The Law of Large Numbers (LLN): meaning, proof and simulations

Pietro Colaguori

November 2023

1 Meaning of the LLN

The Law of Large Numbers is a fundamental concept in probability and statistics that describes the behavior of the average or mean of a large number of independent, identically distributed random variables. In essence, it states that as the sample size increases, the sample mean converges to the expected value of the population from which the sample is drawn.

There are two main forms of the Law of Large Numbers: the Weak Law of Large Numbers and the Strong Law of Large Numbers.

1. **Weak Law of Large Numbers (WLLN)**: The Weak Law of Large Numbers states that the sample mean will converge in probability to the population mean as the sample size increases. In simpler terms, as you take more and more observations from a population, the average of those observations will get closer and closer to the true average of the entire population. However, it doesn't guarantee convergence for every single outcome; rather, it provides a convergence in probability. Mathematically, if $X_1, X_2, ..., X_n$ are independent and identically distributed random variables with a common expected value μ and a common finite variance, then for any positive ϵ:

   $$\lim_{n\to\infty} P\left(\left|\left|\frac{X_1 + X_2 + \ldots + X_n}{n} - \mu\right|\right| \geq \epsilon\right) = 0$$

   This implies that the probability that the sample mean deviates from the population mean by more than ϵ approaches zero as the sample size becomes larger.

2. **Strong Law of Large Numbers (SLLN)**: The Strong Law of Large Numbers is a stronger statement than the Weak Law. It asserts that the sample mean converges almost surely to the population mean. In other words, with probability one, the sample mean will converge to the population mean as the sample size approaches infinity. Mathematically, if

$X_1, X_2, ..., X_n$ are independent and identically distributed random variables with a common expected value μ, then:

$$P\left(\lim_{n\to\infty}\frac{X_1 + X_2 + \ldots + X_n}{n} = \mu\right) = 1$$

This means that as the sample size becomes very large, the sample mean will almost surely converge to the true population mean.

The Law of Large Numbers has important implications in various fields, including statistics, finance, and science. It underlines the reliability of statistical predictions and justifies the use of sample means to estimate population parameters in practice. It provides a theoretical foundation for the concept that, in large enough samples, observed averages are likely to be close to the true population average.

2 Simulations

We can continue by considering some phenomena where the LLN applies, so that we can appreciate the consequences of this important result.

- **Coin Filpping**: consider a fair coin, where the event heads = 1 and the event tail = 0, which means that the sample space is

  $$\Omega = \{0, 1\}$$

  . We can easily calculate probability of each event of the sample space as

  $$\forall X_i \in \Omega : P[X_i] = \frac{1}{n} = \frac{1}{2}, n = |\Omega|$$

  The expected value can also be easily computed as

  $$\mu = \frac{1}{n}\sum_X P[X] = \frac{1}{2}\cdot(P[0] + P[1]) = 0.5$$

  So, according to the LLN, if we flip our fair coin enough times we will get an average on the samples that converges to μ.

- **Dice Rolls**: the same reasoning applied to the previous example can be applied to the rolling of a fair six-sided dice, in this case of course we will have that

  $$\Omega = \{1, 2, 3, 4, 5, 6\}$$

  and that

  $$\forall X_i \in \Omega : P[X_i] = \frac{1}{n} = \frac{1}{6}, n = |\Omega|$$

  The expected value is eeasily computed as:

  $$\mu = \frac{1}{n}\sum_X P[X] = 3.5$$

- **Normal Distributions**: we can generate random numbers from a normal distribution with a specified mean and standard deviation. Then we can alculate the sample mean for varying sample sizes. As the sample size increases, we observe how the sample mean converges to the specified mean of the distribution.
- **Uniform Distribution**: we can generate random numbers from a uniform distribution between 0 and 1 and calculate the sample mean for increasing sample sizes. As the sample size grows, we observe how the sample mean converges to the expected value of 0.5 (the mean of a uniform distribution from 0 to 1).

