Exploratory analysis of TCR:antigen contacts observed in structural data

```
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(reshape2)
```

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

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

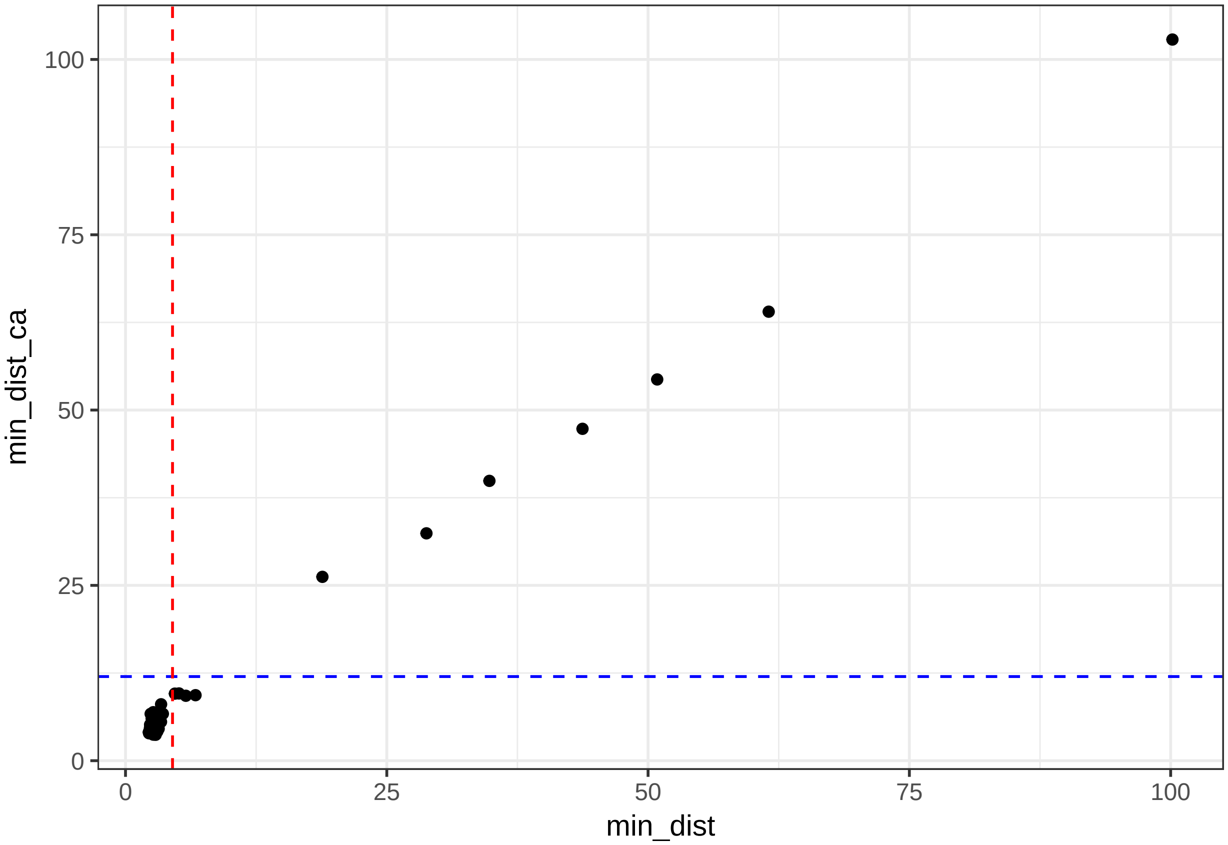
```
library(ggplot2)
library(RColorBrewer)
```

Load data, filter complexes that are too far away

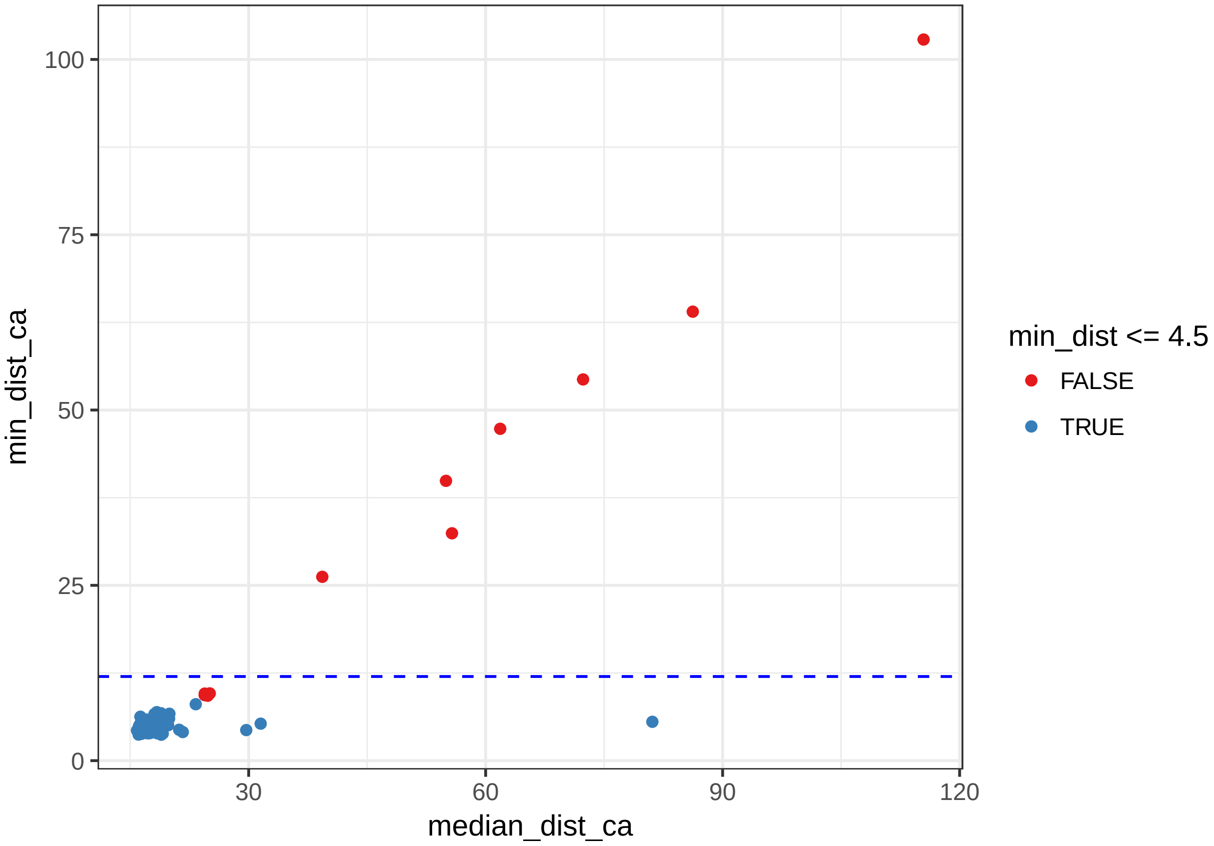
```
dt.struct = fread("preprocessing/output/structure.txt") %>%
  filter(pdb_id != "4p46") # spurious long antigen

dt.struct.mindist = dt.struct %>%
  group_by(pdb_id) %>%
  summarise(min_dist = min(distance),
            min_dist_ca = min(distance_CA),
            median_dist_ca = median(distance_CA))

ggplot(dt.struct.mindist,
       aes(x = min_dist, y = min_dist_ca)) +
  geom_point() +
  geom_hline(yintercept = 12, linetype = "dashed", color = "blue") +
  geom_vline(xintercept = 4.5, linetype = "dashed", color = "red") +
  theme_bw()
```



```
ggplot(dt.struct.mindist,
       aes(x = median_dist_ca, y = min_dist_ca, color = min_dist <= 4.5)) +
  geom_point() +
  geom_hline(yintercept = 12, linetype = "dashed", color = "blue") +
  scale_color_brewer(palette = "Set1") +
  theme_bw()
```



