Esther Wu

Introduction to Data Science

Professor Pascal Wallisch

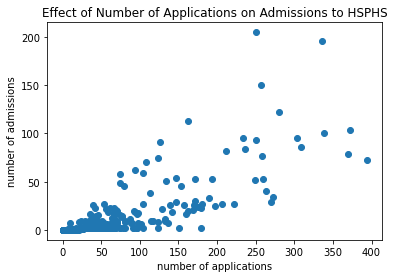
December 22, 2020

**Data Science Final Project**

Before answering these questions, I cleaned up the given dataset by removing the systematically missing data in the columns "Per-student spending, in $" and "Average class size", which are missing for charter schools. This will be the dataset I will be analyzing, unless mentioned otherwise.

**1) What is the correlation between the number of applications and admissions to HSPHS?**

* The correlation between the number of applications and admissions to HSPHS, is about 0.8. I first separated the data into x and y groups where the x group only contained the number of applications while the y group contained the number of admissions. I then visualized the relationship between these two variables, using matplotlib, to get a general idea (positive or negative relationship). Seeing that there was a general linear relationship, I used scipy.stats to find Pearson’s correlation coefficient (r), which is 0.8017265370719306 ≈ 0.8, showing that there is a relatively strong positive relationship between the number of applications and admissions to HSPHS.



**2) What is a better predictor of admission to HSPHS? Raw number of applications or application \*rate\*?**

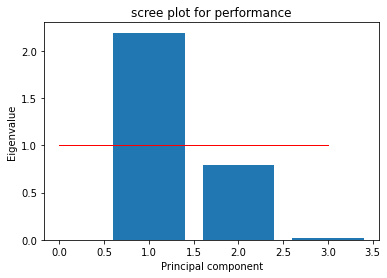
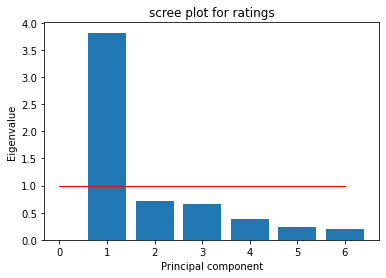
* Considering that each applicant from each school is judged equally, the raw number of applicants is the better predictor. I decided to compare the correlation between the raw number of applications and the number of admissions with the correlation between the application rate and the number of admissions. I created another column that contains the application rate of each row (number of applications/school size). Since there are Nan values in this column, I filled all the Nan with the mean of the column because the correlation, which is what I wanted to calculate from this column, doesn’t rely on each specific value, but rather the general idea. I calculated Pearson’s correlation coefficient for the relationship of application rate and the number of admissions and got 0.658287863431106 ≈ 0.7, meaning that the correlation for the raw number of applications is higher, suggesting that it is the better predictor of admission to HSPHS. This makes sense because the more students that apply from a certain school, the more likely the school will have students accepted to the HSPHS, meaning that the total number of students doesn’t really matter.

**3) Which school has the best \*per student\* odds of sending someone to HSPHS?**

* The school that has the best odds of sending someone to HSPHS is THE CHRISTA MCAULIFFE SCHOOL\I.S. 187. In order to find the best odds, I created another column in the data set that has the ratio of the number of acceptances to the number of not accepted. I then found the school that had the largest ratio, which was I.S 187, suggesting that this school has the best per student odds.

**4) Is there a relationship between how students perceive their school (as reported in columns L-Q) and how the school performs on objective measures of achievement (as noted in columns V-X).**

* First, I had to apply the PCA dimension reduction technique on the dataset of how students perceive their school (6 school climate variables) and the data set of the objective measures of achievement (3 objective achievement variables). After seeing the scree plots, we can see that there is one main factor, determined with the Kaiser criterion. Looking at the loading matrix, I noticed the most representative factor that affects how the school was perceived is how collaborative the teachers are. For the objective measures of achievement, the most representative factor is student achievement. In order to find whether or not there is a relationship between these two variables, I found the correlation between the collaborative teacher ratings and student achievement. The r-value is -0.3530, showing that there is a small negative relationship, where if the ratings are lower, the student achievement is higher.

