot & Correlation Heatmap (Manahil)

- Pairplot
- sns.pairplot(sp, hue='gender')
- Shows relationship between pairs of variables.

- Heatmap
- sns.heatmap(sp[['math score','reading score','writing score']].corr(),annot=True)
- Shows correlation values between scores.

Encoding Categorical Variables (Noor)

- sp_regr = pd.get_dummies(sp, columns=[...], drop_first=True)

Explanation:
Converts text categories into numeric dummy variables for regression.

Splitting Data for Regression (Noor)

- X_train, X_test, y_train_math, y_test_math = train_test_split(...)

Explanation:
20% test, 80% training.
Repeated separately for:

- math
- reading
- writing
- total score
- percentage

Linear Regression Models (Noor)

- Example (Math Regression)
- math_model = LinearRegression()
- math_model.fit(X_train, y_train_math)
- y_pred_math = math_model.predict(X_test)
- Evaluated using:
- mean_squared_error()
- r2_score()
- This is repeated for reading, writing, total, and percentage.

Adding Predictions Back to Dataset (Noor)

- sp['pred_math'] = math_model.predict(X)
- sp['pred_total'] = total_model.predict(X)

Explanation:
Creates new columns comparing actual vs predicted values.

Actual vs Predicted Plot (Noor)

- plt.plot(sp['total_score'].head(50))
- plt.plot(sp['pred_total'].head(50))
- Shows differences between real and predicted performance.

Final Comparison Table (Noor)

- sp[['math score','pred_math', ...]].head(10)
- Shows first 10 rows of actual vs predicted results.

Member-Wise Final Summary

Rao Faris

- Imported libraries
- Loaded dataset
- Missing/duplicate checks
- All bar plots
- All pie charts

Manahil Asghar

- Histograms
- Violin plots
- Gender/lunch/test-prep barplots
- Total score + percentage calculation
- Grade function + grade visualization
- Pairplot + heatmap

Noor-ul-Huda

- Dummy variable encoding
- Train-test split
- All 5 regression models
- Model evaluation
- Predictions added to dataframe
- Actual vs predicted plot
- Final comparison table