Journal of Machine Learning Research 24 (2023) 1-80 Submitted 11/22; Revised 8/23; Published 12/23

# Conformal Frequency Estimation using Discrete Sketched Data with Coverage for Distinct Queries

**Matteo Sesia** sesia@marshall.usc.edu
*Departments of Data Sciences and Operations, and of Computer Science*
*University of Southern California*
*Los Angeles, CA 90089, USA*

**Stefano Favaro** stefano.favaro@unito.it
*Department of Economics and Statistics*
*University of Torino and Collegio Carlo Alberto*
*Piazza Arbarello 8, Torino, Italy*

**Edgar Dobriban** dobriban@wharton.upenn.edu
*Departments of Statistics and Data Science, and of Computer and Information Science*
*University of Pennsylvania*
*Philadelphia, PA 19104, USA*

**Editor:** Michael Mahoney

## Abstract

This paper develops conformal inference methods to construct a confidence interval for the frequency of a queried object in a very large discrete data set, based on a sketch with a lower memory footprint. This approach requires no knowledge of the data distribution and can be combined with any sketching algorithm, including but not limited to the renowned count-min sketch, the count-sketch, and variations thereof. After explaining how to achieve marginal coverage for exchangeable random queries, we extend our solution to provide stronger inferences that can account for the discreteness of the data and for heterogeneous query frequencies, increasing also robustness to possible distribution shifts. These results are facilitated by a novel conformal calibration technique that guarantees valid coverage for a large fraction of distinct random queries. Finally, we show our methods have improved empirical performance compared to existing frequentist and Bayesian alternatives in simulations as well as in examples of text and SARS-CoV-2 DNA data.

**Keywords:** conformal inference, discrete data, distribution shifts, sketching, uncertainty

## 1. Introduction

### 1.1 Estimating Frequencies from Sketched Data

Estimating the frequency of a queried object given a lossy reduced representation, or *sketch*, of a large discrete data set is a classical problem (e.g., Misra and Gries, 1982; Charikar et al., 2002, etc). This task is relevant in diverse fields including machine learning (Shi et al., 2009), cybersecurity (Schechter et al., 2010), natural language processing (Goyal et al., 2012), privacy (Cormode et al., 2018), and biology (Zhang et al., 2014). For example, in biology, researchers may want to efficiently count the occurrences of a contiguous sequence of nucleotides within a large DNA database, as that can help identify common motifs that are associated with evolutionary relatedness between different organisms or are involved in important regulatory processes (Saavedra et al., 2020).

Sketching tends to be motivated either by memory limitations, as large numbers of distinct symbols may otherwise be computationally expensive to analyze (Zhang et al., 2014), or by privacy constraints when dealing with sensitive data (Kockan et al., 2020). Several sketching algorithms can provide compressed data representations that enable accurate approximations of the frequency of any object (Cormode and Yi, 2020). Classical approaches are based on random hashing (Cormode and Yi, 2020), but some recent works have proposed more sophisticated machine learning-driven algorithms that can automatically adapt to the features of the data distribution in order to optimize the data compression (Hsu et al., 2019; Jiang et al., 2019; Aamand et al., 2019; Bertsimas and Digalakis, 2021).

An important statistical problem in the context of sketching is to quantify the uncertainty of frequency queries, as exact recovery of the latter is typically unfeasible due to some loss of information during the data compression. Prior works took a number of very different routes to address this topic, ranging from data-conditional and Bayesian methods to the bootstrap (Cormode and Yi, 2020; Ting, 2018; Cai et al., 2018; Dolera et al., 2021). This paper presents a novel conformal inference method (Vovk et al., 2005). As we will explain, our approach is principled and offers some notable advantages, starting from the ability to obtain informative inferences without any parametric assumptions about the distribution of the sketched data. Further, a key strength of our approach is that it can provide rigorous uncertainty estimates for any sketching algorithm, including the classical count-min sketch (CMS) (Cormode and Muthukrishnan, 2005), its non-linear variations (Estan and Varghese, 2002), the count-sketch (CS) (Charikar et al., 2002), and even more complex learning-based techniques (Bertsimas and Digalakis, 2021). As we shall see, different sketching algorithms can lead to more or less accurate frequency queries for different types of data, and therefore the flexibility of our methods will be practically useful.

After reformulating the problem so that standard split conformal inference can be applied, developing our methodology requires overcoming several challenges. First, standard conformal inference techniques provide relatively weak statistical guarantees, which are less satisfactory than usual in the context of answering frequency queries about discrete data. Indeed, if some objects in the data are much more frequent than others, standard statistical coverage guarantees can be satisfied even by meaningless inferences that are only valid for the most common queries. We address this limitation by proposing two methodological improvements that provide conformal inferences whose validity holds separately for queries with different frequencies, and for all distinct objects in a possibly large set of queries. Further, we prove that our methods are more robust to distribution shifts compared to standard conformal inferences, which rely on the relatively strong assumption of data exchangeability.

## 1.2 Problem Statement and Preview of our Contributions

We now present a simplified version of our problem statement and data observation model; see Section 3 for the complete version. Consider $m$ data points $Z_1, \dots, Z_m \in \mathscr{Z}$, taking

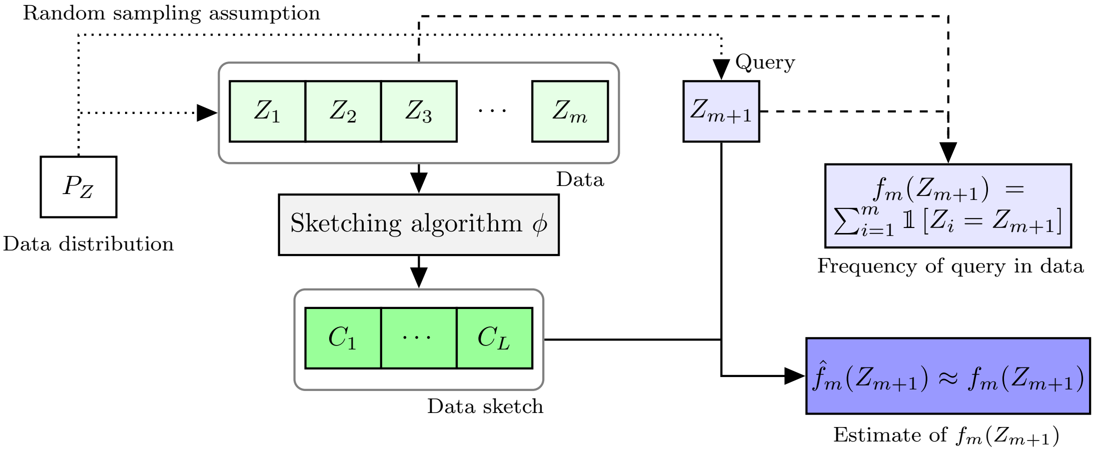
Random sampling assumption
$Z_1$
$Z_2$
$Z_3$
$\cdots$
$Z_m$
Data
$P_Z$
Data distribution
Sketching algorithm $\phi$
$C_1$
$\cdots$
$C_L$
Data sketch
Query
$Z_{m+1}$
$f_m(Z_{m+1}) = \sum_{i=1}^m \mathbb{1}[Z_i = Z_{m+1}]$
Frequency of query in data
$\hat{f}_m(Z_{m+1}) \approx f_m(Z_{m+1})$
Estimate of $f_m(Z_{m+1})$

Figure 1: Schematic visualization of the problem of estimating the empirical frequency of a queried object in a large data set, given a sketched representation of the latter.

values in a discrete and possibly infinite dictionary $\mathscr{X}$. We consider the setting where $m$ is very large, and $\mathscr{X}$ is possibly also large; thus exact computations with $Z_1, \ldots, Z_m$ are infeasible. Instead, the data are processed via an arbitrary *sketching* function $\phi : \mathscr{X}^m \to \mathcal{C}$ that produces a reduced representation of these data with lower memory footprint, where $\mathcal{C}$ consists for instance of $L$ discrete counters, so that $\mathcal{C} = \mathbb{N}^L$ with $L \ll m$. A well-known example of $\phi$ is the CMS (Cormode and Muthukrishnan, 2005), reviewed in Appendix A. The methods developed in this paper can be applied in combination with the CMS or with any other sketching function. However, the choice of $\phi$ is important in practice because it affects the efficiency of the data compression and the informativeness of our inferences, as it will become clear in Sections 6–7.

In general, our target of inference is the number of occurrences (or empirical frequency) of a given object (or query) $z \in \mathscr{X}$ in the data set $Z_1, \ldots, Z_m$:

$$f_m(z) := \sum_{i=1}^{m} \mathbb{1}\left[Z_i = z\right]. \tag{1}$$

Of course, since $Z_1, \ldots, Z_m$ are not available for direct computations, the exact value of $f_m(z)$ is not known. Instead, we aim to approximate these values for an appropriate $z$ using the sketch. Specifically, we seek an informative *confidence interval* for $f_m(z)$ that enjoys precise statistical guarantees in finite samples, as previewed next. As a starting point, we assume that the query, $z = Z_{m+1}$, is a random draw from some distribution $P_Z$, sampled exchangeably with $Z_1, \ldots, Z_m$. See Figure 1 for a schematic visualization of this problem.

The exchangeability of $(Z_1, \ldots, Z_{m+1})$, which will be relaxed later in the paper, imposes additional conditions compared to some classical analyses of sketching algorithms (Cormode and Yi, 2020). Such analyses typically treat the data as arbitrary—and thus can also handle non-stationary streams or adversarial cases. However, we believe our exchangeability condition is often realistic, for instance in applications where the data are processed in a random order; see Sections 7 and 8 for examples. Treating the data as an approximately i.i.d. sample from some distribution has been suggested before in the context of sketching (Ting, 2018; Cai et al., 2018; Aamand et al., 2019; Dolera et al., 2021), but our perspective involves key novelties. First, we assume only exchangeability, not independence. Second, we allow $P_Z$ to be arbitrary and unknown. Third, our results apply to any sketching algorithm.

Section 2 reviews the relevant conformal inference background. Then, Section 3 connects conformal inference to our problem and explains how to construct a confidence interval[1] $[\hat{L}_{m,\alpha}(Z_{m+1}), \hat{U}_{m,\alpha}(Z_{m+1})]$ for $f_m(Z_{m+1})$ with guaranteed *marginal coverage*,

$$\mathbb{P}\left[\hat{L}_{m,\alpha}(Z_{m+1}) \leq f_m(Z_{m+1}) \leq \hat{U}_{m,\alpha}(Z_{m+1})\right] \geq 1-\alpha, \tag{2}$$

at the desired level $\alpha \in (0,1)$. Such a coverage property is called *marginal* because it involves a probability taken with respect to the randomness in both the data and the query. Its interpretation is as follows: the confidence interval will cover $f_m(Z_{m+1})$ for at least a fraction $1-\alpha$ of data points $Z_1, \ldots, Z_m$ and future queries $Z_{m+1}$.

Marginal coverage is not trivial to achieve with a reasonably short interval, but it is also not fully satisfactory because our problem involves discrete data that are likely to include many repeated observations of the same objects. Unfortunately, inferences satisfying (2) are not necessarily reliable for a sufficient proportion of *distinct* or *unique* queries, which is what we would ideally like to guarantee. To the contrary, confidence intervals with marginal coverage are likely to have lower coverage for rarer queries, as illustrated by the following thought experiment. Imagine a distribution $P_Z$ with support on $\mathscr{Z} = \{0, 1, 2, \ldots, 10^{100}\}$, such that $\mathbb{P}[Z_i = 0] = 0.95$ and $\mathbb{P}[Z_i = z] = 0.05/(|\mathscr{Z}| - 1)$ for all $z \in \mathscr{Z} \setminus \{0\}$ and $i \geq 1$. Marginal coverage at level 95% would be satisfied even by a non-informative confidence interval that always contains the true frequency for a new query if $Z_{m+1} = 0$ and is empty otherwise. However, those inferences are incorrect for all but one possible query. This issue motivates the development of methods with stronger coverage guarantees.

In Section 4, we begin to address the limitations of marginal coverage by presenting a method for constructing confidence intervals that are valid for both rarer and more common random queries, taking inspiration from Mondrian conformal inference for classification (Vovk et al., 2003). Section 5 extends these ideas by developing and studying a novel construction of conformal confidence intervals with guaranteed coverage for a large fraction of distinct/unique queries in a possibly redundant test set. This method is related to the works of Dunn et al. (2022) and Park et al. (2022) on conformal inference for hierarchical models and meta-learning, but the specific notion of coverage proposed here had not been investigated before. Coverage for a large fraction of distinct queries implies that less frequent queries are given a higher weight. For instance, the example above, we expect that out of $M = 1000$ test examples 950 are equal to zero and the others are all distinct. Then, covering 95% of the uniques means that we expect to cover approximately $0.95 \cdot 951 \approx 903$ distinct queries. Clearly, this is more informative than an interval that covers only zero.

Exchangeability has a broad scope, and in certain cases it can be ensured by permuting the data—as in the experiments described in this paper. However, in practice, when

[1] Since $f_m(Z_{m+1})$ is also random, it is technically speaking a prediction interval, not a confidence interval. However, we still refer to it as a confidence interval to keep the terminology consistent with prior work.

the data come from a real-time stream—such as a sensor monitoring the weather, internet traffic, etc.—systematic distribution shifts can occur that make test data dissimilar from training data. Motivated by this problem, we will show that our proposed method also leads to increased robustness to distribution shifts, which allows some relaxation of the exchangeability assumptions and thus broadens the relevance of our results to more applications, possibly to online data streams (Cao et al., 2023).

Finally, Sections 6–7 present several experiments and illustrations of our methods, using both synthetic data from realistic power-law distributions and two empirical data examples. The latter concern 16-mers in SARS-CoV-2 DNA sequences and 2-grams in English literature. We consider the classical CMS (Cormode and Muthukrishnan, 2005), the CMS-CU (Estan and Varghese, 2002), the CS (Charikar et al., 2002), and non-random sketches based on data-driven hash functions (Bertsimas and Digalakis, 2021). We compare our methods, according to different performances metrics, to existing uncertainty estimation techniques developed for CMS sketches, including bootstrap and Bayesian approaches (Cormode and Yi, 2020; Ting, 2018; Cai et al., 2018; Dolera et al., 2021). In addition to being more flexible, as we are not limited to working with the CMS, our methods tend to outperform the existing benchmarks even when the latter are applicable, producing shorter confidence intervals with more consistent coverage. Further, we verify that our method aiming for coverage of unique elements has a higher robustness to distribution shifts compared to the simpler approach targeting marginal coverage. Additional experiments are discussed in the appendix. Section 8 concludes with a discussion and some ideas for future work.

### 1.3 Related Work

There exist many algorithms for computing approximate frequency queries given a reduced-memory sketch; some are based on random hashing (Fan et al., 2000; Goyal and Daumé, 2011; Pitel and Fouquier, 2015; Cormode and Yi, 2020), while others may involve complex learning algorithms (Bertsimas and Digalakis, 2021). Several works have also studied the problem of quantifying uncertainty in this context, but we are the first to propose a conformal inference approach that is not limited to a specific sketching algorithm. In fact, to the best of our knowledge, the related prior research has focused on the CMS algorithm (Cormode and Muthukrishnan, 2005). The classical uncertainty estimation strategies treated the data as fixed and leveraged only the randomness in the hash functions of the CMS (Cormode and Muthukrishnan, 2005), which we review in Appendix A. While that approach can lead to rigorous confidence bounds for the unknown empirical frequencies under minimal assumptions, the results are often too conservative to be practically useful (Ting, 2018).

This is why more recent works treated the data as random and either derived frequentist inferences using re-sampling techniques (Ting, 2018) or calculated a Bayesian posterior distribution for the frequency of the queried object starting from a prior model for the sketched data (Cai et al., 2018; Dolera et al., 2021; Beraha and Favaro, 2023). Our work is closer to Ting (2018), as we seek frequentist probabilistic guarantees while treating the data as random, but our solution is very different. The method of Ting (2018) is limited to the CMS, whereas we use conformal inference and can handle any sketching algorithm, including non-linear and learning-based ones (Estan and Varghese, 2002; Hsu et al., 2019; Aamand et al., 2019; Bertsimas and Digalakis, 2021). Such flexibility is useful because different sketching algorithms may allow more efficient data compression and more accurate frequency estimates depending on the data distribution (Aamand et al., 2019).

Conformal inference was pioneered by Vovk and collaborators (Saunders et al., 1999; Vovk et al., 2005) and brought to the statistics spotlight by works such as Lei et al. (2013); Lei and Wasserman (2014); Lei et al. (2018). Although primarily conceived for supervised prediction (Vovk et al., 2009; Vovk, 2015; Lei and Wasserman, 2014; Romano et al., 2019; Izbicki et al., 2019; Park et al., 2021; Qiu et al., 2023), conformal inference has found other applications including outlier and anomaly detection (Bates et al., 2023; Kaur et al., 2022; Li et al., 2022; Liang et al., 2022), causal inference (Lei et al., 2021, e.g.,), and survival analysis (Candès et al., 2023). We mention here that the ideas in conformal prediction have deep roots in statistics, dating back at least to the pioneering works of Wilks (1941), Wald (1943), Scheffe and Tukey (1945), and Tukey (1947, 1948); see also Geisser (2017).

### 1.4 Relation to Shorter Conference Paper

The potential of conformal inference in sketching remained untapped before the shorter version of this work (Sesia and Favaro, 2022), which appeared in the proceedings of the NeurIPS 2022 conference. This extended manuscript contains novel methods and several original theoretical results, in Section 5, studying the construction of confidence intervals with valid coverage for a large fraction of distinct queries. This is stronger and more challenging guarantee compared to marginal coverage, and it is useful because it leads to more easily interpretable inferences when the data are discrete and may involve many repeated observations. Further, we will show that the methodological extensions introduced in this paper improve the robustness to distribution shifts and other possible violations of the data exchangeability assumption (Tibshirani et al., 2019; Barber et al., 2023), which could be relevant for example when sketching streaming data (Cao et al., 2023). Finally, Sections 6–7 of this manuscript contain several additional numerical results, and the whole paper has been re-organized to provide a more general description of the proposed methodology that better highlights its general applicability in combination with any sketching algorithm.

## 2. Preliminaries on Conformal Prediction

Consider *supervised learning*, with data pairs $(X_i, Y_i)$ where $X_i$ are a vector of *features* for the $i$-th observation and $Y_i$ are the corresponding *outcome* or *label*, which may be continuous- or discrete-valued. The usual goal in supervised learning is to use $(X_1, Y_1), \dots, (X_n, Y_n)$ to learn a predictor of an unseen label $Y_{n+1}$ using a new observation with features $X_{n+1}$. Related to this, conformal prediction can be used to construct a prediction interval $[\hat{L}_{n,\alpha}$ $(X_{n+1}), \hat{U}_{n,\alpha}(X_{n+1})]$ with guaranteed marginal coverage,

$$\mathbb{P}[\hat{L}_{n,\alpha}(X_{n+1}) \leq Y_{n+1} \leq \hat{U}_{n,\alpha}(X_{n+1})] \geq 1 - \alpha,$$

for any fixed $\alpha \in (0, 1)$, assuming that $(X_1, Y_1), \dots, (X_{n+1}, Y_{n+1})$ is an exchangeable random sample from some unknown distribution over $(X, Y)$. Conformal prediction can leverage supervised learning methods to approximately reconstruct the relation between $X$ and $Y$, capturing it in $\hat{L}_{n,\alpha}, \hat{U}_{n,\alpha}$, and it automatically calibrates such prediction interval to achieve marginal coverage. While it is sufficient to focus on conformal intervals in this paper, similar techniques can also be used to construct more general prediction sets (e.g., Vovk et al., 2005; Romano et al., 2020b; Angelopoulos et al., 2021, etc).

A simple version of conformal prediction—known as *split* or *inductive* conformal prediction (Papadopoulos et al., 2002; Lei et al., 2018)—begins by randomly splitting the observations into two disjoint subsets: a *training set* and a *calibration set.* The first $n^{\text{train}} \in \{1, \dots, n\}$ data points are used as the training set, to fit a machine learning model for predicting $Y$ given $X$; e.g., a neural network or a random forest. The out-of-sample predictive accuracy of this model is then measured in terms of a *conformity score* for each of the $n - n^{\text{train}}$ held-out data points in the calibration set. In combination with the model learned from the training data, the quantiles of the empirical distribution of these scores are used to construct prediction intervals for future test points as a function of $X_{n+1}$. As detailed shortly, these intervals are guaranteed to cover $Y_{n+1}$ with probability at least $1-\alpha$, treating all data as random. Importantly, the coverage holds in finite samples, regardless of the accuracy of the predictive model, as long as $X_{n+1}$ is exchangeable with the held-out data points. It is unnecessary for the training data to be also exchangeable, as these may be viewed as fixed.

One perspective on conformal prediction is to construct a *nested sequence* (Vovk et al., 2005; Gupta et al., 2022) of prediction intervals $[\hat{L}_{n,\alpha}(x;t), \hat{U}_{n,\alpha}(x;t)]$, indexed by $t \in \mathcal{T} \subseteq \mathbb{R}$ for each $x$; based on the fitted machine learning model. This sequence is nested, in the sense that $\hat{L}_{n,\alpha}(x;t_2) \leq \hat{L}_{n,\alpha}(x;t_1)$ and $\hat{U}_{n,\alpha}(x;t_2) \geq \hat{U}_{n,\alpha}(x;t_1)$ for all $t_2 \geq t_1$. Further, assume there exists $t_\infty \in \mathcal{T}$ such that $\hat{L}_{n,\alpha}(X;t_\infty) \leq Y \leq \hat{U}_{n,\alpha}(X;t_\infty)$ almost surely. For example, one may consider the sequences $\hat{\psi}_n(x) \pm t$, $t \geq 0$, where $\hat{\psi}_n$ is a regression function for a bounded label $Y$ given $X$ output by machine learning model and $t$ plays the role of a "predictive standard error". For one-sided (lower) confidence intervals $[\hat{L}_{n,\alpha}(x;t), \infty)$, we may set $\hat{U}_{n,\alpha}(x;t) = \infty$.

Then, the conformity score for a point with $X = x$ and $Y = y$ is defined as the smallest—infimum—index $t$ such that $y$ is contained in the prediction interval $[\hat{L}_{n,\alpha}(x;t), \hat{U}_{n,\alpha}(x;t)]$:

$$E(x,y) := \inf \left\{ t \in \mathcal{T} : y \in [\hat{L}_{n,\alpha}(x;t), \hat{U}_{n,\alpha}(x;t)] \right\}. \tag{3}$$

Let $\mathcal{I}^{\text{calib}} \subset \{1, \dots, n\}$ be the subset of held-out data points, with cardinality $|\mathcal{I}^{\text{calib}}|$. Let $\hat{Q}_{n,1-\alpha}$ be the $\lceil (1-\alpha)(|\mathcal{I}^{\text{calib}}| + 1) \rceil$-th smallest conformity score $E(X_i, Y_i)$ among all $i \in \mathcal{I}^{\text{calib}}$. The conformal prediction interval for a new data point with features $X_{n+1}$ is:

$$\left[ \hat{L}_{n,\alpha}(X_{n+1}; \hat{Q}_{n,1-\alpha}), \hat{U}_{n,\alpha}(X_{n+1}; \hat{Q}_{n,1-\alpha}) \right]. \tag{4}$$

Intuitively, this satisfies marginal coverage because $Y_{n+1}$ falls outside (4) if and only if $E(X_{n+1}, Y_{n+1}) > \hat{Q}_{n,1-\alpha}$. The rest of the proof is a simple exchangeability argument; see Vovk et al. (2005), Romano et al. (2019), or the proof of Theorem 1 in Appendix D.

## 3. Confidence Intervals with Marginal Coverage

### 3.1 Data Exchangeability and Conformal Confidence Intervals

As anticipated in Section 1.2, we study a sketching problem in which the query $Z_{m+1}$ and $m$ data points, $Z_1, \dots, Z_m$, are an exchangeable random sample from some distribution $P_Z$ on $\mathscr{Z}$. We assume that the full data set is too large to process directly. Recall that our goal is to construct a confidence interval with guaranteed marginal coverage (2) for the number of occurrences $f_m(Z_{m+1})$—defined in (1)—of the query $Z_{m+1}$ in the data set. Since $Z_1, \ldots, Z_m$ cannot be observed, we rely on the information contained in the sketch $\phi(Z_1, \ldots, Z_m)$. Importantly, we would like to retain as much flexibility as possible with regard to the sketching function $\phi$.

To connect this problem with the conformal inference framework reviewed in Section 2, we need to define the appropriate features and outcomes. Our approach is to store the true frequencies for all objects in the first $n$ observations in a *warm-up* stage, for some fixed $n \ll m$ that is sufficiently large subject to memory constraints[2]. An extension of this method allowing $n$ to be data-dependent will be discussed later in Section 3.4. Let $n_0 \leq n$ indicate the number of distinct objects among the first $n$ observations. The memory required to store these frequencies is $O(n_0)$, which is typically negligible if $n$ is small compared to the size of the sketch. We use these stored frequencies to define features and outcomes, transforming our task into supervised prediction, as detailed below.

During the warm-up phase, we store the frequencies of the distinct objects among the first $n$ observations $Z_1, \ldots, Z_n$ from the data stream. We denote these counts as $f_n^{\text{wu}}(z)$, defined for all $z \in \mathcal{Z}$ as

$$f_n^{\text{wu}}(z) := \sum_{i=0}^{n} \mathbb{1}\left[Z_i = z\right]. \tag{5}$$

Next, the remaining $m - n$ data points are streamed and compressed using any black-box sketching function $\phi$ of choice. At the same time, however, we also keep track of the true frequencies for all instances of objects already seen during the warm-up phase. In other words, the following counters are computed and stored along with the sketch[3] $\phi(Z_{n+1}, \ldots, Z_m)$:

$$f_{m-n}^{\text{sv}}(z) := \begin{cases} \sum_{i=n+1}^{m} \mathbb{1}\left[Z_i = z\right], & \text{if } f_n^{\text{wu}}(z) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

Again, this requires only $O(n_0)$ memory. Next, we define the variables $Y_i$ for all $i \in \{1, \ldots, n\} \cup \{m+1\}$ as the true frequencies of $Z_i$ among $Z_{n+1}, \ldots, Z_m$:

$$Y_i := \sum_{i'=n+1}^{m} \mathbb{1}\left[Z_{i'} = Z_i\right]. \tag{7}$$

For all $i \in \{1, \ldots, n\} \cup \{m+1\}$, the frequencies of $Z_i$ can be written as $f_m(Z_i) = Y_i + f_n^{\text{wu}}(Z_i)$. Thus, $f_n^{\text{wu}}(Z_i)$ and $Y_i$ together determine the outcome $f_m(Z_i)$ of interest. For $i \in \{1, \ldots, n\}$, $Y_i$ are observed and equal $Y_i = f_{m-n}^{\text{sv}}(Z_i)$. In contrast, for the query $Z_{m+1}$, we have $Y_{m+1} = f_{m-n}^{\text{sv}}(Z_{m+1})$ only if the value of $Z_{m+1}$ has occurred among $Z_1, \ldots, Z_n$ and thus the frequency of $Z_{m+1}$ has been stored. Otherwise, the value of $Y_{m+1}$ is not known. Since $f_n^{\text{wu}}(Z_{m+1})$ and $Y_{m+1}$ determine $f_m(Z_{m+1})$, it is reasonable to aim to build a predictive model or conformal interval for $Y_{m+1}$ based on the observed data $Z_{m+1}$ and the sketch.

[2] Note that the index $n + 1$ of the test point from Section 2 is now replaced by $m + 1$.

[3] Compared to the setup from Section 1.2, here the sketch is only applied to the observations $Z_{n+1}, \ldots, Z_m$ instead of $Z_1, \ldots, Z_m$, because the frequencies of the first $n$ observations are already stored exactly.

To formalize this, for each $i \in \{1, \dots, n\} \cup \{m+1\}$, define the features $X_i$ as the vectors containing the data point $Z_i$ and the information in the sketch:

$$X_i := (Z_i, \phi(Z_{n+1}, \dots, Z_m)) . \tag{8}$$

To obtain a conformal guarantee, we will rely on result that the pairs $(X_1, Y_1), \dots,$ $(X_n, Y_n)$, $(X_{m+1}, Y_{m+1})$ are exchangeable with one another—where, as discussed, $Y_{m+1}$ is possibly unobserved. All mathematical proofs are in Appendix D.

**Proposition 1** *If the unsketched data points $Z_1, \dots, Z_{m+1}$ are exchangeable, then the pairs of random variables $(X_1, Y_1), \dots, (X_n, Y_n), (X_{m+1}, Y_{m+1})$ in* (7)–(8) *are also exchangeable with one another.*

Proposition 1 opens the door to applying conformal inference to the supervised observations $(X_1, Y_1), \dots, (X_n, Y_n)$ in order to predict $Y_{m+1}$ given $X_{m+1}$, guaranteeing marginal coverage. In particular, using the inductive/split conformal prediction methodology reviewed in Section 2, one can randomly split the observations indexed by $\{1, \dots, n\}$ into a training subset indexed by $\{1, \dots, n^{\text{train}}\}$ for some fixed $n^{\text{train}} < n$, and a disjoint calibration subset indexed by $\{n^{\text{train}} + 1, \dots, n\}$. The training set is used for fitting a predictor for computing nested confidence intervals, $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$, $t \in \mathcal{T}$; see the next sections for further implementation details. The aim is that $Y \in [\hat{L}_{m,\alpha}(X; t), \hat{U}_{m,\alpha}(X; t)]$ holds with large probability; and thus this interval can be used to predict $Y$ and hence also $f_m(Z) = Y + f_n^{\text{wu}}(Z)$. In certain cases, this predictor will leverage a classical deterministic sketching method, making the training step unnecessary.

To choose a suitable value for the parameter $t$, following the general approach reviewed in Section 2, the calibration set of observations indexed by $\{n^{\text{train}}+1, \dots, n\}$ is used to compute conformity scores $E(X_i, Y_i)$ for $i \in \{n^{\text{train}} + 1, \dots, n\}$. Then, with $n^{\text{cal}} = n - n^{\text{train}}$, the conformal interval is constructed as in (4), by setting $t$ as the $\lceil (1-\alpha)(n^{\text{cal}}+1) \rceil$-th smallest score of $E(X_i, Y_i)$ for $i \in \{n^{\text{train}} + 1, \dots, n\}$. The resulting interval from (4) guarantees valid marginal coverage for a new test query in finite samples.

This solution is outlined by Algorithms 1–2 and visualized schematically in Figure 2. Algorithm 2 outputs the final confidence interval after Algorithm 1 sketches and pre-processes the data. This modular organization will prove useful in the following sections to simplify the exposition of extensions of our methodology. The following result states that the confidence interval output by Algorithm 2 has the desired marginal coverage.

