Multiclass Classification: **Predicting students' academic success and dropouts**

**Participants:**

**Problem Abstract**: Education institutions face challenges in tracking students' progress, identifying at-risk students, and ensuring they receive the necessary interventions to help them graduate and take precautions regarding student dropouts. Accurate predictions on student outcomes, whether they will graduate, drop out, or continue their studies, can empower educational institutions to take proactive measures. By leveraging data, we aim to develop a machine learning model to forecast these outcomes, allowing educators to intervene where necessary and maximize the number of successful graduates.

Predicting student outcomes can offer educational institutions insights into the efficacy of their programs, allow for timely interventions, and drive policy changes.

Enter the potential of predictive analytics powered by machine learning. By harnessing the vast amounts of data that schools and colleges accumulate, ranging from attendance records and exam scores to extracurricular activities and socio-economic indicators, we can create predictive models to forecast student outcomes accurately. Such models can provide insights into whether a student is likely to graduate, drop out, or continue their studies. The beauty of this approach is that it's not just reactive; it's proactive. Educators can anticipate challenges and intervene early instead of waiting for students to falter.

**Project Design & Milestones**:

The dataset was created in a project that aims to reduce academic dropouts and failure in higher education by using machine learning models to identify students at risk at an early stage of their educational path so that strategies to support them can be put into place. The following dataset contains 4424 instances and 36 features. The dataset includes information about student enrollment, academic path, demographics, and social-economic factors. The target variable is ‘Target’ and has three categories (dropout, enrolled, and graduate at the end of the average duration of the course). The problem is formulated as a three category classification task.

**Methodology**:

1. **Data Preprocessing**:
   * Identify any imbalance in the dataset and adopt measures to handle an imbalanced dataset.
   * Handle missing values using imputation or deletion.
   * Convert categorical variables using encoding techniques such as one-hot encoding.
   * Normalize or standardize numerical features.
   * Split the dataset into training (70%), validation (15%), and test sets (15%).
2. **Feature Engineering**:
   * Create new variables based on existing features, which can be used for prediction.
3. **Model Selection**:
   * Experiment with various algorithms: Decision Trees, Random Forests, Gradient Boosting Machines, Neural Networks, etc.
   * Evaluate each model's performance based on accuracy, F1-score, and other essential metrics.
   * Implement hyperparameter tuning for the selected model.
4. **Training & Validation**:
   * Train the chosen model on the training dataset and validate using the validation set.
   * Consider techniques like cross-validation, bagging, boosting and other ensemble techniques for a robust evaluation.
5. **Documentation of learnings**:
   * Document all understandings of the dataset and the results of all experiments as a technical paper.

**Machine Learning Models**:

1. **K-Nearest Neighbours:** A supervised classifier that assumes class based on proximity or likelihood with similar data points.
2. **Naïve Bayes Classifier:** Naïve Bayes Classifier is based on the Bayes theorem, which computes probability scores of a data point belonging to a particular class.
3. **Decision Trees**: Effective for tabular data and offers good interpretability.
4. **Random Forest**: An ensemble of decision trees generally provides higher accuracy and can capture complex relationships.
5. **Gradient Boosting Machines (GBM)**: Boosting technique that can provide high performance but may require more tuning.

**Evaluation Metrics**:

1. Accuracy
2. Confusion Matrix
3. F1-Score (for each class)
4. Precision and Recall (for each class)
5. AUC-ROC Curve

**Resources and related articles**:

**References:**

1. Realinho,Valentim, Vieira Martins,Mónica, Machado,Jorge, and Baptista,Luís. (2021). Predict students' dropout and academic success. UCI Machine Learning Repository. <https://doi.org/10.24432/C5MC89>.
2. Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.
3. Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, 1(1), 81-106.
4. Jesse Read, Bernhard Pfahringer, Geoff Holmes, Eibe Frank, “Classifier Chains for Multi-label Classification,” 2009.

**Resources**

1. <https://scikit-learn.org/stable/modules/multiclass.html> This article discusses multi-learning functionality such as multiclass, multilabel, and multioutput classification and regression.
2. <https://www.projectpro.io/article/multi-class-classification-python-example/547> It is the guide to solving the multiclass classification problem using machine learning.
3. <https://builtin.com/machine-learning/multiclass-classification> This article shows how to handle the imbalanced data for multiclass classification.