On Data Complexity of Distinguishing Attacks vs. Message Recovery Attacks on Stream Ciphers

Goutam Paul*, Souvik Ray†
Indian Statistical Institute, Kolkata 700 108, India.
Email: *goutam.paul@isical.ac.in, †souvikr974@gmail.com

Abstract

We revisit the different approaches used in the literature to estimate the data complexity of distinguishing attacks on stream ciphers and analyze their inter-relationships. In the process, we formally argue which approach is applicable (or not applicable) in what scenario. To our knowledge, this is the first kind of such an exposition. We also perform a rigorous statistical analysis of the message recovery attack that exploits a distinguisher and show that in practice there is a significant gap between the data complexities of a message recovery attack and the underlying distinguishing attack. This gap is not necessarily determined by a constant factor as a function of the false positive and negative rate, as one would expect. Rather this gap is also a function of the number of samples of the distinguishing attack. We perform a case study on RC4 stream cipher to demonstrate that the typical complexities for message recovery attack inferred in the literature are but under-estimates and the actual estimates are quite larger.

Index Terms

Data Complexity, Distinguisher, Distinguishing Attack, Message Recovery, Stream Cipher.

I. Introduction

A stream cipher generates a long pseudo-random keystream from a short secret key to encrypt a message by bitwise XOR operation with the keystream. Since the sender and the receiver share the same secret key and the keystream generation algorithm is deterministic, the identical keystream is generated at the receiver side, which when bitwise XOR-ed with the ciphertext recovers the message.

For a stream cipher, if there is an event such that the probability of occurrence of the event is different from the same event in case of a uniformly random sequence of bits, the event is said to be *biased*. If there exists a biased event based only on the bits of the keystream sequence, then such an event gives rise to a *distinguisher* for the cipher. A distinguisher can computationally differentiate between the keystream output of the stream cipher and a truly random sequence of bits.

The efficiency of a distinguisher is measured by the data complexity, i.e., the number of samples required to identify the bias. For an attacker to successfully identify and exploit a bias, one requires to inspect a certain length of the output sequence so that one can collect sufficient number of samples for the event under consideration. The less the number of samples required, the more is the efficiency of the distinguisher. Sometimes the data complexity is expressed in total number of keystream bits required to perform the distinguishing attack.

Very often, a distinguisher is directly used in mounting a message recovery attack on stream ciphers. A famous example is the attack [1] on broadcast RC4. Let Z_r be the r-th keystream byte of RC4 and $N = 256$ be the standard sate array size of RC4. It was proved in [1] that $\Pr(Z_2 = 0) \approx \frac{2}{N}$ for RC4, whereas the same event in an uniformly random bitstream would occur with probability $\frac{1}{N}$. In the broadcast scenario, the same plaintext is encrypted using multiple secret keys, and then the ciphertexts are broadcast to a group of recipients, possessing the corresponding secret keys. For every encryption key, the second message byte M_2 has the probability $\frac{2}{N}$ to be XOR-ed with 0, and the probability $\frac{1}{N}$ to be XOR-ed with each of the other possible bytes. Thus, a fraction of $\frac{2}{N}$ of the second ciphertext bytes C_2 are expected to have the same value as M_2, and thus the most frequent value of C_2 across all the samples is the mostly likely value of M_2. The above approach has been adopted by [2] in mounting message recovery attack on every individual message bytes 3 to 255 based on the distinguishers for RC4 keystream Z_r, $3 \leq r \leq 255$. Later, the work [3] considered the collection of all the biases in all the keystream byte together to perform joint message recovery.

In all the above works, the number of samples required to mount the message recovery attack is considered to be of the same order as that of the underlying distinguisher. However, we observe that in practice it is not always so. For example, for the broadcast attack on RC4 second byte, the complexity of message recovery attack for a success probability of 70% is around 8 times higher than that of the distinguishing attack for the same success probability. In this paper, we perform a rigorous analysis to understand this gap between the data complexities of distinguishing attack and message recovery attack.

A. Our Contributions

We observe that there exist different approaches to estimate the data complexity of a distinguisher, yielding different expressions, albeit sometimes one may be a crude approximation of the other. Moreover, the data complexity of a message recovery attack based on the distinguisher is usually taken to be the same (or of the same order) as the distinguisher itself, though in practice it is not necessarily true in all scenarios.

Our current work has two-fold contributions.

1) We review the different approaches used in the literature to estimate the data complexity and point out their connections and applicable scenarios. To our knowledge, such an expository coverage of the different approaches has not been done so far. Wherever possible, we provide short proofs of the results to make this exposition self-sufficient. For longer proofs, we cite appropriate references.
2) We perform a rigorous statistical analysis of the message recovery attack and show that in practice there is a significant gap between the data complexities of a message recovery attack and the underlying distinguishing attack. This gap is not necessarily determined by a constant factor as a function of the false positive and negative rate, as one would expect. Rather, this gap is also a function of the number of samples of the distinguishing attack. We perform a case study on RC4 stream cipher to demonstrate that the typical message complexities inferred in the literature are but under-estimates and the actual estimates are quite larger.

B. Notations

Before going into technical discussion, we list down some notations frequently used in this article below.

$\mathcal{M}$:	The message space (the set of all possible bytes)
$\mathcal{P}$:	The distribution of the keystream bytes over $\mathcal{M}$
$\mathcal{P} \oplus m$:	Distribution of the random variable $X \oplus m$, where $X \sim \mathcal{P}$ and $m \in \mathcal{M}$
p_z :	Probability of the byte z in the distribution $\mathcal{P}$
$p_{(t)}$:	Distribution of the t-th order statistic for the keystream bytes
$\mathcal{P}^{(k)}$:	The distribution of vector of k keystream bytes over $\mathcal{M}^k$
$p_{\boldsymbol{z}}^{(k)}$:	Probability of the k-byte vector $\boldsymbol{z}$ in the distribution $\mathcal{P}^{(k)}$
$\mathcal{Q}$:	The prior distribution of the plaintext bytes over $\mathcal{M}$
q_z :	Probability of the byte z in the distribution $\mathcal{Q}$
$\mathcal{Q}^{(k)}$:	The distribution of vector of k plaintext bytes over $\mathcal{M}^k$
$q_{\boldsymbol{z}}^{(k)}$:	Probability of the k-byte vector $\boldsymbol{z}$ in the distribution $\mathcal{P}^{(k)}$
D_n :	n-dimensional discrete distribution over some countable set (same is P_n, Q_n)
$E_P[X]$:	Expectation of the random variable $X \sim P$ where P is a distribution.
$\mathcal{R}^c$:	Complement of a set $\mathcal{R}$
$\mathcal{B}er$:	Bernoulli distribution
$\mathcal{B}$:	Binomial distribution
$\mathcal{N}$:	Normal distribution
$A\mathcal{N}$:	Asymptotic Normal distribution

II. Revisiting Data Complexity of Distinguishing Attacks

In this section, we revisit the existing techniques for estimating the data complexity of a distinguisher and point out their relations and subtleties.

A. Distance between Expectations

This approach has been used in [1]. We revisit their main result below.

Theorem 1. *Suppose the event e happens in distribution X with probability p and in distribution Y with probability $p(1+q)$. Then for small p and q, $O(\frac{1}{pq^2})$ samples suffice to distinguish X from Y with a constant probability of success.*

Proof: We follow the proof approach of [1]. Let X_e, Y_e be the random variables specifying the number of occurrences of e in n samples. Then X_e and Y_e have binomial distributions with parameters (n, p) and $(n, p(1+q))$ respectively. Their expectations, variances and standard deviations are (assuming both $p, q \ll 1$) as follows.

$$E[X_e] = np, E[Y_e] = np(1+q),$$

$$V(X_e) = np(1-p) \approx np,$$

$$V(Y_e) = np(1+q)(1-p(1+q)) \approx np(1+q),$$

$$\sigma(X_e) = \sqrt{V(X_e)} \approx \sqrt{np},$$

$$\sigma(Y_e) = \sqrt{V(Y_e)} \approx \sqrt{np(1+q)} \approx \sqrt{np}.$$

The authors of [1] consider the size of n that implies a difference of at least one standard deviation between the expectations of the two distributions:

$$\begin{aligned} E[Y_e] - E[X_e] \geq \sigma X_e \quad &\iff \quad np(1+q) - np \geq \sqrt{np} \\ &\iff \quad npq \geq \sqrt{np} \\ &\iff \quad n \geq \frac{1}{pq^2}. \qquad (1) \end{aligned}$$

Consequently, $O(pq^2)$ samples (the constant depends on the desired success probability) suffice for the distinguishing. ■

Note that when at least one of $p \ll 1$ and $q \ll 1$ does not hold, the above approach does not work.

B. Simple Hypothesis Testing

A more rigorous analysis appeared in [4] that gets rid of the restriction $p \ll 1$ and $q \ll 1$. We revisit this technique here.

Theorem 2. *Suppose the event e happens in uniform random bitstream with probability p and in keystream of a stream cipher with probability $p(1+q)$. Then the data complexity of the distinguisher with false positive and false negative rates α and β is given by*

$$n > \frac{\left(\kappa_1\sqrt{1-p} + \kappa_2\sqrt{(1+q)\,(1-p(1+q))}\right)^2}{pq^2},$$

where $\Phi(-\kappa_1) = \alpha$ and $\Phi(\kappa_2) = 1-\beta$.

Proof: Consider an event e with $\Pr(e) = p^*$, while observing samples of keystream words of a stream cipher. Let $X_r = 1$, if the event e occurs in the r-th sample; $X_r = 0$, otherwise. In other words, $\Pr(X_r = 1) = p^*$ for all r. Thus,

$$X_r \sim \mathcal{B}er(p^*).$$

If we observe n many samples, then

$$\sum_{r=1}^{n} X_r \sim \mathcal{B}(n, p^*).$$

When X_r's are independent and identically distributed (i.i.d.) random variables and n is large enough,

$$\sum_{r=1}^{n} X_r \sim \mathcal{N}\left(np^*, np^*(1-P^*)\right).$$

We are interested in testing the null hypothesis

$$H_0 : p^* = p(1+q), \quad q > 0,$$

against the alternative hypothesis

$$H_1 : p^* = p.$$

The objective is to find a threshold c in $[np, np(1+q)]$ such that

$$P\left(\sum_{r=1}^{n} X_r \le c \mid H_0\right) \le \alpha$$

and

$$P\left(\sum_{r=1}^{n} X_r > c \mid H_1\right) \le \beta.$$

For such a c to exist, we need

$$np(1+q) - np > \kappa_1\sigma_1 + \kappa_2\sigma_2,$$

where

$$\begin{aligned}
\sigma_1^2 &= np(1+q)\,(1-p(1+q))\,, \\
\sigma_2^2 &= np(1-p), \\
\Phi(-\kappa_1) &= \alpha \\
\text{and} \quad \Phi(\kappa_2) &= 1-\beta.
\end{aligned}$$

This gives,

$$n > \frac{\left(\kappa_1\sqrt{1-p} + \kappa_2\sqrt{(1+q)\,(1-p(1+q))}\right)^2}{pq^2}. \tag{2}$$

■

In the special case, when both $p, q \ll 1$, the numerator of Equation (2) is approximately equal to $(\kappa_1 + \kappa_2)^2$, and one needs at least $\frac{(\kappa_1+\kappa_2)^2}{pq^2}$ many samples to perform the test.

