Plots - use-wear archaeology

Lisa Schunk

2021-02-04 17:13:30

# Goal of the script

This script plots all variables to see which ones should be used for further analysis.  
Scatterplot of each variable will be plotted.

dir\_in <- "analysis/derived\_data/"  
dir\_out <- "analysis/plots"

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

# Load packages

library(R.utils)  
library(ggplot2)

Warning: package 'ggplot2' was built under R version 4.0.3

library(tools)  
library(tidyverse)

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

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

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

library(patchwork)

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

library(doBy)

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

library(ggrepel)

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

library(openxlsx)

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

library(wesanderson)

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

library(ggfortify)

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

# Get name, path and information of the file

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

The checksum (MD5 hashes) of the imported file is:

file checksum  
1 Use-wear.Rbin 558d5b8d978e0d27f0cf6d308b0734de

# Load data into R object

imp\_data <- loadObject(data\_file)  
str(imp\_data)

'data.frame': 150 obs. of 57 variables:  
 $ Sample : chr "MU-232" "MU-232" "MU-232" "MU-003" ...  
 $ Site : Factor w/ 3 levels "Balve","Buhlen",..: 1 1 1 1 1 1 1 1 1 1 ...  
 $ Tool.type : Factor w/ 4 levels "Keilmesser","Pradnik scraper",..: 1 1 1 1 1 1 1 1 1 4 ...  
 $ Raw.material : Factor w/ 2 levels "flint","lydite": 2 2 2 2 2 2 2 2 2 2 ...  
 $ Location : chr "B" "B" "B" "D" ...  
 $ Sublocation : chr "2" "2" "2" "1" ...  
 $ Area : chr "01" "01" "01" "01" ...  
 $ Spot : chr "a" "b" "c" "a" ...  
 $ Usewear.type : Factor w/ 11 levels "A","B","B2","C",..: 9 9 9 2 2 2 4 4 4 3 ...  
 $ Objective : Factor w/ 3 levels "20x07","50x075",..: 1 1 1 3 3 3 2 2 2 3 ...  
 $ Analysis.date : Date, format: "2020-09-07" "2020-09-07" ...  
 $ Analysis.time : 'times' num 0.631 0.631 0.631 0.631 0.632 ...  
 ..- attr(\*, "format")= chr "h:m:s"  
 $ Acquisition.date.time : chr "07.07.2020 16:58" "07.08.2020 10:35" "07.08.2020 12:10" "07.03.2020 10:44" ...  
 $ Axis.length.X : num 255 255 255 255 255 ...  
 $ Axis.size.X : num 1198 1198 1198 1198 1198 ...  
 $ Axis.spacing.X : num 0.213 0.213 0.213 0.213 0.213 ...  
 $ Axis.length.Y : num 255 255 255 255 255 ...  
 $ Axis.size.Y : num 1198 1198 1198 1198 1198 ...  
 $ Axis.spacing.Y : num 0.213 0.213 0.213 0.213 0.213 ...  
 $ Axis.length.Z : num 249564 99661 162726 38576 39610 ...  
 $ Axis.size.Z : num 65505 35461 32419 65340 66654 ...  
 $ Axis.spacing.Z : num 3.81 2.81 5.019 0.59 0.594 ...  
 $ NM.points.ratio.Z : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ Sq : num 3243 2493 4332 1912 1936 ...  
 $ Ssk : num 0.0634 -0.9445 0.1816 -0.058 -0.2928 ...  
 $ Sku : num 3.46 7.36 3.08 3.75 3.47 ...  
 $ Sp : num 10477 7460 12748 6231 5796 ...  
 $ Sv : num 10005 12962 16115 6843 6575 ...  
 $ Sz : num 20482 20422 28864 13075 12371 ...  
 $ Sa : num 2506 1813 3409 1464 1495 ...  
 $ Smr : num 0.551 0.697 0.388 0.784 0.586 ...  
 $ Smc : num 3754 2956 5778 2454 2429 ...  
 $ Sxp : num 6582 4878 7854 3949 4400 ...  
 $ Sal : num 25.9 20.5 23.4 24.4 24.9 ...  
 $ Str : num 0.321 0.215 0.241 0.784 0.767 ...  
 $ Std : num 42.5 93 51 103.7 106.7 ...  
 $ Sdq : num 0.603 0.376 0.557 0.301 0.298 ...  
 $ Sdr : num 9.99 5.11 10.54 4.13 4.09 ...  
 $ Vm : num 0.2094 0.1157 0.2311 0.0944 0.0828 ...  
 $ Vv : num 3.96 3.07 6.01 2.55 2.51 ...  
 $ Vmp : num 0.2094 0.1157 0.2311 0.0944 0.0828 ...  
 $ Vmc : num 2.78 1.82 3.63 1.59 1.6 ...  
 $ Vvc : num 3.56 2.73 5.53 2.31 2.24 ...  
 $ Vvv : num 0.403 0.342 0.48 0.238 0.275 ...  
 $ Maximum.depth.of.furrows: num 12698 14381 16377 7155 7130 ...  
 $ Mean.depth.of.furrows : num 2586 2471 3670 2350 2229 ...  
 $ Mean.density.of.furrows : num 2987 1790 1901 2032 2098 ...  
 $ First.direction : num 44.9809 90.00638 89.98321 0.01527 0.00574 ...  
 $ Second.direction : num 26.5 135 63.5 116.5 135 ...  
 $ Third.direction : num 63.5 116.4 45 135 90 ...  
 $ Isotropy : num 13.5 64.5 14.9 87 86.3 ...  
 $ epLsar : num 0.00368 0.0024 0.00301 0.00161 0.00236 ...  
 $ NewEplsar : num 0.0181 0.0177 0.0179 0.0171 0.0171 ...  
 $ Asfc : num 12.8 6.85 12.12 5.51 5.36 ...  
 $ Smfc : num 2.51 67.38 48.16 94.68 55.32 ...  
 $ HAsfc9 : num 0.629 0.444 0.496 0.666 0.75 ...  
 $ HAsfc81 : num 0.81 2.106 1.515 0.845 0.704 ...  
 - attr(\*, "comment")= Named chr [1:44] "µm" "points" "µm" "µm" ...  
 ..- attr(\*, "names")= chr [1:44] "Axis length - X" "Axis size - X" "Axis spacing - X" "Axis length - Y" ...

The imported file is: “~/analysis/derived\_data/Use-wear.Rbin”

# Prepare variables

## Define numeric variables

num.var <- 24:length(imp\_data)

The following variables will be used:

[24] Sq  
[25] Ssk  
[26] Sku  
[27] Sp  
[28] Sv  
[29] Sz  
[30] Sa  
[31] Smr  
[32] Smc  
[33] Sxp  
[34] Sal  
[35] Str  
[36] Std  
[37] Sdq  
[38] Sdr  
[39] Vm  
[40] Vv  
[41] Vmp  
[42] Vmc  
[43] Vvc  
[44] Vvv  
[45] Maximum.depth.of.furrows  
[46] Mean.depth.of.furrows  
[47] Mean.density.of.furrows  
[48] First.direction  
[49] Second.direction  
[50] Third.direction  
[51] Isotropy  
[52] epLsar  
[53] NewEplsar  
[54] Asfc  
[55] Smfc  
[56] HAsfc9  
[57] HAsfc81

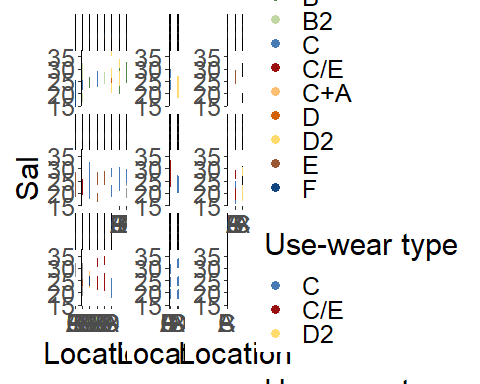
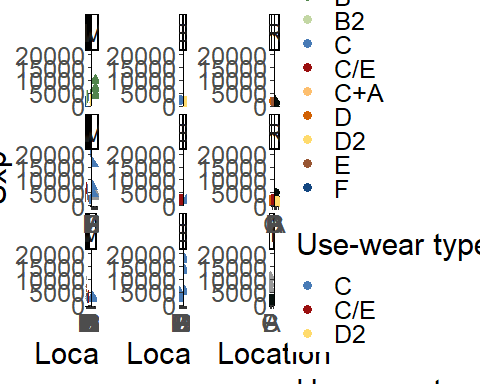
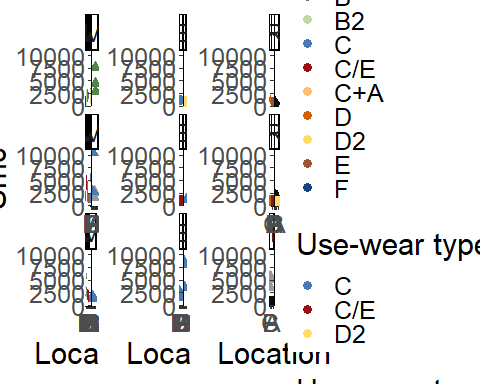
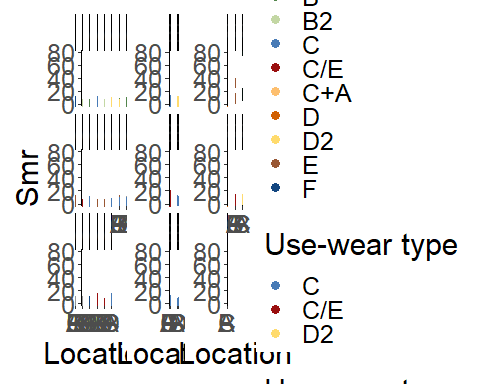
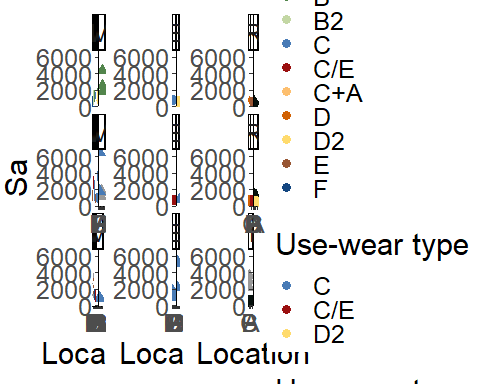
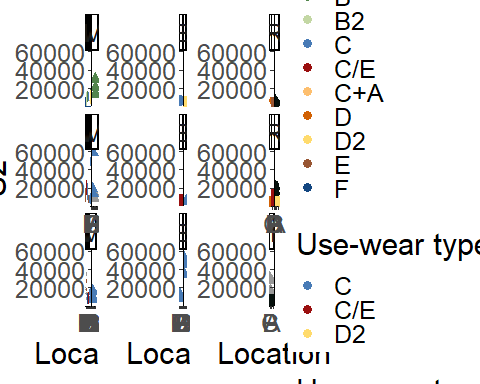
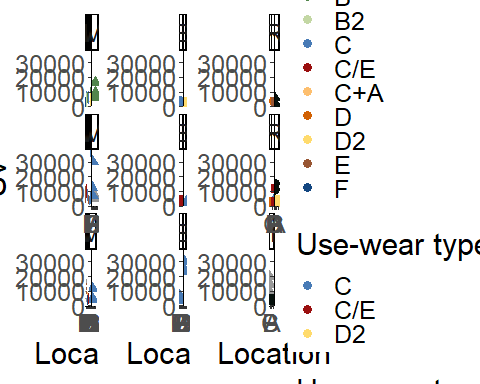
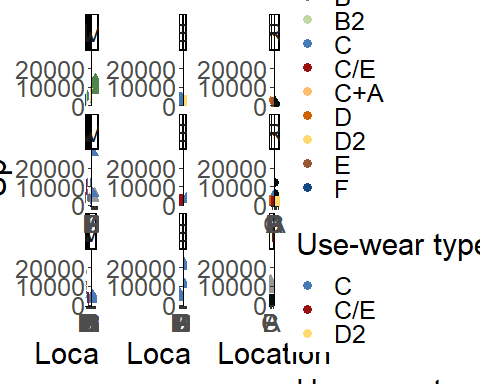
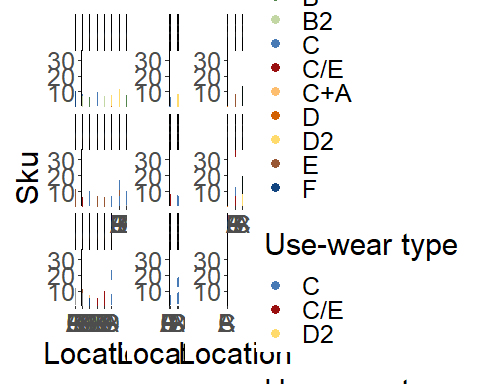
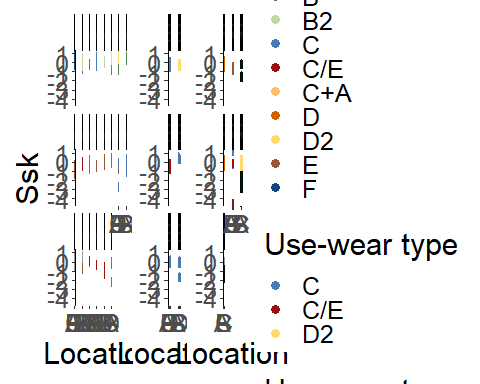
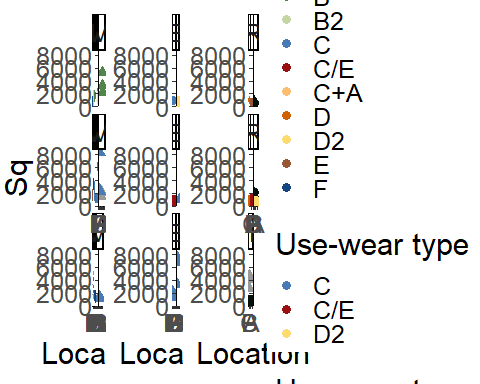
# Plot each of the selected numeric variables

## Colour definitions for use-wear types

#05100c black   
#999999 grey   
#52854c green   
#c3d7a4 light green   
#487bb6 blue   
#a6cee3 light blue   
#9a0f0f red  
#d16103 orange  
#fdbf6f apricot  
#ffdb6d yellow  
#985633 brown   
#134680 dark blue  
  
  
custom.col <- data.frame(type = levels(imp\_data$Usewear.typ),   
 col = c("#999999", "#52854c", "#c3d7a4", "#487bb6", "#9a0f0f",  
 "#fdbf6f",  
 "#d16103", "#ffdb6d", "#985633", "#134680", "#05100c"))

## Plot of all samples with raw material, variables and use-wear type as information

# splits the data in the individual 35 samples  
imp\_data[["Sample\_material"]] <- paste(imp\_data$Raw.material, imp\_data$Sample, sep = " ")  
  
sp <- split(imp\_data, imp\_data[["Site"]])  
  
usewear <- levels(imp\_data$Usewear.type)  
  
  
for (i in num.var){  
   
 # gets the min/max range of the data set   
 range\_var <- range(imp\_data[[i]])   
   
 # plot  
 p <- vector(mode = "list", length = length(sp))  
 names(p) <- names(sp)  
   
 for (j in seq\_along(sp)) {  
 col\_j <- custom.col[custom.col$type %in% levels(factor(sp[[j]][["Usewear.type"]])), "col"]  
  
 p[[j]] <- ggplot(data = sp[[j]], aes\_string(x = "Location", y = names(imp\_data)[i],  
 colour = "Usewear.type", shape =  
 "Sublocation")) +   
 # avoids overplotting  
 geom\_jitter(size = 3, position = position\_jitter(width = 0.4, seed = 1)) +   
 coord\_cartesian(ylim = range\_var) +   
 theme\_classic() +  
 labs(colour = "Use-wear type") +  
 facet\_wrap(~ Sample\_material, nrow = 3) +  
 labs(y = gsub("\\.", " ", names(imp\_data)[i])) +  
 scale\_colour\_manual(values = col\_j) +  
 theme(text = element\_text(size = 23)) +   
 if(j != 1) ylab(NULL)   
  
 }  
 p\_all <- wrap\_plots(p) + plot\_layout(width = c(8/13, 2/13, 3/13), guides = "collect")   
   
 print(p\_all)  
   
 # saves the plots   
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_plot\_",   
 names(imp\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = p\_all, path = dir\_out, device = "pdf", width = 65,  
 height = 53, units = "cm" )  
}

