Simulation of meta-community data subject to environmental and detection filtering

Tobias Roth, Eric Allan, Peter B. Pearman and Valentin Amrhein

2017-06-28

Introduction

We think of a community as a collection of species occurring at a single site, while a meta-community is a collection of communities at multiple sites. In this vignette we adopt the simulation of a meta-community described in chapter 11.2 of Kéry and Royle (2016). Note that we here focus on meta-communities for presence/absence patterns of species, while it is rather straightforward to adopt the concepts to abundance data (see Chapter 11 in Kéry and Royle 2016 for more details).

As a requirement to run the presented code the latest version of the *detectionfilter* package should be downloaded using the following code.

```
library(devtools)
install_github("TobiasRoth/detectionfilter")
library(detectionfilter)
```

The *detectionfilter* package provides the function *simcom()* to simulate community data subject to environmental and detection filtering.

Hierarchical view of a meta-community

A meta-community can be described with a site x species matrix. We call this matrix the community matrix `z`. This community matrix is one of the fundamental biological quantity we are usually focusing in community ecology research. We denote sites with `i` and species with `k`. Thus, `z[i,k]=0` indicates that species `k` is not present at site `i`, while `z[i,k]=1` indicates that the species is present.

One of the key assumption of the Kéry and Royle view of a meta-community is that differences between species can be described with a normal distribution with the species mean and the among-species standard deviation as its parameters. Thus, differences among species in occupancies (i.e. `psi[k]`, the probability a site is occupied by species `k`) can be described with only two parameters. Note, that the normal distribution is assumed on the logit-scale (i.e. `log(psi[k]/(1-psi[k]))`) and not on the occupancy scale.

While this allows for a very parsimonious description of a community, it seems flexible enough to describe real communities. For example, let us have a look at the plants data of the Swiss biodiversity monitoring (`?plantsBDM` for more information). We first obtain the observed community matrix for Swiss plants by assuming a plant species was present at a site if it was observed during at least one of the two surveys. Then, we obtain the occupancies for all species and calculate the mean and standard deviation at the logit-scale.

```
# Calculate observed community matrix
z_obs <- apply(plantsBDM$y, c(1,2), max)

# Calculate observed occupancies for all species
psi <- apply(z_obs, 2, mean)

# Mean and sd at logit-scale
(mean.lpsi <- mean(qlogis(psi)))
```

```
## [1] -2.898826
```

```
(sig.lpsi <- sd(qlogis(psi)))
```

```
## [1] 2.076668
```

We can now compare the distribution of observed plant species' occupancies with the distribution of simulated occupancies using only the mean and standard deviation of species' occupancies. The distribution of observed and simulated occupancies look rather similar with slightly less very rare species in the simulated occupancies.

```
par(mfrow = c(1,2))
hist(psi, breaks = 100, ylim = c(0,500), main = "",
     ylab = "Number of species", xlab = "Occupancy (observed)")
hist(plogis(rnorm(1733, mean.lpsi, sig.lpsi)),
     breaks = 100, ylim = c(0,500), main = "",
     ylab = "Number of species", xlab = "Occupancy (simulated)")
```

