

Question 3: Interpolation Search

Questions:

What are some of the key aspects that makes Interpolation search better than Binary search?

Interpolation search is quicker than binary search, as binary search will always search a set of data by accessing the midpoint of it, while interpolation search is dynamic and can change this access point based on the value being searched. Interpolation also has a better average time complexity of $O(\log(\log(n)))$, compared to binary search with an average time complexity of $O(\log(n))$

An underlying assumption of the interpolation search, is that sorted data are uniformly distributed. What happens if the data follows a different distribution something like normal? Will the performance be affected? Discuss why (whether yes or no)

If a dataset's distribution differs from a uniform distribution, the performance of interpolation search will be negatively affected. A non-uniform distribution may result in inaccurate estimates of the position of target values, and consequently, will require more steps to find the values. With a normal distribution, the extent to which performance is degraded will likely depend on the skew of the data. If there's no large skew, the performance may be hindered slightly, but it will still perform well. In cases of highly-skewed data, the impact on performance will be much greater as the estimates of the position of target values will be much less precise.

If we want to modify the Interpolation Search to follow a different distribution, Which part of the code will be affected?

In the given implementation of interpolation search, the calculation of pos would be affected. Ideally, one would take into account the cumulative distribution function (cdf) of a distribution and use that information to find a more precise range where the target value would lie.

When comparing: linear, Binary, and Interpolation Sort

a. When is Linear Search your only option for searching your data as Binary and Interpolation might fail?

b. What is a case that Linear search will outperform both Interpolation and Binary search, and Why? Is there a way to improve Binary and Interpolation Sort to resolve this issue?

a) This case occurs when the data given is not sorted in any meaningful way and does not follow any distribution. In this case, interpolation and binary sort will not be able to predict the position of target values; therefore, they cannot be used.

b) A case where this situation may occur is in small sets of data. In these cases, binary and interpolation search are computationally taxing, as their complexity is not necessary for such small data sets. The simplicity of linear search is favourable. The only way to improve the performance of interpolation and binary search in these cases is to reduce the number of operations done by them. In already optimized binary and interpolation search algorithms, reducing the number of operations done is likely not possible while maintaining the functionality of the algorithms.