Simulation Challenge

Generative Models and Monte Carlo Simulation

Simulation Challenge - Monte Carlo Analysis

Challenge Overview

Your Mission (along with your Cursor AI partner): Create a comprehensive Quarto document that simulates an investment strategy (or possibly two strategies), analyzes the simulation results, and presents your ability to simulate a scenario and present counter-intuitve results in a compelling way.

The Investment Game

Original Game Strategy

Example 0.1. Imagine you are offered the following game and given a $1,000 budget in a special account to play the game: I will flip a coin, and if it comes up heads, we increase your account's balance by 50%; if it comes up tails, we reduce your account's balance by 40%. We are not only doing this once, but we will do it once per year until you turn 55. When you turn 55, you will receive the balance in your account.

Challenge Requirements

Minimum Requirements for Any Points on Challenge

1. **Create a Quarto Document:** Write a concise quarto markdown file that includes a narrative of what you are doing along with the requested code, results, and visualizations of your simulations.
2. **Render to HTML:** You must render the quarto markdown file to HTML.

3. **GitHub Repository:** The rendered HTML must be uploaded to a new GitHub repository called "simulationChallenge" in your Github account.

4. **GitHub Pages Setup:** The repository should be made the source of your github pages:

   - Go to your repository settings (click the "Settings" tab in your GitHub repository)
   - Scroll down to the "Pages" section in the left sidebar
   - Under "Source", select "Deploy from a branch"
   - Choose "main" branch and "/ (root)" folder
   - Click "Save"
   - Your site will be available at: `https://[your-username].github.io/simulationChallenge/`
   - **Note:** It may take a few minutes for the site to become available after enabling Pages

Grading Rubric

Questions to Answer for 75% Grade on Challenge

1. **Expected Value Analysis:** What is the "expected value" of your account balance after 1 coin flip for the original game?

2. **Expectation vs. Reality:** Is the expected value positive or negative? Do you expect your account to be worth more or less than $1,000 based on this result?

3. **Single Simulation:** Run one simulation showing the dynamics of your account balance over time. Make an object-oriented matplotlib OR ggplot2 plot showing your simulated account balance over time (i.e. as you age). Comment on the results, are you happy?

Questions to Answer for 85% Grade on Challenge

4. **Multiple Simulations:** Run 100 simulations showing the dynamics of your account balance over time. Make an object-oriented matplotlib OR ggplot2 plot showing a probability distribution of account balance at age 55. Comment on the results, are you happy? Why or why not?

Questions to Answer for 95% Grade on Challenge

5. **Probability Analysis:** Based on the 100 simulations above, what is the probability that your account balance will be greater than $1,000 at age 55?

Questions to Answer for 100% Grade on Challenge

6. **Strategy Comparison:** Run 100 simulations for the modified game strategy shown below in Example 0.2. What is the probability that your account balance will be greater than $10,000 at age 55? Is this probability higher or lower than the probability in the original game?

Modified Game Strategy

Example 0.2. Imagine you are offered the following game and given a $1,000 budget in a special account to play the game: I will flip a coin, and if it comes up heads, we increase your bet by 50%; if it comes up tails, we reduce your bet by 40%. You must bet exactly 50% of your current account balance on each flip, and this 50% is locked in for each round. We are not only doing this once, but we will do it once per year until you turn 55. When you turn 55, you will receive the balance in your account.

Technical Implementation Preferences

Setting Up Your Analysis

For R Users: - Use `tidyverse` for data manipulation - Use `ggplot2` for visualizations - Use `set.seed()` for reproducible results

For Python Users: - Use `numpy` for numerical operations - Use `pandas` for data manipulation - Use `matplotlib` (object-oriented) - Use `np.random.seed()` for reproducible results

Visualization Preferences

- **Professional Styling:** Use consistent colors, clear labels, readable fonts, and informative titles

Submission Checklist

Minimum Requirements (Required for Any Points): - [ ] Quarto document created with clear narrative - [ ] Document rendered to HTML successfully - [ ] Repository "simulationChallenge" created - [ ] HTML files uploaded to repository - [ ] GitHub Pages enabled and working - [ ] Site accessible at `https://[your-username].github.io/simulationChallenge/`

75% Grade Requirements: - [ ] Expected value calculations shown (Question 1) - [ ] Expectation vs. reality analysis (Question 2) - [ ] Single simulation with time series plot (Question 3) - [ ] Clear interpretation of single simulation results

85% Grade Requirements: - [ ] 100 simulations with distribution analysis (Question 4) - [ ] Probability distribution plot of final account balances - [ ] Clear interpretation of multiple simulation results

95% Grade Requirements: - [ ] Probability calculations for original strategy (Question 5) - [ ] Analysis of probability that balance > $1,000 at age 55

100% Grade Requirements: - [ ] 100 simulations for modified strategy (Question 6) - [ ] Probability calculations for modified strategy - [ ] Comparative analysis between both strategies - [ ] Analysis of probability that balance > $10,000 at age 55

Code Quality (All Grades): - [ ] Reproducible results (seeds set) - [ ] Clean, well-commented code - [ ] Appropriate use of functions and loops - [ ] Professional visualization styling

Resources

- **Quarto Documentation:** quarto.org/docs
- **R for Data Science:** r4ds.had.co.nz
- **Python Data Science Handbook:** jakevdp.github.io/PythonDataScienceHandbook

Getting Started Tips

- **Start Simple:** Begin with a single simulation to understand the mechanics
- **Document Everything:** Explain your reasoning and interpret your results
- **Forgetting to Set Seeds:** Always set random seeds for reproducible results
- **Total time to complete:** ~3-4 hours for the 100% grade
- **Good luck, and remember simulation will steer you right even when intuition will steer you wrong!**