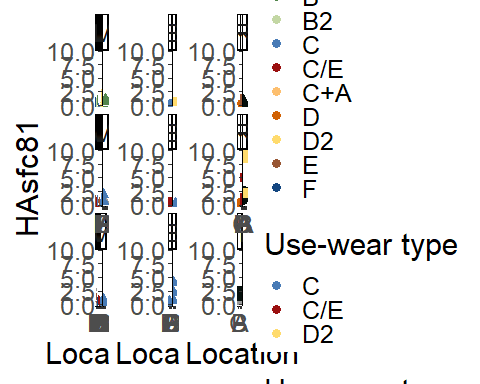
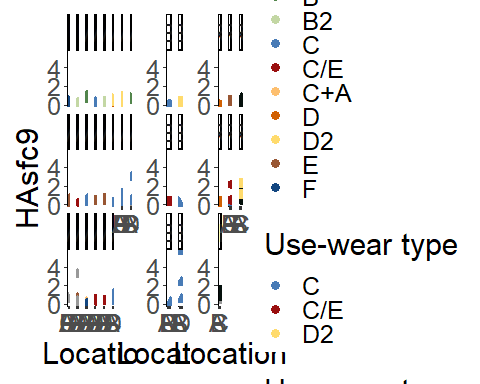
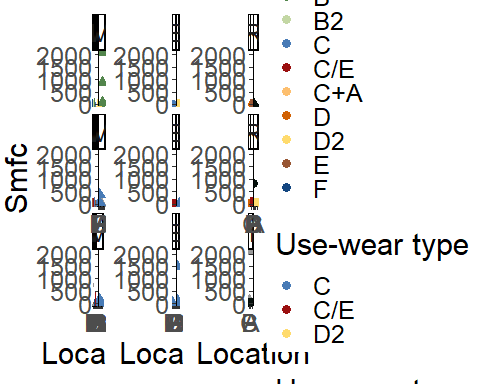
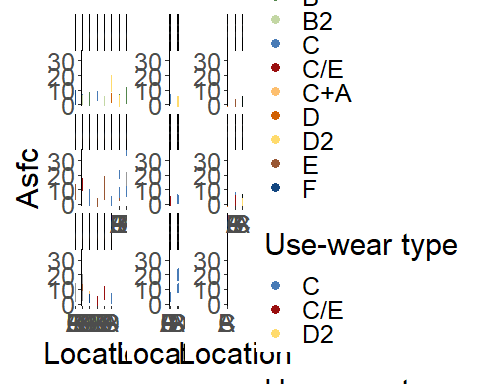
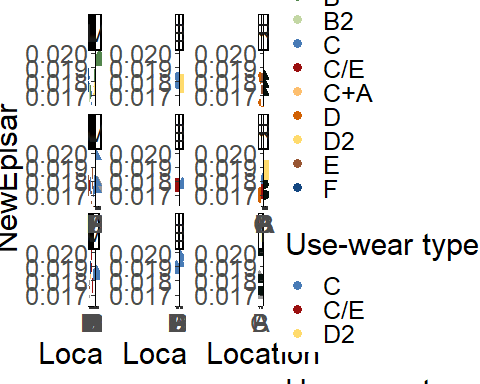
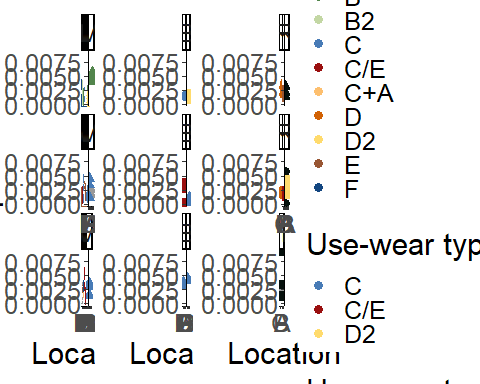
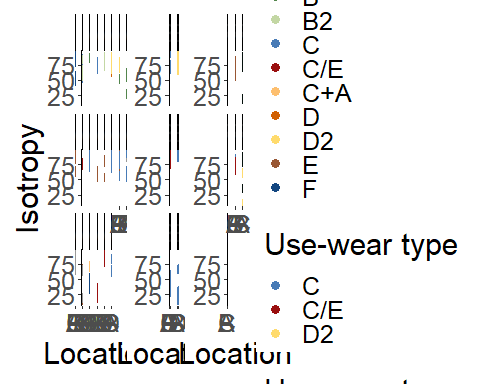
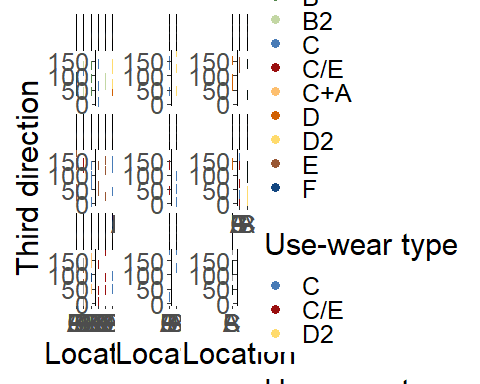
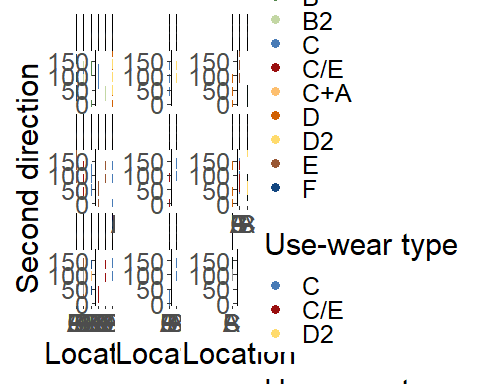
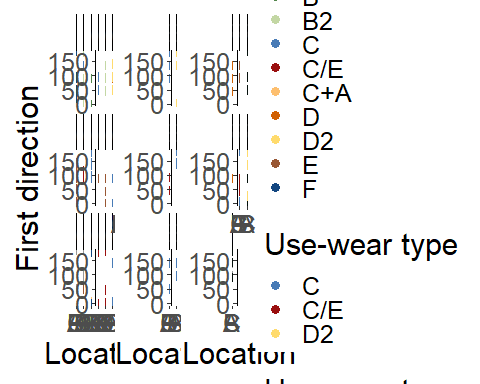
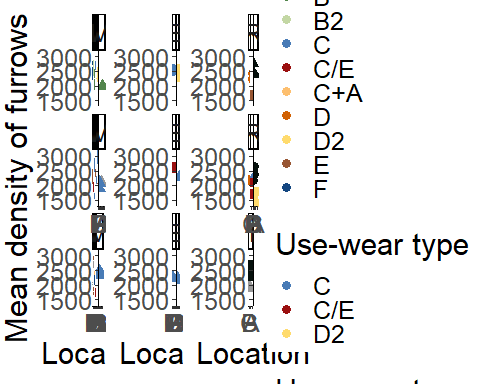
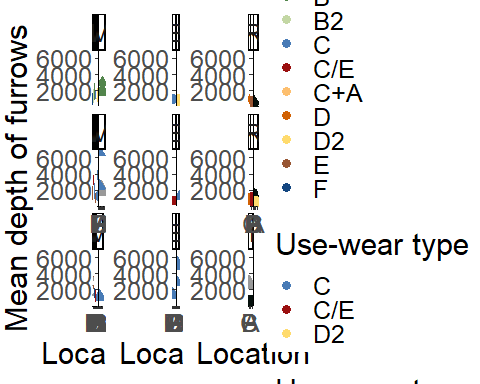
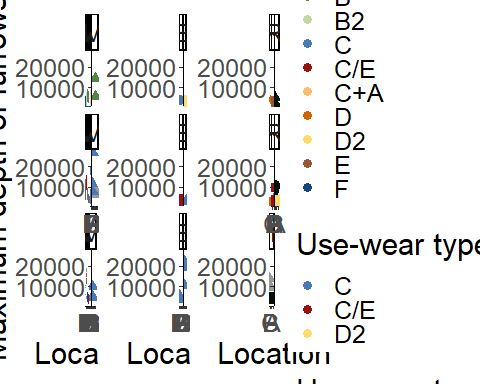
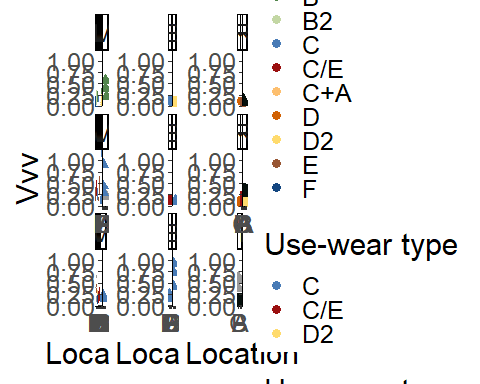
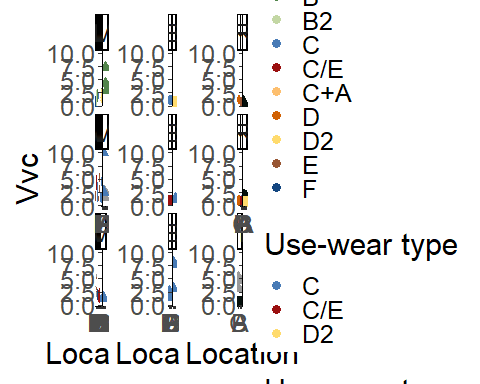
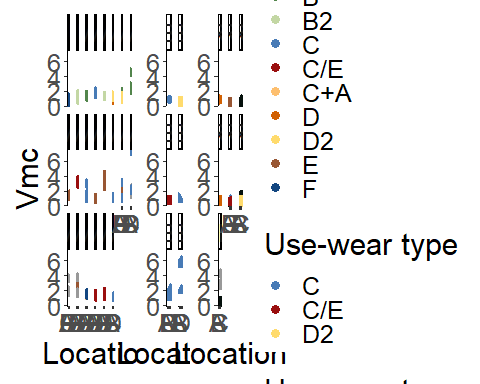
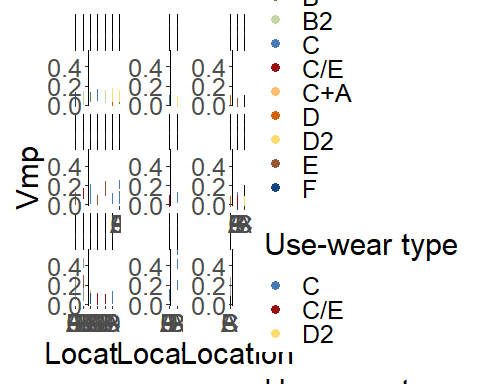
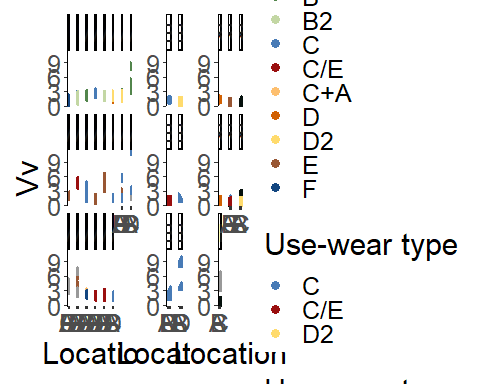
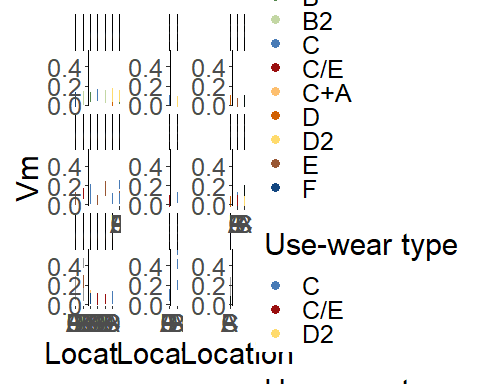
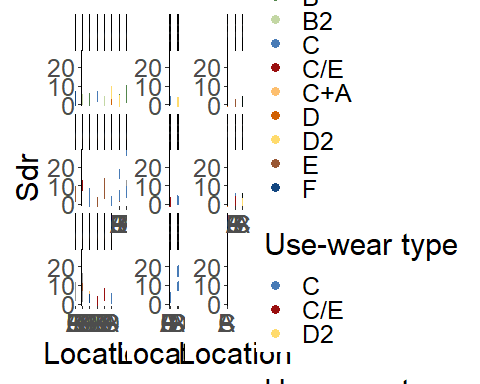
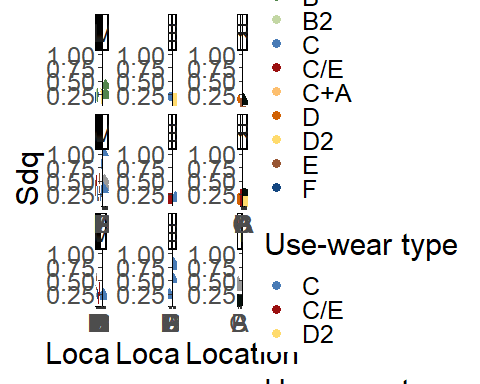
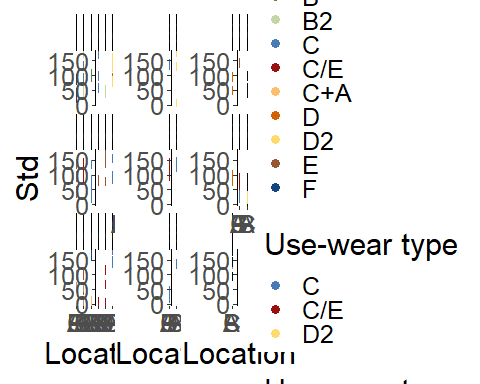
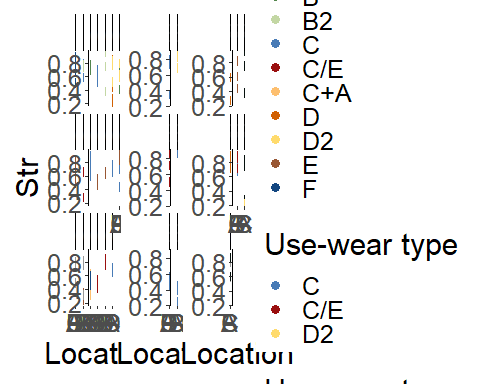


Warning: Removed 5 rows containing missing values (geom\_point).

Warning: Removed 4 rows containing missing values (geom\_point).

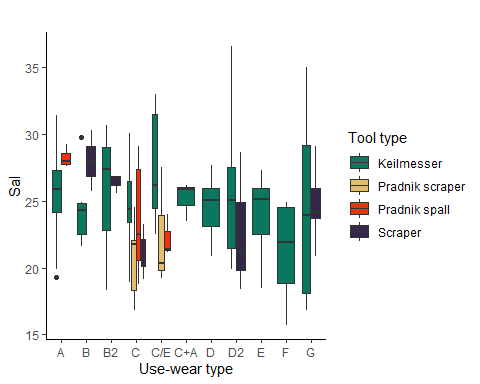
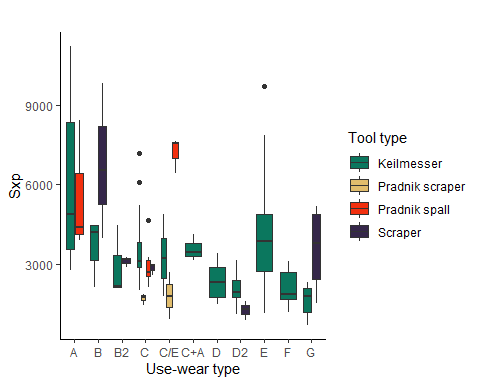
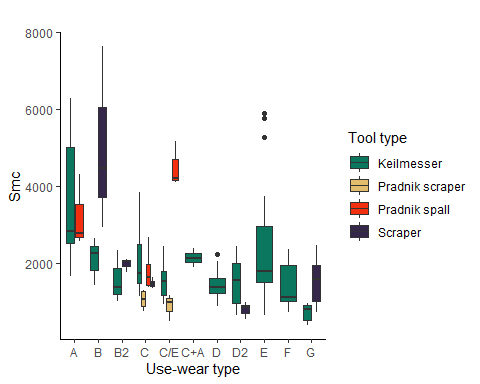
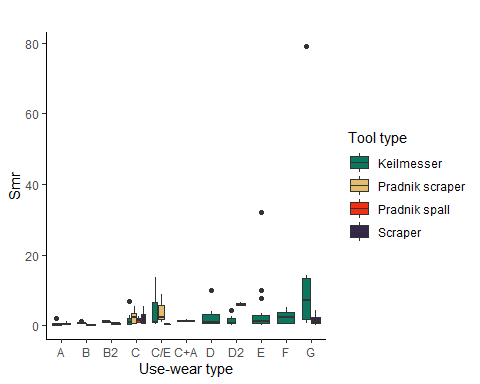
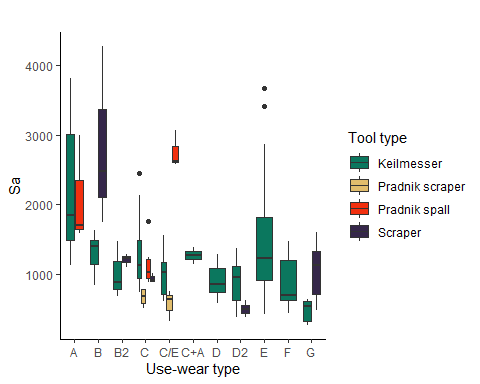
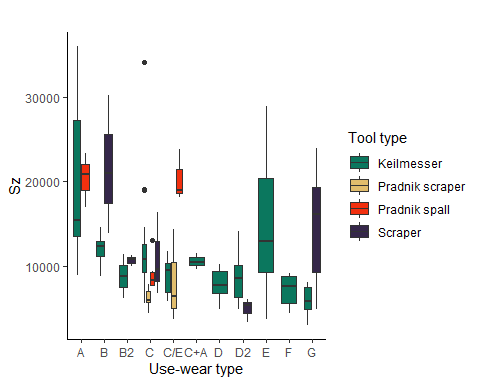
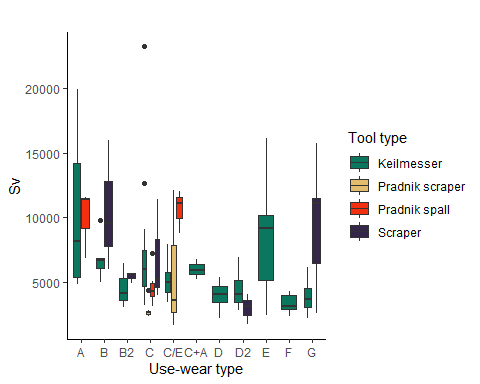
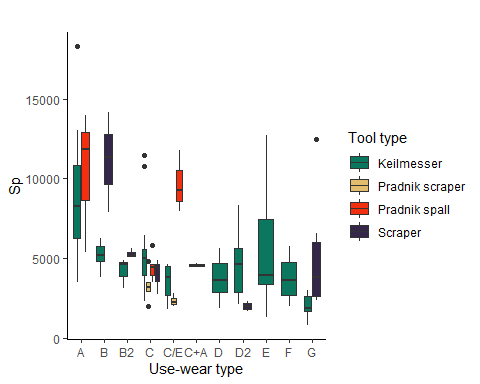
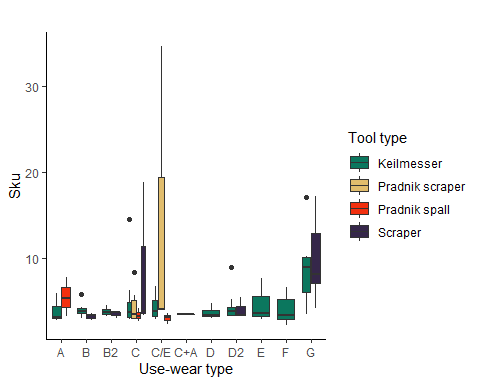
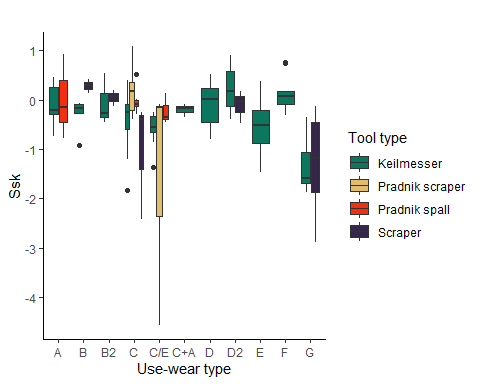
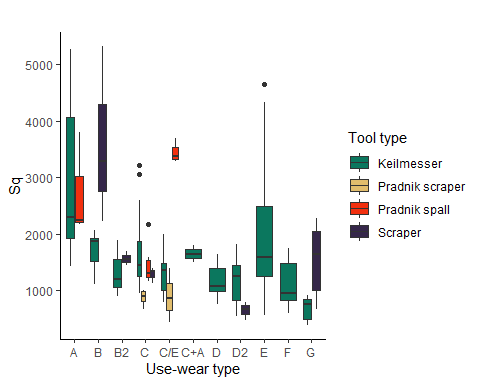
Warning: Removed 5 rows containing missing values (geom\_point).

Warning: Removed 4 rows containing missing values (geom\_point).

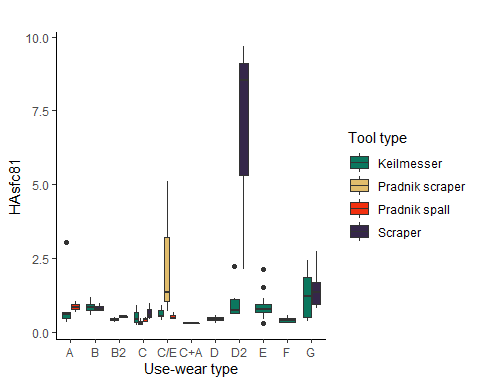
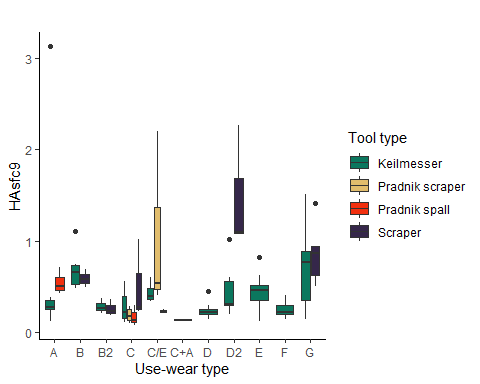
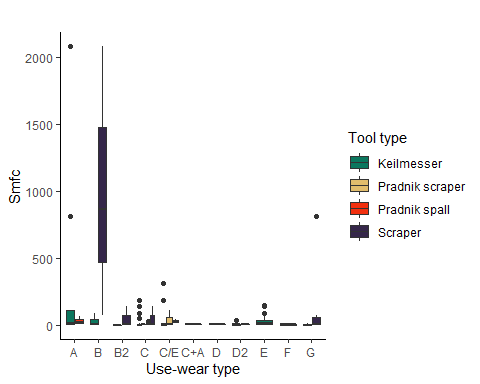
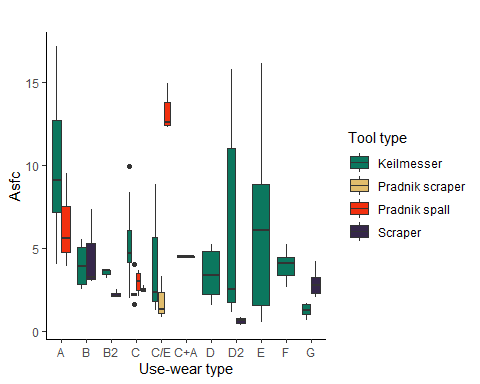
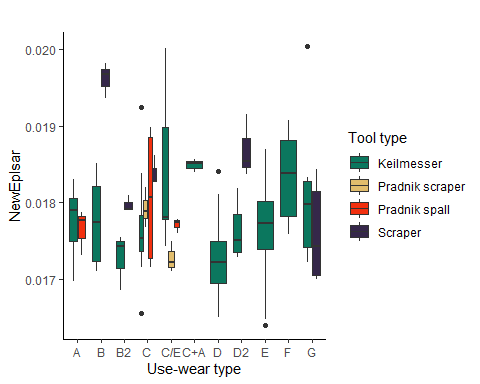
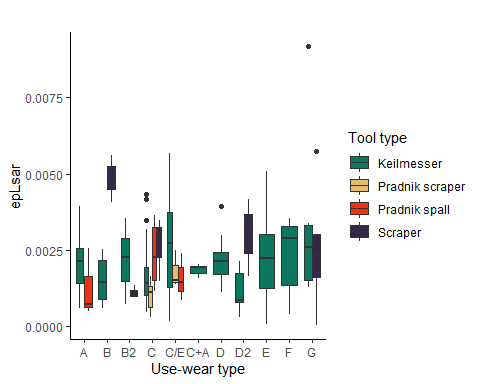
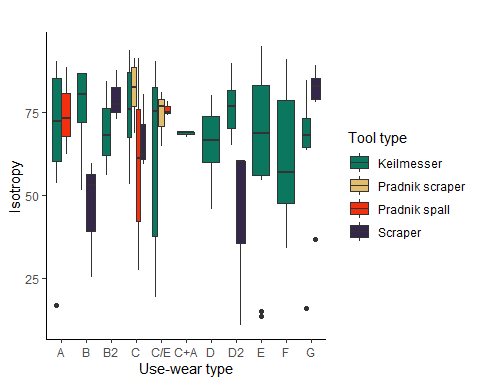
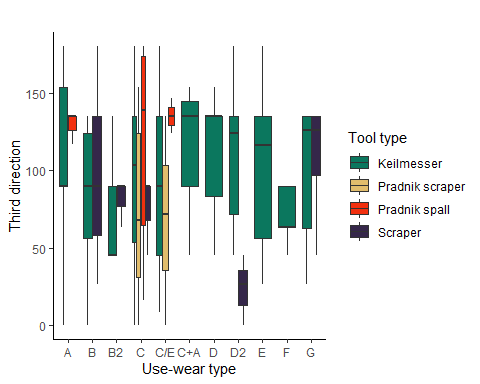
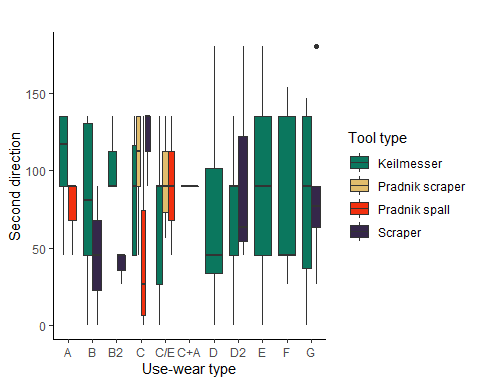
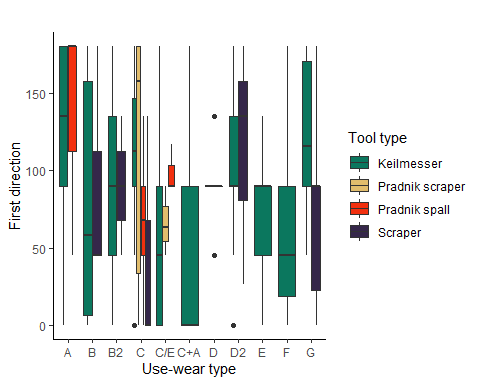
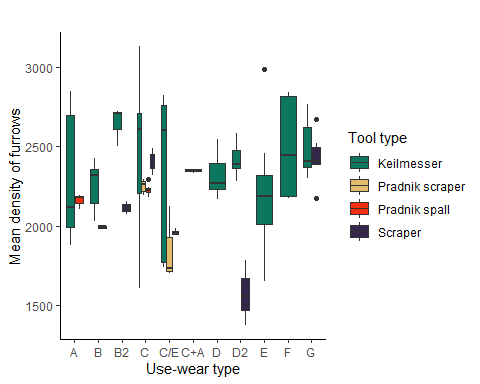
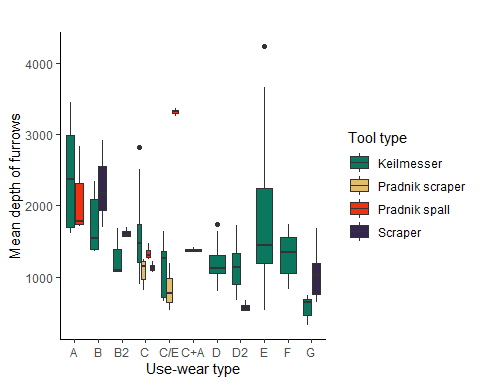
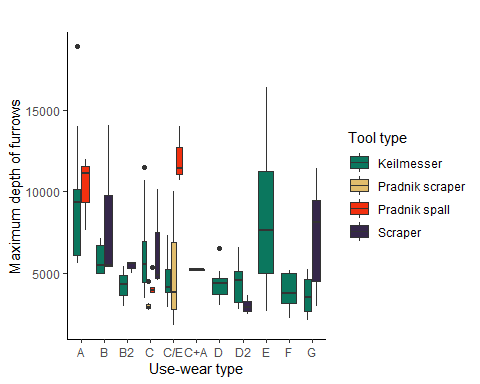
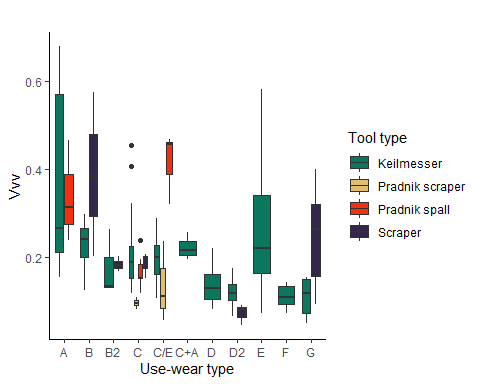
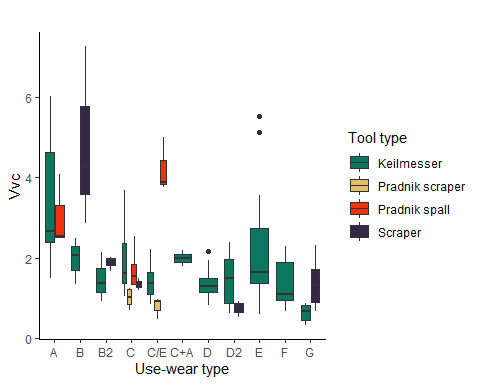
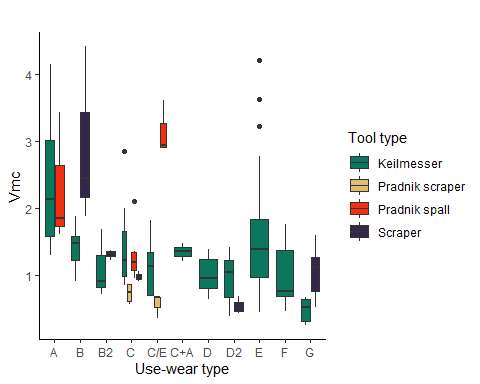
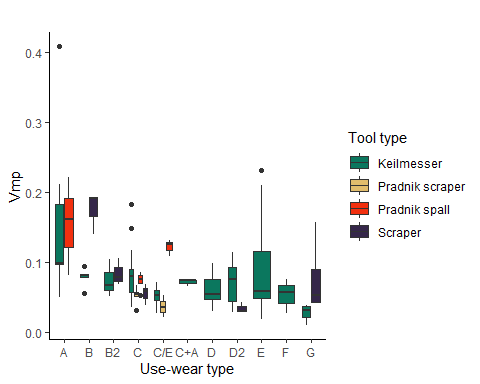
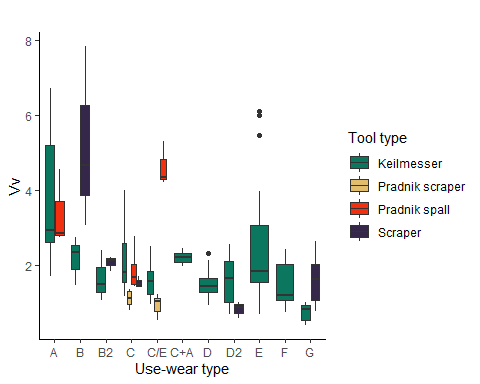
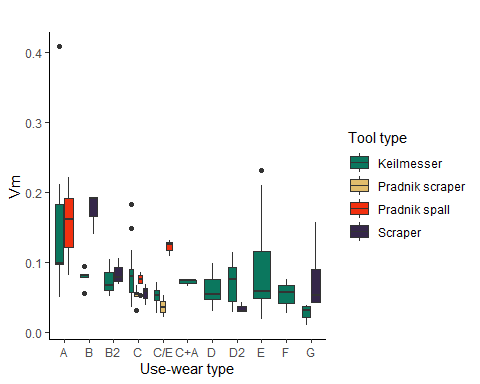
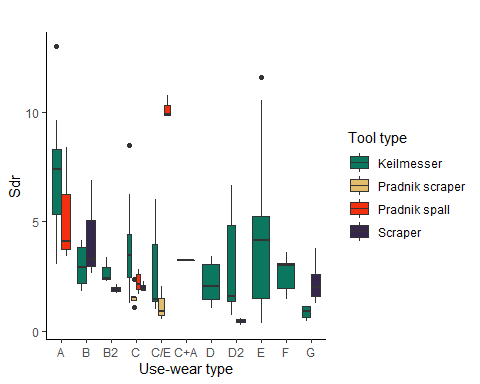
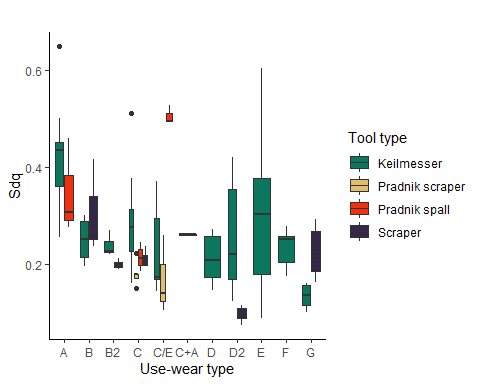
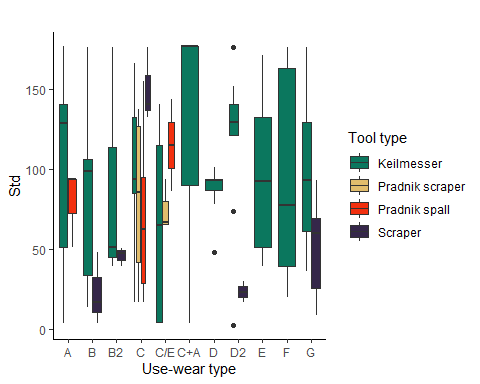
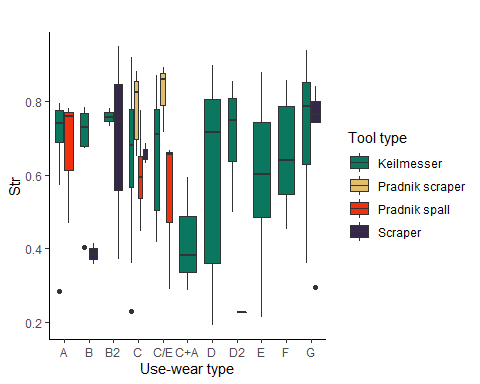


