Melanoma BCR repertoire analysis

Anna Obraztsova, Mikhail Shugay

6 June 2017

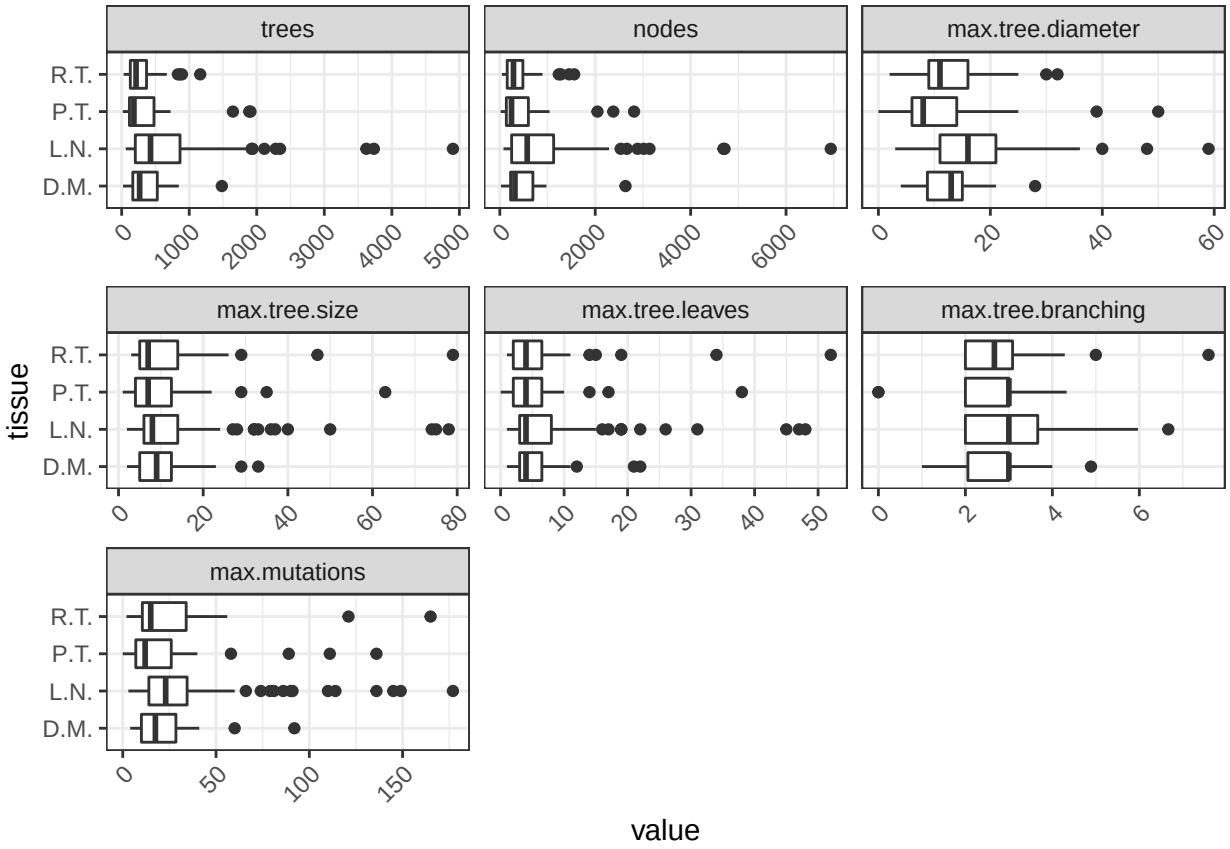
```
library(dplyr)
library(ggplot2)
library(reshape2)
library(plyr)

load('clones.rda')
codes <- read.table('SKCM-codes-samples-for-patients-tss.txt', header=F, sep="\t")
colnames(codes) <- c('patient_id', 'c1', 'c2', 'c3', 'tissue')
codes$tissue <- revalue(codes$tissue, c("Distant Metastasis" = "D.M.",
                                        "Primary Tumor" = "P.T.",
                                        "Regional Cutaneous or Subcutaneous Tissue (includes satellite
codes$tissue = as.factor(ifelse(as.character(codes$tissue) == "", NA, as.character(codes$tissue)))

meta.1 = read.table("full_metadata.txt", header = T, sep ="\t")

meta = meta.1 %>%
  dplyr::select(Code, Dead, OS.corrected, OS.uncorrected, IGH.clonality, IGH.clonality.all, IGH.coverage

colnames(meta) = c("patient_id", "Dead", "OS.corrected", "OS.uncorrected", "IGH.clonality", "IGH.clonali

meta = merge(codes, meta)

clones = merge(clones, meta)
```

```
clones$low_cov = is.na(clones$IGH.clonality)
cc = clones %>%
  group_by(patient_id, tissue, Dead, OS.corrected, OS.uncorrected,
           IGH.clonality, IGH.clonality.all, IGH.coverage.by.MiXCR, Ratio.IGG1.to.IGH.by.MiXCR, low_cov
  dplyr::summarise(trees = n(), max.tree.size = max(nodes),
                   nodes = sum(nodes),
                   max.tree.diameter = max(diameter),
                   max.tree.leaves = max(leaves),
                   max.tree.branching = max(ifelse(is.finite(branching), branching, 0)),
                   max.mutations = max(total.mut)) %>%
  ungroup
```

