Anaylises for SONATA review

Purpose field

Alicia Valdés

25 July 2025

This is a script to perform analyses for the SONATA review paper.

Load libraries

```
library(here)
```

```
## here() starts at C:/Users/alici/OneDrive - Universidad de Oviedo/IMIB/Analyses/SONATA_review
```

```
library(readxl)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

Read data

In this data, in the sheet "env" I have edited the environmental variables manually in order to ALWAYS have ";" separating the different variables in each column.

```
sonata_data <-read_excel(here("data", "edited","database_sonata_v2.2_AV.xlsx"),
                         sheet = "env")
sonata_data
```

```
## # A tibble: 1,569 x 5
##       ID Climate                                       Soil    Topography Biotic
##    <dbl> <chr>                                         <chr>   <chr>      <chr>
##   1     2 <NA>                                          <NA>    aspect;sl~ nDSM,~
##   2     5 <NA>                                          <NA>    DEM; elev~ <NA>
##   3     6 <NA>                                          <NA>    <NA>       <NA>
##   4     9 <NA>                                          <NA>    <NA>       Veget~
##   5    14 annual mean temperature; annual precipitation <NA>    elevation~ veget~
##   6    18 <NA>                                          <NA>    <NA>       <NA>
##   7    19 <NA>                                          <NA>    <NA>       <NA>
##   8    20 <NA>                                          <NA>    1          1
##   9    21 <NA>                                          soils ~ aspect     concu~
## 10    23 1                                             1       1          1
## # i 1,559 more rows
```

Check for duplicate IDs

```
sonata_data %>% count(ID) %>% filter(n > 1)
```

```
## # A tibble: 1 x 2
##      ID     n
##   <dbl> <int>
## 1  2495     2
```

```
sonata_data %>% mutate(row_number = row_number()) %>% filter(ID == 2495)
```

```
## # A tibble: 2 x 6
##      ID Climate                    Soil             Topography Biotic row_number
##   <dbl> <chr>                      <chr>            <chr>      <chr>       <int>
## 1  2495 Temperature; Precipitation Soil moisture; ~ None       LAI, ~        618
## 2  2495 <NA>                       <NA>             <NA>       Veget~        619
```

```
# Remove row number of the wrong entry
sonata_data <- sonata_data %>% slice(-619)
```

Clean environmental variables

Convert all to 0/1

```
sonata_data <- sonata_data %>%
  mutate(Climate_bin = if_else(is.na(Climate) | Climate == "", 0,
                               if_else(Climate == "None", 0, 1)),
         Soil_bin = if_else(is.na(Soil) | Soil == "", 0,
                            if_else(Soil == "None" | Soil == "Soil: none", 0,
                                    1)),
         Topography_bin = if_else(is.na(Topography) | Topography == "", 0,
                                  if_else(Topography == "None" |
                                            Topography == "None specified" |
                                            Topography == "Not explicitly", 0,
                                          1)),
         Biotic_bin = if_else(is.na(Biotic) | Biotic == "", 0,
                              if_else(Biotic == "None", 0, 1)))
```

Climate

```
sonata_climate <- sonata_data %>%
  select(ID, starts_with("Climate")) %>%
  separate(Climate,
           # There is at most 6 different climatic variables
           # Separate into 6 cols
           into = paste0("Climate", 1:6),
           # The different variables in each col are always separated by ";"
           sep = ";",
           fill = "right",
           remove = FALSE, extra = "warn") %>%
  # Remove whitespace
  mutate(across(starts_with("Climate"), str_trim)) %>%
  rowwise() %>%
  # Create new cols to store data on most important Climate variables
  mutate(
    # Any of the Climate cols contains temperature
    Clim_temp = any(str_detect(c_across(Climate1:Climate6),
                               regex("temperature", ignore_case = TRUE))),
    # Any of the Climate cols contains precipitation, rainfall or drought
    Clim_precip = any(str_detect(c_across(Climate1:Climate6),
                                 regex("precipitation|rainfall|drought",
                                       ignore_case = TRUE)))  ,
    # Any of the Climate cols contains any word containing "radi"
    # (like "radiation", "irradiation") and the exact word "PAR"
    Clim_rad = any(str_detect(c_across(Climate1:Climate6),
                              regex("radi|\\bPAR\\b", ignore_case = TRUE))),
    # Any of the Climate cols contains humidity
    Clim_humid = any(str_detect(c_across(Climate1:Climate6),
                                regex("humidity", ignore_case = TRUE))),
    # Any of the Climate cols contains wind
    Clim_wind = any(str_detect(c_across(Climate1:Climate6),
                               regex("wind", ignore_case = TRUE))),
    # Any of the Climate cols contains evapotranspiration
    Clim_evap = any(str_detect(c_across(Climate1:Climate6),
                               regex("evapotranspiration",
                                     ignore_case = TRUE))),
    # There is sth in any of the Climate cols and
    # all previous variables are FALSE
    Clim_other = if_else(Climate1 != 1 & Climate_bin == 1 &
                           all(is.na(c_across(Clim_temp:Clim_evap))), TRUE, NA),
    # Climatic variables not specified
    Clim_unspecif = if_else(Climate1 == 1, TRUE, NA)
    ) %>%
  ungroup()
```

Soil

