Supplementary Materials

This document contains some extra bits of information to supplement the poster, as well as figures that just weren't good enough to make it onto the final version of the poster.

R Session

```
print(sessionInfo(), locale = F)
```

```
R version 4.4.1 (2024-06-14)
Platform: x86_64-pc-linux-gnu
Running under: Pop!_OS 22.04 LTS

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.10.0
LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.10.0

attached base packages:
[1] stats     graphics  grDevices datasets  utils     methods   base

other attached packages:
[1] leh-patterns_0.0.0.9000 here_1.0.1              mice_3.16.0
[4] forcats_1.0.0           tidyr_1.3.1             dplyr_1.1.4
[7] patchwork_1.2.0         ggplot2_3.5.1

loaded via a namespace (and not attached):
 [1] gtable_0.3.5      shape_1.4.6.1     xfun_0.47         remotes_2.5.0
 [5] htmlwidgets_1.6.4 devtools_2.4.5    lattice_0.22-5    vctrs_0.6.5
 [9] tools_4.4.1       generics_0.1.3    tibble_3.2.1      fansi_1.0.6
[13] pan_1.9           pkgconfig_2.0.3   jomo_2.7-6        Matrix_1.6-5
[17] desc_1.4.3        lifecycle_1.0.4   stringr_1.5.1     compiler_4.4.1
[21] munsell_0.5.1     codetools_0.2-19  httpuv_1.6.15     usethis_3.0.0
[25] htmltools_0.5.8.1 yaml_2.3.10       glmnet_4.1-8      urlchecker_1.0.1
[29] later_1.3.2       pillar_1.9.0      nloptr_2.1.1      MASS_7.3-61
```

```
 [33] ellipsis_0.3.2    cachem_1.1.0      sessioninfo_1.2.2 iterators_1.0.14
 [37] rpart_4.1.23      boot_1.3-30       foreach_1.5.2     mitml_0.4-5
 [41] mime_0.12         nlme_3.1-165      tidyselect_1.2.1  digest_0.6.36
 [45] stringi_1.8.4     purrr_1.0.2       splines_4.4.1     rprojroot_2.0.4
 [49] fastmap_1.2.0     grid_4.4.1        colorspace_2.1-1  cli_3.6.3
 [53] magrittr_2.0.3    pkgbuild_1.4.4    survival_3.7-0    utf8_1.2.4
 [57] broom_1.0.6       withr_3.0.1       promises_1.3.0    scales_1.3.0
 [61] backports_1.5.0   rmarkdown_2.28    nnet_7.3-19       lme4_1.1-35.5
 [65] memoise_2.0.1     shiny_1.9.1       evaluate_0.24.0   knitr_1.48
 [69] miniUI_0.1.1.1    profvis_0.3.8     rlang_1.1.4       Rcpp_1.0.13
 [73] xtable_1.8-4      glue_1.7.0        renv_1.0.7        pkgload_1.4.0
 [77] rstudioapi_0.16.0 minqa_1.2.8       jsonlite_1.8.8    R6_2.5.1
 [81] fs_1.6.4
```

Retrieving data

Download zip files from the Wellcome Osteological Research Database (website currently not accessible).

Extract all files.

```
unzip "*.zip"
```

Rename files to .csv

```
for file in $(ls .)
    do
        mv $file $file".csv"
    done
```

Importing data to R

Import all datasets containing relevant dental data into R using DuckDB.

```
con <- dbConnect(duckdb::duckdb(), dbdir = ":memory:")
dental_inventory <- dbGetQuery(con, "SELECT * FROM read_csv('<path/to/folder>/*bones_present
dental_pathology <- dbGetQuery(con, "SELECT * FROM read_csv('<path/to/folder>/*dental_path.ls
```

Total sample size: 1650

Filtering dataset

Filtering the dataset to include adult individuals with permanent dentition only. It was also necessary to combine `SITECODE` (the code for the archaeological site) and `CONTEXT` (the unique identifier for individuals within each site), as `CONTEXT` was not unique for individuals across all sites. This was done by combining `SITECODE` and `CONTEXT`: `ID` = `SITECODE_CONTEXT`.

```
dental_inventory |>
  filter(
    BONE_GP == "Permanent teeth",  # only dental data
    str_detect(AGE, "SUB-ADULT", negate = T) # only adults
  ) |>
  separate_wider_delim(
    BONES_PRESENT,
    delim = " ",
    names = c("region", "side", "type")
  ) |>
  mutate(
    region = case_match(
      region,
      "Maxilla" ~ "U",
      "Mandible" ~ "L"
    ),
    tooth = paste0(region, side, type),
    tooth = case_when(
      stringr::str_detect(tooth, "C$") ~ paste0(tooth, "1"), # add 1 to canine notation (e.g
      TRUE ~ tooth
    )
  ) |>
    # create a unique identifier (some CONTEXT is repeated across sites)
  mutate(ID = paste0(SITECODE, "_", CONTEXT)) |>
  select(ID, tooth)
```

