

Vignette

Ophélie Couriot

12/14/2022

Contents

| | |
|---|----------|
| Estimate parturition in caribou using the TuktuTools Package | 1 |
| Clean data | 1 |
| Focus on the calving period | 2 |
| Get movement rate | 3 |
| Estimate calving status and timing | 5 |

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) Time: 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 +datum=NAD83 +units=m +no_defs")
```

Arguments

df a data frame containing columns: ID as individual identifier, Year, x and y: relocations of individuals (in NAD83 utm zone 19N) Time: 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 minimum 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.

```
data("caribou")
b <- caribou %>% removeOutliers
```

```
## [1] "Cleaning Step 1"
## [1] "Number of removed 'outliers': 5"
## [1] " "
## [1] "Cleaning Step 2"
## [1] "Number of removed 'outliers': 2"
## [1] " "
```

```
b <- b$no.outliers
```

```
c <- b %>% cutperiod(start=c(05,19), end = c(07,07), nfixes=1, dayloss = 10)
```

```
## [1] "Period comprised between May 19 and Jul 7"
## [1] "Number of excluded individuals: 0"
```

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 developed 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 **getSpeed** function in this package.

Usage

```
getSpeed(df, 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, Year, x and y: relocations of individuals (in N Canada Lambert Conformal Conic), Time: 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”)

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 *caribou* contains movement data during the calving period for 4 barren-ground caribou females.

```
c1 <- getSpeed(c)
head(c1)
```

| ## | ID | sex | Time | Lon | Lat | x | y | Month |
|------|------------|----------|---------------------|-----------|----------|-----------|----------|-------|
| ## 1 | Comet | f | 2006-05-19 00:01:00 | -128.4005 | 65.86132 | -378326.7 | 119110.3 | 5 |
| ## 2 | Comet | f | 2006-05-19 08:00:00 | -128.4882 | 65.87401 | -382075.3 | 121005.5 | 5 |
| ## 3 | Comet | f | 2006-05-19 16:01:00 | -128.5320 | 65.88125 | -383929.0 | 122052.1 | 5 |
| ## 4 | Comet | f | 2006-05-20 00:00:00 | -128.5929 | 65.91993 | -386101.6 | 126639.3 | 5 |
| ## 5 | Comet | f | 2006-05-20 08:00:00 | -128.6764 | 65.96338 | -389200.3 | 131883.4 | 5 |
| ## 6 | Comet | f | 2006-05-20 16:00:00 | -128.6409 | 65.97951 | -387384.5 | 133437.7 | 5 |
| ## | start.date | | end.date | dt | yday | speed | | |
| ## 1 | 2006-05-19 | 00:01:00 | 2006-07-07 17:00:00 | 7.983333 | 139 | 526.1588 | | |
| ## 2 | 2006-05-19 | 00:01:00 | 2006-07-07 17:00:00 | 8.016667 | 139 | 265.5360 | | |
| ## 3 | 2006-05-19 | 00:01:00 | 2006-07-07 17:00:00 | 7.983333 | 139 | 635.7797 | | |
| ## 4 | 2006-05-19 | 00:01:00 | 2006-07-07 17:00:00 | 8.000000 | 140 | 761.3963 | | |
| ## 5 | 2006-05-19 | 00:01:00 | 2006-07-07 17:00:00 | 8.000000 | 140 | 298.7743 | | |
| ## 6 | 2006-05-19 | 00:01:00 | 2006-07-07 17:00:00 | 8.000000 | 140 | 120.6265 | | |
| ## | time.mid | | dhours | | | | | |
| ## 1 | 2006-05-19 | 04:00:30 | 0.00000 | | | | | |
| ## 2 | 2006-05-19 | 12:00:30 | 8.00000 | | | | | |
| ## 3 | 2006-05-19 | 20:00:30 | 16.00000 | | | | | |
| ## 4 | 2006-05-20 | 04:00:00 | 23.99167 | | | | | |
| ## 5 | 2006-05-20 | 12:00:00 | 31.99167 | | | | | |
| ## 6 | 2006-05-20 | 20:00:00 | 39.99167 | | | | | |

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 lost her calf, as well as the calving date and the calf death date.

As stated previously, we adapted the individual based method developed 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: $(shape_{calving} * (\frac{shape - shape_{calving}}{k}) * time) * (scale_{calving} * (\frac{scale - scale_{calving}}{k}) * time)$ 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 model 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 of each speed value and the difference in hours between each speed value and the first one and a vector of the animal-year identifier. See `?getSpeed` 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 recommended 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 = TRUE` the function will draw a plot of the speed in function to the date with the prediction line of 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.

