

Multiproxy palaeoenvironmental reconstruction from Tramacastilla

Irene Julián-Posada

2025-07-11

This is the code used for multiproxy analysis from Tramacastilla Lake sequence, included in Julián-Posada et al. (in prep).

1. Load ages and depths

```
head(tram20_depth_ages) # ages and depths for core TRAM20-1B
```

```
##           SampleId composite_depth depth min  max median mean
## 1 TRAM20-1B-3U-19-20           390 390.9 772  937   863  861
## 2 TRAM20-1B-3U-20-21           392 392.9 803 1016   902  903
## 3 TRAM20-1B-3U-21-22           394 394.9 814 1104   939  945
## 4 TRAM20-1B-3U-22-23           395 395.9 820 1149   959  967
## 5 TRAM20-1B-3U-23-24           397 397.9 834 1238   999 1009
## 6 TRAM20-1B-3U-24-25           398 398.9 838 1282  1020 1030
```

```
tram20_depth_ages_sed_rate <- tram20_depth_ages %>%
  rename("age" = "median") %>%
  select(-depth) %>%
  rename("depth" = "composite_depth") %>%
  mutate(depth_diff = c(NA, diff(depth)),
         age_diff = c(NA, diff(age)),
         sed_rate = age_diff/depth_diff)
```

2. Tramacastilla charcoal

2.1. Data upload

```
head(charcoal.tram) #charcoal data from Tramacastilla (core TRAM20-1B) obtained using Win seedle Pro 20
```

```
##           SampleId ImageId ImageType ObjectId          DPI.ProjAreaObj
## 1 TRAM-3U-19-20      1      GLOBAL  Scanner 0.0000 35.211270 34.843210
## 2 TRAM-3U-19-20      1      SEEDLE      1          18403.1051
## 3 TRAM-3U-19-20      2      GLOBAL  Scanner 0.0000 35.211270 34.843210
## 4 TRAM-3U-20-21      1      GLOBAL  Scanner 0.0000 35.211270 34.843210
## 5 TRAM-3U-20-21      1      SEEDLE      1          7361.242
```

```
## 6 TRAM-3U-20-21      2      GLOBAL Scanner 0.0000 35.211270 34.843210
##      NPath.WLRatio TotalProjAreaObj.Hpos NofSeedles
## 1      0.0000      18403.105      1
## 2      0.2858      21110.330      NA
## 3      0.0000      0.000      0
## 4      0.0000      7361.242      1
## 5      1.0063      20897.889      NA
## 6      0.0000      12268.737      1
```

```
head(samples_names) #names of charcoal samples
```

```
##      SampleId
## 1 TRAM-3U-19-20
## 2 TRAM-3U-20-21
## 3 TRAM-3U-21-22
## 4 TRAM-3U-22-23
## 5 TRAM-3U-23-24
## 6 TRAM-3U-24-25
```

```
Tramacastilla_age <- tram20_depth_ages %>% #ages and depths for charcoal samples
  select(SampleId, median, composite_depth) %>%
  rename("age" = "median",
         "depth" = "composite_depth") %>%
  mutate(age = as.numeric(age),
         depth = as.numeric(depth)) %>%
  mutate(SampleId = str_replace_all(SampleId, "TRAM20-1B-", "TRAM-")) %>%
  filter(SampleId %in% samples_names$SampleId) %>%
  mutate_all(~replace(., is.na(.), 0))
```

There are 194 instead of 202 samples, as some of them have no data (no charcoal particles were found), so add them.

```
charcoal.tram.202 <- full_join(charcoal.tram, samples_names, by = "SampleId") %>%
  mutate(ImageType = ifelse(is.na(ImageType), "GLOBAL", ImageType)) #change ImageType to say GLOBAL for
length(unique(charcoal.tram.202$SampleId)) #make sure there are 202 samples
```

```
## [1] 202
```

Now duplicate this new rows to obtain one row for GLOBAL and other for SEEDLE

```
samples_no_data <- charcoal.tram.202 %>%
  filter(is.na(ImageId)) %>%
  mutate(ImageType = "SEEDLE")

charcoal.tram.all <- rbind(charcoal.tram.202, samples_no_data) %>%
  mutate_all(~replace(., is.na(.), 0))

length(unique(charcoal.tram.all$SampleId)) #make sure there are 202 samples
```

```
## [1] 202
```

Create 2 dataframes: one for global and another for seedle

```
Gchar <- subset(charcoal.tram.all, ImageType=="GLOBAL")
Schar <- charcoal.tram.all %>%
  subset(ImageType=="SEEDLE") %>%
  mutate(NPath.WLRatio = ifelse(NPath.WLRatio > 1, 1/NPath.WLRatio, NPath.WLRatio)) %>% #change values
  mutate(NofSeedles = as.numeric(NofSeedles)) %>%
  mutate(DPI.ProjAreaObj = as.numeric(DPI.ProjAreaObj)) %>%
  mutate(ObjectId = as.numeric(ObjectId))

length(unique(Gchar$SampleId)) #make sure there are 202 samples for GLOBAL
```

```
## [1] 202
```

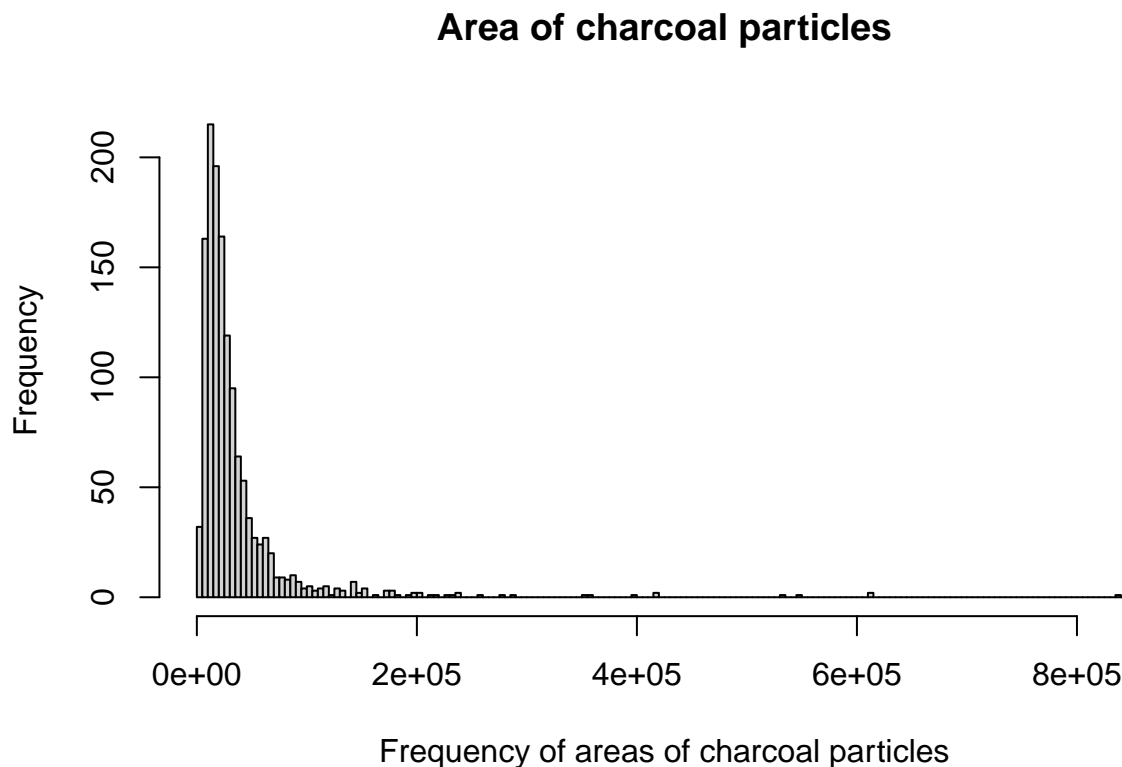
```
length(unique(Schar$SampleId)) #make sure there are 202 samples for SEEDLE
```

```
## [1] 202
```

2.2. Make a new data frame only with charcoal particles larger than 22500

First see the frequencies of area

```
hist(Schar$DPI.ProjAreaObj, breaks = seq(0, 850000, 5000),
     xlab = "Frequency of areas of charcoal particles",
     main = paste("Area of charcoal particles"))
```



```
summary(Schar$DPI.ProjAreaObj)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.     NA's
##         0   13496   22084   35713   36806   836728         2
```

25% of the particles are below 13496 um², which is a sieve diameter of 116 um, and mean 35713. Now filter particles so remove all that are below 110 um, that we assume is the noisy part of the signal -> remove those charcoal with an area <110 um

```
seedles <- charcoal.tram.all %>%
  filter(ImageType == "SEEDLE" & DPI.ProjAreaObj > 12100) %>%
  mutate(NPath.WLRatio = ifelse(NPath.WLRatio > 1, 1/NPath.WLRatio, NPath.WLRatio)) %>% #change values
  as.data.frame() %>%
  mutate(DPI.ProjAreaObj = as.numeric(DPI.ProjAreaObj))
```

Now create another data frame with two columns, SampleId and CharCount, that is the number of particles that meet the requirements

```
counting <- seedles %>%
  group_by(SampleId) %>%
  summarise(CharCount = n())
```

Another data frame with two columns, SampleId and sumCharArea, that is the sum of area of particles that meet the requirements

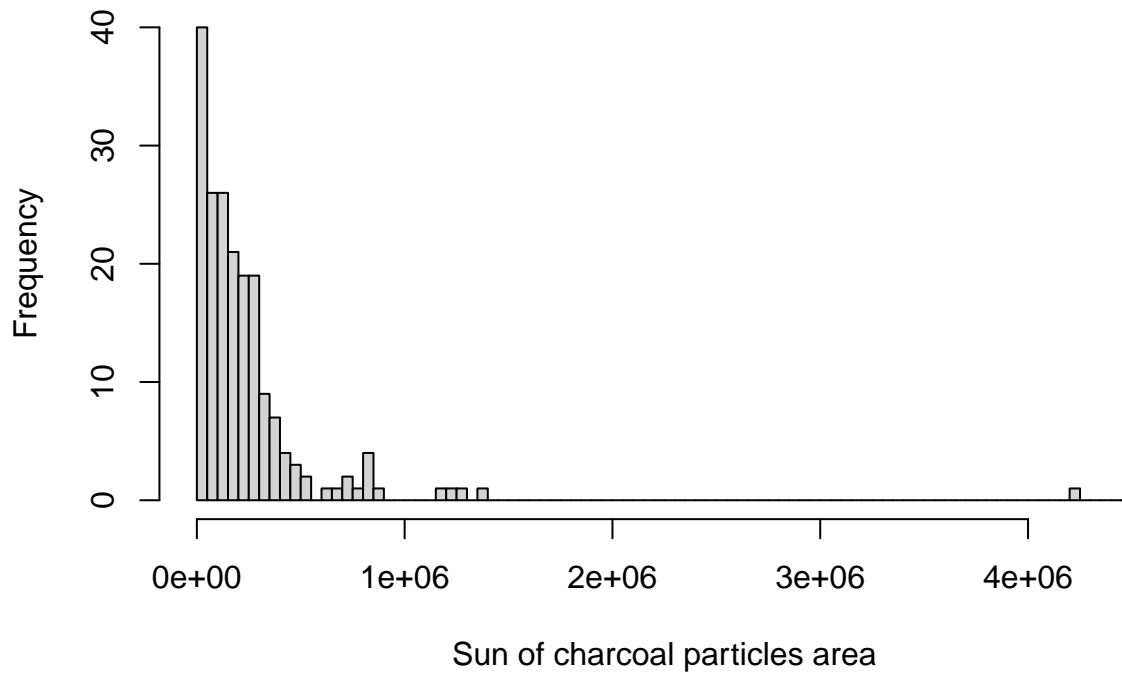
```
totalarea <- seedles %>%
  group_by(SampleId) %>%
  summarise(sumCharArea = sum(DPI.ProjAreaObj, na.rm = TRUE))

summary(totalarea$sumCharArea)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      8588   62571  160720  240249  275433  4230260
```

```
hist(totalarea$sumCharArea, breaks = seq(0, 4500000, 50000),
     main = paste("Frequency of total area of charcoal particles"),
     xlab = "Sum of charcoal particles area")
```

Frequency of total area of charcoal particles



```
seedles.age <- seedles %>%
  merge(Tramacastilla_age, by = "SampleId") %>%
  group_by(SampleId) %>%
  mutate(Color = ifelse(median(NPath.WLRatio) > 0.5, "green", "yellow")) %>%
  distinct() %>%
  ungroup() %>%
  as.data.frame()
```

2.2.1. Create a data frame for plotting

Create a file with one value per sample: count=length sampleID, W:L= mean W:L, area= sum ProjArea

```
CharTotS_G.110 <- merge(counting, totalarea)
CharTotS_G.110.age <- merge(CharTotS_G.110, Tramacastilla_age, by = "SampleId")
```

Check if area and charcoal number are correlated

```
cor.test(CharTotS_G.110.age$sumCharArea, CharTotS_G.110.age$CharCount, method="spearman", exact = FALSE)

##
## Spearman's rank correlation rho
##
## data: CharTotS_G.110.age$sumCharArea and CharTotS_G.110.age$CharCount
## S = 159465, p-value < 2.2e-16
```

```
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.8626815
```

87% of the data is correlated. Now check if that correlation is linnear or not

```
summary(lm(CharTotS_G.110.age$sumCharArea~CharTotS_G.110.age$CharCount))
```

```
##
## Call:
## lm(formula = CharTotS_G.110.age$sumCharArea ~ CharTotS_G.110.age$CharCount)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -424925  -85146   -5418   38736  1024557
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -59588     17264  -3.452  0.000688 ***
## CharTotS_G.110.age$CharCount    44776      1733   25.835 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 176600 on 189 degrees of freedom
## Multiple R-squared:  0.7793, Adjusted R-squared:  0.7781
## F-statistic: 667.4 on 1 and 189 DF,  p-value: < 2.2e-16
```

78% of data is linearly correlated Join these two data frames to have all good charcoal particles with CharCount column

```
CharTotS_S.110 <- seedles %>%
  group_by(SampleId) %>%
  summarise(meanCharWL = mean(NPath.WLRatio, na.rm = TRUE),
            medianCharWL = median(NPath.WLRatio, na.rm = TRUE))

CharTotS.110 <- CharTotS_G.110 %>%
  full_join(CharTotS_S.110, by = "SampleId") %>%
  full_join(samples_names, by = "SampleId") %>% #add samples that are missing
  mutate_all(~replace(., is.na(.), 0))
```

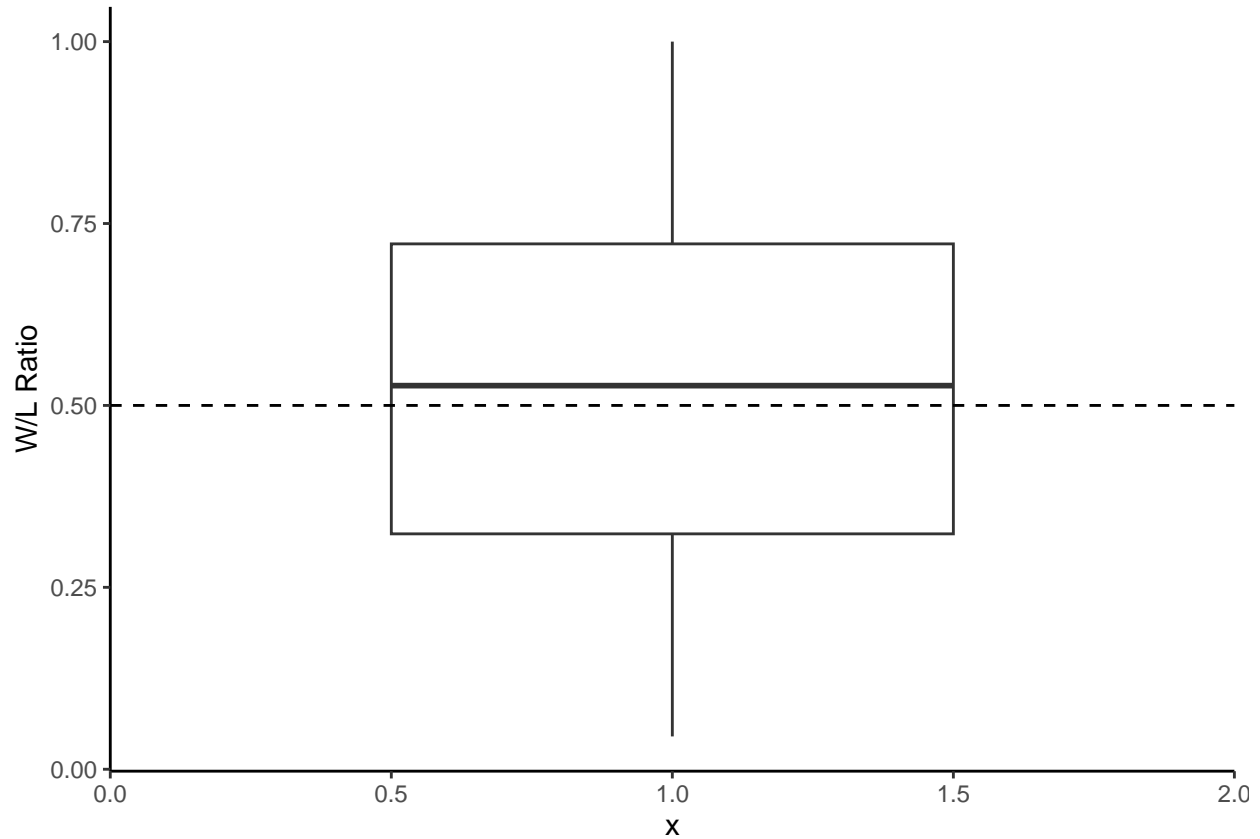
Add ages to order the samples

```
CharTotS.110.age <- full_join(CharTotS.110, Tramacastilla_age, by = "SampleId")
```

2.2.2. Graphic representation

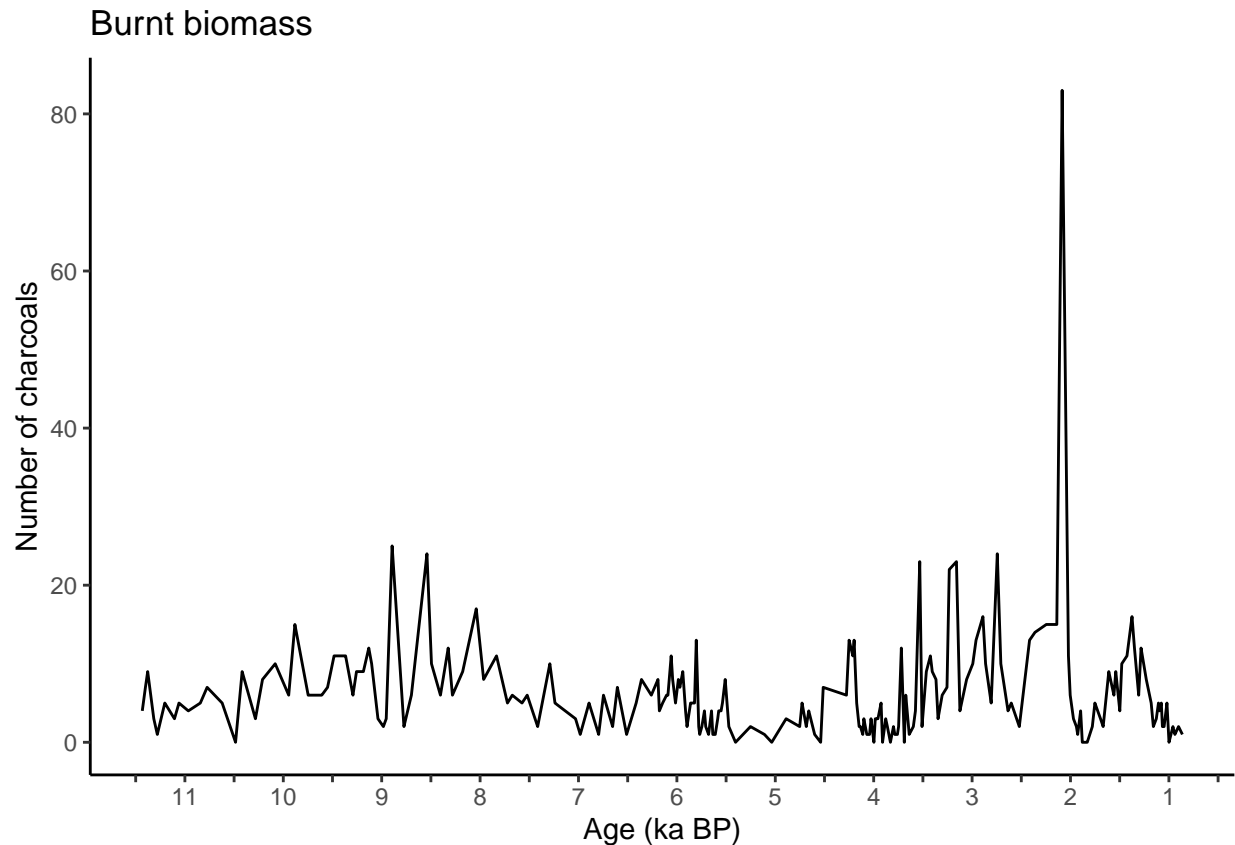
```
seedles %>% #min, max, mean of WLRatio
  ggplot(aes(x= 1, y = NPath.WLRatio)) +
  geom_boxplot(outlier.shape = 20, width = 1) +
```

```
geom_hline(yintercept = 0.5, linetype = "dashed") +
scale_x_continuous(limits = c(0, 2), expand = c(0, 0)) +
theme_classic() +
ylab("W/L Ratio")
```



2.2.3. CharCount

```
CharTotS.110.age %>%
  ggplot(aes(x = age/1000, y = CharCount)) +
  geom_line() +
  xlab("Age (ka BP)") +
  ylab("Number of charcoals") +
  ggtitle("Burnt biomass") +
  theme_classic() +
  scale_x_reverse(breaks = seq(0.5, 12, by = 0.5),
    labels = ifelse(seq(0.5, 12, by = 0.5) %% 1 == 0, as.character(seq(0.5, 12, by = 0.5)),
```



