Community-weighted mean approach (CWM) and the fourth-corner approach

Example 1: The relationship of community weighted mean with environmental variable on the example of forest understory vegetation (using Vltava dataset)

We will use the Vltava dataset to analyse the relationship between the community-weighted mean of one of the leaf traits (specific leaf area, SLA) and one of the environmental variables (light intensity in the understory). The theory predicts that shade-tolerant species will produce leaves with higher SLA (ie per gram of dry mass they will produce leaves with the larger area) to optimize the gain from photosynthesis in the dim light of shaded conditions. Let's use the real data to see whether this works.

Let's import the object `vltava` from `vltava.RData`; the object is a list of components, and among others contains also the matrix of species composition for herb species (`vltava$herbs$spe`), list of environmental variables (in our case we will use the estimated cover of the tree layer, `vltava$env$COVERE32`, as a proxy of light intensity in the understory - higher cover = less light in the understory), and leaf traits downloaded from TRY database for individual species, specifically SLA (`vltava$herbs$SLA`):

```
load (url
('https://raw.githubusercontent.com/zdealveindy/anadat-
r/master/data/vltava.RData'))
com <- vltava$herbs$spe
cover <- vltava$env$COVERE32
SLA <- sqrt (vltava$herbs$traits$SLA) # we directly sqrt-transform the trait to
improve symmetric distribution
```

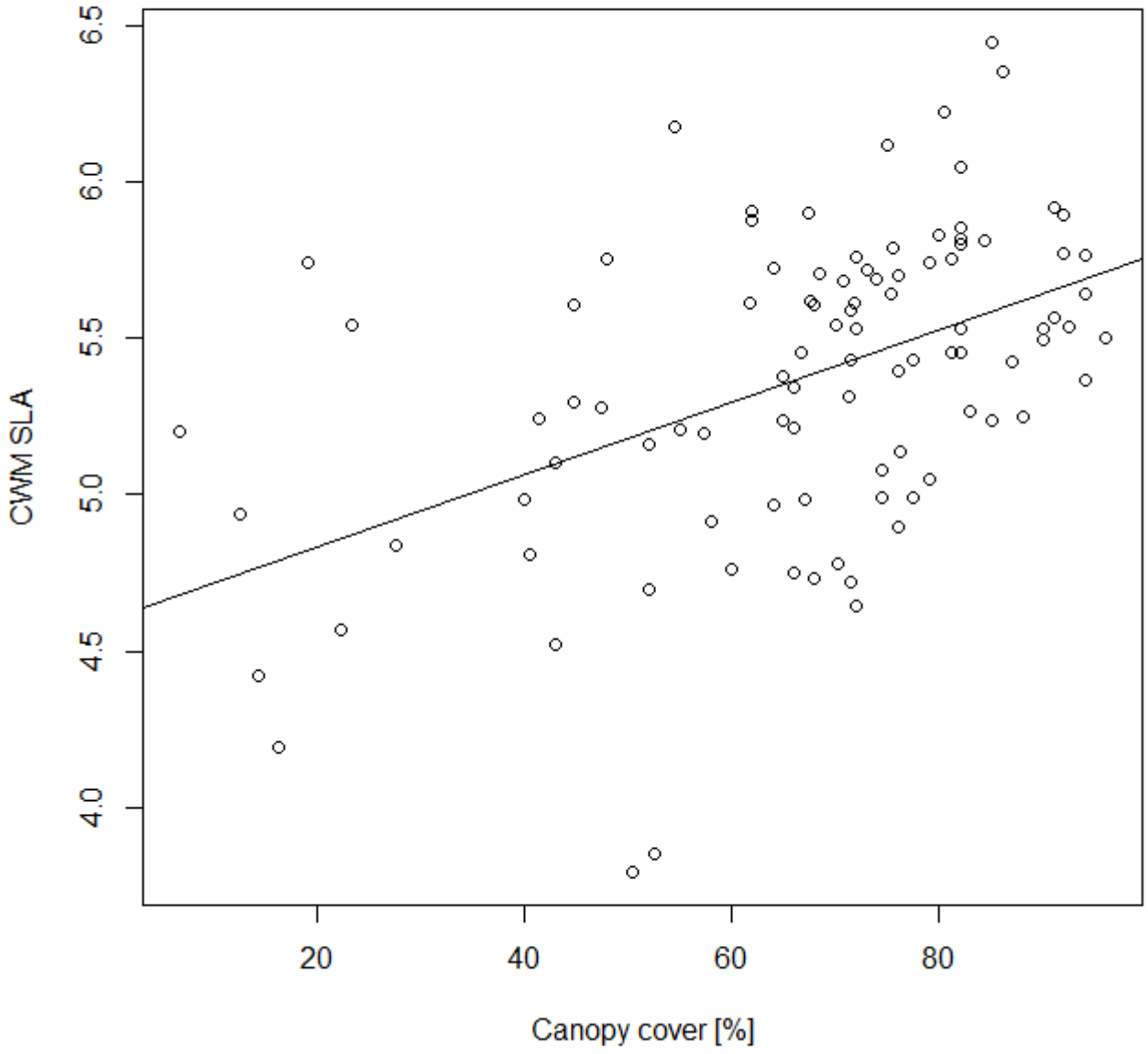
First, let's calculate the community-weighted mean of SLA for individual vegetation plots; since not all species have assigned trait values, we need to first make sure we delete species with missing trait values from both `vltava_spe` and `vltava_traits`:

```
com_1 <- com[,!is.na (SLA)]
SLA_1 <- SLA[!is.na (SLA)]

# calculate CWM for individual sites
CWM_SLA <-  apply (com_1, 1, FUN = function (x) sum (x*SLA_1)/sum (x))

# calculate linear model regression of CWM SLA on the canopy cover:
LM <- lm (CWM_SLA ~ cover)
#... and test the significance using standard parametric F-test:
anova (LM)
```

```
Analysis of Variance Table

Response: SLA
          Df  Sum Sq Mean Sq F value    Pr(>F)
cover      1  5.2702  5.2702  27.471 9.603e-07 ***
Residuals 95 18.2251  0.1918
---
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
# Finally, plot the relationship:
plot (CWM_SLA ~ cover, xlab = 'Canopy cover [%]', ylab = 'CWM SLA')
abline (lm (CWM_SLA ~ cover))
```



