

Appendix S5: Description of statistical analyses (variance components, and analysis and distributions of species characteristics) and associated R code

The R packages 'arm' and 'R2jags' are required for the following analyses.

```
library(R2jags)
library(arm)
```

1 Variance components analysis

The following section describes the method and R code used to perform the variance components analysis.

1.1 Data

Load the exotic cover and associated environmental data.

```
exotics <- read.csv('exotics.csv')
```

Logit transform the total cover of terrestrial exotics in a quadrat.

```
exotics$exo_cover <- qlogis((exotics$exo_cover * .998) + .001)
```

Square root transform and standardise (centre and divide by 1SD) the quadrat depth.

```
exotics$z.dep <- scale(sqrt(exotics$dep))
```

Square root transform and standardise (centre and divide by 1SD) the wetland moisture.

```
exotics$z.moist <- scale(sqrt(exotics$moist))
```

Square root transform and standardise flood frequency.

```
exotics$z.ffreq <- scale(sqrt(exotics$ffreq))
```

Square root transform and standardise wetland hydrological change.

```
exotics$z.change <- scale(sqrt(exotics$change) * -1)
```

Log natural transform and standardise distance of each wetland from nearest town.

```
exotics$z.town <- scale(log(exotics$town))
```

Logit transform and standardise the amount of cattlepugging in a quadrat.

```
exotics$z.pugs <- scale(qlogis((exotics$pugs * .998) + .001))
```

Create workspace variables for all the data needed for the variance components analysis.

```
Y <- exotics$exo_cover
N <- length(Y)
WL <- factor(exotics$wetland)
W <- nlevels(WL)
z.dep <- as.vector(exotics$z.dep)
z.pug <- as.vector(exotics$z.pugs)
z.moist <- as.vector(unique(exotics[, c('wetland', 'z.moist')]))[, 2])
z.change <- as.vector(unique(exotics[, c('wetland', 'z.change')]))[, 2])
z.town <- as.vector(unique(exotics[, c('wetland', 'z.town')]))[, 2])
z.ffreq <- as.vector(unique(exotics[, c('wetland', 'z.ffreq')]))[, 2])
```

Create vector of workspace variable names.

```
D <- c('Y', 'N', 'WL', 'W', 'z.dep', 'z.pug', 'z.change', 'z.town', 'z.ffreq', 'z.moist')
```

1.2 Modelling

Here we present a model of the proportion of a quadrat covered with exotic terrestrial plants, as a function of the depth of water and amount of pugging in the quadrat, and the proportional change in the hydrograph, frequency of flooding and distance from the nearest town of the wetland in which each quadrat is located.

We implemented the model in a Bayesian framework using the open-source software packages R (R Core Team, 2012), and JAGS. The model fit was a hierarchical regression, where the proportional cover of terrestrial exotic plants, P_i , in the i^{th} of n quadrats of the j^{th} of J wetlands is modelled using a logit transformation. We used the equation $\text{logit}((P_i * .998) + .001)$ to transform the quantity P_i into Y_i so we could assume a normal error distribution in our model of Y_i .

The fitted model was of the form,

$$\begin{aligned}
 Y_i &= \alpha_j + \beta_1 \times \text{depth}_i + \beta_2 \times \text{pugging}_i + \epsilon_i \\
 \epsilon_i &\sim N(0, \sigma_0) \\
 \alpha_j &= \gamma_0 + \gamma_1 \times \text{change}_j + \gamma_2 \times \text{town}_j + \gamma_3 \times \text{frequency}_j + \gamma_4 \times \text{moist}_j + \eta_j \\
 \eta &\sim N(0, \sigma_1), \text{ for } i = 1, \dots, n; j = 1, \dots, J.
 \end{aligned}
 \tag{1}$$

The logit of proportion of a quadrat covered by terrestrial exotic plants was modelled as an intercept term, α_j , plus an effect of average quadrat depth, $\beta_1 \times \text{depth}_i$, level of pugging, $\beta_2 \times \text{pugging}_i$, and quadrat level error, ϵ_i . Here depth_i is a vector of square root quadrats depths in meters, pugging_i is a vector of logit (using the same equation as for P_i) proportion of quadrat covered in pugs and β_1 and β_2 are regression coefficients indicating the magnitude and direction of the effects. The error term ϵ_i was modelled with a normal distribution centered on zero and with an estimated standard deviation, σ_0 .

The intercept term was allowed to vary according to the wetland the i^{th} quadrat was located in. The model for α_j had an intercept γ_0 plus an effect of change in hydrology, $\gamma_1 \times \text{change}_j$, effect of distance to nearest town, $\gamma_2 \times \text{town}_j$, effect of flooding frequency, $\gamma_3 \times \text{frequency}_j$, effect of wetland moisture, $\gamma_4 \times \text{moist}_j$ and a wetland level error term, η_j . Here change_j is a vector of square root percentage modification of the upstream hydrograph with natural flow as a baseline, distance_j is a vector of square root of distance to the nearest town in meters, frequency_j is the square root of the number of upstream flood events in the previous ten years, and γ_1 , γ_2 and γ_3 are regression coefficients indicating the magnitude and direction of the effects. The error term η_j was modelled with a normal distribution centered on zero and with an estimated standard deviation, σ_1 .

Vague normal distributions with zero means and standard deviations of one hundred were used for the regression coefficients and intercept term γ_0 . Vague uniform priors with range 0–100 were used for the error term standard deviations σ_0 and σ_1 .

The model fit was run using 3 chains with each chain run for 5000 iterations with the first 1000 iterations of each chain discarded as burnin. Convergence was assessed with visual inspection of trace plots and on the condition that the potential scale reduction factor was < 1.1 and effective number of posterior samples > 100 for each model parameter. All covariate data was centred on zero and scaled by multiplying by one standard deviation.

To determine the relative importance of the model covariates and unexplained quadrat-level and wetland-level variation we undertook a variance components analysis using the method outlined in Hector *et al.* (2011) (see also Gelman, 2005). We compared the magnitudes of the seven variance components; effects of depth, pugging and change in hydrograph, flood frequency, distance to town and unexplained quadrat-level and wetland level variation, on the standard deviation scale. To calculate the components explained by covariates we multiplied the absolute effect-size by the standard deviation of the input variables (in each case this was ≈ 1 due to the scaling of the covariate data). To calculate equivalent components for the unexplained variation we used the standard deviations of the parameters ϵ_j and η_k , the finite population standard deviations, rather than use the parameter estimates for σ_0 and σ_1 , which are the super-population standard deviations.

