Machine Learning of the College Admissions Process
TJHSST Senior Research Project
2009-2010

Sam Rush

April 7, 2010

Abstract

The goal of this project is to analyze the various biases that exist in the college admissions system by attempting to predict college decisions. This project will attempt to reduce college admissions to pure numbers, excluding data that is inaccessible such as essays and teacher recommendations. Past user-submitted data from the 2007[3], 2008[2], and 2009[1] *Senior Destinations* websites will be used to train an artificial neural network in a process known as machine learning to perform a nonlinear least squares fit. Then, factors such as the gender bias and the race bias will not only be proven to exist but will be quantifiable based on their role in the least squares fit.

Keywords: college admissions, machine learning, neural network, nonlinear least squares, Gauss-Newton

1 Introduction

The college application process has become a hypercompetitive environment in which students embark on a four year process of padding their résuméto look impressive to an admissions officer. College admissions is often publicized as a wholistic process in which admissions officers look at everything without "weighting" certain aspects of your application such as GPA. Therefore students look to excel in all areas instead of taking the most efficient path, which is not immediately obvious. So, how do we determine what's really important to a college? In this paper I attempt to answer that question.

2 Background

2.1 Machine Learning

Machine learning is the programming technique in which a programs behavior is altered according to pre-existing data. A computers ability to learn is its ability to recognize patterns among data sets and apply those patterns to other data. This is essentially data interpolation. In this project, supervised machine learning is used to translate an N-dimensional input vector into a single scalar output.

2.2 Senior Destinations

At TJHSST, it has become a tradition for one person in each class to create something called the Senior Destinations website. This site enables those in each class to submit their information (such as GPA, SAT scores, AP scores, etc.) along with where they applied and what happened to each application. Data still exists from the Senior Destinations sites of 2007[3], 2008[2], and 2009[1] and will be used to train the neural network. I should note that while a signifficant portion of the senior class does participate in this each year, the data is somewhat skewed toward the higher achieving portion of the class, since they are more likely to be enthusiastic about college and the admissions process.

3 Development

The project will consist of two parts: the 2010 Senior Destinations website and a College Analysis website. The first site will be coded from scratch in order to be cleaner and more complete than the previous years sites. The College Analysis website will deal with the machine learning and analysis of college admissions.

3.1 Languages

3.1.1 PHP

PHP: **H**ypertext **P**reprocessor is the main language of this project. The output consists of the standard web elements: Hyper Text Markup Language (HTML), JavaScript, and Cascading Style Sheets (CSS). The websites will run on an Ubuntu Linux machine running the Apache HTTP Server.

3.1.2 MySQL

MySQL is a Structured Query Language that is the storage engine for this project due to its integration into PHP and its use with prior *Senior Destinations* websites.

4 Methodology

4.1 College Analysis Website

To make the College Analysis Website, data will first need to be imported from the Senior Destinations sites and missing data will need to be filled in. For example, the classes of 2007, 2008, and 2009 do not have gender and race data readily available. This data will be manually input by me using pictures from TJHSST yearbooks.

Another discrepancy between the data sets is the GPA. The class of 2010s GPAs are calculated in a different way from the other classes due to a system called FAIRGRADE[4]. Luckily, at TJ, ones FAIRGRADE GPA can be fairly easily predicted from their pre-FAIRGRADE GPA. To come up with a transformation between the two, I took the currently submitted 217 GPAs from the class of 2010 and 217 evenly distributed (by class rank) GPAs from the classes of 2007 and 2008. Then, I plotted the pre-FAIRGRADE GPAs on the X-axis and the FAIRGRADE GPAs on the Y-axis and took the quintic of best fit. This process works on the assumption that the distribution of GPAs is constant from class to class.

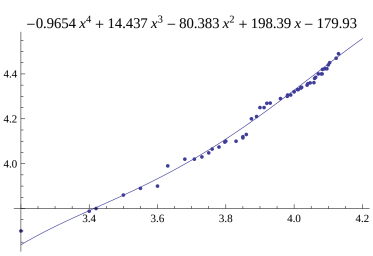

Figure 1: FAIRGRADE GPAs plotted versus Non-FAIRGRADE GPAs with quartic of best fit ($R^2 = .9903$) as the interpolation function

4.2 Least Squares Analysis

The program will perform a least squares analysis in order to find a function $f(x_1, x_2, ..., x_n) = c$ of best fit to the college admissions data with which it is being trained. The least squares approach guarantees us that for our given form, the function f which is found produces the most accurate results. The procedure then goes as follows:

1. Create a matrix A of the form $\begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} & 1 \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} & 1 \end{pmatrix}$ such that $a_{i,j}$ represents the j^{th} factor for the i^{th} person in the training data. The 1 represents the constant term that may need to exist in the predictive algorithm, which essentially relaxes the problem encountered by ill-defined outcomes. Note that this matrix must need not be square.

2. Create a vector B of the form $\begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{pmatrix}$, where d_i is a number representing the decision for the i^{th} student in the training data.

