Prediction of Scores for Public Schools in California

Ahrim Han, Ph.D.

June 15, 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]. In data wrangling (Section 2), we performed data cleaning, fixing missing values, and adding new columns. For obtaining more meaningful analysis results and building a more accurate prediction models, we combined the information on median house prices [4, 5]. Missing values are imputed using the statistics of the *mean* of each column in which the missing values are located. We basically divide 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) and focus to predict the top (Level 4) or bottom (Level 1) groups.

In the data visualization (Section 3) and exploratory data analysis (Section 4), we plotted the various kinds of graphs (including interactive stacked bars using Plotly library) and gained 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 (Section 5), we used the regression and classification algorithms to build predictive models. The regression algorithm predicts the percentage of students who do not meet the standard. 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 (312 out of 8,786 schools). We tried various machine learning techniques to pick the one which performs best. For regression, out of 5 different models, we obtained the best regression model using the random forest regressor with 10 folds cross validation with the accuracy of RMSE 10.77, MAE 7.69, and R^2 0.68. For classification, we tried to solve the class imbalanced problems using the Stratified K-fold cross validation and the weighted evaluation metrics to reflect the mass of the classes. In addition, we scaled the training data and significantly improved the accuracy of the K-Nearest Neighbor algorithm. As a result, we obtained the best classification model using the random forest classifier based on grid search cross validation with the accuracy 0.97 and AUC 0.98.

Based on these results, we identified the top and bottom schools and found the important features determining those schools. We recommended some strategies that effectively increase the achievements for scores in Section 6.

2 Data Wrangling

In this section, we perform data cleaning, fix missing values, and add new columns with meaning values. More details with codes on data wrangling can be found in this IPython notebook.

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.

3 Data Visualization

In this section, we explored the data to find trends, correlations, insights, and potential outliers based on visualization. These graphs and figures are important as a communication tool for collaborating in data science teams or presenting to business-oriented customers. For utilizing advanced features, we used the **seaborn and Plotly libraries** in addition to **Matplotlib library**. In this California score data, there are 52 counties, 784 districts, and 6,539 schools. Please note that the data is **grouped by County levels**. Before building a prediction model in school-level, it is worth to find rough trends in the bigger level such as the counties rather than the levels of districts or schools. More details with codes on data visualization can be found in this IPython notebook.

Hypothesis. We start with the following hypothesis.

- The schools with many Asian students tend to achieve high scores.
- The schools with high-income families tend to achieve high scores.
- The schools with highly educated parents tend to achieve high scores.
- The schools surrounded by high house costs tend to achieve high scores.

Research Questions. Therefore, we investigate the following three research questions.

1. How students are different in achievement levels for each category? (Section 3.1)
2. What features can you find in the top and bottom performance groups? (Section 3.2)
3. Are house prices correlated to the exceeded scores or the inferior scores? (Section 3.3)

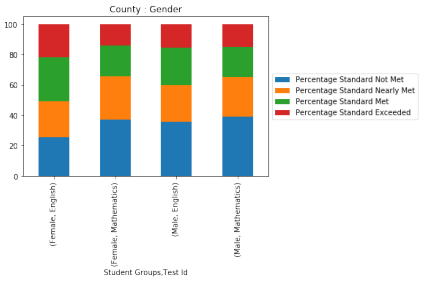
3.1 Comparison of Scores in Each Student Category Using Bar Plots

We provide two different version of bar plots for each category—*all four achievement levels in a stacked bar* and *specific achievement levels in a parallelized bar*.

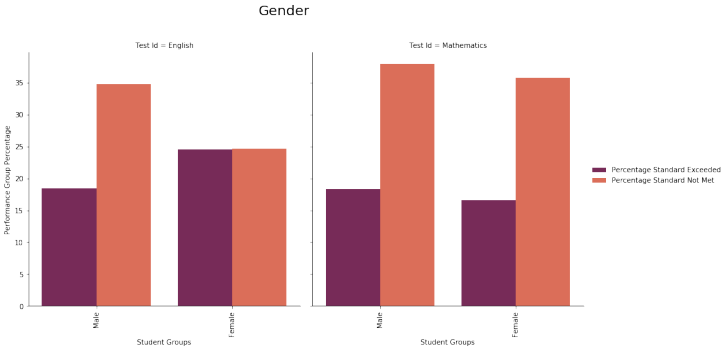
3.1.1 Gender

Figure 1 shows that female students exceed male students in English, while male students exceed female students in Mathematics.

- In the English subject at the ”Standard Exceeded” level, females students are 6.4% more than males students.
- In the mathematics subject at the ”Standard Exceeded” level, males students are 1.4% more than female students.
- At the ”Standard Met” above level (”Standard Exceeded” + ”Standard Met”), females students are 10.8% more than males students in English (females: 50.9% $>$ males: 40.1%). In mathematics, there are not much difference (males: 34.8% $>$ females: 34.6%).
- **The subject difference of female students is much bigger than male students.** At the ”Standard Met” above level , female students are 16.3% more in English than in mathematics. In contrast, the males students are 5.3% more in English than in mathematics.



(a) All achievement levels



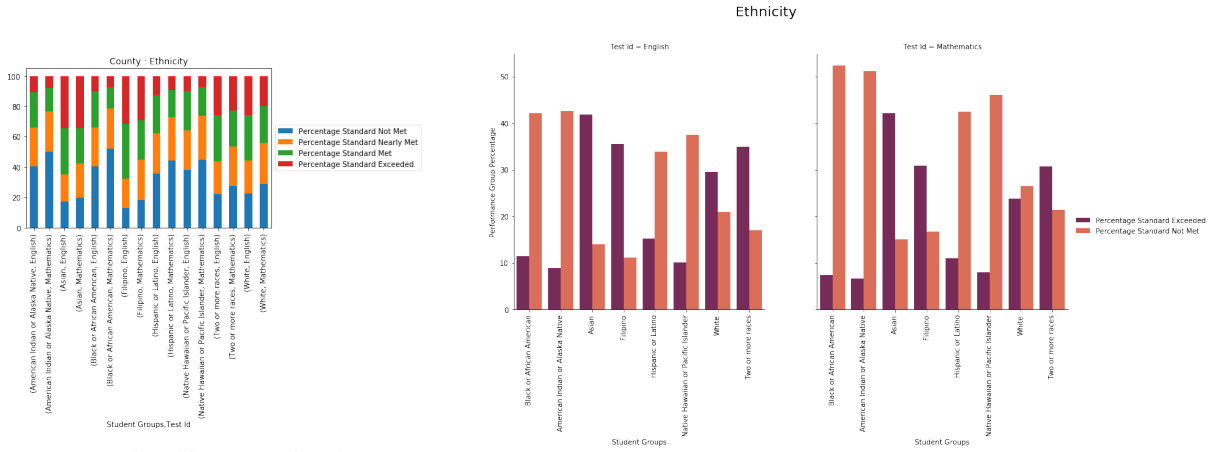
(b) Comparison of highest and lowest achievement levels in English and Mathematics

Figure 1: Bar plots for gender.

3.1.2 Ethnicity

Figure 2 shows that Asian students achieve the best performance, while Black or African American and American Indian or Alaska Native students achieve the lowest performance in both English and mathematics.

- **Students' achievements are higher** (there are the most "Standard Exceeded" students) in the order of Asian, Filipino, two or more races, and white, for both English and mathematics.
- **Students' achievements are lower** (there are the most "Standard Not Met" students) in the order of Black or African American, American Indian or Alaska Native, Native Hawaiian or Pacific Islander, and Hispanic or Latino in both English and mathematics.
- **The ethnic group of students in the "Standard Not Met" level has much more difficulties in mathematics** than English.



(a) All achievement levels

(b) Comparison of highest and lowest achievement levels in English and Mathematics

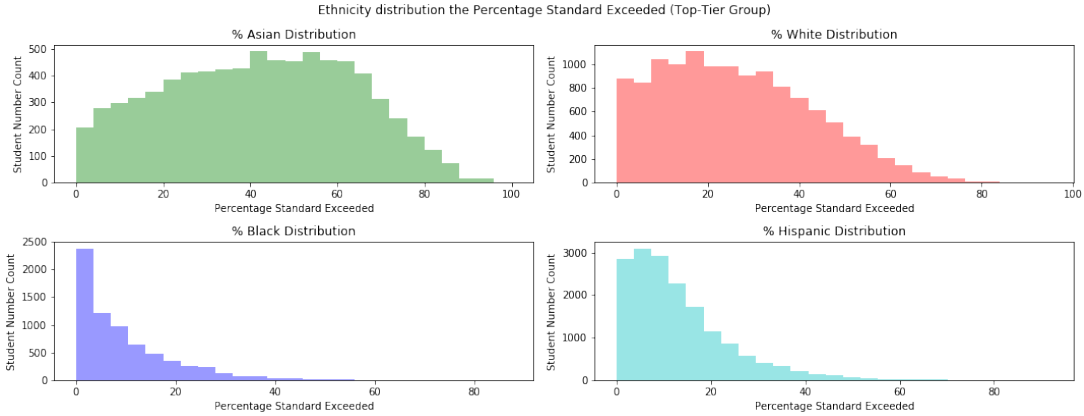
Figure 2: Bar plots for ethnicity.

Ethnicity Distribution for Top and Bottom Scores. We further analyze the distribution of four ethnicity groups (i.e,. Asian, Whites, Black, and Hispanic students) distributed in top scores ("Percentage

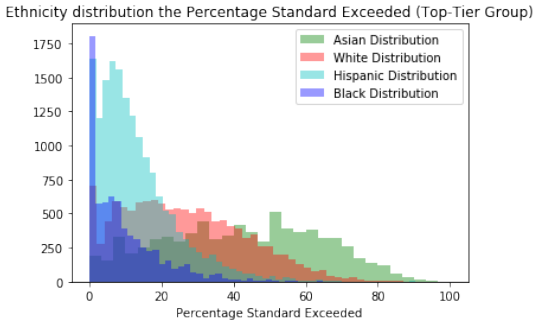
Standard Exceeded") and bottom scores ("Percentage Standard Not Met").

Figure 3 shows that Asian students are in the diverse range of the percentage of high scores. In short, many of the Asian students exceeded in some schools but a small portion of Asian students exceeded in other schools.

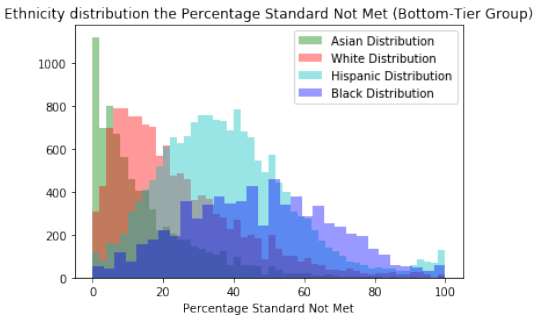
However, there are a few Black and Hispanic students who achieve the exceeded standard scores. As you can see, the graphs in the Black and Hispanic distribution, the graph bar is skewed to the left. This means that a small portion of Black and Hispanic students exceeded some other schools, but there is almost no counts that the majority or a high portion of those Black and Hispanic students achieve the high performances.



(a) All achievement levels



(b) Comparison of highest and lowest achievement levels in English and Mathematics



(c) Comparison of highest and lowest achievement levels in English and Mathematics

Figure 3: Ethnicity distribution in top and bottom scores.

3.1.3 English-Language Fluency

Figure 4 shows that Initial Fluent English Proficient (IFEP) students achieve the best performance in both English and mathematics.

- In California, students whose home language is not English are required by law to be assessed in English language proficiency. Thus, the IFEP students have enough language proficiency or are native language speakers, and their parents may have moved from other countries and are immigrants. **This is very interesting insights that IFEP students highly exceed English only students in both English and mathematics.** The percentage of standard exceeded students of IFEP are 38.2% (English) and 33.1% (mathematics), while those of English only are 20.9% (English) and 15.9% (mathematics). I could observe that this trend becomes more obvious in the districts where many Asian immigrants live. From this result, **I can insist that immigrants have high educational interests and efforts.**

3.1.4 Economic Status

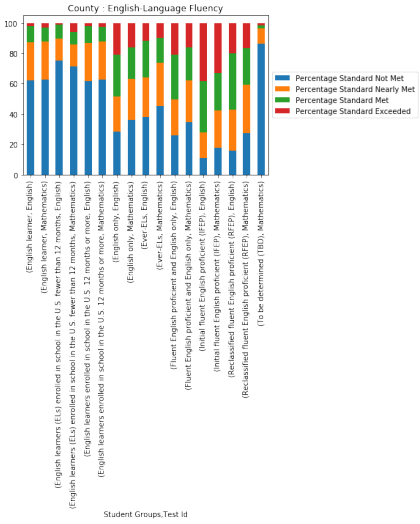
Figure 5 shows that the economically disadvantaged students have much more difficulties than not-economically disadvantaged students.

- **Almost half of the economically disadvantaged students are NOT standard met in mathematics.** For example, 45.4% of economically disadvantaged students are "Standard Not Met" in mathematics and 37.4% are "Standard Not Met" in English.

