Lecture Automotive Vision 2019

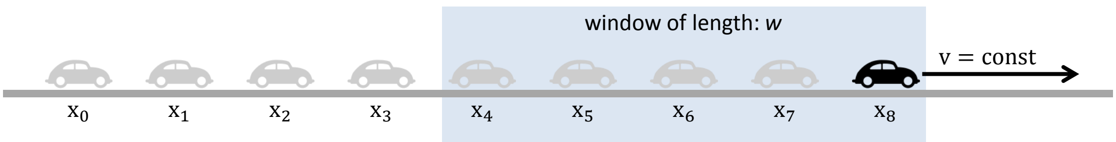
Institut für Mess- und Regelungstechnik, Karlsruher Institut für Technologie
Dr. Martin Lauer, M.Sc. Annika Meyer, M.Sc. Danial Kamran

The objective of this exercise sheet is the implementation of a *Regression Solver* and a *Kalman Filter* for a simplified simulated scenario to estimate the position of a vehicle. The input data for some tasks is provided in a Matlab data file located in *data.mat*. When the data file is loaded into the workspace[1], the following variables are defined in the workspace:

`initial_state`	The initial state x_0 of the vehicle as $[x, y]$.
`numT`	The number of time steps.
`delta_motion`	The delta movement of the vehicle for one discrete time step. The motions are stored as `numT x 2` matrix with each row containing $[\Delta x, \Delta y]$. Note: the first row describes the delta motion from the initial state to x_1, the second row from x_1 to x_2, etc.
`measurements`	The two-dimensional global position measurements for the vehicle stored as `numT x 2` matrix with each row holding $[x, y]$. Note: the measurements are taken at a frequency of 1 s, i.e. the first measurement is taken at point in time 1 s, the second emasurement at point in time 2 s, etc.
`true_poses`	The ground truth vehicle positions, i.e. the correct positions of the vehicle which are used for evaluation purpose. The ground truth data is stored as `numT x 2` matrix with each row describing the position as $[x, y]$.
`noise_system`	The covariance matrix of the system noise as `2 x 2` matrix.
`noise_meas`	The covariance matrix of the measurement noise as `2 x 2` matrix.

Regression

For the regression task we only consider the x-coordinate of the vehicle position and we assume that the vehicle is moving with constant velocity. We want to estimate the position and velocity of the vehicle incrementally for each point in time by using at most the latest w measurements, where $w \in \mathbb{N}$ defines the length of a *sliding window* of measurements. The illustration below shows the case of $w = 5$.



[1]Please refer to the skeleton files for loading data files.

Exercise (4.1):

Implement a *regression* approach for the x-position and velocity of the vehicle for the given sensor measurement data. Use a sliding window over the measurements. The file *regression.m* provides a skeleton for that task. Try different values for w (the length of the sliding window) and observe the precision of the estimated position using the calculated error value.

The following auxiliary function is provided:

`analyze_residual_error`	To inspect the estimation error of the regression model, this function shows the error in x-dimension graphically. The return value is the mean squared error over the total sequence.

The following Matlab function might be useful for this exercise (see the help function of Matlab for further details):

`linsolve`	This function solves a system of linear equations of the form $A \cdot x = B$ with given matrix A and vector B.

Kalman Filter

In the second task the Kalman filter should be used for to estimate the position and velocity of the vehicle in the 2-dimensional case. Again, we want to assume roughly constant velocity of the vehicle. The state vector should contain the position of the vehicle and its velocity, i.e. $\vec{s} = (x, y, v_x, v_y)$.

Exercise (4.2):

Implement a Kalman filter for the estimation of the 2-dimensional position and velocity of the vehicle and apply it to the position measurements given in the variable `measurements`. Use diagonal matrices for Q and R. Vary the entries in the diagonal of Q and R and observe the effects on the output of the Kalman filter. For which choices do we observe reasonable output?

Now, assume that a second sensor is available that measures the relative movement of the vehicle from one point in time to the next one, i.e. a wheel encoder that measures the rotation of the wheels. Assume that theses measurements are available in the world coordinate system, i.e. you do not need to consider the present orientation of the vehicle. These displacement measurements are provided in the variable `delta_motion`, where the k-th row describes the displacement of the vehicle from the k-the point in time to the $k+1$-th point in time.

Exercise (4.3):

Extend your modeling from task 4.2 in order to integrate the relative motion measurements and extend your implementation to cope with the additional type of measurements. Test you implementation with the data provided in the variables `measurements` (positions) and `delta_motion` (displacements).

The following auxiliary functions are provided:

`analyze_state_error`	To inspect the estimation error of the Kalman filter, this function shows the error in x- and y-dimensions graphically. The return value is the mean squared error over the total sequence. The mean error for the given sequence is expected in the range of 0.1 m for both dimensions.
`error_ellipse`	This function may be used to visualize a covariance matrix with an error ellipse. It plots the ellipse centered at the given position with radii according to the given covariance matrix. The larger the radii are, the more uncertain the position is.