AI for the Environment: from AI to Ecological Models

Week 9: afternoon workshop

Rory Gibb & Ella Browning

07/03/2023

Afternoon workshop: Drivers of species occurrence across the Masai Mara using generalised linear (mixed) models

Today we're exploring and analysing some camera trap data from the Masai Mara collected as part of the Biome Health project - see the lecture slides for a general summary of the data and the project, and see this morning's workshop for an introduction to spatial data processing and GIS in R.

The goal of this afternoon's session is to bring together the spatial data collation, processing and exploration from this morning's session, with a particular research question, and explore fitting some generalised linear and mixed effects models to investigate the drivers and distribution of our study species (the Cape hare). There will be code snippets with short exercises interspersed, along with some larger extension exercises at the end if you have time. All the data and environmental layers you'll need for the workshop are in the GitHub, in the "9_AIToEcologicalModels" folder. Please download the whole folder and set this as your working directory, then all the materials you will need are contained within the "data" subfolder.

```
# dependencies
library(dplyr); library(magrittr); library(terra); library(rgdal); library(sf); library(ggplot2)
library(lme4); library(rstudioapi); library(MetBrewer); library(tibble)

# automatically set file path
# (or if this doesn't work,
# manually set your working directory to the folder "9_AItoEcologicalModels")
PATH = dirname(rstudioapi::getSourceEditorContext()$path)
setwd(PATH)
```

Defining our research question

Let's start with a broad question: what is the relationship between level of anthropogenic pressure and spatial occupancy of our focal species?

We can define anthropogenic pressure in many ways, but here, let's focus on agricultural land use and livestock pressure. We might also need to account for other factors that may covary with our drivers of interest and also affect hare presence; here, we'll look at habitat type (proportion of closed habitat) and distance to the nearest water body.

Building a full dataframe of environmental covariates for modelling

This morning's session provided an introduction to the process of combining our ecological survey data with other socio-environmental data from spatial sources. From there, it is possible to combine all those

operations into a pipeline to build a full covariates dataframe, that we can then use for modelling. *In the solutions*, you can see the full code block that we used to produce this dataframe from the raw data (i.e. the sampling locations), but for this worksheet, we will just read in this full processed dataframe for use in our analyses further below.

Combining environmental covariates and species detections to create a modelling dataframe

When we store and work with data to provide to most statistical models in R (and other software), we work with long-form dataframes where each row is a single observation. For these camera trap data, "one observation" is one day when the trap was operational and sampling (i.e. 1 observation per day). In the above code block visible in the solutions, we created this dataframe, with a total n=5792 observation days. From extracting environmental infromation from associated rasters, we have several covariates we can consider as ecologically relevant: proportion of closed/semi-closed habitat or agriculture land use within a 250m radius; distance to the nearest water source; and daily maximum temperature. We don't yet have a livestock grazing pressure covariate - let's come back to that

As we discussed in the lecture, this database has some hierarchical and nested structure within it, which we can look at using the summary and table functions. We can also explore the distributions of our covariates of interest, and whether any of these are correlated with each other.

Firstly, we'll read in the data including covariates for each camera trap site and the locations information.

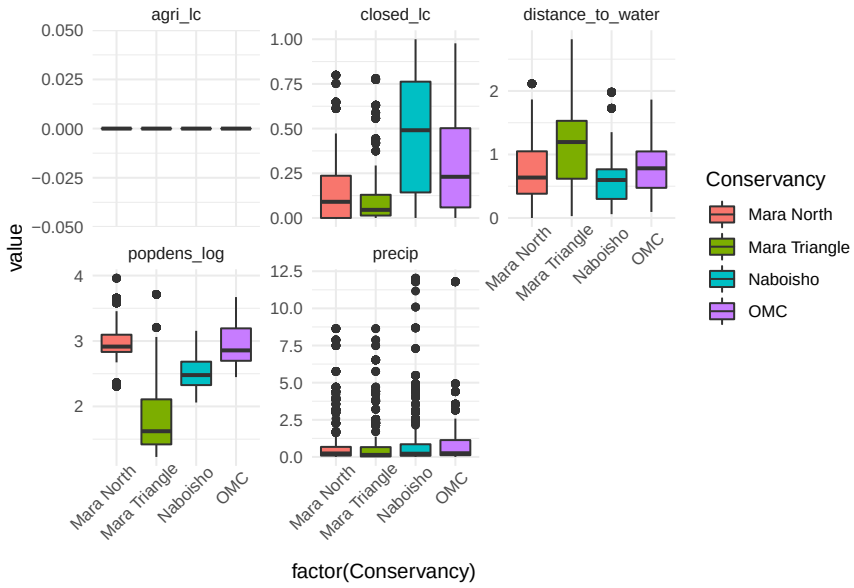
```
# read in covariates per camera trap
dd = read.csv("./data/kenya/data_processed/BH_CTsite_covariates.csv")
# specify date column as date object
dd$Date = as.Date(dd$Date, format="%Y-%m-%d")

# locations sf object
locs = read.csv("./data/kenya/survey/bh_camera_locations.csv") %>%
  sf::st_as_sf(coords = c("Longitude", "Latitude"), crs = 4326) %>%
  sf::st_transform(locs, crs = "+proj=utm +zone=36 +south +datum=WGS84 +units=m +no_defs") %>%
  dplyr::filter(CT_site != "MT34")

# add coordinates columns for XY locations
locs = cbind(locs, sf::st_coordinates(locs))
```

