InfiltrodiscR: an R package for infiltrometer data analysis and an experience for improving data reproducibility in a soil physics laboratory

Carolina V. Giraldo[1,2], Sara E. Acevedo[1]

[1]Pontificia Universidad Católica de Chile
[2]Centro de Desarrollo Urbano Sustentable (CEDEUS)

Introduction

Many soil physics researchers are unacquainted about the functionalities of the programing language R (Sousa et al. 2020). One main functionality that differentiates it from spreadsheet-based programs is that R scripts are text-based, making them easily shareable and reproducible, allowing to replicate analyses. The challenge of achieving reproducibility persists across various scientific disciplines, extending it to soil science as well (Correndo et al. 2023). Also, researchers in soil science, and almost every other field, are being pushed by funding agencies and governmental institutions to increase transparency and reproducibility of their work (Bond-Lamberty, Smith, and Bailey 2016). Open, accessible, reusable, and reproducible soil hydrologic research can have a significant positive impact on the scientific community and broader society (Hall et al. 2022)

Motivation

In 2023, the authors of this work joined to a program lead by Open Life Science; a community-oriented non-profit organisation that promotes open, inclusive and equitable research (Haynes 2023). In addition, the members of the laboratory already had knowledge of R but there was no repository with common functions for the infiltrometer data analysis. Based on the programming expertise of the laboratory members and the need of adoption of open and reproducible science, the R package InfiltrodiscR was developed (Acevedo and Giraldo 2023). The goal of infiltrodiscR is to provide functions for the modeling of data derived from the Minidisk Infiltrometer device.

R package description

The R package is currently hosted in GitHub. This web-based Git repository hosting service is currently used by many scientists to work in teams or collaborative projects (Bond-Lamberty, Smith, and Bailey 2016). Also, the code was deposited in Zenodo, a free service for hosting data and software that offers long-term (20-year) storage and integration with GitHub (Hall et al. 2022). The infiltrodiscR package has a DOI so it can be used as reference in publications and clearly define the software version used.

To install the R package, the users need to run to following lines:

```
# install.packages("devtools")
devtools::install_github("biofisicasuelos/infiltrodiscR")
```

Data needed for running the functions are data stored in **.csv** or **.xlsx** containing columns called as follows:

- texture: soil texture according to USDA: as.character() and lowercase, for example "clay loam".
- suction: as.character() and lowercase, in this format: "2cm". Values allowed: "0.5cm","1cm","2cm","3cm","4cm","5cm","6cm", and "7cm".
- volume: volume recorded in the infiltration measurements in mL, as.numeric().
- time: time recorded in the infiltration measurements in seconds, as.numeric().

Main functions:

`infiltration()`

This function calculates cumulative infiltration and the square root of time, using time and volume recorded based on the relationship described by Philip (1957):

$$I = C_1 t + C_2 t^{0.5}$$

`vg_par`

This function returns the parameter A, *no_h* and *alpha* related to the van Genuchten parameters (Genuchten 1980), from tabulated data calculated for a radius of 2.25 cm, including 12 soil texture classes and suctions from -0.5 cm to -7 cm. Table 1 show selected data gathered from MeterGroup (2023) and Carsel and Parrish (1988).

`parameter_A`

This function returns the parameter A calculated from the equation based on the work developed by Zhang (1997), where the parameters A, *no_h* and *alpha*

Table 1: Table 1. Selected data from the InfiltrodiscR package

texture	alpha	n/ho	2cm	4cm	6cm
sand	0.145	2.68	1.727908	0.8926213	0.4611201
loamy sand	0.124	2.28	2.428600	1.8443631	1.4006735
sandy loam	0.075	1.89	3.909913	3.9541483	3.9988836
loam	0.036	1.56	6.267384	7.5304822	9.0481388
silt	0.016	1.37	8.714378	9.8964325	11.2388254
silt loam	0.020	1.41	7.929874	9.1856010	10.6401773
sandy clay loam	0.059	1.48	4.242925	6.1530807	8.9231845
clay loam	0.019	1.31	6.644845	7.8616094	9.3011807
silty clay loam	0.010	1.23	8.511175	9.4110072	10.4059732
sandy clay	0.027	1.23	4.089288	5.3640585	7.0362184
silty clay	0.005	1.09	6.359575	6.7578953	7.1811641
clay	0.008	1.09	4.300401	4.7393888	5.2231893

