The Law of Large Numbers:

Meaning, Simulations, and Applications in Cybersecurity and Data Analysis

Clara Glotin

December 2025 - Roma

Abstract

This report provides a comprehensive analysis of the Law of Large Numbers (LLN), a cornerstone of statistical theory and probability. Beginning with the formal definitions and proofs of the Weak and Strong laws, we explore advanced theoretical perspectives including Kolmogorov's Zero-One Law and Ergodic Theory. The study bridges theory and practice through Python simulations of dice rolls and Monte Carlo methods. Finally, we demonstrate the critical role of the LLN in modern applications, specifically in cybersecurity (DDoS detection, RNG testing) and Machine Learning (Importance Sampling), highlighting how asymptotic stability underpins data analysis.

Contents

1 Introduction

The Law of Large Numbers (LLN) is the fundamental theorem of probability that describes the result of performing the same experiment a large number of times. According to the law, the average of the results obtained from a large number of trials should be close to the expected value and will tend to become closer to the expected value as more trials are performed.

Historically, this concept was first envisioned by Gerolamo Cardano, but it was Jacob Bernoulli who provided the first rigorous proof for binary events (Bernoulli trials) in 1713. Later, the term "Law of Large Numbers" was coined by Siméon Denis Poisson in 1837 to describe the stability of frequencies in large samples. Today, it serves as the mathematical foundation for statistics, insurance, and the security of digital information, as detailed in classical textbooks [1, 2].

2 Meaning and Formal Statements

To understand the Law of Large Numbers, we distinguish between its two forms. While the Strong Law is the definitive result (implying the Weak Law), both offer different insights. The Weak Law provides useful probability bounds for finite sample sizes, whereas the Strong Law describes the theoretical certainty of the limit.

A crucial prerequisite for both is that the random variables must be integrable (i.e., in L^1, meaning $\mathbb{E}[|X|] < \infty$).

2.1 The Weak Law of Large Numbers (WLLN)

The Weak Law guarantees convergence in probability. It is historically significant because its proof (via Chebyshev's inequality) provides explicit error bounds for a given n.

Theorem 1 (Weak Law of Large Numbers)**.** *Let $X_1, X_2, \dots$ be a sequence of independent and identically distributed (i.i.d.) random variables in L^1 with mean $\mu = \mathbb{E}[X_1]$. The sample average $\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ converges in probability to μ.*

$$\forall \epsilon > 0, \quad \lim_{n\to\infty} P(|\bar{X}_n - \mu| > \epsilon) = 0$$

2.2 The Strong Law of Large Numbers (SLLN)

The Strong Law is the most powerful result. It asserts that the observed average stabilizes on the mean with probability 1.

Theorem 2 (Strong Law of Large Numbers (Kolmogorov))**.** *Let $X_1, X_2, \dots$ be a sequence of independent and identically distributed (i.i.d.) random variables. If $X_1 \in L^1$ (i.e., $\mathbb{E}[|X_1|] < \infty$), then the sample average $\bar{X}_n$ converges almost surely to $\mu = \mathbb{E}[X_1]$.*

$$P\left(\lim_{n\to\infty} \bar{X}_n = \mu\right) = 1$$

2.3 Relation between Weak and Strong Laws

You might wonder: *Is the Weak Law necessary if the Strong Law exists?*

From a purely measure-theoretic perspective, almost sure convergence implies convergence in probability. Therefore, the Weak Law is indeed a corollary of the Strong Law.

However, the Weak Law remains fundamental for two reasons:

1. **Constructive Bounds:** The proof of the Weak Law (using variance) allows us to calculate *how many* samples are needed to stay within an error margin ϵ with a certain confidence (e.g., using Chebyshev's inequality). The Strong Law is purely asymptotic.

2. **Weaker Assumptions (in specific cases):** In some variations where the variance is finite but the variables are not strictly i.i.d. (e.g., weak correlation), a form of the Weak Law may hold even if the Strong Law does not.

In summary: The Strong Law tells us *what* happens (perfect stability), while the Weak Law helps us calculate *margins of error* for finite samples.