**Theorem 1** *If the data $Z_1, \dots, Z_{m+1}$ are exchangeable, the confidence interval output by Algorithm 2 satisfies the marginal coverage property defined in* (2)*.*

**Remark.** Algorithm 2 could be trivially modified to output perfect "singleton" confidence intervals for any new queries that happen to be identical to an object previously observed during the warm-up phase. We will not take advantage of this option in the experiments presented in this paper in order to provide a fairer comparison with alternative methods which do not involve a similar warm-up phase.

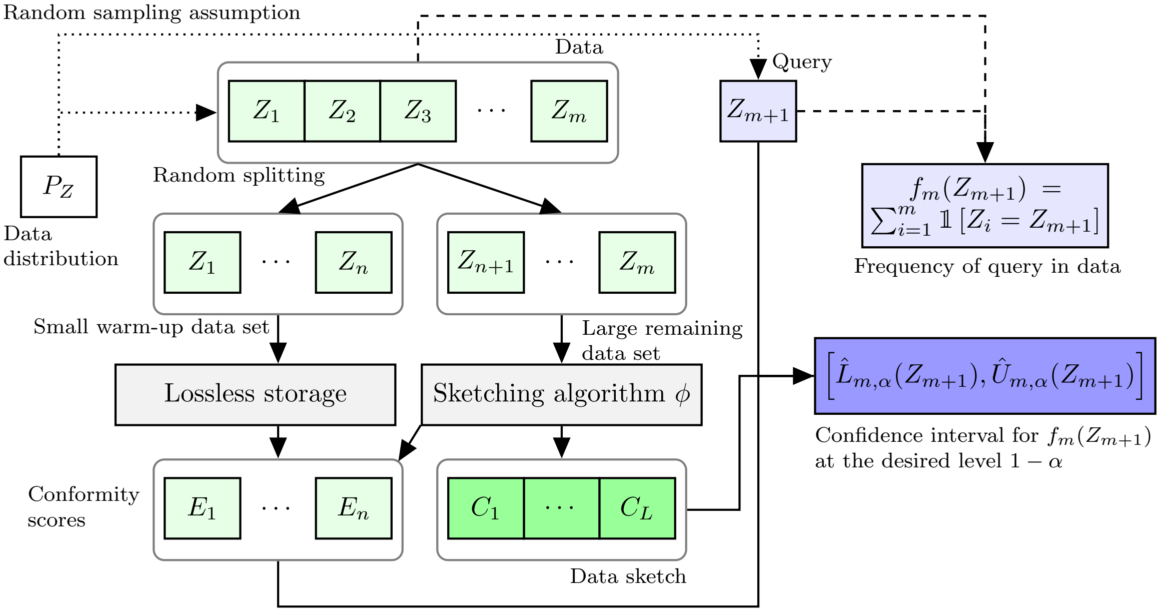
Random sampling assumption
Data
Query
$Z_1$
$Z_2$
$Z_3$
$\cdots$
$Z_m$
$Z_{m+1}$
$P_Z$
Data distribution
Random splitting
$f_m(Z_{m+1}) = \sum_{i=1}^{m} \mathbb{1}[Z_i = Z_{m+1}]$
Frequency of query in data
$Z_1$
$\cdots$
$Z_n$
$Z_{n+1}$
$\cdots$
$Z_m$
Small warm-up data set
Large remaining data set
Lossless storage
Sketching algorithm $\phi$
$\left[\hat{L}_{m,\alpha}(Z_{m+1}), \hat{U}_{m,\alpha}(Z_{m+1})\right]$
Confidence interval for $f_m(Z_{m+1})$ at the desired level $1-\alpha$
Conformity scores
$E_1$
$\cdots$
$E_n$
$C_1$
$\cdots$
$C_L$
Data sketch

Figure 2: A diagram of the conformalized sketching method in Algorithms 1–2.

### 3.2 Conformity Scores for Confidence Intervals with Fixed Width

The method described by Algorithms 1–2 can accommodate any data-adaptive intervals $[\hat{L}_{m,\alpha}(x;t), \hat{U}_{m,\alpha}(x;t)]$—for computing nested confidence intervals, which may depend on the sketch $\phi := \phi(Z_{n+1}, \ldots, Z_m)$. A simple one-sided construction of these confidence intervals is possible if the sketching algorithm provides us with a non-trivial deterministic upper bound $\hat{f}_{\text{up}}(X_{m+1}) = \hat{f}_{\text{up}}(Z_{m+1}, \phi)$ for the query frequency—such that $f(X_{m+1}) \leq \hat{f}_{\text{up}}(Z_{m+1}, \phi)$ for all $Z_{m+1}$—as it is the case with the CMS (Cormode and Muthukrishnan, 2005). In those cases, we suggest to calibrate the parameter $t$ of the following sequence of potential lower bounds on the query frequency:

$$\hat{L}^{\text{fixed}}_{m,\alpha}((Z_{m+1}, \phi); t) := \max\{0, \hat{f}_{\text{up}}(Z_{m+1}, \phi) - t\}, \qquad t \in \{0, 1, \ldots, m\}. \tag{9}$$

In words, a potential lower bound for $f_m(Z_{m+1})$ in (9) is defined by shifting the deterministic upper bound down by a constant $t$. The appropriate value of $t$ guaranteeing marginal coverage is chosen by applying Algorithm 2 at the nominal level $\alpha$. If $Y_i \leq \hat{f}_{\text{up}}(X_i)$ for all $i \in \{n^{\text{train}} + 1, \ldots, n\}$, then the chosen $t$ can also be written as the $\lceil (1-\alpha)(n^{\text{cal}} + 1) \rceil$-th smallest value of $\hat{f}_{\text{up}}(X_i) - Y_i$ among $i \in \{n^{\text{train}} + 1, \ldots, n\}$.

This approach does not require training data, in the sense that it allows one to use $n^{\text{train}} = 0$ and use all $n$ observations with tracked frequencies for calibration. Further, two-sided conformal confidence intervals can be constructed as explained in Appendix B.1.

**Algorithm 1** Conformalized sketching (data sketching, training, and calibration)

```
Input: Data set $Z_1, \dots, Z_m$. Sketching function $\phi$. Warm-up period $n \ll m$.
       A trainable predictor to compute nested intervals $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]_{t \in \mathcal{T}}$.
       Number of data points $n^{\text{train}} < n$ used for training $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$.
Initialize a sparse counter $f_n^{\text{wu}}(z) = 0, \forall z \in \mathcal{Z}$.
for $i = 1, \dots, n$ do
  Increment $f_n^{\text{wu}}(Z_i) \leftarrow f_n^{\text{wu}}(Z_i) + 1$.
Initialize a sparse counter $f_{m-n}^{\text{sv}}(z) = 0, \forall z \in \mathcal{Z}$.
Initialize an empty sketch $\phi(\emptyset)$.
for $i = n+1, \dots, m$ do
  Update the sketch $\phi$ with the new observation $Z_i$.
  if $f_n^{\text{wu}}(Z_i) > 0$ then
    Increment $f_{m-n}^{\text{sv}}(Z_i) \leftarrow f_{m-n}^{\text{sv}}(Z_i) + 1$.
for $i = 1, \dots, n$ do
  Set $X_i = (Z_i, \phi(Z_{n+1}, \dots, Z_m))$ as in (8).
  Set $Y_i = f_{m-n}^{\text{sv}}(Z_i)$.
Train $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$, $t \in \mathcal{T}$, using the data in $\{(X_i, Y_i)\}_{i=1}^{n^{\text{train}}}$.
for $i = n^{\text{train}} + 1, \dots, n$ do
  Compute the conformity score $E(X_i, Y_i)$ with (3), using $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$.
Output: Data sketch $\phi$;
        Sparse counter $f_n^{\text{wu}}(z), \forall z \in \mathcal{Z}$;
        Trained predictor $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$;
        Conformity scores $E(X_i, Y_i)$ for all $i \in \{n^{\text{train}} + 1, \dots, n\}$.
```

**Algorithm 2** Conformalized sketching with marginal coverage

```
Input: Same as for Algorithm 1.
       Random query $Z_{m+1}$. Desired coverage level $1 - \alpha \in (0, 1)$.
Compute using Algorithm 1:
       Data sketch $\phi$; a sparse counter $f_n^{\text{wu}}(z), \forall z \in \mathcal{Z}$;
       Variables $X_i = (Z_i, \phi(Z_{n+1}, \dots, Z_m))$ and $Y_i = f_{m-n}^{\text{sv}}(Z_i)$ for $i \in \{1, \dots, n\}$.
       Trained predictor for computing nested intervals $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]_{t \in \mathcal{T}}$;
       Conformity scores $E(X_i, Y_i)$ for all $i \in \{n^{\text{train}} + 1, \dots, n\}$.
Compute $\hat{Q}_{n^{\text{cal}}, 1-\alpha}$ as the $\lceil (1-\alpha)(n^{\text{cal}} + 1) \rceil$-th smallest score, with $n^{\text{cal}} = n - n^{\text{train}}$.
Set $X_{m+1} = (Z_{m+1}, \phi(Z_{n+1}, \dots, Z_m))$ as in (8).
Output: a $(1 - \alpha)$-level confidence interval

    $\left[ f_n^{\text{wu}}(Z_{m+1}) + \hat{L}_{m,\alpha}(X_{m+1}; \hat{Q}_{n^{\text{cal}},1-\alpha}), f_n^{\text{wu}}(Z_{m+1}) + \hat{U}_{m,\alpha}(X_{m+1}; \hat{Q}_{n^{\text{cal}},1-\alpha}) \right]$

  for the unobserved frequency $f_m(Z_{m+1})$ of $Z_{m+1}$ defined in (1).
```

### 3.3 Conformity Scores for Confidence Intervals with Adaptive Width

A more flexible confidence interval construction with query-dependent width can sometimes lead to more informative predictions compared to the simpler method described in

Section 3.2. This approach, which we call "adaptive", involves training a machine learning model to approximate the optimal width of the confidence intervals, and it is inspired by the methods of Chernozhukov et al. (2021) and Sesia and Romano (2021). For simplicity, we focus here on the construction of one-sided intervals, assuming that a deterministic upper bound $\hat{f}_{\text{up}}$ for the desired query frequency is already available (e.g., as in the case of the CMS). However, the same idea can be generalized easily to construct instead two-sided confidence intervals; see Appendix B.1.

Consider a machine learning model taking as input the deterministic upper bound $\hat{f}_{\text{up}}(Z_i)$ and estimating the conditional distribution of $\hat{f}_{\text{up}}(Z_i) - f_m(Z_i)$ given $\hat{f}_{\text{up}}(Z_i)$. For example, think of a quantile neural network (Taylor, 2000) or a quantile random forest (Meinshausen, 2006). After fitting this model on the training data set of size $n^{\text{train}}$, let $\hat{q}_t$ be the estimated $\alpha_t$-th lower quantile of $\hat{f}_{\text{up}}(Z_i) - f_m(Z_i) \mid \hat{f}_{\text{up}}(Z_i)$, for all $t \in \{1, \dots, T\}$ and some fixed sequence $0 = \alpha_1 < \dots < \alpha_T = 1$. Without loss of generality, assume that quantile crossings do not occur (He, 1997) and let $\hat{q}_0 = 0$, $\hat{q}_T = m - n$. Then, define the following monotone sequence of conformal lower bounds, recalling that $X_{m+1} = (Z_{m+1}, \phi)$:

$$\hat{L}^{\text{adaptive}}_{m,\alpha}((Z_{m+1}, \phi); t) := \max\left\{0, \hat{f}_{\text{up}}(X_{m+1}) - \hat{q}_t\left(\hat{f}_{\text{up}}(X_{m+1})\right)\right\}, \qquad t \in \{0, 1, \dots, m\}.$$

Finally, the calibrated value of $t$ guaranteeing marginal coverage is obtained by applying Algorithm 2 at the nominal level $\alpha$. This approach can lead to a lower confidence bound whose distance from the upper bound is adaptive to the test instance $X_{m+1}$. This can be an advantage because the sketching algorithms may introduce higher uncertainty about common queries compared to rarer ones, or vice versa, depending on the data distribution, and such patterns may be learnt given a sufficient number of observations.

### 3.4 Data-Adaptive Warm-up

Algorithm 1 requires pre-specifying the total number of data points $n$ processed during the warm-up phase. A possible limitation of this approach is that the number of distinct objects $n_0 \leq n$ among the first $n$ observations depends on the data distribution $P_Z$ and cannot be known in advance. In particular, if the data follow a distribution with a power-law tail behaviour, as it is often the case in many practical applications (Clauset et al., 2009), some types of objects may be much more likely than others to be observed, resulting in $n_0 \ll n$. Given that the memory cost of Algorithm 1 depends on the number of distinct objects observed during the warm-up phase, it would be more intuitive to allow the user to control the duration of the warm-up phase by directly specifying the desired value of $n_0$ instead of $n$. In other words, one may want to run the warm-up phase of Algorithm 1 for a flexible number of steps $n$, until exactly $n_0$ distinct objects are observed. Unfortunately, a straightforward implementation of this alternative strategy, which is outlined by Algorithm A7 in Appendix B.2, does not lead to theoretically valid conformal inferences because the randomness in $n$ breaks the desired exchangeability of the calibration data with the test query $Z_{m+1}$. Nonetheless, Algorithm A7 does provide a reasonable heuristic that often works well in practice, as we shall see empirically in Section 6.

Alternatively, a rigorous solution can be obtained by modifying Algorithm A7 so that the conformal inferences are calibrated using only the observations collected during a second distinct warm-up phase, whose duration is fixed conditional on the first warm-up phase. In particular, the duration of the second warm-up phase is set equal to $n$, namely the (random) number of data points collected until $n_0$ distinct objects are observed during the first warm-up phase. By the exchangeability of the data, one thus expects to observe approximately $n_0$ distinct objects in the second warm-up phase. Further, this two-step preserves the exchangeability of the calibration data with the test query $Z_{m+1}$ conditional on the value of $n$ and on all the data observed during the first warm-up phase, thus enabling theoretically valid conformal inferences. See Algorithm A8 for an outline of this procedure.

## 4. Confidence Intervals with Frequency-Conditional Coverage

As explained in Section 1.2, marginal coverage is not fully satisfactory because our data are discrete and more common queries should not be over-counted. Therefore, we begin to address the limitations of marginal coverage by extending the method presented in Section 3 to obtain confidence intervals valid simultaneously for both rarer and more common queries. Our approach is inspired by Mondrian conformal inference (Vovk et al., 2003), which has been previously used—for instance—to construct prediction sets with label-conditional coverage for classification problems (Vovk et al., 2005; Sadinle et al., 2019; Romano et al., 2020b). However, departing from multi-class classification, we will not seek perfect coverage conditional on the exact frequency of the queried object. In fact, that problem is likely to be impossible to solve without stronger assumptions (Barber et al., 2021), as $f_m(Z_{m+1})$ can take a very large number of values when the sketched data set is big. Instead, we will focus on achieving a relaxed version of frequency-conditional coverage which groups together queries with similar frequencies.

Fix any partition $\mathcal{B} = (B_1, \ldots, B_L)$ of $\{1, \ldots, m\}$ into $L$ sub-intervals, for some relatively small integer $L$. The choice of $\mathcal{B}$ and $L$ will be discussed below. For the time being, it suffices to emphasize that this partition may be arbitrary, as long it is fixed prior to seeing the data $Z_1, \ldots, Z_{m+1}$. Our goal is to construct a confidence interval $[\hat{L}_{m,\alpha}(Z_{m+1}), \hat{U}_{m,\alpha}(Z_{m+1})]$ depending on $\phi(Z_1, \ldots, Z_m)$ and $\mathcal{B}$ that is reasonably short in practice and guarantees *frequency-range conditional coverage*:

$$\mathbb{P}\left[\hat{L}_{m,\alpha}(Z_{m+1}) \leq f_m(Z_{m+1}) \leq \hat{U}_{m,\alpha}(Z_{m+1}) \mid f_m(Z_{m+1}) \in B\right] \geq 1 - \alpha, \qquad \forall B \in \mathcal{B}. \quad (10)$$

Thus, coverage is guaranteed for observations $Z_{m+1}$ with $f_m(Z_{m+1}) \in B$, for each $B$. Confidence intervals satisfying (10) can be obtained by modifying Algorithm 2 as outlined in Algorithm 3; by computing empirical quantiles for the conformity scores corresponding to the calibration data points in each frequency bin separately. Then, the final confidence interval for the random query is computed based on the largest quantile across all bins. The theoretical validity of this solution is established below in Theorem 2.

**Theorem 2** *If the data $Z_1, \ldots, Z_{m+1}$ are exchangeable, the confidence interval output by Algorithm 3 satisfies the frequency-conditional property defined in* (10)*.*

**Remark.** The choice of the partition $\mathcal{B}$ involves an important trade-off. On the one side, frequency-conditional coverage (10) becomes stronger with finer partitions; a larger value of $|\mathcal{B}|$ tends to yield more reliable intervals. On the other side, coarser partitions (smaller $|\mathcal{B}|$) enable a larger calibration sample within each bin, leading to tighter and more

**Algorithm 3** Conformalized sketching with frequency-conditional coverage

```
Input: Data set $Z_1, \dots, Z_m$. Sketching function $\phi$. Warm-up period $n \ll m$.
       A (trainable) predictor to compute nested intervals $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]_{t \in \mathcal{T}}$.
       Number of data points $n^{\text{train}} < n$ used for training $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$.
       A partition $\mathcal{B} = (B_1, \dots, B_L)$ of $\{0, \dots, m\}$ into $L$ intervals.
       Random query $Z_{m+1}$. Desired coverage level $1 - \alpha \in (0,1)$.
Compute using Algorithm 1:
       Data sketch $\phi$; a sparse counter $f_n^{\text{wu}}(z), \forall z \in \mathcal{Z}$;
       Variables $X_i = (Z_i, \phi(Z_{n+1}, \dots, Z_m))$ and $Y_i = f_{m-n}^{\text{sv}}(Z_i)$ for $i \in \{1, \dots, n\}$.
       Trained predictor $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$;
       Conformity scores $E(X_i, Y_i)$ for all $i \in \{n^{\text{train}} + 1, \dots, n\}$.
for $i = n^{\text{train}} + 1, \dots, n$ do
    Assign each score $E(X_i, Y_i)$ to an appropriate frequency bin $B \in \mathcal{B}$ based on $Y_i$.
for $l = 1, \dots, L$ do
    Compute the number $n_l$ of scores assigned to bin $B_l$.
    Compute $\hat{Q}_{n_l,1-\alpha}(B_l)$ as the $\lceil (1-\alpha)(n_l + 1) \rceil$-th smallest score in bin $B_l$.
Set $\hat{Q}^*_{n,1-\alpha} = \max_l \hat{Q}_{n_l,1-\alpha}(B_l)$.
Set $X_{m+1} = (Z_{m+1}, \phi(Z_{n+1}, \dots, Z_m))$ as in (8).
Output: a $(1-\alpha)$-level confidence interval

    $\left[ f_n^{\text{wu}}(Z_{m+1}) + \hat{L}_{m,\alpha}(X_{m+1}; \hat{Q}^*_{n,1-\alpha}), f_n^{\text{wu}}(Z_{m+1}) + \hat{U}_{m,\alpha}(X_{m+1}; \hat{Q}^*_{n,1-\alpha}) \right]$

for the unobserved frequency $f_m(Z_{m+1})$ of $Z_{m+1}$ defined in (1).
```

stable intervals. Concretely, the illustrations described in this paper will adopt $|\mathcal{B}| = 5$, although finer partitions may be used when working with very large data sets.

As $|\mathcal{B}|$ should be small relative to the number $n$ of calibration data points to have short intervals, frequency-conditional coverage can only be guaranteed conditional on a relatively rough approximation of the true empirical frequency of a new query. Therefore, rarer queries may still suffer from lower coverage compared to more common queries within the same frequency bin, as we shall see empirically in Section 6. This remaining limitation motivates the more sophisticated approach presented below, which is designed to guarantee valid coverage for a sufficiently large fraction of all distinct queries occurring in random test set with repetitions, regardless of their relative frequencies.

## 5. Confidence Intervals with Valid Coverage for Distinct Queries

Section 5.1 describes our methodology for constructing confidence intervals with valid coverage for distinct queries. Then, Sections 5.2 and 5.3 study some of its robustness properties.

### 5.1 Construction of Confidence Intervals with Coverage for Distinct Queries

First, we introduce some notations. Recall that a *multiset* $V$ of objects $\{v_1, \dots, v_m\}$ is simply the set of $v_1, \dots, v_m$ with repetitions. Since we are dealing with settings where there are potentially a lot of repeated values, it is helpful to refer to the multiset $\mathcal{Z}^{\text{cal}}$ of calibration data points $Z_i$ for all $i \in \mathcal{I}^{\text{cal}} = \{n^{\text{train}} + 1, \dots, n\}$, for an appropriate $n < m$. As above, we define $n^{\text{cal}}$ as the cardinality of $\mathcal{I}^{\text{cal}}$.

Next, for some $M > 0$, we aim for coverage for distinct queries among $M$ new queries. Therefore, we consider a multiset $\mathcal{Z}^{\text{test}}$ of $M$ queries, indexed by $\mathcal{I}^{\text{test}} = \{m+1, \dots, m+M\}$, which we assume to be sampled from $P_Z$ exchangeably with one another as well as with the $m$ sketched data points. This generalizes the setting considered so far, where we had considered $M = 1$.

Define also $\textsc{Unique}(\mathcal{Z}^{\text{test}}) \subseteq \mathcal{Z}^{\text{test}}$ as the subset of unique objects in $\mathcal{Z}^{\text{test}}$. Then, we formalize "coverage over uniques" by first sampling $Z^*$ from the uniform distribution over $\textsc{Unique}(\mathcal{Z}^{\text{test}})$:

$$\begin{aligned} Z_1, \dots, Z_m, Z_{m+1}, \dots, Z_{m+M} &\overset{\text{exch.}}{\sim} P_Z, \\ Z^* &\sim \text{Uniform}\left[\textsc{Unique}(Z_{m+1}, \dots, Z_{m+M})\right]. \end{aligned} \tag{11}$$

Then, the goal is to construct a confidence interval $[\hat{L}_{m,\alpha}(Z^*), \hat{U}_{m,\alpha}(Z^*)]$ achieving coverage of $f_m(Z^*)$ over the random draw of $Z^*$, i.e., on average over the uniques in the test set:

$$\mathbb{P}^* \left[ \hat{L}_{m,\alpha}(Z^*) \leq f_m(Z^*) \leq \hat{U}_{m,\alpha}(Z^*) \right] \geq 1 - \alpha, \tag{12}$$

for any desired $\alpha \in (0, 1)$. Above, the probability $\mathbb{P}^*$ is taken with respect to $Z_1, \dots, Z_{m+M}$ as well as to the randomness in $Z^*$, according to the model defined in (11). Equations (11)–(12) say that our goal is to cover at least a fraction $1 - \alpha$ of the distinct queries in the test set; on average over the distribution of the test and calibration data. In the special case of a test set with cardinality $M = 1$, the property in (12) reduces to marginal coverage.

To achieve (12) with any value of $M$, we follow an approach inspired by Dunn et al. (2022); Park et al. (2022). We randomly partition the calibration data into $G = \lfloor n^{\text{cal}}/M \rfloor$ multisets $\mathcal{Z}_g^{\text{cal}}$, for $g \in [G] := \{1, \dots, G\}$, called *calibration shards*. Without loss of generality, assume the cardinality of each $\mathcal{Z}_g^{\text{cal}}$ is $M$. For our method to be powerful, we will need $n^{\text{cal}} > M$, and ideally we would like $n^{\text{cal}} \gg M$; or equivalently a large $G$.

Following the same notation as above, let $\textsc{Unique}(\mathcal{Z}_g^{\text{cal}}) \subseteq \mathcal{Z}_g^{\text{cal}}$ denote the subset of unique objects in the calibration shard $\mathcal{Z}_g^{\text{cal}}$, for all $g \in [G]$. Then, for each $g \in [G]$, pick an element from each calibration shard $\mathcal{Z}_g^{\text{cal}}$ uniformly at random and call it $\tilde{Z}_g \in \mathcal{Z}$. By construction, the shard-unique element pairs $(\mathcal{Z}_g^{\text{cal}}, \tilde{Z}_g)$, $g \in [G]$, are exchangeable with one another as well as with $(\mathcal{Z}^{\text{test}}, Z^*)$, for all $g \in [G]$. Therefore, a confidence interval $[\hat{L}_{m,\alpha}(Z^*), \hat{U}_{m,\alpha}(Z^*)]$ satisfying (12) can be obtained by applying the method from Section 3 with the calibration set $\mathcal{Z}^{\text{cal}}$ replaced by $(\tilde{Z}_1, \dots, \tilde{Z}_G)$.

This solution is outlined in Algorithm 4 and its theoretical validity is established by Theorem 3. Algorithm 4 is written as to potentially allow the size $M'$ of each of the $G$ calibration shards to be different from the size $M$ of the test set. This generalization of Algorithm 4 with $M' \neq M$ will be studied theoretically in the next section, and it is worth considering because one may sometimes be tempted to apply Algorithm 4 with $M' < M$ in practical applications with limited amounts of data. However, the remainder of this section will continue to focus on the standard choice of $M' = M$.

**Algorithm 4** Conformalized sketching with valid coverage for distinct queries

```
Input: Same as for Algorithm 2, with query $Z^*$.
       Calibration set size $M'$.
Compute using Algorithm 1:
       Data sketch $\phi$; a sparse counter $f_n^{\mathrm{wu}}(z), \forall z \in \mathcal{Z}$;
       Variables $X_i = (Z_i, \phi(Z_{n+1}, \dots, Z_m))$ and $Y_i = f_{m-n}^{\mathrm{sv}}(Z_i)$ for $i \in \{1, \dots, n\}$.
       Trained predictor for computing nested intervals $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$;
       Conformity scores $E(X_i, Y_i)$ for all $i \in \{n^{\mathrm{train}} + 1, \dots, n\}$.
Define $G = \lfloor n^{\mathrm{cal}}/M' \rfloor$, where $n^{\mathrm{cal}} = n - n^{\mathrm{train}}$.
Partition at random $\{n^{\mathrm{train}} + 1, \dots, n\}$ into $G$ subsets $\mathcal{I}_g^{\mathrm{cal}}$.
for $g = 1, \dots, G$ do
  Pick uniformly at random one value $Z_g^*$ from the set $\textsc{Unique}(\{Z_i\}_{i \in \mathcal{I}_g^{\mathrm{cal}}})$.
  Set $X_g^* = \left(Z_g^*, \phi(Z_{n+1}, \dots, Z_m)\right)$ as in (8), and $Y_g^* = f_{m-n}^{\mathrm{sv}}(Z^*)$.
  Set $E_g^* = E(X_g^*, Y_g^*)$.
Compute $\hat{Q}_{G,1-\alpha}$ as the $\lceil (1-\alpha)(G+1) \rceil$-th smallest score in $\{E_g^*\}_{g=1}^G$.
Set $X^* = (Z^*, \phi(Z_{n+1}, \dots, Z_m))$ as in (8).
Output: a $(1-\alpha)$-level confidence interval for the frequency $f_m(Z^*)$ of $Z^*$ defined in (1):
```

$$\left[ f_n^{\mathrm{wu}}(Z^*) + \hat{L}_{m,\alpha}(X^*; \hat{Q}^*_{n^{\mathrm{cal}},1-\alpha}), f_n^{\mathrm{wu}}(Z^*) + \hat{U}_{m,\alpha}(X^*; \hat{Q}^*_{n^{\mathrm{cal}},1-\alpha}) \right].$$

**Theorem 3** *Assume the data $Z_1, \dots, Z_{m+M}$ are exchangeable and the query $Z^*$ is sampled according to* (11). *If Algorithm 4 is applied with parameter $M'$ equal to the test set size $M$, the output confidence interval satisfies the distinct-query coverage property defined in* (12).

**Remark.** The cardinality $M$ of the query set controls the trade-off between the power and reliability of the confidence intervals output by Algorithm 4, assuming the latter is applied with parameter $M' = M$ as prescribed by Theorem 3. On the one hand, smaller values of $M$ lead to tighter and more stable intervals due to a larger number $G$ of data points available for calibration. On the other hand, larger values of $M$ lead to stronger theoretical guarantees, as they reduce the dependence between the expected conditional coverage for a particular query and the population frequency of that query. In general, we recommend that Algorithm 4 should be applied with values of $M$ so large as to result in a number $G$ of final calibration data points in the hundreds. Concretely, the numerical experiments presented in this paper will apply Algorithm 4 with values of $M$ allowing $G \geq 100$.

We conclude this section by emphasizing that Algorithm 4 and Algorithm 3 differ in their formally stated goals (achieving distinct-query coverage and frequency-conditional coverage, respectively), but they are designed to mitigate the same limitation of confidence intervals with marginal coverage. On the one hand, distinct-query coverage is intuitively more appealing and easier to explain compared to frequency-conditional coverage, as anticipated in Section 4. On the other hand, Algorithms 4 and Algorithm 3 require a calibration set that is large relative to the size of the query set. Therefore, the relative advantages of Algorithms 3 and 4 in finite samples may not necessarily be straightforward to see, suggesting the need for a deeper theoretical study of Algorithm 4 (in the remainder of this section) as well as careful simulations (in Section 7).

### 5.2 Robustness to Sample Inflation

To better understand the benefits of Algorithm 4, we study the robustness of its distinct-query coverage guarantee in situations where it is not applied with the default settings, due to a limited sample size. In particular, we are interested in understanding what happens if the size $M'$ of the calibration shards $\mathcal{Z}_g^{\text{cal}}$, for all $g \in [G]$, is smaller than the test sample size $M$. As mentioned, this scenario is motivated when we aim to reach a strong unique-coverage guarantee with a large $M$ despite having only a relatively small calibration sample size.

Let us begin the analysis by recalling the key modelling assumption used throughout this paper: all data points are sampled exchangeably from a discrete distribution $P_Z$ with support on some countable dictionary $\mathcal{Z}$. To facilitate the analysis hereafter, we further assume the data are independent; that is, $Z_i \overset{\text{i.i.d.}}{\sim} P_Z$, for all $i \in [m+M]$. Moreover, we denote $P_Z = \sum_{j\in\mathbb{N}} p_j \delta_{a_j}$, where the $a_j \in \mathcal{Z}$ are the distinct symbols in the dictionary $\mathcal{Z}$, while $p_j \geq 0$ are their respective probabilities for all $j \in \mathbb{N}$, such that $\sum_{j\in\mathbb{N}} p_j = 1$.

