

# Phylogenetic analysis of clonal trees

*Anna Obraztsova*

2/16/2017

Diameter is the largest number of mutations between root and leave. Branching is a number of leaves divided by mean length of path from root to leave. Total.freq is a sum of frequencies of all vertices of tree.

```
library(ggplot2)
library(stringr)
library(reshape2)
library(dplyr)
library(ggbeeswarm)

old_rna = c("Abdulain", "Ilgen", "Mamaev", "Smirnov", "Vlasov")
young_rna = c("Antipyat", "Epifancev", "Hadjibekov", "Koshkin", "Kovalchuk")

gs <- data.frame()

for (sample in old_rna){
  .df <- read.table(paste('~/yf/trees/stat/yf_old_RNA/', sample, ".txt", sep = ""), header=T, sep="\t")
  names(.df) <- c('ndn', 'root.freq', 'v', 'j', 'total.freq', 'total.leaves', 'total.nodes', 'root.mut')
  .df$proj <- "old"
  .df$sample <- sample
  gs <- rbind(gs, .df)
}

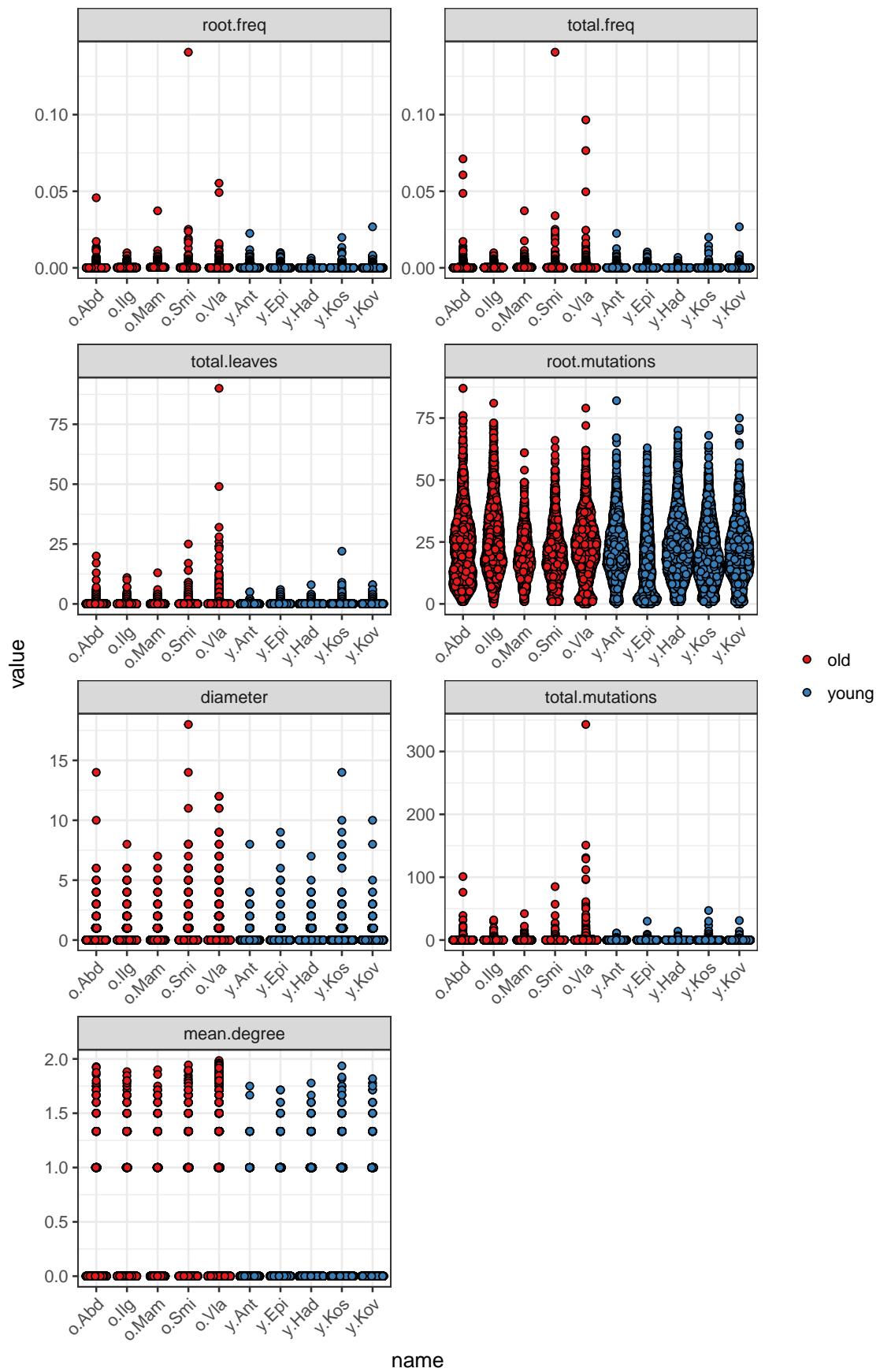
for (sample in young_rna){
  .df <- read.table(paste('~/yf/trees/stat/yf_young_RNA/', sample, ".txt", sep = ""), header=T, sep="")
  names(.df) <- c('ndn', 'root.freq', 'v', 'j', 'total.freq', 'total.leaves', 'total.nodes', 'root.mut')
  .df$proj <- "young"
  .df$sample <- sample
  gs <- rbind(gs, .df)
}

gs <- mutate_each(gs, funs(as.double(.)), -ndn, -proj, -sample, -v, -j)

gs2 <- gs %>% dplyr::select(-j, -v) %>%