3 Proofs

3.1 Detailed Proof of the Weak Law: Markov and Bienaymé-Tchebychev Inequalities

The modern proof of the Weak Law of Large Numbers is based on Pafnuty Chebyshev's seminal article titled *'Des valeurs moyennes'* (On Mean Values) [3], published in 1867. Chebyshev introduced the brilliant concept of using the variance to bound the probability of a deviation from the mean. This approach significantly simplified the original proof proposed by Jacob Bernoulli, which was far more cumbersome and specific to binary trials.

This proof relies on bounding the probability of deviations.

3.1.1 Markov's Inequality

Let X be a non-negative random variable. For any $a > 0$:

$$P(X \geq a) \leq \frac{E[X]}{a}$$

This inequality shows that the probability of a variable being large is limited by its expectation.

3.1.2 Bienaymé-Tchebychev Inequality

Let Y be a random variable with mean μ and variance σ^2. By applying Markov's inequality to $(Y - \mu)^2$, we obtain:

$$P(|Y - \mu| \geq \epsilon) \leq \frac{\sigma^2}{\epsilon^2}$$

3.1.3 Convergence of the Sample Mean

For $\bar{X}_n$, the variance is $\text{Var}(\bar{X}_n) = \frac{\sigma^2}{n}$. Substituting this into the inequality:

$$P(|\bar{X}_n - \mu| \geq \epsilon) \leq \frac{\sigma^2}{n\epsilon^2}$$

As n tends to infinity, the probability of the average deviating from μ drops to zero.

3.2 Proof of the Strong Law of Large Numbers (SLLN)

The formalization of the Strong Law of Large Numbers reached its definitive form through the work of Andrei Kolmogorov in his landmark 1933 publication [4], *'Foundations of the Theory of Probability'*. While the Weak Law discusses probability at a specific point, the Strong Law establishes a much more profound guarantee of convergence. The proof presented here follows the approach refined by Francesco Cantelli [5], utilizing the fourth moment and the Borel-Cantelli Lemma to demonstrate that the sample mean converges almost surely.

It is important to note that the proof provided in this report, which assumes a finite fourth moment, is a specialized version of the theorem. Kolmogorov's true breakthrough in 1933 was proving the Strong Law under the sole condition that the first moment E[|X|] is finite [4]. This general proof is significantly more complex: it requires truncating the random variables and employing the Kolmogorov Maximal Inequality along with the Kronecker Lemma. While the fourth-moment approach (Cantelli's) is sufficient to demonstrate the convergence mechanism for most practical distributions, Kolmogorov's general version remains the definitive theoretical limit of the theorem.

Here we will prove the SLLN under the assumption that the variables X_i have a finite fourth moment, meaning $E[X_i^4] < \infty$. Without loss of generality, let us assume $E[X_i] = \mu = 0$.

3.2.1 The Fourth Moment Analysis

Consider the sum $S_n = X_1 + \cdots + X_n$. We examine the expected value of S_n^4:

$$E[S_n^4] = E[(X_1 + \cdots + X_n)^4]$$

When expanding this multinomial, because the variables are independent and have a mean of zero, most terms (like $E[X_i^3 X_j]$ or $E[X_i X_j X_k X_l]$) vanish. Only terms of the form $E[X_i^4]$ and $E[X_i^2 X_j^2]$ remain:

$$E[S_n^4] = nE[X_i^4] + 3n(n-1)E[X_i^2]^2$$

Since $E[X_i^4]$ is finite, there exists a constant C such that $E[S_n^4] \leq Cn^2$.

3.2.2 Applying the Markov Inequality

Using the fourth power version of the Markov inequality for the sample mean $\bar{X}_n = S_n/n$:

$$P(|\bar{X}_n| > \epsilon) = P(S_n^4 > \epsilon^4 n^4) \leq \frac{E[S_n^4]}{\epsilon^4 n^4} \leq \frac{Cn^2}{\epsilon^4 n^4} = \frac{C}{\epsilon^4 n^2}$$

3.2.3 Borel-Cantelli Lemma and Almost Sure Convergence

The key difference between the Weak and Strong law lies here. We sum the probabilities for all n:

$$\sum_{n=1}^{\infty} P(|\bar{X}_n| > \epsilon) \leq \sum_{n=1}^{\infty} \frac{C}{\epsilon^4 n^2}$$

This series converges because it is a Riemann series ($\sum 1/n^2 < \infty$).

