Structural Variant Analysis on the Exceptional Responders Cohort

2022-01-27

Contents

Packages

```
library(tidyverse)
library(paletteer)
library(ggforce)
library(cowplot)
library(viridis)
```

Data import

```
ExRes <- read_csv("ExRes.csv")

# Clean up useless columns
# ExRes[1:2] <- list(NULL) ## careful with column order
# or more safely:
ExRes$...1 <- NULL
ExRes$...2 <- NULL

# Reordering SV_chrom axis
ExRes$SV_chrom <- factor(ExRes$SV_chrom, levels=c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "1
```

I. Single-Variable Analysis

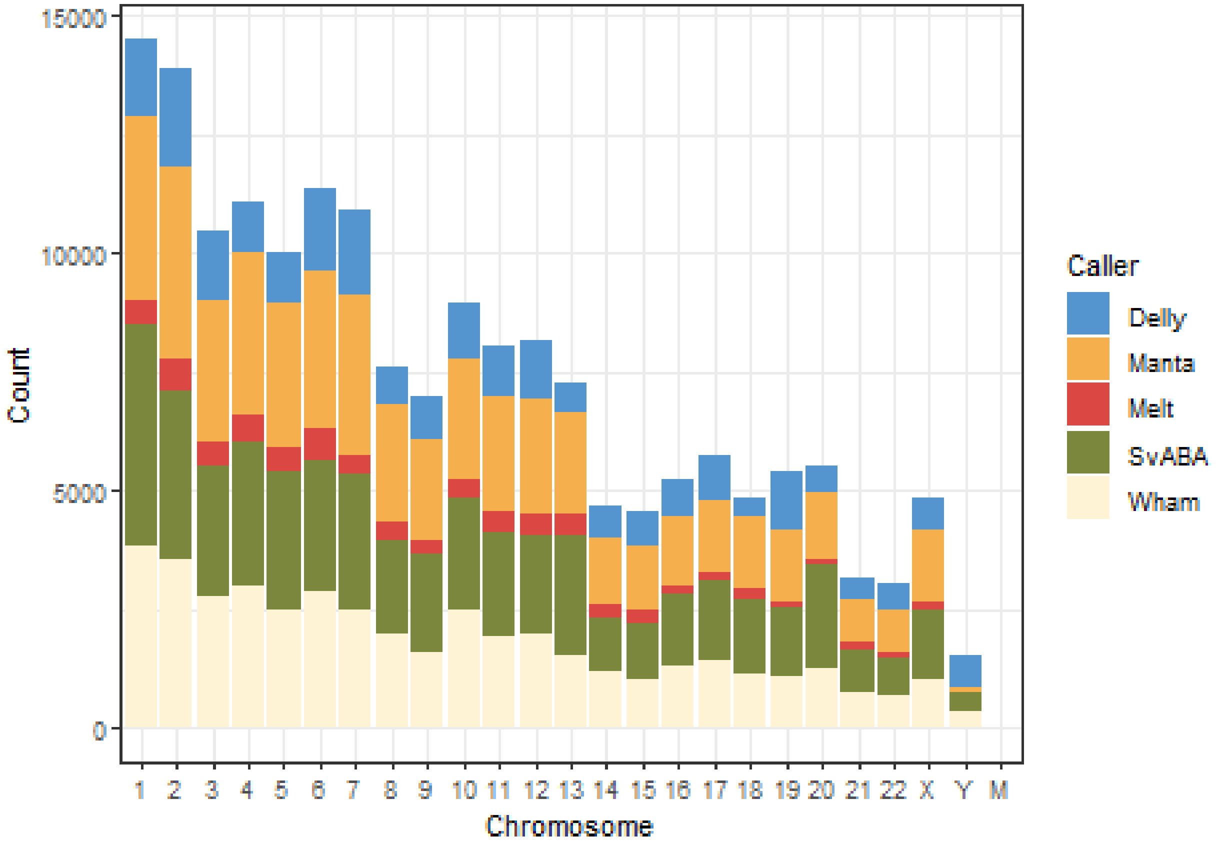
1. SV call location

```
table(ExRes$SV_chrom)
```

```
## 
##      1      2      3      4      5      6      7      8      9     10     11 
## 204069 610820 104155  46524  40185  55793  87177  19506  33269  28504 123780 
##     12     13     14     15     16     17     18     19     20     21     22 
##  65126  18002  21334  29935  57179  41346   8640  28737   9483  10537  11739 
##      X      Y      M 
##  22984   7948     51
```

1-1. SV location by caller

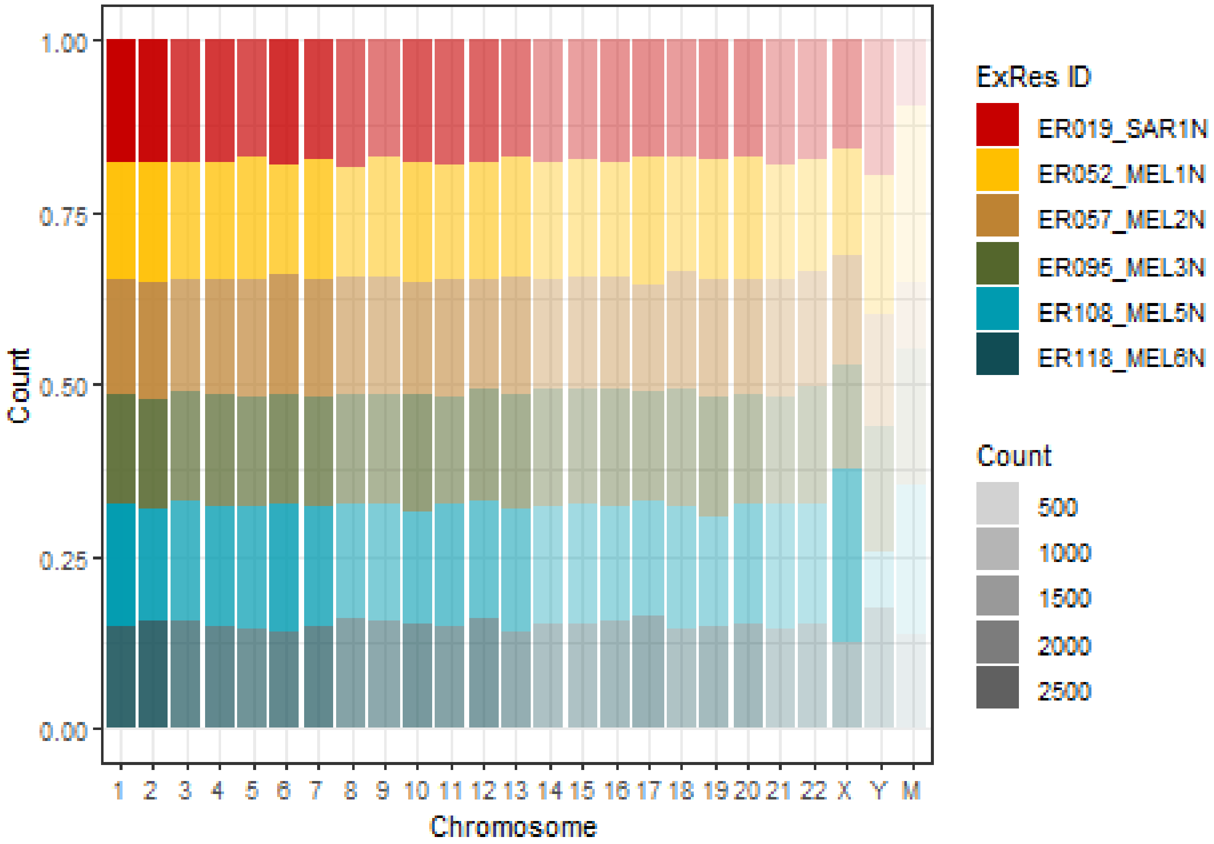
```
# there are duplicates so this isn't an accurate graph
ExRes %>% filter(Annotation_mode == "full") %>%

ggplot(aes(x = SV_chrom, fill=Caller)) +
  geom_bar() +
  theme_bw() +
  scale_fill_paletteer_d("nationalparkcolors::Badlands") +
  labs(x="Chromosome", y="Count")

#ggsave2("SV_count_chr_caller_exres.png")
```



(Superimpose line graph of chromosomal length on diaxis plot)