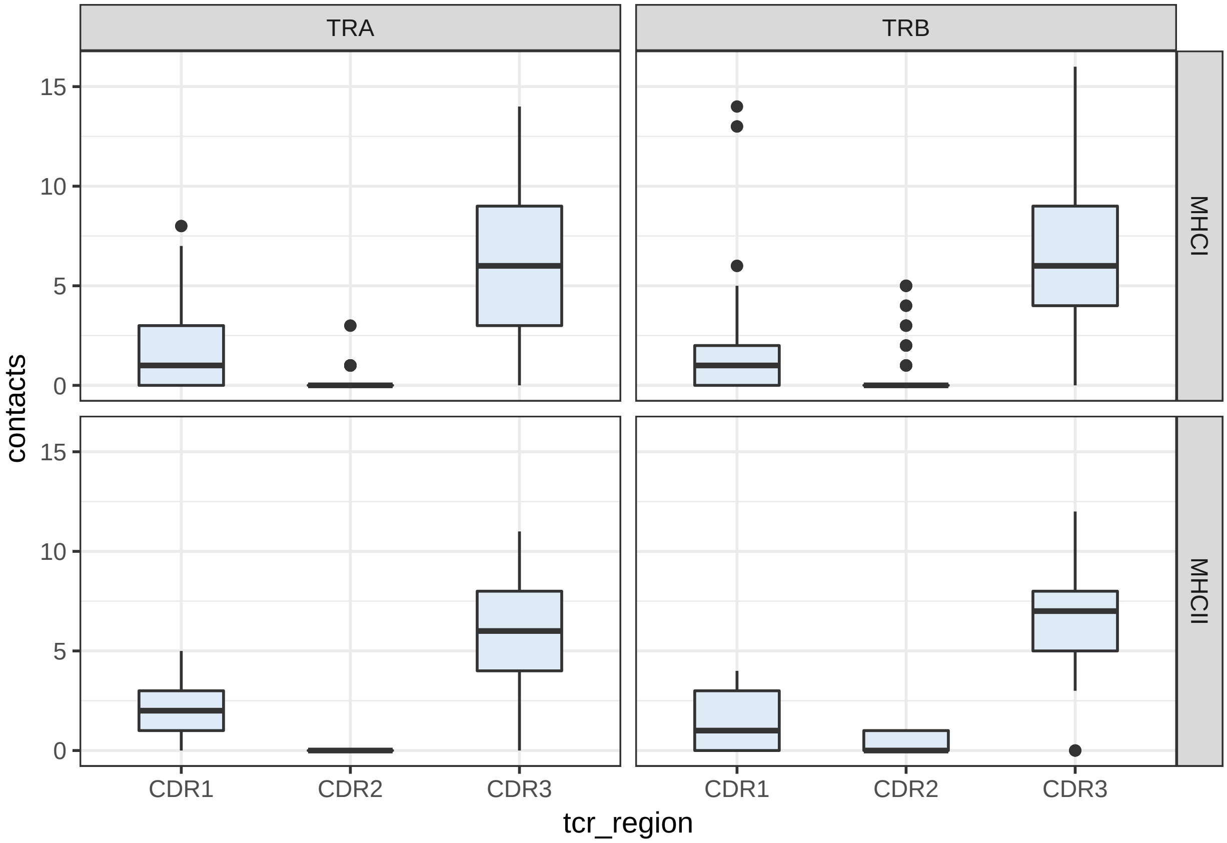
```
pdb_id_good = dt.struct.mindist %>%
  filter(min_dist_ca <= 12) %>%
  .$pdb_id

dt.struct = dt.struct %>%
  filter(pdb_id %in% pdb_id_good)
```

