Stony Brook University

SUNY

Using Data Mining to Model Freshmen Outcomes

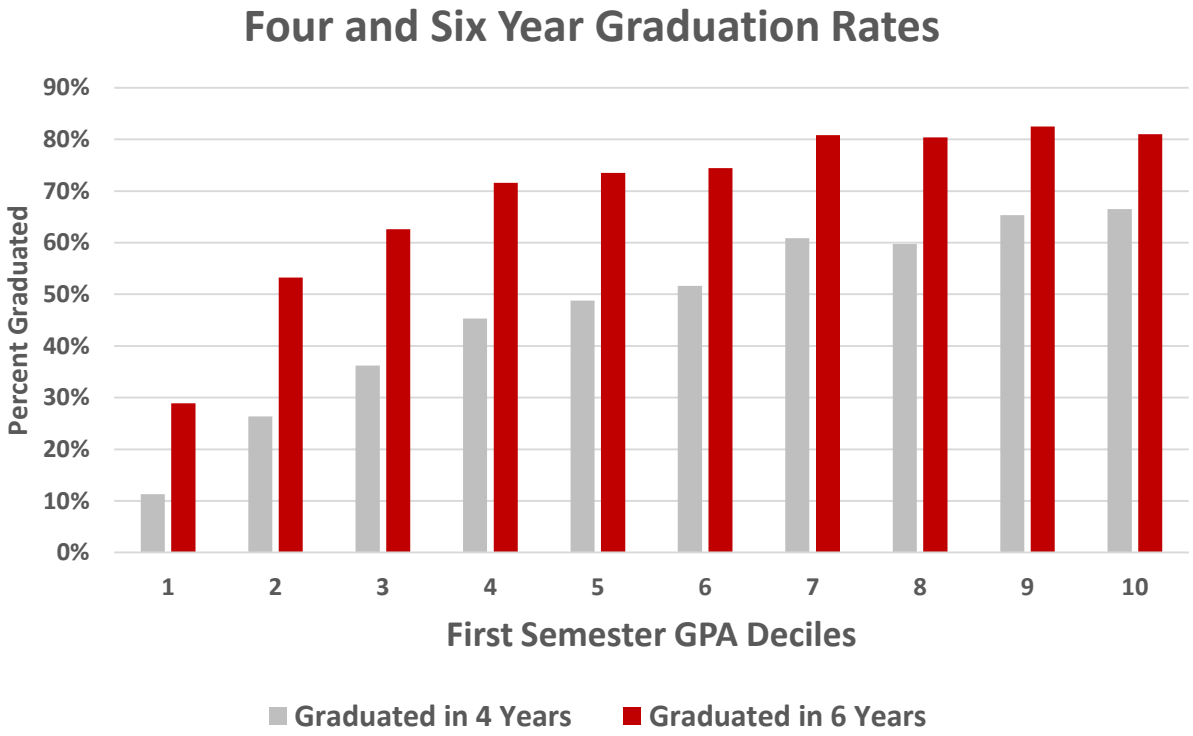
42nd NEAIR Annual Conference

Burlington, VT

Nora Galambos, PhD

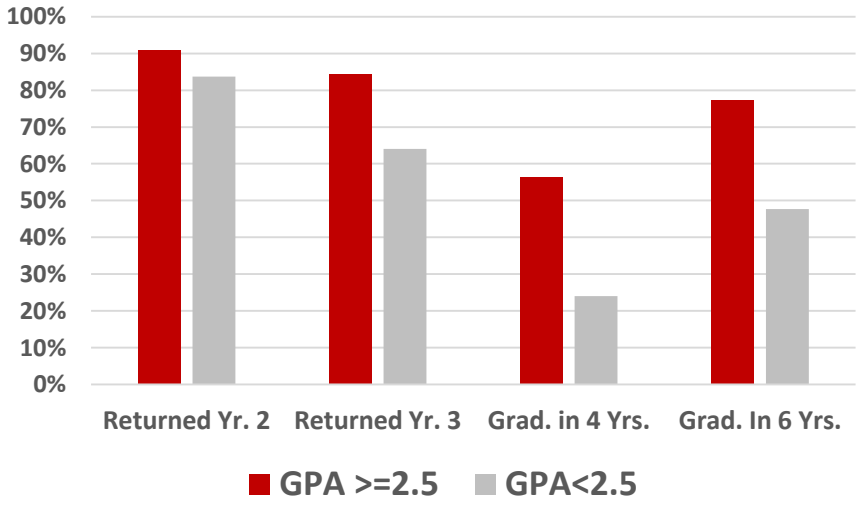
Senior Data Scientist

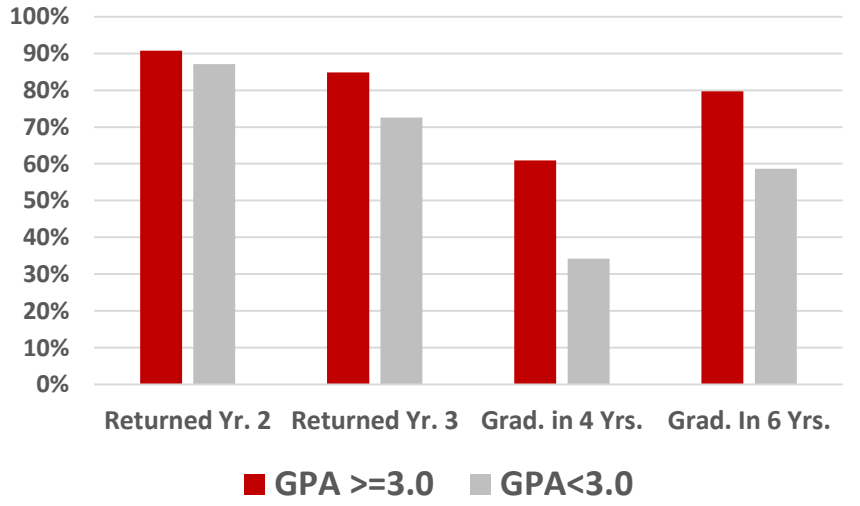
Graduation Rates of First-time Full-time Freshmen by First Semester GPA Deciles*



*The fall freshmen cohorts of 2006 through 2008 were combined.

Graduation and Retention of First-time Full-time Freshmen by First Semester GPA*





Approximately 30% have a GPA below 2.5. 3.0 is approximately the median GPA.

*The fall freshmen cohorts of 2006 through 2008 were combined.

Using Data Mining for the Early Prediction of Freshmen Outcomes

- Enables the extraction of information from large amounts of data.
- Incorporates analytic tools for data-driven decision making.
- Uses modeling techniques to apply results to future data.
  - *The goal is to develop a model rather than finding factors significantly associated with the outcomes.*
- Incorporates statistics, pattern recognition, and mathematics.
- Few assumptions to satisfy relative to traditional hypothesis driven methods.
- A variety of different methods for different types of data and predictive needs.
- Able to handle a great volume of data with hundreds of predictors.

Data Mining: Training and Validation to Develop an Accurate Model

- The purpose of the analysis is to predict the GPA of first-time full-time fall 2014 freshmen using high school characteristics, demographics, and academic information from the first half of the first term—early enough to assign students to interventions.
- Need to find the correct level of model complexity.
  - A model that is not complex enough may lack the flexibility to represent the data, under-fitting.
  - When the model is too complex it can be influenced by random noise, over-fitting.
  - For example, if there are outliers, an overly complex model will be fit to them. Then when the model is run on new data, it may be a poor fit. A poor fitting model will not do a good job in making predictions using new data.

Data Partitioning

- Partitioning is used to avoid over- or under-fitting. Divide the data into three parts: training, validation, and testing.
- The ***training*** partition is used to build the model.
- The ***validation*** partition is set aside and is used to test the accuracy and fine tune the model.
  - The prediction error is calculated using the validation data.
  - An increase in the error in the validation set may be caused by over-fitting. The model may need modification.
  - Often 60% is used for training the model and 40% is used for validation--or 40% for training, 30% for validation, and 30% for testing
  - ***Problem:*** With only 2,852 fall freshmen and over 50 variables, there is not enough data available to develop a model to predict the 300 to 500 low GPA students using traditional partitioning.