1-2. SV location by ExRes ID

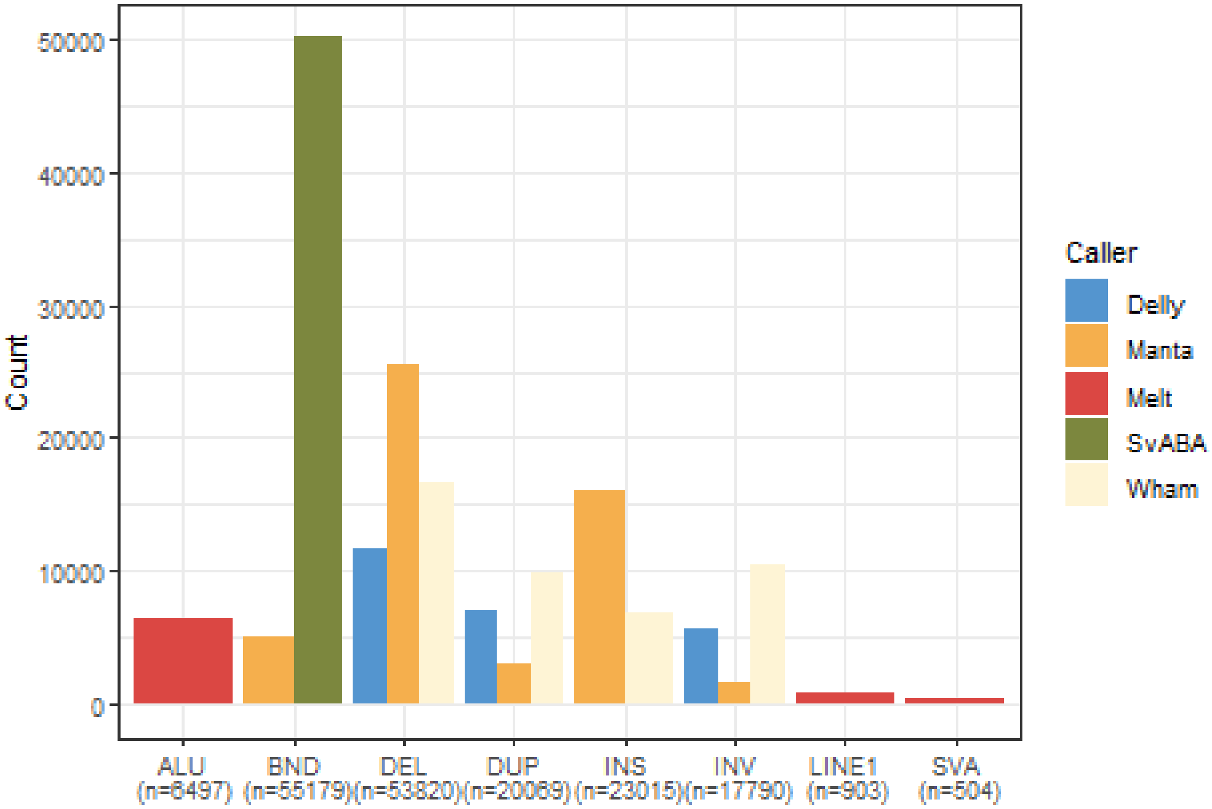
```
# there are duplicates so this isn't an accurate graph
# version A
ExRes %>% filter(Annotation_mode == "full") %>%

ggplot(aes(x = SV_chrom, fill = ExResID)) +
  geom_bar(aes(alpha=..count..), position = "fill") +
  theme_bw() +
  scale_alpha_continuous(name = "Count") +
  scale_fill_paletteer_d("calecopal::kelp1", name = "ExRes ID") +
  labs(x="Chromosome", y="Count")

#ggsave2("SV_count_dist_chr_ID_exres.png")
```



2. Types of Structural Variants

2-1. SV type between callers

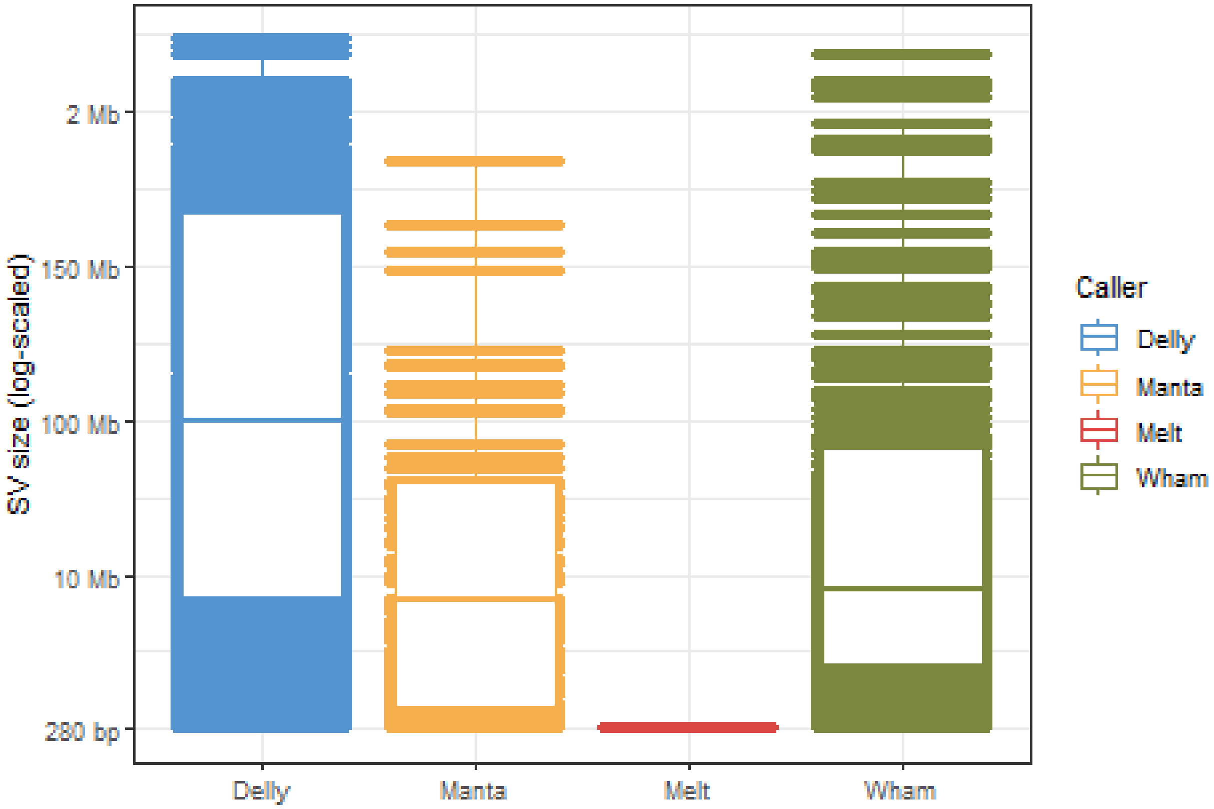
```
# again, multiple callers can call the same variant, so include this as a warning in presentation.
sample_size = ExRes %>% filter(Annotation_mode == "full") %>% group_by(SV_type) %>% summarize(num=n())

ExRes %>%
  filter(Annotation_mode == "full") %>%
  left_join(sample_size) %>%
  mutate(svtype = paste0(SV_type, "\n (n=", num, ")")) %>%

ggplot(aes(x = svtype, fill=Caller)) +
    geom_bar(position = "dodge") +
    theme_bw() + scale_fill_paletteer_d("nationalparkcolors::Badlands") +
    labs(x="", y="Count")

#ggsave2("SV_type_caller_exres.png")

rm("sample_size")
```



3. Size and length of structural variants detected

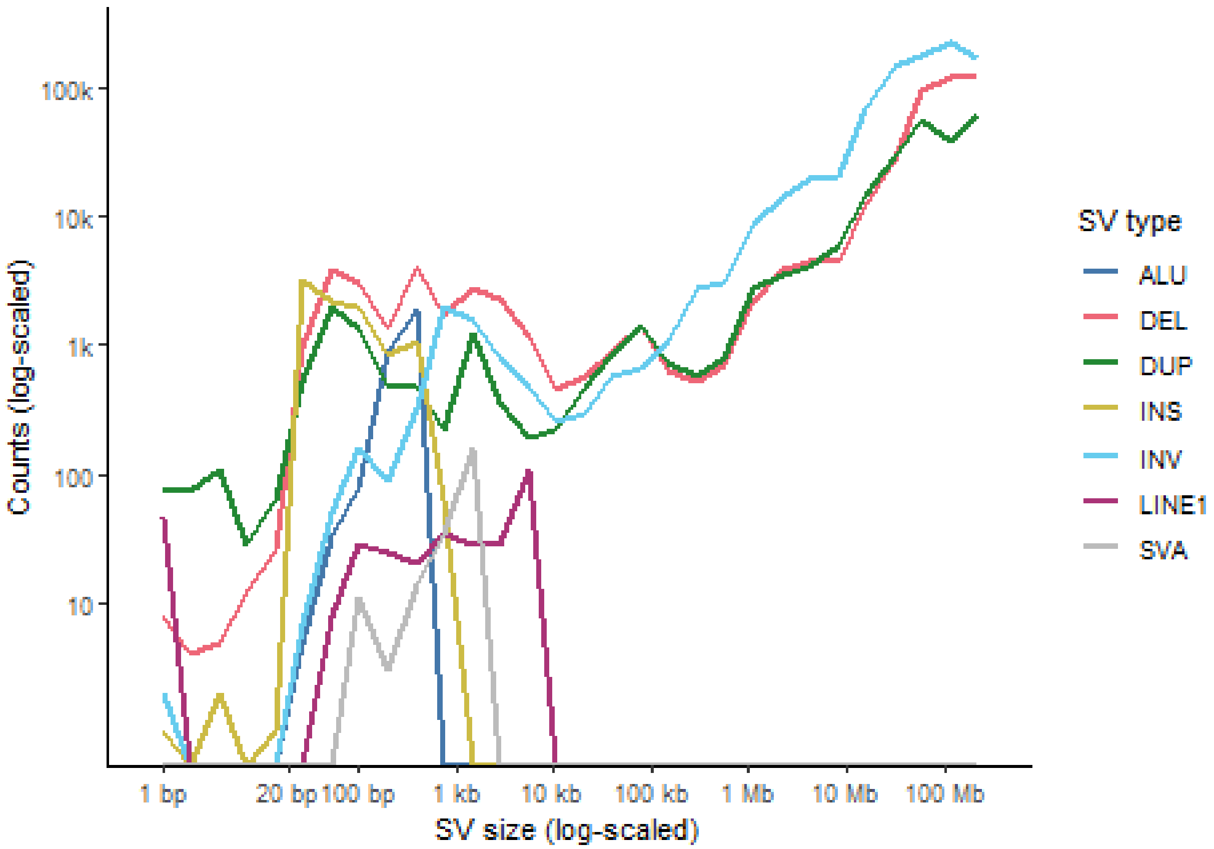
3-1. SV length by caller

```
# inaccurate graph?
ExRes %>% filter(Annotation_mode == "split") %>%
  filter(!is.na(SV_length)) %>%
  filter(SV_length != 0) %>%
  mutate(SV_length = abs(SV_length)) %>%

  ggplot(aes(x=Caller, y=SV_length, color=Caller)) +
  geom_jitter() + geom_boxplot() +
  scale_y_continuous(breaks = c(280, 5e7, 1e8, 1.5e8, 2e8),
                     labels = c("280 bp", "10 Mb", "100 Mb", "150 Mb", "2 Mb")) +
  theme_bw() + scale_color_paletteer_d("nationalparkcolors::Badlands") +
  ylab("SV size (log-scaled)") + xlab("")

#ggsave2("SV_size_exres.png")
```

