Smoothing methods for ARGOS trajectories

Projet long presentation

2015

Projet Long

- Project in collaboration with **CLS**
- Supervisor: **Beatriz Calmettes**, CLS
- Project leader: Jérôme Combanière
- Project team:
  - Jérôme Combanière
  - Anthony Delannoy
  - Benoit Madiot

CLS
COLLECTE LOCALISATION SATELLITES

Context

- Endangered species:
  - Leatherback turtles
  - Elephant seals

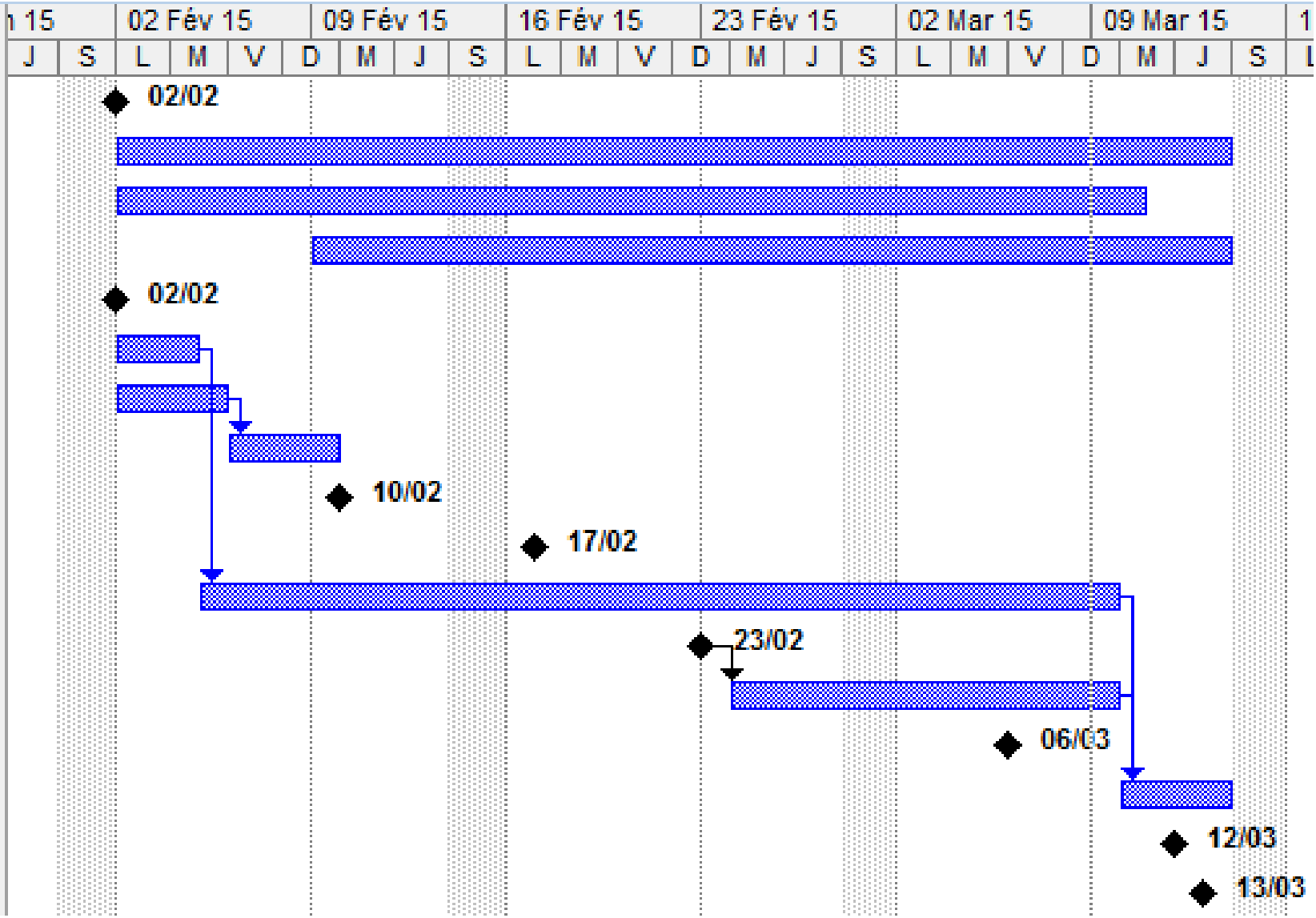
- ARGOS system ⟹ monitoring threatened species
- Creation of marine protected areas
- Matlab ⟹ Python

Contents

- Project management
- Graphical User Interface (GUI)
- Data extraction and common format
- Data processing
- Conclusion

