Final Report for the Capstone Project

Siyu Chen

December 18, 2023

1 Project Outline

1.1 Motivation

My plan for this final project is to help high school students to choose their colleges. Choosing a college can be a daunting task for many students, as there are numerous factors to consider on the institution's side such as location, cost, academic programs, and campus culture, and the student's side such as academic performance, extracurricular activities, and personal preferences.

This project aims to simplify this process by building a recommender system that takes into account a student's performance data and college application results to predict the probability of admission to certain colleges and the future success of the student at those colleges. By providing students with a list of colleges that are likely to be a good fit for them, this project can help students make more informed decisions about their college choices, potentially leading to greater academic success and career opportunities in the future.

1.2 Project Target

I will build a recommender system based on the student's profiles such as the SAT scores, ethnicity, gender, education budget, and so on. Based on the student's profiles, the recommender system will consider the following factors to recommend a college to the student:

- What will be the acceptance rate of this college for this student?
- What will be the best fit for this student given his/her profiles and preferences, e.g., location, future earnings?

The recommender system will output a college ranking that is most likely to be a good fit for the student.

1.3 Data Source

The datasets I'm using in this project include the following:

- College Scorecard released by US Dept of Education (`https://collegescorecard.ed.gov/`). The dataset features 7,000+ colleges and universities in the US and includes information on the institution's location, size, admission rate, average salary after graduation, Post-College Professional Success and so on. The dataset will be used to predict the future income and match the student's preference on the urbanization of the college.

- IPEDS data released by National Center for Education Statistics (https://nces.ed.gov/ipeds/use-the-data). The dataset features the 25% and 75% percentile of test scores of students enrolled in each college. The dataset will be used to predict the acceptance rate. Some other features are also included in this dataset, including percentage of students receiving financial aid, average cost and tuition fees, percent of different races, and so on.

2 Modeling the Admission Rate

I model the acceptance likelihood as a function of the test score (including both the SAT and ACT scores, but without the high school GPA) and the student's gender. We use a product model for the likelihood as the following:

$$\ell(s, r, g, S, C) = \ell_{score}(s) \cdot \ell_{gender}(g)$$

where s is the test score vector, g represents the gender. Since there is only information on the enrolled students' (posterior) rather than the applicants' (prior) features such as percentage of each race, I do not model other factors' influence on the admission likelihood. I illustrate the model in the following.

Test Score Likelihood Model. We model the test score prior as $s \sim p_{score} = \mathcal{N}(\mu, \Sigma)$, where we assume Σ to be diagonal (This assumption is a little bit strong. But as our dataset only contains the 25 and 75 quantile of each individual score, it is hard to extract the correlation between these scores in the variance. Note that we still allow their means to be related.) The distribution of the test score among the admitted students depends on two factors: the school's admission likelihood $f_{school}(s)$ based on the test scores and the student's likelihood of accepting this school's offer $f_{student}(s)$ given his/her test scores. Note that the school's admission likelihood $f_{school}(s)$ should be monotonically increasing with respect to the test scores while the student's preference $f_{student}(s)$ could be monotonically decreasing with respect to the test scores. The rationality behind this is that schools tend to admit students with higher test scores while students with high test scores seldom consider schools with low rankings, which is often associated with low admission standards.

Therefore, we can model enrollment likelihood as $f_{enroll}(s) = f_{school}(s) f_{student}(s)$. By the previous discussion, we can model $f(s)$ as another Gaussian distribution $\mathcal{N}(\mu_1, \Sigma_1)$, where Σ_1 is also diagonal. Therefore, the posterior distribution of scores upon acceptance is just

$$q(s \,|\, a = 1) \propto f(s)\ell_{score}(s) \sim \mathcal{N}\left(\underbrace{(\Sigma^{-1} + \Sigma_1^{-1})^{-1}(\Sigma^{-1}\mu + \Sigma_1^{-1}\mu_1)}_{\mu_2}, \quad \underbrace{(\Sigma^{-1} + \Sigma_1^{-1})^{-1}}_{\Sigma_2} \right),$$

where $a \in \{0, 1\}$ is the indicator for admission. Entrywise, we have

$$\mu_2^i = \frac{(\sigma_1^{(i)})^2 \mu^i + (\sigma^{(i)})^2 \mu_1^i}{(\sigma_1^{(i)})^2 + (\sigma^{(i)})^2}, \quad (\sigma_2^{(i)})^2 = \frac{(\sigma_1^{(i)})^2 \cdot (\sigma^{(i)})^2}{(\sigma_1^{(i)})^2 + (\sigma^{(i)})^2}. \tag{2.1}$$

eq: posterior e

Therefore, we can estimate the posterior Gaussian, and solve for the admission model (μ_1, Σ_1). The remaining question is how to obtain the mean and variance of the prior. Note that the mean and variance are biased in the data since the test scores are collected only for enrolled students here.

