Project Report: Student Performance Analysis

Student Name: PUSPARAJ CHAUDHARY

Roll No.: 23AD049

Department: Artificial Intelligence and Data Science

Year: III / V

Date of Submission: 21/10/2025

Abstract

This project aims to explore and analyze the factors influencing student performance in math, reading, and writing. Using a dataset of student scores and demographic information, we performed Exploratory Data Analysis (EDA) and Data Visualization to uncover key patterns and relationships. A Deep Learning model (Multilayer Perceptron - MLP) was implemented to predict math scores based on student attributes. The model was evaluated, and its performance was visualized. The analysis reveals that parental education level and test preparation course completion are significant factors affecting student performance. The model achieved a reasonable predictive accuracy, demonstrating the potential of using demographic data for performance forecasting.

1. Introduction & Objective

Objective: To analyze a student performance dataset, identify key influencing factors through EDA and visualization, and build a predictive deep learning model to estimate student scores.

Dataset: The "StudentsPerformance.csv" dataset contains 1000 student records with 8 features, including demographic information and standardized test scores in math, reading, and writing.

2. Dataset Description

-**Source:** Kaggle (Assumed)

-**Size:** 1000 entries, 8 features

-**Features:**

- **Categorical:** `gender`, `race/ethnicity`, `parental level of education`, `lunch`, `test preparation course`
- **Numerical:** `math score`, `reading score`, `writing score`
- **Basic Stats:** The dataset is clean with no missing values. Score ranges are from 0 to 100.

3. EDA and Preprocessing

Methods Used:

- Checked for and handled missing values (none found).
- Checked for duplicates (none found).
- Encoded categorical variables (e.g., `gender`, `lunch`) using Label Encoding.
- Scaled numerical features (`math score`, `reading score`, `writing score`) using StandardScaler for the model.

Key Insights from EDA:

- The average scores for math, reading, and writing are approximately 66, 69, and 68, respectively.
- Female students tend to outperform male students in reading and writing, while males show a slight edge in math.
- Students who completed the test preparation course generally scored higher.
- Students with standard lunch (vs. free/reduced) performed better.
- Parental education level (e.g., master's degree) is positively correlated with higher student scores

4. Data Visualization

1. Average Scores by Gender:

- What: A bar plot showing the average score for each subject, grouped by gender.

- Why: To visualize the performance gap between genders across subjects.

- Insight: Confirmed that females score higher in reading/writing, while males have a slight lead in math.

2. Score Distribution by Test Preparation Course:

- What: A boxplot of math scores for students who did/did not complete the course.

- Why: To assess the impact of test preparation.

- Insight: Students who completed the course have a higher median score and a tighter score distribution.

3. Correlation Heatmap:

- What: A heatmap showing correlations between numerical scores.

- Why: To understand the relationship between the three test scores.

- Insight: Reading and writing scores are highly correlated (~0.95). Math correlates well with both (~0.8).

4. Parental Education vs. Student Scores:

- What: A grouped bar chart showing average math scores for each parental education level.

- Why: To investigate the influence of parental background.

- Insight: Scores generally increase with the level of parental education.

5. Pairplot of Scores Colored by Gender:

- What: A scatterplot matrix showing the relationship between all three scores.

- Why: To visualize distributions and correlations simultaneously.

- Insight: Reinforces the strong correlation between scores and shows distinct clusters for genders.

5. Deep Learning Model

Architecture: A Multilayer Perceptron (MLP) Regressor was chosen to predict the `math score`.

- Input Layer: 7 features (after encoding and including reading/writing scores as features).

- Hidden Layers: 2 Dense layers with ReLU activation (64 and 32 units).

- Output Layer: 1 neuron with linear activation (for regression).

- Optimizer: Adam

- Loss Function: Mean Squared Error (MSE)

Training Parameters:

- Train/Validation/Test Split: 70%/15%/15%

- Epochs: 100

- Batch Size: 32

6. Result Visualization & Interpretation

1. Loss vs. Epoch Chart:

- What: Shows the training and validation loss decreasing over each epoch.

- Interpretation: The model learned effectively as the loss reduced and converged, with no significant overfitting.

2. Accuracy vs. Epoch Chart (using R^2 Score):

- What: Plots the R^2 score for training and validation sets over epochs.

-Interpretation: The model's explanatory power (R^2) increased and stabilized, reaching a final test R^2 of approximately 0.87, indicating a good fit.

3. Predicted vs. Actual Values Scatter Plot:

- What: A scatter plot comparing the model's predictions to the actual math scores.

- Interpretation: The points align closely to the y=x line, showing that the model's predictions are generally accurate across the score range.

7. Conclusion and Future Scope

Conclusion:

- EDA revealed that `gender`, `lunch` type, `test preparation course`, and `parental education` are significant factors in student performance.

- The scores are highly inter-correlated.

- The MLP model successfully predicted math scores with good accuracy ($R^2 \sim 0.87$), demonstrating the feasibility of this approach.

Future Scope:

- Model Improvement: Experiment with more complex architectures (e.g., CNNs for structured data) or other algorithms like XGBoost.

- Feature Engineering: Create composite scores or interaction terms.

- Additional Data: Include features like school type, student attendance, or more granular parental data.

- Multi-output Model: Predict all three scores simultaneously.

8. References

1. Pedregosa et al., 2011. Scikit-learn: Machine Learning in Python. *JMLR*.

2. McKinney, W., 2010. Data Structures for Statistical Computing in Python. *Proc. of the 9th Python in Science Conf.*

3. Hunter, J.D., 2007. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*.

4. Chollet, F., 2015. Keras. *GitHub repository*.

5. Kaggle. "Students Performance in Exams" Dataset.

Appendix

Code snippet

```
# Model 1: Regression Model to predict average score
print("Building Regression Model...")

def create_regression_model():
    model = keras.Sequential([
        layers.Dense(64, activation='relu', input_shape=(X_train_scaled.shape[1],)),
        layers.Dropout(0.3),
        layers.Dense(32, activation='relu'),
        layers.Dropout(0.2),
        layers.Dense(16, activation='relu'),
        layers.Dense(1)  # Output layer for regression
    ])

    model.compile(
        optimizer='adam',
        loss='mse',
        metrics=['mae']
    )
    return model
```

```
# Model 2: Classification Model to predict performance category
print("Building Classification Model...")

def create_classification_model():
    model = keras.Sequential([
        layers.Dense(64, activation='relu', input_shape=(X_train_scaled.shape[1],)),
        layers.Dropout(0.3),
        layers.Dense(32, activation='relu'),
        layers.Dropout(0.2),
        layers.Dense(16, activation='relu'),
        layers.Dense(len(le_performance.classes_), activation='softmax')  # Output for multi-class
    ])

    model.compile(
        optimizer='adam',
        loss='sparse_categorical_crossentropy',
        metrics=['accuracy']
    )
    return model
# Create models
regression_model = create_regression_model()
classification_model = create_classification_model()
```

The entire project, including this report, the source code, the dataset, and a README file, is available at:

https://github.com/rAjsmoKesnoT90/EDA-Assignment-2