3. Create a vector x of the form $\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$, where x_i is the weight which will be applied to the ith factor at the end. Note that x_i is just a variable. Our goal will now be to solve the inconsistent[1] matrix system $Ax = B$ using the method of a linear least squares fit.

4. Obtain the QR decomposition of A. Essentially, we want to find an orthogonal[2] matrix Q and an upper-triangular[3] matrix R such that $QR = A$. Q can be obtained using the Gram-Schmidt orthogonalization process[6]. R can be obtained by: $A = QR \rightarrow Q^T A = Q^T QR \rightarrow Q^T A = R$.

5. Solve the matrix system $R'x = (Q^T B)'$, where R' is the upper nxn submatrix of R and $(Q^T B)'$ is the upper n rows of vector $Q^T B$. This system can be solved using gaussian elimination[7]. In gaussian elimination, we use an augmented matrix $A|B$ and form an upper triangular matrix A by successive "eliminations" of columns. Partial pivoting is used to avoid ill-conditioning[4] and the case of a 0 on the diagonal, which causes the system to be inconsistent[6]. After an upper-triangular matrix is obtained, x can

[1] A system is inconsistent if it has more linearly independent equations than variables

[2] A matrix is said to be orthogonal if $Q^T Q = I$, where I is the identity.

[3] A matrix is said to be upper-triangular if all entries below the diagonal are 0

[4] Conditioning refers to the extent to which numerical algorithms are subject to rounding errors.

be found by "back-solving" for each component starting from the bottom and then substituting the value of that component into the next equation.

6. The x we have now obtained is the least squares solution to our inconsistent system $Ax = B$. We can now run students through this program simply by multiplying the row vector $\begin{pmatrix} a_1 & a_2 & \dots & a_n & 1 \end{pmatrix}$ for that student by x to obtain a scalar p. Currently, I am not dealing with wait-list decisions, so to obtain the prediction for that student, I simply round p to the nearest integer, which will be 1 (accept) or 0 (reject).

4.2.1 Further Development

The reason that I took this complicated approach to finding a simple linear regression is that I can now very easily expand this method to nonlinear least squares fits with varying forms. Essentially, I can now alter the form of my function f, which is currently $x_1 a_1 + x_2 a_2 + \ldots$ to anything which I want, or in this case what the machine learns is the best without sacrificing significant computing power. By decomposing A into QR, I have ensured that it the least squares computation for any alteration of the form and/or resultant vector is $O(m * n)$, or linear in the number of data I have. This will allow a neural network to train the computer to learn the optimal form of the function f with the assistance of the Gauss-Newton method for nonlinear least squares.

5 Testing

After the predictive algorithm has been trained by the past admissions' data, the TJHSST class of 2010's application data will be run through the program and the computer will output its predictions of each result. The predictions will be compared with the actual results for accuracy. Then, the algorithm will be retrained with the inclusion of the 2010 data. With four years of data, I can then begin to investigate biases for each individual college. The gender weight, which I will call G, is the node in the neural network that will quantify the gender bias of the system. That is, if the weight is positive, males are more likely to be accepted than females and vice versa. Similarly, there will be weights for each race which will quantify those biases as well.

6 Expected Results

I expect that when introduced to a nonlinear prediction system, the program will be able to predict upwards of 80% of applications for most schools with copious data. Note: I will count a waitlist decision as half of an acceptance plus half of a rejection for these purposes.

7 Current Results

Currently, the computer does a decent job at predicting admissions based only on GPA, SAT scores, and Gender with only a linear regression. Below is a table of prediction rates for a small sample of the class of 2010's applications.

College	# Correct	Out of	Prediction Rate
Brown University	16	22	72.7%
Cornell University	28	37	75.7%
Duke University	16	20	80%
University of Pennsylvania	16	21	76.1%
University of Virginia	78	83	93.9%
Virginia Tech	40	40	100%

Table 1: A sample of prediction success for various colleges.

To illustrate the regression that the machine currently uses, I have included graphs with only SAT and GPA (obviously with a 3rd parameter, we would not have enough physical dimensions to view the graph) below.

