```r
#Not run
# In case of **missing sampling years** in time series; we filled the missing years with NAs
#new.row <- head(Time_series1[NA,], 5)    #Create new row
#new.row["year"] <- c(2007,2013,2014,2015,2016)
#new.row["year"] <- 2006
# assign the year without data
```

```r
#We calculated a modified slope time series following Hamed & Rao 1998; Pilotto et al. 2020


My.mmkh=function (x, ci = 0.95)
{
  x = x
  z = NULL
  z0 = NULL
  pval = NULL
  pval0 = NULL
  S = 0
  Tau = NULL
  essf = NULL
  ci = ci
  if (is.vector(x) == FALSE) {
    stop("Input data must be a vector")
  }
  if (any(is.finite(x) == FALSE)) {
    x <- x[-c(which(is.finite(x) == FALSE))]
    warning("The input vector contains non-finite numbers. An attempt was made to remove them")
  }
  n <- length(x)
  V <- rep(NA, n * (n - 1)/2)
  k = 0
  for (i in 1:(n - 1)) {
    for (j in (i + 1):n) {
      k = k + 1
      V[k] = (x[j] - x[i])/(j - i)
    }
  }
  slp <- median(V, na.rm = TRUE)
  t = 1:length(x)
  xn <- (x[1:n]) - ((slp) * (t))
  for (i in 1:(n - 1)) {
    for (j in (i + 1):n) {
      S = S + sign(x[j] - x[i])
    }
  }
  ro <- acf(rank(xn), lag.max = (n - 1), plot = FALSE)$acf[-1]
  sig <- qnorm((1 + ci)/2)/sqrt(n)
  rof <- rep(NA, length(ro))
  for (i in 1:(length(ro))) {
    if (ro[i] > sig || ro[i] < -sig) {
      rof[i] <- ro[i]
    }
    else {
```

```r
      rof[i] = 0
    }
  }
  cte <- 2/(n * (n - 1) * (n - 2))
  ess = 0
  for (i in 1:(n - 1)) {
    ess = ess + (n - i) * (n - i - 1) * (n - i - 2) * rof[i]
  }
  essf = 1 + ess * cte
  var.S = n * (n - 1) * (2 * n + 5) * (1/18)
  if (length(unique(x)) < n) {
    aux <- unique(x)
    for (i in 1:length(aux)) {
      tie <- length(which(x == aux[i]))
      if (tie > 1) {
        var.S = var.S - tie * (tie - 1) * (2 * tie +
                                            5) * (1/18)
      }
    }
  }
  VS = var.S * essf
  if (S == 0) {
    z = 0
    z0 = 0
  }
  if (S > 0) {
    z = (S - 1)/sqrt(VS)
    z0 = (S - 1)/sqrt(var.S)
  }
  else {
    z = (S + 1)/sqrt(VS)
    z0 = (S + 1)/sqrt(var.S)
  }
  pval = 2 * pnorm(-abs(z))
  pval0 = 2 * pnorm(-abs(z0))
  Tau = S/(0.5 * n * (n - 1))
  return(c("Corrected Zc" = z, "new P-value" = pval, "N/N*" = essf,
           "Original Z" = z0, "old P.value" = pval0, "Tau" = Tau,
           "Sen s slope" = slp, "old.variance" = var.S, "new.variance" = VS, "S statistic" = S, "n" = n
