foieGras an R package for animal movement data: rapid quality control, behavioural estimation and simulation

# Abstract

### Keywords:

# 1 | Introduction

The collection of animal-borne sensor data has become essential for understanding the movements, behaviour, social interactions, foraging ecology, physiology, habitat use and population dynamics of mobile and/or cryptic species . In addition, the sophistication and precision of current sensor technology now enables the use of tagged animals to collect in situ environmental information that compliments data collected from traditional Earth observing platforms . However, in each of these applications, common issues with tracking data such as irregularly timed measurements, sensor biases and location measurement errors must be overcome. Underlying animal movement processes must also be adequately characterised before robust inferences can be made about where animals are and what they are doing when they record or transmit information.

State-space models (SSMs) are powerful tools for conducting quality control of and making behavioural inference from animal tracking data . Within this framework, an animal’s true location and/or behaviour in the wild are unobserved states, while measurements recorded by animal-borne sensors provide noisy observations. The SSM comprises both a movement process model (e.g., a correlated random walk) and a measurement model that are fit to animal tracks to estimate true location and behaviour states, thereby separating ecologically-meaningful signals from the noise of sensor measurements.

Here we introduce foieGras (pronounced “*fwah grah*”), a package for fitting SSMs in R . This package has two aims: (1) to be a simple and fast implementation of SSMs to quality control error-prone animal location data ; and (2) for inference of changes in behaviour along animal tracks . Although several implementations of SSMs for animal movement data exist, e.g. bsam , crawl , and ctmm , their broad utility can be hampered by their technical complexity. foieGras has a more user-friendly implementation suitable for novices, yet users can exert control over many aspects of the package functions via optional arguments, accommodating the needs of experienced users.

We describe the main features of foieGras and illustrate its use through a set of applications drawing on Argos and GPS tracking data. Full R code and data for each of the applications is provided in the Supporting Information. Additional details on package functions and use can be found in the help files and package vignettes.

# 2 | foieGras overview

The workflow for foieGras is deliberately simple, with many of the usual track data processing checks handled automatically. Here we outline the main aspects of the foieGras package. The package’s main functions with brief code examples are listed in Table , and a generalized workflow with example code is depicted in Figure .

## 2.1 | Data preparation

Animal tracking data, consisting of a time-series of location coordinates, can be read into R as a data frame using standard functions such as read.csv. Possible input data formats are shown in Figure . More details on input data formats are provided in the Overview vignette (<https://ianjonsen.github.io/foieGras/articles/Overview.html>).

