HLA

```
library(data.table)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following objects are masked from 'package:data.table':
##
##     dcast, melt
```

```
library(scales)
library(parallel)
library(stringr)
```

Metadata

Load HIP data statistics

```
dt.hip.stats = fread("annotations/hip_stats.txt") %>%
  mutate(count_total = count, occurrences_total = diversity) %>%
  select(sample_id, race, sex, cmv, hla, count_total, occurrences_total)

dt.hip.stats$cmv = with(dt.hip.stats, ifelse(is.na(cmv), "Unknown", cmv))
```

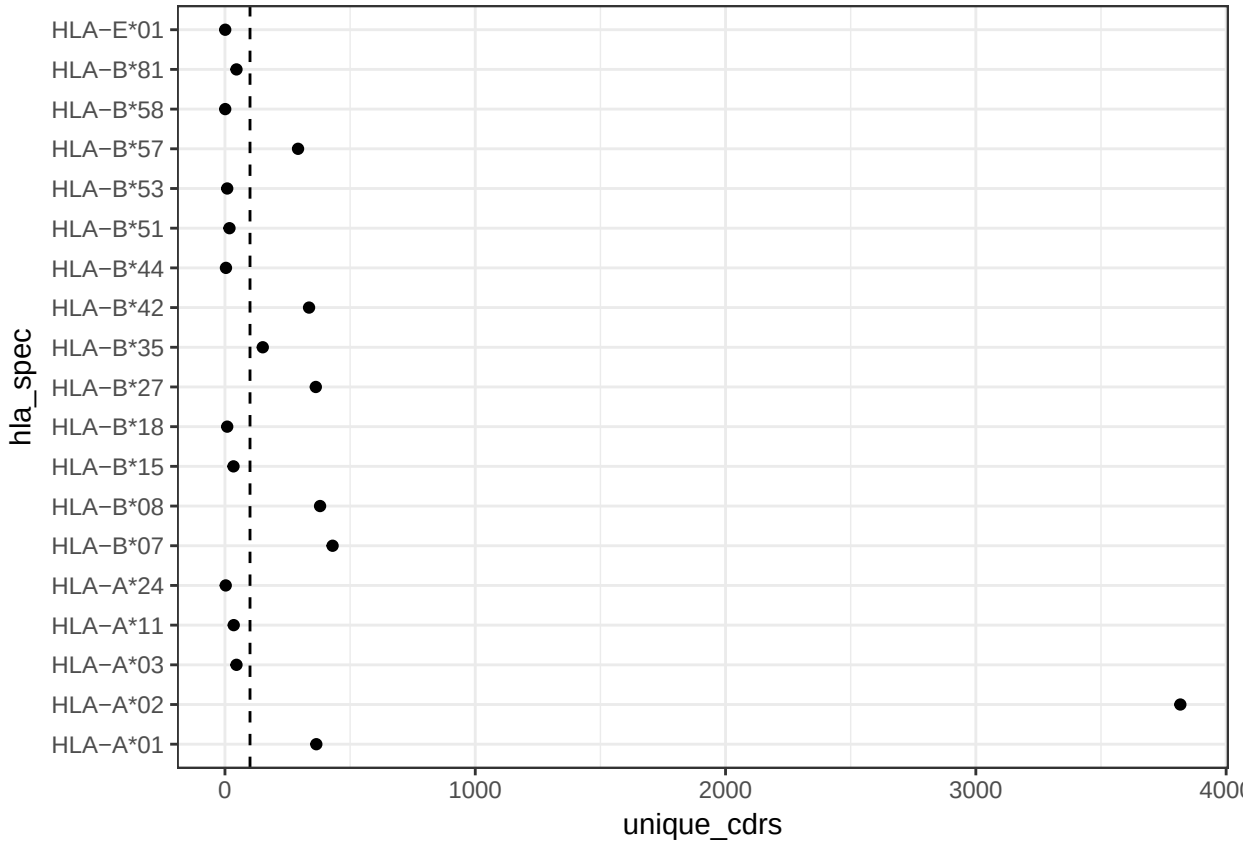
Flattening HLA lists

```
dt.hip.hla.flat = rbindlist(lapply(mapply(list, dt.hip.stats$sample_id, dt.hip.stats$hla, SIMPLIFY=F),
                                   function(x) data.table(sample_id = x[[1]], hla = unlist(strsplit(x[[1
  filter(!is.na(hla)) %>%
  merge(dt.hip.stats %>% select(sample_id, count_total, occurrences_total))
```

