Mr. Anderson's Data Modeling Service

Connor Szabo

CIS 2460

Friday November 29th, 2019

After extensive testing done, I have come to the conclusion that for the wait time to remain under 30 minutes per job, there can be a maximum of 12 users at a time. I created two box plots illustrating the outcomes of 100 executions of my design, each execution returning the average time a job had to wait in the queue to be processed for 1000 completed jobs.

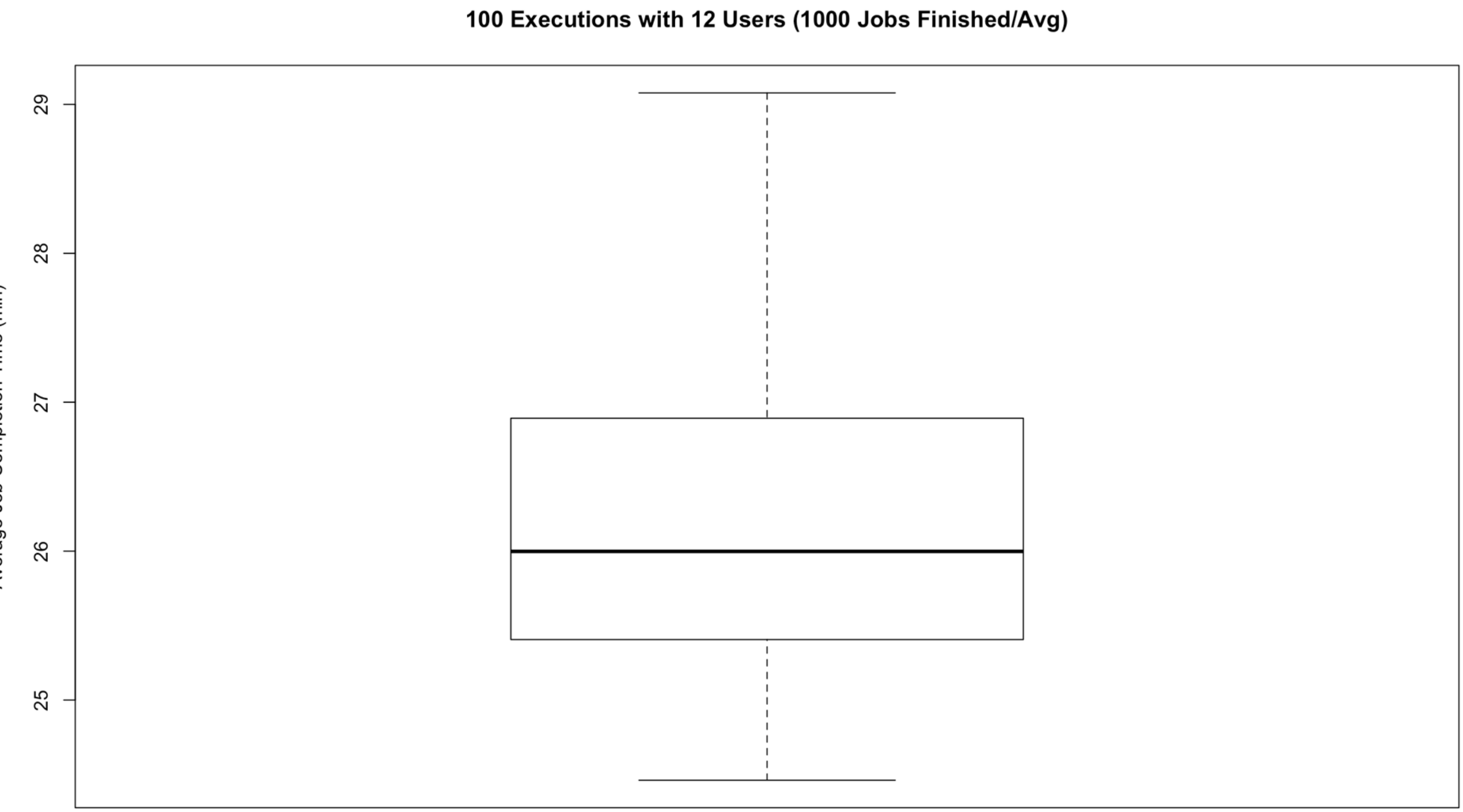

Figure 1. 1

Figure 1.1 above shows that with 12 users there are no data points or even outliers that fall beyond the 30-minute mark. The bulk of the data lies between roughly the 25.5-minute and 27-minute marks, no data being shown reaching or surpassing the 30-minute mark.

