GEORGETOWN UNIVERSITY
School *of* Continuing Studies
Certificate in Data Science Program

Machine Learning Prediction of High School Graduation Rates

Cohort 28 - Capstone Project - “student-outcomes”

Authors:
Amy Wiggington
Blake Baird
Hayat Pacquette

Capstone Advisor:
Molly Morrison

GitHub: https://github.com/georgetown-analytics/student-outcomes

Table of Contents

Abstract

Imagine being able to predict the factors leading to dropout and graduation rates in the United States. In 2018–19, the Adjusted Cohort Graduation Rate (ACGR[1]) ranged from 69% in the District of Columbia to 92% in Iowa. Forty states reported ACGRs ranging from 80% to less than 90% [1-2, Table 219.46]. What accounts for this discrepancy? A staggering 2,829 unique schools had less than 25% graduation rate between 2010-2019. Using a regression model, our project examined over 80,000 instances and aimed to reveal the most significant contributing factors to graduation rates as well as producing a model to predict graduation rate based on features of individual schools. Our top performing model was a random forest model with accuracy 0.709 and root mean square error of 10.48 percentage points. We also produced a random forest classification model which segments public high schools into two groups, <90% and ≥90% graduation rate, with an F1 score of 0.789. These two models could be utilized by secondary education stakeholders for planning or targeted intervention.

Introduction

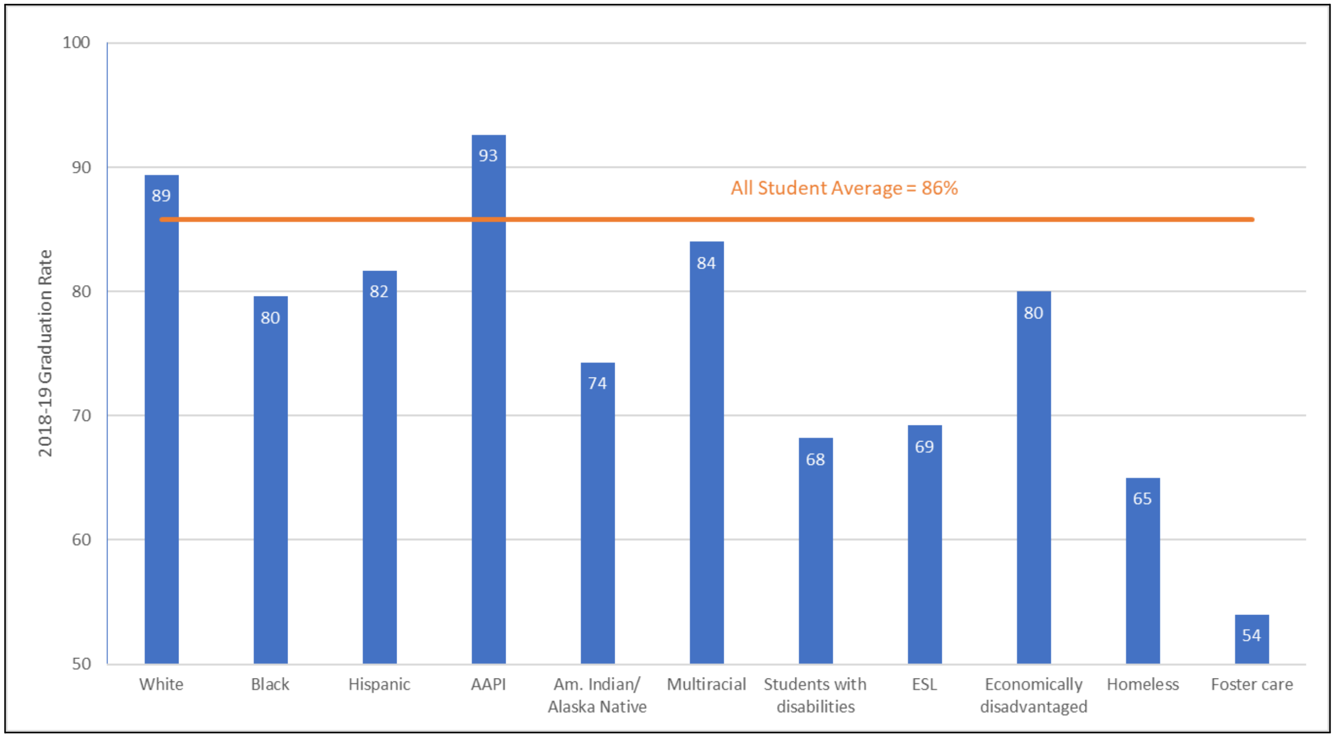
Motivation

Since our team includes two subject matter experts in the field of secondary education- a former educator and a college access professional, focusing our project in this area seemed most fitting. In this way, we could combine what we already know with what we have been learning to apply machine learning and other data science concepts to improving America's education system. In addition, the United States spends approximately $765 billion on K-12 education annually, an important investment that should be providing the intended benefits to our nation's youth [3]. We decided to focus specifically on high school graduation rates, because they are an important metric; students with a high school diploma earned almost $10,000 more per year and have a 2% lower unemployment rate [4].

While the 79% to 86% increase in nationwide adjusted cohort graduation rate (ACGR) during the previous decade seems encouraging, aggregating the information hides the problems faced by smaller sectors and subgroups. For example, Tennessee has one of the highest statewide graduation rates in the country at 91%, but Nashville's Glencliff High School's most recent cohort had a graduation rate of only 53%, being deemed a "dropout factory" [5]. Locally, the graduation rate for the District of Columbia is the lowest in the nation, six percent lower than that of the lowest ranking state [6]. In Henrico, Virginia- neighboring the state capitol- the dropout rate has actually increased in recent years [7]. Even within school, district, state, and national levels, many subgroups of students- including racial minorities, economically disadvantaged, and students with disabilities- are not graduating at the same rate as their peers, as demonstrated in Figure 1, p.2 [1-2, Table 219.46].

[1] Due to the high number of technical terms and subject field jargon involved in this project, we have included a Glossary at the end of this report to serve as a reference for the reader.

Figure 1. ACGR disparities in student subgroups



Data on individual students is not publicly available for privacy reasons, so we turned our attention to school level datasets for this project.

Hypothesis

We set out to build a supervised machine learning model to predict public high school graduation rates based on publicly available school and county data from prior years. We hypothesized that economic factors would be the most significant predictors of graduation rate.

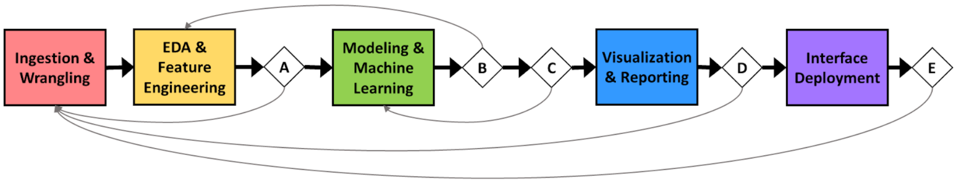
Applications

The key predictive factors uncovered will provide insight that enables all stakeholders- educators, administrators, parents, and students to make data-driven decisions to address graduation rates. Stakeholders can open the deployed model and enter features to predict graduation rates for a cohort. This could be used to target a school or subset of schools for interventions or in the planning phase of new schools.

Project Architecture

For this project, we followed the data science pipeline, which consists of several key processes, shown in the boxes in Figure 2. Each of these processes further elucidates aspects of the data- from what is missing to what is extraneous and from what needs to be organized differently to ways that the hypothesis itself may need to be adjusted; therefore, this pipeline is not a constant, direct procedure, but rather has a circular, iterative nature, indicated by the backward-looping arrows in Figure 2. The diamond-shaped components of the figures represent decision points for our team to determine if our project is prepared to continue to the next phase or if it needs to return to a previous process for adjustment. An example of the type of questions that would be asked at each of these stages is outlined below Figure 2, p.3-4.

Figure 2. Data science pipeline plan for this project


Ingestion & Wrangling
EDA & Feature Engineering
A
Modeling & Machine Learning
B
C
Visualization & Reporting
D
Interface Deployment
E

A: Do we have enough or the right type of data? Did we uncover any outliers that need to be further cleaned?

