What Factors Determine Academic Success?

Research Term Project

Rahul Rajeev

08/07/2023

Predictive Analytics

DSC630-T301 (2237-1)

Introduction

As someone who enjoys teaching and academic advising, I am always curious about what factors affect a student's success in school. Does the parent's educational background matter? What about economic standing or geographic location? And do scholarships play a part? These were all questions I asked myself before beginning this project.

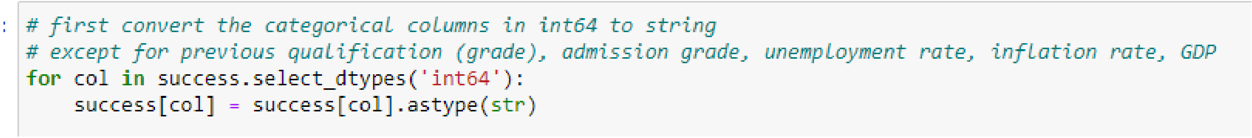
To study the predictability of a student's tendency to dropout, stay enrolled, or graduate successfully, I used a dataset donated to the UC Irvine Machine Learning Repository in 2021, titled "Predict students' dropout and academic success". The premise of my project was to figure out ways to assist the students based on their background. The dataset has about 4424 unique instances (students) and 36 attributes, of which, the target attribute has three categories: dropout, enrolled, and graduate. These attributes include information known at the time of student enrollment up until they graduate.

I would be presenting this project to universities and their academic advising departments as ways to improve graduation rates. While the initial data set is originally based off a Brazilian university and the academic system is much different than ones around the world, I plan on using this project as a basis for testing and modeling for other universities. The data set will demonstrate what features are the most important in determining a student's success whether it be regarding their economic standing, educational background, or the units they are approved for or enrolled in. Additionally, any working model will be interpreted by both me and the respective university to be used for students in need of guidance.

Methodology and Results

Data Preparation

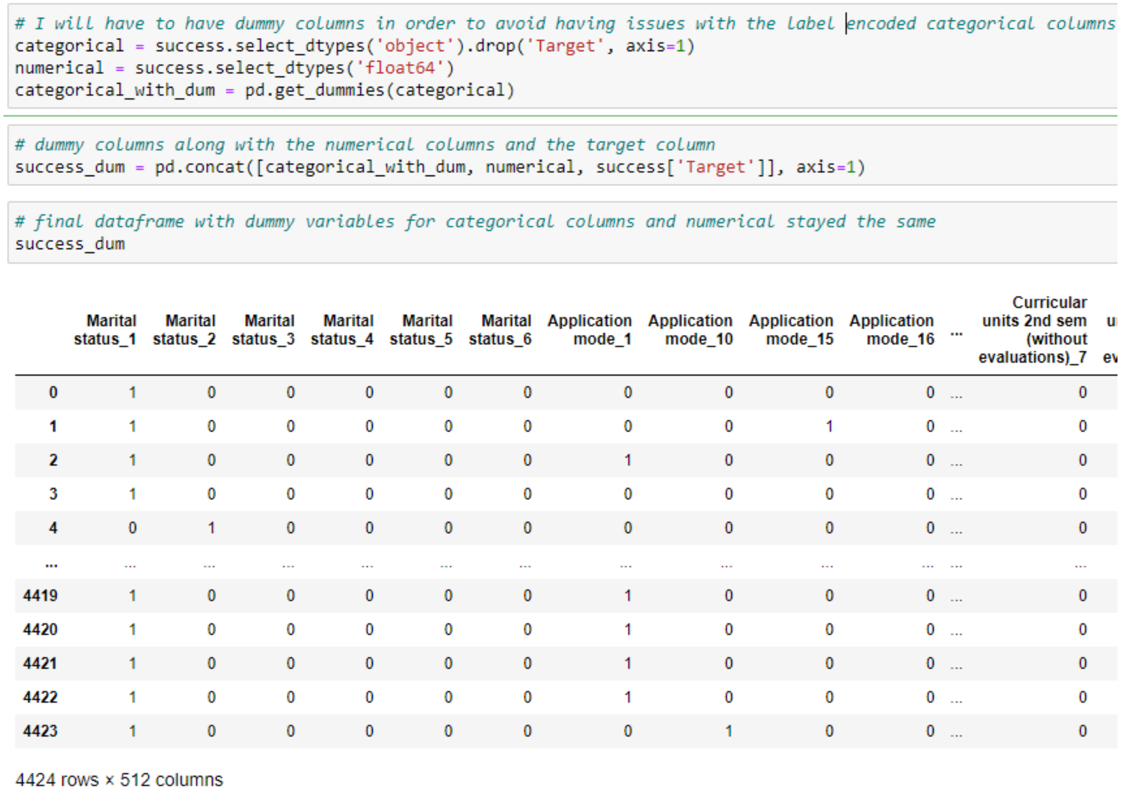
The data has 36 features, all of which represent a unique factor in a student's life, from marital status to age to economic standing. They came in a variety of formats including integer, float64, and object. However, following the data dictionary, I had to adjust the formats of certain features as they were in a label encoded format and mistakenly imported as int64 values. All the values are discrete except for previous qualification (grade), admission grade, unemployment rate, inflation rate, GDP. The last feature is the target, which are three categories in formatted as string.



```
# first convert the categorical columns in int64 to string
# except for previous qualification (grade), admission grade, unemployment rate, inflation rate, GDP
for col in success.select_dtypes('int64'):
    success[col] = success[col].astype(str)
```