  melt(id=c('ndn', 'proj', 'sample')) %>%
  mutate(name=paste(str_sub(proj, 1, 1), str_sub(sample,1, 3), sep='.')) %>%
  filter(variable != 'total.nodes' & variable != 'mean.path')

first = c('root.freq', 'total.freq', 'total.leaves', 'root.mutations', 'diameter', 'total.mutations', 'name')

ggplot(filter(gs2, variable %in% first)) + geom_quasirandom(aes(x = name, y = value, group=proj, fill = facet_wrap(~variable, nrow=4, scales='free') +
  scale_fill_brewer("", palette = "Set1") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```

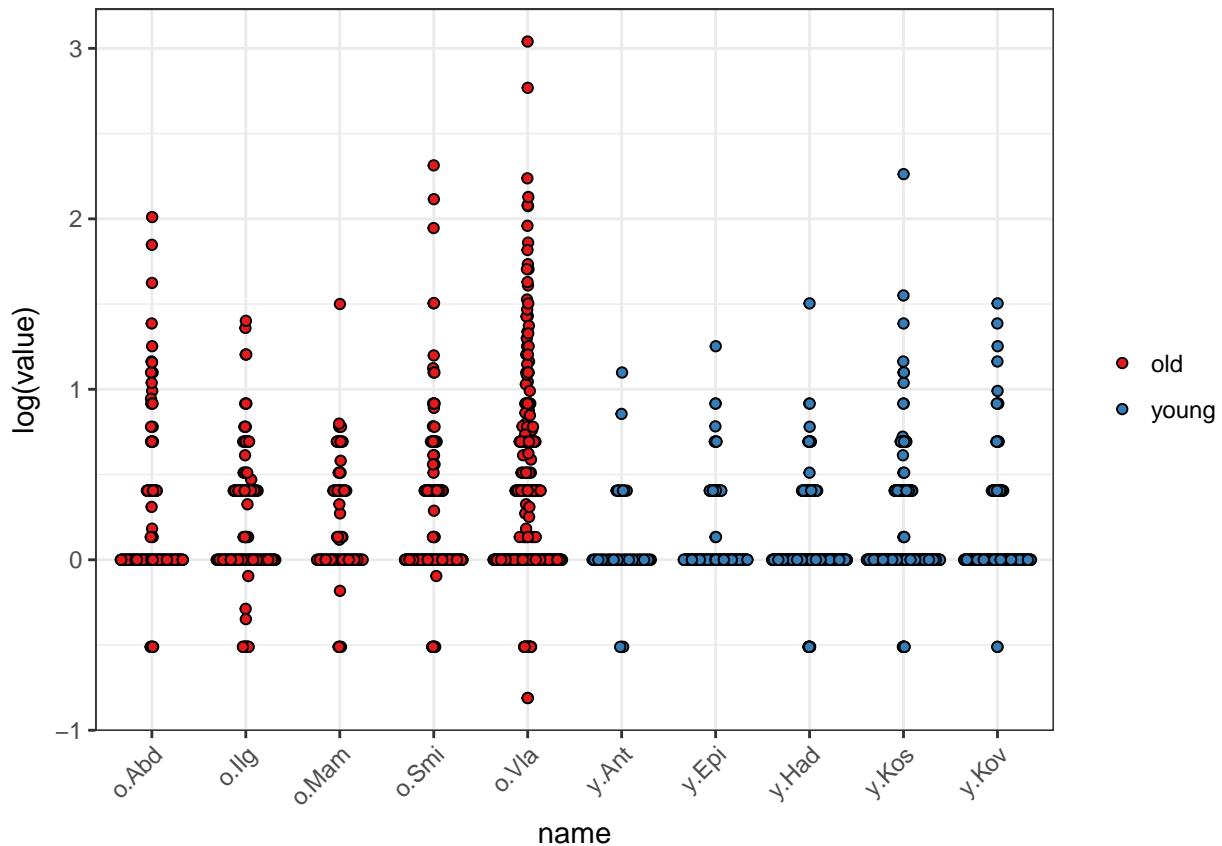
for (i in first){
  cat(i, 't-Test p-value\n', t.test(gs[[i]] ~ gs[['proj']])$p.value*nrow(gs), '\n')
}

## root.freq t-Test p-value
## 318.9466
## total.freq t-Test p-value
## 0.06066829
## total.leaves t-Test p-value
## 8.702628e-12
## root.mutations t-Test p-value
## 2.718511e-28
## diameter t-Test p-value
## 3.074163e-22
## total.mutations t-Test p-value
## 4.541913e-08
## mean.degree t-Test p-value
## 9.298883e-13

ggplot(filter(gs2, variable == 'branching')) + geom_quasirandom(aes(x = name, y = log(value)), group=proj,
scale_fill_brewer("", palette = "Set1") +
theme_bw() +
theme(axis.text.x = element_text(angle = 45, hjust = 1))

## Warning: Removed 16401 rows containing missing values (geom_point).