Exercise 1

- Explore these data by plotting histograms and scatter plots of our covariates of interest. Try calling *table()* on categorical columns to understand the distribution of data between types or *range()* for continuous data. You can also use *ggplot()* to plot boxplots of covariates of interest across different conservancies. What do you notice? Will all of these covariates be suitable to include in a model?

```
# boxplots of how our covariates are distributed across conservancies
dd %>%
  tidyr::pivot_longer(cols = c("closed_lc", "agri_lc",
                               "distance_to_water", "popdens_log", "precip"),
                      names_to="covariate", values_to="value") %>%
  ggplot() +
  geom_boxplot(aes(factor(Conservancy), value, group=Conservancy, fill=Conservancy)) +
  theme_minimal() +
  facet_wrap(~covariate, scales="free_y") +
```

```
  # sets the x-axis text to print at an angle and not overlapping the plot
  theme(axis.text.x = element_text(angle = 45, hjust = 0.8))
```



Now we have a dataframe of environmental covariates and a general understanding of how these are distributed across our study area. The next thing we need to do is incorporate the actual survey data from the camera trap images. If you recall, earlier today we summarised those images at the site-level so we could plot them over space, as proportion of days in which the species was detected. This afternoon, because we're considering each day of sampling as an observation, we need to identify whether our species (*Lepus capensis*, the Cape hare) was detected at each site and day.

```
# our species of interest
sp = "hare"

# our camera trap data (n=601 observations of our study species)
ctd = read.csv("./data/kenya/survey/bh_camera_images_mara.csv") %>%
  dplyr::filter(CT_site %in% locs$CT_site) %>% # ensure sites are in CT location data
  dplyr::mutate(Date = as.Date(Date, format="%Y-%m-%d")) %>%
  dplyr::filter(Species == sp)

# summarise this by site and day
# gives a value of 1 for each day where the species was detected (n=417 days)
ctd_daily = ctd %>%
  dplyr::group_by(CT_site, Date) %>%
  dplyr::summarise(
    Detected = 1,
    Species = "Cape hare"
  )

# left join to our environmental data
# (remember, left_join auto-fills non-matches with NA
# so we replace these with 0, ie. not detected)
dd = dd %>%
  dplyr::left_join(ctd_daily) %>%
  dplyr::mutate(Detected = replace(Detected, is.na(Detected), 0),
```

```
                Species = replace(Species, is.na(Species), "Cape hare"))

# tabulate overall and by conservancy - 417 positive detections, 5192 non-detections
# (8% non-zero - not terrible but definitely quite zero-inflated!)
table(dd$Detected)
table(dd$Conservancy, dd$Detected)
```

Notably, the agriculture land use covariate doesn't contain any useful comparative information for our questions, as it doesn't vary between camera sites. There is agricultural land use around the margins of the study area (you can look at this by calling plot(hab == 6), but because our camera trap network was placed in the conservancies and protected areas, it covers an area of land that is hardly cropped).

