STATS504 Assignment 6: Graduation

December 9 2022

1 Introduction

The admissions criteria of an educational institution are very important to both the student and the institution. Admitting students that have fewer chances of graduating will harm both the students and the institution as students are economically punished for paying tuition without the benefit of a degree, and the intuition could be overshadowed by the low graduation rate. Therefore, the admission office from a higher education institution is interested in understanding how the admission grade impacts graduation, and further adjusting admissions requirements accordingly.

With data collected related to student enrollment in undergraduate courses between 2008 to 2019, this study aims to determine and quantify the causal effect of students' grades required for admissions on graduation. Moreover, we will investigate student features and other external factors, such as age at enrollment and current inflation rates, that possibly contribute to varying admission grades.

Concerning the questions of interest, there are two main findings that are worth mentioning. Firstly, there is a significant causal effect of the admission grade on graduation, a higher grade will bring a higher probability of graduation. Secondly, the student's previous education qualifications and corresponding grade, age at enrollment, student's father's occupation and previous qualification are the top 5 features that affect student's admission grade and therefore affect whether the student is in the treatment group or control group in our model.

2 Method

Our client, the director of admissions at Polytechnic Institute of Portalegre, provided us with cleaned student enrollment data between 2008 to 2019 from students studying for various degrees. The primary purpose of this study is to quantify the causal effect of students' admission grades on graduation. As a secondary goal, we hope to identify important features, including students' demographic, socio-economic and academic information as well as external factors, that potentially influence the admission grade.

For the causal analysis, our exposure of interest (or treatment variable) is the admission grade and the outcome is whether a student graduated or dropout. Since this is an observational study rather than a randomized controlled experiment, causation cannot be assigned without accounting for confounding between variables. A "confounder" refers to a pre-treatment covariate that influences both exposure and outcome. Therefore, it can result in confounding bias which distorts the true relationship between the treatment and outcome by mixing the effects of additional sets of variables, leading to incorrect conclusions if we compare the treatment and control populations directly. For instance, the gender of students is a confounder that can impact both the admission grades and whether they could finish their degree, which makes it less accurate to estimate the association only between admission grades and graduation if we do not consider confounding.

To address the issue mentioned above, we decide to use the Propensity Score Method to estimate the population average treatment effect that changing the admission grade has on graduation. Firstly, we assign propensity scores to weight observations, which eliminates the impact of confounders between the treatment and outcome by balancing the measured covariates. This is to say, given the propensity score, the distribution of measured covariates for the treatment group is similar between the treated and control groups. The advantage of the propensity score method is that it reduces dimensionality by summarizing all covariates into a single score, which eases the computation complexity. Moreover, it allows us to have a good treatment effect estimation in cases the treatment is not randomized. This

greatly relaxes the conditions for similar modeling, allowing us to conduct quantitative analysis on more scenarios. However, this method can be less valid if we do not have sufficient overlap between observations or if there are some unknown confounders.

There are several ways to estimate the propensity score, including logistic regression and gradient boosting models. The potential problem with logistic regression for estimating weights is that it can be challenging to get a good fit under the linearity assumption. The boosting method has benefits that no linearity assumptions need to be met and variable selection is automatic. Therefore, the non-parametric boosting method is used to estimate the propensity score. To acquire a reliable estimation, we choose an easier way to consider the treatment (admission grade) as a binary variable. As for the final outcome analysis, we use a weighted t-test to determine whether the casual effect is statistically significant or not.

In terms of data pre-processing, we decided to drop the "Enrolled" level in our outcome (graduation) because we think it's unrelated to our research question. We will then focus on the "Graduate" and "Dropout" levels to study the causal relationship. In the raw data, the treatment variable "admission grade" is a continuous variable ranging from 0 to 200. The analysis method we choose will require the treatment variable to be a binary variable (1 or 0) to indicate whether it's in the treatment group or not. Therefore, we decide to cut the "admission grade" by its median, converting all values larger or equal to its median to 1 and all values smaller than the median to 0. Split by median is the simplest but most reliable method to convert to binary. In this way, we acquire the binary treatment.

Not all of the features in the data set should be included to estimate the propensity score, but only those confounders, or so-called pre-treatment covariates. A variable is a confounder if it associates with both the admission grade and whether the student is graduated or dropped out. We will use those confounders to estimate the propensity score, thus to re-weight the observations. We consider those variables which are determined before admissions, such as gender, the father's and mother's qualifications, and include them in the boosting model for propensity score estimation. For variables, including tuition fees, course, course time, whether a scholarship holder and application-related and curricular-related variables, they should not be included as they are determined after admission and could only impact the outcome variable (graduation).

3 Result

3.1 Data Overview

The data was collected from several disjoint databases related to student enrollment in undergraduate courses between 2008 to 2019 by a higher education institution, which contains 4424 rows of observations and 37 variables. Each row of observation represents a student's academic and demographic information as well as external social-economic factors known at the time of student enrollment, and corresponding graduation status (our target variable) at the end of the normal duration of the course. There are no missing values for any of the variables, which could save us one step in dealing with missing values.

For the purpose of analyzing the causal effect of students' admission grades on graduation, we dropped rows in which students were still enrolled in the course and there were 3630 rows remaining. The admission grade is a continuous treatment variable, so a median cutoff (126.5) was set to transform it into a binary treatment variable, above the median denoted by 1 and otherwise 0. 17 pre-treatment covariates were selected from all 35 variables except for our treatment and outcome variables as we discussed in the method section.