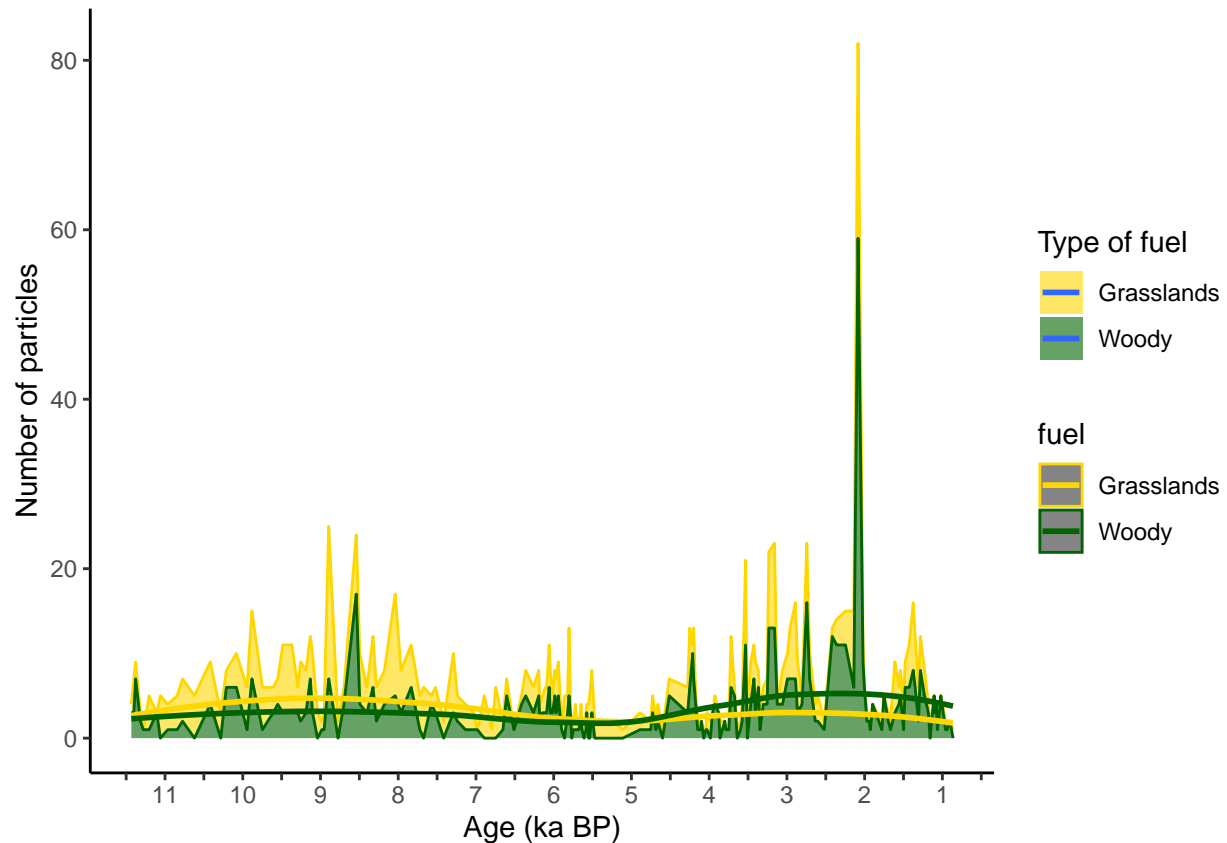
2.2.4. W/L Ratio

```
W.L.ratio <- seedles.age %>%
  group_by(age, depth) %>%
  summarise(Woody = sum(NPath.WLRatio > 0.5),
            Grasslands = sum(NPath.WLRatio < 0.5)) %>%
  pivot_longer(cols = c("Woody", "Grasslands"), names_to = "fuel", values_to = "count")
```

'summarise()' has grouped output by 'age'. You can override using the '.groups' argument.

```
W.L.ratio %>%
  ggplot(aes(x = age/1000, y = count, fill = fuel, color = fuel)) +
  geom_area(alpha = 0.6) +
  geom_smooth(method = "loess", se = FALSE) +
  labs(x = "Age (ka BP)", y = "Number of particles", fill = "Type of fuel") +
  theme_classic() +
  scale_x_reverse(limits = c(max(W.L.ratio$age)/1000, min(W.L.ratio$age)/1000),
                 breaks = seq(0.5, 12, by = 0.5),
                 labels = ifelse(seq(0.5, 12, by = 0.5) %% 1 == 0, as.character(seq(0.5, 12, by = 0.5)),
                                "")) +
  scale_fill_manual(values = c("Woody" = "darkgreen", "Grasslands" = "#FFD700")) +
  scale_color_manual(values = c("Woody" = "darkgreen", "Grasslands" = "#FFD700"))
```

'geom_smooth()' using formula = 'y ~ x'



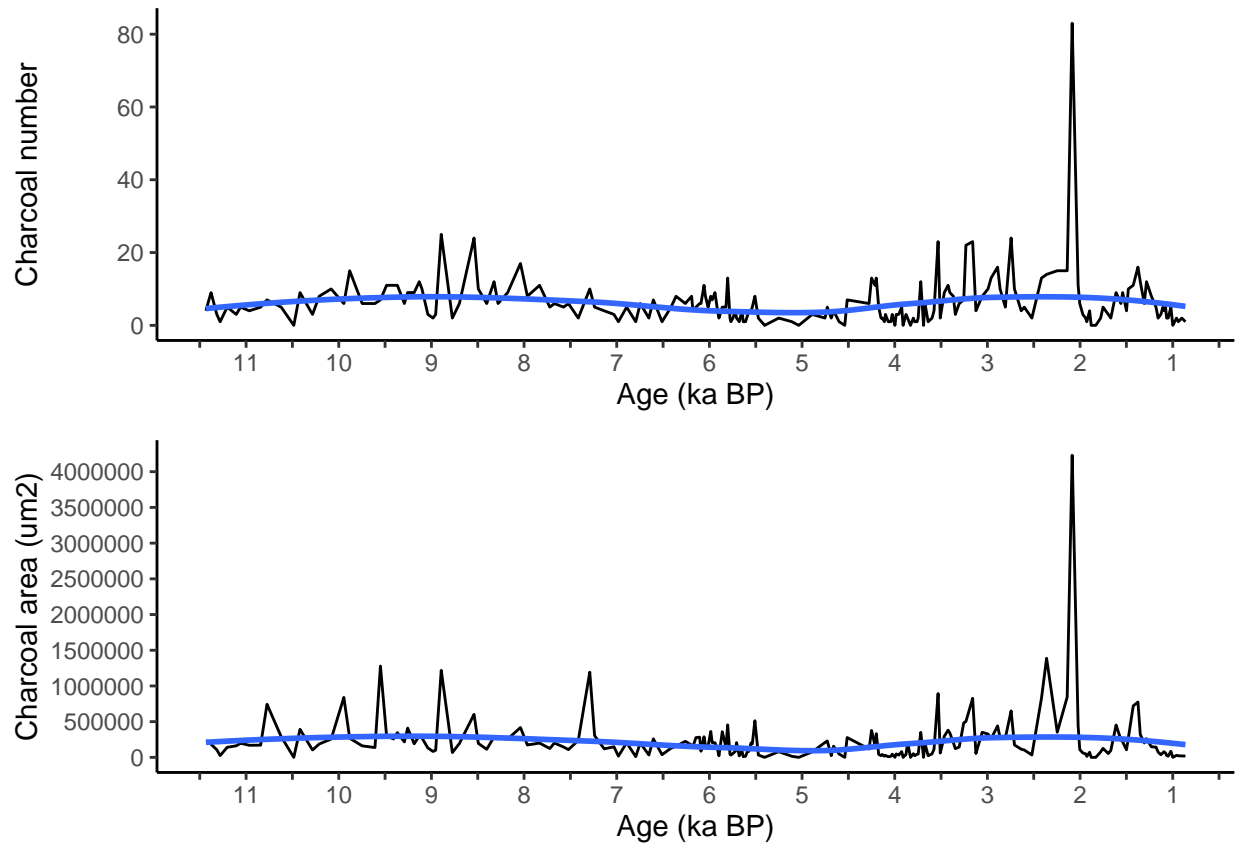
2.2.5. Charcoal number and area

```
CC.110 <- CharTotS.110.age %>%
  ggplot(aes(x = age/1000, y = CharCount)) +
  geom_line() +
  labs(x = "Age (ka BP)", y = "Charcoal number") +
  scale_x_reverse(breaks = seq(0.5, 12, by = 0.5),
                 labels = ifelse(seq(0.5, 12, by = 0.5) %% 1 == 0, as.character(seq(0.5, 12, by = 0.5)),
                                seq(0.5, 12, by = 0.5))) +
  scale_y_continuous(breaks = seq(min(CharTotS.110.age$CharCount), max(CharTotS.110.age$CharCount), by = 20)) +
  theme_classic() +
  geom_smooth(method = "loess", se = FALSE)

SCA.110 <- CharTotS.110.age %>%
  ggplot(aes(x = age/1000, y = sumCharArea)) +
  geom_line() +
  labs(x = "Age (ka BP)", y = "Charcoal area (um2)") +
  scale_x_reverse(breaks = seq(0.5, 12, by = 0.5),
                 labels = ifelse(seq(0.5, 12, by = 0.5) %% 1 == 0, as.character(seq(0.5, 12, by = 0.5)),
                                seq(0.5, 12, by = 0.5))) +
  scale_y_continuous(breaks = seq(min(CharTotS.110.age$sumCharArea), max(CharTotS.110.age$sumCharArea), by = 100)) +
  theme_classic() +
  geom_smooth(method = "loess", se = FALSE)

plot_grid(CC.110, SCA.110, ncol = 1, nrow = 2, align = "v")
```

```
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
```



3. CHAR (charcoal particles · cm⁻² · yr⁻¹)

3.1 Input data

```
head(mass.density) #dataframe with volume and mass information of charcoal samples
```

```
##      SampleId Density..wet.sample...g.cm3. Density..dry.sample...g.cm3.
## 1 TRAM-3U-19-20      1.443714      0.7731429
## 2 TRAM-3U-20-21      1.352385      0.7143846
## 3 TRAM-3U-21-22      1.312714      0.6542143
## 4 TRAM-3U-22-23      1.388818      0.6557727
## 5 TRAM-3U-23-24      1.175850      0.4662500
## 6 TRAM-3U-24-25      1.226727      0.4925000
##      Mass..g.
## 1      1.5025
## 2      1.6349
## 3      1.6743
## 4      1.5234
## 5      1.5846
## 6      1.5901
```

```
head(counting) #dataframe with sampleid and number of charcoal particles found in each sample
```

```
## # A tibble: 6 x 2
##   SampleId      CharCount
##   <chr>          <int>
## 1 TRAM-3U-100-101         3
## 2 TRAM-3U-102-103         3
## 3 TRAM-3U-103-104         1
## 4 TRAM-3U-104-105         1
## 5 TRAM-3U-105-106         3
## 6 TRAM-3U-106-107         1
```

```
head(CharTotS.110.age) #dataframe with sampleid and all charcoal metrics
```

```
##           SampleId CharCount sumCharArea meanCharWL medianCharWL age depth
## 1 TRAM-3U-100-101         3  47848.073  0.4349000      0.4969 3984  521
## 2 TRAM-3U-102-103         3  45394.326  0.4572667      0.3562 4027  524
## 3 TRAM-3U-103-104         1  13495.610  0.5032000      0.5032 4042  525
## 4 TRAM-3U-104-105         1   8588.116  0.4803000      0.4803 4070  527
## 5 TRAM-3U-105-106         3  28218.094  0.6418333      0.7019 4100  529
## 6 TRAM-3U-106-107         1  15949.358  0.5825000      0.5825 4111  530
```

```
sed.rate.charcoal <- tram20_depth_ages_sed_rate %>% #dataframe with sedimentation rate of charcoal samp
mutate(SampleId = str_replace_all(SampleId, "TRAM20-1B-", "TRAM-")) %>% #change SampleId so it is the
mutate(across(c(depth_diff, age_diff, sed_rate), ~ ifelse(row_number() == 1 & is.na(.x), .x[2], .x)))
filter(SampleId %in% mass.density$SampleId)
```

3.2. Calculate CHAR

First calculate CHAR based on the number of particles

```
CHAR_number <- counting %>%
  full_join(mass.density, by = "SampleId") %>%
  select(-3) %>%
  rename("Mass" = "Mass..g.",
         "Density" = "Density..dry.sample...g.cm3.") %>%
  mutate_at("CharCount", ~replace(., is.na(.), 0)) %>% #NAs in CharCount are samples with no charcoal,
  left_join(sed.rate.charcoal %>%
    select(SampleId, age, sed_rate, depth), by = "SampleId") %>%
  relocate("age", .after = "SampleId") %>%
  mutate(age = as.numeric(age)) %>%
  mutate("CHAR.tram" = (CharCount/Mass) * Density * sed_rate) %>%
  mutate_all(~replace(., is.na(.), 0))
```

Calculate CHAR based on the total area of charcoal particles

```
CHAR <- CharTotS.110.age %>%
  select(SampleId, sumCharArea, age, depth) %>%
  full_join(mass.density, by = "SampleId") %>%
  select(-"Density..wet.sample...g.cm3.") %>%
```

```

rename("Mass" = "Mass..g.",
       "Density" = "Density..dry.sample...g.cm3.") %>%
left_join(select(sed.rate.charcoal, SampleId, sed_rate), by = "SampleId") %>%
relocate("age", .after = "SampleId") %>%
mutate("CHAR.tram" = ((sumCharArea/Mass) * Density * sed_rate)/10e6) #mm2·cm-2·yr-1

#Correlation between CHAR and sedimentation rate
cor(CHAR$sed_rate, CHAR$CHAR.tram, method = 'pearson') #cor = 0.36 -> moderate correlation

## [1] 0.362594

```

3.3. Figure CHAR

```

pCHAR <- CHAR %>%
  ggplot(aes(x = age/1000, y = CHAR.tram)) +
  geom_col(width = 0.05) +
  ylab("CHAR (charcoal area mm2·cm-2·yr-1)") +
  xlab("Age (ka BP)") +
  scale_x_reverse(limits = c(max(CHAR$age)/1000, min(CHAR$age)/1000),
                 breaks = seq(0.5, 12, by = 0.5),
                 labels = ifelse(seq(0.5, 12, by = 0.5) %% 1 == 0, as.character(seq(0.5, 12, by = 0.5)),
                                "")) +
  theme_classic()

```

3.4. Figure 6

```

line_depths <- c(449, 497, 537, 620, 719, 765, 782, 822, 847)
zone_labels <- paste0("Z-", 10:1)
#add lines for coniss zones
add_linesy <- function(plot) {plot +
  geom_hline(yintercept = line_depths, color = "black", linetype = "dashed", size = 0.3, alpha = 0.7)}

fig6.1 <- CHAR %>%
  ggplot(aes(x = CHAR.tram, y = depth)) +
  geom_col(width = 1.2, orientation = "y") +
  xlab("CHAR \n(mm2·cm-2·yr-1)") +
  ylab("Depth (cm)") +
  scale_y_reverse(limits = c(880, 380),
                 breaks = seq(380, 880, by = 20) ,
                 labels = ifelse(seq(380, 880, by = 20) %% 40 == 0, as.character(seq(380, 880, by = 20)),
                                "")) +
  scale_x_continuous(expand = c(0.01, 0)) +
  theme_classic() +
  annotate("text", y = 800, x = 0, label = "No data", hjust = -1, color = "black")

fig6.1_coniss <- fig6.1 %>%add_linesy()

```

```

## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.

```

```
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```

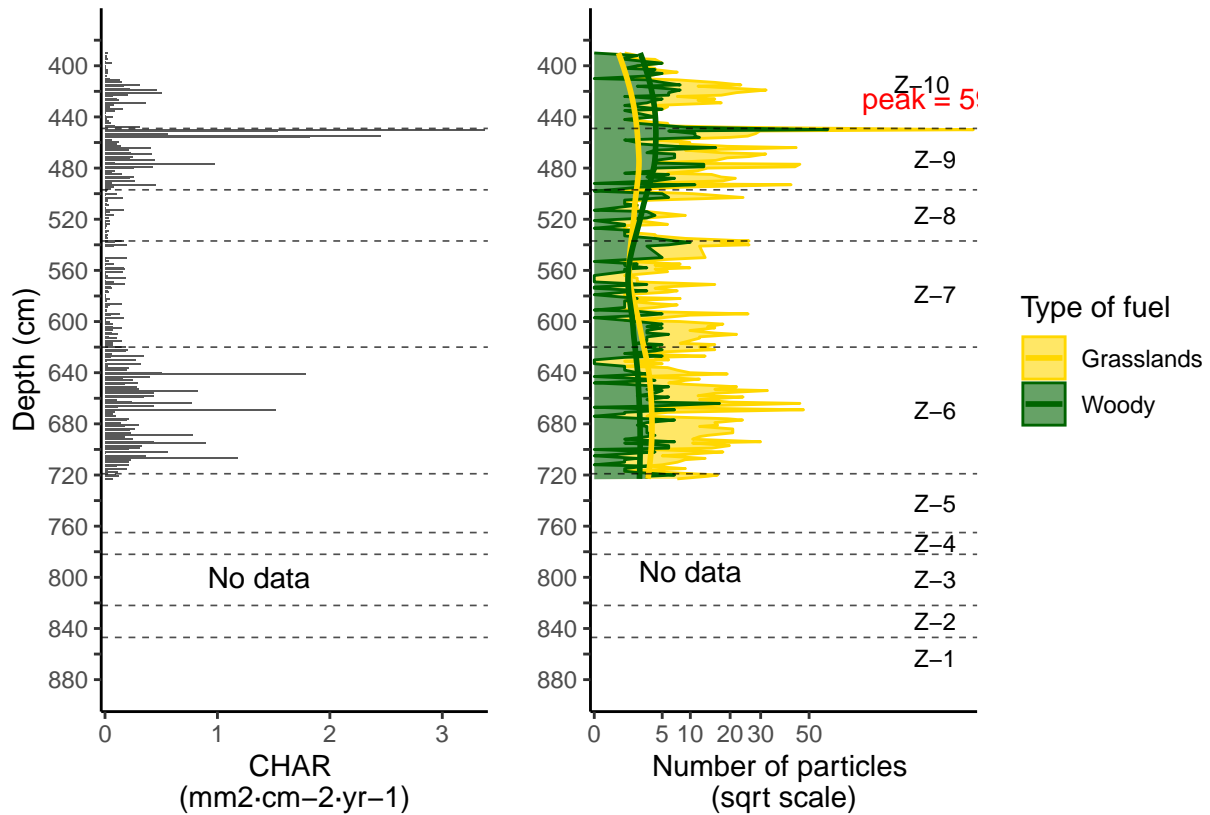
```
fig6.2 <- W.L.ratio %>%
  rename("Type of fuel" = "fuel") %>%
  ggplot(aes(x = count, y = depth, fill = `Type of fuel`, color = `Type of fuel`)) +
  geom_area(alpha = 0.6, position = "stack", orientation = "y") +
  geom_smooth(method = "loess", se = FALSE, size = 0.5, aes(color = `Type of fuel`), linewidth = 1, alpha = 0.5) +
  scale_x_continuous(trans = "sqrt", # transformation with squared root
                     breaks = c(0, 5, 10, 20, 30, 50),
                     expand = c(0.01, 0)) +
  labs(y = "", x = "Number of particles \n(sqrt scale)") +
  theme_classic() +
  scale_y_reverse(limits = c(880, 380),
                 breaks = seq(380, 880, by = 20) ,
                 labels = ifelse(seq(380, 880, by = 20) %% 40 == 0, as.character(seq(380, 880, by = 20)), "")) +
  scale_fill_manual(values = c("Woody" = "darkgreen", "Grasslands" = "#FFD700")) +
  scale_color_manual(values = c("Woody" = "darkgreen", "Grasslands" = "#FFD700")) +
  annotate("text", y = 450, x = 120, label = "peak = 59", vjust = -1, color = "red") +
  annotate("text", y = 820, x = 10, label = "No data", vjust = -1, color = "black")

fig6.2_coniss <- fig6.2 %>%
  add_linesy() +
  annotate("text", y = (c(380, line_depths) + c(line_depths, 880)) / 2, # Mid points
         x = Inf, label = zone_labels, hjust = 1.5, vjust = 0.5, size = 3, color = "black")

fig6.1_coniss/fig6.2_coniss +
  plot_layout(heights = c(1, 1), ncol = 2, nrow = 1, guides = "collect") +
  theme(plot.margin = margin(0, 0, 0, 0, "pt")) #save pdf A5
```

```
## Warning: 'position_stack()' requires non-overlapping x intervals.
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



4. Tramacastilla pollen

```
head(tram20_pollen) #Pollen counts
```

```
summary(tram20_pollen$`Terrestrial pollen sum`)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      313.0   379.0   417.0   425.8   452.0   604.0
```

```
sd(tram20_pollen$`Terrestrial pollen sum`)
```

```
## [1] 58.874
```

```
summary(tram20_pollen$`No of taxa`)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      13.00   21.00   27.00   27.02   33.00   47.00
```

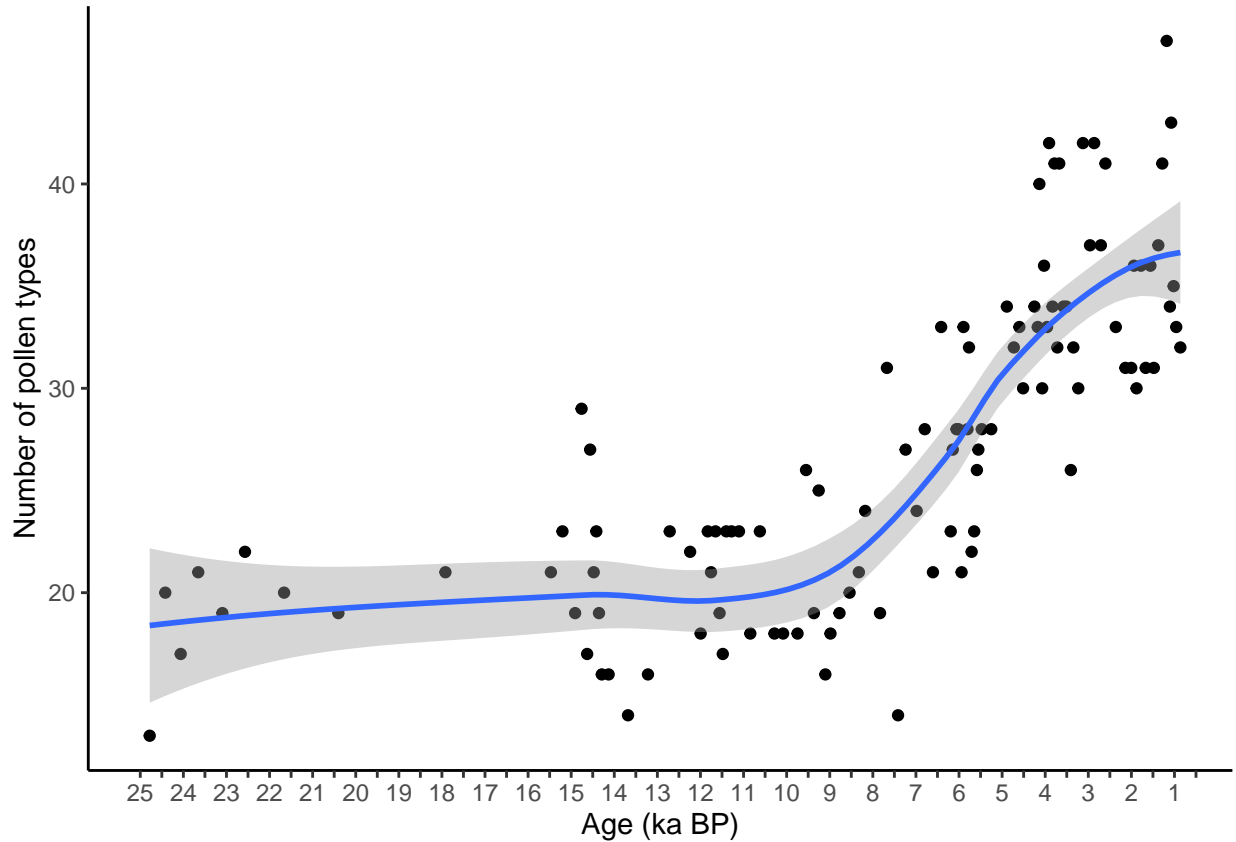
```
sd(tram20_pollen$`No of taxa`)
```

```
## [1] 7.86012
```

