Python for Data Analysis Final Project

By Amee Tan

Data Set Description

5th grade student data from a Title I middle school in San Jose.

Data collected was collected on my students in the 2017-2018 and 2018-2019 school years.

206 rows and 38 columns with 7,828 entries in the original dataset

Data Set Overview

Key Student Data Attributes:

- Race
- Gender
- Primary home language
- English language proficiency level
- Economic Status
- MAP Math and ELA scores
- Fall to Spring MAP growth
- SBAC Math and ELA scores

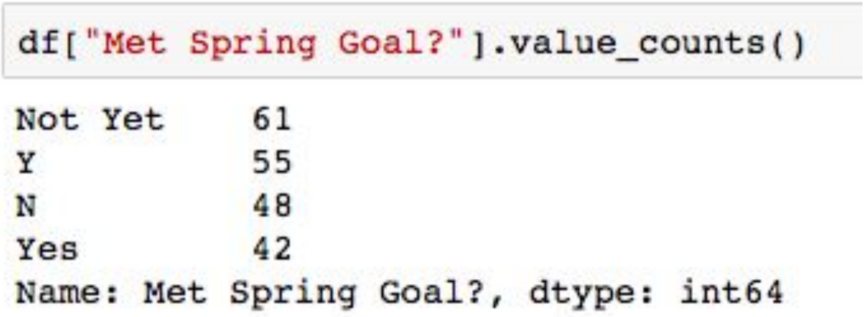
Questions

1. What are the demographics of the students?

2. Which attributes are most strongly correlated to student academic performance?

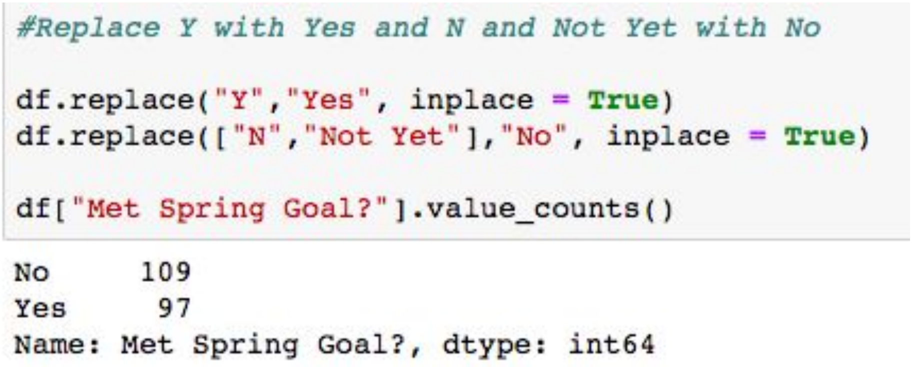
3. What other gaps in student performance exist that need to be addressed?

Data Cleaning - Replacing values



```
df["Met Spring Goal?"].value_counts()
```

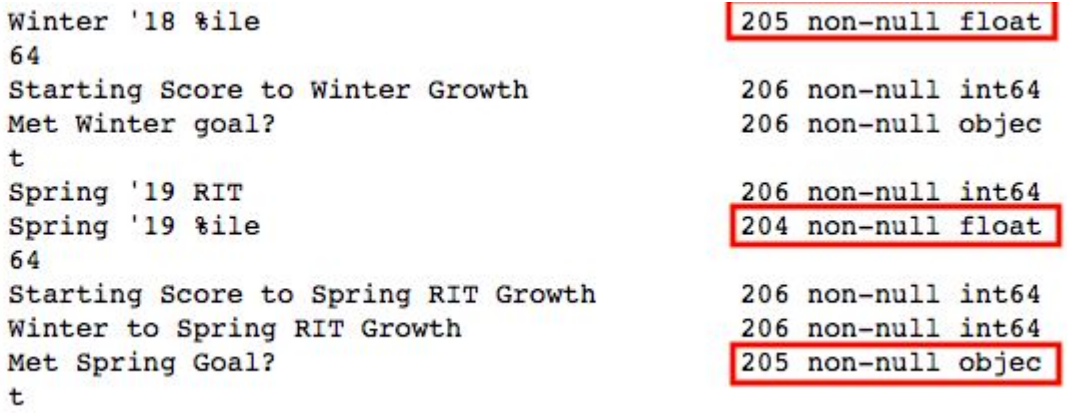
```
Not Yet     61
Y           55
N           48
Yes         42
Name: Met Spring Goal?, dtype: int64
```



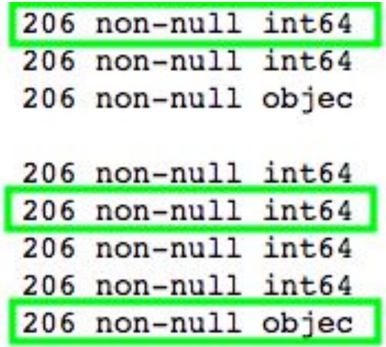
```
#Replace Y with Yes and N and Not Yet with No

df.replace("Y","Yes", inplace = True)
df.replace(["N","Not Yet"],"No", inplace = True)

df["Met Spring Goal?"].value_counts()
```

```
No      109
Yes      97
Name: Met Spring Goal?, dtype: int64
```

Data Cleaning-
df.info() was helpful in finding missing values



```
Winter '18 %ile                             205 non-null float
64
Starting Score to Winter Growth             206 non-null int64
Met Winter goal?                            206 non-null objec
t
Spring '19 RIT                              206 non-null int64
Spring '19 %ile                             204 non-null float
64
Starting Score to Spring RIT Growth         206 non-null int64
Winter to Spring RIT Growth                 206 non-null int64
Met Spring Goal?                            205 non-null objec
t
```



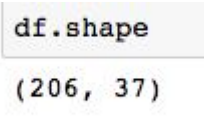
```
206 non-null int64
206 non-null int64
206 non-null objec

206 non-null int64
206 non-null int64
206 non-null int64
206 non-null int64
206 non-null objec
```

Data Cleaning- Adding Columns

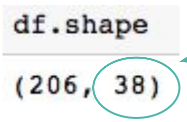
Converting categorical strings into integers using replace

```
df["Num Math AL"] = df["Math Achievement Level"].replace({"Standard
Exceeded": 4, "Standard Met": 3, "Standard Nearly Met": 2, "Standard Not
Met": 1})
```

