shm_analysis

Analysis of substitution type and frequency

Load preprocessed data

```
library(dplyr)
library(ggplot2)
library(data.table)
library(scales)

select = dplyr::select
summarise = dplyr::summarise

load("shm_rep2.rda")
df = data.table(shm) %>%
  mutate(region = as.character(region)) %>%
  filter(type == "RNA") %>%
  mutate(region = factor(region, c("FR1","CDR1","FR2","CDR2","FR3","CDR3","FR4"))) %>%
  mutate(mutation.type = ifelse(as.character(from.aa) == as.character(to.aa), "S", "R")) %>%
  group_by(proj, type, sample, replica) %>%
  mutate(weight2 = 1/n())
```

Check if we observe well-documented increase in replacement:synonimic hypermutation ratio in CDR regions:

```
p5=ggplot(df %>% mutate(weight2 = weight2/sum(weight2))) +
  geom_histogram(binwidth = 1, aes(x=pos.aa, weight=weight2, fill=region)) +
  #geom_density(adjust=2, aes(x=pos.aa, weight=weight, fill=region), position = "stack", color = NA) +
  geom_density(binwidth = 1, aes(x=pos.aa, weight=weight2, linetype = mutation.type)) +
  scale_y_continuous("Hypermutation density", expand = c(0,0)) +
  scale_x_continuous("Position in IG, AA", expand = c(0,0)) +
  scale_linetype("SHM type") +
  scale_fill_brewer("Region", palette = "Paired") +
  theme_bw() +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        legend.position = c(0.75, 0.81),
        legend.direction = "horizontal")
```

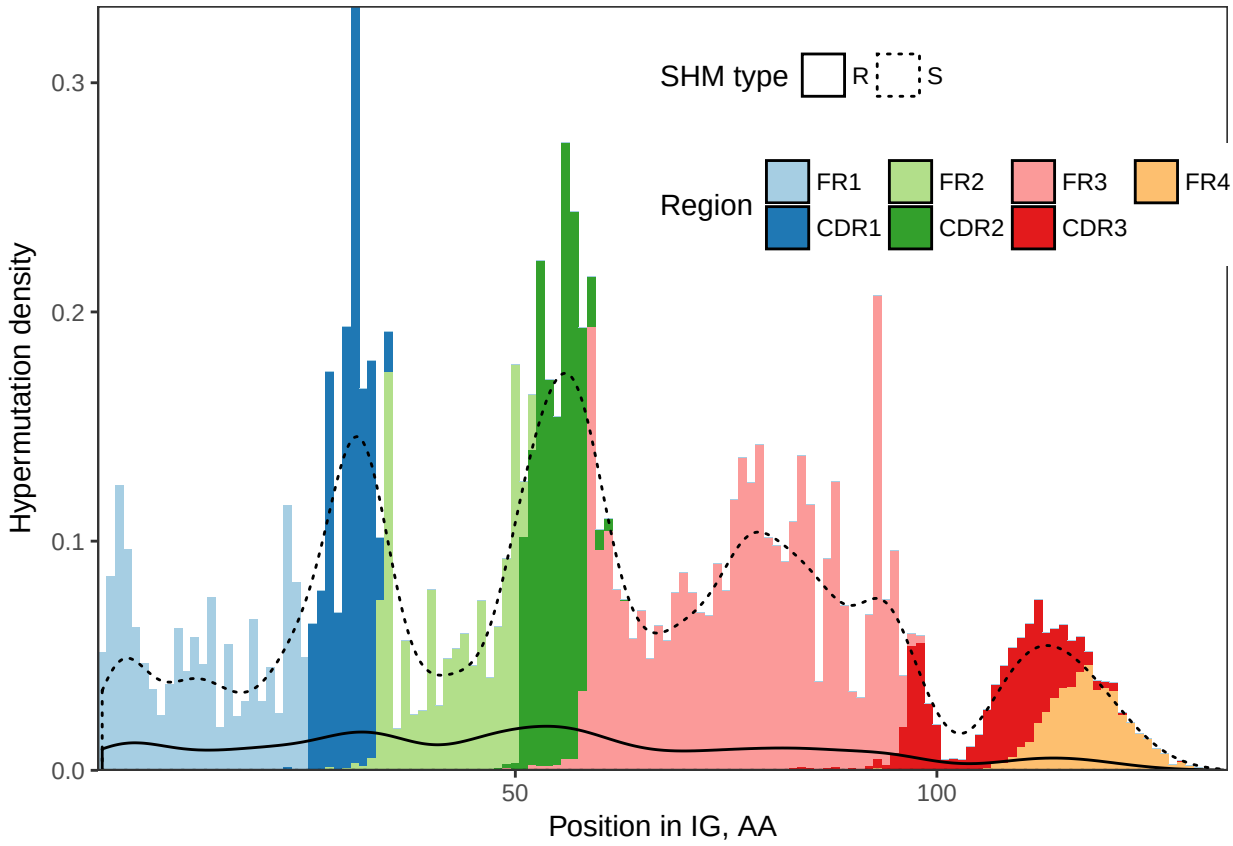
```
## Warning: Ignoring unknown aesthetics: weight
```

```
## Warning: Ignoring unknown parameters: binwidth
```

```
p5
```

```
## Warning in density.default(x, weights = w, bw = bw, adjust = adjust, kernel
## = kernel, : sum(weights) != 1 -- will not get true density
```