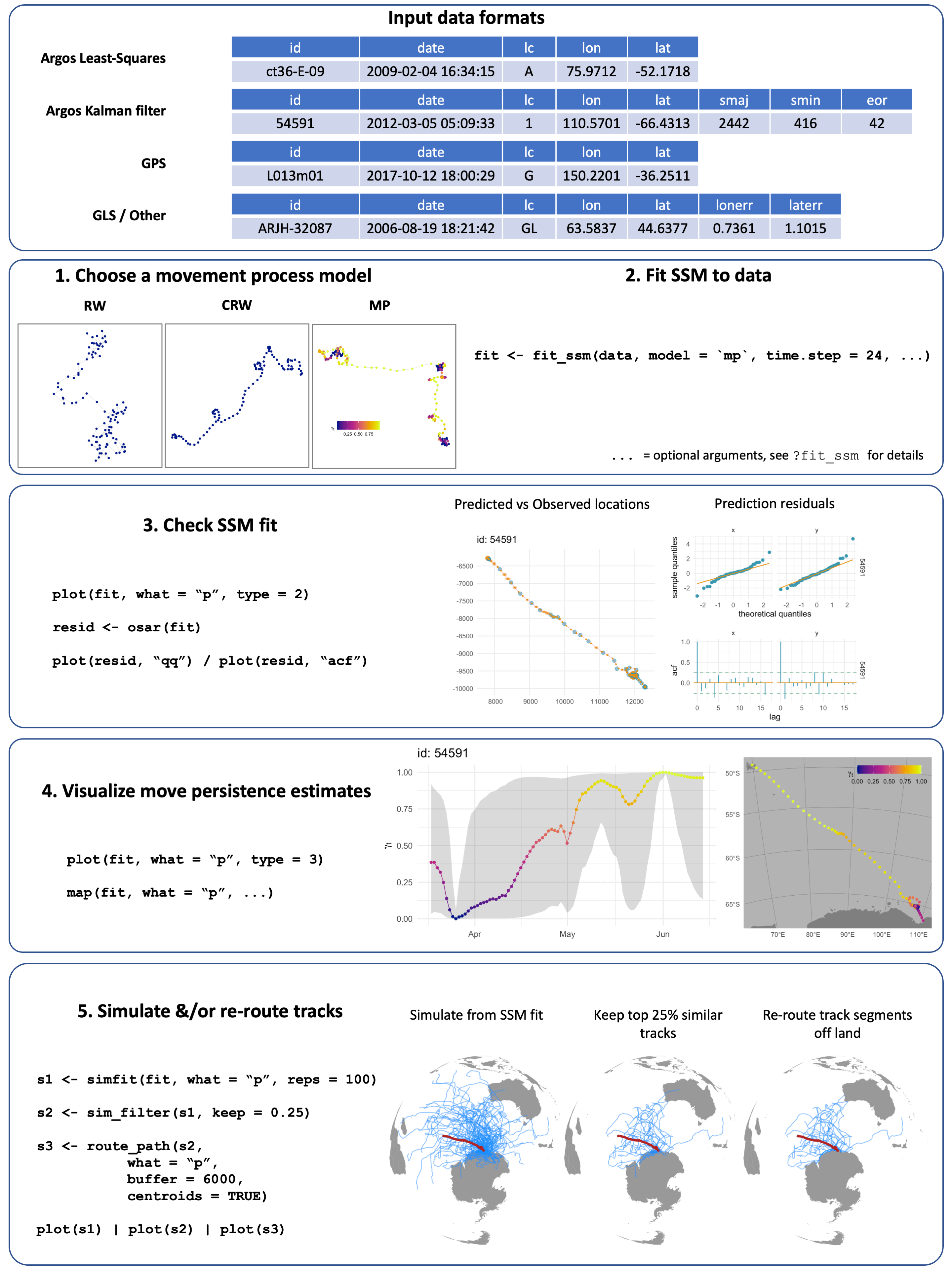
## 2.2 | State-space model fitting - fit\_ssm

State-space models are fit using the function fit\_ssm (Fig. ). The type of location data is automatically detected from the location quality classes that are typical of Argos data and that can be added by the researcher for other data types. Based on the location quality classes and optional information on measurement errors contained in the data, an appropriate measurement error model is selected for each observation . Fits to single versus multiple individuals are handled automatically, with sequential SSM fits occurring in the latter case. No hierarchical or pooled estimation among individuals is currently available.

There are a large number of optional arguments that can be specified in fit\_ssm, and these are explained in the package documentation. We focus only the essential arguments here:

* data the input data structured as illustrated in Fig.
* model the process model to be used (one of rw, crw, or mp)
* time.step the prediction time interval (expressed in hours)

The function first invokes an automated data processing stage where the data type and measurement error model(s) are determined, observation times are sorted and checked for duplicates, and a speed filter identifies potential extreme locations to be ignored by the SSM. The SSM is then fitted to the processed data, with the user-specified process model and automatically selected measurement error model(s). The joint likelihood is optimized numerically using either of the standard R optimizers, optim or nlminb. The R package TMB, Template Model Builder , is used to compute the gradient function in C++. A fit object is returned as a nested data frame, listing the individual animal id(s), basic convergence information and a list of model output including estimated parameters and states, processed data, and diagnostic information. A simple data frame of SSM fitted (location estimates corresponding to the observation times) or predicted values (locations predicted at regular time.step intervals) can be extracted using the grab function. Parameter estimates, AIC and other model fit information can be viewed using the summary function.



A generalized workflow showing required input data formats, model fitting, model checking/validation, visualisaiton of movement behaviour estimates, track simulation and re-routing around land. Example code is displayed for each stage.

## 2.3 | Behavioural estimation - fit\_ssm, fit\_mpm

Move persistence, an index of movement behaviour, can be estimated as a continuous-valued (0 - 1), time-varying latent variable that represents changes in movement pattern based on autocorrelation in speed and direction . There are two approaches in foieGras for estimating move persistence. The first is to use fit\_ssm with model = 'mp', which fits a continuous-time move persistence model in state-space form and thereby simultaneously estimates true locations and move persistence from the error-prone telemetry data (Fig. ). This approach is most appropriate for fitting to irregularly-timed and error-prone Argos data as both aspects are taken into account explicitly. The second is to use fit\_mpm, which can take as input either location data or SSM-estimated locations from an fit\_ssm model fit object. This approach is generally more appropriate when the data have minimal measurement error (e.g., GPS locations; see Application 3.2).