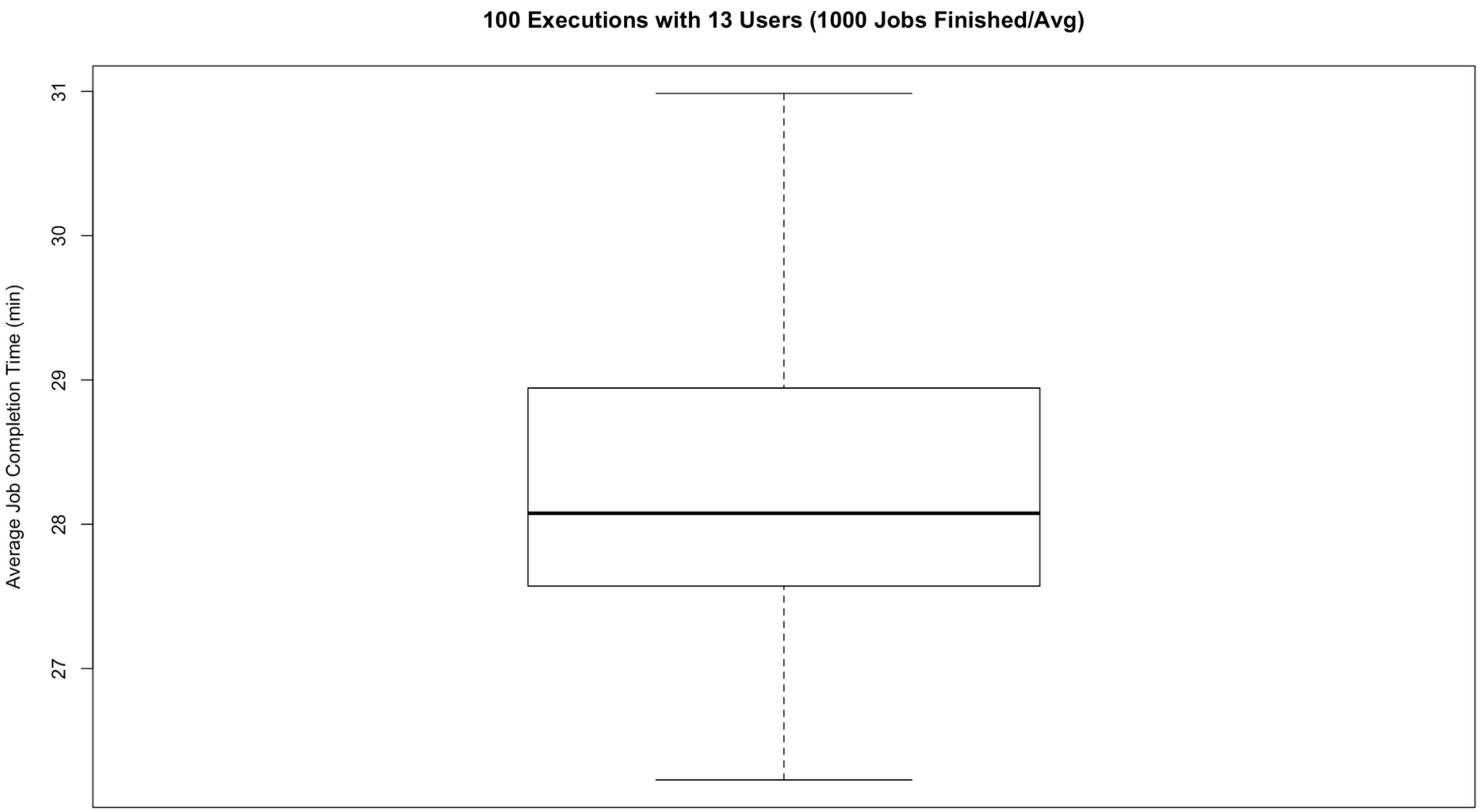

Figure 1. 2

Figure 1.2 shows the same things as the previous box plot, only this time with 13 users instead of 12. As you can see, the bulk of the data lies between roughly the 27.5 and 29-minute mark, however, there is still a significant amount of data that lies between the 29 and 31-minute mark. Mr. Anderson had requested a number of users so that the average wait time would not fall beyond the 30-minute mark, and as my previous box plots have shown 13 users does not satisfy that need, but 12 users does.