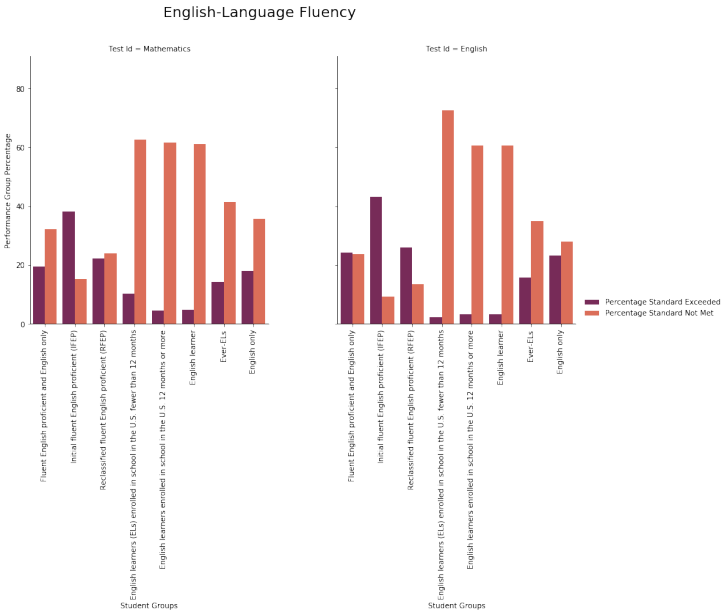
3.1.5 Disability Status

Figure 6 shows that only the small number of students with disabilities could achieve the best performance (English: 4.6%, mathematics: 4.5%).

- The majority of students with disabilities are in the "Standard Not Met" level (English: 66.7%, mathematics: 71.1%).

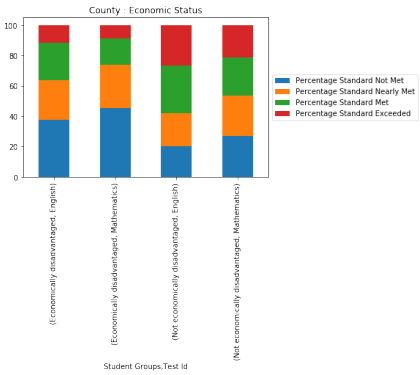


(a) All achievement levels

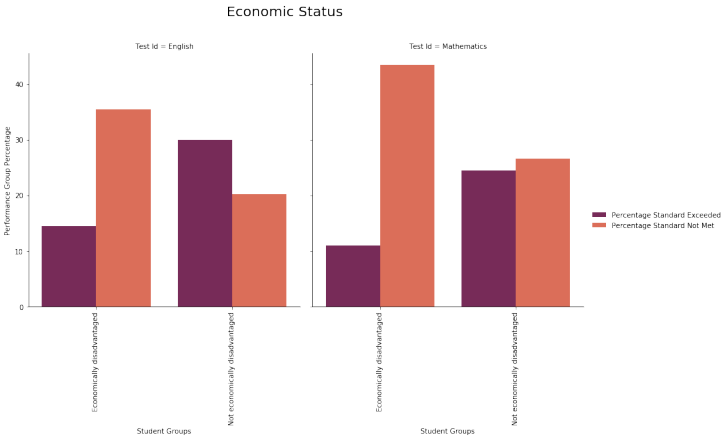


(b) Comparison of highest and lowest achievement levels in English and Mathematics

Figure 4: Bar plots for English-Language fluency.



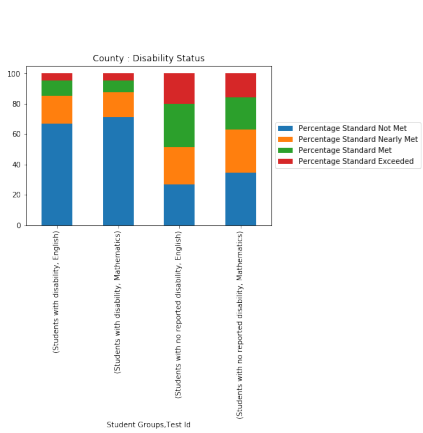
(a) All achievement levels



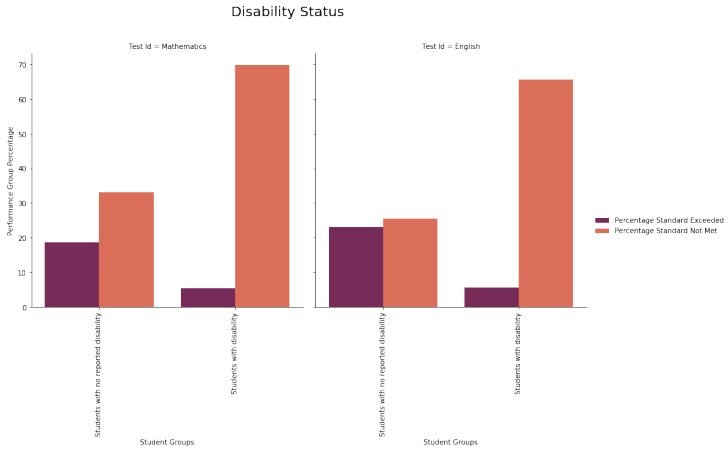
(b) Comparison of highest and lowest achievement levels in English and Mathematics

Figure 5: Bar plots for economic status.

- As in other disadvantaged or minor groups, the students with disability have more difficulties in mathematics.



(a) All achievement levels



(b) Comparison of highest and lowest achievement levels in English and Mathematics

Figure 6: Bar plots for disability status.

3.1.6 Parent Education

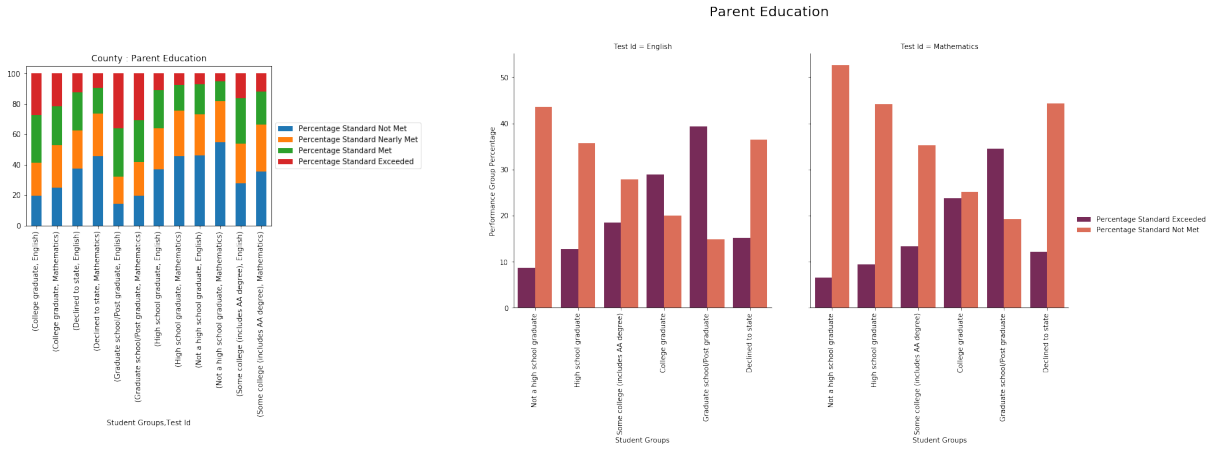
Figure 7 shows that the higher the level of parental education, the higher the achievement of students.

- The graphs apparently show that **students' achievements are higher in the order of the parents' education of "graduate school/post graduate", "college graduate", "some college (includes AA degree)" , "high school graduate", and "not a high school graduate".**

3.2 Comparison in Best or Worst Performance Groups Using Bar Plots

We analyzed the best and worst 10% performing counties (10% out of 58 = 5 counties). The counties can be summarized as follows.

- Top 5 County Names in English: ['Santa Clara', 'Marin', 'Placer', 'San Mateo', 'Orange']
- Top 5 County Names in Mathematics: ['Santa Clara', 'Marin', 'San Mateo', 'Orange', 'Placer']
- Bottom 5 County Names in English: ['Lake', 'Kings', 'Colusa', 'Humboldt', 'Monterey']



(a) All achievement levels

(b) Comparison of highest and lowest achievement levels in English and Mathematics

Figure 7: Bar plots for parent education.

- Bottom 5 County Names in Mathematics: ['Lake', 'Kings', 'Merced', 'Mendocino', 'Monterey']

To have a rough insight, we have drawn the graphs of the percentage of each achievement level for counties ("Performance Group") in Figure 8. Unfortunately, many counties have the most highest percentages in "Percentage Standard Not Met". In addition, the differences are much bigger in two both and worst achievement levels (i.e., "Percentage Standard Exceeded" and "Percentage Standard Not Met") than others. Therefore, it is worth to deeply investigate those best and worst groups to find the features that can effectively help to make better performing schools.

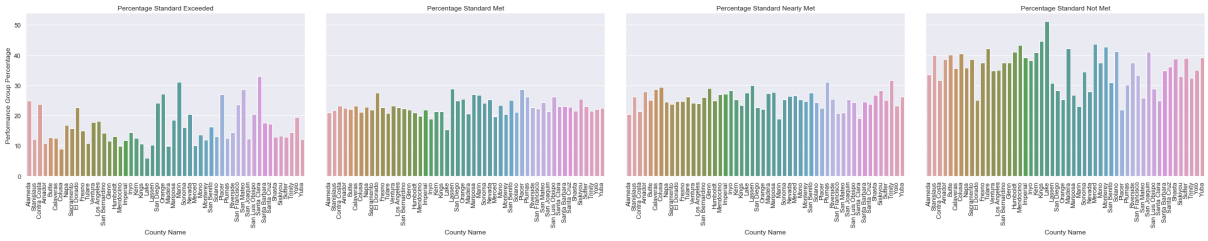
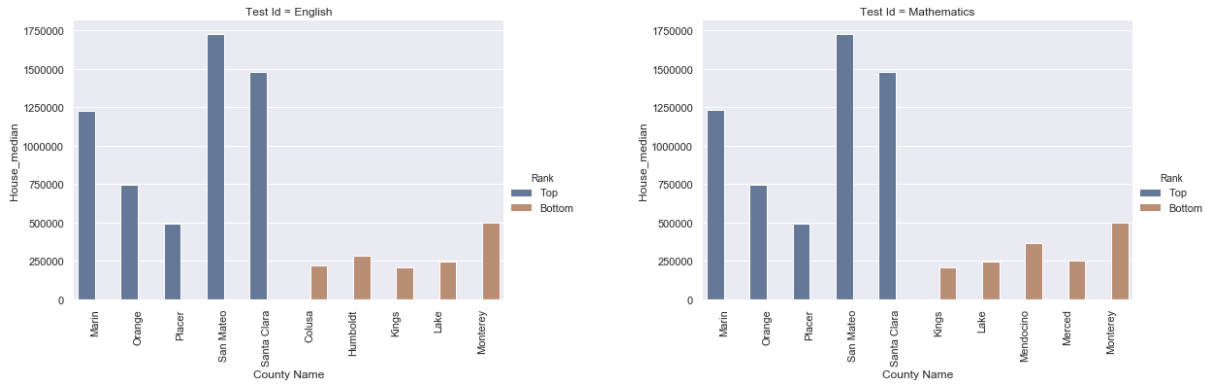


Figure 8: Percentage of each achievement level for counties ("Performance Group").

3.2.1 House Prices

Figure 9 shows that the best performance counties have higher house median prices. In contrast, the worst performance counties have lower house median prices. Thus, **test performance is closely related to the economic capabilities of the family to which the student belongs.**



(a) English (b) Mathematics

Figure 9: House Prices in Best and Worst 10% Performing Counties.

3.2.2 Student Categories

Figure 10 shows that the number of students in each student categories for best or worst performance groups. Here we summarize the results.

- We found that in the best performing counties, the percentage of white students is much higher than the percentage of white students in the whole county.
- Hispanic and Latino students are far more likely to be in the worst performing group than the best performing group. Likewise, Black and American Indian students are more involved in the group with the worst results. In contrast, Asian and white students are more likely to be in the best performing group than the worst performing group.
- The English learners have more difficulties in studying both English and Mathematics than the fluent English speakers.

- When students' parents graduate from graduate schools/post graduates or colleges, students are much more likely to be in the best performing group. For those students, the best performing groups are much larger than the worst performing groups. In contrast, students are more likely to be in the worst performing group when their parents are high school graduates or have lower education.

3.3 Correlations Using Scatterplots

3.3.1 Test Scores vs. House Prices

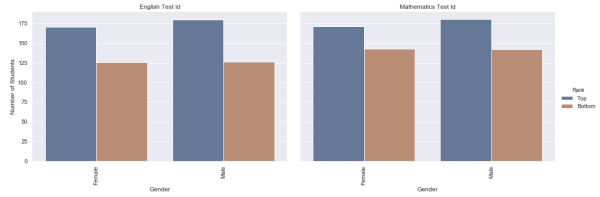
As in Figure 11a, we observe the **strong positive correlations** between the "Percentage of Standard Exceeded" and the house prices. In contrast, as in Figure 11b, we see the **strong negative correlations** between the "Percentage of Standard Not Met" and the house prices. In conclusion, students who live in areas with high housing prices have higher test scores.

