1 2D Random Walk

To solve this problem, we first obtain x_n for one random walker. Put a random walker at the origin. Create a random number *movestep* in range (0,1,2,3), and these four numbers means moving one step right/left/up/down respectively. We use a list to record the number of moving steps in every direction. After we get the moving direction of a step, we add the number of total steps in that certain direction by 1. If the total number of moving steps is n, we repeat the previous procedure n times. The number of total rightward steps minus the number of total leftward steps generates the x-component of the random walker's displacement, x_n. Likewise, the number of total upward steps minus the number of total downward steps generates the y-component of the random walker's displacement, y_n. After that, x_n^2 and r_n^2 are easy to calculate. We use a function named random_walk to return the values of x_n, x_n^2, and r_n^2. Since our goal is averaging over 10^4 different walkers, we need a loop with 10^4 iterations to get the values of x_n, x_n^2, and r_n^2 of each walker, and then realize the average values, $\langle x_n \rangle$, $\langle x_n^2 \rangle$, and $\langle r_n^2 \rangle$.

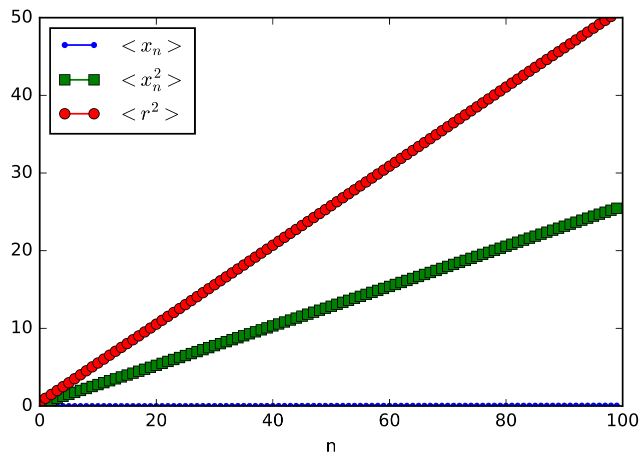

Figure 1: