Predicting Triangles from Streamed Graphs

Implementing Triest Base and Triest Improved Algorithms

William Hibbard

My goal in this experiment was to visualize and understand the behaviors of two different algorithms in predicting the number of triangles in a graph as edges were streamed into the graph over time. The streamed graph had almost 400,000 edges. In my analysis, for each algorithm and allocated memory size, I ran twenty trials and graphed the maximum, minimum, mean, and first and third quartile from these trials at each time step. This means that I considered 8,000,000 data points in each graph. Following my analysis, you will find my ten graphs, with five figures in each of them. The graphs compare the estimated number of triangles in the graph versus the time spent streaming edges. They are labeled by the algorithm used, Base or Improved, denoted *Base* or *Impr* respectively, followed immediately after by the allocated memory size for the predictive algorithm.

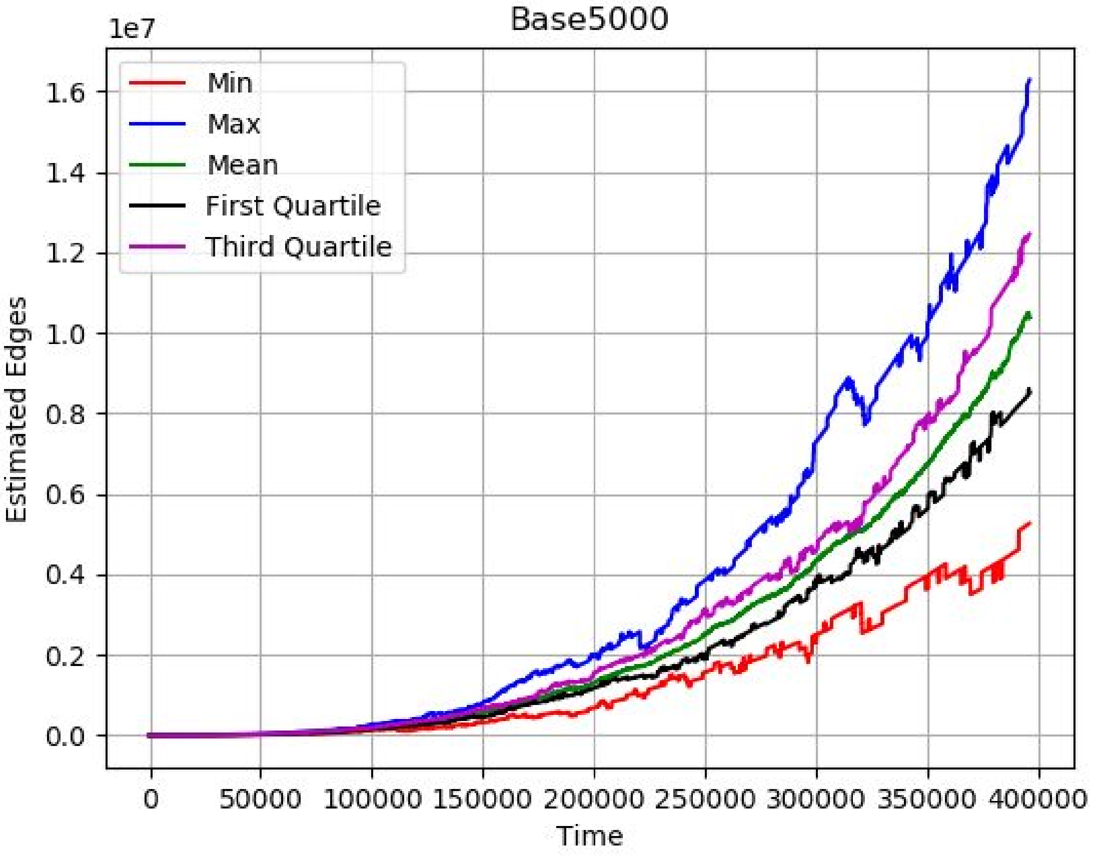
For the Triest Base algorithm, at a low sample size (5000) we see both a large amount of variation in the data as well as very rough looking curves. I will explain why this likely occurs following my analysis of the curves from both experiments. As the sample size increases to sizes of 10,000, 20,000, 30,000, and 40,000, we see the minimum, maximum, mean, and first and third quartile curves begin to tighten on each other as well as smoothen out. In other words, as the sample size increases, the prediction sees less variation between trials at every time step and the prediction begins to see less variation between each time step within each trial as well. By running trials of this algorithm with the sample size larger than the amount of edges added to the graph, the true number of triangles can be found. The Triest Base Algorithm seems to largely overpredict the number of triangles in the graph.

