### Chapter 4: Materials and Methods

#### Objective

This chapter outlines the comprehensive methodologies utilized in constructing a neural network to predict academic performance. It covers the end-to-end process from data collection and processing to the deployment of a user-interactive application.

Для ознакомления с данными и их подготовкой для дальнейшего анализа будет использован язык программирования Python и его библиотеки. Выбор языка программирования Python обусловлен высокой производительностью при обработке данных, простотой и большим количеством библиотек для машинного обучения. Python является одним из лучших языков программирования для работы с данными. В данной работе будут использоваться следующие библиотеки Python:

**Python Libraries Used:**

1. Pandas: This library was employed for data manipulation and analysis. It provided efficient data structures such as DataFrames and Series, which are essential for handling and transforming large datasets efficiently and intuitively. Pandas was instrumental in data cleaning, transformation, and the preparation process, allowing for easy handling of missing data, merging of data sources, and time series analysis.
2. NumPy: Known for its powerful array objects and a wide array of mathematical operations, NumPy was used to perform numerical computations. The library's ability to handle large multidimensional arrays and matrices with high performance made it invaluable for data processing and feature engineering in the project.
3. Matplotlib and Seaborn: These libraries were used for data visualization. Matplotlib provided a wide range of plotting capabilities, from histograms to scatter plots, which are crucial for exploratory data analysis. Seaborn, built on top of Matplotlib, added advanced visualization patterns and easier syntax for complex charts like heat maps and violin plots, which facilitated a more in-depth understanding of the data distributions and relationships.
4. Scikit-learn (sklearn): A fundamental tool for implementing machine learning algorithms, sklearn was used for model selection, training, and evaluation. It provided a broad spectrum of standardized algorithms for supervised and unsupervised learning, including tools for model validation, selection, and performance metrics. This library significantly streamlined the process of applying various machine learning techniques, including regression, classification, and clustering.
5. PyTorch: As an advanced machine learning library, PyTorch offered dynamic computation graphs that are vital for building complex neural network architectures. Its flexibility and speed were crucial for experimenting with and refining deep learning models, particularly in handling backpropagation and other optimization algorithms efficiently.

#### 1. Data Collection

В данной работе для подготовки данных будет использована методика, описанная в статье Губина Е.И. «Методика подготовки больших данных для прогнозного анализа» [7].

The dataset for this project was provided by the academic supervisor, sourced from the learning management system of the institution. It includes data elements such as student ID, absence rate, pre-final and final exam scores, and total values for the course "Mathematical Analysis." The dataset was selected based on its relevance to academic performance metrics and its structured format which is conducive for machine learning analysis. Ethical considerations were rigorously maintained by anonymizing student identifiers to prevent any privacy breaches and using the data solely for educational purposes.

#### 2. Data Preprocessing:

The preprocessing of data is a critical step in preparing for the effective application of neural networks. This process includes:

- \*\*Cleaning Data\*\*: Removing duplicate records, correcting erroneous entries, and filling or omitting missing values.

- \*\*Handling Missing Values\*\*: Employing techniques such as imputation or deletion depending on the nature and randomness of the missing data.

- \*\*Normalizing Data\*\*: Scaling features to ensure that the neural network does not bias toward variables with higher magnitude.

- \*\*Encoding Categorical Variables\*\*: Transforming categorical variables into numeric formats using methods like one-hot encoding or label encoding, enabling the neural network to process these as inputs effectively.

#### 4. Model Selection:

For predicting academic performance, a feedforward neural network is chosen due to its simplicity and effectiveness in handling structured data like student records. This choice is justified over recurrent neural networks (RNNs) or convolutional neural networks (CNNs), which are typically more suited for sequential data or image data, respectively. Comparative analysis with other machine learning models, such as decision trees and support vector machines, demonstrated that neural networks provided superior performance in terms of accuracy and the ability to model non-linear relationships in the data.

#### 5. Training the Neural Network:

#### Neural Network Training

The training of the neural network involved:

- \*\*Data Splitting\*\*: Dividing the dataset into training (60%), testing (20%), and validation (20%) sets.

- \*\*Model Training\*\*: Using the sklearn library to manage the training process, including batch processing and iterative learning.

- \*\*Model Evaluation\*\*: Utilization of validation data to refine models and prevent overfitting.

#### 6. Performance Evaluation:

The performance of the neural network was evaluated using the following metrics:

Accuracy: Overall correctness of the model predictions.

ROC Curve: Receiver Operating Characteristic curve to assess the trade-off between sensitivity and specificity.

Confusion Matrix: Visualization of true vs. predicted classifications to identify model strengths and weaknesses.

Data visualization was enhanced with matplotlib and seaborn to present these metrics clearly.

#### 7. Optimization and Tuning:

Challenges encountered during implementation include handling imbalanced data, which can bias the model toward the majority class, and ensuring that the model does not overfit. Limitations of this approach include the potential for biases in the model due to skewed data distributions and the assumption that historical patterns will continue into the future.

Optimization techniques included:

Grid Search: Systematic exploration of multiple combinations of parameters to find the best performing model configuration.

Cross-validation: Used to ensure the model’s effectiveness and robustness across different subsets of the dataset.

Sklearn facilitated these processes by providing comprehensive tools for hyperparameter tuning and model validation.

#### 8. Automation:

The final step involved deploying a Streamlit-based web application that enables real-time prediction of academic performance. Key features of this application included:

User Interface: Simple and intuitive, allowing users to input data and receive predictions instantly.

Interactive Visualizations: Graphs and charts that update in real time based on user input, enhancing the interpretability of the model's predictions.

The application serves as a practical tool for educators to understand and predict student performance trends, aiding in personalized educational support.

This chapter provided a detailed description of each stage in the development of the neural network model, emphasizing the methodological rigor and innovative tools employed to achieve reliable predictions of academic performance.

Recomendations:

The application of neural networks in predicting academic performance has proven to be effective. However, there are areas for improvement, such as incorporating more dynamic features that capture students' behavioral changes over time. Future research could explore the integration of temporal dynamics using RNNs or enhancing model interpretability with techniques such as feature importance analysis.