Vignette

Ophélie Couriot

09/22/2022

Contents

Estimate parturition in caribou using the TuktuTools Package

The TuktuTools Package has been developed to analyse caribou movement data. This package contains several functions allowing to: - Visualize movement data - Clean and process movement data - Get the movement rate between successive locations for each individuals - Get the euclidean distance between each pairs of individuals from a dataset - Get the utilisation distribution of an individual or all individuals during a given period - Discriminate between parturient and non parturient individuals and get the parturition timing and location - …

The *`parturition.model`* is a function allowing to estimate calving (i.e. parturition) status and calving dates in caribou (*Rangifer tarandus*), using an individual based method developped by DeMars et al. (2013). This method infers calving status and neonate survival from female caribou movement patterns. In particular, DeMars et al. predicted that calving events could be identified by a sudden and marked change – or break point – in female mean step length (i.e. the distance between two successive relocations). The method assumes that, for calving, a female would depresse her movements and that movement rates would remain low as long as the calf was alive, since neonates are not able to move as adults, thus acting as a spatial "anchor". Conversely, if the calf was lost during the neonatal period, a second break point would occur in the female movement pattern, since the female would recover her pre-calving movements abruptly, due to calf loss. Non calving female would have a constant movement rate throughout the entire calving period.

While this method has proven good reliability, it requires an homogeneous time series (i.e. homogeneous time interval between successive relocations). To dispense with data homogenization, we adapted this method by performing analyses on speed (i.e. the distance travelled between two successive relocations on the time between these two relocations) instead of step length.

This document describes the data and the steps to clean data, estimate calving date and location.

Clean data

It appears that sometimes, the GPS device relocate at 1 or 2 minute interval. However, the GPS devices are configured to relocate every day, 8-hours or every hour for some individuals, at fix hours (e.g. 0:00 am, 8:00

am or 4:00 pm). In addition, it is biologically impossible that the animal moves several kilometers in one minute. Thus, the relocations having a time interval in the order of minutes are more prone to be 'outliers'.

We cleaned all the data sets by using the `removeOutliers` function in this package.

The function considers as outliers, fixes that are not biologically probable, using the following rules:

- if speed between the previous fix and this one is higher than 15 km per hour
- if time interval between the previous fix and this one is lower than 2 minutes
- if speed is higher than 10 km per hour and time interval is less than 10 minutes

Usage

```
removeOutliers(df, steps = 10,
               CRS = "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0
               +ellps=GRS80 +datum=NAD83 +units=m +no_defs")
```

Arguments

`df` a data frame containing columns: ID as individual identifiant, x and y: relocations of individuals (metric system specified in CRS) DateTime: vector (of class POSIXct)

`steps` if specified, the number of cleaning steps to be performed (default is 10)

`CRS` the coordinates projection (default is Canada Lambert Conformal Conic: "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0 +ellps=GRS80 +datum=NAD83 +units=m +no_defs")

Value

a list with a dataframe without potential 'outliers' and a dataframe of outliers

Focus on the calving period

After having removed potential outliers, we focused on the calving period, to avoid the individual based method to detect other behaviour than calving. The calving period has been described to be between May 19 and July 07 for barren-ground caribou (Cameron et al. 2018). The function `cutperiod` allows to cut the timeseries for a given start and end date, but also to exclude individuals that do not have monitoring covering this entire period of time. We can also decide the maximum time interval during which there are no relocations (i.e. missing values) and the minimum average time interval needed.

Usage

```
cutperiod(df, start,  end, nfixes, dayloss,
          CRS = "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0   +ellps=GRS80 +datu
```

Arguments

`df` a data frame containing columns: ID_Year as individual identifiant per year, x and y: relocations of individuals (in NAD83 utm zone 19N) DateTime: date and time vector (of class POSIXct)

`start` starting month and day of the period of interest as a vector c(mm,dd) (example for the 19th of May: c(19,05))

`end` end month and day of the period of interest as a vector c(mm,dd) (example for the 15th of July: c(07,07))

`nfixes` minimun number of fixes per day (taking the average number of fixes per day for each individual across the period of interest)

`dayloss` maximum number of days with missing locations (for example, if an individual has loss signal for more than 3 consecutive days, it will be excluded from the dataset)

`CRS` the coordinates projection (default is Canada Lambert Conformal Conic: "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0 +ellps=GRS80 +datum=NAD83 +units=m +no_defs")

Value

The function returns a dataframe containing only concerned period and individuals that match the defined rules.

