Problem Set 3: Simulating a Dice Game

Introduction

In this problem set you will be writing simulation program to estimate some values whose true value is difficult to obtain. Specifically, you will estimate the probability of the game of Craps.

Getting Started

Download ProblemSet3.zip from IVY

Please do not rename the files, change any of the provided helper functions, change function/method names.

Problem : Pass or Don't Pass?

Not all questions about games of chance are so easily answered. In the game craps, the shooter (the person who rolls two dice) chooses between making a "pass line" or a "don't pass line" bet.

- Pass Line: Shooter wins if the first roll is a "natural" (7 or 11) and loses if it is "craps" (2, 3,or 12). If some other number is rolled, that number becomes the "point" and the shooter keeps rolling. If the shooter rolls the point before rolling a 7, the shooter wins. Otherwise the shooter loses.
- Don't pass Line: Shooter loses if the first roll is 7 or 11, wins if it is 2 or 3, and ties (a "push" in gambling jargon) if it is 12. If some other number is rolled, that number becomes the point and shooter keeps rolling. If the shooter rolls a 7 before rolling the point, the shooter wins. Otherwise the shooter loses.

Is one of these a better bet than the other? Is either a good bet? It is possible to analytically derive the answer to these questions, but it seems easier (at least to us) to write a program that simulates a craps game, and see what happens.

In ps3.py, You are to complete the CrapsGame class.

The values of the instance variables of an instance of class CrapsGame record the performance of the pass and don't pass lines since the start of the game. The methods passResults and dpResults return these values. The method playHand simulates one hand of a game. A "hand" starts when the shooter is "coming out", the term used in craps for a roll before a point is established. A hand ends when the shooter has won or lost his or her initial bet.

Before the actual implementation of the simulation, you are to define some basic statistical helper functions:

- mean
- variance
- standard deviation

With the CrapsGame class, you are to define a function crapsSim that simulate a series of craps games.

The structure of crapsSim is typical of many simulation programs:

- it runs multiple games (think of each game as analogous to a trial in our earlier simulations) and accumulates the results. Each game includes multiple hands, so there is a nested loop.
- it then produces and stores statistics for each game.

- finally, it produces and outputs summary statistics. In this case, it prints the expected return on investment (ROI) of each kind of betting line and the standard deviation of that ROI.

Return on investment is defined by the equation, $ROI = \frac{gain\ from\ investment - cost\ of\ investment}{cost\ of\ investment}$.

Since the pass and don't pass lines pay even money (if you bet $1 and win, you gain $1), the ROI is

$$ROI = \frac{number\ of\ wins - number\ of\ losses}{number\ of\ bets}$$

For example, if you made 100 pass line bets and won half, your ROI would be $\frac{50-50}{100} = 0$.

If you bet the don't pass line 100 times and had 25 wins and 5 pushes, the ROI would be $\frac{25-70}{100} = -4.5$

To test your code:

```
>>> random.seed(0)
>>> crapsSim(20,10)
Pass: Mean ROI = 4.0% Std. Dev. = 27.6405%
Don't Pass: Mean ROI = -7.5% Std. Dev. = 24.8244%
>>> random.seed(0)
>>> crapsSim(1000000,10)
Pass: Mean ROI = -1.4204% Std. Dev. = 0.0614%
Don't Pass: Mean ROI = -1.3571% Std. Dev. = 0.0593%
```