

LCN: Lichen interaction network study

MK Lau

Results

```
### REML

### We know from Lamit's dissertation work that lichen communities are
### heritable, largely driven by bark roughness
### Do we find similar patterns?

## Create a list to generate a results table
h2.tab <- matrix("", 8, 4)
colnames(h2.tab) <- c("Response", "H2", "R2", "p-value")

## Total cover ~ genotype
ptc.reml <- lme4::lmer(I(PC^(1/2)) ~ (1 | geno),
                      data = onc.dat, REML = TRUE)
ptc.reml.pval <- RLRsim::exactRLRT(ptc.reml)
ptc.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 2.9627, p-value = 0.038
fligner.test(onc.dat$PC^(1/2), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: onc.dat$PC^(1/2) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 13.751, df = 12, p-value =
## 0.3169
shapiro.test(residuals(ptc.reml))

##
## Shapiro-Wilk normality test
##
## data: residuals(ptc.reml)
## W = 0.95096, p-value = 0.02174
h2.tab[1, "p-value"] <- ptc.reml.pval$"p.value"
h2.tab[1, "H2"] <- H2(ptc.reml, g = onc.dat$geno)
h2.tab[1, "R2"] <- R2(ptc.reml)
R2(ptc.reml)

## R2c
## 0.1727875
```

```

h2.tab[1, "Response"] <- "Percent Lichen Cover"

## Species richness ~ genotype
spr.reml <- lme4::lmer(I(SR^(1/2)) ~ (1 | geno),
                     data = onc.dat, REML = TRUE)
spr.reml.pval <- RLRsim::exactRLRT(spr.reml)
spr.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 1.0001, p-value = 0.1423
shapiro.test(residuals(spr.reml))

##
## Shapiro-Wilk normality test
##
## data: residuals(spr.reml)
## W = 0.97364, p-value = 0.2467
fligner.test(onc.dat$SR^(1/2), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: onc.dat$SR^(1/2) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 13.276, df = 12, p-value =
## 0.3493
h2.tab[2, "p-value"] <- spr.reml.pval$"p.value"
h2.tab[2, "H2"] <- H2(spr.reml, g = onc.dat$geno)
h2.tab[2, "R2"] <- R2(spr.reml)
R2(spr.reml)

## R2c
## 0.09814791
h2.tab[2, "Response"] <- "Lichen Species Richness"

## Bark roughness REML
prb.reml <- lme4::lmer(I(BR^(1/2)) ~ (1 | geno), data = onc.dat, REML = TRUE)
prb.reml.pval <- RLRsim::exactRLRT(prb.reml)
prb.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 10.69, p-value = 2e-04

```

```

fligner.test(onc.dat$BR^(1/2), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data:  onc.dat$BR^(1/2) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 6.1915, df = 12, p-value =
## 0.9061

shapiro.test(residuals(prb.reml))

##
## Shapiro-Wilk normality test
##
## data:  residuals(prb.reml)
## W = 0.97975, p-value = 0.4529

h2.tab[3, "p-value"] <- prb.reml.pval$"p.value"
h2.tab[3, "H2"] <- H2(prb.reml, g = onc.dat$geno)
h2.tab[3, "R2"] <- R2(prb.reml)
R2(prb.reml)

##      R2c
## 0.3783496

h2.tab[3, "Response"] <- "Percent Rough Bark"

## pH ~ genotype
ph.reml <- lme4::lmer(I(log(pH)) ~ (1 | geno),
                    data = na.omit(onc.dat), REML = TRUE)
ph.reml.pval <- RLRsim::exactRLRT(ph.reml)
ph.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.50841, p-value = 0.2095

fligner.test(log(onc.dat$pH), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data:  log(onc.dat$pH) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 22.971, df = 12, p-value =
## 0.02797

shapiro.test(residuals(ph.reml))

##
## Shapiro-Wilk normality test
##
## data:  residuals(ph.reml)
## W = 0.80466, p-value = 6.35e-07

```

```

# h2.tab[1, "p-value"] <- ph.reml.pval$"p.value"
# h2.tab[1, "H2"] <- H2(ph.reml, g = onc.dat$geno)
# h2.tab[1, "R2"] <- R2(ph.reml)
R2(ph.reml)

##          R2c
## 0.1272677

# h2.tab[1, "Response"] <- "Percent Lichen Cover"

## condensed tannins REML
ct.reml <- lme4::lmer(I(CT^(1/4)) ~ (1 | geno), data = onc.dat, REML = TRUE)
ct.reml.pval <- RLRsim::exactRLRT(ct.reml)
ct.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 4.3224, p-value = 0.0153
fligner.test(onc.dat$CT^(1/4), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: onc.dat$CT^(1/4) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 7.8941, df = 12, p-value =
## 0.7933
shapiro.test(residuals(ct.reml))

##
## Shapiro-Wilk normality test
##
## data: residuals(ct.reml)
## W = 0.74892, p-value = 2.431e-08

## CN ratio REML
cnr.reml <- lme4::lmer(I(CN^(1/1)) ~ (1 | geno), data = onc.dat, REML = TRUE)

## boundary (singular) fit: see ?isSingular
cnr.reml.pval <- RLRsim::exactRLRT(cnr.reml)
cnr.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0, p-value = 1

```

```

fligner.test(onc.dat$CN^(1/1), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data:  onc.dat$CN^(1/1) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 8.1116, df = 12, p-value =
## 0.7763

shapiro.test(residuals(cnr.reml))

##
## Shapiro-Wilk normality test
##
## data:  residuals(cnr.reml)
## W = 0.92183, p-value = 0.001754

## Bark roughness PCA

##### This is a rough draft of chem data analysis with new pH #####
pca.onc <- princomp(na.omit(onc.dat[, c("pH", "CT", "CN"))))
cumsum(pca.onc[["sdev"]] / sum(pca.onc[["sdev"]]))

##      Comp.1      Comp.2      Comp.3
## 0.7652602 0.9986463 1.0000000

tpc.onc <- pca.onc[["scores"]][, 1:2]
onc.dat.test <- cbind(onc.dat,
                      tpc.onc[match(rownames(onc.dat), rownames(tpc.onc)), ])

pc1.reml <- lme4::lmer(I(Comp.1^(1/1)) ~ (1 | geno),
                      data = onc.dat.test, REML = TRUE)
RLRsim::exactRLRT(pc1.reml)

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.68862, p-value = 0.1824

pc2.reml <- lme4::lmer(I(Comp.2^(1/1)) ~ (1 | geno),
                      data = onc.dat.test, REML = TRUE)
RLRsim::exactRLRT(pc2.reml)

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.22788, p-value = 0.2815

cn.d.onc.test <- distNet(cn.onc[as.character(onc.dat.test[!is.na(onc.dat.test[, "Comp.1"])]), "tree.id"))
adonis2(cn.d.onc.test ~ Comp.1 * Comp.2, data = onc.dat.test)

```

```

## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 999
##
## adonis2(formula = cn.d.onc.test ~ Comp.1 * Comp.2, data = onc.dat.test)
##           Df SumOfSqs      R2      F Pr(>F)
## Comp.1      1    26.78 0.01962 1.0763  0.287
## Comp.2      1    10.07 0.00738 0.4049  0.561
## Comp.1:Comp.2 1   108.89 0.07978 4.3767  0.036 *
## Residual    49  1219.08 0.89322
## Total      52  1364.81 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

mantel(cn.d.onc.test ~ dist(na.omit(onc.dat[, c("pH", "CN", "CT")]])))

##      mantelr      pval1      pval2      pval3  llim.2.5%  ulim.97.5%
## 0.15698039 0.08400000 0.91700000 0.08400000 -0.07275217 0.25386171

## Is species richness correlated with percent cover?
cor.test(onc.dat[, "SR"], onc.dat[, "PC"], data = onc.dat)

##
## Pearson's product-moment correlation
##
## data:  onc.dat[, "SR"] and onc.dat[, "PC"]
## t = 8.3456, df = 55, p-value = 2.393e-11
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6047186 0.8437321
## sample estimates:
##      cor
## 0.7475023

## Were these correlated with bark roughness?
ptc.prb.lm <- lm(I(PC^(1/2)) ~ I(BR^(1/2)), data = onc.dat)
summary(ptc.prb.lm)

##
## Call:
## lm(formula = I(PC^(1/2)) ~ I(BR^(1/2)), data = onc.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.9770 -1.6378  0.6333  1.9603  3.4658
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.4142     1.0901   4.049 0.000162 ***
## I(BR^(1/2))    0.4942     0.1896   2.607 0.011730 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.485 on 55 degrees of freedom
## Multiple R-squared:  0.11, Adjusted R-squared:  0.09381