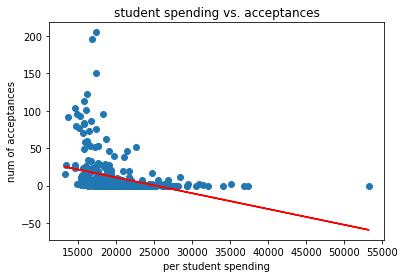


**5) Test a hypothesis of your choice as to which kind of school (e.g. small schools vs. large schools or charter schools vs. not (or any other classification, such as rich vs. poor school)) performs differently than another kind either on some dependent measure, e.g. objective measures of achievement or admission to HSPHS (pick one).**

* I decided to test whether the size of the school affects the number of students that get accepted into an HSPHS. My alternative hypothesis is that the size of the school does affect the number of admissions and my null hypothesis is that the size of the school doesn’t have an effect on the number of admissions. After graphing a histogram of the number of admissions, I noticed that it wasn’t normally distributed so I decided to do a two-sampled Kolmogorov-Smirnov-test between the number of admissions from a large school sample and a small school sample. First, I dropped all the rows that contained Nan values in a dataset that only contained the acceptance and school size columns. I, then, created two samples- small schools and large schools- which I determined by comparing the size to the mean population size. Finally, I used scipy.stats to run a KS-test for the means of the two groups’ acceptance numbers. From this, I found out that the p-value is 1.7763568394002505e-15, much less than the cutoff of 0.05, showing that we can reject the null hypothesis. This indicates that the school size has an effect on the number of acceptances to an HSPHS.

**6) Is there any evidence that the availability of material resources (e.g. per student spending or class size) impacts objective measures of achievement or admission to HSPHS?**

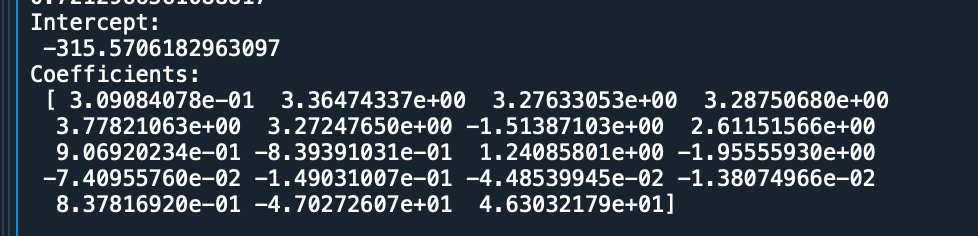
* There is evidence that the availability of material resources impacts the objective measures of achievement. I chose to look at the effect of per-student spending on the number of admissions to HSPHS. First, I graphed these points and the best-fit-line to see the general relationship they had. Then, I found the correlation between these variables, which was -0.342782, showing that there is a general negative relationship. This shows that as the per-student spending increases, the number of admissions to HSPHS decreases. However, since the correlation is rather small, there is a small impact.

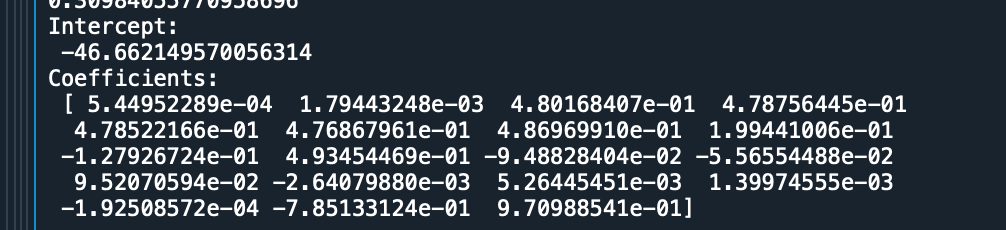


**7) What proportion of schools accounts for 90% of all students accepted to HSPHS?**

* About 20% or 123/ 593 of the schools account for 90% of all students that were accepted to HSPHS. In order to find this proportion, I needed to find the schools that contained the total number of acceptances \* 0.9. After sorting the acceptance column in descending order, I created a while loop that added each school and the number of acceptances they had, to a list to see how many schools it took to cover 90% of the acceptances. From the while loop, it showed that 123 schools had a sum of 4016 acceptances, which is near the 90% mark of 4014.9.