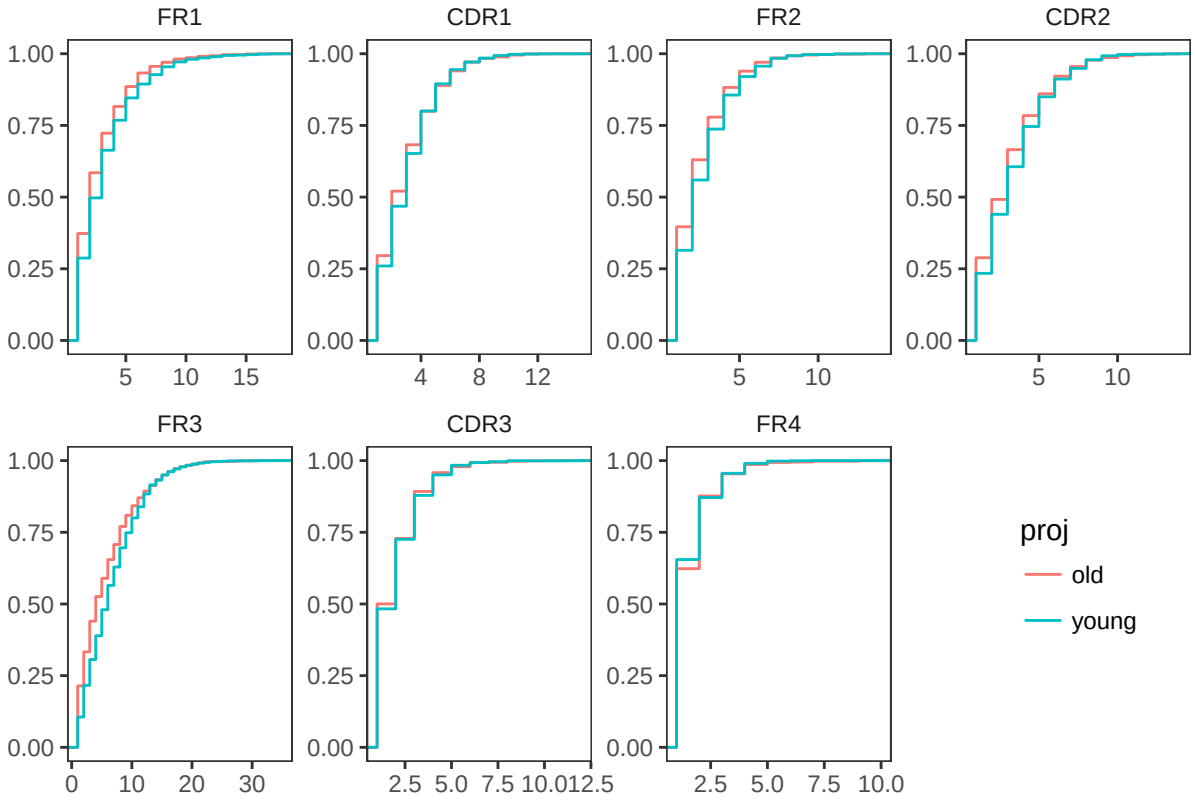
```
## Warning in density.default(x, weights = w, bw = bw, adjust = adjust, kernel
## = kernel, : sum(weights) != 1 -- will not get true density
```



Relative hypermutation burden by region for old and young

```
df <- df %>%
  filter(from.aa != "" | to.aa != "" | is.na(pos.nt))

df.shm.share = df %>%
  group_by(proj, sample, region) %>%
  summarise(count = n()) %>%
  group_by(sample) %>%
  mutate(share = count/sum(count))
```

```
df.shm.1 = df %>%
  group_by(proj, sample, region, clone) %>%
  summarise(shms = n())

ggplot(df.shm.1, aes(x = shms, group = proj, color = proj)) +
  stat_ecdf() +
  scale_y_continuous("") +
  xlab("") +
  scale_fill_brewer("Age", palette = "Set1") +
  facet_wrap(~region, scales = "free", ncol=4) +
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = c(0.90, 0.25),
        strip.background = element_blank())
```

