Accepted by IEEE Transactions on Information and Forensics & Security, 2022 DOI: 10.1109/TIFS.2022.3176185

New Observations on Zipf's Law in Passwords

Zhenduo Hou, Ding Wang

***Abstract*—As password distribution lays the foundation for various password research, accurately characterizing it receives considerable attention. At IEEE TIFS'17, Wang et al. proposed the CDF-Zipf distribution model with the golden-section-search (GSS) fitting method to find the optimal parameters. Their model has been adopted by over 120 password-related studies. In this paper, we address their remaining, fundamental goodness-of-fit issue of password distribution in a *principled approach*. First, we prove that the confidence level of the state-of-the-art Monte Carlo approach (MCA, for the goodness-of-fit test) converges asymptotically to 0. By experimenting on 228.92 million real-world passwords, we confirm Wang et al.'s conjecture on the effect of sample size that minor deviations would lead to statistical significance for large-scale datasets. We propose both *absolute* and *relative* deviation metrics, and find that 1% random deviations in both metrics suffice to reject CDF-Zipf. Second, we attempt to reduce the non-negligible gap between the empirical and fitted distributions (with the maximum deviation of cumulative distribution function (CDF) being 1.91% on average). We explore eight alternative distribution models in two coordinate systems, and find that three models are more accurate than CDF-Zipf, but none can pass MCA. Particularly, we reveal that stretched-exponential, a variant of CDF-Zipf, can on average reduce the maximum CDF deviation from 1.91% to 1.25%. Third, to replace MCA, we introduce a new goodness-of-fit measure based on log-likelihoods. We find that stretched-exponential constantly has a larger log-likelihood than its counterparts. In all, stretched-exponential fits passwords better and further supports Zipf's law in passwords.**

***Index Terms*—Password distribution, Zipf's law, Goodness-of-fit measure, Monte Carlo approach, Stretched-exponential.**

I. Introduction

Textual passwords remain to be the mainstream form of Internet-based authentication [15]. Although the security pitfalls were revealed four decades ago [35], and various alternatives have been proposed (e.g., biometric [32], and multi-factor authentication [48]), passwords are still widely used in authentication due to the remarkable simplicity and low deployment cost. Therefore, passwords are likely to persist in the foreseeable future [15], [52], [56].

Despite their ubiquity, passwords are confronted with a difficult challenge [39], [62], [63]: Truly random passwords are hard to memorize, while most human-chosen passwords are highly predictable. In practice, besides popular passwords (e.g., `123456` and `qwerty` [60]), ordinary users are prone to use personally identifiable information (e.g., name and birthday [53], [56], [57]) to construct their passwords, and $58\%\sim79\%$ of investigated users reuse (or slightly modify) passwords across sites [21], [39], [55], [59]. Such vulnerable behaviors make password distribution nonuniform and highly skewed. However, hundreds of studies (e.g., [2], [18], [30], [49]) assume that "passwords are uniformly distributed". This unrealistic assumption often leads to orders of magnitude underestimates or overestimates of the security of password-related studies (e.g., cryptographic protocols [16], [58], encryption schemes [2], [22], and hash functions [13]). These facts underline the imperative necessity to characterize the skewed password distribution accurately.

Since human language follows Zipf's law [64], Malone and Maher [31] made the first attempt to characterize human-chosen passwords with the Zipf distribution. After conducting experiments with the probability density functions (PDFs) of four password datasets (three are with sizes smaller than 0.1 million), they concluded that these datasets are "unlikely to be Zipf distributed". In 2012, Bonneau [14] also attempted to fit PDFs of password datasets with Zipf's law, and reached a similar conclusion to Malone and Maher [31]. To figure out what is the distribution that passwords follow, in 2017, Wang et al. [54] investigated whether the cumulative distribution function (CDF, which is the summation of PDFs) of a password dataset follows the Zipf's law, and proposed the CDF-Zipf distribution model. They used the golden-section-search (GSS) fitting method (a numerical optimization method, see Sec. II-C) to find the optimal parameters characterizing the CDF (rather than PDF) of a password distribution, and experimented with 14 datasets of sizes ranging from 30 hundred to 32 million. Extensive experiments showed that their model not only provides the smallest Kolmogorov-Smirnov (KS) statistic (which is the maximum over the CDF deviation between the empirical and fitted distributions, see Sec. II-B), but also can cover the entire dataset. So far, their model has been adopted by over 120 password-related studies, such as password encryption [27], policy [61], guessing [45] and cryptographic protocols [42].[1]

The KS statistic between the empirical and fitted CDF distributions is attributed to two types of deviations [17], [20], [40]. Type-1 deviation comes from statistical randomness, which is inherent and *cannot be reduced by optimization or using more accurate models*. Type-2 deviation comes from the use of an inappropriate distribution model, and *can be reduced or even eliminated by using a more accurate model [20], [40]*. If the KS statistic mainly comes from type-1 deviation and the model is accurate, passwords can be generated from (i.e., follow) the fitted distribution with high confidence level [17],

Manuscript received Nov. 16, 2021; revised Apr. 22, 2022; accepted May 10, 2022. This research was supported by the National Natural Science Foundation of China under Grant No. 62172240 and the Natural Science Foundation of Tianjin, China under Grant No. 21JCZDJC00190. The corresponding author is Ding Wang.

Zhenduo Hou is with School of Mathematical Sciences, Peking University, Beijing 100871, China. (e-mail: joehou13@pku.edu.cn).

Ding Wang is with College of Cyber Science, Nankai University, Tianjin 300350, China, and with Tianjin Key Laboratory of Network and Data Security Technology, Nankai University, Tianjin 300350, China (e-mail: wangding@nankai.edu.cn).

[1]The full list of 124 studies can be found at https://bit.ly/3OhODAO .

[20], [40]. Otherwise, it is possible to adopt a more appropriate model to reduce or even eliminate type-2 deviation.

To evaluate whether passwords follow a chosen distribution, we need an independent evaluation named the goodness-of-fit test to distinguish both types of deviations [17], [20], [40]. If type-2 deviation is non-negligible, the chosen distribution model is likely not optimal, and can be further improved [17], [20], [40]. In 2017, Wang et al. [54] used the Monte Carlo approach (MCA) recommended by Clauset et al. [20] to do this test for their CDF-Zipf distribution model. The basic idea of MCA is to first generate a number of (e.g., 10^4) synthetic datasets by using the same distribution parameters fitted from the empirical dataset, and then calculate the proportion (i.e., p-value, see Sec. III-A) of type-1 deviations that are larger than the KS statistic between the empirical and synthetic datasets. If this proportion is large (e.g., $>90\%$), type-1 deviation dominates; Otherwise, type-2 deviation is non-negligible, suggesting the inappropriateness of the examined model.

In their MCA experiment, Wang et al. [54] found the above proportion very small (i.e., $<10^{-4}$), but conjectured that this phenomenon was due to the effect of sample size: "Given a sufficiently large sample, extremely small and non-notable differences can be found to be statistically significant, and statistical significance says nothing about the practical significance of a difference [43]." Despite such a claim, Wang et al. [54] left the proof *unfinished* as future work. Hence, a natural question arises: *Given the CDF-Zipf distribution model, to what extent is MCA an inappropriate goodness-of-fit measure for large-scale datasets?* To the best of our knowledge, no prior work has tackled it. Ignoring this crucial question may result in incorrect claims of password distribution.

Since the CDF-Zipf distribution model cannot pass MCA, the second question arises: *Are there comparable or even more accurate distribution models than CDF-Zipf? In particular, Whether these models can pass MCA?* If an alternative model can achieve this, passwords are better characterized with it than CDF-Zipf (i.e., its KS statistic will be smaller). Otherwise, if all of such more accurate alternative models (most of them are commonly used) still cannot pass MCA, it is natural to *cast doubt* on the intrinsic effectiveness of MCA. Under this doubt, a more appropriate goodness-of-fit measure is necessary. This raises the third question: *Is there a more appropriate goodness-of-fit measure than MCA for large-scale password datasets?* In particular, given various password distribution models with comparable accuracy, which distribution model are passwords more likely (and reasonable) to follow? In all, this work, *for the first time*, pays attention to the above three crucial research questions.

A. Our contributions

In this paper, we make the following contributions.

- **Quantitative analysis of MCA**. We provide both rigorous mathematical proofs and extensive experiments to validate Wang et al.'s [54] folklore of the effect of the sample size. *For the first time*, we prove that under the CDF-Zipf distribution model, type-1 deviation converges asymptotically to 0. Experiment results on subsets of eight large-scale password datasets substantiate the mathematical proofs: When the size of the subsampled dataset ≥ 0.25 million, MCA rejects CDF-Zipf. Second, we propose the *absolute* and *relative* deviation metrics to simulate real-world password deviation. We prove that the maximum of the KS statistic increases *monotonically* with the deviation. With this theoretical guarantee, we implement extensive experiments, and find that 1% random deviations in both metrics is enough for MCA to reject the CDF-Zipf distribution. This shows that MCA is too sensitive to be effective as a goodness-of-fit measure.
- **Alternative password distributions**. We investigate four distribution models in the rank-frequency coordinate system, and four in the frequency-frequency coordinate system, a total of eight alternative models to find if there is a more accurate model than CDF-Zipf. We fit these distribution models with Wang et al.'s [54] GSS fitting method. We find that lognormal in both systems and stretched-exponential in the rank-frequency system are comparably accurate with CDF-Zipf, with the maximum CDF deviation ranging from 1.25% to 1.48% on average. We also revisit MCA for these alternative models. MCA rejects *all* of them regardless of accuracy when the dataset size is large (e.g., ≥ 1 million), but accepts multiple distribution models when the size is small, confirming the ineffectiveness of MCA.
- **A new goodness-of-fit measure**. We introduce the log-likelihood ratio test (LRT) to find which distributions is more likely for passwords to follow. In particular, we investigate the CDF-Zipf and three other comparably accurate alternative models (i.e., lognormal in two coordinate systems and stretched-exponential in the rank-frequency coordinate system). We find that the stretched-exponential model, a variant of CDF-Zipf, has a significantly and constantly larger ($1.26\times10^7 \sim 6.15\times10^8$) log-likelihood ratio than the other three models. Besides, its maximum CDF deviation is $0.46\% \sim 2.49\%$ (avg. 1.25%), while that of CDF-Zipf is larger and is $0.50\% \sim 4.54\%$ (avg. 1.91%). In particular, on six out of eight datasets, stretched-exponential is more accurate than CDF-Zipf. Therefore, stretched-exponential can more accurately characterize password distribution. We also compare LRT with MCA, and show that LRT outperforms MCA in terms of minimizing statistical errors.
- **Some insights.** We obtain a number of insights, some expected and some surprising, from our theories and experiments. To our surprise, we find that some distributions (other than CDF-Zipf) are hard to optimize with GSS, which may come from unexpected singularities in transcendental functions in the CDF expressions. As expected, the larger the dataset, the smaller the statistical randomness for passwords.

II. Preliminaries

In this section, first, we describe the eight large-scale password datasets. Second, we introduce the fitting methodologies. Finally, we elaborate on the goodness-of-fit test.

A. Datasets and ethics

We use eight large-scale datasets, a total of 228.92 million real-world passwords in this study. As shown in Table I, all datasets were hacked or released from 2009 to 2020, and have been publicly available for some time. The justification for

TABLE I: Basic information of password datasets.

Dataset	Service type	Language	When leaked	How leaked	Total passwords
Yahoo†	Portal	English	Aug., 2013	Released	69,301,337
Dodonew	E-commerce	Chinese	Dec., 2011	Hacked	16,258,891
000webhost	Web Host	English	Oct., 2015	Hacked	15,251,073
Rockyou	Social Forum	English	Dec., 2009	Hacked	32,603,388
Tianya	Social Forum	Chinese	Dec., 2011	Hacked	30,816,592
Chegg	Education	English	Apr., 2018	Hacked	38,997,234
Mathway	Education	English	Jan., 2020	Hacked	16,524,045
Wishbone	Chatting	English	Jan., 2020	Hacked	9,171,560

† Yahoo was first collected by Bonneau [14] and later published by Blocki et al. [11] using differential private techniques. The data is available at https://figshare.com/articles/Yahoo_Password_Frequency_Corpus/2057937.

using these datasets lies in four folds: (1) Recently breached password datasets can represent the latest trend in password evolution; (2) Early breached password datasets have been widely used in other work (e.g., [12], [54], [57]), and can make results of this work reproducible; (3) Practical cracking studies suggest that passwords evolve slowly [14], so properties from early and recently breached datasets should be similar; (4) These datasets are of various backgrounds (e.g., languages and service types), and can reflect real-world passwords.

We are also fully aware of the ethics of this study. Although these datasets have been publicly available and widely used in previous research (e.g., [12], [54], [57]), they contain private data (e.g., names and email addresses). In this study, we only use aggregated statistical information like frequency and keep others confidential, so using these datasets will not bring extra risks to users. Finally, our research aims to benefit both the academic and industrial communities.

For any given dataset $\mathcal{DS}$ with $|\mathcal{DS}|$ passwords in total, we rank the N unique passwords from the most frequent to the least, and denote the frequency of the i-th unique password in the real-world dataset RealSet as $\hat{f}_i$: For passwords $PW_1, PW_2, ...PW_N$, it holds that $\hat{p}_1 \geq \hat{p}_2 \geq ... \geq \hat{p}_N$, where $\hat{p}_i = \hat{f}_i/|\mathcal{DS}|$ is the probability density function (PDF), and the cumulative distribution function (CDF) of RealSet is the summation of PDFs, i.e., $\hat{P}_r = \sum_{i=1}^{r} \hat{p}_i$. Similarly, the CDF of the theoretical dataset TheoSet is $P_r = \sum_{i=1}^{r} p_i$.

B. Kolmogorov-Smirnov statistic

The Kolmogorov-Smirnov (KS) statistic [20] is the maximum of the distance between two cumulative distribution functions (CDFs), i.e., the maximum over the absolute value of differences between the two CDFs. Thus, the KS statistic between TheoSet and RealSet is

$$D_{KS} = \max_{1 \leq r \leq N} |P_r - \hat{P}_r|, \tag{1}$$

where $D_{KS} \in [0, 1]$. Since a smaller D_{KS} means a more accurate characterization, the goal is to minimize D_{KS} by searching the parameters (e.g., C and s for CDF-Zipf). This KS statistic measurement is widely used in nonparametric fittings [20], [38], so our usage is justified.

C. Golden-section-search fitting method

Wang et al. [54] introduced the golden-section-search (GSS) fitting method to fit the password dataset. The key idea is to find a synthetic theoretical dataset TheoSet characterized by parameters with the minimal KS statistic with the real-world dataset RealSet. For instance, under the CDF-Zipf distribution where the CDF is $P_r = Cr^s$, minimizing the KS statistic is

$$\min_{C,s} D_{KS}. \tag{2}$$

The details of GSS are presented in Alg. 4 in Appendix A. With this approach, the CDF of passwords depends *numerically* on the frequencies of passwords in TheoSet. Thus, it is recommended to run Alg. 4 multiple times (e.g., 100) and take an average to minimize random fluctuations.

D. Goodness-of-fit test

Since the CDF-Zipf distribution model can characterize password distribution with relatively high accuracy, a natural question arises of *whether passwords truly follow it*. In statistics, the goodness-of-fit test is conducted to answer this question by examining whether the distribution model holds with high statistical confidence [17], [20], [40].

Typically, the goodness-of-fit test is a hypothesis testing method and goes as follows. The claim that passwords follow the $\mathcal{X}$ distribution (e.g., CDF-Zipf) is treated as the null hypothesis H_0. Similarly, the claim that passwords do not follow $\mathcal{X}$ (or follow some other distributions) is treated as the alternative hypothesis H_1. A straightforward idea to know whether H_0 or H_1 holds is to distinguish the source of the KS statistic. If the KS statistic mainly comes from statistical randomness (i.e., type-1 deviation), the chosen model holds with a high confidence level and is unlikely to be further optimized; Otherwise, if the KS statistic mainly comes from the use of a distribution model (i.e., type-2 deviation), it is likely to adopt alternative models to reduce the KS statistic. The idea is used by the Monte Carlo approach (MCA) to determine whether a distribution model holds with high statistical confidence.

If the above straightforward MCA falls short of justifying CDF-Zipf, passwords may follow other distributions that are roughly linear under the log-log scale. Ideally, a good distribution model $\mathcal{X}$ (not CDF-Zipf) should both (1) provide the smaller KS statistic and (2) be supported by the goodness-of-fit test results. Otherwise, a new method is needed to do the goodness-of-fit test for password datasets.

Two types of statistical errors exist for hypothesis tests. Type-1 statistical error is the probability of mistaken rejection of true H_0, which is rejecting $\mathcal{X}$ (e.g., CDF-Zipf) when passwords truly follow it; Type-2 statistical error is the probability of mistaken acceptance of false H_0, which is accepting $\mathcal{X}$ when passwords do *not* follow it. In this paper, we will discuss them when evaluating goodness-of-fit measures MCA (see Sec. IV-C) and log-likelihood ratio test (LRT, see Sec. IV-D).

III. Analysis of Underlying Issues of MCA

In this section, we first use rigorous mathematical proofs and extensive experiments to investigate what dataset size can make MCA reject CDF-Zipf. Then, we introduce two metrics to simulate the real-world deviation, and conduct extensive experiments to find the thresholds rejecting CDF-Zipf.