Table II-B gives the sample complexity, false positive and negative rates and the success probability for some selected values of k_1 and k_2.

TABLE I
SAMPLE COMPLEXITY AND SUCCESS PROBABILITY FOR DISTINGUISHERS.

$k_1 = k_2$	Number of samples	$\alpha = \beta$	Success probability
0.5	$1/pq^2$	0.3085	69.15%
1	$4/pq^2$	0.1587	84.13%
2	$16/pq^2$	0.0228	97.72%

Since $0.6915 > 0.5$ is a reasonably good success probability, $O(\frac{1}{pq^2})$ many samples are enough to mount a distinguisher and this threshold is indeed used as a benchmark to compare the data complexities of different distinguishing attacks in practice.

C. Relative Entropy Between Distributions

This analysis appeared in [5, Appendix A]. The relative entropy between two discrete probability distributions $P(\cdot)$ and $Q(\cdot)$ is given by the Kullback-Leibler divergence [6]

$$D_{KL}(P||Q) := \sum_x P(x)\log_2 \frac{P(x)}{Q(x)}, \tag{3}$$

where x runs over all the sample points. Note that this can also be written as

$$D_{KL}(P||Q) = E_P\left[\log_2 \frac{P(X)}{Q(X)}\right],$$

where $X \sim P$. We have the following straight-forward result.

Proposition 1. *For the above-mentioned single event e with probabilities p and $p(1+q)$ in two different distributions $P(\cdot)$ and $Q(\cdot)$, the relative entropy is approximately equal to pq^2, for small p, q.*

Proof: We have

$$\begin{aligned}
D_{KL}(P||Q) &= p\log_2\left[\frac{p}{p(1+q)}\right] \\
&+ (1-p)\log_2\left[\frac{1-p}{1-p(1+q)}\right] \\
&= p\log_2\left[1-\frac{q}{1+q}\right] \\
&+ (1-p)\log_2\left[1+\frac{pq}{1-p(1+q)}\right] \\
&\approx -p\left(\frac{q}{1+q}\right) + (1-p)\left(\frac{pq}{1-p(1+q)}\right) \\
&\approx pq^2.
\end{aligned}$$

Similarly, for small p, q, we also have,

$$\begin{aligned}
D_{KL}(Q||P) &= p(1+q)\log_2\left[\frac{p(1+q)}{p}\right] \\
&+ (1-p(1+q))\log_2\left[\frac{1-p(1+q)}{1-p}\right] \\
&= p\log_2 q - (1-p(1+q))\log_2\left[1+\frac{pq}{1-p(1+q)}\right] \\
&\approx p(1+q)q - (1-p(1+q))\left(\frac{pq}{1-p(1+q)}\right) \\
&\approx pq^2.
\end{aligned}$$

■

Also, the following small technical result directly follows from the definition in Equation (3).

Proposition 2. *If P, Q are two distributions defined over the domain A and P', Q' are two other distributions defined over the domain B, then it can be shown that the overall relative entropy of the joint distributions (considering independence of the corresponding random variables over the two domains) PP' and QQ' is given by $D_{KL}(PP'||QQ') = D_{KL}(P||Q) + D_{KL}(P'||Q')$.*

Now we can state the following result.

Lemma 1. *For n independent occurrences of the event e with probabilities p and $p(1+q)$ in two different distributions $P(\cdot)$ and $Q(\cdot)$, the relative entropy is approximately equal to npq^2, for small p, q.*

Proof: Applying Proposition 2 to n samples from the same distribution as in Proposition 1, we get the result. ■

Now, according to [7], [8], we have the following result connecting the relative entropy to the false positive and negative rates.

Lemma 2. *Suppose D_n is an unknown discrete distribution and P_n and Q_n are two known distributions. Suppose we have a test (may be randomized) for*

$$H_0 : D_n = P_n \quad vs \quad H_1 : D_n = Q_n\,,$$

based on $\boldsymbol{X} := (X_1, X_2, \ldots, X_n)$, a sample from the distribution D_n, with false positive rate (α) and false negative rate (β). Then we have the following bound

$$D_{KL}(Q_n||P_n) \geq \beta\log_2\frac{\beta}{1-\alpha} + (1-\beta)\log_2\frac{1-\beta}{\alpha}. \tag{4}$$

Proof: Suppose where $\mathcal{S}$ is the sample space and $\phi : \mathcal{S} \longrightarrow [0,1]$ be the *test function* for the concerned test with false positive rate (α) and false negative rate (β), i.e., we reject H_0 with probability $\phi(\boldsymbol{x})$, when $\boldsymbol{X} = \boldsymbol{x}$ is observed. Then we have by definition

$$E_{H_0}[\phi(\boldsymbol{X})] = \sum_{\boldsymbol{x}\in\mathcal{S}} \phi(\boldsymbol{x})P_n(\boldsymbol{x}) = \alpha,$$

$$E_{H_1}[(1-\phi)(\boldsymbol{X})] = \sum_{\boldsymbol{x}\in\mathcal{S}} (1-\phi(\boldsymbol{x}))Q_n(\boldsymbol{x}) = \beta.$$

Note that,

$$\begin{aligned} D_{KL}(Q_n||P_n) &= \sum_{\boldsymbol{x}\in\mathcal{S}} Q_n(\boldsymbol{x})\log_2\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})} \\ &= \sum_{\boldsymbol{x}\in\mathcal{S}} P_n(\boldsymbol{x})\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})}\log_2\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})} \\ &= \sum_{\boldsymbol{x}\in\mathcal{S}} P_n(\boldsymbol{x}) f\left(\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})}\right) \end{aligned}$$

where $f:\mathbb{R}^+\to\mathbb{R}$ defined as $f(z)=z\log_2(z)$, $\forall\, z>0$. Then, $f^{(2)}(z)=(z\ln 2)^{-1}>0$, $\forall\, z>0$; which implies f is *convex* and *continuous* also. Hence, using *Jensen's Inequality*, we have

$$\begin{aligned} &\sum_{\boldsymbol{x}\in\mathcal{S}} \frac{\phi(\boldsymbol{x})P_n(\boldsymbol{x})}{\sum_{\boldsymbol{x}\in\mathcal{S}}\phi(\boldsymbol{x})P_n(\boldsymbol{x})} f\left(\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})}\right) \\ \geq\ & f\left(\sum_{\boldsymbol{x}\in\mathcal{S}} \frac{\phi(\boldsymbol{x})P_n(\boldsymbol{x})}{\sum_{\boldsymbol{x}\in\mathcal{S}}\phi(\boldsymbol{x})P_n(\boldsymbol{x})}\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})}\right) \\ =\ & f\left(\frac{\sum_{\boldsymbol{x}\in\mathcal{S}}\phi(\boldsymbol{x})Q_n(\boldsymbol{x})}{\sum_{\boldsymbol{x}\in\mathcal{S}}\phi(\boldsymbol{x})P_n(\boldsymbol{x})}\right) \\ =\ & f\left(\frac{1-\beta}{\alpha}\right). \end{aligned}$$

Hence,

$$\begin{aligned} \sum_{\boldsymbol{x}\in\mathcal{S}} \phi(\boldsymbol{x})P_n(\boldsymbol{x}) f\left(\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})}\right) &\geq \sum_{\boldsymbol{x}\in\mathcal{S}} \phi(\boldsymbol{x})P_n(\boldsymbol{x}) f\left(\frac{1-\beta}{\alpha}\right) \\ &= \alpha f\left(\frac{1-\beta}{\alpha}\right) \\ &= (1-\beta)\log_2\frac{1-\beta}{\alpha}. \end{aligned}$$

Replacing ϕ by $1-\phi$ and taking similar sums we get,

$$\sum_{\boldsymbol{x}\in\mathcal{S}}(1-\phi(\boldsymbol{x}))P_n(\boldsymbol{x}) f\left(\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})}\right) \geq \beta\log_2\frac{\beta}{1-\alpha}.$$

Summing the above two inequalities we get

$$\begin{aligned} &\sum_{\boldsymbol{x}\in\mathcal{S}} P_n(\boldsymbol{x}) f\left(\frac{Q_n(\boldsymbol{x})}{P_n(\boldsymbol{x})}\right) \\ \geq\ & \beta\log_2\frac{\beta}{1-\alpha}+(1-\beta)\log_2\frac{1-\beta}{\alpha}, \end{aligned}$$

and hence the desired result. ■

Now, combining Lemma 1 and Lemma 2, we have the following result on the data complexity.

Theorem 3. *For n independent occurrences of the event e with probabilities p and $p(1+q)$ in two different distributions, the sample complexity of a distinguisher with false positive and negative rates α and β is given by*

$$n \geq \frac{1}{pq^2}\left(\beta\log_2\frac{\beta}{1-\alpha}+(1-\beta)\log_2\frac{1-\beta}{\alpha}\right),$$

for small p, q.

The equality may hold true only for the *Neymann-Pearson Test* [9], which is the optimal test, i.e., given a fixed level this test maximizes the power. This test is described by the *Fundamental Neymann-Pearson Lemma* [9].

Lemma 3 (Neymann-Pearson Lemma)**.** *Suppose we have $\boldsymbol{X} := (X_1, \ldots, X_n) \sim D_n$, where D_n is an unknown discrete distribution. We are to test the hypothesis $H_0 : D_n = P_n$ versus the alternative $H_1 : D_n = Q_n$. Suppose $\mathcal{S}$ be the set of all possible values that X_i's can take. Then take any arbitrary constant k and consider any test function $\phi : \mathcal{S} \longrightarrow [0, 1]$ satisfying the following conditions:*

$$\begin{aligned} \phi(\boldsymbol{x}) &= 0, if \ \frac{P_n[(x_1, \ldots, x_n)]}{Q_n[(x_1, \ldots, x_n)]} > k, \\ &= 1, if \ \frac{P_n[(x_1, \ldots, x_n)]}{Q_n[(x_1, \ldots, x_n)]} < k. \end{aligned}$$

Define $\alpha = E_{H_0}[\phi(\boldsymbol{X})]$, and $\beta = 1 - E_{H_1}[\phi(\boldsymbol{X})]$. Then for any other test (may be randomized) for the above hypothesis with error probabilities α' and β', we have

$$\alpha' \leq \alpha \Rightarrow \beta \leq \beta'.$$

In other words the test satisfying the conditions stated is the most powerful level α test.

Now though *Fundamental Neymann-Pearson Lemma* gives us the optimum test, the exact values of the optimum error probabilities for this test is difficult to find in general case. So, in that case we use some approximation techniques one of which is just discussed. In this situation, for $\alpha = \beta$, the relation (1) reduces to

$$n \geq \left(\frac{1}{pq^2}\right) \cdot (1 - 2\alpha) \log_2 \frac{1 - \alpha}{\alpha}.$$

In our context, P_n is the distribution of n *i.i.d.* Bernoulli trials with success probability $p(1 + q)$ where Q_n is the same with success probability p. Here *false positive* means that the test sequence is actually from the stream cipher, but we decide it to be random and *false negative* means that the test sequence is actually random, but we decide it to be from the stream cipher.

