Feature Engineering Report

1. Purpose

The main objective of feature engineering in this was to transform the raw student dataset into a form that is both easier to interpret and suitable for building predictive models. By carefully creating and refining features, we ensured that the dataset reflects important aspects of student behavior and performance, while also being compatible with machine learning requirements.

2. Key Feature Engineering Steps

a) Attendance Categories

- From: attendance_rate (numeric %)
- To: attendance_category (Low <50%, Medium 50–75%, High >75%)
- Why: Instead of just seeing a percentage, these categories allow us to quickly identify students with poor, average, or excellent attendance. This makes it easier to compare groups and to flag students at risk due to low attendance.

b) GPA Bands

- From: prior_gpa_10pt (0–10 scale)
- To: gpa_band (Low <5, Medium 5–8, High >8)
- Why: Grouping GPA into ranges simplifies interpretation. Instead of handling dozens of GPA values, we can focus on broad performance groups — struggling students, average performers, and high achievers.

c) Study Hours Binning

- From: study_hours_per_week (continuous hours)
- To:
  - study_hours_bin: equal-width ranges (0–5, 6–10, 11–15, 16–20, 20+)
  - study_hours_qbin: quartiles (Q1–Q4, each containing ~25% of students)
- Why: Students' weekly study hours vary widely. By binning into ranges and quartiles, we can compare "light," "moderate," and "heavy" study groups. Quartiles also help ensure balanced group sizes.

d) Log Transformation

- From: study_hours_per_week
- To: study_hours_per_week_log (log-transformed)
- Why: Some students study very few hours while others study a lot, making the data highly skewed. Applying a log transformation reduces this imbalance and makes the distribution closer to normal, which is better for most models.

e) Encoding Categorical Variables

- From: Features like gender, city, course_stream, has_internet, device_type, parental_education, scholarship, along with engineered categories (attendance_category, gpa_band, study_hours_bin, study_hours_qbin).
- To: One-hot encoded columns (e.g., gender_Female, city_Delhi, attendance_category_High (>75%)) represented as 0/1.
- Why: Machine learning models need numeric input. One-hot encoding allows us to include categorical information without creating false "order" between categories.

f) Scaling Continuous Features

- From: Continuous features like age, study_hours_per_week, attendance_rate, prior_gpa_10pt, test_score, fee_paid_inr, attendance_rate_pct, study_hours_per_week_log.
- To:
    - Standard Scaler: rescaled to have mean = 0 and standard deviation = 1.
    - MinMax Scaler: rescaled to fit between 0 and 1.
- Why: Scaling ensures that features with large values (like fee_paid_inr) don't overshadow smaller-scale features (like age). Standardization makes models more stable and training more efficient.

3. Final Dataset Summary

- Rows: 200
- Columns: 248 (after encoding and transformations)
- Target variable: test_score

Conclusion

Through careful feature engineering, we transformed raw student data into a structured dataset that highlights the most important drivers of performance. The dataset is now well-prepared for machine learning models, while also being easy to interpret for educators and decision-makers.