Monte Carlo

Variance Reduction

Francesco Banterle, Ph.D.

Variance Reduction

Introduction

- Main techniques:
  - Antithetic Sampling
  - Stratification
  - Russian Roulette
  - Importance Sampling
  - Metropolis Sampling

Antithetic Sampling

Antithetic Sampling
Main Idea

- Monte-Carlo leads to error cancellation; and this is our aim when we do antithetic sampling; i.e., we are trying to balance samples with their opposites.

- We look for a value of $f(\mathbf{x})$ that gives us an opposite value $\mathbf{x}^\star$; one time low and the other high.

- How?

  - If the $p(\mathbf{x})$ is symmetric (e.g., the uniform), we can generate $\mathbf{x}^\star$ as:

$$\mathbf{x}^\star = 2\mathbf{c} - \mathbf{x},$$

where $\mathbf{c}$ is the center point of the domain.

Antithetic Sampling
Main Idea

- The estimate, for averages, changes into:

$$\hat{\mu}_n^{\star} = \frac{1}{n}\sum_{i=1}^{\frac{n}{2}}\left(f(\mathbf{x}_i) + f(\mathbf{x}_i^{\star})\right),$$

  where n is even.

- The variance here is defined as:

$$\text{Var}(\hat{\mu}_n^{\star}) = \text{Var}\left(\frac{1}{n}\sum_{i=1}^{\frac{n}{2}} f(\mathbf{x}_i) + f(\mathbf{x}_i^{\star})\right) = \frac{\sigma^2}{n}\left(1+\rho\right) \qquad \rho \in [-1,1].$$

- ρ is the correlation between $f(\mathbf{x})$ and $f(\mathbf{x}^{\star})$. Note that in the best case, $\rho = -1$, we have the exact answer, otherwise in the worst case, $\rho = 1$, we doubled the variance!

Antithetic Sampling

Example: Integration

- Let's integrate

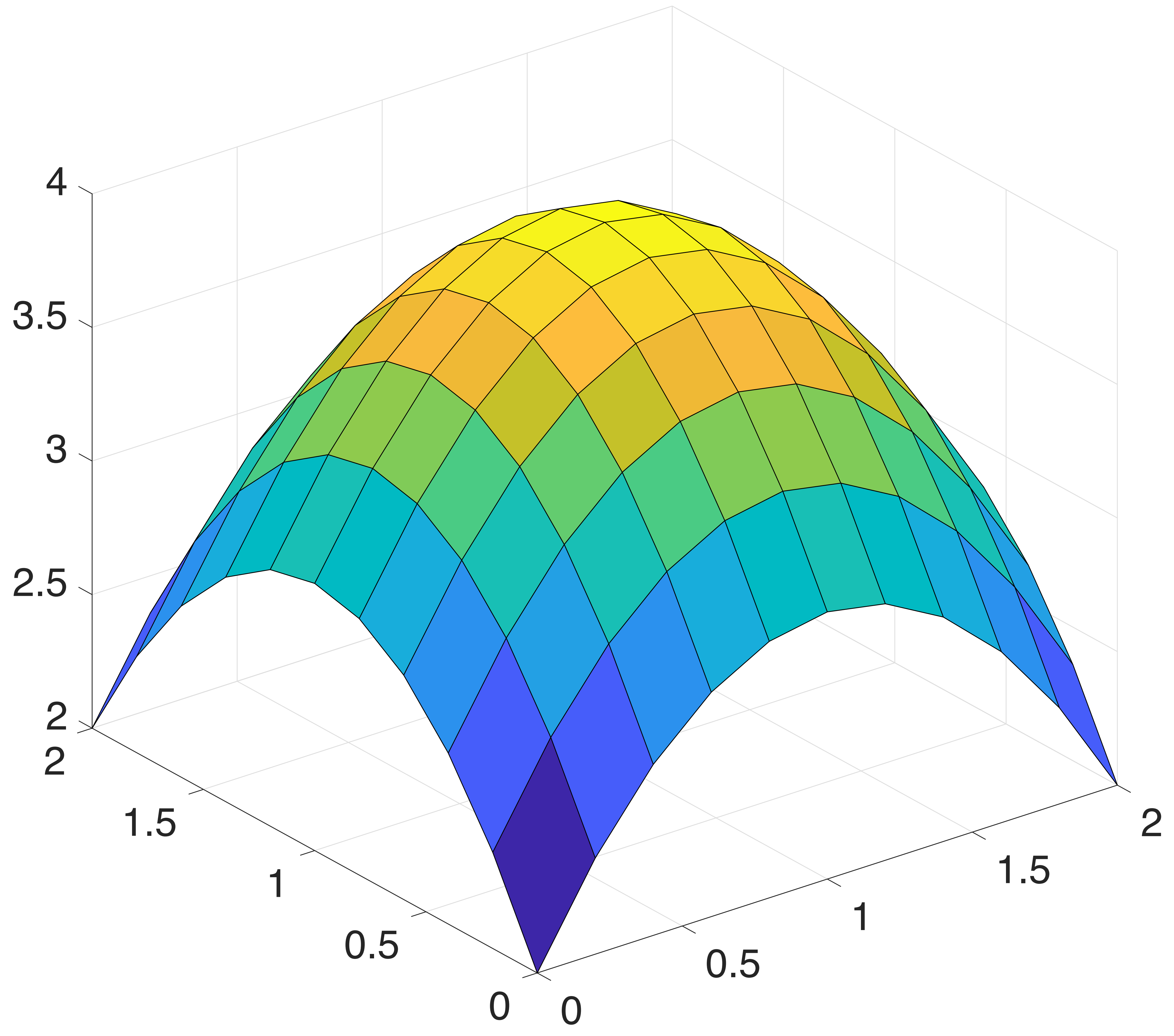
$$f(\mathbf{X}) = -\left((x_1 - c_1)^2 + (x_2 - c_2)^2\right) + 4 \qquad \mathbf{x} \in [0,2]^2, \quad \mathbf{c} = (1,1).$$

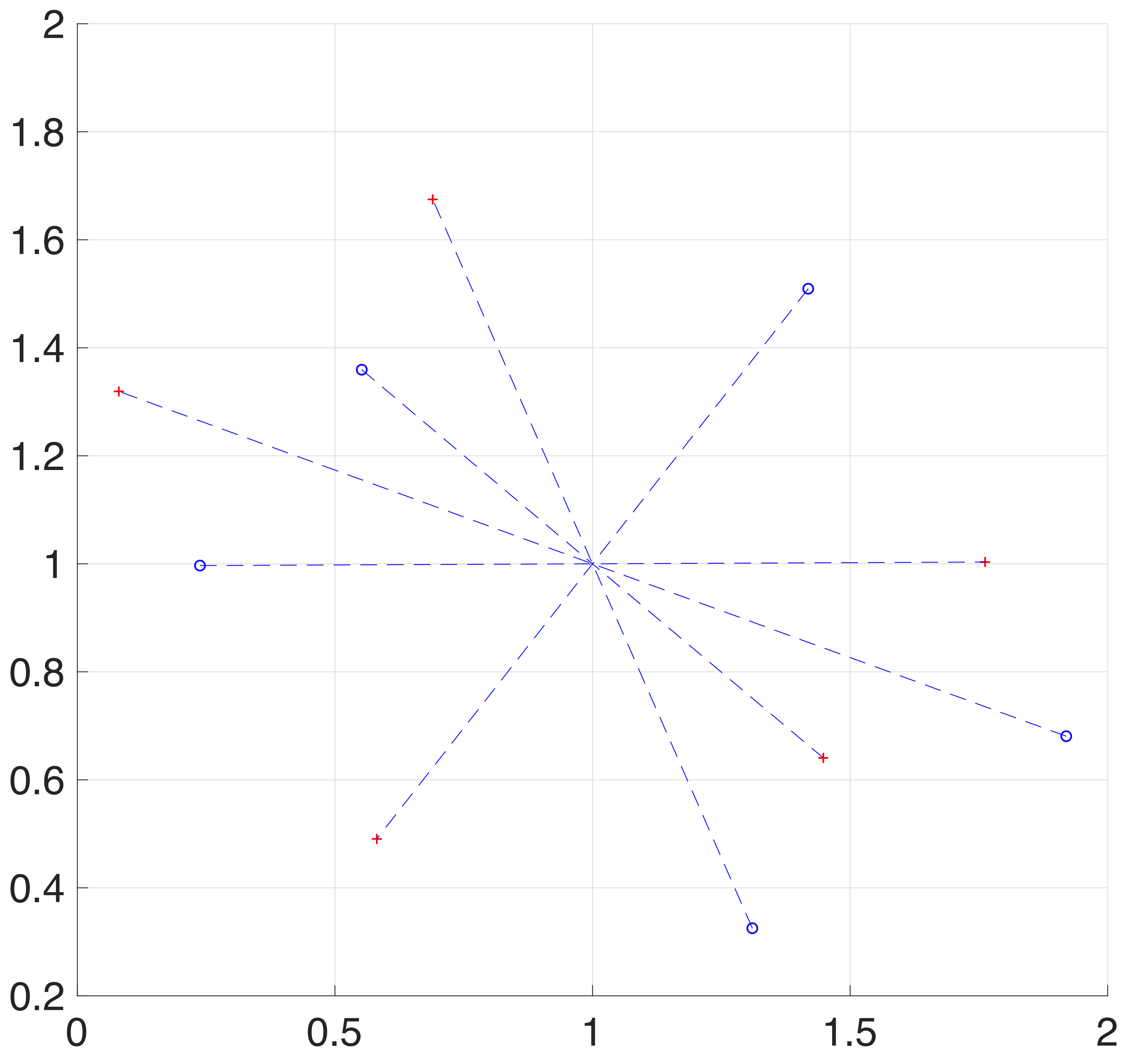
- We create antithetic samples as:

$$\mathbf{x}_i^\star = 2\mathbf{c} - \mathbf{x}_i = (2,2) - \mathbf{x}_i$$

Antithetic Sampling

Example: Integration





Antithetic Sampling

Example: Stocks

- Let's assume we have a return of a portfolio, $\mathbf{X}$, with n stocks with investment proportional to $\alpha_i \geq 0$ (they are normalized):

$$f(\mathbf{X}) = \log\left(\sum_{i=1}^{n} \alpha_i \exp(X_i)\right).$$

- Let's assume that equally invested in each stock; i.e., $\alpha_i = n^{-1}$, and $X_i \sim N(\mu = 0.001, \sigma = 0.03)$. We create antithetic samples as:

$$X_i^{\star} = 2\mu - X_i = 0.002 - X_i$$

Stratification

Stratification

Main Idea

- This strategy is the follow:
  - To split the domain of X into different regions.
  - To sample points in each region
  - To combine the results of each region; e.g., to estimate $\mathbb{E}(f(X))$.
- If each region get an equal number of samples, we should improve the quality of our estimate.