## 2.4 | Model checking and visualization - osar, plot, map

Before using fitted or predicted locations, a fit\_ssm model fit should be checked and visualized to confirm that the model adequately describes the data. There is no simple way to calculate residuals for latent variable models that have non-finite state-spaces and that may be nonlinear, but they can be computed based on iterative forecasts of the model . The osar function computes one-step-ahead (prediction) residuals via the oneStepPredict function from the TMB R package . A set of residuals are calculated for the x and y values corresponding to the fitted values from the SSM. A generic plot (plot.osar) method provides an easy way to visualize the prediction residuals as time-series plots, quantile-quantile plots, or autocorrelation functions (Fig. ).

State-space model fits to data can also be visualised by using the generic plot (plot.ssm\_df) function on a model fit object. Options exist to plot fitted or predicted values along with observations as either paired, 1-D time-series (using the type = 1 argument), or as 2-D tracks (type = 2) with 95% confidence intervals or ellipses. These plots provide a rapid check on SSM fits to data. Additionally, when the fitted SSM is the move persistence model (i.e., model = 'mp'), 1-D time-series (type = 3) or 2-D track plots (type = 4) of move persistence estimates can be displayed (Fig. ).

fit\_ssm model fits can be mapped using the map function for single or multiple individuals. By default, map uses the coastline data from the rnaturalearth R package at medium or high resolution (Fig. ), but can also use tiled maps for finer-scale detail, via the rosm and ggspatial R packages. Mapping aesthetics (e.g., plot symbols, sizes, colours, fills) can be customized via the aes argument and use of the aes\_lst function. See code in SI for examples.

All foieGras visualizations draw on the ggplot2 R package , with multi-panel plots also using the patchwork R package , and generally can be modified through additive calls in the usual ggplot2 manner. See code in SI for examples.

## 2.5 | Simulation - sim, simfit, sim\_filter

Track simulation can be a helpful, yet informal, way of evaluating the degree to which statistical movement models capture essential features of animal movement data . The sim function can simulate a variety of movement process, including the rw, crw, and mp process models, as well as simple multiple movement state switching processes.

Simulation is also used frequently in habitat usage modelling to provide a measure of habitat availability by generating a source of ‘background’ points representing a null model of the distribution of foraging animals in the absence of external drivers . The simfit function extracts movement parameters from an SSM fit object and simulates random tracks of the same duration from these parameters. The argument cpf = TRUE allows simulation of central place foragers by ensuring that tracks start and end at approximately the same location. Movements can also be constrained to remain mostly in water via a potential function , using included gradient rasters and the grad and beta arguments. These are illustrated in the code for Application 3.3.

The choice of null points can impact the performance of habitat suitability models , and so the sim\_filter function provides a tool to filter the simulated tracks based on their similarity to the original path. Filtering uses one of two metrics that capture the difference in the net displacement and bearing between the two paths (see ?similarity\_flag for more detail). These metrics are motivated by the ‘flag value’ described in . The the quantile of flag values to be retain is specified via the keep argument; i.e. keep = 0.25 (the default) will return a simfit object containing those simulated tracks with flag values in the top 25% of values calculated for the input simfit object (Fig. ).

