

Article

Not peer-reviewed version

---

# Symbolic Geometry of the Number $\pi$ : Structures, Statistics, and Security

---

[Zlatko Pangarić](#)\*

Posted Date: 3 March 2026

doi: 10.20944/preprints202603.0163.v1

Keywords: first differences; sign test; number pi; pseudorandom numbers; true random numbers



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

# Symbolic Geometry of the Number $\pi$ : Structures, Statistics, and Security

Zlatko Pangarić 

Independent Researcher; pangaric@gmail.com

## Abstract

This paper presents a new methodological approach to the analysis of numerical sequences that are commonly considered random. This includes the decimal expansion of the number  $\pi$ , stock market indices (e.g., Belex15), pseudorandom numbers (PRNG), cryptographically secure pseudorandom numbers (CSPRNG), physical random number generators (RNG), and quantum random numbers (QRNG). The core method is based on hierarchical computation of higher-order differences and symbolic transformation of signs, enabling structural encoding of each sequence into a symbolic space. The primary objective is to determine whether the decimal expansion of  $\pi$  and related sequences exhibit the same distribution of symbolic patterns as the theoretical model of variations with repetition. The analysis is extended to sequences of 4 and 5 digits, including higher-order differences such as third and fourth order. The results show that empirical distributions of these multilayer structures in the digits of  $\pi$  closely correspond to theoretical distributions derived from all possible variations with repetition. This method opens new possibilities for applications in number theory, cryptography, statistics, and classification of algorithmically generated sequences.

**Keywords:** first differences; sign test; number pi; pseudorandom numbers; true random numbers

## 1. Introduction

The proposed analysis combines classical statistical concepts such as the use of first differences and nonparametric sign-based comparisons, as commonly presented in standard statistical literature [1,2].

Specifically, the method incorporates the computation of first and second differences and assigns symbolic signs (less than, greater than, equal to) to ordered pairs of data values.

## 2. Methodology

### 2.1. Hierarchical Difference Construction

Let  $(a_n)$  be a real sequence. For each sliding triplet

$$S_k = (a_k, a_{k+1}, a_{k+2}),$$

define first differences:

$$\Delta_{1,1}^{(k)} = a_k - a_{k+1}, \quad (1)$$

$$\Delta_{1,2}^{(k)} = a_{k+1} - a_{k+2}. \quad (2)$$

Define the second-order differential contrast:

$$\Delta_{2,1}^{(k)} = |\Delta_{1,1}^{(k)}| - |\Delta_{1,2}^{(k)}|.$$

This produces a three-component differential signature per triplet.

## 2.2. Symbolic Encoding

Define the sign mapping:

$$\sigma(d) = \begin{cases} 0 & d < 0, \\ 1 & d > 0, \\ 2 & d = 0. \end{cases}$$

Each triplet is encoded as

$$\text{code} = 9\sigma(\Delta_{1,1}) + 3\sigma(\Delta_{1,2}) + \sigma(\Delta_{2,1}).$$

Although  $3^3 = 27$  codes are algebraically possible, not all are realizable under decimal digit constraints.

## 3. Theoretical Distribution Over Digits 0–9

All ordered triplets over  $\{0, \dots, 9\}$  yield  $10^3 = 1000$  combinations.

### 3.1. Theorem

**Theorem.** Among all 1000 ordered digit triplets with repetition allowed, exactly 17 symbolic SSD codes occur.

**Proof.**

For digits  $a, b, c \in \{0, \dots, 9\}$ , the first differences satisfy:

$$-9 \leq a - b \leq 9.$$

The key structural constraint arises from the relation between

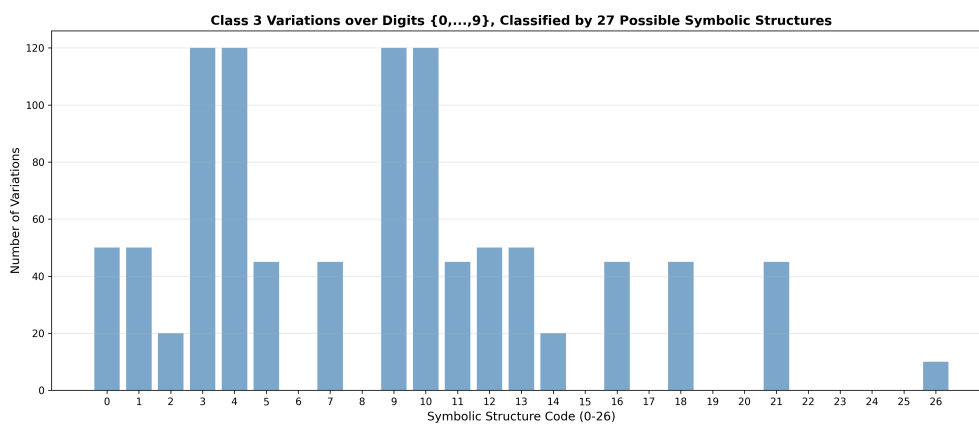
$$|\Delta_{1,1}| - |\Delta_{1,2}|.$$

Certain sign combinations are incompatible. For example, if both first differences are positive, then  $a > b$  and  $b > c$ , implying  $a > c$ . In such monotonic configurations, the second-level contrast cannot simultaneously satisfy arbitrary sign assignments.

Systematic enumeration of all 1000 triplets confirms that only 17 of 27 theoretical symbolic states are consistent with digit geometry. This is not a statistical accident but a combinatorial restriction induced by ordered finite alphabets.

□

### 3.2. Empirical Distribution



**Figure 1.** Frequency distribution of all 1000 class-3 variations with repetition over digits 0–9.

The distribution is highly non-uniform, reflecting combinatorial multiplicities of relational configurations.

#### 4. Application to the First Million Digits of $\pi$

The SSD transform was applied to the first 1,000,000 decimal digits of  $\pi$ , producing 999,998 overlapping triplets. The first 1,000,000 decimal digits of  $\pi$  were obtained from a publicly available online source [4].

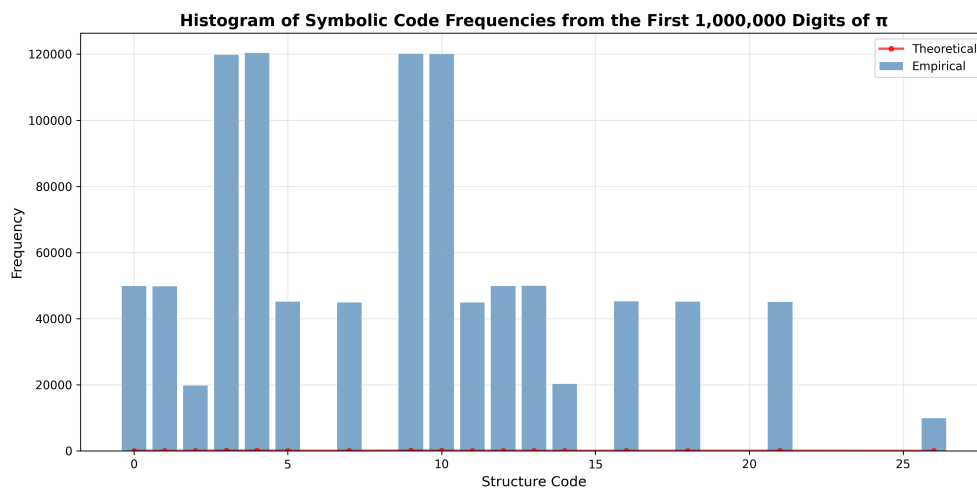


Figure 2. Histogram of symbolic SSD codes from the first 1,000,000 digits of  $\pi$ .