Code Snippet 1: Converting Feature Data Types

The first transformation shown above was done to make sure that the data frame has discrete variables in string rather than int64. Next, I had to transform the discrete columns.



```python
# I will have to have dummy columns in order to avoid having issues with the label encoded categorical columns
categorical = success.select_dtypes('object').drop('Target', axis=1)
numerical = success.select_dtypes('float64')
categorical_with_dum = pd.get_dummies(categorical)
```

```python
# dummy columns along with the numerical columns and the target column
success_dum = pd.concat([categorical_with_dum, numerical, success['Target']], axis=1)
```

```python
# final dataframe with dummy variables for categorical columns and numerical stayed the same
success_dum
```

	Marital status_1	Marital status_2	Marital status_3	Marital status_4	Marital status_5	Marital status_6	Application mode_1	Application mode_10	Application mode_15	Application mode_16	...	Curricular units 2nd sem (without evaluations)_7	u ev
0	1	0	0	0	0	0	0	0	0	0	...	0	
1	1	0	0	0	0	0	0	0	1	0	...	0	
2	1	0	0	0	0	0	1	0	0	0	...	0	
3	1	0	0	0	0	0	0	0	0	0	...	0	
4	0	1	0	0	0	0	0	0	0	0	...	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	
4419	1	0	0	0	0	0	1	0	0	0	...	0	
4420	1	0	0	0	0	0	1	0	0	0	...	0	
4421	1	0	0	0	0	0	1	0	0	0	...	0	
4422	1	0	0	0	0	0	1	0	0	0	...	0	
4423	1	0	0	0	0	0	0	1	0	0	...	0	

4424 rows × 512 columns

Code Snippet 2: Final Data frame with Dummy and Numerical Columns

To optimize performance and avoid assumptions about some of the features being cardinal or ordinal, I decided to use pandas get_dummies function to split the categorical features into dummy columns. The final prepped data involves dummy columns for all the categorical features and the unscaled numerical features, which totaled to about 512 columns.

Visualizations

To get an idea of what my data looks like, I created some preliminary visualizations including a stacked bar chart, a tree chart, bivariate plots for the continuous variables, and a box plot. I used the visualizations to make decisions about what features look interesting, and what features must be dropped in the case of multicollinearity.

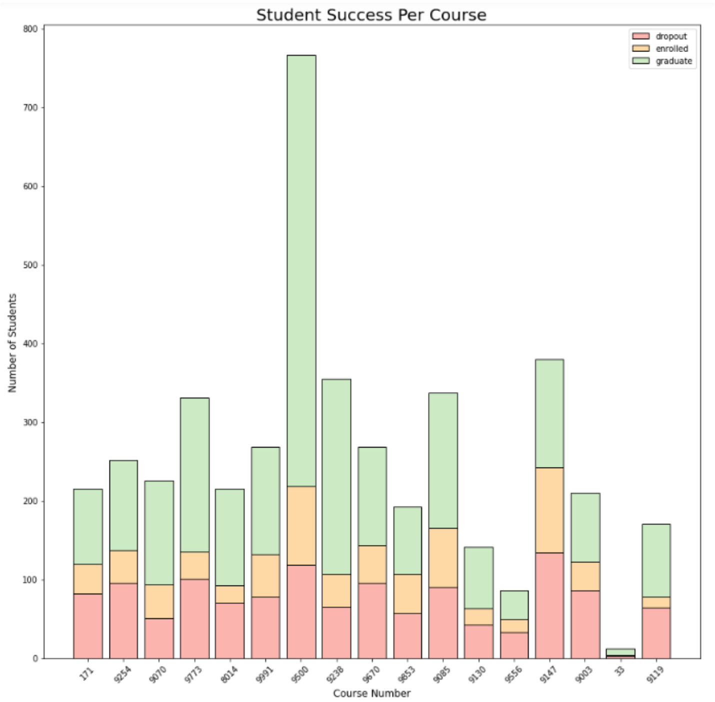


Figure 1: Vertical Stacked Bar Chart (Student Success Per Course)