1.3 Running models

Write JAGS model to a temporary file.

```
cat(c('model', capture.output(body(function() {

  # Loop over 1 to N quadrats.
  for (i in 1:N) {
    # Transformed cover of exotics normally distributed with mean mu[i] and
    # precision, tau.
    Y[i] ~ dnorm(mu[i], tau)
    # Calculate residuals.
    Y.err[i] <- Y[i] - mu[i]
    # Linear model for quadrat expected cover, mu[i]. Intercept varies by
    # wetland, WL, plus effects of depth (b1) and pugging (b2).
    mu[i] <- a[WL[i]] + b1 * z.dep[i] + b2 * z.pug[i]
  }

  # Loop over 1 to W wetlands
  for (j in 1:W) {
    # The varying intercepts are normally distributed with mean mu.a[j] and
    # precision tau.a.
    a[j] ~ dnorm(mu.a[j], tau.a)
    # Calculate wetland level resids
    a.err[j] <- a[j] - mu.a[j]
    # Linear model for wetland expected cover, mu.a[j], with a fixed intercept
    # (g1) plus effects of change in hydrograph (g2), distance to town (g3) and
    # wetland moisture (g4).
    mu.a[j] <- g1 + g2 * z.change[j] + g3 * z.town[j] + g4 * z.ffreq[j] + g5 * z.moist[j]
  }

  # Priors for fixed effects are vague normal distributions.
  b1 ~ dnorm(0, .0001)
  b2 ~ dnorm(0, .0001)
  g1 ~ dnorm(0, .0001)
}
```

```

g2 ~ dnorm(0, .0001)
g3 ~ dnorm(0, .0001)
g4 ~ dnorm(0, .0001)
g5 ~ dnorm(0, .0001)

# Priors on precision parameters are uniform (0-100) on the standard deviation
# scale.
tau <- pow(sd, -2)
tau.a <- pow(sd.a, -2)
sd ~ dunif(0, 100)
sd.a ~ dunif(0, 100)

# Calculate finite population variance components on the standard deviation
# scale for:
# Residual variation.
resid <- sd(Y.err[])
# Unexplained wetland level variation.
sd.site <- sd(a.err[])
# Effect of quadrat depth
sd.dep <- abs(b1) * sd(z.dep[])
# Effect of quadrat pugging
sd.pug <- abs(b2) * sd(z.pug[])
# Effect of wetland percentage change in hydrograph
sd.change <- abs(g2) * sd(z.change[])
# Effect of wetland distance from nearest town
sd.town <- abs(g3) * sd(z.town[])
# Effect of wetland flooding frequency
sd.ffreq <- abs(g4) * sd(z.ffreq[])
# Effect of wetland soil moisture
sd.moist <- abs(g5) * sd(z.moist[])

})), file=(M <- tempfile()))

```

Create a function to initialise chains of the JAGS model.

```

inits <- function() {list(a=rnorm(W), b1=rnorm(1), b2=rnorm(1), g1=rnorm(1),
  g2=rnorm(1), g3=rnorm(1), g4=rnorm(1), g5=rnorm(1), sd=runif(1), sd.a=runif(1))}

```

Run model using MCMC with JAGS and monitor some parameters.

```

model <- jags(D, inits, n.iter=4000, n.burnin=1000, n.thin=3, model.file=M,
  param=c('resid', 'sd.site', 'sd.dep', 'sd.pug', 'sd.change', 'sd.town', 'sd.ffreq',
  'sd.moist', 'g1', 'g2', 'g3', 'g4', 'g5', 'b1', 'b2'))

```

Print the model output.

```
print(model)
```

```
## Inference for Bugs model at "M", fit using jags,
## 3 chains, each with 4000 iterations (first 1000 discarded), n.thin = 3
## n.sims = 3000 iterations saved
##           mu.vect sd.vect   2.5%   25%   50%   75%   97.5% Rhat n.eff
## b1        -1.238  0.161  -1.558  -1.340  -1.237  -1.131  -0.928 1.001 3000
## b2         -0.134  0.177  -0.488  -0.253  -0.129  -0.015   0.203 1.001 2800
## g1        -4.321  0.464  -5.252  -4.616  -4.312  -4.014  -3.398 1.001 3000
## g2         2.050  0.632   0.799   1.632   2.049   2.443   3.302 1.002 1500
## g3        -0.239  0.536  -1.284  -0.585  -0.254   0.106   0.847 1.001 3000
## g4         0.266  0.604  -0.907  -0.127   0.270   0.658   1.478 1.001 2400
## g5        -0.693  0.443  -1.536  -0.980  -0.703  -0.418   0.201 1.001 2100
## resid      3.124  0.020   3.090   3.109   3.122   3.136   3.168 1.001 3000
## sd.change  2.036  0.621   0.798   1.620   2.033   2.424   3.276 1.006 3000
## sd.dep     1.236  0.160   0.927   1.130   1.236   1.339   1.556 1.001 3000
## sd.ffreq   0.518  0.401   0.017   0.202   0.438   0.726   1.498 1.001 3000
## sd.moist   0.756  0.417   0.054   0.455   0.741   1.031   1.613 1.005 1700
## sd.pug     0.177  0.133   0.007   0.071   0.152   0.260   0.488 1.001 3000
## sd.site    2.703  0.137   2.431   2.613   2.703   2.795   2.962 1.001 2700
## sd.town    0.467  0.348   0.018   0.193   0.396   0.672   1.302 1.001 3000
## deviance 2907.852  7.549 2894.853 2902.545 2907.095 2912.554 2924.394 1.001 3000
##
## For each parameter, n.eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor (at convergence, Rhat=1).
##
## DIC info (using the rule, pD = var(deviance)/2)
## pD = 28.5 and DIC = 2936.4
## DIC is an estimate of expected predictive error (lower deviance is better).
```

