Sydney Shearer
Dr. Kim Roth
Data Science and Mathematics Distinction Research
Spring 2022

Predicting Juniata Enrollment Using Random Forest Techniques

0. Abstract

How could one determine whether, once admitted, a student will enroll at Juniata College? This project will investigate how to best predict these admission decisions using random forests, a type of supervised machine learning algorithm used for classification. The data included are variables associated with first-time, first-year applications for students applying for admission to begin in the fall semesters of 2018 through 2021. In this paper a total of nine analyses are compared, created from comparing data from three differently balanced data sets modeled through three different forest methods; these are compared using various measures to find the best model and predictors.

1. Introduction

During the college admission season, graduating high school seniors must choose which institution they will make their home for the next several years. When making this decision, students must choose based on a variety of factors, some of which vary from school to school. Is it possible to determine if an applicant, once they are admitted, will enroll at Juniata College based on variables gathered from the admissions office? This project investigates what type of ensemble supervised learning method is most accurate in predicting this decision, and which variables are seen as important in the most accurate model. Bagging, Random Forest, and Forest PA are explored to determine which model is most accurate in predicting enrollment decisions. The data included are 53 variables associated with first-time, first-year applications for students applying for admission to begin in the fall semesters of 2018 through 2021. The data set includes

7,220 records, 1,655 of which are for applicants who enrolled at Juniata College and 5,565 of which are for applicants who did not enroll at Juniata College. Along with assessing three different models, three varying data sets are considered: the "initial" data set, a data set balanced with over- and under-sampling, and a data set balanced with Synthetic Minority Over-sampling Technique (SMOTE) and under-sampling. Because of this, a total of nine analyses are compared using accuracy, precision, and recall; the highest predictors are then determined from the best model.

2. Literature Review

2.1 Over- and Under-Sampling

As stated above, the data set contains 5,565 records that are classified as 'not enroll' but only 1,655 records that are classified as 'enroll.' This can cause misclassification issues when creating a model, as there are more instances that show what a 'not enroll' record might be and not enough instances that show what an 'enroll' record might be. To combat this, either over-sampling, under-sampling, or a combination can be used to balance the data set. Over-sampling is performed on the minority class, or the class that has fewer records – in this case, the 'enroll' class. It is done by randomly selecting records in the minority class to reuse to 'create' more minority class records. Likewise, under-sampling is performed on the majority class, or the class that has more records – in this case, the 'not enroll' class. It is done by randomly selecting records in the majority class to leave out to 'create' fewer majority class records. For this project, the `ovun.sample` function from the `ROSE` library was used to both over- and under-sample the data set (Lunardon, Menardi, & Torelli, 2014).

2.2 SMOTE

The Synthetic Minority Over-sampling Technique (SMOTE) was introduced by Chawla, Bowyer, Hall, and Kegelmeyer (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). SMOTE was

created to balance a dataset by over-sampling the minority class using synthetic records. The synthetic records are created by looking at the *k*-nearest minority class neighbors, where *k* is chosen by the amount of over-sampling required and neighbors are randomly selected. An example of how synthetic over-sampling is done is provided from the paper written by Chawla, Bowyer, Hall, and Kegelmeyer:

> For instance, if the amount of over-sampling needed is 200%, only two neighbors from the five nearest neighbors are chosen and one sample is generated in the direction of each. Synthetic samples are generated in the following way: Take the difference between the feature vector (sample) between 0 and 1, and add it to the feature vector under consideration. This causes the selection of a random point along the line segment between two specific features. This approach effectively forces the decision region of the minority class to become more general.

This method allows a larger, more diverse sample of minority class records to be created than if only over-sampling on already existing records was used.

Chawla, Bowyer, Hall, and Kegelmeyer (2002) also discuss under-sampling the majority class by randomly choosing records to remove until the minority class is a specific percentage of the majority class. This can be done at various levels so that the classes have either: closer but unequal numbers of records with a larger majority class; closer but unequal numbers of records with a larger minority class; or equal numbers of records. As stated by Chawla, Bowyer, Hall, and Kegelmeyer: "if we under-sample the majority class at 200%, it would mean that the modified dataset will contain twice as many elements from the minority class as from the majority class" (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). This would make the minority class 200% bigger than the majority class. "If the minority class had 50 samples and the majority

class had 200 samples and we under-sample majority at 200%, the majority class would end up having 25 samples [and the minority class would still have 50]" (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). Within the current experiment, I plan to use both SMOTE and under-sampling to create a balanced dataset but will be referring to the combined technique as 'SMOTE'.

2.3 Decision Trees

Decision trees are a supervised machine learning method that provide a prediction for a given observation by separating the data set into regions that can be summarized in a tree-like structure. Trees are built on a 'training' set, or a large portion of the data, and results are checked using a 'testing' set, the smaller portion of the data that were not used to train the model. Classification trees are used to predict a qualitative response, and each prediction is based on the 'most commonly occurring class' in a region. This paper discusses only classification trees and does not include any discussion of regression trees, which are used for quantitative responses. Decision trees are easily interpretable, as they can be visualized in a tree-like structure, but may lack in accuracy and robustness from lack of complexity and overfitting of training data. Several ensemble methods have been created to improve decision tree performance as discussed below; these methods involve the creation of many decision trees in various manners (James, Witten, Hastie, & Tibshirani, 2021).

2.4 Bagging

A first improvement on standard decision trees is bootstrap aggregation, or bagging. A single decision tree is trained on one data set, but ideally, several data sets would exist from which multiple models could be trained. However, multiple data sets typically do not exist for a single problem, so bootstrapping is used to create multiple occurrences of a data set. Bootstrapping is done by taking samples with replacement to create several data sets that are the

same size as the original one. From here, a decision tree model is fit to each bootstrapped data set, and the results are aggregated, typically using a majority voting system for a classification problem, to reach a final decision (James, Witten, Hastie, & Tibshirani, 2021).

2.5 Random Forest

Random forests are another ensemble method that improve upon basic decision trees by combining bagging and random subspace methods (Adnan M. N., 2014). The random subspace method used the full training data set for each tree, but only considers a subset of the attributes at each split, introducing randomness to the model. Random forests are modeled by using a bootstrapped training sample for each tree and considering a random subset of the attributes at each split, introducing another layer of randomness to the model. By bootstrapping and considering only a subset of attributes at each split, the trees that are created are decorrelated, lessening the chance of overfitting to occur. Only considering a random subset of attributes at each split also prevents a very strong predictor, one that would always be chosen first in a method like bagging, from being chosen first every time, which increases the randomness of the trees. From here, a decision tree model in which only some of the attributes are considered at each split is fit to each bootstrapped data set, and the results are aggregated, typically using a majority vote for a classification problem, to reach a final decision (James, Witten, Hastie, & Tibshirani, 2021).

