Rock Lichen data from Sunset Crater

M.K. Lau

July 9, 2021

Data Summary

- This is an analysis of the effect of Pinyon Pine tree traits on the saxicole (lichen and moss) community on rocks under the canopy of the trees.
- Trees were sampled in a pairwise design in which pairs were comprised of one tree that is susceptible to the herbivory of a stem boring moth (*Dioryctria abietella*) and an adjacent tree that is resistant to the moth.
- As tree resistance to the moth is genetically based, pairwise sampling was conducted in order to isolate this genetic effect.
- Some trees that were sampled were dead, these trees were removed from the analysis.
- Plant data were observed by R. Michalet
  - Vegetation.xlsx
  - Light penetration.xls
  - light_&_litter(1).xls

Main Results

- Rock epiphyte communities were adequately sampled, based on species accumulation curves, with moth resistant trees accumulating slightly more lichen species.
- Several tree variables, including light availability, leaf litter abundance and rock abundance, were impacted by moth susceptibility, creating strong differences in sub-canopy conditions.
- Saxicole community abundance, richness, diversity, composition were significantly, generally negatively, affected by moth herbivory.
- Correlation analysis supported an indirect link between genetically based moth susceptibility and impacts on lichen communities via decreasing rock (i.e. habitat) availability through increased leaf abscission and accumulation on rocks under trees.

Analysis and Results

Analyses were conducted in the **R** statistical programming language. The following section loads dependencies and custom functions used in the analysis.

Dependencies

```
cran.pkgs <- c("reshape2", "vegan", "ecodist", "xtable", "knitr",
               "semPlot", "lavaan", "piecewiseSEM", "distantia",
               "tidySEM", "readxl", "psych")

## install packages that are not installed
if (any(!(cran.pkgs %in% installed.packages()[, 1]))){
```

```r
    sapply(cran.pkgs[which(!(cran.pkgs %in%
                             installed.packages()[, 1]))],
           install.packages,
           dependencies = TRUE,
           repos = 'http://cran.us.r-project.org')
}

## Load libraries
sapply(cran.pkgs, library, quietly = TRUE, character.only = TRUE)

## Custom Functions

## se: Calculate the standard error of a variable.
se <- function(x){sd(x) / sqrt(length(x))}
```

Load Data

The following are variable descriptions (Variable, Type, Range, Definition):

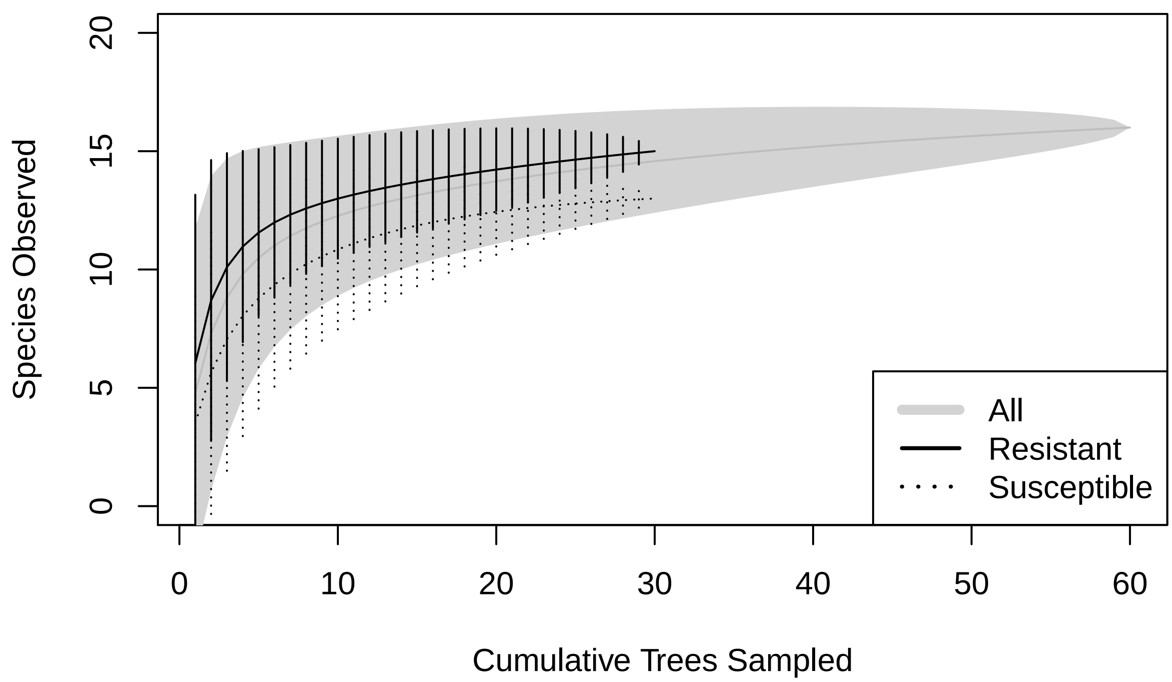
- Moth,categorical,0 or 1,Was the tree susceptible (0) or resistant (1) to moth attack
- Live/Dead,categorical,0 or 1,Was the tree dead (0) or alive (1)
- Litter %,continuous,0 to 100,Percent cover inside quadrat
- Rocks > 3cm %,continuous,0 to 100,Percent cover of rocks > 3cm? inside quadrat
- Rocks < 3cm %,continuous,0 to 100,Percent cover of rocks < 3cm? inside quadrat
- Shrubs %,continuous,0 to 100,Percent cover of shrubs inside quadrat
- Grass %,continuous,0 to 100,Percent cover of grass inside quadrat
- Branches %,continuous,0 to 100,Percent cover of branches on ground inside quadrat
- Distance,continuous,0 to 100,"Distance from main trunk, converted to percent of crown radius at that azimuth"
- Azimuth,continuous,0 to 360,Compass direction from main trunk
- Slope,continuous,0 to 90,Topographical steepness
- Aspect,continuous,0 to 360,Compass direction of slope
- Light,continuous,,Amount of light available to epiliths

```r
## Data are in ../data/scrl
l.dat <- read.csv("./data/spp_env_combined.csv")

## Fix species names
colnames(l.dat)[colnames(l.dat) == "Acasup"] <- "Acaame"

## Summary of data
summary(l.dat)

## remove dead trees
l.dat <- l.dat[l.dat[, "Live.Dead"] != 0, ]

## Lichen species list
spp.l <- c("Acacon", "Acaame", "Acaobp", "Sterile.sp", "Brown.cr",
"Lobalp", "Canros", "Calare", "Phydub", "Rhichr", "Xanlin", "Xanpli",
"Xanele", "GrBr.cr", "Gray.cr")
spp.moss <- c("Synrur", "Cerpur.Bryarg")

## Create a community matrix
com <- l.dat[, colnames(l.dat) %in% c(spp.l, spp.moss)]
```

```r
com.moss <- l.dat[, colnames(l.dat) %in% spp.moss]

## Add the tree labels to the rownames
rownames(com) <- paste(l.dat[, "Moth"], l.dat[, "Tree.pairs"], sep = "_")
rownames(com.moss) <- paste(l.dat[, "Moth"], l.dat[, "Tree.pairs"], sep = "_")
rownames(l.dat) <- paste(l.dat[, "Moth"], l.dat[, "Tree.pairs"], sep = "_")

## Paired environmental differences
total.rocks  <- apply(l.dat[, c("Big.rocks..", "Small.rocks..")], 1, sum)
env <- l.dat[, c("Litter..", "Big.rocks..", "Small.rocks..",
                       "Shrubs..", "Grass..", "Branches..",
                       "Light...N", "Light...S", "Light...average")]
env <- cbind(env, total.rocks)
env.dif <- apply(env, 2, function(x, p) tapply(x, p, diff), p = l.dat[, "Tree.pairs"])
```

Saxicole communities were sufficiently sampled

```r
spa.all <- specaccum(com, method = "exact")
spa.res <- specaccum(com[l.dat[, "Moth"] == 1, ], method = "exact")
spa.sus <- specaccum(com[l.dat[, "Moth"] == 0, ], method = "exact")

plot(spa.all,
     ylim = c(0, 20),
     xlab = "Cumulative Trees Sampled",
     ylab = "Species Observed",
     col = "grey", ci.col = 'lightgrey', ci.type = "poly", ci.lty = 0)
plot(spa.res, ci.col = "black", ci.type = "bar", lty = 1, add = TRUE, ci.lty = 1)
plot(spa.sus, ci.col = "black", ci.type = "bar", lty = 3, add = TRUE, ci.lty = 3)
legend("bottomright",
       legend = c("All", "Resistant", "Susceptible"),
       lty = c(1, 1, 3), lwd = c(5, 2, 2), col = c("lightgrey", "black", "black"))
```



```r
pdf("./results/scrl_spp-accum.pdf", width = 5, height = 5)
plot(spa.all,
     ylim = c(0, 20),
     xlab = "Cumulative Trees Sampled",
     ylab = "Species Observed",
     col = "grey", ci.col = 'lightgrey', ci.type = "poly", ci.lty = 0)
plot(spa.res, ci.col = "black", ci.type = "bar", lty = 1, add = TRUE, ci.lty = 1)
plot(spa.sus, ci.col = "black", ci.type = "bar", lty = 3, add = TRUE, ci.lty = 3)
legend("bottomright",
       legend = c("All", "Resistant", "Susceptible"),
       lty = c(1, 1, 3), lwd = c(5, 2, 2), col = c("lightgrey", "black", "black"))
dev.off()
```

```
## pdf
##   2
```

Moth trees have different microenvironments

```r
env.test.l <- apply(env.dif, 2, t.test)
env.test.l <- lapply(env.test.l, unlist)
env.test.tab <- do.call(rbind, env.test.l)
env.test.tab <- env.test.tab[, c(1, 2, 3, 6, 4, 5)]
env.test.tab <- apply(env.test.tab, 2, as.numeric)
rownames(env.test.tab) <- names(env.test.l)
colnames(env.test.tab) <- c("t", "df", "p-value", "Mean Difference", "Lower CI 95%", "Upper CI 95%")
kable(env.test.tab, digits = 4)
```

	t	df	p-value	Mean Difference	Lower CI 95%	Upper CI 95%
Litter..	2.8665	29	0.0077	15.0700	4.3178	25.8222
Big.rocks..	-2.4617	29	0.0200	-9.6837	-17.7289	-1.6384
Small.rocks..	-2.0792	29	0.0466	-4.9750	-9.8688	-0.0812
Shrubs..	-1.7605	29	0.0889	-0.5147	-1.1126	0.0832
Grass..	-1.0000	29	0.3256	-0.0493	-0.1502	0.0516
Branches..	1.0000	29	0.3256	0.1420	-0.1484	0.4324
Light...N	-8.0191	29	0.0000	-15.9767	-20.0514	-11.9019
Light...S	-7.5187	29	0.0000	-14.2900	-18.1772	-10.4028
Light...average	-9.2728	29	0.0000	-15.1333	-18.4712	-11.7955
total.rocks	-2.8178	29	0.0086	-14.6587	-25.2983	-4.0190

Moth trees have different lichen communities

```r
abun <- apply(com, 1, sum)
rich <- apply(com, 1, function(x) sum(sign(x)))
shan <- apply(com, 1, diversity, index = "shannon")
tt.a <- t.test(tapply(abun, l.dat[, "Tree.pairs"], diff))
tt.r <- t.test(tapply(rich, l.dat[, "Tree.pairs"], diff))
tt.h <- t.test(tapply(shan, l.dat[, "Tree.pairs"], diff))
tt.arh <- do.call(rbind,
                  list(a = unlist(tt.a),
                       r = unlist(tt.r),
                       h = unlist(tt.h)))
tt.arh <- apply(tt.arh[, 1:6], 2, as.numeric)
```

```r
ard.mu <- rbind(tapply(abun, l.dat[, "Moth"], mean),
                tapply(rich, l.dat[, "Moth"], mean),
                tapply(shan, l.dat[, "Moth"], mean))
ard.se <- rbind(tapply(abun, l.dat[, "Moth"], se),
                tapply(rich, l.dat[, "Moth"], se),
                tapply(shan, l.dat[, "Moth"], se))
ard.tab <- cbind(ard.mu[, "0"], ard.se[, "0"],
                 ard.mu[, "1"], ard.se[, "1"])
colnames(ard.tab) <- c("Susceptible Mean", "Susceptible SE",
                       "Resistant Mean", "Resistant SE")
rownames(ard.tab) <- c("Abundance", "Richness", "Diversity (Shannon)")
```

```r
kable(ard.tab, digits = 3)
```

	Susceptible Mean	Susceptible SE	Resistant Mean	Resistant SE
Abundance	1.210	0.351	2.754	0.567
Richness	3.500	0.542	6.033	0.662
Diversity (Shannon)	0.707	0.119	1.144	0.125

```r
kable(tt.arh, digits = 3)
```

statistic.t	parameter.df	p.value	conf.int1	conf.int2	estimate.mean of x
-2.249	29	0.032	-2.948	-0.140	-1.544
-2.955	29	0.006	-4.287	-0.780	-2.533
-2.447	29	0.021	-0.802	-0.072	-0.437

Composition is different (PERMANOVA, in text and supplement)

```r
com.ds <- cbind(com, ds = rep(0.0001, nrow(com)))
com.ds.rel <- apply(com, 2, function(x) x/max(x))
com.ds.rel <- cbind(com.ds.rel, ds = rep(0.0001, nrow(com)))
com.ds.rel[is.na(com.ds.rel)] <- 0

set.seed(123)
ptab.moth <- adonis2(com.ds~ Moth, data = l.dat,
                     strata = l.dat[, "Tree.pairs"],
                     by = "margin", nperm = 100000)
set.seed(123)
ptab.moth.rel <- adonis2(com.ds.rel ~ Moth, data = l.dat,
                         strata = l.dat[, "Tree.pairs"],
                         by = "margin", nperm = 100000)

kable(ptab.moth)
```

	Df	SumOfSqs	R2	F	Pr(>F)
Moth	1	0.8329281	0.0389768	2.352343	0.023
Residual	58	20.5368939	0.9610232	NA	NA
Total	59	21.3698219	1.0000000	NA	NA

```r
kable(ptab.moth.rel)
```

	Df	SumOfSqs	R2	F	Pr(>F)
Moth	1	0.8791695	0.0405034	2.448363	0.021
Residual	58	20.8269063	0.9594966	NA	NA
Total	59	21.7060758	1.0000000	NA	NA

three main species were reduced by moths (FDR paired t-tests, in text + supplement)

```r
ind.spp <- apply(com, 2, function(x, p) t.test(tapply(x, p, diff)), p = l.dat[, "Tree.pairs"])
isp <- apply(do.call(rbind, lapply(ind.spp, unlist)), 2, as.numeric)
```

```
## Warning in apply(do.call(rbind, lapply(ind.spp, unlist)), 2, as.numeric): NAs
## introduced by coercion

## Warning in apply(do.call(rbind, lapply(ind.spp, unlist)), 2, as.numeric): NAs
## introduced by coercion

## Warning in apply(do.call(rbind, lapply(ind.spp, unlist)), 2, as.numeric): NAs
## introduced by coercion
```

```r
rownames(isp) <- names(ind.spp)
isp[, "p.value"] <- p.adjust(isp[, "p.value"], method = "fdr")
isp.all <- isp[, !(apply(isp, 2, function(x) all(is.na(x))))]
isp <- isp[order(isp[, "p.value"]), ]
```

```r
isp.all <- isp.all[, c(1, 2, 3, 6, 4, 5)]
colnames(isp.all) <- c("t", "df", "p-value", "Mean Difference", "Lower CI 95%", "Upper CI 95%")
kable(isp.all, digits = 4)
```

	t	df	p-value	Mean Difference	Lower CI 95%	Upper CI 95%
Acacon	-3.3776	29	0.0159	-0.0447	-0.0717	-0.0176
Acaame	-3.2421	29	0.0159	-0.1607	-0.2620	-0.0593
Acaobp	-1.0747	29	0.4341	-0.2860	-0.8303	0.2583
Sterile.sp	-1.0000	29	0.4341	-0.0020	-0.0061	0.0021
Brown.cr	NaN	29	NaN	0.0000	NaN	NaN
Lobalp	-2.0414	29	0.2016	-0.0047	-0.0093	0.0000
Canros	-3.5819	29	0.0159	-0.3837	-0.6027	-0.1646
Calare	-1.6076	29	0.2563	-0.0307	-0.0697	0.0083
Phydub	-1.9226	29	0.2061	-0.1053	-0.2174	0.0067
Rhichr	-1.5803	29	0.2563	-0.2310	-0.5300	0.0680
Xanlin	-0.6170	29	0.6672	-0.2267	-0.9781	0.5247
Xanpli	-0.2598	29	0.8500	-0.0277	-0.2455	0.1901
Xanele	-1.5662	29	0.2563	-0.0473	-0.1091	0.0145
GrBr.cr	1.0000	29	0.4341	0.0013	-0.0014	0.0041
Gray.cr	0.1093	29	0.9137	0.0003	-0.0059	0.0066
Synrur	0.3628	29	0.8221	0.0220	-0.1020	0.1460
Cerpur.Bryarg	-1.2357	29	0.4027	-0.0173	-0.0460	0.0114

```r
write.csv(round(isp.all, 5), file = "results/scrl_isp_table.csv")
```

Calculate the average abundances of the indicators

```
isp.names <- as.character(na.omit(rownames(isp[isp[, "p.value"] < 0.05, ])))
isp.com <- com[,colnames(com) %in% isp.names]
isp.dif <- apply(isp.com, 2, function(x,y) tapply(x, y, diff), y = l.dat[ ,"Tree.pairs"])
```

Create a multi-bar plot figure for the community.

```
isp.dat <- melt(isp.dif)
colnames(isp.dat) <- c("Tree.pairs", "Species", "diff")
isp.mu <- tapply(isp.dat[, "diff"], isp.dat[, "Species"], mean)
isp.se <- tapply(isp.dat[, "diff"], isp.dat[, "Species"], se)
ard.dif <- cbind(tapply(abun, l.dat[, "Tree.pairs"], diff),
                 tapply(rich, l.dat[, "Tree.pairs"], diff),
                 tapply(shan, l.dat[, "Tree.pairs"], diff))
colnames(ard.dif) <- c("Abundance", "Richness", "Diversity")
ard.dif <- apply(ard.dif, 2, function(x) x / max(abs(x)))
ard.dat <- melt(ard.dif)
colnames(ard.dat) <- c("Tree.pairs", "Stat", "diff")
ard.mu <- tapply(ard.dat[, "diff"], ard.dat[, "Stat"], mean)
ard.se <- tapply(ard.dat[, "diff"], ard.dat[, "Stat"], se)

pdf(file = "./results/plot_isp_ard_lichen.pdf", width = 9, height = 5)

par(mfrow = c(1,2))
bp.out <- barplot(ard.mu, col = "darkgrey", ylim = c(-1.0, 0),
                  ylab = "Relativized Difference (S - R)", border = "NA")
segments(bp.out[, 1], ard.mu + ard.se,
         bp.out[, 1], ard.mu - ard.se,
         lwd = 1.5)
bp.out <- barplot(isp.mu, col = "darkgrey", ylim = c(-0.5, 0),
                  ylab = "Difference (S - R)", border = "NA",
            axisnames = TRUE,
            names.arg = sapply(names(isp.mu),
                function(x)
                                    paste(c(substr(x, 1, 1),
                                            substr(x, 4, 4)), collapse = "")))
segments(bp.out[, 1], isp.mu + isp.se,
         bp.out[, 1], isp.mu - isp.se,
         lwd = 1.5)
dev.off()
```

```
## pdf
##   2
```

Create a plot of the two most indicative species

```
pdf(file = "./results/scrl_complot.pdf", width = 7, height = 7)
plot(com[, c("Acaame", "Canros")], pch = l.dat[, "Moth"] + 1, cex = 3, col = l.dat[, "Moth"] + 1)
legend("topleft", title = "Tree Type", legend = c("Resistant", "Susceptible"), pch = c(2, 1), col = c(2
dev.off()
```

```
## pdf
##   2
```

Create plot with indicator taxa

```
pdf(file = "./results/scrl_pdif.pdf", width = 7, height = 7)
plot(melt(isp.dif)[-1], xlab = "Species", ylab = "Abundance Reduction")
```

```r
dev.off()
```

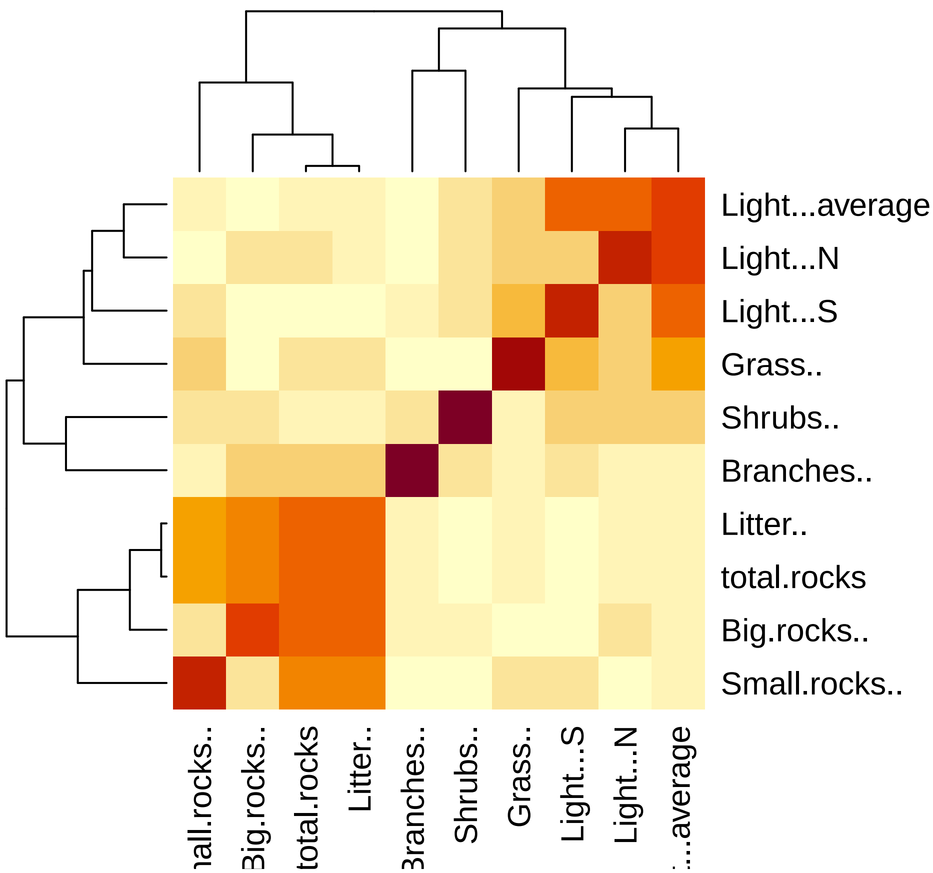
```
## pdf
##   2
```

Litter covering rocks was the main driver

Although light did significantly explain variation in the lichen community, this was not significant once the variation in litter was controlled for.

There was high correlation among environmental variables.

```r
heatmap(abs(round(cor(env.dif), 3)))
```




```r
set.seed(123)
ptab.env <- adonis2(com.ds ~ Litter.. +  Light...average, data = l.dat,
              strata = l.dat[, "Tree.pairs"],
              by = "margin", nperm = 100000)
kable(ptab.env)
```

	Df	SumOfSqs	R2	F	Pr(>F)
Litter..	1	1.0035484	0.0469610	2.972456	0.007
Light. . . average	1	0.4114619	0.0192543	1.218728	0.243
Residual	57	19.2441042	0.9005271	NA	NA
Total	59	21.3698219	1.0000000	NA	NA

```r
set.seed(123)
ptab.env <- adonis2(com.ds ~ Light...average + Litter.. , data = l.dat,
                    strata = l.dat[, "Tree.pairs"],
                    by = "margin", nperm = 100000)
kable(ptab.env)
```

	Df	SumOfSqs	R2	F	Pr(>F)
Light...average	1	0.4114619	0.0192543	1.218728	0.243
Litter..	1	1.0035484	0.0469610	2.972456	0.007
Residual	57	19.2441042	0.9005271	NA	NA
Total	59	21.3698219	1.0000000	NA	NA

```r
set.seed(123)
ptab.env <- adonis2(com.ds ~ Litter.. +  Light...average + Litter.. *  Light...average, data = l.dat,
                    strata = l.dat[, "Tree.pairs"],
                    by = "margin", nperm = 100000)
kable(ptab.env)
```

	Df	SumOfSqs	R2	F	Pr(>F)
Litter..:Light...average	1	0.6021127	0.0281758	1.808729	0.077
Residual	56	18.6419916	0.8723513	NA	NA
Total	59	21.3698219	1.0000000	NA	NA

```r
set.seed(123)
ptab.env <- adonis2(com.ds ~ total.rocks ,
                    strata = l.dat[, "Tree.pairs"],
                    by = "term", nperm = 100000)

kable(ptab.env)
```

	Df	SumOfSqs	R2	F	Pr(>F)
total.rocks	1	1.664876	0.0779078	4.900435	0.002
Residual	58	19.704946	0.9220922	NA	NA
Total	59	21.369822	1.0000000	NA	NA

```r
set.seed(123)
ptab.env <- adonis2(com.ds ~ Big.rocks.. , data = l.dat,
                    strata = l.dat[, "Tree.pairs"],
                    by = "term", nperm = 100000)

kable(ptab.env)
```

	Df	SumOfSqs	R2	F	Pr(>F)
Big.rocks..	1	2.428473	0.1136403	7.436188	0.001
Residual	58	18.941349	0.8863597	NA	NA
Total	59	21.369822	1.0000000	NA	NA

```r
set.seed(123)
ptab.env <- adonis2(com.ds ~ Small.rocks.. , data = l.dat,
                    strata = l.dat[, "Tree.pairs"],
                    by = "term", nperm = 100000)
kable(ptab.env)
```

	Df	SumOfSqs	R2	F	Pr(>F)
Small.rocks..	1	0.2204425	0.0103156	0.604541	0.782
Residual	58	21.1493794	0.9896844	NA	NA
Total	59	21.3698219	1.0000000	NA	NA

```r
set.seed(123)
ptab.env <- adonis2(com.ds ~ Litter.. , data = l.dat,
                    strata = l.dat[, "Tree.pairs"],
                    by = "term", nperm = 100000)
kable(ptab.env)
```

	Df	SumOfSqs	R2	F	Pr(>F)
Litter..	1	1.714256	0.0802185	5.058457	0.002
Residual	58	19.655566	0.9197815	NA	NA
Total	59	21.369822	1.0000000	NA	NA

```r
pdf("./results/scrl_litter_light_rocks.pdf", width = 10, height = 5)
par(mfrow = c(1,2))
plot(Litter.. ~ Light...average, data = env.dif,
     xlab = "Light difference (S - R)", ylab = "Litter difference (S - R)",
     pch = 19, cex = 1.5)
plot(total.rocks ~ Light...average, data = env.dif,
     xlab = "Light difference (S - R)", ylab = "Rock cover difference (S - R)",
     pch = 19, cex = 1.5)
dev.off()
```

```
## pdf
##   2
```

```r
xtable(summary(lm(Litter.. ~ Light...average, data = data.frame(env.dif))))
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:09 2021

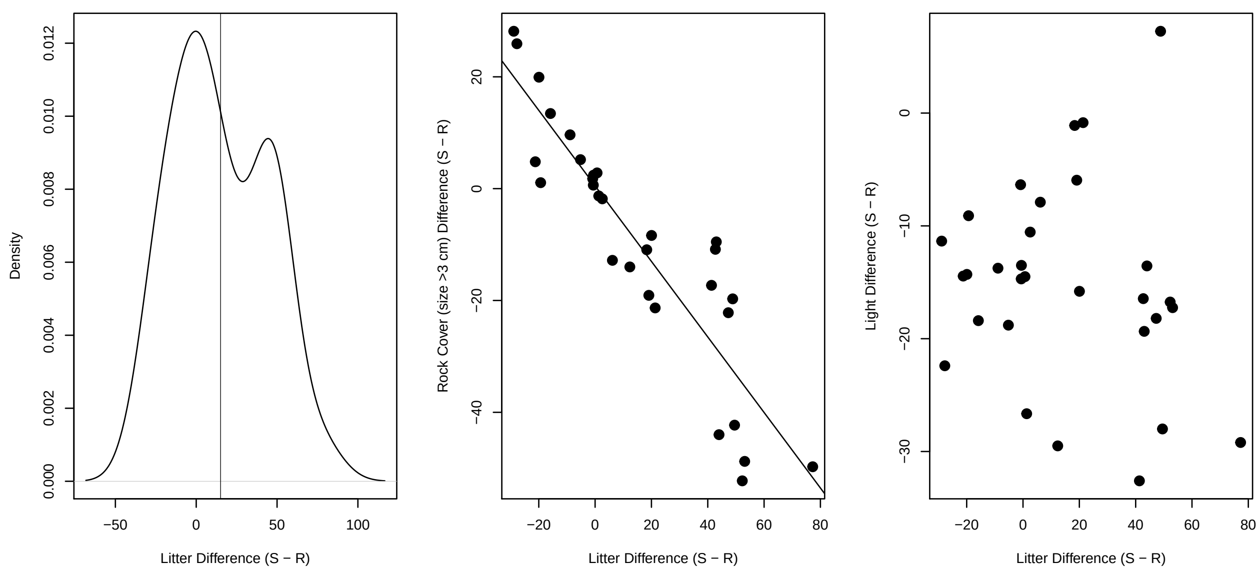
	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	6.7149	10.4959	0.64	0.5275
Light...average	-0.5521	0.5998	-0.92	0.3652

```r
xtable((lm(total.rocks ~ Light...average, data = data.frame(env.dif))))
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:09 2021

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-5.1978	10.3373	-0.50	0.6190
Light...average	0.6252	0.5907	1.06	0.2989

```r
par(mfrow = c(1,3))
plot(density(tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff)),
    main = "", xlab = "Litter Difference (S - R)")
abline(v = mean(tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff)),
    lwd = 0.5)
plot(env.dif[, "Big.rocks.."] ~ env.dif[, "Litter.."],
     xlab = "Litter Difference (S - R)", ylab = "Rock Cover (size >3 cm) Difference (S - R)",
     pch = 19, cex = 1.5)
abline(lm(env.dif[, "Big.rocks.."] ~ env.dif[, "Litter.."]))
plot(tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff),
    tapply(l.dat[, "Light...average"], l.dat[, "Tree.pairs"], diff),
    xlab = "Litter Difference (S - R)", ylab = "Light Difference (S - R)",
    pch = 19, cex = 1.5)
```




```r
pdf("./results/scrl_litter_effects.pdf", width = 10, height = 5)
par(mfrow = c(1,3))
plot(density(tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff)),
    main = "", xlab = "Litter Difference (S - R)")
abline(v = mean(tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff)),
    lwd = 0.5)
plot(env.dif[, "Big.rocks.."] ~ env.dif[, "Litter.."],
     xlab = "Litter Difference (S - R)", ylab = "Rock Cover (size >3 cm) Difference (S - R)",
     pch = 19, cex = 1.5)
abline(lm(env.dif[, "Big.rocks.."] ~ env.dif[, "Litter.."]))
plot(tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff),
    tapply(l.dat[, "Light...average"], l.dat[, "Tree.pairs"], diff),
    xlab = "Litter Difference (S - R)", ylab = "Light Difference (S - R)",
    pch = 19, cex = 1.5)
dev.off()
```