Plot pollen type-richness

```
tram20_pollen %>%  
  ggplot(aes(x = age/1000, y = `No of taxa`)) +  
  geom_point() +  
  scale_x_reverse(limits = c(25, 0.863),  
                 breaks = seq(0.5, 25, by = 0.5),  
                 labels = ifelse(seq(0.5, 25, by = 0.5) %% 1 == 0, as.character(seq(0.5, 25, by = 0.5)),  
                                "")) +  
  theme_classic() +  
  geom_smooth(method = "loess") +  
  labs(x = "Age (ka BP)",  
       y = "Number of pollen types")
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



Import pollen groups

```
pollen_group <- read.table("E:/Saco/IJP/1_data/Tramacastilla/pollen/pollen_groups.csv", header = TRUE,  
                           length(unique(pollen_group$pollen_group))
```

```
## [1] 14
```

```
length(unique(pollen_group$pollen_group_general))
```

```
## [1] 37
```

```
length(unique(pollen_group$pollen_group_discussion))
```

```
## [1] 21
```

Compute percentages for all taxa

```
tram20_pollen_long <- tram20_pollen %>%
  select(-c(`Evergreen trees`:Trees, Shrubs, Herbs, Hygrophytes, Hydrophytes:Spores, `Charcoal <150`)) %>%
  pivot_longer(cols = c("Abies":"Undetermined"),
    values_to = 'pollen_counts',
    names_to = 'Taxon') %>%
  mutate_at("pollen_counts", ~replace(., is.na(.), 0))

tram20_pollen_groups_long <- tram20_pollen_long %>% #long format with groups
  left_join(pollen_group, by = "Taxon")

tram20_pollen_groups_general_long_perc <- tram20_pollen_groups_long %>% #long format with general groups
  group_by(SampleId, pollen_group_general) %>%
  summarise(total_pollen = sum(pollen_counts, na.rm = TRUE)) %>%
  left_join(tram20_pollen_groups_long %>%
    select(SampleId, depth, age, Lycopodium, `Terrestrial pollen sum`) %>%
    distinct(), by = "SampleId") %>%
  mutate(percentage = (total_pollen/`Terrestrial pollen sum`)*100) %>% #compute group percentages in re
  select(SampleId, depth, age, Lycopodium, pollen_group_general, total_pollen, `Terrestrial pollen sum`
  ungroup()
```

'summarise()' has grouped output by 'SampleId'. You can override using the
'.groups' argument.

```
#wide format with pollen general groups and percentages
tram20_pollen_groups_general_perc <- tram20_pollen_groups_general_long_perc %>%
  select(age, depth, pollen_group_general, percentage) %>%
  pivot_wider(names_from = pollen_group_general, values_from = percentage, values_fill = 0) %>%
  select(age, depth, Abies, Pinus, Juniperus, Fagus, Betula, Corylus, `Deciduous Quercus`, Tilia, Ulmus,
    `Other deciduous trees`, Ephedra, Shrubs, Poaceae, `Cereal type`, Apiaceae, Artemisia, Cichorioideae,
    Chenopodiaceae, Helianthemum, Plantago, Ranunculaceae, Thalictrum, Filipendula, Potentilla, Scrophulariaceae,
    Urticaceae, Urtica, `Other herbs`, Hygrophytes, Hydrophytes, Undetermined,
    everything()) %>% #keep order
  arrange(age) %>%
  as.data.frame()

#taxa percentages
tram20_pollen_long_perc <- tram20_pollen_long %>%
  group_by(SampleId, Taxon) %>%
  mutate(percentage = (pollen_counts/`Terrestrial pollen sum`)*100) %>%
  ungroup()
```


5. Pollen Accumulation Rate (PAR)

```
head(tram20_density_pollen)
```

```
## # A tibble: 6 x 2
##   SampleId      Density
##   <chr>         <dbl>
## 1 TRAM20-1B-3U-19-20 0.773
## 2 TRAM20-1B-3U-22-23 0.656
## 3 TRAM20-1B-3U-24-25 0.492
## 4 TRAM20-1B-3U-27-28 0.575
## 5 TRAM20-1B-3U-29-30 0.537
## 6 TRAM20-1B-3U-32-33 0.615
```

```
head(pollen_samples_mass)
```

```
##           SampleId Mass
## 1 TRAM20-1B-5U-2-4-5 3.8
## 2 TRAM20-1B-5U-2-14-15 3.8
## 3 TRAM20-1B-5U-2-24-25 3.6
## 4 TRAM20-1B-5U-2-34-35 4.1
## 5 TRAM20-1B-5U-2-44-45 4.5
## 6 TRAM20-1B-5U-2-54-55 3.8
```

```
sed.rate.pollen <- tram20_depth_ages_sed_rate %>% #dataframe with sedimentation rate of charcoal sample
  mutate(across(c(depth_diff, age_diff, sed_rate), ~ ifelse(row_number() == 1 & is.na(.x), .x[2], .x)))
  filter(SampleId %in% tram20_pollen$SampleId)
```

```
tram20_PAR <- tram20_pollen %>%
  select(-c(`Evergreen trees`:Trees, Shrubs, Herbs, Hygrophytes, Hydrophytes:Spores, `Charcoal <150`)) %>%
  left_join(tram20_density_pollen, by = "SampleId") %>%
  left_join(pollen_samples_mass, by = "SampleId") %>%
  left_join(select(sed.rate.pollen, SampleId, sed_rate), by = "SampleId") %>%
  mutate(age = as.numeric(age)) %>%
  mutate_all(~replace(., is.na(.), 0)) %>%
  mutate(across(!c(SampleId, depth, age, `Lycopodium`, Density, Mass, sed_rate),
    ~ (. / ((`Lycopodium` * Mass)/41696)) * Density * sed_rate,
    .names = "PAR_{.col}")) %>% #41696 Lycopodium spores in each sample
  select(c(depth, age, PAR_Abies:`PAR_Undetermined`)) %>%
  rename_with(~ gsub("^PAR_", "", .x)) %>% #remove prefix
  select(where(~ sum(., na.rm = TRUE) > 0)) #only taxons that appear in these samples
```

Sum of terrestrial PAR in each sample to see relationship between terrestrial vegetation and CHAR

```
PAR_sum <- tram20_PAR %>% mutate(sum = rowSums(across(Abies:Viola))) %>%
  left_join(CHAR %>% select(age, CHAR.tram), by = "age")

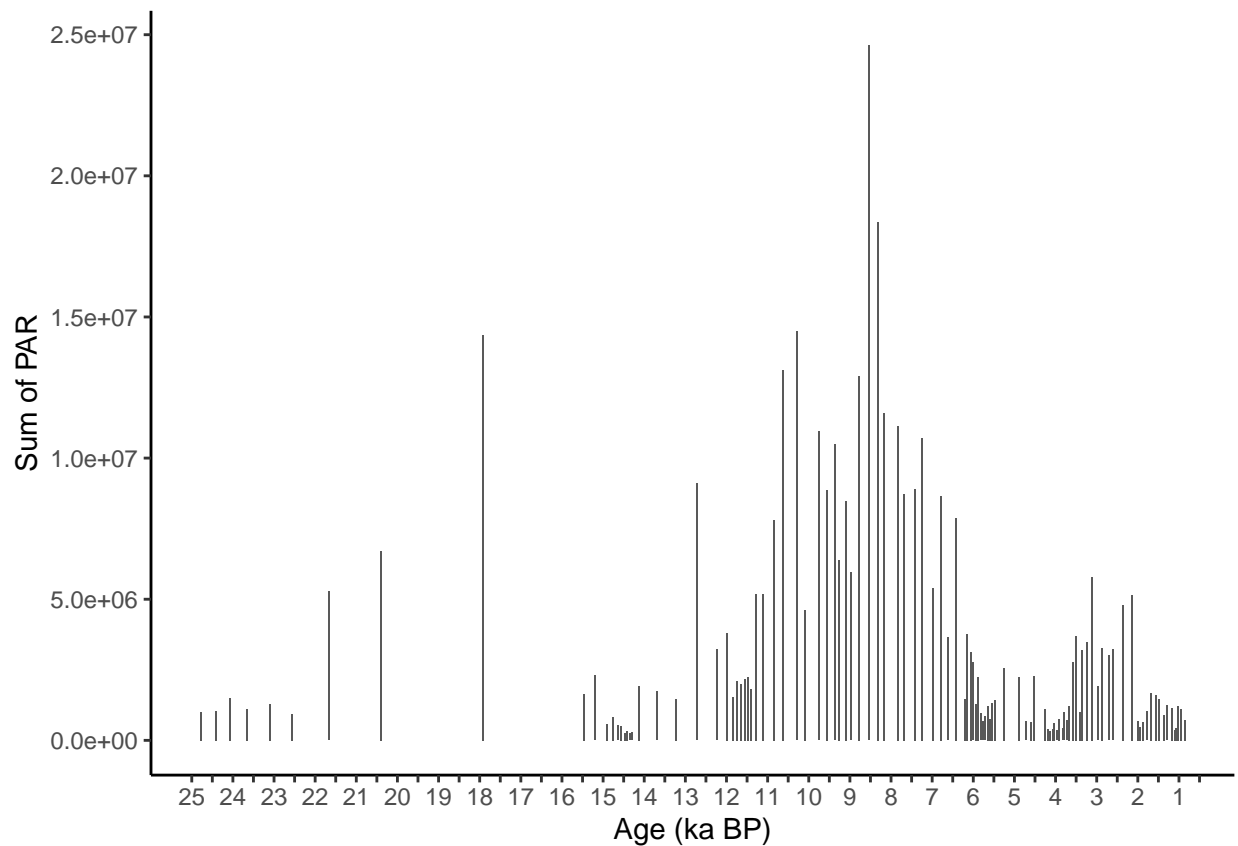
summary(PAR_sum$sum)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## 247223   925492  1730053  3616164  4971759 24615434
```

```
sd(PAR_sum$sum)
```

```
## [1] 4293638
```

```
PAR_sum %>% ggplot(aes(x = age/1000, y=sum)) +
  geom_col() +
  scale_x_reverse(breaks = seq(0.5, 25, by = 0.5),
    labels = ifelse(seq(0.5, 25, by = 0.5) %% 1 == 0, as.character(seq(0.5, 25, by = 0.5)),
  theme_classic() +
  labs(y = "Sum of PAR",
    x = "Age (ka BP)")
```



```
PAR.CHAR <- PAR_sum %>%
  na.omit()

#check if PAR and CHAR are correlated
cor.test(PAR.CHAR$sum, PAR.CHAR$CHAR.tram, method="spearman", exact = FALSE) #61% of the data is correlated

##
## Spearman's rank correlation rho
##
## data: PAR.CHAR$sum and PAR.CHAR$CHAR.tram
## S = 34355, p-value = 1.27e-09
## alternative hypothesis: true rho is not equal to 0
```

```
## sample estimates:
##      rho
## 0.6120734

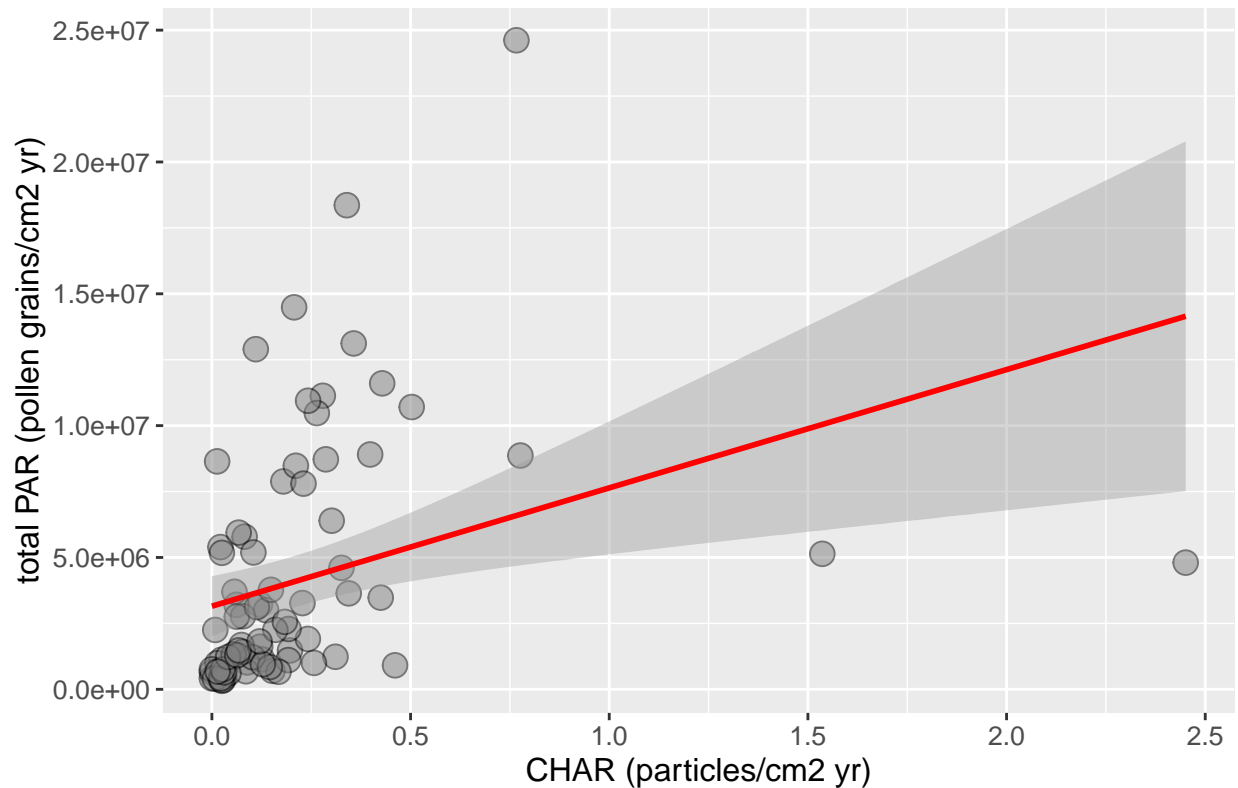
#check if that correlation is linnear or not
summary(lm(PAR.CHAR$sum~PAR.CHAR$CHAR.tram)) #9% is the amount of data linneary correlated

##
## Call:
## lm(formula = PAR.CHAR$sum ~ PAR.CHAR$CHAR.tram)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9342638 -2646973 -1875101  1911061 18024091
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3152829     572386   5.508 4.38e-07 ***
## PAR.CHAR$CHAR.tram 4486080     1467144   3.058 0.00304 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4394000 on 79 degrees of freedom
## Multiple R-squared:  0.1058, Adjusted R-squared:  0.09451
## F-statistic:  9.35 on 1 and 79 DF,  p-value: 0.003044

PAR.CHAR %>%
  ggplot(aes(x = CHAR.tram, y = sum)) +
  geom_point(shape = 21, fill = "gray50", color = "black", size = 4, alpha = 0.5) +
  geom_smooth(method = "lm", col = "red") +
  xlab("CHAR (particles/cm2 yr)") +
  ylab("total PAR (pollen grains/cm2 yr)") +
  ggtitle("PAR vs. CHAR") +
  theme(text = element_text(size = 12),
        plot.title = element_text(size = 16))

## 'geom_smooth()' using formula = 'y ~ x'
```

PAR vs. CHAR



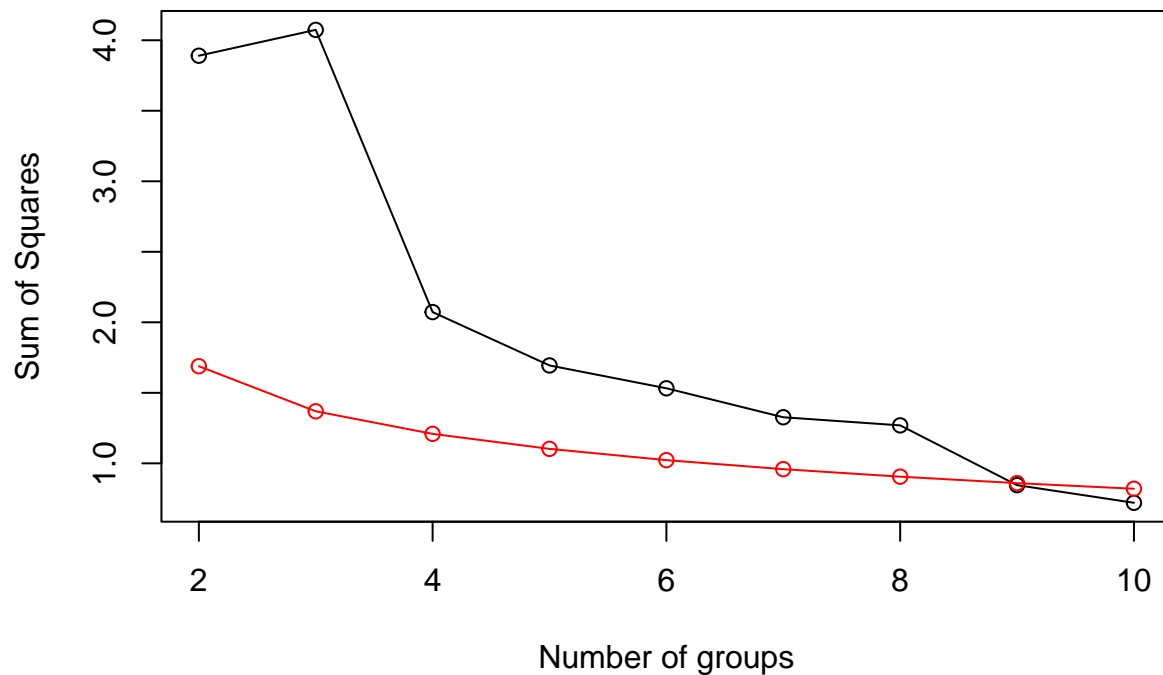
5.1. PAR grouped

```
tram20_PAR_groups <- tram20_PAR %>%
  pivot_longer(cols = Abies:Undetermined, names_to = "Taxon", values_to = "PAR_value") %>% #long format
  left_join(pollen_group, by = "Taxon") %>% #Include pollen groups
  group_by(depth, age, pollen_group_general) %>%
  summarise(PAR_sum = sum(PAR_value, na.rm = TRUE)) %>% #calculate PAR sum for each group
  pivot_wider(names_from = pollen_group_general, values_from = PAR_sum, values_fill = 0) %>% #back to w
  select(age, depth, Abies, Pinus, Juniperus, Fagus, Betula, Corylus, `Deciduous Quercus`, Tilia, Ulmus
    `Evergreen trees`, `Other deciduous trees`, Ephedra, Shrubs, Poaceae, `Cerealia type`, Apiacea
    Cichorioideae, Caryophyllaceae, Chenopodiaceae, Helianthemum, Plantago, Ranunculaceae, Thalicte
    Filipendula, Potentilla, Sanguisorba, Rumex, Scrophulariaceae, Urtica, Urticaceae, `Other herb
    Hygrophytes, Hydrophytes, Undetermined) %>% #keep order
  as.data.frame()
```

'summarise()' has grouped output by 'depth', 'age'. You can override using the
'.groups' argument.

5.1.1. Compute clustering

```
tram.dist.PARgroup <- vegdist(tram20_PAR_groups, method = "bray", binary = FALSE, diag = FALSE, upper = FALSE)
tram.chclust.PARgroup <- chclust(tram.dist.PARgroup, method="coniss")
bstick(tram.chclust.PARgroup)
```