}

#MK <-as.data.frame(do.call(rbind,lapply(xy.list[1:96],function(x)unlist(My.mmkh(x)))))
#head(MK)
```

```r
# Meta-regression modelling

##Meta-regression

df1<- df[!duplicated(df$site_id), ] #Remove duplicated
df1<- df1 %>% mutate(const=1) #Create a new column with "1"

# In the next code E and N refers to coordinates of time series
```

```r
#Dikerogammarus villosus trend
res <- rma.mv(S_Dv, Var_Dv,  random = ~ E+N | const,  struct="SPGAU", data= df1)
res
```

```
##
## Multivariate Meta-Analysis Model (k = 96; method: REML)
##
## Variance Components:
##
## outer factor: const  (nlvls = 1)
## inner term:   ~E + N (nlvls = 92)
##
##                estim     sqrt  fixed
## tau^2      160.8138  12.6812     no
## rho          6.1105              no
##
## Test for Heterogeneity:
## Q(df = 95) = 132.4769, p-val = 0.0067
##
## Model Results:
##
## estimate      se    zval    pval    ci.lb    ci.ub
##   4.7476  7.1808  0.6612  0.5085  -9.3265  18.8217
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Total abundance
res1 <- rma.mv(S_Abun, Var_Abun,  random = ~ E+N | const, struct="SPGAU",
               mods=~ S_Dv +Middle_point , data= df1, control=list(maxiter=1000))
res1
```

```
##
## Multivariate Meta-Analysis Model (k = 96; method: REML)
##
## Variance Components:
##
## outer factor: const  (nlvls = 1)
## inner term:   ~E + N (nlvls = 92)
##
##              estim    sqrt  fixed
## tau^2      6.2111  2.4922     no
## rho        5.5337             no
##
## Test for Residual Heterogeneity:
## QE(df = 93) = 124.0067, p-val = 0.0175
##
## Test of Moderators (coefficients 2:3):
## QM(df = 2) = 3.4372, p-val = 0.1793
##
## Model Results:
##
##                 estimate        se     zval    pval      ci.lb     ci.ub
```

```
## intrcpt       -265.6340  271.4396  -0.9786  0.3278  -797.6458  266.3777
## S_Dv             0.1312    0.0912   1.4390  0.1501    -0.0475    0.3100
## Middle_point     0.1321    0.1352   0.9767  0.3287    -0.1330    0.3971
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Taxa richness
res2 <- rma.mv(S_Rich,Var_Rich, random = ~ E+N | const ,
               mods=~ S_Dv +Middle_point, struct="SPGAU",
               data= df1, control=list(maxiter=1000))
res2
```

```