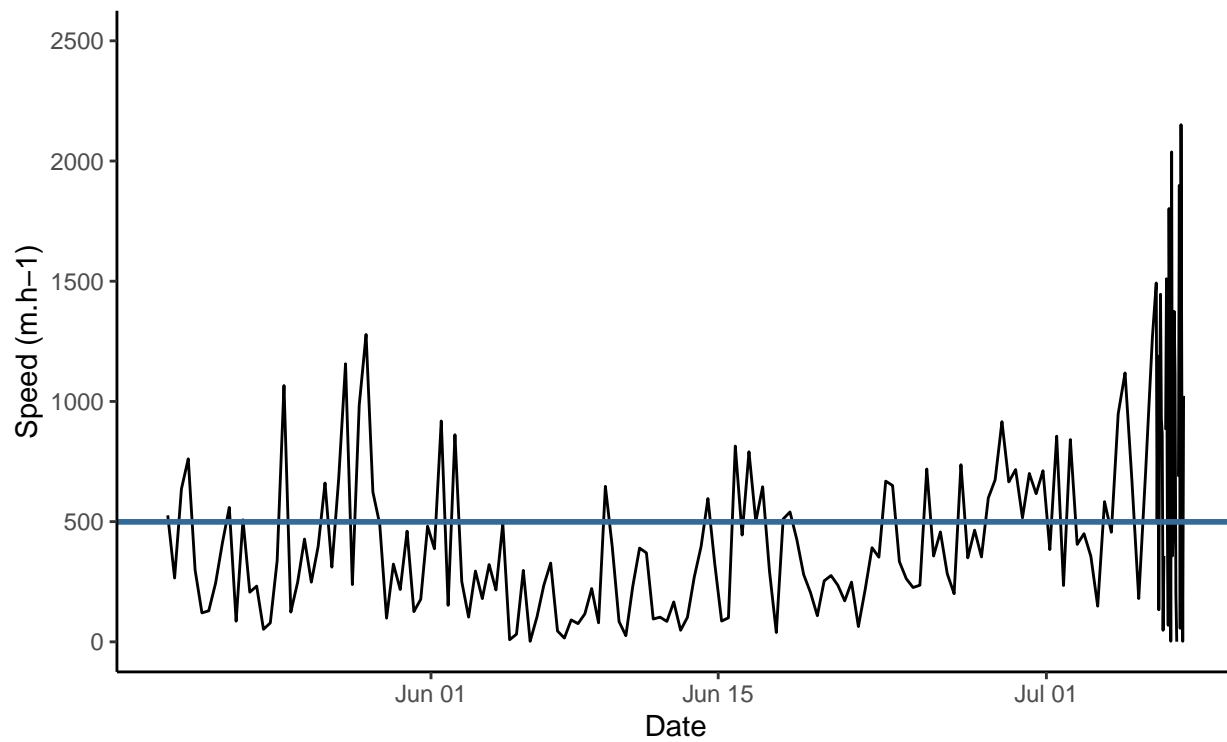
Estimate calving on all females

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=TRUE, saveplot=FALSE)
```

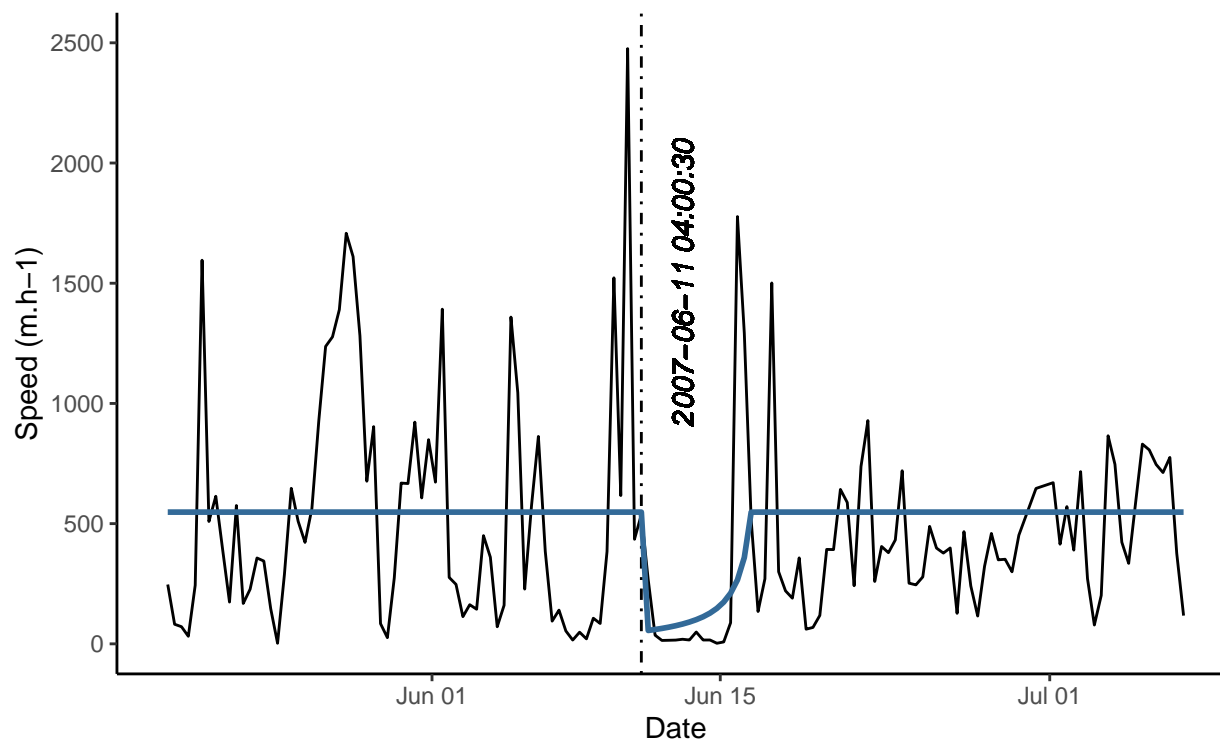