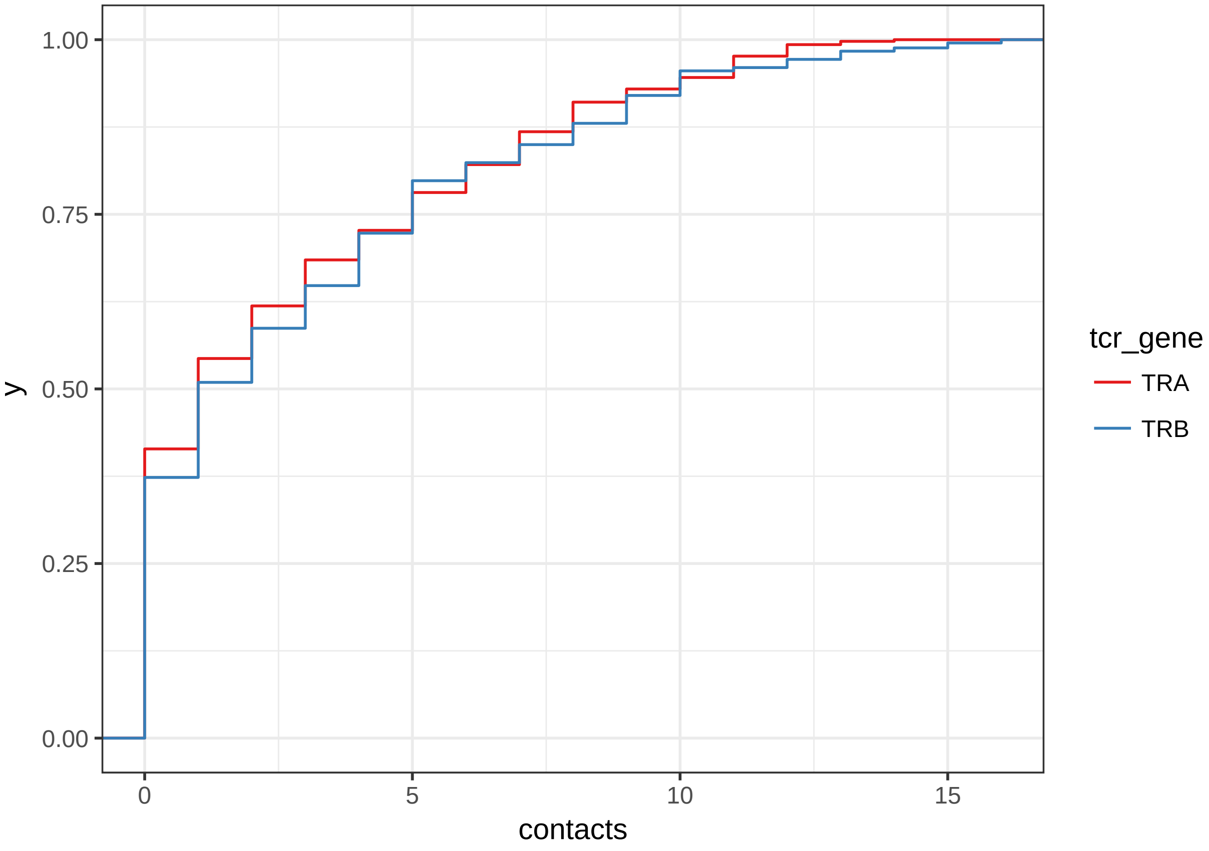
Number of contacts by region/gene/complex type

```
dt.contact.region.s = dt.struct %>%
  group_by(pdb_id, species, mhc_type, tcr_gene, tcr_region) %>%
  summarise(contacts = sum(distance <= 4.5))

ggplot(dt.contact.region.s, aes(x = tcr_region, group = tcr_region, y = contacts)) +
  geom_boxplot(fill = "#deebf7", width = 0.5) +
  facet_grid(mhc_type ~ tcr_gene) +
  theme_bw()
```



```
ggplot(dt.contact.region.s, aes(x = contacts, color = tcr_gene)) +
  stat_ecdf() +
  scale_color_brewer(palette = "Set1") +
  theme_bw()
```

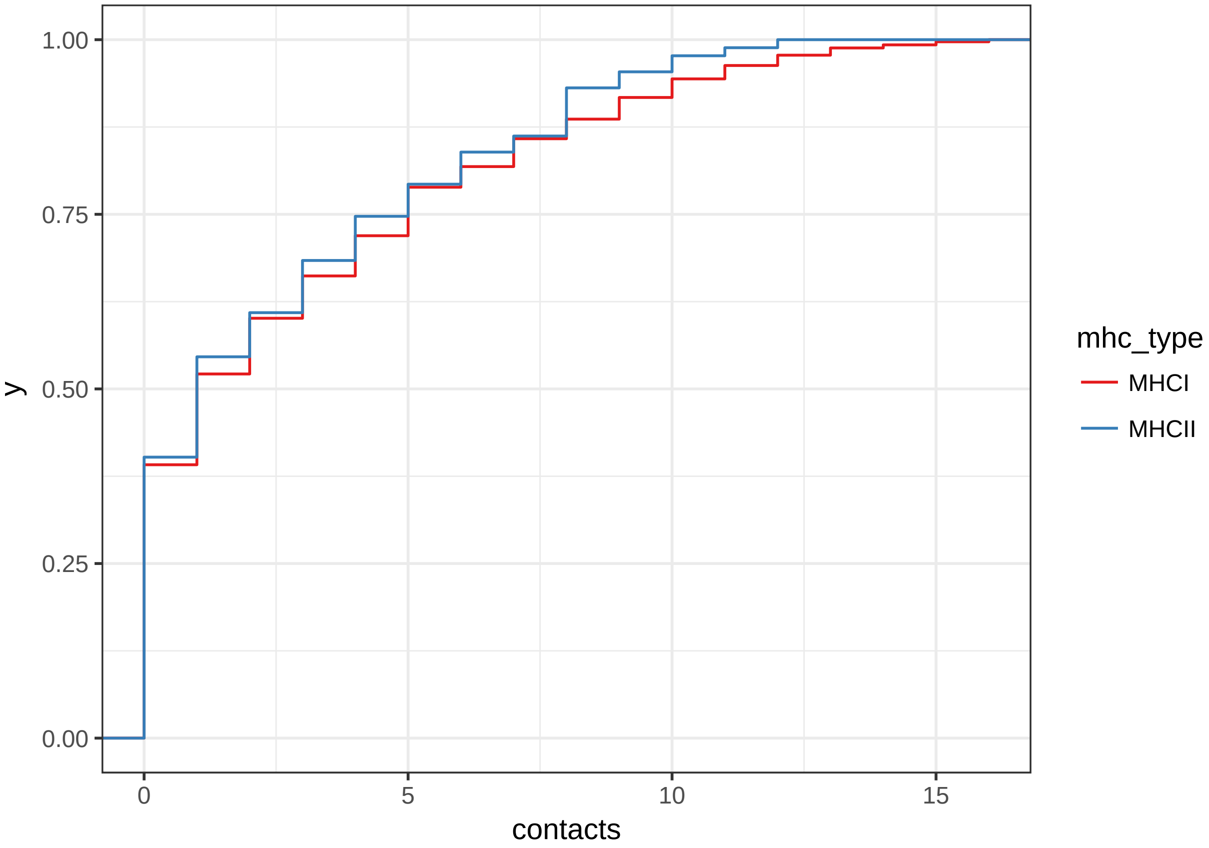


```
ks.test(dt.contact.region.s %>% filter(tcr_gene == "TRA") %>% .$contacts,
        dt.contact.region.s %>% filter(tcr_gene == "TRB") %>% .$contacts)
```

```
## Warning in ks.test(dt.contact.region.s %>% filter(tcr_gene == "TRA") %>% :
## p-value will be approximate in the presence of ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  dt.contact.region.s %>% filter(tcr_gene == "TRA") %>% .$contacts and dt.contact.region.s %>%
## D = 0.040878, p-value = 0.8692
## alternative hypothesis: two-sided
```

```
ggplot(dt.contact.region.s, aes(x = contacts, color = mhc_type)) +
  stat_ecdf() +
  scale_color_brewer(palette = "Set1") +
  theme_bw()
```



```
ks.test(dt.contact.region.s %>% filter(mhc_type == "MHCI") %>% .$contacts,
        dt.contact.region.s %>% filter(mhc_type == "MHCII") %>% .$contacts)
```

