Is there an influence of sampling date?

Jody Daniel

2020-11-03

Background

Some of the trait data may be affected by the date in which they were sampled. In building the final glm, we want to ensure that there is not an un accounted for influence of sampling date on traits. To deal with this confounding factor, we will run a linear regression and use the residuals, versus the raw data, to build the final glm. To do this, I will use the lapply function to run a linear regression on each trait, then pull out the p-value for each model. Models where there are a significant influence of sampling date, I will export the residuals.

```
library(knitr)
library(kableExtra)
library(tidyverse)
library(dplyr)
library(here)
library(skimr)
library(extrafont)
source(here("scripts/archive/1. functions.R"))
theme_set(theme_special())
```

Is there an influence of sampling date?

The first thing we can do is import the data and append sampling date. The sampling date is missing from the data exported from the PCA.

```
rgr_msh_raw_rf <- read_csv(here("data/RGR_MSH_PCA_Raw-RF.csv"),
                           guess_max = 10000,
                           col_types = cols())
rgr_msh_raw_orginal <- read_csv(here("data/RGR_MSH_NA.csv"),
                           guess_max = 10000,
                           col_types = cols())

rgr_msh_julian <-
  rgr_msh_raw_rf[,-(which(names(rgr_msh_raw_rf)=="BIO_GR"))] %>%
  left_join(rgr_msh_raw_orginal[, c("SampleID", "julian.date.2011")], by = "SampleID" )

kable(skim(rgr_msh_julian),"latex", booktabs = T) %>%
  kable_styling()
```

skim_type	skim_variable	n_missing	complete_rate	character.min	character.max	character.empty
character	SampleID	0	1.0000000	6	10	0
numeric	BAI_GR	0	1.0000000	NA	NA	NA
numeric	Height.DBH.Ratio	0	1.0000000	NA	NA	NA
numeric	Estem	0	1.0000000	NA	NA	NA
numeric	Branching.Distance	0	1.0000000	NA	NA	NA
numeric	Twig.Diameter	0	1.0000000	NA	NA	NA
numeric	Twig.Wood.Density	0	1.0000000	NA	NA	NA
numeric	Stem.Wood.Density	0	1.0000000	NA	NA	NA
numeric	Leaf.Mass.Fraction	0	1.0000000	NA	NA	NA
numeric	Leaf.Area	0	1.0000000	NA	NA	NA
numeric	LMA	0	1.0000000	NA	NA	NA
numeric	LCC	0	1.0000000	NA	NA	NA
numeric	LNC	0	1.0000000	NA	NA	NA
numeric	LPC	0	1.0000000	NA	NA	NA
numeric	d15N	0	1.0000000	NA	NA	NA
numeric	t.b2	1	0.9961686	NA	NA	NA
numeric	Ks	1	0.9961686	NA	NA	NA
numeric	Ktwig	1	0.9961686	NA	NA	NA
numeric	Huber.Value	1	0.9961686	NA	NA	NA
numeric	X.Lum	1	0.9961686	NA	NA	NA
numeric	VD	1	0.9961686	NA	NA	NA
numeric	X.Sapwood	1	0.9961686	NA	NA	NA
numeric	d13C	0	1.0000000	NA	NA	NA
numeric	Root.Wood.Density	94	0.6398467	NA	NA	NA
numeric	Twig.branching.angle	63	0.7586207	NA	NA	NA
numeric	Hmax	0	1.0000000	NA	NA	NA
numeric	Shade.Tolerance	22	0.9157088	NA	NA	NA
numeric	Drought.Tolerance	22	0.9157088	NA	NA	NA
numeric	WaterLogging.Tolerance	22	0.9157088	NA	NA	NA
numeric	Soil.Fertility	0	1.0000000	NA	NA	NA
numeric	Light	0	1.0000000	NA	NA	NA
numeric	Temperature	0	1.0000000	NA	NA	NA
numeric	pH	0	1.0000000	NA	NA	NA
numeric	Slope	0	1.0000000	NA	NA	NA
numeric	julian.date.2011	0	1.0000000	NA	NA	NA

```
# skim(rgr_msh_julian) - for markdown visualization
```

Now, we can run the linear regression that is really to remove the influence of sampling date, when there is one.

```
# not all columns are numeric/should be included in this analyses
col_traits <- 3:(ncol(rgr_msh_julian)-1)
# need to determine which of the columns are influenced by sampling dates
p_value_julian <- unname(unlist(lapply(col_traits, function(x){
  julian.date <-  unname(unlist(rgr_msh_julian[ , "julian.date.2011"]))
  y.julian.date <- unname(unlist(rgr_msh_julian[ , x]))
  p_value_julian <- summary(lm(y.julian.date ~ julian.date))$coefficients[,4][2]
})))
```

```r
# add the column names to make this listing make more sense
names(p_value_julian) <- colnames(rgr_msh_julian[,col_traits])
# now, let's see which columns I need to extract residuals for
p_value_julian_names <- names(which(p_value_julian<0.05))
p_value_julian_which <- which(colnames(rgr_msh_julian)%in%p_value_julian_names) # need the column numbe
p_value_julian_names # these are the columns where traits are significantly affected by sampling dates
```

```
##  [1] "Height.DBH.Ratio"     "Estem"                "Branching.Distance"
##  [4] "Twig.Diameter"        "Stem.Wood.Density"    "LMA"
##  [7] "LNC"                  "d15N"                 "Ks"
## [10] "X.Sapwood"            "d13C"                 "Twig.branching.angle"
## [13] "Shade.Tolerance"      "Drought.Tolerance"    "Soil.Fertility"
## [16] "Light"                "Temperature"          "pH"
```