Get movement rate

After having processed, cleaned and cut timeseries to the period of interest and reassigned herds, we can process to the analysis of movement pattern and estimate calving (i.e. parturition) status, calving date and calf death date, if any.

As we adapted the method developped by DeMars et al. (2013), by analysing movement rate instead of step length, we first have to get the movement rate between subsequent relocations for each individual and each year. To do so, we use the `get.speed` function in this package.

Usage

```
get.speed(df, CRS = "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0 +ellps=GRS80 +da
```

Arguments

`df` a data frame containing columns: ID_Year as individual identifiant per year, x and y: relocations of individuals (in N Canada Lambert Conformal Conic), DateTime: date and time vector (of class POSIXct)

`CRS` the coordinates projection (default is Canada Lambert Conformal Conic: "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0 +ellps=GRS80 +datum=NAD83 +units=m +no_defs")

	ID_Year	ID	nickname	DateTime	Lon	Lat	sex	Year	yday
147702	BGCA17131_2017	BGCA17131	BA17131	2017-06-01 08:00:00	-109.1744	66.74685	f	2017	152
147703	BGCA17131_2017	BGCA17131	BA17131	2017-05-19 16:00:00	-110.5404	66.03517	f	2017	139
147704	BGCA17131_2017	BGCA17131	BA17131	2017-06-01 16:00:00	-109.1304	66.75325	f	2017	152
147705	BGCA17131_2017	BGCA17131	BA17131	2017-05-31 00:00:00	-109.1388	66.73405	f	2017	151
147706	BGCA17131_2017	BGCA17131	BA17131	2017-05-31 08:00:00	-109.1102	66.72714	f	2017	151
147707	BGCA17131_2017	BGCA17131	BA17131	2017-05-20 00:00:00	-110.4893	66.12021	f	2017	140

	ID_Year	ID	nickname	DateTime	Lon	Lat	sex	Year	yday
147711	BGCA17131_2017	BGCA17131	BA17131	2017-05-19 00:00:00	-110.6295	65.86413	f	2017	139
147708	BGCA17131_2017	BGCA17131	BA17131	2017-05-19 08:00:00	-110.6010	65.91193	f	2017	139
147703	BGCA17131_2017	BGCA17131	BA17131	2017-05-19 16:00:00	-110.5404	66.03517	f	2017	139
147707	BGCA17131_2017	BGCA17131	BA17131	2017-05-20 00:00:00	-110.4893	66.12021	f	2017	140
147715	BGCA17131_2017	BGCA17131	BA17131	2017-05-20 08:00:00	-110.4349	66.17452	f	2017	140
147712	BGCA17131_2017	BGCA17131	BA17131	2017-05-20 16:00:00	-110.3993	66.22834	f	2017	140

Value

The function returns a data frame with columns speed, which is the movement rate between subsequent relocations (in m.h-1), time.mid corresponding to the middle time between the associated relocations and the next one, dhours which is an index of the cumulative hours of each speed values from the first speed value, dt the time lags between the relocations (in hour), ID as the identifiants of the individuals-year and x,y relocations.

Get movement rate for all herds during the calving period

The data *barrenground* conmtains movement data during the calving period for 5 barren-ground caribou females from the Bathurst herd.

```
data(barrenground)
kable_styling(kable(head(barrenground)))
```

```
c1 <- get.speed(barrenground)
kable_styling(kable(head(c1)))
```

Estimate calving status and timing

We can now estimate calving status, calving date and calf death date, if any. To do so, we use the `parturition.model` function in this package. This function determines calving status of a female, among no calf, with a calf or lorst her calf, as well as the calving date and the calf death date.

