Assessing Sensitivity of Machine Learning Predictions. A Novel Toolbox with an Application to Financial Literacy.

Data application

Falco J. Bargagli-Stoffi, Kenneth de Beckker, Kristof de Witte, Joana E. Maldonado

February 2021

Introduction

In the following document we depict the code used to build, describe and clean the data set, and to perform the machine learning analysis. The functions used for the machine learning analysis can be found here https://github.com/fbargaglistoffi/sensitivity-analysis-machine-learning. Feel free to reach out for any question regarding the following code at: fbargaglistoffi@hsph.harvard.edu.

Upload and Merge Data

One can run the following chunk of code in `Stata` to reconstruct the data set. Let us point out that, through the usage of `R Markdown` one can run `Stata` code directly in `R`.

First, we upload the *Pisa* data (Ω_{pisa}), the financial literacy data (Ω_{finlit}) and we merge the two data sets ($\Omega = \Omega_{pisa} \cup \Omega_{finlit}$). For all the observations in Ω_{finlit} we observe the outcome variable $y_{\Omega_{finlit}}$, while for the observations in Ω_{pisa} we do not.

The aim of the following code is to use the data in Ω_{finlit} to predict $\hat{y}_{\Omega_{pisa}}$. In particular, we will focus on predicting the financial literacy scores of students in Wallonia (for which we observe all the predictors but we do not observe the outcome) by training the machine learning model on the data on Flemish students (for which we observe both the predictors and the outcome).

```
********************************************************************************
***PISA DATA
clear all
cd "G:\Il mio Drive\Research\Financial Literacy\wetransfer-75b189"
use PISA_DATA2015, clear

********************************************************************************
***MERGE & data set SELECTION
//merge general PISA to financial literacy PISA on student ID
merge 1:1 CNT CNTSCHID CNTSTUID using PISA2015FL
    gen generalPISA=1
        replace generalPISA=0 if _merge==2
        label var generalPISA "Student Has Completed General PISA"
    gen finlitPISA=1
        replace finlitPISA=0 if _merge==1
        label var finlitPISA "Student Has a PV for Finanancial Literacy"
    drop _merge

merge m:1 CNT CNTSCHID using PISA_schools_2015
    tab CNTRYID generalPISA
    tab CNTRYID finlitPISA

drop if finlitPISA==1 &  generalPISA==0 //drop if FL PISA but no general PISA
    tab CNT finlitPISA

//available countries: BEL, CAN, ESP, ITA, NLD, POL, SVK, USA
keep if CNT=="BEL" | CNT=="CAN" | CNT=="ESP" | CNT=="ITA" | CNT=="NLD" ///
    | CNT=="POL" | CNT=="SVK" | CNT=="USA"
    tab CNTRYID finlitPISA
    tab CNTRYID generalPISA

********************************************************************************
***CLEAN VARIABLES
//set missings on financial literacy questions equal to missing scores
gen FLindicator=1 //indicator if financial literacy questions are available
    foreach var of varlist CF001Q01S CF006Q02S CF009Q02S CF010Q01S CF010Q02S ///
        CF012Q01S CF012Q02S CF028Q03S CF031Q01S CF031Q02S CF033Q01S CF033Q02S ///
        CF035Q01S CF062Q01S CF069Q01S CF075Q02S CF082Q02S CF095Q01S CF095Q02S ///
        CF097Q01S CF102Q01S CF105Q01S CF105Q02S CF106Q02S CF110Q01S CF202Q01S {
        replace FLindicator=0 if ‘var’==6 | ‘var’==9 | ‘var’==.
    }
    label var FLindicator "Student Has Completed Financial Literacy PISA"

keep PV1FLIT CNTRYID CNT CNTSCHID CNTSTUID Region generalPISA finlitPISA ///
    FLindicator ST001D01T ST004D01T AGE ISCEDD ISCEDO LANGN HEDRES ///
    WEALTH ST013Q01TA IMMIG MISCED FISCED BMMJ1 BFMJ2 EMOSUPS REPEAT OUTHOURS ///
    MMINS LMINS HADDINST PV1MATH PV1READ ANXTEST MOTIVAT W_* ///
    SC001Q01TA SC016Q01TA SC042Q01TA SC042Q02TA SC048Q01NA SC048Q02NA ///
    SC048Q03NA SCHSIZE CLSIZE RATCMP1 RATCMP2 LEADPD SCHAUT EDUSHORT STAFFSHORT ///
    PROAT5AB PROAT5AM PROAT6 PROATCE STUBEHA TEACHBEHA STRATIO
drop W_SCHGRNRABWT

//set non-responders missing
foreach var of varlist BMMJ1 BFMJ2 {
    replace ‘var’=99999 if ‘var’==9999
}
foreach var of varlist OUTHOURS SC016Q01TA SC048Q01NA SC048Q02NA ///
    SC048Q03NA RATCMP1 RATCMP2 SCHAUT STRATIO {
    replace ‘var’=99999 if ‘var’==999
}
foreach var of varlist SC016Q01TA {
    replace ‘var’=99999 if ‘var’==998
}
foreach var of varlist WEALTH MISCED FISCED MOTIVAT ANXTEST EMOSUPS  HEDRES ///
    ST013Q01TA SC001Q01TA CLSIZE LEADPD EDUSHORT STAFFSHORT PROAT5AB PROAT5AM ///
    PROAT6 PROATCE STUBEHA TEACHBEHA {
    replace ‘var’=99999 if ‘var’==99
}
foreach var of varlist ST001D01T {
    replace ‘var’=99999 if ‘var’==96
}
foreach var of varlist IMMIG REPEAT ISCEDD ISCEDO SC042Q01TA SC042Q02TA {
    replace ‘var’=99999 if ‘var’==9
}
foreach var of varlist MMINS LMINS {
    replace ‘var’=99999 if ‘var’>1600
}
foreach var of varlist SCHSIZE {
    replace ‘var’=99999 if ‘var’>4230
}

//group Belgian languages
gen BELANGN=LANGN
    replace BELANGN=2 if BELANGN==230 | BELANGN==493
    replace BELANGN=4 if BELANGN==344 | BELANGN==500 | BELANGN==606 ///
        | BELANGN==615 | BELANGN==804
    replace BELANGN=1 if BELANGN==621 | BELANGN==322
    replace BELANGN=3 if BELANGN==640 | BELANGN==148
label define BELANGN 1 "Dutch" 2 "French" 3 "German" 4 "Other"
label values BELANGN BELANGN
label var BELANGN "Language grouped for BELGIUM"
tab BELANGN if CNT=="BEL"
    drop LANGN

gen BELANGNdum=.
    replace BELANGNdum=0 if BELANGN==4
    replace BELANGNdum=1 if BELANGN==1 | BELANGN==2 | BELANGN==3
label define BELANGNdum 1 "Speaks Belgian Language at Home" 0 "Speaks Other Language at Home"
label values BELANGNdum BELANGNdum
label var BELANGNdum "Speaks Belgian Language at Home"

//FL levels
gen FLITlevel=.
    replace FLITlevel=0 if PV1FLIT<326 & PV1FLIT!=.
    replace FLITlevel=1 if PV1FLIT>=326 & PV1FLIT<400 & PV1FLIT!=.
    replace FLITlevel=2 if PV1FLIT>=400 & PV1FLIT<475 & PV1FLIT!=.
    replace FLITlevel=3 if PV1FLIT>=475 & PV1FLIT<550 & PV1FLIT!=.
    replace FLITlevel=4 if PV1FLIT>=550 & PV1FLIT<625 & PV1FLIT!=.
    replace FLITlevel=5 if PV1FLIT>=625 & PV1FLIT!=.
    label var FLITlevel "Financial Literacy Proficiency Level"

gen FLITbasis=.
    replace FLITbasis=0 if PV1FLIT<400 & PV1FLIT!=.
    replace FLITbasis=1 if PV1FLIT>=400 & PV1FLIT!=.
    label var FLITlevel "Financial Literacy Baseline Level"

//Regional indicator
gen Flanders=.
    replace Flanders=1 if Region==5601
    replace Flanders=0 if Region==5602 | Region==5603
label define Flanders 0 "Wallonia" 1 "Flanders"
label values Flanders Flanders
```

```
//label variables
label var ST001D01T "International Grade"
label var ST004D01T "Gender"
label var ISCEDD "Study Track: ISCED Designation "
label var ISCEDO "Study Track: ISCED Orientation "
label var HEDRES  "Educational Resources at Home"
label var WEALTH "Family Wealth Index (Economic Possessions) "
label var ST013Q01TA "Number of Books at Home"
label var IMMIG "Immigration Status"
label var MISCED "Mother's Education (ISCED)"
label var FISCED "Father's Education (ISCED)"
label var BMMJ1 "Mother's Job (ISEI)"
label var BFMJ2 "Father's Job (ISEI)"
label var EMOSUPS "Parents Emotional Support"
label var OUTHOURS "Out-of-School Study Time per Week"
label var MMINS "Mathematics Learning Time at School"
label var LMINS "Language Learning Time at School"
label var ANXTEST "Personality: Test Anxiety"
label var MOTIVAT "Achievement Motivation"
label var SC001Q01TA "School Community (Location)"
label var SC048Q01NA "Share of Students With a Different Heritage Language"
label var SC048Q02NA "Share of Students With Special Needs"
label var SC048Q03NA "Share of Socioeconomically Disadvantaged Students"
label var SCHSIZE "School Size"
label var RATCMP1 "Number of Available Computers per Student"
label var LEADPD "Teacher Professional Development"
label var SCHAUT "School Autonomy"
label var EDUSHORT "Shortage of Educational Material"
label var STRATIO "Student-Teacher Ratio"
```

```
merged_data <- read.csv("PISA_merged_2015_new.csv", header = TRUE)
```