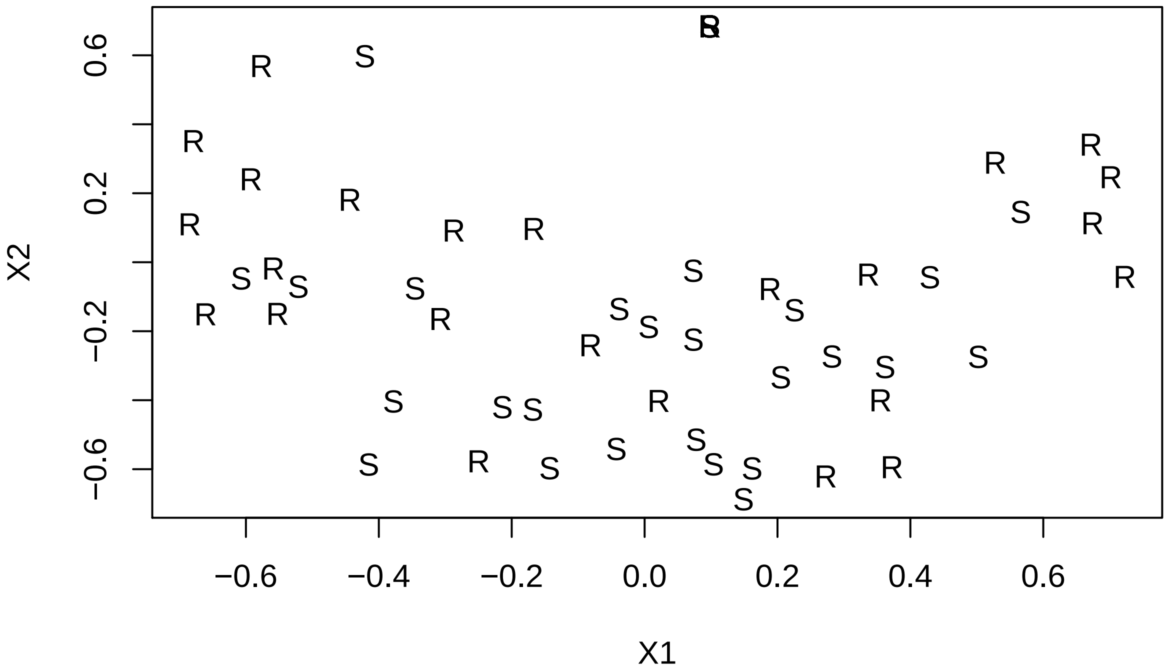
```
## pdf
##   2
```

```r
nmds.out <- nmds(vegdist(com.ds), 2, 2)
ord <- nmds.min(nmds.out, dims = 2)
```

```
## Minimum stress for given dimensionality:  0.2169355
```

```
## r^2 for minimum stress configuration:  0.6416469
```

```
ord.pch <- c("R", "S")[(l.dat[, "Moth"] + 1)]
plot(X2~ X1, data = ord, pch = ord.pch)
```



Litter not light was correlated with large rocks (dist cor, in text). Thus, higher amounts of litter under trees was not related to the penetration of light under the tree canopy.

```
cor.test(tapply(l.dat[, "Big.rocks.."], l.dat[, "Tree.pairs"], diff),
         tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff))
```

```
##
##  Pearson's product-moment correlation
##
## data:  tapply(l.dat[, "Big.rocks.."], l.dat[, "Tree.pairs"], diff) and tapply(l.dat[, "Litter.."], l
## t = -11.106, df = 28, p-value = 9.054e-12
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.9530598 -0.8039735
## sample estimates:
##        cor
## -0.9027609
```

```
cor.test(tapply(l.dat[, "Big.rocks.."], l.dat[, "Tree.pairs"], diff),
         tapply(l.dat[, "Light...average"], l.dat[, "Tree.pairs"], diff))
```

```
##
##  Pearson's product-moment correlation
##
## data:  tapply(l.dat[, "Big.rocks.."], l.dat[, "Tree.pairs"], diff) and tapply(l.dat[, "Light...averag
## t = 0.71624, df = 28, p-value = 0.4798
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2376184  0.4716125
## sample estimates:
```

```
##       cor
## 0.1341335
```

```
cor.test(tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff),
         tapply(l.dat[, "Light...average"], l.dat[, "Tree.pairs"], diff))
```

```
##
##  Pearson's product-moment correlation
##
## data:  tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff) and tapply(l.dat[, "Light...average"]
## t = -0.92053, df = 28, p-value = 0.3652
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.5007401  0.2013096
## sample estimates:
##        cor
## -0.1713898
```

```
cor.test(tapply(l.dat[, "Small.rocks.."], l.dat[, "Tree.pairs"], diff),
         tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff))
```

```
##
##  Pearson's product-moment correlation
##
## data:  tapply(l.dat[, "Small.rocks.."], l.dat[, "Tree.pairs"], diff) and tapply(l.dat[, "Litter.."],
## t = -4.994, df = 28, p-value = 2.819e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.8391386 -0.4332285
## sample estimates:
##        cor
## -0.6863699
```

Vegetation Analysis

Results Summary

- Both vegetation and light from the plant dataset respond to moth susceptibility (see t-tests below)
- Plant cover, richness and Shannon's diversity respond to moth susceptibility (see t-tests below)
- Plant community composition using Bray-Curtis dissimiliarity and a PERMANOVA model that accounts for tree pairs is significantly affected by moth susceptibility (Tables 11-12)
- Using the light, littter and rock cover from the saxicole dataset, plant community composition is significantly correlated with light and litter but not rock cover. Light has a strong effect but the effect of litter is weak and is non-significant after controlling for the effect of light, suggesting that the effect of litter is due to the covariance between light and litter (Tables 13-16)
- Two main species of plant were indicators of moth susceptibility: Apache plume and *Asteraceae ovales*. Both showed reduced cover under moth susceptible trees (Table 17)
- Saxicole and plant communities were not multivariately correlated based on Mantel Tests on both un-relatvized and species max relativized cover (see Mantel Test below)

From Richard Michalet

> First sheet is the vegetation matrix with all relevés.
>
> Second sheet are values of vegetation cover, rock cover and species richness in all replicates of all treatments + mean values of treatments and corresponding graphs.

From what I remember the methods were simple, quadrats of 1square meter in four treatments with a full factorial design, exposure (north and south of the tree), mortality (alive vs dead shrubs), tree susceptibility (resistant vs susceptible) and tree presence (below the canopy or outside the canopy in open conditions at the close vicinity of the trees).

You can see that without stats results are obvious: strong effect of tree susceptibility only below the tree and in both exposure for both alive and dead trees.

```r
veg <- readxl::read_xlsx("data/Vegetation.xlsx")
veg <- as.data.frame(veg)
l.raw <- read.csv("data/rawdata Sunset Crater for Matt.csv")
l.raw <- l.raw[!(grepl("cover", l.raw[,1])),]
le.raw <- read.csv("data/rawdata Sunset Crater for Matt_env.csv")
le.raw <- le.raw[!(grepl("cover", le.raw[,1])),]
le.raw <- na.omit(le.raw)
```

Observation checks

Do the saxicole community and environment data match?

```
## [1] TRUE
```

Are all of the trees in the saxicole dataset represented in the veg dataset?

```
## [1] TRUE
```

Coalesce datasets

```r
l.d <- data.frame(le.raw[, -2:-3], l.raw[, -1:-3])
l.d <- split(l.d, l.d[, "Tree.ID"])
l.d <- l.d[names(l.d) %in% le.raw[, "Tree.ID"]]
l.d <- lapply(l.d, function(x) x[, -1])
l.d <- lapply(l.d, apply, 2, mean)
l.df <- do.call(rbind, l.d)
trt <- strsplit(rownames(l.df), "")
moth.alive <- lapply(trt, function(x) x[x %in% c(letters, LETTERS)][1:2])
moth.alive <- do.call(rbind, moth.alive)
tree <- lapply(trt, function(x) x[x %in% 0:9])
tree <- as.numeric(unlist(lapply(tree, paste, collapse = "")))
l.df <- data.frame(Tree.pairs = tree,
                   Moth = moth.alive[, 1],
                   Live.Dead = moth.alive[, 2],
                   l.df)
l.df <- l.df[l.df[, "Live.Dead"] == "A", ]
l.df[, "Moth"] <- as.character(l.df[, "Moth"])
l.df[l.df[, "Moth"] == "R", "Moth"] <- 1
l.df[l.df[, "Moth"] == "S", "Moth"] <- 0
moth.tree <- paste(l.df[, "Moth"], l.df[, "Tree.pairs"], sep = "_")
l.df <- l.df[match(rownames(l.dat), moth.tree), ]
```

Check that `l.dat` and `l.df` are correctly coalesced:

```
## [1] TRUE
```

```
##  [1]  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE
## [13]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## [25]  TRUE  TRUE
```

Check that the values of the variables match, excluding light:

The following vector should work to match-up the saxicoles with the veg data:

Checking the vegetation and rock cover correlations. We find that vegetation cover is is significantly, but not strongly correlated with rock cover. Large rock cover measurements in the saxicole dataset is strongly correlated with total rock cover in the plant dataset.

Both vegetation and rock cover are strongly affected by moth susceptibility.

```
cor.test(v.dat[, "Vegetation.cover"], v.dat[, "Rock.cover"], alt = "greater")
```

```
##
##  Pearson's product-moment correlation
##
## data:  v.dat[, "Vegetation.cover"] and v.dat[, "Rock.cover"]
## t = 1.8835, df = 58, p-value = 0.03233
## alternative hypothesis: true correlation is greater than 0
## 95 percent confidence interval:
##  0.0269872 1.0000000
## sample estimates:
##       cor
## 0.2400809
```

```
cor.test(l.dat[, "Big.rocks.."], v.dat[, "Rock.cover"], alt = "greater")
```

```
##
##  Pearson's product-moment correlation
##
## data:  l.dat[, "Big.rocks.."] and v.dat[, "Rock.cover"]
## t = 9.5342, df = 58, p-value = 8.816e-14
## alternative hypothesis: true correlation is greater than 0
## 95 percent confidence interval:
##  0.6809688 1.0000000
## sample estimates:
##       cor
## 0.7813334
```

```
t.test(tapply(v.dat[, "Rock.cover"], v.dat[, "Tree.Pair"], diff))
```

```
##
##  One Sample t-test
##
## data:  tapply(v.dat[, "Rock.cover"], v.dat[, "Tree.Pair"], diff)
## t = -3.3582, df = 29, p-value = 0.002208
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -27.621617  -6.711716
## sample estimates:
## mean of x
## -17.16667
```

```
t.test(tapply(v.dat[, "Vegetation.cover"], v.dat[, "Tree.Pair"], diff))
```

```
##
##  One Sample t-test
##
## data:  tapply(v.dat[, "Vegetation.cover"], v.dat[, "Tree.Pair"], diff)
```

```
## t = -7.2026, df = 29, p-value = 6.269e-08
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -28.67505 -15.99162
## sample estimates:
## mean of x
## -22.33333
```

Both plant richness and Shannon's Diversity index were significantly affected by moth susceptibility.

```
v.abun <- v.dat[, "Vegetation.cover"]
v.rich <- apply(v.com, 1, function(x) sum(sign(x)))
v.shan <- apply(v.com, 1, diversity)

t.test(tapply(v.rich, l.dat[, "Tree.pairs"], diff))
```

```
##
##  One Sample t-test
##
## data:  tapply(v.rich, l.dat[, "Tree.pairs"], diff)
## t = -7.477, df = 29, p-value = 3.062e-08
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -1.6555988 -0.9444012
## sample estimates:
## mean of x
##      -1.3
```

```
t.test(tapply(v.shan, l.dat[, "Tree.pairs"], diff))
```

```
##
##  One Sample t-test
##
## data:  tapply(v.shan, l.dat[, "Tree.pairs"], diff)
## t = -4.2192, df = 29, p-value = 0.00022
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.4386895 -0.1522394
## sample estimates:
##  mean of x
## -0.2954645
```

```
v.ard <- rbind(tapply(v.dat[, "Vegetation.cover"], l.dat[, "Moth"], mean),
                tapply(rich, l.dat[, "Moth"], mean),
                tapply(shan, l.dat[, "Moth"], mean))
v.ard <- rbind(tapply(v.dat[, "Vegetation.cover"], l.dat[, "Moth"], se),
                tapply(rich, l.dat[, "Moth"], se),
                tapply(shan, l.dat[, "Moth"], se))
v.ard.tab <- cbind(v.ard[, "0"], v.ard[, "0"],
                   v.ard[, "1"], v.ard[, "1"])
colnames(v.ard.tab) <- c("Susceptible Mean", "Susceptible SE",
                         "Resistant Mean", "Resistant SE")
rownames(v.ard.tab) <- c("Abundance", "Richness", "Diversity (Shannon)")
```

```
kable(v.ard.tab, digits = 3)
```

	Susceptible Mean	Susceptible SE	Resistant Mean	Resistant SE
Abundance	1.511	1.511	2.758	2.758
Richness	0.542	0.542	0.662	0.662
Diversity (Shannon)	0.119	0.119	0.125	0.125

This is a multivariate analysis of the plant community response to moth susceptibility (PERMANOVA). This analysis uses a modified Bray-Curtis Dissimilarity metric, which permits the inclusion of quadrats that had no plants in them. The analysis also accounts for the paired structure of the data (i.e. pairs of moth susceptible and resistant trees).

```
set.seed(123)
ptab.v.moth <- adonis2(v.com.ds ~ Moth, data = l.dat,
                       strata = v.dat[, "Tree.pairs"],
                       by = "margin", nperm = 100000)
set.seed(123)
ptab.v.moth.rel <- adonis2(v.com.ds.rel ~ Moth, data = l.dat,
                           strata = v.dat[, "Tree.pairs"],
                           by = "margin", nperm = 100000)
```

Here are the results of the multivariate plant community response.

```
kable(ptab.v.moth, caption = "PERMANOVA of plant community response to moth.")
```

Table 15: PERMANOVA of plant community response to moth.

	Df	SumOfSqs	R2	F	Pr(>F)
Moth	1	5.174376	0.3081168	25.82917	0.001
Residual	58	11.619181	0.6918832	NA	NA
Total	59	16.793557	1.0000000	NA	NA

Here are the results of the multivariate plant community response after relativizing by species max.

Table 16: PERMANOVA of relativized plant community response to moth.

	Df	SumOfSqs	R2	F	Pr(>F)
Moth	1	5.989174	0.288048	23.46617	0.001
Residual	58	14.803100	0.711952	NA	NA
Total	59	20.792275	1.000000	NA	NA

Do light, litter or rock cover influence plant communities?

```
set.seed(123)
ptab.v.env <- adonis2(v.com.ds ~ Light...average + Litter.. + Big.rocks..,
                      data = l.dat,
                      strata = l.dat[, "Tree.pairs"],
                      by = "margin", nperm = 100000)
set.seed(123)
ptab.v.env.total.rocks <- adonis2(v.com.ds ~ Light...average + Litter.. + total.rocks,
                      data = l.dat,
                      strata = l.dat[, "Tree.pairs"],
```

```
                      by = "margin", nperm = 100000)
set.seed(123)
ptab.v.env.rel <- adonis2(v.com.ds.rel ~ Light...average + Litter.. + total.rocks,
                          data = l.dat,
                          strata = l.dat[, "Tree.pairs"],
                          by = "margin", nperm = 100000)
set.seed(123)
ptab.v.env.int <- adonis2(v.com.ds ~ Light...average + Litter.. + total.rocks +
                              Light...average * Litter.. +
                              Light...average * total.rocks +
                              Litter.. * total.rocks,
                          data = l.dat,
                          strata = l.dat[, "Tree.pairs"],
                          by = "margin", nperm = 100000)
set.seed(123)
ptab.v.env.rel.int <- adonis2(v.com.ds.rel ~ Light...average + Litter.. + total.rocks +
                              Light...average * Litter.. +
                              Light...average * total.rocks +
                              Litter.. * total.rocks,
                          data = l.dat,
                          strata = l.dat[, "Tree.pairs"],
                          by = "margin", nperm = 100000)
```

Light has a strong effect on the plant community. Litter also has an effect but it is small and marginally significant, either un-relativized or relativized, respectively.

Table 17: PERMANOVA of plant community response to several environmental variables.

	Df	SumOfSqs	R2	F	Pr(>F)
Light...average	1	2.8692870	0.1708564	12.696810	0.001
Litter..	1	0.6890028	0.0410278	3.048889	0.049
Big.rocks..	1	0.3621592	0.0215654	1.602582	0.189
Residual	56	12.6551530	0.7535719	NA	NA
Total	59	16.7935571	1.0000000	NA	NA

Table 18: PERMANOVA of relativized plant community response to several environmental variables.

	Df	SumOfSqs	R2	F	Pr(>F)
Light...average	1	3.4724258	0.1670056	12.245941	0.001
Litter..	1	0.3437323	0.0165317	1.212215	0.291
total.rocks	1	0.3501066	0.0168383	1.234694	0.282
Residual	56	15.8792084	0.7637071	NA	NA
Total	59	20.7922745	1.0000000	NA	NA

```
set.seed(123)
ptab.v.env.seq <- adonis2(v.com.ds ~ Light...average + Litter.. + total.rocks,
                      data = l.dat,
                      strata = l.dat[, "Tree.pairs"],
                      by = "term", nperm = 100000)
```

```r
set.seed(123)
ptab.v.env.rel.seq <- adonis2(v.com.ds.rel ~ Light...average + Litter.. + total.rocks,
                          data = l.dat,
                          strata = l.dat[, "Tree.pairs"],
                          by = "term", nperm = 100000)
```

After controlling for the effect of light, the effect of litter is no longer significant, un-relatvizied or relativized, respectivley.

Table 19: Sequential PERMANOVA of plant community response to several environmental variables. Variance is explained sequentially by factors entered into the model from top to bottom.

	Df	SumOfSqs	R2	F	Pr(>F)
Light...average	1	3.2765116	0.1951053	14.567808	0.001
Litter..	1	0.4997333	0.0297574	2.221881	0.098
total.rocks	1	0.4220991	0.0251346	1.876709	0.128
Residual	56	12.5952131	0.7500027	NA	NA
Total	59	16.7935571	1.0000000	NA	NA

Table 20: Sequential PERMANOVA of relativized plant community response to several environmental variables. Variance is explained sequentially by factors entered into the model from top to bottom.

	Df	SumOfSqs	R2	F	Pr(>F)
Light...average	1	3.8762571	0.1864278	13.670102	0.001
Litter..	1	0.6867025	0.0330268	2.421742	0.060
total.rocks	1	0.3501066	0.0168383	1.234694	0.282
Residual	56	15.8792084	0.7637071	NA	NA
Total	59	20.7922745	1.0000000	NA	NA

- Indicator species

```
## Warning in apply(do.call(rbind, lapply(ind.spp.v, unlist)), 2, as.numeric): NAs
## introduced by coercion

## Warning in apply(do.call(rbind, lapply(ind.spp.v, unlist)), 2, as.numeric): NAs
## introduced by coercion

## Warning in apply(do.call(rbind, lapply(ind.spp.v, unlist)), 2, as.numeric): NAs
## introduced by coercion
```

There are two species that are responding to moth susceptibility, Apache plume and *Asteraceae ovales*.

Table 21: Indicator Species Analysis using False Discovery Rate (FDR) adjusted p-values from t-tests of paired differences between resistant and susceptible trees (Resistant - Susceptible).

	t	df	p-value	Mean Difference	Lower CI 95%	Upper CI 95%
Apache.plume	-4.6010	29	0.0007	-10.2667	-14.8304	-5.7029
Asteraceae.ovales	-3.9581	29	0.0020	-8.1333	-12.3360	-3.9307
Rhus.trilobata	-1.8410	29	0.1869	-3.1667	-6.6847	0.3514

	t	df	p-value	Mean Difference	Lower CI 95%	Upper CI 95%
Rabbit.brush	-1.0000	29	0.3256	-0.6667	-2.0302	0.6968
Avena	-1.7951	29	0.1869	-0.2000	-0.4279	0.0279
Juniperus.monosperma	-1.0000	29	0.3256	-0.1667	-0.5075	0.1742
Plante.grise.allongée	-1.0000	29	0.3256	-0.1000	-0.3045	0.1045
Scarlet.glia	-1.0000	29	0.3256	-0.0667	-0.2030	0.0697
Bouteloua.gracilis	NaN	29	NaN	0.0000	NaN	NaN
Pinus.edulis.S	NaN	29	NaN	0.0000	NaN	NaN
Stipa.A	NaN	29	NaN	0.0000	NaN	NaN
Stipa.B	NaN	29	NaN	0.0000	NaN	NaN
Stipa.très.grand	NaN	29	NaN	0.0000	NaN	NaN
Ephedra	NaN	29	NaN	0.0000	NaN	NaN
Grande.grass.corymbe	NaN	29	NaN	0.0000	NaN	NaN
Boraginacée.rosette.grise	NaN	29	NaN	0.0000	NaN	NaN
Grass.à.nœud	NaN	29	NaN	0.0000	NaN	NaN
Brachypode	NaN	29	NaN	0.0000	NaN	NaN
Carex	NaN	29	NaN	0.0000	NaN	NaN
Cactus	NaN	29	NaN	0.0000	NaN	NaN
Hordeum	NaN	29	NaN	0.0000	NaN	NaN
Chenopodiaceae	NaN	29	NaN	0.0000	NaN	NaN
Ribes	NaN	29	NaN	0.0000	NaN	NaN
Aster.grise	NaN	29	NaN	0.0000	NaN	NaN
Rosette.frisée	NaN	29	NaN	0.0000	NaN	NaN
Chamaephyte.gris	NaN	29	NaN	0.0000	NaN	NaN
Castilleja	NaN	29	NaN	0.0000	NaN	NaN
Opuntia	NaN	29	NaN	0.0000	NaN	NaN
Rubiaceae	NaN	29	NaN	0.0000	NaN	NaN
Andropogon	NaN	29	NaN	0.0000	NaN	NaN
Pinus.edulis.R	1.0000	29	0.3256	0.3333	-0.3484	1.0151

```r
v.isp.dat <- melt(d.v.isp)
colnames(v.isp.dat) <- c("Tree.pairs", "Species", "diff")
v.isp.mu <- tapply(v.isp.dat[, "diff"], v.isp.dat[, "Species"], mean)
v.isp.se <- tapply(v.isp.dat[, "diff"], v.isp.dat[, "Species"], se)
v.ard <- t(apply(v.com, 1, function(x) c(A = sum(x),
                                     R = sum(sign(x)),
                                     D = diversity(x))))
v.ard.dif <- apply(v.ard, 2,
                   function(x, p)  tapply(x, p, diff),
                   p = l.dat[, "Tree.pairs"])
colnames(v.ard.dif) <- c("Abundance", "Richness", "Diversity")
v.ard.dif <- apply(v.ard.dif, 2, function(x) x / max(abs(x)))
v.ard.dat <- melt(v.ard.dif)
colnames(v.ard.dat) <- c("Tree.pairs", "Stat", "diff")
v.ard.mu <- tapply(v.ard.dat[, "diff"], v.ard.dat[, "Stat"], mean)
v.ard.se <- tapply(v.ard.dat[, "diff"], v.ard.dat[, "Stat"], se)

pdf(file = "./results/plot_isp_ard_plant.pdf", width = 9, height = 5)

par(mfrow = c(1,2))
bp.out <- barplot(v.ard.mu, col = "darkgrey", ylim = c(-1.0, 0),
                  ylab = "Relativized Difference (S - R)", border = "NA")
```

```r
segments(bp.out[, 1], v.ard.mu + v.ard.se,
         bp.out[, 1], v.ard.mu - v.ard.se,
         lwd = 1.5)
bp.out <- barplot(v.isp.mu, col = "darkgrey", ylim = c(-13, 0),
                  ylab = "Difference (S - R)", border = "NA",
            axisnames = TRUE,
            names.arg = sapply(names(v.isp.mu),
                function(x)
                                    paste(c(substr(x, 1, 1),
                                            substr(x, 4, 4)), collapse = "")))
segments(bp.out[, 1], v.isp.mu + v.isp.se,
         bp.out[, 1], v.isp.mu - v.isp.se,
         lwd = 1.5)
dev.off()
```

```
## pdf
##   2
```

Multivariate Correlation of Plants and Saxicoles

There is no significant multivariate correlation between the veg and saxicole communities, regardless of whether the community data are relativized. This is likely a result of the two communities responded to different variables with low correlation (i.e. rocks = saxicoles and light = plants). This was true either without or with relativization by species max.

```r
v.d <- vegdist(v.com.ds)
l.d <- vegdist(com.ds)

mantel(v.d ~ l.d)
```

```
##      mantelr        pval1        pval2        pval3    llim.2.5%   ulim.97.5%
## -0.002762319  0.513000000  0.488000000  0.914000000 -0.034504235  0.032707393
```

```r
v.d <- vegdist(v.com.ds.rel)
l.d <- vegdist(com.ds.rel)

mantel(v.d ~ l.d)
```

```
##     mantelr       pval1       pval2       pval3   llim.2.5%  ulim.97.5%
##  0.02328021  0.21200000  0.78900000  0.44300000 -0.01176642  0.05838093
```

Structural Equation Modeling

```r
com.prepared <- cbind(id = l.dat[, "Moth"], tree  = l.dat[, "Tree.pairs"], com)
v.com.prepared <- cbind(id = l.dat[, "Moth"], tree  = l.dat[, "Tree.pairs"], v.com)

l.dist.euc <- distancePairedSamples(
    sequences = com.prepared,
    grouping.column = "id",
    time.column = "tree",
    exclude.columns = NULL,
    method = "euclidean",
    sum.distances = FALSE,
    parallel.execution = FALSE
```

```
)

l.dist.man <- distancePairedSamples(
    sequences = com.prepared,
    grouping.column = "id",
    time.column = "tree",
    exclude.columns = NULL,
    method = "manhattan",
    sum.distances = FALSE,
    parallel.execution = FALSE
)

v.dist.euc <- distancePairedSamples(
    sequences = v.com.prepared,
    grouping.column = "id",
    time.column = "tree",
    exclude.columns = NULL,
    method = "euclidean",
    sum.distances = FALSE,
    parallel.execution = FALSE
)

v.dist.man <- distancePairedSamples(
    sequences = v.com.prepared,
    grouping.column = "id",
    time.column = "tree",
    exclude.columns = NULL,
    method = "manhattan",
    sum.distances = FALSE,
    parallel.execution = FALSE
)

cor(l.dist.man[[1]], l.dist.euc[[1]])
```

```
## [1] 0.9422796
```

```
cor(v.dist.man[[1]], v.dist.euc[[1]])
```

```
## [1] 0.9612754
```

```
d.litter <- tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff)
d.rocks <- tapply((l.dat[, "Big.rocks.."] + l.dat[, "Small.rocks.."]),
                  l.dat[, "Tree.pairs"], diff)
d.light <- tapply(l.dat[, "Light...average"], l.dat[, "Tree.pairs"], diff)
d.com <- l.dist.man[[1]]
d.abun <- tapply(abun, l.dat[, "Tree.pairs"], diff)
d.rich <- tapply(rich, l.dat[, "Tree.pairs"], diff)
d.shan <- tapply(shan, l.dat[, "Tree.pairs"], diff)
d.isp <- apply(isp.com, 2, function(x, f) tapply(x, f, diff), f = l.dat[, "Tree.pairs"])
colnames(d.isp) <- paste("d", colnames(isp.com), sep = ".")

round(cor(cbind(d.litter, d.rocks, d.light, d.abun, d.rich, d.shan, d.com)), 3)
```

```
##          d.litter d.rocks d.light d.abun d.rich d.shan  d.com
## d.litter    1.000  -0.998  -0.171 -0.530 -0.695 -0.651  0.154
```

```
## d.rocks    -0.998   1.000   0.196  0.513  0.694  0.656 -0.140
## d.light    -0.171   0.196   1.000  0.108  0.268  0.290 -0.133
## d.abun     -0.530   0.513   0.108  1.000  0.649  0.353 -0.448
## d.rich     -0.695   0.694   0.268  0.649  1.000  0.888 -0.143
## d.shan     -0.651   0.656   0.290  0.353  0.888  1.000 -0.071
## d.com       0.154  -0.140  -0.133 -0.448 -0.143 -0.071  1.000
```

```r
sem.dat <- data.frame(d.litter, d.rocks, d.light, d.abun, d.rich, d.shan, d.com, d.isp)
sem.path <- matrix(c(0, 1, 1, 0,
                     1, 0, 0, 1,
                     0, 0, 0, 1,
                     0, 0, 0, 0), 4, 4, byrow = TRUE)
rownames(sem.path) <- colnames(sem.path) <- c("d.litter", "d.light", "d.rocks", "d.com")

model.com  <- psem(lm(d.rocks ~ d.litter, sem.dat), lm(d.com ~ d.light + d.rocks, sem.dat))
model.com1  <- psem(lm(d.rocks ~ d.litter, sem.dat), lm(d.com ~ d.litter + d.light + d.rocks, sem.dat))
model.abun <- psem(lm(d.rocks ~ d.litter, sem.dat), lm(d.abun ~ d.light + d.rocks, sem.dat))
model.rich <- psem(lm(d.rocks ~ d.litter, sem.dat), lm(d.rich ~ d.light + d.rocks, sem.dat))
model.shan <- psem(lm(d.rocks ~ d.litter, sem.dat), lm(d.shan ~ d.light + d.rocks, sem.dat))
model.Acacon <- psem(lm(d.rocks ~ d.litter, sem.dat), lm(d.Acacon ~ d.light + d.rocks, sem.dat))
model.Acaame <- psem(lm(d.rocks ~ d.litter, sem.dat), lm(d.Acaame ~ d.light + d.rocks, sem.dat))
model.Canros <- psem(lm(d.rocks ~ d.litter, sem.dat), lm(d.Canros ~ d.light + d.rocks, sem.dat))
model.Canros1 <- psem(lm(d.rocks ~ d.litter, sem.dat),  lm(d.Canros ~ d.light + d.rocks, sem.dat))
```

```r
d.litter <- tapply(l.dat[, "Litter.."], l.dat[, "Tree.pairs"], diff)
d.rocks <- tapply((l.dat[, "Big.rocks.."] + l.dat[, "Small.rocks.."]),
                  l.dat[, "Tree.pairs"], diff)
d.light <- tapply(l.dat[, "Light...average"], l.dat[, "Tree.pairs"], diff)

d.v.com <- v.dist.man[[1]]
d.v.abun <- tapply(v.abun, l.dat[, "Tree.pairs"], diff)
d.v.rich <- tapply(v.rich, l.dat[, "Tree.pairs"], diff)
d.v.shan <- tapply(v.shan, l.dat[, "Tree.pairs"], diff)
d.v.isp <- apply(v.isp.com, 2, function(x, f) tapply(x, f, diff), f = l.dat[, "Tree.pairs"])
colnames(d.v.isp) <- paste("d", colnames(v.isp.com), sep = ".")
v.sem.dat <- data.frame(d.litter, d.rocks, d.light, d.v.abun, d.v.rich, d.v.shan, d.v.com, d.v.isp)

model.v.com  <- psem(lm(d.rocks ~ d.litter, v.sem.dat), lm(d.v.com ~ d.light + d.rocks, v.sem.dat))
model.v.com1  <- psem(lm(d.rocks ~ d.litter, v.sem.dat), lm(d.v.com ~ d.litter + d.light + d.rocks, v.se
model.v.abun <- psem(lm(d.rocks ~ d.litter, v.sem.dat), lm(d.v.abun ~ d.light + d.rocks, v.sem.dat))
model.v.rich <- psem(lm(d.rocks ~ d.litter, v.sem.dat), lm(d.v.rich ~ d.light + d.rocks, v.sem.dat))
model.v.shan <- psem(lm(d.rocks ~ d.litter, v.sem.dat), lm(d.v.shan ~ d.light + d.rocks, v.sem.dat))
model.v.Apache.plume <- psem(lm(d.rocks ~ d.litter, v.sem.dat),
                             lm(d.Apache.plume ~ d.light + d.rocks, v.sem.dat))
model.v.Asteraceae.ovales <- psem(lm(d.rocks ~ d.litter, v.sem.dat),
                                  lm(d.Asteraceae.ovales ~ d.light + d.rocks, v.sem.dat))
```

Independent Test Method

Using indeendent tests for different effects along the hypothesized causal model that moth susceptibility affects tree traits (litter production), which affect the local environment (light, rocks), which in turn affect lichen, bryophyte and plant communities (abundance, richness, diversity, indicator species, composition).

moth-susceptibility -> tree traits -> local environment -> community

We can do this by parsing independent tests for each effect OR by using a structural equation model (SEM).

