Report Statistics - Research Track 2

Matteo Carlone s4652067

1. Introduction

In this third part of the "Research Track 2" course assignment, I had the opportunity to deal with two fundamental aspects of research in the robotic and non-robotic fields. Being able to collect data from one's project, even if it is already completed, is not always trivial, and the same data would be worthless if statistical analyses did not confirm or not confirm its significance.

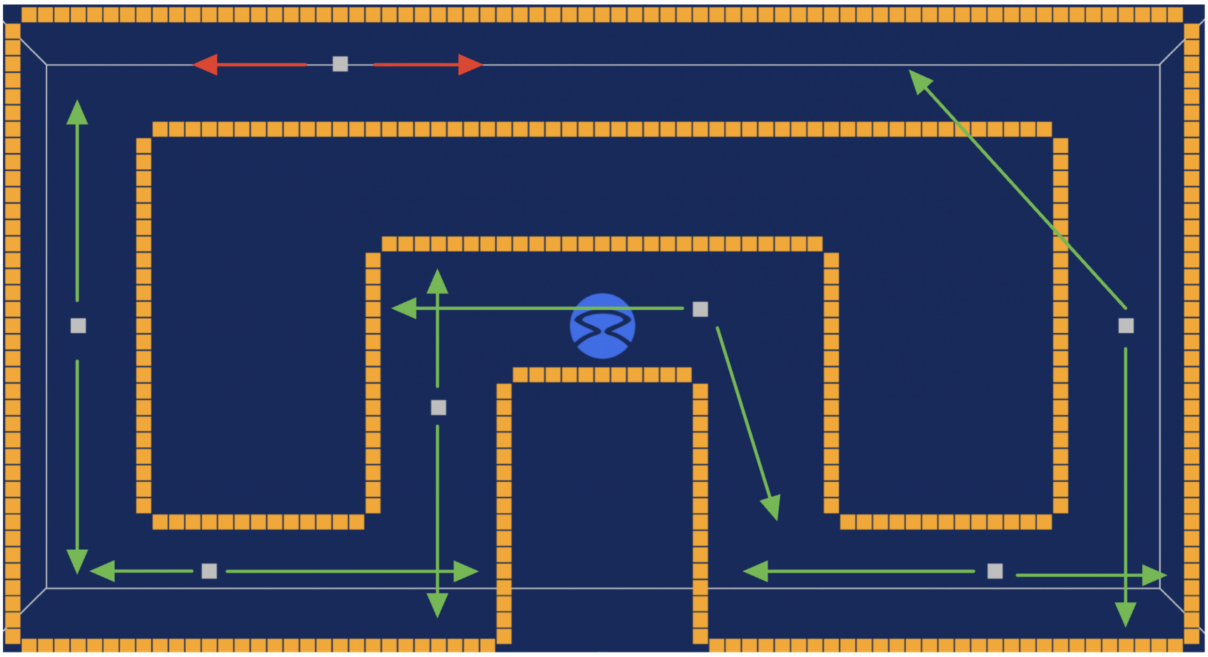
The project on which the analyses were performed is the first assignment of the course "Research Track 1" , and it involves the movement of a holonomic robot in a closed circuit containing silver crates. The robot will have to pick up the crates and place them behind itself before continuing on its anti-clockwise path.
My goal was to collect and analyze data on how well my robot performed in comparison to Prof.'s robot, which is available in this repository:

https://github.com/CarmineD8/python_simulator/tree/rt2

2. Data Collection

I planned to create five different maps that differ from each other in the location of the silver crates. Both my robot and the professor's robot were run for five laps in each of the five configurations, acquiring run-time the following information:

- the robot's distance from the obstacles on the left and right.
- The time it took the robot to complete the path's laps.
- The frequency with which the robot wrongly changes direction.



The last silver crate, as seen in the image, does not have a large excursion because the time it takes the robot to complete one lap of the circuit is calculated as the time it takes the robot to process 7 silver crates. As a result, in order to avoid distorting the time statistics, I prevented varying the position of the seventh crate significantly.