```

```
## F-statistic: 6.797 on 1 and 55 DF,  p-value: 0.01173
fligner.test(onc.dat$PC, onc.dat$BR)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  onc.dat$PC and onc.dat$BR
## Fligner-Killeen:med chi-squared = 27.401, df = 24, p-value =
## 0.2861
shapiro.test(residuals(ptc.prb.lm))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(ptc.prb.lm)
## W = 0.95045, p-value = 0.02061
spr.prb.lm <- lm(I(SR^(1)) ~ I(BR^(1/2)), data = onc.dat)
summary(spr.prb.lm)

##
## Call:
## lm(formula = I(SR^(1)) ~ I(BR^(1/2)), data = onc.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.0420 -1.3123 -0.1178  1.2308  4.3519
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.5015     0.8002   3.126  0.00283 **
## I(BR^(1/2))   0.1709     0.1392   1.228  0.22456
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.824 on 55 degrees of freedom
## Multiple R-squared:  0.0267, Adjusted R-squared:  0.009003
## F-statistic: 1.509 on 1 and 55 DF,  p-value: 0.2246
fligner.test(onc.dat$SR^(1), onc.dat$BR)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  onc.dat$SR^(1) and onc.dat$BR
## Fligner-Killeen:med chi-squared = 26.046, df = 24, p-value =
## 0.3508
shapiro.test(residuals(spr.prb.lm))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(spr.prb.lm)
## W = 0.97168, p-value = 0.2008
```

```

## COM ~ genotype + Bark roughness + PTC + SPR
set.seed(2)
rcom.ng.perm <- vegan::adonis2(onc.com.rel^(1/1) ~ BR + PC + SR,
                              data = onc.dat, perm = 10000, mrank = TRUE)

set.seed(2)
rcom.perm <- vegan::adonis2(onc.com.rel^(1/1) ~ geno + BR + PC + SR,
                           data = onc.dat, perm = 10000, mrank = TRUE)

set.seed(2)
com.ng.perm <- vegan::adonis2(onc.com^(1/1) ~ BR + PC + SR,
                              data = onc.dat, perm = 10000, mrank = TRUE)

set.seed(2)
com.perm <- vegan::adonis2(onc.com^(1/1) ~ geno + BR + PC + SR,
                           data = onc.dat, perm = 10000, mrank = TRUE)

rcom.ng.perm

## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 10000
##
##      Df SumOfSqs      R2      F    Pr(>F)
## BR      1   0.4398 0.03889  3.7408 0.008799 **
## PC      1   3.8618 0.34151 32.8482 9.999e-05 ***
## SR      1   0.7754 0.06857  6.5958 9.999e-05 ***
## Residual 53   6.2309 0.55102
## Total   56  11.3079 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

rcom.perm

## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 10000
##
##      Df SumOfSqs      R2      F    Pr(>F)
## geno   12   2.7463 0.24287  1.8221 0.0031997 **
## BR      1   0.1248 0.01104  0.9938 0.3900610
## PC      1   2.6711 0.23622 21.2661 9.999e-05 ***
## SR      1   0.6159 0.05447  4.9036 0.0009999 ***
## Residual 41   5.1498 0.45541
## Total   56  11.3079 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

h2.tab[4, "p-value"] <- unlist(rcom.perm)["Pr(>F)1"]
h2.tab[4, "H2"] <- H2(rcom.perm, g = onc.dat$geno)
h2.tab[4, "R2"] <- R2(rcom.perm)
h2.tab[4, "Response"] <- "Lichen Community Composition"

## Is network similarity correlated with community composition?
ecodist::mantel(cn.d.onc ~ vegdist(onc.com.rel), mrank = TRUE)

```



```
##      mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## 0.09198784 0.07200000 0.92900000 0.12000000 0.05120132 0.13656424
```

```
spr.d <- dist(onc.dat$SR)
ptc.d <- dist(onc.dat$PC)
prb.d <- dist(onc.dat$BR)
### rough -> cover -> rich -> net
ecodist::mantel(cn.d.onc ~ vegdist(onc.com.rel) + spr.d + ptc.d + prb.d, mrank = TRUE)
```

```
##      mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## 0.06853395 0.15400000 0.84700000 0.31300000 0.02256902 0.13046001
```

```
## Partial Mantels using RFLP distance
ecodist::mantel(cn.mu.d.onc ~ rflp.d)
```

```
##      mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## -0.00603936 0.54500000 0.45600000 0.96700000 -0.15782909 0.18127044
```

```
ecodist::mantel(onc.com.mu.d ~ rflp.d)
```

```
##      mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## 0.1179051 0.2830000 0.7180000 0.4830000 -0.2789494 0.2435282
```

```
ecodist::mantel(cn.mu.d.onc ~ onc.com.mu.d)
```

```
##      mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## 0.29000439 0.08800000 0.91300000 0.08800000 -0.02360565 0.42465976
```

```
## Was lichen network similarity determined by genotype?
set.seed(1234)
cn.perm <- vegan::adonis2(cn.d.onc ~ geno + BR + PC + SR,
                          data = onc.dat, permutations = 10000, mrank = TRUE)
set.seed(1234)
cn.perm.ng <- vegan::adonis2(cn.d.onc ~ BR + PC + SR,
                             data = onc.dat, permutations = 10000, mrank = TRUE)
cn.perm.ng
```

```
## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 10000
##
## vegan::adonis2(formula = cn.d.onc ~ BR + PC + SR, data = onc.dat, permutations = 10000, mrank = TRUE)
##      Df SumOfSqs      R2      F    Pr(>F)
## BR      1      61.42 0.03968  4.1680  0.04050 *
## PC      1      49.47 0.03197  3.3573  0.06549 .
## SR      1     655.76 0.42373 44.5034 9.999e-05 ***
## Residual 53      780.96 0.50462
## Total    56    1547.61 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
cn.perm
```

```
## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 10000
##
```

```

## vegan::adonis2(formula = cn.d.onc ~ geno + BR + PC + SR, data = onc.dat, permutations = 10000, mrank
##           Df SumOfSqs      R2      F    Pr(>F)
## geno      12   450.52 0.29111  2.6902  0.008299 **
## BR         1    29.11 0.01881  2.0858  0.150185
## PC         1    30.01 0.01939  2.1504  0.152285
## SR         1   465.78 0.30097 33.3755 9.999e-05 ***
## Residual  41    572.18 0.36972
## Total     56   1547.61 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

h2.tab[5, "p-value"] <- as.matrix(cn.perm)[1, "Pr(>F)"]
h2.tab[5, "H2"] <- H2(cn.perm, g = onc.dat[, "geno"], perm = 10000)
h2.tab[5, "R2"] <- R2(cn.perm)
h2.tab[5, "Response"] <- "Lichen Network"
                                # db rda for network similarity
dbr.cn.geno <- vegan::dbrda(cn.d.onc ~ geno, data = onc.dat, distance = "bray")
anova(dbr.cn.geno, permutations = 5000)

## Permutation test for dbrda under reduced model
## Permutation: free
## Number of permutations: 5000
##
## Model: vegan::dbrda(formula = cn.d.onc ~ geno, data = onc.dat, distance = "bray")
##           Df Variance      F Pr(>F)
## Model      12     8.045 1.5057  0.138
## Residual  44    19.591
H2(dbr.cn.geno)

## [1] 0.2911089
## What aspects of networks explained the similiarity?
## L = number of edges, LD = link density, C = connectivity,
## dcen = degree centrality
link.reml <- lme4::lmer(I(log(L + 0.00000001)) ~ (1 | geno),
                      data = onc.dat, REML = TRUE)
link.reml.pval <- RLRsim::exactRLRT(link.reml, nsim = 50000)
link.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 50000 simulated values)
##
## data:
## RLRT = 2.0484, p-value = 0.06632
fligner.test(log(onc.dat$L + 0.0000001), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: log(onc.dat$L + 1e-07) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 11.991, df = 12, p-value =
## 0.4464

