Resources selection model application for wildlife monitoring in Gran Sabana, Venezuela

José R. Ferrer-Paris Izabela Stachowicz Ada Y. Sánchez-Mercado

Set up

Packages

First load the packages that we need for the analysis:

```
require(dplyr)
require(tidyr)
require(raster)
require(chron)
require(detect)
require(vegan)
require(stringr)
require(magrittr)
require(ggplot2)
require(forcats)
require(tidyr)
here::i_am("doc/svocc-analysis-GS.Rmd")
```

Spatial data

Read the spatial data stored in a Rdata file:

```
GIS.data <- sprintf(here::here("Rdata","GIS.rda"))
load(GIS.data)
```

Prepare a data frame for prediction:

```
dts <- data.frame(bsq=values(vbsq),
                  dcon=values(dist.conucos),
                  dcom=values(dist.comunidades),
                  frs=values(dist.frs),
                  dbsq=values(dist.dbsq),
                  dcaz1=values(dist.caza1),
                  dcaz2=values(dist.caza2),
                  grid=values(rgrd)) %>%
  group_by(grid) %>%
  summarise(bsq=mean(bsq),
            dbsq=mean(dbsq),
            dcon=mean(dcon),
            dcom=mean(dcom),
            dpob=mean(min(dcon,dcom)),
            dhum=mean(min(dcon,dcom,dcaz1,dcaz2)),
            dcaz1=mean(dcaz1),
```

```
            dcaz2=mean(dcaz2),
            dcaz=mean(min(dcaz1,dcaz2)),
            frs=mean(frs))

mi.data <- dts %>% dplyr::select(bsq:frs) %>% decostand(.,"stand")
```

Model exploration

This will read brief summary of all models explored for all species:

```
input.dir <- here::here("Rdata","svocc","explore")
allmodels <- tibble()
nullmodels <- tibble()

for (k in dir(input.dir,pattern="*rda",full.names=T)) {
  objs <- (load(k))
  spname <- basename(k) %>% str_replace("\\.rda","")
  allmodels %<>% bind_rows(params %>% mutate(sp=spname))
  nullmodels %<>% bind_rows(nulls %>% mutate(sp=spname))
  rm(objs)
}
```

This will show us a summary by applying following steps:

- `filter` models that:
  - passed two test of model converge (looking at size of esimates and standard errors),
  - have AIC lower than the corresponding null model,
  - use the complementary log-log link function
- use `mutate` to calculate delta AIC
- `group by` species and the combination of method/sampling (indicated by `k`)
- `summarise` the number of models explored and the minimum delta AIC for each species/method/sampling combination
- `pivot wider` to compare the number of model explored for each species

We are focusing on the models with complementary log-log link function, as it seems to fit the data better for more species.

High values indicate that the algorithm was successful in exploring different combinations of variables, low number means that most of the models explored failed to converge or performed worse than the corresponding null model.

```
allmodels %>%
  filter(test1,
         test2==5,
         AIC<AICnull,
         linkfuns=="cloglog") %>%
  mutate(deltaAIC=AIC-AICnull) %>%
  group_by(sp,k) %>%
  summarise(n=n(),deltaAIC=min(deltaAIC), .groups = "keep") %>%
  pivot_wider(id_cols="sp",names_from=c(k),values_from=n) %>%
  print(n=100)
```

```
## # A tibble: 25 x 5
## # Groups:   sp [25]
##    sp               `2`   `3`   `4`   `1`
##    <chr>          <int> <int> <int> <int>
```

```
##   1 C.olivaceus        1     4     5    NA
##   2 C.paca             7     8     1    24
##   3 C.thous           23     3     2    57
##   4 D.imperfecta      10     6     3    12
##   5 D.kappleri        17    19     2    41
##   6 D.leporina         6    16     3    19
##   7 D.marsupialis      3     2     3     2
##   8 D.novemcinctus     3    25    12     7
##   9 E.barbara         NA     2    NA    31
##  10 H.hydrochaeris     8     2     1    22
##  11 L.pardalis        10    30    10    34
##  12 L.rufaxilla       NA    NA    NA     1
##  13 L.wiedii           2     1     1     2
##  14 M.americana        3    23     5    13
##  15 M.gouazoubira     30    NA     2    44
##  16 M.tridactyla      17    NA     2    19
##  17 O.virginianus      4     3     1    11
##  18 P.concolor         5     7     5    13
##  19 P.maximus          1     2     3     6
##  20 P.onca            NA    15    NA    43
##  21 P.tajacu           8    NA    NA     5
##  22 T.major            2     6     4     3
##  23 T.pecari          NA     4     2    NA
##  24 T.terrestris       9    NA    NA    19
##  25 T.tetradactyla    17    12     6    10
```

