MS&E 226 Project Part 1

Jennifer Pham

1 Background

1.1 About the data

The selected data set is the reduced version of the College Scorecard from the US Department of Education. The College Scorecard contains aggregate data for institutions across the United States such as enrollment, costs, student demographics and financial outcomes. It contains 42 variables and is representative of 7,804 unique schools. The data is collected by the Integrated Postsecondary Education Data System, the National Students Loan Data System, and the Department of the Treasury, so there should not be any issues regarding the collection or accuracy of the data. A few covariates were missing a sizable amount of data, which led to concerns on how to process the data to build the best model possible, which is discussed in the following sections.

1.2 Response Variables

1.2.1 Continuous Response Variable

The continuous response variable for the prediction task is the median earnings of students working after 10 years. Often one chooses to attend a post-secondary education to increase one's earning potential. The choice in institution can make all the difference – by looking at characteristics such as average test scores, debt, type (public/private). One often gauges the desirability of an institutions based its characteristics such as class size or location, which can ultimate predict one's potential future earnings. This variable was chosen as it was one of the continuous variables available with not many missing data points.

1.2.2 Binary Response Variable

As great of an investment post-secondary education is, the debt that is accumulated can be a major detractor of pursuing a degree. The binary response variable will whether an institution has high debt or low debt levels. Similar to predicting future earnings, we can use various factors to predict how much debt a student will be in post-degree. In order to create a binary variable, institutions will be classified as either high debt or a low debt based on a threshold set at the median debt across all institutions in the data set.

1.3 Data Cleaning

Some cleaning was done on the data set before the 80-20 split. The initial cleaning included removing school names and city. These features did not contribute to any predictive analysis. Though cities can be an indicator of wealth and debt levels, the unique state and cities in the data set that may be difficult to use in the prediction, namely in the cross-validation step. In addition, some of institution location information is reflected in other covariates such as state and local. The covariate corresponding to median debt of completers expressed in 10-year monthly payments was dropped because of collinarity with median debt of completers. Variables such as degrees awarded, control, and locale were factored to ensure they were interpreted as categorical variables.

1.3.1 Missing Values

This data set contains many missing values encoded as "NULL" and "PrivacySuppressed". For the purposes of this analysis, PrivacySuppressed was be treated as a missing value. In order to avoid complications regarding missing values, covariates with more than 90% missing values were removed from the data set. This was done to remove missing values and preserve as many of the observations as possible. This left 364 (~7%) observations with incomplete records. Removing observations can potentionally lead to selection bias. From observation, some of the missing values are missing at random, such as the 'Distance only' missing values are more likely to be missing among 'Not Classified' degree institutions. To handle these missing values, our analysis will be limited to institutions that classify what degree is provided (associate, bachelors, certificate, graduate), so 'Not Classified' degrees will be removed (which only made up a small portion of the total observations). The remaining missing values appear to be missing completely at random since these observations are not related to their value, so they were removed from the data set. These removals should not contribute to major selection bias. By removing missing values, modeling and prediction techniques demonstrated in class can be applied to this data. The final training set has 23 columns and 3,745 observations in the data set.

1.3.2 Highly correlated covariates

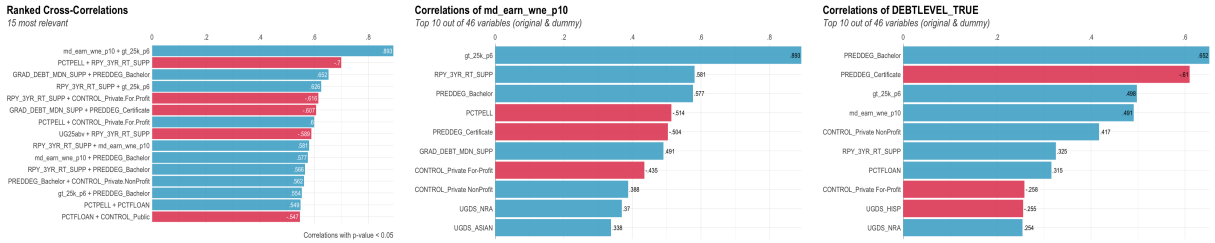

Figure 1: Correlation Coefficients between covariates/covariates and covariates/response variables

