UNIVERSITÀ DEGLI STUDI DI GENOVA

DIBRIS

DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY,
BIOENGINEERING, ROBOTICS AND SYSTEM ENGINEERING

RESEARCH TRACK 2

Third Assignment

Statistical Analysis

Author:

HEBIK Younes

Student ID:

s5813030

Professors:

Carmine Tommaso Recchiuto

May 31, 2024

1 Introduction

Statistical analysis is a powerful tool for evaluating the performance of algorithms in various domains. It encompasses a wide range of methods, including descriptive statistics to summarize and describe data, inferential statistics to make inferences or predictions about a population based on sample data, hypothesis testing to evaluate the significance of relationships or differences, regression analysis to model relationships between variables, and many other techniques. In this report, I present a statistical analysis of two different algorithm implementations for the same task (Research Track1 first assignment), one developed by colleague Aymen Boullala and the other is my implementation. The goal is to perform a statistical analysis by performing different tests to test the hypothesis which will be presented later

2 Experiment Description

Data was collected by running both code implementations in different environments, resulting in multiple performance measurements for each implementation. The collected data was then subjected to statistical analysis to draw meaningful conclusions.

2.1 Used Tools

I worked with the code(algorithm) of the first assignment of RT1, focusing and comparing two implementations:

- My Implementation: https://github.com/younes-hebik/RT01-1st-Assignment
- Boullala's Implementation: https://github.com/aymenbll/ResearchTrackAssignment1$_2$023

The goal of the first RT1 assignment was to develop a program that locates golden boxes, collects them, and places them together in a single location.

In order to carry out a good and reliable statistical analysis, I modified the code such that I can retrieve some parameters (like execution time). I used also Jupyter notebook for developing graphs that were helpful in the analysis.

2.2 Arena Configuration

In order to make different configurations of the placement of the boxes I modified the code to make the boxes randomly placed and I variate the number of boxes. At the end I ended up with the following Arena Configurations :

- 4 Boxes
- 5 Boxes
- 6 Boxes
- 7 Boxes

The position of the boxes is random and it is not in a circle shape as it was in the standard arena, meaning that each time the boxes were added in random positions. The configurations used are shown bellow :

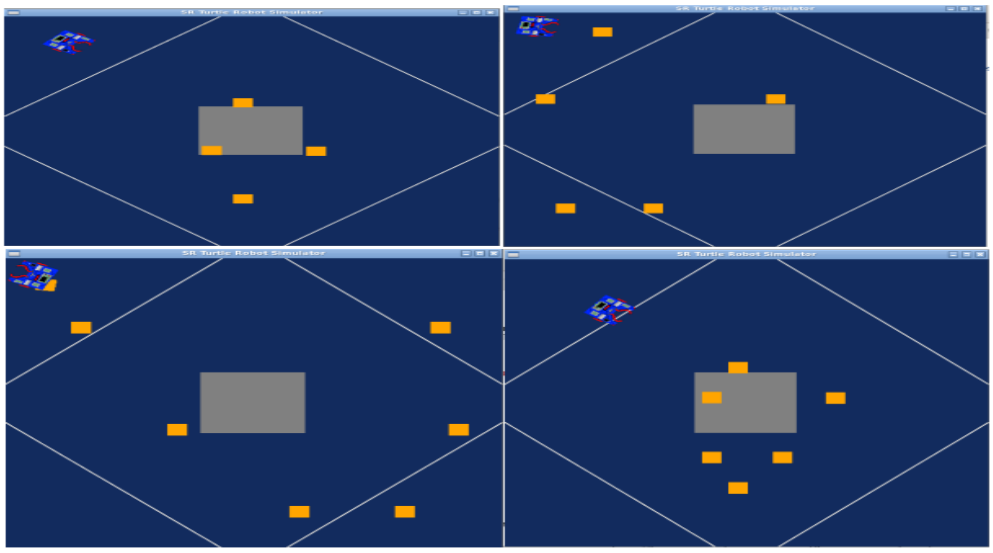

Figure 1: the four Arena's Configurations

2.3 Data Collection

For each code I performed the following experiments:

- For all the configurations (4,5,6,7 tokens) I performed 8 attempts for both codes (my code and my colleague's code) and I recorded the time spent by both algorithms to complete the task.