Sample sizes were calculated for the following samples:

adult individuals with complete dentitions (incl. M3s),

```
# including M3s

dental_inventory_long |>
  group_by(ID) |>
  summarise(n_teeth = sum(presence)) |>
```

```
  filter(n_teeth == 32) |>
  nrow() # 567
```

```
[1] 567
```

adult individuals with complete dentitions AND all teeth scored for LEH lesions (incl. M3s),

```
hypoplasia_long |>
  group_by(ID) |>
  summarise(leh_scores = sum(can_score)) |>
  filter(leh_scores == 32) |>
  nrow() # 12
```

```
[1] 12
```

and Adult individuals with complete dentitions AND all teeth scored for LEH lesions AND LEH lesion present on at least one tooth (incl. M3s).

```
hypoplasia_long |>
  group_by(ID) |>
  summarise(
    leh_scores = sum(can_score),
    leh_bin = sum(leh_bin),
  ) |>
  filter(leh_scores == 32) |>
  filter(leh_bin > 0) |>
  nrow() # 7
```

```
[1] 7
```

The same counts as above, but excluding M3s.

Adult individuals with complete dentitions (excl. M3s),

```
dental_inventory_long |>
  filter(!stringi::stri_detect(tooth, regex = "M3$")) |>
  group_by(ID) |>
  summarise(n_teeth = sum(presence)) |>
  filter(n_teeth == 28) |>
  nrow() # 708
```

```
[1] 708
```

adult individuals with complete dentitions AND all teeth scored for LEH lesions (incl. M3s),

```
hypoplasia_long |>
  filter(!stringi::stri_detect(tooth, regex = "M3$")) |>
  group_by(ID) |>
  summarise(leh_scores = sum(can_score)) |>
  filter(leh_scores == 28) |>
  nrow() # 25
```

```
[1] 25
```

and adult individuals with complete dentitions AND all teeth scored for LEH lesions AND LEH lesion present on at least one tooth (incl. M3s).

```
hypoplasia_long |>
  filter(!stringi::stri_detect(tooth, regex = "M3$")) |>
  group_by(ID) |>
  summarise(
    leh_scores = sum(can_score),
    leh_bin = sum(leh_bin),
  ) |>
  filter(leh_scores == 28) |>
  filter(leh_bin > 0) |>
  nrow() # 16
```

```
[1] 16
```

Still not ideal sample sizes, but definitely an improvement over samples with M3s.

Demographics

Sex and age distributions included for context. Given the low samples size of the LEH-positive sample, there was no sense in doing any analysis on sex and age.

Figures, in case you don't like tables.

Table 1: Demographics for the full sample (n = 1418).

```
hypoplasia_long |>
  left_join(demography) |>
  distinct(ID, .keep_all = T) |>
  group_by(SEX, AGE) |>
  count(SEX, AGE) # |> _$n |> sum() # for sample size
```

```
# A tibble: 30 x 3
# Groups:   SEX, AGE [30]
   SEX     AGE                    n
   <chr>   <chr>              <int>
 1 FEMALE  ADULT 18-25 YEARS     29
 2 FEMALE  ADULT 26-35 YEARS     64
 3 FEMALE  ADULT 36-45 YEARS     74
 4 FEMALE  ADULT >46 YEARS       74
 5 FEMALE  UNCLASSIFIED ADULT     6
 6 FEMALE? ADULT 18-25 YEARS     12
 7 FEMALE? ADULT 26-35 YEARS     25
 8 FEMALE? ADULT 36-45 YEARS     34
 9 FEMALE? ADULT >46 YEARS       23
10 FEMALE? UNCLASSIFIED ADULT    10
# i 20 more rows
```