```

```

shapiro.test(residuals(link.reml))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(link.reml)
## W = 0.83643, p-value = 2.036e-06
h2.tab[6, "p-value"] <- link.reml.pval$"p.value"
h2.tab[6, "H2"] <- H2(link.reml, g = onc.dat$geno)
h2.tab[6, "R2"] <- R2(link.reml)
R2(link.reml)

##          R2c
## 0.1701568
h2.tab[6, "Response"] <- "Number of Network Links"

                                # network centrality
cen.reml <- lme4::lmer(I(Cen^(1/2)) ~ (1 | geno),
                      data = onc.dat, REML = TRUE)
cen.reml.pval <- RLRsim::exactRLRT(cen.reml, nsim = 50000)
cen.reml.pval

##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 50000 simulated values)
##
## data:
## RLRT = 2.7801, p-value = 0.04018
fligner.test(onc.dat$L^(1/1), onc.dat$geno)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  onc.dat$L^(1/1) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 14.241, df = 12, p-value =
## 0.2856
shapiro.test(residuals(cen.reml))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(cen.reml)
## W = 0.90072, p-value = 0.0002041
h2.tab[7, "p-value"] <- cen.reml.pval$"p.value"
h2.tab[7, "H2"] <- H2(cen.reml, g = onc.dat$geno)
h2.tab[7, "R2"] <- R2(cen.reml)
R2(cen.reml)

##          R2c
## 0.2016649

```

```

h2.tab[7, "Response"] <- "Network Centrality"

                                # network modularity
mod.reml <- lme4::lmer(I(onc.ns[, "mod.lik"]^(1/4)) ~ (1 | geno),
                     data = onc.dat, REML = TRUE)
mod.reml.pval <- RLRsim::exactRLRT(mod.reml)
mod.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.23363, p-value = 0.2769
fligner.test(onc.ns[, "mod.lik"]^(1/4), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data:  onc.ns[, "mod.lik"]^(1/4) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 13.439, df = 12, p-value =
## 0.3379
shapiro.test(residuals(mod.reml))

##
## Shapiro-Wilk normality test
##
## data:  residuals(mod.reml)
## W = 0.54001, p-value = 4.252e-12
h2.tab[8, "p-value"] <- mod.reml.pval$"p.value"
h2.tab[8, "H2"] <- H2(mod.reml, g = onc.dat$geno)
h2.tab[8, "R2"] <- R2(mod.reml)
h2.tab[8, "Response"] <- "Network Modularity"

                                # network stats in relation to other variables
L.aov <- aov(I(log(L + 0.000001)) ~ BR + PC + SR, data = onc.dat)
summary(L.aov)

##              Df Sum Sq Mean Sq F value    Pr(>F)
## BR              1  102.3    102.3    2.776   0.1016
## PC              1  239.6    239.6    6.504   0.0137 *
## SR              1  957.0    957.0   25.980 4.71e-06 ***
## Residuals      53 1952.2     36.8
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
shapiro.test(residuals(L.aov))

##
## Shapiro-Wilk normality test
##
## data:  residuals(L.aov)
## W = 0.9629, p-value = 0.07794

```

```
cen.aov <- aov(I(Cen^(1/2)) ~ BR + PC + SR, data = onc.dat)
summary(cen.aov)
```

```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## BR              1   3.77     3.77    2.174    0.146
## PC              1   6.46     6.46    3.724    0.059 .
## SR              1  56.48    56.48   32.552 5.31e-07 ***
## Residuals      53  91.95     1.73
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
shapiro.test(residuals(cen.aov))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(cen.aov)
## W = 0.97222, p-value = 0.2126
```

```
mod.aov <- aov(I(onc.ns[, "mod.lik"]^(1/4)) ~ BR + PC + SR, data = onc.dat)
summary(mod.aov)
```

```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## BR              1 0.0442   0.0442    0.787    0.379
## PC              1 0.0879   0.0879    1.564    0.217
## SR              1 1.3799   1.3799   24.558 7.76e-06 ***
## Residuals      53 2.9781   0.0562
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
shapiro.test(residuals((mod.aov)))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals((mod.aov))
## W = 0.9201, p-value = 0.001078
```

```
##
cor.test(onc.ns[, "L"], onc.ns[, "Cen"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  onc.ns[, "L"] and onc.ns[, "Cen"]
## t = 13.37, df = 55, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7950728 0.9244074
## sample estimates:
##      cor
## 0.8744752
```

are these metrics correlated with network similarity

```
L.d <- dist(onc.dat$L)
cen.d <- dist(onc.dat$Cen)
mod.d <- dist(cen.mod.onc)
```

```
cn.L.cen.perm <- adonis2(cn.d.onc ~ L + Cen, data = onc.dat, mrank = TRUE)
```

```
## So, are there patterns in the centrality of individual lichen species?
sppcen.test <- apply(cen.spp[, apply(cen.spp, 2, sum) >= 2], 2, function(x)
  lme4::lmer(I(x^(1/2)) ~ (1 | geno), data = onc.dat, REML = TRUE))
```

```
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
```

```
sppcen.pval <- lapply(sppcen.test, RLRsim::exactRLRT)
sppcen.tab <- do.call(rbind, lapply(sppcen.pval, function(x)
  c(x[["statistic"]], x[["p.value"]])))
sppcen.h2 <- round(unlist(lapply(sppcen.test, H2)), 3)
sppcen.h2
```

```
##      Xg      Cs      Ls      Ch      Xm      Pm      Rs
## 0.000 0.127 0.000 0.258 0.201 0.000 0.000
```

```
## Mean centrality of species
sort(apply(cen.spp, 2, mean), decreasing = TRUE)
```

```
##          Cs          Ch          Ls          Rs          Xg          Pm
## 0.73204678 0.54157218 0.39722829 0.18378675 0.14553120 0.07914127
##          Xm          Pu          Pa
## 0.06376775 0.02105263 0.00000000
```

```
## Ordinations
### nits = 10,
### iconf = random
### epsilon = 1e-12 = acceptable change in stress
### maxit = 500 = maximum number of iterations
ord.com <- nmds.min(nms.com, 3)
```

```
## Minimum stress for given dimensionality: 0.1008923
## r^2 for minimum stress configuration: 0.9357192
## Minimum stress for given dimensionality: 0.1008923
## r^2 for minimum stress configuration: 0.9357192
ord.cn <- nmds.min(nms.cn, 2)
```

```
## Minimum stress for given dimensionality: 0.01065177
## r^2 for minimum stress configuration: 0.9993026
## Minimum stress for given dimensionality: 0.01065177
## r^2 for minimum stress configuration: 0.9993026
## checking variance explained by ordinations
ord1.cn.reml <- lme4::lmer(I(ord.cn[, 1]^(1/1)) ~ (1 | geno),
  data = onc.dat, REML = TRUE)
ord2.cn.reml <- lme4::lmer(I(ord.cn[, 2]^(1/1)) ~ (1 | geno),
  data = onc.dat, REML = TRUE)
ord1.cn.reml.pval <- RLRsim::exactRLRT(ord1.cn.reml)
ord2.cn.reml.pval <- RLRsim::exactRLRT(ord2.cn.reml)
ord1.cn.reml.pval
```

```
##
## simulated finite sample distribution of RLRT.
```

```

##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 1.0221, p-value = 0.1329
ord2.cn.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.5618, p-value = 0.1936
fligner.test(ord.cn[, 1]^(1/1), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: ord.cn[, 1]^(1/1) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 16.805, df = 12, p-value =
## 0.1571
fligner.test(ord.cn[, 2]^(1/1), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: ord.cn[, 2]^(1/1) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 9.9165, df = 12, p-value =
## 0.6233
ord1.com.reml <- lme4::lmer(I(ord.com[, 1]^(1/1)) ~ (1 | geno),
  data = onc.dat, REML = TRUE)
ord2.com.reml <- lme4::lmer(I(ord.com[, 2]^(1/1)) ~ (1 | geno),
  data = onc.dat, REML = TRUE)
ord1.com.reml.pval <- RLRsim::exactRLRT(ord1.com.reml)
ord2.com.reml.pval <- RLRsim::exactRLRT(ord2.com.reml)
ord1.com.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.1669, p-value = 0.3039
ord2.com.reml.pval

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:

```

```
## RLRT = 0.98197, p-value = 0.1352
fligner.test(ord.com[, 1]^(1/1), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: ord.com[, 1]^(1/1) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 9.3187, df = 12, p-value =
## 0.6755
fligner.test(ord.com[, 2]^(1/1), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: ord.com[, 2]^(1/1) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 16.947, df = 12, p-value =
## 0.1516
fligner.test(ord.com[, 3]^(1/1), onc.dat$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: ord.com[, 3]^(1/1) and onc.dat$geno
## Fligner-Killeen:med chi-squared = 14.943, df = 12, p-value =
## 0.2446
summary(lm(ord.cn[, 1] ~ SR + PC, data = onc.dat))

##
## Call:
## lm(formula = ord.cn[, 1] ~ SR + PC, data = onc.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.6007  -1.7887   0.1726   2.2110   6.7059
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.77927     1.05175   3.593 0.000706 ***
## SR          -2.89115     0.39475  -7.324 1.23e-09 ***
## PC           0.10728     0.02215   4.844 1.11e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.596 on 54 degrees of freedom
## Multiple R-squared:  0.5025, Adjusted R-squared:  0.4841
## F-statistic: 27.27 on 2 and 54 DF, p-value: 6.508e-09
summary(lm(ord.cn[, 2] ~ SR + PC, data = onc.dat))

##
## Call:
## lm(formula = ord.cn[, 2] ~ SR + PC, data = onc.dat)
##
## Residuals:
```



```
##      Min      1Q  Median      3Q      Max
## -2.4080 -0.9426 -0.6151  1.3669  2.9279
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.143124   0.420811  -2.716 0.008846 **
## SR           0.561645   0.157944   3.556 0.000793 ***
## PC          -0.013722   0.008862  -1.548 0.127384
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.439 on 54 degrees of freedom
## Multiple R-squared:  0.2223, Adjusted R-squared:  0.1935
## F-statistic: 7.718 on 2 and 54 DF,  p-value: 0.001127
summary(lm(ord.com[, 1] ~ SR + PC, data = onc.dat))
```

```
##
## Call:
## lm(formula = ord.com[, 1] ~ SR + PC, data = onc.dat)
##
## Residuals:
##      Min      1Q  Median      3Q      Max
## -0.18241 -0.09091 -0.01606  0.05475  0.65204
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.5145496  0.0395271 -13.018 < 2e-16 ***
## SR           0.0527258  0.0148358   3.554 0.000798 ***
## PC           0.0058018  0.0008324   6.970 4.61e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1351 on 54 degrees of freedom
## Multiple R-squared:  0.8048, Adjusted R-squared:  0.7976
## F-statistic: 111.3 on 2 and 54 DF,  p-value: < 2.2e-16
summary(lm(ord.com[, 2] ~ SR + PC, data = onc.dat))
```

```
##
## Call:
## lm(formula = ord.com[, 2] ~ SR + PC, data = onc.dat)
##
## Residuals:
##      Min      1Q  Median      3Q      Max
## -0.54228 -0.11829  0.03558  0.16463  0.50365
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.224171   0.068196  -3.287 0.00178 **
## SR           0.015539   0.025596   0.607 0.54634
## PC           0.002973   0.001436   2.070 0.04328 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```

## Residual standard error: 0.2331 on 54 degrees of freedom
## Multiple R-squared:  0.2151, Adjusted R-squared:  0.1861
## F-statistic: 7.4 on 2 and 54 DF,  p-value: 0.001444
## Lichen size distribution
## X. gallericulata thalli are about 0.22 +/- 0.003 cm2 on average
## with an average median size of 0.12 +/- 0.001 cm2
## and, size does not vary significantly with genotype.
xgs.reml <- lme4::lmer(I(mean.thallus) ~ (1 | geno),
                     data = xgs.data[xgs.data$geno %in% names(which(table(xgs.data$geno) > 2)), ],
                     REML = TRUE)
xgs.median.reml <- lme4::lmer(median.thallus ~ (1 | geno),
                             data = xgs.data[xgs.data$geno %in% names(which(table(xgs.data$geno) > 2)), ],
                             REML = TRUE)
RLRsim::exactRLRT(xgs.reml)

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 2.4792, p-value = 0.0498
RLRsim::exactRLRT(xgs.median.reml)

##
## simulated finite sample distribution of RLRT.
##
## (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.092023, p-value = 0.3395
fligner.test(xgs.data$mean.thallus, xgs.data$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: xgs.data$mean.thallus and xgs.data$geno
## Fligner-Killeen:med chi-squared = 13.244, df = 17, p-value =
## 0.7197
fligner.test(xgs.data$median.thallus, xgs.data$geno)

##
## Fligner-Killeen test of homogeneity of variances
##
## data: xgs.data$median.thallus and xgs.data$geno
## Fligner-Killeen:med chi-squared = 19.374, df = 17, p-value =
## 0.3075
mean(xgs.data$mean.thallus)

## [1] 0.1808442
sd(xgs.data$mean.thallus) / (length(xgs.data$mean.thallus) - 1)

```

```
## [1] 0.001845945
mean(xgs.data$median.thallus)

## [1] 0.1170852
sd(xgs.data$median.thallus) / (length(xgs.data$median.thallus) - 1)

## [1] 0.001223999

                                # ONC and Wild Stand (Uintah)
all.dat <- rbind(wild.dat[, c("BR", "PC", "SR", "L", "Cen")],
                onc.dat[, c("BR", "PC", "SR", "L", "Cen")])
                                # Network distances
cn.all <- cn.wild
for (i in 1:length(cn.wild)){
  cn.all[[i]] <- cn.wild[[i]][match(rownames(cn.onc[[1]]), rownames(cn.wild[[i]])),
                                match(colnames(cn.onc[[1]]), colnames(cn.wild[[i]]))]
}
cn.all <- append(cn.all, cn.onc)
cn.d.all <- distNet(cn.all, method = "bc")
cn.nms.geno <- c(rep("wild", length(cn.wild)), onc.geno)
if (!exists("cn.nms.all")){
  set.seed(12345)
  cn.nms.all <- nmms.min(nmms(cn.d.all, 2, 2))
  vec.all <- envfit(cn.nms.all, all.dat)

                                # jitter identical points
  cn.nms.all[cn.nms.geno == "H10", ] <- cn.nms.all[cn.nms.geno == "H10", ] - 0.2
}
```

Tables

```
h2.tab[, "H2"] <- round(as.numeric(h2.tab[, "H2"]), digits = 5)
h2.tab[, "R2"] <- round(as.numeric(h2.tab[, "R2"]), digits = 5)
h2.tab[, "p-value"] <- round(as.numeric(h2.tab[, "p-value"]), digits = 5)
h2.tab <- h2.tab[order(h2.tab[, "H2"], decreasing = TRUE), ]
h2.xtab <- xtable::xtable(h2.tab, caption =
  "Genotypic effects of cottonwood trees on the associated lichen community.",
  label = "tab:h2_table")
print(h2.xtab,
  type = "latex",
  include.rownames = FALSE,
  include.colnames = TRUE
)
```

% latex table generated in R 3.6.1 by xtable 1.8-4 package % Fri Sep 27 15:24:03 2019

```
                                # community permanova
rcom.ng.perm.xtab <- xtable::xtable(rcom.ng.perm, caption =
  "PerMANOVA Pseudo-F Table showing the predictors of community similarity.",
  label = "tab:com_ng_perm")
print(rcom.ng.perm.xtab,
  type = "latex",
  include.rownames = TRUE,
```

Response	H2	R2	p-value
Percent Rough Bark	0.37835	0.37835	2e-04
Network Centrality	0.20166	0.20166	0.04018
Percent Lichen Cover	0.17279	0.17279	0.038
Number of Network Links	0.17016	0.17016	0.06632
Lichen Community Composition	0.16093	0.24287	0.0032
Lichen Species Richness	0.09815	0.09815	0.1423
Lichen Network	0.06252	0.29111	0.0083
Network Modularity	0.05731	0.05731	0.2769

Table 1: Genotypic effects of cottonwood trees on the associated lichen community.

```
include.colnames = TRUE
)
```

% latex table generated in R 3.6.1 by xtable 1.8-4 package % Fri Sep 27 15:24:03 2019