Testing out the average wait times for n= 10, 20, 40, and 80 came back as follows:

n = 10 had an average wait time of roughly 22 minutes

n = 20 had an average wait time of roughly 42 minutes

n = 40 had an average wait time of roughly 81 minutes

n = 80 had an average wait time of roughly 155 minutes

Through finding the average wait time of n = 20 to be over the set limit of maximum 30 minutes of wait time and finding that the average wait time for n = 10 was well below the set limit, it was clear to me that the n value would lie somewhere between the two. I tested for 15 next, being right between those two averages but that was still too high. I then split between the lower bound of n = 10 and the new upper bound of n = 15 to find that n = 12 satisfied the requirements. Just to be sure, I then tested n = 13 to see what the outcome was and found it broke the limit set by Mr. Anderson.

The following box plots (Figure 2.1 – 2.4) will illustrate how the change in the q value will change the average wait time. Based on the data represented in the graphs, it seems that the higher q values are more efficient when it comes to wait times, a q value of 0.1 having the longest wait times and a q value of 2 having the shortest wait times. Therefore, I believe the best value to assign to q would be 2. This would allow for more of each job to be completed and wastes less time with transfers that would happen more often with the lower q value.

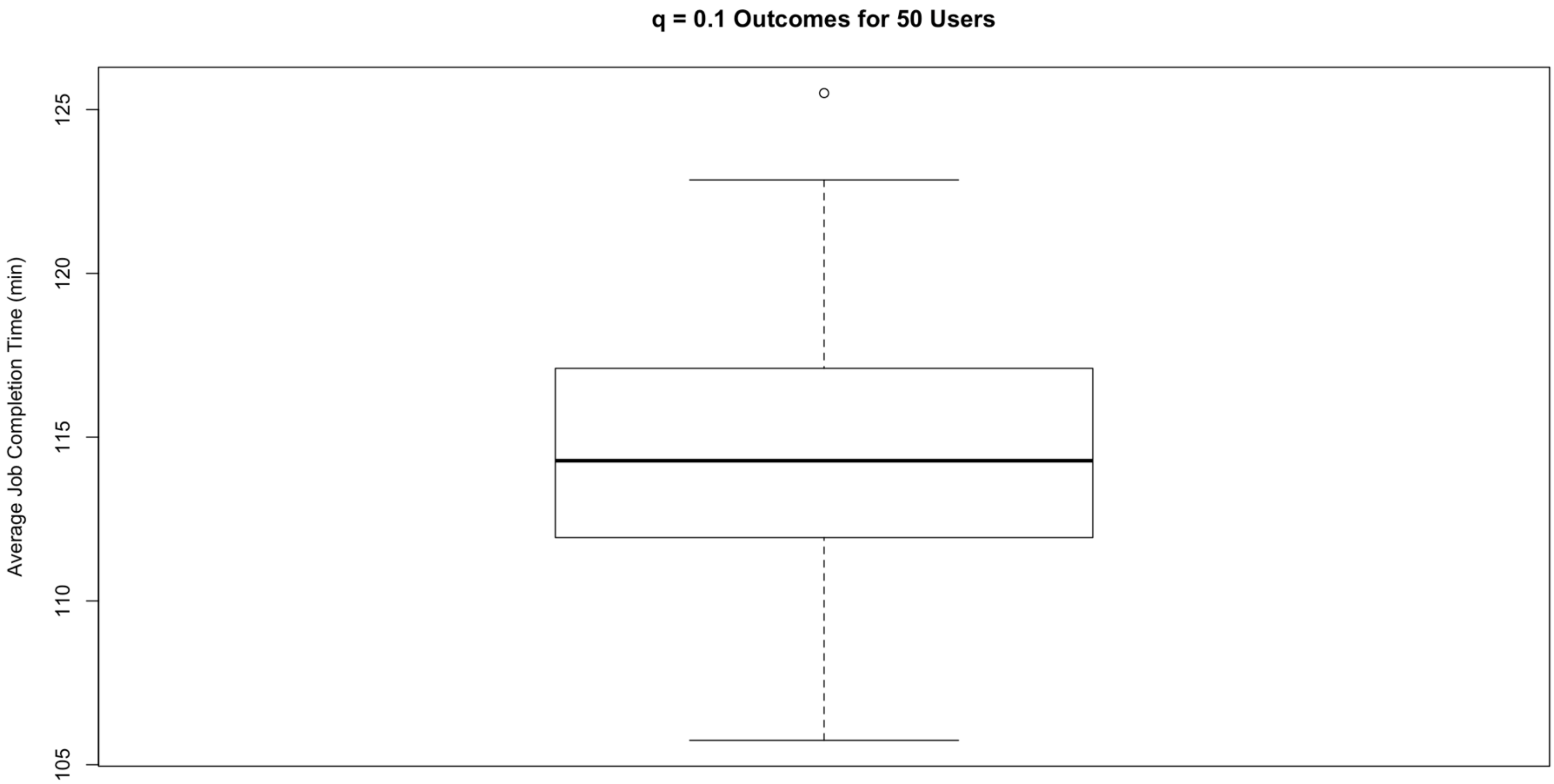

Figure 2. 1

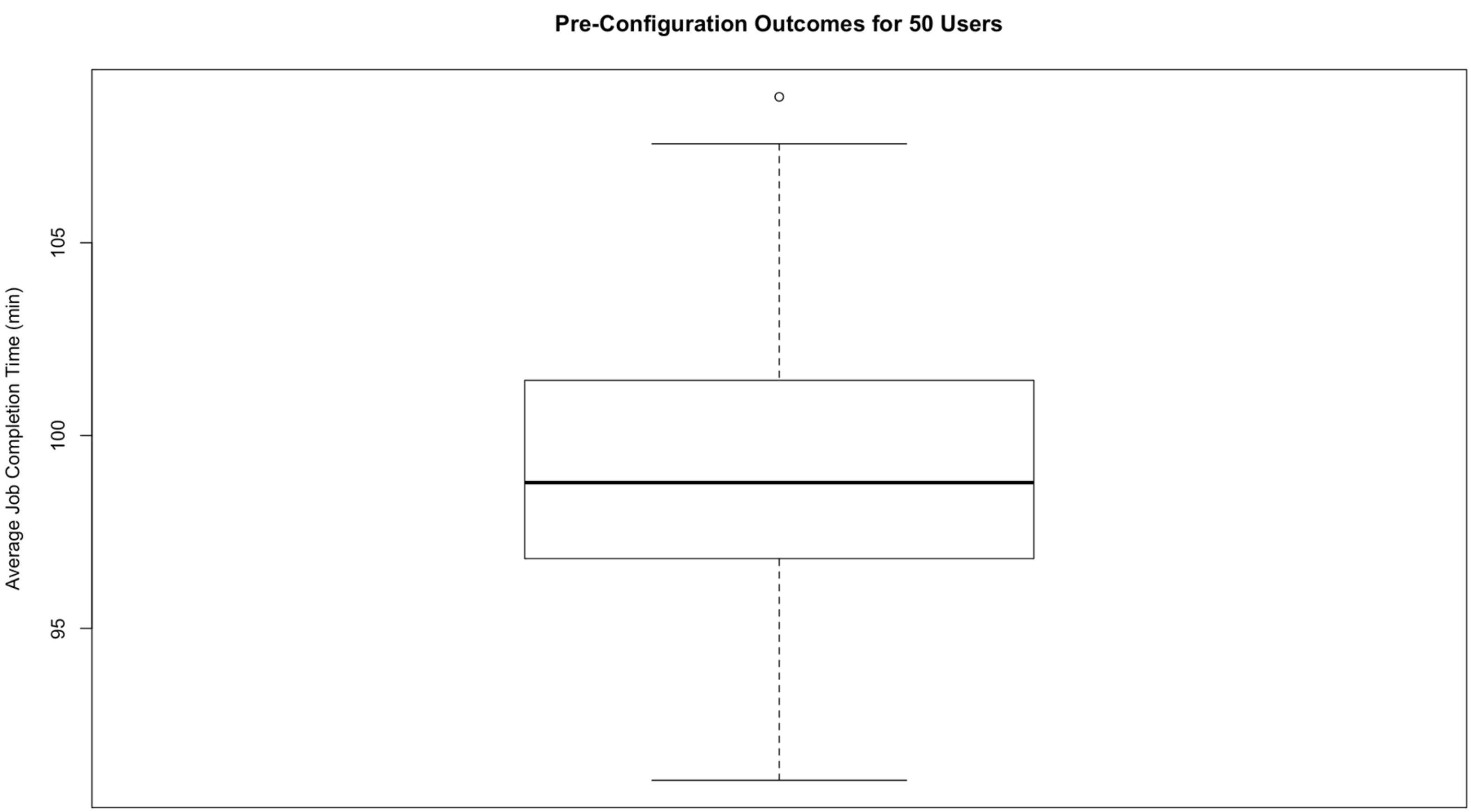

Figure 2. 2

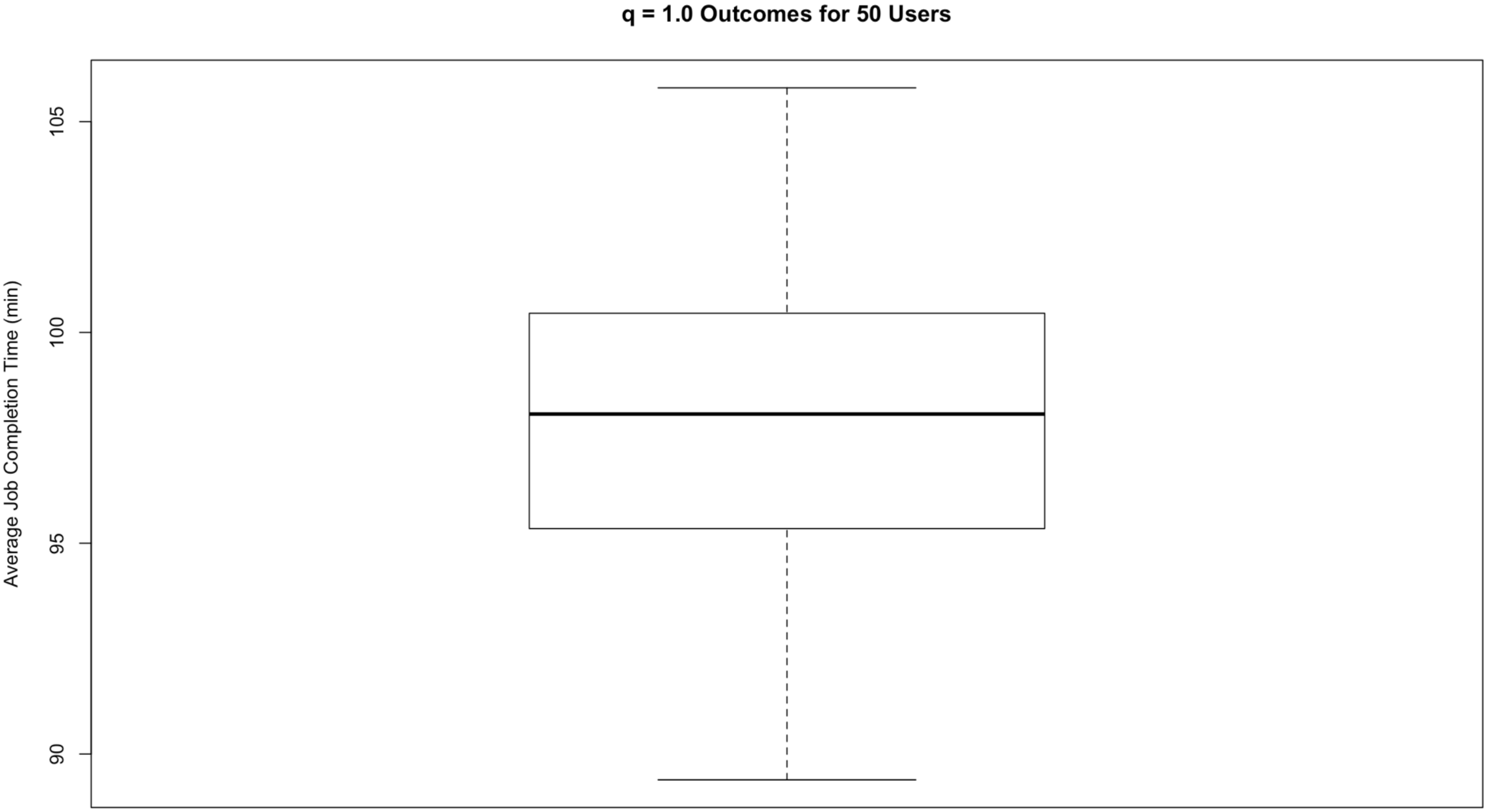

Figure 2. 3

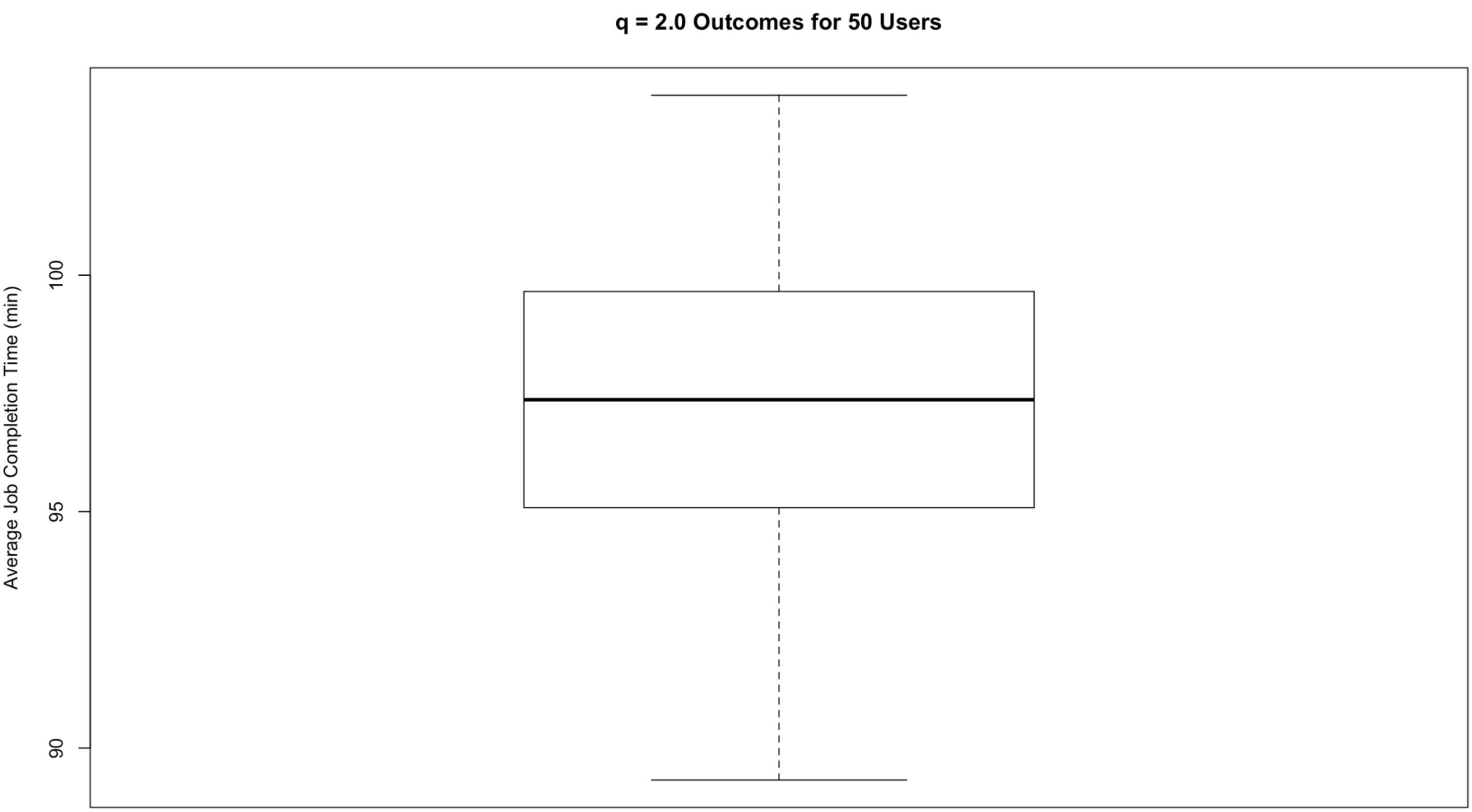

Figure 2. 4

The following box plots illustrate 100 average wait times for 1000 completed jobs.