Data Summary

Below we depict the number of student level observations by country and the number of missing observations for the outcome ($PV1FLIT$). The outcome is missing just for Belgium and Canada: this is due to the fact that in these two nations the financial literacy assessment was made in regions instead of country-wide.

```
pisa_data <- merged_data %>% dplyr::na_if(99999)
pisa_data$AGE <- round(pisa_data$AGE)
table(pisa_data$CNTRYID)
```

```
        Belgium          Canada           Italy     Netherlands          Poland 
           9651           20058           11583            5385            4478 
Slovak Republic           Spain   United States 
           6350            6736            5712 
```

```
table(pisa_data$CNTRYID[which(is.na(pisa_data$PV1FLIT))])
```

```
Belgium  Canada
   3976    6976
```

```
pisa_data <- pisa_data[which(pisa_data$CNTRYID=="Belgium"),]
```

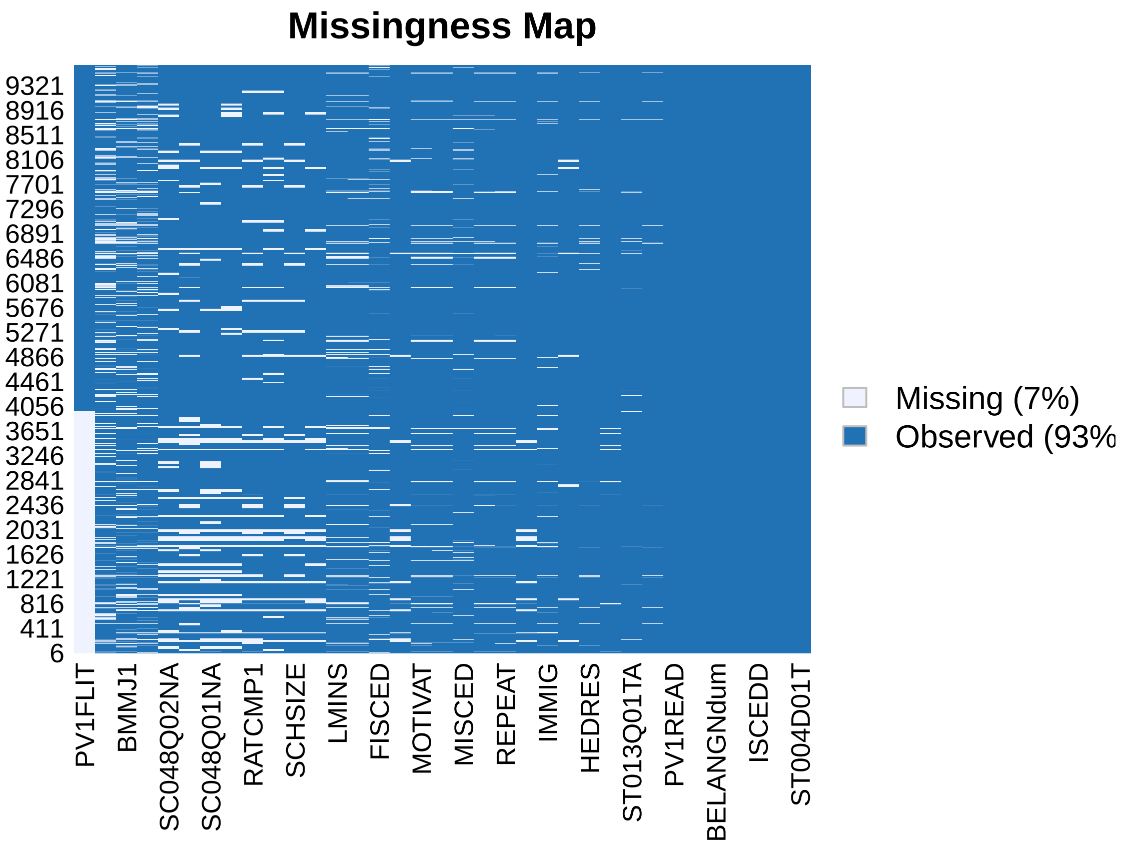
In the following, we define the variables used for the prediction. You can find more details on the variables and their choice in the paper.

```
whichvars = c("CNTRYID", "CNTSCHID","Region","generalPISA","finlitPISA",
"FLindicator","ST001D01T","ST004D01T","AGE","ISCEDD","ISCEDO",
"BELANGNdum","HEDRES","WEALTH","ST013Q01TA","IMMIG",
"MISCED","FISCED","BMMJ1","BFMJ2","EMOSUPS","REPEAT","OUTHOURS",
"MMINS","LMINS","PV1MATH","PV1READ","ANXTEST","MOTIVAT","SC001Q01TA",
"SC048Q01NA","SC048Q02NA","SC048Q03NA","SCHSIZE","CLSIZE","RATCMP1",
"LEADPD","SCHAUT","EDUSHORT","STRATIO", "PV1FLIT", "CNTSTUID")
pisa_data <- pisa_data[whichvars]
sapply(pisa_data, class)
```

```
    CNTRYID    CNTSCHID      Region generalPISA  finlitPISA FLindicator
"character"   "integer" "character"   "integer"   "integer"   "integer"
  ST001D01T   ST004D01T         AGE      ISCEDD      ISCEDO  BELANGNdum
"character" "character"   "numeric" "character" "character" "character"
     HEDRES      WEALTH  ST013Q01TA       IMMIG      MISCED      FISCED
  "numeric"   "numeric" "character" "character" "character" "character"
      BMMJ1       BFMJ2     EMOSUPS      REPEAT    OUTHOURS       MMINS
  "integer"   "integer"   "numeric" "character"   "integer"   "integer"
      LMINS     PV1MATH     PV1READ     ANXTEST     MOTIVAT  SC001Q01TA
  "integer"   "numeric"   "numeric"   "numeric"   "numeric" "character"
 SC048Q01NA  SC048Q02NA  SC048Q03NA     SCHSIZE      CLSIZE     RATCMP1
  "integer"   "integer"   "integer"   "integer"   "integer"   "numeric"
     LEADPD      SCHAUT    EDUSHORT     STRATIO     PV1FLIT    CNTSTUID
  "numeric"   "numeric"   "numeric"   "numeric"   "numeric"   "integer"
```

The map below indicates which are the observations that are missing (roughly 6% of the total). As we do not see clear patterns of missingness we can assume our observation to be either Missing-at-Random or Missining-Completely-at-Random. N.B. We exclude the variables `FLindicator`, `generalPISA` and `CNTSCHID` from the "missingess map" as they are not missing by design.

```
Amelia::missmap(pisa_data[,-which(names(pisa_data) %in% c("FLindicator",
                         "generalPISA", "CNTSCHID", "CNTRYID", "finlitPISA",
                         "CNTSTUID", "Region"))])
```



We use the `mice` package to get an imputation for the missing values. This is done through a CART imputation methodology. For more details, check out the following vignette: https://cran.r-project.org/web/packages/mice/mice.pdf.

```
whichvars = c("ST001D01T","ST004D01T","AGE","ISCEDD","ISCEDO",
"BELANGNdum","HEDRES","WEALTH","ST013Q01TA","IMMIG",
"MISCED","FISCED","BMMJ1","BFMJ2","EMOSUPS","REPEAT","OUTHOURS",
"MMINS","LMINS","PV1MATH","PV1READ","ANXTEST","MOTIVAT","SC001Q01TA",
"SC048Q01NA","SC048Q02NA","SC048Q03NA","SCHSIZE","CLSIZE","RATCMP1",
"LEADPD","SCHAUT","EDUSHORT","STRATIO","CNTSTUID")
imputed <- pisa_data[whichvars]
imputed_data <- mice(imputed, method = ’cart’, maxit = 1,  m = 1,
                     remove.collinear = FALSE)
demdata <- complete(imputed_data)
pisa_data <- cbind(demdata, pisa_data$PV1FLIT)
names(pisa_data)[ncol(pisa_data)] <- "PV1FLIT"
```

Explanatory Data Analysis

Outcome

In the following code I focus on one financial literacy outcome (namely, `PV1FLIT`). You can extend the analysis to the other outcomes by simply changing the name of the outcome in the following chunks of code. Where the outcome name is preceded by \texttt{pisa_data$} you must change just the name of the outcome while keeping \texttt{pisa_data$} fixed: e.g. \texttt{pisa_data$PV1FLIT} becomes \texttt{pisa_data$PV2FLIT}.