Verify the results

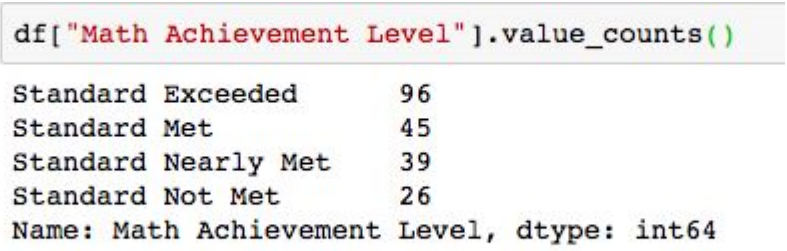


```
df.shape

(206, 37)
```

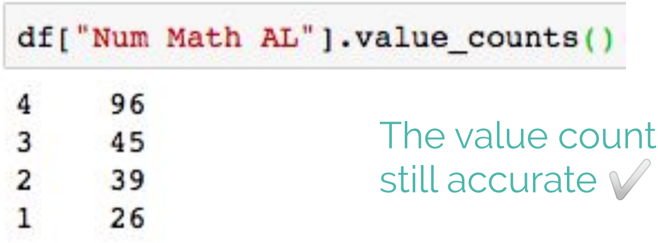


```
df.shape

(206, 38)
```

Now there was an additional column ✓



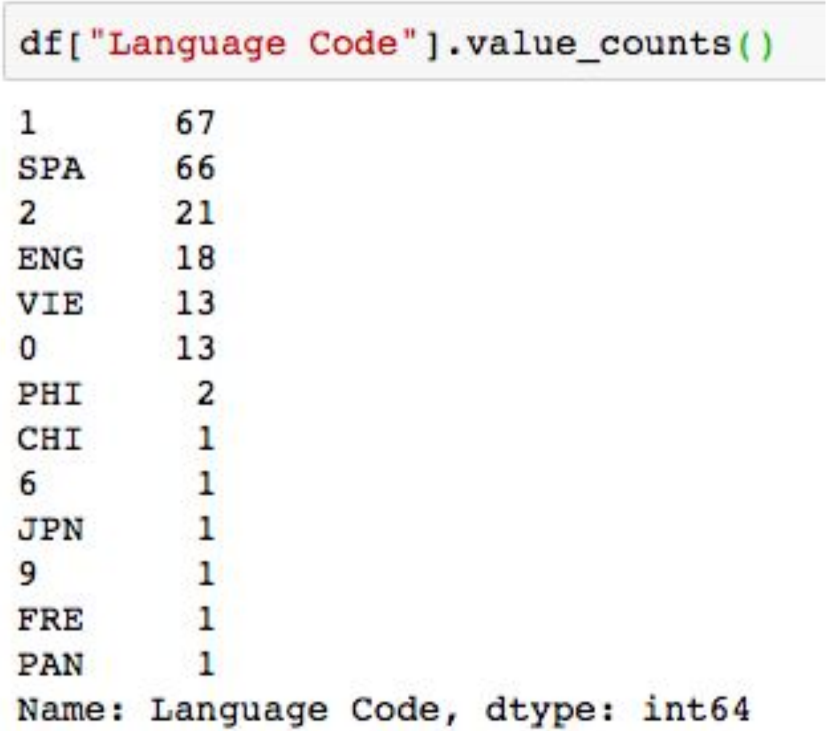
```
df["Math Achievement Level"].value_counts()

Standard Exceeded       96
Standard Met            45
Standard Nearly Met     39
Standard Not Met        26
Name: Math Achievement Level, dtype: int64
```



```
df["Num Math AL"].value_counts()

4     96
3     45
2     39
1     26
```

The value counts were still accurate ✓

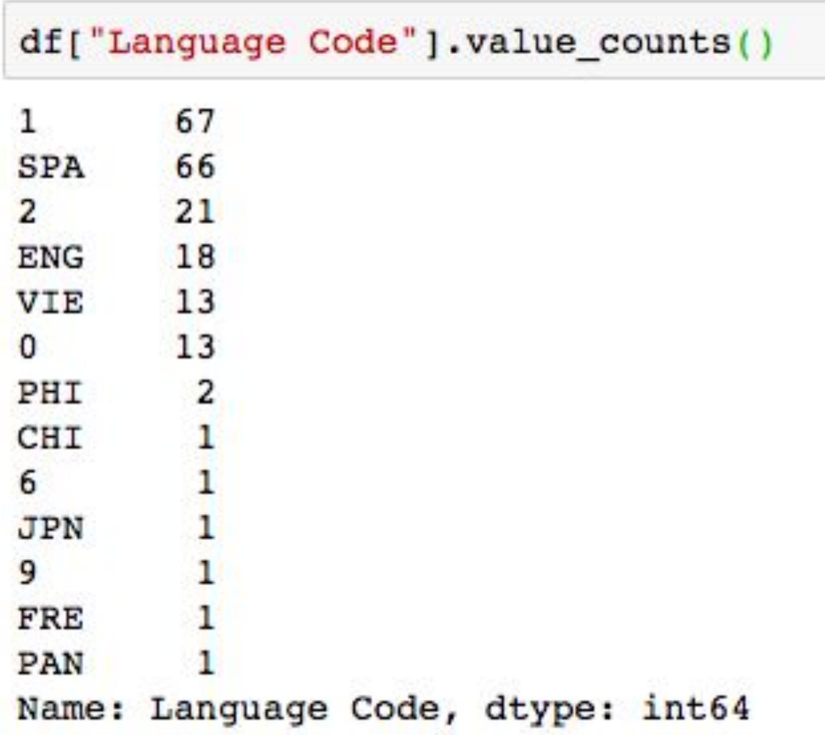
Data Cleaning - Language Code



```
df["Language Code"].value_counts()
```

```
1      67
SPA    66
2      21
ENG    18
VIE    13
0      13
PHI     2
CHI     1
6       1
JPN     1
9       1
FRE     1
PAN     1
Name: Language Code, dtype: int64
```

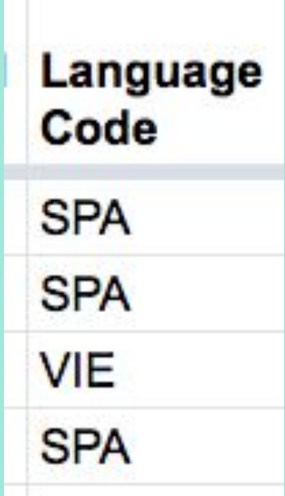
Data Cleaning - Language Code



```
df["Language Code"].value_counts()
```

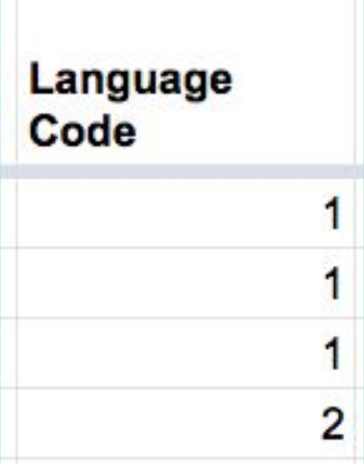
```
1       67
SPA     66
2       21
ENG     18
VIE     13
0       13
PHI      2
CHI      1
6        1
JPN      1
9        1
FRE      1
PAN      1
Name: Language Code, dtype: int64
```

2017-2018 School Year:



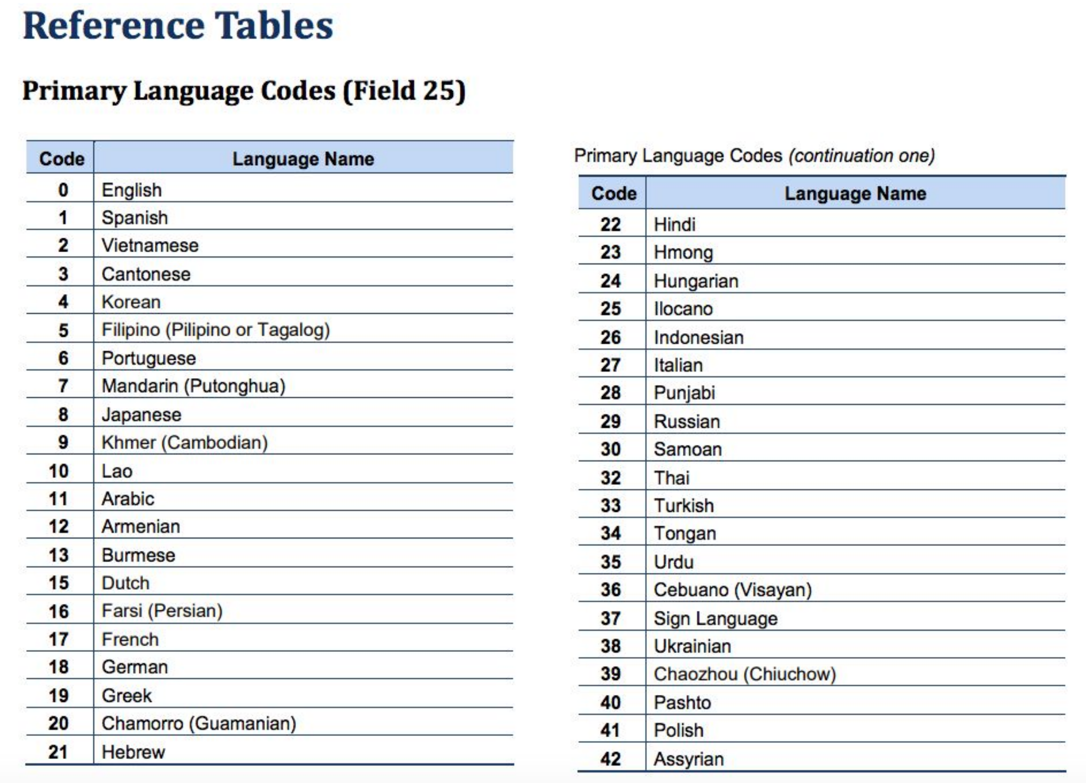
Language Code
SPA
SPA
VIE
SPA

2018-2019 School Year:



Language Code
1
1
1
2

Data Cleaning - Language Code



Reference Tables

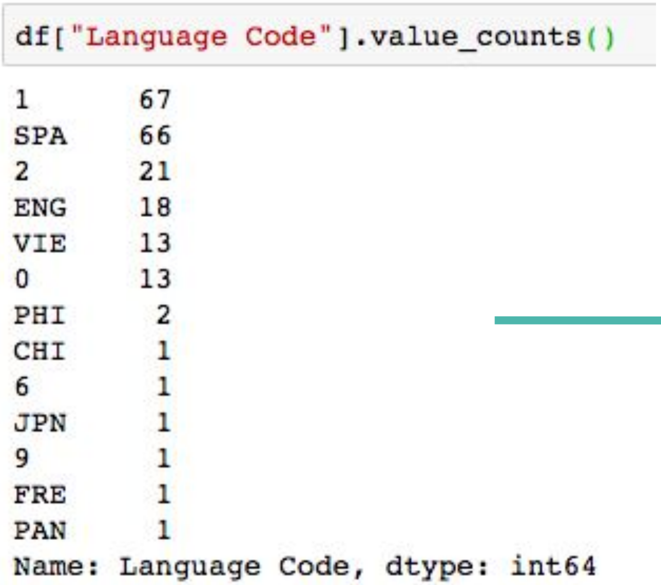
Primary Language Codes (Field 25)

Code	Language Name
0	English
1	Spanish
2	Vietnamese
3	Cantonese
4	Korean
5	Filipino (Pilipino or Tagalog)
6	Portuguese
7	Mandarin (Putonghua)
8	Japanese
9	Khmer (Cambodian)
10	Lao
11	Arabic
12	Armenian
13	Burmese
15	Dutch
16	Farsi (Persian)
17	French
18	German
19	Greek
20	Chamorro (Guamanian)
21	Hebrew

Primary Language Codes *(continuation one)*

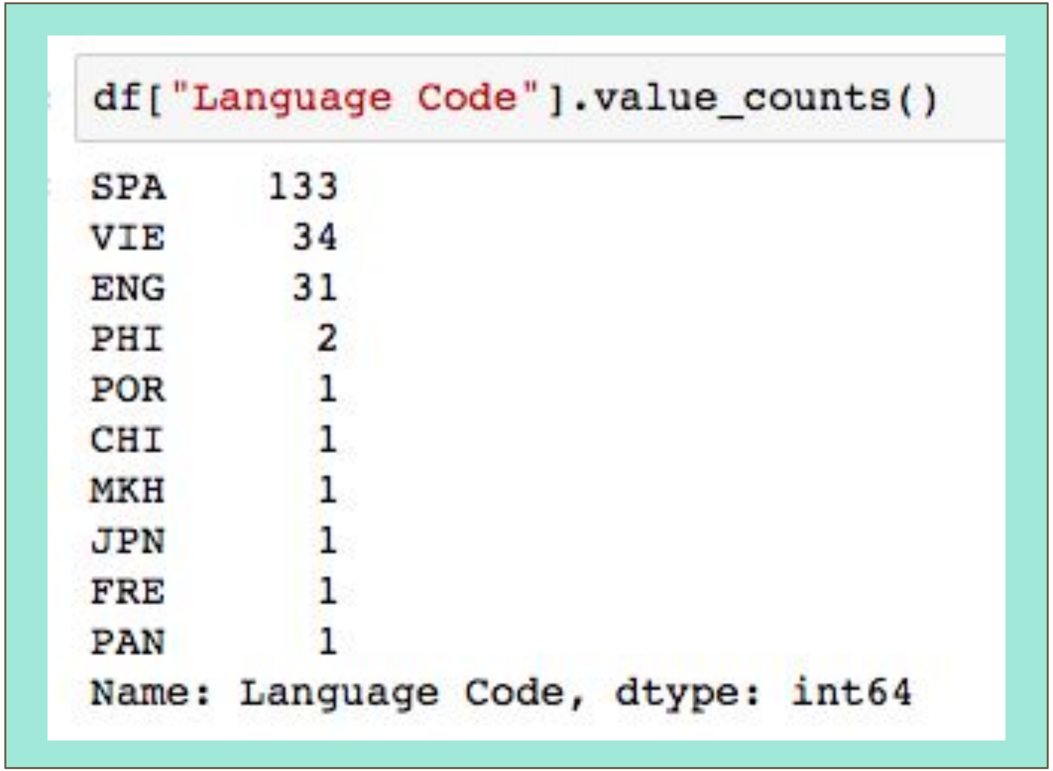
Code	Language Name
22	Hindi
23	Hmong
24	Hungarian
25	Ilocano
26	Indonesian
27	Italian
28	Punjabi
29	Russian
30	Samoan
32	Thai
33	Turkish
34	Tongan
35	Urdu
36	Cebuano (Visayan)
37	Sign Language
38	Ukrainian
39	Chaozhou (Chiuchow)
40	Pashto
41	Polish
42	Assyrian

Data Cleaning - Language Code



```
df["Language Code"].value_counts()
```

```
1      67
SPA    66
2      21
ENG    18
VIE    13
0      13
PHI     2
CHI     1
6       1
JPN     1
9       1
FRE     1
PAN     1
Name: Language Code, dtype: int64
```



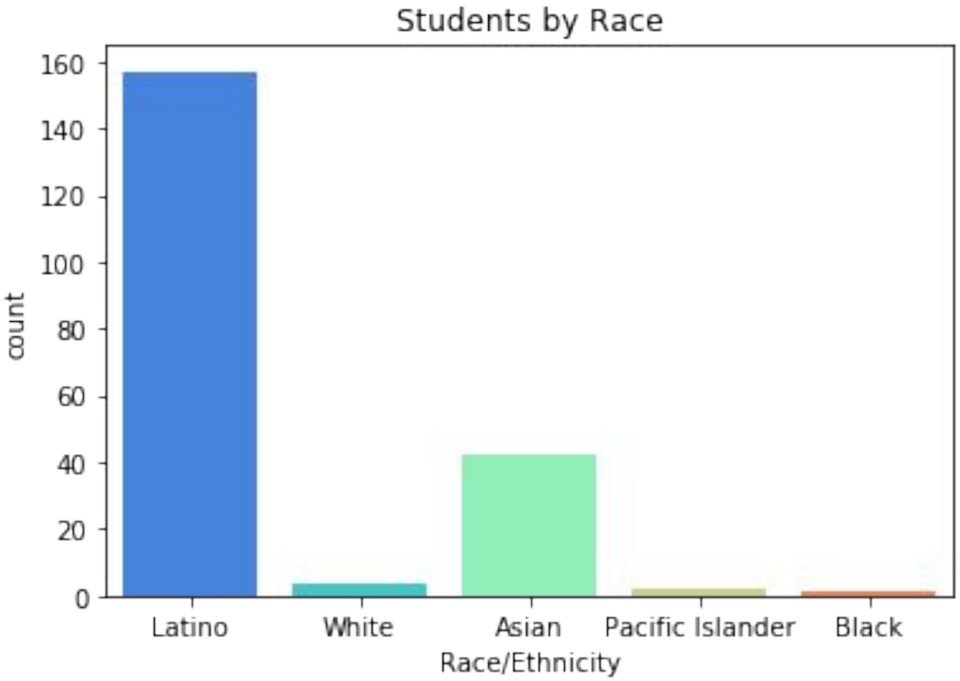
```
df["Language Code"].value_counts()
```

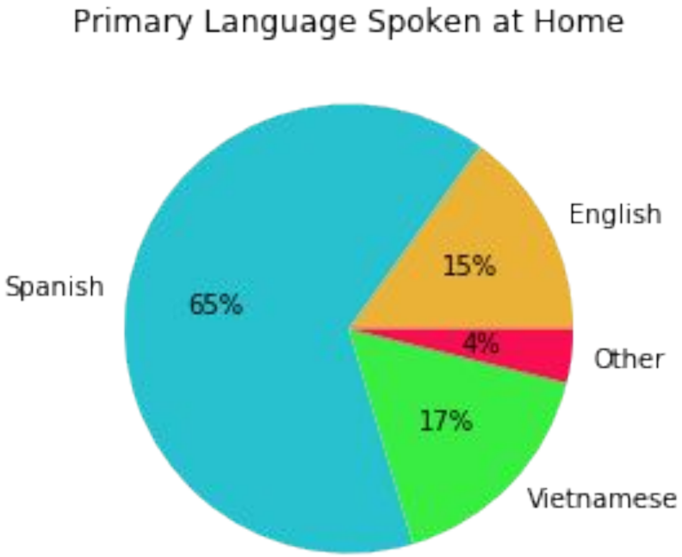
```
SPA    133
VIE     34
ENG     31
PHI      2
POR      1
CHI      1
MKH      1
JPN      1
FRE      1
PAN      1
Name: Language Code, dtype: int64
```