As stated previously, we adapted the individual based method developped by Demars et al. (2013), which has proven good reliability to estimate calving status and calving date for females from the Western Arctic Herd in a previous study (Cameron et al. 2018). However, we adapted this method to be able to infer calving based on the female speed through time. Thus, all models assume speed follows a Gamma distribution and differ in two parameters: shape and scale, which correspond to $\frac{\overline{speed}^2}{var(speed)}$ and $\frac{var(speed)}{\overline{speed}}$, respectively.

The mean speed is thus equal to $shape * scale$.

- For the model representing females that do not calve: the mean speed remains constant through the entire calving period.

- For the model representing females that had a calf who survived 4 weeks after birth: the mean speed is constant before calving, then abruptly drops for calving, creating a break point. After calving, the mean speed increases progressively following: $(shape_{calving} * (\frac{\overline{shape} - shape_{calving}}{k}) * time) * (scale_{calving} * (\frac{\overline{scale} - scale_{calving}}{k}) * time)$ where k is the time, defined in days, required for the calf to achieve adult movement rates.
- For the model representing females losing calves: there is an abrupt change in the slope of the post-calving increase, creating a second break point after which the mean speed immediately recovers its pre-calving value.

The models therefore differ in their number of parameters to estimate: the no calf model has two – shape and scale; the calf model has five – shape and scale before calving, scale at calving, k and calving date; and the calf death model has six – shape, scale before calving, scale at calving, k, calving and calf death dates. We discriminated among models using Akaike's Information Criterion (AIC) with the best model being the one with the lowest AIC value.

Usage

```
parturition.model(df, int, kcons, PlotIt = FALSE,  saveplot = FALSE,
    CRS = "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0 +ellps=GRS80
    +datum=NAD83 +units=m +no_defs")
```

As we could not check for medel accuracy concerning the calf loss model, we decided to only compare the no calf and the calf model and then used the 'parturition.model2' function.

Usage

```
parturition.model2(df, int, kcons, PlotIt = FALSE,  saveplot = FALSE,
    CRS = "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0 +ellps=GRS80
    +datum=NAD83 +units=m +no_defs")
```

Arguments

`df` a dataframe containing the speed between subsequent relocations of an individual, the coordinates of the relocations (in metric system) the date and time oh each speed value and t he difference in hours between each speed value and the first one and a vector of the animal-year identifiant. See ?get.speed for more information on the Data requirements

`int` integer value indicating the minimum number of days between the beginning of the time series and the first BP (calf birth), the last BP (calf death) and the end of the time series, and between the two BPs. The main reason for this constraint is that a minimum number of data points are required in each section of the time-series to be able to estimate model parameters. We recomanded 9 relocations, thereby 3 days.

`kcons` vector of the minimum and maximum time it takes the female to recover normal speed (in days)

`PlotIt` if PlotIt = if PlotIt = TRUE the function will draw a plot of the speed in function to the date with the prediction line ot the best model selected (by 'AIC'), among no-calf, calf, calf death based on the actual speed of the female in function of the date

`saveplot` if saveplot = TRUE, the plot of the best model (either based on AIC or CC depending on either PlotAIC or PLOTCC = TRUE) will be saved

`CRS` the coordinates projection (default is Canada Lambert Conformal Conic: "+proj=lcc +lat_1=50 +lat_2=70 +lat_0=65 +lon_0=-120 +x_0=0 +y_0=0 +ellps=GRS80 +datum=NAD83 +units=m +no_defs")