Let $V = \textsc{Unique}(\mathcal{Z}^{\text{test}})$ denote the set of unique values in the test set $\mathcal{Z}^{\text{test}}$, which contains all $Z_i$ indexed by the test index set $i \in \mathcal{I}^{\text{test}}$. For any positive integers $k$ and $M$ such that $M \geq k$, let $C_{M,k}$ be the set of *k-compositions* of $M$: these are the sequences $c = (c_1, \dots, c_k)$ of positive integers $c_j \geq 1$ such that $\sum_{j=1}^k c_j = M$. For instance, $(1,1,2)$ is a $k = 3$-composition of $M = 4$. It is known that the number of such sequences is $|C_{M,k}| = \binom{M-1}{k-1}$; see e.g., Riordan (2012). For instance, $(1,1,2)$, $(1,2,1)$, and $(2,1,1)$ are all $k = 3$-compositions of $M = 4$, and their number is $\binom{3}{1} = 3$.

With this notation, we can characterize the probability distribution of the set $V$ of unique values among a random sample from $P_Z$; see Proposition A9 in Appendix C. From there, we obtain the following result characterizing the distribution $U_Z^{[M]}$ of a uniformly sampled element $\zeta$ over the set of uniques $V$, when $V \sim P_Z^{[M]}$. This result will be useful in our analysis of the robustness of Algorithm 4 to situations in which $M' \neq M$. We are not aware of Propositions A9 and 4 being known in the literature; we believe they may be of independent interest and could find uses in future analyses of coverage over unique/distinct elements.

**Proposition 4 (Uniform distribution over unique elements)** *Let $\mathcal{Z}^{test}$ be an i.i.d. sample of size $M$ from a discrete distribution $P_Z = \sum_{i\in\mathbb{N}} p_j \delta_{a_j}$, where $a_j \in \mathcal{Z}$ are distinct, and $p_j \geq 0$ for all $j \in \mathbb{N}$. Let $U_Z^{[M]}$ be the distribution of a uniformly sampled element $\zeta$ of $V = \textsc{Unique}(\mathcal{Z}^{test})$. Then, for all $j_1 \in \mathbb{N}$, the probability mass function of $\zeta$ at $a_{j_1}$ is*

$$U_Z^{[M]}(\zeta = a_{j_1}) = \sum_{k=1}^{M} \frac{1}{k} \sum_{J=\{j_1,\dots,j_k\}\subset\mathbb{N}^k,\,|J|=k} \; \sum_{c\in C_{M,k}} \binom{M}{c_1\; c_2\; \dots\; c_k} p_{j_1}^{c_1} \cdots p_{j_k}^{c_k}. \tag{13}$$

*In particular, $U_Z^{[1]} = U_Z^{[2]} = P_Z$, and for all $j_1 \in \mathbb{N}$,*

$$U_Z^{[3]}(\zeta = a_{j_1}) = \frac{p_{j_1}(2p_{j_1}^2 - 3p_{j_1} + 3)}{2} + \frac{p_{j_1}}{2} \sum_{\{j_2,j_3\}\subset(\mathbb{N}\setminus\{j_1\})^2,\,|J|=2} p_{j_2} p_{j_3}.$$

Proposition 4 suggests that one should generally expect to lose coverage over distinct queries when applying Algorithm 4 with a calibration set size $M'$ that is different from the size $M$ of the test set. Indeed, the $U_Z^{[M]}$-probability of the event $\zeta = a_j$ can either increase or decrease as a function of $M$, depending on the probability $p_j$ of $a_j$ under $P_Z$. To see this, define the function $\tau : [0,1] \to [0,1]$, such that for all $p \in [0,1]$,

$$\tau(p) = \frac{p(2p^2 - 3p + 3)}{2}. \tag{14}$$

A plot of $\tau$ is in Figure A10 (a), Appendix E. Then, for $P_Z$ taking only two possible distinct values $a_1$ and $a_2$, with probabilities $p_1$ and $p_2$, respectively, Proposition 4 implies that for $j = 1, 2$, $U_Z^{[3]}(\zeta = a_j) = \tau(p_j)$. Now, for $p \in [0, 1/2)$, $\tau(p) < p$, while for $p \in (1/2, 1]$, $\tau(p) > p$. Assuming $p_1 < p_2$, we have $U_Z^{[3]}(\zeta = a_1) < U_Z^{[2]}(\zeta = a_1)$, while $U_Z^{[3]}(\zeta = a_2) > U_Z^{[2]}(\zeta = a_2)$. Thus, the probability of $\zeta = a_i$ can either increase or decrease as a function of $M$, depending on $p_i$. Hence, we expect that the probability of the coverage event using calibration data points of size $M'$, which is a union of such elementary events, can also increase or decrease as a function of $M$.

More specifically, let $\mathcal{E} = \{\hat{L}_{m,\alpha}(Z^*) \le f_m(Z^*) \le \hat{U}_{m,\alpha}(Z^*)\}$ be the coverage event from (12), whose probability is lower bounded in Theorem 3. Let the random variables $Z_i$, $i \in \mathcal{I}^{\text{cal}}$ that constitute the calibration set of size $n^{\text{cal}}$ and $i \in \mathcal{I}^{\text{test}}$ that are test set of size $M$ be i.i.d. according to $P_Z$. The probability of coverage can be written in terms of the variables $\tilde{Z}_g$, for $g \in [G]$, chosen from the calibration shards, which are i.i.d. following the distribution $U_Z^{[M']}$—abbreviated as $\tilde{Z}_{1:G} \sim (U_Z^{[M']})^{|G|}$—and an independent random variable $Z^*$ chosen uniformly over the test set, which follows the distribution $U_Z^{[M]}$, as

$$\mathbb{P}_{Z^* \sim U_Z^{[M]},\, \tilde{Z}_{1:G} \sim (U_Z^{[M']})^{|G|}}[\mathcal{E}] = \mathbb{E}_{Z^* \sim U_Z^{[M]}} \mathbb{P}_{\tilde{Z}_{1:G} \sim (U_Z^{[M']})^{|G|}}[\mathcal{E}] = \mathbb{E}_{Z^* \sim U_Z^{[M]}}\, e(Z^*). \tag{15}$$

Above, we defined $e(Z^*) = \mathbb{P}_{\tilde{Z}_{1:G} \sim (U_Z^{[M']})^{|G|}}[\mathcal{E}]$ to be the conditional probability of the coverage event $\mathcal{E}$, given $Z^*$. Theorem 3 says the expectation in (15) is at least $1 - \alpha$ if $M' = M$. However, when $M' \neq M$, we have showed that $U_Z^{[M]}$ can be different from $U_Z^{[M']}$. Thus the above expectation of $e(Z^*)$ may decrease, and the method may lose coverage if $M' \neq M$.

Aiming to understand the extent by which the coverage can be affected, we let $\mathcal{P}_{\mathbb{N}}(\mathcal{Z}; K)$ be the set of discrete probability distributions over $\mathcal{Z}$ supported on at most $K$ distinct values. This is of interest especially because smaller $K$ leads to a more analytically tractable theory, as described below. Then, we introduce the quantity

$$\Delta(M, M'; K) = \sup_{P_Z \in \mathcal{P}_{\mathbb{N}}(\mathcal{Z}; K)} \sup_{j \in \mathbb{N}} \left| U_Z^{[M]}(\{a_j\}) - U_Z^{[M']}(\{a_j\}) \right|,$$

which measures the worst-case difference between the probabilities of observing a value $a_j$ according to the distributions $U_Z^{[M]}$ and $U_Z^{[M']}$. Here, we are thinking of $U_Z^{[M']}$ as the calibration distribution and $U_Z^{[M]}$ as the test distribution. Thus, if our conformal prediction algorithm outputs sets of size at most $s \ge 0$, then the probability of those sets differs by at most $s \cdot \Delta(M, M'; K)$ between the training at test distributions.

Studying $\Delta(M, M'; K)$ seems challenging in general, as it involves maximizing differences of probabilities given in (13). These are nontrivial quantities to deal with, because

(a) large values of $M$ lead to large-degree polynomials in the expressions for the $p_j$s, and (b) large values of $K$ lead to large numbers of degrees of freedom (i.e., many different $p_j$s).

To illustrate some of the difficulties, consider for instance the case $K = 3$. Denoting the three objects in $P_Z$ by $a_1$, $a_2$, $a_3$, one can verify using (25) in Proposition A9 and (13) in Proposition 4 that, for $j = 1, 2, 3$,

$$U_Z^{[M]}(\zeta = a_j) = \frac{1 + p_j^M - (1-p_j)^M + 1/2 \sum_{l \neq j}(p_j + p_l)^M}{3}.$$

Therefore,

$$\begin{aligned}\Delta(M, M'; 3) =& \frac{1}{3} \sup_{p,q,r \in [0,1]: p+q+r=1} \Big| p^M - (1-p)^M + 1/2[(p+q)^M + (p+r)^M] \\ & - \Big[ p^{M'} - (1-p)^{M'} 1/2[(p+q)^{M'} + (p+r)^{M'}] \Big] \Big| .\end{aligned}$$

Denoting $a = p + q$, $b = p + r$, and noting $p = a + b - 1$, with the function

$$\Lambda_M(a, b) = (a + b - 1)^M - (2 - a - b)^M + 1/2(a^M + b^M),$$

we find

$$\Delta(M, M'; 3) = \frac{1}{3} \sup_{a,b \in [0,1]: a+b \geq 1} |\Lambda_M(a, b) - \Lambda_{M'}(a, b)| .$$

This expression does not appear to be straightforward to analyze using standard tools. In particular, setting the gradients of the objective to zero in order to understand the maximizing $a, b$ does not seem to lead to a tractable answers, due to the high order polynomials involved. Moreover the problem seems to get even more complicated for larger $K$, with more complicated polynomials to analyze.

The above results have illustrated some of the theoretical challenges that arise when analyzing $\Delta(M, M'; K)$. Therefore, in order to provide some theoretical results, we focus on the simpler but still non-trivial case of $K = 2$; i.e., we imagine there are only two distinct objects in the population $P_Z$. However in our experiments we will continue to use general $K$, and will see experimental results that broadly agree with the message of the theory.

To do this, we can assume without loss of generality that the size $M'$ of the available calibration shards $\mathcal{Z}_g^{\text{cal}}$, for all $g \in [G]$, is smaller than the test sample size $M$, i.e., $M > M'$, as $\Delta$ is symmetric in $M, M'$. Moreover, we can also assume without loss of generality that $M' \geq 2$, since $U_Z^{[1]} = U_Z^{[2]}$ and thus the cases $M' = 1$ and $M' = 2$ are equivalent. For fixed $M > M' \geq 2$, our theoretical results are presented in terms of the function $h : [0, \infty) \to \mathbb{R}$ defined, for all $\delta \in [0, \infty)$, as

$$h(\delta) = \ln \frac{1 + \delta^{M-1}}{1 + \delta^{M'-1}} - (M - M') \ln(1 + \delta). \tag{16}$$

This function comes up after suitable calculations when maximizing $\Delta$. Our next result characterizes $\Delta(M, M'; 2)$ based on the function $h$. The proof relies on carefully studying the monotonicity properties of $\Delta$ using calculus; see Section D.7.

**Proposition 5 (Characterizing $\Delta(M, M'; 2)$)** *Fix $M > M' \geq 2$ and take the function $h$ as defined in* (16). *There is a unique solution $\delta_* \in [0,1]$ to $h(\delta_*) = \ln(M'/M)$, and*

$$\Delta(M, M'; 2) = \frac{1}{2} \left| \frac{1 - \delta_*^M}{(1+\delta_*)^M} - \frac{1 - \delta_*^{M'}}{(1+\delta_*)^{M'}} \right| .$$

As an illustration, we consider the setting where $M = aM'$, for some $a > 1$. This corresponds to applying Algorithm 4 using calibration shards of size $M'$, with $M'$ being smaller than the target test set size $M$ by a factor $1/a$. Naturally, one would like to know how low the coverage can be in this case compared to the ideal situation in which $M' = M$. Our next result shows that the loss in coverage may remain relatively bounded, as long as $a$ is moderate and $M$ is large. The proof leverages Proposition 5 and relies on a detailed analysis of the polynomial equation satisfied by $\delta_*$; see Section D.8.

**Corollary 6 (Asymptotics of $\Delta(M, M/a; 2)$)** *For $M \geq a \max\{2/(a-1), 2 + \log_2[a/(a-1)]\}$, with $\nu(a) := a^{-\frac{1}{a-1}}(1 - \frac{1}{a})$ and*

$$\beta(M, a) := 2^{3-M/a} a^{-1/(a-1)}/(a-1) + 2[M(1-1/a)]^{-M/a}, \tag{17}$$

*we have $|\Delta(M, M/a; 2) - \nu(a)| \leq \beta(M, a)$.*

The error term $\beta(M, a)$ is exponentially small in $M$ for a fixed $a > 1$, and can be viewed as negligible. Moreover, the main term $\nu(a)$ is also quite small; for instance, if $a = 1.1$, we have $\nu(a) \approx 0.035$. Combined with (15) and Theorem 3, Corollary 6 implies that the coverage over unique values for a test set of size $M$ and calibration sets of size $M' = M/a$ satisfies, for $M$ large enough as specified in Corollary 6,

$$\mathbb{P}_{Z^* \sim U_Z^{[M]}, \tilde{Z}_{1:G} \sim (U_Z^{[Ma]})^{|G|}}[\mathcal{E}] \geq 1 - \alpha - 2 \cdot \nu(a) - \beta(M, a).$$

This immediately gives the following result, which guarantees that the coverage of Algorithm 4 when applied with $M' \neq M$ is correct up to a small error term $2\nu(a)$.

**Theorem 7** *Assume that the data $Z_1, \ldots, Z_{m+M}$ are exchangeable and let Algorithm 4 be applied at the nominal coverage level $\alpha \in (0,1)$ with parameter $M' = M/a$ for some $a > 1$, where $M$ is the size of the test set. Then, the output confidence interval satisfies the distinct-query coverage property defined in* (12) *at level $\alpha + 2 \cdot \nu(a) + \beta(M, a)$, where $\nu(a) = a^{-1/(a-1)}(1 - 1/a)$ and $\beta$ is defined in* (17).

To better understand this result, it helps to look at the plot of the function $\nu$ shown in Figure A10 (b). For instance, if $a = 1.2$, we have $\nu(a) \approx 0.067$; therefore, a 95% nominal coverage level may result empirical coverage over distinct queries that is as low as 80.6% when $M = 100$. If $a = 1.1$, we have, as already mentioned, $\nu(a) \approx 0.035$; therefore, a 95% nominal coverage level may result empirical coverage over distinct queries that is as low as 87.0% when $M = 100$. Of course, Theorem 7 gives a conservative lower bound for the coverage over distinct queries which refers to the worst-case scenario over all data distributions $P_Z$. In practice, Algorithm 4 applied with $M' < M$ may sometimes result in higher coverage than anticipated by Theorem 7, as we will see empirically in Sections 6–7.

### 5.3 Robustness to Distribution Shift

An additional advantage of the distinct-query coverage property defined in (11) is that it tends to be more "robust" to certain types of distribution shift compared to the standard notion of marginal coverage. In other words, if Algorithm 4 is applied in a situation where the queried objects are not sampled from the same distribution as the sketched data, its effective coverage over distinct queries may be lower than the ideal $1-\alpha$ expected under perfect exchangeability, but this loss may not be as large as that of Algorithm 2.

Recall that $U_Z^{[M]}$ is the distribution of unique values in a sample $Z_1, \dots, Z_M$ of size $M$ from $P_Z$; and that the coverage over uniques from (12) refers to a test data point from $U_Z^{[M]}$. The next result establishes that, in the special case of a support of size $K = 2$ studied above, the probabilities shift less in the worst case under the distribution $U_Z^{[M]}$ of unique values than under the original distribution $P_Z$, for a large range of probability values $p_i$ of $P_Z$. Experiments presented in Sections 6–7 show similar results for larger $K$ as well. The proof relies on the mean value theorem and can be found in Section D.10.

**Theorem 8 (Bounding the effect of distribution shift)** *Let $Z$ and $Z'$ take two values with probabilities $p_1, p_2$, and $p'_1, p'_2$, respectively. For $M \geq 3$, let $U_Z^{[M]}$ be the distribution of a uniformly sampled element over* Unique$(V)$*, when $V \sim P_Z^{[M]}$; and define $U_{Z'}^{[M]}$ similarly. Define $c \in (0, 1/2)$ as the unique solution of*

$$c^{M-1} + (1-c)^{M-1} = \frac{2}{M}. \tag{18}$$

*Let*

$$S_c = \{P_Z = (p_1, p_2) : \, p_j \in (c, 1-c), \, j = 1, 2\}.$$

*Then, for all $P_Z, P_{Z'} \in S_c$, with $P_Z \neq P_{Z'}$,*

$$\sup_{\mathcal{E} \subset \{a_1, a_2\}^{|G|+1}} \left| \mathbb{P}_{Z^* \sim U_Z^{[M]}, \, \tilde{Z}_{1:G} \sim \left(U_Z^{[M]}\right)^{|G|}}[\mathcal{E}] - \mathbb{P}_{Z^* \sim U_{Z'}^{[M]}, \, \tilde{Z}_{1:G} \sim \left(U_Z^{[M]}\right)^{|G|}}[\mathcal{E}] \right| <$$
$$\sup_{\mathcal{E} \subset \{a_1, a_2\}^{|G|+1}} \left| \mathbb{P}_{Z^* \sim P_Z, \, \tilde{Z}_{1:G} \sim P_Z^{|G|}}[\mathcal{E}] - \mathbb{P}_{Z^* \sim P_{Z'}, \, \tilde{Z}_{1:G} \sim P_Z^{|G|}}[\mathcal{E}] \right|.$$

In other words, since the coverage event $\mathcal{E} = \{\hat{L}_{m,\alpha}(Z^*) \leq f_m(Z^*) \leq \hat{U}_{m,\alpha}(Z^*)\}$ from (12) is included among the sets where the supremum is evaluated, Theorem 8 tells us that the coverage of the sets output by Algorithm 4 tends to be relatively stable for certain classes of data distributions $P_Z$. Specifically, for a given $P_Z$, the change in coverage when shifting from the distribution of uniques $U_Z^{[M]}$ to the distribution of uniques $U_{Z'}^{[M]}$ is strictly smaller, in the worst case, than the corresponding change in coverage when shifting from $P_Z$ to $P_{Z'}$. This suggests that Algorithm 4 may be relatively robust to distribution shifts in the query set.

Now, we can try to better understand the family of $P_Z$ over which the distribution of unique values is more stable. Since $c < 1/2$, we have $c < 1 - c$; thus, it follows from (18) that $(1-c)^{M-1} \leq 2/M \leq 2(1-c)^{M-1}$, which can be rearranged to obtain:

$$1 - (2/M)^{1/(M-1)} \leq c \leq 1 - (1/M)^{1/(M-1)}.$$

Therefore, $c = O(M^{-1} \ln M)$ for large $M$. This implies that the distribution $U_Z^{[M]}$ of the unique values is less affected by changes in the distribution of probabilities in $P_Z$ than $P_Z$ itself, for a large range of possible values of $p_j$ from $O(M^{-1} \ln M)$ to $1 - O(M^{-1} \ln M)$.

While Theorem 8 focuses on a special case in which the data distribution $P_Z$ has support on only two possible objects in order to simplify the theoretical analysis, the relative robustness of Algorithm 4 to distribution shift in more general settings is supported by empirical experiments, as shown in Sections 6–7.

## 6. Experiments with Synthetic Data

Section 6.1 describes experiments in which we seek marginal or frequency-conditional coverage using the CMS sketch. Section 6.2 presents similar experiments based on the CMS-CU. Section 6.3 focuses on coverage for distinct queries. Section 6.4 studies robustness to distribution shifts. Section 6.5 applies our methods in combination with a learning-based sketch (Bertsimas and Digalakis, 2021) and with the CS sketch (Charikar et al., 2002). Section 6.6 summarizes additional results presented in the appendix.

### 6.1 Marginal and Frequency-Conditional Coverage with the CMS

We apply Algorithm 3 in combination with the CMS (Cormode and Muthukrishnan, 2005) on synthetic data. The CMS is implemented using $d = 3$ random hash functions of width $w = 1000$. As this sketch already gives a deterministic upper bound for any frequency query, the goal of our experiments is to compute corresponding lower bounds for 95% coverage.

The data are generated i.i.d. from a Zipf distribution—a standard option to describe power-law tail behavior (Zipf, 2016). Power-law distributions are observed in many scientific applications, and they are useful to understand many natural and social phenomena (Ferrer i Cancho and Solé, 2001; Adamic and Huberman, 2002; Clauset et al., 2009; Muchnik et al., 2013). To be precise, we sample a random query $Z_{m+1}$ and $m = 100,000$ data points according to the law $\mathbb{P}[Z_i = z] = z^{-a}/\zeta(a)$ for all $z \in \{1, 2 \ldots, \}$, where $\zeta$ is the Riemann Zeta function and $a > 1$ controls the power-law tail behavior.

Prior literature has already studied the problem of uncertainty estimation for frequency queries based on the CMS (Cormode and Yi, 2020; Ting, 2018; Cai et al., 2018; Dolera et al., 2021), which provides us with three informative benchmarks. The first one is the *classical* 95% lower bound (Cormode and Muthukrishnan, 2005) obtained by treating the data as fixed and modeling only the randomness in hash functions, as explained in Appendix A.2. This approach is often too conservative when applied to non-adversarial data (Ting, 2018).

The second benchmark is the *Bayesian* method of Cai et al. (2018), which assumes a non-parametric Dirichlet process prior for the distribution of the data, estimates its scaling parameter by maximizing the marginal likelihood of the observed sketch, and then computes the posterior distribution of the queried frequency. The lower 5% quantile of this posterior distribution is taken as the lower confidence bound for a frequency query. The third benchmark is the bootstrap method of Ting (2018), which is also designed for the CMS and does not extend to other non-linear sketches (which we will study later).

Algorithm 3 is applied using the first $n = 5000$ data points for warm-up and then sketching the remaining $95,000$ data points with the CMS, as explained in Section 3.1.

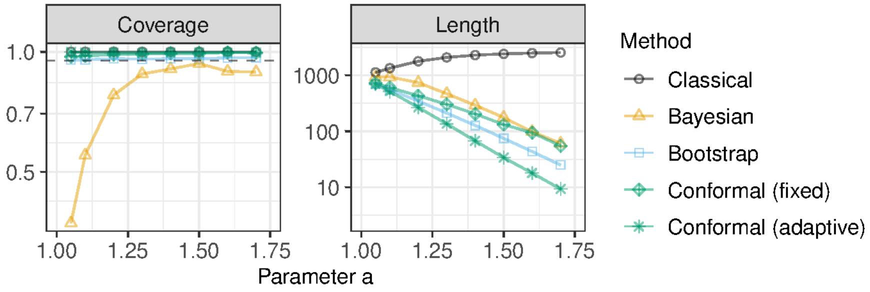

Figure 3: Performance of 95% confidence intervals with simulated Zipf data sketched with the CMS. The results are shown as a function of the Zipf tail parameter $a$.

Two versions of our method are compared: one based on fixed-width conformity scores described in Section 3.2, and one based on the adaptive-width scores from Section 3.3. The latter are implemented using an isotonic distributional regression model (Henzi et al., 2021). In each case, the conformity scores are evaluated separately within $L = 5$ frequency bins, seeking frequency-range conditional coverage (10). The bins are determined so that each contains approximately the same probability mass, by partitioning the range of frequencies for the objects tracked exactly according to the observed empirical quantiles.

Each method is applied to construct one-sided confidence intervals for the frequencies of $10,000$ random queries, sampled i.i.d. from the same distribution as the sketched data. The confidence intervals are evaluated based with two metrics: their average *length* and their *empirical coverage*—the latter is the proportion of queries for which the true frequency is covered. These performance metrics are averaged over 10 independent experiments.

Figure 3 compares our method to three benchmarks on the Zipf data. All methods achieve marginal coverage (2), with the exception of the Bayesian approach whose prior does not match the true data distribution in this case, especially when the tail parameter $a$ is small. As expected, the classical approach turns out to be very conservative, while the bootstrap and conformal methods provide relatively informative confidence intervals, particularly when the tail parameter $a$ is larger and hash collisions become rarer.

The conformal intervals produced by Algorithm 3 are the shortest among all alternatives, especially if they use adaptive conformity scores. The standard errors are omitted because they are relatively small but would clutter the display. Further, Figure 4 stratifies the results of Figure 3 based on the true frequency of each random query. This shows that Algorithm 3 produces valid inferences for both rarer and more common queries, at least within the resolution level considered here. This should not be surprising given that our method controls the frequency-conditional coverage defined in (10). Note that this notion of frequency-conditional coverage would not necessarily be satisfied if the conformal confidence intervals were constructed using Algorithm 2, which guarantees only marginal coverage (2), instead of Algorithm 3; see Figure A11 in Appendix E.

As explained in Sections 1.2 and 4, frequency-range conditional coverage (10) is not always fully satisfactory. In practice, one may ask instead what is the average proportion

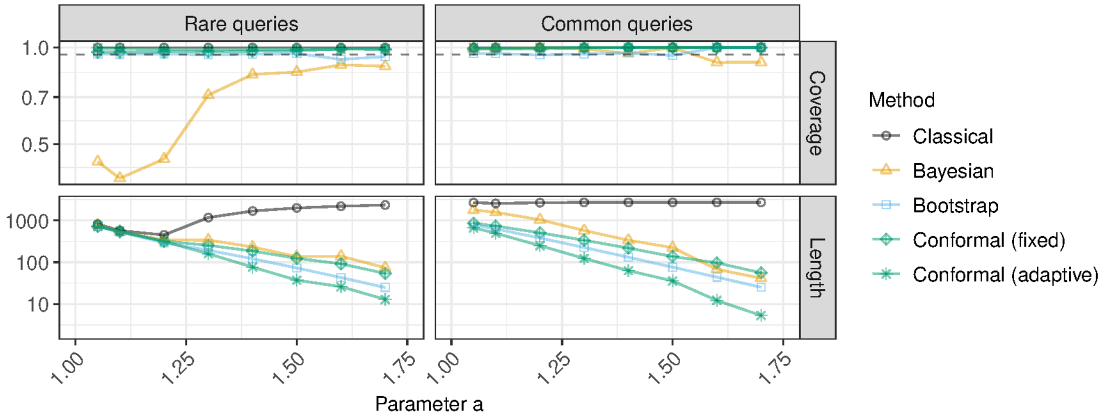

Figure 4: Performance of confidence intervals stratified by the true query frequency. Left: frequency below median; right: above median. Other details are as in Figure 3.

of unique queries in the random test set for which the conformal confidence intervals constructed above indeed provide coverage—this is a more challenging notion of coverage that could be guaranteed rigorously by applying the alternative methods from Section 5. Figure A12 addresses this question by reporting performance metrics for the intervals output by Algorithm 3 that are analogous to those in Figure 3 but evaluated only on distinct queries. The results are encouraging in this case: Algorithm 3 produces intervals that are empirically valid for more than 95% of unique queries across all values of the Zipf tail parameter considered here. Unsurprisingly, though, the same is not true of weaker conformal confidence intervals produced by Algorithm 2, which target the weaker notion of marginal coverage (2); see Figure A13. Further, we shall see in the next section that even Algorithm 3 can practically fail to produce intervals that are valid for a high proportion of unique queries if the data are compressed by a more powerful sketching algorithm; this is what motivates the methods presented in Section 5, which we will apply later in Section 6.3.

### 6.2 Non-Linear Sketching with the CMS-CU

While prior research on uncertainty estimation for frequency queries focused on the CMS, the methods developed in this paper can accommodate any sketching algorithm. Here, we begin to explore this flexibility by conducting experiments similar to those of Section 6.1 but with the CMS replaced by a non-linear variation known as the CMS with *conservative updates* (CMS-CU) (Estan and Varghese, 2002). We refer to Appendix A for a review of this classical sketching algorithm.

Figures A14 and A15 present results analogous to those of Figures 3 and 4, respectively, showing that Algorithms 2 and 3 still lead to shorter confidence intervals with valid coverage. Note that the benchmark approaches are not technically valid here because they were designed for the CMS and not the CMS-CU; nonetheless, their empirical performance remains qualitatively similar to that observed in Section 6.1. Unsurprisingly, our results also confirm that all methods considered here lead to shorter confidence intervals when applied with the CMS-CU instead of the CMS, consistently with the fact that the CMS-CU was designed to improve the compression efficiency by reducing the impact of random hash collisions; see Figure A16 for a direct comparison. Thus, to provide a more practically relevant depiction of each method's performance, the experiments presented in the following sections will adopt the CMS-CU as the baseline sketch instead of the CMS.

We conclude this section by referring to Figures A17 and A18 in the appendix, which investigate the validity of our intervals based on the CMS-CU over distinct queries. These figures report on performance metrics analogous to those shown in Figures A12 and A13, respectively. The results indicate that the intervals targeting marginal (2) or frequency-range conditional (10) coverage at the 95% level tend to be valid for fewer than 95% of all distinct queries, and that such lack of theoretical coverage is more evident now compared to when the data were sketched using the CMS. This observation motivates the experiments described in the next section, in which we apply the stronger methods presented in Section 5.

### 6.3 Coverage for Distinct Queries

This section investigates the performance of Algorithm 4, our proposed method for constructing confidence intervals with guaranteed coverage for distinct queries. These experiments follow the same setup as those in Section 6.2, simulating data from a Zipf distribution with tail parameter $a = 1.5$. The difference is that the coverage and length performance metrics are now averaged only on the distinct queries, $\textsc{Unique}(\mathcal{Z}^{\text{test}})$, from a random test set $\mathcal{Z}^{\text{test}}$ of size $M = 100$. Algorithm 4 is applied at level $1 - \alpha = 95\%$ using the fixed-width one-sided conformity scores described in Section 3.2, and varying $M'$, which controls the size of the calibration shards, as a control parameter between 1 and 100.

Figure 5 confirms that the desired 95% coverage for distinct queries (12) is achieved when Algorithm 4 is applied with $M' \approx M$, as predicted by Theorem 3. By contrast, the coverage for distinct queries is lower when $M'$ is small. This should not be surprising because Algorithm 4 reduces to Algorithm 2 if $M' = 1$, and the latter is designed to provide marginal coverage (2), not coverage for distinct queries (12). In fact, as shown in Figure A19, even Algorithm 3, which targets the relatively stronger notion of frequency-range conditional coverage (10), does not always provide valid inference for distinct queries.

Finally, Figure 5 also highlights that the distinct-query coverage practically achieved by applying on these data Algorithm 4 with smaller values of $M'$ is much higher then the worst-case asymptotic lower bound, $\max(0, 1 - \alpha - 2 \cdot \nu(100/M'))$, given by Theorem 7.

### 6.4 Robustness to Distribution Shifts

This section investigates the robustness of the confidence intervals output by Algorithms 2 and 4 to distribution shifts in the query set. These experiments follow the same setup as those in Section 6.3, simulating data from a Zipf distribution with different values of the tail parameter. The difference is that now the $M = 100$ random test queries are sampled from a mixture distribution with two components. The first component is the same Zipf distribution from which the sketched data are generated, while the second component is an independent continuous uniform distribution on $[0, 1]$.