I wrote a Python script with functions to print all the data to.txt files. To save all of the information, I simply had to import the script into both my robot and professor code.

The only data item that required more effort was the percentage of times the robot changed direction by accident: what I have done is set a value to the `offset` attribute of each silver crate from the `sunny_side_up_arena.py` script so that I can distinguish each crate from one to seven.

```
# X,Y are to coordinates of the crate in the map.
# with i that goes from 1 to 7
token=SilverToken(self,count)
token.location = (X, Y)
self.objects.append(token)
token.marker_info = create_marker_info_by_type(MARKER_TOKEN_SILVER, i)
count+1
```

This procedure allowed me to determine whether the robot processed the same crate twice in a row, which, in other terms, indicates that the robot went back in wrong direction.

3. Data Analysis

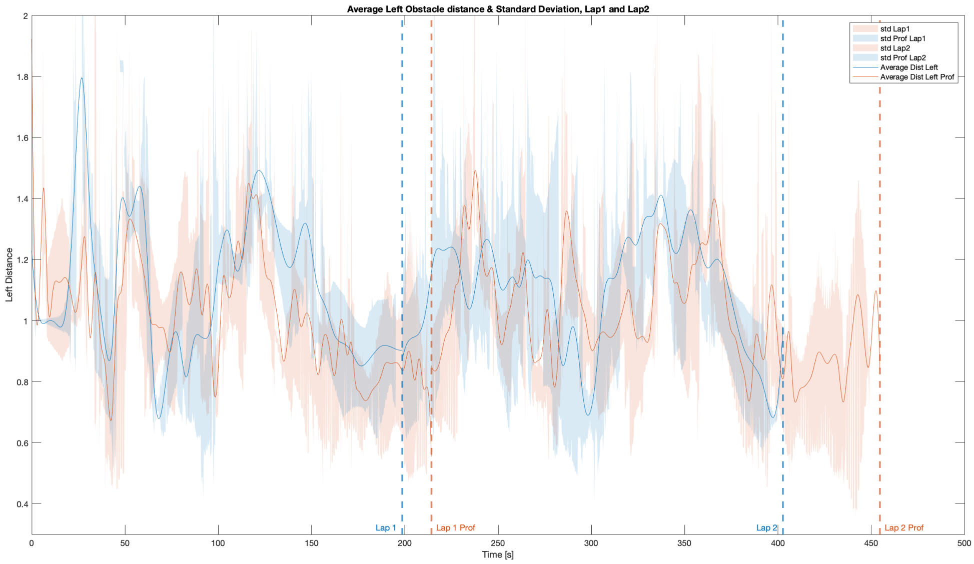
L'analisi dei dati è stata fatta su Matlab. Una volta importati i dati dai file .txt, una fase di "pre-processing" mi ha permesso di organizzare i dati in celle (struct type di Matlab) dividendo in particolare i dati della distanza dagli ostacoli per ogni lap compiuto dal robot.