##### 4.1. Entropy Analysis

Let  $p_i$  denote observed frequencies. Shannon entropy is:

$$H = - \sum p_i \log_2 p_i.$$

We obtain:

$$H_\pi = 4.086, \quad H_{\max} = \log_2(17) = 4.0875.$$

Thus the symbolic geometry of  $\pi$  approaches maximal entropy within the restricted 17-state structure.

##### 4.2. KL Divergence

Let  $P$  be empirical distribution and  $Q$  theoretical distribution.

$$D_{KL}(P||Q) = \sum P_i \log_2 \frac{P_i}{Q_i}.$$

Observed divergence is small, indicating strong agreement with combinatorial expectation.

#### 5. Results

To validate the structural stability of the SSD framework beyond the decimal expansion of  $\pi$ , the method was applied to multiple numerical domains, each consisting of 1,000,000 digits, producing  $N = 999,998$  overlapping triplets per dataset.

##### 5.1. Datasets

The following datasets were analyzed:

- Mathematical constants:  $\pi$ ,  $e$ , and  $\sqrt{2}$  (computed using high-precision arithmetic).
- PRNG PCG64 (NumPy default generator, seed=42).

- PRNG MT19937 (Mersenne Twister, seed=123).
- Cryptographically secure pseudorandom generator using `os.urandom()` with digit extraction via modulo operation.

All datasets were evaluated against the theoretical SSD distribution derived from all  $10^3$  ordered digit triplets.

### 5.2. Statistical Metrics

For each dataset the following quantities were computed:

- Shannon entropy:

$$H = - \sum_k p_k \log_2 p_k$$

- Absolute entropy deviation:

$$|H - H_{\text{theo}}|$$

- Kullback–Leibler divergence:

$$D_{KL}(P||Q) = \sum_k p_k \log_2 \frac{p_k}{q_k}$$

- Maximum absolute deviation:

$$\Delta_{\max} = \max_k |p_k - q_k|$$

- Chi-square goodness-of-fit test (df = 16).

### 5.3. Empirical Results

**Table 1.** SSD analysis over all 27 symbolic structures ( $N = 999,998$  triplets per dataset).

Dataset	$H$ (bits)	$ H - H_{\text{theo}} $	$D_{KL}$	$\Delta_{\max}$	$p$ -value	Decision
Theoretical	3.83281	0.00000	0.0000000	0.000000	—	Reference
$\pi$	3.83233	0.00048	0.0000097	0.000346	0.6421	Do not reject $H_0$
$e$	3.83034	0.00247	0.0000133	0.000662	0.3024	Do not reject $H_0$
$\sqrt{2}$	3.83342	0.00061	0.0000154	0.000545	0.1666	Do not reject $H_0$
PRNG PCG64	3.83276	0.00005	0.0000113	0.000319	0.4729	Do not reject $H_0$
PRNG MT19937	3.83315	0.00034	0.0000197	0.000578	0.0381	Borderline
QRNG simulation (byte mod 10)	3.83201	0.00080	0.0000650	0.000944	$\approx 0$	Reject $H_0$

### 5.4. Discussion of Results

All six empirical datasets activate exactly 17 of the 27 algebraically possible SSD states, confirming the structural restriction proven in the theoretical section.

The mathematical constants ( $\pi$ ,  $e$ ,  $\sqrt{2}$ ) show strong agreement with the theoretical SSD distribution ( $p > 0.05$ ), supporting the hypothesis that their decimal expansions exhibit normal-like higher-order relational structure.

The PCG64 generator produces results statistically indistinguishable from theoretical expectation. The MT19937 generator yields a marginal result ( $p = 0.0381$ ), consistent with known structural correlations of the classical Mersenne Twister.

The QRNG simulation fails the chi-square test due to modulo bias introduced by mapping uniform bytes (0–255) to decimal digits via `byte mod 10`. Since 256 is not divisible by 10, digits 0–5 occur with probability 26/256 and digits 6–9 with probability 25/256, inducing systematic deviation.