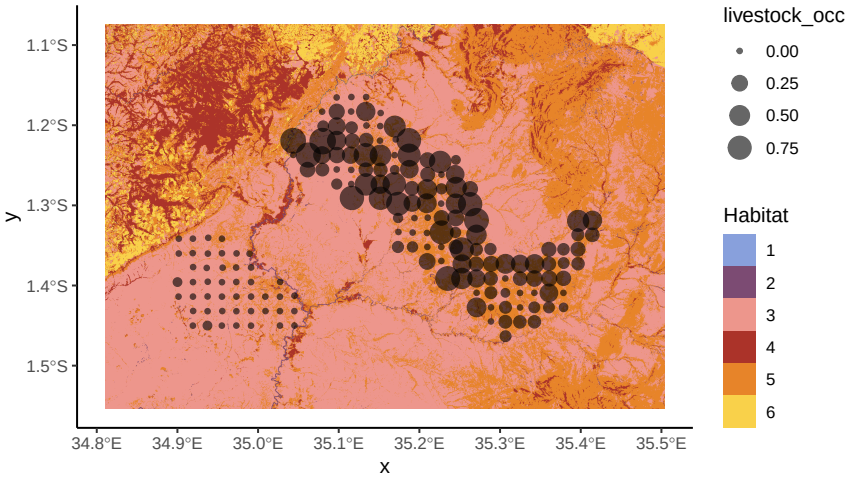
However, one of the major anthropogenic factors in this area is activity and grazing livestock (cattle, sheep and goats). Grazing has potential effects on the plant community composition and consequently resource availability as well as potentially reducing wildlife activity in the areas where livestock are active. So quantifying livestock pressure would be an important dimension of anthropogenic activity to account for.

Fortunately, camera traps also capture many images of livestock, as well as wildlife, so we can return to the tagged image data to calculate a proxy for livestock pressure. Let's define this at the site-level, as the proportion of surveyed days with livestock detected (a measure of intensity of use by livestock). We could probably do a more rigorous job of defining this metric, but this is fine for this workshop.

Exercise 2

- Read in the camera trap image data again and subset to livestock (*hint* you can modify the code above). For each camera trap site calculate the proportion of sampled days when livestock were detected. *Remember* from this morning that **not all** locations and days will necessarily have livestock images, so you will need to calculate the total number of days sampled per camera trap from "bh_camera_samplingeffort.csv".
- Combine the livestock data with the main dataset (*dd*), creating a column called "livestock_occ". Explore how livestock are distributed across conservancies and over space.
- What do you notice about the spatial pattern of anthropogenic factors, as well as environmental factors - how are these different across the study area? *Remember* that a foundational aspect of statistical models is that they assume errors are independent, conditional on the model. There definitely seems to be some spatial structure here that we might want to take into account later.

(The code for this is contained within the solutions if you get stuck!)



Fitting a logistic (binomial) regression model to estimate probability of occupancy

Let's revisit our research question and formulate some specific expectations, focusing on the intersection of livestock activity and habitat. We know that hares are commonly found in grassland and pastoral ecosystems, so if habitat suitability for hares is shaped by pastoral activity we might expect a positive relationship with livestock activity. Alternatively, it is possible that areas with very high levels of livestock activity are too frequently disturbed to provide amenable habitat, in which case we might expect hare occurrence to decline where livestock use intensity is highest. So here we have two alternative, plausible hypotheses. Let's try and answer this question using some generalised linear models.

- What type of data are our response data?

Binomial (1/0) response data (the species was detected or not detected). We can model these using a logistic regression model with a binomial likelihood, where we estimate the effect of covariates $X_1 : X_n$ on the log odds of occurrence.

The model would be formulated as:

$Y_i \sim Bernoulli(p_i)$

$log(p_i/(1-p_i)) = \beta_0 + X\beta_i$

where β is a vector of slope parameters, and X is a matrix of covariates, 1 per parameter.