B: Did the results indicate that better or different data preparation methods could improve our model?

C: Do we need to make adjustments to our model? Have we tested multiple types of models?

D: Do our results indicate that our models are performing poorly and may see improvement through obtaining additional data?

E: Does stakeholder feedback provide ideas for improvement?

Tools Used

This project used strictly open-source software with the exception of the Amazon S3 service, which could be switched for private storage if available. Along with Python3 and Jupyter Notebook, we utilized the following packages and libraries: Bokeh, Boto3, Feature-engine, Joblib, Matplotlib, missingno, NumPy, pandas, scikit-learn, seaborn, and Yellowbrick. Version numbers and information for these tools can be found in Appendix A.

Section 1. Data

Sources

At the outset, the search for datasets that were relevant to the intended project led to the investigation of many different data sources. An emphasis was placed on those with school and academic-related fields, though community social and economic factors were also taken into consideration. Determining that the appropriate instance for the model would be individual schools for each year, sources containing high school-level data were prioritized. While rich datasets regarding public high schools were able to be found, information on private and parochial schools was not as readily available.

After finding a dataset with graduation rates- our key factor- features that were shared across this and multiple other datasets were then identified so that we would be able to merge data from the multiple sources later. A unique identifier for an individual school that is commonly used in school-related datasets is NCESSCH- a 12-digit number assigned uniquely to each school- seemed to be the easiest and cleanest way to merge several found datasets. Attention was also paid to the date ranges covered by each dataset to maximize the amount of data that could be used to train models.

In the end, four datasets were selected to proceed to the next steps. Information about each of these datasets is included in Table 1, p.5.

Table 1. File specifications of the selected datasets

	Adjusted Cohort Graduation Rate [8] ---------- U.S. Department of Education	**Schools Common Core of Data Directory** [9] ---------- Urban Institute[1]	**Assessment Participation**[2] [10] ---------- U.S. Department of Education	**County-Level Unemployment** [11] ---------- U.S. Department of Agriculture
Size	9 CSV files 31.9 MB (Total)	1 CSV file 843.8 MB	16 CSV files (8 /subject) 1.65 GB (Total)	1 XLSX file 2.08 MB
Instances	180,232 (Total) School + Year	3,381,565 School + Year	366,400 (Total) School + Year	3277 County
Features	~29 / file e.g. School, Cohort Size, Grad Rate, Subgroups	52 e.g. School ID, Location, Type, Teachers, Enrollment	~33 / file e.g. School, Participants, Proficiency Scores	96 e.g. County, Labor Force, Unemployment
Scope	SY 2010-2011 - 2018-2019 50 states, DC, Puerto Rico, USVI, BIE School Division, School	SY 1986-1987 - 2021-2022 50 states, DC, U.S. territories, BIE, DoDEA State, Division, School	SY 2012-2013 - 2018-19, 2020-2021[3] 50 states, DC, Puerto Rico, USVI, BIE School Division, School	2000 - 2021 50 states, DC, Puerto Rico Nation, State, County

[1] Data for this project was retrieved from the Urban Institute's Education Data Portal. The original data was collected and compiled by NCES and can be accessed on the CCD Data File website.

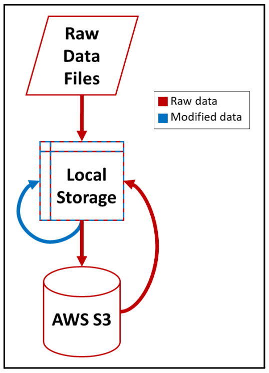
[2] Assessment data was collected separately for two subjects: Math and Reading/Language Arts.

[3] The 2019-20 SY was excluded due to COVID.

Storage

The raw data files and their accompanying documentation were retrieved from their original source locations online and stored as flat files on a dedicated Amazon AWS S3 virtual server. Because the data totaled over two gigabytes, a large cloud storage provider like AWS seemed to be the best option for our data storage. AWS also provides the key features of reliability and scalability at a low cost, becoming a common choice for professional data science projects and one we will likely encounter again in our data science careers [12].

Figure 3. The retrieval and storage of raw and modified data



In this process of retrieving, transferring, and safely storing the files in the AWS S3 server, no modifications were done and they remained in their “raw” state. To protect the integrity of the raw data, we chose to utilize a WORM (”write once, read many”) storage architecture. This is possible due to the low velocity of source data, with each of the selected datasets only being updated with new data on an annual basis. However, it can still be queried many times throughout the year and accessed for cleaning, merging, and other manipulation, as data files that have been altered are cached locally instead of affecting the WORM storage server. This process is demonstrated in Figure 3.

Ingestion and Wrangling

The ingestion process downloads the raw data from this cloud storage. We utilized the Boto3 library to communicate with the S3 cloud application programming interface (API). After ingestion, wrangling processes the data into a form suitable for local storage, feature engineering, and modeling. This includes making determinations about which schools, years, or other data should be excluded due to missing, erroneous, or mismatched information. Because each dataset is different, their wrangling procedures will be different, as well.

Dataset 1. Adjusted Cohort Graduation Rate

Because the data for the Adjusted Cohort Graduation Rate (ACGR) was provided by the U.S. Department of Education in multiple files capturing one year each, the first step was to clean each file so that they had the same column headers and formatting. The subpopulation data in these reports (i.e. breakdowns by race, disability, etc.) showed inconsistencies, explained by the accompanying documentation as related to the Elementary and Secondary Education Act’s flexibility with how states report data [13]. Therefore, we decided to drop those columns and continue only with those features pertaining to the entire cohort, which were not affected by the same issue.

Once the individual files had been cleaned, they were combined into one large *pandas* dataframe, which went through its own wrangling process. Since graduation rate was our target variable, a valid value is required for each instance on which to train or test a model. For this reason, we dropped all rows where this value was completely unavailable, most often due to small cohort sizes (a privacy issue), totalling over 9,500 instances (approximately 1,400 per year). Instead of reporting an exact graduation rate, some schools reported the value as a range, e.g. ‘88-92’ or ‘>75.’ Wanting to retain as much data as possible, we opted to convert these to a numeric value by finding the mean of the indicated range. Next, we researched the few outliers in the dataset and determined them likely to be transcription errors of which we could not determine the true value, so they were also dropped. Finally, this modified dataset was saved as an intermediate file on local storage.

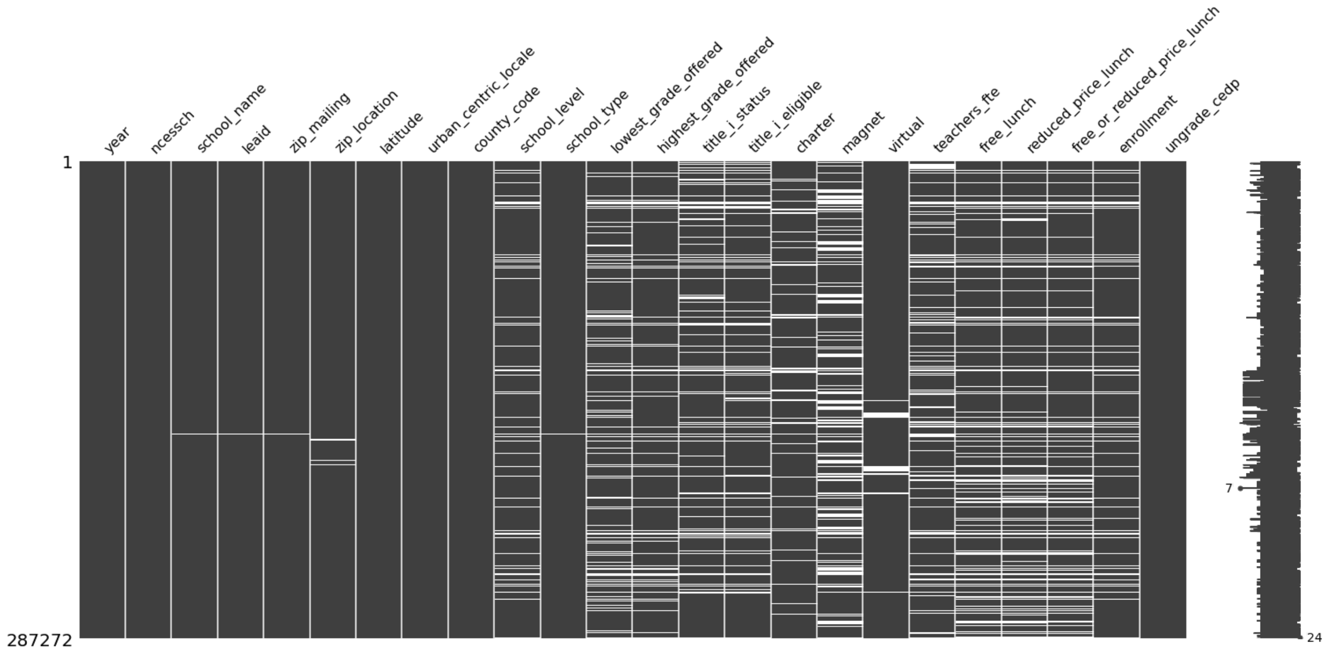
Dataset 2.Schools Common Core of Data Directory (CCD)

