PREDICTING PERFORMANCE OF HIGHER EDUCATION INSTITUTES WITH PATTERNS OF EXPENDITURE

Capstone Project: Machine Learning Engineer Nanodegree

Wei-Chuang Chan

November 3, 2016

I. DEFINITION

PROJECT OVERVIEW

As post-secondary education has become more common, the budget of an education institution is often limited, and funding is competitive. While working closely with many colleges and students, knowing the expectation of their investment is crucial for both. The project aims to figure out whether the allocation of expenditure can help predicting the total number of award, certificate, and degree completed, which defines the performance of the institution, as it has been widely used to evaluate the performance of higher education institutes.

This project is inspired by Udacity's capstone project guidance and uses dataset downloaded from Integrated Postsecondary Education Data System Delta Cost Project Database, which includes the data from the Academy year 1987-1988 to 2012-2013. The Delta Cost Project Database derives data from the Integrated Postsecondary Education Data System (IPEDS) surveys on finance, enrollment, staffing, completions, and student aid, and the data have been translated into analytical formats to allow for longitudinal analyses of trends in postsecondary education with a focus on revenues and expenditure.

PROBLEM STATEMENT

The question asked with this project is whether the investment can be used to predict completion number, which is a reference of performance (Jongbloed & Vossensteyn, 2001), and if spending pattern can make a better predictor for the same target. Expense categories will be from the dataset acquired through delta cost project database, The goal of this project is to identify if the expenditure pattern will be a better reference to predict high completion number than original data.

This project extracts the target variable and the spending from the dataset. 75% of the randomly selected dataset will be used as the training set, while the rest will be the testing set. f eExpenditure in training set will provide the information to acquire the spending patterns. The project will compare the performance of the model trained by original data and the model formed by expenditure pattern.

The models will be regression models, and the estimate of error between predicted values and actual value would be the reference of performance. This project plans to select the best model from multiple candidate models and tune the best-performanced model to achieve a better result. Metrics for performance includes Mean Absolute Error (MAE), Root of Mean Squared Error (RMSE), and R-squared score (R2).

METRICS

Mean Absolute Error calculates the average of the difference between predicted value and actual value.

Root Mean Squared Error takes the root of the mean squared distance between predicted value and real value. Comparing with MAE, RMSE weighs more on the error that is further away from the mean. Both MAE and RMSE are negatively oriented – meaning the less the error, the higher the accuracy is. When MAE equals to RMSE, it means every error are of the same magnitude.

R-squared calculates the variance of the actual dataset and calculates the proportion for which that the predicted data can be accountable. The residual of the variance indicates the variance caused by the difference between actual data and predicted data. The much the variance being explained by predicted data, the higher the score is, which also indicates higher accuracy. R2, however, inflates when adding more predictors (variables) to the metric. The variance caused by predicted value hence increase even without model improvement. Adjusted R-squared score (Adj R2) is developed to counter the inflation and adding a penalty for additional variables entering the metric. Adj R2 is always smaller or equals to R2 score. Predicted R2 is another score that helps to determine if a model fits the original data but less capable of providing valid predictions on new data points.

Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) are also popular tools to evaluate the information loss of a model (Vrieze, 2012). The computation of AIC and BIC is complicated, but the concept is maximum likelihood estimate of the model parameters, which is the estimate of parameters that gives the highest probability. Log of the value of maximum likelihood function is taken, which ranges from negative infinity to 0. Negative two will multiply the value of log. For AIC, this will add two times of the total number of parameters. For BIC, the number of the parameters will multiply the log of the number of observations, which is the number of data points in our case. AIC and BIC are the numbers for comparison, and absolute value interprets nothing. For both criteria, the model with a smaller number of the result is better. AIC penalize optional parameters less heavily than BIC, and BIC should always be lower than AIC when evaluating the same model (Kuha, 2004).

When comparing MAE and RMSE, RMSE puts more weights on the larger error, and MAE behaves less sensitive to outliers. When comparing the model of original data and the model of the transformed dataset, it is likely the dimensionality changes. R2 score should not be the metric due to the inflation, and Adjusted R2 will be a better choice. While R2 ranges from 0 to 1, Adj R2 can be negative, which makes it more complicated than R2 to be understood. This feature makes Adj R2 rather easy to understand and straightforward. AIC and BIC are very common metrics while evaluating model performance, and both have the feature that penalizes over-fitting. This character makes AIC and BIC the most stands out metrics among above. AIC and BIC provide a relative reference of model performance, but no absolute reference that determines whether the model predicts accurately or not. Adjusted R2 will add an extra reference as well.

