**Comprehensive Student Risk Model Overview**

**Summary**

This document provides information about a predictive model for students’ risk of withdrawal in the next year. It defines the population of interest, details the variables in the model, describes the data sources and methodologies used, provides intuitions for how the machine learning algorithms work, outlines the process by which the model predictions are classified, and presents metrics of the overall model performance.

The student risk model uses student data and supplemental socioeconomic data to predict the probability that a student will not persist to the next academic year. This is accomplished by first training the statistical model using historical data from prior year cohorts and then using that trained model to predict the outcomes for the current year cohort.

These predictions are represented as values between zero and one, which are the probability estimates that students will not be retained in the next year. It is important to note that probability values nearing one do not mean each of those students will not be retained. Likewise, probability values nearing zero do not mean each of those students will be retained. Instead, the probabilities reflect average expectations. For example, a probability value of .80 reflects the average expectation that eight out of ten students with that same estimated risk of withdrawal will not be retained in the next year.

**Population**

First-time, full-time undergraduates on the Pullman campus.

**Outcome (w/ source) \***

*enrl\_ind (internal)*

An indicator variable for enrollment in the next year. For the predictive part of the model, the outcome calculated is the predicted probability of non-enrollment in the next year for current year students.

**Features (w/ source)**

*pop\_dens (supplemental)*

*educ\_rate (supplemental)*

*male (internal)*

An indicator variable for gender with a reference category of female.

*underrep\_minority (internal)*

An indicator variable for underrepresented minority status with a reference category of non-underrepresented minority status. Underrepresented minorities defined as Black, Hispanic, American Indian, Native Hawaiian and Pacific Islander, and two or more races.

*pct\_blk (supplemental)*

A continuous variable for the percentage of Black residents in the postal code where the student last attended school.

*pct\_hisp (supplemental)*

A continuous variable for the percentage of Hispanic residents in the postal code where the student last attended school.

*pct\_ai (supplemental)*

A continuous variable for the percentage of American Indian residents in the postal code where the student last attended school.

*pct\_hawi (supplemental)*

A continuous variable for the percentage of Native Hawaiian and Pacific Islander residents in the postal code where the student last attended school.

*pct\_two (supplemental)*

A continuous variable for the percentage of residents of two or more races in the postal code where the student last attended school.

*pell\_eligibility\_ind (internal)*

An indicator variable for Pell grant eligibility with a reference category of non-Pell grant eligibility.

*honors\_program\_ind (internal)*

*AD\_DTA (internal)*

An indicator variable for an associate of arts degree with a reference category of non-degree holder.

*AD\_AST (internal)*

An indicator variable for an associate of science degree with a reference category of non-degree holder.

*AP (internal)*

An indicator variable for AP exam credit holder with a reference category of non-credit holder.

*RS (internal)*

An indicator variable for Running Start credit holder with a reference category of non-credit holder.

*CHS (internal)*

An indicator variable for College in High School credit holder with a reference category of non-credit holder.

*IB\_AICE (internal)*

*first\_gen\_flag (internal)*

An indicator variable for first generation status with a reference category of non-first generation status.

*fall\_avg\_difficulty (internal)*

*fall\_avg\_pct\_CDF (internal)*

*fall\_avg\_pct\_withdrawn (internal)*

A continuous variable for the average percentage of withdrawals to enrollments for the courses in which the student is registered in the fall.

*spring\_avg\_difficulty (internal)*

*spring\_avg\_pct\_CDF (internal)*

*spring\_avg\_pct\_withdrawn (internal)*

A continuous variable for the average percentage of withdrawals to enrollments for the courses in which the student is registered in the spring.

*fall\_lec\_count (internal)*

A discrete variable for the number of lecture courses in which the student is registered in the fall.

*fall\_lab\_count (internal)*

A discrete variable for the number of lab courses in which the student is registered in the fall.

*spring\_lec\_count (internal)*

A discrete variable for the number of lecture courses in which the student is registered in the spring.

*spring\_lab\_count (internal)*

A discrete variable for the number of lab courses in which the student is registered in the spring.

*total\_fall\_contact\_hrs (internal)*

A discrete variable for the total number of contact hours for the courses in which the student is registered in the fall.

*total\_spring\_contact\_hrs (internal)*

A discrete variable for the total number of contact hours for the courses in which the student is registered in the spring.

*resident (internal)*

An indicator variable for residency with a reference category of non-residency.

*gini\_indx (supplemental)*

A continuous variable for the Gini index of income inequality for residents in the postal code where the student last attended school.

*median\_inc (supplemental)* \*\*

A continuous variable for the median income of residents in the postal code where the student last attended school.

*fall\_cum\_gpa (internal)*

A continuous variable for the high school grade point average of the student.

*spring\_midterm\_gpa\_change (internal)*

*spring\_midterm\_gpa\_avg\_ind (internal)*

*remedial (internal)*

An indicator variable for remedial coursework with a reference category of non-remedial. Remedial coursework is defined as the student being registered in one or more courses designated as remedial.

