IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 2017, No. 3, pp. 151–168. DOI:10.13154/tosc.v2017.i3.151-168

Analysis and Improvement of Entropy Estimators in NIST SP 800-90B for Non-IID Entropy Sources

Shuangyi Zhu[1,2,3], Yuan Ma[1,2*], Tianyu Chen[1,2], Jingqiang Lin[1,2,3] and Jiwu Jing[1,2,3]

[1] Data Assurance and Communications Security Research Center, Chinese Academy of Sciences, Beijing, China
{zhushuangyi,yma,tychen,linjq,jing}@is.ac.cn
[2] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[3] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

Abstract. Random number generators (RNGs) are essential for cryptographic applications. In most practical applications, the randomness of RNGs is provided by entropy sources. If the randomness is less than the expected, the security of cryptographic applications could be undermined. Accurate entropy estimation is a critical method for the evaluation of RNG security, and significant overestimation and underestimation are both inadvisable. The NIST Special Publication 800-90B is one of the most common certifications for entropy estimation. It makes no assumption of the entropy source and provides min-entropy estimation results by a set of entropy estimators. It estimates the entropy sources in two tracks: the IID (independent and identically distributed) track and non-IID track. In practice, non-IID entropy sources are more common, as physical phenomenon, sampling process or external perturbation could cause the dependency of the outputs.
In this paper, we prove that the Collision Estimate and the Compression Estimate in non-IID track could provide significant underestimates in theory. In order to accurately estimate the min-entropy of non-IID sources, we provide a formula of min-entropy based on conditional probability, and propose a new estimator to approximate the result of this formula. Finally, we perform experiments to compare our estimator with the NIST estimators using simulated non-IID data. Results show that our estimator gives close estimates to the real min-entropy.

Keywords: Entropy estimation · NIST SP 800-90B · Min-entropy · Random number generator

1 Introduction

Random number generators (RNGs) play an important role in cryptographic systems. They are widely employed in generating secret keys, initialization vectors, nonces, and so on. The security of most cryptographic schemes and protocols is based on the randomness of output numbers generated by RNGs. RNGs are classified into two types: pseudo-random number generators (PRNGs) and true random number generators (TRNGs). In general, PRNGs use deterministic algorithms to extend the seeds for generating long sequences, and the seeds are generated by TRNGs based on entropy sources.

*Yuan Ma is the corresponding author.

Received: 2017-06-01, Accepted: 2017-08-01, Published: 2017-09-19

The security of PRNGs is based on the randomness of their seeds. There have been many examples of systems or platforms which use PRNG seeds with insufficient randomness. Naturally and inevitably, some cryptographic keys generated in these systems or platforms were compromised, as in [DGP09, GPR06, VP16]. Therefore, it is critical to assess and quantify the randomness (i.e., express the level of randomness as an amount) of the entropy source. Randomness has three features: unpredictability, unbiasedness and non-repeatability. Several commonly used statistical test suites are to check whether the outputs are of perfect randomness, such as Diehard [Mar] and the NIST SP 800-22 [R+], but they are not able to quantify the randomness. Instead, people use the concept of entropy to measure the unbiasedness and unpredictability of the entropy source's outputs, and the non-repeatability is validated by restart tests. There are many possible types of entropy; for cryptographic applications, the min-entropy is a proper choice, because min-entropy corresponds to the difficulty of guessing the most likely output of the entropy source [T+16].

Nowadays, there are two main security certifications including entropy estimation. One is the AIS 20/31 [KS] developed by German BSI (Federal Office for Information Security). In this certification, TRNG designers are required to provide a stochastic model of the entropy source, so the entropy can be estimated in theory according to the model. The other certification is the estimation suite in the NIST Special Publication (SP) 800-90B [T+16], and the latest version is the second draft published in 2016. Different with the AIS 20/31, the SP 800-90B provides a serial of statistical algorithms to estimate min-entropy with no assumption. It first checks whether the outputs are IID through a set of statistical tests. In IID track, it only employs one basic estimator; in non-IID track, it further employs additional 9 entropy estimators. The minimum estimate of all estimators is selected as the final result. These two certifications have different emphases. The AIS 20/31 is a more stringent but risky approach, as the entropy estimation heavily depends on the provided stochastic model. On the contrary, the black-box NIST estimation suite may not be very precise, but less risky. Therefore, the NIST certification is more convenient for designers and verifiers to estimate the entropy of entropy sources.

There are a lot of statistical methods on entropy estimation based on output data, some of which have been adopted by the second draft of NIST SP 800-90B. The basic type is the frequency-counts estimator, which approximates the distribution of the outputs on the basis of frequency, such as [Gra03, Pan03, Rou99]. In another way, some use Bayesian methods to estimate the entropy, such as [WW94, NSB01]. Besides these two main methods, compression algorithms are employed in entropy estimation, such as [KASW98, WZ89], and Hagerty and Draper [HD] proposed a conception of "entropic" statistics and used them to estimate the entropy. At CHES 2015, Kelsey *et al.* [KMT15] provided novel predictive models for entropy estimation. This method is based on the knowledge of machine learning, which is a new approach included in the second draft of NIST SP 800-90B.

Motivation. The current version of NIST SP 800-90B [T+16] consists 10 capable estimators, each of which targets one or more properties (such as periodicity) of the assessed entropy source, so overestimation seldom occurs in the final result. Meanwhile, the underestimation of each estimator has a negative impact on the performance of the suite, as the minimum of all estimates is chosen as the final result. The significant underestimations of some estimators have been pointed out by Kelsey *et al.* [KMT15], but the reasons have not been revealed. Furthermore, it is still a challenge to accurately estimate the entropy for non-IID entropy sources, due to the contained complex dependency in these sources.

Our contributions. In this paper, we study the min-entropy estimation methods of SP 800-90B for non-IID entropy sources, especially for the Collision Estimate and the Compression Estimate. We explain the reasons of underestimation for the Collision Estimate and the Compression Estimate by theoretical analysis and experimental validation. In order to accurately estimate the min-entropy of non-IID sources, we provide a formula of

min-entropy based on conditional probability, and propose a new estimator to approximate the result of this formula. Finally, we perform experiments to compare our estimator with the NIST estimators using simulated non-IID data.

In summary, we make the following contributions.

1. We explain the reasons of the underestimation for the Collision Estimate and the Compression Estimate for the first time. Figuring out the reason of this underestimation has a practical significance for improving the draft of SP 800-90B and guiding the design of similar estimators.

2. We provide a specific formula of min-entropy for non-IID entropy sources, and propose a new estimator for non-IID track based on our formula, especially for the entropy source based on Markov model. Comparison result with the Markov Estimate shows that, our estimator has smaller statistical error, lower computational overhead, and wider scope of application.

3. We compare the performance between our estimator and the NIST estimators, using two types of non-IID data with known entropy values. Our estimator gives close estimates to the real min-entropy, and its performance is comparable or much better than that of the NIST estimators.