Pre-filtering

HLA specificities from VDJdb

```
MIN_HLA_CLONOTYPES = 100

dt.vdjdb.hla = fread("rearr_model/VDJDB_fullP_rob_ageing.txt") %>%
  filter(mhc.class == "MHCI") %>%
  mutate(hla_spec = str_split_fixed(mhc.a, pattern = "[:,]", 2)[,1]) %>%
  select(cdr3, hla_spec) %>%
  group_by(hla_spec) %>%
  mutate(unique_cdrs = n())

ggplot(dt.vdjdb.hla %>% select(hla_spec, unique_cdrs) %>% unique,
       aes(x = hla_spec, y = unique_cdrs)) +
  geom_point() +
  geom_hline(yintercept = MIN_HLA_CLONOTYPES, linetype = "dashed") +
  coord_flip() +
  theme_bw()
```

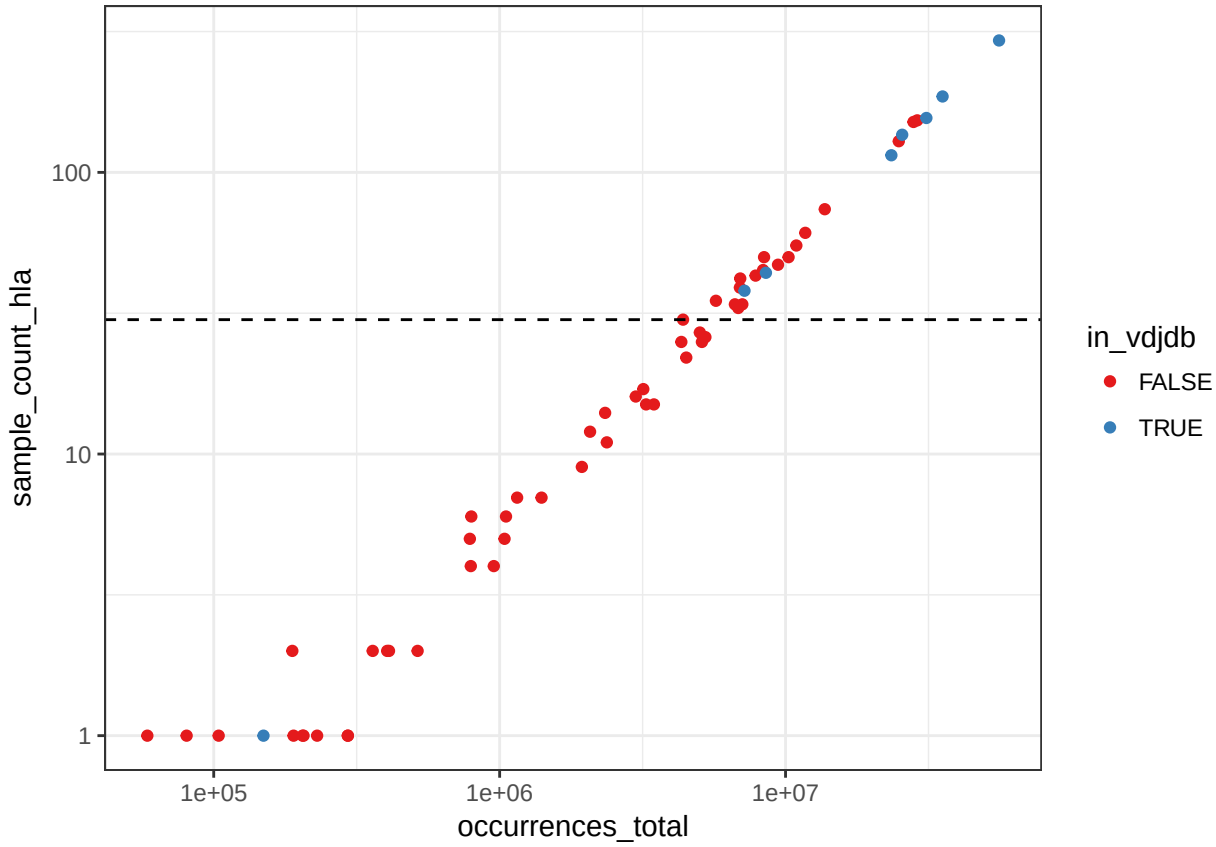



```
good_hla_spec = (dt.vdjdb.hla %>% filter(unique_cdrs > MIN_HLA_CLONOTYPES))$hla_spec %>% unique
```