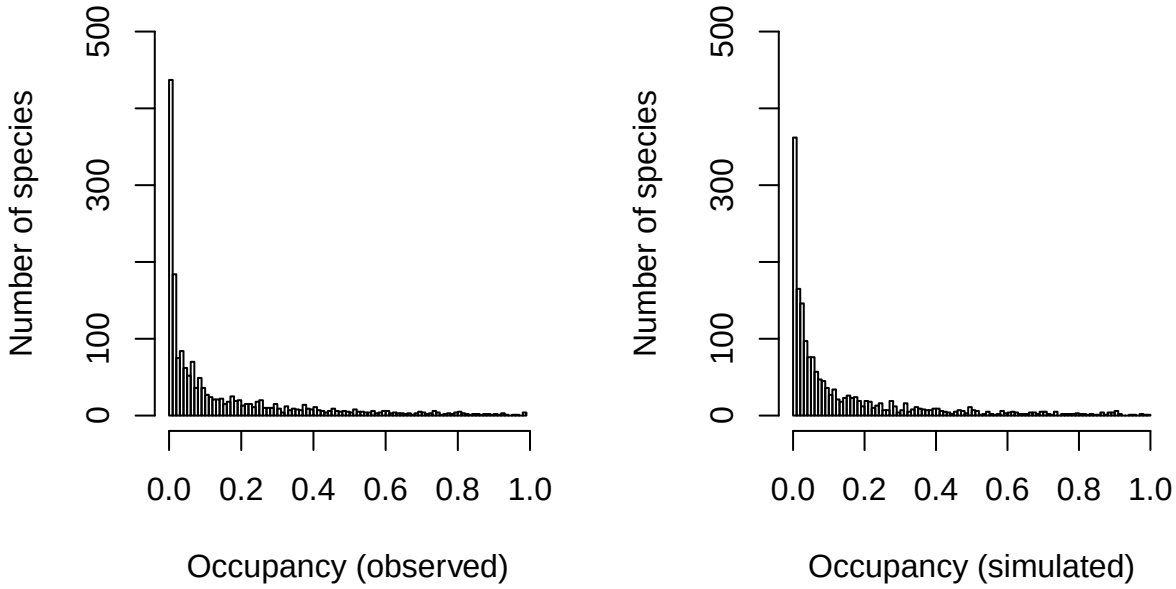

Fig. 1: *Left panel.* Distribution of the occupancies for the 1733 plant species contained in the two surveys of plant communities from Swiss Biodiversity Monitoring. *Right panel.* Distribution of simulated occupancies of the same number of species using only the mean and SD of the observed occupancies as parameter in the simulation.

Thus fare, occupancies of species are constant across sites. However, a more likely scenario would be that the sites are distributed across an environmental gradient and for each species the occupancies would vary along this gradient. A sensible way to describe this might be to use a logistic regression for each species with the occupancy as the dependent variable and the gradient as predictor variable. For simplicity we here use only the linear term (slope) of the gradient, however one could also use linear and quadratic terms to allow for peak occupancy at intermediate stage of the gradient.

Again a very parsimonious description would be to assume that the differences between species in the slope of the gradient are normally distributed. Thus with only four parameters (mean intercept of species, SD of differences in intercept between species, mean slope of gradient effect of species, SD of differences in slope between species) we can describe how the occupancies of species are distributed across an (environmental) gradient.

```
# Choose how many species should be simulated
nspec <- 50

# Choose the values for the four parameters to describe species' occupancies along gradient
```

```r
mean.lpsi <- 1.0         # Mean intercept of species' occupancies
sigma.lpsi <- 1.0        # SD of species differences in intercept
mu.beta.lpsi <- -0.3     # Mean slope of species' occupancies along gradient
sig.beta.lpsi  <- 0.5    # SD of species differences in slopes

# Simulate intercept (b0) and slope (b1) for each species
b0 <- rnorm(nspec, mean.lpsi, sigma.lpsi)
b1 <- rnorm(nspec, mu.beta.lpsi, sig.beta.lpsi)

# Plot species occupancies along gradient
plot(NA, xlim = c(-3, 3), ylim = c(0, 1),
     xlab = "gradient (e.g. elevation)", ylab = "Occupancy")
gradient <- seq(-3, 3, 0.1)
for(k in 1:nspec) {
  lpsi <- b0[k] + b1[k] * gradient
  points(gradient, plogis(lpsi), lwd = 0.5, ty = "l", col = "grey")
}
```