In the following the summary of the outcome variable.

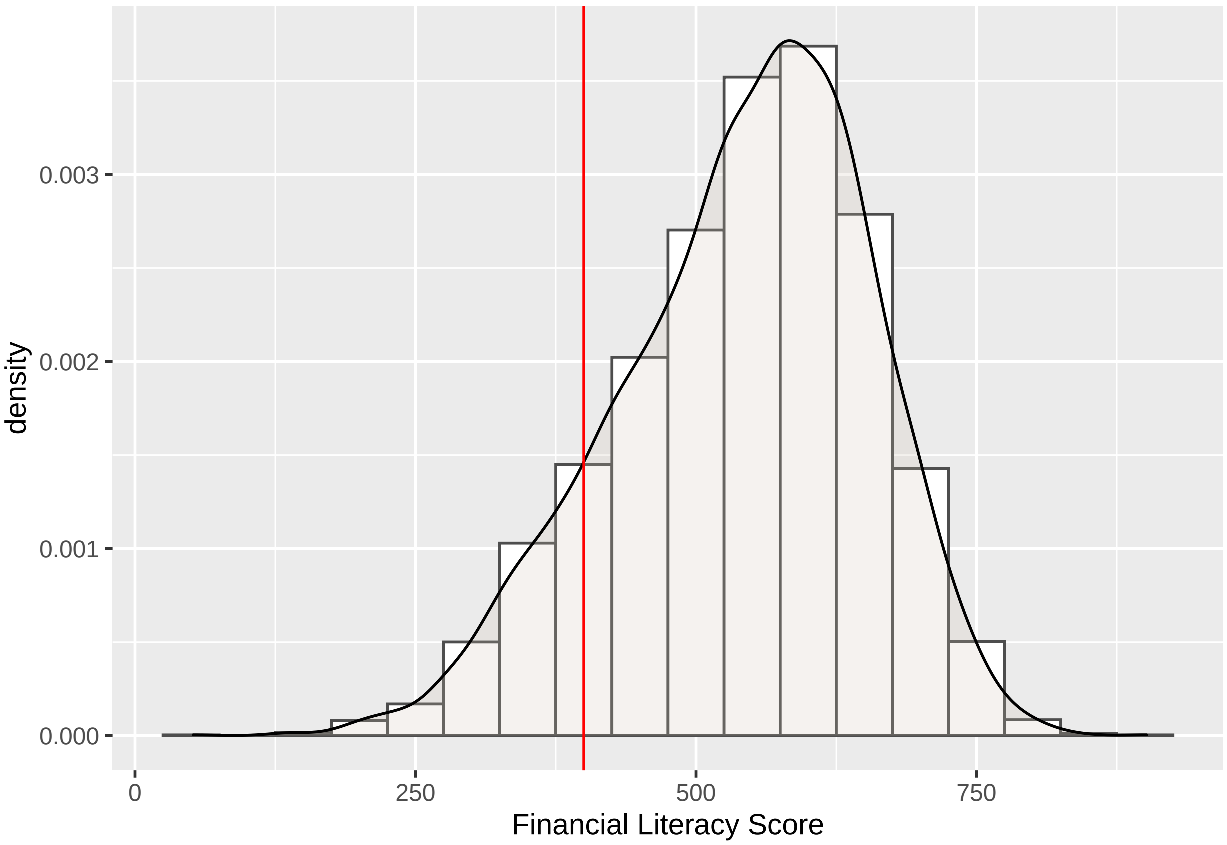
```
summary(pisa_data$PV1FLIT) # change here the outcome
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  51.81  469.04  555.14  541.43  622.56  901.64    3976 
```

```
//summary statistics table of the outcome variable
eststo FLIT: quietly estpost summarize PV1FLIT if Flanders == 1, detail
esttab FLIT using "table_outcome.tex", replace ///
    cells("mean(fmt(2) label(Mean)) sd(fmt(2) label(SD))
          min(fmt(2) label(Minimum)) p50(fmt(2) label(Median))
          max(fmt(2)            label(Maximum)) count(fmt(%9.0g) label(N))")
label nonum gaps mlabels("Flanders", nonum) noobs booktabs
```

In the following an histogram of the outcome variable. The red line indicates the threshold of the baseline level of proficiency in financial literacy. The OECD suggests that students above this threshold of 400 points have financial literacy levels that are sufficient to participate in society.

```
ggplot(pisa_data[which(!is.na(pisa_data$PV1FLIT)),],
       aes(PV1FLIT)) + # change here the outcome
  geom_histogram(aes(y = ..density..),
                 binwidth = 50, color = "grey30", fill = "white") +
  geom_density(alpha = .2, fill = "antiquewhite3") +
  xlab("Financial Literacy Score")  + geom_vline(xintercept = 400, color = "red")
```



Predictors

In the following summary statistics of the predictors by region.

```
preserve
keep if CNT=="BEL"

//correlation between the PISA mathematics score and FLS in Flanders
pwcorr PV1MATH PV1FLIT, sig obs

//summary statistics table of the predictors
local vars PV1FLIT ST001D01T ST004D01T AGE ISCEDD ISCEDO BELANGNdum HEDRES  ///
        WEALTH ST013Q01TA IMMIG MISCED FISCED BMMJ1 BFMJ2 EMOSUPS PV1MATH ///
        PV1READ REPEAT OUTHOURS MMINS LMINS ANXTEST MOTIVAT SC001Q01TA SC048Q01NA ///
        SC048Q02NA SC048Q03NA SCHSIZE CLSIZE RATCMP1 LEADPD SCHAUT EDUSHORT ///
        STRATIO
foreach var of varlist ‘vars’ {
    replace ‘var’=. if ‘var’==99999
}
local predictors ST001D01T ST004D01T AGE ISCEDD ISCEDO BELANGNdum HEDRES  ///
        WEALTH ST013Q01TA IMMIG MISCED FISCED BMMJ1 BFMJ2 EMOSUPS PV1MATH ///
        PV1READ REPEAT OUTHOURS MMINS LMINS ANXTEST MOTIVAT SC001Q01TA SC048Q01NA ///
        SC048Q02NA SC048Q03NA SCHSIZE CLSIZE RATCMP1 LEADPD SCHAUT EDUSHORT ///
        STRATIO
eststo Flanders: quietly estpost summarize ‘predictors’ if Flanders == 1
eststo Wallonia: quietly estpost summarize ‘predictors’ if Flanders == 0
eststo diff: quietly estpost ttest ‘predictors’, by(Flanders)

esttab Flanders Wallonia diff using "table_sum.tex", replace ///
    cells("mean(pattern(1 1 0) fmt(2) label(Mean)) sd(pattern(1 1 0) fmt(2) label(SD))
          count(pattern(1 1 0) fmt(%9.0g) label(N))         p(pattern(0 0 1) fmt(3)
          label(p-value))") ///
    label nonote wide compress alignment(D{.}{.}{-1}) nonum gaps ///
    mlabels("Flanders" "Wallonia" "Difference in Means", nonum) noobs booktabs ///
    order("\" "\textit{Student Characteristics}" ST001D01T ST004D01T AGE ISCEDD
    ISCEDO BELANGNdum "\textit{Socioeconomic Status}"       ///
    HEDRES  WEALTH ST013Q01TA IMMIG MISCED FISCED BMMJ1 BFMJ2 EMOSUPS "
    \textit{Achievement}" PV1MATH ///
    PV1READ REPEAT OUTHOURS MMINS LMINS ANXTEST MOTIVAT "\textit{School Characteristics}"
    SC001Q01TA SC048Q01NA ///
    SC048Q02NA SC048Q03NA SCHSIZE CLSIZE RATCMP1 LEADPD SCHAUT EDUSHORT STRATIO)
restore
```

