Home Assignment 2

Monte Carlo and Empirical Methods for Stochastic Inference

Group 3

Adrian Murphy
ad5880mu-s@student.lu.se

Daniel Larsson
da6625la-s@student.lu.se

March 27, 2023

1 Self-avoiding Walks in $\mathbb{Z}^d$

A self-avoiding walk is a sequence of moves on a d-dimensional grid where a point is not visited more than once. The set S_n of possible walks in n steps is given by

$$S_n(d) = \{x_{0:n} \in \mathbb{Z}^{d(n+1)} : x_0 = \vec{0}, |x_k - x_{k-1}| = 1, x_l \neq x_k, \forall 0 \leq l \leq k \leq n\}$$

where we say that $c_n(d) = |S_n(d)|$ is the number of possible walks.

1.1

We begin by convincing ourselves that for all $n \geq 1$ and $m \geq 1$ it holds that $c_{n+m}(d) \leq c_n(d)c_m(d)$. We can interpret this as having fewer or an equal amount of possible walks by taking $n+m$ consecutive steps compared to the product between the number of walks in n and m independent steps. In an initial step, we can take a step in either direction of d dimensions and thus have $2d$ possible steps. In the subsequent steps, we can at most go in $2d-1$ directions since we may not return to the previous point. Thus, for $c_{n+m}(d)$ we can at most take $2d-1$ steps after the nth step. This is not the case for $c_n(d)c_m(d)$, where we begin a new self-avoiding walk after n steps with $2d$ possible initial steps. Therefore, the inequality in holds.

1.2

A sequence $(a_n)_{n\geq 1}$ is subadditive if $a_{m+n} \leq a_m + a_n$. According to Fekete's lemma, for every subadditive sequence the limit $\lim_{n\to\infty} a_n/n$ exists and is equal to $\inf_{n\geq 1} a_n/n$. We now wish to show that the following limit exists

$$\mu_d = \lim_{n\to\infty} c_n(d)^{1/n}$$

We use that $c_{n+m}(d) \leq c_n(d)c_m(d)$ and

$$\log\, c_{n+m}(d) \leq \log\, c_n(d) + \log\, c_m(d)$$

Then, by definition, $(\log\, c_n(d))_{n\geq 1}$ is a subadditive sequence. We can also take the logarithm of μ_d as

$$\log\, \mu_d = \lim_{n\to\infty} \frac{\log\, c_n(d)}{n}$$

Since we have shown that $(\log\, c_n(d))_{n\geq 1}$ is a subadditive sequence, this limit, by Fekete's lemma, exists and is equal to

$$\log\, \mu_d = \lim_{n\to\infty} \frac{\log\, c_n(d)}{n} = \inf_{n\geq 1} \frac{\log\, c_n(d)}{n}$$

Then, for some base b it holds that

$$\hat{\mu}_d = b^{\mu_d} = \lim_{n\to\infty} b^{\log_b\, c_n(d)/n} = \lim_{n\to\infty} c_n(d)^{1/n}$$

$\square$

1.3

One of the purposes of this assignment is to estimate the so-called connective constant μ_d in the previous exercise. We start by examining the two-dimensional case, i.e. we use the lattice $\mathbb{Z}^2$ and estimate μ_2. For $d = 2$ we, per the instructions, have that

$$c_n(2) \sim A_2 \mu_2^n n^{\gamma_2 - 1} \tag{1}$$

where $\gamma_2 = \frac{43}{32}$ and we let $n \to \infty$.

We can estimate $c_n(2)$ by using sequential importance sampling (SIS). We let the instrumental distribution g_n be a standard random walk $(X_k)_{k=0}^n$ where $X_0 = \vec{0}$ and X_{k+1} is one of X_k's four neighbors with equal probability. Of course, this standard random walk is not self-avoiding. However, we can simulate N random walks on our lattice where we say that N_{SA} is the number of self-avoiding walks among these walks. Using these values we can estimate a ratio of self-avoiding walks, $\frac{N_{\text{SA}}}{N}$, which, if we multiply it with 4^n, becomes an estimate of $c_n(2)$. This is because we can go in four different directions in every step, and multiplying these possible directions with the estimated ratio of self-avoiding walks we get an estimate of the number of possible self-avoiding walks $c_n(2)$.

