# Application of the Extended Kalman Filter for Forecasting Mouse Kinematics and Head Orientation

Joaquín Rapela

June 24, 2025

## Contents



# 1 Introduction

We used a nonlinear state-space model and the Extended Kalman Filter algorithm[1] to infer kinematics and head orientation of simulated (Section 2.1) and real mice (Section 2.2).

[1] https://github.com/joacorapela/lds/blob/master/docs/inference/ekfInference.pdf

# 2 Results

## 2.1 Simulated Data

Figure 1 plots the simulated true mouse positions, and Figure 2 plots the simulated noisy measurements.

[Figure 1 about here.]

[Figure 2 about here.]

### Parameter estimation

Parameter estimation is performed iteratively by maximizing the log-likelihood of the model parameters. Figure 3 plots the obtained log-likelihood as a function of estimation time. The blue and red traces plot the log-likelihood obtained during estimation of the kinematics and head-orientation parameters, respectively. The estimation of the kinematics parameters is much faster since it uses the standard Kalman filter, while the estimation of the head-orientation parameters uses the slower extended Kalman filter. The black horizontal line plots the log-likelihood of the true model parameters. That the log-likelihood of the true model parameter is substantially larger that obtained by the estimated parameters indicates a problem in the estimation method that we will address next (Section 3).

[Figure 3 about here.]

Figure 4 plots the true, initial and estimated model parameters. Figures 5 and 6 plot the initial state mean and standard deviation parameters, respectively.

Except from `omega_Q_std` and most initial state parameters, estimated parameters accurately approximate their true values.

[Figure 4 about here.]

[Figure 5 about here.]

[Figure 6 about here.]

### Forecasting

Figure 7 and 8 plot the true, measured and forecasted positions and head orientations of the simulated mouse. Forecasting was performed with an horizon of 2 samples. Figures 9-12 show forecastings with horizons of 20 and 200 samples.

Forecastings of positions are good for all horizons used. Forecastings of head orientation are good for horizons of 2 and 20 samples, but break down for 200 samples. The 95% confidence intervals appear to be well calibrated, as the true positions and head orientation angles are covered by the confidence interval 95% of the time.

[Figure 7 about here.]

[Figure 8 about here.]

[Figure 9 about here.]

[Figure 10 about here.]

[Figure 11 about here.]

[Figure 12 about here.]

## 2.2 Real Data

Figure 13 plots the real mouse positions and head orientations that we forecast below.

[Figure 13 about here.]

### Parameter estimation

Figure 14 plots the obtained log-likelihood as a function of estimation time. The blue and red traces plot the log-likelihood obtained during estimation of the kinematics and head-orientation parameters, respectively.

[Figure 14 about here.]

Figure 15 plots the initial and estimated model parameters. Figures 16 and 17 plot the initial state mean and standard deviation parameters, respectively.

Figure 15 shows that the estimated model parameters are substantially different from the inital parameters, and Figure 14 indicates that the change in these parameters increased the log-likelihood of the parameters. Thus, parameter estimation was beneficial. Note, however, that learning did not change much the initial values of the mean and standard deviation of the initial state (Figures 16 and 17).

[Figure 15 about here.]

[Figure 16 about here.]

[Figure 17 about here.]

### Forecasting

Figure 18 and 19 plot the measured and forecasted positions and head orientations of a real mouse. Forecasting was performed with an horizon of 2 samples. Figures 20-23 show forecastings with horizons of 10 and 50 samples.

Forecastings of head orientation are reasonable for all horizons. Forecastings of position are good for horizons of 2 and 10 samples, but they break down for the horizon of 50 samples.

[Figure 18 about here.]

[Figure 19 about here.]

[Figure 20 about here.]

[Figure 21 about here.]

[Figure 22 about here.]

[Figure 23 about here.]

# 3 TODO list

1. Fix the problem in the estimation of parameters of the extended Kalman filter model. As shown in Figure 3, the method to estimate parameters of the extended Kalman filter model cannot achieve the log likelihood obtained with the true model parameters.

2. Use ONIX recordings to check the accuracy of the kinematics model.

3. Forecast data from other experimental periods.

4. Evaluate forecasting performance on less noisy real data.

5. Compare forecasting performance of the extended Kalman filter model with that from other forecasters (e.g., RNN, XFADS and simpler forecasters).

6. Evaluate the forecaster with data from a robot with a camera exploring the arena with the 360 degree screen.

7. Integrate forecaster into Bonsai.ML.

# 4 Methods

## 4.1 Simulated data

To perform the simulations we used the script doSimulateNDSwithGaussianNoise.py.

Learning was performed by maximizing the model parameters likelihood by gradient ascent using PyTorch. This maximization was performed in two steps. First we maximized the likelihood of the kinematic parameters, and then we fixed the kinematics parameters and maximised the likelihood with respect to the head orientation parameters. To estimate the kinematic parameters we used the script doEstimateTorchKinematics.py, and to estimate the head orientation parameters we used the script doEstimateTorchKinematicsHO.py.

After learning the model parameters, we performed inference on the model latents using the script doEKFilter.py

After inferring the model latents, we performed forecasting using the script doEKForecasting.py.

Figures 1 and 2 were generated with the script doPlotSimulation.py, Figure 3 was generated with the script doPlotEstimationLogLikelihood.py, Figures 4-6 were generated with the script doPlotTrueInitialEstimatedParams.py, and Figures 7-12 were generated with the script doPlotForecasting.py.

## 4.2 Real data

We limited the analysis to a section lasting 194 seconds, starting at time 1450 seconds, from file `M24086_20250203_0_tracking_2025-02-06T10_39_59.csv` (Figure 13).

As for the simulate data, learning was done by maximizing the likelihood of the model parameters in two steps. First the likelihood of the kinematic parameters was maximized using the script doEstimateTorchKinematics.py, and then the kinematic parameters were fixed and the head orientation parameters were optimized using the script doEstimateTorchKinematicsHO.py.

Filtering was done with the script doEKFkinematicsHO.py and forecasting with the script doEKForecasting.py.