II. ANALYSIS

DATA EXPLORATION

The dataset retrieved from IPDES consists of revenue and expenditure of institutes across the USA. The total number of the original data downloaded is 221597, and the total number of features is 1007. To focus on the expenditure of each institute, only the sum of each category that listed as expenditure is selected, and total completion number is the target variable for this project. The academic year that the data is from and the indicator of whether the data point has total completion number are for verification. In total, the selection preserves 20 columns from the dataset.

By selecting data points that contain total completions, the total number of data shrinks to 148261. Given the features chosen are numerical, the dataset contains a significant amount of null data - which is likely missing data or unreported.

For the data subset took into this project, a brief explanation of each feature will be provided. To The expenditure on salary and wage, total expense reported by FASB, and all subcategories.

- academicyear - the academic year of data reported.
- totalcompletions - the total number of degree, award, certificate granted of the current year. This feature will be the label of performance in this project.
- has_completion - Indicator of whether totalcompletions is reported. (0=not reported; 1=reported)
- instruction01 - instructional expenses for the institution and excludes administration, operations and maintenance.
- research01 - expense used to produce research outcomes excluding operation and maintenance, interest amounts attributed to the research functions.
- rschpub01 - expense for research and public service of current year
- pubserv01 - expense category that provides noninstructional services beneficial to individuals and groups external to the institution such as conferences. Operations and maintenance, interest amounts attributed to the research function are excluded.
- acadsupp01 - expenses to support instruction, research, and public service. This category includes retention, preservation, and display of education materials. Operations and maintenance and interest amounts attributed to the academic support function are excluded.
- studserv01 - expenses associate with admissions, registrar activities, and activities that contribute to students' emotional and physical well-being and their intellectual, cultural, and social development outside the formal instructional program. Operations and maintenance (and interest in the 2009 aligned form) amounts attributed to the student services function are excluded.
- instsupp01 - expense for day-to-day operational support of the institution such as space management, employee personnel, and records. Operations and maintenance and interest amounts attributed to the institutional support are excluded.
- acadinststud01 - academic and institutional support and student service total of current year
- opermain01 - expenses for operations providing service and maintenance related to campus grounds and facilities. Institutions may optionally distribute depreciation expense to this function.
- depreciation01 - total depreciation of current year
- grants01 - the sum of all operating expenses associated with scholarships and fellowships including payment in support of the cost of education or third parties for off-campus housing. Operations and maintenance and interest amounts are excluded.

- auxiliary01 - expense of all operating associated with essentially self-supporting operations of the institution such as student health services. The amount of interest is excluded.
- hospital01 - operating expenses related to a hospital run by the postsecondary institute.
- independ01 - expenses associated with operations that are independent of or unrelated to the primary missions of the institution. The amount of interest attributed to the independent operations function is excluded as well as for the expenses of operations owned and managed as investments of the institution's endowment funds.
- otheroper01 - All expense other than categories above which discontinued after the Academy year 2010.
- othernon01 - All other non-operating expense of current year
- other01 - All other expense

Data points that contain totalcompletions value are going through further preprocessing.

The dataset contains a significant amount of missing values, and the feature includes more than 95% of missing values will be removed. The remaining data points that have missing value will also be deleted. The procedure leaves 4877 instances for the following steps. After preprocessing, the dataset has 13 features that represent the expenditure, and the total number of completions as predicting target.

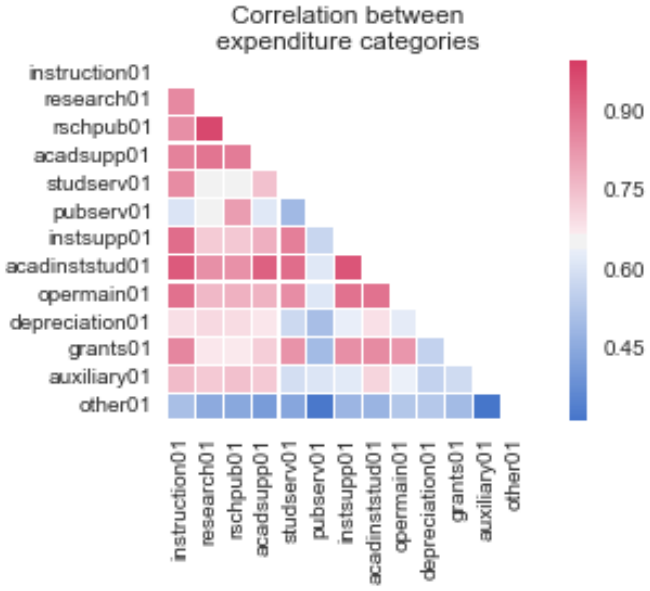

Fig 1. Correlation plot between variables

There are high correlations between some variables; instruction01 seems to correlate with most variables but other01, which has little correlation with all other variables. In this cast, PCA can help setting new dimensions that explained shared variance.