Analysis Overlap of the Distributions

Single Variable (Wealth and Mathematics Score)

In the following chunk, we check how much the wealth distribution of students' families in Wallonia overlaps with the distribution of students' families in Flanders. As one can see from the summary below, the distribution of wealth is shifted on the right for Flanders with respect to Wallonia. This is a well-known difference between these two regions of Belgium as people in Wallonia have had persistently lower incomes than people in Flanders.

```r
# Wallonia
summary(as.numeric(pisa_data$WEALTH[which(!is.na(pisa_data$WEALTH) &
                                            is.na(pisa_data$PV1FLIT))]))
```

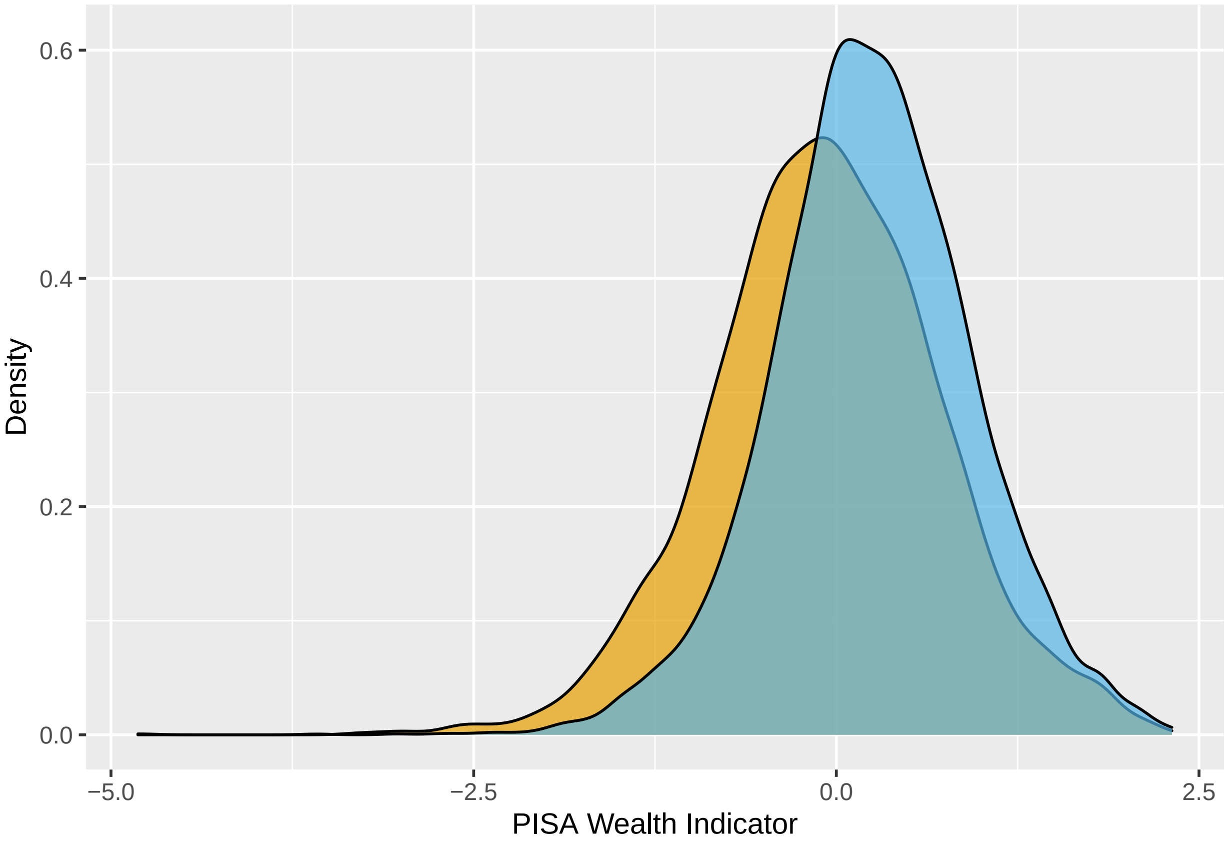
```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
-4.81520 -0.56345 -0.06570 -0.04058  0.46585  4.11970 
```

```r
# Flanders
summary(as.numeric(pisa_data$WEALTH[which(!is.na(pisa_data$WEALTH) &
                                            !is.na(pisa_data$PV1FLIT))]))
```

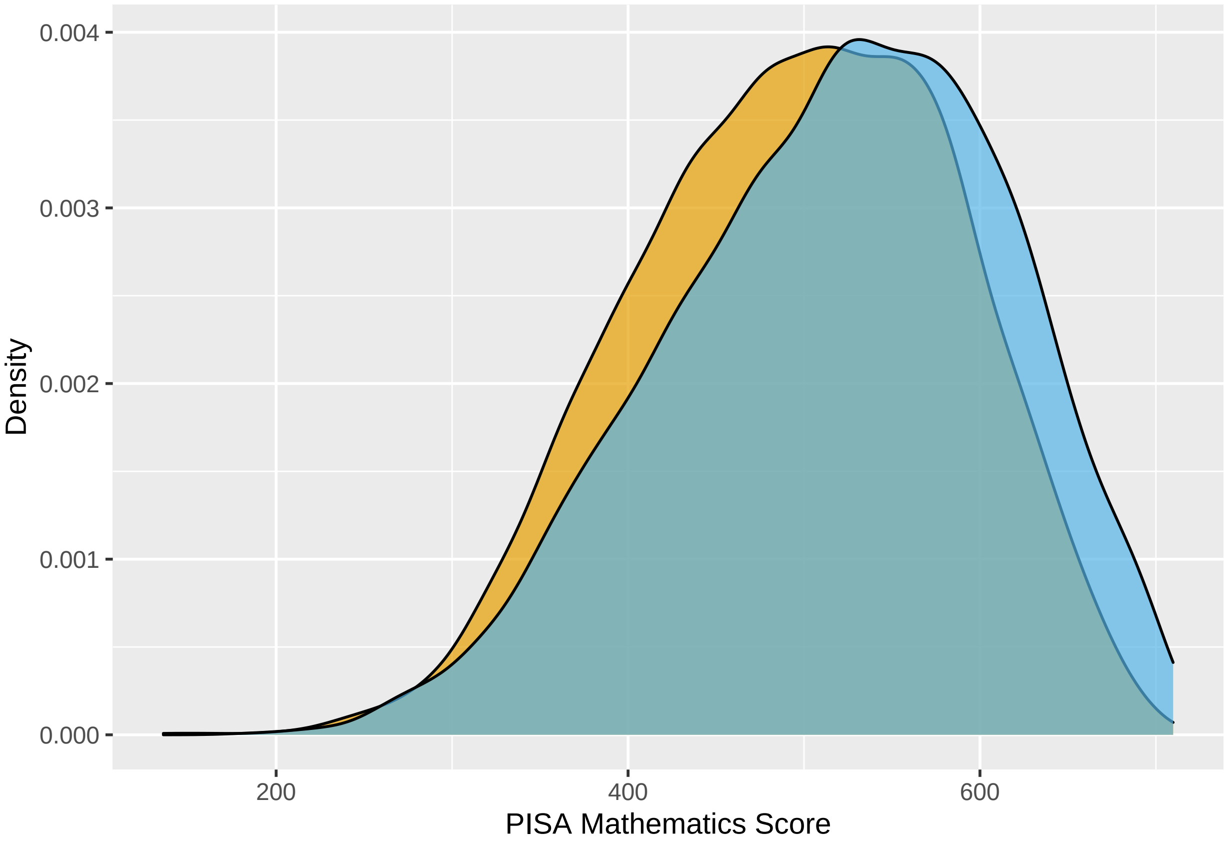
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
-3.5822 -0.1773  0.2461  0.2742  0.7036  4.0887 
```

The following two figures show that of the distribution of family wealth and PISA mathematics scores overlap in the samples of Wallonia and Flanders, but for both variables, the distribution is slightly shifted to the right in Flanders (blue) compared to Wallonia (orange).