Testing for the effect of moth susceptibility:

```
t.test(d.litter)
```

```
##
##  One Sample t-test
##
## data:  d.litter
## t = 2.8665, df = 29, p-value = 0.00765
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##   4.317792 25.822208
## sample estimates:
## mean of x
##     15.07
```

```
t.test(d.light)
```

```
##
##  One Sample t-test
##
## data:  d.light
## t = -9.2728, df = 29, p-value = 3.557e-10
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -18.47119 -11.79547
## sample estimates:
## mean of x
## -15.13333
```

```
t.test(d.rocks)
```

```
##
##  One Sample t-test
##
## data:  d.rocks
## t = -2.8178, df = 29, p-value = 0.008617
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -25.298305  -4.019028
## sample estimates:
## mean of x
## -14.65867
```

Effects of tree traits on local environment and environment correlations:

```
cor.test(d.light, d.litter)
```

```
##
##  Pearson's product-moment correlation
##
## data:  d.light and d.litter
## t = -0.92053, df = 28, p-value = 0.3652
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.5007401  0.2013096
```

```
## sample estimates:
##        cor
## -0.1713898
```

```r
cor.test(d.rocks, d.light)
```

```
##
##  Pearson's product-moment correlation
##
## data:  d.rocks and d.light
## t = 1.0584, df = 28, p-value = 0.2989
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1766215  0.5196770
## sample estimates:
##       cor
## 0.1961275
```

```r
summary(lm(d.rocks ~ d.litter))
```

```
##
## Call:
## lm(formula = d.rocks ~ d.litter)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4466 -0.7468 -0.3273  0.2442  6.9590
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.22870    0.34616   0.661    0.514
## d.litter    -0.98788    0.01079 -91.529   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.674 on 28 degrees of freedom
## Multiple R-squared:  0.9967, Adjusted R-squared:  0.9965
## F-statistic:  8378 on 1 and 28 DF,  p-value: < 2.2e-16
```

Effects of local environment on lichen, and possible direct effects of tree traits:

```r
summary(lm(d.abun ~ d.rocks))
```

```
##
## Call:
## lm(formula = d.abun ~ d.rocks)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8587 -1.3596  0.5429  1.6415  5.8098
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.55053    0.67673  -0.814  0.42279
## d.rocks      0.06777    0.02140   3.166  0.00371 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 3.284 on 28 degrees of freedom
## Multiple R-squared:  0.2637, Adjusted R-squared:  0.2374
## F-statistic: 10.03 on 1 and 28 DF,  p-value: 0.003706
```

```r
summary(lm(d.rich ~ d.rocks))
```

```
##
## Call:
## lm(formula = d.rich ~ d.rocks)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7375 -2.3674 -0.1611  1.6950  7.5293
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.85626    0.70878  -1.208    0.237
## d.rocks      0.11441    0.02242   5.104 2.09e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.44 on 28 degrees of freedom
## Multiple R-squared:  0.4819, Adjusted R-squared:  0.4634
## F-statistic: 26.05 on 1 and 28 DF,  p-value: 2.089e-05
```

```r
summary(lm(d.shan ~ d.rocks))
```

```
##
## Call:
## lm(formula = d.shan ~ d.rocks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.46785 -0.60402  0.04559  0.63369  1.38124
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.106623   0.154747  -0.689    0.496
## d.rocks      0.022537   0.004894   4.605 8.17e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.751 on 28 degrees of freedom
## Multiple R-squared:  0.4309, Adjusted R-squared:  0.4106
## F-statistic:  21.2 on 1 and 28 DF,  p-value: 8.167e-05
```

```r
summary(lm(d.Acacon ~ d.rocks, sem.dat))
```

```
##
## Call:
## lm(formula = d.Acacon ~ d.rocks, data = sem.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.17556 -0.01439  0.01337  0.03252  0.09108
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0238762  0.0126055  -1.894  0.06858 .
## d.rocks      0.0014183  0.0003987   3.557  0.00136 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06117 on 28 degrees of freedom
## Multiple R-squared:  0.3113, Adjusted R-squared:  0.2867
## F-statistic: 12.66 on 1 and 28 DF,  p-value: 0.001357
```

```
summary(lm(d.Acaame ~ d.rocks, sem.dat))
```

```
##
## Call:
## lm(formula = d.Acaame ~ d.rocks, data = sem.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.64206 -0.09675  0.03298  0.07873  0.56715
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.068167   0.042641  -1.599    0.121
## d.rocks      0.006310   0.001349   4.679 6.67e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2069 on 28 degrees of freedom
## Multiple R-squared:  0.4388, Adjusted R-squared:  0.4188
## F-statistic: 21.89 on 1 and 28 DF,  p-value: 6.669e-05
```

```
summary(lm(d.Canros ~ d.rocks, sem.dat))
```

```
##
## Call:
## lm(formula = d.Canros ~ d.rocks, data = sem.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.04560 -0.22148  0.06461  0.28602  0.81105
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.196087   0.096385  -2.034 0.051479 .
## d.rocks      0.012797   0.003048   4.198 0.000247 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4678 on 28 degrees of freedom
## Multiple R-squared:  0.3863, Adjusted R-squared:  0.3643
## F-statistic: 17.62 on 1 and 28 DF,  p-value: 0.0002467
```

```
summary(lm(d.abun ~ d.light))
```

```
##
## Call:
## lm(formula = d.abun ~ d.light)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.3371 -2.7395  0.6687  1.5171  8.1163
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.85872    1.38331  -0.621    0.540
## d.light      0.04528    0.07905   0.573    0.571
##
## Residual standard error: 3.805 on 28 degrees of freedom
## Multiple R-squared:  0.01159,    Adjusted R-squared:  -0.02372
## F-statistic: 0.3282 on 1 and 28 DF,  p-value: 0.5713
```

```
summary(lm(d.rich ~ d.light))
```

```
##
## Call:
## lm(formula = d.rich ~ d.light)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.758 -3.199 -0.836  3.003 12.001
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.40551    1.67397  -0.242    0.810
## d.light      0.14061    0.09565   1.470    0.153
##
## Residual standard error: 4.605 on 28 degrees of freedom
## Multiple R-squared:  0.07164,    Adjusted R-squared:  0.03848
## F-statistic: 2.161 on 1 and 28 DF,  p-value: 0.1527
```

```
summary(lm(d.shan ~ d.light))
```

```
##
## Call:
## lm(formula = d.shan ~ d.light)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5927 -0.7784  0.1074  0.5385  2.1225
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.04306    0.34638   0.124    0.902
## d.light      0.03172    0.01979   1.603    0.120
##
## Residual standard error: 0.9528 on 28 degrees of freedom
## Multiple R-squared:  0.08402,    Adjusted R-squared:  0.05131
## F-statistic: 2.568 on 1 and 28 DF,  p-value: 0.1202
```

```r
summary(lm(d.Acacon ~ d.light, sem.dat))
```

```
##
## Call:
## lm(formula = d.Acacon ~ d.light, data = sem.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.21083 -0.02561  0.02198  0.04135  0.09381
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.007098   0.024294   0.292   0.7723
## d.light     0.003421   0.001388   2.464   0.0201 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06682 on 28 degrees of freedom
## Multiple R-squared:  0.1782, Adjusted R-squared:  0.1489
## F-statistic: 6.072 on 1 and 28 DF,  p-value: 0.02014
```

```r
summary(lm(d.Acaame ~ d.light, sem.dat))
```

```
##
## Call:
## lm(formula = d.Acaame ~ d.light, data = sem.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.85875 -0.06371  0.06088  0.15869  0.27225
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.03200    0.09117   0.351   0.7283
## d.light      0.01273    0.00521   2.444   0.0211 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2508 on 28 degrees of freedom
## Multiple R-squared:  0.1758, Adjusted R-squared:  0.1463
## F-statistic: 5.972 on 1 and 28 DF,  p-value: 0.0211
```

```r
summary(lm(d.Canros ~ d.light, sem.dat))
```

```
##
## Call:
## lm(formula = d.Canros ~ d.light, data = sem.dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9699 -0.3253  0.1547  0.3191  1.2307
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.03300    0.19704   0.168    0.868
```

```
## d.light      0.02753    0.01126   2.445    0.021 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.542 on 28 degrees of freedom
## Multiple R-squared:  0.176,  Adjusted R-squared:  0.1466
## F-statistic:  5.98 on 1 and 28 DF,  p-value: 0.02101
```

```
summary(lm(d.abun ~ d.litter))
```

```
##
## Call:
## lm(formula = d.abun ~ d.litter)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.380 -1.218  0.494  1.607  5.733
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.50153    0.67144  -0.747  0.46132
## d.litter    -0.06917    0.02094  -3.304  0.00261 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.246 on 28 degrees of freedom
## Multiple R-squared:  0.2805, Adjusted R-squared:  0.2548
## F-statistic: 10.92 on 1 and 28 DF,  p-value: 0.002612
```

```
summary(lm(d.rich ~ d.litter))
```

```
##
## Call:
## lm(formula = d.rich ~ d.litter)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7618 -2.0890 -0.0954  1.7166  7.5545
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.82616    0.71101  -1.162    0.255
## d.litter    -0.11328    0.02217  -5.110 2.05e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.438 on 28 degrees of freedom
## Multiple R-squared:  0.4826, Adjusted R-squared:  0.4641
## F-statistic: 26.11 on 1 and 28 DF,  p-value: 2.053e-05
```

```
summary(lm(d.shan ~ d.litter))
```

```
##
## Call:
## lm(formula = d.shan ~ d.litter)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.47085 -0.59769  0.03512  0.59650  1.39944
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.103513   0.156232  -0.663    0.513
## d.litter    -0.022128   0.004871  -4.543 9.68e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7554 on 28 degrees of freedom
## Multiple R-squared:  0.4243, Adjusted R-squared:  0.4037
## F-statistic: 20.64 on 1 and 28 DF,  p-value: 9.675e-05
```

```
summary(lm(d.Acacon ~ d.litter, sem.dat))
```

```
##
## Call:
## lm(formula = d.Acacon ~ d.litter, data = sem.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.17743 -0.01528  0.01435  0.03220  0.09098
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0240028  0.0127820  -1.878  0.07085 .
## d.litter    -0.0013712  0.0003985  -3.441  0.00184 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0618 on 28 degrees of freedom
## Multiple R-squared:  0.2971, Adjusted R-squared:  0.272
## F-statistic: 11.84 on 1 and 28 DF,  p-value: 0.001839
```

```
summary(lm(d.Acaame ~ d.litter, sem.dat))
```

```
##
## Call:
## lm(formula = d.Acaame ~ d.litter, data = sem.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.64969 -0.10426  0.03407  0.08146  0.56925
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.067611   0.043169  -1.566    0.129
## d.litter    -0.006175   0.001346  -4.588 8.56e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2087 on 28 degrees of freedom
## Multiple R-squared:  0.4291, Adjusted R-squared:  0.4087
```

```
## F-statistic: 21.05 on 1 and 28 DF,  p-value: 8.558e-05
```

```r
summary(lm(d.Canros ~ d.litter, sem.dat))
```

```
##
## Call:
## lm(formula = d.Canros ~ d.litter, data = sem.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.06651 -0.21741  0.05103  0.27634  0.81235
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.193646   0.097001  -1.996 0.055705 .
## d.litter    -0.012609   0.003024  -4.169 0.000267 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.469 on 28 degrees of freedom
## Multiple R-squared:  0.383,  Adjusted R-squared:  0.361
## F-statistic: 17.38 on 1 and 28 DF,  p-value: 0.0002666
```

SEM testing for this pathway, note that here community distance is the sum of squared differences for each tree pair (susceptible - resistant) for all species:

```r
summary(model.abun, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.abun
##
## Call:
##   d.rocks ~ d.litter
##   d.abun ~ d.light + d.rocks
##
##     AIC      BIC
##  28.447   38.255
##
## ---
## Tests of directed separation:
##
##            Independ.Claim Test.Type DF Crit.Value P.Value
##   d.abun ~ d.litter + ...      coef 26    -2.1260  0.0432 *
##   d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 14.447 with P-value = 0.006 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##     d.abun   d.light   0.0030    0.0709 27     0.0428  0.9662       0.0072
##     d.abun   d.rocks   0.0676    0.0222 27     3.0408  0.0052       0.5121  **
```

```
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##     d.abun   none      0.26
```

```r
summary(model.rich, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.rich
##
## Call:
##   d.rocks ~ d.litter
##   d.rich ~ d.light + d.rocks
##
##     AIC      BIC
##  23.564   33.372
##
## ---
## Tests of directed separation:
##
##            Independ.Claim Test.Type DF Crit.Value P.Value
##   d.rich ~ d.litter + ...      coef 26    -0.6906  0.4960
##   d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 9.564 with P-value = 0.048 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##     d.rich   d.light   0.0718    0.0729 27     0.9854  0.3332       0.1368
##     d.rich   d.rocks   0.1100    0.0229 27     4.8086  0.0001       0.6674 ***
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none       1.0
##     d.rich   none       0.5
```

```r
summary(model.shan, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.shan
##
```

```
## Call:
##   d.rocks ~ d.litter
##   d.shan ~ d.light + d.rocks
##
##     AIC      BIC
##  22.182   31.99
##
## ---
## Tests of directed separation:
##
##            Independ.Claim Test.Type DF Crit.Value P.Value
##   d.shan ~ d.litter + ...      coef 26    -0.0130  0.9897
##   d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 8.182 with P-value = 0.085 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##     d.shan   d.light   0.0183    0.0158 27     1.1596  0.2563       0.1676
##     d.shan   d.rocks   0.0214    0.0050 27     4.3156  0.0002       0.6236 ***
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##     d.shan   none      0.46
```

```
summary(model.com, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.com
##
## Call:
##   d.rocks ~ d.litter
##   d.com ~ d.light + d.rocks
##
##     AIC      BIC
##  27.066   36.874
##
## ---
## Tests of directed separation:
##
##            Independ.Claim Test.Type DF Crit.Value P.Value
##    d.com ~ d.litter + ...      coef 26     1.7840  0.0861
##   d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
```

```
##
##   Fisher's C = 13.066 with P-value = 0.011 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##      d.com   d.light  -0.0350    0.0617 27    -0.5673  0.5752      -0.1096
##      d.com   d.rocks  -0.0119    0.0193 27    -0.6129  0.5450      -0.1184
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##      d.com   none      0.03
```

```
summary(model.Acacon, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.Acacon
##
## Call:
##   d.rocks ~ d.litter
##   d.Acacon ~ d.light + d.rocks
##
##     AIC      BIC
##  23.133   32.941
##
## ---
## Tests of directed separation:
##
##              Independ.Claim Test.Type DF Crit.Value P.Value
##   d.Acacon ~ d.litter + ...      coef 26     0.5085  0.6154
##     d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 9.133 with P-value = 0.058 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##   d.Acacon   d.light   0.0026    0.0012 27     2.1628  0.0396       0.3252   *
##   d.Acacon   d.rocks   0.0013    0.0004 27     3.2863  0.0028       0.4941  **
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
```

```
##
##   Response method R.squared
##    d.rocks   none      1.00
##   d.Acacon   none      0.41
```

```r
summary(model.Acaame, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.Acaame
##
## Call:
##   d.rocks ~ d.litter
##   d.Acaame ~ d.light + d.rocks
##
##     AIC      BIC
##  22.423   32.231
##
## ---
## Tests of directed separation:
##
##              Independ.Claim Test.Type DF Crit.Value P.Value
##   d.Acaame ~ d.litter + ...      coef 26    -0.1558  0.8774
##     d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 8.423 with P-value = 0.077 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##   d.Acaame   d.light   0.0091    0.0041 27     2.2267  0.0345       0.3009   *
##   d.Acaame   d.rocks   0.0057    0.0013 27     4.4650  0.0001       0.6034 ***
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##   d.Acaame   none      0.53
```

```r
summary(model.Canros, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.Canros
##
## Call:
##   d.rocks ~ d.litter
##   d.Canros ~ d.light + d.rocks
##
##     AIC      BIC
```

```
##  23.898   33.706
##
## ---
## Tests of directed separation:
##
##              Independ.Claim Test.Type DF Crit.Value P.Value
##   d.Canros ~ d.litter + ...      coef 26    -0.8201  0.4196
##     d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 9.898 with P-value = 0.042 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##   d.Canros   d.light   0.0203    0.0093 27     2.1836  0.0379       0.3095   *
##   d.Canros   d.rocks   0.0115    0.0029 27     3.9562  0.0005       0.5608 ***
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##   d.Canros   none      0.48
```

```
summary(lm(d.v.abun ~ d.rocks))
```

```
##
## Call:
## lm(formula = d.v.abun ~ d.rocks)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.548  -9.167  -0.371  11.836  29.860
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -23.61098    3.52322  -6.702 2.83e-07 ***
## d.rocks      -0.08716    0.11143  -0.782    0.441
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.1 on 28 degrees of freedom
## Multiple R-squared:  0.02138,    Adjusted R-squared:  -0.01357
## F-statistic: 0.6118 on 1 and 28 DF,  p-value: 0.4407
```

```
summary(lm(d.v.rich ~ d.rocks))
```

```
##
## Call:
```

```
## lm(formula = d.v.rich ~ d.rocks)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6195 -0.7375  0.2342  0.3760  2.3148
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.259773   0.199030  -6.330 7.57e-07 ***
## d.rocks      0.002744   0.006295   0.436    0.666
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9659 on 28 degrees of freedom
## Multiple R-squared:  0.006742,   Adjusted R-squared:  -0.02873
## F-statistic: 0.1901 on 1 and 28 DF,  p-value: 0.6662
```

```r
summary(lm(d.v.shan ~ d.rocks))
```

```
##
## Call:
## lm(formula = d.v.shan ~ d.rocks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.63077 -0.28155  0.02544  0.29568  0.97384
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.335709   0.078745  -4.263 0.000207 ***
## d.rocks     -0.002745   0.002491  -1.102 0.279691
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3821 on 28 degrees of freedom
## Multiple R-squared:  0.04159,    Adjusted R-squared:  0.007366
## F-statistic: 1.215 on 1 and 28 DF,  p-value: 0.2797
```

```r
summary(lm(d.Apache.plume ~ d.rocks, v.sem.dat))
```

```
##
## Call:
## lm(formula = d.Apache.plume ~ d.rocks, data = v.sem.dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.028  -4.455   4.278   6.677  14.799
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.13756    2.44690  -4.960 3.09e-05 ***
## d.rocks      -0.12763    0.07739  -1.649     0.11
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 11.87 on 28 degrees of freedom
## Multiple R-squared:  0.08854,    Adjusted R-squared:  0.05598
## F-statistic:  2.72 on 1 and 28 DF,  p-value: 0.1103
```

```r
summary(lm(d.Asteraceae.ovales ~ d.rocks, v.sem.dat))
```

```
##
## Call:
## lm(formula = d.Asteraceae.ovales ~ d.rocks, data = v.sem.dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.976  -7.315   5.782   7.526  19.463
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.44665    2.34354  -3.178   0.0036 **
## d.rocks      0.04684    0.07412   0.632   0.5325
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.37 on 28 degrees of freedom
## Multiple R-squared:  0.01406,    Adjusted R-squared:  -0.02115
## F-statistic: 0.3994 on 1 and 28 DF,  p-value: 0.5325
```

```r
summary(lm(d.v.abun ~ d.litter))
```

```
##
## Call:
## lm(formula = d.v.abun ~ d.litter)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.743  -8.907   0.019  11.943  30.269
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -23.44568    3.54674  -6.610  3.6e-07 ***
## d.litter      0.07381    0.11059   0.667     0.51
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.15 on 28 degrees of freedom
## Multiple R-squared:  0.01566,    Adjusted R-squared:  -0.01949
## F-statistic: 0.4455 on 1 and 28 DF,  p-value: 0.5099
```

```r
summary(lm(d.v.rich ~ d.litter))
```

```
##
## Call:
## lm(formula = d.v.rich ~ d.litter)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6111 -0.7427  0.2214  0.3838  2.3153
##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.253709   0.199585  -6.282 8.61e-07 ***
## d.litter    -0.003072   0.006223  -0.494    0.625
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.965 on 28 degrees of freedom
## Multiple R-squared:  0.008626,   Adjusted R-squared:  -0.02678
## F-statistic: 0.2436 on 1 and 28 DF,  p-value: 0.6254
```

```r
summary(lm(d.v.shan ~ d.litter))
```

```
##
## Call:
## lm(formula = d.v.shan ~ d.litter)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.62023 -0.28853  0.04059  0.29668  0.97632
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.332721   0.079334  -4.194 0.000249 ***
## d.litter     0.002472   0.002474   0.999 0.326145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3836 on 28 degrees of freedom
## Multiple R-squared:  0.03444,    Adjusted R-squared:  -3.912e-05
## F-statistic: 0.9989 on 1 and 28 DF,  p-value: 0.3261
```

```r
summary(lm(d.Apache.plume ~ d.litter, v.sem.dat))
```

```
##
## Call:
## lm(formula = d.Apache.plume ~ d.litter, data = v.sem.dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.098  -4.465   4.364   6.975  14.577
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.05623    2.46985  -4.881 3.84e-05 ***
## d.litter      0.11875    0.07701   1.542    0.134
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.94 on 28 degrees of freedom
## Multiple R-squared:  0.07828,    Adjusted R-squared:  0.04536
## F-statistic: 2.378 on 1 and 28 DF,  p-value: 0.1343
```

```r
summary(lm(d.Asteraceae.ovales ~ d.litter, v.sem.dat))
```

```
##
```

```
## Call:
## lm(formula = d.Asteraceae.ovales ~ d.litter, data = v.sem.dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.006  -7.296   5.653   7.482  19.553
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.36833    2.34896  -3.137  0.00399 **
## d.litter    -0.05076    0.07324  -0.693  0.49395
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.36 on 28 degrees of freedom
## Multiple R-squared:  0.01687,    Adjusted R-squared:  -0.01824
## F-statistic: 0.4804 on 1 and 28 DF,  p-value: 0.494
```

```
summary(lm(d.v.abun ~ d.light))
```

```
##
## Call:
## lm(formula = d.v.abun ~ d.light)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -47.204  -7.755   1.085  11.993  31.908
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -23.8611     6.2747  -3.803 0.000711 ***
## d.light      -0.1010     0.3585  -0.282 0.780349
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.26 on 28 degrees of freedom
## Multiple R-squared:  0.002823,   Adjusted R-squared:  -0.03279
## F-statistic: 0.07928 on 1 and 28 DF,  p-value: 0.7803
```

```
summary(lm(d.v.rich ~ d.light))
```

```
##
## Call:
## lm(formula = d.v.rich ~ d.light)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7203 -0.7086  0.2372  0.4718  2.3085
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.10636    0.34979  -3.163  0.00374 **
## d.light      0.01280    0.01999   0.640  0.52727
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.9622 on 28 degrees of freedom
## Multiple R-squared:  0.01443,    Adjusted R-squared:  -0.02077
## F-statistic: 0.4098 on 1 and 28 DF,  p-value: 0.5273
```

```r
summary(lm(d.v.shan ~ d.light))
```

```
##
## Call:
## lm(formula = d.v.shan ~ d.light)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5917 -0.3570  0.1214  0.2817  0.9857
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.362101   0.141162  -2.565    0.016 *
## d.light     -0.004403   0.008066  -0.546    0.589
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3883 on 28 degrees of freedom
## Multiple R-squared:  0.01053,    Adjusted R-squared:  -0.02481
## F-statistic: 0.298 on 1 and 28 DF,  p-value: 0.5895
```

```r
summary(lm(d.Apache.plume ~ d.light, v.sem.dat))
```

```
##
## Call:
## lm(formula = d.Apache.plume ~ d.light, data = v.sem.dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.062  -4.319   4.807   9.297  16.737
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -14.6411     4.4197  -3.313  0.00256 **
## d.light      -0.2891     0.2525  -1.145  0.26208
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.16 on 28 degrees of freedom
## Multiple R-squared:  0.0447, Adjusted R-squared:  0.01058
## F-statistic:  1.31 on 1 and 28 DF,  p-value: 0.2621
```

```r
summary(lm(d.Asteraceae.ovales ~ d.light, v.sem.dat))
```

```
##
## Call:
## lm(formula = d.Asteraceae.ovales ~ d.light, data = v.sem.dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.874  -6.867   6.133   8.134  18.131
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -8.1407349  4.1640687  -1.955   0.0606 .
## d.light     -0.0004891  0.2379432  -0.002   0.9984
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.45 on 28 degrees of freedom
## Multiple R-squared:  1.509e-07,  Adjusted R-squared:  -0.03571
## F-statistic: 4.225e-06 on 1 and 28 DF,  p-value: 0.9984
```

```r
summary(model.v.com, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.v.com
##
## Call:
##   d.rocks ~ d.litter
##   d.v.com ~ d.light + d.rocks
##
##     AIC      BIC
##  28.300   38.108
##
## ---
## Tests of directed separation:
##
##             Independ.Claim Test.Type DF Crit.Value P.Value
##   d.v.com ~ d.litter + ...      coef 26     2.0909  0.0465 *
##    d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 14.3 with P-value = 0.006 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##    d.v.com   d.light   0.0177    0.3475 27     0.0508  0.9598       0.0099
##    d.v.com   d.rocks   0.0595    0.1090 27     0.5453  0.5900       0.1064
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##    d.v.com   none      0.01
```

```r
summary(model.v.abun, .progressBar = FALSE)
```

```
##
```

```
## Structural Equation Model of model.v.abun
##
## Call:
##   d.rocks ~ d.litter
##   d.v.abun ~ d.light + d.rocks
##
##     AIC      BIC
##  28.663   38.471
##
## ---
## Tests of directed separation:
##
##              Independ.Claim Test.Type DF Crit.Value P.Value
##   d.v.abun ~ d.litter + ...      coef 26    -2.1770  0.0387 *
##     d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 14.663 with P-value = 0.005 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##   d.v.abun   d.light  -0.0483    0.3688 27    -0.1310  0.8967      -0.0254
##   d.v.abun   d.rocks  -0.0842    0.1157 27    -0.7277  0.4731      -0.1412
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##   d.v.abun   none      0.02
```

```
summary(model.v.rich, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.v.rich
##
## Call:
##   d.rocks ~ d.litter
##   d.v.rich ~ d.light + d.rocks
##
##     AIC      BIC
##  25.623   35.431
##
## ---
## Tests of directed separation:
##
##              Independ.Claim Test.Type DF Crit.Value P.Value
##   d.v.rich ~ d.litter + ...      coef 26    -1.3873  0.1771
##     d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
```

```
##
## Global goodness-of-fit:
##
##   Fisher's C = 11.623 with P-value = 0.02 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##   d.v.rich   d.light   0.0115    0.0207 27     0.5561  0.5827       0.1082
##   d.v.rich   d.rocks   0.0020    0.0065 27     0.3131  0.7566       0.0609
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##   d.v.rich   none      0.02
```

```r
summary(model.v.shan, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.v.shan
##
## Call:
##   d.rocks ~ d.litter
##   d.v.shan ~ d.light + d.rocks
##
##     AIC      BIC
##  26.895   36.703
##
## ---
## Tests of directed separation:
##
##              Independ.Claim Test.Type DF Crit.Value P.Value
##   d.v.shan ~ d.litter + ...      coef 26    -1.7395  0.0938
##     d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 12.895 with P-value = 0.012 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##   Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##    d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983 ***
##   d.v.shan   d.light  -0.0028    0.0082 27    -0.3397  0.7367      -0.0651
##   d.v.shan   d.rocks  -0.0026    0.0026 27    -0.9971  0.3276      -0.1912
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
```

```
## ---
## Individual R-squared:
##
##   Response method R.squared
##    d.rocks   none      1.00
##   d.v.shan   none      0.05
```

```r
summary(model.v.Apache.plume, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.v.Apache.plume
##
## Call:
##   d.rocks ~ d.litter
##   d.Apache.plume ~ d.light + d.rocks
##
##     AIC      BIC
##  25.830   35.638
##
## ---
## Tests of directed separation:
##
##                      Independ.Claim Test.Type DF Crit.Value P.Value
##   d.Apache.plume ~ d.litter + ...      coef 26    -1.4474  0.1597
##           d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 11.83 with P-value = 0.019 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##         Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
##          d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000      -0.9983
##   d.Apache.plume   d.light  -0.2176    0.2527 27    -0.8611  0.3968      -0.1592
##   d.Apache.plume   d.rocks  -0.1142    0.0793 27    -1.4408  0.1611      -0.2663
##
##   ***
##
##
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##         Response method R.squared
##          d.rocks   none      1.00
##   d.Apache.plume   none      0.11
```

```r
summary(model.v.Asteraceae.ovales, .progressBar = FALSE)
```

```
##
## Structural Equation Model of model.v.Asteraceae.ovales
```

```
##
## Call:
##   d.rocks ~ d.litter
##   d.Asteraceae.ovales ~ d.light + d.rocks
##
##     AIC      BIC
##  24.690   34.498
##
## ---
## Tests of directed separation:
##
##                         Independ.Claim Test.Type DF Crit.Value P.Value
##   d.Asteraceae.ovales ~ d.litter + ...      coef 26    -1.0976  0.2824
##                d.rocks ~ d.light + ...      coef 27     2.5465  0.0169 *
##
## Global goodness-of-fit:
##
##   Fisher's C = 10.69 with P-value = 0.03 and on 4 degrees of freedom
##
## ---
## Coefficients:
##
##             Response Predictor Estimate Std.Error DF Crit.Value P.Value
##              d.rocks  d.litter  -0.9879    0.0108 28   -91.5294  0.0000
##   d.Asteraceae.ovales   d.light  -0.0310    0.2453 27    -0.1262  0.9005
##   d.Asteraceae.ovales   d.rocks   0.0488    0.0770 27     0.6335  0.5317
##   Std.Estimate
##        -0.9983 ***
##        -0.0246
##         0.1234
##
##   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## ---
## Individual R-squared:
##
##             Response method R.squared
##              d.rocks   none      1.00
##   d.Asteraceae.ovales   none      0.01
```

Analyses for Revisions

Tree -> Moth -> Trait -> Loc env -> Community (A, R, D, Comp)

Pair S/R Crown Litter Lichen Rocks Plants Light

Lichen and plant community resposnes are not correlated

```r
mantel(l.com.dif.d ~ v.com.dif.d)
```

```
##     mantelr       pval1       pval2       pval3   llim.2.5%  ulim.97.5%
## -0.11773949  0.79100000  0.21000000  0.43800000 -0.23133491 -0.03334609
```

```r
mantel(l.com.dif.d ~ env.dif.d)
```

```
##     mantelr       pval1       pval2       pval3   llim.2.5%  ulim.97.5%
##  0.01150233  0.44800000  0.55300000  0.93900000 -0.03897137  0.08388580
```

```r
mantel(l.com.dif.d ~ tra.dif.d)
```

```
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
##  0.2323704  0.0390000  0.9620000  0.0420000  0.1419806  0.3350468
```

```r
mantel(v.com.dif.d ~ env.dif.d)
```

```
##     mantelr       pval1       pval2       pval3   llim.2.5%  ulim.97.5%
## -0.11698559  0.88400000  0.11700000  0.25100000 -0.15953527 -0.05108963
```

```r
mantel(v.com.dif.d ~ tra.dif.d)
```

```
##     mantelr       pval1       pval2       pval3   llim.2.5%  ulim.97.5%
## -0.07840365  0.76400000  0.23700000  0.46000000 -0.13547720 -0.01712182
```

Both lichen and vegetation respond to moth susceptibility

```r
set.seed(12345)
xtable::xtable(
            adonis2(l.com.dif.d ~ crown.radius + rock.sm + rock.lg + light + litter,
                    data = data.frame(env.dif, tra.dif), by = "margin")
        )
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Mon May 3 18:38:04 2021

	Df	SumOfSqs	R2	F	Pr(>F)
crown.radius	1	4.68	0.02	0.74	0.4920
rock.sm	1	30.45	0.13	4.78	0.0290
rock.lg	1	29.53	0.13	4.64	0.0270
light	1	2.01	0.01	0.32	0.7820
litter	1	29.47	0.13	4.63	0.0260
Residual	24	152.87	0.67		
Total	29	226.87	1.00		