According to the heat map, instruction01 appears to have a high correlation with other features, and the other expenditure has low correlation with any other features. Features including depreciation01 and pubserv01 have no significant correlation with most other feature in general, but pubserv01 has more than 0.5 of correlation with research01 and rschpub01. Features studserv01, grants01, and auxiliary01 have a high correlation with around half of the other features but also show little correlation with the other half of the features.

As the plot shows, instruction and research expense highly correlate with nearly all kinds of expenditure. The variable other01, pubserv01, and depreciateion01 could be the most distinct features among all.

The statistics of each feature are listed below:

	instruction01	research01	rschpub01	acadsupp01	studserv01	pubserv01	instsupp01
count	4877.0	4877.0	4877.0	4877.0	4877.0	4877.0	4877.0
mean	104930051.671	51624334.7074	71576871.8127	27318777.153	14930861.8847	19952765.6751	26857903.4073
std	193357982.439	128666459.245	166433770.182	48403739.984	22631544.6616	50161840.8916	46267842.2414
min	93194.0	6.0	575.0	25051.0898438	17642.8710938	98.0	42893.0
25%	19349942.0	304089.21875	1892542.0	4355253.0	4014000.0	1032656.0	6178981.0
50%	43533552.0	2428702.0	7412453.0	10317116.0	8127099.0	3566569.0	13104256.0
75%	114612888.0	33105884.0	47508053.0	27784940.0	17248419.0	13578858.0	29344508.0
max	3295913011.0	1586856376.0	1891832868.0	575821869.0	346157859.0	544468000.0	835929442.0

	acadinststud01	opermain01	depreciation01	grants01	auxiliary01	other01
count	4877.0	4877.0	4877.0	4877.0	4877.0	4877.0
mean	69112269.4576	21292200.7827	23181883.1829	13746569.2645	36260982.3721	8777288.45677
std	109454317.728	38245286.9027	60156644.9226	26644014.0198	62627154.0254	43442837.4559
min	134151.0	10090.0	747.0	1132.0	96.0	-23825413.0
25%	15726950.0	4693000.0	2962898.0	2184892.0	4876133.0	251693.0
50%	33217794.0	9812882.0	6853124.0	5973695.0	13732010.0	1338295.0
75%	77137842.0	22100826.0	20105636.0	14820547.0	37336920.0	5383252.0
max	1536690095.0	652409120.0	1794891392.0	432682768.0	728114791.0	1360609408.0

Table 1 Statistics of features

The first thing to be noticed is that huge range within each feature, and large standard deviation indicates the distribution is spread out. Further calculating the skewness and kurtosis of each feature, the distributions are positively skew with high kurtosis.

	instruction01	research01	rschpub01	acadsupp01	studserv01	pubserv01	instsupp01
47324	47934380.0	13237111.0	15223891.0	9641878.0	4780256.0	1986780.0	11228977.0
85527	989605918.0	851197951.0	882964601.0	299645930.0	37588361.0	31766650.0	143535861.0
34927	6245878.0	4027.0	583979.0	2426205.0	749316.0	579952.0	3300617.0

	acadinststud01	opermain01	depreciation01	grants01	auxiliary01	other01
47324	25651111.0	19245540.0	4950312.0	4851802.0	20273220.0	210532.0
85527	480770152.0	130277441.0	243638010.0	101388438.0	192987802.0	12946409.0
34927	6476138.0	1100816.0	497955.0	221986.1875	355444.0	1112424.0

Table 2 Sample Data

From the first 3 data point in the training set we found that:

47324 is between the 50th percentile and the 75th percentile on instruction, between the 25th and the 50th percentile, and relatively small number of all features, 85527 has relatively large value on most features. 34927 is at the end of the lower side, but it spends more on grants and 47324.

ALGORITHMS AND TECHNIQUES

The tasks of this project include: Finding expenditure patterns and training two separate regression-predicting models with the original dataset and the transformed dataset.

The dataset will be processed to eliminate data points that are not helping the training process: data that misses target value, outliers that profoundly influence the first randomly split into a training set and a testing set at 0.75:0.25 ratio. The predicting features of the training set will go through Principal Component Analysis.