When filtering low-coverage samples, the difference is in overall number of trees and clonotypes, and tree diameter (in favour of LN of course)

```
cc1 = cc %>%
  filter(!low_cov) %>% # filter all samples with low coverage - cannot estimate clonality, etc there
  dplyr::select(tissue, patient_id, trees, nodes,
                max.tree.diameter, max.tree.size,
                max.tree.leaves, max.tree.branching, max.mutations) %>%
  melt(id.vars = c('tissue', 'patient_id'))

ggplot(cc1, aes(x = tissue, y = value)) +
  geom_boxplot() + coord_flip() +
```

```
  facet_wrap(~variable, scales="free_x") +
  theme_bw() + #scale_y_log10() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
for (v in unique(cc1$variable)) {
  print(v)
  cc.sub = subset(cc1, variable == v)
  a1 = aov(value~tissue,cc.sub)
  print(summary(a1))
  print(TukeyHSD(a1))
}
```

```
## [1] "trees"
##              Df   Sum Sq Mean Sq F value  Pr(>F)
## tissue        3  5375122 1791707   4.939 0.00241 **
## Residuals   237 85975514  362766
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = value ~ tissue, data = cc.sub)
##
## $tissue
##                 diff         lwr       upr     p adj
## L.N.-D.M.  307.61033    3.618825 611.60184 0.0461357
## P.T.-D.M.   49.63206 -343.087953 442.35207 0.9879017
```

```
## R.T.-D.M.  -35.46472 -428.184727 357.25529 0.9954981
## P.T.-L.N. -257.97828 -565.970021  50.01347 0.1354752
## R.T.-L.N. -343.07505 -651.066795 -35.08330 0.0222574
## R.T.-P.T.  -85.09677 -480.921338 310.72779 0.9447773
##
## [1] "nodes"
##              Df    Sum Sq Mean Sq F value  Pr(>F)
## tissue        3   8744117 2914706   4.346 0.00529 **
## Residuals   237 158959300  670714
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = value ~ tissue, data = cc.sub)
##
## $tissue
##                 diff       lwr       upr     p adj
## L.N.-D.M.  390.74043  -22.6087 804.08956 0.0714403
## P.T.-D.M.   51.32964 -482.6671 585.32638 0.9945855
## R.T.-D.M.  -41.54133 -575.5381 492.45541 0.9971067
## P.T.-L.N. -339.41080 -758.1992  79.37762 0.1570710
## R.T.-L.N. -432.28176 -851.0702 -13.49335 0.0401273
## R.T.-P.T.  -92.87097 -631.0891 445.34716 0.9702550
##
## [1] "max.tree.diameter"
##              Df Sum Sq Mean Sq F value  Pr(>F)
## tissue        3   1248   416.1   5.372 0.00135 **
## Residuals   237  18359    77.5
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = value ~ tissue, data = cc.sub)
##
## $tissue
##                 diff         lwr        upr     p adj
## L.N.-D.M.  4.3913690 -0.05084913  8.8335872 0.0539705
## P.T.-D.M. -1.0433468 -6.78215173  4.6954582 0.9654923
## R.T.-D.M.  0.4082661 -5.33053882  6.1470711 0.9977817
## P.T.-L.N. -5.4347158 -9.93538937 -0.9340423 0.0107286
## R.T.-L.N. -3.9831029 -8.48377646  0.5175706 0.1032956
## R.T.-P.T.  1.4516129 -4.33255880  7.2357846 0.9157084
##
## [1] "max.tree.size"
##              Df Sum Sq Mean Sq F value Pr(>F)
## tissue        3    190   63.29    0.42  0.739
## Residuals   237  35695  150.61
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = value ~ tissue, data = cc.sub)
##
```

```
## $tissue
##                 diff       lwr       upr     p adj
## L.N.-D.M.  2.0318878 -4.162195  8.225970 0.8310165
## P.T.-D.M.  0.5625000 -7.439500  8.564500 0.9978593
## R.T.-D.M.  2.8528226 -5.149178 10.854823 0.7928830
## P.T.-L.N. -1.4693878 -7.744979  4.806203 0.9301596
## R.T.-L.N.  0.8209348 -5.454656  7.096526 0.9866216
## R.T.-P.T.  2.2903226 -5.774936 10.355581 0.8830255
##
## [1] "max.tree.leaves"
##              Df Sum Sq Mean Sq F value Pr(>F)
## tissue        3     92   30.79   0.516  0.672
## Residuals   237  14145   59.68
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = value ~ tissue, data = cc.sub)
##
## $tissue
##                 diff       lwr      upr     p adj
## L.N.-D.M.  0.9955357 -2.903680 4.894752 0.9117227
## P.T.-D.M. -0.1381048 -5.175417 4.899207 0.9998716
## R.T.-D.M.  1.9264113 -3.110901 6.963723 0.7555822
## P.T.-L.N. -1.1336406 -5.084166 2.816885 0.8797949
## R.T.-L.N.  0.9308756 -3.019650 4.881401 0.9289404
## R.T.-P.T.  2.0645161 -3.012617 7.141650 0.7189034
##
## [1] "max.tree.branching"
##              Df Sum Sq Mean Sq F value Pr(>F)
## tissue        3   5.97  1.9896   2.317 0.0763 .
## Residuals   237 203.52  0.8587
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = value ~ tissue, data = cc.sub)
##
## $tissue
##                  diff        lwr        upr     p adj
## L.N.-D.M.  0.21820463 -0.2495029 0.68591216 0.6229459
## P.T.-D.M. -0.23454338 -0.8387646 0.36967779 0.7470201
## R.T.-D.M.  0.03332593 -0.5708952 0.63754710 0.9989609
## P.T.-L.N. -0.45274802 -0.9266101 0.02111409 0.0669742
## R.T.-L.N. -0.18487870 -0.6587408 0.28898341 0.7440503
## R.T.-P.T.  0.26786931 -0.3411284 0.87686701 0.6663636
##
## [1] "max.mutations"
##              Df Sum Sq Mean Sq F value Pr(>F)
## tissue        3   2120   706.5   0.839  0.474
## Residuals   237 199610   842.2
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
```

```
## Fit: aov(formula = value ~ tissue, data = cc.sub)
##
## $tissue
##                diff        lwr      upr     p adj
## L.N.-D.M.  7.614583  -7.032975 22.26214 0.5352103
## P.T.-D.M.  1.507056 -17.415806 20.42992 0.9968977
## R.T.-D.M.  5.410282 -13.512580 24.33314 0.8809328
## P.T.-L.N. -6.107527 -20.947834  8.73278 0.7112645
## R.T.-L.N. -2.204301 -17.044608 12.63601 0.9806571
## R.T.-P.T.  3.903226 -15.169227 22.97568 0.9518490
```