Exploratory Analysis

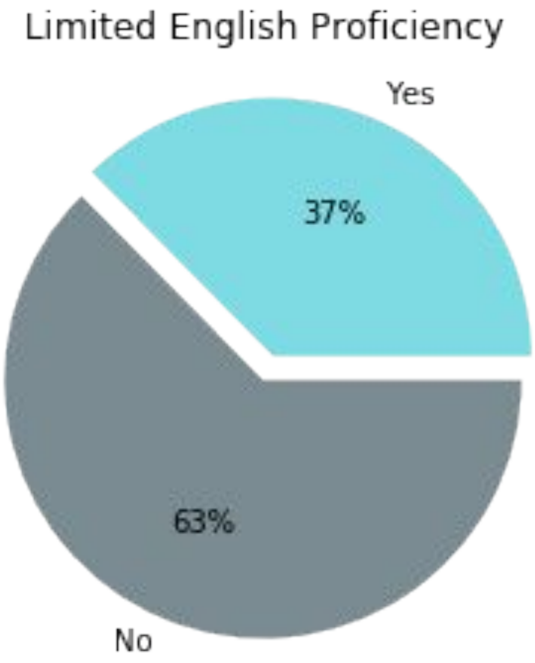
Visualizations

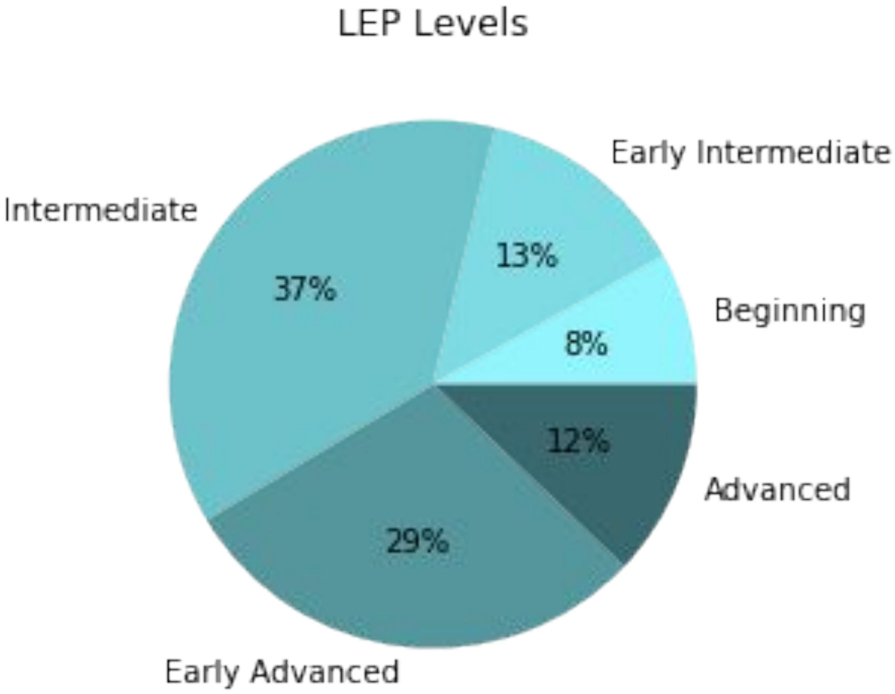
Student Demographics



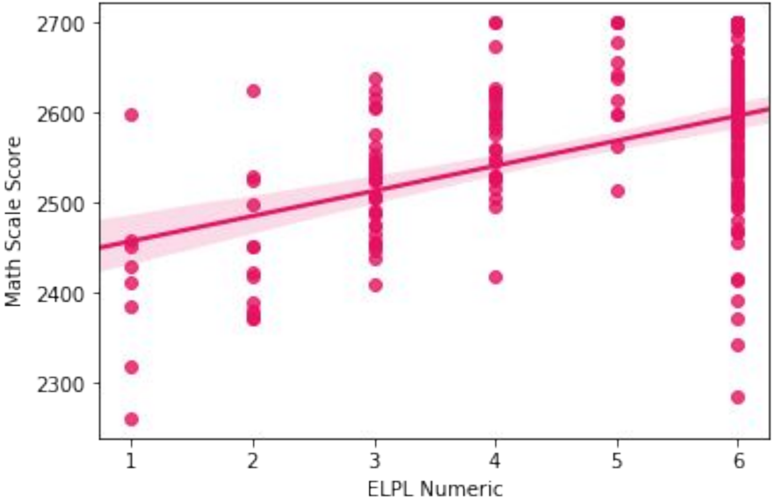


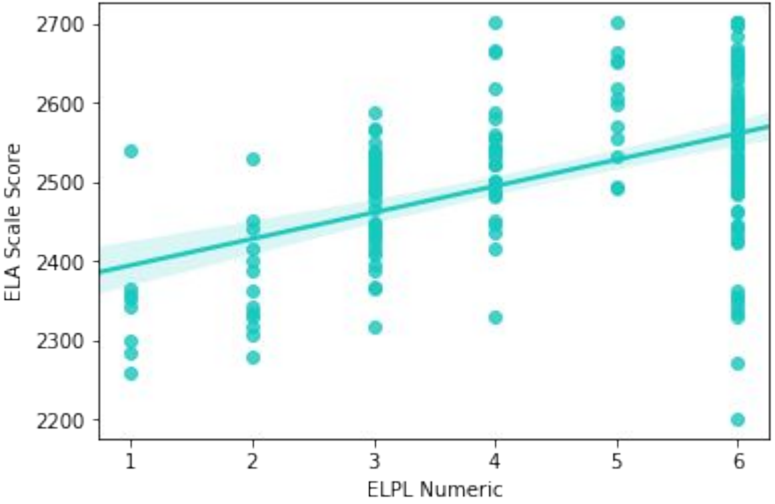
Limited English Language Proficiency



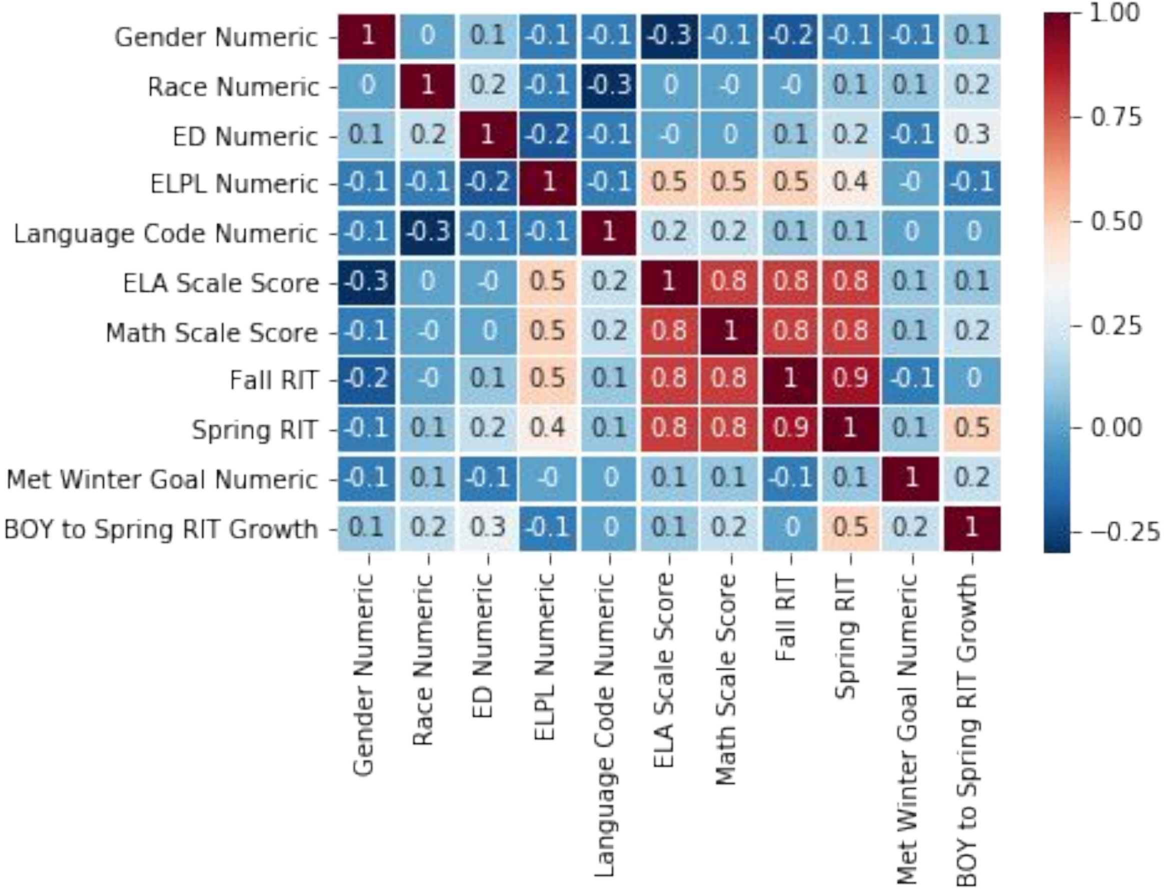


English Language Proficiency is correlated to student performance



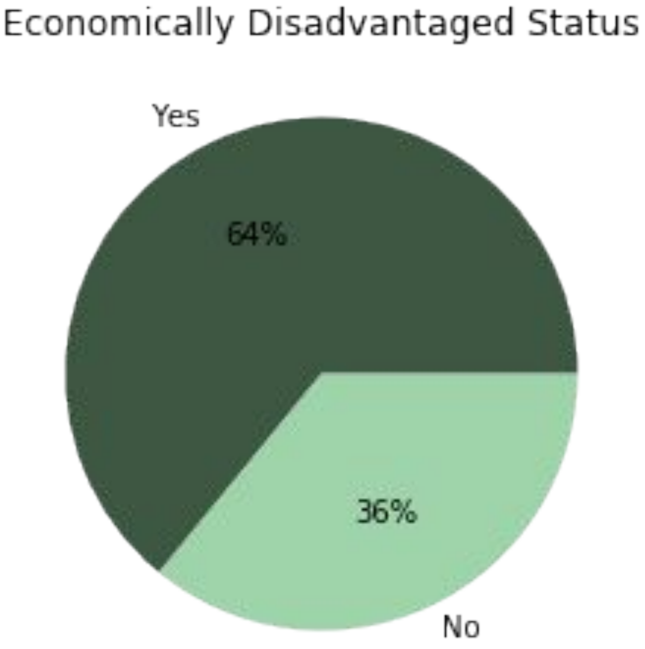


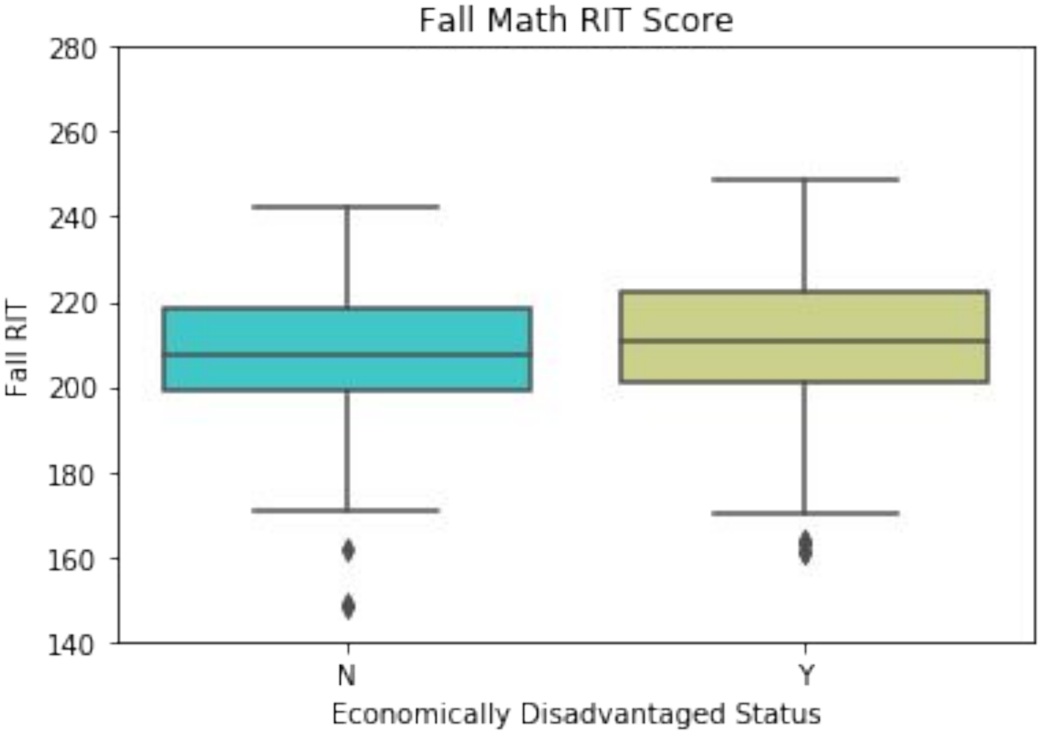
Correlation heat map



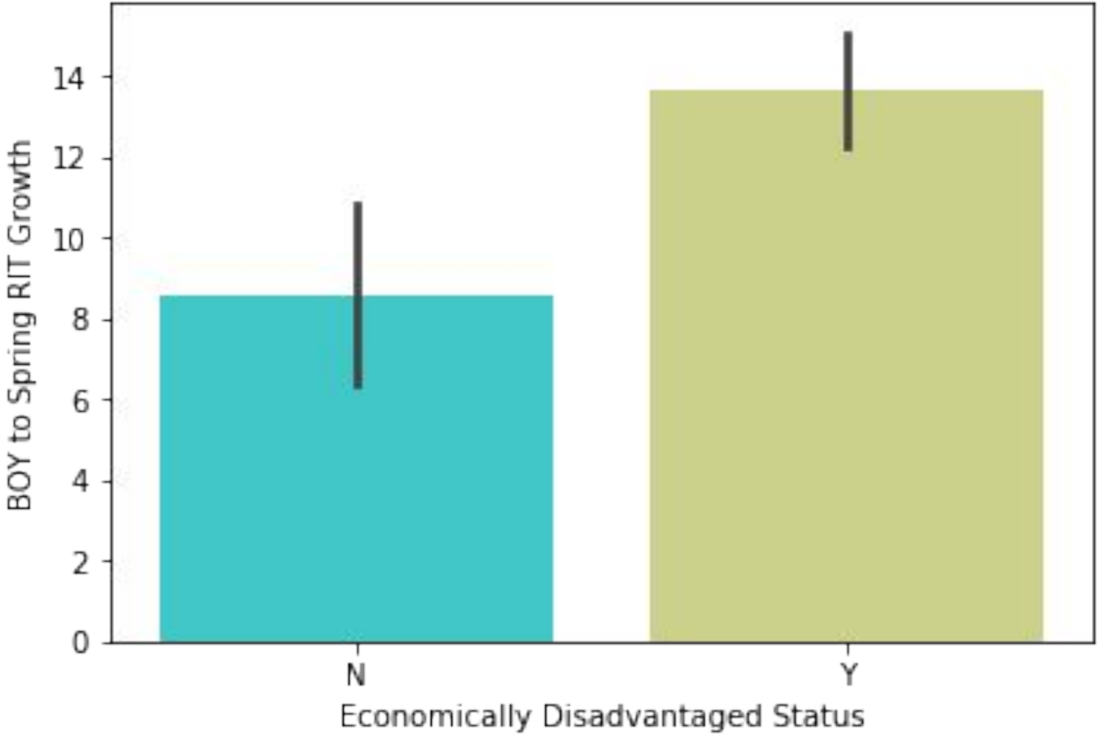
	Gender Numeric	Race Numeric	ED Numeric	ELPL Numeric	Language Code Numeric	ELA Scale Score	Math Scale Score	Fall RIT	Spring RIT	Met Winter Goal Numeric	BOY to Spring RIT Growth
Gender Numeric	1	0	0.1	-0.1	-0.1	-0.3	-0.1	-0.2	-0.1	-0.1	0.1
Race Numeric	0	1	0.2	-0.1	-0.3	0	-0	-0	0.1	0.1	0.2
ED Numeric	0.1	0.2	1	-0.2	-0.1	-0	0	0.1	0.2	-0.1	0.3
ELPL Numeric	-0.1	-0.1	-0.2	1	-0.1	0.5	0.5	0.5	0.4	-0	-0.1
Language Code Numeric	-0.1	-0.3	-0.1	-0.1	1	0.2	0.2	0.1	0.1	0	0
ELA Scale Score	-0.3	0	-0	0.5	0.2	1	0.8	0.8	0.8	0.1	0.1
Math Scale Score	-0.1	-0	0	0.5	0.2	0.8	1	0.8	0.8	0.1	0.2
Fall RIT	-0.2	-0	0.1	0.5	0.1	0.8	0.8	1	0.9	-0.1	0
Spring RIT	-0.1	0.1	0.2	0.4	0.1	0.8	0.8	0.9	1	0.1	0.5
Met Winter Goal Numeric	-0.1	0.1	-0.1	-0	0	0.1	0.1	-0.1	0.1	1	0.2
BOY to Spring RIT Growth	0.1	0.2	0.3	-0.1	0	0.1	0.2	0	0.5	0.2	1