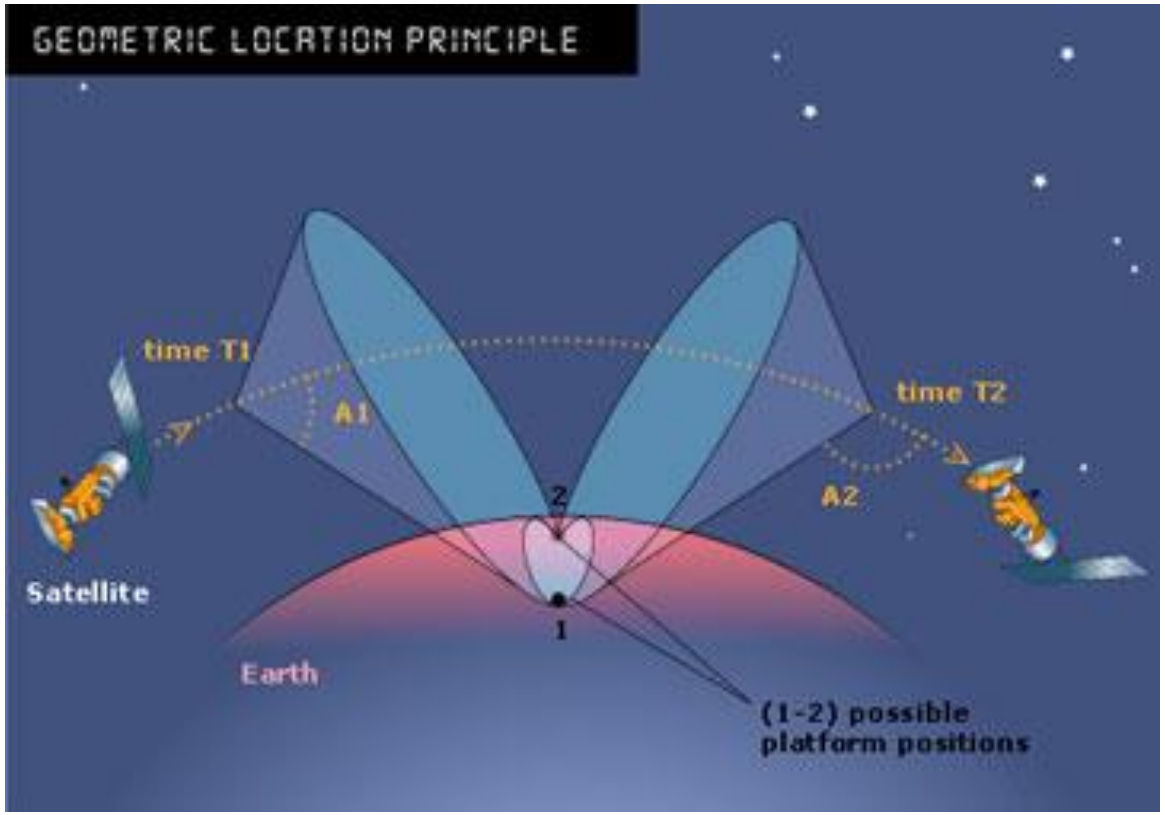
GANTT chart

1	Start of projet long
2	Projet long
3	Update of Gantt chart
4	Report preparation
5	1st meeting with supervisor
6	Familiarization with GitHub / Python / Projet
7	Setting up of the management plan
8	Audit preparation
10	Audit
11	2nd meeting with supervisor
12	Translation Matlab->Python
13	3rd meeting with supervisor
14	Further assignments
15	Mock presentation with supervisor
9	Abstract / oral preparation
16	Oral presentation
17	End of projet long



Graphical User Interface (GUI)

ARGOS system



Data extraction and common format

- ARGOS data stored in three different file formats :
  - CSV
  - DIAG
  - DS
- One program to rule them all, one program to find them, one program to bring them all and in the format bind them
- The remaining format is a list of dictionary

$$\left[\ dico[1] \quad dico[2[\quad \dots \quad dico[n] \ \right]$$

Data extraction and common format

- Each transmission data are stored in a dictionary with a unique keys structure

$$\left\{ \begin{array}{c} "date" \\ "LC" \\ "lat" \\ "lon" \\ "lat\_image" \\ "lon\_image" \end{array} \right\}$$

- Key "date" associates to another dictionary

$$\left\{ \begin{array}{c} "annee" \\ "mois" \\ "jour" \\ "heure" \\ "min" \\ "sec" \end{array} \right\}$$

Data extraction and common format

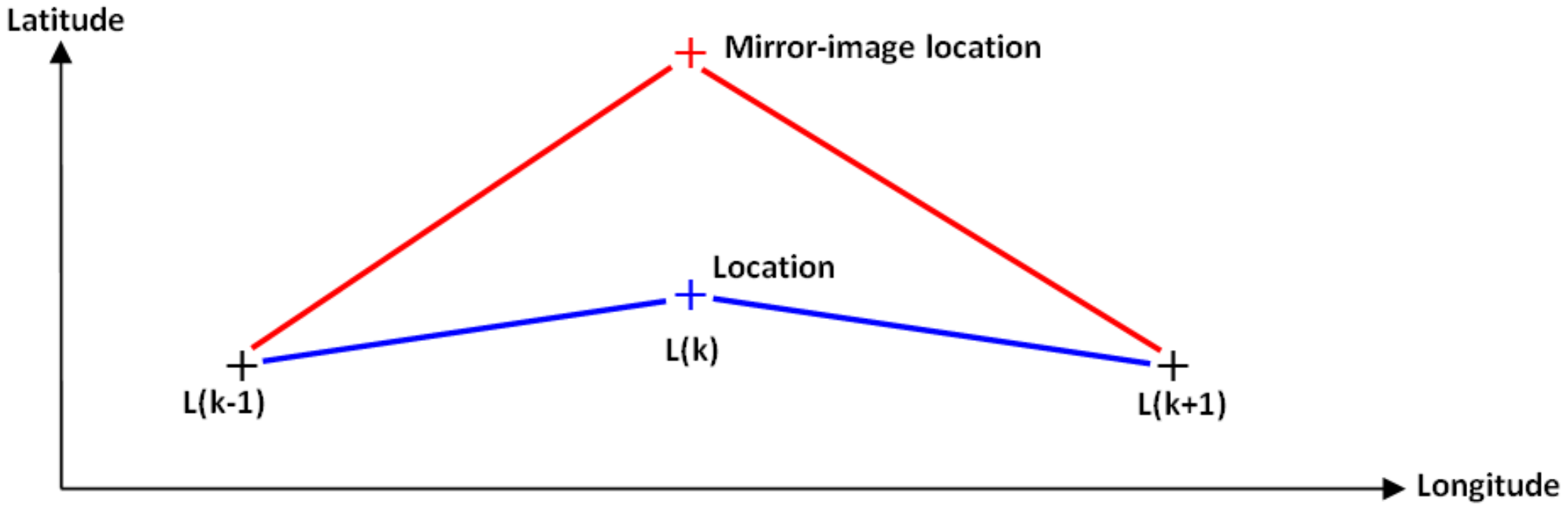
- This unique work format allowed an easier way to program smoothing methods
- XML files contain parameters for smoothing methods and are specific to each species
- These parameters are also stored in a dictionary following XML reading.

