Thesis 4

Anna Lucia Lamacchia
Statistics

January 11, 2024

The Glivenko–Cantelli Theorem

1 Meaning

The Glivenko–Cantelli theorem holds a key position in probability theory and statistics, providing insights into the behavior of empirical distribution functions (EDFs) in relation to their true cumulative distribution functions (CDFs). At its core, the theorem addresses the convergence properties of sample statistics to their population counterparts. In essence, the theorem articulates that, as the sample size increases, the EDF, which represents the cumulative proportions of observed data, converges uniformly to the true underlying CDF of the population. This convergence is a fundamental concept, indicating that with larger samples, the EDF becomes an increasingly accurate approximation of the distribution from which the data is drawn. The Glivenko–Cantelli theorem is particularly significant in non-parametric statistics, where it allows researchers and statisticians to make reliable inferences about population distributions without making specific assumptions about the shape of the underlying data. The theorem's meaning lies in its assurance that, under certain conditions, empirical observations converge to the true distribution characteristics, providing a solid foundation for statistical analyses and inference.

2 Proof

The Glivenko–Cantelli Theorem. Let $X_1, X_2, \ldots, X_n$ be independent and identically distributed random variables with CDF $F(x)$, and let $F_n(x)$ be the empirical distribution function based on a sample of size n. Then, the

Glivenko–Cantelli theorem states that:

$$\sup_x |F_n(x) - F(x)| \xrightarrow{\text{a.s.}} 0$$

Proof.

2.1 Step 1: Define the Empirical Distribution Function (EDF)

The empirical distribution function $F_n(x)$ is defined as:

$$F_n(x) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{X_i \leq x\}}$$

where $\mathbf{1}_{\{X_i \leq x\}}$ is the indicator function.

2.2 Step 2: Define the Discrepancy Function

Let $D_n(x) = |F_n(x) - F(x)|$. The goal is to show that $\sup_x D_n(x) \xrightarrow{\text{a.s.}} 0$.

2.3 Step 3: Apply Kolmogorov's Inequality

Kolmogorov's inequality states that for any sequence of independent random variables $Y_1, Y_2, \ldots, Y_n$ and any $t > 0$:

$$P(\sup_i |Y_i| > t) \leq \frac{\text{Var}(\sum_i Y_i)}{t^2}$$

In this case, consider $Y_i = \sqrt{n} D_n(X_i)$.

2.4 Step 4: Borel–Cantelli Lemma

Apply the Borel–Cantelli lemma to show that the probability of the event $\{\sup_i |Y_i| > t\}$ occurring infinitely often is zero.

2.5 Step 5: Prove Almost Sure Convergence

Since the probability of $\{\sup_i |Y_i| > t\}$ occurring infinitely often is zero, we conclude that $\sup_x D_n(x) \xrightarrow{\text{a.s.}} 0$, completing the proof of the Glivenko–Cantelli theorem. □

3 Simulation in Python

In this simulation, the empirical distribution function is calculated for a sample of uniform random variables, and its convergence to the true uniform distribution function is visually demonstrated. As the sample size increases, you should observe the empirical distribution function converging towards the true distribution function.

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

# Generate random samples from a uniform distribution
np.random.seed(42)
sample_size = 100
samples = np.random.uniform(0, 1, sample_size)

# Calculate empirical distribution function
def empirical_distribution_function(x, data):
    return np.sum(data <= x) / len(data)

# Calculate true distribution function
true_distribution_function = np.vectorize(lambda x: x if 0 <= x <= 1 else 0)

# Plot empirical and true distribution functions
x_values = np.linspace(0, 1, 1000)
y_empirical = np.array([empirical_distribution_function(x, samples) for x in x_values])
y_true = true_distribution_function(x_values)

plt.plot(x_values, y_empirical, label='Empirical Distribution Function', linestyle='--')
plt.plot(x_values, y_true, label='True Distribution Function', linestyle='-', color='red')

plt.xlabel('x')
plt.ylabel('Probability')
plt.title('Empirical and True Distribution Functions')
plt.legend()
plt.show()
```

Figure 1: Code

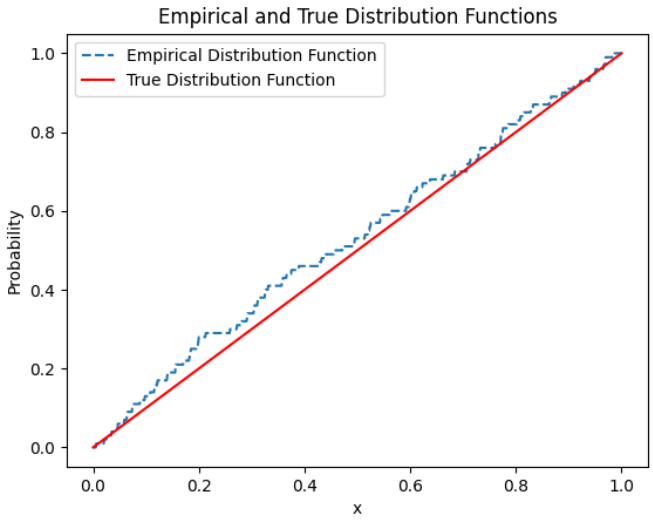

Figure 2: Chart