1.4 Plots

The following function is used to plot the variance components.

```
plot.vc <-  
function (model, comps, ylabs, lab) {  
  vc <- matrix(F, ncol=5, nrow=length(comps))  
  row.names(vc) <- comps  
  for (i in comps) {  
    vc[i, ] <- quantile(  
      eval(parse(text=sprintf('model$BUGSoutput$sims.list%s', i))),  
      prob = c(0.025, 0.1587, .5, 0.8413, 0.975))  
  }  
  plot.new()  
  plot.window(xlim=c(0, max(vc) * 1.1), ylim=c(0, length(comps)))  
  abline(v=0, col='grey75')  
  segments(vc[, 1], seq_len(length(comps)) - .5, vc[, 5], lwd=2)  
  segments(vc[, 2], seq_len(length(comps)) - .5, vc[, 4], lwd=4)  
  points(vc[,3], 1:length(comps) - .5, pch=21, cex=1.2, bg='black')  
  axis(side=1)  
  mtext(side=1, text='Standard deviation', line=2.5, cex=.8)  
  axis(side=2, at=seq_len(length(comps)) - .5, labels=ylabs, las=1, cex.axis=.8)  
  box()  
  text(0.5, 7.5, label=lab)  
}
```

The following function is used to create the partial dependence plots of model covariates.

```
plot.pc <-
function(model, coefs, alpha, labels, xlim, ylim, xlab, xtck, ytck, tpos, lab, bax, sax) {
  labn <- 0
  for (i in coefs) {
    plot.new()
    plot.window(xlim=xlim, ylim=ylim)
    p.mat <- plogis(sapply(seq(xlim[1], xlim[2], length=100),
      function(y) {
        eval(parse(text=sprintf('model$BUGSoutput$sims.list$%s', alpha))) +
        eval(parse(text=sprintf('model$BUGSoutput$sims.list$%s', i))) * y
      }
    )) * 100
    polygon(
      c(seq(xlim[1], xlim[2], length=100), seq(xlim[2], xlim[1], length=100)),
      c(apply(p.mat, 2, function(y) quantile(y, .025)),
        rev(apply(p.mat, 2, function(y) quantile(y, .975))))), col='grey', border=NA)
    points(seq(xlim[1], xlim[2], length=100), colMeans(p.mat), type='l', lwd=3, lend=1)
    if (which(coefs == i) %in% sax) axis(side=2, las=1, at=ytck, labels=ytck / 100)
    if (which(coefs == i) %in% bax) axis(side=1, at=xtck)
    text(0, tpos, labels=labels[which(coefs == i)])
    box()
    labn <- labn + 1
    text(-1.5, 85, label=lab[labn])
  }
  mtext(side=1, text='Standard deviations from mean', line=3, outer=T)
  mtext(side=2, text=xlab, line=3, outer=T)
}
```

Plot the variance components on the standard deviation scale and partial dependences for wetland percentage change in hydrograph, quadrat amount of pugging, distance of wetland to nearest town, flood frequency and wetland quadrat depth.

```
par(mar=c(5, 6, 1, 1))

plot.vc(model=model, lab='',
  comps=c('resid', 'sd.site', 'sd.town', 'sd.ffreq', 'sd.pug', 'sd.dep', 'sd.change',
    'sd.moist'),
  ylab=c('Residuals', 'Site', 'Proximity\nto town', 'Flood\nfrequency', 'Cattle pugs',
    'Water depth', 'Hydrological\nchange', 'Soil moisture')
)
```

```
par(mfrow=c(3, 2), oma=c(5, 6, 4, 5), mar=c(0, 0, 0, 0), lheight=.6)
plot.pc(model, c('g2', 'b1', 'b2', 'g4', 'g3', 'g5'), alpha='g1', xlim=c(-2.1, 2.1),
  ylim=c(0, 90), labels=c('Hydrological\nchange', 'Water depth', 'Cattle pugs',
    'Flood frequency', 'Proximity\nto town', 'Soil moisture'),
  xlab='Proportion of terrestrial cover that is exotic', ytck=0:4 * 20, xtck=-2:2,
  tpos=70, lab=c('a', 'b', 'c', 'd', 'e', 'f'), bax=c(5, 6), sax=c(1, 3, 5)
)
```

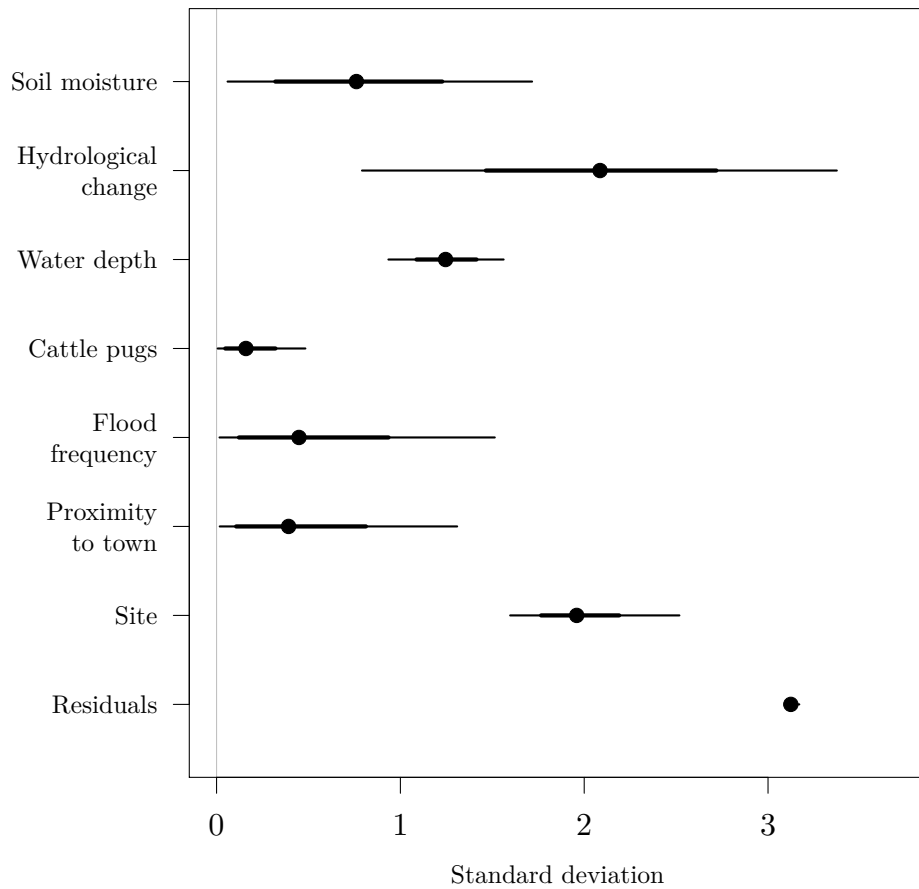


Figure 1: Panel (a) Variance components for a Bayesian model of logit exotic cover. Components are plotted on the standard deviation scale. Black dots show the medians of the posterior densities with thick lines showing the one posterior standard deviations either side, and the thin lines indicating two posterior standard deviations or 95% credible intervals.