Data processing

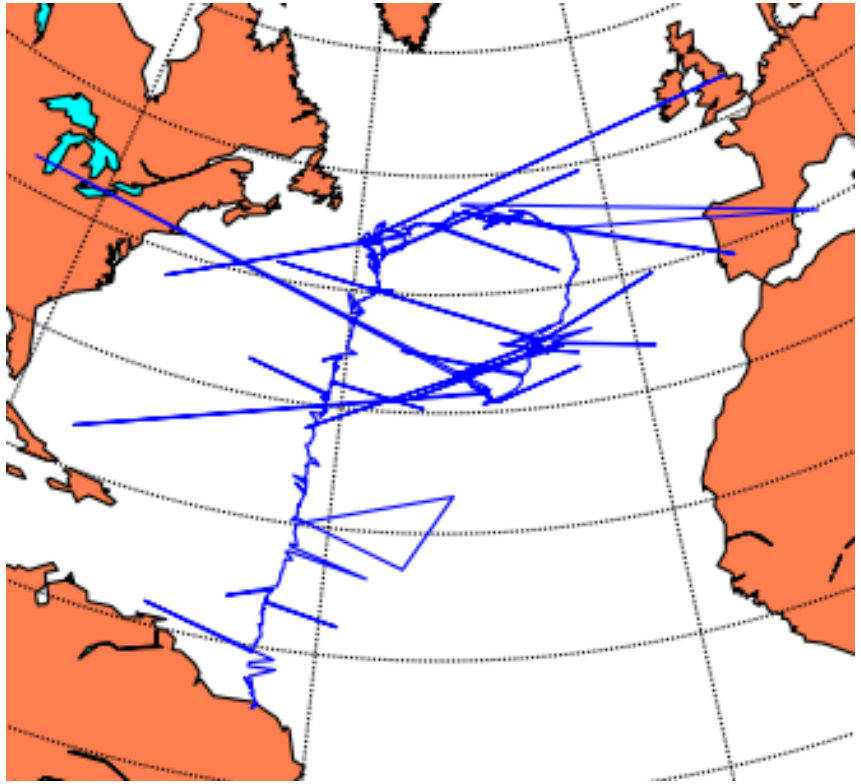
Preprocessing

CHOICE OF LOCATION

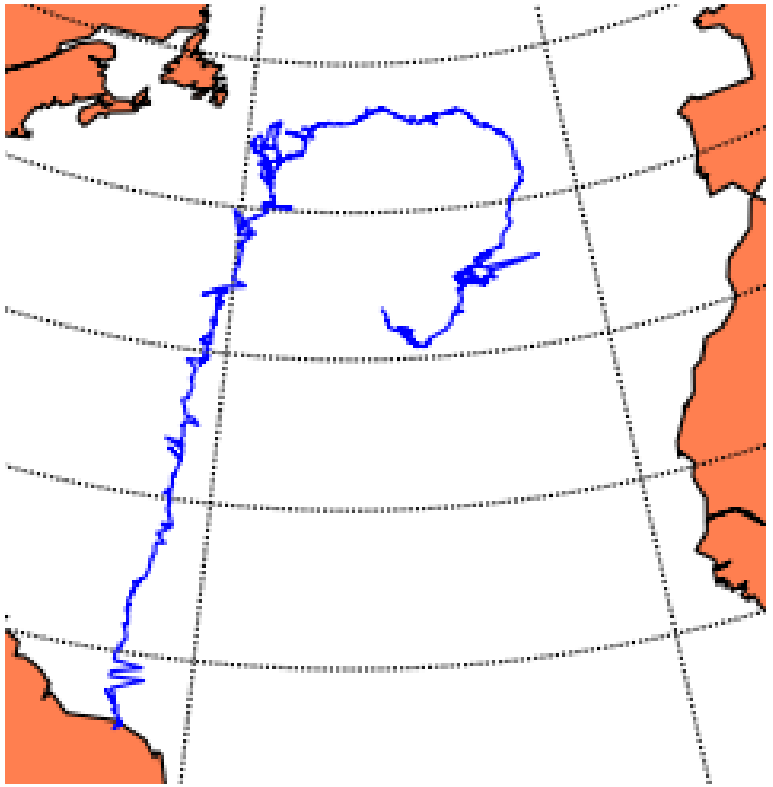
Choice of location



Choice of location



Raw data



Data preprocessed with correction of location

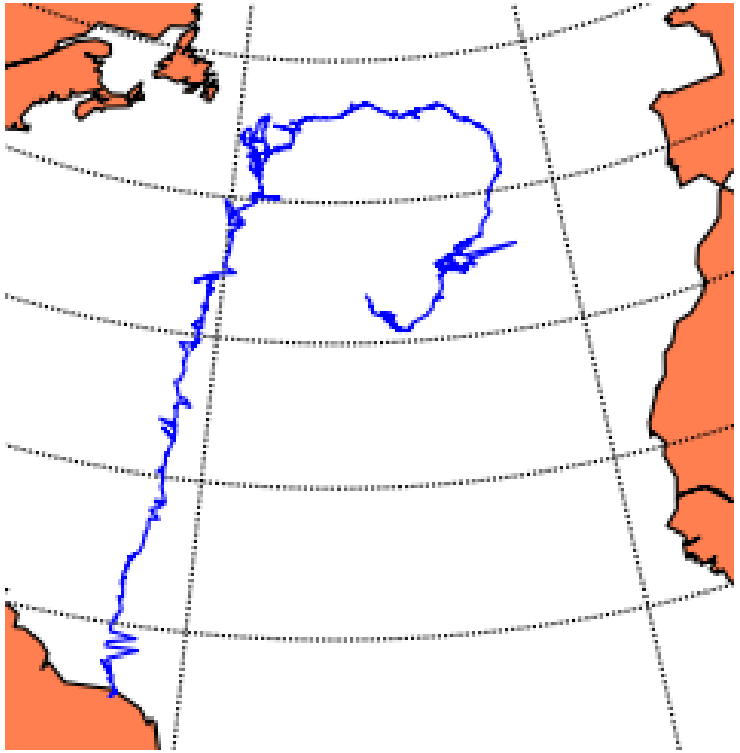
Preprocessing

DELETION OF EXCESSIVE SPEED

Deletion of excessive speed

- Computation of speed between two points
- Criteria of precision of the location
- Comparison with the specie’s maximal speed

Deletion of excessive speed

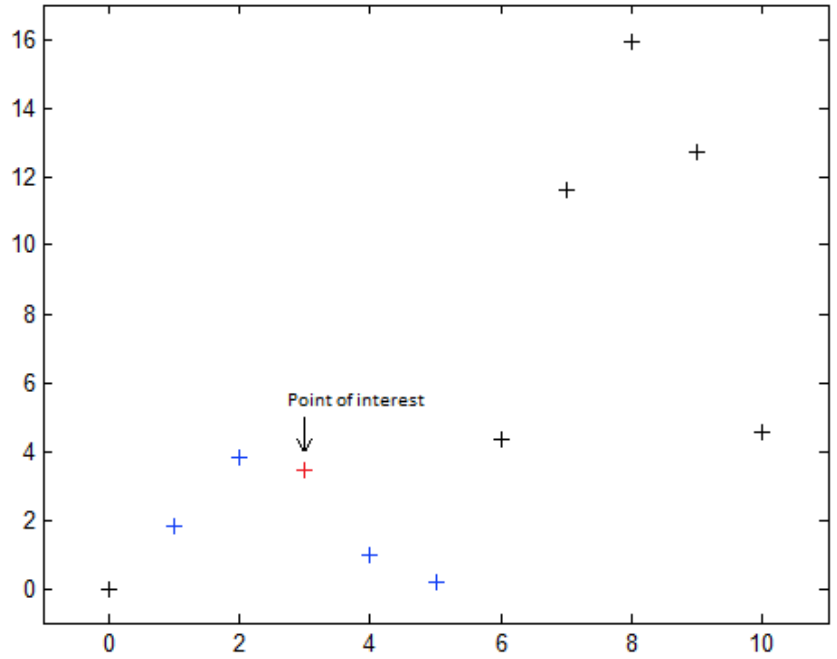


Data before deletion of excessive speed

Data after deletion of excessive speed

Data processing

- Here you estimate one position as the weighted sum of the two previous, current and two following positions