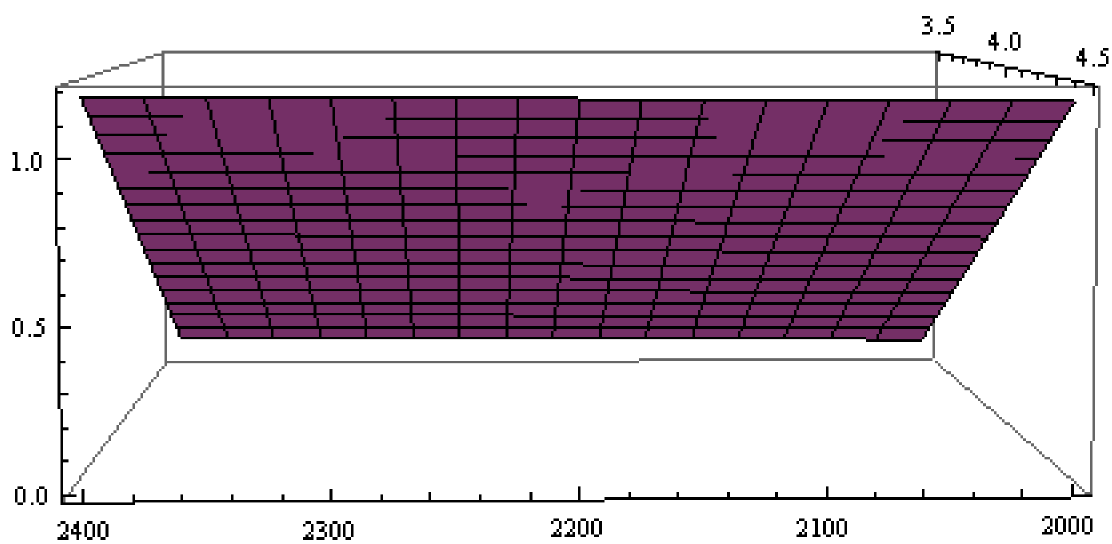

Figure 2: SAT vs. GPA vs. Prediction for the University of Virginia

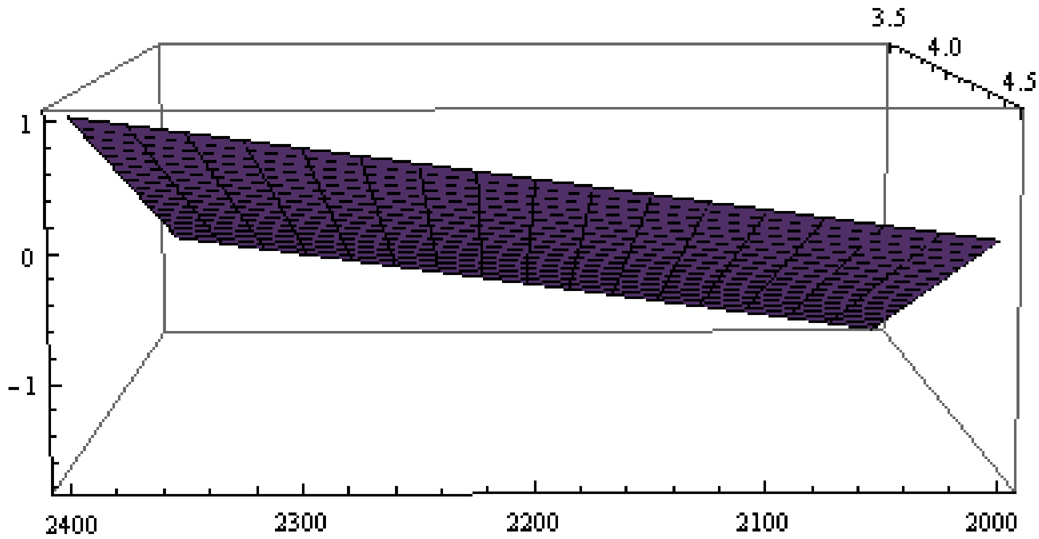

Figure 3: SAT vs. GPA vs. Prediction for the University of Pennsylvania

7.1 Discussion

The varying prediction rates in Table 1 can be attributed to the current constraint that the regression be linear. The machine will next learn the best form of the regression instead of being restricted. The scattergram for Brown University in 2010 appears wildly random, yet the machine can pick out the correct people with decent accuracy.

The two graphs illustrate the different methodologies that these two institutions use to select their students. UVA's graph has a steep slope in the GPA direction and an almost unnoticable slope in the SAT direction, indicating that it cares a lot more about your GPA than your SAT. Penn's graph, on the other hand, has a much larger slope in the SAT direction, but still a greater slope in the GPA direction, indicating that your SAT will be a determining factor in your application, but not as much as your GPA will. Also notice the scales on the axes for the two graphs. A much larger percentage of the UVA graph is in the "accept area" (greater than 0.5 on the z-axis) than the Upenn graph. This should make sense, as it UPenn is generally harder to get into than UVA.

8 Final Results

There will not be any final results until the conclusion of the course. Otherwise, the results would, by definition, not be final.

References

[1] Thiagarajan, Arvind. "TJHSST Class of 2009 Senior Destinations" <http://www.kavitech.com/EduInfo/Destinations/Destinations.html>

[2] Chen, Jeff. "TJHSST Class of 2008 Senior Destinations" <http://www.tjhsst.edu/jchen2/college>

[3] Wang, Jonathan and Zeng, Will. "TJHSST Class of 2007 Senior Destinations" <http://www.tjhsst.edu/pwang/college/base.php>

[4] "Fairfax County School Board Votes to Change Grading Scale." Fairfax County Public Schools, 1/23/2009. <http://commweb.fcps.edu/newsreleases/newsrelease.cfm?newsid=1058>

[5] Chang, Lin "Applying Data Mining to Predict College Admissions Yield: A Case Study" New Directions for Institutional Research, n131 p53-68 Fall 2006

[6] Sauer, Timothy "Numerical Analysis", Addison Wesley, 2005, ISBN 03211268989

[7] Sedgewick, Robert and Wayne, Kevin, "GaussianElimination.java" 9/29/2009 <http://www.cs.princeton.edu/introcs/95linear/GaussianElimination.java.html>