Trait-based responses of annual crops to elevated CO2 and water limitation

Appendix 1: `R` script

Setup

```r
pacman::p_load(dplyr)
str(crop.responses)
```

```
## 'data.frame':    1609 obs. of  8 variables:
##  $ CO2.ppm       : Factor w/ 3 levels "450","575","700": 1 1 1 1 1 1 1 1 1 1 ...
##  $ water.exposure: chr  "Full" "Full" "Full" "Full" ...
##  $ photo.path    : chr  "C3" "C3" "C3" "C3" ...
##  $ funct.group   : chr  "cool-season grass" "cool-season grass" "cool-season grass" "cool-season grass"
##  $ species       : chr  "barley" "barley" "barley" "barley" ...
##  $ pot           : int  21 22 23 24 25 26 27 28 29 30 ...
##  $ value         : num  0.785 0.695 0.572 0.483 0.761 ...
##  $ response      : Factor w/ 4 levels "os1","cfr","LA",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Model selection tables

```r
crop.tabs <- ddply(crop.responses,
                   .(species, response),
                   .fun=AICc.tabler)

# Function to build model selection tables by species
AICc.tabler <- function(dat) {
  require(AICcmodavg)
  resp = unique(dat$response)
  y = dat$value
  CO2 = dat$CO2.ppm
  H2O = dat$water.exposure

  if (resp == "os1") {
    require(betareg)
    m.null <- betareg(y ~ 1)
    m.CO2 <- betareg(y ~ CO2)
    m.H2O <- betareg(y ~ H2O)
    m.add <- betareg(y ~ CO2 + H2O)
    m.int <- betareg(y ~ CO2 * H2O)
  }
  else {
    m.null <- glm(y ~ 1, family=Gamma(link = "identity"))
    m.CO2 <- glm(y ~ CO2, family=Gamma(link = "identity"))
    m.H2O <- glm(y ~ H2O, family=Gamma(link = "identity"))
    m.add <- glm(y ~ CO2+H2O, family=Gamma(link="identity"))
    m.int <- glm(y ~ CO2*H2O, family=Gamma(link="identity"))
    dist.params <- MASS::fitdistr(y, "Gamma")
    phi <- 1/dist.params$estimate[[1]]
  }
```

```r
resp.mod.names <- c("m.null","m.CO2","m.H2O",
                    "m.add","m.int")

resp.mods <- list( )

for(i in 1:length(resp.mod.names)) {
  resp.mods[[i]] <- get(resp.mod.names[i]) }

aic.tab <- as.data.frame(aictab(cand.set = resp.mods,
                                modnames = resp.mod.names))
return(aic.tab)
}
```

Model averaging

```r
crop.CIs <- ddply(crop.responses,
                  species, response),
                  .fun=CI.modav)

# Function to perform model averaging, estimate parameters, by species

CI.modav <- function(dat) {
  require(AICcmodavg)
  resp = unique(dat$response)
  y = dat$value
  CO2 = dat$CO2.ppm
  H2O = dat$water.exposure

  if (resp == "os1") {
      require(betareg)
      m.null <- betareg(y ~ 1)
      m.CO2 <- betareg(y ~ CO2)
      m.H2O <- betareg(y ~ H2O)
      m.add <- betareg(y ~ CO2 + H2O)
    }
  else {
      m.null <- glm(y ~ 1, family=Gamma(link = "identity"))
      m.CO2 <- glm(y ~ CO2, family=Gamma(link = "identity"))
      m.H2O <- glm(y ~ H2O, family=Gamma(link = "identity"))
      m.add <- glm(y ~ CO2 + H2O, family=Gamma(link = "identity"))
      dist.params <- MASS::fitdistr(y, "Gamma")
      phi <- 1/dist.params$estimate[[1]]
       }

  resp.mod.names <- c("m.null","m.CO2" ,"m.H2O", "m.add")
  resp.mods <- list( )

  for(i in 1:length(resp.mod.names)) {
    resp.mods[[i]] <- get(resp.mod.names[i]) }

  terms <- c("(Intercept)", "CO2575", "CO2700", "H2OLimited")
  av.params <- as.data.frame(array(NA,c(length(terms),4)))
  colnames(av.params)<-c("term","estimate","ciL","ciU")
  for(i in 1:length(terms)) {
```

```r
    if (resp == "os1") {
      av <- modavg(parm = paste(terms[i]),
                    cand.set = resp.mods,
                    modnames = resp.mod.names)}
    else {
      av <- modavg(parm = paste(terms[i]),
                          cand.set = resp.mods,
                          modnames = resp.mod.names,
                          gamdisp = phi)
    }
    av.params[i,1] <- terms[i]
    av.params[i,2] <- round(av$Mod.avg.beta, 2)
    av.params[i,3] <- round(av$Lower.CL, 3)
    av.params[i,4] <- round(av$Upper.CL, 3) }

  return(av.params)
}
```