Table 2: Demographics for the LEH-positive sample (n = 16).

```
hypoplasia_present_long |>
  left_join(demography) |>
  distinct(ID, .keep_all = T) |>
  group_by(SEX, AGE) |>
  count(SEX, AGE) # |> _$n |> sum() # for sample size
```

```
# A tibble: 8 x 3
# Groups:   SEX, AGE [8]
  SEX          AGE                   n
  <chr>        <chr>             <int>
1 FEMALE       ADULT 26-35 YEARS     1
2 INTERMEDIATE ADULT 26-35 YEARS     1
3 MALE         ADULT 18-25 YEARS     4
4 MALE         ADULT 26-35 YEARS     3
5 MALE         ADULT 36-45 YEARS     3
6 MALE         ADULT >46 YEARS       1
7 MALE?        ADULT 26-35 YEARS     1
8 MALE?        ADULT 36-45 YEARS     2
```

```
# needs to be proportions?
hypoplasia_long |>
  left_join(demography) |>
  distinct(ID, .keep_all = T) |>
  mutate(leh_present = as_factor(leh_present)) |>
  ggplot(aes(x = SEX)) +
    geom_bar(position = "dodge") +
    # overlay lesions present sample
    geom_bar(data = distinct(left_join(hypoplasia_present_long, demography), ID, .keep_all =
    coord_cartesian(ylim = c(0,400)) # cut y-axis off at 400
```

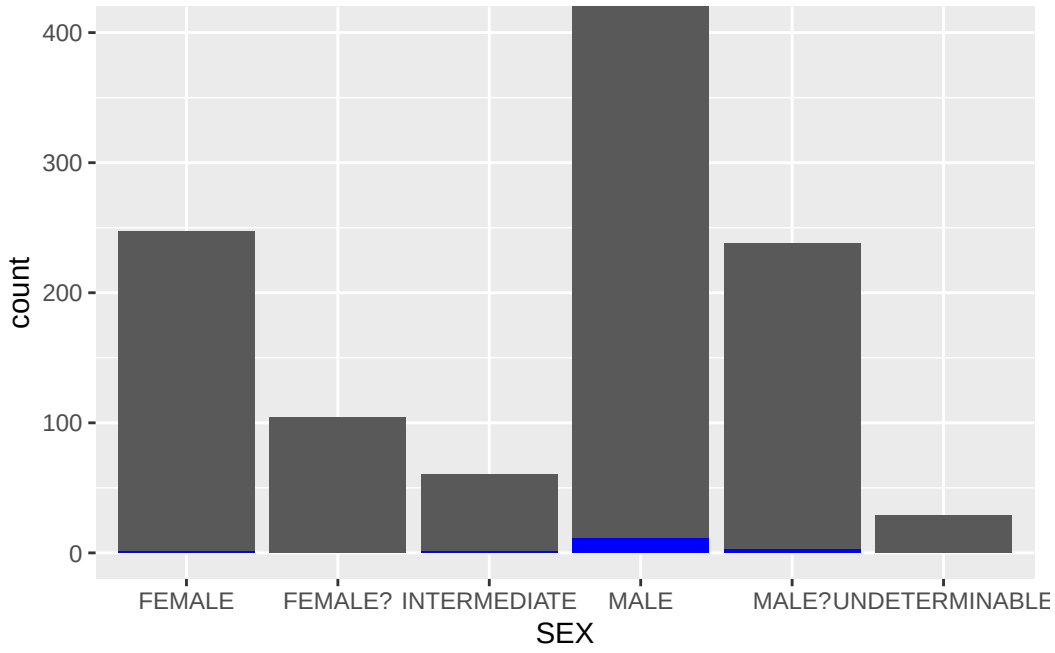

Figure 1: Distribution of sex in the full sample (charcoal; n = 1418) and the LEH-positive sample (blue; n = 16). Y-axis is cut off at 400 to be able to see the LEH-positive counts.

```
# needs to be proportions?
hypoplasia_long |>
  left_join(demography) |>
  distinct(ID, .keep_all = T) |>
  mutate(leh_present = as_factor(leh_present)) |>
  ggplot(aes(x = AGE)) +
    geom_bar(position = "dodge") +
    # overlay lesions present sample
    geom_bar(data = distinct(left_join(hypoplasia_present_long, demography), ID, .keep_all =
    coord_cartesian(ylim = c(0,400)) # cut y-axis off at 400
```