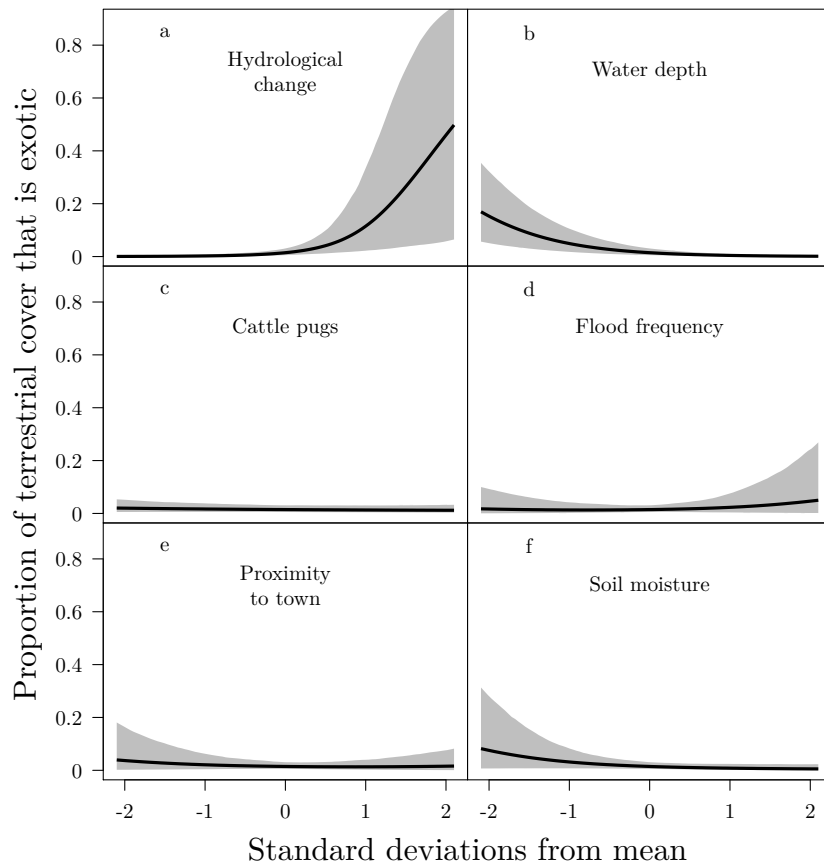


Figure 2: Panels (a-f) Partial dependence plots of percent cover of exotic plants.

2 Species characteristics analysis

The following section outlines the method and R code used to fit and plot the output of the multi-species hierarchical models for exotic and native terrestrial species presence absence data using data about species characteristics. Where "traits" are mentioned in this appendix we are referring to species characteristics. A detailed description of the type of model used can be found in Pollock, Morris & Vesk, 2012.

2.1 Data

Load the species trait data.

```
traits <- read.csv('traits.csv')
```

Natural log transform and standardise maximum plant height.

```
traits$z.height <- scale(log(traits$height))
```

Natural log transform and standardise SLA.

```
traits$z.sla <- scale(log(traits$sla))
```

Natural log transform and standardise seed-mass.

```
traits$z.seed <- scale(log(traits$seed))
```

Load species presence/absence data.

```
species <- read.csv('species.csv')
```

Merge the environmental data from exotic cover dataset with the presence/absence dataset.

```
species <- merge(species,  
  exotics[c('quadrat', 'wetland', 'z.dep', 'z.change', 'z.town')])
```

Merge species presence/absence and trait data.

```
species <- merge(species,  
  traits[c('species', 'native', 'weed', 'cult', 'perennial', 'tda',  
    'z.seed', 'z.sla', 'z.height')])
```