Stratification

Main Idea

- So our goal is to compute:

$$\mathbb{E}(f(\mathbf{X})) = \int_{\Omega} f(\mathbf{x})p(\mathbf{x})d\mathbf{x}.$$

- We partition Ω into K regions, $\Omega_1, \ldots, \Omega_K$, where:

$$\omega_j = P(\mathbf{X} \in \Omega_j) \text{ where } p_j(\mathbf{x}) = \omega_j^{-1} p(\mathbf{x}) 1_{\mathbf{x} \in \Omega_j}$$

- For the j-th region, we generate n_j samples, $\mathbf{X}_{j,1}, \ldots \mathbf{X}_{j,n_j}$, according to $p_j(\mathbf{x})$.

$$\hat{\mu} = \sum_{j=1}^{K} \frac{\omega_j}{n_j} \sum_{i=1}^{n_j} f(\mathbf{X}_{i,j}).$$

Stratification

Main Idea

- This sampling is unbiased:

$$\mathbb{E}(\hat{\mu}) = \sum_{j=1}^{K} \omega_j \mathbb{E}\left(\frac{1}{n_j}\sum_{i=1}^{n_j} f(\mathbf{X}_{j,i})\right) = \sum_{j=1}^{K} \omega_j \int_{\Omega_j} f(\mathbf{x}) p_j(\mathbf{x}) d\mathbf{x}$$

$$= \sum_{j=1}^{K} \int_{\Omega_j} f(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} = \int_{\Omega} f(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} = \mu$$

Stratification

Main Idea

- A typical allocation for n_j is proportional to ω_j:

$$n_j = \lceil n\omega_j \rceil.$$

- This leads to:

$$\hat{\mu} = \frac{1}{n}\sum_{j=1}^{K}\sum_{i=1}^{n_j} f(\mathbf{X}_{i,j}).$$

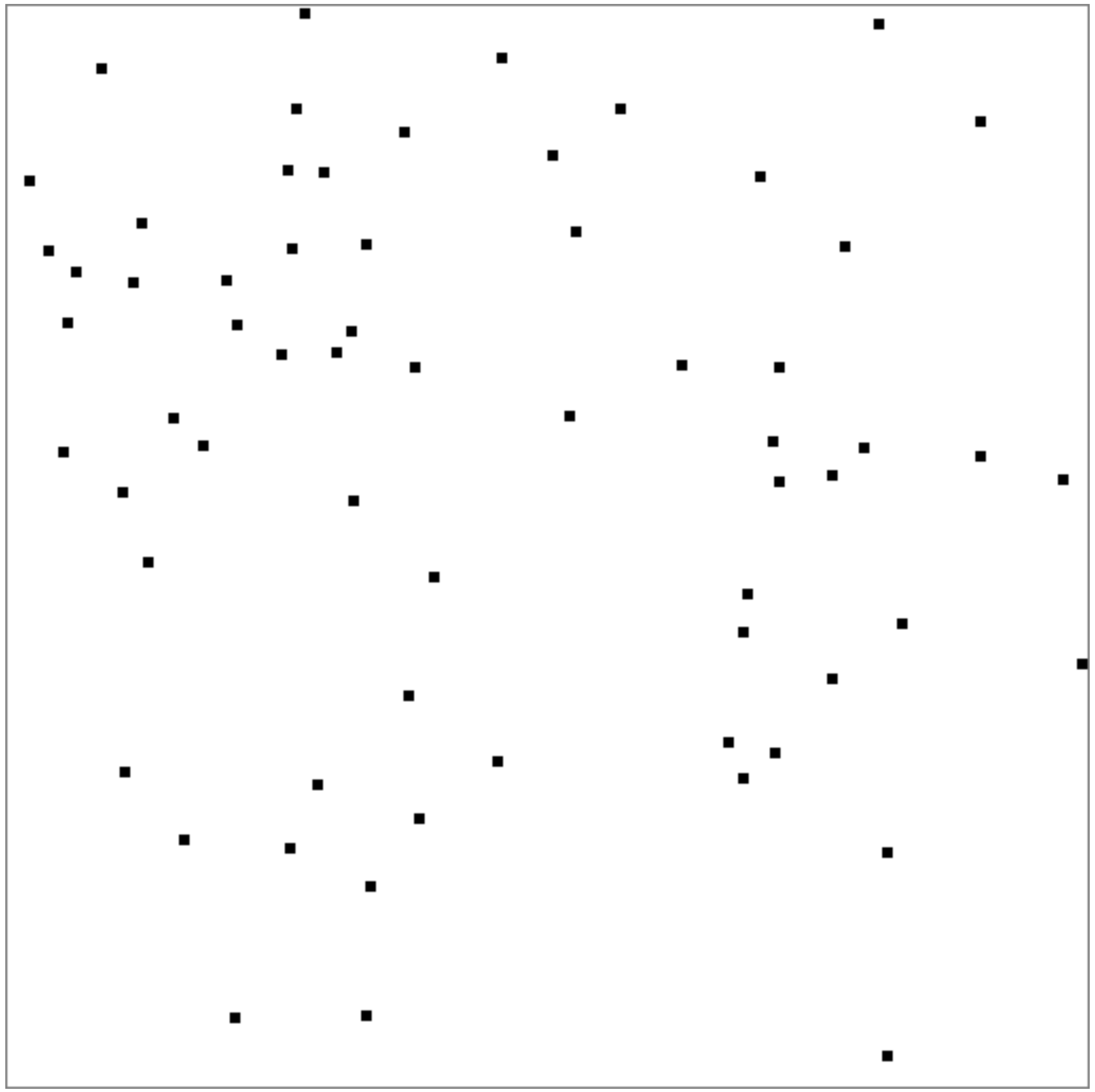
- Note that:

$$\hat{\mu}_j = \frac{1}{n_j}\sum_{i=1}^{n_j} Y_{i,j} \qquad s_j^2 = \frac{1}{n_j - 1}\sum_{i=1}^{n_j}(Y_{i,j} - \hat{\mu}_j)^2 \qquad \hat{\text{Var}}(\hat{\mu}) = \sum_{i=1}^{K}\omega_j^2\frac{s_j^2}{n_j}.$$

- This means that: $\hat{\mu} \pm 2.58\sqrt{\hat{\text{Var}}(\hat{\mu})}$.