```r
set.seed(12345)
xtable::xtable(
            adonis2(sqrt(v.com.dif.d) ~ litter + rock.sm + rock.lg + light + crown.radius,
                    data = data.frame(env.dif, tra.dif), by = "margin")
        )
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Mon May 3 18:38:04 2021

	Df	SumOfSqs	R2	F	Pr(>F)
litter	1	11.61	0.03	0.94	0.4480
rock.sm	1	12.98	0.04	1.05	0.3730
rock.lg	1	11.43	0.03	0.92	0.4540
light	1	7.27	0.02	0.59	0.8290
crown.radius	1	11.07	0.03	0.89	0.5230
Residual	24	297.41	0.83		
Total	29	358.29	1.00		

```r
set.seed(12345)
xtable::xtable(adonis2(com.ds ~ Moth,
                       strata = l.dat[, "Tree.pairs"],
                       data = l.dat,
                       perm = 9999)
               )
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Mon May 3 18:38:05 2021

	Df	SumOfSqs	R2	F	Pr(>F)
Moth	1	0.83	0.04	2.35	0.0305
Residual	58	20.54	0.96		
Total	59	21.37	1.00		

```r
set.seed(12345)
xtable::xtable(adonis2(v.com.ds ~ Moth,
                       strata = l.dat[, "Tree.pairs"],
                       data = l.dat,
                       perm = 9999)
               )
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Mon May 3 18:38:09 2021

	Df	SumOfSqs	R2	F	Pr(>F)
Moth	1	5.17	0.31	25.83	0.0001
Residual	58	11.62	0.69		
Total	59	16.79	1.00		

```r
set.seed(12345)
moth.perm.l <- adonis2(com.ds ~ Moth,
                      strata = l.dat[, "Tree.pairs"],
                      data = l.dat,
                      perm = 100000)

set.seed(12345)
moth.perm.v <- adonis2(v.com.ds ~ Moth,
                     strata = l.dat[, "Tree.pairs"],
                     data = l.dat,
                     perm = 100000)
```

```r
tab.perm.l <- data.frame(moth.perm.l)
tab.perm.v <- data.frame(moth.perm.v)

tab.fact <- rownames(tab.perm.l)

tab.perm.l <- apply(tab.perm.l, 2, as.numeric)
tab.perm.v <- apply(tab.perm.v, 2, as.numeric)

colnames(tab.perm.l) <- c("df", "SS", "R2", "pseudo-F", "p-value")
colnames(tab.perm.v) <- c("df", "SS", "R2", "pseudo-F", "p-value")

tab.perm.l[1, "p-value"] <- round(tab.perm.l[1, "p-value"], 4)
tab.perm.v[1, "p-value"] <- round(tab.perm.v[1, "p-value"], 4)
```

```r
tab.perm.l[1, "pseudo-F"] <- round(tab.perm.l[1, "pseudo-F"], 2)
tab.perm.v[1, "pseudo-F"] <- round(tab.perm.v[1, "pseudo-F"], 2)

tab.perm.l[, "SS"] <- round(tab.perm.l[, "SS"], 2)
tab.perm.v[, "SS"] <- round(tab.perm.v[, "SS"], 2)

tab.perm.l[, "R2"] <- round(tab.perm.l[, "R2"], 2)
tab.perm.v[, "R2"] <- round(tab.perm.v[, "R2"], 2)

tab.perm.l[is.na(tab.perm.l)] <- ""
tab.perm.v[is.na(tab.perm.v)] <- ""

rownames(tab.perm.l) <- tab.fact
rownames(tab.perm.v) <- tab.fact

write.csv(file = "results/table_perm_moth_lichen.csv", tab.perm.l)
write.csv(file = "results/table_perm_moth_plant.csv", tab.perm.v)
```

```r
tab.ttest.ard <- do.call(rbind,
                        lapply(
                            apply(data.frame(l.ard.dif, v.ard.dif), 2,
                                  t.test),
                            unlist))[, c(1, 2, 6, 3)]
tab.lab <- rownames(tab.ttest.ard)
tab.ttest.ard <- apply(tab.ttest.ard, 2, as.numeric)
rownames(tab.ttest.ard) <- tab.lab
xtable::xtable(tab.ttest.ard, digits = 5)
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:11 2021

	statistic.t	parameter.df	estimate.mean of x	p.value
l.A	-2.24873	29.00000	-1.54400	0.03230
l.R	-2.95490	29.00000	-2.53333	0.00615
l.D	-2.44677	29.00000	-0.43698	0.02071
p.A	-7.13460	29.00000	-22.43333	0.00000
p.R	-7.47696	29.00000	-1.30000	0.00000
p.D	-4.21918	29.00000	-0.29546	0.00022

Moth impacts tree traits and the local environment

```r
tab.ttest.envtra <- do.call(rbind,
                        lapply(
                            apply(data.frame(tra.dif, env.dif), 2,
                                  t.test),
                            unlist))[, c(1, 2, 6, 3)]
tab.lab <- rownames(tab.ttest.envtra)
tab.ttest.envtra <- apply(tab.ttest.envtra, 2, as.numeric)
rownames(tab.ttest.envtra) <- tab.lab
xtable::xtable(tab.ttest.envtra, digits = 5)
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:11 2021

	statistic.t	parameter.df	estimate.mean of x	p.value
trunk.radius	-3.59977	29.00000	-3.13667	0.00117
crown.radius	-4.61833	29.00000	-58.48667	0.00007
litter	2.86654	29.00000	15.07000	0.00765
rocks	-2.81780	29.00000	-14.65867	0.00862
rock.lg	-2.46174	29.00000	-9.68367	0.02001
rock.sm	-2.07917	29.00000	-4.97500	0.04655
light	-9.27275	29.00000	-15.13333	0.00000

Tree environment correlate with community

```
set.seed(12345)
xtable::xtable(adonis2(com.ds ~ Big.rocks.. + Small.rocks.. + Light...average,
                       strata = l.dat[, "Tree.pairs"],
                       by = "margin",
                       data = data.frame(env, traits),
                       perm = 9999, rank = TRUE)
               )
```

```
## % latex table generated in R 4.0.4 by xtable 1.8-4 package
## % Wed Apr 21 12:26:26 2021
## \begin{table}[ht]
## \centering
## \begin{tabular}{lrrrrr}
##   \hline
##  & Df & SumOfSqs & R2 & F & Pr($>$F) \\
##   \hline
## Big.rocks.. & 1 & 1.79 & 0.08 & 5.47 & 0.0004 \\
##   Small.rocks.. & 1 & 0.27 & 0.01 & 0.81 & 0.5720 \\
##   Light...average & 1 & 0.39 & 0.02 & 1.20 & 0.2649 \\
##   Residual & 56 & 18.31 & 0.86 &  &  \\
##   Total & 59 & 21.37 & 1.00 &  &  \\
##    \hline
## \end{tabular}
## \end{table}
```

```
set.seed(12345)
xtable::xtable(adonis2(v.com.ds ~ Light...average + Big.rocks.. + Small.rocks..,
                       strata = l.dat[, "Tree.pairs"],
                       by = "margin",
                       data = data.frame(env, traits),
                       perm = 9999)
               )
```

```
## % latex table generated in R 4.0.4 by xtable 1.8-4 package
## % Wed Apr 21 12:26:30 2021
## \begin{table}[ht]
## \centering
## \begin{tabular}{lrrrrr}
##   \hline
##  & Df & SumOfSqs & R2 & F & Pr($>$F) \\
##   \hline
## Light...average & 1 & 2.93 & 0.17 & 13.00 & 0.0001 \\
##   Big.rocks.. & 1 & 0.10 & 0.01 & 0.44 & 0.7243 \\
```

```
##   Small.rocks.. & 1 & 0.73 & 0.04 & 3.26 & 0.0290 \\
##   Residual & 56 & 12.61 & 0.75 &  &  \\
##   Total & 59 & 16.79 & 1.00 &  &  \\
##    \hline
## \end{tabular}
## \end{table}
```

```
summary(lm(l.A ~ rock.lg * rock.sm * light,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.A ~ rock.lg * rock.sm * light, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.5443 -0.9009  0.3873  1.2621  4.7576
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              1.2906171  1.9919281   0.648   0.5237
## rock.lg                  0.2672626  0.1144530   2.335   0.0291 *
## rock.sm                 -0.2489435  0.2305602  -1.080   0.2920
## light                    0.0964938  0.1233636   0.782   0.4424
## rock.lg:rock.sm         -0.0098077  0.0131545  -0.746   0.4638
## rock.lg:light            0.0108967  0.0067177   1.622   0.1190
## rock.sm:light           -0.0130569  0.0118033  -1.106   0.2806
## rock.lg:rock.sm:light -0.0002544  0.0005513  -0.461   0.6490
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.337 on 22 degrees of freedom
## Multiple R-squared:  0.4027, Adjusted R-squared:  0.2127
## F-statistic: 2.119 on 7 and 22 DF,  p-value: 0.08438
```

```
summary(lm(l.R ~ rock.lg * rock.sm * light,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.R ~ rock.lg * rock.sm * light, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.4034 -1.7571  0.5585  2.0862  3.9423
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             2.8682448  1.8246866   1.572   0.1302
## rock.lg                 0.3576352  0.1048436   3.411   0.0025 **
## rock.sm                 0.0782553  0.2112024   0.371   0.7145
## light                   0.2596367  0.1130061   2.298   0.0315 *
## rock.lg:rock.sm         0.0060809  0.0120501   0.505   0.6188
```

```
## rock.lg:light         0.0114837  0.0061537   1.866   0.0754 .
## rock.sm:light         0.0050780  0.0108123   0.470   0.6432
## rock.lg:rock.sm:light 0.0003271  0.0005050   0.648   0.5238
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.057 on 22 degrees of freedom
## Multiple R-squared:  0.6785, Adjusted R-squared:  0.5762
## F-statistic: 6.634 on 7 and 22 DF,  p-value: 0.0002762
```

```
summary(lm(l.D ~ rock.lg * rock.sm * light,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.D ~ rock.lg * rock.sm * light, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.3539 -0.1798  0.1183  0.3590  0.9120
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           7.064e-01  3.914e-01   1.805   0.0848 .
## rock.lg               5.437e-02  2.249e-02   2.418   0.0243 *
## rock.sm               5.766e-02  4.530e-02   1.273   0.2163
## light                 6.085e-02  2.424e-02   2.511   0.0199 *
## rock.lg:rock.sm       2.179e-03  2.585e-03   0.843   0.4082
## rock.lg:light         1.247e-03  1.320e-03   0.945   0.3552
## rock.sm:light         3.242e-03  2.319e-03   1.398   0.1761
## rock.lg:rock.sm:light 8.461e-05  1.083e-04   0.781   0.4431
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6557 on 22 degrees of freedom
## Multiple R-squared:  0.6592, Adjusted R-squared:  0.5508
## F-statistic: 6.079 on 7 and 22 DF,  p-value: 0.0004929
```

```
summary(lm(l.A ~ light *rock.lg * rock.sm,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.A ~ light * rock.lg * rock.sm, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.5443 -0.9009  0.3873  1.2621  4.7576
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            1.2906171  1.9919281   0.648   0.5237
## light                  0.0964938  0.1233636   0.782   0.4424
```

```
## rock.lg                0.2672626  0.1144530   2.335   0.0291 *
## rock.sm               -0.2489435  0.2305602  -1.080   0.2920
## light:rock.lg          0.0108967  0.0067177   1.622   0.1190
## light:rock.sm         -0.0130569  0.0118033  -1.106   0.2806
## rock.lg:rock.sm       -0.0098077  0.0131545  -0.746   0.4638
## light:rock.lg:rock.sm -0.0002544  0.0005513  -0.461   0.6490
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.337 on 22 degrees of freedom
## Multiple R-squared:  0.4027, Adjusted R-squared:  0.2127
## F-statistic: 2.119 on 7 and 22 DF,  p-value: 0.08438
```

```r
summary(lm(l.R ~ light *rock.lg * rock.sm,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.R ~ light * rock.lg * rock.sm, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.4034 -1.7571  0.5585  2.0862  3.9423
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           2.8682448  1.8246866   1.572   0.1302
## light                 0.2596367  0.1130061   2.298   0.0315 *
## rock.lg               0.3576352  0.1048436   3.411   0.0025 **
## rock.sm               0.0782553  0.2112024   0.371   0.7145
## light:rock.lg         0.0114837  0.0061537   1.866   0.0754 .
## light:rock.sm         0.0050780  0.0108123   0.470   0.6432
## rock.lg:rock.sm       0.0060809  0.0120501   0.505   0.6188
## light:rock.lg:rock.sm 0.0003271  0.0005050   0.648   0.5238
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.057 on 22 degrees of freedom
## Multiple R-squared:  0.6785, Adjusted R-squared:  0.5762
## F-statistic: 6.634 on 7 and 22 DF,  p-value: 0.0002762
```

```r
summary(lm(l.D ~ light *rock.lg * rock.sm,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.D ~ light * rock.lg * rock.sm, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.3539 -0.1798  0.1183  0.3590  0.9120
##
## Coefficients:
```

```
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            7.064e-01  3.914e-01   1.805   0.0848 .
## light                  6.085e-02  2.424e-02   2.511   0.0199 *
## rock.lg                5.437e-02  2.249e-02   2.418   0.0243 *
## rock.sm                5.766e-02  4.530e-02   1.273   0.2163
## light:rock.lg          1.247e-03  1.320e-03   0.945   0.3552
## light:rock.sm          3.242e-03  2.319e-03   1.398   0.1761
## rock.lg:rock.sm        2.179e-03  2.585e-03   0.843   0.4082
## light:rock.lg:rock.sm 8.461e-05  1.083e-04   0.781   0.4431
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6557 on 22 degrees of freedom
## Multiple R-squared:  0.6592, Adjusted R-squared:  0.5508
## F-statistic: 6.079 on 7 and 22 DF,  p-value: 0.0004929
```

```
summary(lm(l.A ~ rock.lg + rock.sm + light,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.A ~ rock.lg + rock.sm + light, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7485 -0.6511  0.6642  1.3935  5.4237
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.427328   1.224495  -0.349  0.72991
## rock.lg      0.088123   0.030432   2.896  0.00757 **
## rock.sm      0.022591   0.050663   0.446  0.65935
## light        0.009973   0.071228   0.140  0.88972
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.346 on 26 degrees of freedom
## Multiple R-squared:  0.2904, Adjusted R-squared:  0.2085
## F-statistic: 3.547 on 3 and 26 DF,  p-value: 0.02821
```

```
summary(lm(l.R ~ rock.lg + rock.sm + light,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.R ~ rock.lg + rock.sm + light, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.6550 -1.9714  0.6468  2.0461  6.0752
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  0.371141   1.130676   0.328    0.745
## rock.lg      0.162543   0.028100   5.784  4.3e-06 ***
## rock.sm     -0.005166   0.046781  -0.110    0.913
## light        0.089614   0.065770   1.363    0.185
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.089 on 26 degrees of freedom
## Multiple R-squared:  0.6119, Adjusted R-squared:  0.5672
## F-statistic: 13.67 on 3 and 26 DF,  p-value: 1.515e-05
```

```
summary(lm(l.D ~ rock.lg + rock.sm + light,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.D ~ rock.lg + rock.sm + light, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.20164 -0.37452  0.01855  0.38633  1.20307
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.1937003  0.2527542   0.766    0.450
## rock.lg      0.0315016  0.0062816   5.015 3.23e-05 ***
## rock.sm     -0.0007058  0.0104575  -0.067    0.947
## light        0.0217497  0.0147024   1.479    0.151
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6906 on 26 degrees of freedom
## Multiple R-squared:  0.5531, Adjusted R-squared:  0.5016
## F-statistic: 10.73 on 3 and 26 DF,  p-value: 9.066e-05
```

```
summary(lm(l.A ~ light +rock.lg + rock.sm,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.A ~ light + rock.lg + rock.sm, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7485 -0.6511  0.6642  1.3935  5.4237
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.427328   1.224495  -0.349  0.72991
## light        0.009973   0.071228   0.140  0.88972
## rock.lg      0.088123   0.030432   2.896  0.00757 **
## rock.sm      0.022591   0.050663   0.446  0.65935
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.346 on 26 degrees of freedom
## Multiple R-squared:  0.2904, Adjusted R-squared:  0.2085
## F-statistic: 3.547 on 3 and 26 DF,  p-value: 0.02821
```

```r
summary(lm(l.R ~ light +rock.lg + rock.sm,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.R ~ light + rock.lg + rock.sm, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.6550 -1.9714  0.6468  2.0461  6.0752
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.371141   1.130676   0.328    0.745
## light        0.089614   0.065770   1.363    0.185
## rock.lg      0.162543   0.028100   5.784  4.3e-06 ***
## rock.sm     -0.005166   0.046781  -0.110    0.913
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.089 on 26 degrees of freedom
## Multiple R-squared:  0.6119, Adjusted R-squared:  0.5672
## F-statistic: 13.67 on 3 and 26 DF,  p-value: 1.515e-05
```

```r
summary(lm(l.D ~ light +rock.lg + rock.sm,
           data = data.frame(l.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = l.D ~ light + rock.lg + rock.sm, data = data.frame(l.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.20164 -0.37452  0.01855  0.38633  1.20307
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.1937003  0.2527542   0.766    0.450
## light        0.0217497  0.0147024   1.479    0.151
## rock.lg      0.0315016  0.0062816   5.015 3.23e-05 ***
## rock.sm     -0.0007058  0.0104575  -0.067    0.947
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6906 on 26 degrees of freedom
## Multiple R-squared:  0.5531, Adjusted R-squared:  0.5016
## F-statistic: 10.73 on 3 and 26 DF,  p-value: 9.066e-05
```

```r
summary(lm(p.A ~ rock.lg * rock.sm * light,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.A ~ rock.lg * rock.sm * light, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.808  -8.565   2.356  11.435  25.518
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -24.997498  10.598639  -2.359   0.0276 *
## rock.lg               -0.322706   0.608981  -0.530   0.6015
## rock.sm               -0.574845   1.226763  -0.469   0.6440
## light                 -0.068351   0.656392  -0.104   0.9180
## rock.lg:rock.sm       -0.027964   0.069993  -0.400   0.6934
## rock.lg:light         -0.026183   0.035744  -0.733   0.4716
## rock.sm:light          0.006300   0.062803   0.100   0.9210
## rock.lg:rock.sm:light -0.001141   0.002933  -0.389   0.7011
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.76 on 22 degrees of freedom
## Multiple R-squared:  0.1937, Adjusted R-squared:  -0.06288
## F-statistic: 0.7549 on 7 and 22 DF,  p-value: 0.6297
```

```r
summary(lm(p.R ~ rock.lg * rock.sm * light,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.R ~ rock.lg * rock.sm * light, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.15006 -0.67011 -0.00113  0.40891  2.13338
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -1.121e+00  5.309e-01  -2.111   0.0463 *
## rock.lg               1.329e-02  3.050e-02   0.436   0.6674
## rock.sm              -3.598e-03  6.145e-02  -0.059   0.9538
## light                 1.453e-02  3.288e-02   0.442   0.6629
## rock.lg:rock.sm       1.782e-03  3.506e-03   0.508   0.6163
## rock.lg:light        -4.340e-04  1.790e-03  -0.242   0.8107
## rock.sm:light         1.363e-03  3.146e-03   0.433   0.6690
## rock.lg:rock.sm:light 5.302e-05  1.469e-04   0.361   0.7217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.8894 on 22 degrees of freedom
## Multiple R-squared:  0.3383, Adjusted R-squared:  0.1278
## F-statistic: 1.607 on 7 and 22 DF,  p-value: 0.1857
```

```r
summary(lm(p.D ~ rock.lg * rock.sm * light,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.D ~ rock.lg * rock.sm * light, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.61818 -0.27861 -0.01608  0.24591  0.88670
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -4.975e-01  2.268e-01  -2.194   0.0391 *
## rock.lg                -9.983e-03  1.303e-02  -0.766   0.4518
## rock.sm                -1.668e-02  2.625e-02  -0.635   0.5317
## light                  -1.037e-02  1.405e-02  -0.738   0.4680
## rock.lg:rock.sm        -3.217e-04  1.498e-03  -0.215   0.8319
## rock.lg:light          -7.732e-04  7.648e-04  -1.011   0.3230
## rock.sm:light          -2.122e-04  1.344e-03  -0.158   0.8759
## rock.lg:rock.sm:light -2.246e-05  6.277e-05  -0.358   0.7239
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3799 on 22 degrees of freedom
## Multiple R-squared:  0.2557, Adjusted R-squared:  0.01892
## F-statistic:  1.08 on 7 and 22 DF,  p-value: 0.4088
```

```r
summary(lm(p.A ~ light *rock.lg * rock.sm,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.A ~ light * rock.lg * rock.sm, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.808  -8.565   2.356  11.435  25.518
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -24.997498  10.598639  -2.359   0.0276 *
## light                   -0.068351   0.656392  -0.104   0.9180
## rock.lg                 -0.322706   0.608981  -0.530   0.6015
## rock.sm                 -0.574845   1.226763  -0.469   0.6440
## light:rock.lg           -0.026183   0.035744  -0.733   0.4716
## light:rock.sm            0.006300   0.062803   0.100   0.9210
## rock.lg:rock.sm         -0.027964   0.069993  -0.400   0.6934
## light:rock.lg:rock.sm  -0.001141   0.002933  -0.389   0.7011
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.76 on 22 degrees of freedom
## Multiple R-squared:  0.1937, Adjusted R-squared:  -0.06288
## F-statistic: 0.7549 on 7 and 22 DF,  p-value: 0.6297
```

```
summary(lm(p.R ~ light *rock.lg * rock.sm,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.R ~ light * rock.lg * rock.sm, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.15006 -0.67011 -0.00113  0.40891  2.13338
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -1.121e+00  5.309e-01  -2.111   0.0463 *
## light                   1.453e-02  3.288e-02   0.442   0.6629
## rock.lg                 1.329e-02  3.050e-02   0.436   0.6674
## rock.sm                -3.598e-03  6.145e-02  -0.059   0.9538
## light:rock.lg          -4.340e-04  1.790e-03  -0.242   0.8107
## light:rock.sm           1.363e-03  3.146e-03   0.433   0.6690
## rock.lg:rock.sm         1.782e-03  3.506e-03   0.508   0.6163
## light:rock.lg:rock.sm   5.302e-05  1.469e-04   0.361   0.7217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8894 on 22 degrees of freedom
## Multiple R-squared:  0.3383, Adjusted R-squared:  0.1278
## F-statistic: 1.607 on 7 and 22 DF,  p-value: 0.1857
```

```
summary(lm(p.D ~ light *rock.lg * rock.sm,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.D ~ light * rock.lg * rock.sm, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.61818 -0.27861 -0.01608  0.24591  0.88670
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -4.975e-01  2.268e-01  -2.194   0.0391 *
## light                  -1.037e-02  1.405e-02  -0.738   0.4680
## rock.lg                -9.983e-03  1.303e-02  -0.766   0.4518
## rock.sm                -1.668e-02  2.625e-02  -0.635   0.5317
## light:rock.lg          -7.732e-04  7.648e-04  -1.011   0.3230
```

```
## light:rock.sm         -2.122e-04  1.344e-03  -0.158   0.8759
## rock.lg:rock.sm       -3.217e-04  1.498e-03  -0.215   0.8319
## light:rock.lg:rock.sm -2.246e-05  6.277e-05  -0.358   0.7239
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3799 on 22 degrees of freedom
## Multiple R-squared:  0.2557, Adjusted R-squared:  0.01892
## F-statistic:  1.08 on 7 and 22 DF,  p-value: 0.4088
```

```
summary(lm(p.A ~ rock.lg + rock.sm + light,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.A ~ rock.lg + rock.sm + light, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.955  -8.621   2.115  12.151  28.829
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -23.55502    6.14481  -3.833 0.000721 ***
## rock.lg       0.11754    0.15271   0.770 0.448432
## rock.sm      -0.53383    0.25424  -2.100 0.045607 *
## light         0.02616    0.35744   0.073 0.942215
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.79 on 26 degrees of freedom
## Multiple R-squared:  0.1479, Adjusted R-squared:  0.04957
## F-statistic: 1.504 on 3 and 26 DF,  p-value: 0.2368
```

```
summary(lm(p.R ~ rock.lg + rock.sm + light,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.R ~ rock.lg + rock.sm + light, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.09085 -0.72885  0.07251  0.43267  2.04097
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.027067   0.302605  -3.394  0.00222 **
## rock.lg      0.019656   0.007521   2.614  0.01470 *
## rock.sm     -0.036574   0.012520  -2.921  0.00712 **
## light        0.017481   0.017602   0.993  0.32981
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.8268 on 26 degrees of freedom
## Multiple R-squared:  0.3242, Adjusted R-squared:  0.2462
## F-statistic: 4.157 on 3 and 26 DF,  p-value: 0.01565
```

```r
summary(lm(p.D ~ rock.lg + rock.sm + light,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.D ~ rock.lg + rock.sm + light, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.48929 -0.33019 -0.02457  0.29568  0.88860
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.3546159  0.1309685  -2.708   0.0118 *
## rock.lg      0.0027760  0.0032549   0.853   0.4015
## rock.sm     -0.0142947  0.0054187  -2.638   0.0139 *
## light       -0.0009857  0.0076183  -0.129   0.8980
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3579 on 26 degrees of freedom
## Multiple R-squared:  0.2196, Adjusted R-squared:  0.1296
## F-statistic: 2.439 on 3 and 26 DF,  p-value: 0.08707
```

```r
summary(lm(p.A ~ light +rock.lg + rock.sm,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.A ~ light + rock.lg + rock.sm, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.955  -8.621   2.115  12.151  28.829
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -23.55502    6.14481  -3.833 0.000721 ***
## light         0.02616    0.35744   0.073 0.942215
## rock.lg       0.11754    0.15271   0.770 0.448432
## rock.sm      -0.53383    0.25424  -2.100 0.045607 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.79 on 26 degrees of freedom
## Multiple R-squared:  0.1479, Adjusted R-squared:  0.04957
## F-statistic: 1.504 on 3 and 26 DF,  p-value: 0.2368
```

```r
summary(lm(p.R ~ light +rock.lg + rock.sm,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.R ~ light + rock.lg + rock.sm, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.09085 -0.72885  0.07251  0.43267  2.04097
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.027067   0.302605  -3.394  0.00222 **
## light        0.017481   0.017602   0.993  0.32981
## rock.lg      0.019656   0.007521   2.614  0.01470 *
## rock.sm     -0.036574   0.012520  -2.921  0.00712 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8268 on 26 degrees of freedom
## Multiple R-squared:  0.3242, Adjusted R-squared:  0.2462
## F-statistic: 4.157 on 3 and 26 DF,  p-value: 0.01565
```

```r
summary(lm(p.D ~ light +rock.lg + rock.sm,
           data = data.frame(v.ard.dif, tra.dif, env.dif)))
```

```
##
## Call:
## lm(formula = p.D ~ light + rock.lg + rock.sm, data = data.frame(v.ard.dif,
##     tra.dif, env.dif))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.48929 -0.33019 -0.02457  0.29568  0.88860
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.3546159  0.1309685  -2.708   0.0118 *
## light       -0.0009857  0.0076183  -0.129   0.8980
## rock.lg      0.0027760  0.0032549   0.853   0.4015
## rock.sm     -0.0142947  0.0054187  -2.638   0.0139 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3579 on 26 degrees of freedom
## Multiple R-squared:  0.2196, Adjusted R-squared:  0.1296
## F-statistic: 2.439 on 3 and 26 DF,  p-value: 0.08707
```

Structural Equation Models

```r
l.com.dif <- split(com, l.dat[, "Tree.pairs"])
l.com.dif <- lapply(l.com.dif, function(x) x[2, ] - x[1, ])
```

```r
l.com.dif <- do.call(rbind, l.com.dif)

v.com.dif <- split(v.com, l.dat[, "Tree.pairs"])
v.com.dif <- lapply(v.com.dif, function(x) x[2, ] - x[1, ])
v.com.dif <- do.call(rbind, v.com.dif)

l.com.dif.d <- dist(l.com.dif)
v.com.dif.d <- dist(v.com.dif)

l.com.dif.nms <- nmds(l.com.dif.d, 1, 2)
l.com.dif.ord <- nmds.min(l.com.dif.nms, 2)
```

```
## Minimum stress for given dimensionality:  0.07460277
## r^2 for minimum stress configuration:  0.9809944
```

```r
l.com.dif.vec <- envfit(l.com.dif.ord,
                        data.frame(env.dif, tra.dif)[, c("rock.lg",
                                                         "rock.sm",
                                                         "light",
                                                         "litter")])

v.com.dif.nms <- nmds(v.com.dif.d, 2, 3)
v.com.dif.ord <- nmds.min(v.com.dif.nms, 3)
```

```
## Minimum stress for given dimensionality:  0.03324742
## r^2 for minimum stress configuration:  0.9927886
```

```r
v.com.dif.vec <- envfit(v.com.dif.ord,
                        data.frame(env.dif, tra.dif)[, c("rock.lg",
                                                         "rock.sm",
                                                         "light",
                                                         "litter")])

colnames(l.com.dif.ord) <- paste0("l.", colnames(l.com.dif.ord))
colnames(v.com.dif.ord) <- paste0("p.", colnames(v.com.dif.ord))
```

```r
l.com.dif.ord.proc <- procrustes(env.dif[, "rock.lg"], l.com.dif.ord)$Yrot
```

```
## Warning in procrustes(env.dif[, "rock.lg"], l.com.dif.ord): X has fewer axes than Y: X adjusted to co
```

```r
v.com.dif.ord.proc <- procrustes(env.dif[, "rock.sm"], v.com.dif.ord)$Yrot
```

```
## Warning in procrustes(env.dif[, "rock.sm"], v.com.dif.ord): X has fewer axes than Y: X adjusted to co
```

```r
colnames(l.com.dif.ord.proc) <- paste0("rot.", colnames(l.com.dif.ord))
colnames(v.com.dif.ord.proc) <- paste0("rot.", colnames(v.com.dif.ord))

l.com.dif.vec.rot <- envfit(l.com.dif.ord.proc,
                            data.frame(env.dif[, -1], litter = tra.dif[, "litter"]))
v.com.dif.vec.rot <- envfit(v.com.dif.ord.proc,
                            data.frame(env.dif[, -1], litter = tra.dif[, "litter"]))
```

```r
sem.dat <- data.frame(tra.dif, env.dif, l.ard.dif, v.ard.dif, l.com.dif.ord, v.com.dif.ord, l.com.dif.o
colnames(sem.dat)[colnames(sem.dat) == "crown.radius"] <- "crown"
colnames(sem.dat)[colnames(sem.dat) == "trunk.radius"] <- "trunk"
```

```r
tab.ttest.ldat <- do.call(rbind,
                        lapply(
                            apply(l.dat[, -1:-3], 2,
                                  t.test),
                            unlist))[, c(1, 2, 6, 3)]
tab.lab <- rownames(tab.ttest.ldat)
tab.ttest.ldat <- apply(tab.ttest.ldat, 2, as.numeric)
rownames(tab.ttest.ldat) <- tab.lab
xtable::xtable(tab.ttest.ldat, digits = 5)
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:12 2021

	statistic.t	parameter.df	estimate.mean of x	p.value
Litter..	30.56225	59.00000	79.80633	0.00000
Big.rocks..	7.69468	59.00000	14.90117	0.00000
Small.rocks..	3.84706	59.00000	4.79783	0.00030
Shrubs..	2.61579	59.00000	0.40567	0.01129
Grass..	1.00000	59.00000	0.02467	0.32139
Branches..	1.00000	59.00000	0.07100	0.32139
Light...N	12.09160	59.00000	17.67833	0.00000
Light...S	12.00919	59.00000	17.80833	0.00000
Light...average	13.30890	59.00000	17.74333	0.00000
Acacon	3.91476	59.00000	0.02833	0.00024
Acaame	4.79957	59.00000	0.14000	0.00001
Acaobp	1.12174	59.00000	0.14933	0.26652
Sterile.sp	1.00000	59.00000	0.00100	0.32139
Brown.cr		59.00000	0.00000	
Lobalp	1.98868	59.00000	0.00233	0.05138
Canros	5.70908	59.00000	0.32017	0.00000
Calare	2.04690	59.00000	0.01967	0.04513
Phydub	3.55666	59.00000	0.09633	0.00075
Rhichr	3.82975	59.00000	0.29150	0.00031
Xanlin	3.63277	59.00000	0.62233	0.00059
Xanpli	4.25869	59.00000	0.21150	0.00007
Xanele	2.54509	59.00000	0.03867	0.01356
GrBr.cr	1.00000	59.00000	0.00067	0.32139
Gray.cr	1.69236	59.00000	0.00250	0.09585
Synrur	1.67611	59.00000	0.04933	0.09901
Cerpur.Bryarg	1.23020	59.00000	0.00867	0.22350