Importantly, this deviation arises from digit extraction, not from the SSD framework itself.

## 6. Interpretation

The empirical results confirm the theoretical prediction that only 17 of the 27 algebraically possible SSD states are structurally realizable for decimal digit triplets. This restriction is not imposed externally, but arises intrinsically from the relational structure of ordered triples under the signed absolute-difference operator.

Across all tested domains—mathematical constants and multiple classes of random number generators—the active-state set remained invariant. The empirical entropy values are consistently close to the theoretical entropy of the SSD distribution, and the Kullback–Leibler divergences remain small. For  $\pi$ ,  $e$ ,  $\sqrt{2}$ , and PCG64, chi-square testing does not reject the null hypothesis of agreement with the theoretical SSD distribution.

The marginal deviation observed for MT19937 is consistent with the well-documented fact that classical Mersenne Twister generators may exhibit subtle structural correlations in high-dimensional projections. Importantly, the deviation remains small in magnitude.

The QRNG simulation fails the chi-square test due to digit extraction bias introduced by reducing uniform bytes modulo 10. This effect is external to the SSD framework and illustrates that SSD is sensitive to systematic structural distortions, even when they arise from preprocessing rather than from the generator itself.

Taken together, the results indicate that SSD captures second-order structural balance between adjacent differences, rather than simple digit frequency. The framework is therefore not a digit-distribution test but a relational structural invariant.

This aligns with previous observations on normality properties [? ], but extends analysis to a structured symbolic space.

## 7. Cryptographic Context

The SSD framework is not designed as a cryptographic security proof nor as a replacement for established statistical batteries used in cryptographic evaluation. Instead, it provides a structural diagnostic focused on local second-order relational balance between adjacent digits.

Classical randomness tests typically examine frequency distributions, runs, serial correlations, or transform-based properties. In contrast, SSD encodes the geometric relationship among three consecutive digits by combining sign information and relative magnitude comparison into a symbolic state.

This makes SSD sensitive to systematic distortions that do not necessarily manifest in first-order frequency tests. As demonstrated in the empirical section, even a uniformly distributed byte stream mapped to decimal digits via modulo reduction produces measurable deviation in the SSD distribution. The deviation does not originate from non-randomness of the source, but from structural imbalance introduced by digit extraction.

Therefore, SSD may be viewed as:

- a complementary structural probe rather than a standalone randomness test,
- a detector of local relational asymmetries,
- a compact invariant summarizing second-order difference geometry.

Importantly, statistical agreement with the theoretical SSD distribution does not imply cryptographic security. Conversely, deviation from the SSD distribution does not necessarily imply practical insecurity. The framework should be interpreted strictly as a structural analytical tool.

## 8. Conclusion

We introduced the Symbolic Structural Descriptor (SSD), a ternary encoding of ordered digit triplets based on signed first-order differences and their relative magnitudes.

Theoretically, although 27 symbolic combinations are algebraically possible, only 17 can occur for real digit triples. This structural restriction was formally derived and subsequently confirmed empirically.

The SSD distribution over all  $10^3$  ordered digit triplets defines a theoretical reference measure with entropy

$$H_{\text{theo}} = 3.83281 \text{ bits.}$$

Empirical evaluation over 1,000,000 digits per dataset confirms:

- Exact activation of 17 SSD states in all domains,
- Small entropy deviation from the theoretical value,
- Very low Kullback–Leibler divergence,
- Statistical agreement with the theoretical model for mathematical constants and modern PRNGs,
- Sensitivity to systematic digit extraction bias.

These findings suggest that SSD provides a compact invariant describing local relational balance in digit sequences. It is insensitive to superficial randomness while remaining responsive to structural distortions.

The framework may therefore serve as:

- A structural diagnostic for random number generators,
- A higher-order normality probe for numerical constants,
- A symbolic compression of local difference geometry.

Future work may explore analytical characterization of the 17-state manifold, asymptotic convergence rates, and extension to higher-order difference structures.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The digit sequence of  $\pi$  used in this study is publicly available.