Survival analysis

```
library(party)
```

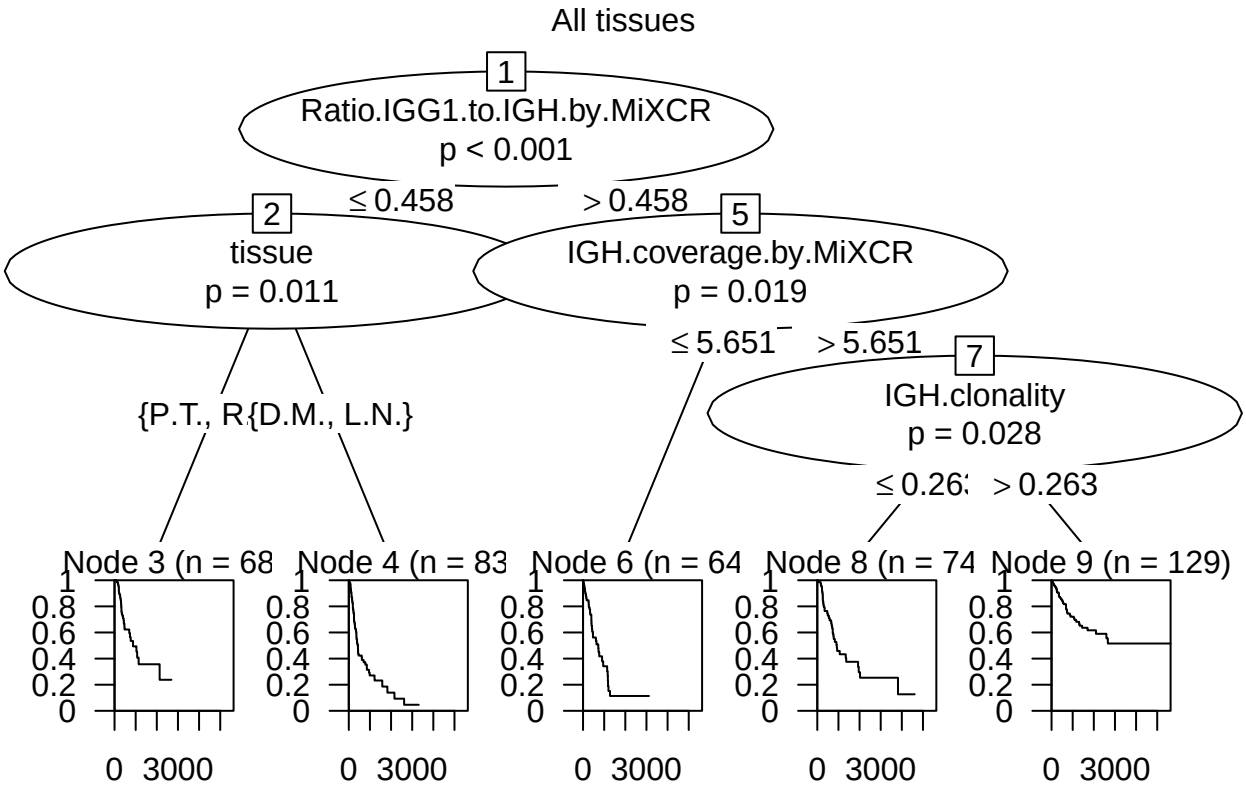
```
## Loading required package: grid
## Loading required package: mvtnorm
## Loading required package: modeltools
## Loading required package: stats4
##
## Attaching package: 'modeltools'
## The following object is masked from 'package:plyr':
##
##     empty
## Loading required package: strucchange
## Loading required package: zoo
##
## Attaching package: 'zoo'
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
## Loading required package: sandwich
```

```
library(survival)

ctree_opts = ctree_control(testtype = "Univariate", minbucket = 45)#mincriterion = 0.9)

make_tree = function(formula, df, ...) {
  dfct = ctree(formula, data = df %>% filter(!is.na(OS.corrected)), controls = ctree_opts)
  print(dfct)
  plot(dfct, ...)
}
```