Our covariates are notably all on quite different scales of magnitude to each other. Recall that a slope parameter describes the change in Y for a single unit change in X. This means that the slope parameter sizes for different covariates will mean different things (e.g. 1 degree of temperature versus moving from 0 to 1 in livestock occupancy). A way to deal with this is to centre and scale covariates to make the estimates comparable - subtract the mean and divide by the SD. This way, slope parameters always describe the change in Y for 1 standard deviation change in X, regardless of what units X was measured in.

```
# visualise raw relationship between binary outcome and covariates using boxplots
dd %>%
  tidyr::pivot_longer(cols = c("closed_lc", "livestock_occ", "distance_to_water",
```

```
                                "popdens_log", "precip"), names_to="covariate",
                      values_to="value") %>%
  ggplot() +
  geom_jitter(aes(factor(Detected), value, group=factor(Detected)), alpha=0.1,
              width=0.5, size=0.2, color="grey70") +
  geom_boxplot(aes(factor(Detected), value, group=factor(Detected)),
               outlier.shape = NULL, color="coral2") +
  theme_minimal() +
  facet_wrap(~covariate, scales="free_y")
```

```
# scale our linear covariates for comparability
dd[ , c("closed_lc", "livestock_occ",
        "distance_to_water",
        "popdens_log", "precip") ] = apply(dd[ , c("closed_lc", "livestock_occ",
                                                   "distance_to_water", "popdens_log",
                                                   "precip") ], 2, scale)
```

We can use the glm function to fit a generalised linear model, defining this formula as "Y ~ 1 + covariate + covariate + . . . ", where 1 refers to the intercept, with a binomial likelihood.

```
# Fit a logistic regression model with an intercept and a slope for livestock
# occurrence (our key predictor of interest)
m1 = glm(Detected ~ 1 + livestock_occ,
         family=binomial(link="logit"),
         data=dd)

# call summary on the model ; this shows an summary of the model residuals,
# and a table of coefficients (fitted model parameters, on the log odds scale)
# a strongly positive slope of livestock occurrence with low uncertainty (a low SE)!
summary(m1)
```

Exercise 3

- What other factors might be influencing hare occurrence and also covary with livestock occurrence? It may be important to include these too, in case they explain some of this relationship (we'll explore this more next week). Fit another model called *m2*, including closed habitat and distance to water as covariates. Call *summary()* and take a look at the model.

Now let's compare the two models. The code below extracts the fitted paramter estimates and calculates the 95% confidence intervals and visuliases them. Use this to plot the parameter estimates for *m1* and *m2*. What do you notice?

```
# Function to plot coefficients and confience intervals.
# The intercept is often at a different scale to the slope parameters
# so creates a separate sub-plot for the intercept).

plotFixedEffects = function(model){

  plot = coef(summary(model)) %>% # extract parameters table from fitted model
    as.data.frame() %>% # convert to df
    tibble::rownames_to_column(var="param") %>% # make a column called "param" from the row names
```

```
    # classify param as either Intercept or Slope
    dplyr::mutate(param_type = ifelse(param == "(Intercept)", "Intercept", "Slope")) %>%
    dplyr::rename("se"=3) %>% # rename std error variable because easier to work with
    ggplot() +
    geom_point(aes(param, Estimate), size=3) + # point estimate
    # 95% confidence interval (1.96 * standard error)
    geom_linerange(aes(param, ymin=Estimate-(1.96*se), ymax=Estimate+(1.96*se))) +
    geom_hline(yintercept=0, lty=2) + # horizontal line marking zero (i.e. no effect)
    theme_minimal() +
    facet_wrap(~param_type, scales="free") + # split plot by parameter type
    theme(axis.text = element_text(size=12),
          axis.title = element_text(size=12),
          strip.text = element_text(size=14)) +
    xlab("Parameter") + ylab("Estimate (95% confidence interval)") +
    coord_flip() # flip so the plot is horizontal

  return(plot)
}
```

What does this initial statistical model suggest about the relationship between hare occupancy, grazing and habitat metrics?

It suggests a substantial positive relationship with livestock occurrence, as well as declining occupancy at further distances from water, and increasing in closed habitat. The latter effect seems quite counterintuitive in that we don't usually think of hares as scrub/forest species. Why might this be? Can you think of some ecological reasons for this relationship?

The next stage is to critique the model. Some questions to keep in mind...

