Untitled

```r
library(Biostrings)
```

```
## Loading required package: BiocGenerics
```

```
## Loading required package: parallel
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB
```

```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
```

```
## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, cbind, colMeans,
##     colnames, colSums, do.call, duplicated, eval, evalq, Filter,
##     Find, get, grep, grepl, intersect, is.unsorted, lapply,
##     lengths, Map, mapply, match, mget, order, paste, pmax,
##     pmax.int, pmin, pmin.int, Position, rank, rbind, Reduce,
##     rowMeans, rownames, rowSums, sapply, setdiff, sort, table,
##     tapply, union, unique, unsplit, which, which.max, which.min
```

```
## Loading required package: S4Vectors
```

```
## Loading required package: stats4
```

```
##
## Attaching package: 'S4Vectors'
```

```
## The following object is masked from 'package:base':
##
##     expand.grid
```

```
## Loading required package: IRanges
```

```
## Loading required package: XVector
```

```
##
## Attaching package: 'Biostrings'
```

```
## The following object is masked from 'package:base':
##
##     strsplit
```

```r
library(parallel)
library(data.table)
```

```
##
## Attaching package: 'data.table'
## The following object is masked from 'package:IRanges':
##
##     shift

## The following objects are masked from 'package:S4Vectors':
##
##     first, second
```

```r
library(dplyr)
```

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

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

## The following objects are masked from 'package:Biostrings':
##
##     collapse, intersect, setdiff, setequal, union

## The following object is masked from 'package:XVector':
##
##     slice

## The following objects are masked from 'package:IRanges':
##
##     collapse, desc, intersect, setdiff, slice, union

## The following objects are masked from 'package:S4Vectors':
##
##     first, intersect, rename, setdiff, setequal, union

## The following objects are masked from 'package:BiocGenerics':
##
##     combine, intersect, setdiff, union

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

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

```r
library(stringr)
library(ggplot2)
library(RColorBrewer)
select = dplyr::select
```

Load V segments

```r
df.v = fread("251117.cdr12.txt") %>%
  mutate(v = gene, gene = substr(gene, 1, 3)) %>%
  select(species, gene, v, seqaa) %>%
  filter(gene %in% c("TRA", "TRB"))
```

All combinations

```r
df.v.comb = df.v %>% mutate(v.1 = v, seqaa.1 = seqaa) %>% select(-v, -seqaa) %>%
  merge(df.v %>% mutate(v.2 = v, seqaa.2 = seqaa) %>% select(-v, -seqaa))
```