However, these websites (https://blog.prepscholar.com/sat-historical-percentiles-for-2014-2013-201 and (https://www.act.org/content/dam/act/unsecured/documents/Natl-Scores-2013-National2013.pdf) come to our rescue as it collectes summary for SAT scores in 2013. We summarize the results on the following table

	mean	standard deviation
SAT Critical Reading	496	115
SAT Mathematics	514	118
SAT Writing	488	114
ACT Composite	20.9	5.4

Note that the estimated standard deviation $\sigma_2^{(i)}$ could potentially be larger than the prior, which leads to no reasonable solution according to (2.1). This could happen when our model does not capture what is happening in the real world, for instance, when the acceptance likelihood is far from a Gaussian distribution. However, as indicated by the experimental results, such an event is very rare in the data (8 in 468 schools). In that case, I still estimate the variance according to (2.1) but set the "nan" and negative values to infinity. The reason is that when a school' admission rate on a spefic score violates the model, a natural thing to do is ignoring the dependency of the admission rate on the violated score, which is achieved by using an extremely flat normal.

We then model the likelihood by the following function "truncated dnorm"

$$\ell_{score}(s) = \prod_i \exp\left(-\frac{(\mathrm{ReLU}(\mu_1^{(i)} - s^{(i)}))^2}{2(\sigma_1^i)^2}\right), \tag{2.2}$$

eq: truncated c

where $\mathrm{ReLU}(x) = \max\{0, x\}$ is the rectified linear unit function, and we allow $\sigma_1^{(i)}$ to be $+\infty$. The intuition for (2.2) is that following what I have assumed $\ell_{score}(s) = f_{school}(s) f_{student}(s)$ and the fact schools always tend to choose students with higher scores, where students tend to choose better schools (with higher admission standards), we can model $f_{school}(s)$ and $f_{student}(s)$ as two "half" Gaussian, i.e.

$$f_{school}(s) = \exp\left(-\frac{(\mathrm{ReLU}(\mu_1^{(i)} - s^{(i)}))^2}{2(\sigma_1^i)^2}\right),$$

$$f_{student}(s) = \exp\left(-\frac{(\mathrm{ReLU}(s^{(i)} - \mu_1^{(i)}))^2}{2(\sigma_1^i)^2}\right).$$

The product of these two functions then gives us the Gaussian likelihood $f_{enroll}(s)$ of enrollment.

Likelihood Model for Gender. We next characterize the influence of gender. We assume a priori that the likelihood of a applicant being male or female is equally likely, i.e., the gender prior is 0.5 for both male and female applications (which is close to the truth but may not be accurate in reality). Suppose that gender is weakly coupled with academy performance. We model the bias between gender as a likelihood ratio, i.e.,

$$\ell_{gender}(g) = \frac{\mathrm{percentage}(g)}{\max\{\mathrm{percentage}(g), 1 - \mathrm{percentage}(g)\}},$$

Note that when there is no bias, the gender does not influence the admission rate. When bias exists, ℓ_{gender} models the "disadvantage" of a specific gender in the admission process.

3 Bayesian Factor

In the analysis, we take an example by using the student's family income to illustrate the computation of Bayes Factors. Let $E = \{\text{student's family-income} > x\}$ be the event of interest. Let H denote the school of interest and Ω denote all the schools. Let's define $p(H)$ as the prior probability of enrollment at school H, namely the proportion of students enrolled in school H among all the students enrolled in Ω. The Bayesian Factor $\mathrm{BF}(E, H)$ is then defined as the likelihood ratio as the following:

$$\mathrm{BF}(E, H) = \log \frac{p(E \mid H)}{p(E \mid \Omega)}.$$

A simple application of the Bayesian rule gives

$$\mathrm{BF}(E, H) = \log \frac{p(E \mid H)}{p(E \mid \Omega)} = \log \frac{p(H \mid E)}{p(H)} = \log \frac{p(H, E)}{p(H)p(E)}.$$

Note that the Bayes Factor reflects the "advantage" of school H over the other schools in terms of the event E. I will include the Bayes Factor in the analysis with some manually selected coefficients to determine the score of each school. I calculate the Bayesian Factor for each school under features "ethnicity" (funtion "predict_BF_ethnicity"), "finacial aid" (function "predict_BF_financial_aid"), and "earnings in 6 years after a student enrolls in college" (function "predict_BF_earning"). Users can select the strength of the Bayesian Factor by adjusting the coefficients in the shiny app.

4 Data Cleaning

4.1 School Selection

We first aim to clean the data "IPEDS_data.xlsx" and identify the schools of interest. The data is from 2013 which is a little bit old. More recent data does not contain some of the test scores that we need. The implemented code filters out the schools based on the following conditions:

1. The school is not included in the list of schools in the file `data/CSV_11292023-550.csv`, where the file is downloaded from https://nces.ed.gov/ipeds/use-the-data for a list of major universities with undergraduates in the US.