K-fold Cross-validation for Evaluating Model Performance

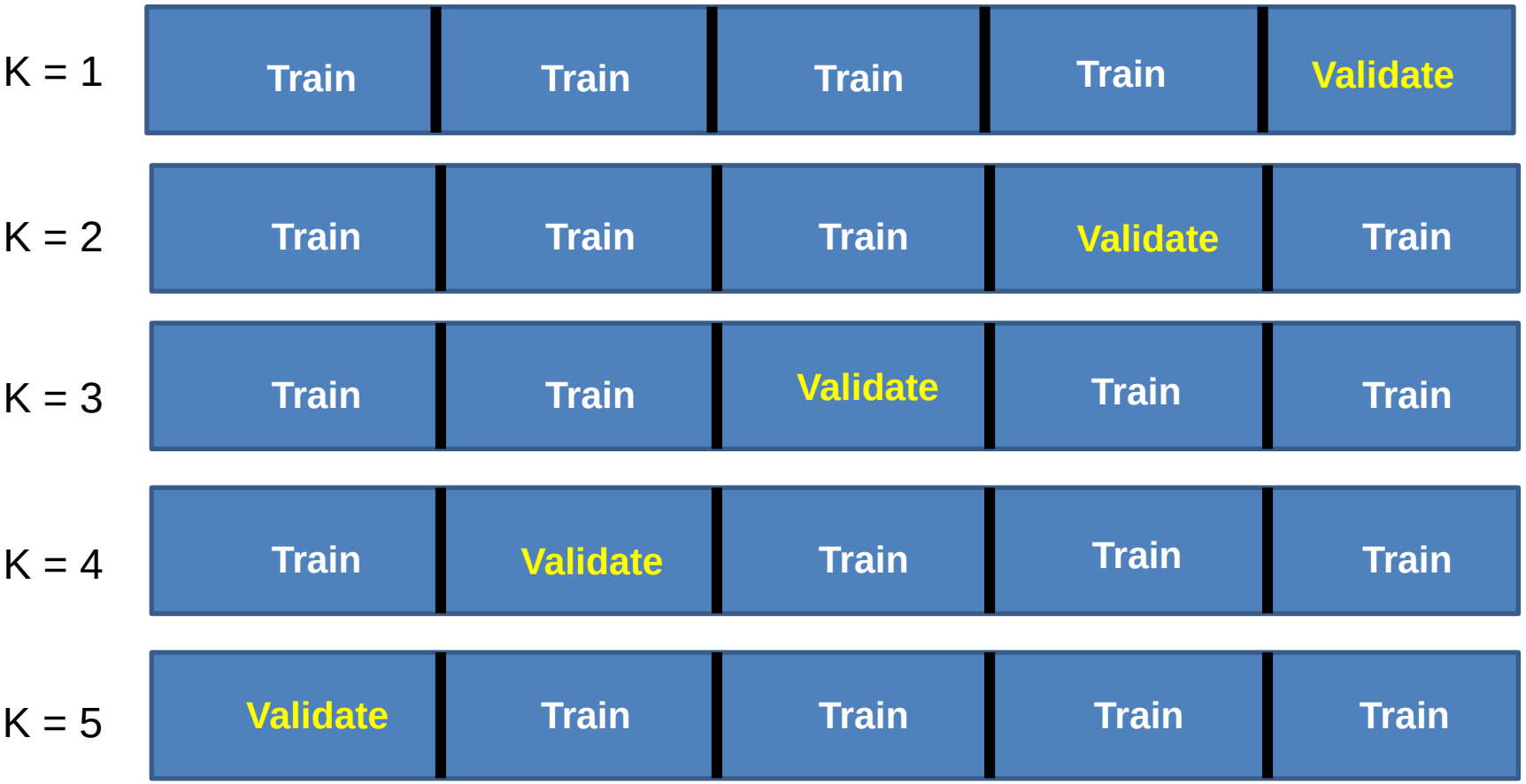
- Why use k-fold cross-validation?
  - Works with limited data.
  - The initial steps are the similar to traditional data analysis.
  - The entire dataset is used to choose the predictors.
  - Cross-validation is used to evaluate the model, not to develop the model.
  - The error is estimated by averaging the error of the K test samples.
- In subsequent years, once more data has been collected, the easier to implement training-validation partitioning method can be used.