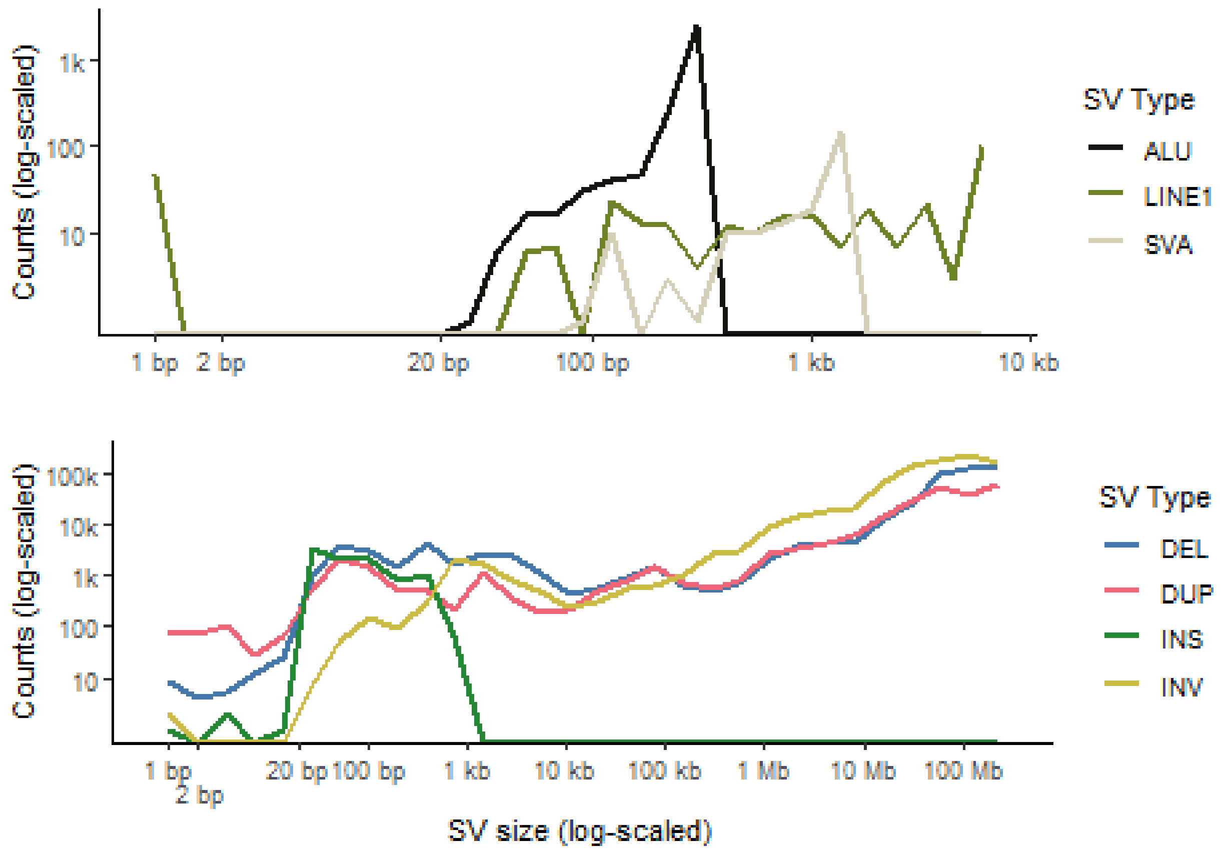


3-2. SV length vs SV type

```
ExRes %>% filter(Annotation_mode == "split") %>%
  filter(!is.na(SV_length)) %>%
  filter(SV_length != 0) %>%

  ggplot(.) + aes(x=abs(SV_length), color = SV_type) +
  geom_line(aes(y=..count..), stat="bin", size = 1.05) +
  theme_classic() +
  paletteer::scale_colour_paletteer_d("khroma::bright", name = "SV type") +
  scale_x_log10(name = "SV size (log-scaled)",
                breaks = c(1, 20, 1e2, 1e3, 1e4, 1e5, 1e6, 1e7, 1e8),
                label = c("1 bp", "20 bp", "100 bp", "1 kb", "10 kb", "100 kb", "1 Mb", "10 Mb", "100 M
  scale_y_continuous(trans = "log10",
                name = "Counts (log-scaled)",
                breaks= c(10, 10^2, 10^3, 10^4, 10^5),
                label = c("10", "100", "1k", "10k", "100k"))

#ggsave2("line_plot_SV_type_ExRes.png")
```



```
# facet_wrap/ggarrange to partition SV subsets by major/minor type
# subset 1. minor SV types (ALU, LINE1, SVA)
ExRes %>% filter(Annotation_mode == "split") %>%
  filter(SV_type %in% c("ALU", "LINE1", "SVA")) %>%
  ggplot(.) + aes(x=abs(SV_length), color = SV_type) +
  geom_line(aes(fill=..count..), stat="bin", size = 1.05) +
  # alternatively, for closed-ended plot, use
  # geom_freqpoly(size = 1.05) +
  theme_classic() +
  paletteer::scale_colour_paletteer_d("lisa::FridaKahlo", name = "SV Type") +
  scale_x_log10(name = "",
                breaks = c(1, 2, 20, 1e2, 1e3, 1e4),
                label = c("1 bp", "2 bp","20 bp", "100 bp", "1 kb", "10 kb")) +
  scale_y_log10(name = "Counts (log-scaled)",
                breaks= c(10, 10^2, 10^3, 10^4),
                label = c("10", "100", "1k", "10k")) -> p1

# subset 2. major SV types (others)
"%ni%" <- Negate("%in%")

ExRes %>% filter(Annotation_mode == "split") %>%
  filter(SV_type %ni% c("ALU", "LINE1", "SVA")) %>%
  ggplot(.) + aes(x=abs(SV_length), color = SV_type) +
  geom_line(aes(fill=..count..), stat="bin", size = 1.05) +
  # alternatively, for closed-ended plot, use
  # geom_freqpoly(size = 1.05) +
  theme_classic() +
  paletteer::scale_colour_paletteer_d("khroma::bright", name = "SV Type") +
  scale_x_log10(name = "SV size (log-scaled)",
                breaks = c(1, 2, 20, 1e2, 1e3, 1e4, 1e5, 1e6, 1e7, 1e8),
                label = c("1 bp", "\n 2 bp", "20 bp", "100 bp", "1 kb", "10 kb", "100 kb", "1 Mb", "10
  scale_y_log10(name = "Counts (log-scaled)",
                breaks= c(10, 10^2, 10^3, 10^4, 10^5),
                label = c("10", "100", "1k", "10k", "100k")) -> p2

p3 <- gridExtra::grid.arrange(p1, p2, nrow = 2)
#ggsave2("two_line_plots_SV_type_exres.png", p3)
```



Problem: 1) BND (svaba) calls ignored 2) INS = ALU + LINE1 + SVA + … (the bottom graph didn't synthesise them)

4. Number of variants detected by all methods

See Common_caller_test.R.

