0

AMATH 582 Final Project

Solomiya Bilyk, Peter Sciuto, Tomas Perez

March 17, 2021

Abstract

In today's competitive market, earning a higher education allows for people to increase their skill sets in order to receive a higher paying job. Individuals are thriving to make the most money but may be unsure of where to invest their money in order to get the desired return. This is a classic problem in econometrics and data analysis. What college major provides the greatest return on investment? Does obtaining a graduate degree increase earnings? Is the cost worth it? These questions is often tackled with the use of simple linear regression. This research paper will take a different approach and focus on using a Naive Bayes Classifier in an exploratory analysis of the relationship between costs and earnings, and various fields of study, degree types, and graduation rates, costs, and earnings.

1 Introduction and Overview

The goal of this study is to explore several key questions that come up for students entering college. Which college majors provide greater earnings outcomes? Does obtaining a graduate degree provide greater earnings outcomes? Should I attend the more expensive school? These questions will be addressed by using a Naive Bayes Classifier to make some cost-benefit comparisons.

1.1 Data

The data used in this paper comes from the U.S. Department of Education's College Scorecard data set. The College Scorecard provides data for individual students that includes institution information, field of study, degree level, debt, and first two years of earnings after graduation. To work with this data set, it was necessary to clean it first and only work with the fields that were important for this study. For example, many inputs in the columns were either privately suppressed or "NaN" meaning that the data was not available and thus had to be removed. The data-set contained different degree levels such as Associate's, Bachelor's, Master's, Graduate certificates, and Doctorate degrees. For the purposes of this study, the data was limited to Bachelor's and Master's level degrees. Additionally, the earnings and debt numbers were provided as strings and they had to be converted into numerical values.

The list of majors was also extensive. Many of the majors were listed as separate, but could be seen as the same broad category. For example, two separate fields of study are **Teacher Education and Professional Development, Specific Levels and Methods** and **Teacher Education and Professional Development, Specific Subject Areas**. Part of data setup was to take similar fields-of-study and combine them into large combined groups. The two majors listed above were combined into the **Education** category. It's worth noting that the methodology for combining certain majors into certain categories is largely subjective, and there can be numerous arguments for and against whether specific majors belong in a particular group.

2 Theoretical Background

2.1 SVD

The single value decomposition is a method of decomposing matrices to analyze the principal components within them. It can be applied to any data matrix and results in a U, S and V matrices. U, S and V are the outputs from the SVD and when multiplied in a certain order, accurately recreate the original data set (shown in Equation 1). In this equation A is the original data, U contains the principal directions of the data, S contains ordered singular values along the diagonal that indicate the strength of the principal directions, and V represents how the data in A projects along the principal components.

$$A = U\Sigma V^* \tag{1}$$

2.2 Naive Bayes

As learned from the following article [1], the Naive Bayes algorithm is a machine learning algorithm and is part of a supervised learning technique. Additionally, it is a classification technique, where the algorithm is based on the Bayes Theorem. Bayes Theorem calculates conditional probability and is formed with these two equations:

$$P(A|B) = \frac{P(B \cap A)}{P(B)} \tag{2}$$

$$P(B|A) = \frac{P(B \cap A)}{P(A)} \tag{3}$$

Where $P(A|B)$ is the conditional probability of A happening if B happened. $P(A)$ is the probability that A happened, $P(B)$ is the probability that B happened, $P(B|A)$ is the conditional probability of B happening if A happened. Furthermore, putting Equation 1 and Equation 2 formulates the following equation, which defines the Bayes Theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{4}$$

Equation 3 is for a single predictor variable, which may not be very useful for big data sets. In order to maximize the benefits of this equation, the classes and predictor variables can be changed to more than one. This leads into the Naive Bayes Algorithm, which allows for more predictor variables and output classes. The purpose of this algorithm is to calculate the conditional probability of an event with a vector $x_1, x_2, ..., x_n$ that is a part of a specific class C_i. The following illustrates the equation for Naive Bayes:

$$P(C_i|x_1, x_2, ..., x_n) = \frac{P(x_1, x_2, ...x_n|C_i)P(Ci)}{P(x_1, x_2, ..., x_n)} for 1 < i < k \tag{5}$$

