Plots - use-wear archaeology & experiments

Lisa Schunk

2021-02-04 14:16:12

# Goal of the script

This script combines the results from the quantitative use-wear analysis performed on archaeological samples as well as on standard samples used during two experiments (aVSn and TFE)

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\_all.Rbin 112c1ab2c73b661bfc7a8791fab8d25d

# Load data into R object

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

'data.frame': 234 obs. of 60 variables:  
 $ Sample : chr "FLT4-12" "FLT4-12" "FLT4-12" "FLT4-12" ...  
 $ Site : Factor w/ 5 levels "aVSn","TFE","Balve",..: 1 1 1 1 1 1 1 1 1 1 ...  
 $ Tool.type : Factor w/ 5 levels "Standard sample",..: 1 1 1 1 1 1 1 1 1 1 ...  
 $ Raw.material : Factor w/ 2 levels "flint","lydite": 1 1 1 1 1 1 1 1 1 1 ...  
 $ Location : chr "B" "B" "B" "B" ...  
 $ Sublocation : chr "1" "1" "1" "1" ...  
 $ Area : chr "01" "01" "01" "01" ...  
 $ Spot : chr "a" "b" "c" "a" ...  
 $ Usewear.type : chr "before" "before" "before" "2000" ...  
 $ Contact.material : chr "skin pad" "skin pad" "skin pad" "skin pad" ...  
 $ Task : chr "cutting" "cutting" "cutting" "cutting" ...  
 $ Edge.angle : chr "60°" "60°" "60°" "60°" ...  
 $ Objective : chr "50x095" "50x095" "50x095" "50x095" ...  
 $ Analysis.date : Date, format: "2020-08-17" "2020-08-17" ...  
 $ Analysis.time : 'times' num 0.617 0.618 0.618 0.619 0.619 ...  
 ..- attr(\*, "format")= chr "h:m:s"  
 $ Acquisition.date.time : chr "7/20/2020 2:39:46 PM" "7/20/2020 2:55:08 PM" "7/20/2020 3:12:51 PM" "7/20/2020 3:29:51 PM" ...  
 $ 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 27287 27231 26655 13395 13291 ...  
 $ Axis.size.Z : num 63694 65201 63762 65466 65538 ...  
 $ Axis.spacing.Z : num 0.428 0.418 0.418 0.205 0.203 ...  
 $ NM.points.ratio.Z : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ Sq : num 160 151 196 227 160 ...  
 $ Ssk : num -0.0199 4.1704 2.0253 2.2084 2.0238 ...  
 $ Sku : num 6.21 37.01 10.42 11.22 9.9 ...  
 $ Sp : num 1153 1693 1606 1740 1174 ...  
 $ Sv : num 716 594 406 473 321 ...  
 $ Sz : num 1870 2286 2013 2212 1496 ...  
 $ Sa : num 115.8 84.7 138.7 154.4 111.9 ...  
 $ Smr : num 12.722 0.832 1.59 1.645 10.656 ...  
 $ Smc : num 177 111 232 250 184 ...  
 $ Sxp : num 362 192 211 245 175 ...  
 $ Sal : num 5.94 7.73 8.18 12.2 8.71 ...  
 $ Str : num 0.0746 0.669 0.226 0.6994 0.1989 ...  
 $ Std : num 170 133 50.5 101 81.5 ...  
 $ Sdq : num 0.1019 0.0833 0.1177 0.1378 0.1048 ...  
 $ Sdr : num 0.513 0.337 0.68 0.916 0.54 ...  
 $ Vm : num 0.0103 0.0176 0.0206 0.0263 0.0175 ...  
 $ Vv : num 0.187 0.128 0.252 0.276 0.202 ...  
 $ Vmp : num 0.0103 0.0176 0.0206 0.0263 0.0175 ...  
 $ Vmc : num 0.119 0.073 0.133 0.143 0.106 ...  
 $ Vvc : num 0.164 0.115 0.241 0.262 0.192 ...  
 $ Vvv : num 0.0233 0.0128 0.0112 0.0138 0.0092 ...  
 $ Maximum.depth.of.furrows: num 905 865 814 881 640 ...  
 $ Mean.depth.of.furrows : num 257 174 386 409 311 ...  
 $ Mean.density.of.furrows : num 3081 3318 3101 3225 3191 ...  
 $ First.direction : num 1.69e+02 1.35e+02 8.76e-03 9.00e+01 9.00e+01 ...  
 $ Second.direction : num 180 90 135 135 45 ...  
 $ Third.direction : num 135 45 117 116 135 ...  
 $ Isotropy : num 6.16 66.63 52.93 77.77 55.86 ...  
 $ epLsar : num 0.00356 0.00393 0.00189 0.00195 0.00041 ...  
 $ NewEplsar : num 0.0187 0.0188 0.0179 0.0179 0.0175 ...  
 $ Asfc : num 1.145 0.703 1.494 2.067 1.207 ...  
 $ Smfc : num 3.08 3.3 2.88 2.52 2.52 ...  
 $ HAsfc9 : num 0.201 0.636 0.191 0.512 0.239 ...  
 $ HAsfc81 : num 0.264 0.939 0.388 0.722 0.386 ...

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

# Prepare variables

## Define numeric variables

num.var <- 27:length(imp\_data)