Using step sizes $1 \leq n \leq 24$ and simulating $N = 10^4$ random walks for each n, we get the following estimates (rounded to the closest integer)

Estimates					
Steps n	$c_n(2)$	Error	Steps n	$c_n(2)$	Error
1	4	0%	13	845572	4.08%
2	12	0%	14	2254858	5.04%
3	37	2.78%	15	6871948	7.1%
4	98	2%	16	18468359	7.09%
5	283	0.35%	17	46385647	0.17%
6	765	1.92%	18	151182849	21.28%
7	2192	0.92%	19	494780232	47.64%
8	6108	3.25%	20	989560465	10.23%
9	15755	3.15%	21	3518437209	46.07%
10	42153	4.41%	22	7036874418	9.19%
11	131701	9.48%	23	14073748836	18.49%
12	320445	1.38%	24	28147497671	39%
Average Error: 10.21%					

Table 1: Estimated $c_n(2)$ using SIS with standard RW as instrumental distribution

The error presented in table 1 is simply the deviation, in percentage and rounded to two decimals, from the true value presented by Slade [6]. Of course, the rounding of $c_n(2)$ to the nearest integer sometimes eliminates the error for small n, which gives a somewhat misleading 0% error. For relatively larger n, we seem to generate much fewer self-avoiding walks (in some simulations, we did in fact not generate a single self-avoiding walk for some $n \geq 20$). Hence, the estimated number of walks $c_n(2)$ becomes unreasonable since it relies on the ratio between self-avoiding and total walks. Since we want to estimate the connective constant with larger n, this simple SIS approach might not be suitable. To further investigate the suitability of this approach when estimating μ_2, we can rewrite equation 1 as

$$\frac{\mu_2^{n+1}}{\mu_2^n} = \mu_2 \sim \frac{c_{n+1}(2)A_2 n^{\gamma_2 - 1}}{c_n(2)A_2(n+1)^{\gamma_2 - 1}} = \frac{c_{n+1}(2)n^{11/32}}{c_n(2)(n+1)^{11/32}}$$

From [7], we know that the connective constant is most likely bounded by $\mu_2 \in [2.62, 2.68]$. If we estimate μ_2 using the equation above for $1 \leq n \leq 25$ and $N = 10^4$, we get

Estimates			
n	μ_2	n	μ_2
1	2.364	13	2.5996
2	2.6822	14	2.9762
3	2.3993	15	2.6285
4	2.6745	16	2.4598
5	2.539	17	3.1958
6	2.7175	18	3.2125
7	2.6615	19	1.965
8	2.4771	20	3.4964
9	2.5804	21	1.9683
10	3.0237	22	1.9697
11	2.3614	23	1.971
12	2.5671	24	7.8885

Table 2: Estimated μ_2 using SIS with standard RW as instrumental distribution

As we can see in table 2, the estimated values for the connective constant seem to deteriorate with larger n, which was expected when taking the errors in 1 into account.

1.4

We now wish to improve the estimates in the previous section by instead using a self-avoiding random walk in $\mathbb{Z}^2$ as instrumental distribution. We start at the origin, $X_0 = \vec{0}$, and choose X_{k+1} as one of X_k's free neighbors, given $X_{0:k}$, with equal probability. If there are no free neighbors, we set $X_{k+1} = X_k$. As described in lecture 7 [4], we then have an estimator

$$\sum_{i=1}^{N} \frac{\omega_n^i}{\sum_{l=1}^{N} \omega_n^l} \phi(X_i^{0:n}) \approx \mathbb{E}_{f_n}(\phi(X_{0:n}))$$

which is updated by the SIS algorithm to

$$\sum_{i=1}^{N} \frac{\omega_{n+1}^i}{\sum_{l=1}^{N} \omega_{n+1}^l} \phi(X_i^{0:n+1}) \approx \mathbb{E}_{f_{n+1}}(\phi(X_{0:n+1}))$$

where we can update the weights sequentially as

$$\omega_{n+1}^i = \frac{z_{n+1}(X_i^{0:n+1})}{z_n(X_i^{0:n}) g_{n+1}(X_i^{0:n+1} | X_i^{0:n})} \cdot \omega_n^i \tag{2}$$