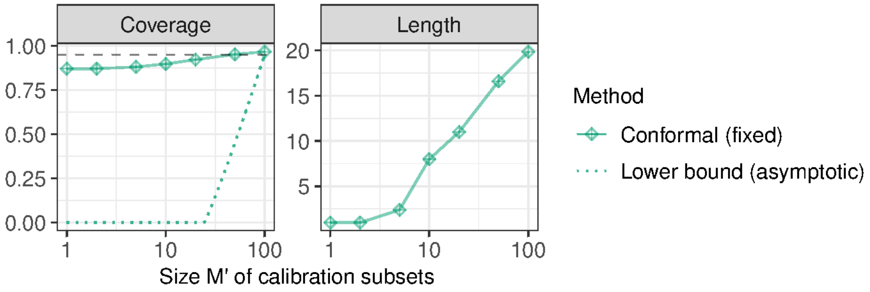

Figure 5: Performance of confidence intervals for distinct queries in a test set of size 100, as a function of the parameter $M'$ of Algorithm 4. The data are simulated from a Zipf distribution with tail parameter $a = 1.5$ and sketched with the CMS-CU. Other details are as in Figure 3.

This setup is designed to study the effects of an extreme form of distributional shift, as objects sampled from the second component of the mixture are almost surely unique (up to rounding errors at machine precision) and are never previously observed in the integer-valued sketched data set. The mixing proportion serves as a control parameter and it is varied from zero (no distribution shift) to one (full shift). Note that this setup is not inconsistent with the original assumption that the data distribution has support on a discrete dictionary $\mathscr{X}$, because even (approximately) uniform random numbers on a computer are in truth discrete.

Figure 6 reports on the results of these experiments. The performance of the conformal confidence intervals output by Algorithm 2, applied with fixed conformity scores, is measured in terms of average coverage and length over all random queries in the test set. By contrast, the performance of the conformal confidence intervals output by Algorithm 4, also applied with fixed conformity scores, is measured in terms of average coverage and length over the distinct queries in the test set. Such choice facilitates the validation of Theorem 8, which suggests Algorithm 2 should be relatively robust to distribution shifts by these metrics.

Indeed, the empirical results confirm the distinct-query coverage guarantee provided by Algorithm 4 is more robust to distribution shifts compared to the marginal coverage property sought by Algorithm 2, although the performances of both methods in this setting also depend on the distribution of the sketched data. It is interesting to note that lower values of the Zipf tail parameter lead to larger numbers of unique objects in the queried data, increasing the robustness of all conformal confidence intervals to distribution shifts corresponding to unusually high proportions of new queries in the test set.

### 6.5 Non-Random Sketching with Data-Driven Hash Functions

To further highlight the flexibility of conformal approach, we apply Algorithms 2 and 3 in combination with an alternative sketching method that departs from the CMS and the CMS-CU in that it is not based on random hash functions. Instead, we follow the approach

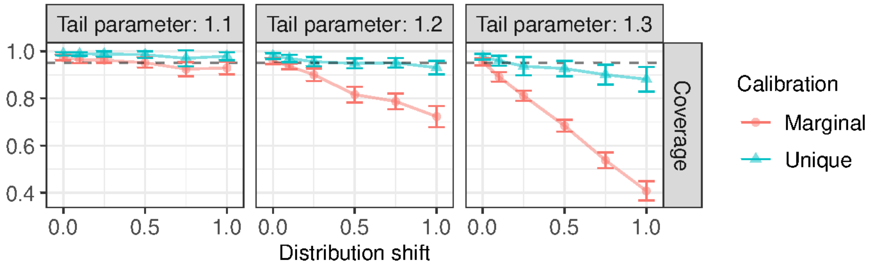

Figure 6: Performance of conformal confidence intervals with marginal (Algorithm 2) or distinct-query (Algorithm 4) coverage in a test set of size $M = 100$ with varying degrees of distribution shift. The data are sketched with the CMS-CU instead of the CMS. Other details are as in Figure 3.

of Bertsimas and Digalakis (2021) and fit a machine learning model to seek a compressed representation of the data that is designed to make frequency queries as efficient as possible. In particular, a neural network model is trained to predict the relative frequency of each object, looking only at a small fraction of the data set aside during an initial warm-up phase—see below for more details about this training data set.

After the machine learning model has been fitted, any new object is assigned to one of $w$ possible buckets based on its predicted frequency, where $w$ is a width parameter that controls the memory footprint of this sketch. If the model is informative, objects with similar frequencies should be assigned to the same bin, and this is the key idea. At the same time, a memory-efficient Bloom filter (Bloom, 1970) is used to approximately keep track of the total number of distinct objects observed in the sketched data set. Thus, a reasonable guess for the frequency of any new query can be obtained by taking the ratio between the number of objects assigned to the same hash bucket and the approximate total number of distinct objects in the hashed data given by the Bloom filter. This procedure is outlined by Algorithm A9. We also refer to Bertsimas and Digalakis (2021) for a full description of this "ML" sketching algorithm, and to our open-source software repository for technical implementation details.

These experiments follow the same setup as those in Section 6.1, simulating data from a Zipf distribution with tail parameter $a = 1.1$. Algorithm A9 is used to construct two-sided confidence intervals with fixed width, as explained in Section 3.2. The ML sketch is fitted on a training set collected in a data-driven way as to include exactly 500 distinct objects, following the same adaptive warm-up strategy presented in Section 3.4. Note that Algorithm A9 evaluates the conformity scores only on observations collected during a second independent warm-up phase. Therefore, the adaptive warm-up rule does not break the exchangeability required to obtain theoretically valid inferences.

These intervals are compared to those obtained by applying Algorithm 2 with the CMS, the CMS-CU, or the CS (Charikar et al., 2002) as baseline sketches, varying the common width of the latter as a control parameter. To keep the comparison fair, the ML sketch always uses the same amount of memory as the other sketches. This is achieved by setting

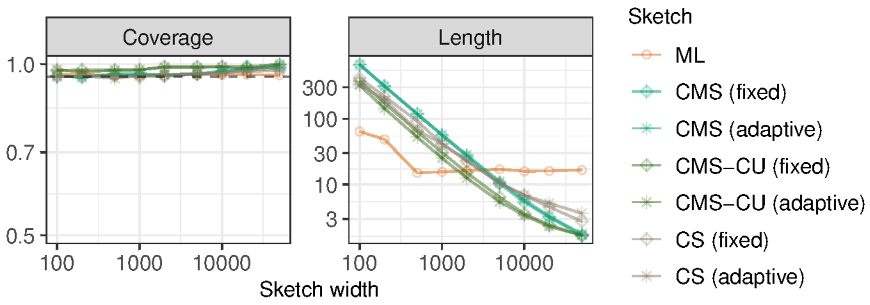

Figure 7: Performance of 95% conformal confidence intervals based on different data sketches, as a function of the sketch width. The data are simulated from a Zipf distribution with tail parameter $a = 1.1$. Other details are as in Figure 3.

the number of buckets in the ML sketch equal to 50% of the CMS, CMS-CU, and CS widths, while dedicating the remaining bits of memory to the Bloom filter. To further facilitate the comparison with Algorithm A9, the conformal confidence intervals based on the CMS, CMS-CU, and CS also utilize a similar adaptive warm-up strategy, specifically by following the heuristic approach of Algorithm A7 in Section 3.4. Note that our conformal confidence interval based on the CS sketch are always two-sided because the CS sketch provides an unbiased estimate of the query frequencies (Cormode and Yi, 2020).

The results in Figure 7 show that all methods achieve the desired 95% marginal coverage, even though the benchmark intervals based on the CMS, CMS-CU, and CS are not known to be theoretically valid due to the heuristic nature of the adaptive warm-up involved in Algorithm A7. In fact, we have observed that Algorithm A7 typically works well in practice, and that the additional data split introduced by its theoretically rigorous alternative (Algorithm A8) may often be unnecessary. The ML intervals produced by Algorithm A9 tend to be relatively more informative if the hash width is small, but they are less efficient compared to the CMS, CMS-CU, and CS as more memory becomes available.

This behavior can be explained by noting that here the performance of the ML sketch may be limited by the accuracy of the machine learning model, which is trained on a fixed number of warm-up data points that does not grow with the sketch width. While increasing the number of training observations could further improve the performance of the ML sketch, it should be kept in mind that the training set must remain small compared to the sketched set in order to avoid excessive memory usage.

In conclusion, we note that the trade-off between random hashing and data-driven sketching may generally be affected several factors, including the amount of available memory and the ease with which a machine learning model can capture useful data patterns. Therefore, different sketches are likely to be preferable in different applications, which highlights the advantage of having a flexible uncertainty estimation framework.

The results of additional experiments involving the ML sketch are in Appendix E. Figure A20 presents results similar to those in Figure 7, with the only difference that the conformal confidence intervals are designed to control frequency-range conditional coverage (10), calibrating the conformity scores separately within $L = 5$ frequency bins, instead of marginal coverage (2). Figures A21 and A22 presents qualitatively similar results from experiments analogous to those in Figures 7 and A20, respectively, in which the sketch width is fixed while the tail parameter of the Zipf distribution is varied.

### 6.6 Additional Numerical Experiments

Appendix E contains the results of supplementary experiments based on the CMS and CMS-CU applied to synthetic data from different distributions. Figures A23–A26 report on experiments based on data from a random probability measure distributed as the Pitman-Yor prior (Pitman and Yor, 1997) with a standard Gaussian base distribution and parameters $\lambda > 0$ and $\sigma \in [0, 1)$, as explained in Appendix B.4. We set $\lambda = 5000$ and vary $\sigma$. For $\sigma = 0$ the Pitman-Yor prior reduces to the Dirichlet prior (Ferguson, 1973), while $\sigma > 0$ results in heavier tails. Figures A27–A28 report on additional experiments in which our methods are applied in combination with the CS sketch (Charikar et al., 2002), in order to compress data with rare high-frequency items (*heavy hitters*). Concretely, these data are generated according to the following probability distribution: a heavy hitter $Z = 0$ is observed with probability $1/\sqrt{m}$, where $m = 100,000$; otherwise $Z \sim \text{Unif}(0, 1)$ with probability $1 - 1/\sqrt{m}$. The results show that the CS leads to more informative conformal confidence intervals compared to the CMS, CMS-CU, or ML sketches. This should not be surprising given that the CS is designed to reduce the negative impact of random hash collisions in such a way as to make frequency queries about heavy hitters relatively more accurate (Charikar et al., 2002). Finally, Figures A29–A32 show the results of simulations involving two-sided confidence intervals, whose detailed setup is explained in Appendix F.

## 7. Illustrations on Empirical Data

Section 7.1 presents illustrations based on 16-mers data in SARS-CoV-2 DNA sequences, while Section 7.2 focuses on counting 2-grams in an English literature data set.

### 7.1 Analysis of 16-Mers in SARS-CoV-2 DNA Sequences

This illustration involves a data set of nucleotide sequences from SARS-CoV-2 viruses made publicly available by the National Center for Biotechnology Information (Hatcher et al., 2017). The data include 43,196 sequences, each consisting of approximately 30,000 nucleotides. The goal is to estimate the empirical frequency of each *16-mer*, a distinct sequence of 16 DNA bases in contiguous nucleotides. Given that each nucleotide has one of 4 bases, there are $4^{16} \approx 4.3$ billion possible 16-mers. Thus, exact tracking of all 16-mers is not unfeasible, which allows us to validate the sketch-based queries. Sequences containing missing values are removed during pre-processing, for simplicity.

The experiments are carried out as in Section 6.1, with the difference that a larger sample of size 1,000,000 is sketched using the CMS-CU due to its higher efficiency; the width $w$ of the hash functions is varied as a control parameter. All 16-mers are processed in a random order, which ensures their exchangeability. Figure 8 compares the performances of all methods as a function of the hash width, in terms of marginal coverage and mean confidence interval width.

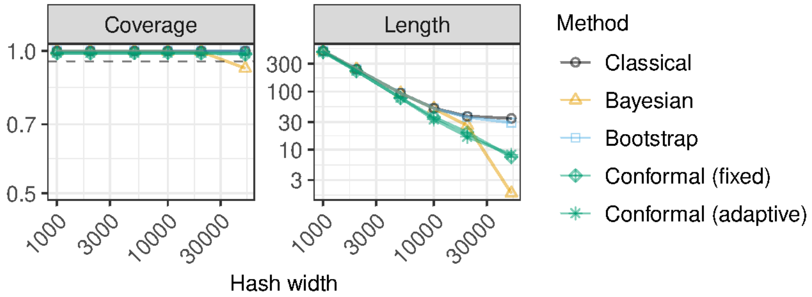

Figure 8: Performance of confidence intervals based on SARS-CoV-2 sequence data. The results are shown as a function of the hash width. The data are sketched with the CMS-CU instead of the CMS. Other details are as in Figure 3.

All methods achieve the desired marginal coverage, except for the Bayesian approach when $w$ is large. For small $w$, all methods return intervals of similar width, because the distribution of SARS-CoV-2 16-mers frequencies is quite concentrated with relatively narrow support (Figure A33), which makes it difficult to compress the data without much loss.

By contrast, the conformal methods yield noticeably shorter confidence intervals if $w$ is large. Figure A34 reports the same results stratified by the frequency of the queried objects. Table A1 lists 10 common and 10 rare queries along with their corresponding deterministic upper bounds for $w = 50,000$, comparing the lower bounds obtained with each method. Table A2 shows analogous results with $w = 5,000$. Figure A35 confirms the advantage of sketching with the CMS-CU instead of the CMS. Similarly, Figure A36 shows the CMS-CU also typically leads to more informative frequency queries compared to the ML sketch discussed in Section 6.5, unless the available memory is very low.

Figure A37 compares the performances of different frequency *point-estimates* in terms of mean absolute deviation from the true frequency. With the classical method, we take the midpoint of the 95% confidence interval as a point estimate, although other approaches are also possible (Cormode and Yi, 2020). For the other methods, the point estimate is the lower confidence bound at level $\alpha = 0.5$; in the Bayesian case, it is the posterior median. Although a conformal lower bound with $\alpha = 0.5$ is not always a reliable estimator of conditional medians (Medarametla and Candès, 2021), this approach outperformed the benchmarks in all of our experiments.

Figure A38 shows the confidence intervals reported in Figure 8 approximately remain valid even if their average coverage is evaluated with respect to distinct queries only; of course, this is not generally guaranteed and may not always be true on other data sets, as seen in Section 6. Figure A39 shows the performance of the procedure described in Algorithm 4 for constructing conformal confidence intervals with valid coverage for distinct queries. These results show that Algorithm 4 leads to valid inference across a wide range of values of its parameter $M'$—the size of the calibration shards—despite the more pessimistic worst-case predictions of Theorem 7. Finally, Figure A40 investigates the robustness of the

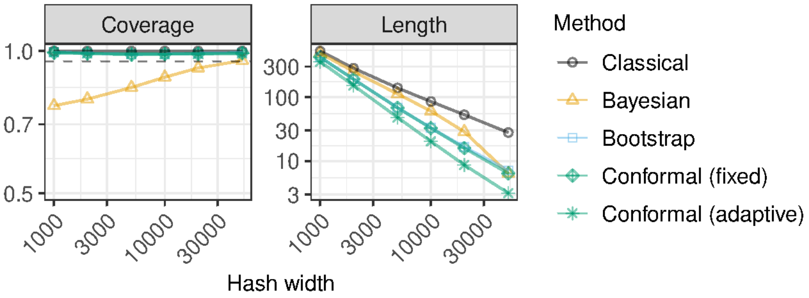

Figure 9: Performance of confidence intervals for random queries, for a sketched data set of English 2-grams in classic English literature. Other details are as in Figure 8.

alternative types of conformal prediction intervals output by Algorithm 2 and Algorithm 4 to distribution shifts in the test queries, similarly to Figure 6.

### 7.2 Analysis of 2-Grams in an English Literature Data Set

This example is based on data consisting of 18 open-domain pieces of classic English literature downloaded from the Gutenberg Corpus (Project Gutenberg, 1971-present) using the NLTK Python package (Bird et al., 2009). The goal is to count the frequencies of all *2-grams*—consecutive pairs of English words. After basic pre-processing to remove punctuation and unusual words (only those in a relatively small dictionary of 25,487 common English words are retained), there are approximately 1,700,000 remaining 2-grams—the total number of all *possible* 2-grams within this dictionary is approximately 650,000,000.

Note that such pre-processing does not remove very common words (such as “the”, “or”, etc.) and it may sometimes lead to unnatural 2-grams whenever a relatively rare word is removed from an otherwise meaningful sentence (e.g., “very uncommon for” would become “very for”). Therefore, our analysis is not fully realistic from a natural language processing perspective but it is computationally efficient and still informative regarding the performance of our uncertainty estimation method. With this setup, the same experiments are then carried out as in Section 7.1, sketching 1,000,000 randomly sampled 2-grams with the CMS-CU and querying 10,000 independent 2-grams. As in the previous experiments, the 2-grams are processed in a random order to ensure exchangeability.

Figure 9 shows the conformal intervals produced by Algorithm 3 using adaptive scores achieve the desired 95% marginal coverage and tend to have the shortest width. By contrast, the Bayesian intervals are not valid unless the hashes are very wide. Here, the conformal approach enjoys a larger improvement in performance compared to the other approaches because these data can be compressed efficiently due to the weaker power-law tail behavior of the frequency distribution of English 2-grams; see Figure A33. Further, Figures A41–A44 and Tables A1–A2 report additional results along the lines of those in the previous section, including empirical evidence of valid frequency-conditional coverage and a comparison of the performances of different linear and non-linear sketches.

Figure A45 shows the confidence intervals reported in Figure 9 approximately remain valid even if their average coverage is evaluated with respect to distinct queries only; of course, this is not guaranteed in general. Figure A46 illustrates the performance of Algorithm 4, showing that valid inference for distinct queries can be achieved with a wide range of the parameter $M'$, despite the more pessimistic worst-case predictions of Theorem 7. Finally, Figure A47 investigates the robustness of the alternative types of conformal intervals output by Algorithms 2 and 4 to distribution shifts in the test queries, similarly to Figure 6.

## 8. Discussion

This work opens several opportunities for further research. In the future one may study and compare theoretically, in some settings, the length of our conformal confidence intervals under different types of coverage guarantees. A possible approach may take inspiration from relevant work in the context of regression by Lei et al. (2018) and Sesia and Candès (2020).

Further, it would be interesting to explore the relevance of the methods and theory presented in Section 5 beyond sketching. For example, the results of Section 5 could be repurposed to construct conformal prediction sets for regression or multi-class classification tasks that achieve valid coverage over subsets of individual test cases with certain unique attributes. In those contexts, our work may lead to an alternative framework for dealing with uncertainty estimation under algorithmic fairness constraints (Romano et al., 2020a) or stratified sampling mechanisms (Dunn et al., 2022; Park et al., 2022).

Finally, the uncertainty estimation methods developed in this paper may also be relevant for more general forms of randomized sketching used for other numerical, statistical, and learning problems (Vempala, 2005; Halko et al., 2011; Mahoney, 2011; Woodruff, 2014; Drineas and Mahoney, 2016; Martinsson and Tropp, 2020); see e.g., Dobriban and Liu (2019); Liu and Dobriban (2019); Lacotte and Pilanci (2020); Yang et al. (2021).

## Software and Computations

Accompanying software and data are available online at `https://github.com/msesia/conformalized-sketching`. Experiments were carried out in parallel on a computing cluster; each experiment required less than a few hours with a standard CPU and less than 5GB of memory (20 GB are needed for the analysis of the SARS-CoV-2 DNA data).

### Acknowledgements

M. S. is supported in part by NSF grant DMS 2210637 and by an Amazon Research Award. S. F. is also affiliated to IMATI-CNR “Enrico Magenes” (Milan, Italy), and received funding from the European Research Council under the European Union’s Horizon 2020 research and innovation program under grant 817257. E. D. is supported in part by the NSF DMS 2046874 (CAREER) award, and ONR grant N00014-21-1-2843. The authors are grateful to the three anonymous referees for their constructive feedback.

## Appendix A. Relevant Background on the Count-Min Sketch

### A.1 The CMS Algorithm

The count-min sketch (CMS) of Cormode and Muthukrishnan (2005) compresses a data set by applying to each observation $d \geq 1$ different $w$-wide *hash* functions $h_j : \mathscr{X} \to [w] := \{1, \ldots, w\}$, for all $j \in [d] := \{1, \ldots, d\}$ and some integer number of buckets $w \geq 1$. Each hash function maps the elements of $\mathscr{X}$ into one of $w$ buckets, so that distinct values of $z$ populate the buckets approximately uniformly. Hash functions are typically chosen at random from a *pairwise independent* family $\mathcal{H}$. This ensures the probability (over the randomness in the choice of hash functions) that two distinct objects $z_1, z_2 \in \mathscr{X}$ are mapped by two different hash functions into the same bucket is $1/w^2$. The data $Z_1, \ldots, Z_m$ are thus compressed into a sketch matrix $C \in \mathbb{N}^{d \times w}$ with rows summing to $m$. The element in the $j$-th row and $k$-th column of $C$ counts the data points mapped by the $j$-th hash function into the $k$-th bucket:

$$C_{j,k} = \sum_{i=1}^{m} \mathbb{1}\left[h_j(Z_i) = k\right], \qquad j \in [d],\, k \in [w]. \tag{19}$$

One chooses $d$ and $w$ such that $d \cdot w \ll m$, and thus the matrix $C$ loses information compared to the full data set; however, it has the advantage of requiring much less space to store.

Given a sketch $C$ from (19), we are interested in estimating the empirical frequency of an object $z \in \mathscr{X}$, as defined in (1). A typical point estimate is the smallest count among the $d$ buckets into which $z$ is mapped:

$$\hat{f}^{\text{CMS}}_{\text{up}}(z) = \min_{j \in [d]} \left\{ C_{j,h_j(z)} \right\}. \tag{20}$$

This procedure is outlined by Algorithm A5.

**Algorithm A5** CMS

```
  **Input:** Data $Z_1, \ldots, Z_m$. Sketch dimensions $d, w$. Hash functions $h_1, \ldots, h_d$. Query $z$.
  **Initialize:** $C_{j,k} = 0$ for all $j \in [d], k \in [w]$.
  **for** $i = 1, \ldots, m$ **do**
      **for** $j = 1, \ldots, d$ **do**
          **Increment** $C_{j,h_j(Z_i)} \leftarrow C_{j,h_j(Z_i)} + 1$
  **Compute** $\hat{f}^{\text{CMS}}_{\text{up}}(z) = \min_{j \in [d]}\{C_{j,h_j(z)}\}$.
  **Output:** deterministic upper-bound for the frequency of $z$ in the data set: $\hat{f}^{\text{CMS}}_{\text{up}}(z)$.
```

### A.2 Classical Upper and Lower Bounds for CMS Frequency Queries

As $\hat{f}^{\text{CMS}}_{\text{up}}(z) \geq f_m(z)$, the expression in (20) always gives a deterministic upper bound for $f_m(z)$; see Cormode and Muthukrishnan (2005). Although $\hat{f}^{\text{CMS}}_{\text{up}}(z)$ may be larger than $f_m(z)$ due to hash collisions, the independence of the hash functions still enables the following classical probabilistic lower bound for $f_m(z)$. Cormode and Muthukrishnan (2005) showed that for any $\delta, \varepsilon \in (0, 1)$, choosing $d = \lceil -\log \delta \rceil$ and $w = \lceil e/\varepsilon \rceil$, for any fixed $z \in \mathscr{X}$, and with $\hat{f}^{\text{CMS}}_{\text{up}}(z)$ from (19),

$$\mathbb{P}_{\mathcal{H}}[f_m(z) \geq \hat{f}^{\text{CMS}}_{\text{up}}(z) - \varepsilon m] \geq 1 - \delta. \tag{21}$$

For example, if $\delta = 0.05$ and thus $d = 3$, this says that $\hat{f}_{\text{up}}^{\text{CMS}}(z) - m \cdot \lceil e/w \rceil$ is a lower bound on $f_m(z)$ with 95% probability. The subscript $\mathcal{H}$ in the bound (21) clarifies that the randomness is with respect to the hash functions, while $Z_1, \dots, Z_m$ and $z$ are fixed. This bound can be useful to inform the choices of $d$ and $w$ prior to sketching, but it is not fully satisfactory as a way of quantifying the uncertainty about the true frequency of a given query. First, it is often too conservative (Ting, 2018) if the data are randomly sampled from some distribution as opposed to being arbitrary and potentially worst-case. Second, it is not flexible: $\delta$ cannot be chosen by the practitioner because it is fixed by $d$, and $\varepsilon$ is uniquely determined by the hash width. Thus, the bound in (21) does not always give practically useful confidence intervals.

### A.3 Bootstrap Confidence Intervals for CMS Frequency Queries

An alternative approach to computing lower and upper bounds for $f_m(z)$ using the CMS was proposed by Ting (2018), in order to address the often excessive conservativeness of the classical bounds described above. The method of Ting (2018) is based on bootstrapping, and departs from classical analysis of the CMS as it leverages randomness in the data instead of randomness in the hash functions. Precisely, it assumes the data and the queried object are an independent and identically distributed (i.i.d.) random sample from some unknown distribution. This condition means that one is interested in the typical behavior of the algorithm over certain scenarios described by the distribution. The condition does not always apply but, when it does, it can be extremely useful because it leads to much more informative confidence intervals. In fact, the confidence intervals described by Ting (2018) are nearly exact for the CMS, up to a finite-sample discrepancy between the bootstrap and population distributions.

A limitation of the bootstrap approach is that it relies on the specific *linear* structure of the CMS—the sketch matrix $C$ in (20) is a linear combination of the true frequencies of all objects in the data set—and is not easily extendable to other sketching algorithms that may outperform the CMS in practice. For example, the CMS is relatively sensitive to random hash collisions, which can result in overly conservative deterministic upper bounds. This challenge has motivated the development of alternative *non-linear* algorithms, such as the CMS with *conservative updates* (CMS-CU) of Estan and Varghese (2002) which we briefly review below.

### A.4 The CMS-CU Algorithm

The difference between the CMS (Cormode and Muthukrishnan, 2005) and the CMS-CU (Estan and Varghese, 2002) is that, whenever a new object $z$ is sketched by the latter, only the row of $C$ with the smallest value of $C_{j,h_j(z)}$ is updated, while the other counters remain unaltered. Then, a valid deterministic upper bound for the CMS-CU can be calculated with the same formula in (20). This procedure is outlined in Algorithm A6. While the CMS-CU can lead to higher query accuracy compared to the vanilla CMS (Estan and Varghese, 2002), the theoretical analysis of the CMS-CU beyond a deterministic upper bound is more challenging, and it appears to be a relatively less explored topic.

**Algorithm A6** CMS-CU

```
  **Input:** Data $Z_1, \dots, Z_m$. Sketch dimensions $d, w$. Hash functions $h_1, \dots, h_d$. Query $z$.
  **Initialize:** $C_{j,k} = 0$ for all $j \in [d], k \in [w]$.
  **for** $i = 1, \dots, m$ **do**
    **Compute** $j^* = \arg\min_{j \in [d]} C_{j,h_j(Z_i)}$.
    **Increment** $C_{j^*,h_{j^*}(Z_i)} \leftarrow C_{j^*,h_{j^*}(Z_i)} + 1$
  **Compute** $\hat{f}_{\mathrm{up}}^{\mathrm{CMS-CU}}(z) = \min_{j \in [d]} \{C_{j,h_j(z)}\}$.
  **Output:** deterministic upper-bound for the frequency of $z$ in the data set: $\hat{f}_{\mathrm{up}}^{\mathrm{CMS-CU}}(z)$.
```

## Appendix B. Additional Methodological Details

### B.1 Constructing Two-Sided Conformal Confidence Intervals

This section describes two alternatives methods for constructing two-sided conformal confidence intervals. The first method, explained in Appendix B.1.1, consists of separately calibrating two sequences of lower and upper one-sided confidence intervals, each adopting the significance level $\alpha/2$ instead of $\alpha$. This is relatively easy to implement but may be less efficient than the second method, explained in Appendix B.1.2, which consists of directly calibrating a sequence of nested two-sided intervals.

#### B.1.1 Construction Based on Bonferroni Correction

One approach to building two-sided conformal confidence intervals for $f_m(X_{m+1})$ at level $1-\alpha$ consists of constructing a pair of lower and upper one-sided confidence intervals at level $1-\alpha/2$. In particular, consider the following two nested sequences $S_t^l$ and $S_t^u$ of one-sided confidence intervals, each indexed by a scalar parameter $t$:

$$S_t^l = [\hat{L}_{m,\alpha/2}(X_{m+1}; t), \hat{f}_{\text{up}}^{\text{CMS}}(X_{m+1})], \qquad\qquad S_t^u = [0, \hat{U}_{m,\alpha/2}(X_{m+1}; t)],$$

where $\hat{f}_{\text{up}}^{\text{CMS}}(X_{m+1})$ is a deterministic upper bound for the unknown true empirical frequency of $X_{m+1}$; e.g., see Appendix A.1. The sequences $S_t^l$ and $S_t^u$ can be separately calibrated using the conformal inference method described in Sections 3 and 4, for any given choice of frequency-range partition $\mathcal{B}$, as we shall make more precise below. This gives two distinct data-adaptive thresholds $\hat{Q}^{*,l}_{n,1-\alpha/2}$ and $\hat{Q}^{*,u}_{n,1-\alpha/2}$, respectively, such that, $\forall B \in \mathcal{B}$,

$$\mathbb{P}\left[f_m(X_{m+1}) \geq \hat{L}_{m,\alpha/2}(X_{m+1}; \hat{Q}^{*,l}_{n,1-\alpha/2}) \mid f_m(Z_{m+1}) \in B\right] \geq 1 - \frac{\alpha}{2},$$

and

$$\mathbb{P}\left[f_m(X_{m+1}) \leq \hat{U}_{m,\alpha/2}(X_{m+1}; \hat{Q}^{*,u}_{n,1-\alpha/2}) \mid f_m(Z_{m+1}) \in B\right] \geq 1 - \frac{\alpha}{2}.$$

By a union bound, we obtain that the following two-sided conformal confidence interval has valid coverage, in the sense of (10), at level $1-\alpha$:

$$[\hat{L}_{m,\alpha/2}(X_{m+1}; \hat{Q}^{*,l}_{n,1-\alpha/2}), \hat{U}_{m,\alpha/2}(X_{m+1}; \hat{Q}^{*,u}_{n,1-\alpha/2})].$$

Different practical implementations are available to construct the sequences of candidate lower bounds $\hat{L}_{m,\alpha/2}(X_{m+1}; t)$ and upper bounds $\hat{U}_{m,\alpha/2}(X_{m+1}; t)$. Two concrete examples are explained below.