According to the First Borel-Cantelli Lemma, if the sum of probabilities of events A_n is finite, then the probability that infinitely many of them occur is 0.

Therefore, $|\bar{X}_n| > \epsilon$ happens only for a finite number of n values. For all n beyond a certain point, the average stays within ϵ of the mean. This proves that $\bar{X}_n$ converges almost surely to μ.

4 Advanced Theoretical Perspectives

The Law of Large Numbers connects probability theory to broader mathematical fields such as Measure Theory and Dynamical Systems. Here we explore two advanced concepts that deepen our understanding of why convergence happens.

4.1 Kolmogorov's Zero-One Law

While the Strong Law asserts that convergence happens with probability 1, Kolmogorov's Zero-One Law explains the structural nature of this probability. It deals with "tail events"—events that are independent of any finite subset of the sequence.

4.1.1 Tail Events and Tail Sigma-algebra

Let $(X_n)_{n\geq 1}$ be a sequence of independent random variables. The tail sigma-algebra, denoted $\mathcal{T}$, contains events whose occurrence does not depend on the first k values of the sequence, no matter how large k is [6]. Formally:

$$\mathcal{T} = \bigcap_{n=1}^{\infty} \sigma(X_n, X_{n+1}, \dots)$$

Examples of tail events include:

- The series $\sum X_n$ converges.
- The sequence X_n visits the value 0 infinitely often.
- The sample average $\bar{X}_n$ converges to a specific limit.

Conversely, an event like "The sum of the first 100 terms is positive" is not a tail event because it relies entirely on the initial values.

4.1.2 The Theorem and its Relation to LLN

Theorem 3 (Kolmogorov's Zero-One Law)**.** *If an event A belongs to the tail sigma-algebra $\mathcal{T}$ of a sequence of independent random variables, then its probability is either 0 or 1:*

$$P(A) \in \{0, 1\}$$

In the context of the Law of Large Numbers, the event $E = \{\omega : \lim_{n\to\infty} \bar{X}_n(\omega) = c\}$ is a tail event because changing a finite number of initial observations does not alter the asymptotic limit of the average.

Therefore, Kolmogorov's law tells us that the convergence of the sample mean is not a matter of chance in the traditional sense: it is a deterministic certainty inherent to the process. The sequence either converges almost surely (Probability 1) or diverges almost surely (Probability 0). There is no middle ground where it converges "50% of the time." This theorem provides the structural guarantee that the limit behavior is stable, even if it does not explicitly calculate the value of the mean.

4.2 Ergodic Theory

The Law of Large Numbers is arguably the most famous example of an Ergodic Theorem. Ergodic theory studies dynamical systems with an invariant measure, linking the behavior of a system over time to its behavior over the space of all possible states.

4.2.1 Time Average vs. Space Average

In statistical mechanics and physics, we distinguish between two types of averages (concepts discussed in [7]):

1. *Time Average:* Observing one specific particle (or one simulation) over a very long time T.

$$\bar{f}_T = \lim_{T\to\infty} \frac{1}{T} \int_0^T f(x_t)dt$$

2. *Space (Ensemble) Average:* Observing all possible particles (the entire population) at a single instant. This corresponds to the expected value $\mathbb{E}[f(X)]$.

4.2.2 The LLN as a Case of Birkhoff's Theorem

A system is called ergodic if the time average equals the space average almost surely. The Strong Law of Large Numbers essentially states that an i.i.d. process is an ergodic system, a result generalized by Birkhoff's Ergodic Theorem [8]. The Time Average (our sample mean $\bar{X}_n$) converges to the Space Average (the theoretical expectation μ).

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} X_i = \mathbb{E}[X_1] \quad \text{(almost surely)}$$

This perspective is crucial in fields like thermodynamics and machine learning (specifically Markov Chain Monte Carlo methods), where we often cannot calculate the theoretical integral (Space Average) directly. Instead, we simulate the system for a long time (Time Average) and rely on the ergodic property provided by the LLN to assume this simulation represents the true theoretical value.

