Admissions Project - Asjal Ahmad

# Introduction and description of the competition

The goal of this project is to construct a classification model to determine whether an accepted applicant will decide to attend the University. The model will be based on information collected regarding the University's recently accepted applicants, ranging in entry term from Fall 2017, Fall 2018, Fall 2019, Fall 2020, and Fall 2021. The classification model and insights obtained from it will allow admissions employees to follow up with accepted applicants and increase the yield of accepted to admitted applicants.

# Brief description of the original set

The dataset we will be using in this project contains information on Trinity University's recently accepted applicants, ranging in entry term from Fall 2017 to Fall 2021. The dataset includes 45 variables, including the applicant's ID, entry term, admit type, permanent postal code, permanent country, sex, ethnicity, race, religion, and various other details related to their application.

One important variable in the dataset is "Decision", which indicates whether the applicant accepted or rejected the offer of admission. This variable will serve as the response variable for our classification model.

# Data Cleaning and Feature Engineering

The data cleaning process involves handling missing values, correcting typos or measurement errors, recoding and/or augmenting existing variables, dealing with outliers, transformations, creating new variables, and removing variables if needed.

## Missing Values

There were missing values in some variables, e.g., the Academic.Index and School.1.Top.Percent.in.Class columns. I replaced the missing values in the Academic.Index column with the value 3.0. To fill in the missing values in the School.1.Top.Percent.in.Class column, I imputed them based on the mean value of each group of Academic.Index using the mean function.

## Outliers

There were no outliers in the dataset that needed to be removed. However, there were variables with extreme values that required a transformation.

## Transformations

The School.1.GPA.Recalculated variable was moderately skewed, so I considered transforming it. I used the skewness function to check the skewness and a histogram to visualize the distribution. I decided not to transform it because the distribution was not significantly skewed.

## Creating new variables

I created a new column called School.1.Top.Percent.in.Class, which is the percentage of students in a student's class who scored lower than him or her. I calculated it using the formula 100\*(School.1.Class.Rank..Numeric. / School.1.Class.Size..Numeric.). I also calculated a new score called ACT\_Recalculated from ACT scores. I used the formula ((ACT.English + ACT.Math + ACT.Reading + ACT.Science.Reasoning)/4) to calculate it.

Missing SAT scores were filled in with converted ACT scores using the act\_to\_sat function. In cases where the ACT score was missing as well, missing SAT scores were predicted using a linear regression model based on the relationship between SAT scores and SAT I.CR...M scores.

To recode the Permanent.Geomarket variable, a function called recode\_geomarket was created to recode the variable based on the region of the world. Specifically, geomarkets starting with "INT-" were recoded based on their continent. Oceania was recoded as Asia, as this is the standard practice in many classification systems.

## Removing variables

To streamline the dataset, I eliminated certain variables that were deemed irrelevant or redundant. These included School.1.Class.Size..Numeric., School.1.GPA, School.1.GPA.Scale, ACT.Composite, ACT.Math, ACT.Science.Reasoning, and ACT.Writing. In some cases, these variables were used to compute other variables, while in others, the majority of values were missing. As a result, their removal helped to simplify the dataset and focus on the most relevant categories. I also removed Sport.1.Rating and Sport.1.Sport as they were causing perfect multicollinearity.

In conclusion, I cleaned the dataset by handling missing values, correcting typos or measurement errors, recoding and/or augmenting existing variables, dealing with outliers, transformations, creating new variables, and removing variables if needed. The resulting cleaned dataset is ready for analysis.

## Interaction terms

I added the interaction term SAT\_GPA at the end to capture the combined effect of the recalculated GPA and the SAT scores in the SAT.R.Evidence.Based.Reading.and.Writing.Section...Math.Section column. By multiplying these two variables, I created a new variable that represents the interaction between SAT scores and GPA, which may provide additional information about academic performance. This interaction term could help in exploring the relationship between academic performance and the combined influence of SAT scores and GPA. Additionally, I created the interaction term Academic\_Performance by multiplying the Academic.Index variable with the squared value of the School.1.Top.Percent.in.Class variable. This interaction term considers the quadratic effect of both variables and aims to capture a more nuanced representation of academic performance, accounting for the interaction between the academic index and the student's class rank. These interaction terms can provide valuable insights into the relationship between different factors and academic outcomes.