Summarise best models

We use this function to summarise AIC and AICc for each model:

```
modelsum <- function(x) {
  ll <-logLik(x)
  fml <- paste(as.character(x$formula$full)[c(2,1,3)],collapse=" ")
  tbl <- tibble(
    formula=fml,
    loglik=as.vector(ll),
    npar=attr(ll,"df"),
    nobs=attr(ll,"nobs"),
    df.null=x$df.null,
    df.residual=df.residual(x),
    AIC=AIC(x),
    AUC=AUC(x)
  )
  return(tbl)
}
```

Now we read the selected model (lowest AICc) for each species. If the model with spatial covariates did not perform better than the null model (AICc higher than the AICc of the null model) then we include only the null model in this table.

```
modeltab <- tibble()
for (k in 1:4) {

  input.dir <- here::here("Rdata","svocc",paste0("best-",k))
  for (j in dir(input.dir,full.names=T)) {
```

```r
    mdls <- (load(j))
    sp = basename(j) %>% str_replace("\\.rda","") %>%
      str_replace("\\.",". ")
    if ("fit.boot" %in% mdls) {
      modeltab %<>%
        bind_rows({
          modelsum(fit.boot) %>%
            mutate(species=sp,ss=k,type="best")}) %>%
        bind_rows({
          modelsum(fit.null) %>%
            mutate(species=sp,ss=k,type="null")})

    } else {
      modeltab %<>%
        bind_rows({
          modelsum(fit.null) %>%
            mutate(species=sp,ss=k,type="null")})
    }
  }
}
```

We will create a table summarising the results for all species:

```r
modeltab %>%
  mutate(
    AICc=AIC+((2*npar^2 + 2*npar)/(nobs-npar-1)),
    method=if_else(ss %in% 1:2,"camera + observations","camera only"),
    sampling=if_else(ss %% 2 == 0,"Warapata","Warapata + Kavanayen")
    ) %>%
  pivot_wider(id_cols = c(species,method,sampling),
              names_from = type,
              values_from = c(formula, loglik, npar, nobs, AICc)) %>%
  dplyr::select(species, method, sampling,
                ends_with("null"), ends_with("best")) %>%
  mutate(deltaAICc = AICc_null - AICc_best) -> tableA1
```

This table shows how many species had a fitted model for each of the four combinations of data and sampling. The first column is the number of species with good fit of spatial covariates when compared with the null model (delta AICc is greater than 2), the second column is the number of species with a spatial covariates models comparable to the null model (delta AICc is less or equal 2), and the third column is the number of species with only a null model (delta AICc less than zero or model did not converge).

```r
with(tableA1,
  table(paste(method,sampling),
        deltaAICc<=2,useNA = "always"))
```

```
##
##                                              FALSE TRUE <NA>
##   camera + observations Warapata                19    2    7
##   camera + observations Warapata + Kavanayen    23    0    5
##   camera only Warapata                          19    1    8
##   camera only Warapata + Kavanayen              20    0    8
##   <NA>                                           0    0    0
```

The models with all available data has more species in the first column than all the other alternatives.

Compare estimated coefficients

Now, for the best models we will extract the coefficients and calculate the 95% confidence intervals.

```
mcoefs <- tibble()
for (k in 1:4) {
  input.dir <- sprintf("../Rdata/svocc/best-%s",k)
  for (j in dir(input.dir,full.names=T)) {
    mdls <- (load(j))
    sp = basename(j) %>% str_replace("\\.rda","") %>%
      str_replace("\\.",". ")
    if ("fit.boot" %in% mdls) {
      cfs <- coefficients(fit.boot)
      cis <- confint(fit.boot)
      nms <- str_split_fixed(names(cfs), "_", n=2)
      mcoefs %<>% bind_rows(tibble(species=sp, ss=k,component=nms[,1],variable=nms[,2],mu=cfs,lower=cis
    }
  }
}
```

```
## Warning in sqrt(diag(vcov(object, model, type))): NaNs produced
```

```
mcoefs %<>% mutate(method=if_else(ss %in% 1:2,"camera + observations","camera only"),
                   sampling=if_else(ss %% 2 == 0,"Warapata","Warapata + Kavanayen"))
```

