Import - Lithic analysis Ramioul

Lisa Schunk

2021-02-04

# Goal of the script

This script reads the xlsx file (database techno-typological analysis) generated with E4 and formats the data for a statistical analysis.  
The script will:

1. Reads in the original xlsx file
2. Changes and sort the data in order to do stats
3. Saves the data as a new xlsx file and R object

dir\_in <- "analysis/Ramioul/raw\_data/"  
dir\_out <- "analysis/Ramioul/derived\_data/"

Raw data must be located in “analysis/Ramioul/raw\_data/”.  
Formatted data will be saved in “analysis/Ramioul/derived\_data/”. The knit directory for this script is the project directory.

# Load packages

library(openxlsx)

Warning: package 'openxlsx' was built under R version 4.0.3

library(readxl)  
library(R.utils)  
library(tools)  
library(data.table)

Warning: package 'data.table' was built under R version 4.0.3

library(chron)  
library(dplyr)

Warning: package 'dplyr' was built under R version 4.0.3

# Get name, path and information of the file

data\_file <- list.files(dir\_in, pattern = "\\.xlsx$", full.names = TRUE)  
md5\_in <- md5sum(data\_file)  
info\_in <- data.frame(files = basename(names(md5\_in)), checksum = md5\_in,   
 row.names = NULL)

The checksum (MD5 hashes) of the imported files are:

files checksum  
1 Ramioul.xlsx 20e268e7a32dbda976241b3124c73557

# Read in original xlsx-file

imp\_data <- read.xlsx(xlsxFile = data\_file, sheet = 1, startRow = 1, colNames = TRUE,  
 rowNames = FALSE, skipEmptyRows = FALSE)   
#select the columns to keep   
keep\_col <- c(1:2, 9:19, 21:35, 37:38, 40:43)  
data\_final <- imp\_data[,keep\_col]  
  
data\_final$length <- as.numeric(data\_final$length)  
data\_final$width <- as.numeric(data\_final$width)  
data\_final$thickness <- as.numeric(data\_final$thickness)  
data\_final$weight <- as.numeric(data\_final$weight)  
data\_final$perimeter.basis.back <- as.numeric(data\_final$perimeter.basis.back)  
data\_final$perimeter.arch <- as.numeric(data\_final$perimeter.arch)

Error in `$<-.data.frame`(`\*tmp\*`, perimeter.arch, value = numeric(0)): replacement has 0 rows, data has 20

data\_final$perimeter.active.edge <- as.numeric(data\_final$perimeter.active.edge)  
data\_final$perimeter.total <- as.numeric(data\_final$perimeter.total)  
data\_final$thickness.back <- as.numeric(data\_final$thickness.back)  
  
str(data\_final)

'data.frame': 20 obs. of 34 variables:  
 $ site : chr "Ramioul" "Ramioul" "Ramioul" "Ramioul" ...  
 $ ID : chr "R-001" "R-002" "R-003" "R-004" ...  
 $ raw.material : chr "baltic\_flint" "baltic\_flint" "baltic\_flint" "baltic\_flint" ...  
 $ technological.class : chr "Keilmesser" "Keilmesser" "scraper" "scraper" ...  
 $ artefact.state : chr "complete" "complete" "complete" "complete" ...  
 $ blank : chr "core" "core" "core" "flake" ...  
 $ morpho.type : chr "Balve" "Klausennische" NA NA ...  
 $ cortex : chr "YES" "YES" "YES" "YES" ...  
 $ cortex.percentage : chr "<25" "<25" "25-50" "25-50" ...  
 $ cortex.location : chr "back" "back" "back" "medial\_dorsa" ...  
 $ morphology.back : chr "cortex/partly\_retouched" "cortex/unworked" NA NA ...  
 $ retouch.active.edge : chr "YES" "YES" "YES" "YES" ...  
 $ retouch.type.edge : chr "bifacial" "semi-bifacial" "semi-bifacial" "unifacial" ...  
 $ tip.morphology : chr "rounded" "rounded" "rounded" "undeterminable" ...  
 $ application.Pradnik.method : chr "YES" "YES" NA NA ...  
 $ frequency.application.Pradnik.method: chr "one" "one" NA NA ...  
 $ type.lateral.sharpening.spall : num NA NA NA NA NA NA NA NA NA NA ...  
 $ tool.lateralisation : chr "dex." "dex." NA NA ...  
 $ length : num 50 44 51 42 54 53 76 75 99 52 ...  
 $ width : num 33 27 31 24 27 31 37 43 42 35 ...  
 $ thickness : num 16 21 20 9 16 16 21 23 25 14 ...  
 $ weight : num 0.025 0.02 0.029 0.01 0.016 0.018 0.062 0.066 0.075 0.0017 ...  
 $ perimeter.basis.back : num 6.7 6.4 6.5 0 0 5.7 9.5 7.3 0 8.2 ...  
 $ perimeter.distal.posterior.part : num 3 2.5 3.1 0 0 3.1 1.9 4.4 0 2 ...  
 $ perimeter.active.edge : num 5 4 4.9 0 0 4.4 6.5 7.6 0 4.5 ...  
 $ perimeter.total : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ thickness.back : num 16 21 14 0 0 6 21 17 0 5 ...  
 $ taphonomic.visual.inspection : chr "sharp\_edges\_and\_patinated\_surface" "sharp\_edges\_and\_patinated\_surface" "sharp\_edges\_and\_patinated\_surface" "sharp\_edges\_and\_patinated\_surface" ...  
 $ tool.edges.preservation : chr "edges\_preserved" "edges\_preserved" "edges\_preserved" "edges\_preserved" ...  
 $ macroscopically.visible.use-wear : chr "NO" "NO" "NO" "NO" ...  
 $ use-wear.analysis : chr "YES" "YES" "NO" "YES" ...  
 $ 3D-scan : chr "YES" "YES" "YES" "YES" ...  
 $ schistosity : chr "N/A" "N/A" "N/A" "N/A" ...  
 $ orientation.schistosity : num NA NA NA NA NA NA NA NA NA NA ...