2 different weights :

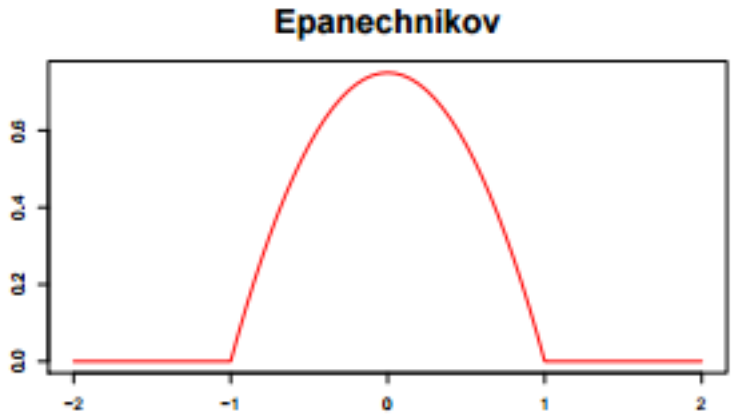
- one from the kernel
- one from the quality of the ARGOS localization

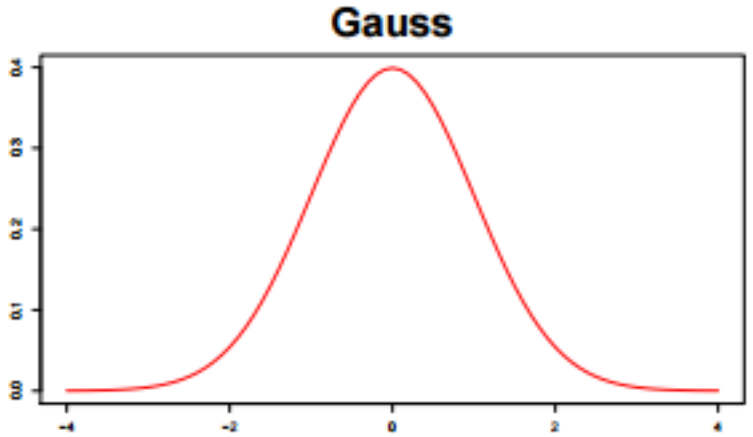
Data processing

- Adaptable size of the support of the epanechnikov kernel :

$\frac{3}{4h}.\left(1-\frac{x^2}{h}\right)$ with 2h = size of the support

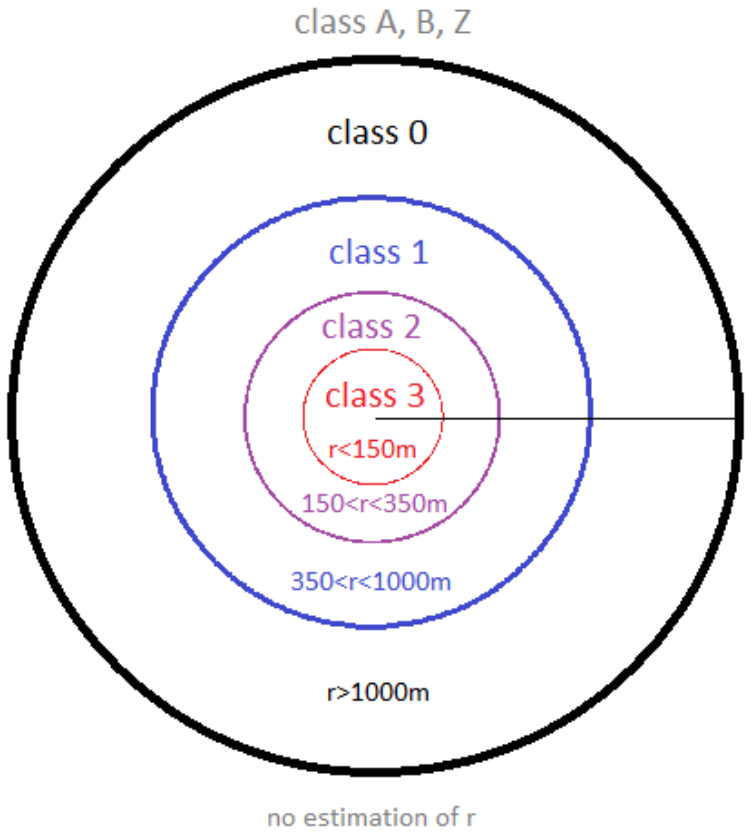
- Epanechnikov kernel minimizes AMISE (Asymptotic Mean Integrated Squared Error) and is therefore optimal.





