




A Bc

daca

-  Quick Submit
-  Quick Submit
-  University of Houston

Document Details

Submission ID

trn:oid:::1:2830630883

Submission Date

Feb 20, 2024, 6:18 PM CST

Download Date

Feb 20, 2024, 6:19 PM CST

File Name

ASP-4.docx

File Size

1.4 MB

10 Pages

1,416 Words

8,265 Characters

How much of this submission has been generated by AI?

0%

of qualifying text in this submission has been determined to be generated by AI.

Caution: Percentage may not indicate academic misconduct. Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Frequently Asked Questions

What does the percentage mean?

The percentage shown in the AI writing detection indicator and in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was generated by AI.

Our testing has found that there is a higher incidence of false positives when the percentage is less than 20. In order to reduce the likelihood of misinterpretation, the AI indicator will display an asterisk for percentages less than 20 to call attention to the fact that the score is less reliable.

However, the final decision on whether any misconduct has occurred rests with the reviewer/instructor. They should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in greater detail according to their school's policies.



How does Turnitin's indicator address false positives?

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be AI-generated will be highlighted blue on the submission text.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

What does 'qualifying text' mean?

Sometimes false positives (incorrectly flagging human-written text as AI-generated), can include lists without a lot of structural variation, text that literally repeats itself, or text that has been paraphrased without developing new ideas. If our indicator shows a higher amount of AI writing in such text, we advise you to take that into consideration when looking at the percentage indicated.

In a longer document with a mix of authentic writing and AI generated text, it can be difficult to exactly determine where the AI writing begins and original writing ends, but our model should give you a reliable guide to start conversations with the submitting student.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify both human and AI-generated text) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Assignment Report

Student Dropout and Academic Success Prediction using Machine Learning

Module Code:

Convenor Name:

Student Name:

Student Number:

Date: 20 Feb, 2024

Actual Hours Spent: 30+ hrs (I work in 2 hour sessions, and take 5 min break every 30 mins. I needed 14 such sessions, and some extra work in between.)

Abstract

This report presents a predictive comprehensive analysis of student dropouts and academic success in higher education institutions via machine learning (ML). Utilizing a dataset

from the UCI Machine Learning Repository, I applied three distinct machine learning models: Logistic Regression, Random Forest Classifier, and a Deep Learning model implemented with TensorFlow. The study aims to proactively identify students at risk of dropping out, allowing for timely intervention strategies.

Background and Problem Addressed

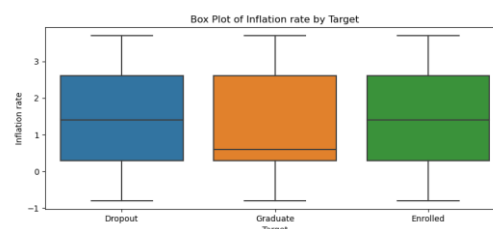
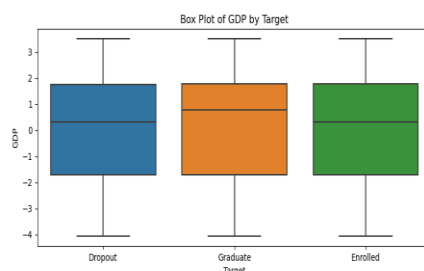
The challenge in the educational sector to predict student dropouts and successes is critical. Proactive identification of at-risk students can lead to timely interventions, improving educational outcomes. At risk students can be offered special lessons or support to improve their chances of getting Graduated. This study leverages machine learning technologies to address these challenges, contributing to the burgeoning field of educational data mining.