5. ACMG class of variants (essentially unpresentable as many caveats)

```
table(ExRes$ACMG_class)
```

```
##
##       1       3       4       5  full=1  full=3  full=4  full=5 full=NA
##   24689   47241     200    1759   11932   30229    2795  587547  876543
```

```
sample_size = ExRes %>% filter(Annotation_mode == "full") %>% group_by(ACMG_class) %>% summarize(num=n(

ExRes %>%
  filter(Annotation_mode == "full") %>%
  left_join(sample_size) %>%
  mutate(ACMG.class = paste0(ACMG_class, "\n (n=", num, ")")) %>%

ggplot(aes(x = ACMG.class, fill=SV_type)) +
    geom_bar() +
    theme_bw() +
    labs(x="ACMG class", y="Count") +
    scale_fill_discrete(name = "SV type")

rm("sample_size")
```

Try removing ACMG = NA entries:

```
filter(ExRes, !is.na(ACMG_class)) %>% filter(ACMG_class != "full=NA") %>%

ggplot(aes(x = ACMG_class, fill=SV_type)) +
    geom_bar() +
    theme_bw() +
    labs(x="ACMG class", y="count") +
    scale_fill_discrete(name = "SV type")
```

6. Detected variants affecting CDS

6-1. CDS-affecting SVs

```
table(ExRes$Location2)
```

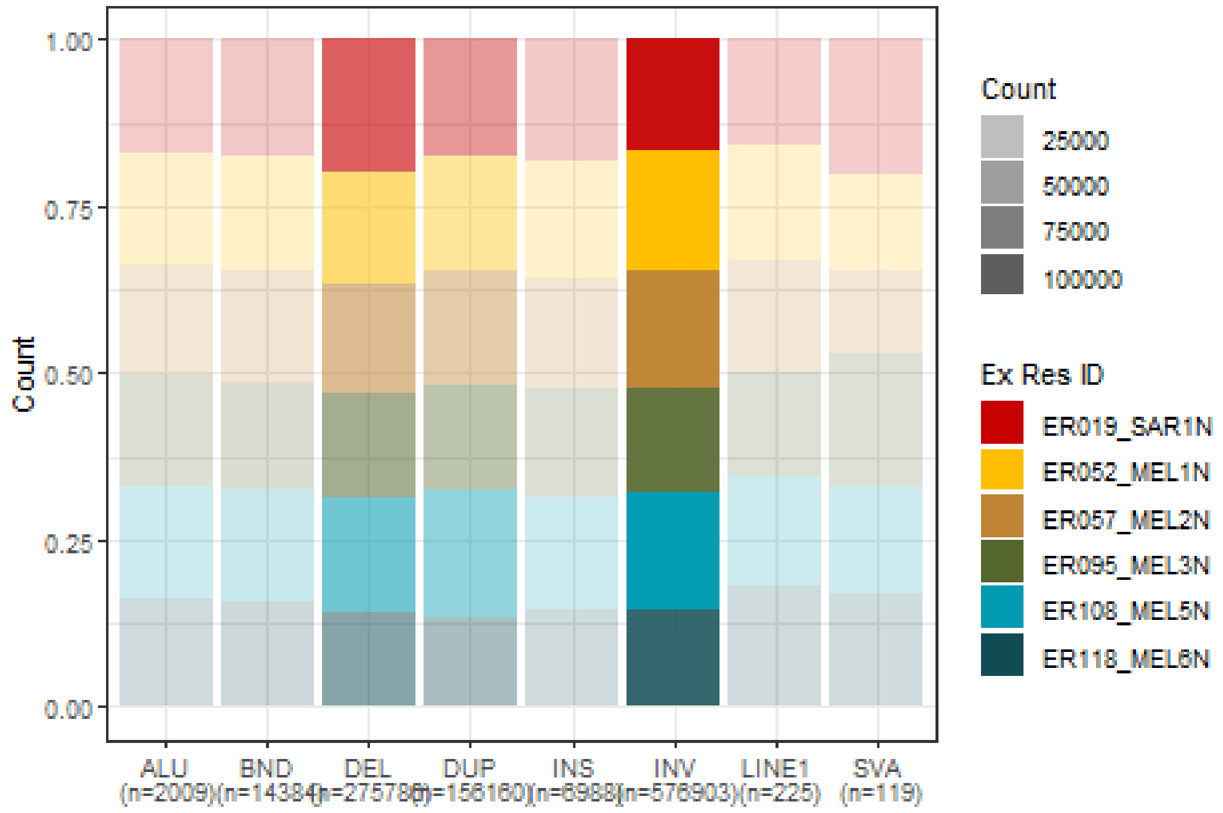
```
##
##       3'UTR       5'UTR 5'UTR-3'UTR   5'UTR-CDS         CDS   CDS-3'UTR
##        1590        9953      980128        2598       47118        2730
##         UTR
##      464929
```

Note that only split AM contains topology data (confirmed).

SV curation set: CDS-affecting only

```
# subset of AM=split data frame containing CDS-affecting entries --- AnnotSV-called
CDS <- filter(ExRes, Location2 %in% c("5'UTR-3'UTR", "5'UTR-CDS", "CDS", "CDS-3'UTR"))
```

```
sample_size = CDS %>% group_by(SV_type) %>% summarize(num=n())

CDS %>%
  left_join(sample_size) %>%
  mutate(SV.type = paste0(SV_type, "\n (n=", num, ")")) %>%

  ggplot(.) + aes(x = SV.type, fill=ExResID) +
    geom_bar(aes(alpha=..count..), position = "fill") +
    scale_alpha_continuous(name = "Count", range = c(0.2, 1)) +
    theme_bw() +
    scale_fill_paletteer_d("calecopal::kelp1", name = "Ex Res ID") +
    labs(x="", y="Count")

#ggsave2("CDS_SV_type_ID_exres.png")

rm(sample_size)
```



(run multi ANOVA: In each SV type, which patient has the most sig. deviation? Star it.)

6-2. Prioritised variant list

<1st layer: AnnotSV input>

```
Raw call set %>% filter(PASS) %>% filter(ACMG = {4,5}) %>%

[R scripts]        [R scripts]            [Task 4]
```

<2nd layer>

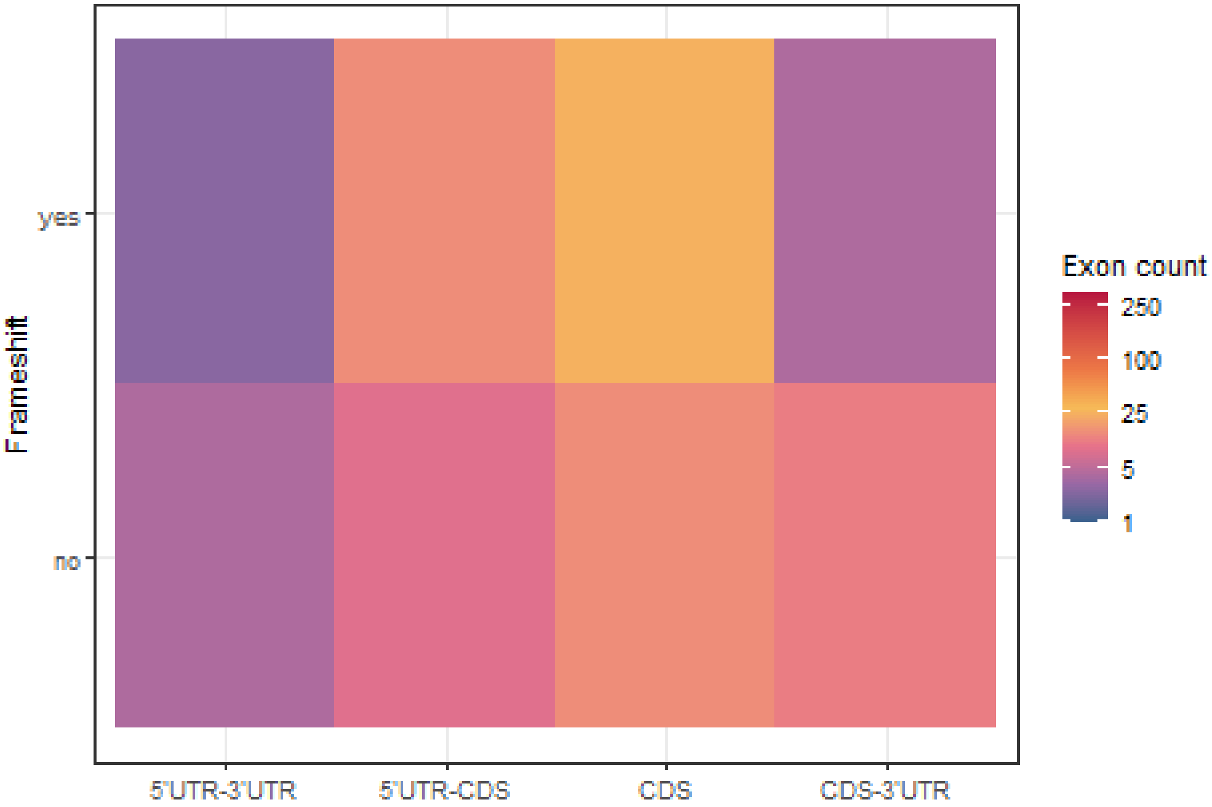
```
      filter(Called by e.g. at least 3 callers out of 5 used) %>%
      [Task 3]


      filter(CDS-affecting) %>%
      [Task 5]


      filter(Known to affect XX tissue)
      [This task]
```