## 2.6 | Path rerouting - route\_path

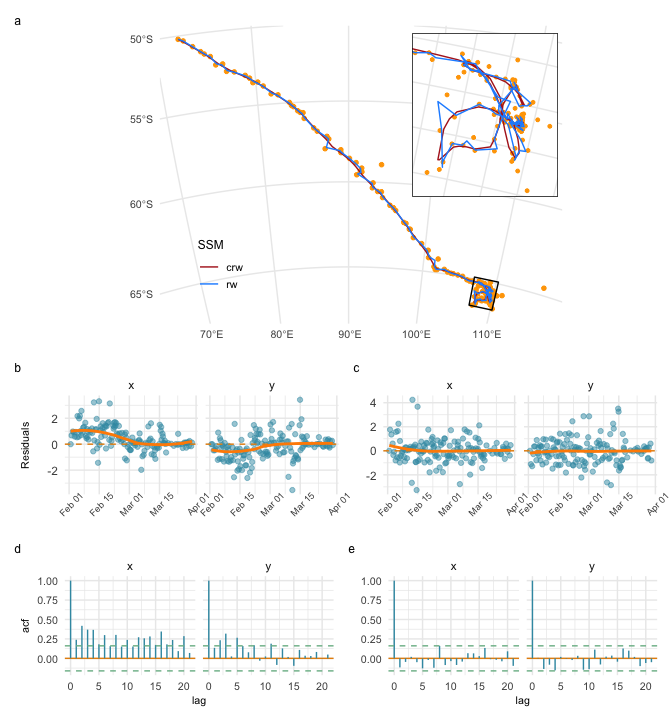
As the SSMs implemented in foieGras have no information about potential barriers to animal movement it is possible for locations to be estimated in implausible locations, such as on land for marine species. To overcome this, foieGras makes use of the pathroutr R package to efficiently re-route locations from land back to water by using visibility graphs . The route\_path function can be applied to either an SSM fit object or the simulations generated by simfit. When the input is an SSM fit object the re-routed path can be appended to the object for visualization and use in subsequent analyses. When the input is a simfit object the locations within the simulation are replaced with the re-routed paths (Fig. ). We illustrate the latter in Application 3.3.

# 3 | Applications

We illustrate the main capabilities of foieGras through a set of applications that are for demonstration purposes. Complete code and data for reproducing the applications and for gaining a deeper understanding of foieGras functions are provided in the Supplementary Information.

## 3.1 | SSM validation with prediction residuals

We used a sub-adult male southern elephant seal (*Mirounga leonina*) track included as example data in foieGras (sese2, id: ct36-E-09), sourced from from the Australian Integrated Marine Observing System (IMOS; data publicly available via [imos.aodn.org.au](http://imos.aodn.org.au)) deployments at Iles Kerguelen in collaboration with the French IPEV and SNO-MEMO programmes. The data are temporally irregular Argos Least-Squares locations. We fitted both the rw and crw models using fit\_ssm() with a speed filter threshold (vmax) of 4 ms-1 and a 12-h time step. We calculated prediction residuals using osar(), and then used the generic plot method for osar residuals to assess and compare the model fits (Fig. ).



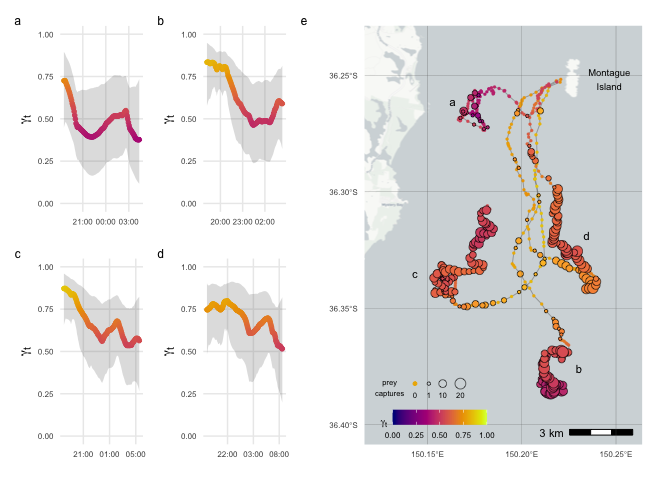
State-space model fits to a southern elephant seal track with finer scale detail inset (a), and diagnostic plots for assessing goodness-of-fit of the rw (b - prediction residual time-series; d - prediction residual autocorrelation) and crw (c,e) state-space models. All residual plots generated using the plot.osar function.

Plots of predicted states on top of the observations suggest both models yield similar global fits (Fig. a; blue versus red lines), but discrepancies are visible at a local scale (Fig. a, inset). Predicted locations from the two models differ by a median 6.62 km (range: 0.02, 53.02 km), and there are marked trends in the residuals for the rw model fit (Fig. b) with significantly positive autocorrelation in both the x and y directions (Fig. d). The crw prediction residuals show no trend through time and have little autocorrelation (Fig. c,e), implying the crw process model is a better fit to the data.