Survival analysis using conventional MiXCR-derived metrics

```
make_tree(Surv(OS.corrected, Dead) ~ tissue +
            IGH.clonality + IGH.coverage.by.MiXCR + Ratio.IGG1.to.IGH.by.MiXCR,
            cc, "All tissues")
```

```
##
##   Conditional inference tree with 5 terminal nodes
##
## Response:  Surv(OS.corrected, Dead)
## Inputs:  tissue, IGH.clonality, IGH.coverage.by.MiXCR, Ratio.IGG1.to.IGH.by.MiXCR
## Number of observations:  418
##
## 1) Ratio.IGG1.to.IGH.by.MiXCR <= 0.4583333; criterion = 1, statistic = 17.797
##   2) tissue == {P.T., R.T.}; criterion = 0.989, statistic = 11.062
##     3)*  weights = 68
##   2) tissue == {D.M., L.N.}
##     4)*  weights = 83
## 1) Ratio.IGG1.to.IGH.by.MiXCR > 0.4583333
##   5) IGH.coverage.by.MiXCR <= 5.650923; criterion = 0.981, statistic = 5.538
##     6)*  weights = 64
##   5) IGH.coverage.by.MiXCR > 5.650923
##     7) IGH.clonality <= 0.2625757; criterion = 0.972, statistic = 4.854
##       8)*  weights = 74
##     7) IGH.clonality > 0.2625757
##       9)*  weights = 129
```