In Python we can generate a graph and observe the LLN in action, in the code written below I will be taking 1000 samples from a uniform distribution and show how as the number of samples taken increases the average converges to the expected value of $\frac{1}{2}$.

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

# initialize variables
samples = 1000
cumulative_mean = 0
cumulative_mean_list = []

# sample from uniform distribution
for i in range(1, samples + 1):
    x = np.random.uniform(0, 1)
    cumulative_mean += x
    cumulative_mean_list.append(cumulative_mean / i)

# plot results
plt.plot(range(0, samples), cumulative_mean_list,
    label='Cumulative Mean')
plt.axhline(y=0.5, color='r', linestyle='--',
    label='Expected Mean')
plt.xlabel('Number of Samples')
plt.ylabel('Cumulative Mean')
plt.title('Law of Large Numbers')
plt.legend()
plt.show()
```

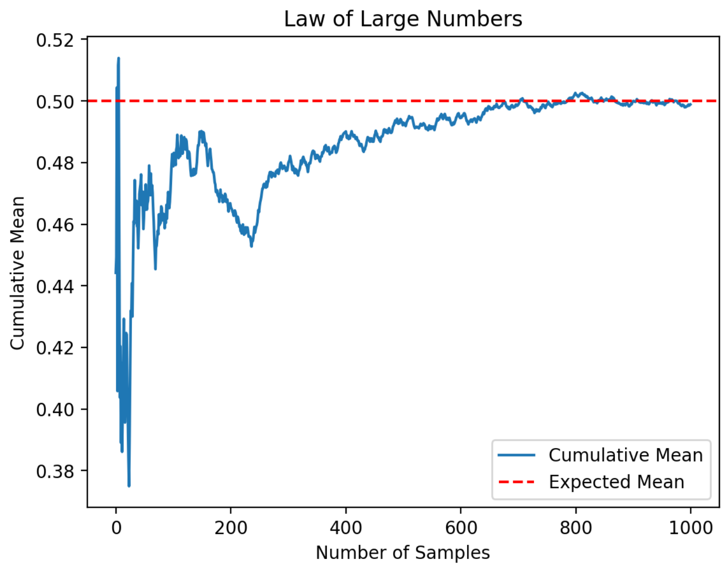

Figure 1: Simulation run with 1000 samples.

Upon running this code, the above graph is generated, showing how our expectations were met.

With 1000 samples I obtained a final average value of 0.51052, with an error from the expected value of 0.5 of:

$$Error = \frac{0.51052 - 0.5}{0.5} = 0.021 \rightarrow Error = 2.1\%$$

Increasing the number of samples, according to the LLN, should also increase my accuracy and reduce the error, the following graph has been generated using 10000 samples, 10 times the number of samples of the previous simulation! The resulting graph is shown in the next page.

This time the final average obtained is 0.50014 and the error, as expected, is much lower:

$$Error = \frac{0.50014 - 0.5}{0.5} = 0.00028 \rightarrow Error = 0.028\%$$

We can also run some other interesting simulations, for instance, I will simulate the toss of a fair coin, which can end up either in 1, meaning heads, or in event 0, meaning tail. I will run the simulation in Python using the following script.

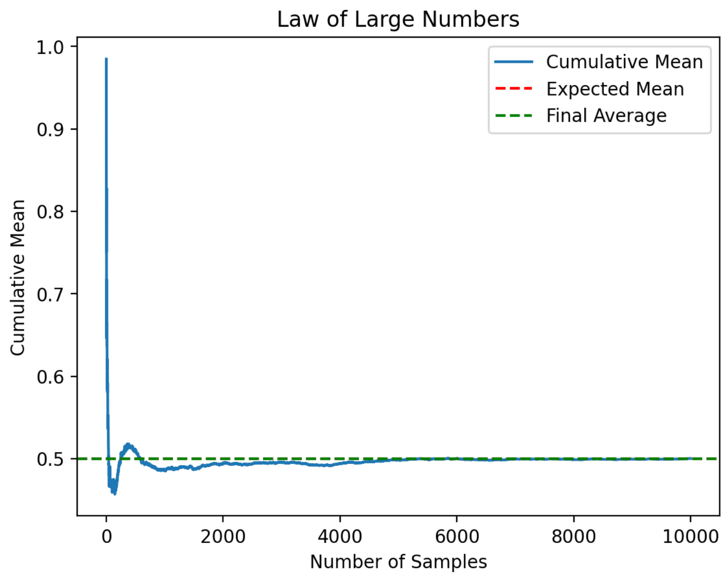

Figure 2: Simulation run with 10000 samples.

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

# initialize variables
samples = 1000
cumulative_mean = 0
cumulative_mean_list = []

# sample from uniform distribution
for i in range(1, samples + 1):
    x = random.randint(0, 1)
    cumulative_mean += x
    cumulative_mean_list.append(cumulative_mean / i)

# plot results
plt.plot(range(0, samples), cumulative_mean_list,
    label='Cumulative Mean')
plt.axhline(y=0.5, color='r', linestyle='--',
    label='Expected Mean')
plt.axhline(y=cumulative_mean / samples, color='g',
    linestyle='--', label='Final Average')
plt.xlabel('Number of Samples')
```

```
plt.ylabel('Cumulative Mean')
plt.title('Law of Large Numbers')
plt.legend()
plt.show()
```