Surprising result

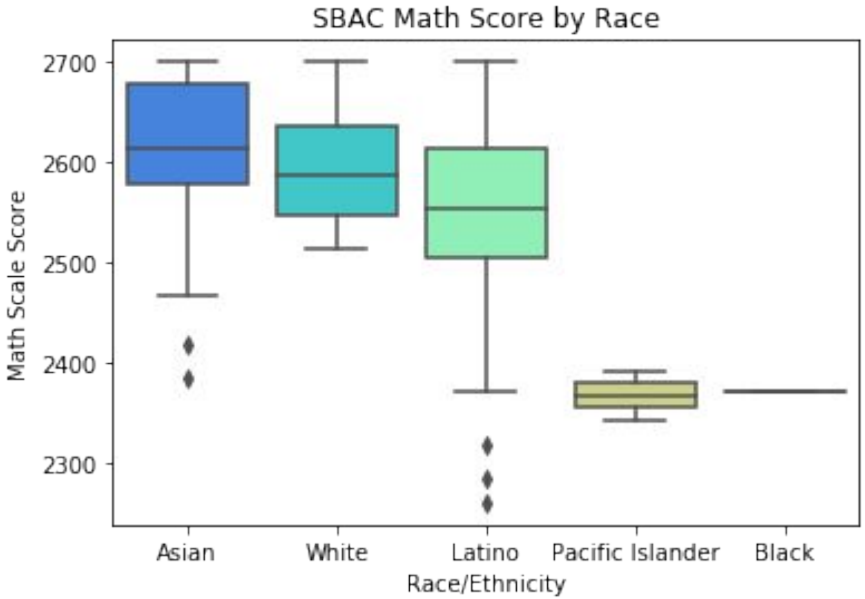


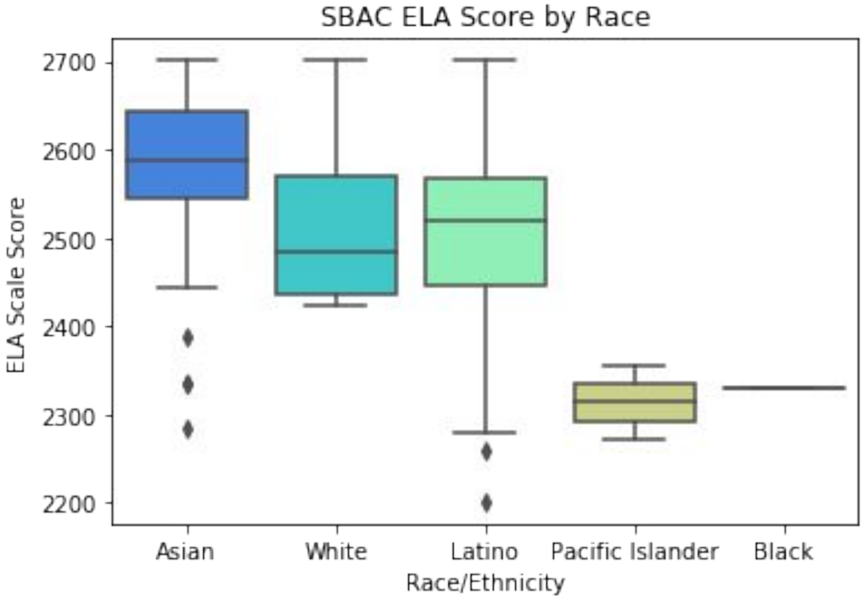


Surprising Result
ED students actually grew more than non-ED students throughout the year



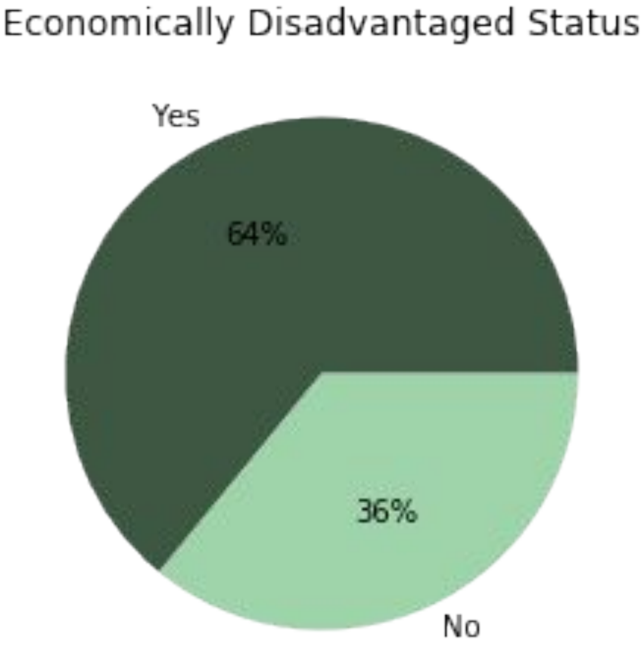
Racial achievement disparities are obvious.....

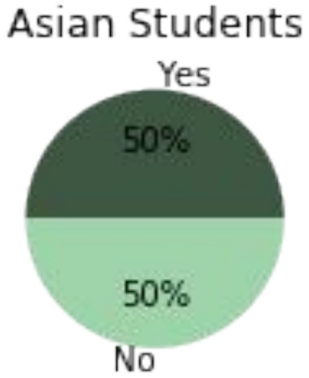


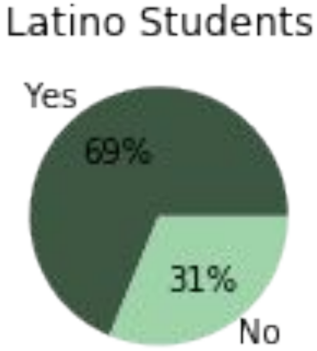


.....but the reason is still unclear.

Economically disadvantaged status is a potential factor….