This the computation of Equation 4:

$$\begin{align} P(x_1, x_2, ..., x_n|C_i)P(C_i) &= P(x_1, x_2, ..., x_n, C_i) \tag{6} \\ P(x_1, x_2, ..., x_nC_i) &= P(x_1|x_2, ..., x_n, C_i)P(x_2, ..., x_n, C_i) \tag{7} \\ &= P(x_1|x_2, ..., x_n, C_i)P(x_2|x_3, ..., x_n, C_i)P(x_3, ..., x_n, C_i) \tag{8} \\ &= ... \tag{9} \\ &= P(x_1|x_2, ..., x_n, C_i)P(x_2|x_3, ..., x_n, C_i)...P(x_{n-1}|x_n, C_i)P(x_n|C_i)P(C_i) \tag{10} \end{align}$$

Because the conditional probability adds up to $P(x_j|C_i)$, Equation 9 can be written as follows:

$$P(C_i|x_1, x_2, ..., x_n) = (\prod_{j=1}^{j=n} P(x_j|C_i) \frac{P(C_i)}{P(x_1, x_2, ..., x_n)} for 1 < i < k \tag{11}$$

Furthermore, $P(x_1, x_2, ..., x_n)$ is a constant, and Equation 10 can be rewritten as follows:

$$P(C_i|x_1, x_2, ..., x_n) \propto (\prod_{j=1}^{j=n} P(x_j|C_i) for 1 < i < k \tag{12}$$

3 Algorithm Implementation and Development

The same code that was used to clean the ‘FieldofStudy’ data was used to clean the aggregate ‘merged’ data, by removing rows and columns that were missing numerical data. The resulting data set was made up of 1209 institutions, each with 23 variables reported. An SVD was performed on the reduced data set using the svd command and the normalized singular values from S were plotted to determine the rank of the data (shown in Figure 1). Though nearly all of the energy is contained in the first 4 modes, a rank of 7 was used since the data set was not large and the computational cost of analysis was low. By analyzing the matrices output by the SVD, it was determined that the overall cost of attendance was the most weighted variable among the principal components (also shown in Figure 1). The relative weight of variables can be seen in Figure 1 with $CostofAttendance = 17$. A 2-D projection of select v-modes was done to attempt to visualize any obvious patterns. Figure 2 shows this projection and there appears to be two main clusters within the data set.

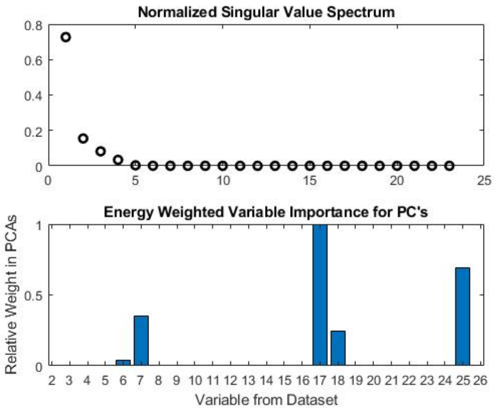


Figure 1: Singular Values and Variable Weight

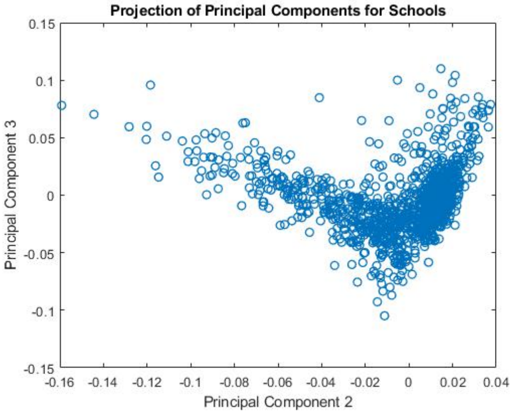


Figure 2: Singular Values and Variable Weight

Using the sort command, the data was sorted by graduation rates and plotted to view how the clustering fit the data set. The split and labeled data set was then sorted into a random shuffling of a training and test set, and the fitgmdist command was used to apply an unsupervised classifier (Guassian Mixture Model) to the data. The classifier was run for 100 iterations, with a different random selection of training and test data to ensure cross-validation.

The principal components of the data were then analyzed further to determine which variables from the data set were most important and could explain the patterns seen above. The reduced U and S matrices are multiplied together to find the relative weight of each variable in each mode.

Once the important variables are determined, the data set is again split and labeled and this time put into two supervised classifiers. The training and test sets are randomly determined again and both the Naïve Bayes and LDA classifiers are run using the fitcnb and fitcdiscr functions, for 1000 iterations in order to cross-validate. The accuracy of these two methods is compared by subtracting the expected values from the predicted values, and summing the number of remaining unitary values.
The code for the Naive Bayes Classifier remains mostly identical for both parts of the study. What changes is the data analyzed. Here are the steps that were taken to achieve the results:

1. Create a separate .mat file with the data-set. This file will include the Naive Baynes algorithm and a separate .mat file will be used to call out this algorithm per specific major field.