Thus, for a given false positive or negative rate α $(= \beta)$, one needs roughly $O(1/pq^2)$ many samples to perform the distinguishing test. In particular, $n \geq 1/pq^2$ signifies $\alpha \approx 0.2227$, i.e., a success probability of approximately 0.7773. Since $0.7773 > 0.5$ is a reasonably good success probability, $O(1/pq^2)$ many samples are considered enough to reliably apply the distinguisher.

D. Asymptotic Approach

Another method to find the expression for the error probabilities for the optimum test is to use the asymptotic analysis given by *Chernoff-Stein Lemma* [10]. This approach has been used by [11] to mount distinguishing attack on the stream-cipher HC-128 [12].

Lemma 4 (Chernoff-Stein Lemma)**.** *Suppose we have $X_1, \ldots, X_n \overset{i.i.d.}{\sim} D$, where D is unknown. We are to test the hypothesis $H_0 : D = P$ versus the alternative $H_1 : D = Q$, where P and Q are two known distributions. Suppose χ be the set of all possible values that X_i's can take. Suppose,P_n and Q_n are the joint distributions of $(X_1, \ldots, X_n)$ under the null and the alternative respectively. Let us fix $0 < \alpha < 0.5$. Define,*

$$\beta_{n,\alpha} := \min\left\{\beta | \mathcal{R} \subset \chi^n, P_n[\mathcal{R}] < \alpha, \beta = 1 - Q_n[\mathcal{R}]\right\},$$

In other words, $\beta_{n,\alpha}$ is the least false negative error probability attainable for level α non-randomized tests. Then

$$\lim_{n \to \infty} \frac{\log_2 \beta_{n,\alpha}}{n} = -D_{KL}(P || Q).$$

Proof: This proof of *Chernoff-Stein Lemma* occurs in [10]. First note that,

$$\log_2\left[\frac{P_n(X_1,\ldots,X_n)}{Q(X_1,\ldots,X_n)}\right]=\sum_{k=1}^{n}\log_2\left[\frac{P(X_k)}{Q(X_k)}\right],$$

and by *Law of Large numbers*

$$\begin{aligned}\frac{1}{n}\sum_{k=1}^{n}\log_2\left[\frac{P(X_k)}{Q(X_k)}\right] &\xrightarrow{p} E_P\left[\log_2\left(\frac{P(X_1)}{Q(X_1)}\right)\right]\\ &= D_{KL}(P||Q),\end{aligned}$$

under the null. Hence,

$$\frac{1}{n}\log_2\left[\frac{P_n(X_1,\ldots,X_n)}{Q(X_1,\ldots,X_n)}\right]\xrightarrow{p} D_{KL}(P||Q),$$

which by definition gives that $\forall \epsilon,\delta>0$, $\exists\ N_{\epsilon,\delta}\in\mathbb{N}$ such that, $\forall\, n\geq N_{\epsilon,\delta}$, we have

$$P_n\left[|\frac{1}{n}\log_2\left[\frac{P_n(\boldsymbol{X})}{Q_n(\boldsymbol{X})}\right]-D_{KL}(P||Q)|<\epsilon\right]\geq 1-\delta, \tag{5}$$

where $D=D_{KL}(P||Q)$ and $\boldsymbol{X}=(X_1,\ldots,X_n)$. Now, define A_n^{ϵ} be the subset of χ^n consisting of all $\boldsymbol{x}=(x_1,\ldots,x_n)$ such that

$$P_n(\boldsymbol{x})2^{-n(D+\epsilon)}<Q_n(\boldsymbol{x})<P_n(\boldsymbol{x})2^{-n(D-\epsilon)},$$

i.e.,

$$|\frac{1}{n}\log_2\left[\frac{P_n(\boldsymbol{x})}{Q_n(\boldsymbol{x})}\right]-D|<\epsilon.$$

Then, Equation (5) gives,

$$P_n(A_n^{\epsilon})\geq 1-\delta,$$

$\forall\, n\geq N_{\epsilon,\delta}$. Also note that

$$\begin{aligned}Q_n(A_n^{\epsilon}) &= \sum_{\boldsymbol{x}\in A_n^{\epsilon}}Q_n((\boldsymbol{x})) &(6)\\ &< \sum_{\boldsymbol{x}\in A_n^{\epsilon}}P_n((\boldsymbol{x}))2^{-n(D-\epsilon)}<2^{-n(D-\epsilon)}, &(7)\end{aligned}$$

and

$$\begin{aligned}Q_n(A_n^{\epsilon}) &= \sum_{\boldsymbol{x}\in A_n^{\epsilon}}Q_n((\boldsymbol{x})) &(8)\\ &> \sum_{\boldsymbol{x}\in A_n^{\epsilon}}P_n((\boldsymbol{x}))2^{-n(D+\epsilon)} &(9)\\ &= 2^{-n(D+\epsilon)}P_n(A_n^{\epsilon})\geq(1-\delta)2^{-n(D+\epsilon)}, &(10)\end{aligned}$$

$\forall\, n\geq N_{\epsilon,\delta}$.

Now consider the test which rejects the null if and only if $\boldsymbol{x}\notin A_n^{\epsilon}$. Then, by equation (6) $\forall\, n\geq N_{\epsilon,\alpha}$,

$$1-P_n(A_n^{\epsilon})<\alpha, Q_n(A_n^{\epsilon})<2^{-n(D-\epsilon)}\Longrightarrow \beta_{n,\alpha}<2^{-n(D-\epsilon)}.$$

Thus we have, $\forall\, n\geq N_{\epsilon,\alpha},\epsilon>0$,

$$\frac{\log_2\beta_{n,\alpha}}{n}<-D+\epsilon\Longrightarrow \limsup_{n\to\infty}\frac{\log_2\beta_{n,\alpha}}{n}\leq -D. \tag{11}$$

On the other hand consider any other test with rejection region $\mathcal{R}$, such that $P_n(\mathcal{R}) < \alpha$. Then we have,$\forall\, n \geq N_{\epsilon,\alpha}$,

$$\begin{aligned} Q_n(\mathcal{R}^c) &\geq Q_n(\mathcal{R}^c \cap A_n^\epsilon) \\ &= \sum_{\boldsymbol{x} \in \mathcal{R}^c \cap A_n^\epsilon} Q_n(\boldsymbol{x}) \\ &> \sum_{\boldsymbol{x} \in \mathcal{R}^c \cap A_n^\epsilon} 2^{-n(D+\epsilon)} P_n(\boldsymbol{x}) \\ &= 2^{-n(D+\epsilon)} P_n(\mathcal{R}^c \cap A_n^\epsilon) \\ &= 2^{-n(D+\epsilon)} (P_n(A_n^\epsilon) - P_n(\mathcal{R})) \\ &\geq 2^{-n(D+\epsilon)} (1 - 2\alpha) \end{aligned}$$

Hence,$\forall\, n \geq N_{\epsilon,\alpha}$,

$$\beta_{n,\alpha} = \min_{\mathcal{R}, P_n(\mathcal{R}) < \alpha} Q_n(\mathcal{R}^c) > 2^{-n(D+\epsilon)}(1 - 2\alpha),$$

which in turn gives,

$$\liminf_{n \to \infty} \frac{\log_2 \beta_{n,\alpha}}{n} \geq -D - \epsilon,\ \forall\, \epsilon > 0.$$

Therefore,

$$\liminf_{n \to \infty} \frac{\log_2 \beta_{n,\alpha}}{n} \geq -D. \tag{12}$$

Combining Equation (11) and Equation (12) we get the desired result. ■

The above lemma states that, whatever be the pre-specified false positive error, asymptotically the best possible false negative error is $2^{-nD_{KL}(P||Q)}$. Suppose now that we fix the false positive error at α and want false negative error to be β. Then the approximate sample size we need is $n \approx -\log_2(\beta)/D_{KL}(P||Q)$. Therefore, combining Lemma 1 and Lemma 4, we have the following result on the data complexity.

Theorem 4. *For n independent occurrences of the event e with probabilities p and $p(1+q)$ in two different distributions, the sample complexity of a distinguisher with false positive and negative rates α and β is given by*

$$n > -\frac{1}{pq^2} \log_2(\beta),$$

for small p, q, and small β.

Remark. Note that, in Theorem 4, β is required to be small, as Lemma 4 is an asymptotic result. So the best possible false negative rate is well approximated for large n and consequently for small β.

E. Comparison amongst the Above Approaches

So far we have discussed four bounds on data complexities obtained from Theorem 1, Theorem 2, Theorem 3 and Theorem 4. We now compare these bounds for small values of p, q and small failure rate α (i.e., both the error rate is α). We first exclude the bound from Theorem 1 from this comparison, as it doesn't consider the error rate in its derivation.

We observe that the bound obtained from Theorem 2 is relatively larger than other two bounds and the "actual complexity"(which is defined as the minimum over the number of samples needed to distinguish between two distributions by all possible test procedures with given false positive and false negative rates). This phenomenon occurs as Theorem 2 considers only non-randomized tests to distinguish between the two distributions and focuses only on the tests which does not reject H_i when the observed value of the data is close to its expectation under H_i, $i = 0, 1$.

On the other hand, the bound given by Theorem 3 is the smallest as it considers all the randomized tests. Moreover, equality holds in this bound only under some strict conditions which will imply the equality condition in *Jensen's inequality*. So, the actual complexity is always somewhat bigger than this bound.

Theorem 4 also considers all randomized tests. However, this bound is derived from an asymptotic result, and hence will be greater than the bound from Theorem 3. It is also easy to prove that the bound obtained from Theorem 4 is larger than that from Theorem 3 for fixed false negative and false positive error rate (both equal to α), as

$$\begin{aligned}\alpha \in (0, \frac{1}{2}) \quad &\Rightarrow \quad (1-\alpha)^{1-2\alpha} < \alpha^{-2\alpha} \\ &\Rightarrow \quad (1-2\alpha)(\log_2(1-\alpha) - \log_2 \alpha) < -\log_2 \alpha \\ &\Rightarrow \quad (1-2\alpha)\log_2\left(\frac{1-\alpha}{\alpha}\right) < -\log_2 \alpha.\end{aligned}$$

Thus, the bound obtained from Theorem 4 lies in between those given by Theorem 2 and Theorem 3.

Thus, the actual complexity and the bound obtain from Theorem 4, both lies between the other two bounds, and they become very close to each other as the error rate becomes small. Now, it is natural to ask: *which bound to use for estimating the data complexity?* From the context, it is clear that for small error rate the bound from Theorem 4 should be preferred. Otherwise, it is better to use the bound from Theorem 2, as this bound is greater than the actual complexity.

F. Other Related Works

In the above discussion, we have considered distinguishing tests which minimize the false negative error for a given false positive error level. The work [13] considered another paradigm considering the tests minimizing the average of both kind of error rates, and derived the data complexity of the optimal distinguishing test in that scenario, which is also of $O(1/pq^2)$, for small p, q. However, we note that joint minimization of both types of errors is an unusual approach in hypothesis testing framework. In practical scenario, often one would like to strictly bound a particular type of error. Thus, it is more pragmatic to fix one error and minimize the other. Moreover, the main focus of [13] is block cipher cryptanalysis and here we concentrate on stream ciphers.

The work [14] considers data complexity of a particular differential cryptanalysis with a set of 2^m sequences, where only one of them verifies the alternative hypothesis and all others verify the null hypothesis. The scenario we consider is completely different and hence we do not discuss the work [14] here.