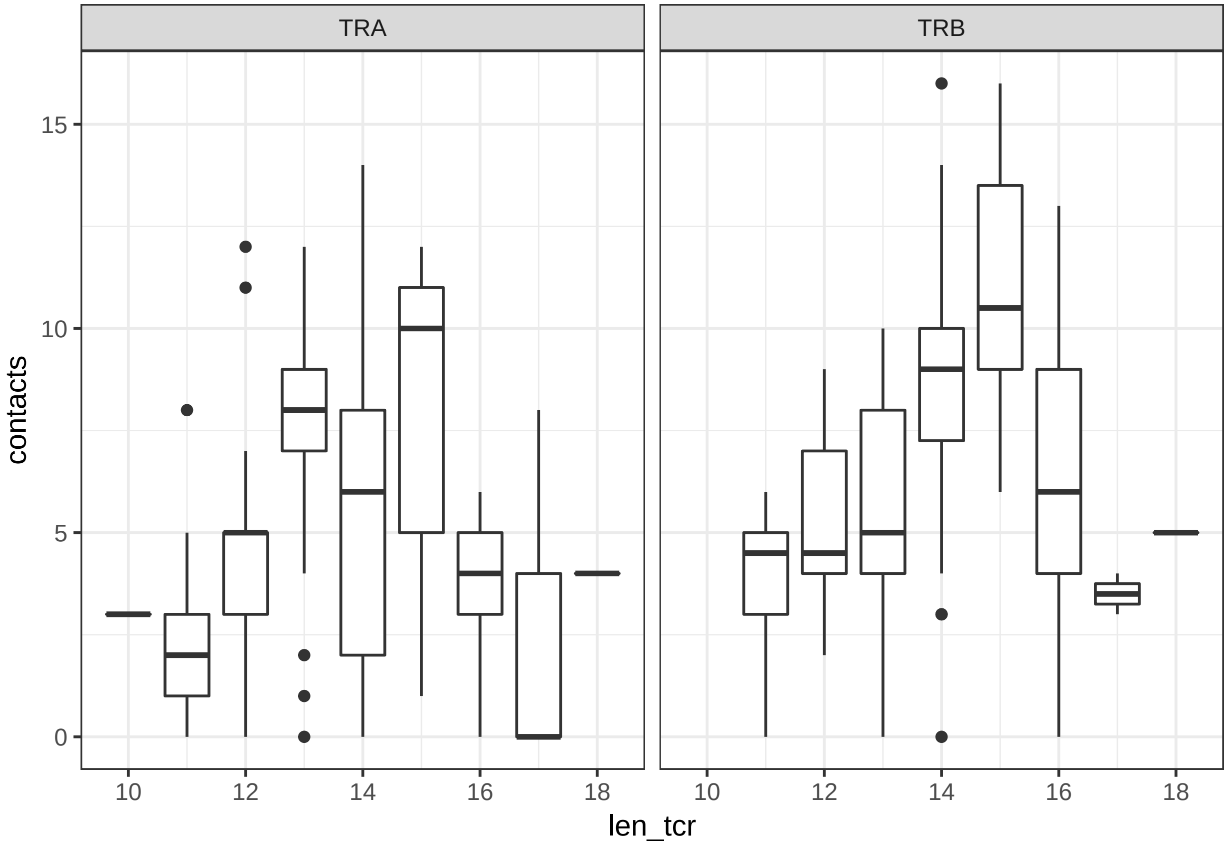
```
## Warning in ks.test(dt.contact.region.s %>% filter(mhc_type == "MHCI") %>% :
## p-value will be approximate in the presence of ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  dt.contact.region.s %>% filter(mhc_type == "MHCI") %>% .$contacts and dt.contact.region.s %>%
## D = 0.044772, p-value = 0.9442
## alternative hypothesis: two-sided
```

Number of contacts and CDR3 length

```
dt.contact.len.tcr = dt.struct %>%
  filter(len_tcr != 5, tcr_region == "CDR3") %>%
  group_by(pdb_id, tcr_gene, len_tcr) %>%
  summarise(contacts = sum(distance <= 4.5))

ggplot(dt.contact.len.tcr, aes(x = len_tcr, y = contacts)) +
  geom_boxplot(aes(group = len_tcr)) +
  facet_wrap( ~ tcr_gene) +
  theme_bw()
```