```r
xtable::xtable(na.omit(tab.ttest.ldat[tab.ttest.ldat[, "p.value"] <= 0.05,]))
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:12 2021

```r
tab.ttest.vdat <- do.call(rbind,
                        lapply(
                            apply(v.dat[, -1:-8], 2,
                                  t.test),
                            unlist))[, c(1, 2, 6, 3)]
tab.lab <- rownames(tab.ttest.vdat)
tab.ttest.vdat <- apply(tab.ttest.vdat, 2, as.numeric)
rownames(tab.ttest.vdat) <- tab.lab
xtable::xtable(tab.ttest.vdat, digits = 5)
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:12 2021

	statistic.t	parameter.df	estimate.mean of x	p.value
Litter..	30.56	59.00	79.81	0.00
Big.rocks..	7.69	59.00	14.90	0.00
Small.rocks..	3.85	59.00	4.80	0.00
Shrubs..	2.62	59.00	0.41	0.01
Light...N	12.09	59.00	17.68	0.00
Light...S	12.01	59.00	17.81	0.00
Light...average	13.31	59.00	17.74	0.00
Acacon	3.91	59.00	0.03	0.00
Acaame	4.80	59.00	0.14	0.00
Canros	5.71	59.00	0.32	0.00
Calare	2.05	59.00	0.02	0.05
Phydub	3.56	59.00	0.10	0.00
Rhichr	3.83	59.00	0.29	0.00
Xanlin	3.63	59.00	0.62	0.00
Xanpli	4.26	59.00	0.21	0.00
Xanele	2.55	59.00	0.04	0.01

```
xtable::xtable(na.omit(tab.ttest.vdat[tab.ttest.vdat[, "p.value"] <= 0.05,]))
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:12 2021

```
lav.l.all <- 'light ~ crown
              litter ~ crown
              rock.lg ~ litter
              l.A ~ light + rock.lg
              l.R ~ light + rock.lg
              l.D ~ light + rock.lg
              l.X1 ~ light + rock.lg
              l.X2 ~ light + rock.lg
              l.A ~~ l.R
              l.A ~~ l.D
              l.R ~~ l.D
              l.A ~~ l.X1
              l.R ~~ l.X1
             '
lav.v.all <- 'light ~ crown
              litter ~ crown
              rock.sm ~ litter
              p.A ~ light + rock.sm
              p.R ~ light + rock.sm  + litter
              p.D ~ light + rock.sm
              p.X1 ~ light + rock.sm
              p.X2 ~ light + rock.sm
              p.X3 ~ light + rock.sm
              p.A ~~ p.X2
              p.A ~~ p.R
              p.A ~~ p.D
              p.R ~~ p.D
              p.A ~~ p.X1
              p.R ~~ p.X1
             '
lav.l.rot.nolight <- 'litter ~ crown
```

	statistic.t	parameter.df	estimate.mean of x	p.value
Apache.plume	4.64843	59.00000	6.53333	0.00002
Juniperus.monosperma	1.00000	59.00000	0.08333	0.32139
Rhus.trilobata	1.80478	59.00000	1.58333	0.07621
Asteraceae.ovales	4.64433	59.00000	6.23333	0.00002
Bouteloua.gracilis		59.00000	0.00000	
Pinus.edulis.R	1.00000	59.00000	0.16667	0.32139
Pinus.edulis.S		59.00000	0.00000	
Stipa.A		59.00000	0.00000	
Stipa.B		59.00000	0.00000	
Stipa.très.grand		59.00000	0.00000	
Ephedra		59.00000	0.00000	
Rabbit.brush	1.00000	59.00000	0.33333	0.32139
Grande.grass.corymbe		59.00000	0.00000	
Boraginacée.rosette.grise		59.00000	0.00000	
Avena	1.76218	59.00000	0.10000	0.08322
Grass.à.nœud		59.00000	0.00000	
Brachypode		59.00000	0.00000	
Carex		59.00000	0.00000	
Cactus		59.00000	0.00000	
Hordeum		59.00000	0.00000	
Chenopodiaceae		59.00000	0.00000	
Ribes		59.00000	0.00000	
Aster.grise		59.00000	0.00000	
Rosette.frisée		59.00000	0.00000	
Chamaephyte.gris		59.00000	0.00000	
Castilleja		59.00000	0.00000	
Opuntia		59.00000	0.00000	
Rubiaceae		59.00000	0.00000	
Plante.grise.allongée	1.00000	59.00000	0.05000	0.32139
Scarlet.glia	1.00000	59.00000	0.03333	0.32139
Andropogon		59.00000	0.00000	

	statistic.t	parameter.df	estimate.mean of x	p.value
Apache.plume	4.65	59.00	6.53	0.00
Asteraceae.ovales	4.64	59.00	6.23	0.00

```
                  rock.lg ~ litter
                  l.A ~ rock.lg
                  l.R ~ rock.lg
                  l.D ~ rock.lg
                  rot.l.X1 ~ rock.lg
                  rot.l.X2 ~ rock.lg
                  l.A ~~ l.R
                  l.A ~~ l.D
                  l.R ~~ l.D
                  l.A ~~ rot.l.X1
                  l.R ~~ rot.l.X1
                 '
lav.v.rot.nolight <- 'litter ~ crown
                  rock.sm ~ litter
                  p.A ~ rock.sm
                  p.R ~ rock.sm
```

```
                 p.D ~ rock.sm
                 rot.p.X1 ~ rock.sm
                 rot.p.X2 ~ rock.sm
                 rot.p.X3 ~ rock.sm
                 p.A ~~ rot.p.X2
                 p.A ~~ p.R
                 p.A ~~ p.D
                 p.R ~~ p.D
                 p.A ~~ rot.p.X1
                 p.R ~~ rot.p.X1
                '
lav.l.rot.all <- 'light ~ crown
                 litter ~ crown
                 light ~ trunk
                 litter ~ trunk
                 rock.lg ~ litter
                 l.A ~ light + rock.lg
                 l.R ~ light + rock.lg
                 l.D ~ light + rock.lg
                 rot.l.X1 ~ light + rock.lg
                 rot.l.X2 ~ light + rock.lg
                 l.A ~~ l.R
                 l.A ~~ l.D
                 l.R ~~ l.D
                 l.A ~~ rot.l.X1
                 l.R ~~ rot.l.X1
                '
lav.v.rot.all <- 'light ~ crown
                 litter ~ crown
                 light ~ trunk
                 litter ~ trunk
                 rock.sm ~ litter
                 p.A ~ light + rock.sm
                 p.R ~ light + rock.sm
                 p.D ~ light + rock.sm
                 rot.p.X1 ~ light + rock.sm
                 rot.p.X2 ~ light + rock.sm
                 rot.p.X3 ~ light + rock.sm
                 p.A ~~ rot.p.X2
                 p.A ~~ p.R
                 p.A ~~ p.D
                 p.R ~~ p.D
                 p.A ~~ rot.p.X1
                 p.R ~~ rot.p.X1
                '

lav.v.rot.norock <- 'light ~ crown
                 litter ~ crown
                 light ~ trunk
                 litter ~ trunk
                 p.A ~ light + litter
                 p.R ~ light + litter
                 p.D ~ light + litter
```

```r
                  rot.p.X1 ~ light + litter
                  rot.p.X2 ~ light + litter
                  rot.p.X3 ~ light + litter
                  p.A ~~ rot.p.X2
                  p.A ~~ p.R
                  p.A ~~ p.D
                  p.R ~~ p.D
                  p.A ~~ rot.p.X1
                  p.R ~~ rot.p.X1
'

lav.v.rot.norock.ind.litter <- 'light ~ crown
                  litter ~ crown
                  light ~ trunk
                  litter ~ trunk
                  p.A ~ light
                  p.R ~ light
                  p.D ~ light
                  rot.p.X1 ~ light
                  rot.p.X2 ~ light
                  rot.p.X3 ~ light
                  p.A ~~ rot.p.X2
                  p.A ~~ p.R
                  p.A ~~ p.D
                  p.R ~~ p.D
                  p.A ~~ rot.p.X1
                  p.R ~~ rot.p.X1
'


std <- function(x){(x - mean(x)) / sd(x)}

set.seed(12345)
fit.l.all.raw <- lavaan::sem(lav.l.all, data = sem.dat)
set.seed(12345)
fit.v.all.raw <- lavaan::sem(lav.v.all, data = sem.dat)
```

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan
## WARNING: some observed variances are (at least) a factor 1000 times larger than
## others; use varTable(fit) to investigate
```

```r
set.seed(12345)
fit.l.all <- lavaan::sem(lav.l.all, data = apply(sem.dat, 2, std))
set.seed(12345)
fit.v.all <- lavaan::sem(lav.v.all, data = apply(sem.dat, 2, std))
set.seed(12345)
fit.l.rot.all <- lavaan::sem(lav.l.rot.all, data = apply(sem.dat, 2, std))
set.seed(12345)
fit.v.rot.all <- lavaan::sem(lav.v.rot.all, data = apply(sem.dat, 2, std))
fit.v.rot.norock <- lavaan::sem(lav.v.rot.norock, data = apply(sem.dat, 2, std))
fit.v.rot.norock.ind.litter <- lavaan::sem(lav.v.rot.norock.ind.litter,
                                           data = apply(sem.dat, 2, std))
set.seed(12345)
fit.l.rot.nolight <- lavaan::sem(lav.l.rot.nolight, data = apply(sem.dat, 2, std))
```

```r
set.seed(12345)
fit.v.rot.nolight <- lavaan::sem(lav.v.rot.nolight, data = apply(sem.dat, 2, std))

summary(fit.l.all.raw, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 121 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        31
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                18.541
##   Degrees of freedom                                13
##   P-value (Chi-square)                           0.138
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   light ~
##     crown            -0.005    0.024   -0.204    0.839
##   litter ~
##     crown             0.216    0.065    3.341    0.001
##   rock.lg ~
##     litter           -0.675    0.059  -11.495    0.000
##   l.A ~
##     light             0.016    0.065    0.239    0.811
##     rock.lg           0.092    0.027    3.417    0.001
##   l.R ~
##     light             0.088    0.060    1.478    0.139
##     rock.lg           0.162    0.025    6.518    0.000
##   l.D ~
##     light             0.022    0.013    1.615    0.106
##     rock.lg           0.031    0.006    5.661    0.000
##   l.X1 ~
##     light             0.029    0.040    0.709    0.479
##     rock.lg           0.037    0.017    2.244    0.025
##   l.X2 ~
##     light             0.025    0.034    0.736    0.462
##     rock.lg          -0.024    0.014   -1.697    0.090
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .l.A ~~
##    .l.R               4.023    1.799    2.236    0.025
##    .l.D              -0.127    0.368   -0.346    0.729
```

```
##   .l.R ~~
##     .l.D              1.363    0.420    3.250    0.001
##   .l.A ~~
##     .l.X1             4.221    1.347    3.132    0.002
##   .l.R ~~
##     .l.X1             2.448    1.111    2.204    0.028
##   .l.A ~~
##     .l.X2            -3.251    1.113   -2.919    0.004
##   .l.R ~~
##     .l.X2            -0.466    0.871   -0.534    0.593
##   .l.D ~~
##     .l.X1             0.048    0.227    0.213    0.832
##     .l.X2             0.196    0.197    0.994    0.320
##   .l.X1 ~~
##     .l.X2             0.297    0.586    0.507    0.612
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     .light           77.135   19.916    3.873    0.000
##     .litter         584.196  150.839    3.873    0.000
##     .rock.lg         83.027   21.438    3.873    0.000
##     .l.A              9.776    2.524    3.873    0.000
##     .l.R              8.276    2.137    3.873    0.000
##     .l.D              0.413    0.107    3.873    0.000
##     .l.X1             3.750    0.968    3.873    0.000
##     .l.X2             2.724    0.703    3.873    0.000
##
## R-Square:
##                    Estimate
##      light            0.001
##      litter           0.271
##      rock.lg          0.815
##      l.A              0.282
##      l.R              0.600
##      l.D              0.538
##      l.X1             0.157
##      l.X2             0.101
```

```
summary(fit.v.all.raw, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 235 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        40
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                12.147
##   Degrees of freedom                                14
##   P-value (Chi-square)                           0.595
##
## Parameter Estimates:
```

```
##
##   Standard errors                              Standard
##   Information                                  Expected
##   Information saturated (h1) model           Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   light ~
##     crown            -0.005    0.024   -0.204    0.839
##   litter ~
##     crown             0.216    0.065    3.341    0.001
##   rock.sm ~
##     litter           -0.312    0.060   -5.169    0.000
##   p.A ~
##     light             0.047    0.328    0.143    0.887
##     rock.sm          -0.477    0.224   -2.128    0.033
##   p.R ~
##     light             0.020    0.016    1.217    0.224
##     rock.sm          -0.046    0.012   -3.835    0.000
##     litter           -0.013    0.003   -4.027    0.000
##   p.D ~
##     light            -0.000    0.007   -0.071    0.944
##     rock.sm          -0.013    0.005   -2.704    0.007
##   p.X1 ~
##     light             0.118    0.200    0.589    0.556
##     rock.sm           0.093    0.136    0.684    0.494
##   p.X2 ~
##     light            -0.018    0.209   -0.086    0.931
##     rock.sm           0.164    0.142    1.151    0.250
##   p.X3 ~
##     light             0.191    0.248    0.771    0.441
##     rock.sm           0.356    0.169    2.108    0.035
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .p.A ~~
##    .p.X2            -89.124   33.272   -2.679    0.007
##    .p.R               5.236    2.439    2.147    0.032
##    .p.D               1.732    1.025    1.690    0.091
##  .p.R ~~
##    .p.D               0.229    0.064    3.602    0.000
##  .p.A ~~
##    .p.X1            -85.041   31.772   -2.677    0.007
##  .p.R ~~
##    .p.X1             -1.262    1.383   -0.913    0.361
##  .p.A ~~
##    .p.X3            -58.230   35.996   -1.618    0.106
##  .p.R ~~
##    .p.X2             -3.100    1.536   -2.019    0.044
##    .p.X3              0.231    1.692    0.137    0.891
##  .p.D ~~
##    .p.X1             -0.109    0.593   -0.184    0.854
##    .p.X2             -1.337    0.666   -2.007    0.045
##    .p.X3              0.241    0.736    0.328    0.743
```

```
##   .p.X1 ~~
##     .p.X2             8.347   17.700    0.472    0.637
##     .p.X3           -12.723   21.025   -0.605    0.545
##   .p.X2 ~~
##     .p.X3           -28.403   22.483   -1.263    0.206
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     .light           77.135   19.916    3.873    0.000
##     .litter         584.195  150.839    3.873    0.000
##     .rock.sm         87.816   22.674    3.873    0.000
##     .p.A            249.877   64.518    3.873    0.000
##     .p.R              0.605    0.156    3.873    0.000
##     .p.D              0.114    0.029    3.873    0.000
##     .p.X1            92.255   23.820    3.873    0.000
##     .p.X2           101.118   26.109    3.873    0.000
##     .p.X3           141.992   36.662    3.873    0.000
##
## R-Square:
##                    Estimate
##     light             0.001
##     litter            0.271
##     rock.sm           0.471
##     p.A               0.131
##     p.R               0.265
##     p.D               0.196
##     p.X1              0.027
##     p.X2              0.042
##     p.X3              0.145
```

```
summary(fit.l.all, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 58 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        31
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                18.541
##   Degrees of freedom                                13
##   P-value (Chi-square)                           0.138
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   light ~
```

```
##     crown             -0.037    0.182   -0.204    0.839
##   litter ~
##     crown              0.521    0.156    3.341    0.001
##   rock.lg ~
##     litter            -0.903    0.079  -11.495    0.000
##   l.A ~
##     light              0.037    0.154    0.239    0.811
##     rock.lg            0.528    0.154    3.417    0.001
##   l.R ~
##     light              0.168    0.114    1.478    0.139
##     rock.lg            0.742    0.114    6.518    0.000
##   l.D ~
##     light              0.197    0.122    1.615    0.106
##     rock.lg            0.691    0.122    5.661    0.000
##   l.X1 ~
##     light              0.118    0.167    0.709    0.479
##     rock.lg            0.374    0.167    2.244    0.025
##   l.X2 ~
##     light              0.128    0.174    0.736    0.462
##     rock.lg           -0.295    0.174   -1.697    0.090
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .l.A ~~
##    .l.R                0.228    0.102    2.236    0.025
##    .l.D               -0.035    0.100   -0.346    0.729
##  .l.R ~~
##    .l.D                0.297    0.091    3.250    0.001
##  .l.A ~~
##    .l.X1               0.520    0.166    3.132    0.002
##  .l.R ~~
##    .l.X1               0.242    0.110    2.204    0.028
##  .l.A ~~
##    .l.X2              -0.490    0.168   -2.919    0.004
##  .l.R ~~
##    .l.X2              -0.056    0.105   -0.534    0.593
##  .l.D ~~
##    .l.X1               0.023    0.108    0.213    0.832
##    .l.X2               0.114    0.114    0.994    0.320
##  .l.X1 ~~
##    .l.X2               0.078    0.154    0.507    0.612
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .light              0.965    0.249    3.873    0.000
##    .litter             0.705    0.182    3.873    0.000
##    .rock.lg            0.179    0.046    3.873    0.000
##    .l.A                0.691    0.178    3.873    0.000
##    .l.R                0.375    0.097    3.873    0.000
##    .l.D                0.432    0.112    3.873    0.000
##    .l.X1               0.806    0.208    3.873    0.000
##    .l.X2               0.876    0.226    3.873    0.000
##
## R-Square:
```

```
##                    Estimate
##     light            0.001
##     litter           0.271
##     rock.lg          0.815
##     l.A              0.282
##     l.R              0.600
##     l.D              0.538
##     l.X1             0.157
##     l.X2             0.101
```

```r
summary(fit.v.all, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 52 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        40
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                12.147
##   Degrees of freedom                                14
##   P-value (Chi-square)                           0.595
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   light ~
##     crown            -0.037    0.182   -0.204    0.839
##   litter ~
##     crown             0.521    0.156    3.341    0.001
##   rock.sm ~
##     litter           -0.686    0.133   -5.169    0.000
##   p.A ~
##     light             0.024    0.170    0.143    0.887
##     rock.sm          -0.363    0.170   -2.128    0.033
##   p.R ~
##     light             0.184    0.152    1.217    0.224
##     rock.sm          -0.634    0.165   -3.835    0.000
##     litter           -0.386    0.096   -4.027    0.000
##   p.D ~
##     light            -0.012    0.164   -0.071    0.944
##     rock.sm          -0.442    0.164   -2.704    0.007
##   p.X1 ~
##     light             0.106    0.180    0.589    0.556
##     rock.sm           0.123    0.180    0.684    0.494
##   p.X2 ~
##     light            -0.015    0.179   -0.086    0.931
```

```
##     rock.sm           0.206    0.179    1.151    0.250
##   p.X3 ~
##     light             0.129    0.167    0.771    0.441
##     rock.sm           0.353    0.167    2.108    0.035
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .p.A ~~
##    .p.X2             -0.495    0.185   -2.679    0.007
##    .p.R               0.319    0.149    2.147    0.032
##    .p.D               0.262    0.155    1.690    0.091
##  .p.R ~~
##    .p.D               0.628    0.174    3.602    0.000
##  .p.A ~~
##    .p.X1             -0.497    0.186   -2.677    0.007
##  .p.R ~~
##    .p.X1             -0.134    0.146   -0.913    0.361
##  .p.A ~~
##    .p.X3             -0.256    0.158   -1.618    0.106
##  .p.R ~~
##    .p.X2             -0.312    0.154   -2.019    0.044
##    .p.X3              0.018    0.134    0.137    0.891
##  .p.D ~~
##    .p.X1             -0.029    0.156   -0.184    0.854
##    .p.X2             -0.334    0.166   -2.007    0.045
##    .p.X3              0.048    0.145    0.328    0.743
##  .p.X1 ~~
##    .p.X2              0.080    0.171    0.472    0.637
##    .p.X3             -0.097    0.160   -0.605    0.545
##  .p.X2 ~~
##    .p.X3             -0.206    0.163   -1.263    0.206
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .light             0.965    0.249    3.873    0.000
##    .litter            0.705    0.182    3.873    0.000
##    .rock.sm           0.511    0.132    3.873    0.000
##    .p.A               0.842    0.218    3.873    0.000
##    .p.R               0.667    0.172    3.873    0.000
##    .p.D               0.775    0.200    3.873    0.000
##    .p.X1              0.936    0.242    3.873    0.000
##    .p.X2              0.927    0.239    3.873    0.000
##    .p.X3              0.812    0.210    3.873    0.000
##
## R-Square:
##                    Estimate
##     light             0.001
##     litter            0.271
##     rock.sm           0.471
##     p.A               0.131
##     p.R               0.265
##     p.D               0.196
##     p.X1              0.027
##     p.X2              0.042
```

```
##     p.X3              0.145
```

```r
summary(fit.l.rot.all, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 58 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        33
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                26.681
##   Degrees of freedom                                19
##   P-value (Chi-square)                           0.112
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   light ~
##     crown            -0.052    0.202   -0.256    0.798
##   litter ~
##     crown             0.594    0.170    3.506    0.000
##   light ~
##     trunk             0.034    0.202    0.169    0.866
##   litter ~
##     trunk            -0.172    0.170   -1.016    0.309
##   rock.lg ~
##     litter           -0.903    0.079  -11.495    0.000
##   l.A ~
##     light             0.037    0.154    0.239    0.811
##     rock.lg           0.528    0.154    3.417    0.001
##   l.R ~
##     light             0.168    0.114    1.478    0.139
##     rock.lg           0.742    0.114    6.518    0.000
##   l.D ~
##     light             0.197    0.122    1.615    0.106
##     rock.lg           0.691    0.122    5.661    0.000
##   rot.l.X1 ~
##     light             0.051    0.161    0.320    0.749
##     rock.lg           0.462    0.161    2.873    0.004
##   rot.l.X2 ~
##     light             0.174    0.180    0.966    0.334
##     rock.lg          -0.023    0.180   -0.130    0.897
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .l.A ~~
```

```
##    .l.R               0.228    0.102    2.236    0.025
##    .l.D              -0.035    0.100   -0.346    0.729
##  .l.R ~~
##    .l.D               0.297    0.091    3.250    0.001
##  .l.A ~~
##    .rot.l.X1          0.677    0.181    3.751    0.000
##  .l.R ~~
##    .rot.l.X1          0.241    0.106    2.266    0.023
##  .l.A ~~
##    .rot.l.X2         -0.098    0.148   -0.662    0.508
##  .l.R ~~
##    .rot.l.X2          0.095    0.110    0.866    0.386
##  .l.D ~~
##    .rot.l.X1         -0.028    0.104   -0.269    0.788
##    .rot.l.X2          0.106    0.118    0.904    0.366
##  .rot.l.X1 ~~
##    .rot.l.X2          0.154    0.156    0.987    0.324
##
## Variances:
##                   Estimate  Std.Err  z-value  P(>|z|)
##    .light             0.964    0.249    3.873    0.000
##    .litter            0.681    0.176    3.873    0.000
##    .rock.lg           0.179    0.046    3.873    0.000
##    .l.A               0.691    0.178    3.873    0.000
##    .l.R               0.375    0.097    3.873    0.000
##    .l.D               0.432    0.112    3.873    0.000
##    .rot.l.X1          0.751    0.194    3.873    0.000
##    .rot.l.X2          0.938    0.242    3.873    0.000
##
## R-Square:
##                   Estimate
##     light            0.002
##     litter           0.295
##     rock.lg          0.815
##     l.A              0.282
##     l.R              0.601
##     l.D              0.539
##     rot.l.X1         0.219
##     rot.l.X2         0.031
```

```r
summary(fit.v.rot.all, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 48 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        41
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                30.762
##   Degrees of freedom                                22
##   P-value (Chi-square)                           0.101
```

```
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   light ~
##     crown            -0.052    0.202   -0.256    0.798
##   litter ~
##     crown             0.594    0.170    3.506    0.000
##   light ~
##     trunk             0.034    0.202    0.169    0.866
##   litter ~
##     trunk            -0.172    0.170   -1.016    0.309
##   rock.sm ~
##     litter           -0.686    0.133   -5.169    0.000
##   p.A ~
##     light             0.024    0.170    0.143    0.887
##     rock.sm          -0.363    0.170   -2.127    0.033
##   p.R ~
##     light             0.197    0.169    1.165    0.244
##     rock.sm          -0.372    0.169   -2.202    0.028
##   p.D ~
##     light            -0.012    0.164   -0.071    0.944
##     rock.sm          -0.442    0.164   -2.704    0.007
##   rot.p.X1 ~
##     light             0.142    0.160    0.888    0.375
##     rock.sm           0.433    0.160    2.711    0.007
##   rot.p.X2 ~
##     light             0.063    0.182    0.344    0.731
##     rock.sm          -0.013    0.182   -0.071    0.944
##   rot.p.X3 ~
##     light             0.070    0.182    0.387    0.699
##     rock.sm          -0.014    0.182   -0.080    0.937
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .p.A ~~
##    .rot.p.X2         -0.344    0.176   -1.953    0.051
##    .p.R               0.363    0.166    2.187    0.029
##    .p.D               0.262    0.155    1.690    0.091
##  .p.R ~~
##    .p.D               0.672    0.191    3.524    0.000
##  .p.A ~~
##    .rot.p.X1         -0.511    0.172   -2.973    0.003
##  .p.R ~~
##    .rot.p.X1         -0.115    0.144   -0.799    0.425
##  .p.A ~~
##    .rot.p.X3          0.300    0.173    1.731    0.083
##  .p.R ~~
##    .rot.p.X2         -0.134    0.165   -0.814    0.416
```

```
##     .rot.p.X3          0.311    0.172    1.807    0.071
##   .p.D ~~
##     .rot.p.X1         -0.068    0.139   -0.486    0.627
##     .rot.p.X2         -0.008    0.158   -0.048    0.962
##     .rot.p.X3          0.300    0.167    1.798    0.072
##   .rot.p.X1 ~~
##     .rot.p.X2         -0.107    0.155   -0.687    0.492
##     .rot.p.X3          0.215    0.159    1.350    0.177
##   .rot.p.X2 ~~
##     .rot.p.X3         -0.181    0.179   -1.011    0.312
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .light             0.964    0.249    3.873    0.000
##    .litter            0.681    0.176    3.873    0.000
##    .rock.sm           0.511    0.132    3.873    0.000
##    .p.A               0.842    0.218    3.873    0.000
##    .p.R               0.825    0.213    3.873    0.000
##    .p.D               0.775    0.200    3.873    0.000
##    .rot.p.X1          0.741    0.191    3.873    0.000
##    .rot.p.X2          0.963    0.249    3.873    0.000
##    .rot.p.X3          0.962    0.248    3.873    0.000
##
## R-Square:
##                    Estimate
##     light             0.002
##     litter            0.295
##     rock.sm           0.471
##     p.A               0.131
##     p.R               0.170
##     p.D               0.196
##     rot.p.X1          0.215
##     rot.p.X2          0.004
##     rot.p.X3          0.005
```

```r
summary(fit.v.rot.norock, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 42 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        39
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                15.966
##   Degrees of freedom                                13
##   P-value (Chi-square)                           0.251
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
```

```
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   light ~
##     crown            -0.052    0.202   -0.256    0.798
##   litter ~
##     crown             0.594    0.170    3.506    0.000
##   light ~
##     trunk             0.034    0.202    0.169    0.866
##   litter ~
##     trunk            -0.172    0.170   -1.016    0.309
##   p.A ~
##     light            -0.029    0.181   -0.162    0.871
##     litter            0.122    0.181    0.676    0.499
##   p.R ~
##     light             0.107    0.181    0.594    0.553
##     litter           -0.074    0.181   -0.412    0.680
##   p.D ~
##     light            -0.073    0.179   -0.408    0.684
##     litter            0.173    0.179    0.967    0.333
##   rot.p.X1 ~
##     light             0.184    0.171    1.077    0.282
##     litter           -0.278    0.171   -1.628    0.104
##   rot.p.X2 ~
##     light             0.077    0.181    0.425    0.670
##     litter            0.100    0.181    0.554    0.580
##   rot.p.X3 ~
##     light             0.049    0.181    0.270    0.787
##     litter           -0.108    0.181   -0.597    0.550
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .p.A ~~
##    .rot.p.X2         -0.351    0.185   -1.895    0.058
##    .p.R               0.496    0.195    2.539    0.011
##    .p.D               0.391    0.186    2.104    0.035
##  .p.R ~~
##    .p.D               0.836    0.229    3.646    0.000
##  .p.A ~~
##    .rot.p.X1         -0.624    0.199   -3.134    0.002
##  .p.R ~~
##    .rot.p.X1         -0.284    0.171   -1.657    0.097
##  .p.A ~~
##    .rot.p.X3          0.317    0.183    1.733    0.083
##  .p.R ~~
##    .rot.p.X2         -0.123    0.175   -0.700    0.484
##    .rot.p.X3          0.309    0.182    1.694    0.090
##  .p.D ~~
##    .rot.p.X1         -0.200    0.166   -1.208    0.227
##    .rot.p.X2         -0.019    0.172   -0.109    0.914
##    .rot.p.X3          0.324    0.181    1.783    0.075
##  .rot.p.X1 ~~
##    .rot.p.X2         -0.086    0.164   -0.521    0.602
```

```
##    .rot.p.X3          0.181    0.167    1.084    0.279
##  .rot.p.X2 ~~
##    .rot.p.X3         -0.170    0.177   -0.965    0.335
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .light             0.964    0.249    3.873    0.000
##    .litter            0.681    0.176    3.873    0.000
##    .p.A               0.950    0.245    3.873    0.000
##    .p.R               0.948    0.245    3.873    0.000
##    .p.D               0.928    0.240    3.873    0.000
##    .rot.p.X1          0.843    0.218    3.873    0.000
##    .rot.p.X2          0.954    0.246    3.873    0.000
##    .rot.p.X3          0.951    0.246    3.873    0.000
##
## R-Square:
##                    Estimate
##     light             0.002
##     litter            0.295
##     p.A               0.016
##     p.R               0.017
##     p.D               0.036
##     rot.p.X1          0.115
##     rot.p.X2          0.016
##     rot.p.X3          0.014
```

```r
summary(fit.l.rot.nolight, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 54 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        24
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                17.024
##   Degrees of freedom                                11
##   P-value (Chi-square)                           0.107
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   litter ~
##     crown             0.521    0.156    3.341    0.001
##   rock.lg ~
##     litter           -0.903    0.079  -11.495    0.000
##   l.A ~
```

```
##     rock.lg           0.533    0.155    3.446    0.001
##   l.R ~
##     rock.lg           0.764    0.118    6.489    0.000
##   l.D ~
##     rock.lg           0.718    0.127    5.643    0.000
##   rot.l.X1 ~
##     rock.lg           0.469    0.161    2.911    0.004
##   rot.l.X2 ~
##     rock.lg           0.000    0.183    0.000    1.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .l.A ~~
##    .l.R               0.234    0.105    2.218    0.027
##    .l.D              -0.028    0.104   -0.266    0.790
##  .l.R ~~
##    .l.D               0.328    0.099    3.303    0.001
##  .l.A ~~
##    .rot.l.X1          0.679    0.181    3.751    0.000
##  .l.R ~~
##    .rot.l.X1          0.250    0.110    2.261    0.024
##  .l.A ~~
##    .rot.l.X2         -0.092    0.150   -0.612    0.541
##  .l.R ~~
##    .rot.l.X2          0.123    0.116    1.058    0.290
##  .l.D ~~
##    .rot.l.X1         -0.018    0.109   -0.169    0.866
##    .rot.l.X2          0.139    0.126    1.107    0.268
##  .rot.l.X1 ~~
##    .rot.l.X2          0.162    0.159    1.023    0.306
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .litter            0.705    0.182    3.873    0.000
##    .rock.lg           0.179    0.046    3.873    0.000
##    .l.A               0.692    0.179    3.873    0.000
##    .l.R               0.402    0.104    3.873    0.000
##    .l.D               0.469    0.121    3.873    0.000
##    .rot.l.X1          0.754    0.195    3.873    0.000
##    .rot.l.X2          0.967    0.250    3.873    0.000
##
## R-Square:
##                    Estimate
##     litter            0.271
##     rock.lg           0.815
##     l.A               0.284
##     l.R               0.584
##     l.D               0.515
##     rot.l.X1          0.220
##     rot.l.X2          0.000
```

