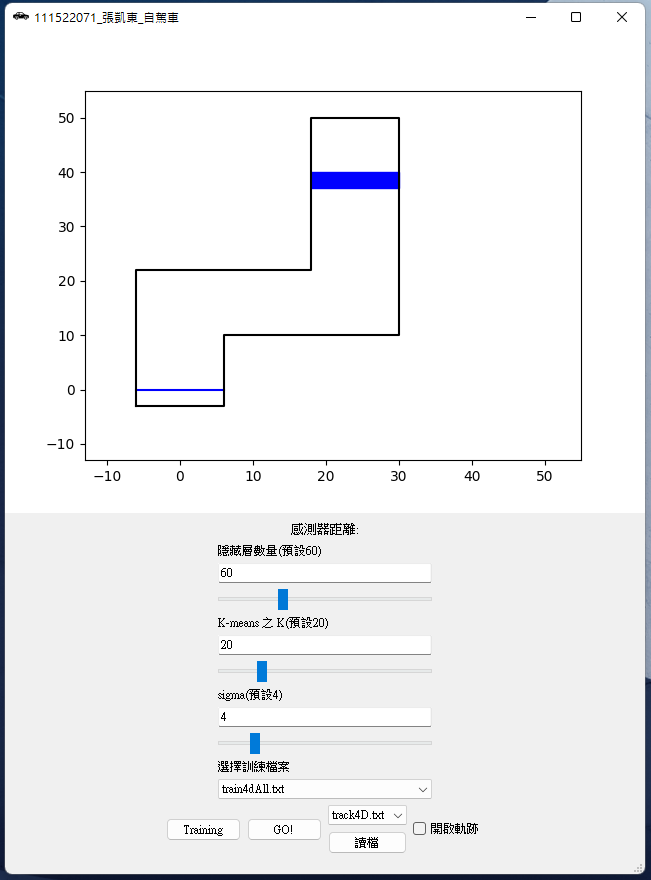
1. **程式介面說明 & 操作說明**

**介面說明:**

主介面從上至下由一張圖片、感測器距離顯示方塊、四個可選擇訓練參數(隱藏層、K-means之K、sigma、訓練檔案)、一個路徑選擇方塊、三個按鈕(訓練、GO、讀檔)以及一個核取方塊(開啟軌跡)組成。四個參數的預設值是跑過4d以及6d後都成功才選出來的數字，直接使用預設值下去訓練出的model沒意外可以成功走到終點。

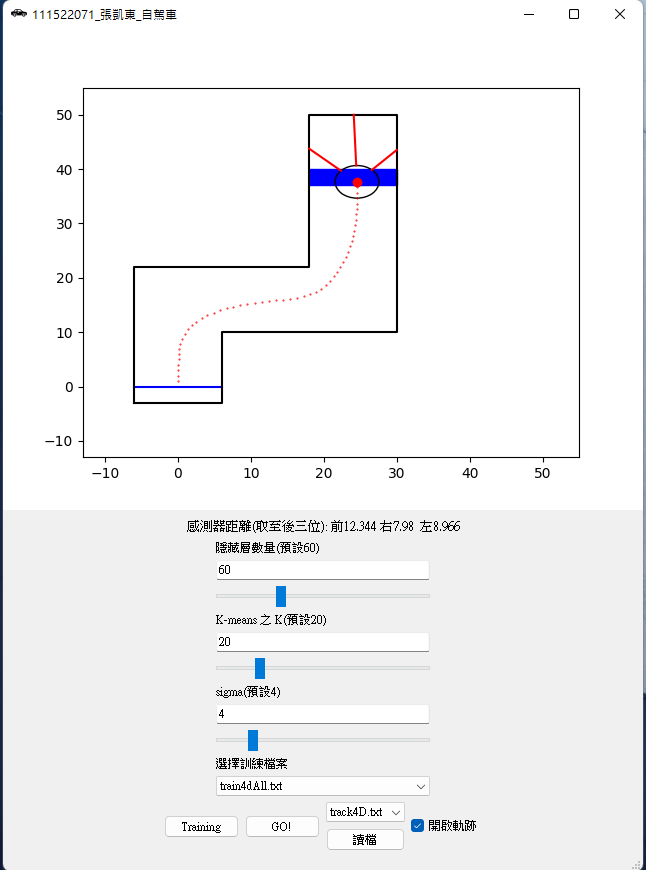
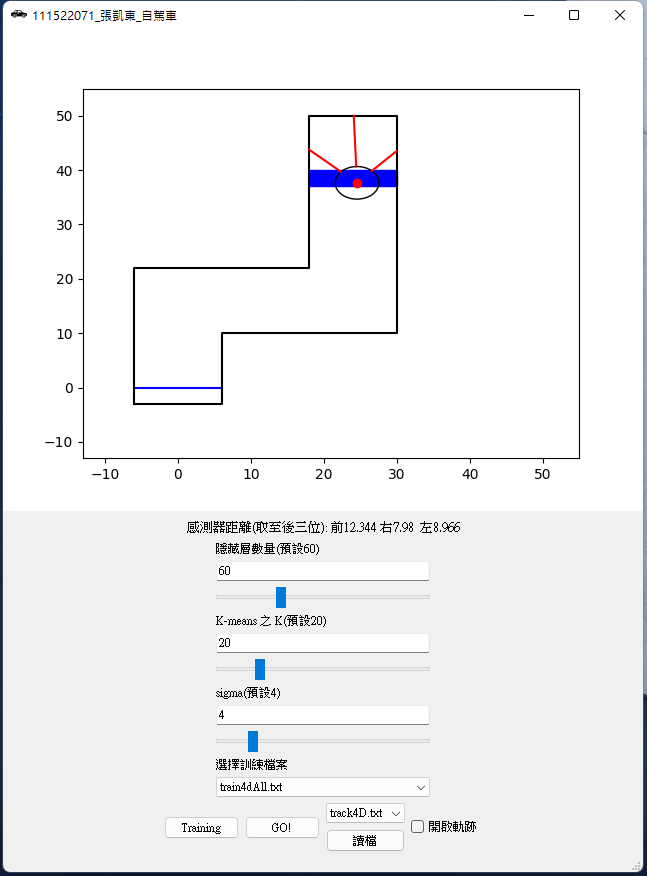