##
## Multivariate Meta-Analysis Model (k = 96; method: REML)
##
## Variance Components:
##
## outer factor: const  (nlvls = 1)
## inner term:   ~E + N (nlvls = 92)
##
##              estim    sqrt  fixed
## tau^2      39.1752  6.2590     no
## rho         0.4714             no
##
## Test for Residual Heterogeneity:
## QE(df = 93) = 196.3242, p-val < .0001
##
## Test of Moderators (coefficients 2:3):
## QM(df = 2) = 6.6325, p-val = 0.0363
##
## Model Results:
##
##                 estimate        se     zval    pval      ci.lb     ci.ub
## intrcpt       -236.2580  416.0980  -0.5678  0.5702  -1051.7952  579.2791
## S_Dv            -0.2589    0.1043  -2.4819  0.0131    -0.4634   -0.0544  *
## Middle_point     0.1175    0.2074   0.5663  0.5712    -0.2891    0.5240
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Shannon diversity
res3 <- rma.mv(S_Diver, Var_Diver, random = ~ E+N | const,
               mods=~ S_Dv +Middle_point, struct="SPGAU",
               data= df1, control=list(maxiter=1000))
res3
```

```
##
## Multivariate Meta-Analysis Model (k = 96; method: REML)
##
## Variance Components:
##
## outer factor: const  (nlvls = 1)
```

```
## inner term:   ~E + N (nlvls = 92)
##
##               estim    sqrt  fixed
## tau^2       14.8776  3.8571     no
## rho          0.5535             no
##
## Test for Residual Heterogeneity:
## QE(df = 93) = 155.6233, p-val < .0001
##
## Test of Moderators (coefficients 2:3):
## QM(df = 2) = 4.9651, p-val = 0.0835
##
## Model Results:
##
##                estimate        se     zval    pval      ci.lb     ci.ub
## intrcpt        -17.9526  320.6955  -0.0560  0.9554  -646.5042  610.5989
## S_Dv            -0.2187    0.0982  -2.2271  0.0259    -0.4112   -0.0262  *
## Middle_point     0.0084    0.1599   0.0523  0.9583    -0.3051    0.3218
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Temporal turnover
res4 <- rma.mv(S_Turn, Var_Turn, random = ~ E+N | const, struct="SPGAU",mods=~ S_Dv +
                 Middle_point, data= df1, control=list(maxiter=1000))
```

```
## Warning: Rows with NAs omitted from model fitting.
```

```
res4
```

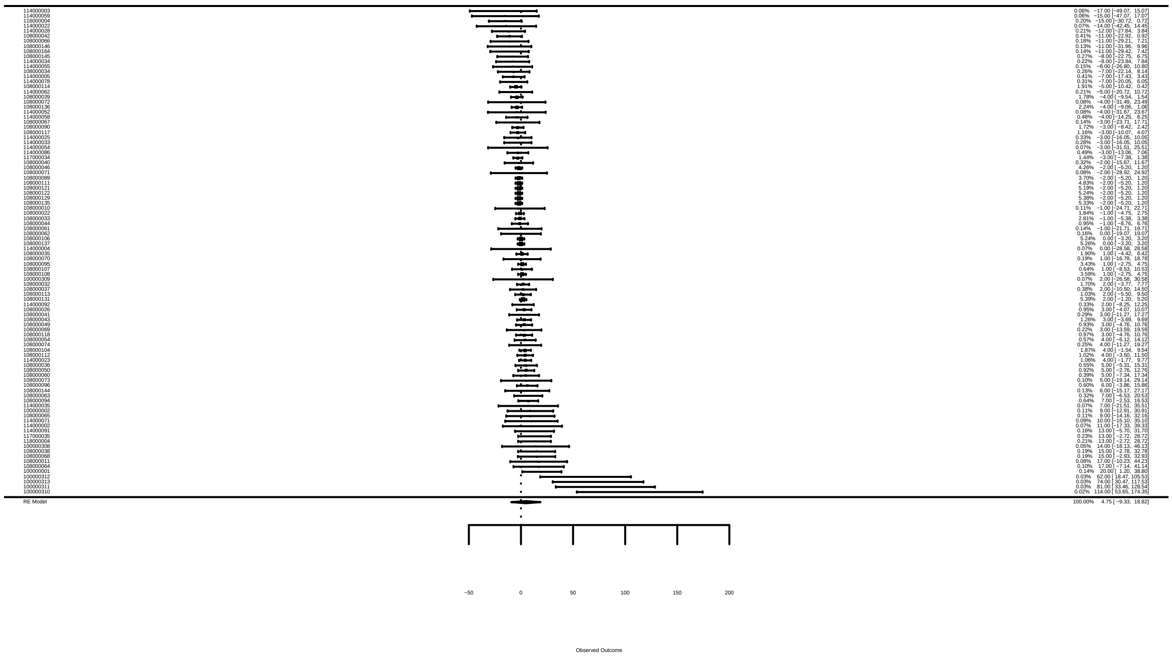
```
