

# Actual and future predicted occupancy of the Black Kite in Spain

- Case study of the course Monitoring and occupancy modelling -

Felix Nößler

2<sup>nd</sup> semester in master program: Ecology, Evolution and Conservation (M.Sc.),

Registration number: 810578

August 26, 2021

Module Quantitative conservation biogeography

Prof. Dr. Damaris Zurell & Dr. Guillermo Fandos-Guzman

---

## ABSTRACT

Climate change affects and will affect all levels of biodiversity. The Black Kite, as an abundant top predator, plays an important ecological role in Spain. The main goal is to assess the current occupancy of the Black Kite in Spain and make predictions for the change of occupancy due to climate change in the future. Furthermore, it is analysed which site covariates are important for the occurrence of the Black Kite. For the climate change we assume in general an increase of the annual mean temperature and a decrease of the total precipitation in Spain in the next 80 years. The main method is a static occupancy model with a spatial scale of 21 km<sup>2</sup> (grid cells). The eBird data set from 2019 is filtered for repeated surveys and used as the data source for fitting the occupancy model. The selection of site covariates is based on ecological knowledge of the focal species and an exploratory analysis with a Random Forests model. As detection covariates the duration of the observation and number of observers are selected. Used site covariates are the tree cover, grass cover, bare soil cover, annual mean temperature, isothermality, annual precipitation, distance to closest landfill and distance to closest river or lake. For the area of the mainland of Spain, the model predicts a mean occupancy probability under actual climate conditions of 52.6 %, in contrast of only 37.8 % under future climate conditions. The main driver is that the occupancy decreases with high annual mean temperature values. The uncertainties of the model are discussed and the transferability of the results to conservation planning is evaluated.

---

# 1 Introduction

Climate change affects all levels of biodiversity (Bellard et al. 2012; Garcia et al. 2014). A main task in ecological research is to build accurate models to predict the biological response to climate change (Urban et al. 2016; Araújo and Rahbek 2006).

One of the affected groups by the climate change are birds of prey. Birds of prey play a crucial role in ecosystems. Very often they are top predators. They can be regarded as flagship species, because they are very vulnerable to human activity, because of their role as top predator in food chains and because of the attractiveness of their behavior to humans. Furthermore, they provide regulating, supporting and cultural ecosystem services (Donázar et al. 2016).

A raptor species with a very wide distribution is the Black Kite, *Milvus migrans* (Boddaert, 1783). The distribution of this species ranges from Western Europe to East Asia and Australia (BirdLife International 2021). The species is mainly migratory and the European population stays in the winter in sub-Saharan Africa. The individuals leave the breeding area between July and October and come back back between February and May (Panuccio et al. 2014; BirdLife International 2021). In this case study the focus lies on the Black Kite population in Spain.

Citizen science data are often used in species distribution modelling, because of the high data availability. However there are some challenges in using these data sets, for example

the spatio-temporal bias of the of the observations. For instance people tend to watch for birds more in the breeding season and around their own home (Robinson et al. 2018; Reich et al. 2018). Two techniques are available to overcome this difficulties. First, it is possible to filter out low quality observations. Second, it is possible to apply statistical techniques to address sampling bias and observational heterogeneity (Steen et al. 2019). In this study the second option is used. Observational heterogeneity and imperfect detection is tackled with an extension of species distribution model, the so-called occupancy model (MacKenzie, Nichols, et al. 2002).

The model is used to learn more about the ecology of the focal species. This knowledge can be used in conservation planning.

The main questions are (1) which site covariates are most important for the occurrence of the Black Kite and (2) what happens to the Black Kite population in Spain under future climate conditions.

## 2 Methods

Data of the species occurrences are from eBird data (Sullivan et al. 2009; Cornell Lab of Ornithology 2021). The dataset is filtered for repeated surveys, that were done three to ten times in the same area between April and June of 2019. Only standing or travelling surveys with a total distance up to 5 km with one to five observers were used. To get an overview a map with all observations was produced using QGIS Development Team (2021).

Data preparation, selection of covariates, model selection, model evaluation and prediction were done with R Core Team (2021). Exploratory analysis of the covariates were done in Van Rossum and Drake (2009). An ODMAP-Protocol (Zurell et al. 2020) describing typical steps in species distribution modelling can be found in the electronic appendix.

### **Spatial resolution of the model**

The home range size differs between younger not breeding individuals, that are one to seven years old, so-called floaters, (Blas et al. 2009) and the breeding individuals. Floaters had in south Spain an home range size over 300 km<sup>2</sup>, breeding females 43 km<sup>2</sup> and breeding males 80 km<sup>2</sup> (Tanferna et al. 2013). However, these home range values were observed with radio-tracking of individuals and later on the calculation of the minimum convex polygon. The main activity is probably more restricted to centre of the area.

A resolution of 2.5 minutes (roughly 21.5 km<sup>2</sup>) was chosen. It is possible to argue that the size is too small for the large home range sizes of the Black Kite. However through spatial subsampling the data loss is higher with a coarser resolution and the connection between the observations and the site variables becomes less strong.

### **Selection of detection covariates**

Detection covariates that are present in the eBird data are used. These are the day of the year, time when observation started, duration

of observation, walking distance and protocol type (standing or travelling). The two most important covariates from the Random Forests model (see section on exploratory analysis of covariates) were used in the final model.

### **Selection of site covariates**

First a preliminary selection of site covariates was compiled. These set include all bioclimatic variables (Fick and Hijmans 2017), land cover fractions, as the tree cover, the grass cover and the bare soil cover (Buchhorn, Smets, et al. 2020), and the distance between the centre of the raster cells to the closest land-fill and the closest river or lake.

In the next step ecological hypotheses were formulated based on existing ecological knowledge from the literature. Some of the covariates were selected based on these hypotheses. As an additionally step, all preliminary covariates were used to fit a Random Forests model and to get the importance of all the covariates for predicting the detection probability of the Black Kite. The most important covariates from these analysis were added to the list of covariates.

The covariates from the ecological hypotheses and from the Random Forests model were tested for collinearity with the Variance Inflation Factor (as implemented in Heiberger 2020). Highly collinear covariates were removed. Maps of the final site covariates can be found in the appendix (see Figure 7). The complete selection process is shown in Figure 1 and further explained in the next sections.

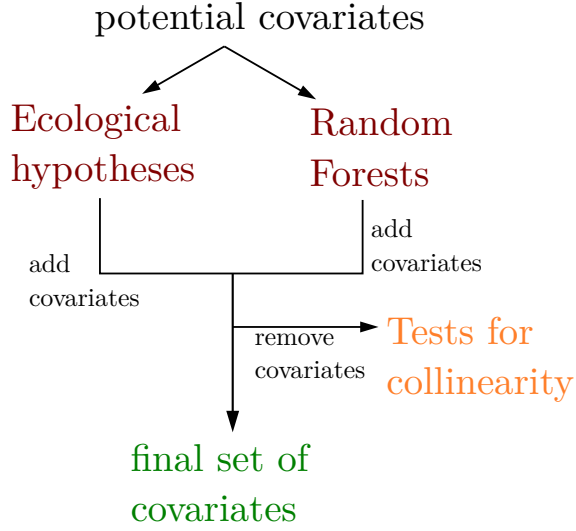


Figure 1: Procedure of selecting site covariates for the occupancy model

### Data preparation of site covariates

For all raster operations Hijmans (2021) was used. Bioclimatic variables were download from Fick and Hijmans (2017) in the target resolution. Tree cover, herbaceous vegetation cover and bare soil cover were downloaded (Buchhorn, Lesiv, et al. 2020; Buchhorn, Smets, et al. 2020). The raster images were merged to one file and then cropped to mainland of Spain. The resolution was resampled to the target resolution using a bilinear interpolation.

Land cover data was retrieved in vector format (Copernicus Land Monitoring Service 2018) and cropped to the mainland of Spain. The distance between each centre of the raster cell to all polygons of landfills and rivers or lakes was calculated with Bivand and Rundel (2020). Afterwards, the minimum distance was saved.

### Ecological justification of site covariates

The selection of covariates is based on existing ecological knowledge about the focal species. A list of all site covariates and a corresponding simplified hypothesis can be found in Table 1.

Table 1: Site covariates and simplified hypothesized response of occupancy of the Black Kite, arrows symbolize the expected occupancy probability with higher values of the respective covariate

Site covariates
Annual Mean Temperature ↘
Annual precipitation ↗
Tree cover ↘
Grass cover ↗
Bare soil cover ↘
Distance to closest river or lake ↘
Distance to closest landfill ↘

The Black Kite uses a variety of feeding sources, for example birds, fish, crayfish, insects, carrion, vegetable matter and smaller mammals (Sergio and Boto 1999; Vinuela and Veiga 1992; BirdLife International 2021). Part of the prey is found in wetlands and marshes and it was proposed that the Black Kite has a habitat binding to wetlands and marshes (Veiga and Hiraldo 1990; Tanferna et al. 2013). Accordingly, the distance to the closest lake or river was calculated and used as a site covariate. The distance to the closed lake or river was used, because it is more important that lakes or rivers are nearby than that the Black

Kite was actually observed in a cell with a lake or a river.

It has been shown that Black Kites visit landfills for feeding (De Giacomo and Guerrieri 2008; Blanco 1994). Therefore the distance to the closest landfill were analysed. The landfill cover was not used, because in most of the raster cells landfills are not present and therefore the model fitting will not be optimal.

Black Kites breed in branches of trees. However, closed woodlands are avoided (Tanferna et al. 2013). Therefore the hypothesis is that the occupancy of the Black Kite follows a unimodal distribution with regard to the tree cover, where the occupancy is high at low to intermediate tree cover. In contrast, there should be a positive relationship with the grass cover, because the Black Kite is searching for prey in open landscapes. The bare soil cover is especially high in the mountains, the hypothesis is that the Black Kites is more a lowland species in Spain and avoids areas with high bare soil cover.

The annual mean temperature is added, because it is hypothesized that the Black Kite is negatively affected by too high annual mean temperatures. Additionally, a high annual precipitation may lead to more vegetation and more prey for the Black Kite. Both covariates are especially important for predicting the change of occupancy due to climate change.

### **Exploratory analysis of site and detection covariates**

An exploratory analysis of the importance of the site and detection covariates was done

using the machine learning method Random Forests (Ho 1995). All covariates as described before were used. These include all bioclim variables, land cover fractions, cover of water bodies, cover of landfills, distance to closest water body, distance to closest landfill and all detection covariates. From each grid cell one observation was selected and related to the site and detection covariates. Because of the higher number of non-detection an balanced Random Forest Classifier was used (Lemaître et al. 2017, with default parameters). Preparing the data, standardization of data and splitting in train and test data (80 % and 20 %) was done with Pedregosa et al. (2011), Harris et al. (2020), and McKinney (2010). 500 simulation were done to account variability in the model fit to the different data sets (each time one of the up to ten observations per grill cell was chosen). The mean and the standard error of the importance of all covariates was calculated. The effect of the most important covariates on the detection probability was tested. Therefore all other covariates were set to the mean and only the focal covariate was changed. The python script for conducting the analysis can be found in the appendix.

### **Selection of best model**

A nullmodel, a model with only detection covariates, a model with only site covariates and a full model (site and detection covariates) were compared with the Akaike information criterion (AIC) and the Akaike information criterion corrected for small sample size

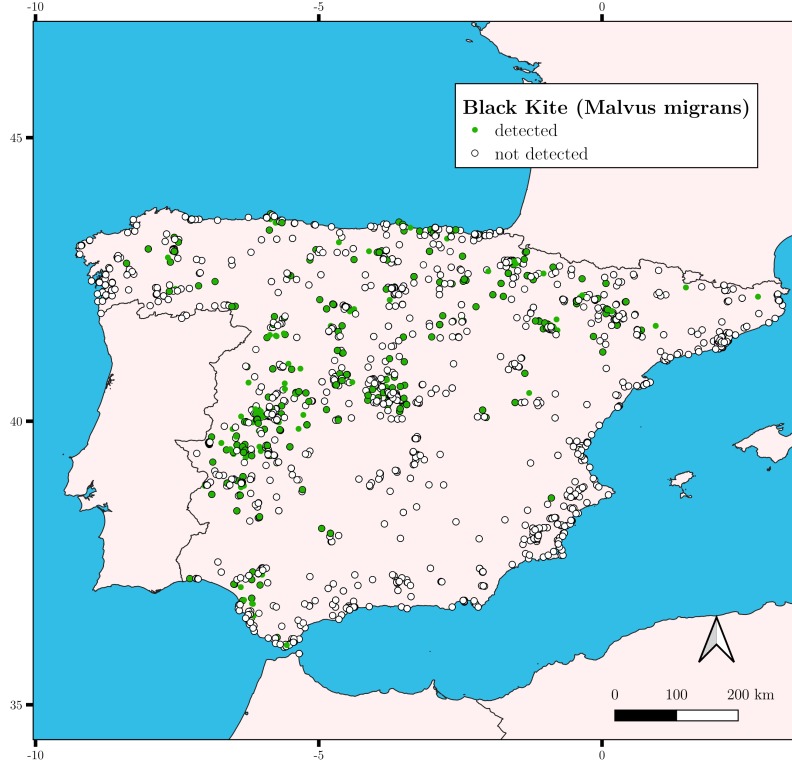


Figure 2: Detection of the Black Kite in the filtered eBird data set in Spain, each dot represents one observation within the repeated surveys, created with QGIS Development Team (2021)

(AICc) with Fiske and Chandler (2011). The model with the lowest AIC and AICc was chosen.