```
## [1] "Comet_2006"
```

No calf model for Comet_2006



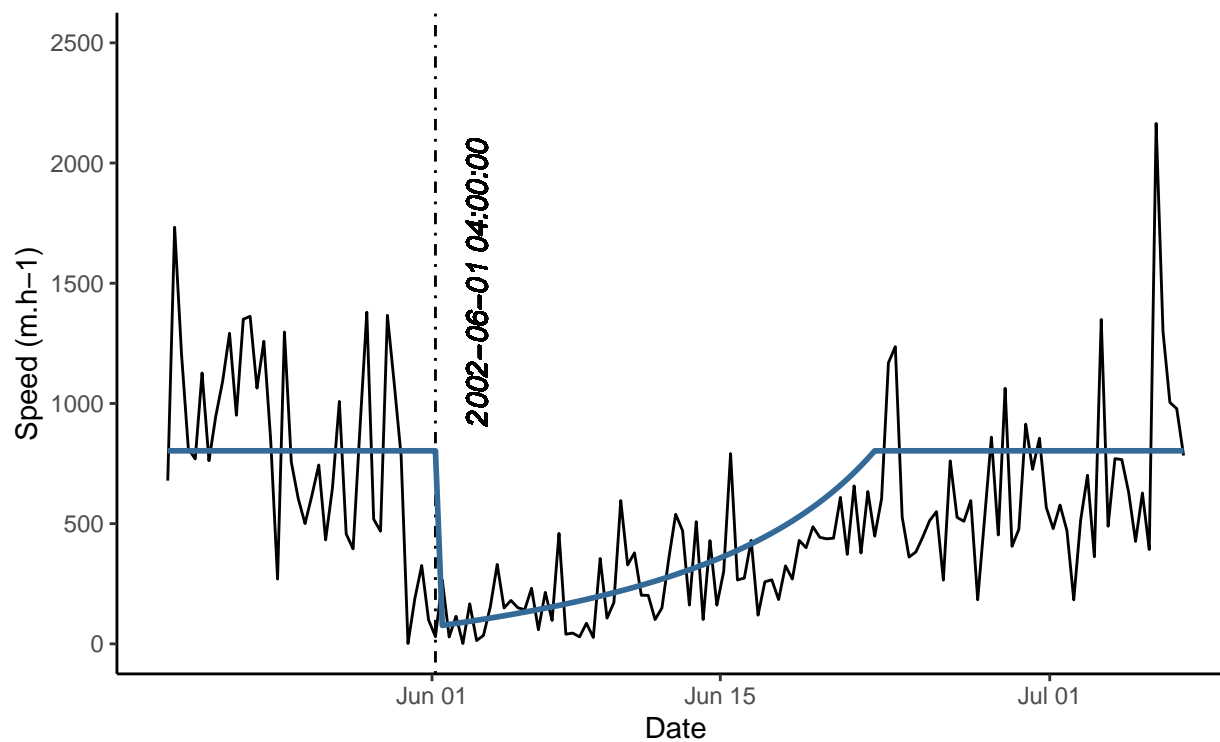
```
## [1] "Comet_2007"
```

Calf model for Comet_2007



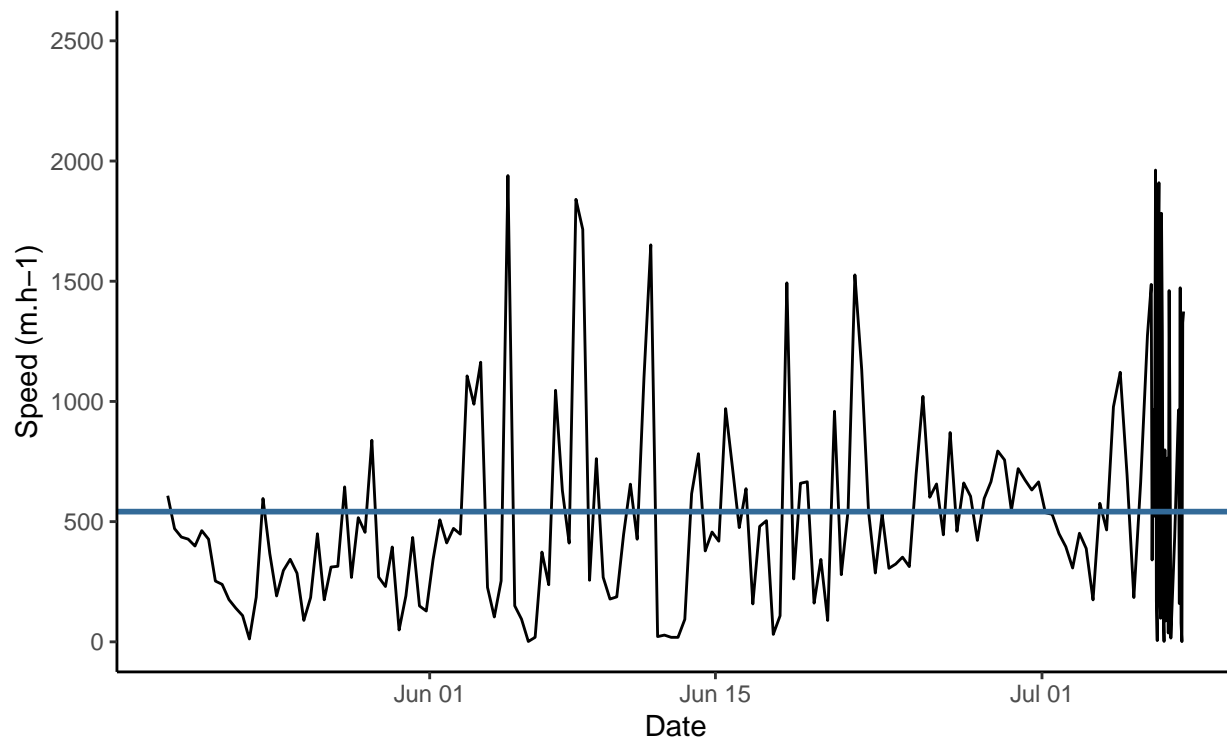