Value

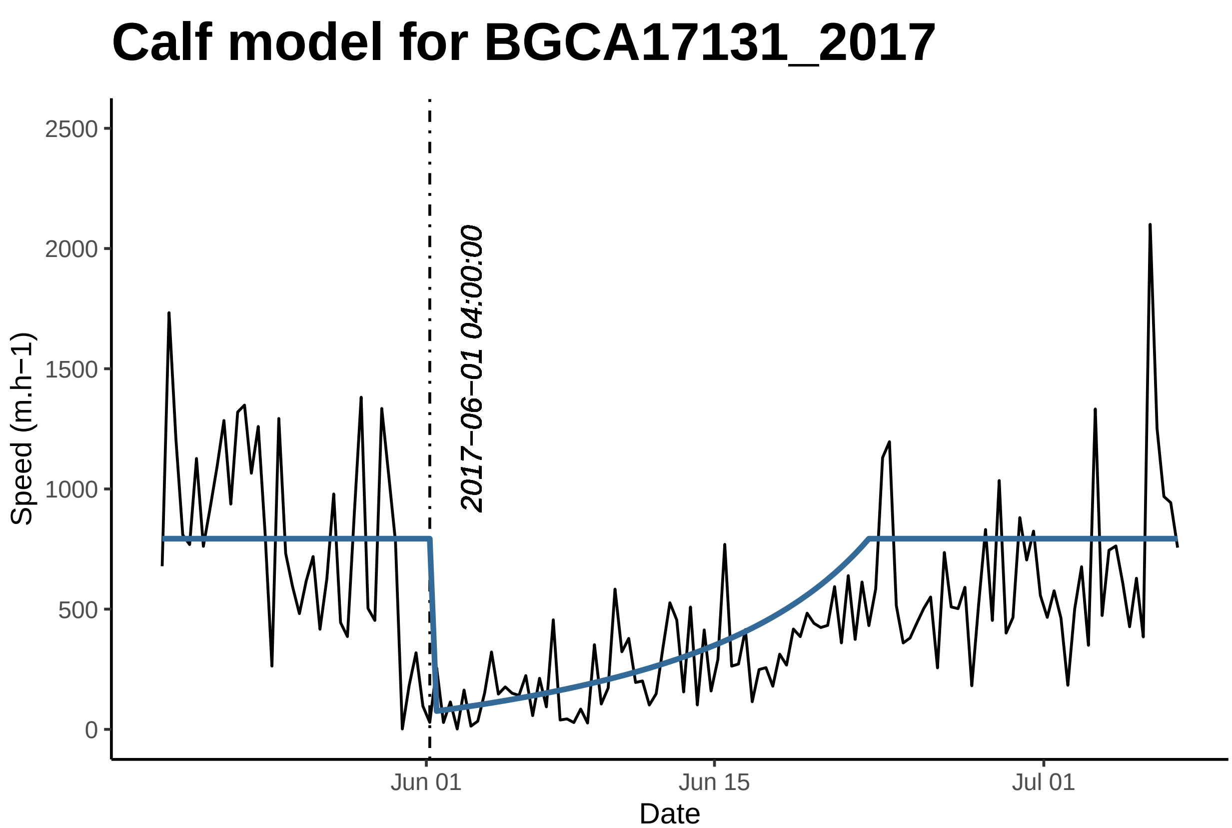
The function returns a list with the AIC of the 2 or 3 models (i.e. no calf model, calf model and calf death model), the best model based on AIC, the dates of calf and death if any Recovery time (in days) for each calf model, Par, the estimated parameters of the best model based on AIC and a summary of the results with the best model based on AIC, the calving date if any, the mortality date of the calf (if any), the recovery time (in days, if any), a Z-score for the calving date, as a deviation from average calving date from Cameron et al. 2018 and the x and y calving. locations.