See more summary statistics for the pre-treatment covariates in Table 1. Note that for better display, only the levels with the highest percentage for 12 categorical covariates (if more than 5 levels) are summarized in this table. The p-value for weighted observations obtained after estimating the propensity score is also included in Table 1.

Covariate	Total ($n = 3630$)	Treatment ($n_1 = 1821$)	Control ($n_2 = 1809$)	p-value (unweighted)	p-value (weighted)
Grade of previous qualification	133.1(125, 140)	140(132, 147)	127(120, 133.1)	0.000	0.000
Unemployment rate(%)	11.1(9.4, 13.9)	11.1(9.4, 13.9)	11.1(9.4, 13.9)	0.941	0.905
Inflation rate(%)	1.4(0.3, 2.6)	1.4(0.3, 2.6)	1.4(0.3, 2.6)	0.501	0.931
GDP	0.32(-1.7, 1.79)	0.32(-1.7, 1.74)	0.32(-1.7, 1.79)	0.049	0.424
Age at enrollment (years)	20(19,25)	20(18, 25)	20(19, 26)	0.084	0.699
Marital status(%)	single:88.13	single:87.59	single:88.67	0.079	0.330
Previous qualification(%)	Secondary education:83.17	Secondary education:79.24	Secondary education:87.12	0.000	0.000
Nationality(%)	Portuguese:97.63	Portuguese:97.14	Portuguese:98.12	0.032	0.098
Mother's qualification(%)	12th year of schooling or eq.:23.83	12th year of schooling or eq.:24.38	4th/5th year of schooling or eq.:23.99	0.024	0.733
Father's qualification(%)	4th/5th year of schooling or eq.:27.82	4th/5th year of schooling or eq.:26.91	4th/5th year of schooling or eq.:28.75	0.027	0.810
Mother's occupation(%)	unskilled workers:36.17	unskilled workers:35.97	unskilled workers:36.37	0.029	0.762
Father's occupation(%)	unskilled workers:23.33	unskilled workers:23.17	unskilled workers:23.49	0.218	0.932
Whether displaced(1) or not(0)(%)	1:54.9 0:45.1	1:54.64 0:45.36	1:55.17 0:44.83	0.749	0.811
Whether with special educational needs(1) or not (0)(%)	1:1.1 0:98.9	1:0.77 0:99.23	1: 1.44 0:98.56	0.054	0.131
Whether with student debt(1) or not (0)(%)	1: 11.38 0:88.62	1:10.21 0:89.79	1:12.55 0:87.45	0.027	0.889
Gender (1: male, 0:female)(%)	1: 34.41 0:65.59	1:35.04 0:64.96	1:33.78 0:66.22	0.424	0.402
Whether international student(1)or not (0)(%)	1:2.37 0:97.63	1:2.86 0:97.14	1:1.88 0:98.12	0.053	0.170

Table 1: Median and interquartile range for numeric variables, or percent of each categorical variable, p-value under t or χ^2 tests between treatment/control of all pre-treatment covariates

3.2 Model Interpretation

Firstly, we calculated the propensity score which summarized all pre-treatment covariates into a single score using the gradient boosting method. Figure 1 illustrates the spread of the estimated propensity scores in the treatment and control groups, where "1" is for the control group (grade below median) and "2" is for the treatment group (grade above median). Overall, the model appears to achieve a good covariate balance across groups using propensity scores. We found there was a moderate overlap between the propensity score assignments, and approximately 3000 iterations were needed to achieve

the maximum balance between the groups (see Appendix page 4).

From Appendix (page 3), we can also get the covariates with the top 5 highest relative influence on whether the admission grade is above the median or not: the grade of the previous education qualification, previous education qualification, age at enrollment, father's occupation, and father's education qualification.