Stratification

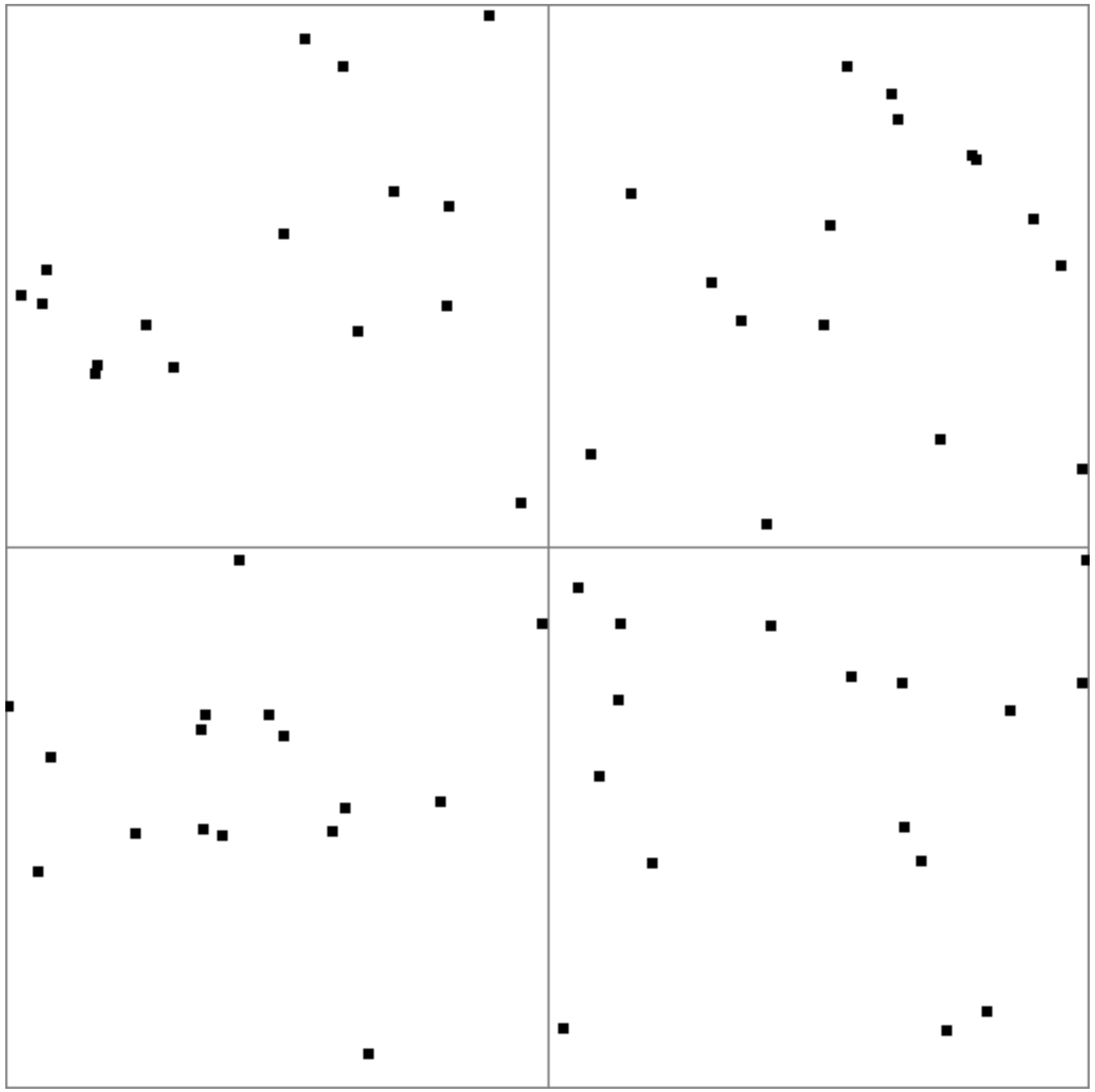
Example



$$n = 64$$
$$s = 1$$
$$n_s = 64$$

Stratification

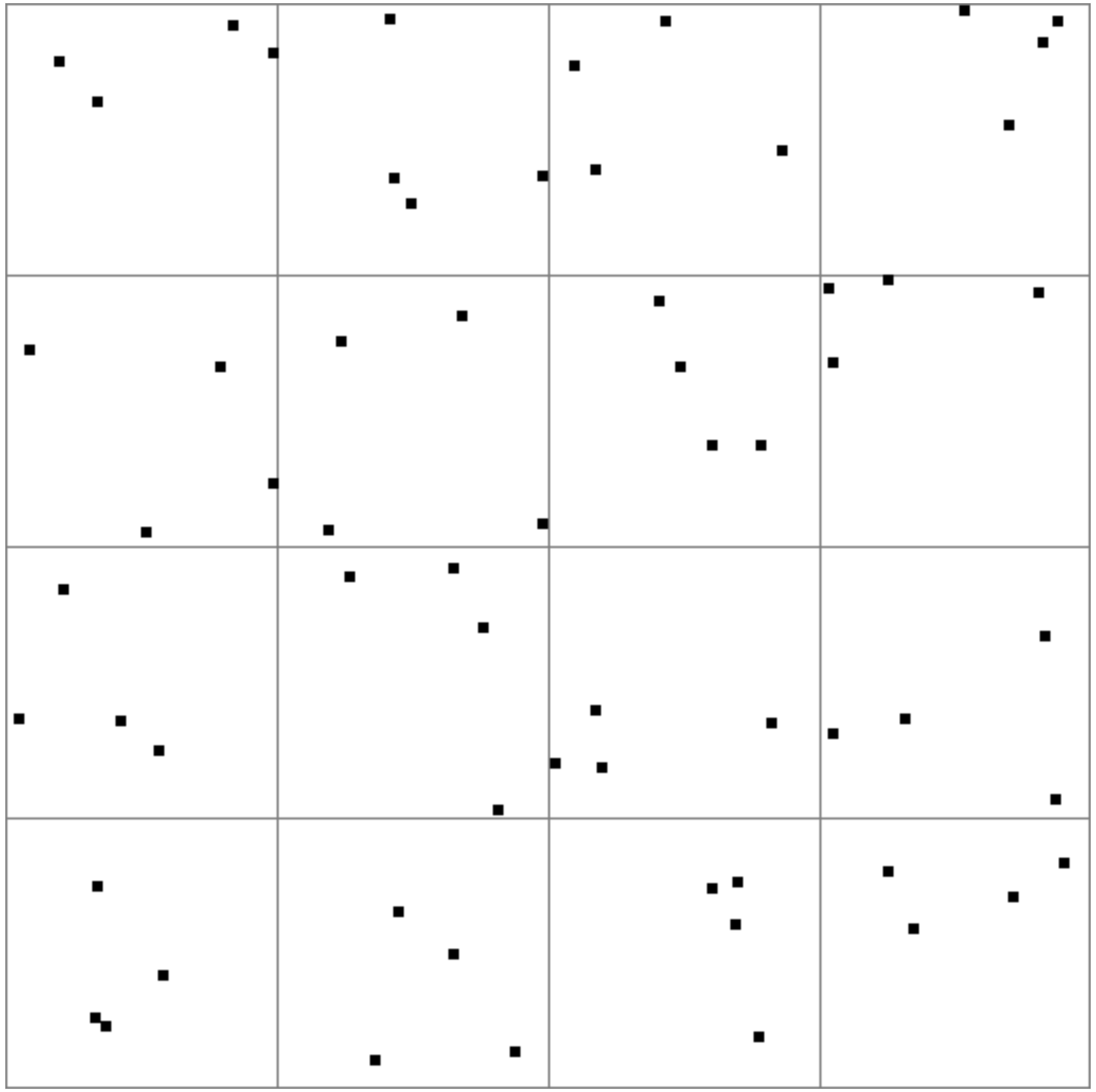
Example



$n = 64$
$s = 4$
$n_s = 16$

Stratification

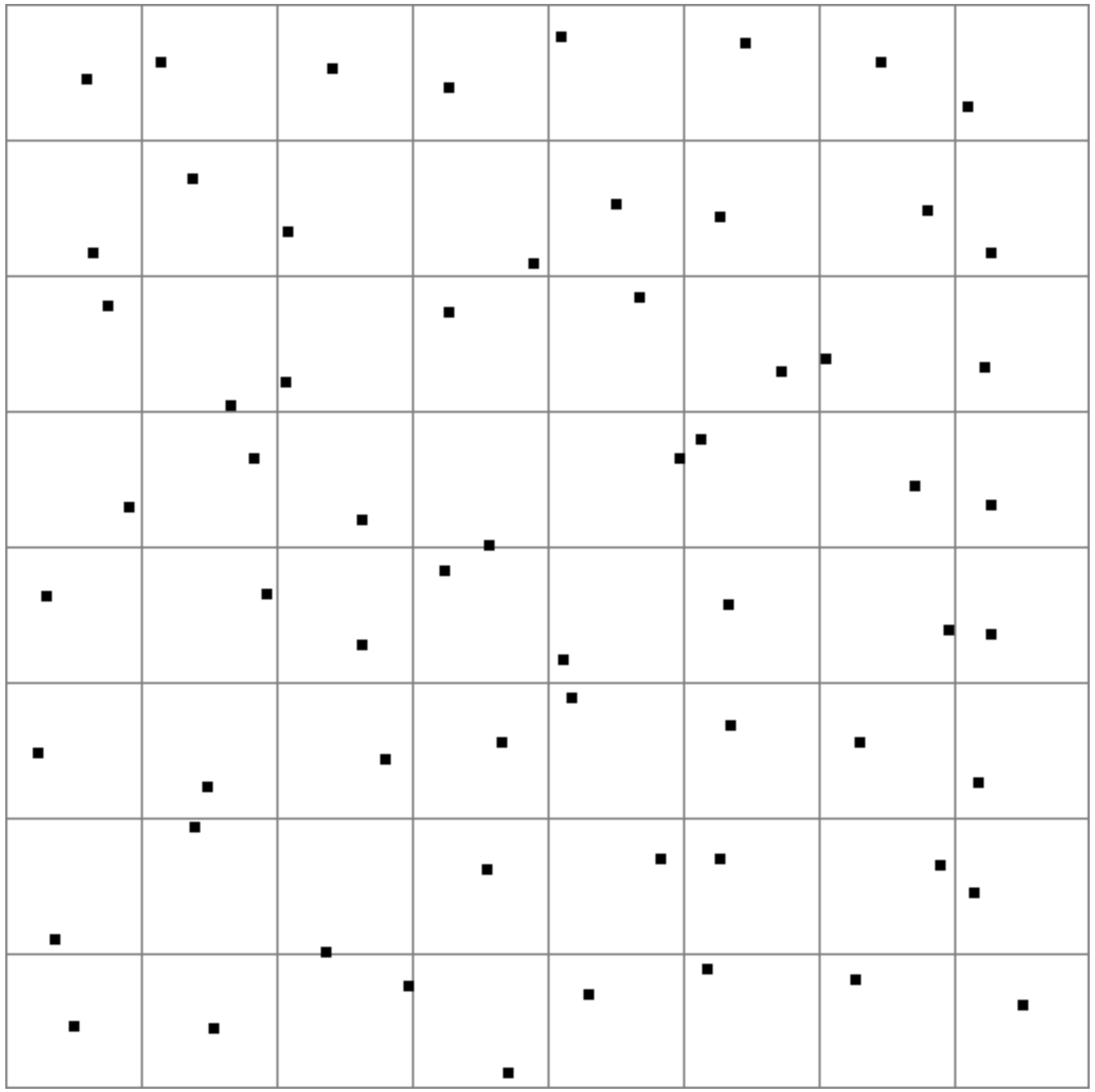
Example



$n = 64$
$s = 16$
$n_s = 4$

Stratification

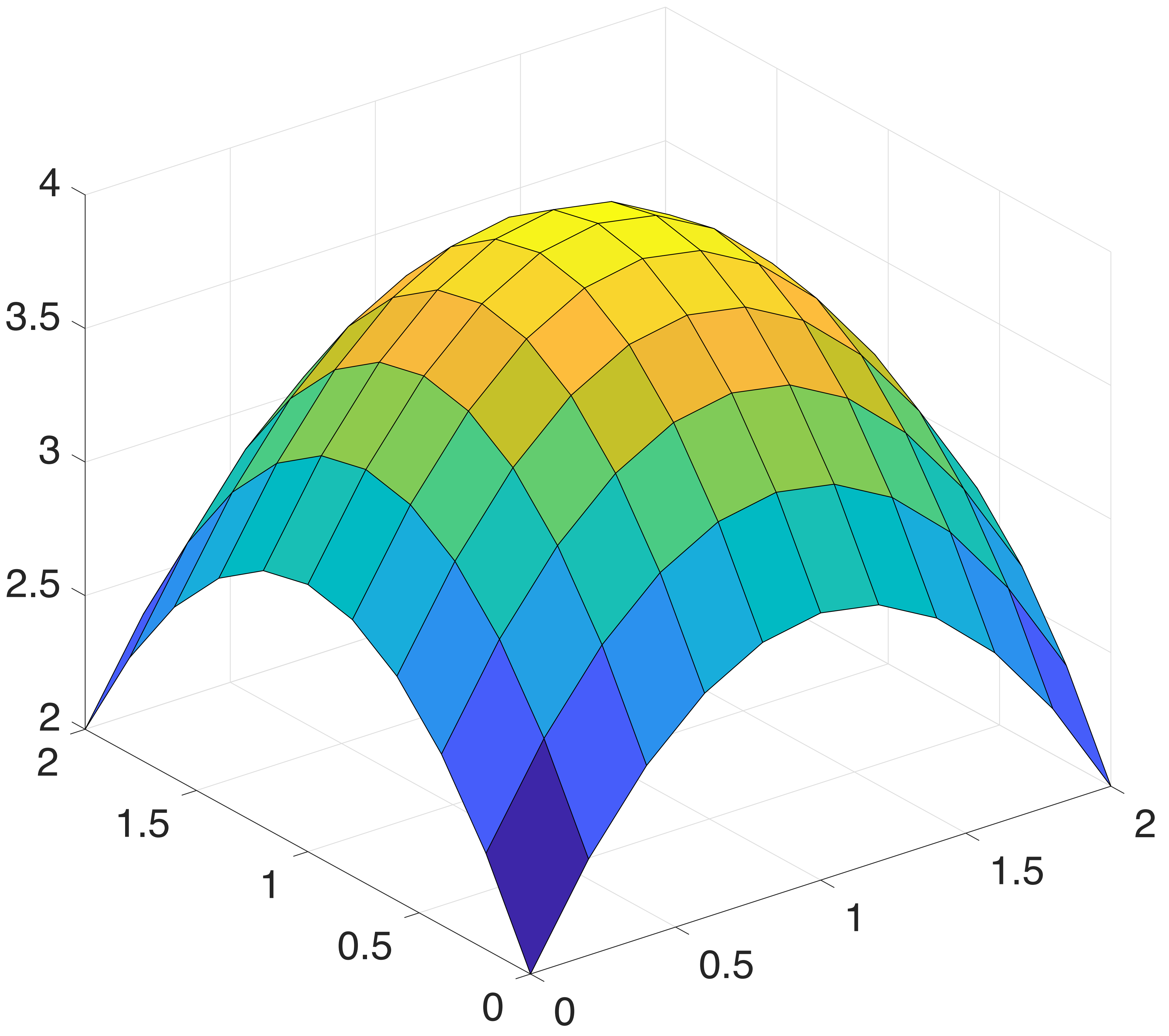
Example

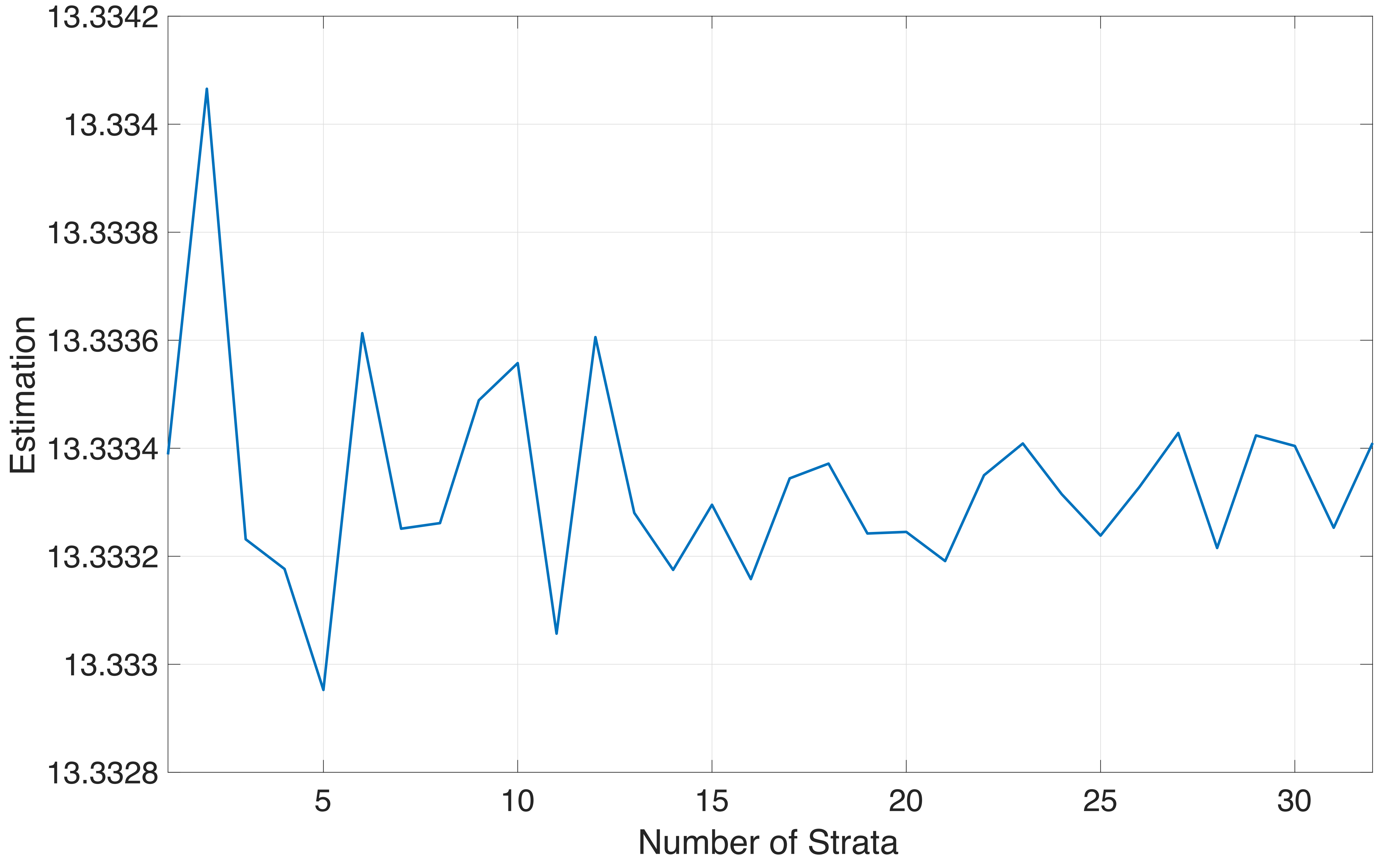


$n = 64$
$s = 64$
$n_s = 1$

Stratification

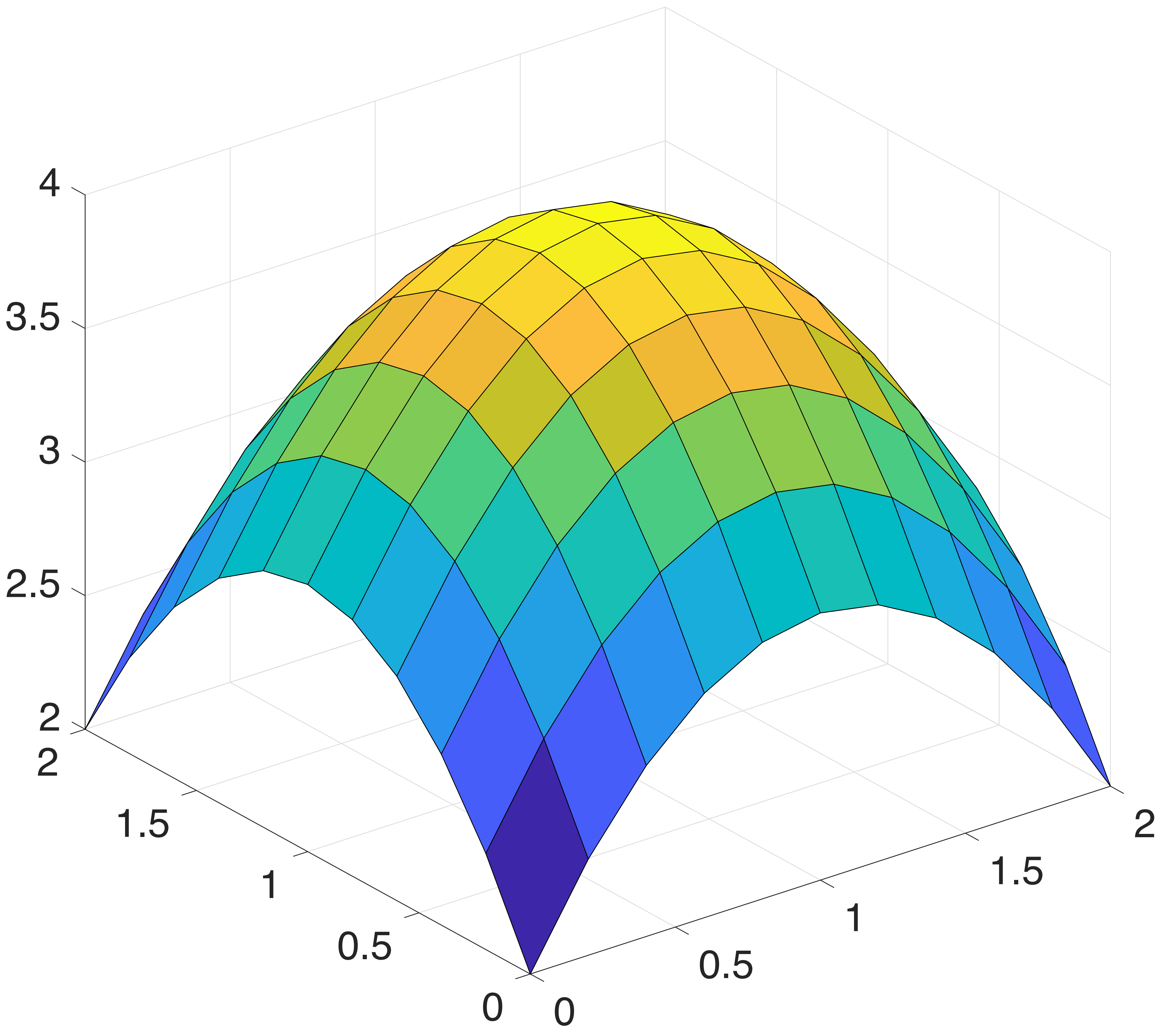
Example

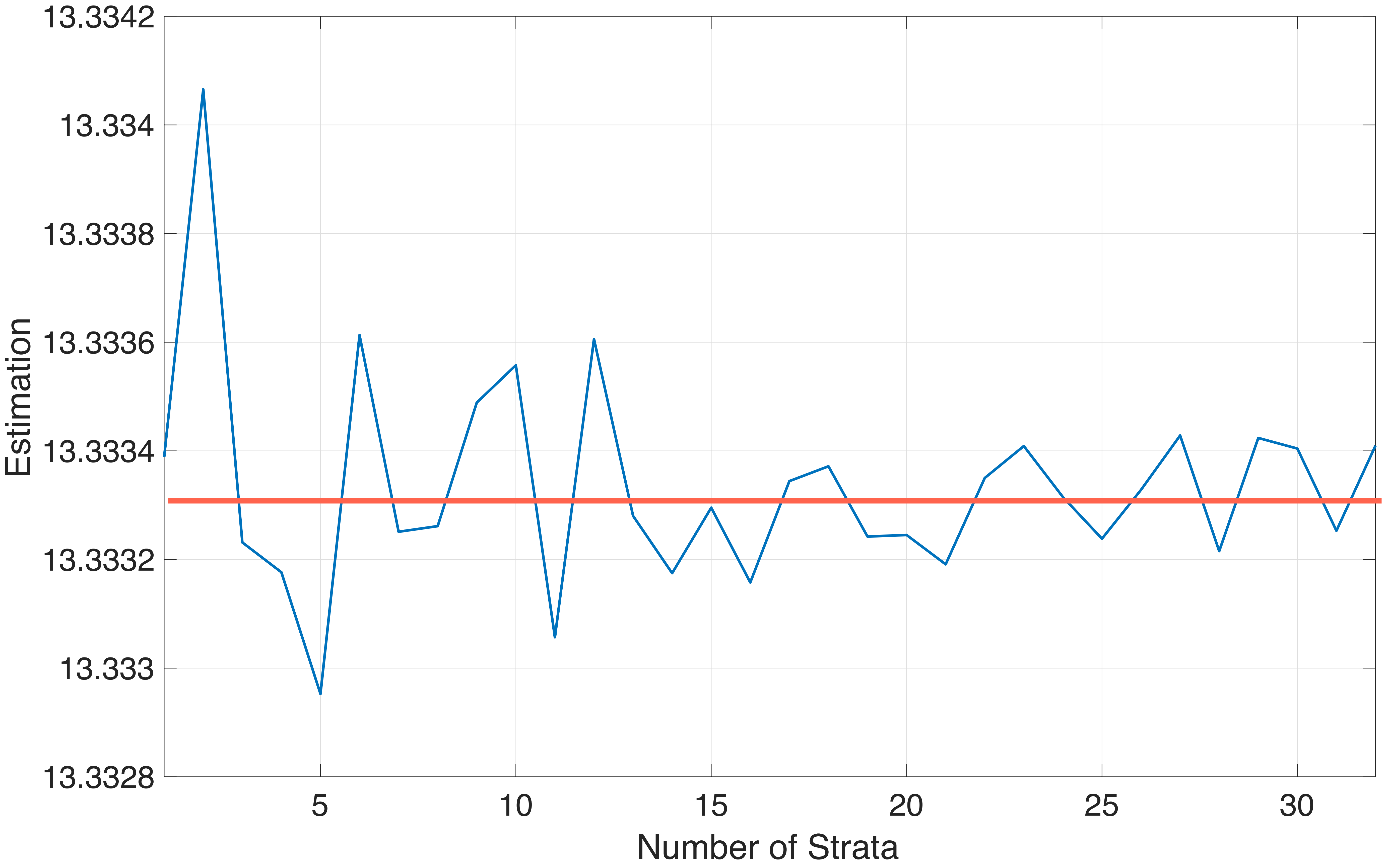




Stratification

