Prediction of Scores for Public Schools in California

Ahrim Han, Ph.D.

June 12 2019

Contents

1 Introduction

Problem Statment. The California Assessment of Student Performance and Progress (CAASPP) is the assessment system to measure how well students are mastering Californias academic standards in English language arts/literacy and mathematics. These results can be used to monitor the student progress and to give feedback the teachers need to change teaching methods more effectively. However, the test scores vary widely from school to school. There is a common belief that affects high test scores such as *schools with many Asian students* or *schools with high-income families*. There is a strong need to find more informed and granular causes that impact the test achievements of schools.

Ultimately, **we aim to predict and find the inferior groups of schools that indeed need help**. I suggest the expected beneficiaries with the provided results.

Expected Beneficiaries.

- To broadening educational opportunities, administrators of the school districts/state departments of education or other organizations can effectively identify the schools that need most supports. Budgets and human resources can be allocated in the order of the needs for tutoring, mentoring, extracurricular programs, educational consultants, and so on. In the equity aspect, schools should strive to create an environment where all students feel valued and all students are learning to high standards.

- Teachers can put much more effort into the under-performing groups to reduce the achievement gaps.

- From a parents perspective, these results can be an indicator to select a good school that meets the high academic standards.

Approach. We acquired the CAASPP test score data in 2018 from the California Department of Education [1, 2, 3]. For obtaining more meaningful analysis results and building a more accurate prediction models, we combined the information on median house prices [4, 5].

We basically aim to predict the percentage of the students into four achievement groups **Standard Exceeded (Level 4), Standard Met (Level 3), Standard Nearly Met (Level 2), Standard Not Met (Level 1)**.

In the data visualization and exploratory data analysis, we plotted the various kinds of graphs (including interactive stacked bars using *plotly* library) and have the insights on **exceeded scores** and **inferior scores** regarding to **gender, ethnicity, english-language fluency, economic status, disability status, and parent educations**. We also performed correlation analysis, univariate selection, and feature importance methods to find the strong indicators affecting lower scores.

In the modeling, we will use regression and classification algorithms to build predictive models. We tried various machine learning techniques to pick the one which performs best. The classification algorithm predicts if the schools **"need help" (1) or "do not need help" (0)**. **We set the "need help" schools that has more than "80% of the standard not met" students (8,786 schools)**.

Based on these results, we aim to identify the schools/districts/groups of students who can effectively raise the test scores. Also, we can suggest the helping strategies by referencing the features of the "exceeded standard" group model.

2 Data Wrangling

In this section, we perform data cleaning, fix missing values, and add new columns with meaning values.

2.1 Data Loading and Manipulating

2.1.1 CAASPP Test Scores

The test type is the Smarter Balanced Assessment Consortium (SBAC) of English Language Arts/Literacy and Mathematics. The data set is too large to commit to GitHub, we uploaded the data in our space. You can download the data here [6]. Besides the CAASPP test scores, **Test data file**, the information is separated into 3 files, **Entity table, Subgroup ID table, and Test Id table**, so that these files must be merged with the test data file to join the names with the appropriate score data.

- The public score data is available between 2015 and 2018 (4 years) [1, 2, 3]. We only used data in the year of 2018.
- For each year data, the test data file is provided in a 'csv file' format. For the record of 2018, for example, there are 3,269,730 rows with 32 columns
- This data contains the scores of two parts, English Language Arts (ELA) and Mathematics, for students in grades 3-8 and grade 11. The test data is comprised of state, counties, districts, and schools along with the test scores. The information on parent education, races, disabilities, gender, English-Language fluency can be combined with the test data.

The data was imported into DataFrame of Pandas. I decided to use only the next columns: 'Country Code', 'District Code', 'School Code', 'Test Year', 'Subgroup ID', 'Grade', 'Test Id', 'Students with Scores', and achievement levels. The minimum and maximum test scale score ranges are provided here [7]. The 'Mean Scale Score' is used to determine four achievement levels: **Percentage Standard Exceeded, Percentage Standard Met, Percentage Standard Nearly Met, Percentage Standard Not Met**. Many studies showed that discretization can lead to improved predictive accuracy and is more understandable [8].

The test score data also has area descriptors [9]. There are 4 areas of reading, writing, listening, and research/inquiry for ELA whereas 3 areas of concepts and procedures, problem solving/modeling and data analysis, and communicating reasoning for mathematics. For each area, the achievement levels are divided into Above Standard, Near Standard, and Below Standard depending on the scale scores compared to the Standard Met achievement level on the total content-area test.

The **Entity table** lists the County, District, and School entity names and codes for all entities as the existed in the administration year selected. This file must be merged with the test data file to join these entity names with the appropriate score data.

Here are detailed explanations and decisions made toward the data.

- To evaluate school performance, we use the four achievement levels instead of Mean Scale Score. The levels are about intervals of numbers which are more concise to represent and specify, easier to use and comprehend as they are closer to a knowledge-level representation than continuous values.