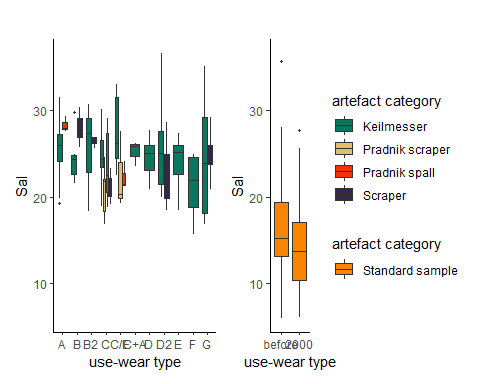
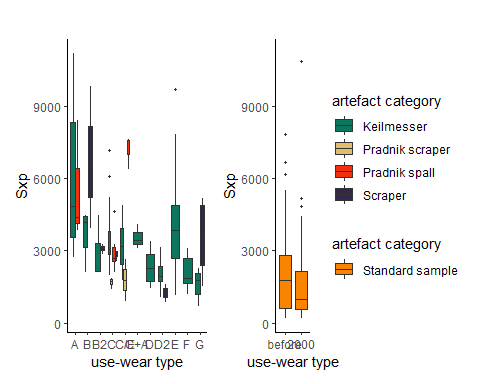
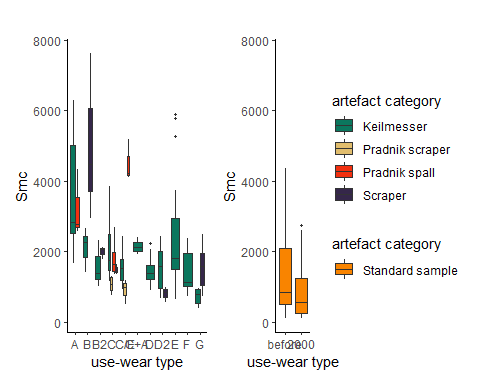
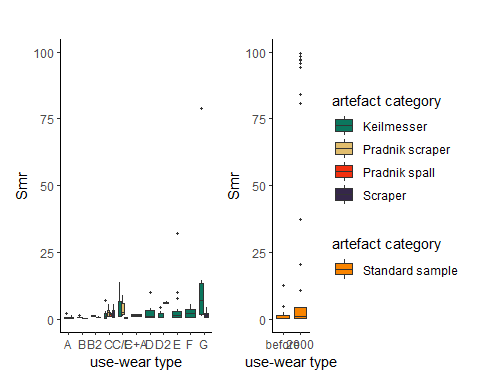
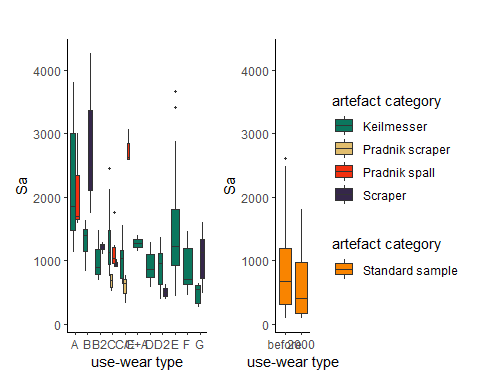
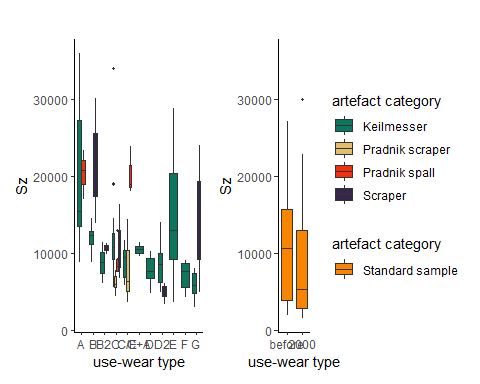
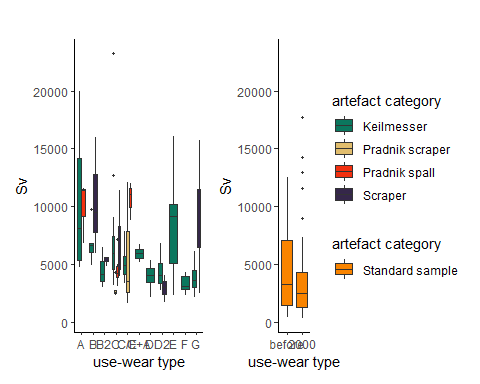
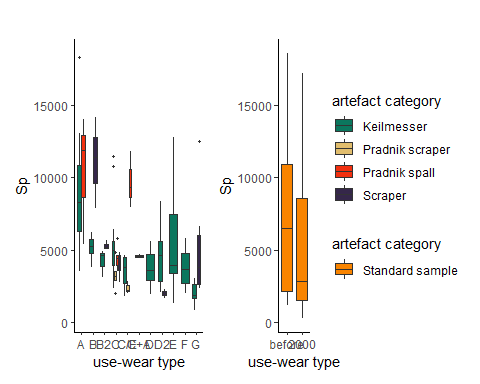
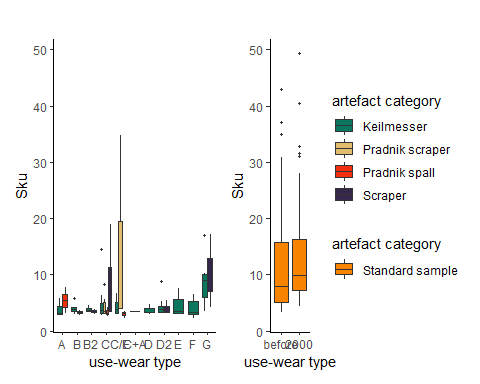
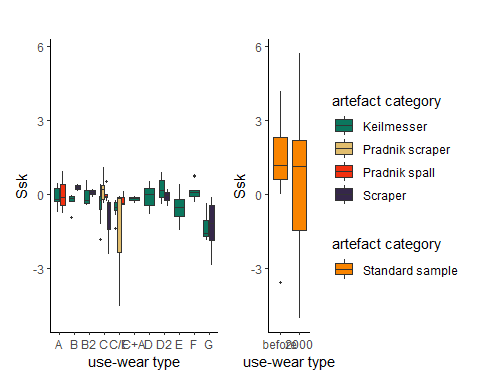
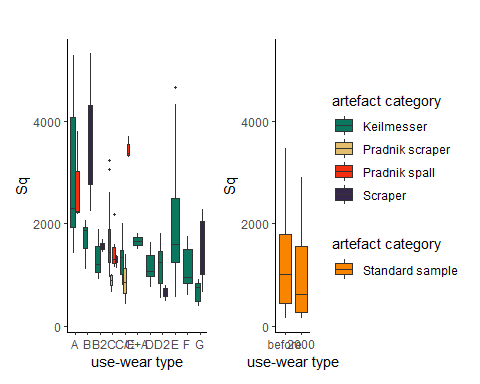
The following variables will be used:

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

# Plot each of the selected numeric variables

## Boxplot of all the variables combined with the artefact category (without outliers)

# excludes the outliers (from both data sets)  
# adds the indices as row numbers   
imp\_data <- imp\_data %>% mutate(id = row\_number())  
imp\_data2 <- imp\_data[-c(19, 20, 55, 59, 62, 70, 139, 147, 199, 200), ]  
  
  
arch <- filter(imp\_data2, Site == "Balve" | Site == "Buhlen" | Site == "Ramioul")   
exp <- filter(imp\_data2, Site == "aVSn" | Site == "TFE")  
  
# converts the data into factor   
arch[["Tool.type"]] <- factor(arch[["Tool.type"]])  
exp[["Tool.type"]] <- factor(exp[["Tool.type"]])  
  
# colours archaeological data  
custom.col2 <- data.frame(type = levels(arch$Tool.typ),   
 col = c( "#0B775E", "#E1BD6D", "#F2300F", "#35274A"))   
arch$col <- custom.col2[arch$Tool.typ %in% levels(arch[["Tool.type"]]), "col"]  
  
# colours experimental data   
# Darjeeling1 = c("#FF0000", "#00A08A", "#F2AD00", "#F98400", "#5BBCD6")  
custom.col6 <- data.frame(type = levels(exp$Tool.typ), col = "#F98400")   
exp$col <- custom.col6[exp$Tool.typ %in% levels(exp[["Tool.type"]]), "col"]  
  