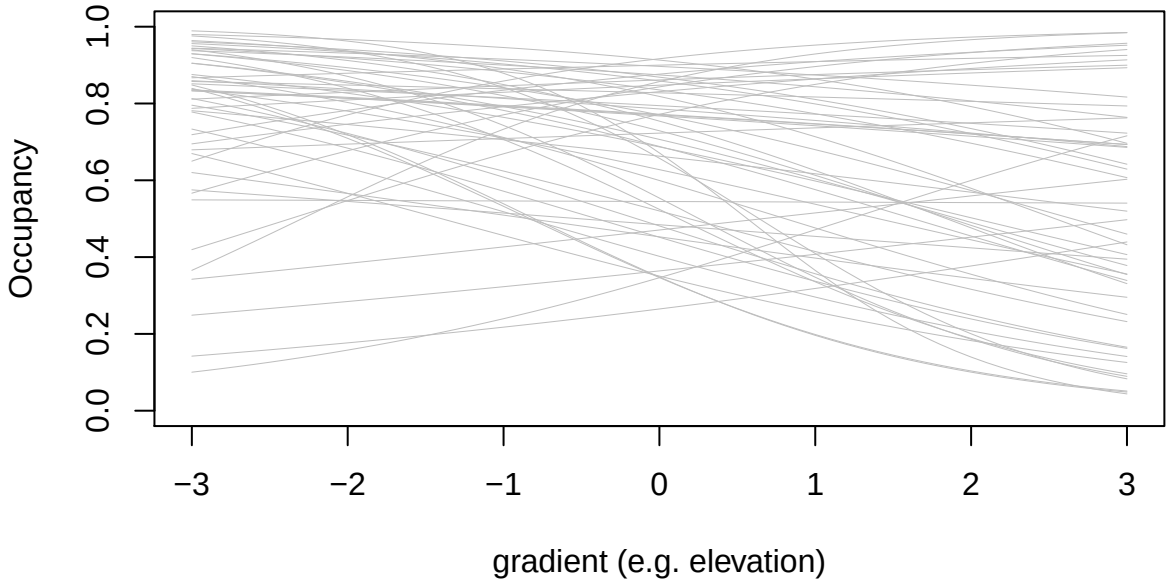

Fig. 2: Occupancy response of 50 simulated species to the gradient.

These curves could be interpreted as a description of the realized niche of each species along the gradient. They describe for each species how likely they are to occur at site at a given location along the elevational gradient. To obtain a realized community we may take for each species the probability that it occurs at that site and sample from a Bernoulli distribution whether the species is indeed present or not. If we repeat that for each species we get the entire community, and if we repeat that for several sites we get a meta-community.

```r
# Number of sites in the meta-community
nsite <- 100

# Distribution of sites along the gradient
# Note that the choice of the distribution is not of particular importance, and
# we may also use a uniform distiruntion
gradient <- rnorm(nsite, 0, 1.5)

# Simulation of the realized meta-community
z <- matrix(NA, nsite, nspec)
```

```
for(k in 1:nspec) {
  lpsi <- b0[k] + b1[k] * gradient
  z[,k] <- rbinom(nspec, 1, plogis(lpsi))
}
```

Environmental filtering

It is now time to add some ideas from functional ecology. A particular ideas would be that the response of species occurrence along a gradient (Fig. 2) does not vary randomly between species but varies according to some specific (functional) traits of the species. If we think of the gradient as some sort of an environmental gradient (e.g. temperature gradient, humidity gradient, ...) and the distribution of species along this gradient can be explained by differences in traits of the species we would speak of environmental filtering.

Given environmental filtering, a prediction would be that the expression of the functional traits of species in the community would change along the gradient. For example we could measure the average expression of a functional trait of occurring species (i.e. the community mean, CM) and infer whether it is changing along the gradient.

```
# Distribution of functional trait expresssion across species
# (we assume that we standardized the trait values and the range
# of trait values covers about 6 standard deviations)
trait <- rnorm(nspec, 0, 1.5)

# Calculate community mean
CM <- apply(z, 1, function(x) mean(trait[x==1]))

# Plot CMs along gradient
plot(gradient, CM, pch = 16, cex = 0.7, ylim = c(-1.5, 1.5))
abline(lm(CM ~ gradient))
abline(h = mean(trait), lty = 2)
```