2.6 Forest PA

The Forest by Penalizing Attributes (shortened to Forest PA) algorithm was introduced by Adnan and Islam (Adnan & Islam, 2017). This decision forest algorithm creates a set of decision trees iteratively by using a weighting technique to impose penalties on attributes that were used in pervious trees. Adnan and Islam (2017) show that, when using Ensemble Accuracy (as is defined in Section 2.7) as a metric, Forest PA outperforms other tested decision algorithms

on several data sets. These algorithms include Bagging, Random Subspace, Random Forest, two variants of Random Feature Weights, Extremely Randomized Trees, and Forest CERN (another method created by Adnan and Islam). The Forest PA algorithm is as follows:

1. Generate a bootstrap sample from the training data set.
2. Generate a decision tree from the bootstrap sample using the weights of the attributes.
3. Update weights and gradual weight increment values of the attributes that are present in the latest tree.
4. Update weights of the applicable attributes, using their respective weight increment values, that are not present in the latest tree. (Adnan & Islam, 2017)

Instead of using simple classification capacities (such as Gini Index), merit values are assigned to each attribute by multiplying the classification capacity and a weight (whose default value is 1.0). These weights allow the classification capacity to be increased (>1.0) or decreased (<1.0) and are updated as described in Step 3 above, which only applies to attributes that are used in the previously created tree. For these attributes, the node that they appear at is considered, and a weight is randomly generated from within the calculated Weight-Range using the formula in Image 1. Again, attributes that are not used in the previously created tree retain the same weight as before. After weights are decreased for attributes used in the previously created tree, weights are gradually increased for attributes not used in the previously created tree. This process is done using the formula in Image 2. This allows these previously unused attributes to have a higher likelihood of being chosen for a tree after they have not been used for some time. In this project, Forest PA will be compared to both the Bagging and Random Forest method to see if improvements in accuracy measures exist.

$$WR^{\lambda} = \begin{cases} \left[0.0000, e^{-\frac{1}{\lambda}}\right], & \text{if } \lambda = 1 \\ \left[e^{-\frac{1}{\lambda-1}} + \rho, e^{-\frac{1}{\lambda}}\right], & \text{if } \lambda > 1. \end{cases}$$

Image 1: Formula for the Weight Range in step 3 of Forest PA (Adnan & Islam, 2017)

$$\sigma_i = \frac{1.0 - \omega_i}{(\eta + 1) - \lambda}$$

Image 2: Formula for weight increase in step 4 of Forest PA (Adnan & Islam, 2017)

2.7 Accuracy Measures

When considering the accuracy of a model, three measures are generally used on the testing data: accuracy, precision, and recall (Ping Shung, 2018). These three measures can be calculated from a confusion matrix, as Pictured in Image 3. The confusion matrix is used to show the number of observations that were classified and whether they were classified correctly or incorrectly. The True Positive (TP) cell contains the number of actual positives that were correctly classified as positive, the False Negative (FN) cell contains the number of actual positives that were incorrectly classified as negative, the False Positive (FP) cell contains the number of actual negatives that were incorrectly classified as positive, and the True Negative (TN) cell contains the number of actual negatives that were correctly classified as negative.

		Predicted/Classified	
		Positive	Negative
Actual	Positive	52 (TP)	18 (FN)
	Negative	7 (FP)	23 (TN)

Image 3: Confusion Matrix

Accuracy is the proportion of correctly classified observations over all classified observations, or (TP+TN)/(TP+FN+FP+TN). In the example confusion matrix above, the accuracy would be (52+23)/(52+18+7+23)=75/100=75.00%. Precision is the proportion of correctly classified positive observations over all positively classified observations, or TP/(TP+FP). In the example confusion matrix above, the precision would be 52/(52+7)=52/59=88.14%. Recall is the proportion of correctly classified positive observations over all actual positive observations, or TP/(TN+FN). In the example confusion matrix above, the recall would be 52/(52+18)=52/70=74.29%.

Amasyali and Ersoy (2014) introduced the ideas of both Accuracy of Ensemble and Average Individual Accuracy. When discussing accuracy in general, as described above, it is calculated as the proportion of correctly classified instances over all instances. Accuracy of Ensemble, or Ensemble Accuracy (EA), is calculated as the average of several model's accuracies. Amasyali and Ersoy use a 5*2 cross validation framework to calculate EA. This involves splitting their data set in half, using one half for training and the other half for testing for the first model, and switching the usage of each half for the second model. This is done a total of five times. Average Individual Accuracy (AIA) is calculated as the average of the accuracy of every tree in a given model.

3. Data Overview

The data used for this analysis were housed in Slate, the platform used by the Juniata College Admission's Office for collection of data related to applicants. Data were pulled for students who applied to be admitted in fall cycles between 2018 and 2021 (inclusive) and included Early Action, Early Decision, and Regular Decision applicants. This resulted in 7,220 observations of first-time, first-year applicants to Juniata College. There were 53 predictor variables that are further described in the Appendix; the response variable, "decision," originally

contained eight categories. These eight categories were sorted into ‘enroll’ and ‘not enroll’ to indicate whether the applicant, upon admission, enrolled at Juniata College or withdrew their application at any stage before enrollment. 1,655 applicants were classified as ‘enroll,’ and 5,565 applicants were classified as ‘not enroll.’

4. Data Cleaning and Processes

Below, Image 4 shows a flowchart to visualize data cleaning and processing. A more in-depth description of this process is provided in Appendix 9.1.

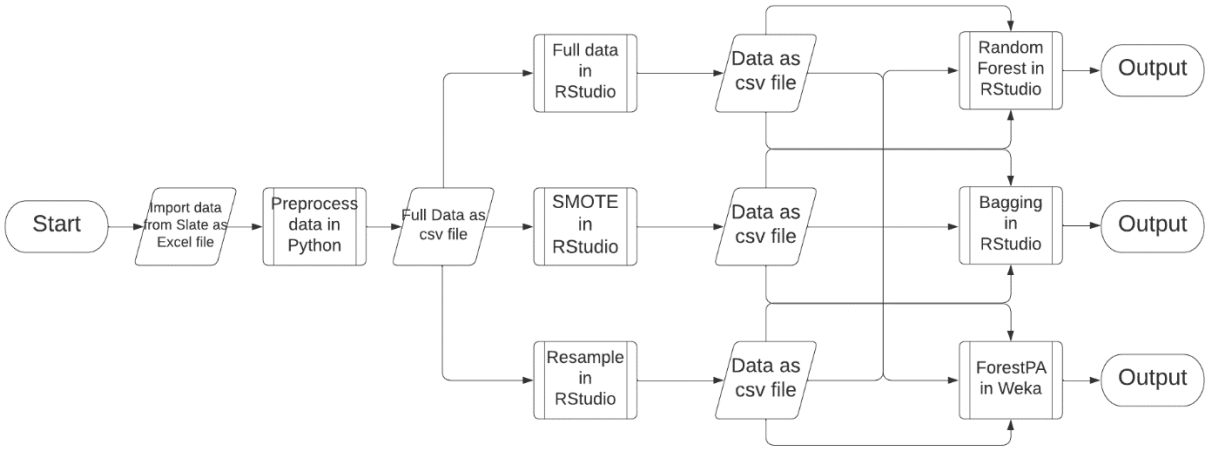

Image 4: Flowchart of process

5. Results

5.1 Forest Comparisons

Below are three tables for various comparison measures between methods. Each row corresponds to a specific data set (Initial, Balanced, or SMOTE) and each column corresponds to a specific forest method (Bagging, Random Forest, or Forest PA). Best results across rows are italicized, best results across columns are in bold, and the overall best result in each table is underlined. Table 1 contains accuracies, Table 2 contains precisions, and Table 3 contains recalls. The results of each table are discussed below them.