Example





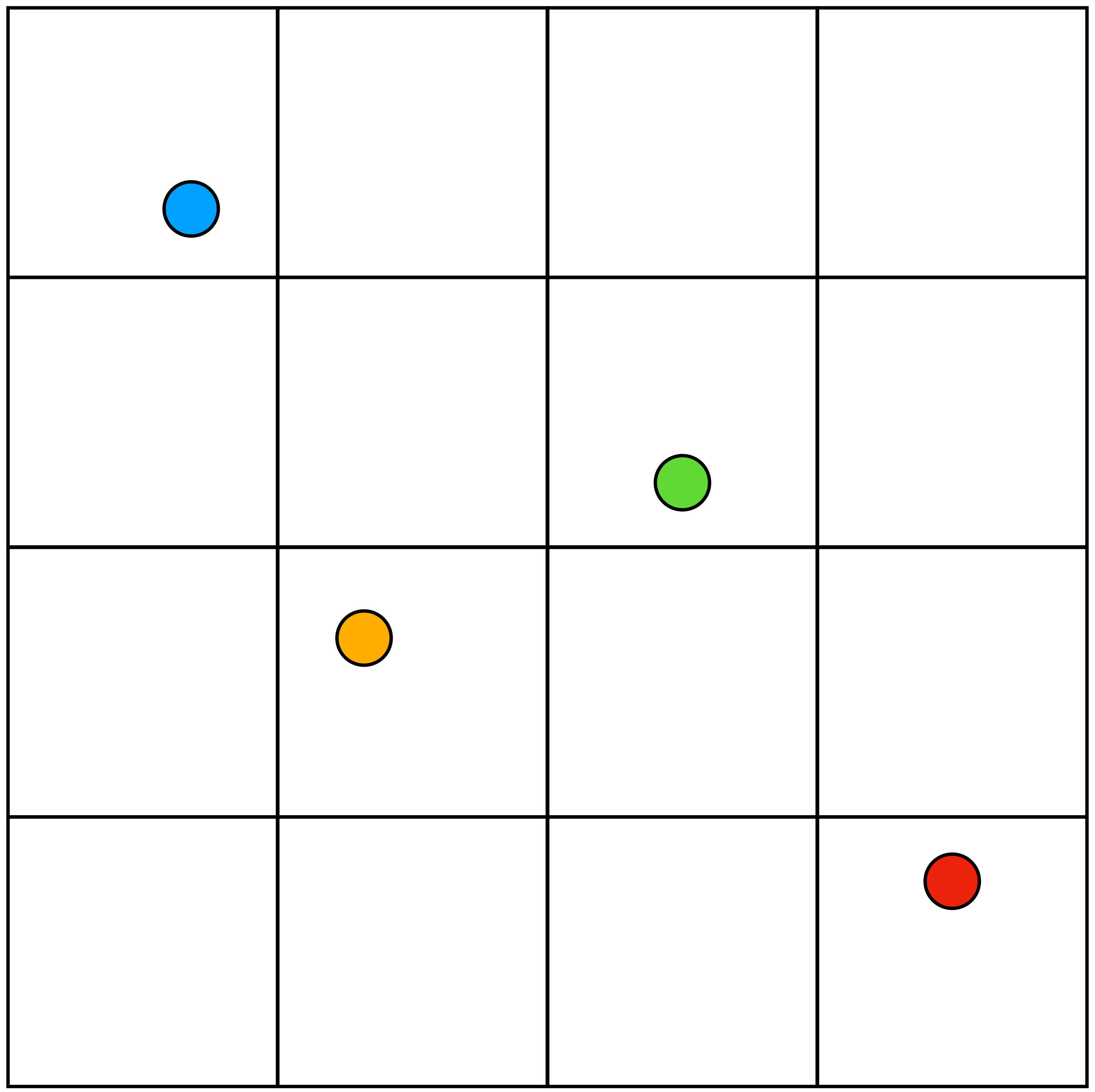
n-Rooks Sampling

n-Rooks Sampling
Main Idea

- Stratification has an issue; the curse of dimensionality:
  - If we divide our domain with sub-cubes with side $1/m$ and we place just a sample per cube, we need $n = m^d$ samples.
  - This becomes a very large numbers when we start to have many dimensions in our problem!
  - This is a similar problem that we have when dealing with regular grids.
  - A solution is to draw a sample for each dimension component for $\mathbf{X} \sim \mathbf{U}(0,1)^d$.