K-fold Cross-validation for Evaluating Model Performance

- The sample is divided into K equal groups, or folds.
  - For this data with the focus on finding the lowest GPA students, only 5 folds were used.
- Next the model is run K times, however each time, one fold is left out.
- For five folds, four are for training and one is for validation
- The procedure is performed K times (in this case five times), each time leaving out a different validation sample

5-Fold Cross Validation Plan

For each K_i the entire dataset was divided into 5 equal parts



Data Mining Methods Compared

- Five data mining methods were compared*
  - CHAID: Chi-Square Automatic Interaction Detection
  - BFOS – CART (Breiman, Friedman, Olshen, Stone Classification and Regression Trees)
  - Decision Tree Combined Methods
  - Gradient Boosting
  - Linear Regression

- The validation and training average squared errors (ASE) were averaged over each of the five-fold runs to determine the method that has, on average, the smallest error and the smallest difference between the validation and training errors.

* Software settings were used to approximate the different decision tree methods.

BFOS CART Method
Breiman, Friedman, Olshen, Stone
Classification and Regression Trees

- The method does an exhaustive search for the best binary split.
- It splits categorical predictors into a smaller number of groups or finds the optimal split in numerical measures.
  - Each successive split is again split in two until no further splits are possible.
  - The result is a tree of maximum possible size, which is then pruned back.
  - For interval targets the variance is use to assess the splits; For nominal targets the Gini impurity measure is used.
  - Pruning starts with the split that has the smallest contribution to the model
  - The missing data is assigned to the largest node of a split
- Creates a set of nested binary decision rules to predict an outcome.

CHAID: Chi-squared Automatic Interaction Detection

- Unlike CART with binary splits evaluated by misclassification measures, the CHAID algorithm uses the chi-square test (or the F test for interval targets) to determine significant splits and find independent variables with the strongest association with the outcome. A Bonferroni correction to the p-value is applied prior to the split.
- It may find multiple splits in continuous variables, and allows splitting of categorical data into more than two categories.
- As with CART, CHAID allows different predictors for different sides of a split.
- The CHAID algorithm will halt when statistically significant splits are no longer found in the data.

Decision Tree Combined Methods

- Up to four-way splits (4 branches) were allowed as opposed to the CART binary split.
- The F test was used to evaluate the variance of the nodes.
- The depth of the overall tree was restricted to 6 levels.
- Missing values were assigned to produce an optimal split.
- The Average Squared Error (ASE) was used to evaluate the subtrees.
- The cross validation option was selected in order to perform the cross validation procedure for each subtree.
    - That results in a sequence of estimates that were used to select the subtree.

Gradient Boosting

- Uses a partitioning algorithm developed by Jerome Friedman.
- Resamples the data set a number of times without replacement.
  - A random sample is drawn at each iteration from the training data set and the sample is used to update the model.
  - The successive resampling results in a weighted average of the re-sampled data.
  - The weights assigned at each iteration improve the accuracy of the predictions.
- The result is a series of decision trees, each one adjusted with new weights based on the accuracy of the estimates or classifications of the previous tree.
- Because the results at each stage are weighted and combined into a final model, there is no resulting tree diagram.
  - The scoring code that is generated allows the model to be used to score new data for predicting outcomes.

Linear Regression in a Data Mining Environment

- Imputation was used to replace missing values.
  - Decision tree methods are able to handle missing values by combining them with another category or placing them in a category with other values. They can also be replaced by using surrogate rules.
  - Linear regression, on the other hand, will listwise delete the missing values.
    - In student data, many predictors do not have complete data—e.g., financial aid measures (not all students apply for financial aid), SAT scores (a certain small percentage of the entering freshmen do not have scores), some students may not have courses utilizing the LMS.
    - These measures result in an excessive amount of listwise deletion.
  - The distribution method was used to replace the missing data
    - Values are calculated from random percentiles of the distributions of the predictors.
  - If the linear regression appears promising, other imputation methods can be explored and studied in greater detail.

Linear Regression in a Data Mining Environment

- Clustering to reduce multicollinearity
  - The other difficulty in using linear regression with so many variables is multicollinearity. With a large volume of predictors, it would be difficult and time consuming to evaluate all of the potential multicollinearity issues.
  - Clustering was used to group highly correlated variables.
  - In each cluster, the variable with the highest correlation coefficient was retained and entered into the modeling process, and the others were eliminated.

