C964 Capstone Project

Using Data to Improve Student Performance and Set Realistic Expectations

Terrence Wideman

Table of Contents

**A1. LETTER OF TRANSMITTAL 4 A2. PROJECT RECOMMENDATION 6**

Problem Summary 6

Application Benefits 6

Application Description 7

Data Description 7

Objective and Hypotheses 7

Methodology 8

Funding Requirements 9

Stakeholders Impact 9

Data Precautions 9

Developer’s Expertise 10

**B. PROJECT PROPOSAL 10**

Problem Statement 10

Customer Summary 11

Existing System Analysis 11

Data 12

Project Methodology 12

Project Outcomes 13

Implementation Plan 13

Evaluation Plan 14

Resources and Costs 14

Timeline and Milestones 15

**D. POST-IMPLEMENTATION REPORT 16**

Project Purpose 16

Datasets 17

Data Product Code 19

Hypothesis Verification 20

Effective Visualization and Reporting 20

Accuracy Analysis 21

Application Testing 21

Application Files 22

User’s Guide 22

Summation of Learning Experience 23

E. **SOURCES 24**

A1. LETTER OF TRANSMITTAL

June 2, 2021

Mary Smith, Principal

Kilgore High School

500 N 38th Street

Kilgore, TX 76543

Dear Mrs. Smith,

As educators, we acknowledge that no two schools are the same. The demographic of each school strongly influences its culture, expectations, and performance. Because of these differences, teaching experiences can be very different across schools. Teachers can have difficulty adjusting to those differences. Establishing baselines and expectations can be difficult for teachers and administrators new to a school. Accurately assessing teacher performance is also tricky because baselines for students are ambiguous.

But what if teachers had a simple, visual tool that could help them understand their students before the year started. This tool would help them with goal setting by giving them the expected performance of their students. Teachers would also be able to use this tool to get an estimate of individual performance. Lastly and most importantly, teachers would gauge student weaknesses and seek ways to improve them early on. Teachers wouldn't have to find out in the middle of the school year that a student needs extra help. Helping teachers set realistic expectations also helps them with the setbacks and difficulties they experience throughout the year.

TeachTech makes software tools to perform those analytics on student data. The data visualizations are easy to understand. After all, we realize that most people are not familiar with statistics. In addition to providing descriptive statistics, TeachTech provides regression analysis that allows a user to estimate a student's performance based on several factors. As educators, we always strive to expand a students' capabilities, but having a baseline will prevent educators from disappointment caused by unrealistic expectations of student performance.

At this point, you are probably wondering how much this data analytics tool will cost. Here is the cost for the starting package:

$20,000 Development (200 development hours)

$10,000 Maintenance and support for two years

$10,000 Data Engineering (scraping and transforming data to work with the software tool)

The total cost with ongoing support and maintenance for two years would be approximately $40,000. This tool will help manage expectations and measure performance accurately, thereby increasing morale and student outcomes. We would love to get this system implemented at your school.

Sincerely,

Terrence Wideman

CEO @ TeachTech

A2. PROJECT RECOMMENDATION

Problem Summary

As expectations and accountability increase in the public education system, the pressure on teachers is at an all-time high. In Texas, nearly 1 in 3 teachers quit before their sixth year (Zelinski 2019). What are some of the contributing factors to this high turnover or burnout rate? From experience, I would say part of the problem is no two schools are the same. I taught at two different low-income high schools, and my experience was drastically different. The issues that each school faces are unique.

Most people go into teaching because they want to make a difference and therefore have very high standards for their student achievements. Most first-year teachers or teachers new to a school are unaware of the student's level of performance. Lack of awareness leads to unreasonably high expectations for students who have historically performed very poorly. If teachers had a more accurate baseline for their student's performance, they could set reasonable goals and accurately measure success.