To compare the stability of the model an average model was build from the best model. Subsets of the best model were chosen with Bartoń (2020), all detection covariates were used as fixed terms (present in all subsetted models). Models were sorted according to the lowest AIC and selected with cumulative Akaike weight (Wagenmakers and Farrell 2004) larger than 95 %. The selected models are weighted with the AICs and averaged with Bartoń (2020).

## Model evaluation

The predictive power of the best model is evaluated with the  $R^2$ -value (Nagelkerke 1991). A parametric bootstrapping approach is used to access the goodness of fit ("parboot" function as implemented in Fiske and Chandler 2011). From the test results the  $\hat{c}$ -value was calculated. If this values is noticeably larger than one, it indicates overdispersion of the model. With this  $\hat{c}$ -value a quasi-Akaike information criterion corrected for small sample size (qAICc) was retrieved to see if there are big differences to the calculated AICc. Another goodness of fit as proposed by MacKenzie and Bailey (2004) was carried out (imple-

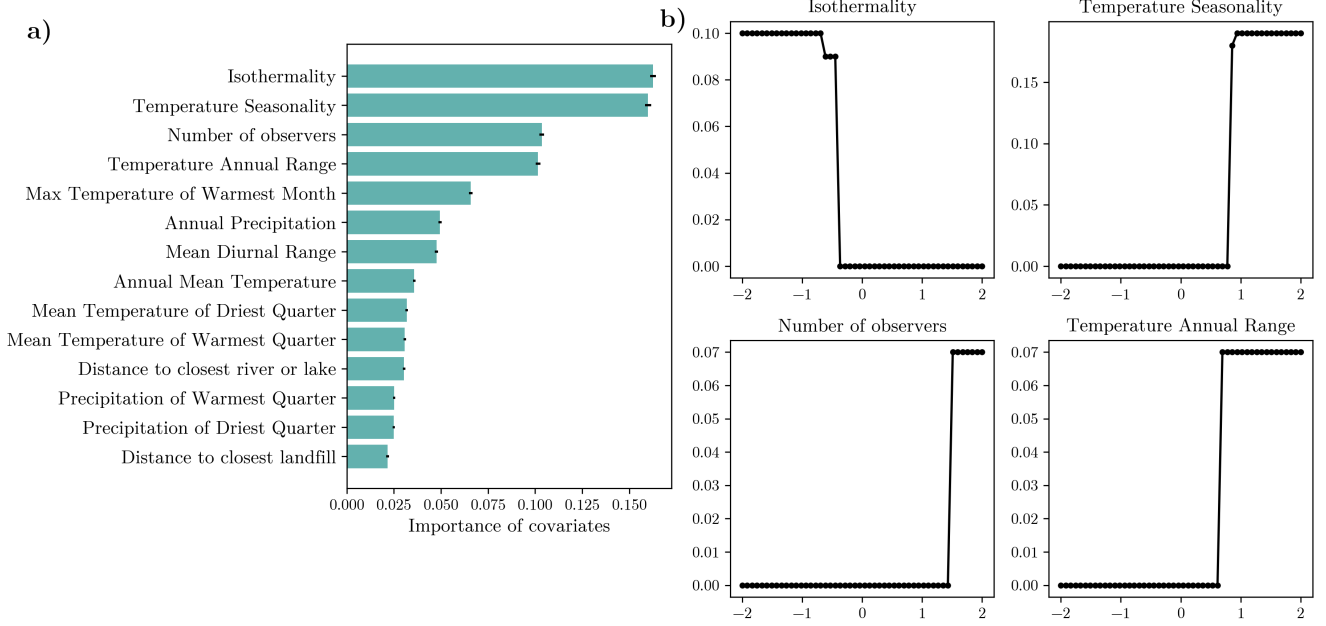


Figure 3: Relationship between covariates and detection probability, (a) importance of the covariates in the Random Forests model (only covariates with importance  $> 0.02$  are included), mean importance values of 500 simulation are shown, black lines represent the standard error, and (b) influence of the four most important covariates on the detection probability of the Black Kite in Spain, the y-axis shows the probability, the x-axis the standardized value of the covariate, created with Hunter (2007)

mented in Mazerolle 2020).

## Prediction

First, predictions were done first on the grid cells with observational data and secondly with the best model for the mainland of Spain. A map was produced with the predicted occupancy and the standard error of the prediction. The outcome of the best model were compared to the results from the average model.

Prediction of future climate conditions were done accordingly to existing knowledge about the climate change. The annual temperature will likely rise around  $3^{\circ}\text{C}$  and the annual rainfall will decrease around 10 % from 2020 to 2100 (State Meteorological Agency (AEMET)

2021). Iturbide et al. (2021) show under the SSP2 4.5 scenario (Riahi et al. 2017) an increase between 2 and  $3^{\circ}\text{C}$  and a decrease of total precipitation between 5 % and 20 % in different areas of Spain (mean values of 32 models, 2081-2100, relative to 1995-2014). The model agreement is high for the temperature increase, however the signal is not robust for the decrease in total precipitation. These data set is used to model the future occupancy of the Black Kite. The interactive atlas provides regional predictions on a spatial scale of 95 km edge length of a raster cell. GeoTIFFs were downloaded for the change in annual mean temperature and the annual total precipitation (change from 1995-2014 to 2081-2100). These

raster files were cropped to the mainland of Spain and resampled to the target resolution (2.5 minutes of a degree). Maps of the changes in the target resolution can be seen in the appendix (see Figure 8). The change in temperature and precipitation were applied to the respective bioclimatic variables (Fick and Hijmans 2017).

We will only incorporate the habitat suitability in relation to climate factors and omit all biological mechanisms that are proposed to play a role for prediction by Urban et al. (2016) like demography or evolution of a population. It is therefore a simplistic model, but the goal is to catch the main trend.

The effect of the site and detection covariates were tested. Accordingly, all other covariate than the target covariate were set to the mean value and the predictions were done in the range of the target covariate that the model was fitted to. Occupancy probabilities were produced for this range the uncertainty was measured with the standard error.

### 3 Results

A map with all filtered observation from the eBird data set is shown in Figure 2. It can be observed that the main distribution is in central Spain. The Black Kite was detected from the coast of the Gulf of Cadiz northwards up to the western Pyrenees. However, in the south-eastern part of Spain detections of the Black Kites are rare.

### Exploratory analysis of covariates

The Random Forests model could explain close to 100 % of the detection of the Black Kite in the train and the test data set. An overview of the most important covariates can be found in Figure 3a.

Especially important are covariates that deal with temperature evenness over the year like isothermality, temperature seasonality and temperature annual range. The only detection covariate that really plays a role is the number of observers. More observers improve the chance to detect a Black Kite (see Figure 3b). Land cover fractions are not represented under the most important covariates. The annual precipitation is sixth influential covariate. The distance to the closest river or lake and the distance to the closest landfill play minor role.

The four most important factor, namely the isothermality, temperature seasonality, temperature annual range and maximal temperature of warmest month were added as preliminary covariates.

### Occupancy model

Seven site covariates were analysed from the ecological hypotheses and four were added from the Random Forests model. Because of the high collinearity temperature seasonality, temperature annual range and maximal temperature of warmest month were excluded from the final list of covariates. After that the variance inflation factor was for all covariates below 1.9. The highest correlation exists



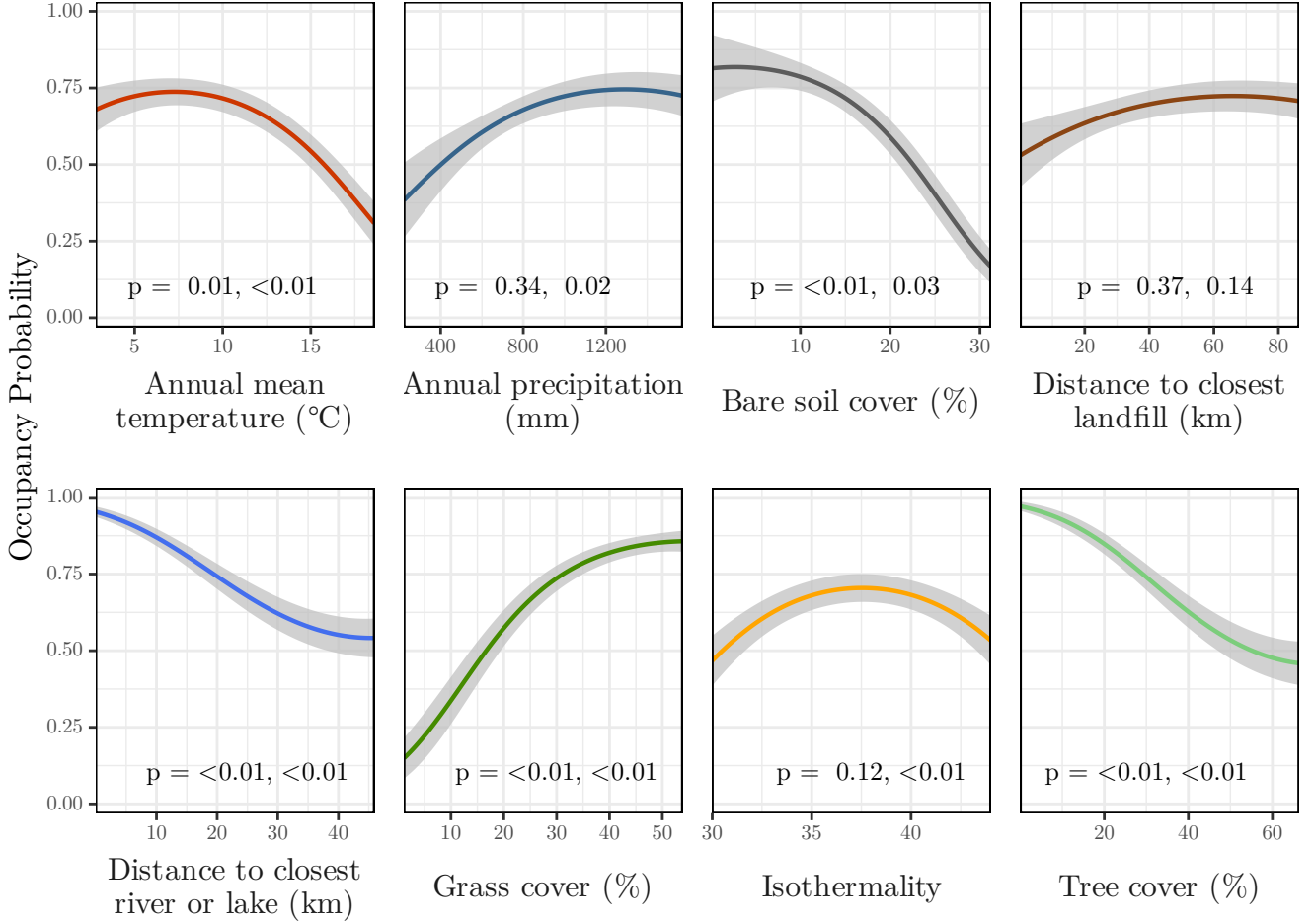


Figure 4: Predicted occupancy of the full model in response to site covariates, for each plot only the focal site covariate was varied, grey areas represent the standard error, the p values show the significance of the first and the second coefficient, created with Wickham (2016)

between the tree cover and the annual precipitation (Pearson  $\rho = 0.62$ ).

The final best model is the full model and consists of 20 parameters and has an AIC of 4578. The second best model with only the site covariates has an AIC that is 143 higher (delta AIC). The AICc gives roughly the same result. In the full model there are two detection covariates, namely the number of observers and the duration of observation. The explained variance in the detection of the Black Kite is

35 % ( $R^2$ -value).

An average model was build from three single models. These are one model with all site covariates except the distance to closest landfill, the full model and a third model with all site covariates except the annual precipitation and the closest distance to landfills.

All site covariates are fitted as polynomials of two degrees. These are in the full model the annual mean temperature, the isothermality, the annual precipitation, the bare soil cover,

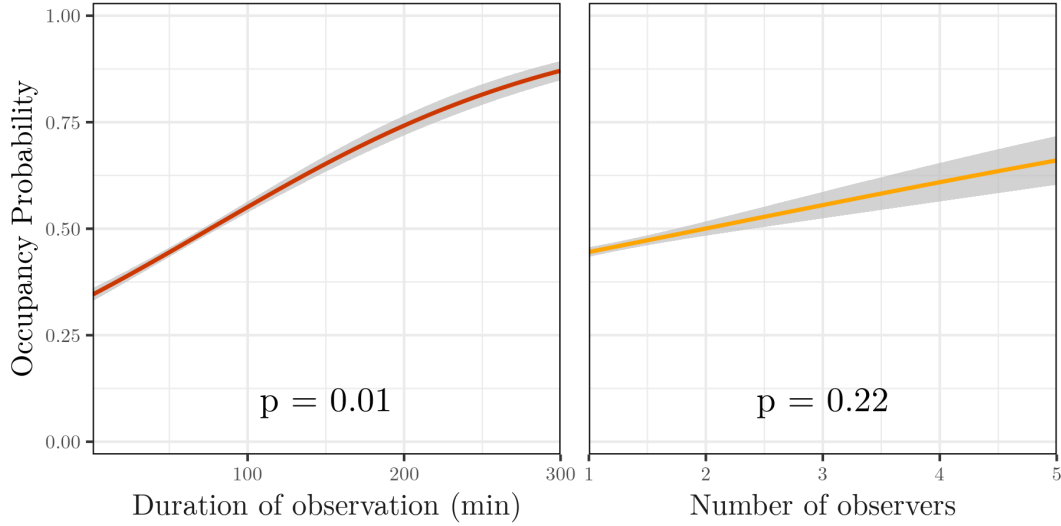


Figure 5: Predicted occupancy in response to detection covariates, for each plot only the focal detection covariate was varied, grey areas represent the standard error, the p values show the significance of the coefficient, created with Wickham (2016)

the grass cover, the tree cover, the distance to closest river or lake and the distance to closest landfill. All estimated values of the site covariates are significant different from zero, except the first coefficients of the isothermality and the annual precipitation and both coefficients of the distance to the closest landfill. The occupancy probability shows a positive relation with the grass cover and the annual precipitation. A unimodal relationship can be seen between the isothermality and the annual mean temperature and the occupancy probability of the Black Kite. Bare soil cover, distance to closest lake or river and a high tree cover have a negative influence on the occupancy probability (see Figure 4). These results go inline with the outcome of the average model. The uncertainty is high in the site covariates annual precipitation and the distance to closest landfill, the occupancy probability shows the same clear response to all other site covariates

as with the full model (see Figure 9 in the appendix).

