

RT2: Report Assignment 1 pt. 2

Matteo Maragliano

April 2022

Chapter 1

Introduction

The 2nd part of the first assignment of Research Track 2 consisted in realizing a statistical analysis on two different type of robot driving:

- one realized in RT1 Assignment 1;
- one provided by the Professor to have a term of comparison.

The analysis consisted in collecting a certain number of data from some experiments on the robots' behaviour and then states which one is "better" in that specific term.

Our experiments were based on two different measurements:

- lap time: the time the robot used to complete a single lap pf the circuit;
- distance: the effective distance covered by the robot during each lap.

Moreover, to have a higher number of results we also provided one additional type of arena, properly designed, faster because just a rectangular and without obstacles.

Just for a reminder: the goal of the robot is to drive along the arena circuit, grab some silver tokens found along the path and put them backwards before continuing the lap.

The second arena was used to evaluate just the time and the distance of the robot without the additional task of the silver tokens.

Chapter 2

Realization

In order to minimize the additional lines to add to the original code it was provided a sort of *library* containing the functions to compute the statistics.

The program is able to write into some dedicated files *.txt* chosen through a parameter passed during the program launch.

2.0.1 Lap Time

The lap time test was computed by the function *time()* which allows to store the instant of time it is called into. This is called at the beginning and at the end of the lap, it is computed the difference and all these values are written into the file.

To have a reliable number of data we computed 30 laps, both for the *original* and for the *fast* arena, both for the *assignment* and the *Prof's* robot.

2.0.2 Distance Travelled

The distance travelled was a little more tricky to be computed. We started from the assumption that the robot started always from the same position in the arena.

The function used to compute the Euclidean distance between two positions, the *previous* and the *current* one of the robot; after that the *previous* is updated with the *current* value. All these values are added to a global variable storing the total distance travelled in each lap. Once a lap is completed, the distance value is written into the dedicated file.

Also for this test we collected 30 values for each robot and each arena.

Chapter 3

Results

All the results obtained were analysed by the help of MATLAB software in order to have a reliable and objective analysis.

3.0.1 Hypothesis

We started from a null hypothesis H_0 which states both robot controller are equally good.
We wanted to see if $\mu_{assignment} = \mu_{robot-sim}$.

So, we tried to reject this hypothesis by demonstrating one is better in term of lap time (speed) or distance travelled (efficiency in driving): both after checking their are good for our statistical analysis.

3.0.2 Procedure

All the data were imported into the MATLAB software by an appropriate parse to have them into a numeric form starting from a text one.

Data elaboration

We computed:

- μ : the mean value of all the data recorded;
- σ : the standard deviation of all the data recorded.

For the standard deviation we used a dedicated MATLAB function, while the mean value was computed manually since easier to do.

Results visualization

Once μ and σ are obtained we plotted the results:

- one plot with the data not sorted and the mean value μ computed;
- one plot with a histogram and its relative normal distribution obtained with μ and σ computed.

The resulting plot are:

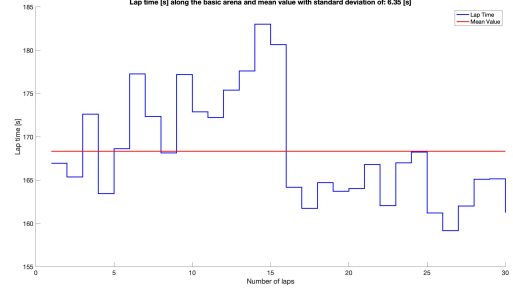
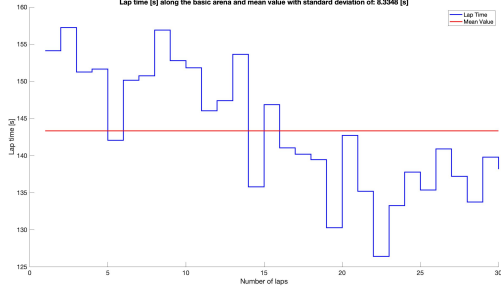


Figure 3.1: Data visualization, not sorted, and mean value for the first arena for lap time.

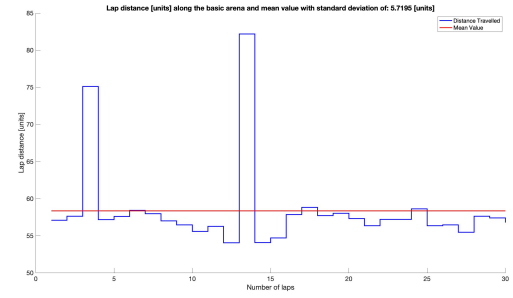
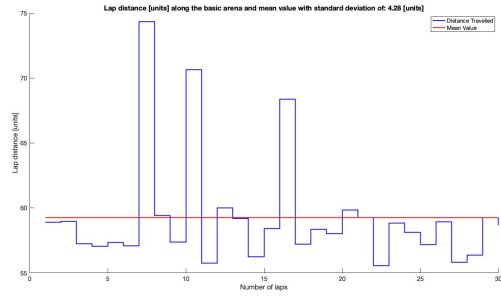


Figure 3.2: Data visualization, not sorted, and mean value for the first arena for distance travelled.