4 Exploratory Data Analysis

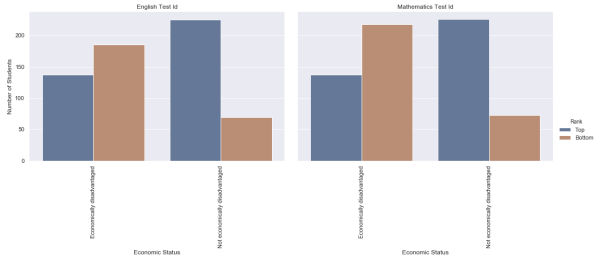
In this section, we use the inferential statistics to identify significant features in the data set. More details with codes on exploratory data analysis can be found in this IPython notebook.

Before performing exploratory data analysis, we need to preprocess the data. **Each school has 47 scores for each student category group**. Each student category group is summarized in Table 1. For predicting school scores, we need to focus on the school-level instances. Therefore, **we transform data for each school**. By using the `'pivot_table'` method in Pandas, **we need to pivot the scores based on a school as an index**. Therefore, 47 scores of each student group is added as features for each school instance.

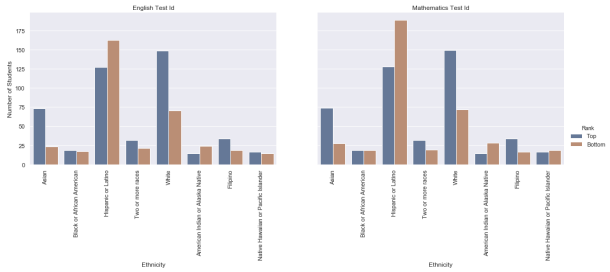
We need to derive new variables (e.g., number to percentage of Asian students) and merge variables (e.g., minor groups of ethnicity such "Native Hawaiian or Pacific Islander" and "American Indian or Alaska Native"). More detailed explanation for variables is in Section 4.4.



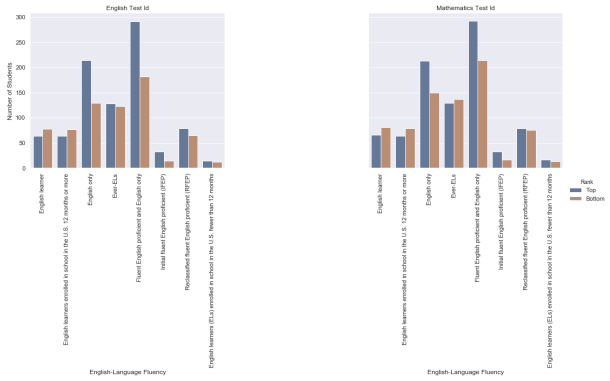
(a) Gender



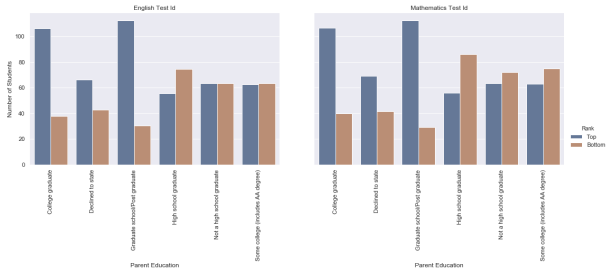
(b) Economic Status



(c) Ethnicity



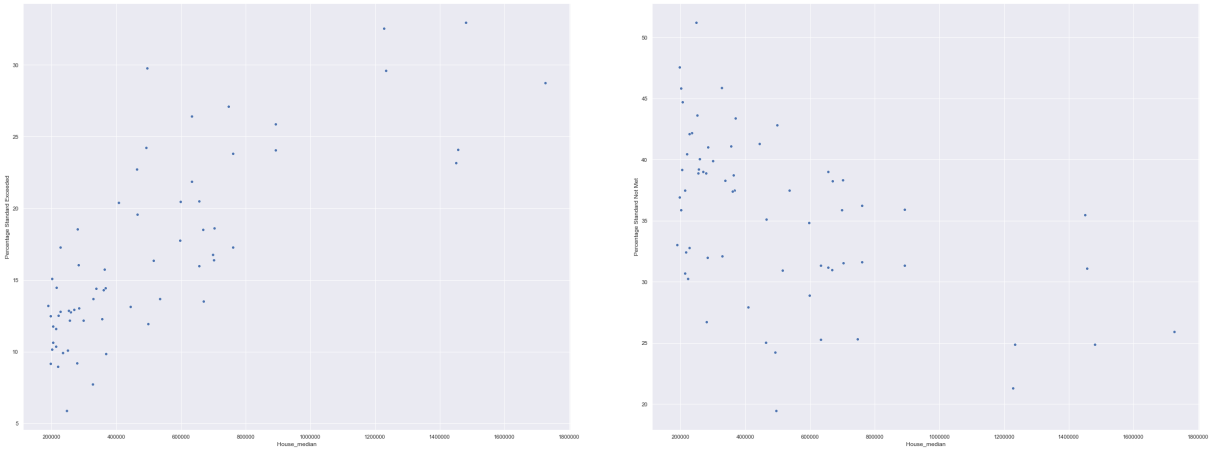
(d) English-Language Fluency



(e) Parent Education

Figure 10: Number of students in each student categories for best or worst performance groups.

A significant number of features could be redundant and irrelevant, therefore it is important to apply feature selection/dimension reduction. We performed the statistical hypothesis testing, correla-



(a) Percentage of Standard Exceeded vs. House Prices (b) Percentage of Standard Not Met vs. House Prices

Figure 11: Correlation graphs between test scores and house prices.

tion test, and feature selection for getting rid of the student group information for generating less number of features. In other words, we aim to generate and use the features that strongly affect for predicting the school scores.

4.1 Statistical Hypothesis Testing

4.1.1 T-Test for means of two independent samples.

We test whether the means of two independent samples are significantly different. If there is no difference (p-value is greater or equal than α = 0.05), then we want to **eliminate or merge** that student group information due to too much generating features.

- H_0: There is **no difference** in students' scores between sample1 and sample2.

- H_1: There **exist difference** in students' scores between sample1 and sample2.

T-Test for All Pairs. We performed a hypothesis test (two-sample test) for all pairs of student groups by assuming two group of samples are independent.

Figure 12 shows the results that the student group features sorted on the order of the occurrences. The following shows the results with the counter number.

Num	Category	Student Groups
1	All Students	All Students
2	Gender	Male
3		Female
4	Ethnicity	American Indian or Alaska Native
5		Asian
6		Black or African American
7		Filipino
8		Hispanic or Latino
9		Native Hawaiian or Pacific Islander
10		Two or more races
11		White
12	English-Language Fluency	English learner (EL)
13		ELs enrolled in school in the U.S. fewer than 12 months
14		ELs enrolled in school in the U.S. 12 months or more
15		English only
16		Ever-ELs
17		Fluent English proficient and English only
18		Initial fluent English proficient (IFEP)
19		Reclassified fluent English proficient (RFEP)
20		To be determined (TBD)
21	Parent Education	College graduate
22		Declined to state
23		Graduate school/Post graduate
24		High school graduate
25		Not a high school graduate
26		Some college (includes AA degree)
27	Economic Status	Economically disadvantaged
28		Not economically disadvantaged
29	Disability Status	Students with disability
30		Students with no reported disability
31	Migrant	Migrant education
32	Ethnicity for Economically Disadvantaged	American Indian or Alaska Native
33		Asian
34		Black or African American
35		Filipino
36		Hispanic or Latino
37		Native Hawaiian or Pacific Islander
38		Two or more races
39		White
40	Ethnicity for Economically Not Disadvantaged	American Indian or Alaska Native
41		Asian
42		Black or African American
43		Filipino
44		Hispanic or Latino
45		Native Hawaiian or Pacific Islander
46		Two or more races
47		White

Table 1: 47 student category groups.

```
[(('Ethnicity for Not Economically Disadvantaged',
   'Native Hawaiian or Pacific Islander'),
  14),
 (('English-Language Fluency', 'To be determined (TBD)'), 10),
 (('Ethnicity for Not Economically Disadvantaged',
   'American Indian or Alaska Native'),
  10),
 (('English-Language Fluency',
   'English learners (ELs) enrolled in school in the U.S. fewer than 12 months'),
  6),
 (('Ethnicity for Economically Disadvantaged',
   'Native Hawaiian or Pacific Islander'),
  5),
 (('Ethnicity', 'American Indian or Alaska Native'), 5),
 (('Ethnicity for Economically Disadvantaged',
   'American Indian or Alaska Native'),
  4),
 (('English-Language Fluency', 'English only'), 4),
 (('Ethnicity for Not Economically Disadvantaged', 'Hispanic or Latino'), 4),
 (('Ethnicity', 'Native Hawaiian or Pacific Islander'), 4),
 (('Migrant', 'Migrant education'), 4),
 (('Disability Status', 'Students with disability'), 3),
 (('Disability Status', 'Students with no reported disability'), 3),
 (('English-Language Fluency', 'English learner'), 3),
 (('English-Language Fluency',
   'English learners enrolled in school in the U.S. 12 months or more'),
  3),
 (('Ethnicity', 'Black or African American'), 3),
 (('Ethnicity for Economically Disadvantaged', 'Hispanic or Latino'), 3),
 (('Gender', 'Female'), 3),
 (('Ethnicity for Economically Disadvantaged', 'Black or African American'),
  3),
 (('Ethnicity for Not Economically Disadvantaged',
   'Black or African American'),
  3),
 (('Parent Education', 'Some college (includes AA degree)'), 3),
 (('Ethnicity', 'White'), 2),
 (('English-Language Fluency', 'Ever-ELs'), 2),
 (('English-Language Fluency',
   'Reclassified fluent English proficient (RFEP)'),
  2),
 (('Ethnicity for Economically Disadvantaged', 'Filipino'), 2),
 (('Ethnicity for Economically Disadvantaged', 'Two or more races'), 2),
 (('Parent Education', 'Declined to state'), 2),
```

Figure 12: T-test for the means of two samples for on target value 'Percentage Standard Exceeded'.

T-Test Between English and Mathematics Subjects. The score differences exist in most of the groups. In these test (Figure 13), when the $p-$value is much smaller than $\alpha = 0.05$, and we reject the null hypothesis that there is no difference. In fact, the $p-$values zero indicates that there is significant differences

between two samples in scores. However, we eliminate the subjects (Test Id) for further analysis or constructing prediction models, because the subject difference is not our major concerns.

```
Among 47 groups, the below listed groups do not show the score differences between subjects of English and Mathematic
s.
(Ethnicity : Asian), t-test: -0.6442098461, p-value: 0.5194586434
(Ethnicity for Economically Disadvantaged : American Indian or Alaska Native), t-test: -0.2165423008, p-value: 0.8288
461695
(Ethnicity for Economically Disadvantaged : Asian), t-test: -1.6896659140, p-value: 0.0911595770
(Ethnicity for Economically Disadvantaged : Native Hawaiian or Pacific Islander), t-test: 1.9116670138, p-value: 0.05
71893681
(Ethnicity for Not Economically Disadvantaged : American Indian or Alaska Native), t-test: 0.6731867988, p-value: 0.5
041967295
(Ethnicity for Not Economically Disadvantaged : Native Hawaiian or Pacific Islander), t-test: -0.0863412353, p-value:
0.9316259214
(Gender : Male), t-test: 0.9157011439, p-value: 0.3598364743
```

Figure 13: T-test between subjects (English and Mathematics).

Stepwise way of Feature Deletion. We can choose to drop all the rows of ('Category', 'Student Group') that do not actually affect a target variable, 'Percentage Standard Exceeded' or 'Percentage Standard Not Met' using the **stepwise way** for removing the student group information.

We analyzed all pairs of two samples using T-test and found the two samples that have no difference (p-value is greater or equal than α = 0.05). Then, we select and delete the most occurrence feature in the T-test results. We then reiterate the T-test process for find and delete next least affecting feature. The following results shows the deleted features (i.e., student group information) for every step with the number of occurrences.

Decisions for Variables. Based on the T-test, we can eliminate or merge the weak affecting student group indicators. By referring top indicators in no difference features, we adjust the following indicators for variables that will be used to make a machine-learning based school score prediction model.

1. Delete the meaningless indicators such as, 'To be determined (TBD)' and 'Declined to state'.

2. Delete the 'Disability Status', 'Economic Status', 'Ethnicity for Economically Disadvantaged', 'Ethnicity for Not Economically Disadvantaged'. It seems redundant and rather trivial that do not produce the new results.

```
[(('Ethnicity for Not Economically Disadvantaged', 'Native Hawaiian or Pacific Islander'), 14)]
[(('English-Language Fluency', 'To be determined (TBD)'), 10)]
[(('Ethnicity for Not Economically Disadvantaged', 'American Indian or Alaska Native'), 9)]
[(('Ethnicity for Economically Disadvantaged', 'Native Hawaiian or Pacific Islander'), 4)]
[(('English-Language Fluency', 'English learners (ELs) enrolled in school in the U.S. fewer than 12 months'), 4)]
[(('Ethnicity for Economically Disadvantaged', 'American Indian or Alaska Native'), 3)]
[(('English-Language Fluency', 'English only'), 3)]
[(('Ethnicity for Not Economically Disadvantaged', 'Hispanic or Latino'), 2)]
[(('Ethnicity', 'White'), 2)]
[(('Ethnicity', 'American Indian or Alaska Native'), 2)]
[(('Ethnicity', 'Black or African American'), 2)]
[(('English-Language Fluency', 'English learner'), 1)]
[(('English-Language Fluency', 'Reclassified fluent English proficient (RFEP)'), 1)]
[(('Ethnicity', 'Filipino'), 1)]
[(('Ethnicity', 'Native Hawaiian or Pacific Islander'), 1)]
[(('Ethnicity for Economically Disadvantaged', 'Black or African American'), 1)]
[(('Ethnicity for Economically Disadvantaged', 'Two or more races'), 1)]
[(('Ethnicity for Not Economically Disadvantaged', 'Black or African American'), 1)]
```

Figure 14: Stepwise way of feature deletion: For each iteration, we remove the feature that most occurring in the T-test results of no differences.