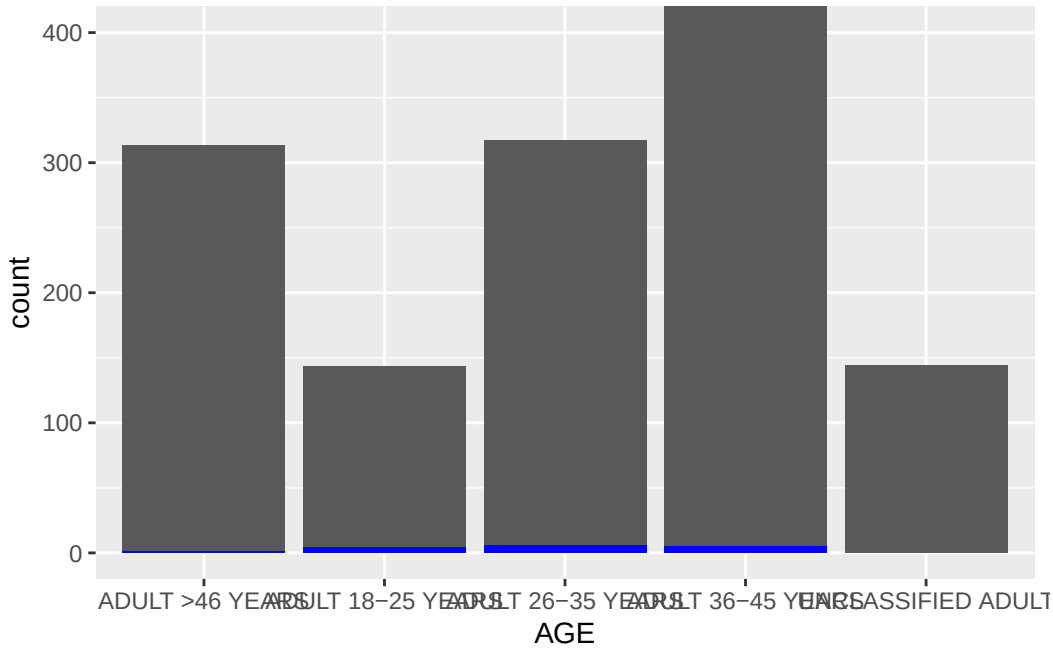

Figure 2: Distribution of age in the full sample (charcoal; n = 1418) and the LEH-positive sample (blue; n = 16). Y-axis is cut off at 400 to be able to see the LEH-positive counts.

Key Results

Most of the analysis was done on the 16 individuals that have complete LEH scores in all teeth (except M3s), AND have at least one LEH lesion present (LEH-positive). The nine other individuals with complete scores but without any lesions are not useful in this analysis.

(Missing) Data patterns

Adults with permanent dentition (excl. M3s) that were scored for LEH.

```
nrow(distinct(hypoplasia_data, ID)) # number of individuals
```

```
[1] 1418
```

```
na_pattern <- md.pattern(hypoplasia_data[,-1], plot = F)
nrow(na_pattern) - 1 # number of patterns
```

```
[1] 1248
```

Adults with a complete permanent dentition missing that were scored for LEH.

```
# can_score
nrow(distinct(hypoplasia_complete_wide, ID)) # number of individuals
```

```
[1] 711
```

```
leh_na_pattern <- md.pattern(hypoplasia_complete_wide[,-1], plot = F)
nrow(leh_na_pattern) - 1 # number of patterns
```

```
[1] 573
```

Before plotting missing data per individual, individuals with all teeth present and endentulous individuals were removed since they are not interesting (the former have no missing data and the later have only missing data). Then individuals were arranged by number of teeth,

```
# number missing teeth per individual (to order heatmaps)
missing_ordered <- dental_inventory_long |>
  #mutate(presence = presence) |>
  group_by(ID) |>
  filter(type != "m3") |>
  summarise(n_teeth = sum(as.numeric(presence))) |>
  filter(
    n_teeth != 28 &
      n_teeth != 0 # endentulous and full dentitions are not informative
  ) |>
  arrange(desc(n_teeth))

dental_inventory_long <- dental_inventory_long |>
  mutate(presence = as_factor(presence))
```

and plotted, with maxillary and mandibular teeth displayed in separate plots.

```
max_inventory_heat <- dental_inventory_long |>
  filter(
    region == "maxilla",
    ID %in% missing_ordered$ID
  ) |>
```

```
  mutate(ID = factor(ID, levels = missing_ordered$ID)) |>
  ggplot(aes(x = tooth, y = ID, fill = presence)) +
  geom_tile() +
  scale_x_discrete(position = "top", guide = guide_axis(angle = 45)) +
  theme(
    #axis.text.x.top = element_text(angle = 45, hjust = 0, vjust = 0),
    axis.text.y = element_blank(),
    axis.title.x = element_blank()
  )

man_inventory_heat <- dental_inventory_long |>
  filter(
    region == "mandible",
    ID %in% missing_ordered$ID
  ) |>
  mutate(ID = factor(ID, levels = missing_ordered$ID)) |>
  ggplot(aes(x = tooth, y = ID, fill = presence)) +
  geom_tile() +
  scale_x_discrete(position = "bottom", guide = guide_axis(angle = 45)) +
  theme(
    #axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1),
    axis.text.y = element_blank()
  )

(max_inventory_heat / man_inventory_heat) +
  plot_layout(guides = "collect", axis_titles = "collect") &
  labs(x = "Tooth", y = "Individuals", fill = "Presence") &
  scale_fill_viridis_d(option = "viridis", begin = 0.1)
```