So far it goes well - as we see, the relationship is positive and significant (P < 0.001)!

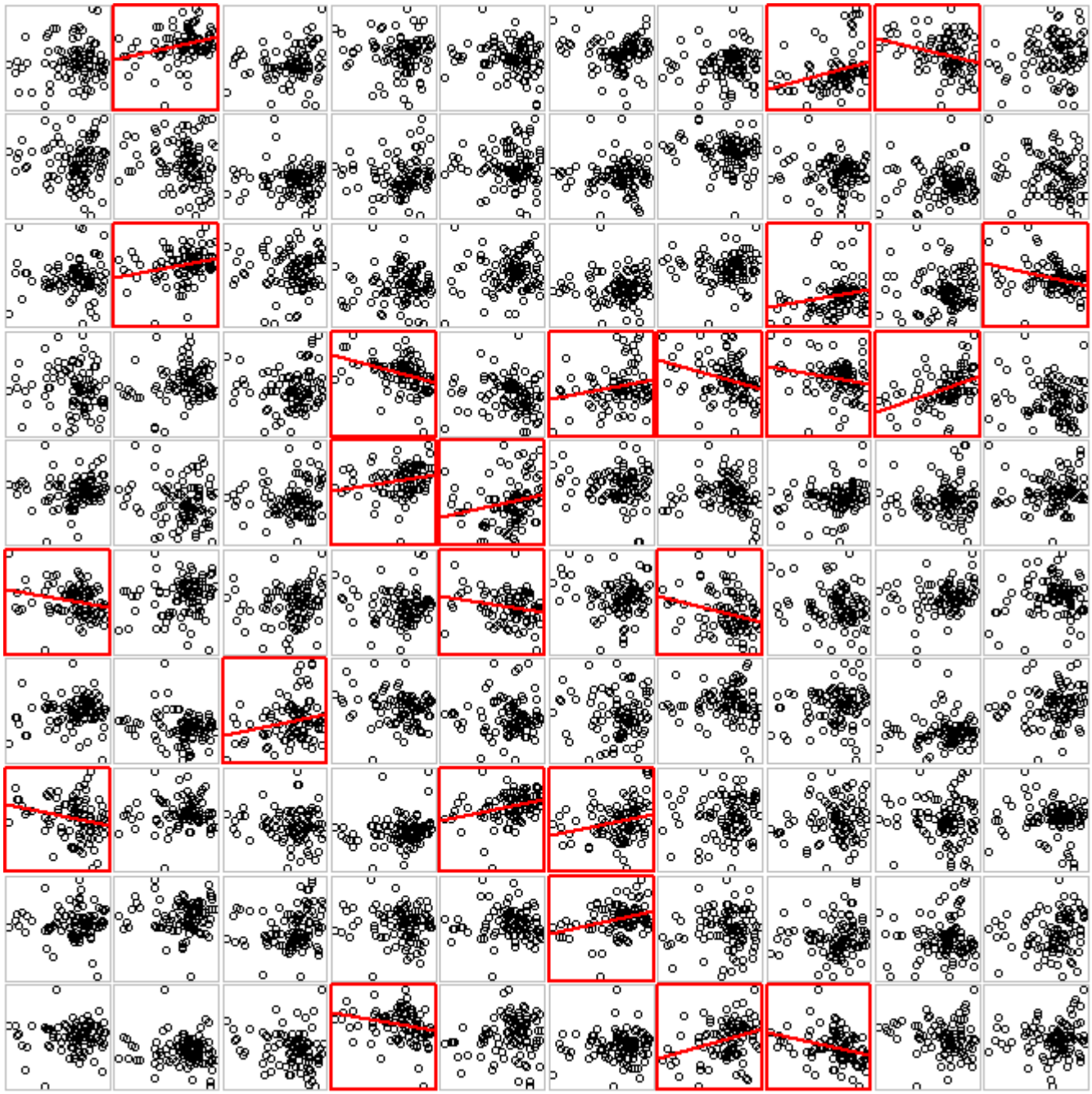
Now, the problem is that even with random traits, we have actually a rather good chance to obtain a significant relationship between CWM and environmental variables. Let's see it on the same data, by permuting the SLA values among species, calculating CWM from these randomized trait values, and testing the regression of this randomized CWM on the original environmental variable:

```
P_rand <- replicate (1000, expr = {
  SLA_rand <- sample (SLA_1)
  CWM_SLA_rand <-  apply (com_1, 1, FUN = function (x) sum (x*SLA_rand)/sum (x))
  LM_rand_col <- lm (CWM_SLA_rand ~ cover)
  P_value <- anova (LM_rand_col)$`Pr(>F)`[1]
```

```
  return (P_value)
})
sum (P_rand < 0.05)
```