Data Set	Bagging	Random Forest	Forest PA
Initial	82.83%	83.32%	*84.78%*
Balanced	**88.81%**	***92.85%***	**89.31%**
SMOTE	84.48%	86.17%	*86.77%*

Table 1: Accuracy for each forest method and data set

Table 1 shows the accuracies for each forest method and data set. When all three models are used, the Balanced data set has the highest accuracy (Bagging, 88.81%; Random Forest, 92.85%; Forest PA, 89.31%). When looking at the Initial and SMOTE data sets, Forest PA outperforms the other methods in accuracy (Initial, 84.78%; SMOTE, 86.77%). When looking at the Balanced data set, Random Forest outperforms the other methods in accuracy (92.85%). Overall, the highest accuracy is achieved by performing Random Forest on the Balanced data set (92.85%).

Data Set	Bagging	Random Forest	Forest PA
Initial	34.78%	32.37%	*44.12%*
Balanced	**83.72%**	***90.34%***	87.66%
SMOTE	75.24%	78.50%	***88.65%***

Table 2: Precision for each forest method and data set

Table 2 shows the precisions for each forest method and data set. When Bagging and Random Forest are used, the Balanced data set has the highest precision (Bagging, 83.72%; Random Forest, 90.34%). When Forest PA is used, the SMOTE data set has the highest precision (88.65%). When looking at the Initial and SMOTE data sets, Forest PA outperforms the other methods in precision (Initial, 44.12%; SMOTE, 88.65%). When looking at the Balanced data set, Random Forest outperforms the other methods in precision (90.34%). Overall, the highest

precision is achieved by performing Random Forest on the Balanced data set (90.34%). Notably, precision for all methods is very poor for the Initial data set.

Data Set	Bagging	Random Forest	Forest PA
Initial	78.26%	*86.45%*	81.42%
Balanced	**93.53%**	***94.92%***	**91.11%**
SMOTE	92.30%	*92.12%*	85.91%

Table 3: Recall for each forest method and data set

Table 3 shows the recalls for each forest method and data set. When all three methods are used, the Balanced data set has the highest recall (Bagging, 93.53%; Random Forest, 94.92%; Forest PA, 91.11%). When looking at all three data sets, Random Forest outperforms the other methods in recall (Initial, 86.45%; Balanced, 94.92%; SMOTE, 92.12%). Overall, the highest recall is achieved by performing Random Forest on the Balanced data set (94.92%).

5.2 Important Variables for Predicting Enrollment

Because it had the highest accuracy, precision, and recall, the Random Forest model with the Balanced Data Set will be used to assess variable importance. The fifteen most important variables were decided using a variable importance plot (Data Camp, n.d.) and are listed in Table 4. The partial dependence plot is shown for each variable; values greater than zero indicate more influence on a prediction of 'not enroll,' and values less than zero indicate more influence on a prediction of 'enroll.' The further a value is from zero, the more influence it has, with values closer to zero not being influential either way.

1.	counselor_rating: Counselor Rating	9.	tuition_discount: Tuition discount
2.	tot_award: Total Award	10.	denom: Denomination
3.	academic_interest_2: Second academic interest	11.	inf_to_app2: Second influence to apply
4.	academic_interest_1: First academic interest	12.	aid_gap: Gap in aid

5.	inf_to_app1: First influence to apply	13.	est_fam_cont: Estimated family contribution
6.	academic_interest_3: Third academic interest	14.	tuition_cost: Cost of tuition
7.	rnl_personicx_life_stage_group: RNL variable	15.	sat: SAT score
8.	rnl_household_income_level: RNL variable		

Table 4: The fifteen most important variables in the Balanced Random Forest analysis

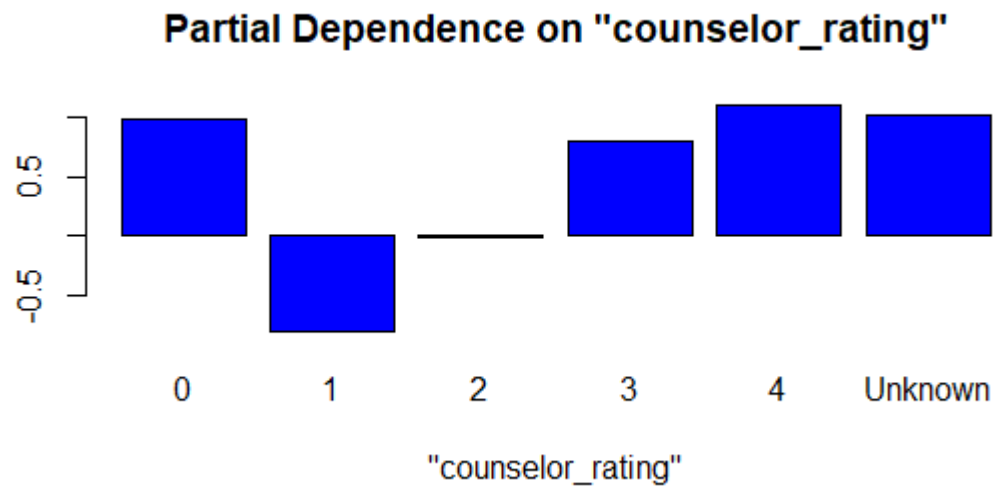

Image 5: Partial Dependence Plot for the Counselor Rating Variable

The most important variable is Counselor Rating (counselor_rating); this variable is assigned to a student after they have met with an admissions counselor at Juniata and is assigned by the admissions counselor. In this variable, 1 indicates “Very likely to enroll,” 2 indicates “Between JC and 1 other,” 3 indicates “Between JC and several others,” 4 indicates “Unlikely to enroll,” and 0 indicates “Unable to rate.” Applicants that are ranked 0, 3, 4, or Unknown are more likely to not enroll (as show by values greater than zero), applicants that are ranked 1 are more likely to enroll (as shown by a value less than zero), and applicants that are ranked 2 could either enroll or not enroll (as shown by a value close to zero).