Since the CCD file covers a much longer timespan than our other selected datasets, we began by saving only a subset which matched the years of the ACGR, SY 2010-2011 through 2018-2019. We also removed most of the columns whose data matched that in the ACGR; however, we did retain two duplicated columns: 1) the National Center for Education Statistics School Identification Code (NCESSCH) which would later be used for merging the datasets and 2) the Local Education Agency Identification (LEAID) number, which would be used for verification that the merge worked correctly. A large number of schools were missing an identifier for physical/virtual, so we decided to save these instances by imputing them as physical due to the lack of any virtual-related terminology in the names (virtual, electronic, digital, etc.) and the rarity of virtual schools, especially in the pre-COVID era. The other features in this dataset with recurrent missingness (see Figure 4, p.8) did not have reasonable resolutions for substituting missing values, so such instances were removed. We also dropped instances with questionable values (e.g. enrollment less than one, more subsidized lunches than enrollment) and schools that do not go up to the 12th grade. After cleaning the dataset through these steps, the intermediate file was saved to local storage.

Dataset 3. Assessment Participation

Like the ACGR dataset, the Assessment Participation data was not available in aggregate but divided into files by year and subject (math and reading/language arts). We started the wrangling process by removing data from any years that do not match our previous two data types. Then, we cleaned each remaining file by removing all of the columns except those of interest to this project, leaving only the NCESSCH number and the percent participation in the assessment for that file’s subject. From this, all of the math data files were combined and the reading/language arts files were combined, while adding a new ‘Year’ column. These two datasets were saved locally before being merged together into a single dataframe. This final participation dataset was cleaned by dropping any instances without any value for participation percentage in either subject (>22,000 rows) and by calculating the midpoint of any value provided as a range. This intermediate file was saved to local storage.

Figure 4. Nullity matrix providing a visual representation of missingness in the CCD dataset



Note: Each of the features in the dataframe is associated with a labeled column, and the total number (24 features) is displayed on the lower right. The thin stripes within the columns represent the instances in the dataframe (numbered from 1 in the upper left to 287,272 in the lower left) and are filled in dark gray for present values and left white for missing values. The sparkline (the right-most, unlabeled column) is meant to show missingness per instance but shows little detail with this high number of instances.

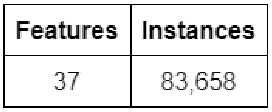
Dataset 4. County-Level Unemployment

Although the County-Level Unemployment dataset did not have instances per school, we decided to include this information anyway because it includes other features on which we would be able to merge it with the other datasets and because it adds a new dynamic of information for modeling. After dropping irrelevant columns, we reshaped the dataframe from having separate features for each year to having one set of features including ‘Year’ so that it has the same format as the other datasets.

Merging

At this point, we merged the three cleaned school-level datasets using the ‘NCESSCH’ and ‘Year’ features. Then, we added the county-level dataset using ‘Year’ and ‘Area_name’ (county name). Once all of our datasets had been merged together, we visually validated that the school district names appeared to match the area names for the counties to make sure the two are consistent since they are from two different datasets. We also dropped duplicate features, title-cased the column names, and renamed some columns for clarity. The final result was saved to local disk.This combined dataset produced during ingestion (shape shown in Table 2) would serve as the input for the EDA and Feature Engineering phases to follow.

Table 2. Shape of dataset after wrangling and merging



Features	Instances
37	83,658

Exploratory Data Analysis

Once we had satisfactorily dealt with the unnecessary, missing, or problematic aspects of our raw data and combined them into a single dataframe, we used a variety of exploratory data analysis (EDA) techniques to explore the data through numerical, tabular, and graphical investigations. This was to help us better understand the data and notice areas of interest, concern, or bias. While the dataset was not being modified during this step, the trends discovered and knowledge learned would be an important part of the model design and preparation.

Sample Population

Our study attempts to estimate characteristics of the public school population; however, during the wrangling process, some instances (i.e. school+year) had to be removed from our dataset due to missing, misformatted, or questionable values in our features of interest. Because of this, we needed to understand the ways that our dataframe may no longer be representative of reality or if we had a complete and accurate set of data.

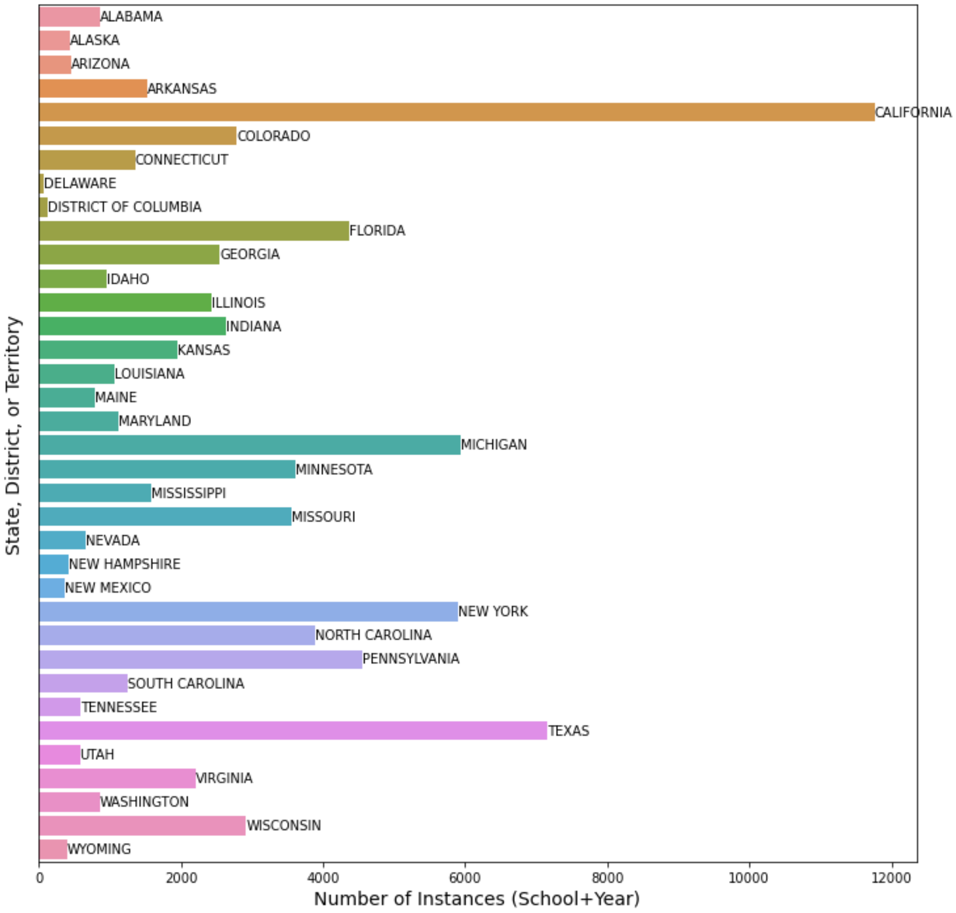
Of the 163,836 instances reported by the U.S. Department of Education during 2012-2018, only 137,653 (84%) were included in the raw data files and only 83,658 (51%) made it through the wrangling process [14]. Even within these, the distribution by year is uneven as shown in Table 3, p.10, meaning that the set of schools included in the dataframe varies from year to year.