List curation

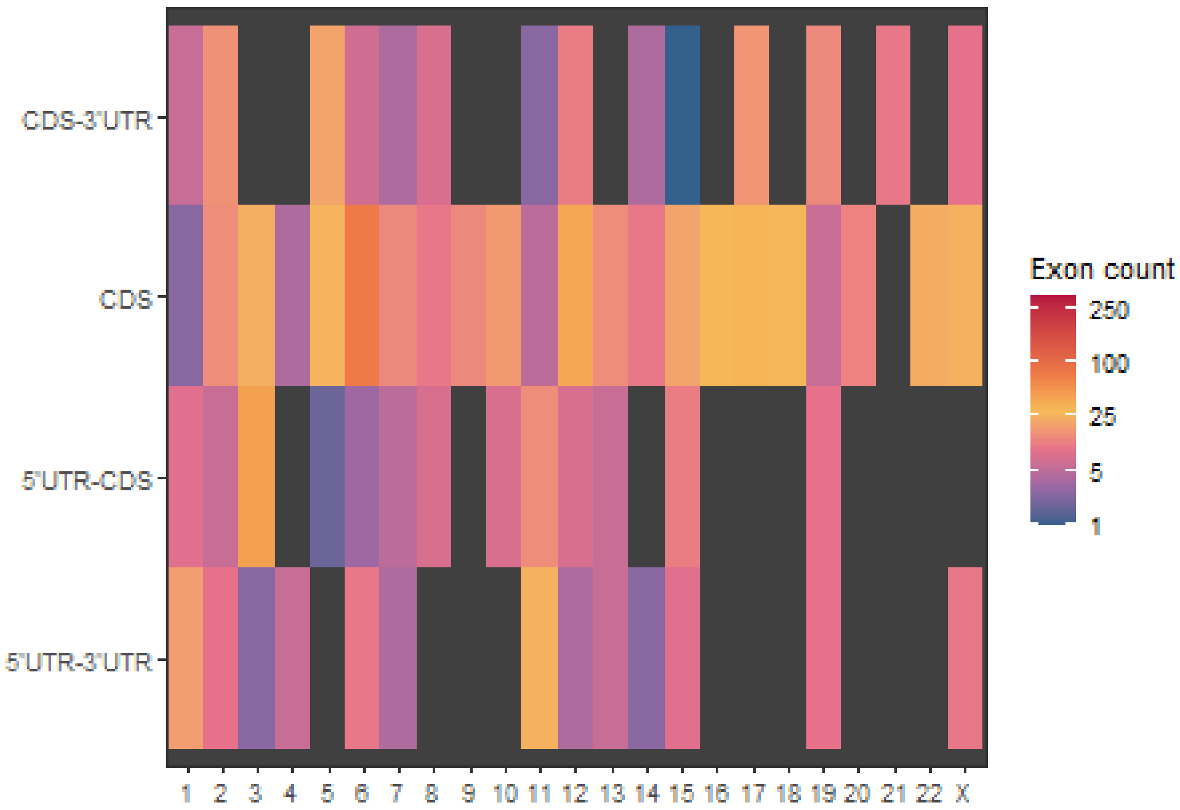
```
ExRes %>% filter(ACMG_class %in% c(4, 5, "full=4", "full=5")) %>% ## select pathogenic variants
  #filter(Caller_count >= 3) %>% ## TODO confidence of call based on num of caller consensus
  filter(Location2 %in% c("5'UTR-3'UTR", "5'UTR-CDS", "CDS", "CDS-3'UTR")) %>% ## CDS-affecting
  #filter(Tissue == "which") %>% ## TODO specify tissue affected, if available (Gene_name could help?)
  filter(abs(SV_length) <= 200000) %>% ## arbitrary limit to focus on important variants
# selecting only variables we are interested in clinically
  subset(
    select=c("SV_chrom",    "SV_start", "SV_end",   "SV_length",    "SV_type", ## basic SV info
             "ID",  "REF",  "ALT",  "FILTER", "ExResID", "Caller", "Annotation_mode", ## basic SV profi
             "CytoBand",    "Gene_name",    "Gene_count", "Location", "Location2", ## gene profile
             "Exon_count", "Frameshift", ## exon num / FS
             "Intersect_start", "Intersect_end", ## what are these?
             "Overlapped_CDS_length", "Overlapped_CDS_percent", ## CDS overlap
             "ACMG_class",  "GenCC_disease", "GenCC_moi", "GenCC_classification", ## ACMG + GenCC clini
             "ExAC_delZ",   "ExAC_dupZ",    "ExAC_cnvZ",    "ExAC_synZ",    "ExAC_misZ", ## ExAC Z scor
             "GnomAD_pLI",  "ExAC_pLI" ## GnomAD/ExAC pLIs
                )
  ) -> CDS_variants_exres

# write to csv for further use
#write.csv(CDS_variants_exres, "200K_CDS_SV_ExRes.csv")
```

6-3. Analysis on prioritised variants

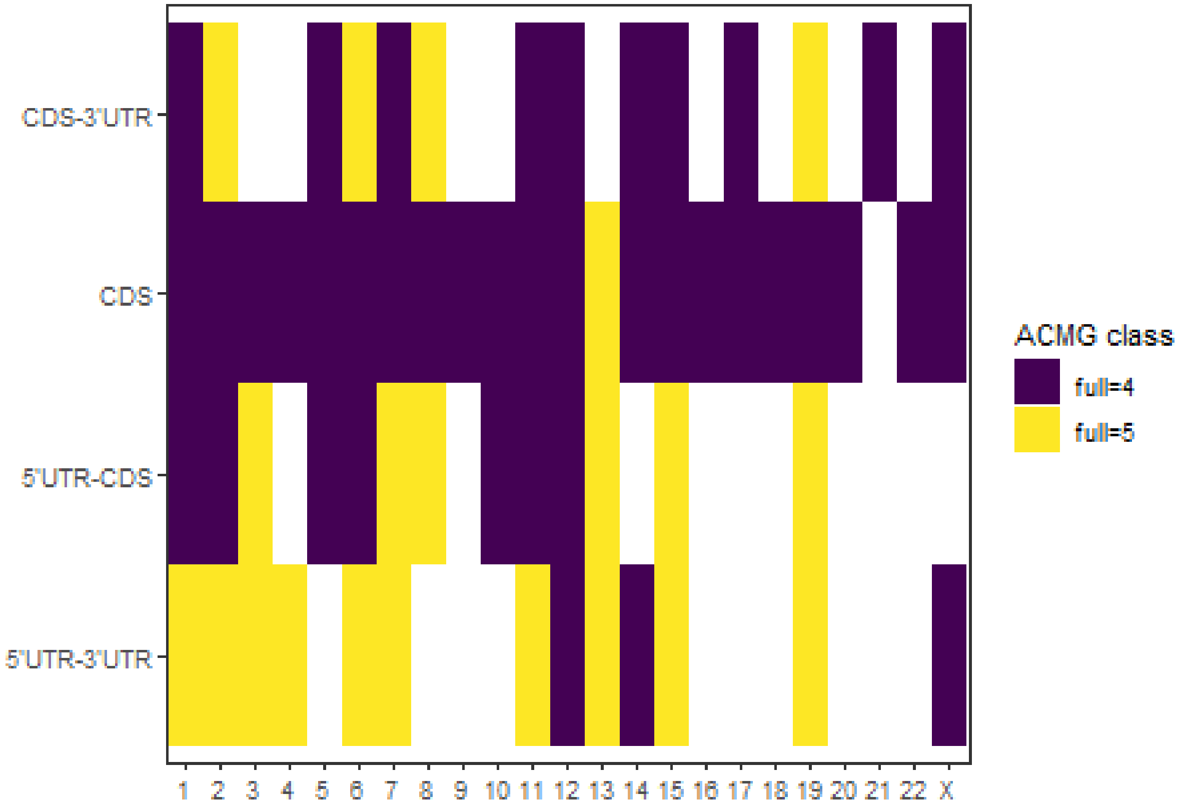
```
# exon count and frameshift profile
CDS_variants_exres %>%
  ggplot() + aes(x=Location2, y=Frameshift, fill=sqrt(sqrt(sqrt(Exon_count)))) + geom_tile() +
  scale_fill_paletteer_c("ggthemes::Sunset-Sunrise Diverging", name = "Exon count",
                          labels = paste(c(1, 5, 25, 100, 250))) +
  xlab("") + theme_bw()

#ggsave2("exon_count_frameshift_profile_exres.png")
```