Figure 13 was generated using the script doPlotData.py, Figure 14 was generated with the script doPlotEstimationLogLikelihood.py, Figures 15-17 were generated with the script doPlotInitialEstimatedParams.py, and Figures 18-23 were generated using the script doPlotForecasting.py.

# List of Figures

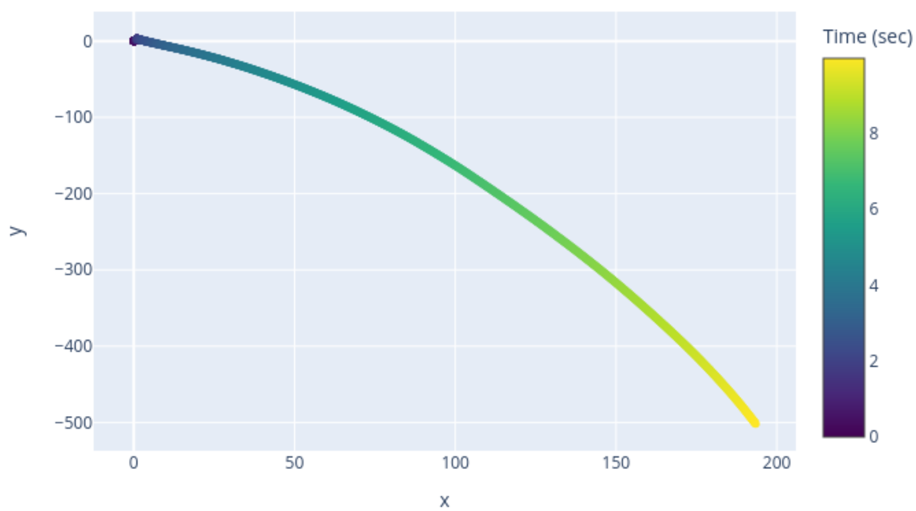


Figure 1: True simulated mouse positions and head orientations. Click on the image to get its interactive version, and zoom in the plot to view the head orientations (blue arrow).

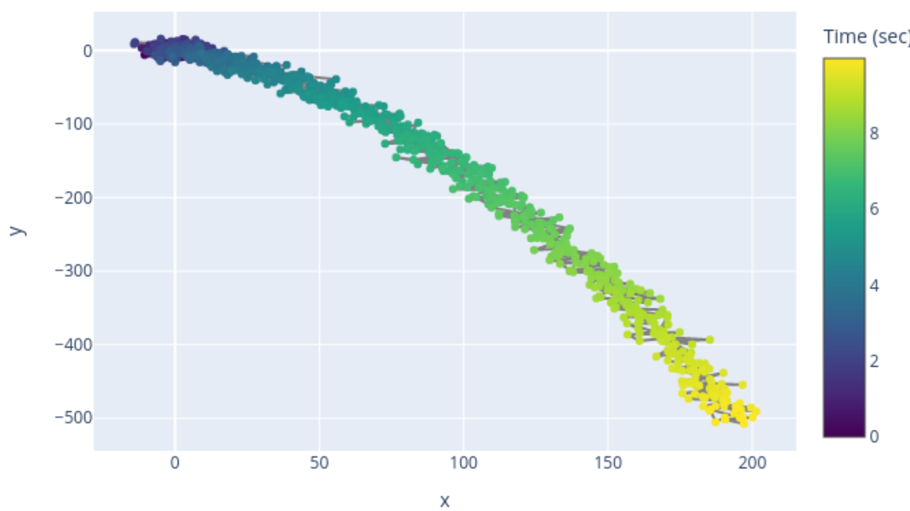


Figure 2: Simulated noisy mouse positions and head orientations. Click on the image to get its interactive version, and zoom in the plot to view the head orientations (blue arrow).

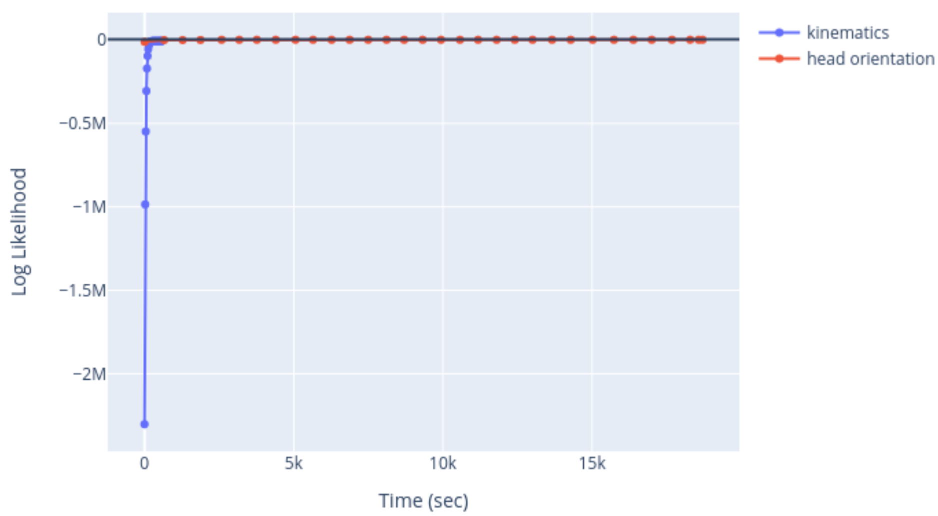


Figure 3: Log-likelihood of estimated parameters as a function of estimation time. The blue and red traces plot the log-likelihood obtained during the estimation of the kinematics and head-orientation parameters, respectively. The black horizontal line plots the log-likelihood of the true model parameters. Click on the image to get its interactive version.