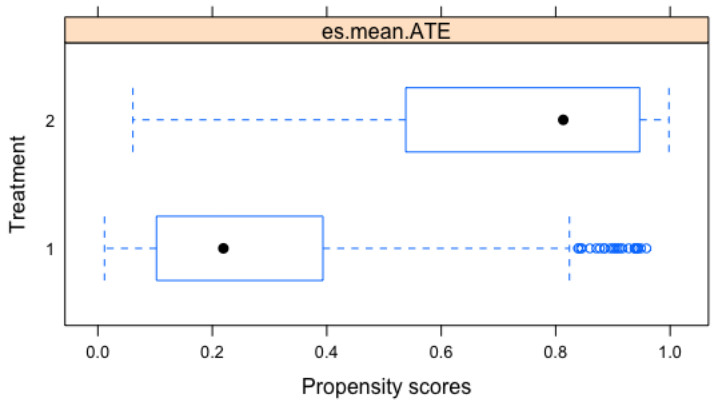


Figure 1: Boxplot of the propensity scores for the treatment and control cases

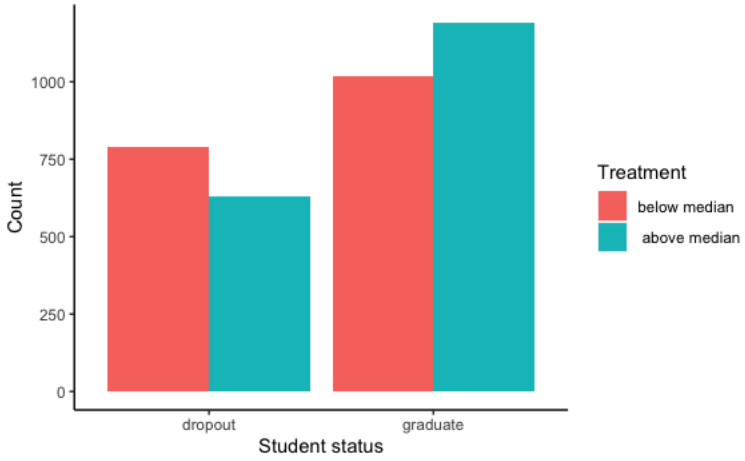


Figure 2: Histogram of students' graduation status by treated and control groups

We then performed the outcomes analysis on whether the admission grade impacts graduation assigning the propensity score weights. Interpretation of model results can be obtained from information included in Table 2. Overall, the analysis suggests that the admission grade is statistically significant at a $\alpha = 0.05$ significance level and the student's probability of graduation will increase by 0.24 for those with admission grades above the median.

Variable	Coefficient (95% CI)	p-value
Admission grades above median	0.24 (0.065, 0.416)	0.00726

Table 2: Variable name, estimated coefficient and p-value for the causal analysis using propensity scores

Finally, we looked at the distribution of our outcome across groups in Figure 2 after we'd done the causal analysis. We can find there are more students whose admissions grades are above the median

graduating successfully compared to those with lower admissions grades. Therefore, the admission office could set a higher grade required for admission in order to have more students who can graduate successfully.

4 Conclusion

In this study, we explored the causal relationship between a student's grade required at admission and whether a student can graduate successfully or not. By the propensity score method, we accomplished the goal of determining and quantifying the causal effect between admission grade and graduation, and also identifying important features that contribute to different admission grades.

There are two major findings that are worth noting. Firstly, could a higher admission grade influences whether a student can graduate successfully or not, and this causal effect is statistically signification. On average, the student's probability of graduation will increase by 0.24 for those with admission grades above the median compared to those with admission grades lower than the median. Secondly, the grade of the previous education qualification, previous education qualification, age at enrollment, father's occupation, and father's education qualification are the top 5 covariates with the highest relative importance when we estimated propensity scores, thus impacting the treatment variable.

Nevertheless, there still exist limitations to the current analytical framework. One limitation results from categorical variables with too many levels. The p-values from the χ^2 test in Table 1 might not be accurate as the sample can be too small from each level to suggest a significant difference. Moreover, we might get more interesting findings from models using a continuous treatment if time permits. Additionally, we want to use the Boosting propensity score estimation method to correct the distributions of each covariate in the treatment and control groups to be similar (mean, variance, skew, shape, etc). But this is rarely possible, so we might only restrict balance to mean and standard deviation. We might explore more methods to bring similar treatment and control groups if we have time. Last but not least, we only measure confounders in the dataset. However unmeasured confounders might exist as well, and therefore using propensity scores might not give an accurate estimation on the causal effect.

Appendix - EDA

In [1]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

In [2]:
```
# Read data
data = pd.read_csv("data.csv", sep = ";")
```

In [3]:
```
# Check the data shape
data.shape
```

Out[3]:
```
(4424, 37)
```

In [4]:
```
# Show the head of data
data.head()
```

Out[4]:

	Marital status	Application mode	Application order	Course	Daytime/evening attendance\t	Previous qualification	Previous qualification (grade)	Nacionality	Mother's qualification	Father's qualification	...
0	1	17	5	171	1	1	122.0	1	19	12	...
1	1	15	1	9254	1	1	160.0	1	1	3	...
2	1	1	5	9070	1	1	122.0	1	37	37	...
3	1	17	2	9773	1	1	122.0	1	38	37	...
4	2	39	1	8014	0	1	100.0	1	37	38	...

5 rows × 37 columns

In [5]:
```
# Show all the column names of data
data.columns
```

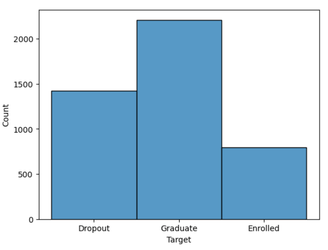
Out[5]:
```
Index(['Marital status', 'Application mode', 'Application order', 'Course',
       'Daytime/evening attendance\t', 'Previous qualification',
       'Previous qualification (grade)', 'Nacionality',
       'Mother's qualification', 'Father's qualification',
       'Mother's occupation', 'Father's occupation', 'Admission grade',
       'Displaced', 'Educational special needs', 'Debtor',
       'Tuition fees up to date', 'Gender', 'Scholarship holder',
       'Age at enrollment', 'International',
       'Curricular units 1st sem (credited)',
       'Curricular units 1st sem (enrolled)',
       'Curricular units 1st sem (evaluations)',
       'Curricular units 1st sem (approved)',
       'Curricular units 1st sem (grade)',
       'Curricular units 1st sem (without evaluations)',
       'Curricular units 2nd sem (credited)',
       'Curricular units 2nd sem (enrolled)',
       'Curricular units 2nd sem (evaluations)',
       'Curricular units 2nd sem (approved)',
       'Curricular units 2nd sem (grade)',
       'Curricular units 2nd sem (without evaluations)', 'Unemployment rate',
       'Inflation rate', 'GDP', 'Target'],
      dtype='object')
```