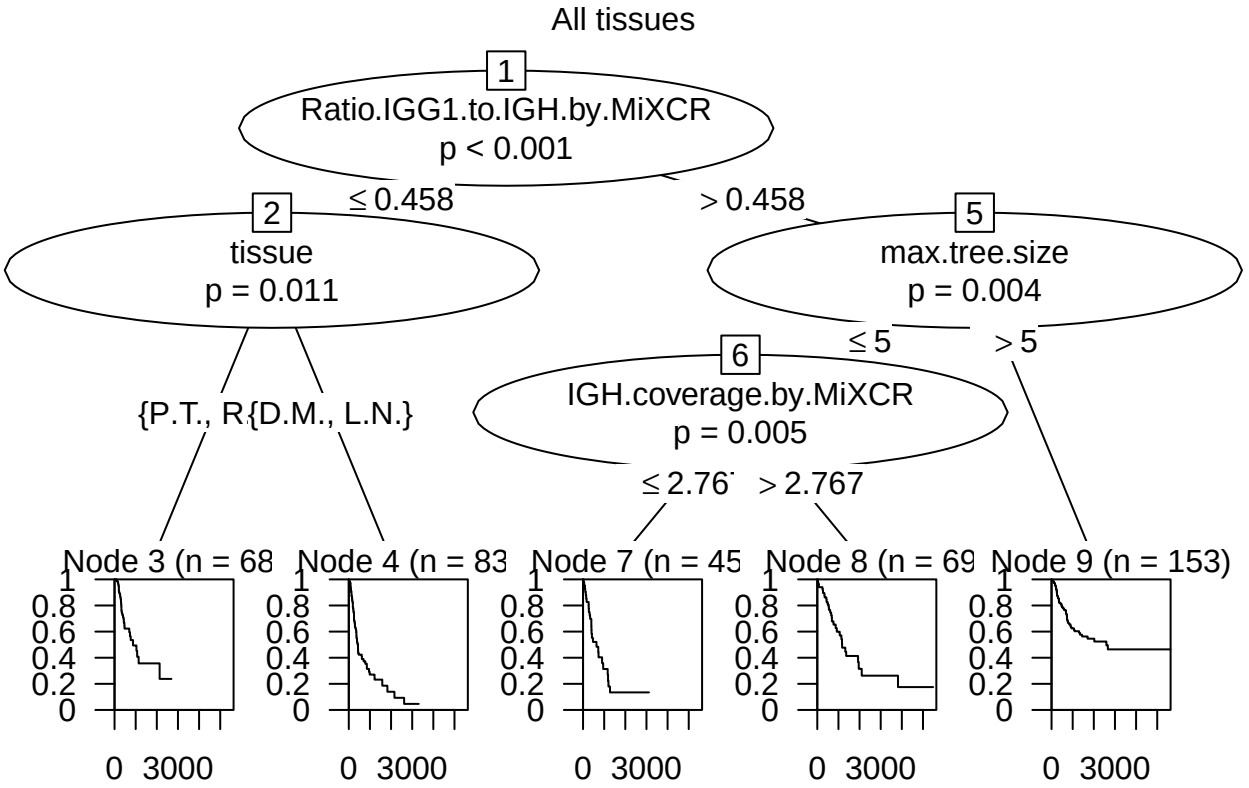


Survival analysis for different tissues including tree-based parameters

```
make_tree(Surv(OS.corrected, Dead) ~ tissue +
            IGH.clonality + IGH.coverage.by.MiXCR + Ratio.IGG1.to.IGH.by.MiXCR +
            max.tree.diameter + max.tree.size +
            max.tree.leaves + max.tree.branching + max.mutations,
            cc, "All tissues")
```

```
##
```

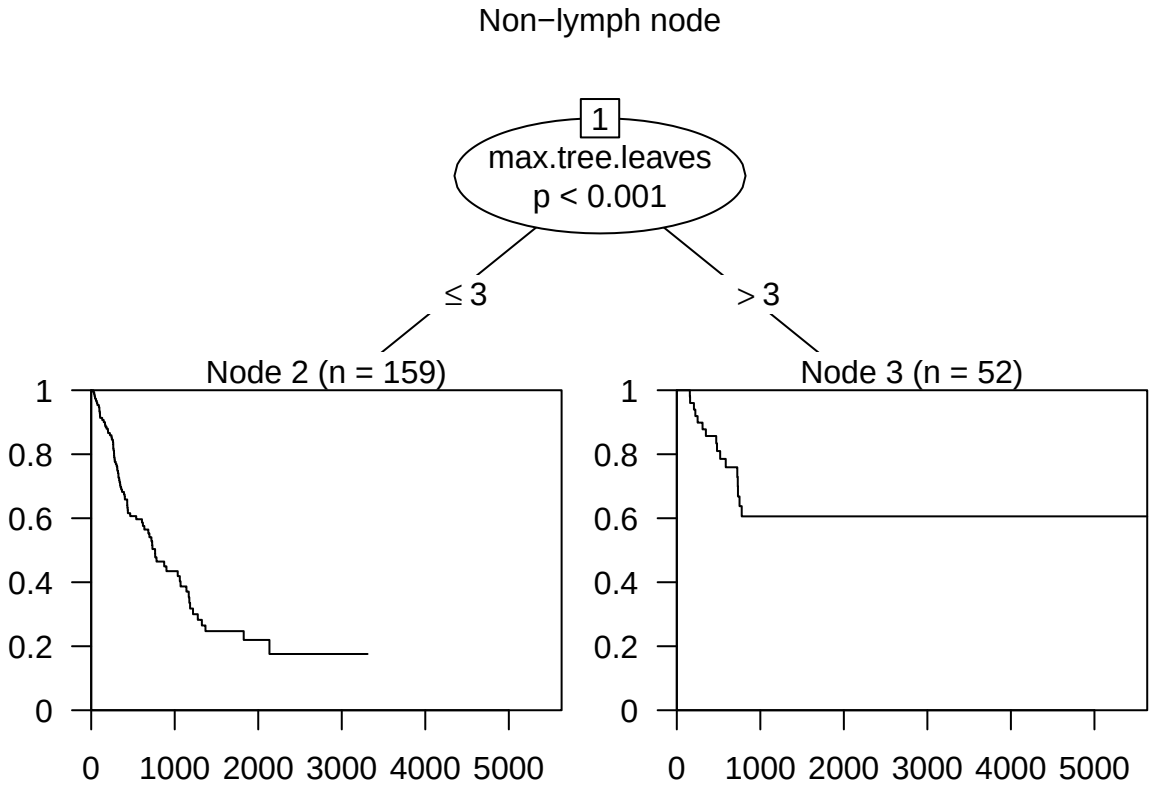
```
##   Conditional inference tree with 5 terminal nodes
##
## Response:  Surv(OS.corrected, Dead)
## Inputs:  tissue, IGH.clonality, IGH.coverage.by.MiXCR, Ratio.IGG1.to.IGH.by.MiXCR, max.tree.diameter
## Number of observations:  418
##
## 1) Ratio.IGG1.to.IGH.by.MiXCR <= 0.4583333; criterion = 1, statistic = 17.797
##   2) tissue == {P.T., R.T.}; criterion = 0.989, statistic = 11.062
##     3)*  weights = 68
##   2) tissue == {D.M., L.N.}
##     4)*  weights = 83
## 1) Ratio.IGG1.to.IGH.by.MiXCR > 0.4583333
##   5) max.tree.size <= 5; criterion = 0.996, statistic = 8.249
##     6) IGH.coverage.by.MiXCR <= 2.766602; criterion = 0.995, statistic = 13.949
##       7)*  weights = 45
##     6) IGH.coverage.by.MiXCR > 2.766602
##       8)*  weights = 69
##   5) max.tree.size > 5
##     9)*  weights = 153
```




```
make_tree(Surv(OS.corrected, Dead) ~
            IGH.clonality + IGH.coverage.by.MiXCR + Ratio.IGG1.to.IGH.by.MiXCR +
            max.tree.diameter + max.tree.size +
            max.tree.leaves + max.tree.branching + max.mutations,
            cc %>% filter(tissue != "L.N."), "Non-lymph node")
```

