Analyses in Main Paper

## Method

### Dataset

data <- read.delim("data.csv", header = TRUE)  
  
## Items should be scored 1-5, 0 may be missings  
data[ , 1:48][sapply(data[ , 1:48], function(x) x == 0)] <- NA  
data <- data[complete.cases(data[ , 1:48]), ]  
  
## Select only university students  
data <- data[data$education >= 3, ]  
# table(data$major)  
psych\_ids <- rowSums(sapply(c("psych", "psyhcology", "psycotherapy", "couns",   
 "behavior", "behaviour", "neuro"),  
 function(x) grepl(x, data$major, ignore.case = TRUE)))  
anim\_ids <- grepl("anim", data$major, ignore.case = TRUE) ## exclude animal psych  
data$major <- factor(ifelse(psych\_ids > 0, "psychology", "other"))  
data$major[anim\_ids > 0 & psych\_ids > 0] <- "other"  
  
set.seed(42)  
test\_ids <- sample(1:nrow(data), ceiling(nrow(data)/4))  
train\_ids <- which(!1:nrow(data) %in% test\_ids)  
train\_y <- as.numeric(data$major)[train\_ids] - 1  
test\_y <- as.numeric(data$major)[test\_ids] - 1  
  
data$Real <- rowSums(data[ , paste0("R", 1:8)])  
data$Inve <- rowSums(data[ , paste0("I", 1:8)])  
data$Arti <- rowSums(data[ , paste0("A", 1:8)])  
data$Soci <- rowSums(data[ , paste0("S", 1:8)])  
data$Ente <- rowSums(data[ , paste0("E", 1:8)])  
data$Conv <- rowSums(data[ , paste0("C", 1:8)])

format(nrow(data), nsmall = 0, big.mark = ",")

## [1] "55,593"

format(unname(round(100\*table(data$major)[2] / nrow(data), digits = 2)),   
 nsmall = 0, big.mark = ",")

## [1] "19.42"

format(unname(round(100\*table(data$major)[1] / nrow(data), digits = 2)),   
 nsmall = 0, big.mark = ",")

## [1] "80.58"

### Model fitting and evaluation

format(length(train\_ids), nsmall = 0, big.mark = ",")

## [1] "41,694"

format(unname(round(100\*table(data$major[train\_ids])[2] / length(train\_ids),   
 digits = 2)), nsmall = 0, big.mark = ",")

## [1] "19.46"

format(length(test\_ids), nsmall = 0, big.mark = ",")

## [1] "13,899"

format(unname(round(100\*table(data$major[test\_ids])[2] / length(test\_ids),   
 digits = 2)), nsmall = 0, big.mark = ",")

## [1] "19.3"

paste0(R.Version()$major, ".", R.Version()$minor)

## [1] "4.1.0"

## Results

varnames\_i <- paste0(rep(c("R", "I", "A", "S", "E", "C"), each = 8), 1:8)  
varnames\_s <- c("R", "I", "A", "S", "E", "C")

### (Penalized) Logistic regression

glmod\_s <- glm(major ~ Real + Inve + Arti + Soci + Ente + Conv,   
 data = data[train\_ids , ], family = "binomial")  
#summary(glmod\_s)  
glm\_preds\_train\_s <- predict(glmod\_s, newdata = data[train\_ids, ], type = "response")  
glm\_preds\_test\_s <- predict(glmod\_s, newdata = data[test\_ids, ], type = "response")

library("glmnet")  
X <- as.matrix(data[train\_ids, varnames\_i])  
set.seed(42)   
glmod\_i <- glmnet(X, train\_y, family = "binomial", alpha = 1, lambda = 0.0003568404)  
glm\_preds\_train\_i <- predict(glmod\_i, newx = X, type = "response")  
glm\_preds\_test\_i <- predict(glmod\_i, newx = as.matrix(data[test\_ids, varnames\_i]),   
 type = "response")

