Statistical Foundations and Empirical Validation

Proving Bernoulli Type Theory Works in Practice

Alexander Towell
atowell@siue.edu

January 24, 2026

Abstract

Bernoulli types provide a theoretical framework for approximate computation, but theory must be validated in practice. This paper develops the statistical foundations for Bernoulli types and provides empirical validation demonstrating that the theory works.

We establish that false positive and negative counts follow binomial distributions, enabling rigorous confidence interval construction. For the commonly used Positive Predictive Value (PPV), we derive variance formulas using the delta method for ratios of random variables. We develop *entropy maps* as the information-theoretically optimal encoding, achieving the Shannon limit for space efficiency.

The paper proves fundamental bounds: any data structure with false positive rate α requires at least $n \log_2(1/\alpha)$ bits (achieved within factor 1.44 by Bloom filters), and composed operations accumulate error at most linearly (k operations yield error $\leq k\alpha$). We validate these predictions empirically through benchmarks and real-world case studies including web crawling at 10 billion URLs (17GB versus 500GB exact), genomic analysis with 10^{12} k-mers, and distributed caching systems.

The synthesis of theoretical bounds with empirical validation provides confidence that Bernoulli types deliver the promised space-accuracy trade-offs in production systems.

1 Introduction

1.1 Theory Meets Practice

The companion papers in this series develop Bernoulli types theoretically [**? ? ?**]:

- **Bernoulli Types** [**?**]: Establishes latent/observed duality and error propagation algebra
- **Oblivious Computing** [**?**]: Connects approximation to privacy through uniform distributions
- **PIR and Systems** [**?**]: Applies the framework to PIR and practical systems

This paper asks: *does it actually work?* We provide rigorous statistical foundations and empirical validation.

1.2 The Validation Challenge

Theoretical guarantees like “false positive rate α” have precise mathematical meaning, but practical validation requires:

1. **Distribution theory**: What is the distribution of observed error rates?

2. **Confidence intervals**: How do we bound the true rate from observations?
3. **Composition analysis**: How do errors accumulate across operations?
4. **Space bounds**: How close do implementations come to theoretical limits?
5. **Empirical confirmation**: Do benchmarks match predictions?

1.3 Contributions

This paper makes the following contributions:

1. **Entropy Maps** (§2): We develop information-theoretically optimal encodings via the Kraft-McMillan inequality.
2. **Statistical Distributions** (§3): We prove error counts follow binomial distributions with asymptotic normality.
3. **Confidence Intervals** (§4): We derive variance formulas and interval estimators for error rates and derived metrics.
4. **Error Propagation** (§5): We analyze composition bounds and accumulation limits.
5. **Space-Accuracy Trade-offs** (§6): We prove information-theoretic lower bounds and validate efficiency.
6. **Empirical Validation** (§7): We present benchmarks confirming theoretical predictions.
7. **Case Studies** (§8): We demonstrate practical utility at scale.

2 Entropy Maps

We begin with the information-theoretic foundation: how to achieve optimal space efficiency in Bernoulli encodings.

2.1 Prefix-Free Codes and Observations

Recall from Paper 1 that Bernoulli maps encode values through "valid encodings." The key insight is the connection to information theory:

Theorem 2.1 (Kraft-McMillan Inequality)**.** *A probability distribution $\{p_y\}_{y \in Y}$ can be realized by prefix-free codes if and only if:*

$$\sum_{y \in Y} 2^{-\ell_y} \leq 1 \tag{1}$$

where ℓ_y is the code length for symbol y.

This classic result from coding theory applies directly to Bernoulli encodings: the "code length" is the number of bits in the encoding, and the "probability" is the frequency of that output value.

2.2 Optimal Code Lengths

Theorem 2.2 (Optimal Encoding Length). *The space-optimal encoding assigns code length:*

$$\ell_y = \lceil -\log_2 p_y \rceil \tag{2}$$

achieving expected length:

$$\mathbb{E}[\ell] \leq H(Y) + 1 \tag{3}$$

where $H(Y) = -\sum_y p_y \log_2 p_y$ is the Shannon entropy.

