Predicting post Collegiate Earnings and Debt

November 4 , 2016

Akshat Tandon, Prakhar Pandey

Project objective:

Predicting post collegiate earnings and debt of students based on factors that reflect the current status of each institute such as majors offered, SAT scores, selection rates, student demographics etc.

Arizona State University
SAT Score
Degrees Offered
Total Black Undergraduates
Average faculty salary

Earnings After 6 years

Debt After 6 years

About The Dataset

Dataset Information

The U.S. Department of Education launched College Scorecard in September 2015, a data set which describes around 1800 features such as earnings, debts, passing rates, admission rates, demographics etc, for most of the colleges in America.

Source: https://collegescorecard.ed.gov/data/

DataSet Information

- There are about 1700 features for some 7600 colleges.
- Some feature values are absent for some of the colleges.
- Some features that are not available are marked with ‘NaN’
- If data was prepared from less than 30 students, it was labelled as “Privacy Suppressed”
- We have data from years 1996 to 2013. We choose data of year 2011 because it is least sparse.

331	Certificate of less than one academic year in Agriculture, Agriculture Operations, And Related Sciences.
332	Certificate of at least one but less than two academic years in Agriculture, Agriculture Operations, And Related Sciences.
333	Associate degree in Agriculture, Agriculture Operations, And Related Sciences.
334	Awards of at least two but less than four academic years in Agriculture, Agriculture Operations, And Related Sciences.
335	Bachelor's degree in Agriculture, Agriculture Operations, And Related Sciences.
336	Certificate of less than one academic year in Natural Resources And Conservation.
337	Certificate of at least one but less than two academic years in Natural Resources And Conservation.
338	Associate degree in Natural Resources And Conservation.
339	Award of at least two but less than four academic years in Natural Resources And Conservation.
340	Bachelor's degree in Natural Resources And Conservation.
341	Certificate of less than one academic year in Architecture And Related Services.
342	Certificate of at least one but less than two academic years in Architecture And Related Services.
343	Associate degree in Architecture And Related Services.
344	Award of more than two but less than four academic years in Architecture And Related Services.
345	Bachelor's degree in Architecture And Related Services.
346	Certificate of less than one academic year in Area, Ethnic, Cultural, Gender, And Group Studies.
347	Certificate of at least one but less than two academic years in Area, Ethnic, Cultural, Gender, And Group Studies.
348	Associate degree in Area, Ethnic, Cultural, Gender, And Group Studies.
349	Award of more than two but less than four academic years in Area, Ethnic, Cultural, Gender, And Group Studies.
350	Bachelor's degree in Area, Ethnic, Cultural, Gender, And Group Studies.
351	Certificate of less than one academic year in Communication, Journalism, And Related Programs.
352	Certificate of at least one but less than two academic years in Communication, Journalism, And Related Programs.
353	Associate degree in Communication, Journalism, And Related Programs.
354	Award of more than two but less than four academic years in Communication, Journalism, And Related Programs.
355	Bachelor's degree in Communication, Journalism, And Related Programs.
356	Certificate of less than one academic year in Communications Technologies/Technicians And Support Services.
357	Certificate of at least one but less than two academic years in Communications Technologies/Technicians And Support Services.
358	Associate degree in Communications Technologies/Technicians And Support Services.
359	Award of more than two but less than four academic years in Communications Technologies/Technicians And Support Services.
360	Bachelor's degree in Communications Technologies/Technicians And Support Services.

Problem Challenges:

- Handling NaN values in the data
- Handling ‘Privacy Suppressed’ tag in data
- Handling of categorical values, unrelated features.
- Handling of those features in which same value repeats many times
- Remove features which are strongly correlated to our Prediction Variable
- Building different prediction models and comparing their accuracies

PART 1

Data Preprocessing & Feature Extraction

PART 1 : DATA PREPROCESSING

Handle NaN and Categorical :

NaN was replaced with 0

Categorical values were removed

Removal of Non Informative Features :

Non informative features like ‘College Name’ , ‘ZIP code’ were simply removed as they offered no predictive power to our models.

PART 1 : DATA PREPROCESSING

Handle Privacy Suppressed :

Methods Tried:

- **Replace with 0 , mean**
- **Replace Numeric with median and Categorical with Max Frequency Item**
- **Interpolation of Data**

Last method worked best

PART 1 : Feature Extraction

Feature Selection (Manual Feature Removal):
Features related to Debt , Earnings and Repayment.

Features Removed:

Median Debt After 10 years

Median Income after 8 years

Federal Loans Given

850 features were manually selected and removed.