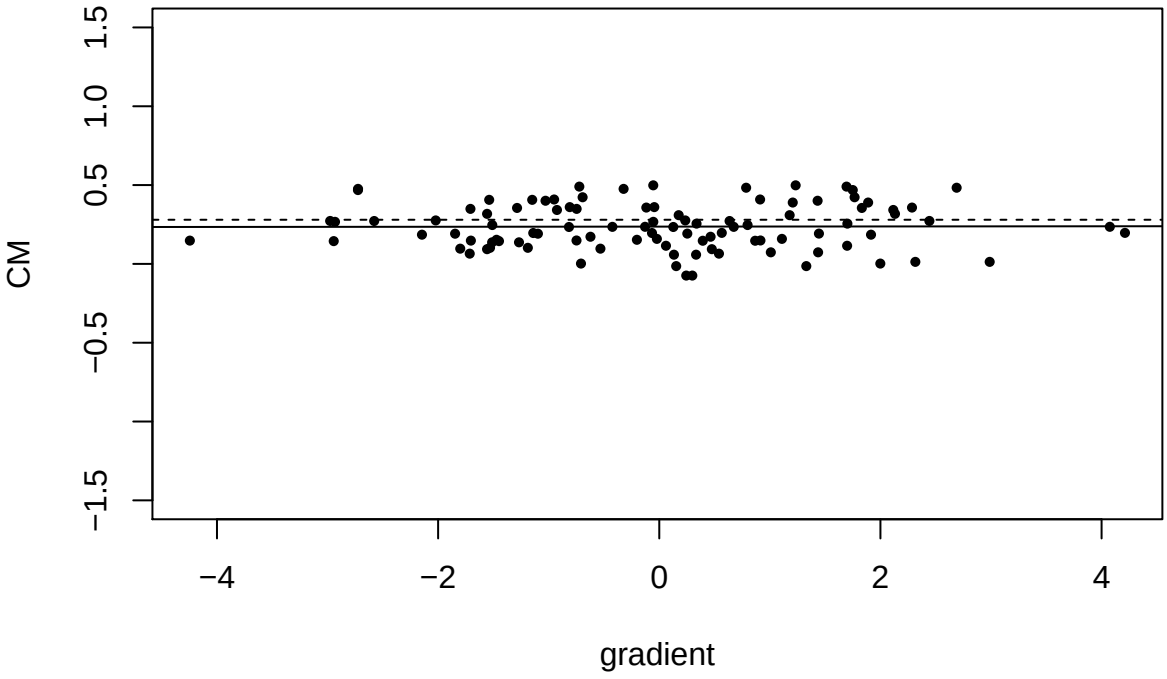

Fig. 3: Mean trait value of occurring species (i.s. CM) at communities along the gradient. The solid line gives the regression line. For reference the dashed line gives the average trait expression for all species.

Thus fare, we assumed that differences in the slope of species response to the gradient changed according to a normal distribution but did not vary according to the functional trait of the species. Thus we did not expect CMs to vary along the elevational gradient. In fact, since species presence in the communities is independent of the trait values, we expected CM to randomly vary along the entire gradient around average trait expression of all species. Figure 3 suggests that the result of the simulation complies with this expectation.

Now, we aim to add to the simulation an effect of the function trait on species' response (slope) to the gradient. For example we could assume that we infer a plant meta-community and the functional trait is a measures of plant height, while the gradient reflects elevation. Since occupancies of larger plants is likely to decrease with elevation we set a negative value for the effect of the functional trait on species response to gradient.

```
# Effect of functional trait on species response to gradient
mu.FTfilter.lpsi <- -1

# Species response (slope) also depends on functional trait
b1 <- rnorm(nspec, mu.beta.lpsi + mu.FTfilter.lpsi * trait, sig.beta.lpsi)

# Simulation of the realized meta-community (same code as above)
z <- matrix(NA, nsite, nspec)
for(k in 1:nspec) {
  lpsi <- b0[k] + b1[k] * gradient
  z[,k] <- rbinom(nspec, 1, plogis(lpsi))
}

# Calculate community mean
CM <- apply(z, 1, function(x) mean(trait[x==1]))

# Plot CMs along gradient
plot(gradient, CM, pch = 16, cex = 0.7, ylim = c(-1.5, 1.5),
     xlab = "Gradient (e.g. elevation)", ylab = "CM (e.g. average plant height)")
abline(lm(CM~gradient))
abline(h = mean(trait), lty = 2)
```