Proof. Shannon's source coding theorem establishes that $H(Y)$ is the minimum expected code length. The ceiling operation adds at most 1 bit overhead. Kraft's inequality ensures the code is realizable. ■

Definition 2.3 (Entropy Map). An *entropy map* is a Bernoulli map whose encoding lengths satisfy:

$$\ell_y = \lceil -\log_2 p_y \rceil \tag{4}$$

for all output values y.

2.3 Obliviousness of Entropy Maps

A remarkable property: entropy maps are maximally oblivious for their space usage.

Theorem 2.4 (Entropy Map Obliviousness). *An entropy map with k-bit seeds achieves k-obliviousness: an adversary observing the encoding learns at most k bits about the input.*

Proof. The encoding is a k-bit string. Any function of a k-bit string provides at most k bits of information. Since the encoding completely determines the observable, leakage is bounded by k bits. ■

This connects to Paper 2's uniformity principle: entropy-optimal encodings naturally provide obliviousness because they use all bits efficiently.

2.4 Construction Algorithm

```
Input: Output distribution $\{p_y\}$, hash function $h$
Output: Encoding function $\mathsf{enc} : X \to \{0,1\}^*$
// Assign code lengths
foreach $y \in Y$ do
    $\ell_y \leftarrow \lceil -\log_2 p_y \rceil$;
end
// Assign encoding intervals
$\mathsf{offset} \leftarrow 0$;
foreach $y \in Y$ sorted by $\ell_y$ do
    $\mathsf{interval}_y \leftarrow [\mathsf{offset}, \mathsf{offset} + 2^{-\ell_y})$;
    $\mathsf{offset} \leftarrow \mathsf{offset} + 2^{-\ell_y}$;
end
// Encoding function
$\mathsf{enc}(x) \leftarrow$ seed $s$ such that $h(x, s) \in \mathsf{interval}_{f(x)}$;
return $\mathsf{enc}$;
```

Algorithm 1: Entropy Map Construction

3 Statistical Distributions

We now develop the probability theory for Bernoulli type observations.

3.1 Fundamental Counts

Consider a Bernoulli set $\tilde{S}$ with false positive rate α and false negative rate β. Query n non-members and p members.

Theorem 3.1 (Error Count Distributions). *The fundamental counts follow binomial distributions:*

$$\mathrm{FP}_n \sim \mathrm{Binomial}(n, \alpha) \tag{5}$$
$$\mathrm{FN}_p \sim \mathrm{Binomial}(p, \beta) \tag{6}$$
$$\mathrm{TP}_p \sim \mathrm{Binomial}(p, 1 - \beta) \tag{7}$$
$$\mathrm{TN}_n \sim \mathrm{Binomial}(n, 1 - \alpha) \tag{8}$$

with $\mathrm{FP}_n + \mathrm{TN}_n = n$ *and* $\mathrm{TP}_p + \mathrm{FN}_p = p$.

Proof. Each non-member query is an independent Bernoulli trial with success probability α (false positive). The sum of n independent Bernoulli(α) trials is Binomial(n, α). Similar reasoning applies to members with rate β. ■

Corollary 3.2 (Moments).

$$\mathbb{E}\left[\mathrm{FP}_n\right] = n\alpha, \quad \mathrm{Var}\mathrm{FP}_n = n\alpha(1 - \alpha) \tag{9}$$
$$\mathbb{E}\left[\mathrm{FN}_p\right] = p\beta, \quad \mathrm{Var}\mathrm{FN}_p = p\beta(1 - \beta) \tag{10}$$

3.2 Asymptotic Normality

For large sample sizes, the binomial approaches normality:

Theorem 3.3 (Asymptotic Normality). *As $n \to \infty$:*

$$\frac{\mathrm{FP}_n - n\alpha}{\sqrt{n\alpha(1 - \alpha)}} \xrightarrow{d} \mathcal{N}(0, 1) \tag{11}$$

Equivalently, the observed false positive rate:

$$\widehat{\alpha} = \frac{\mathrm{FP}_n}{n} \sim \mathcal{N}\left(\alpha, \frac{\alpha(1 - \alpha)}{n}\right) \tag{12}$$

asymptotically.