##
## Multivariate Meta-Analysis Model (k = 84; method: REML)
##
## Variance Components:
##
## outer factor: const  (nlvls = 1)
## inner term:   ~E + N (nlvls = 92)
##
##              estim    sqrt  fixed
## tau^2       3.1708  1.7807     no
## rho         0.1649             no
##
## Test for Residual Heterogeneity:
## QE(df = 81) = 122.8135, p-val = 0.0019
##
## Test of Moderators (coefficients 2:3):
## QM(df = 2) = 4.2444, p-val = 0.1198
##
## Model Results:
##
##                estimate        se     zval    pval      ci.lb     ci.ub
## intrcpt         61.5749  192.4691   0.3199  0.7490  -315.6577  438.8074
## S_Dv            -0.1631    0.0798  -2.0436  0.0410    -0.3195   -0.0067  *
```

```
## Middle_point   -0.0305    0.0960  -0.3177  0.7507    -0.2187    0.1577
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

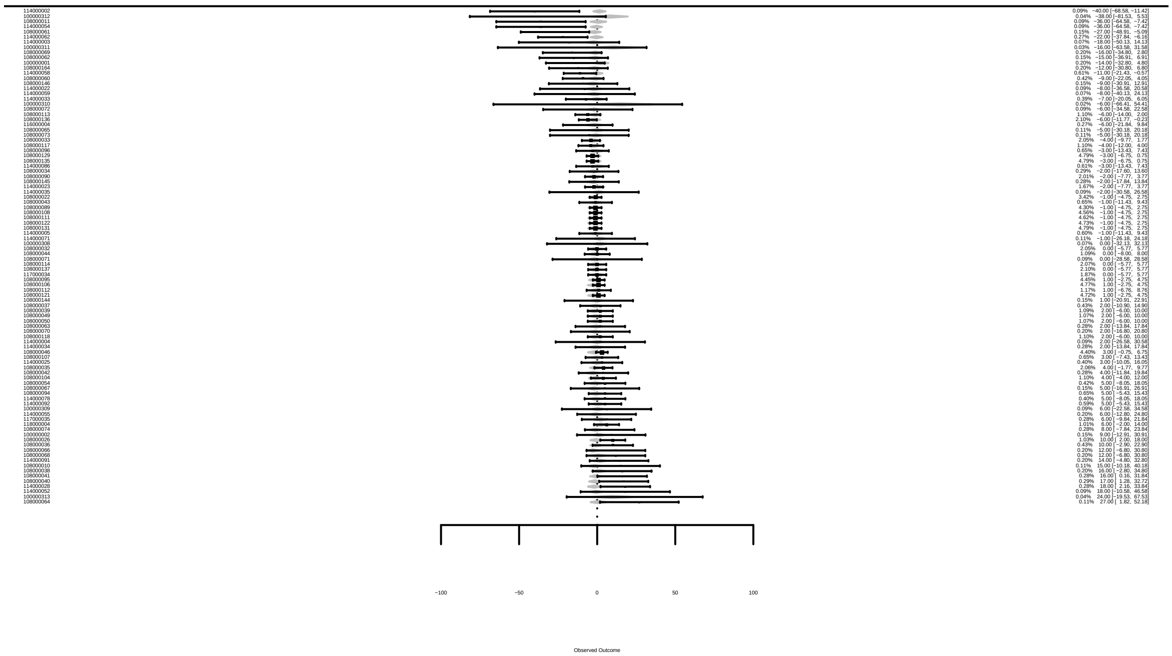
```r
#Temporal Evenness
res5 <- rma.mv(S_Eve, Var_Eve, random = ~ E+N | const, struct="SPGAU",mods=~ S_Dv +
                 Middle_point, data= df1, control=list(maxiter=1000))
res5
```

```
##
## Multivariate Meta-Analysis Model (k = 96; method: REML)
##
## Variance Components:
##
## outer factor: const  (nlvls = 1)
## inner term:   ~E + N (nlvls = 92)
##
##             estim    sqrt  fixed
## tau^2      3.6182  1.9022     no
## rho        0.9510             no
##
## Test for Residual Heterogeneity:
## QE(df = 93) = 123.3304, p-val = 0.0193
##
## Test of Moderators (coefficients 2:3):
## QM(df = 2) = 2.8616, p-val = 0.2391
##
## Model Results:
##
##               estimate        se     zval    pval      ci.lb     ci.ub
## intrcpt        67.2929  245.2473   0.2744  0.7838  -413.3830  547.9688
## S_Dv           -0.1512    0.0914  -1.6535  0.0982    -0.3303    0.0280  .
## Middle_point   -0.0342    0.1224  -0.2791  0.7801    -0.2740    0.2057
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

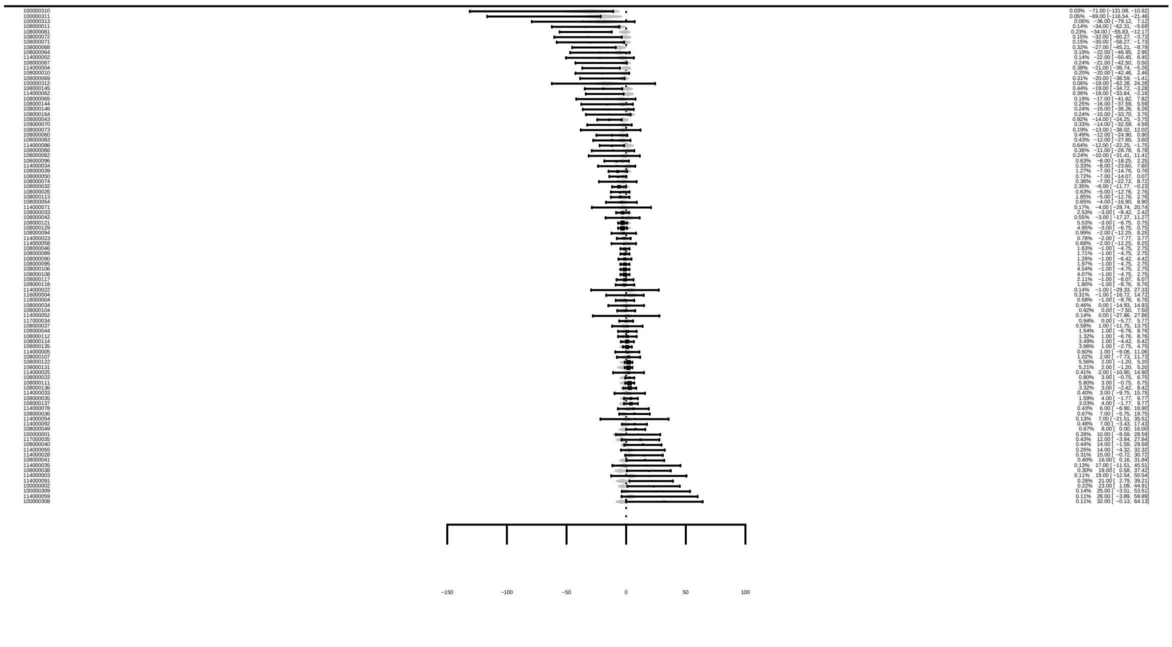
```r
###Funnel plot
#D. villosus
forest(res, showweights = T, order="obs", slab= df1$site_id)
```



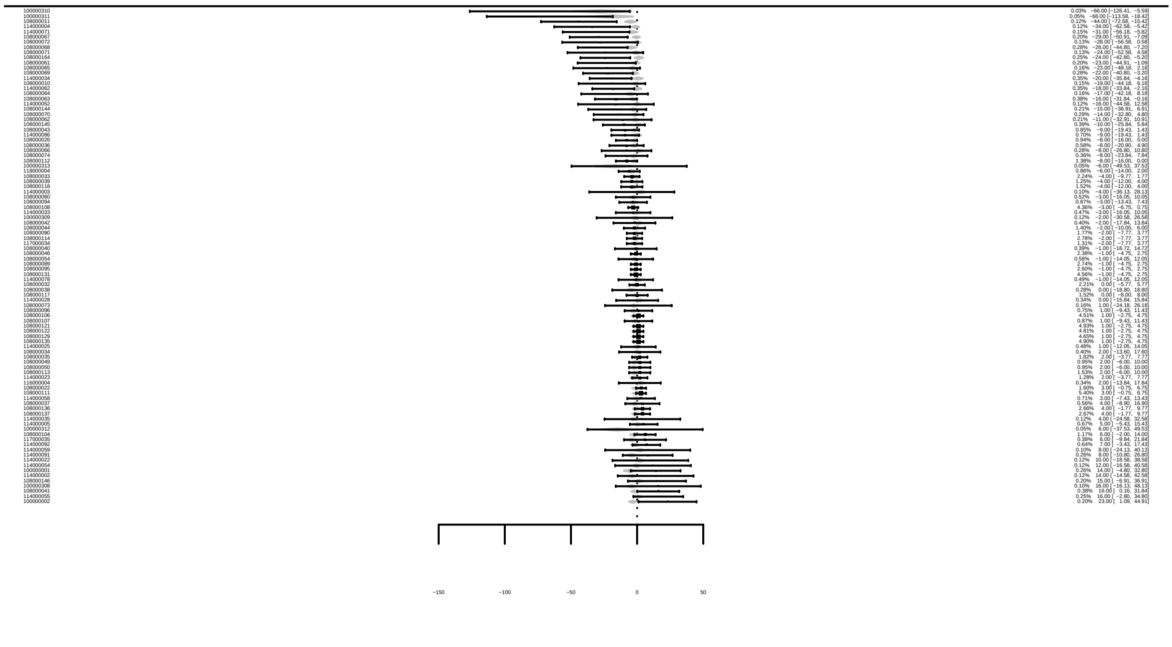

```
#Total abundance
forest(res1, showweights = T, order="obs", slab= df1$site_id)
```



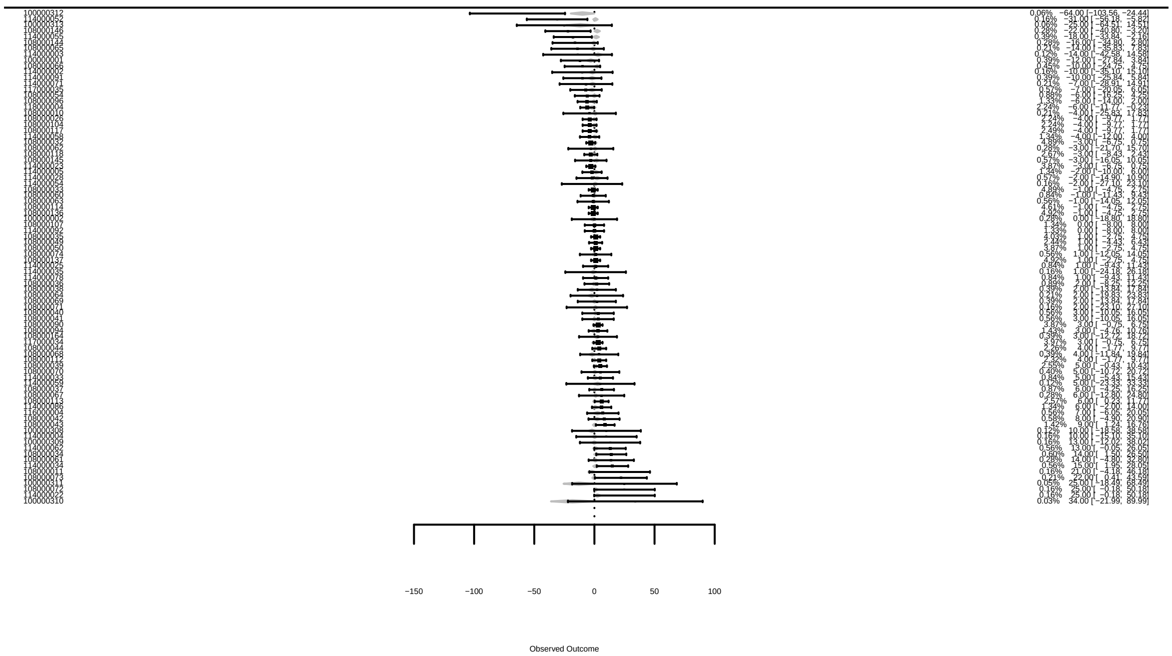
```
#Taxa Richness
forest(res2, showweights = T, order="obs", slab= df1$site_id)
```



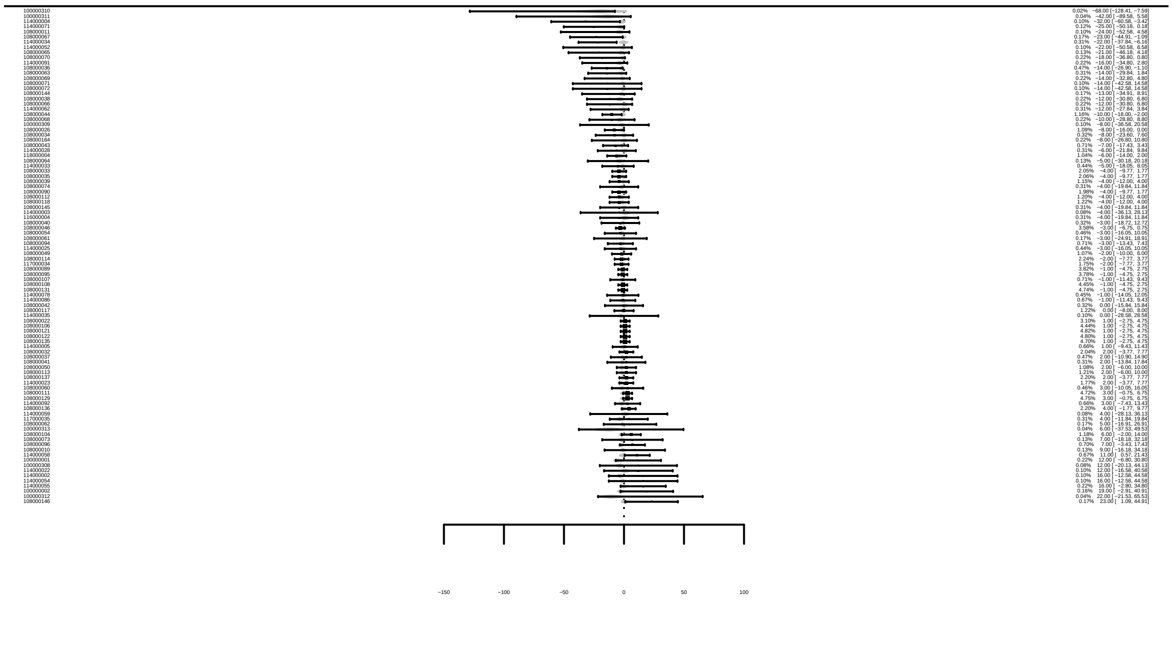

```
#Shannon diversity
forest(res3, showweights = T, order="obs", slab= df1$site_id)
```




```r
#Temporal turnover
forest(res4, showweights = T, order="obs", slab= df1$site_id)
```



```r
#Temporal Evenness
forest(res5, showweights = T, order="obs", slab= df1$site_id)
```




```r
### Egger Test

resid = rstandard(res)
eggers <- regtest(x = resid$resid, sei =sqrt(df1$Var_Dv), model = "lm")
