Formulation

I am not a dietician but I'm going to attempt to model my own weight loss journey with a simple extended kalman filter.

States

Here are the following states that I want to estimate

- x_1 - "True Weight" (weight with daily fluctuations smoothed)
- x_2 - Daily "True Weight" loss or gain
- x_3 - Caloric conversion (lbs per thousand calories)
- x_4 - Non-active daily calories (BMR + TEF + NEAT)

Inputs

- u_1 - Calories ingested
- u_2 - Calories associated with active exercise EAT

Measurements

- z_1 - Weight as reported by my crappy scale from amazon

Noise Parameters

This is not a tutorial on kalman filtering , and I'm mainly doing this for myself. So here's what I've chosen my noise parameters to look like.

Process noise

$$Q = \begin{bmatrix} 1.0 & 0 & 0 & 0 \\ 0 & (0.1)^2 & 0 & 0 \\ 0 & 0 & (0.1)^2 & 0 \\ 0 & 0 & 0 & 10^2 \end{bmatrix}$$

Input Noise

$$M = \begin{bmatrix} 150^2 & 0 \\ 0 & 50^2 \end{bmatrix}$$

Measurement Noise

$$R = [(0.1)^2]$$

State Transition Model

$$f(x) = \begin{bmatrix} x_1 + x_2 \\ \frac{x_3}{1000}(u_1 - u_2 - x_4) \\ x_3 \\ x_4 \end{bmatrix}$$

State Transition Jacobian

$$\frac{\partial f}{\partial x} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{1000}(u_1 - u_2 - x_4) & -fracx_31000 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

State Input Jacobian

$$\frac{\partial f}{\partial u} = \begin{bmatrix} 0 & 0 \\ \frac{x_3}{1000} & -\frac{x_3}{1000} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Prediction

$$x^+ = f(x)$$

Note the special prediction that includes the input noise

$$P^+ = \frac{\partial f}{\partial x} P \frac{\partial f}{\partial x}^\top + \frac{\partial f}{\partial u} M \frac{\partial f}{\partial u}^\top + Q$$

Measurement Model

$$h(x) = [x_1]$$

We will use the typical KF kalman update since this model is linear.

Initialization

I think the Initialization here is fairly straight-forward. I'm going to use 180lbs (my current weight) as x_1, 0 as x_2. $1/3.5$ as x_3 given the heuristic that 3500 calories is a lb of fat. 1900 as x_4 given some sloppy calculations I did with data from my apple watch.