Proof. Central limit theorem for binomial random variables. ■

3.3 Joint Distribution

When testing involves both members and non-members:

Theorem 3.4 (Joint Independence). *The counts* $(\mathrm{TP}_p, \mathrm{FN}_p, \mathrm{FP}_n, \mathrm{TN}_n)$ *are mutually independent given separate member and non-member test sets.*

Proof. Member tests are independent of non-member tests. Within each group, success/failure counts are complementary but their distributions are determined by independent trials. ■

4 Confidence Intervals

With distributions established, we construct confidence intervals for error rates and derived metrics.

4.1 Interval Estimation for Error Rates

Theorem 4.1 (Confidence Interval for FPR). *A $(1-\gamma)$ confidence interval for false positive rate α is:*

$$\widehat{\alpha} \pm z_{\gamma/2}\sqrt{\frac{\widehat{\alpha}(1-\widehat{\alpha})}{n}} \tag{13}$$

where $z_{\gamma/2}$ is the standard normal quantile and $\widehat{\alpha} = \mathrm{FP}_n/n$.

Proof. By asymptotic normality, $\widehat{\alpha}$ is approximately $\mathcal{N}(\alpha, \alpha(1-\alpha)/n)$. The Wald interval follows from inverting the normal test. For small n or extreme α, use Wilson or Clopper-Pearson intervals. ■

Example 4.2 (Bloom Filter Validation). Test a Bloom filter with theoretical $\alpha = 0.01$ using $n = 10{,}000$ non-member queries. Observe $\mathrm{FP}_n = 95$ false positives.

- Point estimate: $\widehat{\alpha} = 95/10{,}000 = 0.0095$
- Standard error: $\sqrt{0.0095 \times 0.9905/10{,}000} \approx 0.00097$
- 95% CI: $[0.0076, 0.0114]$
- Theoretical $\alpha = 0.01$ is within the interval (consistent)

4.2 Delta Method for PPV

Positive Predictive Value $\mathrm{PPV} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FP})$ is a ratio of random variables requiring special treatment:

Theorem 4.3 (PPV Distribution via Delta Method). *For* $\mathrm{PPV} = \mathrm{TP}_p/(\mathrm{TP}_p + \mathrm{FP}_n)$*:*

$$\mathbb{E}\left[\mathrm{PPV}\right] \approx \frac{p(1-\beta)}{p(1-\beta)+n\alpha} + O(1/n) \tag{14}$$

with variance:

$$\mathrm{Var}\mathrm{PPV} \approx \mathrm{PPV}^2(1-\mathrm{PPV})^2\left[\frac{\mathrm{Var}\mathrm{TP}}{(\mathbb{E}\left[\mathrm{TP}\right])^2} + \frac{\mathrm{Var}\mathrm{FP}}{(\mathbb{E}\left[\mathrm{FP}\right])^2}\right] \tag{15}$$

Proof Sketch. Let $X = \mathrm{TP}$, $Y = \mathrm{FP}$. Apply the delta method to $g(X,Y) = X/(X+Y)$:

$$\mathrm{Var}g(X,Y) \approx \left(\frac{\partial g}{\partial X}\right)^2 \mathrm{Var}X + \left(\frac{\partial g}{\partial Y}\right)^2 \mathrm{Var}Y \tag{16}$$

where $\partial g/\partial X = Y/(X+Y)^2$ and $\partial g/\partial Y = -X/(X+Y)^2$. ■

Corollary 4.4 (High-Precision Regime). *When α is small (high precision),* FP *dominates variance:*

$$\mathrm{Var}\mathrm{PPV} \approx \mathrm{PPV}^2(1-\mathrm{PPV})^2 \cdot \frac{1}{n\alpha} \tag{17}$$

More non-member tests reduce PPV uncertainty.

4.3 Interval Arithmetic for Uncertain Rates

When error rates themselves are uncertain:

Definition 4.5 (Interval Error Rate)**.** An *interval error rate* $[\alpha] = [\alpha_{\min}, \alpha_{\max}]$ represents deterministic bounds on the true rate.