	Df	SumOfSqs	R2	F	Pr(>F)
BR	1	0.44	0.04	3.74	0.0088
PC	1	3.86	0.34	32.85	0.0001
SR	1	0.78	0.07	6.60	0.0001
Residual	53	6.23	0.55		
Total	56	11.31	1.00		

Table 2: PerMANOVA Pseudo-F Table showing the predictors of community similarity.

```
rcom.perm.xtab <- xtable::xtable(rcom.perm, caption =
  "PerMANOVA Pseudo-F Table showing the predictors of community similarity.",
  label = "tab:rcom_perm")
print(rcom.perm.xtab,
  type = "latex",
  include.rownames = TRUE,
  include.colnames = TRUE
)
```

% latex table generated in R 3.6.1 by xtable 1.8-4 package % Fri Sep 27 15:24:03 2019

	Df	SumOfSqs	R2	F	Pr(>F)
geno	12	2.75	0.24	1.82	0.0032
BR	1	0.12	0.01	0.99	0.3901
PC	1	2.67	0.24	21.27	0.0001
SR	1	0.62	0.05	4.90	0.0010
Residual	41	5.15	0.46		
Total	56	11.31	1.00		

Table 3: PerMANOVA Pseudo-F Table showing the predictors of community similarity.

```
# network permanova
cn.perm.ng.xtab <- xtable::xtable(cn.perm.ng, caption =
  "PerMANOVA Pseudo-F Table showing the predictors of network similarity.",
  label = "tab:cn_perm_ng")
print(cn.perm.ng.xtab,
  type = "latex",
  include.rownames = TRUE,
```

```
include.colnames = TRUE
)
```

% latex table generated in R 3.6.1 by xtable 1.8-4 package % Fri Sep 27 15:24:04 2019

	Df	SumOfSqs	R2	F	Pr(>F)
BR	1	61.42	0.04	4.17	0.0405
PC	1	49.47	0.03	3.36	0.0655
SR	1	655.76	0.42	44.50	0.0001
Residual	53	780.96	0.50		
Total	56	1547.61	1.00		

Table 4: PerMANOVA Pseudo-F Table showing the predictors of network similarity.

```
cn.perm.xtab <- xtable::xtable(cn.perm, caption =
  "PerMANOVA Pseudo-F Table showing the predictors of network similarity.",
  label = "tab:cn_perm")
print(cn.perm.xtab,
  type = "latex",
  include.rownames = TRUE,
  include.colnames = TRUE
)
```

% latex table generated in R 3.6.1 by xtable 1.8-4 package % Fri Sep 27 15:24:04 2019

	Df	SumOfSqs	R2	F	Pr(>F)
geno	12	450.52	0.29	2.69	0.0083
BR	1	29.11	0.02	2.09	0.1502
PC	1	30.01	0.02	2.15	0.1523
SR	1	465.78	0.30	33.38	0.0001
Residual	41	572.18	0.37		
Total	56	1547.61	1.00		

Table 5: PerMANOVA Pseudo-F Table showing the predictors of network similarity.

```
# network metrics anova
L.aov.xtab <- xtable::xtable(L.aov, caption =
  "ANOVA F Table showing the predictors of the number of network links.",
  label = "tab:L_aov")
print(L.aov.xtab,
  type = "latex",
  include.rownames = TRUE,
  include.colnames = TRUE
)
```

% latex table generated in R 3.6.1 by xtable 1.8-4 package % Fri Sep 27 15:24:04 2019

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
BR	1	102.25	102.25	2.78	0.1016
PC	1	239.57	239.57	6.50	0.0137
SR	1	956.96	956.96	25.98	0.0000
Residuals	53	1952.23	36.83		

Table 6: ANOVA F Table showing the predictors of the number of network links.

```
cen.aov.xtab <- xtable::xtable(cen.aov, caption =
  "ANOVA F Table showing the predictors of network centralization.",
  label = "tab:cen_aov")
print(cen.aov.xtab,
  type = "latex",
  include.rownames = TRUE,
  include.colnames = TRUE
)
```

% latex table generated in R 3.6.1 by xtable 1.8-4 package % Fri Sep 27 15:24:05 2019

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
BR	1	3.77	3.77	2.17	0.1463
PC	1	6.46	6.46	3.72	0.0590
SR	1	56.48	56.48	32.55	0.0000
Residuals	53	91.95	1.73		

Table 7: ANOVA F Table showing the predictors of network centralization.

```
# networks and network metrics
# permanova
cn.L.cen.perm.xtab <- xtable::xtable(cn.L.cen.perm, caption =
  "PerMANOVA Pseudo-F Table showing the predictors of network similarity.",
  label = "tab:cn_L_cen_perm")
print(cn.L.cen.perm.xtab,
  type = "latex",
  include.rownames = TRUE,
  include.colnames = TRUE
)
```

% latex table generated in R 3.6.1 by xtable 1.8-4 package % Fri Sep 27 15:24:05 2019

	Df	SumOfSqs	R2	F	Pr(>F)
L	1	1330.80	0.86	734.67	0.0010
Cen	1	118.99	0.08	65.69	0.0010
Residual	54	97.82	0.06		
Total	56	1547.61	1.00		

Table 8: PerMANOVA Pseudo-F Table showing the predictors of network similarity.

Plots

Figure: Genotype barplots Community composition NMDS with vectors

```
par(mfrow = c(1, 1), mar = c(5.1, 4.1, 4.1, 2.1) / 1)
chp.coord <- ch.plot(ord.com[, 1:2], onc.geno,
  cex = 2, mu.pch = 19,
  pt.col = "white",
  bar.col = "darkgrey")
text(chp.coord, labels = rownames(chp.coord))
plot(vec.com, col = "black", lwd = 4)
```