A recent work [15] that has been carried out simultaneously with and independently of our current work, takes a detailed look at the error in normal approximations and points out several limitations in applying these approximations to block cipher cryptanalysis. A more recent work [16] (by the same authors as [15]) derives rigorous upper bounds on the data complexity (i.e., the no. of plaintext-ciphertext pairs) required to achieve at least a pre-specied success probability (for key recovery of a block cipher) and at least a pre-specified advantage (if the advantage is a, then the number of false alarms is a fraction 2^{-a} of the number of possible values of the sub-key which is the target of the attack). We emphasize again that our motivation is completely different, i.e., to analyze the gap between the data complexity estimates of distinguishing attacks and message recovery attacks in the context of stream ciphers.

III. Data Complexity of Message Recovery Attacks using Sample Mode Approach

Now we turn to Message Recovery Attack under the *Broadcast* scenario for stream ciphers. We shall consider two approaches: one is a simple message recovery attack exploiting the largest bias in the key-stream byte distribution and the other is maximum likelihood estimation. The first one is called Sample

Mode approach discussed in this section and the second one is called Bayesian approach, discussed in the next section.

We consider again the single byte-bias attack. Suppose m be the mode of the distribution $\mathcal{P}$. Here, we assume that the distribution is unimodal (note that if the distribution is multimodal, then this approach will fail). Suppose M is the secret message and $Z_1, \ldots, Z_n \overset{i.i.d}{\sim} \mathcal{P}$ be the keys for n broadcasts. We observe the ciphertexts $C_i = M \oplus Z_i(\mod N)$, $\forall i = 1, \ldots, N$.

Now, as m is the mode, we expect it to occur more frequently in the i.i.d. sample $Z_1, \ldots, Z_n$ and hence we expect $M + m(\mod N)$ to be most frequent in the ciphertext sample $C_1, \ldots, C_n$. Hence, we estimate (or guess) M by $\hat{M} = (Mode(C_1, \ldots, C_n) - m)(\mod N)$. So, here the probability of success is

$$\begin{aligned} & \Pr(\hat{M} = M) \\ = \ & \Pr[(Mode(C_1, \ldots, C_n) - m)(\mod N) = M] \\ = \ & \Pr(Mode(Z_1, \ldots, Z_n) = m). \end{aligned}$$

So, it boils down to the problem of finding the probability of sample mode being equal to the population mode for an *i.i.d.* sample from distribution $\mathcal{P}$. Suppose Y_i be the frequency of k in the sample $\mathbf{Z} = (Z_1, \ldots, Z_n)$, $\forall\, k = 0, \ldots, N-1$. Then

$$\mathbf{Y} := (Y_0, \ldots, Y_{N-1}) \sim Multinomial(n; p_0, \ldots, p_{N-1});$$

where $p_0, \ldots, p_{N-1}$ are the p.m.f.'s for the corresponding points. For simplicity of notation we assume $m = 0$ (however, the result holds for any mode). So,

$$\Pr(Mode(Z_1, \ldots, Z_n) = 0) = \Pr(Y_0 \geq Y_k, \ \forall\, k).$$

First note that, by *Law of Large Numbers*, we have,

$$\frac{1}{n}(Y_0, \ldots, Y_{N-1}) \xrightarrow{a.s.} (p_0, \ldots, p_{N-1});$$

i.e.

$$\frac{1}{n}(Y_0 - Y_1, \ldots, Y_0 - Y_{N-1}) \xrightarrow{a.s.} (p_0 - p_1, \ldots, p_0 - p_{N-1}). \tag{13}$$

Since we have $p_0 > p_k, \ \forall k = 1, \ldots, N-1$, we get from (8),

$$\Pr(Y_0 > Y_k) \longrightarrow 1, \ \forall\, k = 1, \ldots, N-1.$$

Thus, $\Pr(Mode(Z_1, \ldots, Z_n) = m) \longrightarrow 1$, as $n \to \infty$. Thus $\hat{M} \xrightarrow{p} M$, as $n \to \infty$, i.e., the estimator is at least consistent.

Again, by *Central Limit Theorem*, we have

$$\mathbf{Y} \sim AN(n\mathbf{p}, n(diag(\mathbf{p}) - \mathbf{p}\mathbf{p}^T));$$

i.e.,

$$n^{-1}\mathbf{Y} \sim AN(\mathbf{p}, n^{-1}(diag(\mathbf{p}) - \mathbf{p}\mathbf{p}^T)),$$

where $\mathbf{p} := (p_0, \ldots, p_{N-1})^T$. So, for large n we have by normal approximation

$$\begin{aligned} & \Pr(Y_0 \geq Y_k, \ \forall\, k = 0, \ldots, N-1) \\ \approx \ & \Pr(U_0 \geq U_k, \ \forall\, k = 0, \ldots, N-1); \end{aligned}$$

where $\mathbf{U} := (U_0, \ldots, U_{N-1})^T \sim \mathcal{N}_N(\mathbf{p}, \frac{1}{n}(diag(\mathbf{p}) - \mathbf{p}\mathbf{p}^T))$.

But evaluating this probability is too nasty, even numerically. So, we consider two different cases and go through some further approximations.

A. Second Highest Probability far from Lowers Ones

In this case, we consider the situation where the second highest probability in the distribution $\mathcal{P}$ is distinguishably apart from the other lower probabilities in that distribution, i.e., if p_i and p_j are respectively the second and third highest probabilities in the distribution p then $p_i - p_j \neq 0$. Then we can come to the following result.

Theorem 5. *If $\mathcal{P}$ is the distribution of the keystream bytes on the space $\mathcal{M}$, with population mode 0, and the second and third highest probabilities of $\mathcal{P}$, say p_{N-2} and p_{N-3} are distinguishably apart, then the data complexity of the message recovery attack with failure probability at most α using sample mode approach is given by*

$$n > \left(\frac{\Phi^{-1}(1-\alpha)}{\delta_i}\right)^2, \tag{14}$$

where

$$\delta_k := \frac{p_0 - p_k}{\sqrt{p_0 + p_k - (p_0 - p_k)^2}} > 0, \ \forall\, k = 1, \ldots, N-1.$$

and $\delta_i = \min\{\delta_k : k = 1, \ldots, N-1\}$.

Proof: Note that

$$\begin{aligned} \sum_{r=1}^{N-1} \Pr(U_0 \leq U_r) &\geq \Pr(U_0 \leq U_k \ for\ some\ k) \\ &\geq \Pr(U_0 \leq U_l), \ \forall\, l = 1, \ldots, N-1. \end{aligned}$$

So, $\forall\, k = 1, \ldots, N-1$,
we have, $\Pr(U_0 \leq U_k) = \Pr(U_0 - U_k \leq 0)$ and

$$U_0 - U_k \sim \mathcal{N}(p_0 - p_k, \frac{1}{n}(p_0(1-p_0) + p_k(1-p_k) + 2p_0p_k)).$$

Hence,

$$\Pr(U_0 - U_k \leq 0) = \Phi(-\sqrt{n}\delta_k). \tag{15}$$

Thus,

$$\begin{aligned} \sum_{r=1}^{N-1} \Phi(-\sqrt{n}\delta_r) &\geq \Pr(U_0 \leq U_k \ for\ some\ k) \\ &\geq \Phi(-\sqrt{n}\delta_l), \ \forall\, l = 1, \ldots, N-1. \end{aligned}$$

As, p_i is the second highest probability, δ_k is minimum and hence $\Phi(-\sqrt{n}\delta_k)$ is maximum for $k = i$. Therefore,

$$\begin{aligned} \sum_{r=1}^{N-1} \Phi(-\sqrt{n}\delta_r) &\geq \Pr(U_0 \leq U_k \ for\ some\ k) \\ &\geq \Phi(-\sqrt{n}\delta_i). \end{aligned}$$

Now, we shall show that the ratio of the two extremes in the inequality stated above goes to 1 as n goes to infinity, i.e., for large n they are quite close and then we can approximate the middle term by the right-hand extreme. The limit we get by using *L'Hospital's Rule* as follows.

$$\begin{aligned} \lim_{n\to\infty} \frac{\Phi(-\sqrt{n}\delta_i)}{\sum_{r=1}^{N-1} \Phi(-\sqrt{n}\delta_r)} &= \lim_{n\to\infty} \frac{-\delta_i e^{\frac{-n\delta_i^2}{2}}}{\sum_{r=1}^{N-1} -\delta_r e^{\frac{-n\delta_r^2}{2}}} \\ &= 1 \end{aligned}$$

as $\delta_i \neq \delta_k,\ \forall k \neq 0, i$. So, $\Pr(U_0 \leq U_k\ for\ some\ k) \approx \Phi(-\sqrt{n}\delta_i)$. Hence to get success probability at least $1-\alpha$, we should have $\alpha \geq \Phi(-\sqrt{n}\delta_i)$. Hence, the sample size we need is

$$n \geq \left(\frac{\Phi^{-1}(1-\alpha)}{\delta_i}\right)^2 .$$

■

But the above approximations performs miserably if all the probabilities in the p.m.f. $\mathcal{P}$, except the highest one, are very close to each other as then the limit above converges very slowly. We shall consider this situation in the next case.

B. Almost-uniform Except the Highest Probability-point

Here, suppose all the probabilities in the p.m.f. $\mathcal{P}$, except the highest one i.e. p_0, are equal to r. Hence, $p_0 + (N-1)r = 1$. Let us define $\mathbf{V} = (V_1, \ldots, V_{N-1})^T := (U_0 - U_1, \ldots, U_0 - U_{N-1})^T$. Therefore, we have

$$\mathbf{V} \sim \mathcal{N}_{N-1}((p_0 - r)\mathbf{1}, \Sigma)$$

where $\mathbf{1}$ is the $(N-1)$-dimensional column vector with all entries equal to 1, and Σ is a positive-definite matrix with all diagonal entries equal to $\sigma^2 := n^{-1}(p_0 + r - (p_0 - r)^2)$ and all non-diagonal entries equal to $\rho\sigma^2 := n^{-1}(p_0 - (p_0 - r)^2)$, where σ^2 and ρ are the common variance and common correlation coefficient. So, $\mathbf{V}$ has the equicorrelation structure and the correlation coefficient $\rho > 0$. Now consider

$$W_0, \ldots, W_{N-1} \overset{i.i.d.}{\sim} \mathcal{N}(0,1);$$

and define

$$S_k := (p_0 - r) + \sigma(\sqrt{\rho}W_0 + \sqrt{1-\rho}W_k),\ \ \forall\ k = 1, \ldots, N-1.$$