Surprisingly, around 280 out of 1000 regressions are significant - but if the test has a correct Type I error rate, it should be around 5% = 50 (out of 1000) significant results. Clearly, the standard parametric test has an inflated Type I error rate. Let's draw 100 of the permuted results and highlight significant relationships:

```
par (mar = c(.1,.1,.1,.1), mfrow = c(10,10))
replicate (100, expr = {
  SLA_rand <- sample (SLA_1)
  CWM_SLA_rand <-  apply (com_1, 1, FUN = function (x) sum (x*SLA_rand)/sum (x))
  LM_rand_col <- lm (CWM_SLA_rand ~ cover)
  P_value <- anova (LM_rand_col)$`Pr(>F)`[1]
  plot (CWM_SLA_rand ~ cover, ann = FALSE, axes = F)
  if (P_value <= 0.05) box (col = 'red', lwd = 2) else box (col = 'grey')
  if (P_value <= 0.05) abline (LM_rand_col, col = 'red', lwd = 2)
})
```



As you can see, a rather high proportion of results are significant, even if the traits used to calculate CWM are in fact randomized values without any ecological meaning.

The solution to this problem is to apply the “max” permutation test, which combines results of row- and column-based permutation, and chooses the higher (less significant) P-value. The “max” test can be calculated by package `weimea` (name derived from the community weighted mean), using the function `test_cwm`.

Note that `weimea` contains also the function `cwm`, which offers quick calculation of CWM values from species composition and trait matrix, and can treat the missing values - no need to remove them first. Still, the number of species in the species composition matrix (columns) and species trait matrix (rows) must be identical, and the order of species must be also the same - `cwm` function does not check for this.

```
library (weimea)
CWM_SLA_w <- cwm (com = com, traits = SLA)
```

The variable `CWM_SLA_w` is like a data frame with calculated CWM, but is much more than that; the object is of the class `cwm`, and contains also the original data used for calculation of CWM (that is why I used a different name of the object from `SLA_CWM` calculated above, even though the calculated values are identical - you may

check if you want). It can thus be used directly for testing the regression of CWM and environmental variable, using the function `test_cwm`; you need to also select the method of relating CWM to environmental variables (currently only `lm` for linear model regression, and `cor` for Pearson's correlation, are available):