3.1 Right and Left Distance

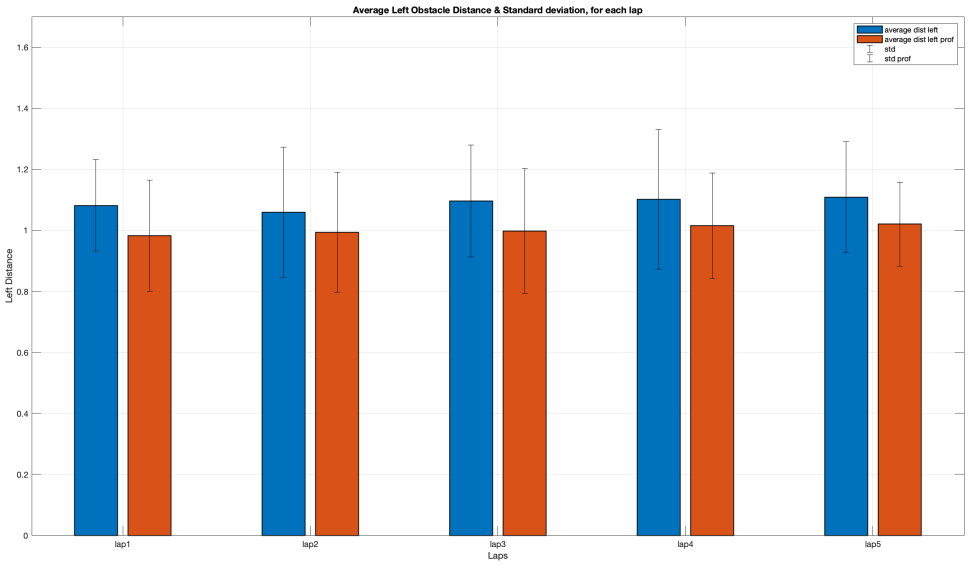
Concerning the distance to obstacles, I focused on highlighting the differences between the average distance held by my robot on both sides and that held by the professor's robot.

The graph below depicts the average trend of the distance held by the robots to their left during the first and second laps of the circuit. Distances greater than 2 that I considered insignificant were eliminated because I was interested in proximity to obstacles. These discarded values were interpolated into the vector's rest. Moreover, to improve the visibility of the average trend in the graph, the data were smoothed by graphing the average envelope of the data. The shading represents the standard deviation.

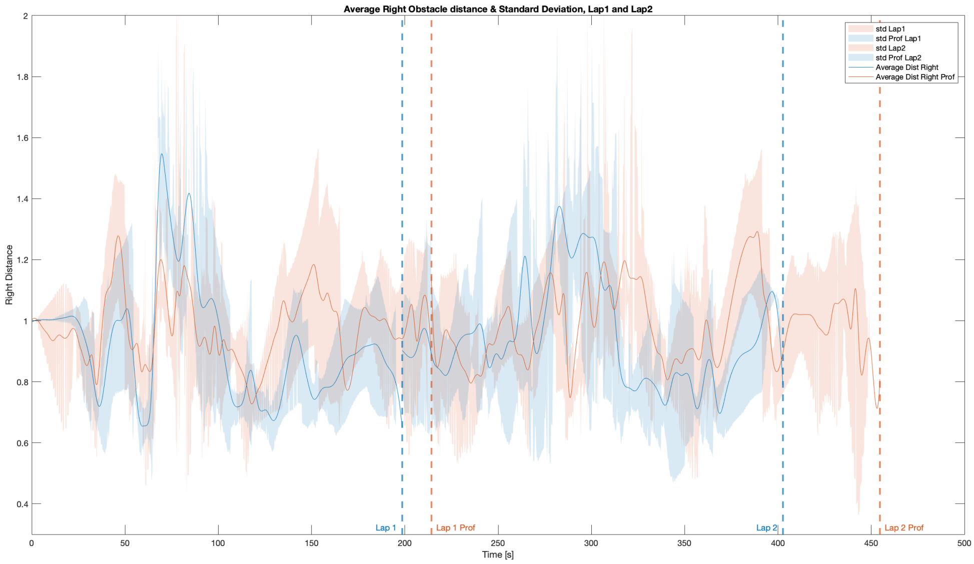
I also tried to emphasize the fact that my robot does not just finish the first and second lap faster than the professor's, but the time discrepancy increases from one lap to another. This is why I decided to represent only two laps here; the time difference would have made the graph indecipherable.