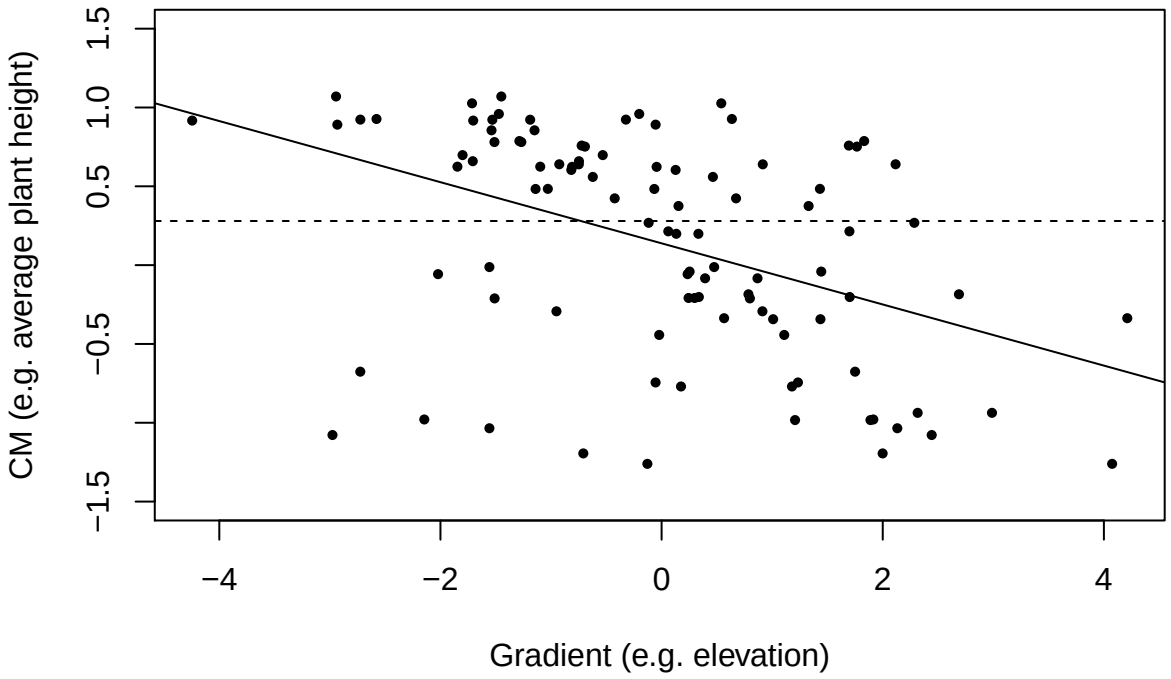

Fig. 4: Mean trait value of occurring species (i.s. CM) at communities along the gradient. The solid line gives the regression line. For reference the dashed line gives the average trait expression for all species.

Now the CM decreases along the gradient and the regression line clearly deviates from the zero-line. In our example this pattern is explained by the filtering of larger species at higher elevations (and the apposite at low elevations.)

Detection filtering

Up to now, we have simulated a meta-community `z` subject to environmental filtering. However, if we do a field surveys to infer the presence of species in multiple communities, we are unlikely to get exactly the community matrix `z`. This is because measurement errors seems likely, especially because some of the species are not seen even tough they were present (i.e. false-negatives). Lets define the observations as `y[i,k]=0` if we did not observe species `k` at site `i` and `y[i,k]=1` if we observed the species. Thus, whenever a species is not present in a community, that is `z[i,k]=0`, we can not observe the species (`y[i,k]=0`). However, if a species is present in a community, that is `z[i,k]=1`, we may or may not observe the species with a certain probability. We may call this probability as the detection probability `p`. Thus, whether we observe a species present in a community we can simulate from a Bernoulli distribution with `p` as its parameter.

Similar as for occupancy, the detection probability may vary between species and between some characteristics of the survey. For example, the date when a biologist surveyed a community may vary between communities. This may well affect the probability to detect species for instance because a plant grows higher during the season and might thus be easier to be detected later on. We call this effect of the date on the detection probability of a species the phenological effect.

Again we can choose a parsimonious description to describe different detectabilities between species by describing the differences between species with normal distribution.

```
# Choose the values for the four parameters to describe species' detection probability
mean.lp <- 0.8         # Mean intercept of species' detection probability
sigma.lp <- 1.0        # SD of species differences in intercept
mu.beta.lp <- 1        # Mean species' phenological effect on detection probability
sig.beta.lp  <- 1      # SD of species differences in phenological effect
```

```r
# Simulate intercept (b0) and slope (b1) for each species
a0 <- rnorm(nspec, mean.lp, sigma.lp)
a1 <- rnorm(nspec, mu.beta.lp, sig.beta.lp)

# Distribution of data of surveys
date <- round(rnorm(nspec, 0, 10), 0)

# Simulation of observed community
y <- matrix(NA, nsite, nspec)
for(k in 1:nspec) {
  lp <- a0[k] + a1[k] * date
  y[,k] <- rbinom(nspec, 1, z[,k]*plogis(lpsi))
}
```