3. For 'Ethnicity', delete 'Two or more races', merge "Native Hawaiian or Pacific Islander" and "American Indian or Alaska Native", and create Minor races (i.e., Pct_Avg_Multi_Ethnicity_Minor, Pct_Multi_Ethnicity_Minor_English, and Pct_Multi_Ethnicity_Minor_Mathematics).

4. For 'English-Language Fluency', we organize the indicators:

   - Delete 'English learners (ELs) enrolled in school in the U.S. fewer than 12 months' and 'English learners enrolled in school in the U.S. 12 months or more' and use the 'English learner' only instead
   - Delete 'Ever-ELs' which indicates 'Reclassified fluent English proficient (RFEP)' + 'English learner'

5. For 'Parent Education', delete 'Some college (includes AA degree)'

4.2 Correlation Tests

Correlation states how the features are related to each other or the target variable. Correlation can be positive (increase in one value of feature increases the value of the target variable) or negative (increase in one value of feature decreases the value of the target variable).

4.2.1 Matrix with Heatmap

Heatmap makes it easy to identify which features are most related to the target variable, we will plot heatmap of correlated features using the seaborn library.

Correlation Table with Number/Percentage Related Features. Figure 15 and Figure 16 show that, just as assumed, the high score (‘Target_Avg_Percentage Standard Exceeded‘) is correlated to the higher house price (‘House_median‘), the higher education (‘Num_Avg_Parent Education_Graduate school/Post graduate‘), and good economic status (‘Num_Avg_Economic Status_Not economically disadvantaged‘).

It is interesting that the number of Hispanics (‘Num_Avg_Ethinicity_Hispanic or Latino‘) is highly correlated (**0.94**) with the number of economically disadvantaged students. The percent of Hispanics (‘Pct_Avg_Ethinicity_Hispanic or Latino‘) is correlated (**0.78**) but not as strong as the number feature. In California, there is the largest number of Hispanic students among other ethnicity students (see Figure 17), and this can be the cause of the high correlation.

4.2.2 Pearson’s Correlation Coefficient

Pearson’s correlation coefficient tests whether two samples have a linear relationship.

Assumptions:

- Observations in each sample are independent and identically distributed. - Observations in each sample are normally distributed. - Observations in each sample have the same variance.

Interpretation:

- H_0: The two samples are independent.

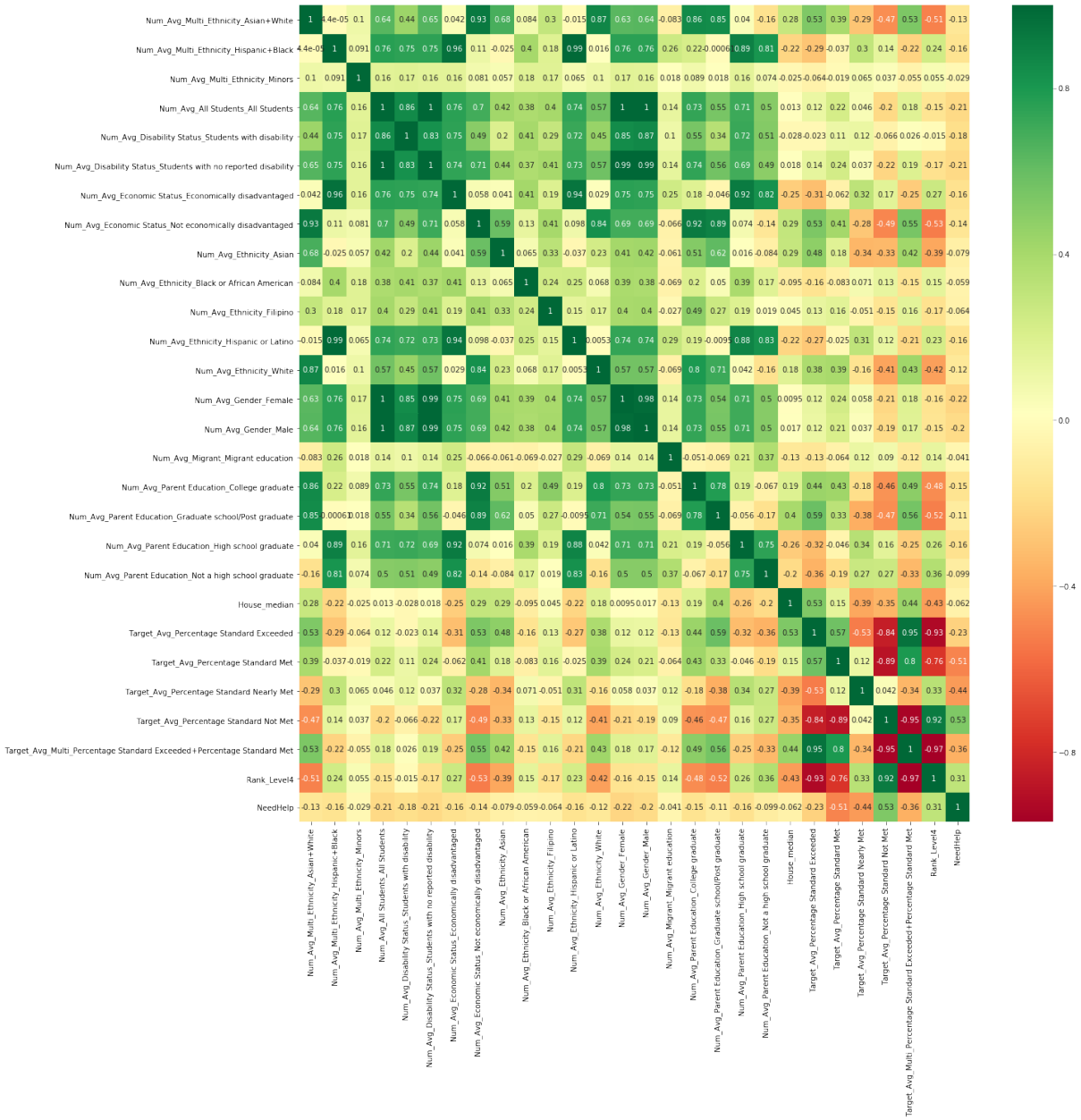


Figure 15: Correlation Table with Number Related Features.

- H_1: There is a dependency between the samples.

For example, we calculate the Pearson's Correlation Using SciPy, 'scipy.stats.pearsonr(x, y)'. For example, we investigated the relationship between 'Pct_Ethnicity_Asian_Mathematics' and 'Target_Percentage

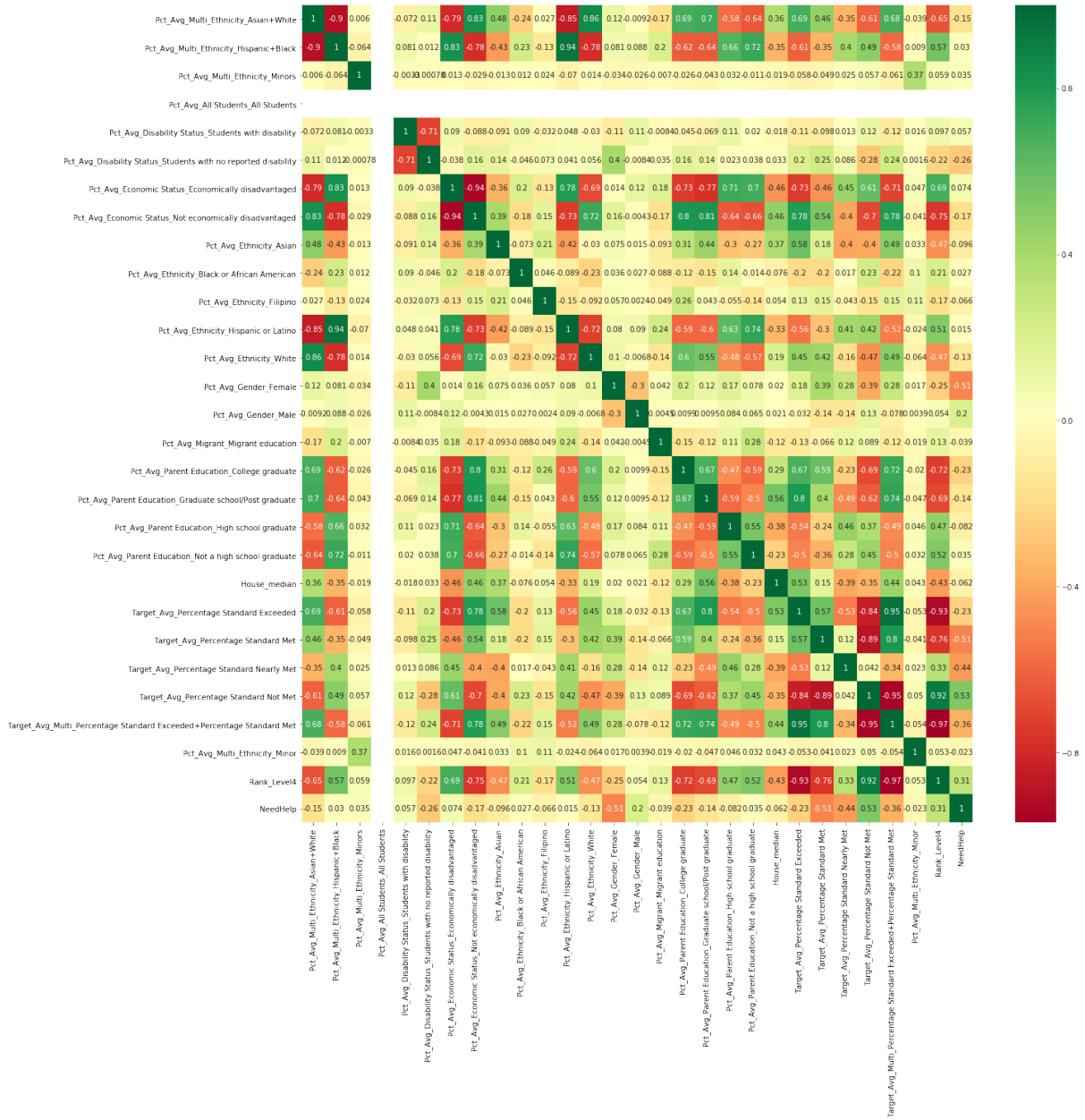


Figure 16: Correlation Table with Percentage Related Features.

Standard Exceeded_Mathematics' and obtained the results as below.

`* Spearman Rank Correlation between 'Pct\_Ethnicity\_Asian\_Mathematics' and 'Target\_Percentage Standard Exceeded\_Mathematics':`

	Category	Student Groups	School Name
14	Ethnicity	American Indian or Alaska Native	276
15	Ethnicity	Asian	7667
16	Ethnicity	Black or African American	6945
17	Ethnicity	Filipino	3428
18	Ethnicity	Hispanic or Latino	16782
19	Ethnicity	Native Hawaiian or Pacific Islander	412
20	Ethnicity	Two or more races	6595
21	Ethnicity	White	12612

Figure 17: Number of collected school data for each ethnicity.

```
corr: 0.6111925793, p-value: 0.0000000000
```

We reject the null hypothesis H_0. The portion of Asian students and the higher scores in Mathematics is **not independent but strongly correlated**.

4.2.3 Spearman's Rank Correlation

Spearman's correlation measures the strength and direction of monotonic association between two variables. Spearmans rank correlation is the Pearsons correlation coefficient of the ranked version of the variables. We can define a function for calculating the spearman's rank correlation.

Assumptions:

Observations in each sample are independent and identically distributed. Observations in each sample can be ranked.

Interpretation:

- H_0: The two samples are independent.

- H_1: There is a dependency between the samples.

For example, we calculate Spearmans Rank Correlation Using SciPy, 'scipy.stats.spearmanr(x, y)'.

For example, we investigated the relationship between 'House_median' and 'Target_Avg_Percentage Standard Exceeded'.

```
* Spearman Rank Correlation between 'House_median'
and 'Target_Avg_Percentage Standard Exceeded':
corr: 0.4723283465, p-value: 0.0000000000
```

We reject the null hypothesis H_0. The house prices and high scores is **not independent but correlated**.

4.3 Feature Selection

4.3.1 Univariate Selection

Statistical tests can be used to select those features that have the strongest relationship with the output variable. The scikit-learn library provides the SelectKBest class that can be used with a suite of different statistical tests to select a specific number of features. We use the chi-squared (chi^2) statistical test for non-negative features to select 20 best features.