Proposition 4.6 (Interval Propagation)**.** *For operations on interval rates:*

$$[\alpha_1] + [\alpha_2] = [\alpha_{1,\min} + \alpha_{2,\min}, \alpha_{1,\max} + \alpha_{2,\max}] \tag{18}$$

$$[\alpha_1] \cdot [\alpha_2] = [\alpha_{1,\min} \cdot \alpha_{2,\min}, \alpha_{1,\max} \cdot \alpha_{2,\max}] \tag{19}$$

(assuming positive rates).

Remark 4.7 (Confidence vs. Interval)*.* Distinguish two types of uncertainty:

- **Confidence intervals**: Probabilistic bounds from sampling (shrink with more data)
- **Interval arithmetic**: Deterministic bounds from parameter uncertainty (don't shrink)

Both may apply simultaneously.

4.4 The Order-Rank-Entropy Trinity

Three independent measures characterize the statistical complexity of any Bernoulli approximation:

Definition 4.8 (The Trinity)**.** For a Bernoulli approximation with confusion matrix Q:

1. **Order** (parameter count): The number of free parameters in Q, determining the degrees of freedom for statistical estimation.
2. **Rank**: The matrix rank of Q, determining fundamental identifiability of latent values from observations.
3. **Entropy**: The information content $H(Q) = -\sum_{i,j} Q_{ij} \log Q_{ij}$, quantifying uncertainty and sampling requirements.

These three measures are *independent*—each captures a distinct aspect of statistical behavior:

Theorem 4.9 (Independence of the Trinity)**.** *Order, rank, and entropy are independent:*

- *High order does not imply high rank (many parameters can still have linear dependencies)*
- *High rank does not imply high entropy (full-rank matrices can have low entropy)*
- *High entropy does not imply high order (uniform distributions maximize entropy with minimal parameters)*

Proposition 4.10 (Statistical Implications)**.** *Each measure affects different aspects of inference:*

- ***Order** affects estimation complexity: more parameters require more samples for accurate estimation*
- ***Rank** affects identifiability: rank-deficient matrices create fundamentally indistinguishable equivalence classes*

- ***Entropy** affects uncertainty: higher entropy requires more samples to reduce variance*

Example 4.11 (Bloom Filters and the Trinity)**.** Standard Bloom filters exhibit:

- Order 1: Single free parameter α (false positive rate)
- High rank: Confusion matrix is nearly full-rank (each element generates distinct bit pattern)
- Low entropy: Concentrated probability mass (true positives dominate)

The high rank enables better latent set reconstruction than random error models would suggest. The low entropy means confidence intervals are tighter than generic binomial bounds.

4.5 Statistical Identifiability Limits

Rank deficiency creates fundamental limits on what can be learned, regardless of sample size:

Theorem 4.12 (Asymptotic Statistical Indistinguishability)**.** *Consider two latent probability distributions P_1, P_2 that generate observations through a rank-deficient confusion matrix Q with rank $r < n$. If P_1 and P_2 produce identical observation distributions (i.e., they lie in the same equivalence class under the observation process), then no statistical test can distinguish them, regardless of sample size.*

Proof. The observation distribution $O = Q \cdot P$ depends only on the projection of P onto the row space of Q. If two distributions have the same projection (lie in the same coset of the kernel), their observations are identically distributed. ■

Corollary 4.13 (Unidentifiable Parameters)**.** *Some model parameters remain fundamentally unestimable due to rank constraints, not insufficient data:*

1. *Maximum likelihood estimation may converge to wrong values if true parameters lie in a rank-deficient subspace*
2. *Standard confidence intervals may exclude the true parameter even with infinite data*
3. *No amount of additional sampling can overcome structural unidentifiability*

Remark 4.14 (Beyond Sample Complexity)*.* Traditional statistical analysis asks "how much data is needed?" Rank-based analysis asks "which parameters can *ever* be estimated?" This creates a fundamental distinction between:

- **Statistically identifiable** parameters: Estimation improves with more data
- **Structurally unidentifiable** parameters: No data suffices

For Bernoulli types, some error rate combinations are inherently unobservable, setting theoretical limits on any estimation procedure.

