

Survey123 plot data

Norwegian institute for nature research

09/18/2024

This script converts data imported from Survey123 into the format used in the masterfile: Herminium_monorchis_masterfile_v_2024.

Before getting started, we can run this line to make sure R can read Norwegian letters.

```
Sys.setlocale(locale='no_NB.utf8')
```

```
## [1] "LC_COLLATE=no_NB.utf8;LC_CTYPE=no_NB.utf8;LC_MONETARY=no_NB.utf8;LC_NUMERIC=C;LC_TIME=no_NB.utf8"
```

First, we need to load the required packages.

```
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  1.9.3      v tidyr      1.3.1
```

```
## v purrr      1.0.2
```

```
## -- 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
```

```
library(readxl)
```

```
library(vegan)
```

```
## Loading required package: permute
```

```
## Loading required package: lattice
```

```
## This is vegan 2.6-8
```

```
library(car)
```

```
## Loading required package: carData
```

```
##
```

```
## Attaching package: 'car'
```

```
##
```

```
## The following object is masked from 'package:dplyr':
```

```
##
```

```
##      recode
```

```
##
```

```
## The following object is masked from 'package:purrr':
```

```
##
```

```
##      some
```

```
library(goveg)
```

```
## This is GoeVeg 0.7.5 - build: 2024-05-17
```

```
library(ggrepel)
```

```
#NB use install.packages("package name") if the package is not installed yet.  
#Using qmd requires the "rmarkdown" package installed.
```

We can import the raw data from Survey123.

```
env <- read_excel("P:/154027_effektovervaking_av_trua_arter_og_naturtyper_2024/04 Honningblom/Herminium
```

```
## New names:
```

```
## * `Kommentarer:` -> `Kommentarer:...10`
```

```
## * `Kommentarer:` -> `Kommentarer:...11`
```

```
## * `Kommentarer:` -> `Kommentarer:...18`
```

```
## * `Kommentarer:` -> `Kommentarer:...29`
```

```
species <- read_excel("P:/154027_effektovervaking_av_trua_arter_og_naturtyper_2024/04 Honningblom/Hermi
```

The job at hand is to combine these two datasets, and create one dataset that we can analyze in R.

The column named ParentGlobalID in the species data matches with GlobalID in the env data. GlobalID in the species data is a unique identifier in that dataset, and we do not need it at this point. We can thus use GlovalID (env) and ParentGlobalID (species) as identifiers to merge the data. We can call it ParentGlobalID in both datasets for ease of joining. Multiple rows of data per GlobalParent may be created (e.g., multiple species per GlobalParentID).

```
#First we need to fix the plot names
```

```
env <- env %>%
```

```
tidyr::separate("Rute-ID", into = c("Locality_ID", "Plot_number"), "(?<=[A-Z])(?=[0-9])", fill = "left")
```

```
mutate(Locality_ID = if_else(`Polygon ID` == "Fillettassen", "FI",
```

```
                        if_else(`Polygon ID` == "Skipstadsand", "SKI",
```

```
                        if_else(`Polygon ID` == "Skjellvik", "SKJ",
```

```
                        if_else(`Polygon ID` == "Teneskjær", "TEN", NA)))) %>%
```

```
unite(Plot_ID_new, Locality_ID:Plot_number, sep = "", remove = FALSE) #creates new Plot_ID like we want
```

```
#Now we can rename the GlobalID so we can join the datafiles
```

```
env <- env %>%
```

```
  rename(ParentGlobalID = GlobalID)
```

```
#Join the species dataset to the environmental dataset
```

```
df_plot_plants <- env %>%
```

```
  left_join(species, by = "ParentGlobalID")
```

Now we have a datafile with a lot of columns we do not need. To match the master datafile for Herminium monorchis, we only need:

- Date (date of registration)
- Locality (full locality name)
- Locality_ID (abbreviation for the locality)
- Plot_ID (plot number)
- Plot_ID_new (consists of the locality ID and plot number)
- Count Herminium monorchis, vegetative (we do not strictly need this in this dataset, but it is needed to filter out obligatory H. orchis registrations in a quadrat for plots that did not have H. monorchis. At these plots, no full species composition was surveyed, so we want to remove them or substitute with NAs)

- Count *Herminium monorchis*, fertile
- Species_norwegian (norwegian species name)
- Presence_quadrat (the quadrats the species is registered in, formatted as #,#,#,etc)
- Quadrats (the number of quadrats the species is present in - this one does not exist in the S123 data, so needs to be created later)
- Cover_vascular.plants
- Cover_DOM
- Cover_bare.soil (we do not have this on in S123)
- Cover_bryophytes
- Cover_woody_fieldlayer
- Cover_shrubs
- Height_1
- Height_2
- Height_3
- Height_4
- Comments (any comments to species registrations)