Exploratory Data Analysis

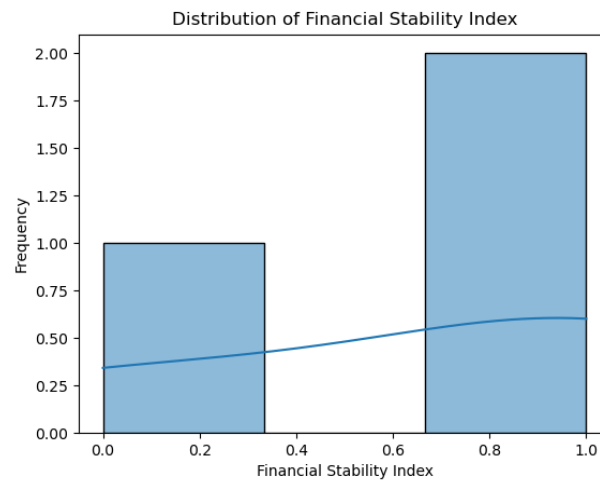
The dataset comprises 4424 instances with 36 features, including demographic, academic, and socio-economic factors. Initial explorations reveal a balanced representation of features like Marital Status, Gender, and others, with insights gained from distribution plots and correlation analyses.



This figure shows the imbalanced nature of our dataset, and as we can see, the Enrolled class has the least example, and our models have a harder time correctly finding this class.

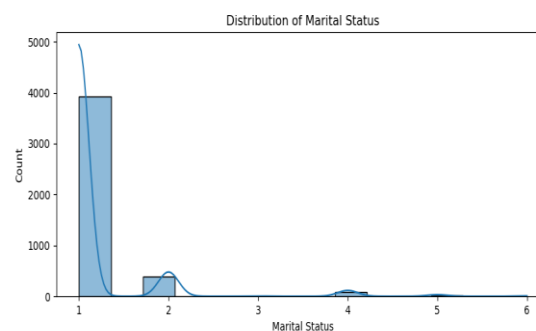
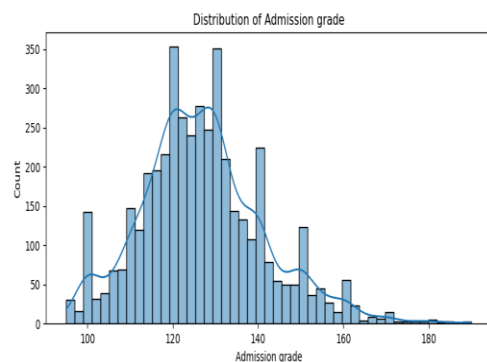
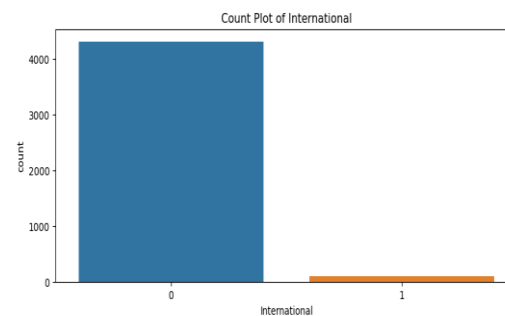
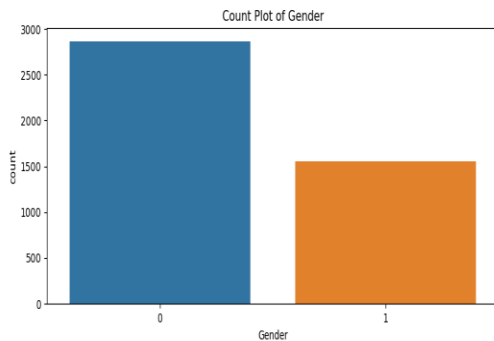


These figures show that current economic factors (GDP, Inflation) affect students' decisions. Then, we developed a feature that condenses these factors. We called it the Financial