## Boxplot of all the variables combined with the use-wear type (without outliers)

# Wes Anderson color palettes Rushmore = c("#E1BD6D", "#EABE94", "#0B775E", "#35274A" ,"#F2300F")  
custom.col2 <- data.frame(type = levels(imp\_data$Tool.typ),   
 col = c( "#0B775E", "#E1BD6D", "#F2300F", "#35274A"))   
imp\_data$col <- custom.col2[imp\_data$Tool.typ, "col"]  
  
  
# excludes the outliers   
# adds the indices as row numbers   
imp\_data <- imp\_data %>% mutate(id = row\_number())  
imp\_data2 <- imp\_data[-c(55, 63, 115, 116), ]  
  
  
# plot  
for (i in num.var){  
   
 p2 <- ggplot(data = imp\_data2, aes\_string(x = "Usewear.type", y = names(imp\_data)[i],  
 fill = "Tool.type")) +  
 geom\_boxplot() +  
 scale\_fill\_manual(values = custom.col2$col)+  
 theme\_classic() +  
 labs( x = "Use-wear type", title = " ") +  
 labs(y = gsub("\\.", " ", names(imp\_data)[i])) +  
 labs(fill = "Tool type")   
   
 print(p2)  
  
 # saves the plots   
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_boxplot\_",   
 names(imp\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = p2, path = dir\_out, device = "pdf", width = 17,  
 height = 25, units = "cm")  
   
}

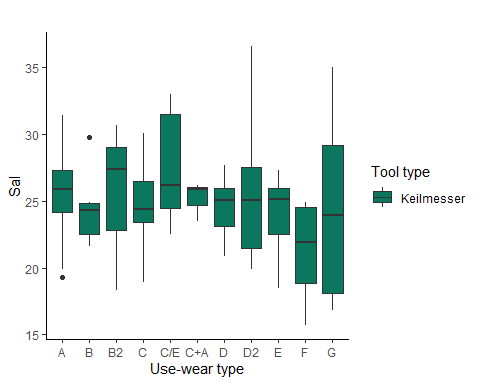
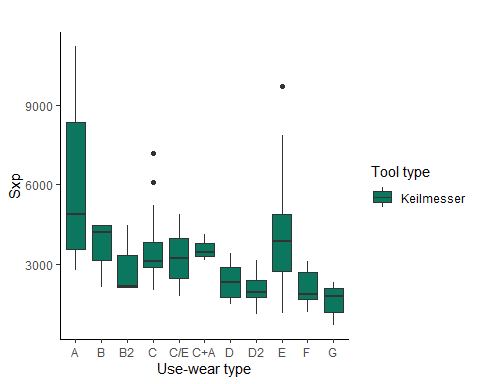
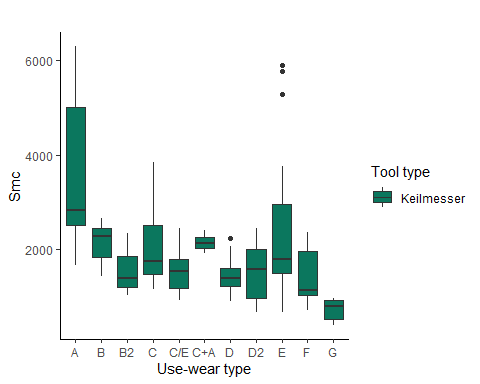
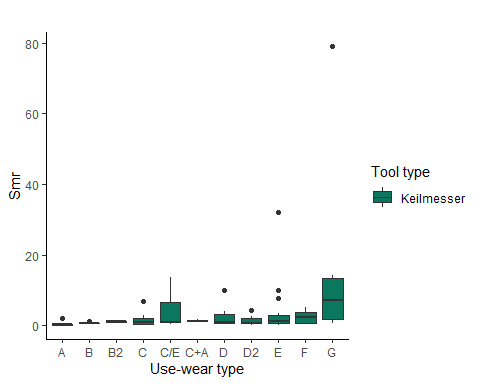
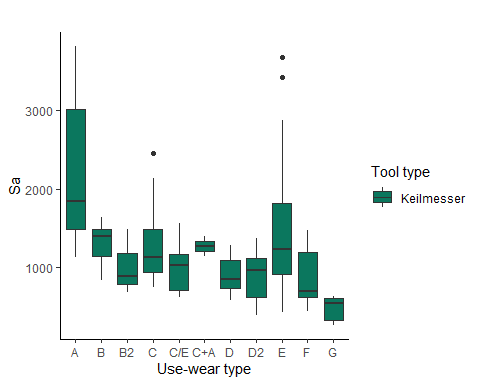
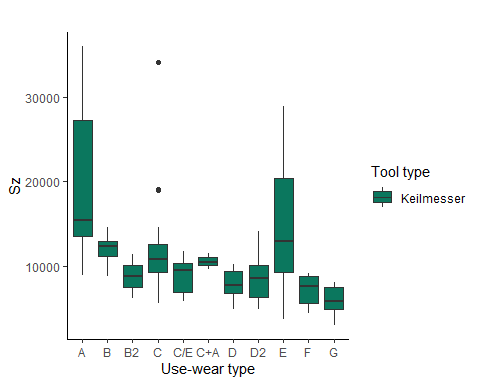
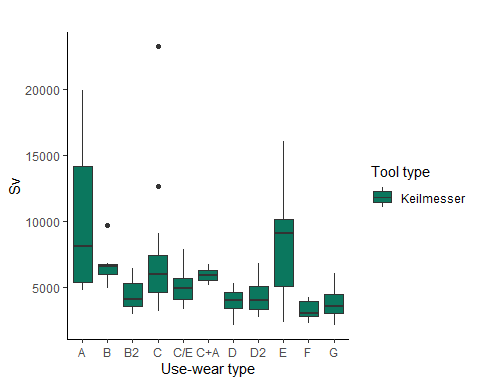
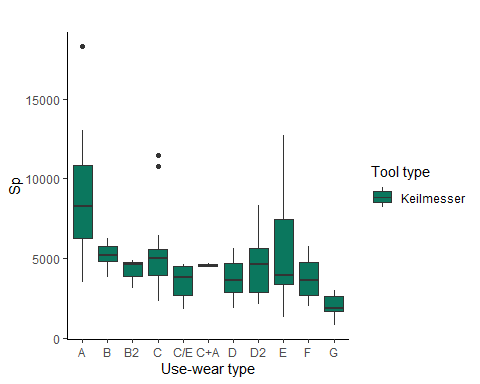
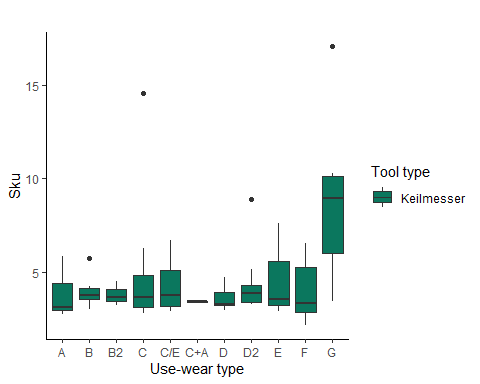
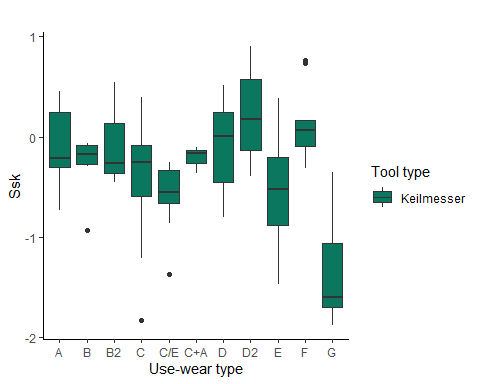
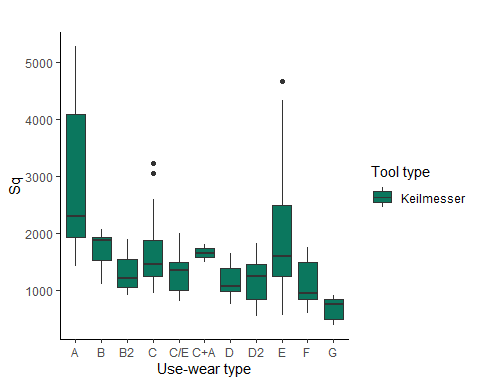


Warning: Removed 9 rows containing non-finite values (stat\_boxplot).  
  
Warning: Removed 9 rows containing non-finite values (stat\_boxplot).

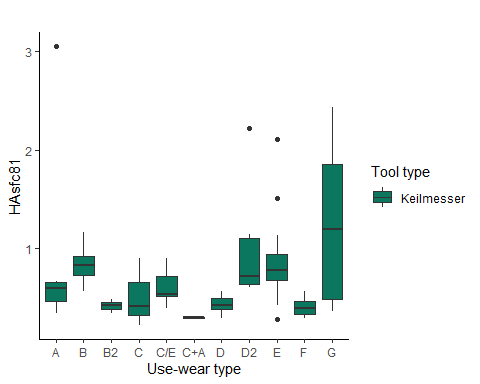
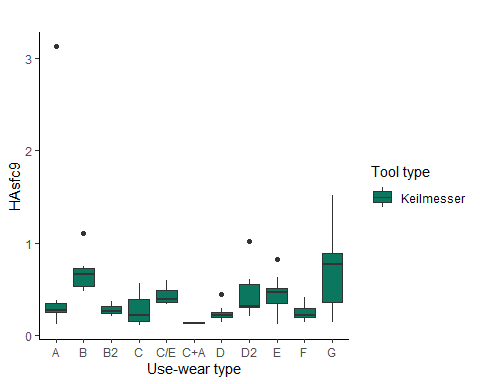
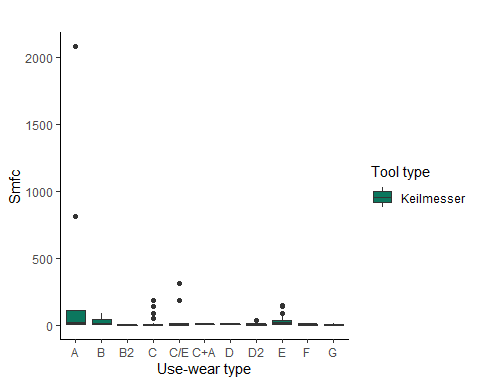
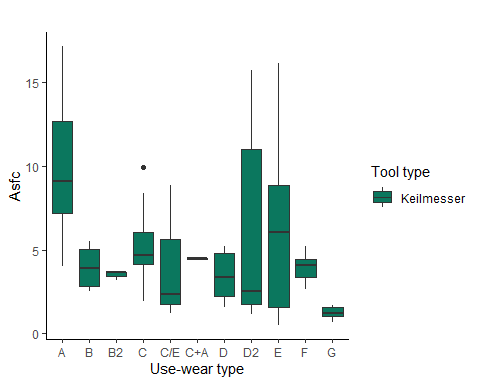
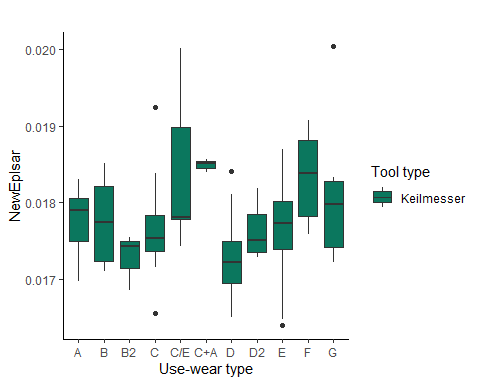
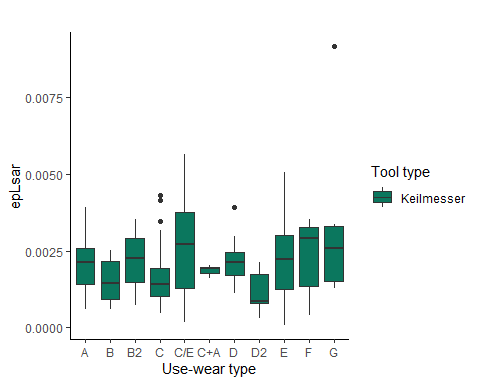
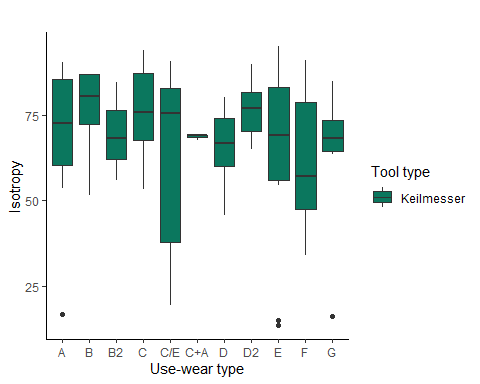
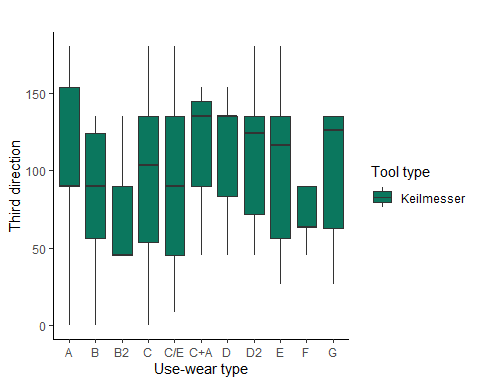
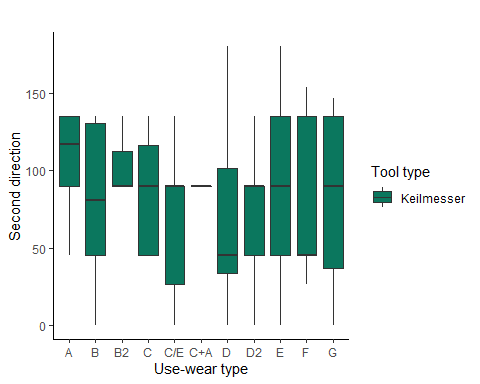
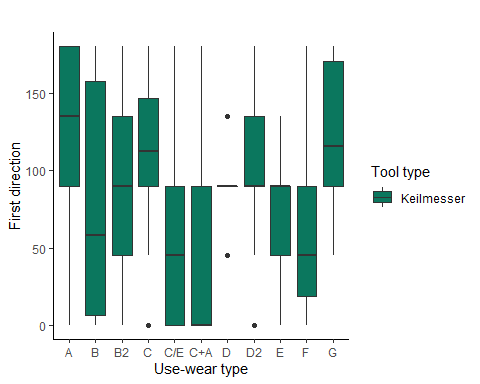
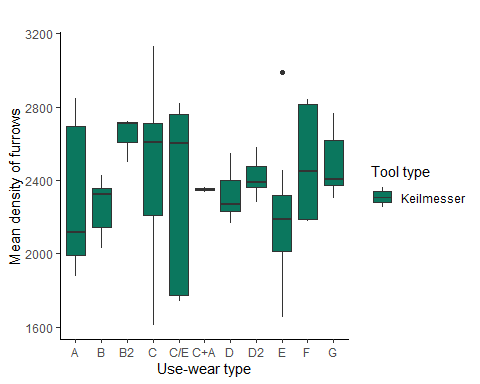
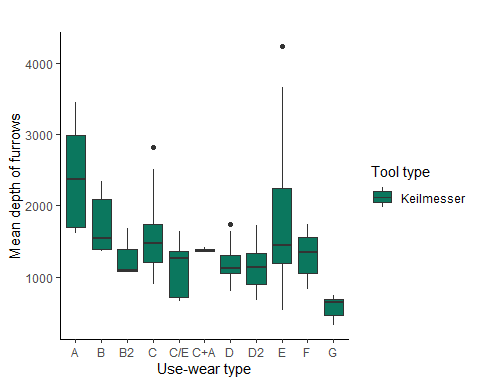
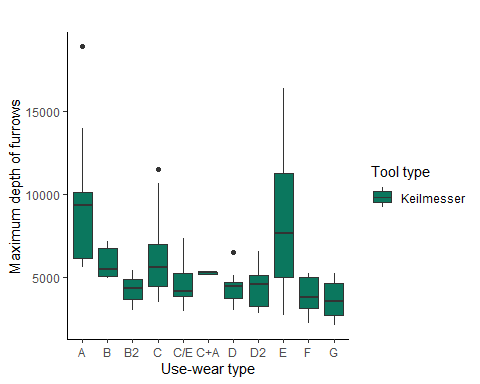
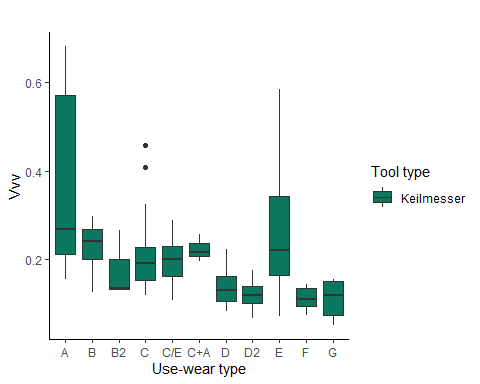
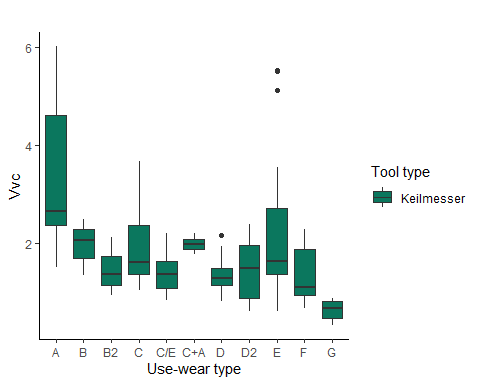
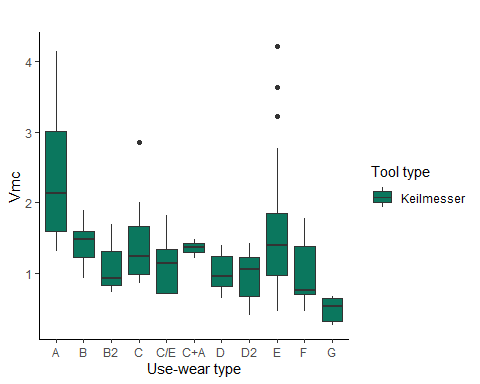
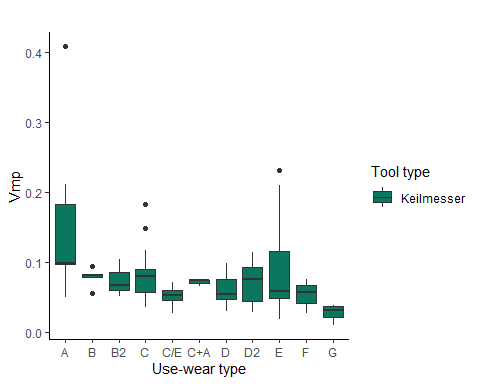
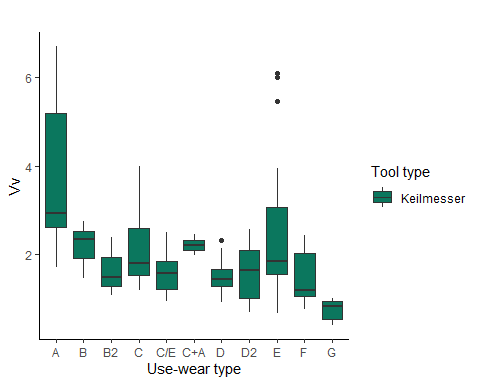
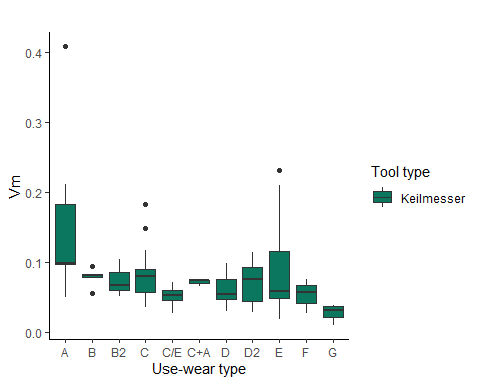
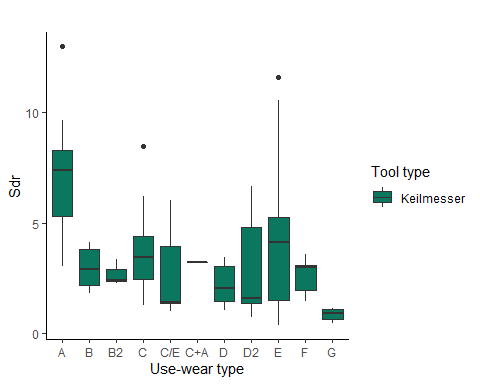
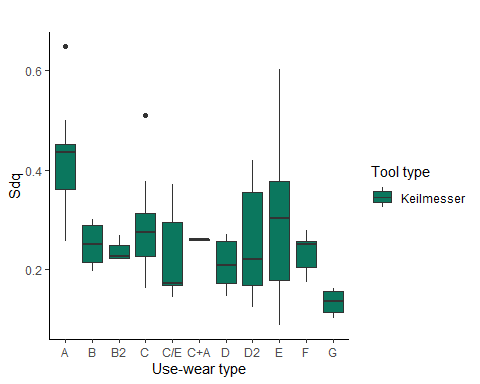
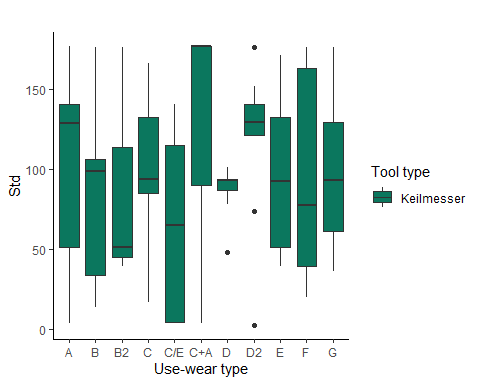
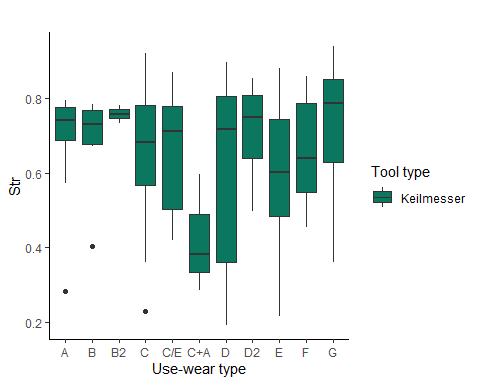


## Boxplot of all the variables combined with the use-wear type - tool types separated (without outliers)

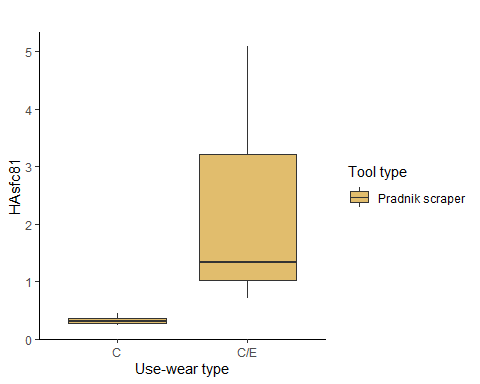
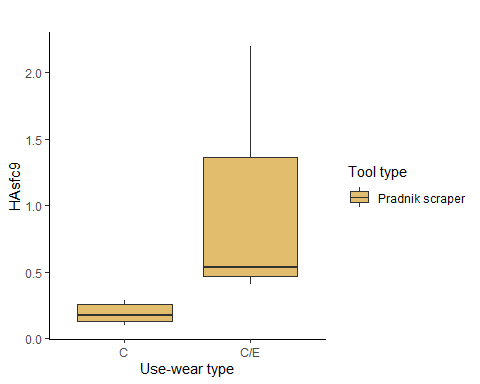
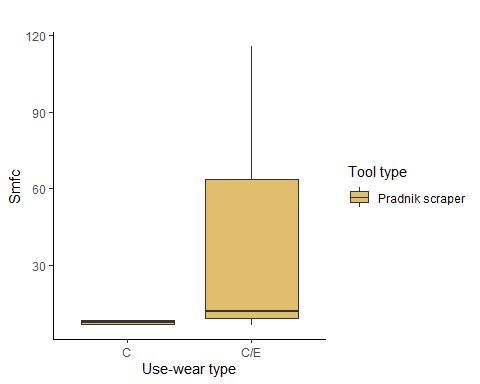
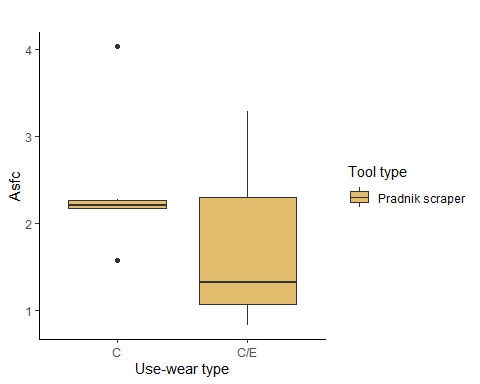
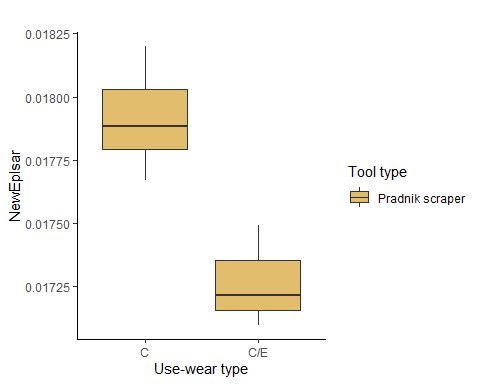
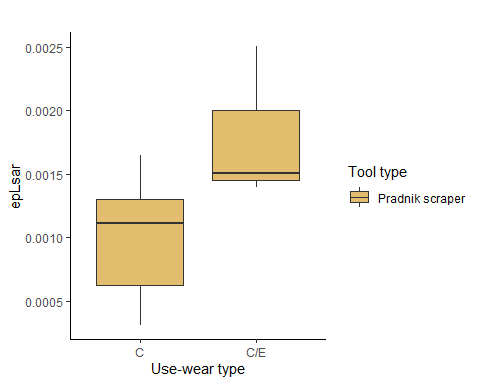
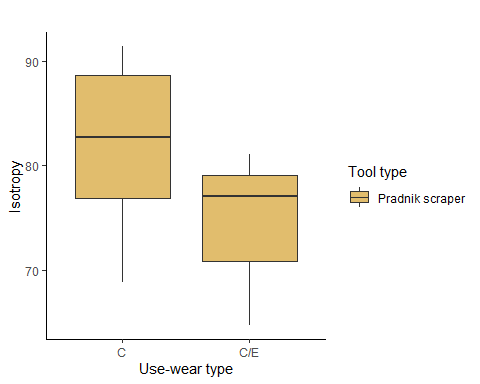
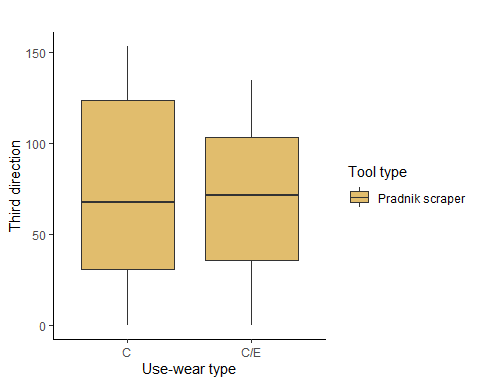
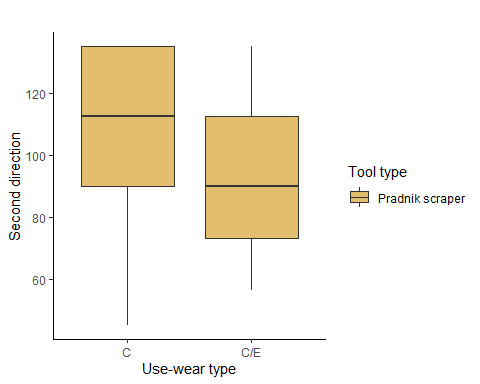
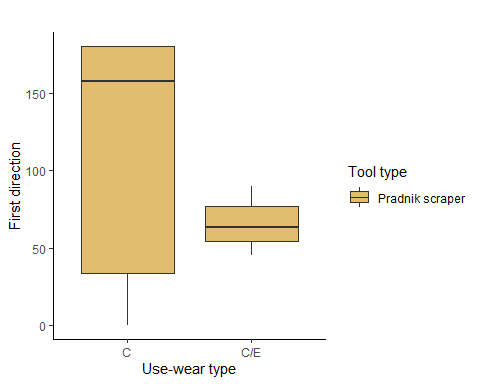
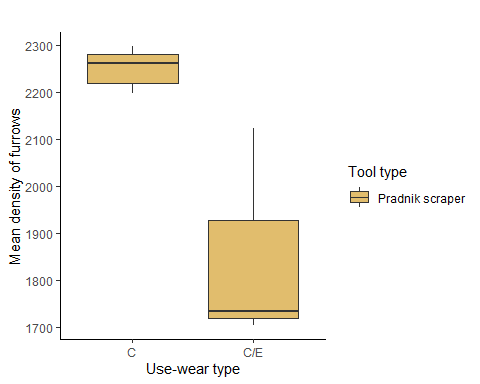
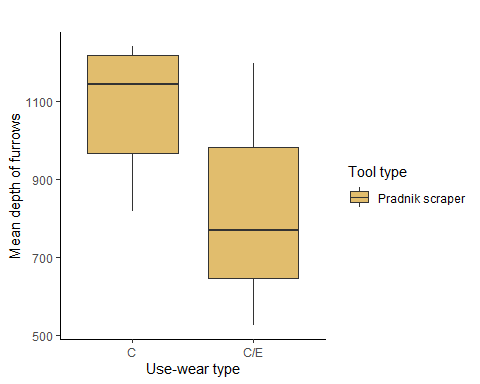
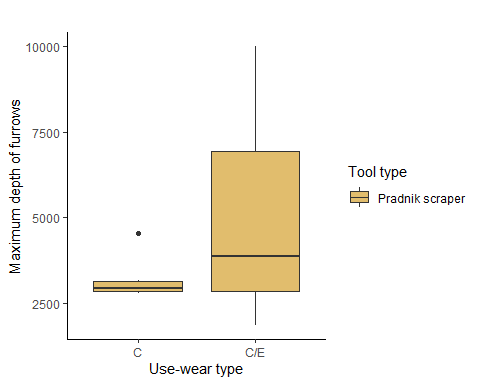
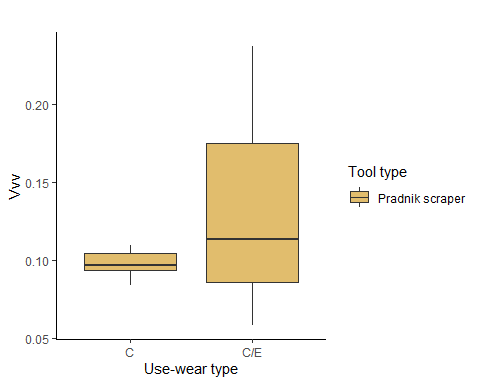
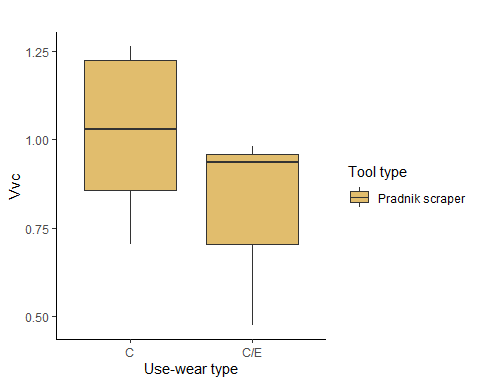
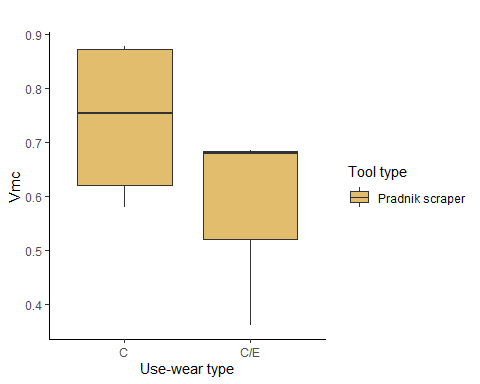
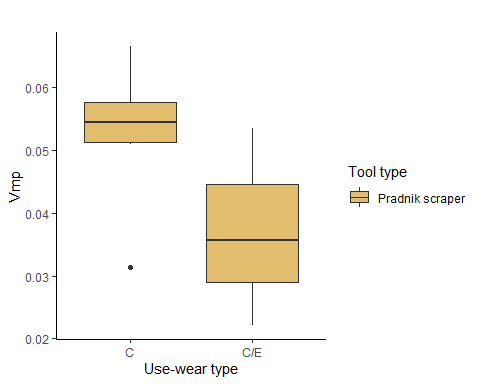
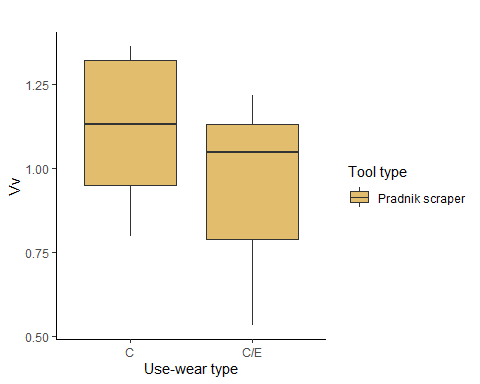
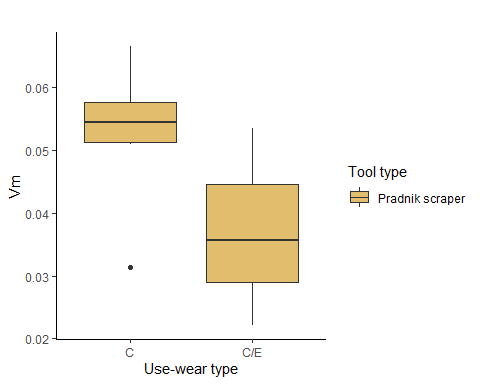
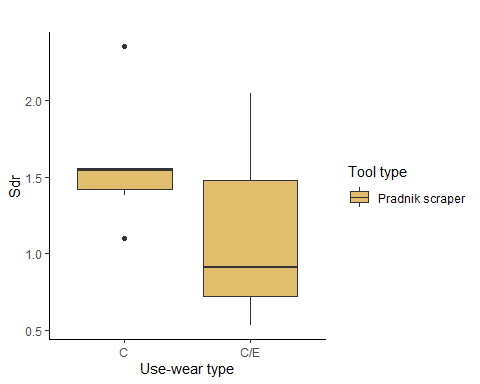
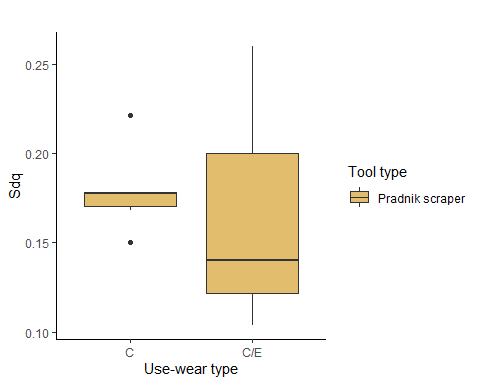
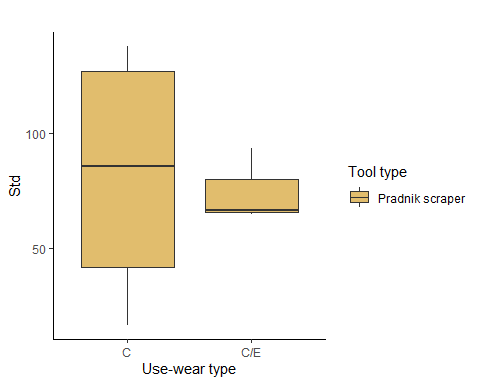
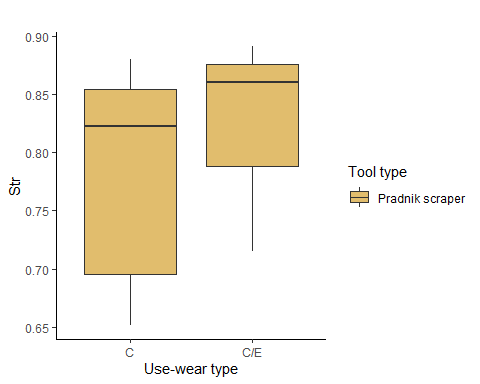
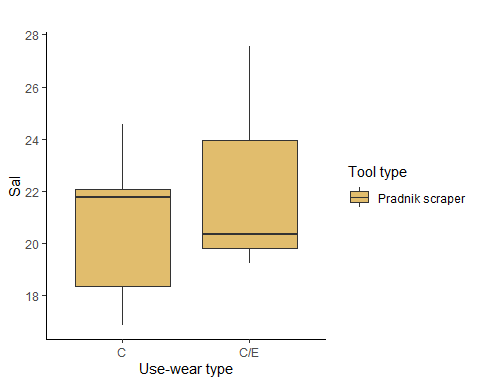
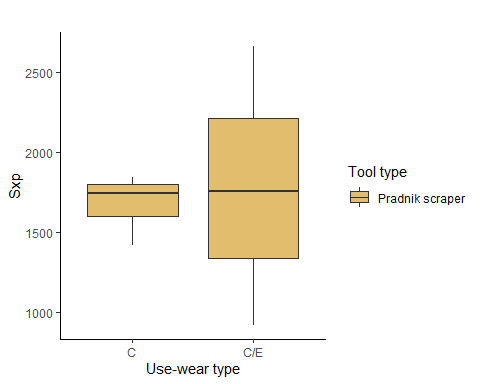
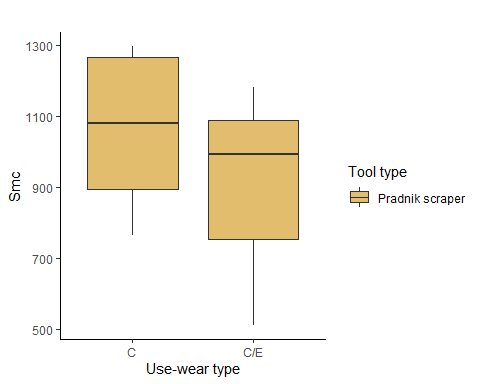
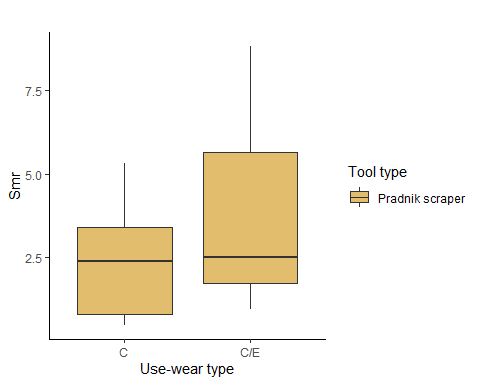
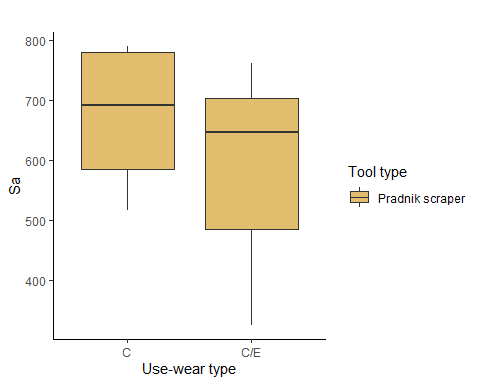
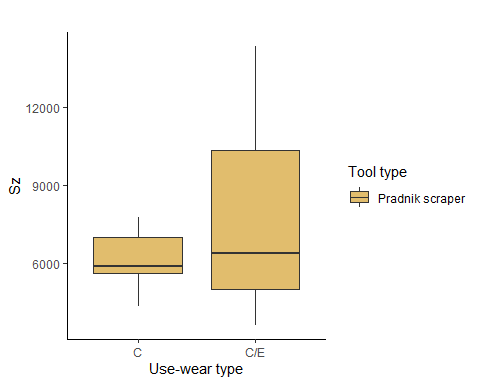
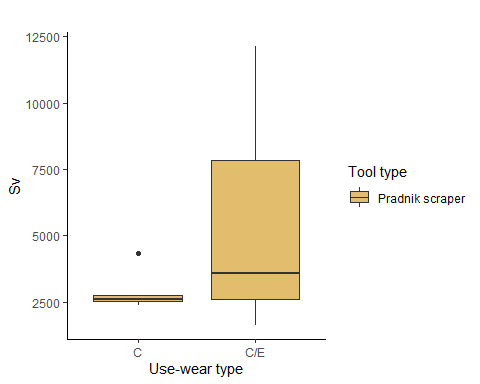
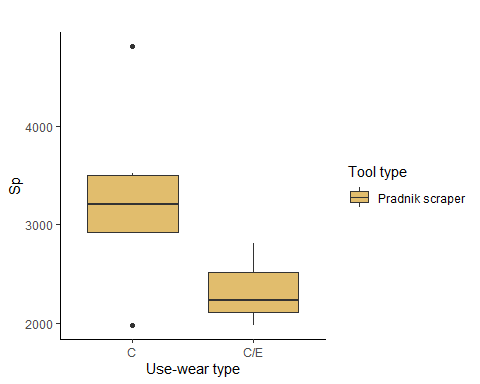
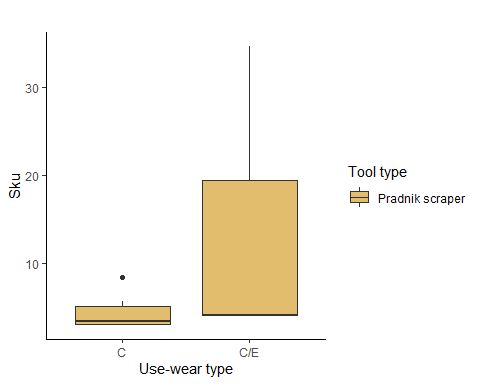
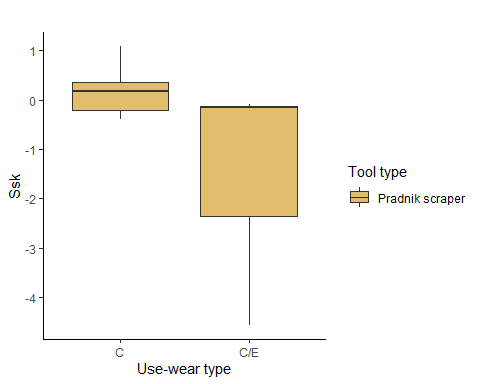
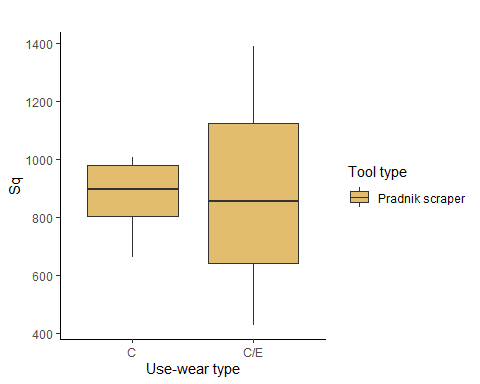
# Keilmesser  
# sorts the data according to the technological class   
sort\_data <- imp\_data2[ , ] %>% arrange(Tool.type)  
# adds indices as row names   
row.names(sort\_data) <- 1:nrow(sort\_data)  
# excludes all other tool types   
KM\_data <- sort\_data [1:107, ]  
  