Another contributing factor to the burnout of teachers is unrealistic expectations set by the administration. Without an accurate benchmark, teachers are evaluated on their overall standardized test pass rate, without consideration for the level of students in the class. This factor incentivizes teachers to find ways to remove troublesome or low-performing students from their courses rather than help them get better.

Another problem is training. Teachers in the state of Texas must complete 150 continuing education clock hours every five years. Clock hours are usually less than real-time hours, resulting in teachers spending a lot of time training. Statistical analysis can identify trends and patterns. Then administrators could focus all training on these trends and patterns.

Application Benefits

The application will have three main benefits. The first benefit is the application will provide quick and easy visual statistics for the user. The statistics will help the user to assess the learning environment of the school. Schools have very different learning environments due to the makeup of the school. The descriptive statistics will show average scores across ethnicity, socioeconomic status, gender, etc. The second benefit of the program is a predictive component. The software uses regression analysis to allow the user to forecast the scores of a particular student. The last benefit is the descriptive statistics are a better starting point for improvement. Instead of relying on very general categories like race and scores, the software provides more subcategories. These subcategories are a starting point for intervention. For example, if a school has many students whose parents have little education, the school can focus on explaining and demonstrating the value of college.

Application Description

The application is in Python and uses its data analytics libraries. Python is fast, easy, and scalable. Python has tons of data analysis libraries like NumPy, Pandas, and SciPy. The fact that Python has these libraries makes Python an easy choice for the application. It means that support for the language and libraries will continue for an extended period. The application's graphical user interface (GUI) will be made with a Python library called Tkinter. Upon running the application, the user will have an option to run a prediction model and afterward view the descriptive statistics of the school. The user will input the factors of the student into the predictive tool and click run to estimate the student's baseline performance on reading, writing, and standardized math exams.

Data Description

The example dataset is a CSV file composed of 1000 fictional students. Specifically, the columns are gender, race/ethnicity, parental level of education, type of lunch, test preparation course, standardized math score, standardized reading score, and standardized writing score. Gender has two categories: male and female. Race/ethnicity has the categories, Group A through Group E. Socioeconomic status is measured by the type of lunch: standard or free/reduced. The parental level of education categories is some high school, high school, some college, associate’s degree, bachelor’s degree, and master’s degree. The options for test preparation course are none and completed. The scores are scores from the standardized exams in their respective subjects.

Objective and Hypotheses

The application has two objectives. The first objective is the provide a snapshot of the school. Given that schools are dynamic and complex organizations, generally, teachers have little to no idea of the challenges that await them at the school. The descriptive statistics will help identify the makeup of the school and have a game plan for the students at the beginning of the year and make recognizing patterns and trends throughout the year easier. The second objective of the project is the regression section. The regression analysis can be used to estimate the baseline performance of a student. The intention is not to set definitive boundaries on the students' education but to prevent teachers and administration from having unrealistic expectations of student performance. The hypothesis for the regression analysis is that the individual categories affect performance. After determining which factors affect student performance, the school can focus its programs and interventions on helping students foster the beliefs of the most successful students.

Methodology

The methodology used for the project will be the waterfall methodology. The waterfall methodology operates in phases. Once one step is complete, the development process moves into the next stage. There is no backtracking in the waterfall methodology. The phases are requirements, design, implementation, verification, and maintenance.

There are two reasons for this choice. First, the project is small in scale, and the desired end product is known. Second, teachers have their strategies and curriculums set for the year. For example, if the school buys a curriculum for students struggling in math, the school district will not buy another mid-year. School districts have district assessments, which mirror the end-of-year standardized exam, but students do not take it seriously, and the data is unreliable. Teachers usually do their strategies for the entire year and gauge the plan's success with the standardized test results when students are trying their best. There are many negative repercussions if a student does not pass the standardized test, like taking extra courses, having to retake the test when your friends get off, and being made fun of by your peers.