In [6]:
```
# Check the level of target varialbe
set(data["Target"])
```

Out[6]:
```
{'Dropout', 'Enrolled', 'Graduate'}
```

In [7]:
```
# Check the distrubution of target variable
sns.histplot(data["Target"])
```

Out[7]:
```
<AxesSubplot:xlabel='Target', ylabel='Count'>
```



In [8]:
```
# Check null values
pd.DataFrame(data.isna().sum()).rename(columns = {0:"number"})
```

Out[8]:

	number
Marital status	0
Application mode	0
Application order	0
Course	0
Daytime/evening attendance\t	0
Previous qualification	0
Previous qualification (grade)	0
Nacionality	0
Mother's qualification	0
Father's qualification	0
Mother's occupation	0
Father's occupation	0
Admission grade	0
Displaced	0
Educational special needs	0
Debtor	0
Tuition fees up to date	0
Gender	0
Scholarship holder	0
Age at enrollment	0
International	0
Curricular units 1st sem (credited)	0
Curricular units 1st sem (enrolled)	0
Curricular units 1st sem (evaluations)	0
Curricular units 1st sem (approved)	0
Curricular units 1st sem (grade)	0
Curricular units 1st sem (without evaluations)	0
Curricular units 2nd sem (credited)	0
Curricular units 2nd sem (enrolled)	0
Curricular units 2nd sem (evaluations)	0
Curricular units 2nd sem (approved)	0
Curricular units 2nd sem (grade)	0
Curricular units 2nd sem (without evaluations)	0
Unemployment rate	0
Inflation rate	0
GDP	0
Target	0

In [9]:
```
# Convert the admission grade to categorical variable
df = data.copy()
df = df[df['Target'].isin(["Dropout", "Graduate"])]
med = df['Admission grade'].median()
df['Admission grade'] = np.where(df['Admission grade'] >= med, 1, 0)
```

In [10]:
```
# Split the treatment group and control group
df_trt = df[df['Admission grade']==1]
df_ctl = df[df['Admission grade']==0]
```

In [11]:
```
# Get the shape of treatment group data
df_trt.shape
```

Out[11]:
```
(1821, 37)
```

In [12]:
```
# Get the shape of control group data
df_ctl.shape
```

Out[12]:
```
(1809, 37)
```

In [ ]:
```
# Calculate the percentage for categorical variables for treatment group
def get_categorical_percentages(df, categorical_list):
    for var in categorical_list:
        perc = df[var].value_counts() / df[var].count()
        print(var)
        print(perc)

categorical_list = list(data.columns)
continuous_list = ["Previous qualification (grade)",
                   "Unemployment rate",
                   "Inflation rate",
                   "GDP",
                   "Age at enrollment",
                   "Admission grade"]
categorical_list = [var for var in categorical_list if var not in continuous_list]

get_categorical_percentages(df_trt, categorical_list)
```

In [ ]:
```
# Calculate the percentage for categorical variables for control group
get_categorical_percentages(df_ctl, categorical_list)
```

In [15]:
```
# Calculate the IQR for numerical variables in treatment group
df_trt[continuous_list].describe().round(2).transpose()
```

Out[15]:

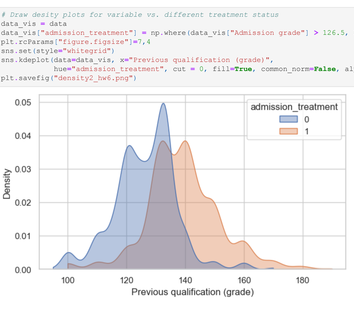
	count	mean	std	min	25%	50%	75%	max
Previous qualification (grade)	1821.0	139.37	12.65	100.00	132.0	140.00	147.00	190.00
Unemployment rate	1821.0	11.63	2.67	7.60	9.4	11.10	13.90	16.20
Inflation rate	1821.0	1.22	1.41	-0.80	0.3	1.40	2.60	3.70
GDP	1821.0	-0.08	2.22	-4.06	-1.7	0.32	1.74	3.51
Age at enrollment	1821.0	23.24	7.76	17.00	18.0	20.00	25.00	61.00
Admission grade	1821.0	1.00	0.00	1.00	1.0	1.00	1.00	1.00

In [16]:
```
# Calculate the IQR for numerical variables in control group
df_ctl[continuous_list].describe().round(2).transpose()
```

Out[16]:

	count	mean	std	min	25%	50%	75%	max
Previous qualification (grade)	1809.0	126.43	10.33	95.00	120.0	127.00	133.10	170.00
Unemployment rate	1809.0	11.63	2.67	7.60	9.4	11.10	13.90	16.20
Inflation rate	1809.0	1.25	1.35	-0.80	0.3	1.40	2.60	3.70
GDP	1809.0	0.06	2.29	-4.06	-1.7	0.32	1.79	3.51
Age at enrollment	1809.0	23.69	7.90	18.00	19.0	20.00	26.00	70.00
Admission grade	1809.0	0.00	0.00	0.00	0.0	0.00	0.00	0.00

In [18]:
```
# Draw desity plots for variable vs. different treatment status
data_vis = data
data_vis["admission_treatment"] = np.where(data_vis["Admission grade"] > 126.5, 1, 0)
plt.rcParams["figure.figsize"]=7,4
sns.set(style="whitegrid")
sns.kdeplot(data=data_vis, x="Previous qualification (grade)",
            hue="admission_treatment", cut = 0, fill=True, common_norm=False, alpha=0.4)
plt.savefig("density2_hw6.png")
```