*cum\_adj\_transfer\_hours (internal)*

*parent1\_highest\_educ\_lvl (internal)*

*parent2\_highest\_educ\_lvl (internal)*

*unmet\_need\_ofr (internal)*

A continuous variable for the unmet need of students relative to the amount of financial aid offered to them.

count\_week\_from\_term\_begin\_dt (internal)

Footnotes:

\* The outcome variable is used for model training based on prior years’ data. For the current year, the trained model tries to predict student enrollment outcomes next year.

\*\* These features have been adjusted for inflation in constant 2018 US dollars.

**Data**

*Internal*

All institutional characteristics of students are drawn from admissions or census data. These data adhere to the practices and procedures adopted by the Office of Institutional Research and the University.

*Supplemental*

The social and economic characteristics of the locales where students last attended school comes from the American Community Survey (ACS) TIGER/Line with Selected Demographic and Economic Data: <https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-data.html>. Complete coverage of each of the geographies provided by the US Census Bureau occurs by aggregating single-year estimates over five-year periods. For example, the 2018 ACS 5-year Summary File spans from 2014 to 2018, while the 2017 ACS 5-year Summary File spans from 2013 to 2017.

Although widely used in research, academic literature varies in its approach to these multi-year data. Some scholars elect to have the aggregated data correspond to the ending year. In effect, the 2018 data would then be treated as if it were representative of the social and economic characteristics of the locales in 2018. Other scholars elect to have the aggregated data correspond to the middle year. In effect, the 2017 data would then be treated as if it were representative of the social and economic characteristics of the locales in 2015.

Due to data limitations, as well as the practical needs of the predictive model, the decision was made to follow the former logic with the addition of a lag period of two years. This was done for two reasons. First, the 2018 ACS 5-year Summary File is the latest data available from the US Census Bureau, with these releases typically occurring on a yearly basis on or around December 10. Second, given that new students entering in the current year likely attended high school in the prior year, a lag of a year or more was appropriate to better capture the time-ordered effects that the social and economic characteristics of the locales would have on students’ last attended schools. (An example SAS code file for preparing ACS data for the student risk model can be found here: [Z:\Nathan\Models\student\_risk\student\_risk\_acs\_prep.sas](file:///Z:\Nathan\Models\student_risk)).

Theoretically speaking, these accommodations can be justified in this way: as the primary interest is treating the variables derived from the ACS as social and economic characteristics that load on the locale where the students’ last attended schools, and not on the students themselves, the use of multi-year data in conjunction with a two-year lag provides the most relevant historical context. A student arriving in the 2020 academic year (Fall 2019 term) is using data from the 2018 ACS 5-year Summary File—spanning from 2014 to 2018—which covers the range of their high school career.

When necessary, the economic variables were adjusted for inflation according to the final year present in the released data. The methodology for this adjustment, as well as the annual average consumer price index data needed to calculate it, are provided by the US Census Bureau: <https://www.census.gov/topics/income-poverty/income/guidance/current-vs-constant-dollars.html>.

**Models**

The modeling strategy used is that of an ensemble model, which in this case employs any number of machine learning algorithms commonly used for binary classification: logistic regression, support vector classification, multi-layer perceptron classification, and random forest classification. The intuition behind this strategy is that each algorithm has its own inherent strengths and weaknesses that can be balanced against one another to yield an aggregate model that is more robust to overfitting while also maintaining predictive accuracy.

*Logistic Regression*

Logistic regression with L2 regularization. Regularization is a method of penalizing complex models to prevent overfitting by including a penalty term along with the loss function being minimized. For this variant of L2 regularization, the penalty term is the sum of the squared coefficients scaled by the inverse of a hyperparameter, which is then added to the error term. A hyperparameter being an exogenous adjustable parameter. Lowering the value of the inverse hyperparameter promotes model underfitting, while raising the value of the inverse hyperparameter promotes model overfitting. The ideal value for the inverse hyperparameter produces a model that generalizes well to new data.

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html>

*Support Vector Classification*

Support vector classifier with a linear kernel. Support vector classification represents the data in an input space the dimension of the number of features. In the case of a binary classifier, the goal is to split the multi-dimensional input space into two partitions that divide the classes using a hyperplane. A hyperplane being an affine subspace that is of one dimension less than the input space. With a linear kernel, this partitioning is done linearly, which does not necessitate mapping the data to a higher-dimensional feature space as required by non-linear kernels. The general rule of thumb is parsimony. A linear kernel is preferred over non-linear kernels wherever suitable, especially when there are many features.

<https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>

*Multi-Layer Perceptron*

Multi-layer perceptron neural network with a rectified linear unit (ReLU) activation function. While neural networks are mathematically more complex than other classification models, the intuition behind them is not quite so complicated. The objective is to find a set of mathematical functions (represented by the hidden layers) that map the features of the observations (represented by the input layer) to their target values (represented by the output layer) with minimal error. Neural networks allow for the hidden layers to be inherently nonlinear through their combined use of weights, biases, and activation functions. An activation function is a nonlinear monotonic function. Using the ReLU activation function, the output is of a given node is zero when its input is less than zero and it is equal to its input otherwise. One advantage of ReLU is its efficiency. More sophisticated activation functions can make it harder for the neural network to learn the weights and biases that produce the minimal amount of error.