2. In the Naive Baynes function file, the data is reduced to Bachelor’s degree and Master’s degree. Based on earnings and debt fields, the algorithm will project which classification group it will fall within.

3. In the Naive Baynes function file, there was an equation to find the mean value of the debt and earnings. Furthermore, the standard deviation was calculated for the debt and earnings data.
4. Then, the debts and the earnings are being sorted from minimum to maximum in order to use the meshgrid command.
5. Fitcnb command is used to calculate the Naive Baynes algorithm based on the probability of the debt and earning values for each degree type.
6. After the algorithm is calculated, a plot is drawn.
7. In order to call the Naive Baynes function, a separate file is created where the specific major needs to be projected.

Algorithm 1: Naive Bayes

```
data = readtable('DataWithDebtAndEarningValues.csv')
delete data for degree types that are not Bachelor's or Master's
delete data for majors not being used in classifier
NaiveBayesFigureFunction(MajorData, 'MajorData Classification');
```

4 Computational Results

4.1 Graduation Rates and Cost of Attendance

The dominant variable (cost of attendance) was plotted against graduation rate to see the relation between variables (Figure 3). The results are interesting and appear to indicate two distinct groups of institutions with a rough separation value of about \$32,000. Each group has a higher graduation rate as the cost goes up, but it seems to reset after the \$32,000 threshold.

Since clustering based on cost presented itself within the data and this variable was dominant in the principal components, it was selected as a way to separate the data and attempt supervised classifying. The data set was split at the \$32,000 threshold and 1000 iterations of Naïve Bayes and LDA were run. Both methods were exceptionally accurate at classifying the data by overall attendance cost with Naïve Bayes having 96.33% accuracy and LDA having 98.44% accuracy (results shown in Figure 4), confirming that the data clustering can be split by this variable.

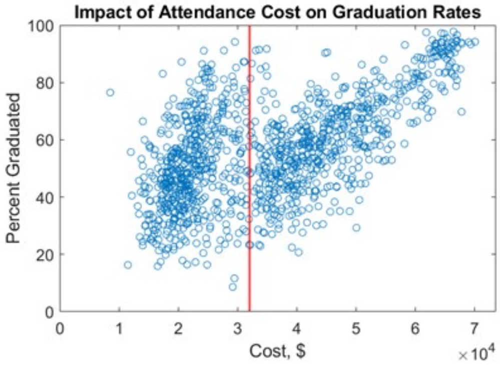


Figure 3: Graduation Rates

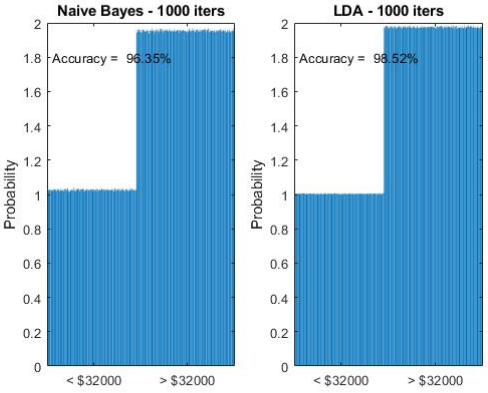


Figure 4: NBC vs LDA

4.2 Fields of Study

Six broadly defined ”fields” were created from the selection of majors in the College Scorecard dataset: Arts, Business, Computer Science, Education, Engineering, and Social Sciences. The decision to compare these five fields of study was arbitrary. Appendix A [1] holds visualizations the the classification of two fields of study. The fields are classified based on the normalized average debt accumulated (x-axis) and the normalized average earnings (y-axis) within the first couple years after graduation.

Immediately noticeable is the relatively worse outcomes of Arts and Social Science majors compared to the other fields of study. They trend towards the lower half of the Earnings axis, and spread along the debt axis. Compared to one another, the Naive Bayes Classifier struggles to distinguish the two. Education majors are also generally concentrated in the lower half of the earnings axis. Though slightly better off than the arts and social sciences. There is no clear indicator for education that more debt correlates to higher earnings. Business majors are generally difficult to separate. Their outcomes are heavily clustered in the bottom-left near the mean, but trend outward with a higher correlation between debt and earnings. This likely points to the idea that obtaining a graduate degree in business could be lead to better earnings outcomes. Unfortunately for the classifier, it makes classification more difficult since the results are more widespread across earnings and debt. Computer science and engineering trend similarly and are therefore difficult to distinguish between one another. Most of the data is clustered to the left side of the debt axis, with outcomes varying without a clear trend as more debt is accumulated. This could indicate that obtaining a graduate degree in either of these fields doesn’t necessarily imply greater earnings after graduation.

