Modelling Report

Team 2

1 Objective

The goal of this project is to accurately predict tuition payment for the year 2023 using relevant features from the 2022 data. Among all available features, **Tuition Payment 2022** was found to be the most informative predictor. This report also explores clustering techniques to uncover patterns in the student data.

2 Data Overview

- **Target variable**: Tuition Payment 2023
- **Primary feature used**: Tuition Payment 2022
- **Train-test split**: 80% training, 20% testing

3 Regression Model Performance

Model	**MSE**	**R^2 Score**
Linear Regression	0.0156	0.8801
Ridge Regression	0.0156	0.8801
Lasso Regression	0.0261	0.7995

Table 1: Regression performance metrics

Note: Linear and Ridge Regression yield identical results, indicating no overfitting in the Ridge model. Lasso performs slightly worse due to stronger regularization.

4 Classification Model Performance

Classification was performed by adding a classifier head or thresholding on the regression output. Results are as follows:

Model	**Accuracy**
Linear Regression + Classifier	11.72%
Ridge + Classifier	98.41%
Lasso + Classifier	98.41%
Logistic Regression	98.41%
Random Forest	98.41%
XGBoost	98.41%
Deep Neural Network	98.41%
Ensemble	98.41%

Table 2: Classification accuracy across models

Key Insight: All models, except Linear Regression with a classifier head, converge to approximately **98.5% accuracy**, demonstrating model saturation. This high accuracy is largely attributed to the 80-20 train-test split, where the large training data provides substantial advantage. On reducing the test size to 10%, **accuracy reached 100%**, suggesting that models were near-perfectly fitting the training data.