In our case, we let z_n be an indicator function yielding 1 if there are free neighbors given $X_i^{0:n}$ and 0 otherwise. g_n is the instrumental distribution, in this case given by the self-avoiding walk, and $g_{n+1}(X_i^{0:n+1}|X_i^{0:n})$ is then equivalent to the probability of going to $X_i^{0:n+1}$ given $X_i^{0:n}$. Since g is just a uniform distribution, equation 2 should be interpreted as obtaining the subsequent weight by multiplying the previous weight with the number of free neighbors given $X_i^{0:n}$. As long as the walk does not terminate by a lack of free neighbors, yielding a weight equal to 0, ω_{n+1}^i can be interpreted as a simulation of the number of possible walks in n steps. We then have an unbiased estimate of c_n, as the average of the simulated possible walks, in

$$c_n \approx \frac{1}{N} \sum_{i=1}^{N} \omega_n^i$$

After initializing variables X with zeros, ω with ones and `steps` as a 4×2 matrix where each row corresponds to a step in the grid, we implement the following procedure in Matlab

```
for k = 0 → n do
    for i = 1 → N do
        A = X_i^{0:k}
        b = X_i^k
        free = {b + steps} \ A

        if free == ∅ then
            X_i^{k+1} ← b
```

```
            ω_i^{k+1} ← 0
        else
            X_i^{k+1} ← random_U(b + steps)
            ω_i^{k+1} ← ω_i^k · #free
        end if
    end for
    c_{k+1}(2) ← mean(ω^{k+1})
end for
```

and run it for $1 \le n \le 24$ and $N = 10^4$

Estimates					
Steps n	$c_n(2)$	Error	Steps n	$c_n(2)$	Error
1	4	0%	13	880649	0.1%
2	12	0%	14	2368193	0.26%
3	36	0%	15	6391181	0.4%
4	100	0%	16	17134526	0.64%
5	284	0%	17	46112662	0.76%
6	778	0.26%	18	124247419	0.33%
7	2170	0.09%	19	334976817	0.04%
8	5915	0.02%	20	892336432	0.6%
9	16267	0.01%	21	2401619264	0.3%
10	44105	0.01%	22	6415640460	0.45%
11	120201	0.08%	23	17214194063	0.3%
12	324071	0.26%	24	46103429450	0.09%
Average Error: 0.21%					

Table 3: Estimated $c_n(2)$ using SIS with SAW as instrumental distribution

In table 3, we can see an immense improvement over the naive approach in 1.3, especially for relatively larger n. The average error for all the c_n, $1 \le n \le 24$, is now 0.21% compared to 10.21% earlier. Unfortunately, there seems to be a slight upward trend in the error. As we will see later, further improvements can be made to this approach.

We can again give approximations of the connective constant using the relation in equation 1

Estimates			
n	μ_2	n	μ_2
1	2.364	13	2.6215
2	2.6097	14	2.6355
3	2.5162	15	2.6221
4	2.6303	16	2.6357
5	2.573	17	2.642
6	2.6453	18	2.6464
7	2.6035	19	2.6173
8	2.641	20	2.6466
9	2.6149	21	2.629
10	2.6375	22	2.6425
11	2.6166	23	2.6393
12	2.6437	24	2.6453

Table 4: Estimated μ_2 using SIS with SAW as instrumental distribution

Here, we are most often in the neighborhood of the true $\mu_2 \in [2.62, 2.68]$, which was expected given the reduced errors in table 3. Given the slight upward trend in error, these estimates may get worse over time. To test this, we simulate $1 \leq n \leq 150$ with $N = 10^4$ and estimate μ_d for $1 \leq n \leq 149$.

