Prediction of Post Collegiate Earnings Using the College Scorecard Dataset

Kira Mulshine

Integrated Innovation Institute, Carnegie Mellon University
4612 Forbes Ave, Pittsburgh, PA 15213

Introduction

Entering and completing a degree at a high level educational institution (college or university) is a large time and energy investment for students globally. In the United States, it is also an immense financial investment for most people who attend, though many students still do enroll, because it is generally agreed that completing a four year undergraduate degree will increase an individual's likelihood of being fiscally stable throughout the remainder of his or her life. Part of the reason for this is that almost all stable-income and corporate jobs list having a Bachelor's degree as a job requirement and jobs that are not corporate will often choose a candidate with a Bachelor's degree over someone who hasn't received one. Though this is believed, the number of students who enroll in higher level degree programs after high school graduation peaked in 2010 and has been declining since, from 25.2 million to 22.4 million in 2017. The below table shows this decline, which was published by Statista in 2018[1].

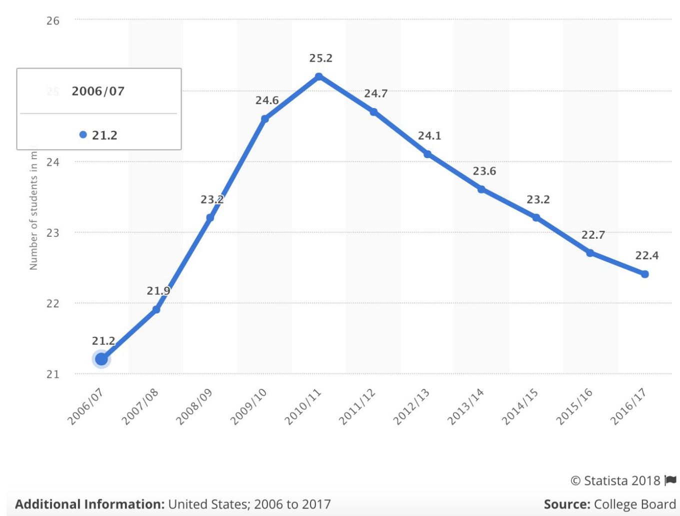

Table 1: Undergraduate Enrollment in Universities in the United States
https://www.statista.com/statistics/235406/undergraduate-enrollment-in-us-universities/

[1] "Undergraduate Enrollment in U.S. Universities, 2006 to 2017 | Statistic." Statista, Statista, www.statista.com/statistics/235406/undergraduate-enrollment-in-us-universities/.

The New York Times reported in 2014 that the decline, as stated by economist Heidi Shierholz, “indicates that upward mobility may become more difficult for working-class and disadvantaged high school graduates.[2]” Though the number of students enrolling in these institutions has been slowly declining, there is still a large number of students enrolling and those that enroll hope that the time, energy and financial investments they are making by doing so pay off after graduation by securing them a more likely financially stable future.

The College Scorecard was created to be a tool for students to make data driven decisions about the institutions they choose to attend, so that they can optimize their future experiences by attending the best university for them[3]. The College Scorecard Dataset is updated on a regular basis to power the site, and it is also shared publicly, so that other people can help to improve the dataset and understanding of it to support students in making decisions about what schools to attend[4]. I am interested in exploring which features from a 2017 version of the dataset can predict whether university graduates from certain institutions make more or less money after graduating (within 6 years of graduation). The basic question I am asking is, “Will I have high salary after attending this institution?” My hope is that the model could be used to make a tool that will be useful in supporting a prospective student to make a decision about which university to attend based on recent versions of this dataset and with the goal of having a high paying job after he or she graduates.

Related Work

Other work has been done with prior versions of this dataset to predict the same outcome. For example, Monica Agrawal, Priya Ganesan, and Keith Wyngarden at Stanford University published a paper in 2015 titled “Prediction of Post-Collegiate Earnings and Debt” in which they used the 2011 dataset, because it had little missing values in comparison to more recent datasets. They determined that Locally Weighted Linear Regression (under 10% error) could be used to help fill in the gaps for missing information to help predict higher earnings after

---

[2] Norris, Floyd. “Fewer U.S. Graduates Opt for College After High School.” The New York Times, The New York Times, 25 Apr. 2014, www.nytimes.com/2014/04/26/business/fewer-us-high-school-graduates-opt-for-college.html
[3] “College Scorecard.” College Scorecard, collegescorecard.ed.gov/.
[4] “College Scorecard Data.” College Scorecard Data, College Scorecard, 17 Dec. 2017, collegescorecard.ed.gov/data/.