PART 1 : Feature Extraction

Useless Feature Removal :

Those features that offered very low information were removed .For example if for all the instances a variable have the same value, it is of no use, so it was removed. Count of 6500 was used as cutoff.

So total 579 features were left to build model from 1729 original ones.

PART 1 : Feature Extraction

Recursive Feature Extraction :

Had to reduce size because just 4000 training example and 579 features.

Trouble would increase with more complex models.

Use RFE to further reduce the size of feature set to 170

Used Linear Regression as Estimator

Why Not to Use PCA ?

Exploring The Prediction Variables

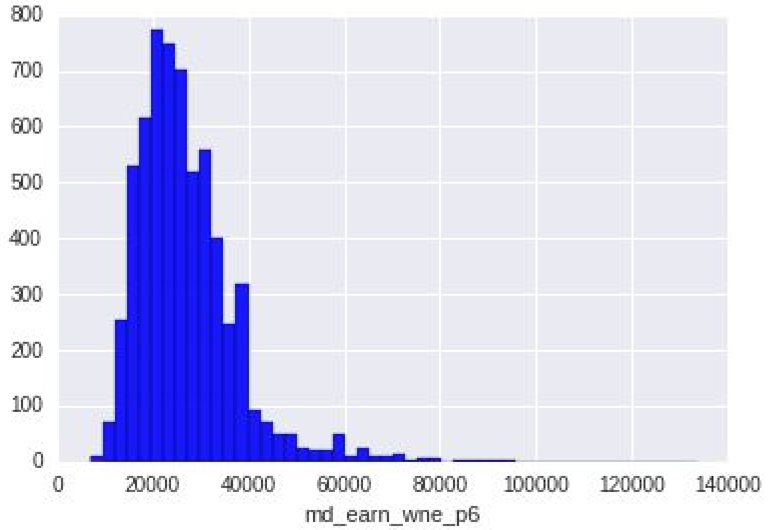
The Variables to Predict :

1) Median earnings of students after 6 years of graduation

```
Some Statistics of the Income:

count      6224.000000
mean      27045.453085
std       10750.511323
min        7000.000000
25%       20000.000000
50%       25200.000000
75%       31600.000000
max      133600.000000
Name: md_earn_wne_p6, dtype: float64

<matplotlib.axes._subplots.AxesSubplot at 0x7fd92ec507b8>
```



The Variables to Predict :

1) Median earnings of students working and not enrolled 6 years after entry

PAR_ED_PCT_HS	-0.497050
PCTPELL	-0.472607
NOT1STGEN_ENRL_ORIG_YR2_RT	0.396357
PCIP12	-0.391295
ENRL_ORIG_YR2_RT	0.390351
FIRSTGEN_ENRL_ORIG_YR2_RT	0.336683
UNKN_ORIG_YR6_RT	-0.323120
PAR_ED_PCT_MS	-0.313580
UNKN_ORIG_YR2_RT	-0.299512
PCIP14	0.244295
COMP_4YR_TRANS_YR4_RT	0.238807

--Percent of students whose parents' highest educational level is high school
--Percentage of undergraduates who receive a Pell Grant

The Variables to Predict :

2) Median debt of People who complete the degree.

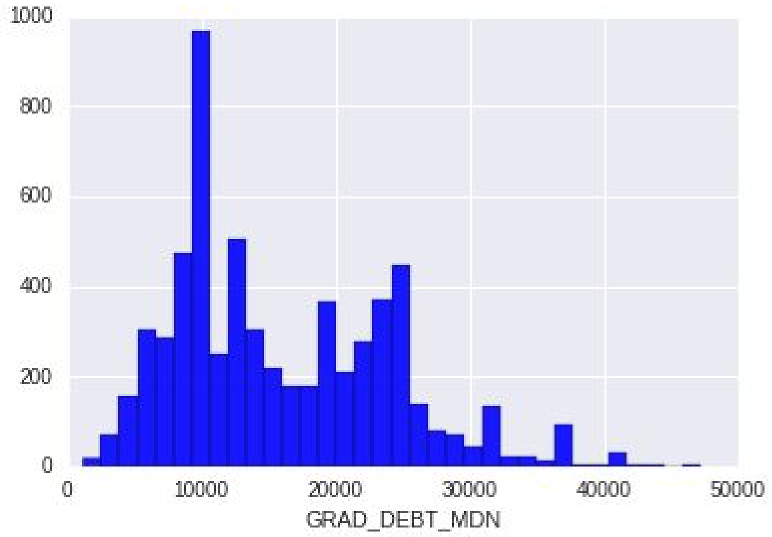
The Data is Not Normally Distributed

Didn’t Use any Gaussian Method

```
Some Statistics of the Income:

count     6236.000000
mean     15890.949487
std       8108.544677
min       1144.000000
25%       9500.000000
50%      13871.500000
75%      22220.500000
max      47186.500000
Name: GRAD DEBT MDN, dtype: float64

<matplotlib.axes._subplots.AxesSubplot at 0x7f4eacec9550>
```



The Variables to Predict :

2) Median debt of People who complete the degree.