n-Rooks Sampling

Main Idea



n-Rooks Sampling

Main Idea

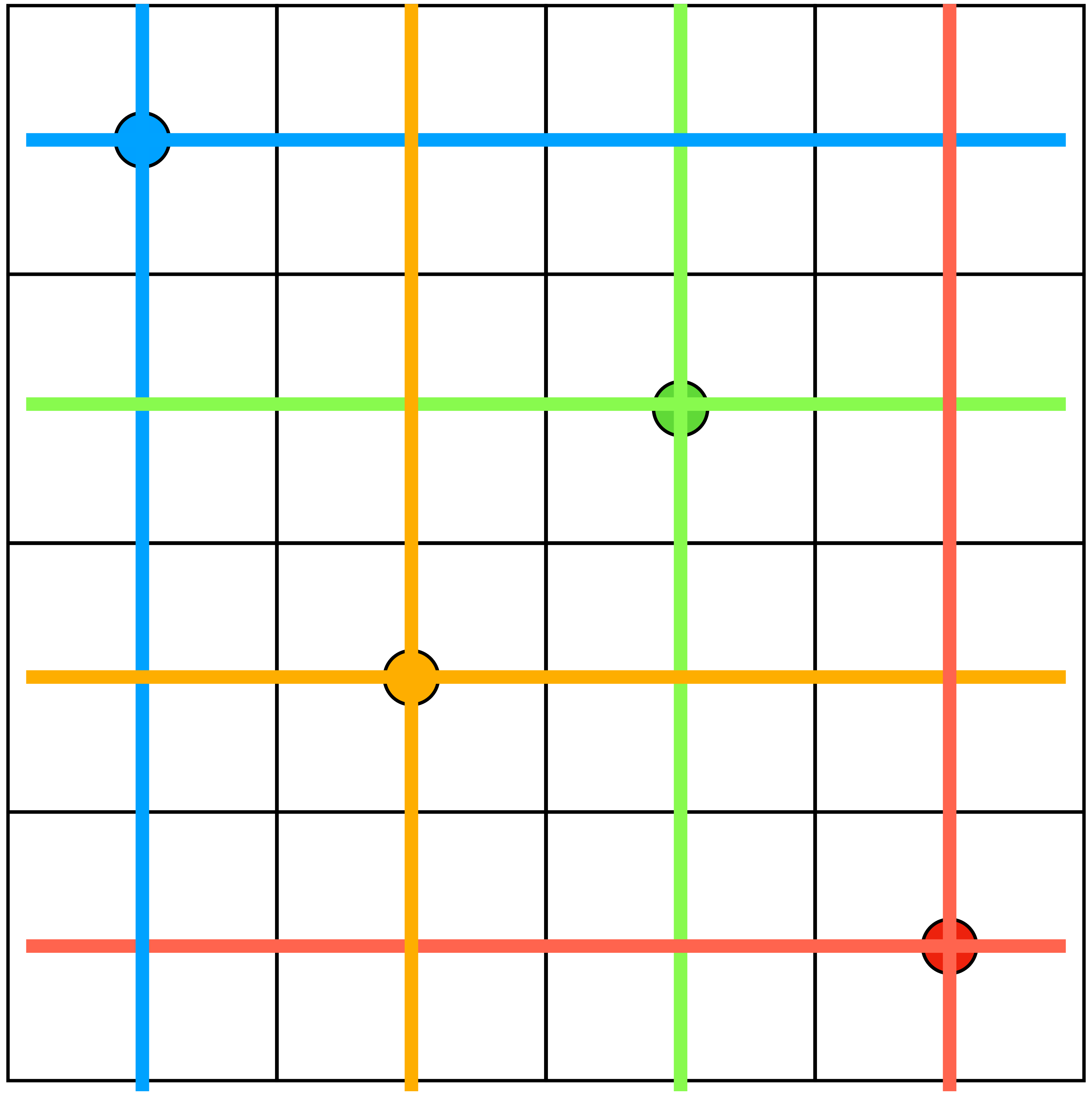
- How do we generate samples?

$$X_{i,j} = \frac{1}{n}\left(\pi_j(i-1) + U_{i,j}\right) \qquad i \in [1,n], j \in [1,d]$$

- Where π_j is a uniform random permutations of the set $\{0,\ldots,n-1\}$ and $U_{i,j} \sim \mathbf{U}[0,1)$.
- The name comes from a chess analogy: we place a sample as it were a rook controlling the rows and columns where it is placed on the chess board.
    - This method has been discovered several times in different fields and it has different names: Latin Hypercube Sampling, Lattice Sampling, etc.

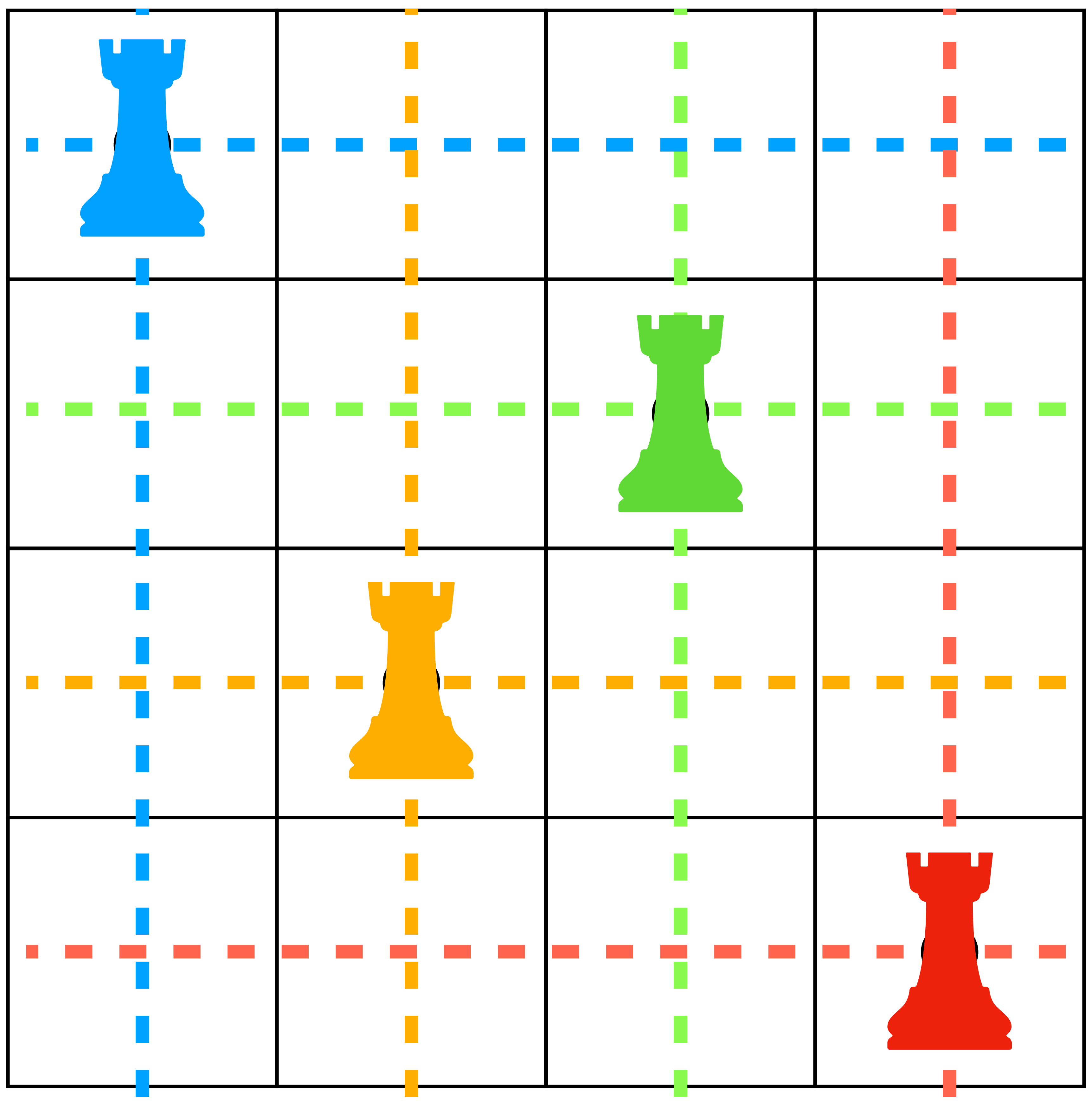
n-Rooks Sampling

Main Idea



n-Rooks Sampling

Main Idea



n-Rooks Sampling
Main Idea

- This sampling works best when we have additive functions:

$$f_a(\mathbf{x}) = f_0 + \sum_{i=1}^{d} f_i(x_i).$$

- This means that its variance is:

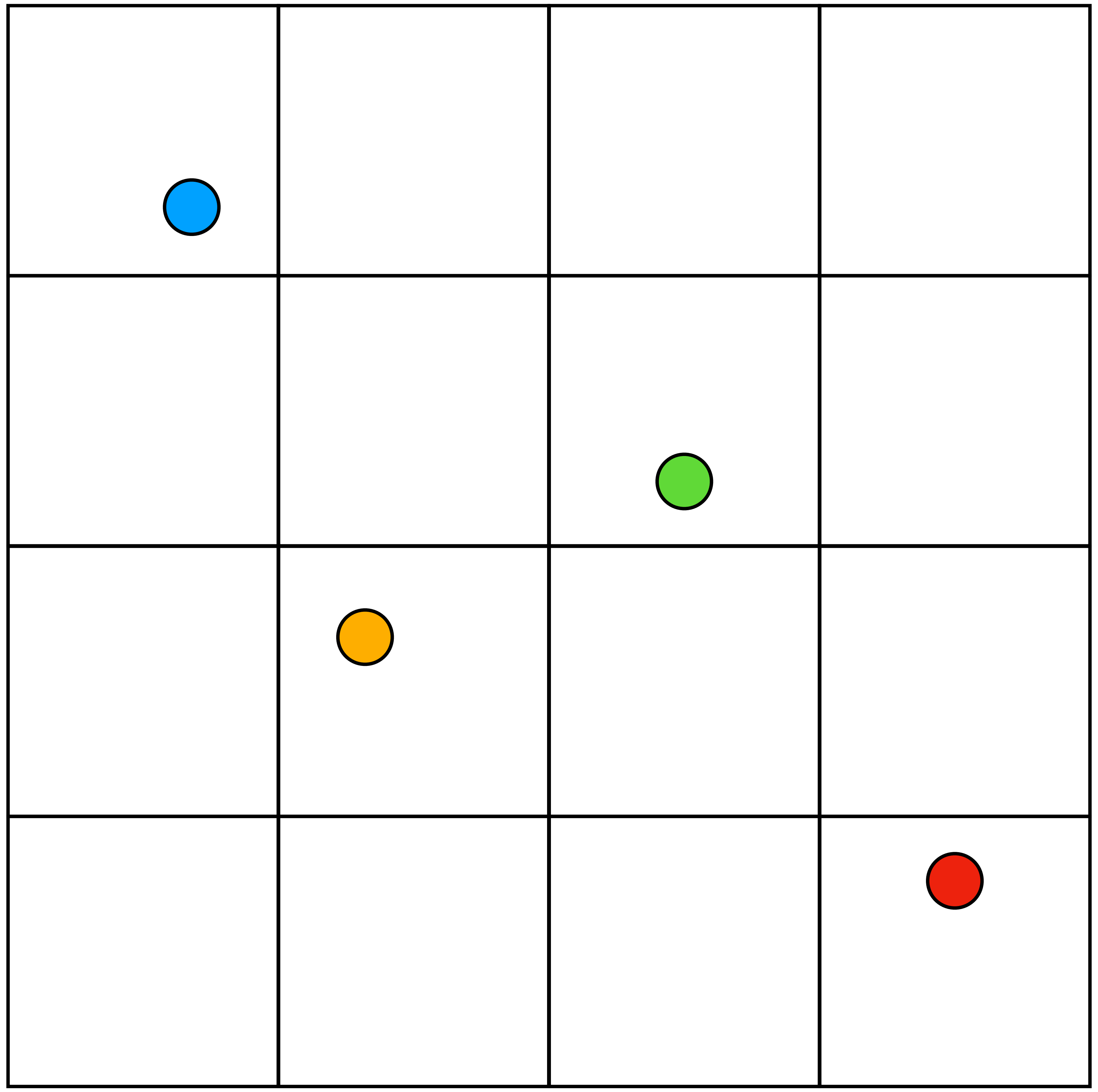
$$\text{Var}(\hat{\mu}) = \frac{1}{n}\int r(\mathbf{x})^2 d\mathbf{x} \leq \frac{\sigma^2}{n-1} + o\left(\frac{1}{n}\right) \qquad r(\mathbf{x}) = f(\mathbf{x}) - f_a(\mathbf{x}).$$