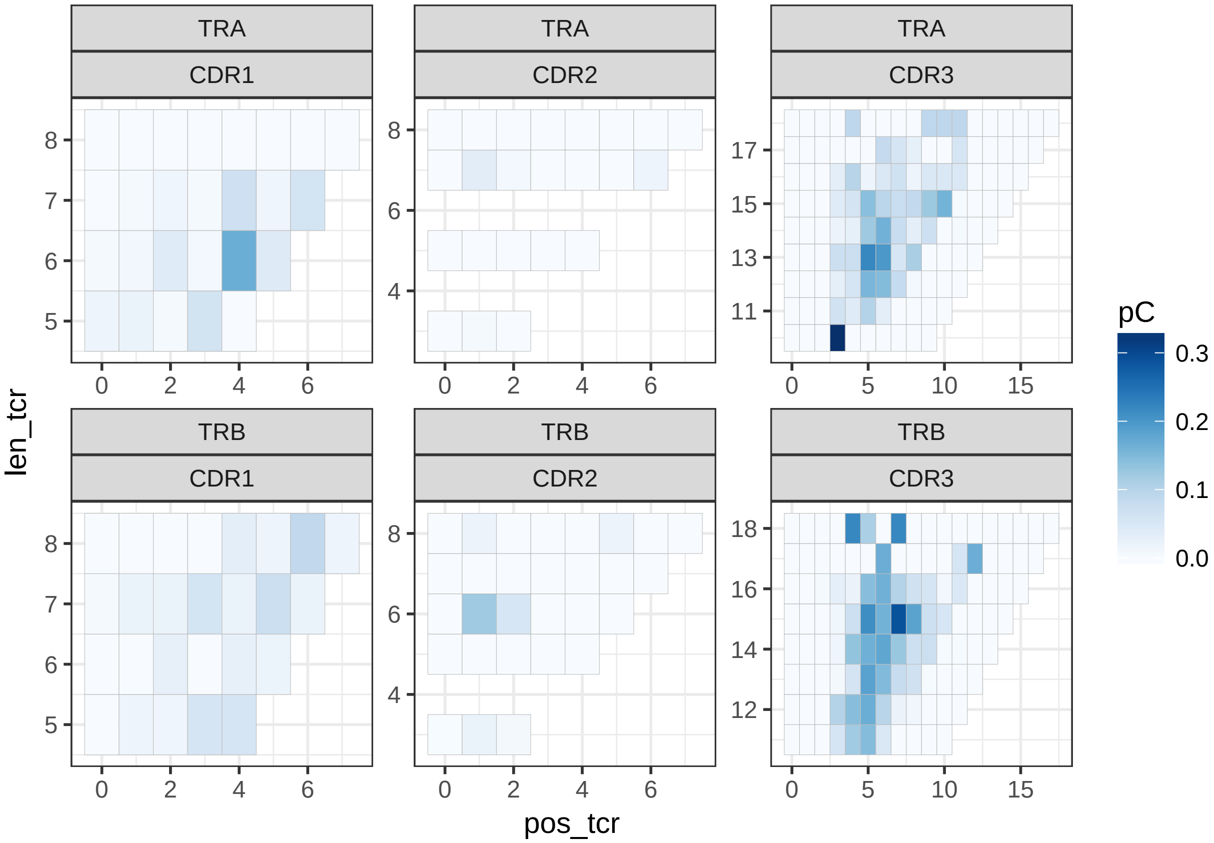


Number of contacts by TCR position $P(C|i, L, R, G)$, 3-4 rule

```
dt.contact.pos.tcr = dt.struct %>%
  group_by(tcr_gene, tcr_region, len_tcr, pos_tcr) %>%
  summarise(pC = mean(distance <= 4.5))

ggplot(dt.contact.pos.tcr %>% filter(!(len_tcr == 5 & tcr_region == "CDR3")),
       aes(x = pos_tcr, y = len_tcr, fill = pC)) +
  geom_tile(color = "grey") +
  facet_wrap(tcr_gene~tcr_region, scales = "free") +
  scale_fill_gradientn("pC", colors=colorRampPalette(brewer.pal(11, 'Blues'))(32)) +
  theme_bw()
```

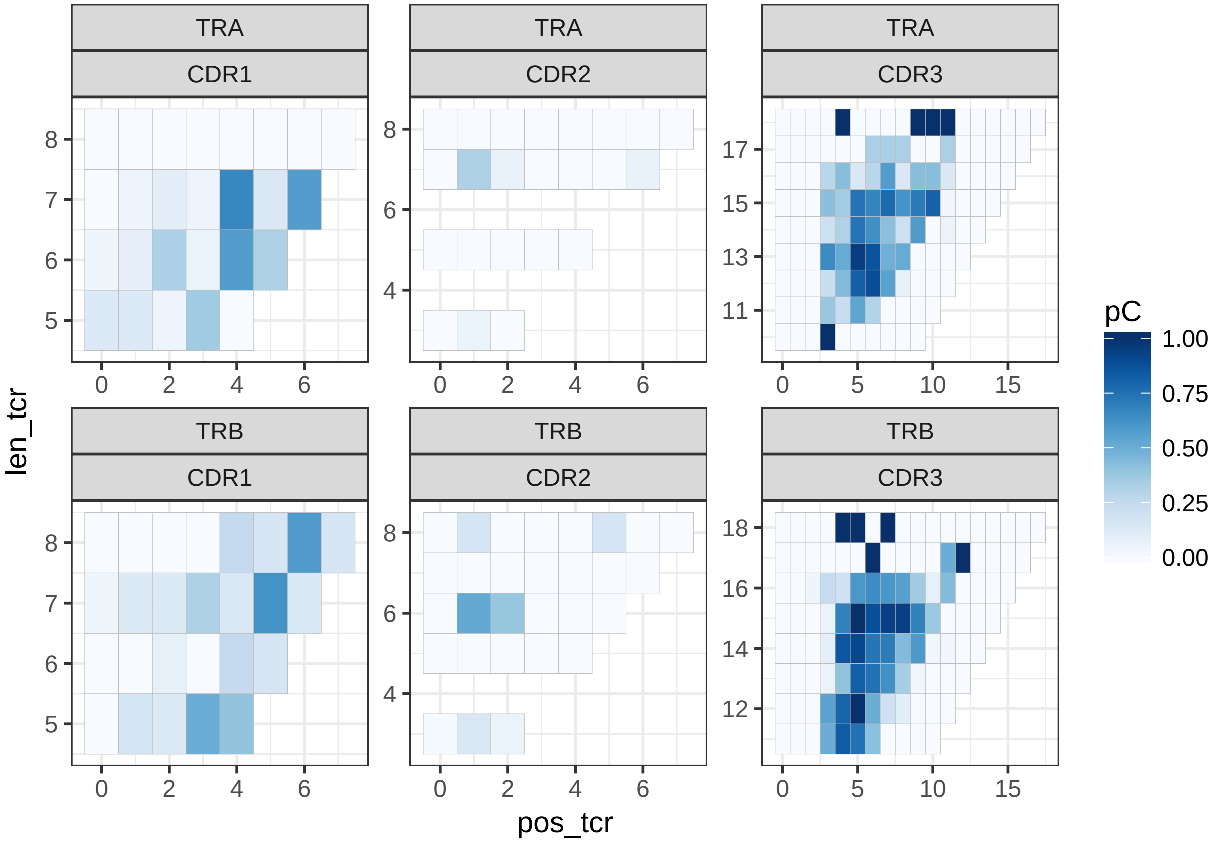
```
## Warning in brewer.pal(11, "Blues"): n too large, allowed maximum for palette Blues is 9
## Returning the palette you asked for with that many colors
```



```
dt.contact.pos.tcr1 = dt.struct %>%
  group_by(pdb_id, tcr_gene, tcr_region, len_tcr, pos_tcr) %>%
  summarise(contact = sum(distance <= 4.5) > 0) %>%
  group_by(tcr_gene, tcr_region, len_tcr, pos_tcr) %>%
  summarise(pC = mean(contact))

ggplot(dt.contact.pos.tcr1 %>% filter(!(len_tcr == 5 & tcr_region == "CDR3")),
       aes(x = pos_tcr, y = len_tcr, fill = pC)) +
  geom_tile(color = "grey") +
  facet_wrap(tcr_gene~tcr_region, scales = "free") +
  scale_fill_gradientn("pC", colors=colorRampPalette(brewer.pal(11, 'Blues'))(32)) +
  theme_bw()
```