**Acknowledgments:** The author thanks the open mathematical computing community for publicly accessible high-precision datasets.

**Conflicts of Interest:** The author declares no conflict of interest.

## Appendix A Python Code for Verification

```
import math
from collections import Counter
from itertools import product

def sgn(d):
    if d < 0: return 0
    elif d > 0: return 1
    else: return 2

def ssd_code(a,b,c):
    d11 = a-b
    d12 = b-c
    d21 = abs(d11)-abs(d12)
    return 9*sgn(d11)+3*sgn(d12)+sgn(d21)

# Theoretical distribution
theoretical = Counter()
for t in product(range(10), repeat=3):
    theoretical[ssd_code(*t)] += 1
```

```

# Normalize
theoretical_total = sum(theoretical.values())
p_theoretical = {k:v/theoretical_total for k,v in theoretical.items()}

def entropy(dist):
    return -sum(p*math.log2(p) for p in dist.values() if p>0)

def kl_divergence(p,q):
    return sum(p[i]*math.log2(p[i]/q[i]) for i in p if p[i]>0)

def psi_index(p):
    m=len(p)
    return sum((p[i]-1/m)**2 for i in p)

print("Theoretical entropy:", entropy(p_theoretical))

```

\section{Python Code for Full 27-State Verification}

```

\begin{verbatim}
import math
from collections import Counter
from itertools import product
from scipy.stats import chisquare

ALL_CODES = list(range(27))

def sgn(d):
    return 0 if d < 0 else (1 if d > 0 else 2)

def ssd_code(a, b, c):
    d11 = a - b
    d12 = b - c
    d21 = abs(d11) - abs(d12)
    return 9*sgn(d11) + 3*sgn(d12) + sgn(d21)

# --- Theoretical distribution over all 1000 digit triplets ---
theoretical = Counter()
for t in product(range(10), repeat=3):
    theoretical[ssd_code(*t)] += 1

total_theo = sum(theoretical.values())
q27 = {k: theoretical.get(k, 0)/total_theo for k in ALL_CODES}

def entropy(p27):
    return -sum(v*math.log2(v) for v in p27.values() if v > 0)

def kl_div(p27, q27):
    return sum(p27[k]*math.log2(p27[k]/q27[k])
              for k in ALL_CODES if p27[k] > 0)

```

```

def chi2_test(counts, q27, total):
    obs, exp = [], []
    for k in ALL_CODES:
        e = q27[k]*total
        o = counts.get(k, 0)
        if e > 0 or o > 0:
            obs.append(o)
            exp.append(e if e > 0 else 1e-12)
    return chisquare(obs, f_exp=exp)

def analyze(digits):
    counts = Counter(ssd_code(digits[i], digits[i+1], digits[i+2])
                    for i in range(len(digits)-2))
    N = sum(counts.values())
    p27 = {k: counts.get(k, 0)/N for k in ALL_CODES}
    return {
        "H": entropy(p27),
        "KL": kl_div(p27, q27),
        "dmax": max(abs(p27[k]-q27[k]) for k in ALL_CODES),
        "p_chi2": chi2_test(counts, q27, N)[1],
        "active": sum(1 for k in ALL_CODES if p27[k] > 0),
    }

```

## References

1. Montgomery, D.C.; Runger, G.C. *Applied Statistics and Probability for Engineers*, 6th ed.; Wiley: Hoboken, NJ, USA, 2014.
2. Mood, A.M.; Graybill, F.A.; Boes, D.C. *Introduction to the Theory of Statistics*, 3rd ed.; McGraw-Hill: New York, NY, USA, 1974.
3. Belgrade Stock Exchange. Available online: <https://www.belex.rs/> (accessed on 13 March 2023).
4. Pi Digits Source. Available online: <https://math.tools/numbers/pi/100000> (accessed on 1 March 2025).

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.