The duration of the observation has a positive influence on the occupancy probability and the coefficient is significant different from zero. In contrast, the number of observers does not show a significant result (see Figure 5).

The full model predicts for the grid cell with observational data that 522 out of 1032 grid cells are occupied (51 %, median best unbiased predictor from occurrence state). In 503 grid cells was the Black Kite detected (49 %). The MacKenzie and Bailey (2004) goodness-of-fit test shows that the full model adequately fits the data ( $\hat{c} = 0.83$ ,  $p = 1$ ). Also the parametric bootstrap test does not indicate a lack of fit ( $\hat{c} = 0.97$ ,  $p = 0.86$ ). The  $\hat{c}$ -values are below zero, for this reason no QAIC and QAICc are calculated (they have the same values as the AIC and AICc when setting the  $\hat{c}$ -value to one).

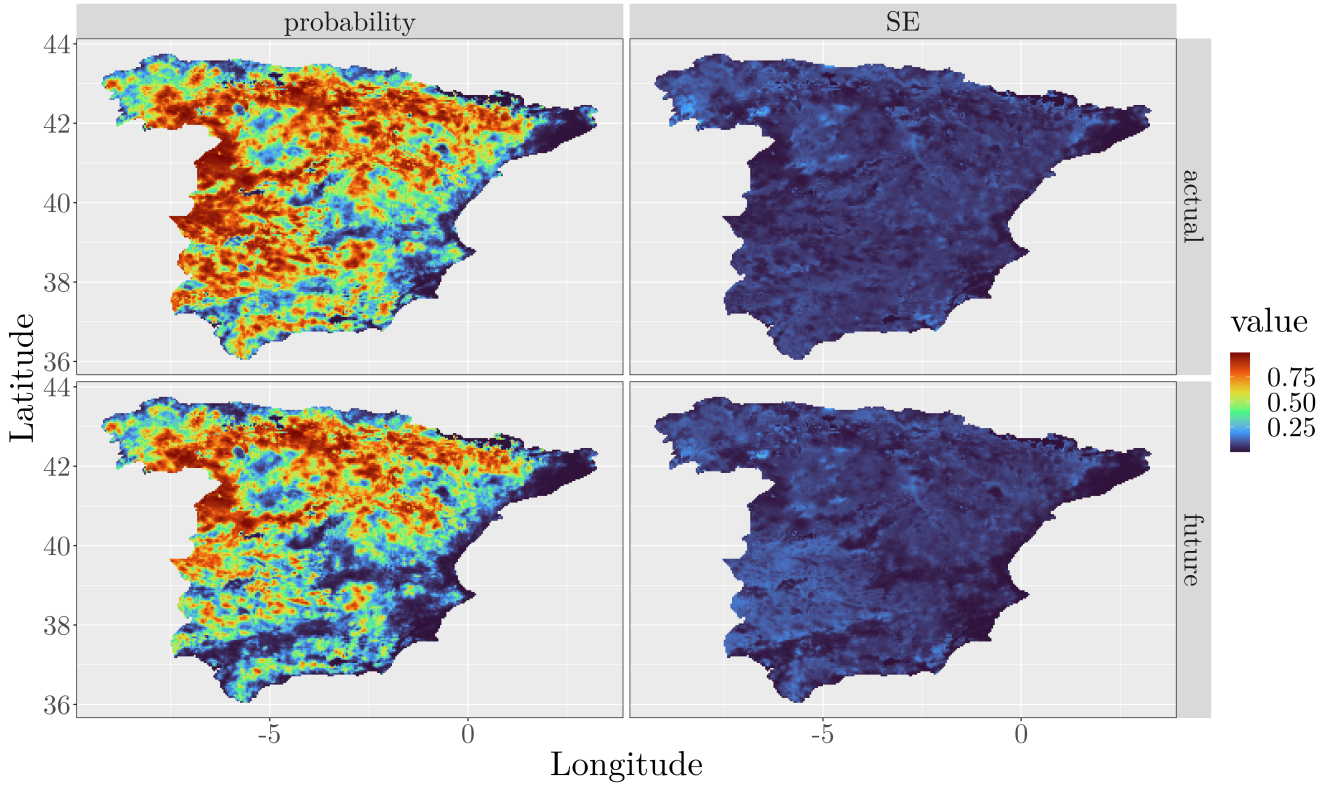


Figure 6: Occupancy map under actual and future climate conditions predicted with the full model, the occupancy probability is shown in the left maps, the corresponding standard error in the plots on the right side, created with Wickham (2016)

For the area of the mainland of Spain, the mean occupancy probability under actual climate conditions is 52.6 % (16748 grid cells with occupancy probability  $\geq 0.5$ ). The mean occupancy probability decreases and is 37.8 % under future climate conditions (10290 grid cells with occupancy probability  $\geq 0.5$ ). This decrease in occupancy is also visible in the map especially between Sevilla and Madrid (see Figure 6). In 92 % of all grid cells there is a lower occupancy probability under future climate conditions, in contrast in only 8 % of all grid cells the occupancy probability is higher under future climate conditions. The average model predicts very similar results (see Figure

10 in the appendix).

## 4 Discussion

Our results imply that the climate change may negatively affect the occupancy of the Black Kite in Spain. However even if higher annual mean temperature has a clear negative effect on the occupancy of the Black Kite, the uncertainty for the annual precipitation is high. The climate predictions from the IPCC (Iturbide et al. 2021) had a very coarse scale in comparison with the scale of the occupancy model (95 km versus 4.6 km edge length) and the mod-

els had no robust signal about the decrease in annual precipitation in Spain. Furthermore, the annual mean temperature and the annual precipitation are not the only factors that influence the occupancy of the Black Kite. Also changes in the habitat will affect the occurrence of the Black Kite. Additionally, many biological mechanisms of the Black Kite are not covered in this simple model.

Tanferna et al. (2013) point out some challenges in the protection of birds of prey. These are the large home range sizes (especially for younger not breeding individuals), they uses different habitat types in different seasons of the year and they often need non-identical habitat characteristics on different spatial scales. That is why it is not so easy to derive conservation management measures from this study. Nevertheless, the grass cover and distance to closest river or lakes play an important role for the occupancy probability of the Black Kite in Spain as already mentioned in the literature (Tanferna et al. 2013; Veiga and Hiraldo 1990). It can be concluded that the Black Kite searches in open landscape and along rivers and lakes for prey. Conservation of the prey of these landscapes will probably also positively affect the population of the Black Kite.

However, no effect of the distance to closest landfill was detected here as proposed by Blanco (1997) and Blanco (1994). It is unlikely that landfills really do not play a role for searching for food. A possible explanations for this mismatch are that large landfills that are probably well visible in the Corine Land Cover data set (Copernicus Land Monitoring Service 2018) are often covered in reality and not accessible to the Black Kite. It is also possible that the citizen scientist did not look often around landfills and therefore the data fit is not good.

The duration of observation is the only significant detection covariate in this study. Maybe it would help to add other detection covariate to tackle the imperfect detection. An example would be the weather. The weather can influence the sight of the observers and the activity of the Black Kites. Beyond that study it would be possible to implement more detection covariates.

A possible extension of this static occupancy model would be a dynamic occupancy model. A dynamic occupancy model can give more insights on the population trends and can make more accurate predictions (Green et al. 2019). However, data from more than one year is needed to perform such an analysis.

## References

- Araújo, M. B. and C. Rahbek (2006). “How does climate change affect biodiversity?” In: *Science* 313.5792, pp. 1396–1397. DOI: 10.1126/science.1131758.
- Bartoń, K. (2020). *MuMIn: Multi-Model Inference*. URL: <https://cran.r-project.org/package=MUMIn>.

- Bellard, C., C. Bertelsmeier, P. Leadley, W. Thuiller, and F. Courchamp (2012). “Impacts of climate change on the future of biodiversity”. In: *Ecology Letters* 15.4, pp. 365–377. DOI: 10.1111/j.1461-0248.2011.01736.x.
- BirdLife International (2021). *Species factsheet: Milvus migrans*. URL: <http://datazone.birdlife.org/species/factsheet/black-kite-milvus-migrans/> (visited on 07/19/2021).
- Bivand, R. and C. Rundel (2020). *rgeos: Interface to Geometry Engine - Open Source ('GEOS')*. URL: <https://cran.r-project.org/package=rgeos>.
- Blanco, G. (1994). “Seasonal abundance of Black kites associated with the rubbish dump of Madrid, Spain”. In: *Journal of Raptor Research* 28.4, pp. 242–245.
- (1997). “Role of refuse as food for migrant, floater and breeding Black Kites (*Milvus migrans*)”. In: *Journal of Raptor Research* 31.1, pp. 71–76.
- Blas, J., F. Sergio, and F. Hiraldo (2009). “Age-related improvement in reproductive performance in a long-lived raptor: A cross-sectional and longitudinal study”. In: *Ecography* 32.4, pp. 647–657. DOI: 10.1111/j.1600-0587.2008.05700.x.
- Buchhorn, M., M. Lesiv, et al. (2020). “Copernicus Global Land Cover Layers: Collection 2”. In: *Remote Sensing* 12.6, p. 1044. DOI: 10.3390/rs12061044.
- Buchhorn, M., B. Smets, et al. (2020). *Copernicus Global Land Service: Land Cover 100m: collection 3: epoch 2019: Globe*. DOI: 10.5281/zenodo.3939050.
- Copernicus Land Monitoring Service (2018). *Corine Land Cover (CLC) 2018*. European Union, European Environment Agency (EEA). URL: <https://land.copernicus.eu/pan-european/corine-land-cover/clc2018>.
- Cornell Lab of Ornithology (2021). *eBird Basic Dataset*. Ithaca, New York. URL: <https://ebird.org/data/download> (visited on 03/01/2021).
- De Giacomo, U. and G. Guerrieri (2008). “The feeding behavior of the Black Kite (*Milvus migrans*) in the rubbish dump of Rome”. In: *Journal of Raptor Research* 42.2, pp. 110–118. DOI: 10.3356/JRR-07-09.1.
- Donázar, J. A. et al. (2016). “Roles of Raptors in a Changing World: From Flagships to Providers of Key Ecosystem Services”. In: *Ardeola* 63.1, pp. 181–234. DOI: 10.13157/arla.63.1.2016.rp8.
- Fick, S. E. and R. J. Hijmans (2017). “WorldClim 2: new 1-km spatial resolution climate surfaces for global land areas”. In: *International Journal of Climatology* 37.12, pp. 4302–4315. DOI: 10.1002/joc.5086.
- Fiske, I. and R. Chandler (2011). “unmarked: An R Package for Fitting Hierarchical Models of Wildlife Occurrence and Abundance”. In: *Journal of Statistical Software* 43.10, pp. 1–23. DOI: 10.18637/jss.v043.i10.

- Garcia, R. A., M. Cabeza, C. Rahbek, and M. B. Araújo (2014). “Multiple dimensions of climate change and their implications for biodiversity”. In: *Science* 344.6183. DOI: 10.1126/science.1247579.
- Green, A. W., D. C. Pavlacky, and T. L. George (2019). “A dynamic multi-scale occupancy model to estimate temporal dynamics and hierarchical habitat use for nomadic species”. In: *Ecology and Evolution* 9.2, pp. 793–803. DOI: 10.1002/ece3.4822.
- Harris, C. R. et al. (2020). “Array programming with NumPy”. In: *Nature* 585.7825, pp. 357–362. DOI: 10.1038/s41586-020-2649-2.
- Heiberger, R. M. (2020). *HH: Statistical Analysis and Data Display: Heiberger and Holland*. URL: <https://cran.r-project.org/package=HH>.
- Hijmans, R. J. (2021). *raster: Geographic Data Analysis and Modeling*. URL: <https://cran.r-project.org/package=raster>.
- Ho, T. K. (1995). “Random decision forests”. In: *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR* 1, pp. 278–282. DOI: 10.1109/ICDAR.1995.598994.
- Hunter, J. D. (2007). “Matplotlib: A 2D graphics environment”. In: *Computing in Science & Engineering* 9.3, pp. 90–95. DOI: 10.1109/MCSE.2007.55.
- Iturbide, M. et al. (2021). *Repository supporting the implementation of FAIR principles in the IPCC-WG1 Atlas*. DOI: 10.5281/zenodo.3691645. URL: <https://github.com/IPCC-WG1/Atlas> (visited on 08/13/2021).
- Lemaître, G., F. Nogueira, and C. K. Aridas (2017). “Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning”. In: *Journal of Machine Learning Research* 18.17, pp. 1–5. URL: <http://jmlr.org/papers/v18/16-365>.
- MacKenzie, D. I. and L. L. Bailey (2004). “Assessing the fit of site-occupancy models”. In: *Journal of Agricultural, Biological, and Environmental Statistics* 9.3, pp. 300–318. DOI: 10.1198/108571104X3361.
- MacKenzie, D. I., J. D. Nichols, et al. (2002). “Estimating site occupancy rates when detection probabilities are less than one”. In: *Ecology* 83.8, pp. 2248–2255. DOI: 10.1890/0012-9658(2002)083[2248:ESORWD]2.0.CO;2.
- Mazerolle, M. J. (2020). *AICcmodavg: Model selection and multimodel inference based on (Q)AIC(c)*. URL: <https://cran.r-project.org/package=AICcmodavg>.
- McKinney, W. (2010). “Data Structures for Statistical Computing in Python”. In: *Proceedings of the 9th Python in Science Conference*. Ed. by S. van der Walt and J. Millman, pp. 56–61. DOI: 10.25080/Majora-92bf1922-00a.
- Nagelkerke, N. J. D. (1991). “A note on a general definition of the coefficient of determination”. In: *Biometrika* 78.3, pp. 691–692. DOI: 10.1093/biomet/78.3.691.