A. Revisit of MCA process

As mentioned in Sec. II-B, we use the KS statistic to measure the CDF deviation. First, we introduce the idea of MCA goodness-of-fit measure. Since the theoretical dataset TheoSet is characterized by the distribution parameters (e.g., C and s for CDF-Zipf), the KS statistic of fitting TheoSet (denoted as D'_{KS}) can be seen as type-1 deviation (i.e., statistical

TABLE II: Parameters, orders of magnitude of deviations, and p-values calculated using Monte Carlo Approach (MCA).†

Dataset	C	$O(\sigma_C)$	s	$O(\sigma_s)$	D_{KS}	$O(\sigma_{D_{KS}})$	C'	$O(\sigma_{C'})$	s'	$O(\sigma_{s'})$	D'_{KS}	p-value
Yahoo	0.033148	10^{-4}	0.180907	10^{-4}	0.040775	10^{-5}	0.033207	10^{-17}	0.180752	10^{-16}	0.000298∼0.000697	$<\mathbf{10^{-4}}$
Dodonew	0.019255	10^{-5}	0.211921	10^{-5}	0.004979	10^{-4}	0.019459	10^{-4}	0.211798	10^{-4}	0.000101∼0.000186	$<\mathbf{10^{-4}}$
000webhost	0.005738	10^{-5}	0.282561	10^{-4}	0.005022	10^{-4}	0.005665	10^{-5}	0.283099	10^{-4}	0.000506∼0.001544	$<\mathbf{10^{-4}}$
Rockyou	0.038208	10^{-5}	0.185939	10^{-16}	0.045357	10^{-4}	0.037543	10^{-6}	0.186960	10^{-15}	0.000310∼0.000693	$<\mathbf{10^{-4}}$
Tianya	0.062337	10^{-4}	0.155266	10^{-4}	0.022925	10^{-5}	0.062095	10^{-4}	0.155464	10^{-4}	0.000200∼0.002294	$<\mathbf{10^{-4}}$
Chegg	0.008297	10^{-4}	0.234966	10^{-4}	0.008617	10^{-4}	0.008186	10^{-5}	0.236053	10^{-4}	0.000043∼0.001214	$<\mathbf{10^{-4}}$
Mathway	0.010541	10^{-5}	0.245255	10^{-4}	0.011059	10^{-4}	0.010548	10^{-5}	0.245325	10^{-4}	0.000413∼0.001594	$<\mathbf{10^{-4}}$
Wishbone	0.017144	10^{-4}	0.230503	10^{-4}	0.014775	10^{-4}	0.017125	10^{-4}	0.230626	10^{-4}	0.000192∼0.002275	$<\mathbf{10^{-4}}$

† $O(\sigma)$ denotes the orders of magnitude of the standard deviations. D_{KS} is the Kolmogorov-Smirnov (KS) statistic resulting from fitting the *real-world* dataset RealSet with C and s, and D'_{KS} is the KS statistic (denoted in range) resulting from fitting the *theoretical* dataset TheoSet in the Monte Carlo approach (MCA). The results show that (1) $D_{KS}>D'_{KS}$; (2) p-values $<10^{-4}$), and (3) $C' \approx C$ and $s' \approx s$ hold for all datasets.

randomness). The CDF deviation (denoted as D_{KS}) between the real-world dataset RealSet and the theoretical dataset TheoSet contains both type-1 and type-2 deviations. Thus, if a significant proportion of D'_{KS} has $D'_{KS}>D_{KS}$, CDF deviations mainly come from type-1 deviation, and passwords follow the distribution model with a high confidence level.

In more detail, we do MCA as follows: (1) Use the golden-section-search (GSS) to fit the real-world dataset RealSet with the CDF-Zipf distribution model to obtain the KS statistic D_{KS}; (2) Use the same distribution parameters characterizing RealSet to generate J_0 theoretical datasets, i.e., $\mathsf{TheoSet}_1, \cdots, \mathsf{TheoSet}_{J_0}$; (3) Use GSS to fit each $\mathsf{TheoSet}_j$ ($1 \leq j \leq J_0$) individually and independently, and calculate the corresponding D'_{KS_j} for each $\mathsf{TheoSet}_j$; (4) Calculate the proportion of $D'_{KS}>D_{KS}$ as the p-value (the confidence level of the distribution model); We add one in both denominator and numerator to make the p-value smooth [23]; (5) We set the p-value threshold to be 0.01: If the p-value$>$0.01, the null hypothesis H_0 that passwords follow CDF-Zipf should be accepted; Otherwise, it should be rejected. The process of MCA with CDF-Zipf is shown in Alg. 1.

Algorithm 1: Monte Carlo approach (MCA) on CDF-Zipf.†

Input : The real-world dataset RealSet.
Output: The confidence level p-value.

1 **begin**
2 $(C, s, D_{KS}) = \mathrm{GSS}(\mathsf{RealSet})$;
3 **for** $j = 1$ **to** J_0 **do**
4 $\mathsf{TheoSet_j} = \mathrm{THEOGEN}(C, s, |\mathcal{DS}|)$; /* Generate J_0 theoretical datasets, $|\mathcal{DS}|$ is the dataset size. */
5 $(C'_j, s'_j, D'_{KS_j}) = \mathrm{GSS}(\mathsf{TheoSet_j})$; /* Fit J_0 theoretical datasets with the same fitting method of the real-world dataset. */
6 p-value = $(\#\{D'_{KS_j} | D'_{KS_j} > D_{KS}, 1 \leq j \leq J_0\} + 1)/(J_0 + 1)$; /* Smooth the p-value. */
7 **if** *p-value* > 0.01 **then**
8 Accept the CDF-Zipf distribution model;
9 **else**
10 Reject the CDF-Zipf distribution model;
11 **Output**: p-value.

† GSS is the golden-section-search fitting method (see Alg. 4 in Appendix A) used by Wang et al. [54]. THEOGEN characterizes the theoretical dataset TheoSet with the conversion method in [20] (see Alg. 3 in Appendix A).

We now explain why we choose the p-value threshold to be 0.01. First, like the threshold 0.05 [25], this 0.01 has also been widely used in various research fields (e.g., epidemiology [26], psychology [51], and cyber security [19]). Second, the above way of defining p-value indicates that the smaller the threshold, the harder for MCA to reject CDF-Zipf. Hence, using 0.01 can make our analysis more rigorous with a smaller type-1 statistical error: When p-value$<$0.382 and H_0 and H_1 hold with approximately equal prior probability (i.e., $P(H_0) \approx P(H_1)$, usually true suggested by Berger et al. [9]), type-1 statistical error e_1 (which is a function of p-value) is

$$e_1 = (1 + (2\sqrt{p\text{-value}})^{-1})^{-1} \quad (3)$$

based on [46]. Thus, type-1 statistical error can be reduced from 30.90% to 16.67% by setting the p-value threshold to be 0.01 rather than 0.05, making our results more reliable. Furthermore, to ensure the accuracy of the p-value, we take J_0=10,000 (the number of generated TheoSets), so the random fluctuation ϵ_p of the p-value is $<$0.005 ($\epsilon_p=\frac{1}{\sqrt{4J_0}}$ [20]). This also indicates that the smallest p-value is $\frac{1}{10{,}001}<10^{-4}$ (with smoothing, see Line 6 of Alg. 1), when all type-1 deviations (statistical randomness) is smaller than the KS statistic (i.e., $D'_{KS}<D_{KS}$). Third, we also do not mean to exclude the possibility of other thresholds. As with prior research (e.g., [33], [37], [47], [50]), we provide details including p-value, type-1 deviation, and KS statistic data calculated using MCA (see Tables II and VIII), so practitioners can set their own p-value thresholds (e.g., 0.005 [7], [28], 0.05 [25], and 0.1 [20]) based on their needs. Our theories will also reveal (see Sec. III-B) that for sufficiently large (e.g., $\geq$ 1 million) datasets, the exact p-value threshold is *irrelevant* to the overall conclusion.

Table II shows the p-value and parameter results of fitting the eight large-scale password datasets with the CDF-Zipf distribution. We can see that: (1) D'_{KS} is at least one order of magnitude smaller than D_{KS}, so type-2 deviation invariably dominates the KS statistic; Hence, the null hypothesis H_0 that passwords follow the CDF-Zipf distribution is rejected; (2) The p-value is invariably $<10^{-4}$, *and MCA will not accept CDF-Zipf regardless of p-value thresholds (e.g., 0.01, 0.05, and 0.1);* (3) C and C', as well as s' and s values are very close, with differences $\Delta C = |C - C'|$ ranging from 10^{-7} to 10^{-4} and $\Delta s = |s - s'|$ ranging from 10^{-6} to 10^{-3}; In such a case, it is justified to treat $C' \approx C$ and $s' \approx s$ in MCA.

B. Relation between p-value and dataset size

We show how the p-value changes with the password dataset size $|\mathcal{DS}|$ using both rigorous mathematical proofs and extensive experiments. Since the p-value is the proportion of $D'_{KS}>D_{KS}$, we focus on how the maximum of type-1 deviation $\max D'_{KS}$ changes with the dataset size $|\mathcal{DS}|$.

Theories. In the process of generating TheoSet, each password PW_i can be seen as a random Bernoulli variable with mean p_i^m and standard deviation $\sqrt{p_i^m(1-p_i^m)}$, where p_i^m is the *true* probability of PW_i determined by the distribution model (e.g., CDF-Zipf) [14], [54]. Hence, the frequency f_i of PW_i after $|\mathcal{DS}|$ sampling follows the binomial distribution with the mean $\mu_i = p_i^m|\mathcal{DS}|$ and standard deviation $\sigma_i = \sqrt{p_i^m(1-p_i^m)|\mathcal{DS}|}$. Since $1 - p_i^m < 1$, there is

TABLE III: p-values of subsets of password datasets calculated using Monte Carlo Approach (MCA).[†]

Dataset	Subset size	ΔC	Δs	D_{KS}	D'_{KS}	p-value	Dataset	Subset size	ΔC	Δs	D_{KS}	D'_{KS}	p-value
Yahoo	$0.05M$	0.000033	0.005277	0.004660	0.000320∼0.008640	**0.054**	Tianya	$0.05M$	0.000001	0.000191	0.008620	0.000480∼0.012180	0.005
	$0.1M$	0.000091	0.010370	0.005752	0.000300∼0.007180	0.006		$0.1M$	0.000134	0.000654	0.007910	0.000390∼0.008400	0.003
	$0.25M$	0.000071	0.015701	0.007834	0.000500∼0.003880	$<10^{-4}$		$0.25M$	0.000123	0.000468	0.008195	0.000500∼0.003780	$<10^{-4}$
	$0.5M$	0.000085	0.008823	0.010275	0.000324∼0.003286	$<10^{-4}$		$0.5M$	0.000143	0.000386	0.008951	0.000388∼0.003956	$<10^{-4}$
	$1M$	0.000002	0.004011	0.011862	0.000214∼0.002057	$<10^{-4}$		$1M$	0.000024	0.000004	0.010125	0.000258∼0.003534	$<10^{-4}$
Dodonew	$0.05M$	0.000032	0.000095	0.006177	0.000500∼0.010240	**0.150**	Chegg	$0.05M$	0.000038	0.000652	0.001338	0.000260∼0.004440	**0.032**
	$0.1M$	0.000040	0.000575	0.005494	0.000310∼0.005640	0.006		$0.1M$	0.000001	0.000292	0.000987	0.000110∼0.003240	$<10^{-4}$
	$0.25M$	0.000015	0.000080	0.004864	0.000360∼0.002796	0.004		$0.25M$	0.000005	0.000034	0.000827	0.000148∼0.003420	0.003
	$0.5M$	0.000044	0.000110	0.005001	0.000248∼0.002970	$<10^{-4}$		$0.5M$	0.000005	0.000042	0.000689	0.000146∼0.002154	0.003
	$1M$	0.000004	0.000011	0.004946	0.000197∼0.002882	$<10^{-4}$		$1M$	0.000004	0.000060	0.000545	0.000102∼0.002059	$<10^{-4}$
000webhost	$0.05M$	0.000027	0.000214	0.004180	0.000200∼0.006060	0.007	Mathway	$0.05M$	0.000030	0.000273	0.003438	0.000240∼0.006980	**0.050**
	$0.1M$	0.000011	0.001858	0.004201	0.000220∼0.005480	0.003		$0.1M$	0.000053	0.000836	0.003122	0.000210∼0.006460	**0.020**
	$0.25M$	0.000001	0.000627	0.004242	0.000232∼0.002364	$<10^{-4}$		$0.25M$	0.000004	0.000017	0.003319	0.000196∼0.002448	0.002
	$0.5M$	0.000015	0.000357	0.004186	0.000126∼0.001766	$<10^{-4}$		$0.5M$	0.000002	0.000001	0.003655	0.000202∼0.002408	$<10^{-4}$
	$1M$	0.000003	0.000042	0.004232	0.000095∼0.002406	$<10^{-4}$		$1M$	0.000001	0.000027	0.004427	0.000219∼0.002110	$<10^{-4}$
Rockyou	$0.05M$	0.000004	0.000133	0.002823	0.000460∼0.013540	$<10^{-4}$	Wishbone	$0.05M$	0.008346	0.000024	0.002338	0.000380∼0.009300	**0.140**
	$0.1M$	0.000016	0.000155	0.002294	0.000510∼0.007280	$<10^{-4}$		$0.1M$	0.000020	0.000129	0.001735	0.000320∼0.006180	0.007
	$0.25M$	0.000029	0.000074	0.001621	0.000588∼0.004480	$<10^{-4}$		$0.25M$	0.000013	0.000077	0.001342	0.000276∼0.004892	$<10^{-4}$
	$0.5M$	0.000082	0.000229	0.001328	0.000386∼0.004058	$<10^{-4}$		$0.5M$	0.000015	0.000112	0.001099	0.000188∼0.003582	$<10^{-4}$
	$1M$	0.000008	0.000041	0.001131	0.000210∼0.003073	$<10^{-4}$		$1M$	0.000011	0.000064	0.000896	0.000159∼0.002873	$<10^{-4}$

[†] $\Delta C = |C - C'|$ and $\Delta s = |s - s'|$ are differences between C and C', as well as s and s'. D_{KS} is the KS statistic, and $1M = 10^6$, i.e., one million. The **bold** p-values are those $>\mathbf{0.01}$. Both ΔC and Δs are very small (10^{-7}∼10^{-3}, except two cases in Yahoo), so $C \approx C'$ and $s \approx s'$. Besides, only if the subset size is $\leq 0.1M$ can the p-value be $>\mathbf{0.01}$ to support the CDF-Zipf distribution model.

$\sigma_i < \sqrt{p_i^m |\mathcal{DS}|} = \sqrt{\mu_i}$. For popular passwords with $f_i \geq f_b$ (e.g., $f_b = 10$), using f_i to approximate μ_i is accurate because $\sigma_i/\mu_i < \sqrt{1/\mu_i}$. For unpopular passwords with $f_i < f_b$, exploratory experiments show $\sigma_i \to 0$. Based on these observations, we have the following theorem.

***Theorem** 1:* Suppose C and s of RealSet and C' and s' of TheoSet satisfy $C \approx C'$ and $s' \approx s$; For each password of $f_i \geq f_b$, the estimation error $\epsilon_i = f_i - \mu_i$ follows the normal distribution $N(0, \sigma_i^2)$, and for passwords of $f_i < f_b$, $\sigma_i = 0$. In this case, the maximum of type-1 deviation (i.e., statistical randomness) $\max D'_{KS}$ decreases as $|\mathcal{DS}|$ increases, and there is $\lim_{|\mathcal{DS}|\to\infty} D'_{KS} = 0$. As a consequence, p-value decreases as $|\mathcal{DS}|$ increases, and there is $\lim_{J_0\to\infty} p\text{-value} = 0$ (see the proof in Appendix B).

Discussion. To begin with, we show that conditions in Theorem 1 are realistic. First, as shown in Table II, $C' \approx C$ and $s' \approx s$ hold for all our eight large-scale password datasets, so we can treat C' and C, as well as s' and s statistically equivalent. This means that Theorem 1 holds regardless of the exact C and s values. Second, the normality assumption, i.e., $\epsilon_i \sim N(0, \sigma_i^2)$ is not only guaranteed by the central limit theorem, but also recommended by the NIST standard of statistical methods [1]. Third, preliminary results show that the actual σ_i is *smaller* than the theoretical maximum $\sqrt{\mu_i}$, and becomes very small (e.g., $<10^{-5}$) after the first $3,000$ unique passwords. Hence, it is reasonable to assume that $\sigma_i = 0$ for unpopular passwords with $f_i < f_b$.

We also discuss the implications of Theorem 1. It reveals that once the dataset size $|\mathcal{DS}|$ is sufficiently large (e.g., $\geq$ 1 million), type-1 deviation (i.e., statistical randomness) D'_{KS} will be close to 0. As a result, the KS statistic mainly comes from type-2 deviation, i.e., using the CDF-Zipf distribution model, and the p-value is invariably $<10^{-4}$ when J_0=10,000. *This means that for a sufficiently large dataset, whatever the exact p-value threshold is, MCA will reject CDF-Zipf, and type-2 statistical error (rejecting CDF-Zipf when passwords truly follow it) is unlikely to occur*. In addition, since $\lim_{|\mathcal{DS}|\to\infty} D'_{KS} = 0$ (and thus $D'_{KS} \ll D_{KS}$) holds for each generated TheoSet, a straightforward deduction is that $\lim_{J_0\to\infty} p\text{-value} = 0$. This demonstrates that using more TheoSets in MCA (i.e., doing a larger-scale MCA experiment) does *not* change the essence that $D'_{KS} \ll D_{KS}$, and CDF-Zipf will still be rejected as long as the dataset is large.

Empirical results. We experiment on subsets of our eight large-scale datasets to see how the p-value changes as the dataset size $|\mathcal{DS}|$ varies, and find the threshold making the p-value$<$0.01. By randomly sub-sampling datasets (without replacement) of sizes $0.05M$, $0.1M$, $0.25M$, $0.5M$, and $1M$ ($1M = 10^6$, i.e., one million), we calculate the p-values and parameters of these subsets, and show the results in Table III.

Table III shows that: (1) For all subsets, the difference ΔC between C and C', as well as the difference Δs between s and s', are about 10^{-7}∼10^{-3} (with only two exceptions in Yahoo), so Theorem 1 is also practical for subsets; (2) Both the maximum and minimum of D'_{KS} *decrease* as the subset size increases, and the p-value decreases monotonically: When the subset size is $\geq 0.25M$, the p-value is $<$0.01; When the size is $\geq 1M$, the p-value is $<10^{-4}$ and $D'_{KS} < D_{KS}$; *In addition, even if the p-value threshold is <0.01 (e.g., 0.005) and its convergence rates are different across datasets, MCA will always reject CDF-Zipf when the dataset size exceeds* $1M$; These findings are consistent with Theorem 1; (3) When the datasets size is $<0.1M$, the p-value is $>$0.01 (except Tianya), consistent with Wang et al.'s observation [54] that p-values of small datasets (with size about 10^4∼10^5) can pass MCA (e.g., Myspace of $0.04M$ passwords [54]). All this substantiates the correctness of Theorem 1, and reveals the threshold of $0.25M$ above which MCA rejects the CDF-Zipf distribution model.