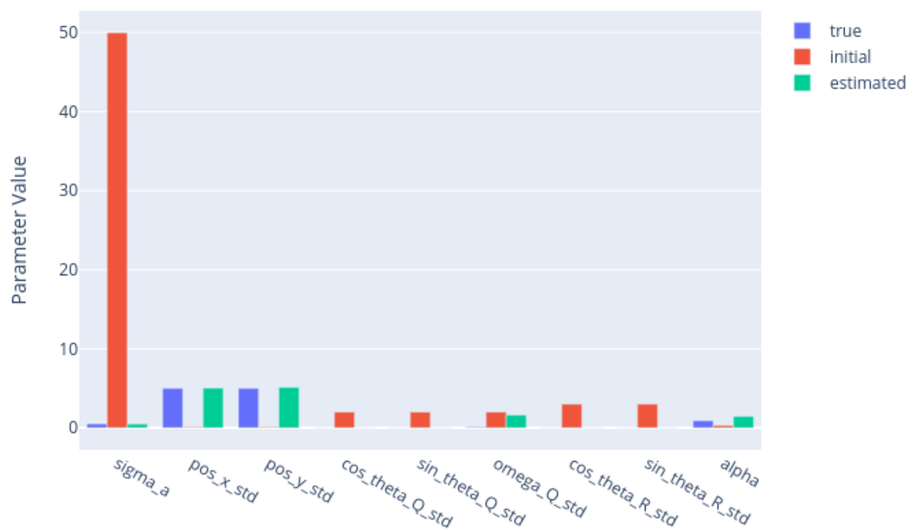


Figure 4: True, initial and estimated model parameters. True and estimated model parameters are not displayed for `cos_theta_Q_std`, `sin_theta_Q_std`, `cos_theta_R_std` and `sin_theta_R_std` are not displayed because they all have small absolute value.

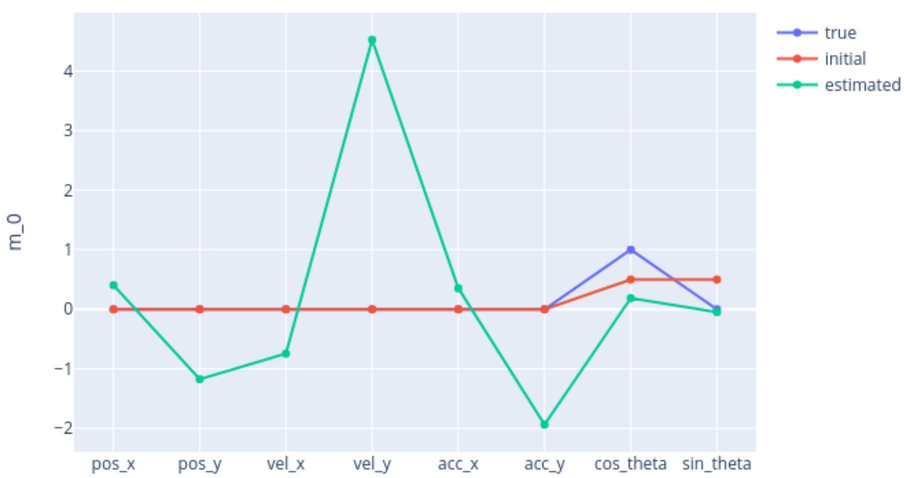


Figure 5: True, initial and estimated inital state mean parameters.

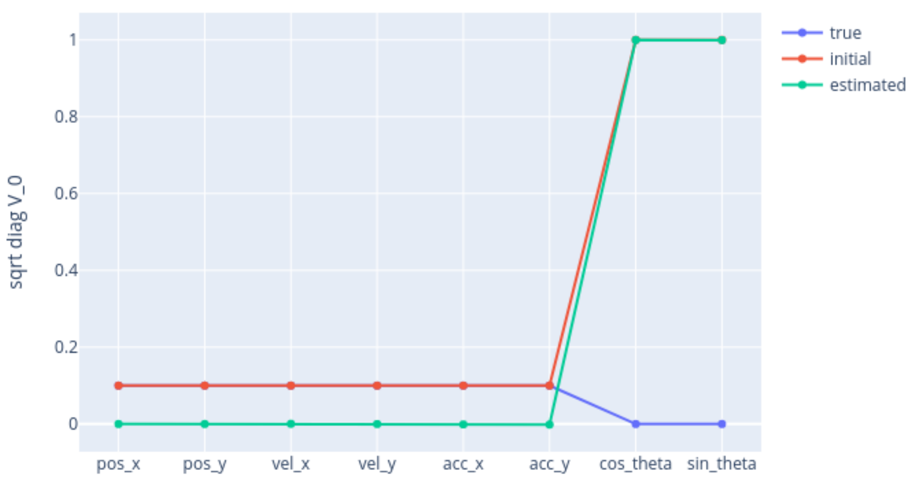


Figure 6: True, initial and estimated initial state standard deviation parameters.

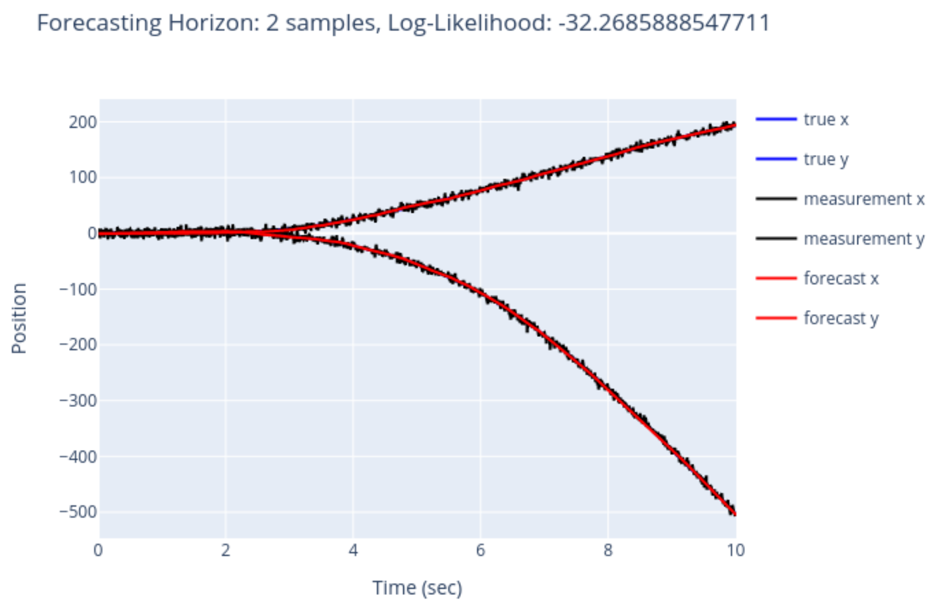


Figure 7: True, measured and forecasted (horizon h=2 samples) horizontal (x) and vertical (y) position of the simulated mouse. Click on the image to get its interactive version.