**Constant conformity scores.** A simple option to construct $\hat{L}_{m,\alpha/2}(X_{m+1}; t)$ is to directly apply the method described in Section 3.2, for example by shifting $\hat{f}_{\text{up}}^{\text{CMS}}(X_{m+1})$ downward by a constant $t$. Then, the conformalized threshold $\hat{Q}^{*,l}_{n,1-\alpha/2}$ can be calibrated as usual. The sequence of candidate upper bounds $\hat{U}_{m,\alpha/2}(X_{m+1}; t)$ can also be constructed similarly to $\hat{L}_{m,\alpha/2}(X_{m+1}; t)$, for example by adding a constant $t$ to the trivial lower bound of 0, up to the deterministic upper bound $\hat{f}_{\text{up}}^{\text{CMS}}(X_{m+1})$. The threshold $\hat{Q}^{*,u}_{n,1-\alpha/2}$ for $\hat{U}_{m,\alpha/2}(X_{m+1}; t)$ can then be calibrated as usual with Algorithm 2.

**Bootstrap conformity scores.** An alternative option to construct the sequence $\hat{L}_{m,\alpha/2}(X_{m+1};t)$ consists of shifting downward by a constant $t$ the bootstrap lower bound calculated with the method of Ting (2018), at level $\alpha/2$. Similarly, the sequence $\hat{U}_{m,\alpha/2}(X_{m+1};t)$ can be obtained by shifting upward by a constant $t$ the analogous bootstrap upper bound at level $1-\alpha/2$. Thus, in the special case of the vanilla CMS, our conformal confidence intervals based on these scores intuitively become very similar to the bootstrap confidence intervals of Ting (2018). In general, however, the difference remains that the intervals of Ting (2018) rely on the linearity of the CMS, while ours are theoretically valid regardless of how the data are sketched. We have observed this option works well in practice, at least within the scope of our numerical experiments. Therefore, this is the implementation adopted in our numerical experiments described in Section F.

### B.1.2 Construction Based on Conditional Histograms

Two-sided conformal confidence intervals for $f_m(X_{m+1})$ can be constructed by following the general recipe outlined in Section 3.1. To implement this method practically, one needs to fix an increasing sequence of candidate intervals $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$, depending on $Z_{m+1}$ and $\phi(Z_{n+1},\dots,Z_m)$. Possible choices for such sequence may be directly borrowed from the existing literature on conformal inference for regression, including for example the quantile regression approach of Romano et al. (2019) or the conditional histogram approach of Sesia and Romano (2021). Here, we describe a particular implementation that combines the idea in Sesia and Romano (2021) with a Bayesian model, in continuity with the works of Cai et al. (2018) and Dolera et al. (2021) on Bayesian empirical frequency estimation from sketched data. However, the same idea could easily accommodate a quantile regression model or any other machine learning algorithm instead of the Bayesian model, as explained in Sesia and Romano (2021). Note that the following paragraphs largely retrace the same steps as in Sesia and Romano (2021), which are however useful to recap here to make the presentation self contained.

For any $j \in [m]$, let $\hat{\varphi}_j(x)$ indicate the posterior probability of $f_m(X_{m+1}) = j$ for $X_{m+1} = x$ as estimated by any Bayesian model for frequency estimation given sketched data, such as that of Cai et al. (2018) based on a Dirichlet process prior, for example. For convenience of notation, we will sometimes refer to the full posterior distribution of $f_m(X_{m+1})$ simply as $\hat{\varphi}$. Note that, in general, the form of the posterior distribution $\hat{\varphi}$ may depend on $m$ as well as on the sketched data in $\phi(Z_{n+1},\dots,Z_m)$. Following in the footsteps of Sesia and Romano (2021), define the following bi-valued function $\mathcal{S}$ taking as input a query $x$, the posterior distribution $\hat{\varphi}$, a scalar threshold $t \in [0,1]$, and two intervals $S^-, S^+ \subseteq \{1,\dots,m\}$:

$$\mathcal{S}(x,\hat{\varphi},S^-,S^+,t) := \underset{(l,u)\in\{1,\dots,m\}^2 \,:\, l\leq u}{\arg\min} \left\{ |u-l| : \sum_{j=l}^{u} \hat{\varphi}_j(x) \geq t,\ S^- \subseteq [l,u] \subseteq S^+ \right\}. \quad (22)$$

Above, it is implied that we choose the value of $(l,u)$ minimizing $\sum_{j=l}^{u} \hat{\varphi}_j(x)$ among the feasible ones with minimal $|u-l|$, whenever the optimization problem does not have a unique solution. Therefore, we can assume without loss of generality that (22) has a unique solution; if that is not the case, we can break the ties at random by adding a little noise to $\hat{\varphi}$. As explained in Sesia and Romano (2021), the problem defined in (22) can be solved efficiently, at computational cost linear in $m$. Note that we will sometimes refer to subintervals of $[m]$ as either contiguous subsets of $\{1,\dots,m\}$ (e.g., $S^-$) or as pairs of lower and upper endpoints (e.g., $[l,u]$).

If $S^- = \emptyset$ and $S^+ = \{1,\dots,m\}$, the expression in (22) computes the shortest interval with total posterior probability mass above $t$. In general, the optimization in (22) involves the additional *nesting* constraint that the output $\mathcal{S}$ must satisfy $S^- \subseteq \mathcal{S} \subseteq S^+$, which will be needed to guarantee the resulting sequence of confidence intervals indexed by $t$ is nested. Note that the inequality in (22) involving $t$ may not be binding at the optimal solution due to the discrete nature of the optimization problem. However, the above construction could be easily modified by introducing some suitable randomization leading to confidence intervals that are even tighter on average, as explained in Sesia and Romano (2021).

For any integer $T \geq 1$, consider an increasing sequence $t_\tau \in [0,1]$, for $\tau \in \{0,\dots,T\}$. A nested sequence of $T$ intervals indexed by $\tau \in \{0,\dots,T\}$, which may be written as

$$S_t = \left[\hat{L}_{m,\alpha}(X_{m+1};t_\tau), \hat{U}_{m,\alpha}(X_{m+1};t_\tau)\right],$$

for appropriate endpoints $\hat{L}_{m,\alpha}(X_{m+1};t_\tau)$ and $\hat{U}_{m,\alpha}(X_{m+1};t_\tau)$, respectively, is then constructed from (22) as follows. First, fix any *starting index* $\bar{\tau} \in \{0,1,\dots,T\}$ and define $S_{\bar{\tau}}$ by applying (22) without the nesting constraints (with $S^- = \emptyset$ and $S^+ = \{1,\dots,m\}$):

$$S_{\bar{\tau}} := \mathcal{S}(x, \hat{\varphi}, \emptyset, \{1,\dots,m\}, t_{\bar{\tau}}), \tag{23}$$

Note the explicit dependence on $x$ and $\hat{\varphi}$ of the left-hand-side above is omitted for simplicity, although it is important to keep in mind that $S_{\bar{\tau}}$ does of course depend on these quantities.

Having computed the initial interval $S_{\bar{\tau}}$, we recursively extend the definition to the wider intervals indexed by $\tau = \bar{\tau}+1,\dots,T$ as follows:

$$S_\tau := \mathcal{S}(x, \hat{\varphi}, S_{\tau-1}, \{1,\dots,m\}, t_\tau).$$

See Sesia and Romano (2021) for a schematic visualization of this step. Similarly, the narrower intervals $S_\tau$ indexed by $\tau = \bar{\tau}-1, \bar{\tau}-2, \dots, 0$ are defined recursively as:

$$S_\tau := \mathcal{S}(x, \hat{\varphi}, \emptyset, S_{\tau+1}, t_\tau).$$

See Sesia and Romano (2021) for a schematic visualization of this step. As a result of this construction, the sequence of intervals $\{S_\tau\}_{\tau=0}^T$ is nested regardless of the starting point $\bar{\tau}$ in (23), for which a typical choice is such that $t_{\bar{\tau}} = 1-\alpha$. Then, two-sided conformal confidence intervals for $f_m(X_{m+1})$ can be obtained by applying Algorithm 2 with this particular sequence of input nested intervals. We refer to Sesia and Romano (2021) for further details on the construction of nested intervals outlined above.

### B.2 Conformalized Sketching with Adaptive Warm-Up Period

**Algorithm A7** Conformalized sketching with adaptive warm-up period (heuristic)

```
Input: Data set $Z_1, \dots, Z_m$. Sketching function $\phi$.
       Number $n_0 \ll m$ of unique objects to be observed during the warm-up phase.
       A (trainable) predictor to compute nested intervals $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]_{t \in \mathcal{T}}$.
       Number of data points $n^{\text{train}} < n$ used for training $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$.
Initialize a sparse counter $f_n^{\text{wu}}(z) = 0, \forall z \in \mathcal{Z}$.
for $i_{\text{wp}} = 1, \dots, m$ do
    Increment $f_n^{\text{wu}}(Z_i) \leftarrow f_n^{\text{wu}}(Z_i) + 1$.
    if Number of unique observed objects $\geq n_0$ then
        Break
Set $n = i_{\text{wp}}$.
Initialize a sparse counter $f_{m-n}^{\text{sv}}(z) = 0, \forall z \in \mathcal{Z}$.
Initialize an empty sketch $\phi(\emptyset)$.
for $i = n+1, \dots, m$ do
    Update the sketch $\phi$ with the new observation $Z_i$.
    if $f_n^{\text{wu}}(Z_i) > 0$ then
        Increment $f_{m-n}^{\text{sv}}(Z_i) \leftarrow f_{m-n}^{\text{sv}}(Z_i) + 1$.
for $i = 1, \dots, n$ do
    Set $X_i = (Z_i, \phi(Z_{n+1}, \dots, Z_m))$ as in (8).
    Set $Y_i = f_{m-n}^{\text{sv}}(Z_i)$.
Train $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$ using the data in $\{(X_i, Y_i)\}_{i=1}^{n^{\text{train}}}$.
for $i = n^{\text{train}} + 1, \dots, n$ do
    Compute the conformity score $E(X_i, Y_i)$ with (3), using $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$.
Output: Data sketch $\phi$;
        Sparse counter $f_n^{\text{wu}}(z), \forall z \in \mathcal{Z}$;
        Trained predictor $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$;
        Conformity scores $E(X_i, Y_i)$ for all $i \in \{n^{\text{train}} + 1, \dots, n\}$.
```

**Algorithm A8** Conformalized sketching with two-step adaptive warm-up period

```
Input: Data set $Z_1, \dots, Z_m$. Sketching function $\phi$.
       Number $n_0 \ll m$ of unique objects to be observed during the warm-up phase.
       A (trainable) predictor to compute nested intervals $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]_{t \in \mathcal{T}}$.
       Number of data points $n^{\text{train}} < n$ used for training $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$.
Initialize a sparse counter $f_n^{\text{wu}}(z) = 0, \forall z \in \mathcal{Z}$.
for $i_{\text{wp}} = 1, \dots, m$ do
  Increment $f_n^{\text{wu}}(Z_i) \leftarrow f_n^{\text{wu}}(Z_i) + 1$.
  if Number of unique observed objects $\geq n_0$ then
    Break
Set $n = i_{\text{wp}}$.
Initialize a sparse counter $f_n^{\text{wu},2}(z) = 0, \forall z \in \mathcal{Z}$.
for $i = n + 1, \dots, 2n$ do
  Increment $f_n^{\text{wu}}(Z_i) \leftarrow f_n^{\text{wu}}(Z_i) + 1$.
  Increment $f_n^{\text{wu},2}(Z_i) \leftarrow f_n^{\text{wu},2}(Z_i) + 1$.
Initialize a sparse counter $f_{m-n}^{\text{sv}}(z) = 0, \forall z \in \mathcal{Z}$.
Initialize an empty sketch $\phi(\emptyset)$.
for $i = 2n + 1, \dots, m$ do
  Update the sketch $\phi$ with the new observation $Z_i$.
  if $f_n^{\text{wu},2}(Z_i) > 0$ then
    Increment $f_{m-n}^{\text{sv}}(Z_i) \leftarrow f_{m-n}^{\text{sv}}(Z_i) + 1$.
for $i = n + 1, \dots, 2n$ do
  Set $X_i = (Z_i, \phi(Z_{n+1}, \dots, Z_m))$ as in (8).
  Set $Y_i = f_{m-n}^{\text{sv}}(Z_i)$.
Train $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$ using the data in $\{(X_i, Y_i)\}_{i=n+1}^{n+n^{\text{train}}}$.
for $i = n + n^{\text{train}} + 1, \dots, 2n$ do
  Compute the conformity score $E(X_i, Y_i)$ with (3), using $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$.
Output: Data sketch $\phi$;
        Sparse counter $f_n^{\text{wu}}(z), \forall z \in \mathcal{Z}$;
        Trained predictor $[\hat{L}_{m,\alpha}(\cdot; t), \hat{U}_{m,\alpha}(\cdot; t)]$;
        Conformity scores $E(X_i, Y_i)$ for all $i \in \{n + n^{\text{train}} + 1, \dots, 2n\}$.
```

### B.3 Conformalized Sketching with ML Algorithms

**Algorithm A9** Conformalized sketching with data-driven ML sketch

```
Input: Data set $Z_1, \dots, Z_m$.
       Number $n_0 \ll m$ of unique objects to be observed during the warm-up phase.
       A trainable model $\mathcal{M}$ to predict the relative frequency of an object.
       A Bloom filter $\mathcal{F}$.
       A (trainable) predictor to compute nested intervals $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]_{t\in\mathcal{T}}$.
       Number of data points $n^{\text{train}} < n$ used for training $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$.
Initialize a sparse counter $f_n^{\text{wu}}(z) = 0, \forall z \in \mathcal{Z}$.
for $i_{\text{wp}} = 1, \dots, m$ do
  Increment $f_n^{\text{wu}}(Z_i) \leftarrow f_n^{\text{wu}}(Z_i) + 1$.
  if Number of unique observed objects $\geq n_0$ then
    Break
Set $n = i_{\text{wp}}$.
Train the model $\mathcal{M}$ using the data in $\{(Z_i, f_n^{\text{wu}}(Z_i))\}_{i\in[1,\dots,n]}$.
Initialize the ML sketch $\phi$ based on $\mathcal{M}$ and $\mathcal{F}$, as explained in Section E.3.
Initialize a sparse counter $f_n^{\text{wu},2}(z) = 0, \forall z \in \mathcal{Z}$.
for $i = n+1, \dots, 2n$ do
  Increment $f_n^{\text{wu}}(Z_i) \leftarrow f_n^{\text{wu}}(Z_i) + 1$.
  Increment $f_n^{\text{wu},2}(Z_i) \leftarrow f_n^{\text{wu},2}(Z_i) + 1$.
Initialize a sparse counter $f_{m-n}^{\text{sv}}(z) = 0, \forall z \in \mathcal{Z}$.
Initialize an empty sketch $\phi(\emptyset)$.
for $i = 2n+1, \dots, m$ do
  Update the sketch $\phi$ with the new observation $Z_i$.
  if $f_n^{\text{wu},2}(Z_i) > 0$ then
    Increment $f_{m-n}^{\text{sv}}(Z_i) \leftarrow f_{m-n}^{\text{sv}}(Z_i) + 1$.
for $i = n+1, \dots, 2n$ do
  Set $X_i = (Z_i, \phi(Z_{n+1}, \dots, Z_m))$ as in (8).
  Set $Y_i = f_{m-n}^{\text{sv}}(Z_i)$.
Train $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$ using the data in $\{(X_i, Y_i)\}_{i=n+1}^{n+n^{\text{train}}}$.
for $i = n + n^{\text{train}} + 1, \dots, 2n$ do
  Compute the conformity score $E(X_i, Y_i)$ with (3), using $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$.
Output: Data sketch $\phi$;
        Sparse counter $f_n^{\text{wu}}(z), \forall z \in \mathcal{Z}$;
        Trained predictor $[\hat{L}_{m,\alpha}(\cdot;t), \hat{U}_{m,\alpha}(\cdot;t)]$;
        Conformity scores $E(X_i, Y_i)$ for all $i \in \{n + n^{\text{train}} + 1, \dots, 2n\}$.
```

### B.4 Sampling from a Pitman-Yor Predictive Distribution

The data points are sampled sequentially from the following predictive distribution, which has parameters $\lambda > 0$ and $\sigma \in [0, 1)$. After sampling $Z_1$ from a standard normal distribution, $\mathcal{N}(0, 1)$, fix any $i \geq 1$ and let $Z_1, \dots, Z_i$ indicate the data stream observed up to that point. Denote by $k_i$ the number of distinct elements within it, and by $V_i = (V_{i,1}, \dots, V_{i,k_i})$ the set of such distinct values. Further, let $c_{i,l}$ indicate the number of times that object $V_{i,l}$ has been observed in $Z_1, \dots, Z_i$, for $l \in \{1, \dots, k_i\}$. Then, $Z_{i+1}$ is generated as follows:

$$Z_{i+1} \mid Z_1, \dots, Z_i = \begin{cases} V_{i,l}, & \text{with probability } \frac{c_{i,l}-\sigma}{\lambda+i}, \text{ for } l \in \{1, \dots, k_i\}, \\ \mathcal{N}(0, 1), & \text{with probability } \frac{\lambda+k_i\sigma}{\lambda+i}. \end{cases}$$

Above, the second case which occurs with probability $(\lambda + k_i\sigma)/(\lambda + i)$ corresponds to sampling a new unique value from the standard normal distribution.

## Appendix C. Auxiliary Theoretical Results

### C.1 Probability Distribution of the Set of Uniques

Note that the size of $V$ is between 1 and $M$; and the values taken by $V$ range over subsets $\{a_{j_1}, \dots, a_{j_k}\} \subseteq \mathcal{Z}$, where $1 \leq k \leq M$ and $j_1, \dots, j_k \in \mathbb{N}$ are distinct indices.

**Proposition A9 (Probability distribution of the set of uniques)** *Let $\mathcal{Z}^{test}$ be an i.i.d. sample of size $M$ from a discrete distribution $P_Z = \sum_{i \in \mathbb{N}} p_j \delta_{a_j}$, where $a_j \in \mathcal{Z}$ are distinct, and $p_j \geq 0$ for all $j \in \mathbb{N}$. Let $P_Z^{[M]}$ be the probability distribution of $\mathcal{Z}^{test}$. Let $V = \textsc{Unique}(\mathcal{Z}^{test})$ denote the set of unique values in $\mathcal{Z}^{test}$. For any $1 \leq k \leq M$, and any distinct indices $j_1, \dots, j_k \in \mathbb{N}$, the probability mass function of $V$ at $\{a_{j_1}, \dots, a_{j_k}\}$ equals*

$$P_Z^{[M]}(V = \{a_{j_1}, \dots, a_{j_k}\}) = \sum_{c \in C_{M,k}} \binom{M}{c_1 \, c_2 \, \dots \, c_k} p_{j_1}^{c_1} \cdots p_{j_k}^{c_k} \tag{24}$$

$$= \sum_{S \subset \{j_1, \dots, j_k\}} (-1)^{k+|S|} \left( \sum_{j \in S} p_j \right)^M . \tag{25}$$

The proof of (24) follows directly from the definitions, while that of (25)—which we will use extensively later—relies on a careful combinatorial argument, pairing sets of odd and even sizes; see Appendix D.5.

To better understand (24), consider the trivial example in which $M = 1$. In this case, $P_Z^{[1]}(V = \{a_j\}) = p_j$ for all $j \in \mathbb{N}$. Thus, $P_Z^{[1]}$, the distribution of uniques when sampling a single element from the distribution $P_Z$, is equal precisely to $P_Z$ itself; i.e., $P_Z^{[1]} = P_Z$. For $M = 2$, we have that $P_Z^{[2]}(V = \{a_j\}) = p_j^2$ for all $j \in \mathbb{N}$; this is the probability of observing $a_i$ twice in a row. Further, for all $j_1, j_2 \in \mathbb{N}$ with $j_1 \neq j_2$, we have that $P_Z^{[2]}(V = \{a_{j_1}, a_{j_2}\}) = 2p_{j_1}p_{j_2}$; this is the probability of observing $(a_{j_1}, a_{j_2})$ or $(a_{j_2}, a_{j_1})$, so that the set of uniques is $\{a_{j_1}, a_{j_2}\}$. One can also verify that (24) leads to the same results. Continuing the above example, for $M = 2$, for all $j_1, j_2 \in \mathbb{N}$ with $j_1 \neq j_2$, (25) leads to $P_Z^{[2]}(V = \{a_{j_1}, a_{j_2}\}) = (p_{j_1} + p_{j_2})^2 - p_{j_1}^2 - p_{j_2}^2 = 2p_{j_1}p_{j_2}$, agreeing with (24).

## Appendix D. Mathematical Proofs

### D.1 Proof of Proposition 1

**Proof** Consider $((X_{\pi(1)}, Y_{\pi(1)}), \dots, (X_{\pi(n)}, Y_{\pi(n)}), (X_{\pi(m+1)}, Y_{\pi(m+1)}))$ for any permutation $\pi$ of $\{1, \dots, n, m+1\}$. This is equal to $((X'_1, Y'_1), \dots, (X'_n, Y'_n), (X'_{m+1}, Y'_{m+1}))$, defined by applying the functions in (7)–(8) to a shuffled data set $Z_{\tilde{\pi}(1)}, \dots, Z_{\tilde{\pi}(m+1)}$, where $\tilde{\pi}$ indicates a permutation of $\{1, \dots, m+1\}$ that agrees with $\pi$ on $\{1, \dots, n, m+1\}$ and leaves $\{n+1, \dots, m\}$ unchanged. Therefore,

$$\begin{aligned}\big((X_{\pi(1)}, Y_{\pi(1)}), \dots, (X_{\pi(n)}, Y_{\pi(n)}), (X_{\pi(m+1)}, Y_{\pi(m+1)})\big) \\ = \big((X'_1, Y'_1), \dots, (X'_n, Y'_n), (X'_{m+1}, Y'_{m+1})\big) \\ \overset{d}{=} ((X_1, Y_1), \dots, (X_n, Y_n), (X_{m+1}, Y_{m+1}))\,,\end{aligned}$$

where the last equality in distribution follows directly from the assumption that $Z_1, \dots, Z_{m+1}$ are exchangeable. ■

### D.2 Proof of Theorem 1

**Proof** We refer to the proof of the more general Theorem 2, of which this result is a special case. In fact, Algorithm 2 corresponds to Algorithm 3 applied with trivial partitions that divide the range of frequencies into a single bin: $L = 1$. Further, the marginal coverage property in (2) is a special case of the frequency-conditional coverage property in (10) with the trivial partitions corresponding to $L = 1$. ■

### D.3 Proof of Theorem 2

The following notation will be helpful: let $B(Y_i) \in \mathcal{B}$ indicate the frequency bin into which $Y_i$ belongs, for $i \in \{1, \dots, n, m+1\}$. We begin by proving the result for the simpler case in which Algorithm 2 is applied using conformity scores that do not require training, in which case $n^{\text{train}} = 0$. For $i \in \{1, \dots, n, m+1\}$, define the random variables $Y_i$ and $X_i$ as in (7)–(8), respectively. We already know from Proposition 1 that $(X_1, Y_1), \dots, (X_n, Y_n), (X_{m+1}, Y_{m+1})$ are exchangeable. This implies that the conformity scores $E(X_i, Y_i)$ are exchangeable with one another, for $i \in \{1, \dots, n, m+1\}$, because each of them only depends on $X_i, Y_i$ and on the separate data points in the sketch $\phi(Z_{n+1}, \dots, Z_m)$. Therefore, $E_{m+1}$ is also exchangeable with the subset of conformity scores with indices in $\{i \in \{1, \dots, n\} : B(Y_i) = B(Y_{m+1})\}$.

Now, fix any bin $B^* \in \mathcal{B}$ and assume $B(Y_{m+1}) = B^*$. Now, note that the interval output by Algorithm 2 does not cover the true frequency $f_m(Z_{m+1})$ if and only if $E_{m+1} > \hat{Q}_{n,1-\alpha} \geq \hat{Q}_{n_l,1-\alpha}(B^*)$. However, a standard exchangeability argument for the conformity scores in $\{i \in \{1, \dots, n\} : B(Y_i) = B^*\}$ shows that $\mathbb{P}[E_{m+1} > \hat{Q}_{n_l,1-\alpha}(B^*) \mid B(Y_{m+1}) = B^*] \leq 1 - \alpha$; for example, see Lemma 1 of Romano et al. (2019). This completes the first part of the proof.

The second part with $n^{\text{train}} > 0$ follows very similarly: Proposition 1 implies that $(X_{n^{\text{train}}+1}, Y_{n^{\text{train}}+1}), \dots, (X_n, Y_n), (X_{m+1}, Y_{m+1})$ are exchangeable, and so must be the conformity scores $E_i$ for $i \in \{n^{\text{train}} + 1, \dots, n, m+1\}$ because each of them only depends on the corresponding $X_i, Y_i$ and on the separate set of observations indexed by $\{1, \dots, n^{\text{train}}\}$, as well as on the sketch $\phi(Z_{n+1}, \dots, Z_m)$. The rest of the proof is exactly the same as in the first part because the empirical quantiles $\hat{Q}_{n_l,1-\alpha}(B)$ are only computed on subsets of the data indexed by $\{n^{\text{train}} + 1, \dots, n\}$.

### D.4 Proof of Theorem 3

Following the same notation as in Algorithm 4, let $Z^*$ indicate a random object sampled uniformly from $\textsc{Unique}(\{Z_{m+1}, \dots, Z_{m+M}\})$. Define also $X^* = (Z^*, \phi(Z_{n+1}, \dots, Z_m))$. By construction, $Z^*$ is exchangeable with all $Z^*_g$ for $g \in [G]$, and $X^*$ is exchangeable with all $X^*_g$ for $g \in [G]$. This implies that the conformity scores $E^*_g = E(X^*_g, Y^*_g)$ are exchangeable with one another, for all $g \in [G]$, as well as with $E^* = E(X^*, Y^*)$. The result is then established with the same argument as in the proof of Theorem 2. The true frequency $f_m(Z^*)$ is not covered by the output confidence interval if and only if $E^* > \hat{Q}_{G,1-\alpha}$, whose probability is bound from above by $1 - \alpha$ according to classical results about tolerance regions (Krishnamoorthy and Mathew, 2009), see also Lemma 1 in Romano et al. (2019).

### D.5 Proof of Proposition A9

**Proof** To prove (24), note that $V = \{a_{j_1}, \dots, a_{j_k}\}$ if and only if there is a $k$-composition $c = (c_1, \dots, c_k)$ of $M$ such that, for all $l \in [k]$, the sequence $(Z_{m+1}, \dots, Z_{m+M}) = (a_{t_1}, \dots, a_{t_M})$ contains exactly $c_l$ values of $a_{j_l}$. For a given $k$-composition $c = (c_1, \dots, c_k)$, there are $\binom{M}{c_1\, c_2\, \dots c_k}$ indices $t_1, t_2, \dots, t_M \in \mathbb{N}$ such that for all $l \in [k]$, exactly $c_l$ of them are equal to $j_l$. The probability that $(Z_{m+1}, \dots, Z_{m+M})$ equals any one of them is $p_{j_1}^{c_1} \cdots p_{j_k}^{c_k}$, showing (24).

To prove (25), note that, for any $S \subset \mathbb{N}$, any product arising from the expansion of $\left(\sum_{l \in S} p_l\right)^M$ has at least one and at most $M$ distinct indices $l$. Collecting the products $p_{i_1} p_{i_2} \dots p_{i_M}$ by the number $d \in \{1, \dots, M\}$ of distinct indices among their factors, we find

$$\left(\sum_{l \in S} p_l\right)^M = \sum_{d=1}^{M} \sum_{\{l_1,\dots,l_d\} \subset S,\ l_i \neq l_j \text{ for } i \neq j} \sum_{c \in C_{M,d}} \binom{M}{c_1\, c_2\, \dots\, c_d} p_{l_1}^{c_1} \cdots p_{l_d}^{c_d}.$$

Now fix any $\{l_1, \dots, l_d\} \subset \{j_1, \dots, j_k\}$, and any $c \in C_{M,d}$. Using the previous formula for each $S$ on the right hand side of (25), the total coefficient of $p_{l_1}^{c_1} \cdots p_{l_d}^{c_d}$ is the following sum over subsets $S$

$$\binom{M}{c_1\, c_2\, \dots\, c_d} \sum_{S \subset \{j_1, \dots, j_k\}} (-1)^{k+|S|} I\left(\{l_1, \dots, l_d\} \subset S\right).$$

Writing the indicator $I\left(\{l_1, \dots, l_d\} \subset S\right)$ inside the summation constraint, and factoring out $(-1)^k$, this equals

$$(-1)^k \binom{M}{c_1\, c_2\, \dots\, c_d} \sum_{\{l_1, \dots, l_d\} \subset S \subset \{j_1, \dots, j_k\}} (-1)^{|S|}.$$

Now, if $\{l_1, \dots, l_d\} = \{j_1, \dots, j_k\}$, the above summation (after the pre-factor) has only one term—$S = \{j_1, \dots, j_k\}$—and equals $(-1)^{|S|} = (-1)^k$.

Otherwise, the above summation contains $2^{k-d} > 1$ terms. We now construct a pairing of the sets $S$ that index of the summation, such that each pair $(S_1, S_2)$ contains an odd and even sized set. There must be an index $j_a$, $a \in [k]$, such that $j_a \notin \{l_1, \ldots, l_d\}$. Suppose without loss of generality that we have $j_k \notin \{l_1, \ldots, l_d\}$ (otherwise rename the indices $j_a$ and $j_k$).

Then, for any set $S_1$ such that $\{l_1, \ldots, l_d\} \subset S_1 \subset \{j_1, \ldots, j_k\}$ that does not contain $j_k$, there is a corresponding set $S_2 = S_1 \cup \{j_k\}$ such that $\{l_1, \ldots, l_d\} \subset S' \subset \{j_1, \ldots, j_k\}$. Moreover, all sets $S$ such that $\{l_1, \ldots, l_d\} \subset S \subset \{j_1, \ldots, j_k\}$ fall into exactly one such pair. Further, in each pair, there is one set of an odd size and one set of an even size.

Thus, in each pair, we have

$$(-1)^{|S_1|} + (-1)^{|S_2|} = 0,$$

Therefore, when $\{l_1, \ldots, l_d\} \neq \{j_1, \ldots, j_k\}$

$$\sum_{\{j_1,\ldots,j_d\}\subset S\subset\{i_1,\ldots,i_k\}} (-1)^{|S|} = 0.$$

Hence, the coefficient of $p_{j_1}^{c_1} \cdots p_{l_d}^{c_d}$ in the expression on the right hand side of (25) is nonzero only when $\{l_1, \ldots, l_d\} = \{j_1, \ldots, j_k\}$, in which case it equals

$$(-1)^{2k}\binom{M}{c_1\, c_2\, \ldots\, c_d} = \binom{M}{c_1\, c_2\, \ldots\, c_d}.$$

This shows that (24) and (25) coincide, completing the proof.