We now explain Table III in the view of statistical errors. Since p-values$<$0.01 for subsets $\geq 0.25M$, and p-values$<10^{-4}$ for subsets $\geq 1M$, the chance of accepting CDF-Zipf is low in MCA. Thus, for large-scale datasets $\geq 0.25M$, we only need to consider type-1 statistical error, which is 16.67% when p-value$<$0.01 and 1.96% when p-value$<10^{-4}$. Similarly, when the dataset size is small (e.g., $\leq 0.1M$), we only consider type-2 statistical error. We will discuss these two types of statistical errors of MCA in Sec. IV-C in detail.

C. Stimulation of real-world frequency deviations

The above findings raise a natural question: *What degree of deviation can make MCA reject CDF-Zipf for a large-scale password dataset?* To answer this question, we first measure the frequency difference of a password (e.g., with rank i) in the real-world dataset RealSet and the theoretical dataset TheoSet; We then propose two metrics (i.e., absolute and relative deviation metrics) to simulate the real-world deviation.

TABLE IV: Point-wise deviations of the top-10 unique passwords.†

Rank	Yahoo		Dodonew		000webhost		Rockyou		Tianya		Chegg		Mathway		Wishbone	
	Password	$p\hat{d}v_i$	Password	$p\hat{d}v_i$	Password	$p\hat{d}v_i$	Password	$p\hat{d}v_i$	Password	$p\hat{d}v_i$	Password	$p\hat{d}v_i$	Password	$p\hat{d}v_i$	Password	$p\hat{d}v_i$
1	123456	-67.16%	123456	-25.52%	abc123	-71.15%	123456	-76.16%	123456	-35.96%	default‡	-68.47%	123456	-74.30%	123456	-62.94%
2	password	-51.37%	a123456	24.58%	123456a	-18.30%	12345	-53.21%	111111	41.25%	123456	3.91%	mathway	-14.07%	password	-12.81%
3	welcome	-69.23%	111111	57.33%	12qw23we	-5.15%	123456789	-29.90%	000000	36.77%	Chegg123	12.78%	123456789	24.10%	123456789	-7.55%
4	ninja	-62.27%	123456789	48.03%	123abc	13.58%	password	-28.28%	123456789	76.50%	password	25.56%	12345	58.26%	wishbone	-29.07%
5	abc123	-54.42%	a321654	50.71%	a123456	23.50%	iloveyou	-25.91%	123123	30.24%	Testing1	46.77%	password	64.92%	12345678910	-21.35%
6	123456789	-46.69%	123123	61.63%	123qwe	39.83%	princess	-41.83%	123321	-12.33%	testing	54.83%	abc123	7.68%	unicorn	-17.06%
7	12345678	-43.09%	5201314	77.92%	secret666	47.13%	1234567	-56.55%	5201314	-4.41%	Chegg1	29.79%	12345678	3.13%	1234567890	-23.65%
8	sunshine	-42.99%	123456a	84.73%	YfDbUfNjH10305070‡	60.20%	rockyou	-52.95%	12345678	4.20%	A1b2c3	13.92%	1234567890	4.90%	12345678	-16.19%
9	princess	-38.09%	0	47.92%	asd123	69.87%	12345678	-48.71%	666666	6.97%	default‡	24.89%	mathsucks	-0.67%	qwertyuiop	-13.68%
10	qwerty	-45.77%	000000	46.77%	qwerty123	82.33%	abc123	-54.55%	111222tianya	9.86%	010203Zaq	32.05%	qwerty	7.71%	1234567	-10.45%
%*	1.89%		3.28%		0.79%		2.05%		7.43%		0.98%		1.20%		1.64%	

† $p\hat{d}v_i$ is the point-wise deviation of the i-th unique password of the real-world dataset RealSet, and can be extended on interval $[i_1, i_2]$ through Definition 2.
‡ The 8th unique password YfDbUfNjH10305070 of 000webhost is a default value [57]. The top-1 and top-9 defaults in Chegg correspond to MD5 hashes (without salt) that cannot be recovered. Why these passwords are popular may due to system settings (e.g., system-generated passwords) that are accepted by users. The exact reasons are beyond our comprehension and are *unlikely to alter our analysis in any significant way*.
∗ % means the proportion of top-10 unique passwords. This shows that Chinese passwords are more concentrated than their English counterparts as concluded in [56].

Point-wise deviation. We define the point-wise deviation to measure the difference in frequencies between the i-th unique password in the real-world and theoretical datasets.

***Definition** 1:* For the i-th unique password, the point-wise deviation $p\hat{d}v_i$ of the i-th unique password in RealSet is

$$p\hat{d}v_i = (\hat{f}_i - f_i)/f_i, \tag{4}$$

where $\hat{f}_i$ and f_i are the i-th unique password of RealSet and TheoSet, respectively. The sign of $p\hat{d}v_i$ is denoted as $\hat{\delta}_i$ and $\hat{\delta}_i \in \{-1, 1\}$.

Here we explain Definition 1. First, if the sign $\hat{\delta}_i$ is positive, then $\hat{f}_i > f_i$, and vice versa. Second, the absolute value $|p\hat{d}v_i|$ measures *the degree of point-wise deviation*, namely, the *relative error* between frequencies of the password with the same rank in RealSet and TheoSet. We use the term *point-wise deviation* to emphasize that the deviation is defined on *an individual password, which can be extended point by point to an interval of passwords with consecutive ranks*. Hence, by treating the sign and degree separately, we extend the domain of point-wise deviation from an individual password with rank i to an interval of passwords with ranks in $[i_1, i_2]$ as follows.

***Definition** 2:* If all passwords within ranks $[i_1, i_2]$ have the same sign. The point-wise deviation on interval $[i_1, i_2]$ is

$$p\hat{d}v_{[i_1,i_2]} = \hat{\delta}_{[i_1,i_2]} \min_{i\in[i_1,i_2]} |p\hat{d}v_i|, \tag{5}$$

where $\hat{\delta}_{[i_1,i_2]}$ satisfies

$$\hat{\delta}_{[i_1,i_2]} = \begin{cases} 1 & \text{If } p\hat{d}v_i \geq 0 \text{ for all } i \in [i_1, i_2] \\ -1 & \text{If } p\hat{d}v_i < 0 \text{ for all } i \in [i_1, i_2]. \end{cases} \tag{6}$$

The top-10 unique passwords, and their point-wise deviations of our eight datasets, are shown in Table IV. For each dataset, passwords are summarized in the first column, and their point-wise deviations are shown in the second column.

We demonstrate the implications of point-wise deviation results in Table IV. Firstly, the sign of the first (i.e., top-1) password is negative, i.e., $\hat{\delta}_1 < 0$ and $\hat{f}_1 < f_1$, indicating the frequency of the top-1 unique password in RealSet is smaller than that in TheoSet given by the CDF-Zipf distribution model. One possible reason is that the password 123456 is the most popular password in most of our datasets (except for 000webhost and Chegg due to their password policies [55]), and is widely regarded as weak and users consciously avoid choosing it. Secondly, passwords with the same sign are often connected, so the definition of the point-wise deviation on an interval (i.e., Definition 2) is practical. Since both $\hat{f}_i$ and f_i decrease gradually with i, $p\hat{d}v$ can be seen as a *continuous function* with i as the variable. Based on the sign-preserving property, if $pdv_{i_0} > 0$ (resp. <0) for some i_0, there exists a neighbourhood $[i_0, i_0 + Len]$ that $pdv_i > 0$ (resp. <0) for all $i \in [i_0, i_0 + Len]$. Furthermore, since CDF-Zipf is accurate, the slopes of RealSet and TheoSet are close enough, so Len is often large (e.g., $\hat{\delta}_{[2,136]}$=1 and Len=135 for Dodonew).

Fig. 1 takes a grasp of the frequencies of the top-100 unique passwords in RealSet and TheoSet under the log-log scale. The signs of passwords with a rank in $[2, 100]$ can be both negative and positive, which relates to users' languages and service types. For instance, the signs of the 2nd to 10th unique passwords in two Chinese datasets (Dodonew and Tianya) are all positive, i.e., $\hat{f}_i > f_i$, which may be because Chinese passwords are more concentrated (see Table IV) than their English counterparts [56].

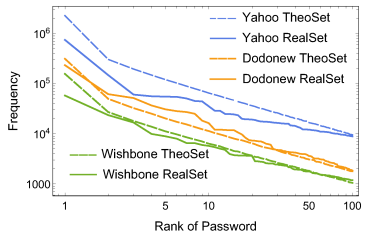


Fig. 1: A grasp of the frequencies of the top-100 unique passwords in Yahoo, Dodonew and Wishbone datasets. It is observed that RealSet is *below* TheoSet in Yahoo and Wishbone, while RealSet is *above* TheoSet in Dodonew after the top-2 passwords. Therefore, according to Definition 1, $p\hat{d}v_i < 0$ for $i \in [1, 100]$ in Yahoo and Wishbone, and $p\hat{d}v_i > 0$ for $i \in [2, 100]$ in Dodonew. This shows that passwords with the same sign are often connected, so Definition 2 is well-defined.

Based on the above observations, we generate the simulated dataset SmuSet by adjusting the point-wise deviation (denoted as pdv^s) in the following two metrics to simulate the real-world password deviations.

Absolute deviation metric. We suppose the first N_0 unique passwords are deviated, i.e., the deviation range is $[1, N_0]$. First, we determine the sign of the point-wise deviation. For RealSet, since passwords with the same signs are usually connected, there exist intervals in which each unique password has the same sign (e.g., $[2, 10]$ in Dodonew). Thus, we divide $[1, N_0]$ accordingly into *disjoint unions* of intervals $[i_1, i_2]$:

$$\begin{aligned} &[1, N_0] = \bigcup [i_1, i_2] \qquad \hat{\delta}_i = \hat{\delta}_{i_1} \text{ for all } i \in [i_1, i_2] \\ &[i_1, i_2] \bigcap [i_1', i_2'] = \phi \qquad \text{for any two different intervals.} \end{aligned} \tag{7}$$

On each interval $[i_1, i_2]$, we set the point-wise deviation pdv^s of the simulated dataset SmuSet based on these two rules:

1) The simulated point-wise deviation has the same sign as the real-world one, i.e., $\delta^s_{[i_1,i_2]} = \hat{\delta}_{[i_1,i_2]}$ on $[i_1, i_2]$.
2) The absolute value of the simulated point-wise deviation is *no more than* that of the real-world one, i.e., $0 < k_A = |pdv^s_{[i_1,i_2]}| \leq |\hat{pdv}_{[i_1,i_2]}|$ on $[i_1, i_2]$.

Hence, the i-th deviated password in SmuSet has frequency

$$f^s_i = f_i(1 + pdv^s_{[i_1,i_2]}) = f_i(1 + \hat{\delta}_{[i_1,i_2]}k_A). \quad (8)$$

As shown in Table IV, the point-wise deviation $\hat{pdv}_1$ of the first password is often different in sign and with a much larger degree. Therefore, we treat the first password separately.

Based on these rules, we calculate the p-value in the absolute deviation metric. Similar to Alg. 1, first, we generate J_0=10,000 TheoSets, deviate them in the absolute deviation metric and obtain SmuSets. Second, we fit each SmuSet with the GSS fitting method proposed by Wang et al. [54], and get (C^s, s^s, D^s_{KS}) as the KS statistic and characterization parameters. Third, we calculate the p-value for each SmuSet, and take an average (denoted as p-value^s) to mitigate statistical fluctuations. The process is shown in Alg. 2.

Algorithm 2: Calculate the p-value in absolute deviation metric (AbsDev).

Input : The parameters C and s, real-world dataset RealSet, point-wise deviation degree k_A and range $[1, N_0]$.
Output: p-value^s of the simulated dataset SmuSet.

1 **begin**
2 **for** $j = 1$ **to** J_0 **do**
3 $\text{TheoSet}_j = \text{THEOGEN}(C, s, |\mathcal{DS}|)$; /* Generate J_0 theoretical datasets. */
4 $(C'_j, s'_j, D'_{KS\,j}) = \text{GSS}(\text{TheoSet}_j)$; /* Fit J_0 theoretical datasets with the same method fitting the real-world dataset. */
5 **for** $j = 1$ **to** J_0 **do**
6 Divide $[1, N_0] = \bigcup[i_1, i_2]$ with $[i_1, i_2] \bigcap [1'_1, i'_2] = \phi$ and $\delta_{[i_1,i_2]}$ is fixed for each $[i_1, i_2]$; /* Divide the deviation ranges into disjoint unions of intervals. */
7 **for** $i = 1$ **to** N_0 **do**
8 $f^s_1 = f_1(1 + \delta_1 k_1)$; /* Deviate the first password. */
9 $f^s_i = f_i(1 + \hat{\delta}_{[i_1,i_2]}k_A)$; /* Deviate passwords ranking from i_1 to i_2 in each theoretical dataset. */
10 Rank $f^s_1, f^s_2 \ldots$ in descending order; $\text{SmuSet}_j = \{f'_1, f'_2 \ldots\}$;
11 $(C^s_j, s^s_j, D^s_{KS\,j}) = \text{GSS}(\text{SmuSet}_j)$; /* Fit J_0 simulated datasets with the GSS fitting method. */
12 $p\text{-value}^s_j = (\#\{D'_{KS\,j} | D'_{KS\,j} > D^s_{KS\,j}, 1 \leq j \leq J_0\} + 1)/(J_0 + 1)$; /* Calculate the p-value for each simulated dataset and smooth the p-value^s. */
13 $p\text{-value}^s = (p\text{-value}^s_1 + p\text{-value}^s_2 .. + p\text{-value}^s_{J_0})/J_0$; /* Take the average as the p-value^s. */
14 **Output**: p-value^s of the simulated dataset.

Relative deviation metric. We also explore the relative deviation metric, where the point-wise deviation pdv^s_i of the i-th unique password in SmuSet is proportional to that in RealSet, i.e., $pdv^s_i = \hat{pdv}_i k_R$ with the deviation parameter $0 < k_R \leq 1$. In this case, the frequency f^s_i in SmuSet has

$$f^s_i = f_i(1 + \hat{pdv}_i \cdot k_R) = f_i(1 - k_R) + \hat{f}_i k_R. \quad (9)$$

Since each deviated password has $\delta^s_i = \hat{\delta}_i$, there is no need to divide $[1, N_0]$ into disjoint unions of intervals. The process is similar to Alg. 2 except Lines 8 and 9 are replaced with Eq. 9, so we omit its presentation here.

Theories of deviation metrics. In this part, we *prove* that the maximum of D^s_{KS} (KS statistic of fitting SmuSet) increases with the deviation degrees k_A and k_R. As a result, we only need to try a limited number of k_A and k_R values to find the thresholds of deviations that make MCA reject CDF-Zipf.

Before entering the proofs, we state our assumptions. Firstly, similar to Sec. III-B, we assume $C^s \approx C'$ and $s^s \approx s'$ for SmuSet. Secondly, the rank of a password does not change significantly after deviation, that is, the i-th password in TheoSet also ranks approximately i in SmuSet. Suppose two passwords PW_i and PW_j have $f_i > f_j$ in TheoSet; (1) If both PW_i and PW_j are deviated, there is $f^s_i > f^s_j$ in SmuSet according to Eqs. 8 and 9, so their ranks in the deviation range are unchanged; (2) If PW_i is deviated but PW_j is not, the deviation degree can be tuned to minimize changes in rank. Thirdly, the direct maximum CDF deviation between TheoSet and its derived SmuSet is used as the maximum $\max D^s_{KS}$, i.e., $\max D^s_{KS} = |\text{CDF(SmuSet)} - \text{CDF(TheoSet)}|$. This is justified because D^s_{KS} resulting from GSS cannot exceed the direct maximum CDF deviation. Based on these assumptions, we state the two theorems on password deviations as follows.

***Theorem** 2:* In the absolute deviation metric, if passwords in SmuSet are deviated as $pdv^s_i = \hat{\delta}_{[i_1,i_2]} \cdot k_A$ for $i \in [i_1, i_2]$, and the deviation degree k_A satisfies $0 \leq k_A \leq |\hat{pdv}_{[i_1,i_2]}|$, then the maximum $\max D^s_{KS}$ increases as k_A increases.

***Theorem** 3:* In the relative deviation metric, if passwords are deviated as $pdv^s_i = \hat{\delta}_i \cdot |\hat{pdv}_i| k_R$ for $i \in [1, N_0]$ and $0 < k_R \leq 1$, then the maximum $\max D^s_{KS}$ increases as k_R increases (see proofs of Theorems 2 and 3 in Appendix B).

Discussion. We now make a few justifications. Firstly, we only explore the case where only one interval $[i_1, i_2]$ is involved in the absolute deviation case. For multiple intervals with signs fixed on each of them, we can apply Theorem 2 inductively to obtain the global solution. Secondly, we clarify why not consider the relationship between the KS statistic D^s_{KS} and the deviation range N_0. In the absolute deviation case, $\hat{\delta}_i$ on multiple is complicated; In the relative deviation case, there is also no simple monotonicity. Hence, we mainly focus on how $\max D^s_{KS}$ changes with deviation degrees k_A and k_R.

Theorems 2 and 3 reveal that in both deviation metrics, $\max D^s_{KS}$ increases as the deviation degrees k_A and k_R increase, so the p-value^s calculating the proportion of type-1 deviation larger than the KS statistic of SmuSet (i.e., $D'_{KS} > D^s_{KS}$) should decrease as k_A and k_R increase. As a result, for a given interval of passwords with a rank in $[i_1, i_2]$ (resp. $[1, N_0]$), if a specific k_A (resp. k_R) value can make p-value^s<0.01, so a larger value can also achieve this goal. With this monotonicity guarantee, we only need to try a limited number of k_A and k_R values to find the thresholds.

D. Numerical experiments of simulations

In this section, we describe the experiment setups and results of deviation threshold investigations, in both absolute and relative deviation metrics for MCA to reject CDF-Zipf.

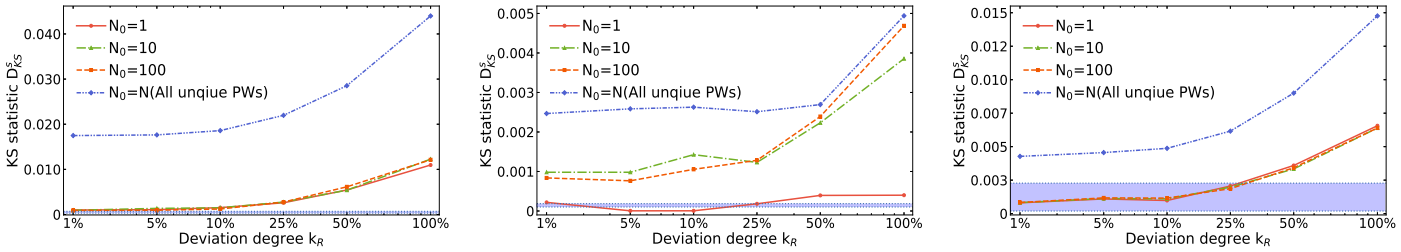
Experiment setups. We first consider the absolute deviation metric. We set the deviation range parameter $N_0 \in \{1, 10, 100\}$, divide $[1, N_0]$ into disjoint unions of intervals, and treat the first unique password separately, as shown in Alg.