Some covariates that are strongly correlated to the continuous response variable include: percentage of students earning 25,000 dollars after 6 years, 3-year repayment rate, Bachelor granting institutions, percentage of Pell Grant recipients, and Certificate granting institutions. Some covariates that are strongly correlated to the binary response variable include: Bachelor granting institutions, Certificate granting institutions, percentage of students earning $25k after 6 years, median wage after 10 years, and Private Non-profit Institutions. These covariates may be important predictors for our models in later sections.

1.4 What questions can be answered utilizing this data set?

This data set can help us answer questions relating to degree-granting institutions (certificate, associate, bachelors, graduate). The goal of this project is to answer the question "What is the expected earnings for a given institution with certain characteristics?" On the other side, this project also seeks to answer the question "Is a given institution an high debt or low debt institution?" Some future questions that might motivate future projects is looking at private vs. public institutions and how they differ in their earning and debt. What makes this data unique are the range of covariates that can be utilized to answer these questions pressing to the current field of education. In the next few sections, models will be built that will help predict the earnings and debt of an students at a given institution post-graduate utilizng some characteristics, such as student body demographics, population, type of institution, and more.

2 Prediction

2.1 Regression

2.1.1 Baseline Model

As the models are built, having a baseline models can be a good comparison point. To assess the baseline model (and subsequent models) 10-fold cross validation was used which reports the Root Mean Squared Error (RMSE). The first baseline model only includes the intercept, or in other words predicts the mean earnings ($33653) for every observation. The first baseline model resulted in a RMSE of 11984.22. The second baseline model is a linear model that includes all the covariates. This resulted in a RMSE of 4788.234. The all covariate model shows a substantial improvement over the mean-only model.

2.1.2 Forward/Backward Stepwise Regression

Forward and Backward Stepwise Regression was also run on the transformed data. The forward stepwise regression resulted in a RMSE of 4761.493. The forward model is similar to our last model and utilized all but two covariates from the previous model. We see an improvement from the OLS w/ transformed variables. The backward stepwise regression resulted in a RMSE of 11992.89. The backward model produced a model that only included the intercept. As discussed in the 'Baseline Model' section, this was deemed not a good model compared to some other options.

2.1.3 Ridge & Lasso Regression

Another prediction method that was used to predict future income was Ridge and Lasso regression. Cross validation was utilized to select the best lambda value. Cross validation was performed to assess the performance of the models. The RMSE for Ridge Regression is 4667.46. The RMSE for Lasso Regression is 4674.42. This is an improvement from our baseline model.

2.1.4 Summary of Regression Models

model	RMSE
Baseline OLS - intercept only	11984.220
Baseline OLS - all covariate	4788.234
Forward Stepwise	4796.309
Backward Stepwise	4799.064
Ridge	4688.790
Lasso	4697.589

As observed the best model based on RMSE is the ridge regression model with a RMSE of 4688.79. Though ridge regression provided the lowest RMSE, this may not be the most favorable model due to it being less interpretable than OLS. However, ridge reduced the RMSE a noticeable amount compared to other models so ridge is the best overall model. Models better than the baseline will not reduced the RMSE close to zero due to the noise inherent in the data set. A reduction of 100 in the RMSE translates the original units, thus is a substantial difference from the baseline which is the reason why ridge was chosen as the best regression model. Since cross validation was used to assess the models, the RMSE is a good estimate for the test error since cross validation partitions the training set and retrains the model on 9 of the partitions and tests on the 10th.

2.2 Classification

For the classification task, a logistic regression model was used. As discussed in the 'Binary Response Variable' section, the classification task will identity whether certain institutions are classified as Higher Debt institutions or Lower Debt institutions. The cutoff for Higher debt was debt above the median, and values lower than the median were classified as Lower debt. The 0-1 cross-validation was used to assess the models.

2.3 Baseline model

The baseline model was a logistic regression model with all covariates. This model resulted in a 10.5% 0-1 error from 10-fold cross validation. 0-1 loss is defined as the number of incorrectly categorized observations.