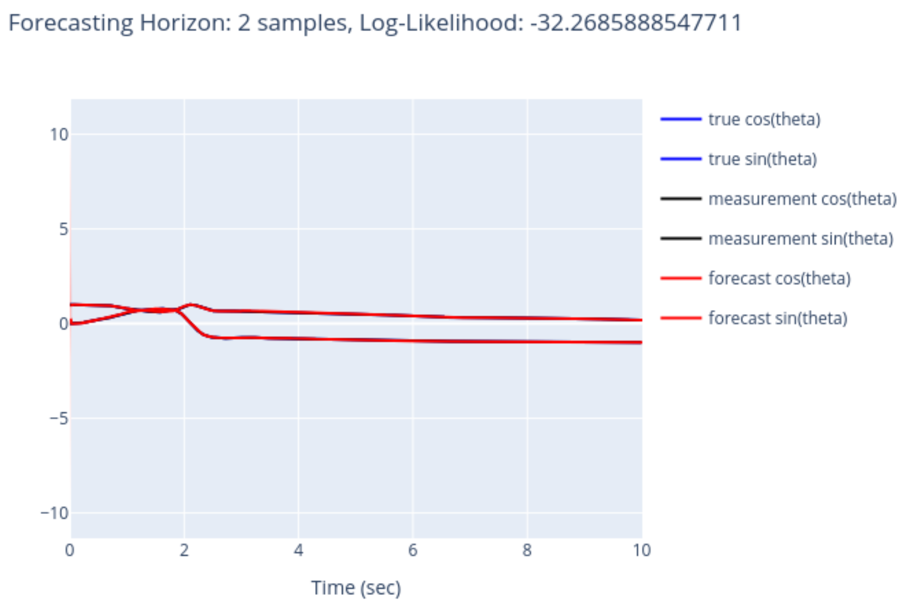


Figure 8: True, measured and forecasted (horizon h=2 samples) sine and cosine of the head-orientation angle of the simulated mouse. Click on the image to get its interactive version.

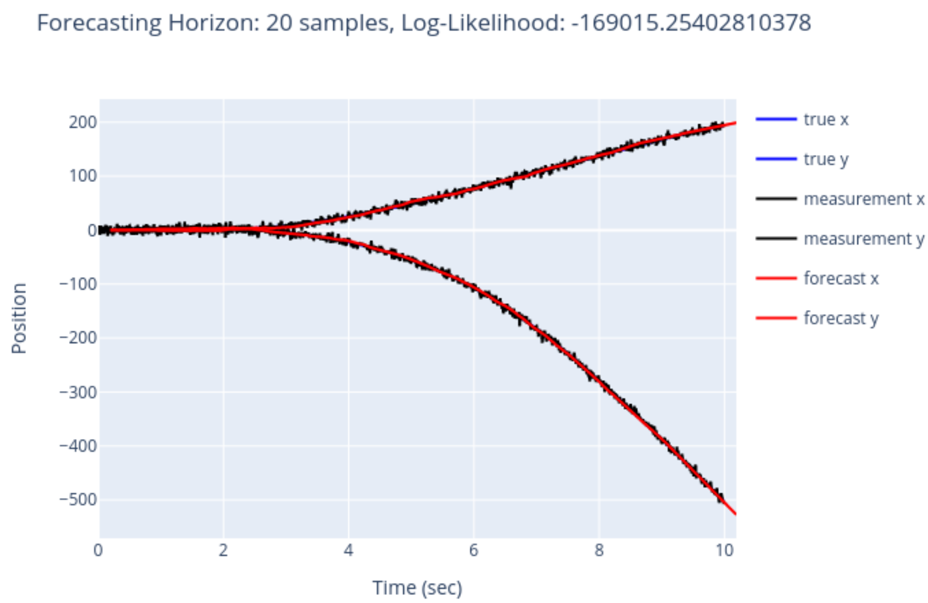


Figure 9: True, measured and forecasted (horizon h=20 samples) horizontal (x) and vertical (y) position of the simulated mouse. Click on the image to get its interactive version.

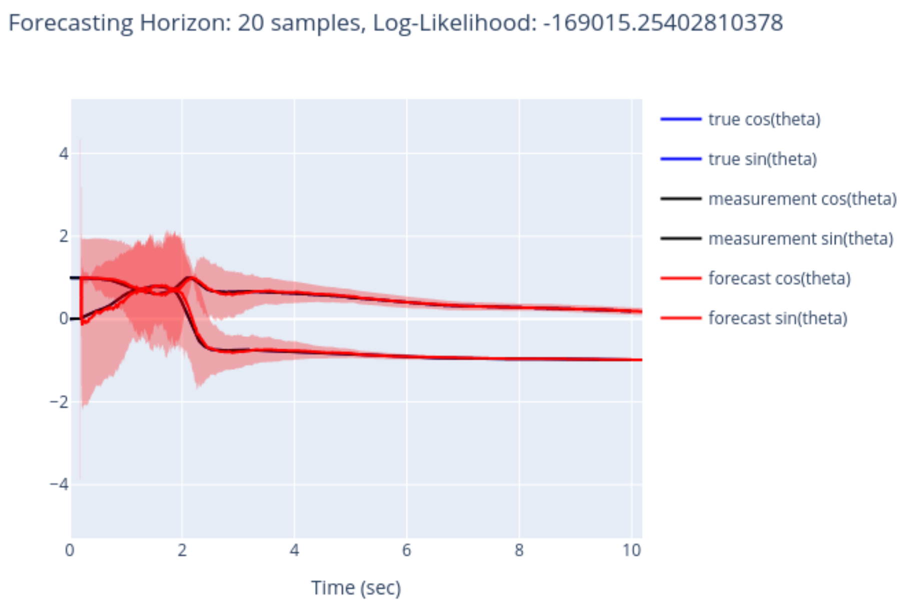


Figure 10: True, measured and forecasted (horizon h=20 samples) sine and cosine of the head-orientation angle of the simulated mouse. Click on the image to get its interactive version.