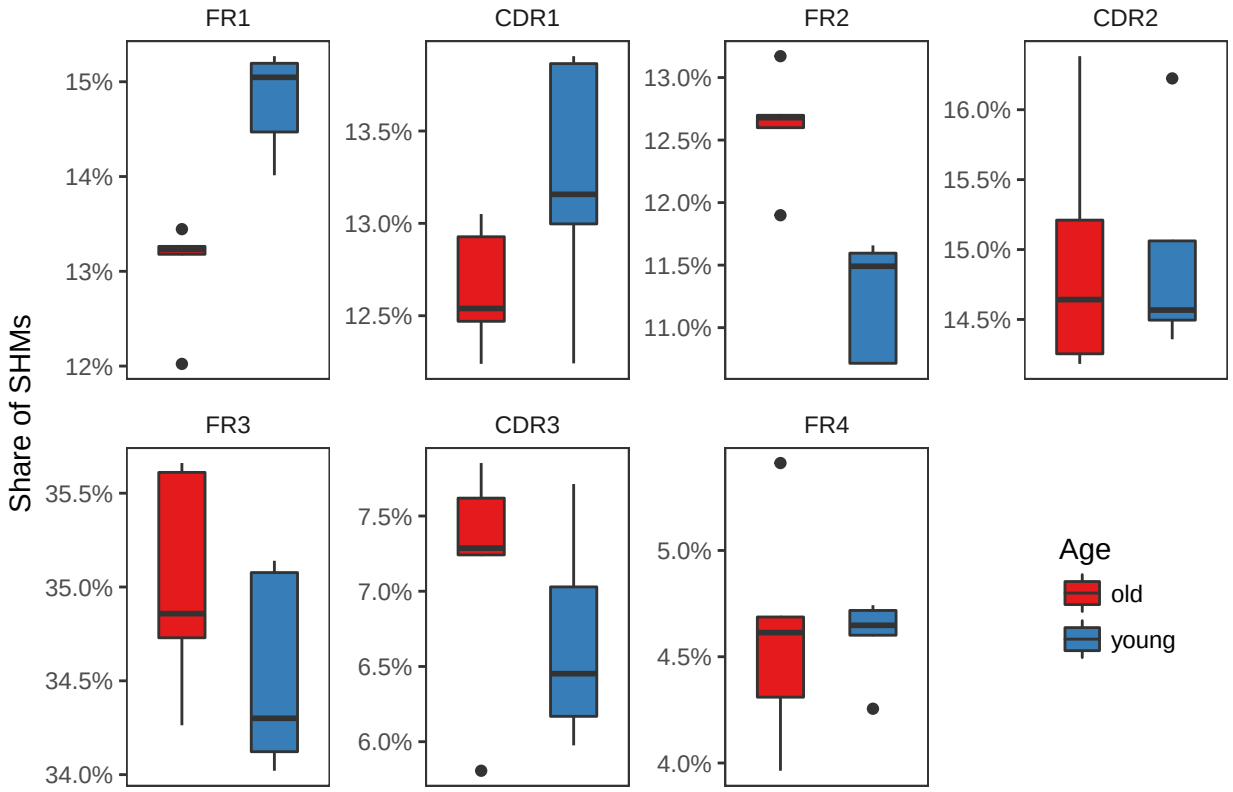



```
dt.p = data.table()
for (r in unique(df.shm.share$region)) {
  tt = t.test(share~proj, df.shm.share %>% filter(region == r))
  p = tt$p.value
  dt.p = rbind(dt.p,
               data.table(region = r, p.adj = p))
}
dt.p$p.adj = p.adjust(dt.p$p.adj, method="BH")
print(dt.p)
```

```
##    region       p.adj
## 1:    FR1 0.005831331
## 2:   CDR1 0.323091924
## 3:    FR2 0.005831331
## 4:   CDR2 0.989881416
## 5:    FR3 0.365904903
## 6:   CDR3 0.463264841
## 7:    FR4 0.989881416
```

```
ggplot(df.shm.share, aes(x = proj, group = proj, y = share, fill = proj)) +
  geom_boxplot(width = 0.5) +
  scale_y_continuous("Share of SHMs", label = percent) +
  xlab("") +
  scale_fill_brewer("Age", palette = "Set1") +
  facet_wrap(~region, scales = "free", ncol=4) +
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = c(0.90, 0.25),
```

```
        axis.text.x = element_blank(), axis.ticks.x = element_blank(),
        strip.background = element_blank())
```



