Migrating distance sampling projects from Distance for Windows to the Distance R package

Eric Rexstad, David L. Miller, Laura Marshall and Len Thomas

Centre for Research into Ecological and Environmental Modelling University of St Andrews

ISEC International Statistical Ecology Conference

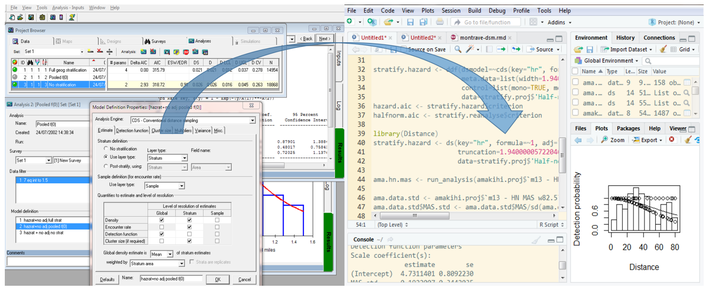
CREEM Centre for Research into Ecological and Environmental Modelling

University of St Andrews FOUNDED 1413

distancesampling.org

Introduction

The Distance software (Thomas et al., 2010) has been downloaded >40,000 times in its 20-year history. It consists of a Windows-based Graphical User Interface (GUI) that gives users ready access to survey design tools (written in Visual Basic) and analysis engines (written in FORTRAN and, more recently, R). For some users, there may be benefits to performing their analyses directly with the underlying R code, rather than working with the GUI.

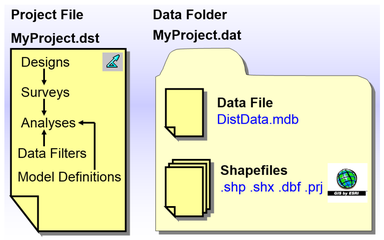


Challenges hindering the transition between analysis with the GUI and analyses in R are two-fold:

- Legacy data reside in Distance (GUI) projects, unavailable for importing into R, and
- Analyses that are easily described using the GUI may be difficult to specify, particularly if analyst is not proficient in R.

This poster describes an R package, `readdst`, intended to alleviate these challenges.

Getting data out of Distance



Distance sampling analysis are defined in Distance project files (an Access database).

- Data
- Model definitions
- Analysis results

`readdst` uses `RODBC` (Windows, the 32-bit version of R must be used for RODBC to function properly) or `mdb-tools` (Mac) to mine the information from the database. With the database components available, a sequence of translation steps are undertaken.

- Translate model definitions into R code
- Save data into an R `environment`
- Extract results from the database for completed analyses

Converting Distance projects to R

Workhorse function in `readdst` package is `convert_projects()`. Its single argument is the absolute path to an existing Distance GUI project (created by Distance 6 or 7). `convert_projects()` interrogates the Access database and translates the contents of database tables into an R object of class `converted_distance_analyses`, a list of lists of class `converted_distance_analysis`.

Object produced by `convert_project()`

The list returned by `convert_project()`, for a given entry (analysis), contains all the salient information of a Distance analysis extracted from the Access database. We point out two elements of the list critical to subsequent processing of the now-extracted data.

call R code call to the function `ddf()` in package `mrds` to duplicate the analyses done in the Distance GUI.

env An *environment* consisting of a set of data frames. These data frames include the original data extracted from the Distance project, along with the observation, sample and region tables describing the hierarchial nature of the data base.

The `converted_distance_analysis` list can be passed as an argument to `run_analysis()`. Given `object` has been created by a call to `convert_project`, `run_analysis()` will perform an `mrds` analysis by processing data in `object$env` using `ddf()` syntax in `object$call` to conduct a distance sampling analysis.

Learning distance sampling in R

Distance GUI users can use `converted_distance_analysis` objects to learn how to perform corresponding analyses using the `mrds` R package:

```
> library(readdst)
> stratify.proj <- convert_project("C:/Distance Projects/myproject")
> stratify.proj
  ID                                                Name  Status
1 13 Half-normal cosine no stratification exact           Ran OK
2 16           Half-normal cosine pooled detfn            Ran OK
3 15    Half-normal cosine strat-specific detfn Ran with warnings
> stratify.proj[[1]]
Model name  : Half-normal cosine no stratification exact
ID          : 13
Data filter :
mrds call   : mrds::ddf(dsmodel=~cds(key="hn", formula=~1,
                adj.series="cos", adj.order=NULL),
                meta.data=list(width=NA, left=0),
                control=list(mono=TRUE, mono.strict=TRUE),
                method="ds", data=data)
```

Note too that components from objects of type `converted_distance_analysis` can be analysed using the `ds()` function from the `Distance` package.

Comparative analysis of data sets

The `test_stats()` function in `readdst` performs the analysis residing in the `call` element of the `converted_distance_analysis`, producing estimates of parameters of distance sampling analyses (*e.g.* AIC scores, $\hat{P}_a$, $\hat{D}$, $\hat{D}$ and associated measures of precision). These results are contrasted with estimates of the same parameters stored in the Access data base, as computed by the Distance GUI.

The resulting comparison is produced in tabular form with the difference between estimates presented as proportions. Those comparative estimates within a specified tolerance are designated with **X**.

Statistic	Distance_value	mrds_value	Difference	Pass
n	90	90	0	X
parameters	1	1	0	X
AIC	63.880100250244	63.879819152981	0.000004396471	X
Chi^2 p	0.1707298	0.9906514	4.802452	
P_a	0.451959013939	0.451956754149	0.000004969147	X
CV(P_a)	0.07679188997	0.07679084338	0.00001362931	X
log-likelihood	−30.94005	−30.93991	0	X
C-vM p	1	0.9	0.1	
density	0.05059615	0.02768243	0.4528748	
CV(density)	0.2693232	0.3070729	0.1401652	
individuals	34658	18962.4623604	0.4528691	
CV(individuals)	0.2693232	0.3070729	0.1401652	

This testing capability assists our strategy to migrate software development from the existing Distance GUI to R packages. We can check consistency of results between the Distance GUI and developing R code.

Caveats

`readdst` is not able to translate all GUI analyses into R code. Current limitations are inability to translate

- analyses using the `dsm`, `mads` and `Dssim` engines,
- analyses using post-stratification and
- bootstrapping for variance estimation.

Additional information

References

Miller, D. L., E. Rexstad, L. Thomas, L. Marshall, and J. Laake. 2016. Distance Sampling in R. *bioRxiv*.

Thomas, L., S. T. Buckland, E. A. Rexstad, J. L. Laake, S. Strindberg, S. L. Hedley, J. R. Bishop, T. A. Marques, and K. P. Burnham. 2010. Distance software: design and analysis of distance sampling surveys for estimating population size. *Journal of Applied Ecology*, 47(1):5–14.

QR code readdst Github repository

QR code Miller et al. (bioRxiv)

QR code Thomas et al. (2010)