- Panuccio, M., N. Agostini, U. Mellone, and G. Bogliani (2014). “Circannual variation in movement patterns of the Black Kite (*Milvus migrans migrans*): A review”. In: *Ethology Ecology and Evolution* 26.1, pp. 1–18. DOI: 10.1080/03949370.2013.812147.
- Pedregosa, F. et al. (2011). “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12, pp. 2825–2830. URL: <http://jmlr.org/papers/v12/pedregosa11a.html>.
- QGIS Development Team (2021). *QGIS Geographic Information System*. QGIS Association. URL: <https://www.qgis.org>.
- R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: <https://www.r-project.org/>.
- Reich, B. J., K. Pacifici, and J. W. Stallings (2018). “Integrating auxiliary data in optimal spatial design for species distribution modelling”. In: *Methods in Ecology and Evolution* 9.6, pp. 1626–1637. DOI: 10.1111/2041-210X.13002.
- Riahi, K. et al. (2017). “The Shared Socioeconomic Pathways and their energy, land use, and greenhouse gas emissions implications: An overview”. In: *Global Environmental Change* 42, pp. 153–168. DOI: 10.1016/j.gloenvcha.2016.05.009.
- Robinson, O. J., V. Ruiz-Gutierrez, and D. Fink (2018). “Correcting for bias in distribution modelling for rare species using citizen science data”. In: *Diversity and Distributions* 24.4, pp. 460–472. DOI: 10.1111/ddi.12698.
- Sergio, F. and A. Boto (1999). “Nest dispersion, diet, and breeding success of Black Kites (*Milvus migrans*) in the Italian pre-Alps”. In: *Journal of Raptor Research* 33.3, pp. 207–217.
- State Meteorological Agency (AEMET) (2021). *Climate projections for the XXI Century*. URL: [http://www.aemet.es/en/serviciosclimaticos/cambio%7B%5C\\_%7Dclimat/](http://www.aemet.es/en/serviciosclimaticos/cambio%7B%5C_%7Dclimat/) (visited on 08/02/2021).
- Steen, V. A., C. S. Elphick, and M. W. Tingley (2019). “An evaluation of stringent filtering to improve species distribution models from citizen science data”. In: *Diversity and Distributions* 25.12, pp. 1857–1869. DOI: 10.1111/ddi.12985.
- Sullivan, B. L. et al. (2009). “eBird: A citizen-based bird observation network in the biological sciences”. In: *Biological Conservation* 142.10, pp. 2282–2292. DOI: 10.1016/j.biocon.2009.05.006.
- Tanferna, A., L. López-Jiménez, J. Blas, F. Hiraldo, and F. Sergio (2013). “Habitat selection by Black kite breeders and floaters: Implications for conservation management of raptor floaters”. In: *Biological Conservation* 160, pp. 1–9. DOI: 10.1016/j.biocon.2012.12.031.
- Urban, M. C. et al. (2016). “Improving the forecast for biodiversity under climate change”. In: *Science* 353.6304. DOI: 10.1126/science.aad8466.

- Van Rossum, G. and F. L. Drake (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace. ISBN: 1441412697.
- Veiga, J. P. and F. Hiraldo (1990). “Food habits and the survival and growth of nestlings in two sympatric kites (*Milvus milvus* and *Milvus migrans*)”. In: *Ecography* 13.1, pp. 62–71. DOI: 10.1111/j.1600-0587.1990.tb00590.x.
- Vinuela, J. and J. P. Veiga (1992). “Importance of rabbits in the diet and reproductive success of black kites in southwestern Spain”. In: *Ornis Scandinavica* 23.2, pp. 132–138. DOI: 10.2307/3676440.
- Wagenmakers, E. J. and S. Farrell (2004). “AIC model selection using Akaike weights”. In: *Psychonomic Bulletin and Review* 11.1, pp. 192–196. DOI: 10.3758/BF03206482.
- Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN: 978-3-319-24277-4. URL: <https://ggplot2.tidyverse.org>.
- Zurell, D. et al. (2020). “A standard protocol for reporting species distribution models”. In: *Ecography* 43.9, pp. 1261–1277. DOI: 10.1111/ecog.04960.



## Supplementary Information

The appendix consists of extra graphics and code for repeating the analysis.

### Graphics

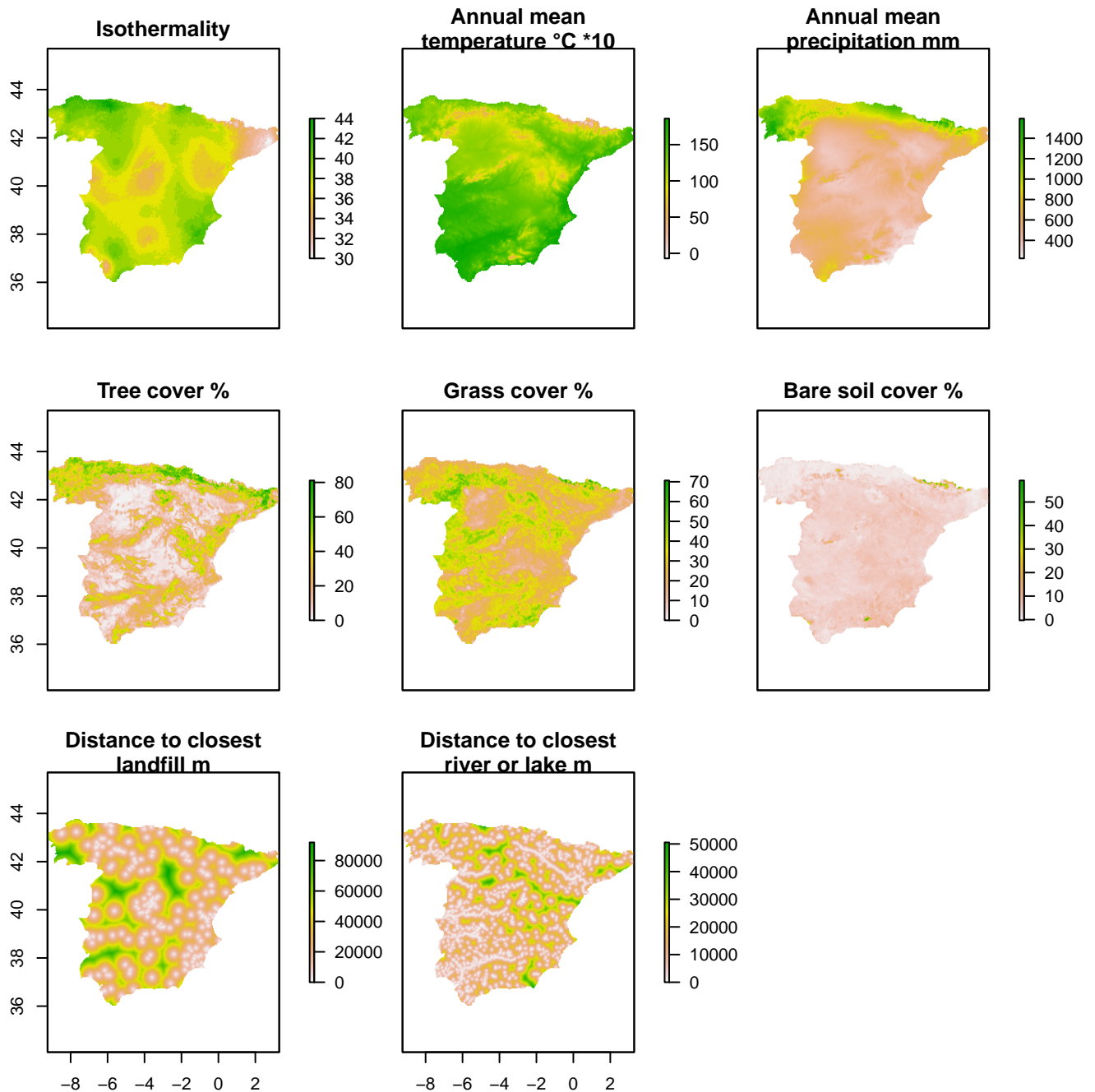


Figure 7: Maps of all site covariates, created with Hijmans (2021)

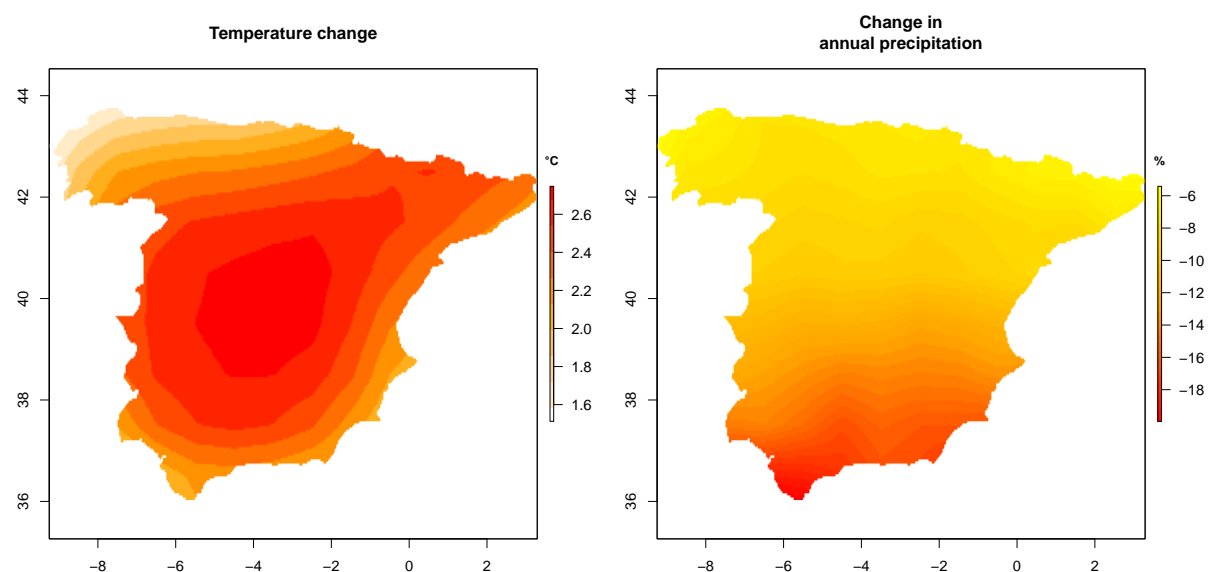


Figure 8: Change in the annual mean temperature and the annual precipitation between 1995-2014 and 2081-2100 with the SSP2 4.5 pathway, mean of 34 models (Iturbide et al. 2021), created with Hijmans (2021)