### Generalized additive model

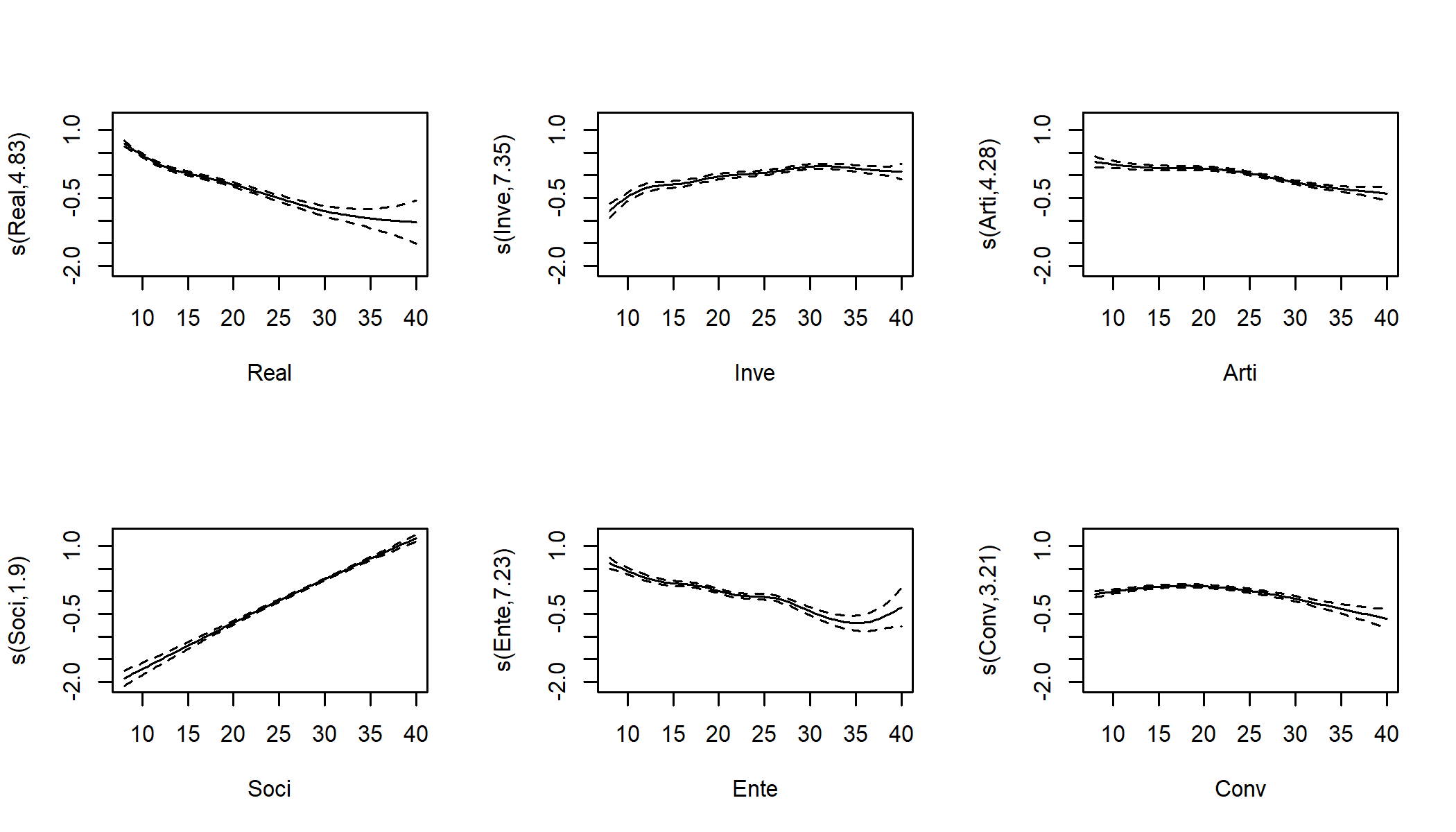
library("mgcv")  
gamod\_s <- gam(major ~ s(Real) + s(Inve) + s(Arti) + s(Soci) + s(Ente) + s(Conv),   
 data = data[train\_ids , ], family = "binomial")

## Loading required package: nlme

## This is mgcv 1.8-35. For overview type 'help("mgcv-package")'.

**Figure 1**

par(mfrow = c(2, 3))  
plot(gamod\_s)



gam\_preds\_train\_s <- predict(gamod\_s, newdata = data[train\_ids, ], type = "response")  
gam\_preds\_test\_s <- predict(gamod\_s, newdata = data[test\_ids, ], type = "response")

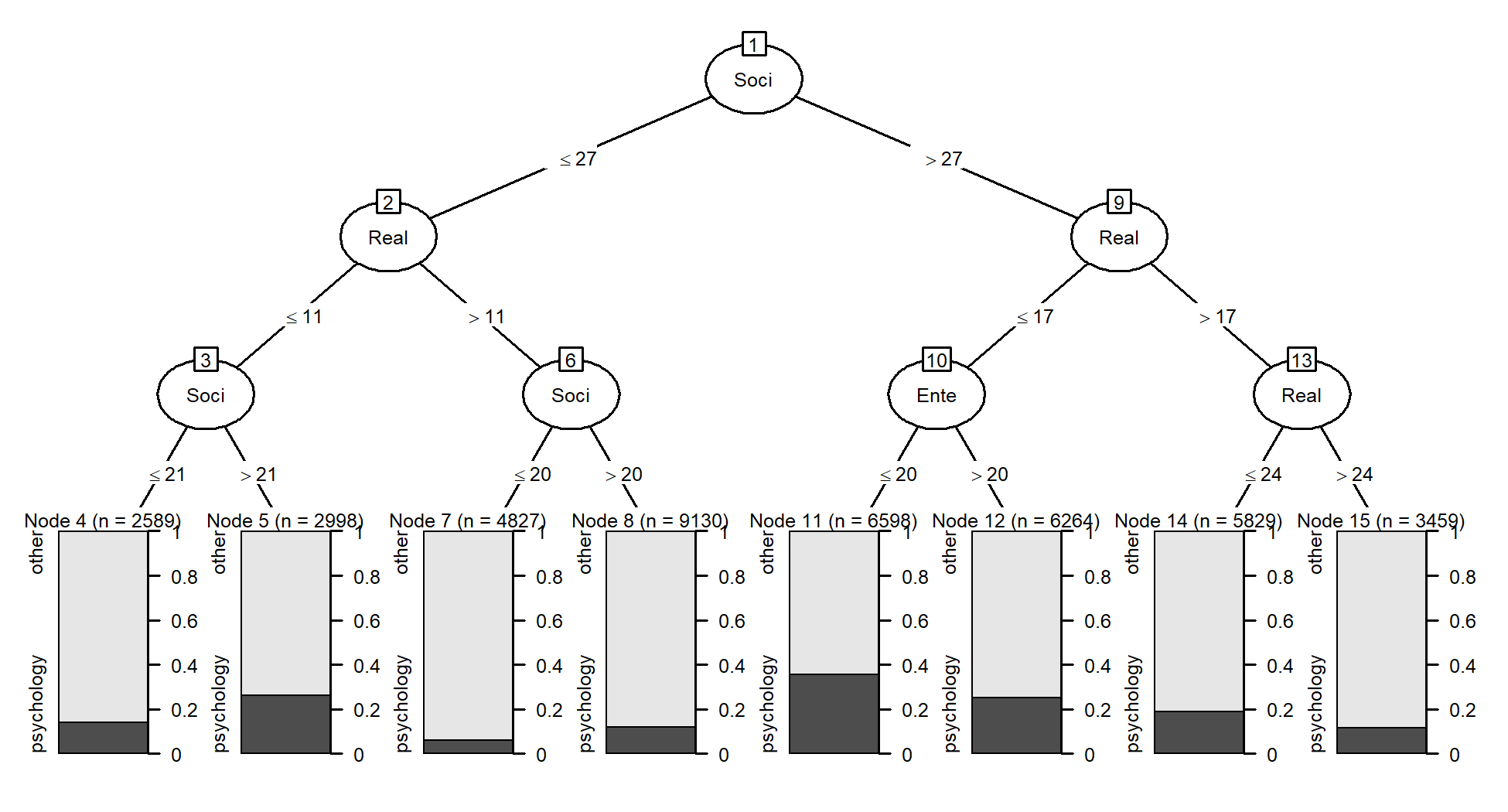
gamod\_i <- gam(gam\_form, data = data[train\_ids , ], family = "binomial",   
 select = TRUE)