圖一、主介面

**操作說明:**

程式開始運行後會先依照預設值訓練一個預設model(隱藏層60、K=20、sigma=4、file=”train4dAll.txt”)，所以開啟時會停頓1~2秒(預設model訓練完後才會顯示視窗)，主視窗顯示後可以直接點擊”GO”按鈕讓自駕車使用預設model做移動。

想訓練新的model就把四個參數調整完後按下”Training”，調整的方式可以直接輸入數字或是拖曳silder更改數值。Training期間兩個按鈕會disable，直至訓練完成後才會再次enable，訓練完成後就可以按下”GO”用剛剛訓練完的model讓自駕車移動。(訓練注意: 參數請用整數；隱藏層數必須>=K ；K越大訓練時間會越久，K設置上限100時，訓練時間會落在15秒左右(RTX3060)，隱藏層數量跟sigma的影響則不大)



圖二、設有”開啟軌跡”的核取方塊，勾選時會將車子軌跡一併畫出，取消勾選時則無軌跡。

可以選擇”track4D.txt”以及”track6D.txt”後按下讀檔還原路徑，這兩個路徑是使用(layers=60/k=30/sigma=5)

1. **程式碼說明**

**main.py:**

運行視窗應用程式，讀入MyGUI.ui並顯示主介面，import Playground、MyRBFN等class做使用，可以訓練model也可以運行model並將結果視覺化於主介面。