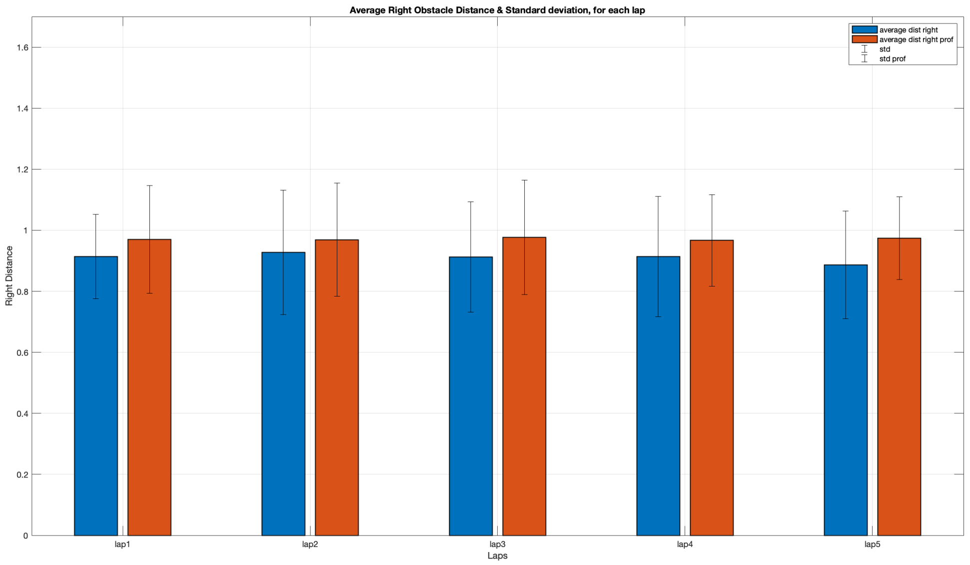
Although already partially visible from the first graph, it is evident in this barplot that on average my robot stands farther from the left-hand obstacles per lap. the standard deviation is also clearly visible.



The next graph is similar to the first one proposed, but represents the distance of the two robots from the obstacles to their right. e can already see here the obvious difference from before, the average distance to obstacles on the right turns out to be less for my robot than for the professor's robot.



The same result best seen in the barpolot. the standard deviation is also clearly visible



To check whether the previously described results were significant, I decided to proceed with a paired T-Test. In a paired *t*-test, each subject or entity is measured twice, resulting in *pairs* of observations. this approach is very suitable for my statistics because the comparison is done between distances coupled by the lap in which they are detected, therefore there's a correlation between the trials.

At First I checked for the paired t-test precondition: the samples should belong to a normal density population.

The made use of a **Lilliefors test,** It is used to test the null hypothesis that data come from a normally distributed population.

Lilliefors Test - Left Distance
h = 0
p = 0.7383

The null hypotesis has been accepted (h = 0) at 5% significance level, so the right distance comes from a normally distributed population with a probability

	given by the p-value of ~ 70 % , I considered this result acceptable to continue the analysis with the paired t-test

Lilliefors Test - Left Distance
h = 0
p = 0.3210

The null hypotesis has been accepted (h = 0) at 5% significance level, so the right distance comes from a normally distributed population with a probability given by the p-value of ~ 30 % ,Altough this outcome is not-significative, I considered this result acceptable to continue the analysis with the paired t-tes

For both the Test the *p*-value has been determined using a Monte Carlo approximation with a maximum Monte Carlo standard error of 0.05.
The Monte Carlo standard error is the error due to simulating the *p*-value.

The Paired Ttest has been performed betweem the mean minor distance between the robot ad obstable in each lap keept by my robot and the one kept by the professer's robot.

