ST2195 COURSEWORK REPORT

Programming for Data Science

LORENCIA LO

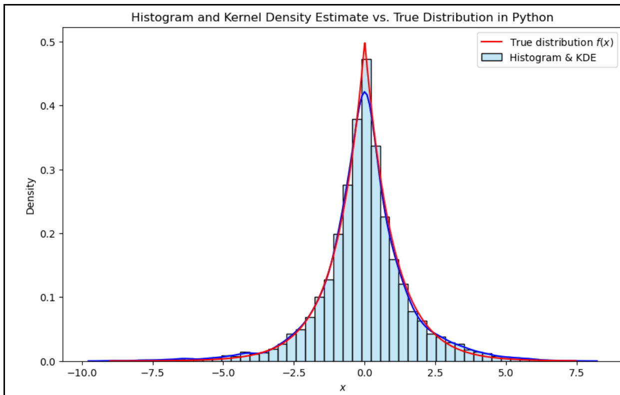
STUDENT NUMBER : 220458845

Table of Contents

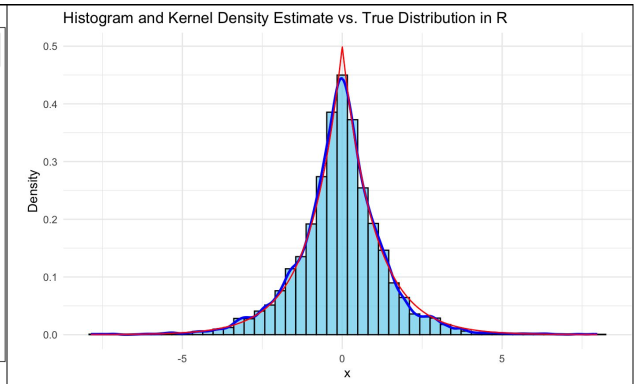
Introduction

This report encompasses a comprehensive examination of the Markov Chain Monte Carlo (MCMC) algorithm, specifically focusing on the Metropolis-Hastings algorithm. This algorithm is pivotal in statistical computing for generating random numbers from complex probability distributions where direct sampling is not feasible. The project is divided into two main parts, each designed to deepen the understanding of MCMC techniques and their practical applications in generating and analyzing statistical data.

Part 1(a) : Random Walk Metropolis Algorithm



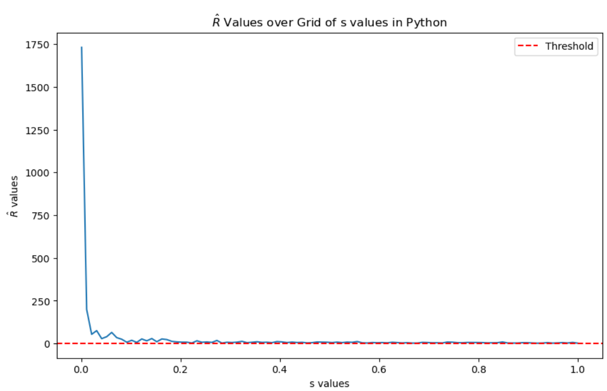
Sample Mean: -0.025851114883755 69
Sample Standard Deviation: 1.5907356012269236

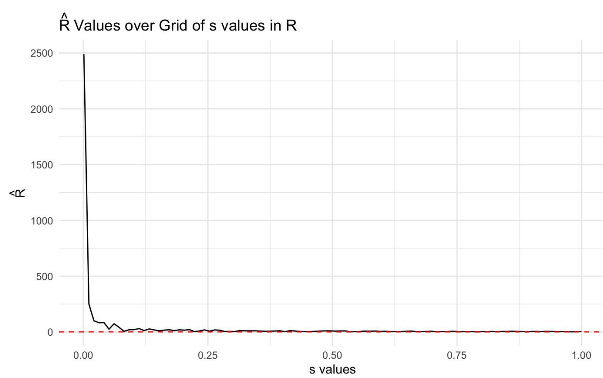


Sample Mean: -0.06286599
Sample Standard Deviation: 1.389465

The data produced by the simulations in both Python and R are visualized through histograms and kernel density plots, with the intent of estimating the PDF from the simulated data. The histograms provide a discrete representation of the data's distribution, while the kernel density estimates provide a smooth representation of the PDF. In comparison to the actual PDF, the visuals from both programming languages shows close alignment. The Python simulation generates a sample mean around -0.026 and a standard deviation close to 1.59, whereas the R simulation produces a sample mean near -0.063 and a standard deviation close to 1.39, illustrating the algorithm's effectiveness in mirroring the distribution's expected mean and variability.

Part 1(b) : Convergence Diagnostic – $\hat{R}$ values





The provided graphs display the $\hat{R}$ value's response to changing the scale parameter s in the Random Walk Metropolis algorithm. The plots reveal a sharp decrease in $\hat{R}$ as s increases from 0.001, stabilizing around a value that suggests convergence according to the $\hat{R}$ diagnostic. Notably, both graphs align on the inference that after an initial decline, a higher s value enhances the effectiveness of the sampling process. This is evidenced by the $\hat{R}$ values nearing 1, suggesting that the algorithm has successfully captured the characteristics of the target distribution, an essential factor for subsequent analytical procedures.