- In the worst case, this sampling increases the variance by:

$$\frac{n}{n-1}.$$

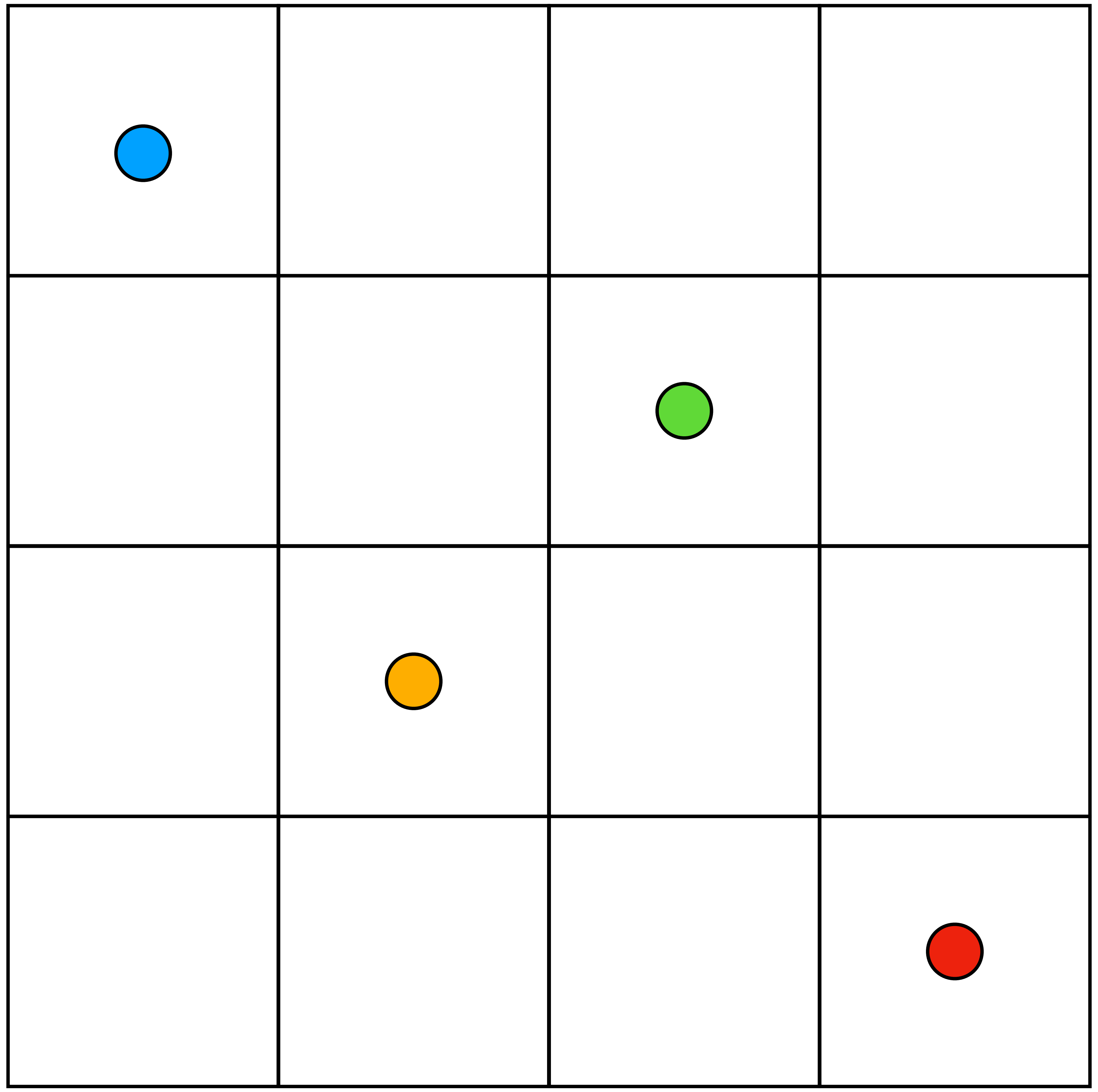
n-Rooks Sampling

Main Idea



n-Rooks Sampling

Main Idea



Russian Roulette

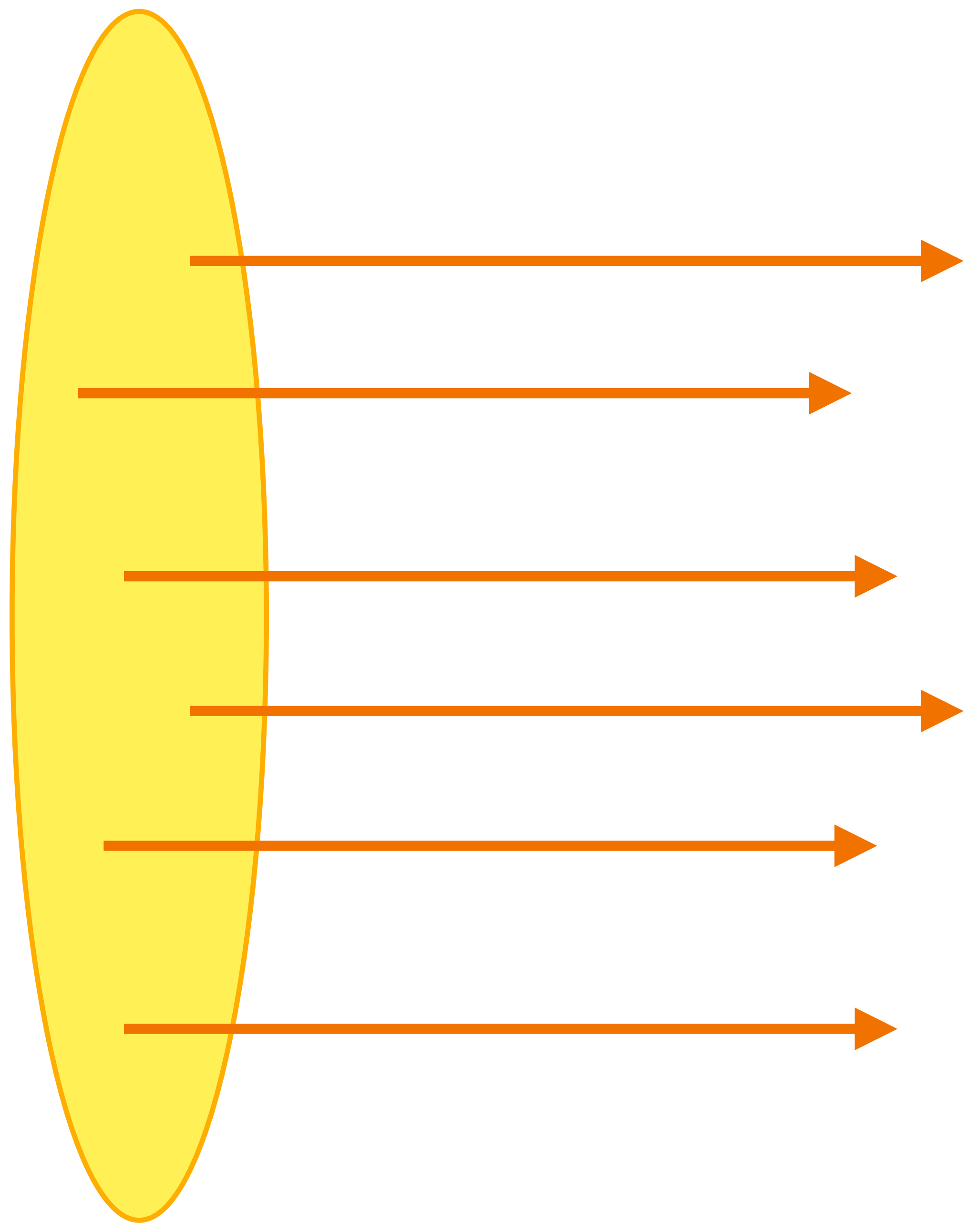
Russian Roulette

Main Idea

- Von Neumann and Ulam introduced this method that removes samples with low probability.
- Firstly, we need to split our domain into n sub-regions.
- For each sub-region, we need to know the probability, p_i, of that region to be picked.
- We generate a sample, $\mathbf{x}_i$, for that region with probability p_i.
- NOTE: This method can be used in both integration and simulations.

Russian Roulette

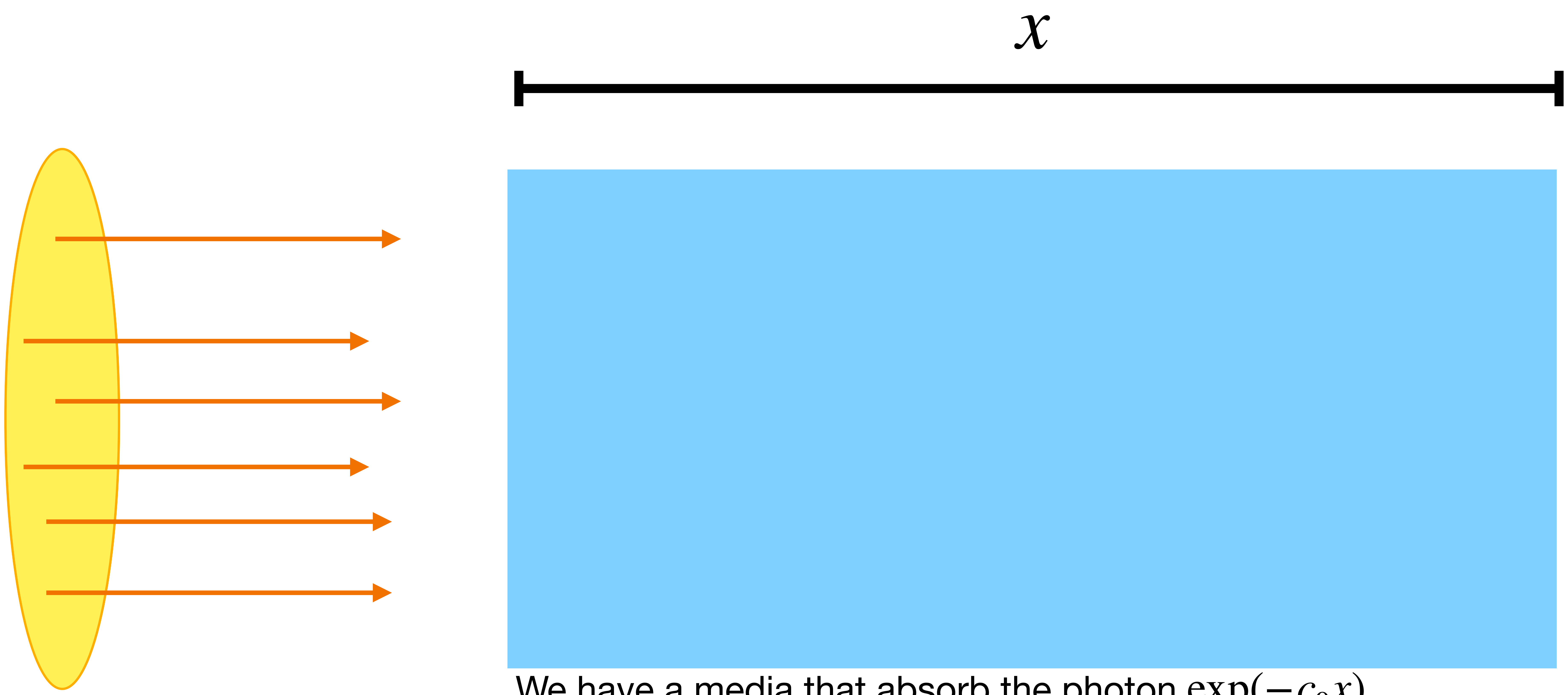
Example



We emit photons with power E.

Russian Roulette

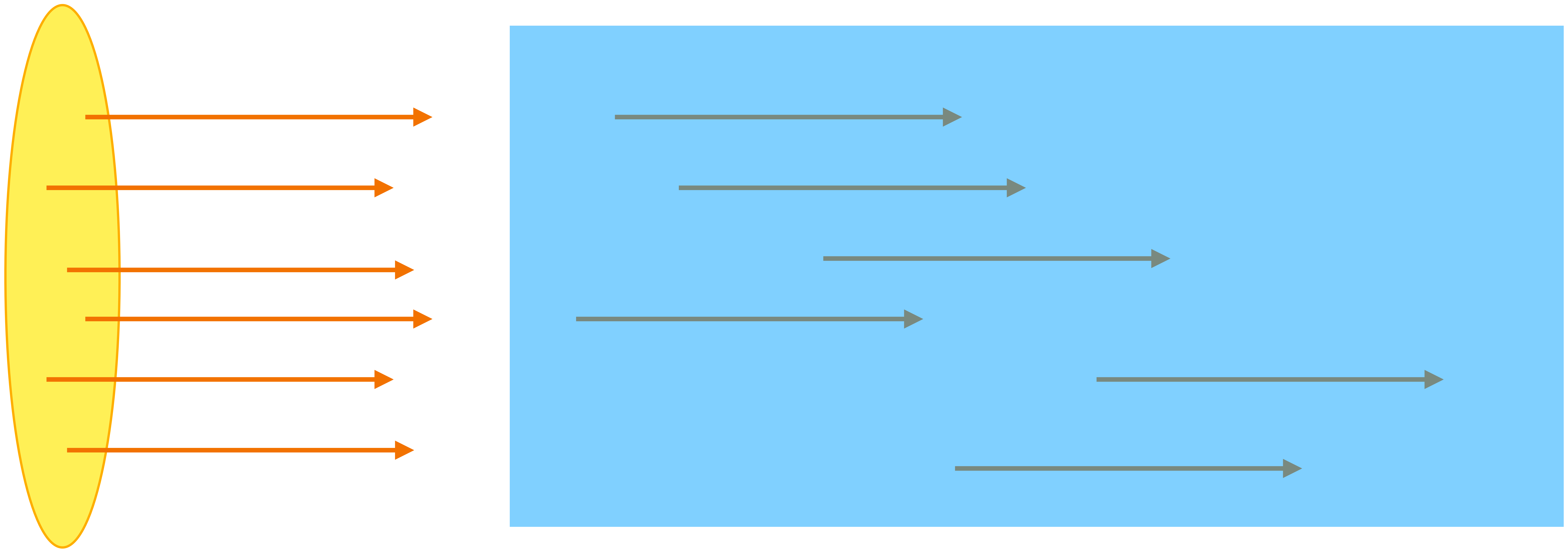
Example



We have a media that absorb the photon $\exp(-c_0 x)$, Beer's Law, where c is a constant of the media and x is the travelled distance.