TABLE V: KS statistics and p-values of simulated datasets.[†]

Dataset	Absolute point-wise deviation[‡]									Relative point-wise deviation								
	Point-wise deviation	D^s_{KS}	p-value^s	Point-wise deviation	D^s_{KS}	p-value^s	Point-wise deviation	D^s_{KS}	p-value^s	Point-wise deviation	D^s_{KS}	p-value^s	D^s_{KS}	p-value^s	D^s_{KS}	p-value^s	D^s_{KS}	p-value^s
Yahoo	Dev-range $N_0=1$			Dev-range $N_0=10$ $[1,10]=\{1\}\bigcup[2,10]$			Dev-range $N_0=100$ $[1,100]=\{1\}\bigcup[2,10]\bigcup[11,100]$			Dev-range $N_0=1$			Dev-range $N_0=10$		Dev-range $N_0=100$		All unique PWs	
	-1%	0.000928	$<10^{-4}$	-1%	0.000842	$<10^{-4}$	-1%	0.001066	$<10^{-4}$	1%	0.000986	$<10^{-4}$	0.000937	$<10^{-4}$	0.000898	$<10^{-4}$	0.017486	$<10^{-4}$
	-5%	0.001409	$<10^{-4}$	-5%, -1%	0.001206	$<10^{-4}$	-5%, -1%	0.000952	$<10^{-4}$	5%	0.001076	$<10^{-4}$	0.001355	$<10^{-4}$	0.000939	$<10^{-4}$	0.017633	$<10^{-4}$
				-5%	0.001002	$<10^{-4}$	-5%	0.001317	$<10^{-4}$	10%	0.001520	$<10^{-4}$	0.001473	$<10^{-4}$	0.001213	$<10^{-4}$	0.018583	$<10^{-4}$
	-10%	0.001542	$<10^{-4}$	-11%	0.001900	$<10^{-4}$	-10%, -1%	0.002226	$<10^{-4}$	25%	0.002595	$<10^{-4}$	0.002756	$<10^{-4}$	0.002749	$<10^{-4}$	0.021969	$<10^{-4}$
	-25%	0.004221	$<10^{-4}$	-20%	0.002415	$<10^{-4}$	-10%, -5%	0.001968	$<10^{-4}$	50%	0.005425	$<10^{-4}$	0.005384	$<10^{-4}$	0.006131	$<10^{-4}$	0.028538	$<10^{-4}$
				-25%	0.004221	$<10^{-4}$	-25%, -1%	0.004710	$<10^{-4}$	100%	0.010961	$<10^{-4}$	0.012264	$<10^{-4}$	0.012122	$<10^{-4}$	0.043985	$<10^{-4}$
Dodo-new	Dev-range $N_0=1$			Dev-range $N_0=10$ $[1,10]=\{1\}\bigcup[2,10]$			Dev-range $N_0=100$ $[1,100]=\{1\}\bigcup[2,10]\bigcup[11,100]$			Dev-range $N_0=1$			Dev-range $N_0=10$		Dev-range $N_0=100$		All unique PWs	
	-1%	0.001203	**0.185**	-1%, 1%	0.001352	**0.029**	-1%, 1%	0.000851	**0.361**	1%	0.000218	0.003	0.000980	**0.265**	0.000837	**0.381**	0.002467	$<10^{-4}$
	-5%	0.000990	**0.240**	-5%, 5%	0.001221	**0.055**	-5%, 1%	0.000770	**0.408**	5%	0.000004	**0.376**	0.000980	**0.256**	0.000764	**0.423**	0.002586	$<10^{-4}$
	-10%	0.001270	**0.079**	-10%, 1%	0.001187	**0.074**	-10%, 1%	0.001323	**0.029**	10%	0.000005	**0.015**	0.001426	**0.032**	0.001054	**0.169**	0.002628	$<10^{-4}$
				-10%, 5%	0.001775	**0.012**	-10%, 5%, 1%	0.001625	0.008	25%	0.000182	**0.026**	0.001230	**0.040**	0.001283	**0.050**	0.002513	$<10^{-4}$
	-25%	0.002953	$<10^{-4}$	-25%, 1%	0.002915	$<10^{-4}$	-25%, 1%	0.002849	$<10^{-4}$	50%	0.000394	0.005	0.002232	0.004	0.002391	$<10^{-4}$	0.002693	$<10^{-4}$
				-25%, 5%	0.002962	$<10^{-4}$	-25%, 5%, 1%	0.002918	$<10^{-4}$	100%	0.000400	$<10^{-4}$	0.003855	$<10^{-4}$	0.004687	$<10^{-4}$	0.004941	$<10^{-4}$
Wish-bone	Dev-range $N_0=1$			Dev-range $N_0=10$ $[1,10]=\{1\}\bigcup[2,10]$			Dev-range $N_0=100$ $[1,100]=\{1\}\bigcup[2,62]\bigcup[63,100]$			Dev-range $N_0=1$			Dev-range $N_0=10$		Dev-range $N_0=100$		All unique PWs	
	-1%	0.000823	**0.689**	-1%	0.000800	**0.710**	-1%, 1%	0.001813	**0.010**	1%	0.000806	**0.709**	0.000818	**0.698**	0.000831	**0.681**	0.004274	$<10^{-4}$
	-5%	0.001078	**0.357**	-5%, -1%	0.001049	**0.355**	-5%, -1%	0.001189	**0.172**	5%	0.001094	**0.290**	0.001125	**0.247**	0.001173	**0.189**	0.004558	$<10^{-4}$
				-5%	0.001219	**0.143**	-5%, 1%	0.001155	**0.210**	10%	0.000979	**0.466**	0.001102	**0.278**	0.001158	**0.206**	0.004874	$<10^{-4}$
	-10%	0.001991	0.004	-10%, -1%	0.001916	0.007	-10%, -1%, 1%	0.001763	**0.011**	25%	0.002060	0.005	0.001974	0.005	0.001862	0.008	0.006160	$<10^{-4}$
	-25%	0.002952	$<10^{-4}$	-25%, -1%	0.002709	$<10^{-4}$	-25%, -1%, 1%	0.002816	$<10^{-4}$	50%	0.003594	$<10^{-4}$	0.003337	$<10^{-4}$	0.003425	$<10^{-4}$	0.008999	$<10^{-4}$
	-50%	0.005170	$<10^{-4}$	-50%, -1%	0.005182	$<10^{-4}$	-50%, -1%, 1%	0.005180	$<10^{-4}$	100%	0.006563	$<10^{-4}$	0.006391	$<10^{-4}$	0.006394	$<10^{-4}$	0.014763	$<10^{-4}$

[†] D^s_{KS} is the KS statistic of fitting the simulated dataset SmuSet, and the p-value^s is the corresponding p-value. The **bold** p-value^s results are the ones $\geq$ **0.01** threshold, meaning that CDF-Zipf can pass Monte Carlo approach (MCA), so passwords are supposed to follow it. We can see that with small (i.e., 1∼25%) deviations in both absolute and relative deviation metrics, the p-value <0.01 and even $<10^{-4}$, so MCA is not an effective method in telling whether a large-scale password dataset follows CDF-Zipf.

‡ In the absolute point-wise deviation metric, the deviation range (Dev-range) is divided into disjoints unions of intervals to ensure the sign on each interval is fixed, i.e., $[1, N_0] = \bigcup[i_1, i_2]$ with $[i_1, i_2]\bigcap[i'_1, i'_2] = \phi$ for any two different intervals.



(a) Relative deviation of Yahoo. (b) Relative deviation of Dodonew. (c) Relative deviation of Wishbone.

Fig. 2: Relative deviations of Yahoo, Dodonew and Wishbone datasets. Each curve represents a different deviation range. The shadow area corresponds to the maximum and minimum of KS statistic D'_{KS} resulting from fitting the theoretical dataset TheoSet (used to measure statistical randomness). Only within the shadow area, the Monte Carlo approach (MCA) p-value^s can possibly be >0.01. However, all D^s_{KS} resulting from fitting SmuSet surpasses the shadow area with $k_R \geq 50\%$.

2. We also set $|pdv_{[i_1,i_2]}|>|pdv_{[i_3,i_4]}|$ for other consecutive intervals to smooth deviations. The reason not to consider other passwords (e.g., the first 100∼1,000 unique passwords) lies in two folds: (1) These passwords often have much smaller frequencies compared with the top-100 passwords, so their deviation effects are less significant; (2) Point-wise deviations of these passwords are often sharply different, which complicates the simulation. For instance, in Yahoo, $\hat{pdv}_{115} = 0.68\%$ but $\hat{pdv}_{1,000} = 35\%$, so considering them will unnecessarily complicate the simulation.

Second, we consider the deviation degree. We investigate the real-world deviation degree $\hat{pdv}_{[i_1,i_2]}$ on each disjoint interval $[i_1, i_2]$ and set it as the maximum of the simulated deviation degree, i.e., $k_A = |pdv^s_{[i_1,i_2]}| = 1\%, 5\%, 10\%, 25\%, \ldots, |\hat{pdv}_{[i_1,i_2]}|$, (where $|\hat{pdv}_{[i_1,i_2]}| = \min_{i\in[i_1,i_2]} |\hat{pdv}_i|$, see Definition 2). We also consider two levels of p-value thresholds as in Sec. III-B: (1) p-value$^s<0.01$ (the standard threshold) and (2) p-value$^s<10^{-4}$, i.e., $D^s_{KS} > \max D'_{KS}$, where type-2 deviation >0 for *all* generated simulated datasets SmuSets.

In the relative deviation metric, apart from deviation ranges in the absolute metric, we also set $N_0 = N$, i.e., *all* passwords are deviated. As mentioned in Sec. III-C, we deviate passwords directly based on Eq. 9, and set the deviation parameter $k_R = 1\%, 5\%, 10\%, 25\%, 50\%$, and 100%.

Experiment results. Without loss of generality, Table V shows the KS statistics and p-value^{s}s of fitting SmuSets of Yahoo, Dodonew and Wishbone (results of others are similar). The left-hand columns record the results of absolute deviations, and the right-hand columns record the results of relative deviations.

Table V demonstrates that: (1) In both deviation metrics, the KS statistic D^s_{KS} increases monotonically as deviation degree pdv^s_i with only a few exceptions (e.g., $pdv^s_1 = -5\%$ in Dodonew), which agrees well with Theorems 2 and 3; (2) In both cases, pdv^s_i only need to be 1%∼25% to make p-value$^s<0.01$, and 1%∼50% to make p-value$^s<10^{-4}$ (with a few exceptions); Particularly, when *all* passwords are deviated (i.e., N_0=N in the relative case), deviation as small as 1% is enough to make p-value$^s<10^{-4}$. This means even if the p-value threshold is <0.01 (e.g., 0.005), MCA will always reject CDF-Zipf with the real-world deviations for large-scale datasets. All this *substantiates* Wang et al.'s [54] conjecture on the effect of sample size for CDF-Zipf [54], so small and non-notable deviations would indeed lead to statistical significance for large-scale datasets. We also present how the KS statistic D^s_{KS} changes with the relative deviation degree k_R in Fig. 2.

Summary. We use both theories and experiments to investigate the extents to which MCA rejects CDF-Zipf. *For the first time*, we *prove* that under the CDF-Zipf distribution model and the GSS fitting method, type-1 deviation (i.e., statistical randomness) converges to 0 asymptotically as the dataset size increases. As a result, the KS statistic of a large password dataset (e.g., $\geq$ 1 million) mainly comes from type-2

TABLE VI: Alternative distribution models to CDF-Zipf.†

Distribution‡	Distribution model in rank-frequency (RF) coordinate system		Distribution model in frequency-frequency (FF) coordinate system, but converted to RF system	
	PDF kernel	CDF	PDF kernel	CDF
CDF-Zipf [54]	r^{s-1}	Cr^s	r^{s-1}	Cr^s
Exponential [20]	$\exp(-\lambda r)$	$1-\exp(-\lambda r)$	$\ln\frac{r}{\lambda\lvert\mathcal{DS}\rvert}$	$-\frac{r}{\lambda\lvert\mathcal{DS}\rvert}\ln\frac{r}{\lambda\lvert\mathcal{DS}\rvert}$
Lognormal [20]	$\frac{1}{r}\exp(-\frac{(\ln r-\mu)^2}{2\sigma^2})$	$\Phi(\frac{\ln r-\mu}{\sqrt{2}\sigma})$	$\exp(\sigma\Phi^{-1}(\frac{\exp(\mu+\sigma^2/2)}{\lvert\mathcal{DS}\rvert}r+\frac{1}{2}))$	$\Phi\left(2(\sigma-\Phi^{-1}(1-\frac{\exp(\mu+\sigma^2/2)}{\lvert\mathcal{DS}\rvert}r))\right)-\frac{1}{2}$
Zipf-cutoff [20]	$r^{-\alpha}\exp(-\lambda r)$	$Q(1-\alpha,\lambda r)$	$Q^{-1}(1-\alpha,\frac{(1-\alpha)\cdot r}{\lambda\lvert\mathcal{DS}\rvert})$	$Q\left(2-\alpha,Q^{-1}(1-\alpha,\frac{(1-\alpha)r}{\lambda\lvert\mathcal{DS}\rvert})\right)$
Stretched-exponential [20]	$r^{\alpha-1}\exp(-\lambda r^{\alpha})$	$1-\exp(-\lambda r^{\alpha})$	$\left(\ln\frac{\Gamma(1+1/\alpha)r}{\lambda\lvert\mathcal{DS}\rvert}\right)^{\frac{1}{\alpha}}$	$Q\left(1+\frac{1}{\alpha},-\ln\frac{\Gamma(1+\frac{1}{\alpha})r}{\lambda\lvert\mathcal{DS}\rvert}\right)$

† The kernel is the term in the PDF expression which only includes variables but no coefficients. All distributions are considered in both rank-frequency (RF) and frequency-frequency (FF) coordinate systems. For distributions in the FF systems, they have been converted to the RF system as shown in the right half of the table.
‡ Since the power-law and Zipf are equivalent, their PDF kernel has the same form in two coordinate systems. For lognormal, $\Phi(x)$ is the CDF of the standard normal distribution $N(0,1)$, and Φ^{-1} is the inverse function of Φ. For Zipf-cutoff, Q is the regularized gamma function defined as $Q(s,x)=\frac{\Gamma(s,x)}{\Gamma(s)}$, where $\Gamma(s,x)=\int_x^\infty t^{s-1}\exp(-t)dt$ and $\Gamma(x)=\int_0^\infty t^{s-1}\exp(-t)dt$. There also has to be $s,x>0$ to make $Q(s,x)$ to be a real number, so $\alpha<1$.

deviation (using CDF-Zipf distribution model), and the p-value is $<10^{-4}$. Experiment results on eight large-scale datasets reveal that the dataset size only needs to be $\geq 0.25M$ to make p-value<0.01 to reject CDF-Zipf, and $\geq 1M$ to make p-value$<10^{-4}$. Next, we propose both the absolute and relative deviation metrics to simulate real-world deviations, and reveal that 1% random deviations in both metrics suffice to make MCA reject CDF-Zipf. These rigorous mathematical proofs and extensive experiments substantiate Wang et al.'s [54] conjecture on the effect of sample size on CDF-Zipf.

IV. New Models for Password Distribution

Since MCA always rejects CDF-Zipf for large-scale datasets, A natural question arises: *Whether there are better distribution models that can provide higher accuracy and pass MCA?* We compare the fitting accuracy of CDF-Zipf with other possible distributions, and use MCA to calculate their p-values to answer this question.

A. Necessity of considering alternative distributions

We show the necessity to explore alternative distributions. If passwords follow CDF-Zipf (i.e., $P_r=Cr^s$), the CDF curve should be a straight line under the log-log scale. However, due to point-wise deviations revealed in Sec. III-C (particularly Table IV and Fig. 1), a distorted line but not a straight one is more realistic in practice. Therefore, there are likely to be more accurate distribution models with *roughly linear* curves under the log-log scale. However, to the best of our knowledge, no prior work has explored these alternative distribution models.

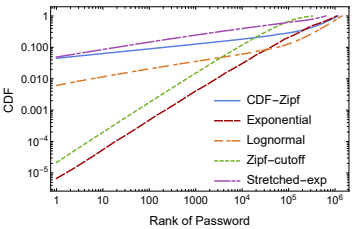


Fig. 3: An exhibition of alternative distribution models. Besides CDF-Zipf, the other four distribution models (stretched-exp represents the stretched-exponential, see details in Table VI) are also roughly linear under the log-log scale, so passwords may also follow one of them.

We present an example of five distributions in Fig. 3. All CDFs (see details in Table VI) are roughly linear under the log-log scale, but only the blue curve is actually drawn from CDF-Zipf. In addition, when using linear regression to fit these CDFs, we find coefficients of determination $0.850<R^2<0.999$ (where $R^2\in[0,1]$ and the larger the R^2, the closer the data to a line). There is potential that, among these four (and possibly other) alternative models, someone(s) will be more accurate than CDF-Zipf. We confirm this in what follows.

Two coordinate systems. We present two different coordinate systems used to analyze the password frequency, and investigate the relation between these two systems.

First, we discuss the rank-frequency coordinate system (abbreviated as the RF system) used in Sec. III. In this system, the X-axis records the rank r of a password, and the Y-axis records the frequency f_r. Though in practice, the Y-coordinate is converted to record the CDF $P_r=\sum_{i=1}^{r}f_i/|\mathcal{DS}|$, (where $|\mathcal{DS}|$ is the dataset size) [54], CDF is equivalent to frequency essentially, so the rank-frequency essence is still maintained when the CDF is on the Y-axis.

Besides rank-frequency, another way to see the data is to count the number (i.e., frequency) of unique passwords n_k that are each used by exactly k users. In this system, the X-axis records the password frequency f_r, which is the same as the Y-axis of the RF system. Accordingly, the Y-axis records the frequency n_{f_r} of *unique passwords occurring* f_r *times*. We denote this coordinate system as the frequency-frequency system (abbreviated as FF system) with x as the variable. In practice, the FF system is widely used in research areas like physics [20], complex networks [5], and marketing [24], so our consideration is reasonable.