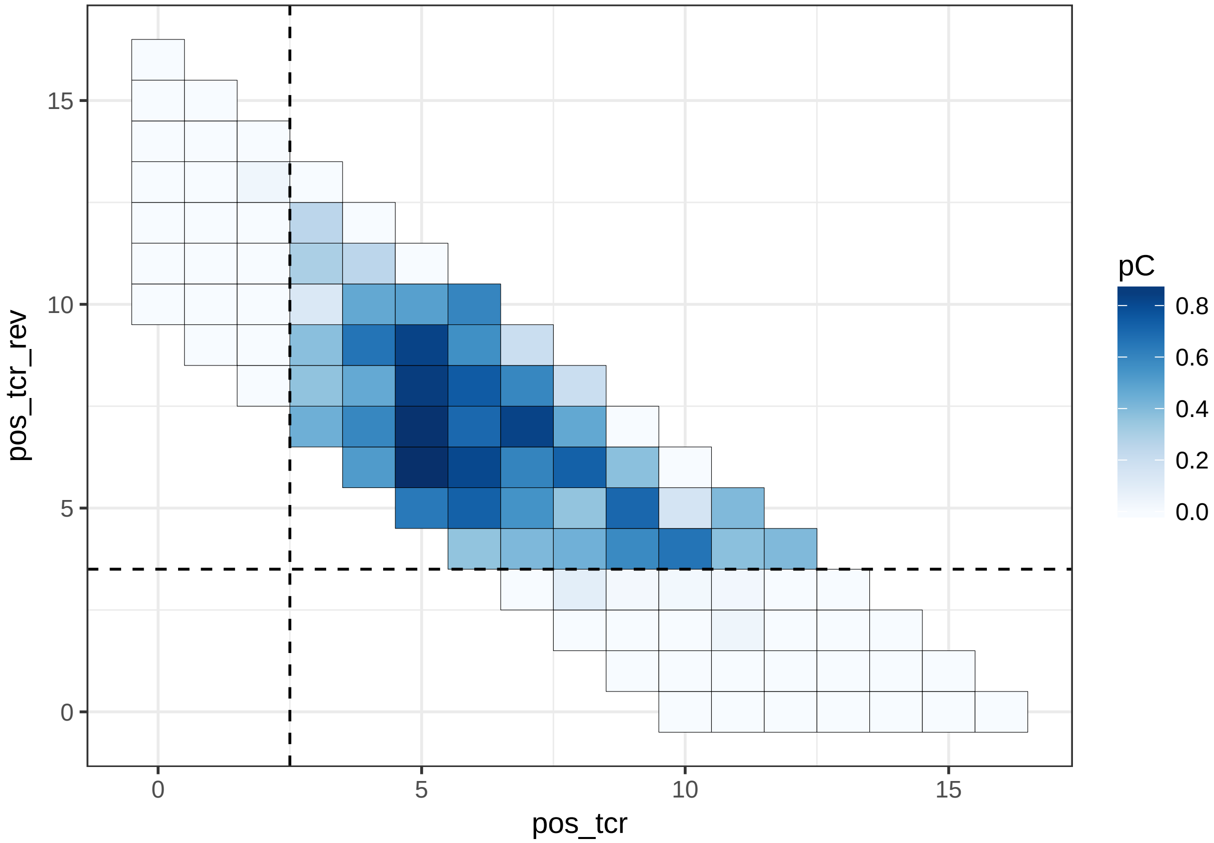
```
## Warning in brewer.pal(11, "Blues"): n too large, allowed maximum for palette Blues is 9
## Returning the palette you asked for with that many colors
```



```
dt.contact.pos.tcr2 = dt.struct %>%
  as.data.frame %>%
  filter(len_tcr != 5, tcr_region == "CDR3") %>%
  group_by(pdb_id, tcr_gene, tcr_region, len_tcr, pos_tcr) %>%
  summarise(contact = sum(distance <= 4.5) > 0) %>%
  group_by(pos_tcr, pos_tcr_rev = len_tcr - pos_tcr - 1) %>%
  summarise(pC = mean(contact), total = n())

ggplot(dt.contact.pos.tcr2 %>% filter(total >= 3),
       aes(x = pos_tcr, y = pos_tcr_rev, fill = pC)) +
  geom_tile(color = "black") +
  geom_vline(xintercept = 2.5, linetype = "dashed") +
  geom_hline(yintercept = 3.5, linetype = "dashed") +
  scale_fill_gradientn("pC", colors=colorRampPalette(brewer.pal(11, 'Blues'))(32)) +
  theme_bw()
```

```
## Warning in brewer.pal(11, "Blues"): n too large, allowed maximum for palette Blues is 9
## Returning the palette you asked for with that many colors
```