We will use colours to highlight significant results:

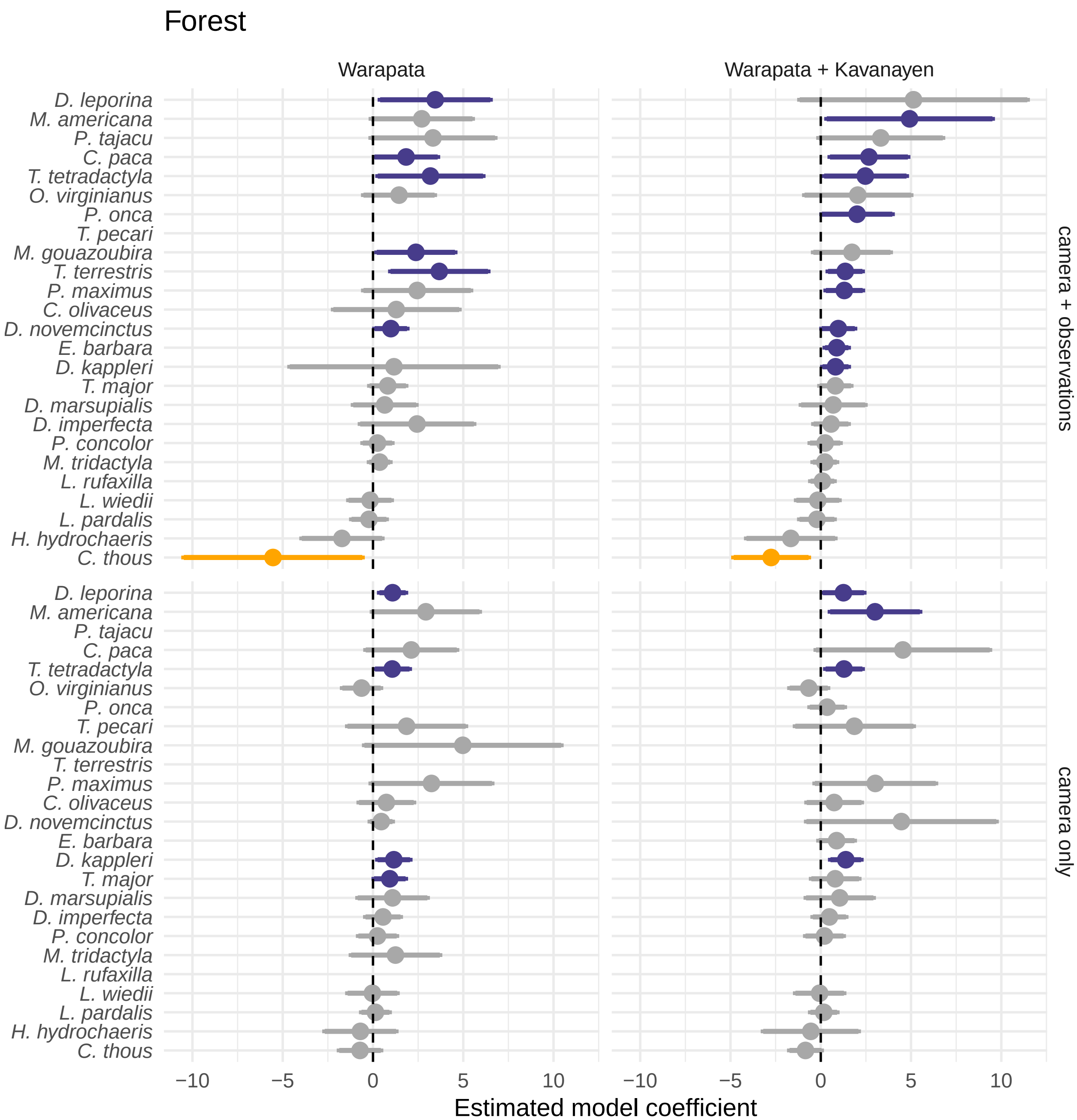
```
cols <- c("sig. pos." = "slateblue4", "sig. neg." = "orange", "not significant" = "grey66")
```