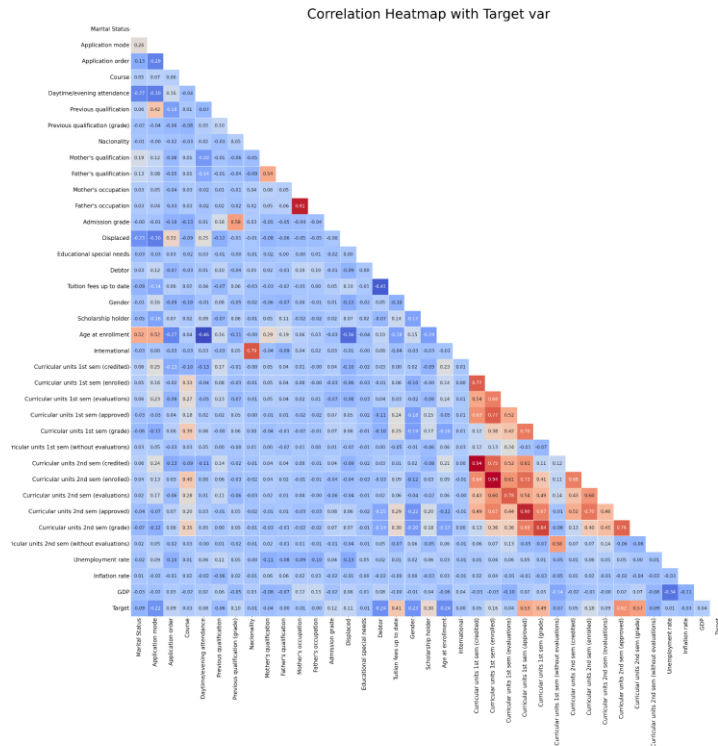


Stability Index.

These figures show the imbalance of gender, international/national students, marital status and the admission grades.



The Heatmap Below shows which factors are Correlated to the target.



Data Pre-processing and Feature Selection

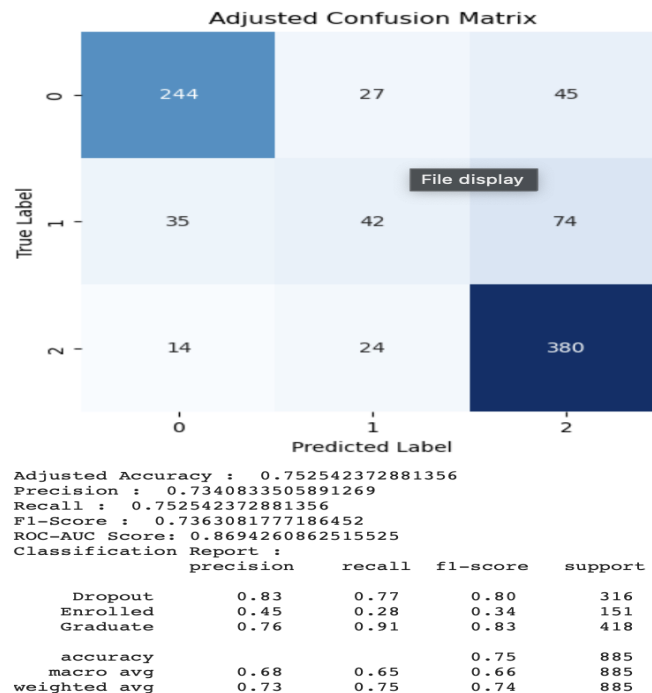
Data preprocessing involved normalization, checking for missing values (there were none in the dataset), and encoding categorical variables. Feature engineering enhanced the dataset, introducing new features like 'Approval Rate' and 'Financial Stability Index' while ensuring the relevance and impact of each feature on the model.

Machine learning models

Model 1: Logistic Regression

- **Summary:** Employed for its interpretability and efficiency, serving as a baseline model.
- **Parameters:** max_iter=500.
- **Rationale for using the Model:** Selected for its simplicity and interpretability.
- **Training and Evaluation:** Achieved an accuracy of 75.25%, with a weighted precision of 73%.