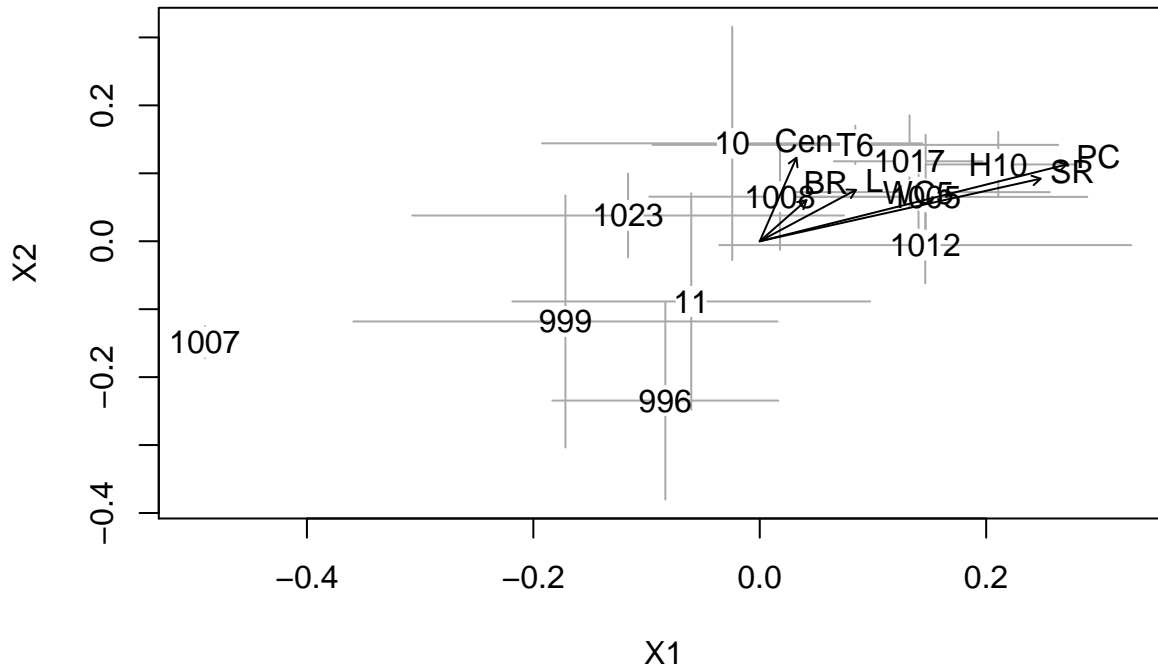


Figure: Lichen networks

```
par(mfrow = c(2, 2), mar = c(0, 0.1, 1.0, 0.1))
set.seed(123)
net.col <- sign(meanNet(cn.mu.onc))
net.col[net.col == -1] <- 2
net.col[net.col == 1] <- 1
coord <- gplot(abs(meanNet(cn.mu.onc)), gmode = "digraph",
  displaylabels = TRUE,
  edge.lwd = abs(meanNet(cn.mu.onc)) * 20,
  edge.col = net.col,
  vertex.col = "black",
  vertex.cex = 0.5,
  arrowhead.cex = 0.5,
  label.cex = 1,
  main = "All Genotypes")
cn.mu.plot <- cn.mu.onc[names(cn.mu.onc) %in% c("996", "11", "1008")]
for (i in 1:length(cn.mu.plot)){
  net.col <- sign(cn.mu.plot[[i]])
  net.col[net.col == -1] <- 2
  net.col[net.col == 1] <- 1
  set.seed(123)
  gplot(abs(cn.mu.plot[[i]]), gmode = "digraph",
    displaylabels = TRUE,
    coord = coord,
    edge.lwd = abs(cn.mu.plot[[i]]) * 20,
    edge.col = net.col,
    vertex.col = "black",
    vertex.cex = 0.5,
    arrowhead.cex = 0.5,
    label.cex = 1,
```

```

    main = names(cn.mu.plot)[i])
}

```

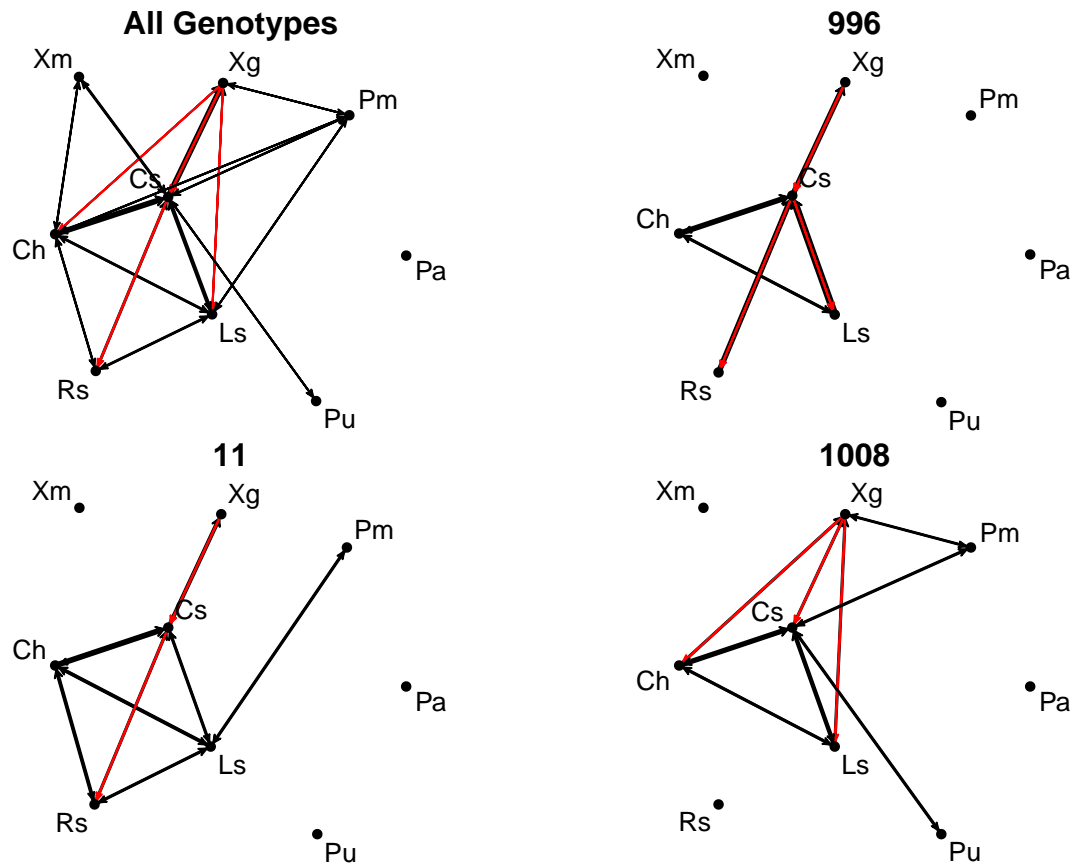


Figure: Genotype network similarity by genotype

```

par(mfrow = c(1, 1), mar = c(5.1, 4.1, 4.1, 2.1))
chp.coord <- ch.plot(cn.nms.onc, onc.geno,
  cex = 2, mu.pch = 19,
  pt.col = "white",
  bar.col = "darkgrey")
text(chp.coord, labels = rownames(chp.coord))
plot(vec.cn, col = "black")

```

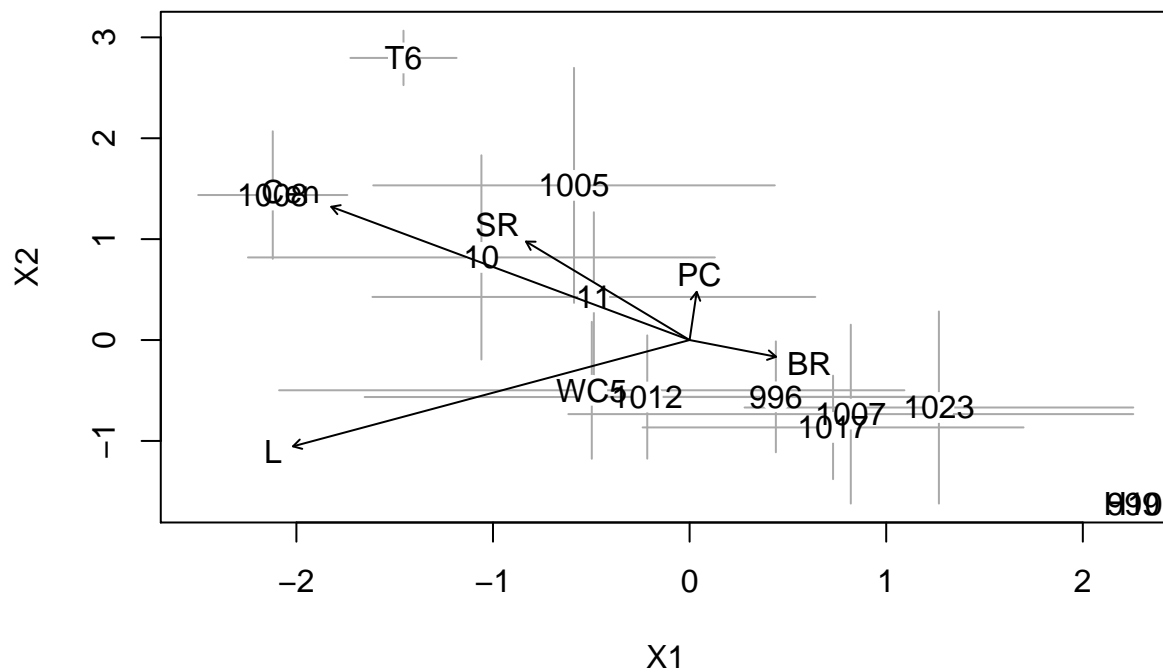
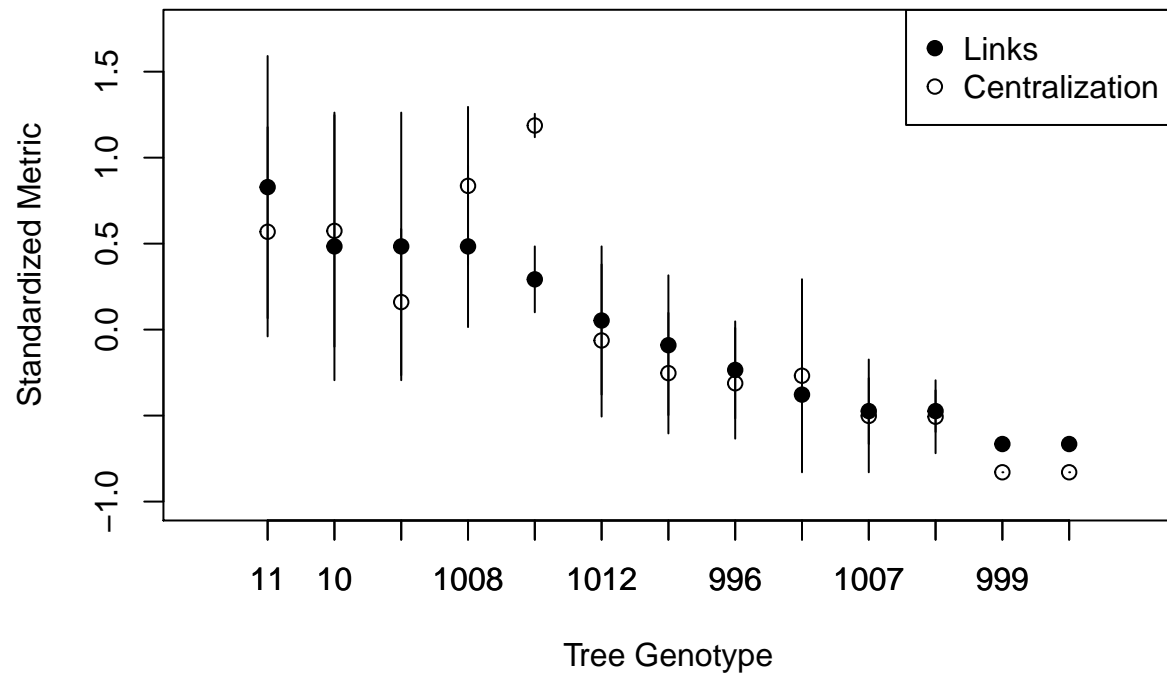