Organization. The rest of this paper is organized as follows. In Sect. 2, we introduce the definition of min-entropy and the NIST estimation suite. In Sect. 3, we explain the underestimation for the Collision Estimate and the Compression Estimate, and perform the theoretical analysis and experimental validation. In Sect. 4, we propose our estimator and compare it with the Markov Estimate and prediction based estimators in the NIST estimation suite. We also perform the experiment to compare the performance between our estimator and the NIST estimators. In Sect. 5, we conclude the paper.

2 Preliminary

In this section, we introduce the definition and formula of the min-entropy. We also introduce the entropy estimation procedure and the estimators employed in the second draft of SP 800-90B [T+16].

2.1 Min-entropy of Entropy Sources

The entropy source has been defined in [T+16]. As assumed in [T+16], the outputs obtained from an entropy source take values from a finite alphabet.

We take the next output from the entropy source as a random variable X. For the independent discrete random variable X, the NIST SP 800-90B [T+16] gives the definition of min-entropy: if X takes value from the set $A = \{x_1, x_2, ..., x_k\}$ with probability $\Pr\{X = x_i\} = p_i$ for $i = 1, ..., k$, the min-entropy

$$\begin{aligned} H &= \min_{1 \le i \le k} (-\log_2 p_i) \\ &= -\log_2 \max_{1 \le i \le k} p_i. \end{aligned} \tag{1}$$

If the min-entropy of X is h, then the occurring probability of any particular value is no greater than 2^{-h}. The maximum possible value for min-entropy of an IID random variable with k distinct values is $\log_2 k$, which is achieved when the random variable has an uniform probability distribution, i.e., $p_1 = p_2 = ... = p_k = \frac{1}{k}$.

For Markov process, Kelsey *et al.* gave a definition of min-entropy in [KMT15]. A stochastic process $\{s_n\}_{n\in\mathbb{N}}$ that takes values from a finite set $A = \{x_1, ..., x_k\}$ is called a first-order Markov chain, if

$$\Pr\{s_{n+1} = q_{n+1}|s_n = q_n, s_{n-1} = q_{n-1}, ..., s_0 = q_0\} = \Pr\{s_{n+1} = q_{n+1}|s_n = q_n\},$$

where $q_0, ..., q_{n+1} \in A$. For a d^{th}-order Markov Model, the transition probabilities satisfy

$$\Pr\{s_{n+1} = q_{n+1}|s_n = q_n, ..., s_0 = q_0\} = \Pr\{s_{n+1} = q_{n+1}|s_n = q_n, ..., s_{n-d+1} = q_{n-d+1}\}.$$

The min-entropy of the first-order Markov chain of length $c + 1$ is defined as

$$H = -\log_2\big(\max_{(q_0,...,q_c)\in A^{c+1}} p_{q_0}\prod_{j=1}^{c} p_{q_j q_{j+1}}\big), \tag{2}$$

where $p_{q_0} = \Pr\{s_0 = q_0\}$ and $p_{q_j q_{j+1}} = \Pr\{s_{j+1} = q_{j+1}|s_j = q_j\}$. The entropy per sample can be approximated by dividing H by $c + 1$.

2.2 Entropy Estimation Procedure in Draft SP 800-90B

Firstly, the tested data from entropy sources are divided into two tracks through a set of statistical tests: IID track and non-IID track. Then, different estimators are employed in different tracks. There are 10 estimators for non-IID track: the Most Common Value Estimate, the Collision Estimate, the Markov Estimate, the Compression Estimate, the t-Tuple Estimate, the Longest Repeated Substring (LRS) Estimate, the Multi Most Common in Window (MultiMCW) Prediction Estimate, the Lag Prediction Estimate, the MultiMMC Prediction Estimate, and the LZ78Y Prediction Estimate. According to the underlying methods they employ, we divide these estimators into 3 classes: *basic type*, *statistic based type*, and *prediction based type*, which are shown in Table 1. For IID track, only the Most Common Value Estimate is employed.

In the estimation process, each estimator calculates its own estimate independently. Among all estimates, the minimum one is selected as the final estimate for the entropy source.

Table 1: Classification of NIST estimators

Basic type	**Statistic based type**	**Prediction based type**
Most Common Value Estimate	Collision Estimate	MultiMCW Prediction Estimate
t-Tuple Estimate		Lag Prediction Estimate
LRS Estimate	Compression Estimate	MultiMMC Prediction Estimate
Markov Estimate		LZ78Y Prediction Estimate

2.3 Estimators in Draft SP 800-90B

We first introduce the basic type estimators. The Most Common Value Estimate computes entropy based on the frequency of the most common value concluded in the tested data. The t-Tuple Estimate and LRS Estimate are extensions of the Most Common Value Estimate. They compute entropy based on the frequency of the most common tuples in the tested data. The Markov Estimate computes entropy by assuming that the tested data obey a first-order Markov model.

The following are the detailed introductions to the statistic based and the prediction based estimators included in the second draft of SP 800-90B.

The Collision Estimate. This estimator is proposed by Hagerty and Draper [HD]. In the tested data sequence, if any value repeats, we call there exists a collision. This estimator calculates the mean number of samples to the first collision as the statistic. On the basis of this statistic, it provides an estimate of the probability of the most-likely output value. The following are the calculation steps of the Collision Estimate, given the inputs $S = (s_1, ..., s_L)$, where $s_i \in A = \{x_1, ..., x_k\}$.

1. Let v denote the total number of collisions in S. Let t_i denote the number of samples that are used to generate the i^{th} collision. Calculate the mean $\bar{X}$ that is the statistic, and the sample standard deviation $\hat{\sigma}$, of t_i as

$$\bar{X} = \frac{1}{v}\sum_{i=1}^{v} t_i,\ \hat{\sigma} = \sqrt{\frac{1}{v}\sum_{i=1}^{v}(t_i - \bar{X})^2}.$$

2. Compute the lower bound of the 99% confidence interval for the statistic, based on a normal distribution,

$$\bar{X'} = \bar{X} - 2.576\frac{\hat{\sigma}}{\sqrt{v}}.$$

3. Let k be the number of possible values of the inputs. Using a binary search, solve for the parameter p, such that

$$\bar{X'} = pq^{-2}\big(1 + \frac{1}{k}(p^{-1} - q^{-1})\big)F(q) - pq^{-1}\frac{1}{k}(p^{-1} - q^{-1}), \tag{3}$$

where

$$q = \frac{1-p}{k-1},$$

$$p \geq q,$$

$$F(\frac{1}{z}) = \Gamma(k+1, z)z^{-k-1}e^{z},$$

and $\Gamma(a, b)$ is the incomplete Gamma function,

$$\Gamma(a, b) = \int_b^{+\infty} x^{a-1}e^{-x}dx.$$

4. If the binary search yields a solution, then the min-entropy estimate is calculated as $-\log_2(p)$, otherwise the min-entropy estimate is calculated as $\log_2(k)$.