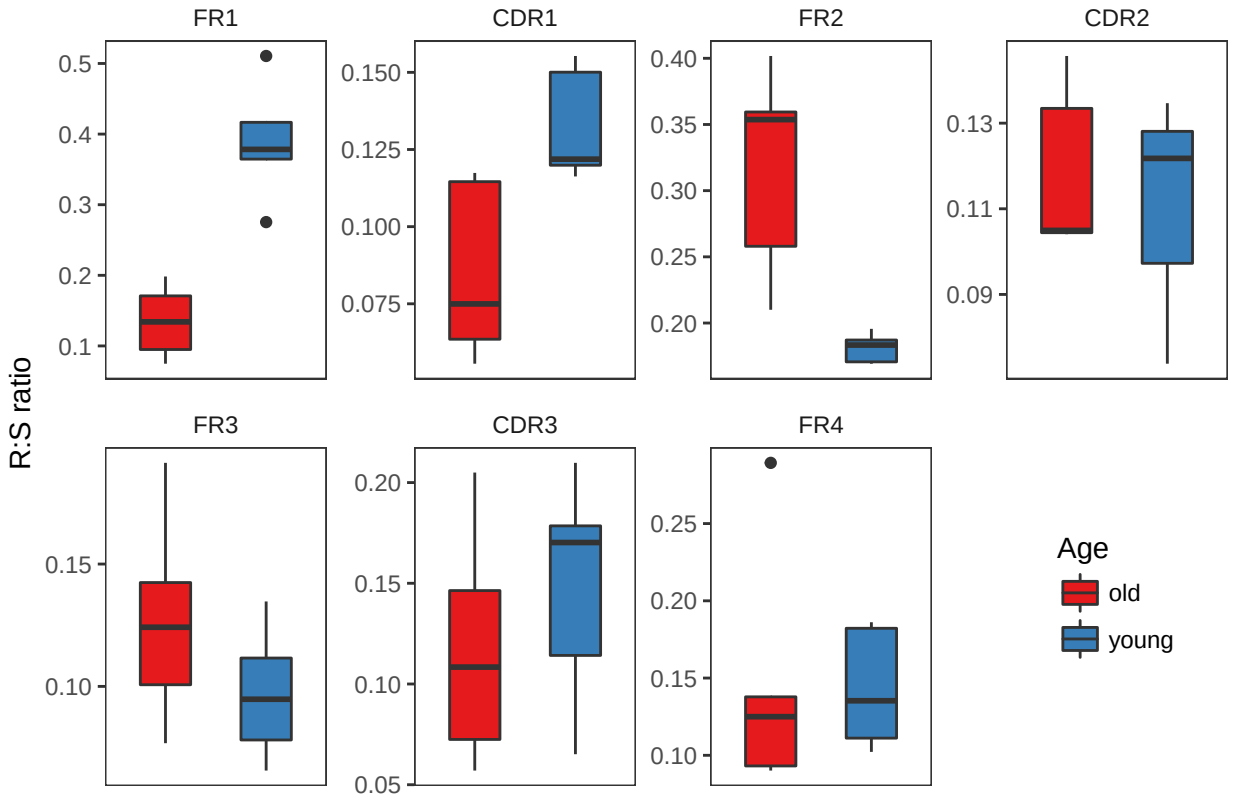

```
dt.p = data.table()
for (r in unique(df.shm.share$region)) {
  tt = t.test(share~proj, df.shm.share %>% filter(region == r))
  p = tt$p.value
  dt.p = rbind(dt.p,
               data.table(region = r, p.adj = p))
}
dt.p$p.adj = p.adjust(dt.p$p.adj, method="BH")
print(dt.p)
```

```
##    region       p.adj
## 1:    FR1 0.005831331
## 2:   CDR1 0.323091924
## 3:    FR2 0.005831331
## 4:   CDR2 0.989881416
## 5:    FR3 0.365904903
## 6:   CDR3 0.463264841
## 7:    FR4 0.989881416
```

Replacement to silent ratio

```
df.shm.rs = df %>%
  group_by(proj, sample, region, mutation.type) %>%
  summarise(count = n()) %>%
  group_by(proj, sample, region) %>%
  summarise(rs = sum(c(0,count[which(mutation.type == "R")])) / sum(c(0,count[which(mutation.type == "S
```

```
ggplot(df.shm.rs, aes(x = proj, group = proj, y = rs, fill = proj)) +
  geom_boxplot(width = 0.5) +
  scale_y_continuous("R:S ratio") +
  xlab("") +
  scale_fill_brewer("Age", palette = "Set1") +
  facet_wrap(~region, scales = "free", ncol=4) +
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = c(0.90, 0.25),
        axis.text.x = element_blank(), axis.ticks.x = element_blank(),
        strip.background = element_blank())
```