for (i in num.var){  
   
 KM <- ggplot(data = KM\_data, aes\_string(x = "Usewear.type", y = names(KM\_data)[i],  
 fill = "Tool.type")) +  
 geom\_boxplot() +  
 theme\_classic() +  
 labs( x = "Use-wear type", title = " ") +  
 labs(y = gsub("\\.", " ", names(KM\_data)[i])) +  
 labs(fill = "Tool type") +  
 scale\_fill\_manual(values = "#0B775E")  
   
 print(KM)  
  
 # saves the plots   
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_KM\_boxplot",   
 names(KM\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = KM, path = dir\_out, device = "pdf", width = 17,  
 height = 25, units = "cm")  
   
}



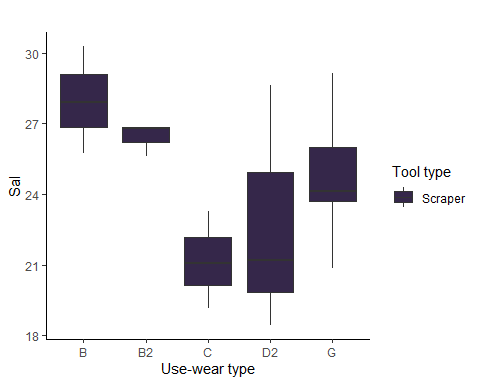
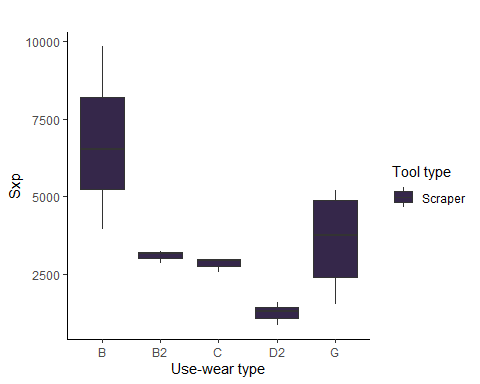
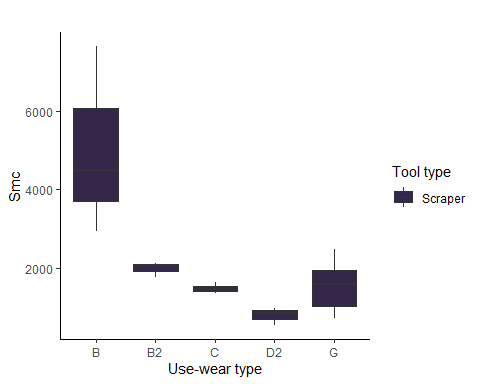
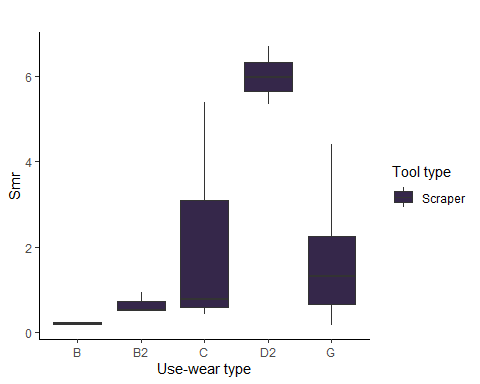
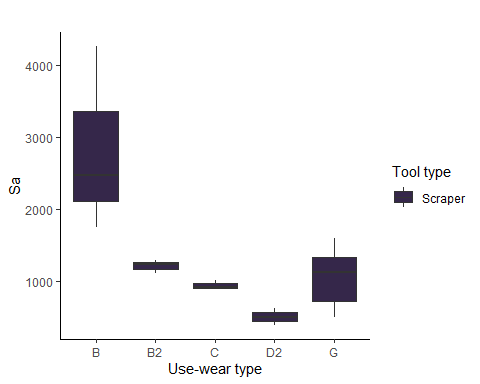
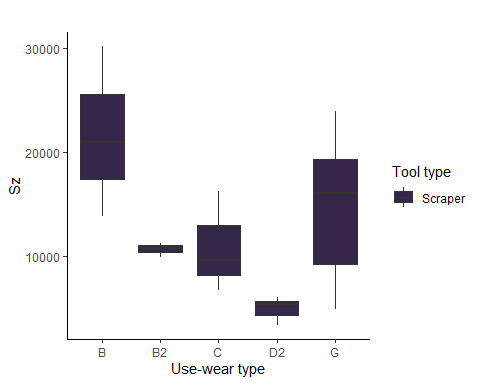
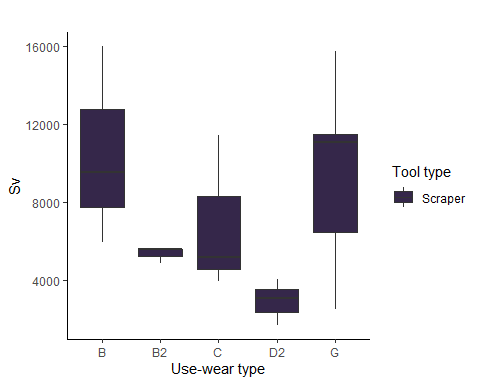
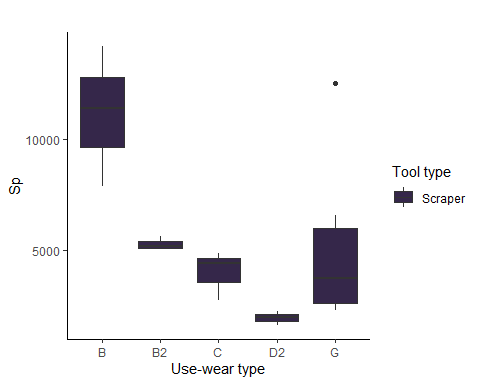
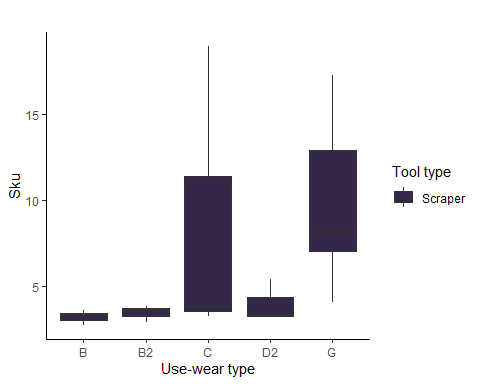
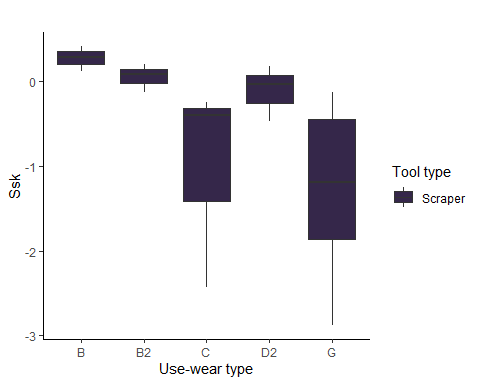
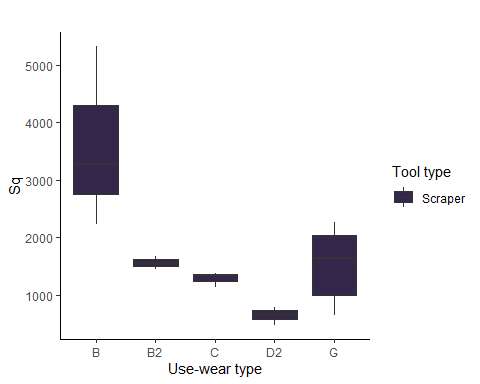
Warning: Removed 6 rows containing non-finite values (stat\_boxplot).  
  
Warning: Removed 6 rows containing non-finite values (stat\_boxplot).



# Pradnik scraper   
# excludes all other tool types   
PS\_data <- sort\_data [108:116, ]  
  
for (i in num.var){  
   
 PS <- ggplot(data = PS\_data, aes\_string(x = "Usewear.type", y = names(PS\_data)[i],  
 fill = "Tool.type")) +  
 geom\_boxplot() +  
 theme\_classic() +  
 labs( x = "Use-wear type", title = " ") +  
 labs(y = gsub("\\.", " ", names(PS\_data)[i])) +  
 labs(fill = "Tool type") +  
 scale\_fill\_manual(values = "#E1BD6D")  
   
 print(PS)  
  
 # saves the plots   
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_PS\_boxplot",   
 names(PS\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = PS, path = dir\_out, device = "pdf", width = 17,  
 height = 25, units = "cm")  
   
}

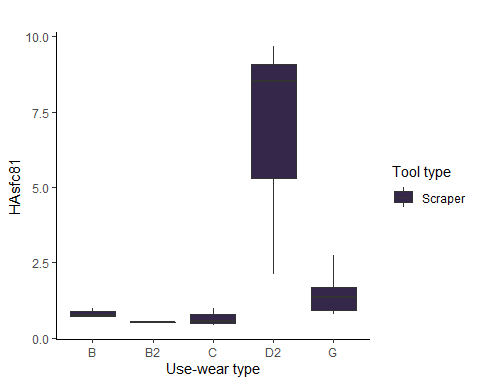
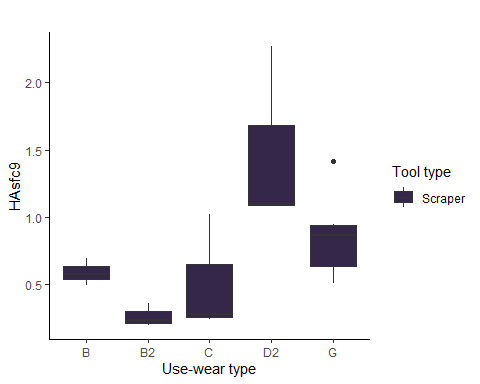
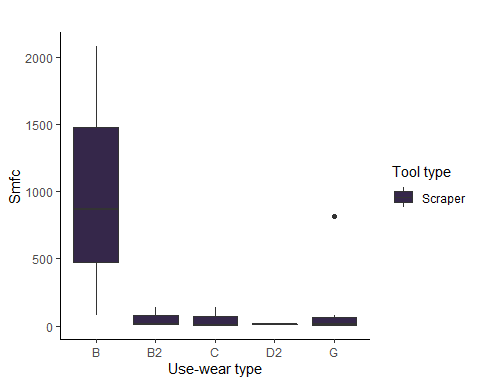
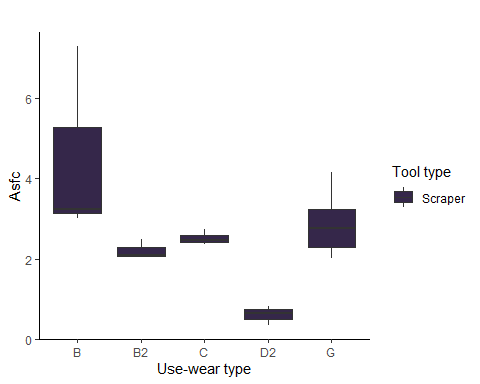
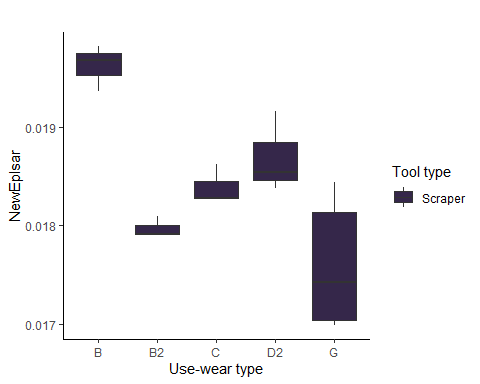
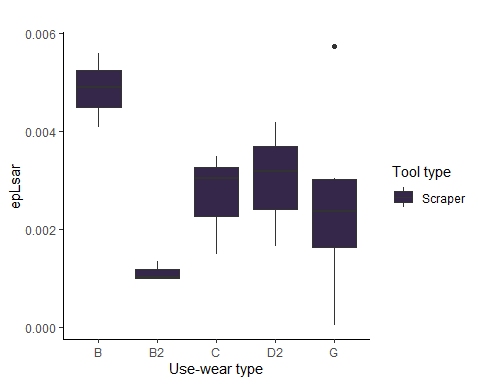
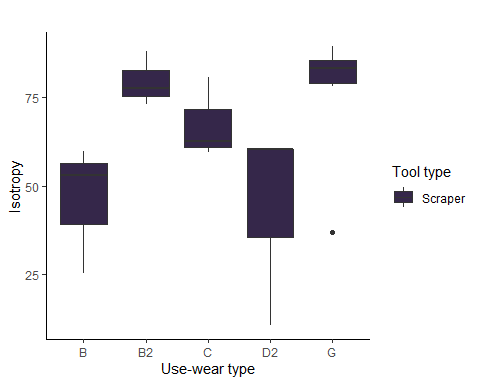
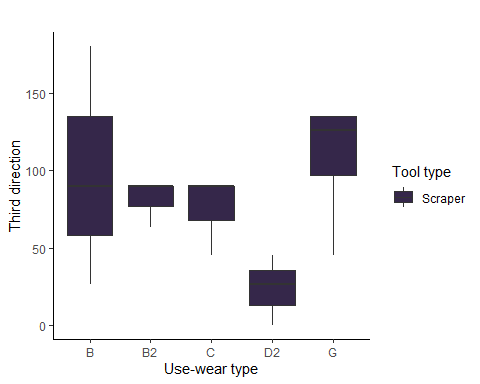
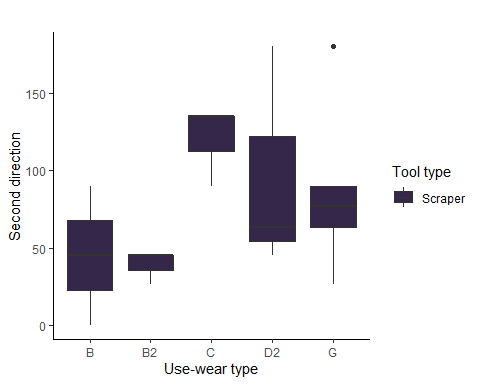
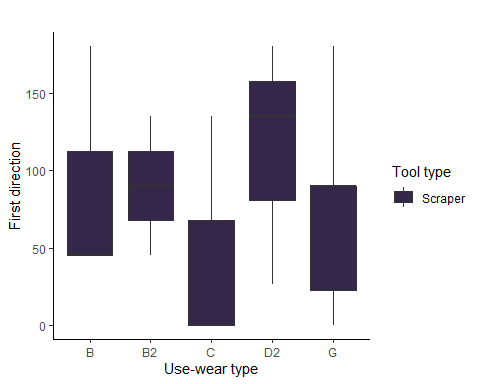
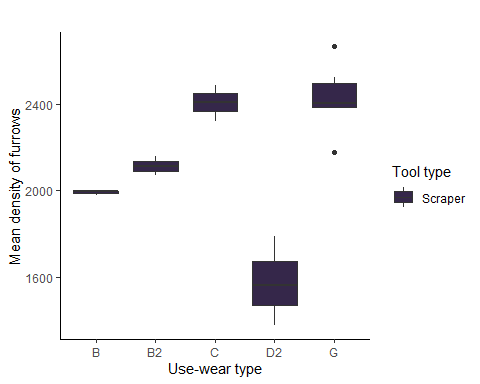
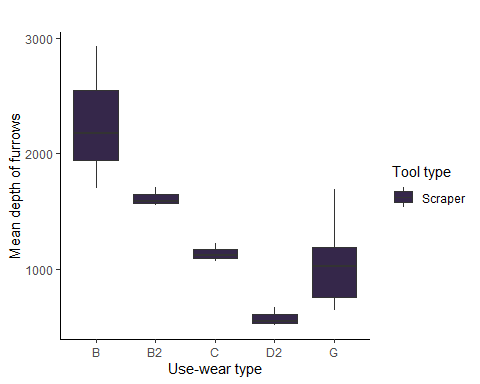
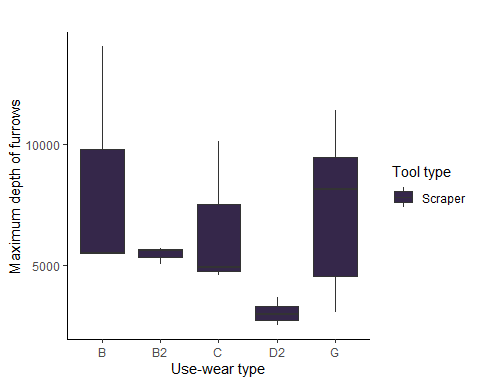
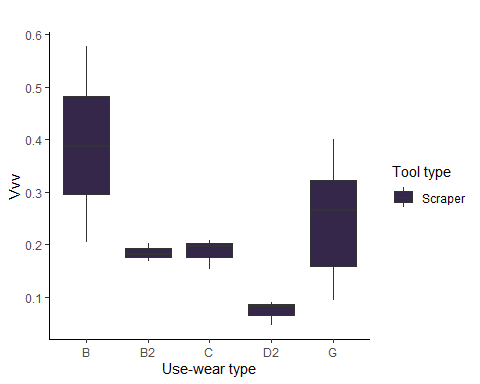
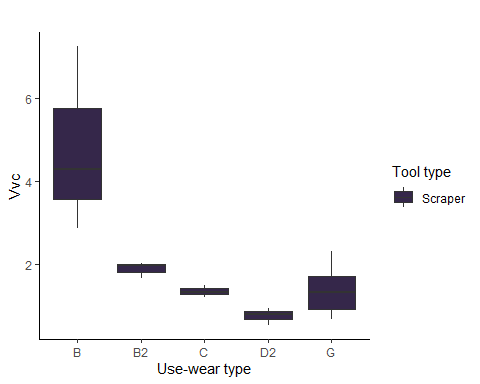
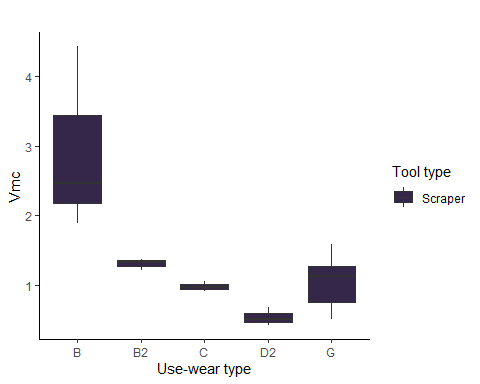
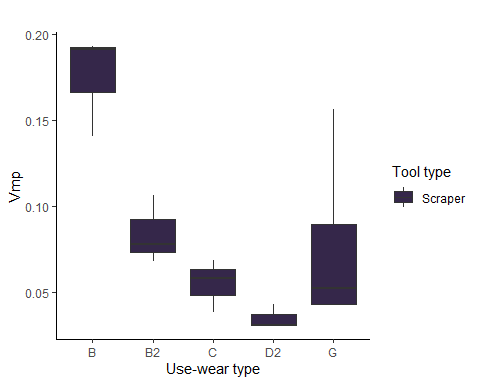
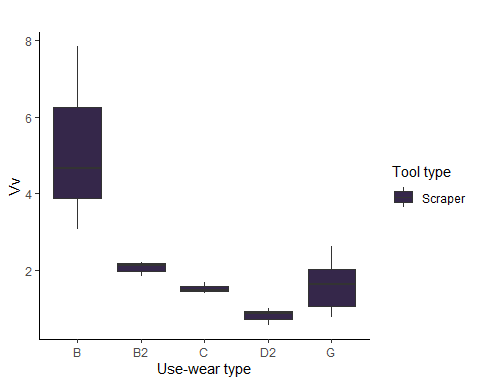
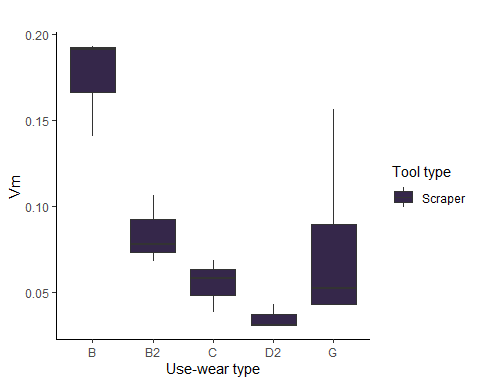
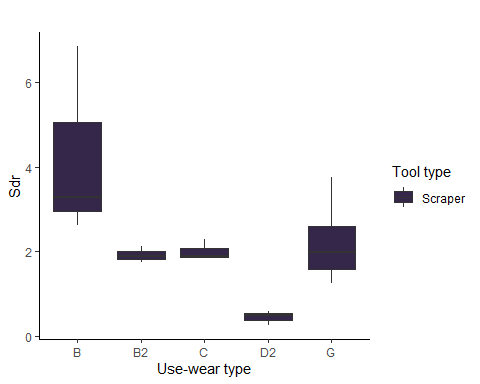
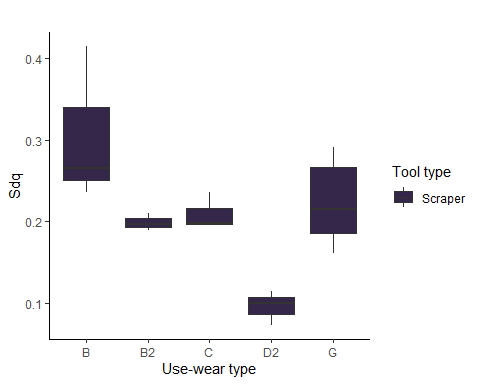
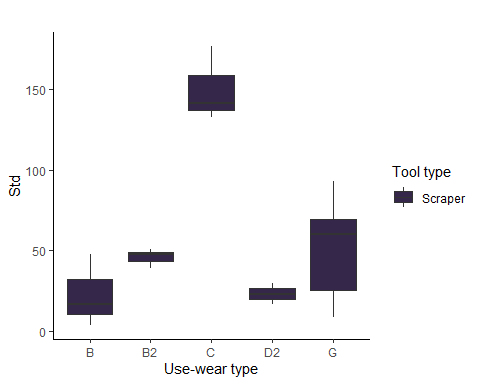
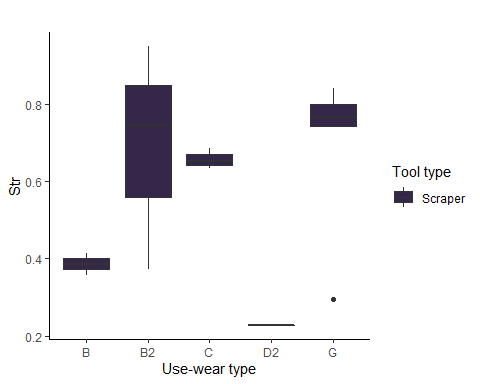


