

QUICK START GUIDE

Step-by-step instructions to run the analysis

INITIAL SETTINGS

Download R and RStudio

Ensure that you have R version 4.4.1 or higher installed on your computer. The necessary software for this analysis can be found at:

- R: <https://cran.r-project.org/>
- RStudio: <https://posit.co/downloads/>
- Rtools: <https://cran.r-project.org/bin/windows/Rtools/rtools44/rtools.html>

Open R project

- On your computer, navigate to your working directory that contains the R project and initial files. Then, click on the R project named *KoalaModellingAnalysis.Rproj*.
- Navigate to *File*, click *Open file...*, go to the code folder within your working directory, select *setup_script.R*, and click *Open*. Once the script is open, click *Source* to download the latest code from GitHub (<https://github.com/seq-koala-monitoring/stats-model/tree/main>) and configure the working directory to ensure that the analyses run properly.

Parameters

We designed this project to give users flexibility to adjust parameters while still achieving reliable results. If you need to modify any spatial parameter from the default, adjust the arguments in the *parameters_init.txt* file.

The parameters are:

- **primary_grid_size**: A numeric value representing the spatial dimensions (in metres) of each cell in the analysis and mapping grid.
`primary_grid_size <- 500`
- **secondary_grid_multiple**: A numeric value that multiplies the primary grid size

to create a coarser secondary grid.

```
secondary_grid_multiple <- 10
```

- **line_transect_buffer:** A numeric value (in metres) for the estimated width of line transects (i.e., one side from the line transect).

```
line_transect_buffer <- 28.7
```

- **cov_impute_buffer:** A numeric value (in metres) for the buffer around areas for covariate imputation.

```
cov_impute_buffer <- 0
```

- **area_buffer:** A numeric value (in metres) defining the buffer around the study area.

```
area_buffer <- 0
```

- **genetic_units:** The filename (with extension and within quotation marks) of the spatial layer containing boundaries of genetic populations.

```
genetic_units <- "Population_boundaries_v2.shp"
```

- **gen_pop_column_id:** A string (within quotation marks) specifying the column name that identifies each genetic unit in the spatial layer.

```
gen_pop_column_id <- "GENPOP_ID"
```

- whether to aggregate the **genetic population** into three regions or not

```
gen_pop_agg <- TRUE
```

- **update_database:** Should the koala survey database be updated with new data? This integrates new survey data into the modelling database by performing data integrity checks, matching the Site_ID with the spatial locations of survey sites, assigning a genetic unit (i.e., "genetic" population), estimating missing perpendicular distances, and formatting the data to meet modelling requirements. Optionally, the function can also incorporate the spatial representation of the new surveys into the existing spatial file containing all koala surveys (default is TRUE).

```
update_database <- TRUE
```

- specifies whether to **run data processing in parallel** (requires RStudio API if true)

```
use_parallel <- TRUE
```

- specify whether to **run the covariate extraction** algorithm; if not, read from disc in the output folder (computationally intensive, 30 minute run if TRUE)

```
run_cov_extraction <- TRUE
```

- specify whether to **use spatial data imputation** (default = FALSE - unlikely to be needed)

```
use_imputation <- FALSE
```

- list of **static predictor** variables to use,

- Variables removed by default due to high correlations (> 0.6 or < -0.6) were:
- elevation,
- terrain ruggedness index,
- persistent green,
- intensive land-use in a 2 km buffer,
- maximum temperature,

```
static_variables <- c("htslo", "hspc1", "hspc2", "hcltp", "hcltt", "hhgde")
```

- list of **time dynamic predictor** variables to use,

- Variables removed by default due to high correlations (> 0.6 or < -0.6) were:
- elevation,
- terrain ruggedness index,
- persistent green,
- intensive land-use in a 2 km buffer,
- maximum temperature,

```
dynamic_variables <- c("hhfwc", "hhpgr2km", "htpls2km", "hcpre", "hctmn",  
"hseas", "hhkha", "htlus")
```

- list of **observer error predictor** variables,

- Variables removed by default due to too much missing data were:
- weather,
- cloud cover,
- wind,
- canopy cover,
- subcanopy cover,

```
obs_variables <- c("hhcht", "hhunf", "hhchtunf", "hdtma", "hdpre")
```

- set a start date for modelling using the format dd/mm/yyyy (e.g. "01/01/2025"). If you want to model the entire survey period (i.e., from 1996 to current), leave it as NULL and the model will then start from the date of the first survey.
FirstDate <- NULL
- When extracting ObserverID as a predictor for observer error, **should observers be grouped** (from 2021 onwards)? This parameter lets the user choose between two options: (1) Using each observer ID as a variable, which may result in models that are easier to converge but require long computing times; (2) Grouping observers based on their affiliation, as specified in the group_observers.csv file located in input/group_observers/. This option reduces computing time but may lead to convergence issues in the models.
obs_groups <- FALSE
- set whether to **mask rainforest**,
RainMask <- TRUE

You also need to provide an API code that allows R to download a few soil variables from TERN. Please, follow the steps below:

1. Navigate to <https://portal.tern.org.au/browse/theme>
2. Click "Sign in" in the right-hand side
3. Sign in with an account
4. Once logged in, click your name and TERN account
5. Click API Keys > Create API Keys > Type a name > Request API key
6. Copy or save the API key in a secure location
7. Open the file *apis.R* in R or any text editor like Notepad. The file is located within the keys folder in your working directory
8. Replace the sequence of numbers and letters with your API code. Make sure to keep it within quotes

COVARIATES AND KOALA SURVEY DATA

Download covariates

The R script *covariate_processing.R* was designed to update the covariates required for the Bayesian state-space model used to estimate koala densities across

Southeast Queensland.