The first visualization I created was a stacked bar chart describing the student success per course. By using the corresponding course labels (see notebook), I can see that most students enrolled were in the nursing program and the program with the least enrolled was biofuel production technologies. The nursing program also had the largest number of graduates, while the rest of the programs have comparable numbers of students who have dropped out versus graduated. The visualization hinted at the fact that the course program itself did not help determine a student’s success.

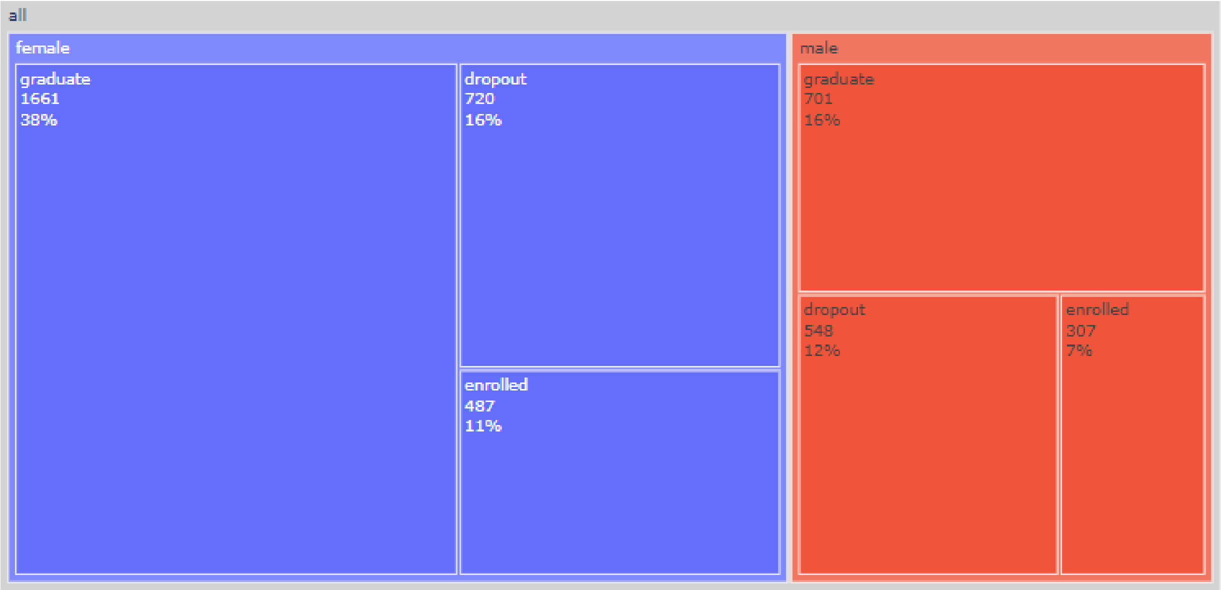


Figure 2: Tree Map

The second visualization I created was a tree map looking at the success of students per gender. Since the nursing program had mostly female students it also makes sense that the university had mostly female students at the time of this study. As far as the other sections of the tree map, it seems similarly distributed across the genders for graduate, dropout, and enrolled, so gender may or may not play a role in academic success.