# gets new order   
exp$Usewear.type <- factor(exp$Usewear.type, levels=c("before", "2000"))  
  
  
# plot tool type   
for (i in num.var){  
   
 # gets the min/max range of the data set  
range\_var <- range(imp\_data2[[i]])   
  
   
 p\_arch <- ggplot(data = arch, aes\_string(x = "Usewear.type", y = names(imp\_data)[i],  
 fill = "Tool.type")) +  
 geom\_boxplot(outlier.size = 0.7, lwd = 0.2) +  
 theme\_classic() +  
 coord\_cartesian(ylim = range\_var) +   
 labs( x = "use-wear type", title = " ") +  
 labs(y = gsub("\\.", " ", names(imp\_data)[i])) +  
 labs(fill = "artefact category") +   
 scale\_fill\_manual(values = arch$col)  
   
   
 p\_exp <- ggplot(data = exp, aes\_string(x = "Usewear.type", y = names(imp\_data)[i],  
 fill = "Tool.type")) +  
 geom\_boxplot(outlier.size = 0.7, lwd = 0.2) +  
 theme\_classic() +  
 coord\_cartesian(ylim = range\_var) +   
 labs( x = "use-wear type", title = " ") +  
 labs(y = gsub("\\.", " ", names(imp\_data)[i])) +  
 labs(fill = "artefact category") +   
 scale\_fill\_manual(values = exp$col)  
   
  