The basic workflow involves opening the file in R, checking if the input folder already has the latest covariate file. If it doesn't, the code will automatically download, process, and save the most recent file the correct location.

Before anything, start fresh by clicking Session > Restart R. After that, please select all lines by pressing Ctrl + A on a Windows PC or Command + A on a Mac. Then, run these lines by pressing Ctrl + Enter on a Windows PC or Command + Return on a Mac. Alternatively, you can click on Source near the top right corner of the script panel. This code may take anywhere from a few minutes to days to run, depending on how many files need updating and your computer's specifications.

When you see "THIS CODE HAS FINISHED" in the Console panel (usually at the bottom left), you're ready to start processing these covariates.

NOTE: Disregard any warnings on the taskbar about packages that are not installed

Covariates that are NOT downloaded automatically

Some covariate maps are not regularly updated by their data custodians. Before re-running this analysis with new survey data, users should manually check whether updated versions of these maps are available.

Pre-clear Koala Habitat Map and Remnant Vegetation Cover

Our final analysis used the *KoalaSurveyStrata_v3_ful* map directly provided by DETSI and the remnant vegetation cover based on the 2021 remnant regional ecosystem mapping (available at:

[\[https://qldspatial.information.qld.gov.au/catalogue/custom/detail.page?fid={F5CF90D6-5881-4D8F-9581-D8F55D25F9CE}}\]](https://qldspatial.information.qld.gov.au/catalogue/custom/detail.page?fid={F5CF90D6-5881-4D8F-9581-D8F55D25F9CE}})).

The user only needs to update it if a new version of the "pre-clear koala habitat map" or remnant vegetation map are produced/released.

To do so, follow the steps below:

1. Obtain the new version of the layers from its source.
2. Find the folder where the old version is saved. The location currently (June 2025) is: input/covariates/raw_data/preclean_koala_habitat/. Save the new file in this folder.
3. Open the file named *covariate_processing.R* in R.
4. Go to line 516.

5. Simply replace the old file name (KoalaSurveyStrata_v3_ful.shp) inside the quotation marks with the name of the new file you just downloaded. For example, if the new file is called KoalaHabitat_v4.shp, the line should now look like this:

```
reClearHabitat <-  
vect("input/covariates/raw_data/preclean_koala_habitat/KoalaHabitat_v4.shp")  
%>% project(Bounding)
```

Groundwater Dependence

This map shows the ecosystems that are likely to depend on groundwater to survive. Our final analysis used the version published on 28 May 2018 available at

[<https://qldspatial.information.qld.gov.au/catalogue/custom/detail.page?fid={2DF30B15-FA92-47EC-BD2F-5FF1F311DC69}>]. To update it, follow the steps below:

1. Obtain the new version of the layers from its source.
2. The folder for this layer is: input/covariates/raw_data/gde/. Save the new file here.
3. Open the file named *covariate_processing.R*.
4. Go to line 605.
5. You will see a file path inside quotation marks. You need to replace the name of the database file (it ends in .gdb) with the name of the new file you downloaded. For example, if the new file is GDE_2025.gdb, you would update the line to look like this:

```
GDEVect <- st_read("input/covariates/raw_data/gde/GDE_2025.gdb", layer =  
"gw_gde_hgpot") %>% vect() %>% project(Bounding) %>% crop(Bounding)  
%>% aggregate("HGPOT7_DESC")
```

Elevation

This is a "Digital Elevation Model" or DEM. It shows how high the land is above sea level. Our final analysis used the SRTM-derived 1 Second Digital Elevation Models Version 1.0e found at

[<https://ecat.ga.gov.au/geonetwork/srv/eng/catalog.search#/metadata/72759>].

To update it, follow the steps below:

1. Obtain the new version of the layers from its source (probably from [<https://ecat.ga.gov.au>]).
2. The folder for this layer is: input/covariates/raw_data/dem/. Save the new file here.
3. Open the file named *covariate_processing.R*.

4. Go to line 56.
5. You will need to replace the name of the old elevation file (it ends in .tif) with the name of the new one.

For example, if the new file is DEM_1sec_v2.tif, you would update the line to look like this:

```
DEM <- rast(paste(getwd(),
"input/covariates/raw_data/dem/DEM_1sec_v2.tif",sep = "")) %>%
```

Land Use

These maps show how land is being used. Our final analysis used land use maps for 1999 and 2017 (available at:

[\[https://qldspatial.information.qld.gov.au/catalogue/custom/detail.page?fid={273F1E50-DD95-4772-BD6C-5C1963CAA594}}\]](https://qldspatial.information.qld.gov.au/catalogue/custom/detail.page?fid={273F1E50-DD95-4772-BD6C-5C1963CAA594}})).