determined previously are input in the following equations described in Meter-Group (2023) and Surda et al. (2019)

$$A = \frac{11.65(n^{0.1}-1)exp[2.92(n-1.9)\alpha h_0]}{(\alpha r_0)^{0.91}}; n \geq 1.9$$

$$A = \frac{11.65(n^{0.1}-1)exp[7.5(n-1.9)\alpha h_0]}{(\alpha r_0)^{0.91}}; n < 1.9$$

Practical example

First, some dummy data about infiltration and soils is created. In this case: "soil_a" and "soil_b".

```
infiltration_data <- tibble(
  soil = c(rep("soil_a",11), rep("soil_b",11)),
  time = c(0, 30, 60, 90, 120, 150, 180, 210, 240, 270, 300, # seconds
           0, 35, 65, 95, 125, 155, 185, 215, 245, 275, 305),
  volume = c(95, 89, 86, 83, 80, 77, 74, 73, 71, 69, 67, # mL
             83, 77, 64, 61, 58, 45, 42, 35, 29, 17, 15)
)

soil_data <- tibble(soil = c("soil_a", "soil_b"),
                    texture = c("sandy loam", "clay loam"), #USDA
                    suction = c("4cm","2cm"),
                    om_content = c(1,10))

head(infiltration_data,4) # check the infiltration data
```

```
# A tibble: 4 x 3
  soil    time volume
  <chr>  <dbl>  <dbl>
1 soil_a     0     95
2 soil_a    30     89
3 soil_a    60     86
4 soil_a    90     83
```

```
soil_data # check the soil data
```

```
# A tibble: 2 x 4
  soil   texture    suction om_content
  <chr>  <chr>      <chr>        <dbl>
1 soil_a sandy loam 4cm              1
2 soil_b clay loam  2cm             10
```

Then, using the function **`infiltration()`** the cumulative infiltration and the square root of time are calculated. Notice that the package was coded tidy-oriented (tidyverse package is required). Also, it is recommended to use nested tibbles for data manipulation.

```
infilt_cum_sqrt <-
infiltration_data %>%
group_by(soil) %>% # grouped calculation by soil
nest() %>%
```

```r
  mutate(data = map(data, ~ infiltrodiscR::infiltration(.), data = .x))

infilt_cum_sqrt # nested tibble
```

```
# A tibble: 2 x 2
# Groups:   soil [2]
  soil   data
  <chr>  <list>
1 soil_a <tibble [11 x 5]>
2 soil_b <tibble [11 x 5]>
```

The nested tibble has the infiltration calculation for each soil. For details of **`infilt_cum_sqrt`**, the **`unnest()`** function can be used

```r
infilt_cum_sqrt %>%
  unnest(data)
```

```
# A tibble: 22 x 6
# Groups:   soil [2]
   soil    time volume sqrt_time volume_infiltrated infiltration
   <chr>  <dbl>  <dbl>     <dbl>              <dbl>        <dbl>
 1 soil_a     0     95      0                     0         0
 2 soil_a    30     89      5.48                  6         0.38
 3 soil_a    60     86      7.75                  9         0.57
 4 soil_a    90     83      9.49                 12         0.75
 5 soil_a   120     80     11.0                  15         0.94
 6 soil_a   150     77     12.2                  18         1.13
 7 soil_a   180     74     13.4                  21         1.32
 8 soil_a   210     73     14.5                  22         1.38
 9 soil_a   240     71     15.5                  24         1.51
10 soil_a   270     69     16.4                  26         1.63
# i 12 more rows
```

Now the soil data can be joined to the infiltration data and the Van Genuchten parameters can be obtained. It is mandatory to have a column called **`texture`** and another **`suction`**

```r
infilt_cum_sqrt %>%
  left_join(soil_data) %>%
  infiltrodiscR::vg_par()
```

```
Joining with `by = join_by(soil)`
Joining with `by = join_by(texture)`
```

```
# A tibble: 2 x 8
# Groups:   soil [2]
```

```
  soil   data                texture    suction om_content alpha  n_ho value_A
  <chr>  <list>              <chr>      <chr>        <dbl> <dbl> <dbl>   <dbl>
1 soil_a <tibble [11 x 5]> sandy loam 4cm              1 0.075  1.89    3.95
2 soil_b <tibble [11 x 5]> clay loam  2cm             10 0.019  1.31    6.64
```