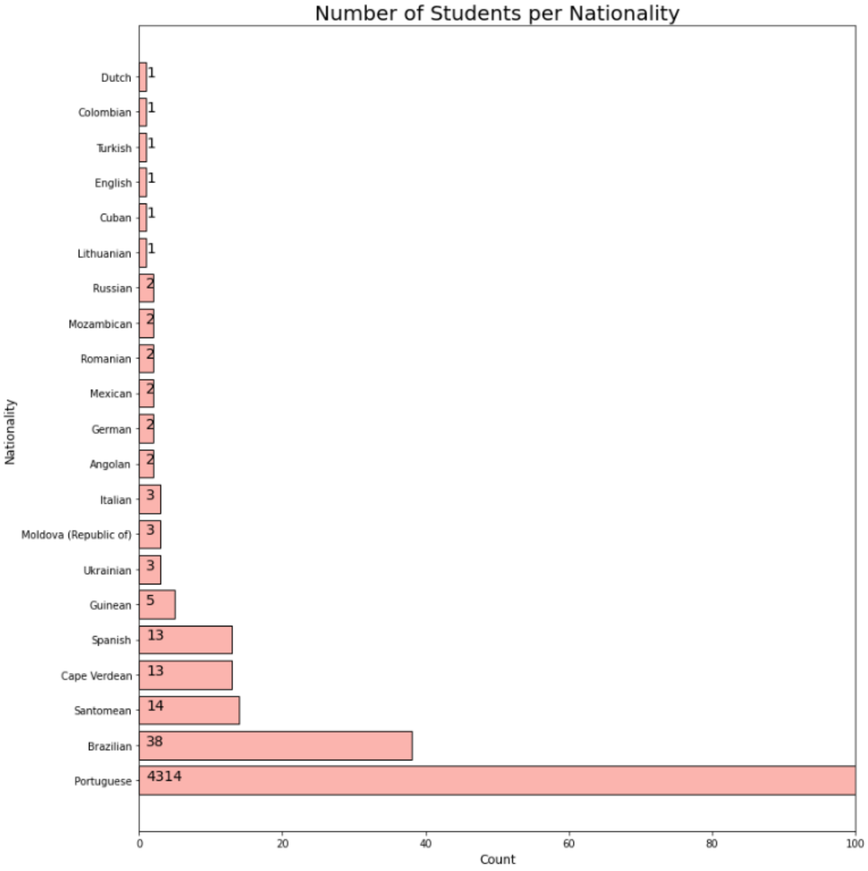
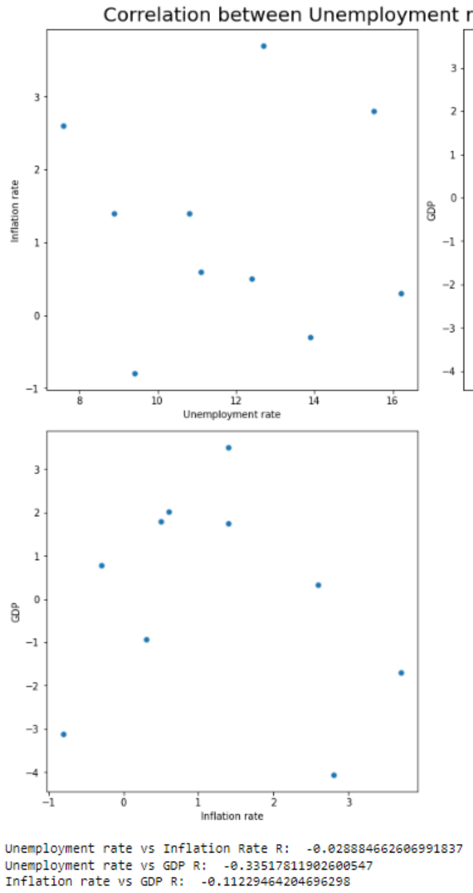


Figure 3: Horizontal Bar Chart

I also created a horizontal bar chart looking at the distribution for nationality and found that most students at the time of this study were Portuguese, followed by Brazilian, and Santomean which makes sense because the university is Instituto Polytechnic de Port Alegre in Portugal. The other nationality numbers exist which could imply that this university also has many international transfer students. Students from different nationalities may find difficulties in getting accustomed to the environment, but it probably won't differentiate between those graduating or dropping out.



```
Unemployment rate vs Inflation Rate R:  -0.028884662606991837
Unemployment rate vs GDP R:  -0.33517811902600547
Inflation rate vs GDP R:  -0.11229464204696298
```

Figure 4: Scatter Plots

The fourth visualization I created was checking for collinearity between unemployment rate, inflation rate and GDP. I found that they weren’t collinear (as evident by the R correlation coefficients) which means it's ok to have them as features in our model. In terms of what they are for, I think it could possibly represent the economic status of students from different areas. Economic standing could be a possible factor in determining academic success, but I had to wait for results to get a better idea.