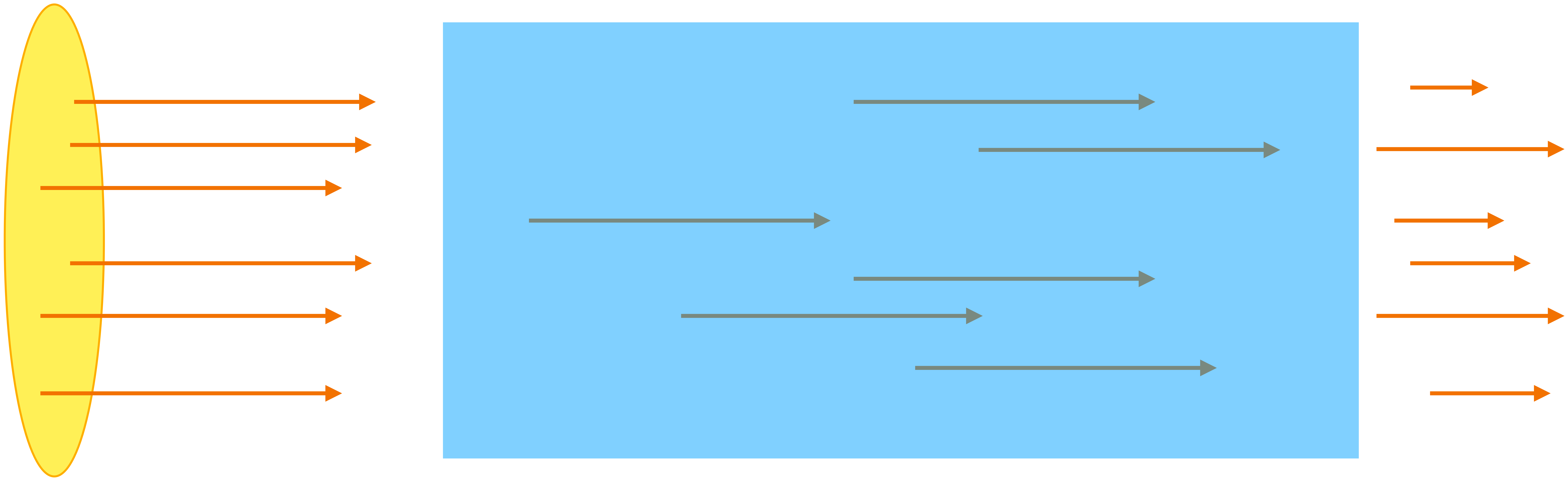
Russian Roulette

Example



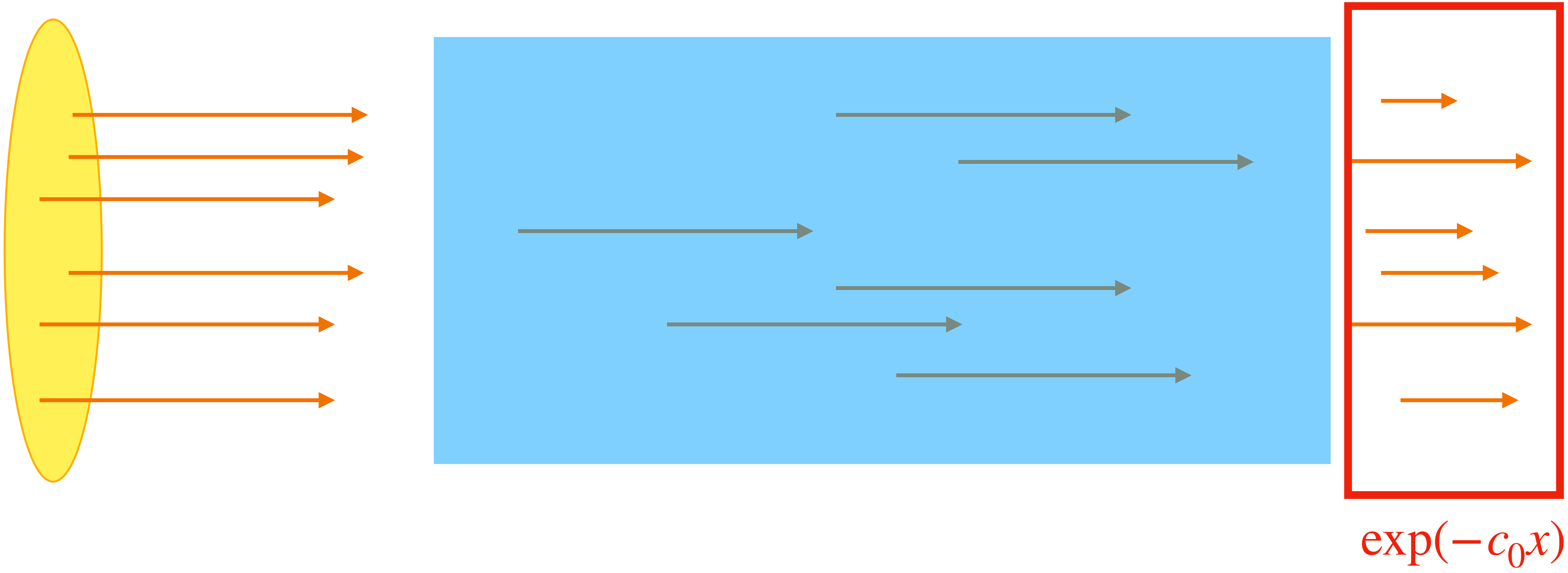
Russian Roulette

Example



Russian Roulette

Example



Russian Roulette

Example

- In this case, we want to estimate the mean energy reaching the end of the media, which absorbs photons’ energy.
- If we reduce the power of each photon, we are left with low energy photons, so we start to have tiny values —> numerical issues with floating point numbers!
- We use Russian Roulette to avoid to sum tiny values up:
  - We keep photons with probability:

$$u < e^{-c_0 x} \qquad u \in \mathbf{U}(0,1).$$

Importance Sampling

Importance Sampling

In Integration

- Importance sampling can be a powerful tool for reducing variance quickly.
- Let's recap how we estimate our averages:

$$\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} g(\mathbf{x}_i) = \frac{1}{n}\sum_{i=1}^{n} \frac{f(\mathbf{x}_i)}{p(\mathbf{x}_i)} \qquad \mathbf{x}_i \sim^{\text{i.i.d.}} p\,.$$

- How can we speed this up?
  - Drawing samples with a PDF that is close to our function f that we want to integrate.
  - We need to do this with care, it may backfire badly; e.g., infinite variance for a problem with finite variance!

Importance Sampling

Main Idea: The Ideal Distribution

- The ideal distribution for sampling would be

$$p(\mathbf{x}_i) \propto f(\mathbf{x}_i) \rightarrow p(\mathbf{x}_i) = cf(\mathbf{x}_i) \ .$$

- This means:

$$c = \frac{1}{\int_{\mathscr{D}} f(\mathbf{x}) d\mathbf{x}} \ .$$

- Note that this require to know the integral that we want to estimate!

Importance Sampling

Main Idea

- More in general, in our usual estimation:

$$\mathbb{E}(f(\mathbf{x})) = \int_{\mathscr{D}} f(\mathbf{x})p(\mathbf{x})d\mathbf{x}\ ,$$

we would like to speed estimation using $\mathbf{X} \sim q$, where q is a PDF. So, we have to:

$$\mathbb{E}(f(\mathbf{x})) = \int_{\mathscr{D}} f(\mathbf{x})p(\mathbf{x})d\mathbf{x} = \int_{\mathscr{D}} \frac{f(\mathbf{x})p(\mathbf{x})}{q(\mathbf{x})}q(\mathbf{x})d\mathbf{x} = \mathbb{E}\left(\frac{f(\mathbf{x})p(\mathbf{x})}{q(\mathbf{x})}\right).$$

Importance Sampling

Main Idea

- This leads to the importance sampling estimator:

$$\hat{\mu}_{q,n} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{f(\mathbf{X}_i)p(\mathbf{X})_i}{q(\mathbf{X}_i)}\right) \qquad \mathbf{X}_i \sim q\ .$$

- The important thing here is that we can compute the term:

$$\frac{f(\mathbf{X}_i)p(\mathbf{X})_i}{q(\mathbf{X}_i)}\ .$$

Importance Sampling

Main Idea

- When $q(\mathbf{X}_i) > 0$ and $f(\mathbf{X}_i)p(\mathbf{X})_i \neq 0$, we have that:

$$\mathbb{E}(\hat{\mu}_{q,n}) = \mu \qquad \text{Var}(\hat{\mu}_{q,n}) = \sigma_q^2, \text{ where:}$$

$$\sigma_q^2 = \int_{\mathcal{Q}} \frac{(f(\mathbf{x})p(\mathbf{x}))^2}{q(\mathbf{x})} d\mathbf{x} - \mu^2 = \int_{\mathcal{Q}} \frac{(f(\mathbf{x})p(\mathbf{x}) - \mu q(\mathbf{x}))^2}{q(\mathbf{x})} d\mathbf{x} \qquad \mathcal{Q} = \{\mathbf{x} \,|\, q(\mathbf{x}) > 0\}.$$

- **A good q helps us to reduce variance!**
- **$(f(\mathbf{x})p(\mathbf{x}) - \mu q(\mathbf{x}))^2$ is small when $q(\mathbf{x}) \propto f(\mathbf{x})p(\mathbf{x})$.**
- Small values of $q(\mathbf{x})$ destroys being proportional to $f(\mathbf{x})p(\mathbf{x})$.

Importance Sampling

Main Idea

- When $q(\mathbf{X}_i) > 0$ and $f(\mathbf{X}_i)p(\mathbf{X})_i \neq 0$, we have that:

$$\mathbb{E}(\hat{\mu}_{q,n}) = \mu \qquad \text{Var}(\hat{\mu}_{q,n}) = \sigma_q^2, \text{ where:}$$

$$\sigma_q^2 = \int_{\mathcal{Q}} \frac{(f(\mathbf{x})p(\mathbf{x}))^2}{q(\mathbf{x})} d\mathbf{x} - \mu^2 = \int_{\mathcal{Q}} \frac{(f(\mathbf{x})p(\mathbf{x}) - \mu q(\mathbf{x}))^2}{q(\mathbf{x})} d\mathbf{x} \qquad \mathcal{Q} = \{\mathbf{x} \,|\, q(\mathbf{x}) > 0\}.$$

- **A good q helps us to reduce variance!**
- **$(f(\mathbf{x})p(\mathbf{x}) - \mu q(\mathbf{x}))^2$ is small when $q(\mathbf{x}) \propto f(\mathbf{x})p(\mathbf{x})$.**
- Small values of $q(\mathbf{x})$ destroys being proportional to $f(\mathbf{x})p(\mathbf{x})$.

Importance Sampling

Main Idea

- When $q(\mathbf{X}_i) > 0$ and $f(\mathbf{X}_i)p(\mathbf{X})_i \neq 0$, we have that:

$$\mathbb{E}(\hat{\mu}_{q,n}) = \mu \qquad \text{Var}(\hat{\mu}_{q,n}) = \sigma_q^2, \text{ where:}$$

$$\sigma_q^2 = \int_{\mathcal{Q}} \frac{(f(\mathbf{x})p(\mathbf{x}))^2}{q(\mathbf{x})} d\mathbf{x} - \mu^2 = \int_{\mathcal{Q}} \frac{(f(\mathbf{x})p(\mathbf{x}) - \mu q(\mathbf{x}))^2}{q(\mathbf{x})} d\mathbf{x} \quad \mathcal{Q} = \{\mathbf{x} \,|\, q(\mathbf{x}) > 0\}.$$

- **A good q helps us to reduce variance!**
- $(f(\mathbf{x})p(\mathbf{x}) - \mu q(\mathbf{x}))^2$ **is small when** $q(\mathbf{x}) \propto f(\mathbf{x})p(\mathbf{x})$.
- Small values of $q(\mathbf{x})$ destroys being proportional to $f(\mathbf{x})p(\mathbf{x})$.