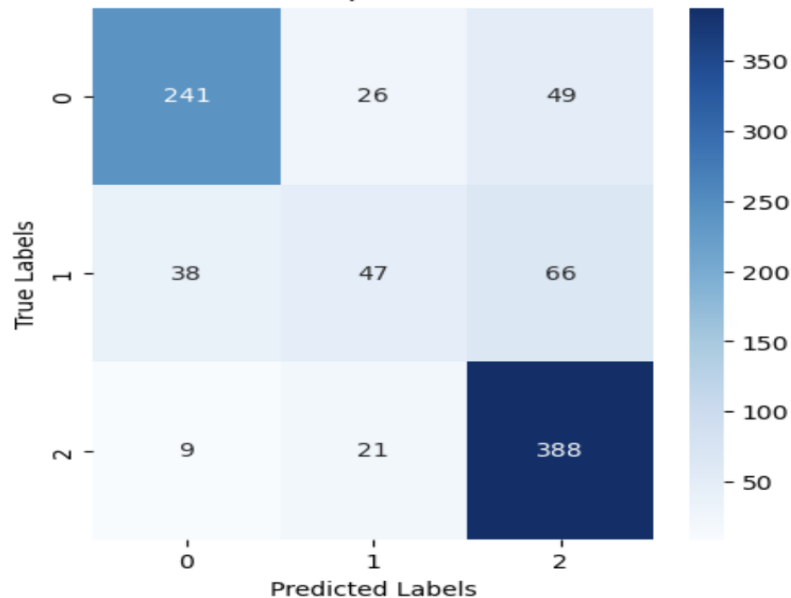
- **Analysis:** Overall balanced performance but slightly lower compared to other 2 models.



Model 2: Random Forest Classifier

- **Summary:** I made a Decision Trees model, and although very easy to build, it showed promising results, so I finally did a RFC model for its robustness against overfitting and ability to capture nonlinear relationships. Results similar to Logistic Regression indicate that the data aren't nonlinear in nature.
- **Parameters:** Used default parameters.
- **Rationale for using the Model:** Robustness against overfitting, nonlinear pattern capturing capabilities.
- **Training and Evaluation:** Achieved the highest ROC-AUC score of 0.8714, indicating superior class differentiation.
- **Analysis:** The best overall performance. Some details here -

Confusion Matrix Heatmap for RandomForestClassifier



```
Random Forest Accuracy: 0.7638418079096045
Random Forest Precision: 0.7484338476360919
Random Forest Recall: 0.7638418079096045
Random Forest F1-Score: 0.7483592948498395
Random Forest ROC-AUC Score: 0.8712373920524056
Random Forest Classification Report:
              precision    recall  f1-score   support

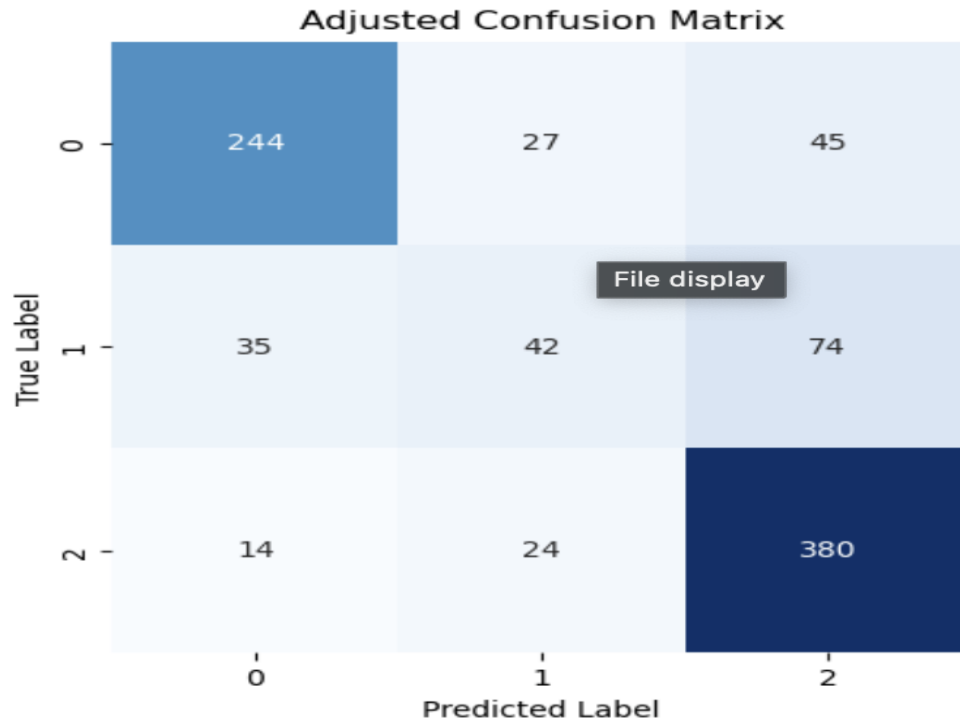
   Dropout              0.84       0.76       0.80         316
   Enrolled              0.50       0.31       0.38         151
   Graduate              0.77       0.93       0.84         418

   accuracy              0.76              885
  macro avg              0.70       0.67       0.67       885
 weighted avg              0.75       0.76       0.75       885
```