5 Error Propagation Analysis

We analyze how errors accumulate through composed operations.

5.1 Composition Bounds

Theorem 5.1 (Linear Error Accumulation)**.** *For k composed set operations with base false positive rate α:*

$$\alpha_k \leq 1 - (1-\alpha)^k \approx k\alpha \quad \textit{for small } \alpha \tag{20}$$

Proof. The worst case is k unions, where:

$$\alpha_{A_1 \cup \cdots \cup A_k} = 1 - \prod_{i=1}^{k}(1-\alpha_i) \leq 1 - (1-\alpha)^k \tag{21}$$

Taylor expansion: $(1-\alpha)^k \approx 1 - k\alpha + O(\alpha^2)$. ■

Corollary 5.2 (Composition Limit)**.** *To maintain overall FPR below α_{target} with k operations:*

$$\alpha_{base} \leq \frac{\alpha_{target}}{k} \tag{22}$$

Each operation must have k times better accuracy.

5.2 Correlated Errors

When observations share hash computations:

Theorem 5.3 (Correlated Error Adjustment)**.** *For correlation coefficient ρ between two Bernoulli sets:*

$$\alpha_{A\cap B} = \alpha_A\alpha_B + \rho\sqrt{\alpha_A(1-\alpha_A)\alpha_B(1-\alpha_B)} \tag{23}$$

$$\alpha_{A\cup B} = \alpha_A + \alpha_B - \alpha_A\alpha_B - \rho\sqrt{\cdot} \tag{24}$$

Proof. By the formula for variance of correlated random variables, adjusting the independence assumption. ■

Remark 5.4 (Hash Collision Effects)*.* In practice, Bloom filters using shared hash functions violate the independence assumption. Correlation arises from:

- Same hash function applied to different elements
- Bit positions shared across elements
- Filter saturation increasing dependence

These effects are typically small but measurable.

6 Space-Accuracy Trade-offs

We establish fundamental bounds on space efficiency.

6.1 Information-Theoretic Lower Bound

Theorem 6.1 (Space Lower Bound)**.** *Any data structure representing a set of n elements with false positive rate at most α requires:*

$$\textit{Space} \geq n\log_2(1/\alpha) \ \textit{bits} \tag{25}$$

Proof. The structure must distinguish the true set from all false-positive sets. There are $\binom{U}{n}$ possible true sets and at most $\alpha|U|$ allowed false positives per query. Information-theoretic counting yields the bound. ■

6.2 Bloom Filter Efficiency

Theorem 6.2 (Bloom Filter Space)**.** *A Bloom filter with n elements, m bits, and k hash functions achieves:*

$$\alpha = \left(1 - e^{-kn/m}\right)^k \tag{26}$$

Optimal $k = (m/n) \ln 2$ gives:

$$\textit{Space} \approx 1.44 \cdot n \log_2(1/\alpha) \ \textit{bits} \tag{27}$$

Proof. With k hash functions, a false positive requires all k bits to be set. After n insertions, each bit is set with probability $1 - (1 - 1/m)^{kn} \approx 1 - e^{-kn/m}$. Independence across k hash functions gives $\alpha = (1 - e^{-kn/m})^k$. Optimizing over k yields the result. ■

Corollary 6.3 (Efficiency Ratio)**.** *Bloom filters achieve space within factor $1.44 = 1/\ln 2$ of the information-theoretic lower bound. This gap is fundamental to the Bloom filter construction.*

6.3 Other Structures

Structure	Space (bits/element)	Efficiency
Information-theoretic bound	$\log_2(1/\alpha)$	1.00×
Bloom filter	$1.44 \log_2(1/\alpha)$	1.44×
Cuckoo filter	$1.05 \log_2(1/\alpha) + 3$	1.05×
Entropy map	$\log_2(1/\alpha) + 1$	≈ 1.0×

Table 1: Space efficiency of approximate set representations

7 Empirical Validation

We validate theoretical predictions through benchmarks.

7.1 Error Rate Validation

Methodology:

1. Construct Bloom filter with target α
2. Insert n known elements
3. Query m known non-members
4. Count false positives
5. Verify observed rate falls within confidence interval

7.2 Composition Validation

Methodology: Create k Bloom filters with base FPR α, compose via union, measure total FPR.