In [ ]:
```
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5      v purrr   0.3.4
## v tibble  3.1.6      v dplyr   1.0.10
## v tidyr   1.1.4      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.1
```

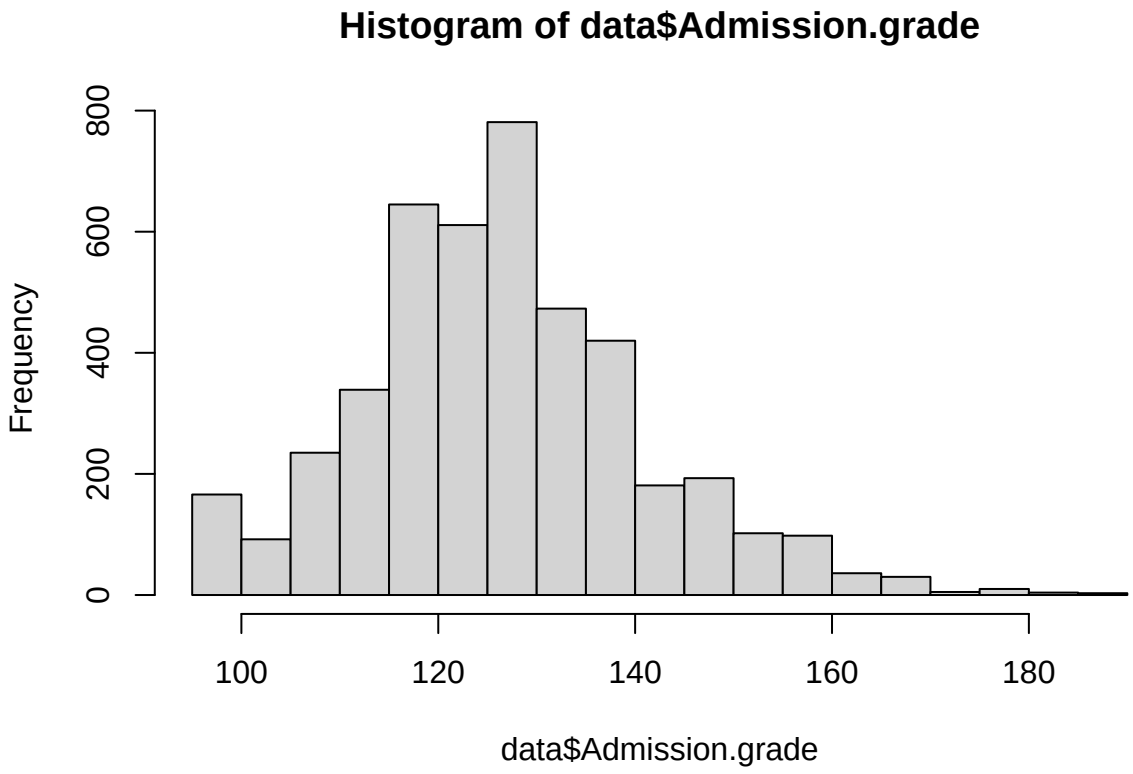
```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
data = read.csv("./graduation.csv")[-1]
sum(is.na(data))
```

```
## [1] 0
```

```
# summary(data)
```

```
hist(data$Admission.grade)
```




```
# do not use target variable before propensity score analysis
data %>% group_by(Target) %>% summarise(count=n())
```

```
## # A tibble: 3 x 2
##   Target   count
##   <chr>    <int>
## 1 Dropout   1421
## 2 Enrolled   794
## 3 Graduate  2209
```

We only need pre-treatment covariates, so drop: Tuition.fees.up.to.date, Scholarship.holder, Curricu-

lar.units.xxx (22-33 cols), course and application information.

```
# preprocessing
df = data[data$Target %in% c("Dropout", "Graduate"), ]
df$Target = ifelse(df$Target=="Graduate", 1, 0)

# select pre-treatment covariates
df = df[-seq(22, 33)] %>%
  select(-Tuition.fees.up.to.date, -Scholarship.holder,
         -Application.mode, -Application.order,
         -Course, -Daytime.evening.attendance.)
num_vars = c("Previous.qualification..grade.", "Age.at.enrollment",
             "Unemployment.rate", "GDP", "Inflation.rate")
all_vars = colnames(df %>% select(-Target, -Admission.grade))
cate_vars = all_vars[!all_vars %in% num_vars]
for(var in cate_vars){
  df[, var] = as.factor(df[, var])
}
```

```
# cutoff for admission grade (continuous treatment)
# 1. use median: below median - low, above median - high
# mean(df$Admission.grade) # 127.2939
med = median(df$Admission.grade) # 126.5
df$treatment = ifelse(df$Admission.grade>=med, 1, 0)
df = df %>% select(-Admission.grade)
```

Baseline table: split data according to the treatment (how the treatment/controlled population are different) after choosing a cutoff for the treatment

```
# use t-test/chi-square to see if confounding possible (influence both trt and tgt)
for(var in cate_vars){
  print(paste0("Pre-treatment covariate is:", var))
  print(chisq.test(df[, var], df[, "treatment"]))
}
```

```
## Warning in chisq.test(df[, var], df[, "treatment"]): Chi-squared approximation
## may be incorrect

## Warning in chisq.test(df[, var], df[, "treatment"]): Chi-squared approximation
## may be incorrect

## Warning in chisq.test(df[, var], df[, "treatment"]): Chi-squared approximation
## may be incorrect

## Warning in chisq.test(df[, var], df[, "treatment"]): Chi-squared approximation
## may be incorrect

## Warning in chisq.test(df[, var], df[, "treatment"]): Chi-squared approximation
## may be incorrect

## Warning in chisq.test(df[, var], df[, "treatment"]): Chi-squared approximation
## may be incorrect

## Warning in chisq.test(df[, var], df[, "treatment"]): Chi-squared approximation
## may be incorrect
```

```
for(var in num_vars){
 print(paste0("Pre-treatment covariate is:", var))
  print(t.test(df[, var] ~ df[, "treatment"]))
}
```