Table 3. Percentage of schools from each year in our sample population

Year	2012	2013	2014	2015	2016	2017	2018
%	14.1	14.4	11.7	12.0	15.1	16.3	16.3

Additionally, we looked into the instance distribution by state to see if any inferences could be drawn from that. Although this visualization in Figure 5 cannot show us the difference between number of schools in our dataset and number of schools in actuality, it did make us aware that West Virginia and Puerto Rico are missing entirely, a sign of possible selection bias.

Figure 5. Instances per state

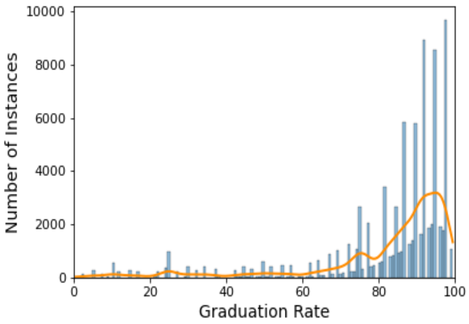


Notable Features

By further exploring our dataset, we were able to note several other compelling aspects of the dataset and note how they might affect our eventual model.

Our target variable, graduation rate (GR), is measured in percent and has a mean of 82% and a median of 90% for our sample population, meaning that half of the schools have a dropout rate larger than 1 out of every 10 students. Figure 6 shows our dataset's full distribution.

Figure 6. Graduation rate histogram of the number of instances (school+year) at each possible graduation rate



Note: The orange line is the kernel density estimate, a smoothing function.

Despite having to remove the smallest schools in our original dataset, the high school graduating cohort size is heavily right-skewed with a mean of 200 students and a median of 126 students, even including schools with cohorts as high as 2651. Due to this distribution (shown in Figure 7), we can infer that most models may be less accurate for schools with very large cohort sizes, since they would be trained mostly on instances of schools with much smaller cohorts. This is worth consideration, since several of the largest cohorts in our dataset are from virtual schools. Since this dataset ends at SY 2018, post-Covid data is likely to show more, large virtual schools, as reports state that virtual school enrollment in 2020 and 2021 are 170-180% of pre-Covid levels [15].

Figure 7. Cohort size histogram

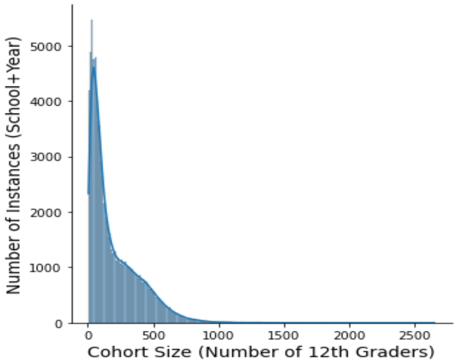
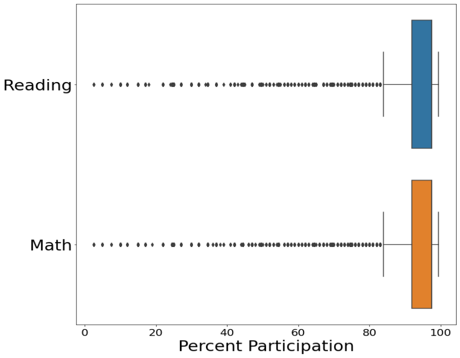


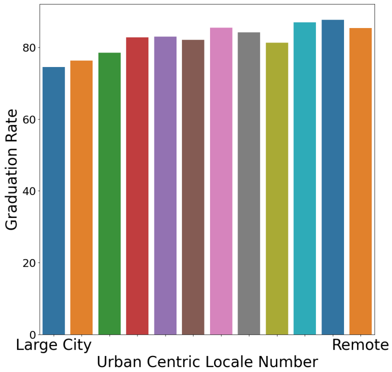
Figure 8. State assessment participation boxplots by subject



State assessment data was collected for the subject areas of math and reading/language arts. The percent of students participating in these assessments is very high for most schools but also has many lower outliers, as shown by the long tails on the boxplots in Figure 8. Although these boxplots appear very similar to each other, we would discover later that the reading/language arts participation would be a much more important feature in our modeling phase regarding predicting graduation rate.

The NCES released urban centric locale codes to categorize schools by the urban milieu they inhabit [16]. Figure 9 shows the diversity of graduation rate outcomes among this feature. Here “Large City” is defined as an urban area inside of a large city with a population of 350K or more, and a location more than 25 files from an urbanized area and more than 10 miles from an urban cluster would be categorized as “Remote.” (All 12 of the category names and definitions can be found in [16].) We were surprised to see the trend in Figure 9 that more urban areas tended to have lower graduation rates and more rural areas have higher graduation rates; our apparently-biased assumption was that it would be the opposite.

Figure 9. Mean graduation rate for each of the twelve urban centric locale codes



We found it important to note that additional features in our data (e.g. the USDA Urban Influence Codes defined in 2013) are similar to this one in that they aim to represent urbanicity, population density, etc. though they may characterize these concepts in different ways [17]. Having multiple sources attempt to represent the same indistinct meaning was thought helpful to correct for inaccuracies in either source.

What would turn out to be the top feature in feature importance calculations with both linear and random forest models was School Type. We had not expected this feature to be so impactful in our models, because this feature’s values are highly homogeneous with 89% being “Regular” schools (see Figure 10). However, the difference in graduation rates between school types is the key aspect of this feature. As can be seen in Figure 11, “Regular” and “Vocational” schools have graduation rates above the total mean of 82%, while “Special Education” and “Alternate” schools fall far below this.

Figure 10. Count plot (bar graph) of the number of instances of each school type

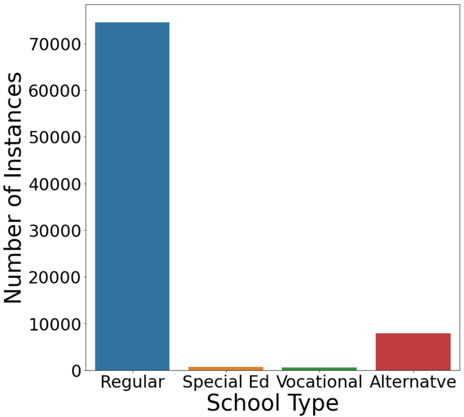
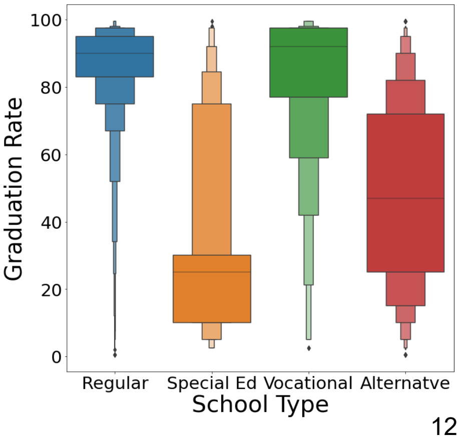


Figure 11. Boxenplot of graduation rate for the School Type categorical variable



Section 2. Feature Engineering

After discovering trends, opportunities, and limitations within our data during EDA, we transitioned to the feature engineering process. This part of the data science pipeline is crucial to the success of a project, as expressed by data scientists, professors, and authors Brad Boehmke and Brandon Greenwell:

> “Data preprocessing and engineering techniques generally refer to the addition, deletion, or transformation of data…feature engineering can make or break an algorithm’s predictive ability and deserves your continued focus and education.” [18]

Encoding and Scaling

Categorical variables cannot be utilized as is and require preprocessing into a form that can be read by models. Therefore, we encoded them using a “one-hot” scheme that would exchange each categorical feature with a series of numerical features, each representing one of the possible categorical values of the original column. Then, it would fill these new columns with 0 or 1 (for “No” or “Yes”) regarding whether that column’s assigned value was the same as the value in the original column. This can be better understood through Table 3 below. Note how this causes the column count to increase from one to four.

Table 4. Effect of one-hot encoding on a categorical variable

	Before	**After**			
Name	School_Type	School_Type_1	School_Type_2	School_Type_3	School_Type_4
Value	1, 2, 3, or 4	0 or 1	0 or 1	0 or 1	0 or 1

Henceforth, all figures will show evidence of this, for example, including School_Type_1, School_Type_2, etc. but not School_Type.