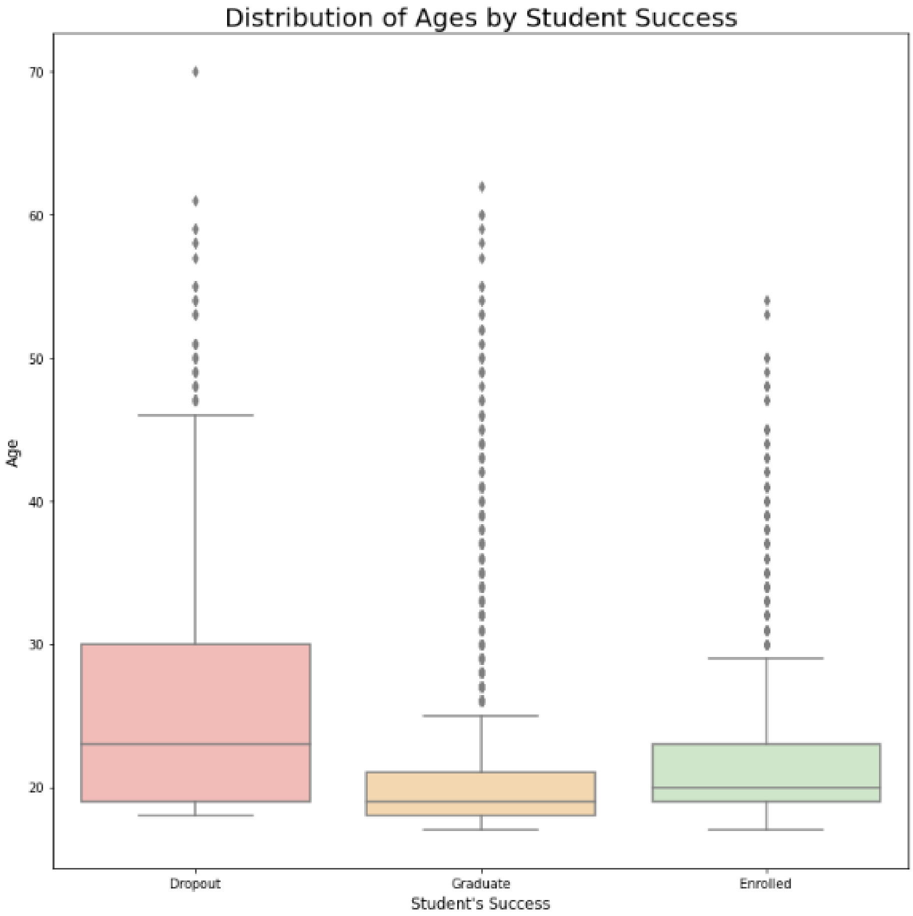
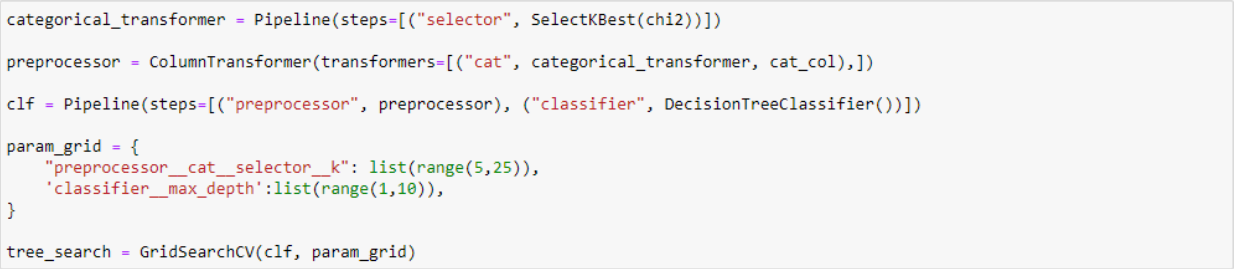


Figure 5: Boxplot

The last visualization I created was a boxplot comparing the distribution of ages for each of the student success categories: dropout, graduate, and enrolled. Many of the students are in the 20-30 range, but there are several outliers evident in the boxplot which show students older than this age range. The university is most likely a commuter school, where students of all ages are welcome to study and graduate. Therefore, students above the age range could have issues with attending the university consistently and consequently be a factor in the model.