Principal Component Analysis will be implemented to find the dimensions that explained the most of the variance. The captured principal components become predicting features for comparison. To find the best model to be trained with transformed data, multiple models will be compared with the metric chosen.

Acquired dimensions will be used as the new feature to predict the target. 10-fold cross-validation will be used to evaluate the fit of the model. The use of grid search cross-validation will help to tune the model. AIC, BIC, Adjusted R2 are the metrics for evaluation of final performance.

BENCHMARK

The aim is to compare the performance of the model on original data and PCA-transformed data, and the performance score of the model trained by original data will be the benchmark for the model trained with the PCA-transformed dataset.

The benchmark for this project is the AIC, BIC, and Adjusted R2 score for an untuned k-nearest neighbor model fitted to original training data.

Benchmark: AIC: 16445.95; BIC: 16512.33; Adj R2: 0.955671

III. METHODOLOGY

DATA PREPROCESSING

The features are selected for analysis as stated above, among all the features taken (Guyon & Elisseeff, 2003). As discovered in data exploration, the dataset contains a significant amount of missing data. The features have more than 95% of missing value will also be removed. To obtain the complete data from the remaining data, we will remove any data contains any missing value removed (Williams, 2015).

As described above, the dataset is highly skewed, and the range of number is large. The dataset should be normalized before PCA to avoid any feature with a high variance to carry too much weight in principal components. The outliers of the dataset will remain to keep most of the information.

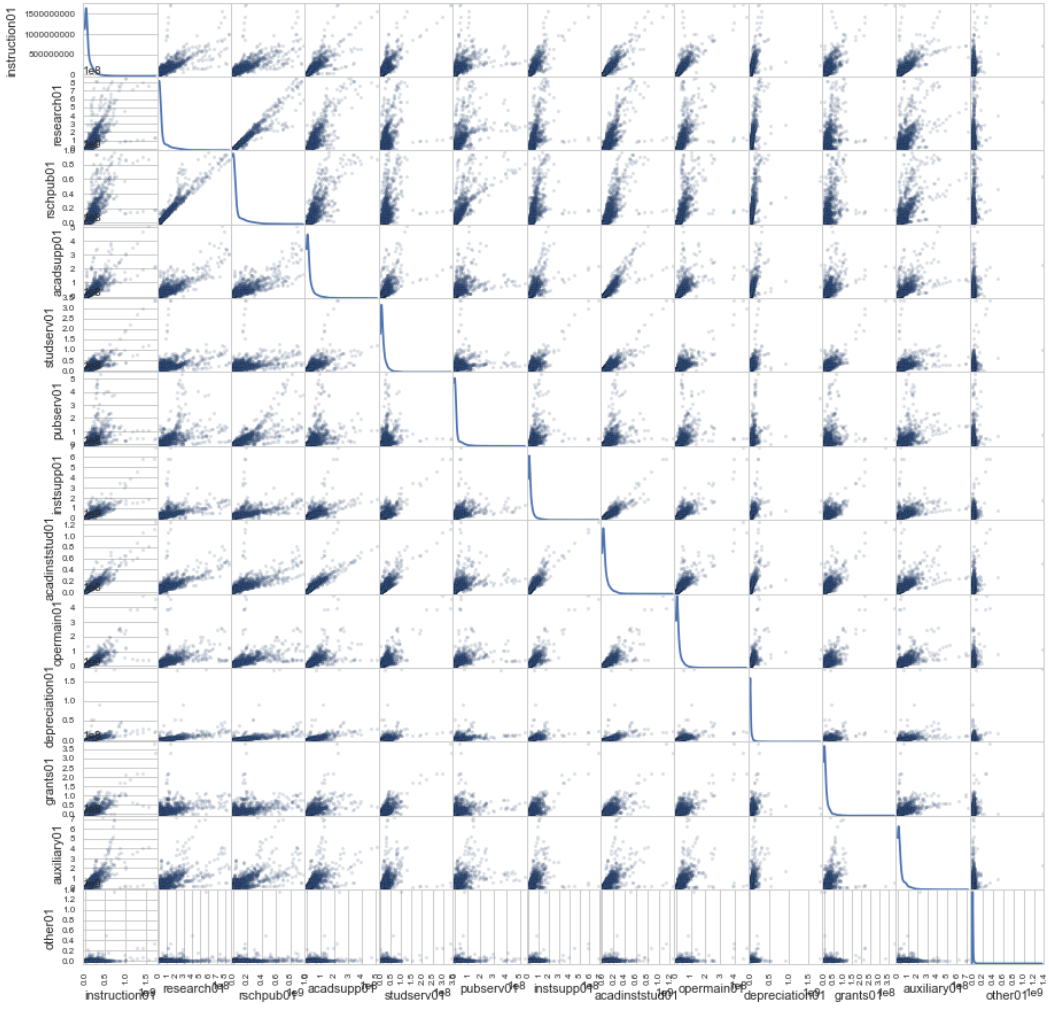

Fig 2 Scatter Matrix of features

IMPLEMENTATION

Loading the dataset with python pandas library, and the dataset can be stored as a data frame. After removing useless feature and data points, the training set and the testing set was randomly selected at 3:1 ration.