ATE estimates the change in the outcome if the treatment were applied to the entire population versus if the control were applied to the entire population.

```
library(twang)
```

```
## To reproduce results from prior versions of the twang package, please see the version="legacy" optio
```

```
df_sub = df %>% select(-Target)
# calculates propensity scores using gradient boosted logistic regression
boosted.mod = ps(treatment ~ .,
                 data=df_sub,
                 estimand="ATE", # the causal effect of interest
                 n.trees=10000, # number of gbm iterations
                 interaction.depth=3, # default of tree depth used in gb
                 perm.test.iters=0, # p-value estimation
                 verbose=FALSE,
                 stop.method=c("es.mean"))
summary(boosted.mod)
```

```
##             n.treat n.ctrl ess.treat  ess.ctrl    max.es    mean.es    max.ks
## unw            1821   1809  1821.000 1809.0000 0.9771914 0.04511029 0.4855987
## es.mean.ATE    1821   1809  1128.874  960.2814 0.3298046 0.02468700 0.1409446
##             max.ks.p     mean.ks iter
## unw               NA 0.008645762   NA
## es.mean.ATE       NA 0.003754655 3017
```

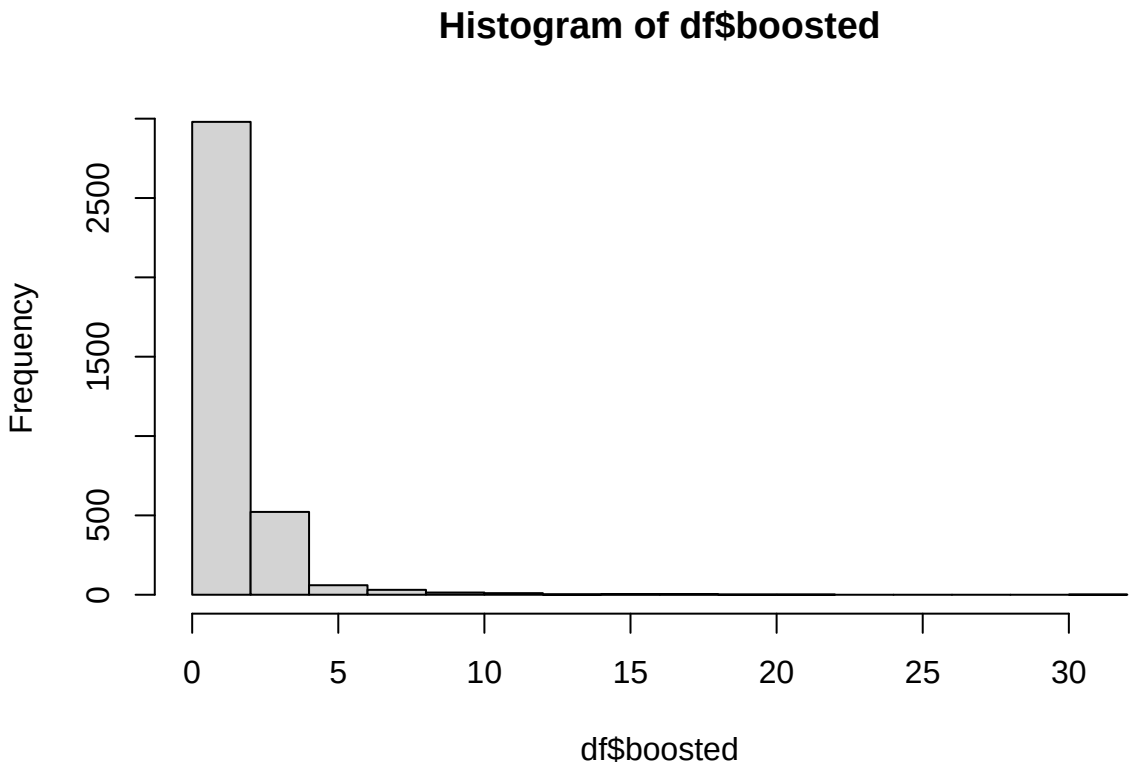
```
summary(boosted.mod$gbm.obj,
        n.trees=boosted.mod$desc$es.mean.ATE$n.trees,
        plot=FALSE)
```

```
##                                                        var    rel.inf
## Previous.qualification..grade. Previous.qualification..grade. 66.6952392
## Previous.qualification                 Previous.qualification  8.4219879
## Age.at.enrollment                           Age.at.enrollment  7.9611064
## Father.s.occupation                       Father.s.occupation  4.5502649
## Father.s.qualification                 Father.s.qualification  2.6225436
## Mother.s.occupation                       Mother.s.occupation  2.3114703
## Mother.s.qualification                 Mother.s.qualification  2.1341326
## Marital.status                                 Marital.status  2.0395956
## Gender                                                 Gender  1.2102053
## Nacionality                                       Nacionality  0.6286985
## Inflation.rate                                 Inflation.rate  0.5324174
## Unemployment.rate                           Unemployment.rate  0.4022262
## Educational.special.needs           Educational.special.needs  0.2484688
## GDP                                                       GDP  0.1259934
## Displaced                                           Displaced  0.1156500
## Debtor                                                 Debtor  0.0000000
## International                                   International  0.0000000
```