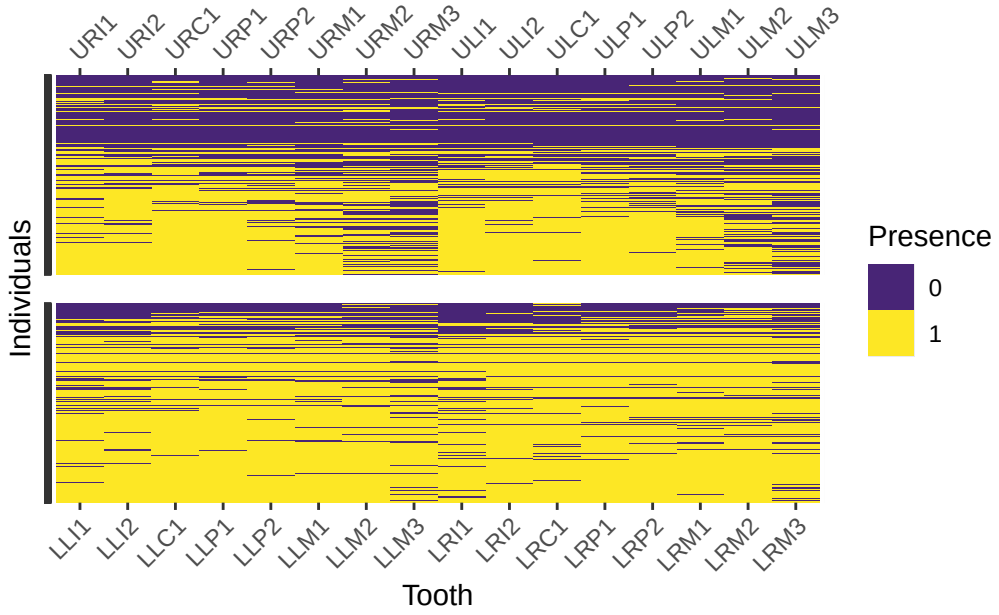

Figure 3: Heat Map of present and missing teeth for all adults with permanent dentition (n = 1650). Maxillary (top) and mandibular (bottom) teeth arranged according to position in the mouth.

The same was done for hypoplasia scores. If a tooth could be scored, it received a score of 1, if not, it received a score of 0.

```
scoreable_ordered <- hypoplasia_complete_long |>
  group_by(ID) |>
  filter(type != "m3") |>
  summarise(n_teeth = sum(as.numeric(can_score))) |>
  filter(
    n_teeth != 28 &
      n_teeth != 0 # endentulous and full dentitions are not informative
  ) |>
  arrange(desc(n_teeth))
```

Then separate heatmaps were made for maxillary and mandibular teeth.

```
max_scoreable_heat <- hypoplasia_complete_long |>
  filter(
    region == "maxilla",
```

```
    ID %in% scoreable_ordered$ID
  ) |>
  mutate(ID = factor(ID, levels = scoreable_ordered$ID)) |>
  ggplot(aes(x = tooth, y = ID, fill = as_factor(can_score))) +
  geom_tile() +
  scale_x_discrete(position = "top", guide = guide_axis(angle = 45)) +
  theme(
    #axis.text.x.top = element_text(angle = 45, hjust = 0, vjust = 0),
    axis.text.y = element_blank(),
    axis.title.x = element_blank()
  )

man_scoreable_heat <- hypoplasia_complete_long |>
  filter(
    region == "mandible",
    ID %in% scoreable_ordered$ID
  ) |>
  mutate(ID = factor(ID, levels = scoreable_ordered$ID)) |>
  ggplot(aes(x = tooth, y = ID, fill = as_factor(can_score))) +
  geom_tile() +
  scale_x_discrete(guide = guide_axis(angle = 45)) +
  theme(
    #axis.text.x.top = element_text(angle = 45, hjust = 0, vjust = 0),
    axis.text.y = element_blank(),
    #axis.title.x = element_blank()
  )

(max_scoreable_heat / man_scoreable_heat) +
  plot_layout(guides = "collect", axis_titles = "collect") &
  labs(x = "Tooth", y = "Individuals", fill = "Scoreable") &
  scale_fill_viridis_d(option = "viridis", begin = 0.1)
```