2. The school has less than 50% of freshmen submitting SAT scores or ACT scores.

The filtered data is then stored in the file `data/IPEDS_with_scores.csv`.

4.2 Fill in Missing Scores

The missing values in the test scores columns are filled using linear regression. The process is as follows:

1. Load the necessary libraries and data. The data is loaded from the `IPEDS_with_scores.csv` and `IPEDS_for_regression.csv` files, where the latter file contains all the universities that fully report their test scores. The columns to be filled are

- SAT Critical Reading 25th percentile score
- SAT Critical Reading 75th percentile score
- SAT Math 25th percentile score
- SAT Math 75th percentile score
- SAT Writing 25th percentile score
- SAT Writing 75th percentile score
- ACT Composite 25th percentile score
- ACT Composite 75th percentile score

2. Iterate over all possible patterns of missingness. For each pattern, create two lists: `missing_columns` and `nonmissing_columns`. The `missing_columns` list contains the names of the columns that have missing values, and the `nonmissing_columns` list contains the names of the columns that don't have missing values. If there are no missing values for the current pattern, skip to the next pattern. If there are missing values, run a linear regression on the non-missing rows to predict the missing values. The `train` function from the `caret` library is used to train the linear regression model. The `predict` function is then used to predict the missing values. Then I store the predicted values in the original data frame at the correct rows and columns.

3. Finally, write the data with the filled values to a new CSV file named `IPEDS_filled.csv`.

4.3 Clean Other Data

Filter the `MERGED2013_PP.csv` data by selecting the universities of interest, and add a column `School_Prob` to the data, where the probability $p(H)$ for each school is calculated by

$$p(H) = \frac{\text{number of students enrolled in school } H}{\text{total number of students enrolled in all schools}}.$$

I also filter the Scorecard data similarly. Note that the year 2011 is chosen because it is the latest year (before 2013) that has survey data for the earnings after 6 years of enrollment. The Scorecard data is not included in the submission as it is too large. The full data can be downloaded from https://collegescorecard.ed.gov/.

Shiny App Design

The source code for the Shiny app is in the `shiny_app.R` file.

User Interface (UI) Design

The Shiny app features a user-friendly interface with the following main components:

- **Test Scores Input:** Numeric input fields for SAT Critical Reading, SAT Math, SAT Writing, and ACT Composite scores.

- **Personal Information Input:** Select inputs for Gender, Ethnicity, State, and a multi-select input for Urbanization.

- **Preference Adjustment:** Slider inputs allow users to adjust preferences for ethnicity strength, financial aid strength, and earning strength.
- **Filtering Criteria:** Slider inputs for minimal and maximal cost of attendance, minimal and maximal admission rate, and a reset button to restore default preferences.
- **Reset Personal Info:** A button to reset personal information inputs.

Functionality

The app functionality involves dynamic computation and ranking of universities based on user preferences. Key functionalities include:

- **Data Loading:** Upon app initiation, data is loaded from CSV files, and relevant datasets are merged.
- **Likelihood Calculation:** Reactive expressions compute the likelihood of admission according to the test scores and gender information provided by the user.
- **Bayesian Factor Calculation:** Bayesian factors for ethnicity, financial aid, and earning are calculated based on user preferences and data.
- **Cost and Admission Rate Filtering:** Universities are filtered based on user-defined criteria for cost, admission rate and urbanization.
- **Ranking:** Universities satisfying user criteria are ranked in descending order of the combined Bayesian factor.