```r
ggplot() +
  geom_density(data=pisa_data[which(!is.na(pisa_data$WEALTH) & is.na(pisa_data$PV1FLIT) &
  pisa_data$WEALTH < as.numeric(quantile(pisa_data$WEALTH, 0.99, na.rm = TRUE))),],
  aes(WEALTH),
  fill = "#E69F00", color = "black", alpha = 0.7) +
  geom_density(data=pisa_data[which(!is.na(pisa_data$WEALTH) & !is.na(pisa_data$PV1FLIT) &
  pisa_data$WEALTH < as.numeric(quantile(pisa_data$WEALTH, 0.99, na.rm = TRUE))),],
  aes(WEALTH),
  fill = "#56B4E9", color = "black", alpha = 0.7) +
  xlab("PISA Wealth Indicator") + ylab("Density")
```



```r
ggplot() +
  geom_density(data=pisa_data[which(!is.na(pisa_data$PV1MATH) & is.na(pisa_data$PV1FLIT) &
  pisa_data$PV1MATH < as.numeric(quantile(pisa_data$PV1MATH, 0.99, na.rm = TRUE))),],
  aes(PV1MATH),
  fill = "#E69F00", color = "black", alpha = 0.7) +
  geom_density(data=pisa_data[which(!is.na(pisa_data$PV1MATH) & !is.na(pisa_data$PV1FLIT) &
  pisa_data$PV1MATH < as.numeric(quantile(pisa_data$PV1MATH, 0.99, na.rm = TRUE))),],
  aes(PV1MATH),
  fill = "#56B4E9", color = "black", alpha = 0.7) +
  xlab("PISA Mathematics Score") + ylab("Density")
```



# Multivariate Overlap

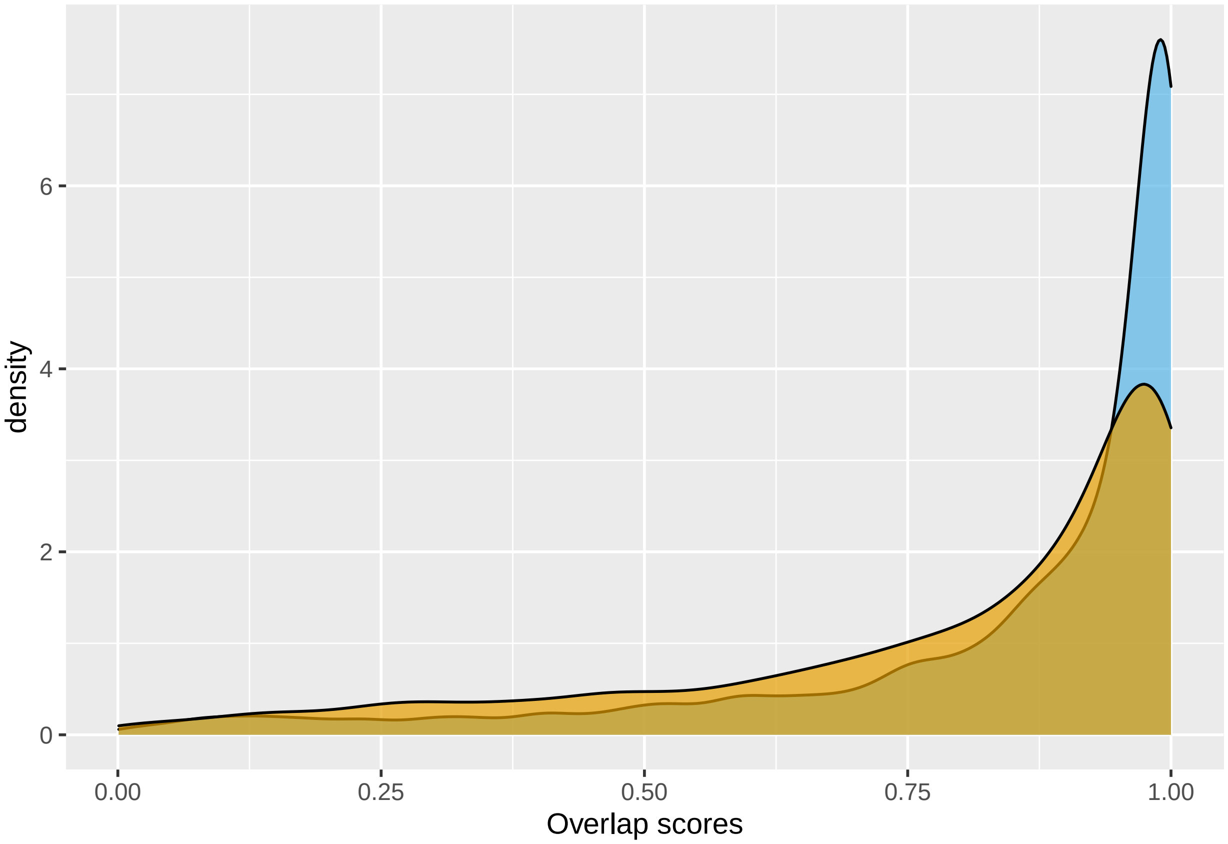
Below, we run a Logit model for the estimation of the overlap score.

```
fullmod = glm(study_pop ~., data = pisa_logit[,c(vars_prediction, "study_pop")],
              family = "binomial")
```

```
datmodel = pisa_logit[,c(vars_prediction, "study_pop", "CNTSTUID")]
datoutcome = pisa_logit[,c("PV1FLIT", "CNTSTUID")]
datmodel = na.omit(datmodel)
datmodel = merge(datmodel, datoutcome , by="CNTSTUID", all.x=TRUE)
datmodel$p.scores <- predict(fullmod,
                    data = datmodel,
                    type="response")  # predicted scores
```

In the chunk of code below, I show the distribution of the propensity score in the group of people in the financial literacy study and in the group of people in the PISA data, respectively.

```
ggplot() +
  geom_density(data = datmodel[which(datmodel$study_pop==1),],
  aes(p.scores),
  fill = "#56B4E9", color = "black", alpha = 0.7) +
  geom_density(data = datmodel[which(datmodel$study_pop==0),],
  aes(1 - p.scores),
  fill = "#E69F00", color = "black", alpha = 0.7) +
  xlab("Overlap scores")
```



Main Analysis

The machine learning model used for the prediction are a Bayesian Additive Regression Trees (BART) model (Chipman et al., 2010) and a Random Forest(RF) model (Breiman, 2001).

Goodness of Fit of the Prediction

Below we train the model on the data for which the outcome is observed ($y_{\Omega_{pisa}}$) and we evaluate the fit with a 10-folds cross-validation procedure.

BART

```
vars_prediction = c("ST001D01T","ST004D01T","AGE","BELANGNdum","ISCEDD","ISCEDO",
"HEDRES","WEALTH","ST013Q01TA","IMMIG", "SC001Q01TA","SC048Q01NA",
"SC048Q02NA","SC048Q03NA","MISCED","FISCED","BMMJ1","BFMJ2","EMOSUPS",
"REPEAT","OUTHOURS","MMINS","LMINS","PV1MATH","PV1READ","ANXTEST",
"MOTIVAT","SCHSIZE","CLSIZE","RATCMP1","LEADPD","SCHAUT","EDUSHORT","STRATIO")

datmodel = pisa_data[,c(vars_prediction, "PV1FLIT", "CNTSTUID")]
```

```r
# Convert character to factor
datmodel[sapply(datmodel, is.character)] <- lapply(datmodel[sapply(datmodel,
                                                   is.character)],
                                                   as.factor)

fin_lit <- datmodel[which(!is.na(datmodel$PV1FLIT)),]
sample <- fin_lit[sample(nrow(fin_lit)),]

# Create 10 equally size folds
set.seed(2019)
folds <- cut(seq(1, nrow(sample)), breaks = 10, labels = FALSE)

# Generate Vectors to Store Results
RMSE_bart <- Rsquared_bart <- MAE_bart <- c()

# Perform 10 fold cross validation
system.time({
for(i in 1:10){

  # Segment the data by fold using the which() function
  index <- which(folds==i,arr.ind=TRUE)
  test <- sample[index, ]
  train <- sample[-index, ]
  x_train <- train[,c(vars_prediction)]
  x_test <- test[,c(vars_prediction)]
  y_train <- as.vector(train[,c("PV1FLIT")])
  y_test <- as.vector(test[,c("PV1FLIT")])

  # Run BART model
  set.seed(2020)
  bart_cv <- wbart(x.train = x_train,
                   y.train = y_train,
                   x.test = x_test,
                   nskip = 1000,
                   ndpost = 1000)

