**Reanalysis of a football kick using the Kalman Filter**

As in the video, we will use the model of the inclined throw (with a simple friction term) and images of a camera (or: a video with 10 frames per second) to obtain a reanalysis of the ball's trajectory.

For convenience we perform a number of simplifications. First, we assume that the state vector has only two elements, namely the ball's coordinates in $x$ (horizontal) and $z$ directions (vertical), hence $\mathbf{x} = \{x, z\}$. We neglect friction, so the velocity $u$ (in $x$-direction) is constant and $w$ (in $z$-direction) depends only on gravity. Here are the equations of the model

$$x_t = x_0 + u_0\,\Delta t$$

$$z_t = z_0 + w_0\,\Delta t - 0.5\,g\,\Delta t^2$$

The time step $\Delta t$ is taken to be the interval between two video frames (0.1 s). Note that the starting conditions are such that the football (as in the video) is kicked from right to left, hence, $u_0$ is negative and the starting position $x_0$ is at 23.4 m.

When assimilating observations, $x_t$ and $z_t$ become the "background" $x_b$ and $z_b$, which is adjusted based on observations. The "analysis" $x_a$ and $z_a$ will then be used in the next time as step $x_0$ and $z_0$ for the model forecast. Since they deviate slightly from $x_t$ and $z_t$, also the velocities must be adjusted. In other words, the new $u_0$ and $w_0$ must be recalculated from the previous time step:

$$u_{0\_new} = (x_t - x_{t-1}) / \Delta t$$

$$w_{0\_new} = (z_t - z_{t-1}) / \Delta t - 0.5\,g\,\Delta t$$

As to the observations, we assume that all photos are taken from the same camera. For simplicity we assume that the projection plane of all photos aligns with the plane of the trajectory. However, the camera uses a pixel coordinate system ($i, j$) with origin in top left corner (positive to the right for $i$ and downward for $j$). The $H$ operators for the two pixel dimensions ($i, j$) are:

$$H(x) = a\,x$$

$$H(z) = k - a\,z$$

with $a$ = 44.2 pixels $m^{-1}$ and $k$ = 500 pixels

We further assume that the errors (standard deviations) of model and of the camera are 0.1 m and 3 pixels, respectively, and we assume that there is no correlation between $x$ and $z$ coordinates of the ball in $\mathbf{B}$ or between the pixel errors in $i$ and $j$ direction in $\mathbf{R}$. Note that this a massive oversimplification. In fact, data assimilation in essence draws from correlations of variables within the state vector, but this is less trivial to calculate. As a consequence of our assumptions, $\mathbf{B}$ and $\mathbf{R}$ are both diagonal matrices with time-invariant diagonal vectors $\mathbf{b}$ = {0.01 $m^2$, 0.01 $m^2$} and $\mathbf{r}$ = {9 $\text{pixels}^2$, 9 $\text{pixels}^2$}. Note that $\mathbf{B}$ is not updated here.

Observations of the camera for each frame are given in the Excel Sheet. Before turning to the sheet, typing the model and solving the assimilation, we should have a look at the algebra.

Please simplify the solution:

$$\mathbf{x} = \mathbf{x}_b + \mathbf{BH}^T(\mathbf{R}+\mathbf{HBH}^T)^{-1}\,(\mathbf{y}-H[\mathbf{x}_b])$$

for the assumptions made in this exercise, so that it can easily be implemented in Excel.

Then use the Excel file to solve the assimilation.