This assignment consists of two parts.

Please use Python for implementation using only standard python libraries (e.g., numpy). **Do not use any third party libraries/implementations for algorithms**.

Please prepare a report to accompany your implementation. The report should contain detailed responses to questions. Briefly describe the key findings/insights for the graphs. Ensure the reproduction of graphs (modulo probabilistic execution) for your submission. Submit the report along with your code.

For plotting, please use plotly library as it can generate interactive html plots. Here is an example showing how to plot 3D plots in plotly. Export all your plots in html format and upload them to a drive folder. Ensure that no changes are made to the folder after the deadline. Include the link to the folder in your report, and ensure that the link is working. Also provide snapshots of the plots in the report wherever required.

Part 1 (i) is an optional question. However, attempting this question will earn bonus points.

Additional readings: Artificial Intelligence: A Modern Approach (Ch. 15) and Probabilistic Robotics (Ch. 2 and Ch. 3).

Please refer to the submission instructions on the course webpage. We encourage you to seek clarifications in the problem statements on Piazza should you have any.

As always, submission will be thoroughly analysed for plagiarism; any cases found will be heavily penalised and dealt with as per the course policy.

1. State Estimation using Kalman Filters (60 points)

Consider an airplane flying in a 3D world. A noisy sensor (e.g., GPS positions) provides measurements $z_t = [x'_t, y'_t, z'_t]$. The plane is controlled by providing velocity increments $u_t = [\delta\dot{x}_t, \delta\dot{y}_t, \delta\dot{y}_t]$, which get added to the velocity components $\dot{x}_t$, $\dot{y}_t$ and $\dot{y}_t$ at each time step. The uncertainty in the motions is characterized by a presence of Gaussian noise $\epsilon_t \sim \mathcal{N}(0, R)$. Similarly, the measurement uncertainty is characterized by presence of Gaussian noise $\delta_t \sim \mathcal{N}(0, Q)$. Our goal is to estimate its positions $[x_t, y_t, z_t]$ and velocities $[\dot{x}_t, \dot{y}_t, \dot{z}_t]$ at time instant t from noisy observations $[x'_t, y'_t, z'_t]$.

Environment Setup and Filtering (10 points)

(a) Implement and formally describe the motion and observation models for this problem. Plot the actual trajectory and the observed trajectory of the vehicle.

Unless otherwise stated, make the following assumptions for parts (a) through (f).

1. Simulations and filter updates have to be performed for $T = 500$ time steps at 1 Hz.
2. The vehicle is initially at rest and at starting position $(0, 0, 0)$.
3. The control policy provides velocity increments, $u_t = [\sin(t), \cos(t), \sin(t)]$.
4. The Observation noise is an isotropic Gaussian distribution with a standard deviation of $\sigma_s = 7$.
5. Noise in the motion is parameterised by: σ_{rx}, σ_{ry}, $\sigma_{rz} = 1.2$ and $\sigma_{r\dot{x}}$, $\sigma_{r\dot{y}}$, $\sigma_{r\dot{z}} = 0.01$ forming the covariance matrix R as $diag(\sigma_{rx}^2, \sigma_{ry}^2, \sigma_{rz}^2\sigma_{r\dot{x}}^2, \sigma_{r\dot{y}}^2, \sigma_{r\dot{z}}^2)$ where $diag$ denotes the diagonal elements of R with off-diagonals as zero.

(b) Implement a Kalman filter for the problem to estimate the vehicle state given the assumptions above. Write down formally the model for estimation.

Experiments (25 points)

(c) Plot the actual trajectory $[x_t, y_t, z_t]$, the noisy observations $[x'_t, y'_t, z'_t]$ and the trajectory estimated by the filter $[\hat{x}_t, \hat{y}_t, \hat{z}_t]$. Additionally, plot the uncertainty ellipses for the projection of the estimated trajectory on the XY plane. An uncertainty ellipse denotes the locus of points that are one standard deviation away from the mean.

Assume, for parts (c) through (f) that we have a prior belief over the vehicle's initial state with a standard deviation of 0.01 for each state variable.

(d) We have described 3 types of noise parameters : noise in position update (σ_{rx}, σ_{ry}, σ_{rz}), noise in velocity update ($\sigma_{r\dot{x}}$, $\sigma_{r\dot{y}}$, $\sigma_{r\dot{z}}$) and noise in the sensor measurements(σ_s). Qualitatively describe how the trajectories (actual, observed and estimated) and uncertainty ellipses change on varying the each of the 3 noise parameters. Explain your observations.