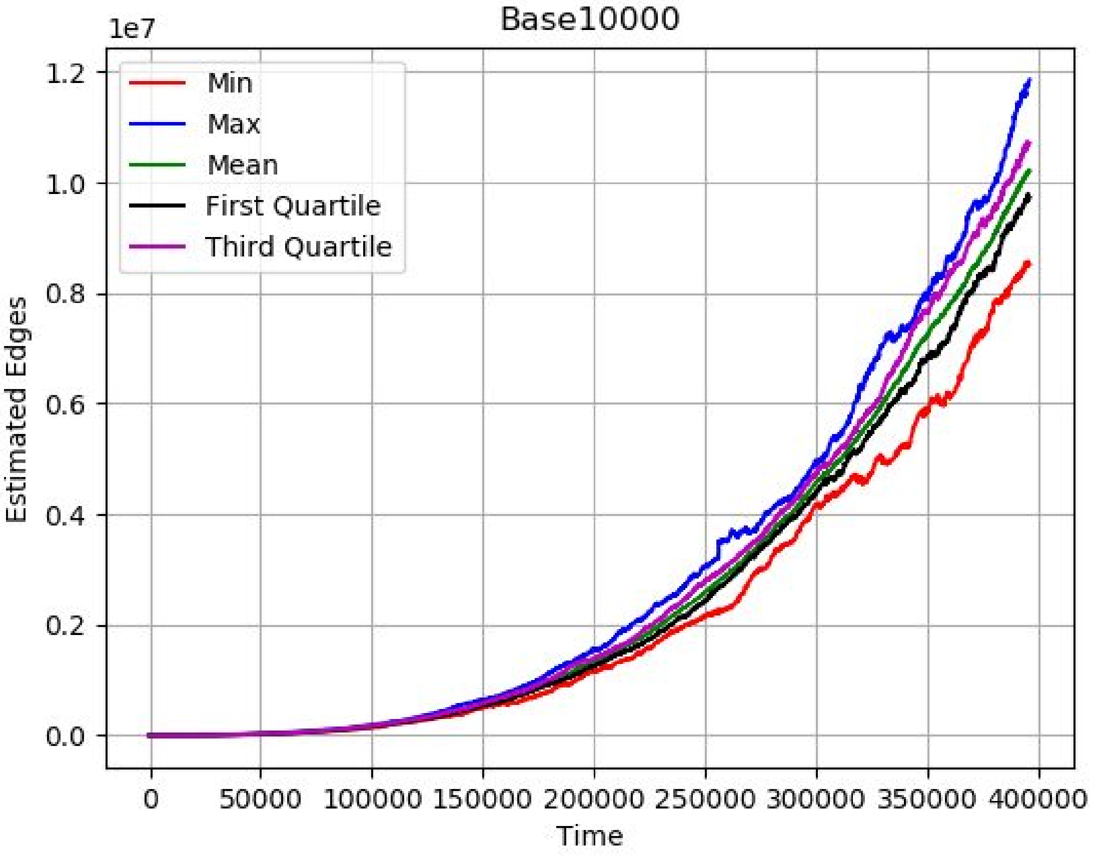
For the Triest Improved algorithm, at a low sample size (5000) we see a much smoother curve than the equivalent size of the base algorithm. We also see much less variation between trials which is apparent by the consistency between the minimum, maximum, mean, and first and third quartile curves. Again, as the sample size increases, the curves smoothen out even more and they tighten even closer together. By sample size 40,000, we see almost no difference between the minimum and maximum curves, and thus everything else in between. The Triest Improved algorithm is extremely precise over many trials although not as accurate as we might hope.