# Scraper   
# excludes all other tool types   
S\_data <- sort\_data [129:146, ]  
  
for (i in num.var){  
   
 S <- ggplot(data = S\_data, aes\_string(x = "Usewear.type", y = names(S\_data)[i],  
 fill = "Tool.type")) +  
 geom\_boxplot() +  
 theme\_classic() +  
 labs( x = "Use-wear type", title = " ") +  
 labs(y = gsub("\\.", " ", names(S\_data)[i])) +  
 labs(fill = "Tool type") +  
 scale\_fill\_manual(values = "#35274A")  
   
 print(S)  
  
 # saves the plots   
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_S\_boxplot",   
 names(S\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = S, path = dir\_out, device = "pdf", width = 17,  
 height = 25, units = "cm")  
   
}

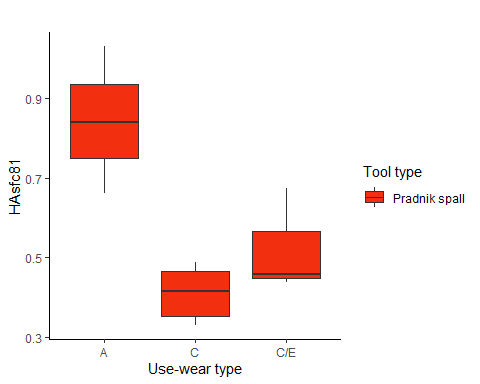
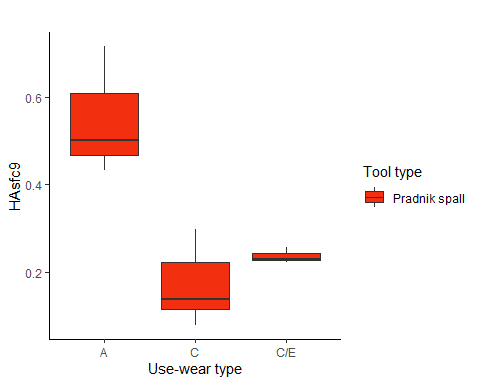
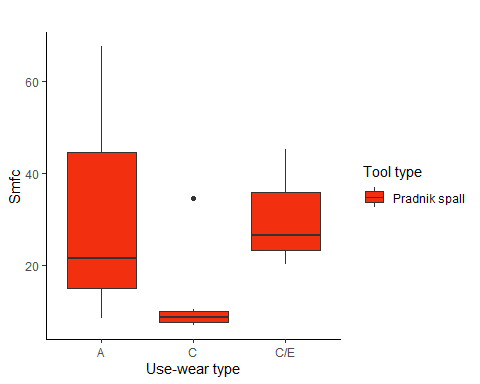
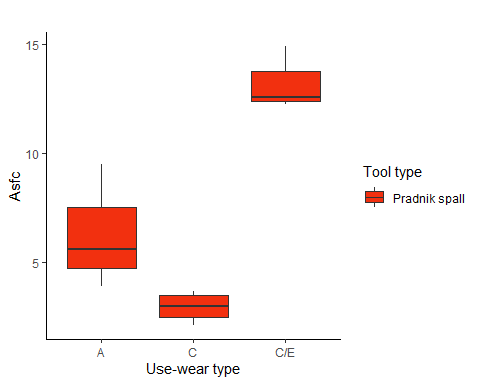
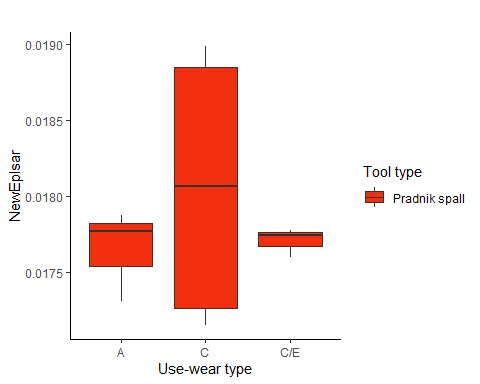
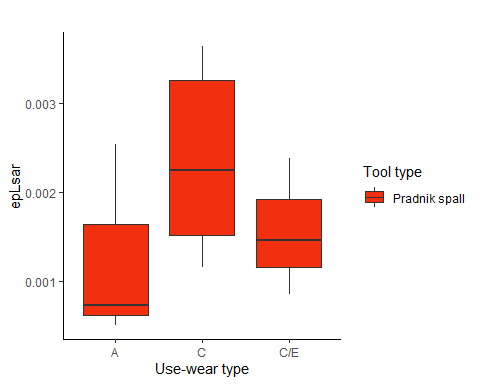
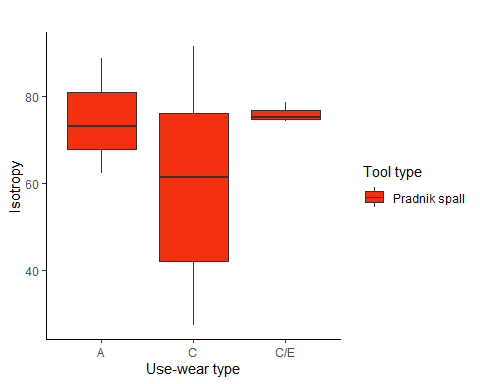
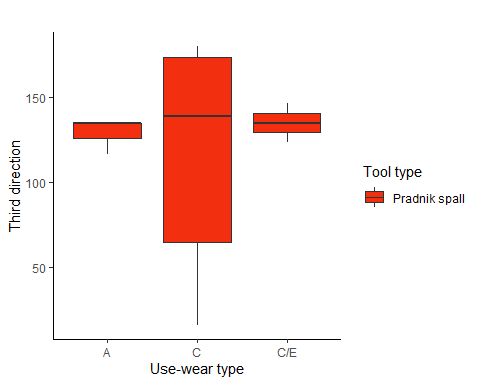
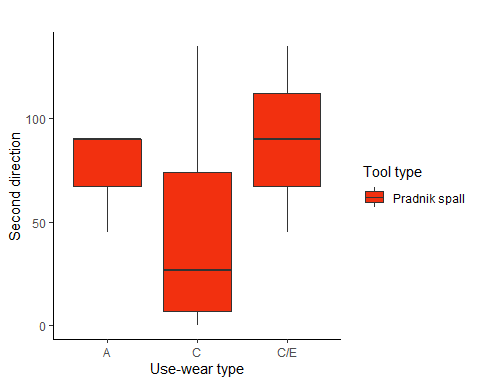
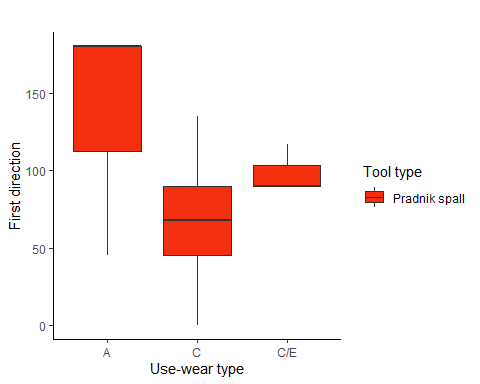
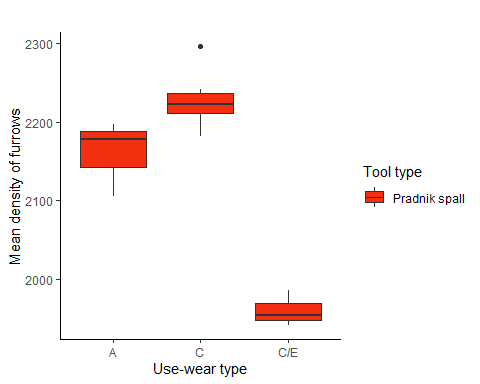
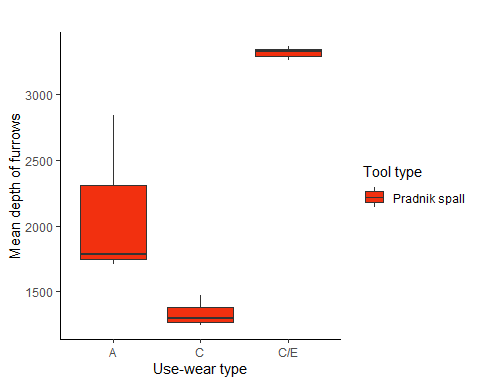
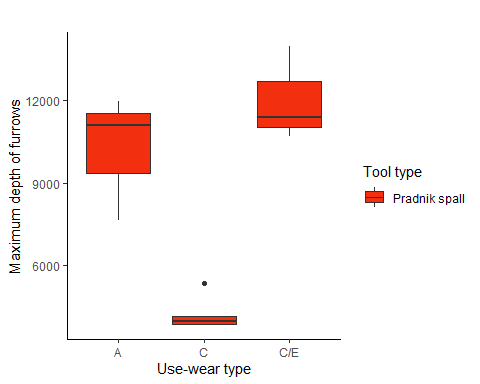
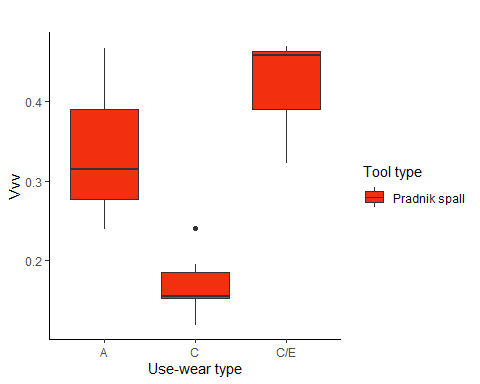
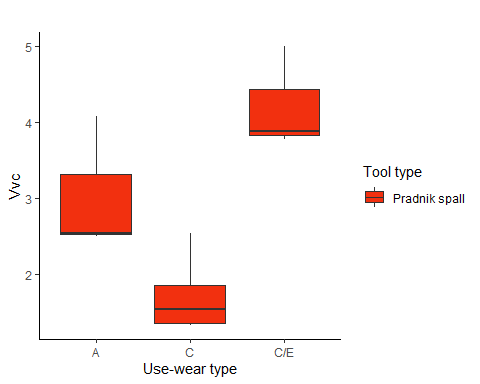
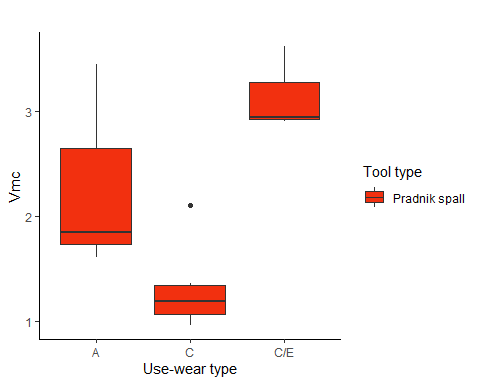
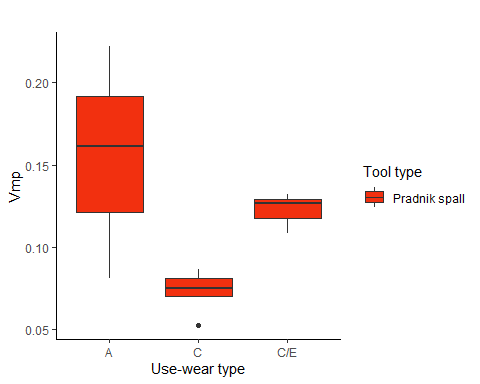
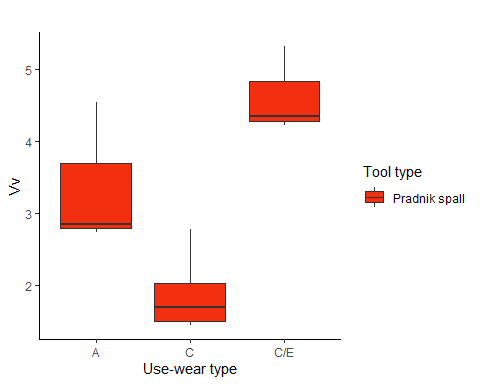
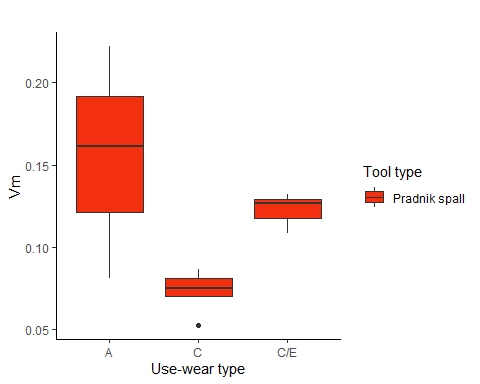
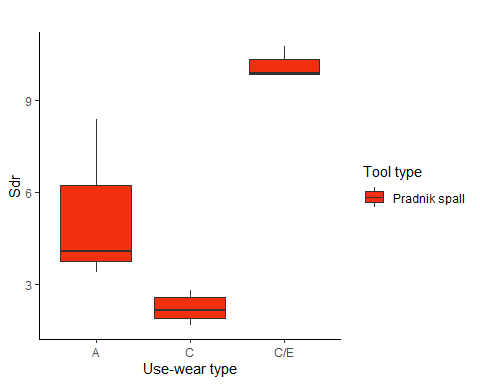
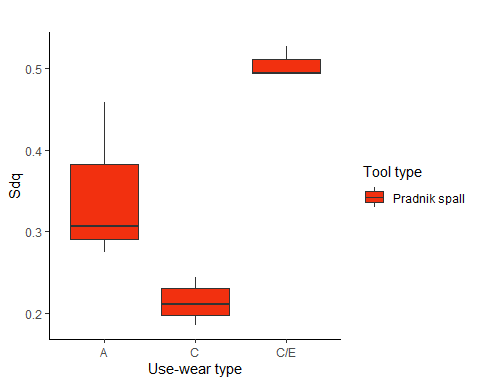
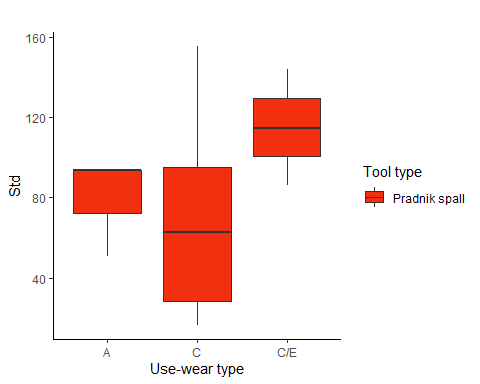
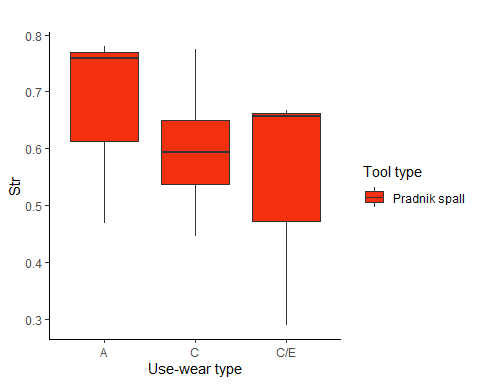
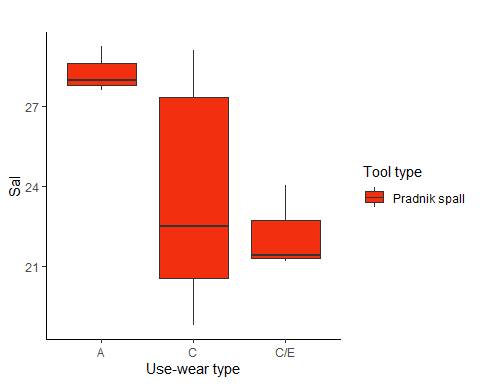
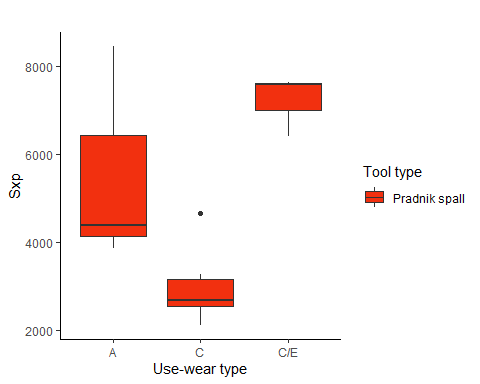
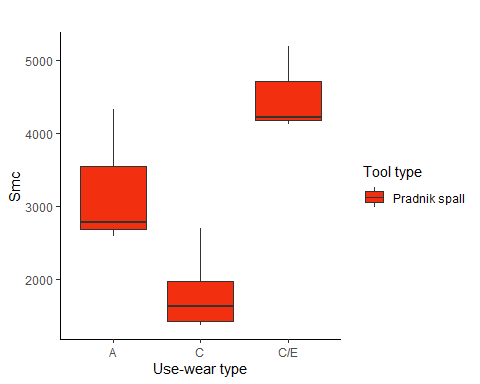
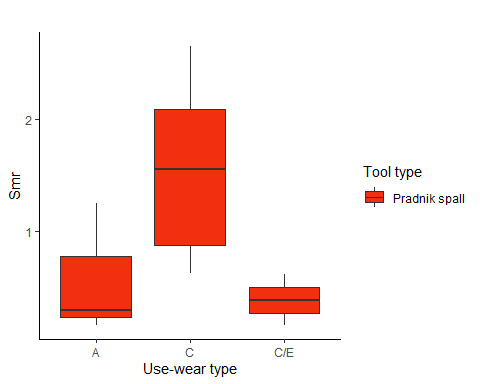
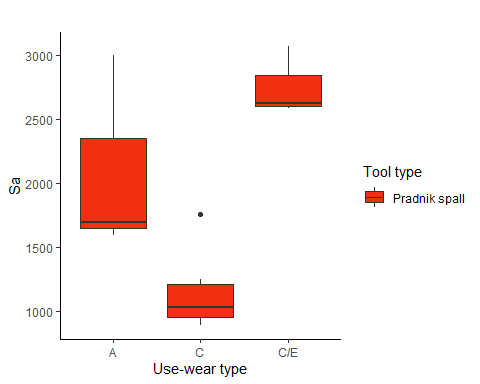
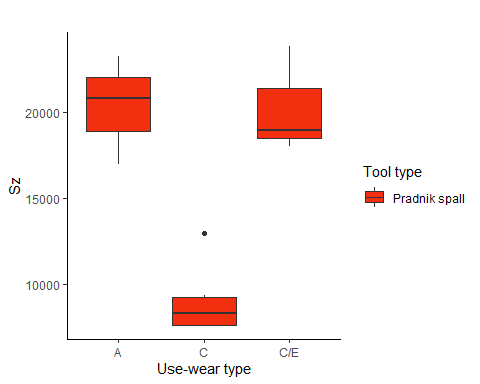
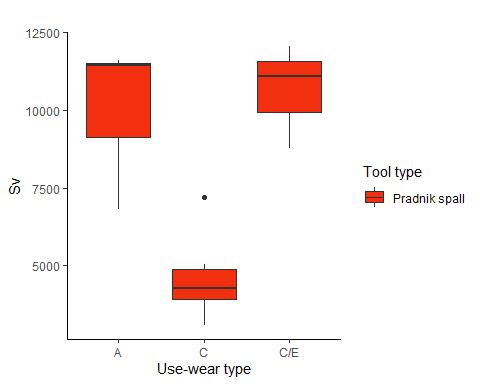
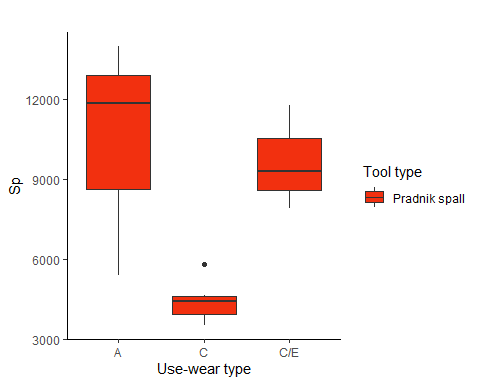
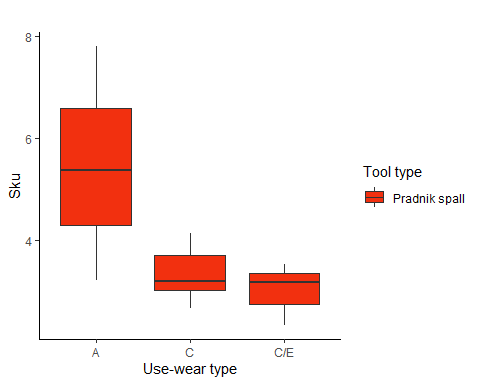
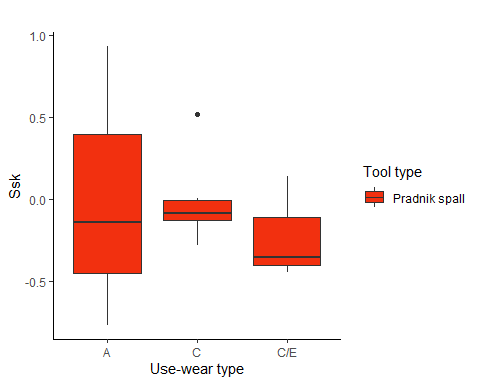
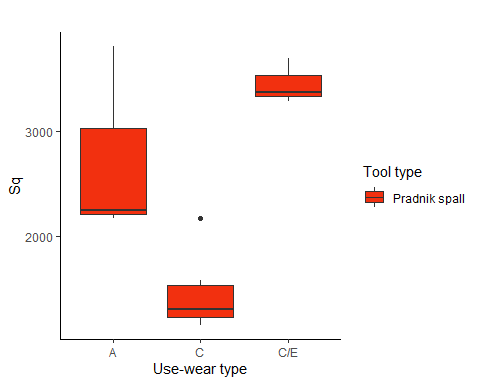