# Model Comparison and Evaluation

|  |  |  |
| --- | --- | --- |
|  | model\_name | test\_kappa\_scores |
| 1 | Logistic Reg. | 0.5463333 |
| 2 | KNN | 0.3780120 |
| 3 | Simple Clas. Tree | 0.4208681 |
| 4 | Tree Pruning | 0.4163069 |
| 5 | Class. Tree + Bagging | 0.4954846 |
| 6 | Class. Trees + RF | 0.5086026 |
| 7 | Class. Trees + Boosting | 0.5361463 |
| 8 | SVM + Linear. kernel | 0.4531047 |
| 9 | SVM + Poly. kernel | 0.4741636 |
| 10 | SVM + Radial kernel | 0.4866242 |

Figure 1 Model comparison table

## Comparison

1. Performance:
   * The logistic regression model achieves a Kappa score of 0.5463333, indicating a moderate level of agreement between predicted and actual values.
   * Among the other models, the Kappa scores range from 0.3780120 (KNN) to 0.5361463 (Classification Trees + Boosting).
   * Given that the goal is to increase the yield (accepted/admitted), having a logistic model that can predict whether an accepted applicant will attend the University is valuable.
2. Interpretability:
   * Logistic regression provides interpretable coefficients, allowing for the assessment of predictor importance and directionality.
   * Other models like KNN, SVM, and classification trees offer less interpretability since their decision boundaries or predictions are based on complex combinations of features.
   * Logistic regression provides interpretable coefficients, allowing us to assess the importance and directionality of predictors. This interpretability is valuable in understanding the factors that influence an applicant's decision to attend the University.
3. Model Complexity:
   * Logistic regression is a relatively simple model, often requiring fewer computational resources and having fewer hyperparameters to tune.
   * Models like SVM with different kernel functions or ensemble methods like random forests and boosting might have higher complexity and require more computational resources.
4. Model Assumptions:
   * Logistic regression assumes a linear relationship between predictors and the log-odds of the outcome. Violations of this assumption can affect its performance.
   * Other models like KNN, SVM, and classification trees do not make the same linearity assumptions and can capture non-linear relationships.
5. Flexibility:
   * Models like KNN, SVM with different kernels, and classification trees can handle various types of data and accommodate non-linear relationships more naturally than logistic regression.
   * Logistic regression, on the other hand, is limited in capturing complex non-linear relationships unless additional transformations or interactions are included in the model, which they are.
6. Model Training and Implementation:

* Logistic regression is a well-established and widely used statistical model with a long history of research and implementation.
* Other models like KNN, SVM, and ensemble methods require specific parameter tuning and selection of appropriate hyperparameters for optimal performance. Implementation of these models may vary depending on the software or packages used.

1. Robustness to Outliers and Noise:

* Logistic regression can be sensitive to outliers, as it tries to fit a linear decision boundary. Outliers can have a strong influence on the coefficients and affect predictions. In our data, outliers are not significantly present, so logistic model is a reasonable choice.
* Models like KNN and classification trees are generally more robust to outliers and noise, as they make decisions based on the local neighborhood or majority votes, respectively.

1. Handling Imbalanced Data:

* Logistic regression can be sensitive to imbalanced datasets, where the distribution of classes is significantly skewed. It may struggle to capture the minority class effectively, resulting in biased predictions.
* Models like SVM with class weights, ensemble methods with sampling techniques (e.g., bagging or boosting), or specialized algorithms for imbalanced data can better handle imbalanced datasets.

1. Scalability:

* Logistic regression is computationally efficient and can handle large datasets with a moderate number of predictors. It can be trained on high-dimensional data without much concern for computational limitations. This is beneficial when dealing with the admissions process, which typically involves a considerable amount of data.
* Some other models, such as SVM with certain kernel functions or ensemble methods, may have higher computational costs and become more time-consuming as the dataset size or dimensionality increases.

Considering these factors, logistic regression appears to be a suitable choice for constructing a classification model to predict whether an accepted applicant will attend the University. It provides a balance between interpretability, predictive accuracy, computational efficiency, and transparency, which align with the project's goal of increasing the yield of accepted applicants.

