

Stat 243 Final Project Writeup

Vaibhav Ramamoorthy, Colin Kou, Brandon Mannion, and Jonathan Lee

12/7/2018

Functions

We created a very modular solution for our Adaptive-Rejection Sampler, which handled unique subtasks within individual functions. Below, we list each function we created and its modular purpose:

- `get_initial_abscissae()`: Takes in the log of the density and the domain of the density as well as optional arguments `k`, the number of abscissae; `x_start`, the initial abscissae values to try; and `x_step`, the amount to step if the derivative value at the `x_start` is not the appropriate sign. This function is called within `ars()` to set the initial abscissae.
- `get_z()`: Takes in an index `j`, a vector of abscissae `x`, and a concave function `h` and calculates the coordinate of the intersection point between the tangent lines to `h` at `x[j]` and `x[j+1]`. This function is called within `get_z_all()` in a vectorized fashion to get the vector of `z`'s.
- `get_z_all()`: Takes in a vector of abscissae `x` and a concave function `h` and calculates the vector of `z`'s (tangent line intersection points) by calling an apply function that applies `get_z()` to the elements of `x`. This function is called within `ars()` to recreate the `z` vector each time the abscissae get updated.
- `get_u_segment()`: Takes in an index `j`, a vector of abscissae `x`, and a concave function `h` and calculates the slope and intercept of the tangent line to `h` at `x[j]`. This function is called within `get_u()` in a vectorized fashion to get all the piece-wise segments of the upper bound of `h`.
- `get_u()`: Takes in a vector of abscissae `x` and a concave function `h` and applies `get_u_segment()` to get the entire piece-wise upper bound of `h`. This function is called within `ars()` to recreate `u` each time the abscissae get updated.

Tests

Team Member Contributions