To add a new version, follow the steps below:

1. Obtain the new version of the layers from its source (probably from [\[https://qldspatial.information.qld.gov.au\]](https://qldspatial.information.qld.gov.au)).
2. The folder for this layer is: input/covariates/raw_data/land_use/. Save the new file here.
3. Open the code file *covariate_processing.R* and follow the next steps carefully.
4. Go to line 239 in *covariate_processing.R*. Click your mouse at the very end of the line and press "Enter" to create a new, empty line below it.
5. Copy the entire code block below and paste it into the empty space you just created.

```
LandUse[YYYY] <-
vect("input/covariates/raw_data/land_use/[NEW_LANDUSE_FILENAME.shp]") %>%
project(Bounding) %>% crop(Bounding) %>% rasterize(rast(Bounding, resolution =
30), field = "Secondary")
LULookup2017 <-
read_csv("input/covariates/raw_data/land_use/land_use_2017_lookup_raw.csv")
LandUse[YYYY] <- classify(LandUse[YYYY], LULookup2017)
writeRaster(LandUse[YYYY], "input/covariates/output/htlus[YYYY][MM].tif", overwrite
= TRUE)
Intensive[YYYY] <- classify(LandUse[YYYY], cbind(c(1, 2, 3, 4, 5, 6, 7), c(0, 0, 0, 0, 1, 0,
0)))
Buff1 <- focalMat(Intensive[YYYY], 1000, "circle", fillNA = TRUE)
Buff2 <- focalMat(Intensive[YYYY], 2000, "circle", fillNA = TRUE)
Buff3 <- focalMat(Intensive[YYYY], 3000, "circle", fillNA = TRUE)
```

```
focal(Intensive[YYYY], Buff1, filename = paste0("input/covariates/output/htilu1km",
"[YYYY][MM]", ".tif"),
      overwrite = TRUE, na.rm = TRUE)
focal(Intensive[YYYY], Buff2, filename = paste0("input/covariates/output/htilu2km",
"[YYYY][MM]", ".tif"),
      overwrite = TRUE, na.rm = TRUE)
focal(Intensive[YYYY], Buff3, filename = paste0("input/covariates/output/htilu3km",
"[YYYY][MM]", ".tif"),
      overwrite = TRUE, na.rm = TRUE)
```

6. Replace every [YYYY] with the year of the new data. For example, if the data is from 2025, change LandUse[YEAR] to LandUse2025. Do this everywhere you see [YEAR].
7. Replace [NEW_LANDUSE_FILENAME.shp] with the actual name of the file you downloaded.
8. Replace every [YYYY][MM] with the year and two-digit month. For example, if the data is for May 2025, you would use 202505.
9. Go to line 244 in the file. Click at the end of the line and press "Enter" to create another new, empty line.
10. Copy the code block below and paste it into this new space.

```
# save rasters
writeRaster(LandUse[YEAR]_mask,
"input/covariates/output/mask/lu[YEAR]_mask.tif")
```

11. Replace [NEW_LANDUSE_FILENAME.shp] with the name of the file you downloaded (the same one as in Part A).
12. Replace both instances of [YEAR] with the year of the new data. For example, this would become LandUse2025_mask and lu2025_mask.tif.

Update survey database and process data for modelling

The R script *data_processing.R* was designed to process all covariate data related to the survey data. It also formats the data appropriately for the Bayesian state-space model used to estimate koala densities in Southeast Queensland.

The basic workflow involves three main steps:

1. Updating the koala survey database

2. Extracting covariate values for the transects in the koala survey database
3. Preparing the data for the modelling phase

In RStudio, start fresh by clicking Session > Restart R. After that, please select all lines by pressing Ctrl + A on a Windows PC or Command + A on a Mac. Then, run these lines by pressing Ctrl + Enter on a Windows PC or Command + Return on a Mac. Alternatively, you can click on Source near the top right corner of the script panel.

When you see "THIS CODE HAS FINISHED" in the Console panel (usually at the bottom left), you're ready to the modelling stage.

NOTE:

4. Disregard any warnings on the taskbar about packages that are not installed
5. This code may take anywhere from a few minutes to days to run, depending on how many files need updating and your computer's specifications.

MODELLING

Fit models

The R script *model_runs.R* was designed to fit the Bayesian state-space model used to estimate koala densities across Southeast Queensland. The model itself is specified in file *nimble_code.R*. You only need to open the *model_runs.R* file in R, select all the lines, and click run. The process will automatically load the parameters previously defined in the *parameter_init.txt* file.

Make predictions

The R script *predictions.R* was developed to estimate koala densities across the whole study area, based on relationships learned from survey data and spatial covariates. Once more, all you need is to open *predictions.R*, select all, lines, and click run.