Employing Machine Learning Methods in Predicting Incomes of Recent College Graduates

Using a principled machine-learning approach, we predict recent college graduates' earnings using data from the College Scorecard. These predictions are estimated using elastic net regularization and the random forest algorithm, regression-based methods adept at producing parsimonious statistical models and reducing bias. Our results support the predictive capabilities of institutional characteristics like school classification, overall debt repayment rates and family income on recent graduate earnings.

QUESTIONS

1. What institutional/program level variables are most predictive of median first-year earnings?
2. What are the patterns of association (positive/negative) for these variables?

DATA

College Scorecard

- 2019–2020
- Includes over 2,000 variables for 6,700 accredited higher education institutions (HEIs) in the U.S.
- Significant portions of the data are privacy suppressed

American Community Survey

- 2015–2019
- Geography of interest: County level (using FIPS)
- Matching HEIs with unique county FIPS codes
- Helps recover information lost from missing data in the Scorecard

UF | College of Education
UNIVERSITY *of* FLORIDA

WORKFLOW

Using a Tidymodels workflow (Kuhn and Silge, 2022) in the R language, we preprocess our data and perform analyses following a standardized recipe structure that allows for transparent parameter tuning and model adjustments. Below is example code for our random forest model.

Setup

```
## libraries
library(tidyverse)
library(tidymodels)
library(parallel)
library(doParallel)
library(vip)
```

Define regression formula

```
## regress median earnings after one year
## against all available variables
formula <- ("earn_mdn_hi_1yr ~ .")
```

Preprocess data via recipe

```
## read in data
df <- read_rds("cleaned_data.RDS")
## preprocess using recipe
recipe <- recipe(df, formula) |>
   step_zv(all_predictors()) |>
   step_nzv(all_numeric_predictors()) |>
   update_role(-opeid6,
               new_role = "predictor") |>
   update_role(earn_mdn_hi_1yr,
               new_role = "outcome") |>
   update_role(opeid6,
               new_role = "id variable") |>
   step_naomit(all_predictors()) |>
   step_other(all_nominal_predictors(),
              threshold = .005) |>
   step_dummy(all_nominal_predictors()) |>
   step_zv(all_predictors(),
           skip = TRUE) |>
   step_nzv(all_numeric_predictors(),
            skip = TRUE) |>
   step_normalize(all_numeric_predictors()) |>
   step_corr(all_numeric_predictors(),
             skip = TRUE,
             threshold = .95)
```

Set workflow and add recipe/model

```
## set workflow set up above
prep <- prep(recipe)
```

Set tuning grid/parallelization

```
## set tuning specification
tune_spec <- rand_forest(mtry = tune(),
                         trees = 1000,
                         min_n = tune()) |>
set_mode("regression") |>
set_engine("ranger")
## set tuning workflow
tune_wf <- workflow() |>
   add_recipe(recipe) |>
   add_model(tune_spec)
## set number of available cores
registerDoParallel(cores = detectCores())
## use 10-fold crossvalidation to tune
tune_res <- tune_grid(tune_wf,
                      resamples = vfold_cv(df, v = 10),
                      grid = 20)
```

Run method with defined workflow and tuning grid

```
## select best tuning parameters based on RMSE
best_rmse <- select_best(tune_res,
                         "rmse")
## finalize the model using best tuning spec
final_rf <- finalize_model(tune_spec,
                           best_rmse)
## compute variable importance
vi_final_rf <- final_rf |>
   set_engine("ranger",
              importance = "permutation") |>
    fit(earn_mdn_hi_1yr ~ .,
        data = juice(prep)) |>
vi(scale=TRUE)
```

MODELS

Elastic net regularization (Zou & Hastie, 2005)

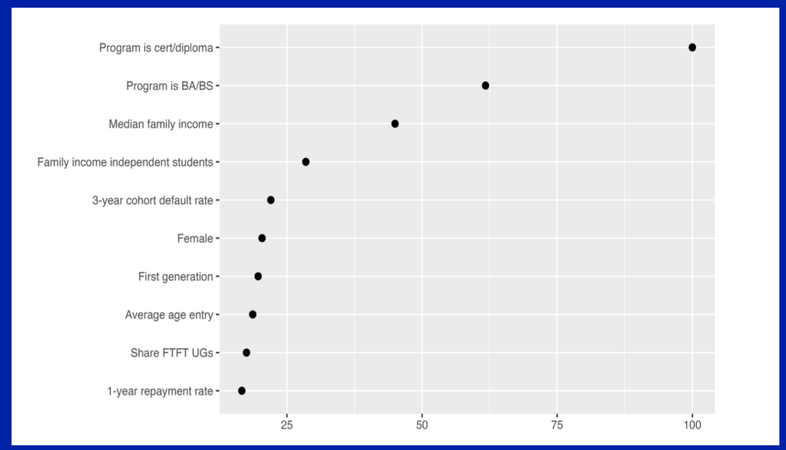
- Combines LASSO/ridge regression penalties to mitigate issues of bias and variance introduction in your statistical model

Random forest

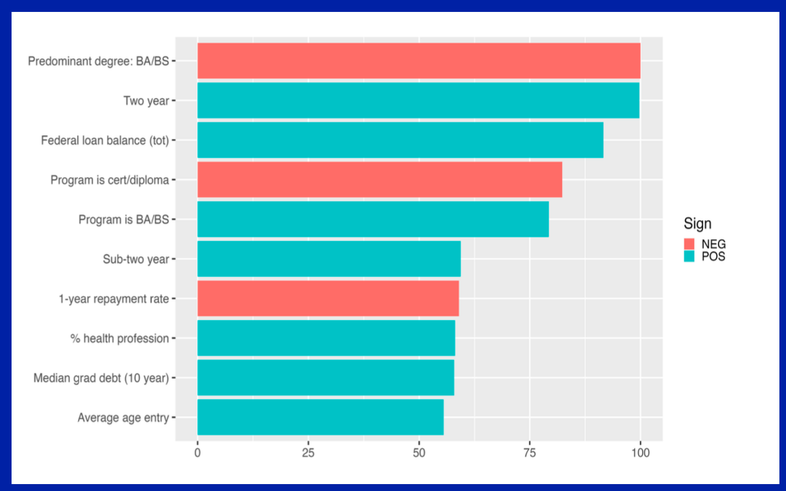
- Randomly samples covariates in a regression model, splits observations on covariates to maximize reduction of RMSE

RESULTS

RQ1 (Variable Importance)



RQ2 (Elastic Net Model)



AUTHORS

Olivia Morales, University of Florida
Benjamin Skinner, University of Florida
William Doyle, Vanderbilt University