Estimate calving on all females

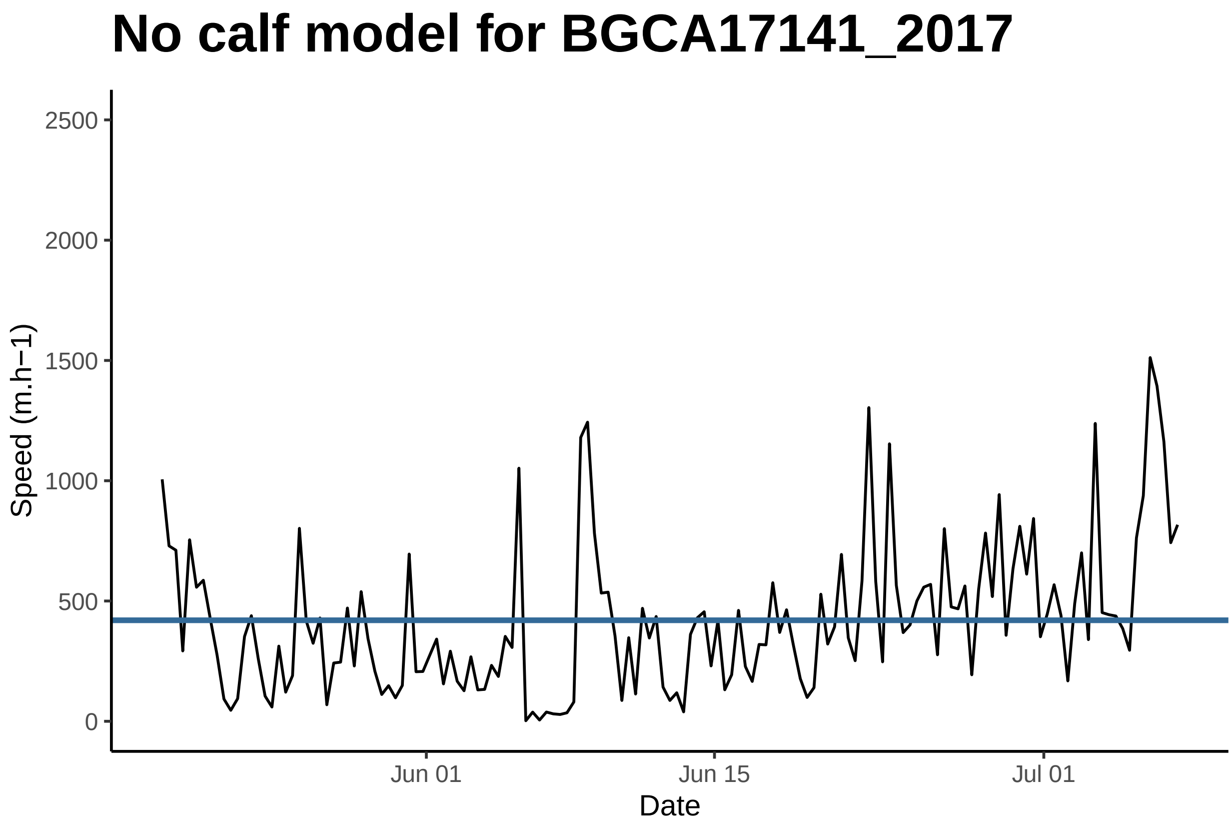
Here is an example on two females

```
# Will generate a sample of two different individuals each time
part <- parturition.model2(df = c1, int=3, kcons=c(5,21), PlotIt=T, saveplot=F)
```