5 Empirical Verification

To bridge the gap between theory and practice, we perform a numerical simulation using dice rolls. This visual experiment confirms that as the sample size n grows, the empirical results align with the theoretical predictions provided by the Law of Large Numbers.

We consider the rolling of a fair six-sided die. The theoretical expectation is calculated as:

$$\mu = \frac{1+2+3+4+5+6}{6} = 3.5$$

According to the LLN, the average of the results of n rolls should converge to 3.5.

The following Python code simulates 10 independent experiments, each consisting of 20,000 rolls, to observe this stabilization.

```
import numpy as np
import matplotlib.pyplot as plt
import random

random.seed(1)
n = 20000

# Plot the theoretical mean line
plt.axhline(y=3.5, color='k', linestyle='--', label='Expected␣Value␣(3.5)')

# Run 10 experiments
```

```
for i in range(10):
    rolls = np.random.randint(1, 7, size=n)
    # Calculate cumulative average
    cum_avg = np.cumsum(rolls) / np.arange(1, n + 1)

    plt.plot(cum_avg, label=f'Experiment {i+1}')

plt.xlabel('Number of trials (n)')
plt.ylabel('Empirical Mean')
plt.title('Convergence of Dice Rolls toward Expected Value')
# Moving legend outside if too crowded, or keeping it default
plt.legend(loc='upper right', fontsize='small')
plt.show()
```

Listing 1: Python simulation of 10 sequences of dice rolls

The resulting graph (Figure 1) illustrates the core mechanism of the Law of Large Numbers. At the beginning of the experiment ($n < 100$), the curves show high volatility, with averages deviating significantly from 3.5. However, as n increases, the variance decreases (proportional to $1/n$), and all 10 trajectories tightly converge onto the theoretical line.

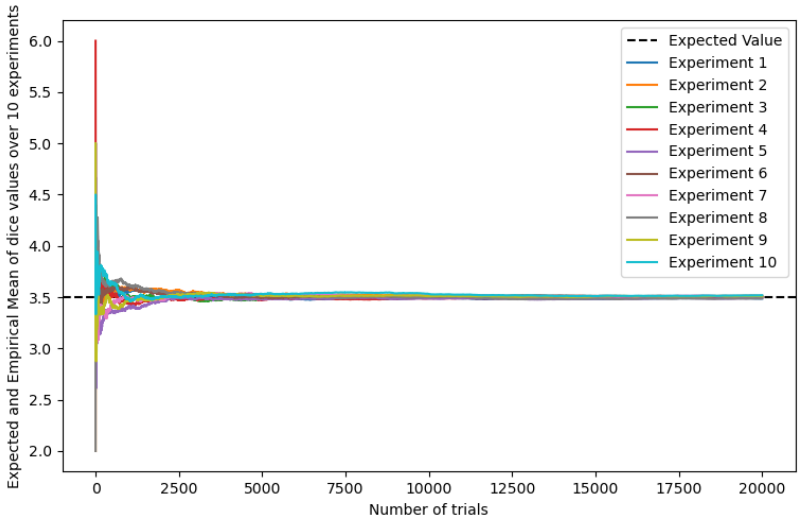

Figure 1: Evolution of the empirical mean of dice rolls over 20,000 trials. The trajectories dampen and converge to $\mu = 3.5$.

6 Applications in Cybersecurity and Data Analysis

The Law of Large Numbers is not merely a theoretical construct; it is the statistical engine behind modern anomaly detection and cryptographic security. By guaranteeing that long-term averages stabilize around a predictable mean, the LLN allows engineers to distinguish between "noise" and "signal" (or attacks).

6.1 Anomaly Detection (DDoS Protection)

Distributed Denial of Service (DDoS) attacks aim to overwhelm a system by flooding it with traffic. Detecting these attacks requires distinguishing between a legitimate spike in user activity and malicious interference. The LLN is the foundation of Traffic Baseline Profiling.

6.1.1 Establishing the Baseline

Under normal operating conditions, network traffic metrics (such as packet arrival rate or request size) can be modeled as random variables $X_1, X_2, \dots$ with an unknown but fixed expectation μ_{normal}. According to the Weak Law of Large Numbers (WLLN), if we monitor the traffic over a sufficiently large window n, the empirical mean $\bar{X}_n$ will provide a high-confidence estimate of μ_{normal}.