```
#to see depths and ages of coniss zones
coniss.zones.10 <- cutree(tram.chclust.PARgroup, k = 10)
cluster_boundaries <- which(diff(coniss.zones.10)!=0)
coniss.depths <- tram20_PAR_groups$depth[cluster_boundaries] #depths = c(447, 495, 535, 618, 717, 763, 803, 843, 883, 923)
coniss.ages <- tram20_PAR_groups$age[cluster_boundaries] #ages = c(2003, 3576, 4173, 6192, 11280, 12713, 14180, 15650, 17120, 18590)
```

5.2. PAR grouped groups_discussion

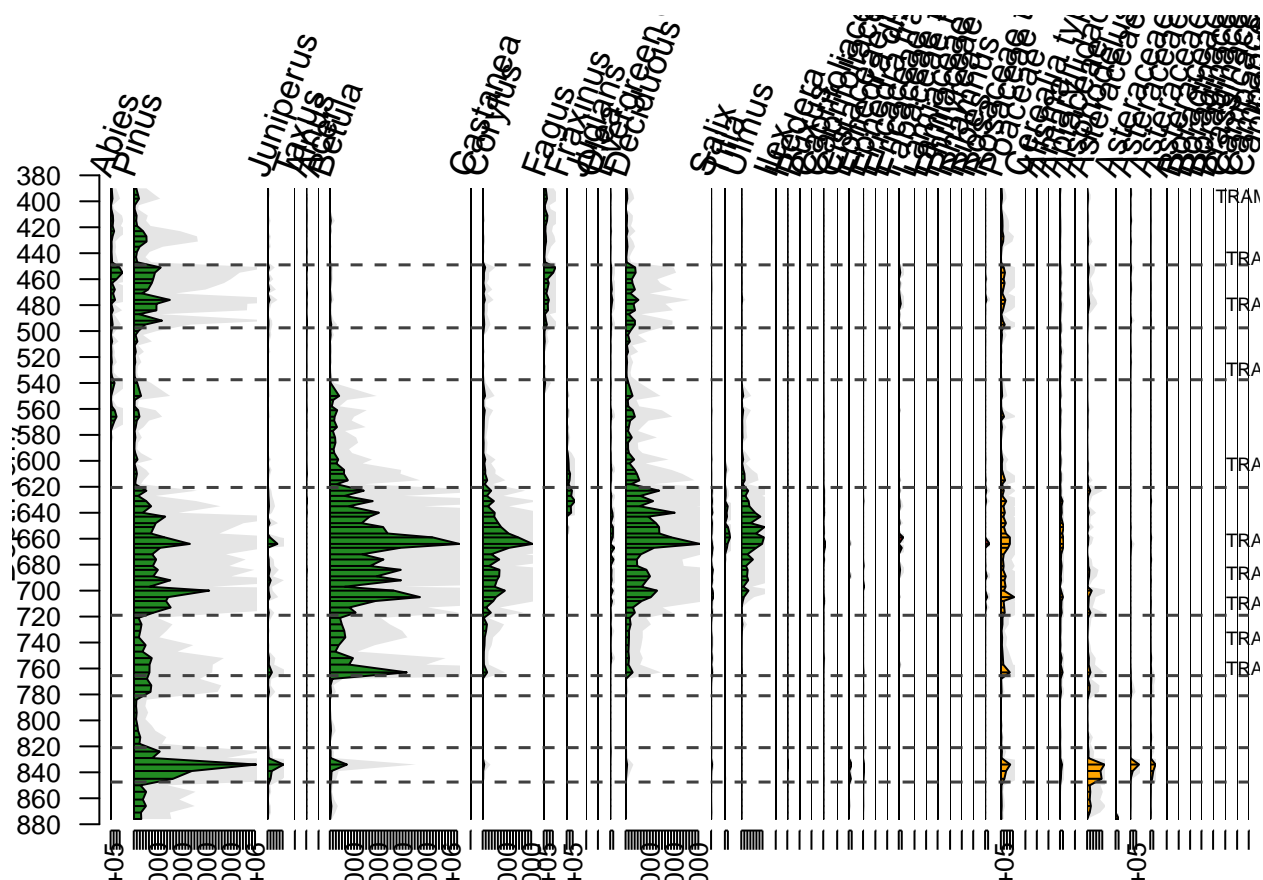
```
tram20_PAR_groups_discussion <- tram20_PAR %>%
  pivot_longer(cols = Abies:Undetermined, names_to = "Taxon", values_to = "PAR_value") %>% #long format
  left_join(pollen_group, by = "Taxon") %>% #Include pollen groups
  group_by(depth, age, pollen_group_discussion) %>%
  summarise(PAR_sum = sum(PAR_value, na.rm = TRUE)) %>% #calculate PAR sum for each group
  pivot_wider(names_from = pollen_group_discussion, values_from = PAR_sum, values_fill = 0) %>% #back to wide
  select(age, depth, Abies, Pinus, Juniperus, Fagus, Betula, Corylus, `Deciduous Quercus`, Tilia, Ulmus,
        `Evergreen trees`, `Other deciduous trees`, Shrubs, Poaceae, `Steppe taxa`, `Other herbs`, `Angiosperms`,
        Hygrophytes, Hydrophytes, Undetermined) %>% #keep order and exclude Ferns and Selaginella
  as.data.frame()
```

```
## 'summarise()' has grouped output by 'depth', 'age'. You can override using the
## '.groups' argument.
```

5.3. PAR all pollen types and aquatics - Supplementary Figure 5

Plot bar plot

```
zone_labels_pollen <- paste0("TRAMZ-", 10:1)
labels_points <- (c(380, line_depths) + c(line_depths, 880)) / 2 #for coniss labels
p.col.all.1 <- c(rep("forestgreen", times=18), rep("brown", times = 18), rep("orange", times=18))
pollen.plot.PAR1 <- strat.plot(tram20_PAR %>%
  select(Abies:`Caprifoliaceae` t Centranthus`),
  yvar=tram20_PAR %>%
  select(depth),
  ylabel="Depth (cm)", y.tks=seq(380, 880, 20), y.rev=TRUE,
  plot.line=FALSE, plot.poly=TRUE, plot.bar=TRUE, exag=TRUE,
  srt.xlabel=70, col.poly= p.col.all.1, col.poly.line = "black", col.bar =
  scale.percent=TRUE, x.pc.inc=200000, xSpace=0.009, x.pc.lab=TRUE, x.pc.or
addClustZone(pollen.plot.PAR1, tram.chclust.PARgroup, nZone=10, lwd=1.5, lty=2, col="grey25") #save pdf
text(x = par("usr")[1] + 1.15 * diff(par("usr")[1:2]),
  y = labels_points, labels = zone_labels_pollen, pos = 2, cex = 0.6, xpd = TRUE)
```

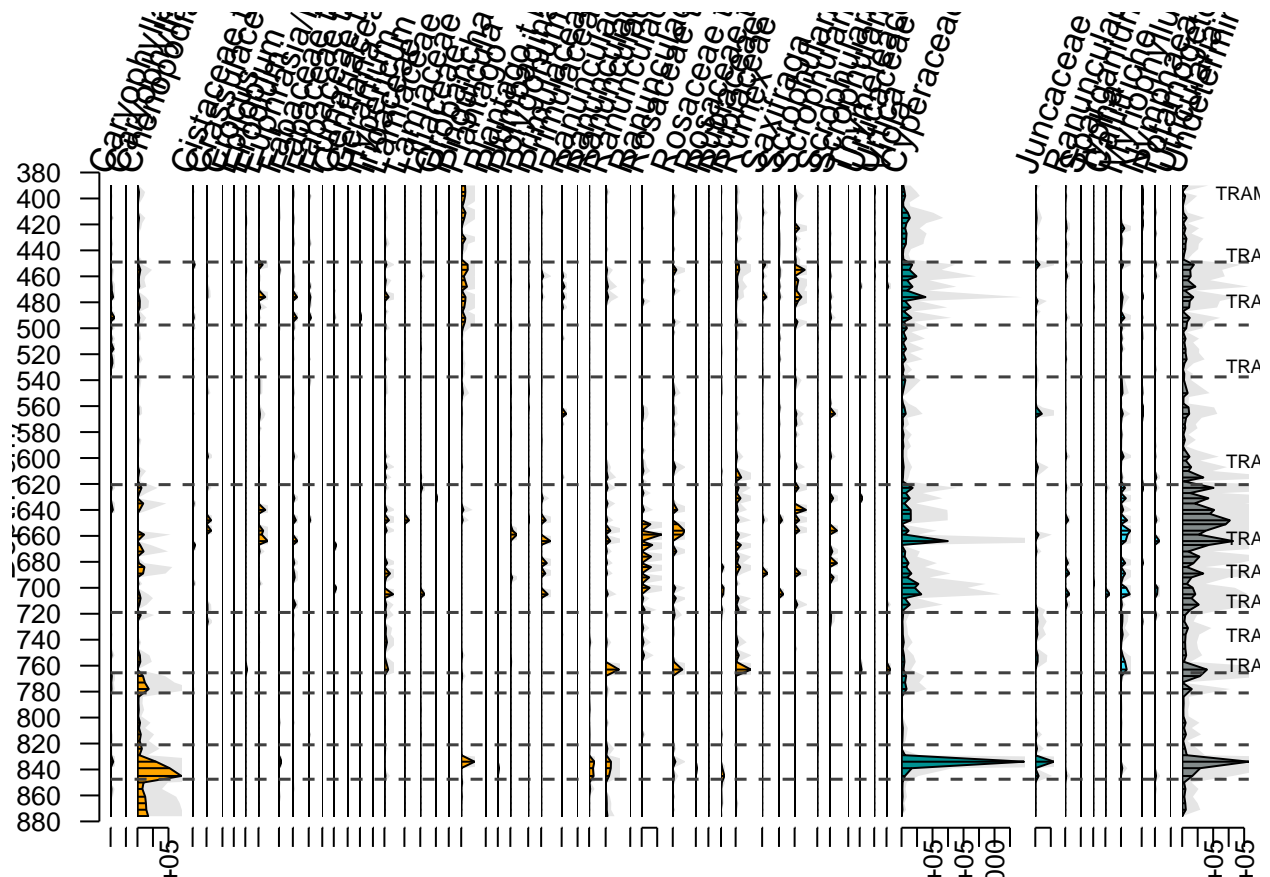


```
p.col.all.2 <- c(rep("orange", times=48), rep("#009292", times = 5), rep("#33E4FF", times = 5), "#838B8B")
pollen.plot.PAR2 <- strat.plot(tram20_PAR %>%
```

```

        select(Caryophyllaceae:Undetermined),
        yvar = tram20_PAR %>%
        select(depth),
        ylabel="Depth (cm)", y.tks=seq(380, 880, 20), y.rev=TRUE,
        plot.line=FALSE, plot.poly=TRUE, plot.bar=TRUE, exag = TRUE, srt.xlabel=
        col.poly = p.col.all.2, col.bar= "black",col.poly.line = "black", scale
        xSpace=0.009, x.pc.lab=TRUE, x.pc.omit0=TRUE, las=2)
addClustZone(pollen.plot.PAR2, tram.chclust.PARgroup, nZone=10, lwd=1.5, lty=2, col="grey25") #save pdf
text(x = par("usr")[1] + 1.15 * diff(par("usr")[1:2]),
      y = labels_points, labels = zone_labels_pollen, pos = 2, cex = 0.6, xpd = TRUE)

```



6. Pollen percentages with all taxa - Supplementary Figure 4

```

tram20_pollen_perc <- tram20_pollen_long_perc %>%
  select(depth, age, Taxon, percentage) %>%
  pivot_wider(names_from = Taxon,
              values_from = percentage)

# Figure 4 Supp Mat
pollen.plot.all.1 <- strat.plot(tram20_pollen_perc %>%
  select(Abies:`Caprifoliaceae` + Centranthus`),
  yvar = tram20_pollen_perc %>%

```



```

# Figure 4
figure4_Cap2 <- strat.plot(tram20_CHAR_pollen_groups_general_perc %>%
  select(sed_rate:CHAR) %>%
  rename("sedimentation \n rate (yrs/cm)" = "sed_rate"),
  yvar=tram20_CHAR_pollen_groups_general_perc %>%
  select(depth),
  ylabel="Depth (cm)", y.tks= seq(380, 880, 20), y.rev=TRUE,
  plot.line=TRUE, plot.poly=FALSE, plot.bar=FALSE,
  col.line = c.sed.rate.line.1, scale.percent=TRUE, exag = FALSE,
  srt.xlabel=70, cex.xlabel = 0.8, x.pc.inc=5, cex.axis=0.6,
  xSpace=0.005, x.pc.lab=TRUE, x.pc.omit0=TRUE, las=2)
figure4_Cap2 <- strat.plot(tram20_CHAR_pollen_groups_general_perc %>%
  select(sed_rate:CHAR),
  yvar=tram20_CHAR_pollen_groups_general_perc %>%
  select(depth),
  y.tks= FALSE, y.rev=TRUE,
  plot.line=FALSE, plot.poly=TRUE, plot.bar=TRUE, col.bar = c.pollen.bar.1, col
  col.poly.line = c.pollen.bar.1,
  scale.percent=TRUE, exag = ex.1, x.names="", x.axis=FALSE,
  xSpace=0.005, x.pc.lab=FALSE, x.pc.omit0=TRUE, las=2, add = TRUE)
figure4_Cap2 <- strat.plot(tram20_CHAR_pollen_groups_general_perc %>%
  select(sed_rate:CHAR),
  yvar=tram20_CHAR_pollen_groups_general_perc %>%
  select(depth),
  y.tks= FALSE, y.rev=TRUE,
  plot.line=FALSE, plot.poly=FALSE, plot.bar=TRUE, col.bar = c.char.bar.1, lwd
  scale.percent=TRUE, exag = ex.1, x.names="", x.axis=FALSE, x.pc.inc=5,
  xSpace=0.005, x.pc.lab=FALSE, x.pc.omit0=TRUE, las=2, add = TRUE)

text(x = par("usr")[1] + 1.05 * diff(par("usr")[1:2]),
  y = labels_points, labels = zone_labels_pollen, pos = 2, cex = 0.6, xpd = TRUE)

```


8. Temperature ~ CHAR

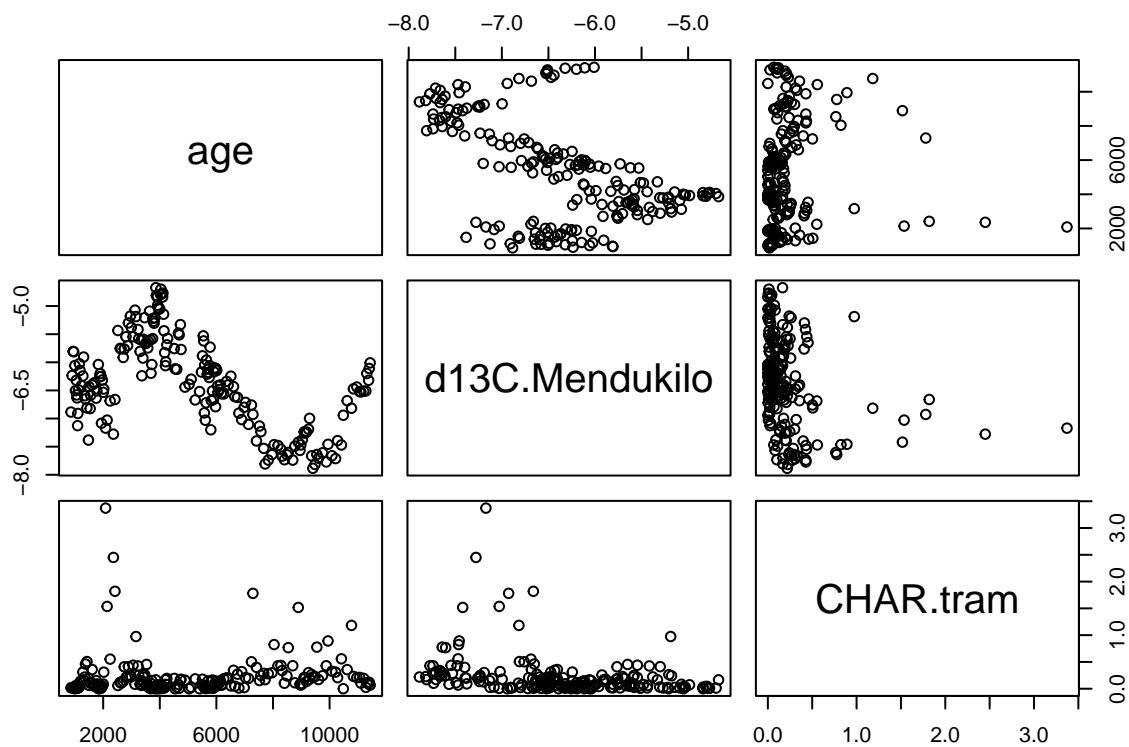
```
bins <- CHAR$age
head(Mendukilo) #temperature reconstruction from Mendukilo cave
```

```
##   age depth..mm. d13C.Mendukilo
## 1   0          0          -8.12
## 2  10          1          -9.50
## 3  10          2          -9.49
## 4  20          3          -9.03
## 5  20          4          -8.46
## 6  30          5          -8.44
```

```
Mendukilo_binning <- Mendukilo %>%
  mutate(binned_age = sapply(age, function(x) bins[which.min(abs(bins-x))]))
```

```
Mendukilo_binned <- Mendukilo_binning %>%
  group_by(binned_age) %>%
  summarise(d13C.Mendukilo = mean(d13C.Mendukilo, na.rm = TRUE)) %>%
  rename("age" = "binned_age") %>%
  ungroup() %>%
  as.data.frame()
```

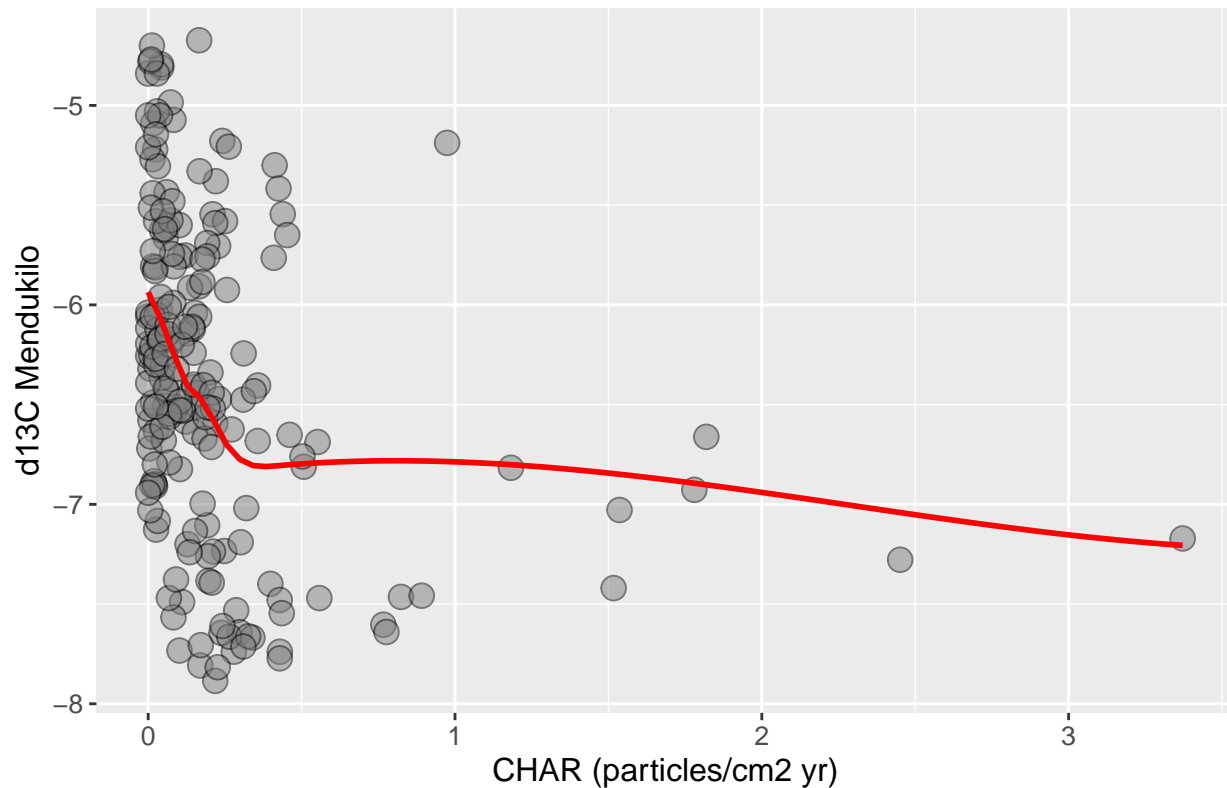
```
temp.char <- Mendukilo_binned %>%
  left_join(CHAR %>%
    select(age, CHAR.tram), by = "age")
pairs(temp.char)
```



```
temp.char %>%
  ggplot(aes(x = CHAR.tram, y = d13C.Mendukilo)) +
  geom_point(shape = 21, fill = "gray50", color = "black", size = 4, alpha = 0.5) +
  geom_smooth(method = "loess", se = FALSE, col = "red") +
  xlab("CHAR (particles/cm2 yr)") +
  ylab("d13C Mendukilo") +
  ggtitle("Temperature vs. CHAR") +
  theme(text = element_text(size = 12),
        plot.title = element_text(size = 16))
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```

Temperature vs. CHAR



```
temp.char.gls <- gls(d13C.Mendukilo ~ CHAR.tram, data = temp.char)
summary(temp.char.gls)
```

```
## Generalized least squares fit by REML
## Model: d13C.Mendukilo ~ CHAR.tram
## Data: temp.char
##      AIC      BIC    logLik
## 470.9609 480.8408 -232.4805
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept) -6.233022 0.0616125 -101.16489    0
## CHAR.tram   -0.588892 0.1373206  -4.28845    0
##
## Correlation:
##      (Intr)
## CHAR.tram -0.49
##
## Standardized residuals:
##      Min      Q1      Med      Q3      Max
## -2.00066567 -0.67755158 -0.03018005  0.72504162  2.17692699
##
## Residual standard error: 0.7613767
## Degrees of freedom: 201 total; 199 residual
```

```
cor(temp.char$d13C.Mendukilo, predict(temp.char.gls))^2 #R2 = 0.08
```

```
## [1] 0.08459781
```

9. XRF data

Import table with all composite depths

```
head(tram20_depths_XRF)
```

```
## # A tibble: 6 x 2
##   SampleId      composite_depth
##   <chr>          <dbl>
## 1 TRAM20-1B-1U-10.5      1.01
## 2 TRAM20-1B-1U-11      1.69
## 3 TRAM20-1B-1U-11.5    2.36
## 4 TRAM20-1B-1U-12      3.03
## 5 TRAM20-1B-1U-12.5    3.7
## 6 TRAM20-1B-1U-13      4.37
```

```
head(tram20_LECO_3U_4U_5U)
```

```
##           SampleId  TC   TN   TS   TIC   TOC   TOC/TN
## 1  TRAM20-1B-3U-2  1.95 0.190 0.0573 0.14060 1.80940 9.523158
## 2  TRAM20-1B-3U-4  2.51 0.229 0.0460 0.09386 2.41614 10.550830
## 3  TRAM20-1B-3U-6  2.17 0.229 0.0546 0.09527 2.07473 9.059956
## 4  TRAM20-1B-3U-8  2.18 0.214 0.0508 0.10260 2.07740 9.707477
## 5  TRAM20-1B-3U-10 2.51 0.248 0.0575 0.14810 2.36190 9.523790
## 6  TRAM20-1B-3U-12 2.31 0.230 0.0600 0.14360 2.16640 9.419130
```

```
head(tram20_LECO_1U_2U)
```

```
##           SampleId  TC   TN   TS   TIC   TOC   TOC/TN
## 1  TRAM20-1B-1U-10 1.38 0.212 0.0472 0.287 1.093 5.155660
## 2  TRAM20-1B-1U-12 1.32 0.231 0.0518 0.349 0.971 4.203463
## 3  TRAM20-1B-1U-14 1.56 0.240 0.0514 0.617 0.943 3.929167
## 4  TRAM20-1B-1U-16 1.15 0.227 0.0471 0.448 0.702 3.092511
## 5  TRAM20-1B-1U-18 1.21 0.216 0.0421 0.161 1.049 4.856481
## 6  TRAM20-1B-1U-20 1.32 0.231 0.0518 0.176 1.144 4.952381
```

```
tram20_LECO <- bind_rows(tram20_LECO_1U_2U, tram20_LECO_3U_4U_5U)
```

```
head(tram20_MS_IPE)
```

```
## # A tibble: 6 x 2
##   SampleId      MS
##   <chr>      <dbl>
## 1 TRAM20-1B-1U-0 -0.03
## 2 TRAM20-1B-1U-1 -0.06
```



```
## 3 TRAM20-1B-1U-2 -0.1
## 4 TRAM20-1B-1U-3 0.07
## 5 TRAM20-1B-1U-4 0.4
## 6 TRAM20-1B-1U-5 0.99
```