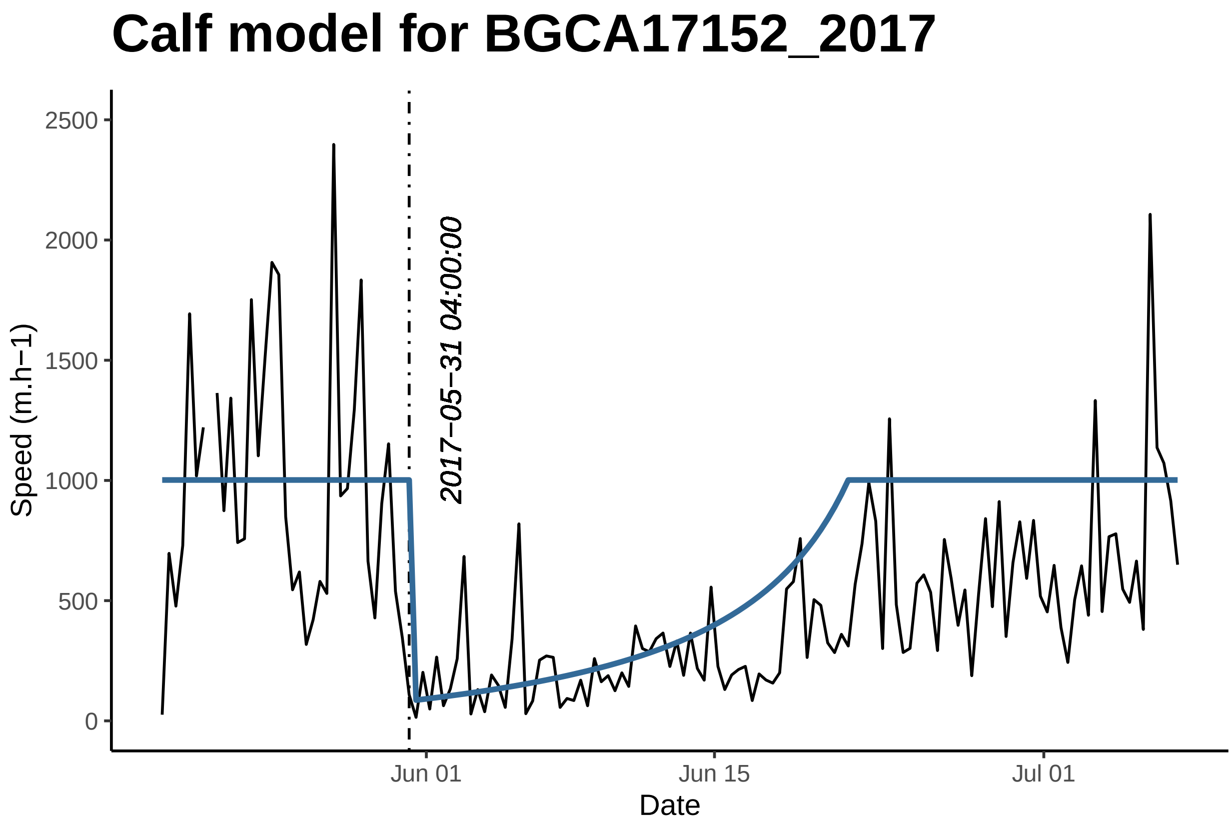
```
## [1] "BGCA17131_2017"
```



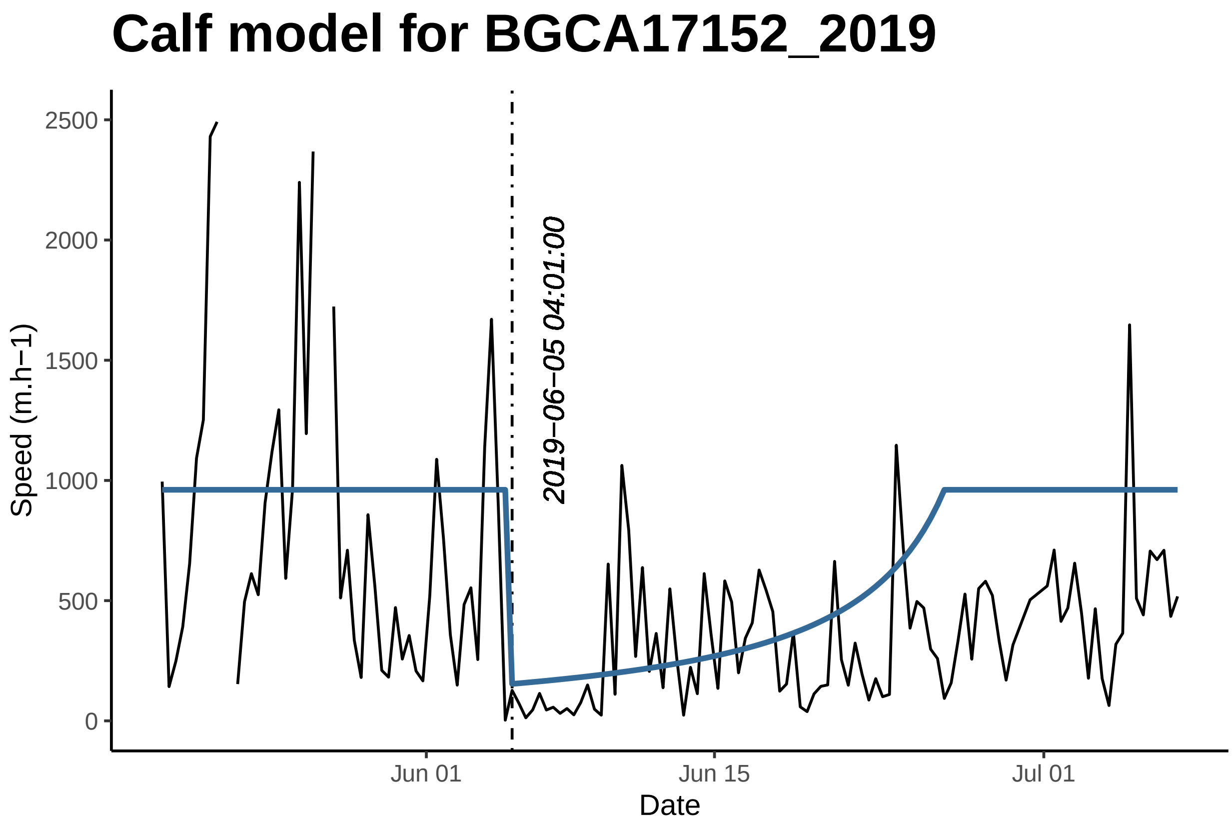

```
## [1] "BGCA17141_2017"
```