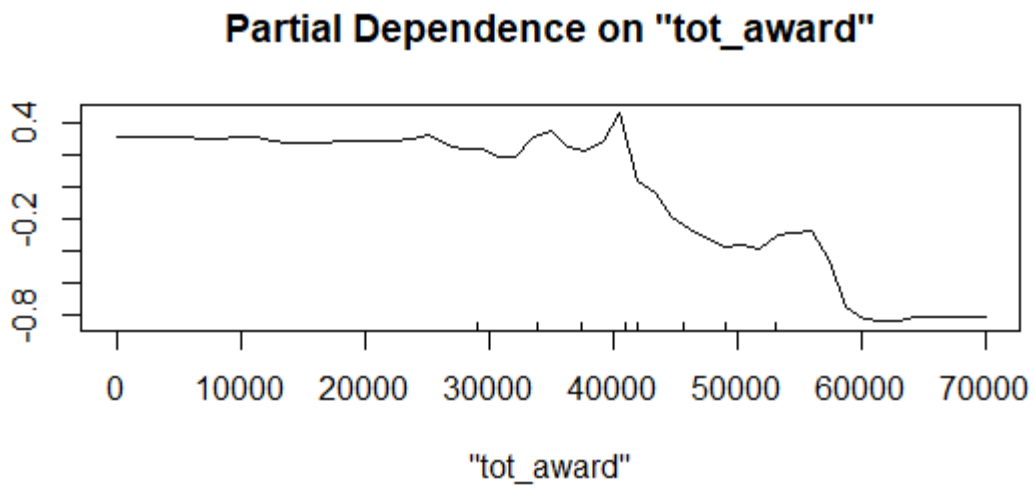

Image 6: Partial Dependence Plot for the Total Award Variable

Total award (tot_award) is the second most important variable and is the total amount of federal, state, and Juniata scholarships/grants that an applicant receives. A total award of $40,000 is chose as a 'cut off' point, since a high spike is seen here in the graph followed by a decrease in y-axis values. This tells us that applicants who receive an award lower than approximately $40,000 are more likely to not enroll (as shown by values greater than zero), and applicants who receive an award higher than approximately $40,000 are more likely to enroll (as shown by values less than zero), with the chance of enrollment increasing as the award amount increases (as shown by values getting further away from zero).

The third, fourth, and sixth most important variables are related to the applicant's academic interest (academic_interest_2, academic_interest_1, and academic_interest_3). There are 30 categories in this variable ranging the fields of study offered at Juniata. For Academic Interest 2, applicants in all categories are more likely to not enroll (as shown by values greater than zero), with applicants in Anthropology and Religion being the most likely to not enroll (as

shown by values further from zero) and applicants in Environmental Sciences, Data Science, and Social Work/Criminal Justice/Sociology being more uncertain (as shown by values closer to zero). For Academic Interest 1, applicants in all categories are more likely to not enroll (as shown by values greater than zero), with applicants in Anthropology, Music, and Religion being the most likely to not enroll (as shown by values further from zero) and applicants in Communication, Geology, and Unknown being more uncertain (as show by values closer to zero). For Academic Interest 3, applicants in all categories are more likely to not enroll (as show by values greater than zero), with applicants in Anthropology and Science being the most likely to not enroll (a shown by values further from zero) and applicants in Art and Unknown being more uncertain (as shown by values closer to zero). Partial dependence plots for these variables have been omitted for unreadability.

The fifth and eleventh most important variables are related to an applicant's influence to apply (inf_to_app1 and inf_to_app2). There are 21 categories in these variables that indicate who or what influenced the applicant to apply to Juniata College. For Influence to Apply 1, applicants in all categories except Family are more likely to not enroll (as shown by values greater than zero), with College Fair being more likely to not enroll (as shown by values further from zero) and Visit to Campus, Coach, Internet Research, and Juniata Alumni being more uncertain (as shown by values closer to zero). Family indicates a slight likelihood to enroll (as shown by values less than but close to zero). For Influence to Apply 2, applicants in all categories except Juniata Alumni are more likely to not enroll (as shown by values greater than zero), with Church Groups and Fiske Guide being more likely to not enroll (as shown by values further from zero) and Coach, Family, and Visit to Campus being more uncertain (as shown by values closer to zero). Juniata Alumni indicates a very slight likelihood to enroll (as show by

values less than but close to zero). Partial dependence plots for these variables have been omitted for unreadability.

The seventh and eighth most important variables are assigned by Ruffalo Noel Levitz (RNL) based on applicant demographics (rnl_personicx_life_stage_group and rnl_household_income_level). Groups shown in the graphs below with higher y-axis values are more likely to indicate not enrolling, and groups shown in the graphs below with y-axis values closer to zero are more uncertain. It is notable that there are certain peaks and valleys in the graph that may be related to cutoffs for need-based financial aid; this could be a place for future research. Partial dependence plots for these variables have been omitted for unreadability.

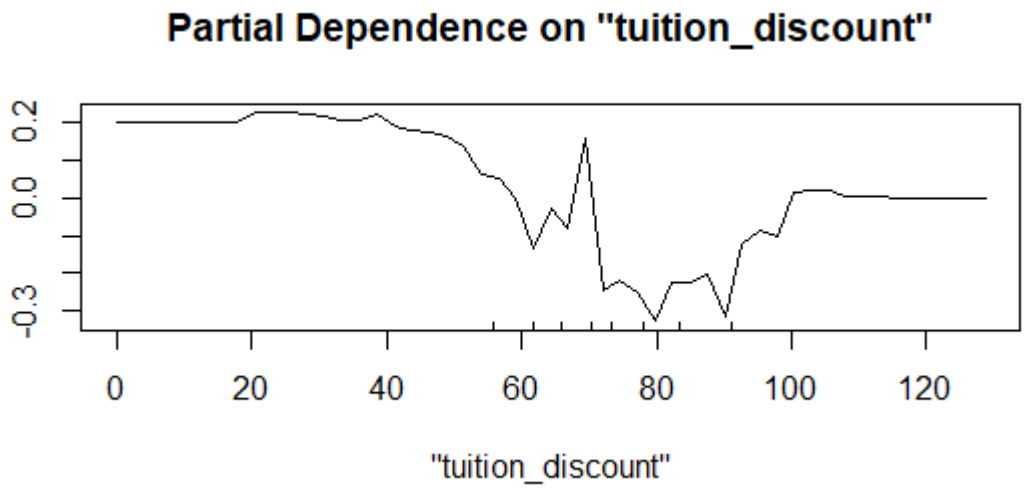

Image 14: Partial Dependence Plot for the Tuition Discount Variable

The ninth most important variable is tuition discount (tuition_discount), or the percentage of Juniata aid that discounts net charged tuition. Applicants who receive a tuition discount lower than approximately 60% are more likely to not enroll (as shown by values greater than zero), applicants between 60-70% are uncertain (as shown by values close to zero), and applicants who

receive a tuition discount higher than approximately 70% are more likely to enroll (as shown by values less than zero), with the chance of enrollment leveling off as uncertain after a 100% tuition discount (as shown by values close to zero). It is unclear why there are values greater than 100% on the x-axis; this could be an area for future research and improvement.