The hydraulic conductivity of the soil K at a specific suction is calculated as:

$$K_{(h)} = \frac{C_1}{A}$$

Parameter C_1 is calculated fitting a polynomial function of the second degree (y = ax2+b), where a is parameter C_1, x is the square root of time and y is the cumulative infiltration calculated previously. For this step, we use the package broom and base R. The column estimate corresponds to the parameter C_1.

```
processed_data <-
infilt_cum_sqrt %>%
  left_join(soil_data) %>%
  infiltrodiscR::vg_par() %>%
    mutate(
    fit = map(data,
              ~ lm(infiltration ~ poly(sqrt_time, 2, raw = TRUE),
                   data = .x)), #polynomial function
    tidied = map(fit, broom::tidy) #coefficients
  ) %>%
  unnest(tidied) %>%
filter(term == "poly(sqrt_time, 2, raw = TRUE)2") %>% #slope
rename(C1 = estimate)
```

```
Joining with `by = join_by(soil)`
Joining with `by = join_by(texture)`
```

Finally, the hydraulic conductivity of the soil K is calculating using the parameter C_1 and A. If seconds and mL were used as inputs for infiltration data, the units of K are cm/s.

```
processed_data %>%
  infiltrodiscR::parameter_A() %>%
  mutate(K_h = C1 / parameter_A) %>%
  select(soil, texture, suction, K_h)
```

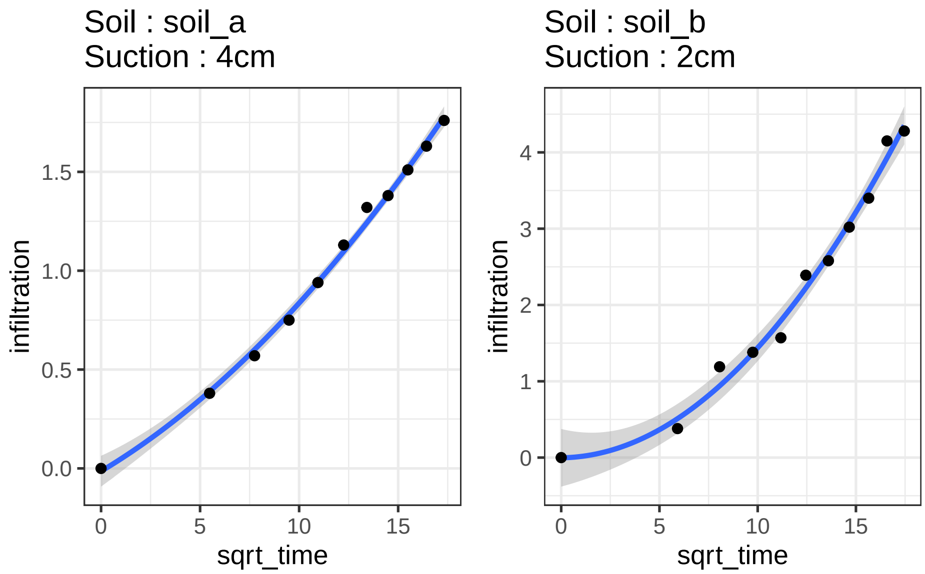
```
# A tibble: 2 x 4
# Groups:   soil [2]
  soil   texture    suction      K_h
  <chr>  <chr>      <chr>      <dbl>
1 soil_a sandy loam 4cm     0.000638
2 soil_b clay loam  2cm     0.00212
```

Using this tidy-oriented approach, it is simple to complement the functions presented with plotting.

```
infiltration_plot <-
infilt_cum_sqrt %>%
  left_join(soil_data) %>%
  mutate(plot = map2(
    data, soil,
    ~ ggplot(data = .x, aes(x = sqrt_time, y = infiltration)) +
      ggtitle(glue::glue("Soil : {soil}
                 Suction : {suction}")) +
      stat_smooth(method='lm', formula = y~poly(x,2)) +
      geom_point() +
      theme_bw()))
```

```
Joining with `by = join_by(soil)`
```

```
patchwork::wrap_plots(infiltration_plot$plot, ncol = 2)
```



Conclusions and future work

The learning curve in R programming and open science practices is not the same for every researcher, nor is it a dedicated line of research in graduate programs dedicated to soil physics. Therefore, this experience in creating an R package homogenizing the data analysis methodology in a laboratory shows that if there is interest in developing this approach, further advances in collaboration and reproducibility can be made.