2.2 Modelling

Fit the multi-species hierarchical trait model for exotics terrestrials and display results summary.

```
traits.mod.exo <-glmer(  
  present ~  
  z.dep + z.change +  
  z.dep:perennial + z.dep:weed + z.dep:cult + z.dep:tda +  
  z.dep:z.seed + z.dep:z.sla + z.dep:z.height +  
  z.change:perennial + z.change:weed + z.change:cult + z.change:tda +  
  z.change:z.sla + z.change:z.height + z.change:z.seed +  
  (1 + z.dep + z.change | species) +  
  (1 | wetland),  
  family=binomial(link=logit),  
  data=subset(species, !native)  
)
```

```
summary(traits.mod.exo)
```

```
## Generalized linear mixed model fit by maximum likelihood ['glmerMod']
## Family: binomial ( logit )
## Formula: present ~ z.dep + z.change + z.dep:perennial + z.dep:weed + z.dep:cult + z.dep:tda + z
## Data: subset(species, !native)
##
##      AIC      BIC  logLik deviance
##    2320    2513   -1136    2272
##
## Random effects:
## Groups Name      Variance Std.Dev. Corr
## species (Intercept) 4.222    2.055
##      z.dep      2.930    1.712    0.91
##      z.change    0.197    0.444    0.60 0.72
## wetland (Intercept) 1.011    1.006
## Number of obs: 22720, groups: species, 40; wetland, 24
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -7.8537    0.4987  -15.75 < 2e-16 ***
## z.dep         -2.4679    0.6768   -3.65 0.00027 ***
## z.change       0.3943    0.4493    0.88 0.38026
## z.dep:perennial 0.5145    0.3662    1.41 0.15997
## z.dep:weed    -0.0207    0.5250   -0.04 0.96847
## z.dep:cult    -0.9117    0.4340   -2.10 0.03565 *
## z.dep:tda     -0.3972    0.3856   -1.03 0.30297
## z.dep:z.seed   0.5431    0.2492    2.18 0.02931 *
## z.dep:z.sla    0.1949    0.2663    0.73 0.46431
## z.dep:z.height -0.1594    0.1950   -0.82 0.41350
## z.change:perennial 0.3001    0.2435    1.23 0.21766
## z.change:weed  0.6460    0.3903    1.66 0.09787 .
## z.change:cult -0.6727    0.2930   -2.30 0.02170 *
## z.change:tda  -0.2468    0.2667   -0.93 0.35463
## z.change:z.sla 0.0751    0.1791    0.42 0.67506
## z.change:z.height -0.0936    0.1291   -0.73 0.46842
## z.change:z.seed 0.3082    0.1810    1.70 0.08868 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) z.dep  z.chng z.dp:p z.dp:w z.dp:c z.dp:t z.dp:z.sd z.dp:z.sl z.dp:z.h
## z.dep          0.521
## z.change       0.027 0.235
## z.dep:prnlnl -0.087 -0.517 -0.141
## z.dep:weed    -0.052 -0.632 -0.187 0.394
## z.dep:cult    0.063 -0.100 -0.077 -0.045 -0.282
## z.dep:tda    -0.003 -0.024 -0.008 -0.091 -0.155 -0.380
## z.dep:z.sed  -0.013 -0.026 0.017 0.051 0.195 -0.189 -0.297
## z.dep:z.sla   0.007 -0.333 -0.111 0.379 0.123 0.205 -0.035 -0.290
## z.dp:z.hght  -0.006 0.028 -0.011 0.037 0.093 -0.177 -0.029 -0.114 0.094
## z.chng:prnn  0.024 -0.120 -0.504 0.338 0.132 -0.003 -0.054 0.012 0.131 0.066
## z.change:wd  -0.001 -0.163 -0.687 0.113 0.287 -0.085 -0.035 0.055 0.055 0.034
## z.chang:clt  0.035 -0.058 -0.085 -0.006 -0.099 0.365 -0.115 -0.091 0.075 -0.064
```

```

## z.change:td 0.037 0.017 -0.001 -0.052 -0.045 -0.107 0.304 -0.076 -0.062 -0.016
## z.chng:z.sl -0.001 -0.108 -0.349 0.128 0.059 0.075 -0.065 -0.086 0.325 0.066
## z.chng:z.hg 0.003 -0.012 0.042 0.062 0.036 -0.063 -0.015 -0.019 0.062 0.324
## z.chng:z.sd -0.061 -0.022 -0.056 0.015 0.064 -0.086 -0.074 0.252 -0.078 -0.016
##          z.chng:p z.chng:w z.chng:c z.chng:t z.chng:z.sl z.chng:z.h
## z.dep
## z.change
## z.dep:prnrl
## z.dep:weed
## z.dep:cult
## z.dep:tda
## z.dep:z.sed
## z.dep:z.sla
## z.dp:z.hght
## z.chng:prnn
## z.change:wd 0.388
## z.chang:clt -0.072 -0.292
## z.change:td -0.067 -0.174 -0.408
## z.chng:z.sl 0.398 0.074 0.215 0.058
## z.chng:z.hg -0.027 0.078 -0.195 0.060 0.014
## z.chng:z.sd 0.039 0.190 -0.104 -0.353 -0.284 -0.152

```

Fit the multi-species hierarchical trait model for exotics terrestrials and display results summary.

```
traits.mod.native <- glmer(  
  present ~  
  z.dep + z.change +  
  z.dep:perennial + z.dep:weed + z.dep:cult + z.dep:tda +  
  z.dep:z.seed + z.dep:z.sla + z.dep:z.height +  
  z.change:perennial + z.change:weed + z.change:cult + z.change:tda +  
  z.change:z.sla + z.change:z.height + z.change:z.seed +  
  (1 + z.dep + z.change | species) +  
  (1 | wetland),  
  family=binomial(link=logit),  
  data=subset(species, native)  
)
```

```
summary(traits.mod.native)
```

```
## Generalized linear mixed model fit by maximum likelihood ['glmerMod']
## Family: binomial ( logit )
## Formula: present ~ z.dep + z.change + z.dep:perennial + z.dep:weed + z.dep:cult + z.dep:tda + z
## Data: subset(species, native)
##
##      AIC      BIC  logLik deviance
##    2769    2960   -1361    2721
##
## Random effects:
## Groups Name          Variance Std.Dev. Corr
## species (Intercept)  7.35     2.712
##      z.dep           2.04     1.427  0.91
##      z.change        1.13     1.061  0.64 0.91
## wetland (Intercept)  0.71     0.842
## Number of obs: 20448, groups: species, 36; wetland, 24
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -7.83660   0.58886  -13.31 < 2e-16 ***
## z.dep         -3.66159   0.49265   -7.43 1.1e-13 ***
## z.change      -1.42126   0.56401   -2.52  0.012 *
## z.dep:perennial  0.64282   0.45697    1.41  0.160
## z.dep:weed     0.02495   0.50532    0.05  0.961
## z.dep:cult    -0.02566   0.33200   -0.08  0.938
## z.dep:tda     0.33052   0.41438    0.80  0.425
## z.dep:z.seed   0.30979   0.15075    2.06  0.040 *
## z.dep:z.sla   -0.38517   0.18520   -2.08  0.038 *
## z.dep:z.height -0.11746   0.16666   -0.70  0.481
## z.change:perennial 0.58557   0.57075    1.03  0.305
## z.change:weed   0.80668   0.59655    1.35  0.176
## z.change:cult   0.24656   0.42558    0.58  0.562
## z.change:tda   -1.05444   0.53312   -1.98  0.048 *
## z.change:z.sla  0.36295   0.23749    1.53  0.126
## z.change:z.height 0.00535   0.21546    0.02  0.980
## z.change:z.seed -0.15826   0.19683   -0.80  0.421
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) z.dep  z.chng z.dp:p z.dp:w z.dp:c z.dp:t z.dp:z.sd z.dp:z.sl z.dp:z.h
## z.dep          0.592
## z.change       0.316  0.660
## z.dep:prnlnl  0.013 -0.518 -0.332
## z.dep:weed    0.051 -0.264 -0.174  0.178
## z.dep:cult   -0.039 -0.148 -0.122 -0.268 -0.253
## z.dep:tda    0.066 -0.061 -0.181 -0.313  0.072  0.117
## z.dep:z.sed  -0.036  0.151  0.130 -0.027 -0.199 -0.021 -0.123
## z.dep:z.sla  0.070  0.003  0.018  0.261 -0.183  0.147 -0.164 -0.508
## z.dp:z.hght -0.024  0.178  0.087 -0.215 -0.403  0.065  0.141 -0.401  0.539
## z.chng:prnn -0.052 -0.347 -0.625  0.341  0.073  0.117  0.057 -0.007  0.062 -0.108
## z.change:wd  -0.087 -0.255 -0.407  0.068  0.460 -0.196  0.212 -0.077 -0.208 -0.090
## z.chang:clt -0.008 -0.109 -0.149  0.133 -0.197  0.370 -0.077  0.022  0.183  0.070
```

```

## z.change:td 0.062 -0.109 -0.057 0.044 0.198 -0.079 0.002 -0.209 0.055 0.026
## z.chng:z.sl -0.039 -0.052 -0.084 0.084 -0.182 0.170 0.057 -0.185 0.537 0.378
## z.chng:z.hg 0.032 0.109 0.222 -0.094 -0.062 0.057 0.027 -0.298 0.378 0.602
## z.chng:z.sd 0.063 0.180 0.252 -0.010 -0.047 0.020 -0.232 0.543 -0.173 -0.308
## z.chng:p z.chng:w z.chng:c z.chng:t z.chng:z.sl z.chng:z.h
## z.dep
## z.change
## z.dep:prnrl
## z.dep:weed
## z.dep:cult
## z.dep:tda
## z.dep:z.sed
## z.dep:z.sla
## z.dp:z.hght
## z.chng:prnn
## z.change:wd 0.226
## z.chang:clt -0.259 -0.294
## z.change:td -0.310 0.085 0.089
## z.chng:z.sl 0.240 -0.203 0.169 -0.158
## z.chng:z.hg -0.230 -0.340 0.088 0.146 0.572
## z.chng:z.sd -0.045 -0.204 -0.028 -0.096 -0.512 -0.424