And we create this function to layout the figures for each variable:

```
plotModCoef <- function(varcode,varname) {
  dat <- {mcoefs %>% filter(component=="sta",variable==varcode) %>% mutate(
    species = fct_reorder(species, mu,.fun=first),
    response=case_when(
      lower>0 ~ "sig. pos.",
      upper<0 ~ "sig. neg.",
      TRUE ~ "not significant",
    )) %>% arrange(mu)}
  p <- ggplot(data=dat) +
    geom_point(aes(x=mu,y=species,colour=response),cex=3) +
    geom_errorbar(aes(y=species,xmin=lower,xmax=upper,colour=response), width = 0.2,lwd=1.05) + facet_g
    scale_colour_manual(values = cols) +
    theme_minimal() + theme(legend.position="none", axis.text.y=element_text(face = "italic")) + geom_v
  return(p)
}
```

We start with the covariate forest cover, which is include in all models:

```
plotModCoef("bsq","Forest")
```



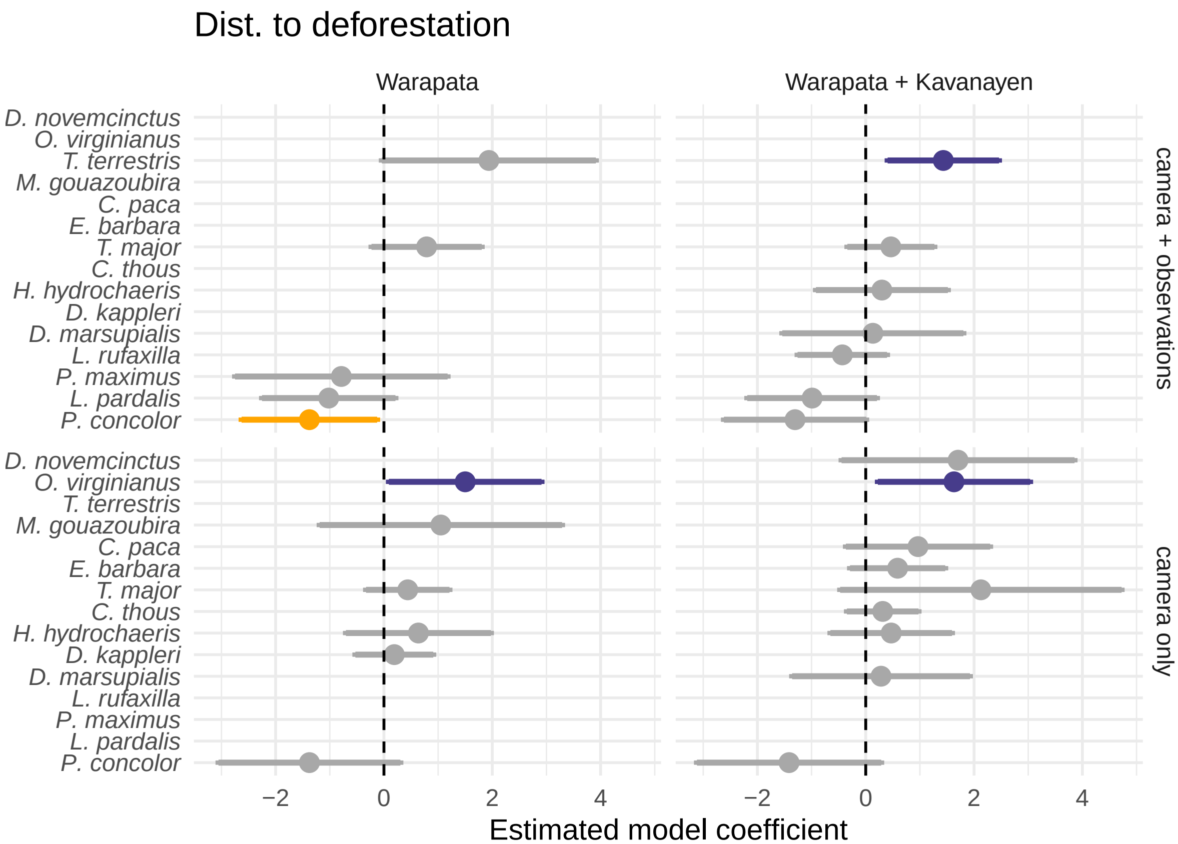
The model with all data estimates that seven species have significant positive relationship with forest cover and one with sig. negative (C.thous).

The model based on all evidence from one region agrees in most species, but models based only on camera trap data only estimate sig. effects in one to four species.

T. tetradactyla appears to have a sig. positive relationship based on models fitted with subsets, but this is not picked up by the full model.