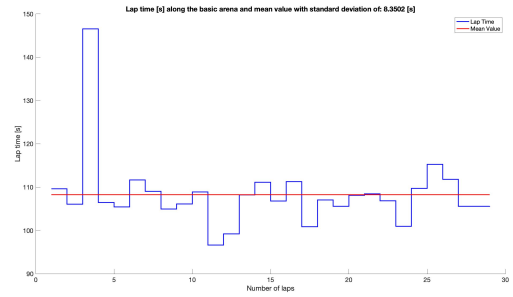
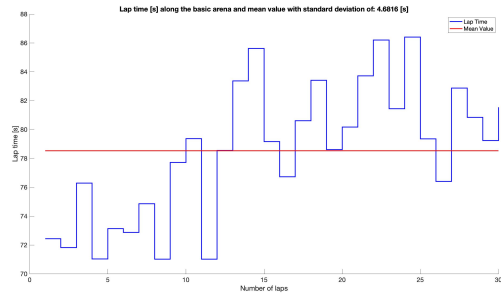


Figure 3.3: Data visualization, not sorted, and mean value for the second arena for lap time.

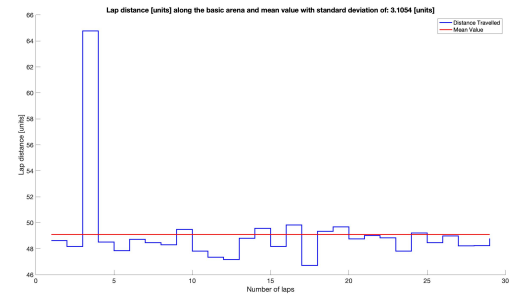
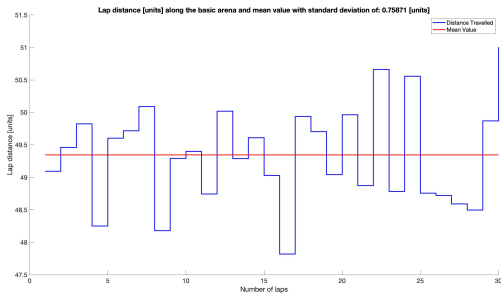


Figure 3.4: Data visualization, not sorted, and mean value for the second arena for distance travelled.

For the probability density function:

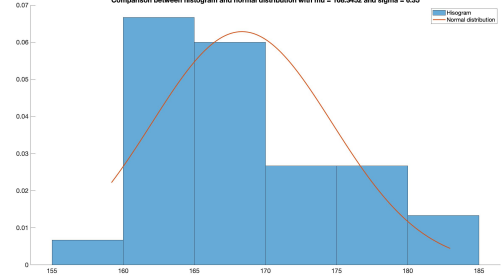
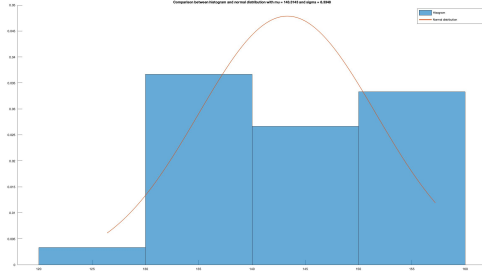


Figure 3.5: Data histogram and probability density function visualization for the first arena for lap time.

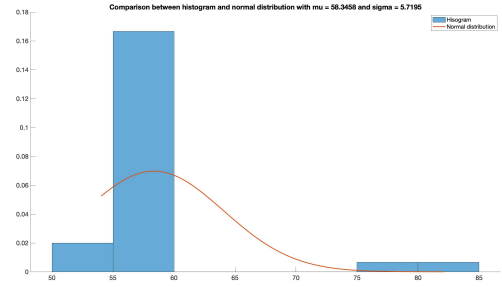
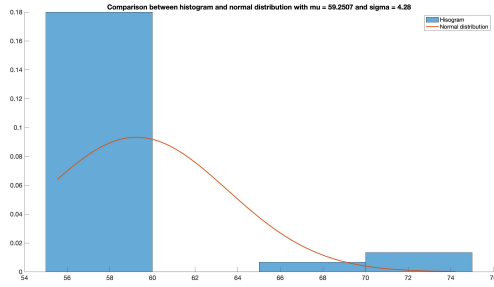


Figure 3.6: Data histogram and probability density function visualization for the first arena for distance travelled.