HLA summary from HIP data

```
MIN_HLA_SAMPLES = 30

dt.hip.hla.flat.summary = dt.hip.hla.flat %>% group_by(hla) %>%
  mutate(sample_count_hla = length(unique(sample_id))) %>%
  group_by(hla, sample_count_hla) %>%
  summarise(occurrences_total = sum(occurrences_total))

dt.hip.hla.flat.summary$in_vdjdb = dt.hip.hla.flat.summary$hla %in% good_hla_spec
```

```
ggplot(dt.hip.hla.flat.summary,
       aes(occurrences_total, sample_count_hla, color = in_vdjdb)) +
  geom_point() +
  geom_hline(yintercept = MIN_HLA_SAMPLES, linetype = "dashed") +
  scale_x_log10() +
  scale_y_log10() +
  scale_color_brewer(palette = "Set1") +
  theme_bw()
```




```
good_hla = (dt.hip.hla.flat.summary %>% filter(sample_count_hla >= MIN_HLA_SAMPLES))$hla
good_hla_spec = intersect(good_hla_spec, good_hla) # HLA spec should be present in HIP HLA for comparis
```

HIP annotation data

Load VDJdb annotations with 1 mismatch for HIP data (time consuming, ~ 2mln clonotypes)

```
dt.hip = rbindlist(mclapply(as.list(dt.hip.stats$sample_id),
                    function(x) fread(paste0("annotations/split_1mm/", x, ".annot.txt")) %>%
                      mutate(sample_id = x), mc.cores = 40)) %>%
  group_by(sample_id, cdr3) %>%
  summarise(count = sum(count), occurrences = n())
```