gam\_preds\_train\_i <- predict(gamod\_i, newdata = data[train\_ids, ], type = "response")  
gam\_preds\_test\_i <- predict(gamod\_i, newdata = data[test\_ids, ], type = "response")

### Decision tree

**Figure 2**

library("partykit")  
ct\_s <- ctree(major ~ Real + Inve + Arti + Soci + Ente + Conv, data = data[train\_ids , ],   
 maxdepth = 7)  
ct3 <- ctree(major ~ Real + Inve + Arti + Soci + Ente + Conv, data = data[train\_ids , ],   
 maxdepth = 3)  
plot(ct3, gp = gpar(cex = .5), ip\_args = list(pval = FALSE))



ct\_preds\_train\_s <- predict(ct\_s, type = "prob")[ , "psychology"]  
ct\_preds\_test\_s <- predict(ct\_s, newdata = data[test\_ids , ], type = "prob")[ , "psychology"]

ct\_form <- formula(paste("major ~", paste(varnames\_i, collapse = "+")))  
ct <- ctree(ct\_form, data = data[train\_ids , ], maxdepth = 9)  
ct\_preds\_train\_i <- predict(ct, type = "prob")[ , "psychology"]  
ct\_preds\_test\_i <- predict(ct, newdata = data[test\_ids , ], type = "prob")[ , "psychology"]

### Gradient boosted tree ensemble

library("gbm")  
set.seed(42)  
gb\_s <- gbm(I(as.numeric(major)-1) ~ Real + Inve + Arti + Soci + Ente + Conv,   
 n.trees = 1100, interaction.depth = 3L, shrinkage = 0.01,   
 data = data[train\_ids , ])

gb\_preds\_train\_s <- predict(gb\_s, newdata = data[train\_ids, ], type = "response")

## Using 1100 trees...

gb\_preds\_test\_s <- predict(gb\_s, newdata = data[test\_ids, ], type = "response")

## Using 1100 trees...

library("gbm")  
set.seed(42)  
gbm\_form <- formula(paste("I(as.numeric(major)-1) ~ ",   
 paste(paste0(rep(c("R", "I", "A", "S", "E", "C"),   
 each = 8), 1:8), collapse = "+")))  
gb\_i <- gbm(gbm\_form, n.trees = 3500, interaction.depth = 5L, shrinkage = 0.01,   
 data = data[train\_ids , ])  
sum\_i <- summary(gb\_i, plotit = FALSE, method = permutation.test.gbm)

gb\_preds\_train\_i <- predict(gb\_i, newdata = data[train\_ids, ], type = "response")

## Using 3500 trees...

gb\_preds\_test\_i <- predict(gb\_i, newdata = data[test\_ids, ], type = "response")

## Using 3500 trees...

### Random forest

library("ranger")  
set.seed(42)  
rf\_s <- ranger(major ~ Real + Inve + Arti + Soci + Ente + Conv, data = data[train\_ids , ],  
 probability = TRUE, mtry = 3L, min.node.size = 500,   
 importance = "permutation")

rf\_preds\_train\_s <- predict(rf\_s, data = data[train\_ids, ])$predictions[ , "psychology"]  
rf\_preds\_test\_s <- predict(rf\_s, data = data[test\_ids, ])$predictions[ , "psychology"]

set.seed(42)  
varnames <- paste0(rep(c("R", "I", "A", "S", "E", "C"), each = 8), 1:8)  
rf\_form <- formula(paste("major ~", paste(varnames, collapse = "+")))  
rf\_i <- ranger(rf\_form, data = data[train\_ids , ], probability = TRUE,   
 mtry = 10L, min.node.size = 500, importance = "permutation")

rf\_preds\_train\_i <- predict(rf\_i, data = data[train\_ids, ])$predictions[ , "psychology"]  
rf\_preds\_test\_i <- predict(rf\_i, data = data[test\_ids, ])$predictions[ , "psychology"]

### Prediction rule ensembling

**Table 1**

pr\_preds\_train\_s <- predict(pr\_s, type = "response")  
pr\_preds\_test\_s <- predict(pr\_s, newdata = data[test\_ids , ], type = "response")  
imps <- pre::importance(pr\_s, plot=FALSE)  
varimps\_pre <- imps$varimps  
imps <- imps$baseimps[1:6, c("description", "coefficient")]  
colnames(imps) <- c("Description", "Coefficient")  
imps$Coefficient <- round(imps$Coefficient, digits = 3)  
kable(imps, row.names = FALSE, align = c("l", "c"))

|  |  |
| --- | --- |
| Description | Coefficient |
| Soci > 27 & Ente <= 31 & Conv <= 30 | 0.182 |
| Soci > 23 & Ente <= 29 & Real <= 24 | 0.181 |
| Real > 10 & Soci <= 35 | -0.175 |
| Real <= 22 & Soci > 19 & Inve > 18 | 0.138 |
| Inve > 10 & Real <= 13 | 0.120 |
| Conv <= 23 & Arti <= 29 & Soci > 21 | 0.112 |

pr\_preds\_train\_i <- predict(pr\_i, type = "response")  
pr\_preds\_test\_i <- predict(pr\_i, newdata = data[test\_ids , ], type = "response")