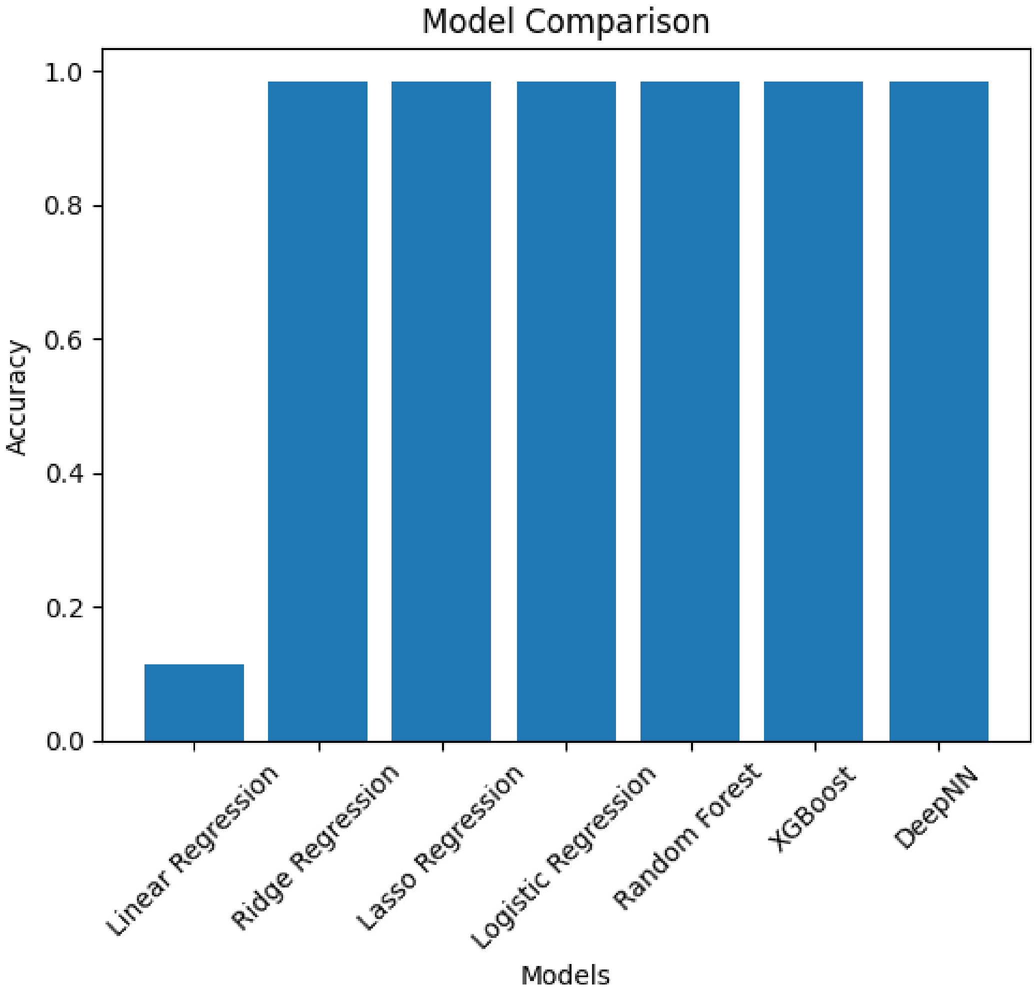

Figure 1: Model comparison chart across classifiers

5 Deep Neural Network Architecture

```
nn.Sequential(
    nn.Linear(input_dim, 256),
    nn.BatchNorm1d(256),
    nn.ReLU(),
    nn.Dropout(0.3),
    nn.Linear(256, 128),
    nn.BatchNorm1d(128),
    nn.ReLU(),
    nn.Dropout(0.3),
    nn.Linear(128, 64),
    nn.ReLU(),
    nn.Linear(64, num_classes)
)
```

The DNN was trained with AdamW optimizer and StepLR scheduler. Its performance matched other saturated models with 98.41% accuracy.

6 Clustering Analysis

KMeans Clustering (PCA-Reduced Data)

- **K = 2**: Purity = 0.8429
- **K = 3**: Purity = **0.9419** (best result)

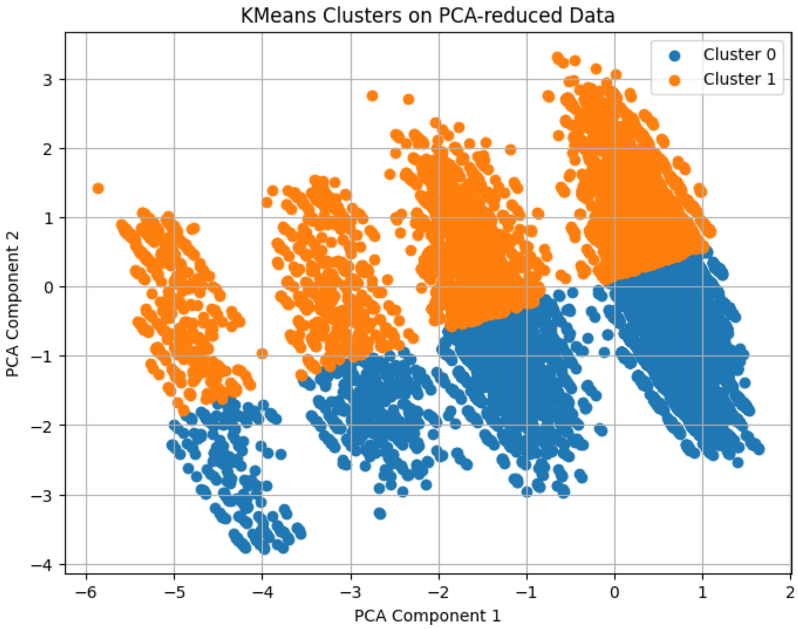

Figure 2: KMeans Clustering (2 Clusters)