eggers
```

```
##
## Regression Test for Funnel Plot Asymmetry
##
## Model:     weighted regression with multiplicative dispersion
## Predictor: standard error
##
## Test for Funnel Plot Asymmetry: t = 3.1424, df = 94, p = 0.0022
## Limit Estimate (as sei -> 0):   b = -6.8256 (CI: -8.1298, -5.5215)
```

```r
resid = rstandard(res1)
eggers <- regtest(x = resid$resid, sei =sqrt(df1$Var_Abun), model = "lm")
eggers
```

```
##
## Regression Test for Funnel Plot Asymmetry
##
## Model:     weighted regression with multiplicative dispersion
## Predictor: standard error
##
```

```
## Test for Funnel Plot Asymmetry: t = -0.9547, df = 94, p = 0.3422
## Limit Estimate (as sei -> 0):   b = 1.6932 (CI: 0.1133, 3.2731)
```

```r
resid = rstandard(res2)
eggers <- regtest(x = resid$resid, sei =sqrt(df1$Var_Rich), model = "lm")
eggers
```

```
##
## Regression Test for Funnel Plot Asymmetry
##
## Model:     weighted regression with multiplicative dispersion
## Predictor: standard error
##
## Test for Funnel Plot Asymmetry: t = -2.6753, df = 94, p = 0.0088
## Limit Estimate (as sei -> 0):   b = 2.4668 (CI: 0.6173, 4.3163)
```

```r
resid = rstandard(res3)
eggers <- regtest(x = resid$resid, sei =sqrt(df1$Var_Diver), model = "lm")
eggers
```

```
##
## Regression Test for Funnel Plot Asymmetry
##
## Model:     weighted regression with multiplicative dispersion
## Predictor: standard error
##
## Test for Funnel Plot Asymmetry: t = -3.3503, df = 94, p = 0.0012
## Limit Estimate (as sei -> 0):   b = 2.8965 (CI: 1.2260, 4.5669)
```

```r
###

resid = rstandard(res5)
eggers <- regtest(x = resid$resid, sei =sqrt(df1$Var_Eve), model = "lm")
eggers
```

```
##
## Regression Test for Funnel Plot Asymmetry
##
## Model:     weighted regression with multiplicative dispersion
## Predictor: standard error
##
## Test for Funnel Plot Asymmetry: t = -2.5005, df = 94, p = 0.0141
## Limit Estimate (as sei -> 0):   b = 1.9234 (CI: 0.3999, 3.4469)
```

```r
## Generalised Lineal Model

#Before run this model, first we checked the collinearity using the corvif function
# Functions from Zuur et al., 2009
myvif <- function(mod) {
  v <- vcov(mod)
  assign <- attributes(model.matrix(mod))$assign
```

```
  if (names(coefficients(mod)[1]) == "(Intercept)") {
    v <- v[-1, -1]
    assign <- assign[-1]
  } else warning("No intercept: vifs may not be sensible.")