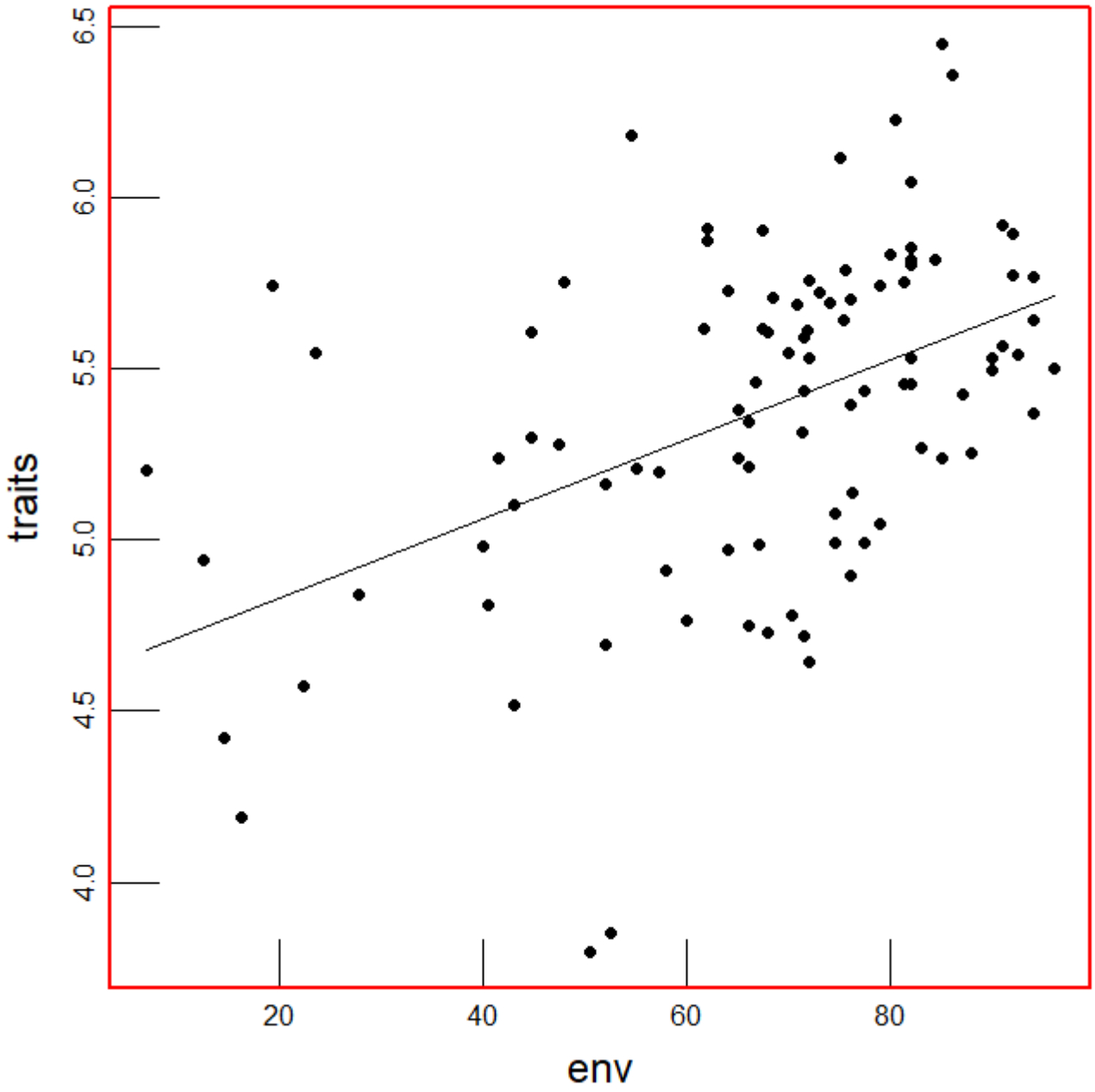
```
CWM_cover <- test_cwm (cwm = CWM_SLA_w, env = cover, method = 'lm', test = 'all')
CWM_cover
```

```
Call:
test_cwm(cwm = CWM_SLA_w, env = cover, method = "lm", test = "all")

                interc interc_se     coef  coef_se     r2  r2adj
1 traits ~ env 4.598542  0.154621 0.011589 0.002211  0.224  0.216
  r2adj_mod n_sit n_spe      F  P_par     P_row     P_col     P_max
1     0.198    97   201 27.471 <0.001 *** 0.002 **  0.002 **  0.002 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The test output is longish. Important is the P_max value in the end, which is significant (P = 0.02). You can also see how it was calculated, because the function returns P-values of both row- (`P_row`) and column-based (`P_col`) permutation test. `P_par` is the result of the parametric F-test, identical to the result of `anova` function above.

The package `weimea` also offers dedicated function to plot the results of the regression:

```
plot (CWM_cover)
```



The conclusion is that the relationship between CWM of SLA for herb species in the forest understory and the light availability is significant even if tested by the “max” test. This offers a strong suggestion that stronger light may filter the species into the community with lower SLA leaves.

Example 2: The relationship of traits and environment calculated by the fourth corner method (using Vltava dataset)

This example directly follows the previous example, in which we related CWM of SLA calculated from herb understory in the forest sites to canopy cover in the same sites as a proxy of the light available for the understory plants. We found a significant positive relationship, i.e. higher cover (less light) = higher SLA.

In this exercise, we will calculate the same relationship, but using the fourth corner method, introduced by Pierre Legendre and colleagues. The method is implemented in the package `ade4`, and also `weimea`.

Let's see the `ade4` implementation first. Remember to remove the missing species from both species composition and trait matrices, since the function `fourthcorner` cannot handle them. Also, `fourthcorner` function expects that all objects representing data will be data frames, so better to convert data into them right

at the beginning.

```
library (ade4)
cover_df <- as.data.frame (cover)
SLA_1_df <- as.data.frame (SLA_1)
fc_ade4 <- fourthcorner (tabR = cover_df, tabL = com_1, tabQ = SLA_1_df, modeltype
= 6)
fc_ade4
```

```
Fourth-corner Statistics
------------------------
Permutation method  Comb. 2 and 4  ( 999  permutations)

Adjustment method for multiple comparisons:   holm
call:  fourthcorner(tabR = cover_df, tabL = com_1, tabQ = SLA_1_df, modeltype = 6,
nrepet = 999)

---

           Test Stat       Obs  Std.Obs     Alter Pvalue Pvalue.adj
1 cover / SLA_1    r 0.1883631 4.206257 two-sided  0.001      0.001 ***

---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The fourth corner statistic r = 0.188, and the P-value of it (calculated by max test, since the argument `modeltype = 6`) is significant.