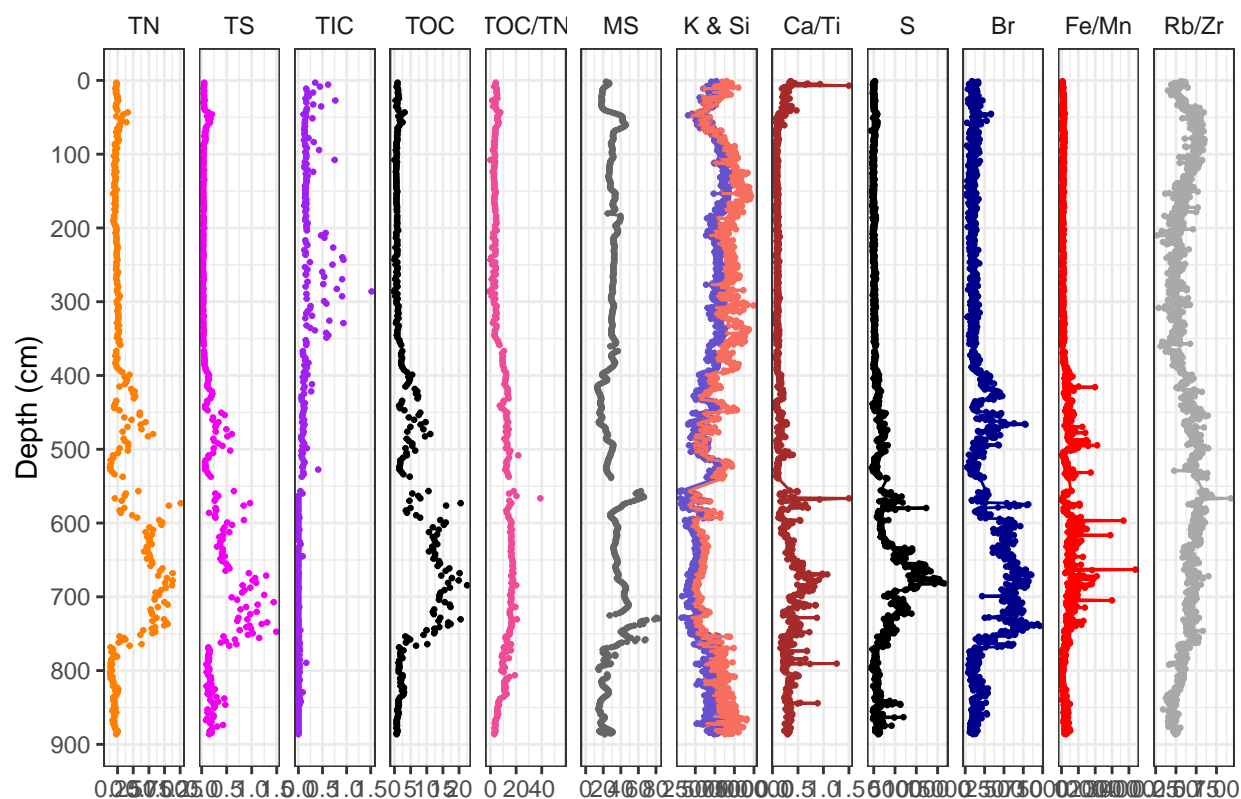
```
head(tram20_XRF_LECO_MS) #to facilitate visualisation peaks in Ca/Ti with a value >1.5 will be reduced
```

```
## # A tibble: 6 x 15
##   SampleId      composite_depth    TN      TS      TIC      TOC 'TOC/TN'      MS      K
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1 TRAM20-1B-1~      1.01 NA      NA      NA      NA      NA      NA      52022
## 2 TRAM20-1B-1~      1.69 NA      NA      NA      NA      NA      23.2  45600
## 3 TRAM20-1B-1~      2.36 NA      NA      NA      NA      NA      NA      46548
## 4 TRAM20-1B-1~      3.03 0.231 0.0518 0.349 0.971    4.20  26.5  51030
## 5 TRAM20-1B-1~      3.7  NA      NA      NA      NA      NA      NA      52587
## 6 TRAM20-1B-1~      4.37 NA      NA      NA      NA      NA      26.8  48977
## # i 6 more variables: Si <dbl>, 'Ca/Ti' <dbl>, S <dbl>, Br <dbl>,
## #   'Fe/Mn' <dbl>, 'Rb/Zr' <dbl>
```

```
tram20_XRF_LECO_MS %>%
  pivot_longer(cols = c("TN":"Rb/Zr"),
    values_to = 'values',
    names_to = 'elements') %>%
  mutate(facet_group = case_when(elements == "TN" ~ "TN",
    elements == "TS" ~ "TS",
    elements == "TIC" ~ "TIC",
    elements == "TOC" ~ "TOC",
    elements == "TOC/TN" ~ "TOC/TN",
    elements == "MS" ~ "MS",
    elements %in% c("K", "Si") ~ "K & Si",      # i want to have K and Si
    elements == "S" ~ "S",
    elements == "Br" ~ "Br",
    elements == "Ca/Ti" ~ "Ca/Ti",
    elements == "Fe/Mn" ~ "Fe/Mn",
    elements == "Rb/Zr" ~ "Rb/Zr"),
    facet_group = fct_relevel (facet_group, "TN", "TS", "TIC", "TOC", "TOC/TN", "MS", "K & Si", "Ca/Ti", "Fe/Mn", "Rb/Zr"))
ggplot(aes(x = values, y = composite_depth, fill = elements)) +
  geom_point(aes(color = elements), size = 0.5) + geom_lineh(aes(color = elements)) +
  scale_y_reverse(breaks = seq(0, 900, 100)) +
  scale_x_continuous() +
  facet_geochem_gridh(vars(facet_group)) +
  labs(y = "Depth (cm)",
    x = "") +
  theme_paleo() +
  theme(legend.position = "none") +
  scale_color_manual(values = c("darkblue", "brown", "red", "#6551CC", "#666666", "darkgrey",
    "black", "#F76D5E", "purple", "#FF7F00", "black", "#EE4C97", "#F100F1"))
```

```
## Warning: Removed 6102 rows containing missing values or values outside the scale range
## ('geom_point()').
```

```
## Warning: Removed 770 rows containing missing values or values outside the scale range
## ('geom_lineh()').
```



9.1. Figure 3 - XRF data with MS, LECO, XRF

Ordered: MS, K & Si, Ca/Ti, S, TIC, TOC, Inc/Coh, Br, Fe/Mn, Rb/Zr

```
tram20_XRF_LECO_MS %>%
  select(SampleId, composite_depth, MS) %>%
  pivot_longer(cols = "MS",
               values_to = 'values',
               names_to = 'elements') %>%
  na.omit() %>%
  ggplot(aes(x = values, y = composite_depth, fill = elements)) +
  geom_lineh(aes(color = elements)) +
  scale_y_reverse(limits = c(900, 0), breaks = seq(0, 900, 100), labels = seq(0, 900, 100)) +
  scale_x_continuous(n.breaks = 3) +
  facet_geochem_gridh(vars(elements)) +
  theme_minimal_grid() +
  theme(legend.position = "none",
        axis.title.y = element_blank(),
        axis.title.x = element_blank(),
        axis.text.x = element_text(angle = 45, hjust = 1),
        plot.margin = margin(0, 0, 0, 0, "cm")) +
  scale_color_manual(values = "#666666") +
  tram20_XRF_LECO_MS %>%
  select(c(SampleId, composite_depth, K, Si, S, `Ca/Ti`)) %>%
  pivot_longer(cols = c("K": "Ca/Ti"),
```

```

      values_to = 'values',
      names_to = 'elements') %>%
mutate(facet_group = case_when(elements %in% c("K", "Si") ~ "K & Si",      # i want to have K and Si
                             elements == "S" ~ "S",
                             elements == "Ca/Ti" ~ "Ca/Ti"),
      facet_group = fct_relevel (facet_group, "K & Si", "Ca/Ti", "S")) %>%
ggplot(aes(x = values, y = composite_depth, fill = elements)) +
geom_lineh(aes(color = elements)) +
scale_y_reverse(breaks = seq(0, 900, 100)) +
scale_x_continuous(n.breaks = 3) +
facet_geochem_gridh(vars(facet_group)) +
theme_minimal_grid() +
theme(legend.position = "none",
      axis.title.x = element_blank(),
      axis.ticks.y = element_blank(),
      axis.text.y = element_blank(),
      axis.title.y = element_blank(),
      axis.text.x = element_text(angle = 45, hjust = 1),
      plot.margin = margin(0, 0, 0, 0, "cm")) +
scale_color_manual(values = c("brown", "#6551CC", "purple", "#FF7F00")) +
tram20_XRF_LECO_MS %>%
select(SampleId, composite_depth, TIC, TOC) %>%
pivot_longer(cols = c("TIC":"TOC"),
             values_to = 'values',
             names_to = 'elements') %>%
mutate(elements = fct_relevel (elements, "TIC", "TOC")) %>%
na.omit() %>%
ggplot(aes(x = values, y = composite_depth, fill = elements)) +
geom_lineh(aes(color = elements)) +
scale_y_reverse(limits = c(900, 0), breaks = seq(0, 900, 100), labels = seq(0, 900, 100)) +
scale_x_continuous(n.breaks = 3) +
facet_geochem_gridh(vars(elements)) +
theme_minimal_grid() +
theme(legend.position = "none",
      axis.title.y = element_blank(),
      axis.ticks.y = element_blank(),
      axis.text.y = element_blank(),
      axis.title.x = element_blank(),
      axis.text.x = element_text(angle = 45, hjust = 1),
      plot.margin = margin(0, 0, 0, 0, "cm")) +
scale_color_manual(values = c("#EE4C97", "#F100F1")) +
tram20_XRF_LECO_MS %>%
select(c(SampleId, composite_depth, Br, `Fe/Mn`, `Rb/Zr`)) %>%
pivot_longer(cols = c("Br":"Rb/Zr"),
             values_to = 'values',
             names_to = 'elements') %>%
mutate(facet_group = case_when(elements == "Br" ~ "Br",
                             elements == "Fe/Mn" ~ "Fe/Mn",
                             elements == "Rb/Zr" ~ "Rb/Zr"),
      facet_group = fct_relevel (facet_group, "Br", "Fe/Mn", "Rb/Zr")) %>%
ggplot(aes(x = values, y = composite_depth, fill = elements)) +
geom_lineh(aes(color = elements)) +
scale_y_reverse(breaks = seq(0, 900, 100)) +

```

```

scale_x_continuous(n.breaks = 3) +
facet_geochem_gridh(vars(facet_group)) +
theme_minimal_grid() +
theme(legend.position = "none",
      axis.title.x = element_blank(),
      axis.text.y = element_blank(),
      axis.ticks.y = element_blank(),
      axis.title.y = element_blank(),
      axis.text.x = element_text(angle = 45, hjust = 1),
      plot.margin = margin(0, 0, 0, 0, "cm")) +
scale_color_manual(values = c("darkblue", "red", "#FF7F00")) +
plot_layout(widths = c(1, 3, 2, 4))

```

```

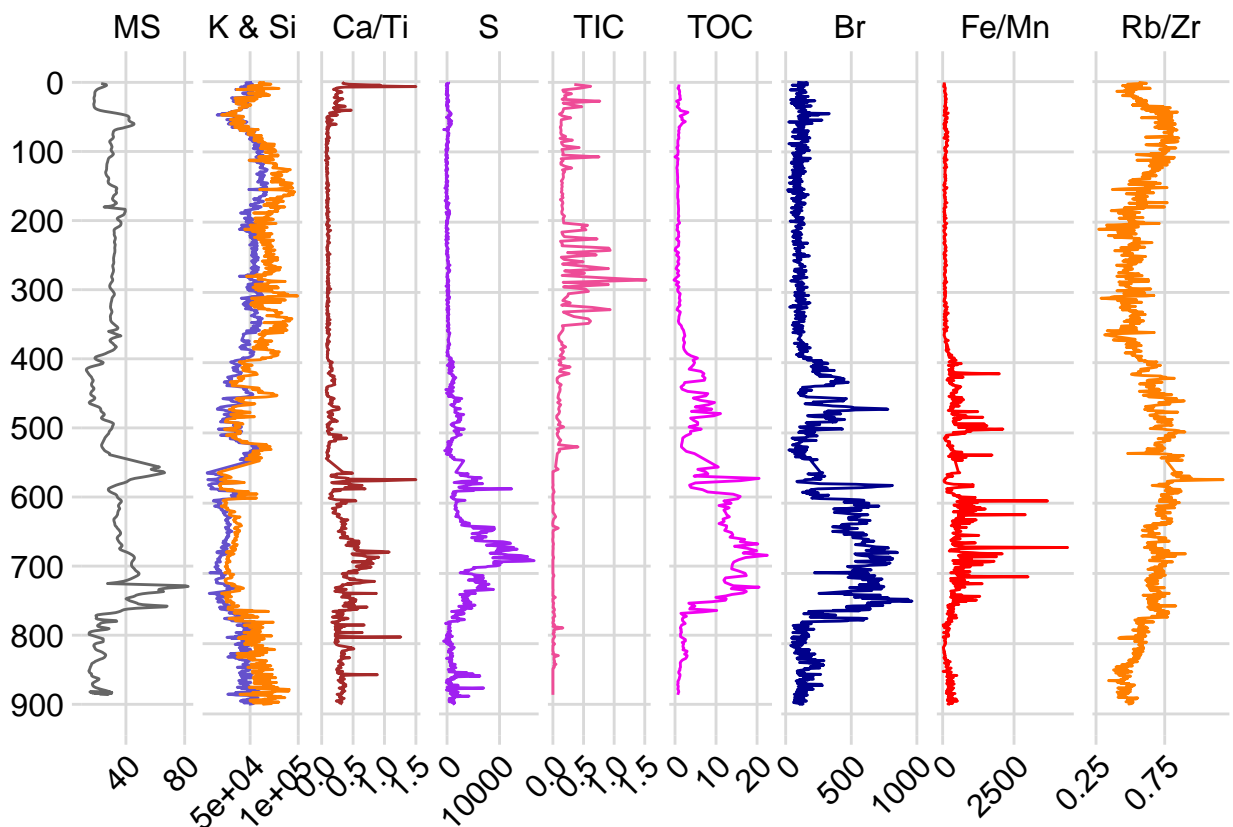
## Warning: Removed 232 rows containing missing values or values outside the scale range
## ('geom_lineh()').

```

```

## Warning: Removed 174 rows containing missing values or values outside the scale range
## ('geom_lineh()').

```



9.2. Supplementary Figure - all XRF, MS, LECO data

```
head(tram20_XRF_LECO_MS_SuppFig)
```

```
## # A tibble: 6 x 19
##   SampleId composite_depth  TN      TS      TIC      TOC      MS      K      Si      Ca
##   <chr>          <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 TRAM20-1~      1.01 NA      NA      NA      NA      NA  52022 65246 9921
## 2 TRAM20-1~      1.69 NA      NA      NA      NA      NA  45600 59359 8795
## 3 TRAM20-1~      2.36 NA      NA      NA      NA      NA  46548 61599 8140
## 4 TRAM20-1~      3.03 0.231 0.0518 0.349 0.971 26.5 51030 67768 9881
## 5 TRAM20-1~      3.7  NA      NA      NA      NA      NA  52587 71651 11477
## 6 TRAM20-1~      4.37 NA      NA      NA      NA      NA  48977 61069 17485
## # i 9 more variables: Ti <dbl>, S <dbl>, Br <dbl>, Fe <dbl>, Mn <dbl>,
## #   Rb <dbl>, Zr <dbl>, Pb <dbl>, Sr <dbl>
```

```
tram20_XRF_LECO_MS_SuppFig %>%
  select(SampleId, composite_depth, MS) %>%
  pivot_longer(cols = "MS",
               values_to = 'values',
               names_to = 'elements') %>%
  na.omit() %>%
  ggplot(aes(x = values, y = composite_depth, fill = elements)) +
  geom_lineh(aes(color = elements)) +
  scale_y_reverse(limits = c(900, 0), breaks = seq(0, 900, 100), labels = seq(0, 900, 100)) +
  scale_x_continuous(n.breaks = 3) +
  facet_geochem_gridh(vars(elements)) +
  theme_minimal_grid() +
  theme(legend.position = "none",
        axis.title.y = element_blank(),
        axis.title.x = element_blank(),
        axis.text.x = element_text(angle = 45, hjust = 1),
        plot.margin = margin(0, 0, 0, 0, "cm")) +
  scale_color_manual(values = "#666666") +
  tram20_XRF_LECO_MS_SuppFig %>%
  select(SampleId, composite_depth, TN, TS, TIC, TOC) %>%
  pivot_longer(cols = c("TN":"TOC"),
               values_to = 'values',
               names_to = 'elements') %>%
  mutate(elements = fct_relevel(elements, "TN", "TS", "TIC", "TOC")) %>%
  na.omit() %>%
  ggplot(aes(x = values, y = composite_depth, fill = elements)) +
  geom_lineh(aes(color = elements)) +
  scale_y_reverse(limits = c(900, 0), breaks = seq(0, 900, 100), labels = seq(0, 900, 100)) +
  scale_x_continuous(n.breaks = 3) +
  facet_geochem_gridh(vars(elements)) +
  theme_minimal_grid() +
  theme(legend.position = "none",
        axis.title.y = element_blank(),
        axis.ticks.y = element_blank(),
        axis.text.y = element_blank(),
        axis.title.x = element_blank(),
        axis.text.x = element_text(angle = 45, hjust = 1),
        plot.margin = margin(0, 0, 0, 0, "cm")) +
```

```

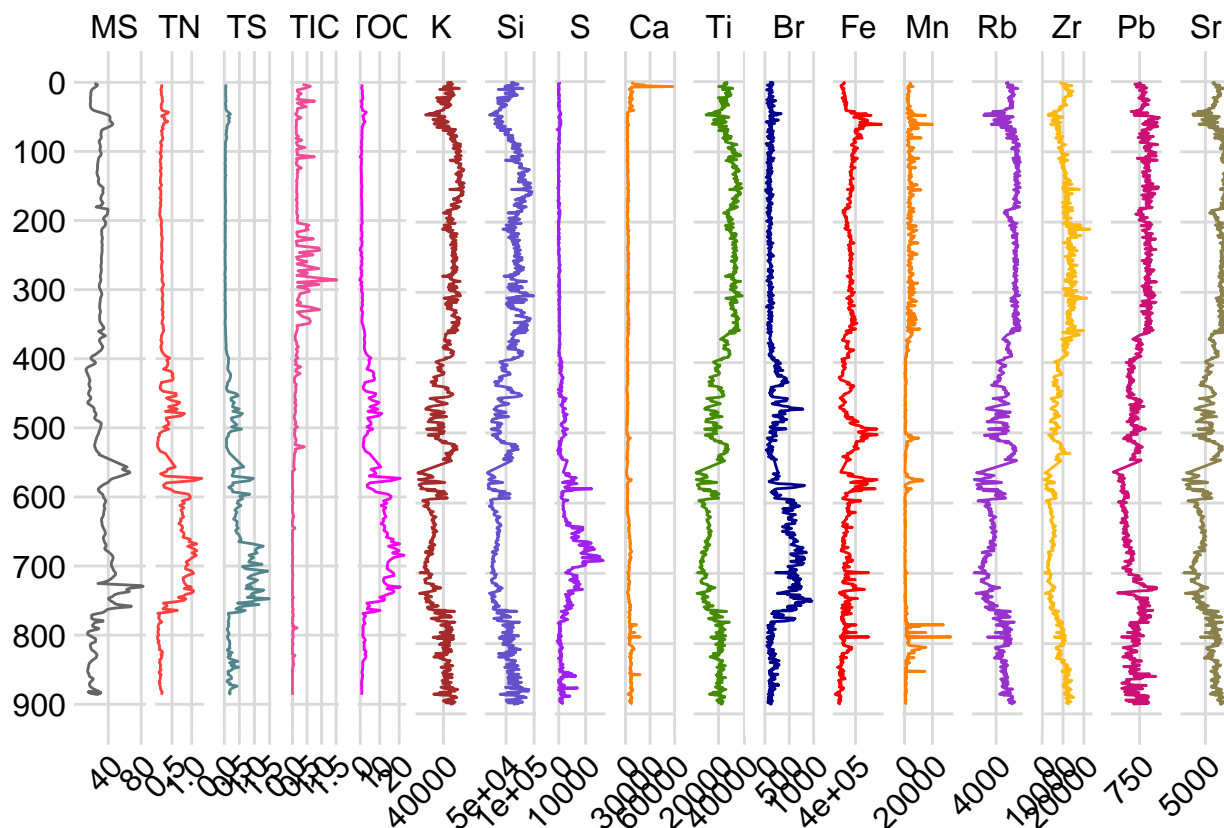
scale_color_manual(values = c("#FF4040", "#53868B", "#EE4C97", "#F100F1")) +
tram20_XRF_LECO_MS_SuppFig %>%
select(c(SampleId, composite_depth, K, Si, Ca, Ti, S, Br, Fe, Mn, Rb, Zr, Pb, Sr)) %>%
pivot_longer(cols = c("K":"Sr"),
              values_to = 'values',
              names_to = 'elements') %>%
mutate(elements = fct_relevel (elements, "K", "Si", "S", "Ca", "Ti", "Br", "Fe", "Mn", "Rb", "Zr", "Pb", "Sr"))
ggplot(aes(x = values, y = composite_depth, fill = elements)) +
geom_lineh(aes(color = elements)) +
scale_y_reverse(breaks = seq(0, 900, 100)) +
scale_x_continuous(n.breaks = 3) +
facet_geochem_gridh(vars(elements)) +
theme_minimal_grid() +
theme(legend.position = "none",
      axis.title.x = element_blank(),
      axis.text.y = element_blank(),
      axis.ticks.y = element_blank(),
      axis.title.y = element_blank(),
      axis.text.x = element_text(angle = 45, hjust = 1),
      plot.margin = margin(0, 0, 0, 0, "cm")) +
scale_color_manual(values = c("brown", "#6551CC", "purple", "#FF7F00", "#458B00", "darkblue",
                              "red", "#FF7F00", "#9932CC", "#FFB90F", "#CD1076", "#8B814C")) +
plot_layout(widths = c(1, 4, 13))

```

```

## Warning: Removed 696 rows containing missing values or values outside the scale range
## ('geom_lineh()').

