

自主駕駛車技術 作業 3 報告

李承翰

311511043

1. Introduction

這次作業是要我們實作 kalman filter，我們需要完成 kalman filter 的演算法，再給定輸入 u 以及量測 z 的情況下，找出車子在 2D 平面上移動的軌跡，並且將它畫出來與實際軌跡的 ground truth 進行比較，而這次車子的狀態 x 總共有三個，分別是 $[x, y, yaw]$ ，輸入 u 則分別為 $[diff_x, diff_y, diff_yaw]$ ，觀測 z 為 $[measurement_x, measurement_y]$ 。

2. Briefly introduce your code

Kalman filter 主要分為三個部分，分別為 initialization、prediction、以及 update。

在 initialization 階段我們需要分別初始化一些我們演算法需要用到的變數，首先是 A、B 矩陣，用 $Ax + Bu$ 的方式去表示我們的狀態轉移方程式，而在這邊我們的設計下，A、B 就是兩個 diagonal matrix，對角線值皆是 1。

接著 P 矩陣代表的是我們針對目前狀態的估測的 covariance，在一開始也都是 1。

而 C 矩陣代表我們能夠量測到的資料，因此是一個對角線為 1 的 2*2 的矩陣再來是比較重要的 R、Q 兩個矩陣，分別代表著我們的 state transition error 以及 measurement error，而根據助教所提供的資訊，我們就可以將兩個矩陣填上對應的 covariance。

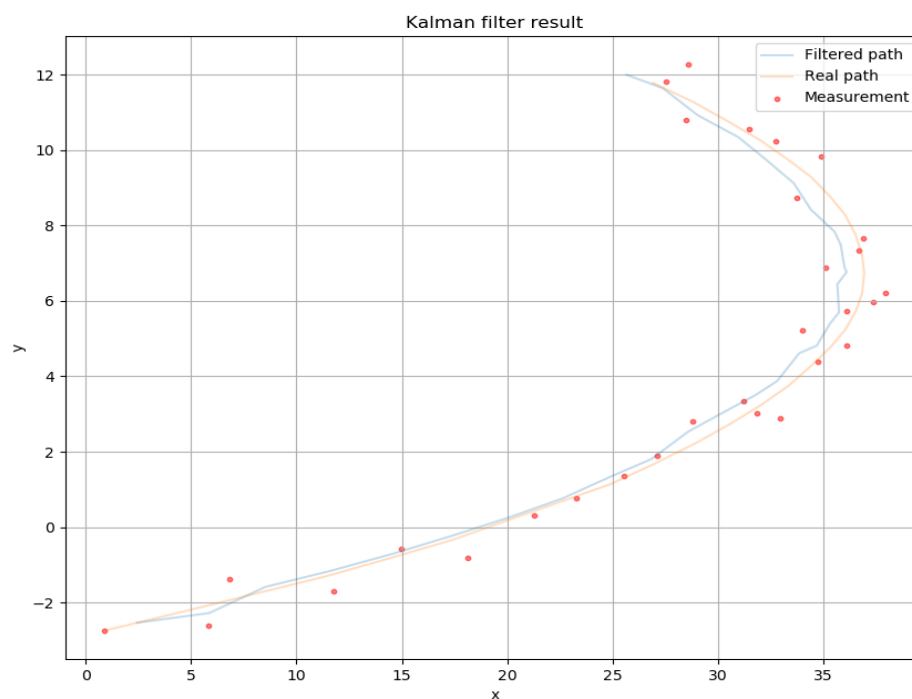
```
self.x = np.array([x, y, yaw])
# Transition matrix
self.A = np.identity(3)
self.B = np.identity(3)
# Error matrix = sigma_t
self.P = np.identity(3) * 1
# Observation matrix
self.C = np.array(
    [
        [1, 0],
        [0, 1]
    ]
)
# State transition error covariance
self.R = np.array([[0.05, 0, 0], [0, 0.05, 0], [0, 0, 0.05]])
# Measurement error
self.Q = np.array([[0.75, 0], [0, 0.75]])
```

接著 prediction 以及 update 的部分，就只是照著 pseudo code 將它實際轉成 python 的程式碼

```
def predict(self, u):
    # self.x = self.A @ self.x + self.B @ u
    self.x = np.matmul(self.A, self.x) + np.matmul(self.B, u)
    # self.P = self.A @ self.P @ np.transpose(self.A) + self.R
    self.P = np.matmul(np.matmul(self.A, self.P), np.transpose(self.A)) + self.R
    # raise NotImplementedError

def update(self, z):
    kalman_gain = np.matmul(self.P[0:2,0:2], np.matmul(self.C, np.linalg.inv(np.matmul(np.matmul(self.C, self.P[0:2,0:2]), np.transpose(self.C)) + self.R)))
    self.x[0:2] += np.matmul(kalman_gain, z - np.matmul(self.C, self.x[0:2]))
    self.P[0:2,0:2] = np.matmul(np.eye(2) - np.matmul(kalman_gain, self.C), self.P[0:2,0:2])
    # raise NotImplementedError
    return self.x, self.P
```

3. Include the output image.



4. How you design the covariance matrices(Q, R)?

如第二小題說的，我直接將助教提供的兩個 noise 的 covariance 的值當作我們 Q、R 兩個對角矩陣的值。

5. How will the value of Q and R affect the output of Kalman filter?

Q 跟 R 會影響我們的 kalman filter 對哪一個來源的 source 有比較高的信任的趨向，當 Q 比較小的時候，kalman filter 就會傾向於去採用 measurement 的結果，而反之當 R 比較小的時候，kalman filter 就會傾向於去採用 state transition 計算出來的結果。