6 years post graduation[5]. Because this dataset is from 2011 and enrollment in universities has been declining since then, I believe there is an opportunity to refer to a more recent dataset to enhance or clean up the feature space and to perform tests on it that make up for the missing information and provide insight into the decline. I also believe that recent data should give students more accurate information about what university is best for them.

Miranda Strand and Tommy Truong at Stanford University published a paper titled “Predicting Student Earnings After College,” which took a different approach to a similar prediction problem using the 2013 College Scorecard dataset. They were aiming to predict the earnings of a student 10 years after graduation based on data specifically about the college and concluded that using Lasso and Random Forest algorithms resulted in the least error for this prediction[6]. I liked this approach because it looked at feature data that was relevant to the students; based on what is generally known about people's interests, the features it focused on were those that would likely be researched by prospective students first and intuitively when making a decision about a university to attend, such as student gender, age, ethnic, and income information at the schools. Because this would likely be the most searched for information, one can expect that more people would see it, so the correlations found between this data and the classifier would increase the positive impact by supporting more people to make the right decisions about which schools to attend. My goal with this project was to focus on similar features, but to perform tests on a more recent dataset with a class value of six years out. I focused on six years, because ten years is almost half of the lifetime a new undergraduate student right out of high school likely already experienced (they are around eighteen years of age). I am assuming that it is difficult for them to think about life ten years out because of this and that they will be more eager to use a tool if it can help them to make a decision for themselves that will give them the positive results they want in the more foreseeable future.

Data Preparation

The data set I used is published by College Scorecard and is publicly available online[7]. I downloaded a version of the dataset that was last updated on December 19, 2017 that contained 7,147 instances, each representing a different United States university and various,

---

[5] Agrawal, Monica, Priya Ganesan, and Keith Wyngarden. "Prediction of Post-Collegiate Earnings and Debt."
[6] Strand, Miranda, and Tommy Truong. "Predicting Student Earnings After College."
[7] “College Scorecard Data.” College Scorecard Data, College Scorecard, 17 Dec. 2017, collegescorecard.ed.gov/data/.

almost only numeric, features associated with it. The feature categories range from demographic information about the schools and students to debt repayment, earnings, wage and even death rate information. Upon reviewing the dataset, I realized that the data was not all relevant to my prediction task. I chose “GT_25K_P6” as my classifier, or variable to predict, which is the amount of students who are earning over $25,000 per year, as threshold earnings, after six years. The information used to create this class value was aggregated by College Scorecard from the United States Treasury website, as confirmed in their data dictionary download[8]. If this value is closer to 1, then the amount of students who make over $25,000 is higher than other schools, and students who are seeking to make more money by getting a degree at that university would be more likely to choose to go to those schools, too.

After choosing this as the classifier and further reviewing the data, I realized that a lot of the features were not relevant to my prediction class and so I removed these features. For example, I removed school name, because it is not numerical. I also removed all features with all “0” values and features related to death rates, because it seemed unlikely to influence prediction of my class value. Lastly, I removed many of the other features related to earnings and loan repayment, because they are too similar to my class value and would naturally predict it, but left some features in these categories that seemed different enough so that I could see how predictive they might be. I kept data such as ethnicity, highest degree offered, state, percent of degree awarded in various subjects and some information related to debt, loans and repayment, such as percent of students who received a student loan or the median debt of completers.

I noticed that the feature that showed a school’s religious affiliation assigned a different number to each instance for the type of religious affiliation, including numbers for “no affiliation” or “not reported”, with a total of 64 different numbers.I also noticed that most of the universities in the dataset didn’t report or affiliate with a religion at all and decided that because of this, it was more valuable to learn whether a university that did report any type of religion would help to predict the amount of money someone would make after attending that school. I then turned this into a binary feature so that “0” meant “no affiliation/no reported affiliation” and “1” meant “reported religious affiliation”.

With 48 features in my initial dataset, I then converted all NULL and “Privacy Suppressed” variables to “0’s” so that they could be usable when performing tests on the

[8] “College Scorecard Data.” College Scorecard Data, College Scorecard, 17 Dec. 2017, collegescorecard.ed.gov/data/.

numeric dataset, and then scaled all features that were not already binary or scaled at between 0 and 1. I did this with the exception of the state feature, because I wanted to see if there was a correlation between the school's location and the class value. I then divided up the data into three separate sets, as outlined in Table 2.