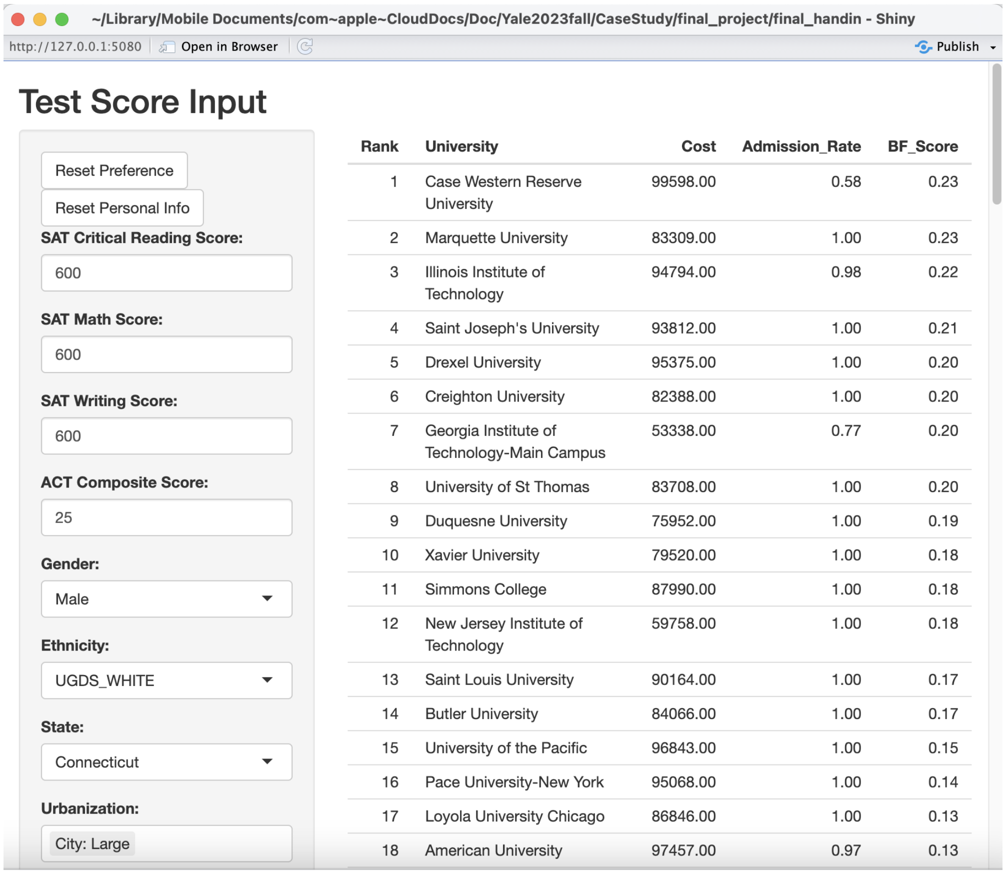
Output Calculation

The output is presented in a ranked table format, displaying universities along with their corresponding ranks, costs, admission rates, and Bayesian scores. The design enables users to make informed decisions based on their preferences and criteria.

The interface for the shiny app looks as the following Here, in Figure 1, we assume there is a white male student with SAT scores 600 for each subject and ACT score 25 who comes from Connecticut. He targets at a school located in a large city with a cost of attendance between $30,000 and $100,000 and with an admission rate larger than 0.5. His main concerns is the future earnings, therefore he sets the strength of the Bayesian Factor for earnings to be the largest. The app then returns a list of candidate universities ranked in descending order of the Bayesian Factor scores.

References

https://collegescorecard.ed.gov/
https://nces.ed.gov/ipeds/use-the-data
https://www.kaggle.com/code/sumithbhongale/notebook4186f97a6b
https://www.kaggle.com/code/kerneler/starter-college-admissions-c79081de-c
https://www.kaggle.com/code/apollostar/which-college-is-best-for-you#student-case-studies
https://www.kaggle.com/code/yatin9045/us-university-selection-analysis/input


~/Library/Mobile Documents/com~apple~CloudDocs/Doc/Yale2023fall/CaseStudy/final_project/final_handin - Shiny
http://127.0.0.1:5080
Open in Browser
Publish
Reset Preference
Reset Personal Info

Test Score Input

SAT Critical Reading Score: 600

SAT Math Score: 600

SAT Writing Score: 600

ACT Composite Score: 25

Gender: Male

Ethnicity: UGDS_WHITE

State: Connecticut

Urbanization: City: Large

Rank	University	Cost	Admission_Rate	BF_Score
1	Case Western Reserve University	99598.00	0.58	0.23
2	Marquette University	83309.00	1.00	0.23
3	Illinois Institute of Technology	94794.00	0.98	0.22
4	Saint Joseph's University	93812.00	1.00	0.21
5	Drexel University	95375.00	1.00	0.20
6	Creighton University	82388.00	1.00	0.20
7	Georgia Institute of Technology-Main Campus	53338.00	0.77	0.20
8	University of St Thomas	83708.00	1.00	0.20
9	Duquesne University	75952.00	1.00	0.19
10	Xavier University	79520.00	1.00	0.18
11	Simmons College	87990.00	1.00	0.18
12	New Jersey Institute of Technology	59758.00	1.00	0.18
13	Saint Louis University	90164.00	1.00	0.17
14	Butler University	84066.00	1.00	0.17
15	University of the Pacific	96843.00	1.00	0.15
16	Pace University-New York	95068.00	1.00	0.14
17	Loyola University Chicago	86846.00	1.00	0.13
18	American University	97457.00	0.97	0.13

Figure 1: The interface for the shiny app

⟨fig:app⟩

https://blog.prepscholar.com/sat-historical-percentiles-for-2014-2013-2012-and-2011
https://www.act.org/content/dam/act/unsecured/documents/Natl-Scores-2013-National2013.pdf