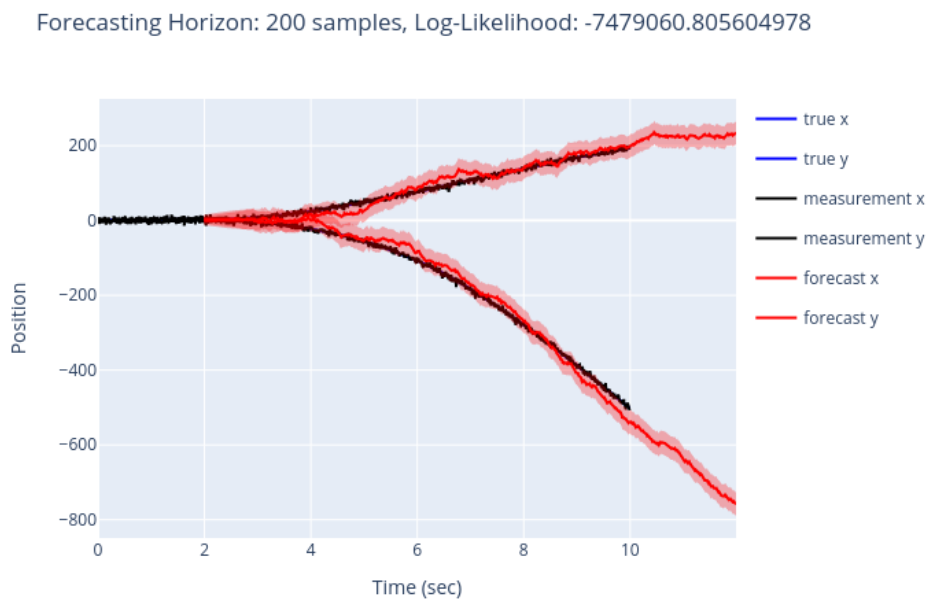


Figure 11: True, measured and forecasted (horizon h=200 samples) horizontal (x) and vertical (y) position of the simulated mouse. Click on the image to get its interactive version.

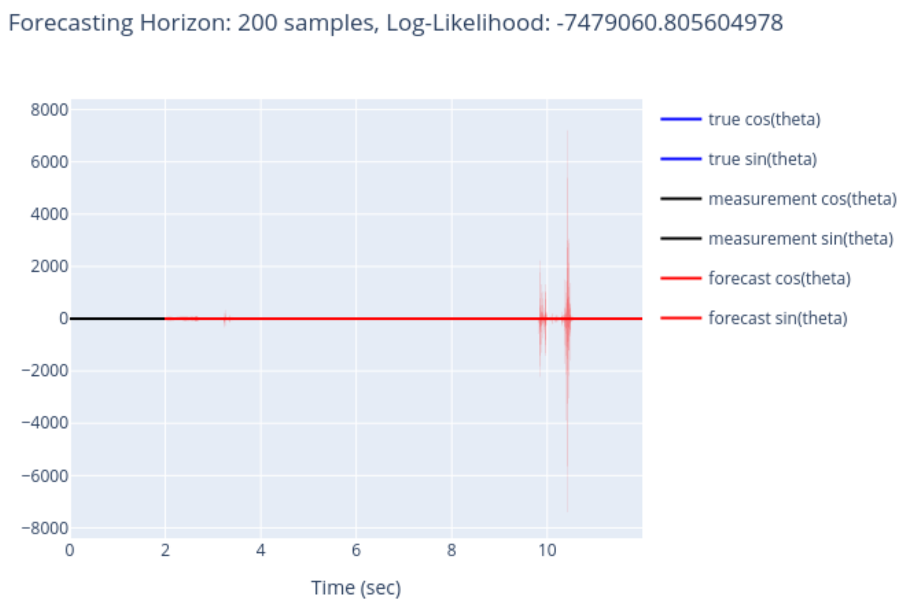


Figure 12: True, measured and forecasted (horizon h=200 samples) sine and cosine of the head-orientation angle of the simulated mouse. Click on the image to get its interactive version.

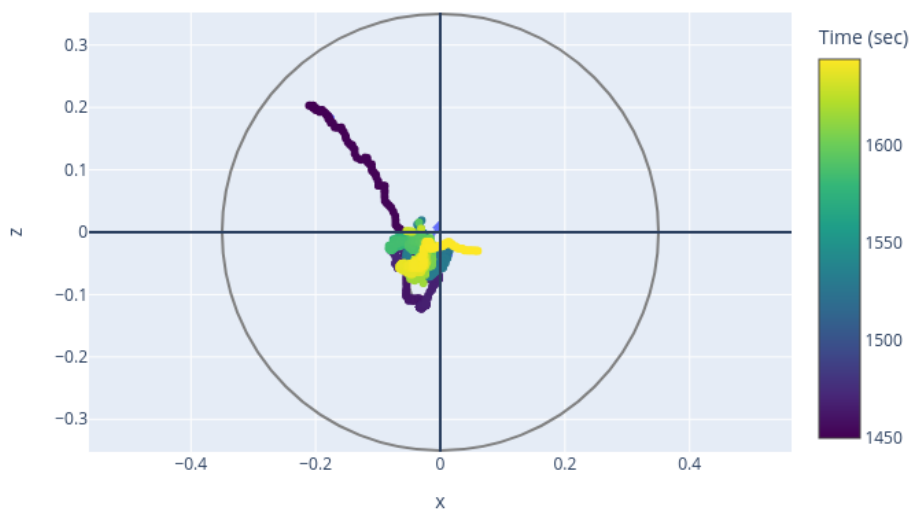


Figure 13: Positions and head orientation of the real mouse used for forecasting below.