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i \xrightarrow{P} \mu_{normal}$$

Security systems constantly calculate this moving average to define what "normal" looks like for a specific network.

6.1.2 Thresholding and Deviation

Once μ_{normal} is established, the system sets dynamic thresholds (often based on variance, e.g., $\mu \pm 3\sigma$). During a DDoS attack, the incoming traffic follows a different distribution with a much higher mean μ_{attack}. Because the sample size n (packets per second) is massive, the LLN ensures that the observed average converges strictly to μ_{attack} and not μ_{normal}.

Consequently, the probability of a "false positive" (normal traffic looking like an attack) becomes negligible as n increases. If $|\bar{X}_{observed} - \mu_{normal}| >$ threshold, the system triggers mitigation protocols (filtering or blackholing), confident that the deviation is structural, not accidental.

6.2 Cryptographic Randomness and RNG Testing

In cryptography, security relies on the unpredictability of keys and initialization vectors (IVs). These are generated by Random Number Generators (RNGs). The security of algorithms like RSA or AES collapses if the RNG is biased.

6.2.1 The Requirement for Uniformity

For a binary stream of bits generated by a True Random Number Generator (TRNG), the probability of generating a '0' or a '1' must be exactly equal: $P(0) = P(1) = 0.5$. Let X_i be a random variable representing the i-th bit generated, where $X_i \in \{0, 1\}$. The expected value is $\mathbb{E}[X] = 0.5$.

6.2.2 The Frequency (Monobit) Test

The NIST Statistical Test Suite (SP 800-22) [9], the global standard for certifying RNGs, relies directly on the Law of Large Numbers. The most fundamental test is the Frequency (Monobit) Test.

Consider a sequence of n bits. We transform the bits to values $+1$ and -1 (where $0 \to -1$ and $1 \to +1$). Let $S_n = \sum_{i=1}^{n}(2X_i - 1)$. According to the LLN, as $n \to \infty$, the sum S_n should oscillate near 0. If the generator is truly random:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} X_i = 0.5$$

If a generator produces a sequence of 1,000,000 bits and the average is 0.55 instead of 0.5, the LLN tells us this is statistically impossible for a fair generator (the p-value would be close to 0). The generator is deemed biased, implying it has low entropy, making the encryption keys predictable and susceptible to brute-force attacks. Practical demonstrations of these statistical vulnerabilities on basic ciphers (Caesar) and naive RSA implementations are detailed in the author's experimental portfolio [10].

7 Monte Carlo Methods and Machine Learning Applications

The Monte Carlo method is a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results. Its mathematical foundation rests directly on the Strong Law of Large Numbers (SLLN).

Since the SLLN guarantees that the sample mean converges almost surely to the expected value, Monte Carlo methods allow us to approximate complex integrals, sums, or system behaviors by simulating a large number of random events. The more random variables used in the simulation (n), the better the approximation of the exact value.

7.1 Foundational Example: Estimation of π

A classic illustration of the Law of Large Numbers is the geometric estimation of π. We consider a unit square containing a quarter of a circle. We generate n random points (x, y) uniformly distributed between 0 and 1. A point is inside the quarter circle if $x^2 + y^2 \leq 1$.

The probability of a point falling inside is the ratio of areas: $\frac{\pi/4}{1} = \frac{\pi}{4}$. Thus, we can estimate π using the formula:

$$\pi \approx 4 \times \frac{\text{Points Inside}}{\text{Total Points}}$$

The following Python code simulates this process over 10 experiments to visualize the convergence.

```
import numpy as np
import matplotlib.pyplot as plt
import random

random.seed(1)
n = 20000

# Figure 1: Convergence Plot
plt.figure(figsize=(10, 6))
```

```
plt.axhline(y=3.14159, color='k', linestyle='--')

for i in range(10):
    x = np.random.uniform(0, 1, size=n)
    y = np.random.uniform(0, 1, size=n)
    # Check if point is inside unit circle
    inside_circle = (x**2 + y**2 <= 1)
    cum_avg = 4 * (np.cumsum(inside_circle) / np.arange(1, n + 1))

    plt.plot(cum_avg)

plt.xlabel('Number of trials')
plt.ylabel('Real and estimated value of pi over 10 experiments')
plt.legend(['Real pi value'] + [f'Experiment {k+1}' for k in range(10)])
plt.show()

# Figure 2: Geometric Repartition (Last Experiment)
def get_color(is_inside):
    return 'green' if is_inside else 'red'

plt.figure(figsize=(6, 6))
colors = [get_color(val) for val in inside_circle]
plt.scatter(x, y, marker=".", c=colors, s=1)
plt.xlabel('Random x coordinate')
plt.ylabel('Random y coordinate')
plt.title('Distribution of 20,000 random points')
plt.show()
```

Listing 2: Monte Carlo simulation for π estimation

7.1.1 Visual Results of Convergence

The first figure illustrates the SLLN: initially volatile, the estimation stabilizes around the true value of π (3.14159...) as n increases.

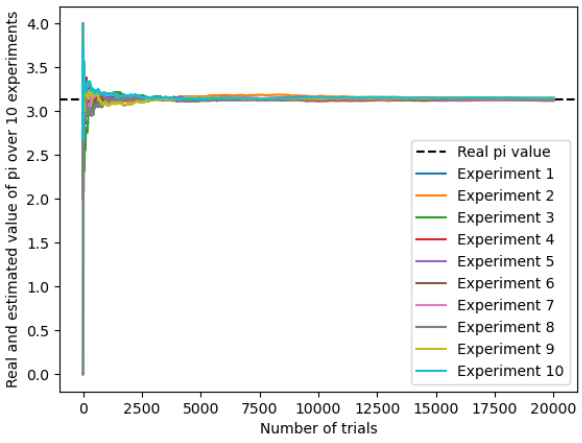

Figure 2: Convergence of the Monte Carlo estimate of π over 10 independent experiments.

The second figure visualizes the randomness. The ratio of green points (inside) to total points approaches $\pi/4$.

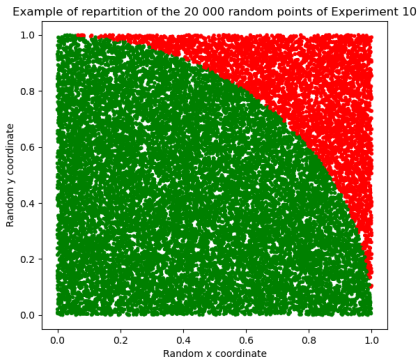

Figure 3: Geometric representation: Green points satisfy $x^2 + y^2 \leq 1$.

7.2 Monte Carlo in Machine Learning

While the estimation of π is an intuitive example, the primary power of Monte Carlo methods lies in high-dimensional problems found in Machine Learning.

Many problems in this field are so difficult that we can never expect to obtain precise answers using deterministic algorithms. Instead, we must use deterministic approximate algorithms such as Monte Carlo approximations [11].

Machine learning algorithms frequently draw samples from probability distributions to:

- Approximate complex sums and integrals.
- Provide a significant speedup to a costly but tractable sum (e.g., subsampling the training cost with mini-batches).
- Approximate intractable sums, such as the gradient of the log partition function of an undirected model.

7.2.1 Formalizing the Approximation

When a sum or an integral cannot be computed exactly—for example, if the sum has an exponential number of terms—it is possible to view it as an expectation. Let S be a sum to estimate, where $p(x)$ is a probability distribution:

$$S = \sum_x p(x) f(x) = \mathbb{E}_p[f(x)]$$

We can approximate S by drawing n samples $x^{(1)}, \ldots, x^{(n)}$ from p and forming the empirical average:

$$\hat{S}_n = \frac{1}{n} \sum_{i=1}^{n} f(x^{(i)})$$

This approximation is rigorously justified by the Law of Large Numbers. If the samples $x^{(i)}$ are independent and identically distributed (i.i.d.) and the variance is bounded, then:

$$\lim_{n \to \infty} \hat{S}_n = S \quad \text{(almost surely)}$$

7.2.2 Advanced Technique: Importance Sampling

Standard Monte Carlo sampling draws from the natural distribution $p(x)$. However, this is not always efficient. An important step in advanced Monte Carlo methods is deciding which part of the summand should play the role of the probability.