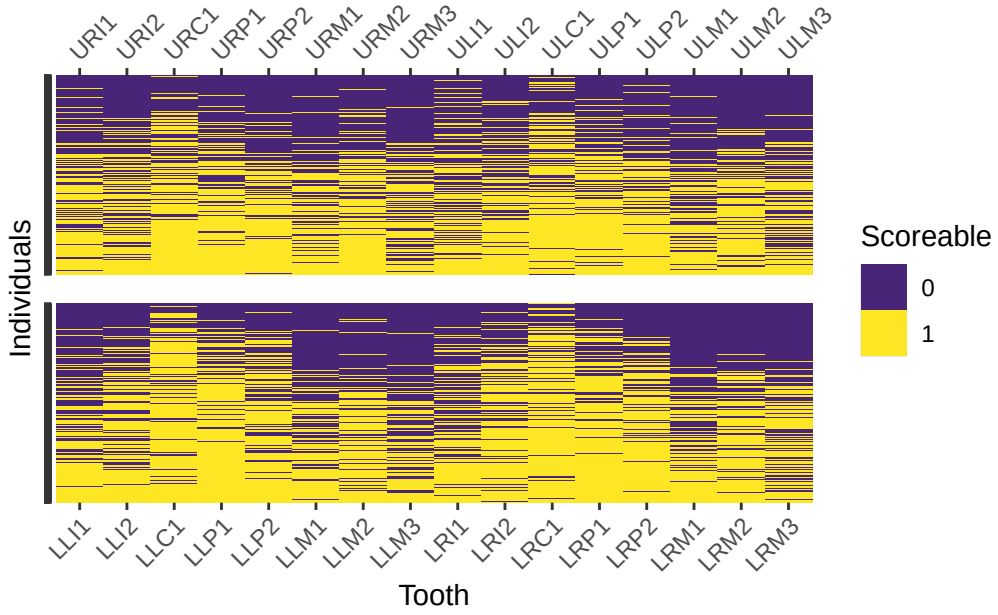

Figure 4: Heat Map of whether teeth could be scored for hypoplasia in the full sample (n = 1418). Maxillary (top) and mandibular (bottom) teeth arranged according to position in the mouth.

LEH lesion patterns

To explore patterns of the presence and absence of LEH using the `mice` package, scores of 0, i.e. absence of a lesion, were converted to `NA`.

```
hypoplasia_present_wide <- hypoplasia_present_long |>
  select(ID, tooth, score) |>
  mutate(score = if_else(score == 0, NA, 1)) |>
  pivot_wider(names_from = "tooth", values_from = "score")
```

This allowed us to use the missing data functions on the hypoplasia scores to explore score patterns, which was done on the LEH-positive sample (n = 16).

```
leh_score_pattern <- md.pattern(hypoplasia_present_wide[,-1])
```

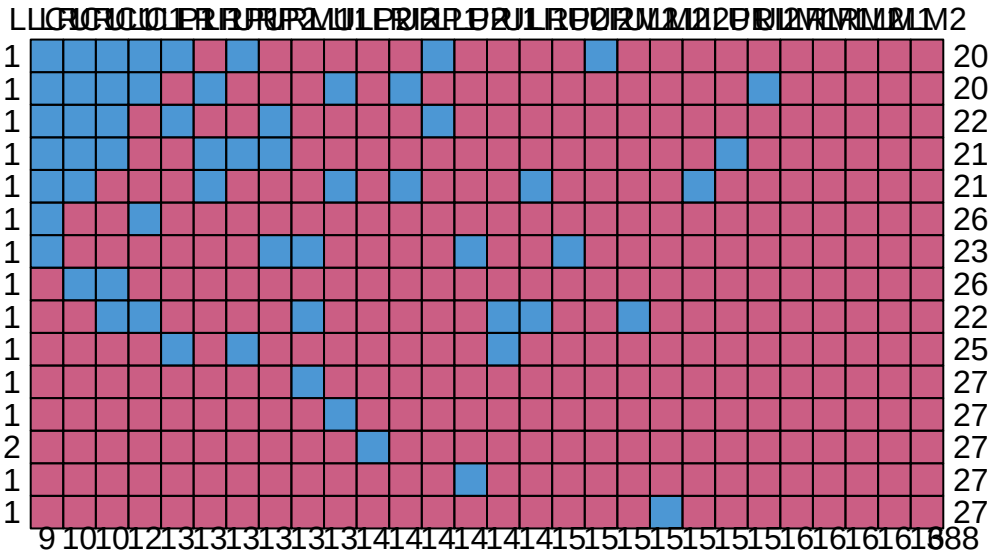

Figure 5: Pattern of scores in the LEH-positive sample. Each row represents a different pattern of scores. Row names is the number of times the pattern occurs in the sample.