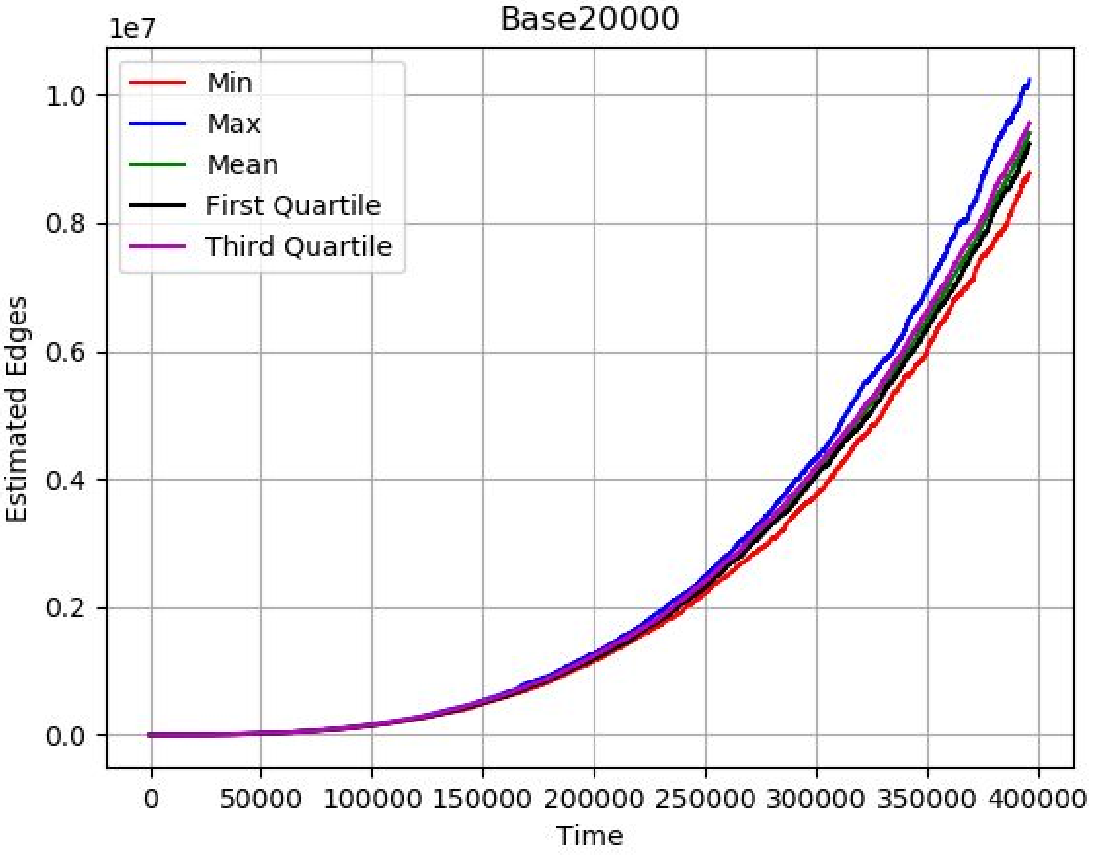
The difference between these two algorithms seems minor but is significant on their effective precisions. The Triest Base algorithm keeps a tally of the actual number of triangles in the graph at any given time. At any given time step, the algorithm decides on whether to add a new edge to the graph and remove an edge at random based on a weighted coin flip that is determined by a ratio between sample size and time. Whether or not an edge is removed, the algorithm then makes a prediction based on the number of triangles in the graph at that time multiplied by a ratio that is again determined by a ratio between time and memory.

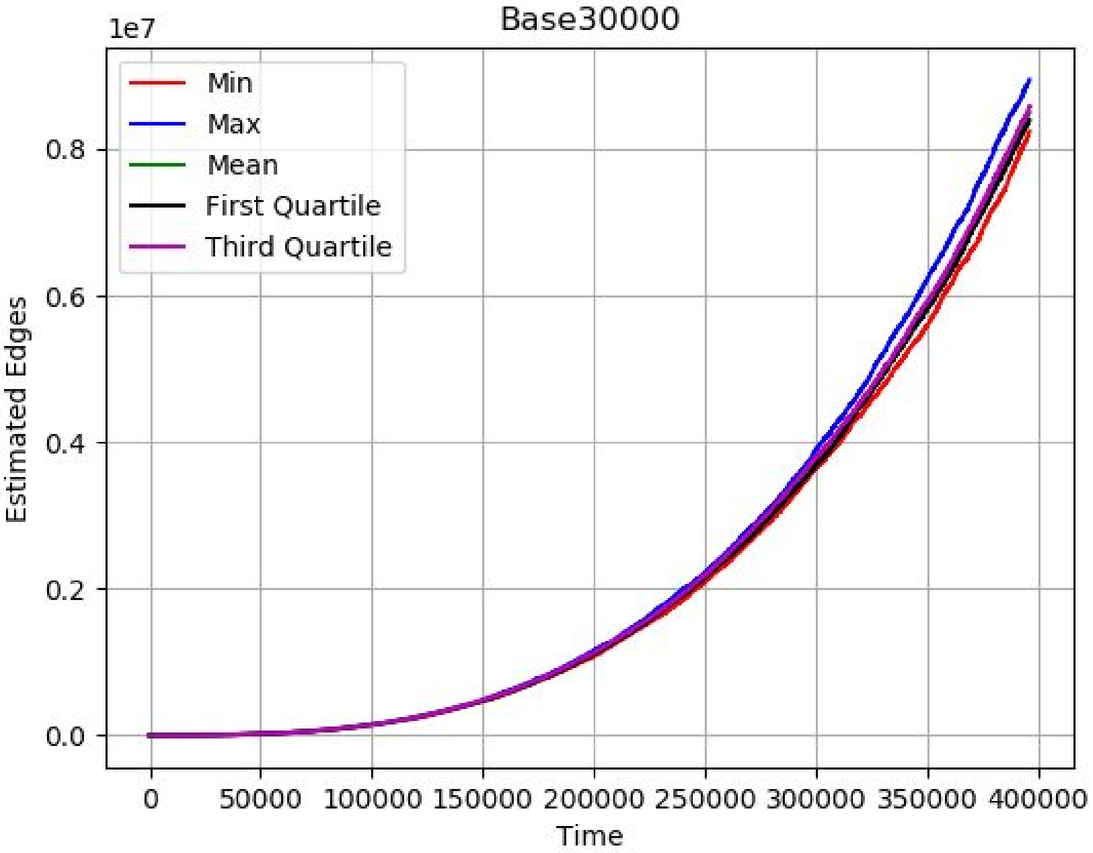
The Triest Improved algorithm instead keeps a running tally of the predicted number of triangles. At every time step, an edge is added to the graph, and the predicted number of triangles is incremented by the number of triangles that the new edge added to the graph multiplied by a ratio between time and sample size. The algorithm then determines whether to keep that edge permanently and remove an edge at random or to just remove the new edge.