0	WDRAW_ORIG_YR8_RT	0.533430
1	ENRL_ORIG_YR2_RT	0.499865
2	COMP_ORIG_YR2_RT	-0.484720
3	ENRL_4YR_TRANS_YR2_RT	0.458423
4	FIRSTGEN_DEATH_YR3_RT	0.410122
5	WDRAW_4YR_TRANS_YR2_RT	0.409997
6	WDRAW_4YR_TRANS_YR6_RT	0.400837
7	MALE_WDRAW_ORIG_YR3_RT	0.396100
8	FEMALE_WDRAW_4YR_TRANS_YR2_RT	0.392956

1)Percent withdrawn from original institution within 8 years
3)Percent of low-income students who completed within 2 years at original institution

PART 2

Prediction Models

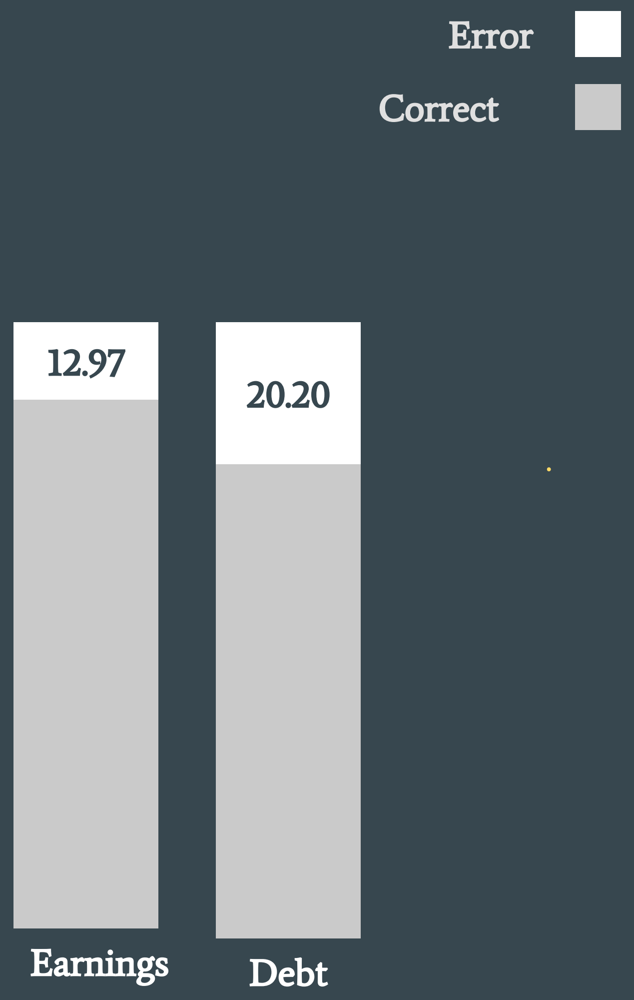
Linear Regression

Model

Given a list of features for a school, predict students median debt at graduation and median earnings 6 years after graduation

Result:

- 54.69% mean absolute error for earnings.



Linear Regression

Pitfalls

As we saw, the results were not satisfactory using linear regression.

How to improve:

- Tuning number of privacy suppressed features
- Pruning feature space
- Enabling model to learn non linear relationships b/w features and earnings/debt.

Locally weighted linear regression

Captures nonlinearity!

- In linear regression we find parameters which minimize, $\sum_i (y^{(i)} - \theta^T x^{(i)})^2.$
- In locally weighted we find parameters which minimize, $\sum_i w^{(i)} (y^{(i)} - \theta^T x^{(i)})^2.$

We assign weights to each of our training samples, such that the samples which are closer to the test point are allocated higher weights.

Thus while estimating the parameters, the points which are closer to the test point will have a higher contribution as compared to the farther off points.

Locally weighted linear regression

- A fairly standard choice for the weights is $$w^{(i)} = \exp\left(-\frac{(x^{(i)} - x)^2}{2\tau^2}\right)$$

- Another thing to note, LWR requires the entire data set every time you try to make a prediction making it much more computationally expensive compared to the simple linear regression. We have to do this because every time we try to make a prediction we are constructing a regression line that's local to the data point of our interest.