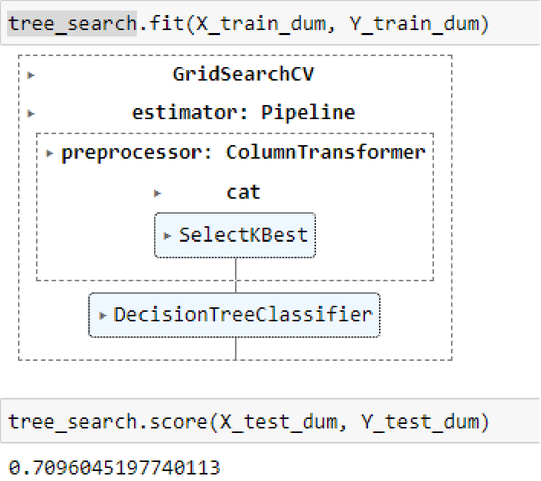
Modeling

I decided to go with a decision tree as my final model. To optimize the hyperparameters of the decision tree and the k value for SelectKBest, I used a grid search and selected the best model. I found that the optimized decision tree had an accuracy of 70.96%. Although the accuracy had gone down compared to initial attempts, I believed that using SelectKBest cut down on the overfitting that may have occurred while modeling the decision tree off label encoded data at first and followed by the complete dummy encoded data set.



```
categorical_transformer = Pipeline(steps=[("selector", SelectKBest(chi2))])

preprocessor = ColumnTransformer(transformers=[("cat", categorical_transformer, cat_col),])

clf = Pipeline(steps=[("preprocessor", preprocessor), ("classifier", DecisionTreeClassifier())])

param_grid = {
    "preprocessor__cat__selector__k": list(range(5,25)),
    'classifier__max_depth':list(range(1,10)),
}

tree_search = GridSearchCV(clf, param_grid)
```

Code Snippet 3: Model with Pipeline and GridSearch



```
tree_search.fit(X_train_dum, Y_train_dum)
```

```
tree_search.score(X_test_dum, Y_test_dum)
```

```
0.7096045197740113
```

Code Snippet 4: Pipeline and Accuracy for Model

The decision tree (not presented) easy to interpret. It reflected paths upon which students mostly regarding the units they enroll in the first or second semester then gets very specific until it reaches a decision of whether the student will dropout, graduate, or stay enrolled. The decision tree itself is quite long and took up 4 pages here on the report, so I would suggest following it on the notebook file because the concatenated pdf form doesn't display it fully. Additionally, I don't believe that the decision tree would be interpretable for the public as it involves specific terminology familiar to the university so I will not be including it in this report.

Results

Since a working model has already been found for the Brazilian university for which this data is based on, my real plan is to start studying data from other universities internationally and see whether I can apply the same working properties of the decision tree model I finalized with to other projects. Before I do that, I still need to interpret the results I got while using this model.

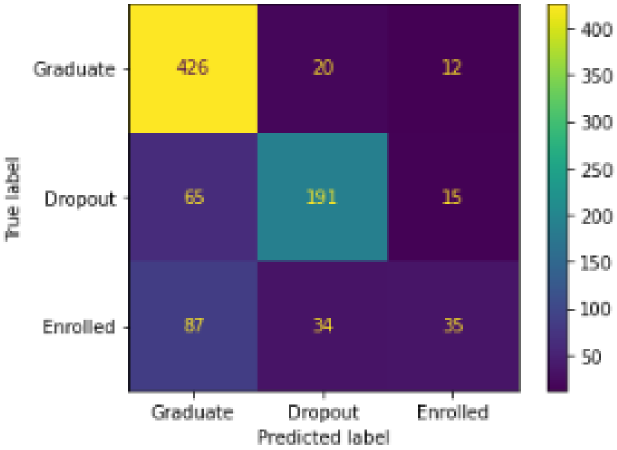
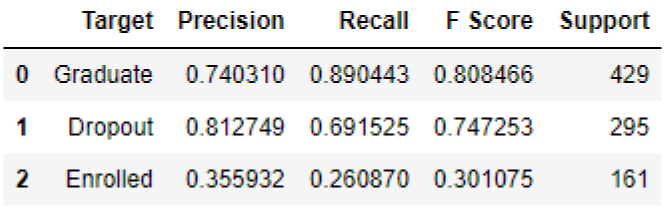