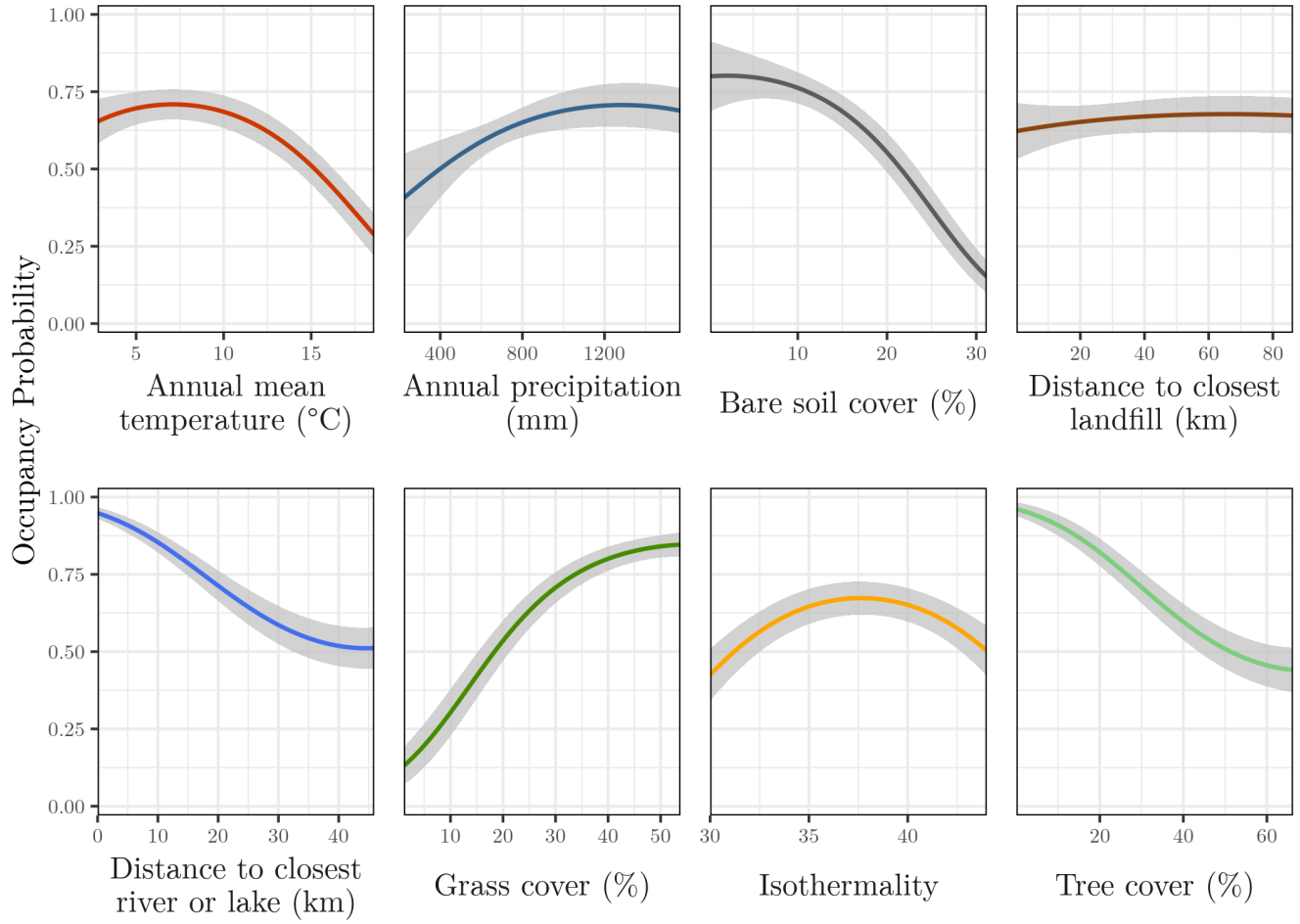


Figure 9: Predicted occupancy of the average model in response to site covariates, for each plot only the focal site covariate was varied, grey areas represent the standard error, created with Wickham (2016)

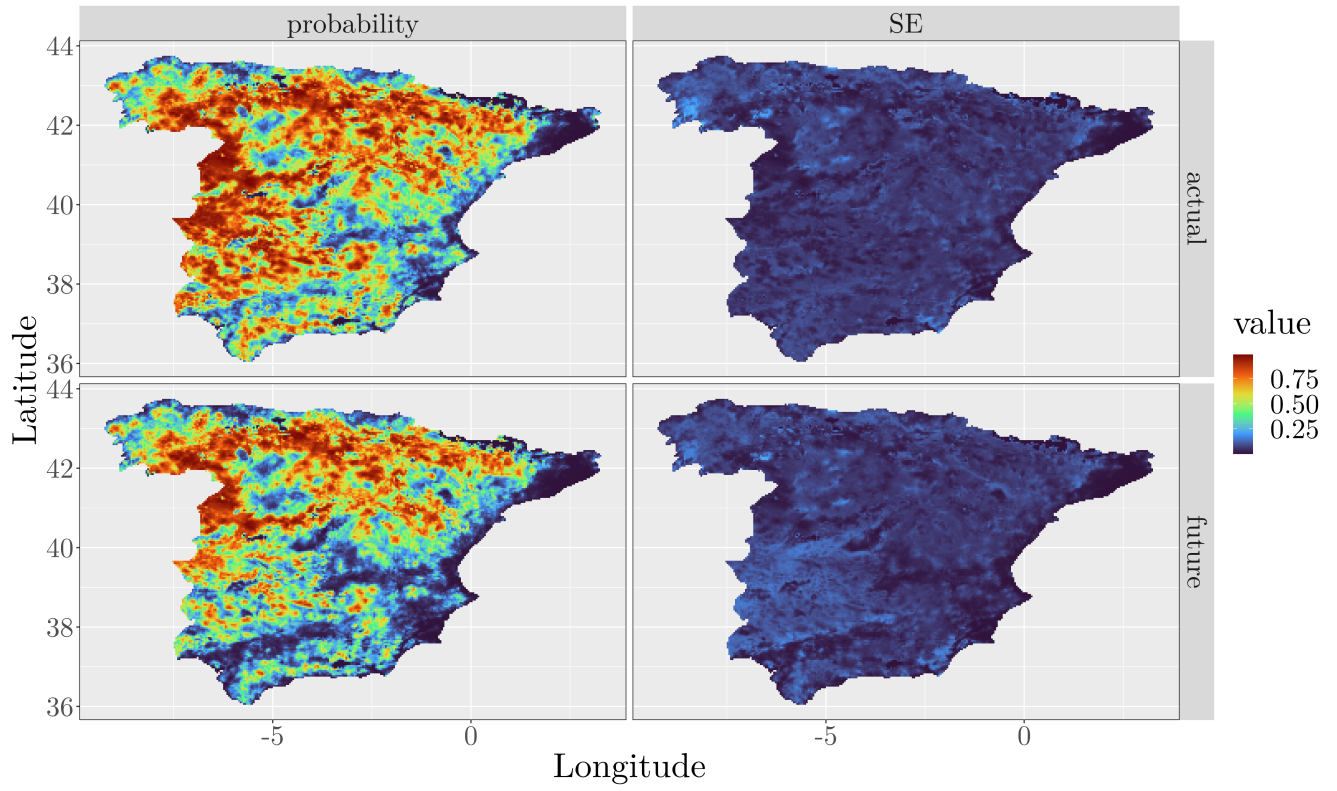


Figure 10: Occupancy map under actual and future climate conditions predicted with the average model, the occupancy probability is shown in the left maps, the corresponding standard error in the plots on the right side, created with Wickham (2016)

## Code

The data and the R and Python Code can be found at  
[https://github.com/FelixNoessler/QCB\\_Black\\_Kite](https://github.com/FelixNoessler/QCB_Black_Kite)

The R-scripts for running the analysis are also embedded here (Python script is further down on page 54):

```
1 #####
2 # First script
3 # Prepare the environmental data
4 #   - Land cover data
5 #   - Bioclimatic variables
6 #   - Distance to closest landfill and river or lake
7 #
8 # save everything as a raster stack in the target
9 # spatial resolution
10 #
11 # Prepare the change in annual temperature and
12 # in annual precipiytion for the prediction of
13 # of the futre climate condtions
14 #
15 #####
16
17
18
19 # Install required packages -----
20 packages <- c("ggplot2", "gridExtra", "dplyr",
21             "tidyr", "purrr", "HH", "psych", "MuMIn",
22             "rnatualearth", "rmapshaper",
23             "auk", "unmarked", "AICcmodavg",
24             "raster", "rgeos", "sp", "sf")
25
26 install.packages(setdiff(packages, rownames(installed.packages())))
27
28
29 # Loading packages -----
30 library(ggplot2)
31 library(dplyr)
```

```

32
33
34 # Loading geometries for the mainland of Spain -----
35 spain <- rnaturalearth::ne_countries(country = 'spain',
36                                     scale = 'medium',
37                                     returnclass = 'sf')
38 # spain %>%
39 #   ggplot()+
40 #   geom_sf(fill='black')
41
42
43 spain_crop <- rmapshaper::ms_filter_islands(spain,
44                                             min_area = 100000000000,
45                                             drop_null_geometries=T)
46 # plot(sf::st_geometry(spain_crop))
47
48
49 # Prepare climate data -----
50 ### Climate data
51
52 if (!file.exists('data/environmental_data/clim.grd')) {
53   clim <- raster::getData('worldclim',
54                           var = 'bio',
55                           res = 2.5,
56                           download = F,
57                           path = 'data/environmental_data')
58
59   raster::xres(clim) * 111.19
60
61   clim <- raster::crop(clim, spain_crop)
62   clim <- raster::mask(clim, spain_crop)
63
64   raster::writeRaster(clim,
65                       'data/environmental_data/clim.grd',
66                       format = 'raster',
67                       options = 'INTERLEAVE=BAND',
68                       overwrite = TRUE)
69 } else {
70   clim <- raster::brick('data/environmental_data/clim.grd')

```

```

71 }
72
73
74 # Prepare land cover data -----
75
76 ### Tree cover
77
78 if (!file.exists('data/environmental_data/lc_tree.grd')) {
79   lc_tree1 <- raster::raster('data/environmental_data/tree_cover1.tif')
80   lc_tree1 <- raster::crop(lc_tree1, spain_crop)
81   lc_tree1 <- raster::mask(lc_tree1, spain_crop)
82
83   lc_tree2 <- raster::raster('data/environmental_data/tree_cover2.tif')
84   lc_tree2 <- raster::crop(lc_tree2, spain_crop)
85   lc_tree2 <- raster::mask(lc_tree2, spain_crop)
86
87   lc_tree3 <- raster::raster('data/environmental_data/tree_cover3.tif')
88   lc_tree3 <- raster::crop(lc_tree3, spain_crop)
89   lc_tree3 <- raster::mask(lc_tree3, spain_crop)
90
91   lc_tree4 <- raster::raster('data/environmental_data/tree_cover4.tif')
92   lc_tree4 <- raster::crop(lc_tree4, spain_crop)
93   lc_tree4 <- raster::mask(lc_tree4, spain_crop)
94
95   lc_tree <- raster::merge(lc_tree1, lc_tree2, lc_tree3, lc_tree4)
96   raster::plot(lc_tree)
97
98   lc_tree_cover <- raster::resample(lc_tree, clim, method = 'bilinear')
99   names(lc_tree_cover) <- 'tree_cover'
100
101   raster::writeRaster(lc_tree_cover,
102                       'data/environmental_data/lc_tree.grd',
103                       format = 'raster',
104                       options = 'INTERLEAVE=BAND',
105                       overwrite = TRUE)
106   rm(lc_tree1, lc_tree2, lc_tree3, lc_tree4, lc_tree)
107 } else {
108
109   lc_tree_cover <- raster::raster('data/environmental_data/lc_tree.grd')

```

```

110 }
111
112
113 ### Herbaceous vegetation
114
115 if (!file.exists('data/environmental_data/lc_herbs.grd')) {
116
117   lc_herbs1 <- raster::raster(
118     'data/environmental_data/herbaceous_vegetation1.tif')
119   lc_herbs1 <- raster::crop(lc_herbs1, spain_crop)
120   lc_herbs1 <- raster::mask(lc_herbs1, spain_crop)
121
122   lc_herbs2 <- raster::raster(
123     'data/environmental_data/herbaceous_vegetation2.tif')
124   lc_herbs2 <- raster::crop(lc_herbs2, spain_crop)
125   lc_herbs2 <- raster::mask(lc_herbs2, spain_crop)
126
127   lc_herbs3 <- raster::raster(
128     'data/environmental_data/herbaceous_vegetation3.tif')
129   lc_herbs3 <- raster::crop(lc_herbs3, spain_crop)
130   lc_herbs3 <- raster::mask(lc_herbs3, spain_crop)
131
132   lc_herbs4 <- raster::raster(
133     'data/environmental_data/herbaceous_vegetation4.tif')
134   lc_herbs4 <- raster::crop(lc_herbs4, spain_crop)
135   lc_herbs4 <- raster::mask(lc_herbs4, spain_crop)
136
137   lc_herbs <- raster::merge(lc_herbs1, lc_herbs2,
138                             lc_herbs3, lc_herbs4)
139
140   # raster::plot(lc_herbs)
141
142   lc_herb_cover <- raster::resample(lc_herbs,
143                                     clim,
144                                     method = 'bilinear')
145   names(lc_herb_cover) <- 'grass_cover'
146
147   raster::writeRaster(lc_herb_cover,
148                       'data/environmental_data/lc_herbs.grd',

```



```

149         format = 'raster',
150         options = 'INTERLEAVE=BAND',
151         overwrite = TRUE)
152
153     rm(lc_herbs1, lc_herbs2, lc_herbs3, lc_herbs4, lc_herbs)
154 } else {
155
156     lc_herb_cover <- raster::raster('data/environmental_data/lc_herbs.grd')
157 }
158
159
160 ### Bare soil
161
162 if (!file.exists('data/environmental_data/lc_bare_soil.grd')) {
163     lc_bare1 <- raster::raster('data/environmental_data/bare_soil1.tif')
164     lc_bare1 <- raster::crop(lc_bare1, spain_crop)
165     lc_bare1 <- raster::mask(lc_bare1, spain_crop)
166
167     lc_bare2 <- raster::raster('data/environmental_data/bare_soil2.tif')
168     lc_bare2 <- raster::crop(lc_bare2, spain_crop)
169     lc_bare2 <- raster::mask(lc_bare2, spain_crop)
170
171     lc_bare3 <- raster::raster('data/environmental_data/bare_soil3.tif')
172     lc_bare3 <- raster::crop(lc_bare3, spain_crop)
173     lc_bare3 <- raster::mask(lc_bare3, spain_crop)
174
175     lc_bare4 <- raster::raster('data/environmental_data/bare_soil4.tif')
176     lc_bare4 <- raster::crop(lc_bare4, spain_crop)
177     lc_bare4 <- raster::mask(lc_bare4, spain_crop)
178
179     lc_bare <- raster::merge(lc_bare1, lc_bare2, lc_bare3, lc_bare4)
180
181     # raster::plot(lc_bare)
182
183     lc_bare_soil <- raster::resample(lc_bare, clim, method = 'bilinear')
184     names(lc_bare_soil) <- 'bare_soil'
185
186     raster::writeRaster(lc_bare_soil,
187                         'data/environmental_data/lc_bare_soil.grd',