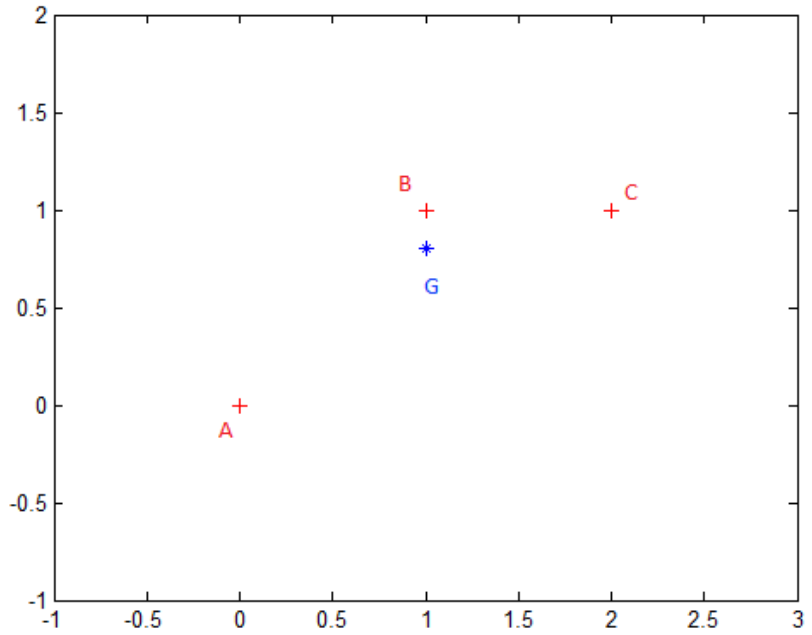
Data processing

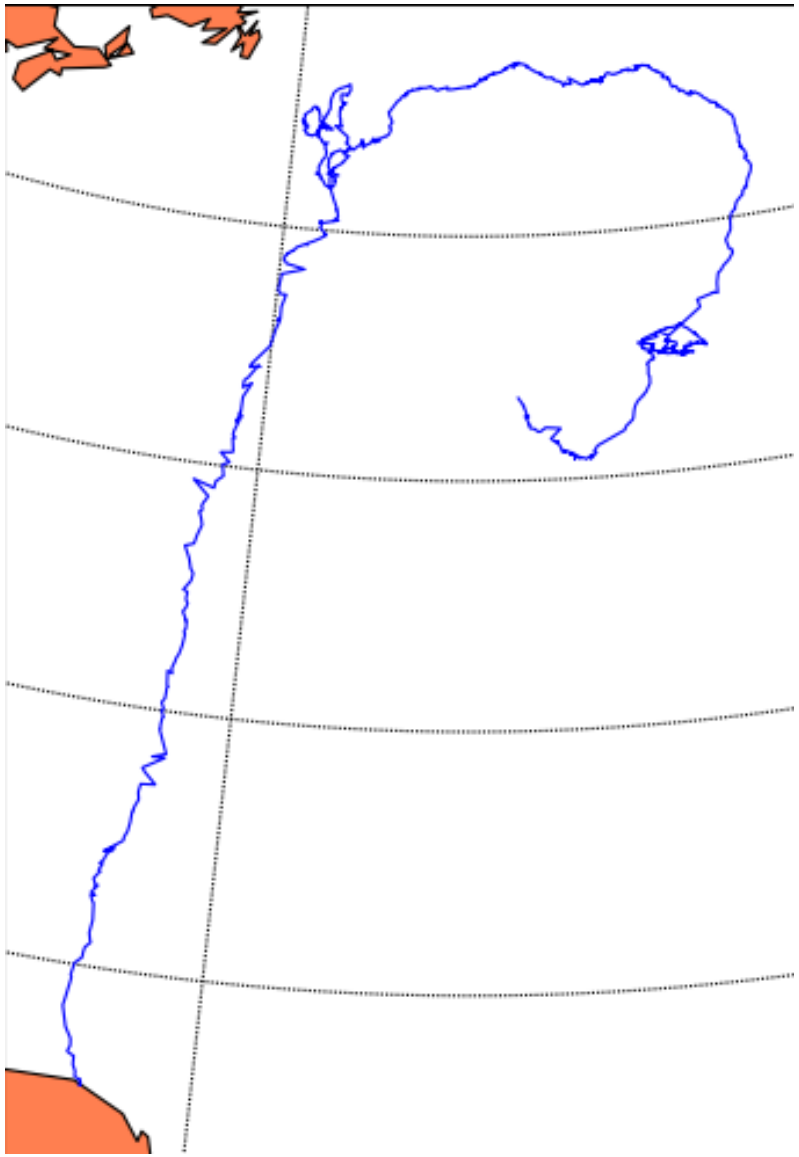
- The weights increase as the precision of measurement increases



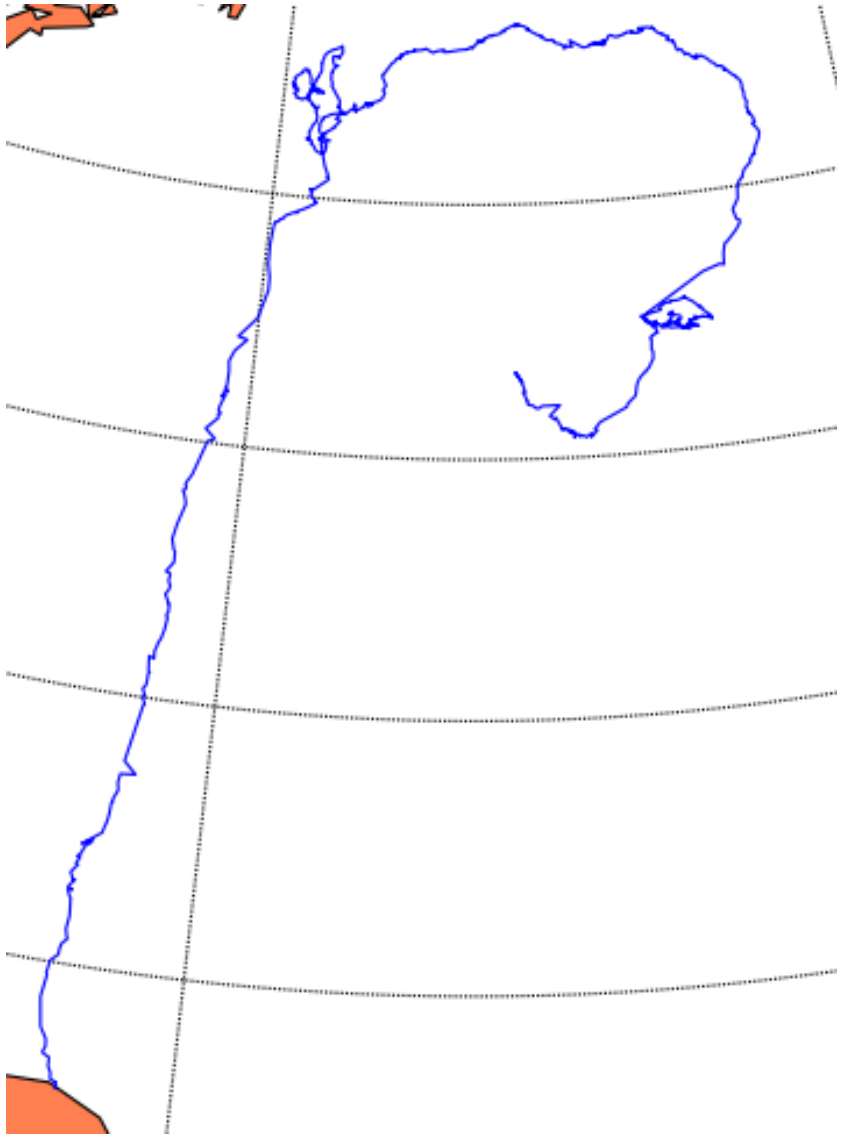
Data processing

If the estimated position is too far from the ARGOS position, this position is removed