Figure 6: Confusion Matrix

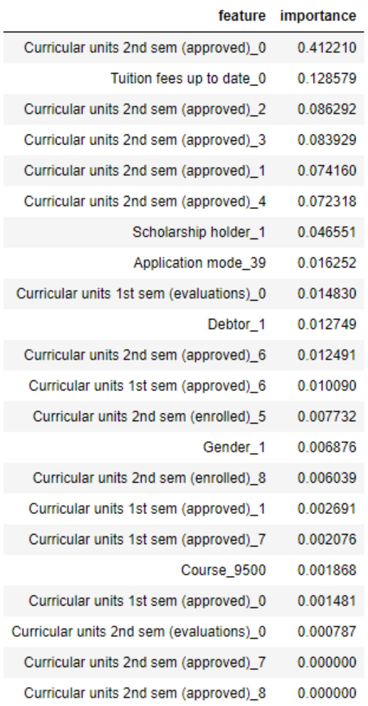
Previous iterations of optimized classification models tended to be more accurate in predicting graduate and dropout students over enrolled students. I think since the data provided only covered students over one academic year it's hard to tell whether the student is planning to stay or not. However, it may be easier to predict whether a student graduates or drops out over a single year based on existing conditions. The confusion matrix highlights this as the graduate prediction is the most accurate, followed by the dropout prediction accuracy.



	Target	Precision	Recall	F Score	Support
0	Graduate	0.740310	0.890443	0.808466	429
1	Dropout	0.812749	0.691525	0.747253	295
2	Enrolled	0.355932	0.260870	0.301075	161

Figure 7: Accuracy Metrics

The next set of accuracy metrics I decided to calculate were precision, recall, f score, and support. If you look at the figure above, the precision and recall of identifying graduate and dropout students were very high. The enrolled students had a very low precision and recall score. This could also be attributed to poor model performance with enrolled students and considering the smaller proportion of enrolled students compared to the other two, enforced by the support metric. The model's accuracy is also reflected in the f score as both graduates and dropouts are predicted well whereas the enrolled are not.



feature	importance
Curricular units 2nd sem (approved)_0	0.412210
Tuition fees up to date_0	0.128579
Curricular units 2nd sem (approved)_2	0.086292
Curricular units 2nd sem (approved)_3	0.083929
Curricular units 2nd sem (approved)_1	0.074160
Curricular units 2nd sem (approved)_4	0.072318
Scholarship holder_1	0.046551
Application mode_39	0.016252
Curricular units 1st sem (evaluations)_0	0.014830
Debtor_1	0.012749
Curricular units 2nd sem (approved)_6	0.012491
Curricular units 1st sem (approved)_6	0.010090
Curricular units 2nd sem (enrolled)_5	0.007732
Gender_1	0.006876
Curricular units 2nd sem (enrolled)_8	0.006039
Curricular units 1st sem (approved)_1	0.002691
Curricular units 1st sem (approved)_7	0.002076
Course_9500	0.001868
Curricular units 1st sem (approved)_0	0.001481
Curricular units 2nd sem (evaluations)_0	0.000787
Curricular units 2nd sem (approved)_7	0.000000
Curricular units 2nd sem (approved)_8	0.000000

Figure 8: Feature Importance

Since I couldn't provide the actual decision tree model to point out each of the major nodes, I decided instead to share feature importance for the features picked after feature selection using SelectKBest. As mentioned, curricular units and their different categorical values appear to dominate most of the data frame here. In terms of what each of the categories represents, I would most likely have to dive into the data dictionary to decipher even a little bit of it. Since the curricular units were defined to be discrete values, I would assume that having a certain number of units being approved for, enrolled, or evaluated can make or break a student's academic progress at a certain point.

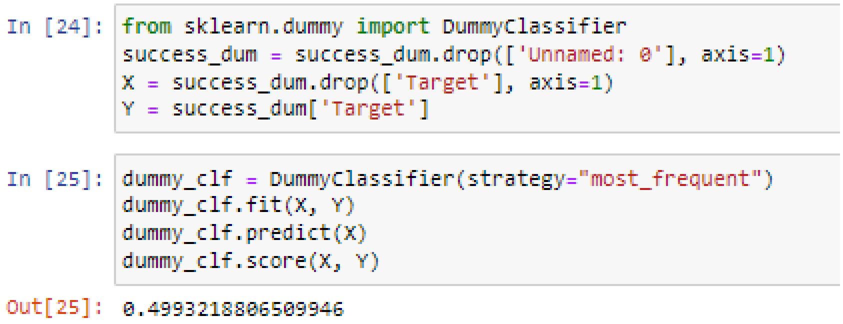
As far as the other more interesting features, tuition fees being up to date are second on the list and 0 meaning that they haven't been paid for. That could explain why a student may have to drop out of their program. Application mode 39 refers to students who are above the age of 23 enrolled in the university. This could play a role as most of them could be commuting students and not having a mode of transportation or finding transportation too costly could lead to some issues. Scholarship holder 1 (meaning the student has a scholarship) and debtor 1 (meaning that the student has debt) are also decently important compared to the features of lower importance and both have valid reasoning as to why they could affect a student.