The tenth most important variable is denomination (denom). There are 32 categories in this variable that indicate what denomination the applicant follows. Applicants in all categories are more likely to not enroll (as shown by values greater than zero), with Aglican/Episcopal, Methodist, and Seventh Day Adventist being more likely to not enroll (as shown by values further from zero) and Lutheran and Pentacostal being more uncertain (as shown by values close to zero). The partial dependence plot for this variable has been omitted for unreadability.

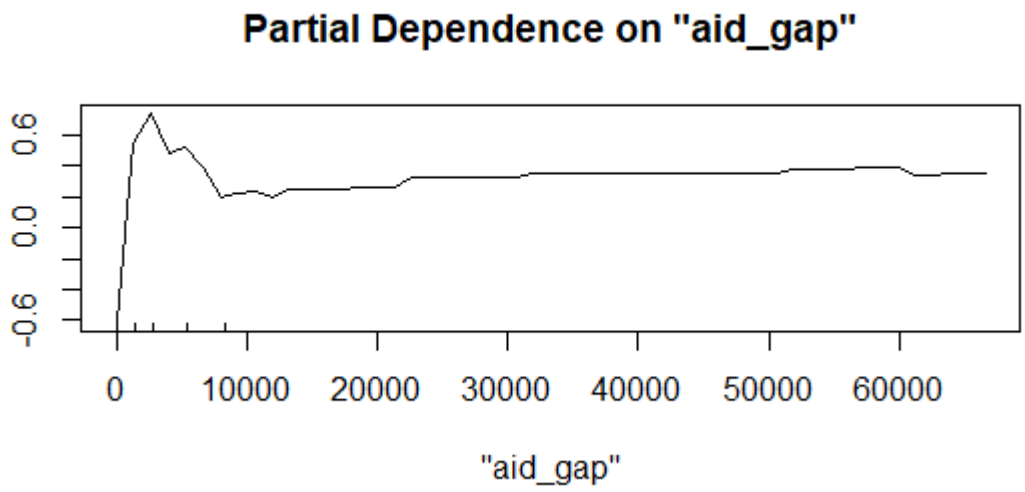

Image 16: Partial Dependence Plot for the Aid Gap Variable

The twelfth most important variable is the applicant's gap in aid (aid_gap), as determined by the FAFSA. Applicants whose aid gap is lower than $10,000 are more likely to not enroll (as

shown by values greater than zero), and applicants whose aid gap is higher than approximately $10,000 are more uncertain (as shown by values close to zero).

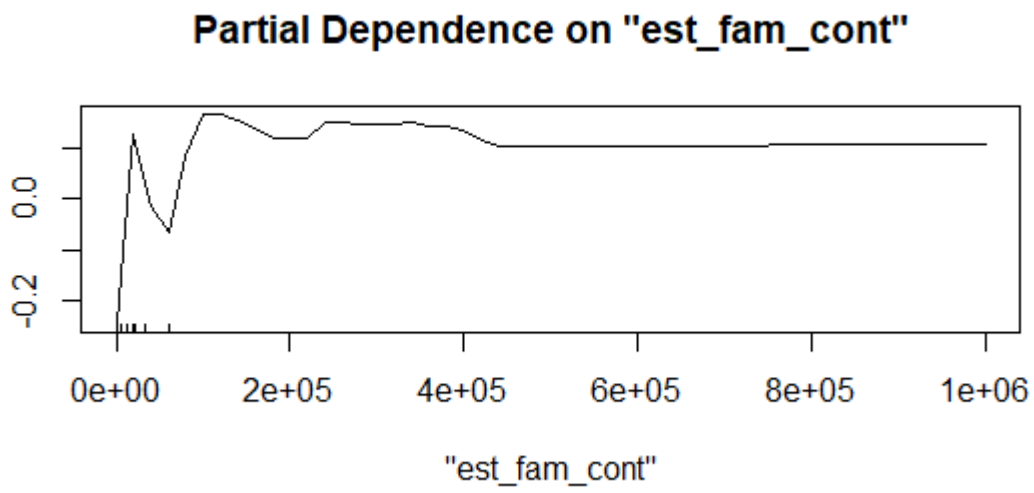

Image 17: Partial Dependence Plot for the Estimated Family Contribution Variable

The thirteenth most important variable is the applicant's estimated family contribution (est_fam_cont). Applicants whose estimated family contribution is lower than approximately $10,000 are more likely to enroll (as shown by values less than zero), and applicants whose estimated family contribution is higher than approximately $10,000 are more uncertain (as shown by values close to zero).

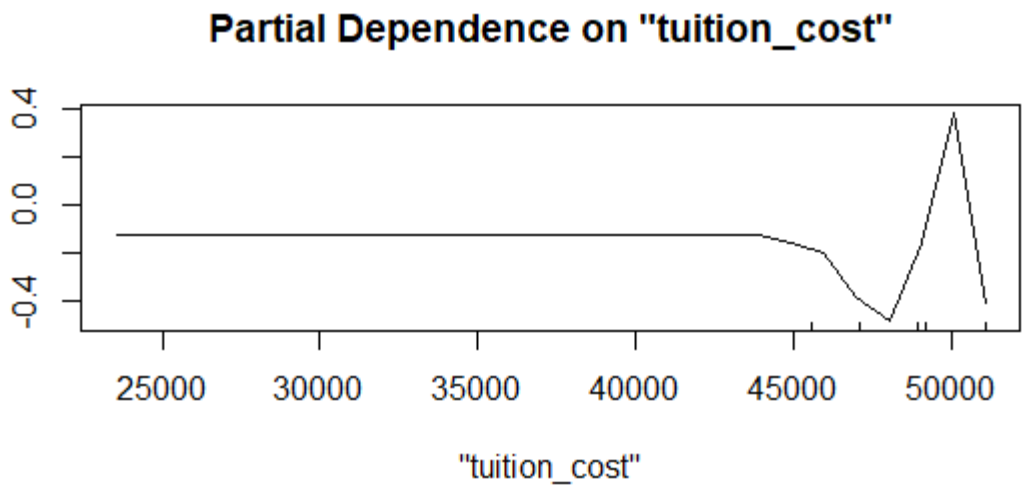

Image 18: Partial Dependence Plot for the Tuition Cost Variable

The fourteenth most important variable is the cost of tuition and mandatory fees (tuition_cost). Applicants whose tuition cost is lower than approximately $45,000 are more uncertain (as shown by values close to zero), and applicants whose tuition cost is higher than approximately $45,000 are more likely to enroll (as shown by values less than zero), with the exception of an odd prediction of not enrolling at approximately $50,000 (as shown by values greater than zero).