```



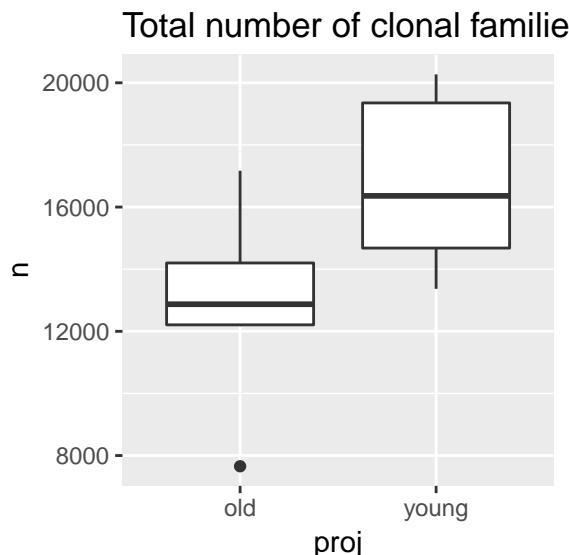
```

gs3 <- gs2 %>% dplyr::group_by(sample, proj) %>% summarise(n = n())
ggplot(gs3) + geom_boxplot(aes(x = proj, y = n)) + ggtitle('Total number of clonal families')

```

```
t.test(n ~ proj, gs3)

##
##  Welch Two Sample t-test
##
## data: n by proj
## t = -1.9578, df = 7.8131, p-value = 0.08682
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -8699.7232 728.5232
## sample estimates:
## mean in group old mean in group young
## 12820.8 16806.4
```