## 3.2 | Inferring move persistence

To illustrate how move persistence () can be estimated from GPS data, we use four little penguin (*Eudyptula minor*) tracks of daily foraging trips during the chick-rearing period from Montague Island, NSW, Australia . The data are temporally irregular GPS locations, with high frequency sampling (mean 15 s) intermittently disrupted by the birds’ diving behaviour, and are assumed to have minimal measurement error. We fitted the crw SSM to the GPS data to predict temporally regular locations at 5-min intervals, and assumed consistently small bivariate normal location measurement errors (ie. 30 m sd). We then used fit\_mpm() to estimate from these regularized locations. Finally, we visually compared estimates with prey capture events inferred from accelerometry data to ascertain whether reduced is indicative of active foraging by these penguins.

We provide an extra application to illustrate how move persistence can be inferred from Argos data using fit\_ssm() (Appendix S4).



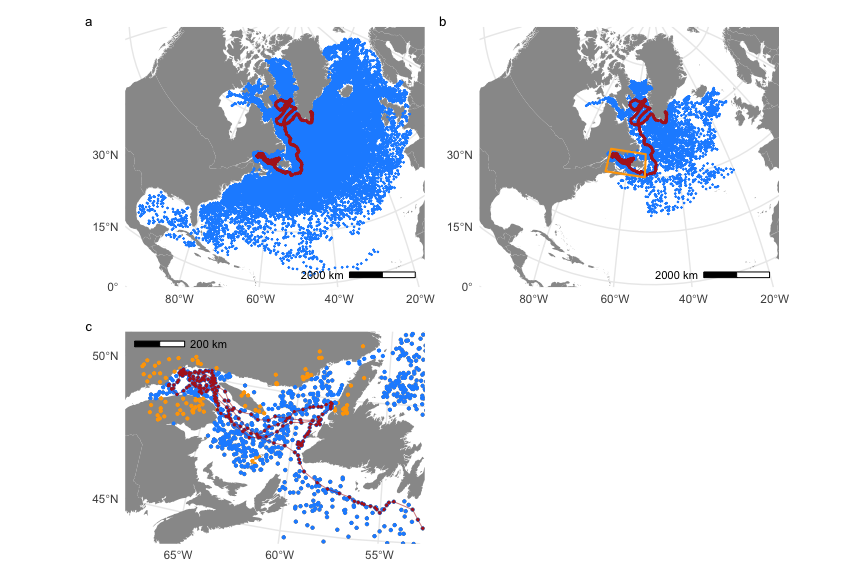
Inferred move persistence, , 1-D time-series (a-d; grey envelopes are 95 % CI’s) and along SSM-predicted little penguin tracks (e). Location symbol sizes are proportional to the number of prey capture events, inferred from accelerometry data, at 5-min intervals corresponding to the SSM predicted location times.

The little penguin GPS tracks did not exhibit strong contrast in move persistence, with declining below 0.5 for only two of the birds (Fig. a-d). Nonetheless, the move persistence estimates highlight change in movement pattern over the course of the penguins’ daily foraging trips (Fig. e). The penguins departed Montague Is. with relatively fast movements, three directed southward (Fig. tracks b-d in e) and one less directed and remaining close to the island (track a), before slowing down and engaging in meandering movements (orange - red in Fig. e). Lower was associated with consistently higher and more frequent prey captures, implying that low move persistence can be indicative of foraging activity by these penguins. The spatially diffuse bouts of lower move persistence may reflect the fine-scale patchiness of the penguins’ forage-fish prey with search and prey-capture occurring both within and among discrete neighbouring prey aggregations .

## 3.3 | Simulating tracks from foieGras model fits

To illustrate how to simulate tracks from foieGras model fits we use a juvenile harp seal (*Pagophilus groenlandicus*) tracked from the Gulf of St Lawrence, Canada, and . The data are temporally irregular Argos locations including error ellipse information. We fit the crw model using fit\_ssm with a 4 ms-1 speed filter threshold (vmax) and a 12-h prediction interval (time.step).

We simulate 50 animal movement paths from the crw process model using simfit, and apply a potential function using the grad and beta arguments to constrain the simulated paths to largely remain in water. These tracks are then filtered based on their similarity to the original path using sim\_filter and the top 10% retained (keep = 0.1)(Fig. a,b). As the potential function does not guarantee all locations remain off land, we re-route any remaining simulated locations from land back to water using route\_path (Fig. c). In combination, these functions provide a pragmatic, non-statistical method to generate and objectively filter pseudo-tracks for use in movement or habitat modelling applications.



Simulating (a) 100 movement paths from a correlated random walk process model; (b) filtering those tracks to select the top 10% based on their similarity to the original SSM-predicted track (red); and (c) re-routing simulated locations on land (orange) back to ocean (blue). The orange box in (b) indicates region magnified in (c). SSM-predicted track (red) overlaid in all panels for context.

