INTRODUCTION TO THE RESEARCH

Even though greedy has the worst case out of all the possible matching algorithms in this study it focuses on figuring out why it is the most effective in real world capabilities. For instance, companies like Uber/Doordash use greedy algorithm to match the server (driver) to the user (person requesting). The idea of greedy is essentially matching the newest user to the unmatched server. In experimental studies there are real-time taxi with 10000 users and servers and results are surprising cause even though its worst-case greedy beats out the other matching algorithms like HST and is almost on par with offline matching (most effective however can’t be applied in real-world due to it always being online). Therefore, the goal of my project is to test why through programming test trials in python that will allow me to use randomized data

METHODS

The methods used for this study is python testing where essentially I utilized the PyCharm IDE. Then through using python made it so that it generated random users/servers as coordinate points and then applied the greedy algorithm. After that there was an optimal algorithm in where it would find the optimal through methods defined, often looking at the minimum value present in the randomly generated coordinate points. Additionally, did it in a two-dimensional scale which is more realistic such and through utilizing the distance formula was able to get a methodology similar to how companies like Uber/Doordash have their set of users/servers and apply the algorithm

CONCLUSIONS

Therefore, the conclusion proves that greedy algorithm is optimal for figuring out how to match servers/users despite having the worst run time theoretically. As noted in the sample the ratio of one and two dimensions are extremely close to the value of ~1 which means greedy is super close to being optimal and is great for real-world applications, the two-dimensional optimal output can be referenced below, and it shows this exact fact of how the optimal and greedy are both close to 1 when you take the ratio of them which means that they are the around the same proving that greedy is great in a real world scenario with receivers and servers being Uber drivers and those who call the Ubbers. Furthermore, this can be applied to numerous real-world applications where there are a server and receiver, and this research helps contribute to the fact that for data matching greedy algorithm is optimal in these situations.