```
## [1] "Dancer_2002"
```


Calf model for Dancer_2002



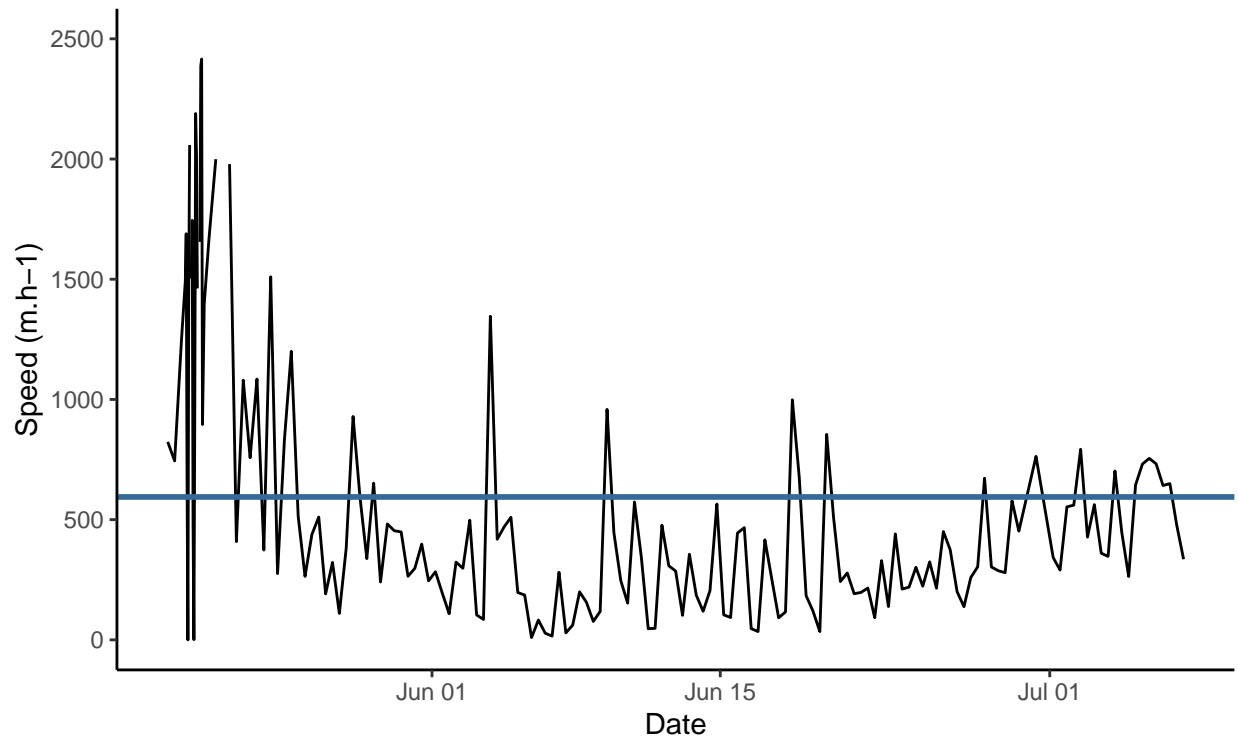
```
## [1] "Dancer_2003"
```

No calf model for Dancer_2003



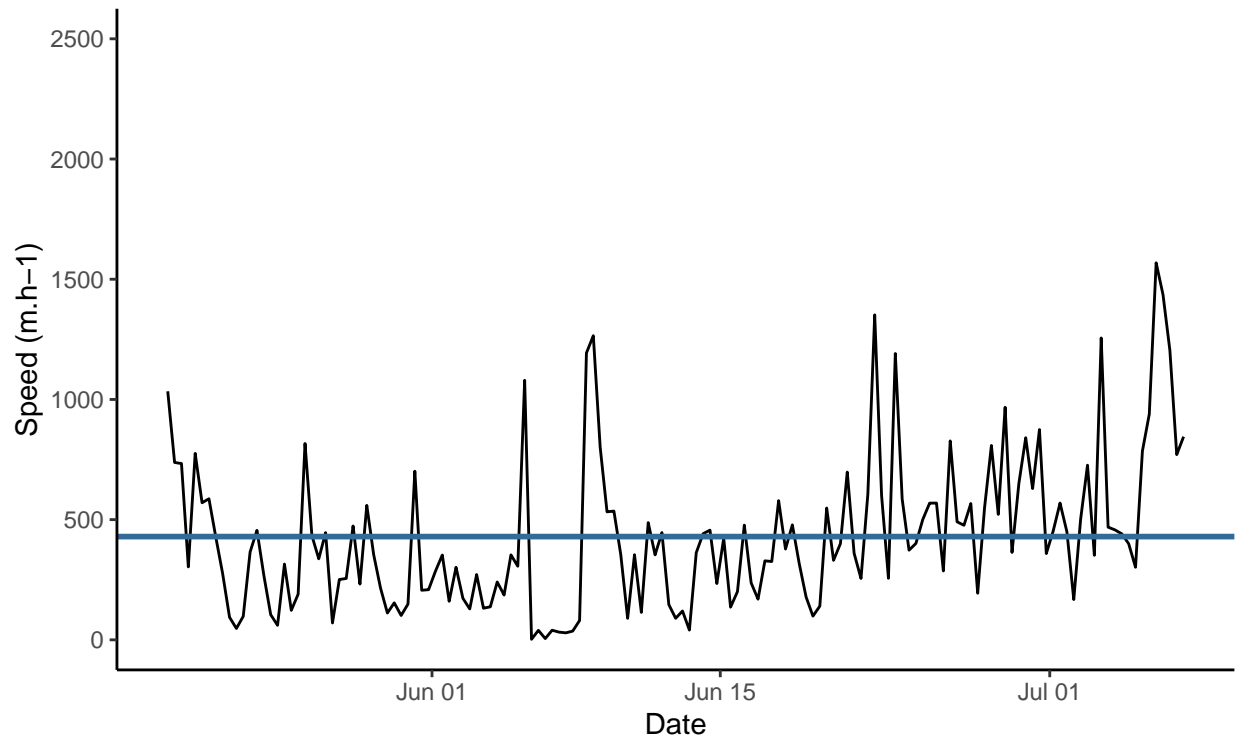