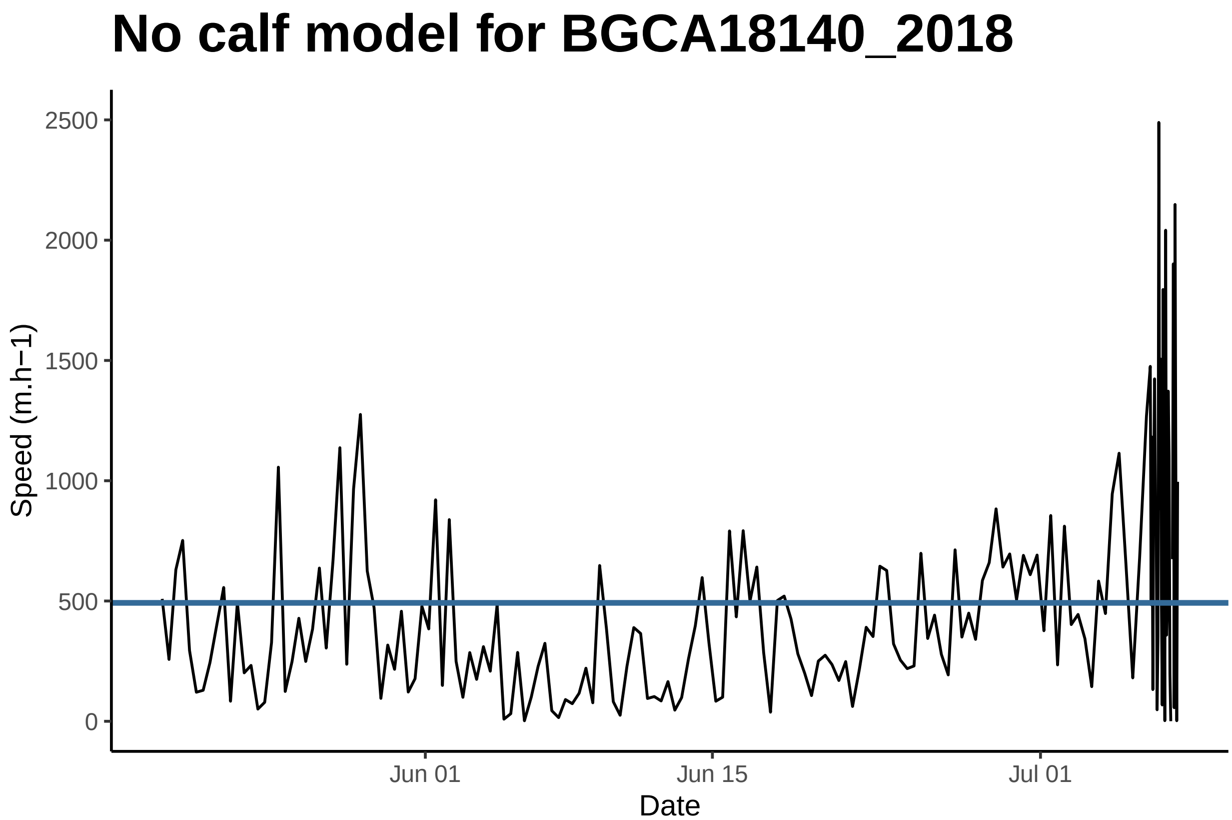

```
## [1] "BGCA17152_2017"
```




```
## [1] "BGCA17152_2019"
```




```
## [1] "BGCA18140_2018"
```