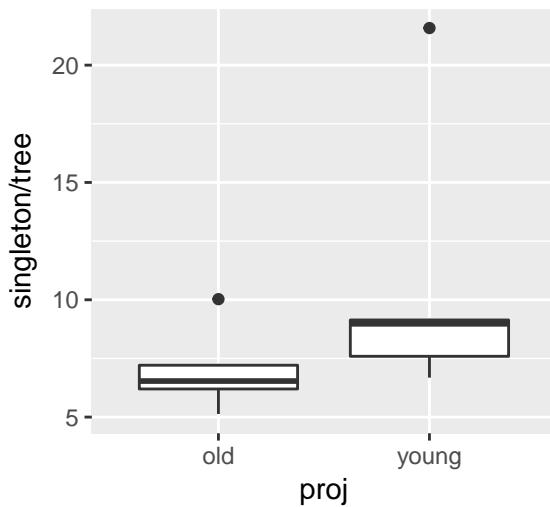


```
gs4 <- gs %>% mutate(single = ifelse(total.leaves == 0, 'singleton', 'tree')) %>%
  dplyr::group_by(sample, proj, single) %>% summarise(n = n()) %>%
  dcast(sample + proj ~ single)
ggplot(gs4, aes(x=proj, y = singleton/tree)) + geom_boxplot() + ggtitle('Single clones number to clonal')

t.test.singleton.tree ~ proj, gs4
```

```
##
##  Welch Two Sample t-test
##
## data: singleton/tree by proj
## t = -1.321, df = 4.7182, p-value = 0.2469
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -11.248738 3.702736
## sample estimates:
## mean in group old mean in group young
## 7.019801 10.792802
```