Now, the same in `weimea` package (we can use the original data, without removing the missing values, the function can handle them):

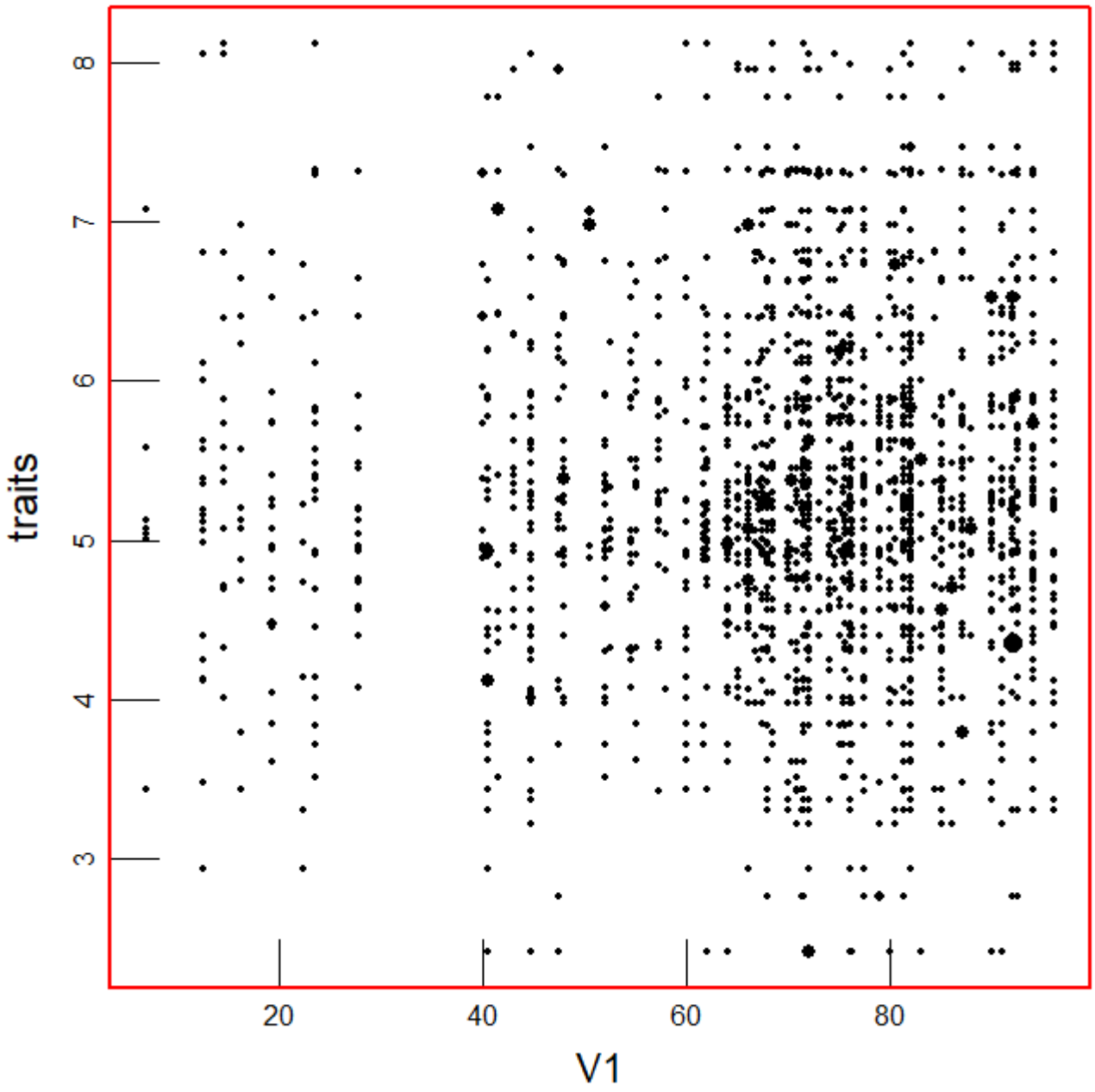
```
fc_weimea <- test_fourth (com = com, env = cover, traits = SLA, perm = 999, test =
'max')
fc_weimea
```

```
Call:
test_fourth(env = cover, com = com, traits = SLA, perm = 999, test = "max")

                r_fc   r_ch n_sit n_spe P_max
1 traits / V1 0.1884 0.2173    97   201 0.001 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

You can see that the fourth corner statistic r_fc = 0.118 is significant at P = 0.001 (minimum P-value we can obtain with 999 permutations).

We can also plot the result into the scatterplot, where each dot represents one species-site combination of trait and environmental variable:

```
plot (fc_weimea)
```



We can calculate the fourth corner for more than a single combination of trait and environmental variable. Let's use all three traits available in `vltava$herbs$traits`, namely SLA, plant height and seed weight, and also choose three environmental variables which may have some relationship to them, namely heat load (combination of slope and aspect, indicating radiation income of the site), soil depth, and cover of canopy layer.

Again, remember to remove missing values! First, let's see how to calculate and visualize this in ade4 package, and then in weimea.