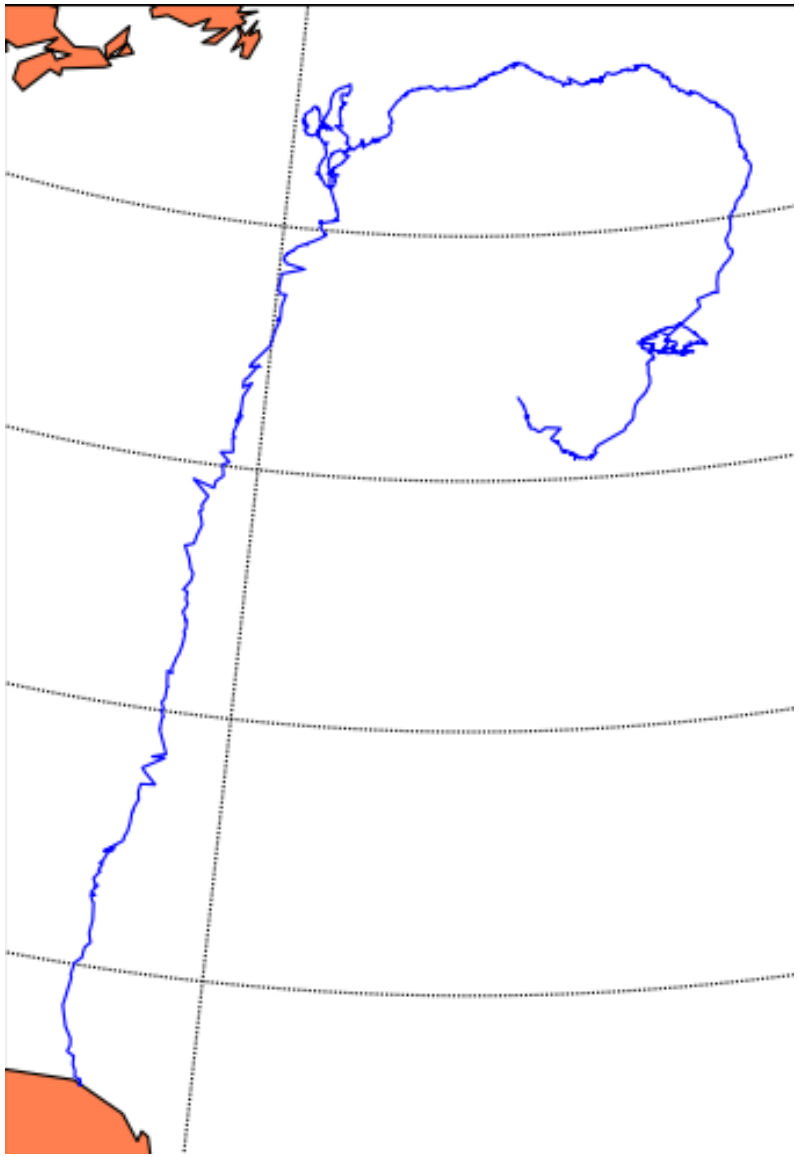




Trajectory before the estimation



Trajectory with an Epanechnikov kernel

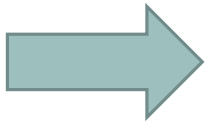


Trajectory before the estimation