## Logistic Model Evaluation

|  |  |  |
| --- | --- | --- |
|  | **predictor** | **value** |
| 1 | AthleteAthlete, Opt Out | -18.1642 |
| 2 | LegacyLegacy, Opt Out | -7.05472 |
| 3 | Decision.PlanEarly Decision II | 4.018445 |
| 4 | Decision.PlanEarly Decision I | 3.66097 |
| 5 | Merit\_Aid\_Group40.5+ | 3.494474 |
| 6 | (Intercept) | 3.350807 |
| 7 | Total.Event.Participation2 or more | 3.095283 |
| 8 | Count.of.Campus.Visits4 or more | 2.378326 |
| 9 | Total.Event.Participation1 | 2.04788 |
| 10 | Merit\_Aid\_Group30.5-35 | 1.981761 |

Figure 2 Top 10 predictors in the logistic model

The coefficients in a logistic regression model provide information on the direction and magnitude of the relationship between each predictor and the log-odds of the outcome variable (in this case, the decision to attend Trinity University). Here are some interpretations of the coefficients for the top 10 predictors:

1. AthleteAthlete, Opt Out: This predictor has the largest negative coefficient, indicating that applicants who are athletes and opt out are much less likely to attend Trinity. This might suggest that being an athlete is an important factor for some applicants when deciding to attend a university.
2. LegacyLegacy, Opt Out: Legacy applicants who opt out also have a negative coefficient, but not as large as the athlete predictor. This suggests that being a legacy applicant (i.e., having a parent or other family member who attended Trinity) might be a factor in the decision to attend, but not as strong as being an athlete.
3. Decision.PlanEarly Decision II: This predictor has a positive coefficient, indicating that applicants who applied through the Early Decision II process are more likely to attend Trinity than those who did not. This might suggest that applicants who are highly committed to attending the university are more likely to be accepted and ultimately attend.
4. Decision.PlanEarly Decision I: This predictor has a positive coefficient, similar to the Early Decision II predictor. This suggests that applying through the Early Decision process, regardless of the specific timeframe, is a strong indicator of the likelihood of attending Trinity.
5. Merit\_Aid\_Group40.5+: This predictor has a positive coefficient, indicating that applicants who received merit aid in the highest range (40.5+ on a scale of 0-60) are more likely to attend Trinity. This suggests that financial incentives might play a role in the decision to attend the university.
6. Intercept: The intercept represents the estimated log-odds of the outcome variable when all other predictors are equal to zero. In this case, the intercept is positive, indicating that on average, applicants are more likely to attend Trinity than not.
7. Total.Event.Participation2 or more: This predictor has a positive coefficient, suggesting that applicants who participated in two or more events (such as campus visits, information sessions, or other outreach activities) are more likely to attend Trinity.
8. Count.of.Campus.Visits4 or more: Similar to the event predictor, applicants who visited the campus four or more times have a positive coefficient. This suggests that active physical engagement with the university might be a strong indicator of likelihood to attend.
9. Total.Event.Participation1: This predictor has a smaller positive coefficient than the 2+ event predictor, but still suggests that participating in at least one event is associated with a higher likelihood of attending.
10. Merit\_Aid\_Group30.5-35: Applicants who received merit aid in the mid-range (30.5-35) have a positive coefficient, but not as large as the higher range. This suggests that financial incentives might be less of a factor for some applicants, or that the specific range of merit aid is less important than simply receiving it.

# Conclusion and Personal Reflection

Reflecting on the challenges I faced during my previous project, I recognized that a significant hurdle was my inadequate time management skills. Determined not to let the same issues plague my new project, I implemented proactive strategies to enhance my time management abilities.

To start, I devised a comprehensive project plan, complete with specific milestones and deadlines. This approach allowed me to break the project into smaller, manageable tasks and monitor my progress effectively. Furthermore, I dedicated specific blocks of time each week to focus solely on the project, committing to prioritize this work and eliminate distractions during these periods. By applying the valuable lessons learned from my prior experience, I successfully completed this new endeavor on schedule and to a high standard.

As I move forward, I am committed to further developing my expertise in data analysis and machine learning while exploring new research areas that pique my curiosity. I also plan to incorporate the insights gained from this project, particularly in terms of effective time management and meticulous evaluation of information sources. Ultimately, I am confident that this project has contributed to my growth as a well-rounded researcher and data analyst. I eagerly anticipate the ways in which these skills will enhance my future work.