For models that require it, numerical features were centered and normalized using a normalization scheme, giving a mean of zero and variance of 1. Note that this was not performed until the time of modeling to avoid data leakage between cross-validation folds. We also tested another scheme, centering and scaling with subtraction and division, but this process reduced performance on this data and was therefore not included.

Feature Addition

Having finished formatting all of our data features into numerical values, our next step was to optimize our data for modeling by imputing new features that our experience in the field along with common sense logic suggested could be useful for a predictive model.

Table 5. Features imputed by the project team

Feature Name	Description
Num_Grades	(Highest grade offered) - (Lowest grade offered)
Teacher_Ratio1	(Teacher full time equivalents) / (Total school enrollment)
Teacher_Ratio2	(Teacher full time equivalents) / (Cohort size)

After creating these three features, we added second-order interaction terms for the whole dataset. These are features that can improve the model by turning non-linear relationships into linear terms and allow us to compare the modeling power of our linear model to more complicated models. Note that certain models (like generalized linear models) are more likely to benefit from interaction terms than others, based on whether they inherently capture non-linearity like the random forest class does. With these additional terms, we computed the feature importance using a random forest regressor model and parameters obtained during preliminary tuning.

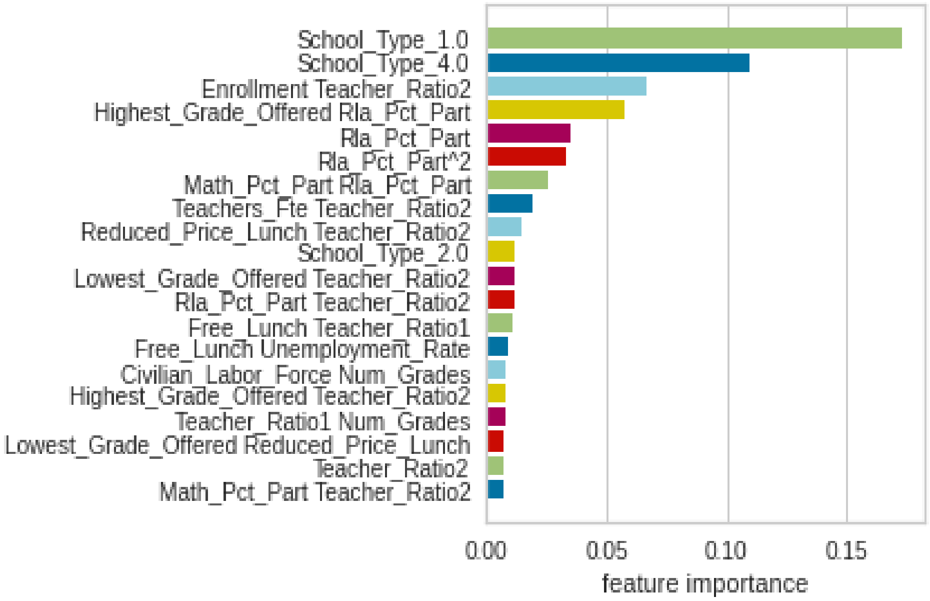
The resulting feature importance ranking with the interaction terms included is shown in Figure 12, p.15. Because adding the interaction terms vastly increases the number of features in our dataset, running the feature reduction with them became a resource limitation concern (e.g. 15+ hours with 4 cores of the machine running). Due to this issue, we chose to continue further modeling only including 5 of the top interaction features (listed in Table 6 below).

Note that the “discovery” of these terms- and their basis in real world functioning- are intriguing to the authors as a route for further inquiry. Why should combining enrollment, teacher count, and cohort size (the most impactful interaction term, as shown in Table 6 and Figure 12, p.15) be such a good predictor of graduation rate? It cannot merely be a proxy for school size since other terms are more obviously related to that aspect, so additional study and research would be needed to understand the relationships between these factors.

Table 6. Interaction terms added to the dataset

Name	Description
Enrollment Teacher_Ratio2	(Total enrollment) x (Teacher ratio 2)
Highest_Grade_Offered Rla_Pct_Part	(Highest grade) x (Reading test participation)
Rla_Pct_Pt^2	(Reading test participation) squared
Math_Pct_Part Rla_Pct_Part	(Math test participation) x (Reading test participation)
Reduced_Price_Lunch Teacher_Ratio2	(Students on lunch subsidy) x (Teacher Ratio 2)

Figure 12. Top 20 features after one-hot encoding and second-order interaction term creation



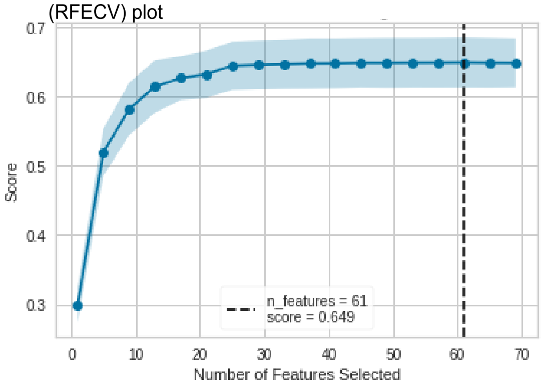
Note: Here, “feature Importance” has been determined by measuring the total reduction in sum of squared error (SSE), which is positive even if the term reduces the target.

Feature Reduction

After the feature addition steps, our dataset included 69 features, so we next utilized the *Yellowbrick* package’s Recursive Feature Elimination method to determine how many of those features were actually adding significant predictive power to the model. This provided a clear path to reduce the size and complexity of the dataset used in the final modeling phase. Reducing the number of features has several benefits, including shorter modeling times, improved interpretability, and a decreased chance of overfitting.

The results of the process in Figure 13 show an algorithm-determined optimal point at 61 features, but visually, we can see that model performance actually levels off at about 30 features. This indicates that the additional features have miniscule benefit on the model score and would be better to remove from our dataset. Out of an abundance of caution, we selected 35 as the number of the predictive features to keep for our final dataset because we

Figure 13. Recursive feature elimination cross validation (RFECV) plot



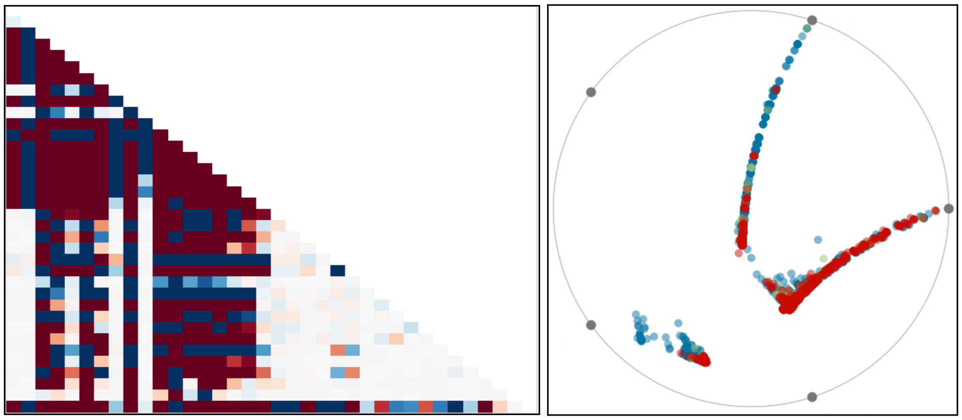
wanted to try several model types that will not all rank the importance exactly the same. To obtain the final feature selection, we trained a random forest regressor model on the entire dataset, sorted the features by the size of their reduction in SSE, and kept the top 35. The final set of features selected for modeling after the process of feature engineering is listed and described in Appendix B.

Feature Visualization

The shape of our final modeling dataset was 83,658 instances and 40 features (including the target variable and four identifiers for validation purposes).

Separability of groups of instances with a similar target value is critical to a model performing well. A higher separability means that the model will better be able to make accurate determinations based on a given value or set of values. We analyzed this property with *Yellowbrick's* Rank2D and RadViz classes with the graduation rate separated into three equally-populated bins: low (0), medium (1), and high (2). Meaningful but imperfect separability can be seen in the example results in the RadViz plot in Figure 14, where the almost vertical blue line and the cluster in the lower left are well separated. The long red line in the lower right has high overlap with blue and green circles. This is consistent with our modeling scores, which are strong, being in the 70-80% accuracy range but are not 90% or above.