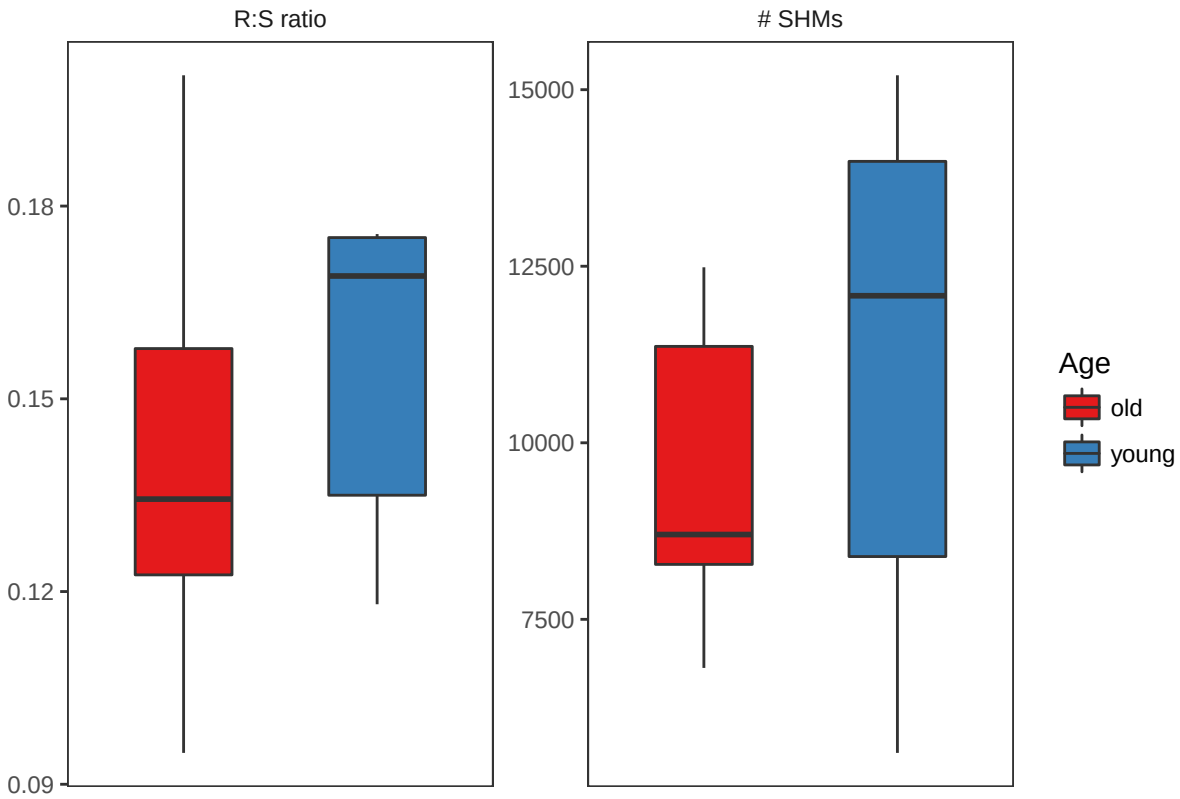

```
dt.p = data.table()
for (r in unique(df.shm.rs$region)) {
  tt = t.test(rs~proj, df.shm.rs %>% filter(region == r))
  p = tt$p.value
  dt.p = rbind(dt.p,
               data.table(region = r, p=p,p.adj = p))
}
dt.p$p.adj = p.adjust(dt.p$p.adj, method="BH")
print(dt.p)
```

```
##    region           p       p.adj
## 1:    FR1 0.0009172687 0.006420881
## 2:   CDR1 0.0182061406 0.042556498
## 3:    FR2 0.0182384989 0.042556498
## 4:   CDR2 0.6200315297 0.723370118
```

```
## 5:    FR3 0.2332620000 0.408208500
## 6:   CDR3 0.4460072914 0.624410208
## 7:    FR4 0.9310035113 0.931003511
```

Overall statistics

```
df.shm.share.s = df %>%
  group_by(proj, sample, stat="# SHMs") %>%
  summarise(value = n())

df.shm.rs.s = df %>%
  group_by(proj, sample, mutation.type) %>%
  summarise(count = n()) %>%
  group_by(proj, sample, stat="R:S ratio") %>%
  summarise(value = sum(c(0,count[which(mutation.type == "R")])) / sum(c(0,count[which(mutation.type ==

tmp = rbind(df.shm.share.s,
            df.shm.rs.s)

p6=ggplot(tmp,
       aes(x = proj, y = value, fill = proj))+
geom_boxplot(width = 0.5) +
  scale_y_continuous("") +
  xlab("") +
  scale_fill_brewer("Age", palette = "Set1") +
  facet_wrap(~stat, scales = "free") +
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.text.x = element_blank(), axis.ticks.x = element_blank(),
        strip.background = element_blank())
p6
```




```
dt.p = data.table()
for (ss in unique(tmp$stat)) {
  tt = t.test(value ~ proj, tmp %>% filter(stat == ss))
  p = tt$p.value
  dt.p = rbind(dt.p,
               data.table(stat = ss, p=p))
}
print(dt.p)
```

```
##         stat         p
## 1:    # SHMs 0.4862387
## 2: R:S ratio 0.5745291
```