# Data analsysis - sorting

## Dimension

# keeps only columns relevant for dimensions and sorts them based on   
# their technological class  
keep\_col <- c(1:2, 4:5,19:21)  
dimensions <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_dimensions <- dimensions[3:11, ]   
PS\_dimensions <- dimensions[12:14, ]   
F\_dimensions <- dimensions[1:2, ]  
S\_dimensions <- dimensions[15:20, ]

## Perimeter

# keeps only columns relevant for perimeter measurements and sorts them   
# based on their technological class  
keep\_col <- c(1:2, 4:5,7,23:26)  
perimeter <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_perimeter <- perimeter[3:11, ]   
PS\_perimeter <- perimeter[12:14, ]

## Weight

# keeps only columns relevant for weight measurements and sorts them based   
# on their technological class  
keep\_col <- c(1:2, 4:5, 22)  
weight <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_weight <- weight[3:11, ]   
PS\_weight <- weight[12:14, ]   
F\_weight <- weight[1:2, ]   
S\_weight <- weight[15:20, ]

## Raw material

# keeps only columns relevant for raw material classification and sorts them   
# based on their technological class  
keep\_col <- c(1:2, 4:5, 3)  
raw\_material <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_raw\_material <- raw\_material[3:11, ]   
PS\_raw\_material <- raw\_material[12:14, ]   
F\_raw\_material <- raw\_material[1:2, ]   
S\_raw\_material <- raw\_material[15:20, ]

## Cortex + blanks

# keeps only columns relevant for cortex and blank classification and sorts them   
# based on their technological class  
keep\_col <- c(1:2, 4:5, 6, 8:10)  
cortex\_blanks <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_cortex\_blanks <- cortex\_blanks[3:11, ]   
PS\_cortex\_blanks <- cortex\_blanks[12:14, ]

## Back

# keeps only columns relevant for back modifications and sorts them based   
# on their technological class  
keep\_col <- c(1:2, 4:5, 11, 27)  
back <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_back <- back[3:11, ]   
PS\_back <- back[12:14, ]

## Edge retouch

# keeps only columns relevant for edge retouch classification and sorts them   
# based on their technological class  
keep\_col <- c(1:2, 4:5, 12:13, 29)  
edge\_retouch <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_edge\_retouch <- edge\_retouch[3:11, ]  
PS\_edge\_retouch <- edge\_retouch[12:14, ]   
F\_edge\_retouch <- edge\_retouch[1:2, ]   
S\_edge\_retouch <- edge\_retouch[15:20, ]

## Morpho type

# keeps only columns relevant for morpho type classification and sorts them   
# based on their technological class  
keep\_col <- c(1:2, 4:5, 7, 19:21)  
morpho.type <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_morpho.type <- morpho.type[3:11, ]   
PS\_morpho.type <- morpho.type[12:14, ]