**8) Build a model of your choice – clustering, classification, or prediction – that includes all factors – as to what school characteristics are most important in terms of a) sending students to HSPHS, b) achieving high scores on objective measures of achievement?**

1. In order to find the most important characteristics in terms of acceptances to HSPHS, I first performed a PCA on the dataset so I can focus on the most important characteristics. Then I made a multiple regression model, in order to find out which out of the limited characteristics had the largest effect on the number of acceptances. By looking at the beta values/ coefficients, the largest ones showed me the most impactful variables, which is diversity and inclusion. 
2. I did the same thing as I mentioned above in order to find the important characteristics in terms of achievement. This revealed that just like before, diversity and inclusion are essential, as well as student achievement.



**9) Write an overall summary of your findings – what school characteristics seem to be most relevant in determining the acceptance of their students to HSPHS?**

* By analyzing this dataset, I was able to see that the school characteristics that seemed to be the most relevant in determining the acceptance of their students to HSPHS are: diversity, inclusion, number of applications, and student achievement. Through our analysis of the dataset, we can see that these are the factors that have a positive relationship with the number of acceptances. However, there are factors that have a smaller relationship with the number of acceptances, such as per-student spending and teacher’s collaborative rating. This suggests that as the spending and ratings go up, the number of acceptances go down; in contrast, as diversity, inclusivity, and student achievement increases, schools will see an increase in student acceptance to HSPHS.

**10) Imagine that you are working for the New York City Department of Education as a data scientist (like one of my former students). What actionable recommendations would you make on how to improve schools so that they a) send more students to HSPHS and b) improve objective measures or achievement.**

* In order to send more students to HSPHS, schools need to make their students feel more included and increase diversity as it shows that the number of acceptances increases as these factors increase. Schools with better student achievements and applications also have a higher chance of sending more students to HSPHS. Also shown with the data, schools that have lower per-student spending also have higher acceptance rates, so a decrease in student spending will also improve the school.
* In order to improve objective measures or achievement in general, schools should place their attention on having a more diverse population, which promotes inclusivity. From the data, it showed that if ratings for collaborative teachers are lower, it also promotes higher student achievement. So, if teachers are less collaborative, objective measures of achievement seem to be higher.

APPENDIX

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from scipy.stats import pearsonr

from sklearn.decomposition import PCA

from scipy import stats

from sklearn import linear\_model

import statsmodels.api as sm

data = pd.read\_csv("middleSchoolData.csv", delimiter = ",")

# created two datasets - one with all systematically missing rows are removed

# columns about "Per student spending, in $" and "Average class size"

# are missing for charter schools

# columns are dropped

df\_noEF = data.drop(["per\_pupil\_spending", "avg\_class\_size"], axis = 1)

# new\_df = df\_noEF.fillna(df\_noEF.mean())

'''

"""

# dropping rows that have nan data

# this drops the charter schools, which is fine because I will just analyze

# the relationships of the variables of public high schools

# charter schools were missing info about student spending and average class size

# which I thought was important to study when determing HSPHS enrollment

df = data.dropna(axis = 0)

print(df) # <-- df has no Nan and only public schools

"""

#1 correlation between the number of applications(2) and admissions to HSPHS(3)

x = df\_noEF.iloc[:, 2] # applications

y = df\_noEF.iloc[:, 3] # number of admissions

plt.plot(x, y, "o")

plt.xlabel("number of applications")

plt.ylabel("number of admissions")

plt.title("Effect of Number of Applications on Admissions to HSPHS")

r, p = pearsonr(x, y)

#2 Raw number of applications vs application \*rate\*

df\_noEF["application rate"] = df\_noEF["applications"] / df\_noEF["school\_size"]

x\_rate = df\_noEF.iloc[:, 22]