4.3 Obtaining a Graduate Degree

The results of this study were focused on predicting whether a person has a Bachelor’s or Master’s degree based on the same criteria as the comparison of different fields of study: debt and earnings. Figure 1 illustrates the Applied Mathematics Classification of Naive Bayes algorithm. It is seen from this plot that there was not a lot of data to work with, however the algorithm produced two clear categories where a prediction would fall to. The probability dots do not mix between each other’s degree fields, therefore making this chart very reliable.

Figures 5 - 10 illustrates the Naive Bayes classification for the different fields of study. Of the six fields of study, business majors proved to be the most difficult for the NBC to classify. While obtaining a Master’s degree, and subsequently obtain more debt, does trend towards better earnings outcomes, there is still a large concentration of graduates around the mean regardless of the degree type. Classifying computer science and engineering produced similar results. The NBC did a better job compared to business majors, but there was still a fair amount of overlap around the probability edges. Arts and social sciences were somewhat easier to classify, however the data indicates that obtaining a Master’s degree does not lead to significantly improved earnings. Any improvement in earnings is coupled with a larger accumulation of debt. Education has the clearest indicator of improved earnings after obtaining a Master’s. Most of the data points that did get classified incorrectly lie closer to the the probability edge. Overall, the plots still show decent conclusions as the majority of the probabilities stay close to each other in their labeled classifications.

[1] The graphs use color to distinguish the different classes and how the Naive Bayes algorithm classifies them. Unfortunately, the manner in which the classifier defines those colors is somewhat random. So the color assigned to each variable and the color that the NBC uses to indicate it’s classification isn’t consistent. Please, pay careful attention to the index on each graph.