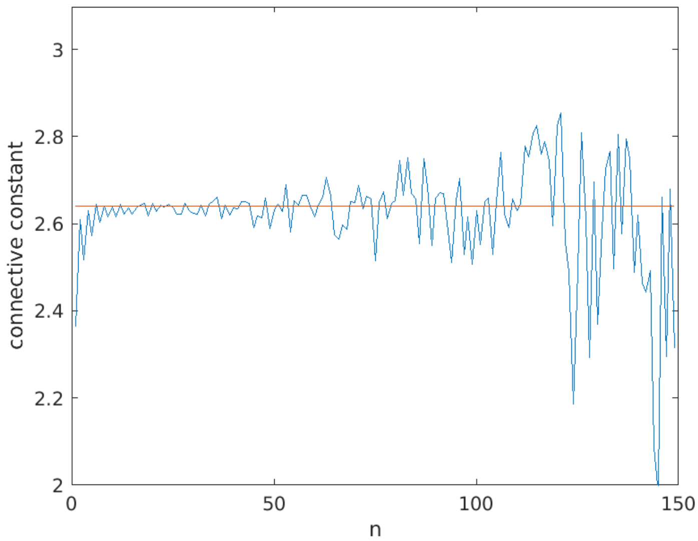

Figure 1: Simulated μ_d using SIS in blue with an approximately true $\mu_d \approx 2.64$ in orange

As we can see, our hypothesis about an upward trend in error seems to be true. For $n \geq 75$, the accuracy of the estimated values for the connective constant, and thus also for c_n, seems to deteriorate.

1.5

To see if we can improve the accuracy of our estimated connective constants for larger n, we implement sequential importance sampling with resampling (SISR), still using a self-avoiding walk as instrumental distribution. As described in lecture 7 [5], the idea behind SISR is that particles with large weights should be preserved and duplicated, whereas particles with smaller weights should not.

We approach the problem slightly differently compared to SIS. In each time step, we resample our particles by drawing new $\tilde{X}_1^{0:n}, ..., \tilde{X}_N^{0:n}$ with replacement from $X_1^{0:n}, ..., X_N^{0:n}$ with probability

$$\mathbb{P}(\tilde{X}_i^{0:n} = X_j^{0:n}) = \frac{\omega_n^j}{\sum_{l=1}^N \omega_n^l}$$

By doing this, we are more likely to keep particles with larger weights. The rest of the procedure is similar to the one presented in the previous exercise except that we draw $X_i^{0:n+1} \sim g_{n+1}(x_{n+1}|\tilde{X}_i^{0:n})$ and set $X_i^{0:n+1} = (X_i^{n+1}, \tilde{X}_i^{0:n})$. There are also two differences in how we handle the weights and do our estimations of $c_n(2)$:

1. A weight ω_n^i is no longer an approximation of the total amount of walks until that step. Instead, we set the weight for a certain particle and step to the amount of free neighbors, i.e. $\omega_i^{k+1} \leftarrow \#\text{free}$

2. Because of the different interpretation of the weights, we have to use a different estimator. We use that $c_{n+1}(2) \leftarrow c_n(2) \cdot \text{mean}(\omega_{n+1})$ with $c_0(2) = 1$.

We run a simulation for $1 \leq n \leq 150$ and $N = 10^4$

Estimates					
Steps n	$c_n(2)$	Error	Steps n	$c_n(2)$	Error
1	4	0%	13	882606	0.13%
2	12	0%	14	2390627	0.68%
3	36	0%	15	6489835	1.14%
4	100	0%	16	17393407	0.86%
5	284	0%	17	46880450	0.89%
6	779	0.13%	18	125175490	0.41%
7	2164	0.37%	19	336346542	0.37%
8	5869	0.79%	20	903359543	0.63%
9	16117	0.93%	21	2438438414	1.23%
10	43733	0.83%	22	6504778313	0.93%
11	119671	0.52%	23	17399631509	0.77%
12	324428	0.16%	24	46025505268	0.26%
Average Error: 0.5%					