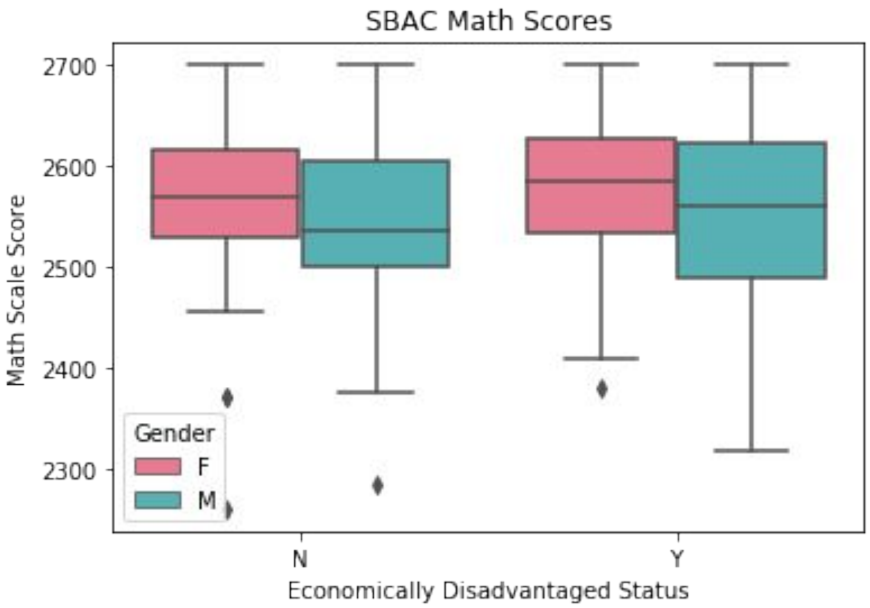

Economically Disadvantaged Status

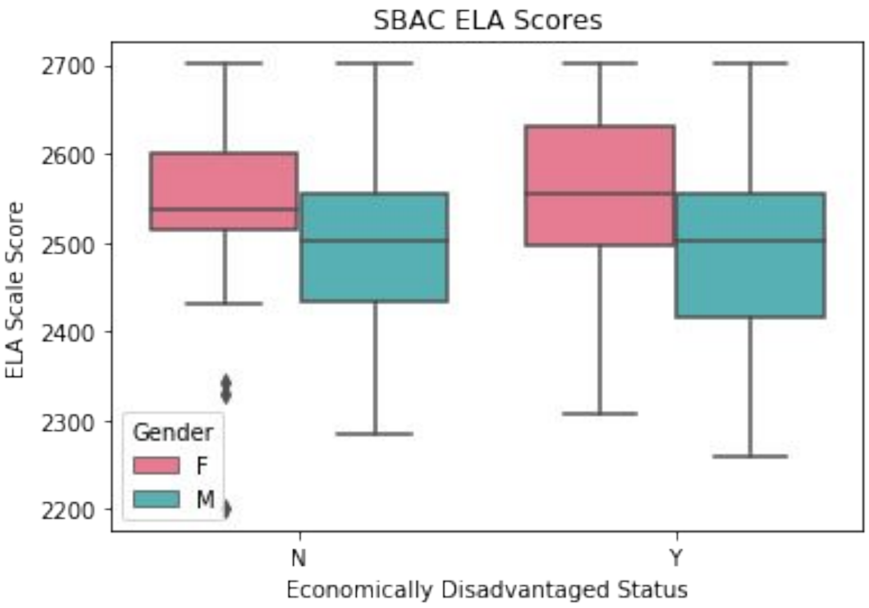

Asian Students


Latino Students

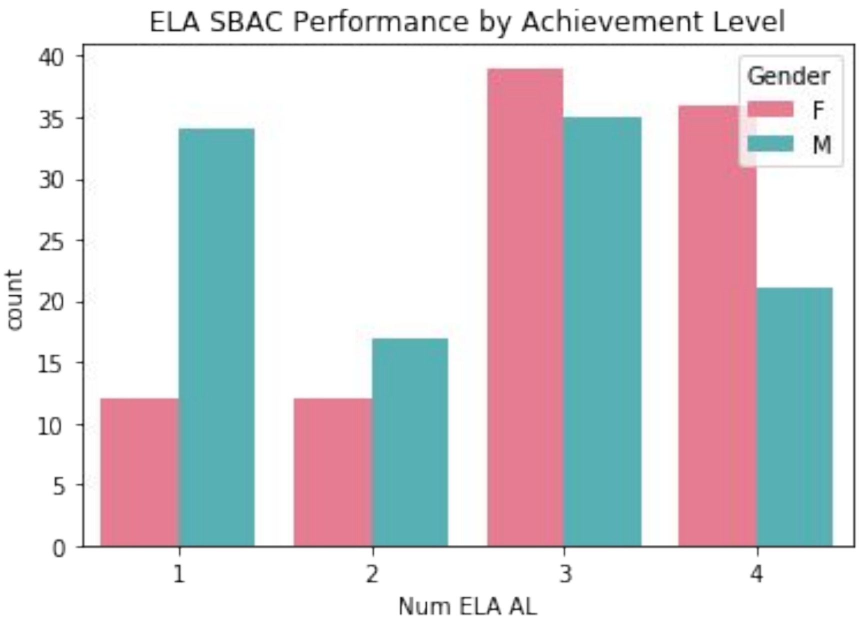
A greater percent of the Latino students are economically disadvantaged than Asian students.

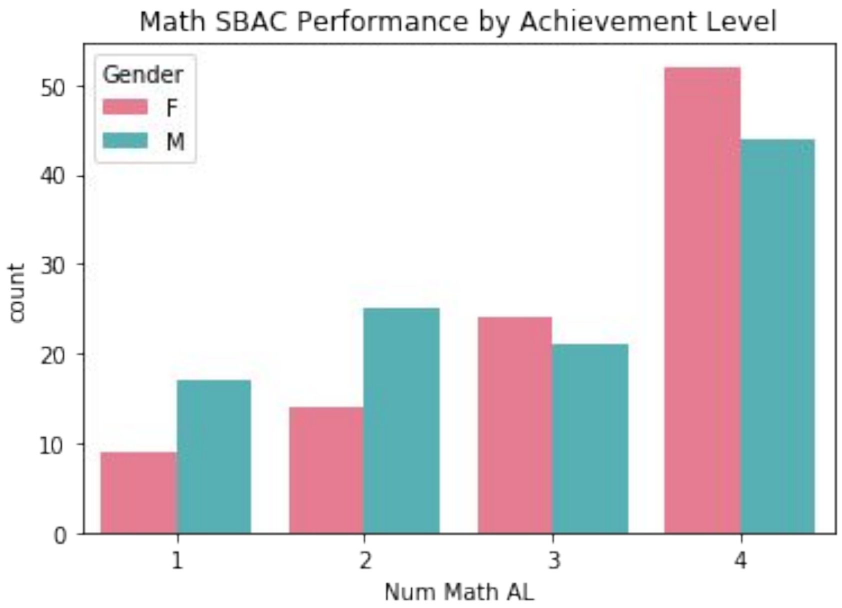
There were also disparities in performance by gender