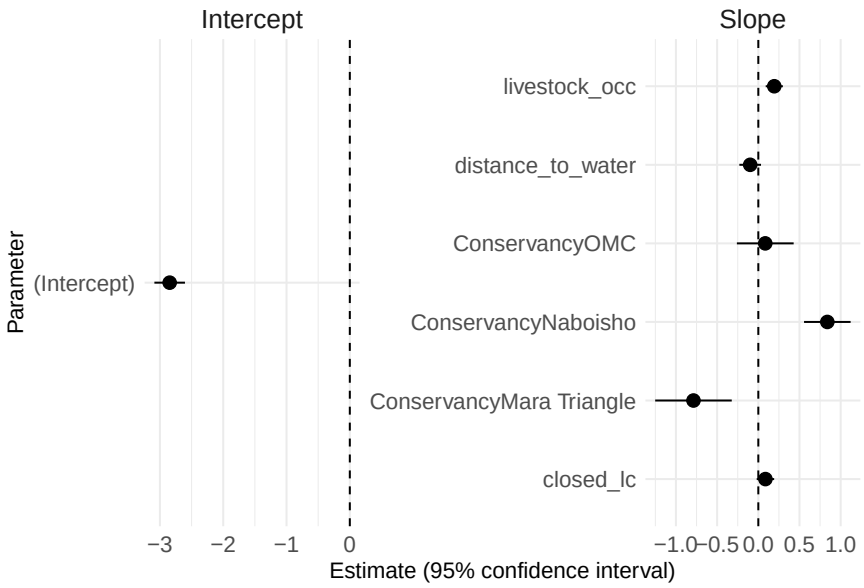
- Are there obvious clusters or nested structures in the data that we haven't accounted for that could be affecting our inferred relationships?
- What do the residuals from the model look like? (i.e. the remaining error not explained by the model)
- Are the errors independent from each other - i.e. are there any obvious structures or patterns in the residuals that indicate we are missing something from the model?
- Thinking about the dataset, how it was collected and its sampling design, what do you think? Is there anything we are obviously missing?

The conservancy in which the sampling was carried out! These are community-level conservancies where the livelihoods and land use patterns might differ substantially enough to affect the ecological community. Also as we saw in our boxplots above, livestock activity is definitely concentrated in the 3 more eastward conservancies with very little in the national park (Mara Triangle), and similarly there is less open habitat in the Mara Triangle - perhaps the apparent relationships are influenced by these differences. Let's take a look.

Exercise 4

- Add conservancy as a fixed effect to your model and save this model as *m3*. Has adding this changed the estimates of the slope parameters?

Note that conservancy is a categorical variable, so here rather than a slope we are effectively estimating how the intercept is different between each level of the categorical variable (i.e. are detections higher or lower in each conservancy?). For a categorical covariate we will be estimating n-1 new parameters (where is the number of levels of the covariate), because one of the categories becomes the intercept (the base factor).



Has including conservancy improved the model?

We'll explore what we mean by "improved" in more depth in next week's workshop, but for now we can look at some summary metrics of model goodness-of-fit to understand a bit more about how well the model is fitted to our data. In particular we can easily look at 2 metrics based on the log-likelihood (which maximum likelihood fitting approaches aim to maximise).

Firstly, the deviance - a measure of how much of the total variation in the observed data is explained by the model (lower values = more variation explained = better model). Secondly, the AIC (Akaike Information Criterion), which incorporates both the improvement in log-likelihood provided by including more parameters, while penalising model complexity to avoid models that overfit to the data (lower values = better).

We can also look at the distribution of residuals (the unexplained error not accounted for by the model) to check model assumptions. These plots get tricky to interpret visually for GLMs and especially logistic regression, because our outcome is binary, and different likelihoods make different assumptions about how error is distributed around the expected value (e.g. Poisson regression assumes error is wider with higher expected values; see the lecture slides). There are various ways to calculate residuals that account for different model likelihoods; we won't dig into this much, but this is a helpful resource: https://bookdown.org/ltupper/340f21_notes/deviance-and-residuals.html

```
# extract fit metrics from the model - are the deviance and AIC lower or higher
# in the model including conservancy?
metrics = data.frame(
  model = c("without_conservancy", "with_conservancy"),
  deviance = as.numeric( c(deviance(m2), deviance(m3)) ),
  AIC = as.numeric( c(AIC(m2), AIC(m3)) )
)
```

The GLM function also gives us a summary of the difference between the null and residual deviance for any model (see table at bottom of *summary()* function). This is the difference between the deviance of the simplest possible model explaining least variation (an intercept only model; the "null deviance"), and the deviance of this model with its covariates (the "residual deviance"). A significant reduction in the residual deviance compared to the null supports the inclusion of covariates in the model.

```
summary(m3)

# plot model fitted (expected) values against deviance residuals
# difficult tointerpret visually!
plot(
  fitted(m3),
  resid(m3, "deviance")
)
```

Accounting for repeat sampling and clustered sampling using mixed-effects models

One thing we haven't accounted for in the model yet is that we have multiple observations from at the same camera site (i.e. repeat sampling) across multiple rows. **Remember**, the model doesn't know anything by itself - we have to *tell* it that these observations on different days come from the same place. Otherwise we are violating the assumption that each observation is independent from all the others! (Certain sites might have higher or lower occurrence of hares because of something we haven't measured, which could bias our model results).