```
## [1] "Dancer_2004"
```

No calf model for Dancer_2004



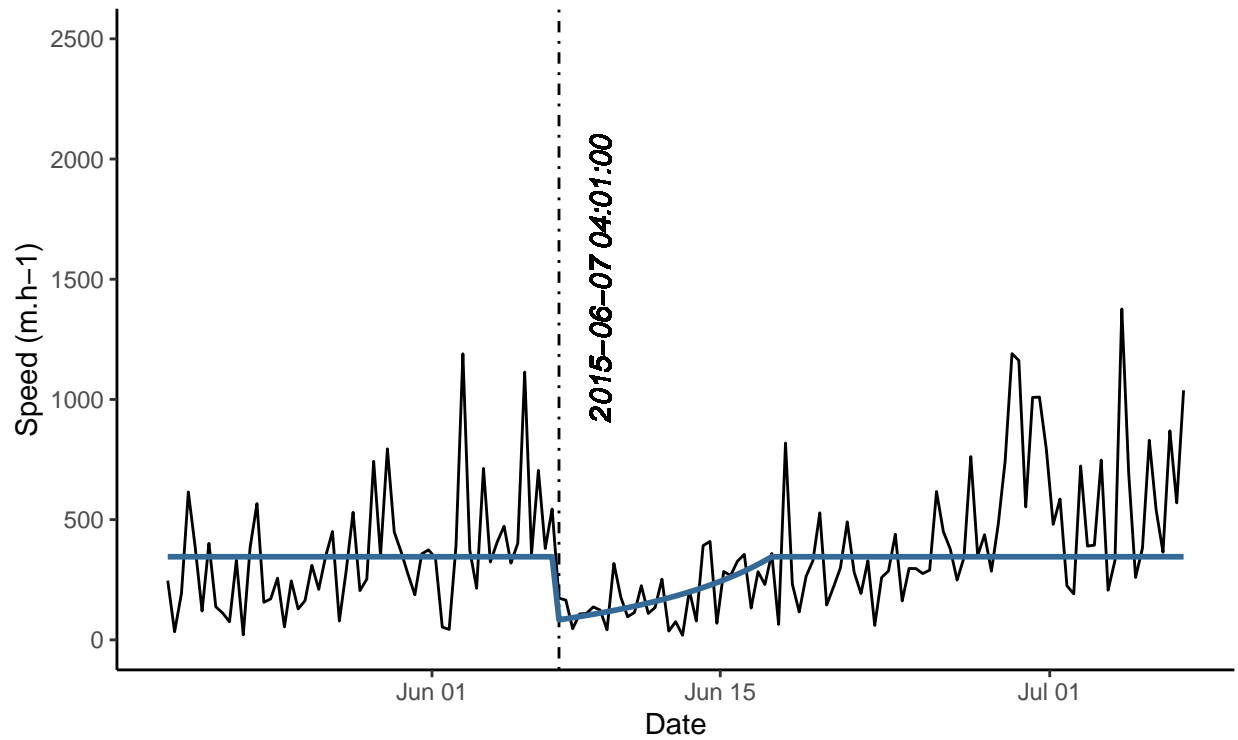
```
## [1] "Prancer_2014"
```

No calf model for Prancer_2014