# fill the na with mean of column

new\_x = x\_rate.fillna(x\_rate.mean())

plt.plot(new\_x, y, "o", color = "red")

plt.xlabel("application rate")

plt.ylabel("number of admissions")

plt.title("Effect of Application Rate and Number of Apps on Admissions to HSPHS")

r\_rate, p\_rate = pearsonr(new\_x, y)

'''

#%%3 best \*per student\* odds of sending someone to HSPHS

df\_noEF["student odd"] = df\_noEF["acceptances"] / (df\_noEF["applications"] - df\_noEF["acceptances"])

max\_rate = df\_noEF[df\_noEF["student odd"] == df\_noEF["student odd"].max()]

# %%4 relationship between how students perceive their school and how the school performs on objective measures of achievement

# PCA to see which factors of each variable matter most

# PCA of how students perceive their school

# since there is systematically missing data, removing rows that have Nan values

# removed together because we need to have the same size data in order to find correlation

ratings\_perform = df\_noEF.iloc[:, 9:22]

new\_rp = ratings\_perform.fillna(ratings\_perform.mean())

# PCA on ratings dataset

ratings = new\_rp.iloc[:, 0:6]

# show correlation between each variable, not that many highly correlated variables

r = np.corrcoef(ratings, rowvar = False)

#plt.imshow(r)

#plt.colorbar()

# create normally distributed data

z\_ratings = stats.zscore(ratings)

pca = PCA()

pca.fit(z\_ratings)

eig\_vals = pca.explained\_variance\_

loadings = pca.components\_

rotated\_data\_1 = pca.fit\_transform(z\_ratings)

covar\_explained = eig\_vals / sum(eig\_vals) \* 100

num\_ratings = len(eig\_vals)

"""

plt.bar(np.linspace(1, num\_ratings, num\_ratings), eig\_vals)

plt.title("scree plot for ratings")

plt.xlabel('Principal component')

plt.ylabel('Eigenvalue')

plt.plot([0,num\_ratings],[1,1],color='red',linewidth=1) # Kaiser criterion line

plt.bar(np.linspace(1, num\_ratings, num\_ratings), loadings[:,0]) #2 (collaborative teachers)

plt.xlabel('Question')

plt.ylabel('Loading')

"""

# draw PCA plot

perform = new\_rp.iloc[:, 10:13]

z\_perform = stats.zscore(perform)

pca = PCA()

pca.fit(z\_perform)

eig\_vals = pca.explained\_variance\_

loadings = pca.components\_

rotated\_data\_2 = pca.fit\_transform(z\_perform)

covar\_explained = eig\_vals / sum(eig\_vals) \* 100

num\_perform = len(eig\_vals)

plt.bar(np.linspace(1, num\_perform, num\_perform), eig\_vals)

plt.title("scree plot for performance")

plt.xlabel('Principal component')

plt.ylabel('Eigenvalue')

plt.plot([0,num\_perform],[1,1],color='red',linewidth=1) # Kaiser criterion line

plt.bar(np.linspace(1, num\_perform, num\_perform), loadings[:,0]) #1 (student achievement)

plt.xlabel('Question')

plt.ylabel('Loading')

rotated\_rating = rotated\_data\_1[:, 0]

rotated\_achieve = rotated\_data\_2[:, 0]

new\_corr = np.corrcoef(rotated\_rating, rotated\_achieve)

#new\_ratings = ratings["collaborative\_teachers"]

#new\_perform = perform["student\_achievement"]

#relation = np.corrcoef(new\_ratings, new\_perform)

#corr, \_ = pearsonr(new\_ratings, new\_perform)

# %% 5 hypothesis small vs large school admission to HSPHS

# null: The size of the school doesn't have an effect on the number of admissions to HSPHS

# alternate: The size of the school does effect the number of admissions

# ks\_test

new\_df = df\_noEF[["acceptances", "school\_size"]].dropna()

size\_mean = new\_df["school\_size"].mean()

small\_school = new\_df[new\_df["school\_size"] <= size\_mean]

large\_school = new\_df[new\_df["school\_size"] > size\_mean]

hist = plt.hist(new\_df["acceptances"])

x = small\_school["acceptances"]

y = large\_school["acceptances"]

KS, p = stats.ks\_2samp(x, y)