Predictive Measures

- Demographics
  - Gender, ethnicity, geographic residence when admitted.
- Pre-college academic characteristics
  - SAT scores, high school GPA, average SAT scores of the high school (to control for high school GPA).
- College academic characteristics
  - Credits accepted when admitted, AP credits, number of STEM and non-STEM courses enrolled in, enrollment in high DFW courses, area of major.
- Transactions, service utilization, activities.
  - Learning management system (LMS) logins, advising visits, tutoring center utilization, intramural and fitness class participation.
- Financial aid
  - Expected family contribution, AGI, types and amounts of disbursed aid, Pell, Tuition Assistance Program (TAP).

About the Data

- Much of the data pertaining to interactions with student services and learning management system logins has not been stored long term. As a result, part of the data mining process includes collecting, saving, and storing the data.
- Programs are being developed to automate the formatting and aggregation of the transactional data so it can be merged with student records and utilized in the data mining process.
    - For the modeling use of the learning management system logins, the data were summarized as follows:
        - Only one login per course per hour was counted, so a course can have up to 24 logins per day. This eliminated multiple logins that occurred just few minutes apart.

About the Data Continued

- Total logins were summarized as:
  - Each student's courses were categorized as STEM or non-STEM
  - The STEM and non-STEM logins were totaled for week 1 and separately for weeks 2 through 6.
  - STEM and non-STEM logins were divided by the respective STEM and non-STEM course totals to obtain per-course login rates (by STEM and non-STEM).
- High DFW rate courses
  - Courses with an enrollment of 70 or more having high percentages of D's, F's, and W's were identified and categorized as STEM and non-STEM
  - The number of high DFW courses, and the highest DFW rate for each student was included in the model.
- ***Research and evaluation of the methods for summarizing and using the data in the model is ongoing. Additional data sources will be added.***

Average Squared Error (ASE) Results for the Five Data Mining Methods

	Gradient Boosting		BFOS-CART		CHAID		Decision Tree		Linear Regression	
K Folds	Validation ASE	Training ASE	Validation ASE	Training ASE	Validation ASE	Training ASE	Validation ASE	Training ASE	Validation ASE	Training ASE
1	0.333	0.363	0.394	0.427	0.444	0.355	0.421	0.335	0.374	0.396
2	0.353	0.358	0.425	0.423	0.479	0.325	0.432	0.330	0.477	0.388
3	0.377	0.351	0.429	0.432	0.508	0.312	0.472	0.325	0.515	0.363
4	0.391	0.351	0.436	0.433	0.510	0.304	0.495	0.304	0.522	0.376
5	0.422	0.343	0.525	0.393	0.511	0.345	0.515	0.312	0.561	0.371
Average ASE	**0.375**	**0.353**	**0.442**	**0.422**	**0.490**	**0.328**	**0.467**	**0.321**	**0.490**	**0.379**

- ASE = (Sum of Squared Errors)/N
- Gradient boosting had the smallest average ASE followed by CART
- Gradient boosting and BFOS-CART, on average, had the smallest differences between the validation and training errors
- The CART method was chosen for the modeling process.
  - Relatively low ASE.
  - Gradient boosting, without an actual tree diagram, would make the results more difficult to explain to administrators.

Relative Importance of Variables as Evaluated by CART

The "importance" evaluates a measure's contribution to the model, utilizing the relative reduction in the sum of squared errors when a node is split..

Variable*	Relative Importance
High School GPA	1.0000
Scholarship Aid	0.9643
Total AP non-STEM course credits	0.8980
Total AP STEM course credits	0.8729
LMS logins per STEM courses weeks 2 -6	0.8619
Total LMS STEM course logins, weeks 2 -6	0.8542
LMS logins per non-STEM courses, weeks 2 -6	0.8214
Area of residence when admitted	0.7921
Total LMS non-STEM logins, weeks 2 – 6	0.7888
Declared major	0.6902
Total non-STEM units	0.6859
Total LMS non-STEM course logins, week 1	0.6712
Total STEM units	0.5789
Avg. high school SAT Math, CR, writing score	0.5577
Student SAT Math + CR	0.5540
Avg. high school SAT CR score	0.5357
Total LMS STEM course logins, week 1	0.5307
Avg. high school SAT Math, CR total score	0.5176
Total STEM courses	0.5119
Avg. high school SAT Math score	0.5080
Total non-STEM courses	0.4808
Type of math course in term 1	0.4636