Then, clearly $\mathbf{S} := (S_1, \ldots, S_{N-1})^T \overset{\mathcal{D}}{=} \mathbf{V}$. Hence, we have

$$\begin{aligned}
& \Pr(U_0 \geq U_k,\ \forall\ k = 0, \ldots, N-1) \\
= \ & \Pr(V_k \geq 0,\ \forall\ k = 0, \ldots, N-1) \\
= \ & \Pr(S_k \geq 0,\ \forall\ k = 0, \ldots, N-1) \\
= \ & \Pr(\sqrt{\rho}W_0 + \sqrt{1-\rho}W_k \geq \frac{-(p_0 - r)}{\sigma}, \\
& \quad \forall\ k = 0, \ldots, N-1) \\
= \ & \Pr[\sqrt{\rho}W_0 + \sqrt{1-\rho}W_{N-1} \geq \cdots \geq \sqrt{\rho}W_0 + \sqrt{1-\rho}W_1 \\
& \qquad\qquad \geq \frac{-(p_0 - r)}{\sigma}](N-1)! \\
= \ & \Pr(W_{N-1} \geq \cdots \geq W_1 \geq \\
& \quad -\frac{(p_0 - r)}{\sigma\sqrt{1-\rho}} - \frac{\sqrt{\rho}}{\sqrt{1-\rho}}W_0)(N-1)! \\
= \ & \Pr(W_{(1)} \geq T_0)
\end{aligned}$$

where we define $W_{(1)} := min\{W_j : 1 \leq j \leq N-1\}$ and $T_0 \sim \mathcal{N}(-\sqrt{n}\gamma, \sigma_0^2)$ and T_0 is independent of $(W_1, \ldots, W_{N-1})$, hence with $W_{(1)}$ and

$$\gamma := \frac{p_0 - r}{\sqrt{1-\rho}\sqrt{p_0 + r - (p_0 - r)^2}}, \tag{16}$$

$$\sigma_0^2 := \frac{\rho}{1-\rho}. \tag{17}$$

We also note that

$$\begin{aligned}
\Pr(W_{(1)} \geq T_0) &= \int_{\mathbb{R}} (1-\Phi(x))^{N-1} \frac{1}{\sigma_0} \phi\left(\frac{x+\sqrt{n}\gamma}{\sigma_0}\right) dx \\
&= E((1-\Phi(T_0))^{N-1})
\end{aligned}$$

Hence, for the data complexity of the message recovery attack in this case using sample mode approach we have the following result.

Theorem 6. *If the distribution $\mathcal{P}$ of the keystream bytes on the space $\mathcal{M}$, has only one mode at* 0 *and all remaining probabilities are equal, then the success probability of the message recovery attack using the sample mode approach is $E((1-\Phi(T_0))^{N-1})$, where $T_0 \sim \mathcal{N}(-\sqrt{n}\gamma, \sigma_0^2)$ and γ and σ^2 are as defined as in Equation* (16) *and* (17).

Remark. For general $p.m.f.'s$ on the message space, the above integral is very difficult to work out analytically. Hence, to proceed towards further analysis we must pass through numerical methods to approximate the above probability. Two possible ways are approximating the above integral by different available numerical integration method or simulating large number of times independently from the distribution of T_0 and take the sample mean of the function $(1-\Phi(\cdot))^{N-1}$ to obtain an approximation of the above expectation (by *Law of Large Numbers*).

IV. Data Complexity of Message Recovery Attacks using Bayesian Approach

The Bayesian method was first discussed in [3, Section 4.1], and it finally boils down to maximum likelihood estimation. Here we shall only explore the method for some general version of single-byte bias attack. The Bayesian approach for multiple-byte bias attacks are defined similarly. The set up is as follows.

Suppose, $\mathcal{M} = \{0, 1, \ldots, N-1\}$, where N is the size of the message space. For simplicity of language we shall call the elements of the message space as *bytes*. $\mathcal{P}$ is the distribution of the keystream bytes of the concerned stream cipher. Suppose M is the secret message-byte (or, plaintext) and $Z_1, \ldots, Z_n \overset{i.i.d}{\sim} \mathcal{P}$ be the keys for n broadcasts. We observe the ciphertext bytes $C_i = M \oplus Z_i,\ \forall i = 1, \ldots, n,$ where the XOR sum indicates *summing modulo* N.

We also assume a prior distribution on the message space $\mathcal{M}$, say $\mathcal{Q}$. Then q_x denotes the relative frequency of the message-byte x in a large message. If enough prior information is not available, this prior distribution is taken as uniform (i.e., a non-informative prior). In this Bayesian approach, we want to maximize the posterior probability of the message-byte given the ciphertext bytes, i.e., we want to maximize $\Pr(M = m | C_i = c_i,\ \forall\, i = 1, \ldots, n)$, over $m \in \mathcal{M}$.

Now, before going into the maximization problem, we introduce a notation

$$N_{m,z} := |\,\{i : c_i = z \oplus m\}\,| \tag{18}$$

$\forall\, z \in \mathcal{M}$. That means $N_{m,z}$ denotes the number of occurrences of the byte z in the n keystream bytes, where the plaintext is m and ciphertexts are $c_1, \ldots, c_n$. Now we have the following result according to [17].

Lemma 5. *Maximization of* $\Pr(M = m | C_i = c_i,\ \forall\ i = 1, \ldots, n)$, *over $m \in \mathcal{M}$ is equivalent to maximizing*

$$h(m) := \log(q_m) + \sum_{z \in \mathcal{M}} N_{m,z} \log(p_z),$$

over $m \in \mathcal{M}$.

Proof:

Note that if $c_1, \ldots, c_n$ are the observed ciphertexts then

$$\begin{aligned}
& \Pr(M = m | C_i = c_i,\ \forall\, i = 1, \ldots, n) \\
= \ & \Pr(C_i = c_i,\ \forall\, i = 1, \ldots, n | M = m) \\
& \frac{\Pr(M = m)}{\Pr(C_i = c_i,\ \forall\, i = 1, \ldots, n)} \\
= \ & \Pr(Z_i = z_i,\ \forall\, i = 1, \ldots, n) \frac{\Pr(M = m)}{\Pr(C_i = c_i,\ \forall\, i = 1, \ldots, n)};
\end{aligned}$$

where $z_i := c_i - m (\text{mod } N);\ \forall i = 1, \ldots, n$. Now the denominator in the above expression, $\Pr(C_i = c_i,\ \forall\, i = 1, \ldots, n)$ is independent of the choice of m, because

$$\begin{aligned}
& \Pr(C_i = c_i,\ \forall\, i = 1, \ldots, n) \\
= \ & \sum_{m \in \mathcal{M}} \Pr(Z_i = c_i - m (\text{mod } N), i = 1, \ldots, n).
\end{aligned}$$

So, we are to maximize only

$$\Pr(Z_i = z_i,\ \forall\, i = 1, \ldots, n) \Pr(M = m),$$

over $m \in \mathcal{M}$. As the broadcasts are independent, we have

$$\begin{aligned}
\Pr(Z_i = z_i, \forall i = 1, \ldots, n) &= \prod_{i=1}^{n} \Pr(Z_i = z_i) \\
&= \prod_{i=1}^{n} p_{z_i} = \prod_{z \in \mathcal{M}} p_z^{N_{m,z}},
\end{aligned}$$

and we are to maximize

$$g(m) := \Pr(M = m) \prod_{z \in \mathcal{M}} p_z^{N_{m,z}} = q_m \prod_{z \in \mathcal{M}} p_z^{N_{m,z}}, \tag{19}$$

over $m \in \mathcal{M}$. Taking $\log$ on both sides in (6) (as computationally it is easy to work with the function after logarithm), we get

$$h(m) := \log g(m) = \log(q_m) + \sum_{z \in \mathcal{M}} N_{m,z} \log(p_z). \tag{20}$$

■

So, we get the estimator for the unknown plaintext byte as

$$\hat{M} := \arg\max_{m \in \mathcal{M}} g(m).$$

If prior information is not available and we take $\mathcal{Q}$ to be uniform over the message space, then our objective boils down to maximizing

$$h_0(m) := \sum_{z \in \mathcal{M}} N_{m,z} \log(p_z); \tag{21}$$

over $m \in \mathcal{M}$, and this objective function is nothing but the constant times log-likelihood for the data $C_1, \ldots, C_n \overset{i.i.d}{\sim} \mathcal{P} \oplus m$, where $m \in \mathcal{M}$ acts as the unknown parameter. So, in this case, this is the maximum likelihood estimator.

Now, if we use the above idea for multiple-byte bias attack, the message space $\mathcal{M}$ will be substituted by $\mathcal{M}^k$, for some $k \in \mathbb{N}$. The basic methodology remains same. For this type of attack, we need a prior distribution $\mathcal{Q}^{(k)}$ on $\mathcal{M}^k$, as the prior distribution of the plaintexts, and the distribution $\mathcal{P}^{(k)}$ of vector of k keystream-bytes, where we shall exploit the biases in the later distribution to mount the message recovery attack. Let $p_{\boldsymbol{z}}^{(k)}$ and $q_{\boldsymbol{z}}^{(k)}$ denote the probabilities of a k-byte vector $\boldsymbol{z}$ in the keystream according to the corresponding distributions. By similar arguments used in Lemma 5 our objective is to maximize

$$h(\boldsymbol{m}) = \log(q_{\boldsymbol{m}}^{(k)}) + \sum_{\boldsymbol{z}\in\mathcal{M}^k} N_{\boldsymbol{m},\boldsymbol{z}} \log(p_{\boldsymbol{z}}^{(k)}),$$

over $\boldsymbol{m} \in \mathcal{M}^k$, where $N_{\boldsymbol{m},\boldsymbol{z}}$ is defined similarly for a vector of k bytes as previous.

The above maximization may be computationally difficult as $\mathcal{M}^k$ may contain a very large number of elements. As for example, in the case of RC4, only for k=3,4 it becomes of high complexity. So, we go for some approximation techniques.

One of these approximation techniques is based on the assumption that if $\boldsymbol{Z}_i := (Z_{i1}, \ldots, Z_{ik})$ be the i-th sample, then $Z_{i1}, \ldots, Z_{ik}$'s are independent for $i = 1, \ldots, n$. This approximation is very good for the cases where the single-byte biases are dominant. Then we have

$$p_{\boldsymbol{z}}^{(k)} = \prod_{i=1}^{k} p_{z_{(i)}},$$

for all $\boldsymbol{z} = (z_{(1)}, \ldots, z_{(k)}) \in \mathcal{M}^k$. Our new objective function becomes

$$\begin{aligned} h(\boldsymbol{m}) &= \log(q_{\boldsymbol{m}}^{(k)}) + \sum_{\boldsymbol{z}\in\mathcal{M}^k} \sum_{i=1}^{k} N_{\boldsymbol{m},\boldsymbol{z}} \log(p_{z_{(i)}}) \\ &= \log(q_{\boldsymbol{m}}^{(k)}) + \sum_{i=1}^{k} \sum_{z\in\mathcal{M}} N_{m_{(i)},z} \log(p_z) \end{aligned}$$

where $N_{m_{(i)},z} := |\left\{j : c_{j(i)} = z \oplus m_{(i)}\right\}|$, and $(c_{i(1)}, \ldots, c_{i(k)})$ be the i-th ciphertext sample.

If $q_{m_{(1)},\ldots,m_{(k)}}^{(k)} = q_{m_{(1)}} \ldots q_{m_{(k)}}$ (note that this is a weaker assumption than uniformity of $q^{(k)}$), then it reduces to nothing but single-byte bias attack at k plaintext points.