#NB use `variabe name` when the variable name (column) contains spaces or other signs that are incompat

```
df_plot_plants <- df_plot_plants %>%
  select(`Registreringsdato`,
        `Polygon ID.x`,
        Locality_ID,
        Plot_number,
        Plot_ID_new,
        `Antall honningblom:vegetative planter`,
        `Antall honningblom:fertile planter`,
        sp_species,
        `Smårutene nr.`,
        `Kommentarer:...10`,
        `Kommentarer:...11`,
        `Kommentarer:...18`,
        `Kommentarer:...29`,
        `Dekning % av karplanter i feltsjikt`,
        `Dekning % av strø`,
        `Dekning % av bunnsjikt`,
        `Dekning % av vedplanter i feltsjikt`,
        `Dekning % av vedplanter i busksjikt`,
        `Vegetasjonshøyde måling 1`,
        `Vegetasjonshøyde måling 2`,
        `Vegetasjonshøyde måling 3`,
        `Vegetasjonshøyde måling 4`)
```

#Now we can rename the non-compatible column names

```
df_plot_plants <- df_plot_plants %>%
  dplyr::rename("Date" = `Registreringsdato`, #rename uses format new name = old name
               "Locality" = `Polygon ID.x`,
```

```

"Locality_ID" = Locality_ID,
"Plot_ID" = Plot_number,
"Plot_ID_new" = Plot_ID_new,
"Herminium_count_vegetative" = `Antall honningblom:vegetative planter`,
"Herminium_count_fertile" = `Antall honningblom:fertile planter`,
"Species_both" = sp_species, #to be splitted in Norwegian and scientific names later
"Presence_quadrat" = `Smårutene nr.`,
"Comment_1" = `Kommentarer:...10`,
"Comment_2" = `Kommentarer:...11`,
"Comment_3" = `Kommentarer:...18`,
"Comment_4" = `Kommentarer:...29`,
"Fieldlayer_cover" = `Dekning % av karplanter i feltsjikt`,
"Litter_cover" = `Dekning % av strø`,
"Groundlayer_cover" = `Dekning % av bunnsjikt`,
"Woody_fieldlayer_cover" = `Dekning % av vedplanter i feltsjikt`,
"Shrub_cover" = `Dekning % av vedplanter i busksjikt`,
"Vegetation_height_1" = `Vegetasjonshøyde måling 1`,
"Vegetation_height_2" = `Vegetasjonshøyde måling 2`,
"Vegetation_height_3" = `Vegetasjonshøyde måling 3`,
"Vegetation_height_4" = `Vegetasjonshøyde måling 4`)

```

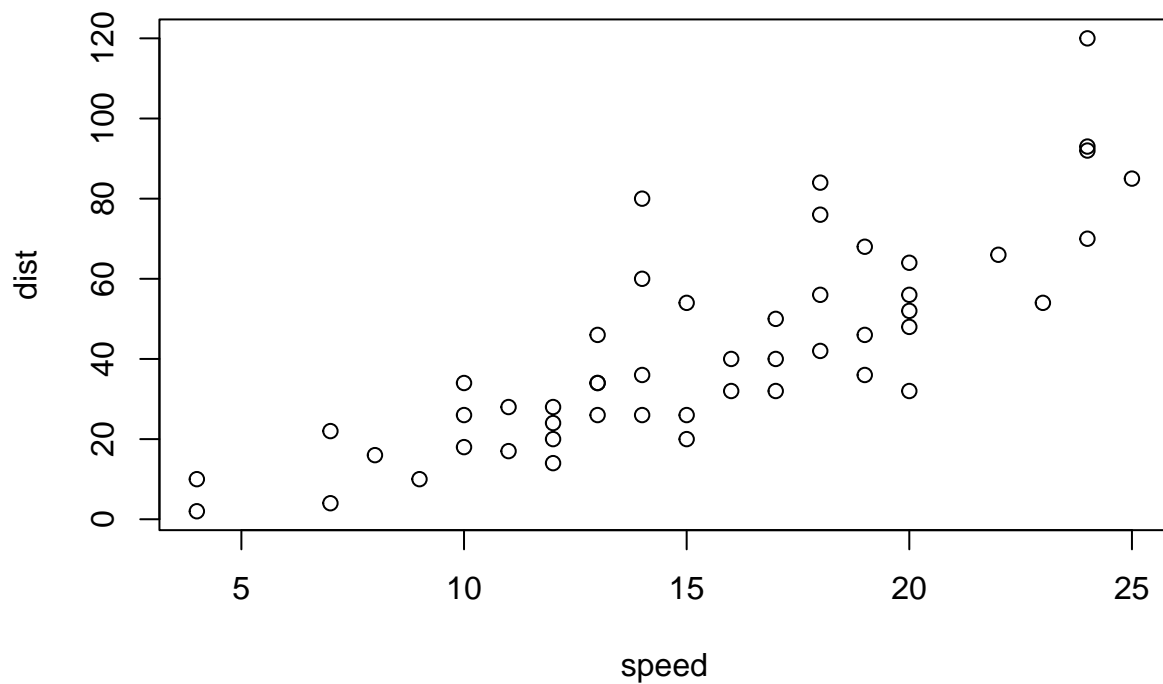
#Split up the species names in latin and norwegian

Ruteregistreringer for Honningblom må fjernes når det er en observasjon i rute 1 men tellingen i rute 1 viser 0 individer

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

```
plot(cars)
```



Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.