Figure 14. Examples of *Yellowbrick* visualizations of the separability and covariance of key features (Rank2D left, RadViz right)



Note: We used a variety of feature visualization classes from *Yellowbrick* (including Rank2D and RadViz) to look at the correlations and separability of our data. To see all of the full visualizations, check out the Feature Engineering section of our project GitHub repository.

Section 3. Machine Learning

Model Types

We selected several regression models to test. The model type and its respective *scikit-learn* implementation class are as follows:

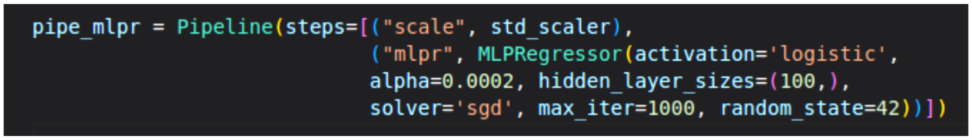
- generalized linear model (ElasticNet)
- support vector machine (SVR)
- neural network (Multilayer Perceptron)
- boosted decision tree (AdaBoost)
- decision tree ensemble (RandomForest)

Optimized parameters were saved in JSON (JavaScript Object Notation) format to the logs folder of the project's GitHub.

Note on Pipelines

The *scikit-learn* Pipeline class was used to prevent subtle data leakage between cross-validation folds and succinctly communicate code to technical stakeholders, i.e. colleagues, production engineers, or customer engineers. An example of this type of code is demonstrated in Figure 15 below. After using *Yellowbrick* for model evaluation*,* we used a different *scikit-learn* companion library, *Feature-engine*, for feature transformation (documentation available in Appendix A). The transformers in this library keep the data as a *pandas* dataframe at each step of modeling, allowing for selection by name, tighter pipeline code, and easier debugging.

Figure 15. Pipeline example code with two steps, scaling and estimating



```
pipe_mlpr = Pipeline(steps=[("scale", std_scaler),
                            ("mlpr", MLPRegressor(activation='logistic',
                            alpha=0.0002, hidden_layer_sizes=(100,),
                            solver='sgd', max_iter=1000, random_state=42))])
```

Note: The std_scaler variable is a *feature-engine* SklearnTransformWrapper class around StandardScaler

Preprocessing

After thorough data wrangling and feature engineering processes, the only additional preprocessing required before modeling was the *scikit-learn* StandardScaler class (documentation available in Appendix A). A MinMaxScaler transformation did not perform as well so we did not use it.

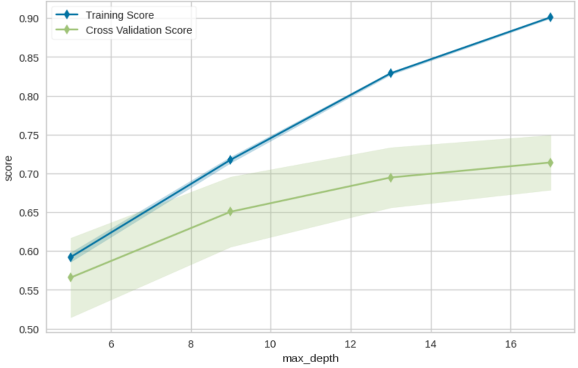
Train-Test-Split

We split our dataset into two sets according to best practices: 1) 70% for training and 2) 30% for testing. We incorporated stratification on the target to maintain equivalent (and verified) distributions of graduation rates in both sets.

Hyperparameter Tuning

Two rounds of grid search tuning were done for each model using the GridSearchCV class and an industry standard 10 cross validation folds. To control the complexity of random forest models, we utilized early stopping (aka pre-pruning) by capping tree depth at 13. Overfitting is evident when the test score stops improving or decreases while the training score is still improving. In Figure 16, this can be seen at depth 13; thus, we chose to implement this cap to prevent overfitting.

Figure 16. Validation curve for RandomForestRegressor showing



After this process of tuning, the final parameters used for modeling were set and can be seen in Table 7.

Table 7. Hyperparameters- Random seed set to 42 where applicable

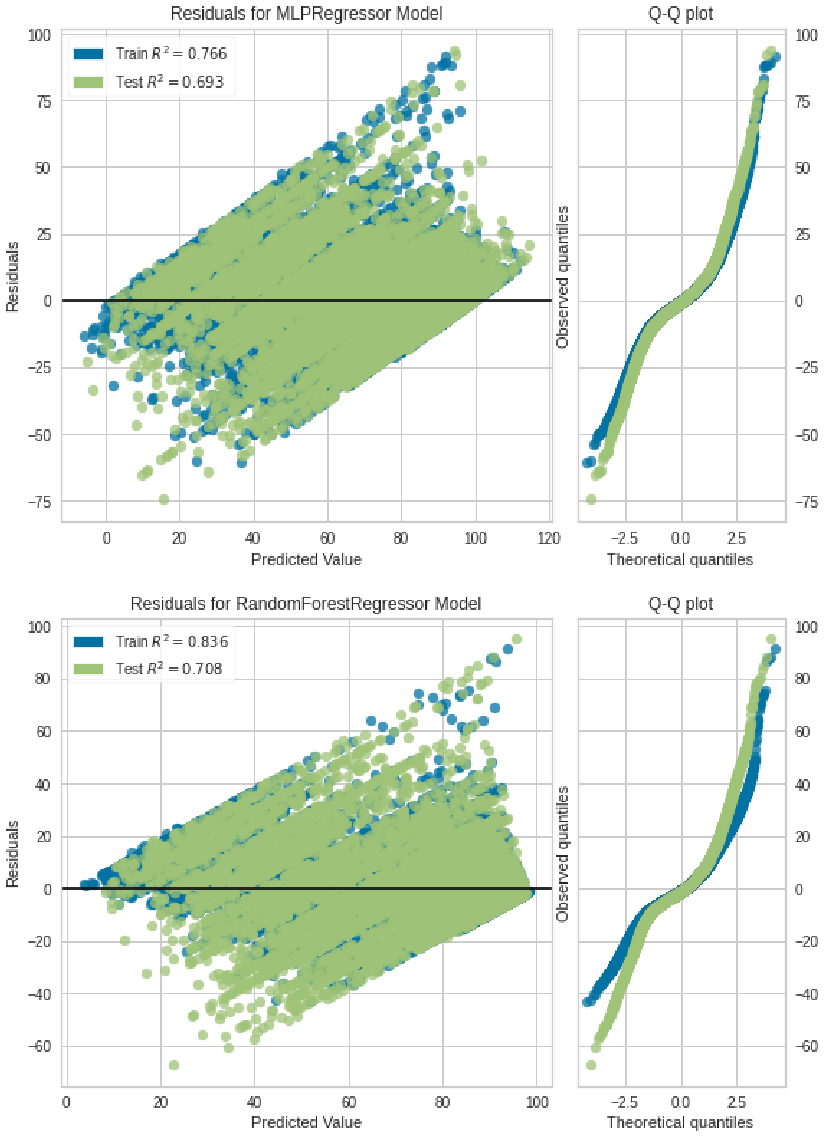
Model	ElasticNet	SVR	MLPR	AdaBoost	Random Forest
Params	L1 ratio 1	C 7 Epsilon 0.6 Kernel ”rbf”	Activation logistic Alpha 2E-3 Hidden layer 100x1 Solver sgd Max iterations 1000	Estimators 800 Learning rate 0.1 Loss exponential Max features 0.9 Max depth 11	Estimators 1000 Max features 0.8 Max depth 13

Model Evaluation and Selection

We evaluated regression results using the *Yellowbrick* package’s Residuals Plot visualizer. The plots of the two top performing models are shown in Figure 16, p.19. The Q-Q Plot side plot was added as a quick assessment of how normal the errors were. Model errors far away from normal can indicate more optimization is needed. In addition, we calculated the root mean squared error (RMSE) to get a sense of the typical deviation.

We found that adding 1/400th of a standard deviation of noise to the training data changed the metrics by +0.005 (train R^2), -0.012 (test R^2), and +0.23 (RMSE). This is supportive evidence that the model will generalize acceptably. If it had been extreme, we would have suspected our model was memorizing certain specific values.