Merge annotations with metadata + select good HLAs

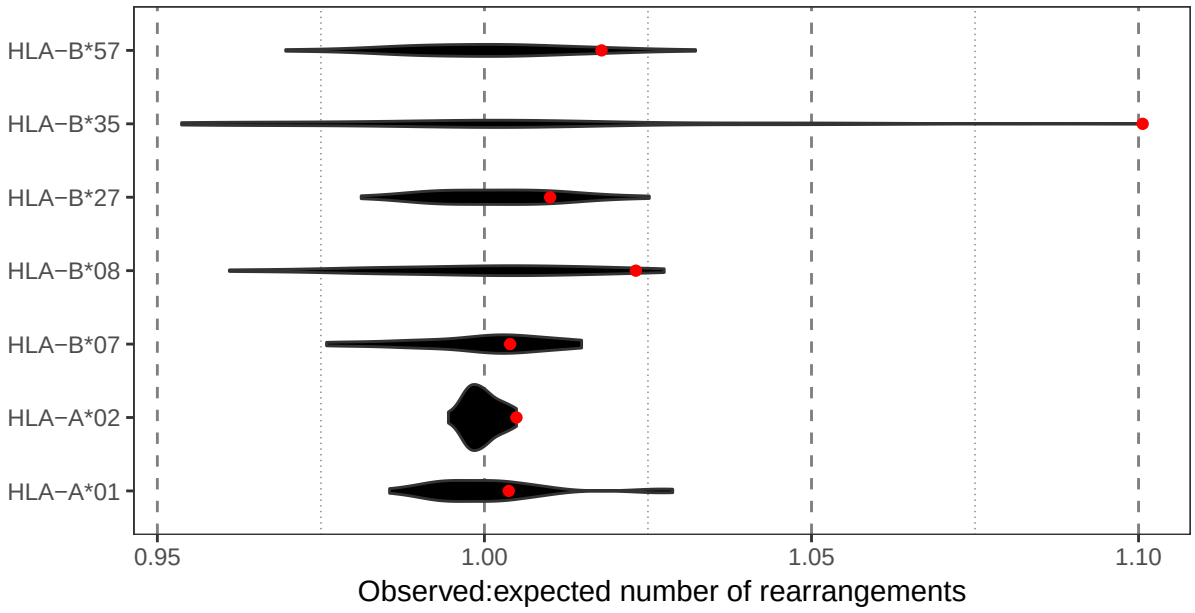
```
dt.hip.m = dt.hip %>%
  merge(dt.hip.hla.flat %>% filter(hla %in% good_hla)) %>%
  merge(dt.vdjdb.hla %>% filter(hla_spec %in% good_hla_spec))
```

Summarise and compute observed:expected ratio

```
dt.hip.s = dt.hip.m %>%
  group_by(hla, hla_spec) %>%
  summarise(occurrences = sum(occurrences)) %>%
  group_by(hla) %>%
  mutate(occurrences_total_h = sum(occurrences)) %>%
  group_by(hla_spec) %>%
  mutate(occurrences_total_s = sum(occurrences)) %>%
  ungroup() %>%
  mutate(occurrences_total = sum(occurrences)) %>%
  mutate(obsexpratio = as.numeric(occurrences_total)*occurrences/occurrences_total_s/occurrences_total_
```

Plot observed:expected number of rearrangements for matched and mismatched HLA specificity + donor HLA

```
p12=ggplot(dt.hip.s, aes(x=hla_spec, y = obsexpratio)) +
  geom_violin(aes(group = hla_spec), fill = "black") +
  geom_point(data=dt.hip.s %>% filter(hla == hla_spec), color = "red") +
  ylab("Observed:expected number of rearrangements") + xlab("") +
  coord_flip() +
  theme_bw() +
  theme(aspect = 0.5, panel.grid.major.x = element_line(colour="grey50", linetype="dashed"),
        panel.grid.minor.x = element_line(colour="grey50", linetype="dotted"),
        panel.grid.major.y = element_blank(), panel.grid.minor.y = element_blank())

p12
```