There is no unique decomposition because:

$$p(x)f(x) = q(x)\left(\frac{p(x)f(x)}{q(x)}\right)$$

where we can now sample from a different distribution q (the proposal distribution) and average the quantity $\frac{pf}{q}$. The optimal choice q^* corresponds to "optimal importance sampling".

Importance sampling is particularly useful to improve the estimate of the gradient of the cost function in neural networks. Often, most of the cost function's value comes from a small number of "difficult" examples. A recent study by Katharopoulos and Fleuret [12] proposed an efficient importance sampling scheme that accelerates training by focusing computation on samples that introduce the biggest change in parameters, significantly reducing the variance of gradient estimates.

8 Conclusion

This report has explored the Law of Large Numbers (LLN) not merely as a static mathematical theorem, but as the fundamental bridge connecting theoretical probability to the practical realities of data science and cybersecurity.

Through our theoretical analysis, we established the rigorous distinction between the Weak Law and the Strong Law, reinforced by advanced concepts such as Ergodic Theory which guarantees that time averages inevitably align with space averages. Our numerical simulations—ranging from simple dice rolls to the geometric estimation of π—visually confirmed this transition from initial volatility to asymptotic stability.

We also highlighted the critical role of the LLN in modern technology. In cybersecurity, it serves as the baseline for distinguishing normal traffic from DDoS attacks. In Machine Learning, it justifies Monte Carlo approximations, enabling the training of complex models that would otherwise be computationally intractable.

In addition to these theoretical findings, a series of complementary numerical experiments—ranging from RSA cryptanalysis and Poisson processes to the simulation of Brownian Motion—have been documented in a dedicated portfolio [10], illustrating the practical implementation of these stochastic concepts.

In summary, the Law of Large Numbers transforms the uncertainty of individual data points into the certainty of global trends. As we advance further into the era of Big Data and AI, the assurance that "order emerges from chaos" remains the essential axiom upon which our digital infrastructure is built.

References

[1] William Feller. *An Introduction to Probability Theory and Its Applications*, volume 1. Wiley, 1968. Chapters VII and VIII on LLN.

[2] Geoffrey Grimmett and David Stirzaker. *Probability and Random Processes*. Oxford University Press, 2001.

[3] P. L. Tchebychev. Des valeurs moyennes. *Journal de mathématiques pures et appliquées*, 1867. Original introduction of the variance-based proof for WLLN.

[4] A. N. Kolmogorov. *Foundations of the Theory of Probability*. Springer-Verlag, 1933. Original title: Grundbegriffe der Wahrscheinlichkeitsrechnung. Axiomatic foundation of modern probability.

[5] F. P. Cantelli. Sulla probabilità come limite della frequenza. *Rendiconti della Reale Accademia dei Lincei*, 1917. Refinement of the SLLN proof using convergent series.

[6] Patrick Billingsley. *Probability and Measure*. John Wiley & Sons, 3rd edition, 1995. Reference for Borel-Cantelli and Kolmogorov's Zero-One Law.

[7] Peter Walters. *An Introduction to Ergodic Theory*. Springer-Verlag, 1982. Concepts of Time Average vs Space Average.

[8] Rick Durrett. *Probability: Theory and Examples*. Cambridge University Press, 4th edition, 2010. Reference for Birkhoff's Ergodic Theorem and LLN.

[9] Andrew Rukhin et al. A statistical test suite for random number generators for cryptographic applications. Technical Report Special Publication 800-22 Rev 1a, National Institute of Standards and Technology (NIST), 2010.

[10] Clara Glotin. Statistics and cybersecurity: A portfolio of stochastic simulations. Personal Academic Blog, `https://sites.google.com/view/claraglotin/home`, 2025. A collection of 11 experimental reports covering Welford's Algorithm, RSA Cryptanalysis, Random Walks, Poisson Processes, and Brownian Motion simulations.

[11] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[12] Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *Proceedings of the 35th International Conference on Machine Learning*, pages 2525–2534. PMLR, 2018.