Another approximation reduces multiple-byte bias attacks to double byte bias attacks. For this, we make the Markovian assumption, i.e., for the i-th key $\boldsymbol{Z}_i$, the random variables $Z_{i1}, \ldots, Z_{ik}$ satisfies the Markov property. Therefore,

$$\begin{aligned} p_{\boldsymbol{z}}^{(k)} &= P[\boldsymbol{Z}_i = \boldsymbol{z}] \\ &= P[Z_{i1} = z_{(1)}] \prod_{j=2}^{k} P[Z_{ij} = z_{(j)} | Z_{i(j-1)} = z_{(j-1)}] \\ &= \frac{\prod_{j=2}^{k} p_{(z_{(j)},z_{(j-1)})}^{(2)}}{\prod_{j=2}^{k-1} p_{z_{(j)}}} \quad \text{[Note: } k = 2 \text{ here]} \end{aligned}$$

So, the objective function turns out to be

$$\begin{aligned} h(\boldsymbol{m}) &= \log(q^{(k)}_{\boldsymbol{m}}) + \sum_{j=2}^{k} \sum_{\boldsymbol{z} \in \mathcal{M}^k} N_{\boldsymbol{m},\boldsymbol{z}} \log(p^{(2)}_{(z_{(j)}, z_{(j-1)})}) \\ &\qquad - \sum_{\boldsymbol{z} \in \mathcal{M}^k} \sum_{j=2}^{k-1} N_{\boldsymbol{m},\boldsymbol{z}} \log(p_{z_{(j)}}) \\ &= \log(q^{(k)}_{\boldsymbol{m}}) + \sum_{j=2}^{k} \sum_{z,y \in \mathcal{M}} N_{m_{(j)}, m_{(j-1)}, z, y} \log(p^{(2)}_{(z,y)}) \\ &\qquad - \sum_{j=2}^{k-1} \sum_{z \in \mathcal{M}} N_{m_{(j)}, z} \log(p_z), \end{aligned}$$

where $N_{m_{(i)}, m_{(i-1)}, z, y}$ is equal to

$$|\left\{ j : c_{j(i)} = z \oplus m_{(i)}, c_{j(i-1)} = y \oplus m_{(i-1)} \right\}|$$

We shall now discuss about the performance of this Bayesian estimation technique. We shall concentrate only on the case where the prior distribution is taken to be uniform and hence the procedure boils down to ML estimation. It is to be noted that the exact calculation of the success probabilities is very difficult in this set up. So we shall continue by doing some approximate calculations. We shall consider two cases and concentrate on the single-byte bias attack.

A. Unimodal Keystream Distribution

Here, we shall consider the situation where the keystream-byte distribution is unimodal. In that case, suppose the probabilities for the keystream-bytes (in the distribution $\mathcal{P}$) are $p_{(0)} \leq p_{(1)} \leq \cdots \leq p_{(N-2)} < p_{(N-1)}$, in increasing order corresponding to the bytes $k_0, \ldots, k_{N-1}$ (these bytes are nothing but a permutation of $0, \ldots, N-1$ in increasing order of their probabilities). Then in long run, i.e., for large sample size, by *Law of Large Numbers*, we have $Y_{k_0} \leq Y_{k_1} \leq \cdots \leq Y_{k_{N-2}} < Y_{k_{N-1}}$ with probability 1, where Y_i denotes the number of occurrences of byte i in $Z_1, \ldots, Z_n$. On the other hand, considering Equation (21), we are to maximize

$$\sum_{z \in \mathcal{M}} N_{m,z} \log(p_z) = \sum_{z=0}^{N-1} N_{m,k_z} \log(p_{k_z}),$$

over m.

By *Rearrangement Inequality* [18], it will be maximized at $m \in \mathcal{M}$ if $N_{m,k_0} \leq \cdots \leq N_{m,k_{N-1}}$. If the sample mode of the ciphertext stream is c, then for large sample size the plaintext must be $m^* = c - k_{N-1} (\mod N)$ with very high probability and we also note that $N_{m^*,k_0} \leq \cdots \leq N_{m^*,k_{N-1}}$ for long run with very high probability. And therefore in this case the maximum likelihood estimate would be equal to the estimate based on the sample mode (discussed thoroughly in next section) for large sample size. The data complexity can then be estimated by Theorem 5 and Theorem 6.

B. Multimodal Keystream Distribution

Now we shall concentrate on multi-modal distribution (by multi-modal, we actually intend to mean that the highest probabilities have negligible difference) for the key-stream bytes. Like previous case, here also we need $N_{m,k_0} \leq \cdots \leq N_{m,k_{N-1}}$, for some m to get direct optimization using *Rearrangement Inequality* [18]. Here we may have more than one sample modes for the ciphertexts (or the sample mode of the ciphertext may not be equal to the actual plaintext XOR-ed with the population mode of the key-stream distribution, i.e., the population and sample mode for the Z's may differ even in the long run). But if the distributions of $\mathcal{P} \oplus m$, where $m \in \mathcal{M}$, are different for different m's, then also the ML estimation succeeds in long run as it not only take into account the sample mode of the ciphertexts but also other points, and hence able to distinguish between the distributions $\mathcal{P} \oplus m$, for all possible m's.

To estimate the data complexity of the ML method based attack in this scenario, we shall use Theorem 5 and Theorem 6. Recall that our ML estimate was

$$\hat{M} = \arg\max_{m \in \mathcal{M}} \sum_{z \in \mathcal{M}} N_{m,z} \log(p_z), \tag{22}$$

Then, define

$$\boldsymbol{t}_k := (\log(p_{0-k}), \ldots, \log(p_{(N-1)-k}))^T, \forall\, k = 0, \ldots N-1,$$

and

$$\boldsymbol{N} := (N_{0,0}, \ldots, N_{0,N-1})^T.$$

Now, as $N_{m \oplus k, z} = N_{m, z \oplus k}$, it is easy to see that Equation 22 reduces to

$$\hat{M} = \arg\max_{m \in \mathcal{M}} \boldsymbol{t}_m{}^T \boldsymbol{N}. \tag{23}$$

Note that, $N_{0,z}$ is the number of occurrences of the byte z in the n obtained cipher-text bytes. Suppose the unknown plain-text is m^*. Define, $\boldsymbol{q}^T := (p_{0-m^*}, \ldots, p_{N-1-m^*})$. Then, we have

$$\boldsymbol{N} \sim Multinomial(n; \boldsymbol{q}^T),$$

and hence, by *Central Limit Theorem*,

$$n^{-1}\boldsymbol{N} \sim A\mathcal{N}(\boldsymbol{q}, n^{-1}(diag(\boldsymbol{q}) - \boldsymbol{q}\boldsymbol{q}^T)) \tag{24}$$

Now, define

$$\boldsymbol{N}_0 := (\boldsymbol{t}_0^T \boldsymbol{N}, \ldots, \boldsymbol{t}_{N-1}^T \boldsymbol{N}) = (R_0, \ldots, R_{N-1})^T.$$

and let $\boldsymbol{t}_i^T \boldsymbol{q} = r_i$, $\forall\, i = 0, \ldots, N-1$. Then by (24) we have,

$$n^{-1}\boldsymbol{N}_0 \sim A\mathcal{N}((r_0, \ldots, r_{N-1})^T, n^{-1}\boldsymbol{\Sigma}'),$$

where $\boldsymbol{\Sigma}' = ((\sigma_{ij}))_{i,j}$ is the corresponding covariance matrix. Note that r_i is maximum if $i = m^*$, and

$$r_{m^*} = \sum_{k=0}^{N-1} p_k \log(p_k),$$

i.e., doesn't depend on the value of m^*. Now, we can readily recognize the set-up as the same which was in Theorem 5, as here success probability is $\Pr(\hat{M} = m^*)$, which is nothing but the probability of the maximum of $\boldsymbol{N}_0$s co-ordinates occurring in the co-ordinate with highest mean. So, like previous we define

$$\eta_k := \frac{r_{m^*} - r_k}{\sqrt{\sigma_{m^*m^*} + \sigma_{kk} - 2\sigma_{m*k}}}; \forall\, k \neq m^*. \tag{25}$$

Note that this quantities doesn't depend on the absolute value of m^*, only just the difference between m^* and k. Hence, their ordered sequence is fixed and known, call it $\eta = \eta_{(1)} \leq \eta_{(2)} \leq \cdots \leq \eta_{(N-1)}$. Then using Theorem 5, we have the following result.

Corollary 1. *If $\eta_{(1)}$ and $\eta_{(2)}$ are distinguishably apart, then the data complexity of the message recovery attack with failure probability at most α using ML approach is given by*

$$n > \left(\frac{\Phi^{-1}(1-\alpha)}{\eta}\right)^2, \tag{26}$$

where $\eta_{(1)}, \eta_{(2)}, \eta$ are as defined as in Equation 25.

The condition needed here is that $\eta_{(1)}$ and $\eta_{(2)}$ are distinguishably apart, which doesn't hold true implies the distribution of the key-stream bytes is nearly non-identifiable, and hence ML method performs miserably.

However, if there are two or more $m \in \mathcal{M}$ such that $\mathcal{P} \oplus m$ have same distributions, then ML method fail. We take an example for this. Consider $\mathcal{M} = \{0, \dots, 255\}$ and suppose that the distribution $\mathcal{P}$ is such that 0 and 128 are two modes and all other probabilities are equal. Then if the objective function is maximized in m then will also be maximized in $m \oplus 128$ as in this case $h_0(m) = h_0(m \oplus 128)$. So, ML method would fail in all cases in this example. Not only ML method, any estimator fails here miserably as the parameter m is not *identifiable* [19] in this case.

C. Comparison between Bayesian and Sample Mode Approach

So far we have discussed two ways for message recovery attack, one by *Sample Mode Approach* and another by *Bayesian Approach*.

In this context we would like to point out the following result for unimodal case of Bayesian approach.

Lemma 6. *If the distribution $\mathcal{P}$ of the keystream bytes on the space $\mathcal{M}$, has only one mode at 0 and all remaining probabilities are equal, then the message recovery attack using ML estimation and sample mode approach give the same result.*

Proof: Here ML method maximizes

$$\begin{aligned} h_0(m) &= \sum_{k=1}^{N-1} N_{m,k} \log r + N_{m,0} \log p_0 \\ &= N_{m,0} \log p_0 + (n - N_{m,0}) \log r; \end{aligned}$$

i.e., practically we are to maximize $N_{m,0}(\log p_0 - \log r)$ therefore only $N_{m,0}$ over m. Clearly, it is maximized if we take m to be equal to sample mode of the ciphertexts and thus the two estimates coincide. ■

Note that if the mode is different from 0, then also the same result holds.

In general, as the Bayesian approach uses more information about the keystream distribution (as it considers biases in all positions) than in the later approach, it has always greater success probability. But on the other hand, in the first approach, we have to maximize a complex function over a huge set, which may turn out to be computationally inefficient sometime. Therefore, we may follow the following rules while deciding which method to apply:

1) As discussed earlier, for unimodal key-stream byte distribution, both methods gives same estimate for large sample size, and hence in that we should go for computationally more efficient sample mode approach.
2) For small sample sizes or multi-modal distributions (with identifiable plaintext parameter), we should go for Bayesian approach.
3) For, non-identifiable plaintext case, both methods fail miserably, and hence none is preferred.

V. Connecting the Complexities of Distinguisher and Message Recovery: a Case Study

We are interested in the relation between the data complexities of *Distinguishing* and *Message Recovery Attack*. We define the function *Distinguish-equivalent* on the set of natural numbers as follows.