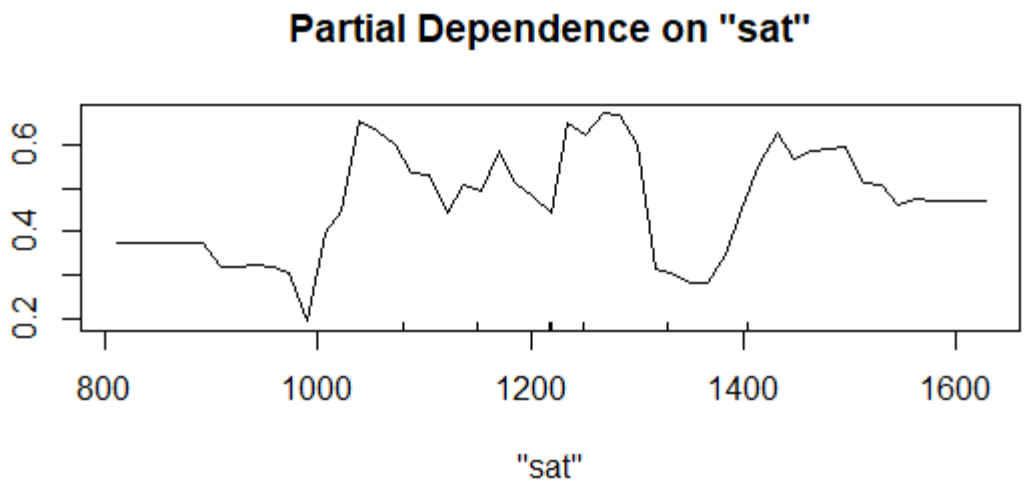

Image 19: Partial Dependence Plot for the SAT Variable

The fifteenth most important variable is the applicant's SAT score (sat). All applicants are more likely to not enroll (as shown by values greater than zero), with SAT scores between 1000-1300 and above 1400 being more likely to not enroll (as shown by values further from zero) and SAT scores less than 1000 and between 1300-144 being more uncertain (as shown by values close to zero).

6. Conclusion

Overall, variable importance results that were achieved from the model were not surprising retrospectively – students will typically choose to attend a college where they receive adequate financial support, where there are programs that interest them, and where they have a preexisting connection. The largest finding from this study, besides the understanding of further data sampling and supervised machine learning methods, was that Juniata College Admissions counselors are able to accurately guess whether applicants will enroll.

6.1 Future Directions

If there were more time to work on this project, models would have been re-run using subsets of variables, including running a model that does not include Counselor Rating. By re-running models in this fashion, only variables that were directly related to the applicant would be included, hopefully providing more predictive power and further actions that could be taken to increase enrollment. On top of this, some variables, such as SAT, had many values that were interpolated. These variables could cause errors in the data set, as much of the data were originally missing, and using the median to fill missing values could cause errors in prediction. A model should be run in the future without these heavily interpolated variables.

When balancing data, future work could be done to remove testing data before balancing the data through either over-/under-sampling or SMOTE. This would guarantee that identical records were not included in both the training and testing data, allowing for more certainty of predictive power of the model.

After contacting the creators of the Forest PA code, variable selection could hopefully be done for the Forest PA model in Weka, allowing for more understanding of the models created with that method.

More research could be done concerning partial dependence plots and how to interpret them to best understand how variables are used within the models.

7. References

Adnan, M. N. (2014). On Dynamic Selection of Subspace for Random Forest. *Advanced Data Mining and Applications,* 370-379. Retrieved from https://link.springer.com/chapter/10.1007%2F978-3-319-14717-8_29

Adnan, M. N., & Islam, M. Z. (2017). Forest PA: Constructing a decision forest by penalizing attributes used in previous trees. *Expert Systems with Applications, 89,* 389-403. doi:10.1016/j.eswa.2017.08.002

Breiman, L. (2001). Random Forests. *Machine Learning, 45*(1), 5-32.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321-357. doi:10.1613/jair.953

Data Camp. (n.d.). *varImpPlot: Variable Importance Plot*, 4.7-1. Retrieved from RDocumentation: https://www.rdocumentation.org/packages/randomForest/versions/4.7-1/topics/varImpPlot

Islam, M. Z. (2020, July 22). *ForestPA*. Retrieved from Github: https://github.com/zislam/ForestPA/

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning with Applications in R* (2 ed.). Springer.

Lunardon, N., Menardi, G., & Torelli, N. (2014). ROSE: a Package for Binary Imbalanced Learning. *R Journal*(6), 82-92.

pandas development team. (2022, February 12). *pandas documentation*, 1.4.1. Retrieved from pandas: https://pandas.pydata.org/docs/

Ping Shung, K. (2018, March 15). *Accuracy, Precision, Recall, or F1?* Retrieved from Towards Data Science: https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9

The R Foundation. (n.d.). *About R*. Retrieved from The R Project for Statistical Consulting: https://www.r-project.org/

weka. (n.d.). *Weka Homepage*. Retrieved from weka: https://www.weka.io/

Welcome to Slate.org! (n.d.). Retrieved from Slate.org: https://slate.org/

8. Acknowledgements

- Dr. Kim Roth for advising and assisting and for reviewing my thesis for distinction
- Juniata College Admissions Office for permission to use their data and assistance in interpreting the variables
- Dr. Matthew Powell (Director of Institutional Research) for assistance collecting data
- Elliot Hirshon for assistance with language variables
- Dr. Melissa Innerst for reviewing my thesis for distinction

9. Appendix

9.1 Data Cleaning and Processes

9.1.1 Acquiring Data

10,257 observations of 324 variables were gathered from Slate, a platform used by Juniata College's Office of Admission (Welcome to Slate.org!, n.d.), with the assistance of Juniata College's Director of Institutional Research.

9.1.2 Data Preprocessing

The three Excel files collected from Slate were then imported into a Python Jupyter notebook for preprocessing. The `pandas` package (pandas development team, 2022) was used throughout data preprocessing. The files were combined into one `pandas` data frame. Several hundred variables were removed after manual inspection; these variables were removed either because they were deemed unnecessary or were missing values for more than half of the records. A total of 54 variables were included in the final data set, 53 of which came directly from the gathered data and one of which (days_taken) was calculated from two original variables. For all 54 variables, null values were filled with "Unknown"/"None" for categorical variables, False for Boolean variables, or the median for numerical variables. Columns were renamed conventionally and minor errors in specific columns were fixed, such as removing extra characters or changing data types. Finally, the data frame was exported as a csv file.