### *k* Nearest neighbours

library("class")  
  
## Model for training predictions  
knn\_mod <- knn(train = data[train\_ids , varnames\_s],   
 test = data[train\_ids , varnames\_s],   
 cl = as.factor(data[train\_ids, "major"]),   
 k = 300, use.all = TRUE, prob = TRUE)  
## Need to obtain predicted probability for second class  
knn\_preds\_train\_s <- ifelse(  
 knn\_mod == "psychology", attr(knn\_mod, "prob"), 1 - attr(knn\_mod, "prob"))

## Model for testing predictions  
knn\_mod <- knn(train = data[train\_ids , varnames\_s],   
 test = data[test\_ids , varnames\_s],   
 cl = as.factor(data[train\_ids, "major"]),   
 k = 300, use.all = TRUE, prob = TRUE)  
knn\_preds\_test\_s <- ifelse(  
 knn\_mod == "psychology", attr(knn\_mod, "prob"), 1 - attr(knn\_mod, "prob"))

## Model for training predictions  
knn\_mod <- knn(train = data[train\_ids , varnames\_i],   
 test = data[train\_ids , varnames\_i],   
 cl = as.factor(data[train\_ids, "major"]),   
 k = 100, use.all = TRUE, prob = TRUE)  
## Need to obtain predicted probability for second class  
knn\_preds\_train\_i <- ifelse(  
 knn\_mod == "psychology", attr(knn\_mod, "prob"), 1 - attr(knn\_mod, "prob"))

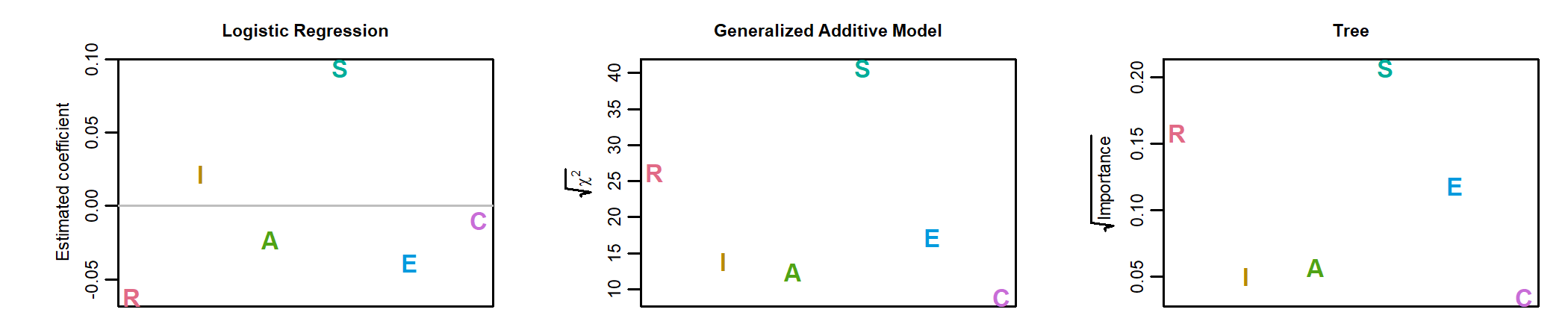
## Model for testing predictions  
knn\_mod <- knn(train = data[train\_ids , varnames\_i],   
 test = data[test\_ids , varnames\_i],   
 cl = as.factor(data[train\_ids, "major"]),   
 k = 100, use.all = TRUE, prob = TRUE)  
knn\_preds\_test <- ifelse(  
 knn\_mod == "psychology", attr(knn\_mod, "prob"), 1 - attr(knn\_mod, "prob"))

## Model comparisons

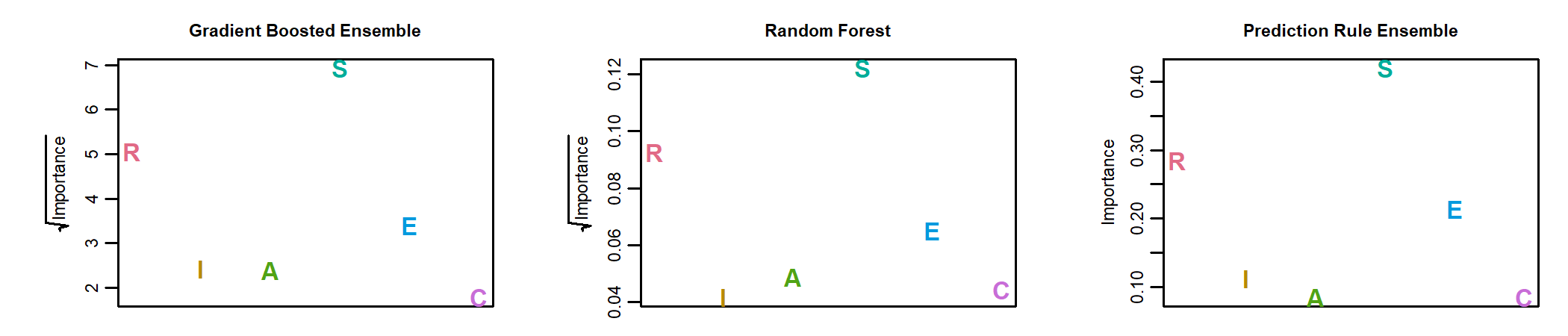
### Variable contributions

**Figure 3**

par(mfrow = c(1, 3))  
par(mar = c(1, 4, 2, 1), mgp = c(1.5, .5, 0), tck = -0.05)  
library("colorspace")  
  
## Logistic regression  
plot(coef(glmod\_s)[-1], xaxt = "n", ylab = "Estimated coefficient",  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",   
 main = "Logistic Regression", cex.main = .7)  
text(coef(glmod\_s)[-1], labels = varnames\_s, cex = 1,   
 col = rep(qualitative\_hcl(6)), font = 2)  
abline(0, 0, col = "grey")  
  