Number of patterns: 15

Number of insults per tooth, tooth type, and tooth class in the LEH-positive sample. Calculated as the number of affected teeth divided by the number of teeth present in the sample.

```
# Convert scores of 0 to NA
hypoplasia_present_na <- hypoplasia_present_wide |>
  mutate(across(-ID, \(x) if_else(x == 0, NA, x)))

score_pairs <- md.pairs(hypoplasia_present_na[,-1])
```

(A)Symmetry in LEH insults.

Symmetry was calculated as agreement between tooth isomeres and antimeres, i.e., if either both teeth (in an antimere or isomere pair) had a lesion or had no lesion, this was considered agreement.

```
isomere_score_agreement <- hypoplasia_present_long |> # symmetry is not informative on indivi
  group_by(ID, side, type) |> # group by side to compare isomeres
  summarise(
```

Table 3: Ratios of LEH per tooth, tooth type, and tooth class, respectively, in the LEH-positive sample.

```
# tooth
hypoplasia_present_long |>
  group_by(tooth) |>
  summarise(
    n = n(),
    present = sum(leh_bin),
    .groups = "keep"
  ) |>
  mutate(prop = present / n) |>
  arrange(desc(prop))
```

```
# A tibble: 28 x 4
# Groups:   tooth [28]
   tooth     n present  prop
   <fct> <int>   <dbl> <dbl>
 1 LLC1     16       7 0.438
 2 URC1     16       6 0.375
 3 LRC1     16       6 0.375
 4 ULC1     16       4 0.25
 5 URP2     16       3 0.188
 6 URM1     16       3 0.188
 7 ULP1     16       3 0.188
 8 LLI1     16       3 0.188
 9 LRI1     16       3 0.188
10 LRP1     16       3 0.188
# i 18 more rows
```

```
# tooth type
hypoplasia_present_long |>
  group_by(type) |>
  summarise(
    n = n(),
    present = sum(leh_bin),
    .groups = "keep"
  ) |>
  mutate(prop = present / n) |>
  arrange(desc(prop))
```

```
# A tibble: 7 x 4
# Groups:   type [7]
  type      n present   prop
  <chr> <int>   <dbl>  <dbl>
1 c        64      23 0.359
2 i1       64      10 0.156
3 pm1      64       9 0.141
4 pm2      64       8 0.125
5 i2       64       5 0.0781
6 m1       64       3 0.0469
7 m2       64       2 0.0312
```

Table 4: Counts of teeth with an LEH 'present' score in the LEH-positive sample.

```
score_pairs$rr |>
  as_tibble(rownames = "tooth")
```

```
# A tibble: 28 x 29
   tooth  ULC1  ULP1  ULP2  ULM1  LRI1  LRI2  LRC1  LRP1  LRP2  LRM1  LRM2  ULI2
   <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 ULC1      4     1     0     0     1     1     2     1     0     0     0     1
 2 ULP1      1     3     0     0     0     0     2     2     0     0     0     1
 3 ULP2      0     0     2     0     0     0     0     0     0     0     0     0
 4 ULM1      0     0     0     0     0     0     0     0     0     0     0     0
 5 LRI1      1     0     0     0     3     2     3     1     0     0     0     0
 6 LRI2      1     0     0     0     2     2     2     0     0     0     0     0
 7 LRC1      2     2     0     0     3     2     6     2     0     0     0     1
 8 LRP1      1     2     0     0     1     0     2     3     0     0     0     1
 9 LRP2      0     0     0     0     0     0     0     0     1     0     0     0
10 LRM1      0     0     0     0     0     0     0     0     0     0     0     0
# i 18 more rows
# i 16 more variables: URP1 <dbl>, URP2 <dbl>, URM1 <dbl>, URM2 <dbl>,
#   ULM2 <dbl>, LLI1 <dbl>, LLI2 <dbl>, LLC1 <dbl>, LLP1 <dbl>, LLP2 <dbl>,
#   LLM1 <dbl>, LLM2 <dbl>, URC1 <dbl>, URI1 <dbl>, URI2 <dbl>, ULI1 <dbl>
```