The Compression Estimate. This estimator computes the entropy rate of inputs on the basis of Maurer Universal Statistic presented in [Mau92]. In order to calculate the statistic, the inputs are first partitioned into two disjoint parts. The first part is used to establish a dictionary for the universal compression algorithm; the second part serves as the test group to calculate the Maurer statistic. The calculation steps in the Compression Estimate are listed as follows, given the inputs $S = (s_1, ..., s_L)$, where $s_i \in A = \{x_1, ..., x_k\}$.

1. Partition the inputs into two disjoint groups.
   (a) Create the dictionary from the first $d = 1000$ observations, $(s_1, s_2, ..., s_d)$.
   (b) Use the remaining $v = L - d$ observations, $(s_{d+1}, ..., s_L)$, for testing.

2. The Maurer statistic is computed as:

$$\bar{X} = \frac{1}{v} \sum_{i=d+1}^{L} \log_2 A_i,$$

where

$$A_i = \begin{cases} i, & \text{if } s_i \neq s_j \ \forall\ 0 < j < i, \\ i - \max\{j : j < i \text{ and } s_j = s_i\}, & \text{otherwise.} \end{cases} \tag{4}$$

3. Compute the lower bound of the 99% confidence interval for $\bar{X}$, using

$$\bar{X}' = \bar{X} - \frac{2.576\hat{\sigma}}{\sqrt{v}},$$

where

$$\hat{\sigma} = \alpha\sqrt{\frac{\sum_{i=d+1}^{L}(\log_2 A_i)^2}{v} - \bar{X}^2}, \quad \alpha = 0.7 - \frac{0.8}{b} + \frac{(4 + \frac{32}{b})v^{-3/b}}{15},$$

and $b = \lfloor \log_2 k \rfloor + 1$.

4. Using a binary search, solve for the parameter p, such that following equation is satisfied:

$$\bar{X}' = G(p) + (k-1)G(q) \tag{5}$$

where

$$G(z) = \frac{1}{v} \sum_{t=d+1}^{L} \sum_{u=1}^{t} \log_2(u) F(z,t,u), \quad q = \frac{1-p}{k-1},$$

and

$$F(z,t,u) = \begin{cases} z^2(1-z)^{u-1}, & if\ u < t, \\ z(1-z)^{t-1}, & if\ u = t. \end{cases}$$

5. If the binary search yields a solution, then the min-entropy estimate is calculated as $-\log_2(p)$, otherwise the min-entropy estimate is calculated as $\log_2(k)$.

Prediction Based Type Estimators. Estimators of prediction based type are proposed by Kelsey *et al.* [KMT15] and adopted by the second draft of SP 800-90B. These estimators attempt to predict the next sample in a sequence based on previous samples, and provide an estimate based on the probability of successfully predicting a source's outputs.

Each predictor (i.e., prediction based estimator) consists of a set of subpredictors. The predictor takes a competition strategy among its subpredictors, and it chooses the subpredictor with the highest rate of successful predictions to predict the subsequent output.

Below we introduce the specific mechanisms of the subpredictors [KMT15].

1. **Most Common in Window Subpredictor.** This kind of subpredictors includes a sliding window to record the most recently observed t samples. It predicts the subsequent output on the basis of the most common value in the window. The window size t takes value in {63, 255, 1023, 4095}.

2. **The Lag Subpredictor.** This kind of subpredictors is designed to detect the correlation and periodicity. The prediction value is the same as the t^{th} former value. The range of the parameter t is set from 1 to 128 in this predictor.

3. **The Markov Model Counting Subpredictor.** This kind of subpredictors records the occurrences of transitions from a fixed length pattern to a subsequent output, and makes a prediction, based on the most frequently observed transition from the current outputs.

4. **The LZ78Y Subpredictor.** This kind of subpredictors keeps a dictionary to record tuples that have appeared in previous outputs. The subpredictor is based on the LZ78 algorithm and the dictionary has a fixed size of 65536.

3 On the Underestimation of Statistic Based Estimators

In this section, we first discuss the negative impact of underestimation on the suite, and explain the reasons why the Collision Estimate and the Compression Estimate often provide underestimations. We present both theoretical analysis and experimental validation.

3.1 Impact of Underestimation

In non-IID track, estimators in the NIST estimation suite may give both underestimation and overestimation. As the SP 800-90B suite consists a serial of capable estimators and the minimum estimate is selected as the final result, overestimation seldom occurs in the final result, as long as one estimator provides a (nearly) correct estimate. Therefore, most overestimated values are not reflected in the final result. For example, the Most Common Value Estimate always gives the overestimates, but these results are discarded in the end. However, if an estimator provides a significant underestimate, the final result will be assigned to this underestimated value no matter how correct other estimators are. Therefore, we hold the opinion that, the underestimations of estimators are more harmful to the performance of the NIST estimation suite, thus should be avoided.

3.2 Analysis on Collision Estimate

We find that the Collision Estimate could underestimate the entropy source in both IID and non-IID tracks due to the following reasons.

1. For non-IID track, this estimator approximates the min-entropy on the basis of a statistic calculated from tested non-IID data, however, the formula used to calculate the statistic holds only under the hypothesis that the sources are IID.

2. For IID track, the Collision Estimate is highly sensitive to the statistic when the entropy sources are (almost) perfectly random.

Non-IID situation. This estimator calculates the min-entropy primarily according to Eq. (3). In order to illustrate the unsuitability of Eq. (3) for non-IID sources, we calculate a new equation of $\bar{X}$ and p when the inputs obey a first-order Markov process as an example, as shown in Theorem 1.

Theorem 1. *Let $\{s_n\}_{n\in\mathbb{N}}$ be a first-order Markov process of $\{0,1\}$, whose transfer matrix is $\begin{pmatrix} p & 1-p \\ 1-p & p \end{pmatrix}$, where $p > 0.5$. Then the theoretical min-entropy H equals to $-\log_2(3 - E(\bar{X}))$, where $E(\bar{X})$ is the expectation of collision statistic $\bar{X}$ of $\{s_n\}$.*

Proof. Let t be the number of samples to generate a collision. Since all the possible collision patterns are “00”,“11”,“010”,“011”,“100”, and “101”, the value of t can only be 2

or 3. Then, $E(\bar{X})$ is calculated as

$$\begin{aligned} E(\bar{X}) &= 2 \times \Pr\{t=2\} + 3 \times \Pr\{t=3\} \\ &= 2 \times p + 3 \times (1-p) \\ &= 3 - p. \end{aligned}$$

For the first-order Markov process, according to Eq. (2), the theoretical min-entropy per sample is

$$H = -\frac{1}{n}\log_2(0.5 \times p^{n-1}) \approx -\log_2(p) = -\log_2(3 - E(\bar{X})),$$

when n is large. □

Fig. 1 shows the difference of the min-entropy derived from Theorem 1 and the Collision Estimate. It is observed that the results from the Collision Estimate are always lower than the theoretical ones, and the errors cannot be ignored at most cases of $\bar{X}$. Since the SP 800-90B chooses the minimum estimate among all estimates as the final result, the suite significantly underestimates the min-entropy of this kind of non-IID entropy sources.