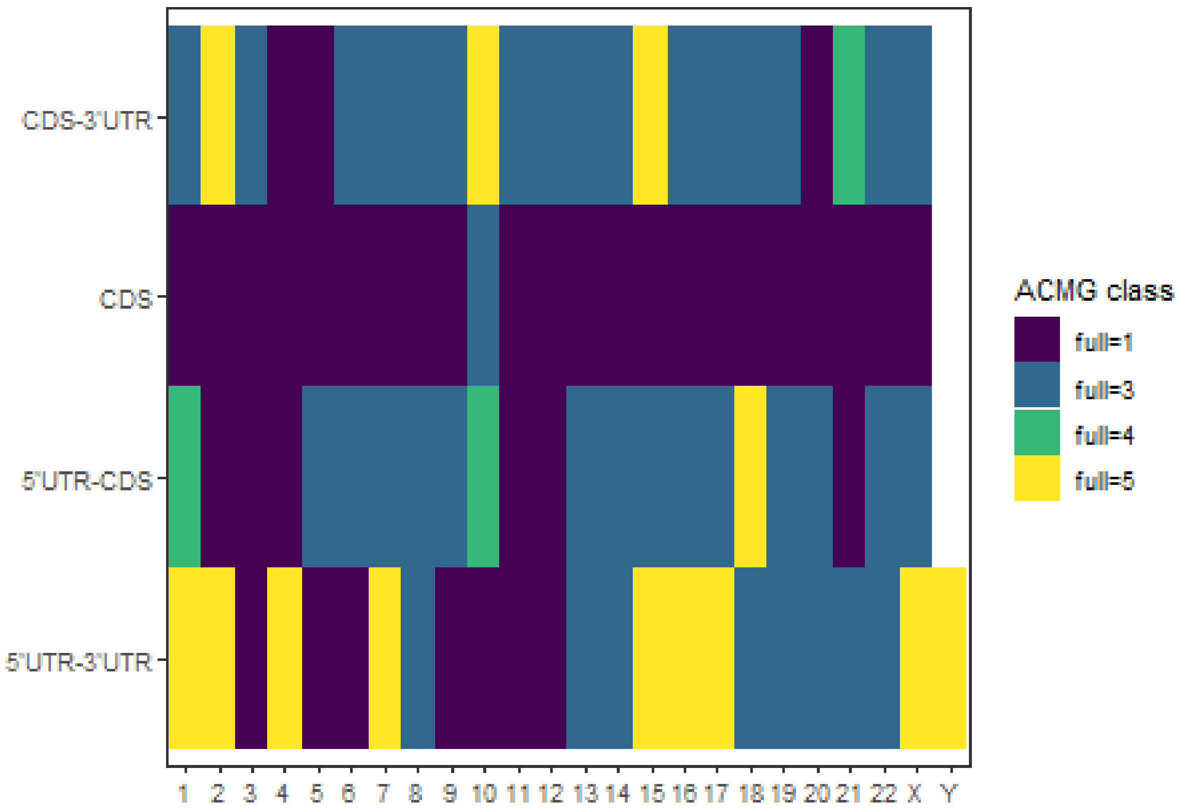


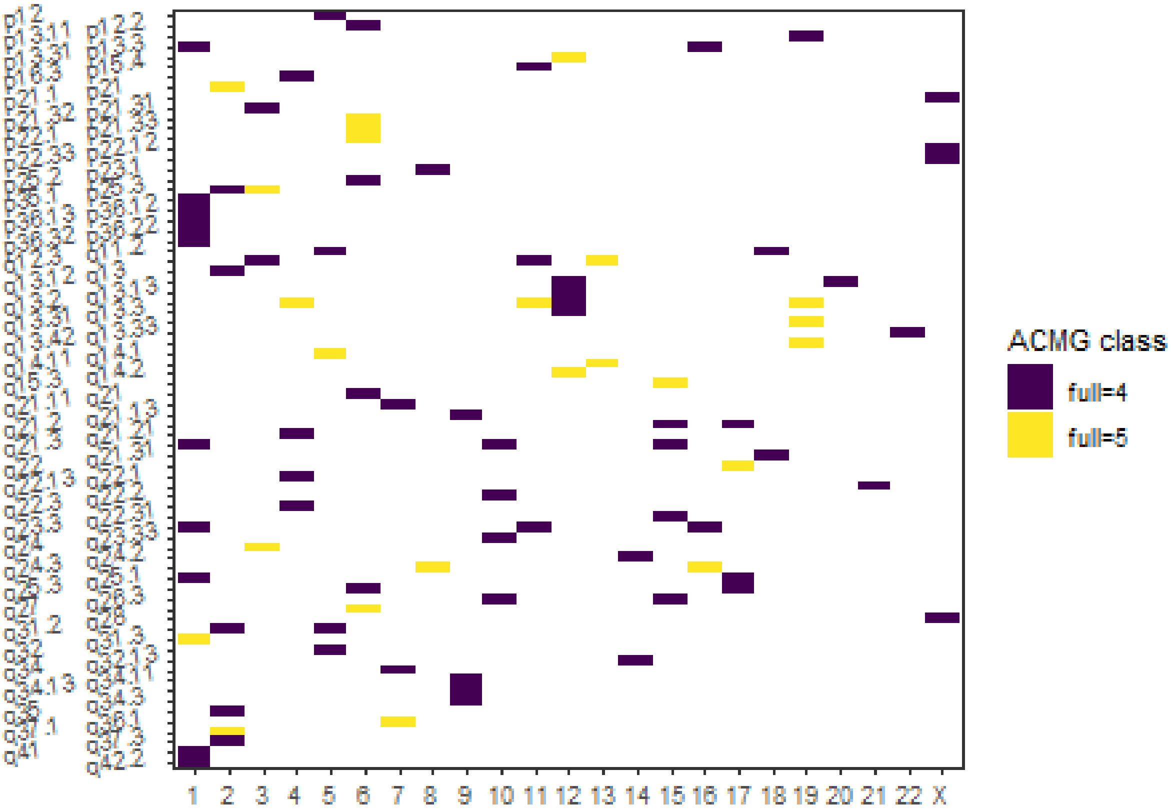
```
# exon count and chromosome profile
CDS_variants_exres %>%
  ggplot() + aes(x=SV_chrom, y=Location2, fill=sqrt(sqrt(sqrt(Exon_count)))) + geom_tile() +
  scale_fill_paletteer_c("ggthemes::Sunset-Sunrise Diverging", name = "Exon count",
                         labels = paste(c(1, 5, 25, 100, 250))) +
  xlab("") + ylab("") +
  theme_bw() + theme(panel.grid.major = element_blank(), panel.background = element_rect(fill = "grey25

#ggsave2("exon_count_chr_profile_exres.png")
```



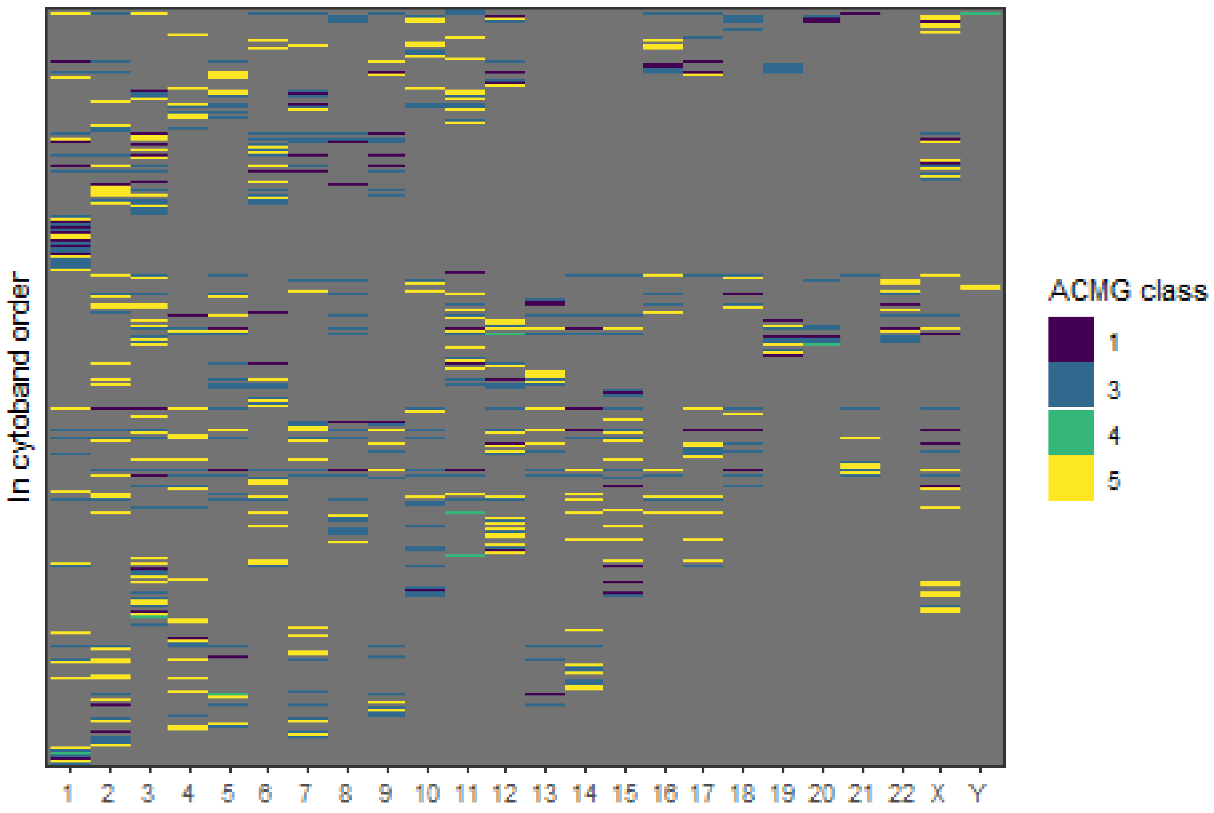
```
# ACMG and chromosome profile: Curated set <pathogenic & CDS-affecting & has length < 200K>
CDS_variants_exres %>%
  ggplot() + aes(x=SV_chrom, y=Location2, fill=ACMG_class) + geom_tile() +
  scale_fill_viridis_d(name = "ACMG class") +
  xlab("") + ylab("") +
  theme_bw() + theme(panel.grid.major = element_blank())

#ggsave2("exon_count_chr_profile_exres.png")

# With all CDS-affecting variants...
CDS %>% filter(ACMG_class != "full=NA") %>%
  ggplot() + aes(x=SV_chrom, y=Location2, fill=ACMG_class) + geom_tile() +
  scale_fill_viridis_d(name = "ACMG class") +
  xlab("") + ylab("") +
  theme_bw() + theme(panel.grid.major = element_blank())

# Include cytobands
CDS_variants_exres %>%
  ggplot() + aes(x=SV_chrom, y=reorder(CytoBand, desc(CytoBand)), fill=ACMG_class) +
  geom_tile(aes(alpha = Gene_count)) +
  scale_fill_viridis_d(name = "ACMG class") +
  xlab("") + ylab("") +
  theme_bw() + theme(panel.grid.major = element_blank(),
                     axis.text.y=element_text(angle = 10, hjust = 0)) +
  scale_y_discrete(guide = guide_axis(n.dodge=2))
```