**MyRBFN.py:**

模型主要架構，包含讀入訓練檔案、基底函數計算、虛擬反置矩陣運算、模型訓練、模型預測等等。

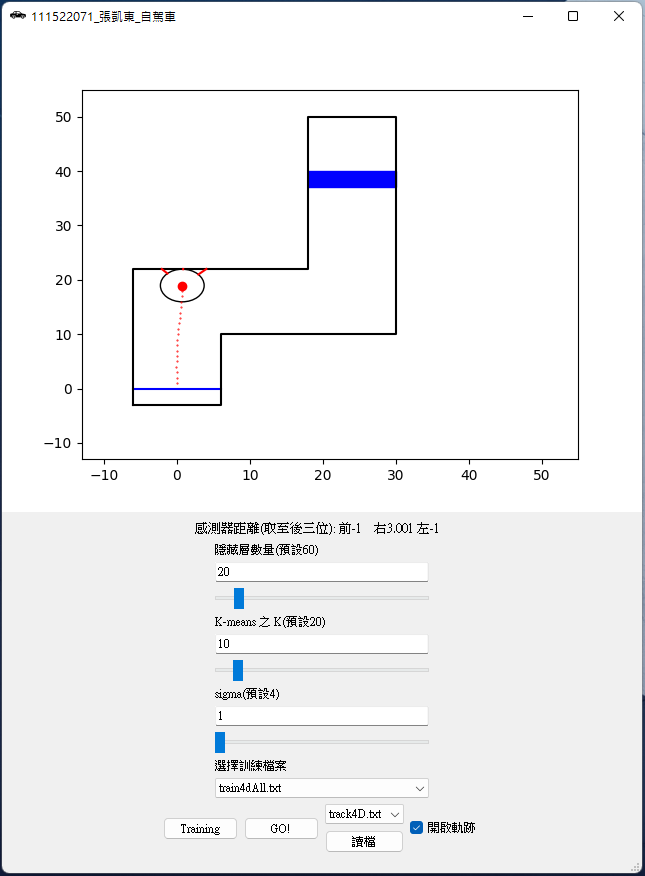
**MyKMeans.py:**

簡單寫的一個K-Means演算法，根據給定的K值選出K個簇中心，如果第t+1次分類完的簇中心等同於第t次則會馬上結束分群，或是達到最大分群次數50次也會結束分群。每次會選出一個資料點與當前的簇中心做歐式距離計算，並分配到最適合的簇。

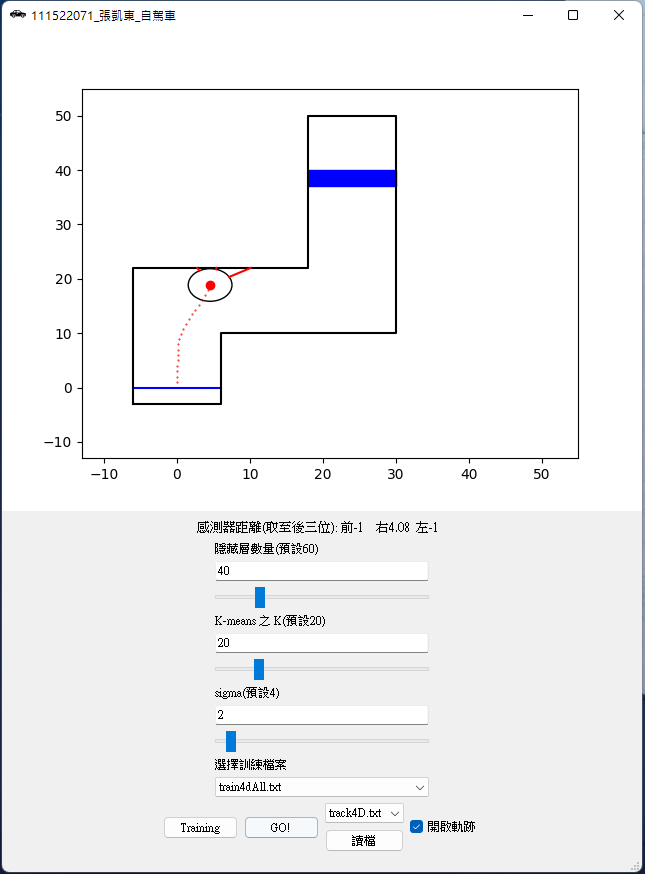
**simple\_playground\_new.py & simple\_geometry.py**