Dataset	Amount of instances
Train Set	(70%) 5,002
Development Set	(20%) 1,429
Test Set	(10%) 714

Table 2: Organization of Initial Dataset

Feature Selection

Before training and building the model, I wanted to learn more about my dataset and how the features interacted, because I believed there were other ways I could augment the feature space to improve learning, by potentially combining features and creating new ones. To do this, I used Attribute Selection filter and the "AttributeSelectionClassifier" with the Ranker Search method in Weka and performed a linear regression, 10 fold cross-validation test on the "Train Set" to determine which features were the best predictors of the class value. I used linear regression because the prediction task I am working on is a regression problem and linear regression is a standard algorithm to use when doing so. Table 3 shows the output of this attribute selection for reference.

```
GT_25K_P6 =
     2.0231 * MD_EARN_WNE_P10_SCALED +
     0.2239 * RPY_3YR_RT_SUPP +
     0.2064 * GRAD_DEBT_MDN_SUPP_SCALED +
    -0.5023 * PCIP54 +
     0.3766 * UGDS_SCALED +
    -0.0264 * HCM2 +
    -0.065  * PCIP39 +
    -0.0854 * PCIP12 +
```

Table 3: Output using Best First search method

As predicted, a total of 3 of the 9 features that I kept in the dataset related to earnings, loans and repayment were high predictors of the class value (Median earnings after 10 years, median grad debt, and 3 year repayment rate). I chose to remove these features, because as previously discussed, in this project I am more interested in learning what other types of features are predictors of higher income after graduating.

Three of the other features were features of the percentage of graduates who earned different types of degrees (1 = history, 2 = theology, 3 = personal and culinary services) and all have a strong negative correlation with the class value, meaning that, if the percentage of students graduating with these degrees at the university was low, then students from that university were more likely to make more money after graduation. Because of this, I decided to consolidate the features for “percent of students with a certain degree” into degree categories for type of degree, so that when training I could see what type of degree was more predictive of the class value. To do this, I hand labelled each of the “percent of students with a certain degree” features as either STEM, Liberal Arts, Trade or Social Science degrees. I then combined those in similar categories by averaging the already scaled values. By eliminating the three earnings features and consolidating the “type of degree” features, I reduced my feature space from 48 to 31.

The feature for “schools that are on Heightened Cash Monitoring by the Department of Education” remained unedited and, though the weight was close to zero (-.03), it was still one of the features chosen as a high predictor of the class value. The feature for the amount of undergraduate students enrolled in the university also remained unedited, and was a high positive predictor of the class value, which makes sense because if there are more students at a university, there are more student graduates that could possibly make money after graduating and, therefore, more that could then make over $25,000 per year after 6 years.

Baseline Performance

I then performed regression tests on my training set to test performance in predicting the class value. I chose linear regression as the algorithm to use to measure baseline performance using a 10-fold cross validation, because this is a standard algorithm used to do so for regression problems, and I also chose to do a 10-fold cross validation test using the Multilayer Perceptron algorithm to see if there might be different features that, weighted differently, came together and could be high predictors of the class value. I did these tests using Weka and changed no parameters in the process because, for baseline tests, you should use the default settings. Table 4 shows the results for both algorithms on the baseline training set data.

Algorithm	Performance	Mean Absolute Error
Linear Regression	Correl Coeff = .74	MAE = .13 (13%)
MP	Correl Coeff = .61	MAE = .18 (18%)

Table 4: Training Set - Baseline Experiment Results

Linear Regression has higher performance (.74) and lower percent error (13%) than Multilayer Perceptron (.61 and 18%), which may mean that this is a simple prediction problem that performs poorly with Multilayer Perceptron because it is over-complicated by the multiple layered neural network and weighted approach to solving the problem.

Error Analysis

I decided to perform an error analysis to determine whether there were any other features I could improve upon to enhance performance of both algorithms, and not just Linear Regression, because I thought that it was also possible that MP was performing poorly because the feature space could be further augmented. To set up for this, I performed Linear Regression and MP tests on the development baseline dataset to see how the performance of the development set compared to the training set and its baseline performance. I did this by using Weka Explorer and uploading the development set as the “supplied test set,” and then running the same default experiments on this set. Table 5 shows the results of both the training and development tests.