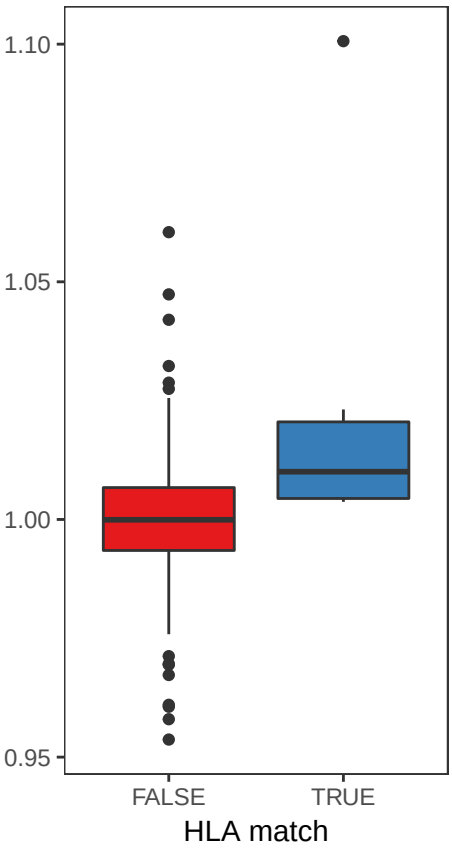
Stat test and plot test results

```
dt.hip.s$hla_match = with(dt.hip.s, hla_spec == hla)
res = wilcox.test(obsexpratio~hla_match, dt.hip.s)
print(res)
```

```
##
```

```
##  Wilcoxon rank sum test with continuity correction
##
## data:  obsexpratio by hla_match
## W = 205, p-value = 0.003586
## alternative hypothesis: true location shift is not equal to 0
```

```
p13=ggplot(dt.hip.s, aes(x=hla_spec == hla, y = obsexpratio)) +
  geom_boxplot(aes(fill = hla_spec == hla)) +
  #annotate("text", x = 1.5, y = 1.125, label = paste("italic(P) ==", round(res$p.value,3)), parse = T)
  scale_fill_brewer(guide = F, palette = "Set1") +
  ylab("") + xlab("HLA match") +
  theme_bw() +
  theme(aspect = 2,
        panel.grid.major = element_blank(), panel.grid.minor = element_blank())
p13
```



CMV clonal expansions

Select CMV-specific clonotypes

```
dt.vdjdb.hla.cmv = fread("rearr_model/VDJDB_fullP_rob_ageing.txt") %>%
  filter(mhc.class == "MHCI", antigen.species %in% c("CMV", "EBV"), gene == "TRB") %>%
  mutate(hla_spec = str_split_fixed(mhc.a, pattern = "[:,]", 2)[,1]) %>%
  select(cdr3, hla_spec, antigen.species)
```

Merge VDJdb clonotypes with HIP annotations

```
dt.hip.p = dt.hip %>%
  merge(dt.vdjdb.hla.cmv, by = "cdr3") %>%
  merge(dt.hip.hla.flat %>% filter(hla %in% good_hla), by = "sample_id") %>%
  merge(dt.hip.stats %>% select(sample_id, cmv))
```