 # combines the plots   
 p <- p\_arch + p\_exp + plot\_layout(width = c(4, 1), guides = 'collect')  
   
 print(p)  
   
 # saves the plots   
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_boxplot\_arch.exp\_",   
 names(imp\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = p, path = dir\_out, device = "pdf",   
 width = 250, height = 170, units = "mm")  
   
}

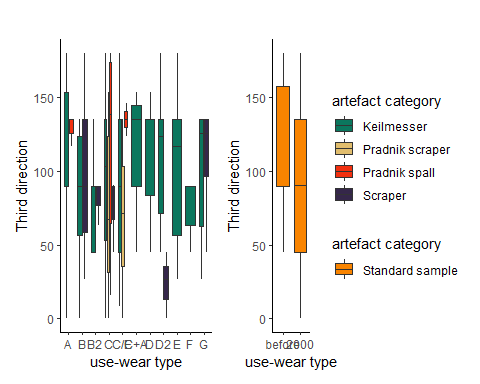
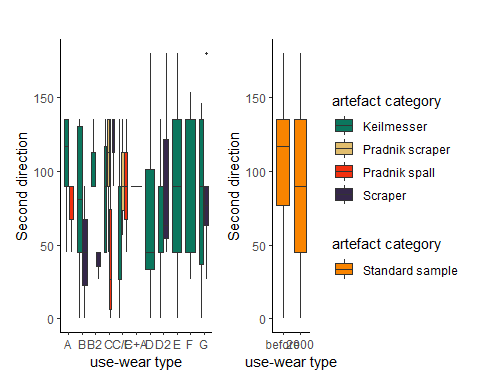
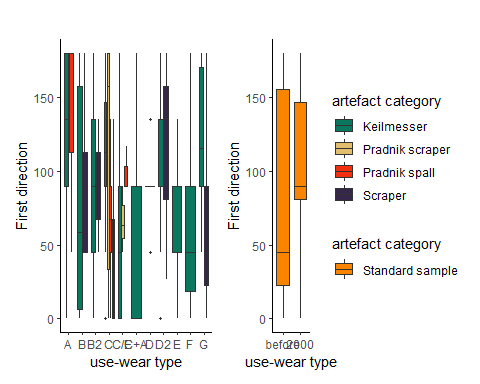
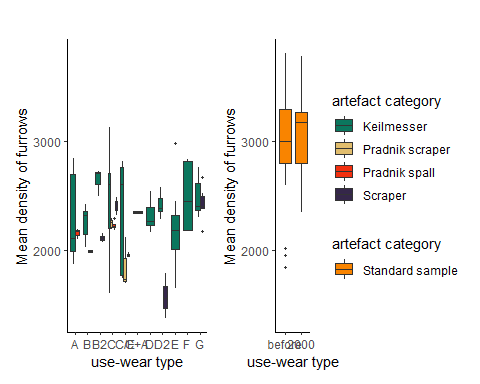
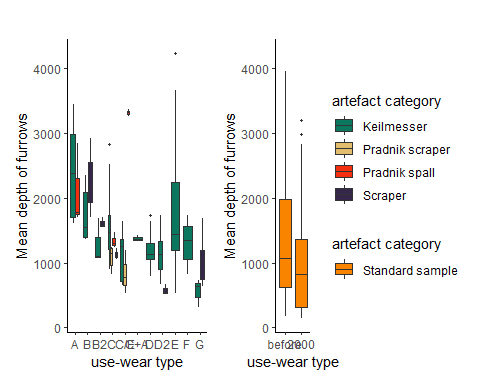
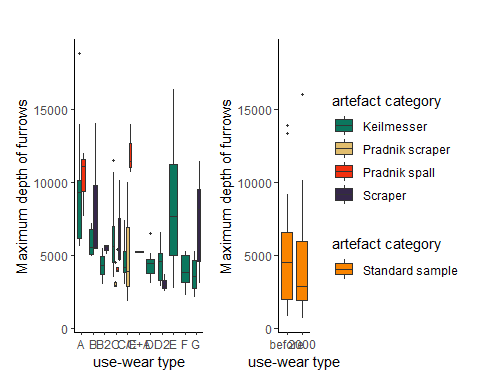
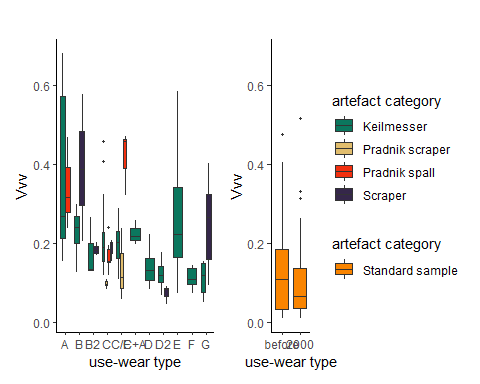
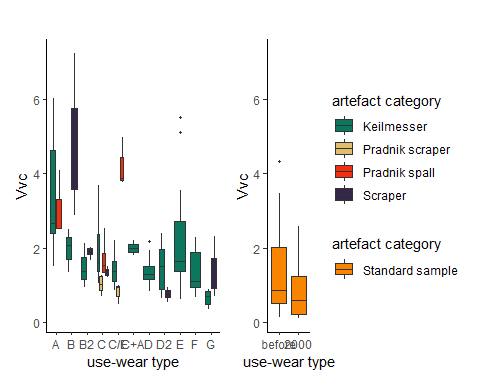
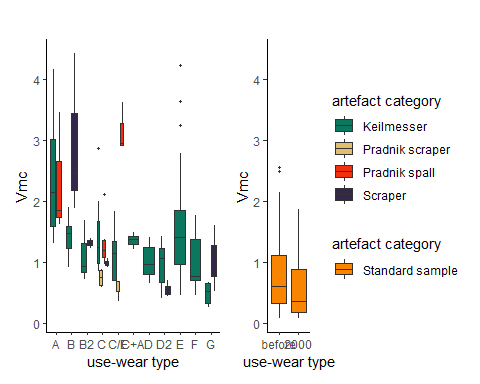
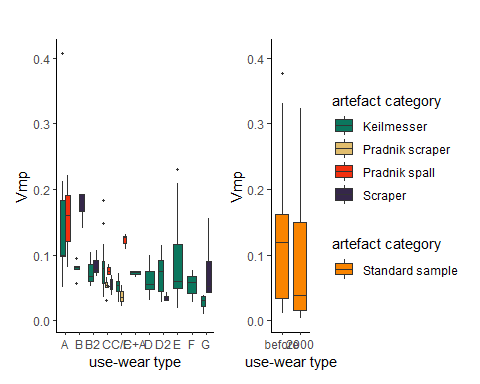
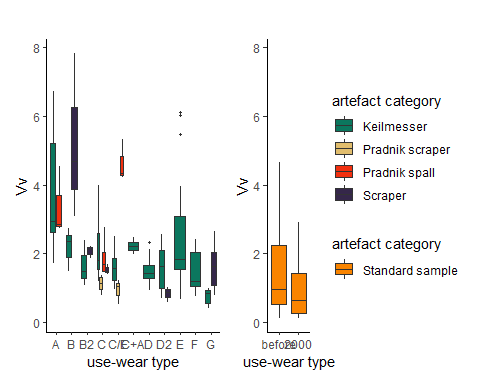
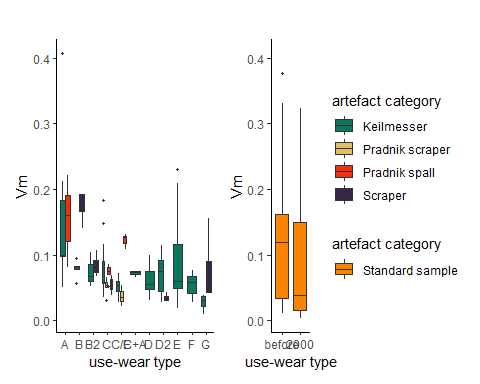
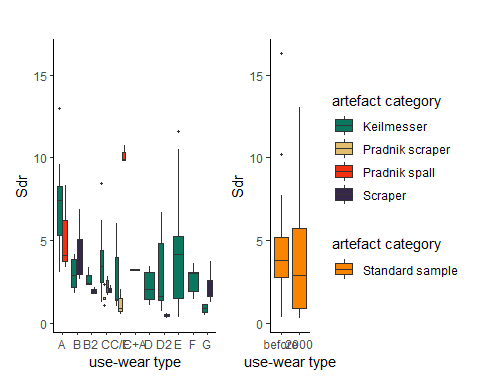
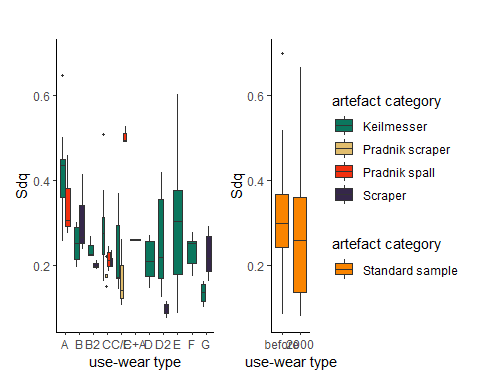
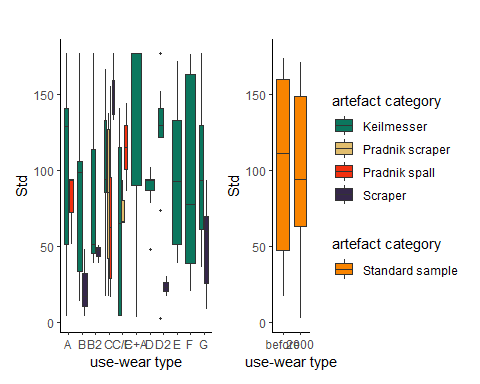
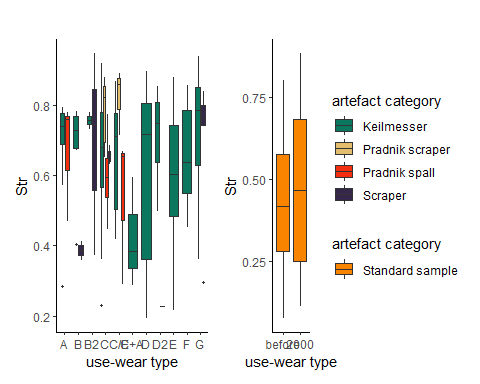