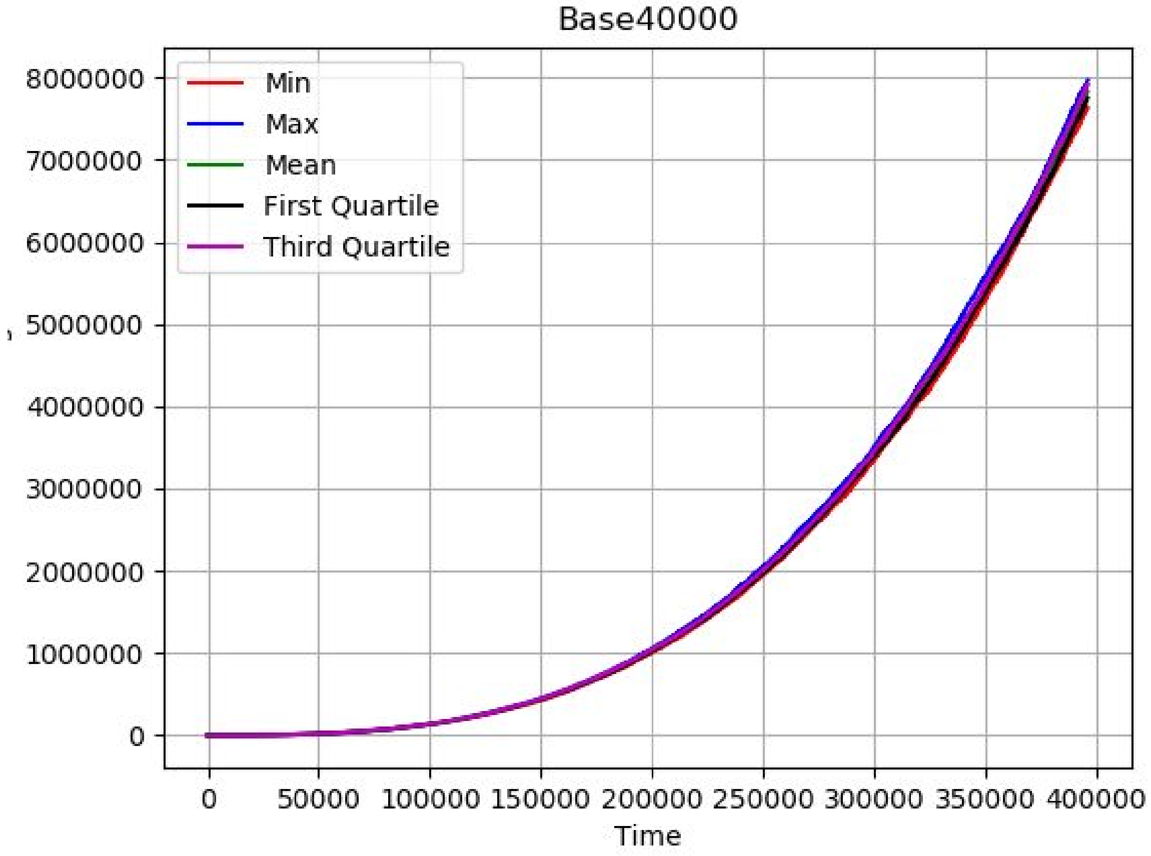
This means that the Triest Improved algorithm is always working on one prediction while Triest Base is making a new prediction at every time step, and thus the Triest Improved has a much more fluid graph. This allows for smoother looking curves because the prediction is only ever increasing, never decreasing.