```
traits <- vltava$herbs$traits
traits$seed.weight <- log (traits$seed.weight)
traits_2 <- traits[complete.cases(traits),]
com_2 <- com[, complete.cases(traits)]
env <- vltava$env[,c('HEAT.LOAD', 'SOILDPT', 'COVERE32')]

traits_fc_ade4 <- fourthcorner (tabR = env, tabL = com_2, tabQ = traits_2)
traits_fc_ade4
```

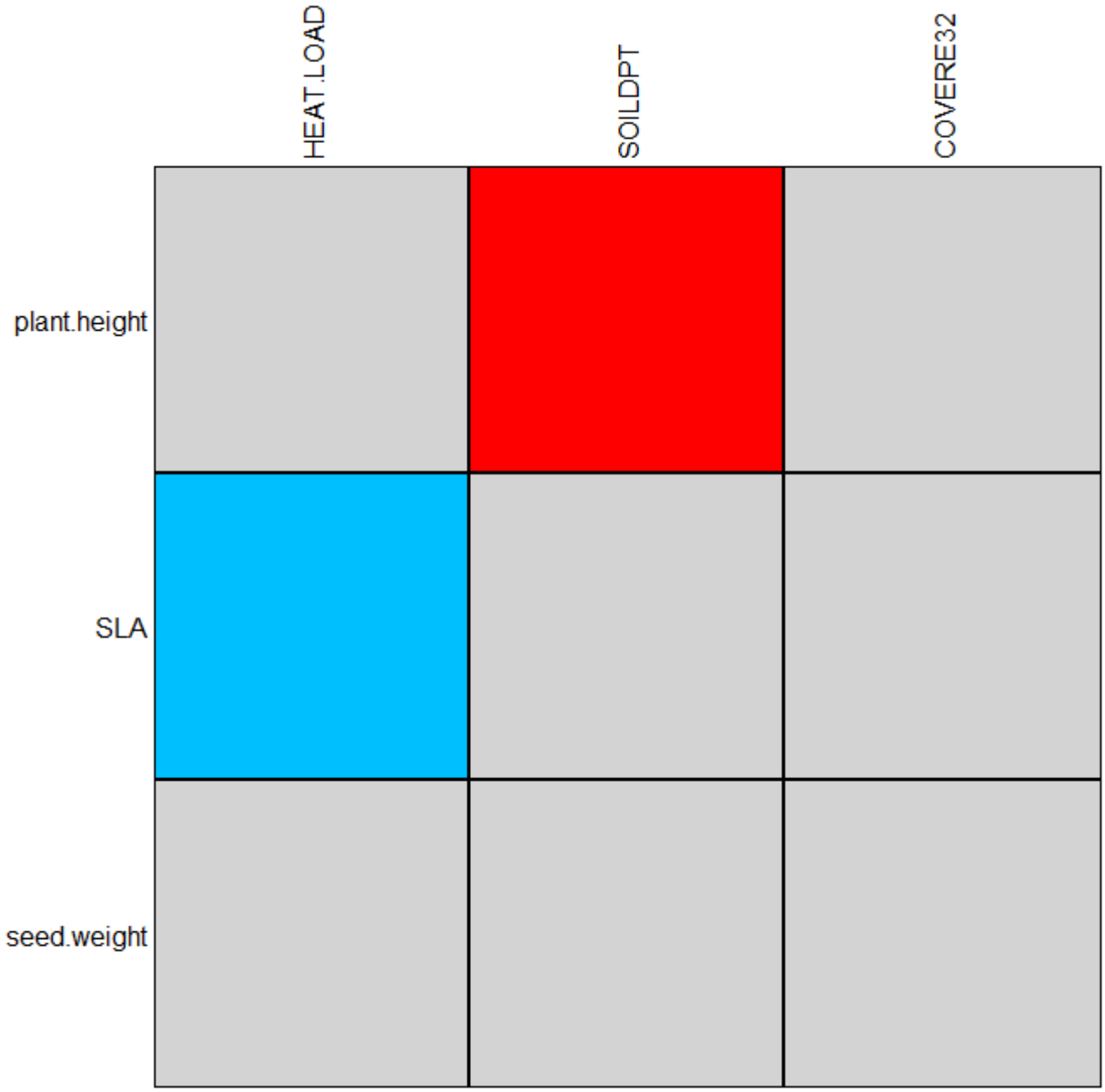
```
Fourth-corner Statistics
```

```
-----------------------
Permutation method  Comb. 2 and 4  ( 999  permutations)

Adjustment method for multiple comparisons:   holm
call:  fourthcorner(tabR = env, tabL = com_2, tabQ = traits_2, nrepet = 999)

