Part 6 - Statistical Significance: Hypothesis Testing for Forward A vs. Adaptive A**

To determine whether the observed differences in performance metrics between Forward A* and Adaptive A* are statistically significant or due to random variation, we will perform a paired t-test and compute the p-value.

1. **Defining the Hypothesis**
    - **Null Hypothesis (H_0)**: There is no significant difference in the performance of Forward A* and Adaptive A*. The observed differences in expanded nodes, runtime, or path length are due to random variation.
    - **Alternative Hypothesis (H_1)**: There is a statistically significant difference in the performance of the two algorithms, meaning the observed differences are not due to random variation.

    This test will determine whether one algorithm systematically outperforms the other.

2. **Choosing the Statistical Test: Paired t-test**

    Since each maze was tested with both Forward A* and Adaptive A*, the data is paired. The correct test is a paired t-test, which measures whether the mean difference between paired samples is statistically significant.

    We will perform the test on three different performance metrics:

    1. Expanded Nodes (computational efficiency)
    2. Path Length (solution quality)
    3. Runtime (execution speed)

    The paired t-test is appropriate if the differences are normally distributed; otherwise, we might need to use a non-parametric test like the Wilcoxon signed-rank test (not required here since we are focusing on t-tests).

3. Steps to Perform the Paired t-test

Step 1: Collect Data

- X_i = Performance of Forward A* on maze i (expanded nodes, runtime, or path length).
- Y_i = Performance of Adaptive A* on maze i.
- Compute the difference: $D_i = Y_i - X_i$

This represents how much Adaptive A* differs from Forward A* on each maze.

Step 2: Compute the Mean and Standard Deviation of Differences

1. Mean difference: Đ = $(\Sigma D_i)/n$ ← Đ means D-bar

where $n = 50$ (the number of mazes tested).

2. Standard deviation of differences: $S_D = (\Sigma(D-Đ)^2 / (n - 1))^{(1/2)}$

Step 3: Compute the t-statistic

The t-statistic measures how significant the difference is: $t = Đ / (S_D / n^{(1/2)})$

Where:

- D = Mean difference
- S_D= Standard deviation of differences
- $n = 50$ (number of test cases)

A larger $|t|$ value means a stronger difference between the two algorithms.

Step 4: Compute the p-value

The p-value tells us how likely it is that the observed difference happened by chance.

- If $p < 0.05$, we reject H_0 and conclude that there is a statistically significant difference between Forward A* and Adaptive A*.
- If $p \geq 0.05$, we fail to reject H_0, meaning the difference could be due to random noise, and we do not have enough evidence to claim one algorithm is systematically better.

We compare our computed t-value to the critical t-value from a t-distribution table at 49 degrees of freedom (n-1)to determine if the result is statistically significant.

4. Interpretation of Results

1. Expanded Nodes:
    - If $p < 0.05$, Adaptive A* expands significantly more (or fewer) nodes than Forward A*.
    - If $p \geq 0.05$, the difference in expanded nodes is not statistically significant.
2. Path Length:
    - If $p < 0.05$, one algorithm finds significantly longer or shorter paths.
    - If $p \geq 0.05$, the path length difference is likely due to random noise.
3. Runtime:
    - If $p < 0.05$, there is a significant runtime difference.
    - If $p \geq 0.05$, the runtime difference is not significant.

If all p-values are high (above 0.05), the differences in performance between Forward A* and Adaptive A* are likely due to random variation in the mazes rather than a systematic difference.

Conclusion

To determine whether Forward A* or Adaptive A* is truly better, we perform a paired t-test on the expanded nodes, path length, and runtime. The p-value will tell us whether the observed differences are statistically significant.

If $p < 0.05$, we conclude that one algorithm systematically outperforms the other. Otherwise, any performance differences are likely due to random variation in the test cases rather than a fundamental advantage of one algorithm over the other.