Figure: (A) Linkage and centrality by genotype and (B) Total cover and species richness predict L and Cen

```
mdc.plot(onc.dat[, "geno"], onc.dat[, "L"], ylim = c(-1, 1.75),
         xlab = "Tree Genotype", ylab = "Standardized Metric",
         ord = order(tapply(onc.dat[, "L"], onc.dat[, "geno"], mean), decreasing = TRUE))
mdc.plot(onc.dat[, "geno"], onc.dat[, "Cen"], add = TRUE, pch = 1,
         ord = order(tapply(onc.dat[, "L"], onc.dat[, "geno"], mean), decreasing = TRUE))
legend("topright", legend = c("Links", "Centralization"), pch = c(19, 1), bty = "none")
```



Supplementary Figure: Lichen size distribution

```
plot(density(xgs.data$median.thallus),
     xlab = "Median Lichen Thallus Area (cm^2)",
     main = "")
abline(v = median(xgs.data$median.thallus, na.rm = TRUE), lty = 2)
```

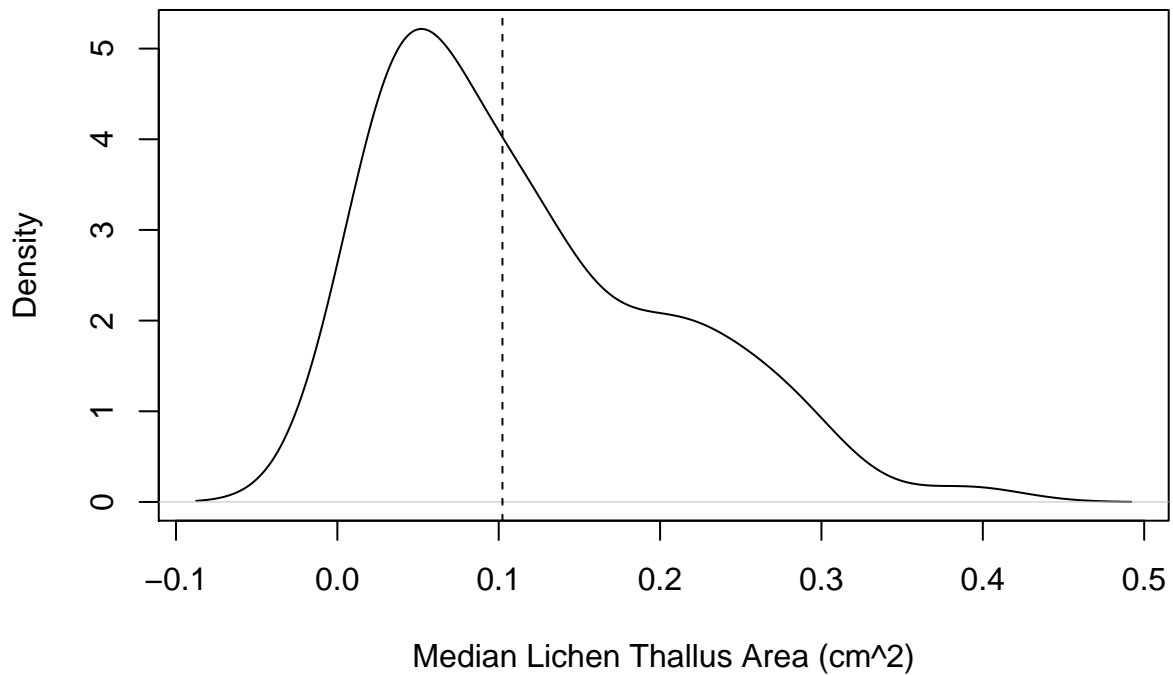
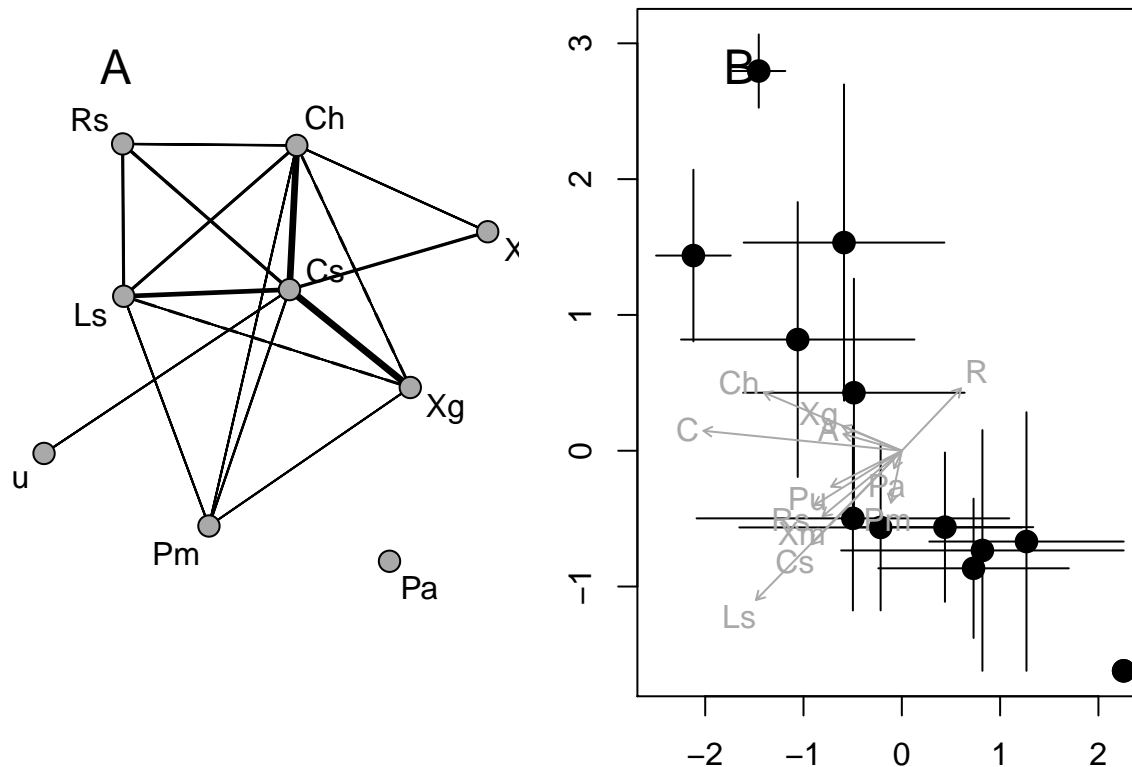