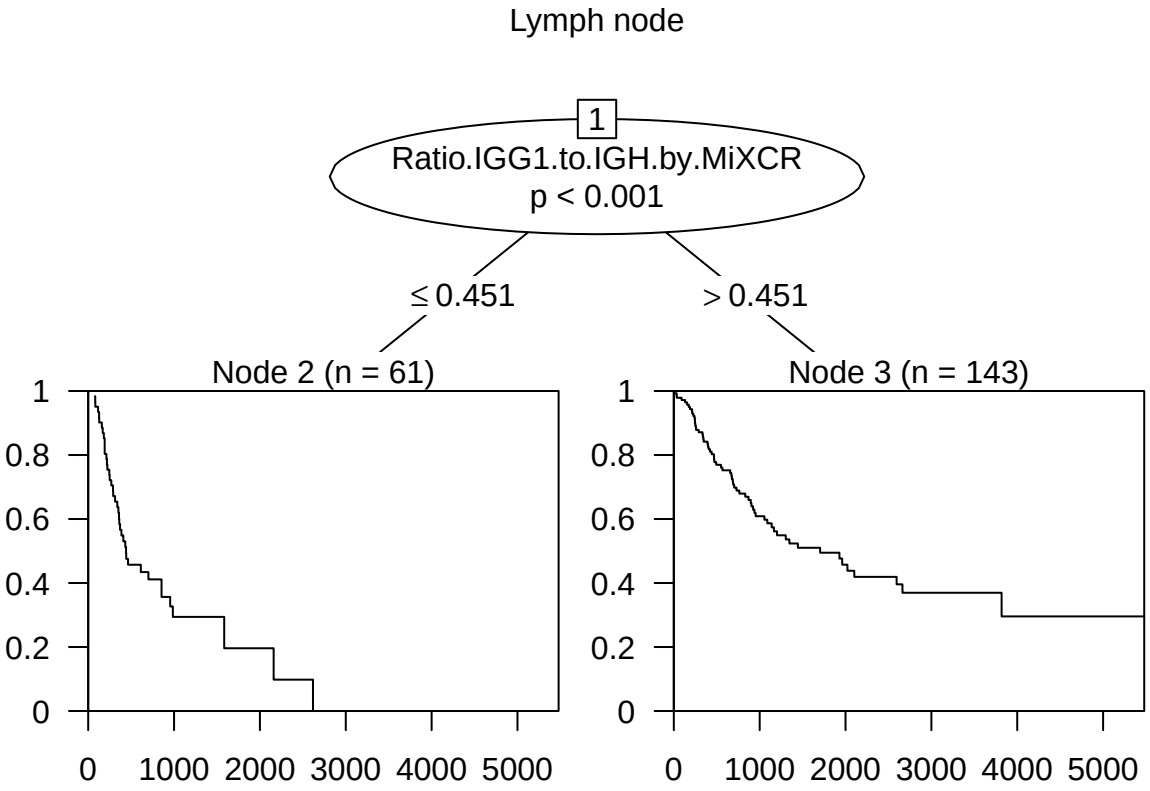
```
##
##   Conditional inference tree with 2 terminal nodes
##
```

```
## Response:  Surv(OS.corrected, Dead)
## Inputs:  IGH.clonality, IGH.coverage.by.MiXCR, Ratio.IGG1.to.IGH.by.MiXCR, max.tree.diameter, max.tr
## Number of observations:  211
##
## 1) max.tree.leaves <= 3; criterion = 0.999, statistic = 10.892
##   2)*  weights = 159
## 1) max.tree.leaves > 3
##   3)*  weights = 52
```




```
make_tree(Surv(OS.corrected, Dead) ~
            IGH.clonality + IGH.coverage.by.MiXCR + Ratio.IGG1.to.IGH.by.MiXCR +
            max.tree.diameter + max.tree.size +
            max.tree.leaves + max.tree.branching + max.mutations,
            cc %>% filter(tissue == "L.N."), "Lymph node")
```

```
##
##   Conditional inference tree with 2 terminal nodes
##
## Response:  Surv(OS.corrected, Dead)
## Inputs:  IGH.clonality, IGH.coverage.by.MiXCR, Ratio.IGG1.to.IGH.by.MiXCR, max.tree.diameter, max.tr
## Number of observations:  204
##
## 1) Ratio.IGG1.to.IGH.by.MiXCR <= 0.4508876; criterion = 1, statistic = 13.491
##   2)*  weights = 61
## 1) Ratio.IGG1.to.IGH.by.MiXCR > 0.4508876
##   3)*  weights = 143
```