<https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html>

*Stochastic Gradient Descent*

Stochastic gradient descent with a modified Huber loss function. This classification model is a generalization of other stochastic gradient descent algorithms. Used with a modified Huber loss function, it is equivalent to a quadratically-smoothed support vector machine. Support vector machines represent the data in an input space the dimension of the number of features. In the case of a binary classifier, the goal is to split the multi-dimensional input space into two partitions that divide the classes using a hyperplane. A hyperplane being an affine subspace that is of one dimension less than the input space.

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html>

*Random Forest Classification*

Random forest classifier with a Gini impurity criterion. Random forest classification aggregates several decision tree classifiers that are fit to datasets sub-sampled from the original dataset using replacement. Random forest classification searches for split points among the bootstrapped data that partition the classes so that the resulting Gini impurity is minimized. The Gini impurity of a split point is the weighted average of one minus the sum of the squared probabilities of each classification after the split. The root node of every decision tree is that feature represented in the sub-sampled dataset (and its associated value) that results in the lowest calculated Gini impurity. The decision tree algorithm then branches out iteratively by finding the next best feature in the sub-sampled dataset (and its associated value) that results in the next lowest calculated Gini impurity for the previously split nodes and so on until there is no improvement. The randomness introduced by combining the results of decision tree classifiers run on many bootstrapped datasets produces an averaged classifier that is more robust to overfitting than standard decision tree classifiers.

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

**Classification**

*Voting Classification*

Voting classifier with weighted average probabilities. Voting classification is an ensemble method that combines the predictions of the included estimators to improve their generalizability. The weighted average of the predicted probabilities for the base models is calculated for each observation. The chosen weights determine the relative contribution of each algorithm to the average outcome.

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.VotingClassifier.html>

*Risk Thresholds*

The risk thresholds are set based on the predicted probabilities of non-enrollment in the next year produced by the voting classifier. Those students with predicted probabilities equal to or above the .66 level are considered high risk, those equal to or above the .33 level and below the .66 level are considered medium risk, and those below the .33 level are considered low risk.

**Performance**

*Baseline*

Table 1a and 1b present logistic regression results for the training data. Standard logistic regression allows for the estimation of unbiased coefficients and p-values of the included variables, which cannot be obtained from the machine learning models. As such, it provides useful context for understanding those features in the ensemble model that are significant factors for predicting student success. Note that this model uses the indicator variable for enrollment in the next year as its outcome. Positive log odds coefficients denote those variables that increase the odds of retention, while negative log odds coefficients denote those variables that decrease the odds of retention. Exponentiating the log odds yields the odds ratios.

|  |
| --- |
| ***Table 1a. Pullman freshmen midterm logistic regression*** |
|  |

|  |
| --- |
| ***Table 1b. Pullman freshmen midterm logistic regression (continued)*** |
|  |

Figure 1 presents the ensemble model receiver operating characteristic (ROC) curve for the training data. The dashed line is the line of no discrimination where a hypothetical model would predict the same proportion of correctly classified and wrongly classified outcomes. Moving along the diagonal from the lower left corner to the upper right corner coincides with decreasing the threshold for classification. In this case, the threshold being the value above which a predicted outcome would be classified as retained after a year and below which a predicted outcome would be classified as withdrawn after a year. A common measure for the discriminatory ability of classification models is the area under the curve (AUC), which is calculated across the range of threshold values from higher (and more conservative) values to lower (and more liberal) values. Thus the AUC provides a global measure of model performance. By comparison, the overall accuracy provides a local measure of model performance at a set threshold, which by default for binary classification is .50.

The ROC curve for the Pullman freshmen ensemble model is displayed in the figure as a solid line. The AUC for the ensemble model is .8869. The overall accuracy of the Pullman freshmen ensemble model at the .50 threshold level is .8646.

|  |
| --- |
| ***Figure 1. Pullman freshmen ensemble model ROC curve*** |
|  |

**Conclusion**

Based on the calculated AUC, the performance of the pre-census student risk model in predicting students’ risk of withdrawal in the next year is considered to be good (Gorunescu et al. 2011). Further, the ensemble model leverages multiple machine learning algorithms commonly used for the classification of binary data that when combined avoid overfitting the data while maintaining high levels of accuracy in the predictions. These results represent the baseline of performance that can expected under this modeling strategy for the midterm snapshot. Further performance improvements could be gained by developing new variables or tuning the models based on their observed outcomes next fall.

**Citations**

Gorunescu, Florin, Janusz Kacprzyk, and Lakhmi C. Jain. 2011. *Classification Performance Evaluation*,

*Vol. 12, 1st Ed.* Springer: Berlin. DOI: 10.1007/978-3-642-19721-5\_6.