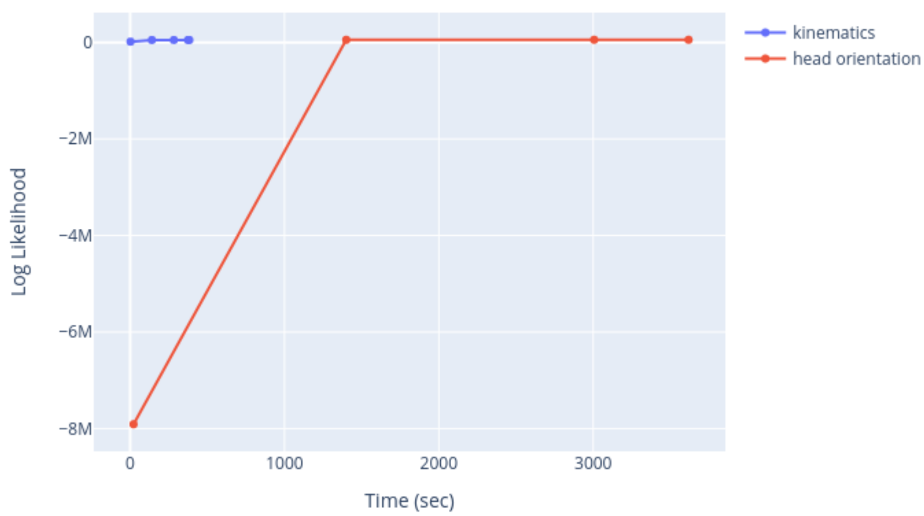


Figure 14: Log-likelihood of estimated parameters as a function of estimation time. The blue and red traces plot the log-likelihood obtained during the estimation of the kinematics and head-orientation parameters, respectively.

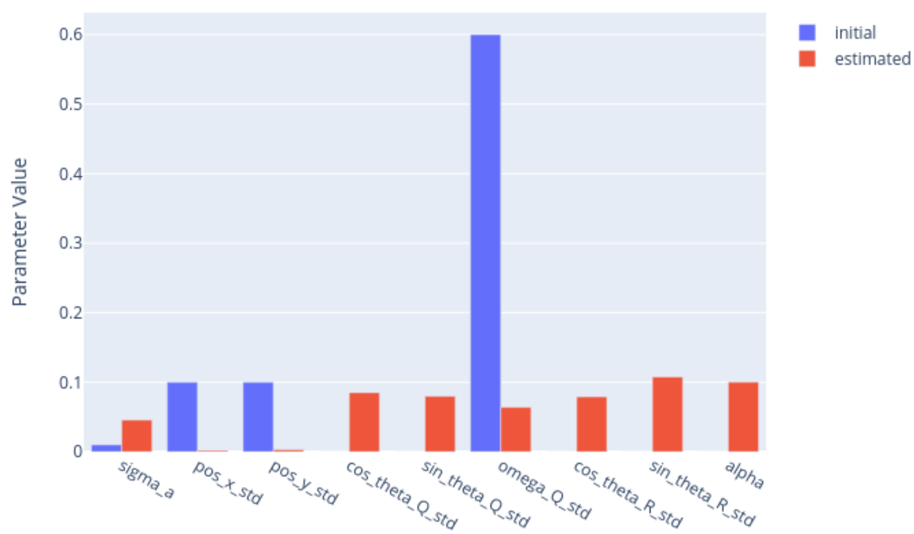


Figure 15: Initial and estimated model parameters.

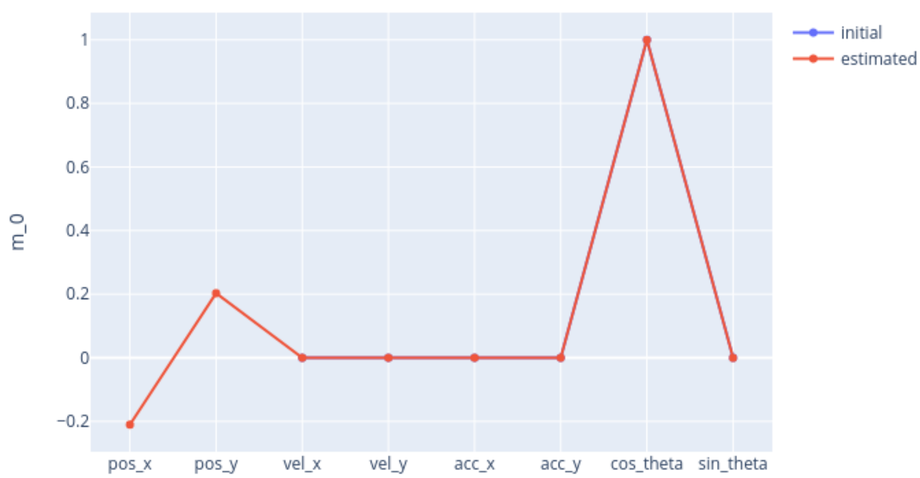


Figure 16: Initial and estimated inital state mean parameters.

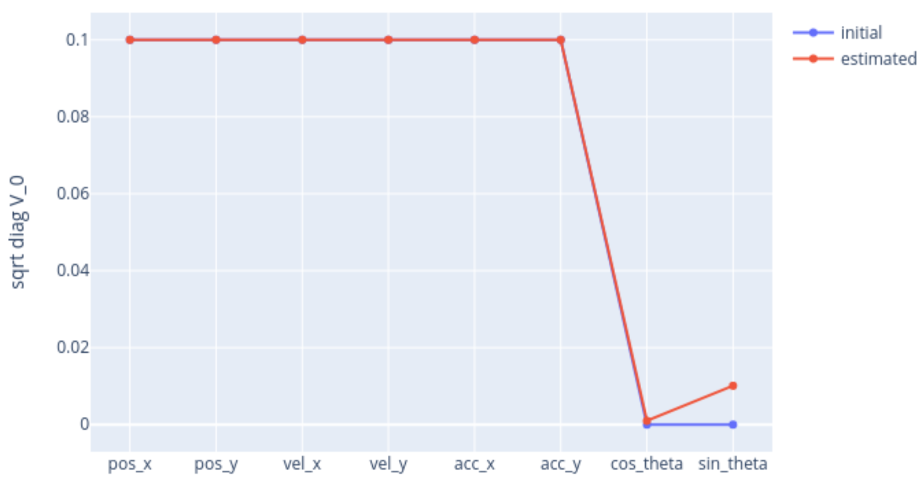


Figure 17: Initial and estimated initial state standard deviation parameters.