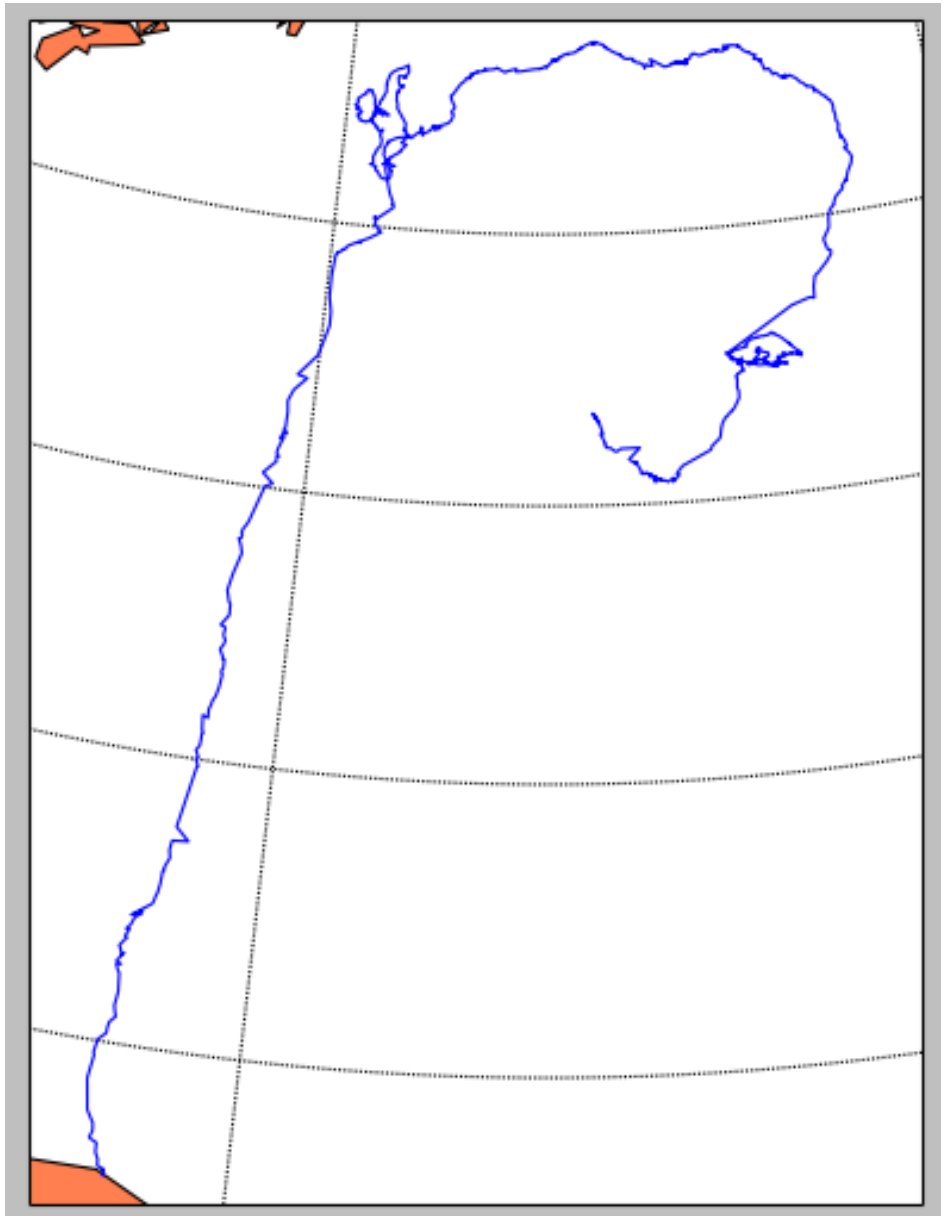
Trajectory with a Gaussian kernel

2nd estimation

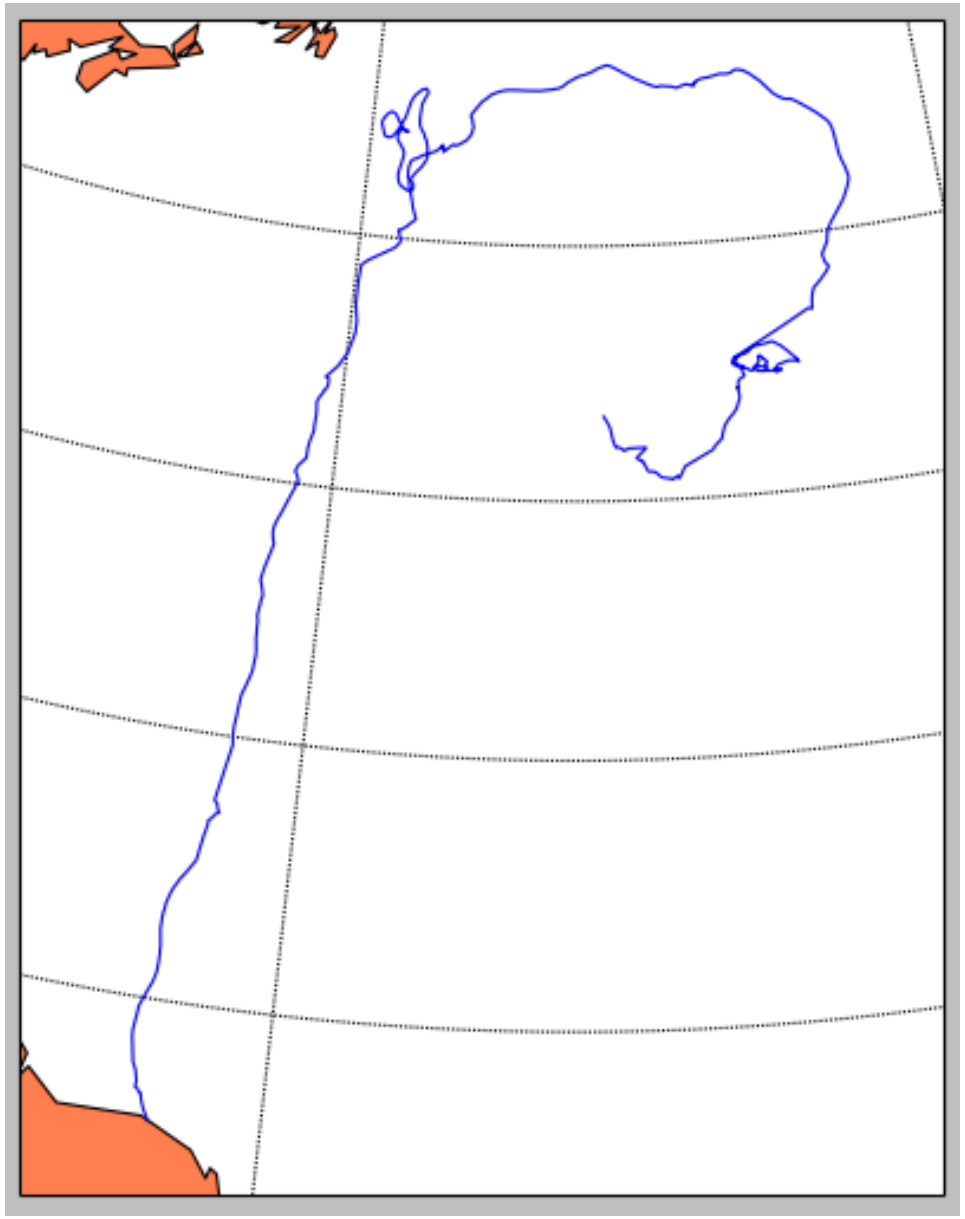
- Linear regression
- Resampling trajectory