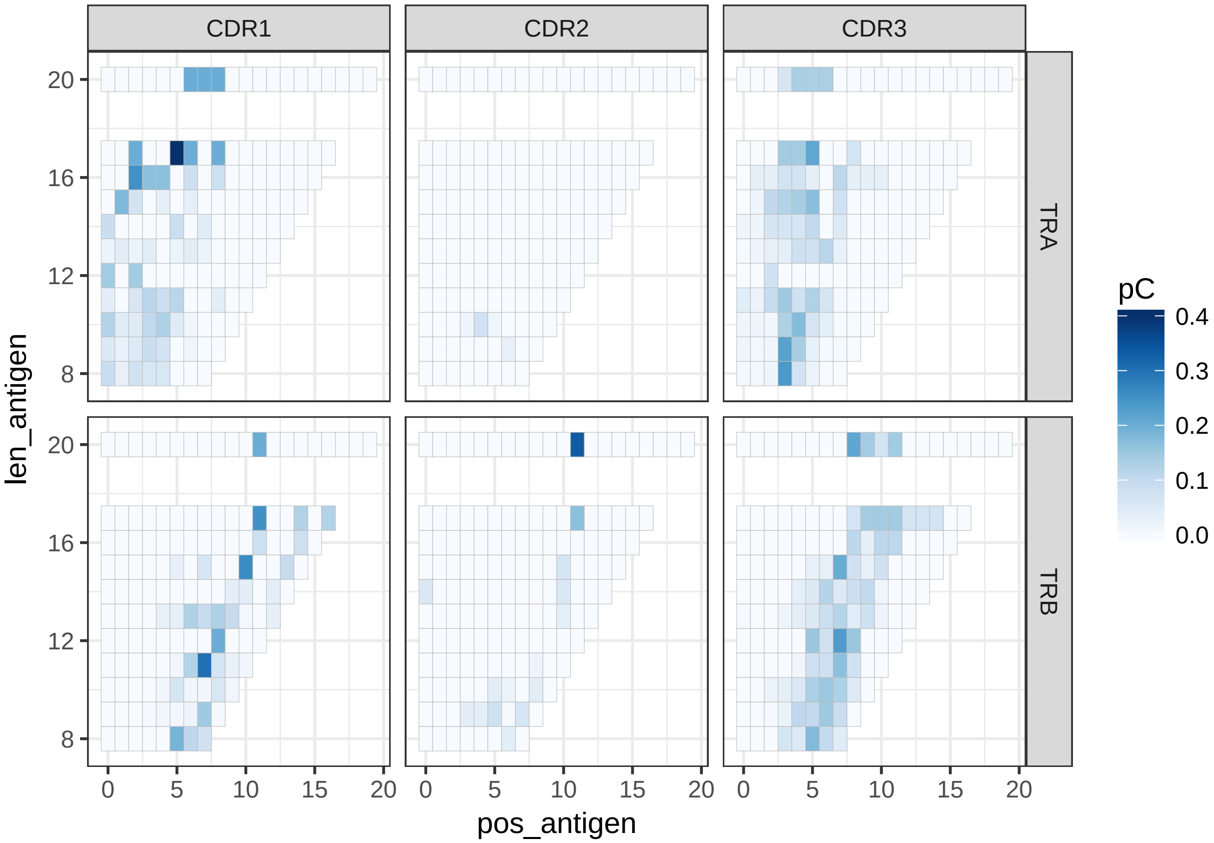


Number of contacts by antigen position $P(C|i, L, R, G)$, MHCI-II difference

```
dt.contact.pos.ag = dt.struct %>%
  group_by(tcr_gene, tcr_region, len_antigen, pos_antigen) %>%
  summarise(pC = mean(distance <= 4.5))

ggplot(dt.contact.pos.ag,
       aes(x = pos_antigen, y = len_antigen, fill = pC)) +
  geom_tile(color = "grey") +
  facet_grid(tcr_gene~tcr_region) +
  scale_fill_gradientn("pC", colors=colorRampPalette(brewer.pal(11, 'Blues'))(32)) +
  theme_bw()
```

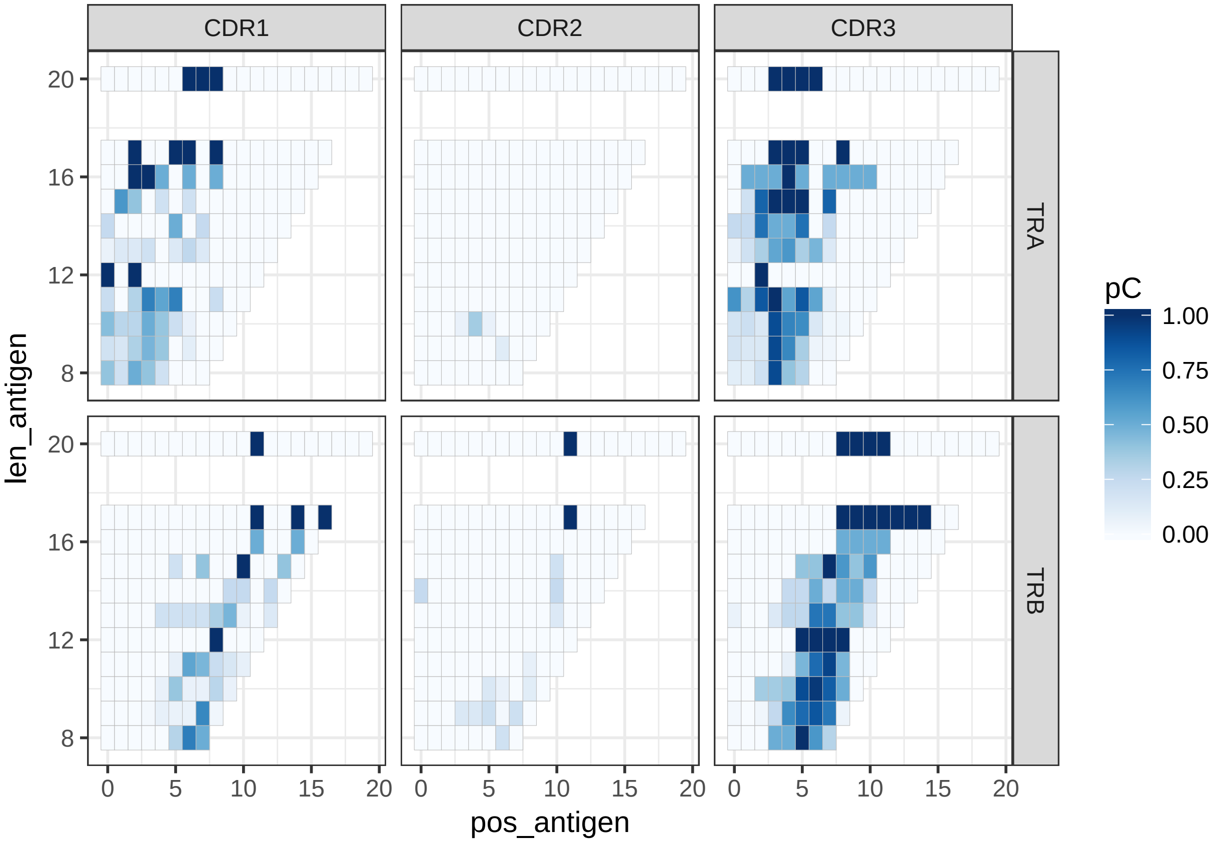
```
## Warning in brewer.pal(11, "Blues"): n too large, allowed maximum for palette Blues is 9
## Returning the palette you asked for with that many colors
```