```
summary(fit.v.rot.nolight, rsquare = TRUE)
```

```
## lavaan 0.6-8 ended normally after 41 iterations
##
```

```
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        31
##
##   Number of observations                            30
##
## Model Test User Model:
##
##   Test statistic                                20.525
##   Degrees of freedom                                13
##   P-value (Chi-square)                           0.083
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   litter ~
##     crown             0.521    0.156    3.341    0.001
##   rock.sm ~
##     litter           -0.686    0.133   -5.169    0.000
##   p.A ~
##     rock.sm          -0.358    0.170   -2.098    0.036
##   p.R ~
##     rock.sm          -0.331    0.172   -1.921    0.055
##   p.D ~
##     rock.sm          -0.445    0.164   -2.719    0.007
##   rot.p.X1 ~
##     rock.sm           0.463    0.162    2.859    0.004
##   rot.p.X2 ~
##     rock.sm          -0.000    0.183   -0.000    1.000
##   rot.p.X3 ~
##     rock.sm           0.000    0.183    0.000    1.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .p.A ~~
##    .rot.p.X2         -0.342    0.176   -1.942    0.052
##    .p.R               0.367    0.169    2.169    0.030
##    .p.D               0.262    0.155    1.688    0.091
##  .p.R ~~
##    .p.D               0.670    0.193    3.474    0.001
##  .p.A ~~
##    .rot.p.X1         -0.508    0.173   -2.933    0.003
##  .p.R ~~
##    .rot.p.X1         -0.089    0.149   -0.602    0.547
##  .p.A ~~
##    .rot.p.X3          0.301    0.174    1.735    0.083
##  .p.R ~~
##    .rot.p.X2         -0.123    0.168   -0.730    0.466
##    .rot.p.X3          0.324    0.177    1.834    0.067
```

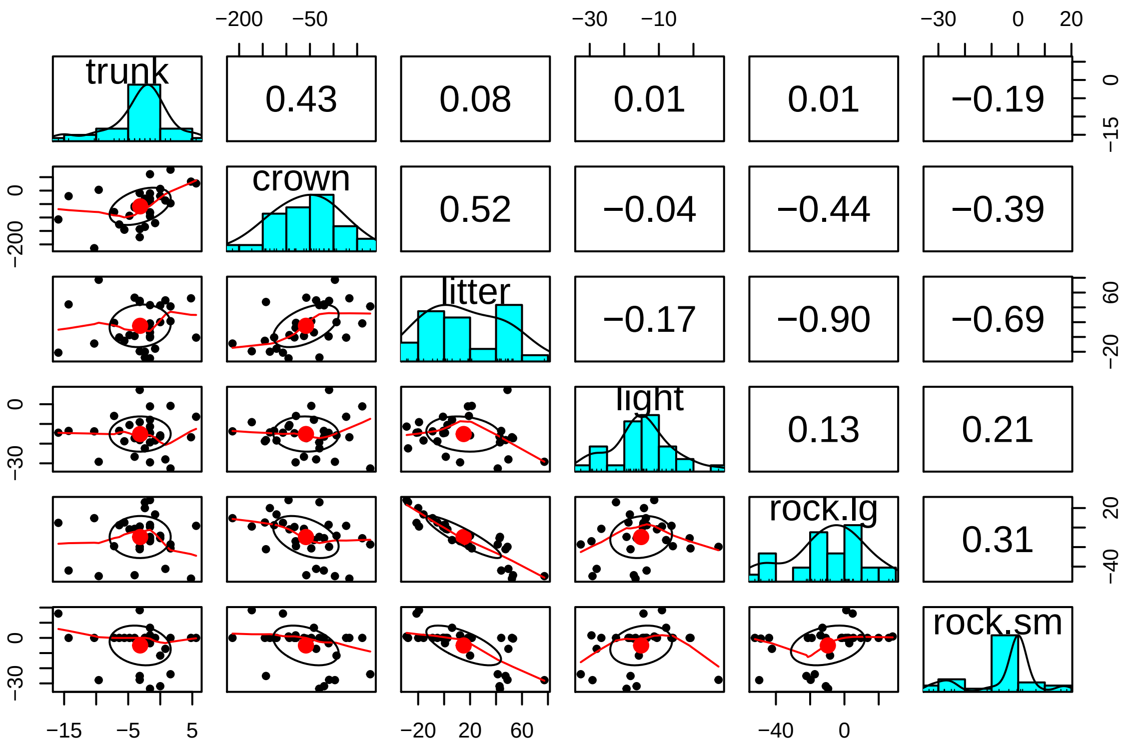
```
##   .p.D ~~
##     .rot.p.X1         -0.069    0.141   -0.491    0.623
##     .rot.p.X2         -0.008    0.158   -0.052    0.958
##     .rot.p.X3          0.299    0.167    1.790    0.074
##   .rot.p.X1 ~~
##     .rot.p.X2         -0.099    0.157   -0.626    0.532
##     .rot.p.X3          0.224    0.162    1.385    0.166
##   .rot.p.X2 ~~
##     .rot.p.X3         -0.177    0.179   -0.985    0.325
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .litter            0.705    0.182    3.873    0.000
##    .rock.sm           0.511    0.132    3.873    0.000
##    .p.A               0.843    0.218    3.873    0.000
##    .p.R               0.861    0.222    3.873    0.000
##    .p.D               0.776    0.200    3.873    0.000
##    .rot.p.X1          0.760    0.196    3.873    0.000
##    .rot.p.X2          0.967    0.250    3.873    0.000
##    .rot.p.X3          0.967    0.250    3.873    0.000
##
## R-Square:
##                    Estimate
##     litter            0.271
##     rock.sm           0.471
##     p.A               0.128
##     p.R               0.110
##     p.D               0.198
##     rot.p.X1          0.214
##     rot.p.X2          0.000
##     rot.p.X3          0.000
```

SEM Variable R-Squares

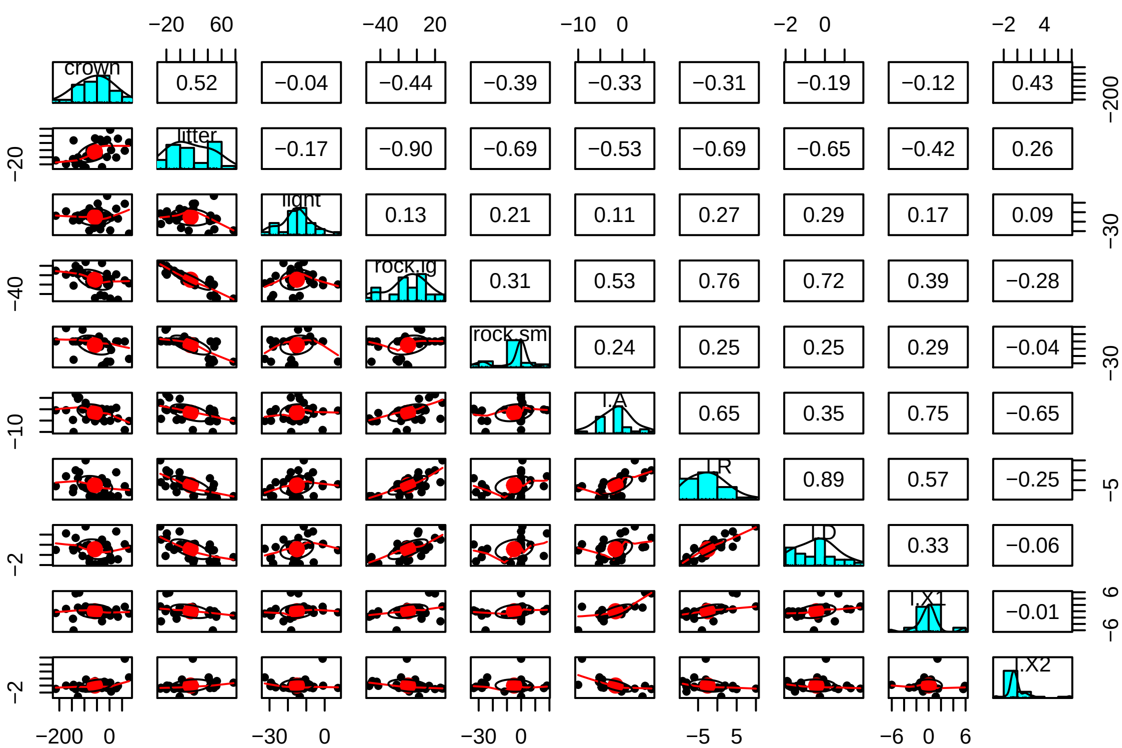
```r
get_R2 <- function(x){
    out <- capture.output(summary(x, rsquare = TRUE))
    out <- out[grep("R-Square:",out):length(out)]
    out <- out[!(grepl("R-Square:", out)) & !(grepl("Estimate", out))]
    out <- out[out != ""]
    out <- strsplit(out, " ")
    out <- lapply(out, function(x) x[x != ""])
    out <- do.call(rbind, out)
    out.names <- out[, 1]
    out <- as.numeric(out[, 2])
    names(out) <- out.names
    return(out)
}

r2.l.rot.all <- get_R2(fit.l.rot.all)
r2.v.rot.all <- get_R2(fit.v.rot.all)
r2.v.rot.norock <- get_R2(fit.v.rot.norock)
```

SEM variable inter-correlations plot

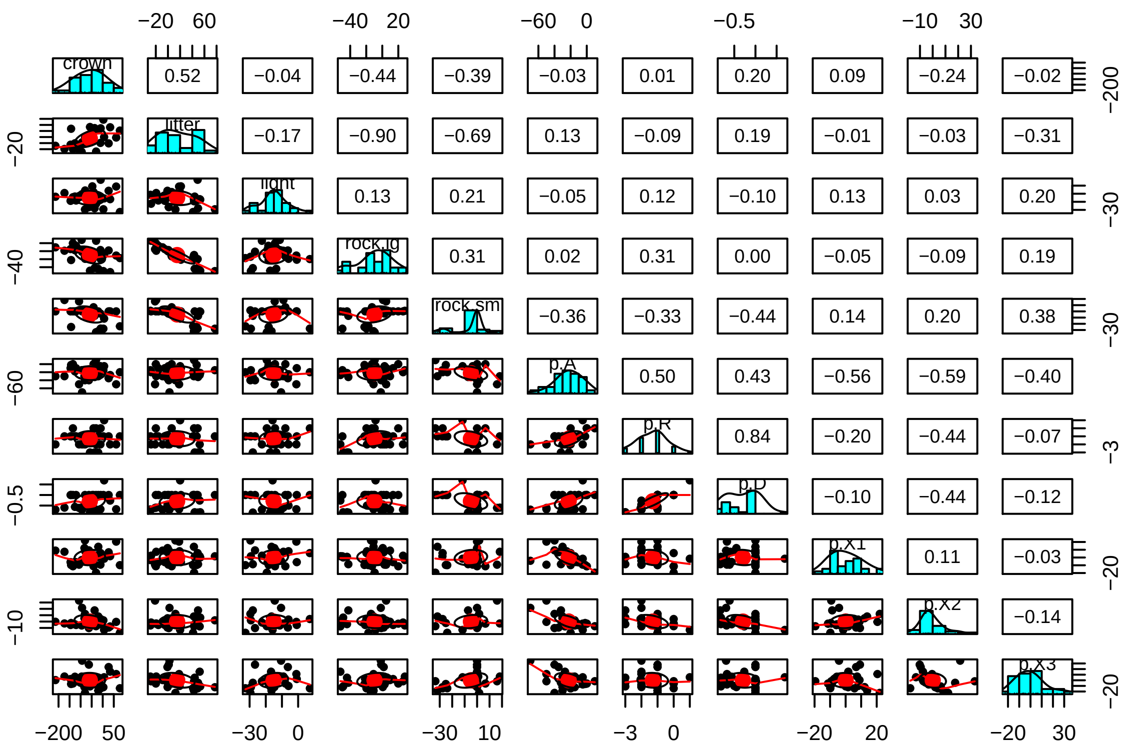
```
pairs.panels(sem.dat[, c("trunk", "crown", "litter", "light", "rock.lg", "rock.sm")])
```



```
pairs.panels(sem.dat[, c("crown", "litter", "light", "rock.lg", "rock.sm",
                         "l.A", "l.R", "l.D", "l.X1", "l.X2")])
```



```r
pairs.panels(sem.dat[, c("crown", "litter", "light", "rock.lg", "rock.sm",
                         "p.A", "p.R", "p.D", "p.X1", "p.X2", "p.X3")])
```



SEM Skew-Kurtosis Check

```r
skc.sem <- skew_kurtosis(sem.dat)
skc.sem.flags <- skc.sem[(abs(skc.sem[, "skew_2se"]) > 1 |
                          abs(skc.sem[, "kurt_2se"]) > 1), ]
kable(skc.sem.flags)
```

	skew	skew_2se	kurt	kurt_2se
l.X1	0.3854049	0.4514075	2.683946	1.6115042
l.X2	2.6079952	3.0546283	9.344848	5.6108660
p.X2	1.2498696	1.4639165	1.705619	1.0240934
rot.l.X2	1.9574132	2.2926307	5.705369	3.4256372
rot.p.X1	1.1461696	1.3424572	1.444625	0.8673866

SEM Modification Indices

```r
kable(modindices(fit.l.rot.all))
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
34	crown	~~	crown	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
35	crown	~~	trunk	0.0000000	0.0000000	0.0000000	NA	0.0000000
36	trunk	~~	trunk	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
37	light	~~	litter	0.9253730	-0.1423451	-0.1423451	-0.1756296	-0.1756296
38	light	~~	rock.lg	0.0765092	-0.0209739	-0.0209739	-0.0505006	-0.0505006

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
39	light	~~	l.A	3.8139378	1.0607456	1.0607456	1.2991988	1.2991988
40	light	~~	l.R	0.6087020	-0.8964530	-0.8964530	-1.4900542	-1.4900542
41	light	~~	l.D	0.0070214	-0.1167580	-0.1167580	-0.1808793	-0.1808793
42	light	~~	rot.l.X1	4.2635667	-1.1805998	-1.1805998	-1.3870428	-1.3870428
43	light	~~	rot.l.X2	6.4181608	4.6145175	4.6145175	4.8516938	4.8516938
44	litter	~~	rock.lg	0.0192385	-0.0162625	-0.0162625	-0.0465935	-0.0465935
45	litter	~~	l.A	2.4897792	-0.0475173	-0.0475173	-0.0692528	-0.0692528
46	litter	~~	l.R	2.4132029	0.0989621	0.0989621	0.1957332	0.1957332
47	litter	~~	l.D	1.2942229	-0.0878875	-0.0878875	-0.1620133	-0.1620133
48	litter	~~	rot.l.X1	1.8801842	0.0434674	0.0434674	0.0607675	0.0607675
49	litter	~~	rot.l.X2	4.5054948	-0.2143578	-0.2143578	-0.2681806	-0.2681806
50	rock.lg	~~	l.A	0.5422132	0.0082109	0.0082109	0.0233527	0.0233527
51	rock.lg	~~	l.R	0.6194903	0.0185662	0.0185662	0.0716605	0.0716605
52	rock.lg	~~	l.D	0.5294596	-0.0208148	-0.0208148	-0.0748785	-0.0748785
53	rock.lg	~~	rot.l.X1	1.0563577	-0.0120643	-0.0120643	-0.0329133	-0.0329133
54	rock.lg	~~	rot.l.X2	0.4209804	0.0242624	0.0242624	0.0592355	0.0592355
55	light	~	litter	0.9253727	-0.2089867	-0.2089867	-0.2089867	-0.2089867
56	light	~	rock.lg	0.4900536	0.1465075	0.1465075	0.1465075	0.1465075
57	light	~	l.A	0.3589030	0.2368901	0.2368901	0.2363719	0.2363719
58	light	~	l.R	0.4161302	0.1818842	0.1818842	0.1793174	0.1793174
59	light	~	l.D	0.4464054	0.2020992	0.2020992	0.1989812	0.1989812
60	light	~	rot.l.X1	0.3542422	0.2681741	0.2681741	0.2674558	0.2674558
61	light	~	rot.l.X2	0.9264304	3.7340793	3.7340793	3.7357769	3.7357769
62	litter	~	light	0.9253729	-0.1475967	-0.1475967	-0.1475967	-0.1475967
63	litter	~	rock.lg	0.0192361	-0.0909198	-0.0909198	-0.0909198	-0.0909198
64	litter	~	l.A	0.0545386	0.0631888	0.0631888	0.0630505	0.0630505
65	litter	~	l.R	0.0005491	0.0075966	0.0075966	0.0074894	0.0074894
66	litter	~	l.D	0.4223160	-0.1972020	-0.1972020	-0.1941595	-0.1941595
67	litter	~	rot.l.X1	0.0084626	-0.0240029	-0.0240029	-0.0239386	-0.0239386
68	litter	~	rot.l.X2	3.6210617	-0.4381403	-0.4381403	-0.4383395	-0.4383395
69	rock.lg	~	light	0.0687830	-0.0206025	-0.0206025	-0.0206025	-0.0206025
70	rock.lg	~	l.A	0.5182769	-0.0739799	-0.0739799	-0.0738181	-0.0738181
71	rock.lg	~	l.R	0.0000099	-0.0004202	-0.0004202	-0.0004143	-0.0004143
72	rock.lg	~	l.D	0.0010939	0.0040913	0.0040913	0.0040281	0.0040281
73	rock.lg	~	rot.l.X1	0.9296071	-0.0949520	-0.0949520	-0.0946977	-0.0946977
74	rock.lg	~	rot.l.X2	0.1070795	-0.0283634	-0.0283634	-0.0283763	-0.0283763
75	rock.lg	~	crown	0.2251407	0.0436470	0.0436470	0.0436470	0.0443932
76	rock.lg	~	trunk	1.0683474	0.0814475	0.0814475	0.0814475	0.0828398
77	l.A	~	litter	0.5422038	0.0414436	0.0414436	0.0415344	0.0415344
82	l.A	~	crown	6.6330433	0.0706717	0.0706717	0.0708267	0.0720375
83	l.A	~	trunk	0.9789875	0.0240210	0.0240210	0.0240736	0.0244852
84	l.R	~	litter	0.6194853	0.0937114	0.0937114	0.0950528	0.0950528
89	l.R	~	crown	1.3606611	-0.0677119	-0.0677119	-0.0686812	-0.0698553
90	l.R	~	trunk	0.3367190	-0.0298014	-0.0298014	-0.0302280	-0.0307447
91	l.D	~	litter	0.5294622	-0.1050618	-0.1050618	-0.1067081	-0.1067081
96	l.D	~	crown	1.2520321	0.0787678	0.0787678	0.0800020	0.0813697
97	l.D	~	trunk	2.5961539	0.1003502	0.1003502	0.1019227	0.1036651
98	rot.l.X1	~	litter	1.0563433	-0.0608934	-0.0608934	-0.0610570	-0.0610570
103	rot.l.X1	~	crown	6.4206172	-0.0731930	-0.0731930	-0.0733896	-0.0746442
104	rot.l.X1	~	trunk	0.6268203	-0.0202332	-0.0202332	-0.0202876	-0.0206344
105	rot.l.X2	~	litter	0.4209743	0.1224617	0.1224617	0.1224061	0.1224061
110	rot.l.X2	~	crown	9.1991034	0.2790993	0.2790993	0.2789724	0.2837415

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
111	rot.l.X2	~	trunk	0.7556307	0.0707706	0.0707706	0.0707385	0.0719478
112	crown	~	light	0.0000000	0.0059830	0.0059830	0.0059830	0.0059830
113	crown	~	litter	0.0000000	-0.0002645	-0.0002645	-0.0002645	-0.0002645
114	crown	~	rock.lg	0.0021982	-0.0212571	-0.0212571	-0.0212571	-0.0212571
115	crown	~	l.A	1.2520562	-0.2418542	-0.2418542	-0.2413252	-0.2413252
116	crown	~	l.R	0.5813503	-0.2070305	-0.2070305	-0.2041088	-0.2041088
117	crown	~	l.D	0.0001803	-0.0034837	-0.0034837	-0.0034299	-0.0034299
118	crown	~	rot.l.X1	0.9462834	-0.2035827	-0.2035827	-0.2030374	-0.2030374
119	crown	~	rot.l.X2	2.6919234	0.3145204	0.3145204	0.3146634	0.3146634
120	crown	~	trunk	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
121	trunk	~	light	0.0000000	-0.0178932	-0.0178932	-0.0178932	-0.0178932
122	trunk	~	litter	0.0000000	-0.0006573	-0.0006573	-0.0006573	-0.0006573
123	trunk	~	rock.lg	0.9199921	0.3722825	0.3722825	0.3722825	0.3722825
124	trunk	~	l.A	1.4211601	0.2270328	0.2270328	0.2265362	0.2265362
125	trunk	~	l.R	5.1014816	0.5381001	0.5381001	0.5305063	0.5305063
126	trunk	~	l.D	5.6259503	0.5406119	0.5406119	0.5322714	0.5322714
127	trunk	~	rot.l.X1	1.1149690	0.1948187	0.1948187	0.1942969	0.1942969
128	trunk	~	rot.l.X2	0.0132654	-0.0194927	-0.0194927	-0.0195015	-0.0195015
129	trunk	~	crown	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000

```
kable(modindices(fit.v.rot.all))
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
42	crown	~~	crown	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
43	crown	~~	trunk	0.0000000	0.0000000	0.0000000	NA	0.0000000
44	trunk	~~	trunk	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
45	light	~~	litter	0.9253744	-0.1423451	-0.1423451	-0.1756298	-0.1756298
46	light	~~	rock.sm	0.4714777	0.0880291	0.0880291	0.1253632	0.1253632
47	light	~~	p.A	2.2167382	-2.1334059	-2.1334059	-2.3668039	-2.3668039
48	light	~~	p.R	1.4811007	-2.2191517	-2.2191517	-2.4879305	-2.4879305
49	light	~~	p.D	0.4049198	1.1572340	1.1572340	1.3381526	1.3381526
50	light	~~	rot.p.X1	1.4543537	-1.8132886	-1.8132886	-2.1448505	-2.1448505
51	light	~~	rot.p.X2	0.7667922	-2.3489600	-2.3489600	-2.4373701	-2.4373701
52	light	~~	rot.p.X3	1.1832121	2.7072615	2.7072615	2.8105610	2.8105610
53	litter	~~	rock.sm	0.0028827	-0.0106433	-0.0106433	-0.0180360	-0.0180360
54	litter	~~	p.A	1.9035536	0.0920829	0.0920829	0.1215594	0.1215594
55	litter	~~	p.R	4.8655856	-0.1873452	-0.1873452	-0.2499278	-0.2499278
56	litter	~~	p.D	2.0893398	0.1224395	0.1224395	0.1684716	0.1684716
57	litter	~~	rot.p.X1	0.9418090	0.0679663	0.0679663	0.0956631	0.0956631
58	litter	~~	rot.p.X2	0.2918095	0.0674942	0.0674942	0.0833361	0.0833361
59	litter	~~	rot.p.X3	2.1741563	-0.1709325	-0.1709325	-0.2111583	-0.2111583
60	rock.sm	~~	p.A	0.0032929	0.0039517	0.0039517	0.0060211	0.0060211
61	rock.sm	~~	p.R	9.4849724	-0.2698923	-0.2698923	-0.4155763	-0.4155763
62	rock.sm	~~	p.D	4.5254970	0.1859292	0.1859292	0.2952848	0.2952848
63	rock.sm	~~	rot.p.X1	0.0189776	-0.0099548	-0.0099548	-0.0161722	-0.0161722
64	rock.sm	~~	rot.p.X2	0.0231067	-0.0195967	-0.0195967	-0.0279279	-0.0279279
65	rock.sm	~~	rot.p.X3	0.0118401	-0.0130153	-0.0130153	-0.0185578	-0.0185578
66	light	~	litter	0.9253768	-0.2089871	-0.2089871	-0.2089873	-0.2089873
67	light	~	rock.sm	1.2895086	0.2231951	0.2231951	0.2231953	0.2231953
68	light	~	p.A	1.7616629	-0.7137450	-0.7137450	-0.7149366	-0.7149366
69	light	~	p.R	1.8172585	-0.7080027	-0.7080027	-0.7177260	-0.7177260

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
70	light	~	p.D	1.5523085	-0.5510927	-0.5510927	-0.5505584	-0.5505584
71	light	~	rot.p.X1	1.4005260	0.5340416	0.5340416	0.5277861	0.5277861
72	light	~	rot.p.X2	0.0579317	0.9385102	0.9385102	0.9386540	0.9386540
73	light	~	rot.p.X3	0.5445146	-2.8540016	-2.8540016	-2.8545554	-2.8545554
74	litter	~	light	0.9253741	-0.1475969	-0.1475969	-0.1475968	-0.1475968
75	litter	~	rock.sm	0.0028822	-0.0208157	-0.0208157	-0.0208157	-0.0208157
76	litter	~	p.A	0.0000133	-0.0007198	-0.0007198	-0.0007210	-0.0007210
77	litter	~	p.R	3.5540127	-0.3639516	-0.3639516	-0.3689496	-0.3689496
78	litter	~	p.D	0.4962395	-0.1434317	-0.1434317	-0.1432925	-0.1432925
79	litter	~	rot.p.X1	0.2807778	-0.1083684	-0.1083684	-0.1070990	-0.1070990
80	litter	~	rot.p.X2	0.0245579	0.0293540	0.0293540	0.0293584	0.0293584
81	litter	~	rot.p.X3	1.8414407	-0.2541131	-0.2541131	-0.2541622	-0.2541622
82	rock.sm	~	light	0.4420515	0.0883055	0.0883055	0.0883054	0.0883054
83	rock.sm	~	p.A	0.8477791	-0.1906872	-0.1906872	-0.1910054	-0.1910054
84	rock.sm	~	p.R	5.0625582	-0.4501048	-0.4501048	-0.4562861	-0.4562861
85	rock.sm	~	p.D	1.0419524	-0.2204629	-0.2204629	-0.2202490	-0.2202490
86	rock.sm	~	rot.p.X1	0.0674534	0.0558543	0.0558543	0.0552000	0.0552000
87	rock.sm	~	rot.p.X2	0.5333280	0.1409747	0.1409747	0.1409962	0.1409962
88	rock.sm	~	rot.p.X3	0.6264974	-0.1527010	-0.1527010	-0.1527306	-0.1527306
89	rock.sm	~	crown	0.0849920	-0.0453408	-0.0453408	-0.0453408	-0.0461159
90	rock.sm	~	trunk	1.0080751	-0.1337643	-0.1337643	-0.1337643	-0.1360511
91	p.A	~	litter	0.0032925	0.0053048	0.0053048	0.0052960	0.0052960
97	p.A	~	crown	3.2974919	-0.1307777	-0.1307777	-0.1305597	-0.1327917
98	p.A	~	trunk	0.3572878	-0.0402494	-0.0402494	-0.0401823	-0.0408693
99	p.R	~	litter	9.4849646	-0.3623277	-0.3623277	-0.3574194	-0.3574194
105	p.R	~	crown	2.0765087	-0.1320648	-0.1320648	-0.1302757	-0.1325028
106	p.R	~	trunk	0.1823532	-0.0365920	-0.0365920	-0.0360962	-0.0367133
107	p.D	~	litter	4.5255211	0.2496088	0.2496088	0.2498513	0.2498513
113	p.D	~	crown	1.7400093	0.1205696	0.1205696	0.1206867	0.1227498
114	p.D	~	trunk	1.0552353	0.0877901	0.0877901	0.0878753	0.0893776
115	rot.p.X1	~	litter	0.0189792	-0.0133647	-0.0133647	-0.0135231	-0.0135231
121	rot.p.X1	~	crown	2.4883291	-0.1192096	-0.1192096	-0.1206226	-0.1226846
122	rot.p.X1	~	trunk	0.4040789	-0.0449158	-0.0449158	-0.0454482	-0.0462251
123	rot.p.X2	~	litter	0.0231094	-0.0263099	-0.0263099	-0.0263059	-0.0263059
129	rot.p.X2	~	crown	0.8164665	-0.1218236	-0.1218236	-0.1218050	-0.1238873
130	rot.p.X2	~	trunk	0.0138774	-0.0148500	-0.0148500	-0.0148477	-0.0151015
131	rot.p.X3	~	litter	0.0118395	-0.0174725	-0.0174725	-0.0174691	-0.0174691
137	rot.p.X3	~	crown	4.8456188	0.2753588	0.2753588	0.2753055	0.2800119
138	rot.p.X3	~	trunk	2.8221563	0.1964824	0.1964824	0.1964443	0.1998026
139	crown	~	light	0.0000005	-0.0418068	-0.0418068	-0.0418068	-0.0418068
140	crown	~	litter	0.0000000	-0.0006309	-0.0006309	-0.0006309	-0.0006309
141	crown	~	rock.sm	0.0553444	0.0632445	0.0632445	0.0632445	0.0632445
142	crown	~	p.A	2.3007661	-0.2796266	-0.2796266	-0.2800934	-0.2800934
143	crown	~	p.R	2.4359309	-0.2901203	-0.2901203	-0.2941046	-0.2941046
144	crown	~	p.D	0.5920379	-0.1451116	-0.1451116	-0.1449709	-0.1449709
145	crown	~	rot.p.X1	0.4320157	0.1267677	0.1267677	0.1252827	0.1252827
146	crown	~	rot.p.X2	0.3932952	0.1116361	0.1116361	0.1116532	0.1116532
147	crown	~	rot.p.X3	0.0198633	0.0251006	0.0251006	0.0251055	0.0251055
148	crown	~	trunk	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
149	trunk	~	light	0.0000018	0.1678111	0.1678111	0.1678110	0.1678110
150	trunk	~	litter	0.0000000	-0.0019588	-0.0019588	-0.0019588	-0.0019588
151	trunk	~	rock.sm	1.0028627	-0.2299916	-0.2299916	-0.2299917	-0.2299917

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
152	trunk	~	p.A	1.9526134	0.2392522	0.2392522	0.2396516	0.2396516
153	trunk	~	p.R	4.0591922	0.3476922	0.3476922	0.3524670	0.3524670
154	trunk	~	p.D	4.6662093	0.3771134	0.3771134	0.3767476	0.3767476
155	trunk	~	rot.p.X1	0.1754520	-0.0747795	-0.0747795	-0.0739035	-0.0739035
156	trunk	~	rot.p.X2	0.0029738	-0.0090704	-0.0090704	-0.0090718	-0.0090718
157	trunk	~	rot.p.X3	3.7033485	0.3202415	0.3202415	0.3203035	0.3203035
158	trunk	~	crown	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000