Compute distances

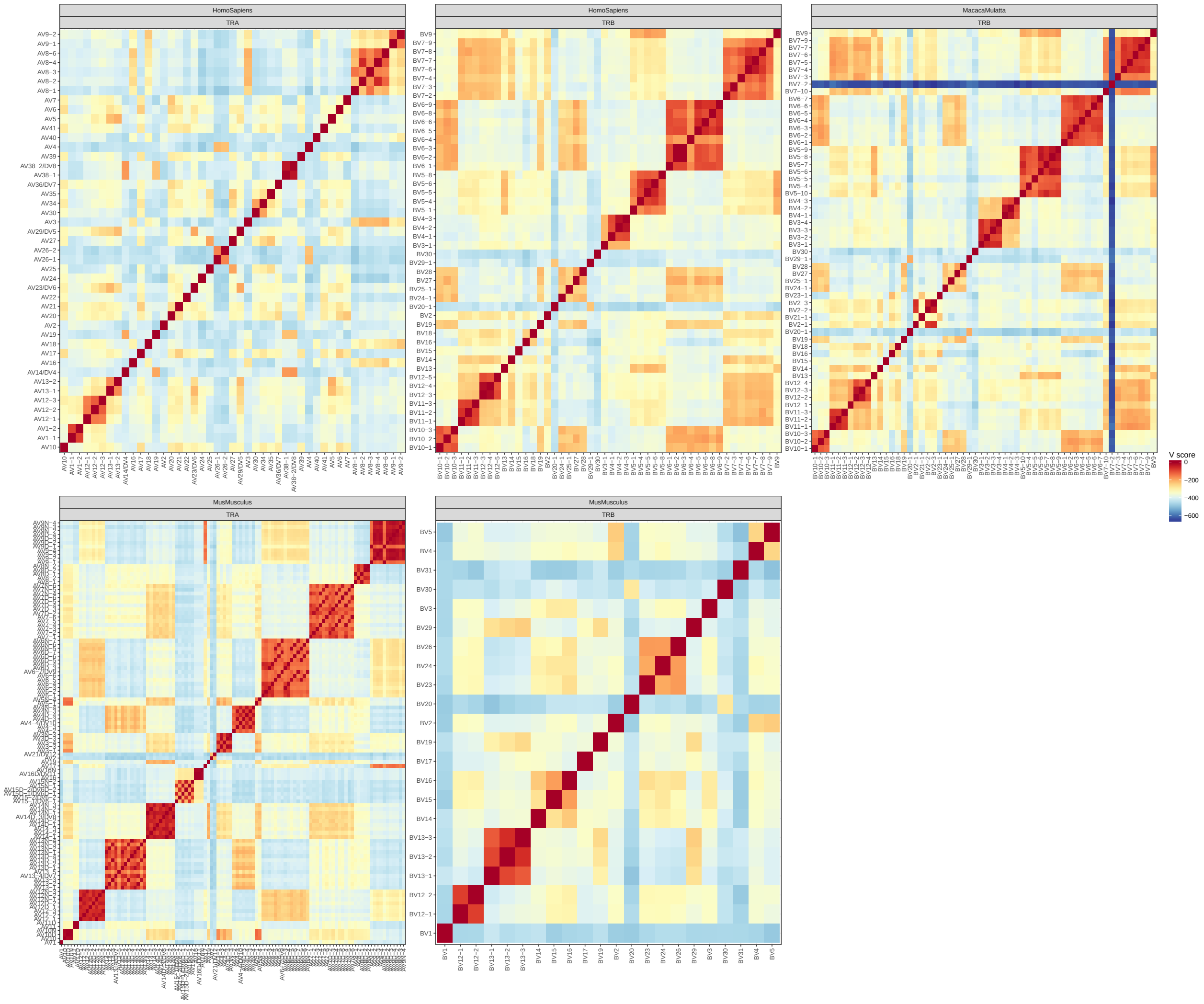
```r
aln_fun = function(s1, s2) {
  score(pairwiseAlignment(AAString(s1),
                          AAString(s2),
                          substitutionMatrix = "BLOSUM62"))
}

df.v.comb$score = mcmapply(function(a,b) aln_fun(a, b),
                           df.v.comb$seqaa.1, df.v.comb$seqaa.2,
                           mc.cores = 80)
```

Normalize distances

```r
df.v.comb = df.v.comb %>%
  group_by(species, gene) %>%
  mutate(score.s = sum(score)) %>%
  group_by(species, gene, v.1) %>%
  mutate(score.s1 = sum(score), score.m1 = score[which(v.2 == v.1)]) %>%
  group_by(species, gene, v.2) %>%
  mutate(score.s2 = sum(score), score.m2 = score[which(v.2 == v.1)]) %>%
  ungroup %>%
  mutate(score.norm.m = score - pmax(score.m1, score.m2),
         score.norm.s = score * score.s / score.s1 / score.s2)
```

```r
ggplot(df.v.comb, aes(x=gsub("TR", "", str_split_fixed(v.1, fixed("*"), 2)[,1]),
                      y=gsub("TR", "", str_split_fixed(v.2, fixed("*"), 2)[,1]),
                      fill = score.norm.m)) +
  geom_tile() +
  facet_wrap(species ~ gene, scales = "free") +
  xlab("") + ylab("") +
  scale_fill_gradientn("V score",
                       colors=colorRampPalette(rev(brewer.pal(11, 'RdYlBu')))(32)) +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))
```


HomoSapiens
TRA
HomoSapiens
TRB
MacacaMulatta
TRB
MusMusculus
TRA
MusMusculus
TRB
V score
0
−200
−400
−600
BV1
BV12−1
BV12−2
BV13−1
BV13−2
BV13−3
BV14
BV15
BV16
BV17
BV19
BV2
BV20
BV23
BV24
BV26
BV29
BV3
BV30
BV31
BV4
BV5