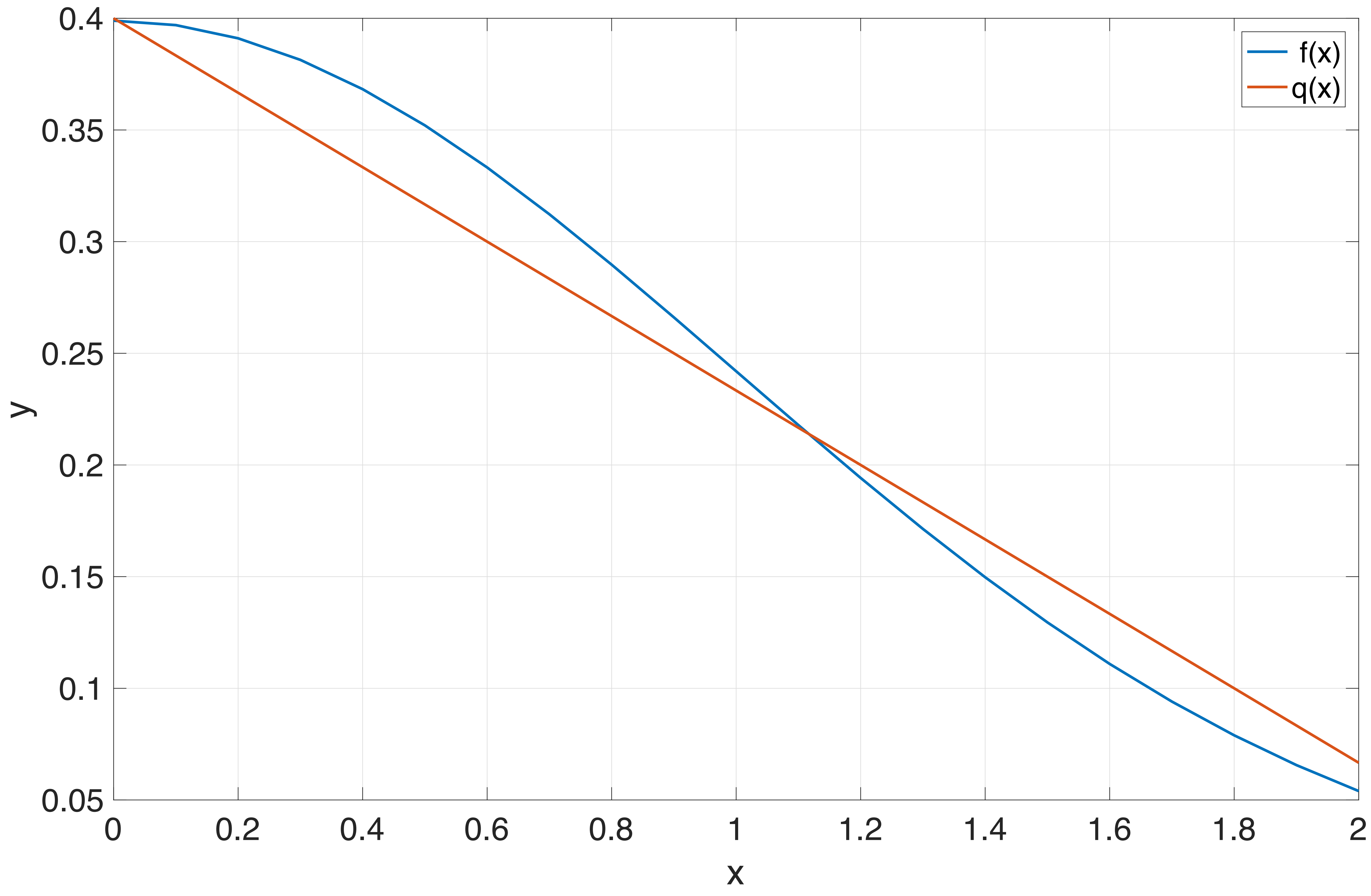
Importance Sampling

Main Idea

- Another insights is that a zero variance $q(\mathbf{x})$ means that we just need $f(\mathbf{x})$ and $p(\mathbf{x})$ to compute our estimate!
- How does this help us?
    - We should design $q(\mathbf{x})$ to follow energy peaks when $f(\mathbf{x})p(\mathbf{x})$ does!
    - To achieve this, we have to know the specific problem.

Importance Sampling
Example

- We want to integrate $N(0,1)$ in [0,2].
- We use as guiding PDF: $q(x) = \frac{15}{7}\left(\frac{2}{5} - \frac{x}{6}\right)$.
- With 10,000 sample we get a $\sigma = 0.0445$, and this is less than $\sigma = 0.2291$ using uniform sampling.



Metropolis Sampling

Metropolis Sampling

Markov Chain

- A Markov Chain is a sequence of random variables, X_i.
- Such random variables have to satisfy the following:

$$P(X_{i+1} = x_{i+1} \mid X_i = x_i, \ldots, X_1 = x_1) = P(X_{i+1} = x_{i+1} \mid X_i = x_i),$$

where $X_i \in \Omega$, the space state.

- This means that a Markov Chain does not have a memory; i.e., the next state depends only on the previous one.

Metropolis Sampling

Markov Chain

- When we have n states, we can define a $n \times n$ matrix called the transition matrix:

$$P = \begin{bmatrix} 0.0 & 0.5 & 0.25 & 0.25 \\ 0.1 & 0.05 & 0.8 & 0.05 \\ 0.9 & 0.05 & 0.0 & 0.5 \\ 0.2 & 0.35 & 0.45 & 0.0 \end{bmatrix}.$$

Note that $\forall_{i,j} P(i,j) \geq 0$ and $\sum_{j=1}^{n} P(i,j) = 1$.

Metropolis Sampling

Markov Chain

- A distribution, π, over Ω is stationarity when:

$$\forall_{x \in \Omega} \pi(x) = \sum_{\Omega} \pi(y) P(y \to x), \text{ this means } \pi = \pi P,$$

where $P(x, y) = P(x \to y)$, $\sum_{\Omega} \pi(x) = 1$, and $\forall_{\mathbf{x}} \pi(\mathbf{x}) \geq 0$.

Metropolis Sampling

Main Idea

- Metropolis-Hastings Sampling draws samples by knowing only our PDF $p(\mathbf{x})$ or $\pi(\mathbf{x})$ in Markov Chain theory. Requirements:
  - $\pi(\mathbf{x})$ has to be positive;
  - We can evaluate $\pi(\mathbf{x})$.
- No need to:
  - Compute the CDF;
  - Invert the CDF.

Metropolis Sampling

Main Idea

- In MH, we accept/use a new generated sample $\mathbf{x}_{i+1}$ as:

$$u \leq A(\mathbf{x}_i \rightarrow \mathbf{x}_{i+1}) \qquad u \in \mathbf{U}(0,1),$$

where:

$$A(\mathbf{x}_i \rightarrow \mathbf{x}_{i+1}) = \min\left(1, \frac{\pi(\mathbf{x}_{i+1})T(\mathbf{x}_{i+1} \rightarrow \mathbf{x}_i)}{\pi(\mathbf{x}_i)T(\mathbf{x}_i \rightarrow \mathbf{x}_{i+1})}\right).$$

Metropolis Sampling

Main Idea

- If the distribution is already at equilibrium we have detailed balance. This is defined as:

$$\pi(\mathbf{x}_{i+1})T(\mathbf{x}_{i+1} \to \mathbf{x}_i)A(\mathbf{x}_i \to \mathbf{x}_{i+1}) = \pi(\mathbf{x}_i)T(\mathbf{x}_i \to \mathbf{x}_{i+1}).$$

- Note that our problem is:

$$\mathbb{E}(f(\mathbf{x})) = \int_{\mathcal{D}} f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x},$$

and our classic estimator is:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i) \qquad \mathbf{x}_i \sim \pi.$$

Metropolis Sampling

Main Idea: Generating New Samples and the Transition Function

- How do we generate $\mathbf{x}_{i+1}$?
    - We start from $\mathbf{x}_i$, and we modify/mutate it:
        - For this mutation, we have to know how to compute its PDF or:

$$T(\mathbf{x}_{i+1} \rightarrow \mathbf{x}_i) \, .$$

        - Note that if $T(\mathbf{x}_{i+1} \rightarrow \mathbf{x}_i) = T(\mathbf{x}_i \rightarrow \mathbf{x}_{i+1})$, we can simplify A as:

$$A(\mathbf{x}_i \rightarrow \mathbf{x}_{i+1}) = \min\left(1, \frac{\pi(\mathbf{x}_{i+1})}{\pi(\mathbf{x}_i)}\right).$$

Metropolis Sampling

Main Idea: Mutations

- How do we mutate samples?
  - We should perturb samples with big changes rather than small ones:
    - We have to explore as fast as possible the entire domain to find peaks.
    - We do not want to explore a local minima:
      - Variance will increase as well if we do not move our samples around.
    - We also have to find a balance because too large mutations may be rejected more easily:
      - We can rely on more than one mutation strategy.

Metropolis Sampling

Main Idea: Start-up Bias

- How do we pick $\mathbf{x}_0$?
  - **Warm-up**: We may start with a random $\mathbf{x}_0$, run some iterations of MH, and then we have to hope for the best. How many? $b = n/2$. So our estimator becomes:

$$\hat{\mu} = \frac{1}{n-b} \sum_{i=b+1}^{n} f(\mathbf{x}_i) \quad b < n.$$

  - **Weighting**: We sample $\mathbf{x}_0 \sim p$, and then we need to take into account of this PDF by scaling the samples that we will draw by:

$$\frac{\pi(\mathbf{x}_0)}{p(\mathbf{x}_0)}.$$

Metropolis Sampling

Main Idea: Error

- To have an estimate of the error interval of our estimate, we use batching.
- We get n samples in total, which are the sum of l batches with k consecutive samples. We analyze our batches:

$$\forall_{j\in[1,k]} \quad \bar{y}_j = \frac{1}{k}\sum_{i=(j-1)k+1}^{jk} y_i \qquad y_i = f(\mathbf{x}).$$

- So our error interval is given by

$$\bar{y} \pm t_{k-1}^{0.995} s \qquad s^2 = \frac{1}{k(k-1)}\sum_{i=1}^{k}(\bar{y}_i - \bar{y})^2,$$

where t is the Student's t function

Some Extra Stuff

Common Random Numbers

Main Idea

- In some cases, we have to estimate:

$$\mathbb{E}(f(\mathbf{x}) - g(\mathbf{x})) = \mathbb{E}(f(\mathbf{x})) - \mathbb{E}(g(\mathbf{x})) \qquad \mathbf{x} \sim p.$$

- Therefore, we can do our estimate in two ways:

$$\hat{\mu}_n^C = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i) - g(\mathbf{x}_i) \qquad \hat{\mu}_n^I = \frac{1}{n_f}\sum_{i_f=1}^{n_f} f(\mathbf{x}_{i_f}) - \frac{1}{n_g}\sum_{i_g=1}^{n_g} g(\mathbf{x}_{i_g}).$$

- Note that the variance varies:

$$\text{Var}(\hat{\mu}_n^C) = \frac{1}{n}\left(\sigma_f^2 + \sigma_g^2 - 2\rho\sigma_f\sigma_g\right) \quad \rho \in [-1,1] \qquad \text{Var}(\hat{\mu}_n^I) = \frac{1}{n}\left(\sigma_f^2 + \sigma_g^2\right).$$

Moment Matching

Main Idea

- In some cases, we know $\mathbb{E}(\mathbf{X}) = \mu_{\mathbf{X}}$. When this happens, we can improve:

$$\hat{\mu}_n = \mathbb{E}(f(\mathbf{x})),$$

by adjusting samples mean as:

$$\hat{\mathbf{x}}_i = \mathbf{x}_i - \overline{\mathbf{x}} + \mu_{\mathbf{X}}.$$

- This can be extended to variance as well if we have it.

Bibliography

- Art Owen. “Chapter 8: Variance reduction” from the book “Monte Carlo theory, methods and examples”. 2013.
- Art Owen. “Chapter 10: Advanced variance reduction” from the book “Monte Carlo theory, methods and examples”. 2013.
- Art Owen. “Chapter 11: Markov chain Monte Carlo” from the book “Monte Carlo theory, methods and examples”. 2013.
- Matt Pharr, Wenzel Jakob, and Greg Humphreys. “Physically Based Rendering: From Theory To Implementation”. Chapter 13: “ Monte Carlo Integration”. Morgan Kaufmann. 2016.

Thank you for your attention!