Figure 16. Residuals plots for 2 best models: MLPRegressor (top) and RandomForestRegressor (bottom)



Through this process, we were able to determine the Train R^2, Test R^2, and RMSE of each of our tested models. These values were collected in Table 8 below. Because the RandomForest Regressor is highest in both R^2 values (a measure of accuracy) and also has the lowest RMSE value (a measure of error), we determined this to be our best model.

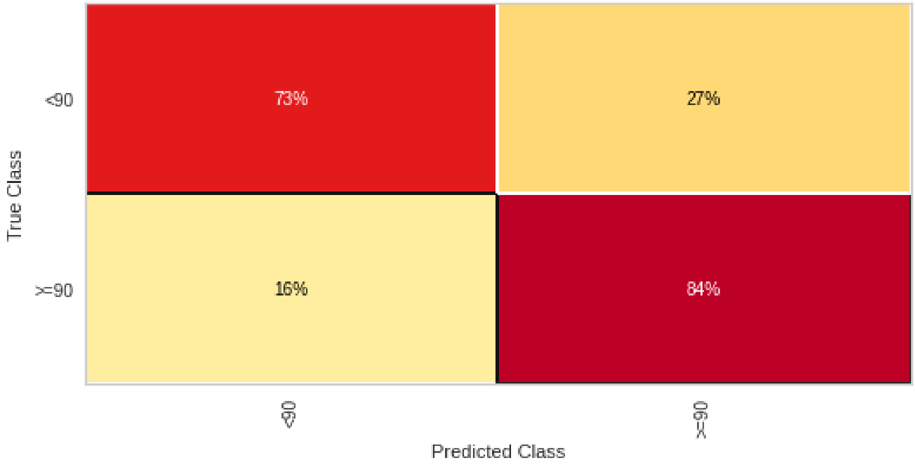
Table 8. Evaluation metrics of tested models (selected final model in blue)

Model	**Train R^2**	**Test R^2**	**RMSE**
ElasticNet (Lasso penalty)	0.554	0.553	12.99
SVR	0.687	0.660	11.33
AdaBoost	0.830	0.679	11.00
Multilayer Perceptron	0.766	0.693	10.77
RandomForest	0.836	0.708	10.48

Classification Results

Although we liked the possibility for greater accuracy and more refined target options, we decided also to perform brief classification modeling (still at depth 13) to learn how well a classification-type model would perform with this dataset. We started by binning the target variable into two equal populations with the split at the median 90% value. The confusion matrix in Figure 17 shows that the model was 11% more accurate at predicting the >90% class with an F1 score of 0.789. While this model is neither great nor terrible at predicting a school's graduation rate in the correct of the two ranges, the ranges themselves are too large to be helpful for our objectives.

Figure 17. RandomForestClassifier Confusion Matrix for the <90% / ≥90% binary classifier



Deployment

Once we had finalized our decision to use the RandomForest model, we created a Jupyter notebook that loads the saved model file and asks the user to input feature values. For each feature, the user is shown summary statistics and is sent a warning if their inputted value is outside of the range seen during training. When all of the input values are added, the notebook displays the graduation rate predicted by the model. While currently only an early draft application, this rudimentary user interface could be passed on to production to operationalize the model. The features affecting the model are listed and described in Appendix B.

Conclusion

Our hypothesis predicted that economic factors would be the most significant. After completing the project, however, we discovered that a number economic and also other factors are predictive of adjusted cohort graduation rates including the following:

- school type
- state assessment test participation
- total teacher to cohort size ratio
- urban versus rural environment of the school
- school lunch subsidies
- unemployment rate
- charter and virtual designations

These features were more predictive than we had expected, as we were able to obtain roughly 70% accurate regression scores using such limited data. Recognizing that any found correlation cannot be considered the same as causation, the results are still encouraging and possibly useful. Further, we can conclude that a classification of the nation's secondary schools into over and under groups around the ACGR of 90% is possible with this limited dataset, with an F1 of roughly 80%.

Interesting trends we found are the following:

- Vocational and regular schools have a much higher graduation rate than special education and alternative schools.
- Proximity of schools to cities and the size of the cities is negatively correlated with graduation rate and- roughly speaking, though it is not monotonic- the graduation rate rise as we move outward from large city centers.
- State assessment participation correlates with graduation rates fairly strongly with reading having a larger effect than math.

Clearly, the effect of school-level and county-level variables on graduation rate is complex and nonlinear. Through this project, we were able to find that RandomForest models capture these better than the other models we tested.

Further Work

- Model racial and ethnic group subpopulation data: Each SEA has some flexibility in determining the major racial/ethnic groups it will use for reporting, and due to states' varied implementation of the Elementary and Secondary Education Act, not all possible groupings of racial/ethnic identification are reported in the same way as individual subgroups. Therefore, information collected this way can be missing or incomplete, not counting every student in this type of feature.
- Further optimization, e.g. selectively applying different scaling schemes to different subsets of features
- Further model validation including partial dependence plots
- Hypothesis testing for whether our sample represents the full population accurately
- Deploy our model to an Amazon EC2 instance: Early in the project, we successfully deployed a *bokeh* prototype there but have not had the chance to do so with our completed project.

Lessons Learned / Reflections

Blake

- Save feature metadata in machine readable format early.
- EDA and model validation are first class citizens with the rest of the data science pipeline.
- Ingestion and wrangling can be very time consuming; real data is messy.
- This project and program tied together many different data science and software engineering topics for me. The group nature brought useful lessons in team dynamics too.

Hayat

There was a sizable amount of learning to be done. Youtube, Coursera and Stackoverflow became my student companions throughout this certificate program.The machine learning portion in particular required significant additional research. What is MLP and the difference between SVR and SVM, what does Yellowbrick's confusion matrix do? Additionally, identifying each group member's strengths and motivation so that we can contribute in a meaningful way was also a challenge. Overall, the experience was highly rewarding.

Amy

- ➔ Organize early- trying to start later might actually be too late
- ➔ Anytime I see/hear the word "recommended," mentally replace it with "required"
- ➔ Practice, practice, practice coding. The classes assign little to no homework outside of the capstone assignments, so utilize youtube, coursera, codecademy, codewars, or even a tutor to get outside project (major or mini) to work on
- ➔ Data documentation can be just as important as the data itself
- ➔ Be clear with myself and others, whether it's communicating team member responsibilities or utilizing markdowns in coding and notebook

Acknowledgments

We would like to thank all of our instructors for their time both during lectures and outside of them in office hours. We would also like to thank our capstone advisor for advice, suggestions, and multiple check-in meetings throughout the course of the certificate program. Thank you for your time:

- Molly Morrison
- Kristen McIntyre
- Sam Goodgame
- Allen Leis
- Blake Bledar Zenuni
- Garin Kessler
- Prema Roman
- Kyle Rossetti

Glossary of Terms and Acronyms

AAPI	Asian American and Pacific Islander
ACGR	Adjusted Cohort Graduation Rate; describes the number of students in the graduating class after adjusting for students who join mid-year or who transfer to another school or pass away
API	Application programming interface. How your code accessed third-party code, whether on local computer or the internet
AY	Academic Year, e.g. School year 2012-2013 is equivalent to cohort year 2012
BIE	Bureau of Indian Education
CCD	Common Core of Data
Cohort	Group of students with same or similar expected graduation date
Cohort Year	e.g. A cohort year of 2012 is equivalent to 2012-2013
DoDEA	Department of Defense Education Activity
FRPL	Free or reduced price-lunch
FTE	Full-time equivalent
Instance	A single occurrence of features, often a row of data
LEA	Local Educational Agency (school division)
LEAID	LEA Identification Number
NCES	National Center for Education Statistics
NCESSCH	NCES School Identification Code- Unique 12 digit identifying number for public schools in the United States (a 7 digit agency ID + a 5 digit NCES school ID)
RMSE	Root mean squared error
SEA	State Educational Agency; this could be the board of education or other agency designated to oversee public k-12 education
SSE	Sum of squared error
SY	School Year, e.g. School year 2012-2013 is equivalent to cohort year 2012
USDA	United States Department of Agriculture
USVI	United States Virgin Islands
WORM	Write Once, Read Many