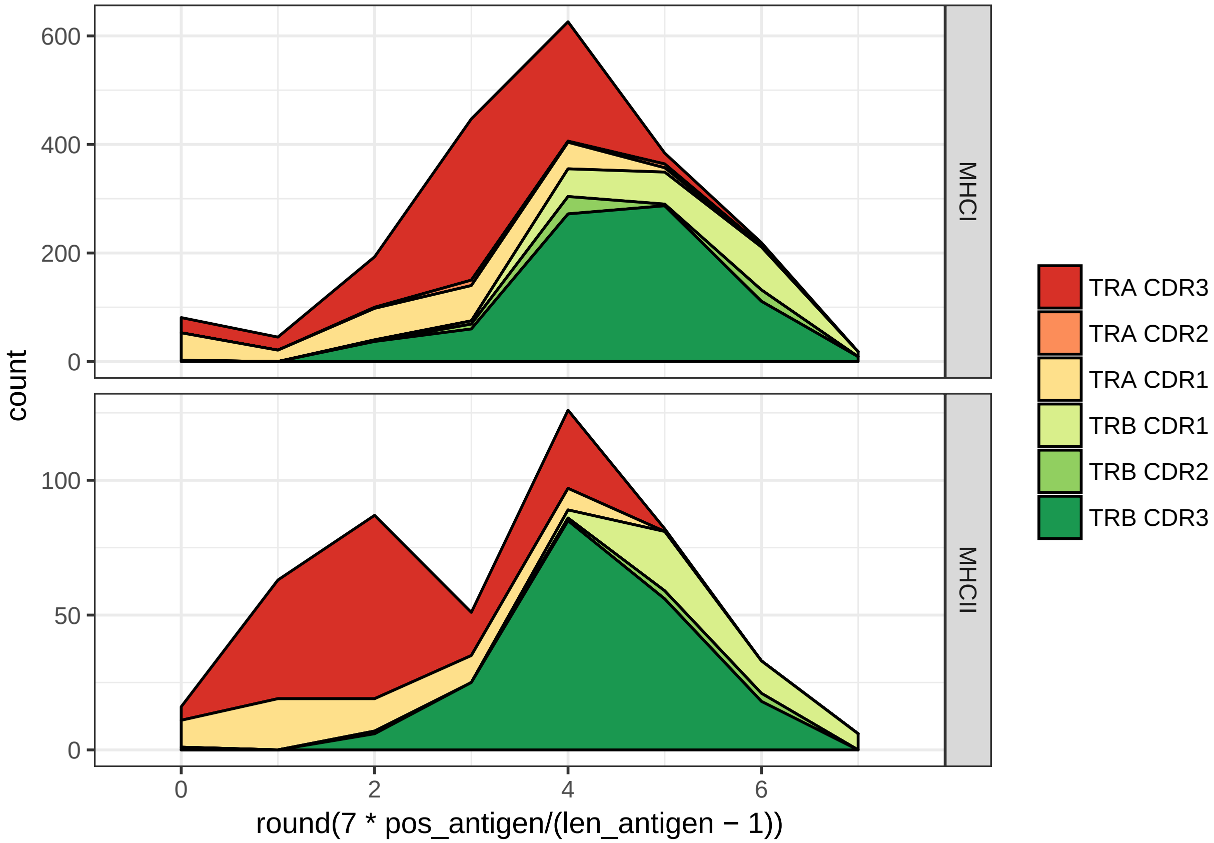
```
dt.contact.pos.ag1 = dt.struct %>%
  group_by(pdb_id, tcr_gene, tcr_region, len_antigen, pos_antigen) %>%
  summarise(contact = sum(distance <= 4.5) > 0) %>%
  group_by(tcr_gene, tcr_region, len_antigen, pos_antigen) %>%
  summarise(pC = mean(contact))

ggplot(dt.contact.pos.ag1,
       aes(x = pos_antigen, y = len_antigen, fill = pC)) +
  geom_tile(color = "grey") +
  facet_grid(tcr_gene ~ tcr_region) +
  scale_fill_gradientn("pC", colors=colorRampPalette(brewer.pal(11, 'Blues'))(32)) +
  theme_bw()
```

```
## Warning in brewer.pal(11, "Blues"): n too large, allowed maximum for palette Blues is 9
## Returning the palette you asked for with that many colors
```



```
ggplot(dt.struct %>% filter(distance <= 4.5),
       aes(x = round(7*pos_antigen / (len_antigen - 1)),
           fill = factor(paste(tcr_gene, tcr_region),
                         levels = c("TRA CDR3",
                                    "TRA CDR2",
                                    "TRA CDR1",
                                    "TRB CDR1",
                                    "TRB CDR2",
                                    "TRB CDR3")))) +
  geom_area(aes(y = ..count..), stat = "bin", binwidth = 1, position = "stack", color ="black") +
  scale_fill_brewer("", palette = "RdYlGn") +
  facet_grid(mhc_type~., scales = "free_y") +
  theme_bw()
```


600
400
200
0
count
100
50
0
0
2
4
6
round(7 * pos_antigen/(len_antigen − 1))
MHCI
MHCII
TRA CDR3
TRA CDR2
TRA CDR1
TRB CDR1
TRB CDR2
TRB CDR3