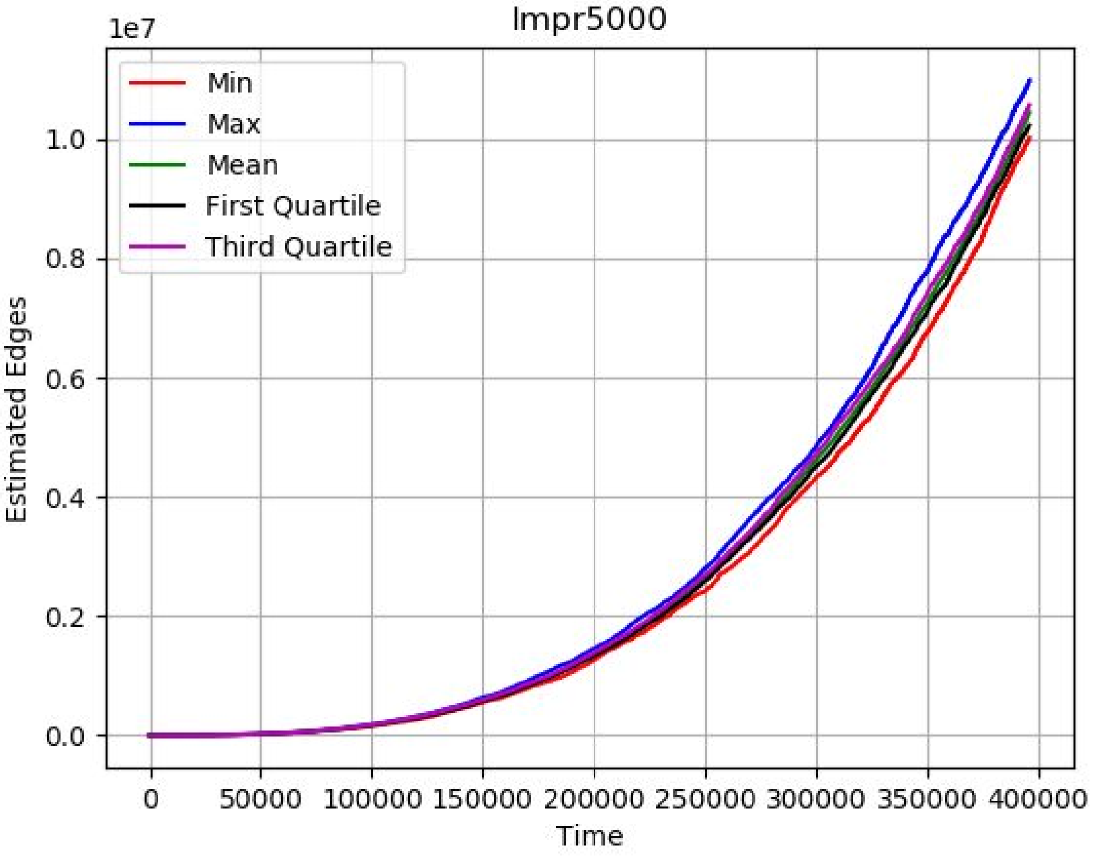

Base5000
1e7
Min
Max
Mean
First Quartile
Third Quartile
Estimated Edges
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
0
50000
100000
150000
200000
250000
300000
350000
400000
Time

