Report Research Track 2

Statistical analisys on the performances
of two first assignments

Matteo Forni

May 2023

1 Problem Specifications:

In this report, I am going to show the results of a statistical analysis that I performed on two first assignments of the course *Research Track 1*.
For reference, I compared my first assignment with the one made by my colleague Simone Borelli.

For this statistical analysis, I took into consideration two factors: the full time of execution and the ”research time”, which is the time it takes the robot to find all the tokens (both silver and golden) and it is calculated by taking the full time of execution calculated before and subtracting to it the total amount of time during which the robot is moving towards a token that it has already found.
The goal of this statistical analysis is to find out which one among these two ”token research algorithm” complete the task in less time, looking at both ”research time” and full execution time.
Another goal is to see if the ”research time” has some kind of relevant impact on the full time of execution.

2 Hypothesis:

As it was mentioned in the abstract, there are two null hypothesis to test.
The first one is:

- **Null Hypothesis:** The two algorithms have no relevant difference in the research time;
- **Alternative Hypothesis:** The two algorithms have some differences in terms of research time.

The second hypothesis is:

- **Null Hypothesis:** There is no difference between the full time of execution of the two algorithms;
- **Alternative Hypothesis:** There is a difference between the full time of execution of the two algorithms.

For both these hypothesis I have collected data from the two assignments and I performed a statistical test, to see if there was any relevant difference.
Moreover, for both tests I chose a level of significance equal to 5%.

3 Statistical analysis:

In order to perform a statistical analysis, I collected all the times from both algorithms: I executed 20 times each program and collected both the research time and the full execution time (and kept them paired for each execution).
These are the times that I collected for the two algorithms (for semplicity, I referred to my algorithm as M and to the other one as S):

Algorithm	Research time	Execution time
M	45:57	1:31:27
M	46:75	1:31:74
M	46:84	1:31:94
M	46:89	1:30:51
M	46:90	1:31:19
M	46:99	1:30:91
M	47:02	1:30:07
M	47:14	1:30:12
M	47:15	1:30:58
M	47:27	1:31:13
M	47:41	1:30:77
M	47:48	1:30:55
M	47:49	1:31:99
M	47:50	1:31:06
M	47:54	1:30:57
M	47:59	1:31:91
M	47:70	1:31:31
M	47:82	1:29:74
M	47:83	1:31:10
M	47:91	1:30:87
S	35:55	1:36:61
S	36:92	1:36:43
S	45:67	1:36:20
S	51:34	1:50:31
S	52:13	1:49:73
S	53:10	1:52:94
S	53:60	1:49:52
S	54:31	1:54:18
S	54:90	1:53:06
S	55:48	1:51:72
S	55:56	1:50:79
S	55:70	1:52:96
S	55:83	1:53:07
S	55:92	1:54:15
S	56:06	1:51:09
S	56:13	1:52:64
S	56:56	1:54:09
S	57:01	1:57:01
S	57:03	1:55:73
S	57:15	1:54:03

Then, to see if there is any significant difference among the performances of the two algorithms, I performed two U-tests on the research time and on the full execution time.
The U-test is a non-parametric test, which can be used to determine if the two population from which the data is collected are equal (which, in our case, means that the two algorithms perform the same way in terms of research time and execution time).
To perform the first test, I gathered the research times from both algorithms and ordered them in an increasing order.
Then, I assigned a rank to each time, starting from the lowest one (rank = 1) and increasing the rank for every new time in the ordered list.
Here are the tables of all the ranks I assigned to for the first U-test.

Algorithm	Research time	Rank
S	35:55	1
S	36:92	2
M	45:57	3
S	45:67	4
M	46:75	5
M	46:84	6
M	46:89	7
M	46:90	8
M	46:99	9
M	47:02	10
M	47:14	11
M	47:15	12
M	47:27	13
M	47:41	14
M	47:48	15
M	47:49	16
M	47:50	17
M	47:54	18
M	47:59	19
M	47:70	20
M	47:82	21
M	47:83	22
M	47:91	23
S	51:34	24
S	52:13	25
S	53:10	26
S	53:60	27
S	54:31	28
S	54:90	29
S	55:48	30
S	55:56	31
S	55:70	32
S	55:83	33
S	55:92	34
S	56:06	35
S	56:13	36
S	56:56	37
S	57:01	38
S	57:03	39
S	57:15	40

Then, I computed the rank sums of the two algorithms, which are respectively the sum of all the ranks assigned to a time belonging to the first algorithm and the sum of all the ranks assigned to a time belonging to the second algorithm.
The values I obtained are the following ones:

$$RM = 269$$
$$RS = 551$$

After that, I calculated the U-value for both algorithms:

$$UM = n1 * n2 + \frac{n1*(n1+1)}{2} - RM$$
$$US = n1 * n2 + \frac{n2*(n2+1)}{2} - RS$$