The best features in the larger order of scores are as follows:

1. 'House_median'
2. 'Rank_Level4'
3. 'Num_Avg_Economic Status_Not economically disadvantaged'
4. 'Num_Avg_Multi_Ethnicity_Asian+White'
5. 'Num_Avg_Parent Education_Graduate school/Post graduate'
6. 'Num_Avg_Ethnicity_Asian'
7. 'Num_Avg_Economic Status_Economically disadvantaged'
8. 'Num_Avg_Multi_Ethnicity_Hispanic+Black'
9. 'Num_Avg_Ethnicity_White'
10. 'Num_Avg_Ethnicity_Hispanic or Latino'
11. 'Num_Avg_Parent Education_College graduate'

12. 'Num_Avg_Parent Education_Not a high school graduate'
13. 'Num_Avg_All Students_All Students'
14. 'Num_Avg_Disability Status_Students with no reported disability'
15. 'Pct_Avg_Economic Status_Not economically disadvantaged'
16. 'Pct_Avg_Parent Education_Graduate school/Post graduate'
17. 'Num_Avg_Parent Education_High school graduate'
18. 'Pct_Avg_Multi_Ethnicity_Asian+White'
19. 'Pct_Avg_Ethnicity_Asian'
20. 'Target_Avg_Percentage Standard Not Met'

As expected, for the higher achievement (Percentage of Standard Exceeded), higher house prices, higher economic status, Asians and Whites in Ethnicity, and higher education. 'Rank_Level4' is derived from the Percentage of Standard Exceeded, so it must be strongly correlated.

4.3.2 Feature Importance

We obtain the feature importance of each feature of the dataset by using the feature importance property of the model. Feature importance gives you a score for each feature of the data, the higher the score more important or relevant is the feature towards the output variable. Feature importance is an inbuilt class that comes with Tree Based Classifiers, we will be using Extra Tree Classifier for extracting the top 20 features for the dataset.

As can be seen in Figure 18, the 'Rank_Level4' and the other score variables are identified as the important features.

4.4 Variables for Modeling

Newly Derived Variables. For independent variable, we add new variables by combining Asian and Whites as well as Hispanic and Black students in the Ethnicity, so we expect this addition to tell new insights. In addition, we merge variables for "Native Hawaiian or Pacific Islander" and "American In-

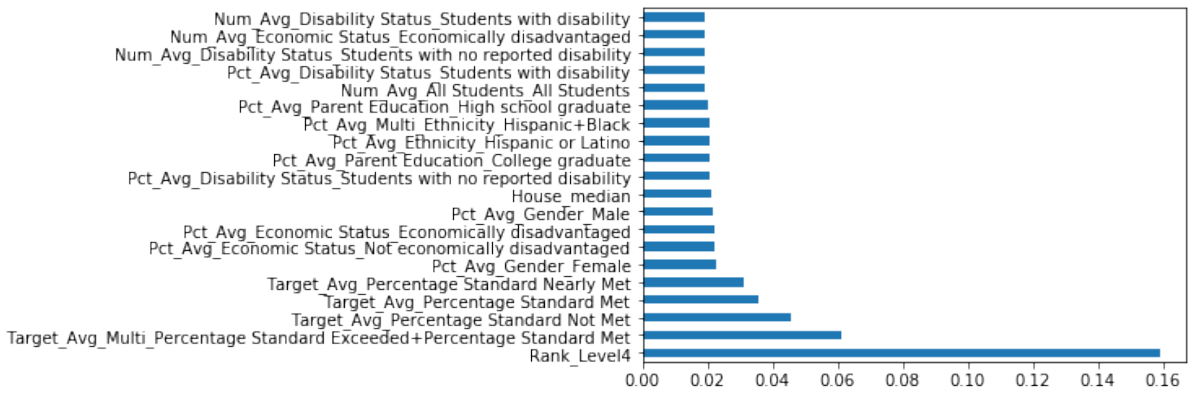


Figure 18: Feature importance using Extra Tree Classifier for extracting the top 20 features.

dian or Alaska Native" for the minor groups of ethnicity, so we expect this merging can reduce the dimensionality.

For target variables, we created the label "NeedHelp". The school is encoded to 1 when the 'Percentage Standard Not Met' $>$ 80%, otherwise 0. The 312 schools have been identified to this inferior group, while only 8 schools have been found when the 'Percentage Standard Exceeded' $>$ 80%. By analyzing the inferior group, many of those schools have zero percent of 'Percentage of Standard Exceeded' students.

4.4.1 Independent variables

We summarize the independent variables as follows.

Organized variables:

- ['Num'] x ['Category' + 'Student Groups' + 'Test Id']:
  - 'Num': 'Students with Scores' (Number of students)
  - 'Test Id' = [English, Mathematics]
  - 'Category' and 'Student Groups': (47 student category groups in Table 1.)
- House_median: House median prices in the school zones

New variables:

- [‘Pct’] x [‘Category’ + ‘Student Groups’ + ‘Test Id’]:

    - ‘Pct’: Percentage of students over all students in a school

- ([‘Num‘] | [‘Pct’]) x [‘Avg’ + ‘Category’ + ‘Student Groups’]:

    - ’Avg’ means the average number of percentage of students for English and Mathematics,

    - ‘Avg’ = (English + Mathematics) / 2

- [’Pct’] x ([’Multi’ + ’Test Id’] — [’Avg’ + ’Multi’]):

    - ’Multi’:

        - ’Asian+White’ or ’Hispanic+Black’ in ’Ethnicity’

        - ‘Minor’ indicates ‘American Indian or Alaska Native’ and ‘Native Hawaiian or Pacific Islander’

4.4.2 Dependent Variable

Then, we set a target variable, e.g., ’Percentage Standard Exceeded’ or ’Percentage Standard Not Met’, to be investigated. We are interested in the group of students whose performance achievements are exceeded or too inferior. By knowing the characteristics affecting those groups, we can make a score prediction and suggest recommendations later. We summarize the target variables as follows.

Continuous:

- Average percentage (Target_Avg) for all four achievement levels:

    - ‘Percentage Standard Exceeded’: Exceeded (Level 4)

    - ‘Percentage Standard Met’: Standard (Level 3)

    - ‘Percentage Standard Nearly Met’: Nearly (Level 2)

    - ‘Percentage Standard Not Met’: NotMet (Level 1)

- [’Target_Avg_Multi_Percentage Standard Exceeded+Percentage Standard Met’]:

    - Sum of two levels (Level 4 + Level 3) that can represent the portions that achieve the standards in a school.

Ordinal:

- 'Rank_Level4': ranked in a descending order for scores of 'Percentage Standard Exceeded' (Level 4). In short, the 1st indicates the top school.

- 'Rank_Level1': ranked in a descending order for scores of 'Percentage Standard Not Met' (Level 1). In short, the 1st indicates the inferior school.

Categorical:

- 'Need Help' [1] ('Percentage Standard Not Met' $>$ 80%) / 'No Need Help' [0] (others) labels (for Classification)

5 Modeling

The aim is to predict the inferior scores (i.e., percentage of the standard "NOT" met) of schools. We built machine learning models using regression and classification algorithms. Based on these prediction models, we can 1) identify the schools that need help and 2) obtain important features affecting the lower scores of schools. More details with codes on machine learning modeling can be found in this IPython notebook.

5.1 Regression

Regression analysis is a subfield of supervised machine learning. It aims to model the relationship between a certain number of features and a continuous target variable. In the regression, we use the 'Target_Avg_Percentage Standard Not Met' variable as a target variable.

5.1.1 Cross Validation: Train/Test Split, Leave One Out (LOO), K-Fold CV

We need to split the data into training and testing sets, fitted a regression model to the training data, made predictions based on this data and tested the predictions on the test data using the cross validation.

However, the train/test split technique takes to one extreme, K may be set to 1 such that a single train/test split is created to evaluate the model. Thus, the **train/test split technique is not stable in that it may not split the data randomly and the data can be selected only from specific groups.** This will

result in overfitting.

The Leave One Out Cross Validation (LOOCV) takes to another extreme, K may be set to the total number of observations in the dataset such that each observation is given a chance to be the held out of the dataset. This is called leave-one-out cross-validation, or LOOCV for short. However, **LOO requires quite a large computation time.**

Therefore, we the cross validations: K-Fold. This K-Fold cross validation is enough and appropriate for our model prediction.

5.1.2 Evaluation Metrics: MAE, RMSE, and R^2

- Mean Absolute Error (MAE): MAE is the mean of the absolute value of the errors.
- Root Mean Squared Error (RMSE): RMSE is the square root of the mean of the squared errors
- R^2: R^2 is the number that indicates the proportion of the variance in the dependent variable that is predictable from the independent variables. Basically, R^2 represents how accurate our model is. R^2 shows how well terms (data points) fit a curve or line. Adjusted R^2 also indicates how well terms fit a curve or line, but adjusts for the number of terms in a model.

5.1.3 Algorithms: Linear Regression, Random Forest Regressor, Gradient Boosting Regressor

Linear Regression: Linear regression attempts to model the relationship between two variables by fitting a linear equation to observed data. One variable is considered to be an explanatory variable, and the other is considered to be a dependent variable.

When the outcome we are trying to predict depends on more than one variable, we can make the multiple linear regression model which is more complicated model that takes this higher dimensionality into account. As long as they are relevant to the problem faced, using more predictor variables can help us to get a better prediction.

- **Train/Test Split Cross Validation for Linear Regression:** For a simple example, we split data

into 70% train and 30% test data. Out of 8,768 instances with 40 features, 6,137 is train data and 2,631 is test data. We fit the model and present the coefficients of the regression model.

Figure 19 represents the sorted the coefficients in a descending order of absolute values. The major affecting features for predicting the percentage of the standard "NOT" met schools are the number of Black or Hispanic students.

The results of the train and test split for Linear Regression model are as follows: RMSE: 11.2853, MAE: 8.2113, and R^2 score: 0.6614.

- **Leave One Out Cross Validation (LOOCV) for Linear Regression:** The results of the *leave one out* cross validated (number of splits: 8,768) Linear Regression model are as follows: RMSE: 11.3417 and MAE: 8.2913.

- **10-Fold Cross Validation for Linear Regression:** After fitting a model, we plotted the actual values (X-axis) and predicted values (Y-axis) (Figure 20). The results of the 10 fold cross validated Linear Regression model are as follows: RMSE: 11.7262, MAE: 8.5554, and R^2 score: 0.6233.

Random Forest Regressor: A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement if bootstrap = True (default). The results of the 10 fold cross validated Random Forest Regressor model are as follows: RMSE: 10.7661, MAE: 7.6911, and R^2 score: 0.6763.

Gradient Boosting Regressor: GB builds an additive model in a forward stage-wise fashion; it allows for the optimization of arbitrary differentiable loss functions. In each stage a regression tree is fit on the negative gradient of the given loss function. The results of the 10 fold cross validated Gradient Boosting for Regression model are as follows: RMSE: 11.4108, MAE: 8.3881, and R^2 score: 0.6368.

	Coefficient
Num_Avg_Ethnicity_Black or African American	1.18
Num_Avg_Ethnicity_Hispanic or Latino	1.14
Num_Avg_Multi_Ethnicity_Hispanic+Black	-1.13
Num_Avg_Ethnicity_Asian	0.98
Num_Avg_Ethnicity_White	0.98
Num_Avg_Multi_Ethnicity_Asian+White	-0.95
Pct_Avg_Parent Education_College graduate	-0.55
Pct_Avg_Migrant_Migrant education	-0.46
Pct_Avg_Ethnicity_Asian	-0.42
Pct_Avg_Gender_Female	-0.32
Pct_Avg_Ethnicity_White	-0.27
Pct_Avg_Gender_Male	0.26
Num_Avg_Disability Status_Students with no reported disability	-0.24
Pct_Avg_Ethnicity_Black or African American	0.21
Pct_Avg_Economic Status_Not economically disadvantaged	-0.18
Pct_Avg_Parent Education_Graduate school/Post graduate	-0.17
Num_Avg_Disability Status_Students with disability	-0.16
Num_Avg_Economic Status_Economically disadvantaged	0.15
Num_Avg_Economic Status_Not economically disadvantaged	0.15
Pct_Avg_Multi_Ethnicity_Asian+White	0.15
Pct_Avg_Multi_Ethnicity_Hispanic+Black	-0.13
Num_Avg_Migrant_Migrant education	0.13
Pct_Avg_Parent Education_High school graduate	-0.13
Pct_Avg_Disability Status_Students with no reported disability	-0.11
Pct_Avg_Multi_Ethnicity_Minors	0.10
Pct_Avg_Parent Education_Not a high school graduate	0.10
Num_Avg_Gender_Female	0.07
Num_Avg_Multi_Ethnicity_Minors	0.06
Pct_Avg_Multi_Ethnicity_Minor	0.06

Figure 19: Coefficients of the regression model with train (70%) and test (30%) split data.