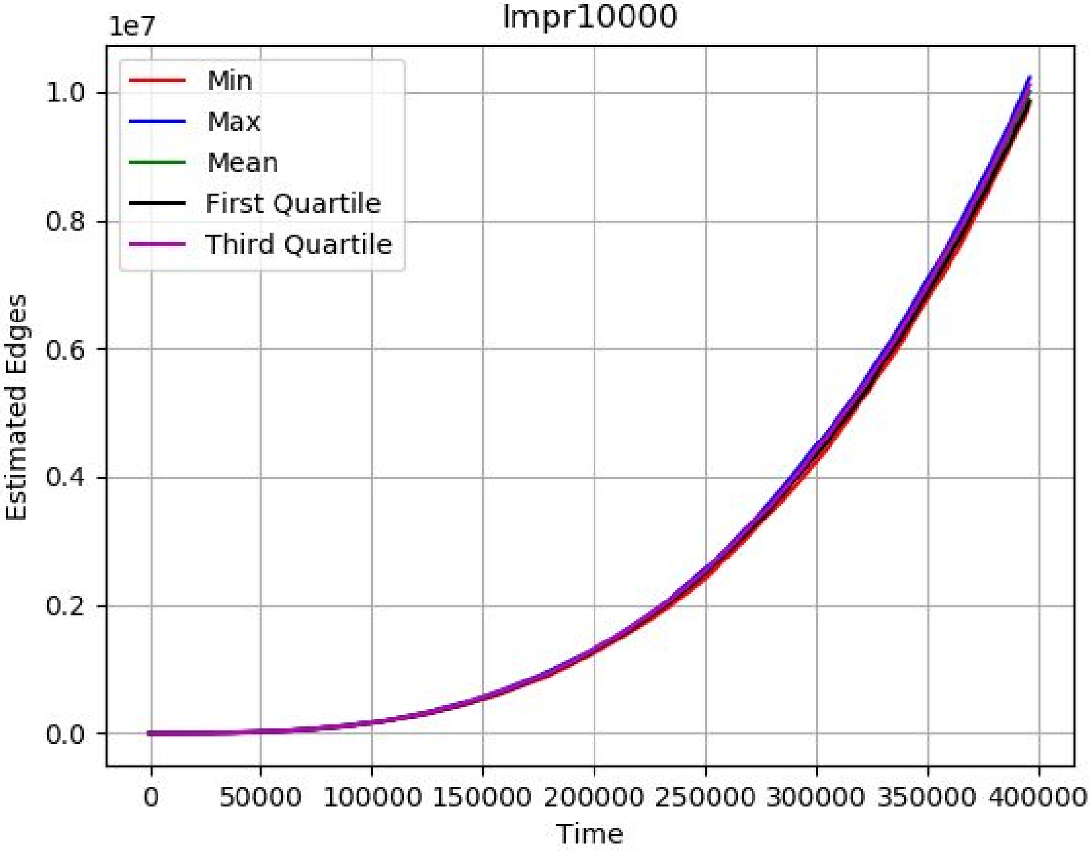

Base10000
1e7
Min
Max
Mean
First Quartile
Third Quartile
Estimated Edges
0.0
0.2
0.4
0.6
0.8
1.0
1.2
Time
0
50000
100000
150000
200000
250000
300000
350000
400000