We now show the relationship between the RF and FF systems. If a password PW has the rank r and frequency f_r in a dataset, its coordinate in the RF system is (r,f_r). Besides, if there are n_{f_r} unique passwords with frequency f_r in the dataset, the coordinate of PW in the FF system is (f_r,n_{f_r}). Hence, the complementary CDF in the FF system has

$$P(X\geq x)=r/N, \tag{10}$$

where N is the number of unique passwords. Based on this property, we can use the same techniques of Adamic's work [3] to convert *any* distribution in the FF to the RF system.

B. Alternative distribution models

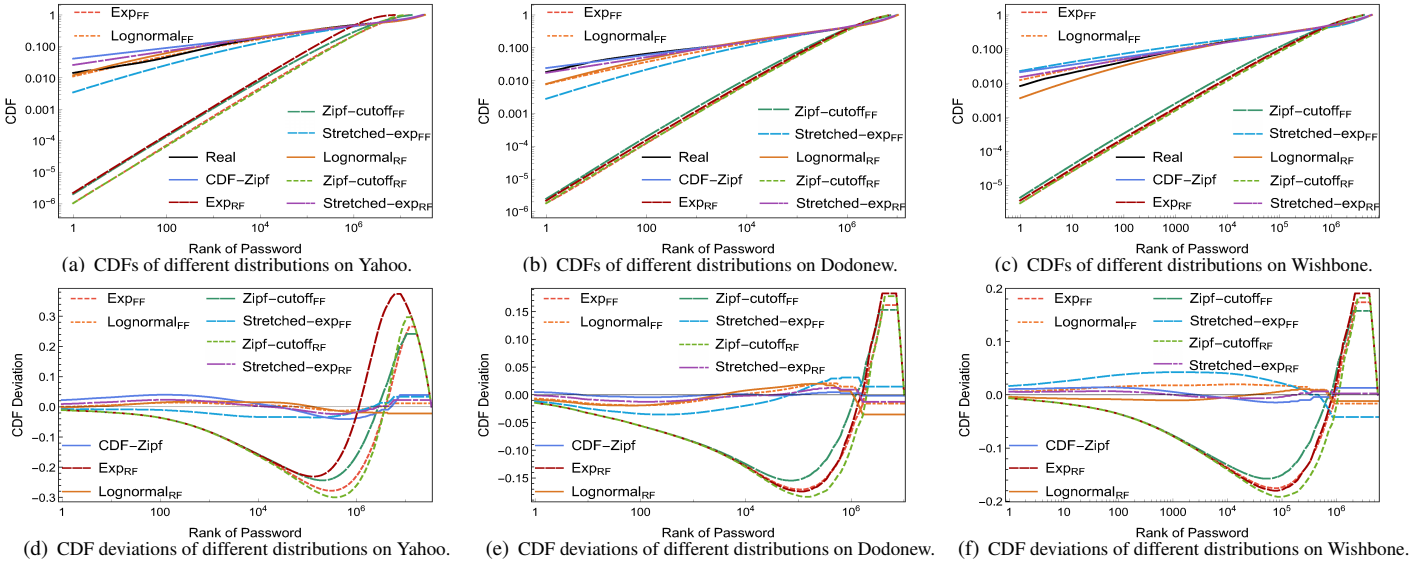
In 2009, Clauset et al. [20] compared four alternative distribution models with power-law (equivalent to CDF-Zipf [3]) in the FF system. Inspired by this idea, we consider these four models in both RF and FF systems, a total of eight alternative models. The PDF and CDF expressions in the RF system are shown in the first two columns of Table VI.

We justify investigating models in Table VI in both coordinate systems. First, regardless of coordinate systems, these distributions are roughly linear under the log-log scale as shown in Fig. 3, so it is necessary to consider them in both

TABLE VII: GSS fitting results of alternative distribution models.†

Dataset	Distribution	θ_1	θ_2	$D_{KS}^{\ddagger}$	p-value‡	Dataset	Distribution	θ_1	θ_2	$D_{KS}^{\ddagger}$	p-value‡
Yahoo	CDF-Zipf(C,s)	0.033148	0.180907	**0.040775**	$<10^{-4}$	Tianya	CDF-Zipf(C,s)	0.062022	0.155290	**0.022940**	$<10^{-4}$
	Exponential(λ)$_{\mathrm{RF}}$	5.24×10^{-7}	—	**0.394789**	$<10^{-4}$		Exponential(λ)$_{\mathrm{RF}}$	1.11×10^{-6}	—	**0.300070**	$<10^{-4}$
	Exponential(λ)$_{\mathrm{FF}}$	0.288010	—	**0.277861**	$<10^{-4}$		Exponential(λ)$_{\mathrm{FF}}$	0.144383	—	**0.292062**	$<10^{-4}$
	Lognormal(μ,σ)$_{\mathrm{RF}}$	15.01919	6.359214	**0.018063**	$<10^{-4}$		Lognormal(μ,σ)$_{\mathrm{RF}}$	13.29193	6.780193	**0.022918**	$<10^{-4}$
	Lognormal(μ,σ)$_{\mathrm{FF}}$	-25.90537	5.160470	**0.014101**	$<10^{-4}$		Lognormal(μ,σ)$_{\mathrm{FF}}$	-32.91796	5.856597	**0.019428**	$<10^{-4}$
	Zipf-cutoff(λ,α)$_{\mathrm{RF}}$	0.000027	-6159.767	**0.300296**	$<10^{-4}$		Zipf-cutoff(λ,α)$_{\mathrm{RF}}$	0.000123	-5802.884	**0.313710**	$<10^{-4}$
	Zipf-cutoff(λ,α)$_{\mathrm{FF}}$	0.106206	0.996573	**0.154641**	$<10^{-4}$		Zipf-cutoff(λ,α)$_{\mathrm{FF}}$	0.054899	0.983886	**0.263220**	$<10^{-4}$
	Stretched-exponential(λ,α)$_{\mathrm{RF}}$	0.020419	0.238575	**0.023873**	$<10^{-4}$		Stretched-exponential(λ,α)$_{\mathrm{RF}}$	0.041705	0.214004	**0.012353**	$<10^{-4}$
	Stretched-exponential(λ,α)$_{\mathrm{FF}}$	19.34152	0.050895	**0.035185**	$<10^{-4}$		Stretched-exponential(λ,α)$_{\mathrm{FF}}$	36.71749	0.026701	**0.021080**	$<10^{-4}$
Dodonew	CDF-Zipf(C,s)	0.019255	0.211921	**0.004979**	$<10^{-4}$	Mathway	CDF-Zipf(C,s)	0.010541	0.245255	**0.011059**	$<10^{-4}$
	Exponential(λ)$_{\mathrm{RF}}$	3.07×10^{-7}	—	**0.178653**	$<10^{-4}$		Exponential(λ)$_{\mathrm{RF}}$	2.23×10^{-7}	—	**0.145354**	$<10^{-4}$
	Exponential(λ)$_{\mathrm{FF}}$	0.856726	—	**0.170507**	$<10^{-4}$		Exponential(λ)$_{\mathrm{FF}}$	1.152190	—	**0.147374**	$<10^{-4}$
	Lognormal(μ,σ)$_{\mathrm{RF}}$	15.98301	6.364878	**0.019549**	$<10^{-4}$		Lognormal(μ,σ)$_{\mathrm{RF}}$	16.22701	5.446235	**0.007611**	$<10^{-4}$
	Lognormal(μ,σ)$_{\mathrm{FF}}$	-26.81404	5.035345	**0.021462**	$<10^{-4}$		Lognormal(μ,σ)$_{\mathrm{FF}}$	-20.13930	4.289310	**0.005521**	$<10^{-4}$
	Zipf-cutoff(λ,α)$_{\mathrm{RF}}$	0.000036	-6003.371	**0.184108**	$<10^{-4}$		Zipf-cutoff(λ,α)$_{\mathrm{RF}}$	0.000094	-6062.899	**0.320328**	$<10^{-4}$
	Zipf-cutoff(λ,α)$_{\mathrm{FF}}$	0.349199	0.993975	**0.154641**	$<10^{-4}$		Zipf-cutoff(λ,α)$_{\mathrm{FF}}$	0.199062	0.997929	**0.152297**	$<10^{-4}$
	Stretched-exponential(λ,α)$_{\mathrm{RF}}$	0.013778	0.254693	**0.013313**	$<10^{-4}$		Stretched-exponential(λ,α)$_{\mathrm{RF}}$	0.006868	0.292796	**0.004638**	$<10^{-4}$
	Stretched-exponential(λ,α)$_{\mathrm{FF}}$	20.66430	0.052236	**0.036053**	$<10^{-4}$		Stretched-exponential(λ,α)$_{\mathrm{FF}}$	14.24821	0.072425	**0.016458**	$<10^{-4}$
000webhost	CDF-Zipf(C,s)	0.005738	0.282561	**0.005084**	$<10^{-4}$	Chegg	CDF-Zipf(C,s)	0.008178	0.235996	**0.008171**	$<10^{-4}$
	Exponential(λ)$_{\mathrm{RF}}$	4.98×10^{-8}	—	**0.181766**	$<10^{-4}$		Exponential(λ)$_{\mathrm{RF}}$	3.60×10^{-8}	—	**0.139127**	$<10^{-4}$
	Exponential(λ)$_{\mathrm{FF}}$	1.440604	—	**0.109905**	$<10^{-4}$		Exponential(λ)$_{\mathrm{FF}}$	1.993823	—	**0.130947**	$<10^{-4}$
	Lognormal(μ,σ)$_{\mathrm{RF}}$	16.29745	4.835015	**0.017584**	$<10^{-4}$		Lognormal(μ,σ)$_{\mathrm{RF}}$	18.86203	6.624927	**0.015775**	$<10^{-4}$
	Lognormal(μ,σ)$_{\mathrm{FF}}$	-15.49782	3.703538	**0.013719**	$<10^{-4}$		Lognormal(μ,σ)$_{\mathrm{FF}}$	-29.71207	5.114532	**0.003523**	$<10^{-4}$
	Zipf-cutoff(λ,α)$_{\mathrm{RF}}$	0.000020	-4394.015	**0.122391**	$<10^{-4}$		Zipf-cutoff(λ,α)$_{\mathrm{RF}}$	0.000010	-9932.162	**0.138310**	$<10^{-4}$
	Zipf-cutoff(λ,α)$_{\mathrm{FF}}$	0.630114	0.994591	**0.096630**	$<10^{-4}$		Zipf-cutoff(λ,α)$_{\mathrm{FF}}$	0.733076	0.997743	**0.118005**	$<10^{-4}$
	Stretched-exponential(λ,α)$_{\mathrm{RF}}$	0.003668	0.330296	**0.008472**	$<10^{-4}$		Stretched-exponential(λ,α)$_{\mathrm{RF}}$	0.006048	0.268884	**0.004963**	$<10^{-4}$
	Stretched-exponential(λ,α)$_{\mathrm{FF}}$	18.60976	0.062591	**0.015775**	$<10^{-4}$		Stretched-exponential(λ,α)$_{\mathrm{FF}}$	46.07984	0.024394	**0.013773**	$<10^{-4}$
Rockyou	CDF-Zipf(C,s)	0.038208	0.185939	**0.045357**	$<10^{-4}$	Wishbone	CDF-Zipf(C,s)	0.017144	0.230503	**0.014775**	$<10^{-4}$
	Exponential(λ)$_{\mathrm{RF}}$	5.63×10^{-7}	—	**0.288448**	$<10^{-4}$		Exponential(λ)$_{\mathrm{RF}}$	5.52×10^{-7}	—	**0.183852**	$<10^{-4}$
	Exponential(λ)$_{\mathrm{FF}}$	0.159546	—	**0.279785**	$<10^{-4}$		Exponential(λ)$_{\mathrm{FF}}$	0.175439	—	**0.082278**	$<10^{-4}$
	Lognormal(μ,σ)$_{\mathrm{RF}}$	13.44272	5.420420	**0.006333**	$<10^{-4}$		Lognormal(μ,σ)$_{\mathrm{RF}}$	14.86753	5.343869	**0.010363**	$<10^{-4}$
	Lognormal(μ,σ)$_{\mathrm{FF}}$	-20.92089	4.713954	**0.015878**	$<10^{-4}$		Lognormal(μ,σ)$_{\mathrm{FF}}$	-30.92769	5.413191	**0.019383**	$<10^{-4}$
	Zipf-cutoff(λ,α)$_{\mathrm{RF}}$	0.000078	-5106.710	**0.303566**	$<10^{-4}$		Zipf-cutoff(λ,α)$_{\mathrm{RF}}$	0.000063	-5712.203	**0.190858**	$<10^{-4}$
	Zipf-cutoff(λ,α)$_{\mathrm{FF}}$	0.998474	0.070754	**0.248137**	$<10^{-4}$		Zipf-cutoff(λ,α)$_{\mathrm{FF}}$	0.345017	0.995556	**0.157640**	$<10^{-4}$
	Stretched-exponential(λ,α)$_{\mathrm{RF}}$	0.022520	0.253404	**0.024906**	$<10^{-4}$		Stretched-exponential(λ,α)$_{\mathrm{RF}}$	0.012051	0.277269	**0.007151**	$<10^{-4}$
	Stretched-exponential(λ,α)$_{\mathrm{FF}}$	19.06201	0.050322	**0.029889**	$<10^{-4}$		Stretched-exponential(λ,α)$_{\mathrm{FF}}$	55.37731	0.019271	**0.042505**	$<10^{-4}$

† The RF in subscript means the distribution model is original in the rank-frequency (RF) system, and the FF subscript means the distribution model is original in the frequency-frequency (FF) system, but converted to the RF system. θ_1 and θ_2 denote the parameters characterizing each distribution, e.g., $\theta_1 = C$ and $\theta_2 = s$ for the CDF-Zipf distribution model. The two exponential distribution models, exponential$_{\mathrm{RF}}$ and exponential$_{\mathrm{FF}}$ only have one parameter λ, and thus θ_1 is needed.
‡ The smallest KS statistic D_{KS} among CDF-Zipf and alternative models are colored, and the p-value denotes Monte Carlo approach (MCA) results of alternative models.



(a) CDFs of different distributions on Yahoo. (b) CDFs of different distributions on Dodonew. (c) CDFs of different distributions on Wishbone.

(d) CDF deviations of different distributions on Yahoo. (e) CDF deviations of different distributions on Dodonew. (f) CDF deviations of different distributions on Wishbone.

Fig. 4: CDFs and CDF deviations (whose maximum absolute value are the KS statistics) of alternative distributions. The RF in the subscript means the distribution model is original in the rank-frequency (RF) system, and the FF means the distribution model is originally in the frequency-frequency (FF) system and converted to the RF system.

systems. Second, these distribution models are also meaningful. For instance, the lognormal distribution often characterizes the random multiplicative process [34] of variables. Another example is the stretched-exponential, which is generated similar to CDF-Zipf but with more stringent constraints [29]. Third, though there may be other distribution models for passwords to follow, they are either less common, or similar to our considered models (e.g., inverse gamma distribution). Therefore, our consideration of alternative distribution models is as comprehensive as possible.

We use Eq. 10 as the key to convert distributions in the FF system to the RF system (see Appendix C for more details). The converted PDFs and CDFs are shown in the last two columns of Table VI. Justification of this conversion lies in the fact the GSS fitting method in the RF system performs the best for CDF-Zipf [54]: GSS not only leads to the smallest KS statistic, but also can cover the entire dataset. Thus, GSS can optimize other models in the RF system as well. As a result, even if the converted PDFs and CDFs in the RF system are complicated and are not commonly used, we still fit passwords with them as their equivalences are common in the FF system.