Girls outperform boys in both ELA & Math

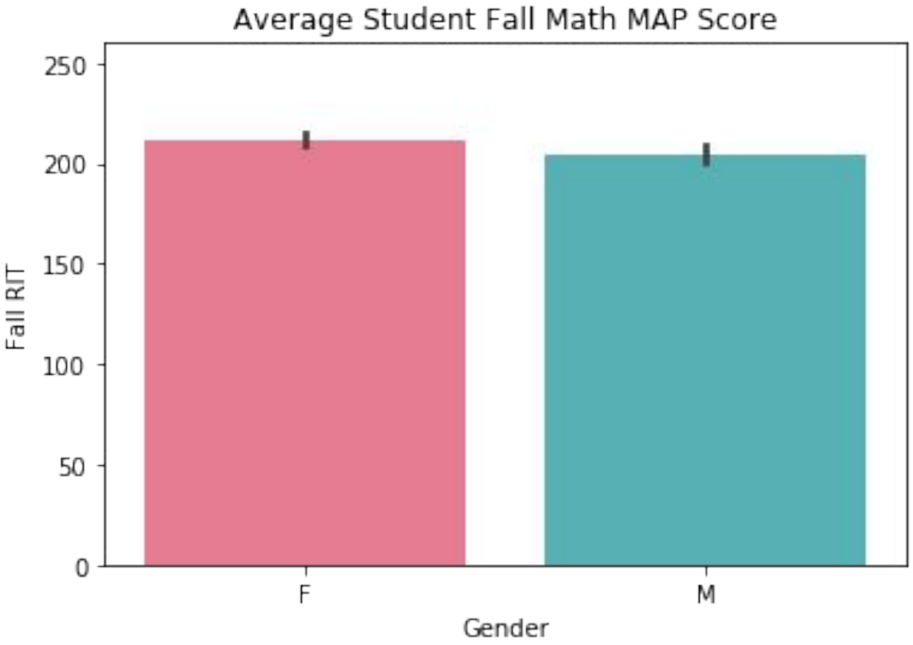


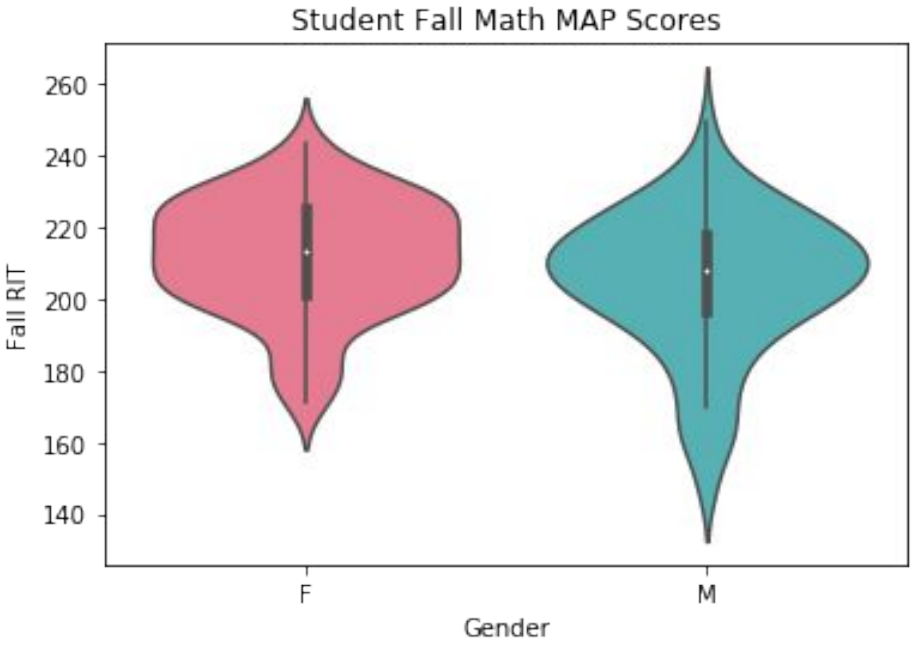


Achievement Levels

1 = Standard Not Met 2 = Standard Nearly Met 3 = Standard Met 4 = Standard Exceeded

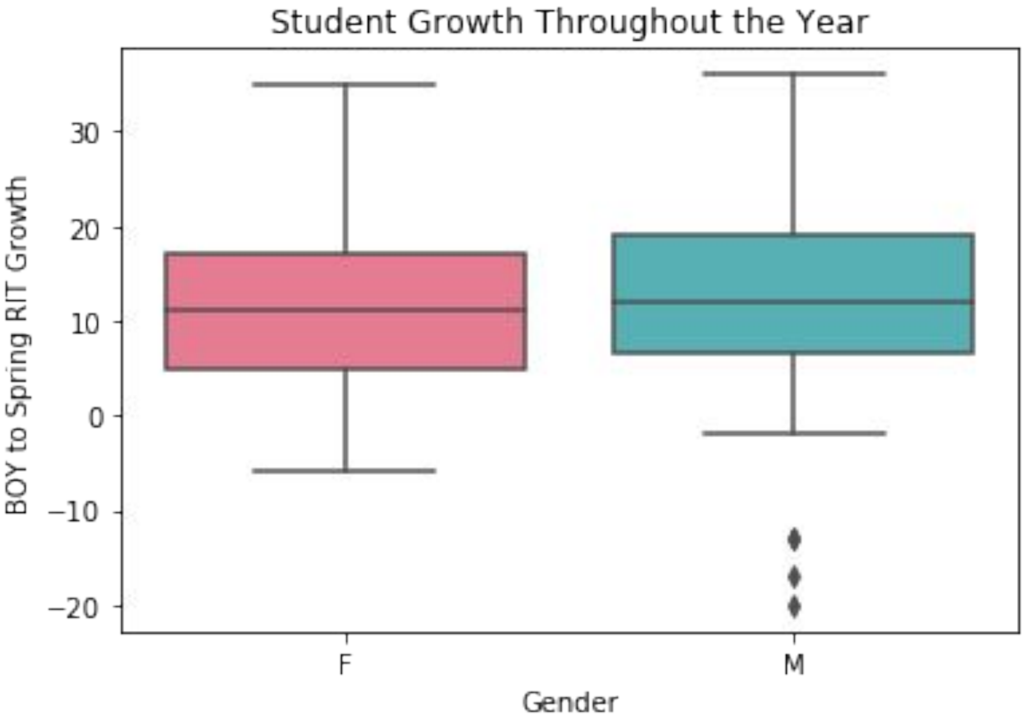
Where did kids start at the beginning of the year?



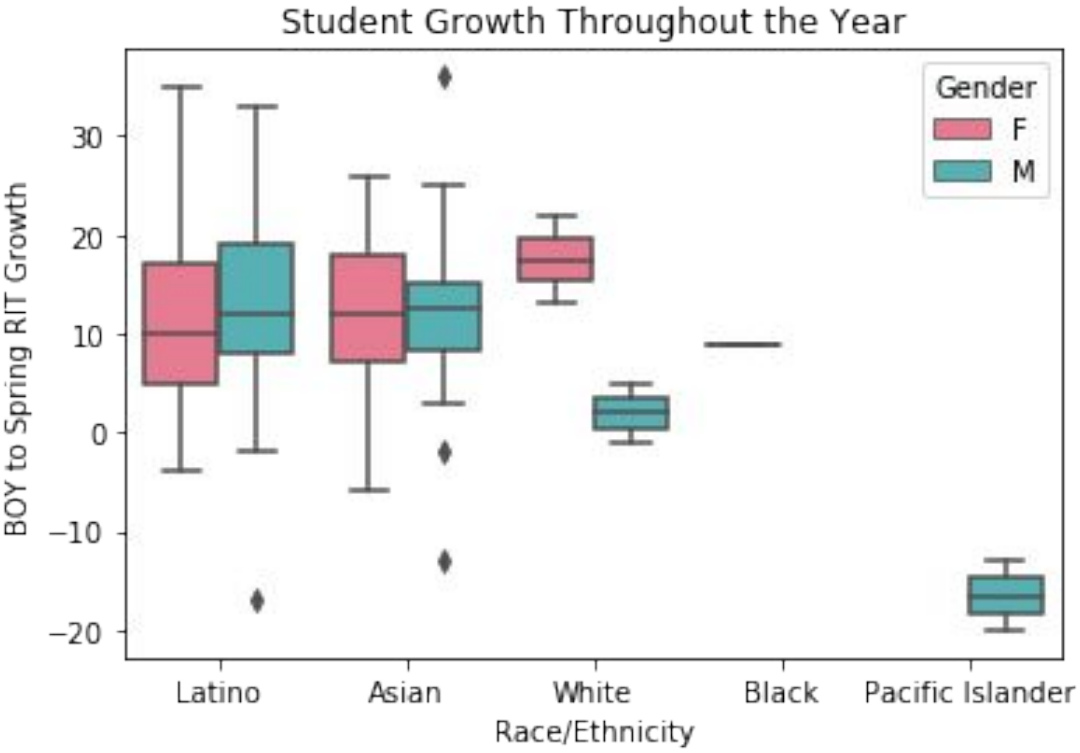


Average Fall MAP Score
Females: 212 **Males: 205**

On average, boys grew slightly more than girls



Latino and Asian boys showed the most growth



Conclusions

Areas for Improvement

- The school needs to focus on better-supporting male students

- More supports need to be put in place to support the English Language development of LEP students

- More data is needed to determine the reasons why boys are not performing as well as girls and why Latino students are not performing as well as Asian students

Other Lessons Learned

- Data cleaning takes a long time, but is super important
- Organization of the Jupyter notebook is KEY
- Have a plan - it is easy to get lost in the analysis and plots