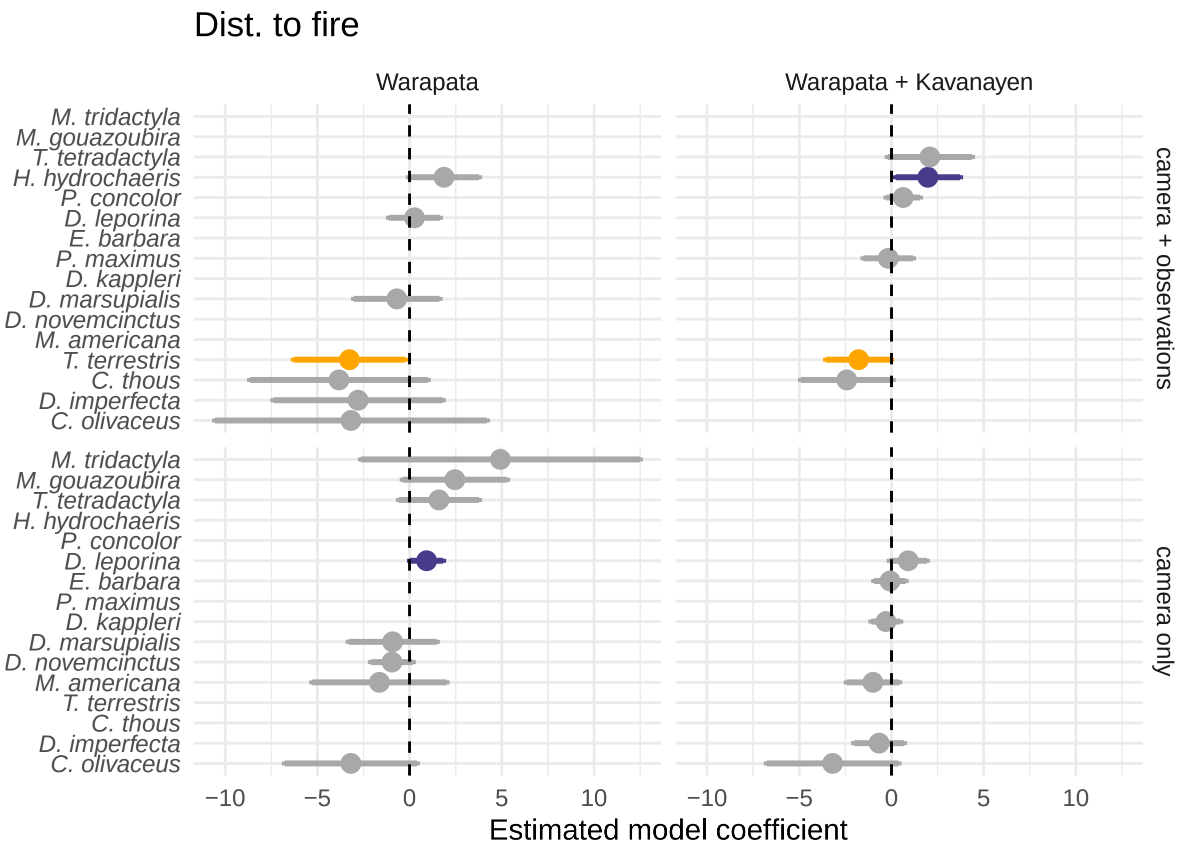
Now we do the same for all the covariates that indicate drivers of forest conversion or habitat change. First distance to deforestation events:

```
plotModCoef("dbsq","Dist. to deforestation")
```