Definition 1. *Distinguish-equivalent*(n) is the number of samples needed in the distinguishing attack to have the same success probability as that in the message recovery attack for sample size equal to n.

To compare the data complexities in the two cases, we define another function *Multiplier* on the set of natural number as follows.

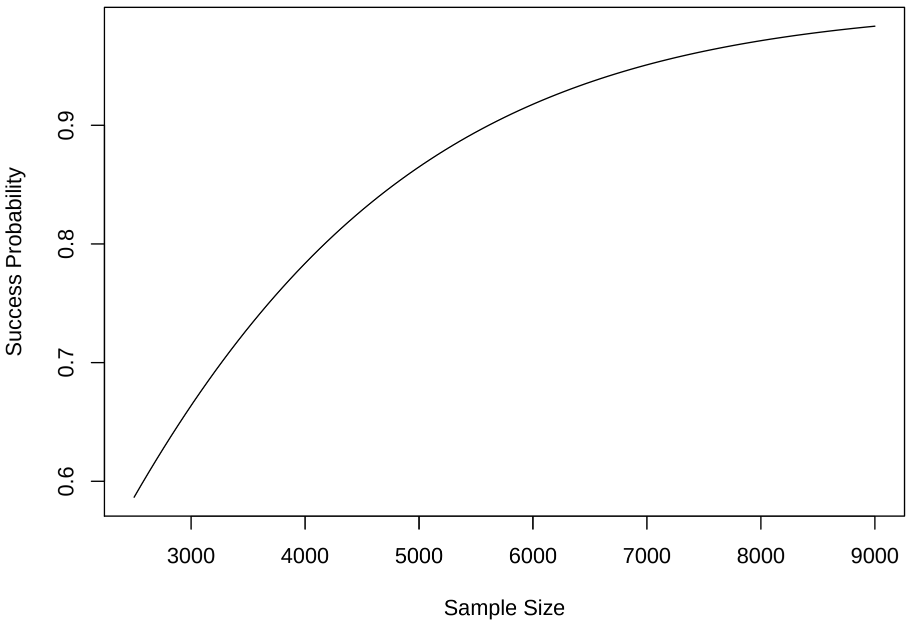

Fig. 1. Success Rate for Different Sample Sizes for RC4 Second Byte

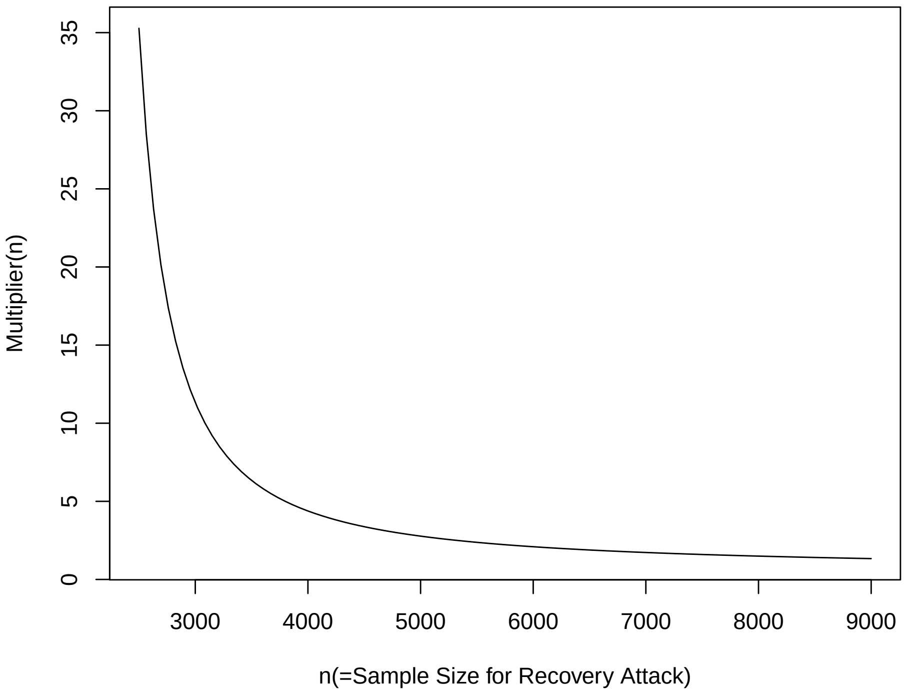

Fig. 2. Multiplier for Different Sample size in Recovery Attack

Definition 2. *Multiplier*(n) is the ratio $\frac{n}{\mathit{Distinguish\text{-}equivalent}(n)}$, which indicates how many times more sample is needed to recover the message than to only distinguish from uniform distribution with same probability of success.

Due to the remark after Theorem 6, derivation of a closed-form expression of this quantity is not possible.

Now, we illustrate our previously derived results by pursuing the attack on RC4 stream cipher based on its second byte bias. Here $N = 256$ and the distributional node is 0 with $p_0 \approx \frac{2}{256}$. In the other sample points, biases are very small which makes the other probabilities (i.e., $p_1, \ldots, p_{255}$) almost equal. Therefore, according to Lemma 6, one may use either ML or Sample Mode approach.

In a broadcast attack scenario, we have same message encrypted by different RC4 keystreams (say n times). We collect the second byte of the ciphertexts $C_1, \ldots, C_n$ and guess the secret message by $Mode(C_1, \ldots, C_n)$. Our probability of success is given by the above integral or expectation. In this particular context

$$\gamma = 0.06286849 \ , \ \sigma_0^2 = 2.003922.$$

The change in success probability for different sample size is given in *Fig. 1*.

The behaviour of the *Multiplier* function with n is shown in *Fig. 2*, where the distinguishing data complexity is calculated using the result stated in *Theorem 2* and taking the both way success probabilities equal (i.e., equal false positive and negative errors).

From *Fig.2* we see that the *Multiplier* function is continuously decreasing and decreasing very rapidly at the recovery attack sample size 2400 to 3000 (i.e., at success rate in $[0.55, 0.66]$. At success rate 0.7, i.e., near sample size 3250 for recovery attack, we note that we need almost 8 times more samples in the recovery attack than that in the distinguishing attack, whereas near success rate in $[0.97, 0.98]$, it becomes almost 1.3 to 1.35.

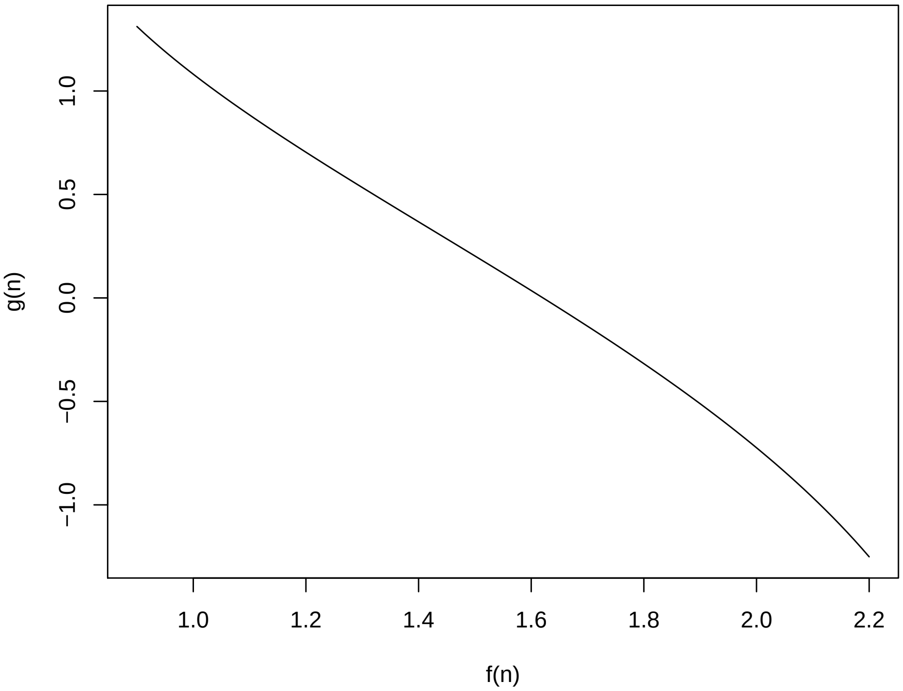

Fig. 3. Graph of $f(n)$ vs. $g(n)$

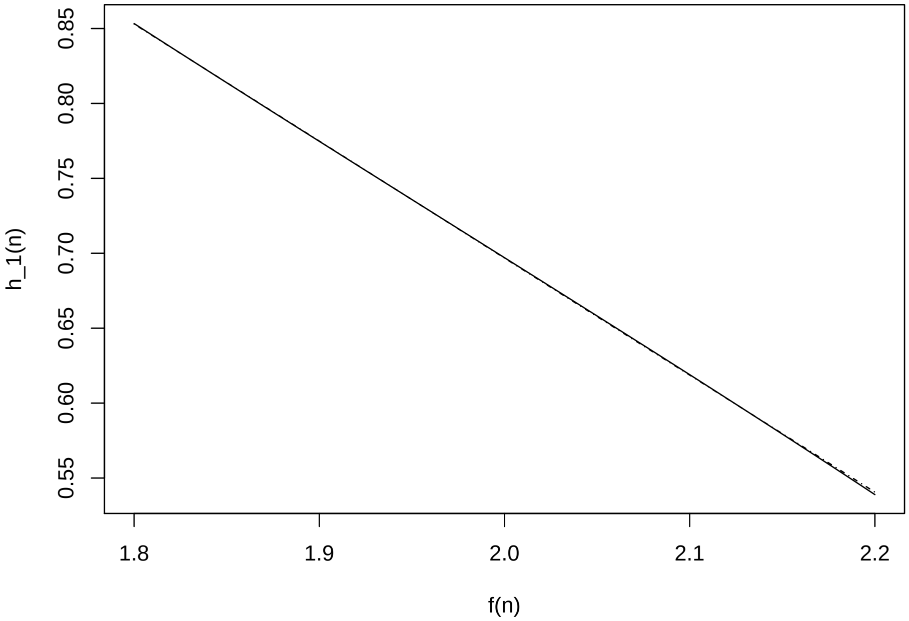

Fig. 4. Graph of $h_1(n)$ and its estimator linear function

We would now be interested in estimating the *Multiplier* function in terms of some known handy function. It is very interesting to note that we find empirically $\ln(n)$ and $\ln(\ln(Multiplier(n)))$ to be highly linearly related. We define two functions f and g on the set of natural numbers as follows:

$$f(n) := \ln\left(\frac{n}{1000}\right);\ \ g(n) := \ln(\ln(Multiplier(n))).$$

The graph of $f(n)$ vs $g(n)$ in *Fig.3* shows empirically high linear relationship.

Fig.3. shows the function $g(n)$ is slightly convex in the region $(0.9, 1.4)$ and slightly concave in the region $(1.8, 2.2)$ with respect to $f(n)$. In the middle region it is almost linear. So, we try to approximate the above relationship i.e. the function $g(n)$ in these three regions separately.

For the region $(1.8, 2.2)$ we empirically find that $h_1(n) := exp\left(\frac{g(n)}{2}\right)$ is almost linear w.r.t. $f(n)$. We try to estimate $h_1(n)$ by a linear function of $f(n)$ by minimizing the distance over the range $(1.8, 2.2)$(where the distance between two integrable functions f and g over the range (a, b) is defined as $\int_a^b (f(x) - g(x))^2\, dx$.) The function $h_1(n)$ and its estimate looks like in *Fig.4* where the estimating linear function $h'_1(n) := 2.249515 - 0.7759676 f(n)$ is in dotted line.