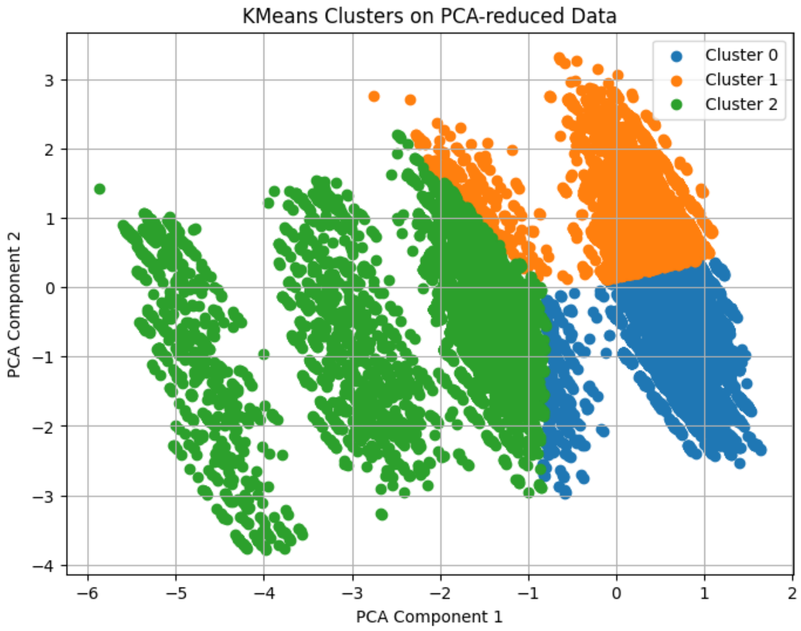

Figure 3: KMeans Clustering (3 Clusters)

DBSCAN Clustering

- Number of clusters = 338 (very high)
- Purity = 0.0438 (extremely poor performance)

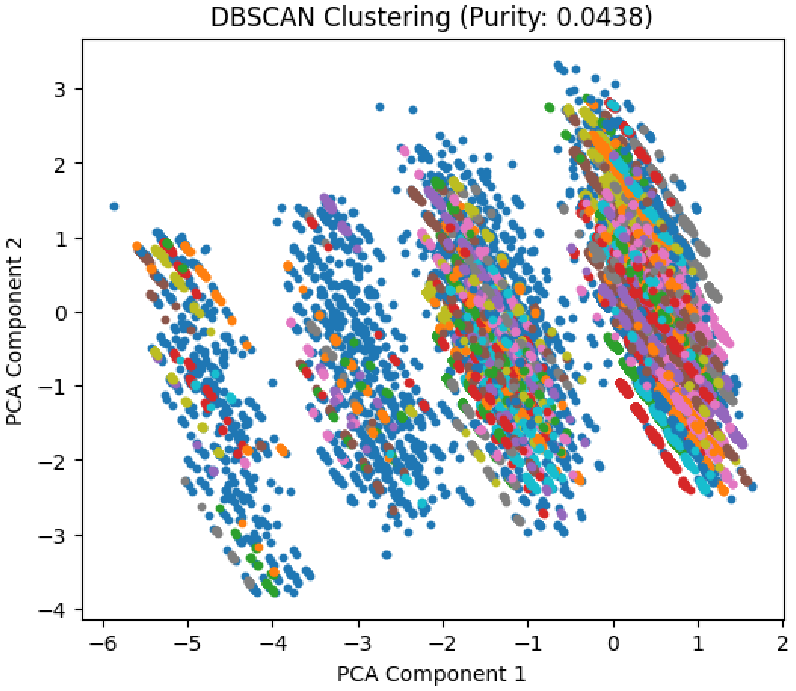

Figure 4: DBSCAN Clustering Results

Conclusion: KMeans is far superior for this dataset. DBSCAN fails due to the lack of dense clusters and high fragmentation.

7 Conclusion

- Ridge and Linear Regression achieve high regression accuracy ($R^2 \approx 0.88$).
- All classification models, except for Linear Regression + Classifier, converge to **98.5% accuracy**.
- With a smaller test size (10%), **100% accuracy** is achieved by most models.
- KMeans with 3 clusters achieves the best clustering purity (0.9419).
- DBSCAN is not suitable for this dataset due to high fragmentation.