  # Model Fit
  cv_fit <- postResample(bart_cv$yhat.test.mean, y_test)
  RMSE_bart[i] <- cv_fit[1]
  Rsquared_bart[i] <- cv_fit[2]
  MAE_bart[i] <- cv_fit[3]
}
})
```

```r
cbind(mean(RMSE_bart), mean(Rsquared_bart), mean(MAE_bart))
```

```
         [,1]      [,2]     [,3]
[1,] 57.96264 0.7332139 45.59895
```

The computing time is approximately 10 minutes on a `MacBook Pro` 2.4 GHz 8-Core Intel Core i9. The Root Mean Squared Error (RMSE) is ≈ 57.80, the R^2 is ≈ 0.7350 and the Mean Absolute Error (MAE) is ≈ 45.50.

RF

```
# Formula for RF
formula <- as.formula(paste("PV1FLIT ~", paste(vars_prediction, collapse="+")))

# Generate Vectors to Store Results
RMSE_rf <- Rsquared_rf <- MAE_rf <- c()

# Perform 10 fold cross validation
system.time({
for(i in 1:10){

  # Segment the data by fold using the which() function
  index <- which(folds==i,arr.ind=TRUE)
  test <- sample[index, ]
  train <- sample[-index, ]
  x_train <- train[,c(vars_prediction)]
  x_test <- test[,c(vars_prediction)]
  y_train <- as.vector(train[,c("PV1FLIT")])
  y_test <- as.vector(test[,c("PV1FLIT")])
  y_train <- as.data.frame(y_train)
  colnames(y_train) <- ("PV1FLIT")

  # Run RF model
  set.seed(2020)
  rf <- randomForest(formula, data = as.data.frame(cbind(y_train, x_train)),
                     na.action = na.roughfix)

  # Model Fit
  fitted_rf <- predict(rf, x_test,  type="response")
  cv_fit <- postResample(y_test, fitted_rf)
  RMSE_rf[i] <- cv_fit[1]
  Rsquared_rf[i] <- cv_fit[2]
  MAE_rf[i] <- cv_fit[3]
}
})
```

```
cbind(mean(RMSE_rf), mean(Rsquared_rf), mean(MAE_rf))
```

```
       [,1]      [,2]     [,3]
[1,] 58.678 0.7275555 46.05601
```

The computing time is 10 minutes on a `MacBook Pro` 2.4 GHz 8-Core Intel Core i9. The RMSE is 58.7810, the R^2 is 0.7252 and the MAE is 46.1493. Hence, we chose to use BART as its performance is slightly better than the performance of RF and the computational time of the two algorithms is similar.

ML Analysis to Predict FLS for Wallonia

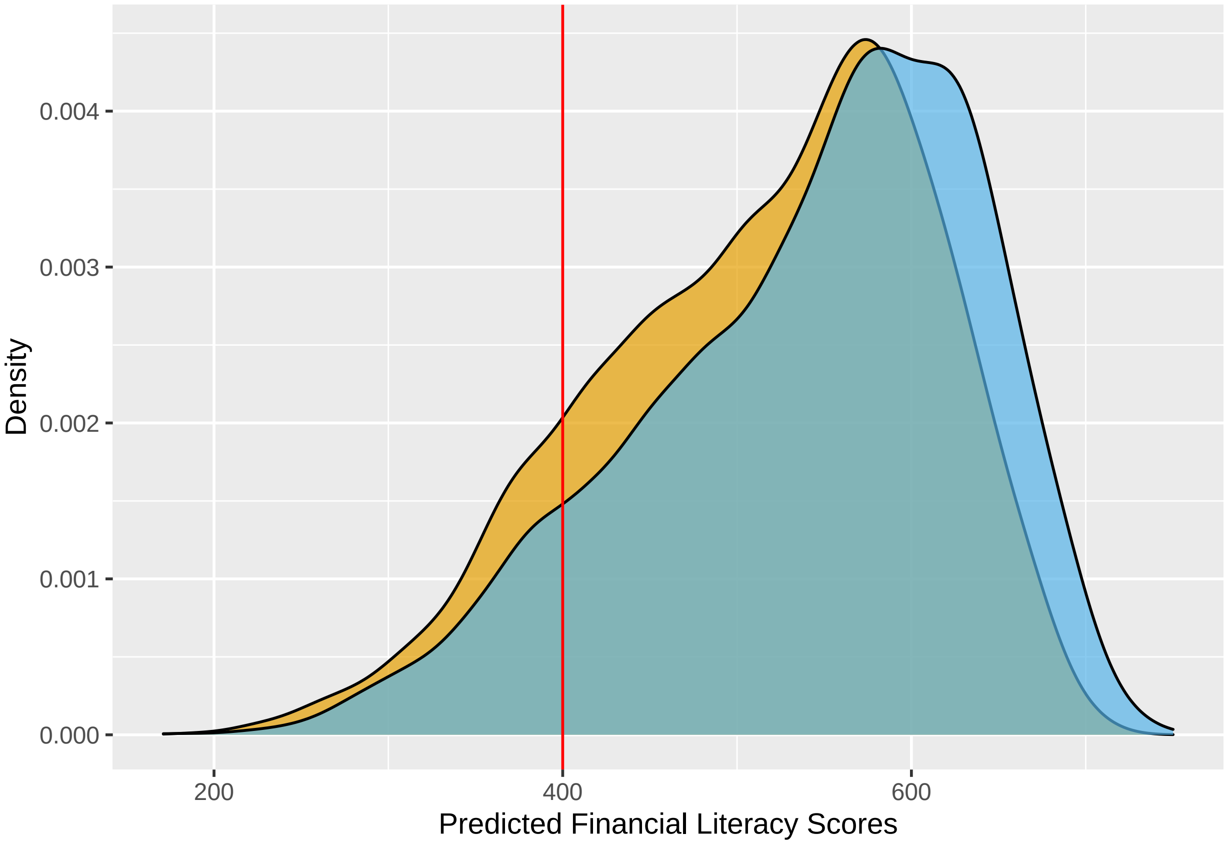
```
x_train <- datmodel[,c(vars_prediction)][which(!is.na(datmodel$PV1FLIT)),]
x_test <- datmodel[,c(vars_prediction)][which(is.na(datmodel$PV1FLIT)),]
y_train <- as.vector(datmodel[,c("PV1FLIT")][which(!is.na(datmodel$PV1FLIT))])

set.seed(2020)
main_bart <- wbart(x.train = x_train,
                    y.train = y_train,
                    x.test = x_test,
                    nskip=1000,
                    ndpost=1000)
```

```
# Wallonia is yellow, Flanders is blue
results_bart<-as.data.frame(cbind(main_bart$yhat.test.mean,main_bart$yhat.train.mean))
```

```
Warning in base::cbind(...): number of rows of result is not a multiple of
vector length (arg 1)
```

```
ggplot() +
  geom_density(data = results_bart,
  aes(V1), # Results for Wallonia
  fill = "#E69F00", color = "black", alpha = 0.7) +
  geom_density(data = results_bart,
  aes(V2), # Results for Flanders
  fill = "#56B4E9", color = "black", alpha = 0.7) +
  xlab("Predicted Financial Literacy Scores") + ylab("Density") +
  geom_vline(xintercept = 400, color = "red")
```



```
summary(main_bart$yhat.test.mean)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  189.2   442.8   522.8   510.5   583.5   715.5 
```

```
sd(main_bart$yhat.test.mean)
```

```
[1] 93.56914
```

```
summary(main_bart$yhat.train.mean)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  170.9   477.7   559.2   541.4   616.6   750.1 
```

```
sd(main_bart$yhat.train.mean)
```

```
[1] 96.83687
```

ML Analysis to Discover Students with low FLSs

In the following chunks of code, we introduce the functions build to predict the posterior distribution of the financial literacy scores and to detect the observation with posterior predicted financial literacy scores statistically significantly different than the posterior predicted sample mean score.

The function is called by using the `outlier_bart` command. This function computes the posterior distribution of both train and test data and it outputs the means and standard deviation of each unit level posterior prediction.

The function needs to specify: (i) a matrix of predictors in the training sample (`x_train`), (ii) a matrix of predictors in the test sample (`x_test`), (iii) a vector of observed values for the financial literacy score in the test sample (`y_test`), (iv) a number of burn-in samples (iterations) to be discarded from the computation of the posterior means and standard deviation (`nburn`), (v) a number of iterations used for the computation (`nsamp`), (vi) an outlier coefficient (`outlier_coef`): e.g., if this coefficient is set to 2, it means that the algorithm will consider as outliers those observations with posterior mean larger than two standard deviations from the mean (i.e., observation outside the 95% confidence intervals for the mean value) and/or two absolute deviations from the median.

The function outputs: (i) observations with posterior predicted financial literacy scores significantly lower (`low outliers (sd)`) or higher (`high outliers (sd)`) than the posterior predicted sample mean score (the significance level can be tuned by changing the `outlier_coef` parameter); (ii) observations with posterior predicted financial literacy scores k (this parameter is set using the \texttt{outlier_coef}) absolute deviations lower (`low outliers (sd)`) or higher (`high outliers (sd)`) than the predicted sample median score.