Warning: Removed 3 rows containing non-finite values (stat\_boxplot).

Warning: Removed 3 rows containing non-finite values (stat\_boxplot).

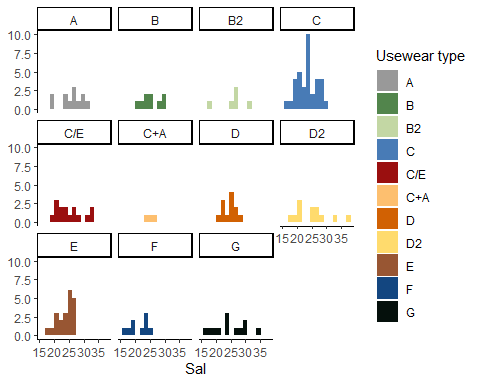
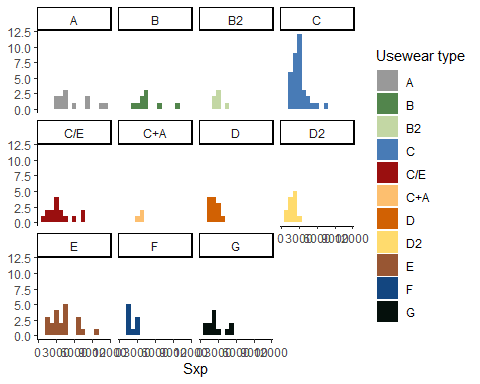
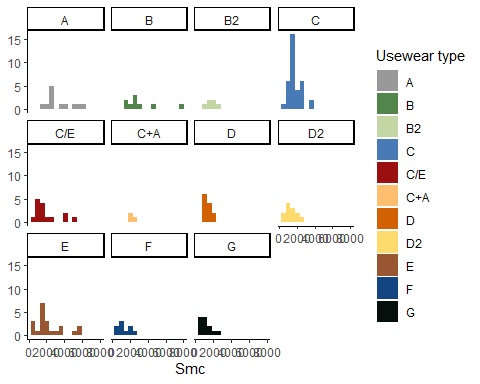
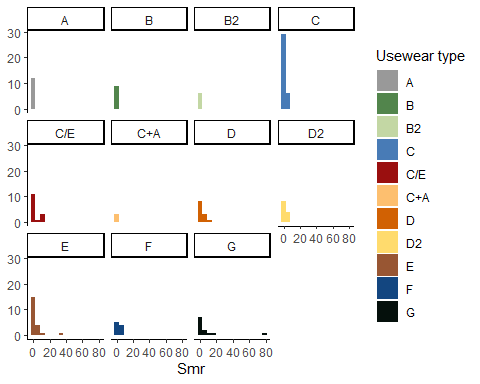
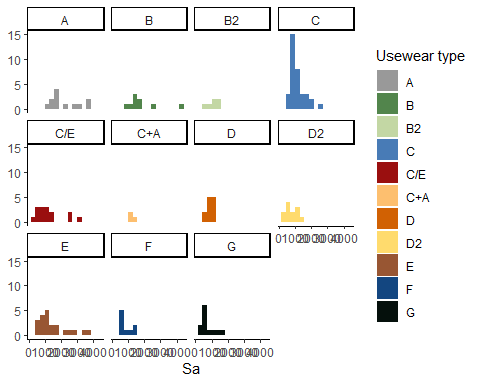
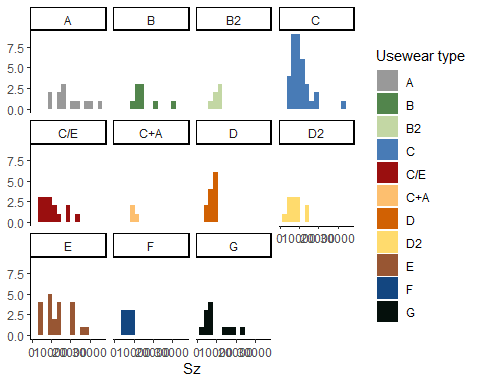
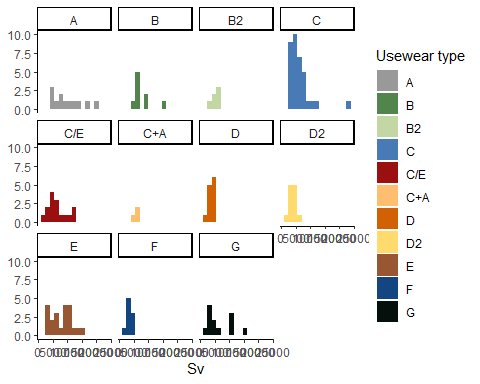
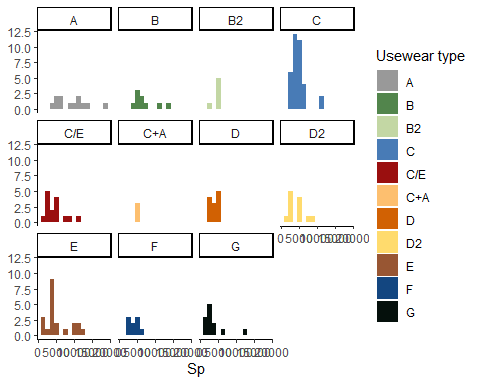
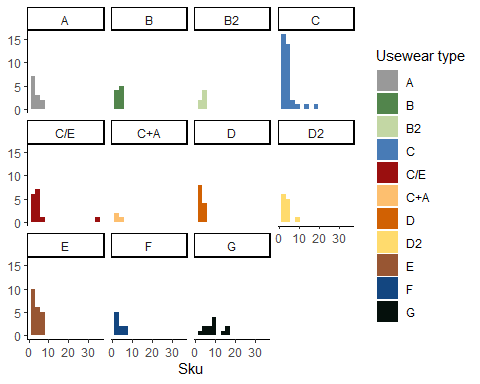
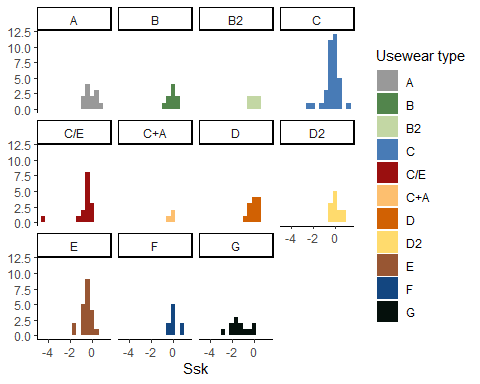
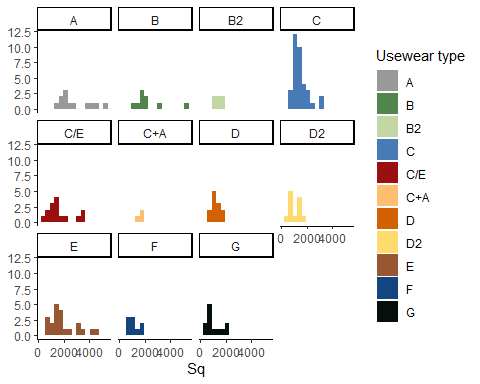


# Pradnik spall   
# excludes all other tool types   
LSS\_data <- sort\_data [117:128, ]  
  
for (i in num.var){  
   
 LSS <- ggplot(data = LSS\_data, aes\_string(x = "Usewear.type", y = names(LSS\_data)[i],  
 fill = "Tool.type")) +  
 geom\_boxplot() +  
 theme\_classic() +  
 labs( x = "Use-wear type", title = " ") +  
 labs(y = gsub("\\.", " ", names(LSS\_data)[i])) +  
 labs(fill = "Tool type") +  
 scale\_fill\_manual(values = "#F2300F")  
   
 print(LSS)  
  
 # saves the plots   
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_LSS\_boxplot",   
 names(LSS\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = LSS, path = dir\_out, device = "pdf", width = 17,  
 height = 25, units = "cm")  
   
}

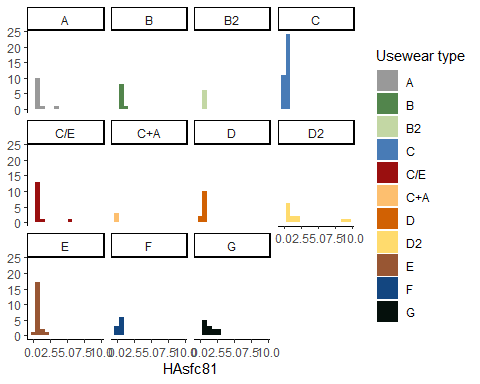
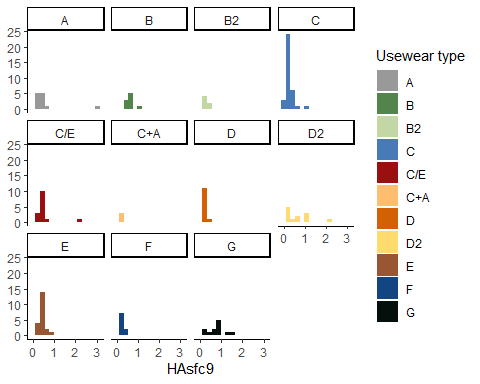
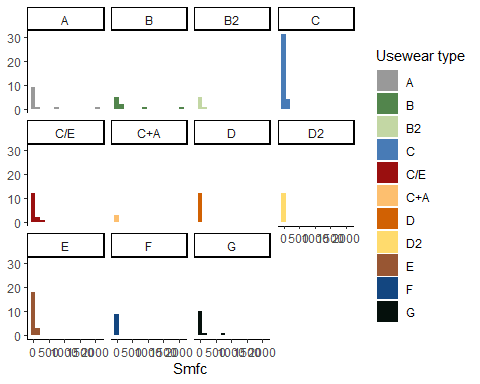
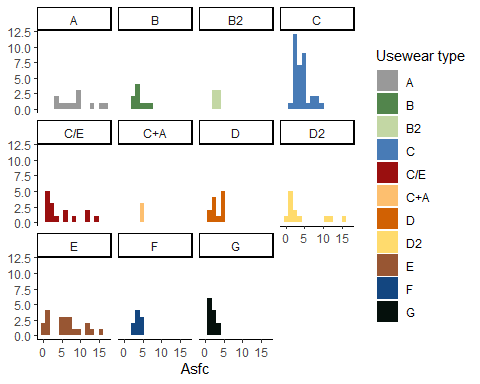
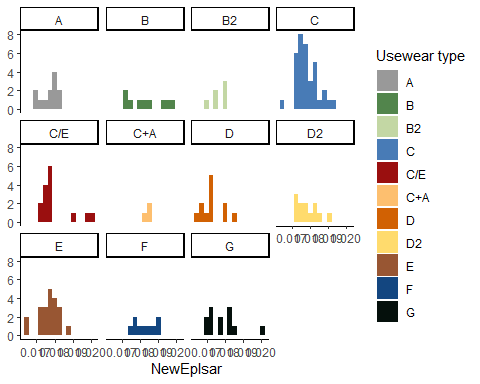
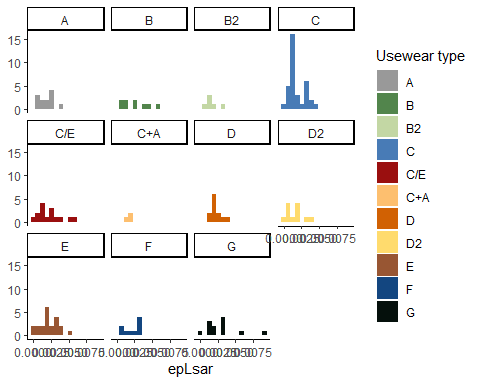
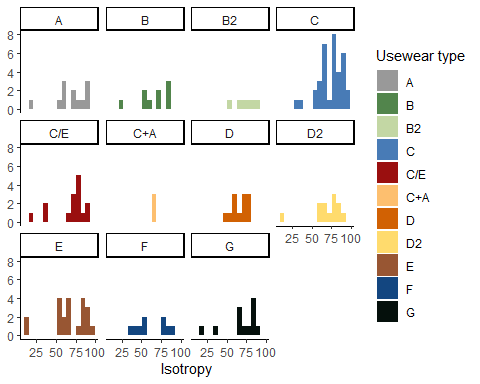
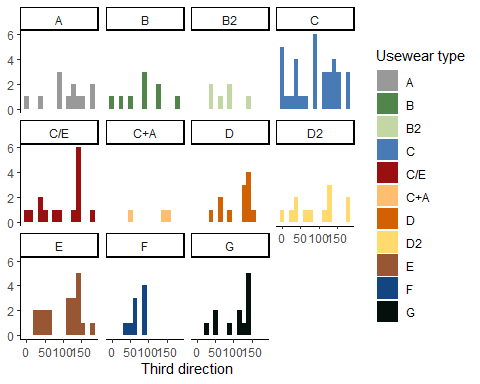
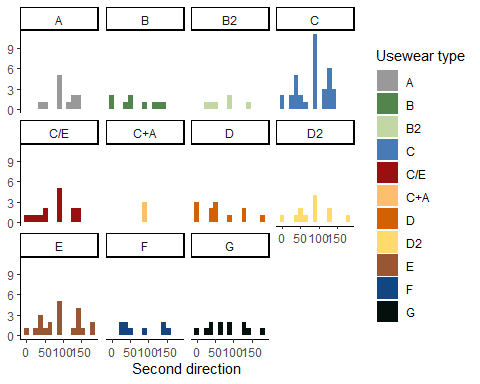
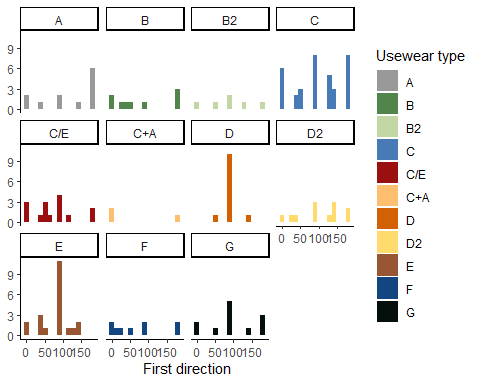
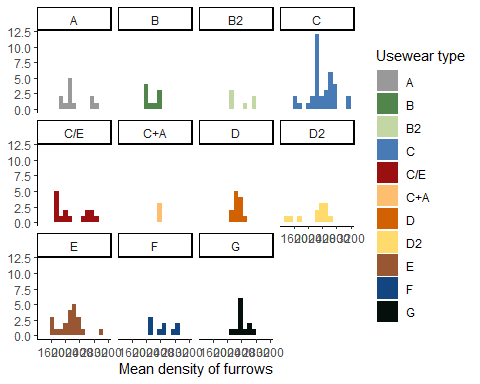
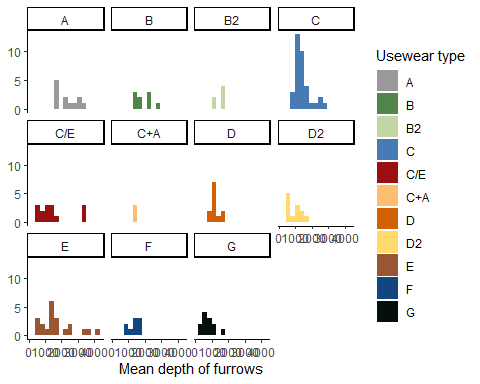
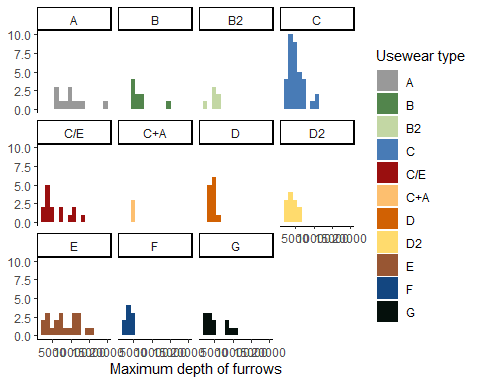
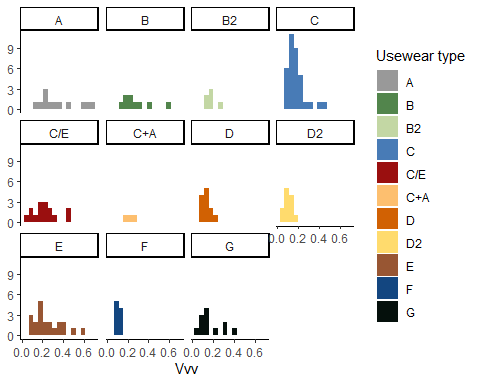
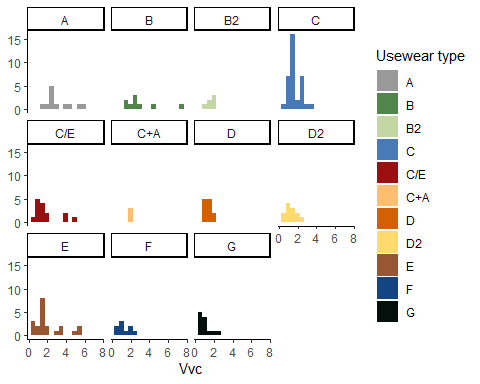
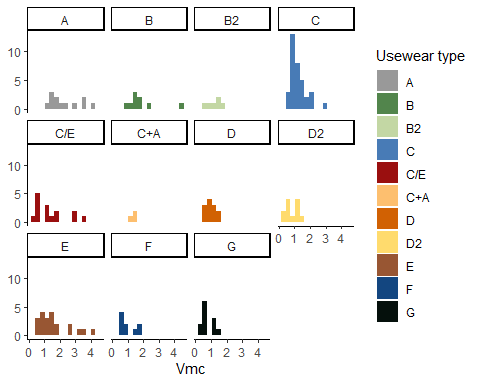
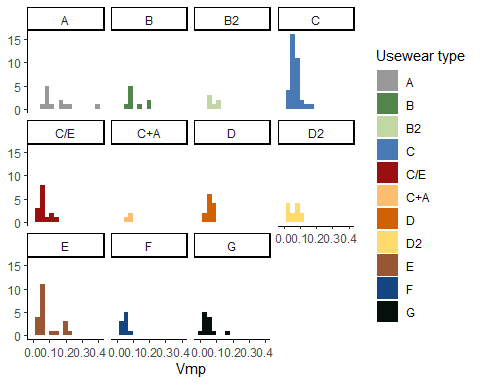
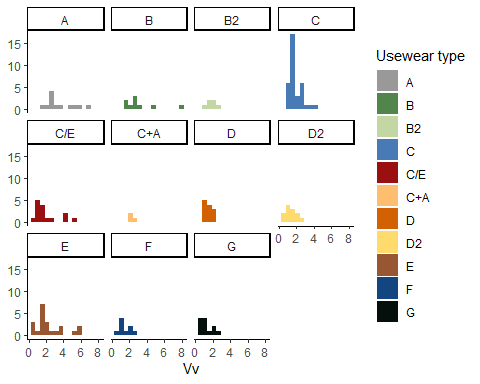
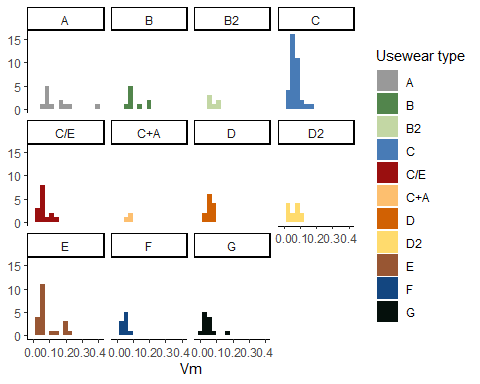
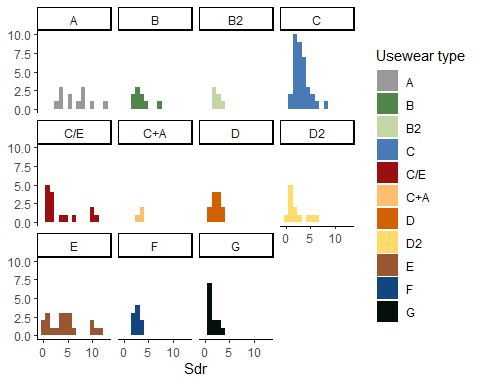
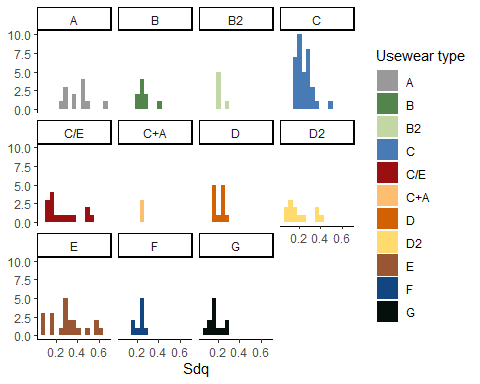
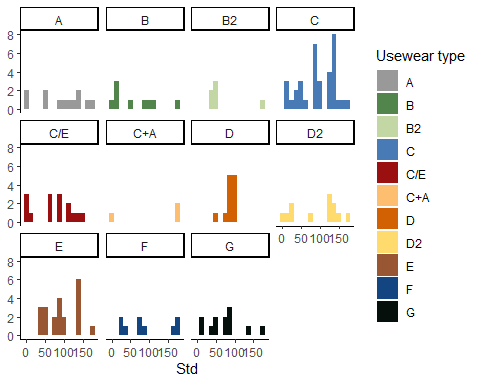
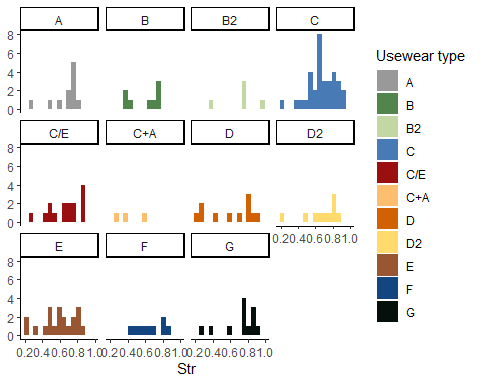