It is now widely accepted that ignoring the effect of imperfect detection may cause bias if species richness is compared between communities. However, the effect of detection probability on function diversity has largely be neglected. Let's thus compare species richness and community mean of the functional trait calculated from the community matrix `z` (i.e. the true community matrix) and calculated from observations `y`.

```r
# Calculate species richness
SR_true <- apply(z, 1, sum)
SR_obs <- apply(y, 1, sum)

# Calculate community mean of trait expression
CM_true <- apply(z, 1, function(x) mean(trait[x==1]))
CM_obs <- apply(y, 1, function(x) mean(trait[x==1]))

# Make plots
par(mfrow = c(1,2))
plot(gradient, SR_obs, pch = 16, cex = 0.7, ylab = "Species richness", ylim = c(0,50))
abline(lm(SR_obs ~ gradient))
points(gradient, SR_true, cex = 0.7, col = "orange")
abline(lm(SR_true ~ gradient), col = "orange")
plot(gradient, CM_obs, pch = 16, cex = 0.7, ylab = "CM", ylim = c(-2,2))
abline(lm(CM_obs ~ gradient))
points(gradient, CM_true, cex = 0.7, col = "orange")
abline(lm(CM_true ~ gradient), col = "orange")
```

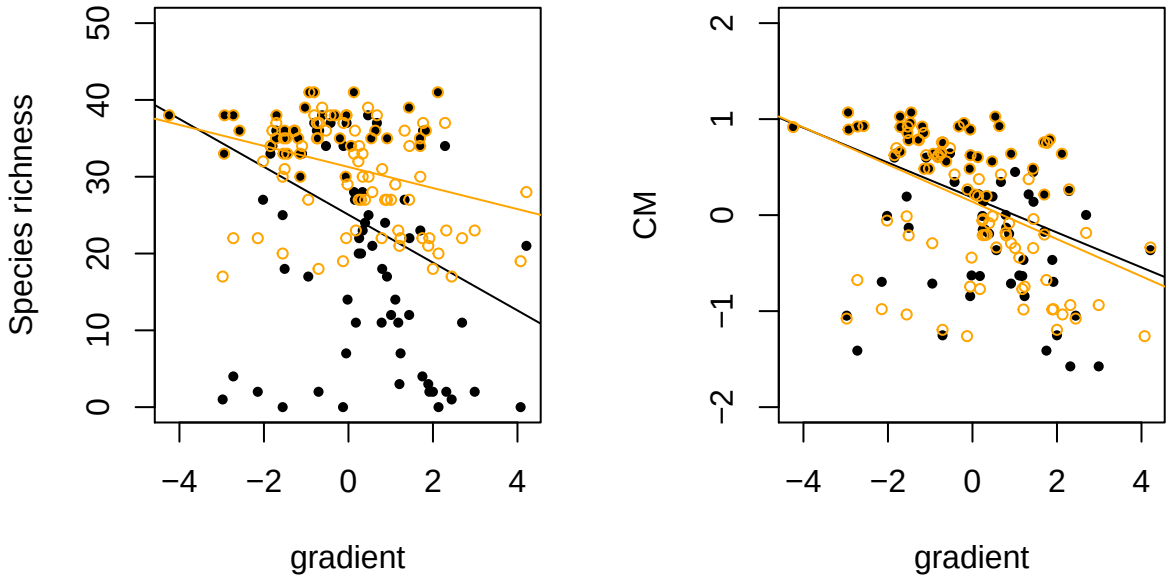

Fig. 5: *Left panel.* Mean species richness of true communities (open orange dots) and observed species richness (black dots) along elevational gradient. *Right panel.* Mean trait expression (CM) of true communities (open orange dots) and CMs of observed communities (black dots) along elevational gradient.

Clearly observed species richness is biased low as compared to the true species richness in communities. Furthermore, the trend of species richness along the gradient seem to differ between observed and true species richness. This would be especially problematic as inferring patterns of community change along a gradient is a nice way to infer the driving patterns that drive the assembly of communities. In contrast, the community means of the functional trait is not affected by imperfect detection. This is not surprising as detection probability of a species in this example was unrelated to the expression of the functional trait in that species.