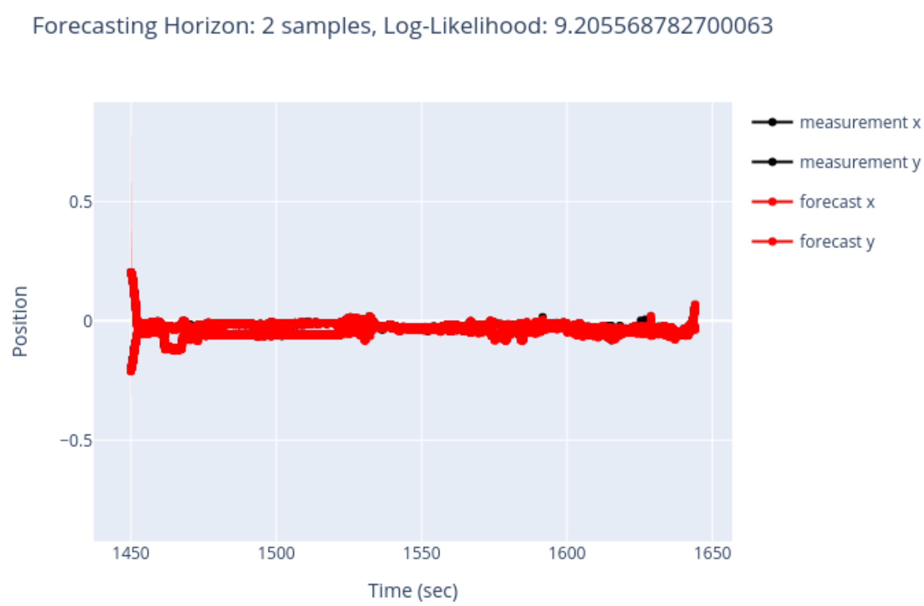


Figure 18: Measured and forecasted (horizon h=2 samples) horizontal (x) and vertical (y) position of a real mouse. Click on the image to get its interactive version.

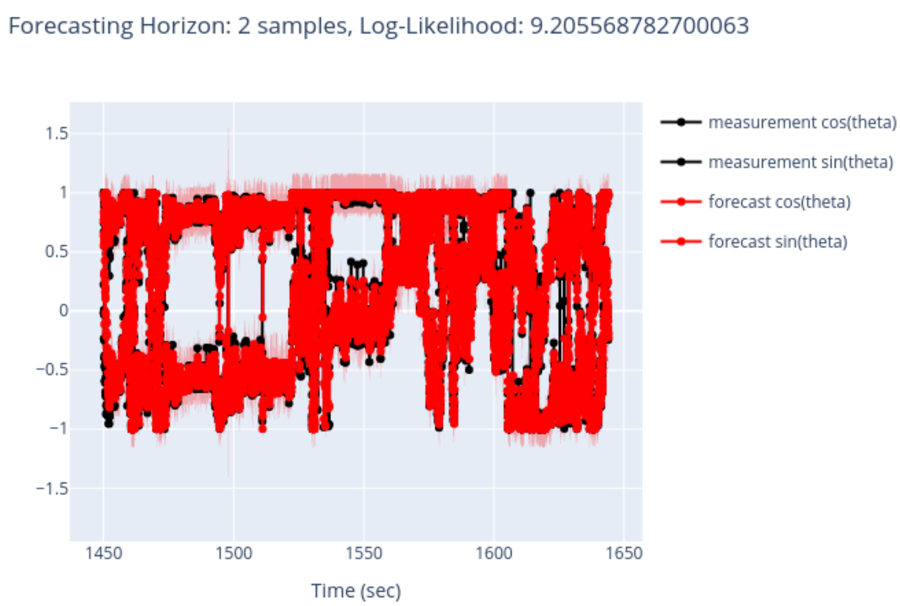


Figure 19: Measured and forecasted (horizon h=2 samples) sine and cosine of the head-orientation angle of a real mouse. Click on the image to get its interactive version.

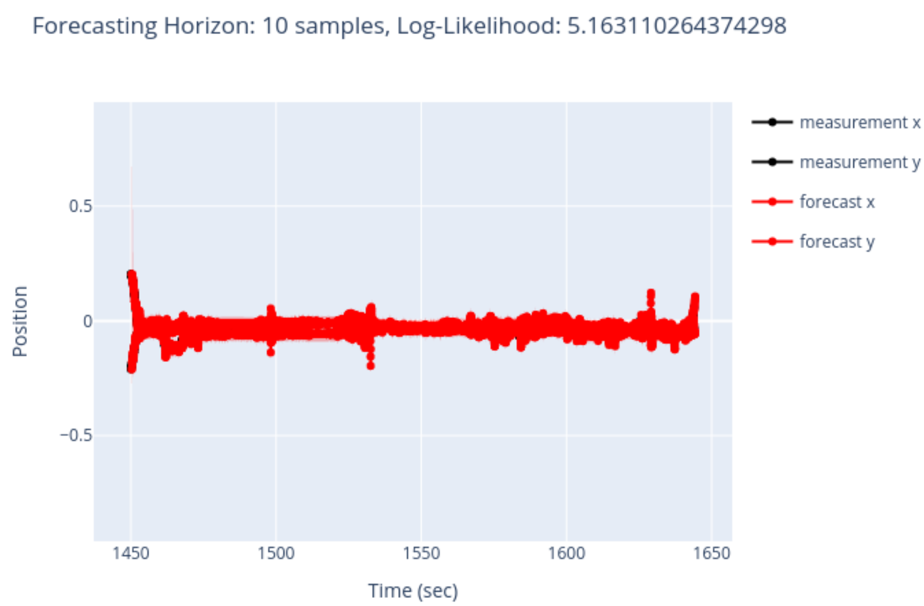


Figure 20: Measured and forecasted (horizon h=10 samples) horizontal (x) and vertical (y) position of a real mouse. Click on the image to get its interactive version.