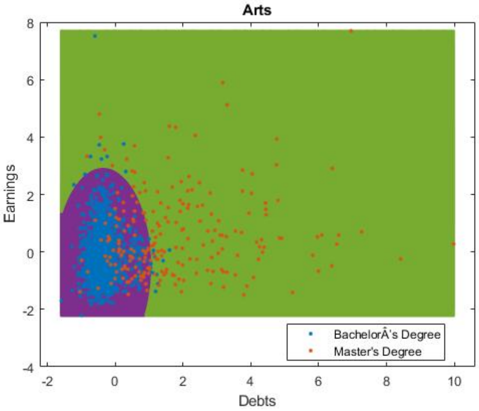


Figure 5: Degree Type: Arts

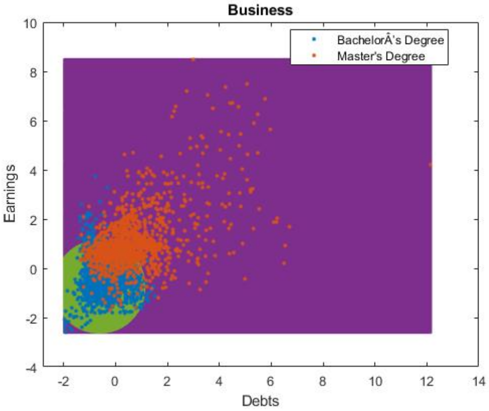


Figure 6: Degree Type: Business

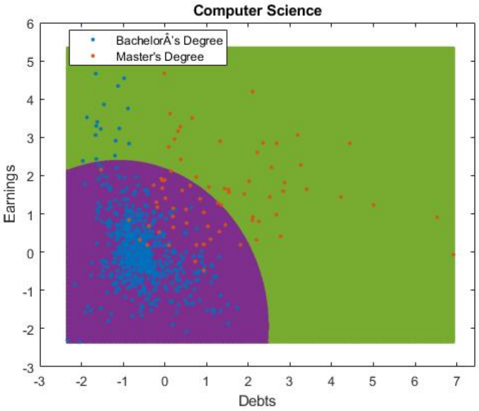


Figure 7: Degree Type: Computer Science

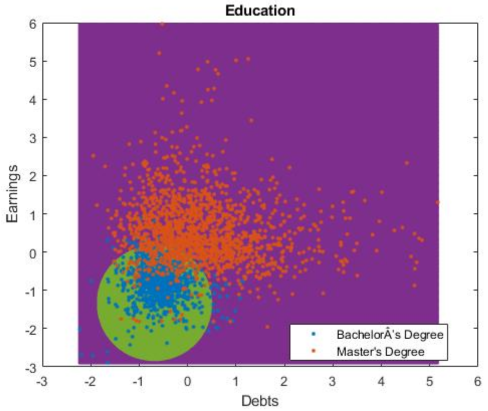


Figure 8: Degree Type: Education

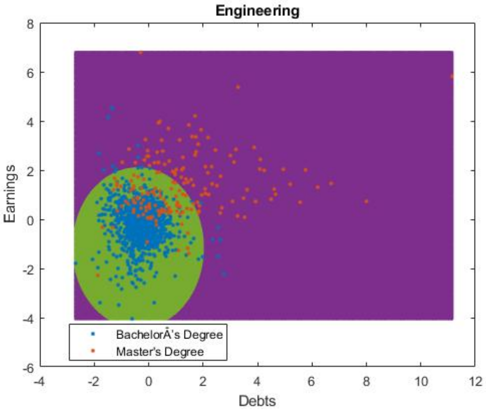


Figure 9: Degree Type: Engineering

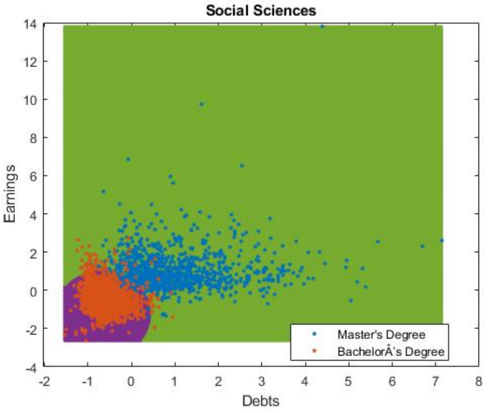


Figure 10: Degree Type: Social Science

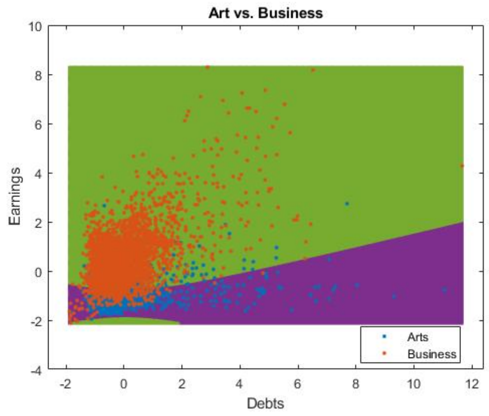
5 Summary and Conclusions

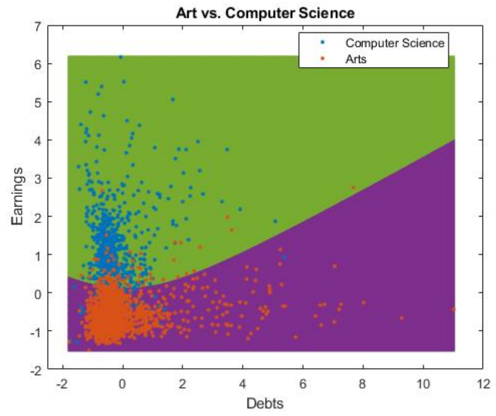
While no groundbreaking results were found while exploring the College Scorecards datasets, the experience provided the opportunity to with large, messy datasets to attempt to find and quantify some patterns within. The variables present in the principal components were successfully identified and two methods of supervised classification, NBC and LDA, were also successful in distinguishing institution with different costs of attendance. The robustness of the supervised classifiers could be tested by cleaning and importing data from other years to use as a test set against the training set. Classifying fields of study based on earnings and debt produced similarly predictable results. Some fields were more easily classified than others by the Naive Bayes Classifier. The largest differences appeared to be between STEM fields and fields that could be considered liberal arts. Business proved to be the most difficult to distinguish from other majors. Business was also the most difficult to separate undergrads and graduates, while education is arguably the easiest field of study to make an argument for getting a graduate degree.