```r
sonata_soil <- sonata_data %>%
  select(ID, starts_with("Soil")) %>%
  separate(Soil,
           # There is at most 8 different soil variables
           # Separate into 8 cols
           into = paste0("Soil", 1:8),
           # The different variables in each col are always separated by ";"
           sep = ";",
           fill = "right",
           remove = FALSE, extra = "warn") %>%
  # Remove whitespace
  mutate(across(starts_with("Soil"), str_trim)) %>%
  rowwise() %>%
  # Create new cols to store data on most important Soil variables
  mutate(
    # Any of the soil cols contains any word containing "type" or "class"
    Soil_type = any(str_detect(c_across(Soil1:Soil8),
                               regex("type|class", ignore_case = TRUE))),
    # Any of the Soil cols contains texture
    Soil_text = any(str_detect(c_across(Soil1:Soil8),
                               regex("texture", ignore_case = TRUE))),
    # Any of the soil cols contains moisture, water or wetness
    Soil_moist = any(str_detect(c_across(Soil1:Soil8),
                                regex("moisture|water|wetness",
                                      ignore_case = TRUE))),
    # Any of the soil cols contains depth
    Soil_depth = any(str_detect(c_across(Soil1:Soil8),
                                regex("depth", ignore_case = TRUE))),
    # Any of the soil cols contains pH or acidity
    Soil_ph = any(str_detect(c_across(Soil1:Soil8),
                             regex("pH|acidity", ignore_case = TRUE))),
    # Any of the soil cols contains carbon
    Soil_carbon = any(str_detect(c_across(Soil1:Soil8),
                                 regex("carbon", ignore_case = TRUE))),
    # Any of the soil cols contains roughness
    Soil_rough = any(str_detect(c_across(Soil1:Soil8),
                                regex("roughness", ignore_case = TRUE))),
    # Any of the soil cols contains any word containing "ferti"
    Soil_ferti = any(str_detect(c_across(Soil1:Soil8),
                               regex("ferti", ignore_case = TRUE))),
    # Any of the soil cols contains nitrogen
    Soil_nitro = any(str_detect(c_across(Soil1:Soil8),
                                regex("nitrogen", ignore_case = TRUE))),
    # Any of the soil cols contains bulk
    Soil_bulk = any(str_detect(c_across(Soil1:Soil8),
                               regex("bulk", ignore_case = TRUE))),
    # There is sth in any of the Soil cols and
    # all previous variables are FALSE
    Soil_other = if_else(Soil1 != 1 & Soil_bin == 1 &
                           all(is.na(c_across(Soil_type:Soil_bulk))), TRUE, NA),
    # Soil variables not specified
    Soil_unspecif = if_else(Soil == 1, TRUE, NA)
    ) %>%
  ungroup()
```

topography

```
sonata_topo <- sonata_data %>%
  select(ID, starts_with("Topo")) %>%
  separate(Topography,
           # There is at most 7different topo variables
           # Separate into 7 cols
           into = paste0("Topo", 1:7),
           # The different variables in each col are always separated by ";"
           sep = ";",
           fill = "right",
           remove = FALSE, extra = "warn") %>%
  # Remove whitespace
  mutate(across(starts_with("Topo"), str_trim)) %>%
  rowwise() %>%
  # Create new cols to store data on most important Topo variables
  mutate(
    # Any of the Topo cols contains aspect
    Topo_aspect = any(str_detect(c_across(Topo1:Topo7),
                               regex("aspect", ignore_case = TRUE))),
    # Any of the Topo cols contains slope
    Topo_slope = any(str_detect(c_across(Topo1:Topo7),
                               regex("slope", ignore_case = TRUE))),
    # Any of the Topo cols contains elevation or altitude
    Topo_elev = any(str_detect(c_across(Topo1:Topo7),
                               regex("elevation|altitude",
                                     ignore_case = TRUE))),
    # Any of the Topo cols contains TWI or wetness
    Topo_twi = any(str_detect(c_across(Topo1:Topo7),
                               regex("TWI|wetness", ignore_case = TRUE))),
    # Any of the Topo cols contains curvature
    Topo_curv = any(str_detect(c_across(Topo1:Topo7),
                               regex("curvature", ignore_case = TRUE))),
    # There is sth in any of the Topo cols and
    # all previous variables are FALSE
    Topo_other = if_else(Topo1 != 1 & Topography_bin == 1 &
                           all(is.na(c_across(Topo_aspect:Topo_curv))), TRUE,
                         NA),
    # Topo variables not specified
    Topo_unspecif = if_else(Topography == 1, TRUE, NA)
    ) %>%
  ungroup()
```

Merged data

