Kalman Filtering in Biosensing

Vincent Eurasto

November 15, 2022

Contents

1 Introduction

Biosensing is the use of sensors to measure a *biological process*. The measurement device or a *biosensor*, usually measures biological markers, *biomarkers*, in a system and converts the response into an electrical signal. For example, in order to measure the glucose level of the blood, we need to somehow measure the sugar level of the blood and convert it into an electric signal for further processing. In this example, we have the following classification

1. *Biological process*: varying glucose level of the blood
2. *Biosensor*: glucose sensor
3. *Biomarker*: glucose

Every biological process and measurement device contains some noise that corrupts the signal. Fortunately, different signal processing techniques can help us to reduce the noise. That is the reason why signal processing is usually an important step in biosensing. There are several methods for filtering noise from a signal but one of the most common noise-removing filters is called the Kalman filter.

This literature review gives an introduction and a review of the Kalman filter and its applications in biosensing. Additionally, simple example usage of the Kalman filter in glucose level monitoring is developed and the results of this experiment are provided in the end.

2 Kalman filter

Even though the Kalman filter (KF) was developed in the 1960s, it's still one of the most important and common data fusion algorithms in use today. The KF is typically used in smoothing noise data and providing estimates of parameters of interest. [1] The applications of the Kalman filter include object tracking, economics, navigation, and medical imaging. However, in this chapter, we address some applications of the KF in biosensing. Furthermore, an introduction and the mathematical foundation of the Kalman filter algorithm are provided.

2.1 Kalman filter in biosensing

Kalman filter can be used to remove noise and estimate the parameters of interest from a biological system. For example, in a continuous glucose monitoring system, we would like to know the blood sugar level accurately. However, the glucose sensor might have some noise in it, and the measured data could get corrupted with noise. In order to extract the original signal from the noisy data, we could use the Kalman filter.

The KF can also be used in, for example, to smooth ECG measurement data, in applications where optical sensors are used, and in many other applications.

2.2 The Kalman filter algorithm

The Kalman filter requires us to know the behavior of the system of interest and how it evolves over time. In the mathematical sense, the Kalman filter assumes that the state of a system evolves according to the equation

$$\mathbf{x}_t = \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{B}_t \mathbf{u}_t + \mathbf{w}_t, \tag{1}$$

where

- $\mathbf{x}_t$ is the state vector containing the parameters of interest at time t
- $\mathbf{F}_t$ is the state transition matrix, which applies the transformation from the previous state to the next
- $\mathbf{B}_t$ is the control input matrix, which applies the transformation to the external input vector $\mathbf{u}_t$ to the system
- $\mathbf{w}_t$ is the process noise vector

Additionally, the Kalman filter requires measurements from the system to be performed according to the model

$$\mathbf{z}_t = \mathbf{H}_t \mathbf{x}_t + \mathbf{v}_t, \tag{2}$$

where

- $\mathbf{z}_t$ is the vector of measurements
- $\mathbf{H}_t$ is the state transition matrix that maps the state vector (parameters of interest) into the measurement domain
- $\mathbf{v}_t$ is the noise vector from the measurement device

Now that we have the needed space-state model (1) and the measurement model (2) we can form the Kalman filter algorithm. [1]

Algorithm 1 Kalman filter

Require: Suitable initial mean $\hat{\mathbf{x}}_{0|0} = \mathbf{m}_0$ and covariance matrix $\mathbf{P}_{0|0} = \mathbf{P}_0$

for $t = 1, 2, ...$ **do**

Prediction step:

$$\begin{aligned} \hat{\mathbf{x}}_{t|t-1} &= \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} + \mathbf{B}_t \mathbf{u}_t \\ \mathbf{P}_{t|t-1} &= \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^T + \mathbf{Q}_t \end{aligned}$$

Measurement update:

$$\begin{aligned} \mathbf{K}_t &= \mathbf{P}_{t|t-1} \mathbf{H}_t^T \left(\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^T + \mathbf{R}_t \right)^{-1} \\ \hat{\mathbf{x}}_{t|t} &= \hat{\mathbf{x}}_{t|t-1} \mathbf{K}_t \left(\mathbf{z}_t - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1} \right) \\ \mathbf{P}_{t|t} &= \mathbf{P}_{t|t-1} - \mathbf{K}_t \mathbf{H}_t \mathbf{P}_{t|t-1} \end{aligned}$$

end for

3 Experiment

In this chapter, a small example scenario of blood sugar level monitoring is developed and presented. Additionally, a Kalman filter algorithm is implemented and bench-marked against using only the raw measurements to give the reader an intuition on what the Kalman filter can do. First, we will present the models used in the simulations. After this, the Kalman filter algorithm is implemented in Python and the results are presented in the end.

3.1 Blood glucose level and monitoring

Continuous glucose monitoring is important for a diabetic's well-being. Without monitoring, a diabetic patient can become ill because the patient's glucose levels are not where they are supposed to be. [2]

Blood sugar levels can be monitored using a continuous glucose monitor (CGM). A CGM uses a sensor inserted under the skin to measure blood sugar levels continuously, usually every few minutes.

The blood sugar levels vary during the day. For example, blood sugar levels rise when we eat and decrease when we don't eat. Normal blood sugar level lays between two typical targets: [3]

- Before a meal: 80 to 130 mg/dL,
- Two hours after a meal: lower than 180 mg/dL.