Points spaced with a constant time step size



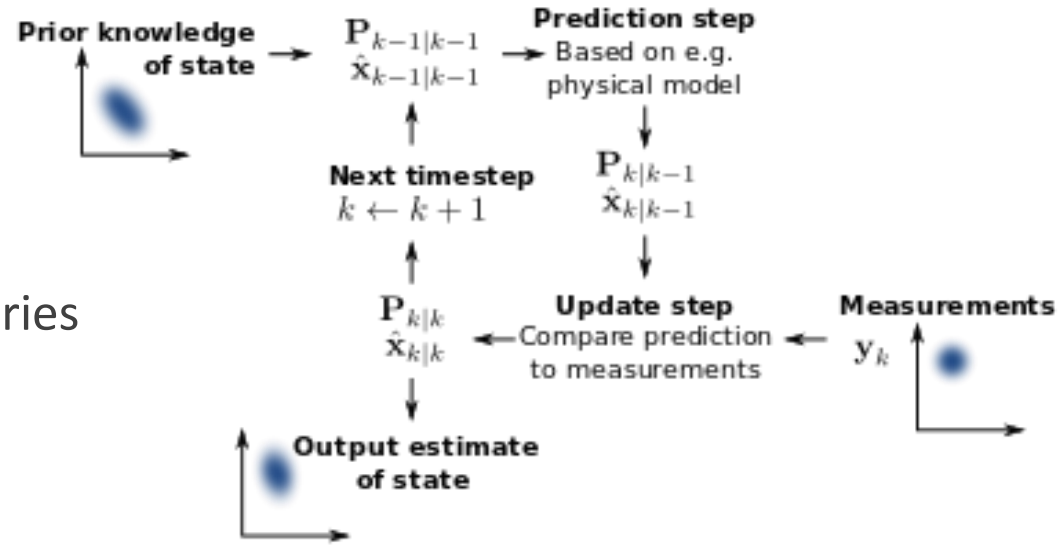
Output without the 2nd estimation

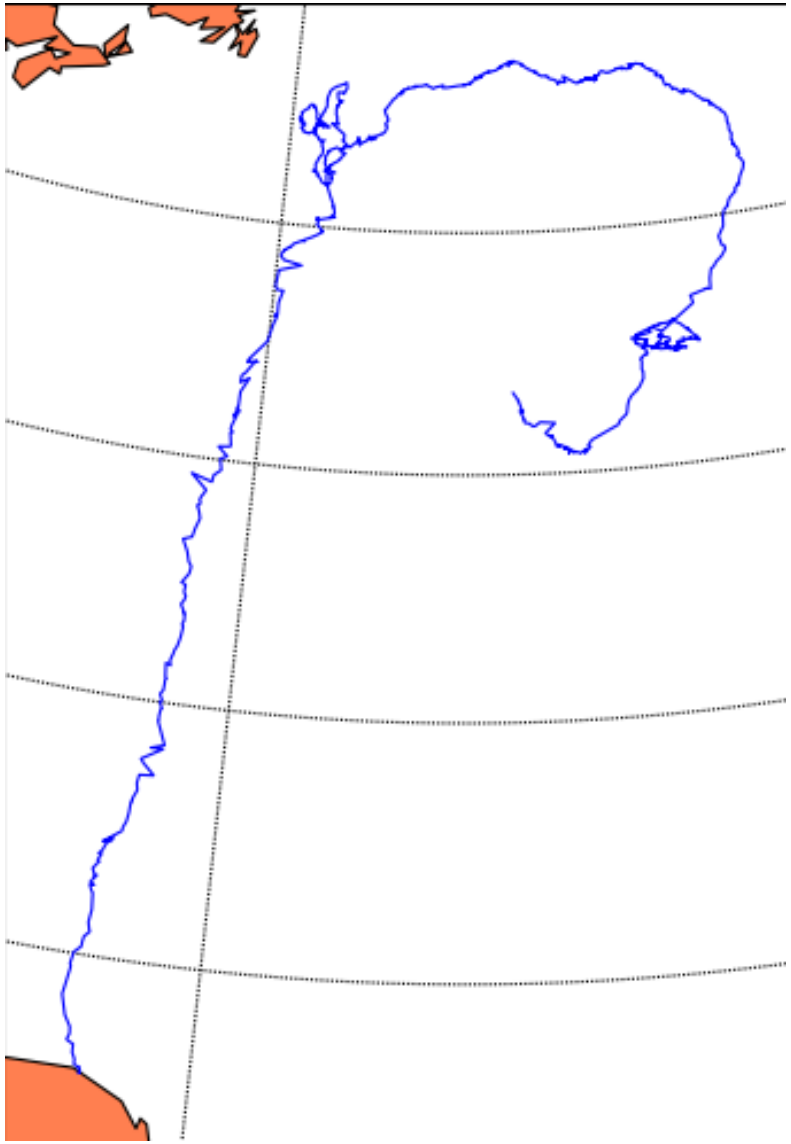


Output with the 2nd estimation

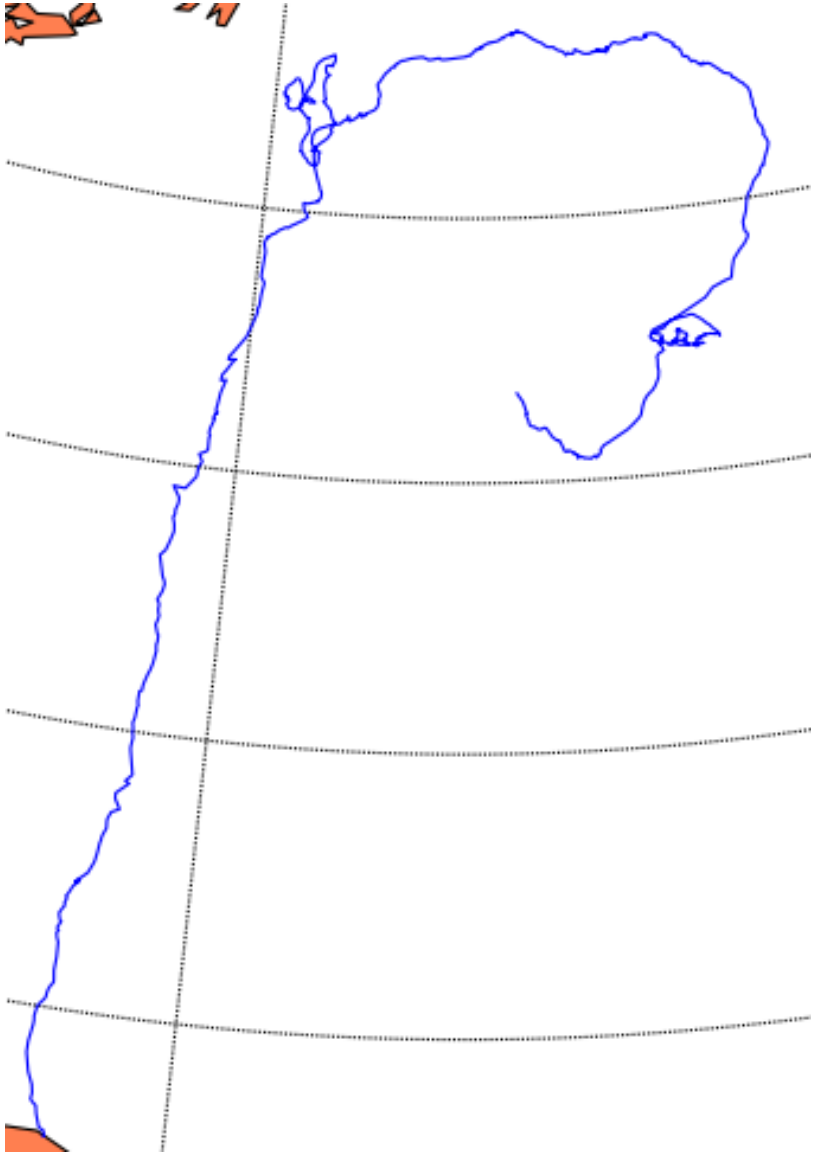
Kalman smoothing

- An EM algorithm estimates the transition matrix
- Use of all the data in order to smooth the trajectories





Trajectory before Kalman smoothing



Trajectory after Kalman smoothing

Conclusion

- Efficient and reliable algorithms
- Work achieve intends to monitor endangered species
- Trajectories can be plotted and exploited using the GUI
- Further improvements:
  - Comparison with GPS data
  - Handle new ARGOS data