■

### D.6 Proof of Proposition 4

**Proof** The formula in (13) follows directly from Equation (24) in Proposition A9, because for a set $V$ of size $k$, the probability of any element being the selected unique $\zeta$ equals $1/k$. Next, $U_Z^{[1]} = P_Z$ by definition. In addition,

$$\begin{aligned} U_Z^{[2]}(\zeta = a_{j_1}) &= p_{j_1}^2 + \frac{1}{2} \sum_{J=\{j_1,j_2\}\subset\mathbb{N}^2, |J|=2} \; \sum_{c\in C_{2,2}} \binom{2}{c_1\, c_2} p_{j_1}^{c_1} p_{j_2}^{c_2} \\ &= p_{j_1}^2 + \frac{1}{2} \sum_{j_2\in\mathbb{N}\setminus\{j_1\}} 2p_{j_1}p_{j_2} = p_{j_1}^2 + p_{j_1}(1-p_{j_1}) = p_{j_1}. \end{aligned}$$

This shows that $U_Z^{[2]} = P_Z$. Finally,

$$\begin{aligned} &U_Z^{[3]}(\zeta = a_{j_1}) \\ &= p_{j_1}^3 + \frac{1}{2} \sum_{J=\{j_1,j_2\}\subset\mathbb{N}^2, |J|=2} \; \sum_{c\in C_{3,2}} \binom{3}{c_1\, c_2} p_{j_1}^{c_1} p_{j_2}^{c_2} + \frac{1}{3}\binom{3}{1\,1\,1} \sum_{J=\{j_1,j_2,j_3\}\subset\mathbb{N}^3, |J|=3} p_{j_1}p_{j_2}p_{j_3} \\ &= p_{j_1}^3 + \frac{1}{2} \sum_{J=\{j_1,j_2\}\subset\mathbb{N}^2, |J|=2} \; \sum_{c\in C_{3,2}} \binom{3}{c_1\, c_2} p_{j_1}^{c_1} p_{j_2}^{c_2} + 2 \sum_{J=\{j_1,j_2,j_3\}\subset\mathbb{N}^3, |J|=3} p_{j_1}p_{j_2}p_{j_3}. \end{aligned}$$

This further equals

$$\begin{aligned}
&p_{j_1}^3 + \frac{3}{2} \sum_{j_2 \in \mathbb{N} \setminus \{j_1\}} (p_{j_1}^2 p_{j_2} + p_{j_1} p_{j_2}^2) + 2 p_{j_1} \sum_{\{j_2, j_3\} \subset (\mathbb{N} \setminus \{j_1\})^2, \, |J|=2} p_{j_2} p_{j_3} \\
&= p_{j_1}^3 + \frac{3}{2} p_{j_1}^2 (1 - p_{j_1}) + p_{j_1} \left( \frac{3}{2} \sum_{j_2 \in \mathbb{N} \setminus \{j_1\}} p_{j_2}^2 + 2 \sum_{\{j_2, j_3\} \subset (\mathbb{N} \setminus \{j_1\})^2, \, |J|=2} p_{j_2} p_{j_3} \right).
\end{aligned}$$

By expanding the square in $(1 - p_{j_1})^2 = (\sum_{j_2 \in \mathbb{N} \setminus \{j_1\}} p_{j_2})^2$, this further equals

$$\begin{aligned}
&\frac{p_{j_1}^2 (3 - p_{j_1})}{2} + p_{j_1} \left( \frac{3}{2} \left[ (1 - p_{j_1})^2 - \sum_{\{j_2, j_3\} \subset (\mathbb{N} \setminus \{j_1\})^2, \, |J|=2} p_{j_2} p_{j_3} \right] \right. \\
&\qquad \left. + 2 \sum_{\{j_2, j_3\} \subset (\mathbb{N} \setminus \{j_1\})^2, \, |J|=2} p_{j_2} p_{j_3} \right) \\
&= \frac{p_{j_1} (2 p_{j_1}^2 - 3 p_{j_1} + 3)}{2} + \frac{p_{j_1}}{2} \sum_{\{j_2, j_3\} \subset (\mathbb{N} \setminus \{j_1\})^2, \, |J|=2} p_{j_2} p_{j_3}.
\end{aligned}$$

This finishes the proof. ■

### D.7 Proof of Proposition 5

**Proof** Let the two objects be denoted by $a_1$ and $a_2$. Then, one can verify using (25) in Proposition A9 and (13) in Proposition 4 that, for $j = 1, 2$,

$$U_Z^{[M]}(\zeta = a_j) = \frac{1 + p_j^M - (1 - p_j)^M}{2}. \tag{26}$$

Therefore,

$$\Delta(M, M'; 2) = \frac{1}{2} \sup_{p \in [0,1]} \left| p^M - (1-p)^M - \left[ p^{M'} - (1-p)^{M'} \right] \right|.$$

Let $\delta = (1 - p)/p \geq 0$, so that $p = 1/(1 + \delta)$, and suppose without loss of generality that $\delta \leq 1$; otherwise, change variables to $1 - p \leftarrow p$. Then, the term inside the absolute value above can be written as

$$A(\delta) = \frac{1 - \delta^{M'}}{(1 + \delta)^{M'}} - \frac{1 - \delta^M}{(1 + \delta)^M} \geq 0. \tag{27}$$

Now, denoting, for $c \geq 1$, $g(\delta, c) = \frac{1 - \delta^c}{(1+\delta)^c}$, we have

$$\begin{aligned}
\frac{\partial g(\delta, c)}{\partial \delta} &= \frac{-c \delta^{c-1} (1 + \delta)^c - (1 - \delta^c) \cdot c (1 + \delta)^{c-1}}{(1 + \delta)^{2c}} \\
&= -c \frac{\delta^{c-1} (1 + \delta) + (1 - \delta^c)}{(1 + \delta)^{c+1}} = -c \frac{1 + \delta^{c-1}}{(1 + \delta)^{c+1}}.
\end{aligned}$$

Hence,

$$A'(\delta) = -M' \frac{1+\delta^{M'-1}}{(1+\delta)^{M'+1}} + M \frac{1+\delta^{M-1}}{(1+\delta)^{M+1}}.$$

Thus, $A'(\delta) \geq 0$ is equivalent to

$$\frac{1+\delta^{M-1}}{1+\delta^{M'-1}} \geq \frac{M'}{M}(1+\delta)^{M-M'},$$

or, with the function $h$ defined as in (16), to $h(\delta) \geq \ln(M'/M)$. Now,

$$h'(\delta) = \frac{(M-1)\delta^{M-2}}{1+\delta^{M-1}} - \frac{(M'-1)\delta^{M'-2}}{1+\delta^{M'-1}} - \frac{M-M'}{1+\delta}.$$

We claim that $h'(\delta) < 0$ for all $\delta \in [0,1)$. Indeed, this is equivalent to the function

$$\tilde{\psi}(M) = \frac{M}{1+\delta} - \frac{(M-1)\delta^{M-2}}{1+\delta^{M-1}}$$

being increasing in $M$, for all $M \geq 2$. Denote $x = M-1 \geq 1$, $\psi(x) = \delta \cdot \tilde{\psi}(x+1)$, and $a = 1/\delta \geq 1$. Then,

$$\psi(x) = \frac{x+1}{1+a} - \frac{x}{1+a^x}$$

and

$$\psi'(x) = \frac{1}{1+a} - \frac{1+a^x - xa^x \ln a}{(1+a^x)^2}.$$

Hence, $\psi'(x) > 0$ is equivalent to

$$(1+a)(1+a^x - xa^x \ln a) < (1+a^x)^2.$$

Now, since $a \geq 1$ and $x \geq 1$, we have $1+a \leq 1+a^x$ and $xa^x \ln a \geq 0$. Equality happens in both equations if and only if $x=1$ and $a=1$. This corresponds to $M=2$ and $\delta=1$. Thus, the above inequality holds for all $\delta \in [0,1)$. This shows that $h$ is decreasing for $\delta \in [0,1)$. Since $h(0)=0$ and $h(1) = M'-M \leq \ln(M'/M)$, and as $h$ is continuous on $[0,1]$, there is a unique solution $\delta_* \in [0,1]$ to $h(\delta_*) = \ln(M'/M)$. This proves the first claim. Based on our analysis, it follows that $A$ is maximized over $[0,1]$ at $\delta_*$. This finishes the proof. ■

### D.8 Proof of Corollary 6

**Proof** Recalling the form of the function $h$ from (16), the equation for $\delta \in [0,1]$ from Proposition 5 is

$$\frac{M}{M'}(\delta^{M-1}+1) = (1+\delta^{M'-1})(1+\delta)^{M-M'}.$$

For $M = aM'$, this becomes

$$a(\delta^{aM'-1}+1) = (1+\delta^{M'-1})(1+\delta)^{(a-1)M'}. \tag{28}$$

Now, as $a \geq 1$ and $\delta \leq 1$, we have $2 \geq (1+\delta)^{(a-1)M'}$, and thus using the inequality $2 \leq (1+1/x)^x$ for $x \geq 1$ with $x = (a-1)M'$, we find that[4] as soon as $M' \geq 1/(a-1)$,

$$1 + \delta \leq 2^{1/[(a-1)M']} \leq 1 + \frac{1}{(a-1)M'}.$$

Hence, $\delta \leq \frac{1}{(a-1)M'}$; and for $M' \geq 2/(a-1)$, we thus find $\delta \leq 1/2$. Using this in (28), we obtain

$$\frac{a}{1+2^{-(M'-1)}} \leq (1+\delta)^{(a-1)M'} \leq a\left(1+2^{-(aM'-1)}\right).$$

Therefore,

$$|(1+\delta)^{(a-1)M'} - a| \leq a \max\left\{2^{-(aM'-1)}, \frac{2^{-(M'-1)}}{1+2^{(M'-1)}}\right\} \leq 2^{1-M'}a.$$

Hence, using that $x \mapsto x^{-a/(a-1)}$ is decreasing on $(0,\infty)$, as well as the inequality $1 > (1-1/x)^c \geq 1 - c/x$ for all $x, c \geq 1$, applied to $x = 2^{M'-1}$ and $c = a/(a-1)$,

$$\begin{aligned}
&\left|\frac{1}{(1+\delta)^{aM'}} - a^{-a/(a-1)}\right| \leq \left|\left[a(1-2^{1-M'})\right]^{-a/(a-1)} - a^{-a/(a-1)}\right| \\
&= a^{-a/(a-1)}\left[\left(1-2^{1-M'}\right)^{-a/(a-1)} - 1\right] \\
&\leq a^{-a/(a-1)}\left[\left(1-2^{1-M'}a/(a-1)\right)^{-1} - 1\right] \leq 2a^{-a/(a-1)}2^{1-M'}a/(a-1),
\end{aligned}$$

as long as $2^{1-M'}a/(a-1) \leq 1/2$, i.e., $M' \geq 2 + \log_2(a/(a-1))$. Similarly,

$$\left|\frac{1}{(1+\delta)^{M'}} - a^{-1/(a-1)}\right| \leq 2a^{-1/(a-1)}2^{1-M'}/(a-1),$$

as long as $2^{1-M'}/(a-1) \leq 1/2$, i.e., $M' \geq 2 + \log_2(1/(a-1))$. Thus, with $A$ from (27), using also that $\delta \leq 1/[(a-1)M']$,

$$\begin{aligned}
&|A(\delta) - a^{-1/(a-1)}(1-1/a)| = \left|\frac{1-\delta^{M'}}{(1+\delta)^{M'}} - \frac{1-\delta^{aM'}}{(1+\delta)^{aM'}} - \left(a^{-1/(a-1)} - a^{-a/(a-1)}\right)\right| \\
&\leq \left|\frac{1}{(1+\delta)^{M'}} - a^{-1/(a-1)}\right| + \left|\frac{1}{(1+\delta)^{aM'}} - a^{-a/(a-1)}\right| + \frac{\delta^{M'}}{(1+\delta)^{M'}} + \frac{\delta^{aM'}}{(1+\delta)^{aM'}} \\
&\leq 2^{2-M'}\left[a^{-a/(a-1)}a/(a-1) + a^{-1/(a-1)}/(a-1)\right] + 2[(a-1)M']^{-M'} \\
&\leq 2^{3-M'}a^{-1/(a-1)}/(a-1) + 2[(a-1)M']^{-M'}.
\end{aligned}$$

Finally, denoting by $\delta_*$ the unique solution of (28), $\Delta(M, M'; 2) = A(\delta_*)$ from the proof of Proposition 5, completing this proof. ∎

[4] All inequalities in this argument will hold for $M'$ sufficiently large, and having determined the required range, we will not repeatedly specify it.

### D.9 Proof of Theorem 7

**Proof** This follows immediately by combining Corollary 6 with (15) and Theorem 3. ■

### D.10 Proof of Theorem 8

**Proof** First, we aim to show that, for $p_j \neq p'_j$,

$$|U_Z^{[M]}(\{a_j\}) - U_{Z'}^{[M]}(\{a_j\})| < |p_j - p'_j|. \tag{29}$$

For simplicity of notation, define $p := p_i$ and $q := p'_i$. Define also $d : [0,1] \to \mathbb{R}$ as $d(p) = [p^M - (1-p)^M]/2$, for all $p \in [0,1]$. It follows from (26) that we need to show

$$|d(p) - d(q)| < |p - q|.$$

Suppose without loss of generality that $p < q$. By the mean value theorem applied to $d$, there exists a $\omega \in [p,q]$, such that $d(p) - d(q) = d'(\omega)(p-q)$. Therefore, it suffices to show that $|d'(\omega)| < 1$ for $\omega \in (c, 1-c)$. Now, $d'(\omega) = M[\omega^{M-1} + (1-\omega)^{M-1}]/2$. We note here that $d'(0) = M/2 > 1$, $d'(1/2) = M/2^{M-1} < 1$ (as $M \geq 3$), and $d'$ is strictly decreasing as a function of $\omega$ for $\omega \in [0, 1/2]$. Therefore, the equation $d'(c) = 1$ has a unique solution over $c \in [0, 1/2)$. This shows that $c$ in (18) is well defined.

Moreover, because $d'$ is strictly decreasing between $[0, 1/2]$, it follows that $d'$ is maximized within the interval $[c, 1-c]$ at $c$ and (by symmetry) at $1-c$. Therefore, $|d'(\omega)| < 1$ for $\omega \in (c, 1-c)$, and (29) follows.

Let $TV(\cdot,\cdot)$ be the total variation distance. Then, for all $P_Z, P_{Z'} \in S_c$, with $P_Z \neq P_{Z'}$,

$$TV(U_Z^{[M]}, U_{Z'}^{[M]}) < TV(P_Z, P_{Z'}). \tag{30}$$

Following (15), define $e_U(Z^*) = \mathbb{P}_{\tilde{Z}_{1:G} \sim \left(U_Z^{[M]}\right)^{|G|}}[\mathcal{E}]$ and $e(Z^*) = \mathbb{P}_{\tilde{Z}_{1:G} \sim P_Z^{|G|}}[\mathcal{E}]$. Let $A_U, A$ be the sets of functions over which $e_U, e$ can range, respectively. We need to show that

$$\sup_{e_U \in A_U} \left| \mathbb{E}_{Z^* \sim U_Z^{[M]}} e_U(Z^*) - \mathbb{E}_{Z^* \sim U_{Z'}^{[M]}} e_U(Z^*) \right| < \sup_{e \in A} \left| \mathbb{E}_{Z^* \sim P_Z} e(Z^*) - \mathbb{E}_{Z^* \sim P_{Z'}} e(Z^*) \right|.$$

Because $\mathcal{E} \subset \{a_1, a_2\}^{|G|+1}$ is arbitrary, the possible values of $e_U$ include zero and unity, for any value $Z^* = z$. Hence, the above inequality is equivalent to (30), completing the proof. ■

## Appendix E. Supplementary Figures and Tables

### E.1 Theoretical Analysis of Robustness to Sample Inflation

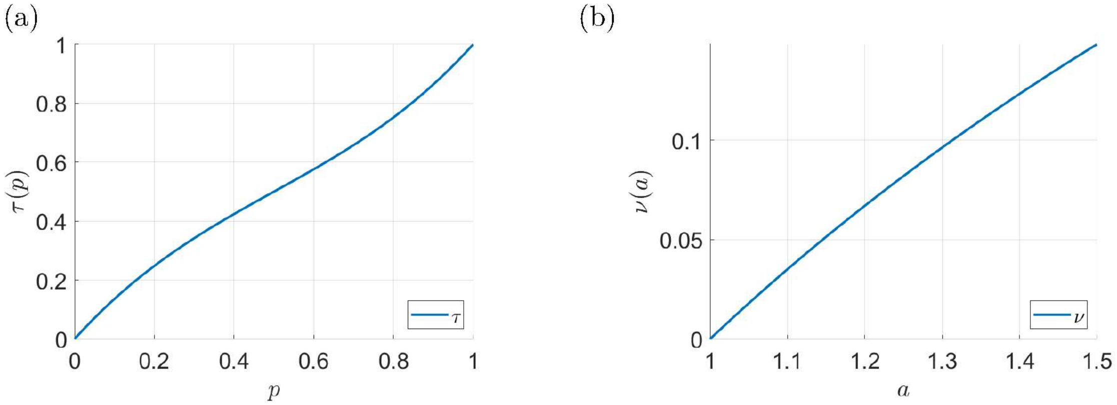

Figure A10: (a) Plot of the function $\tau$ defined in (14). (b) Plot of the function $a \mapsto \nu(a) := a^{-1/(a-1)}(1 - 1/a)$ defined in Corollary 6.

### E.2 Experiments with Synthetic Zipf Data

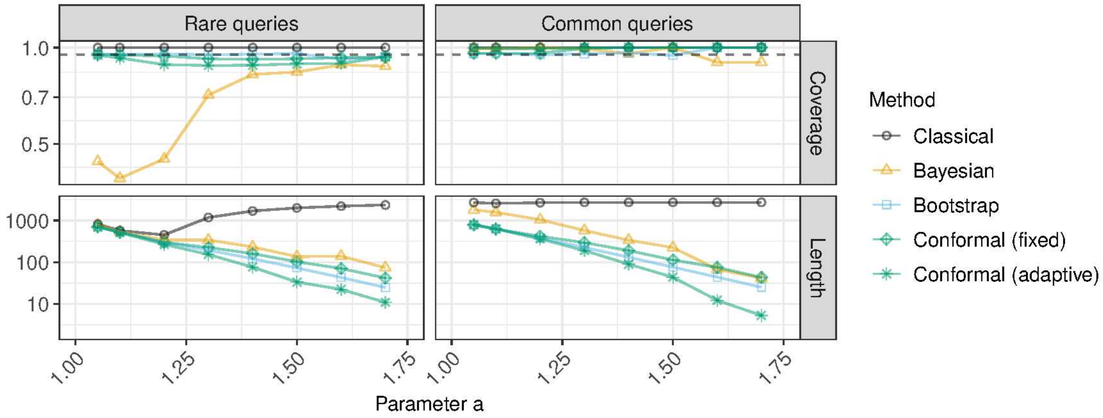

Figure A11: Performance of confidence intervals stratified by the true query frequency. Left: frequency below median; right: above median. Here, the conformal confidence intervals are constructed using Algorithm 2, which seeks marginal coverage (2), instead of Algorithm 3, which seeks frequency-range conditional coverage (10). Other details are as in Figure 3.

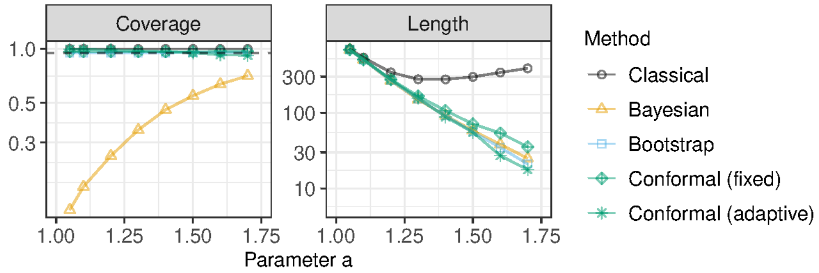

Figure A12: Performance of confidence intervals for random queries on synthetic Zipf data, keeping only distinct queries. The coverage is the empirical proportion of distinct queries whose frequency is covered by the output confidence intervals. The conformal confidence intervals are computed by applying Algorithm 3 with $L = 5$ frequency bins. Other details are as in Figure 3.

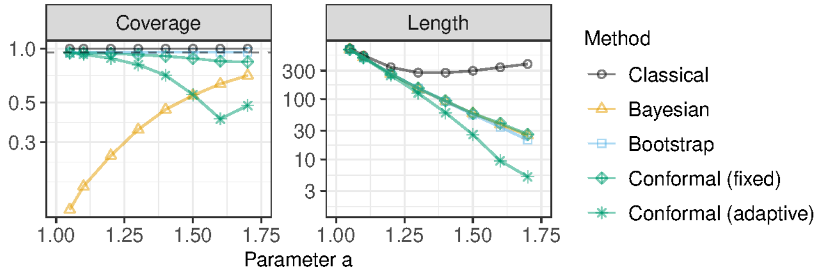

Figure A13: Performance of confidence intervals for random queries on synthetic Zipf data, keeping only distinct queries. The conformal confidence intervals are computed by applying Algorithm 2, seeking marginal coverage (2), instead of Algorithm 3. Other details are as in Figure A12.

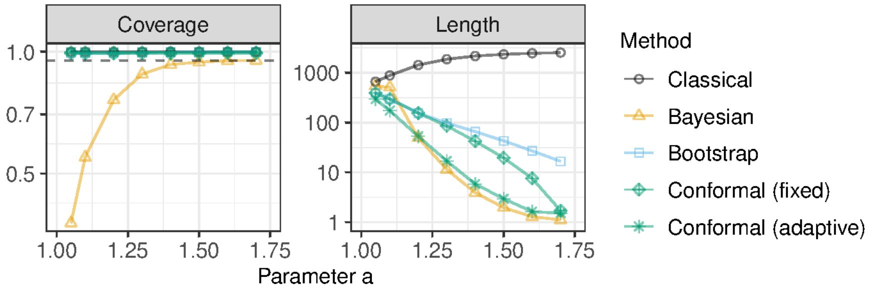

Figure A14: Performance of 95% confidence intervals with simulated Zipf data sketched with the CMS. The results are shown as a function of the Zipf tail parameter $a$. The data are sketched with the CMS-CU instead of the CMS. Other details are as in Figure 3.

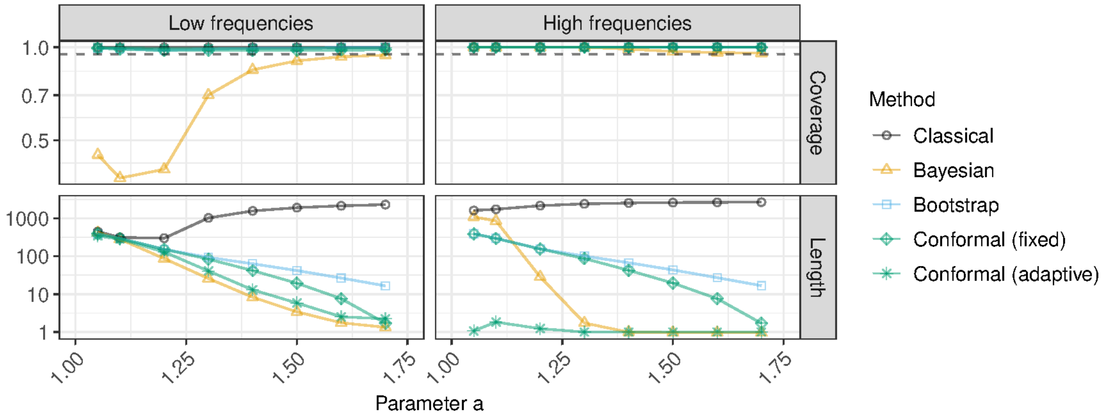

Figure A15: Performance of confidence intervals stratified by the true query frequency. Left: frequency below median; right: above median. The data are sketched with the CMS-CU instead of the CMS. Other details are as in Figure 4.

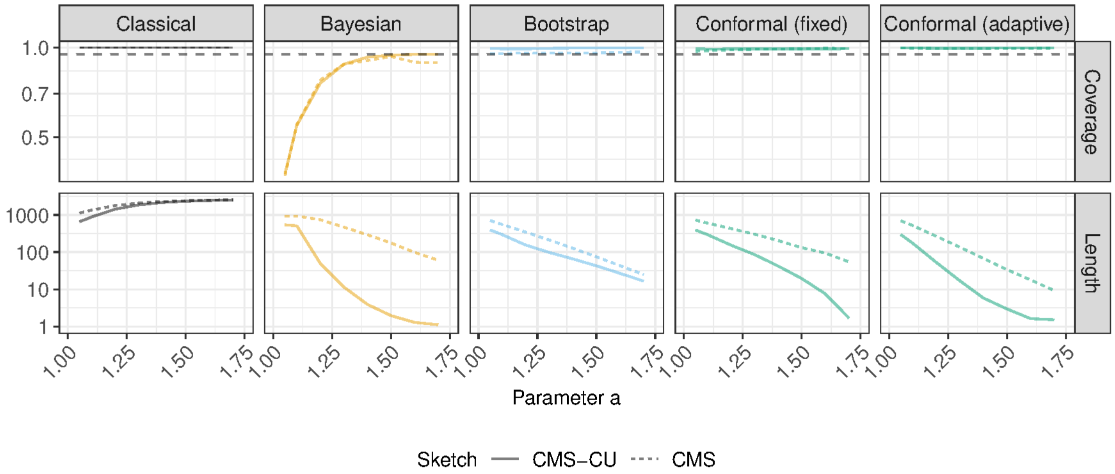

Figure A16: Performance of 95% confidence intervals for random queries, based on synthetic data from a Zipf distribution. The data are sketched with either the vanilla CMS or the CMS-CU. The results are shown as a function of the Zipf tail parameter $a$. Other details are as in Figure 3.

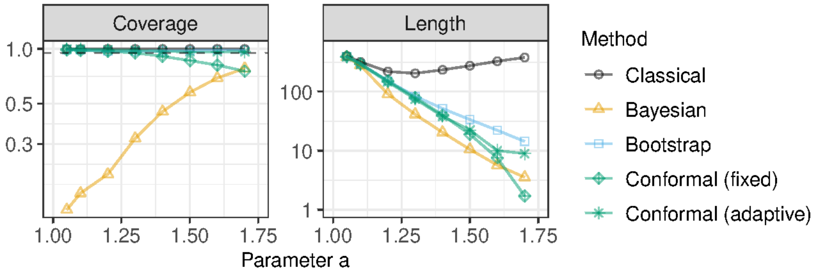

Figure A17: Performance of confidence intervals for random queries on synthetic Zipf data, keeping only distinct queries. The coverage is the empirical proportion of distinct queries whose frequency is covered by the output confidence intervals. The conformal confidence intervals are computed by applying Algorithm 3 with $L = 5$ frequency bins. The data are sketched with the CMS-CU instead of the CMS. Other details are as in Figure A12.

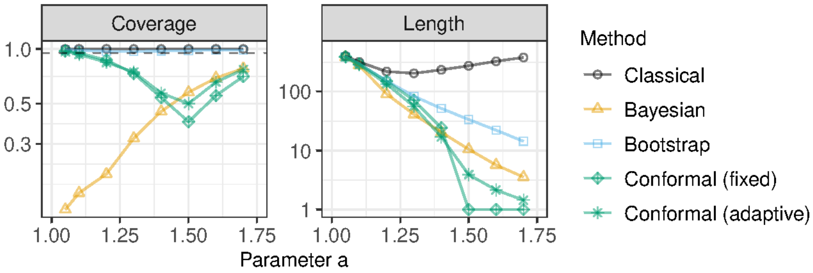

Figure A18: Performance of confidence intervals for random queries on synthetic Zipf data, keeping only distinct queries, after sketching the CMS-CU. The conformal confidence intervals are computed by applying Algorithm 2, seeking marginal coverage (2), instead of Algorithm 3. Other details are as in Figure A17.

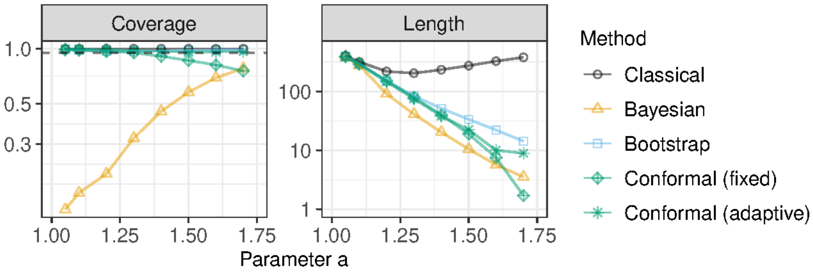

Figure A19: Performance of confidence intervals for random queries on synthetic Zipf data, keeping only distinct queries. The coverage is the empirical proportion of distinct queries whose frequency is covered by the output confidence intervals. The conformal confidence intervals are computed by applying Algorithm 3 with $L = 5$ frequency bins. Other details are as in Figure A14.

### E.3 Non-Random Sketching with Data-driven Hash Functions

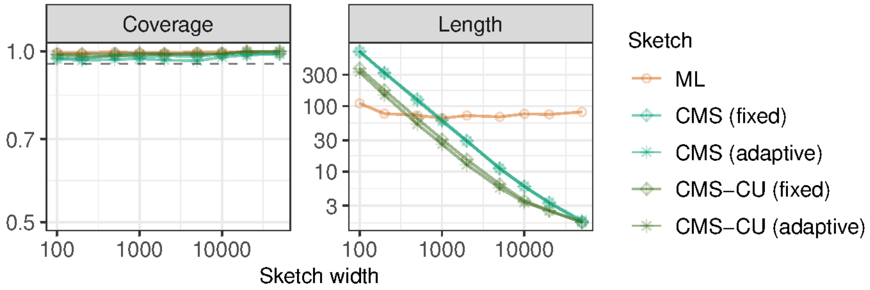

Figure A20: Performance on simulated Zipf data of conformal confidence intervals based on different data sketches, as a function of the sketch width. The conformity scores are evaluated separately within $L = 5$ frequency bins, seeking frequency-range conditional coverage (10). Other details are as in Figure 7.