```



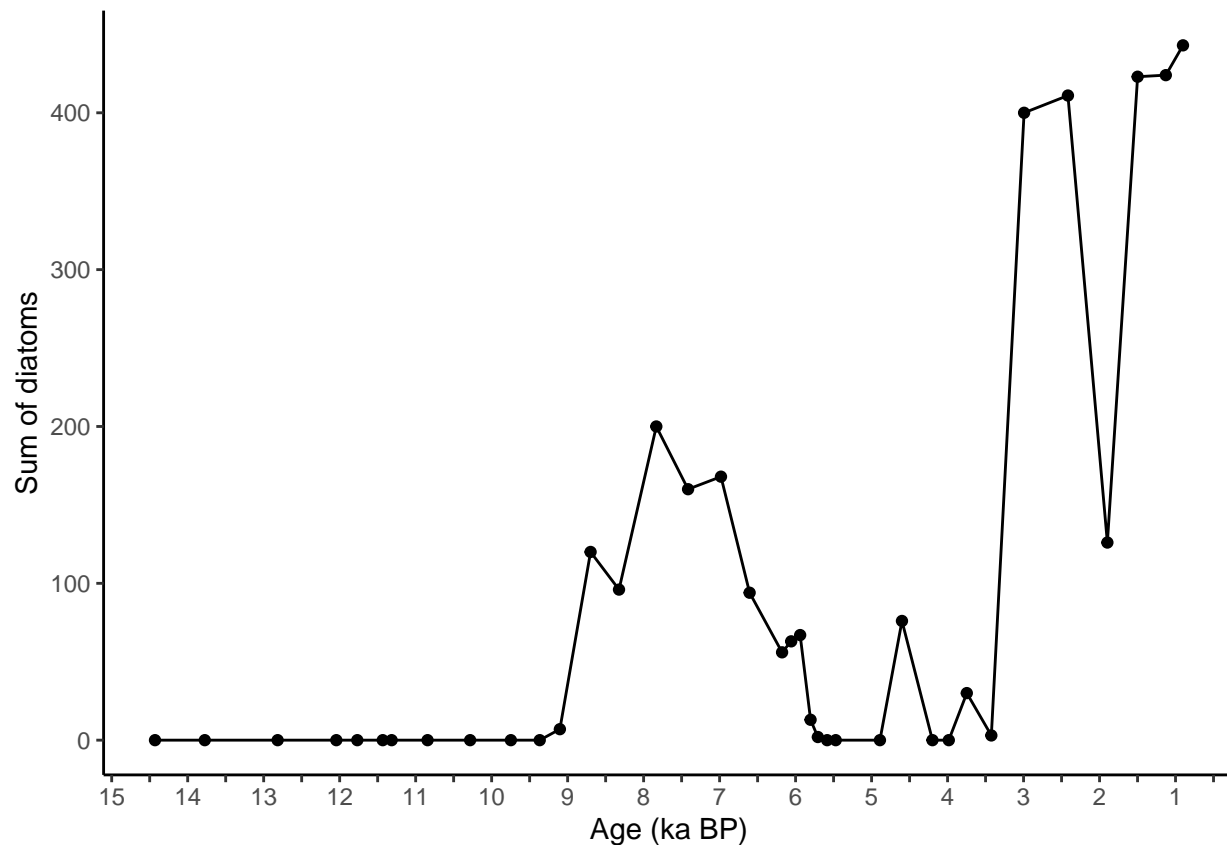
10. Diatoms

```
head(tram20_diatoms)
```

```
#diatoms accumulation
summary(tram20_diatoms$sum_diatoms)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      0.00   0.00    7.00   91.41 120.00  443.00
```

```
tram20_diatoms %>%
  ggplot(aes(x = age/1000, y = sum_diatoms)) +
  geom_point() +
  geom_line() +
  scale_x_reverse(breaks = seq(0.5, 15, by = 0.5),
                 labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)),
                                "")) +
  theme_classic() +
  labs(y = "Sum of diatoms",
       x = "Age (ka BP)")
```

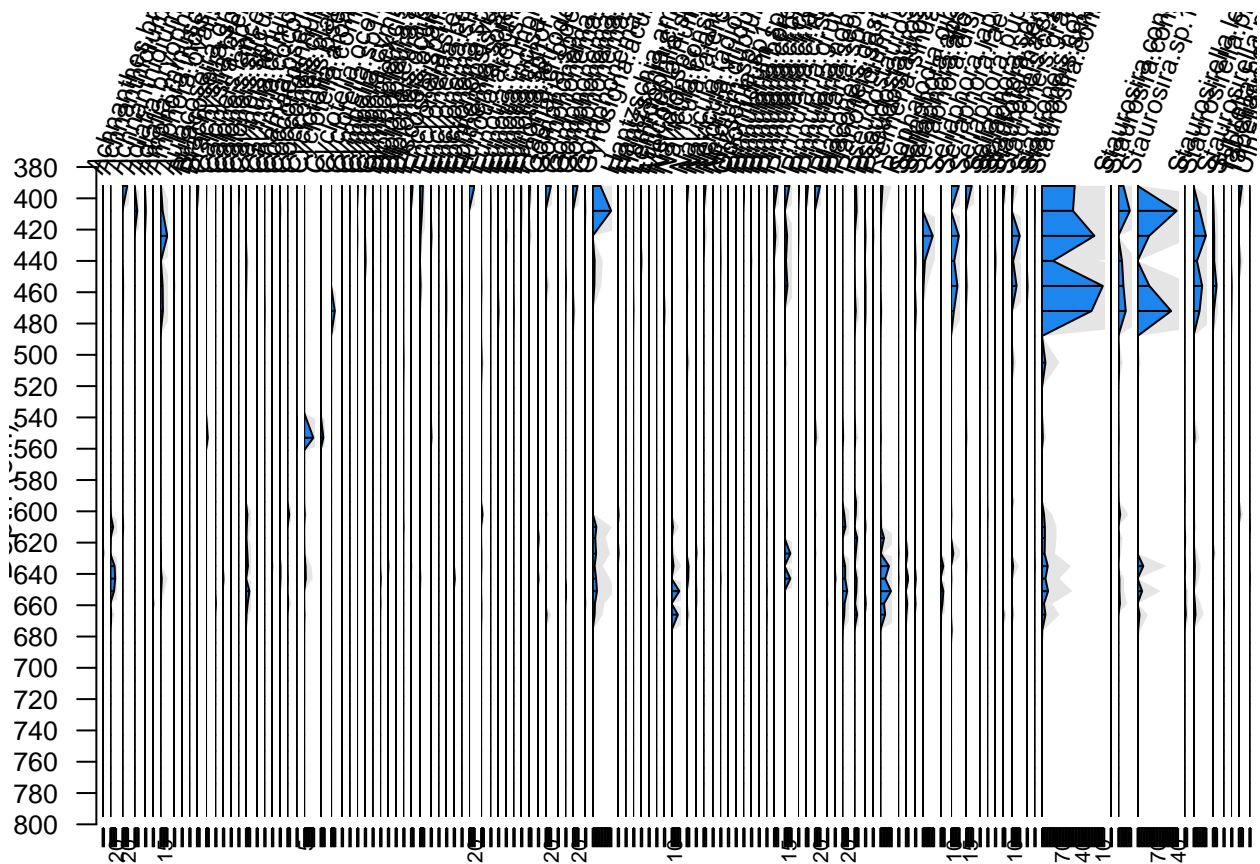


```
#diatoms richness
summary(tram20_diatoms$richness)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      0.00   0.00    6.00   11.89  25.00   46.00
```

Figure 7 - Selection of diatoms

```
inc_diatoms <- rep(5, ncol(tram20_diatoms %>%
  select(Achnanthes.brevipes:Ulnaria.sp) %>%
  select(where(~ sum(., na.rm = TRUE) > 5))))
ex.diat <- rep(TRUE, times = 55)
diatoms_selection <- strat.plot(tram20_diatoms %>%
  select(Achnanthes.brevipes:Ulnaria.sp) %>%
  select(where(~ sum(., na.rm = TRUE) > 5)),
  yvar=tram20_diatoms %>%
  select(depth),
  ylabel="Depth (cm)", y.tks=seq(380, 800, 20), y.rev=TRUE, scale.percent
  plot.line=FALSE, plot.poly=TRUE, plot.bar=TRUE, col.bar="black", col.po
  srt.xlabel=70, cex.xlabel = 1, cex.axis=0.8,
  exag = ex.diat, x.pc.inc=5,
  x.pc.omit0=TRUE, las=2, xSpace=0.007)
text(x = par("usr")[1] + 1.2 * diff(par("usr")[1:2]),
  y = labels_points, labels = zone_labels_pollen, pos = 2, cex = 0.6, xpd = TRUE)
```

11. Discussion figure vertical

11.1. Import sedaDNA data and create data frames for plotting

11.1.1. Plant sedaDNA

```
head(plant_dna)
```

```
plant_rai <- plant_dna %>% #obtain dataframe
  mutate(across(.cols = starts_with("weightrep_sample.TRAM_1b_"),
    .fns = ~ if_any(starts_with("proprep_sample.TRAM_1b_"), ~ . <= 0.125) * 0 +
      if_all(starts_with("proprep_sample.TRAM_1b_"), ~ . > 0.125) * .)) %>% #discard sample
  mutate(across(starts_with("totread_sample."),
    ~. / sum(., na.rm = TRUE), .names = "norm_{col}")) %>% #normalize totread
  mutate(across(starts_with("norm_totread_sample."),
    ~. * get(gsub("norm_totread_sample.", "weightrep_sample.", cur_column()),
      .names = "RAI_{sub('norm_totread_sample.', '', col)}")) %>% #totread*weightrep
  ungroup() %>%
  rename_with(~ gsub("RAI_.norm_totread_sample.", "RAI_", .), starts_with("RAI_.norm_totread")) %>% #re
  rename_with(~ gsub("_rpt$", "", .), starts_with("RAI_")) %>% #rename
  select(-c(starts_with("totread_sample"), starts_with("weightrep_sample"), starts_with("norm_totread_s
  select(-starts_with("RAI_TRAM_C")) #remove controls
```

```

plant_rai_perc <- plant_rai %>%
  filter(!group %in% c("Bryophyte", "Hydrophyte", "Hygrophyte",
    "Not native", "Algae", "Positive control",
    "Other plant", "Other tree")) %>% #remove some groups from percentages so that w
  mutate(across(starts_with("RAI_TRAM"), ~./sum(., na.rm = TRUE)*100))

plant_rai_long <- plant_rai %>%
  rename_with(~ gsub("RAI_", "", .), starts_with("RAI_")) %>%
  pivot_longer(cols = c("TRAM_1b_3U_11.12":"TRAM_1b_4U_109.110"),
    values_to = 'RAI',
    names_to = 'SampleId') %>%
  left_join(read.csv ("E:/Saco/IJP/1_data/Tramacastilla/depth/TRAM21/TRAM21_composite_depth_ages.csv") %>%
    rename("SampleId" = "SampleID",
      "age" = "median") %>%
    select("SampleId", "age"),
    by = "SampleId")

plant_rai_perc_long <- plant_rai_perc %>%
  rename_with(~ gsub("RAI_", "", .), starts_with("RAI_")) %>%
  pivot_longer(cols = c("TRAM_1b_3U_11.12":"TRAM_1b_4U_109.110"),
    values_to = 'RAI',
    names_to = 'SampleId') %>%
  left_join(read.csv ("E:/Saco/IJP/1_data/Tramacastilla/depth/TRAM21/TRAM21_composite_depth_ages.csv") %>%
    rename("SampleId" = "SampleID",
      "age" = "median") %>%
    select("SampleId", "age"),
    by = "SampleId")

```

Trees

```

trees_dna <- plant_rai %>%
  filter(scientific_name %in% c("Abies alba", "Betula", "Betulaceae", "Fagus",
    "Fagus sylvatica", "Pinus", "Quercus 1")) %>%
  rename_with(~ gsub("RAI_", "", .), starts_with("RAI_")) %>%
  select(scientific_name, TRAM_1b_3U_11.12:TRAM_1b_4U_109.110) %>%
  mutate(scientific_name = case_when(scientific_name %in% c("Fagus", "Fagus sylvatica") ~ "Fagus sylvat
    TRUE ~ scientific_name )) %>%
  mutate(group = paste0(scientific_name, "_dna")) %>%
  pivot_longer(cols = c("TRAM_1b_3U_11.12":"TRAM_1b_4U_109.110"),
    values_to = 'RAI',
    names_to = 'SampleId') %>%
  left_join(read.csv ("E:/Saco/IJP/1_data/Tramacastilla/depth/TRAM21/TRAM21_composite_depth_ages.csv") %>%
    rename("SampleId" = "SampleID",
      "age" = "median") %>%
    select("SampleId", "age"),
    by = "SampleId") %>%
  select(-SampleId) %>%
  group_by(group, age) %>%
  summarise(Influx = sum(RAI, na.rm = TRUE), .groups = "drop")

```

Herbaceous taxa

```

herbs_dna <- plant_rai %>%
  filter(group2 %in% c("Novel herbaceous taxa", "Herbaceous taxa", "Memoral")) %>%
  mutate(group = "Herbs_dna") %>%
  rename_with(~ gsub("RAI_", "", .), starts_with("RAI_")) %>%
  select(group, TRAM_1b_3U_11.12:TRAM_1b_4U_109.110) %>%
  pivot_longer(cols = c("TRAM_1b_3U_11.12":"TRAM_1b_4U_109.110"),
    values_to = 'RAI',
    names_to = 'SampleId') %>%
  left_join(read.csv ("E:/Saco/IJP/1_data/Tramacastilla/depth/TRAM21/TRAM21_composite_depth_ages.csv") %>%
    rename("SampleId" = "SampleID",
      "age" = "median") %>%
    select("SampleId", "age"),
    by = "SampleId") %>%
  group_by(group, age) %>%
  summarise(Influx = sum(RAI, na.rm = TRUE), .groups = "drop")

```

Steppe taxa

```

steppe_dna <- plant_rai %>%
  filter(family_name == "Asteraceae" |
    family_name == "Chenopodiaceae" |
    family_name == "Brassicaceae" |
    genus_name == "Helianthemum" |
    genus_name == "Artemisia" |
    genus_name == "Plantago" |
    genus_name == "Rumex" |
    genus_name == "Ephedra") %>%
  rename_with(~ gsub("RAI_", "", .), starts_with("RAI_")) %>%
  mutate(group = "Steppe_dna") %>%
  select(group, TRAM_1b_3U_11.12:TRAM_1b_4U_109.110) %>%
  pivot_longer(cols = c("TRAM_1b_3U_11.12":"TRAM_1b_4U_109.110"),
    values_to = 'RAI',
    names_to = 'SampleId') %>%
  left_join(read.csv ("E:/Saco/IJP/1_data/Tramacastilla/depth/TRAM21/TRAM21_composite_depth_ages.csv") %>%
    rename("SampleId" = "SampleID",
      "age" = "median") %>%
    select("SampleId", "age"),
    by = "SampleId") %>%
  group_by(group, age) %>%
  summarise(Influx = sum(RAI, na.rm = TRUE), .groups = "drop")

```

Poaceae

```

poaceae_dna <- plant_rai %>%
  filter(family_name == "Poaceae") %>%
  rename_with(~ gsub("RAI_", "", .), starts_with("RAI_")) %>%
  mutate(group = "Poaceae_dna") %>%
  select(group, TRAM_1b_3U_11.12:TRAM_1b_4U_109.110) %>%
  pivot_longer(cols = c("TRAM_1b_3U_11.12":"TRAM_1b_4U_109.110"),
    values_to = 'RAI',
    names_to = 'SampleId') %>%
  left_join(read.csv ("E:/Saco/IJP/1_data/Tramacastilla/depth/TRAM21/TRAM21_composite_depth_ages.csv") %>%
    rename("SampleId" = "SampleID",

```

```

      "age" = "median") %>%
    select("SampleId", "age"),
    by = "SampleId") %>%
group_by(group, age) %>%
summarise(Influx = sum(RAI, na.rm = TRUE), .groups = "drop")

```

Anthropogenic taxa

```

anthropogenic_dna <- plant_rai %>%
  filter(family_name == "Brassicaceae" |
    species_name == "Urtica dioica" |
    genus_name == "Hordeum" |
    genus_name == "Plantago" |
    genus_name == "Rumex") %>%
  rename_with(~ gsub("RAI_", "", .), starts_with("RAI_")) %>%
  mutate(group = "Anthropogenic_dna") %>%
  select(group, TRAM_1b_3U_11.12:TRAM_1b_4U_109.110) %>%
  pivot_longer(cols = c("TRAM_1b_3U_11.12":"TRAM_1b_4U_109.110"),
    values_to = 'RAI',
    names_to = 'SampleId') %>%
  left_join(read.csv ("E:/Saco/IJP/1_data/Tramacastilla/depth/TRAM21/TRAM21_composite_depth_ages.csv") %>%
    rename("SampleId" = "SampleID",
      "age" = "median") %>%
    select("SampleId", "age"),
    by = "SampleId") %>%
  group_by(group, age) %>%
  summarise(Influx = sum(RAI, na.rm = TRUE), .groups = "drop")

```

Now join all herbs DNA into one dataframe CAREFUL: the herbaceous component includes poaceae, anthropogenic and steppe, so i will have to exclude them from the herbs_dna dataframe for the figure

```

herbs_poaceae_steppe_anthropogenic_dna <- herbs_dna %>%
  full_join(poaceae_dna, by = c("group", "age", "Influx")) %>%
  full_join(steppe_dna, by = c("group", "age", "Influx")) %>%
  full_join(anthropogenic_dna, by = c("group", "age", "Influx")) %>%
  pivot_wider(names_from = group, values_from = Influx) %>%
  mutate(Herbs_clean_dna = Herbs_dna - Anthropogenic_dna - Steppe_dna - Poaceae_dna) %>%
  pivot_longer(cols = c(Poaceae_dna, Steppe_dna, Anthropogenic_dna, Herbs_clean_dna),
    names_to = "group",
    values_to = "Influx")

```

11.1.2. Animal sedaDNA

```
head(RAI.animals)
```

11.2. Join all data

```

figure_all <- tram20_PAR_groups_discussion %>%
  select(-depth) %>%
  full_join(RAI.animals %>%
    rename("Cattle" = "Bos taurus",
           "Sheep" = "Ovis aries",
           "Goat" = "Capra hircus") %>%
    select (age, Cattle, Sheep, Goat), by = "age") %>%
  full_join(CHAR %>%
    select(age, CHAR.tram), by = "age") %>%
  left_join(Mendukilo_binned, by = "age") %>%
  pivot_longer(cols = c("Abies": "d13C.Mendukilo"),
               values_to = 'Influx',
               names_to = 'group') %>%
  filter(age < 15000) %>%
  as.data.frame()

XRF_selection <- tram20_XRF_LECO_MS %>%
  right_join(tram20_depths_ages_XRF %>%
    select(SampleId, age), by = "SampleId") %>%
  relocate(age, .after = "composite_depth") %>%
  select(SampleId, age, Si, TOC)

```

11.3. Plot

```

p.char <- figure_all %>%
  dplyr::filter(group %in% "CHAR.tram") %>%
  ggplot(aes(x=age/1000, y=round(Influx, 2), fill = group)) +
  ylab("") +
  theme_classic() +
  geom_col(width = 0.10) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
                                           breaks = seq(0.5, 15, by = 0.5),
                                           labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)), "")) +
  guides(fill = guide_legend (title = "")) +
  scale_fill_manual(values = "black",
                   labels = "CHAR") +
  scale_y_continuous() +
  theme(axis.title.x = element_blank(),
        legend.position = "right",
        plot.margin = margin(0, 0, 0, 0, "cm"),
        legend.text = element_text(size = 11))

p.w.l.ratio <- W.L.ratio %>%
  rename("Type of fuel" = "fuel") %>%
  ggplot(aes(x = age/1000, y = count, fill = `Type of fuel`, color = `Type of fuel`)) +
  geom_area(alpha = 0.6, position = "stack") +
  geom_smooth(method = "loess", se = FALSE, aes(color = `Type of fuel`), linewidth = 1, alpha = 0.8) +
  scale_y_continuous(trans = "sqrt", # transformation with squared root
                    breaks = c(0, 5, 20, 50)) +
  labs(y = "") +
  theme_classic() +
  scale_x_reverse(limits = c(14.909, min(W.L.ratio$age)/1000),

```

```

        breaks = seq(0.5, 15, by = 0.5),
        labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)),
scale_fill_manual(values = c("Woody" = "darkgreen", "Grasslands" = "#FFD700")) +
scale_color_manual(values = c("Woody" = "darkgreen", "Grasslands" = "#FFD700")) +
theme(axis.title.x = element_blank(),
      legend.position = "right",
      plot.margin = margin(0, 0, 0, 0, "cm"),
      legend.text = element_text(size = 11)) +
annotate("text", x = 2, y = 120, label = "peak = 59", vjust = -1, color = "red")

p.trees <- figure_all %>%
  dplyr::filter(group %in% c("Abies", "Fagus", "Pinus")) %>%
  mutate(group = fct_relevel(group, "Pinus", "Abies", "Fagus")) %>%
  ggplot(aes(x=age/1000, y=round(Influx, 2) , group=group, fill=group))+
  guides(fill = guide_legend(title = "Plant group")) +
  ylab("")+
  theme_classic() +
  geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
                                          breaks = seq(0.5, 15, by = 0.5),
                                          labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)),
scale_fill_manual(values = c("#422CB2", "#BFB2FF", "#E18727")) +
scale_y_continuous(limits = c(0, 1.25e07),
                    breaks = c(0, 4e+6, 8e+6, 1.2e+7),
                    labels = c("0", "4e+6", "8e+6", "1.2e+7")) +
  theme(axis.title.x = element_blank(),
        legend.position = "right", legend.text = element_text(face = "italic", size = 11),
        plot.margin = margin(0, 0, 0, 0, "cm"),
        legend.title = element_text(size = 14))

p.trees_dna <- trees_dna %>%
  dplyr::filter(group %in% c("Abies alba_dna", "Fagus sylvatica_dna", "Pinus_dna")) %>%
  mutate(group = fct_relevel(group, "Pinus_dna", "Abies alba_dna", "Fagus sylvatica_dna")) %>%
  ggplot(aes(x=age/1000, y=round(Influx, 2) , group=group, fill=group))+
  ylab("")+
  theme_classic() +
  geom_col_pattern(width = 0.15, pattern = "stripe", pattern_density = 0.15, pattern_spacing = 0.05,
                  pattern_angle = 45, pattern_fill = "black", pattern_colour = NA, aes(pattern_type = "dna"),
                  show.legend = c(fill = FALSE, pattern = TRUE)) +
  scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
                  breaks = seq(0.5, 15, by = 0.5),
                  labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)),
scale_fill_manual(values = c("#422CB2", "#BFB2FF", "#E18727")) +
scale_pattern_type_manual(values = "stripe", name = NULL, labels = "sedaDNA") +
scale_y_continuous(limits = c(0, 0.6),
                    breaks = c(0, 0.2, 0.4, 0.6),
                    labels = c("0", "0.2", "0.4", "0.6")) +
  theme(axis.title.x = element_blank(),
        plot.margin = margin(0, 0, 0, 0, "cm"),
        legend.position = "none")

p.dec.trees <- figure_all %>%
  dplyr::filter(group %in% c("Betula", "Corylus", "Deciduous Quercus")) %>%
  ggplot(aes(x=age/1000, y=round(Influx, 2) , group=group, fill=group))+