  terms <- labels(terms(mod))
  n.terms <- length(terms)
  if (n.terms < 2) stop("The model contains fewer than 2 terms")
  if (length(assign) > dim(v)[1] ) {
    diag(tmp_cor)<-0
    if (any(tmp_cor==1.0)){
      return("Sample size is too small, 100% collinearity is present")
    } else {
      return("Sample size is too small")
    }
  }
  R <- cov2cor(v)
  detR <- det(R)
  result <- matrix(0, n.terms, 3)
  rownames(result) <- terms
  colnames(result) <- c("GVIF", "Df", "GVIF^(1/2Df)")
  for (term in 1:n.terms) {
    subs <- which(assign == term)
    result[term, 1] <- det(as.matrix(R[subs, subs])) * det(as.matrix(R[-subs, -subs])) / detR
    result[term, 2] <- length(subs)
  }
  if (all(result[, 2] == 1)) {
    result <- data.frame(GVIF=result[, 1])
  } else {
    result[, 3] <- result[, 1]^(1/(2 * result[, 2]))
  }
  invisible(result)
}

corvif <- function(dataz) {
  dataz <- as.data.frame(dataz)
  #correlation part
  cat("Correlations of the variables\n\n")
  tmp_cor <- cor(dataz,use="complete.obs")
  print(tmp_cor)

  #vif part
  form    <- formula(paste("fooy ~ ",paste(strsplit(names(dataz)," "),collapse=" + ")))
  dataz   <- data.frame(fooy=1,dataz)
  lm_mod  <- lm(form,dataz)

  cat("\n\nVariance inflation factors\n\n")
  print(myvif(lm_mod))}
```

```
#VIF threshold = 5
df1 <- df[!duplicated(df$site_id),]
#VIF for Dikerogammarus villosus trend (GLM)
moderators <- c("S_Dv", "E","N","Elevation","Slope_preci","Slope_temp","Mean_precipita",
                "Mean_temp","DistanceKm","slope","Avg_Tmin","Slope_Tmin","Slope_Tmax","Avg_Tmax","Middl
```

```
corvif(mutate_if(df1[, moderators], is.factor, as.numeric))
```

```
## Correlations of the variables
##
##                         S_Dv           E            N   Elevation Slope_preci
## S_Dv             1.000000000 -0.29498281 -0.351797217  0.15779035  0.14928080
## E               -0.294982812  1.00000000 -0.509808913  0.10971656 -0.31387298
## N               -0.351797217 -0.50980891  1.000000000 -0.58055294  0.25742546
## Elevation        0.157790351  0.10971656 -0.580552937  1.00000000 -0.16321570
## Slope_preci      0.149280802 -0.31387298  0.257425459 -0.16321570  1.00000000
## Slope_temp      -0.176766023  0.09972639  0.135946452 -0.03922977  0.04496056
## Mean_precipita  0.298390969 -0.79633102  0.278502472  0.11345394  0.25228742
## Mean_temp       -0.003013052 -0.04015251 -0.003959704  0.01862232 -0.03544936
## DistanceKm      -0.041428717 -0.71546194  0.627811104 -0.15855460  0.35163388
## slope           -0.084355395 -0.14827363  0.253301241 -0.15305841 -0.01553525
## Avg_Tmin         0.485955006 -0.20680600 -0.270498368 -0.14832896  0.09242185
## Slope_Tmin       0.693021874 -0.01561980 -0.483938608  0.16111401 -0.11848532
## Slope_Tmax       0.590978218  0.03641882 -0.453067046  0.03735041 -0.03169006
## Avg_Tmax         0.049125350 -0.06433243  0.049799087 -0.09281866  0.02712434
## Middle_point     0.149434563  0.66537721 -0.734498506  0.38435663 -0.36950036
##                   Slope_temp Mean_precipita    Mean_temp  DistanceKm       slope
## S_Dv            -0.176766023    0.298390969 -0.003013052 -0.04142872 -0.08435539
## E                0.099726391   -0.796331024 -0.040152514 -0.71546194 -0.14827363
## N                0.135946452    0.278502472 -0.003959704  0.62781110  0.25330124
## Elevation       -0.039229769    0.113453943  0.018622320 -0.15855460 -0.15305841
## Slope_preci      0.044960563    0.252287415 -0.035449362  0.35163388 -0.01553525
## Slope_temp       1.000000000   -0.085450335  0.002311115  0.01443316  0.03836502
## Mean_precipita -0.085450335    1.000000000 -0.073249397  0.47207321  0.16067729
## Mean_temp        0.002311115   -0.073249397  1.000000000  0.04169471 -0.06960749
## DistanceKm       0.014433162    0.472073214  0.041694706  1.00000000  0.13214946
## slope            0.038365023    0.160677291 -0.069607495  0.13214946  1.00000000
## Avg_Tmin        -0.233667316    0.290416759 -0.106685807  0.03322479  0.02036989
## Slope_Tmin      -0.201452407    0.057561053  0.006972887 -0.30794912 -0.10458632
## Slope_Tmax      -0.134045510   -0.003790154 -0.018320256 -0.25094452 -0.07649401
## Avg_Tmax        -0.015277968    0.097924450 -0.016744689  0.05111826  0.20573252
## Middle_point     0.005417120   -0.345651098 -0.079491787 -0.68769070 -0.18344594
##                   Avg_Tmin   Slope_Tmin   Slope_Tmax    Avg_Tmax Middle_point
## S_Dv            0.48595501  0.693021874  0.590978218  0.04912535   0.14943456
## E              -0.20680600 -0.015619797  0.036418818 -0.06433243   0.66537721
## N              -0.27049837 -0.483938608 -0.453067046  0.04979909  -0.73449851
## Elevation      -0.14832896  0.161114014  0.037350413 -0.09281866   0.38435663
## Slope_preci     0.09242185 -0.118485323 -0.031690062  0.02712434  -0.36950036
## Slope_temp     -0.23366732 -0.201452407 -0.134045510 -0.01527797   0.00541712
## Mean_precipita  0.29041676  0.057561053 -0.003790154  0.09792445  -0.34565110
## Mean_temp      -0.10668581  0.006972887 -0.018320256 -0.01674469  -0.07949179
## DistanceKm      0.03322479 -0.307949116 -0.250944519  0.05111826  -0.68769070
## slope           0.02036989 -0.104586317 -0.076494009  0.20573252  -0.18344594
## Avg_Tmin        1.00000000  0.387700973  0.401550422  0.06856120   0.18040436
## Slope_Tmin      0.38770097  1.000000000  0.841810142 -0.01953010   0.35880710
## Slope_Tmax      0.40155042  0.841810142  1.000000000 -0.07777827   0.28159836
## Avg_Tmax        0.06856120 -0.019530096 -0.077778269  1.00000000  -0.08413697
## Middle_point    0.18040436  0.358807104  0.281598356 -0.08413697   1.00000000
##
```

```
##
## Variance inflation factors
##
##                     GVIF
## S_Dv            2.930936
## E               8.885482
## N               7.404985
## Elevation       3.885924
## Slope_preci     1.361869
## Slope_temp      1.129546
## Mean_precipita 4.266825
## Mean_temp       1.064678
## DistanceKm      3.574575
## slope           1.168368
## Avg_Tmin        2.733108
## Slope_Tmin      5.566225
## Slope_Tmax      4.377839
## Avg_Tmax        1.120344
## Middle_point    4.310337
```

```
#GLM model of D. villosus trend
m1 <- glm(S_Dv ~
            Elevation+  #Elevation
            Slope_preci+ #Trend precipitation
            Slope_temp+  # Trend of temperature
            Mean_precipita + #Average precipitation
            Mean_temp+  # Average temperature
            DistanceKm+ #Distance to the next barrier
            slope+  #Slope of stream
            Avg_Tmax+  # Avg maximum temperature
            Avg_Tmin+  # Aveg minimum temperature
            Slope_Tmax+ # Trend of maximum temperature
            Middle_point+ #Middle point of time series
            Biogeo+ #Biogeographical regions
            Ecosystem, #Ecosystem type (stream/canal)
          data = df1, na.action="na.fail")

summary(m1)
```

```
##
## Call:
## glm(formula = S_Dv ~ Elevation + Slope_preci + Slope_temp + Mean_precipita +
##     Mean_temp + DistanceKm + slope + Avg_Tmax + Avg_Tmin + Slope_Tmax +
##     Middle_point + Biogeo + Ecosystem, data = df1, na.action = "na.fail")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -34.289   -5.946    0.234    4.610   57.075
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -1.251e+03  9.146e+02  -1.368   0.1751