However, it seems likely that at least some of the functional traits that are often used in functional ecology research predict how likely it is to detect a species. For instance plant height is an important functional trait and to detect larger plants species seems more likely than to detect smaller plant species. Thus, although present in the (true) community some of the species are filtered from the observed communities and this filtering is related to the functional trait. We therefore speak of *detection filtering*. To implement detection filtering in the simulation we add an other parameter that describes how strongly the average detection probability change with the expression of the functional trait.

```
# Effect of functional trait on the average detection probability of a species
mu.FTfilter.lp <- 2

# Average detection probability (intercept) also depends on functional trait
a0 <- rnorm(nspec, mean.lp + mu.FTfilter.lp * trait,  sigma.lp)

# Simulation of observed community (same code as above)
y <- matrix(NA, nsite, nspec)
for(k in 1:nspec) {
  lp <- a0[k] + a1[k] * date
  y[,k] <- rbinom(nspec, 1, z[,k]*plogis(lp))
}

# Calculate community mean of trait expression
CM_obs <- apply(y, 1, function(x) mean(trait[x==1]))

# Make plots
par(mfrow = c(1,2))
```

```
plot(trait, plogis(a0), pch = 16, cex = 0.7, ylim = c(0,1),
     ylab = "Avg.  detection probability", xlab = "Trait (e.g. plant height)")
plot(gradient, CM_obs, pch = 16, cex = 0.7, ylab = "CM", ylim = c(-2,2))
abline(lm(CM_obs ~ gradient))
points(gradient, CM_true, cex = 0.7, col = "orange")
abline(lm(CM_true ~ gradient), col = "orange")
```

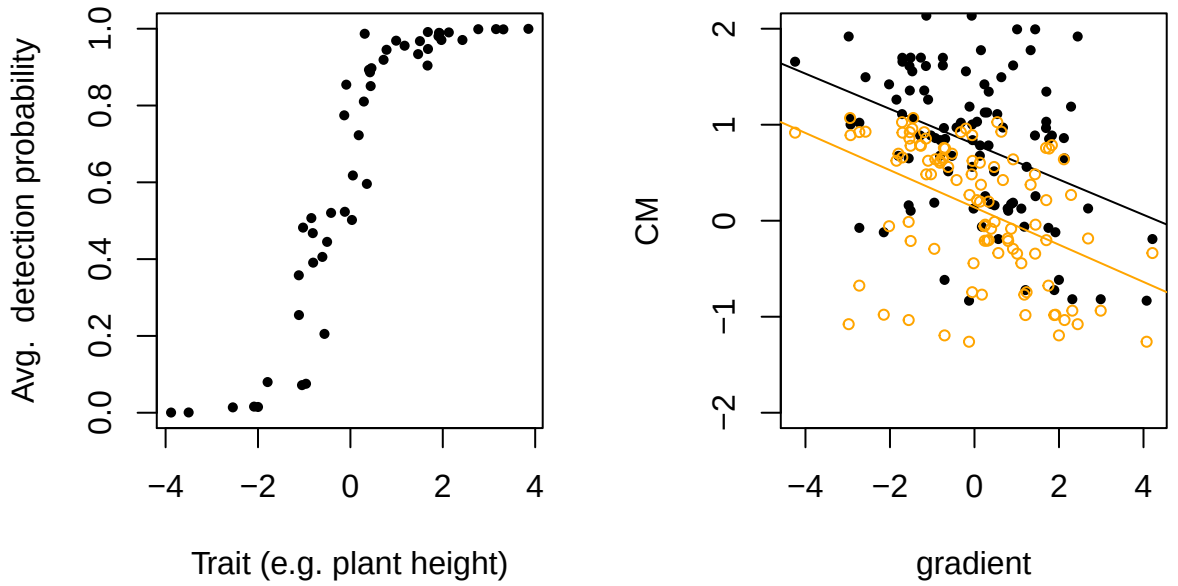

Fig. 6: *Left panel.* The average detection probability of a species is increasing with the expression of the trait value in that species. *Right panel.* Mean trait expression (CM) of true communities (open orange dots) and CMs of observed communities (black dots) along elevational gradient.

The observed communities tend to reveal lower community mean values of the trait expression as compared to the (true) communities. This is because species with low trait value are more likely to be filtered from the observed communities as their detection probability is lower than for species with larger trait value. Note that in the simulation their is no direct effect of the gradient on detection probabilities. Nonetheless, the effect of detection filtering tend to decrease with the gradient. This is because at the lower end of the gradient, the communities consist of fewer species with small trait values and thus with low detection probabilities. If these species are filtered from the observed communities the CMs of these communites will strongly increase. Detection filtering has thus a stronger effect compared to communities at the upper side of the gradient, where the community contains mainly of species with low trait values with less strong effect on CMs if some of these species are filtered from the observed communities. As a result, the effect of detection filtering on measures of functional traits may vary with gradient.

References

Kéry, M.; Royle, J. A., 2016. Applied Hierarchical Modelling in Ecology. Academic Press.