Model 3: Deep Learning with TensorFlow

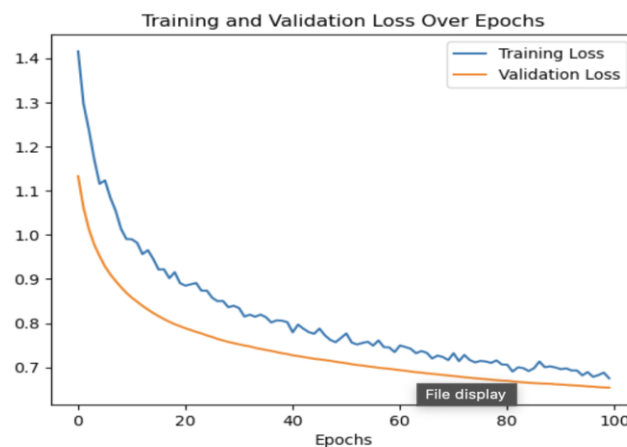
- **Summary:** The model's architecture, focused on scalability, incorporates a sequential layout with multiple layers, each designed to progressively refine the learning from the data.
 - Structure : A Sequential model is employed, a common choice for a stack of layers where each layer has exactly one input tensor and one output tensor.
 - Layers : Dense layers form the core of this model. These layers are fully connected, meaning each neuron in a layer receives input from all neurons of the previous layer, enhancing the model's ability to learn complex patterns.
 - Activation Function : I've used the very common 'relu' (rectified linear unit) activation function. It is known for its efficiency and effectiveness and helps in mitigating the vanishing gradient problem, which is crucial for DL.

- Regularization : I've applied L2 regularization, aiming to prevent overfitting by penalizing large weights. This approach helps the model to learn smaller weights, leading to simpler models that generalize better.
 - Dropout Rate This is to stop overfitting. While training, half of the neurons in a layer are randomly deactivated (dropout rate of 0.5 implies that)
- **Parameters:** Sequential model with Dense layers, relu activation, L2 regularization, and dropout rate of 0.5, Learning rate = 0.0001
- **Rationale for using the Model:** Adopted for its capability to model complex relationships.
- **Training and Evaluation:** The model has shown performance comparable to a Random Forest model, an encouraging sign given Random Forest's reputation for robustness. Achieving a ROC-AUC of 0.8663 is notable. This metric indicates good discriminatory ability.
- **Analysis:** I believe that with some proper tuning, maybe a different architecture, or hyperparameter readjustment could help DL models outperform the other ones. Here are the current results that I could push the model to (so far) -



Adjusted Accuracy : 0.752542372881356
 Precision : 0.7340833505891269
 Recall : 0.752542372881356
 F1-Score : 0.7363081777186452
 ROC-AUC Score: 0.8694260862515525
 Classification Report :

	precision	recall	f1-score	support
Dropout	0.83	0.77	0.80	316
Enrolled	0.45	0.28	0.34	151
Graduate	0.76	0.91	0.83	418
accuracy			0.75	885
macro avg	0.68	0.65	0.66	885
weighted avg	0.73	0.75	0.74	885



Performance Measures and Evaluation Strategies

I've embraced a multifaceted approach to evaluate their performance. One of my go-to tools is the **ROC-AUC Score**, which brilliantly showcases a model's knack for distinguishing between various classes. To get a well-rounded view, I also lean on **Accuracy**, **Precision**, **Recall**, and the **F1-Score**. They provide a comprehensive and balanced view of how well my models are performing. Then there's the **Confusion Matrix** which offers insights into the classification accuracy across different categories.