Table 5: Estimated $c_n(2)$ using SISR with SAW as instrumental distribution

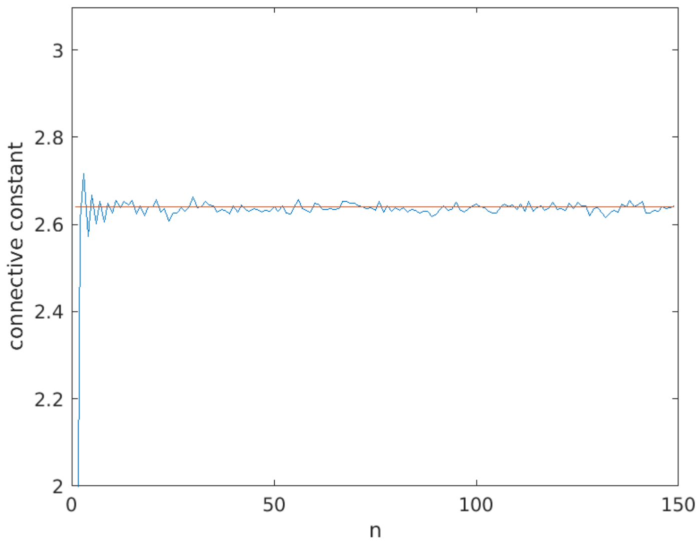

Figure 2: Simulated μ_d using SISR in blue with an approximately true $\mu_d \approx 2.64$ in orange

As we can see in table 5, our estimates are still a lot better for $n \leq 24$ compared to the naive approach in 1.3 (0.5% average error compared to 10.21%). However, the error is slightly larger for these small n compared to the approach presented in 1.4. We believe this is due to the increased variance that is introduced by the resampling, which intuitively adds some randomness to the model.

Considering figure 2, our estimates do not seem to deteriorate for larger n as was the case with regular SIS. The resampling alleviates the degeneration of the weights and gives us more accurate estimates. For smaller n, we may still prefer SIS, but in the general case SISR vastly outperforms SIS.

1.6

We would like to estimate the parameters in equation 1 by means of a linear regression. We can accomplish this by first rewriting 1 as

$$\ln(c_n(2)) = \ln(A_2) + n\ln(\mu_2) + (\gamma_2 - 1)\ln(n)$$

where n is known and we can obtain $c_n(2)$ by using SISR. We then have a regression task where we want to fit some β_1, β_2 and β_3 to

$$\begin{cases} \ln(c_1(2)) = \beta_1 + 1 \cdot \beta_2 + \ln(1)\beta_3 \\ \ln(c_2(2)) = \beta_1 + 2 \cdot \beta_2 + \ln(2)\beta_3 \\ \vdots \\ \ln(c_n(2)) = \beta_1 + n \cdot \beta_2 + \ln(n)\beta_3 \end{cases}$$

which can be solved by firstly defining a vector $\vec{y}$ containing all $\ln(c_n(2))$ and a matrix B with the coefficients in front of the betas, ultimately solving the system with `y\B` in Matlab. We then transform

$$\begin{aligned} A_2 &= e^{\beta_1} \\ \mu_2 &= e^{\beta_2} \\ \gamma_2 &= \beta_3 + 1 \end{aligned}$$

We repeat the above-mentioned process 10 times, each time running SISR with $n = 100$ and $N = 10^3$ to obtain estimates for $c_n(2)$ and doing a subsequent linear regression, which yields the following results

Parameter	Mean	Max	Min	Variance
A_2	1.4088	1.5309	1.2917	0.0066
μ_2	2.6404	2.6496	2.6322	$2.79 \cdot 10^{-5}$
γ_2	1.2897	1.3389	1.2466	0.0012

Table 6: Estimated parameters from linear regression

Considering the variances in table 6, it seems that A_2 is the hardest to estimate and μ_2 the easiest. From the relation in equation 1, we can see that A_2 grows the slowest out of all the parameters (linearly). This should make it the hardest to estimate, since fixing μ_2 and γ_2 requires a larger change in A_2 to compensate for changes in $c_n(2)$. On the other hand, μ_2 grows the fastest and by a similar argument it should therefore be the easiest to estimate.