The result of this structure is teachers and administrators needed complete software tools before the school year starts. Teachers and administrators can determine what adjustments are necessary for the software after receiving the standardized test scores. Any essential updates could be made during the summer. The waterfall methodology is perfect for this structure. All the requirements are known upfront, and the product is produced. The product is used for a year. If adjustments are needed, another smaller project will be created in the summer.

Funding Requirements

The funding requirements are assessed based on the different phases of the project. Of the 200 development hours, 40 of those hours are used for gathering the requirements and designing the product. The hourly rate of development hours is $100 per hour, so the total cost for gathering the requirements and creating the initial product is $4000. Once the implementation phase begins, our data engineers will start formatting the provided data. They covert the data to the correct data types and fill in missing or inaccurate data, so it will not affect the accuracy of the results. This is the start of the implementation part of the process and is billed at $100 per hour for 100 data engineering hours for a total of $10,000. The next part of the process is making the software tool. The product will take 160 hours at $100 per hour for a total of $16,000. After the product is verified and complete, it is implemented in the school. The 2-year maintenance and support plan is $ 10,000. The total is $40,000 for the entire project.

Stakeholders Impact

The impact of the project has both short-term and long-term implications. In the short term, if Kilgore accepts the offer, TeachTech will earn $40,000 of revenue. In the long term, TeachTech can sell its product to other schools and different schools in the Kilgore district. Most districts have 20-30 schools in their community, which could mean a lot of future revenue for TeachTech. If TeachTech were to win a contract for an entire school, the income could be substantial. If a school district has 20 schools, the revenue could be $800,000. Lastly, education as a whole is not very innovative. A lot of school districts and teachers use the same programs, training seminars, etc. If successful, the likelihood of other school districts adopted the product is high.

Data Precautions

The dataset is a fictional dataset from Kaggle.com, so there are no data precautions. Laws like HIPPA, FERPA, and PCI DISS do not apply to this fictional dataset. This dataset is also available for public use, so there are no licensing concerns. The only issue with this dataset is it may not paint a clear picture in identifying trends and patterns since it is fictional.

Developer’s Expertise

The developers are familiar with Python but have not used the data analytics side of Python. They are new to Jupyter notebooks, Voila, and Tkinter. The data analytics are done in the Jupyter Notebook. Voila produces a webpage with the results of the data analysis done in the Jupyter Notebook. Voila hides the code used to create the notebook. Voila is very simple to use since it only has to be installed and ran. Tkinter is used to produce the graphical user interface or GUI. The developers are intermediate programmers and used courses to learn Jupyter Notebooks, Volia, and Tkinter. The benefits of using the software and libraries are they are designed for a specific purpose and make creating a final product easier. The developers can produce a high-quality product with a slight learning curve.

B. PROJECT PROPOSAL

Problem Statement

Every school is different. A school's culture is made up of all the students who attend it and the school's location. The result is every school has unique and surprising challenges. Teachers are unaware of the challenges they may face when they start at a new school. Teacher turnover is high, with almost a third of teachers not making it to their 6th year. We believe this problem is two-fold. Teachers and administrators have unrealistic expectations about student performance, and teachers not identifying the potential challenges that await them. As an example, a school may have a high limited English proficiency population (LEP). If a teacher is aware of this, they can work to incorporate strategies in the lesson plans. Similarly, if a school has issues with attendance, a teacher can use counselors and truancy officers when a student’s attendance starts to slip instead of waiting until the student has missed a significant amount of school.

The proposed application solved both of these problems in two different ways. The first way is the application provides the descriptive statistics of the school. Instead of going in blind, teachers can get an insight into their students before the first day of school. Suppose their school has a high percentage of students with parents who only have a high school education. In that case, teachers will know that their students may not value education as much as a population with a higher percentage of parents with more schooling. Teachers will see that they will have to spend time showing/proving the value of education to get their students more involved. The descriptive statistics provide a screenshot of the environment beforehand so teachers can prepare for the challenges they face. The opposite approach is for a teacher to figure out what the problem is mid-year and then implement a solution that is too late. Administrators can also use descriptive statistics to decide which type of programs and interventions to acquire.