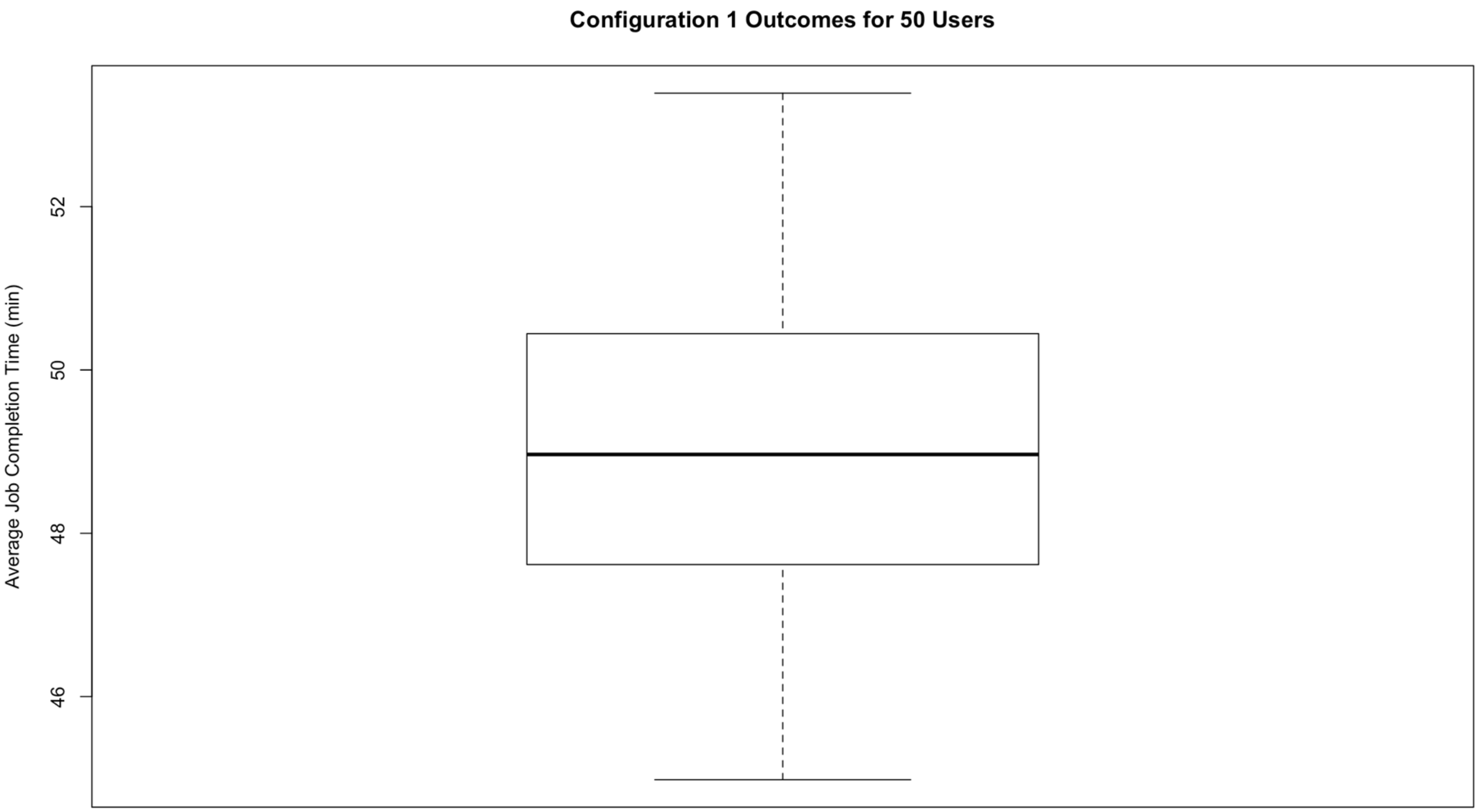

Figure 3. 1

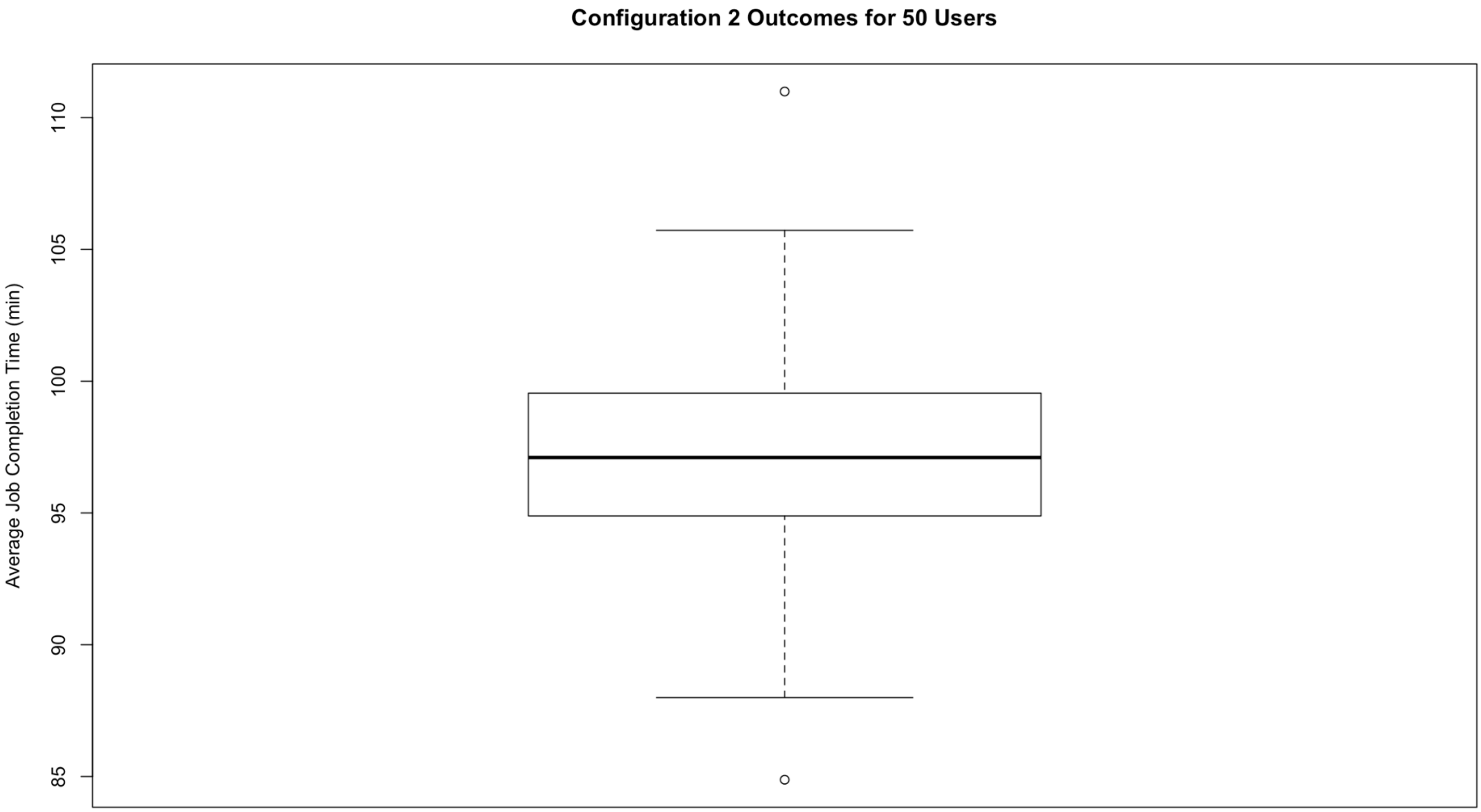

Figure 3. 2

Figures 3.1 and 3.2 show that Configuration one is obviously more efficient, having significantly lower wait times. However, we need to conclude whether or not it is cost effective. I will compare the cost of each to the improvements figures 3.1 and 3.2 show against figure 2.2.

Configuration One

- Cost: $5000
- Average Wait Time: 47.5 – 50.5 minutes for 50 Users

Configuration Two

- Cost: $1000

- Average Wait Time: 94.5 – 99 minutes for 50 Users

<u>Pre-Configuration</u>

- Cost: N/A
- Average Wait Time: 96.5 – 102 minutes for 50 Users

From the statistics shown, Configuration one has a wait time decrease of 49-51.5 minutes. This means that with a cost of $5000 that it would be a cost of between $97.09 and $102.04 to decrease the wait time by 1 minute.

Configuration Two has a wait time decrease of 2 – 3 minutes. At a cost of $1000, this means that the wait time would decrease by 1 minute for a cost of between $333.33 and $500.

With this evidence, I think it is clear that Configuration one is the more efficient choice, both from a cost and performance standpoint.