R:S ratio

> TODO: merge with trees, stop codon to stop codon mutations.. - where are they from, perhaps should filter all non-functional

```
library(tidyr)
```

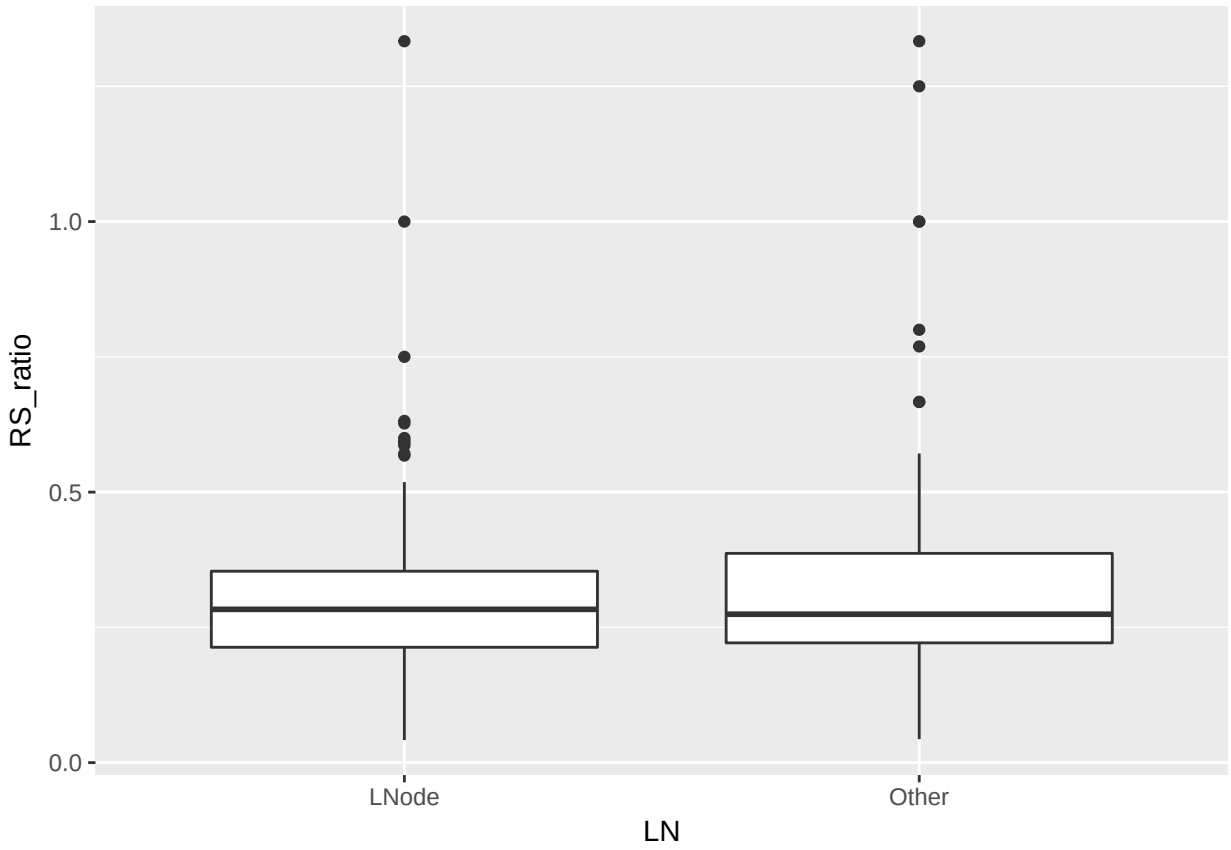
```
##
## Attaching package: 'tidyr'
```

```
## The following object is masked from 'package:reshape2':
##
##     smiths
```

```
library(reshape2)

load('shm.rda')

s1 <- s %>% group_by(patient_id, LN, mut.type) %>%
  dplyr::summarise(n = n()) %>%
  dcast(patient_id + LN ~ mut.type, value.var = "n") %>%
  mutate(RS_ratio = R/S)

s1 = merge(s1, meta)
```

```
ggplot(s1, aes(x = LN, y = RS_ratio)) + geom_boxplot()
```