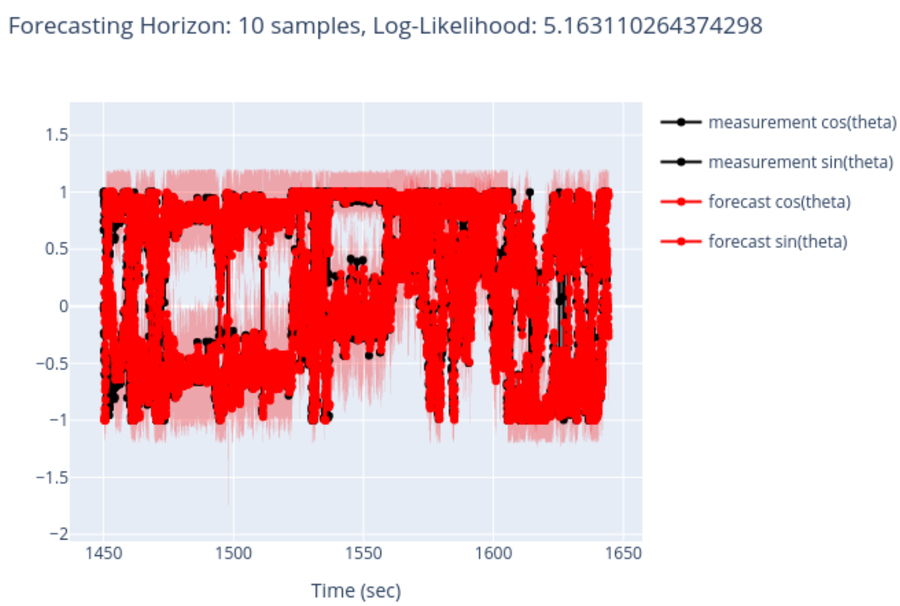


Figure 21: Measured and forecasted (horizon h=10 samples) sine and cosine of the head-orientation angle of a real mouse. Click on the image to get its interactive version.

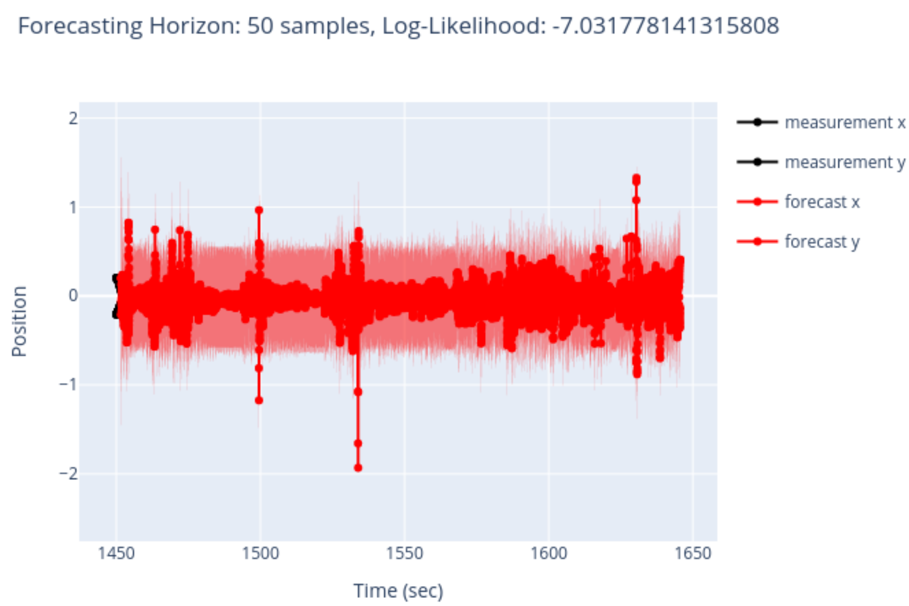


Figure 22: Measured and forecasted (horizon h=50 samples) horizontal (x) and vertical (y) position of a real mouse. Click on the image to get its interactive version.

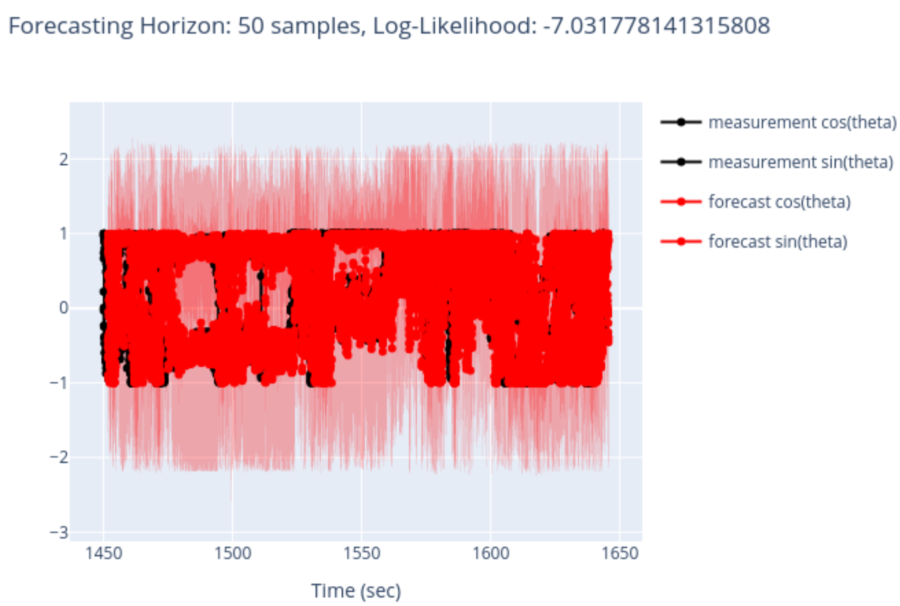


Figure 23: Measured and forecasted (horizon h=50 samples) sine and cosine of the head-orientation angle of a real mouse. Click on the image to get its interactive version.