```


3 Summary table

Create a table summarising the fixed effects of the five models.

```
SummaryTable <-
  data.frame(
    "Exotics"=
      paste(round(fixef(traits.mod.exo), 2), ' (',
            round(fixef(traits.mod.exo) - se.fixef(traits.mod.exo) * 1.96, 2), ', ',
            round(fixef(traits.mod.exo) + se.fixef(traits.mod.exo) * 1.96, 2), ')',
            sep=''),
    "Natives"=
      paste(round(fixef(traits.mod.native), 2), ' (',
            round(fixef(traits.mod.native) - se.fixef(traits.mod.native) * 1.96, 2), ', ',
            round(fixef(traits.mod.native) + se.fixef(traits.mod.native) * 1.96, 2), ')',
            sep=''),
    check.names=F)
rownames(SummaryTable) <- names(fixef(traits.mod.exo))
```

Load the xtable package.

```
library(xtable)
```

```
SumTab1 <- xtable(SummaryTable,
  caption='Summary table of fixed effects for models of terrestrial wetland species')
align(SumTab1) <- c('l', rep('p{1.5in}', 2) )
SumTab1
```

	Exotics	Natives
(Intercept)	-7.85 (-8.83, -6.88)	-7.84 (-8.99, -6.68)
z.dep	-2.47 (-3.79, -1.14)	-3.66 (-4.63, -2.7)
z.change	0.39 (-0.49, 1.27)	-1.42 (-2.53, -0.32)
z.dep:perennial	0.51 (-0.2, 1.23)	0.64 (-0.25, 1.54)
z.dep:weed	-0.02 (-1.05, 1.01)	0.02 (-0.97, 1.02)
z.dep:cult	-0.91 (-1.76, -0.06)	-0.03 (-0.68, 0.63)
z.dep:tda	-0.4 (-1.15, 0.36)	0.33 (-0.48, 1.14)
z.dep:z.seed	0.54 (0.05, 1.03)	0.31 (0.01, 0.61)
z.dep:z.sla	0.19 (-0.33, 0.72)	-0.39 (-0.75, -0.02)
z.dep:z.height	-0.16 (-0.54, 0.22)	-0.12 (-0.44, 0.21)
z.change:perennial	0.3 (-0.18, 0.78)	0.59 (-0.53, 1.7)
z.change:weed	0.65 (-0.12, 1.41)	0.81 (-0.36, 1.98)
z.change:cult	-0.67 (-1.25, -0.1)	0.25 (-0.59, 1.08)
z.change:tda	-0.25 (-0.77, 0.28)	-1.05 (-2.1, -0.01)
z.change:z.sla	0.08 (-0.28, 0.43)	0.36 (-0.1, 0.83)
z.change:z.height	-0.09 (-0.35, 0.16)	0.01 (-0.42, 0.43)
z.change:z.seed	0.31 (-0.05, 0.66)	-0.16 (-0.54, 0.23)

Table 1: Summary table of fixed effects for models of terrestrial wetland species

3.1 Plots

The following function plots partial dependencies representing the interactions between species traits and responses to the environment and overlays the species estimated effect sizes.

```
part.plot.spp <-
function(MODEL, TRAIT, ENV, xaxis=TRUE, yaxis=TRUE,
  N=1000, xlim=c(-2, 2), ylim=c(-5, 3), binary=FALSE, hlsp=matrix(' ', 2, 2)) {
  xseq <- seq(from=xlim[1], to=xlim[2], length.out=N)
  if(binary) xlim <- c(-.5, 1.5)
  plot.new()
  plot.window(xlim=xlim, ylim=ylim, xaxs='i')
  col <- 'black'
  cnt <- 1
  for (i in MODEL) {
    fixefs <- fixef(i)[c(ENV, sprintf('%s:%s', ENV, TRAIT))]
    se.fixefs <- se.fixef(i)[c(ENV, sprintf('%s:%s', ENV, TRAIT))]
    pmat <- rnorm(N, fixefs[1], se.fixefs[1]) +
      outer(rnorm(N, fixefs[2], se.fixefs[2]), xseq)
    points(xseq, apply(pmat, 2,
      function(x) quantile(x, .025)), type='l', lty=2, col=col, lwd=2)
    points(xseq, apply(pmat, 2,
      function(x) quantile(x, .975)), type='l', lty=2, col=col, lwd=2)
    curve(fixefs[1] + fixefs[2] * x, xlim[1], xlim[2], lwd=2, add=T, col=col)
    trait.vals <- unique(i@frame[, c('species', TRAIT)]), 2]
    pchs <- rep(19, nrow(coef(i)$species))
    pchs[which(rownames(coef(i)$species) == hlsp[1, cnt])] <- 17
    pchs[which(rownames(coef(i)$species) == hlsp[2, cnt])] <- 9
    cnt <- cnt + 1
    if(binary) {
      points(jitter(trait.vals, .3), coef(i)$species[, ENV] +
        fixefs[2] * trait.vals, pch=pchs, col=col)
    } else {
      points(trait.vals, coef(i)$species[, ENV] +
        fixefs[2] * trait.vals, pch=pchs, col=col)
    }
    col <- 'grey'
  }
  box()
  if(xaxis && binary) axis(1, at=0:1, labels=c('FALSE', 'TRUE'))
  if(xaxis && !binary) axis(1, at=-1:1)
  if(yaxis) axis(2, las=1)
}
```