2.4 Logistic w/ Transformed Variables

Covariates that demonstrated a long tail was transformed. A logistic regression was run on this transformed data. This led to an 0-1 error rate of 10.3% from 10-fold cross validation.

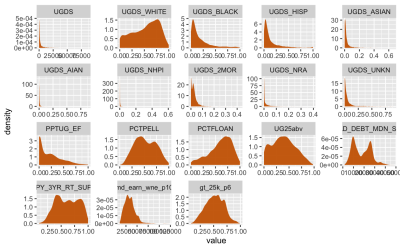

Figure 2: Histograms of covariates

2.5 Foward/Backward Regression

Model selection can assist with identifying the most significant covariates. In this case, the backward and forward stepwise regression resulted in an error rate of 10.1% and 10.3% respectively.

2.5.1 Summary of Regression Models

Model	X0.1.error.rate
Baseline - all covariate	10.5
Log-Transformed	10.3
Forward Stepwise	10.1
Backward Stepwise	10.3

As seen, Forward Stepwise Regression produced the best model with the lowest error One thing to note is the 4 models produced very similar error rates and the baseline model had a low starting error rate. Forward Stepwise may be no better than the baseline, however due to fewer covariates in the model, may actually be favorable because its interpretability and simplicity compared to the baseline. This error rate is a good predictor of the test error because cross validation was used to assess each model.

3 Appendix

3.1 Code for Best Models

```
# Ridge Regression for Regression Task
X = model.matrix(md_earn_wne_p10 ~ ., stan_train)
Y = stan_train$md_earn_wne_p10

cv.outr = cv.glmnet(X, Y, alpha = 0, lambda = lambdas)
bestlamr = cv.outr$lambda.min
fm.ridge = glmnet(X, Y, alpha = 0, lambda = bestlamr, thresh = 1e-12)

cvFit(fm.ridge, data = X, y = Y, K = 10)  #CV RMSE = 4688.79
```

```
# Forward Stepwise Logistic Regression for Classification
set.seed(1)
fm.lower.class = glm(formula = DEBTLEVEL ~ 1, family = "binomial",
    data = train_class)
fm.upper.class = glm(formula = DEBTLEVEL ~ ., family = "binomial",
    data = train_class)
forward <- step(fm.lower.class, scope = list(lower = fm.lower.class,
    upper = fm.upper.class), direction = "forward", trace = FALSE)
# md_earn_wne_p10 ~ gt_25k_p6 + UGDS_ASIAN + PREDDEG + STABBR +
# PPTUG_EF + GRAD_DEBT_MDN_SUPP + PCTFLOAN + CONTROL +
# UGDS_WHITE + UGDS_NRA + DISTANCEONLY + UG25abv + PCTPELL +
# UGDS_2MOR + RPY_3YR_RT_SUPP

k = 10
f <- createFolds(y = train_class_log$DEBTLEVEL, k)
train_fold <- function(i) {
    train_class_log[-unlist(f[i]), ]
}
test_fold <- function(i) {
    train_class_log[unlist(f[i]), ]
}

accuracy <- c()

for (i in 1:k) {
    b_model = glm(formula = DEBTLEVEL ~ PREDDEG + PCTFLOAN + STABBR +
        md_earn_wne_p10 + CONTROL + PCTPELL + RPY_3YR_RT_SUPP + UG25abv +
        PPTUG_EF + UGDS_2MOR + gt_25k_p6 + DISTANCEONLY + UGDS_NRA +
        UGDS_ASIAN + UGDS + UGDS_NHPI, family = "binomial", data = train_fold(i))
    predict_result <- predict(b_model, newdata = test_fold(i), type = "response")
    predict_logit <- ifelse(predict_result >= 0.5, 1, 0)
    confusion <- table(predict_logit, test_fold(i)$DEBTLEVEL)
    accuracy[i] = (confusion[1, 1] + confusion[2, 2])/dim(test_fold(i))[1]
}

1 - mean(accuracy)  # CV 0-1 Loss 0.1006718
```