CDS-3'UTR
CDS
5'UTR-CDS
5'UTR-3'UTR
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
X
ACMG class
full=4
full=5


CDS-3'UTR
CDS
5'UTR-CDS
5'UTR-3'UTR
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
X
Y
ACMG class
full=1
full=3
full=4
full=5

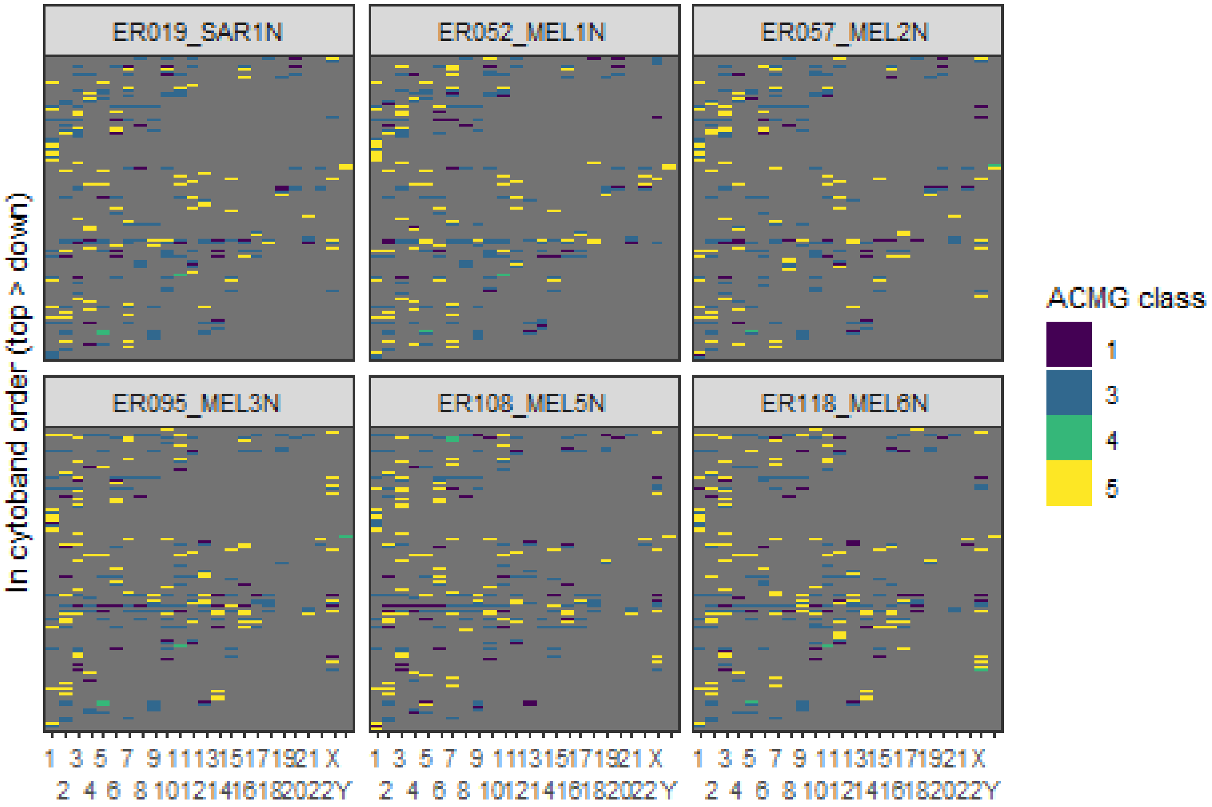


```
# With all CDS-affecting variants...
CDS %>% filter(ACMG_class != "full=NA") %>%
  ggplot() + aes(x=SV_chrom, y=reorder(CytoBand, desc(CytoBand)), fill=ACMG_class) +

  ## change to dplyr:: "Count the number of ACMG_class of this type in this cytoband
  geom_tile(aes(alpha = Gene_count)) +

  scale_fill_viridis_d(name = "ACMG class", labels = paste(c("1", "3", "4", "5"))) +
  xlab("") + ylab("In cytoband order") +
  theme_bw() + theme(panel.grid.major = element_blank(), axis.text.y=element_blank(), axis.ticks.y = ele
                     panel.background = element_rect(fill = "grey45"))

#ggsave2("ACMG_profile_by_location_exres_dark.png")
```

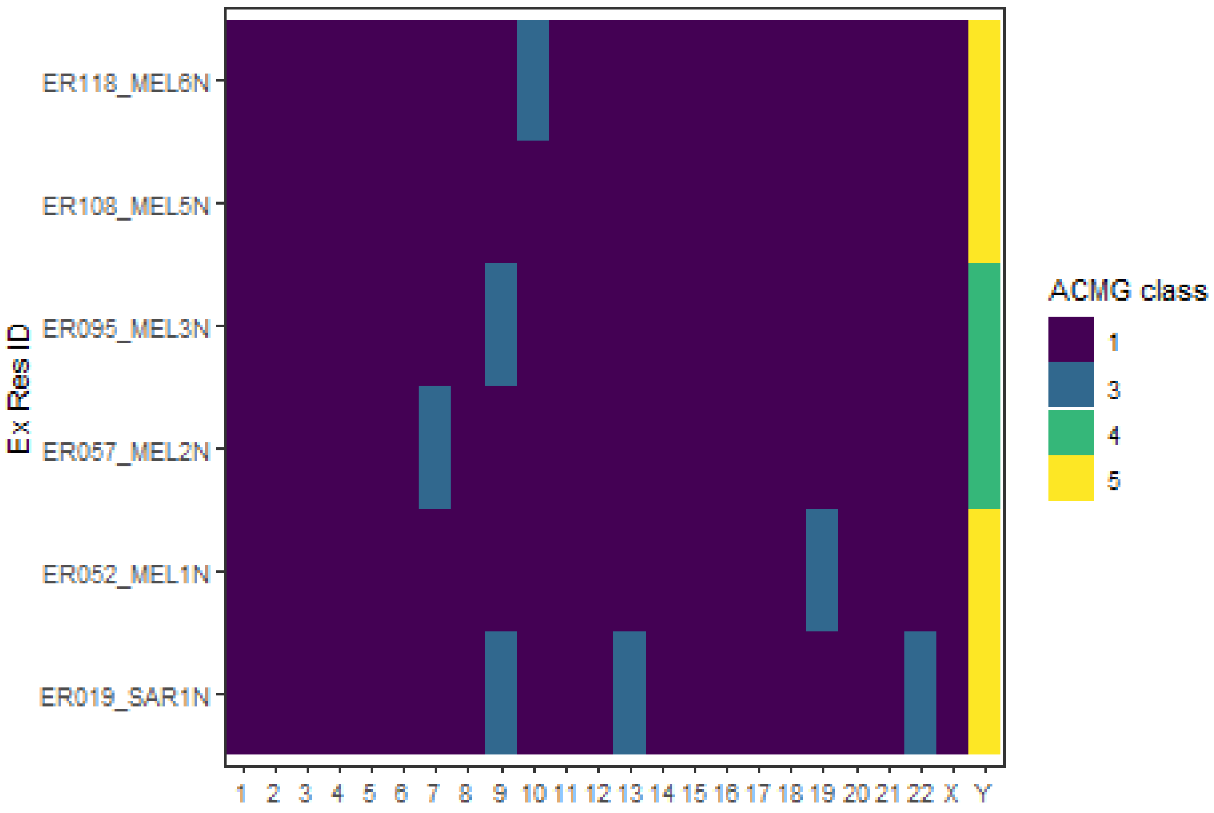


How to correctly interpret this plot?