```
kable(modindices(fit.v.rot.norock.ind.litter))
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
40	crown	~~	crown	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
41	crown	~~	trunk	0.0000000	0.0000000	0.0000000	NA	0.0000000
42	trunk	~~	trunk	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
43	light	~~	litter	0.9574770	-0.1453263	-0.1453263	-0.1785911	-0.1785911
44	light	~~	p.A	2.4977478	-2.1401060	-2.1401060	-2.2178956	-2.2178956
45	light	~~	p.R	1.6116148	-2.1840163	-2.1840163	-2.2783814	-2.2783814
46	light	~~	p.D	1.1850466	1.9339623	1.9339623	2.0128205	2.0128205
47	light	~~	rot.p.X1	2.6234405	-2.3709319	-2.3709319	-2.5201940	-2.5201940
48	light	~~	rot.p.X2	1.0361404	-2.5820014	-2.5820014	-2.6788477	-2.6788477
49	light	~~	rot.p.X3	1.2397111	2.6082783	2.6082783	2.7061802	2.7061802
50	light	~	litter	0.9179865	-0.2073190	-0.2073190	-0.1999227	-0.1999227
51	light	~	p.A	1.3330903	-1.0308362	-1.0308362	-1.0302918	-1.0302918
52	light	~	p.R	0.0133394	-0.3269785	-0.3269785	-0.3266952	-0.3266952
53	light	~	p.D	1.0622249	-1.1538373	-1.1538373	-1.1514333	-1.1514333
54	light	~	rot.p.X1	0.8702730	0.5691343	0.5691343	0.5637485	0.5637485
55	light	~	rot.p.X2	0.1345684	-0.9318035	-0.9318035	-0.9324185	-0.9324185
56	light	~	rot.p.X3	0.7096730	0.8432189	0.8432189	0.8422646	0.8422646
57	litter	~	light	0.9574770	-0.1506879	-0.1506879	-0.1562627	-0.1562627
58	litter	~	p.A	0.9576394	10.1567189	10.1567189	10.5269135	10.5269135
59	litter	~	p.R	0.9574401	-1.3406412	-1.3406412	-1.3890348	-1.3890348
60	litter	~	p.D	0.9574954	2.0151564	2.0151564	2.0853552	2.0853552
61	litter	~	rot.p.X1	0.9574816	-0.8448638	-0.8448638	-0.8678296	-0.8678296
62	litter	~	rot.p.X2	0.9574852	-2.1686239	-2.1686239	-2.2503386	-2.2503386
63	litter	~	rot.p.X3	0.9574062	-4.1840113	-4.1840113	-4.3338922	-4.3338922
64	p.A	~	litter	3.4565293	-0.2685807	-0.2685807	-0.2591356	-0.2591356
70	p.A	~	crown	3.5615195	-0.1277330	-0.1277330	-0.1278004	-0.1299852
71	p.A	~	trunk	0.3959225	-0.0425623	-0.0425623	-0.0425848	-0.0433128
72	p.R	~	litter	2.1533806	-0.2693328	-0.2693328	-0.2599493	-0.2599493
78	p.R	~	crown	2.1663585	-0.1265737	-0.1265737	-0.1266834	-0.1288491
79	p.R	~	trunk	0.1922313	-0.0376845	-0.0376845	-0.0377172	-0.0383620
80	p.D	~	litter	2.4024594	0.2937729	0.2937729	0.2838837	0.2838837
86	p.D	~	crown	3.0898460	0.1560994	0.1560994	0.1564253	0.1590994
87	p.D	~	trunk	1.2749503	0.1002193	0.1002193	0.1004285	0.1021453
88	rot.p.X1	~	litter	3.7491927	-0.3023743	-0.3023743	-0.2943724	-0.2943724
94	rot.p.X1	~	crown	3.9467125	-0.1453525	-0.1453525	-0.1467410	-0.1492496
95	rot.p.X1	~	trunk	0.5252320	-0.0529921	-0.0529921	-0.0534983	-0.0544129
96	rot.p.X2	~	litter	1.2045572	-0.2969979	-0.2969979	-0.2862133	-0.2862133
102	rot.p.X2	~	crown	1.0933983	-0.1325737	-0.1325737	-0.1324862	-0.1347511
103	rot.p.X2	~	trunk	0.0216673	-0.0186510	-0.0186510	-0.0186386	-0.0189573
104	rot.p.X3	~	litter	3.2240137	0.4487315	0.4487315	0.4332128	0.4332128

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
110	rot.p.X3	~	crown	4.6445170	0.2523426	0.2523426	0.2526284	0.2569472
111	rot.p.X3	~	trunk	2.8821770	0.1986637	0.1986637	0.1988887	0.2022888
112	crown	~	light	0.0000011	0.0710311	0.0710311	0.0710311	0.0710311
113	crown	~	litter	0.7903036	0.3266272	0.3266272	0.3149746	0.3149746
114	crown	~	p.A	0.4048304	-0.1051622	-0.1051622	-0.1051067	-0.1051067
115	crown	~	p.R	0.4383524	-0.1101584	-0.1101584	-0.1100630	-0.1100630
116	crown	~	p.D	0.0420285	0.0340294	0.0340294	0.0339585	0.0339585
117	crown	~	rot.p.X1	0.0974335	-0.0529132	-0.0529132	-0.0524125	-0.0524125
118	crown	~	rot.p.X2	0.3731262	0.1010774	0.1010774	0.1011441	0.1011441
119	crown	~	rot.p.X3	0.0229465	0.0250659	0.0250659	0.0250375	0.0250375
120	crown	~	trunk	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
121	trunk	~	light	0.0000051	-0.3220724	-0.3220724	-0.3220725	-0.3220725
122	trunk	~	litter	0.0299576	-0.0635756	-0.0635756	-0.0613075	-0.0613075
123	trunk	~	p.A	1.3044468	0.1886589	0.1886589	0.1885594	0.1885594
124	trunk	~	p.R	3.0147832	0.2887119	0.2887119	0.2884619	0.2884619
125	trunk	~	p.D	3.2508146	0.2990989	0.2990989	0.2984758	0.2984758
126	trunk	~	rot.p.X1	0.0205052	-0.0242602	-0.0242602	-0.0240307	-0.0240307
127	trunk	~	rot.p.X2	0.0039620	-0.0104092	-0.0104092	-0.0104160	-0.0104160
128	trunk	~	rot.p.X3	3.6469683	0.3158122	0.3158122	0.3154549	0.3154549
129	trunk	~	crown	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000

```
kable(modindices(fit.v.rot.norock))
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
40	crown	~~	crown	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
41	crown	~~	trunk	0.0000000	0.0000000	0.0000000	NA	0.0000000
42	trunk	~~	trunk	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
43	light	~~	litter	0.9253761	-0.1423451	-0.1423451	-0.1756299	-0.1756299
44	light	~~	p.A	2.7067361	-2.5661488	-2.5661488	-2.6807157	-2.6807157
45	light	~~	p.R	0.0004979	0.0387333	0.0387333	0.0405190	0.0405190
46	light	~~	p.D	0.0473781	-0.3892301	-0.3892301	-0.4113507	-0.4113507
47	light	~~	rot.p.X1	1.5947109	-2.0863600	-2.0863600	-2.3142877	-2.3142877
48	light	~~	rot.p.X2	0.7809072	-2.5736308	-2.5736308	-2.6835094	-2.6835094
49	light	~~	rot.p.X3	1.5097778	3.3216057	3.3216057	3.4678385	3.4678385
50	litter	~~	p.A	4.1210614	0.2046187	0.2046187	0.2543518	0.2543518
51	litter	~~	p.R	0.0254310	0.0178888	0.0178888	0.0222677	0.0222677
52	litter	~~	p.D	0.0586232	-0.0279791	-0.0279791	-0.0351852	-0.0351852
53	litter	~~	rot.p.X1	2.7772999	0.1779269	0.1779269	0.2348498	0.2348498
54	litter	~~	rot.p.X2	0.9976306	0.1879807	0.1879807	0.2332334	0.2332334
55	litter	~~	rot.p.X3	4.4295096	-0.3676642	-0.3676642	-0.4567542	-0.4567542
56	light	~	litter	0.9253803	-0.2089873	-0.2089873	-0.2089877	-0.2089877
57	light	~	p.A	1.9927797	-2.3241897	-2.3241897	-2.3229612	-2.3229612
58	light	~	p.R	0.0729725	0.6572014	0.6572014	0.6564280	0.6564280
59	light	~	p.D	1.3883205	-1.4234846	-1.4234846	-1.4208369	-1.4208369
60	light	~	rot.p.X1	1.1222991	0.8177216	0.8177216	0.8115633	0.8115633
61	light	~	rot.p.X2	0.5301042	-1.4127543	-1.4127543	-1.4143692	-1.4143692
62	light	~	rot.p.X3	0.6911663	1.5190024	1.5190024	1.5178223	1.5178223
63	litter	~	light	0.9253758	-0.1475972	-0.1475972	-0.1475969	-0.1475969
64	litter	~	p.A	1.1523470	0.3050716	0.3050716	0.3049098	0.3049098
65	litter	~	p.R	0.0011097	0.0093110	0.0093110	0.0093000	0.0093000
66	litter	~	p.D	0.0167033	0.0368688	0.0368688	0.0368001	0.0368001

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
67	litter	~	rot.p.X1	0.8211030	-0.2575303	-0.2575303	-0.2555903	-0.2555903
68	litter	~	rot.p.X2	0.2890875	-0.1512052	-0.1512052	-0.1513777	-0.1513777
69	litter	~	rot.p.X3	1.1002189	-0.2971375	-0.2971375	-0.2969061	-0.2969061
75	p.A	~	crown	4.1473144	-0.1610012	-0.1610012	-0.1610866	-0.1638404
76	p.A	~	trunk	0.3928545	-0.0424343	-0.0424343	-0.0424568	-0.0431826
82	p.R	~	crown	0.0744294	-0.0240036	-0.0240036	-0.0240319	-0.0244428
83	p.R	~	trunk	0.1463388	-0.0288230	-0.0288230	-0.0288570	-0.0293503
89	p.D	~	crown	0.3038455	0.0499609	0.0499609	0.0500541	0.0509098
90	p.D	~	trunk	0.9712247	0.0764926	0.0764926	0.0766352	0.0779453
96	rot.p.X1	~	crown	3.0180536	-0.1454784	-0.1454784	-0.1465825	-0.1490884
97	rot.p.X1	~	trunk	0.5009895	-0.0507580	-0.0507580	-0.0511432	-0.0520175
103	rot.p.X2	~	crown	0.8962825	-0.1397513	-0.1397513	-0.1395919	-0.1419783
104	rot.p.X2	~	trunk	0.0210633	-0.0183464	-0.0183464	-0.0183255	-0.0186388
110	rot.p.X3	~	crown	6.0844065	0.3379773	0.3379773	0.3382405	0.3440228
111	rot.p.X3	~	trunk	2.8878923	0.1993995	0.1993995	0.1995548	0.2029663
112	crown	~	light	0.0000016	-0.0743735	-0.0743735	-0.0743734	-0.0743734
113	crown	~	litter	0.0000000	-0.0008096	-0.0008096	-0.0008096	-0.0008096
114	crown	~	p.A	1.4297525	-0.2361358	-0.2361358	-0.2360107	-0.2360107
115	crown	~	p.R	0.2764706	-0.1039834	-0.1039834	-0.1038609	-0.1038609
116	crown	~	p.D	0.1364636	-0.0738025	-0.0738025	-0.0736652	-0.0736652
117	crown	~	rot.p.X1	0.4272900	0.1370648	0.1370648	0.1360324	0.1360324
118	crown	~	rot.p.X2	0.1419960	0.0742779	0.0742779	0.0743627	0.0743627
119	crown	~	rot.p.X3	0.3196673	0.1115895	0.1115895	0.1115027	0.1115027
120	crown	~	trunk	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
121	trunk	~	light	0.0000062	0.3107492	0.3107492	0.3107488	0.3107488
122	trunk	~	litter	0.0000001	-0.0027069	-0.0027069	-0.0027069	-0.0027069
123	trunk	~	p.A	1.6244704	0.2148225	0.2148225	0.2147086	0.2147086
124	trunk	~	p.R	2.8843572	0.2866514	0.2866514	0.2863137	0.2863137
125	trunk	~	p.D	4.0222155	0.3419735	0.3419735	0.3413370	0.3413370
126	trunk	~	rot.p.X1	0.1673554	-0.0732147	-0.0732147	-0.0726633	-0.0726633
127	trunk	~	rot.p.X2	0.0004855	0.0037068	0.0037068	0.0037110	0.0037110
128	trunk	~	rot.p.X3	3.4180286	0.3114251	0.3114251	0.3111828	0.3111828
129	trunk	~	crown	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000

SEM Parameter Estimates

```
xtable::xtable(table_results(fit.l.all))
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:20 2021

```
xtable::xtable(table_results(fit.v.all))
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:20 2021

```
xtable::xtable(table_results(fit.l.rot.all))
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:21 2021

```
xtable::xtable(table_results(fit.v.rot.all))
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:21 2021

	label	est_sig	se	pval	confint
1	light.ON.crown	-0.04	0.18	0.84	[-0.39, 0.32]
2	litter.ON.crown	0.52***	0.16	0.00	[0.22, 0.83]
3	rock.lg.ON.litter	-0.90***	0.08	0.00	[-1.06, -0.75]
4	l.A.ON.light	0.04	0.15	0.81	[-0.27, 0.34]
5	l.A.ON.rock.lg	0.53***	0.15	0.00	[0.22, 0.83]
6	l.R.ON.light	0.17	0.11	0.14	[-0.05, 0.39]
7	l.R.ON.rock.lg	0.74***	0.11	0.00	[0.52, 0.96]
8	l.D.ON.light	0.20	0.12	0.11	[-0.04, 0.44]
9	l.D.ON.rock.lg	0.69***	0.12	0.00	[0.45, 0.93]
10	l.X1.ON.light	0.12	0.17	0.48	[-0.21, 0.45]
11	l.X1.ON.rock.lg	0.37*	0.17	0.02	[0.05, 0.70]
12	l.X2.ON.light	0.13	0.17	0.46	[-0.21, 0.47]
13	l.X2.ON.rock.lg	-0.30	0.17	0.09	[-0.64, 0.05]
14	l.A.WITH.l.R	0.23*	0.10	0.03	[0.03, 0.43]
15	l.A.WITH.l.D	-0.03	0.10	0.73	[-0.23, 0.16]
16	l.R.WITH.l.D	0.30**	0.09	0.00	[0.12, 0.48]
17	l.A.WITH.l.X1	0.52**	0.17	0.00	[0.19, 0.85]
18	l.R.WITH.l.X1	0.24*	0.11	0.03	[0.03, 0.46]
19	Variances.light	0.97***	0.25	0.00	[0.48, 1.45]
20	Variances.litter	0.70***	0.18	0.00	[0.35, 1.06]
21	Variances.rock.lg	0.18***	0.05	0.00	[0.09, 0.27]
22	Variances.l.A	0.69***	0.18	0.00	[0.34, 1.04]
23	Variances.l.R	0.38***	0.10	0.00	[0.19, 0.57]
24	Variances.l.D	0.43***	0.11	0.00	[0.21, 0.65]
25	Variances.l.X1	0.81***	0.21	0.00	[0.40, 1.21]
26	Variances.l.X2	0.88***	0.23	0.00	[0.43, 1.32]
27	l.A.WITH.l.X2	-0.49**	0.17	0.00	[-0.82, -0.16]
28	l.R.WITH.l.X2	-0.06	0.11	0.59	[-0.26, 0.15]
29	l.D.WITH.l.X1	0.02	0.11	0.83	[-0.19, 0.23]
30	l.D.WITH.l.X2	0.11	0.11	0.32	[-0.11, 0.34]
31	l.X1.WITH.l.X2	0.08	0.15	0.61	[-0.22, 0.38]
32	Variances.crown	0.97	0.00		[0.97, 0.97]

SEM Model Fit Measures

```r
sem.fm.l.all <- fitMeasures(fit.l.all)
sem.fm.v.all <- fitMeasures(fit.v.all)

sem.fm.l.rot.all <- fitMeasures(fit.l.rot.all)
sem.fm.v.rot.all <- fitMeasures(fit.v.rot.all)
sem.fm.v.rot.norock <- fitMeasures(fit.v.rot.norock)

sem.fm.tab.all <- rbind(sem.fm.l.all[c("chisq", "df", "pvalue")],
                        sem.fm.v.all[c("chisq", "df", "pvalue")])
rownames(sem.fm.tab.all) <- c("Lichens", "Plants")
colnames(sem.fm.tab.all) <- c("$\\chi^{2}$", "\\textit{df}", "\\textit{p}-value")
sem.fm.tab.rot.all <- rbind(sem.fm.l.rot.all[c("chisq", "df", "pvalue")],
                        sem.fm.v.rot.all[c("chisq", "df", "pvalue")])
rownames(sem.fm.tab.rot.all) <- c("Lichens", "Plants")
colnames(sem.fm.tab.rot.all) <- c("$\\chi^{2}$", "\\textit{df}", "\\textit{p}-value")
sem.fm.tab.rot.norock <- rbind(sem.fm.l.rot.all[c("chisq", "df", "pvalue")],
                        sem.fm.v.rot.norock[c("chisq", "df", "pvalue")])
```

```r
rownames(sem.fm.tab.rot.norock) <- c("Lichens", "Plants")
colnames(sem.fm.tab.rot.norock) <- c("$\\chi^{2}$", "\\textit{df}", "\\textit{p}-value")

print(xtable::xtable(sem.fm.tab.all, digits = 3),
                     sanitize.text.function = function(x) {x})
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:21 2021

```r
print(xtable::xtable(sem.fm.tab.rot.all, digits = 3),
      sanitize.text.function = function(x) {x})
```

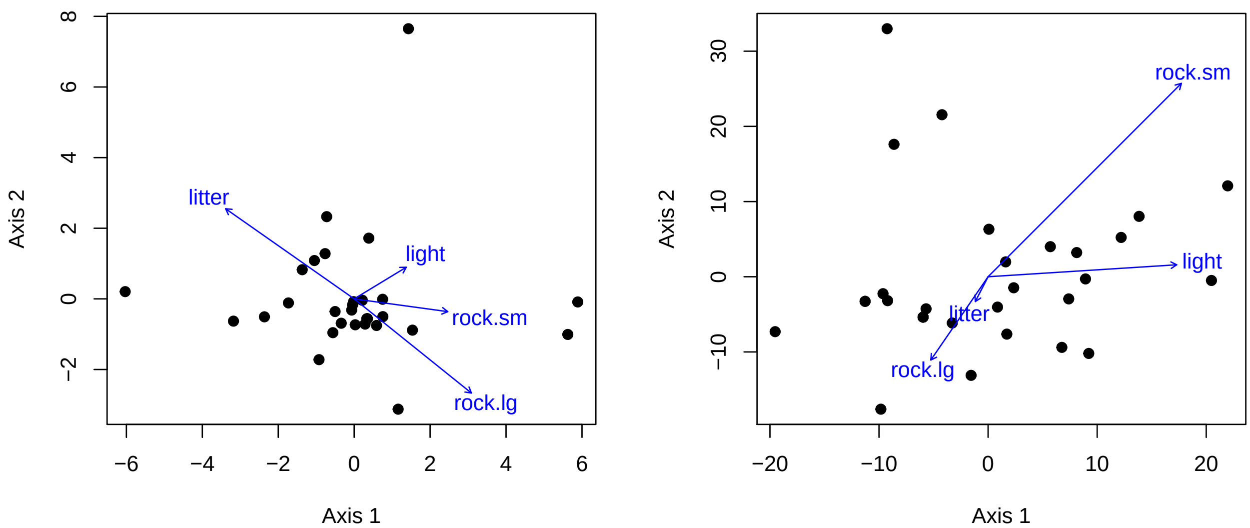
% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:21 2021

```r
print(xtable::xtable(sem.fm.tab.rot.norock, digits = 3),
                     sanitize.text.function = function(x) {x})
```

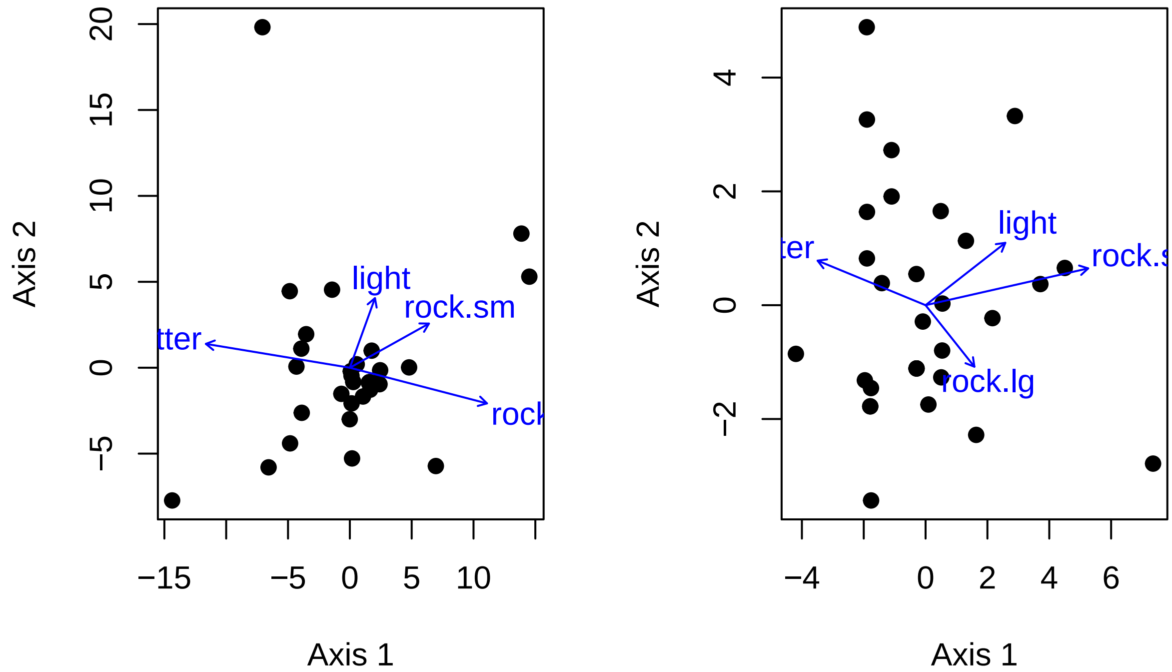
% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:21 2021

Ordination Plots

```r
par(mfrow = c(1,2))
plot(l.com.dif.ord[, 1:2], xlab = "Axis 1", ylab = "Axis 2", pch = 19)
plot(l.com.dif.vec, add = TRUE)
plot(v.com.dif.ord[, 1:2], xlab = "Axis 1", ylab = "Axis 2", pch = 19)
plot(v.com.dif.vec, add = TRUE)
```




```r
par(mfrow = c(1,2))
plot(l.com.dif.ord.proc[, 1:2], xlab = "Axis 1", ylab = "Axis 2", pch = 19)
plot(l.com.dif.vec.rot, add = TRUE)
plot(v.com.dif.ord.proc[, 1:2], xlab = "Axis 1", ylab = "Axis 2", pch = 19)
plot(v.com.dif.vec.rot, add = TRUE)
```



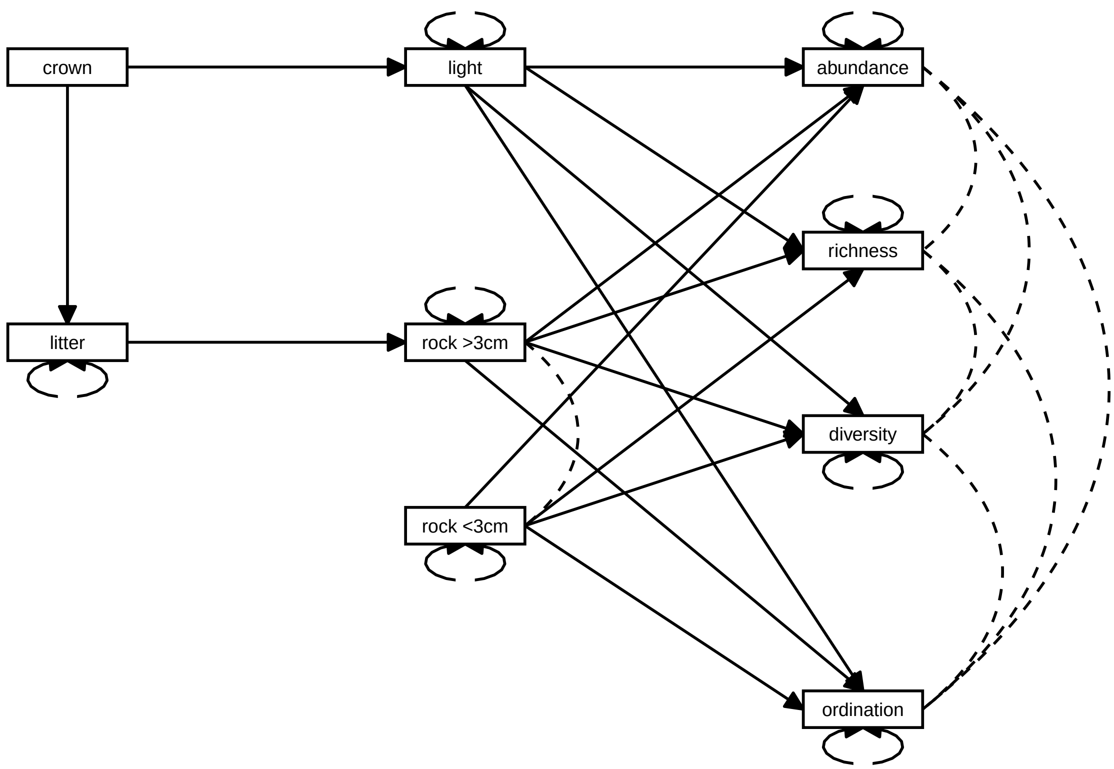
SEM Plots

```
apriori.dat <- data.frame(crown = rnorm(30),
                          light = rnorm(30),
                          litter = rnorm(30),
                          rock.lg = rnorm(30),
                          rock.sm = rnorm(30),
                          richness = rnorm(30),
                          abundance = rnorm(30),
                          diversity = rnorm(30),
                          ordination = rnorm(30))
lav.apriori <- 'light ~ crown
                litter ~ crown
                rock.lg ~ litter
                abundance ~ light + rock.lg + rock.sm
                richness ~ light + rock.lg + rock.sm
                diversity ~ light + rock.lg + rock.sm
                ordination ~ light + rock.lg + rock.sm
                rock.lg ~~ rock.sm
               '
fit.apriori <- lavaan::sem(lav.apriori, data = apriori.dat)
```

```
## Warning in lav_partable_vnames(FLAT, "ov.x", warn = TRUE): lavaan WARNING:
##     model syntax contains variance/covariance/intercept formulas
##     involving (an) exogenous variable(s): [rock.sm]; These variables
##     will now be treated as random introducing additional free
##     parameters. If you wish to treat those variables as fixed, remove
##     these formulas from the model syntax. Otherwise, consider adding
##     the fixed.x = FALSE option.
```

```
lay.apriori <- get_layout("crown", "", "light", "",     "abundance",
                  "", "",       "", "", "",
```

```r
                  "", "",        "", "",          "richness",
                  "litter", "", "rock.lg", "", "",
                  "", "",        "", "", "diversity",
                  "", "", "rock.sm", "", "",
                  "", "",        "", "", "",
                  "", "",     "", "",     "ordination",
                  rows = 8)

tg.apriori <- prepare_graph(fit.apriori,
                            layout = lay.apriori,
                            text_size = 2.5)
nodes(tg.apriori)[nodes(tg.apriori)[, "name"] ==
                    "rock.lg", "label"] <- "rock >3cm"
nodes(tg.apriori)[nodes(tg.apriori)[, "name"] ==
                  "rock.sm", "label"] <- "rock <3cm"
edges(tg.apriori)[, "label"] <- ""

plot(tg.apriori)
```

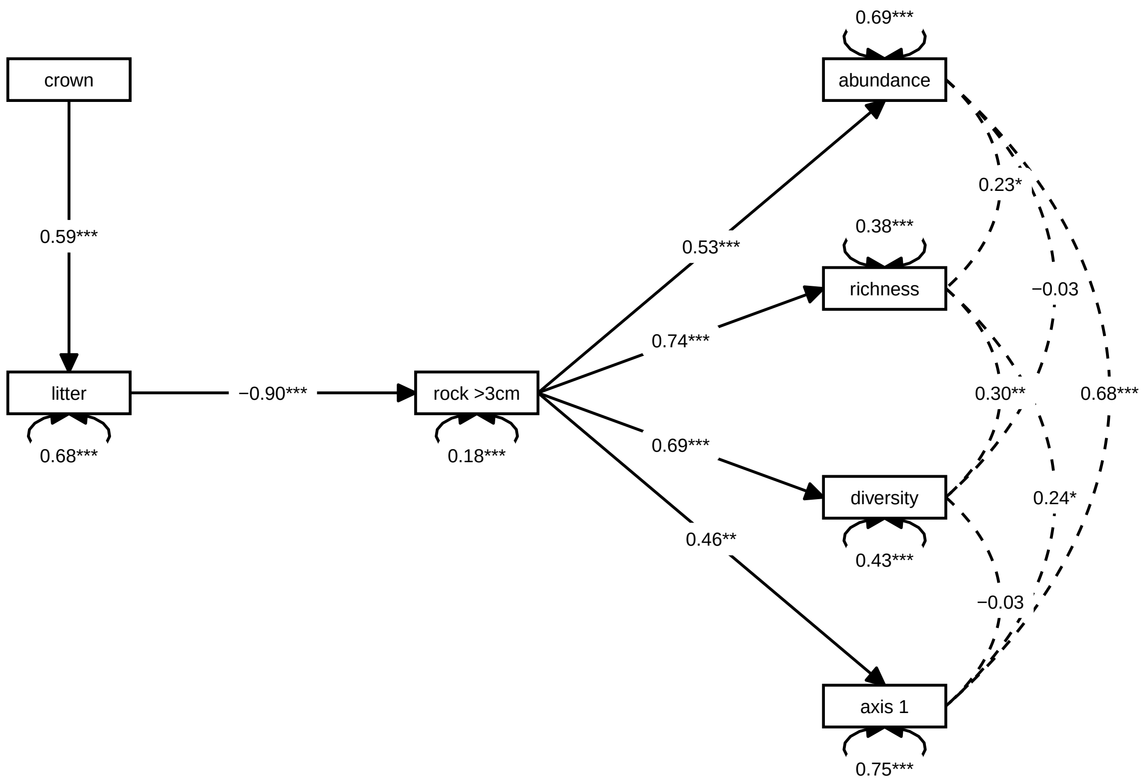



```r
lay <- get_layout("crown", "", "", "",      "l.A",
                  "", "",        "", "", "",
                  "", "",        "", "",          "l.R",
                  "litter", "", "rock.lg", "", "",
                  "", "",        "", "", "l.D",
                  "", "", "", "", "",
                  "", "",        "", "",     "rot.l.X1",
                  rows = 7)
tg.l.rot.all <- prepare_graph(fit.l.rot.all,
                              layout = lay,
```

```
                              text_size = 2.6)
```

```
## Some edges involve nodes not in layout. These were dropped.
```

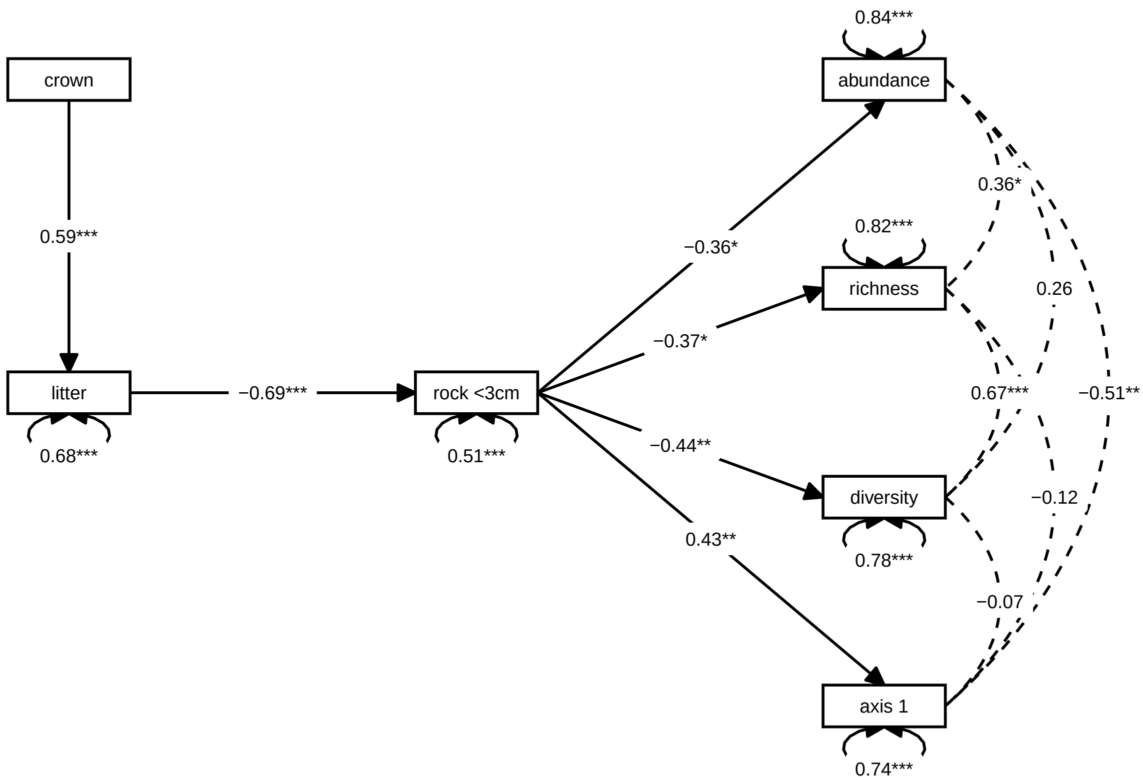
```
nodes(tg.l.rot.all)[nodes(tg.l.rot.all)[, "name"] ==
                    "l.A", "label"] <- "abundance"
nodes(tg.l.rot.all)[nodes(tg.l.rot.all)[, "name"] ==
                    "l.R", "label"] <- "richness"
nodes(tg.l.rot.all)[nodes(tg.l.rot.all)[, "name"] ==
                    "l.D", "label"] <- "diversity"
nodes(tg.l.rot.all)[nodes(tg.l.rot.all)[, "name"] ==
                    "rock.lg", "label"] <- "rock >3cm"
nodes(tg.l.rot.all)[nodes(tg.l.rot.all)[, "name"] ==
                    "rot.l.X1", "label"] <- "axis 1"
plot(tg.l.rot.all)
```




```
lay <- get_layout("crown", "", "", "",     "p.A",
                  "", "",       "", "", "",
                  "", "",       "", "",          "p.R",
                  "litter", "", "rock.sm", "", "",
                  "", "",       "", "", "p.D",
                  "", "", "", "", "",
                  "", "",       "", "",    "rot.p.X1",
                  rows = 7)
tg.v.rot.all <- prepare_graph(fit.v.rot.all,
                              layout = lay,
                              text_size = 2.6)
```

```
## Some edges involve nodes not in layout. These were dropped.
```

```r
nodes(tg.v.rot.all)[nodes(tg.v.rot.all)[, "name"] ==
                    "p.A", "label"] <- "abundance"
nodes(tg.v.rot.all)[nodes(tg.v.rot.all)[, "name"] ==
                    "p.R", "label"] <- "richness"
nodes(tg.v.rot.all)[nodes(tg.v.rot.all)[, "name"] ==
                    "p.D", "label"] <- "diversity"
nodes(tg.v.rot.all)[nodes(tg.v.rot.all)[, "name"] ==
                    "rock.sm", "label"] <- "rock <3cm"
nodes(tg.v.rot.all)[nodes(tg.v.rot.all)[, "name"] ==
                    "rot.p.X1", "label"] <- "axis 1"
plot(tg.v.rot.all)
```



Community Analyses

To examine the lichen and plant community responses to moth susceptibility we analyzed both univariate and multivariate community metrics. All univariate metrics, which included total abundance (as total % cover), species richness and Shannon's diversity, were analyzed using *t*-tests of the differences between susceptible and resistant trees as done for the tree traits (Pearson 1895). Mulivariate community responses were analyzed with paired PERMANOVAs (Anderson 2001) using Bray-Curtis dissimilarity (Bray & Curtis 1957) adjusted to include zero-sum observations and 10,000 permutations. Mantel correlation was conducted to test for multivariate similarity of lichen and plant community responses to moth susceptibility.