One way we could account for this is to include site as a categorical covariate, which would involve fitting a different individual beta parameter for each site - n=175! That's a lot of parameters to estimate and uses up a lot of degrees of freedom in the model (i.e. it is gobbling up a lot of information that could be used to infer other parameters).

An alternative would be to fit a mixed-effects (multilevel/hierarchical) model where, rather than assuming each site is its own completely independent entity, we instead assume that each site has its own intercept, and that these intercepts are realisations of an underlying population that is described by a normal distribution with some variance (sigma). A large sigma would mean that there is a lot of variation among sites in hare occupancy; a low sigma would mean that all the sites are actually quite similar.

As we saw in the lecture, the model structure would look like this, with α_s being 175 parameters, 1 for each site.

$Yi \sim Bernoulli(p_i)$

$log(p_i/(1-p_i)) = \beta_0 + X\beta_i + \alpha_s$

$\alpha_s \sim N(0, \sigma_s)$

Here, we are assuming that a normal distribution can describe this range of intercepts at site-level. What is nice about this approach is that, by assuming each site is an independent realisation of an underlying process described by the $N(0, \sigma_s)$, the underlying normal distribution model acts as a constraint to avoid the model overfitting parameters to any individual site. So we use up fewer degrees of freedom, and we pool information from across sites to learn something about how variable they are (the sigma, σ, parameter). (We are still treating each site as independent from the others, though - keep this in mind, as it will be important we when we get to spatial models next week!)

Let's take a look at this in practice. For this we'll use the glmer() function from the lme4 package.

```
# approach 1: fit a GLM with CT_site as a fixed effect, i.e. fitting an
# independent parameter per camera trapping site
# (this might throw an error message)
m4 = glm(Detected ~ 1 + livestock_occ + closed_lc + distance_to_water +
           Conservancy + CT_site,
         family=binomial(link="logit"),
         data=dd)
```

```
# call summary on the model - yuck, that's a lot of coefficients and an obvious
# issue with fitting the model
summary(m4)

# approach 2: define our site as a random intercept within a mixed-effects model
# we define a random intercept in the formula as "+ (1|covariate)"
m5 = lme4::glmer(Detected ~ 1 + livestock_occ + closed_lc + distance_to_water +
                   Conservancy + (1|CT_site),
                 family=binomial(link="logit"),
                 data=dd)

# call summary on m5
# now, as well as our intercept and slope parameters (fixed effects), we also
# have a random effects table which shows our sigma parameter (it shows both the
# variance and sd, i.e. sqrt(variance))
# What does this suggest about between-site variability in hare occurrence?

# extract and plot the fitted random intercepts for our sites
# we can see which sites have particularly high occurrence probabilities
# (high positive values) or low (strongly negative values)
site_ranefs = ranef(m5)$CT_site %>%
  as.data.frame() %>%
  tibble::rownames_to_column(var="CT_site") %>%
  dplyr::rename("Random_intercept"=2)

site_ranefs %>%
  ggplot() +
  geom_point(aes(CT_site, Random_intercept)) +
  geom_hline(yintercept=0) +
  theme_minimal() +
  ylab("Random intercept (log odds scale)") +
  coord_flip()
```