Warning: Removed 9 rows containing non-finite values (stat\_boxplot).

Warning: Removed 8 rows containing non-finite values (stat\_boxplot).

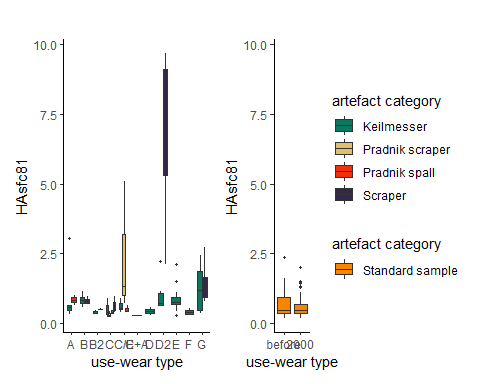
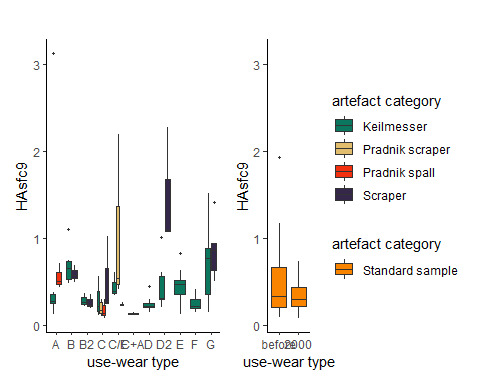
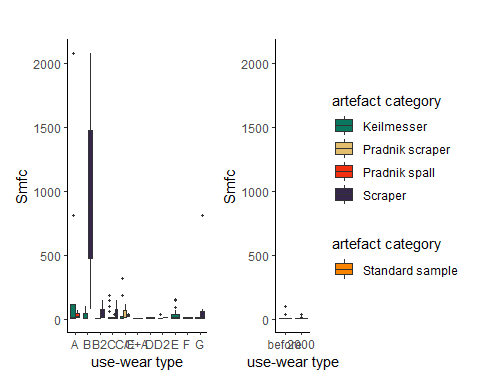
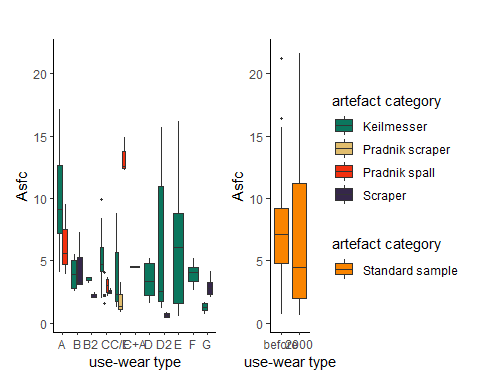
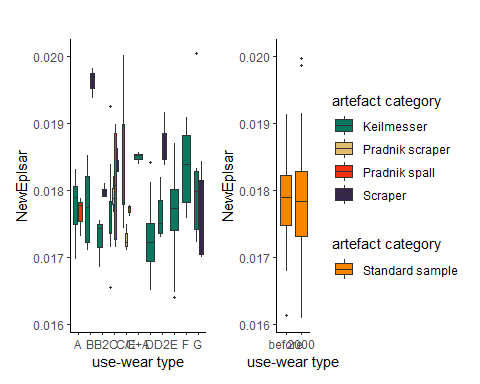
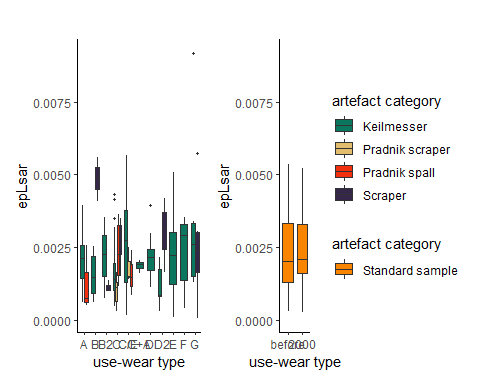
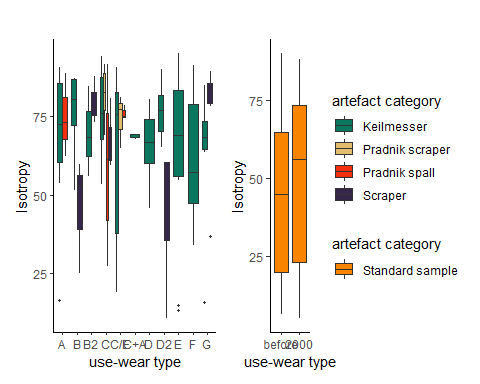
Warning: Removed 9 rows containing non-finite values (stat\_boxplot).

Warning: Removed 8 rows containing non-finite values (stat\_boxplot).



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

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



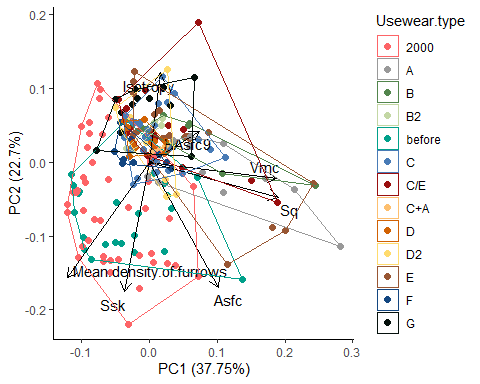
## Principal component analysis (without outliers)

# PCA use-wear types  
# removes rows with na values   
data\_pca <- na.omit(imp\_data2)  
# uses for the PCA only selected variables: Sq, SSK, Vmc, Isotropy, Mean density of furrows, Asfc, HAsfc9   
imp\_data.pca <- prcomp(data\_pca[, c(27:28, 45, 50, 54, 57, 59)], scale. = TRUE)   
# converts the data into factor   
data\_pca[["Usewear.type"]] <- factor(data\_pca[["Usewear.type"]])  
  
  
custom.col1 <- data.frame(type = levels(data\_pca$Usewear.typ),   
 col = c("#FD6467", "#999999", "#52854c", "#c3d7a4", "#00A08A",  
 "#487bb6", "#9a0f0f", "#fdbf6f", "#d16103", "#ffdb6d",  
 "#985633", "#134680", "#05100c"))   
data\_pca$col <- custom.col1[data\_pca$Usewear.typ, "col"]  
  
  
# Using ggfortify  
PCA<- autoplot(imp\_data.pca, data = data\_pca, colour = "Usewear.type", size = 2,  
 loadings = TRUE, loadings.colour = "black", loadings.label = TRUE, loadings.label.colour = "black",   
 loadings.label.size = 4, loadings.label.repel = TRUE,   
 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.