# 4 | Conclusions

The foieGras package was developed to ease fitting state-space models for quality control of animal location data and for inference of behavioural change along animal tracks. We achieve these primarily through a simple yet extensible workflow, model parsimony, and computational speed. Combined, these traits accommodate both novice or occasional and advanced users, and facilitate use in automated, operational quality-assurance/quality-control processes for animal-borne ocean observations .

The foieGras package is an intermediate analysis toolbox where location quality control typically occurs after some initial data processing but prior to any comprehensive, final analysis. In this vein, the move persistence model tools provide a rapid approach for objectively identifying changes in movement behaviour along animal tracks without any required *a priori* knowledge or decisions about the kind or number of behavioural states potentially hidden within the data. Subsequent analysis could entail use of a hidden Markov model to infer potential drivers of observed movements . Alternatively, move persistence - covariate relationships among multiple individuals can be inferred rapidly in a mixed-effects model framework using the mpmm package .

The foieGras package will undergo further development, expanding the range of available SSM’s via new movement process models and/or enhanced measurement models for other electronic tracking data types. Additionally, we will seek to enhance integration with other R packages for processing and analysis of animal movement data, where feasible. Feedback from users is invaluable and encouraged. Users may submit bug reports and enhancement suggestions via the foieGras GitHub issues page (<https://github.com/ianjonsen/foieGras/issues/new/choose>). More general feedback is always welcome by contacting the lead author directly.

# Acknowledgements

We thank Marie Auger-Méthé for contributing original code to the movement persistence models. IDJ acknowledges support from a Macquarie University co-Funded Fellowship and from partners: the US Office of Naval Research, Marine Mammal Program (grant N00014-18-1-2405); the Integrated Marine Observing System (IMOS); Taronga Conservation Society; the Ocean Tracking Network; Birds Canada; and Innovasea/VEMCO. TAP was supported by CSIRO Oceans & Atmosphere internal research funding scheme. The Integrated Marine Observing System (IMOS) supported seal fieldwork. IMOS is a national collaborative research infrastructure, supported by the Australian Government and operated by a consortium of institutions as an unincorporated joint venture, with the University of Tasmania as Lead Agent. Field work at Illes Kerguelen was conducted as part of the IPEV programme No 109 (PI H. WEIMERSKIRCH) and of the SNO-MEMO programme (PI C. GUINET) in collaboration with IMOS. CTD tags were partly funded by CNES-TOSCA and IMOS. Little penguin fieldwork was supported by an Australian Research Council Linkage grant to IDJ, GC and RGH (LP160100162). Harp seal fieldwork was funded by Fisheries and Oceans Canada, a bursary from Department for Business, Energy and Industrial Strategy (BEIS) administered by the NERC Arctic Office, and the ARISE project (NE/P006035/1 and NE/P00623X/1), part of the Changing Arctic Ocean programme jointly funded by the UKRI Natural Environment Research Council (NERC) and the German Federal Ministry of Education and Research (BMBF). All animal tagging procedures approved and executed under the Animal Ethics Committee guidelines of the University of Tasmania (elephant seals), Macquarie University (little penguins), and University of St Andrews (harp seals).

# Author’s Contributions

IDJ developed the R package; WJG contributed harp seal data and to the R package; LP, GC, and RGH contributed little penguin data; CRM, RGH and MAH contributed Southern elephant seal data; IDJ and TAP developed the state-space models; IDJ wrote an initial draft of the manuscript with contributions from WJG; all authors contributed to the final manuscript.

# Data Accessibility

All code and data used here are provided in the foieGras package for R or in the Supplementary Information. The latest stable and cross-platform tested version of the package (currently, 1.0-7) is available via ROpenSci’s R-universe, at <https://ianjonsen.r-universe.dev/ui#package:foieGras>. The latest partially tested stable and development versions are available on the GitHub repository: <https://github.com/ianjonsen/foieGras>. An older version of foieGras (0.7-6) remains on CRAN at , however, we recommend users upgrade to the latest R-universe version for full access to the functionality presented here.

# ORCID

*Ian D Jonsen*   
*W James Grecian*   
*Lachlan Phillips*   
*Gemma Carroll*   
*Clive R McMahon*   
*Robert G Harcourt*   
*Mark A Hindell*   
*Toby A Patterson*

# References