Variable	Relative Importance
Total STEM courses using LMS	0.4258
Advising visits, week 1 pertaining to registration	0.3826
Ethnic group	0.3609
Highest DFW rate in non-STEM course	0.3425
Student SAT Math score	0.3197
Total non-STEM courses using LMS	0.3115
Total Athletics Aid	0.2736
Total high DFW STEM units	0.2714
Intramural sports participation	0.2548
Tutoring Center visits for STEM courses, wks 1 – 6	0.2533
Fitness Class attendance	0.2378
Student SAT CR score	0.2146
Highest DFW STEM rate	0.1868
Honors College or Women in Science & Eng.	0.1827
Total high DFW STEM courses	0.1624
Stony Brook Math Placement Exam score	0.1500
Student SAT Writing Score	0.1495
Total grant aid	0.1436
% of freshmen in highest DFW rate STEM course	0.1191
Total loans distributed (per Fin. Aid Off. Records)	0.1155
Advising visit during week 1, not registration-related	0.1149
% of 1st-years in highest DFW rate non-STEM crs	0.0721

* Measures selected by CART for the model appear in red.

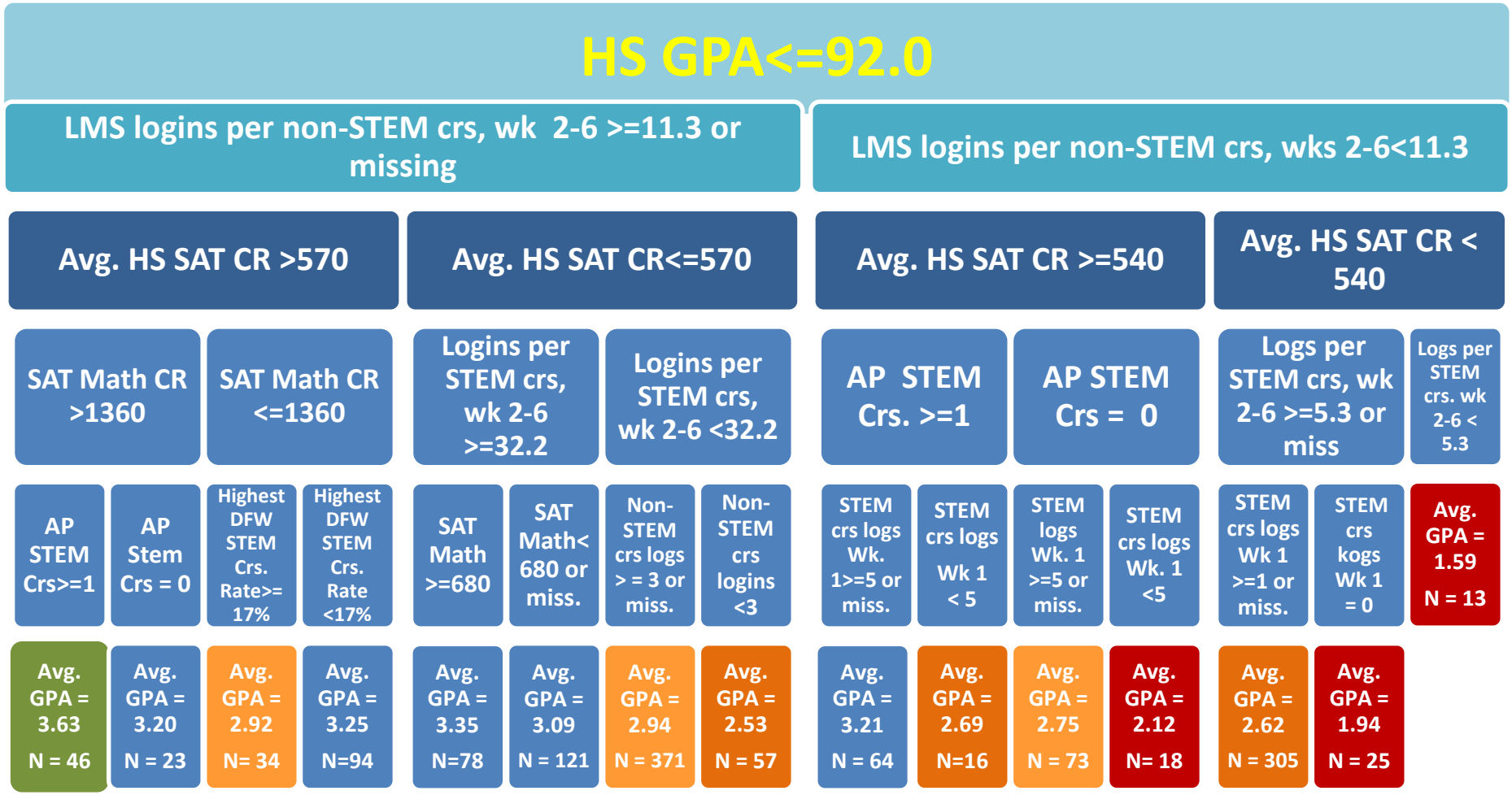
Assessment Score Distribution

Range for Predicted	Avg. F14 GPA	Number of Observations	Model Score
3.64 - 3.76	3.76	37	3.70
3.51 - 3.64	3.60	459	3.57
3.39 - 3.51	3.46	257	3.45
3.27 - 3.39	3.35	78	3.33
3.14 - 3.27	3.23	344	3.21
3.02 - 3.14	3.08	665	3.08
2.90 - 3.02	2.93	478	2.96
2.65 - 2.78	2.74	89	2.71
2.53 - 2.65	2.61	362	2.59
2.41 - 2.53	2.52	16	2.47
2.04 - 2.16	2.12	18	2.10
1.92 - 2.04	1.94	25	1.98
1.55 - 1.67	1.59	13	1.61
1.30 - 1.43	1.30	11	1.36

Stony Brook University

SUNY

Decision Tree Model for F14 Freshmen GPA: Part 1—HS GPA <= 92.0



Stony Brook University

SUNY

Decision Tree Model for F14 Freshmen GPA: Part 2—HS GPA > 92.0