經過微調後的助教給的code。

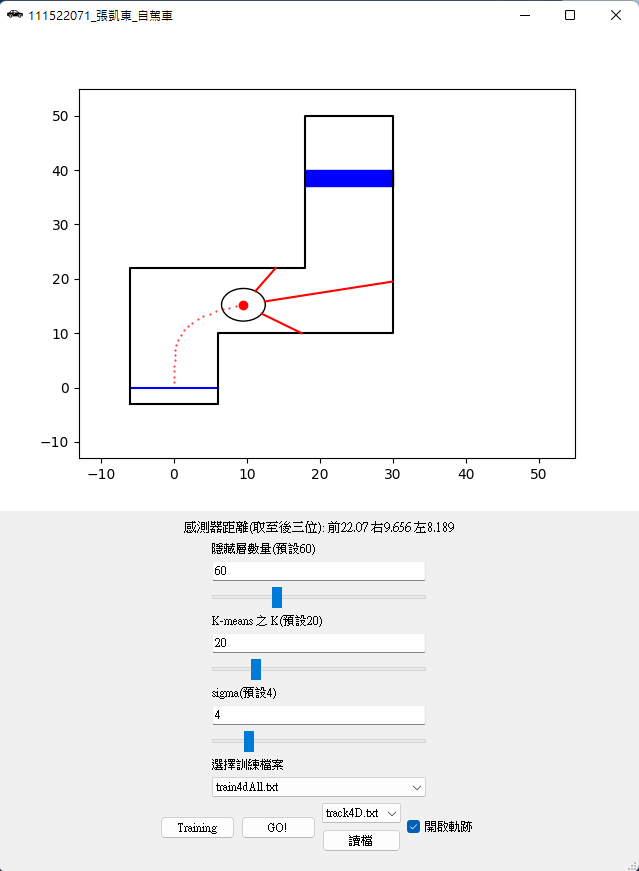
1. **實驗結果(包含移動軌跡截圖)**



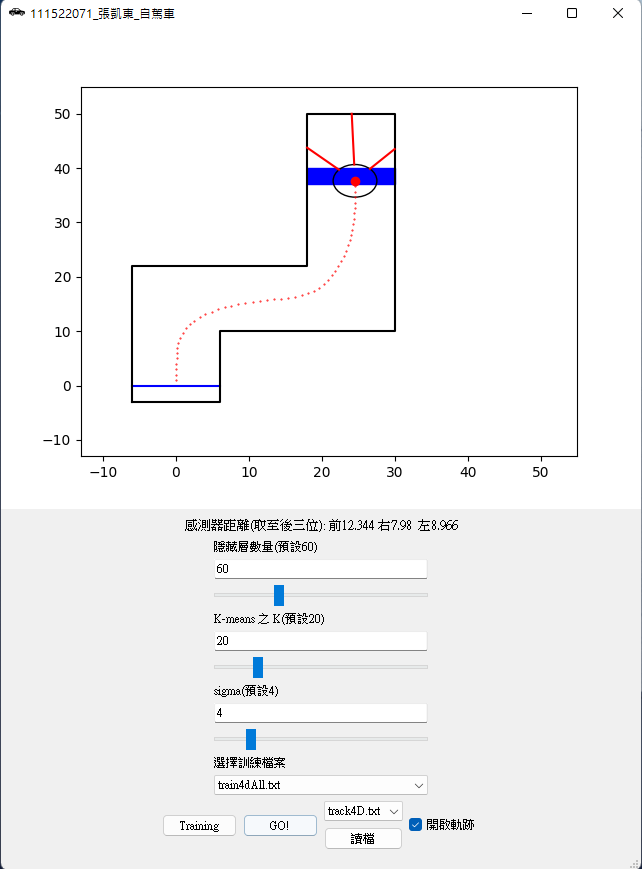
最一開始使用20層以及K=10、sigma=1去做訓練，發現車子只能微微右轉但免不了撞牆。



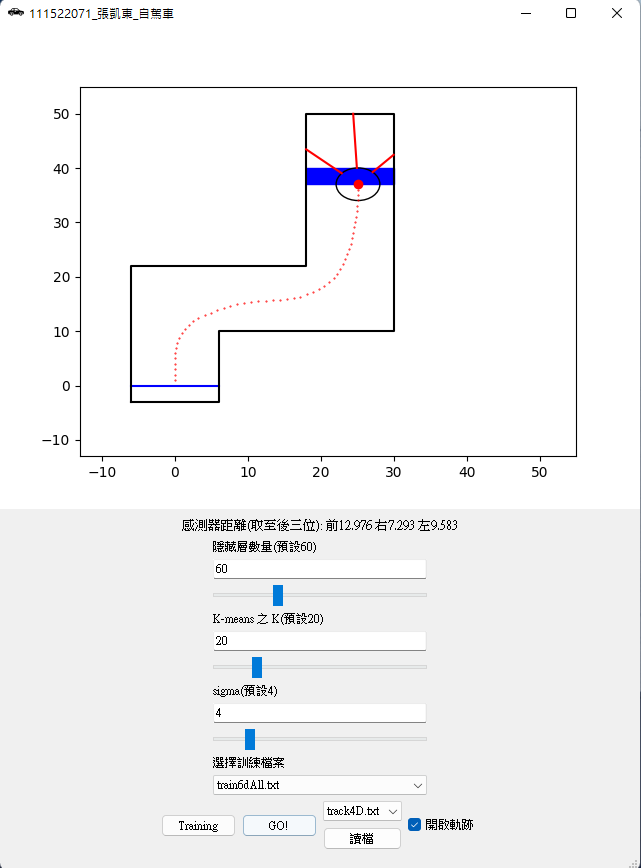
再來調高K以及sigma讓model可以模擬出更接近原始數據的預測，車子右轉的弧度更大了。