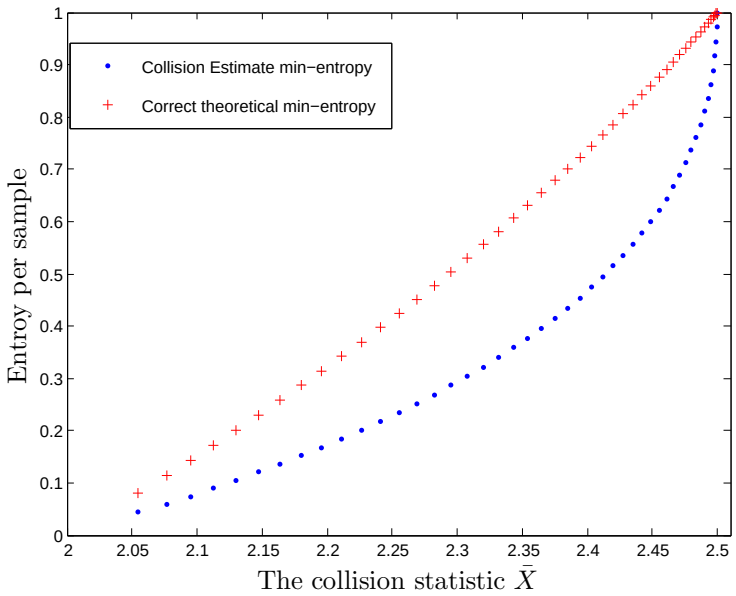

Figure 1: The comparison of min-entropy derived from Theorem 1 and the Collision Estimate

IID situation. Although the Collision Estimate is designed under the IID assumption (but used in non-IID track), we find that it still often underestimates the entropy sources that are perfectly random (i.e., the outputs have a uniform probability distribution). One reason is that the Collision Estimate is extremely sensitive to the statistic $\bar{X}$ when the min-entropy is close to its upper bound. In addition, it takes the 99% confidence interval's lower bound of $\bar{X}$ to calculate the min-entropy, which means a value slightly smaller than $\bar{X}$ is calculated to estimate the min-entropy. However, this causes a significant reduction of the min-entropy estimate due to the high sensitivity to $\bar{X}$. We present Theorem 2 to prove the above observations.

Theorem 2. *The min-entropy H derived by the Collision Estimate is a function of $\bar{X}$. 1) When H approaches $\log_2(k)$, $\frac{dH}{d\bar{X}}$ approaches infinity. 2) The Collision Estimate underestimates a perfect entropy source with probability 0.99.*

Proof. According to Eq. (3), $\bar{X}$ is a function of p. Hagerty and Draper [HD] proved that

$$\frac{d\bar{X}}{dp} < 0 \text{ for } p \in \left(\frac{1}{k}, 1\right], \text{ and } \frac{d\bar{X}}{dp} = 0 \text{ when } p = \frac{1}{k}, \tag{6}$$

where k and p are the same as the notations in Sect. 2.3. We can consider p as a function of $\bar{X}$ and $\lim\limits_{p\to\frac{1}{k}} \frac{dp}{d\bar{X}} = \infty$ according to Eq. (6). Then we can get,

$$\begin{aligned} \lim_{H\to\log_2(k)} \frac{dH}{d\bar{X}} &= \lim_{p\to\frac{1}{k}} \frac{dH}{dp} \times \frac{dp}{d\bar{X}} \\ &= \frac{k}{\ln(2)} \times \infty \\ &= \infty. \end{aligned}$$

That is, when H approaches $\log_2(k)$, $\frac{dH}{d\bar{X}}$ approaches infinity.

Then we will prove the second statement. According to Eq. (6), we can conclude that H is a monotonic increasing function of $\bar{X}$. As we mentioned in Sect. 2.3, the statistic $\bar{X}$ approximately obeys a normal distribution $\mathcal{N}(\mu, \hat{\sigma}^2)$, where μ and $\hat{\sigma}^2$ are the expectation and variance of $\bar{X}$, respectively. The theoretical entropy of a perfect entropy source is $H(\mu) = \log_2(k)$, and the derived min-entropy of the Collision Estimate is $H(\bar{X'})$, where $\bar{X'} = \bar{X} - 2.576\frac{\hat{\sigma}}{\sqrt{v}}$ and v is the number of collisions. Then we get the probability that the Collision Estimate underestimates a perfect entropy source as

$$\Pr\{H(\bar{X'}) - H(\mu) < 0\} = \Pr\{\bar{X'} < \mu\} = 0.99.$$

□

The two statements in Theorem 2 jointly cause the Collision Estimate to give an obvious underestimation of the min-entropy for (almost) perfectly random IID outputs. We note that, for other types of estimators, the second statement in Theorem 2 is also applicable, but the first statement does not hold. These estimators directly take the upper bound of the probability p instead of $\bar{X}$, yielding that $\lim\limits_{p\to\frac{1}{k}} \frac{dH}{dp} = -k/\ln(2)$ which is finite. Therefore, they do not have this problem.

3.3 Analysis on Compression Estimate

The Compression Estimate may underestimate the non-IID sources for the same reason with the Collision Estimate, as its statistic is also calculated under IID assumption. This estimator calculates the min-entropy primarily according to Eq. (5). We employ the same method to illustrate the unsuitability of Eq. (5) for non-IID sources in Theorem 3.

Theorem 3. *Let $\{s_n\}_{n\in\mathbb{N}}$ be a first-order Markov process of $\{0,1\}$, whose transfer matrix is $\begin{pmatrix} p & 1-p \\ 1-p & p \end{pmatrix}$, where $p > 0.5$. Then $E(\bar{X}) = \sum\limits_{i=2}^{\infty} \log_2(i) \times (1-p)^2 p^{i-2}$, where $E(\bar{X})$ is the expectation of compression statistic $\bar{X}$ of $\{s_n\}$.*

Proof. Following the definition of A_i in Eq. (4), we get the probability

$$\Pr\{A_i = i\} = \begin{cases} p, & i = 1, \\ (1-p)^2 p^{i-2}, & i > 1. \end{cases}$$

Then we have the expectation

$$\begin{aligned} E(\bar{X}) &= \sum_{i=1}^{\infty} \log_2(i) \times \Pr\{A_i = i\} \\ &= \sum_{i=2}^{\infty} \log_2(i) \times (1-p)^2 p^{i-2} + \log_2(1) \times p \\ &= \sum_{i=2}^{\infty} \log_2(i) \times (1-p)^2 p^{i-2}. \end{aligned} \tag{7}$$

□

According to Theorem 3, p can be calculated based on $E(\bar{X})$ by Eq. (7). Then, we can get the min-entropy $H = -\log_2(p)$, which has been proved in Theorem 1.