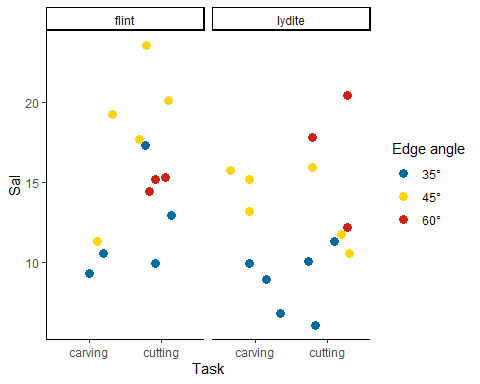
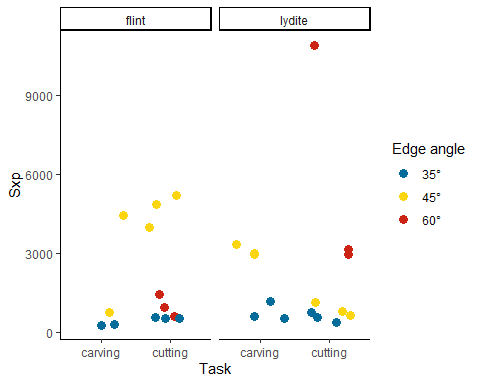
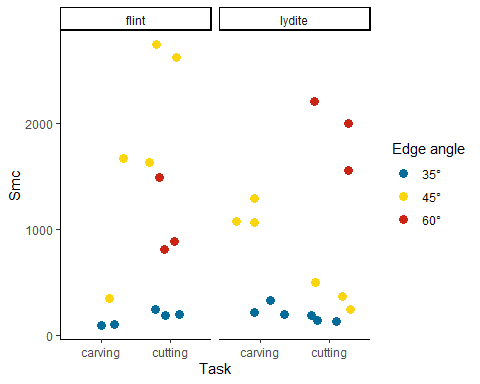
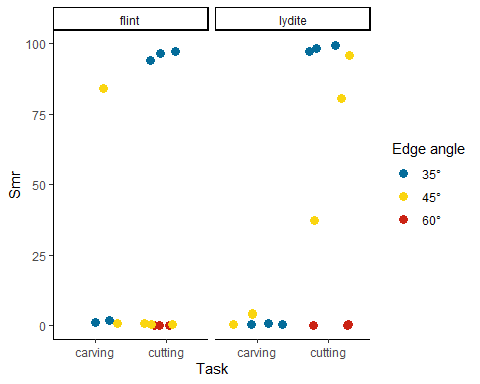
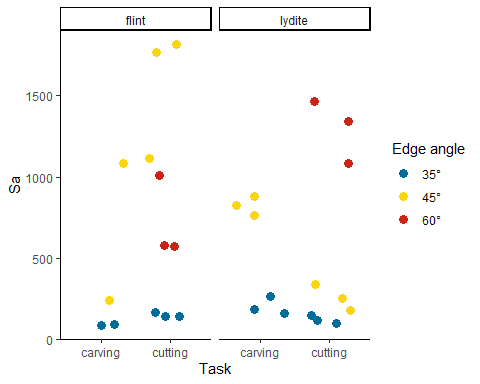
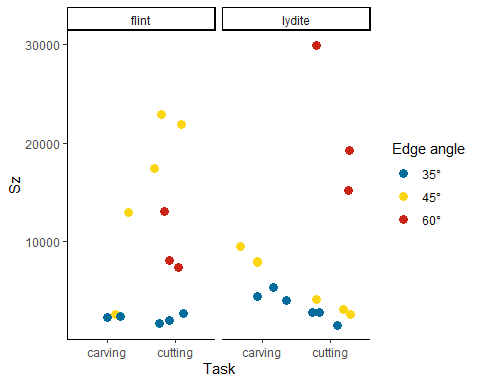
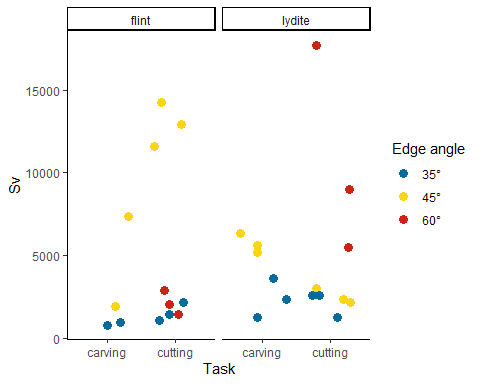
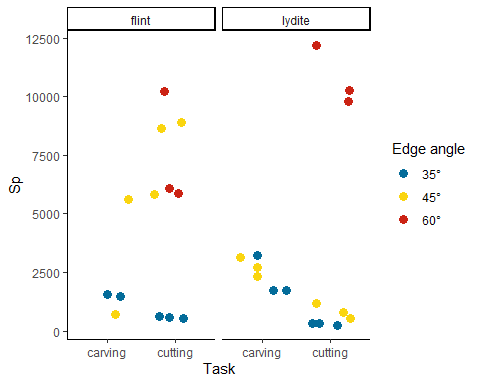
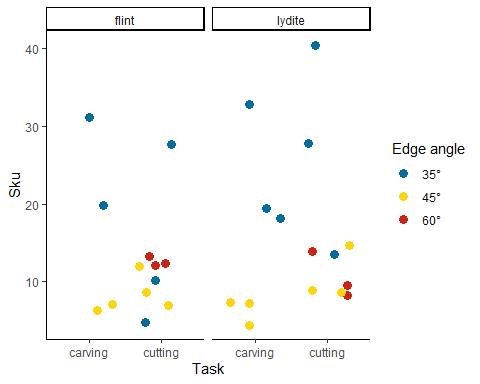
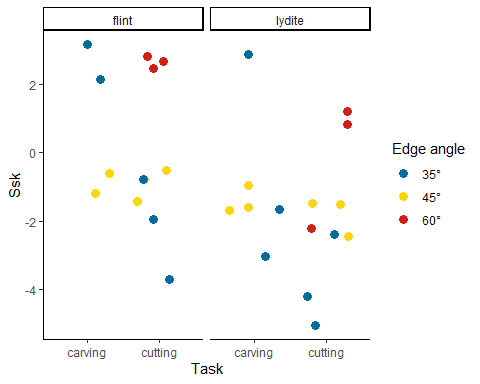
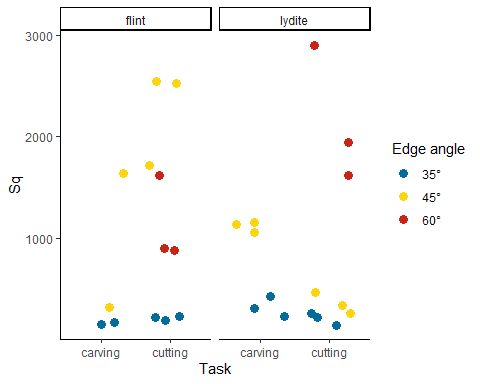
print(PCA)