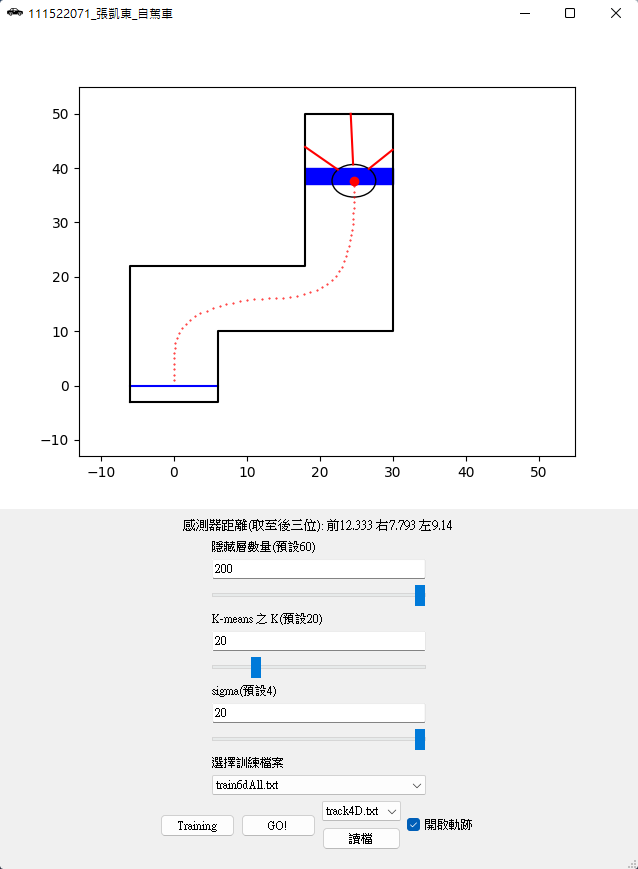
最後則是發現在隱藏層60以及K=20、sigma=4的情況下成功彎過第一個彎，並也成功彎過第二個彎。



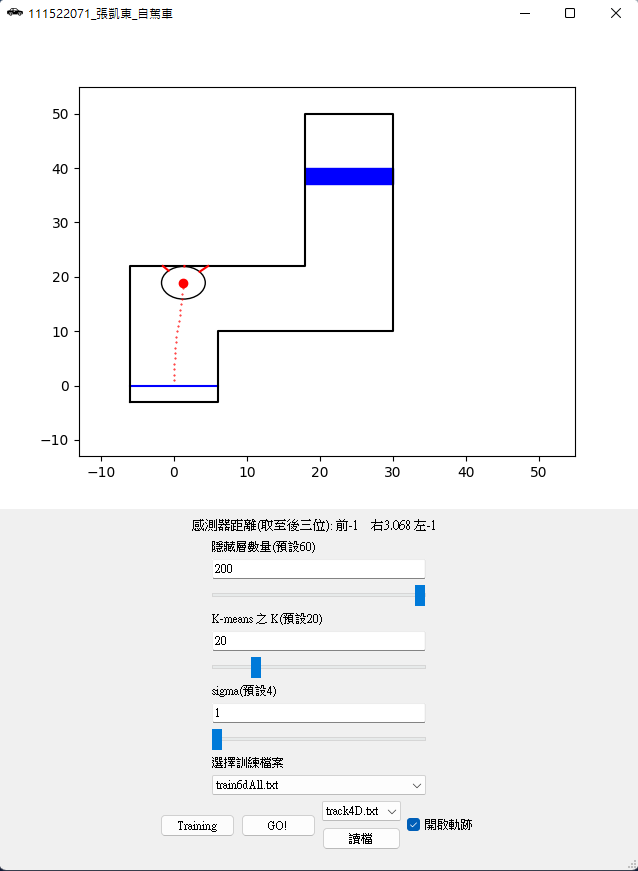
成功到達終點。



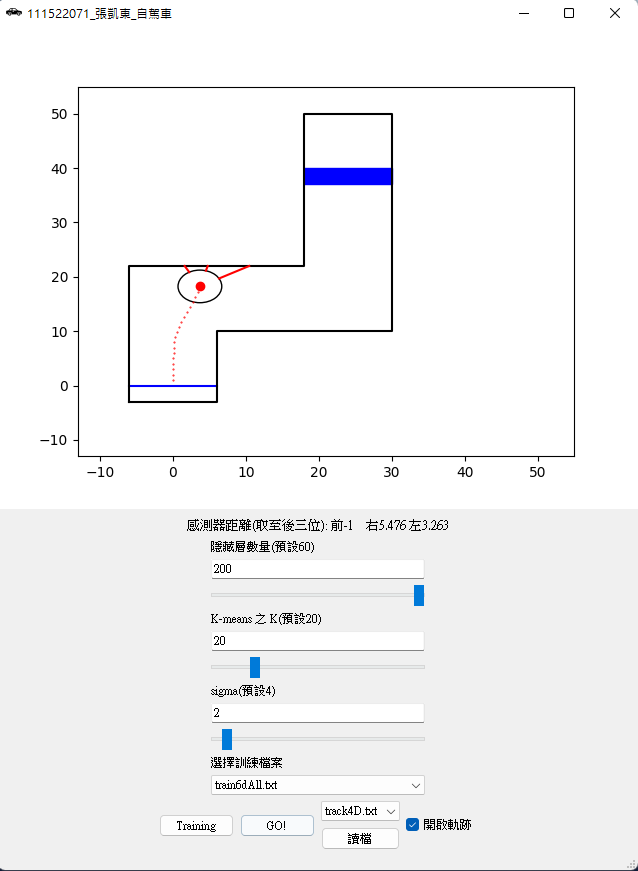
直接使用60/20/4的配置去訓練”train6dAll.txt”也發現可以直接到達終點。