5.1.4 Results of Regression

The results of the accuracy for regression models is summarized in Table 2. The **Random Forest Regressor worked best** with Root Mean Squared Error (RMSE) 10.7672, Mean Absolute Error (MAE)

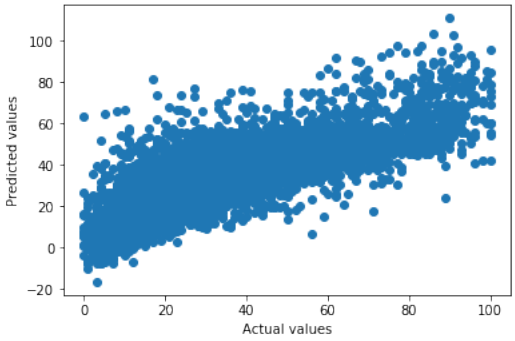


Figure 20: Plotted actual and predicted values using the Linear Regression with 10-fold Cross Validation.

7.6985, and R^2 0.6761.

Model Name	RMSE	MAE	R^2
Linear Regression with 1 folds Train and test split	11.2853	8.2113	0.6614
Linear Regression with 8,768 folds Leave One Out (LOO)	11.3417	8.2913	0.0000
Linear Regression with 10 folds CV	11.7262	8.5554	0.6233
Random Forest Regressor with 10 folds CV	10.7661	7.6911	0.6763
Gradient Boosting for Regression with 10 folds CV	11.4108	8.3881	0.6368

Table 2: Results of the accuracy for regression models.

5.2 Classification

New Binary Target Variable ('NeedHelp'): We use the 'NeedHelp' variable as a target variable. The variables used for modeling are explained in Section 4.4. Given the brief explanation, the 'NeedHelp' indicates that if a school needs a help (1) or not (0). We have labeled schools with more than 80% of students who do not meet the standard as needing help (1) otherwise (0).

Resolving Imbalanced Classes: We observed that the 'NeedHelp' has imbalanced classes: 3.69% of our dataset belong to the target class 'NeedHelp'. To overcome the problem of the **imbalanced classes**, we need to deal with this imbalanced classes properly: 1) Stratified K-Folds Cross Validation and 2) weighted evaluation metrics to reflect the mass of the classes.

Data Splitting into Train Data and Test Data: We basically split data into train data and test data int the ratio of 70% and 30%. For parameter tuning, we use the cross validation in the train data and build

the machine learning model, then validate the model with the remained test data. This more detailed explanation is given in Section 5.2.3 in Figure 24.

Scaling: For the K-Nearest Neighbor algorithm, we scale the independent variables (X_train and X_test) into the range such that the range is now between 0 and 1. If the distribution is not Gaussian or the standard deviation is very small, the min-max scaler works better than standard scaler.

5.2.1 Cross Validation: Stratified K-Folds Cross Validation

We used the 'Stratified K-Folds Cross Validation' [10]. This cross-validation object is a variation of KFold that returns stratified folds. The folds are made by preserving the percentage of samples for each class. In short, the stratification will ensure that the percentages of each class in your entire data will be the same (or very close to) within each individual fold.

5.2.2 Evaluation Metrics: Accuracy, AUC, Precision, Recall, F1

We use the weighted option when calculating the precision, recall, and f1 scores to reflect the mass of the classes. It calculates metrics for each label and finds their average weighted by support (the number of true instances for each label). This alters 'macro' to account for label imbalance; it can result in an F-score that is not between precision and recall. We also present the Receiver Operating Characteristic (ROC) curve and Area Under the Curve (AUC).

5.2.3 Algorithms: Logistic Regression, Decision Tree, GridSearchCV for Parameter Tuning for Decision Tree, Random Forest Classifier, and k-Nearest Neighbors Classifier

Logistic Regression: Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

- **Train/Test Split Cross Validation for Logistic Regression:** Figure 21 shows the results of the Logistic Regression with train (70%) and test (30%) split data.

```
**Results**
Model: Logistic Regression, Cross Validation: Train and test split, Number splits: 1
Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      2545
           1       0.44      0.26      0.32        86

   micro avg       0.97      0.97      0.97      2631
   macro avg       0.71      0.62      0.65      2631
weighted avg       0.96      0.97      0.96      2631


Accurcy:  0.9650
roc_auc_score:  0.6224
**Weighted average scores**
Precision : 0.9577
Recall    : 0.9650
F-score   : 0.9605
```

Figure 21: Logistic Regression with train and test split data.

- **Stratified 5-Folds Cross Validation for Logistic Regression:** Figure 22 represents the results of the Logistic Regression with Stratified 5-Folds CV. The results model evaluation are as follows: accuracy: 0.9656, roc_auc_score: 0.9656, weighted avg precision: 0.9646, weighted avg recall: 0.9656, and weighted avg f1-score: 0.9597.

Decision Tree: The decision tree classifier iteratively divides the working area (plot) into subpart by identifying lines. There are three key terms related to decision tree classifiers:

Criterion

- Impurity: Impurity is when we have a traces of one class division into other.
- Entropy: Entropy is a degree of randomness of elements. In other words, it is a measure of impurity. It is the negative summation of probability times the log of the probability of item x.
- Information gain: Information Gain (n) = Entropy(x)ŁŁ([weighted average] * entropy(children for feature))

  At every stage, a decision tree selects the one that gives the best information gain. An information gain of 0 means the feature does not divide the working set at all.

Stratified K-Folds Cross Validation: Stratified K-Folds CV (K = 5)

K = 1

	precision	recall	f1-score	support
0	0.98	0.99	0.98	1692
1	0.53	0.41	0.46	63
micro avg	0.97	0.97	0.97	1755
macro avg	0.75	0.70	0.72	1755
weighted avg	0.96	0.97	0.96	1755

K = 2

	precision	recall	f1-score	support
0	0.98	0.97	0.97	1691
1	0.33	0.37	0.35	63
micro avg	0.95	0.95	0.95	1754
macro avg	0.65	0.67	0.66	1754
weighted avg	0.95	0.95	0.95	1754

K = 3

	precision	recall	f1-score	support
0	0.97	1.00	0.99	1691
1	1.00	0.21	0.35	62
micro avg	0.97	0.97	0.97	1753
macro avg	0.99	0.60	0.67	1753
weighted avg	0.97	0.97	0.96	1753

K = 4

	precision	recall	f1-score	support
0	0.97	1.00	0.98	1691
1	1.00	0.15	0.25	62
micro avg	0.97	0.97	0.97	1753
macro avg	0.98	0.57	0.62	1753
weighted avg	0.97	0.97	0.96	1753

K = 5

	precision	recall	f1-score	support
0	0.97	1.00	0.98	1691
1	0.76	0.21	0.33	62
micro avg	0.97	0.97	0.97	1753
macro avg	0.87	0.60	0.66	1753
weighted avg	0.96	0.97	0.96	1753

(a) Each Fold in Stratified 5-Fold CV.

Results
Model: Logistic Regression, Cross Validation: Stratified K-Folds CV (K = 5), Number splits: 5
Accurcy: 0.9656
Precision: 0.7248
Recall: 0.2685
f1-score: 0.3479
roc_auc_score: 0.6299
Weighted average scores
Weighted Avg Precision: 0.9646
Weighted Avg Recall: 0.9656
Weighted Avg f1-score: 0.9597

(b) Mean results of 5-fold Cross Validated Logistic Regression.

Figure 22: Results of the Logistic Regression with Stratified 5-Folds CV.

Optimizing Decision Tree Performance

- criterion : optional (default=gini) or Choose attribute selection measure: This parameter allows us to use the different-different attribute selection measure. Supported criteria are gini for the Gini

index and entropy for the information gain.

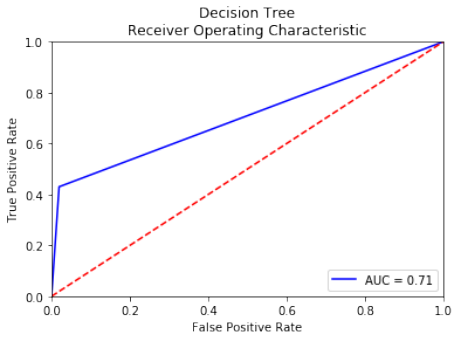
- splitter : string, optional (default=best) or Split Strategy: This parameter allows us to choose the split strategy. Supported strategies are best to choose the best split and random to choose the best random split.

- max_depth : int or None, optional (default=None) or Maximum Depth of a Tree: The maximum depth of the tree. If None, then nodes are expanded until all the leaves contain less than min_samples_split samples. The higher value of maximum depth causes overfitting, and a lower value causes underfitting (Source).

- **Stratified 5-Folds Cross Validation for Decision Tree:** The Figure Figure 23 represents the results of the Decision Tree with Stratified 5-Folds CV.The results model evaluation are as follows: accuracy: 0.9596, roc_auc_score: 0.7320, weighted avg precision: 0.9660, weighted avg recall: 0.9596, and weighted avg f1-score: 0.9614.

```
**Results**
Model: Decision Tree, Cross Validation: Stratified K-Folds CV (K = 5), Number splits: 5
Accuracy: 0.9596
Precision: 0.5614
Recall: 0.4868
f1-score: 0.4876
roc_auc_score: 0.7320
**Weighted average scores**
Weighted Avg Precision : 0.9660
Weighted Avg Recall : 0.9596
Weighted Avg f1-score : 0.9614
```

(a) Result



(b) AUC graph

Figure 23: Results of the Decision Tree with Stratified 5-Folds CV.

GridSearchCV for Parameter Tuning: The grid search cross validation for parameter tuning process is as follows (see Figure 24). We first split the train data and test data in the ratio of 70% and 30%. In the train data, we use the k-fold cross validation for finding (tuning) the parameters. After the finding parameter process is finished, we use the remained test data to evaluate the model.

Using the ‘GridSearchCV‘ for parameter tuning can be burden in the aspect of time and computation. For example, for a model, if we consider 10-fold validation, 3 parameters in which one of each has 5 values, then the model needs to be run 1,250 (= 5 * 5 * 5 * 10) times.

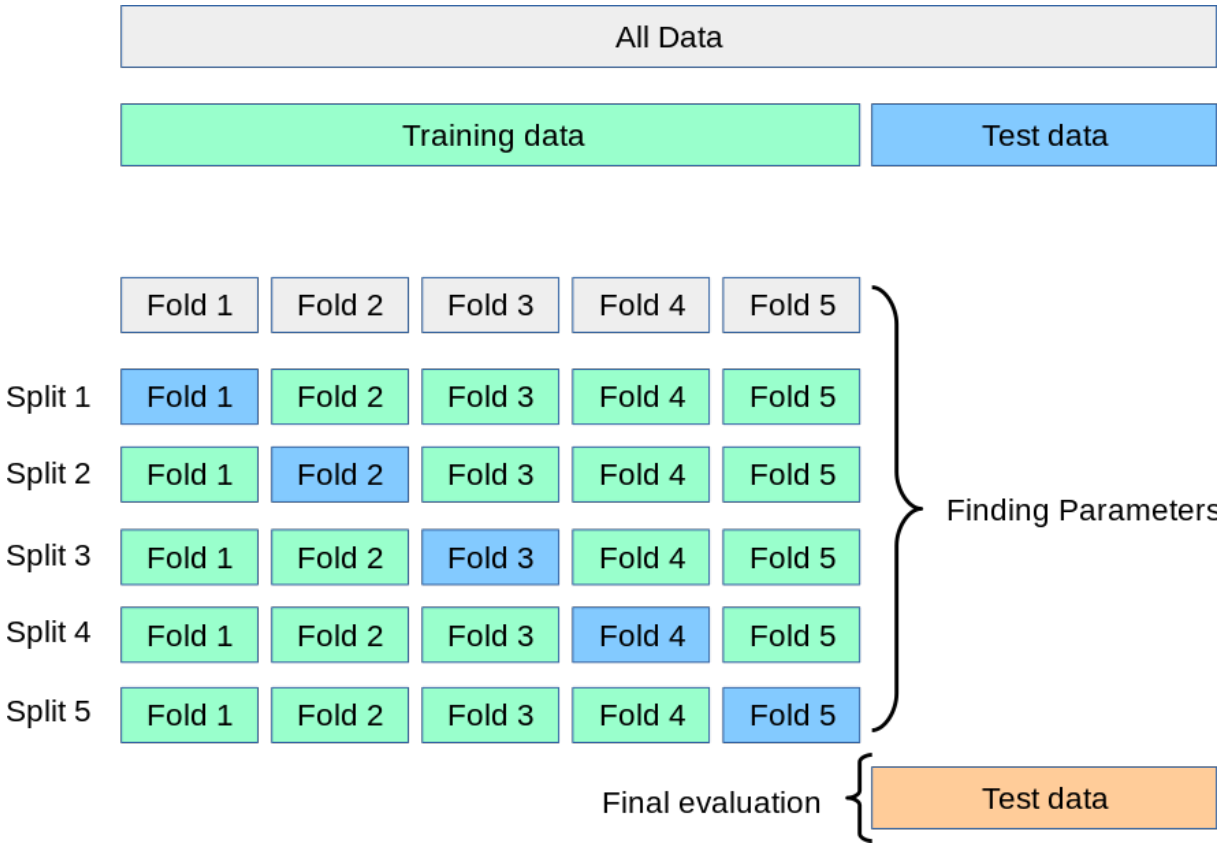


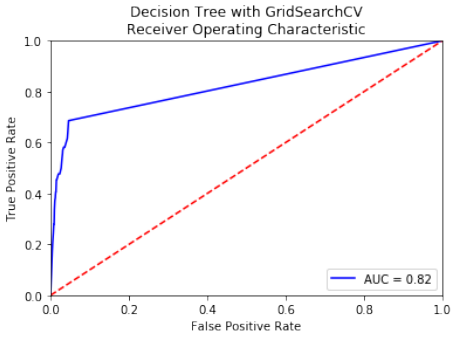
Figure 24: Grid Search Cross Validation for Parameter Tuning using Training Set and Final Evaluation using Test Set [11].