So, we get

$$g(n) \approx 2(\ln(2.249515 - 0.7759676 f(n));$$

$$for\ \ 1.8 \leq f(n) \leq 2.2\ \ i.e.\ 6000 \leq n \leq 9000.$$

For the region $(1.4, 1.8)$ we estimate the function $g(n)$ itself by linear functions by same method as described above. The function and its estimating line $g'(n) := 2.758714 - 1.703975 f(n)$ looks like *Fig.5*. So, we get

$$g(n) \approx 2.758714 - 1.703975 f(n));$$

$$for\ \ 1.4 \leq f(n) \leq 1.8\ \ i.e.\ 4000 \leq n \leq 6000.$$

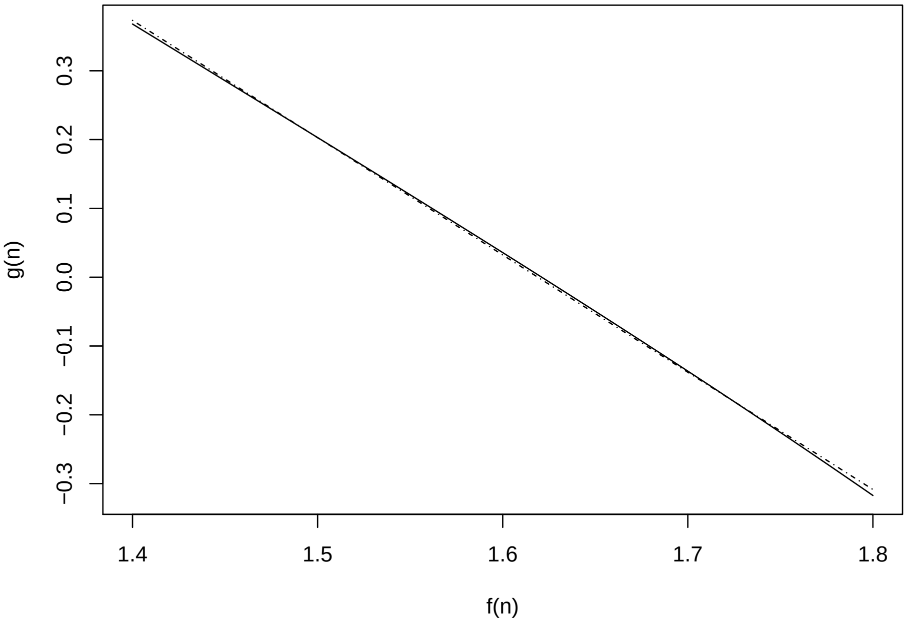

Fig. 5. Graph of $g(n)$ and its estimator linear function (in dotted line)

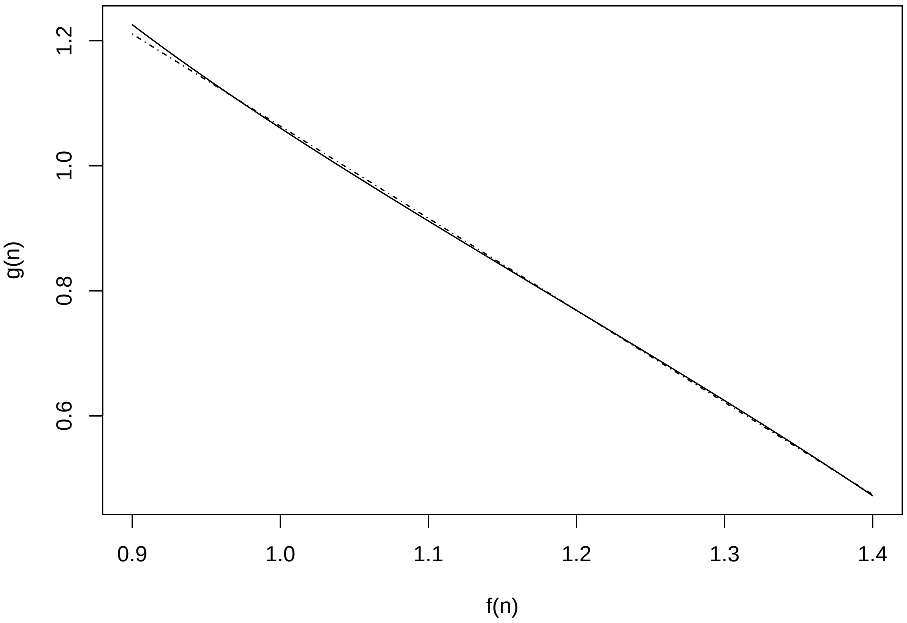

Fig. 6. Graph of $h_3(n)$ and its estimator linear function (in dotted line)

For the region $(0.9, 1.4)$ we find empirically the function $h_3(n) := (g(n))^{\frac{3}{4}}$ to be highly linearly related with $f(n)$. As previous we estimate by linear function by minimizing the distance and the estimating line $h'_3(n) := 2.393338 - 1.336852 f(n)$ looks like in *Fig.6*.

So, we get

$$g(n) \approx (2.393338 - 1.336852 f(n))^{\frac{4}{3}};$$

$$for \;\; 0.9 \leq f(n) \leq 1.4 \;\; i.e.\; 2450 \leq n \leq 4000.$$

VI. Conclusion

In this paper, we review different approaches towards estimating the data complexity of distinguishing attacks on stream ciphers and analyze their inter-relationships and applicable scenarios. We also formally analyze the data complexity of message recovery attack that exploits a distinguisher and show that in practice there is a significant gap between the two complexities. This gap turns out to be a function of the number of samples of the distinguishing attack. We perform a case study on RC4 stream cipher to demonstrate how these two complexities are related.

References

[1] I. Mantin and A. Shamir, “A practical attack on broadcast RC4,” in *Fast Software Encryption, 8th International Workshop, FSE 2001 Yokohama, Japan, April 2-4, 2001, Revised Papers*, ser. Lecture Notes in Computer Science, M. Matsui, Ed., vol. 2355. Springer, 2001, pp. 152–164. [Online]. Available: http://dx.doi.org/10.1007/3-540-45473-X_13

[2] S. S. Gupta, S. Maitra, G. Paul, and S. Sarkar, “(non-)random sequences from (non-)random permutations - analysis of RC4 stream cipher,” *J. Cryptology*, vol. 27, no. 1, pp. 67–108, 2014. [Online]. Available: http://dx.doi.org/10.1007/s00145-012-9138-1

[3] N. J. AlFardan, D. J. Bernstein, K. G. Paterson, B. Poettering, and J. C. N. Schuldt, “On the security of RC4 in TLS,” in *Proceedings of the 22th USENIX Security Symposium, Washington, DC, USA, August 14-16, 2013*, S. T. King, Ed. USENIX Association, 2013, pp. 305–320. [Online]. Available: https://www.usenix.org/conference/usenixsecurity13/technical-sessions/paper/alFardan

[4] R. Basu, S. Ganguly, S. Maitra, and G. Paul, “A complete characterization of the evolution of RC4 pseudo random generation algorithm,” *J. Mathematical Cryptology*, vol. 2, no. 3, pp. 257–289, 2008. [Online]. Available: http://dx.doi.org/10.1515/JMC.2008.012

[5] I. Mantin, “Predicting and distinguishing attacks on RC4 keystream generator,” in *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, ser. Lecture Notes in Computer Science, R. Cramer, Ed., vol. 3494. Springer, 2005, pp. 491–506. [Online]. Available: http://dx.doi.org/10.1007/11426639_29

[6] S. Kullback and R. A. Leibler, “On information and sufficiency,” *Annals of Mathematical Statistics*, vol. 22, pp. 49–86, 1951.

[7] S. R. Fluhrer and D. A. McGrew, “Statistical analysis of the alleged RC4 keystream generator,” in *Fast Software Encryption, 7th International Workshop, FSE 2000, New York, NY, USA, April 10-12, 2000, Proceedings*, ser. Lecture Notes in Computer Science, B. Schneier, Ed., vol. 1978. Springer, 2000, pp. 19–30. [Online]. Available: http://dx.doi.org/10.1007/3-540-44706-7_2

[8] R. E. Blahut, *Principles and Practice of Information Theory*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1987.

[9] J. Neyman and E. S. Pearson, “On the problem of the most efficient tests of statistical hypotheses,” *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 231, pp. pp. 289–337, 1933. [Online]. Available: http://www.jstor.org/stable/91247

[10] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006 (page 383).

[11] P. Stankovski, S. Ruj, M. Hell, and T. Johansson, “Improved distinguishers for HC-128,” *Des. Codes Cryptography*, vol. 63, no. 2, pp. 225–240, 2012. [Online]. Available: http://dx.doi.org/10.1007/s10623-011-9550-9

[12] H. Wu, “The stream cipher HC-128,” in *New Stream Cipher Designs - The eSTREAM Finalists*, ser. Lecture Notes in Computer Science, M. J. B. Robshaw and O. Billet, Eds. Springer, 2008, vol. 4986, pp. 39–47. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-68351-3_4

[13] T. Baignères, P. Junod, and S. Vaudenay, “How far can we go beyond linear cryptanalysis?” in *Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5-9, 2004, Proceedings*, ser. Lecture Notes in Computer Science, P. J. Lee, Ed., vol. 3329. Springer, 2004, pp. 432–450. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-30539-2_31

[14] J. Aumasson, S. Fischer, S. Khazaei, W. Meier, and C. Rechberger, “New features of latin dances: Analysis of salsa, chacha, and rumba,” in *Fast Software Encryption, 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers*, ser. Lecture Notes in Computer Science, K. Nyberg, Ed., vol. 5086. Springer, 2008, pp. 470–488. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-71039-4_30

[15] S. Samajder and P. Sarkar, “Another look at normal approximations in cryptanalysis,” *IACR Cryptology ePrint Archive*, vol. 2015, p. 679, 2015. [Online]. Available: http://eprint.iacr.org/2015/679

[16] ——, “Rigorous upper bounds on data complexities of block cipher cryptanalysis,” *IACR Cryptology ePrint Archive*, vol. 2015, p. 916, 2015. [Online]. Available: http://eprint.iacr.org/2015/916

[17] C. Garman, K. G. Paterson, and T. V. der Merwe, “Attacks only get better: Password recovery attacks against RC4 in TLS,” in *24th USENIX Security Symposium, USENIX Security 15, Washington, D.C., USA, August 12-14, 2015.*, J. Jung and T. Holz, Eds. USENIX Association, 2015, pp. 113–128. [Online]. Available: https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/garman

[18] G. Hardy, J. Littlewood, and G. Pólya, *Inequalities*, ser. Cambridge Mathematical Library. Cambridge University Press, 1952. [Online]. Available: https://books.google.com.sg/books?id=t1RCSP8YKt8C

[19] R. Christensen, *Plane Answers to Complex Questions - The Theory of Linear Models*, ser. Springer Texts in Statistics. Springer, 2011. [Online]. Available: https://books.google.co.in/books?id=8EunecOyv-EC&hl=en