```
x_train <- datmodel[,c(vars_prediction)][which(!is.na(datmodel$PV1FLIT)),]
x_test <- datmodel[,c(vars_prediction)][which(is.na(datmodel$PV1FLIT)),]
y_train <- as.vector(datmodel[,c("PV1FLIT")][which(!is.na(datmodel$PV1FLIT))])

set.seed(2019)
bart_post <- outliers_bart(x_train=x_train,
                     y_train=y_train,
                     x_test=x_test,
                     nburn=1000,
                     nsamp=1000,
                     outlier_coeff = 2)
```

Below the number of observations with posterior predicted financial literacy scores significantly lower than the posterior predicted sample mean score.

```
# Outliers (2 standard deviations from the mean)
nrow(as.data.frame(bart_post$‘Low outliers (sd)‘))
```

```
[1] 1391
```

```
nrow(as.data.frame(bart_post$‘Low outliers (sd)‘))/nrow(datmodel)
```

```
[1] 0.1441301
```

```
# Outliers (2 absolute deviation from the median)
nrow(as.data.frame(bart_post$‘Low outliers (mad)‘))
```

```
[1] 390
```

```
nrow(as.data.frame(bart_post$‘Low outliers (mad)‘))/nrow(datmodel)
```

```
[1] 0.04041032
```

Below the number of observations with posterior predicted financial literacy scores significantly higher than the posterior predicted sample mean score.

```
# Outliers (2 standard deviations from the ean)
nrow(as.data.frame(bart_post$`High outliers (sd)`))
```

```
[1] 936
```

```
nrow(as.data.frame(bart_post$`High outliers (sd)`))/nrow(datmodel)
```

```
[1] 0.09698477
```

```
# Outliers (2 absolute deviation from the median)
nrow(as.data.frame(bart_post$`High outliers (mad)`))
```

```
[1] 1
```

```
nrow(as.data.frame(bart_post$`High outliers (mad)`))/nrow(datmodel)
```

```
[1] 0.0001036162
```

```
low <- bart_post$`Low outliers (sd)`
high <- bart_post$`High outliers (sd)`
data <- bart_post$Data
```

Sensitivity Analysis

Here we perform the "sensitivity analysis" for the predictions obtained from the BART algorithm. Since these predictions are the foundations of our paper, it is important to check whether or not they are stable. Let's now see in detail how we implement these robustness checks in `R`.

The function `sensitivity_bart` needs to specify: (i) a matrix of predictors in the training sample (`x_train`), (ii) a matrix of predictors in the test sample (`x_test`), (iii) a vector of observed values for the financial literacy score in the training sample (`y_train`), (iv) a number of burn-in samples (iterations) to be discarded from the computation of the posterior means and standard deviation (`nburn`), (v) a number of iterations used for the computation (`nsamp`), (vi) the level of significance for computing differences between posterior predicted values (original model vs augmented model): `alpha`.

The function outputs: (i) `rmse_original`: estimated RMSE for the original model, (ii) `rmse_augmented`: estimated RMSE for the augmented model, (iii) `rsquared_original`: estimated R squared for the original model, (iv) `rsquared_augmented`: estimated R squared for the augmented model.

```
set.seed(2019)
sensitivity <- sensitivity_bart(x_train=x_train,
                     y_train=y_train,
                     x_test=x_test,
                     nburn=1000,
                     nsamp=1000,
                     alpha = 0.1)
```

```
sensitivity
```

```
$‘Proportion of statistically different PPVs‘
[1] 0 0 0 0 0

$‘RMSE original model‘
[1] 54.44526

$‘RMSE augmented model‘
[1] 54.27522 54.32749 53.91954 53.21707 52.54359

$‘Rsquared original model‘
[1] 0.7644546

$‘Rsquared augmented model‘
[1] 0.7659211 0.7654583 0.7689740 0.7749595 0.7805976
```

Testing Differences in RMSE and R squared

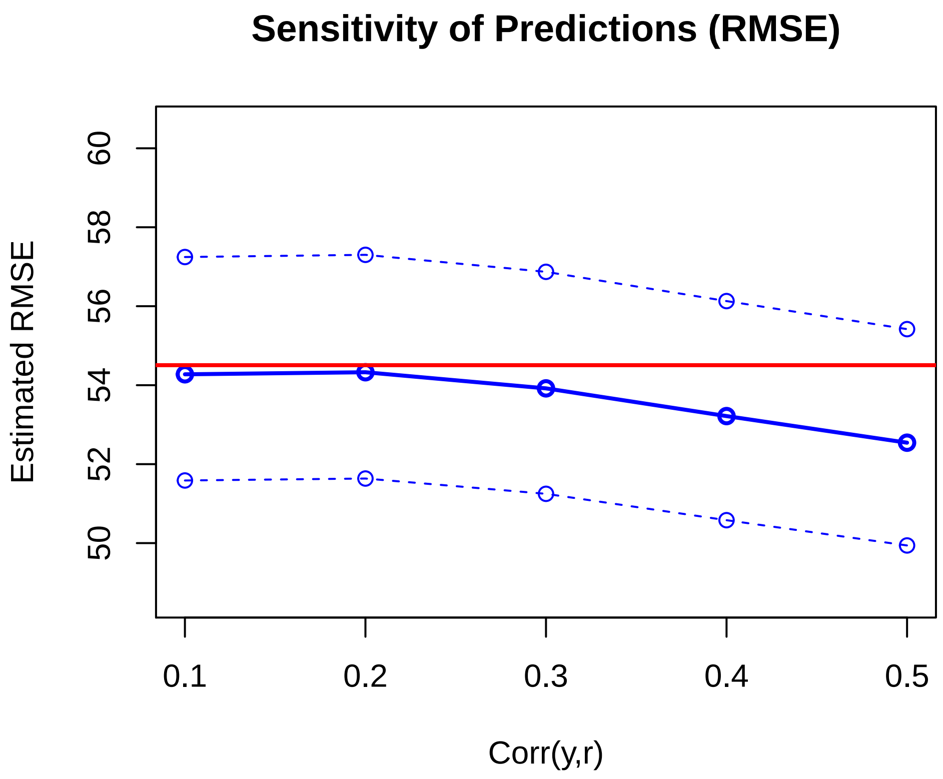
In order to test if the predictive performance of the augmented model is better than the one of the original model we construct the condidence intervals for both the RMSE and R^2 of the augmented model and we check if they overlap the values for the original model. In the following we depict the 99% confidence intervals

RMSE

In the following chunk of code we construct confidence intervals for the $RMSE$. The standard error of the $RMSE$ can be derived as follows:

$$se_{RMSE} = \sqrt{\frac{n}{\chi^2_{1-\alpha,n}}} \cdot RMSE. \tag{1}$$