Let's extract the residuals for each of these columns and replace these values in the dataset

```r
# need to determine which of the columns are influenced by sampling dates
residuals_julian <- lapply(p_value_julian_which, function(x){
  julian.date <-  unname(unlist(rgr_msh_julian[ , "julian.date.2011"]))
  y.julian.date <- unname(unlist(rgr_msh_julian[ , x]))
  residuals_julian <- lm(y.julian.date ~ julian.date)$residuals
})

rgr_msh_residuals_matrix <- matrix(NA, nrow = nrow(rgr_msh_julian), ncol = length(p_value_julian_which))

# convert to a dataframe and give correct column names
for (i in 1:length(p_value_julian_which)){
    row_member <- c(which(!is.na(rgr_msh_julian[,p_value_julian_which[i]])),
                    which(is.na(rgr_msh_julian[,p_value_julian_which[i]])))
    row_NAs <- rep(NA,length(which(is.na(rgr_msh_julian[,p_value_julian_which[i]]))))
    rgr_msh_residuals_matrix[row_member,i] <- c(residuals_julian[[i]], row_NAs)
  }
colnames(rgr_msh_residuals_matrix) <- p_value_julian_names

# now, let's make a new data frame that has these new columns
# must ensure to drop the columns from the raw data where there was a significant effect

rgr_msh_residuals_julian_df <- tibble(rgr_msh_julian[, -p_value_julian_which],
                                      data.frame(rgr_msh_residuals_matrix))

kable(skim(rgr_msh_residuals_julian_df),"latex", booktabs = T) %>%
  kable_styling()
```

skim_type	skim_variable	n_missing	complete_rate	character.min	character.max	character.empty
character	SampleID	0	1.0000000	6	10	0
numeric	BAI_GR	0	1.0000000	NA	NA	NA
numeric	Twig.Wood.Density	0	1.0000000	NA	NA	NA
numeric	Leaf.Mass.Fraction	0	1.0000000	NA	NA	NA
numeric	Leaf.Area	0	1.0000000	NA	NA	NA
numeric	LCC	0	1.0000000	NA	NA	NA
numeric	LPC	0	1.0000000	NA	NA	NA
numeric	t.b2	1	0.9961686	NA	NA	NA
numeric	Ktwig	1	0.9961686	NA	NA	NA
numeric	Huber.Value	1	0.9961686	NA	NA	NA
numeric	X.Lum	1	0.9961686	NA	NA	NA
numeric	VD	1	0.9961686	NA	NA	NA
numeric	Root.Wood.Density	94	0.6398467	NA	NA	NA
numeric	Hmax	0	1.0000000	NA	NA	NA
numeric	WaterLogging.Tolerance	22	0.9157088	NA	NA	NA
numeric	Slope	0	1.0000000	NA	NA	NA
numeric	julian.date.2011	0	1.0000000	NA	NA	NA
numeric	Height.DBH.Ratio	0	1.0000000	NA	NA	NA
numeric	Estem	0	1.0000000	NA	NA	NA
numeric	Branching.Distance	0	1.0000000	NA	NA	NA
numeric	Twig.Diameter	0	1.0000000	NA	NA	NA
numeric	Stem.Wood.Density	0	1.0000000	NA	NA	NA
numeric	LMA	0	1.0000000	NA	NA	NA
numeric	LNC	0	1.0000000	NA	NA	NA
numeric	d15N	0	1.0000000	NA	NA	NA
numeric	Ks	1	0.9961686	NA	NA	NA
numeric	X.Sapwood	1	0.9961686	NA	NA	NA
numeric	d13C	0	1.0000000	NA	NA	NA
numeric	Twig.branching.angle	63	0.7586207	NA	NA	NA
numeric	Shade.Tolerance	22	0.9157088	NA	NA	NA
numeric	Drought.Tolerance	22	0.9157088	NA	NA	NA
numeric	Soil.Fertility	0	1.0000000	NA	NA	NA
numeric	Light	0	1.0000000	NA	NA	NA
numeric	Temperature	0	1.0000000	NA	NA	NA
numeric	pH	0	1.0000000	NA	NA	NA

```
# skim(rgr_msh_residuals_julian_df) - for markdown visualization

# now, we can export these data
write_csv(rgr_msh_residuals_julian_df, here("data/rgr_msh_residuals_julian_df.csv"))
```

We are ready! Now, we can build the linear regression model.