HS GPA>92.0 or Missing

Scholarship = Yes

HS GPA >=96.5 or missing

- Math Placement Exam >= 5
  - Logs per STEM Crs., wks 2-6 >=15.6 — Avg. GPA = 3.63 N = 285
  - Logs per STEM Crs, wk 2-6 <15.6 — Avg. GPA 3.40 N = 83
- Math Placement Exam < 5
  - Ethnic Group = White, Hisp. — Avg. GPA =3.50 N= 73
  - Ethnic Group= Asian, Afr. Amer., Unk. — Avg. GPA = 3.05 N=30

HS GPA < 96.5

- Logs per non-STEM crs,wks 2-6 >=29.1
  - SAT Math >=700 — Avg. GPA = 3.76 N=26
  - SAT Math< 700 or miss. — Avg. GPA = 3.52 N = 74
- Logs per non-STEM crs,wks 2-6 <29.1
  - Avg HS. CR, M Wrt >=1830 miss — Avg. GPA = 3.59 N = 54
  - Avg. HS CR, M, Wrt< 1830 — Avg. GPA = 3.13 N = 54

Scholarship = No

LMS logins per non-STEM crs. Wk 2-6 >=10.4

- AP STEM Crs. >=1
  - DFW STEM Crs Total >=2 — Avg. GPA = 3.23 N = 163
  - DFW STEM Crs Total <2 — Avg. GPA = 3.49 N=101
- AP STEM Crs = 0
  - SAT Math >=760 — Avg. GPA = 3.76 N = 11
  - SAT Math <760 — Avg. GPA = 3.03 N= 194

LMS logins per non-STEM crs. wk 2-6 < 10.4

- Logs per STEM crs, wks 2-6 >=10.9 or miss.
  - DFW non-STEM 1st yrs >=28% — Avg. GPA = 3.05 N = 72
  - DFW non-STEM 1st yrs <28% — Avg. GPA = 2.90 N = 73
- Logs per STEM crs. wks 2 6 < 10.9
  - STEM Crs logs Wk 1 >=8 — Avg. GPA= 1.30 N=11
  - STEM Crs logs Wk 1 <8 or miss — Avg. GPA = 2.52 N = 16

How Can the Results be Used?

- The model as presented can be used to assign students to designed interventions, e.g., tutoring.
- Model results can be shared with departments to inform their advising and intervention efforts.
- *The goal is to find the students who need assistance to fulfill their potential, and reduce the number who end up leaving due to poor performance.*