TABLE VIII: p-values of password subsets and entire datasets of the stretched-exponential distribution model in RF system.†

Dataset	Dataset size	D_{KS}	D'_{KS}	p-value	$\Delta\lambda$	$\Delta\alpha$	Dataset	Dataset size	D_{KS}	D'_{KS}	p-value	$\Delta\lambda$	$\Delta\alpha$
Yahoo	$0.05M$	0.004348	0.000420 ∼ 0.008380	**0.058941**	10^{-4}	10^{-3}	Tianya	$0.05M$	0.007597	0.000600 ∼ 0.011220	**0.010989**	10^{-4}	10^{-4}
	$0.1M$	0.004147	0.000400 ∼ 0.006650	**0.035964**	10^{-5}	10^{-4}		$0.1M$	0.007633	0.000440 ∼ 0.009490	0.002997	10^{-4}	10^{-4}
	$0.25M$	0.005693	0.000372 ∼ 0.004884	$<10^{-4}$	10^{-5}	10^{-4}		$0.25M$	0.009860	0.000396 ∼ 0.004860	$<10^{-4}$	10^{-5}	10^{-5}
	$0.5M$	0.006759	0.000294 ∼ 0.002962	$<10^{-4}$	10^{-5}	10^{-4}		$0.5M$	0.011142	0.000396 ∼ 0.003090	$<10^{-4}$	10^{-5}	10^{-5}
	Entire set	0.023873	0.000733 ∼ 0.000990	$<10^{-4}$	10^{-6}	10^{-4}		Entire set	0.012353	0.000274 ∼ 0.001570	$<10^{-4}$	10^{-4}	10^{-4}
Dodonew	$0.05M$	0.007377	0.000480 ∼ 0.009960	0.001000	10^{-5}	10^{-4}	Chegg	$0.05M$	0.005935	0.000200 ∼ 0.005820	$<10^{-4}$	10^{-3}	10^{-2}
	$0.1M$	0.006994	0.000320 ∼ 0.007350	0.001190	10^{-5}	10^{-4}		$0.1M$	0.006080	0.000150 ∼ 0.003950	$<10^{-4}$	10^{-5}	10^{-5}
	$0.25M$	0.005081	0.000272 ∼ 0.004516	$<10^{-4}$	10^{-7}	10^{-5}		$0.25M$	0.005475	0.000248 ∼ 0.002492	$<10^{-4}$	10^{-6}	10^{-5}
	$0.5M$	0.004433	0.000192 ∼ 0.003234	$<10^{-4}$	10^{-6}	10^{-6}		$0.5M$	0.004772	0.000174 ∼ 0.002098	$<10^{-4}$	10^{-5}	10^{-4}
	Entire set	0.013313	0.000133 ∼ 0.001081	$<10^{-4}$	10^{-6}	10^{-4}		Entire set	0.012353	0.000365 ∼ 0.000816	$<10^{-4}$	10^{-6}	10^{-5}
000webhost	$0.05M$	0.006582	0.000160 ∼ 0.006100	$<10^{-4}$	10^{-5}	10^{-3}	Mathway	$0.05M$	0.005060	0.000200 ∼ 0.006440	0.007992	10^{-5}	10^{-5}
	$0.1M$	0.006723	0.000180 ∼ 0.005120	$<10^{-4}$	10^{-5}	10^{-3}		$0.1M$	0.004073	0.000210 ∼ 0.004590	0.001998	10^{-3}	10^{-1}
	$0.25M$	0.006016	0.000332 ∼ 0.001988	$<10^{-4}$	10^{-3}	10^{-4}		$0.25M$	0.003451	0.000252 ∼ 0.003304	$<10^{-4}$	10^{-5}	10^{-4}
	$0.5M$	0.004818	0.000234 ∼ 0.001858	$<10^{-4}$	10^{-3}	10^{-4}		$0.5M$	0.002505	0.000244 ∼ 0.002192	$<10^{-4}$	10^{-3}	10^{-4}
	Entire set	0.008472	0.000429 ∼ 0.001195	$<10^{-4}$	10^{-6}	10^{-5}		Entire set	0.004638	0.000212 ∼ 0.001349	$<10^{-4}$	10^{-5}	10^{-5}
Rockyou	$0.05M$	0.006677	0.000560 ∼ 0.010600	**0.018000**	10^{-5}	10^{-4}	Wishbone	$0.05M$	0.005762	0.000440 ∼ 0.009520	**0.016983**	10^{-5}	10^{-4}
	$0.1M$	0.008781	0.000490 ∼ 0.006980	$<10^{-4}$	10^{-5}	10^{-5}		$0.1M$	0.004427	0.000340 ∼ 0.006200	**0.010989**	10^{-5}	10^{-4}
	$0.25M$	0.011188	0.000456 ∼ 0.003676	$<10^{-4}$	10^{-5}	10^{-4}		$0.25M$	0.004552	0.000364 ∼ 0.004608	**0.019802**	10^{-5}	10^{-4}
	$0.5M$	0.013306	0.000416 ∼ 0.002798	$<10^{-4}$	10^{-6}	10^{-5}		$0.5M$	0.005141	0.000242 ∼ 0.003024	$<10^{-4}$	10^{-5}	10^{-5}
	Entire set	0.024906	0.000653 ∼ 0.001072	$<10^{-4}$	10^{-6}	10^{-4}		Entire set	0.007151	0.000346 ∼ 0.001634	$<10^{-4}$	10^{-5}	10^{-5}

† $\Delta\lambda = |\lambda - \lambda'|$ and $\Delta\alpha = |\alpha - \alpha'|$ record differences between λ and λ' as well as α and α'. The **bold** p-values are those $>\mathbf{0.01}$. Both $\Delta\lambda$ and $\Delta\alpha$ are very small, and only when the size of a subset is no greater than $0.5M$ ($1M = 10^6$, i.e., one million) can the p-value be $>\mathbf{0.01}$ to make MCA supports the model.

We name alternative distribution models based on their origins: For the four models *original in the RF system*, we name them exponential$_{\text{RF}}$, lognormal$_{\text{RF}}$, Zipf-cutoff$_{\text{RF}}$ and stretched-exponential$_{\text{RF}}$. *Accordingly, we replace RF with FF in the subscript for models that are original in the FF system and then converted to the RF system*. The fitted parameters (denoted as θ_1 and θ_2) and KS statistics of these models with eight large-scale datasets are shown in Table VII. We also present the CDF and KS statistic plots of Yahoo, Dodonew, and Wishbone datasets in Fig. 4 for an intuitive exhibition.

Table VII shows that there are three other distributions, i.e., lognormal$_{\text{RF}}$, lognormal$_{\text{FF}}$ and stretched-exponential$_{\text{RF}}$, providing comparable accuracy to the CDF-Zipf distribution. In particular, lognormal$_{\text{RF}}$ or lognormal$_{\text{FF}}$ is the most accurate on three datasets (i.e., Yahoo, Rockyou, Chegg), and so is stretched-exponential$_{\text{RF}}$ (i.e., Tianya, Mathway, Wishbone). Now, whether these comparably accurate models can pass the MCA goodness-of-fit test is crucial: If there is one model that can pass MCA, passwords are more likely to follow it than the state-of-the-art CDF-Zipf distribution. Accordingly, we need to conduct a goodness-of-fit test on alternative distributions.

C. Revisit MCA in alternative distributions

We do MCA with all eight alternative models on eight large-scale datasets, and the results show that MCA rejects *all* of them (see the p-value columns in Table VII). This brings out the question of whether MCA is suitable for large datasets.

Without loss of generality, we use MCA results of the stretched-exponential distribution in the RF system (i.e., stretched-exponential$_{\text{RF}}$) as an example, and the cases are similar for the other seven models. The fitting results (including the parameters λ and α, and the KS statistic D_{KS}) of subsets and the entire datasets are shown in Table VIII. It shows that: (1) The differences $\Delta\lambda$ (between λ and λ') and $\Delta\alpha$ (between α and α') are comparatively small (between $10^{-6}\sim10^{-3}$ with only one exception in Chegg), so there can reach a similar conclusion to Theorem 1 (which holds under CDF-Zipf) under the stretched-exponential$_{\text{RF}}$ distribution; (2) For subsets and entire datasets whose sizes are larger than $0.5M$ (i.e., 0.5 million), while some KS statistic D_{KS} values are smaller (e.g., on Yahoo, Rockyou and Chegg) than those of CDF-Zipf, they are still constantly larger than the corresponding type-1 deviations (i.e., statistical randomness) D'_{KS}, so p-values are $<10^{-4}$. It seems that, regardless of distribution models, MCA will invariably reject as long as the dataset size is large (e.g., $\geq 0.5M$).

After evaluating eight alternative models, three problems can be identified within MCA. First, like that of CDF-Zipf in Sec. III-B, for large datasets (e.g., $\geq 1M$), according to Eq. 3, type-1 statistical error of MCA (rejecting the hypothesized $\mathcal{X}$ when passwords truly follow it) is *theoretically* to be as low as 1.96% when the p-value is $<10^{-4}$, but it is actually large because MCA rejects all candidate models. One possible reason is that the condition that $P(H_0)\approx P(H_1)$ of Eq. 3 for MCA (see Sec. III-B) does not hold in practice. This is likely when H_1 is the negation of H_0, so H_1 can comprise multiple distributions (e.g., lognormal$_{\text{RF}}$ and stretched-exponential$_{\text{RF}}$) and $P(H_1) > P(H_0)$. Second, MCA generates non-negligible p-values (e.g.,$>10^{-4}$) only when dataset sizes are small (e.g., $\leq 0.25M$). That is, MCA will accept a distribution when the dataset is small, but reject when the size is large, suggesting this method is incomplete. Third, for small datasets, MCA accepts multiple distributions. For instance, when the Yahoo subset is $0.05M$, p-values are >0.01 for CDF-Zipf, lognormal$_{\text{RF}}$, lognormal$_{\text{FF}}$, stretched-exponential$_{\text{RF}}$, that is, MCA will accept all these three distributions. Because passwords are unlikely to follow multiple distributions simultaneously, type-2 statistical error (accepting a distribution when passwords do not truly follow it) is non-negligible for a small dataset.

Accordingly, we can analyze deviation metrics of the alternative distribution models, and obtain results similar to Theorems 2 and 3 (which is for CDF-Zipf, see Sec. III-C). Due to these defects, a new goodness-of-fit measure aimed to evaluate these distribution models is necessary.

D. Log-likelihood ratio test

We introduce a new goodness-of-fit measure based on the likelihood of distributions to replace the ineffective MCA. In statistics, the log-likelihood ratio test (LRT) explores the intuition that the event with the largest joint probability (i.e., likelihood) is most likely with the empirically observed data [17], [20], [40]. In practice, the log form of likelihood (i.e., log-likelihood) is often used to facilitate computation. Therefore, when both the distribution models $\mathcal{X}$ and $\mathcal{Y}$ can characterize password distribution accurately, the model with a larger log-likelihood is more likely to happen and thus be

TABLE IX: Log-likelihood ratios of alternative models against CDF-Zipf for small password subsets.†

Dataset	Dataset size	Lognormal$_{RF}$	Lognormal$_{FF}$	Stretched -exponential$_{RF}$	Dataset	Dataset size	Lognormal$_{RF}$	Lognormal$_{FF}$	Stretched -exponential$_{RF}$
Yahoo	$0.05M$	-5.11×10^{3}	-2.31×10^{3}	4.85×10^{5}	Tianya	$0.05M$	-2.33×10^{3}	-3.82×10^{3}	4.43×10^{5}
	$0.1M$	-1.00×10^{4}	-3.44×10^{3}	1.01×10^{6}		$0.1M$	-8.28×10^{3}	-1.17×10^{4}	9.09×10^{5}
	$0.25M$	-3.98×10^{4}	-2.02×10^{4}	2.58×10^{6}		$0.25M$	-3.28×10^{4}	-3.33×10^{4}	2.29×10^{6}
	$0.5M$	-1.21×10^{5}	-4.64×10^{4}	5.18×10^{6}		$0.5M$	-9.33×10^{4}	-1.01×10^{5}	4.56×10^{6}
	Entire set	-8.06×10^{7}	-3.76×10^{7}	5.54×10^{8}		Entire set	-3.56×10^{7}	-2.22×10^{7}	2.08×10^{8}
Dodonew	$0.05M$	8.62×10^{2}	-8.91×10^{2}	4.95×10^{5}	Chegg	$0.05M$	-2.10×10^{3}	-1.30×10^{3}	5.24×10^{5}
	$0.1M$	-3.28×10^{3}	-2.74×10^{3}	1.04×10^{6}		$0.1M$	1.56×10^{3}	-1.14×10^{3}	1.11×10^{6}
	$0.25M$	7.36×10^{3}	-1.48×10^{4}	2.71×10^{6}		$0.25M$	-5.79×10^{2}	-1.27×10^{4}	2.94×10^{6}
	$0.5M$	-3.49×10^{4}	-4.79×10^{4}	5.54×10^{6}		$0.5M$	-2.94×10^{4}	-1.93×10^{5}	6.09×10^{6}
	Entire set	-9.61×10^{6}	-4.19×10^{6}	1.67×10^{8}		Entire set	-6.27×10^{6}	-4.17×10^{6}	5.01×10^{8}
000webhost	$0.05M$	-8.53×10^{2}	-1.13×10^{4}	5.20×10^{5}	Mathway	$0.05M$	-1.66×10^{3}	-2.54×10^{3}	5.14×10^{5}
	$0.1M$	-1.18×10^{3}	-3.02×10^{3}	1.09×10^{6}		$0.1M$	-1.09×10^{4}	-3.17×10^{3}	1.08×10^{6}
	$0.25M$	1.17×10^{3}	-9.46×10^{4}	2.90×10^{6}		$0.25M$	-1.26×10^{4}	-1.49×10^{4}	2.83×10^{6}
	$0.5M$	-1.85×10^{4}	-2.55×10^{4}	5.99×10^{6}		$0.5M$	-3.41×10^{4}	-3.13×10^{4}	5.80×10^{6}
	Entire set	-5.90×10^{6}	-4.22×10^{6}	1.73×10^{8}		Entire set	-1.10×10^{7}	-1.71×10^{7}	1.82×10^{8}
Rockyou	$0.05M$	-9.05×10^{3}	-2.42×10^{3}	4.63×10^{5}	Wishbone	$0.05M$	-3.18×10^{3}	-9.50×10^{2}	4.94×10^{5}
	$0.1M$	-2.48×10^{4}	-1.09×10^{4}	9.35×10^{5}		$0.1M$	-4.69×10^{3}	-3.10×10^{3}	4.93×10^{5}
	$0.25M$	-6.80×10^{4}	-3.45×10^{4}	2.35×10^{6}		$0.25M$	-2.64×10^{4}	-3.75×10^{4}	2.63×10^{6}
	$0.5M$	-1.14×10^{5}	-7.71×10^{4}	4.66×10^{6}		$0.5M$	-7.71×10^{4}	-1.06×10^{5}	5.33×10^{6}
	Entire set	-2.97×10^{7}	-2.07×10^{7}	2.19×10^{8}		Entire set	-3.88×10^{6}	-4.63×10^{6}	8.90×10^{7}

† All p-values=0, and are calculated by treating $LR_i = \ln(p_{H_{1i}}) - \ln(p_{H_{0i}})$ as a random variable. Details can be seen in Clauset et al.'s work [20].

observed. This method is not only widely used in various fields (e.g., physics and biology [4], [20], [41]), but also recommended by the NIST standard of statistical methods [1].

Typically, LRT deals with the same distribution model with different parameters. For instance, in CDF-Zipf, the null hypothesis H_0 is: $C = C_0$ and $s = s_0$, and the alternative hypothesis H_1 is: $C = C_1$ and $s = s_1$ (or $C \neq C_0$ and $s \neq s_0$). Thus, the log-likelihood ratio is

$$LR = \sum_{i=1}^{N} (\ln(p_{H_{1i}}) - \ln(p_{H_{0i}})), \tag{11}$$

where N is the number of unique passwords, and $p_{H_{0i}}$ and $p_{H_{1i}}$ are the probability density functions (PDFs) of models supposed by H_0 and H_1 hypothesises. If $LR > 0$, H_1 is more likely than H_0, and vice versa.

Inspired by this idea, we set H_0 as passwords following CDF-Zipf, and H_1 as passwords following $\mathcal{X}$ (one alternative distribution model). Since all models are characterized by the parameters resulting from the GSS fitting method, we can figure out which best-fitted model is most likely. It should be noted that LRT is only needed when a model is accurate (e.g., D_{KS}<0.1), because inaccurate models (e.g., two Zipf-cutoff models, see Table VII) will be ruled out in the first place. Thus, we focus on the lognormal$_{RF}$, lognormal$_{FF}$, and stretched-exponential$_{RF}$ models.

LRT results of the subsets and entire datasets of alternative models against CDF-Zipf (using CDF-Zipf as H_0) are shown in Table IX. The results of entire datasets demonstrate that: (1) Only stretched-exponential$_{RF}$ can *constantly* and *significantly* outperform CDF-Zipf in log-likelihood; Although the other two lognormal distribution models are promising, they have significantly lower log-likelihoods than CDF-Zipf, let alone stretched-exponential$_{RF}$, so they are less likely; (2) The KS statistic of stretched-exponential$_{RF}$ is as small as 0.004638∼0.024906 (avg. 0.012459), while that of CDF-Zipf is larger and is 0.004979∼0.045357 (avg. 0.019142); (3) On six out of eight datasets, stretched-exponential$_{RF}$ is more accurate than CDF-Zipf. In all, passwords are more likely to follow the stretched-exponential$_{RF}$ distribution.

Comparison between LRT and MCA. We compare LRT with MCA, and reveal how LRT addresses the three problems identified within MCA (see Sec. IV-C). First, for Problem-1, in LRT, for each alternative model, H_1 is comprised of *one* specific distribution model rather than the negation of H_0, so the condition $P(H_0)\approx P(H_1)$ in [9] is likely to hold and Eq. 3 is applicable. Furthermore, since all calculated p-values are close to 0 regardless of dataset sizes, type-1 statistical error is also constantly close to 0. Second, for Problem-2, LRT results reveal that stretched-exponential$_{RF}$ outperforms others regardless of dataset sizes. This addresses the incompleteness problem that a distribution model will be accepted for small datasets but rejected for large ones. Third, for Problem-3, even if the dataset size is as small as $0.05M$∼$0.5M$, stretched-exponential$_{RF}$ has the largest log-likelihood across datasets. This means that LRT will not accept multiple distributions for small datasets, and type-2 statistical error is relatively low.

Discussion. We discuss CDF-Zipf, Zipf-cutoff, and stretched-exponential models. As shown in Table VI, in each system, the PDF kernels of these distributions are similar. We use stretched-exponential (denoted the variable as x) as an example, and expand the Taylor series of the exponential term $\exp(-\lambda x^{\alpha})$ in the PDF kernel as follows

$$\exp(-\lambda x^{\alpha}) = 1 - \lambda x^{\alpha} + \frac{\lambda^2}{2} x^{2\alpha} + O(x^3). \tag{12}$$

If we only take the constant term 1 in the expansion, stretched-exponential is reduced to CDF-Zipf. If we take $\alpha = 1$, the exponent $\exp(-\lambda x)$ of stretched-exponential is equal to that of Zipf-cutoff, so CDF-Zipf and Zipf-cutoff can be seen as variants of stretched-exponential.

This also partially explains why stretched-exponential is considerably more accurate than Zipf-cutoff in both coordinate systems. With an extra α, $\exp(-\lambda x^{\alpha})$ (resp. $\exp(-\lambda r^{\alpha})$) will not be too small when x (resp. r) is large, so it can adjust the Zipf term $x^{-\alpha}$ (resp. r^{s-1}) more flexibly. Numerical evidence also confirms this point: (1) For Zipf-cutoff$_{FF}$, there is $\alpha \approx 1$, which is a direct result of the very small $\exp(-\lambda x)$; (2) The Zipf-cutoff$_{RF}$ performs even worse with $\alpha \ll 0$. Thus, Zipf-cutoff models in both systems are not accurate.

Besides, the stretched-exponential$_{RF}$ model is also efficient as it only takes about 25% more time than CDF-Zipf. For instance, when fitting the 32 million Rockyou passwords on an Intel E5-2680 v4 2.4GHz CPU, the running time is 2.33 hours for stretched-exponential$_{RF}$ and 1.86 hours for CDF-Zipf.

We finally discuss the LRT method. Though it can effectively distinguish distributions, there is always potential room for improvements. On the one hand, other methods like Mann–Whitney U test may also be effective, but they

are often complicated and computationally costly, so they are less widely used in practice. On the other hand, as innovative statistical tests (e.g., [7], [8], [28], [36]) are developed, more suitable methods may emerge to address the goodness-of-fit issue on large-scale password datasets in the future.

Summary. We first investigate eight alternative distribution models in two coordinate systems, and find that three models are comparable to the state-of-the-art CDF-Zipf [54]. Second, we revisit MCA on alternative models, and find that MCA rejects all of them. Third, we introduce a new goodness-of-fit measure based on the log-likelihoods, and find that stretched-exponential in the rank-frequency coordinate system always has the largest log-likelihood. We also consider the fitting efficiency of the above stretched-exponential, and find it only costs about 25% (e.g., half an hour for a 30 million sized dataset) more time than CDF-Zipf, which is acceptable in practice. Forth, we reveal that LRT outperforms MCA in terms of minimizing statistical errors, so it is more suitable for goodness-of-fit usage on password distribution.