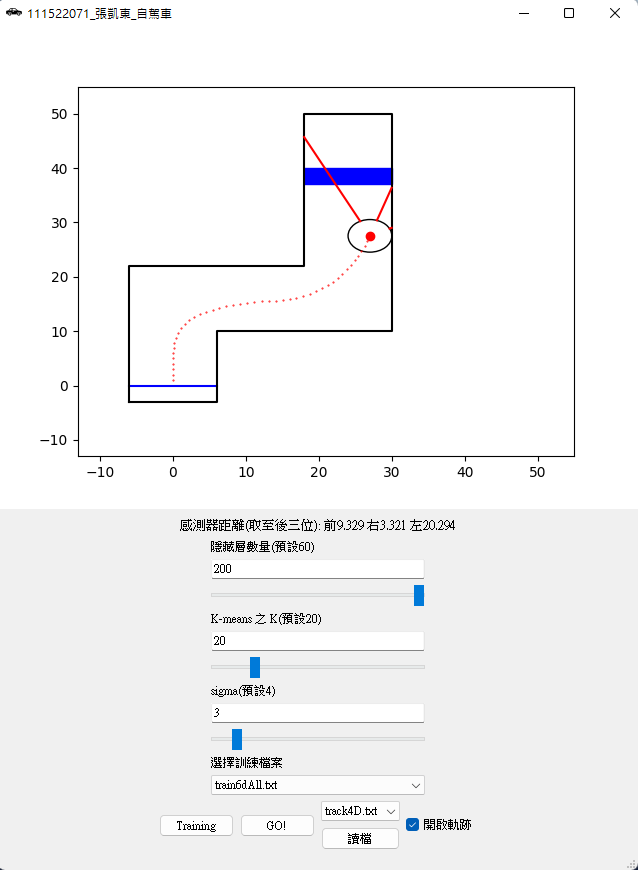
進一步將隱藏層及sigma調到200/20的配置，發現過彎有稍微更滑順的感覺。



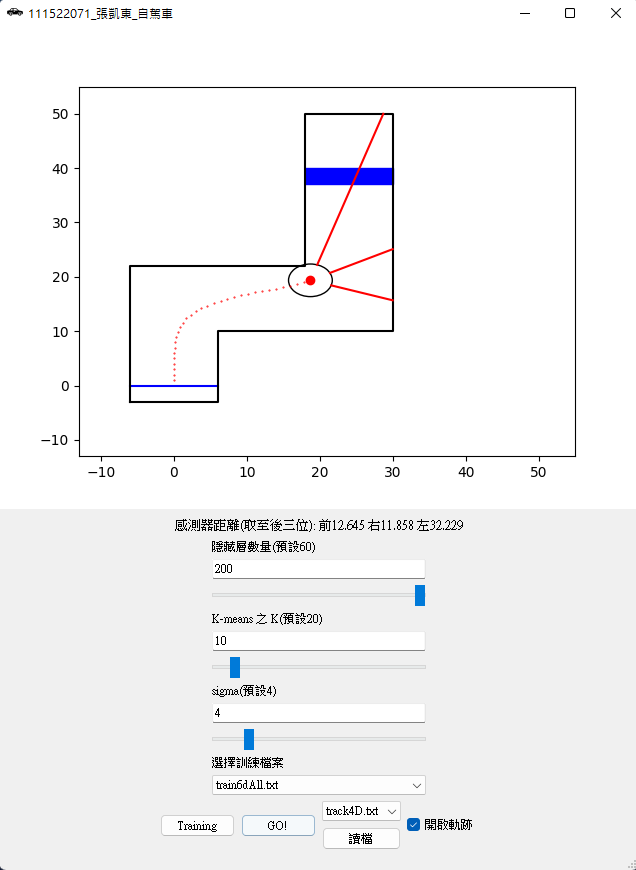
回頭將sigma調回1發現直接撞爛。



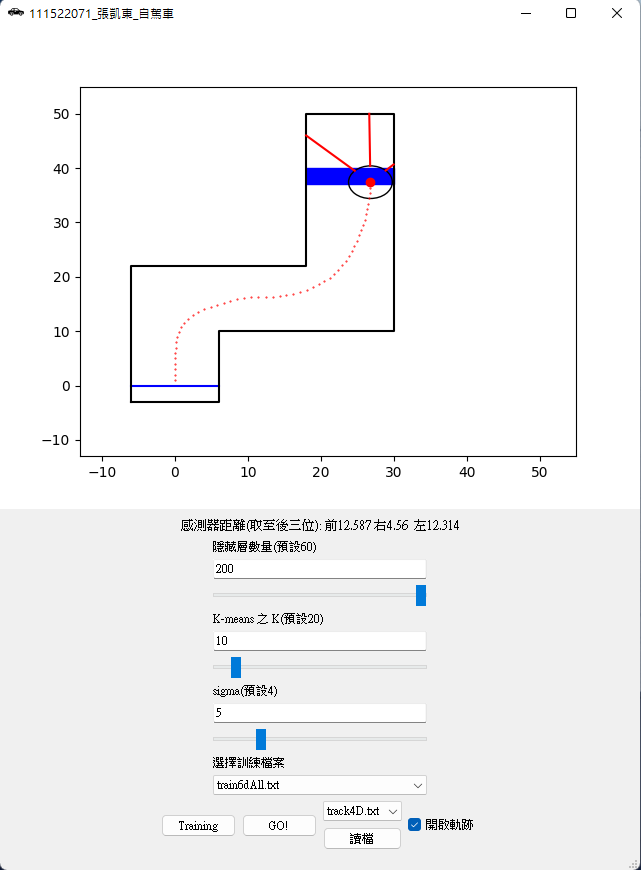
Sigma=2也是撞爛。



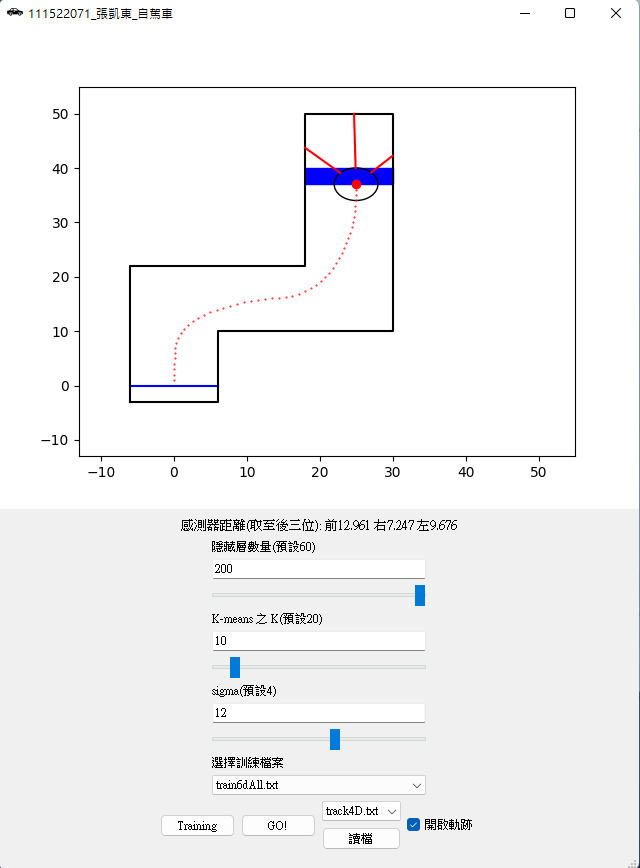
Sigma=3雖然過了第一個彎但第二個彎直接外拋撞爛。



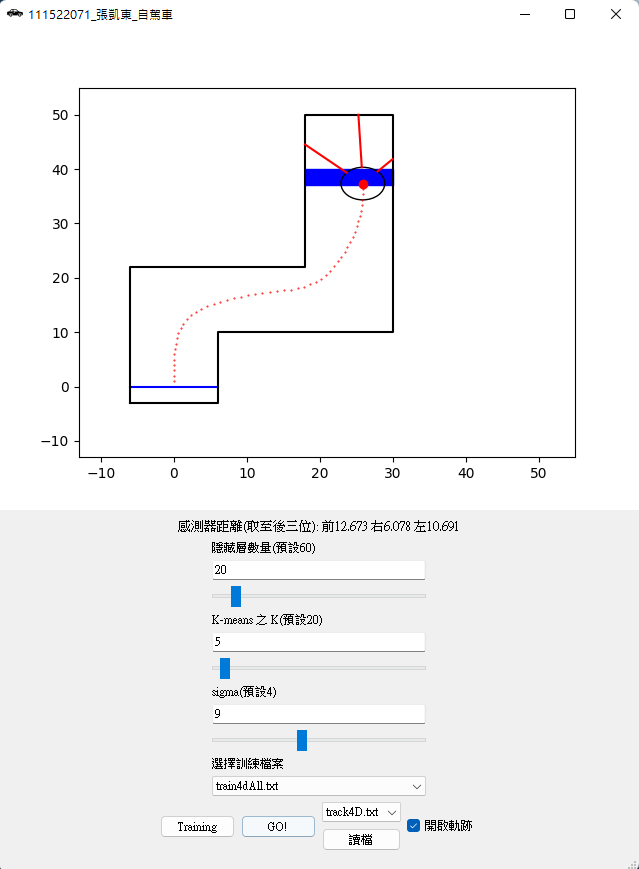
即使隱藏層=200，sigma=4，K太小也會擦撞到牆壁。