- **Decision Tree with GridSearchCV:** Figure 25 presents the results of Decision Tree with Grid Search Cross Validation (Stratified 5-Folds CV) with the following parameters: {’max_depth’: [50, 75, 100], ’min_samples_leaf’: [1, 2, 4, 8, 10]}. Here is the best parameters for the Decision Tree model: {’max_depth’: 50, ’min_samples_leaf’: 8}. The results model evaluation are as follows: best accuracy: 0.9684, best roc_auc_score: 0.9070, weighted avg precision: 0.9666, weighted avg recall: 0.9684, and weighted avg f1-score: 0.9674.

Please note that when obtaining the ROC and AUC, we used the `predict_prob` for the prediction of the probability of the data instance belonging to each class. This is called a probability prediction where given a new instance, the model returns the probability for each outcome class as a value between 0 and 1.

```
Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      2545
           1       0.50      0.41      0.45        86

   micro avg       0.97      0.97      0.97      2631
   macro avg       0.74      0.70      0.72      2631
weighted avg       0.96      0.97      0.97      2631


Confusion Matrix:
                 [Predicted Not] NeedHelp  [Predicted] NeedHelp
[True Not] NeedHelp                    2510                    35
[True] NeedHelp                          51                    35

Best parameters:  {'max_depth': 50, 'min_samples_leaf': 8}
Best model:  DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=50,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=8, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=7,
            splitter='best')

Best Accuracy: 0.9684
Best roc_auc_score: 0.9070
**Weighted average scores**
Weighted Avg Precision: 0.9666
Weighted Avg Recall: 0.9684
Weighted Avg f1-score: 0.9674
```

(a) Result



(b) AUC graph

Figure 25: Results of the Decision Tree with GridSearchCV.

Decision Tree Visualization: To have the insights from the selected features used for constructing, we visualize the best Decision Tree in Figure 26. The blue nodes and the orange nodes represent the ”NeedHelp” nodes and the ”Not NeedHelp” nodes, respectively.

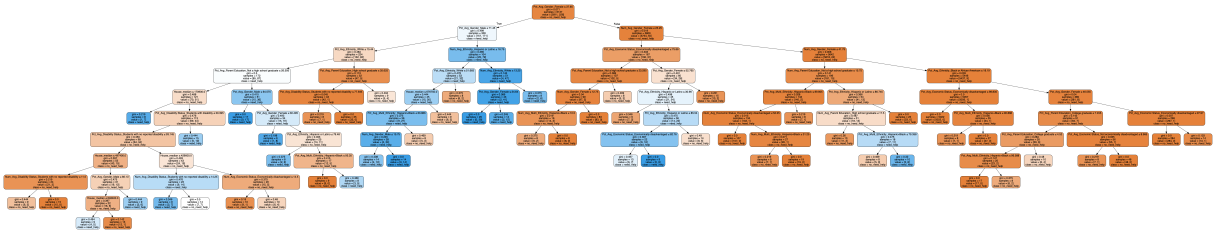


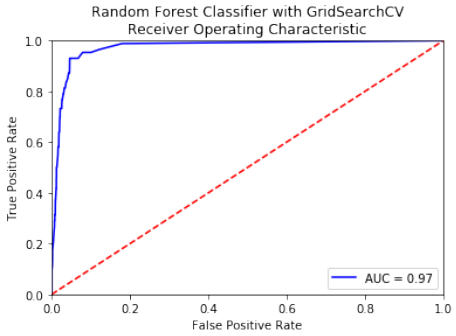
Figure 26: Result of the constructed Decision Tree.

- **Random Forest Classifier with GridSearchCV:** Figure 27 presents the results of Random Forest Classifier with Grid Search Cross Validation (Stratified 5-Folds CV) with the following param-

eters: {'n_estimators': [100, 150, 200], 'max_depth': [100, 150, 200], 'min_samples_leaf': [1, 2, 4]}. Here is the best parameters for the Random Forest Classifier model: {'max_depth': 100, 'min_samples_leaf': 1, 'n_estimators': 200}. The results model evaluation are as follows: best accuracy: 0.9733, best roc_auc_score: 0.9774, weighted avg precision: 0.9711, weighted avg recall: 0.9733, and weighted avg f1-score: 0.9718.

```
Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      2545
           1       0.56      0.45      0.50        86

   micro avg       0.97      0.97      0.97      2631
   macro avg       0.77      0.72      0.74      2631
weighted avg       0.97      0.97      0.97      2631


Confusion Matrix:
                     [Predicted Not] NeedHelp  [Predicted] NeedHelp
[True Not] NeedHelp                      2514                    31
[True] NeedHelp                            47                    39

Best parameters:  {'max_depth': 150, 'min_samples_leaf': 1, 'n_estimators': 100}
Best model:  RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=150, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=100, n_jobs=None,
            oob_score=False, random_state=None, verbose=0,
            warm_start=False)

Best Accuracy: 0.9730
Best roc_auc_score: 0.9779
**Weighted average scores**
Weighted Avg Precision: 0.9707
Weighted Avg Recall: 0.9730
Weighted Avg f1-score: 0.9715
```

(a) Result



(b) AUC graph

Figure 27: Results of the Random Forest Classifier with GridSearchCV.

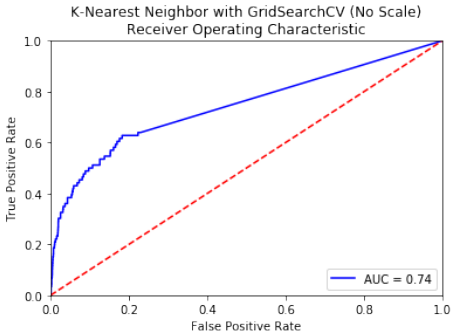
- **K-Nearest Neighbor with GridSearchCV (No Scaling):**

  Figure 28 presents the results of K-Nearest Neighbor without scaling with Grid Search Cross Validation (Stratified 5-Folds CV) with the following parameters: {'n_neighbors': [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29], 'weights': ['uniform', 'distance'], 'metric': ['euclidean', 'manhattan']}. Here is the best parameters for the K-Nearest Neighbor: {'metric': 'manhattan', 'n_neighbors': 11, 'weights': 'distance'}. The results model evaluation are as follows: best accuracy: 0.9650, best roc_auc_score: 0.7309, weighted avg precision: 0.9556, weighted avg recall: 0.9650, and weighted avg f1-score: 0.9526.

- **K-Nearest Neighbor with GridSearchCV (Scaling):**

```
Classification Report:
              precision    recall  f1-score   support

           0       0.97      1.00      0.98      2545
           1       0.38      0.03      0.06        86

   micro avg       0.97      0.97      0.97      2631
   macro avg       0.67      0.52      0.52      2631
weighted avg       0.95      0.97      0.95      2631


Confusion Matrix:
                     [Predicted Not] NeedHelp  [Predicted] NeedHelp
[True Not] NeedHelp                      2540                     5
[True] NeedHelp                            83                     3

Best parameters:  {'metric': 'manhattan', 'n_neighbors': 11, 'weights': 'distance'}
Best model:  KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='manhattan',
           metric_params=None, n_jobs=None, n_neighbors=11, p=2,
           weights='distance')

Best Accuracy: 0.9650
Best roc_auc_score: 0.7309
**Weighted average scores**
Weighted Avg Precision: 0.9556
Weighted Avg Recall: 0.9650
Weighted Avg f1-score: 0.9526
```

(a) Result



(b) AUC graph

Figure 28: Results of the K-Nearest Neighbor with GridSearchCV (without scaling).

Scaling: The independent variables (X_train and X_test) are scaled into the range such that the range is now between 0 and 1 using the min-max scaler. The scaling gives significant accuracy improvement than the model without scaling as in Figure 29.

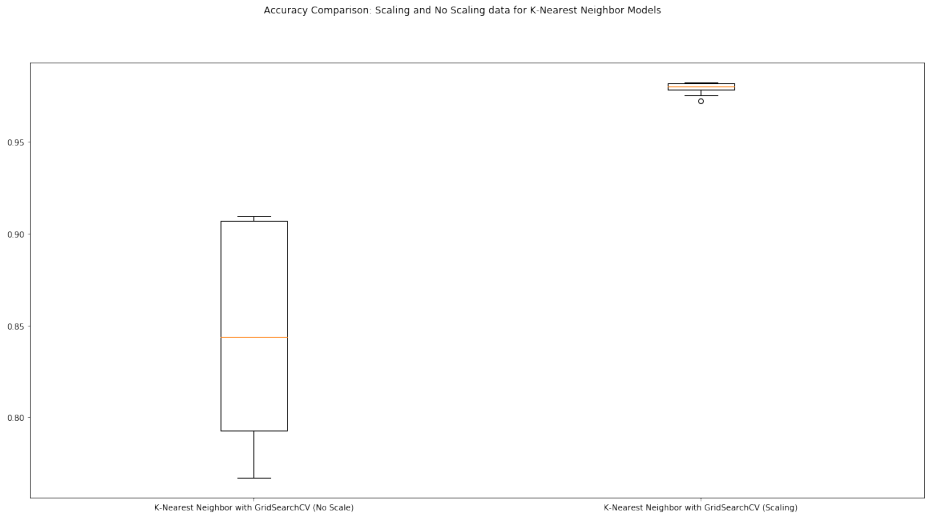


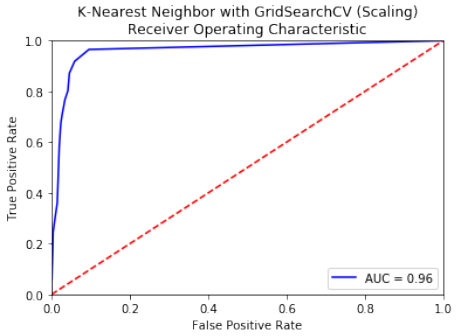
Figure 29: The effects of scaling in K-Nearest Neighbor Models (Scaling Before vs. After).

Figure 30 presents the results of K-Nearest Neighbor using the scaling with Grid Search Cross Vali-

dation (Stratified 5-Folds CV). The cross validated parameters are same with the K-Nearest Neighbor models without scaling. Here is the best parameters for the K-Nearest Neighbor: {'metric': 'manhattan', 'n_neighbors': 19, 'weights': 'uniform'}. The results model evaluation are as follows: best accuracy: 0.9728, best roc_auc_score: 0.9618, weighted avg precision: 0.9692, weighted avg recall: 0.9728, and weighted avg f1-score: 0.9695.

```
Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      2545
           1       0.53      0.30      0.39        86

   micro avg       0.97      0.97      0.97      2631
   macro avg       0.75      0.65      0.68      2631
weighted avg       0.96      0.97      0.96      2631


Confusion Matrix:
                     [Predicted Not] NeedHelp  [Predicted] NeedHelp
[True Not] NeedHelp                      2522                    23
[True] NeedHelp                            60                    26

Best parameters:  {'metric': 'manhattan', 'n_neighbors': 19, 'weights': 'uniform'}
Best model:  KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='manhattan',
           metric_params=None, n_jobs=None, n_neighbors=19, p=2,
           weights='uniform')

Best Accuracy: 0.9728
Best roc_auc_score: 0.9618
**Weighted average scores**
Weighted Avg Precision: 0.9692
Weighted Avg Recall: 0.9728
Weighted Avg f1-score: 0.9695
```

(a) Result



(b) AUC graph

Figure 30: Results of the K-Nearest Neighbor with GridSearchCV (Scaling).

5.2.4 Results of Classification:

In Figure 31, we have plotted the accuracy for models using the Grid Search Cross Validation. The range for each model indicates the accuracy results obtained from all parameters. **The Random Forest Classifier model has the highest accuracy.** The difference between maximum and minimum accuracy of the Random Forest Classifier is very small. We also noted that **after applying the scaler to the K-Nearest Neighbor model, the accuracy has been significantly improved**.