Table 5: Counts of teeth with an LEH 'absent' score in the LEH-positive sample.

```
score_pairs$mm |>
  as_tibble(rownames = "tooth")
```

```
# A tibble: 28 x 29
   tooth  ULC1  ULP1  ULP2  ULM1  LRI1  LRI2  LRC1  LRP1  LRP2  LRM1  LRM2  ULI2
   <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 ULC1     12    10    10    12    10    11     8    10    11    12    12    12
 2 ULP1     10    13    11    13    10    11     9    12    12    13    13    13
 3 ULP2     10    11    14    14    11    12     8    11    13    14    14    13
 4 ULM1     12    13    14    16    13    14    10    13    15    16    16    15
 5 LRI1     10    10    11    13    13    13    10    11    12    13    13    12
 6 LRI2     11    11    12    14    13    14    10    11    13    14    14    13
 7 LRC1      8     9     8    10    10    10    10     9     9    10    10    10
 8 LRP1     10    12    11    13    11    11     9    13    12    13    13    13
 9 LRP2     11    12    13    15    12    13     9    12    15    15    15    14
10 LRM1     12    13    14    16    13    14    10    13    15    16    16    15
# i 18 more rows
# i 16 more variables: URP1 <dbl>, URP2 <dbl>, URM1 <dbl>, URM2 <dbl>,
#   ULM2 <dbl>, LLI1 <dbl>, LLI2 <dbl>, LLC1 <dbl>, LLP1 <dbl>, LLP2 <dbl>,
#   LLM1 <dbl>, LLM2 <dbl>, URC1 <dbl>, URI1 <dbl>, URI2 <dbl>, ULI1 <dbl>
```

```
    n = n(), # sanity check
    score = sum(leh_bin)
  ) |>
  mutate(
    agree = if_else(score == 1, FALSE, TRUE)
  ) |>
  ungroup()

# antimere symmetry

antimere_score_agreement <- hypoplasia_present_long |>
  group_by(ID, region, type) |> # group by region to compare antimeres
  summarise(
    n = n(), # sanity check
    score = sum(leh_bin)
  ) |>
  mutate(
    agree = if_else(score == 1, FALSE, TRUE)
  ) |>
  ungroup()
```

Plots

```
isomere_symm_plot <- isomere_score_agreement |>
  ggplot(aes(x = type, y = ID, fill = agree)) +
    geom_tile() +
    labs(caption = "Isomere symmetry of lesions.")

antimere_symm_plot <- antimere_score_agreement |>
  ggplot(aes(x = type, y = ID, fill = agree)) +
    geom_tile() +
    labs(caption = "Antimere symmetry of lesions.") +
    theme(
      axis.text.y = element_blank(),
      axis.ticks.y = element_blank()
    )

isomere_symm_plot + antimere_symm_plot + plot_layout(guides = "collect", axis_titles = "colle
```

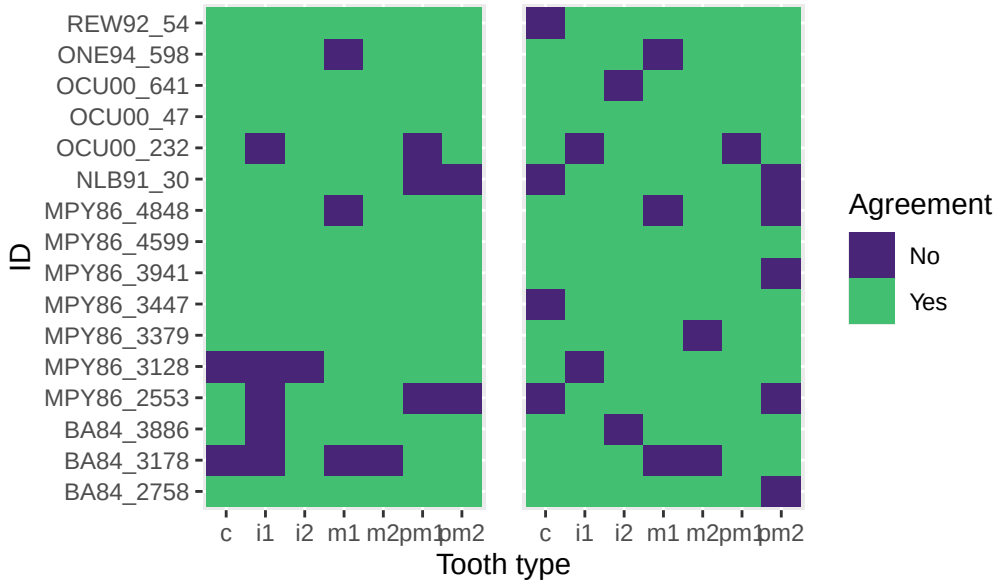

Figure 6

References

WORD, Museum of London. (2013). www.museumoflondon.org.uk/ Collections-Research/LAARC/Centre-for-Human-Bioarchaeology/ Accessed April 2024.