(e) The airplane passes through dense clouds where we are only periodically able to acquire the sensor measurements for 50 seconds, followed by 50 seconds of radio-silence. Specifically, we have only measurements from $t = [1, 50] \cup [100, 150] \cup \cdots \cup [400, 450]$. Qualitatively explain the plots obtained for the estimated vs the actual trajectories. Show the evolution of uncertainty in the plane's estimated position projected on XY plane $[x_t, y_t]$ by plotting the uncertainty ellipses.

(f) Plot the estimated velocities $[\hat{\dot{x}}_t, \hat{\dot{y}}_t, \hat{\dot{z}}_t]$ and the true velocities of the vehicle $[\dot{x}_t, \dot{y}_t, \dot{z}_t]$. Briefly explain if the estimator can or cannot track the true values.

Data association (25 points)

Data association problem - Consider a system with n airplanes, indexed by i $\in \{1, 2, \ldots, n\}$. The sensor receives n sets of measurements, $s_t^j = [x_t'^j, y_t'^j, z_t'^j]$, for j $\in \{1, 2, \ldots, n\}$, but does not know which measurement arrived from which airplane. The order in which we receive the measurements may be different at different time steps. Thus, the estimator would now require a strategy for associating observations s_t^j with their respective latent states x_t^i at every time instant t.

Simulate n=4 vehicles in the environments with linear Gaussian motion and sensor models (as above) with different initial state estimates and noise characteristics. At any time instant t, our goal is to estimate the latent states of all the n vehicles, i.e $(x_t^i)_{i=1}^n$ from the unassociated observations $\{s_t^j\}_{j=1}^n$

Implement the following data association strategies and study their behavior in your simulation. Observe and report how these strategies behave in different initial settings. Fiddle with initial settings and investigate the behavior of the estimated trajectories when the actual trajectories approach or intersect each other.

Brute Force (10 points)

(g) Consider all possible permutations of *associations* (n!). First, implement a notion of distance (or suitability) for a particular measurement to belong to a particular trajectory. Using the defined distance, engineer a metric that indicates the "goodness" of an *association* (or a permutation) w.r.t. the already estimated trajectories over all the n airplanes. Choose the permutation which optimises this metric. Mention the intuition behind your choices.

Hungarian (15 points)

(h) The Hungarian algorithm is an optimisation algorithm that operates in polynomial time. You can find the algorithmic details and an example here.

Hint : Minimising the total cost in the worker-job paradigm is equivalent to minimising the total "error" in the airplane-measurement paradigm.

Extra Credit (Optional) (upto 10 points)

(i) Implement a well-performing strategy other than the ones enlisted above. Qualitatively describe the motivation behind your implementation. Enlist the strengths and weaknesses of your approach.

2. Landmark Localization (40 points)

In this part, we extend the problem setup of the previous question (in the single agent setting) by incorporating an additional observation.

Assume that there are certain landmarks (e.g., air traffic control towers) at the following known locations in the environment: $(150, 0, 100)$, $(-150, 0, 100)$, $(0, 150, 100)$ and $(0, -150, 100)$. The aircraft can measure the Euclidean distance (range) to a landmark when its true position within a certain range of the landmark and the measurement is corrupted with Gaussian noise $\eta_t \sim \mathcal{N}(0, S)$. At any instant, the agent receives a single measurement from the nearest landmark and the measurement can be uniquely associated with the corresponding landmark. Note that the landmark-distance observation is in addition to the GPS-positions that the agent is receiving. The remaining problem setup is considered same as the previous question.

Extension for Landmark Observations (20 points)

(a) Formally describe how the additional *landmark*-distance observation can be incorporated in your estimator developed in the last question.

(b) Extend your simulation to account for the landmark observations as the agent moves through the environment. Assume an range of 90m for observing a landmark. Assume isotropic Gaussian noise with standard deviation of 0.01, 10 and 1 in the motion model, GPS-position observations and landmark-distance observations respectively.

Experiments (20 points)

(c) Simulate $T = 200$ steps with the filter updates at 1Hz with the agent starting at $(100, 0, 0)$ with a velocity of 4 units in the +y direction. The velocity increments provided to the plane are $u_t = [-0.128 \cos(0.032t),\ -0.128 \sin(0.032t),\ 0.01]$.

Plot the true trajectory of the vehicle, the trajectory estimated by your filter, and the regions of influence of the landmarks. Additionally, plot the uncertainty ellipses for the projection of the estimated trajectory on the XY plane. Briefly describe your observations.

(d) Next, increase and decrease the uncertainty in the landmark measurements in relation to the uncertainty in the position measurements (as in the previous question) by varying the standard deviation in the landmark observations as 0.1, 1 and 20 and explain your observations.