By doing 1000 tosses we get a final average of 0.475, which means an error of 5%, but if we increase the number of tosses to 10000 we get a final average of 0.4994, which means an error of 0.12%.

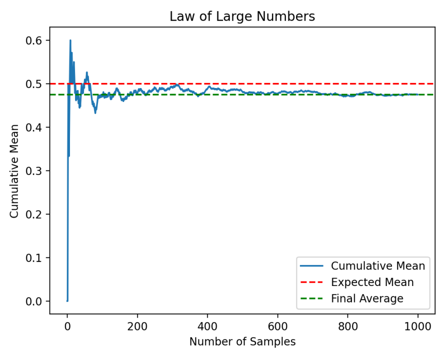

Figure 3: With 1000 samples

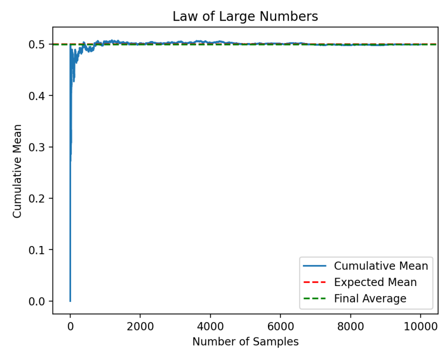

Figure 4: With 10000 samples

We can easily adapt this simulation to simulate the throw of a fair dice of n faces by simply selecting our random integer x between 1 and n instead of between 0 and 1, like we are doing for the coin tosses.

As a final simulation, we can sample values from a normal distribution of mean μ and notice how, as the number of sampling increases, the computed average converges to the value of μ, which is arbitrarly chosen, e.g. $\mu = 3$.

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

# initialize variables
samples = 1000
cumulative_mean = 0
cumulative_mean_list = []

# sample from uniform distribution
for i in range(1, samples + 1):
    x = np.random.normal(3, 1)
    cumulative_mean += x
    cumulative_mean_list.append(cumulative_mean / i)