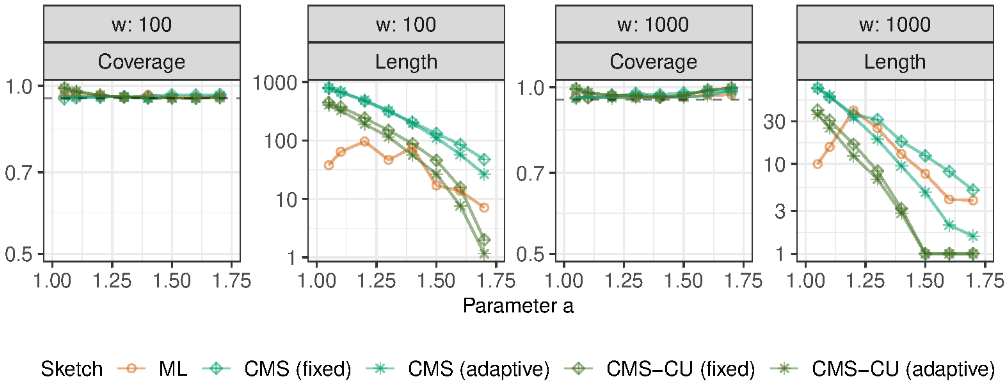

Figure A21: Performance of 95% confidence intervals based on different data sketches, either with width $w = 100$ or $w = 1000$. The results are shown as a function of the tail parameter of the Zipf distribution from which the data are sampled. Other details are as in Figure 7.

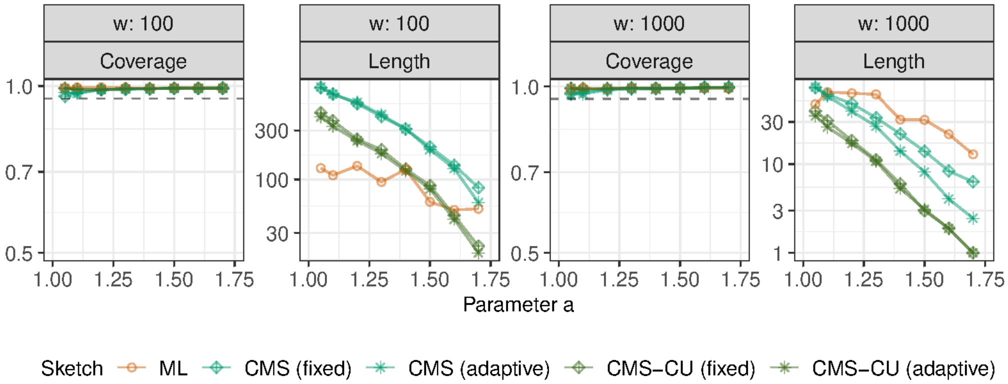

Figure A22: Performance of 95% confidence intervals based on different data sketches, either with width $w = 100$ or $w = 1000$. The results are shown as a function of the tail parameter of the Zipf distribution from which the data are sampled. The conformity scores are evaluated separately within $L = 5$ frequency bins, seeking frequency-range conditional coverage (10). Other details are as in Figure A21.

### E.4 Experiments with Synthetic Pitman-Yor Prior Data

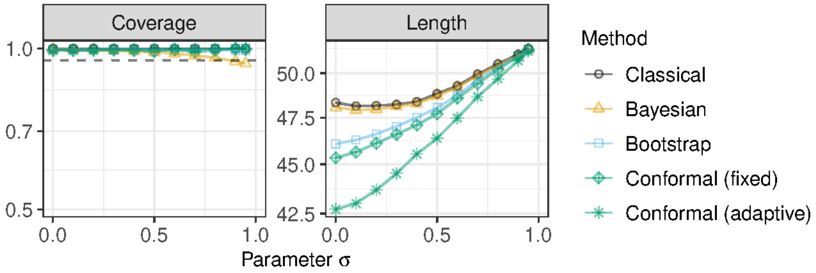

Figure A23: Empirical coverage and length of 95% confidence intervals for random queries on synthetic data from the predictive distribution of a Pitman-Yor process. The data are sketched with the CMS-CU. The results are shown as a function of the Pitman-Yor process parameter $\sigma$. Other details are as in Figure 3.

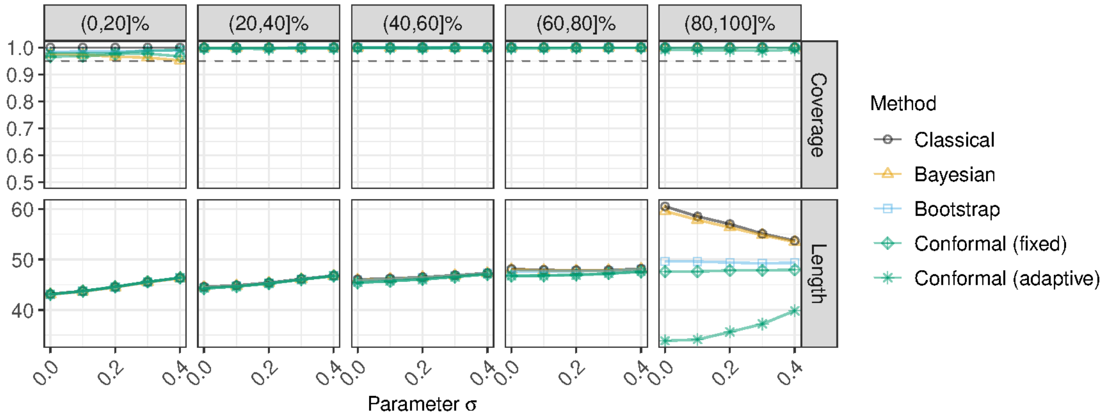

Figure A24: Performance of confidence intervals for random queries on synthetic data from the predictive distribution of a Pitman-Yor process. The results are stratified by the quintile of the true query frequency. Other details are as in Figure A23.

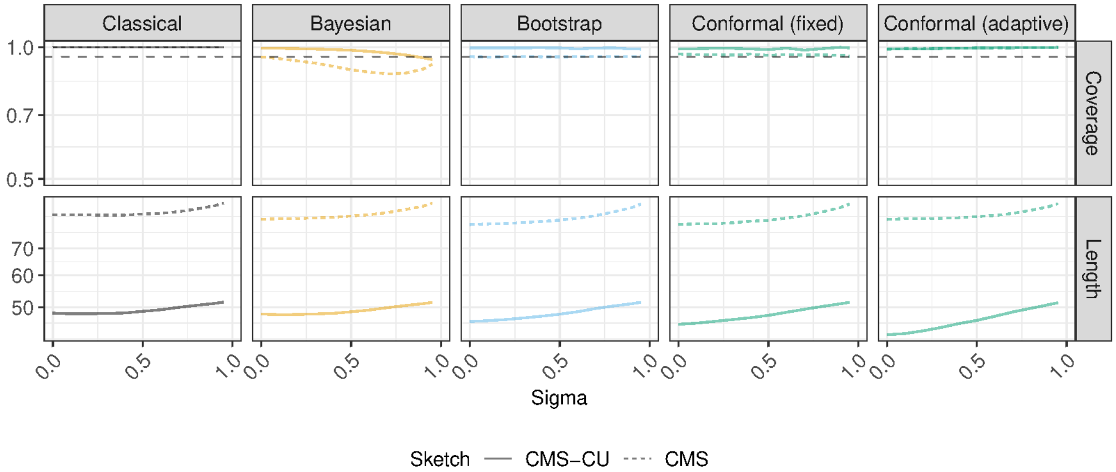

Figure A25: Performance of 95% confidence intervals for random queries, based on synthetic data from the predictive distribution of a Pitman-Yor process and sketched with either the vanilla CMS or the CMS-CU. The results are shown as a function of the Pitman-Yor process parameter $\sigma$. Other details are as in Figure A23.

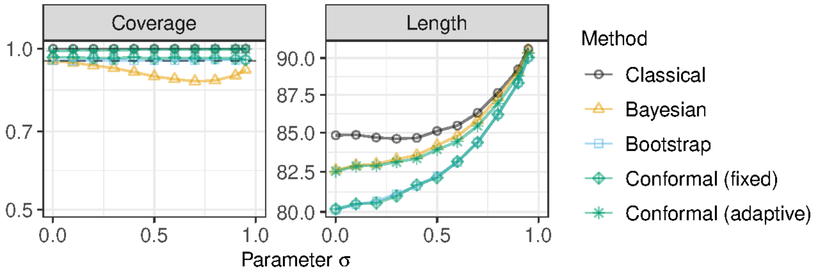

Figure A26: Performance of 95% confidence intervals for random queries, based on synthetic data from the predictive distribution of a Pitman-Yor process and sketched with the vanilla CMS. The results are shown as a function of the Pitman-Yor process parameter $\sigma$. Other details are as in Figure A23.

### E.5 Experiments with Heavy-Hitter Synthetic Data

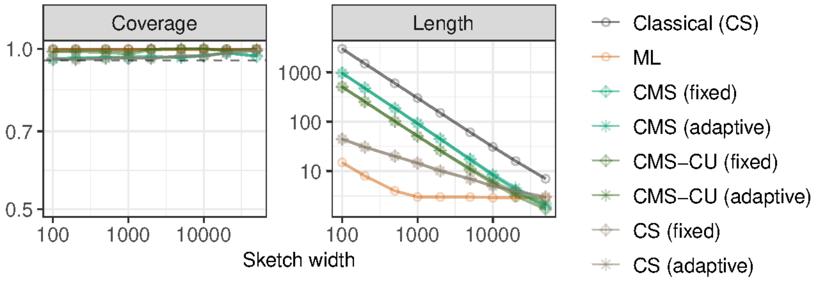

Figure A27: Performance of 95% confidence intervals for random queries, based on synthetic data including heavy hitters. The data are generated according to the following probability distribution: $Z = 0$ with probability $1/\sqrt{m}$, where $m = 100,000$, and $Z \sim \text{Unif}(0, 1)$ otherwise. The results are shown as a function of the hash width. Other details are as in Figure 3. Note that the classical worst-case error bound for the CS is similar to that for the CMS, as described in Appendix A.2. Specifically, this bound is derived by combing Markov's inequality with a Chernoff bound argument; e.g., see Cormode and Yi (2020) for further details.

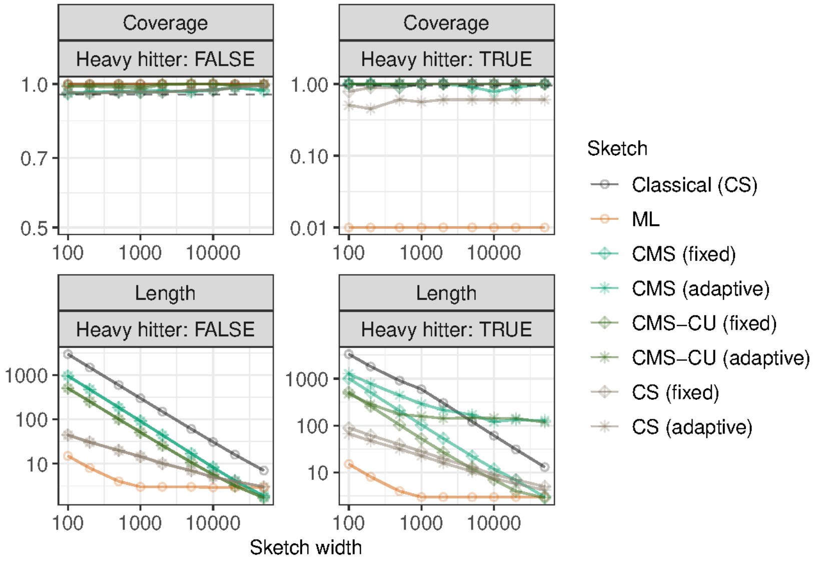

Figure A28: Performance of 95% confidence intervals for random queries, based on synthetic data with heavy hitters. The coverage and average width of the conformal confidence intervals is reported separately for heavy hitters and all other objects. Other details are as in Figure A27. These results show that the CS sketch applied in combination with our method leads to the most informative confidence intervals—it achieves shorter width while ensuring valid coverage conditional on whether the queried object is a heavy hitter. Note that the true conditional coverage obtained with ML sketch for heavy hitter queries is zero, even though a truncated value of 0.01 is shown on the logarithmic scale for convenience.

### E.6 Experiments with Two-Sided Confidence Intervals

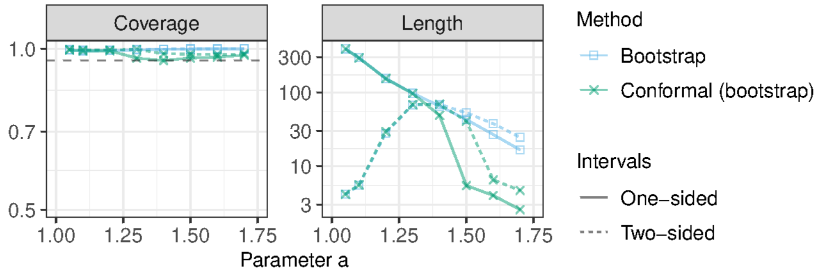

Figure A29: Performance of 95% one-sided and two-sided confidence intervals with data from a Zipf distribution, sketched with the CMS-CU. The results are shown as a function of the Zipf tail parameter $a$. Standard errors would be too mall to be clearly visible in this figure, and are hence omitted. The two dashed curves for the two-sided intervals are nearly indistinguishable from one another for $a < 1.3$. Other details are as in Figure 3.

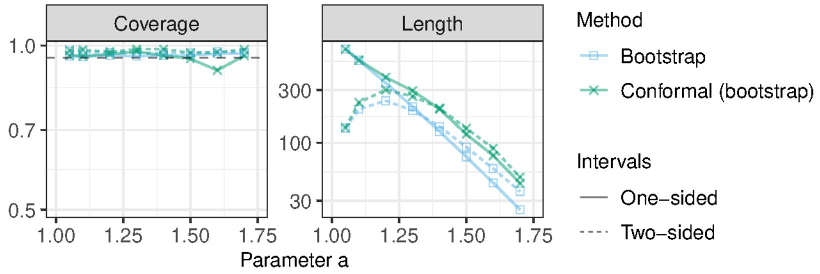

Figure A30: Performance of 95% one-sided and two-sided confidence intervals with data from a Zipf distribution, sketched with the vanilla CMS. The results are shown as a function of the Zipf tail parameter $a$. The two dashed curves for the two-sided intervals are nearly indistinguishable from one another for $a < 1.1$. Other details are as in Figure A29.

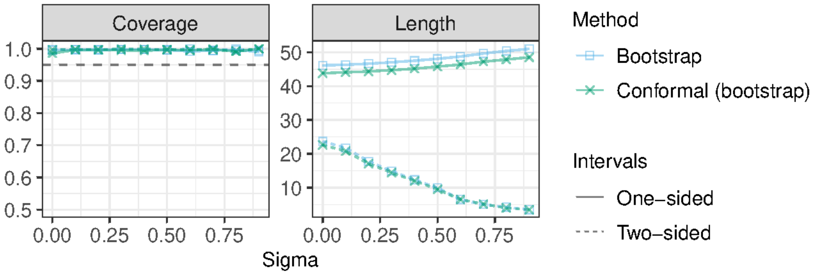

Figure A31: Performance of 95% one-sided and two-sided confidence intervals with data set sampled from the predictive distribution of a Pitman-Yor process and sketched with the CMS-CU. The results are shown as a function of the Pitman-Yor process parameter $\sigma$. The two dashed curves for the two-sided intervals are nearly indistinguishable from one another. Other details are as in Figure A23.

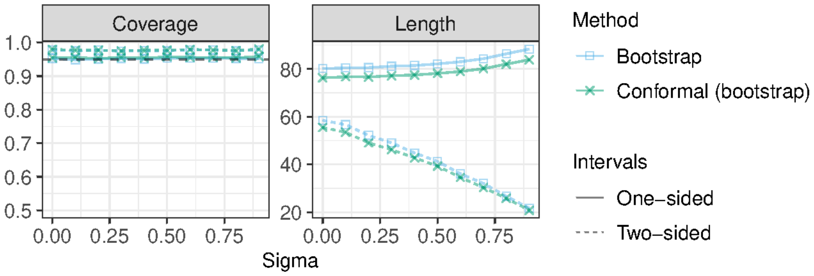

Figure A32: Performance of 95% one-sided and two-sided confidence intervals with data set sampled from the predictive distribution of a Pitman-Yor process and sketched with the vanilla CMS. The results are shown as a function of the Pitman-Yor process parameter $\sigma$. Other details are as in Figure A31.

### E.7 Illustration on SARS-CoV-2 DNA Data

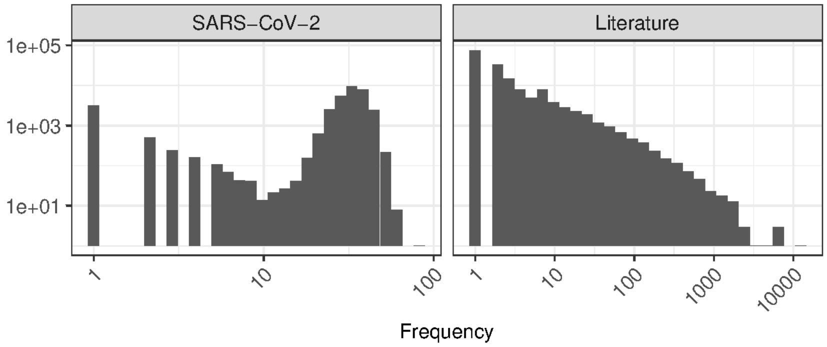

Figure A33: True frequency distribution of unique objects in two empirical data sets. Left: sequenced SARS-CoV-2 DNA 16-mers. Right: English 2-grams in a corpus of classic English literature.

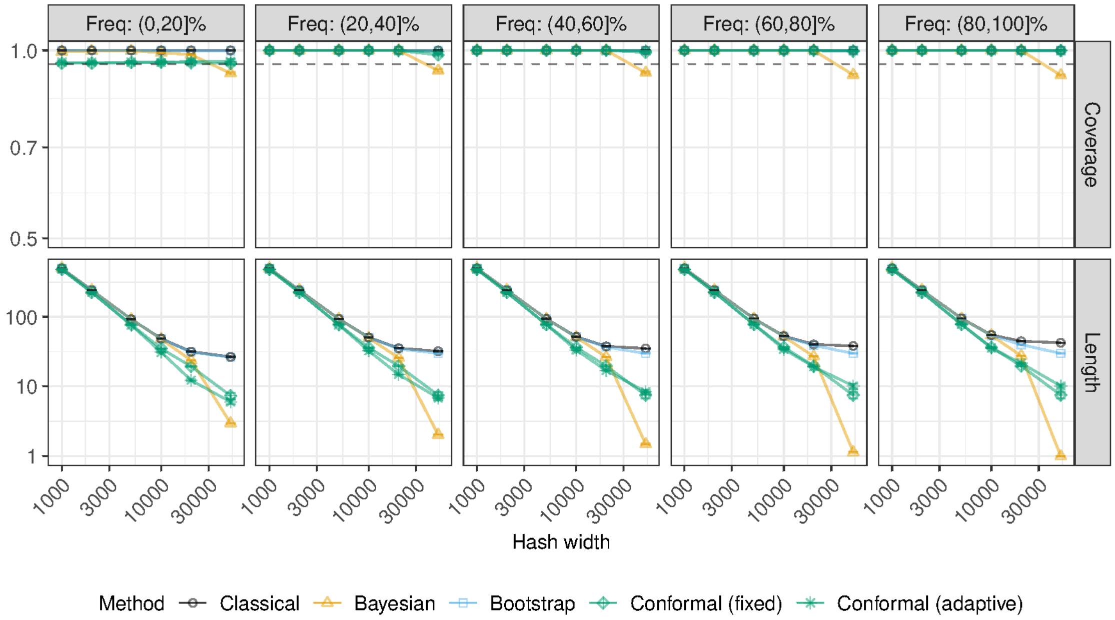

Figure A34: Performance of 95% confidence intervals for random queries on SARS-CoV-2 sequence data sketched with the CMS-CU. The results are shown as a function of the hash width and stratified by the quintile of the true query frequency. Other details are as in Figure 8.

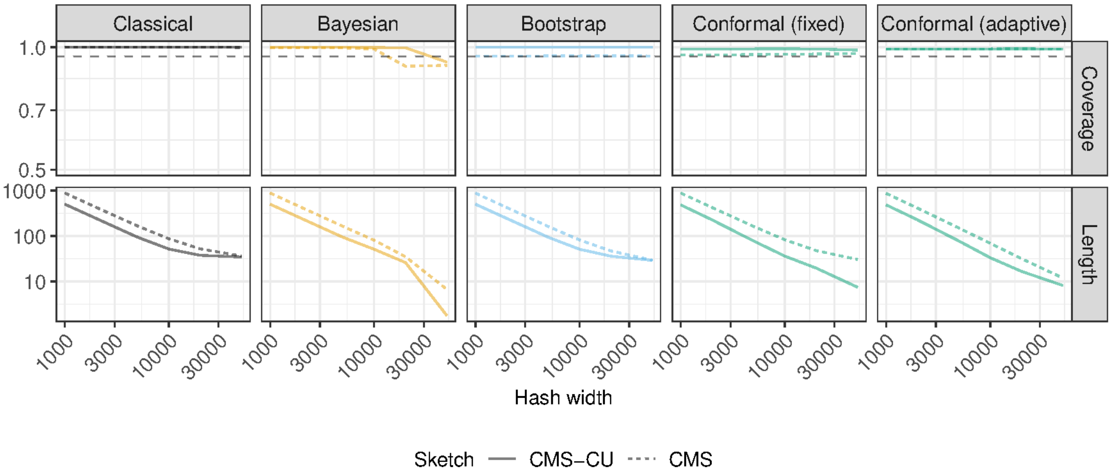

Figure A35: Performance of 95% confidence intervals for random queries on SARS-CoV-2 sequence data. The data are sketched with either the vanilla CMS or the CMS-CU. The results are shown as a function of the hash width. Other details are as in Figure 8.

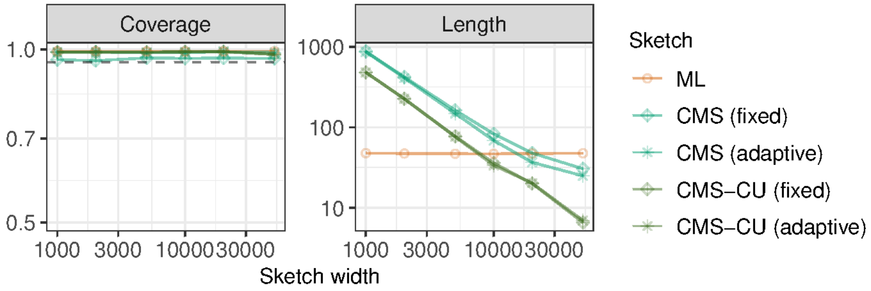

Figure A36: Performance of 95% confidence intervals computed by Algorithm 2 using different data sketches on SARS-CoV-2 sequence data, as a function of the sketch width. The results are shown as a function of the hash width. Other details are as in Figure 8.

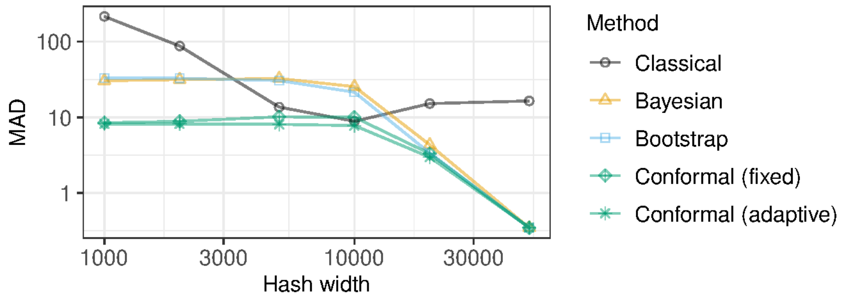

Figure A37: Median absolute deviation of point estimates for random queries on SARS-CoV-2 sequence data sketched with the CMS-CU. The results are shown as a function of the hash width. Other details are as in Figure 8.

### E.8 Illustration on English Literature Data

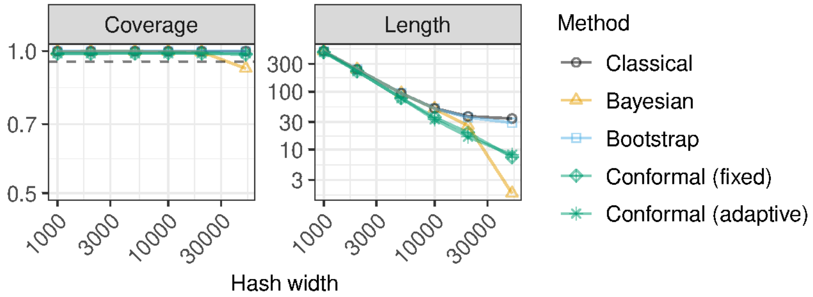

Figure A38: Performance of 95% confidence intervals for random queries, on a sketched data set of 2-grams in classic English literature, keeping only distinct queries. The coverage is the empirical proportion of distinct queries whose frequency is covered by the output confidence intervals. The conformal confidence intervals are computed by applying Algorithm 3 with $L = 5$ frequency bins. The data are sketched with the CMS-CU. Other details are as in Figure 8.

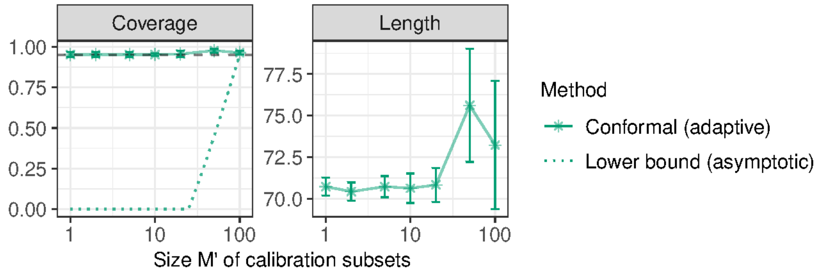

Figure A39: Performance on sketched SARS-CoV-2 data of confidence intervals for distinct queries in a test set of size $M = 100$, as a function of the parameter $M'$ of Algorithm 4 for constructing conformal confidence intervals satisfying (12). The hash width is $w = 5000$. Other details are as in Figure 8.

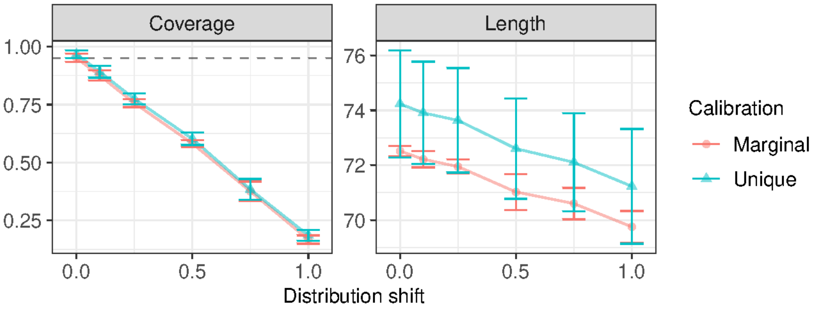

Figure A40: Performance on sketched SARS-CoV-2 data of conformal confidence intervals with marginal (Algorithm 2) or distinct-query (Algorithm 4) coverage in a test set of size 100 with varying degrees of distribution shift. Other details are as in Figure 8.

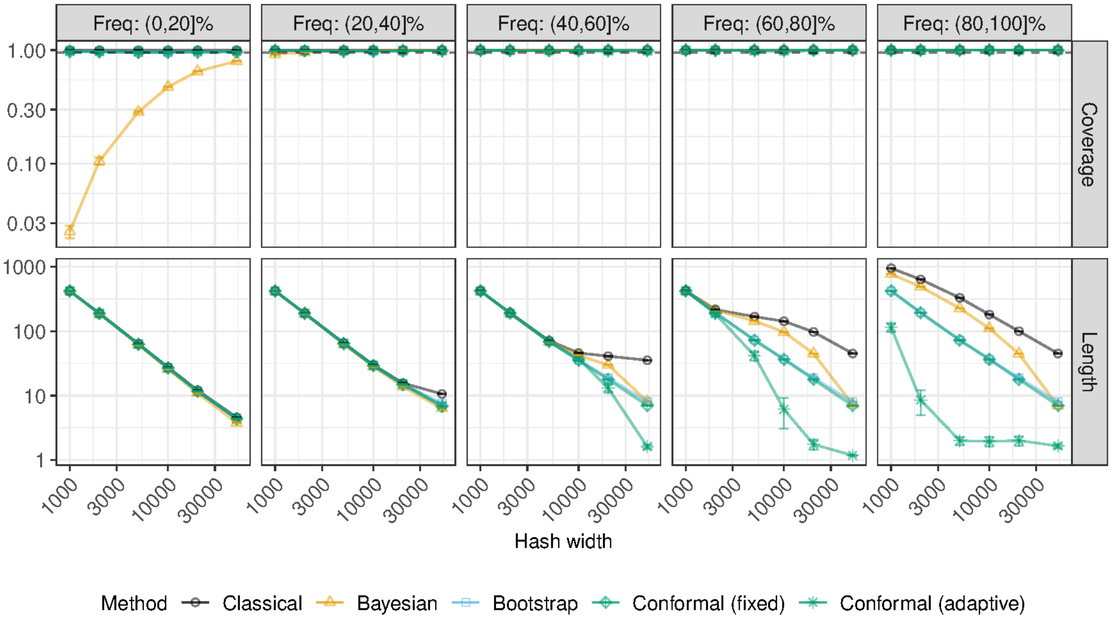

Figure A41: Performance of 95% confidence intervals for random queries on a data set of 2-grams in classic English literature, sketched with the CMS-CU. The results are are shown as a function of the hash width and stratified by the quintile of the true query frequency. Other details are as in Figure 9.

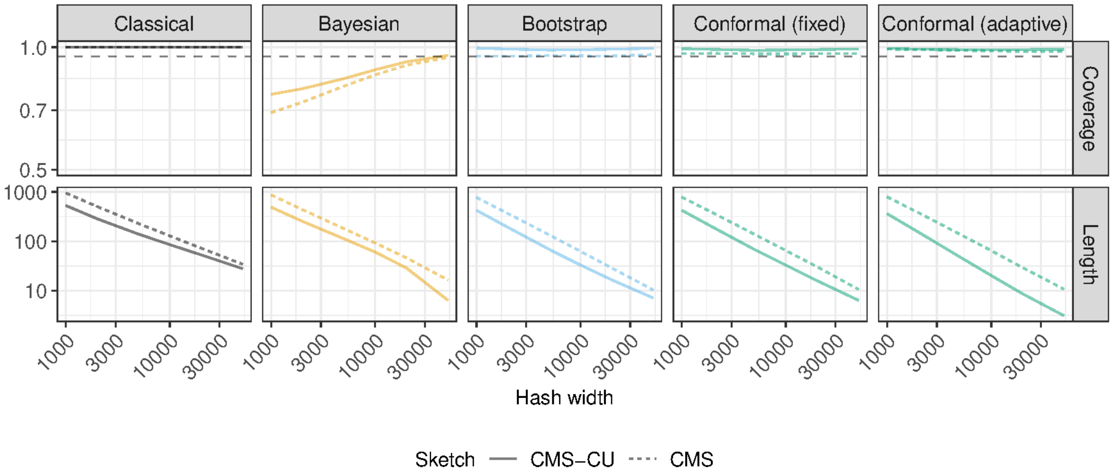

Figure A42: Performance of 95% confidence intervals for random queries on a data set of 2-grams in classic English literature. The data are sketched with either the vanilla CMS or the CMS-CU. The results are shown as a function of the hash width. Other details are as in Figure 9.