Plot the trait and environment interactions.

```
par(mfcol=c(2, 4), mar=c(0, 0, 0, 0), oma=c(6, 5, 1, 1))
part.plot.spp(list(traits.mod.native, traits.mod.exo) , 'z.height', 'z.dep',
  FALSE, TRUE, ylim=c(-7, 3))
mtext('Effect Size (depth)', 2, 3)
part.plot.spp(list(traits.mod.native, traits.mod.exo) , 'z.height', 'z.change',
  TRUE, TRUE, ylim=c(-7, 3))
mtext('Height', 1, 3)
mtext('Effect Size (change)', 2, 3)
part.plot.spp(list(traits.mod.native, traits.mod.exo) , 'z.seed', 'z.dep',
```

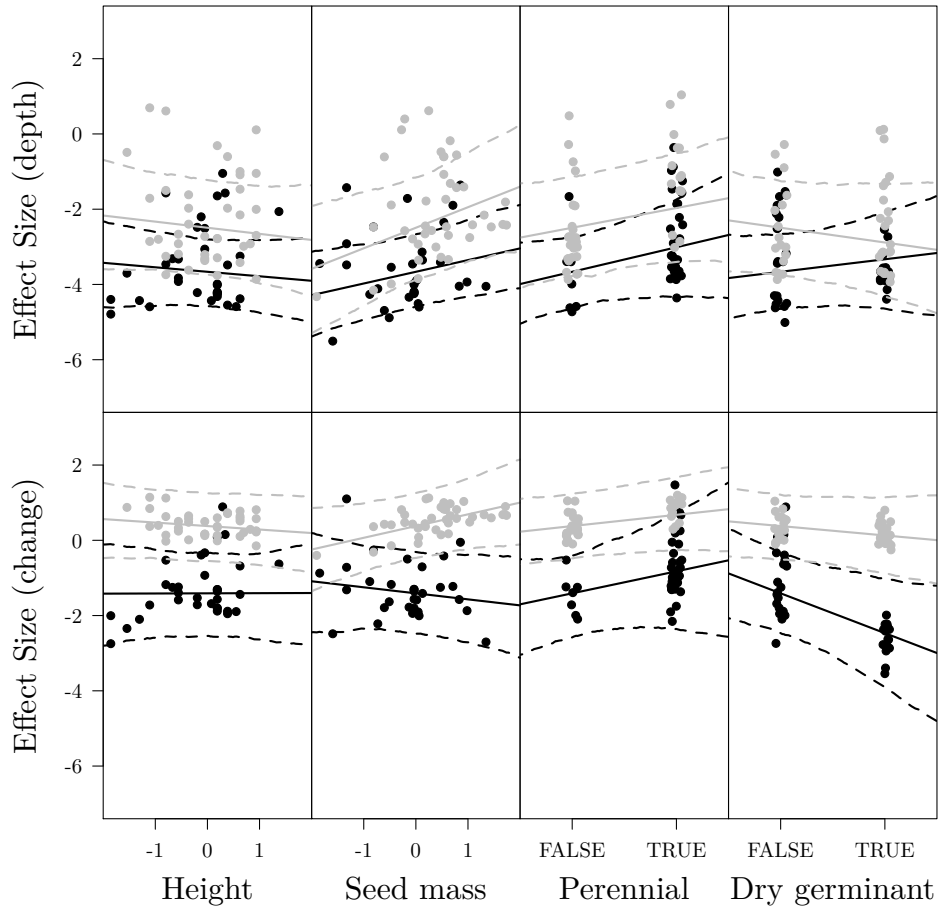


Figure 3: Partial dependency of Seed mass on the effect sizes of change in hydrograph for native (black) and exotic (grey) terrestrial plant species. Dots are individual species estimated effect sizes

```

FALSE, FALSE, ylim=c(-7, 3))
part.plot.spp(list(traits.mod.native, traits.mod.exo) , 'z.seed', 'z.change',
TRUE, FALSE, ylim=c(-7, 3))
mtext('Seed mass', 1, 3)
part.plot.spp(list(traits.mod.native, traits.mod.exo) , 'perennial', 'z.dep',
FALSE, FALSE, binary=TRUE, ylim=c(-7, 3))
part.plot.spp(list(traits.mod.native, traits.mod.exo) , 'perennial', 'z.change',
TRUE, FALSE, binary=TRUE, ylim=c(-7, 3))
mtext('Perennial', 1, 3)
part.plot.spp(list(traits.mod.native, traits.mod.exo) , 'tda', 'z.dep',
FALSE, FALSE, binary=TRUE, ylim=c(-7, 3))
part.plot.spp(list(traits.mod.native, traits.mod.exo) , 'tda', 'z.change',
TRUE, FALSE, binary=TRUE, ylim=c(-7, 3))
mtext('Dry germinant', 1, 3)

```

4 Species Characteristic Distributions

The following section describes the methods and R code used to compare the distribution of key functional traits for exotic and native terrestrial plant species. We used maximum likelihood methods to fit normal distributions to the trait data of each plant species type. We inferred approximate 95% credible intervals via simulation.