Locally weighted linear regression

Data Standardization

To make the Euclidean distance (the norm in the equation above) meaningful, we standardized features to zero mean and unit variance prior to computing the weights.

46.91% error in Earnings

KNN Regression

The KNN algorithm predicts by taking K nearest neighbours average from train Dataset.

With Euclidean Distance :

Error : 17.67 for K = 10 (by trying out different K values like 7,10,15,20)

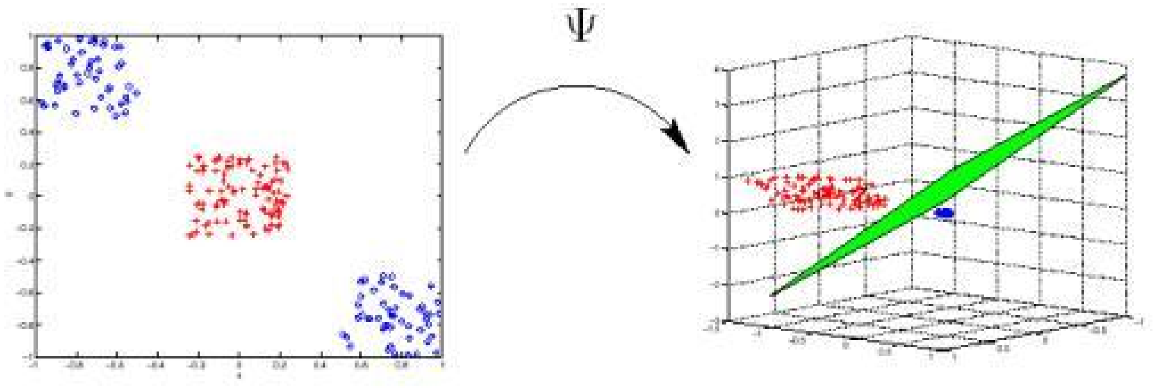
With Mahalanobis Distance :

Error : 19.72 for K = 10 (by trying out different K values like 7,10,15,20)

Support Vector Machines (Regression) :

In SVM Regression , input is mapped to higher dimensional feature space, then a linear model is constructed

- 30.66 % Error
- Using RBF kernel of degree 3
- C=1 (penalty)
- Epsilon= 0.1 (insensitivity)
- Gamma = 1/n_features



Neural Nets

- Trained simple neural networks with a single hidden layer, using the previous feature selection and imputation for privacy-suppressed values
- A single hidden layer was chosen because there was insufficient training data (number of schools) to fit a model with more parameters without significant overfitting
- A hidden layer with 10 nodes performed optimally for earnings with 24.19% error

Neural Net learning via levenberg-marquardt algo

- In our simple gradient descent based training of the neural net, convergence can take an extremely long time

$$w_{i+1} = w_i + delta * \nabla (E)$$

- It is because due to the rigidness of the step size which does not depend on the curvature of the error surface.
- When descending the walls of a very steep local minima we must use a small step size to avoid missing out the minima.
- When moving along a gently sloping part of the error surface we want to take large steps else it will take forever to reach the minima

Neural Net learning via levenberg-marquardt algo

Momentum alleviates many problems described previously.

Levenberg-marquardt algo is used for estimating the parameters of a model curve so that it fits a given dataset.

$$\hat{\beta} = \underset{\beta}{\mathbf{argmin}} \sum_{i=1}^{m} [y_i - f(x_i, \boldsymbol{\beta})]^2 .$$

It improves basic gradient descent by estimating curvature information.

Neural Net learning via levenberg-marquardt algo

The Levenberg-Marquardt algorithm is a very simple, but robust, method for approximating a function. Basically, it consists in solving the equation at every epoch:

$$(J^T J + \lambda I)\delta = J^T E$$

$$\delta = (J^T J + \lambda I)^{-1}(J^T E)$$

$$W_{t+1} = W_t + \delta$$

Where J is the Jacobain matrix, E is the error vector, δ is the weight update vector and λ is the damping factor which we increase or decrease depending on our E vector.

Neural Net learning via levenberg-marquardt algo

Failure !!

- In theory this method should lead to a faster convergence but in our case it’s just the reverse.
- For a case involving a few hundreds of weights, this method converges faster than the simple gradient descent with momentum but since our neural net has 4889 weights, at every epoch we are required to take the inverse of a matrix with dimensions 4889x4889 which kind of kills any cleverness exhibited by the algorithm.

Neural Net learning via levenberg-marquardt algo Failure !!

- Also this method find the local minimum and not the global one and is also very susceptible to the initial weights and also cause
- Due to the slow learning as well as susceptibility to initial weights, we were not able to get an acceptable training.
- This method resulted in a mean error of 89%