V. Conclusion

In this paper, we have studied the goodness-of-fit issue of password distribution in a principled approach. Particularly, we used both theories and experiments to validate the folklore of the effect of sample size, and *quantitatively* revealed that the Monte Carlo approach (MCA) is *undesirable* when the real-world password dataset is large (e.g., $\geq$0.25 million). We also studied the real-world password deviation, and used simulation to find the threshold for MCA to reject CDF-Zipf. We found a 1% random deviation is enough to reach a reject, revealing that MCA is ineffective for testing whether large-scale password datasets follow the CDF-Zipf distribution.

We further investigated eight alternative distribution models that passwords may follow. We explored these models in two different coordinate systems, and found that three models are comparably accurate. In particular, the maximum CDF deviation of the stretched-exponential model is $0.004638\sim0.024906$ (avg. 0.012459), while that of CDF-Zipf is $0.004979\sim0.045357$ (avg. 0.019142). Besides, on six out of eight datasets, stretched-exponential is more accurate than CDF-Zipf. We also revisited the MCA on all alternative distribution models, and further demonstrated its *ineffectiveness*. As a replacement, we introduced a better goodness-of-fit measure named likelihood ratio test (LRT), which supports that passwords are more likely to follow stretched-exponential. We believe this work provides a better understanding of password distribution, and facilitates the evaluation of password-related applications that involve password distribution.

References

[1] "NIST/SEMATECH e-Handbook of statistical methods," Oct. 2021, http://www.itl.nist.gov/div898/handbook/.

[2] M. Abdalla, F. Benhamouda, and P. MacKenzie, "Security of the J-PAKE password-authenticated key exchange protocol," in *Proc. IEEE S&P 2015*.

[3] L. A. Adamic, "Zipf, Power-laws, and Pareto - a ranking tutorial," *Information Dynamics Lab, HP Labs*, 2000, https://www.hpl.hp.com/research/idl/papers/ranking/ranking.html.

[4] M. Anisimova, J. P. Bielawski, and Z. Yang, "Accuracy and power of the likelihood ratio test in detecting adaptive molecular evolution," *Mol. Biol. Evol.*, vol. 18, no. 8, pp. 1585–1592, 2001.

[5] A.-L. Barabási, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek, "Evolution of the social network of scientific collaborations," *Physica A*, vol. 311, no. 3-4, pp. 590–614, 2002.

[6] E. Barker and J. Kelsey, "NIST Special Publication 800-90A Revision 1: Recommendation for random number generation using deterministic random bit generators," NIST, Reston, VA, Tech. Rep., June 2015.

[7] D. J. Benjamin, J. O. Berger, M. Johannesson, B. A. Nosek, E.-J. Wagenmakers, R. Berk, K. A. Bollen, B. Brembs, L. Brown, C. Camerer *et al.*, "Redefine statistical significance," *Nat. Hum. Behav.*, vol. 2, no. 1, pp. 6–10, 2018.

[8] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *J. R. Stat. Soc. Ser. B Methodol.*, vol. 57, no. 1, pp. 289–300, 1995.

[9] J. O. Berger, L. D. Brown, and R. L. Wolpert, "A unified conditional frequentist and Bayesian test for fixed and sequential simple hypothesis testing," *Ann. Stat.*, pp. 1787–1807, 1994.

[10] J. Blair, C. Edwards, and J. Johnson, "Rational Chebyshev approximations for the inverse of the error function," *Math. Comput*, vol. 30, no. 136, pp. 827–830, 1976.

[11] J. Blocki, A. Datta, and J. Bonneau, "Differentially private password frequency lists," in *Proc. NDSS 2016*.

[12] J. Blocki, B. Harsha, and S. Zhou, "On the economics of offline password cracking," in *Proc. IEEE S&P 2018*.

[13] J. Blocki and A. Sridhar, "Client-CASH: Protecting master passwords against offline attacks," in *Proc. ASIACCS 2016*, pp. 165–176.

[14] J. Bonneau, "The science of guessing: Analyzing an anonymized corpus of 70 million passwords," in *Proc. IEEE S&P 2012*.

[15] J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano, "Passwords and the evolution of imperfect authentication," *Comm. ACM*, vol. 58, no. 7, pp. 78–87, 2015.

[16] T. Bradley, J. Camenisch, S. Jarecki, A. Lehmann, G. Neven, and J. Xu, "Password-authenticated public-key encryption," in *Proc. ACNS 2019*.

[17] K. A. Brownlee, "Statistical theory and methodology in science and engineering," *A Wiley Publ. in Appl. Stat.*, 1965.

[18] S. A. Chaudhry, A. Irshad, K. Yahya, N. Kumar, M. Alazab, and Y. B. Zikria, "Rotating behind privacy: An improved lightweight authentication scheme for cloud-based IoT environment," *ACM Trans. Internet Technol.*, vol. 21, no. 3, pp. 1–19, 2021.

[19] Y. Cheng, C. Xu, Z. Hai, and Y. Li, "DeepMnemonic: Password mnemonic generation via deep attentive encoder-decoder model," *IEEE Trans. Depend. Secur. Comput.*, vol. 19, pp. 77–90, 2020.

[20] A. Clauset, C. R. Shalizi, and M. E. Newman, "Power-law distributions in empirical data," *SIAM Rev.*, vol. 51, no. 4, pp. 661–703, 2009.

[21] A. Das, J. Bonneau, M. Caesar, N. Borisov, and X. Wang, "The tangled web of password reuse," in *Proc. NDSS 2014*.

[22] P. Das, J. Hesse, and A. Lehmann, "DPaSE: Distributed password-authenticated symmetric encryption," *IACR Cryptol. ePrint Arch.*, p. 1443, 2020.

[23] A. C. Davison and D. V. Hinkley, *Bootstrap methods and their application*. Cambridge University Press, 1997.

[24] T. N. Dinh, H. Zhang, D. T. Nguyen, and M. T. Thai, "Cost-effective viral marketing for time-critical campaigns in large-scale social networks," *IEEE/ACM Trans. Netw.*, vol. 22, no. 6, pp. 2001–2011, 2014.

[25] R. A. Fisher, "Statistical methods for research workers," in *Breakthroughs in statistics*. Springer, 1992, pp. 66–70.

[26] S. N. Goodman, "Of p-values and Bayes: A modest proposal," *Epidemiology*, vol. 12, no. 3, pp. 295–297, 2001.

[27] Z. Huang, E. Ayday, J. Fellay, J.-P. Hubaux, and A. Juels, "GenoGuard: Protecting genomic data against brute-force attacks," in *Proc. IEEE S&P 2015*.

[28] J. P. Ioannidis, "The proposal to lower p value thresholds to. 005," *JAMA*, vol. 319, no. 14, pp. 1429–1430, 2018.

[29] H. Jeong, Z. Néda, and A.-L. Barabási, "Measuring preferential attachment in evolving networks," *Euro. Phys. Lett.*, vol. 61, no. 4, p. 567, 2003.

[30] F. Kiefer and M. Manulis, "Zero-knowledge password policy checks and verifier-based PAKE," in *Proc. ESORICS 2014*.

[31] D. Malone and K. Maher, "Investigating the distribution of password choices," in *Proc. WWW 2012*.

[32] F. Mathis, H. I. Fawaz, and M. Khamis, "Knowledge-driven biometric authentication in virtual reality," in *Proc. ACM CHI 2020*.

[33] P. Mayer, Y. Zou, F. Schaub, and A. J. Aviv, ""Now I'm a bit angry:" Individuals' awareness, perception, and responses to data breaches that affected them," in *Proc. USENIX SEC 2021*.

[34] M. Mitzenmacher, "A brief history of generative models for power law and lognormal distributions," *Internet Math.*, vol. 1, no. 2, pp. 226–251, 2004.
[35] R. Morris and K. Thompson, "Password security: A case history," *Comm. ACM*, vol. 22, no. 11, pp. 594–597, 1979.
[36] S. P. Nguyen, U. H. Pham, T. D. Nguyen, and H. T. Le, "A new method for hypothesis testing using inferential models with an application to the changepoint problem," in *Proc. IUKM 2016*.
[37] S. Oesch and S. Ruoti, "That was then, this is now: A security evaluation of password generation, storage, and autofill in browser-based password managers," in *Proc. USENIX SEC 2020*.
[38] M. L. Pao, "An empirical examination of Lotka's law," *J. Amer. Soc. Inform. Sci,*, vol. 37, no. 1, pp. 26–33, 1986.
[39] S. Pearman, J. Thomas, P. E. Naeini, H. Habib, L. Bauer, N. Christin, L. F. Cranor, S. Egelman, and A. Forget, "Let's go in for a closer look: Observing passwords in their natural habitat," in *Proc. ACM CCS 2017*.
[40] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical recipes in Pascal: The art of scientific computing*. Cambridge University Press, 1989.
[41] R. Protassov, D. A. Van Dyk, A. Connors, V. L. Kashyap, and A. Siemiginowska, "Statistics, handle with care: Detecting multiple model components with the likelihood ratio test," *Astrophys. J.*, vol. 571, no. 1, p. 545, 2002.
[42] F. Raque, M. Obaidat, K. Mahmood, M. F. Ayub, J. Ferzund, and S. A. Chaudhry, "An efficient and provably secure certificateless protocol for industrial Internet of Things," *IEEE Trans. Ind. Inform.*, 2022.
[43] R. M. Royall, "The effect of sample size on the meaning of significance tests," *Amer. Statist.*, vol. 40, no. 4, pp. 313–315, 1986.
[44] W. Rudin, *Principles of mathematical analysis*. McGraw-hill New York, 1964, vol. 3.
[45] S. Salamatian, W. Huleihel, A. Beirami, A. Cohen, and M. Médard, "Why botnets work: Distributed brute-force attacks need no synchronization," *IEEE Trans. Inform. Foren. Secur.*, vol. 14, no. 9, pp. 2288–2299, 2019.
[46] T. Sellke, M. Bayarri, and J. O. Berger, "Calibration of ρ values for testing precise null hypotheses," *Amer. Statist.*, vol. 55, no. 1, pp. 62–71, 2001.
[47] R. Shay, P. G. Kelley, S. Komanduri, M. L. Mazurek, B. Ur, T. Vidas, L. Bauer, N. Christin, and L. F. Cranor, "Correct horse battery staple: Exploring the usability of system-assigned passphrases," in *Proc. SOUPS 2012*.
[48] M. Shirvanian and S. Agrawal, "2D-2FA: A new dimension in two-factor authentication," in *Proc. ACSAC 2021*.
[49] J. Srinivas, A. K. Das, M. Wazid, and N. Kumar, "Anonymous lightweight chaotic map-based authenticated key agreement protocol for Industrial Internet of Things," *IEEE Trans. Depend. Secur. Comput.*, vol. 17, no. 6, pp. 1133–1146, 2018.
[50] J. Tan, L. Bauer, N. Christin, and L. F. Cranor, "Practical recommendations for stronger, more usable passwords combining minimum-strength, minimum-length, and blocklist requirements," in *Proc. ACM CCS 2020*.
[51] J. Tsai, R. El-Gabalawy, W. H. Sledge, S. M. Southwick, and R. H. Pietrzak, "Post-traumatic growth among veterans in the USA: Results from the national health and resilience in veterans study," *Psychol. Med.*, vol. 45, no. 1, pp. 165–179, 2015.
[52] L. Vaas, *People like using passwords way more than biometrics*, Aug. 2016, https://nakedsecurity.sophos.com/2016/08/16/people-like-using-passwords-way-more-than-biometrics/.
[53] R. Veras, C. Collins, and J. Thorpe, "On semantic patterns of passwords and their security impact," in *Proc. NDSS 2014*.
[54] D. Wang, H. Cheng, P. Wang, X. Huang, and G. Jian, "Zipf's law in passwords," *IEEE Trans. Inform. Foren. Secur.*, vol. 12, no. 11, pp. 2776–2791, 2017.
[55] D. Wang, D. He, H. Cheng, and P. Wang, "fuzzyPSM: A new password strength meter using fuzzy probabilistic context-free grammars," in *Proc. IEEE/IFIP DSN 2016*.
[56] D. Wang, P. Wang, D. He, and Y. Tian, "Birthday, name and bifacial-security: Understanding passwords of Chinese web users," in *Proc. USENIX SEC 2019*.
[57] D. Wang, Z. Zhang, P. Wang, J. Yan, and X. Huang, "Targeted online password guessing: An underestimated threat," in *Proc. ACM CCS 2016*.
[58] K. C. Wang and M. K. Reiter, "Detecting stuffing of a user's credentials at her own accounts," in *Proc. USENIX SEC 2020*.
[59] K. C. Wang and M. K. Reiter, "How to end password reuse on the web," in *Proc. NDSS 2019*.
[60] S. Wodinsky, *The 200 worst passwords of 2021 are here and oh my God*, Nov. 2021, https://gizmodo.com/the-200-worst-passwords-of-2021-are-here-and-oh-my-god-1848073946.
[61] Y. Xiao and J. Zeng, "Dynamically generate password policy via Zipf distribution," *IEEE Trans. Inform. Foren. Secur.*, vol. 17, pp. 835–848, 2022.
[62] J. Yan, A. F. Blackwell, R. J. Anderson, and A. Grant, "Password memorability and security: Empirical results." *IEEE Secur. Priv.*, vol. 2, no. 5, pp. 25–31, 2004.
[63] W. Yang, N. Li, O. Chowdhury, A. Xiong, and R. W. Proctor, "An empirical study of mnemonic sentence-based password generation strategies," in *Proc. ACM CCS 2016*.
[64] G. K. Zipf, *Human behavior and the principle of least effort: An introduction to human ecology*. Ravenio Books, 2016.

APPENDIX

A. Random numbers and golden-section-search fitting method

We introduce the conversion method [20], [40] used to generate random numbers following a given distribution. Suppose p_r is the probability density function (PDF) of the given distribution, and $u \in (0, 1]$ is a uniformly distributed random number (e.g., a standard pseudo-random number [6]).

$$p_r = p(u)\frac{du}{dr} = \frac{du}{dr}, \tag{13}$$

Integrating both sides with respect to r, we have

$$P_r = \int_r^{\infty} p_r dr = \int_u^1 du = 1 - u, \tag{14}$$

Algorithm 3: Generating random data following given distribution (THEOGEN).

Input : Parameters θ_1 and θ_2 of the given distribution $\mathcal{X}$, the dataset size $|\mathcal{DS}|$.
Output: The theoretical dataset TheoSet.

begin
  for $i = 1$ **to** $|\mathcal{DS}|$ **do**
    $u = U(0, 1]$; /* $U(0, 1]$ denotes uniformly distributed random numbers $u \in (0, 1]$. */
    $r = \lfloor P^{-1}(u) \rfloor$; /* $P^{-1}(u)$ is the inverse function of P_r of the distribution $\mathcal{X}$. */
    $\mathsf{Temp}[r] = \mathsf{Temp}[r] + 1$;
    Count frequency of r as f_r;
  for $\langle r, f \rangle \in$ Temp **do**
    TheoSet.*append*(f);
  Rank TheoSet in descending order;
Output: TheoSet.

Algorithm 4: Golden-section-search fitting (GSS).

Input : The real-world dataset RealSet, the dataset size $|\mathcal{DS}|$, and the distribution $\mathcal{X}$.
Output: KS statistic D_{KS}, the corresponding parameters θ_1, θ_2 under the given distribution $\mathcal{X}$.

begin
  N is the number of unique passwords;
  for $i = 1$ **to** *MaxIteration* **do**
    for $j = 1$ **to** *MaxIteration* **do**
      $\mathsf{TheoSet} = \text{THEOGEN}^{\mathcal{X}}(\theta_1, \theta_2, |\mathcal{DS}|)$; /* Generate a dataset with $|\mathcal{DS}|$ passwords following $\mathcal{X}$ (characterized by θ_1, θ_2). */
      $D_{KS} = \max_{1 \le r \le N} |\text{CDF}(\mathsf{TheoSet}) - \text{CDF}(\mathsf{RealSet})|$;
      /* CDF is the cumulative distribution of a dataset. */
      $(\theta_1, \theta_2) = (\text{GSS}_{1d}(\theta_1), \text{GSS}_{1d}(\theta_2))$;/* The GSS_{1d} is the ordinary one-dimensional golden-section-search. */
  Output: $(\theta_1, \theta_2, D_{KS})$;

so $r = \lfloor P^{-1}(1-u) \rfloor$, where P^{-1} is the inverse function of the cumulative distribution function (CDF). The process is shown in Alg. 3. Hence, we can generate random numbers following each distribution based on the CDF expressions in Table VI.

What's more, this algorithm also enables us to do the golden-section-search (GSS) fitting method used by Wang et al. [54], and the process is shown in Alg. 4.

B. Proof of Theorems 1, 2 and 3

Before entering Theorem 1, we first state the concepts of uniform convergence (i.e., converge uniformly) and Dirichlet's test used in examining whether a sequence of functions converges uniformly in mathematical analysis.

***Definition** 3:* A sequence of functions $g_i(x)$ $(i \in \mathbb{N})$ converges uniformly on domain D, if for every $\epsilon > 0$ there is an $N(\epsilon) \in \mathbb{N}$, such that for all $i \geq N(\epsilon)$ and all $x \in D$, one has $|g_i(x) - g(x)| < \epsilon$, and is denoted as $g_i(x) \rightrightarrows g(x)$.

***Lemma** 1:* (Dirichlet's test [44]). For the function sequences $\sum_{i=0}^{\infty} a_i(x)b_i(x)$ where $x \in D$, if the following two conditions are satisfied, then $\sum_{i=0}^{\infty} a_i(x)b_i(x)$ converges uniformly on D.

1) For each given $x \in D$, $\{a_i(x)\}_{i=0}^{\infty}$ is monotonic in respect to i, and $a_i(x) \rightrightarrows 0$.
2) The partial sum $|\sum_{i=0}^{n} b_i(x)| \leq M$ for some M for any $x \in D$ and $n \in \mathbb{N}$.