## Histogram of the use-wear types (without outliers)

custom.col <- data.frame(type = levels(imp\_data$Usewear.typ),   
 col = c("#999999", "#52854c", "#c3d7a4", "#487bb6", "#9a0f0f",  
 "#fdbf6f",  
 "#d16103", "#ffdb6d", "#985633", "#134680", "#05100c"))   
  
col <- custom.col[custom.col$type %in% levels(imp\_data[["Usewear.type"]]), "col"]  
  
  
  
# plot  
for (i in num.var){  
   
   
 p\_use <- ggplot(data = imp\_data2, aes\_string(x = names(imp\_data)[i])) +  
 geom\_histogram(bins = 15, aes(fill = Usewear.type)) +  
 theme\_classic() +  
 labs(x = gsub("\\.", " ", names(imp\_data)[i])) +  
 labs(fill = "Usewear type", y = NULL) +  
 facet\_wrap(~Usewear.type)+  
 scale\_fill\_manual(values = col)  
   
 print(p\_use)  
  
 # saves the plots   
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_histogram\_UW\_Type\_",   
 names(imp\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = p\_use, path = dir\_out, device = "pdf")  
   
}

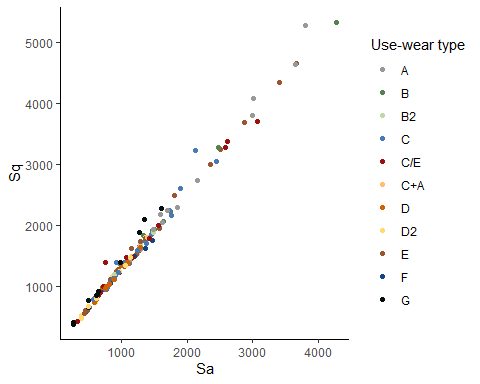


Warning: Removed 9 rows containing non-finite values (stat\_bin).  
  
Warning: Removed 9 rows containing non-finite values (stat\_bin).

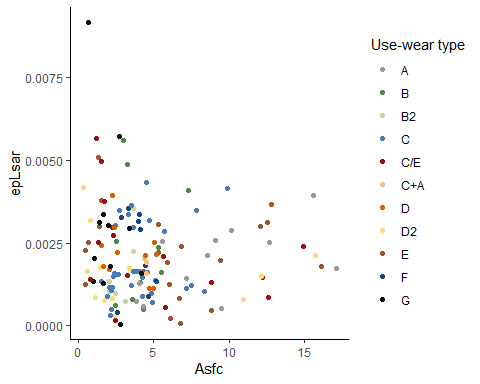


## Scatterplots of selected variables combined with the use-wear type (without outliers)

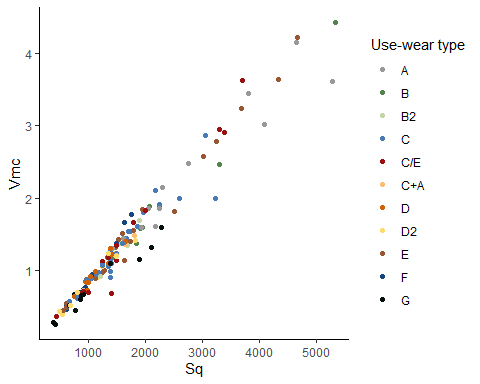
custom.col <- data.frame(type = levels(imp\_data$Usewear.typ),   
 col = c("#999999", "#52854c", "#c3d7a4", "#487bb6", "#9a0f0f",  
 "#fdbf6f",  
 "#d16103", "#ffdb6d", "#985633", "#134680", "#05100c"))   
  
col <- custom.col[custom.col$type %in% levels(imp\_data[["Usewear.type"]]), "col"]  
  
# plot   
# plots Sa against Sq  
p3 <- ggplot(data = imp\_data2) +   
 geom\_point(mapping = aes(x = Sa, y = Sq, colour = Usewear.type)) +  
 theme\_classic() +  
 labs(colour = "Use-wear type") +  
 scale\_colour\_manual(values = col)  
   
print(p3)



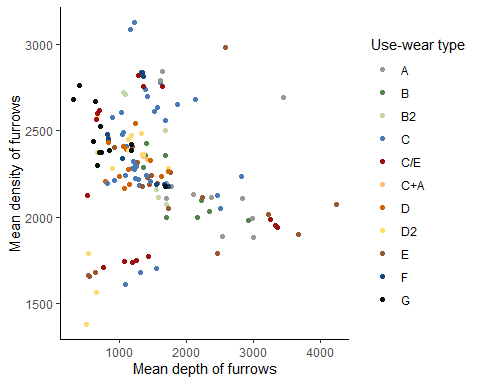
# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_SA\_scatterplot\_", ".pdf")  
ggsave(filename = file\_out, plot = p3, path = dir\_out, device = "pdf")  
  
   
# plots epLsar against Asfc   
p4 <- ggplot(data = imp\_data2) +   
 geom\_point(mapping = aes(x = Asfc, y = epLsar, colour = Usewear.type)) +  
 theme\_classic() +  
 labs(colour = "Use-wear type") +  
 scale\_colour\_manual(values = col)  
   
print(p4)



# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_Asfc\_scatterplot\_", ".pdf")  
ggsave(filename = file\_out, plot = p4, path = dir\_out, device = "pdf")  
  
   
# plots Sq against Vmc   
p5 <- ggplot(data = imp\_data2) +   
 geom\_point(mapping = aes(x = Sq, y = Vmc, colour = Usewear.type)) +  
 theme\_classic() +  
 labs(colour = "Use-wear type") +  
 scale\_colour\_manual(values = col)  
   
print(p5)



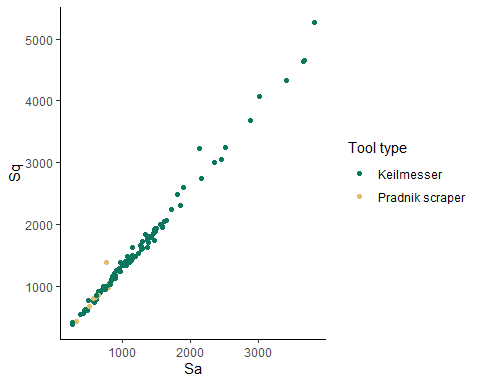
# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_Sq\_scatterplot\_", ".pdf")  
ggsave(filename = file\_out, plot = p5, path = dir\_out, device = "pdf")  
  
  
# plots Mean depth of furrows against mean density of furrows   
p6 <- ggplot(data = imp\_data2) +   
 geom\_point(mapping = aes(x = Mean.depth.of.furrows, y = Mean.density.of.furrows,  
 colour = Usewear.type)) +  
 theme\_classic() +  
 labs(colour = "Use-wear type") +  
 scale\_colour\_manual(values = col) +  
 labs(x = "Mean depth of furrows", y = "Mean density of furrows")  
   
print(p6)



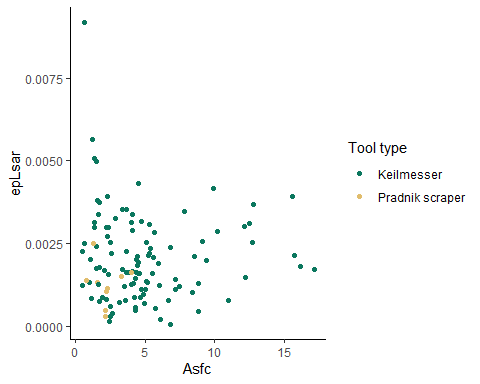
# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_furrows\_scatterplot\_", ".pdf")  
ggsave(filename = file\_out, plot = p6, path = dir\_out, device = "pdf")

## Scatterplots of selected variables combined with the two tool types: Keilmesser and Pradnik scraper (without outliers)

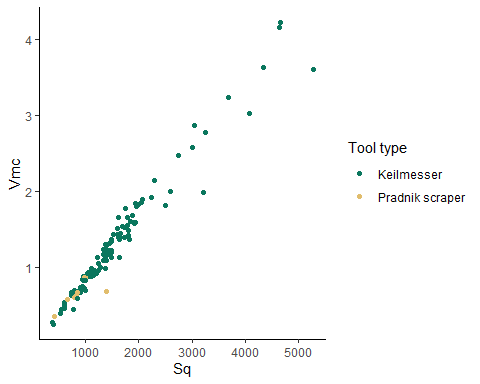
# selects only Keilmesser and Pradnik scraper  
KM\_PS <- filter(imp\_data2, Tool.type == "Keilmesser" | Tool.type == "Pradnik scraper")   
  
  
custom.col2b <- data.frame(type = unique(KM\_PS$Tool.type),   
 col = c("#0B775E", "#E1BD6D"))   
  
col2b <- custom.col2b[custom.col2b$type %in% unique(KM\_PS[["Tool.type"]]), "col"]  
  
# plot   
# plots Sa against Sq  
p7 <- ggplot(data = KM\_PS) +   
 geom\_point(mapping = aes(x = Sa, y = Sq, colour = Tool.type)) +  
 theme\_classic() +  
 labs(colour = "Tool type") +  
 scale\_colour\_manual(values = col2b)  
   
print(p7)



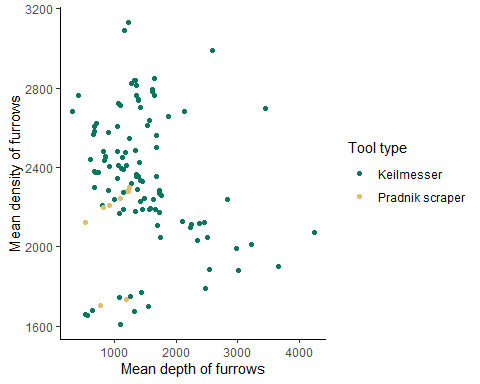
# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_SA\_KM.PS\_scatterplot\_", ".pdf")  
ggsave(filename = file\_out, plot = p7, path = dir\_out, device = "pdf")  
  
   
# plots epLsar against Asfc   
p8 <- ggplot(data = KM\_PS) +   
 geom\_point(mapping = aes(x = Asfc, y = epLsar, colour = Tool.type)) +  
 theme\_classic() +  
 labs(colour = "Tool type") +  
 scale\_colour\_manual(values = col2b)  
   
print(p8)



# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_Asfc\_KM.PS\_scatterplot\_", ".pdf")  
ggsave(filename = file\_out, plot = p8, path = dir\_out, device = "pdf")  
  
   
# plots Sq against Vmc   
p9 <- ggplot(data = KM\_PS) +   
 geom\_point(mapping = aes(x = Sq, y = Vmc, colour = Tool.type)) +  
 theme\_classic() +  
 labs(colour = "Tool type") +  
 scale\_colour\_manual(values = col2b)  
   
print(p9)



# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_Sq\_KM.PS\_scatterplot\_", ".pdf")  
ggsave(filename = file\_out, plot = p9, path = dir\_out, device = "pdf")  
  
  
# plots Mean depth of furrows against mean density of furrows   
p10 <- ggplot(data = KM\_PS) +   
 geom\_point(mapping = aes(x = Mean.depth.of.furrows, y = Mean.density.of.furrows,  
 colour = Tool.type)) +  
 theme\_classic() +  
 labs(colour = "Tool type") +  
 scale\_colour\_manual(values = col2b) +  
 labs(x = "Mean depth of furrows", y = "Mean density of furrows")  
   
print(p10)



# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_furrows\_KM.PS\_scatterplot\_",  
 ".pdf")  
ggsave(filename = file\_out, plot = p10, path = dir\_out, device = "pdf")

## Principal component analysis

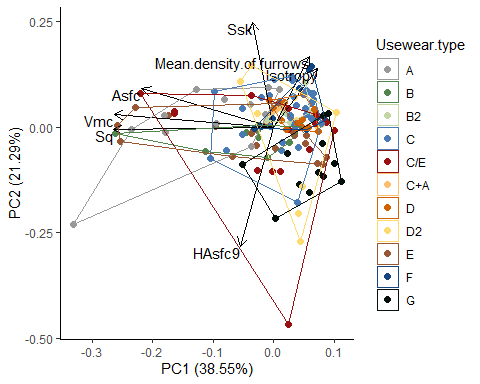
### PCA Use-wear type (without outliers)

# uses for the PCA only selected variables: Sq, SSK, Vmc, Isotropy, Mean, density of furrows, Asfc, HAsfc9   
imp\_data.pca <- prcomp(imp\_data2[, c(24:25, 42, 47, 51, 54,56)], scale. = TRUE)   
  
  
custom.col1 <- data.frame(type = levels(imp\_data$Usewear.typ),   
 col = c("#999999", "#52854c", "#c3d7a4", "#487bb6", "#9a0f0f", "#fdbf6f",  
 "#d16103", "#ffdb6d", "#985633", "#134680", "#05100c"))   
imp\_data$col <- custom.col1[imp\_data$Usewear.typ, "col"]  
  
# Using ggfortify  
a<- autoplot(imp\_data.pca, data = imp\_data2, colour = "Usewear.type", size = 2,  
 loadings = TRUE, loadings.colour = "black", loadings.label = TRUE, loadings.label.colour = "black",   
 loadings.label.size = 4, loadings.label.hjust = 1, loadings.label.vjust = 1,   
 frame = TRUE, frame.type = "convex", frame.colour = "Usewear.type",   
 frame.alpha = 0) +   
 theme\_classic() +  
 scale\_colour\_manual(values = custom.col1$col)

Warning: `select\_()` is deprecated as of dplyr 0.7.0.  
Please use `select()` instead.  
This warning is displayed once every 8 hours.  
Call `lifecycle::last\_warnings()` to see where this warning was generated.

Warning: `group\_by\_()` is deprecated as of dplyr 0.7.0.  
Please use `group\_by()` instead.  
See vignette('programming') for more help  
This warning is displayed once every 8 hours.  
Call `lifecycle::last\_warnings()` to see where this warning was generated.

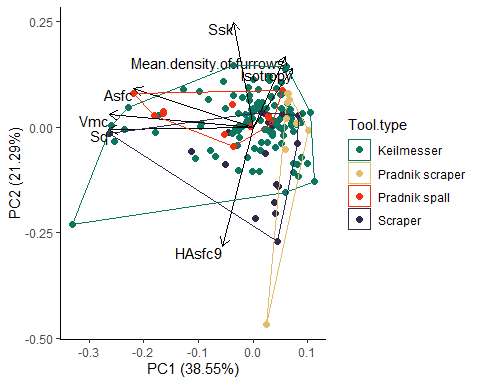
print(a)



# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_PCA\_usewear\_type", ".pdf")  
ggsave(filename = file\_out, plot = a, path = dir\_out, device = "pdf")

### PCA Tool type (without outliers)

# uses for the PCA only selected variables: Sq, SSK, Vmc, Isotropy, Mean density of furrows,  
# Asfc, HAsfc9   
imp\_data.pca <- prcomp(imp\_data2[, c(24:25, 42, 47, 51, 54,56)], scale. = TRUE)   
  
# Wes Anderson color palettes Rushmore = c("#E1BD6D", "#EABE94", "#0B775E", "#35274A",  
#"#F2300F")  
custom.col2 <- data.frame(type = levels(imp\_data$Tool.typ),   
 col = c( "#0B775E", "#E1BD6D", "#F2300F", "#35274A"))   
imp\_data$col <- custom.col2[imp\_data$Tool.typ, "col"]  
  
b <- autoplot(imp\_data.pca, data = imp\_data2, colour = "Tool.type", size = 2,  
 loadings = TRUE, loadings.colour = "black", loadings.label = TRUE,  
 loadings.label.colour = "black",   
 loadings.label.size = 4, loadings.label.hjust = 1, loadings.label.vjust = 1,   
 frame = TRUE, frame.type = "convex", frame.colour = "Tool.type",   
 frame.alpha = 0) +   
 theme\_classic() +  
 scale\_colour\_manual(values = custom.col2$col)  
  
print(b)



# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_PCA\_tool\_tpye", ".pdf")  
ggsave(filename = file\_out, plot = b, path = dir\_out, device = "pdf")

The files will be saved as “~/analysis/plots.[ext]”.

# 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] ggfortify\_0.4.11 wesanderson\_0.3.6 openxlsx\_4.2.3 ggrepel\_0.9.1   
 [5] doBy\_4.6.8 patchwork\_1.1.1 forcats\_0.5.1 stringr\_1.4.0   
 [9] dplyr\_1.0.3 purrr\_0.3.4 readr\_1.4.0 tidyr\_1.1.2   
[13] tibble\_3.0.6 tidyverse\_1.3.0 ggplot2\_3.3.3 R.utils\_2.10.1   
[17] R.oo\_1.24.0 R.methodsS3\_1.8.1  
  
loaded via a namespace (and not attached):  
 [1] Rcpp\_1.0.6 lubridate\_1.7.9.2 lattice\_0.20-41 assertthat\_0.2.1   
 [5] digest\_0.6.27 R6\_2.5.0 cellranger\_1.1.0 backports\_1.2.1   
 [9] reprex\_1.0.0 evaluate\_0.14 highr\_0.8 httr\_1.4.2   
[13] pillar\_1.4.7 rlang\_0.4.10 readxl\_1.3.1 rstudioapi\_0.13   
[17] Matrix\_1.2-18 rmarkdown\_2.6 labeling\_0.4.2 munsell\_0.5.0   
[21] broom\_0.7.4 compiler\_4.0.2 Deriv\_4.1.2 modelr\_0.1.8   
[25] xfun\_0.20 pkgconfig\_2.0.3 htmltools\_0.5.1.1 tidyselect\_1.1.0   
[29] gridExtra\_2.3 crayon\_1.4.0 dbplyr\_2.0.0 withr\_2.4.1   
[33] MASS\_7.3-51.6 grid\_4.0.2 jsonlite\_1.7.2 gtable\_0.3.0   
[37] lifecycle\_0.2.0 DBI\_1.1.1 magrittr\_2.0.1 scales\_1.1.1   
[41] zip\_2.1.1 cli\_2.3.0 stringi\_1.5.3 farver\_2.0.3   
[45] fs\_1.5.0 xml2\_1.3.2 ellipsis\_0.3.1 generics\_0.1.0   
[49] vctrs\_0.3.6 glue\_1.4.2 hms\_1.0.0 yaml\_2.2.1   
[53] colorspace\_2.0-0 rvest\_0.3.6 knitr\_1.31 haven\_2.3.1

RStudio version 1.3.1073.

END OF SCRIPT