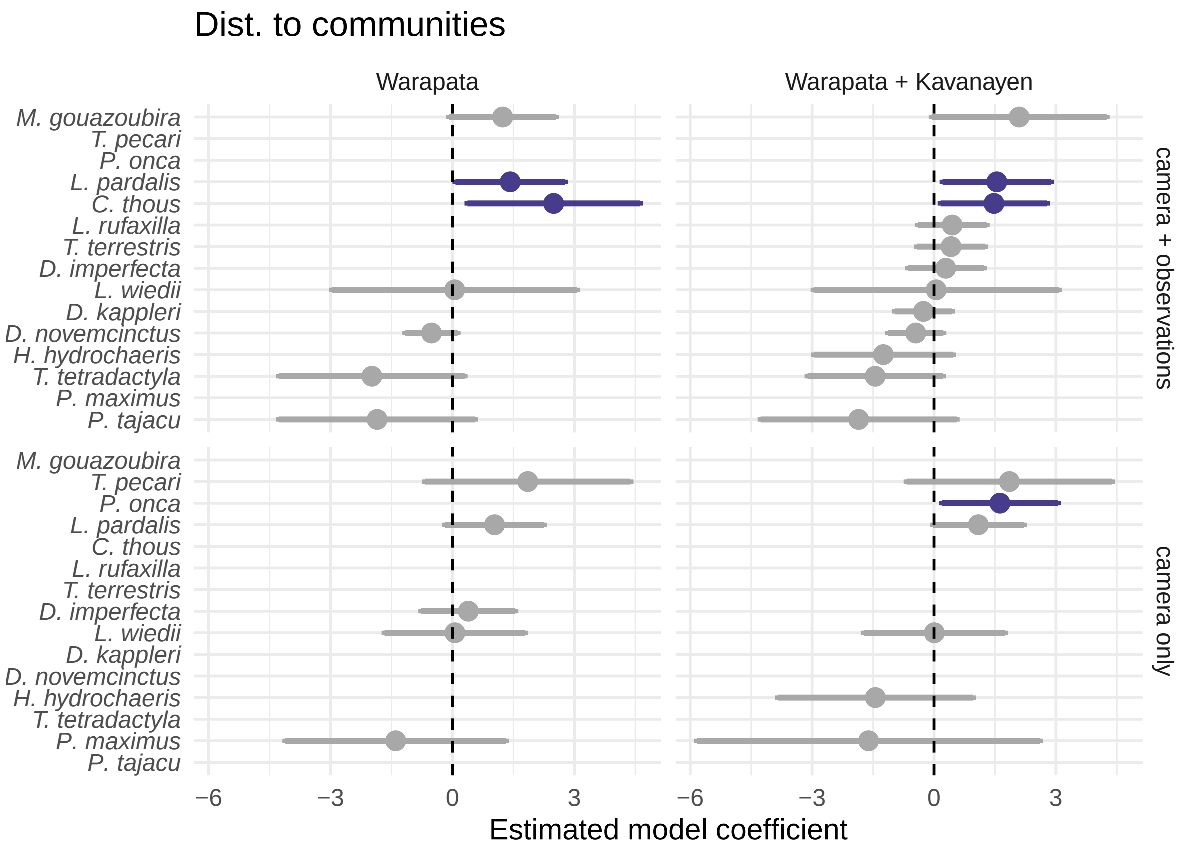
Now distance to recent fires:

```
plotModCoef("frs","Dist. to fire")
```



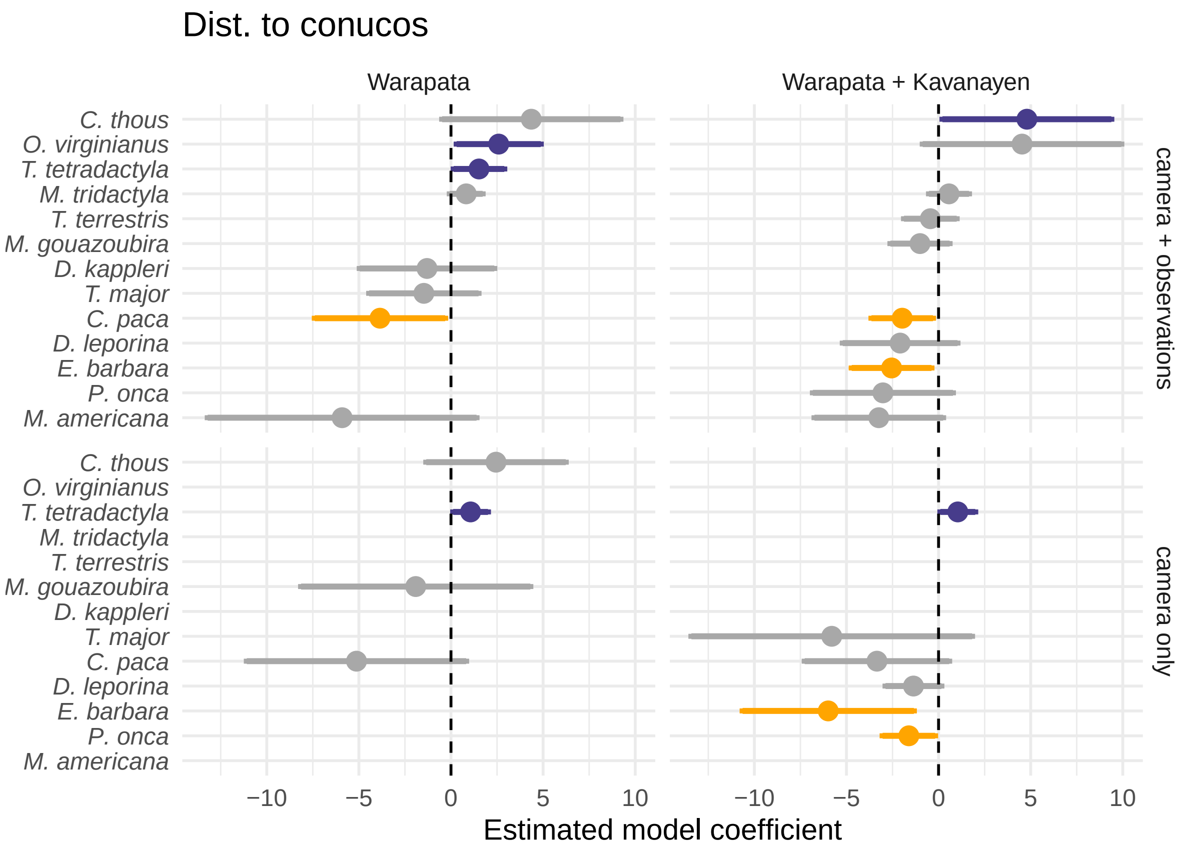
Distance to communities:

```
plotModCoef("dcom","Dist. to communities")
```