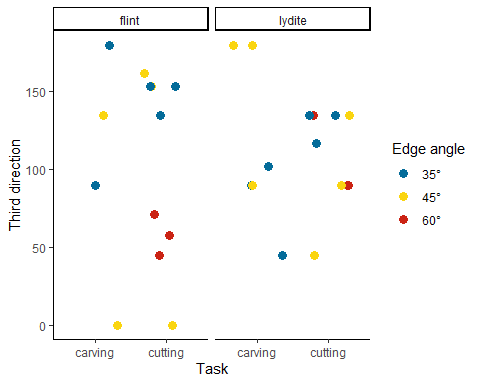
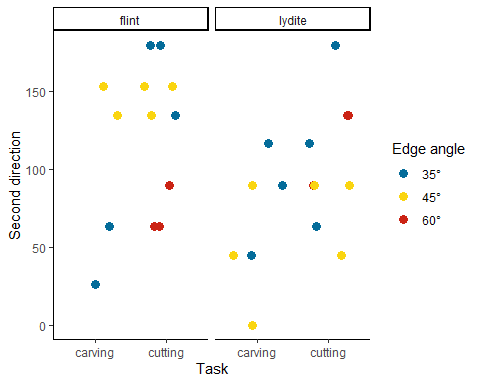
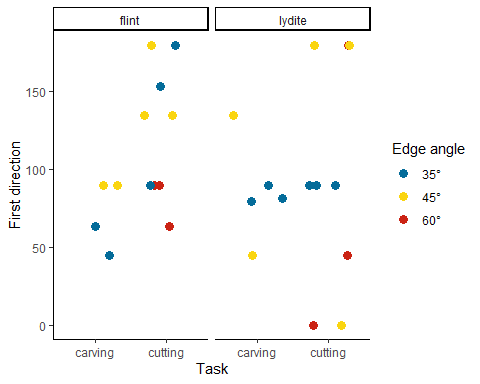
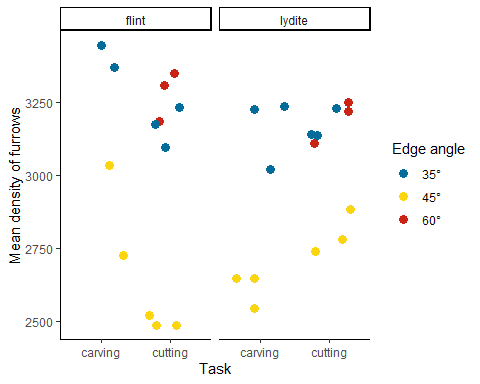
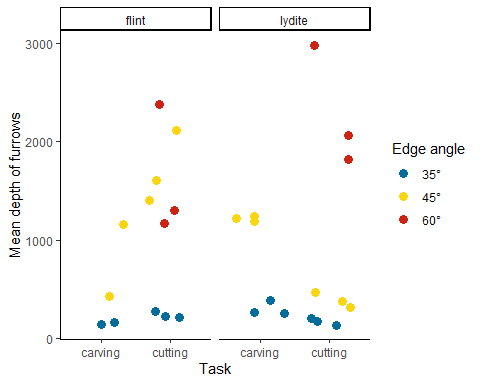
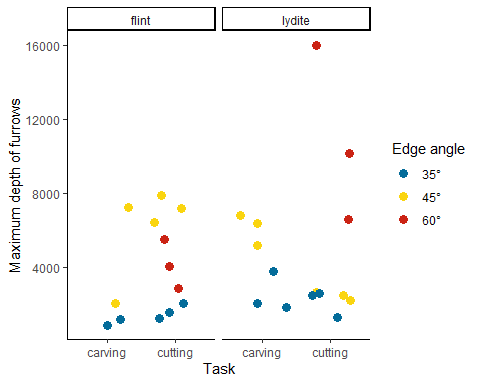
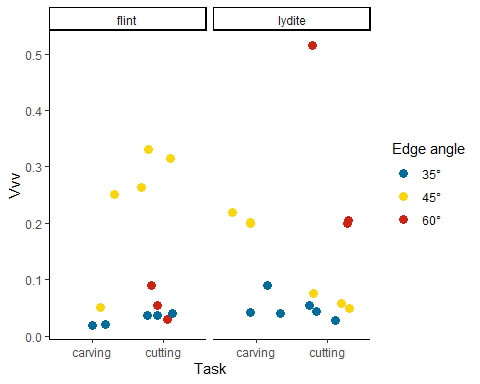
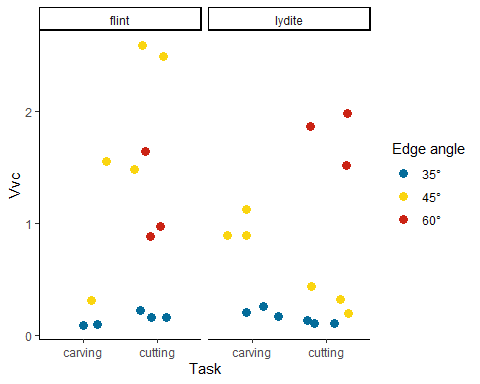
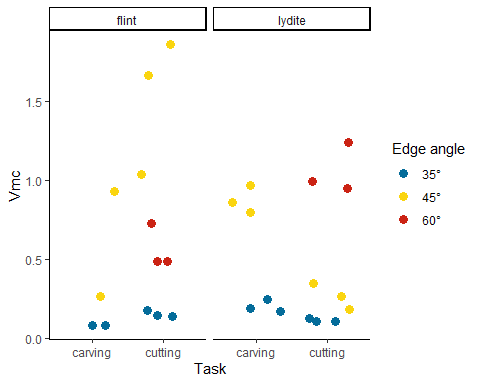
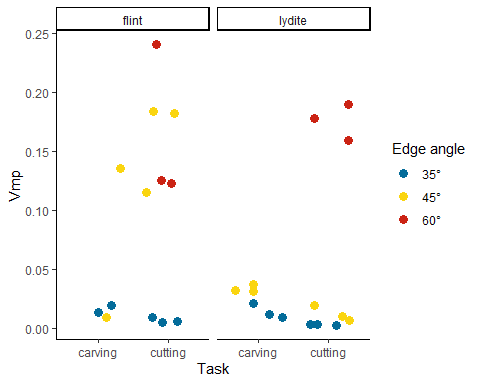
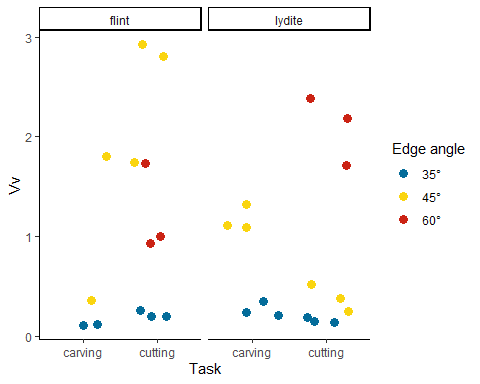
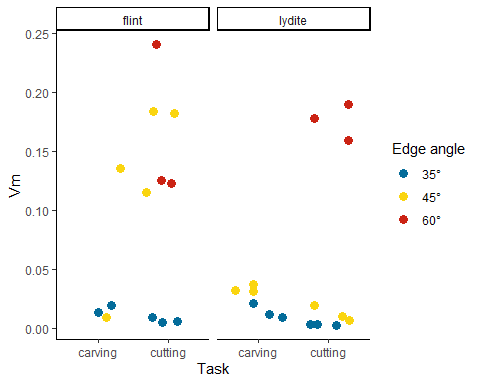
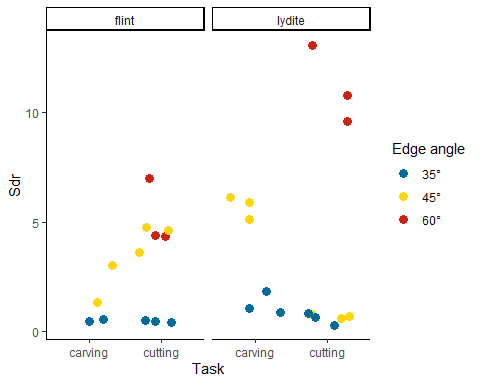
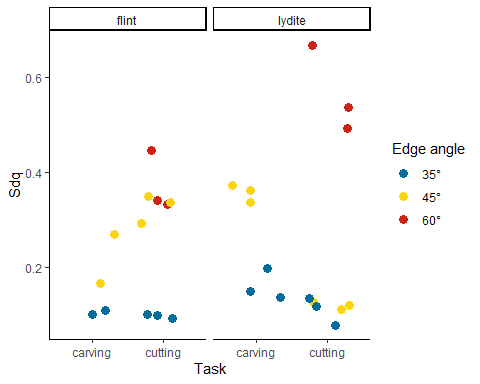
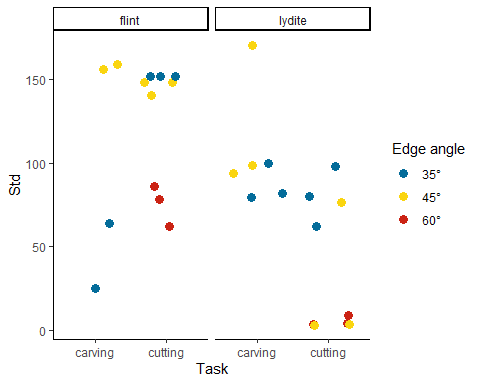
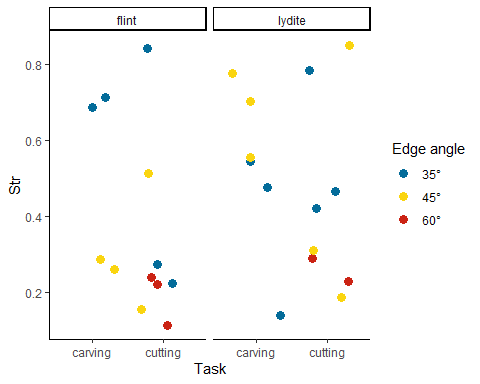
# saves the plot  
file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_PCA\_all.data", ".pdf")  
ggsave(filename = file\_out, plot = PCA, path = dir\_out, device = "pdf")