Fig. 2 depicts the difference of the min-entropy derived from Theorem 3 and the Compression Estimate. We can draw the similar conclusion with the Collision Estimate that the Compression Estimate distinctly underestimates the min-entropy for the entropy source following a first-order Markov process.

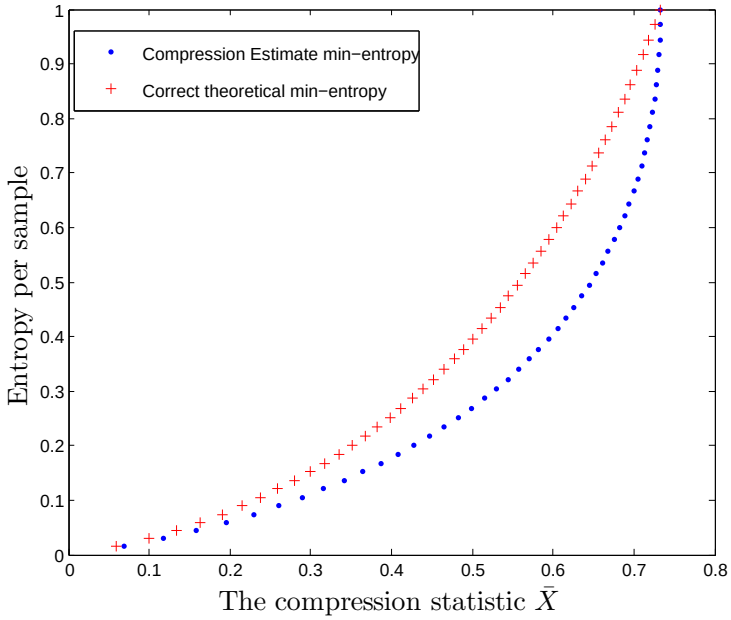

Figure 2: The comparison of min-entropy derived from Theorem 3 and the Compression Estimate

3.4 Experimental Validation

Non-IID source. Using MATLAB, we simulate an entropy source based on the first-order Markov process defined in Theorems 1 and 3 with different p's. The data size of each group is 10^6. Then we test the outputs with all estimators in the SP 800-90B. The experimental results are listed in Table 2.

Table 2: Entropy estimates for first-order Markov process with different p's

Estimator	$p = 0.8$	$p = 0.75$	$p = 0.7$	$p = 0.65$	$p = 0.6$	$p = 0.5$
1. MostCommon	0.9956	0.9958	0.9942	0.9948	0.9960	0.9959
2. LRS	0.5603	0.6836	0.7943	0.8763	0.9234	0.9886
3. MultiMCW	0.4843	0.6249	0.7007	0.8200	0.8758	0.9985
4. Markov	0.3256	0.4184	0.5152	0.6232	0.7365	0.9975
5. t-Tuple	0.3230	0.4134	0.5385	0.6134	0.7219	0.9469
6. Lag	0.3199	0.4132	0.5113	0.6198	0.7333	0.9966
7. MultiMMC	0.3199	0.4132	0.511	0.6198	0.7333	0.9962
8. LZ78Y	0.3199	0.4132	0.5113	0.6198	0.7333	0.9970
9. **Compression**	**0.2223**	**0.2844**	**0.3518**	**0.4304**	**0.5228**	1
10. **Collision**	**0.1704**	**0.2266**	**0.2895**	**0.3681**	**0.4618**	**0.9178**
Theoretical min-entropy	0.3219	0.4150	0.5141	0.6214	0.7369	1

From Table 2 we observe that, for this kind of data, the Collision Estimate always provides the minimum estimate among all estimators. The min-entropy from the Compression Estimate is larger than that from the Collision Estimate but also lower than other estimates, except for $p = 0.5$. The prediction based estimators and Markov estimator provide close estimates with the theoretical min-entropy. We present Fig. 3 to depict the results in Table 2 intuitively. It is observed that, the estimates of the No. 4 to No. 8 estimators are similar, while estimates of the No. 9 and No. 10 estimators (i.e. the Collision Estimate and the Compression Estimate) have significant reductions. This confirms the inaccuracy of the Collision Estimate and the Compression Estimate.

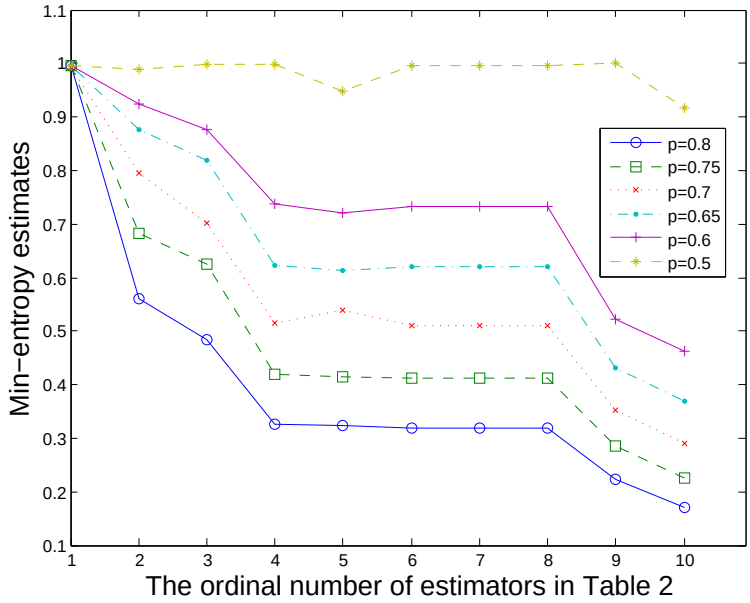

Figure 3: Min-entropy estimates of the NIST estimators for the first-order Markov progress

IID source. To confirm that the Collision Estimate also underestimates the perfectly random entropy sources, we choose several popular PRNGs including Blum-Blum-Shub generator (BBS) [BBS86], Linear Congruential Generator (LCG), Modular Exponentiation Generator (MODEXPG), and Micall-Schnorr Generator (MSG). We employ these PRNGs

Table 3: Entropy estimates for (almost) perfectly random data generated by different PRNGs

Estimator	BBS	LCG	MODEXPG	MSG
MostCommon	0.9877	0.9893	0.9882	0.9832
LRS	0.9984	0.9996	0.9951	0.9803
MultiMCW	0.9873	0.9918	0.9834	0.9910
Markov	0.9927	0.9941	0.9853	0.9856
t-Tuple	0.9227	0.9359	0.9319	0.9430
Lag	0.9881	0.9947	0.9913	0.9931
MultiMMC	0.9910	0.9926	0.9883	0.9912
LZ78Y	0.9945	0.9910	0.9931	0.9924
Compression	1	1	1	1
Collision	**0.8250**	**0.8527**	**0.8630**	**0.8742**

to generate (almost) perfectly random numbers with 10^5 bits for each data set. Table 3 shows that, the Collision Estimate's results are all under 0.9 and obviously lower than the results of other estimators, which is consistent with our theoretical conclusion.