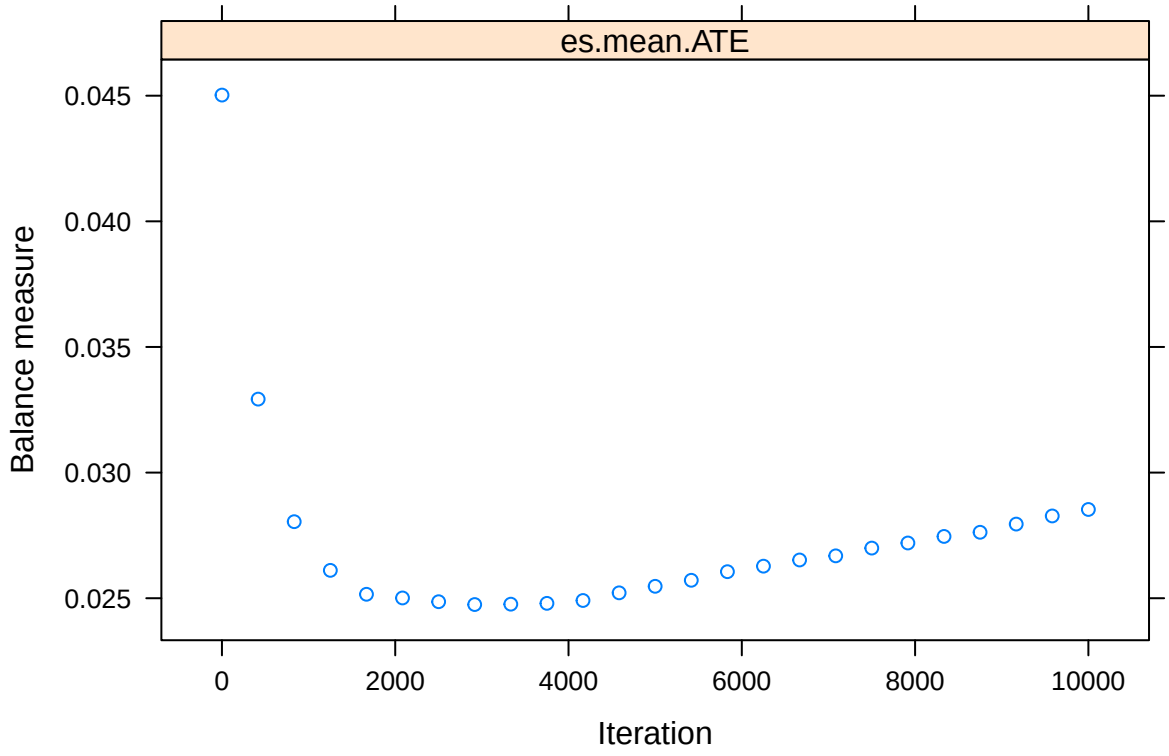
```
df$boosted = get.weights(boosted.mod)
```

```
## Warning in get.weights(boosted.mod): No stop.method specified.  Using es.mean.ATE
```