The second part of the project is regression analysis. The regression analysis provides a baseline for individual student performance. It is only an estimate calculated by statistics, but it is better than having no forecast. From experience, teachers develop unrealistically high expectations of student performance due to pressure put on by the administration and themselves. They lead to stress and frustration, and discontent with even great results. Using multi-linear regression, teachers can identify expected scores for students. When it is time for evaluation of the student's performance, the tool can be used as a baseline for student performance. This tool will help determine which teachers are improving student outcomes instead of teachers who happen to get students with higher results in the first place.

Customer Summary

The data analysis tool can be used for many purposes. The ideal customer would be teachers and administrators. Both groups can use the data insights to produce results in the workplace. Teachers can use the tool to gather insights about their students and plan and make curriculum adjustments. Administrators can affect change at a more of a high level. They can use the tool to select staff training and decide where to focus their energy. Superintendents could use the tool to analyze trends across their school and their district.

Existing System Analysis

To configure the existing, the IT department would need to use JupyterHub. JupyterHub can be configured on a web server. After JupyterHub is configured, the server will be able to multiple Jupyter notebook environments. Voila is what hides the code and makes Jupyter notebooks user-friendly to non-technical users. As a result, the only configuration necessary would be JupyterHub to a web server.

Data

The data is a fictional dataset from [www.kaggle.com](http://www.kaggle.com). The direct link is <https://www.kaggle.com/spscientist/student-performance-in-exams>. The information is a CSV with 1000 rows and eight columns. The data has been made without any errors, so there is nothing to convert on any missing blanks to fill.

Although the data set is fictional, it serves as an excellent example of what to expect in a real-life scenario. Some common factors which could affect school performance are gender, race/ethnicity, parental level of education, lunch, test preparation course. Most of the elements are self-explanatory, but some are not. Since this is a made-up dataset, race/ethnicity is separated into group A through group E. These are not codenames for particular ethnicities but groups. In a real-life project, these would be actual ethnicities. Lunch is used as a determinant of socioeconomic status. The two options for lunch are standard and free/reduced lunch. Test preparation course has two options, none or completed. None or completed refers to whether or not students took a test preparation course for their standardized exams.

Schools typically have programs that gather and format data. They usually lack data analysis and data visualization and the ability to connect the data to actionable items. TeachTech feels the acquisition of data will not be complex. Schools will also have options to select categories they determine are valuable. This dataset is an excellent example because it is similar to real-life, but schools will also have a degree of flexibility in deciding their categories.

Project Methodology

Given the nature of teaching, a place where strategies are set and evaluated at year-end, TeachTech believes the best project Methodology is the waterfall methodology. Schools need a completed software tool that remains unchanged throughout the year. After the year-end assessment through standardized tests, schools can request changes to the software tool. Here is the project methodology in further detail.

Requirements: TeachTech will meet with the client to determine the requirements. In schools, the primary decision-maker is either the superintendent of the school or the administration (principals and instructional staff). We believe teachers should be involved in deciding what statistics are vital to them, but realize that ultimately administrators are held accountable to the performance of the school and, like sports coaches, can be removed in a few years if performance doesn’t meet expectations. The requirements phase would determine which categories and data visualizations are essential to the school.

Design: In the design process, the preliminary design is made. In this phase, all the interfaces, categories, data visualizations, and logins are implemented. Once the design process is complete and the final design approved, the process is executed.

Implementation: Implementation starts with data. Schools usually have tons of data available to them. State and federal laws require some things; however, if there is any additional data that needs to be collected, TeachTech will have to figure out ways to collect the data. Once the data is collected, TeachTech will transform and cleanse the data. After that, descriptive statistics, prescriptive statistics, and dashboards are created. From there, the last piece is configuring JupyterHub to the web server and creating the logins.