## Generalized additive model  
sum <- summary(gamod\_s)  
plot(sqrt(sum$chi.sq), xaxt = "n", ylab = expression(sqrt(chi^2)),  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",   
 main = "Generalized Additive Model", cex.main = .7)  
text(sqrt(sum$chi.sq), labels = varnames\_s, cex = 1,   
 col = rep(qualitative\_hcl(6)), font = 2)  
  
## Conditional inference tree  
ct6 <- cforest(major ~ Real + Inve + Arti + Soci + Ente + Conv,   
 data = data[train\_ids , ], ntree = 1L, mtry = 6,  
 perturb = list(replace = FALSE, fraction = 1L),  
 control = ctree\_control(maxdepth = 6))  
imps <- varimp(gettree(ct6), risk = "loglik")  
imps <- imps[c("Real", "Inve", "Arti", "Soci", "Ente", "Conv")]  
plot(sqrt(imps), xaxt = "n", ylab = expression(sqrt(Importance)),  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",   
 main = "Tree", cex.main = .7)  
text(sqrt(imps), labels = varnames\_s, cex = 1,   
 col = rep(qualitative\_hcl(6)), font = 2)

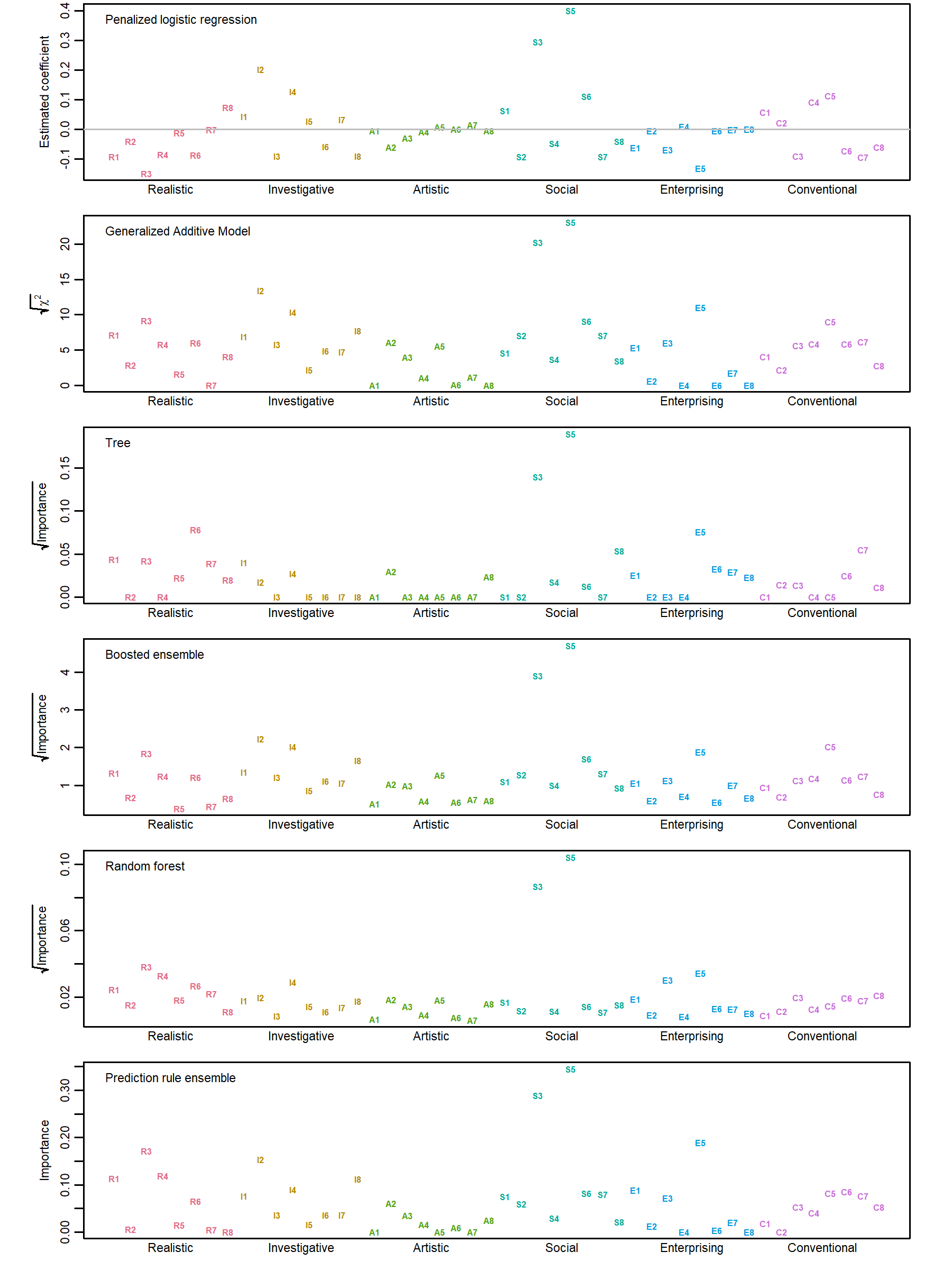


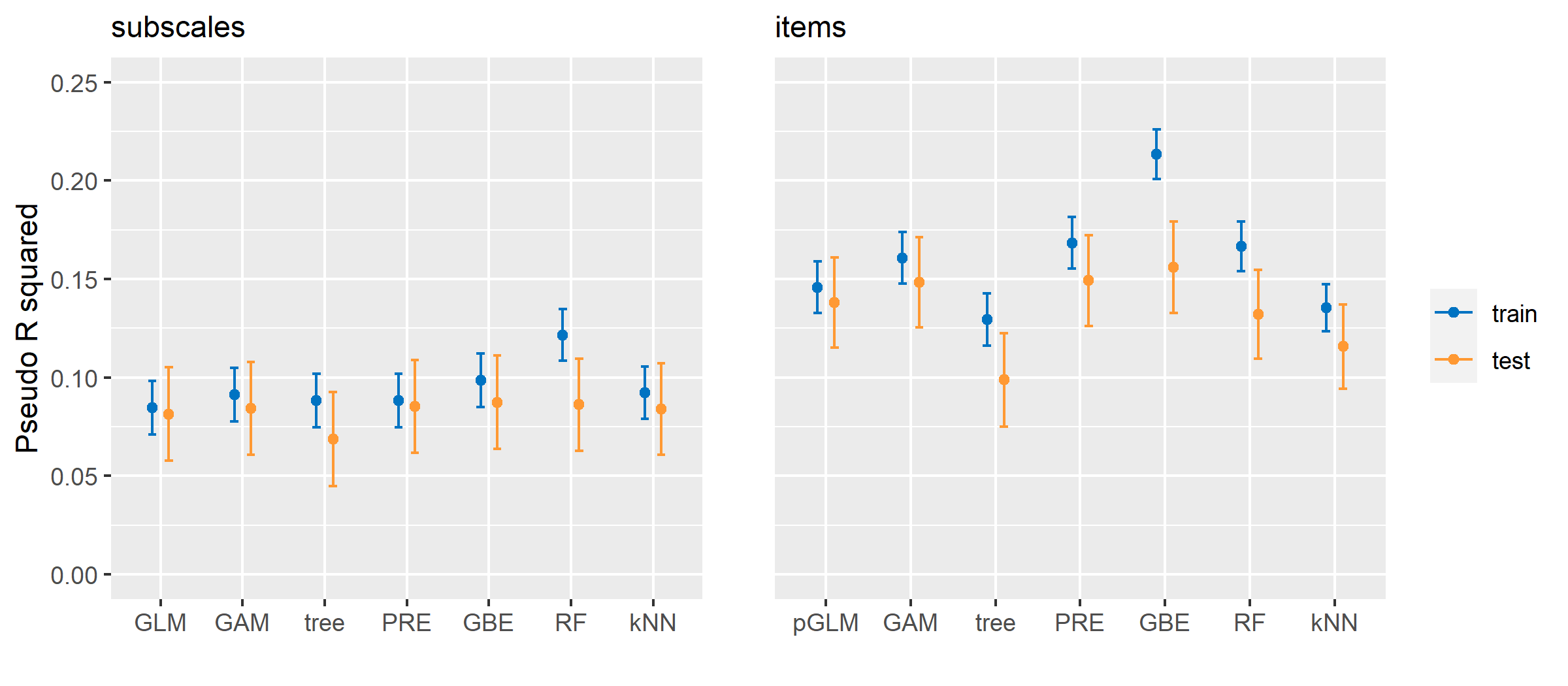
## Gradient boosted ensemble  
sum <- summary(gb\_s, plotit = FALSE, method = permutation.test.gbm)  
imps <- sum$rel.inf  
names(imps) <- sum$var  
imps <- imps[c("Real", "Inve", "Arti", "Soci", "Ente", "Conv")]  
plot(sqrt(imps), xaxt = "n", ylab = expression(sqrt(Importance)),  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",   
 main = "Gradient Boosted Ensemble", cex.main = .7)  
text(sqrt(imps), labels = varnames\_s, cex = 1,   
 col = rep(qualitative\_hcl(6)), font = 2)  
  
## Random forest  
library("ranger")  
load(file = "RF\_subscales.Rda")  
imps <- ranger::importance(rf\_s)  
plot(sqrt(imps), xaxt = "n", ylab = expression(sqrt(Importance)),  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",   
 main = "Random Forest", cex.main = .7)  