After trying the algorithms, I calculated the average time requires for all the environment configurations, so that I can analyze and compare the performance and to find out if the difference is significant. The use of different arenas configurations ensures that the statistical analysis covers the different possible performances.

3 Hypothesis

As the null hypothesis, I assume that there is no significant difference in the execution time between the two codes, suggesting that both codes have similar performance. However, the alternative hypothesis challenges this assumption and proposes that there is indeed a significant difference in the execution time. I hypothesize that there is a significant difference in the execution time between my code and my colleague's code. Specifically, I believe that my colleague's implementation performs faster than mine in terms of execution time. To test this hypothesis, I collected data by measuring the execution time for the same tasks using both implementations. Statistical methods, such as a t-test, will be employed to analyze the data and determine if there is a statistically significant difference in the execution time. By evaluating the results, I aim to gain insights into the relative performance of the two codes and understand which hypothesis will be confirmed.

4 Experiment results

After collecting all the data regarding the execution time, all the averages were made for each configuration. I decide to make a comparison between the mean execution time of both algorithms in different configurations. Using jupyter Notbook, I got the following plots :
code 1: My code
code 2: My colleague's code

- Configuration1: 4 boxes Environment

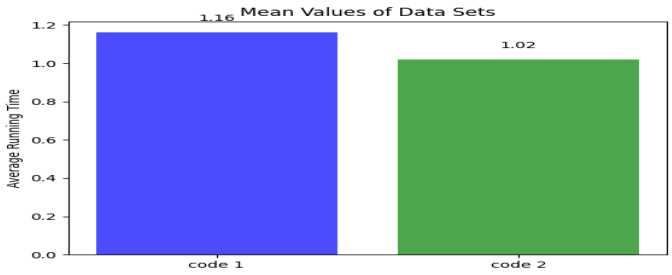

Figure 2: Average Running Time for 4 Tokens environment

- Configuration2: 5 boxes Environment

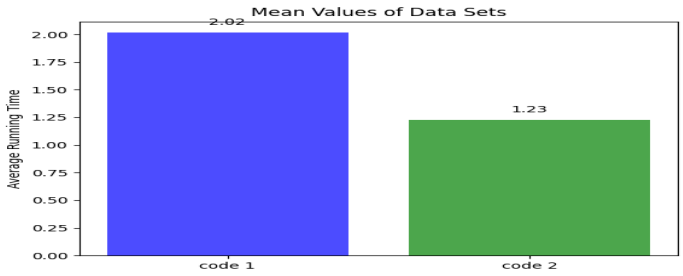

Figure 3: Average Running Time for 5 Tokens environment

- Configuration3: 6 boxes Environment

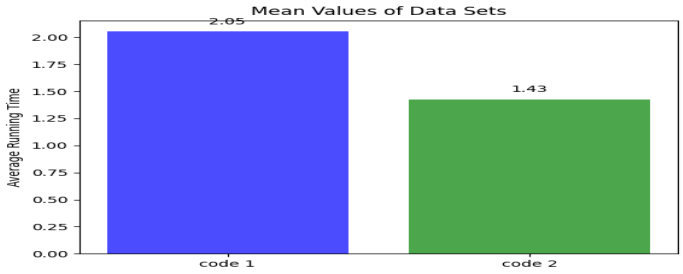

Figure 4: Average Running Time for 6 Tokens environment

- Configuration4: 7 boxes Environment

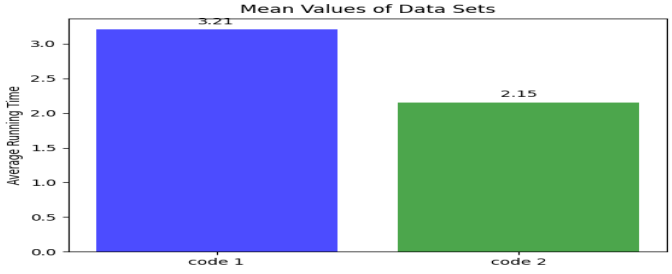

Figure 5: Average Running Time for 7 Tokens environment

In Figures 2, 3, 4 and 5, we can see two different representations of the average execution time for both algorithms in the environments presented earlier. Upon analyzing the bar charts and visualizing the overall mean results plot, it can be observed that My colleague's code consistently demonstrates lower execution times compared to my code. The bar charts indicate a clear trend where 2nd code exhibits faster performance across various configurations. This suggests that 2 nd code is optimized for faster execution, We can observe also regarding the relationship between the number of tokens and the difference in code implementation time. It is important to conduct a statistical analysis to further validate the Observations. A suitable approach would be to perform a statistical test, such as the t-test, to assess the significance of the observed differences in code implementation time.

5 Tests