Algorithm	Baseline Perf (Train Set)	Mean Abs Error (Train Set)	Baseline Perf (Development Set)	Mean Abs Error (Development Set)
LR	Correl Coeff = .74	MAE = .13 (13%)	Correl Coeff =.85	MAE = .12 (12%)
MP	Correl Coeff = .61	MAE = .18 (18%)	Correl Coeff = .50	MAE = .20 (20%)

Table 5: Train & Development Set - Baseline Experiment Results

The development set performance is significantly lower for Multilayer Perceptron and the MAE increases by 2%. For Linear Regression, performance surprisingly increases and MAE decreases slightly, by 1%. This is unusual, so I looked more closely at the test results to review and noticed that the state feature was likely confusing the model for both algorithms. For linear regression, the model grouped states together and turned them into features that then had different weights, instead of weighting each state separately and providing a weight for that instance's location. Many of the states repeated occurrences in these new groupings, which would not support the prediction problem the model was trying to solve and likely confused, but since most of the weights for these categories were almost "0" in value (between -.1 and .1), they were likely not influencing it too much and just creating noise. I also looked at how the state feature influenced the MP model's performance and noticed that it weighted each state separately, as I was hoping it would, but that these weights were incredibly inconsistent across the 41 nodes. For example, for California, the weights range from -4.78 to 5.10, which is incredibly extreme. This occurred with several other states, too, which also made me believe that the states were creating noise in the data set and confusing the model with MP.

To remove this noise, I decided to change this feature into the state rankings for "Best States for Higher Education" as depicted by a dataset created and published by McKinsey & Company and released by US News [9]. I hand edited the feature by writing in the ranking for each state and scaling it so that the #1 state for higher education (Florida) would have a value of 1 and the #50 state (Pennsylvania) would have a value of 0. By replacing the noisy state feature with this new one, I hoped that performance of the model would improve.

To see if the addition of this new feature helped to improve the performance of the model by decreasing noise in the dataset, I performed a test in Weka Experimenter that compared the performance of the baseline development set and the states ranked development set for both Linear Regression and MP using 10 fold cross validation. Table 6 shows the results of this experiment.

Dataset	LR - Correl Coefficient	LR - Mean Abs Error (MAE)	MP- Correl Coefficient	MP - Mean Abs Error (MAE)
Baseline-Dev	.76	.12	.64	.18

[9] "These Are the Most Educated States in America." U.S. News & World Report, U.S. News & World Report, www.usnews.com/news/best-states/rankings/education/higher-education.

StatesRanked-Dev	.76	.13	.67	.17

Table 6: Weka Experimenter Results - Baseline vs. States Ranked) - Correlation Coefficients

The results show that for Linear Regression, adding the “states ranked feature” didn’t improve the performance, but it didn’t decrease it either. It did slightly increase the MAE by 1%, which is not significant. For MP, performance improved by .03 and MAE decreased by 1%. Though these results are not significant either, they do show that this new feature did improve performance and, in a later section of this paper, I explore whether performance of MP could be further improved by tuning the hiddenLayers parameter. I believe it is important to point out that performance of the linear regression model with the states ranked development set in the Weka Experimenter decreased from the performance with the baseline development set in the Weka Explorer from .85 to .76, while the difference in results for these same sets using the MP model significantly increased from .50 to .67, which makes me wonder if tuning MP is the stronger algorithm to use for this prediction problem and if its performance could improve even further with tuning. Still, performance with the linear regression model is higher than performance using the MP model (.76 > .67). Table 7 below shows this difference more clearly.

Dataset	Explorer Performance (Correlation Coeff)	Experimenter Performance (Correlation Coeff)
Baseline-Dev	.85	.50
StatesRanked-Dev	.76 - **decrease (.11)**	.67 - **increase (.17)**

Table 7: Performance Differences between Dev sets using Explorer vs Experimenter

Ideas for Improvement

There is an opportunity for improvement of performance for Linear Regression by tuning the attribute selection method used (default is M5 method). This may have been why it was confused by the state data before being replaced by state rankings, and why it improved so much on the baseline development dataset. Even without the noisy state feature, the selection method used could provide different performance results for this algorithm with the new states ranked feature.

Another idea for improvement is to create more new features for information about other demographics from the universities that prospective students may be interested in, but that are not in the dataset now. These include features that represent other generally known areas of interest to prospective students, such as “presence of Greek life”, “amount of clubs”, “involvement of alumni”, “physical size of campus”, “4 year on campus housing”, “presence of mandatory core curriculum,” as well as many others. These features would need to be manually sourced or created, which could take a great deal of time, but could provide valuable insight into this prediction problem. Creation and a better understanding of these features would be useful to build a tool for prospective undergraduate students as they make decisions about where to invest their time, money and brainpower with the goal of making more money after graduating.