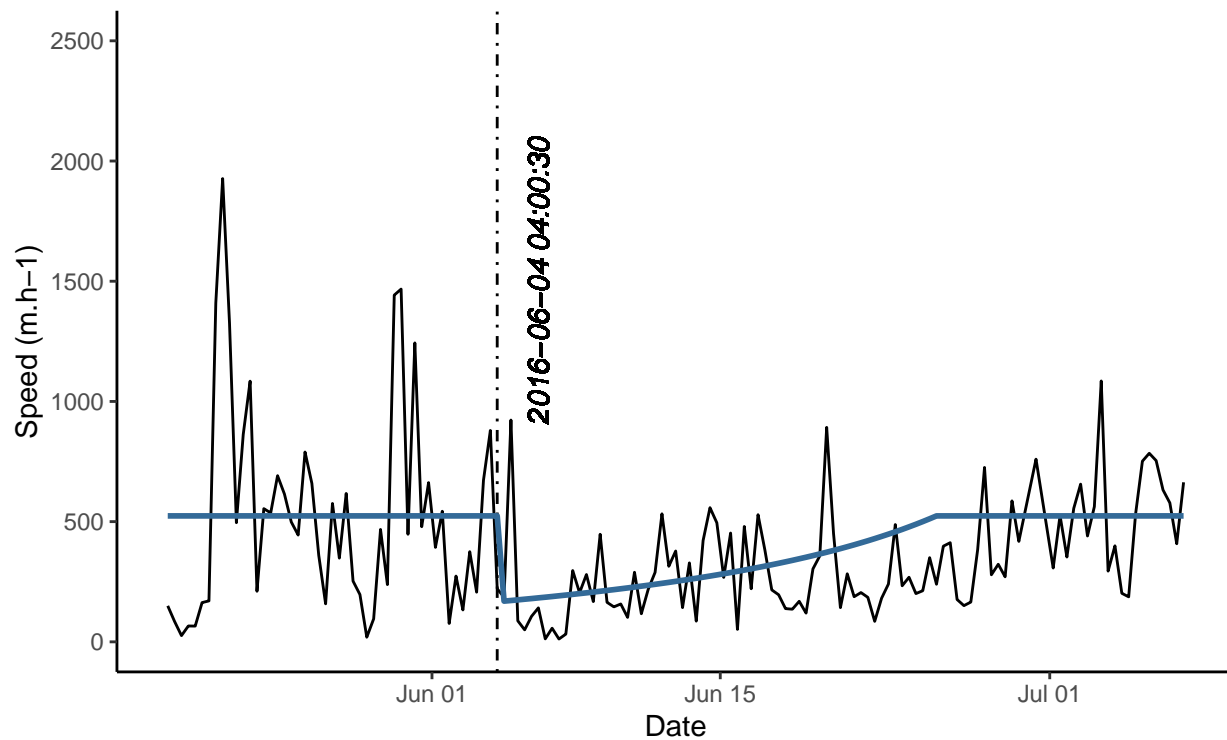
```
## [1] "Prancer_2015"
```

Calf model for Prancer_2015



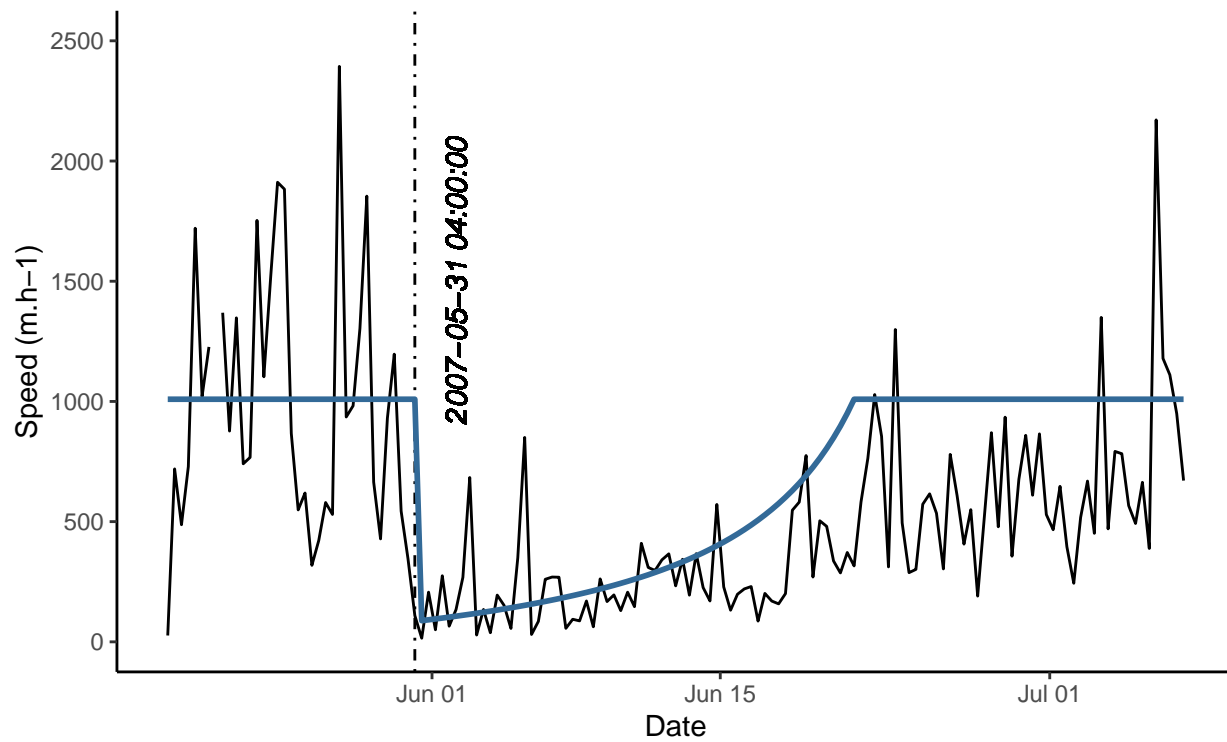
```
## [1] "Prancer_2016"
```

Calf model for Prancer_2016