Acknowledgements

The authors thank the OLS team for their time and dedication in motivating researchers to adopt open software practices. Sara Acevedo thanks the research support provided by CEDEUS (ANID/FONDAP 1522A0002) and the financial support from Postdoctorado Ingeniería UC 2023.

References

Acevedo, Sara, and Carolina Giraldo. 2023. "infiltrodiscR: R Package." Zenodo. https://doi.org/10.5281/ZENODO.8001894.

Bond-Lamberty, Ben, A Peyton Smith, and Vanessa Bailey. 2016. "Running an Open Experiment: Transparency and Reproducibility in Soil and Ecosystem Science." *Environmental Research Letters* 11 (8): 084004. https://doi.org/10.1088/1748-9326/11/8/084004.

Carsel, Robert F., and Rudolph S. Parrish. 1988. "Developing Joint Probability Distributions of Soil Water Retention Characteristics." *Water Resources Research* 24 (5): 755–69. https://doi.org/10.1029/wr024i005p00755.

Correndo, Adrian A., Austin Pearce, Carl H. Bolster, John T. Spargo, Deanna Osmond, and Ignacio A. Ciampitti. 2023. "The Soiltestcorr r Package: An Accessible Framework for Reproducible Correlation Analysis of Crop Yield and Soil Test Data." *SoftwareX* 21: 101275. https://doi.org/https://doi.org/10.1016/j.softx.2022.101275.

Genuchten, M. Th. van. 1980. "A Closed-Form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils." *Soil Science Society of America Journal* 44 (5): 892–98. https://doi.org/10.2136/sssaj1980.03615995004400050002x.

Hall, Caitlyn A., Sheila M. Saia, Andrea L. Popp, Nilay Dogulu, Stanislaus J. Schymanski, Niels Drost, Tim van Emmerik, and Rolf Hut. 2022. "A Hydrologist's Guide to Open Science." *Hydrology and Earth System Sciences* 26 (3): 647–64. https://doi.org/10.5194/hess-26-647-2022.

Haynes, Sam. 2023. "OLS: Capacity Building in Open Science with a Peer-Led, Global, and Diverse Community." *Edinburgh Open Research*, June. https://doi.org/10.2218/eor.2023.8120.

MeterGroup. 2023. *Mini Disk Infiltrometer - Meter Group*. http://publications.metergroup.com/Manuals/20421_Mini_Disk_Manual_Web.pdf.

Philip, J. R. 1957. "THE THEORY OF INFILTRATION: 4. SORPTIVITY AND ALGEBRAIC INFILTRATION EQUATIONS." *Soil Science* 84: 257–64.

Sousa, Decíola Fernandes de, Sueli Rodrigues, Herdjania Veras de Lima, and Lorena Torres Chagas. 2020. "R Software Packages as a Tool for Evaluating Soil Physical and Hydraulic Properties." *Comput. Electron. Agric.* 168 (C). https://doi.org/10.1016/j.compag.2019.105077.

Surda, Peter, Justina Vitkova, Katarina Brezianska, and Lubomir Lichner. 2019. "Influence of the Infiltration Disk Radius on Determination of Unsaturated Hydraulic Conductivity of Non-Structural Sandy Soil." *IOP Conference Series: Earth and Environmental Science* 221 (March): 012024. https://doi.org/10.1088/1755-1315/221/1/012024.

Zhang, Renduo. 1997. "Infiltration Models for the Disk Infiltrometer." *Soil Science Society of America Journal* 61 (6): 1597–1603. https://doi.org/10.2136/sssaj1997.03615995006100060008x.