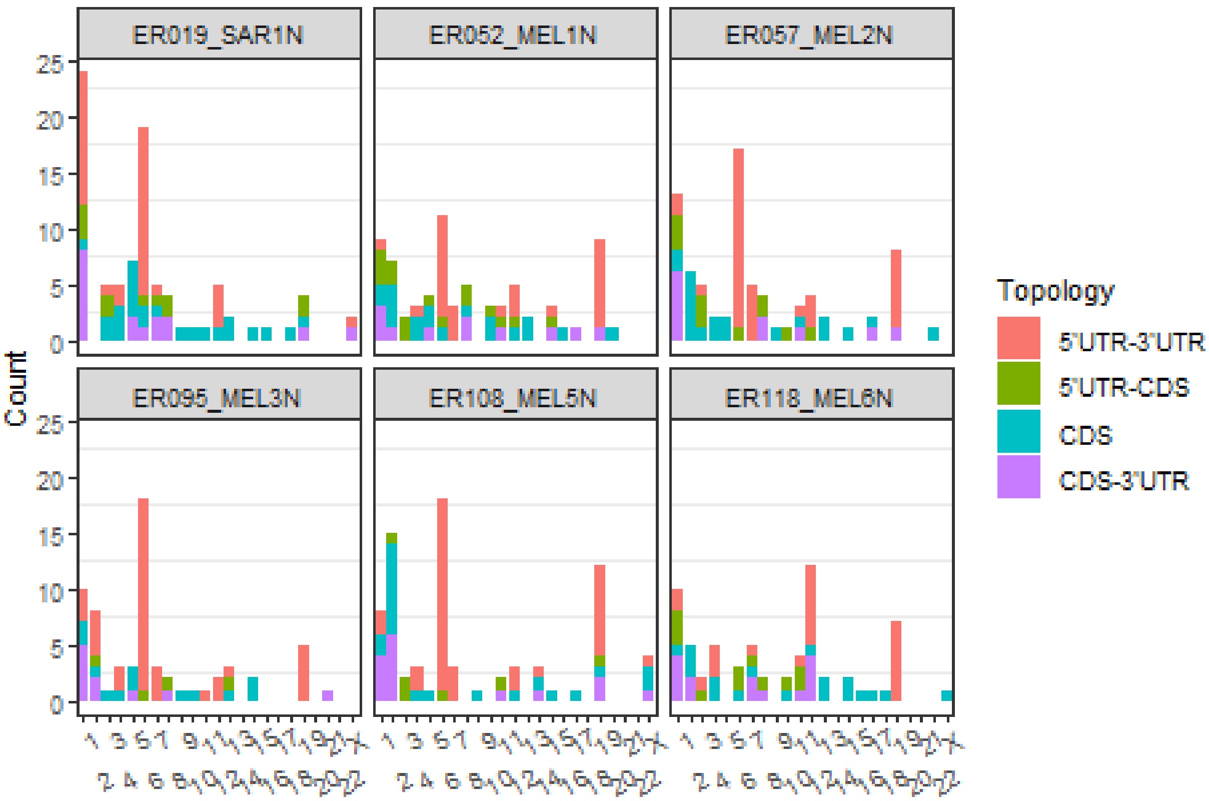
```
# Juxtapose six ExRes sample genomes
CDS %>% filter(ACMG_class != "full=NA") %>%
  ggplot() + aes(x=SV_chrom, y=reorder(CytoBand, desc(CytoBand)), fill=ACMG_class) +
  geom_tile(aes(alpha = Gene_count)) +
  scale_fill_viridis_d(name = "ACMG class", labels = paste(c("1", "3", "4", "5"))) +
  xlab("") + ylab("In cytoband order (top > down)") +
  scale_x_discrete(guide = guide_axis(n.dodge=2)) +
  theme_bw() + theme(panel.grid.major = element_blank(), panel.background = element_rect(fill = "grey45
                     axis.text.y=element_blank(), axis.ticks.y = element_blank()) +
  facet_wrap(~ExResID)

#ggsave2("ACMG_profile_by_location_exres.png")
```



```r
# ACMG class across genome among patients
CDS %>% filter(ACMG_class != "full=NA") %>%
  ggplot() + aes(x=SV_chrom, y=ExResID, fill=ACMG_class) +
  geom_tile(aes(alpha = Gene_count)) +
  scale_fill_viridis_d(name = "ACMG class", labels = paste(c("1", "3", "4", "5"))) +
  xlab("") + ylab("Ex Res ID") +
  theme_bw() + theme(panel.grid.major = element_blank())
```



```
# Distribution of detected variant topology across each patient's genome
CDS_variants_exres %>%
  ggplot() + aes(x=SV_chrom, fill=Location2) +
  geom_bar() + facet_wrap(~ExResID) +
  scale_fill_discrete(name = "Topology") + ylab("Count") + xlab("") +
  theme_bw() + theme(panel.grid.major = element_blank(),
                     axis.text.x=element_text(angle = 30)) +
  scale_x_discrete(guide = guide_axis(n.dodge=2))

#ggsave2("variant_topology_by_exres_ID.png")
```



II. Visualising high-dimensional data

Sandbox

Create a smaller subset of the master file

```
# AM=full across whole SVs
profile.full <- subset(benchmark.full,

       select=c("SV_chrom", "SV_start", "SV_end",   "SV_length", "SV_type", ## basic SV info
                "ID",   "REF",  "ALT",  "QUAL", "FILTER",   "INFO", "Coverage", "Caller", ## basic SV p
                "AnnotSV_ranking_score",    "AnnotSV_ranking_criteria"
                ))


# AM=split across single genes
profile.split <- subset(benchmark.split,

        select=c("SV_chrom",    "SV_start", "SV_end",   "SV_length",    "SV_type", ## basic SV info
                "ID",   "REF",  "ALT",  "QUAL", "FILTER",   "INFO", "Coverage", "Caller", ## basic SV p
                "CytoBand", "Gene_name",    "Gene_count", "Location2", ## gene profile
                "n_Exon", "Frameshift", "Overlapped_CDS_length", "Overlapped_CDS_percent", ## topologic
                "ACMG_class",   "GenCC_disease", "GenCC_moi", "GenCC_classification", ## ACMG + GenCC c
                "ExAC_delZ",    "ExAC_dupZ",    "ExAC_cnvZ",    "ExAC_synZ",    "ExAC_misZ", ## ExAC Z
                "GnomAD_pLI",   "ExAC_pLI", ## GnomAD/ExAC pLIs
                ))
```

Footnotes

Session Info

```
sessionInfo()
```

```
## R version 4.1.2 (2021-11-01)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19043)
## 
## Matrix products: default
## 
## locale:
## [1] LC_COLLATE=English_Australia.1252  LC_CTYPE=English_Australia.1252   
## [3] LC_MONETARY=English_Australia.1252 LC_NUMERIC=C                      
## [5] LC_TIME=English_Australia.1252    
## system code page: 950
## 
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base     
## 
## other attached packages:
##  [1] viridis_0.6.2     viridisLite_0.4.0 cowplot_1.1.1     ggforce_0.3.3    
##  [5] paletteer_1.4.0   forcats_0.5.1     stringr_1.4.0     dplyr_1.0.7      
##  [9] purrr_0.3.4       readr_2.1.1       tidyr_1.1.4       tibble_3.1.6     
## [13] ggplot2_3.3.5     tidyverse_1.3.1  
## 
## loaded via a namespace (and not attached):
##  [1] httr_1.4.2       bit64_4.0.5      vroom_1.5.7      jsonlite_1.7.2  
##  [5] modelr_0.1.8     assertthat_0.2.1 cellranger_1.1.0 yaml_2.2.1      
##  [9] pillar_1.6.4     backports_1.4.1  glue_1.6.0       digest_0.6.29   
## [13] polyclip_1.10-0  rvest_1.0.2      colorspace_2.0-2 htmltools_0.5.2 
## [17] pkgconfig_2.0.3  broom_0.7.11     haven_2.4.3      scales_1.1.1    
## [21] tweenr_1.0.2     tzdb_0.2.0       generics_0.1.1   farver_2.1.0    
## [25] ellipsis_0.3.2   withr_2.4.3      cli_3.1.0        magrittr_2.0.1  
## [29] crayon_1.4.2     readxl_1.3.1     evaluate_0.14    fs_1.5.2        
## [33] fansi_0.5.0      MASS_7.3-54      xml2_1.3.3       ggthemes_4.2.4  
## [37] tools_4.1.2      hms_1.1.1        lifecycle_1.0.1  munsell_0.5.0   
## [41] reprex_2.0.1     compiler_4.1.2   rlang_0.4.12     grid_4.1.2      
## [45] rstudioapi_0.13  labeling_0.4.2   rmarkdown_2.11   gtable_0.3.0    
## [49] DBI_1.1.2        rematch2_2.1.2   R6_2.5.1         gridExtra_2.3   
## [53] lubridate_1.8.0  knitr_1.37       fastmap_1.1.0    bit_4.0.4       
## [57] utf8_1.2.2       prismatic_1.1.0  stringi_1.7.6    parallel_4.1.2  
## [61] Rcpp_1.0.8       vctrs_0.3.8      dbplyr_2.1.1     tidyselect_1.1.1
## [65] xfun_0.29
```