text(sqrt(imps), labels = varnames\_s, cex = 1,   
 col = rep(qualitative\_hcl(6)), font = 2)  
  
## Prediction rule ensemble  
imps <- varimps\_pre$imp  
names(imps) <- varimps\_pre$varname  
imps <- imps[c("Real", "Inve", "Arti", "Soci", "Ente", "Conv")]  
plot(imps, xaxt = "n", ylab = "Importance",  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",   
 cex.main = .7, main = "Prediction Rule Ensemble")  
text(imps, labels = varnames\_s, cex = 1,   
 col = rep(qualitative\_hcl(6)), font = 2)



**Figure 4**

par(mar = c(1.5, 4, 0.2, 2), mgp = c(1.5, .5, 0), tck = -0.05)  
par(mfrow = c(6, 1))  
  
library("glmnet")  
plot(coef(glmod\_i)[-1], xaxt = "n", ylab = "Estimated coefficient",  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",   
 main = " ", cex.main = .7)  
text(coef(glmod\_i)[-1], labels = varnames\_i, cex = .5,   
 col = rep(qualitative\_hcl(6), each = 8), font = 2)  
abline(0, 0, col = "grey")  
legend("topleft", legend = "Penalized logistic regression", cex = .7, bty = "n")  
axis(1, 4.5 + c(0:5)\*8, tick = FALSE, padj = -1.5,  
 labels = c("Realistic", "Investigative", "Artistic", "Social" ,   
 "Enterprising", "Conventional"),   
 cex.axis = .7)  
  
library("mgcv")  
load(file = "GAM\_items.Rda")  
sum <- summary(gamod\_i)  
plot(sqrt(sum$chi.sq), xaxt = "n", ylab = expression(sqrt(chi^2)),   
 main = " ", cex.main = .7,  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ")  
text(sqrt(sum$chi.sq), labels = varnames\_i, cex = .5,   
 col = rep(qualitative\_hcl(6), each = 8), font = 2)  
legend("topleft", legend = "Generalized Additive Model", cex = .7, bty = "n")  
axis(1, 4.5 + c(0:5)\*8, tick = FALSE, padj = -1.5,  
 labels = c("Realistic", "Investigative", "Artistic", "Social" ,   
 "Enterprising", "Conventional"),   
 cex.axis = .7)  
  