***Lemma** 2:* The error function $\mathrm{erf}(y) = 2\pi^{-1/2}\int_0^y e^{-t^2}dt$ has $\mathrm{erf}(x) = 2\Phi(\sqrt{2}x) - 1$, and its inverse $\mathrm{erf}^{-1}(x)$ function that can be expanded as follows when $x \to 1$

$$\begin{aligned}\mathrm{erf}^{-2}(x) \sim & \sum_{i=0}^{\infty} \eta^{-i} Q_i(\ln\eta) \\ = & \eta - \frac{1}{2}\ln\eta + \eta^{-1}(\frac{1}{4}\ln\eta - \frac{1}{2}) \\ & + \eta^{-2}(\frac{1}{16}\ln^2\eta - \frac{3}{8}\ln\eta + \frac{7}{8}) \\ & + \eta^{-3}(\frac{1}{48}\ln^3\eta - \frac{7}{32}\ln^2\eta + \frac{17}{16}\ln\eta - \frac{107}{48}) + ...,\end{aligned} \tag{15}$$

where $\eta = -\ln(\pi^{1/2}(1-x))$, $Q_i(\ln\eta)$ is a polynomial of degree i with $\ln(\eta)$ as the variable, and $\mathrm{erf}^{-2}(x)$ is the square of $\mathrm{erf}^{-1}(x)$. The detail can be seen in [10].

Now, we prove Theorems 1, 2 and 3.

***Theorem** 1:* Suppose C and s of RealSet and C' and s' of TheoSet satisfy $C \approx C'$ and $s' \approx s$; For each password of $f_i \geq f_b$, the estimation error $\epsilon_i = f_i - \mu_i$ follows the normal distribution $N(0, \sigma_i^2)$, and for passwords of $f_i < f_b$, $\sigma_i = 0$. In this case, the maximum of type-1 deviation (i.e., statistical randomness) $\max D'_{KS}$ decreases as $|\mathcal{DS}|$ increases, and there is $\lim_{|\mathcal{DS}|\to\infty} D'_{KS} = 0$. As a consequence, p-value decreases as $|\mathcal{DS}|$ increases, and there is $\lim_{J_0\to\infty} p\text{-value} = 0$.

***Proof** 1:* First, we determine the number of unique passwords N_b whose $f_i \geq f_b$. Since f_b is the boundary, we have $|\mathcal{DS}|p_{N_b} = f_b$ with $p_{N_b} = C \cdot s \cdot {N_b}^{s-1}$, so

$$N_b = (\frac{f_b}{|\mathcal{DS}|C \cdot s})^{-\frac{1}{1-s}} \tag{16}$$

Second, because $\epsilon_i \sim N(0, \sigma_i^2)$, there is $\xi_r = P_r - P_r^m = \sum_{i=1}^{r} \epsilon_i / |\mathcal{DS}|$ follows $N(0, \frac{\sum_{i=1}^{r}\sigma_i^2}{|\mathcal{DS}|^2})$ based on the property of the normal distribution. Moreover, considering the fact that $\sigma_i < \sqrt{\mu_i}$ when $f_i \geq f_b$ and $\sigma_i = 0$ when $f_i < f_b$, we have

$$\sqrt{\frac{\sum_{i=1}^{r}\sigma_i^2}{|\mathcal{DS}|^2}} < \sqrt{\frac{\sum_{i=1}^{r}\mu_i}{|\mathcal{DS}|^2}} = \sqrt{\frac{\sum_{i=1}^{r}\mu_i}{|\mathcal{DS}|} \cdot \frac{1}{|\mathcal{DS}|}} = \sqrt{\frac{P_r^m}{|\mathcal{DS}|}}, \tag{17}$$

where P_r^m is the *true* CDF up to rank r. Therefore the corresponding probability is,

$$P\left(D'_{KS} \leq 2a_0\sqrt{P_{N_b}^m/|\mathcal{DS}|}\right) > 2(1-\Phi(a_0)) \tag{18}$$

holds. Thus, for *the largest order statistic* $\max_r |\xi_r|$, we have

$$\begin{aligned} & P\left(\max_r |\xi_r| \leq a_0\sqrt{P_r^m/|\mathcal{DS}|}\right) \\ & = 1 - P(\exists r : |\xi_r| > a_0\sqrt{P_r^m/|\mathcal{DS}|}) \\ & \geq 1 - \sum_{r=1}^{N_b}\left(1 - P(|\xi_r| \leq a_0\sqrt{P_r^m/|\mathcal{DS}|})\right) \\ & \geq 2N_b(\Phi(a_0) - 1) + 1. \end{aligned} \tag{19}$$

Based on this result, we study type-1 deviation D'_{KS} (i.e., statistical randomness) resulting from fitting TheoSet. As $C' \approx C$, $s' \approx s$, D'_{KS} is the *maximum* CDF deviation between two *randomly* generated TheoSets with CDFs $P_r^{(1)}$ and $P_r^{(2)}$. Thus, $D'_{KS} = \max_r |P_r^{(1)} - P_r^{(2)}| \leq 2\max_r |\xi_r|$ and there is

$$\begin{aligned} & P\left(D'_{KS} \leq 2a_0\sqrt{P_r^m/|\mathcal{DS}|}\right) \\ & > P\left(\max_r |\xi_r| \leq a_0\sqrt{P_r^m/|\mathcal{DS}|}\right) \\ & > 2N_b(\Phi(a_0) - 1) + 1. \end{aligned} \tag{20}$$

Set $\alpha = 2N_b(\Phi(a_0) - 1) + 1$, $\max D'_{KS}$ has

$$\begin{aligned} \max D'_{KS} &= 2\sqrt{P_{N_b}^m/|\mathcal{DS}|} \cdot \Phi^{-1}(1 - \frac{1-\alpha}{2N_b}) \\ &= 2\sqrt{2P_{N_b}^m/|\mathcal{DS}|} \cdot \mathrm{erf}^{-1}\left(1 - (1-\alpha)(\frac{f_b}{|\mathcal{DS}|Cs})^{\frac{1}{1-s}}\right), \end{aligned} \tag{21}$$

where $\Phi^{-1}(x) = \mathrm{erf}^{-1}(2x-1)$ based on their definitions. With this, we will prove that $\lim_{|\mathcal{DS}|\to\infty} \max D'_{KS} = 0$.

First, as given in Lemma 2, we take $x = (1 - (1 - \alpha)(\frac{f_b}{|\mathcal{DS}|Cs})^{\frac{1}{1-s}})$ and $\eta = -\ln(\pi^{1/2}(1-x))$, so when $|\mathcal{DS}| \to \infty$, there have $x \to 1$ and $\eta \to \infty$. Second, we let $a_i(\eta) = \eta^{-(i-1)}$ and $b_i(\eta) = \eta^{-1}Q_i(\ln\eta)$ for $i \geq 1$. We now prove that $\lim_{\eta\to\infty} \frac{\ln^i\eta}{\eta} = 0$ to show that $\lim_{\eta\to\infty} b_i(\eta) = 0$ for any $i \geq 1$. First, when $k = 1$, $\lim_{\eta\to\infty} \frac{\ln\eta}{\eta} = \lim_{\eta\to\infty} \frac{1}{\eta} = 0$ (L'Hôpital's rule). By the mathematical induction, we suppose $\lim_{\eta\to\infty} \frac{\ln^i\eta}{\eta} = 0$, and when $k = i+1$ there is

$$\lim_{\eta\to\infty} \frac{\ln^{k+1}\eta}{\eta} = \lim_{\eta\to\infty} \frac{(\ln^{i+1}\eta)'}{\eta'} = (i+1)\lim_{\eta\to\infty} \frac{\ln^i\eta}{\eta} = 0. \tag{22}$$

Hence, $\lim_{\eta\to\infty} \frac{\ln^i\eta}{\eta} = 0$ for any $i \geq 1$. Since $b_i(\eta) = \sum_{k=0}^{i} c_k \frac{\ln^k(\eta)}{\eta}$ ($c_0, c_1, ...c_i$ are coefficients), $\lim_{\eta\to\infty} b_i(\eta) = 0$ and $\lim_{\eta\to\infty} |\sum_{i=0}^{n} b_i(\eta)| = 0$ for any given $n \geq 1$. This means there exists η_0 and M, for any $\eta > \eta_0$ and $n \geq 1$, $|\sum_{i=0}^{n} b_i(\eta)| < M$. Besides, since $a_i(\eta) = \eta^{-(i-1)} \rightrightarrows 0$ when $\eta \to \infty$ and $\{a_i(\eta)\}_{i=0}^{\infty}$ decreases monotonically, $\sum_{i=1}^{\infty} \eta^{-i}Q_i(\ln\eta) = \sum_{i=1}^{\infty} a_i(\eta)b_i(\eta)$ converges uniformly (see Lemma 1). Therefore, the limit operation and the infinite summation operation are commutative, i.e.,

$$\lim_{\eta\to\infty} \sum_{i=1}^{\infty} \eta^{-i}Q_i(\ln\eta) = \sum_{i=1}^{\infty} \lim_{\eta\to\infty} \eta^{-i}Q_i(\ln\eta) = \sum_{i=1}^{\infty} 0 = 0, \tag{23}$$

so $\max D'_{KS}$ depends on the first two terms of Eq. 15 i.e.,

$$\lim_{|\mathcal{DS}|\to\infty} \sqrt{(\eta - \frac{1}{2}\ln\eta)/|\mathcal{DS}|}. \tag{24}$$

Since $\eta = -\ln(\pi^{1/2}((1-\alpha))(\frac{f_b}{Cs|\mathcal{DS}|})^{\frac{1}{1-s}})$, Eq. 24 is proved to converges to 0 according to L'Hôpital's rule. As a result, there is $\lim_{|\mathcal{DS}|\to\infty} \max D'_{KS} = 0$ and consequently

$\lim_{|\mathcal{DS}|\to\infty} D'_{KS} = 0$ based on the squeeze theorem. As a consequence, p-value= $(\#\{D'_{KS_j}|D'_{KS_j} > D_{KS}, 1 \le j \le J_0\}+1)/(J_0+1)$ (see Line 6 in Alg. 1) decreases as $|\mathcal{DS}|$ increases, and becomes $1/(J_0+1)$ when $J_0 \to \infty$.

***Theorem** 2:* In the absolute deviation metric, if passwords in SmuSet are deviated as $pdv_i^s = \hat{\delta}_{[i_1,i_2]} \cdot k_A$ for $i \in [i_1, i_2]$, and the deviation degree k_A satisfies $0 \le k_A \le |\hat{pdv}_{[i_1,i_2]}|$, then the maximum $\max D_{KS}^s$ increases as k_A increases.

***Proof** 2:* As noted in Sec. III-C, the supereum of D'_{KS} has $\max D_{KS}^s = |\mathsf{CDF(SmuSet)} - \mathsf{CDF(TheoSet)}|$, that is,

$$\max D_{KS}^s = \begin{cases} \max\limits_{1\le r<r_1} \frac{k_A P_r P_{[i_1,i_2]}}{1+\hat{\delta}\cdot k_A P_{[i_1,i_2]}} & 1 \le r < r_1 \\ \max\limits_{r_1\le r<r_2} \frac{k_A P_r (1-P_{[i_1,i_2]})}{1+\hat{\delta}\cdot k_A P_{[i_1,i_2]}} & r_1 \le r < r_2 \\ \max\limits_{r_2\le r\le N} \frac{k_A P_{[i_1,i_2]}(1-P_r)}{1+\hat{\delta}\cdot k_A P_{[i_1,i_2]}} & r_2 \le r \le N. \end{cases} \tag{25}$$

where $P_r = \sum_{i=1}^{r} f_i$, $P_{[i_1,i_2]} = \sum_{i_1}^{i_2} p_i$ and $\hat{\delta} = \hat{\delta}_{[i_1,i_2]}$. We can see that: (1) D_{KS}^s increases as r increases when $1 \le r < r_2$, (2) and decreases as r increases when $r_2 \le r \le N$, so D_{KS}^s gets its maximum at $r = r_2$ and has

$$\max D_{KS}^s = \frac{k_A P_{[i_1,i_2]}(1-P_{[i_1,i_2]})}{1+\hat{\delta} k_A P_{[i_1,i_2]}}. \tag{26}$$

Suppose i_1 is fixed, we study the monotonicity of $\max D_{KS}^s$ against i_2 and k_A, we have the partial derivatives as follows.

$$\begin{cases} \frac{\partial \max D_{KS}^s}{\partial i_2} = \frac{k_A(-\hat{\delta} k_A \cdot P_{[i_1,i_2]}^2 - 2k_A \cdot P_{[i_1,i_2]} + 1)}{(1+\hat{\delta} k_A P_{[i_1,i_2]})^2} \\ \frac{\partial \max D_{KS}^s}{\partial k_A} = \frac{P_{[i_1,i_2]}(1-P_{[i_1,i_2]})}{(1+\hat{\delta} k_A P_{[i_1,i_2]})^2}. \end{cases} \tag{27}$$

On the one hand, $\frac{\partial \max D_{KS}^s}{\partial k_A} > 0$ always holds. On the other hand, to make $\frac{\partial \max D_{KS}^s}{\partial i_2} > 0$, i.e., there needs

$$-\hat{\delta} k_A \cdot P_{[i_1,i_2]}^2 - 2P_{[i_1,i_2]} + 1 > 0. \tag{28}$$

If $\hat{\delta} = 1$, for any k_A there is

$$\frac{-1-\sqrt{1+k_A}}{k_A} < P_{[i_1,i_2]} < \frac{-1+\sqrt{1+k_A}}{k_A}. \tag{29}$$

To make Eq. 28 hold, $P_{[i_1,i_2]} < \inf \frac{-1+\sqrt{1+k_A}}{k_A} \approx 0.41$, which is satisfied for our considered passwords (e.g., the top 100 unique passwords). In this case, $\max D_{KS}$ increases with i_2 increases. Otherwise, if the sign is negative, i.e., $\hat{\delta} = -1$, there needs $P_{[i_1,i_2]} < \inf \frac{1-\sqrt{1-k_A}}{k_A} < \frac{1}{2}$ or $P_{[i_1,i_2]} > \sup \frac{1+\sqrt{1-k_A}}{k_A} \to \infty$ to make Eq. 28 hold, which is also satisfied. In summary, $\max D_{KS}^s$ increases as i_2 increases whenever $\hat{\delta} = 1$ or -1.

***Theorem** 3:* In the relative deviation metric, if passwords are deviated as $pdv_i^s = \hat{\delta}_i \cdot |\hat{pdv}_i| k_R$ for $i \in [1, N_0]$ and $0 < k_R \le 1$, then the maximum $\max D_{KS}^s$ increases as k_R increases.

***Proof** 3:* Similar to the proof of the absolute point-wise deviation, the maximum $\max D_{KS}^s$ can be expressed as,

$$\max D_{KS}^s = \begin{cases} \max\limits_{1\le r<N_0} |\frac{(1-P_r)W_r + P_r W_{[r+1,N_0]}}{1+W_{N_0}}| k_R & 1 \le r < N_0 \\ \max\limits_{N_0\le r\le N} \frac{|W_r|(1-P_r)}{1+W_{N_0}} k_R & N_0 \le r \le N, \end{cases} \tag{30}$$

where $P_r = \sum_{i=1}^{r} p_i$ is the CDF in TheoSet, $W_r = \sum_{i=1}^{r} p_i \cdot pdv_i$ and $W_{[r+1,N_0]} = \sum_{i=r+1}^{N_0} p_i \cdot pdv_i$. Thus, D_{KS}^s increases as k_R increases. However, since the sign of $\hat{pdv}_i$ is not fixed, we cannot know where $\max D_{KS}^s$ gets its maximum.

C. Conversion of distributions

In this section, we use the conversion of power-law (i.e., Zipf) in Adamic's work [3] to show how to convert distribution models from the frequency-frequency (denoted as FF) to rank-frequency (denoted as RF) coordinate systems.

***Proposition** 1:* The power-law distribution with PDF $p(x) = (\alpha-1)x^{-\alpha}$ in the FF system can be converted to the RF system with PDF $p_r \propto r^{s-1}$, where $s = \frac{2-\alpha}{1-\alpha}$.

***Proof** 4:* On the one hand, in the FF system, the complementary CDF records the probability that a random variable X is larger than a given number x can be expressed as,

$$P(X \ge x) = \int_x^{\infty} p(x)dx = x^{-\alpha+1}. \tag{31}$$

On the other hand, in the RF system, if the password PW occurring x times is with the rank r, the event $X \ge x$ can be interpreted as *the proportion of unique passwords whose ranks are no more than the rank* r. Hence, there is $P(X \ge x) = r/N$ (N is the number of unique passwords). Since $x = |\mathcal{DS}|p_r$, ($|\mathcal{DS}|$ is the dataset size and p_r is the PDF of the r-th password), we have

$$P(X \ge |\mathcal{DS}|p_r) = x^{-\alpha+1} = r/N. \tag{32}$$

In this way, $p_r \propto r^{\frac{1}{1-\alpha}}$, and the CDF $P_r \propto r^{\frac{2-\alpha}{1-\alpha}}$. Therefore, power-law in the FF system can be converted into CDF-Zipf in the RF system, and they are equivalent.

Zhenduo Hou received his B.S. degree in mathematics from the Sichuan University, Chengdu, P. R. China, in June. 2015. He is currently working toward the Ph.D. degree from the School of Mathematical Sciences, Peking University, Beijing, P. R. China. He has received the National Scholarship of China, Teaching Assistance Fellowship of Peking University. His research interests include applied cryptography and password-based authentication.

Ding Wang received his Ph.D. degree in Information Security at Peking University in 2017. Currently, he is a Full Professor at Nankai University. As the first author (or corresponding author), he has published more than 80 papers at venues like IEEE S&P, ACM CCS, NDSS, USENIX Security, IEEE TDSC and IEEE TIFS. His research has been reported by over 200 medias like Daily Mail, Forbes, IEEE Spectrum and Communications of the ACM, appeared in the Elsevier 2017 "Article Selection Celebrating Computer Science Research in China", and resulted in the revision of the authentication guideline NIST SP800-63-2. He has been involved in the community as a PC Chair/TPC member for over 60 international conferences such as ACM CCS 2022, NDSS 2023, PETS 2023/2022, ACSAC 2020-2022, ACM AsiaCCS 2022/2021, ICICS 2018-2022, and SPNCE 2020-2022. He has received the "ACM China Outstanding Doctoral Dissertation Award", the Best Paper Award at INSCRYPT 2018, the Outstanding Youth Award of China Association for Cryptologic Research, and the First Prize of Natural Science Award of Ministry of Education. His research interests focus on passwords, authentication and provable security.