The metrics chosen were written as functions calculated through basic formula based on the residual sum of squared error and built-in R2 score function of scikit learn's base model.

Before PCA, standardizing the dataset, and transformed dataset will be used as new features to predict the target (Tipping & Bishop ,1999).

PCA with a default setting, which has the same number of features, avoid whitening to keep information of the transformed data, set singular value decomposition solver to auto. The result shows only 12 dimensions can explain the total variance of the dataset. Therefore the number of components is set as 12 to reduce the dimensionality.

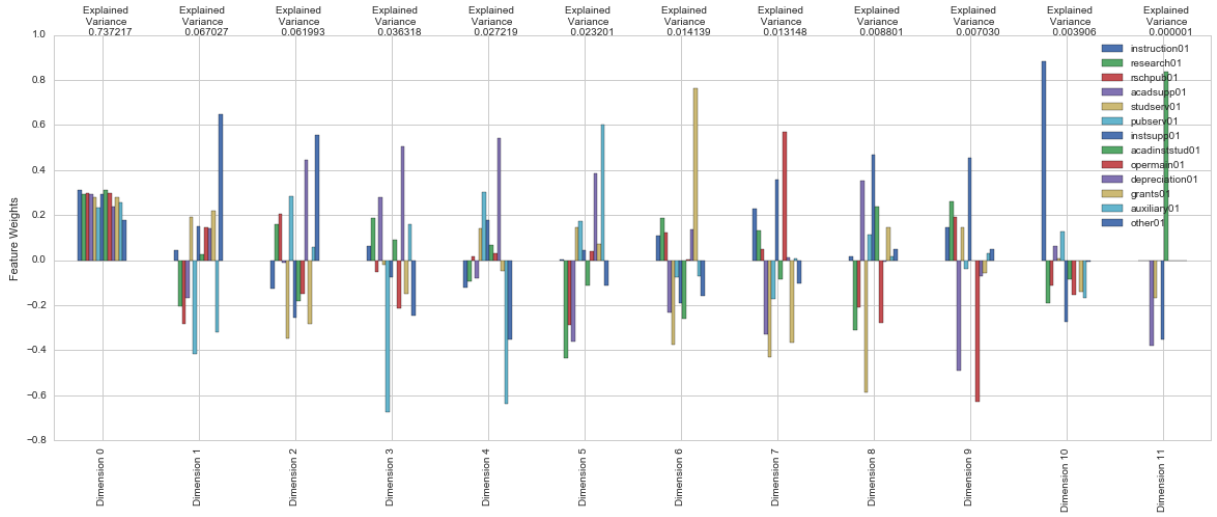

Fig. 3. Weight of each original features in each principal components

From Fig. 3 , we can see the weight difference between each feature. In the first principal component, where other and public service are weighted less than other features. The second principal component consists of a higher weight on other but negative weight on public service. The weight is a relative value, and negative weight indicates the feature is relatively less important in this component.

Comparison of candidate models through the mean of metrics calculated by k-fold cross-validation. The project uses 10-fold cross-validation. Models are from scikit-learn's library, which is a widely used library that contains various commonly used base models. Chosen models are listed and explained shortly below.

K-Nearest Neighbor Regressor (Peterson, 2009): The model estimates value by the closest data points, and the model costs less with minimal tuning.

Decision Tree Regressor (Breiman, Friedman, Stone & Olshen, 1984 ; Friedman, Hastie & Tibshirani, 2001): The model is easy to implement and also performs very fast. Once trained the model predicts very quick. The model gives a clear structure of how it makes a prediction, but the downside is the prediction may lack interpretation.