# plot results
plt.plot(range(0, samples), cumulative_mean_list,
    label='Cumulative Mean')
plt.axhline(y=3, color='r', linestyle='--',
    label='Expected Mean')
plt.axhline(y=cumulative_mean / samples, color='g',
    linestyle='--', label='Final Average')
plt.xlabel('Number of Samples')
plt.ylabel('Cumulative Mean')
plt.title('Law of Large Numbers')
plt.legend()
plt.show()
```

The results observed show a convergence of the computed average to the value $\mu = 3$.

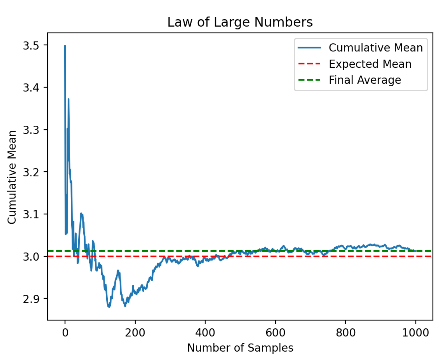

Figure 5: With 1000 samples

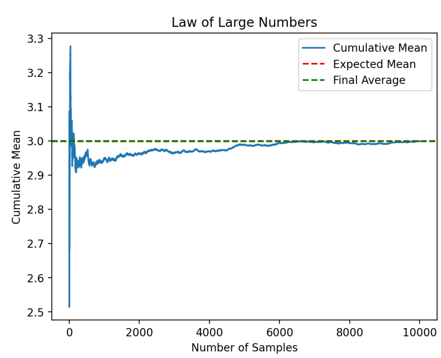

Figure 6: With 10000 samples

3 Proof

1. **Proof of the Weak Law (WLLN)**: We first start by stating the thesis, given $X_1, X_2, ...$ an infinite sequence of independent and identically distributed (i.i.d.) random variables with finite expected values $E(X_1) = E(X_2) = ... = \mu < \infty$, we want to calculate the convergence of the sample average which is:

$$\overline{X}_n = \frac{1}{n}(X_1 + ... + X_n)$$

For the Weak Law of Large Numbers (WLLN) we know that

$$\overline{X}_n \to \mu, n \to \infty$$

We can prove this by using Chebyshev's inequality and assuming finite variance $Var(X_i) = \sigma^2, \forall i$. The independence of the random variables implies the following:

$$Var(\overline{X}_n) = Var(\frac{1}{n}\sum_{i=1}^{n} X_i) = \frac{1}{n^2}Var(\sum_{i=1}^{n} X_i) = \frac{n\sigma^2}{n^2} = \frac{\sigma^2}{n}$$

By observing that $E(\bar(X)_n) = \mu$, we can apply Chebyshev's inequality on $\bar(X)_n$:

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

We can then use this result to obtain the following:

$$P(|\overline{X}_n - \mu| < \epsilon) = 1 - P(|\overline{X}_n - \mu| \geq \epsilon) \geq 1 - \frac{\sigma^2}{n\epsilon^2}$$

We can clearly see that the expression above approaches 1 when $n \to \infty$. In conclusion, by the definition of convergence in probability, we have obtained the thesis:

$$\overline{X}_n \to \mu, n \to \infty$$

□

The method just shown is not the only way to prove the WLLN, in fact we can also do so as follows, by using the convergence of characteristic functions. By Taylor's theorem for complex functions, the characteristic function of any random variable X with finite mean μ, can be written as:

$$\phi_X(t) = 1 + it\mu + o(t), t \to \infty$$

It is important to note the two following properties of characteristic functions:

$$\phi_{\frac{1}{n}X}(t) = \phi_X(\frac{t}{n})$$

and

$$\phi_{X+Y}(t) = \phi_X(t)\phi_Y(t)$$

which holds if X and Y are independent random variables. Now we can calculate the characteristic function of $\overline{X}_n$ in terms of ϕ_X:

$$\phi_{\overline{X}_n}(t) = [\phi_x(\frac{t}{n})]^n = [1 + i\mu\frac{t}{n} + o(\frac{t}{n})]^n \to e^{it\mu}, n \to \infty$$

By the Lévy continuity theorem $\bar(X)_n$ converges in distribution to μ:

$$\overline{X}_n \to^D \mu, n \to \infty$$

since μ is a constant, by the convergence of random variables, we know that the convergence in distribution is equivalent to the convergence in probability.

$$\overline{X}_n \to^P \mu, n \to \infty$$

which is the thesis. □

2. **Proof of the Strong Law (SLLN)**: for this proof too we assume the variables X_i are i.i.d. and have a finite expected value $E(X_i) = \mu < \infty, \forall i$. We also assume that the random variables have a finite fourth moment. Without loss of generality we can assume that $\mu = 0$ by centering. By Chebyshev, we have:

$$P(|\overline{X}_n|^2 > \epsilon^2) \leq \frac{Var(\overline{X^2}_n}{\epsilon^4} = \frac{Var(S_n^2)}{\epsilon^4 n^4} \leq \frac{E(S_n^4)}{\epsilon^4 n^4}$$

Note that the random variables, for the way we defined them, have $\mu = 0$, which means that we can write:

$$E(S_n^4) = E((\sum_{i=1}^{n} X_i)^4) = nE(X^4) + 3n(n-1)E(X^2)^2$$

By Jensen, we know that $E(X^4) > E(X^2)^2$, which implies:

$$E(S_n^4) \leq 3n^2 E(X^4)$$

We can plug the result we just obtained into the first inequality as follows:

$$P(|\overline{X}_n|^2 > \epsilon^2) \leq \frac{3E(X^2)}{\epsilon^4 n^2}$$