We may now check the feasibility of the estimated parameters. We have that $\gamma_2 = \frac{43}{32} \approx 1.34$, which is reasonably close to the estimated γ_2. We also have that $\mu_2 \in [2.62, 2.68]$ from before, which our mean γ_2 is within. Since the relation in equation 1 applies as $n \to \infty$, we could of course use a larger n to obtain more accurate estimates.

1.7

We now want to verify that the following bound holds

$$d \leq \mu_d \leq 2d - 1$$

Consider a walk that is less strict than the self-avoiding walk where the only restriction is that we may not return to the position in the previous step. For this walk, we can of course go in $2d$ directions in the first step and $2d-1$ directions in the subsequent steps. The amount of possible such walks is then $2d(2d-1)^{n-1}$. For this type of walk, we can thus say that $\hat{c}_n(d) = 2d(2d-1)^{n-1}$ and consequently

$$\begin{aligned} \hat{\mu}_d = \lim_{n\to\infty} \hat{c}_n(d)^{1/n} = \lim_{n\to\infty} (2d(2d-1)^{n-1})^{1/n} = \lim_{n\to\infty} (\frac{2d(2d-1)^n}{2d-1})^{1/n} \\ = \lim_{n\to\infty} (\frac{2d}{2d-1})^{1/n}(2d-1) = 2d-1 \end{aligned}$$

Obviously, for a stricter self-avoiding random walk with $c_n(d)$ and μ_d, it holds that

$$c_n(d) \leq \hat{c}_n(d) \implies \mu_d \leq \hat{\mu}_d = 2d - 1$$

We have thus shown the upper bound. We can obtain the lower bound by considering a self-avoiding walk where we may only head in a positive direction. That is, we may only increment our position in any dimension in one step. The amount of possible such walks is d^n, since we may go in d directions in every step. Such a walk is obviously stricter than a regular self-avoiding walk and we thus have $d^n \leq c_n(d)$. Moreover,

$$d^n \leq c_n(d) \implies d \leq c_n(d)^{1/n} \implies d \leq \mu_d$$

$\square$

1.8

We also want to show that the following bound holds

$$A_d \geq 1 \text{ for } d \geq 5$$

We may recall that for $d \geq 5$ and as $n \to \infty$

$$c_n(d) \sim A_d \mu_d^n n^{\gamma_d - 1}$$

and also, as shown previously, that

$$c_{n+m}(d) \leq c_n(d)c_m(d)$$

which implies that

$$\begin{gathered} A_d\mu_d^{n+m}(n+m)^{\gamma_d-1} \leq A_d\mu_d^n n^{\gamma_d-1}A_d\mu_d^m m^{\gamma_d-1} = A_d^2\mu_d^{n+m}(nm)^{\gamma_d-1} \iff \\ A_d(n+m)^{\gamma_d-1} \leq A_d^2(nm)^{\gamma_d-1} \end{gathered} \tag{3}$$

where $n, m \in \mathbb{N}$. Per the instructions, $\gamma_d \geq 1$ for $d \geq 5$ and we thus have that $(nm)^{\gamma_d-1} \geq (n+m)^{\gamma_d-1}$ for **most** n and m. However, for some combinations like $n = m = 1$ or $n = 2,\ m = 1$, the opposite holds. Considering these opposite cases and that A_d does not depend on n and m, it must hold that

$$A_d \leq A_d^2 \implies |A_d| \geq 1$$

From the previous exercise we have that $1 \leq d \leq \mu_d$. We moreover have that $n, m \in \mathbb{N}$. In other words, all the factors in equation 3 are positive. Since $c_n(d)$ trivially is positive, which equation 3 approaches, it must follow that A_d also is positive and $A_d \geq 1$ for $d \geq 5$.