Support Vector Regressor (Smola & Schölkopf, 2004): The model is also a commonly used model. SVR has solid founding theory, less prone to over-fitting, and need less tuning

Multiple Layer Perceptron Regressor (Glorot & Bengio, 2010): The model uses the structure of a neural network and is regaining popularity due to numerous applications on voice and image recognition. The model consists 3 or multiple layers and each layer contains perceptrons which follow their activation functions to decide whether to send an output or not. The neural network model is easier to conceptualize and has an enormous amount of related research, but to outperform other modern models, it requires more tuning and usually harder to train. However, it can be used to tackle a lot of problem without too much understanding.

The following two algorithms use multiple trees and combine the outputs.

Random Forest Regressor (Breiman, 2001): The model creates multiple decision trees and trains with data through bootstrap sampling. Every node on the branch will randomly choose a small amount of the features. The trees will be tested with data not sampled. This method can avoid over-fitting to training data by its randomness. The setting has ten decision trees as estimators with no maximum depth and considers all the features when looking for the best split. No maximum for the leaf node and the minimum samples leaf is one.

Gradient Boosting Regressor (Friedman, 2001): The model builds decision trees sequentially to make a stronger model. It uses the later model to correct the previous model. The process can take longer than Random Forest, but the result shows better performance than Random Forest. On the contrast, Gradient Boosting is more prone to over-fitting. The setting used includes 100 base decision trees, learning rate at 0.1, and the maximum depth of each tree is 3. No maximum leaf node is set, and minimum samples leaf is one.

The model performance evaluation uses the metrics chosen combining with k-fold cross validation within training set. The model with the best performance will go to grid search stage for tuning.

REFINEMENT

The initial performance of models using transformed data with 10-fold validation are as following:

	Decision Tree	Support Vector Machine	Multi-Layer Perceptron	K-Nearest Neighbor	Random Forest	Gradient Boosting	Tuned K-NN	AdaBoost Tuned K-NN
AIC	5246.72	6050.15	5528.30	4871.94	5087.81	5024.95	4737.43	4737.48
BIC	5293.55	6096.97	5575.66	4918.76	5134.63	5071.77	4784.25	4784.31
Adj R2	0.8737	-0.03	0.7414	0.9573	0.9233	0.9352	0.9705	0.9704
Time (sec)	0.5318	5.2259	18.2670	0.3367	2.8348	2.8106	0.6512	35.5534

Table 3 Report of metrics and execution time when fitted to transformed data

K-nearest neighbor regressor stands out on all metric and will go through grid search for fine tuning. Comparing with the default K-Nearest Neighbor, tuned model performs better on all three metrics (AIC: 8.41%, BIC: 8.41%, Adjusted R2: 4.36%).

Tuned KNN model uses one closest neighbor for estimation, and the leaf size is 1. This setting indicates that the one closest data point the estimator can make prediction better than the situation using more samples. With lower leaf size the memory used should be less and the query speed should be faster. This fine-tuned model will be the base model for AdaBoost model to achieve higher accuracy.

AdaBoost implementation iterates the base estimator 30 times with 1.0 learning rate. The AdaBooster will iterate base estimator, the K-Nearest Neighbor model that just got tuned, for multiple times and assign the sample where an error is made heavier weight in the next iteration, and lighter weight for correctly predicted samples (Drucke, 1997).

AdaBoost model has similar performance as tuned KNN model but longer time consumption. Given the nature of AdaBoost is to iterate the estimator for multiple times, the time is 54.60 times of the tuned KNN model. Consider the cost efficiency of AdaBoost model; it doesn't seem to be an appealing method with this dataset.

IV. RESULTS

MODEL EVALUATION AND VALIDATION

With all variables retrieved from the original dataset, 13 features go through the process of unsupervised learning. PCA reduced the dimensionality to 12 dimensions. Using data transformed by PCA as the new predicting features, K-Nearest Neighbor has the best performance among all candidate models. The after-tuned model has the parameters as following:

KNeighborsRegressor(algorithm='auto', leaf_size=1, metric='minkowski', metric_params=None, n_jobs=1, n_neighbors=1, p=2, weights='uniform')

The tuned model uses uniform weights for all neighbors, and choose the one closest data point to make an estimate, and metric= minkowski and p=2 meaning the model uses the vertical distance between points as metric.

AIC and BIC evaluate the information loss and apply a penalty to additional parameter added. The final model used the transformation of original data, but the reduction of dimensionality makes it reserves relatively more information with fewer parameters. As reported in Refinement, K-Nearest Neighbor has already been the best performance among all candidate models without tuning. Using the tuned KNN model as the base estimator, AdaBoost's performance was also tested but not exceeding the efficiency of tuned KNN model. Training the tuned KNN model with full training dataset and tested on the testing set.