## Elevation             6.460e-02  4.323e-02   1.494   0.1391
## Slope_preci           3.625e-01  2.019e-01   1.795   0.0764 .
```

```
## Slope_temp          -2.839e-02  1.972e-01  -0.144   0.8859
## Mean_precipita       4.819e+00  6.854e+00   0.703   0.4841
## Mean_temp            5.715e-02  1.190e-01   0.480   0.6322
## DistanceKm          -1.705e-02  8.027e-03  -2.124   0.0368 *
## slope               -2.910e+01  5.395e+01  -0.539   0.5911
## Avg_Tmax             8.386e-02  5.204e-02   1.611   0.1111
## Avg_Tmin             5.593e+00  2.762e+00   2.025   0.0463 *
## Slope_Tmax           8.463e-01  1.587e-01   5.334 8.92e-07 ***
## Middle_point         5.936e-01  4.569e-01   1.299   0.1976
## BiogeoAtlantic       1.382e+01  1.244e+01   1.111   0.2699
## BiogeoContinental    1.758e+01  1.092e+01   1.610   0.1114
## BiogeoMediterranean  3.155e+01  1.417e+01   2.226   0.0289 *
## BiogeoPannonian     -6.068e+00  1.248e+01  -0.486   0.6281
## Ecosystemminor       5.059e+00  3.530e+00   1.433   0.1558
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 145.8679)
##
##     Null deviance: 32545  on 95  degrees of freedom
## Residual deviance: 11524  on 79  degrees of freedom
## AIC: 768.07
##
## Number of Fisher Scoring iterations: 2
```

```
# VIF for realtive abundance of D. villosus over time (threshold =5)
moderators <- c("Proportion", "E","N","Elevation","Precipitation","Temperature",
                "DistanceKm","slope","Middle_point","Tmax","Tmin","year","Avg_Tmax","Avg_Tmin","Mean_pre

corvif(mutate_if(df[, moderators], is.factor, as.numeric))
```

```
## Correlations of the variables
##
##                  Proportion           E           N   Elevation Precipitation
## Proportion      1.0000000000 -0.13771421 -0.08933161  0.04772116   -0.02575671
## E              -0.1377142066  1.00000000 -0.37871524  0.11495884   -0.63538152
## N              -0.0893316100 -0.37871524  1.00000000 -0.63008742    0.13473235
## Elevation       0.0477211574  0.11495884 -0.63008742  1.00000000    0.06423241
## Precipitation  -0.0257567082 -0.63538152  0.13473235  0.06423241    1.00000000
## Temperature     0.0227585144 -0.01084357 -0.10461510  0.01740506   -0.02646033
## DistanceKm      0.0649135474 -0.71496572  0.67036889 -0.28847010    0.37476891
## slope          -0.0420845137 -0.12623753  0.30788133 -0.20134744    0.08586506
## Middle_point   -0.0314412131  0.59524009 -0.78820626  0.50144197   -0.24895233
## Tmax            0.0004026618 -0.02758705  0.02080298 -0.02992577    0.03180707
## Tmin            0.0747607842 -0.28548342 -0.30581310 -0.04284096    0.26605626
## year            0.0155292051  0.50860439 -0.66934395  0.41649051   -0.25044199
## Avg_Tmax       -0.0054822228 -0.08156937  0.06151025 -0.08848448    0.05915762
## Avg_Tmin        0.1221882346 -0.35777588 -0.38325360 -0.05368950    0.30583845
## Mean_precipita  0.0351239648 -0.85389563  0.18182249  0.11447899    0.74554944
## Mean_temp       0.0214439705 -0.04139097 -0.03098955  0.02179044   -0.03658465
##                 Temperature  DistanceKm       slope Middle_point          Tmax
## Proportion      2.275851e-02  0.06491355 -0.04208451  -0.03144121  0.0004026618
## E              -1.084357e-02 -0.71496572 -0.12623753   0.59524009 -0.0275870527
## N              -1.046151e-01  0.67036889  0.30788133  -0.78820626  0.0208029849
```

```
## Elevation       1.740506e-02 -0.28847010 -0.20134744   0.50144197 -0.0299257670
## Precipitation  -2.646033e-02  0.37476891  0.08586506  -0.24895233  0.0318070691
## Temperature     1.000000e+00 -0.02951438 -0.01198523   0.03069880 -0.0043279443
## DistanceKm     -2.951438e-02  1.00000000  0.16953588  -0.76693620  0.0252418673
## slope          -1.198523e-02  0.16953588  1.00000000  -0.23176037  0.0763297380
## Middle_point    3.069880e-02 -0.76693620 -0.23176037   1.00000000 -0.0390550776
## Tmax           -4.327944e-03  0.02524187  0.07632974  -0.03905508  1.0000000000
## Tmin            8.403918e-02  0.03128626 -0.03114691   0.13939180  0.0303460163
## year            5.264916e-02 -0.66321528 -0.18757368   0.84510087 -0.0579431434
## Avg_Tmax       -3.058632e-03  0.07463513  0.22569171  -0.11547803  0.3382035588
## Avg_Tmin        1.111711e-01  0.03920882 -0.03903418   0.17468997  0.0159486509
## Mean_precipita  4.249386e-05  0.51963356  0.12021276  -0.34753921  0.0341113361
## Mean_temp       9.496474e-02  0.02947892 -0.03453937  -0.05452362 -0.0042694408
##                       Tmin        year     Avg_Tmax    Avg_Tmin Mean_precipita
## Proportion      0.07476078  0.01552921 -0.005482223  0.12218823   3.512396e-02
## E              -0.28548342  0.50860439 -0.081569374 -0.35777588  -8.538956e-01