The object *part* is a list of 4 data frames:

- one containaing the Best model selected based on the AIC, the estimated calving date, (calf death date if the 'parturition.model' has been used) and recovery time (in days), for each individual
- one containing the AIC of each model as well as the calving date (calf death date if 'parturition.model' has been used) and the recovery time (in days)
- one containing the estimated parameters
  - alpha.mean: shape mean
  - alpha.calf: shape at the calving date
  - beta.mean: scale mean
  - beta.calf: scale at the calving date
  - recovery: time to rcover normal speed (in hours)
- one containing a summary of the results of the analysis for each individuals with
  - the best model based on the AIC
  - the calving date
  - the calf death date
  - the recovery time (in days)
  - calving date score: a z-score based on the average calving date in the Western Arctic Herd (Cameron et al. 2018), which calculates the deviation from the average calving date. A negative score means that the female calved before the average calving date, and vice versa. The higher the deviation from the average date is (negatively or positively), the less probable it is
  - the calving location coordinates (x, y) (in the same project than provided)

ID_Year	Best.Model	M0.AIC	Mcalf.AIC	M0.mnll	Mcalf.mnll
BGCA17131_2017	calf	2167.076	2089.290	-1081.538	-1039.645
BGCA17141_2017	nocalf	2086.240	2103.517	-1041.120	-1046.758
BGCA17152_2017	calf	2179.466	2118.445	-1087.733	-1054.222
BGCA17152_2019	calf	2151.822	2117.651	-1073.911	-1053.825
BGCA18140_2018	nocalf	2567.061	2586.115	-1281.530	-1288.057

ID_Year	alpha.0	beta.0	alpha.mean1	beta.mean1	alpha.calf1	beta.calf1	BP.calf1	calving.da
BGCA17131_2017	1.8588422	0.0035238	3.584089	0.0045192	1	0.0131630	13	2017-06-0
BGCA17141_2017	1.8444134	0.0043923	2.001204	0.0051007	1	0.0102013	7	2017-05-2
BGCA17152_2017	1.3723089	0.0024262	2.700643	0.0026956	1	0.0115853	12	2017-05-3
BGCA17152_2019	0.7761781	0.0013846	1.188776	0.0012368	1	0.0065040	17	2019-06-0
BGCA18140_2018	0.9899890	0.0020110	1.844900	0.0044741	1	0.0089481	15	2018-06-0

ID_Year	Best.Model	calving.date	Recovery	calving.date.score	calf.loc.x	calf.loc.y
BGCA17131_2017	calf	2017-06-01 04:00:00	21	-0.7042889	470656.9	231493.9
BGCA17141_2017	nocalf	NA	NA	NA	NA	NA
BGCA17152_2017	calf	2017-05-31 04:00:00	21	-0.9300226	472705.4	240646.9
BGCA17152_2019	calf	2019-06-05 04:01:00	21	0.1986456	459887.1	243723.1
BGCA18140_2018	nocalf	NA	NA	NA	NA	NA