Verification: After implementation, verification of the product begins. In this stage, we test the product. Individuals components are tested via unit testing, and the entire system is testing via integration testing. After all, tests are complete; the final user tries the product. In other words, acceptance testing is used to determine if the project meets the user's requirements.

Maintenance: The maintenance phase is where TeachTech offers ongoing software support and customer support. Minor changes can be made to software during the summer if necessary.

Product Outcomes

The outcome of the design phase is a diagram showing the categories and options to select. The result of the product phase is screenshots of the final working product. The screenshots will highlight the different kinds of data visualizations and a sample regression analysis.

Implementation Plan

After the product is designed, it will be implemented. The data engineers will gather and clean the data while the software engineers work writing the Python code in Jupyter notebooks for the statistical analysis and data visualizations. After the information is prepared, the data will be loaded into the software tool. Voila will be installed on the program to create the unalterable dashboards. The result of all of this action will be a complete product. The last step is for the school to configure its web server with JupyterHub. This will allow users to log in and produce multiple different Jupyter notebooks.

Evaluation Plan

The evaluation plan is based on the requirements. The general conditions are that the tool produces both descriptive statistics and predictive statistics. The device will create 1-3 items in each of the main categories: graphical/pictorial methods, measures of central tendency (mean, median, mode), and measures of dispersion (range, variance, standard deviation, skew). This is the industry standard. In the waterfall method, the verification of this will take place after implementation. The verification phase also includes acceptance in which the end-user will be able to determine if the product meets their needs and the requirements.

In the waterfall methodology, testing begins in the verification phase or after the product is built. The product is fixed, so no changes will be made after the product is created. The test phase will start with testing the data. The data will be tested to see if it works as expected, but also to know that it doesn't work as expected as well. Invalid inputs will be used to try this. The data will also be tested to make sure all data is accurate and valid. Afterward, the unit testing will begin. The unit testing verifies that the code is written correctly. Next is the integration testing; it checks to see if the system as a whole works together. Finally, there is acceptance testing to see if the finished product meets the requirements and is accepted by the user.

Resources and Costs

There are no costs associated with running the project in a production environment. Python, Voila, JupyterHub, and Jupyter Notebooks are all open source and free.

Programming Environment-Since schools are already equipped with web servers and computers; there is also no additional to run the project in the programming environment. As stated above, all software is open source and free as well.

Environmental Costs-There is minimal environmental costs. The schools will have to configure the webserver with JupyterHub. The price will be determined by how much the IT staff is paid and how long it takes for this configuration. The next environmental cost would be installing Python on all the teachers' computers. These environmental costs are negligible.

Human Resource Requirements-Part of the cost is determined by the development hours, and the rest is a flat fee for software support and maintenance. Here is an outline of the labor costs and corresponding timeframes:

|  |  |  |  |
| --- | --- | --- | --- |
| Hours | Activity | Cost | Timeframe |
| 40 hours | Requirements and Design | $4,000 | 1 week |
| 100 hours | Data Engineering | $10,000 | 2 weeks |
| 160 hours | Software Development | $16,0000 | 2 weeks |
| N/A | Software support &  Maintenance | $10,000 | 2 years |

Timeline and Milestones

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Event | Start Date | End Date | Duration  (Business Days) | Dependencies | Resources Assigned |
| 1. Requirements and Design | 06/01/21 | 06/09/21 | 7 | None | Stakeholders and developers |
| 2. Gathering, cleansing, and transforming data | 06/10/21 | 06/28/21 | 14 days | 1 | Data engineers |
| 3. Building the software tool | 06/29/21 | 07/19/21 | 14 days | 1, 2 | Software Developers |
| 4. Verification and acceptance testing | 07/19/21 | 07/28/21 | 7 days | 3 | Developers, Users, Stakeholders |
| 5. Ongoing sales support | 07/29/21 | 07/29/21 | 2 years from the start date of the product | 4 | Software support |

D. POST-IMPLEMENTATION REPORT

Project Purpose

This project was multi-faceted. We wanted to address multiple problems and therefore had multiple goals with this project. I feel we succeeded. The data product is impressive in terms of utility and providing structure for an otherwise unstructured environment. We had many goals with this project. The first goal was to provide descriptive statistics on the school's population. Although schools use descriptive statistics, they do not use them in much detail and are rendered useless.

