FIN 5350- Homework 3

Tyler J. Brough

November 16, 2019

Numerical Problems

Problem 1

- Start with the file `pricers.py` that I provided in class. Add to the module a function titled `naive_monte_carlo_pricer` that implements the naive Monte Carlo method to price European calls and puts. Use the class `VanillaOption` to pass a first argument `option` to the pricer.
- Make sure the pricer function returns a `namedtuple` that contains the estimated price and the standard error. This might look something like this:
- See the jupyter notebook titled `Namedtuple-Hints.ipynb` for help with this.
- Price European call and put options where $S_T = \$41.0$, $K = \$40.0$, $r = 8\%$, $\sigma = 30\%$, $\delta = 0.0$, $T = 1.0$ using your new pricer function.
- Create a markdown table that presents the number of repititions, the estimated price and the standard error for $M = 1000, 10000, 25000, 50000, 750000,$ and 100000 where M is the number of repitions in the Monte Carlo simulations.

Problem 2

- Write another new pricer function titled `antithetic_monte_carlo_pricer` that implements the antithetic variance reduction technique.
- Reprice the call and puts above and reproduce and add to the table a comparison of the naive Monte Carlo results with the antithetic sampling results. Is there a reduction in the standard error?

Problem 3

- Write another new pricer function titled `stratified_monte_carlo_pricer` that implements the stratified sampling variance reduction technique.
- Add the new results to the table. Now compare naive Monte Carlo, antithetic Monte Carlo, and stratified Monte Carlo. What do you notice?