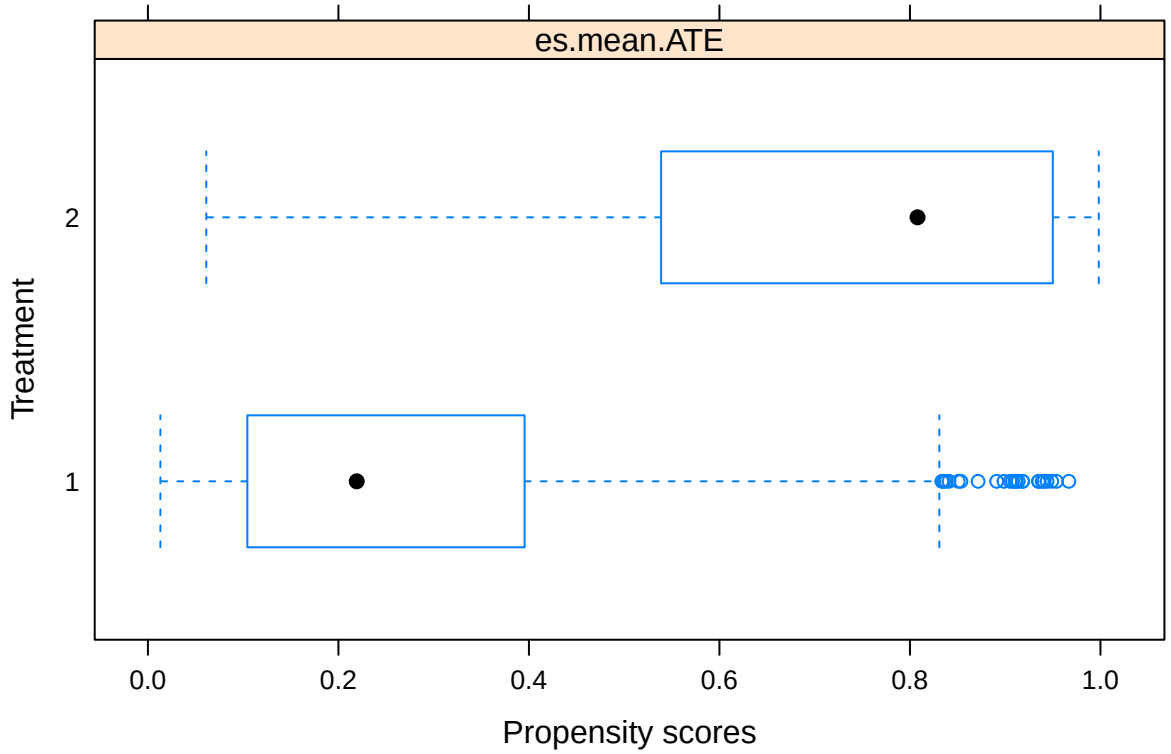
```
hist(df$boosted)
```



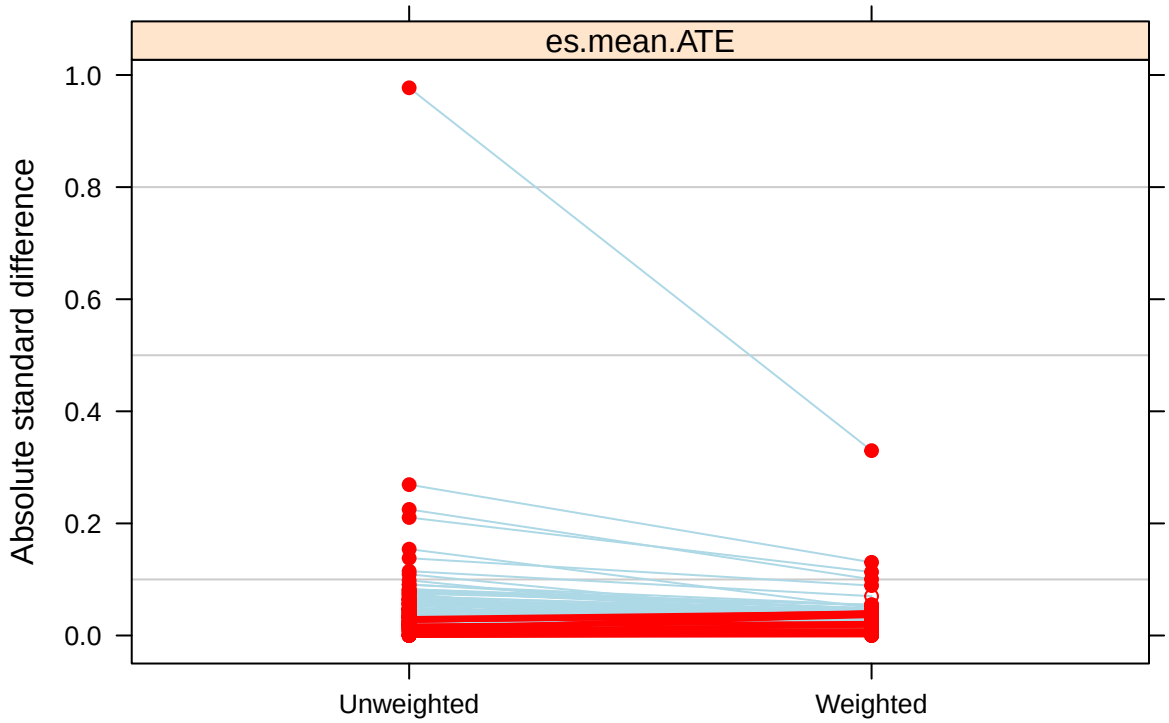

```
# a plot of the balance criteria as a function of the GBM iteration
plot(boosted.mod)
```



```
# boxplots of the propensity scores for the treatment and control cases
plot(boosted.mod, plots=2)
```




```
plot(boosted.mod, plots=3)
```




```
# baseline table
bal.table(boosted.mod)
```

```r
# evaluate ATE
library(survey)
```

```
## Loading required package: grid
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loading required package: survival
```

```
##
## Attaching package: 'survey'
```

```
## The following object is masked from 'package:graphics':
##
##     dotchart
```

```r
design = svydesign(ids=~1, weights=~boosted, data=df)
glm = svyglm(Target ~ treatment, design=design, family=quasibinomial())
summary(glm)
```

```
##
## Call:
## svyglm(formula = Target ~ treatment, design = design, family = quasibinomial())
##
## Survey design:
## svydesign(ids = ~1, weights = ~boosted, data = df)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.28983    0.06468   4.481 7.66e-06 ***
## treatment    0.24073    0.08962   2.686  0.00726 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasibinomial family taken to be 1.000276)
##
## Number of Fisher Scoring iterations: 4
```

For propensity score analysis, shouldn't look at outcome until calculating ATE (avoid p-hacking). Plot the admission grade wrt target after.

```r
df$treatment = as.factor(df$treatment)
levels(df$treatment) = c("below median", " above median")
df$Target = as.factor(df$Target)
levels(df$Target) = c("dropout", "graduate")
ggplot(df, aes(x=Target, fill=treatment)) +
  labs(x="Student status", y="Count") +
  scale_fill_discrete(name = "Treatment") +
  geom_bar(position = "dodge") +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
```

```
        panel.background = element_blank(),
        axis.line = element_line(color = "black"))
```