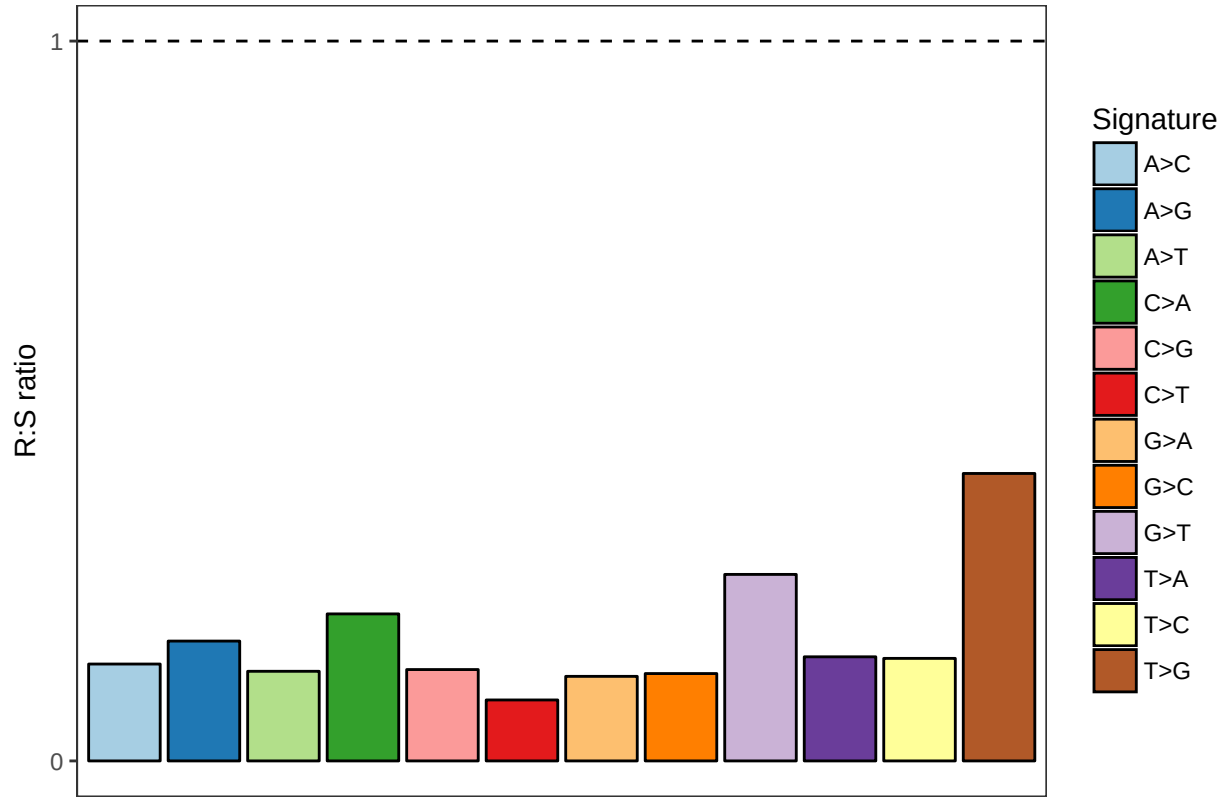
Substitution patterns

```
df.sign = df %>%
  mutate(mutation.signature = paste(from.nt, to.nt, sep = ">"))
```

R:S ratio across different substitution patterns at nucleotide level

```
df.sign.rs = df.sign %>%
  group_by(mutation.signature, mutation.type) %>%
  summarise(count = n()) %>%
  dcast(mutation.signature ~ mutation.type, value.var = "count") %>%
  mutate(rs = R/S)

p7=ggplot(df.sign.rs, aes(x = mutation.signature, y = rs, fill = mutation.signature)) +
  geom_bar(stat="identity", color="black") +
  geom_hline(yintercept = 1, linetype="dashed") +
  scale_fill_brewer("Signature", palette = "Paired") +
```

```
  scale_y_continuous("R:S ratio", breaks=0:8) +
  xlab("") +
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.text.x = element_blank(), axis.ticks.x = element_blank())
p7
```