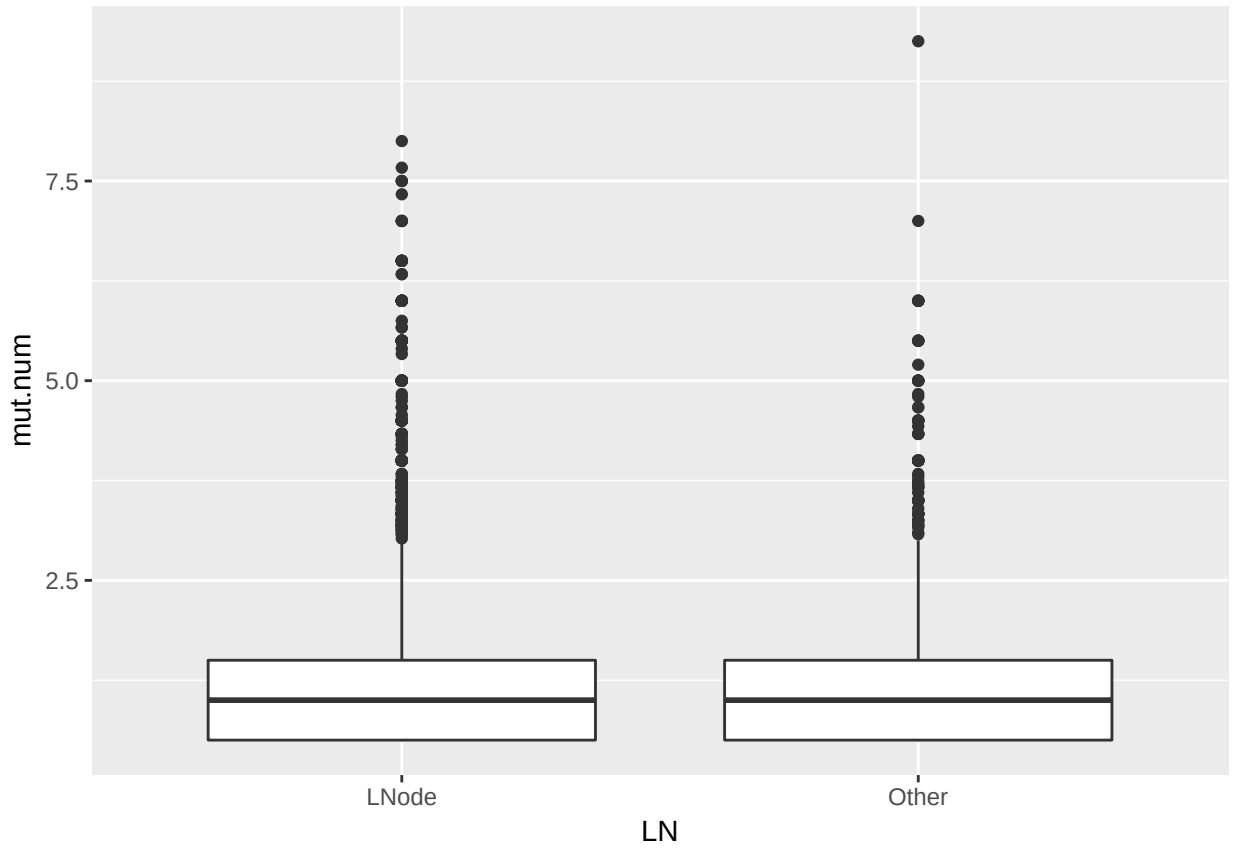
```
## Warning: Removed 54 rows containing non-finite values (stat_boxplot).
```




```
print(t.test(s1$RS_ratio ~ s1$LN)$p.value)
```

```
## [1] 0.2063957
```

```
s2 <- s %>% group_by(patient_id, LN, tree.id, child) %>%
  dplyr::summarise(mut.num = n()) %>%
  group_by(patient_id, LN, tree.id) %>%
  dplyr::summarise(mut.num = sum(mut.num)/(n()+1))

ggplot(s2, aes(x = LN, y = mut.num)) + geom_boxplot()
```




```
print(t.test(s2$mut.num ~ s2$LN)$p.value)
```

```
## [1] 0.1951248
```