4 Proposed Estimator Based on Conditional Probability

In this section, we propose a new estimator to calculate the min-entropy according to conditional probability. Then, we compare it with the Markov Estimate and prediction based estimators, and analyze the advantages of our estimator. Finally, we perform experiments to validate our analysis.

4.1 Proposed Formula of Min-entropy

As introduced in Sect. 2.1, Eq. (1) provided by the SP 800-90B [T+16] is only suitable for the IID entropy source, and Eq. (2) presented by Kelsey *et al.* [KMT15] targets the first-order Markov process. Therefore, we provide a new formula for high order Markov process, as this process is a very common model for non-IID entropy sources in real world.

We denote the outputs of an entropy source as $\{s_n\}_{n\in\mathbb{N}}$, where $s_n \in A = \{x_1, ..., x_k\}$. We focus on the most-likely probability of the value of the subsequent output. By assuming that any output s_q only depends on its previous d samples $\{s_{q-d}, s_{q-d+1}, ..., s_{q-1}\}$ denoted as $\mathbf{s}_d$, the probability distribution of s_q is represented as $\Pr\{s_q = x_i | \mathbf{s}_d = \xi\} = p_{\xi,i}$, where $\xi \in A^d$ and $i = 1, ..., k$. As the previous output sequence $\mathbf{s}_d$ has different possible values, we employ the weighted mean value of $\max\limits_{1\le i\le k} p_{\xi,i}$ to calculate the min-entropy

$$H_D = -\log_2 \big(\sum_{\xi\in A^d} \Pr\{\mathbf{s}_d = \xi\} \max_{1\le i\le k} p_{\xi,i} \big). \tag{8}$$

We note that Eq. (8) is a theoretical formula based on conditional probability. Next, we propose an estimator for H_D.

4.2 Proposed Estimator Based on Conditional Probability

We use similar idea of the Most Common Value Estimate to estimate the values of conditional probabilities. The following are the estimating steps of our estimator.

Given the inputs $S = (s_1, ..., s_L)$, where $s_i \in A = \{x_1, ..., x_k\}$,

1. Set the length w of $\mathbf{s}_w$ to 16 (this value can change according to the data size).
2. Record the pattern of $(s_i, s_{i+1}, ..., s_{i+w-1})$, from $i = 1$ to $i = L - w$. Count the number of different kinds of patterns, denoted as Z, and denote these patterns as $\xi_1, \xi_2, ..., \xi_Z$. Then, count the number of their occurrences, denoted as $B[1]$, $B[2]$,..., $B[Z]$.
3. If $\min\limits_{1\le i\le Z} B[i] < len = 166$, set $w = w - 1$, and go to Step 2.
4. For each pattern, record the value of the subsequent element and count the number of the most common value. Denote these numbers as $M[1], M[2], ..., M[Z]$.
5. Approximate $\Pr\{\mathbf{s}_w = \xi_i\}$ by $\frac{B[i]}{L-w}$ and approximate $\max\limits_{1\le j\le k} p_{\xi_i,j}$ by $\frac{M[i]}{B[i]}$.
6. Compute the probability p of the most-likely subsequent output,
$$p = \sum_{i=1}^{Z} \left(\frac{M[i]}{B[i]} \times \frac{B[i]}{L-w}\right) = \sum_{i=1}^{Z} M[i]/(L-w).$$
7. Calculate the upper bound of the 99% confidence interval of p, $p_{0.99} = \min\left(1, p + 2.576\sqrt{\frac{p(1-p)}{L-w}}\right)$.
8. The min-entropy H_D is estimated as $-\log_2(p_{0.99})$.

Example. For illustrative purposes, we suppose that $w = 2$, $k = 2$, $L = 17$ and $S = (1, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1)$. Then, the number of patterns is 4, and we calculate that $\mathbf{s}_1 = (1, 0)$, $B[1] = 3$, $M[1] = 3$, $\mathbf{s}_2 = (0, 0)$, $B[2] = 6$, $M[2] = 3$, $\mathbf{s}_3 = (0, 1)$, $B[3] = 3$, $M[3] = 2$, $\mathbf{s}_4 = (1, 1)$, $B[4] = 3$, and $M[4] = 2$. Then we get $p = \frac{3+3+2+2}{17-2} = \frac{2}{3}$, $p_{0.99} = 0.98$, and $H_D = -\log_2(p_{0.99}) = 0.02$. Note that, H_D is very small in this example, as the sample length is too short.

Parameter setting. We set the initial value of w as 16, because the highest order of the Markov model considered in the SP 800-90B is 16. We set the value of len as 166 to ensure the statistical errors of $\max\limits_{1\le j\le k} p_{\xi_i,j}$ are no larger than $2.576 * \sqrt{\frac{p*(1-p)}{166}}$ (≤ 0.1) in the 99% confidence level. These statistical errors are due to using $\frac{M[i]}{B[i]}$ to approximate $\max\limits_{1\le j\le k} p_{\xi_i,j}$. As these statistical errors offset each other when calculating the weighted average value in Step 6, the final statistical error of p is much less than 0.1 in practice.

Computational overhead. The overhead of our proposed estimator is comparable to that of one subpredictor of the MultiMMC Prediction Estimate, thus is not expensive.

Recommended size for tested data. If the subsequent output of an entropy source mainly depends on its previous d samples ($d < w$), the data length $len \times k^d$ will be enough for our estimator to provide an accurate estimate. If data size is a concern, users can decrease the value of len to improve the applicability.

4.3 Advantages over Markov Estimate

The Markov Estimate assumes that the tested data obey a first-order Markov model. It calculates the initial probability of each possible value x_i in $A = \{x_1, ..., x_k\}$ and the $k \times k$ transition matrix. Then, it computes the maximal possibility $P_{\max}$ of a sequence with length 128 according to the calculated initial probabilities and transition matrix. The min-entropy estimate is $-\frac{1}{128}\log_2(P_{\max})$.

In fact, the Markov Estimate is basically equivalent to our estimator with $w = 1$. The main difference is that, the Markov Estimate focuses on the maximal probability

of a 128-length sequence, while our estimator focuses on the maximal probability of one subsequent sample given previous samples.

Compared with the Markov Estimate, our estimator has an obvious advantage in the estimation for d^{th}-order ($d > 1$) Markov model, while the Markov Estimate is only suitable for first-order Markov model due to the heavy computation complexity. Another advantage is that, our estimator has a lower statistical error thanks to the offset operation in Step 6.