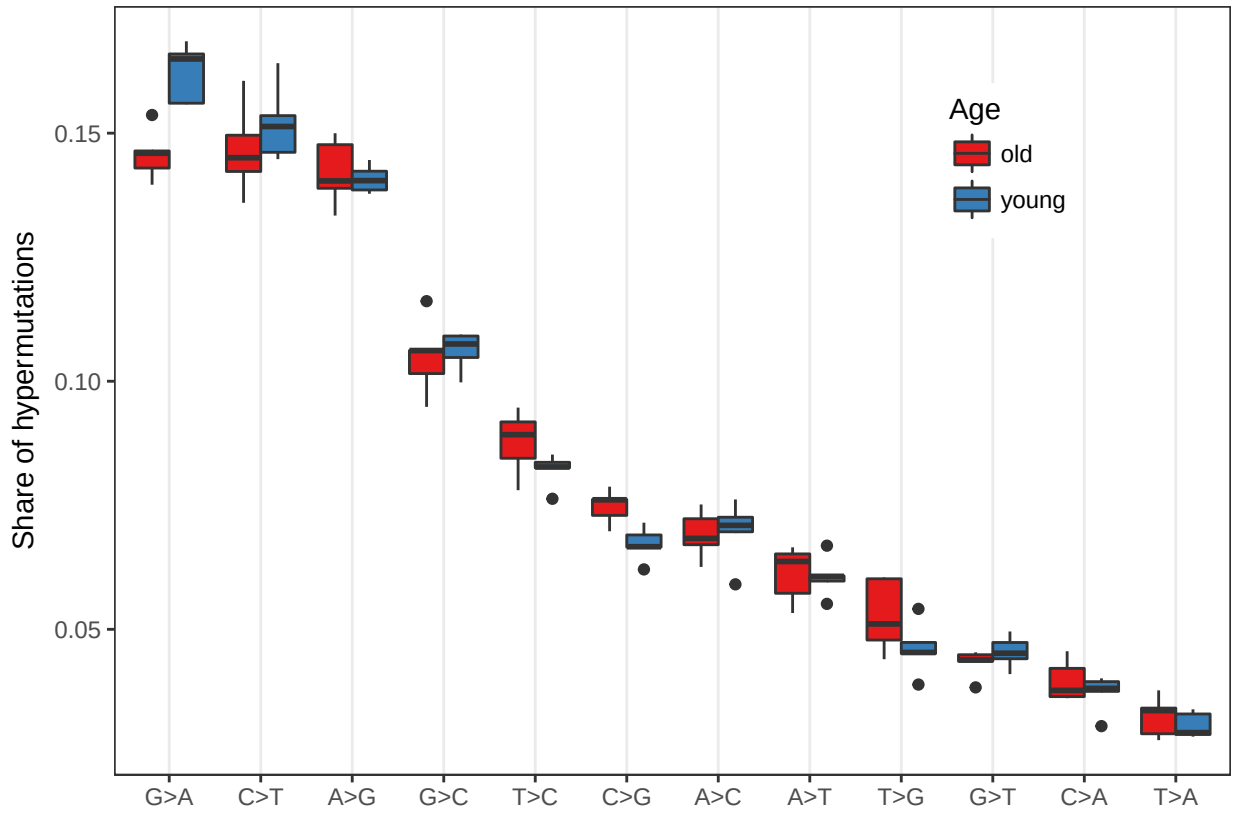


Share of different substitution patterns compared between young and old. Age-related difference observed for certain substitution patterns aross hypermutations.

```
sign.dict = data.frame(mutation.signature = c("A>C","A>G","A>T","C>A","C>G","C>T","G>A",
                                              "G>C","G>T","T>A","T>C","T>G"),
                       mutation.signature.rep = c("A>C,T>G","A>G,T>C","A>T,T>A","C>A,G>T",
                                                  "C>G,G>C","C>T,G>A","C>T,G>A","C>G,G>C",
                                                  "C>A,G>T","A>T,T>A","A>G,T>C","A>C,T>G"))

df.sign.total = df.sign %>%
  group_by(sample) %>%
  dplyr::summarize(total = n())

df.sign.s = df.sign %>%
  group_by(proj, sample, mutation.signature) %>%
  summarise(count = n()) %>%
  merge(df.sign.total) %>%
  mutate(freq = count / total)
```

```
df.sign.s$mutation.signature = factor(df.sign.s$mutation.signature,
                                      levels = with(df.sign.s %>% group_by(mutation.signature) %>%
```

```
                                                  summarise(freq = sum(freq)),
                                                mutation.signature[order(-freq)]))

p8=ggplot(df.sign.s, aes(x=mutation.signature, y = freq, fill = proj)) +
  geom_boxplot() +
  ylab("") + xlab("") +
  scale_fill_brewer("Age", palette = "Set1") +
  ylab("Share of hypermutations") +
  theme_bw() +
  theme(panel.grid.major.y = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = c(0.8, 0.8))
p8
```




```
a = aov(freq ~ proj * mutation.signature, df.sign.s)
summary(a)
```

```
##                          Df  Sum Sq  Mean Sq F value   Pr(>F)
## proj                      1 0.00000 0.000000    0.00 1.000000
## mutation.signature       11 0.21447 0.019497  681.23  < 2e-16 ***
## proj:mutation.signature  11 0.00113 0.000102    3.58 0.000301 ***
## Residuals                96 0.00275 0.000029
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

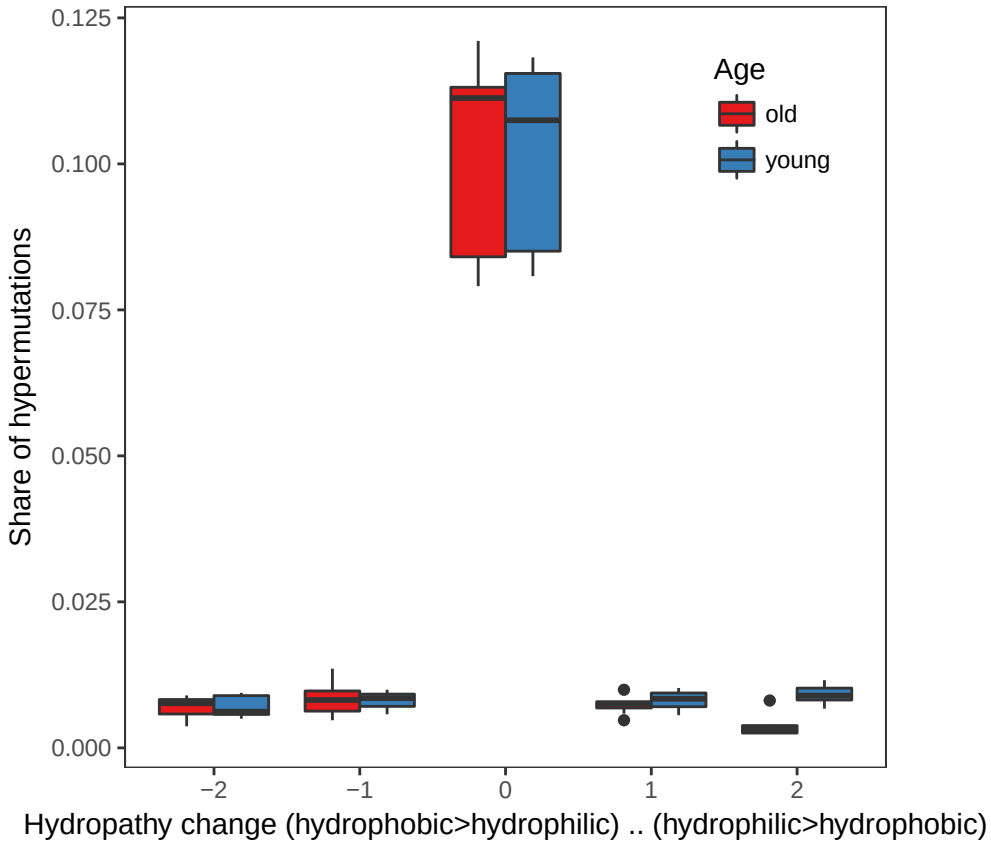
```
dt.p = data.table()