### Single clones number to clonal

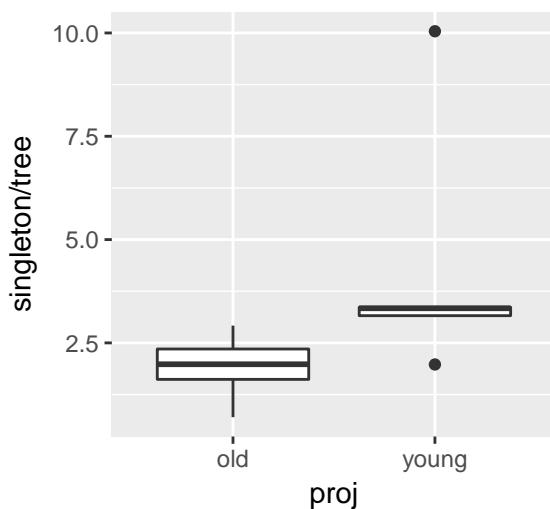


```
gs4.1 <- gs %>% mutate(single = ifelse(total.leaves == 0, 'singleton', 'tree')) %>%
  dplyr::group_by(sample, proj, single) %>% summarise(n = sum(total.freq)) %>%
  dcast(sample + proj ~ single)
ggplot(gs4.1, aes(x=proj, y = singleton/tree)) + geom_boxplot() + ggtitle('Single clones total frequency')

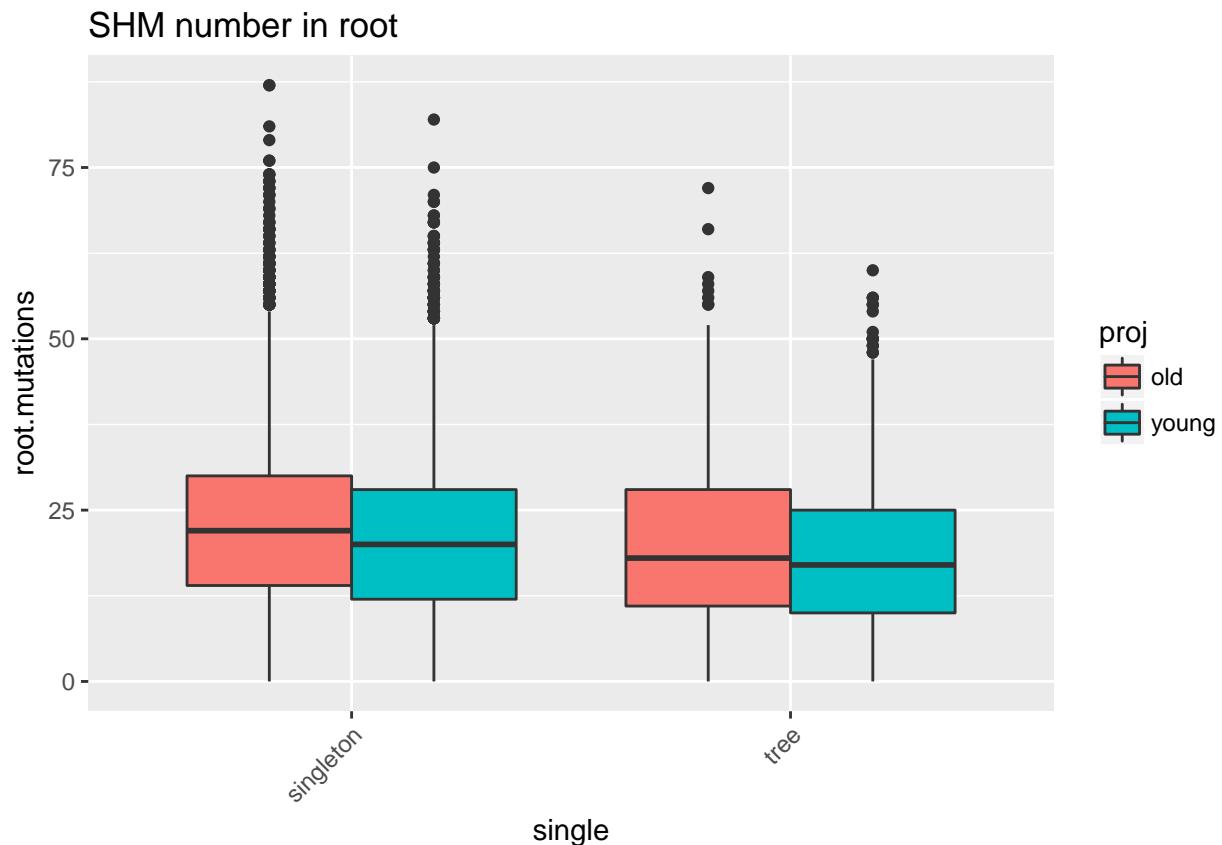
t.test(singleton/tree ~ proj, gs4.1)

## 
##   Welch Two Sample t-test
##
## data: singleton/tree by proj
## t = -1.6556, df = 4.5309, p-value = 0.1647
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -6.404852 1.482364
## sample estimates:
## mean in group old mean in group young
## 1.914684 4.375928
```

### Single clones total frequency 1



```
gs4.2 <- gs %>% mutate(single = ifelse(total.leaves == 0, 'singleton', 'tree')) %>%
  mutate(name=paste(str_sub(proj, 1, 1), str_sub(sample, 1, 3), sep='.'))
ggplot(gs4.2, aes(x = single, y = root.mutations, fill=proj)) + geom_boxplot() + ggtitle('SHM number in root')
theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
a <- aov(root.mutations ~ single * proj, gs4.2)
summary(a)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)						
## single	1	13071	13071	88.355	<2e-16 ***						
## proj	1	22890	22890	154.734	<2e-16 ***						
## single:proj	1	309	309	2.091	0.148						
## Residuals	18513	2738702	148								
## ---											
## Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'.'	0.1	' '	1

```
gs5 <- gs %>% dplyr::group_by(sample, proj, v, j) %>% summarise(total.freq = sum(total.freq))

ggplot(filter(gs5, proj=='old'), aes(x = v, y = j, size=total.freq)) + geom_point() +
  facet_wrap(~sample, ncol=1, scales='free') +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

ggplot(filter(gs5, proj=='young'), aes(x = v, y = j, size=total.freq)) + geom_point() +
  facet_wrap(~sample, ncol=1, scales='free') +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```