ct6 <- cforest(ct\_form,   
 data = data[train\_ids , ], ntree = 1L, mtry = 6,  
 perturb = list(replace = FALSE, fraction = 1L),  
 control = ctree\_control(maxdepth = 6))  
imps <- varimp(gettree(ct6), risk = "loglik")  
imp\_names <- names(imps)  
imps <- c(imps, rep(0, times = 48 - length(imps)))  
names(imps) <- c(imp\_names, varnames\_i[!varnames\_i %in% imp\_names])  
plot(sqrt(imps[varnames\_i]), xaxt = "n", ylab = expression(sqrt(Importance)),  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",   
 main = " ", cex.main = .7)  
text(sqrt(imps[varnames\_i]), labels = varnames\_i, cex = .5,   
 col = rep(qualitative\_hcl(6), each = 8), font = 2)  
legend("topleft", legend = "Tree", cex = .7, bty = "n")  
axis(1, 4.5 + c(0:5)\*8, tick = FALSE, padj = -1.5,  
 labels = c("Realistic", "Investigative", "Artistic", "Social" ,   
 "Enterprising", "Conventional"),   
 cex.axis = .7)  
  
load(file = "gb\_i\_summary.Rda")  
imps <- sum\_i[match(varnames\_i, sum\_i$var), ]  
plot(sqrt(imps$rel.inf), xaxt = "n", main = " ",  
 ylab = expression(sqrt(Importance)), cex.main = .7,  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ")  
text(sqrt(imps$rel.inf), labels = imps$var, cex = .5,   
 col = rep(qualitative\_hcl(6), each = 8), font = 2)  
legend("topleft", legend = "Boosted ensemble", cex = .7, bty = "n")  
axis(1, 4.5 + c(0:5)\*8, tick = FALSE, padj = -1.5,  
 labels = c("Realistic", "Investigative", "Artistic", "Social" ,   
 "Enterprising", "Conventional"),   
 cex.axis = .7)  
  
imps <- ranger::importance(rf\_i)  
plot(sqrt(imps), xaxt = "n", main = " ",  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ",  
 ylab = expression(sqrt(Importance)), cex.main = .7)  
text(sqrt(imps), labels = names(imps), cex = .5,   
 col = rep(qualitative\_hcl(6), each = 8), font = 2)  
legend("topleft", legend = "Random forest", cex = .7, bty = "n")  
axis(1, 4.5 + c(0:5)\*8, tick = FALSE, padj = -1.5,  
 labels = c("Realistic", "Investigative", "Artistic", "Social" ,   
 "Enterprising", "Conventional"),   
 cex.axis = .7)  
  
imps <- pre::importance(pr\_i, cex.axis = .7, plot = FALSE)$varimps  
zero\_vars <- varnames\_i[!varnames\_i %in% imps[ , 1]]  
imps <- rbind(imps, data.frame(varname = zero\_vars,   
 imp = rep(0, times = length(zero\_vars))))  
imps <- imps[match(varnames\_i, imps$varname), ]  
plot(imps$imp, xaxt = "n", main = " ",  
 ylab = "Importance",  
 col = "white", cex.lab = .7, cex.axis = .7, xlab = " ", cex.main = .7)  
text(imps$imp, labels = imps$varname, cex = .5,   
 col = rep(qualitative\_hcl(6), each = 8), font = 2)  
legend("topleft", legend = "Prediction rule ensemble", cex = .7, bty = "n")  
axis(1, 4.5 + c(0:5)\*8, tick = FALSE, padj = -1.5,  
 labels = c("Realistic", "Investigative", "Artistic", "Social" ,   
 "Enterprising", "Conventional"),   
 cex.axis = .7)



**Figure 5**  


# Appendix B: Uni- and bivariate sample descriptives

data$age[data$age > 103] <- NA ## some crazy ages were supplied  
gender\_tab <- prop.table(table(data$gender)) ## 0=missing, 1=Male, 2=Female, 3=Other  
marital\_tab <- prop.table(table(data$married)) ## 0=missing, 1=Never married, 2=Currently married, 3=Previously married  
urban\_tab <- prop.table(table(data$urban)) ## 0=missing, 1=Rural (country side), 2=Suburban, 3=Urban (town, city)  
race\_tab <- prop.table(table(data$race)) ## 0=missing, 1=Asian, 2=Arab, 3=Black,   
## 4=Indigenous Australian / Native American / White, 5=Other (There was a coding mistake resulting in cat 4)  
language\_tab <- prop.table(table(data$engnat)) ## 0=missing, 1=Yes, 2=No

format(nrow(data), nsmall = 0, big.mark = ",")

## [1] "55,593"

format(mean(data$age, na.rm = TRUE), digits = 2, nsmall = 2, big.mark = ",")

## [1] "33.06"

format(sd(data$age, na.rm = TRUE), nsmall = 2, digits = 2, big.mark = ",")

## [1] "11.68"

format(100\*gender\_tab[3], nsmall = 0, digits = 0, big.mark = ",")

## 2   
## "67"

format(100\*gender\_tab[2], nsmall = 0, digits = 0, big.mark = ",")

## 1   
## "32"

format(100\*gender\_tab[4], nsmall = 0, digits = 0, big.mark = ",")

## 3   
## "1"

format(100\*marital\_tab[2], nsmall = 0, digits = 0, big.mark = ",")

## 1   
## "58"

format(100\*marital\_tab[3], nsmall = 0, digits = 0, big.mark = ",")

## 2   
## "33"

format(100\*marital\_tab[4], nsmall = 0, digits = 0, big.mark = ",")

## 3   
## "8"

format(100\*urban\_tab[2], nsmall = 0, digits = 0, big.mark = ",")

## 1   
## "22"

format(100\*urban\_tab[3], nsmall = 0, digits = 0, big.mark = ",")

