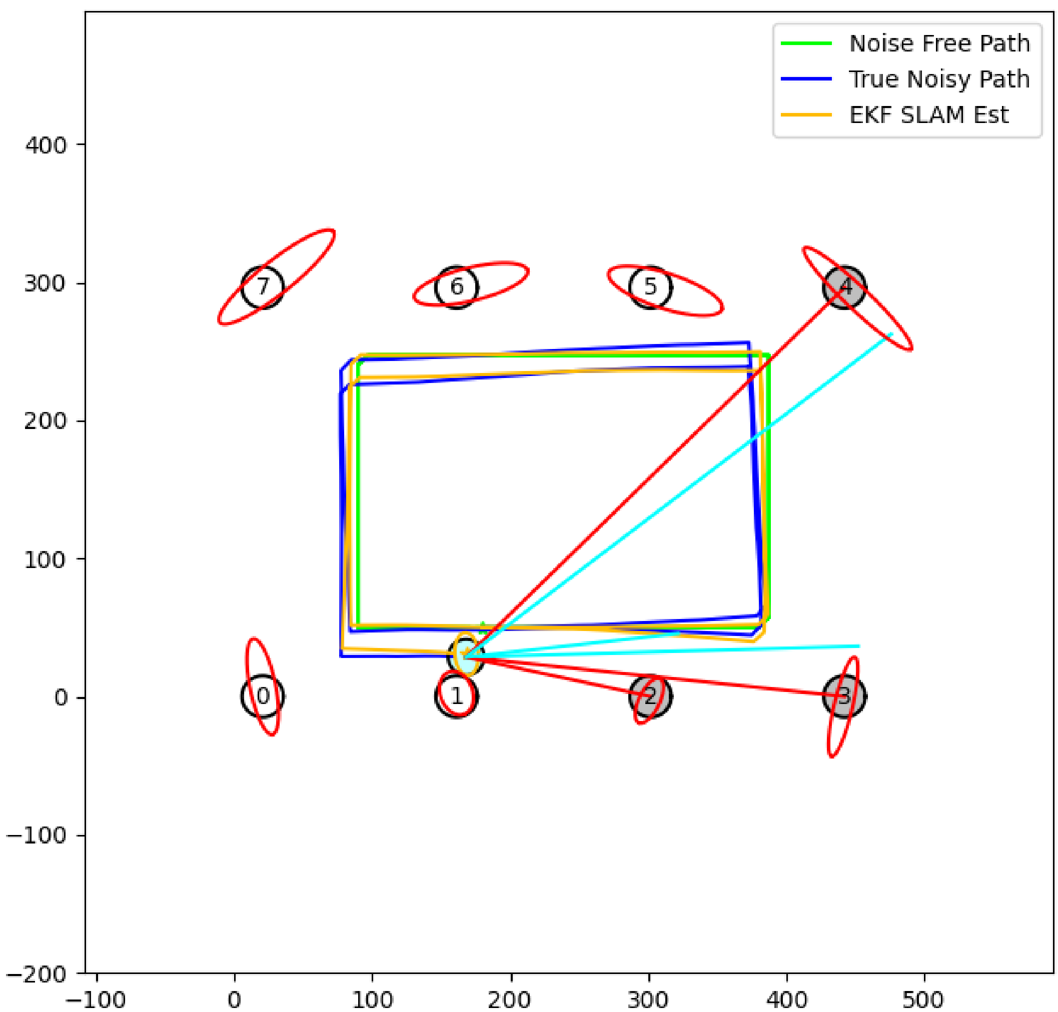