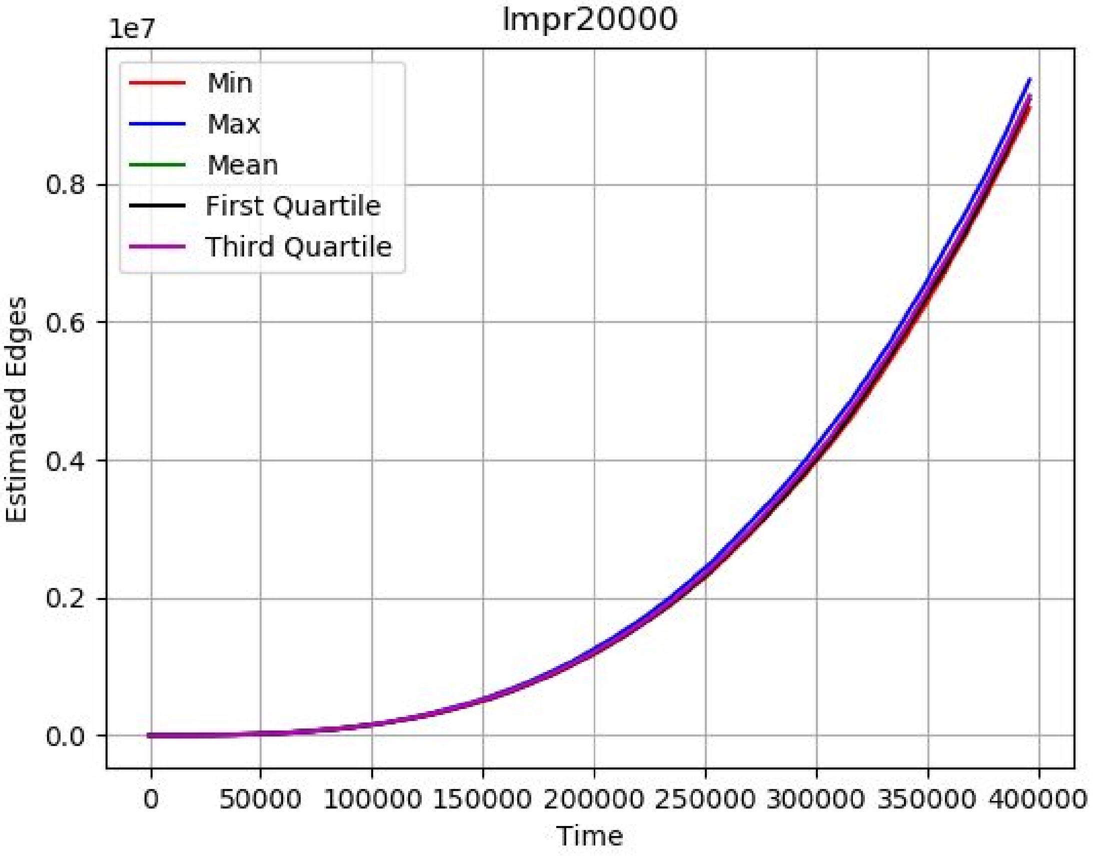

Base20000
1e7
Min
Max
Mean
First Quartile
Third Quartile
1.0
0.8
0.6
0.4
0.2
0.0
Estimated Edges
0
50000
100000
150000
200000
250000
300000
350000
400000
Time

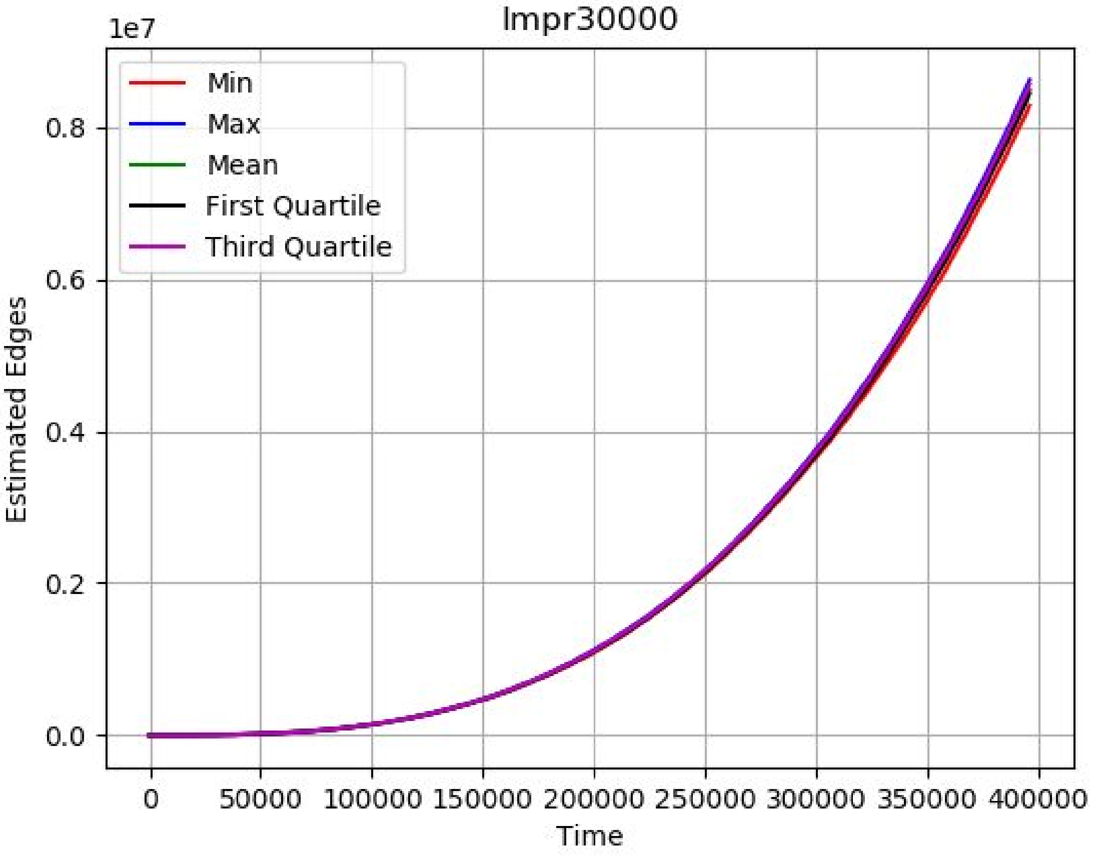

Base30000
1e7
Min
Max
Mean
First Quartile
Third Quartile
Estimated Edges
0.0
0.2
0.4
0.6
0.8
0
50000
100000
150000
200000
250000
300000
350000
400000
Time