Then, I selected the U-value to use in the test by choosing the lowest value among the two U-values of the two algorithms.

$$U = min(UM, US)$$

The values that I obtained are the following ones:

$$UM = 20 * 20 + \frac{20*(20+1)}{2} - 269 = 341$$
$$US = 20 * 20 + \frac{20*(20+1)}{2} - 551 = 59$$

Therefore, I obtained a U-value for the research time equal to 59 (the lower of the two U-values computed).
Then, I compared the obtained values to the value inside the table of critical values for the U-test, considering that n1 = n2 = 20 and the level of significance is 5%.
The table I used is shown below:

n1\n2	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2							0	0	0	0	1	1	1	1	1	2	2	2	2
3				0	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8
4			0	1	2	3	4	4	5	6	7	8	9	10	11	11	12	13	14
5		0	1	2	3	5	6	7	8	9	11	12	13	14	15	17	18	19	20
6		1	2	3	5	6	8	10	11	13	14	16	17	19	21	22	24	25	27
7		1	3	5	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34
8	0	2	4	6	8	10	13	15	17	19	22	24	26	29	31	34	36	38	41
9	0	2	4	7	10	12	15	17	20	23	26	28	31	34	37	39	42	45	48
10	0	3	5	8	11	14	17	20	23	26	29	33	36	39	42	45	48	52	55
11	0	3	6	9	13	16	19	23	26	30	33	37	40	44	47	51	55	58	62
12	1	4	7	11	14	18	22	26	29	33	37	41	45	49	53	57	61	65	69
13	1	4	8	12	16	20	24	28	33	37	41	45	50	54	59	63	67	72	76
14	1	5	9	13	17	22	26	31	36	40	45	50	55	59	64	69	74	78	83
15	1	5	10	14	19	24	29	34	39	44	49	54	59	64	70	75	80	85	90
16	1	6	11	15	21	26	31	37	42	47	53	59	64	70	75	81	86	92	98
17	2	6	11	17	22	28	34	39	45	51	57	63	69	75	81	87	93	99	105
18	2	7	12	18	24	30	36	42	48	55	61	67	74	80	86	93	99	106	112
19	2	7	13	19	25	32	38	45	52	58	65	72	78	85	92	99	106	113	119
20	2	8	14	20	27	34	41	48	55	62	69	76	83	90	98	105	112	119	127

As we can see, the value I obtained is inferior to the correspondent value on the table, which means that there is a significant difference among the two group of times; hence, the first null hypothesis can be rejected.
Then, I performed the same analysis on the full execution times.
Again, here is the table with the ranks:

Algorithm	Research time	Rank
M	1:29:74	1
M	1:30:07	2
M	1:30:12	3
M	1:30:51	4
M	1:30:55	5
M	1:30:57	6
M	1:30:58	7
M	1:30:77	8
M	1:30:87	9
M	1:30:91	10
M	1:30:99	11
M	1:31:06	12
M	1:31:10	13
M	1:31:13	14
M	1:31:19	15
M	1:31:27	16
M	1:31:31	17
M	1:31:74	18
M	1:31:94	19
M	1:31:99	20
S	1:36:20	21
S	1:36:43	22
S	1:36:61	23
S	1:49:52	24
S	1:49:73	25
S	1:50:31	26
S	1:50:79	27
S	1:51:09	28
S	1:51:72	29
S	1:52:64	30
S	1:52:94	31
S	1:52:96	32
S	1:53:06	33
S	1:53:07	34
S	1:54:03	35
S	1:54:09	36
S	1:54:15	37
S	1:54:18	38
S	1:55:73	39
S	1:57:01	40

Then, I calculated the rank sums:

$$RM = 210$$
$$RS = 610$$

And the U-values:

$$UM = 20 * 20 + \frac{20*(20+1)}{2} - 210 = 400$$
$$US = 20 * 20 + \frac{20*(20+1)}{2} - 610 = 0$$

This time, I obtained a U-value equal to 0 (which means that the higher time of the first algorithm is inferior to the lower time of the second algorithm).
Again, the U-value obtained is clearly inferior to the value inside the table, hence the second null hypothesis can also be rejected.

4 Conclusions:

What are the conclusions that can be drawn from this experiment? We saw that, in both cases, the null hypothesis has been rejected, which means that there is a significant difference in the performances of the two algorithms in both the research time and the full execution time.
So we can state that the algorithm M performs better than algorithm S both in terms of research time and full execution time.
These results may also suggest a correlation between the research time and the full execution time, since the algorithm with the most performing research time, was the one that performed better in the full execution time.
So it can be that the overall performance in terms of time of an algorithm for this program can be improved focusing on reducing the research time.
In fact, while it is easily possible to reduce the non-research time by simply finding the optimal robot velocity, improving the research time could be more complicated, but crucial as well in order to achieve better performances.