T-test on different implementations of exploration algorithm

Aurora Durante

May 23, 2022

Contents

1 Introduction

By performing a statistical analysis, it is possible to prove right or wrong a statement based on the result of the test. There exist different types of analysis, among which the *t-test* is applied whenever the population has an unknown standard deviation and it is normally distributed.
Another characteristic of the t-test is that it can be performed as a *two-tailed* or a *one-tailed* one based on how the statement we want to prove and the relative hypothesis are defined. In this case, a one-tailed t-test is performed.
At last, even if the distribution of the population from which observations of the data are retrieved is unknown, the *central limit theorem* says that whenever the samples size is grater than 30, then the distribution approaches a normal one. Thus, applying the t-test is possible.
This been said, a formal statement is made and the parameters of the test defined in order to perform the analysis on two different implementations of an exploration algorithm.

2 Method

Every step of the test can be found in the related MatLab file present in this GitHub repository and data are retrieved from this two implementation:

- `https://github.com/AuroraD-Hub/RT1_I-assignment.git`, this is my implementation;
- `https://github.com/CarmineD8/python_simulator/tree/rt2`, this is the reference implementation.

I want to test that my implementation needs less time to complete a lap in the circuit with respect to the reference one. So, I define the null hypothesis $\mathtt{H}_0$ and the alternative one $\mathtt{H}_a$ as follows and where μ represents the mean of the population data from my implementation and μ_r the mean of the other one:

$$\mathtt{H}_0\text{: } \mu = \mu_r$$
$$\mathtt{H}_a\text{: } \mu < \mu_r$$

Then, other parameters are defined, such as the *level of significance* at 5% and the choice of performing a one-tailed t-test.

The data are obtained by timing the robot in the circuit, where the starting position is in top left corner as the ending one. Timing of my implementation are stored in vector `data`, whereas `data_ref` contains the timing of the reference implementation. A simple *t-test* is then performed by computing the following value of t:

$$t = \frac{m - m_r}{\sigma}$$

In this equation, $\mathtt{m}$ and $\mathtt{m}_r$ are the mean values of the vector data and σ is obtained in the following way:

1. define the standard deviation values of the data (`s` and $\mathtt{s}_r$) with *std* MatLab command and the number of samples in them (`N` and $\mathtt{N}_r$) with *length* command;
2. define the square of the *pooled standard deviation* as:

$$\sigma_p^2 = \frac{(N-1)s^2 + (N_r-1)s_r^2}{N + N_r - 2}$$

3. finally obtain

$$\sigma = \sqrt{\sigma_p^2(\frac{1}{N} + \frac{1}{N_r})}$$

Once the *t value* is calculated, I compared it with the appropriate value in the table considering α=5%, DoF=60 and one-tailed test to define whether H_0 should be accepted. Note that the t value to compare is chosen accordingly to the nearest DoF available in the table to the test one ($\mathtt{N+N}_r\mathtt{-2=58}$) and, since H_a is in the form "less than", the critical region of interest is the left tail of t-distribution and this means that **-t** $<$ $\mathbf{-t}_{table}$ should be valid in order to reject H_0.
To furthermore prove the reliability of this test, I also executed the native t-test function of MatLab (`ttest2`) and a *paired t-test* that can be also found in the MatLab script. Moreover, I also obtained the *p-value* with the first one.

3 Results

In this section, I present the results of the test.
Considering $\mathtt{t}_{table}$ `= -1,671`, in the simple t-test I obtained `t = -16,07` and, thus, I can reject H_0. This is also proved by the native MatLab function and the paired t-test, which also reject the null hypothesis. Moreover, the function `ttest2` calculates the p-value of the test and in this case it results being `p = 4,98*10`$^{-23}$.
From this results I can conclude that the two set of data belongs to two different distribution with an uncertainty defined by the p-value. Furthermore, this is also evident by looking at the data since actually it can be seen that my implementation is on average 1 minute slower than the reference one. In fact, the calculated t value is big, hinting to a wide difference of the means.