---

                      Test Stat         Obs    Std.Obs     Alter Pvalue Pvalue.adj
1 HEAT.LOAD / plant.height    r -0.01429234 -0.2044059 two-sided  0.836      0.836
2   SOILDPT / plant.height    r  0.24784528  3.4754378 two-sided  0.001      0.009
**
3  COVERE32 / plant.height    r  0.16849585  2.3590387 two-sided  0.017      0.102
4          HEAT.LOAD / SLA    r -0.13407584 -2.7269813 two-sided  0.006      0.036
*
5            SOILDPT / SLA    r  0.16344960  2.0470901 two-sided   0.05       0.25
6           COVERE32 / SLA    r  0.17214075  2.4664297 two-sided  0.012      0.084
.
7  HEAT.LOAD / seed.weight    r -0.06877695 -1.3946687 two-sided  0.178      0.616
8    SOILDPT / seed.weight    r  0.07439938  0.9105103 two-sided  0.357      0.714
9   COVERE32 / seed.weight    r  0.10144684  1.4580585 two-sided  0.154      0.616

---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Note that the function automatically applies correction of P-values for multiple testing by Holm's method (the last column, `Pvalue.adj`).

To plot the results, use:

```
plot (traits_fc_ade4)
```



Two of the relationships are significant: plant height and soil depth (red = positive) and SLA and heat load (blue = negative). SLA and cover didn't make it through the Holm's correction.

Now, weimea, using the original data without removing the missing values:

```
traits_fc_weimea <- test_fourth (com = com, env = env, traits = traits, adjustP =
TRUE, perm = 999)
traits_fc_weimea
```

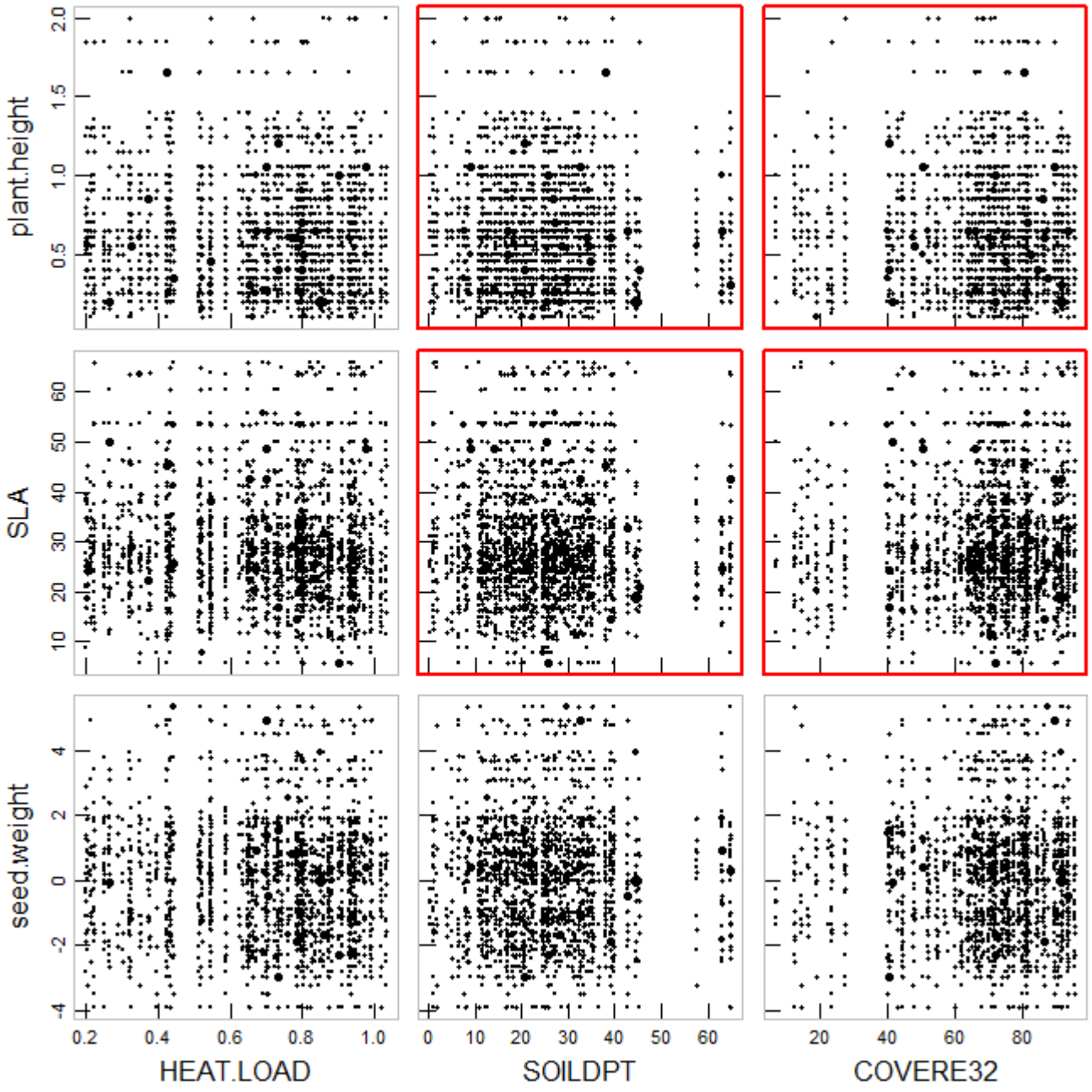
```
Call:
test_fourth(env = env, com = com, traits = traits, perm = 999,
    adjustP = TRUE)

                              r_fc      r_ch n_sit n_spe P_max     P_max_adj
1 plant.height / HEAT.LOAD -0.009418 -0.011042    97   219 0.874         0.874
2          SLA / HEAT.LOAD -0.116195 -0.134066    97   201 0.016 *       0.080 .
3  seed.weight / HEAT.LOAD -0.079453 -0.093118    97   183 0.080 .       0.320
4   plant.height / SOILDPT  0.250548  0.293747    97   219 0.001 ***     0.009 **
```

```
5            SLA / SOILDPT  0.187244  0.216043    97   201 0.006 **       0.036 *
6    seed.weight / SOILDPT  0.059680  0.069945    97   183 0.431          0.862
7  plant.height / COVERE32  0.162151  0.190109    97   219 0.004 **       0.028 *
8            SLA / COVERE32  0.162716  0.187742    97   201 0.003 **       0.024 *
9   seed.weight / COVERE32  0.108765  0.127471    97   183 0.117          0.351
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The values of the fourth corner statistic are different from the ade4 version - this is because here each trait-env correlation is using a different number of species (different values are missing in different traits, and this function removes the missing value only for given trait, not for the whole trait matrix).

Plotting in weimea is more informative:

```
plot (traits_fc_weimea)
```



Example 3: Ecological interpretation of the unconstrained axes in constrained ordination using mean Ellenberg-like indicator values

```
library (vegan)
library (weimea)

# prepare data
data (vltava)
spe <- vltava$spe  # rename variables to make them shorter
env <- vltava$env
eiv <- vltava$eiv

# transform species composition data
spe.log <- log1p (spe)  # species data are in percentage scale which is strongly
rightskewed, better to transform them
spe.hell <- decostand (spe.log, 'hell')  # we are planning to do tb-RDA, this is
Hellinger pre-transformation

# calculate tbRDA with pH and soil depth as explanatory variables
tbRDA <- rda (spe.hell ~ pH + SOILDPT, data = env)
tbRDA
# Call: rda(formula = spe.hell ~ pH + SOILDPT, data = env)
#
# Inertia Proportion Rank
# Total         0.70476    1.00000
# Constrained   0.06250    0.08869    2
# Unconstrained 0.64226    0.91131   94
# Inertia is variance
#
# Eigenvalues for constrained axes:
#   RDA1    RDA2
# 0.04023 0.02227
#
# Eigenvalues for unconstrained axes:
#   PC1     PC2     PC3     PC4     PC5     PC6     PC7     PC8
# 0.07322 0.04857 0.04074 0.03144 0.02604 0.02152 0.01917 0.01715
# (Showing 8 of 94 unconstrained eigenvalues)


ordiplot (tbRDA, choices = c(3,4), type = 'n')
points (tbRDA, choices = c(3,4), display = 'sites', pch = as.character (env$GROUP),
col = env$GROUP)

# calculate mean Ellenberg indicator values, using presence-absence species
composition data
mean_eiv <- cwm (decostand (spe, 'pa'), eiv)

# use standard envfit function to fit these mean_eiv onto ordination diagram (3rd
and 4th (unconstrained) axes)
ef <- envfit (tbRDA, mean_eiv, choices = c(3,4), permutations = 499)
ef
# ***VECTORS
#
#            PC1      PC2     r2 Pr(>r)
# light -0.93106 -0.36487 0.6277  0.002 **
```

```
# temp  -0.97261 -0.23245 0.2356  0.002 **
# cont  -0.86747 -0.49748 0.0891  0.016 *
# moist  0.44535 -0.89535 0.4705  0.002 **
# react  0.95649 -0.29177 0.1160  0.002 **
# nutr   0.94391 -0.33021 0.4522  0.002 **
#   ---
#   Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
# Permutation: free
# Number of permutations: 499
plot (ef, col = 'grey', p.max = 0.05)

# Calculate modified permutation test
ef_modif <- envfit_cwm (tbRDA, cwm = mean_eiv, choices = c(3,4))
ef_modif
# ***VECTORS
#
#            PC1      PC2     r2 Pr(>r)
# light -0.93106 -0.36487 0.6277  0.001 ***
# temp  -0.97261 -0.23245 0.2356  0.178
# cont  -0.86747 -0.49748 0.0891  0.565
# moist  0.44535 -0.89535 0.4705  0.011 *
# react  0.95649 -0.29177 0.1160  0.458
# nutr   0.94391 -0.33021 0.4522  0.004 **
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

plot (ef_modif, col = 'red', p.max = 0.05)
```

From:
https://anadat-r.davidzeleny.net/ - **Analysis of community ecology data in R**

Permanent link:
https://anadat-r.davidzeleny.net/doku.php/en:cwm_fc_examples

Last update: **2021/09/20 21:42**