SMAI A1

Akshith

August 2025

Contents

1 Question 1

Question 1.1

1.1.a

Gender Distribution
Plot Chosen: Bar Chart
Reason: Gender is a variable with only Male, Female, Other. A bar chart makes it easy to compare the number of students in each category.

Major Distribution
Plot Chosen: Bar Chart
Reason: B.Tech, MS, PhD are discrete categories. A bar chart clearly shows how many students are enrolled in each major.

Program Distribution
Plot Chosen: Bar Chart
Reason: CSE, ECE, CHD, CND are categorical. A bar chart helps visualize which programs are most or least popular among students.

4. GPA Distribution
Plot Chosen: Histogram with KDE Curve
Reason: GPA is a continuous variable. A histogram shows the overall spread of GPA values, while the KDE curve provides a smooth estimate of the distribution shape.

5. Program Conditioned on Major
Plot Chosen: Grouped Bar Chart
Reason: This shows how program enrollment varies across majors (i.e., distribution of Program | Major). Grouped bars allow side-by-side comparison.

6. GPA Conditioned on Major
Plot Chosen: Boxplot
Reason: GPA distributions across majors need to be compared. Boxplots show median, spread, and outliers effectively, making differences between majors easy to interpret.

7. GPA Conditioned on Program
Plot Chosen: Boxplot
Reason: Similar reasoning as above, but across programs instead of majors.

8. GPA Conditioned on Program and Major
Plot Chosen: Grouped Boxplot
Reason: Grouped boxplots allow comparison of GPA within each program, separated by major.

9. Gender, Major, Program, and GPA of 100 Sampled Students
Plot Chosen: Pairplot
Reason: A small sample allows visualization of relationships across multiple variables simultaneously. Using color (for gender/major) makes clusters and trends easier to see.

10. Entire Dataset Summary
Plot Chosen: Pairplot
Reason: A pairplot provides a high-level summary of the dataset, showing correlations, distributions, and class separability across multiple variables at once. It is ideal for exploratory analysis.

1.1.b

Mean GPA: 7.33, Standard Deviation: 1.04

This is simialar to the mean and deviation of Btech as they are the majority of student population.

1.1.c

This is dependent on the percentages given in the data

Table 1: Program and Major Counts

Program	Major	Count
CHD	B.Tech	738
CHD	MS	421
CHD	PhD	264
CND	B.Tech	650
CND	MS	388
CND	PhD	234
CSE	B.Tech	2839
CSE	MS	586
CSE	PhD	269
ECE	B.Tech	2796
ECE	MS	556
ECE	PhD	259

1.2

mean deviation
7.325124718915341 0.04419263605280592
7.3302922686132606 0.039820426803726036

Both methods give a mean GPA close to the overall population mean. Stratified sampling typically has lower standard deviation, because it ensures proportional representation of each major, reducing variance due to imbalance.

1.4

Sampling is mostly done without replacement to avoid duplicates. If some GPA bins are too sparse, then with replacement is necessary to maintain approximate uniformity.

1.5

Yes, small groups were handled by sampling with replacement.

2 Question 2: KNN Classification on Student Dataset

1. Train/Validation/Test Split & Feature Encoding We split the dataset into train, validation, and test sets using `train_val_test_split`. Applied per-feature transformations:

- **StandardScaler** for numeric (GPA).
- **OneHotEncoder** for categorical (Program).

- **OrdinalEncoder** for ordered categorical (Major: B.Tech $<$ MS $<$ PhD). Took the order just to check ordinal Encoders.

2. Best k (Euclidean Distance)

- Test odd values of $k = \{1, 3, 5, \ldots, 33\}$.
- For each k, compute validation accuracy using Euclidean distance.
- Plot accuracy vs. k.
- Select the k with the highest accuracy.

3. Compare Other Distance Metrics We repeat the same experiment with:

- Manhattan (L1) distance
- Cosine distance

Then compare validation accuracy vs. k across the three metrics.

4. Validation F1 Score vs. k Instead of accuracy, compute F1 score for each k and each distance metric. Collect results into a matrix: (distance metric $\times$ k).

5. Heatmap of F1 Scores Use `plot_knn_f1_heatmap` to visualize F1 scores:

- Rows: distance metrics (Euclidean, Manhattan, Cosine)
- Columns: values of k

This helps quickly spot the best metric–k combination.

6. Which Distance Metric is Better?

- **Euclidean:** performs best when continuous features are normalized.
- **Manhattan:** can work better with mixed distributions or when outliers exist.
- **Cosine:** often effective for sparse, high-dimensional features (like one-hot encoding), but may underperform if dataset is small or categorical-heavy.

Here Cosine encoding shows slightly better accuracy, though marginal.

7. Single-Feature F1 Table We evaluate using individual features:

- GPA (scaled)
- Major (ordinal)
- Program (one-hot)

We collect F1 scores for all k values across all distance metrics, presenting them in a table (rows: k, columns: features).

8. Which Single Feature Performs Best? Compare single-feature models against the all-features model.

- Often, GPA (numeric and well-scaled) is a strong predictor.
- In this data set, too, GPA is the better predictor, but marginally as the data show very little patterns.

3 Question 3: Polynomial Regression with Regularization

1. Polynomial Regression Across Degrees (1–6) We fit polynomial regression models of degrees 1 through 6 under three setups:

- No regularization
- L1 regularization (Lasso)
- L2 regularization (Ridge)

Trend: Training MSE decreases as degree increases (more flexible model). Validation MSE initially decreases, but after degree ≈ 2 it begins to rise due to overfitting. This trend is consistent across all three setups, though regularization mitigates overfitting.

2. Regularization Strength Selection For each degree and regularizer, we tuned α using validation MSE. At the best degree ($d = 2$), we plotted $\log(\alpha)$ vs validation MSE. The curve shows a U-shape:

- Very small α: almost no regularization, prone to overfitting.
- Very large α: excessive shrinkage, underfitting.
- Optimal α: balanced bias-variance, lowest validation MSE.

3. Best Setup The best-performing configuration was:

- Regularizer: L1 (Lasso)
- Polynomial degree: 2
- $\alpha \approx 4.6 \times 10^{-4}$
- Train MSE: 0.813
- Validation MSE: 0.861
- Test MSE: 0.800

4. Feature Importance

L1 (sparse model): Non-zero features were

$$\{x0,\, x3,\, x5,\, x7,\, x0^2,\, x0x3,\, x0x4,\, x0x5,\, x0x6,\, x1x6,\, x2^2,\, x2x3,\, x2x4,\, x2x5,\, x2x6,\, x3^2,\, x3x4,\, x3x5,\, x3x7\}.$$

This indicates that many features were shrunk to zero, with $x3$ and its interactions dominating.

L2 (dense model): Top predictors by coefficient magnitude were

$$\{x3^2,\, x3,\, x1x3,\, x0x3,\, x3x7,\, x3x6,\, x3x5,\, x3x4,\, x2x3,\, x2x6\}.$$

Unlike L1, all features remain, but the coefficients highlight the dominance of $x3$.

5. Comments

- Without regularization, higher-degree models severely overfit.
- L1 regularization provided the best test performance and interpretability, as irrelevant features were removed.
- L2 regularization stabilized coefficients but does not reduce test error as effectively.
- As data is still scattered not very ordered the differences are minimal.
- In the context The choice between L1 and L2 depends on whether you need feature selection (L1) or a more stable model with all features having some influence (L2). SO here we prefer L1
- **Best overall setup:** Degree 2 with L1 regularization ($\alpha \approx 4.6 \times 10^{-4}$).