```
par(mar=c(5.1, 4.1, 4.1, 8.1), xpd=TRUE)
plot(sensitivity$‘RMSE augmented model‘,
     main = "Sensitivity of Predictions (RMSE)",
     xlab = "Corr(y,r)",
     ylab = "Estimated RMSE",
     xaxt=’n’,
     type = "o",
     col = "blue",
     lwd = 2,
     ylim=c(min(postResample(main_bart$yhat.train.mean,
                             y_train)[1]*sqrt(1000/qchisq(0.05,df = 1000)))-8,
            max(postResample(main_bart$yhat.train.mean,
                             y_train)[1]*sqrt(1000/qchisq(0.95,df = 1000)))+8))
par(xpd = FALSE)
lines(sensitivity$‘RMSE augmented model‘*sqrt(1000/qchisq(0.99,df = 1000)),
      col = "blue", lty=2, type = "o")
lines(sensitivity$‘RMSE augmented model‘*sqrt(1000/qchisq(0.01,df = 1000)),
      col = "blue", lty=2, type = "o")
abline(h= postResample(main_bart$yhat.train.mean, y_train)[1], col = "red", lwd = 2)
axis(1, at=1:(length(sensitivity$‘RMSE augmented model‘)), labels=c(seq(0.1,0.5,0.1)))
```



R squared

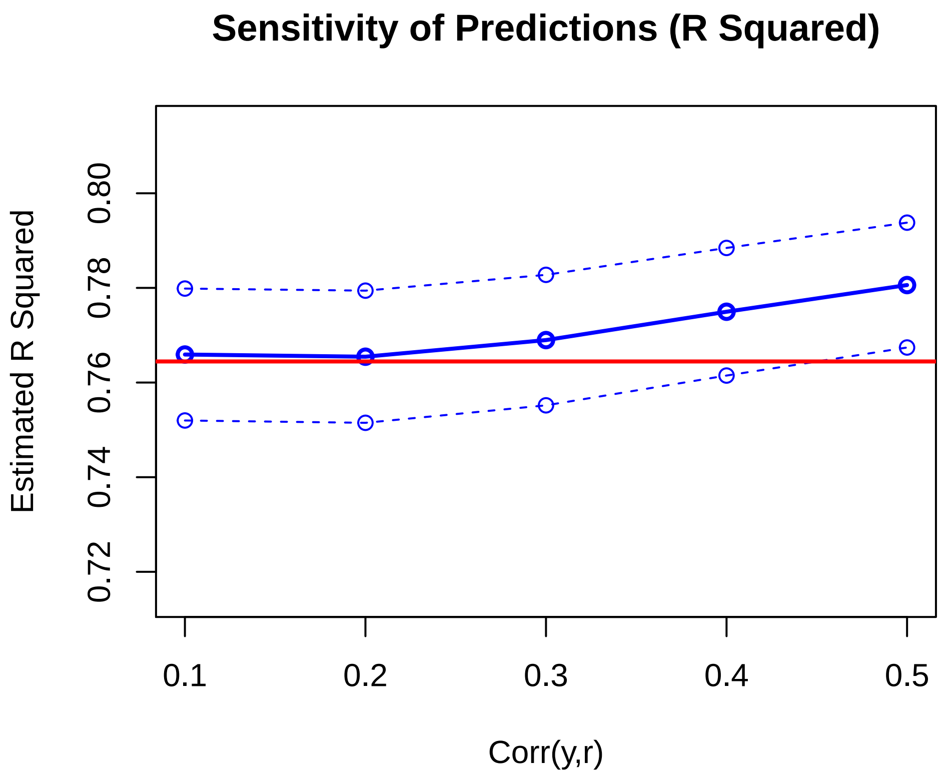
In the following chunk of code we construct confidence intervals for the R^2. The standard error of the R^2 can be derived as follows:

$$se_{R^2} = \sqrt{\frac{4R^2(1-R^2)^2(n-k-1)^2}{(n^2-1)(n+3)}} \tag{2}$$

where k is the number of predictors. See Cohen et al. (2003), “Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences”, p. 88 for details.

```
r2 = sensitivity$‘Rsquared original model‘
n = nrow(x_train)
k = ncol(x_train)
ub_new <- lb_new <- c()
for(j in (1:length(sensitivity$‘Rsquared augmented model‘))){
  r2_new = sensitivity$‘Rsquared augmented model‘[j]
  se_r2_new <- sqrt((4*r2_new*(1-r2_new)^2*(n-k-1)^2)/((n^2-1)*(n+3)))
  ub_new[j] = r2_new + 2.58*se_r2_new # Change Z-score for difference alpha levels
  lb_new[j] = r2_new - 2.58*se_r2_new # Change Z-score for difference alpha levels
}

par(mar=c(5.1, 4.1, 4.1, 8.1), xpd=TRUE)
plot(sensitivity$‘Rsquared augmented model‘,
     main = "Sensitivity of Predictions (R Squared)",
     xlab = "Corr(y,r)",
     ylab = "Estimated R Squared",
     xaxt=’n’,
     type = "o",
     col = "blue",
     lwd = 2,
     ylim=c(r2-0.05,r2+0.05))
par(xpd = FALSE)
lines(lb_new, col = "blue", lty=2, type = "o")
lines(ub_new, col = "blue", lty=2, type = "o")
abline(h= r2, col = "red", lwd = 2)
axis(1, at=1:(length(sensitivity$‘Rsquared augmented model‘)), labels=c(seq(0.1,0.5,0.1)))
```



Standardized Difference in Means

Moreover, the standardized difference in the means between $\hat{p}(Y_i = 1|X_i = x)$ and $\hat{p}(Y_i = 1|X_i = x, G_i = g)$ is not significant in all the cases.

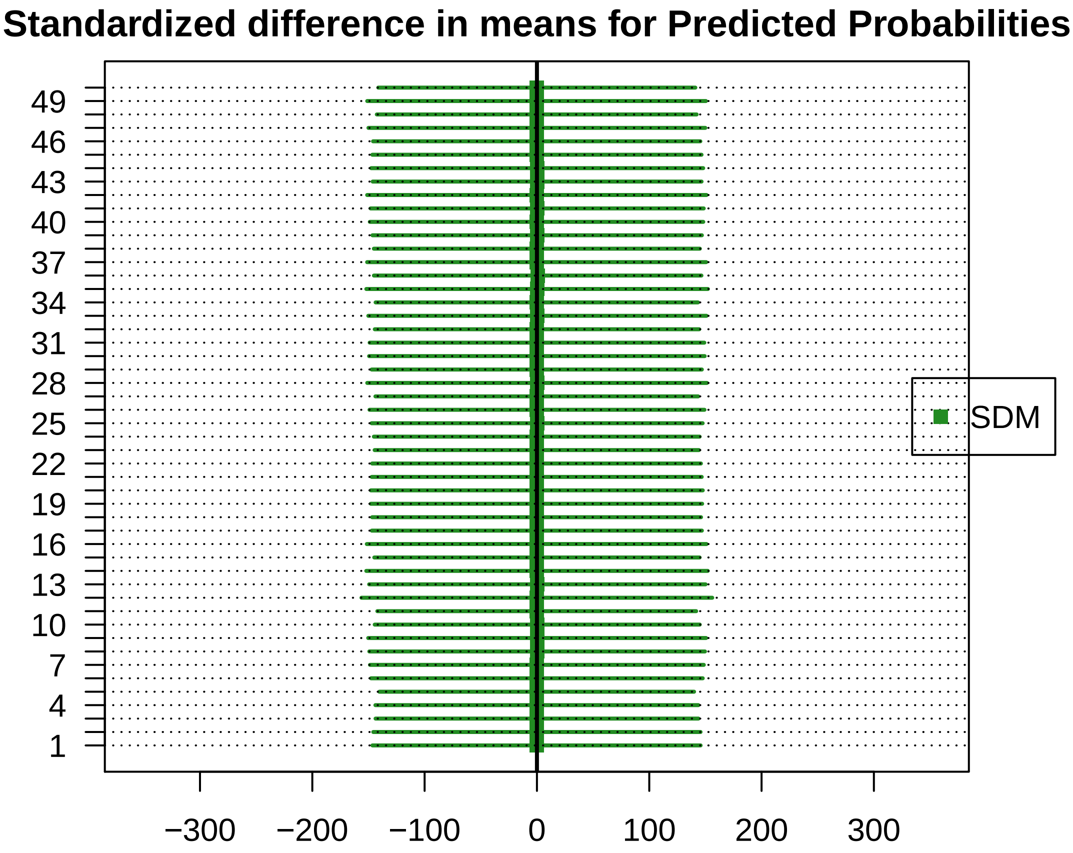
Standardized differences in mean and their standard deviations from Cohen (1988, p.44).

Standardized differences in mean and their standard deviations from Cohen (1988, p.44).

```
# Standardized difference in means
diff.means <- ppd_mean - new_ppd_mean
standard.diff.means <-  (diff.means)/sqrt((ppd_sd^2 + new_ppd_sd^2)/2)

# 99% CI (t-student distribution)
x0 <- standard.diff.means - 2.58 *
  sqrt((ppd_sd^2 + new_ppd_sd^2)/2)
x1 <-standard.diff.means  + 2.58 *
  sqrt((ppd_sd^2 + new_ppd_sd^2)/2)
```

This can be seen from the plot of the Standardized difference in means for the probabilities predicted by the two models.


Standardized difference in means for Predicted Probabilities
49
46
43
40
37
34
31
28
25
22
19
16
13
10
7
4
1
−300
−200
−100
0
100
200
300
🟩 SDM