ATLAS - Data filtering.

Introduction:

There have been many requests and questions regarding data filtering in the past and I am going to shed some light on these. As you may or may not know there are already variety of filters in Kamadata. One question in particular has been why filtered data in Kamadata cannot be downloaded. The answer is simply that the data is not stored anywhere but only exists in the memory of your computer the reason for this is twofold:

1. filtering the data online is simply highly inefficient (having to wait for queries and storing all the filtered data in memory is just not very feasible)

2. filtered data would have to be labeled after download explaining what filters have already been applied to the data, which could lead to confusion later down the line.

I do believe its in the best interest of everyone that the data is stored in an unfiltered raw format and that filters are only applied during actual analysis to ensure that the researcher analyzing the data is at any given time fully aware of the filters being applied to his data. As the method of filtering and the specific parameters used during filtering have a strong impact on the actual data and the requirements for data filtering are strongly dependent on the questions being asked of the data **there is no way a rigid implementation or a general filter will likely please everyone.**

So far the question of data filtering has been left to the individual researchers, but this has proven to be somewhat inefficient and it is clear that not everyone possess the necessary mathematical and computational understanding to design and implement filters by themselves, thus we have decided to work on a library of filters specifically designed for the ATLAS system and provide implementations for both Matlab and R. See https://github.com/ATLAS-HUJI/ for example implementations of these filters and other ATLAS related tutorials and code examples.

Basic Filters:

the most basic filter that can be applied to data collected through the ATLAS system is a median filter. Implementations of which are pretty straight forward (see . Currently not all ATLAS systems have access to **co-variance matrices (which are required for implementation of our current Kalman filter implementation - see below),** restricting the number of applicable filters. The median filter is to be preferred over any averaging or least squares implementation, as the error in the ATLAS data is not normally distributed, strongly favoring non-parametric approaches. When implementing median filters it is (as with any filter) important to consider the time interval over which a median is formed and pick a **time interval that is relevant for the animal**. Also important to consider are the behaviors that are to be investigated and the **likely duration of these behaviors**, that is if we are interested in short term behaviors over a few seconds a shorter time window is necessary as opposed to long term behaviors where larger time windows can be used to potentially reduce the error in the localization estimation. Additionally the **sampling rate of the ATLAS system is another important factor**, there is no point in forming an media over 2 or 3 samples. **This means the filter settings will highly depend on the animal, the behavior in question and the data sampling rate.**

Kalman filter:

[please note that the following explanations are very much simplified]

The Kalman filter [1] is the **gold standard in engineering** and produces vastly superior results to any other filtering approach. An implementation of a Kalman filter in position estimation is technically straight forward, provided we know how the system in question operates. Given a position estimate and a prior movement or so called action command we can make predictions about the likely position (given the known movement) and compare this with the current position.

This of course complicates things, as the systems, we as biologists research, are not fully understood and are also not controlled systems. In a implementation for localization of a plane for example we know:

- the position provided by a positioning system
- the thrust and heading direction of the plane.

While the ATLAS system provides us with the former the later is not known to us. Thus inherently any prediction has to be based solely on past movement data Introducing artificial correlations and linearities into the filtered data.

The current implementation is an **adaptation by Paige and Saunders [2] utilizing known co-variance matrices** (thanks to Sivan for finding this one, and also writing the Matlab code) allowing us to circumvent some of these issues, by essentially filtering based on the ATLAS system and not only the unknown system i.e. the animal and provides smoothing and basic interpolation of missing data. Still issues remain and anyone using the filter should be aware of these issues:

- the Kalman filter assumes linearity (and almost nothing in the real world is linear)
- the Kalman filter assumes a Gaussian error distribution (the error in the ATLAS system does not follow a strict Gaussian error distribution)

The current implementation is a first pass, which means we will continue working on the implementation and further improve it in the future. In its current form the filter will smooth and filter whatever data you pass on to it, with the predictive nature of the filter making it even more important to select an appropriate Behavioral time window. If e.g. data for an entire day is passed onto the filter the behavior at the end of the day will have a certain impact on the filtered data at the beginning of the day.

if your interested in the basic implementation this can be found here
https://github.com/ATLAS-HUJI/

Particle Filters:

In the future we will work on an implementation of particle filters, which basically, without going into much detail, are a combination of a Kalman filter approach combined with Monte Carlo simulations. The benefit of particle filters are that they don't assume linearity and make no assumptions about the type of error distribution. In an initial pass we will work on a simple implementation based directly on localizations, but there is potential for a more direct implementation into the system on the level of time of arrival data used by the system to calculate the localizations thus further improving the base performance of localization in the ATLAS system. so the goals here are twofold:

1. provide an alternative approach to a median filter when co-variance matrices are not available
2. provide a more direct and fully integrated approach to further enhance the precision of localization estimation in the ATLAS system.

References:

1. Kalman RE 1960: A new approach to linear filtering and prediction problems, Trans. ASME Ser. D J. Basic Engrg., 82D, pp. 35-45.

Paige CC, Saunders MA 1977: Least squares estimation of discrete linear dynamic systems using orthogonal transformations. SIAM Journal on Numerical Analysis. 14(2):180-93.

Del Moral P 1997: Nonlinear filtering: Interacting particle resolution. Comptes Rendus de l'Académie des Sciences-Series I-Mathematics, 325(6), pp.653-658