From the start, the product provides a more robust understanding of the underlying population. First and foremost, more data was used, but since this is a data product, everything was created for the teachers and administrators, so they do not have to figure out how to gather and display the data. We used gender, race/ethnicity, parental level of education, lunch, test preparation course, math score, reading score, and writing score. Quartiles, mean, and standard deviation are presented for the teachers to get a feel for the overall student performance. K-Means clustering is used to provide a visual of the clusters of students' scores on all exams. It is also used to visualize the student's test scores on all exams versus their parental level of education. Histograms are used to provide a picture of the distribution of scores, and boxplots give the user a visual representation of the ranges and concentration of scores. Scatterplots are used to provide a visual representation of how each category affects the score. Overall the descriptive statistics paint a picture for the new teacher and gives them a starting point for improvement. For the experienced teacher, it gives them more profound insight into patterns and trends they possibly suspect but aren't sure of.

The second part of the project is benchmarking student scores. In teaching, teachers are not evaluated directly on their student's scores, but many decisions made by management are determined by how good a teacher's standardized test scores are. This leads to unwanted behaviors. Classes have a lot of randomness in them, so sometimes teachers get many strong students and vice versa. Since teachers feel pressure to have students pass, teachers have an incentive to finds ways to get disruptive low performing students out of their classes. For example, if you have a disruptive, low-performing student, you can write them up frequently in hopes of them sending them to an alternative school. Another unwanted behavior is teachers do not share information or instructional strategies with one another. An instructional strategy is a way to teach a topic effectively, as a game, or related the subject to something students are familiar with.

In teaching, the randomness of the class strength is unaccounted for. We are happy to say the data product helps in this area. The regression estimation tool helps predict the students' scores on all three exams. The Graphical User Interface (GUI) has dropdowns of all options for each category. The user simply has to select the properties of a student, and the tools predict their scores. This helps teachers and administrators assess which teachers are improving student performance and disincentivizes the behaviors mentioned above. Teachers can be measured on student improvement instead of the percentage of students who pass the standardized exam. If student improvement is the metric, there is more incentive to collaborate.

Lastly, the data product gives insight into where to invest training and intervention dollars. For example, suppose a school has a high percentage of students with parents with low levels of education. In that case, the school can invest money into training teachers on engaging students who do not value education as much. The school can also invest in programs to educate students on the value of education or focus on building education degree paths related to specific careers.

Datasets

The data is a fictional dataset from [www.kaggle.com](http://www.kaggle.com). The direct link is <https://www.kaggle.com/spscientist/student-performance-in-exams>. The information consists of 1,000 rows of students and eight columns describing the students. The maker of the dataset did not provide any inaccurate or missing data, so the data was not cleaned in any way. The information was checked to verify there were no missing data, see figure D1. The Pandas library in Python can read a CSV, format it, and import it into Jupyter Notebooks, see Figure D2. The Pandas library makes the data beautiful and easy to read. The only significant change to the data was the addition of codes for the qualitative categories: gender, race/ethnicity, parental level of education, lunch, and completion of test preparation course. To perform linear regression, categorical data has to be transformed into numerical data.

An example of this is the category code for male is 0 and female is 1. Linear regression is calculated using the 0's and 1's since it can't be computed from male and female (see figure D3. The same was done for the categorical data on some of the K-Means clustering plots.