```
sonata_data %>%
  # Add Climate
  left_join(sonata_climate %>% select(ID, Clim_temp:Clim_unspecif)) %>%
  # Add Soil
  left_join(sonata_soil %>% select(ID, Soil_type:Soil_unspecif)) %>%
  # Add Topo
  left_join(sonata_topo %>% select(ID, Topo_aspect:Topo_unspecif))
```

```
## Joining with ‘by = join_by(ID)‘
## Joining with ‘by = join_by(ID)‘
## Joining with ‘by = join_by(ID)‘
```

```
## # A tibble: 1,568 x 36
##       ID Climate     Soil  Topography Biotic Climate_bin Soil_bin Topography_bin
##     <dbl> <chr>       <chr> <chr>      <chr>        <dbl>    <dbl>          <dbl>
##  1      2 <NA>        <NA>  aspect;sl~ nDSM,~           0        0              1
##  2      5 <NA>        <NA>  DEM; elev~ <NA>             0        0              1
##  3      6 <NA>        <NA>  <NA>       <NA>             0        0              0
##  4      9 <NA>        <NA>  <NA>       Veget~           0        0              0
##  5     14 annual mea~ <NA>  elevation~ veget~           1        0              1
##  6     18 <NA>        <NA>  <NA>       <NA>             0        0              0
##  7     19 <NA>        <NA>  <NA>       <NA>             0        0              0
##  8     20 <NA>        <NA>  1          1                0        0              1
##  9     21 <NA>        soil~ aspect     concu~           0        1              1
## 10     23 1           1     1          1                1        1              1
## # i 1,558 more rows
## # i 28 more variables: Biotic_bin <dbl>, Clim_temp <lgl>, Clim_precip <lgl>,
## #   Clim_rad <lgl>, Clim_humid <lgl>, Clim_wind <lgl>, Clim_evap <lgl>,
## #   Clim_other <lgl>, Clim_unspecif <lgl>, Soil_type <lgl>, Soil_text <lgl>,
## #   Soil_moist <lgl>, Soil_depth <lgl>, Soil_ph <lgl>, Soil_carbon <lgl>,
## #   Soil_rough <lgl>, Soil_ferti <lgl>, Soil_nitro <lgl>, Soil_bulk <lgl>,
## #   Soil_other <lgl>, Soil_unspecif <lgl>, Topo_aspect <lgl>, ...
```

Session info

```
sessionInfo()
```

```
## R version 4.4.2 (2024-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 11 x64 (build 26100)
##
## Matrix products: default
##
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## time zone: Europe/Madrid
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] lubridate_1.9.4 forcats_1.0.0   stringr_1.5.1   dplyr_1.1.4
##  [5] purrr_1.0.4     readr_2.1.5     tidyr_1.3.1     tibble_3.2.1
##  [9] ggplot2_3.5.1   tidyverse_2.0.0 readxl_1.4.3    here_1.0.1
##
## loaded via a namespace (and not attached):
##  [1] gtable_0.3.6      compiler_4.4.2    tidyselect_1.2.1  scales_1.3.0
##  [5] yaml_2.3.10       fastmap_1.2.0     R6_2.6.1          generics_0.1.3
##  [9] knitr_1.49        munsell_0.5.1     rprojroot_2.0.4   pillar_1.10.1
## [13] tzdb_0.4.0        rlang_1.1.5       utf8_1.2.4        stringi_1.8.4
## [17] xfun_0.50         timechange_0.3.0  cli_3.6.3         withr_3.0.2
## [21] magrittr_2.0.3    digest_0.6.37     grid_4.4.2        rstudioapi_0.17.1
## [25] hms_1.1.3         lifecycle_1.0.4   vctrs_0.6.5       evaluate_1.0.3
## [29] glue_1.8.0        cellranger_1.1.0  colorspace_2.1-1  rmarkdown_2.29
## [33] tools_4.4.2       pkgconfig_2.0.3   htmltools_0.5.8.1
```