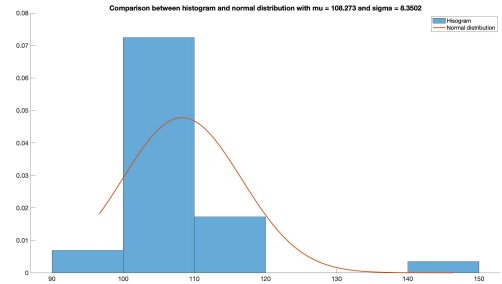
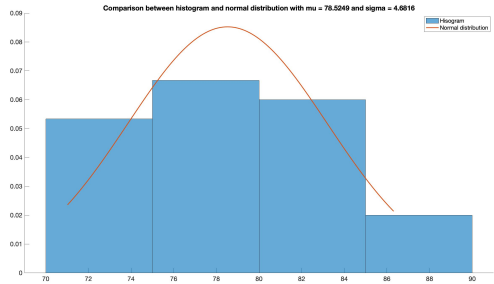


Figure 3.7: Data histogram and probability density function visualization for the second arena for lap time.

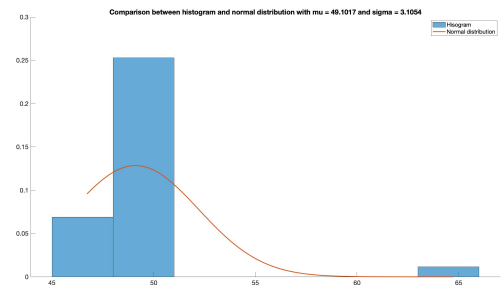
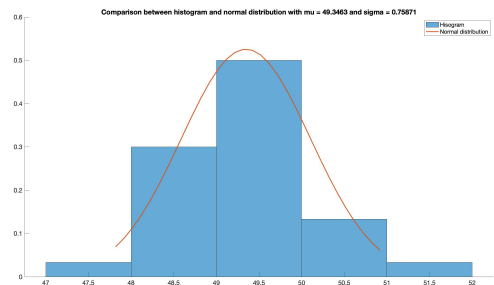


Figure 3.8: Data histogram and probability density function visualization for the second arena for distance travelled.

All the graphs have their comparison with a probability density function obtained with the μ and σ computed; the function is drawn following:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

where:

- $f(x)$: probability density function;
- μ : mean value;
- σ : standard deviation.

3.0.3 Lilliefors test

Since we wanted to know if the data acquired belong to a normal distribution, we applied a normality test based on the Kolmogorov-Smirnov.

The *Lilliefors test* is aimed at proving the null hypothesis that the sample belongs to a normal distribution.

However, without specifying which normal distribution; i.e., it does not specify the expected value and variance of the distribution.

Null hypothesis H_0

The null hypothesis for this test is that all the data are from a normal distribution, however, without specifying which normal distribution; i.e. it does not specify the expected value and variance of the distribution.

The function used is *lillietest()* and it returns two values as output:

- 1: if the null hypothesis has to be rejected (the data do not belong to a normal distribution);
- 0: if the null hypothesis has not to be rejected (the data belong to a normal distribution).

With the data acquired we obtained:

Normal distribution: H_0 accepted	Lap time assignment controller (Arena 1)
	Lap time robot-sim controller (Arena 1)
	Lap time assignment controller (Arena 2)
	Lap time robot-sim controller (Arena 2)
	Distance travelled assignment controller (Arena 2)
Not normal distribution: H_0 rejected	Distance travelled assignment controller (Arena 1)
	Distance travelled robot-sim controller (Arena 1)
	Distance travelled robot-sim controller (Arena 2)

Chapter 4

Conclusion

We can say that the majority of the data acquired belong to a normal distribution according to the Lilliefors test.

It is important to know that during the simulation and the acquiring data process the PC may not be totally dedicated to it and, according to its RAM and CPU usage there can be few errors: sometimes we had for example to stop the simulation and start it again because the robot hit a wall or turned too much around its z -axis thus bringing errors in simulation and in the data.

The test can be expanded by introducing more statistical analysis for example the number of time the robot hits a wall or the number of time the robot chooses a wrong direction with respect to the total number of laps analysed.

Thanks to this test we can state which controller resulted to be better. Of course we can say it only with respect to the lap time, quite all the data recorded for the distance travelled resulted to be not a normal distribution.

Hypothesis verification

Since we started from a null hypothesis of the same greatness for both the controllers, in which we stated $\mu_{assignment} = \mu_{robot-sim}$, we can now check if it was correct.

We can check the standard deviation σ we obtained and, according to H_0 they should be the same.

We can see from the results obtained that we can just check the greatness of a controller from the lap time simulation because its data belong to a normal distribution.

One stated that, we can compare the two mean values μ obtained for both controllers and we can clearly see that:

$$\mu_{assignment} < \mu_{robot-sim}$$

Thanks to this, we can reject our null hypothesis H_0 and we can accept all the alternative hypothesis H_a , which include also our result.

At the end we can conclude that, in term of lap time, the assignment controller performs better with respect to the robot-sim one.