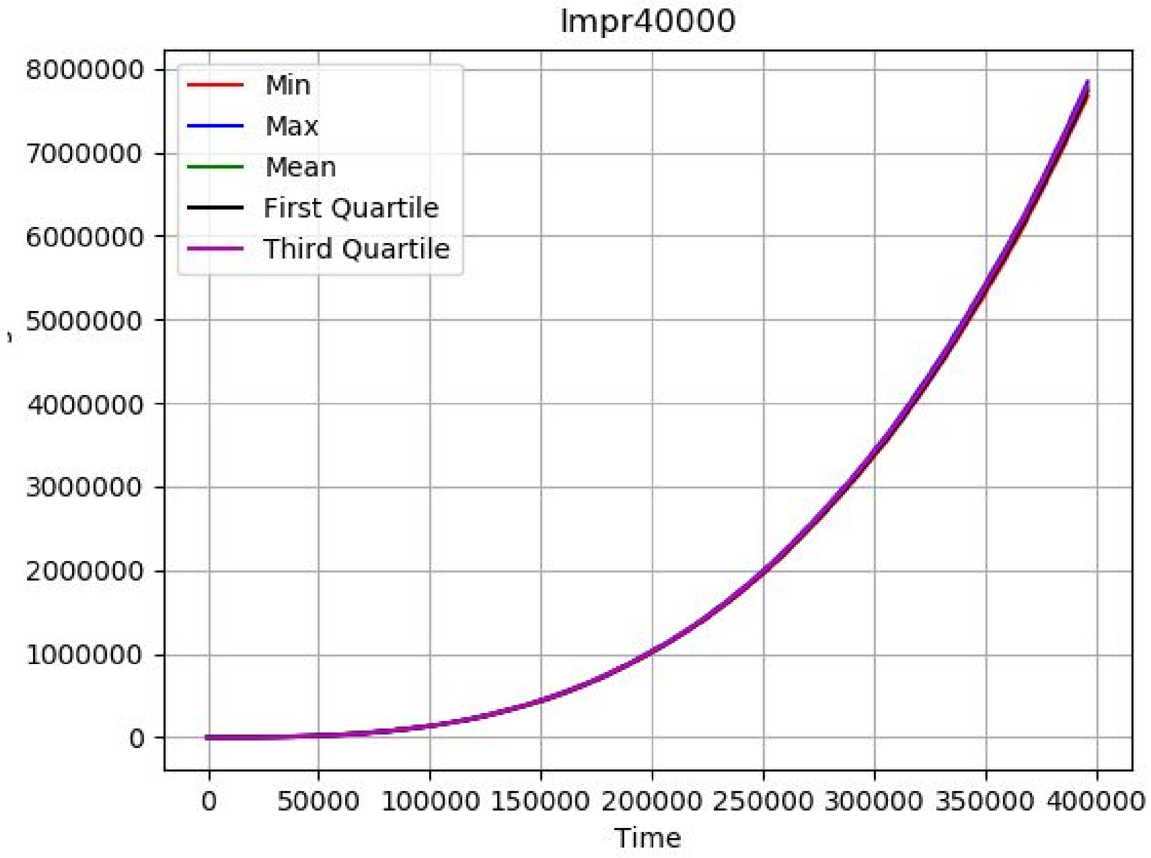

Base40000
Min
Max
Mean
First Quartile
Third Quartile
8000000
7000000
6000000
5000000
4000000
3000000
2000000
1000000
0
0
50000
100000
150000
200000
250000
300000
350000
400000
Time


Impr5000
1e7
Min
Max
Mean
First Quartile
Third Quartile
Estimated Edges
1.0
0.8
0.6
0.4
0.2
0.0
0
50000
100000
150000
200000
250000
300000
350000
400000
Time


Impr10000
1e7
Min
Max
Mean
First Quartile
Third Quartile
Estimated Edges
1.0
0.8
0.6
0.4
0.2
0.0
0
50000
100000
150000
200000
250000
300000
350000
400000
Time


Impr20000
1e7
Min
Max
Mean
First Quartile
Third Quartile
Estimated Edges
0.0
0.2
0.4
0.6
0.8
Time
0
50000
100000
150000
200000
250000
300000
350000
400000


Impr30000
1e7
0.8
0.6
0.4
0.2
0.0
Estimated Edges
0
50000
100000
150000
200000
250000
300000
350000
400000
Time
Min
Max
Mean
First Quartile
Third Quartile


Impr40000
Min
Max
Mean
First Quartile
Third Quartile
8000000
7000000
6000000
5000000
4000000
3000000
2000000
1000000
0
0
50000
100000
150000
200000
250000
300000
350000
400000
Time