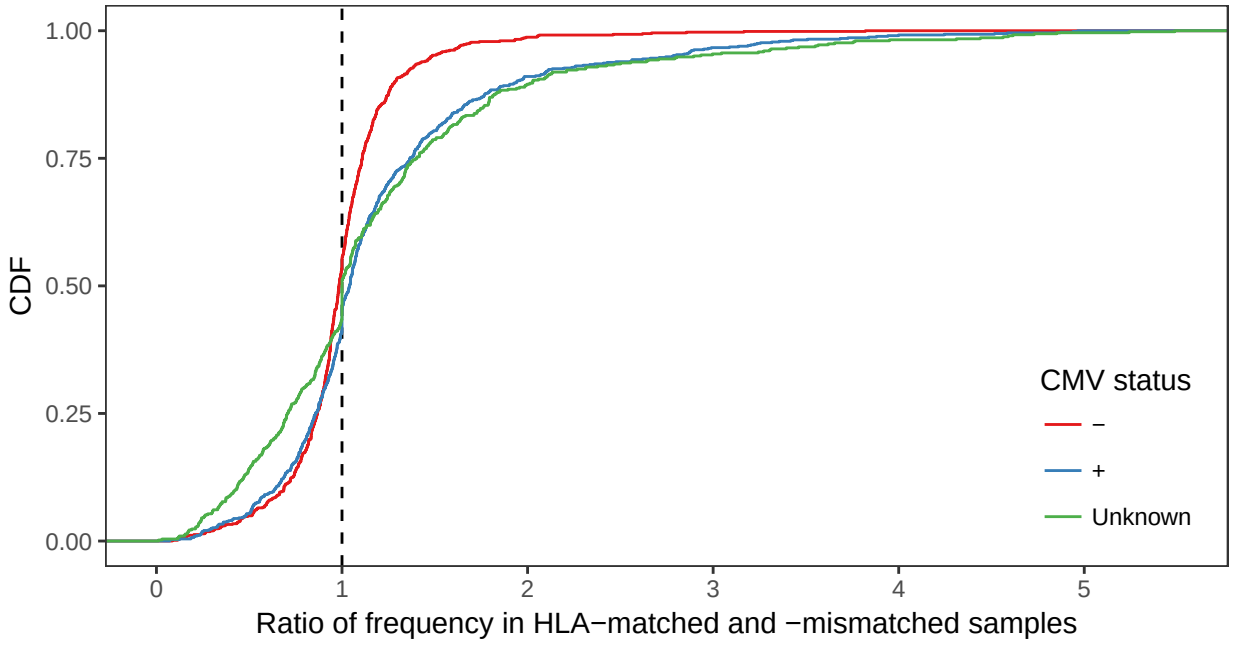
Compute observed and expected occurrences

```
dt.hip.p.s = dt.hip.p %>%
  mutate(hla_match = hla == hla_spec) %>%
  group_by(cdr3, cmv, hla_spec, hla_match, antigen.species) %>%
  summarise(count = sum(count),
            count_total = sum(as.numeric(count_total)))

dt.hip.p.s = dt.hip.p.s %>%
  merge(dt.hip.p.s %>%
              ungroup %>%
  group_by(cdr3, cmv, antigen.species, hla_spec) %>%
  summarise(total = n()) %>%
  filter(total == 2) %>%
  select(cdr3, cmv, antigen.species, hla_spec))

dt.hip.p.s = dt.hip.p.s %>%
  group_by(cdr3, cmv, antigen.species, hla_spec) %>%
  summarise(freq_ratio = count[which(hla_match)] / count_total[which(hla_match)] /
           (count[which(!hla_match)] / count_total[which(!hla_match)]))
```

Plotting CMV-specific clonotype expansions

```
p14=ggplot(dt.hip.p.s %>% filter(antigen.species == "CMV"),
       aes(x=freq_ratio, color = cmv)) +
  geom_vline(xintercept = 1, linetype = "dashed") +
  stat_ecdf() +
  ylab("CDF") +
  scale_x_continuous("Ratio of frequency in HLA-matched and -mismatched samples", breaks = c(0,1,2,3,4,5
  scale_color_brewer("CMV status", palette = "Set1") +
  theme_bw() +
  theme(aspect = 0.5, legend.direction = "vertical", legend.position = c(0.9, 0.2),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank())

p14
```

```
## Warning: Removed 6 rows containing non-finite values (stat_ecdf).
```




```
ks.test((dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "+"))$freq_ratio,
        (dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "-"))$freq_ratio)
```

```
## Warning in ks.test((dt.hip.p.s %>% filter(antigen.species == "CMV", cmv
## == : p-value will be approximate in the presence of ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  (dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "+"))$freq_ratio and (dt.hip.p.s %>% f
## D = 0.18401, p-value = 5.84e-11
## alternative hypothesis: two-sided
```

```
ks.test((dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "+"))$freq_ratio,
        (dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "Unknown"))$freq_ratio)
```

```
## Warning in ks.test((dt.hip.p.s %>% filter(antigen.species == "CMV", cmv
## == : p-value will be approximate in the presence of ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  (dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "+"))$freq_ratio and (dt.hip.p.s %>% f
## D = 0.11988, p-value = 0.000359
## alternative hypothesis: two-sided
```

```
ks.test((dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "-"))$freq_ratio,
        (dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "Unknown"))$freq_ratio)
```

```
## Warning in ks.test((dt.hip.p.s %>% filter(antigen.species == "CMV", cmv
## == : p-value will be approximate in the presence of ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  (dt.hip.p.s %>% filter(antigen.species == "CMV", cmv == "-"))$freq_ratio and (dt.hip.p.s %>% f
```

```
## D = 0.21928, p-value = 7.789e-13
## alternative hypothesis: two-sided
```