```

```

188         format = 'raster',
189         options = 'INTERLEAVE=BAND',
190         overwrite = TRUE)
191
192     rm(lc_bare1, lc_bare2, lc_bare3, lc_bare4, lc_bare)
193 }else{
194
195     lc_bare_soil <- raster::raster('data/environmental_data/lc_bare_soil.grd')
196 }
197
198
199
200 # Calculate distance to closest river or lake -----
201
202
203 if (!file.exists('data/environmental_data/distance_to_water.grd')) {
204     water <- sf::st_read('data/environmental_data/clc2018_vector/clc2018.gpkg',
205                         query = "SELECT * FROM clc2018
206                                WHERE Code_18 == 511
207                                OR Code_18 == 512")
208
209     plot(sf::st_geometry(water), axes = TRUE)
210
211
212 ## load only the first
213     clim <- raster::raster('data/environmental_data/clim.grd')
214
215
216
217
218     raster_points <- as(clim, "SpatialPoints")
219     water_poly <- as(water, "Spatial")
220
221     water_poly@proj4string
222     raster_points@proj4string
223
224
225
226     crs1 <- sp::CRS('+proj=laea

```

```

227         +lat_0=52
228         +lon_0=10
229         +x_0=4321000
230         +y_0=3210000
231         +ellps=GRS80
232         +units=m
233         +datum=WGS84 +no_defs')
234
235 raster_points_transformed <- sp::spTransform(raster_points, crs1)
236 plot(sf::st_as_sf(raster_points_transformed), cex=0.2, pch=15, axes=T)
237
238
239 water_poly_transformed <- sp::spTransform(water_poly, crs1)
240 plot(sf::st_as_sf(water_poly_transformed), axes = TRUE)
241
242
243 dist1 <- rgeos::gDistance(raster_points_transformed[1:10000,],
244                           water_poly_transformed,
245                           byid=T)
246
247 dist2 <- rgeos::gDistance(raster_points_transformed[10001:20000,],
248                           water_poly_transformed,
249                           byid=T)
250
251 dist3 <- rgeos::gDistance(raster_points_transformed[20001:30220,],
252                           water_poly_transformed,
253                           byid=T)
254
255
256 min_distances <- c(apply(dist1,2,min),apply(dist2,2,min),apply(dist3,2,min))
257 data.frame(min_distances)
258
259 raster_points_df <- sp::SpatialPointsDataFrame(raster_points,
260                                                data=data.frame(min_distances))
261
262
263 crs2 <- raster::projection(clim)
264 raster_points_df_backtransformed <- sp::spTransform(raster_points_df,
265                                                    crs2)

```

```

266
267 dist_raster <- raster::rasterFromXYZ(raster_points_df_backtransformed)
268
269 #raster::projection(dist_raster)
270 #raster::projection(clim)
271 #raster::plot(dist_raster)
272 #raster::plot(clim)
273
274 names(dist_raster) <- 'distance_to_water'
275 raster::writeRaster(dist_raster,
276                     'data/environmental_data/distance_to_water.grd',
277                     format = 'raster',
278                     options = 'INTERLEAVE=BAND',
279                     overwrite = TRUE)
280
281 distance_to_water <- dist_raster
282 }else {
283   distance_to_water <- raster::raster(
284     'data/environmental_data/distance_to_water.grd')
285 }
286
287
288
289 # Calculate distance to closest landfill -----
290
291 if (!file.exists('data/environmental_data/distance_to_landfill.grd')) {
292   landfills <- sf::st_read(
293     'data/environmental_data/clc2018_vector/clc2018.gpkg',
294     query = "SELECT * FROM clc2018
295     WHERE Code_18 == 132")
296
297   plot(sf::st_geometry(landfills), axes = TRUE)
298   landfills
299
300   ## load only the first
301   clim <- raster::raster('data/environmental_data/clim.grd')
302
303
304

```

```

305
306 raster_points <- as(clim,"SpatialPoints")
307 landfills_poly <- as(landfills, "Spatial")
308
309 landfills_poly@proj4string
310 raster_points@proj4string
311
312
313
314 crs1 <- sp::CRS('+proj=laea
315                 +lat_0=52
316                 +lon_0=10
317                 +x_0=4321000
318                 +y_0=3210000
319                 +ellps=GRS80
320                 +units=m
321                 +datum=WGS84 +no_defs')
322
323 raster_points_transformed <- sp::spTransform(raster_points, crs1)
324 plot(sf::st_as_sf(raster_points_transformed), cex=0.2, pch=15, axes=T)
325
326
327 landfills_poly_transformed <- sp::spTransform(landfills_poly, crs1)
328 plot(sf::st_as_sf(landfills_poly_transformed), axes = TRUE)
329
330
331 dist1 <- rgeos::gDistance(raster_points_transformed[1:10000,],
332                           landfills_poly_transformed,
333                           byid=T)
334
335 dist2 <- rgeos::gDistance(raster_points_transformed[10001:20000,],
336                           landfills_poly_transformed,
337                           byid=T)
338
339 dist3 <- rgeos::gDistance(raster_points_transformed[20001:30220,],
340                           landfills_poly_transformed,
341                           byid=T)
342
343

```

```

344
345
346 min_distances <- c(apply(dist1,2,min),
347                     apply(dist2,2,min),
348                     apply(dist3,2,min))
349 data.frame(min_distances)
350
351 raster_points_df <- sp::SpatialPointsDataFrame(
352   raster_points,
353   data=data.frame(min_distances))
354
355
356 crs2 <- raster::projection(clim)
357 raster_points_df_backtransformed <- sp::spTransform(raster_points_df,
358                                                    crs2)
359
360 dist_raster <- raster::rasterFromXYZ(raster_points_df_backtransformed)
361
362
363 #raster::projection(dist_raster)
364 #raster::projection(clim)
365 #raster::plot(log(dist_raster+1))
366 #raster::plot(clim)
367
368 names(dist_raster) <- 'distance_to_landfill'
369
370 raster::writeRaster(dist_raster,
371                    'data/environmental_data/distance_to_landfill.grd',
372                    format = 'raster',
373                    options = 'INTERLEAVE=BAND',
374                    overwrite = TRUE)
375
376 distance_to_landfill <- dist_raster
377 } else {
378   distance_to_landfill <- raster::raster(
379     'data/environmental_data/distance_to_landfill.grd')
380 }
381
382

```

```

383
384 # Save all site covariates as one raster stack -----
385
386 variables <- raster::stack(clim,
387                             lc_tree_cover,
388                             lc_herb_cover,
389                             lc_bare_soil,
390                             distance_to_water,
391                             distance_to_landfill)
392
393
394
395 raster::writeRaster(variables,
396                     'data/environmental_data/variables_spain.grd',
397                     format = 'raster',
398                     options = 'INTERLEAVE=BAND',
399                     overwrite = TRUE)
400
401 # rm(clim, lc_tree_cover, lc_herb_cover, lc_bare_soil,
402 #     distance_to_landfill, distance_to_water,
403 #     spain, spain_crop, variables)
404
405
406
407
408 # Changes in temperature and precipitation -----
409 prec_change <- raster::raster(
410     'data/environmental_data/climate_change/precipitation.tiff')
411 temp_change <- raster::raster(
412     'data/environmental_data/climate_change/temperature.tiff')
413
414
415 ### Precipitation
416 prec_change <- raster::crop(prec_change, raster::extent(c(-12,5,30,50)))
417
418
419 #### edge length of one raster cell in km
420 prec_change_crs_m <- raster::projectRaster(
421     prec_change,

```

```

422   crs = '+proj=laea +lat_0=52 +lon_0=10 +x_0=4321000
423   +y_0=3210000 +ellps=GRS80 +units=m +no_defs')
424   poly1 <- raster::rasterToPolygons(prec_change_crs_m)
425   sqrt(raster::area(poly1[1,])) / 1000
426
427   prec_change_rs <- raster::resample(prec_change, clim, method = 'bilinear')
428   prec_change_rs <- raster::crop(prec_change_rs, spain_crop)
429   prec_change_rs <- raster::mask(prec_change_rs, spain_crop)
430
431   ### Temperature
432   temp_change <- raster::crop(temp_change, raster::extent(c(-12,5,30,50)))
433
434   temp_change_rs <- raster::resample(temp_change, clim, method = 'bilinear')
435   temp_change_rs <- raster::crop(temp_change_rs, spain_crop)
436   temp_change_rs <- raster::mask(temp_change_rs, spain_crop)
437
438
439
440   # Plot the changes in a map -----
441   pdf('results/climate_change.pdf', width=15, height=7)
442   par(mfrow=c(1,2),
443       oma = c(3, 4, 1, 2) + 0.1,
444       mar = c(0, 4, 4, 2) + 0.1)
445   pal1 = colorRampPalette(c('white', 'orange', 'red'))
446   raster::plot(temp_change_rs, main='Temperature change',
447               col = pal1(10),
448               legend.args = list(text = '°C', side = 3,
449                                   font = 2, line = 1, cex = 0.8))
450
451   pal2 = colorRampPalette(c('red', 'orange', 'yellow'))
452   raster::plot(prec_change_rs, main='Change in\annual precipitation',
453               col = pal2(30),
454               legend.args = list(text = '%', side = 3,
455                                   font = 2, line = 1, cex = 0.8))
456
457   dev.off()
458   par(mfrow=c(1,1))
459
460

```



```

461 # Save file as csv -----
462 points <- raster::rasterToPoints(raster::brick(temp_change_rs,
463                                             prec_change_rs))
464 change <- data.frame(points)
465
466 # avoid precision loss when saving the data frame
467 change$x <- sprintf("%.20f", change$x)
468 change$y <- sprintf("%.20f", change$y)
469
470 write.csv(change, "results/change_temp_prec.csv", row.names = FALSE)
471
472
473 # Clean up -----
474 rm(list = ls())

```

```

1 #####
2 # Second script
3 #
4 #     1. Join the site covariates with the eBird data set
5 #     2. Standardize the site covariates
6 #     3. Do spatial subsampling
7 #     4. Store the data frame in the unmarked format
8 #
9 #####
10
11
12 # Loading packages -----
13 library(dplyr)
14
15
16 # Load data -----
17
18 ### Site covariates
19 variables <- raster::stack('data/environmental_data/variables_spain.grd')
20
21 ### Bird data
22 milmig <- readr::read_csv('data/milmig.csv')
23
24 # filter for observations from the mainland of spain
25 # milmig <- milmig %>%
26 #   filter(!state_code=='ES-CN')
27
28
29 # Join eBird data and site covariates -----
30 occ_var <- milmig %>%
31   cbind(as.data.frame(
32     raster::extract(variables,
33                       milmig[,c('longitude', 'latitude')],
34                       cellnumbers=T))) %>%
35   tidyr::drop_na(bio1, tree_cover)
36
37 ## save a data frame with not standardized covariates
38 write.csv(occ_var, "results/milmig_not_std.csv", row.names = FALSE)
39

```

```

40
41 # Standardize site covariates -----
42
43 occ_var_std <- occ_var%>%
44   dplyr::mutate_at(c('bio1', 'bio2', 'bio3', 'bio4', 'bio5', 'bio6',
45                     'bio7', 'bio8', 'bio9', 'bio10', 'bio11', 'bio12',
46                     'bio13', 'bio14', 'bio15', 'bio16', 'bio17',
47                     'bio18', 'bio19',
48                     'tree_cover', 'grass_cover', 'bare_soil',
49                     'distance_to_water', 'distance_to_landfill'),
50   ~(scale(.) %>% as.vector))
51
52
53 # Convert data frame to unmarked format -----
54 occ_wide <- auk::format_unmarked_occu(
55   occ_var_std, site_id = 'site',
56   response = 'species_observed',
57   site_covs = c('cells', 'n_observations', 'latitude', 'longitude',
58                 'bio1', 'bio2', 'bio3', 'bio4', 'bio5', 'bio6',
59                 'bio7', 'bio8', 'bio9', 'bio10', 'bio11', 'bio12',
60                 'bio13', 'bio14', 'bio15', 'bio16', 'bio17',
61                 'bio18', 'bio19',
62                 'tree_cover', 'grass_cover', 'bare_soil',
63                 'distance_to_water', 'distance_to_landfill'),
64   obs_covs = c('time_observations_started', 'duration_minutes',
65                'effort_distance_km', 'number_observers', 'protocol_type',
66                'day_of_year'))
67
68
69 ### Convert the detection histories in 1=# presence/ 0= absence
70 # instead of TRUE/FALSE
71 cols <- sapply(occ_wide, is.logical)
72 occ_wide[, cols] <- lapply(occ_wide[, cols], as.numeric)
73
74
75 # Do spatial subsampling -----
76 # We can only have one (3 - 10 times repeated) observation per one grid cell!
77 occ_wide_clean <- occ_wide[!duplicated(occ_wide$cells),]
78