4.4 Comparison with Prediction Based Estimators

The prediction based estimators also focus on the subsequent sample as our proposed estimator. A predictor consists of a set of subpredictors, and each subpredictor predicts the next output based on previous samples by a specific algorithm. A predictor takes a competition strategy among its subpredictors. For each prediction, it predicts the next output employing the subpredictor with the highest rate of successful predictions. Then, the predictor uses the correctly predicting probability to calculate the min-entropy.

We point out that the correctly predicting probability of a predictor is also a kind of conditional probability (as the basis of our proposed estimator) in Theorem 4. To prove that, we first show the relationship between the correctly predicting probabilities, of a predictor and its subpredictors, in Lemma 1.

Lemma 1. *We denote the correctly predicting probability of the predictor as P_0, and denote that of each subpredictor as P_i, where $i = 1, 2, ..., l$, and l is the number of subpredictors. When the length L of inputs is large enough, we have $\lim\limits_{L\to\infty} P_0 = \max\limits_{i\in(1,...,l)} P_i$.*

Proof. We denote the j^{th} prediction value of the predictor as $Pre_{0,j}$ and that of each subpredictor as $Pre_{i,j}$. Without loss of generality, we assume that $P_l > P_{l-1} > ... > P_1$. According to the Bernoulli law of large numbers, we have

$$\forall \varepsilon, \exists M, \quad s.t. \ \text{ when } j > M, \ \Pr\{Pre_{0,j} = Pre_{l,j}\} > 1 - \varepsilon.$$

That is to say, after M samples, the l^{th} subpredictor will be chosen almost all the time. Assuming that L is large enough, if the first M predictions are all successful, we get

$$\lim_{L\to\infty} P_0 \leq \lim_{L\to\infty} \frac{M + (L - M) \times P_l}{L} = P_l;$$

if the first M predictions all fail, we get

$$\lim_{L\to\infty} P_0 \geq \lim_{L\to\infty} \frac{(L - M) \times P_l - M}{L}) = P_l.$$

So we have $\lim\limits_{L\to\infty} P_0 = \max\limits_{i\in(1,...,l)} P_i$. □

Theorem 4. *When the entropy source follows a stationary process and the tested sequence is long enough, the correctly predicting probability of a predictor is no larger than the maximal conditional probability.*

Proof. We denote the tested samples as $S = (s_1, ..., s_L)$, where $s_i \in A = \{x_1, ..., x_k\}$, and the j^{th} prediction value of the subpredictor as Pre_j. The subpredictor predicts the subsequent output s_q only depending on a fixed length t of previous samples $\mathbf{s}_t = (s_{q-t}, s_{q-t+1}, ..., s_{q-1})$, where $q = t + 1, t + 2, ..., L$. Then we can easily get

$$\Pr\{s_q = Pre_q | \mathbf{s}_t = \xi\} \leq \max_{1\leq i\leq k} \Pr\{s_q = x_i | \mathbf{s}_t = \xi\}.$$

Since the entropy source is stationary, $\Pr\{s_q = x_i | \mathbf{s}_t = \xi\}$ does not change with time. According to the total probability formula, we have

$$\Pr\{s_q = Pre_q\} \leq \sum_{\xi \in \mathbf{A}^t} \Pr\{\mathbf{s}_t = \xi\} \times \max_{1 \leq i \leq k} \Pr\{s_q = x_i | \mathbf{s}_t = \xi\}.$$

According to Lemma 1, we can easily get that, when the tested sequence is long enough, the correctly predicting probability of a predictor is no larger than the maximal conditional probability. □

It is reasonable to assume that the entropy source follows a stationary process. If the inherent entropy of the entropy source varies with time, the static entropy estimation is meaningless. When the conditions in Theorem 4 are satisfied, a well-designed prediction based estimator will provide an estimate equaling to H_D which is the target of our estimator. Assume that the subsequent output of an entropy source mainly depends on previous d samples. When d is not large (e.g., d is less than the chosen w in our estimator), the min-entropy estimate provided by our estimator is no worse than that of any predictor. However, when d is large, our estimator may have non-negligible statistical errors due to the lack of data, while the prediction based estimators may have lower requirement for the data amount.

4.5 Experimental Results

To evaluate the effectiveness of our proposed estimator, we perform experiments on non-IID entropy sources to compare our estimator with the NIST estimators. We simulate non-IID data of the following two common types.

– *Markov model.* In this model, the samples are generated from the first-order and second-order Markov processes. The correct theoretical min-entropy of this model can be calculated by Eq. (8). We generate 10 sets of data with different min-entropy in this model.

– *Oscillator-based model.* In this model, the samples are simulated by sampling an oscillating signal with phase noise. The correct theoretical min-entropy of this model can be approximated by the lower bound of entropy in [BLMT11]. We generate 6 sets of data with different min-entropy in this model.

For each data set with the length 10^6, we first calculate the correct theoretical min-entropy, and then perform the estimation by using our estimator and the NIST estimation suite. The results are shown in Fig. 4 and 5, where we only present our estimate and three representative estimates: the highest, lowest, and best estimates in the NIST estimation suite.

Fig. 4 shows the estimation results of simulated data from Markov models. We observe that estimates provided by our estimator are close to the correct estimates. The highest estimates are significantly larger than the correct estimates, which are mostly provided by the Most Common Value Estimate and the Collision Estimate. The lowest estimates which are close to the correct estimates are provided by prediction based estimators, while the other lowest estimates, which are much lower than the correct estimates, are provided by the Collision Estimate. The best estimates in the NIST suite are all provided by prediction based estimators.

Fig. 5 shows the estimation results of simulated data from the oscillator-based model. This model only provides binary data, so the entropy per sample is no more than 1. Our estimates and the best estimates from NIST estimation suite are very close to the correct estimates. But the highest estimates and lowest estimates are both far away from the correct estimates. The Most Common Value Estimate always provides the highest estimates and the Collision Estimate always provides the lowest estimates, which are consistent with our analysis in Sect. 3.

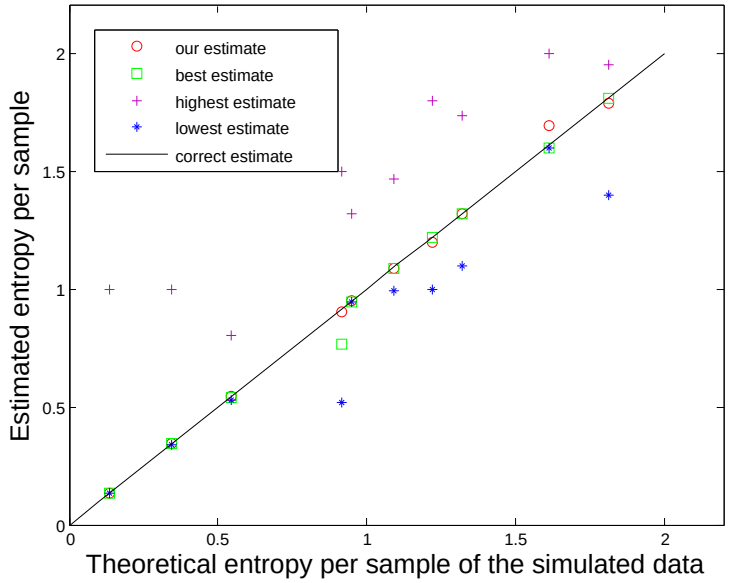

Figure 4: The comparison between our estimator and NIST estimation suite using data from Markov model