In this section of the report, I will evaluate the first hypothesis by applying statistical tests to the sampled data. Since the distribution of the original population is unknown, I have collected a sufficiently large sample of 32 data points. This sample size allows us to assume that the sample data follows a normal distribution, enabling us to proceed with parametric analysis.

5.1 T Test

The T-test is a statistical method used for hypothesis testing to compare the means of one or two populations. It helps determine whether there is a significant difference between two groups or if a single group differs from a known value. To compare the performance of my code with my colleague's code in various environments, I employed a paired one-tailed t-test to identify which algorithm demonstrates a significantly different execution time.

Procedure

1. Gather two sets of related observations.
2. Calculate the Differences: For each pair of observations, calculate the difference.
3. Calculate the Mean and Standard Deviation of Differences
4. Calculate the Test Statistic (t) and find DOF
5. Determine the critical value and compare it with the previously calculated t-value.

the figure below shows the mean value of the difference

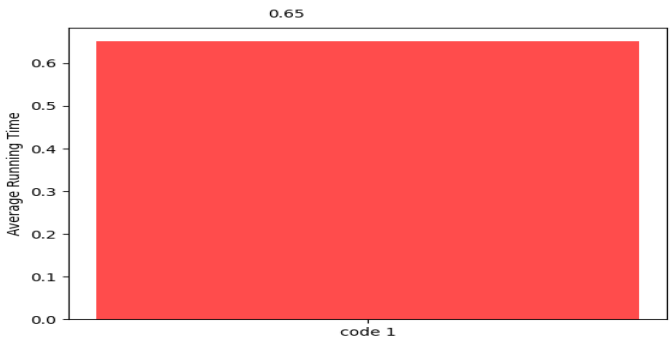

Figure 6: Mean of the difference

After calculating the differences between the two sample data sets and computing the mean and standard deviation, we obtained the following results:
Mean: 0.651
Standard Deviation: 0.814

$$S_E(d) = \frac{d}{\sqrt{n}} = \frac{0.651}{\sqrt{32}} = 0.14$$

$$T = \frac{d}{S_E(d)} = 4.53$$

$$DOF = n - 1 = 31$$

Now, we proceed to compare our calculated t-statistic with the values in the t-distribution table. For our analysis, we've opted for a confidence level of 99.5%. If our calculated parameter falls below the 5% significance level in the corresponding table, we fail to reject the null hypothesis. Conversely, if it exceeds this threshold, we can reject the null hypothesis.

t Static	DOF	t Table	P value
4.53	31	2.75	1.594e-10

In this case t static is greater than t table- value and P value is less than 0.05 than null Hypothesis can be rejected with confidence of 99.5%

6 Conclusion

In conclusion, the statistical analysis performed in this study has shed light on the comparison between my code and my colleague's code in terms of performance. The results have provided evidence to support or reject the null hypothesis and validate the alternative hypothesis. Regarding the null hypothesis, the t-test results have demonstrated that there are indeed significant differences between my code and my colleague's code in terms of execution time . In all configurations tested, the p-value was significantly below the chosen significance level, indicating a rejection of the null hypothesis. Therefore, the Null hypothesis (H0) was rejected. Conversely, the alternative hypothesis, which proposes a significant difference in the execution time, has been confirmed. The t-test results have indicated that my colleague's code exhibits faster execution times compared to my code in all configurations.