The following function was used to plot the distribution of exotic and native plant traits modelled on the log scale assuming normal distributions.

```
plot.trait.dist.mod <- function(trait, group) {
  require(MASS)
  trait <- scale(log(trait))
  mintrait <- -4
  maxtrait <- 3
  mod.group1 <- fitdistr(na.omit(trait[group]), 'normal')
  mod.group2 <- fitdistr(na.omit(trait[!group]), 'normal')
  sim1 <- mvrnorm(1001, coef(mod.group1), vcov(mod.group1))
  sim1 <- mapply(function(x, y) { dnorm(seq(mintrait, maxtrait, length.out=1001), x, y)},
    sim1[, 1], sim1[, 2])
  lo1 <- apply(sim1, 1, function(x) quantile(x, .025))
  up1 <- apply(sim1, 1, function(x) quantile(x, .975))
  sim2 <- mvrnorm(1001, coef(mod.group2), vcov(mod.group2))
  sim2 <- mapply(function(x, y) { dnorm(seq(mintrait, maxtrait, length.out=1001), x, y)},
    sim2[, 1], sim2[, 2])
  lo2 <- apply(sim2, 1, function(x) quantile(x, .025))
  up2 <- apply(sim2, 1, function(x) quantile(x, .975))
  plot.new()
  plot.window(xlim=c(mintrait, maxtrait),
    ylim=c(0 - max(up1, up2) * .1, max(up1, up2) * 1.1), xaxs='i')
  curve(dnorm(x, coef(mod.group1)[1], coef(mod.group1)[2]), mintrait, maxtrait, lwd=2,
    n=1001, add=T)
  points(seq(mintrait, maxtrait, length.out=1001), lo1, type='l', lty=2)
  points(seq(mintrait, maxtrait, length.out=1001), up1, type='l', lty=2)
  curve(dnorm(x, coef(mod.group2)[1], coef(mod.group2)[2]), mintrait, maxtrait, lwd=2,
    n=1001, col='grey', add=T)
  points(seq(mintrait, maxtrait, length.out=1001), lo2, type='l', lty=2, col='grey')
  points(seq(mintrait, maxtrait, length.out=1001), up2, type='l', lty=2, col='grey')
  rug(trait[group] + runif(length(trait[group]), -.09, .09), side=1, tcl=.6, lwd=2)
  rug(trait[!group] + runif(length(trait[!group]), -.09, .09), side=3, col='grey',
    tcl=.6, lwd=2)
  box()
}
```

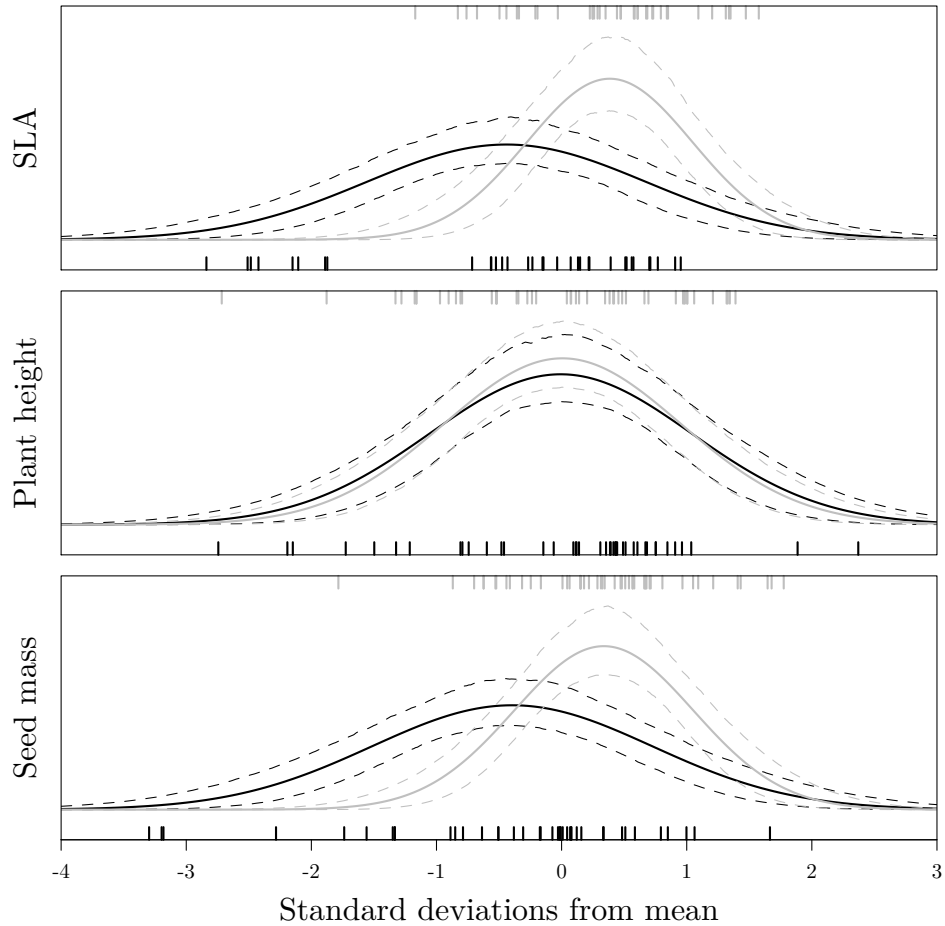


Figure 4: Comparison of modelled distribution of logged trait values for native (black) and exotic (grey) plants. Rugs show observed average species trait values. Top panel: SLA. Middle panel: plant height. Bottom panel: seed mass

Plot trait distributions.

```

par(mfrow=c(3, 1), oma=c(4, 3, 1, 1), mar=c(1, 0, 0, 0), las=1, xaxs='i')
with(subset(traits, !tree), plot.trait.dist.mod(sla, native))
mtext(side=2, text='SLA', las=0, line=1)
with(subset(traits, !tree), plot.trait.dist.mod(height, native))
mtext(side=2, text='Plant height', las=0, line=1)
with(subset(traits, !tree), plot.trait.dist.mod(seed, native))
mtext(side=2, text='Seed mass', las=0, line=1)
axis(side=1)
mtext(side=1, text='Standard deviations from mean', line=3)

```

References

- Gelman, A. (2005). Analysis of variance—why it is more important than ever. *The Annals of Statistics* **33**, 1–53.
- Hector, A., T. Bell, Y. Hautier, F. Isbell, M. Kery, P. Reich, J. van Ruijven & B. Schmid (2011). BUGS in the analysis of biodiversity experiments: species richness and composition are of similar importance for grassland productivity. *PloS one* **6**, e17434.
- Pollock, L. J., W. K. Morris & P. A. Vesik (2012). The role of functional traits in species distributions revealed through a hierarchical model. *Ecography* **35**, 716–725.
- R Core Team (2012). *R: A Language and Environment for Statistical Computing* R Foundation for Statistical Computing (Vienna, Austria). <http://www.R-project.org/>.