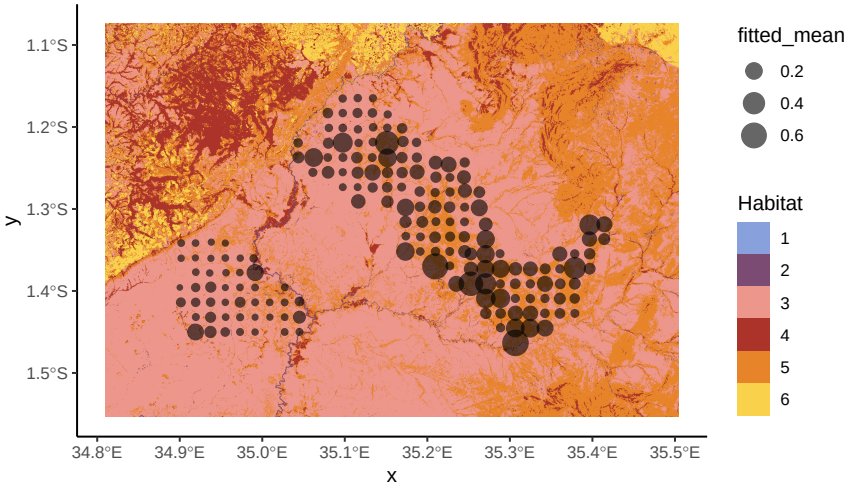
```
# we can map these too - which sites have higher or lower occurrence than expected
# conditional on all the other model components?
hab_df = as.data.frame(hab, xy=TRUE)
locs2 = left_join(locs, site_ranefs)
ggplot() +
  geom_raster(data = hab_df, aes(x, y, fill=factor(habitatfinal))) +
  geom_sf(data=locs2, color="black", aes(size=Random_intercept), alpha=0.6) +
  scale_fill_discrete(type = as.vector(MetBrewer::met.brewer(name="Archambault", n=6)),
                      name="Habitat") +
  theme_classic()
```

```
# let's look at the fixed effects, now we've accounted for site-level repeat sampling -
# has anything changed?
plotFixedEffects(m5)
```

```
# what about the goodness-of-fit statistics?
# compare AIC for the glm without the site effect, and for m5 with
# the site effect - what do you notice?
summary(m5)
```

Having fitted our model, it can be useful to visualise its predictions (the expected values) in relation to our study objective. For example, here, if we wanted to visualise where hare occurrence is expected to be highest or lowest, we could map the predicted values at each site.

```
# extract fitted values and calculate the mean fitted value per site
# (i.e. mean probabilty of occurrence across all sampled days)
site_preds = dd %>%
  dplyr::mutate(fitted = fitted(m5)) %>%
  dplyr::group_by(CT_site) %>%
  dplyr::summarise(fitted_mean = mean(fitted))

# map over space
hab_df = as.data.frame(hab, xy=TRUE)
locs2 = left_join(locs, site_preds)
ggplot() +
  geom_raster(data = hab_df, aes(x, y, fill=factor(habitatfinal))) +
  geom_sf(data=locs2, color="black", aes(size=fitted_mean), alpha=0.6) +
  scale_fill_discrete(type = as.vector(MetBrewer::met.brewer(name="Archambault", n=6)),
                      name="Habitat") +
  theme_classic()
```



Extension Exercises

Now we have several exercises to explore the approach we have taken above, focusing on building mixed-effects models of species occurrence across our camera trap network. While you are working through these, think about some of the aspects of the data and sampling process that we have not yet taken into account. These might include...

- Imperfect species detection and what's driving it
- Species behaviour over space
- Unmeasured local factors (what didn't we measure?)

Exercise 5

Another dimension of the sampling we haven't taken into account yet in the model is the temporal dimension. Perhaps there were particular days when our species was particularly active (e.g. because of weather, or seasonal behaviour trends)? Explore variation in detections among dates, and try incorporating date as a random intercept into the model.

- Does this improve model fit, and if so, does it change our findings? Is there any evidence that including the date of sampling is substantially affecting our inference?

Exercise 6

We could also explore whether weather affects our species' probability of occurrence. The dataframe contains an estimate of local daily precipitation, extracted from the ERA5 reanalysis dataset for the study time period. Plot this against date to look at how much this varies across the time period, then try including this as a covariate in the model.

- *Think*: what would we causally expect precipitation to act on, the true species occupancy, or our likelihood of detecting it?

Exercise 7

There are several other wildlife species included in the camera trap tagged images data frame. Try developing your own models and hypotheses based on these other species data and our available environmental and social covariates.

- How different are your findings for different species? Are there any consistent patterns?

Exercise 8

Explore the sensitivity of your results to varying other aspects of the models. In particular, the most appropriate size of the buffer zone to calculate habitat or human population density might be different between different species (e.g. between wide- and narrow-ranging species), and we might want to explore whether this affects our findings.