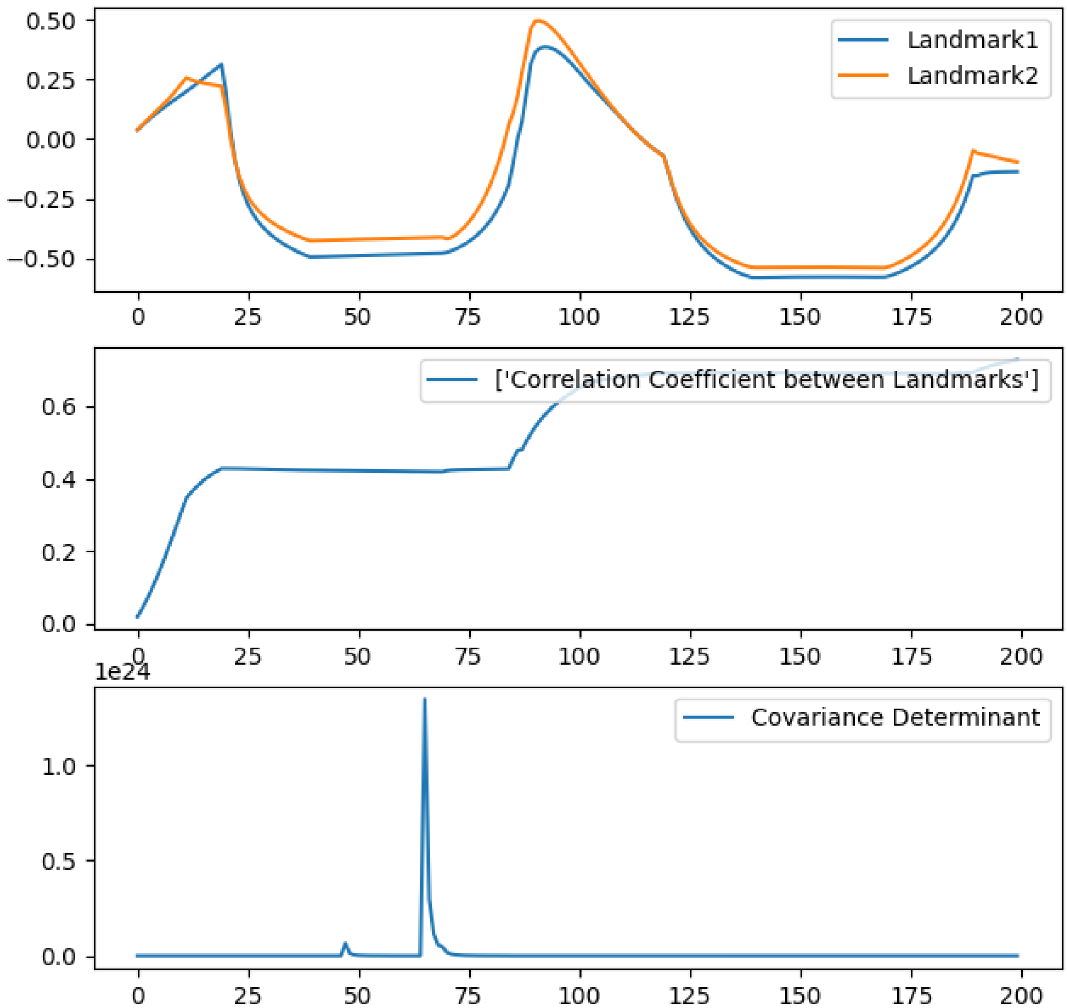
Coding Set 5 Write-up
Daniel Cheney