## (facet plot = 1 plot for flint, 1 plot for lydite)

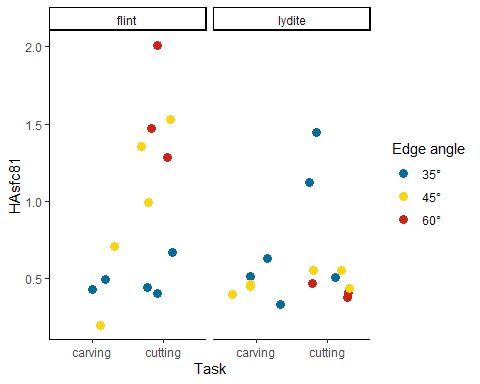
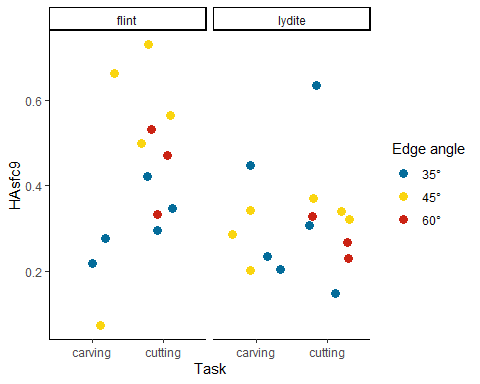
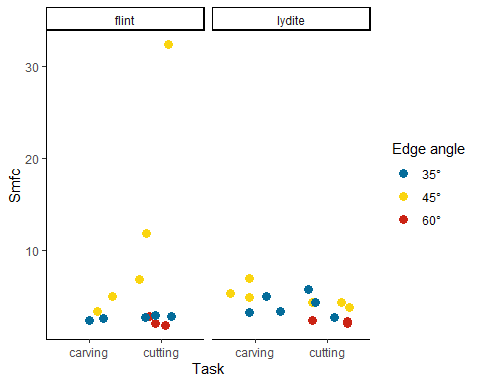
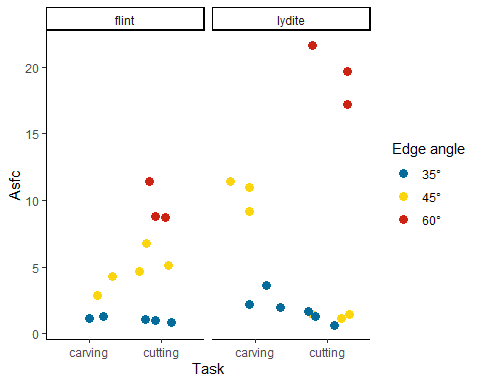
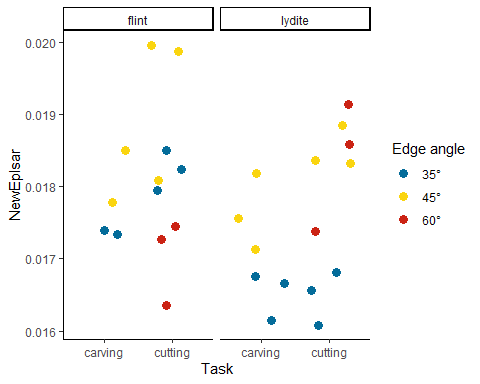
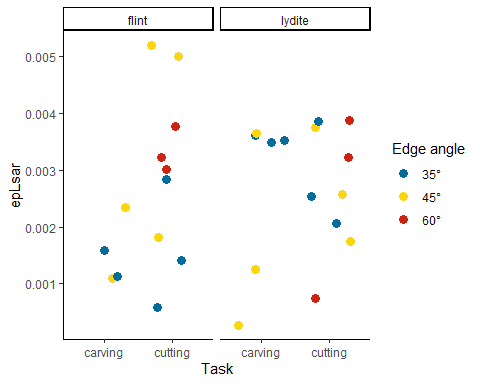
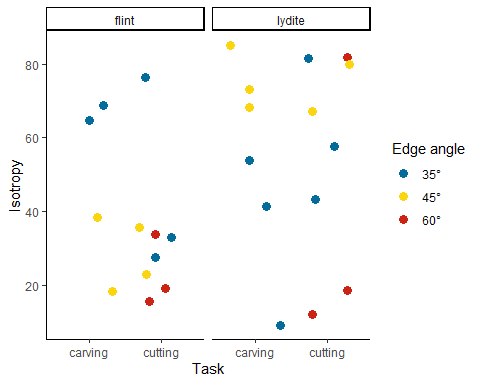
# selects only the data from after 2000 strokes on the bone plate (experimental data)   
exp2000 <- filter(exp, Usewear.type == "2000" & Contact.material == "bone plate")  
  
for (i in num.var){  
 #plot  
 range\_var <- range(exp2000[[i]]) # gets the min/max range of the data set   
   
 p <- ggplot(data = exp2000, aes\_string(x = "Task", y = names(exp2000)[i],  
 colour = "Edge.angle")) +   
 # avoids overplotting  
 scale\_color\_manual(values=c("#046C9A","#FAD510", "#CB2314")) +  
 geom\_jitter(size = 3, position = position\_jitter(width = 0.35, seed = 1)) +   
 coord\_cartesian(ylim = range\_var) +   
 theme\_classic() +   
 # removes the "." between "Edge.angle" in the legend   
 labs(colour = gsub("\\.", " ", "Edge.angle")) +   
 # removes the "." between the variable names  
 labs(y = gsub("\\.", " ", names(exp2000)[i])) +   
 # combines the flint and the lydite plot into one   
 facet\_wrap(~Raw.material)   
   
   
 print(p)  
  
  
 #save to PDF  
 file\_out <- paste0(file\_path\_sans\_ext(info\_in[["file"]]), "\_exp\_2000\_plot\_",   
 names(imp\_data)[i], ".pdf")  
 ggsave(filename = file\_out, plot = p, path = dir\_out, device = "pdf")  
}



Warning: Removed 2 rows containing missing values (geom\_point).  
  
Warning: Removed 2 rows containing missing values (geom\_point).



Warning: Removed 2 rows containing missing values (geom\_point).  
  
Warning: Removed 2 rows containing missing values (geom\_point).



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