- The Grade represent 3-6 grades (elementary schools), 7-8 grades (middle schools), and 11 grade (high schools). The Grade 13 denotes all grades [**?**], so we decided to use data only 13 for minimum sample size. I believe the aggregated data at each school level is enough for representing the characteristics of public schools in California.

- The **Subgroup ID** lists the codes with the groups (e.g., gender, English-language fluency, economic status, ethnicity, (ethnicity for economically disadvantaged, ethnicity for not economically disadvantaged), disability status, parent education, migrant).

- The **Test Id** is 1-4; 1 represents ELA and 2 represents mathematics, respectively. We only consider Test Id 1 and 2. The Test Id 3 and 4 are excluded because they are CAA

(California Alternative Assessments) scores. The CAA scores are taken by students in grades 38 and grade 11 whose individualized education program (IEP) teams have determined that the student's cognitive disabilities prevent him or her from taking the online CAASPP Smarter Balanced assessments.

2.1.2 House Price

The Zillow Home Value Index (ZHVI) data [4] was imported and loaded. The ZHVI is a seasonally adjusted measure of the median estimated home value across a given region and housing type. The data was collected from April 1996 to November 2018 on monthly basis. I cleaned up the data by dropping house prices that are less than 2018. To analyze the school performance on a yearly basis, **the monthly prices were grouped by each year into a median value**.

I had tried two different versions when dealing with time data: data manipulation using 1) *DatetimeIndex* objects and 2) using user-defined functions. I found the first method is more convenient and safe for dealing with time related data. For example, even though I assured to use years as an *'int64'* type, it may cause unexpected spaces to be inserted. However, this can cause errors because when merging data tables requires keywords of the same data type. Therefore, we first manipulate the time data using *DatetimeIndex* objects, and then finally, we convert columns of 'Test Year' from *DatetimeIndex* to *'int64'* for compatibility.

2.2 Joining Multiple Datasets and Cleaning Data

There are multiple dataset and we need to merge efficiently to obtain useful and clean data. The diagram in Figure 1 shows multiple data and merging keys among them. To obtain the test scores of specific schools, districts, or counties, we first should get the exact school codes from entity tables. When finding the school codes, you should specify a county, a district, and a

school names because there may exist several schools with the same names. These are denoted as the CDS. Please note that if we specify only the school name(s), we could retrieve the several schools with the same names. It is important to include these three codes to avoid the double-counting in any summary calculations. Using the CDS codes, we then retrieve the DataFrame of the test data scores of a school. In the same way, we can retrieve the DataFrames of the county and the district. We append the specific names to the test score DataFrame by merging two tables (Test data + entities). We dropped the columns Type Id and Test Type since they are not have significant meanings as school performance indicators. At last, we merge the house prices and test score data.

2.3 Detecting and Imputing Missing Values

Detecting missing values. Many of the data of Subgroup ID are missing. In the test scores, the missing data is filled with some symbols (e.g., * and -). Thus, even DataFrame.info() function retrieves as all data are existed, we need to substitute these symbols as NaN (missing data).

A basic strategy to use incomplete datasets is to discard entire rows and/or columns containing missing values. However, this comes at the price of losing data which may be valuable (even though incomplete). So, we first **drop the 252,877 rows having NaN in all scores**.

Imputing for missing values. Before we put features into a model, missing values must be filled and all features must be encoded. Datasets such as blanks, NaNs or other placeholders are incompatible with *scikit-learn* estimators which assume that all values in an array are numerical, and that all have and hold meaning. A good strategy is to impute the missing values, i.e., to infer them from the known part of the data (https://scikit-learn.org/stable/modules/impute.html). To deal with the missing values, we use the basic strategies for imputing missing values. **Missing values are imputed using the statistics of the *mean* of each**

column in which the missing values are located.

Finally, we also drop the schools containing names of ”Program” and ”Alternative”. Then, we preliminarily finalize our data at this stage.

References and Notes

[1] California Assessment of Student Performance and Progress (CAASPP) Results from California Department of Education, https://caaspp.cde.ca.gov/.

[2] CAASPP Score Definition, https://caaspp.cde.ca.gov/sb2018/research_fixfileformat18.

[3] Research Files for Smarter Balanced Assessments, https://caaspp.cde.ca.gov/sb2018/ResearchFileList.

[4] Zillow research data, https://www.zillow.com/research/data/.

[5] Civil Rights Data Collection, https://ocrdata.ed.gov/.

[6] CAASPP Test Scores Download.

[7] Smarter Balanced Scale Score Ranges, https://caaspp.cde.ca.gov/sb2016/ScaleScoreRanges.

[8] "Discretization: An Enabling Technique", Liu, Huan, Farhad Hussain, Chew Lim Tan, and Manoranjan Dash, *Data mining and knowledge discovery 6, no. 4 (2002): 393-423*, https://cs.nju.edu.cn/zhouzh/zhouzh.files/course/dm/reading/reading03/liu_dmkd02.pdf.

[9] Understanding CAASPP Reports: Definitions, Reporting Calculation, Achievement Level Descriptors, https://caaspp.cde.ca.gov/sb2018/UnderstandingCAASPPReports.