Distance to conucos:

```
plotModCoef("dcon","Dist. to conucos")
```



Spatial prediction of resource use

Finally we use the models to predict resource use across the spatial grids, we have 10 blocks of 25 grid cells each, so the maximum number of cells is 250.

The standard error of the prediction is calculated from the non-parametric bootstrap of the model. We use a simple formula to propagate the errors of the individual predictions to the sum of all predictions.

```
preds <- tibble()
for (k in 1:4) {
  input.dir <- sprintf("../Rdata/svocc/best-%s",k)
  for (j in dir(input.dir,full.names=T)) {
    mdls <- (load(j))
    sp = basename(j) %>% str_replace("\\.rda","") %>%
      str_replace("\\.",". ")
    if ("fit.boot" %in% mdls) {
      #
      prd0 <- predict(fit.boot,mi.data,type="response",se.fit=T)
      preds %<>% bind_rows(tibble(
        species=sp, ss=k, model="full",
        best=sum(prd0$fit),
        se=sqrt(sum(prd0$se.fit^2))))
    } else {
      prd0 <- predict(fit.null,
                   data.frame(bloque=rep(fit.null$levels$bloque,25)),
                     type="response",se.fit=T)
      preds %<>% bind_rows(tibble(
```

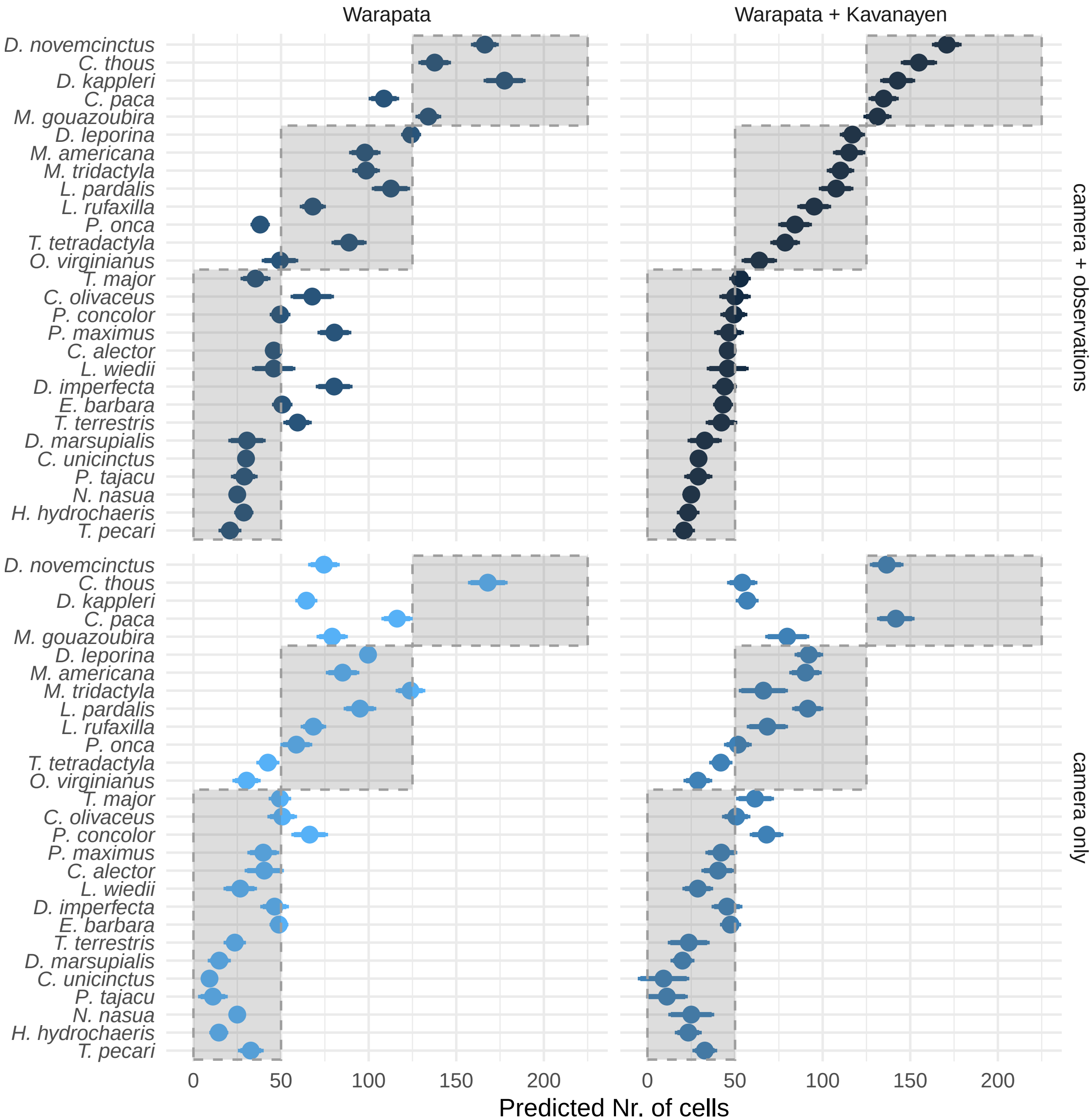
```
        species=sp, ss=k, model="null",
        best=sum(prd0$fit),
        se=sqrt(sum(prd0$se.fit^2))))   # propagate errors
    }

  }
}

preds %<>% mutate(
  method=if_else(ss %in% 1:2,"camera + observations","camera only"),
  sampling=if_else(ss %% 2 == 0,"Warapata","Warapata + Kavanayen"))
```

Now we compare the predicitions.