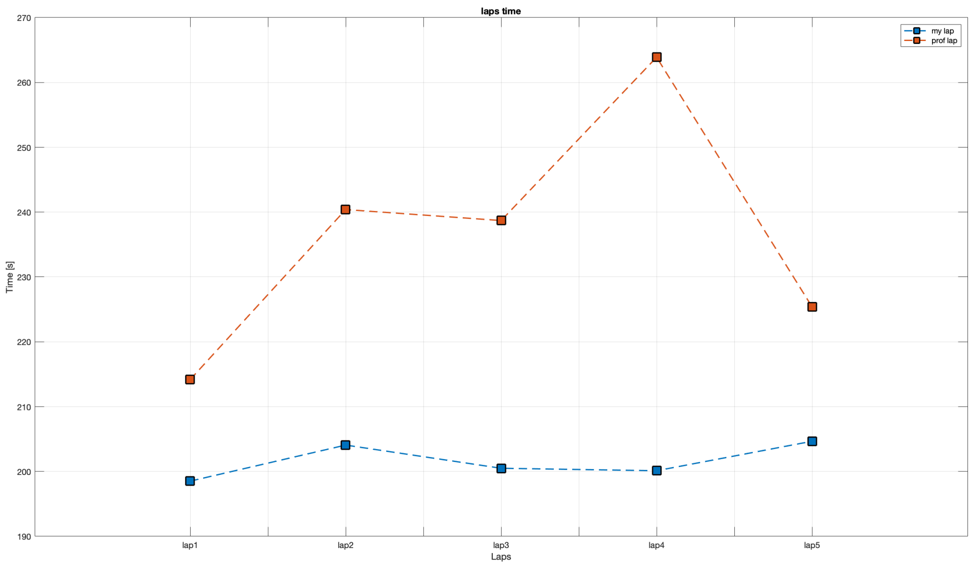
Both the Tests for left and right distances reject the null hypotesys (h=1) at 5% significance level, that means that there's a difference between the two compared robots, The low amplitude of the p-value means that I'm sure with rejection of the null hypotesis, Therefore the outcome is significative. The difference between my value and the professor's was also visible by the graphs mainly due to the different behaviour in the maze.

Paired T-test - Left Distance
h = 1
p = 0.0014

Paired T-test - Right Distance
h = 1
p = 1.4639e-04

3.2 Lap Time

This Graph represent the medium time to conclude a lap a of the circuit. The evidence is that my robot is faster than the professor's one, we can also see that the time increase from one lap to another. Moreover you can notice a decreasing of the time spent to conclude the fifth lap, I noticed that this is mainly due to the fact that the silver crates are, in average, well organized after several laps and that results in a decrease in time."



A Paired T-test has been done to verify some statistical difference between my lap's time and the professor's one . Altough from the graph a difference is visible, there's so significance in the outcome of the test. h is equal to 0 but the p-value is two low to be significative.

Paired T-test - Lap Time
h = 0
p = 0.4072

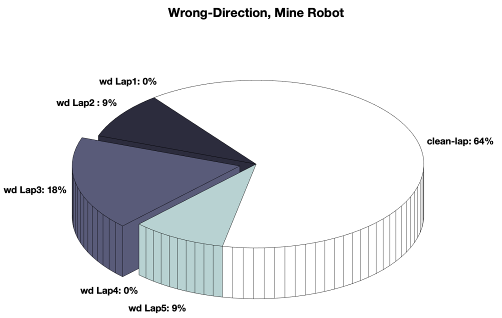
3.3 Wrong Direction

As mentioned in the data collection, I have been able to associate the silver crates with a code ranging from one to 7. If the robot picks up a crate with the same row code twice, it has misdirected.
The one given below is not a true statistic because the misdirecting event is extremely rare and I have too few samples to generate any significance. However, it may be useful to note the percentage of times my robot and the professor's robot went in the wrong direction, and to correlate this data with circuit laps.

Only for this part , I decided to increase the number of trials coming to a total of 10 runnings ,5 laps each. the configuration of the crates in the circuit have always been randomized.

3.3.1 Misdirection - my robot

	Probability of Wrong-Direction
Lap 1	0 %
Lap 2	9 %
Lap 3	18 %
Lap 4	0 %
Lap 5	9 %



3.3.2 Misdirection - Professor's robot

	Probability of Wrong-Direction
Lap 1	6 %
Lap 2	25 %
Lap 3	13 %
Lap 4	0 %
Lap 5	6 %