1. **Assumptions of Kalman filter (10%): Give two examples in which the assumptions of Kalman filter do not hold.**
   - Weather Forecasting System
     Since Kalman filter assumes that the task is a linear system, weather forecasting system, a classic non-linear system, can't utilize it. Weather changes are influenced by many complex factors such as atmosphere, pressure, temperature, etc. and the relation of these factors are all non-linear.
   - Stock Market Analysis
     Since Kalman filter assumes that noise follows Gaussian distribution, stock market analysis can't utilize it. In the stock market environment, price movements often do not follow a Gaussian distribution, but rather exhibit fat-tailed characteristics(the probability of extreme price fluctuations is higher than predicted by a Gaussian distribution.)
2. **Orientation estimation using a gyroscope and a magnetometers via Kalman filter (20%):**
   - Find the following terms/values of KF
     - $X = \theta_z$
     - $\mu = \omega_z\ (data2), Z = Z_\theta\ (data2)$
     - $A = 1, B = 0.01, C = 1$
     - $R = std(gyrp\_data1)^2$
     - $Q = std(Z_\theta\ of\ heading\ estimation\ data1\ )^2$
3. **Inclination estimation using a gyroscope and an accelerometer via extended Kalman filter (20%):**
   - Find the following terms/values of KF
     - $X = \theta_x$
     - $\mu = \omega_x, Z = X_\theta\ (atan2(acc\_y, acc\_z))$
     - $A = 1, B = 0.02, C = 1$
     - $R = 0.1$
     - $Q = 0.001$
4. **In Target estimation using laser data (50%):**
   - Find the following terms/values of KF
     - $X = [x\ , V_x\ ,\ y\ , V_y]$
     - $\mu = 0\ , Z = [measure\ x, measure\ y]$
     - $A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = 1\ , H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

- $R = \begin{bmatrix} 0.0004 & 0.004 & 0 & 0 \\ 0.004 & 0.04 & 0 & 0 \\ 0 & 0 & 0.0004 & 0.004 \\ 0 & 0 & 0.004 & 0.04 \end{bmatrix}$
- $Q = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$

5. **(Extra credit) Estimation on Mars (20%): If a robot's estimation system is built based on the gravity of the earth, what could happen when this robot moves on Mars?**

In light of the distinct gravitational forces between Earth and Mars, robot's locomotion on Mars surface could potentially manifest several notable variances.

- Estimation deviation due to gravity differences
  On Mars, gravity is approximately 38% that of Earth's. Consequently, a robot designed based on Earth's gravity may exhibit miscalculations in estimating jumps, movements, or when handling objects. For instance, it might jump higher or farther than intended, or find lifting objects easier than on Earth.
- Alteration in energy consumption
  The disparity in gravitational force could also influence the robot's energy expenditure. Certain activities might require less energy, such as moving or lifting objects. However, tasks demanding stability and precise control might increase energy consumption due to the need for constant adjustments.

Reference

[1] H.-C. Chao, "自駕車學習之路4," Medium. Accessed: Dec. 09, 2023. [Online]. Available: https://hsinchengchao.medium.com/%E8%87%AA%E9%A7%95%E8%BB%8A%E5%AD%B8%E7%BF%92%E4%B9%8B%E8%B7%AF4-5668b45cfb1c

[2] D. B. Or, "Tuning Q matrix for CV and CA models in Kalman Filter," Medium. Accessed: Dec. 10, 2023. [Online]. Available: https://towardsdatascience.com/tuning-q-matrix-for-cv-and-ca-models-in-kalman-filter-67084185d08c

[3] Vince, "Kalman Filter," vswe. Accessed: Dec. 05, 2023. [Online]. Available: https://medium.com/vswe/kalman-filter-987a17bb8f07

[4] "Kalman-and-Bayesian-Filters-in-Python/11-Extended-Kalman-Filters.ipynb at master · rlabbe/Kalman-and-Bayesian-Filters-in-Python," GitHub. Accessed: Dec. 10, 2023. [Online]. Available: https://github.com/rlabbe/Kalman-and-Bayesian-Filters-in-Python/blob/master/11-Extended-Kalman-Filters.ipynb

[5] "L8 Kalman filter," 國立中央大學 「新 ee-class 系統」. Accessed: Dec. 07, 2023. [Online]. Available: https://ncueeclass.ncu.edu.tw/media/doc/134664

[6] “marcociccone/2D-tracking-EKF: This is a simple matlab implementation of a 2D tracking with Extended Kalman Filter.” Accessed: Dec. 10, 2023. [Online]. Available: https://github.com/marcociccone/2D-tracking-EKF

[7] “Kalman filter,” *Wikipedia*. Dec. 03, 2023. Accessed: Dec. 05, 2023. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Kalman_filter&oldid=1188168686

[8] *Nonlinear State Estimators | Understanding Kalman Filters, Part 5*, (May 18, 2017). Accessed: Dec. 10, 2023. [Online Video]. Available: https://www.youtube.com/watch?v=Vefia3JMeHE

[9] *Extended Kalman Filter - Sensor Fusion #3 - Phil’s Lab #37*, (Nov. 01, 2021). Accessed: Dec. 10, 2023. [Online Video]. Available: https://www.youtube.com/watch?v=hQUkiC5o0JI

[10] *卡尔曼滤波器的原理以及在matlab中的实现*, (Jul. 21, 2014). Accessed: Dec. 06, 2023. [Online Video]. Available: https://www.youtube.com/watch?v=2-lu3GNbXM8

[11] “【演算法】卡爾曼濾波器 Kalman Filter,” Jason Chen’s Blog. Accessed: Dec. 05, 2023. [Online]. Available: http://jason-chen-1992.weebly.com/1/post/2022/03/-kalman-filter.html

[12] “Simple Kalman filter for tracking using OpenCV 2.2 [w/ code] – More Than Technical.” Accessed: Dec. 10, 2023. [Online]. Available: https://www.morethantechnical.com/blog/2011/06/17/simple-kalman-filter-for-tracking-using-opencv-2-2-w-code/