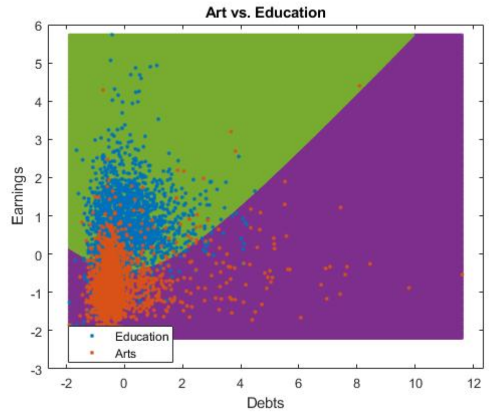
References

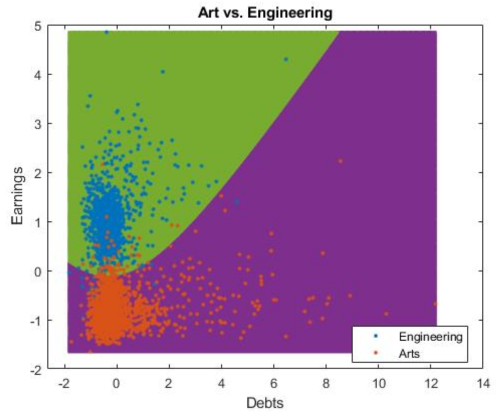
[1] *A Comprehensive Guide To Naive Bayes In R.* URL: `https://www.edureka.co/blog/naive-bayes-in-r/`.