## 2   
## "37"

format(100\*urban\_tab[4], nsmall = 0, digits = 0, big.mark = ",")

## 3   
## "40"

format(100\*language\_tab[2], nsmall = 0, digits = 0, big.mark = ",")

## 1   
## "66"

format(100\*language\_tab[3], nsmall = 0, digits = 0, big.mark = ",")

## 2   
## "34"

format(100\*race\_tab[2], nsmall = 0, digits = 0, big.mark = ",")

## 1   
## "20"

format(100\*race\_tab[3], nsmall = 0, digits = 0, big.mark = ",")

## 2   
## "1"

format(100\*race\_tab[4], nsmall = 0, digits = 0, big.mark = ",")

## 3   
## "7"

format(100\*race\_tab[5], nsmall = 0, digits = 0, big.mark = ",")

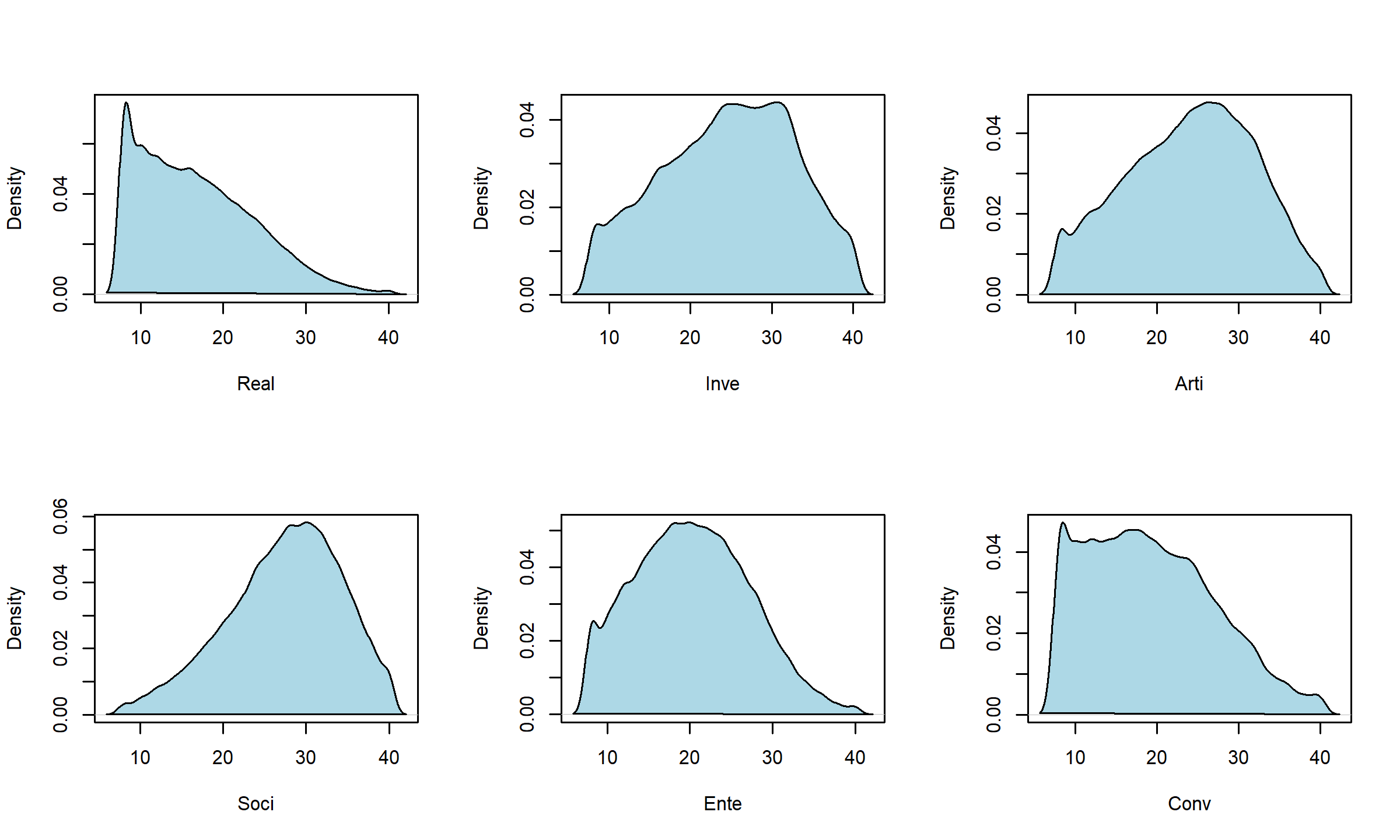
## 4   
## "61"

format(100\*race\_tab[1], nsmall = 0, digits = 0, big.mark = ",")

## 0   
## "1"

**Figure 6**

par(mfrow = c(2, 3))  
for (i in c("Real", "Inve", "Arti", "Soci" , "Ente", "Conv")) {  
 dens <- density(data[ , i])  
 plot(dens, main = "", xlab = i)  
 polygon(dens, col="lightblue", border="black")   
}



**Table 2**

tab <- round(cor(data[ , c("Real", "Inve", "Arti", "Soci" , "Ente", "Conv")]),   
 digits = 3L)  
kable(tab)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Real | Inve | Arti | Soci | Ente | Conv |
| Real | 1.000 | 0.332 | 0.182 | 0.049 | 0.305 | 0.460 |
| Inve | 0.332 | 1.000 | 0.329 | 0.141 | 0.016 | 0.065 |
| Arti | 0.182 | 0.329 | 1.000 | 0.290 | 0.253 | -0.056 |
| Soci | 0.049 | 0.141 | 0.290 | 1.000 | 0.356 | 0.124 |
| Ente | 0.305 | 0.016 | 0.253 | 0.356 | 1.000 | 0.464 |
| Conv | 0.460 | 0.065 | -0.056 | 0.124 | 0.464 | 1.000 |