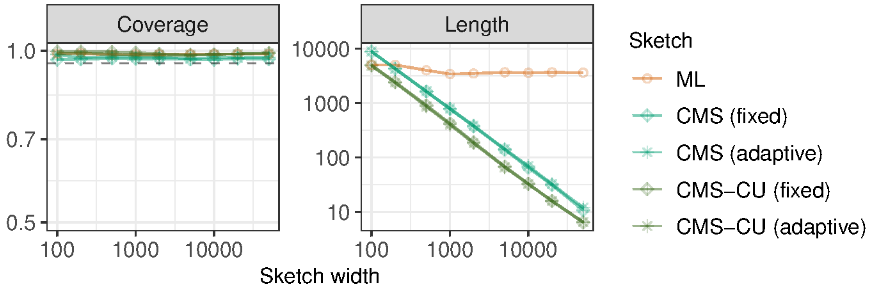

Figure A43: Performance of 95% confidence intervals computed by Algorithm 2 using different data sketches on a data set of 2-grams in classic English literature, as a function of the sketch width. The results are shown as a function of the hash width. Other details are as in Figure 9.

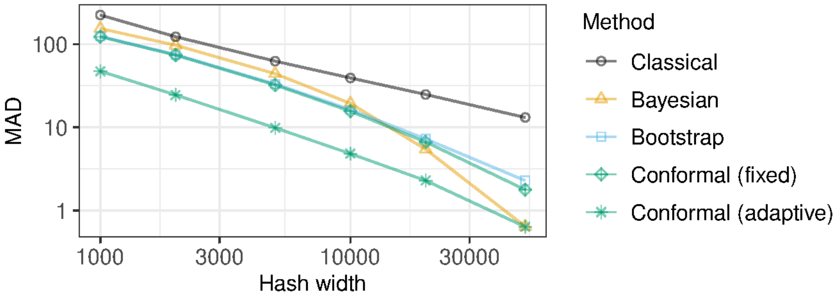

Figure A44: Median absolute deviation of point estimates for random queries on a data set of 2-grams in classic English literature, sketched with the CMS-CU. The results are shown as a function of the hash width. Other details are as in Figure 9.

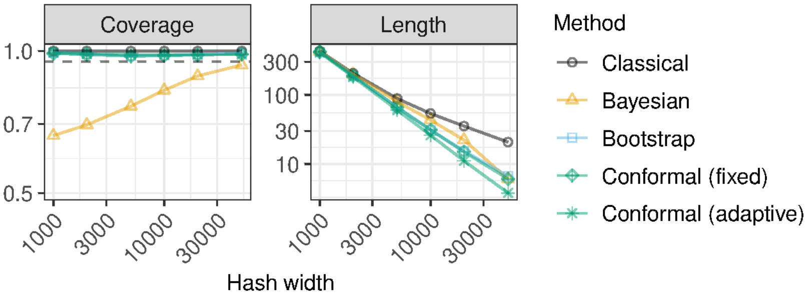

Figure A45: Performance of 95% confidence intervals for random queries, on a sketched data set of 2-grams in classic English literature, keeping only distinct queries. The coverage is the empirical proportion of distinct queries whose frequency is covered by the output confidence intervals. The conformal confidence intervals are computed by applying Algorithm 3 with $L = 5$ frequency bins. Other details are as in Figure 9.

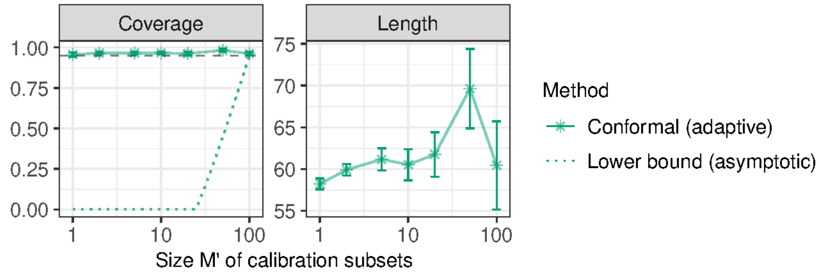

Figure A46: Performance on sketched English literature data of confidence intervals for distinct queries in a test set of size $M = 100$, as a function of the parameter $M'$ of Algorithm 4 for constructing conformal confidence intervals satisfying (12). The hash width is $w = 5000$. Other details are as in Figure 9.

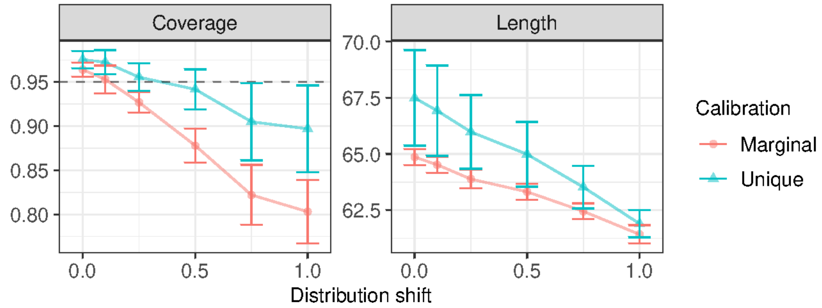

Figure A47: Performance on sketched English literature data of conformal confidence intervals with marginal (Algorithm 2) or distinct-query (Algorithm 4) coverage in a test set of size 100 with varying degrees of distribution shift. Other details are as in Figure 9.

| | | | 95% Lower bound | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | Conformal | |
| Data | Frequency | Upper bound | Classical | Bayesian | Bootstrap | Fixed | Adaptive |
| **SARS-CoV-2** | | | | | | | |
| AATTATTATAAGAAAG | 81 | 81 | 26 | **81** | 52 | 50 | 36 |
| TCAGACAACTACTATT | 76 | 76 | 21 | **55** | 47 | 45 | 32 |
| AAAGTTGATGGTGTTG | 73 | 73 | 18 | **59** | 44 | 42 | 31 |
| CAATTATTATAAGAAA | 63 | 63 | 8 | **48** | 34 | 32 | 26 |
| ATCAGACAACTACTAT | 60 | 60 | 5 | **44** | 31 | 29 | 26 |
| ACCTTTGACAATCTTA | 55 | 55 | 0 | **52** | 26 | 24 | 27 |
| ATTTGAAGTCACCTAA | 55 | 55 | 0 | **55** | 26 | 24 | 27 |
| CATGCAAATTACATAT | 54 | 54 | 0 | **54** | 25 | 23 | 26 |
| GAATTTCACAGTATTC | 54 | 54 | 0 | **54** | 25 | 23 | 27 |
| TTTGTAGAAAACCCAG | 53 | 53 | 0 | **53** | 24 | 22 | 27 |
| AGTTGCAGAGTGGTTT | 24 | 24 | 0 | 13 | 0 | 0 | **20** |
| TCTTCACAATTGGAAC | 24 | 24 | 0 | 12 | 0 | 1 | **20** |
| TTCTGCTCGCATAGTG | 24 | 24 | 0 | 12 | 0 | 0 | **20** |
| CTACTTTAGATTCGAA | 23 | 23 | 0 | 11 | 0 | 0 | **19** |
| GCTGGTGTCTCTATCT | 23 | 23 | 0 | **23** | 0 | 1 | 19 |
| TTCTAAGAAGCCTCGG | 23 | 24 | 0 | 14 | 0 | 0 | **20** |
| GGGCTGTTGTTCTTGT | 22 | 24 | 0 | 12 | 0 | 0 | **20** |
| ACGTTCGTGTTGTTTT | 20 | 20 | 0 | **20** | 0 | 0 | 16 |
| GAAGTCTTTGAATGTG | 20 | 20 | 0 | **20** | 0 | 0 | 16 |
| CAAACCTGGTAATTTT | 3 | 3 | 0 | **3** | 0 | 0 | 0 |
| **Literature** | | | | | | | |
| of the | 12565 | 12568 | 12513 | 12544 | 12557 | 12556 | **12562** |
| in the | 6188 | 6190 | 6135 | 6169 | 6179 | 6179 | **6180** |
| and the | 6173 | 6175 | 6120 | 6151 | 6164 | 6164 | **6165** |
| the of | 6015 | 6017 | 5962 | 5990 | 6006 | 6006 | **6007** |
| the lord | 4186 | 4195 | 4140 | 4165 | 4184 | 4184 | 4184 |
| to the | 3465 | 3467 | 3412 | 3445 | 3456 | 3456 | **3463** |
| the and | 2250 | 2251 | 2196 | 2227 | 2240 | 2240 | **2248** |
| all the | 2226 | 2230 | 2175 | 2207 | 2219 | 2219 | **2224** |
| and he | 2169 | 2173 | 2118 | 2153 | 2162 | 2162 | **2167** |
| to be | 2062 | 2064 | 2009 | 2043 | 2053 | 2053 | **2060** |
| man on | 22 | 29 | 0 | 10 | 18 | 18 | 18 |
| their hand | 22 | 24 | 0 | 9 | 13 | 13 | 0 |
| no need | 20 | 28 | 0 | 9 | 17 | 17 | 16 |
| and brother | 12 | 14 | 0 | 2 | 3 | 3 | 0 |
| miss would | 10 | 13 | 0 | **3** | 2 | 2 | 0 |
| i please | 8 | 12 | 0 | **3** | 1 | 1 | 1 |
| also how | 3 | 13 | 0 | 2 | 2 | 2 | 0 |
| in under | 3 | 9 | 0 | **2** | 0 | 0 | 0 |
| ten old | 3 | 6 | 0 | **1** | 0 | 0 | 0 |
| fault he | 1 | 9 | 0 | **1** | 0 | 0 | 0 |

Table A1: True frequencies, deterministic upper bounds, and 95% lower bounds for 10 common (top) and 10 rare (bottom) random queries in two sketched data sets. Sketching with CMS-CU with $w = 50,000$. Lower bounds written in green are below the true frequency; those in red are above. For each query, the highest lowest bound below the true frequency is highlighted in bold.

| | | | 95% Lower bound | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | Conformal | |
| Data | Frequency | Upper bound | Classical | Bayesian | Bootstrap | Fixed | Adaptive |
| **SARS-CoV-2** | | | | | | | |
| AATTATTATAAGAAAG | 81 | 209 | 0 | 4 | 0 | 0 | **18** |
| TCAGACAACTACTATT | 76 | 213 | 0 | 8 | 0 | 0 | **18** |
| AAAGTTGATGGTGTTG | 73 | 130 | 0 | 2 | 0 | 1 | **18** |
| CAATTATTATAAGAAA | 63 | 233 | 0 | 4 | 11 | 6 | **19** |
| ATCAGACAACTACTAT | 60 | 179 | 0 | 2 | 0 | 0 | **18** |
| ACCTTTGACAATCTTA | 55 | 292 | 0 | 15 | 70 | 67 | **22** |
| ATTTGAAGTCACCTAA | 55 | 258 | 0 | 11 | **36** | 31 | 20 |
| CATGCAAATTACATAT | 54 | 204 | 0 | 3 | 0 | 0 | **18** |
| GAATTTCACAGTATTC | 54 | 260 | 0 | 12 | **38** | 35 | 22 |
| TTTGTAGAAAACCCAG | 53 | 246 | 0 | 7 | **24** | 18 | 20 |
| ATGCTGCAATCGTGCT | 24 | 139 | 0 | 2 | 0 | 0 | **17** |
| ATTTCCTAATATTACA | 24 | 92 | 0 | 1 | 0 | 0 | **17** |
| CTCTATCATTATTGGT | 24 | 121 | 0 | 1 | 0 | 0 | **17** |
| TGTTTTATTCTCTACA | 24 | 199 | 0 | 3 | 0 | 1 | **19** |
| CAGTACATCGATATCG | 23 | 119 | 0 | 2 | 0 | 0 | **17** |
| TAATGGTGACTTTTTG | 23 | 92 | 0 | 1 | 0 | 0 | **17** |
| CAACCATAAAACCAGT | 22 | 105 | 0 | 1 | 0 | 0 | **17** |
| AGTTATTTGACTCCTG | 21 | 97 | 0 | 1 | 0 | 1 | **18** |
| ATAAAGGAGTTGCACC | 19 | 218 | 0 | 5 | 0 | 0 | **18** |
| **Literature** | | | | | | | |
| of the | 12565 | 12630 | 12086 | 12325 | 12463 | 12454 | **12563** |
| in the | 6188 | 6242 | 5698 | 5906 | 6075 | 6067 | **6096** |
| and the | 6173 | 6314 | 5770 | 5972 | 6147 | 6139 | **6169** |
| the of | 6015 | 6162 | 5618 | 5834 | 5995 | 5985 | **6014** |
| the lord | 4186 | 4289 | 3745 | 3975 | 4122 | 4114 | **4185** |
| to the | 3465 | 3558 | 3014 | 3217 | 3391 | 3380 | **3464** |
| the and | 2250 | 2413 | 1869 | 2081 | 2246 | 2237 | **2249** |
| all the | 2226 | 2346 | 1802 | 1993 | 2179 | 2170 | **2225** |
| and he | 2169 | 2293 | 1749 | 1937 | 2126 | 2117 | **2168** |
| to be | 2062 | 2121 | 1577 | 1770 | 1954 | 1945 | **2061** |
| very for | 15 | 59 | 0 | **2** | 0 | 0 | 0 |
| and faithful | 14 | 94 | 0 | **3** | 0 | 0 | 0 |
| but found | 9 | 74 | 0 | **2** | 0 | 0 | 0 |
| my speech | 6 | 98 | 0 | **3** | 0 | 0 | 0 |
| of eight | 5 | 66 | 0 | **2** | 0 | 0 | 0 |
| and soul | 4 | 140 | 0 | 6 | 0 | 0 | 0 |
| her prow | 3 | 79 | 0 | **2** | 0 | 0 | 0 |
| usual as | 2 | 56 | 0 | **2** | 0 | 0 | 0 |
| a invitation | 1 | 80 | 0 | 2 | 0 | 0 | 0 |
| angular log | 0 | 146 | 0 | 5 | 0 | 0 | 0 |

Table A2: True frequencies, upper and lower bounds for 10 common (top) and 10 rare (bottom) random queries in two sketched data sets. Hash width $w = 50,000$. Other details are as in Table A1.

## Appendix F. Additional Experiments with Two-Sided Confidence Intervals

This section describes additional experiments with synthetic data similar to those described in Figures 3 (Zipf distribution) and A23 (Pitman-Yor process prior), constructing two-sided instead of one-sided confidence intervals. For simplicity, we focus on one-sided 95% conformalized bootstrap confidence intervals based on the simpler Bonferroni approach described in Appendix B.1.1. The performance of these intervals is compared to those of one and two-sided standard bootstrap confidence intervals obtained with the method of Ting (2018).

Figure A29 reports on results based on data generated from a Zipf distribution and sketched with the CMS-CU, similarly to Figure 3. Here, all methods achieve the desired 95% marginal coverage level, but the conformal confidence intervals are shorter when the Zipf tail parameter $a$ is larger and hash collisions become rarer, consistently with Figure 3. It is interesting to note that the two-sided conformal confidence intervals are much narrower than their one-sided counterparts when $a$ is small and hash collisions are very common, but this is not true if $a$ is large. The latter is likely a limitation of the specific construction we have adopted, described in Appendix B.1.1, which may be too conservative in some cases due to the Bonferroni correction. A suitable implementation of the more sophisticated conditional histogram (Sesia and Romano, 2021) approach described in Appendix B.1.2 should be expected to produce two-sided intervals that are always narrower than their one-sided counterparts. Figure A30 reports on results similar to those in Figure A29, with the only difference that now the data are sketched with the vanilla CMS instead of the CMS-CU.

Figure A31 reports on results based on data generated from a Pitman-Yor process prior and sketched with the CMS-CU, similarly to Figure A23. Here, all methods achieve the desired 95% marginal coverage level, and two-sided intervals are generally much shorter than their one-sided counterparts. Across all values of $\sigma$, the conformal confidence intervals tend to be shorter than the bootstrap intervals, although this difference becomes very small in the case of two-sided intervals for large values of $\sigma$. Finally, Figure A32 reports on results similar to those in Figure A31, with the only difference that now the data are sketched with the vanilla CMS instead of the CMS-CU.

## References

Anders Aamand, Piotr Indyk, and Ali Vakilian. (Learned) frequency estimation algorithms under zipfian distribution. *arXiv preprint arXiv:1908.05198*, 2019.

Lada A Adamic and Bernardo A Huberman. Zipf's law and the internet. *Glottometrics*, 3 (1):143–150, 2002.

Anastasios Nikolas Angelopoulos, Stephen Bates, Michael I. Jordan, and Jitendra Malik. Uncertainty sets for image classifiers using conformal prediction. In *9th International Conference on Learning Representations, 2021, Virtual Event, Austria, May 3-7, 2021*, 2021.

Rina F Barber, Emmanuel J Candès, Aaditya Ramdas, and Ryan J Tibshirani. The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, 10(2):455–482, 2021.

Rina Foygel Barber, Emmanuel J Candès, Aaditya Ramdas, and Ryan J Tibshirani. Conformal prediction beyond exchangeability. *The Annals of Statistics*, 51(2):816–845, 2023.

Stephen Bates, Emmanuel Candès, Lihua Lei, Yaniv Romano, and Matteo Sesia. Testing for outliers with conformal p-values. *The Annals of Statistics*, 51(1):149–178, 2023.

Mario Beraha and Stefano Favaro. Random measure priors in Bayesian frequency recovery from sketches. *arXiv preprint arXiv:2303.15029*, 2023.

Dimitris Bertsimas and Vassilis Digalakis. Frequency estimation in data streams: Learning the optimal hashing scheme. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009.

Burton H Bloom. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426, 1970.

Diana Cai, Michael Mitzenmacher, and Ryan P Adams. A Bayesian nonparametric view on count-min sketch. In *Advances in Neural Information Processing Systems 31*, pages 8782–8791, 2018.

Emmanuel Candès, Lihua Lei, and Zhimei Ren. Conformalized survival analysis. *Journal of the Royal Statistical Society Series B*, 85(1):24–45, 2023.

Yukun Cao, Yuan Feng, and Xike Xie. Meta-sketch: A neural data structure for estimating item frequencies of data streams. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6916–6924, 2023.

Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. In *International Colloquium on Automata, Languages, and Programming*, pages 693–703. Springer, 2002.

Victor Chernozhukov, Kaspar Wüthrich, and Yinchu Zhu. Distributional conformal prediction. *Proceedings of the National Academy of Sciences*, 118(48), 2021.

Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. Power-law distributions in empirical data. *SIAM review*, 51(4):661–703, 2009.

Graham Cormode and Shan Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. *Journal of Algorithms*, 55(1):58–75, 2005.

Graham Cormode and Ke Yi. *Small summaries for big data.* Cambridge University Press, 2020.

Graham Cormode, Somesh Jha, Tejas Kulkarni, Ninghui Li, Divesh Srivastava, and Tianhao Wang. Privacy at scale: Local differential privacy in practice. In *Proceedings of the 2018 International Conference on Management of Data*, pages 1655–1658, 2018.

Edgar Dobriban and Sifan Liu. Asymptotics for sketching in least squares regression. In *Advances in Neural Information Processing Systems*, pages 3675–3685, 2019.

Emanuele Dolera, Stefano Favaro, and Stefano Peluchetti. A Bayesian nonparametric approach to count-min sketch under power-law data streams. In *International Conference on Artificial Intelligence and Statistics*, pages 226–234. PMLR, 2021.

Petros Drineas and Michael W Mahoney. RandNLA: randomized numerical linear algebra. *Communications of the ACM*, 59(6):80–90, 2016.

Robin Dunn, Larry Wasserman, and Aaditya Ramdas. Distribution-free prediction sets for two-layer hierarchical models. *Journal of the American Statistical Association*, 0(0):1–12, 2022.

Cristian Estan and George Varghese. New directions in traffic measurement and accounting. In *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 323–336, 2002.

Li Fan, Pei Cao, Jussara Almeida, and Andrei Z Broder. Summary cache: a scalable wide-area web cache sharing protocol. *IEEE/ACM transactions on networking*, 8(3):281–293, 2000.

S. Thomas Ferguson. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1:209–230, 1973.

Ramon Ferrer i Cancho and Ricard V Solé. Two regimes in the frequency of words and the origins of complex lexicons: Zipf's law revisited. *Journal of Quantitative Linguistics*, 8 (3):165–173, 2001.

Seymour Geisser. *Predictive inference: an introduction.* Chapman and Hall/CRC, 2017.

Amit Goyal and Hal Daumé. Lossy conservative update (LCU) sketch: Succinct approximate count storage. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 25, pages 878–883, 2011.

Amit Goyal, Hal Daumé III, and Graham Cormode. Sketch algorithms for estimating point queries in nlp. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 1093–1103, 2012.

Chirag Gupta, Arun K Kuchibhotla, and Aaditya K Ramdas. Nested conformal prediction and quantile out-of-bag ensemble methods. *Pattern Recognition*, 127, 2022.

Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

Eneida L Hatcher, Sergey A Zhdanov, Yiming Bao, Olga Blinkova, Eric P Nawrocki, Yuri Ostapchuck, Alejandro A Schäffer, and J Rodney Brister. Virus variation resource–improved response to emergent viral outbreaks. *Nucleic acids research*, 45(D1):D482–D490, 2017.

Xuming He. Quantile curves without crossing. *The American Statistician*, 51(2):186–192, 1997.

Alexander Henzi, Johanna F. Ziegel, and Tilmann Gneiting. Isotonic distributional regression. *Journal of the Royal Statistical Society Series B*, 83(5):963–993, 2021.

Chen-Yu Hsu, Piotr Indyk, Dina Katabi, and Ali Vakilian. Learning-based frequency estimation algorithms. In *International Conference on Learning Representations*, 2019.

Rafael Izbicki, Gilson T Shimizu, and Rafael B Stern. Flexible distribution-free conditional predictive bands using density estimators. *preprint at arXiv:1910.05575*, 2019.

Tanqiu Jiang, Yi Li, Honghao Lin, Yisong Ruan, and David P Woodruff. Learning-augmented data stream algorithms. In *International Conference on Learning Representations*, 2019.

Ramneet Kaur, Susmit Jha, Anirban Roy, Sangdon Park, Edgar Dobriban, Oleg Sokolsky, and Insup Lee. iDECODe: In-distribution equivariance for conformal out-of-distribution detection. *Association for the Advancement of Artificial Intelligence (AAAI)*, 2022.

Can Kockan, Kaiyuan Zhu, Natnatee Dokmai, Nikolai Karpov, M Oguzhan Kulekci, David P Woodruff, and S Cenk Sahinalp. Sketching algorithms for genomic data analysis and querying in a secure enclave. *Nature methods*, 17(3):295–301, 2020.

Kalimuthu Krishnamoorthy and Thomas Mathew. *Statistical tolerance regions: theory, applications, and computation.* John Wiley & Sons, 2009.

Jonathan Lacotte and Mert Pilanci. Optimal randomized first-order methods for least-squares problems. In *International Conference on Machine Learning*, pages 5587–5597. PMLR, 2020.

Jing Lei and Larry Wasserman. Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society: Series B*, 76(1):71–96, 2014.

Jing Lei, James Robins, and Larry Wasserman. Distribution-free prediction sets. *Journal of the American Statistical Association*, 108(501):278–287, 2013.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

Lihua Lei, Emmanuel J Candès, et al. Conformal inference of counterfactuals and individual treatment effects. *Journal of the Royal Statistical Society Series B*, 83(5):911–938, 2021.

Shuo Li, Xiayan Ji, Edgar Dobriban, Oleg Sokolsky, and Insup Lee. PAC-wrap: Semi-supervised PAC anomaly detection. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 945–955, 2022.

Ziyi Liang, Matteo Sesia, and Wenguang Sun. Integrative conformal p-values for powerful out-of-distribution testing with labeled outliers. *arXiv preprint arXiv:2208.11111*, 2022.

Sifan Liu and Edgar Dobriban. Ridge regression: Structure, cross-validation, and sketching. *International Conference on Learning Representations (ICLR) 2020*, 2019.

Michael W Mahoney. Randomized algorithms for matrices and data. *Foundations and Trends in Machine Learning*, 3(2):123–224, 2011.

Per-Gunnar Martinsson and Joel A Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403–572, 2020.

Dhruv Medarametla and Emmanuel Candès. Distribution-free conditional median inference. *Electronic Journal of Statistics*, 15(2):4625–4658, 2021.

Nicolai Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7:983–999, 2006.

Jayadev Misra and David Gries. Finding repeated elements. *Science of computer programming*, 2(2):143–152, 1982.

Lev Muchnik, Sen Pei, Lucas C Parra, Saulo DS Reis, José S Andrade Jr, Shlomo Havlin, and Hernán A Makse. Origins of power-law degree distribution in the heterogeneity of human activity in social networks. *Scientific reports*, 3(1):1783, 2013.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *European Conference on Machine Learning*, pages 345–356. Springer, 2002.

Sangdon Park, Edgar Dobriban, Insup Lee, and Osbert Bastani. PAC prediction sets under covariate shift. *International Conference on Learning Representations (ICLR) 2022*, 2021.

Sangdon Park, Edgar Dobriban, Insup Lee, and Osbert Bastani. PAC prediction sets for meta-learning. In *Advances in Neural Information Processing Systems*, 2022.

Guillaume Pitel and Geoffroy Fouquier. Count-min-log sketch: Approximately counting with approximate counters. *arXiv preprint arXiv:1502.04885*, 2015.

Jim Pitman and Marc Yor. The two parameter poisson-dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, 25:855–900, 1997.

Project Gutenberg. Project Gutenberg. `www.gutenberg.org`, 1971-present. Accessed: 2022-02-05.

Hongxiang Qiu, Edgar Dobriban, and Eric Tchetgen Tchetgen. Distribution-free prediction sets adaptive to unknown covariate shift. *Journal of the Royal Statistical Society Series B*, page qkad069, 07 2023.

John Riordan. *Introduction to combinatorial analysis.* Courier Corporation, 2012.

Yaniv Romano, Evan Patterson, and Emmanuel Candès. Conformalized quantile regression. In *Advances in Neural Information Processing Systems*, pages 3538–3548, 2019.

Yaniv Romano, Rina Foygel Barber, Chiara Sabatti, and Emmanuel Candès. With Malice Toward None: Assessing Uncertainty via Equalized Coverage. *Harvard Data Science Review*, 2(2), apr 30 2020a.

Yaniv Romano, Matteo Sesia, and Emmanuel Candès. Classification with valid and adaptive coverage. *Advances in Neural Information Processing Systems*, 33:3581–3591, 2020b.

Antonio Saavedra, Hans Lehnert, Cecilia Hernández, Gonzalo Carvajal, and Miguel Figueroa. Mining discriminative k-mers in dna sequences using sketches and hardware acceleration. *IEEE Access*, 8:114715–114732, 2020.

Mauricio Sadinle, Jing Lei, and Larry Wasserman. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114(525): 223–234, 2019.

Craig Saunders, Alexander Gammerman, and Volodya Vovk. Transduction with confidence and credibility. In *IJCAI*, 1999.

Stuart Schechter, Cormac Herley, and Michael Mitzenmacher. Popularity is everything: A new approach to protecting passwords from {Statistical-Guessing} attacks. In *5th USENIX Workshop on Hot Topics in Security (HotSec 10)*, 2010.

Henry Scheffe and John W Tukey. Non-parametric estimation. i. validation of order statistics. *The Annals of Mathematical Statistics*, 16(2):187–192, 1945.

Matteo Sesia and Emmanuel J Candès. A comparison of some conformal quantile regression methods. *Stat*, 9(1):e261, 2020.

Matteo Sesia and Stefano Favaro. Conformal frequency estimation with sketched data. *Advances in Neural Information Processing Systems*, 35, 2022.

Matteo Sesia and Yaniv Romano. Conformal prediction using conditional histograms. *Advances in Neural Information Processing Systems*, 34, 2021.

Qinfeng Shi, James Petterson, Gideon Dror, John Langford, Alex Smola, and SVN Vishwanathan. Hash kernels for structured data. *Journal of Machine Learning Research*, 10 (11), 2009.

James W. Taylor. A quantile regression neural network approach to estimating the conditional density of multiperiod returns. *Journal of Forecasting*, 19(4):299–311, 2000.

Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candès, and Aaditya Ramdas. Conformal prediction under covariate shift. *Advances in Neural Information Processing Systems*, 32, 2019.

Daniel Ting. Count-min: Optimal estimation and tight error bounds using empirical error distributions. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2319–2328, 2018.

John W Tukey. Non-parametric estimation ii. statistically equivalent blocks and tolerance regions–the continuous case. *The Annals of Mathematical Statistics*, pages 529–539, 1947.

John W Tukey. Nonparametric estimation, iii. statistically equivalent blocks and multivariate tolerance regions–the discontinuous case. *The Annals of Mathematical Statistics*, pages 30–39, 1948.

Santosh S Vempala. *The random projection method*, volume 65. American Mathematical Soc., 2005.

Vladimir Vovk. Cross-conformal predictors. *Annals of Mathematics and Artificial Intelligence*, 74(1-2):9–28, 2015.

Vladimir Vovk, David Lindsay, Ilia Nouretdinov, and Alex Gammerman. Mondrian confidence machine. Technical report, Royal Holloway, University of London, 2003.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer, 2005.

Vladimir Vovk, Ilia Nouretdinov, and Alex Gammerman. On-line predictive linear regression. *The Annals of Statistics*, 37(3):1566–1590, 2009.

Abraham Wald. An extension of Wilks' method for setting tolerance limits. *The Annals of Mathematical Statistics*, 14(1):45–55, 1943.

S. S. Wilks. Determination of sample sizes for setting tolerance limits. *The Annals of Mathematical Statistics*, 12(1):91–96, 1941.

David P Woodruff. Sketching as a tool for numerical linear algebra. *Foundations and Trends in Theoretical Computer Science*, 10(1–2):1–157, 2014.

Fan Yang, Sifan Liu, Edgar Dobriban, and David P Woodruff. How to reduce dimension with PCA and random projections? *IEEE Transactions on Information Theory*, 67(12): 8154–8189, 2021.

Qingpeng Zhang, Jason Pell, Rosangela Canino-Koning, Adina Chuang Howe, and C Titus Brown. These are not the k-mers you are looking for: efficient online k-mer counting using a probabilistic data structure. *PloS one*, 9(7):e101271, 2014.

George Kingsley Zipf. *Human behavior and the principle of least effort: An introduction to human ecology.* Ravenio Books, 2016.