SEM Methods

We used non-metric Multi-Dimensional Scaling (NMDS) to generate ordinations of the community differences between tree pairs. For both communities (lichens and plants) ordinations were conducted using 100 random initial configurations with a maximum of 1000 iterations and a change in stress threshold of less than 10^{-12}. This was repeated for one to four dimension configurations, and the configuration with the lowest dimensionality and an unexplained variation less than 10% was selected. Ordinated scores were then rotated for maximum correlation with the tree trait variables using a procrustes rotation (Oksanen et al. 2019).

Applying an ecological causal modeling approach (Grace and Bollen 2008), we constructed *a priori* models based on our hypotheses of the effects of tree traits on the two communities, lichens and plants (Figure Supplemental *a priori models*). We used the differences beteween moth susceptible and resistant trees for all variables in structural model using linear regressions with only the measured variables. Models were fit to the standardized variables using a maximum likelihood estimator and a X^2 goodness of fit test. We modeled the two communities (lichens and plants) separately because we found no significant correlation between the response of the two communities to moth herbivory (see Results).

Moth Susceptibility Impacted Lichen and Plant Communities

We found significant lichen and plant community differences between moth susceptibile and resistant trees. Abundance, richness and diversity were all lower under susceptible trees for both communities (TABLE abundance richness and diversity t-tests). As a whole both lichen ($R^2 = 0.04$, p-value = 0.031) and plant (*R ˆ2* = 0.31, p-value = 0.0001) communities were signifcantly predicted by moth susceptibility. Although the moth effect was significant for both lichens and plants, their multivariate differences were not correlated (Mantel r = -0.12, p-value = 0.44).

Causal Pathway of Moth Susceptibility Effects on Communities

Moth susceptiblity indirectly influenced lichen and plants community composition by impacting local environmental conditions via altered tree traits. Both SEMs fit the lichen (df = 19, X^2 = 26.6808212, p-value = 0.1123101) and plant (df = 22, X^2 = 30.7624162, p-value = 0.1010805) data well, as neither model showed significant differences from their observed covariance matrices based on their respective X^2 Goodness of Fit tests. Moth crown herbivory effects on litter explained significant amounts of community variation in differences in lichen abundance ($R^2 = 0.282$), richness ($R^2 = 0.601$) and diversity ($R^2 = 0.539$), as well as in plant community differences in abundance ($R^2 = 0.131$), richness ($R^2 = 0.131$) and diversity ($R^2 = 0.196$). Whole community differences between moth susceptible and resistant trees were also signifcantly explained by crown size differences for both lichens ($R^2 = 0.219$) and plants ($R^2 = 0.215$). Together, both of the SEM support a causal pathway from moth crown herbivory increased litter, which altered the abundance of rocks of different size classes, ultimately impacting the lichen and plant communities (Fig. SEM PATH DIAGRAM). However, causal pathways involving light differences between resulting from moth susceptibility were not supported in either the lichen or the plant SEM (SUPPLEMENTARY TABLE SEM path coefficents).

Removing rocks from the plant model we find the following. The model displays good fit (df = 13, X^2 = 15.9661323, p-value = 0.2509635) data well, as neither model showed significant differences from their observed covariance matrices based on their respective X^2 Goodness of Fit tests. The only significant path in this model is from difference in tree crown size to difference in litter (R^2 = NA).

Software and Data

All analyses were done with R version 4.0.4 (R Core Team 2021). Univariate t-tests were conducted using the *stats* package (R Core Team 2021). Multivariate analyses were conducted using the *ecodist* package for distance calculations (Goslee & Urban 2007) and the *vegan* package (Oksanen et al. 2019) for PERMANOVA and Mantel tests and to conduct the ordination and procrustes rotation. The structural equation modeling was conducted using the *lavaan* (Rosseel 2012) and *tidygraph* (Pedersen 2020) packages. Data and software for all the analyses are deposited as a reproducible workflow using the drake package (Landau 2018) at Zenodo (https://zenodo.org/record/4531170).

Citations

Grace, James B. and Bollen, Kenneth A. 2008. Representing general theoretical concepts in structural equation models: the role of composite variables. Environmental and Ecological Statistics, Vol. 15, Issue. 2, p. 191.

Oksanen, Jari F. Guillaume Blanchet, Michael Friendly, Roeland Kindt, Pierre Legendre, Dan McGlinn, Peter R. Minchin, R. B. O'Hara, Gavin L. Simpson, Peter Solymos, M. Henry H. Stevens, Eduard Szoecs and Helene Wagner (2019). vegan: Community Ecology Package. R package version 2.5-6.

Pedersen, Thomas Lin (2020). tidygraph: A Tidy API for Graph Manipulation. R package version 1.2.0.

R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Rosseel, Yves (2012). lavaan: An R Package for Structural Equation Modeling. Journal of Statistical Software, 48(2), 1-36.

Figure Captions

Path diagrams of the structural equation models (SEM) for the (A) lichen and (B) plant communities. Boxes are measured variables and single-headed arrows show the directed hypothesized causal link between variables with the standardized path coeffiecients overlayed onto their respective arrows. Double-headed arrows and undirected dashed lines show the variances and co-variances, respectively. Only variables with at least one significant path coefficient are shown for clarity and estimates for all modeled pathways can be found in Supplementary Table ????.

Bivariate plots of the procrustes rotated ordinations for the (A) lichen and (B) plant communities. Overlaid vectors show the magnitude and direction of the correlations for variables indicated by their respective labels.

Table Legends

Combined results from the univariate *t*-tests of the differences of the community metrics (abundance, richness and diversity) between moth susceptible and resistiant trees (S -R).

```r
tab.ard.combined <- tab.ttest.ard
community <- do.call(rbind,
                     strsplit(rownames(tab.ard.combined), "\\."))[, 1]
community <- gsub("l", "lichens", community)
community <- gsub("p", "plants", community)
metric <- do.call(rbind,
                     strsplit(rownames(tab.ard.combined), "\\."))[, 2]
metric <- gsub("A", "abundance", metric)
metric <- gsub("R", "richness", metric)
metric <- gsub("D", "diversity", metric)
tab.ard.combined <- data.frame(community, metric, tab.ard.combined)
colnames(tab.ard.combined)[colnames(tab.ard.combined) == "statistic.t"] <- "t"
colnames(tab.ard.combined)[colnames(tab.ard.combined) == "parameter.df"] <- "df"
colnames(tab.ard.combined)[colnames(tab.ard.combined) == "estimate.mean.of.x"] <- "mean"
tab.ard.combined[, "t"] <- round(tab.ard.combined[, "t"], 3)
tab.ard.combined[, "df"] <- round(tab.ard.combined[, "df"], 0)
tab.ard.combined[, "mean"] <- round(tab.ard.combined[, "mean"], 3)
tab.ard.combined[, "p.value"] <- round(tab.ard.combined[, "p.value"], 4)
write.csv(file = "results/table_ard_combined.csv", tab.ard.combined)
print(xtable::xtable(tab.ard.combined,
               digits = c(0, 0, 0, 3, 0, 3, 4)),
               include.rownames = FALSE)
```

% latex table generated in R 4.0.4 by xtable 1.8-4 package % Fri Jul 9 10:56:27 2021

Standardized path coefficients, variance and covariance statistics from the Lavann structural equation modeling (SEM) for the lichen community. The labels show the pathway for path coefficients with the directionality indicated by "ON" and covariances are indicated by "WITH". Paths common to both models are only shown once while other varialbes specific to each model are preceded by "l" for lichen and "p" for plant community.

The *est_sig* column contains the standardize path coefficient and possibly asterisks indicating the level of significance. The standard error and p-value for the linear regression for each path are in the following columns *se* and *pval*, respectively.

```
sem.combined <- rbind(
    table_results(fit.l.rot.all)[, c("label", "est_sig", "se", "pval")],
    table_results(fit.v.rot.all)[, c("label", "est_sig", "se", "pval")])
sem.combined <- sem.combined[!(duplicated(sem.combined[, 1])), ]
sem.combined[,1] <- gsub("rot.", "", sem.combined[, 1])
sem.combined[,1] <- gsub(".A.", ".abundance.", sem.combined[, 1])
sem.combined[,1] <- gsub(".R.", ".richness.", sem.combined[, 1])
sem.combined[,1] <- gsub(".D.", ".diversity.", sem.combined[, 1])
sem.combined[,1] <- gsub(".X", ".axis", sem.combined[, 1])
```

```
sem.norock <- rbind(
    table_results(fit.v.rot.norock)[, c("label", "est_sig", "se", "pval")])
sem.norock <- sem.norock[!(duplicated(sem.norock[, 1])), ]
sem.norock[,1] <- gsub("rot.", "", sem.norock[, 1])
sem.norock[,1] <- gsub(".A.", ".abundance.", sem.norock[, 1])
sem.norock[,1] <- gsub(".R.", ".richness.", sem.norock[, 1])
sem.norock[,1] <- gsub(".D.", ".diversity.", sem.norock[, 1])
sem.norock[,1] <- gsub(".X", ".axis", sem.norock[, 1])
```

	label	est_sig	se	pval
1	light.ON.crown	-0.05	0.20	0.80
2	litter.ON.crown	0.59***	0.17	0.00
3	light.ON.trunk	0.03	0.20	0.87
4	litter.ON.trunk	-0.17	0.17	0.31
5	rock.lg.ON.litter	-0.90***	0.08	0.00
6	l.abundance.ON.light	0.04	0.15	0.81
7	l.abundance.ON.rock.lg	0.53***	0.15	0.00
8	l.richness.ON.light	0.17	0.11	0.14
9	l.richness.ON.rock.lg	0.74***	0.11	0.00
10	l.diversity.ON.light	0.20	0.12	0.11
11	l.diversity.ON.rock.lg	0.69***	0.12	0.00
12	l.axis1.ON.light	0.05	0.16	0.75
13	l.axis1.ON.rock.lg	0.46**	0.16	0.00
14	l.axis2.ON.light	0.17	0.18	0.33
15	l.axis2.ON.rock.lg	-0.02	0.18	0.90
16	l.abundance.WITH.l.R	0.23*	0.10	0.03
17	l.abundance.WITH.l.D	-0.03	0.10	0.73
18	l.richness.WITH.l.D	0.30**	0.09	0.00
19	l.abundance.WITH.l.axis1	0.68***	0.18	0.00
20	l.richness.WITH.l.axis1	0.24*	0.11	0.02
21	Variances.light	0.96***	0.25	0.00
22	Variances.litter	0.68***	0.18	0.00
23	Variances.rock.lg	0.18***	0.05	0.00
24	Variances.l.A	0.69***	0.18	0.00
25	Variances.l.R	0.38***	0.10	0.00
26	Variances.l.D	0.43***	0.11	0.00
27	Variances.l.axis1	0.75***	0.19	0.00
28	Variances.l.axis2	0.94***	0.24	0.00
29	l.abundance.WITH.l.axis2	-0.10	0.15	0.51
30	l.richness.WITH.l.axis2	0.10	0.11	0.39

	label	est_sig	se	pval
31	l.diversity.WITH.l.axis1	-0.03	0.10	0.79
32	l.diversity.WITH.l.axis2	0.11	0.12	0.37
33	l.axis1.WITH.l.axis2	0.15	0.16	0.32
34	Variances.crown	0.97	0.00	NA
35	crown.WITH.trunk	0.41	0.00	NA
36	Variances.trunk	0.97	0.00	NA
41	rock.sm.ON.litter	-0.69***	0.13	0.00
42	p.abundance.ON.light	0.02	0.17	0.89
43	p.abundance.ON.rock.sm	-0.36*	0.17	0.03
44	p.richness.ON.light	0.20	0.17	0.24
45	p.richness.ON.rock.sm	-0.37*	0.17	0.03
46	p.diversity.ON.light	-0.01	0.16	0.94
47	p.diversity.ON.rock.sm	-0.44**	0.16	0.01
48	p.axis1.ON.light	0.14	0.16	0.37
49	p.axis1.ON.rock.sm	0.43**	0.16	0.01
50	p.axis2.ON.light	0.06	0.18	0.73
51	p.axis2.ON.rock.sm	-0.01	0.18	0.94
52	p.axis3.ON.light	0.07	0.18	0.70
53	p.axis3.ON.rock.sm	-0.01	0.18	0.94
54	p.abundance.WITH.p.axis2	-0.34	0.18	0.05
55	p.abundance.WITH.p.R	0.36*	0.17	0.03
56	p.abundance.WITH.p.D	0.26	0.16	0.09
57	p.richness.WITH.p.D	0.67***	0.19	0.00
58	p.abundance.WITH.p.axis1	-0.51**	0.17	0.00
59	p.richness.WITH.p.axis1	-0.12	0.14	0.42
62	Variances.rock.sm	0.51***	0.13	0.00
63	Variances.p.A	0.84***	0.22	0.00
64	Variances.p.R	0.82***	0.21	0.00
65	Variances.p.D	0.78***	0.20	0.00
66	Variances.p.axis1	0.74***	0.19	0.00
67	Variances.p.axis2	0.96***	0.25	0.00
68	Variances.p.axis3	0.96***	0.25	0.00
69	p.abundance.WITH.p.axis3	0.30	0.17	0.08
70	p.richness.WITH.p.axis2	-0.13	0.16	0.42
71	p.richness.WITH.p.axis3	0.31	0.17	0.07
72	p.diversity.WITH.p.axis1	-0.07	0.14	0.63
73	p.diversity.WITH.p.axis2	-0.01	0.16	0.96
74	p.diversity.WITH.p.axis3	0.30	0.17	0.07
75	p.axis1.WITH.p.axis2	-0.11	0.16	0.49
76	p.axis1.WITH.p.axis3	0.21	0.16	0.18
77	p.axis2.WITH.p.axis3	-0.18	0.18	0.31

label	est_sig	se	pval
light.ON.crown	-0.05	0.20	0.80
litter.ON.crown	0.59***	0.17	0.00
light.ON.trunk	0.03	0.20	0.87
litter.ON.trunk	-0.17	0.17	0.31
p.abundance.ON.light	-0.03	0.18	0.87
p.abundance.ON.litter	0.12	0.18	0.50
p.richness.ON.light	0.11	0.18	0.55
p.richness.ON.litter	-0.07	0.18	0.68

label	est_sig	se	pval
p.diversity.ON.light	-0.07	0.18	0.68
p.diversity.ON.litter	0.17	0.18	0.33
p.axis1.ON.light	0.18	0.17	0.28
p.axis1.ON.litter	-0.28	0.17	0.10
p.axis2.ON.light	0.08	0.18	0.67
p.axis2.ON.litter	0.10	0.18	0.58
p.axis3.ON.light	0.05	0.18	0.79
p.axis3.ON.litter	-0.11	0.18	0.55
p.abundance.WITH.p.axis2	-0.35	0.19	0.06
p.abundance.WITH.p.R	0.50*	0.20	0.01
p.abundance.WITH.p.D	0.39*	0.19	0.04
p.richness.WITH.p.D	0.84***	0.23	0.00
p.abundance.WITH.p.axis1	-0.62**	0.20	0.00
p.richness.WITH.p.axis1	-0.28	0.17	0.10
Variances.light	0.96***	0.25	0.00
Variances.litter	0.68***	0.18	0.00
Variances.p.A	0.95***	0.25	0.00
Variances.p.R	0.95***	0.24	0.00
Variances.p.D	0.93***	0.24	0.00
Variances.p.axis1	0.84***	0.22	0.00
Variances.p.axis2	0.95***	0.25	0.00
Variances.p.axis3	0.95***	0.25	0.00
p.abundance.WITH.p.axis3	0.32	0.18	0.08
p.richness.WITH.p.axis2	-0.12	0.17	0.48
p.richness.WITH.p.axis3	0.31	0.18	0.09
p.diversity.WITH.p.axis1	-0.20	0.17	0.23
p.diversity.WITH.p.axis2	-0.02	0.17	0.91
p.diversity.WITH.p.axis3	0.32	0.18	0.07
p.axis1.WITH.p.axis2	-0.09	0.16	0.60
p.axis1.WITH.p.axis3	0.18	0.17	0.28
p.axis2.WITH.p.axis3	-0.17	0.18	0.33
Variances.crown	0.97	0.00	NA
crown.WITH.trunk	0.41	0.00	NA
Variances.trunk	0.97	0.00	NA

```r
write.csv(file = "results/sem_results_combined.csv",
          sem.combined,
          row.names = FALSE)

write.csv(file = "results/sem_results_norock.csv",
          sem.norock,
          row.names = FALSE)
```

	label	est_sig	se	pval	confint
1	light.ON.crown	-0.04	0.18	0.84	[-0.39, 0.32]
2	litter.ON.crown	0.52***	0.16	0.00	[0.22, 0.83]
3	rock.sm.ON.litter	-0.69***	0.13	0.00	[-0.95, -0.43]
4	p.A.ON.light	0.02	0.17	0.89	[-0.31, 0.36]
5	p.A.ON.rock.sm	-0.36*	0.17	0.03	[-0.70, -0.03]
6	p.R.ON.light	0.18	0.15	0.22	[-0.11, 0.48]
7	p.R.ON.rock.sm	-0.63***	0.17	0.00	[-0.96, -0.31]
8	p.R.ON.litter	-0.39***	0.10	0.00	[-0.57, -0.20]
9	p.D.ON.light	-0.01	0.16	0.94	[-0.33, 0.31]
10	p.D.ON.rock.sm	-0.44**	0.16	0.01	[-0.76, -0.12]
11	p.X1.ON.light	0.11	0.18	0.56	[-0.25, 0.46]
12	p.X1.ON.rock.sm	0.12	0.18	0.49	[-0.23, 0.48]
13	p.X2.ON.light	-0.02	0.18	0.93	[-0.37, 0.34]
14	p.X2.ON.rock.sm	0.21	0.18	0.25	[-0.14, 0.56]
15	p.X3.ON.light	0.13	0.17	0.44	[-0.20, 0.46]
16	p.X3.ON.rock.sm	0.35*	0.17	0.03	[0.02, 0.68]
17	p.A.WITH.p.X2	-0.50**	0.18	0.01	[-0.86, -0.13]
18	p.A.WITH.p.R	0.32*	0.15	0.03	[0.03, 0.61]
19	p.A.WITH.p.D	0.26	0.16	0.09	[-0.04, 0.57]
20	p.R.WITH.p.D	0.63***	0.17	0.00	[0.29, 0.97]
21	p.A.WITH.p.X1	-0.50**	0.19	0.01	[-0.86, -0.13]
22	p.R.WITH.p.X1	-0.13	0.15	0.36	[-0.42, 0.15]
23	Variances.light	0.97***	0.25	0.00	[0.48, 1.45]
24	Variances.litter	0.70***	0.18	0.00	[0.35, 1.06]
25	Variances.rock.sm	0.51***	0.13	0.00	[0.25, 0.77]
26	Variances.p.A	0.84***	0.22	0.00	[0.42, 1.27]
27	Variances.p.R	0.67***	0.17	0.00	[0.33, 1.00]
28	Variances.p.D	0.78***	0.20	0.00	[0.38, 1.17]
29	Variances.p.X1	0.94***	0.24	0.00	[0.46, 1.41]
30	Variances.p.X2	0.93***	0.24	0.00	[0.46, 1.40]
31	Variances.p.X3	0.81***	0.21	0.00	[0.40, 1.22]
32	p.A.WITH.p.X3	-0.26	0.16	0.11	[-0.57, 0.05]
33	p.R.WITH.p.X2	-0.31*	0.15	0.04	[-0.61, -0.01]
34	p.R.WITH.p.X3	0.02	0.13	0.89	[-0.24, 0.28]
35	p.D.WITH.p.X1	-0.03	0.16	0.85	[-0.33, 0.28]
36	p.D.WITH.p.X2	-0.33*	0.17	0.04	[-0.66, -0.01]
37	p.D.WITH.p.X3	0.05	0.15	0.74	[-0.24, 0.33]
38	p.X1.WITH.p.X2	0.08	0.17	0.64	[-0.25, 0.42]
39	p.X1.WITH.p.X3	-0.10	0.16	0.55	[-0.41, 0.22]
40	p.X2.WITH.p.X3	-0.21	0.16	0.21	[-0.52, 0.11]
41	Variances.crown	0.97	0.00		[0.97, 0.97]

	label	est_sig	se	pval	confint
1	light.ON.crown	-0.05	0.20	0.80	[-0.45, 0.34]
2	litter.ON.crown	0.59***	0.17	0.00	[0.26, 0.93]
3	light.ON.trunk	0.03	0.20	0.87	[-0.36, 0.43]
4	litter.ON.trunk	-0.17	0.17	0.31	[-0.50, 0.16]
5	rock.lg.ON.litter	-0.90***	0.08	0.00	[-1.06, -0.75]
6	l.A.ON.light	0.04	0.15	0.81	[-0.27, 0.34]
7	l.A.ON.rock.lg	0.53***	0.15	0.00	[0.22, 0.83]
8	l.R.ON.light	0.17	0.11	0.14	[-0.05, 0.39]
9	l.R.ON.rock.lg	0.74***	0.11	0.00	[0.52, 0.96]
10	l.D.ON.light	0.20	0.12	0.11	[-0.04, 0.44]
11	l.D.ON.rock.lg	0.69***	0.12	0.00	[0.45, 0.93]
12	rot.l.X1.ON.light	0.05	0.16	0.75	[-0.26, 0.37]
13	rot.l.X1.ON.rock.lg	0.46**	0.16	0.00	[0.15, 0.78]
14	rot.l.X2.ON.light	0.17	0.18	0.33	[-0.18, 0.53]
15	rot.l.X2.ON.rock.lg	-0.02	0.18	0.90	[-0.38, 0.33]
16	l.A.WITH.l.R	0.23*	0.10	0.03	[0.03, 0.43]
17	l.A.WITH.l.D	-0.03	0.10	0.73	[-0.23, 0.16]
18	l.R.WITH.l.D	0.30**	0.09	0.00	[0.12, 0.48]
19	l.A.WITH.rot.l.X1	0.68***	0.18	0.00	[0.32, 1.03]
20	l.R.WITH.rot.l.X1	0.24*	0.11	0.02	[0.03, 0.45]
21	Variances.light	0.96***	0.25	0.00	[0.48, 1.45]
22	Variances.litter	0.68***	0.18	0.00	[0.34, 1.03]
23	Variances.rock.lg	0.18***	0.05	0.00	[0.09, 0.27]
24	Variances.l.A	0.69***	0.18	0.00	[0.34, 1.04]
25	Variances.l.R	0.38***	0.10	0.00	[0.19, 0.57]
26	Variances.l.D	0.43***	0.11	0.00	[0.21, 0.65]
27	Variances.rot.l.X1	0.75***	0.19	0.00	[0.37, 1.13]
28	Variances.rot.l.X2	0.94***	0.24	0.00	[0.46, 1.41]
29	l.A.WITH.rot.l.X2	-0.10	0.15	0.51	[-0.39, 0.19]
30	l.R.WITH.rot.l.X2	0.10	0.11	0.39	[-0.12, 0.31]
31	l.D.WITH.rot.l.X1	-0.03	0.10	0.79	[-0.23, 0.18]
32	l.D.WITH.rot.l.X2	0.11	0.12	0.37	[-0.12, 0.34]
33	rot.l.X1.WITH.rot.l.X2	0.15	0.16	0.32	[-0.15, 0.46]
34	Variances.crown	0.97	0.00		[0.97, 0.97]
35	crown.WITH.trunk	0.41	0.00		[0.41, 0.41]
36	Variances.trunk	0.97	0.00		[0.97, 0.97]

	label	est_sig	se	pval	confint
1	light.ON.crown	-0.05	0.20	0.80	[-0.45, 0.34]
2	litter.ON.crown	0.59***	0.17	0.00	[0.26, 0.93]
3	light.ON.trunk	0.03	0.20	0.87	[-0.36, 0.43]
4	litter.ON.trunk	-0.17	0.17	0.31	[-0.50, 0.16]
5	rock.sm.ON.litter	-0.69***	0.13	0.00	[-0.95, -0.43]
6	p.A.ON.light	0.02	0.17	0.89	[-0.31, 0.36]
7	p.A.ON.rock.sm	-0.36*	0.17	0.03	[-0.70, -0.03]
8	p.R.ON.light	0.20	0.17	0.24	[-0.13, 0.53]
9	p.R.ON.rock.sm	-0.37*	0.17	0.03	[-0.70, -0.04]
10	p.D.ON.light	-0.01	0.16	0.94	[-0.33, 0.31]
11	p.D.ON.rock.sm	-0.44**	0.16	0.01	[-0.76, -0.12]
12	rot.p.X1.ON.light	0.14	0.16	0.37	[-0.17, 0.46]
13	rot.p.X1.ON.rock.sm	0.43**	0.16	0.01	[0.12, 0.75]
14	rot.p.X2.ON.light	0.06	0.18	0.73	[-0.29, 0.42]
15	rot.p.X2.ON.rock.sm	-0.01	0.18	0.94	[-0.37, 0.34]
16	rot.p.X3.ON.light	0.07	0.18	0.70	[-0.29, 0.43]
17	rot.p.X3.ON.rock.sm	-0.01	0.18	0.94	[-0.37, 0.34]
18	p.A.WITH.rot.p.X2	-0.34	0.18	0.05	[-0.69, 0.00]
19	p.A.WITH.p.R	0.36*	0.17	0.03	[0.04, 0.69]
20	p.A.WITH.p.D	0.26	0.16	0.09	[-0.04, 0.57]
21	p.R.WITH.p.D	0.67***	0.19	0.00	[0.30, 1.05]
22	p.A.WITH.rot.p.X1	-0.51**	0.17	0.00	[-0.85, -0.17]
23	p.R.WITH.rot.p.X1	-0.12	0.14	0.42	[-0.40, 0.17]
24	Variances.light	0.96***	0.25	0.00	[0.48, 1.45]
25	Variances.litter	0.68***	0.18	0.00	[0.34, 1.03]
26	Variances.rock.sm	0.51***	0.13	0.00	[0.25, 0.77]
27	Variances.p.A	0.84***	0.22	0.00	[0.42, 1.27]
28	Variances.p.R	0.82***	0.21	0.00	[0.41, 1.24]
29	Variances.p.D	0.78***	0.20	0.00	[0.38, 1.17]
30	Variances.rot.p.X1	0.74***	0.19	0.00	[0.37, 1.12]
31	Variances.rot.p.X2	0.96***	0.25	0.00	[0.48, 1.45]
32	Variances.rot.p.X3	0.96***	0.25	0.00	[0.48, 1.45]
33	p.A.WITH.rot.p.X3	0.30	0.17	0.08	[-0.04, 0.64]
34	p.R.WITH.rot.p.X2	-0.13	0.16	0.42	[-0.46, 0.19]
35	p.R.WITH.rot.p.X3	0.31	0.17	0.07	[-0.03, 0.65]
36	p.D.WITH.rot.p.X1	-0.07	0.14	0.63	[-0.34, 0.20]
37	p.D.WITH.rot.p.X2	-0.01	0.16	0.96	[-0.32, 0.30]
38	p.D.WITH.rot.p.X3	0.30	0.17	0.07	[-0.03, 0.63]
39	rot.p.X1.WITH.rot.p.X2	-0.11	0.16	0.49	[-0.41, 0.20]
40	rot.p.X1.WITH.rot.p.X3	0.21	0.16	0.18	[-0.10, 0.53]
41	rot.p.X2.WITH.rot.p.X3	-0.18	0.18	0.31	[-0.53, 0.17]
42	Variances.crown	0.97	0.00		[0.97, 0.97]
43	crown.WITH.trunk	0.41	0.00		[0.41, 0.41]
44	Variances.trunk	0.97	0.00		[0.97, 0.97]

	χ^2	*df*	*p*-value
Lichens	18.541	13.000	0.138
Plants	12.147	14.000	0.595

	χ^2	*df*	*p*-value
Lichens	26.681	19.000	0.112
Plants	30.762	22.000	0.101

	χ^2	df	p-value
Lichens	26.681	19.000	0.112
Plants	15.966	13.000	0.251

community	metric	t	df	mean	p.value
lichens	abundance	-2.249	29	-1.544	0.0323
lichens	richness	-2.955	29	-2.533	0.0062
lichens	diversity	-2.447	29	-0.437	0.0207
plants	abundance	-7.135	29	-22.433	0.0000
plants	richness	-7.477	29	-1.300	0.0000
plants	diversity	-4.219	29	-0.295	0.0002