Task 1

Figure 1. Plot of final map showing 3-σ error ellipses for feature positions and true beacon locations from simulation dataset using known correspondences



All Below
Figure 2. Correlation Coefficients between x coordinates of the feature location for a number of map elements
Figure 3. Correlation Coefficients between x coordinates of the robot and a number of map elements
Figure 4. Determinant of the feature covariance matrix for all map features



The claims in the paper are semi-validated. I think this is a great solution, but it is not the “ultimate” solution. This is a great solution and will work, but must be applied carefully while understanding key assumptions and drawbacks. They assume consistent and linear observations, which may not always be the case. The correlation goes up as the number of steps goes up and more measurements are received, which is good for lowering uncertainty in position.

Batch updates are more preferable because Sigma Bar is updated each iteration and will affect following iterations. In the Batch method, Sigma Bar is updated at once through matrix multiplication. This causes the algorithm to be more stable when using the Batch Update method.

Task 2

The plots below show the data association during NN and NNDG algorithms. The darker the tile the more times the association was attempted.

Figure 5. Plot of data associations as compared to ground-truth with Nearest Neighbor Algorithm

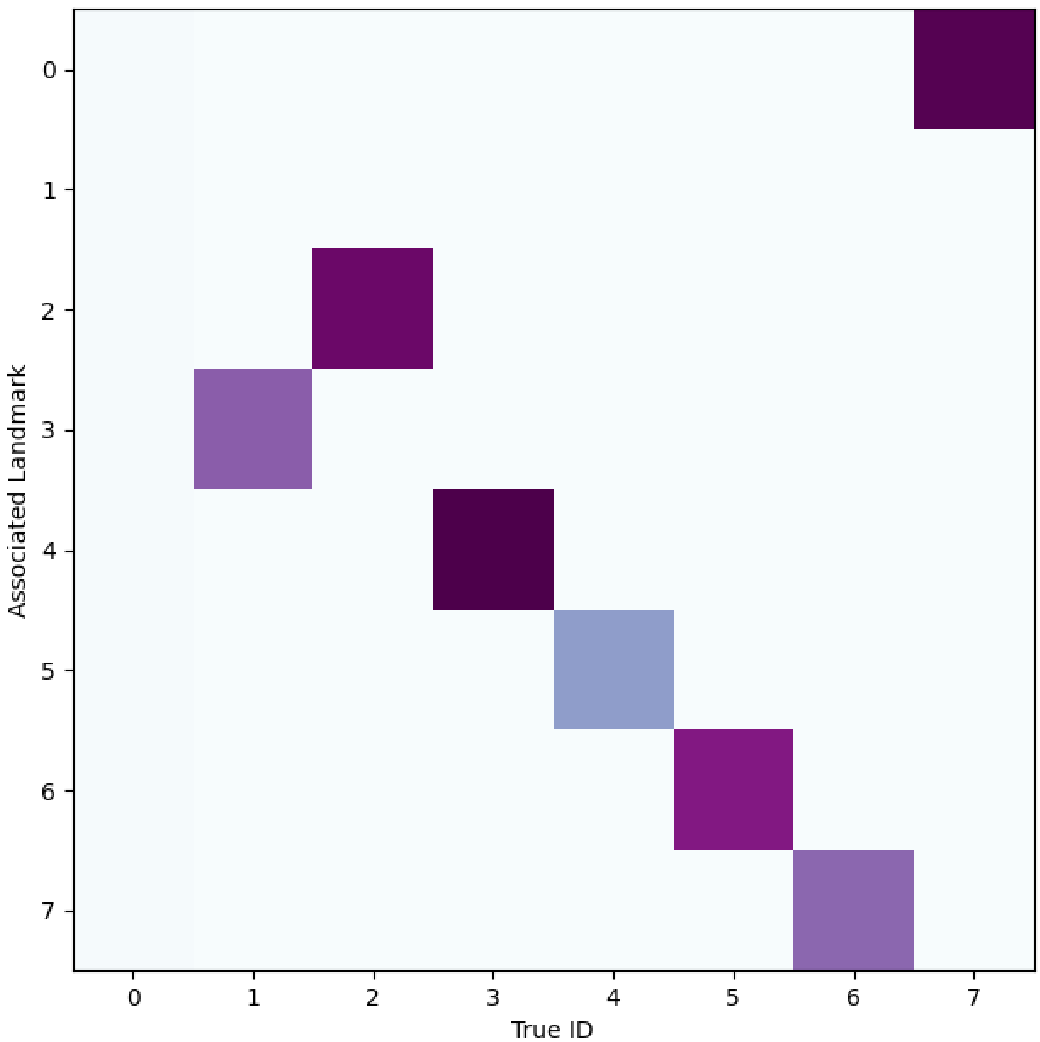
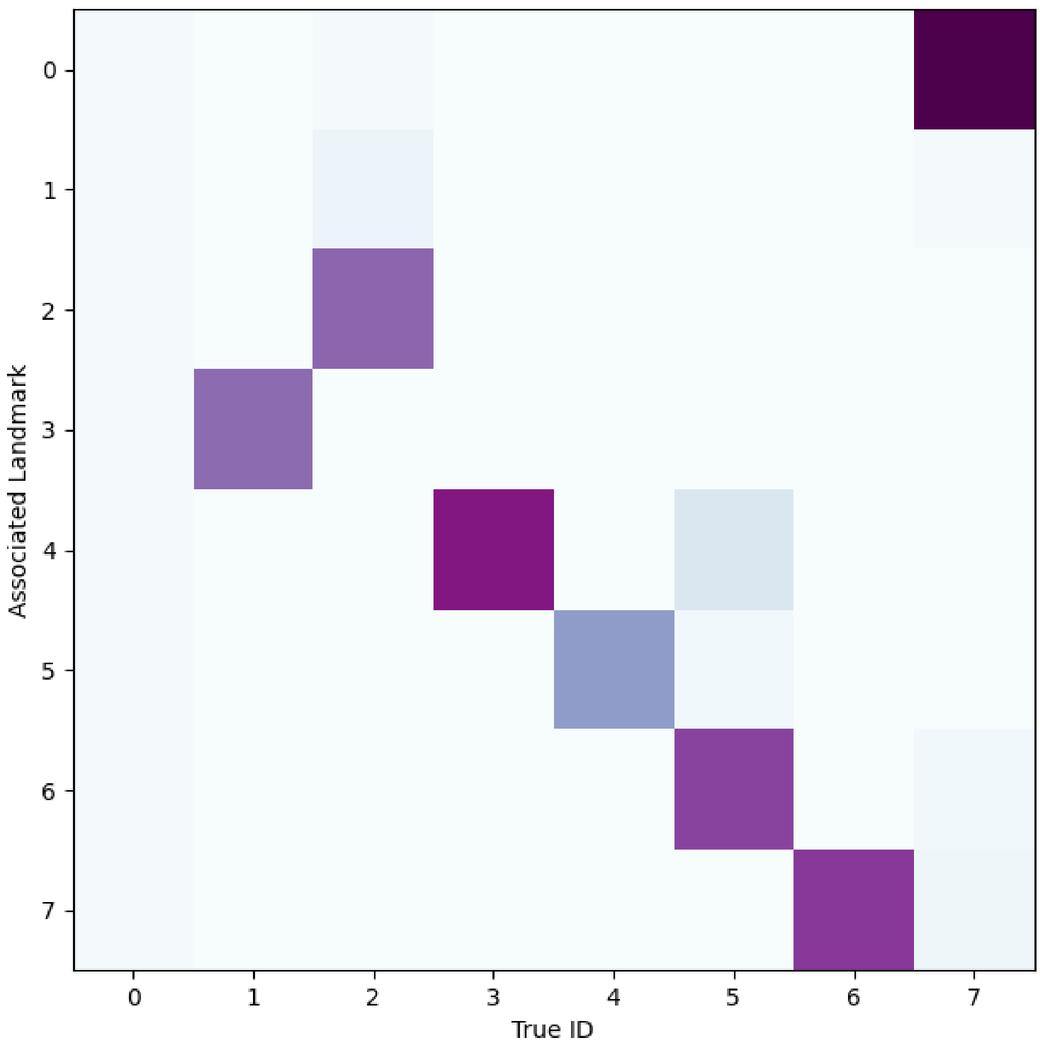


Figure 6. Plot of data associations as compared to ground-truth with Double Gated Nearest Neighbor Algorithm



I had a few iterations that some landmarks were double counted, or depending on my threshold values, landmarks could be missed. This happened especially when landmarks were cluttered. This can affect performance because the worse our map is, the less our uncertainty and position estimates are accurate. This can lead to poor decisions by the robot.