```

```
79 # Part that is removed:
80 1- nrow(occ_wide_clean)/nrow(occ_wide)
81
82
83 # Save unmarked data as csv -----
84 write.csv(occ_wide_clean, "results/milmig.csv", row.names = FALSE)
85
86
87 # Clean up -----
88 rm(list = ls())
```

```

1 #####
2 # Third script
3 #
4 # Selection of covariates:
5 #     ... are based on ecological hypotheses and from
6 #     from the explanatory analysis with a Random
7 #     Forest model
8 #
9 #     --> here some covariates are excluded because of
10 #     collinearity
11 #
12 #####
13
14
15 # Load data -----
16 occ_wide_clean <- read.csv("results/milmig.csv")
17
18
19 # Selected covariates -----
20 random_forest_selection <- c('bio3', 'bio4', 'bio7', 'bio5')
21
22 ecological_selection <- c('bio1', 'bio12',
23                           'tree_cover', 'grass_cover', 'bare_soil',
24                           'distance_to_landfill', 'distance_to_water')
25
26 selected_covariates <- c(random_forest_selection, ecological_selection)
27 selection <- occ_wide_clean[, selected_covariates]
28
29
30 # Exclude some covariates -----
31 selected_names <- names(selection)[! names(selection) %in% c('bio7', 'bio4', 'bio5')]
32
33
34 # Make a cluster dendrogram -----
35 ## cluster with all selected covariates
36 cor1 <- abs(as.dist(cor(selection)))
37 clust1 <- hclust(1- cor1)
38 plot(clust1)
39

```

```

40 ## cluster with some covariates removed
41 cor1 <- abs(as.dist(cor(selection[, selected_names])))
42 clust1 <- hclust(1- cor1)
43 plot(clust1)
44
45
46 # Correlation plots -----
47 ## Correlation plots with all covariates
48 psych::pairs.panels(selection)
49
50 ## Correlation plots with some covariates removed
51 psych::pairs.panels(selection[, selected_names])
52
53
54 # Test for collinearity -----
55 ## Test for collinearity with all covariates
56 HH::vif(selection)
57
58 ## Test for collinearity with some covariates removed
59 HH::vif(selection[, selected_names])
60
61
62 # Make maps of the final selected covariates -----
63 site_covariates <- raster::brick('data/environmental_data/variables_spain.grd')
64 site_covariates <- site_covariates[[selected_names]]
65
66 pdf('results/site_covs.pdf')
67 par(mfrow=c(1,1),
68     oma = c(0, 0, 0, 1) + 0.1,
69     mar = c(0, 4, 10, 2) + 0.1)
70 raster::plot(site_covariates, main=c('Isothermality',
71                                     'Annual mean\ntemperature °C *10',
72                                     'Annual mean\nprecipitation mm',
73                                     'Tree cover %',
74                                     'Grass cover %',
75                                     'Bare soil cover %',
76                                     'Distance to closest\nlandfill m',
77                                     'Distance to closest\nriver or lake m'))
78 dev.off()

```

```
79
80
81
82 # Clean up -----
83 rm(list = ls())
84 dev.off(dev.list()["RStudioGD"])
```

---

```

1 #####
2 # Fourth script
3 #
4 # Build models with the unmarked package
5 #
6 # --> Compare different models with
7 #     information criterion
8 #
9 # --> evaluate the best model
10 #
11 # --> build an average model
12 #
13 #####
14
15 #
16 setwd("/home/felix/Dokumente/studium/Potsdam/Module/Biogeography/Black kite/R")
17
18 # Load packages -----
19 library(dplyr)
20 library(ggplot2)
21 library(unmarked)
22
23 # Load data -----
24 occ_um <- formatWide(read.csv("results/milmig.csv"), type = "unmarkedFrameOccu")
25 summary(occ_um)
26
27
28 # Build models -----
29 ## Null model
30 occ_null <- occu(~ 1 ~ 1, occ_um)
31 summary(occ_null)
32 backTransform(occ_null, "state")
33
34 ## Model only with detection covariates
35 detection_cov_model <- occu(~ duration_minutes
36                             + effort_distance_km
37                             ~ 1, data=occ_um)
38 summary(detection_cov_model)
39

```



```

40
41 ## Model only with site covariates
42 site_cov_model <- occu(~ 1
43                        ~ poly(bio1, 2)
44                        + poly(bio3, 2)
45                        + poly(bio12, 2)
46                        + poly(tree_cover, 2)
47                        + poly(grass_cover, 2)
48                        + poly(bare_soil, 2)
49                        + poly(distance_to_water, 2)
50                        + poly(distance_to_landfill, 2)
51                        , data = occ_um)
52 summary(site_cov_model)
53
54
55
56
57 ## Full model with covariates
58 full_model <- occu(~ duration_minutes
59                  + number_observers
60                  ~ poly(bio1, 2)
61                  + poly(bio3, 2)
62                  + poly(bio12, 2)
63                  + poly(tree_cover, 2)
64                  + poly(grass_cover, 2)
65                  + poly(bare_soil, 2)
66                  + poly(distance_to_water, 2)
67                  + poly(distance_to_landfill, 2), data = occ_um)
68 summary(full_model)
69
70 re <- ranef(full_model)
71 sum(bup(re, stat="mode"))
72 sum(bup(re, stat="mean"))
73
74
75 # Model selection -----
76 ## Model selection with AIC
77 models_list <- list(NULL = occ_null,
78                    detection = detection_cov_model,

```

```

79         site = site_cov_model,
80         full_model = full_model)
81
82 un_models <- fitList(fits = models_list)
83 ModSelect <- modSel(un_models, nullmod = "Null")
84 ModSelect
85
86
87 ## Model selection with AICc
88 AICcmmodavg::aictab(models_list, second.ord = T)
89
90
91 best_model <- full_model
92
93
94
95 if (!file.exists('models.rda')) {
96
97   # Goodness of fit test of best model -----
98   GOF <- parboot(best_model, nsim=500, ncores=8, report=T)
99   GOF
100
101   cHat <- GOF@t0 / mean(GOF@t.star)
102   cHat
103
104   ### Another goodness of fit test
105   AICcmmodavg::mb.gof.test(best_model,
106                             nsim=500,
107                             plot.hist = F,
108                             parallel=T,
109                             ncores=8)
110
111
112
113   ### QAICc
114   GOF1 <- AICcmmodavg::aictab(models_list, c.hat = 1)
115   # --> it is the same as above, because the cHat value is below one
116
117

```

```

118 # Build an average model -----
119 ## Get the names of the detection covariates
120 det_terms <- MuMIn::getAllTerms(best_model) %>%
121   purrr::discard(stringr::str_detect, pattern = "psi")
122
123 ## Get combination of models, detection covariates are always present
124 occ_dredge <- MuMIn::dredge(best_model, fixed = det_terms)
125
126 ## Get the best models from the model list
127 occ_dredge_95 <- MuMIn::get.models(occ_dredge,
128                                   subset = cumsum(weight) < 0.95)
129
130 ## Get the average model based on model weights
131 #occ_avg <- MuMIn::model.avg(occ_dredge, fit = TRUE, revised.var = TRUE)
132 occ_avg <- MuMIn::model.avg(occ_dredge_95, fit=T)
133
134 ## Calculate the AICc for the average model
135 sum(occ_avg$msTable$AICc * occ_avg$msTable$weight)
136
137 ## Model coefficients of the average model
138 t(occ_avg$coefficients)
139
140 MuMIn::importance(occ_avg)
141
142 save(occ_avg, best_model, GOF, cHat, GOF1, file='models.rda')
143 }
144
145 # Clean up -----
146 rm(list = ls())

```

```

1 #####
2 # Fifth script
3 #
4 #   Make all plots:
5 #       - Maps
6 #       - response of occupancy due to covariates
7 #
8 #####
9
10 library(dplyr)
11 library(ggplot2)
12
13 # Load data -----
14 load('models.rda')
15 change <- read.csv('results/change_temp_prec.csv')
16 variables <- raster::brick("data/environmental_data/variables_spain.grd")
17
18
19 # Prepare data for predictions -----
20 variables_selection <- c("bio1",
21                          "bio3",
22                          "bio12",
23                          "tree_cover",
24                          "grass_cover",
25                          "bare_soil",
26                          "distance_to_water",
27                          "distance_to_landfill")
28
29 variables.sel <- variables[[variables_selection]]
30
31 p_variables <- data.frame(raster::rasterToPoints(variables.sel) )
32 p_variables <- p_variables %>%
33   tidyr::drop_na(tree_cover, bio1)
34
35 change
36
37
38
39 change_joined <- p_variables %>%

```

```

40   left_join(change, by = c('x', 'y')) %>%
41   select(x,y, temperature, precipitation) %>%
42   mutate(temperature = temperature * 10,
43          precipitation = 1 + precipitation/100)
44
45   mean(change_joined$temperature, na.rm=T)/10
46   mean(change_joined$precipitation, na.rm=T)
47
48
49
50   p_variables_std <- p_variables %>%
51     mutate_at(variables_selection, ~(scale(.) %>% as.vector))
52
53   sd_bio1 <- sd(p_variables$bio1)
54   mean_bio1 <- mean(p_variables$bio1/ sd(p_variables$bio1))
55
56   sd_bio12 <- sd(p_variables$bio12)
57   mean_bio12 <- mean(p_variables$bio12/ sd(p_variables$bio12))
58
59   p_variables_std_future <- p_variables_std
60   p_variables_std_future$bio1 <-
61     (p_variables$bio1 + change_joined$temperature) / sd_bio1 - mean_bio1
62   p_variables_std_future$bio12 <-
63     (p_variables$bio12 * change_joined$precipitation) / sd_bio12 - mean_bio12
64
65   # Make predictions -----
66   # actual
67
68   # occ_avg, best_model
69   pred_actual <- unmarked::predict(occ_avg,
70                                   newdata = select(p_variables_std,
71                                                    -x, -y),
72                                   type = "state")
73
74   # Predicted, fit
75   # SE, se.fit
76   actual_climate <- bind_cols(p_variables_std,
77                               probability = pred_actual$fit,
78                               SE = pred_actual$se.fit) %>%

```

```

79   select(x, y, probability, SE) %>%
80   tidyr::pivot_longer(cols = c(probability, SE))
81
82
83   # future
84
85   # occ_avg, best_model
86   pred_future <- unmarked::predict(occ_avg,
87                                   newdata = select(p_variables_std_future,
88                                                    -x, -y),
89                                   type = "state")
90
91   # Predicted, fit
92   v
93   future_climate <- bind_cols(p_variables_std_future,
94                               probability = pred_future$fit,
95                               SE = pred_future$se.fit) %>%
96   select(x, y, probability, SE) %>%
97   tidyr::pivot_longer(cols = c(probability, SE))
98
99   # join the data, preparation for plotting
100  data <- actual_climate %>%
101    inner_join(future_climate, by = c("x", "y", "name")) %>%
102    rename(actual = value.x,
103           future = value.y,
104           type = name) %>%
105    tidyr::pivot_longer(cols=c(actual, future))
106
107  # results/predictions_best_model.csv, or results/predictions_avg_model.csv
108  write.csv(data, 'results/predictions_avg_model.csv', row.names = F)
109
110  ##### Plotting the map
111
112
113  # Plot the map -----
114
115  # or load data:
116  # data <- read.csv('results/predictions_best_model.csv')
117  # data <- read.csv('results/predictions_avg_model.csv')

```

```

118
119 data %>%
120   ggplot(aes(x,y, fill=value))+
121   geom_raster()+
122   scale_fill_viridis_c(name="value", option="turbo")+
123   theme(panel.border=element_rect(color="black",fill="transparent"),
124         text = element_text(size=20))+
125   labs(x="Longitude", y="Latitude")+
126   coord_fixed()+
127   facet_grid(~name ~ type)+
128   theme(text = element_text(size=30, family = "LM Roman 10"))
129 # 'results/best_model_map.png' or 'results/avg_model_map.png'
130 ggsave('results/avg_model_map.png', width = 16, height=10)
131
132
133 # Mean probabilities
134 data %>%
135   group_by(name) %>%
136   filter(type == 'probability') %>%
137   summarise(p = mean(value))
138
139 # sum of ells occupied
140 data %>%
141   group_by(name) %>%
142   filter(type == 'probability') %>%
143   mutate(occ = value >= 0.5) %>%
144   summarise(s = sum(occ))
145
146 data %>%
147   tidyr::pivot_wider(names_from=name, values_from = value) %>%
148   filter(type == 'probability') %>%
149   mutate(lower = actual > future) %>%
150   mutate(higher = actual < future) %>%
151   summarise(lower = sum(lower), higher = sum(higher))
152
153 # total grid cells
154 nrow(data) / 4
155
156 # Prediction of covariates -----

```

```

157
158 raw_data <- readr::read_csv('results/milmig_not_std.csv')
159 model_statistics <- readr::read_csv2('results/best_model.csv')
160
161 model_labels <- model_statistics %>%
162   rename(p = `P(>|z|)` ) %>%
163   tidyr::pivot_wider(values_from=p, names_from = coef_no, id_cols=name) %>%
164   filter(name != '-') %>%
165   rename(first = `1`,
166           second = `2`) %>%
167   mutate(first = ifelse( round(first, 2) == 0,
168                          '<0.01',
169                          paste('', round(first, 2))),
170          second = ifelse(round(second, 2) ==0,
171                          '<0.01',
172                          paste('', round(second, 2))),
173          p = paste('p = ',first, ', ', second, sep='')) %>%
174   select(name, p) %>%
175   mutate(name =
176           recode(name,
177                 bare_soil = "Bare soil cover (%)",
178                 tree_cover = "Tree cover (%)",
179                 grass_cover = "Grass cover (%)",
180                 bio1 = "Annual mean\ntemperature (°C)",
181                 bio3 = "Isothermality",
182                 bio12 = "Annual precipitation\n(mm)",
183                 distance_to_water = "Distance to closest\nriver or lake (km)",
184                 distance_to_landfill = "Distance to closest\nlandfill (km)"
185           )) %>%
186   mutate(x = c(10, 38, 800, 30, 30, 15, 25, 45),
187          y = rep(0.1, 8))
188
189 variable_names <- c("bio1",
190                    "bio3",
191                    "bio12",
192                    "bare_soil",
193                    "tree_cover",
194                    "grass_cover",
195                    "distance_to_water",