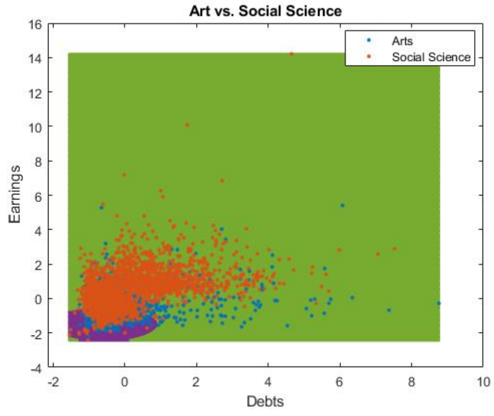
Appendix A Field of Study Comparison

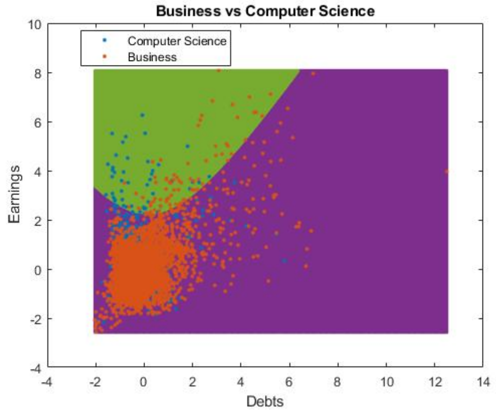

Art vs. Business

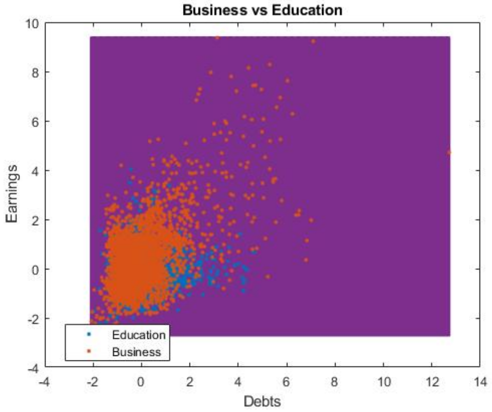

Art vs. Computer Science

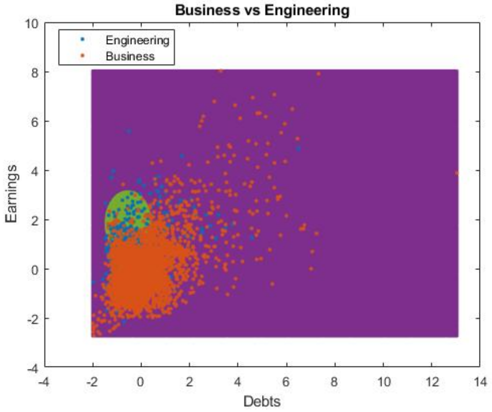

Art vs. Education

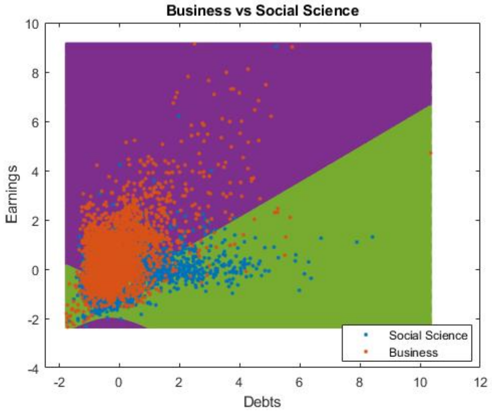

Art vs. Engineering

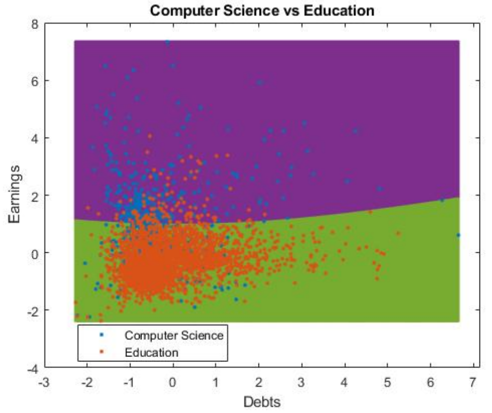

Art vs. Social Science

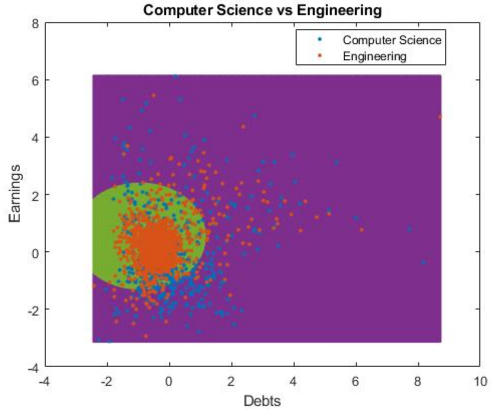

Business vs Computer Science

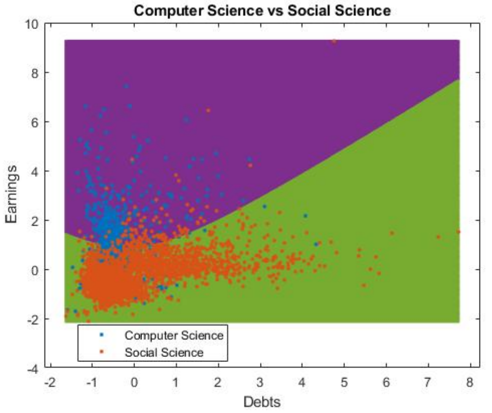

Business vs Education

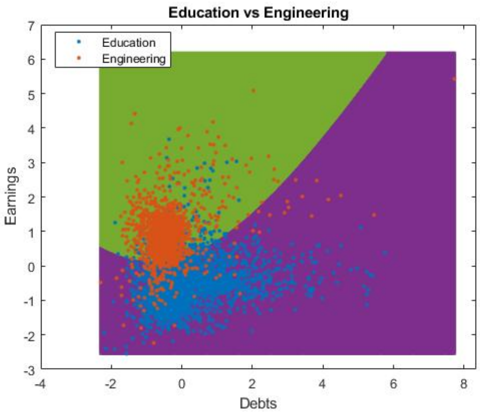

Business vs Engineering

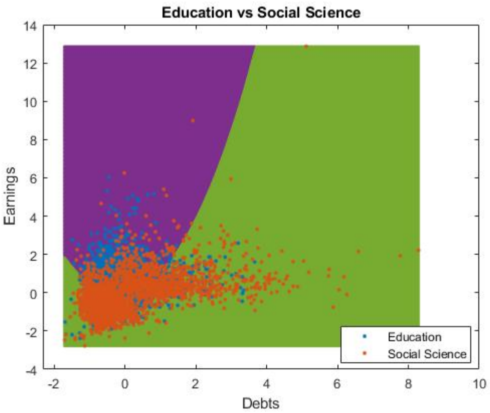

Business vs Social Science

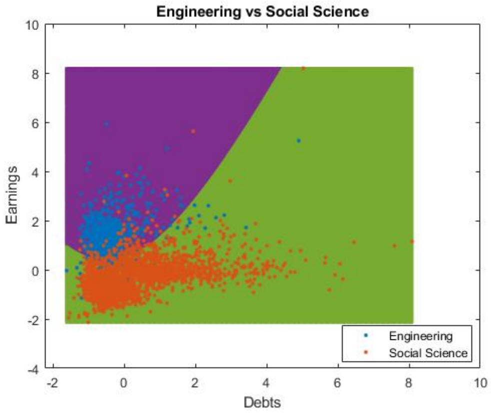

Computer Science vs Education


Computer Science vs Engineering


Computer Science vs Social Science







Appendix B MATLAB Functions

Below are the important MATLAB functions that were used in this research assignment:

- `M = mean(A)` returns the mean of the elements of A along the first array dimension whose size does not equal 1.
- `S = std(A)` returns the standard deviation of the elements of A along the first array dimension whose size does not equal 1.
- `M = min(A)` returns the minimum elements of an array.
- `M = max(A)` returns the maximum elements of an array.
- `Mdl = fitcnb(Tbl, formula)` returns a multiclass naive Bayes model (Mdl), trained by the predictors in table Tbl. formula is an explanatory model of the response and a subset of predictor variables in Tbl used to fit Mdl.
- `ypred = predict(mdl,Xnew)` returns the predicted response values of the linear regression model mdl to the points in Xnew.
- `gscatter(x,y,g)` creates a scatter plot of x and y, grouped by g. The inputs x and y are vectors of the same size.
- `T = readtable(filename)` creates a table by reading column oriented data from a file.
- `[X,Y] = meshgrid(x,y)` returns 2-D grid coordinates based on the coordinates contained in the vectors `x` and `y`. X is a matrix where each row is a copy of `x`, and `Y` is a matrix where each column is a copy of `y`. The grid represented by the coordinates `X` and `Y` has `length(y)` rows and `length(x)` columns.

- Md1 = fitdiscr(Tbl, formula) returns a fitted discriminant analysis model based on the input variables contained in the table Tbl. formula is an explanatory model of the response and a subset of predictor variables in Tbl used to fit Mdl.

Appendix C MATLAB Code

Add your MATLAB code here. This section will not be included in your page limit of six pages.

```
function NBFunc(data, titleName)

    %data has two majors in it
    meanDebt = mean(data.P_DEBTMEAN);
    stdDebt = std(data.P_DEBTMEAN);
    stand_P_DEBTMEAN = (data.P_DEBTMEAN - meanDebt) / stdDebt;
    data.P_DEBTMEAN = stand_P_DEBTMEAN;

    meanEarn = mean(data.P_MD_EARN_WNE);
    stdEarn = std(data.P_MD_EARN_WNE);
    stand_P_MD_EARN_WNE = (data.P_MD_EARN_WNE - meanEarn) / stdEarn;
    data.P_MD_EARN_WNE = stand_P_MD_EARN_WNE;

    debt = min(data.P_DEBTMEAN) : 0.01 : max(data.P_DEBTMEAN);
    earn = min(data.P_MD_EARN_WNE) : 0.01 : max(data.P_MD_EARN_WNE);

    % Create classification model from data
    model = fitcnb(data, 'P_MAJORID~P_DEBTMEAN+P_MD_EARN_WNE');

    [d, e] = meshgrid(debt, earn);
    ms = predict(model, [d(:) e(:)]);

    color = lines(6); % Generate color values
    gscatter(d(:), e(:), ms, color(4:6,:));

    hold on;

    title(titleName);
    xlabel('Debts');
    ylabel('Earnings');

    gscatter(data.P_DEBTMEAN, data.P_MD_EARN_WNE, data.P_MAJOR, color(1:6,:), '.', 8);
end
```

Listing 1: Naive Bayes Function Algorithm

```matlab
clc;clear all;close all;
%warning off;

% P_MAJORID P_MAJOR (general, poorly-defined categories for certain major types)
% 1 Agriculture
% 2 Arts (check)
% 3 Business (check)
% 4 Communication
% 5 Computer Science (check)
% 6 Construction
% 7 Education (check)
% 8 Engineering (check)
% 9 Information
% 10 Language
% 11 Law
% 12 Math
% 13 Medical
% 14 Nature
% 15 Social Science/Studies (check)
% 16 Sciences
MajorList=[2,3,5,7,8,12,13,15,16,17];

data = readtable('DataWithDebtAndEarningValuesTEST.csv');

%removes degrees that are not bachelors(3) or masters(5) or doctors(6)
%toDelete = (data.P_CREDLEV ~= 3 & data.P_CREDLEV ~= 5 & data.P_CREDLEV ~= 6);
toDelete = (data.P_CREDLEV ~= 3 & data.P_CREDLEV ~= 5);
data(toDelete,:) = [];


% Make new matrix of just two major categories
MajorData = data; %currently has everything in it
major1=3;
major2=15;
% remove every row that isn't one of the two defined major categories
toDelete = (MajorData.P_MAJORID ~= major1 ...
    & MajorData.P_MAJORID ~= major2);
MajorData(toDelete,:) = [];
% now MajorData just has data for two major categories

%pass in matrix of two majors
NBFunc(MajorData, 'Business vs Social Science');
```

Listing 2: File that calls the Naive Bayes algorithm to run specific degree fields