The AIC is 15405.52, BIC is15466.80, and adjusted R2 is 0.9807. Comparing with the benchmark, AIC is 8.4143% lower, BIC is 8.4114% lower, and Adjusted R2 is 4.3630% higher.

JUSTIFICATION

The tuned KNN model improves the prediction performance in a unneglectable level.

According to Bayes Factors (Kass & Raftery, 1995), the strength of the evidence against the models with higher BIC is strong when **ΔBIC** is between 6 to 10. The final KNN model appears to have significant difference against the benchmark, and the adjusted R2 score is also better than the benchmark.

To understand why tuned KNN model fitted to transformed data performs better than unturned KNN model fitted to original data, the step-by-step analysis of process is below.

PCA transformation reduced the dimensionality, and a default KNN model fitted to transformed dataset has higher adjusted R2 score than benchmark score. Tuned KNN model uses one closest neighbor instead of three nearest neighbors for estimation. Using less nearest neighbors when making prediction may allow tuned KNN model to avoid over-fitting to the training set. On the other hand, the model still makes more precise prediction than KNN model using more neighbors.

V. CONCLUSION

FREE-FORM VISUALIZATION

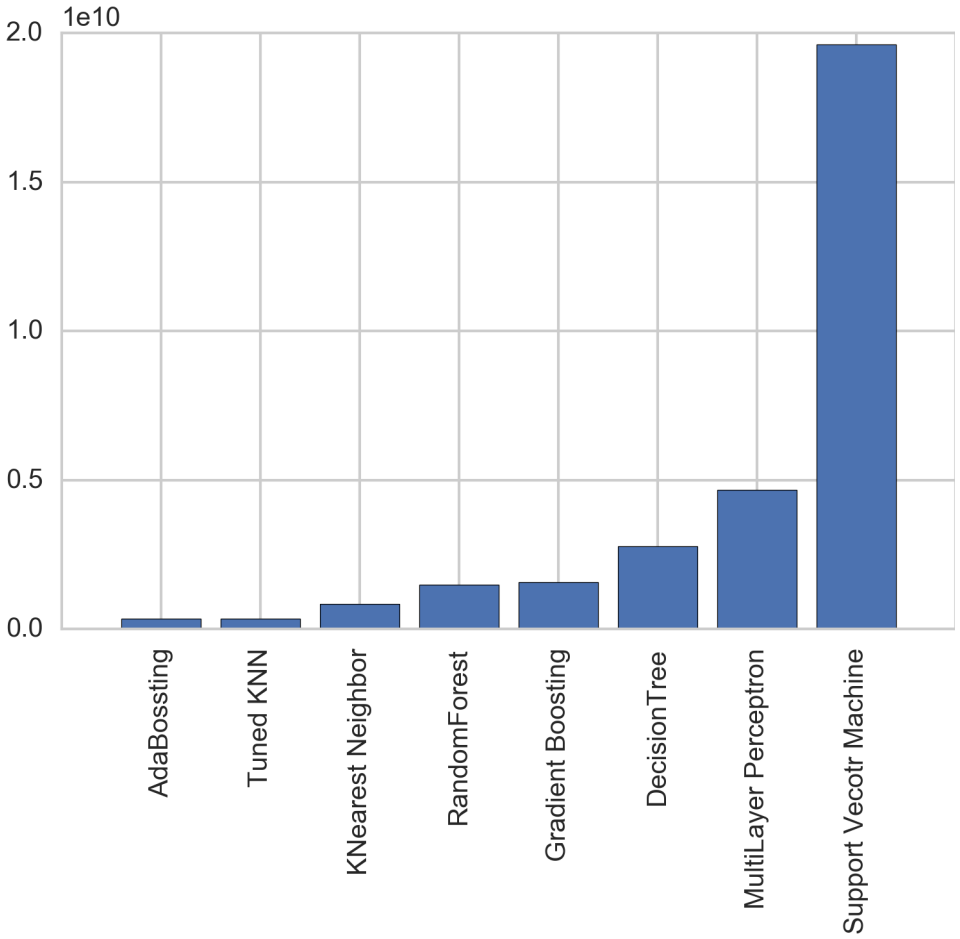

Fig. 4 Residual Sum of Square Comparison between models

Fig. 4 shows each model and the sum of squared difference between prediction and actual completion number.

From the gra,ph it shows that tuned KNN leads with the least RSS, followed by AdaBoost model, and SVM has far more error than other candidates.

REFLECTION

This project targeted to find the principal components of the expenditure of higher education institutes and used them as a better predictor of performance, and the goal of this project is predicting the total number of degree, certificate, and award completion. The project successfully to find the expenditure combination that serves as better predictors with principal component analysis and creates a model that performs better than the benchmark.