9.1.3 Loading into RStudio

The csv file described in the previous section was loaded into an R markdown file in RStudio. The decision variable was recoded into 'enroll' or 'not enroll' instead of containing eight separate categories. Data types were then converted to either numeric or factor, and a new csv file (the "Initial data set") was saved for future use in Weka. The data set was split into 75% training / 25% testing data sets.

9.1.4 Creating SMOTE Data Set (RStudio)

SMOTE was performed using the `SMOTE` function in the `DMwR` package (Chawla, Bowyer, Hall, & Kegelmeyer, 2002), and the argument `perc.over` was set to 100, which drove how much over-sampling was done. The "SMOTE data set" csv file was saved for future use in Weka. The data set was split into 75% training / 25% testing data sets.

9.1.5 Creating Balanced Data Set (RStudio)

A mixture of over- and under-sampling was done using the `ovun.sample` function in the `ROSE` package (Lunardon, Menardi, & Torelli, 2014), and over- and under-sampling rates were determined by the package. The "Balanced data set" csv file was saved for future use in Weka. The data set was split into 75% training / 25% testing data sets.

9.1.6 Running Bagging (RStudio)

Bagging was run using the `randomForest` function in the `randomForest` package (Breiman, 2001), using all 53 variables as candidates at each split. A bagging model of 100 trees was created on all three data sets using the training data. Accuracy, precision, and recall were calculated on the testing data for comparison across data sets and methods.

9.1.7 Running Random Forests (RStudio)

Random forests were run using the `randomForest` function in the `randomForest` package (Breiman, 2001), using seven variables as candidates at each split. A random forest model of 100 trees was created on all three data sets using the training data. Accuracy, precision, and recall were calculated on the testing data for comparison across data sets and methods.

9.1.8 Running Forest PA (Weka)

The Forest PA algorithm was run in Weka (weka, n.d.) using the `Forest PA` package (Islam, 2020). First, all three csv files were converted to arff files for use in Weka. Weka created an additional variable for the index, this was removed. Two variables (acad_rating and counselor_rating) were converted using `StringtoNominal`, and three variables (round_yr,

birth_month, and hs_grad_year) were converted using `NumerictoNominal`. All other variables remained as imported. Forest PA was run using at 75% training / 25% testing split; these splits were different than the splits done in R (The R Foundation, n.d.), but the same percentages were used for consistency. Accuracy, precision, and recall were calculated on the testing data for comparison across data sets and methods.

9.2 Data Dictionary

Below is a data dictionary containing the variable names, types, values, and descriptions.

Name	**Type**	**Values**	**Description**
acad_rating	Categorical	16 (1595) 2 (1482) 1 (1351) 3 (1308) Unknown (690) 4 (571) 5 (212) 6 (11)	Applicant's academic rating

academic_interest_1/2/3	Categori cal	Health Professions (1104, 681, 392) ABE (1069, 1226, 931) Environmental Sciences (693, 636, 539) Biology (599, 517, 265) Psychology (464, 326, 279) Exploratory (395, 131, 157) Information Technology and Computer Science (392, 215, 124) Education (368, 295, 132) Chemistry and Biochemistry (334, 352, 315) Physics and	Applicant's top three academic interests

		Engineering Physics (248, 174, 110) Politics and Pre Law (247, 257, 201) International Studies (163, 176, 204) Social Work, Criminal Justice, and Sociology (160, 203, 183) Communication (148, 146, 179) Unknown (130, 741, 2202) English (130, 172, 132) Mathematics (112, 99, 74) Integrated Media Arts (99, 109, 74) Art (71, 113, 91) History and Museum Studies (66, 60, 67)	

		Theatre (59, 79, 83) Geology (43, 85, 85) World Languages and Cultures (42, 207, 205) Philosophy (28, 61, 70) Peace and Conflict Studies (25, 66, 65) Anthropology (16, 18, 17) Data Science (10, 37, 26) Music (2, 15, 3) Science (2, 11, 0) Religion (1, 12, 15)	
aid_gap	Numeric al	mean 3665.56 std 8664.37 min 0.0 25% 0.0 50% 1374.0 75% 3823.0 max 66575.0	Applicant's aid gap

birth_month	Categorical	10 (651) 12 (633) 6 (619) 9 (617) 4 (609) 7 (609) 8 (607) 5 (600) 3 (593) 11 (584) 2 (556) 1 (542)	Applicant's birth month
comm_nom	Boolean	False (6420) True (800)	Whether the applicant had a community nominator
counselor_call	Boolean	False (7139) True (81)	Whether the applicant received a call from an admissions counselor
counselor_rating	Categorical	0 (3044) 3 (1742) Unknown (879) 2 (664)	Counselor's prediction as to whether the applicant will enroll (1) or not (4)

		1 (647) 4 (244)	
days_taken	Numeric al	mean 35.65 std 37.46 min 0.00 25% 2.00 50% 24.00 75% 59.00 max 406.00	How many days between application creation and submission
decision	Categori cal	Admit Withdraw (2986) Drop after Decision (2491) Enroll (1401) Defer (116) Admit (115) Paid Withdraw (88) Deposit Pending (22) Deposited (1)	Applicant's decision in attending Juniata
decision	Boolean	enroll (1655) not enroll (5565)	Converted from decision (categorical)

denom	Categorical	None (3423) Roman Catholic (1126) Other - Christian (736) Methodist (265) Lutheran (198) Other - Non-Christian (193) Jewish (190) Baptist (185) Hindu (156) Presbyterian (149) Church of Christ (116) Muslim (114) Pentacostal (55) Female (50) Buddhist (48) Unitarian Universalist (UU) (36) Orthodox (36) Congregationalist	Applicant's denomination