Tuning

The Linear Regression model was confused by the noisy state feature data. Now that the noisy data was removed and replaced by state rankings, the performance of LR has decreased. I’d like to see if the performance of LR on the development set can be improved by tuning the “attributeSelectionMethod” parameter. By changing this setting, the algorithm would use different methods on each of five different folds when selecting attributes to build the model, and doing this could improve its performance. I would also like to see if the Multilayer Perceptron model could be improved by tuning the “hiddenLayers” parameter. By changing this parameter, the model would be built using a different number of hidden layers that makes up the neural network it creates to solve the prediction problem. This could influence the feature weights, interactions and, ultimately, the performance, too. To set up for tuning, I divided the “states ranked” training data into 10 folds (5 train folds, 5 test folds) and tested the performance of each training fold to determine the average baseline performance of Linear Regression with default settings on the “states ranked” training data. Table 8 shows the results on each fold and the overall baseline performance of each model with this data. As expected based on prior tests, the performance of the LR model continues to be better than the performance of the MP model.

Fold #	**LR - Baseline - Final Train Set (Correlation Coeff)**	**MP - Baseline - Final Train Set (Correlation Coeff)**
1	.67	.53
2	.75	.69
3	.71	.65

4	.70	.55
5	.76	.68

Baseline performance LR = .72
Baseline performance MP = .62

Table 8: Baseline performance for LR - Final train set - Tuning

Next, I chose three different settings to test for each algorithm. For LR, I changed the "attributeSelectionMethod" parameter settings (Setting 1 = M5 method, Setting 2 = No attribute selection, Setting 3 = Greedy method) and, for MP, I changed the "hiddenLayers" parameter settings (Setting 1 = a, Setting 2 = "2,4", Setting 3 = "1,3,5"). I tested performance of these on each training dataset fold and on the full training dataset by using the Weka Experimenter. Table 9, below, shows the results for LR and Table 10, the results for MP. Each displays fold results, the optimal setting from the 3 options and then the test set performance estimate when using this setting and testing on the test sets for each fold in the Weka Explorer. Ultimately, the overall test set performance estimate with tuning for LR is .72 and for MP is .71.

Fold #	LR - 1 M5 Method	LR - 2 No Attribute Selection	LR - 3 Greedy Method	Optimal Setting	Test set performance estimate
1	.75	.75	.75	1	.67
2	.73	.73	.73	1	.75
3	.74	.74	.74	1	.71
4	.74	.74	.74	1	.70
5	.73	.73	.73	1	.76

Test set performance estimate = .72
Table 9: Tuning Performance LR - test set performance estimate

Upon completing tuning for LR and reviewing the results, I learned that tuning this parameter would not be worth it because the optimal setting for each fold remains the default setting, so the model should be built using the M5 method as it originally had been. Because there was no change in the test set performance estimate after tuning LR, I did not complete a significance test to calculate the p value; clearly there is no significant improvement if nothing changed.

Fold #	**MP - 1 a**	**MP - 2 2,4**	**MP - 3 1,3,5**	**Optimal Setting**	**Test set performance estimate**
1	.68	.73	.67	2	.62
2	.66	.71	.65	2	.73
3	.69	.73	.66	2	.69
4	.66	.71	.66	2	.77
5	.66	.71	.65	2	.75

Test set performance estimate = .71
Table 10: Tuning Performance MP - test set performance estimate

Upon completing tuning for MP and reviewing the results, I noticed that setting 2 (hiddenLayers = 2,4) test performance estimate of .71 was higher than the baseline by .09. To evaluate the significance, I completed a t-test and calculated the p value of .05, which means that the difference is significant and tuning would be worth it for the MP model, using setting 2. Additionally, the MP model performs almost as well as the LR model with tuning, which makes me think that MP could perform well on the test set, too. Since setting 2 performed better on each fold, it would make sense to perform a test on the final test set with the MP model that uses setting 2.

Results

I decided to test the final test set (10% of the overall data) with both the LR and the MP models (MP parameter hiddenLayers = 2,4) and to see how both models performed. I expected LR to perform better, as it has done so consistently on the training and development datasets, but wanted to see how MP performed, too, because, after tuning, the performance was almost equivalent to the performance of the LR model (.72 = LR, .71 = MP). I used the Weka

Experimenter to compare performance of the final test set to the train set performance with both models. Table 11 shows the results of this test.