Selecting the features for analysis is the first obstacle in this project. While some institution runs a hospital and others don't, there are some fields excluded for not relevant to all dataset. Although the project eventually excluded data points with missing data, imputation is an alternative method. However, exclusion was chosen to avoid changing the statistics of the dataset. Choosing candidate models is also a struggling; I selected several base models solely for personal practicing. Gradient Boosting Decision Trees and Random Forests are both well-known ensemble method, but the characters of these two models might not be the best option while dealing with this dataset.

Delta Cost Project provides a dataset with quite a good shape, and simple analysis can bring a lot of information. During the contemplation of this project, several ideas using the same dataset pop up. The project itself also has room for improvement.

IMPROVEMENT

While predicting the performance of an institute, rather than using expenditure of current year, previous years also accountable for the contribution of any degree, certificate, awards can take longer than a year to be completed. On the execution part, this project may need a comparison with the version includes all sub-categories of expenditure. Missing value treatment can also be reviewed again in this case. Feature selection process can also be refined in order to pick up relevant information for making the prediction.

PCA successfully reduced dimensionality and provides the principal components. With further clustering, we might be able to categorize numerous education institutes into groups and find the high-perform expenditure pattern for each cluster. Institute can use the result to determine which group they are in or want to identify them, and using the spending pattern for future investment reference.

Models comparison includes some models that require tuning to perform better. For a model collection including those models, it might be better to compare after proper tuning. Initial plan also includes xgboost model (Chen & He ,2015), but it is removed after research on the dataset and the advantage of models.

In the process of this project, coding becomes a constraint and costs much time. With more practice, the code can be cleaner.

Reference

Akaike Information Criterion. (n.d.). In *Wikipedia* Retrieved November 3, 2016 from https://en.wikipedia.org/wiki/Akaike_information_criterion

Andreas, M. B., (2012, October 10). What is the AIC formula [Msg 1]. Message posted to https://www.researchgate.net/post/What_is_the_AIC_formula

Bayesian Information Criterion. (n.d.). In *Wikipedia* Retrieved November 3, 2016 from https://en.wikipedia.org/wiki/Bayesian_information_criterion

Bozdogan, H. (1987). Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions. *Psychometrika*, *52*(3), 345-370

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

Chen, T., & He, T. (2015). xgboost: eXtreme Gradient Boosting. *R package version 0.4-2*.

Drucker, H. (1997, July). Improving regressors using boosting techniques. In *ICML* (Vol. 97, pp. 107-115).

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning* (Vol. 1). Springer, Berlin: Springer series in statistics.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232

Glorot, X., & Bengio, Y. (2010, May). Understanding the difficulty of training deep feedforward neural networks. In *Aistats* (Vol. 9, pp. 249-256).

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, *3*(Mar), 1157-1182.

Jongbloed, B., & Vossensteyn, H. (2001). Keeping up performances: An international survey of performance-based funding in higher education. *Journal of Higher Education Policy and Management*, *23*(2), 127-145.

Karen, *Outlier: To Drop or Not to Drop* Retrieved from http://www.theanalysisfactor.com/outliers-to-drop-or-not-to-drop/

Kass, R. E., & Raftery, A. E. (1995). Bayes factors. *Journal of the american statistical association*, *90*(430), 773-795..

Kuha, J. (2004). AIC and BIC comparisons of assumptions and performance. *Sociological Methods & Research*, *33*(2), 188-229.

Larget, B (2003, April 7). *Cp, AIC, and BIC*. [PDF Document]. Retrieved from http://www.stat.wisc.edu/courses/st333-larget/aic.pdf

National Conference of State Legislatures (2015, July 31). *Performance-based Funding for Higher Education* Retrieved from http://www.ncsl.org/research/education/performance-funding.aspx

Peterson, L. E. (2009). K-nearest neighbor. *Scholarpedia*, *4*(2), 1883.

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, *14*(3), 199-222.

Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *61*(3), 611-622.

United States Department of Education. (2016). *College Affordability and Completion* Retrieved from http://www.ed.gov/college-completion

Vrieze, S. I. (2012). Model selection and psychological theory: a discussion of the differences between the Akaike information criterion (AIC) and the Bayesian information criterion (BIC). *Psychological methods*, *17*(2), 228.

Williams, R. (2015). *Missing Data Part 1: Overview, Traditional Methods*. http://www3.nd.edu/~rwilliam/stats2/l12.pdf