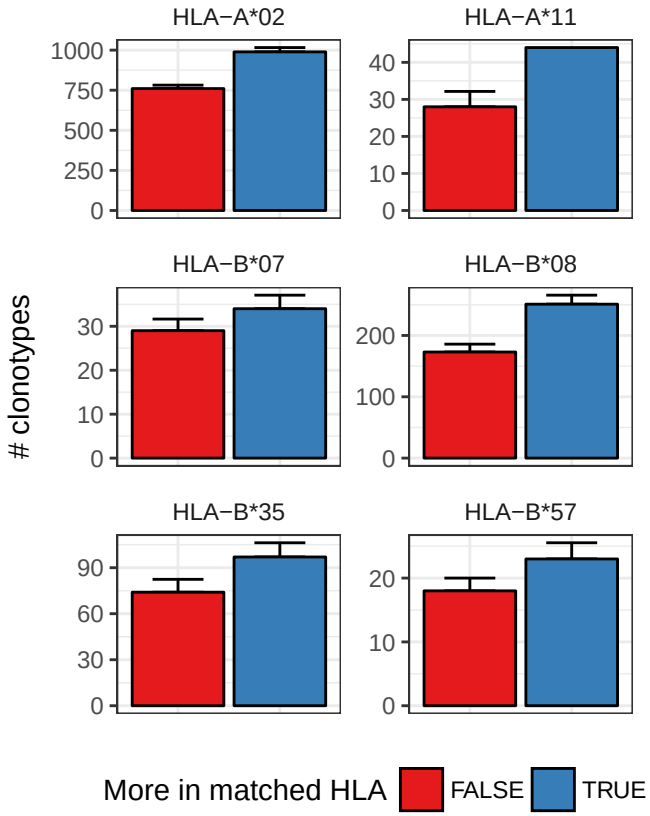
EBV-specific expansions by HLA (note EBV is extremely common)

```
dt.hip.p.ebv = dt.hip.p.s %>% filter(antigen.species == "EBV",
                                     hla_spec != "HLA-B*44") %>%
  group_by(enriched = freq_ratio>1, hla_spec) %>%
  summarise(count = n()) %>%
  merge(dt.vdjdb.hla %>% select(hla_spec, unique_cdrs) %>% unique) %>%
  mutate(p = count / unique_cdrs, sd = sqrt(count * p * (1-p)))
```

```
## Warning in sqrt(count * p * (1 - p)): NaNs produced
```

```
p15=ggplot(dt.hip.p.ebv, # Only 3 clonotypes here
       aes(x=enriched, y = count, fill = enriched)) +
  geom_errorbar(aes(ymin = count, ymax = count + 1.96 * sd), width = 0.5) +
  geom_bar(stat="identity", color = "black") +
  facet_wrap(~hla_spec, scales = "free", ncol = 2) +
  ylab("# clonotypes") +
  xlab("") +
  scale_fill_brewer("More in matched HLA", palette = "Set1") +
  theme_bw() +
  theme(aspect = 0.8, legend.position = "bottom",
        axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        strip.background = element_blank())
p15
```

```
## Warning: Removed 1 rows containing missing values (geom_errorbar).
```



Figures

```
ggsave("figures/p12.pdf", p12, width = 2*4, height = 4)
ggsave("figures/p13.pdf", p13, width = 4, height = 4)
ggsave("figures/p14.pdf", p14, width = 2*4, height = 4)
```

```
## Warning: Removed 6 rows containing non-finite values (stat_ecdf).
```

```
ggsave("figures/p15.pdf", p15, width = 4, height = 8)
```

```
## Warning: Removed 1 rows containing missing values (geom_errorbar).
```