The last two interesting features include gender 1 (meaning male student) and course 9500 (nursing) also have importance, however, near the bottom of the importance list. However, I can't really think of a good reason why gender or specifically the nursing program would play a role in terms of students becoming graduates, dropouts, or staying enrolled.

Conclusion

I can conclude that my model is a great start in terms of understanding what features are the most important to determine a student's academic success and their status at the end of the year. I learned that a student's educational background takes precedence over personal and economic background when determining their success. I don't know exactly what model was applied to the data since the paper isn't available to view anymore, but I'm assuming if the study was successful that they have found something which is reliable for helping students. If the model is just used to predict whether a student is graduating or dropping out after an academic year, then perhaps it can. Regardless, I believe this information should be used to help students determine whether they will be able to graduate or what additional work they will need to complete and stay enrolled.

Even with the accuracy I have (~71 %) and confidence I have with it being a decently reliable model, I still don't believe it can be applied to live data just yet because of the accuracy with predicting enrolled students. Just to confirm whether this model has reliable accuracy, I decided to use a Dummy Classifier from sklearn to get a baseline accuracy for the data.



```
In [24]: from sklearn.dummy import DummyClassifier
         success_dum = success_dum.drop(['Unnamed: 0'], axis=1)
         X = success_dum.drop(['Target'], axis=1)
         Y = success_dum['Target']

In [25]: dummy_clf = DummyClassifier(strategy="most_frequent")
         dummy_clf.fit(X, Y)
         dummy_clf.predict(X)
         dummy_clf.score(X, Y)

Out[25]: 0.4993218806509946
```

Code Snippet 5: Baseline Accuracy using Dummy Classifier

As seen above, the baseline accuracy by fitting a Dummy Classifier using the “most frequent” strategy, is about 50%. This is a good indicator that my model can be reliable since 71% accuracy is higher than the baseline.

Future Recommendations

For future recommendations I would maybe try to extend the same methodology for creating models using data from other universities and figure out what features are most important for a student’s academic success. For example, if I could access information about the UC system here in California, I could attempt to help provide the academic administrators with insight as to how to assist the student’s best. And since I am more familiar with the schooling system in the United States, I would be able to provide enough background information and assistance on interpreting the model more than I have with the current model.

Additionally, there was a lack of numerical feature dependence in the model. I would assume that for other universities, educational background will remain the most important features for a student’s success, but it would be interesting to see other background features playing a part such as location, parent background, and economic standing.

A final, more ambitious recommendation would be to work on a universal recommendation model for students internationally. While each academic system has unique attributes, I believe there will be a way one day to scale the data accordingly into universal features that can apply for any student and use it to create a decision tree model like this one.

Ethical Considerations

There were ethical considerations considered before starting the project and after receiving the results. Here was the past ethical consideration that ended up not being an issue. Being a former student myself and an advocate for education for all people regardless of background, I admit predicting a student’s success based on their economic background and ethnicity feels ethically wrong. If I were to bring this model to live production, it would receive backlash based on the opinion that everyone would have a unique experience in university. However, now that I have run through preliminary models and understood the implications of the results, I can explain that the model takes educational background as greater importance over anything else.

Presenting the results could be in the form of the physical decision tree as a way for academic advisers to guide students through each branch. They wouldn't necessarily have to show the student what path they may be on or where they are starting from, but it works as a step-by-step method. The ethical implications of having the decision tree could be that the student is confined to a certain set of decisions instead of being free to do what they please. If the student is at risk, it would be for the best. Or the feature importances could be provided instead to point out what areas the academic advising team should focus on for students. The ethical implication here would be that the academic advising team would be unaware of how to work with individuals over groups. This could lead to issues down the road when there are special cases or exceptions with admission. Which can also be considered the third ethical implication as students with unique circumstances would be harder to fit under the decision tree.

To mitigate these concerns, the academic advising team and the university should be well informed of how the model works, what the feature importances entail for groups of students, and how the decision tree is meant to be used. Obviously if there are certain academic measures in discrete/categorical form, the university would be able to understand them better than I would using the data dictionary hopefully provided along with the dataset.

References

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**.**

Garg, R. (2021, October 7). *7 types of classification algorithms*. Analytics India Magazine. https://analyticsindiamag.com/7-types-classification-algorithms/

Ellis, C. (2022, May 30). *When to use multinomial regression*. Crunching the Data. https://crunchingthedata.com/when-to-use-multinomial-regression/