Since this is summable, by the Borel-Cantelli lemma we know that:

$$\overline{X}_n \to 0$$

□

4 Applications in Cybersecurity

Statistics, and in particular the LLN too, are applied to analyze and mitigate cyber threats, we can think of different scenarios in which this is true:

- **Large Database Analysis**: With a vast amount of cybersecurity data generated daily (logs, network traffic, system events), the Law of Large Numbers allows security professionals to analyze large datasets to identify patterns and anomalies. This aids in incident response and forensic investigations, enabling the detection of unusual activities that may signify a security incident.

- **Behavioural Analysis**: Analyzing the behavior of users and devices over a large dataset can help establish a baseline of normal behavior. Deviations from this baseline may indicate potential security threats, such as insider threats or compromised accounts. For instance, an user might be trying to perform some sort of code injection attack, e.g. SQL injection, to access an account that does not belong to them.

- **User and Entity Behaviour Analytics**: UEBA leverages the Law of Large Numbers to establish a baseline of typical user behavior. Deviations, such as unusual login times or access patterns, can be flagged as potential security incidents.

- **Simulated Attacks**: Red teaming exercises and penetration testing involve simulating cyber attacks to identify vulnerabilities. The Law of Large Numbers can be applied to perform statistically significant testing across various attack vectors, helping organizations understand their overall security posture.

- **Training AI models**: Machine learning algorithms and artificial intelligence systems benefit from large amounts of training data to learn and improve their accuracy. The Law of Large Numbers is essential in training models to recognize patterns indicative of security threats, such as malware or phishing attacks.

In essence, the Law of Large Numbers enables cybersecurity professionals to make informed decisions by leveraging large datasets to analyze patterns, identify anomalies, and derive meaningful insights. It's a foundational concept for statistical analysis in the field, contributing to the development of effective cybersecurity strategies and defenses.

5 Historical Overview

The Law of Large Numbers, initially proven by the Swiss mathematician Jakob Bernoulli in 1713, emerged during a period when he and his contemporaries were pioneering the formalization of probability theory, primarily to analyze games of chance. Bernoulli conceptualized an infinite sequence of repeated trials in a game characterized by pure chance, where the only possible outcomes were either a win or a loss. In this framework, he assigned the probability of a win as p.

Bernoulli's focus was on understanding the behavior of the fraction representing the number of wins in this game over a substantial number of repetitions. In simpler terms, he aimed to explore how often a win would occur in the long run. It was a prevailing belief at the time that this fraction should gradually converge to p as the number of repetitions increased.

Bernoulli provided a precise and rigorous proof, demonstrating that as the number of repetitions approached infinity, the probability of the fraction being within any predetermined distance from p would tend to 1. In other words, as the trials continued indefinitely, the observed fraction of wins would inevitably get exceedingly close to the true probability p.

This foundational concept laid the groundwork for the Law of Large Numbers, emphasizing that with a sufficiently large number of trials, the empirical probability would converge to the theoretical probability, reinforcing the stability and predictability of outcomes in the long run.

There is also a more general version of the law of large numbers for averages, proved more than a century later by the Russian mathematician Pafnuty Chebyshev.