□

1.9

The next task at hand is to estimate the values A_d, μ_d, and γ_d using the SISR approach for some $d \geq 3$ via following relation:

$$c_n(d) = \begin{cases} A_d\mu_d^n n^{\gamma_d-1}, d = 1, 2, 3,\ d \geq 5 \\ A_d\mu_d^n \log(n)^{1/4},\ d = 4 \end{cases} \quad \text{as } n \to \infty$$

Since we have already estimated the same values for $d = 2$ in task 1.6, we will use the same principle to estimate the values this time as well. After having chosen $d = 5$ in the SISR, we achieve the following results from the regression:

$$\begin{aligned} A_5 &= 1.1410 \\ \mu_5 &= 8.8360 \\ \gamma_5 &= 1.0263 \end{aligned}$$

To further validate the reliability of these estimates, we can check whether they are within the bounds of the previous two problems. From section 1.7 we have that

$$d \leq \mu_d \leq 2d - 1$$

giving

$$5 \leq \mu_5 \leq 2 \cdot 5 - 1 \implies 5 \leq 8.8360 \leq 9$$

which confirms the feasibility of μ_5. Moving on, we use the following bound from section 1.8

$$A_d \geq 1 \text{ for } d \geq 5$$

to verify our A_5, and since $A_5 = 1.1410 \geq 1$ this too seems like a feasible estimation.

As a final check, we compare our μ_5 with the asymptotic bound on μ_d for large d found in Graham [3]

$$\mu_d \sim 2d - 1 - 1/(2d) - 3/(2d)^2 - 16/(2d)^3 - O(1/d^4)$$

which gives

$$\mu_5 \sim 2 \cdot 5 - 1 - 1/(2 \cdot 5) - 3/(2 \cdot 5)^2 - 16/(2 \cdot 5)^3 - O(1/5^4) \approx 8.854$$

While this value does not correspond exactly to our estimated μ_5, they are still well within an acceptable distance from each other. The difference could be explained by the number of steps ($n = 150$) and walks ($N = 10000$) used, as they may not be enough to simulate enough walks of certain lengths to give more precise estimates. This also applies to our $\gamma_5 = 1.0263$, which per the instructions should be equal to 1. Considering the computational power at hand we would like to deem our results, for all three estimates, as precise enough. If this self-avoiding walk model is to be applied for some larger cause however, one might want to increase the amount of steps and walks.

2 Filter Estimation of Noisy Population Measurements

In this second assignment we will investigate the growth and decline of a population for some organism throughout k generations. The size of the population in generation k will be described as X_k, having a simplified model of relative population following

$$X_{k+1} = B_{k+1} X_k (1 - X_k)$$

where $B_{k+1} \in U(A, B)$ with $A = 0.9$ and $B = 3.9$. For the first generation, X_0, we assume that $X_0 \in U(C, D)$ with $C = 0.6$ and $D = 0.99$

Since it is difficult to measure the relative population precisely, we instead get a measurement from

$$Y_k | X_k = x \in U(Gx, Hx)$$

where $G = 0.7$ and $H = 1.2$.

This can be described as a hidden Markov model, setting the hidden Markov chain as our relative population size X. With this, we can estimate the filter expectation $\tau_k = E[X_k|Y_{0:k}]$ for $k = 0, 1, 2, ..., n$.

The relative population sizes are set between zero and one, with zero indicating that the population has become extinct and one indicating the maximum size relative to the carrying capacity of the environment. If the the population size is closer to being zero, there will be more resources in the environment to be consumed resulting in a larger next generation, and vice versa for being closer to one.

2.1

In this first section, we estimate the filter expectation τ_k for $k = 0, 1, 2, ..., 50$. To achieve this, we implement the SISR algorithm, setting

$$\sum_{i=1}^{N} \frac{w_k^i}{\sum_{l=1}^{N} w_k^l} \phi(\tilde{X}_i^{0:k})$$

as estimator for τ_k. Using this while implementing the algorithm found on slide 24 of lecture 7 written by Wiktorsson [1], exchanging the observation density, initialization, and mutation to the above distributions, we achieve following results:

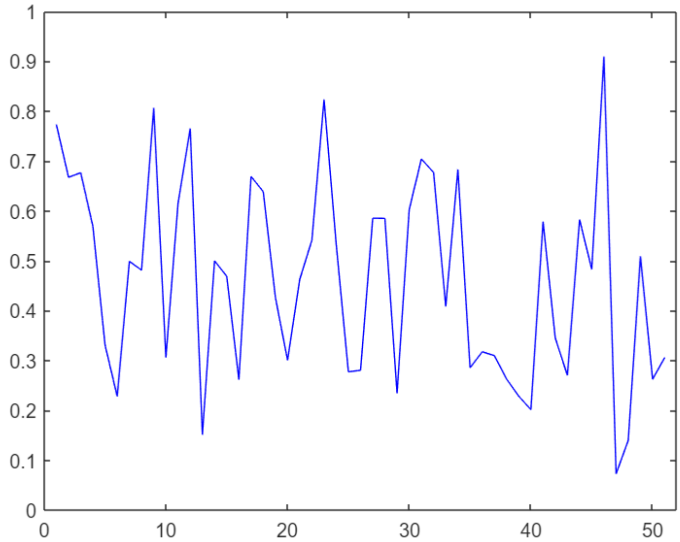

Figure 3: Filter expectation

Figure 3 shows the change in τ_k throughout the generations, indeed bobbing up and down between zero and one depending on how many resources there are to consume.

2.2

Continuing on, we wish to make a point wise 95% confidence interval of X_k for $k = 0, 1, 2, ..., 50$, checking whether our true X_k remain within the bounds. We implement this using the additional code found on slide 25 lecture 7 written by Wiktorsson [2].

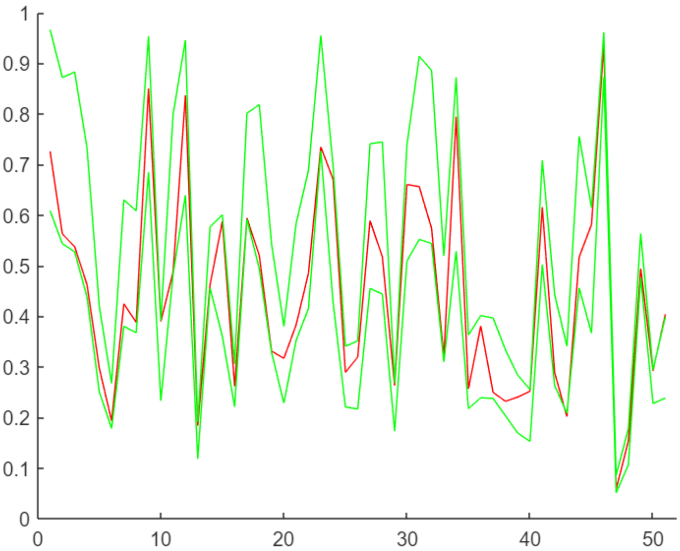

Figure 4: Confidence interval

Figure 4 shows the resulting graph, where the green lines are our confidence interval bounds and the red line being the true X_k. While the confidence interval is rather tight at times, our hidden values seems to remain within the bounds. This indicates that the estimated values τ_k are reliable, and can be used as an estimate to the relative population size.

References

[1] M. Wiktorsson, *Example: Linear/Gaussian HMM, SISR implementation*, February 2023, lecture 7, slide 24

[2] M. Wiktorsson, *Approximative confidence intervall for the hidden state*, February 2023, lecture 7, slide 25

[3] Graham, BT. (2018) *Borel-type bounds for the self-avoiding walk connective constant*, Available at: http://arxiv.org/pdf/0911.5163.pdf

[4] M. Wiktorsson, *Last time: Sequential importance sampling (SIS)*, February 2023, lecture 7, slide 6-8

[5] M. Wiktorsson, *Multinomial resampling*, February 2023, lecture 7, slide 19

[6] Slade, Gordon. (2011) *The self-avoiding walk: A brief survey*, Surveys in Stochastic Processes, 181–199. Available at: https://www.math.ubc.ca/∼slade/spa_proceedings.pdf

[7] Wolfram MathWorld, *Self-Avoiding Walk Connective Constant*. Available at https://mathworld.wolfram.com/Self-AvoidingWalkConnectiveConstant.html