Figure D1. Checking for missing data

Table

Description automatically generated

Figure D2. Importing the data from CSV to Jupyter Notebook



Table

Description automatically generated

Figure D3. Changing categorical data into numerical category codes for regression

Text

Description automatically generated

Table

Description automatically generated

Data Product Code

In this section, we will talk about the functionality of the code. The data product is mainly two parts. The first part is primarily short lines of code that produce many descriptive statistics about the student body. The second part of the code is the GUI and multi-linear regression analysis. The data product was created using many libraries, and therefore explaining the functionality of the code includes talking about how the libraries simplified our code. The libraries used were Pandas, NumPy, Matplotlib, Dabl, Plotly, Seaborn, and Sklearn.

For the first part of the project, the data was imported into Jupyter Notebooks. After displaying the data and showing some descriptive statistics, the data was formatted to computer and display the K-Means clusters. The first step was separating the data into two columns, for example, math score vs. reading score or reading score vs. writing score. After these separate DataFrames were made, the Sklearn library fits the data to the number of clusters desired. In our case, we chose three because we thought they represented the data well. Also, using the Sklearn library, the three centroids (red dots) were added to the graphs. Most of the other visualizations (scatterplots, histograms, boxplots) were created using one or two lines from a library.

For the multi-linear regression part of the data product, a lot more code was needed. The linear regression score estimator tool is the output of three linear regression models. We felt it would be tedious and unrealistic for teachers to use other scores as inputs to the regression tool. Therefore the teachers only have to enter the categorical data: gender, race/ethnicity, parental level of education, lunch, and test preparation. Then the tool will output the prediction for all three exams.

For this tool, we used X as the categorical data. Then we created three different Y’s: math score, reading score, and writing score. We used Sklearn to perform the linear regression and fit the data. Afterward, the intercepts and coefficients were outputted. From there, the GUI was made. The GUI is a screen with dropdown options for each of the five categories. Since the user inputs a string or word like 'male' or 'some high school,' this data is changed back into categorical data code like 0 or 1 and used to calculate the prediction of the scores. When the user hits submit, he will get the forecast for that particular student. The GUI only shows the prediction score, but after the GUI is closed, the Jupyter Notebook shows the intercept and coefficients for each of the test scores.

Hypothesis Verification

The initial hypothesis for the project has been verified. There are many noticeable trends from the data visualizations and even more when looking at the multi-linear regression coefficients. Females performed better in reading and writing, and males performed better in math. For race and ethnicity, Group A performed the worst and Group E the best. The coefficient for ethnicity was the largest for the math score. The higher the parental level of education, the higher the students performed. Students from lower socioeconomic status performed the worst. Those were the students with free or reduced lunch. Students who completed a test preparation course tended to perform better than those who did not.

Effective Visualizations and Reporting

Our approach to the data product was to provide a plethora of visualizations. We know that every teacher has a different knowledge base of statistics, and we wanted to offer a wide range of visualizations to aid users at every level.

A couple of visualizations were very effective even to the most basic of users. The K-Means clustering of each test score vs. parental level of education shows that the higher level of parental education, the higher the student tended to perform on all three scores. This pattern is easy to spot and occurs on all three exams. Also, in the data product, there are box plots that show the relationship that ethnicity, parental level of education, socioeconomic status, and test preparation affect student scores. Lastly, the linear regression tool helps users determine relationships among data. A user can play with the different dropdown options to see how it affects the students predicted score.

Accuracy Analysis

At first glance, the accuracy of the linear regression tool may seem low. The adjusted R-squared is .24, .22, and .32 for math, reading, and writing scores. As a general rule, people consider .75 substantial, .50 moderate, and .25 weak. However, because humans are unpredictable, “Any study that attempts to predict human behavior will tend to have R-squared values less than 50%.” (Frost 2018). Knowing that studies with human behavior generally have R-squared values less than 50% adjusted R-squared values in the .24 to .32 range more acceptable.

Furthermore, research showed a similar study by the Iowa Department of Education.

Similar parameters were used like free or reduced lunch and percent minority, etc. Multi-linear regression was used for these variables, and the student standardized test scores were predicted. The adjusted R-squared values ranged from .16 to .27. Given that adjusted R-squared values are usually lower than 50% for human subjects and the range of values is similar to a large-scale project, TeachTech believes the regression equation is a good fit. The regression tool should be a good estimator of performance.

Application Testing

While testing the descriptive statistics and visualizations, a lot of unit testing was completed. The output of the code was tested, and also Python generated errors were addressed. The K-Means code was checked with another high school dataset to verify validity.

The bulk of the testing was done on the regression estimator tool with GUI. There were so many unit tests conducted. The code that produces the regression variables was checked. The dropdown and submit buttons on the GUI were reviewed, and the code transformed the selections to the selection's category code. After each piece of the code was tested, integration tests were performed to make sure the application worked together as a whole. Afterward, system tests were conducted to verify that the GUI worked correctly with Voila and JupyterHub. Lastly, the data product was tested by the user as the final acceptance test.

Application Files

All files are located in a folder name Terrence\_Wideman\_Cap\_Project. The files contain:

* StudentPerformance.csv: The dataset used for the analysis.
* Project.ipynb: This is the Jupyter Notebook Python file that contains all the code for the descriptive statistics, visualizations, and the linear regression estimator tool with GUI
* Terrence Wideman Capstone.docx: The write-up of the project.

\*Directions on how to run the file in voila are down below in the titled User’s Guide.

User’s Guide

These instructions are for non-technical users. A technical user can run the linear regression estimator tool from the Jupyter Notebook separately. These instructions will bring up the linear regression tool first, and then when the app has closed, the dashboard with data visualizations will appear.

1. Download the Terrence\_Wideman\_Cap\_Project folder.
2. Open up a terminal on Mac or Command Prompt on Windows.
3. Change directories until you are inside the Terrence\_Wideman\_Cap\_Project folder.
4. Run the command: voila project.ipynb
5. **Important**: Your web browser will open and say executing 0 of 29. It will continue until it says 27 of 29. Then the Linear Regression Score Estimator will pop up in a different window.
6. Use the linear regression tool to predict the desired student or exit the tool, and then the descriptive statistics will pop up on the web browser.
7. The last score prediction will is printed out at the bottom of the Voila dashboard.

Summation of Learning Experience

The learning experience was enjoyable. I have some prior experience in statistics so understanding the statistics was not difficult. I also have taught high school for four years, so coming up with a worthwhile and exciting project was easy. I have also used Python in the past. These parts of the project came easy to me.

I had to seek a lot of assistance in the beginning to get started. I had no experience with Jupyter notebooks. After I learned Jupyter Notebooks, I had to ask the professor which library to use to make the GUI. I was perplexed about how Jupyter Notebooks and Tkinter worked together, so I had to research that. Ultimately I decided to use Voila for the dashboards and Tkinter for the GUI for the linear regression estimator tool. I also used the internet to find ways to perform and plot K-means clustering and linear regression.

I feel this project has contributed to the concept of life-long learning for me. I had to learn almost everything I used for this project. If I had not been open to learning, I wouldn't have produced this valuable tool. This project also gave me a sense of what it's like to be a software developer. In this project, I experienced all phases of the software development process.

E. Sources

Zelinski, A. (2019, May 31). *Teacher turnover remains high in Texas public schools*. Houston Chronicle. https://www.houstonchronicle.com/news/houston-texas/houston/article/Teacher-turnover-remains-high-in-Texas-public-13908607.php.

Frost, J. (2018, November 12). *How High Does R-squared Need to Be?* Statistics By Jim. https://statisticsbyjim.com/regression/how-high-r-squared/.

Pennington, J. (2007). (rep.). *DISTRICT CHARACTERISTICS: WHAT FACTORS IMPACT STUDENT ACHIEVEMENT?* (pp. 1–14). Des Moines, IA: Iowa Department of Education.