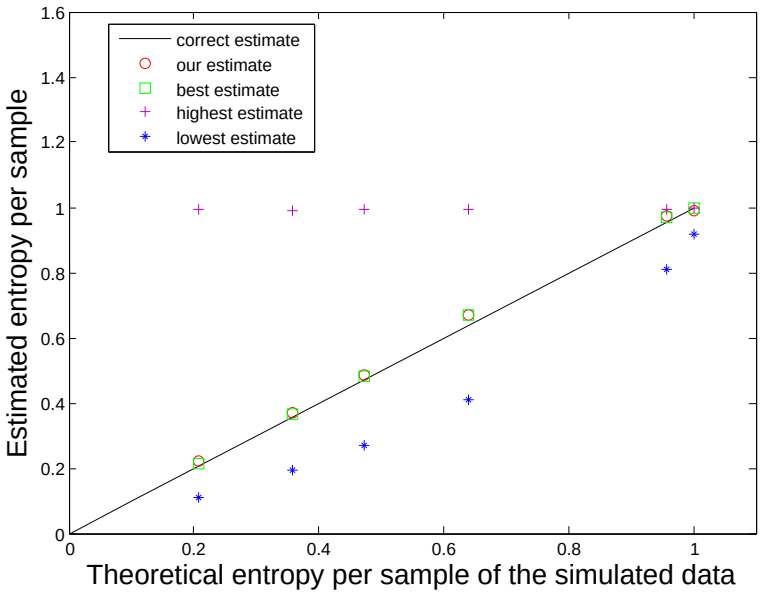

Figure 5: The comparison between our estimator and NIST estimation suite using data from oscillator-based model

In the experimental results, the best estimates in the NIST suite are always generated by the prediction based estimators for either type of entropy source. Meanwhile, our estimator has comparable performance with these estimators, which is consistent with the conclusion of Theorem 4.

5 Conclusion

Accurate entropy estimation is critical for the evaluation of RNG security. We prove that the Collision Estimate and the Compression Estimate could provide significant underestimates for non-IID data. Furthermore, we propose a novel estimator to calculate the min-entropy of non-IID sources, from the perspective of the maximal probability of the subsequent output. Compared with the Markov Estimate, our estimator has a wider scope of application. We also prove that our proposed estimator is able to show comparable performance with the excellent prediction based estimators under common conditions. Experiments on simulated non-IID sources show that our estimator provides close estimates to the real min-entropy.

Acknowledgments

The authors would like to thank the anonymous reviewers for their invaluable suggestions and comments to improve the quality and fairness of this paper. This work was partially supported by National Basic Research Program of China (973 Program No. 2013CB338001), Cryptography Development Foundation of China (No. MMJJ20170205), and National Natural Science Foundation of China (No. 61602476 and No. 61772518).

References

[BBS86] Lenore Blum, Manuel Blum, and Mike Shub. A simple unpredictable pseudo-random number generator. *SIAM J. Comput.*, 15(2):364–383, 1986.

[BLMT11] Mathieu Baudet, David Lubicz, Julien Micolod, and André Tassiaux. On the security of oscillator-based random number generators. *J. Cryptology*, 24(2):398–425, 2011.

[DGP09] Leo Dorrendorf, Zvi Gutterman, and Benny Pinkas. Cryptanalysis of the random number generator of the windows operating system. *ACM Trans. Inf. Syst. Secur.*, 13(1):10:1–10:32, 2009.

[GPR06] Zvi Gutterman, Benny Pinkas, and Tzachy Reinman. Analysis of the linux random number generator. In *2006 IEEE Symposium on Security and Privacy, 21-24 May 2006, Berkeley, California, USA*, pages 371–385, 2006.

[Gra03] Peter Grassberger. Entropy estimates from insufficient samplings. arXiv:physics/0307138, 2003.

[HD] Patrick Hagerty and Tom Draper. Entropy bounds and statistical tests. http://csrc.nist.gov/groups/ST/rbg_workshop_2012/hagerty_entropy_paper.pdf.

[KASW98] Ioannis Kontoyiannis, Paul H. Algoet, Yuri M. Suhov, and Abraham J. Wyner. Nonparametric entropy estimation for stationary processesand random fields, with applications to english text. *IEEE Trans. Information Theory*, 44(3):1319–1327, 1998.

[KMT15] John Kelsey, Kerry A. McKay, and Meltem Sönmez Turan. Predictive models for min-entropy estimation. In *Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings*, pages 373–392, 2015.

[KS] Wolfgang Killmann and Werner Schindler. A proposal for: Functionality classes for random number generators. AIS 20/31. http://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_20_Functionality_classes_for_random_number_generators_e.pdf;jsessionid=A837238C46E2F9205B0C2AF043153011.2_cid294?__blob=publicationFile.

[Mar] George Marsaglia. Diehard battery of tests of randomness. http://www.stat.fsu.edu/pub/diehard/.

[Mau92] Ueli M. Maurer. A universal statistical test for random bit generators. *J. Cryptology*, 5(2):89–105, 1992.

[NSB01] Ilya Nemenman, F. Shafee, and William Bialek. Entropy and inference, revisited. In *Advances in Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada*, pages 471–478, 2001.

[Pan03] Liam Paninski. Estimation of entropy and mutual information. *Neural Computation*, 15:1191–1253, 2003.

[R+] Andrew Rukhin et al. A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST Special Publication 800–22. http://csrc.nist.gov/publications/nistpubs/800-22-rev1a/SP800-22rev1a.pdf.

[Rou99] Mark S. Roulston. Estimating the errors on measured entropy and mutual information. *Physica D: Nonlinear Phenomena*, 125:285–294, 1999.

[T+16] Meltem Sünmez Turan et al. Recommendation for the entropy sources used for random bit generation. (Second DRAFT) NIST Special Publication 800–90B. http://csrc.nist.gov/publications/drafts/800-90/sp800-90b_second_draft.pdf, January 2016.

[VP16] Mathy Vanhoef and Frank Piessens. Predicting, decrypting, and abusing WPA2/802.11 group keys. In *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, pages 673–688, 2016.

[WW94] David H. Wolpert and David R. Wolf. Estimating functions of probability distributions from a finite set of samples, part 1: Bayes estimators and the shannon entropy. arXiv preprint comp-gas/9403001, 1994.

[WZ89] Aaron D. Wyner and Jacob Ziv. Some asymptotic properties of the entropy of a stationary ergodic data source with applications to data compression. *IEEE Trans. Information Theory*, 35(6):1250–1258, 1989.