```
dat <- {preds %>%
    mutate(species = fct_reorder(species, best,.fun=first)) %>% arrange(best)}
grps = tibble(xmin=c(0),xmax=c(50),ymin=c(1),ymax=c(13))
ggplot(data=dat) +
  geom_point(aes(x=best,y=species,colour=ss),cex=3) +
  geom_errorbar(aes(y=species,xmin=best-(2*se),xmax=best+(2*se),colour=ss), width = 0.2,lwd=1.05) + face
  theme_minimal() + theme(legend.position="none", axis.text.y=element_text(face = "italic")) +
  annotate("rect", xmin = 0, xmax = 50, ymin = .5, ymax = 15.5,
  alpha = .2, lty=2, col="grey62") +
  annotate("rect", xmin = 50, xmax = 125, ymin = 15.5, ymax = 23.5,
  alpha = .2, lty=2, col="grey62") +
  annotate("rect", xmin = 125, xmax = 225, ymin = 23.5, ymax = 28.5,
  alpha = .2, lty=2, col="grey62") +
  ylab("") + xlab("Predicted Nr. of cells")
```



Based on the model with all the available data in the top-right corner, we have three groups of species:

- five widespread species predicted in more than 150 cells
- eight species with intermediate predictions (50 to 120 cells)
- the rest are restricted to less than 50

The ranking of the species is different when we use different subsets of the data, but most species remain in these three groups, with a few exceptions:

- For the models based on all evidence from the Warapata region, the distribution of C. thous, E. barbara, T. terrestris, O. virginianus and P. maximus is higher, and the distribution of P. concolor is lower.
- Models that only use cameras appear to underestimate the distribution of the more widespread species,