```



```

196         "distance_to_landfill")
197
198 rm(old_data)
199 for (i in seq_along(variable_names)) {
200     variable_str <- variable_names[i]
201     print(variable_str)
202
203
204     newdata <- setNames(data.frame(
205         matrix(ncol = length(variable_names),
206             nrow = 1000)),
207         variable_names)
208
209     newdata[, i] <- seq(min(raw_data[, variable_str]),
210         max(raw_data[, variable_str]),
211         length.out = 100)
212
213     newdata[is.na(newdata)] <- 0
214
215     sd1 <- sd(newdata[, variable_str])
216     mean1<- mean(newdata[, variable_str]/sd(newdata[, variable_str]))
217
218     newdata[, variable_str] <- as.numeric(scale(newdata[, variable_str]) )
219
220     # best_model, occ_avg
221     predict_newdataset <- unmarked::predict(occ_avg,
222         newdata = newdata,
223         type = "state")
224
225     # Predicted, fit
226     # SE, se.fit
227     plotting_data <- bind_cols(newdata,
228         occ_prob = predict_newdataset$fit,
229         occ_se = predict_newdataset$se.fit) %>%
230         select(matches(variable_str), occ_prob, occ_se)
231
232     plotting_data$x <- (plotting_data[, variable_str] + mean1)* sd1
233
234     if (variable_str %in% c('bio1', 'bio2')) {
235         plotting_data$x <- plotting_data$x / 10

```

```

235 } else if (variable_str == 'landfills'){
236   plotting_data$x <- plotting_data$x * 100
237 } else if (variable_str %in% c('distance_to_water',
238                               'distance_to_landfill')){
239   plotting_data$x <- plotting_data$x /1000
240 }
241
242 new_data <- plotting_data %>%
243   mutate(lower_se = occ_prob - occ_se,
244          upper_se = occ_prob + occ_se) %>%
245   select(x, occ_prob, lower_se, upper_se) %>%
246   mutate(name = variable_str)
247
248 if (i == 1){
249   old_data <- new_data
250 } else {
251   old_data <- old_data %>%
252     bind_rows(new_data)
253 }
254 }
255
256
257
258 old_data %>%
259   mutate(name = recode(name,
260                        bare_soil = "Bare soil cover (%)",
261                        tree_cover = "Tree cover (%)",
262                        grass_cover = "Grass cover (%)",
263                        bio1 = "Annual mean\ntemperature (°C)",
264                        bio3 = "Isothermality",
265                        bio12 = "Annual precipitation\n(mm)",
266                        distance_to_water = "Distance to closest\nriver or lake (km)",
267                        distance_to_landfill = "Distance to closest\nlandfill (km)")) %>%
268   #filter(name != "Distance to closest\nlandfill (km)") %>%
269   ggplot()+
270   geom_text(data = model_labels,
271            aes(x = x, y = y, label = p),
272            family="LM Roman 10",
273            size=3.5)+

```

```

274 geom_ribbon(aes(ymin = lower_se,
275                 ymax = upper_se,
276                 x = x),
277             fill="gray", alpha=0.7) +
278 geom_line(aes(x=x, y=occ_prob, color=factor(name)),
279           size=0.8)+
280 scale_color_manual(values =
281                     c("Bare soil cover (%)" = "grey36",
282                       "Annual mean\ntemperature (°C)" = "orangered3",
283                       "Distance to closest\nriver or lake (km)" = "royalblue2",
284                       "Distance to closest\nlandfill (km)" = "chocolate4",
285                       "Tree cover (%)" = "palegreen3",
286                       "Grass cover (%)" = "chartreuse4",
287                       "Annual precipitation\n(mm)" = "steelblue4",
288                       "Isothermality" = "orange"))+
289 labs(x=NULL,
290      y="Occupancy Probability")+
291 theme_bw()+
292 theme(legend.position = "none",
293       panel.border = element_rect(color="black",fill="transparent"),
294       text = element_text(size=10, family="LM Roman 10"),
295       plot.margin=unit(c(2, 5, -5, 2), "points"),
296       panel.spacing = unit(0.8, "lines"),
297       strip.background = element_blank(),
298       strip.placement = "outside",
299       strip.text = element_text(size=12, face='plain',
300                                margin = margin(t = 0, r = 0, b = 10, l = 0)),
301       axis.title = element_text(size=12, face='plain'))+
302 scale_x_continuous(expand = expansion(mult = c(0, 0))) +
303 scale_y_continuous(limits = c(0,1),expand = expansion(mult = c(0.03, 0.03))) +
304 facet_wrap(. ~ name, scales="free_x", strip.position = 'bottom', ncol=4)
305 # 'results/avg_model_site_covariates.png', 'results/site_covariates.png'
306 ggsave('results/avg_model_site_covariates.png', width = 7, height = 5)
307
308
309 ##### detection covariates
310
311 summary(raw_data$duration_minutes)
312 summary(raw_data$number_observers)

```

```

313
314 variable_names <- c('number_observers', 'duration_minutes')
315 newdata = setNames(data.frame(
316   matrix(ncol = length(variable_names),
317     nrow = 2000)),
318   variable_names)
319 newdata[1:1000, variable_names[1]] <- seq(1, 5, length.out=1000)
320 newdata[1:1000, variable_names[2]] <- colMeans(raw_data[, variable_names[2]],
321   na.rm=T)
322
323 newdata[1001:2000, variable_names[2]] <- seq(1, 300, length.out=1000)
324 newdata[1001:2000, variable_names[1]] <- colMeans(raw_data[, variable_names[1]],
325   na.rm=T)
326
327 best_model
328
329 predict_labels <- data.frame(
330   name = c('Number of observers', 'Duration of observation (min)'),
331   p = c('p = 0.22', 'p = 0.01'),
332   x = c(3, 150),
333   y = c(0.1, 0.1))
334
335
336 predict_newdataset <- unmarked::predict(best_model,
337   newdata = newdata,
338   type = "det")
339
340 plotting_data <- bind_cols(newdata,
341   occ_prob = predict_newdataset$Predicted,
342   occ_se = predict_newdataset$SE) %>%
343   mutate(lower_se = occ_prob - occ_se,
344     upper_se = occ_prob + occ_se,
345     name = c(rep('Number of observers', 1000),
346       rep('Duration of observation (min)', 1000)))
347
348 plotting_data$x <- c(plotting_data[1:1000, 'number_observers'],
349   plotting_data[1001:2000, 'duration_minutes'])
350
351 plotting_data %>%

```

```

352 ggplot()+
353 geom_text(data = predict_labels,
354           aes(x = x, y = y, label = p),
355           family="LM Roman 10", size=5)+
356 geom_ribbon(aes(ymin = lower_se,
357               ymax = upper_se,
358               x = x),
359             fill="gray", alpha=0.7) +
360 geom_line(aes(x=x, y=occ_prob, color=factor(name)),
361           size=0.8)+
362 scale_color_manual(values =
363                   c("Number of observers" = "orange",
364                     "Duration of observation (min)" = "orangered3"))+
365 labs(x=NULL,
366      y="Occupancy Probability")+
367 theme_bw()+
368 theme(legend.position = "none",
369       panel.border = element_rect(color="black",fill="transparent"),
370       text = element_text(size=10, family="LM Roman 10"),
371       plot.margin=unit(c(2, 5, -5, 2), "points"),
372       panel.spacing = unit(0.8, "lines"),
373       strip.background = element_blank(),
374       strip.placement = "outside",
375       strip.text = element_text(size=12, face='plain',
376                                margin = margin(t = 0, r = 0,
377                                                b = 10, l = 0)),
378       axis.title = element_text(size=12, face='plain'))+
379 scale_x_continuous(expand = expansion(mult = c(0, 0))) +
380 scale_y_continuous(limits = c(0,1), expand =
381                     expansion(mult = c(0.03, 0.03))) +
382 facet_wrap(. ~ name, scales="free_x", strip.position = 'bottom', ncol=4)
383 ggsave('results/det_covariates.png', width = 6, height = 3)

```

Python script for running the Random Forest model:

```
1 from sklearn import model_selection, preprocessing
2 from imblearn import ensemble
3 import pandas as pd
4 import numpy as np
5 import matplotlib.pyplot as plt
6
7 plt.rcParams.update({
8     "font.family": "sans-serif",
9     "font.sans-serif": ["Latin Modern Roman"]})
10
11 data = pd.read_csv('milmig.csv')
12
13 ## observational covariates
14 observational_covariates = data.loc[:, 'time_observations_started.1':'day_of_year.10']
15 observational_covariates.loc[:, 'site'] = data.site
16 stubnames = ['time_observations_started.', 'duration_minutes.', 'effort_distance_km.',
17             'number_observers.', 'day_of_year.', 'protocol_type.']
18 obs_covariates_long = pd.wide_to_long(observational_covariates,
19                                     stubnames,
20                                     i='site',
21                                     j='observation')
22 obs_covariates_long = obs_covariates_long.rename(
23     columns={"time_observations_started.": "daytime",
24            "duration_minutes.": "duration",
25            "effort_distance_km.": "distance_km",
26            "number_observers.": "n_observers",
27            "day_of_year.": "day_of_year",
28            "protocol_type.": "protocol_type"})
29 obs_covariates_long['protocol_type'] = preprocessing.LabelEncoder().fit_transform(
30     obs_covariates_long['protocol_type'])
31 sc = preprocessing.StandardScaler(with_mean=False, with_std=False)
32 obs_covariates_transformed = sc.fit_transform(obs_covariates_long)
33
34
35 ## site covariates
36 site_covariates = data.loc[:, 'bio1':'marshes']
37 site_covariates_long = site_covariates.loc[
```

```

38     np.tile(np.arange(0, len(site_covariates)), 10)]
39
40
41
42 ## join covariates to one table
43 covs = np.concatenate((site_covariates_long.values,
44                         obs_covariates_transformed),
45                         axis=1)
46 column_names = np.append(site_covariates_long.columns,
47                           obs_covariates_long.columns)
48
49
50 ## detection
51 observed = pd.melt(data.loc[:, 'y.1':'y.10'])
52
53 ## simulation
54 n_sim = 500
55
56 feature_imp = []
57 pred_data_all = []
58 for i in range(n_sim):
59
60     random_numbers = np.random.randint(1, data.n_observations)
61
62     y = observed.iloc[random_numbers, 1]
63     x = covs[random_numbers, :]
64
65     x_train, x_test, y_train, y_test = model_selection.train_test_split(
66         x, y, test_size=0.0001)
67     clf = ensemble.BalancedRandomForestClassifier()
68
69     clf.fit(x_train, y_train)
70     feature_imp.append(clf.feature_importances_)
71
72     predict_data_list = []
73     for name in column_names:
74         mean_x = np.mean(x, axis=0)
75         dummy_data = np.tile(mean_x[:, None], 50)
76         covariate = np.linspace(-2, 2, 50)

```

```

77     dummy_data[column_names == name] = covariate
78     pred = clf.predict_proba(dummy_data.T)
79     predict_data_list.append(pred[:, 1])
80
81     pred_data_all.append(predict_data_list)
82
83
84 feature_imp = np.array(feature_imp)
85 importance = np.mean(feature_imp.T, axis=1)
86 importance_sd = np.std(feature_imp.T, axis=1) / np.sqrt(n_sim)
87
88 proper_names = {"bio1": "Annual Mean Temperature",
89                 "bio2": "Mean Diurnal Range",
90                 "bio3": "Isothermality",
91                 "bio4": "Temperature Seasonality",
92                 "bio5": "Max Temperature of Warmest Month",
93                 "bio7": "Temperature Annual Range",
94                 "bio9": "Mean Temperature of Driest Quarter",
95                 "bio10": "Mean Temperature of Warmest Quarter",
96                 "bio12": "Annual Precipitation",
97                 "bio17": "Precipitation of Driest Quarter",
98                 "bio18": "Precipitation of Warmest Quarter",
99                 "distance_to_water": "Distance to closest river or lake",
100                "distance_to_landfill": "Distance to closest landfill",
101                "n_observers": "Number of observers",
102                "duration": "Duration of observation"}
103
104 for key, val in zip(proper_names.keys(), proper_names.values()):
105     column_names[key == column_names] = val
106
107 filter_index = importance > 0.02
108 selected_columns = column_names[filter_index]
109 selected_columns
110
111 plt.barh(np.arange(len(selected_columns)), np.sort(importance[filter_index]),
112          xerr=importance_sd[filter_index][np.argsort(importance[filter_index])],
113          align='center', color='#21918C', alpha=0.7)
114 plt.yticks(np.arange(len(importance[filter_index])),
115            selected_columns[np.argsort(importance[filter_index])], size=12)

```



```

116 plt.xlabel('Importance of covariates', size=12)
117 plt.tight_layout()
118
119 plt.savefig('importance_features.png', dpi=300)
120 plt.show()
121
122
123
124 #####
125 pred_data_all = np.array(pred_data_all)
126 plt.figure(figsize=(6, 6))
127
128 for i, name in enumerate(
129     selected_columns[np.argsort(importance[filter_index])][::-1]):
130     if i < 4:
131         plt.subplot(2,2, i+1)
132         data = pred_data_all[:, column_names == name, :]
133         y_pred = np.mean(data, axis=1)[1]
134
135         plt.plot(np.linspace(-2, 2, 50), y_pred, '.-k')
136         plt.title(name)
137 plt.tight_layout()
138 plt.savefig('response.png', dpi=300)
139 plt.show()

```