```



```

guides (fill = guide_legend(title = "")) +
ylab("")+
theme_classic() +
geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
                                         breaks = seq(0.5, 15, by = 0.5),
                                         labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)),
scale_fill_manual(values = c("#749B58", "#99CC00", "#AE1F63")) +
scale_y_continuous(limits = c(0, 1.25e07),
                   breaks = c(0, 4e+6, 8e+6, 1.2e+7),
                   labels = c("0", "4e+6", "8e+6", "1.2e+7")) +
theme(axis.title.x = element_blank(),
      legend.position = "right", legend.text = element_text(face = "italic", size = 11),
      plot.margin = margin(0, 0, 0, 0, "cm"))

p.dec.trees_dna <- trees_dna %>%
dplyr::filter(group %in% c("Betula_dna", "Betulaceae_dna", "Quercus 1_dna")) %>%
mutate(group = fct_relevel(group, "Betula_dna", "Betulaceae_dna", "Quercus 1_dna")) %>%
ggplot(aes(x = age/1000, y = Influx, group = group, fill = group)) +
ylab("") +
theme_classic() +
geom_col_pattern(width = 0.15, pattern = "stripe", pattern_density = 0.15, pattern_spacing = 0.05,
                pattern_angle = 45, pattern_fill = "black", pattern_colour = NA, aes(pattern_type = "stripe"),
                show.legend = c(fill = FALSE, pattern = TRUE)) +
scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
               breaks = seq(0.5, 15, by = 0.5),
               labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)),
scale_fill_manual(values = c("#749B58", "#99CC00", "#AE1F63"), guide = "none") +
scale_pattern_type_manual(values = "stripe", name = NULL, labels = "sedaDNA") +
scale_y_continuous(limits = c(0, 0.6),
                   breaks = c(0, 0.2, 0.4, 0.6),
                   labels = c("0", "0.2", "0.4", "0.6")) +
theme(axis.title.x = element_blank(),
      legend.position = "right", legend.text = element_text(face = "italic", size = 11),
      plot.margin = margin(0, 0, 0, 0, "cm")) +
guides(pattern_type = guide_legend(title = NULL, override.aes = list(fill = "gray90", pattern = "stripe")))

p.herbs_anthropogenic <- figure_all %>%
dplyr::filter(group %in% c("Poaceae", "Steppe taxa", "Other herbs", "Anthropogenic taxa")) %>%
mutate(group = fct_relevel(group, "Poaceae", "Steppe taxa", "Other herbs", "Anthropogenic taxa")) %>%
ggplot(aes(x=age/1000, y=round(Influx, 2) , group=group, fill=group))+
guides (fill = guide_legend(title = "")) +
ylab("")+
theme_classic() +
geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
                                         breaks = seq(0.5, 15, by = 0.5),
                                         labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)),
scale_fill_manual(values = c("#FFB547", "#FFFF00", "#9ECAE1", "#FF86FF")) +
scale_y_continuous(limits = c(0, 1.25e07),
                   breaks = c(0, 4e+6, 8e+6, 1.2e+7),
                   labels = c("0", "4e+6", "8e+6", "1.2e+7")) +
theme(axis.title.x = element_blank(),
      legend.position = "right",
      plot.margin = margin(0, 0, 0, 0, "cm"),

```



```

    legend.text = element_text(size = 11))

p.herbs_dna <- herbs_poaceae_steppe_anthropogenic_dna %>%
  mutate(group = fct_relevel(group, "Poaceae_dna", "Steppe_dna", "Herbs_clean_dna", "Anthropogenic_dna"))
  ggplot(aes(x=age/1000, y=round(Influx, 2) , group=group, fill=group))+
  guides (fill = guide_legend(title = "")) +
  ylab("")+
  theme_classic() +
  geom_col_pattern(width = 0.15, pattern = "stripe", pattern_density = 0.15, pattern_spacing = 0.05,
    pattern_angle = 45, pattern_fill = "black", pattern_colour = NA, aes(pattern_type = 
      show.legend = c(fill = FALSE, pattern = TRUE)) +
  scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
    breaks = seq(0.5, 15, by = 0.5),
    labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(
  scale_fill_manual(values = c("#FFB547", "#FFFF00", "#9ECAE1", "#FF86FF")) +
  scale_pattern_type_manual(values = "stripe", name = NULL, labels = "sedaDNA") +
  scale_y_continuous(limits = c(0, 0.6),
    breaks = c(0, 0.2, 0.4, 0.6),
    labels = c("0", "0.2", "0.4", "0.6")) +
  theme(axis.title.x = element_blank(),
    legend.position = "none",
    plot.margin = margin(0, 0, 0, 0, "cm"))

p.xrf.selection <- XRF_selection %>%
  filter(!is.na(TOC)) %>%
  ggplot(aes(x = age / 1000)) +
  geom_line(aes(y = Si/50), color = "#FF7F00", linewidth = 0.5) + # Divided by 50
  geom_line(aes(y = TOC * 100), color = "black", linewidth = 0.5) + # Multiplied by 100
  labs(x = "") +
  scale_y_continuous(breaks = c(0, 500, 1000, 1500, 2000),
    labels = c("0", "2.5e+4", "5e+4", "7.5e+4", "1e+5"),
    sec.axis = sec_axis(~./100)) +
  scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
    breaks = seq(0.5, 15, by = 0.5),
    labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)
  theme_classic() +
  theme(axis.title.x = element_blank(),
    axis.title.y.left = element_blank(),
    axis.title.y.right = element_blank(),
    legend.position = "none",
    plot.margin = margin(0, 0, 0, 0, "cm")) +
  scale_color_manual(values = c("#FF7F00", "black"))

p.men <- Mendukilo %>%
  ggplot(aes(x=age/1000, y=d13C.Mendukilo))+
  xlab("Age (ka BP)") +
  ylab("")+

  theme_classic() +
  geom_line() + scale_y_reverse() +
  scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
    breaks = seq(0.5, 15, by = 0.5),
    labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)

```

```

  theme(plot.margin = margin(0, 0, 0, 0, "cm"),
        legend.text = element_text(size = 11))

p.char/p.w.l.ratio/p.trees/p.dec.trees/p.herbs_anthropogenic/p.trees_dna/p.dec.trees_dna/p.herbs_dna/p.
plot_layout(heights = c(1, 1, 2, 2, 2, 1, 1, 1, 1, 2), guides = "collect") +
  theme(plot.margin = margin(0, 0, 0, 0, "pt")) #save pdf A4 vertical

## Warning: 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 67 rows containing missing values or values outside the scale range
## ('geom_col()').

## 'geom_smooth()' using formula = 'y ~ x'

## Warning: 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 507 rows containing missing values or values outside the scale range
## ('geom_col()').

## Warning: 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 508 rows containing missing values or values outside the scale range
## ('geom_col()').

## Warning: 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 676 rows containing missing values or values outside the scale range
## ('geom_col()').

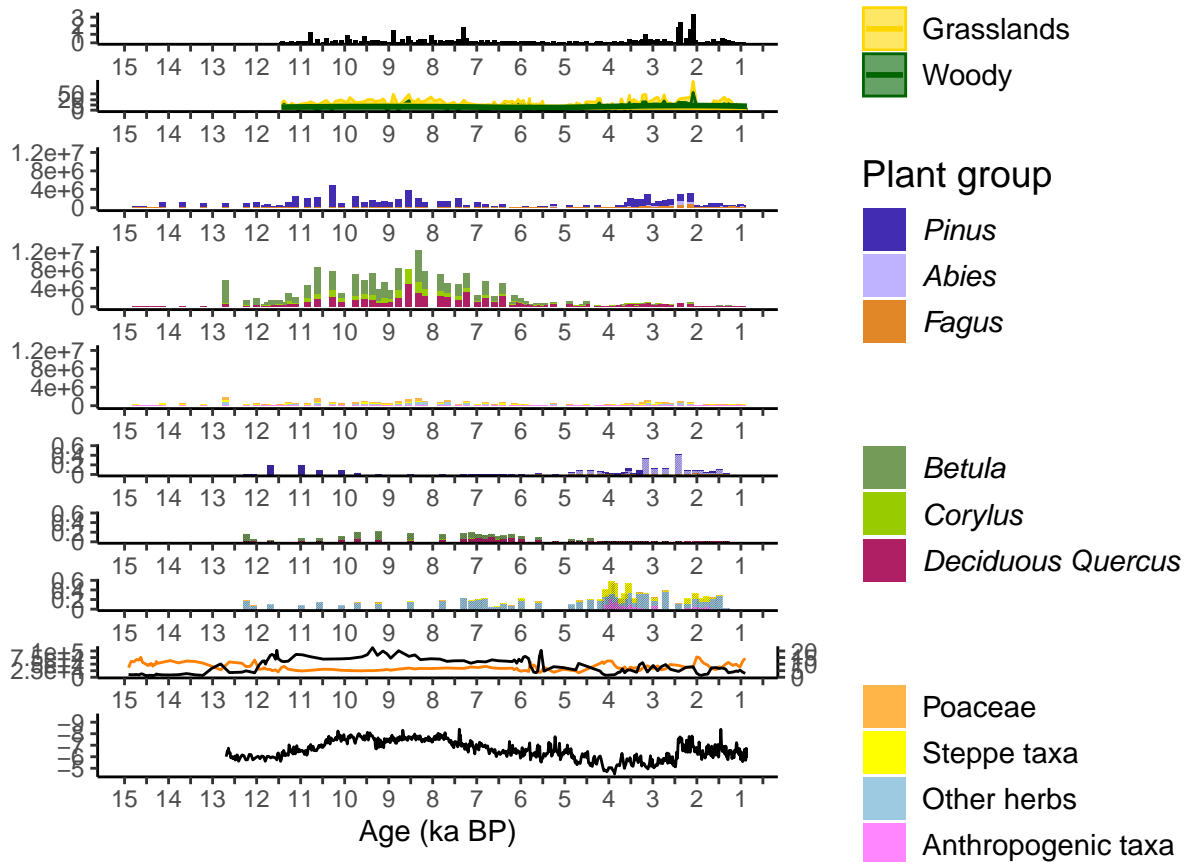
## Warning: 'position_stack()' requires non-overlapping x intervals.
## 'position_stack()' requires non-overlapping x intervals.
## 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 2 rows containing missing values or values outside the scale range
## ('geom_col_pattern()').

## Warning: Removed 36 rows containing missing values or values outside the scale range
## ('geom_line()').
## Removed 36 rows containing missing values or values outside the scale range
## ('geom_line()').

## Warning: Removed 208 rows containing missing values or values outside the scale range
## ('geom_line()').

```



12. Discussion figure methodological approach

```
figure_percentages <- tram20_pollen_groups_long %>%
  group_by(SampleId, pollen_group_discussion) %>%
  summarise(total_pollen = sum(pollen_counts, na.rm = TRUE)) %>%
  left_join(tram20_pollen_groups_long %>%
    select(SampleId, depth, age, Lycopodium, `Terrestrial pollen sum`) %>%
    distinct(), by = "SampleId") %>%
  mutate(percentage = (total_pollen / `Terrestrial pollen sum`) * 100) %>%
  select(SampleId, depth, age, Lycopodium, pollen_group_discussion, total_pollen, `Terrestrial pollen sum`) %>%
  rename("group" = "pollen_group_discussion") %>%
  filter(age < 15000) %>%
  as.data.frame()
```

'summarise()' has grouped output by 'SampleId'. You can override using the
'.groups' argument.

```
p8.trees_perc <- figure_percentages %>%
  dplyr::filter(group %in% c("Abies", "Fagus", "Pinus")) %>%
  mutate(group = fct_relevel(group, "Pinus", "Abies", "Fagus")) %>%
  ggplot(aes(x=age/1000, y=round(percentage, 2), group=group, fill=group)) +
  guides(fill = guide_legend(title = "Plant group")) +
```



```

guides (fill = guide_legend(title = "")) +
ylab("")+
theme_classic() +
geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
breaks = seq(0.5, 15, by = 0.5),
labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)), "")) +
scale_fill_manual(values = c("#749B58", "#99CC00", "#AE1F63")) +
scale_y_continuous(limits = c(0, 75), breaks = c(0, 25, 50, 75)) +
theme(axis.title.x = element_blank(),
legend.position = "top", legend.text = element_text(face = "italic", size = 11),
plot.margin = margin(0, 0, 0, 0, "cm"))

p8.dec.trees_PAR <- figure_all %>%
dplyr::filter(group %in% c("Betula", "Corylus", "Deciduous Quercus")) %>%
ggplot(aes(x=age/1000, y=round(Influx, 2) , group=group, fill=group))+
guides (fill = guide_legend(title = "")) +
ylab("")+
theme_classic() +
geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
breaks = seq(0.5, 15, by = 0.5),
labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)), "")) +
scale_fill_manual(values = c("#749B58", "#99CC00", "#AE1F63")) +
scale_y_continuous(limits = c(0, 1.25e07),
breaks = c(0, 4e+6, 8e+6, 1.2e+7),
labels = c("0", "4e+6", "8e+6", "1.2e+7")) +
theme(axis.title.x = element_blank(),
legend.position = "none",
plot.margin = margin(0, 0, 0, 0, "cm"))

p8.dec.trees_dna <- trees_dna %>%
dplyr::filter(group %in% c("Betula_dna", "Betulaceae_dna", "Quercus 1_dna")) %>%
mutate(group = fct_relevel(group, "Betula_dna", "Betulaceae_dna", "Quercus 1_dna")) %>%
ggplot(aes(x = age/1000, y = Influx, group = group, fill = group)) +
ylab("") + xlab("Age (ka BP)") +
theme_classic() +
geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
breaks = seq(0.5, 15, by = 0.5),
labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)), "")) +
scale_fill_manual(values = c("#749B58", "#99CC00", "#AE1F63"), guide = "none") +
scale_y_continuous(limits = c(0, 0.6),
breaks = c(0, 0.2, 0.4, 0.6),
labels = c("0", "0.2", "0.4", "0.6")) +
theme(legend.position = "none",
plot.margin = margin(0, 0, 0, 0, "cm"))

p8.herbs_perc <- figure_percentages %>%
dplyr::filter(group %in% c("Poaceae", "Steppe taxa", "Other herbs", "Anthropogenic taxa")) %>%
mutate(group = fct_relevel(group, "Poaceae", "Steppe taxa", "Other herbs", "Anthropogenic taxa")) %>%
ggplot(aes(x=age/1000, y=round(percentage, 2) , group=group, fill=group))+
guides (fill = guide_legend(title = "")) +
ylab("")+
theme_classic() +
geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
breaks = seq(0.5, 15, by = 0.5),
labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(seq(0.5, 15, by = 0.5)), "")) +
scale_fill_manual(values = c("#749B58", "#99CC00", "#AE1F63")) +
scale_y_continuous(limits = c(0, 100), breaks = c(0, 20, 40, 60, 80, 100)) +
theme(axis.title.x = element_blank(),
legend.position = "top", legend.text = element_text(face = "italic", size = 11),
plot.margin = margin(0, 0, 0, 0, "cm"))

```

```

                                breaks = seq(0.5, 15, by = 0.5),
                                labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(
scale_fill_manual(values = c("#FFB547", "#FFFF00", "#9ECAE1", "#FF86FF")) +
scale_y_continuous(limits = c(0, 75), breaks = c(0, 25, 50, 75)) +
theme(axis.title.x = element_blank(),
      legend.position = "top",
      plot.margin = margin(0, 0, 0, 0, "cm"),
      legend.text = element_text(size = 11))

p8.herbs_PAR <- figure_all %>%
  dplyr::filter(group %in% c("Poaceae", "Steppe taxa", "Other herbs", "Anthropogenic taxa")) %>%
  mutate(group = fct_relevel(group, "Poaceae", "Steppe taxa", "Other herbs", "Anthropogenic taxa")) %>%
  ggplot(aes(x=age/1000, y=round(Influx, 2) , group=group, fill=group))+
  guides (fill = guide_legend(title = "")) +
  ylab("")+
  theme_classic() +
  geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
                                breaks = seq(0.5, 15, by = 0.5),
                                labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(
scale_fill_manual(values = c("#FFB547", "#FFFF00", "#9ECAE1", "#FF86FF")) +
scale_y_continuous(limits = c(0, 1.25e07),
      breaks = c(0, 4e+6, 8e+6, 1.2e+7),
      labels = c("0", "4e+6", "8e+6", "1.2e+7")) +
  theme(axis.title.x = element_blank(),
      legend.position = "none",
      plot.margin = margin(0, 0, 0, 0, "cm"))

p8.herbs_dna <- herbs_poaceae_steppe_anthropogenic_dna %>%
  mutate(group = fct_relevel(group, "Poaceae_dna", "Steppe_dna", "Herbs_clean_dna", "Anthropogenic_dna")) %>%
  ggplot(aes(x=age/1000, y=round(Influx, 2) , group=group, fill=group))+
  guides (fill = guide_legend(title = "")) +
  ylab("") + xlab("Age (ka BP)") +
  theme_classic() +
  geom_col(width = 0.15) + scale_x_reverse(limits = c(max(figure_all$age)/1000, min(figure_all$age)/1000),
                                breaks = seq(0.5, 15, by = 0.5),
                                labels = ifelse(seq(0.5, 15, by = 0.5) %% 1 == 0, as.character(
scale_fill_manual(values = c("#FFB547", "#FFFF00", "#9ECAE1", "#FF86FF")) +
scale_y_continuous(limits = c(0, 0.6),
      breaks = c(0, 0.2, 0.4, 0.6),
      labels = c("0", "0.2", "0.4", "0.6")) +
  theme(legend.position = "none",
      plot.margin = margin(0, 0, 0, 0, "cm"))

p8.trees_perc/p8.dec.trees_perc/p8.herbs_perc/p8.trees_PAR/p8.dec.trees_PAR/p8.herbs_PAR/p8.trees_dna/p8.dec.trees_dna
  plot_layout(heights = c(3, 4, 4), ncol = 3, nrow = 3) +
  theme(plot.margin = margin(0, 0, 0, 0, "pt")) #save pdf A4 horizontal

```

```
## Warning: 'position_stack()' requires non-overlapping x intervals.
```

```
## Warning: Removed 6 rows containing missing values or values outside the scale range
## ('geom_col()').
```

```
## Warning: 'position_stack()' requires non-overlapping x intervals.
```

```
## Warning: Removed 6 rows containing missing values or values outside the scale range
## ('geom_col()').

## Warning: 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_col()').

## Warning: 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 507 rows containing missing values or values outside the scale range
## ('geom_col()').

## Warning: 'position_stack()' requires non-overlapping x intervals.

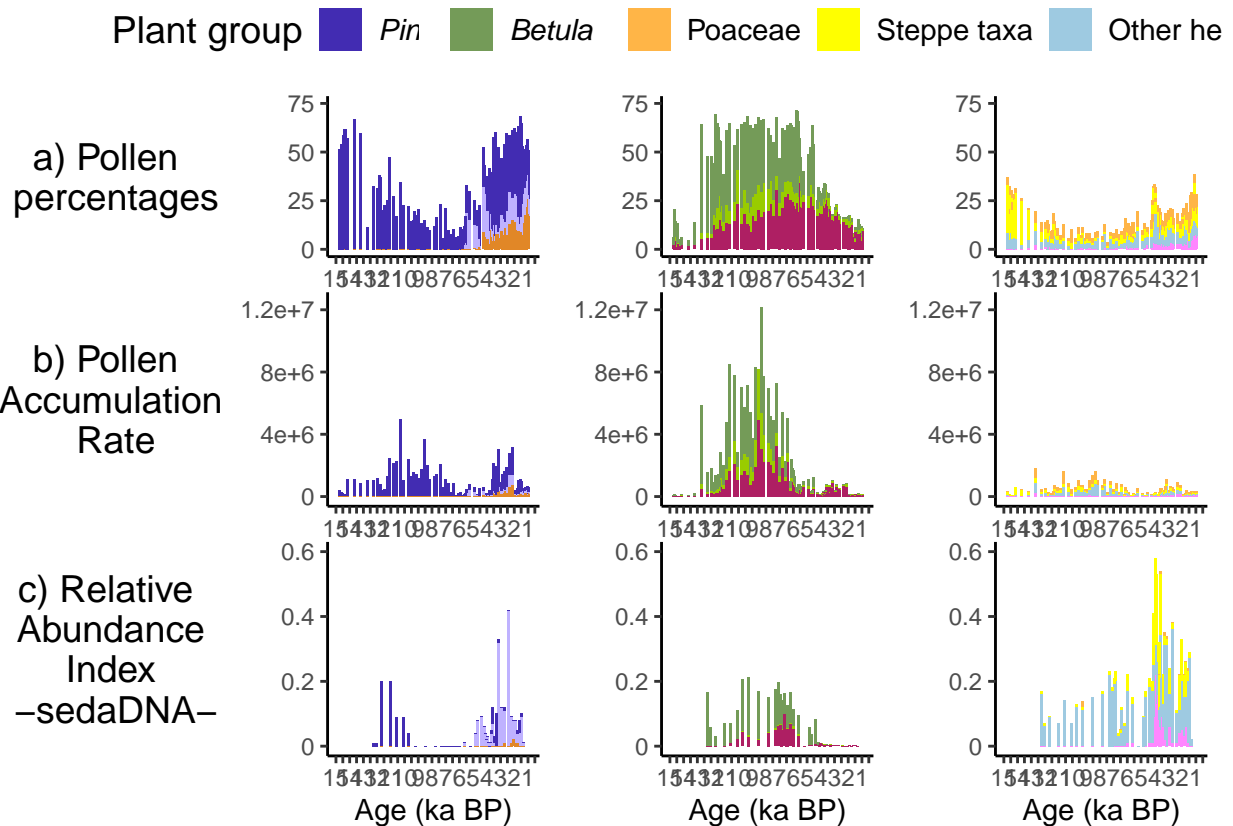
## Warning: Removed 508 rows containing missing values or values outside the scale range
## ('geom_col()').

## Warning: 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 676 rows containing missing values or values outside the scale range
## ('geom_col()').

## Warning: 'position_stack()' requires non-overlapping x intervals.
## 'position_stack()' requires non-overlapping x intervals.
## 'position_stack()' requires non-overlapping x intervals.

## Warning: Removed 2 rows containing missing values or values outside the scale range
## ('geom_col()').
```