Results Comparison and Analysis

For this classification problem at hand, I crafted 8 models, each with its own pros and cons. I zeroed in on the top 3 for a more detailed study, finding that these models had a special knack for handling unseen test data. It was a close call, but the Random Forest stood out, not just in ROC-AUC and accuracy but across various metrics.

Deep Learning, with its complex neural networks, hinted at a vast potential yet to be fully tapped. Logistic Regression, though a bit more modest in performance, still held its ground with admirable effectiveness. These 3 models were very close to each other. Check the confusion matrices.

Gradient Boosting Machines (GBM) and Support Vector Machines (SVM) were like the strong, silent types – powerful and accurate. The Naive Bayes Classifier, with its simplicity, was surprisingly quick and accurate, showing promise in certain situations. K-Nearest Neighbors (KNN) and Decision Trees were reliable and straightforward. All of these models show accuracy near 65-76%, and ROC-AUC scores between 0.78 to 0.87.

Conclusion, Recommendations, and Future Work

Throughout this assignment, I've worked with lots of ML models for multiclass classification techniques and it's clear that ML is a powerful tool for predicting student outcomes in educational settings. Each model brought something unique to the table, with the Random Forest Classifier emerging as the most balanced one. For future works -

- **Further Feature Engineering:** Looking ahead, I'm excited to do an in-depth Analysis of feature impacts and model behaviors. We need better Feature Engineering to give the models a sharper edge.
- **Over/Undersampling for Class Imbalance:** *All of our models suffered in predicting the 'Enrolled' category. I have tried weighted variations, but that*

didn't work very well for me. So, In future, I'd try sampling techniques, and explore and compare different techniques like Synthetic Minority Over-sampling Technique (SMOTE) or Adaptive Synthetic Sampling (ADASYN)

- **Model Tuning and Ensemble Methods:** A mix of hyperparameter tuning and Ensemble Methods could be the secret sauce for even better performance.
- **Cross-Validation Strategy:** To strengthen the robustness and interpretability of the models.
- **Ethical Considerations:** It's always important to consider the ethical implications of predictive modeling. I have to ensure fairness and make sure that my model avoids biases that could adversely affect certain student groups before they are put to use in the real world. Future work could involve a thorough assessment of model biases and steps taken to mitigate them.

References

1. Leo, Breiman. (2001). Random Forests. 45(1):5-32. doi: 10.1023/A:1010933404324
2. Lenis, Wong. (2023). Model for the Prediction of Dropout in Higher Education in Peru applying Machine Learning Algorithms: Random Forest, Decision Tree, Neural Network and Support Vector Machine. 116-124. doi: 10.23919/FRUCT58615.2023.10143068
3. Ioanna, Lykourantzou., Ioannis, Giannoukos., Vassilis, Nikolopoulos., George, Mpardis., Vassili, Loumos. (2009). Dropout prediction in e-learning courses through the combination of machine learning techniques. Computer Education, 53(3):950-965. doi: 10.1016/J.COMPEDU.2009.05.010
4. Tahira, Alam., Chowdhury, Farhan, Ahmed., Sabit, Anwar, Zahin., Muhammad, Asif, Hossain, Khan., Maliha, Tashfia, Islam. (2019). An Effective Recursive Technique for Multi-Class Classification and Regression for Imbalanced Data. IEEE Access, 7:127615-127630. doi: 10.1109/ACCESS.2019.2939755