Dataset	LR - Correl Coeff	MP - Correl Coeff	LR - MAE	MP - MAE
Final Train Set	.70	.71	.14	.15
Final Test Set	.84	.86	.11	.11

Table 11: Final Test Set Performance - LR (default settings) and MP (hiddenLayers = 2,4)

As you can see in Table 11, the performance on the final test set improves upon the baseline (train set) for both the LR and MP models and MAE decreases for both as well. Surprisingly, the MP model performed slightly better than the LR model on both datasets, and, though .01 and .02 increases in correlation coefficient are not large enough to be statistically significant, they are still interesting and unexpected. Clearly, tuning the hiddenLayers parameter helped to improve performance of the MP, since we know that the test set performance estimate without tuning was .62. It is also interesting that MAE decreased for both LR and MP.

When comparing the performance of the datasets on the same models, you see that the correlation coefficient increased from the train set to test set (increase of .14) for the LR model and for the MP model (increase of .15). These are improvements upon the baseline for both models, and, ultimately, the MP model performs slightly better. Both models have 11% MAE, which is the lowest seen throughout testing. Based on these results, I conclude that both LR and MP with hiddenLayers set to “2,4” are strong models to use to predict higher earnings 6 years after attending a certain university.

Discussion

Completing this project taught me that building machine learning models to solve problems and to uncover insightful information is more of an art than a specific science or skill to be learned. This became especially obvious over the semester as I completed several different error analyses on my data and then edited my feature space and even the way the data was displayed (i.e. scaling or changing to binary values), based on the results of these. The results could be shockingly different depending on the edits I made or the algorithms I chose to use for these analyses.

I learned that there are an overwhelming number of variables that can influence the performance of a model, which include some items like where the data is found, who originally collected and/or annotated it, how a feature space is set up, how old the data is, how much data is missing, and the reasoning used when making decisions about how and why to use certain algorithms and to tune their parameters. Every decision made about the dataset has the potential to influence the performance of the model and to communicate a different idea or insight from the dataset depending on what is done to build it. All of this freedom gives the person who is building the model a lot of power because he or she is in charge of and trusted to provide the most useful insights from data that are then used to build products, services and to support in making all types of decisions across industries and the world. This is similar to the freedom a painter or musician has when building a piece from nothing with the goal of ultimately exposing some sort of truth through the medium of their work, too.

This project has increased my empathy for people who spend their careers in machine learning related roles because, though it is an art, they are not as free as standard artists are to create whatever type of end result they want. Because of the large scale influence the results of their work have on other people, there is ultimately tremendous pressure on them to always make the right decisions and to remain accountable for their work when they fail.

I hope that the results of this project provide further improvement upon the accuracy of the College Scorecard and help students to determine whether attending a certain institution will provide them with increased likelihood of having a higher salary after graduating. The model I built could be incorporated into a tool to help students determine this or into something like the College Scorecard online website (https://collegescorecard.ed.gov/) in an interactive visualization tool that supports people to find the answer to the basic question I was asking in this report, which was, “Will I have a high salary after attending this institution?” Having an answer to this question helps support students in picking the best undergraduate university for them, if their main goal is to make as much money as they can after graduation.

References

Agrawal, Monica, Priya Ganesan, and Keith Wyngarden. "Prediction of Post-Collegiate Earnings and Debt."

Strand, Miranda, and Tommy Truong. "Predicting Student Earnings After College."

“College Scorecard.” *College Scorecard*, collegescorecard.ed.gov/.

“College Scorecard Data.” *College Scorecard Data*, College Scorecard, 17 Dec. 2017, collegescorecard.ed.gov/data/.

Norris, Floyd. “Fewer U.S. Graduates Opt for College After High School.” *The New York Times*, The New York Times, 25 Apr. 2014, www.nytimes.com/2014/04/26/business/fewer-us-high-school-graduates-opt-for-college.html.

“These Are the Most Educated States in America.” *U.S. News & World Report*, U.S. News & World Report, www.usnews.com/news/best-states/rankings/education/higher-education.

“Undergraduate Enrollment in U.S. Universities, 2006 to 2017 | Statistic.” *Statista*, Statista, www.statista.com/statistics/235406/undergraduate-enrollment-in-us-universities/.