for (ms in unique(df.sign.s$mutation.signature)) {
  tt = t.test(freq ~ proj, df.sign.s %>% filter(mutation.signature == ms))
```

```
  p = tt$p.value
  dt.p = rbind(dt.p,
               data.table(mutation.signature = ms, p=p))
}

dt.p$p.adj = p.adjust(dt.p$p, method = "BH")
print(dt.p %>% arrange(p.adj))
```

```
##    mutation.signature           p      p.adj
## 1                 G>A 0.001634609 0.01961531
## 2                 C>G 0.008651607 0.05190964
## 3                 T>C 0.143563917 0.45932501
## 4                 T>G 0.153108337 0.45932501
## 5                 C>A 0.357554836 0.61295115
## 6                 C>T 0.352831310 0.61295115
## 7                 G>T 0.279190083 0.61295115
## 8                 T>A 0.439025235 0.65853785
## 9                 A>C 0.870625042 0.87062504
## 10                A>G 0.701246870 0.87062504
## 11                A>T 0.863525705 0.87062504
## 12                G>C 0.780921733 0.87062504
```

```
aa.classes = data.table(aa = strsplit("I,V,L,F,C,M,A,W,G,T,S,Y,P,H,N,D,Q,E,K,R", ",")[[1]],
                        hydrop = c(rep("hydrophobic", 8), rep("neutral", 6),
                                   rep("hydrophilic", 6)))

df.aachange = df %>%
  merge(aa.classes %>% mutate(from.aa = aa, from.value = hydrop) %>% select(from.aa, from.value)) %>%
  merge(aa.classes %>% mutate(to.aa = aa, to.value = hydrop) %>% select(to.aa, to.value))
```

```
hydrop_toint = function(x) {
  ifelse(x == "hydrophobic", 1, ifelse(x == "neutral", 0, -1))
}

dt.aachange.s = df.aachange %>%
  group_by(sample, proj, from.value, to.value) %>%
  summarise(count = n()) %>%
  group_by(sample, proj) %>%
  mutate(freq = count / sum(count),
         hydrop.change = hydrop_toint(to.value) - hydrop_toint(from.value)) %>%
  group_by(hydrop.change) %>%
  mutate(freq2 = freq / length(unique(paste(from.value, to.value))))

p9=ggplot(dt.aachange.s, aes(x=hydrop.change, group=paste(hydrop.change, proj), fill = proj, y= freq2))
  geom_boxplot() +
  scale_fill_brewer("Age", palette = "Set1") +
  ylab("Share of hypermutations") +
  xlab("Hydropathy change (hydrophobic>hydrophilic) .. (hydrophilic>hydrophobic)") +
  theme_bw() +
  theme(aspect=1,
        legend.position = c(0.85,0.85),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())
p9
```




```
dt.p = data.table()

for (hc in unique(dt.aachange.s$hydrop.change)) {
  tt = t.test(freq ~ proj, dt.aachange.s %>% filter(hydrop.change == hc))
  p = tt$p.value
  dt.p = rbind(dt.p,
               data.table(hydrop.change = hc, p=p))
}

dt.p$p.adj = p.adjust(dt.p$p, method = "BH")
print(dt.p %>% arrange(p.adj))
```

```
##   hydrop.change           p      p.adj
## 1             2 0.005610143 0.02805071
## 2             1 0.269552193 0.67388048
## 3             0 0.880382343 0.89206698
## 4            -2 0.892066979 0.89206698
## 5            -1 0.851821299 0.89206698
```

```
ggsave("figures/p5.pdf", p5, width = 10, height = 5)
```

```
## Warning in density.default(x, weights = w, bw = bw, adjust = adjust, kernel
## = kernel, : sum(weights) != 1 -- will not get true density

## Warning in density.default(x, weights = w, bw = bw, adjust = adjust, kernel
## = kernel, : sum(weights) != 1 -- will not get true density
```

```r
ggsave("figures/p6.pdf", p6, width = 5, height = 5)
ggsave("figures/p7.pdf", p7, width = 5, height = 5)
ggsave("figures/p8.pdf", p8, width = 8, height = 5)
ggsave("figures/p9.pdf", p9, width = 7, height = 5)
```