		(UCC) (20) Latter-day Saint (Mormon) (17) Christian Scientist (16) Brethren (16) Seventh Day Adventist (14) Friend (Quaker) (14) Wiccan (Pagan) (12) Anglican (Episcopal) (9) Disciples of Christ (8) Sikh (5) Jain (4) Reformed (4) Baha'i (2) Moravian (2) Jehovah's Witness (1)	
eagles_abroad	Boolean	False (7033) True (187)	Whether the applicant won an Eagles Abroad scholarship

employee_benefit	Boolean	False (7179) True (41)	Whether the applicant received an employee benefit
est_fam_cont	Numeric al	mean 29949.82 std 50145.39 min 0.0 25% 12470.50 50% 18842.0 75% 27224.25 max 999999.0	Applicant's estimated family contribution
fafsa	Boolean	True (4645) False (2575)	Whether the applicant submitted a FAFSA
fin_aid_intent	Boolean	True (6153) False (1067)	???
fin_need	Numeric al	mean 40347.95 std 18257.20 min 0.0 25% 35891.75 50% 44330.0 75% 50833.0 max 67176.0	Applicant's financial need

first_gen	Boolean	False (5020) True (2200)	Whether the applicant is a first generation college student
has_language2	Boolean	False (4804) True (2416)	Whether the applicant has a second language
hpo_affil	Boolean	False (6431) True (789)	Whether the applicant had a health professions affiliate
hs_grad_year	Categorical	2021 (1873) 2020 (1869) 2019 (1745) 2018 (1648) 2017 (53) 2015 (10) 2016 (7) 2014 (5) 2013 (5) 2012 (3) 2023 (2)	What year the applicant graduated high school

inf_to_app1/2	Categorical	Coach (1000, 90) Family (710, 132) Guidance Counselor (706, 155) Internet Research (660, 281) Friends (591, 206) College Fair (587, 172) College that Changes Lives Book (428, 143) E-mail from Juniata College (416, 238) Visit to Campus (390, 237) Other (361, 55) None (360, 4984) Juniata Alumni (284, 168) Teacher (209, 64) Mailings from the College (176, 116) Naviance (101, 63)	Applicant's top two influences to apply

		College that Changes Lives Tour (81, 41) College Reference Guides (70, 21) Princeton Review (35, 7) Fiske Guide (32, 39) Brethren (19, 5) Church Groups (4, 3)	
jc_relation	Boolean	False (6261) True (959)	Whether the applicant had a relation at JC
language_1_first	Boolean	True (6576) False (644)	Whether the applicant marked Language 1 as their first language
lives_with	Categorical	Both Parents (5318) Parent 1 (1590) Unknown (129) Parent 2 (73) Legal Guardian (66) Other (43)	Who the applicant lives with

		Ward of the Court/State (1)	
misc_cost	Numeric al	mean 1404.39 std 678 min 625 25% 1250 50% 1250 75% 1350 max 14300	Applicant's miscellaneous cost
parent_status	Categori cal	Married (5172) Divorced (937) Never Married (480) Separated (317) Widowed (156) Unknown (140) Civil Union/Domestic Partners (18)	Applicant's parent marital status
plexus_fellowship	Boolean	False (6132) True (1088)	Whether the applicant received a PLEXUS fellowship

plexus_underrep_pop	Boolean	False (6409) True (811)	Whether the applicant belongs to a plexus underrepresented population
race	Categorical	White (4354) Hispanic (810) International (715) Black or African American (574) Asian (361) Multiracial (276) Unknown (116) American Indian or Alaska Native (11) Native Hawaiian or Other Pacific (3)	Applicant's race (IPEDS categories)
ray_day	Boolean	False (6625) True (595)	Whether the applicant received a ray day scholarship
res_status	Categorical	On Campus (7020) Off Campus (191) Unknown (7) Commuter (2)	Applicant's residential status regarding campus

rnl_avg_income	Numeric al	mean 93498 std 37596 min 15158 25% 74655 50% 83413 75% 134563 max 149330	Average income assigned by Ruffalo Noel Levitz
rnl_household_income_level	Categori cal	O-$100,000 to $149,999 (1315) N-$75,000 to $99,999 (1302) Unknown (832) M-$65,000 to $74,999 (786) S-$250,000 and Above (474) L-$60,000 to $64,999 (303) I-$45,000 to $49,999 (263) K-$55,000 to $59,999 (243) H-$40,000 to $44,999	Household income grouping assigned by Ruffalo Noel Levitz

		(241) J-$50,000 to $54,999 (220) P-$150,000 to $174,999 (196) G-$35,000 to $39,999 (179) R-$200,000 to $249,999 (142) E-$25,000 to $29,999 (140) C-$15,000 to $19,999 (136) Q-$175,000 to $199,999 (133) F-$30,000 to $34,999 (93) A-Less than $10,000 (91) B-$10,000 to $14,999 (75) D-$20,000-$24,999 (56)	

rnl_inq_score	Numeric al	mean 0.664 std 0.183 min 0.010 25% 0.580 50% 0.680, 75% 0.770 max 1.000	Assigned by Ruffalo Noel Levitz
rnl_personicx_life_stage_grou p	Categori cal	Affluent Households (1984) Unknown (832) Comfortable Households (631) Top Wealth (580) Solid Prestige (518) Large Households (332) Community Minded (297) Diverging Paths (278) Rural-Metro Mix (231) Taking Hold (204) Working & Studying	Assigned by Ruffalo Noel Levitz

		(168) Career Oriented (167) Working Households (166) Living Well (161) Starting Out (128) Bargain Hunters (107) Thrifty and Active (102) Social Connectors (96) Leisure Seekers (71) Busy Households (62) Settling Down (57) Comfortable Independence (48)	
round	Categorical	Early Action (4075) Regular Decision (2943) Early Decision (202)	Round applied during
round_yr	Categorical	2021 (2047) 2020 (1854)	Which round the applicant applied during

		2019 (1671) 2018 (1648)	
sat	Numeric al	mean 1238 std 123 min 730 25% 1190 50% 1220 75% 1292 max 1628	Applicant's SAT score (ACT scores also converted to SAT for ease of comparison)
school_#1_school_type_descr iption_x	Categori cal	Public (5254) Independent (734) Religious (679) Charter (287) Unknown (201) Home School (65)	Applicant's prior school description
school_1_type	Categori cal	H (7191) U (16) Unknown (13)	Applicant's prior school type
secondary_citizenship	Boolean	False (7055) True (165)	Whether the applicant has a secondary citizenship

sex	Categorical	F (4310) M (3087) Unknown (3)	Applicant's sex
sport	Categorical	True (4150) False (3070)	Whether the applicant plays a sport
test_opt	Boolean	False (4834) True (2386)	Whether the applicant was test optional
top_scholar_nominee	Boolean	False (5592) True (1628)	Whether the applicant was a top scholar nominee
tot_award	Numerical	mean 39807.94 std 10479.67 min 0 25% 36000 50% 39807.94 75% 45591 max 70000	Applicant's total award
tuition_cost	Numerical	mean 48598 std 1927 min 23537 25% 47075 50% 49175	Applicant's tuition cost

		75% 49175 max 52626	
tuition_discount	Numeric al	mean 70.45 std 15.65 min 0.0 25% 63.87 50% 70.32 75% 79.14 max 129.81	Applicant's tuition discount (percent)
tuition_exchange	Boolean	False (7123) True (97)	Whether the applicant participated in tuition exchange
unofficial_school_1_gpa_con verted	Numeric al	mean 3.507 std 0.585 min 0.000 25% 3.520 50% 3.620 75% 3.704 max 4.000	Applicant's prior GPA (converted to 4.0 scale)

us_stud	Boolean	True (6505) False (715)	Whether the applicant is a domestic or international resident