13. Discussion figure Pyrenean records

13.1. Import data

13.1.1. Pllan dEstan

```
head(PDE_pollen)
```

13.1.2. Marbore

```
head(marbore_PAR)
head(marbore_pollen)
```

13.1.3. Basa de la Mora

```
head(bsm_PAR)
head(bsm_pollen)
```


13.1.4. Bassa Nera

```
head(bassanera)
```

```
bassanera_groups <- bassanera %>%
  filter(SampleId == "group") %>%
  select(-SampleId, -age) %>%
  pivot_longer(everything(), names_to = "taxon", values_to = "group")

bassanera_pollen <- bassanera %>%
  filter(SampleId != "group") %>%
  select(-SampleId) %>%
  mutate(across(everything(), as.numeric)) %>%
  rowwise() %>%
  mutate_all(~replace(., is.na(.), 0)) %>%
  mutate(tree = sum(c_across(any_of(bassanera_groups$taxon[bassanera_groups$group == "TR"]))),
         herb = sum(c_across(any_of(bassanera_groups$taxon[bassanera_groups$group == "UPHE"])))) %>%
  ungroup() %>%
  mutate(pollen_sum = rowSums(select(., Abies:Ulmus), na.rm = TRUE)) %>%
  relocate(pollen_sum, .after = "age") %>%
  pivot_longer(cols = c("Abies":"herb"),
               values_to = "pollen_counts",
               names_to = "taxon") %>%
  mutate_at("pollen_counts", ~replace(., is.na(.), 0)) %>%
  group_by(age, taxon) %>%
  mutate(percentage = (pollen_counts/pollen_sum)*100)
```

13.1.5. Portalet

```
head(portalet_pollen)
```

13.1.6. Estanya

```
head(estanya)
```

```
estanya_groups <- estanya %>%
  pivot_longer(Abies:Ruppia, names_to = "taxon", values_to = "pollen_count") %>%
  mutate(group = case_when(taxon %in% c("Betula", "Corylus", "Alnus", "Carpinus", "Salix", "Ulmus",
                                         "Populus", "Acer", "Fagus", "Fraxinus", "Tilia", "Juglans", "Carya",
                                         "Quercus caducifolio") ~ "deciduous_trees",
                           taxon %in% c("Poaceae", "Artemisia", "Chenopodiaceae", "Caryophyllaceae", "Urticaceae",
                                         "Rumex", "Cichorioideae", "Asteroideae", "Carduae", "Plantago") ~ "other_herbs",
                           taxon %in% c("Cichorioideae", "Asteroideae", "Carduae", "Centaurea") ~ "other_flores",
                           TRUE ~ "Other"))

estanya_pollen <- estanya %>%
  rowwise() %>%
  mutate(deciduous_trees = sum(c_across(any_of(estanya_groups$taxon[estanya_groups$group == "deciduous_trees"])))
```

```

    steppe = sum(c_across(any_of(estanya_groups$taxon[estanya_groups$group == "steppe"]))),
    other_asteraceae = sum(c_across(any_of(estanya_groups$taxon[estanya_groups$group == "other_ast
ungroup() %>%
relocate(pollen_sum, .after = "age") %>%
pivot_longer(cols = c("Abies": "other_asteraceae"),
              values_to = "percentage",
              names_to = "taxon")

```

13.2. Plot Holocene figure

```

periods_lines_holocene <- c(4.2, 8.2, 11.7) #to add Holocene periods
p11.tramacastilla.v <- tram20_pollen %>%
  select(-c(`No of taxa`, Lycopodium, `Pteropsida trilete undiff`: `Charcoal <150`)) %>%
  pivot_longer(cols = c("Abies": "Hygrophytes"),
               values_to = 'pollen_counts',
               names_to = 'taxon') %>%
  mutate_at("pollen_counts", ~replace(., is.na(.), 0)) %>%
  group_by(SampleId, taxon) %>%
  mutate(percentage = (pollen_counts / `Terrestrial pollen sum`) * 100) %>%
  filter(taxon %in% c("Betula", "Corylus", "Deciduous Quercus", "Abies", "Pinus", "Herbs")) %>%
  mutate(taxon = factor(taxon)) %>%
  mutate(taxon = fct_relevel(taxon, "Abies", "Pinus", "Betula", "Corylus", "Deciduous Quercus", "Herbs"))
ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
  geom_area() +
  scale_x_reverse(limits = c(11.7, 0),
                 breaks = seq(0, 12, by = 0.5),
                 labels = ifelse(seq(0, 12, by = 0.5) %% 1 == 0, as.character(seq(0, 12, by = 0.5)), ""))
  scale_fill_manual(values = c("#BFB2FF", "#422CB2", "#749B58", "#99CC00", "#AE1F63", "#9ECAE1")) +
  theme_classic() +
  labs(x = "Age (ka BP)", title = "Tramacastilla \n (1682 m a.s.l.)") +
  theme(axis.title.y = element_blank(),
        axis.title.x = element_text(size = 14),
        legend.position = "none",
        plot.margin = margin(0, 0, 0, 0, "cm"),
        plot.title.position = "panel") +
  geom_vline(xintercept = periods_lines_holocene, color = "black", linetype = "dashed", linewidth = 0.3)

p11.portalet.v <- portalet_pollen %>%
  filter(taxon %in% c("Betula", "Corylus", "deciduous Quercus", "Abies", "Pinus", "herb")) %>%
  mutate(taxon = factor(taxon)) %>%
  mutate(taxon = fct_relevel(taxon, "Abies", "Pinus", "Betula", "Corylus", "deciduous Quercus", "herb"))
ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
  geom_area() +
  scale_x_reverse(limits = c(11.7, 0),
                 breaks = seq(0, 12, by = 0.5),
                 labels = ifelse(seq(0, 12, by = 0.5) %% 1 == 0, as.character(seq(0, 12, by = 0.5)), ""))
  scale_y_continuous(limits = c(0, 100)) +
  scale_fill_manual(values = c("#BFB2FF", "#422CB2", "#749B58", "#99CC00", "#AE1F63", "#9ECAE1")) +
  theme_classic() +
  labs(title = "El Portalet peatbog \n (1850 m a.s.l.)") +
  theme(axis.title.y = element_blank(),

```

```

axis.title.x = element_blank(),
legend.position = "none",
plot.margin = margin(0, 0, 0, 0, "cm"),
plot.title.position = "panel") +
geom_vline(xintercept = periods_lines_holocene, color = "black", linetype = "dashed", linewidth = 0.3)

p11.marbore.v <- marbore_pollen %>%
  filter(taxon %in% c("Betula", "Corylus", "Dec_Querc", "Abies", "Pinus", "herb")) %>%
  mutate(taxon = factor(taxon)) %>%
  mutate(taxon = fct_relevel(taxon, "Abies", "Pinus", "Betula", "Corylus", "Dec_Querc", "herb")) %>%
  ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
  geom_area() +
  scale_x_reverse(limits = c(11.7, 0),
    breaks = seq(0, 12, by = 0.5),
    labels = ifelse(seq(0, 12, by = 0.5) % 1 == 0, as.character(seq(0, 12, by = 0.5)), "
  scale_fill_manual(values = c("#BFB2FF", "#422CB2", "#749B58", "#99CC00", "#AE1F63", "#9ECAE1")) +
  theme_classic() +
  labs(title = "Marboré \n (2612 m a.s.l.)") +
  theme(axis.title.x = element_blank(),
    axis.title.y = element_blank(),
    legend.position = "none",
    plot.margin = margin(0, 0, 0, 0, "cm"),
    plot.title.position = "panel") +
  geom_vline(xintercept = periods_lines_holocene, color = "black", linetype = "dashed", linewidth = 0.3)

p11.bsm.v <- bsm_pollen %>%
  filter(taxon %in% c("Betula", "Corylus", "Dec_Querc", "Abies", "Pinus", "herb")) %>%
  mutate(taxon = factor(taxon)) %>%
  mutate(taxon = fct_relevel(taxon, "Abies", "Pinus", "Betula", "Corylus", "Dec_Querc", "herb")) %>%
  ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
  geom_area() +
  scale_x_reverse(limits = c(11.7, 0),
    breaks = seq(0, 12, by = 0.5),
    labels = ifelse(seq(0, 12, by = 0.5) % 1 == 0, as.character(seq(0, 12, by = 0.5)), "
  scale_fill_manual(values = c("#BFB2FF", "#422CB2", "#749B58", "#99CC00", "#AE1F63", "#9ECAE1")) +
  theme_classic() +
  labs(title = "Basa de la Mora \n (1914 m a.s.l.)") +
  theme(axis.title.x = element_blank(),
    axis.title.y = element_blank(),
    legend.position = "none",
    plot.margin = margin(0, 0, 0, 0, "cm"),
    plot.title.position = "panel") +
  geom_vline(xintercept = periods_lines_holocene, color = "black", linetype = "dashed", linewidth = 0.3)

p11.bassanera.v <- bassanera_pollen %>%
  filter(taxon %in% c("Betula", "Corylus", "Quercus (deciduous)", "Abies", "Pinus", "herb")) %>%
  mutate(taxon = factor(taxon)) %>%
  mutate(taxon = fct_relevel(taxon, "Abies", "Pinus", "Betula", "Corylus", "Quercus (deciduous)", "herb
  mutate(taxon = fct_recode(taxon,
    "Deciduous~italic(Quercus)" = "Quercus (deciduous)",
    "italic(Abies)" = "Abies",
    "italic(Pinus)" = "Pinus",
    "italic(Betula)" = "Betula",

```

```

        "italic(Corylus)" = "Corylus",
        "Herbaceous~taxa" = "herb")) %>%
ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
geom_area()+
scale_x_reverse(limits = c(11.7, 0),
                breaks = seq(0, 12, by = 0.5),
                labels = ifelse(seq(0, 12, by = 0.5) %% 1 == 0, as.character(seq(0, 12, by = 0.5)), "
scale_fill_manual(values = c("#BFB2FF", "#422CB2", "#749B58", "#99CC00", "#AE1F63", "#9ECAE1"),
                    labels = c(expression(italic(Abies)), expression(italic(Pinus)), expression(italic(
                        expression(italic(Corylus)), expression("Deciduous"-italic(Quercus)), ex
guides (fill = guide_legend(title = "Plant group")) +
theme_classic() +
labs(y = "Pollen percentages", title = "Bassa Nera \n (1891 m a.s.l.)") +
theme(axis.title.x = element_blank(),
      legend.position = "right",
      legend.text = element_text(size = 12),
      legend.title = element_text(size = 14),
      axis.title.y = element_text(size = 14),
      plot.margin = margin(0, 0, 0, 0, "cm"),
      plot.title.position = "panel") +
geom_vline(xintercept = periods_lines_holocene, color = "black", linetype = "dashed", linewidth = 0.3)

p11.marbore.v/p11.bsm.v/p11.bassanera.v/p11.portalet.v/p11.tramacastilla.v +
plot_layout(ncol = 1, nrow = 5, guides = "collect") +
theme(plot.margin = margin(0, 0, 0, 0, "pt")) #save pdf A4 horizontal

```

```

## Warning: Removed 6 rows containing non-finite outside the scale range
## ('stat_align()').

```

```

## Warning: Removed 30 rows containing non-finite outside the scale range
## ('stat_align()').

```

```

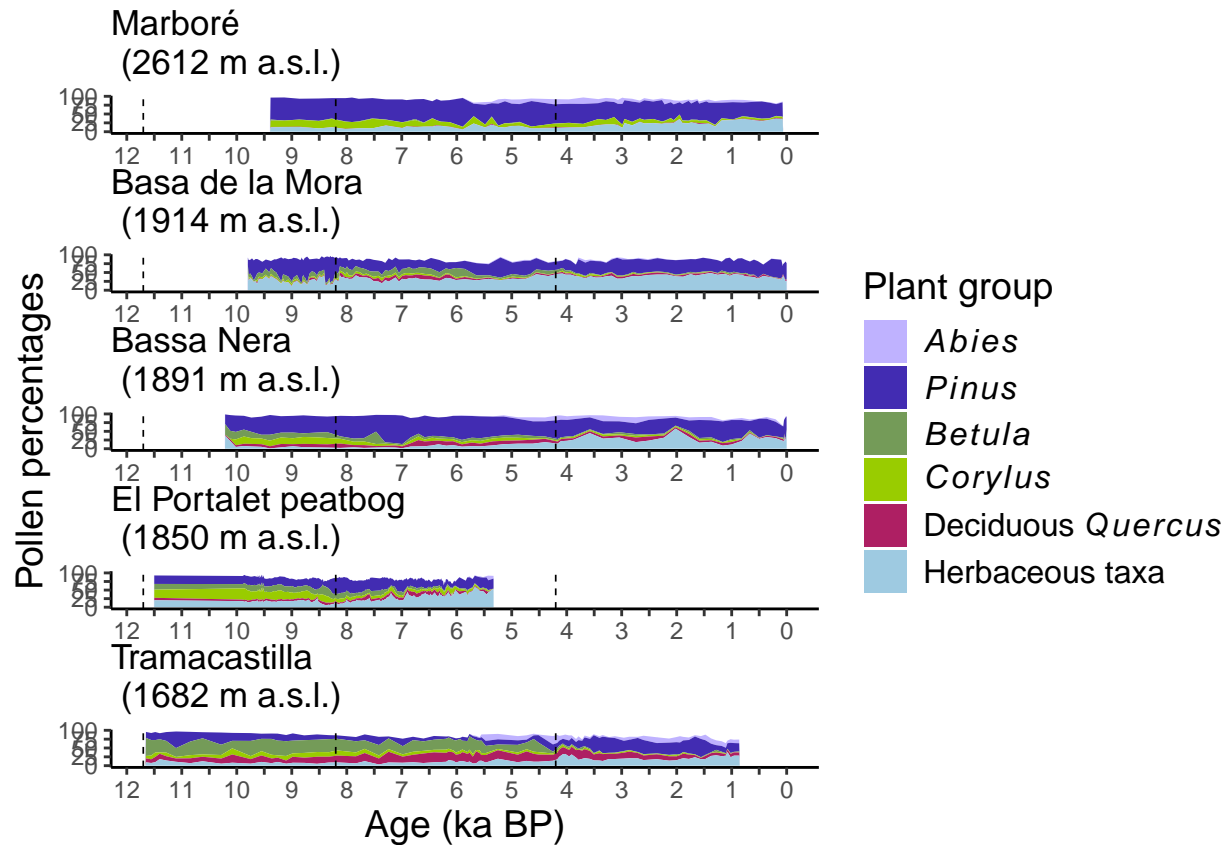
## Warning: Removed 282 rows containing non-finite outside the scale range
## ('stat_align()').

```

```

## Warning: Removed 162 rows containing non-finite outside the scale range
## ('stat_align()').

```



13.3 Plot Lateglacial records

```
periods_lines_lateglacial <- c(11.7, 12.9, 14, 14.9) # to add Lateglacial periods
p12.portalet <- portalet_pollen %>%
  filter(taxon %in% c("deciduous_trees", "Juniperus", "steppe", "Artemisia", "other_asteraceae")) %>%
  mutate(taxon = factor(taxon)) %>%
  mutate(taxon = fct_relevel(taxon, "deciduous_trees", "Juniperus", "steppe", "Artemisia", "other_asteraceae")) %>%
  ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
  geom_area() +
  scale_x_reverse(limits = c(15, 11),
                 breaks = seq(11, 15, by = 0.5),
                 labels = function(x) ifelse(x == floor(x), as.integer(x), x)) +
  scale_y_continuous(limits = c(0, 100)) +
  scale_fill_manual(values = c("#749B58", "#993D00", "#FFFF00", "#7EC3E5", "#FF99BF")) +
  theme_classic() +
  labs(title = "El Portalet peatbog \n (1850 m a.s.l.)") +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        legend.position = "none",
        plot.margin = margin(0, 0, 0, 0, "cm"),
        plot.title.position = "panel") +
  geom_vline(xintercept = periods_lines_lateglacial, color = "black", linetype = "dashed", linewidth = 1)

p12.pde <- PDE_pollen %>%
```

```

filter(taxon %in% c("deciduous_trees", "Juniperus", "steppe", "Artemisia", "other_asteraceae")) %>%
mutate(taxon = factor(taxon)) %>%
mutate(taxon = fct_relevel(taxon, "deciduous_trees", "Juniperus", "steppe", "Artemisia", "other_asteraceae"))
ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
geom_area()+
scale_x_reverse(limits = c(15, 11),
breaks = seq(11, 15, by = 0.5),
labels = function(x) ifelse(x == floor(x), as.integer(x), x)) +
scale_y_continuous(limits = c(0, 100)) +
scale_fill_manual(values = c("#749B58", "#993D00", "#FFFF00", "#7EC3E5", "#FF99BF")) +
theme_classic() +
labs(title = "Pllan d'Están \n (1840 m a.s.l.)") +
theme(axis.title.x = element_blank(),
axis.title.y = element_blank(),
legend.position = "none",
plot.margin = margin(0, 0, 0, 0, "cm"),
plot.title.position = "panel") +
geom_vline(xintercept = periods_lines_lateglacial, color = "black", linetype = "dashed", linewidth = 1)

p12.tram <- tram20_pollen %>%
select(-c(`No of taxa`, Lycopodium, `Pteropsida trilete undiff`:`Charcoal <150`)) %>%
rowwise() %>%
mutate(steppe = sum(c_across(c(`Ephedra distachya`, `Ephedra fragilis`, `Asteraceae Artemisia`, `Asteraceae Cichorioideae`, `Asteraceae Carduae`, Brassicaceae, Chenopodiaceae,
other_asteraceae = sum(c_across(c(`Asteraceae Centaurea`, `Asteraceae Cichorioideae`,
`Asteraceae Cichorioideae`, `Asteraceae Carduae`))), na.rm = TRUE)

ungroup() %>%
rename("deciduous_trees" = "Deciduous trees",
"Artemisia" = "Asteraceae Artemisia") %>%
pivot_longer(cols = c("Abies": "other_asteraceae"),
values_to = 'pollen_counts',
names_to = 'taxon') %>%
mutate_at("pollen_counts", ~replace(., is.na(.), 0)) %>%
group_by(SampleId, taxon) %>%
mutate(percentage = (pollen_counts/`Terrestrial pollen sum`)*100) %>%
filter(taxon %in% c("deciduous_trees", "Juniperus", "steppe", "Artemisia", "other_asteraceae")) %>%
mutate(taxon = factor(taxon)) %>%
mutate(taxon = fct_relevel(taxon, "deciduous_trees", "Juniperus", "steppe", "other_asteraceae", "Artemisia"))
ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
geom_area()+
scale_x_reverse(limits = c(15, 11),
breaks = seq(11, 15, by = 0.5),
labels = function(x) ifelse(x == floor(x), as.integer(x), x)) +
scale_y_continuous(limits = c(0, 100)) +
scale_fill_manual(values = c("#749B58", "#993D00", "#FFFF00", "#7EC3E5", "#FF99BF"),
labels = c(expression("Deciduous trees"), expression(italic(Juniperus)),
expression("Steppe taxa"), expression("Other Asteraceae"), expression(italic(Artemisia)))
guides (fill = guide_legend(title = "Plant group")) +
theme_classic() +
labs(title = "Tramacastilla lake \n (1682 m a.s.l.)", y = "Pollen percentages") +
theme(axis.title.x = element_blank(),
axis.title.y = element_text(size = 14),
legend.position = "right",

```

```

    legend.text = element_text(size = 12),
    legend.title = element_text(size = 14),
    plot.margin = margin(0, 0, 0, 0, "cm"),
    plot.title.position = "panel") +
  geom_vline(xintercept = periods_lines_lateglacial, color = "black", linetype = "dashed", linewidth = 0.5)

p12.estanya <- estanya_pollen %>%
  filter(taxon %in% c("deciduous_trees", "Juniperus", "steppe", "Artemisia", "other_asteraceae")) %>%
  mutate(taxon = factor(taxon)) %>%
  mutate(taxon = fct_relevel(taxon, "deciduous_trees", "Juniperus", "steppe", "Artemisia", "other_asteraceae"))
ggplot(aes(x = age/1000, y = percentage, group = taxon, fill = taxon)) +
  geom_area() +
  scale_x_reverse(limits = c(15, 11),
    breaks = seq(11, 15, by = 0.5),
    labels = function(x) ifelse(x == floor(x), as.integer(x), x)) +
  scale_y_continuous(limits = c(0, 100)) +
  scale_fill_manual(values = c("#749B58", "#993D00", "#FFFF00", "#7EC3E5", "#FF99BF")) +
  theme_classic() +
  labs(title = "Estanya \n (670 m a.s.l.)", x = "Age (ka BP)") +
  theme(axis.title.x = element_text(size = 14),
    axis.title.y = element_blank(),
    legend.position = "none",
    plot.margin = margin(0, 0, 0, 0, "cm"),
    plot.title.position = "panel") +
  geom_vline(xintercept = periods_lines_lateglacial, color = "black", linetype = "dashed", linewidth = 0.5)

p12.portalet/p12.pde/p12.tram/p12.estanya +
  plot_layout(ncol = 1, nrow = 4, guides = "collect") +
  theme(plot.margin = margin(0, 0, 0, 0, "pt")) #save pdf A4 horizontal

```

```

## Warning: Removed 1035 rows containing non-finite outside the scale range
## ('stat_align()').

```

```

## Warning: Removed 65 rows containing non-finite outside the scale range
## ('stat_align()').

```

```

## Warning: Removed 445 rows containing non-finite outside the scale range
## ('stat_align()').

```

```

## Warning: Removed 290 rows containing non-finite outside the scale range
## ('stat_align()').

```