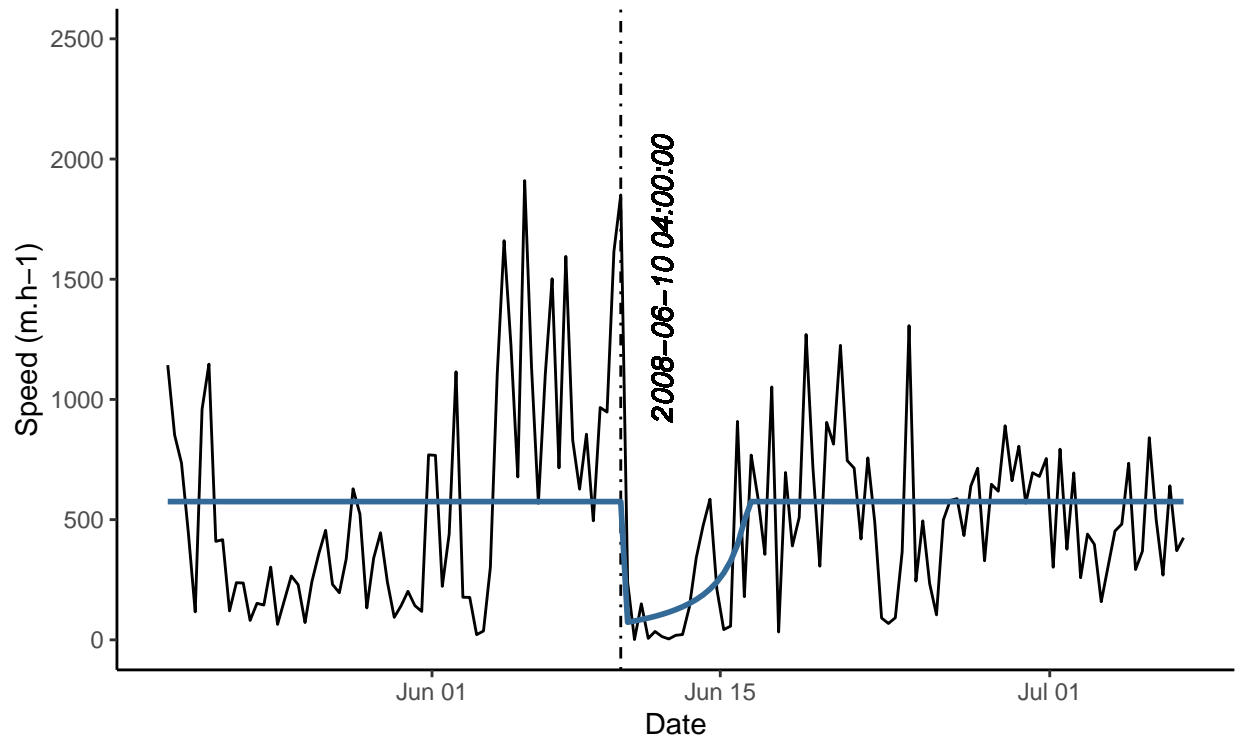
```
## [1] "Vixen_2007"
```

Calf model for Vixen_2007



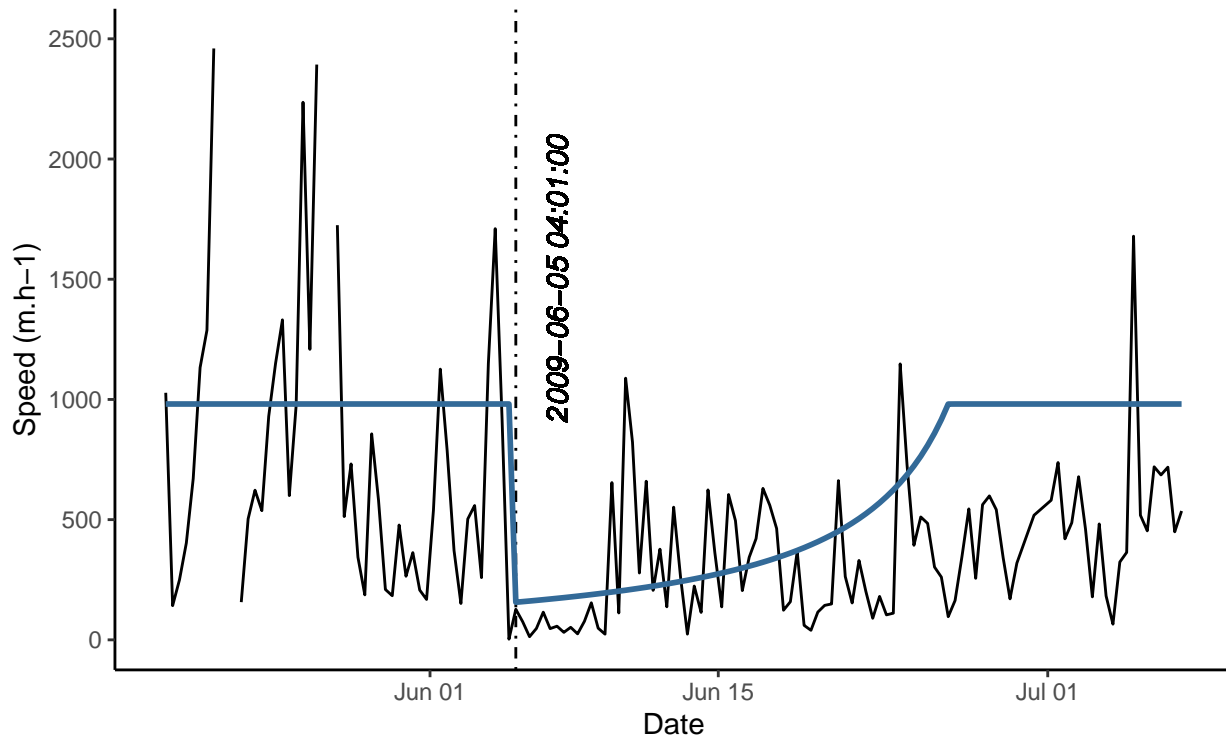
```
## [1] "Vixen_2008"
```

Calf model for Vixen_2008