## N              -0.30581310 -0.66934395  0.061510248 -0.38325360   1.818225e-01
## Elevation      -0.04284096  0.41649051 -0.088484483 -0.05368950   1.144790e-01
## Precipitation   0.26605626 -0.25044199  0.059157623  0.30583845   7.455494e-01
## Temperature     0.08403918  0.05264916 -0.003058632  0.11117114   4.249386e-05
## DistanceKm      0.03128626 -0.66321528  0.074635132  0.03920882   5.196336e-01
## slope          -0.03114691 -0.18757368  0.225691706 -0.03903418   1.202128e-01
## Middle_point    0.13939180  0.84510087 -0.115478027  0.17468997  -3.475392e-01
## Tmax            0.03034602 -0.05794314  0.338203559  0.01594865   3.411134e-02
## Tmin            1.00000000  0.21945107  0.037628389  0.79793928   3.121934e-01
## year            0.21945107  1.00000000 -0.097225051  0.15529389  -2.954997e-01
## Avg_Tmax        0.03762839 -0.09722505  1.000000000  0.04715696   1.008604e-01
## Avg_Tmin        0.79793928  0.15529389  0.047156958  1.00000000   3.912496e-01
## Mean_precipita  0.31219343 -0.29549965  0.100860370  0.39124960   1.000000e+00
## Mean_temp      -0.01180118 -0.04595187 -0.012623879 -0.01478957  -3.240469e-02
##                   Mean_temp
## Proportion      0.021443971
## E              -0.041390968
## N              -0.030989554
## Elevation       0.021790436
## Precipitation  -0.036584647
## Temperature     0.094964735
## DistanceKm      0.029478919
## slope          -0.034539366
## Middle_point   -0.054523621
## Tmax           -0.004269441
## Tmin           -0.011801180
## year           -0.045951873
## Avg_Tmax       -0.012623879
## Avg_Tmin       -0.014789571
## Mean_precipita -0.032404686
## Mean_temp       1.000000000
##
##
## Variance inflation factors
##
##                     GVIF
## Proportion     1.118597
## E              8.011324
```

```
## N              8.710754
## Elevation      4.286684
## Precipitation  2.320554
## Temperature    1.036741
## DistanceKm     4.336794
## slope          1.201090
## Middle_point   7.356723
## Tmax           1.134826
## Tmin           3.065245
## year           3.979254
## Avg_Tmax       1.215896
## Avg_Tmin       5.920547
## Mean_precipita 6.753363
## Mean_temp      1.048184
```

```
#Relative abundance of D. villosus model (GLM)
## Relative abundance model (i.e. Dominance).
m3 <- glm(Proportion ~  #Relative abundance
            Elevation+ #Elevation
            Precipitation  + #Precipitation (per year)
            Temperature    + #Temperature (per year)
            Mean_temp+ #Avg temperature
            DistanceKm+ #Distance to the next barrier
            slope+  #Slope of stream
            Tmin          +  #Minimum temperature (each year)
            Avg_Tmax      + #Avg Maximum temperature
            Tmax          + #Maximum temperature (each year)
            year          + #Sampling years
            Biogeo+ #Biogeographical region
            Ecosystem,  #Ecosystem type
          data = df,family = "quasibinomial",  na.action="na.fail")

summary(m3)
```

```
##
## Call:
## glm(formula = Proportion ~ Elevation + Precipitation + Temperature +
##     Mean_temp + DistanceKm + slope + Tmin + Avg_Tmax + Tmax +
##     year + Biogeo + Ecosystem, family = "quasibinomial", data = df,
##     na.action = "na.fail")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.6756  -0.3069  -0.1310   0.1126   1.7835
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -5.073e+01  2.277e+01  -2.228 0.026171 *
## Elevation              4.168e-03  1.601e-03   2.603 0.009437 **
## Precipitation         -1.851e-01  1.165e-01  -1.589 0.112488
## Temperature           -4.201e-03  1.014e-02  -0.414 0.678810
## Mean_temp              1.130e-03  6.713e-03   0.168 0.866338
## DistanceKm             1.074e-03  3.615e-04   2.970 0.003079 **
## slope                  1.342e+00  2.102e+00   0.638 0.523527
```

```
## Tmin                 -1.184e-01  6.547e-02  -1.808 0.070975 .
## Avg_Tmax             -5.165e-04  2.062e-03  -0.250 0.802283
## Tmax                  1.558e-04  6.562e-04   0.237 0.812437
## year                  2.389e-02  1.138e-02   2.100 0.036110 *
## BiogeoAtlantic        1.195e+00  4.227e-01   2.827 0.004834 **
## BiogeoContinental     1.645e+00  3.438e-01   4.786 2.07e-06 ***
## BiogeoMediterranean  2.450e+00  4.259e-01   5.751 1.30e-08 ***
## BiogeoPannonian       1.090e+00  3.979e-01   2.739 0.006319 **
## Ecosystemminor       -4.295e-01  1.213e-01  -3.541 0.000424 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasibinomial family taken to be 0.1617545)
##
##     Null deviance: 118.32  on 743  degrees of freedom
## Residual deviance: 101.08  on 728  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 5
```