Finally, Table 3 shows the results of the performance for classification models. The **Random Forest Classifier with GridSearchCV worked best** with best accuracy: 0.9733, best roc_auc_score: 0.9774, weighted avg precision: 0.9711, weighted avg recall: 0.9733, and weighted avg f1-score: 0.9718. The

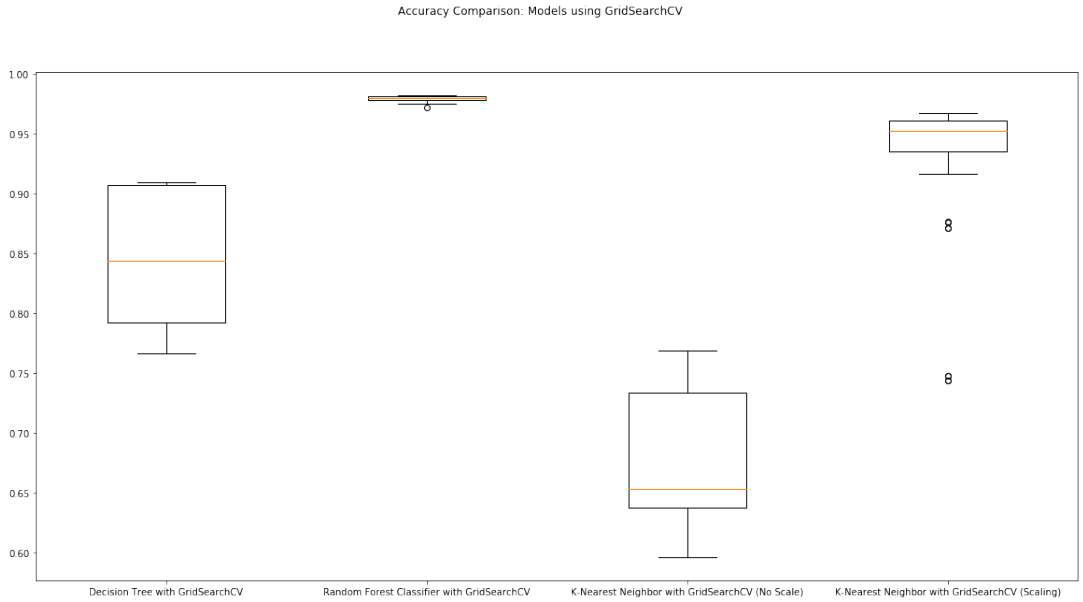


Figure 31: Boxplots of Accuracy Comparison for GridSearch CV Models.
Decision Tree, Random Forest Classifier, K-Nearest Neighbor (no scale), and K-Nearest Neighbor (scaling) with GridSearchCV

best parameters for the Random Forest Classifier model is {'max_depth': 100, 'min_samples_leaf': 1, 'n_estimators': 200}.

Model Name	accuracy	auc	precision	recall	f1
Logistic Regression with Stratified 5-Folds CV	0.9656	0.9656	0.9646	0.9656	0.9597
Decision Tree with Stratified 5-Folds CV	0.9596	0.7320	0.9660	0.9596	0.9614
Decision Tree with GridSearchCV	0.9684	0.9070	0.9666	0.9684	0.9674
Random Forest Classifier with GridSearchCV	0.9733	0.9774	0.9711	0.9733	0.9718
K-Nearest Neighbor with GridSearchCV (No Scale)	0.9650	0.7309	0.9556	0.9650	0.9526
K-Nearest Neighbor with GridSearchCV (Scaling)	0.9728	0.9618	0.9692	0.9728	0.9695

Table 3: Results for the performance of classification models.

6 Limitation and Recommendation

Limitation. I assumed that the family incomes of students could be an important factor affecting the school performance achievements in scores. Therefore, I used the median/mean house prices from Zillow [4] by matching the zip codes of schools. However, after analyzing in detail, I found that the range

covered by postal codes is too broad, so median/mean housing prices do not properly reflect the school's family income. For example, Mission Education Center school (located in San Francisco Unified School District in San Francisco County) has the median house price of $1,662,300 but 89% of students from low-income families. I should have considered another complementary variables for the family incomes of students such as "low income family ratio".

The additional datasets can be considered to obtain more accurate prediction or more valuable insights. For example, we can collect the following data:

- Teacher demographics (from Civil Rights Data Collection)
- School profile, school reviews, school census data, nearby schools

Recommendation. It is obvious that that the high scores of schools are strongly correlated with the students raised in high-income families. The students in high income families are more exposed to various learning opportunities including lessons in sports, musical instruments, arts or other activities. Learning achievement naturally leads to academic achievement, but students who have not had the pleasure of learning may give up their academic endeavors early or even not start studying at all.

Thus, in my opinion, **the schools need the help** if the schools have more than 73.14% of students of low-income families, the house median prices are less than $335,500 (more urgent help is needed when the house prices are when less than $194,350), the parents who do not graduate high schools is more than 89.1%, the parents who do not graduate colleges is more than 84.9%, or the Hispanic or Black students is more than 67.2%. For reference, we provide the distribution graphs of important features for bottom 5% (181 schools) and top 5% (179 schools) performing school data in Figure 32. We made the above suggestions by referencing the decision trees constructed using each of important features as Figures 33.

I examined the Google reviews written by the students who attended the bottom 5% schools, and consistently noticed the following comments:

"The class is boring. There are no effective approach for homework. Teachers are not effective in teaching. The teachers don't seem interested in the students succeed."

To increase academic achievement effectively, more budgets need to be allocated to schools to hire teachers or staffs for providing the 1 to 1 interaction or private tutoring. These schools need more money to purchase academic applications and electronic devices that help students learn in the fun and independent environment.

7 Conclusion and Future Work

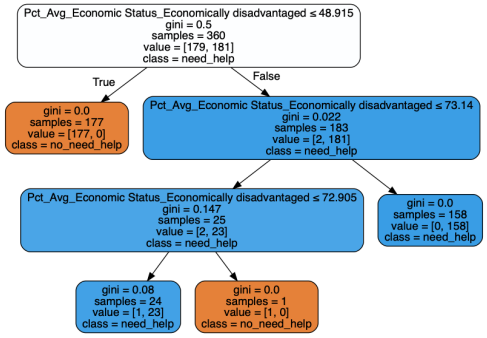
We have analyzed the CAASPP score data to help to predict and find the inferior groups of schools that indeed need help.

Data Wrangling. In the data wrangling, we performed data cleaning, fixing missing values, and adding new columns. Missing values are imputed using the statistics of the *mean* of each column in which the missing values are located. We add the new variables by manipulating or merging existing variables to tell new insights or to reduce the dimensionality.

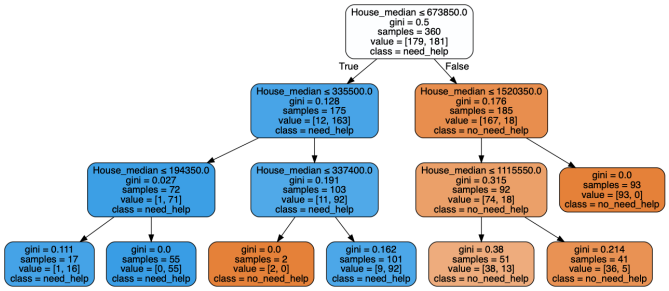
Exploratory Data Analysis. In the data visualization, we investigated the three research questions. To answer how the students are different in achievement levels, we provided the **bar plots for the comparison for each category of gender, ethnicity, english-language fluency, economic status, disability status, and parent educations**. For utilizing the advanced features, we also used the Plotly libraries for drawing interactive graphs. The major finding are as follows:

- Female students exceed male students in English, while male students exceed female students in Mathematics.
- Asian students achieve the best performance, while Black or African American and American Indian or Alaska Native students achieve the lowest performance in both English and mathematics.

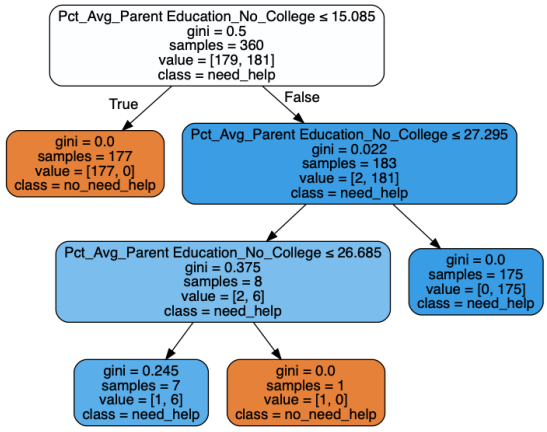
Figure 32: Distribution of important features for bottom 5% and top 5% (Level 1 vs Level 4).
X-axis: 1) House median($), 2) Parent Education: Graduates(%), 3) Parent Education: Not High School Graduates(%), 4) Ethnicity: Hispanic and Black(%), and 5) Economically Disadvantaged(%).



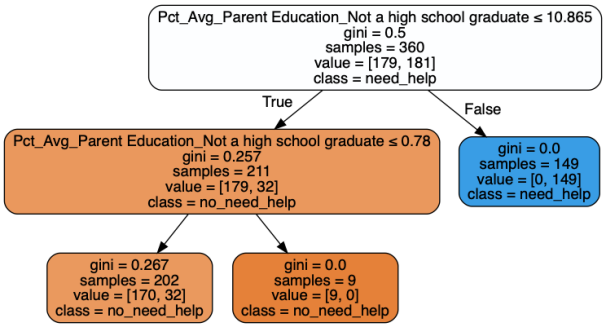
(a) Percentage of economically disadvantaged students



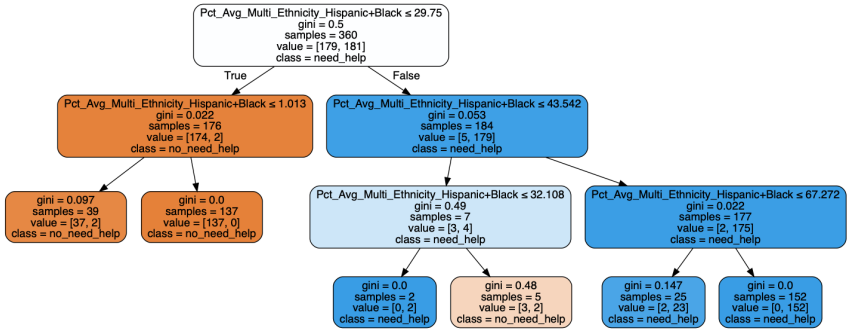
(b) House median prices



(c) Percentage of students whose parents do not graduated the colleges



(d) Percentage of students whose parents do not graduated the high schools



(e) Percentage of Hispanic and Black students

Figure 33: Decision Trees for each of important feature using the bottom 5% and top 5% performing school data.

- Initial fluent English proficient students achieve the best performance in both English and mathematics.

- The economically disadvantaged students have much more difficulties than not-economically disadvantaged students.
- Only the small number of students with disabilities could achieve the best performance.
- The higher the level of parental education, the higher the achievement of students.

To find the features in the top and bottom performance groups, we compared the best and worst 10% performing counties (10% out of 58 = 5 counties) using the bar plots. We found that the test performance is closely related to the economic capabilities of the family to which the student belongs. We could observe that Hispanic and Latino students are far more likely to be in the worst performing group than the best performing group. In contrast, Asian and white students are more likely to be in the best performing group than the worst performing group. For the last, to investigate how house prices are correlated to the exceeded scores or inferior scores, we analyzed the correlations using scatterplots. We observed the strong positive correlations between the "Percentage of Standard Exceeded and the house prices whereas the strong negative correlations between the"Percentage of Standard Not Met and the house prices.

In the exploratory data analysis, we used the inferential statistics to identify significant features. A significant number of features could be redundant and irrelevant, therefore it is important to apply feature selection/dimension reduction. We performed the statistical hypothesis testing, correlation test, and feature selection for getting rid of the student group information for generating less number of features. We first test whether the means of two independent samples are significantly different and eliminated or merged the weak affecting student group indicators. For correlation analysis, we used the matrix with Heatmap, Pearson's correlation coefficient, and Spearman's rank correlation methods and found the meaningful strong relationships between pairs of features (e.g., house prices and high scores, advanced education of students' parents and high scores). For feature selection, using the univariate selection and feature importance techniques, we could obtain a score for each feature of the data and prioritize the in the order of importance.

Models. The aim is to predict the inferior scores of schools. We built machine learning models using regression and classification algorithms. The regression algorithm predicts the percentage of students who do not meet the standard. The classification algorithm predicts if the schools "need help" or "do not need help". We set the "need help" schools that has more than "80% of the standard not met" students (312 out of 8,786 schools). We tried various machine learning techniques to pick the one which performs best. For regression, out of 5 different models, we obtained the best regression model using the random forest regressor with 10 folds cross validation with the accuracy of RMSE 10.77, MAE 7.69, and R^2 0.68. For classification, we tried to solve the class imbalanced problems using the Stratified K-fold cross validation and the weighted evaluation metrics to reflect the mass of the classes. In addition, we scaled the training data and significantly improved the accuracy of the K-Nearest Neighbor algorithm. As a result, out of 5 different models, we obtained the best classification model using the random forest classifier based on grid search cross validation (three parameters each of three values) with the accuracy 0.97, AUC 0.98, precision 0.97, recall 0.97, and f1-score 0.97.

For the future work, to identify the factors that could effectively improve the scores, we will investigate the scores of the 5 consecutive years (2014 to 2018 available in [1]). We expect to find the important features on the schools in which the scores have been dramatically improved.

For the final comment, I hope this school score prediction analysis could be a little help to administrators to the California state departments of education, teachers and parents to broadening educational opportunities.

I give my special thanks to Tony Baek who has been my mentor for completing this project.

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.

[10] Stratified Cross Validation, https://stackoverflow.com/questions/32615429/k-fold-stratified-cross-validation-with-imbalanced-classes.

[11] GridSearchCV for Parameter Tuning, https://scikit-learn.org/stable/modules/cross_validation.html.