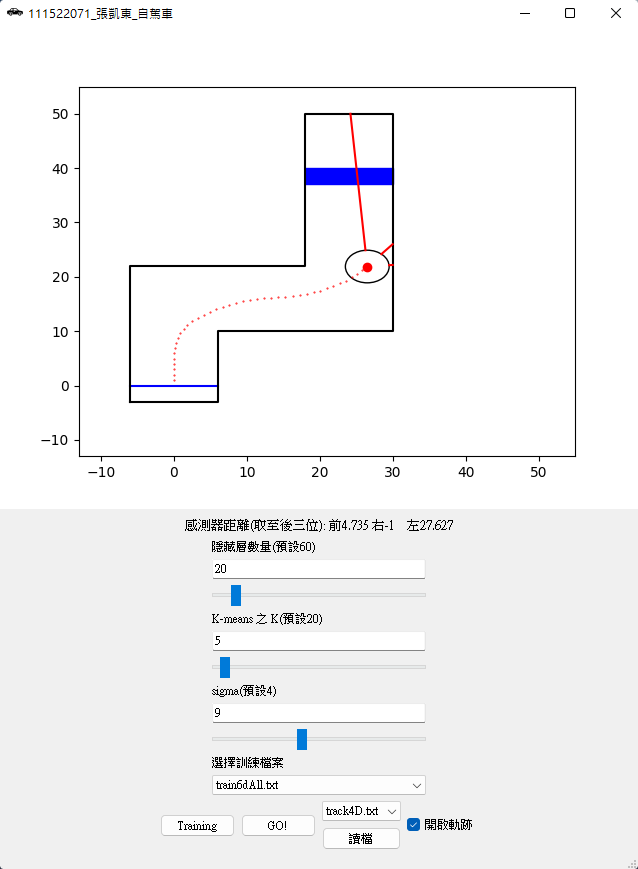
把sigma調大可以成功把外拋的車子龍頭再轉回來。



再次調大sigma發現，即使K不理想加大sigma依然有機會可以讓模型學出正確答案。



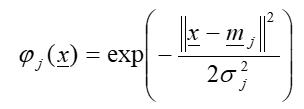
相同為比較艱難的設定下”train4dAll.txt”可以走到終點。



相同為比較艱難的設定下”train46All.txt”走不到終點。

1. **分析**

這次的作業我用的是RBFN模型並使用K-means做群聚分析，基底函數的選擇如圖三，更新model時使用虛擬反置矩陣更新參數。



圖四、選擇之基底函數

在一連串實驗後發現，相比於隱藏層的數量，RBFN在選擇K跟Sigma時更為重要，不好的K無法找出Ground Truth的分布，太小的sigma無法讓模型預測出更靈活的答案，導致出現方線盤轉不過去的現象發生。

這其中我覺得sigma的重要程度又大於K，因為sigma能直接反映出模型的靈活度，越靈活的模型越能找到最佳解，上面也有實驗發現相同的K不同的sigma，比較大的sigma表現較好。

至於訓練檔案的選擇，我發現4d比較容易走到終點，6d因為要考慮車子座標反而不容易走到終點，我覺得是因為維度從四維提升到六維的情況下，需要預測的Ground Truth更複雜，所以6d需要更大的sigma跟K才能得到比較好的表現，相同設定下6d表現則會比4d差。

1. **結語**

以上是這次的作業說明，因為是使用RBFN實作再請助教幫我額外加分，謝謝。