```
## [1] "Vixen_2009"
```


Calf model for Vixen_2009



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

- one containing 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 recover 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)

\$coeffs

```

##      ID_Year      ID Year Best.Model    M0.AIC Mcalf.AIC    M0.mnll Mcalf.mnll
## 1  Comet_2006  Comet 2006    nocalf 2571.571 2591.141 -1283.786 -1290.570
## 2  Comet_2007  Comet 2007     calf 2094.951 2074.976 -1045.476 -1032.488
## 3  Dancer_2002 Dancer 2002     calf 2172.876 2094.953 -1084.438 -1042.476
## 4  Dancer_2003 Dancer 2003    nocalf 2589.066 2596.082 -1292.533 -1293.041
## 5  Dancer_2004 Dancer 2004    nocalf 2443.880      NA -1219.940      NA
## 6  Prancer_2014 Prancer 2014    nocalf 2093.033 2108.979 -1044.516 -1049.490
## 7  Prancer_2015 Prancer 2015     calf 2030.848 2016.675 -1013.424 -1003.337
## 8  Prancer_2016 Prancer 2016     calf 2025.376 2017.182 -1010.688 -1003.591
## 9   Vixen_2007  Vixen 2007     calf 2183.890 2122.607 -1089.945 -1056.303
## 10  Vixen_2008  Vixen 2008     calf 2166.031 2132.615 -1081.016 -1061.308
## 11  Vixen_2009  Vixen 2009     calf 2157.692 2123.400 -1076.846 -1056.700
##
## $par
##      ID Year      alpha.0      beta.0 alpha.mean1  beta.mean1 alpha.calf1
## 1  Comet 2006 1.0063171 0.002016636    1.856118 0.004438177      1
## 2  Comet 2007 1.1567511 0.002456545    1.127544 0.002057608      1
## 3  Dancer 2002 1.8696387 0.003477401    3.662273 0.004560918      1
## 4  Dancer 2003 1.4856275 0.002743134    1.171732 0.002595860      1
## 5  Dancer 2004 0.9070836 0.001526156      NA      NA      NA
## 6  Prancer 2014 1.8280855 0.004256245    2.334704 0.005017785      1
## 7  Prancer 2015 1.8236568 0.005008929    2.069779 0.005987478      1
## 8  Prancer 2016 1.5116464 0.003805367    1.417805 0.002706086      1
## 9   Vixen 2007 1.3871350 0.002413968    2.719623 0.002694995      1
## 10  Vixen 2008 1.6398207 0.003176486    1.376718 0.002393949      1
## 11  Vixen 2009 0.7773956 0.001358667    1.190649 0.001213788      1
##      beta.calf1 BP.calf1      calving.date1 recovery1
## 1  0.008876354      15 2006-06-03 04:00:00 319.9919
## 2  0.017943508      23 2007-06-11 04:00:30 120.0000
## 3  0.013023939      13 2002-06-01 04:00:00 504.0000
## 4  0.005191720      24 2003-06-12 04:00:00 120.0000
## 5      NA      NA      <NA>      NA
## 6  0.010035570       4 2014-05-23 04:00:00 504.0000
## 7  0.011974956      19 2015-06-07 04:01:00 247.9842
## 8  0.005903958      16 2016-06-04 04:00:30 504.0000
## 9  0.011341444      12 2007-05-31 04:00:00 504.0000
## 10 0.013640607      22 2008-06-10 04:00:00 140.5728
## 11 0.006393034      17 2009-06-05 04:01:00 504.0000
##
## $results
##      ID_Year      ID Year Best.Model      calving.date Recovery calf.loc.x
## 1  Comet_2006  Comet 2006    nocalf      <NA>      NA      NA
## 2  Comet_2007  Comet 2007     calf 2007-06-11 04:00:30      5 -384895.0
## 3  Dancer_2002 Dancer 2002     calf 2002-06-01 04:00:00     21 -414718.5
## 4  Dancer_2003 Dancer 2003    nocalf      <NA>      NA      NA
## 5  Dancer_2004 Dancer 2004    nocalf      <NA>      NA      NA
## 6  Prancer_2014 Prancer 2014    nocalf      <NA>      NA      NA
## 7  Prancer_2015 Prancer 2015     calf 2015-06-07 04:01:00     10 -425089.5
## 8  Prancer_2016 Prancer 2016     calf 2016-06-04 04:00:30     21 -397031.9
## 9   Vixen_2007  Vixen 2007     calf 2007-05-31 04:00:00     21 -409893.1
## 10  Vixen_2008  Vixen 2008     calf 2008-06-10 04:00:00      6 -391770.5
## 11  Vixen_2009  Vixen 2009     calf 2009-06-05 04:01:00     21 -421668.8
##      calf.loc.y
## 1      NA

```

| | | |
|----|----|-----------|
| ## | 2 | 148300.64 |
| ## | 3 | 111261.02 |
| ## | 4 | NA |
| ## | 5 | NA |
| ## | 6 | NA |
| ## | 7 | 127582.41 |
| ## | 8 | 132940.89 |
| ## | 9 | 119344.99 |
| ## | 10 | 88470.37 |
| ## | 11 | 126170.94 |