I was able to get similar performance between Nearest Neighbor and Double Gated Nearest Neighbor depending on how I tuned the thresholds. DGNN would be more computationally efficient due to the ability to ignore certain measurements. Overall, I did not see much difference between the two methods.

Nearest Neighbor struggles as there are more landmarks, especially the closer together they are. The double gated method does much better in this aspect. This will have a huge impact on SLAM for path planning because if the robot thinks there is a landmark that is not actually there, it could cause the robot to take an unnecessarily long or dangerous path.

Task 3

Figure 7. Plot of final map showing 3-σ error ellipses for feature positions and true beacon locations from Victoria Park Dataset

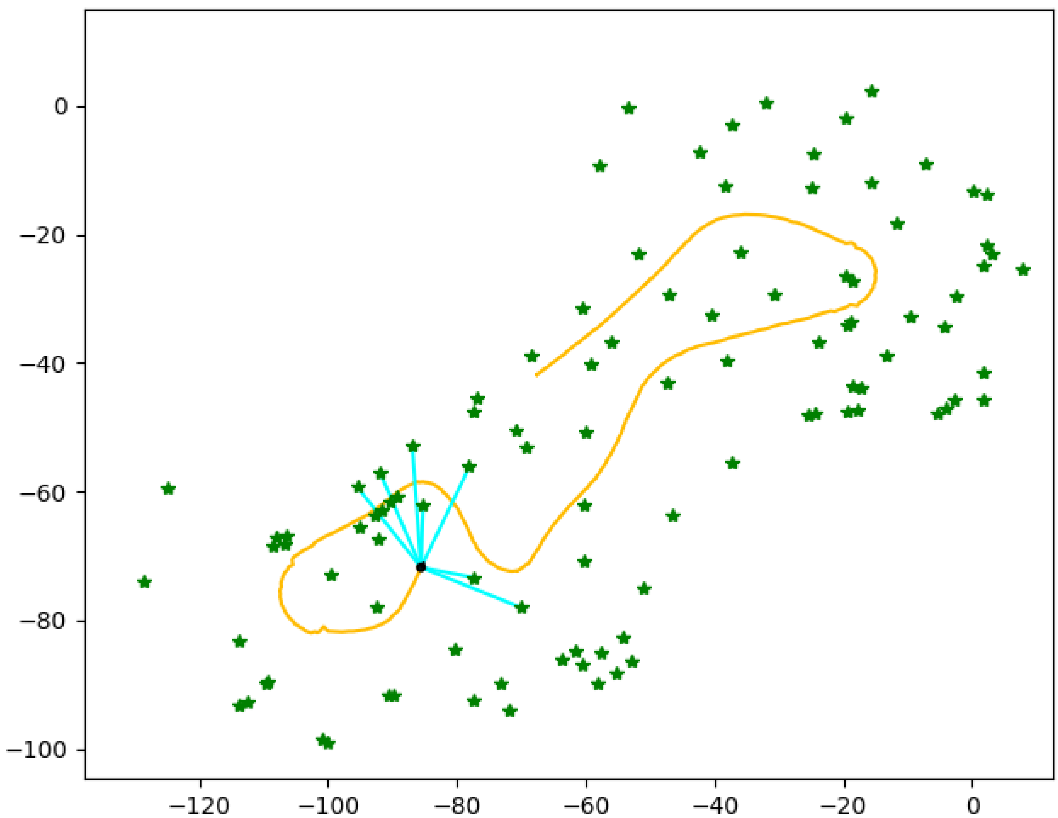


Figure 8. Plot showing CPU time for update and prediction steps based on number of landmarks.