Figure 2

```
par(mfrow = c(1, 2), mar = c(5.1, 4.1, 4.1, 2.1) / 2)
gplot(meanNet(cn.mu.onc), gmode = "graph",
      displaylabels = TRUE,
      edge.lwd = meanNet(cn.mu.onc) * 20,
      vertex.col = "darkgrey")
legend("topleft", legend = "A", bty = "n", cex = 1.5)
chp.coord <- ch.plot(cn.nms.onc, onc.geno, cex = 1.5)
plot(nv.onc, col = "darkgrey")
legend("topleft", legend = "B", bty = "n", cex = 1.5)
```



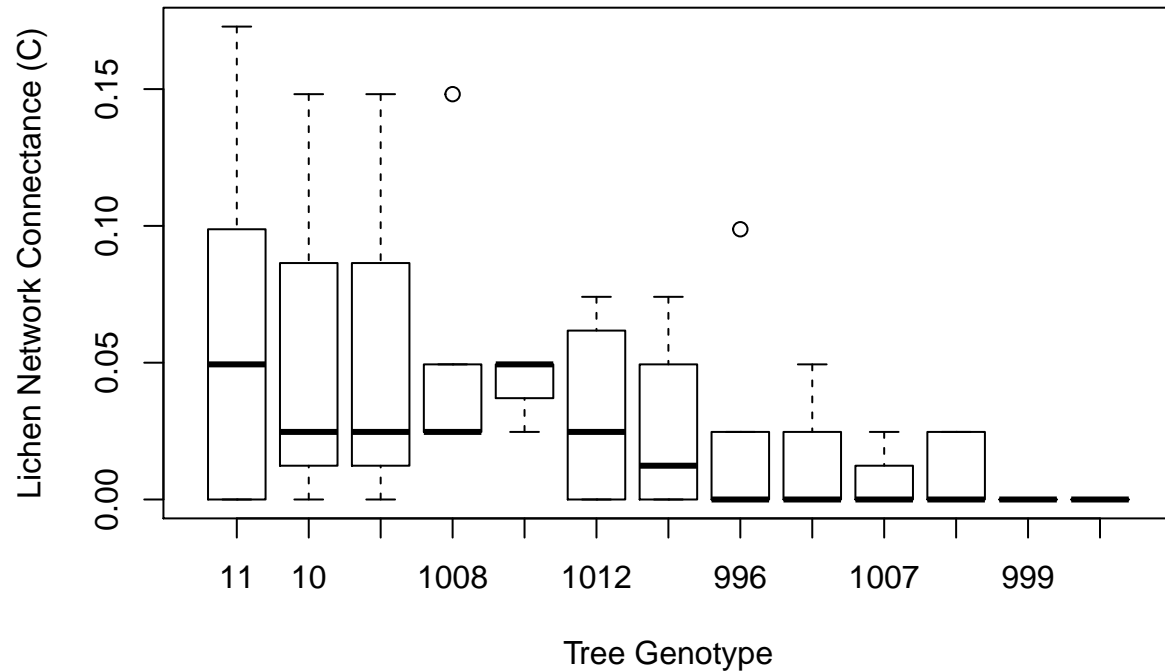
```
par(mfrow = c(1, 1), mar = c(5.1, 4.1, 4.1, 2.1))
bipartite::plotweb(pw.onc, method = "normal",
                  text.rot = 45,
                  col.low = col.pal[mods.onc$tree],
                  col.high = col.pal[mods.onc$sp],
                  bor.col.low = col.pal[mods.onc$tree],
                  bor.col.high = col.pal[mods.onc$sp],
                  col.interaction = "grey70",
                  bor.col.interaction = "grey70",
                  labsize = 1.5)
```



```
## H10 -0.011830997 0.122603983
## T6 0.002941633 0.064173827
## WC5 -0.128224482 0.016507373
```

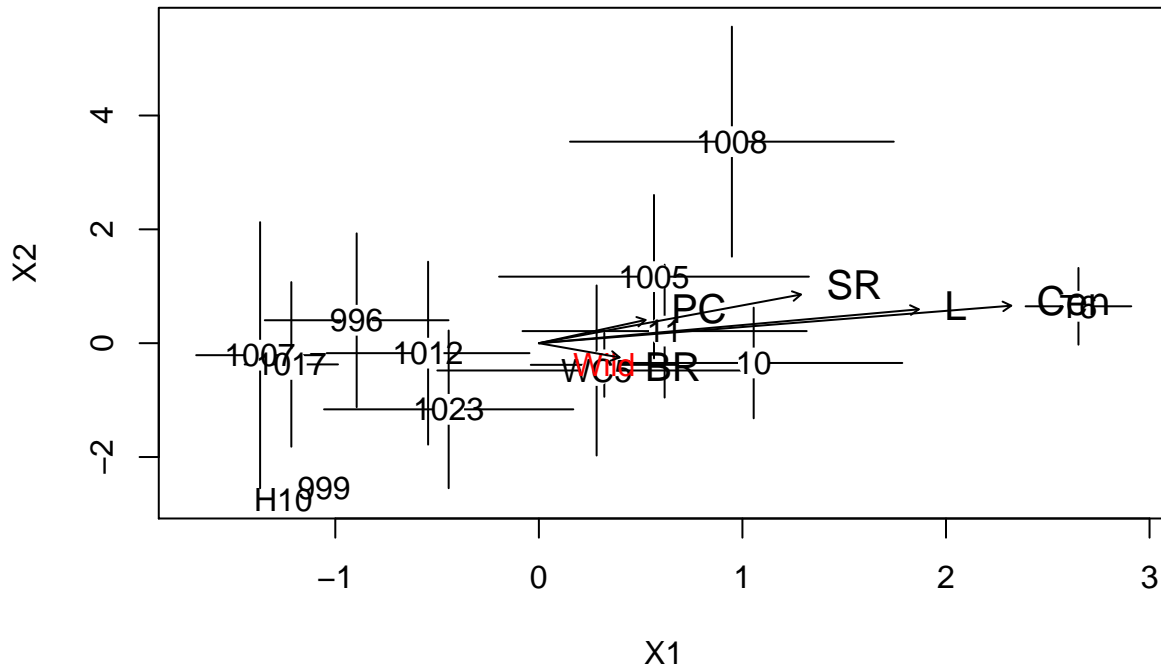
```
## plot(cv.onc, col = "grey30")
## legend("topleft", legend = "A")
```

```
g.order <- tapply(ns.onc[, "C"], onc.geno, mean)
g.order <- names(g.order)[order(g.order, decreasing = TRUE)]
onc.g <- factor(onc.geno, levels = g.order)
plot(ns.onc[, "C"] ~ onc.g, xlab = "Tree Genotype", ylab = "Lichen Network Connectance (C)")
```



Which wild uintah trees are similar to garden trees?

```
coords <- ch.plot(cn.nms.all, cn.nms.geno, mu.pch = "", cex = 2)
points(coords, pch = 19, col = "white", cex = 2)
text(coords[!grepl("wild", rownames(coords)), ],
      labels = rownames(coords)[!grepl("wild", rownames(coords))],
      col = "black")
text(coords[grepl("wild", rownames(coords)), 1],
      coords[grepl("wild", rownames(coords)), 2],
      labels = "Wild", col = "red")
plot(vec.all, col = "black", cex = 1.23)
```



Send results to manuscript

```
manuscript.dir <- "../..//lcn_manuscript"
### Send tables and figures to manuscript directory
if (exists("manuscript.dir")){
  tabs.figs <- dir(manuscript.dir)
  tab.fig.update <- dir("../results/lcn_notebook_files/figure-latex/",
                        full.names = TRUE)[
                        dir("../results/lcn_notebook_files/figure-latex/") %in% tabs.figs]
  tab.fig.update <- c(tab.fig.update,
                     dir("../docs", full.names = TRUE)[dir("../docs") %in% tabs.figs])
  sapply(tab.fig.update, file.copy, to = manuscript.dir, overwrite = TRUE)
  # supplementary figures
  si.dir <- paste0(manuscript.dir, "/supplement")
  si <- dir(si.dir)
  si.update <- dir("../results/lcn_notebook_files/figure-latex/",
                  full.names = TRUE)[
                  dir("../results/lcn_notebook_files/figure-latex/") %in% si]
  si.update <- c(si.update, dir("../docs", full.names = TRUE)[dir("../docs") %in% si])
  sapply(si.update, file.copy, to = si.dir,
        overwrite = TRUE)
}

## named list()
```

Loading and pre-processing data

```
## This is a place-holder for the echoing the data loading code.
```