Diabetics need to constantly take insulin in order to keep their blood glucose levels normal.

3.2 Experiment models

The models used in this experiment are not accurate and should only be used for educational purposes. It is a difficult engineering task to come up with an accurate model of how blood glucose levels evolve over time. This is why the models used in this experiment should only give the reader an intuition on how the Kalman filter is implemented and bench-marked in practice.

First of all, we assume that there is no input in our system, i.e., $\mathbf{u}_t = \vec{0}$, and thus $\mathbf{B}$ matrix is not necessary. The state vector $\mathbf{x}_t$ contains two different parameters and their derivatives. The first two elements of the state vector are the glucose level and its derivative. The last two elements are the short-term effect of eating and its derivative. The initial state for the blood glucose level is set to 100 mg/dL. The actual measured glucose level is the sum of the blood glucose level and the short-term effect of eating. Thus, the models used in this experiment are

$$\mathbf{x}_t = \begin{bmatrix} 100 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{F}_t = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 \\ 0 & c & 0 & 0 \end{bmatrix}, \quad \mathbf{H}_t = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}},$$

where $\Delta t = 1$ is the time step size (one minute), $c = -\frac{1}{2}$ is the time constant that takes the short-term effect of eating to equilibrium. Additionally, we are using the following co-variance matrices for the process and measurement noise

$$\mathbf{Q}_t = \begin{bmatrix} q_t & 0 & 0 & 0 \\ 0 & q_t & 0 & 0 \\ 0 & 0 & q_t & 0 \\ 0 & 0 & 0 & q_t \end{bmatrix}, \quad \mathbf{R}_t = \begin{bmatrix} r_t \end{bmatrix},$$

where $q_t = 0.01$ is the variance of each process terms and $r_t = 0.5$ is the variance in the measurement device.

3.3 Results

Illustration of the imaginary scenario where the blood sugar level is monitored for 300 minutes is depicted in figure 1. The Kalman filter can clearly remove

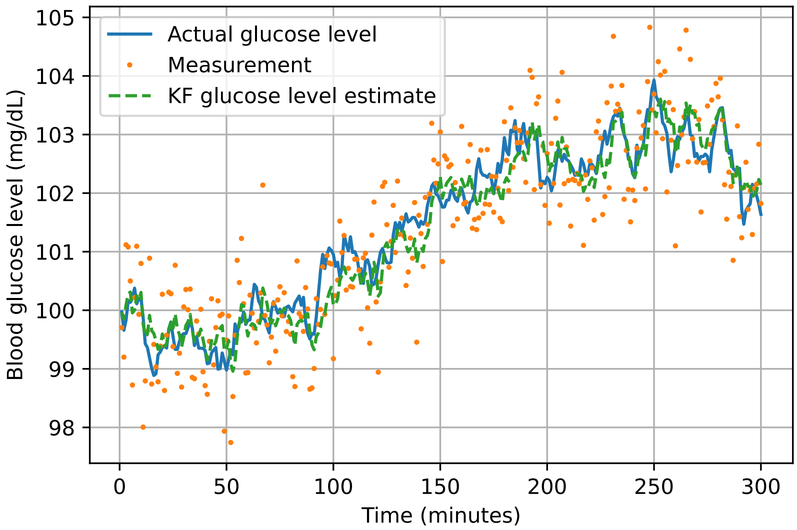

Figure 1: Illustration of the imaginary scenario where the blood sugar level is monitored for 300 minutes.

some noise compared to using only the raw measurements alone for the blood sugar level estimation.

The performance of the KF can be assessed with root mean square error (RMSE)

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i}^{N}(y_i - \hat{y}_i)^2}, \tag{3}$$

where N is the number of sample data points, y_i is the actual value of the data and $\hat{y}_i$ is the estimated value of the data. The results of this experiment are presented in table 1.

Method	RMSE
Raw measurement	0.7319
Kalman filter	0.3667

Table 1: Root mean square errors can be used as an assessment of the two different methods to predict the actual blood glucose level. By using the models and covariance matrices presented in the section 3.2 clearly shows the advantages of the Kalman filter. The RMSE for the raw measurements is almost 100% larger than what it is for the Kalman filter.

4 Conclusions and discussion

In this literature survey, an introduction of the Kalman filter was described, and a noisy experiment model for the evolving blood glucose level was developed and simulated. Additionally, the Kalman filter algorithm was programmed with Python, and it was used to reduce the noise in an imaginary example scenario. By investigating the results, we can conclude that the Kalman filter improved the overall accuracy of the data. However, as stated, the model in this experiment is not accurate and requires more engineering effort to make it usable in actual real-life scenarios.

References

[1] R. Faragher, "Understanding the basis of the kalman filter via a simple and intuitive derivation [lecture notes]," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 128–132, 2012.

[2] Bio-Resource, "How does glucometer or glucose monitoring device work?." Online, 2015. https://technologyinscience.blogspot.com/2015/01/how-does-glucometer-or-glucose.html.

[3] Centers for Disease Control and Prevention, "Manage blood sugar." Online, 2021. https://www.cdc.gov/diabetes/managing/manage-blood-sugar.html.