## Application ‘Pradnik method’

# keeps only columns relevant for 'morpho type 'Pradnik method' classification   
# and sorts them based on their technological class  
keep\_col <- c(1:2, 4:5, 15:16)  
Pradnik.method <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_Pradnik.method <- Pradnik.method[3:11, ]  
PS\_Pradnik.method <- Pradnik.method[12:14, ]

## Lateralisation

# keeps only columns relevant for lateralisation and sorts them based on their  
# technological class  
keep\_col <- c(1:2, 4:5, 18)  
lateralisation <- data\_final[, keep\_col] %>% arrange(technological.class)  
  
KM\_lateralisation <- lateralisation[3:11, ]   
PS\_lateralisation <- lateralisation[12:14, ]

# Save data

## Format name of output file

file\_out <- "Ramioul\_lithic\_analysis"

The files will be saved as “~/Ramioul\_lithic\_analysis.[ext]”.

## Write to XLSX

write.xlsx(list(data = data\_final, dimensions = dimensions, KM\_dimensions = KM\_dimensions,  
 PS\_dimensions = PS\_dimensions, F\_dimensions = F\_dimensions, S\_dimensions =  
 S\_dimensions, KM\_perimeter = KM\_perimeter, PS\_perimeter = PS\_perimeter,  
 KM\_weight = KM\_weight, PS\_weight = PS\_weight, F\_weight = F\_weight,  
 S\_weight = S\_weight, KM\_raw\_material = KM\_raw\_material,PS\_raw\_material =  
 PS\_raw\_material, F\_raw\_material = F\_raw\_material, S\_raw\_material =  
 S\_raw\_material, KM\_cortex\_blanks = KM\_cortex\_blanks, PS\_cortex\_blanks =  
 PS\_cortex\_blanks, KM\_back = KM\_back, PS\_back = PS\_back, KM\_edge\_retouch  
 = KM\_edge\_retouch, PS\_edge\_retouch = PS\_edge\_retouch, F\_edge\_retouch =  
 F\_edge\_retouch, S\_edge\_retouch = S\_edge\_retouch, KM\_morpho.type =  
 KM\_morpho.type, PS\_morpho.type = PS\_morpho.type, KM\_Pradnik.method =  
 KM\_Pradnik.method, PS\_Pradnik.method = PS\_Pradnik.method,  
 KM\_lateralisation = KM\_lateralisation, PS\_lateralisation =  
 PS\_lateralisation),  
 file = paste0(dir\_out, file\_out, ".xlsx"))

## Save R object

saveObject(data\_final, file = paste0(dir\_out, file\_out, ".Rbin"))

# sessionInfo() and RStudio version

sessionInfo()

R version 4.0.2 (2020-06-22)  
Platform: x86\_64-w64-mingw32/x64 (64-bit)  
Running under: Windows 10 x64 (build 19041)  
  
Matrix products: default  
  
locale:  
[1] LC\_COLLATE=German\_Germany.1252 LC\_CTYPE=German\_Germany.1252   
[3] LC\_MONETARY=German\_Germany.1252 LC\_NUMERIC=C   
[5] LC\_TIME=German\_Germany.1252   
  
attached base packages:  
[1] tools stats graphics grDevices utils datasets methods   
[8] base   
  
other attached packages:  
[1] dplyr\_1.0.3 chron\_2.3-56 data.table\_1.13.6 R.utils\_2.10.1   
[5] R.oo\_1.24.0 R.methodsS3\_1.8.1 readxl\_1.3.1 openxlsx\_4.2.3   
  
loaded via a namespace (and not attached):  
 [1] Rcpp\_1.0.6 knitr\_1.31 magrittr\_2.0.1 tidyselect\_1.1.0   
 [5] R6\_2.5.0 rlang\_0.4.10 stringr\_1.4.0 xfun\_0.20   
 [9] DBI\_1.1.1 ellipsis\_0.3.1 htmltools\_0.5.1.1 assertthat\_0.2.1   
[13] yaml\_2.2.1 digest\_0.6.27 tibble\_3.0.5 lifecycle\_0.2.0   
[17] crayon\_1.4.0 zip\_2.1.1 purrr\_0.3.4 vctrs\_0.3.6   
[21] glue\_1.4.2 evaluate\_0.14 rmarkdown\_2.6 stringi\_1.5.3   
[25] pillar\_1.4.7 compiler\_4.0.2 cellranger\_1.1.0 generics\_0.1.0   
[29] pkgconfig\_2.0.3

RStudio version 1.3.1056.

END OF SCRIPT