#%% 6 availability of material resources (e.g. per student spending or class size) impacts objective measures of achievement or admission to HSPHS

import pandas as pd

data = pd.read\_csv("middleSchoolData.csv", delimiter = ",")

df = data.dropna(axis = 0)

x = df["per\_pupil\_spending"]

y = df["acceptances"]

m, b = np.polyfit(x, y, 1)

plt.plot(x, y, "o")

plt.plot(x, m \* x + b, color = "red")

plt.xlabel("per student spending")

plt.ylabel("num of acceptances")

plt.title("student spending vs. acceptances")

spending\_corr = np.corrcoef(x, y)

# %% 7 proportion of schools accounts for 90% of all students accepted to HSPHS

total\_acceptances = np.array(df\_noEF["acceptances"])

acceptance\_sum = np.sum(total\_acceptances)

sorted\_acceptance = np.array(df\_noEF.sort\_values("acceptances", ascending = False))

percent = acceptance\_sum \* 0.9

counter = 0

position = 0

schools = []

while counter < percent:

counter += sorted\_acceptance[position][3]

schools.append(sorted\_acceptance[position][1])

position += 1

print(percent)

print(counter)

print(len(schools))

print(position)

# %%8 multiple regression

import statsmodels.api as sm

# a - sending students to HSPHS

# PCA to see most important factors

df\_acceptances = df\_noEF.dropna(axis = 0)

#applications = df\_acceptances.iloc[:, 2]

factors = df\_acceptances.iloc[:, 2:22]

#complete\_factors = pd.concat([applications, factors], axis = 1)

z\_accept\_factors = stats.zscore(factors)

pca = PCA()

pca.fit(factors)

e\_factors = pca.explained\_variance\_

l\_factors = pca.components\_

rotated\_data\_factor = pca.fit\_transform(factors)

covar\_explained = eig\_vals / sum(eig\_vals) \* 100

num\_factors = len(e\_factors)

"""

plt.bar(np.linspace(1, num\_factors, num\_factors), e\_factors)

plt.title("scree plot for factors")

plt.xlabel('Principal component')

plt.ylabel('Eigenvalue')

plt.plot([0,num\_factors],[1,1],color='red',linewidth=1) # Kaiser criterion line

plt.bar(np.linspace(1, num\_factors, num\_factors), l\_factors[:,2]) #1,8 (application, collaborative teacher)

plt.xlabel('Question')

plt.ylabel('Loading')

"""

factor\_arr = np.array(df\_acceptances)

A = np.transpose([factor\_arr[:,2],factor\_arr[:,4],factor\_arr[:,5],

factor\_arr[:,6],factor\_arr[:,7],factor\_arr[:,8],

factor\_arr[:,9], factor\_arr[:,10],factor\_arr[:,11],

factor\_arr[:,12],factor\_arr[:,13],factor\_arr[:,14],

factor\_arr[:,15],factor\_arr[:,16],factor\_arr[:,17],

factor\_arr[:,18],factor\_arr[:,19],factor\_arr[:,20],

factor\_arr[:,21]]) # 21

B = np.transpose([factor\_arr[:,2],factor\_arr[:,3],factor\_arr[:,4],

factor\_arr[:,5],factor\_arr[:,6],factor\_arr[:,7],

factor\_arr[:,8],factor\_arr[:,9], factor\_arr[:,10],

factor\_arr[:,11],factor\_arr[:,12],factor\_arr[:,13],

factor\_arr[:,14],factor\_arr[:,15],factor\_arr[:,16],

factor\_arr[:,17],factor\_arr[:,18],factor\_arr[:,20],

factor\_arr[:,21]]) # 21

Y = df\_acceptances["acceptances"] # acceptances 11(student achievement), 19 (supportive\_environment)

Z = df\_acceptances["student\_achievement"]

regr = linear\_model.LinearRegression() # linearRegression function from linear\_model

regr.fit(B,Z) # fit model

print(regr.score(B,Z)) # r^2

print('Intercept: \n', regr.intercept\_)

print('Coefficients: \n', regr.coef\_) #beta, larger = more important