References

[1] Office of Elementary and Secondary Education, “Consolidated State Performance Report, 2018–19,” U.S. Department of Education, Feb. 2021. Accessed: Dec. 2022. [Online]. Available: https://nces.ed.gov/programs/digest/d20/tables/dt20_219.46.asp

[2] National Center for Education Statistics, “EDFacts file 150, Data Group 695, and EDFacts file 151, Data Group 696, 2018–19,” U.S. Department of Education, Feb. 2021. Accessed: Dec. 2022. [Online]. Available: https://nces.ed.gov/programs/digest/d20/tables/dt20_219.46.asp

[3] M. Hanson. “U.S. Public Education Spending Statistics.” EducationData.org. Accessed: Dec. 2022. [Online]. Available: https://educationdata.org/public-education-spending-statistics

[4] U.S. Bureau of Labor Statistics, "Education pays, 2021," *Career Outlook*, May 2022. Accessed: Dec. 2022. [Online]. Available: https://www.bls.gov/careeroutlook/2022/data-on-display/education-pays.htm

[5] J. DelPilar. “‘We don't like it either:' Metro school board member discusses high school dropout factory.” Fox 17 WZTV Nashville. Accessed: Dec. 2022. [Online.] Available: https://fox17.com/news/local/we-dont-like-it-either-metro-school-board-member-discusses-high-school-dropout-factory-crisis-in-the-classroom-nashville-middle-tennessee-area-local-news

[6] National Center for Education Statistics, “Public High School Graduation Rates,” *Condition of Education*, 2022. U.S. Department of Education, Institute of Education Sciences. Accessed: Dec. 2022. [Online.] Available: https://nces.ed.gov/programs/coe/indicator/coi

[7] M. Moors. “Henrico Schools addressing decline in graduation rates.” NBC12 WWBT. Accessed: Dec. 2022. [Online.] Available: https://www.nbc12.com/2022/11/21/henrico-schools-addressing-decline-graduation-rates/

[8] U.S. Department of Education, Nov. 2022, “Adjusted Cohort Graduation Rate,” EDFacts. [Online]. Available: https://www2.ed.gov/about/inits/ed/edfacts/data-files/index.html

[9] Urban Institute, Jun. 2022, “Schools CCD Directory, 1986–2020,” Education Data Portal (Version 0.16.0). [Online]. Available: https://educationdata.urban.org/documentation/schools.html#ccd_directory

[10] U.S. Department of Education, Nov. 2022, “Assessment Participation,” EDFacts. [Online]. Available: https://www2.ed.gov/about/inits/ed/edfacts/data-files/index.html

[11] U.S. Department of Agriculture, Jun. 2022, “Unemployment and median household income for the U.S., States, and counties, 2000-2021,” Economic Research Service. [Online]. Available: https://www.ers.usda.gov/data-products/county-level-data-sets/county-level-data-sets-download-data/

[12] Amazon Web Services, Inc. “Amazon S3.” aws. Accessed: Dec. 2022. [Online.] Available: https://aws.amazon.com/s3/

[13] U.S. Department of Education. *Four-Year Adjusted-Cohort Graduation Rates - School Year 2018-19 EDFacts Data Documentation*. (2020). Accessed: Dec. 2020. [Online]. Available: https://www2.ed.gov/about/inits/ed/edfacts/data-files/acgr-sy2018-19-public-file-documentation.docx

[14] National Center for Education Statistics, “Number of educational institutions, by level and control of institution: 2009–10 through 2019–20,” Digest of Education Statistics, 2022. U.S. Department of Education, Institute of Education Sciences, Mar. 2022. Accessed: Dec. 2022. [Online.] Available: https://nces.ed.gov/programs/digest/d21/tables/dt21_105.50.asp

[15] A. Lehrer-Small. “Virtual School Enrollment Kept Climbing Even As COVID Receded, New Data Reveal.” The74. Accessed: Dec. 2022. [Online.] Available: https://www.the74million.org/article/virtual-school-enrollment-kept-climbing-even-as-covid-receded-new-data-reveal/

[16] *Standards for Defining Metropolitan and Micropolitan Statistical Areas*, Federal Register (65) No. 249, Office of Management and Budget, Dec. 2000. [Online]. Available: https://nces.ed.gov/pubs2007/ruraled/exhibit_a.asp

[17] J. Cromartie, May 2013, “2013 Urban Influence Codes,” U.S. Department of Agriculture Economic Research Service. [Online]. Available: https://www.ers.usda.gov/data-products/urban-influence-codes/

[18] B. Boehmke and B. Greenwell, *Hands-On Machine Learning with R*. Boca Raton, FL, USA: CRC Press, 2020, pp. 41.

Appendix A: Data Science Tools

Boto3 API Reference (v1.26.25)
https://boto3.amazonaws.com/v1/documentation/api/latest/guide/new.html

Feature-Engine API Reference (v1.5.2)
https://feature-engine.readthedocs.io/en/latest/

Joblib API Reference (v1.2.0)
https://joblib.readthedocs.io/en/latest/

Matplotlib API Reference (v3.62)
https://matplotlib.org/stable/api/index.html

Pandas API Reference (v1.5.2)
https://pandas.pydata.org/docs/reference/index.html

Python3 API Reference (v3.10.8)
https://www.python.org/downloads/release/python-3108/

Seaborn API Reference (v0.12.1)
https://seaborn.pydata.org/api.html

Sklearn API Reference (v1.2.0)
https://scikit-learn.org/stable/modules/classes.html

Yellowbrick API Reference (v1.5)
https://www.scikit-yb.org/en/latest/api/

Appendix B: Deployed Model Predictive Features by Importance

	Feature Name	Description	Source
1	School_Type_1.0	school is a regular school	CCD
2	School_Type_4.0	other/alternative school	CCD
3	Enrollment Teacher_Ratio2	enrollment x teacher ratio 2	Interaction
4	Highest_Grade_Offered Rla_Pct_Part	highest grade offered x reading participation	Interaction
5	Rla_Pct_Part	percent who took state reading test	Ed-AP
6	Rla_Pct_Part ^2	reading participation squared	Interaction
7	Reduced_Price_Lunch Teacher_Ratio2	# kids with reduced price lunch x teacher ratio 2	Interaction
8	Teacher_Ratio2	# teachers / number of students who could have graduated	Imputed
9	Math_Pct_Part Rla_Pct_Part	math participation x reading participation	Interaction
10	Teacher_Ratio1	# teachers / total enrollment in the school	Imputed
11	Free_Lunch	number of students receiving a free lunch	CCD
12	Civilian_Labor_Force	labor force in the school's county	USDA
13	All_Cohort	number of students who could have graduated	Ed-ACGR
14	Enrollment	total enrollment in school	CCD
15	Free_Or_Reduced_Price_Lunch	kids receiving either free or reduced	CCD
16	Teachers_Fte	number of teachers	CCD
17	Reduced_Price_Lunch	kids receiving reduced price lunch	CCD

	Feature Name	Description	Source
18	Math_Pct_Part	percent who took state math test	Ed-AP
19	Unemployment_Rate	unemployment rate in school's county	USDA
20	Title_I_Status_5.0	school eligible for Title I and provides it	CCD
21	Charter_0.0	school is NOT a charter school	CCD
22	Urban_Centric_Locale_11.0	school is in a large City	CCD
23	Num_Grades	number of grades offered at school	Imputed
24	Lowest_Grade_Offered	lowest grade offered at the school	CCD
25	Virtual_1.0	school is virtual	CCD
26	Metro_Or_Not_0.0	school is NOT in a metro area	USDA
27	Rural_Urban_Continuum_Code_1.0	metro in county of 1 million or more people	USDA
28	Title_I_Eligible_1.0	eligible for state Title I program	CCD
29	Title_I_Status_6.0	school is not eligible for title I or targeted assistance	CCD
30	Urban_Influence_Code_1.0	large in metro area with 1 million or more	USDA
31	School_Level_4.0	school level designation "other"	CCD
32	Urban_Centric_Locale_12.0	city midsize	CCD
33	School_Level_3.0	school level designation "high"	CCD
34	Virtual_0.0	not a virtual school	CCD
35	School_Type_3.0	school is a vocational school	CCD