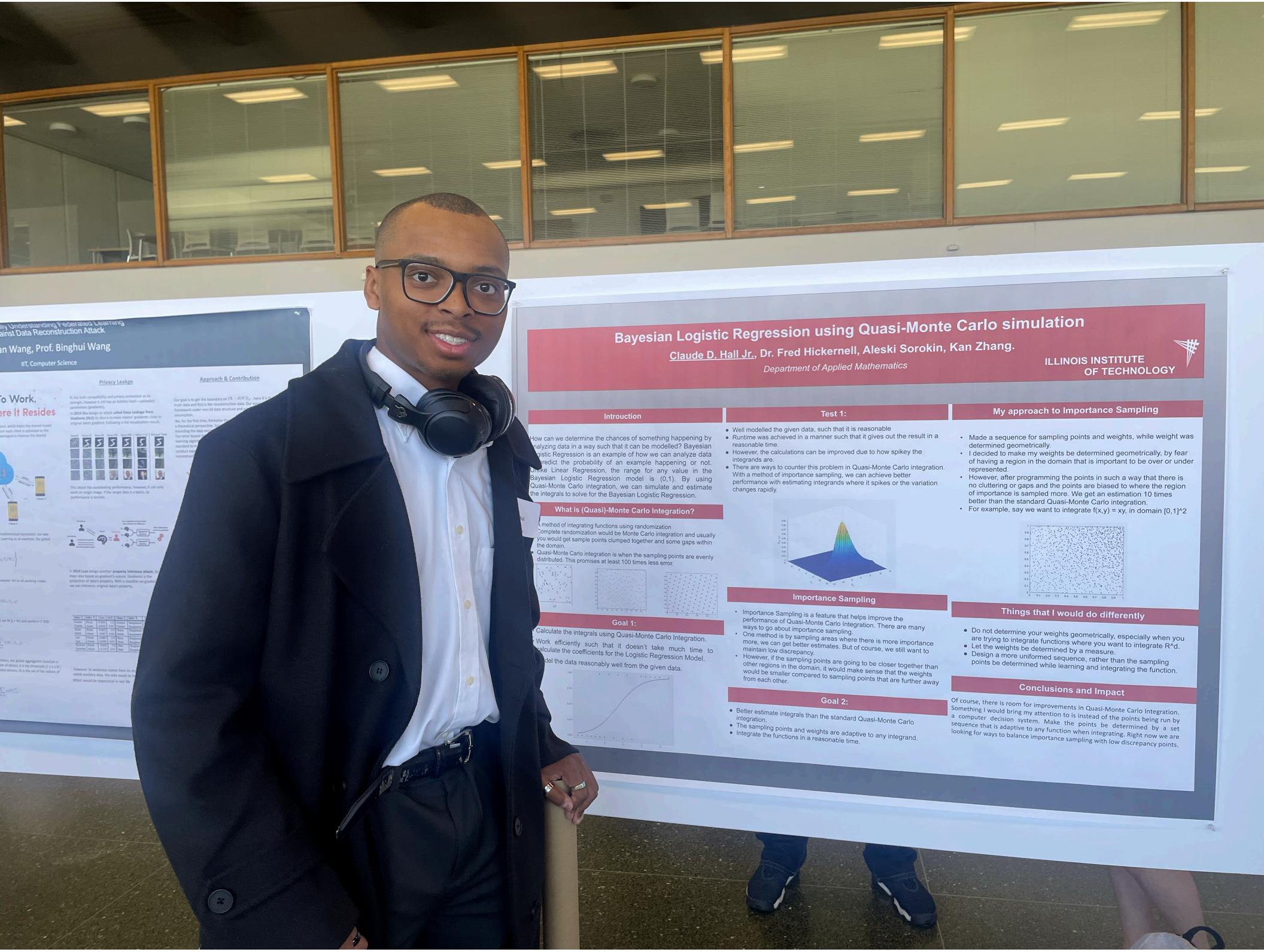
Bayesian Logistic Regression using Quasi-Monte Carlo simulation
Claude D. Hall Jr., Dr. Fred Hickernell, Aleski Sorokin, Kan Zhang.
Department of Applied Mathematics
ILLINOIS INSTITUTE OF TECHNOLOGY
Introuction
How can we determine the chances of something happening by analyzing data in a way such that it can be modelled? Bayesian Logistic Regression is an example of how we can analyze data to predict the probability of an example happening or not. Unlike Linear Regression, the range for any value in the Bayesian Logistic Regression model is (0,1). By using Quasi-Monte Carlo integration, we can simulate and estimate the integrals to solve for the Bayesian Logistic Regression.
What is (Quasi)-Monte Carlo Integration?
A method of integrating functions using randomization
Complete randomization would be Monte Carlo integration and usually you would get sample points clumped together and some gaps within the domain.
Quasi-Monte Carlo integration is when the sampling points are evenly distributed. This promises at least 100 times less error.
Goal 1:
Calculate the integrals using Quasi-Monte Carlo Integration.
Work efficiently such that it doesn't take much time to calculate the coefficients for the Logistic Regression Model.
del the data reasonably well from the given data.
Test 1:
Well modelled the given data, such that it is reasonable
Runtime was achieved in a manner such that it gives out the result in a reasonable time.
However, the calculations can be improved due to how spikey the integrands are.
There are ways to counter this problem in Quasi-Monte Carlo integration. With a method of importance sampling, we can achieve better performance with estimating integrands where it spikes or the variation changes rapidly.
Importance Sampling
Importance Sampling is a feature that helps improve the performance of Quasi-Monte Carlo Integration. There are many ways to go about importance sampling.
One method is by sampling areas where there is more importance more, we can get better estimates. But of course, we still want to maintain low discrepancy.
However, if the sampling points are going to be closer together than other regions in the domain, it would make sense that the weights would be smaller compared to sampling points that are further away from each other.
Goal 2:
Better estimate integrals than the standard Quasi-Monte Carlo integration.
The sampling points and weights are adaptive to any integrand.
Integrate the functions in a reasonable time.
My approach to Importance Sampling
Made a sequence for sampling points and weights, while weight was determined geometrically.
I decided to make my weights be determined geometrically, by fear of having a region in the domain that is important to be over or under represented.
However, after programming the points in such a way that there is no cluttering or gaps and the points are biased to where the region of importance is sampled more. We get an estimation 10 times better than the standard Quasi-Monte Carlo integration.
For example, say we want to integrate f(x,y) = xy, in domain [0,1]^2
Things that I would do differently
Do not determine your weights geometrically, especially when you are trying to integrate functions where you want to integrate R^d.
Let the weights be determined by a measure.
Design a more uniformed sequence, rather than the sampling points be determined while learning and integrating the function.
Conclusions and Impact
Of course, there is room for improvements in Quasi-Monte Carlo Integration. Something I would bring my attention to is instead of the points being run by a computer decision system. Make the points be determined by a set sequence that is adaptive to any function when integrating. Right now we are looking for ways to balance importance sampling with low discrepancy points.