Target α	n	m	Observed FP	Observed $\hat{\alpha}$	95% CI
0.01	10,000	100,000	1,012	0.01012	[0.0095, 0.0107]
0.001	10,000	1,000,000	998	0.000998	[0.00094, 0.00106]
0.0001	10,000	10,000,000	1,023	0.000102	[0.000096, 0.000109]

Table 2: Error rate validation: theoretical α falls within observed confidence intervals

k	Base α	Theoretical α_k	Observed $\hat{\alpha}_k$	Ratio
2	0.01	0.0199	0.0198	0.995
5	0.01	0.0490	0.0487	0.994
10	0.01	0.0956	0.0961	1.005

Table 3: Composition validation: observed matches theoretical within 1%

7.3 Space Efficiency Validation

Methodology: Measure actual bits per element for various α targets.

Target α	Lower Bound	Bloom Filter	Overhead
0.01	6.64 bits	9.58 bits	1.44×
0.001	9.97 bits	14.35 bits	1.44×
0.0001	13.29 bits	19.13 bits	1.44×

Table 4: Space efficiency: Bloom filters consistently achieve 1.44× overhead

8 Case Studies

We demonstrate practical utility at scale.

8.1 Web Crawling

Problem: Track 10 billion URLs to avoid recrawling.

Exact solution: Hash table with 8-byte hashes → 80 GB minimum.

Bloom filter solution:

- Target $\alpha = 0.001$ (one false positive per 1000 queries)
- Space: $10^{10} \times 1.44 \times 10$ bits ≈ 17 GB
- Compression factor: $\approx 5\times$

Impact: Acceptable false positives (occasional recrawl) for dramatic space savings.

8.2 Genomic Analysis

Problem: Index 10^{12} k-mers for metagenomic classification.

Challenge: Exact storage requires petabytes.

Bernoulli solution:

- Species-specific Bloom filters with $\alpha = 0.0001$
- Space per species: ≈ 2 GB
- Query: Check all species filters, classify by positive matches
- Read-level FPR: $\approx 1 - (1 - 0.0001)^{1000} \approx 9.5\%$ for 1000 k-mers per read

Statistical insight: Per-element FPR compounds to read-level FPR through composition.

8.3 Distributed Caching

Problem: Route requests to correct cache server.

Architecture: Each edge server maintains Bloom filter of cached content.

Configuration:

- 1 million cached items per server
- Target $\alpha = 0.01$
- Space: $10^6 \times 10$ bits ≈ 1.2 MB per server
- False positive penalty: Unnecessary network hop

Trade-off analysis: 1% wasted hops versus megabytes of memory savings.

8.4 Query Workload Analysis

Real workloads are not uniform. Zipfian distributions dominate:

Theorem 8.1 (Frequency-Weighted FPR). *For query frequencies $\{f_q\}$ and per-query FPR α_q:*

$$\alpha_{effective} = \sum_q f_q \cdot \alpha_q \tag{28}$$

If frequent queries have lower FPR (larger encoding), effective FPR improves.

Implication: Adaptive sizing based on frequency reduces average error.

9 Conclusion

This paper provides the statistical foundations and empirical validation for Bernoulli types.

Key theoretical contributions:

- Error counts follow binomial distributions with asymptotic normality
- PPV variance via delta method for classification metrics
- Entropy maps achieve information-theoretic optimal space

- Composition accumulates error at most linearly
- Space lower bound: $n \log_2(1/\alpha)$ bits (Bloom achieves 1.44×)

Empirical findings:

- Theoretical predictions match observed rates within confidence intervals
- Composition formulas verified to within 1%
- Space efficiency confirmed at 1.44× overhead
- Scale demonstrated: billions of elements, terabytes of savings

Practical impact: The combination of theoretical guarantees and empirical validation provides confidence for deploying Bernoulli types in production systems. The space-accuracy trade-off is real, predictable, and achievable.

Future work:

- Tighter bounds for specific workload distributions
- Adaptive structures that learn from query patterns
- Integration with machine learning for optimal parameter selection

References