Introduction to Data Science:
Homework 1 - Data collection (ROS) and Bayes filter

In this homework, you need to write code for collecting laser and IMU data. Then, you can use Bayes filter to estimate the probability distribution of the data.

The installation steps are as follows:

1. There are two ways to run ROS.
1.1 Linux Ubuntu: Install ROS on your PC/latop.
1.2 Windows: Install Virtualbox with the minibot image.
(If you are a newbie for ROS, try solution 2.)
2. Download the sample code from *http : //github.com/kuoshih/hypharos_minibot* (link).
3. Run the sample code. You should see the real-time laser data on "rviz".

1. Collect laser data (30%):

1. Move the robot for one minute (rosrun hypharos_minibot teleop_keyboard.py) and save the laser data as a .dat file. (You need to modify the "main.cpp" code for saving data).
Hint: use "fprintf" to save float data.
2. Load the .dat file and plot the laser data (laser.mat) as an animation in MATLAB (see Fig. 1)

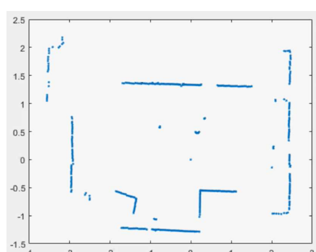

Figure 1: Illustration of Laser scans.

2. Collect IMU data (30%):

1. Copy *getimu.py* to pi/catkin_ws/src/hypharos_minibot/script on Minibot's Ubuntu.
2. Open two terminals.
$roscore (1st terminal will open a ros master node)
$rosrun hypharos_minibot getimu.py (2nd terminal will run python to get imu data)
Then, Minibot will save IMU data for 10 seconds. Rotate Minibot from 0 to 90 degrees and collect the IMU data. The data file will be saved in Ubuntu Home folder.

3. Copy the data file to your Matlab folder.
4. Load first 100 data from the save file.
5. Load the .dat file and plot the IMU data (IMU.mat) as an animation in MATLAB. The Y axis and X axis are the accelerometer output and the time, respectively.
[$HINT$:] You can select one of a_x, a_y and a_z data.

3. Bayesian filter (40%):

Assume you got 10 sequential accelerometer data for detecting the inclination of the Minibot. $Z_{g,1:10} = \{9{,}8, 9.8, 9.8, 8.3, 8.2, 8.4, 8.5, 9.8, 9.8, 9.8\}$. The data is discretized as $Z_{1:10} = \{0, 0, 0, 1, 1, 1, 1, 0, 0, 0\}$. You adopt Bayes filter to estimate the probability distribution of the inclination. There is a Hidden Markov Model in Figure 1. Assume the hidden variables X_i to be boolean ($X_i \in \{1, 0\}$), where i = $0 \sim 10$. Assume the measurement variables Z_i to be boolean ($z_i \in \{1, 0\}$),where i = $1 \sim 10$. Let $P(X_0 = 1) = P(X_0 = 0) = 0.5$. Let the transition matrix $P(X_{t+1}|X_t)$ and sensor matrix $P(Z_t|X_t)$ be given by

$$T = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}, Z = \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix}$$

, where in the T matrix,
$T_{11} = P(X_{t+1} = 1|X_t = 1); T_{12} = P(X_{t+1} = 0|X_t = 1);$
$T_{21} = P(X_{t+1} = 1|X_t = 0); T_{22} = P(X_{t+1} = 0|X_t = 0);$
and in the Z matrix,
$Z_{11} = P(Z_t = 1|X_t = 1); Z_{12} = P(Z_t = 0|X_t = 1);$
$Z_{21} = P(Z_t = 1|X_t = 0); Z_{22} = P(Z_t = 0|X_t = 0);$
Consider the sequences of measurements $Z_{1:10} = \{0, 0, 0, 1, 1, 1, 1, 0, 0, 0\}$
Please answer the following questions.
1) Derive the matrix form of Bayes filter via aforementioned notations. (20%)
2) Given $Z_{1:10}$, compute the filtering estimates of X_t, where t=1$\sim$10. (20%)
This code should display: The probability of the state at each time step.

Code Delivery:

Once you finished the code, upload your code with a zip file to LMS.
1. Compress *main.cpp*, *laser.m* and *laser.dat*
2. Compress *IMU_data.dat*, *IMU.m*
3.1 *BF.pdf* (your math derivation).
3.2 Compress *filter.m*.

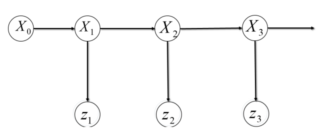

Figure 2: Illustration of Hidden Markov Model(HMM).

Python and C/C++ sample code for robots:

1. Python for TB3 and minibot (link).
2. C/C++ for TB3 (link).
3. C/C++ for minibot (link).